Tableau Server on Linux
管理者ガイド

最終更新 2025/12/11

© 2025 Salesforce, Inc.

内容

Tableau Server のリリース ノート 1

展開の計画 3

サーバー展開計画の検証 3

サーバー管理者の概要 3

サーバー展開計画の検証 4

アーキテクチャの概要 4

Tableau およびデータ 6

ユーザー アクセス 7

ネットワークのどこに Tableau Server をインストールすればよいか。 7

サイジングと拡張性 8

Tableau Server 管理モデル 9

管理者の役割 9

Tableau Server 管理者 9

TSM 管理者 10

Tableau ポータル管理者 10

管理ツール 10

セキュリティ 11

Tableau Services Manager の概要 12

機能 12

コンポーネント 13

TSM 認証 13

カスタム PAM サービスの定義 14
TSM 認可グループ 14
TSM クライアントの接続 14
インフラストラクチャ計画 16
サーバー展開計画の検証 16
インストールの前に... 16
サーバー展開計画の検証 17
本番環境インストールに関するハードウェアの推奨事項 17
オペレーティング システム要件 20
インストール ディレクトリ 24
データディレクトリ 24
Tableau Prep Conductor 25
その他の要件 25
パッケージ要件 29
ディスク容量の要件 30
Tableau Server への抽出のパブリッシュ 31
抽出の更新 31
多数のワークブックの作成 31
ロギング 31
バックアップと復元のプロセス 31
復元のディスク空き容量の要件 33
サイトのエクスポートとインポート 33

推奨されるベースライン構成 33
サーバー展開計画の検証 34
本番環境インストールに関するハードウェアの推奨事項 34
ディスク領域の推定 36
ベースライン構成 36
単一サーバー インストール 36
推奨事項 36
サーバー構成 38
マルチノード インストール 40
2 ノード インストール - 抽出負荷の高い環境用 40
推奨事項 40
サーバー構成 42
2 ノード インストール - フロー環境用 45
サーバー構成 46
高可用性インストール (HA) 49
推奨事項 49
サーバー構成 49
仮想マシンおよびパブリック クラウド展開 52
ベースライン以上の構成 52
高 VizQL のピーク使用に関する考慮事項 53
災害復旧の考慮事項 53
Tableau Server のスケーラビリティ 54

アイデンティティストア 55
ローカル アイデンティティストア 55
外部のアイデンティティストア 55
LDAP バインド 56
GSSAPI (Kerberos) バインドを使用した LDAP 57
SSL 経由の LDAP 57
システム ユーザーとグループ 58
ユーザー 58
グループ 59
クライアントの認証 59
外部 ID ストアを使用した展開におけるユーザー管理 60
任意の LDAP ディレクトリ 60
サインイン時のユーザー バインド動作 60
Active Directory 60
Active Directory ユーザー認証と Tableau Server 61
Active Directory ユーザー名属性と Tableau Server 61
Active Directory からユーザーを追加する 62
ユーザー グループの追加 62
Active Directory からユーザーを削除する際の動作の同期 62
ドメインのニックネーム 63
複数ドメインのサポート 64
重複する表示名 64

NetBIOS 名を使用して Tableau Server にサインインする 65
Active Directory 展開のドメイン信頼要件 65
ドメインの信頼のためにバインド アカウントを複製する 66
一方向信頼関係シナリオでのライブ データへの接続 67
インターネットとの通信 67
Tableau がインターネットと通信する方法 68
Tableau Server のプロキシとロード バランサーの設定 72
誰がこの記事を読むべきか 72
フォワード プロキシ サーバーの構成 73
フォワード プロキシを使用するための Tableau Server on Linux の構成 73
サーバー クラッシュ レポーター 76
リバース プロキシとロード バランサーが Tableau Server と連携するしくみ 76
TLS/SSL 77
モバイル アクセス 77
リバース プロキシ、ロード バランサー、ユーザー認証 78
リバース プロキシ サーバーやロード バランサーと連携させるための Tableau Server の設定 79
Tableau Server と連携させるためのリバース プロキシ サーバーやロード バランサーの設定 82
リバース プロキシとロード バランサーの設定を検証する 84
関連トピック 85
展開 87
サーバー展開計画の検証 87

Tableau Server のインストールと構成 87
その他のインストール方法 88
サーバー展開計画の検証 88
はじめる前に 88
インストール手順 89
インストールの前に... 89
サーバー展開計画の検証 90
本番環境インストールに関するハードウェアの推奨事項 90
オペレーティング システム要件 93
インストール ディレクトリ 97
データディレクトリ 97
Tableau Prep Conductor 98
その他の要件 98
パッケージ要件 102
Tableau Server の最小ハードウェア要件と推奨事項 103
インストールの最小ハードウェア要件 104
本番環境の最小ハードウェア推奨事項 106
TSM のインストールと初期化 109
前提条件 109
オプションの初期化パラメーターの確認 110
Tableau Server のインストール 110
Tableau Server パッケージのインストール 111

TSM の初期化 113
次のステップ 116
Tableau Server のライセンス認証と登録 116
前提条件 116
TSM Web インターフェイスの使用 116
TSM CLI の使用 120
Tableau Server をライセンス認証する 120
Tableau Server の登録 120
次の手順 122
実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する 122
Tableau Server ATR の仕組み 123
ATR 期間 123
サーバー ATR 期間を設定または変更する 124
Tableau Server on Linux 125
Tableau Server in a Container 125
オフライン アクティベーションの ATR 期間 126
オフライン環境での ATR 期間の更新 127
サーバー ATR プロダクトキーを別の Tableau Server に移行する 127
別の Tableau Server で再利用するためにプロダクトキーを非アクティブ化する 128
バージョン 2021.4 以降でのプロダクトキーの非アクティブ化 128
バージョン 2021.3 以前でのプロダクトキーの非アクティブ化 128
オフラインでの Tableau Server の有効化 128

オフラインでのアクティブ化とログインベースのライセンス管理 (LBLM) 129

オフラインでの有効化と更新可能なサブスクリプション ライセンス (USL) 130

オフライン アクティブ化の概要 130

オフライン アクティブ化ファイル名の変更 131

TSM Web インターフェイスの使用 131

TSM CLI の使用 134

ステップ 1. Tableau サービス マネージャーへのログイン 134

ログインできない場合 135

ステップ 2. オフライン アクティブ化要求ファイルを生成する 135

ステップ 3. オフライン アクティブ化要求を Tableau アクティベーション Web サイトにアップロードする 135

ステップ 4. ライセンスを初期化または認証します 136

初期ノード設定の構成 137

前提条件 137

TSM Web インターフェイスの使用 137

アイデンティティストア設定 138

ゲートウェイ ポート 142

製品利用状況データ 142

サンプル ワークブックのインストール 143

インストールの初期化 143

TSM CLI の使用 144

アイデンティティストア設定の構成 144

ゲートウェイ設定の構成 (オプション) 145

製品利用状況データの構成 (オプション) 145
サンプル ワークブックのインストールの構成 (オプション) 145
保留中の構成の変更を適用する 146
LDAP 構成の確認 (オプション) 146
Tableau Server の初期化と起動 147
次のステップ 147
構成ファイルの例 147
エンティティとキー 148
サーバーの利用状況データ 149
利用状況データの共有を無効にする 150
インストール時に利用状況データの共有を無効にする 150
インストール後に利用状況データの共有を無効にする 150
TSM Web インターフェイスの使用 150
TSM CLI の使用 151
製品基礎データ 151
個々のコンピューターで製品基礎データの共有を無効にする 152
エンタープライズ レベルで製品基礎データの共有を無効にする 153
管理者アカウントの追加 153
前提条件 154
Web UI を使用する 154
tabcmd CLI を使用する 155
次のステップ 155

インストールの検証 ... 156
前提条件 ... 156
PostgreSQL ドライバーのインストールとインストールの検証 ... 156
最初のノードのインストールの既定 ... 157
インストールのジャンプスタート ... 158
はじめる前に ... 158
ステップ 1: Tableau Server パッケージをインストールして Tableau Services Manager を起動する ... 159
ステップ 2: Tableau Server のライセンス認証と登録 ... 160
ステップ 3: ローカル アイデンティティ ストアの構成 ... 161
ステップ 4: インストールの仕上げ ... 161
ステップ 5: PostgreSQL ドライバーのインストール ... 162
ローカル ファイアウォールの構成 ... 162
シングル ノード構成 ... 163
マルチノード クラスタ構成 ... 164
はじめる前に ... 164
ステップ 1: 最初のノードの構成 ... 164
ステップ 2: 追加ノードの構成 ... 166
Tableau Server を未接続の (エア ギャップ) 環境にインストールする ... 167
前提条件 ... 168
Tableau Server を Linux を実行するエア ギャップ環境にインストールする ... 168
Tableau Server をエア ギャップ環境でライセンス認証する ... 171
オフライン アクティブ化の概要 ... 171

オフライン アクティブ化 ファイル名の変更 172

ステップ 1.Tableau サービス マネージャーへのログイン 173

ステップ 2.Tableau Server ライセンスの種類を確認する 173

ステップ 3 (サーバー ATR ライセンス タイプ) json コンテンツを生成し、アクティベーション ページにコピーします。 174

ステップ 3 (レガシー ライセンス タイプ) エア ギャップ状態のシステムからアクティベーション要求テンプレートにデータを転記します。 174

ステップ 4.ライセンスを初期化または認証します 176

エア ギャップ環境でマップを表示する 177

オフライン マッピング用に Tableau Server を構成する 177

Tableau Server のクローンを作成 177

セキュリティに関する考慮事項 177

クローン ペイロードの制約 178

clone コマンドを使用して Tableau Server のコピーを作成する 178

クローン ペイロードを作成する 178

クローン ペイロードを使用して Tableau Server のコピーを作成する 179

コンテナ 180

Tableau Server のマルチノード展開をコンテナで再作成する 180

Tableau Server in a Container 181

概要 181

Tableau Server in a Container に関する制限事項 182

Tableau Server in a Container の基本的なワークフロー 182

Tableau Server の機能に関する考慮事項 182

Active Directory 182
AD ドメイン コントローラーを設定する 182
AD 証明書を Tableau Server キーストアにインポートする 183
展開構成の例 184
Docker 184
Tableau Server in a Container の基本的な使用例 184
自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例 184
TSM 専用モード 184
マルチノードの基本的な使用例 185
最初のノード 185
追加ノード 185
データ使用の外部化 185
Init コンテナーの基本的な使用例 186
シングルノードバックアップからの基本的な復元 186
Docker の構成 186
Tableau Server in a Container - セットアップ ツールの使用 187
概要 187
Tableau Server in a Container に関する制限事項 188
Tableau Server in a Container のセットアップ ツール 188
構築でサポートされているディストリビューション 188
必要なファイルをダウンロードする 189
インストール 189

登録フォームを完成させる 189

ツールの使用方法 191

オーケストレーション 192

イメージをカスタマイズする 192

構築時に環境変数を設定する 192

環境ファイル 192

使用例: 192

環境変数 193

ドライバー、証明書、その他のファイル 194

Tableau 管理ビュー 195

カスタム ベース イメージを定義する 195

yum とpip の内部リポジトリを使用する 196

ベース イメージとセキュリティ 196

Tableau Server in a Container を強化する 197

展開構成の例 198

Docker 198

Tableau Server in a Container の基本的な使用例 198

自動化された初期管理ユーザーによるTableau Server in a Container の基本的な使用例 198

TSM 専用モード 198

マルチノードの基本的な使用例 198

最初のノード 198

追加ノード 199

データ使用の外部化 199
Init コンテナーの基本的な使用例 199
シングルノードバックアップからの基本的な復元 200
Docker の構成 200
Tableau Server in a Container - イメージの使用 201
概要 201
Tableau Server in a Container に関する制限事項 201
Tableau Server in a Container のイメージ 202
前提条件 202
configure-container-host スクリプトを実行する 202
イメージを実行する 203
基本的な実行引数の概要 203
Tableau Server 初期管理者の作成の自動化 204
ライセンス発行 204
コンテナーでのライセンス発行 204
ライセンス環境変数 205
ライセンス ファイル 205
要求されたライセンスのリース期間 206
初期化されていないイメージを実行する 206
イメージを操作する 207
TSM Web UI とリモート CLI 207
TSM リモートユーザーを作成する 208

TSM リモートユーザーのパスワードを設定する 208
シークレットファイル (推奨) 208
環境変数 209
TSM リモートユーザーのパスワードをローテーションする方法 209
Tableau Server in a Container を新たに起動する 209
実行中のコンテナー内でパスワードを手動でローテーションする 210
初期構成のオプション 210
ランタイム環境変数 210
読み取り専用環境変数 217
構築時の環境変数 219
Tableau Server 構成の上書き 220
初期化前および初期化後のコマンド 220
初期化前スクリプト 220
初期化後のスクリプト 220
手順 221
ユーザー構成 221
Tableau Server in a Container のユーティリティとツール 221
ファイル パーミッションの管理 221
パスワード管理 222
例 222
シークレットファイルの使用例 222
シークレットファイルの内容の例 223

例 223

Tableau Server を実行した後に構成する 223

ステータス 223

稼働状態チェック 223

準備完了チェック 224

データの永続化 224

Tableau Server データ 224

静的ホスト名 225

完全な例 225

バックアップと復元 225

Tableau Server コンテナーでのバックアップ 226

Tableau Server コンテナー内で復元する 227

前提条件 227

Tableau Server から Tableau Server in a Container に移行する 228

Tableau Server バージョンをアップグレードする 228

アップグレードイメージ メソッドによるアップグレード 228

例: 229

バックアップ復元 メソッドによるアップグレード 231

マルチノードの Tableau Server in a Container 231

マルチノードの基本的な使用例 231

最初のノード 231

追加ノード 232

ライセンス発行ポートとTSM ポートの公開 232

ホスト名の解決 233

追加ノードをブートストラップする 233

セキュリティに関する考慮事項 234

Bootstrap の有効期限 234

bootstrap ファイルを転送する 234

セキュリティで保護されたネットワーク経由でファイルを転送する 235

ネットワークファイル マウントを使用する 235

その他 235

追加ノードを起動する 235

追加ノードを構成する 235

Tableau Server の機能に関する考慮事項 236

Active Directory 236

AD ドメイン コントローラーを設定する 236

AD 証明書を Tableau Server キーストアにインポートする 236

展開構成の例 237

Docker 237

Tableau Server in a Container の基本的な使用例 237

自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例 237

TSM 専用モード 237

マルチノードの基本的な使用例 238

最初のノード 238

追加ノード 238
データ使用の外部化 238
Init コンテナーの基本的な使用例 239
シングルノードバックアップからの基本的な復元 239
Docker の構成 239
Tableau Server in a Container - トラブルシューティング 240
概要 240
制限事項 241
トラブルシューティング 241
インストールと初期化の失敗 241
失敗したインストールをデバッグする 242
Tableau Support とKubernetes 242
コンテナ内の Tableau Serverのタイムゾーンの変更 243
ログ 243
すべてのログを抽出する 243
Ziplogs 243
手動の tar コマンド 244
ログの操作とデバッグのヒント 244
コンテナーを起動する(初期/インストール) 244
サービスログ - サーバー ランタイム 246
停止したコンテナー 248
認証プロパティの設定に失敗しました 248

新しい起動時のエラー (Tableau Server が起動しない理由など) 249
既存のデータを使用してコンテナーを再起動または起動する際の障害 249
サーバーが Postgres (または他のプロセス) を起動しない 249
展開構成の例 250
Docker 250
Tableau Server in a Container の基本的な使用例 250
自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例 250
TSM 専用モード 250
マルチノードの基本的な使用例 250
最初のノード 250
追加ノード 251
データ使用の外部化 251
Init コンテナーの基本的な使用例 251
シングルノードバックアップからの基本的な復元 252
Docker の構成 252
Tableau Server in a Container のクイックスタート 253
概要 253
Tableau Server in a Container に関する制限事項 253
Tableau Server in a Container - クイックスタートガイド 254
はじめる前に 254
展開構成の例 258
Docker 258

Tableau Server in a Container の基本的な使用例 258

自動化された初期管理ユーザーによるTableau Server in a Container の基本的な使用例 258

TSM 専用モード 258

マルチノードの基本的な使用例 258

最初のノード 258

追加ノード 259

データ使用の外部化 259

Init コンテナーの基本的な使用例 259

シングルノードバックアップからの基本的な復元 260

Docker の構成 260

インストール後のタスク 261

セキュリティ強化チェックリスト 261

セキュリティ更新のインストール 261

1.現在のバージョンへの更新 261

2.有効な信頼された証明書を使用してSSL/TLS を構成する 262

3.古いバージョンの TLS の無効化 262

4.内部トラフィック用に SSL 暗号化を構成する 263

5.ファイアウォールによる保護を有効にする 263

6.サーバーコンピューターや重要なディレクトリへのアクセスを制限する 264

7.新しいシークレットとトークンの生成 264

8.使用していないサービスの無効化 265

JMX サービス 265

9.セッション存続時間設定の検証 265
10.ファイルベースのデータソース用にサーバーの許可リストを設定する 266
11.Web ブラウザー クライアントでの HTTP Strict Transport Security の有効化 267
12.ゲストアクセスの無効化 268
13.Referrer-Policy HTTP ヘッダーを 'same-origin' に設定する 269
14.TLS による SMTP 接続の構成 269
15.LDAP の SSL を設定 271
変更リスト 271
SMTP セットアップの構成 272
セキュア SMTP 272
TSM Web インターフェイスの使用 272
TSM CLI の使用 273
SMTP CLI 構成リファレンス 275
TLS 暗号 278
TSM におけるファイルとパーミッション 280
個別のファイルにパーミッションを設定する 282
ディレクトリのパーミッション設定 282
サーバー イベント通知の構成 283
TSM Web インターフェイスの使用 283
TSM CLI の使用 287
個別に通知の値を設定する 287
単一の json ファイルを使用してすべての通知の値を設定する 289

データキャッシュの構成 ... 290
データベース ドライバー ... 291
クラスタでのドライバーのインストール ... 291
サーバー クラッシュ レポーター ... 291
サーバー クラッシュ レポーターの構成 ... 292
TSM Web インターフェイスの使用 ... 293
TSM CLI の使用 ... 294
クラッシュ レポーターの設定 ... 294
構成テンプレート ... 294
構成ファイル リファレンス ... 295
Tableau Web 環境の管理者エリアの移動 ... 296
サイト ロールとサイト数に基づくアクセス ... 296
サーバー管理者 ... 297
サイト管理者 ... 299
サーバー管理者のタスク ... 300
サイト管理者のタスク ... 301
Tableau Server を別のドライブに移動させる ... 301
はじめる前に ... 301
Tableau Server 製品のプロダクト キーのライセンス認証 ... 303
分散型高可用性 Tableau Server インストール ... 304
サーバー展開計画の検証 ... 304
インストールのタイプ ... 305

外部リポジトリ 306
前提条件 307
ライセンス発行 307
分散 Tableau Server インストール環境の作成 307
高可用性 (HA) Tableau Server インストール環境の作成 308
最初のノードに障害が発生した場合 309
最初のノードの障害を復旧する 310
追加のノード上で調整サービス アンサンブルを構成する 310
追加のノードにクライアント ファイル・サービス (CFS) を追加する 310
Tableau Server サービス ライセンス チェック 311
分散環境の要件 311
ハードウェア 311
高可用性を得るためのハードウェアのガイドライン 312
マルチノード サーバーのソフトウェア要件 312
インストール場所 313
ネットワークとポート 313
ベストプラクティス 314
SSL 314
分散インストールの推奨事項 314
すべてのインストールでの推奨事項 315
追加のノードのインストールと構成 316
インストール場所 317

TSM Web インターフェイスの使用 ... 317
ノードbootstrap ファイルの生成 ... 317
追加ノードのインストールと初期化 ... 319
クラスタコントローラーを使用した追加ノードの構成 ... 321
TSM CLI の使用 ... 323
ノードbootstrap ファイルの生成 ... 323
追加ノードのインストールと初期化 ... 325
追加のノードを構成する ... 327
ドライバーのインストール ... 329
ドライバーと管理ビュー ... 329
データベースドライバー ... 329
クラスタでのドライバーのインストール ... 330
例: 3 ノードHA クラスタのインストールおよび構成 ... 330
単一サーバーシステム ... 330
3 ノードシステム ... 333
構成の手順 ... 333
はじめる前に ... 334
TSM Web インターフェイスの使用 ... 334
ステップ1: 最初のノードをインストールする ... 334
ステップ2: 追加ノード用の bootstrap ファイルを生成する ... 334
ステップ3: ノード2 をインストールして初期化する ... 336
ステップ4: ノード3 をインストールして初期化する ... 338

ステップ 5: 最小トポロジを使用して新しいノードを構成する 338
ステップ 6: 調整サービス アンサンブルを展開する 340
ステップ 7: 追加のノードでクライアント ファイル・サービス (CFS) を構成する 342
ステップ 8: ノード 2 のプロセスを構成する 343
ステップ 9: ノード 3 のプロセスを構成する 344
TSM CLI の使用 345
ステップ 1: 最初のノードをインストールする 345
ステップ 2: 追加ノード用の bootstrap ファイルを生成する 346
ステップ 3: ノード 2 をインストールして初期化する 347
ステップ 4: ノード 3 をインストールして初期化する 349
ステップ 5: プロセスを追加ノードに追加する 352
ステップ 6: 調整サービス アンサンブルを展開する 353
ステップ 7: 追加のノードでクライアント ファイル・サービス (CFS) を構成する 355
ステップ 8: ノード 2 のプロセスを構成する 356
ステップ 9: ノード 3 のプロセスを構成する 357
ステップ 10: ファイアウォールのルールを構成する (オプション) 358
ロード バランサーの追加 358
ロード バランサーの追加 359
ロード バランサーを使用するための Tableau Server の構成 360
調整サービス アンサンブルの展開 360
ハードウェア要件 360
調整サービス クォーラム 361

使用する調整サービスインスタンス数 361
新規調整サービス アンサンブルの展開 363
クライアント ファイル サービスの構成 365
追加ノードでの CFS の構成 366
リポジトリフェールオーバー 366
自動リポジトリフェールオーバー 367
手動リポジトリフェールオーバー 367
優先するアクティブなリポジトリ 368
初期ノードの障害からの回復 368
最初のノードに障害が発生した場合 369
一般的な要件 370
TSM コントローラー、ライセンス サービス、アクティブ化サービスを別のノードに移動する 370
ノードの障害からの回復 376
一般的な要件 377
障害が発生したノードの削除 377
ノードの構成 380
TSM Web インターフェイスの使用 381
TSM CLI の使用 383
ノードにプロセスを追加する 384
ノードのプロセス数を変更する 384
ノードからプロセスのインスタンスをすべて削除する 385
あるノードから別のノードにプロセスのすべてのインスタンスを移す 385

ノードロールによるワークロード管理 386
バックグラウンダー ノードロール 386
バックグラウンダー ノードロールの使用 387
構成オプション 387
ライセンスの要件 388
考慮事項 389
ファイル ストア ノードロール 390
抽出更新ワークロードとバックアップまたは復元ワークロードを最適化するためのガイドライン 391
抽出クエリワークロード管理の微調整 393
構成オプション 394
ライセンスの要件 395
ノードロールの表示方法 395
実行可能なユーザー 395
2 ノードクラスターに Tableau Server をインストールする 395
複数ノードTableau Server コンピューターの再起動 396
分散環境の管理 396
リポジトリプロセスの移動 397
TSM Web インターフェイスの使用 398
リポジトリの新しいインスタンスの追加 398
リポジトリのインスタンスの削除 399
TSM CLI の使用 399
リポジトリの新しいインスタンスの追加 400

リポジトリのインスタンスの削除 400
ファイル ストアのプロセスの移動 401
TSM Web インターフェイスの使用 401
ファイル ストアの 2 番目のインスタンスを追加する 401
ファイル ストアのインスタンスをコミッション解除および削除する 402
TSM CLI の使用 403
ファイル ストアの 2 番目のインスタンスを追加する 403
ファイル ストアのインスタンスをコミッション解除および削除する 404
メッセージング サービス プロセスの移動 405
TSM Web インターフェイスの使用 405
メッセージング サービスの の移動 405
TSM CLI の使用 406
メッセージング サービスの の移動 406
ノードの削除 407
ノードを削除するための前提条件 407
TSM Web インターフェイスの使用 408
TSM CLI の使用 409
調整サービス専用ノードで高可用性用を Tableau Server に設定する 409
前提条件 410
調整サービス専用ノードでアンサンブルを展開する 410
ロード バランサーの追加 413
ロード バランサーを使用するための Tableau Server の構成 414

Tableau Server のアップグレードの概要 414
コンピューターのオペレーティング システムのアップグレード 414
アップグレード パスを選択する 415
青/緑のアップグレード 415
インプレース アップグレード 415
アップグレードの準備 416
リリース ナビゲーター 416
サーバーのアップグレード- 最小ハードウェア推奨事項 417
サーバーのアップグレード- 変更された内容の確認 420
サーバーのアップグレード- 構成の詳細の収集 422
スクリーン ショットの取得 422
オブジェクト数の記録 422
ファイアウォール構成の記録 422
Tableau サービス マネージャー コントローラー証明書の有効期限の確認 423
アセット ファイルの収集 423
カスタムの構成情報の収集 424
セキュア SMTP 424
分析拡張機能 424
外部 リポジトリ 425
外部 ファイル ストア 426
ポートのカスタマイズ 426
サーバーのアップグレード- ライセンス ステータスの確認 426

サーバーのアップグレード- アカウントの確認 428
サーバーのアップグレード- Tableau Server のバックアップ 428
サーバーのアップグレード- セットアップのダウンロード 429
Tableau Server のアップグレードの仕組み 430
変更箇所 - アップグレード前に知っておく事柄 431
Tableau Server コンピューターの更新に関するベストプラクティス 431
2018.1 以降からのアップグレード(Linux) 432
Tableau Server のアップグレードに役立つサポートとサービス 434
サーバー アップグレード- スケジュールされたタスクの無効化 434
単一サーバーのアップグレード-- セットアップの実行 435
セットアップの実行 435
マルチノード アップグレード-- セットアップの実行 438
セットアップの実行 438
マルチノード アップグレード-- 各ノードでセットアップを実行する 439
セットアップの実行 439
マルチノード アップグレード-- アップグレード スクリプトの実行 440
アップグレード スクリプトの実行 440
Tableau Server のアップグレードの確認 441
Tableau Service プロセスの確認 441
Tableau サービス マネージャーのグローバル設定の確認 442
サブスクリプションとスケジュールの有効化 442
ユーザー アクセスの確認 443

パブリッシュされたワークブックの表示 443
ワークブックとデータソースのパブリッシュの確認 443
確認 Tableau Prep Builder 443
Tableau オブジェクトの数の確認 443
API 機能の確認 444
REST API 444
互換性テスト 444
アップグレード後のクリーニング 445
以前のバージョンをアンインストールする 445
Tableau Server のアップグレードにブルー/グリーン アプローチを使用する 446
アップグレードのテスト 450
テスト環境の準備 450
テスト環境のアップグレード 451
すべてが期待どおりに機能していることを確認する 451
パフォーマンスおよびユーザー受け入れテスト 452
新機能のテスト 452
アップグレードに関する通信 453
Tableau Server のインストールとアップグレードのトラブルシューティング 453
一般的なトラブルシューティングの手順 453
Tableau Server のインストールに関する全般的な問題 454
インストール ログの場所 454
インストールの試行が何度も失敗する 454

ハードウェア要件のためインストールが失敗する 455

CPU 要件が原因でインストールまたはアップグレード失敗する 455

Tableau Server のアップグレードに関する全般的な問題 456

アップグレードログの場所 456

アップグレード後にマップが表示されないか、完全に表示されません 456

アップグレードスクリプトエラー:"Tableau Server Version change validation failed. (Tableau Server のバージョン変更の検証に失敗しました。)" 456

マルチノードをアップグレードし、追加ノードを初期化すると"Enter your credentials again (認証資格情報を再度入力してください)" というエラーで失敗する 457

ディスク容量の不足によるアップグレードの失敗 457

RebuildSearchIndex ジョブでアップグレードが失敗する 458

2022.1 以降でアップグレードに失敗する 458

2020.4.0 以降でアップグレードが失敗する 459

バックアップ/復元ファイルの場所にパーミッションの問題があるため、アップグレードに失敗する 460

アップグレードは成功したが、パブリッシュ済みデータソースにアクセスできない 461

影響なし 461

詳細 461

共通設定のインポートに関する問題 462

設定ファイルをインポートすると、サービスが欠落しているために "not present on any node (ノードに存在しません)" 検証エラーが発生する 462

設定ファイルをインポートすると"configuration value you specified does not match" (指定した構成値が一致しません) エラーが発生する 463

エラー:「調整サービスのインスタンスを直接変更することはできません」 464

設定ファイルをインポートした後にこのエラーが表示される場合: 464

調整サービスのプロセスカウントを手動で設定する際にエラーが表示される場合: 465

Tableau Server の起動 466

Tableau Server は完全に起動したことを確認できません 466

Tableau Server が起動しない 466

Tableau index および Tableau Server (2023.3 以降) の再構築 466

インデックス作成/検索サーバーのインデックスを再構築することによって解決できる問題 466

Tableau Server の検索 & 参照 (2023.1 以前の場合) のインデックス再作成 467

検索とブラウズの再インデックスによって解決する他の問題 467

Tableau Server のライセンス認証 467

Tableau Server のライセンス認証の失敗 467

ライセンシング サーバーへアクセスできることを確認する 467

日付と時刻を確認します 469

プロダクトキーを強制的に再度読み取る 469

信頼されているストレージのコンテンツを Tableau サポートに送信する 470

tabcmd のインストールの問題 470

tabcmd を個別にインストールする 470

Linux で tabcmd をインストールときの問題 470

Java がインストールされていない 471

Java のバージョンが正しくない 471

Tableau Server のアンインストール 471

Tableau Server のアンインストールと完全な削除 472
Tableau Server パッケージのアンインストール 472
誤ってアンインストールした Tableau Server パッケージの再インストール 473
コンピュータから Tableau Server を削除する 474
tableau-server-obliterate の実行内容 475
Tableau Server のバックアップとログファイルを保持する 476
tableau-server-obliterate スクリプトの実行 476
サーバー ライセンスを削除せずに Tableau Server を完全に削除するには 477
Tableau Server とライセンスを完全に削除するには 477
tableau-server-obliterate スクリプトのヘルプ出力 478
出力 479
移行 480
Tableau Server から Tableau Cloud への移行 480
Tableau Server から Tableau Cloud に移行する際の技術的な考慮事項 480
概要 481
セキュリティ、管理、ガバナンス 483
セキュリティとコンプライアンス 483
ホスティングとアップグレード 483
サイトレベルの管理 484
可観測性データ 484
ビューの読み込みパフォーマンス 484
データ接続 485

ファイル 487
アプリケーションとデータベース 487
カスタム接続 487
キューブ データ ソース 488
データ準備 488
ライセンス発行とユーザー管理 488
ライセンス発行 488
シングル サインオン 488
Active Directory、Kerberos、LDAP 489
ユーザーとグループの自動管理 489
拡張性と外部インテグレーション 489
自動化ツール 489
拡張性 490
埋め込み 490
カスタマイズ 491
サーバーからサーバーへの移行 491
新しいハードウェアへの移行 491
Tableau Server 製品のプロダクト キーのライセンス認証 493
Tableau Server の Windows から Linux への移行 494
ステップ 1: 移行の計画を立てる 495
ステップ 2: バックアップを作成する 496
ステップ 3: Tableau Server on Linux をインストールして Windows のバックアップを復元する 496

ステップ 4: Tableau Server on Linux のテストを行う 497

ステップ 5: Tableau Server on Linux を本番環境にインストールし、Windows のバックアップを復元する 499

Tableau Server のオンプレミスのコンピューターからクラウド内の VM への移行 500

前提条件 500

Tableau Server を Cloud 内の VM に移行するには 500

アイデンティティストアの変更 503

警告 504

コンテンツとパーミッションの復元方法 505

ユーザー フィルター 506

ユーザー名と Tableau アイデンティティストア 506

方法 1: サイトのエクスポートとインポートを使用する 507

方法 2: フレッシュインストール — ユーザーがコンテンツを再パブリッシュする 507

バックアップを作成し、削除してから再インストール 508

ステップ 1: Tableau Server のバックアップ 508

ステップ 2: Tableau Server の削除 508

ステップ 3: 新しい認証タイプを使用して Tableau Server を再インストールする 508

個別のサイトの管理 509

サイトとは 509

サイト管理者のタスク 510

サイトの設定手順 511

サイトの計画 512

プロジェクト 513

ユーザーとグループ ... 513
サイト ロールとパーミッション ... 514
抽出更新スケジュール ... 515
サイト設定 リファレンス ... 515
サイト設定へのアクセス ... 515
[全般] タブ ... 517
[認証] タブ (Tableau Cloud) ... 529
[Bridge] タブ (Tableau Cloud) ... 530
[拡張機能] タブ ... 530
[統合] タブ ... 531
[接続済みアプリ] タブ ... 532
[Mobile] タブ ... 532
ユーザーとグループの管理 ... 534
サイトへのユーザーの追加 ... 534
ユーザー管理へのサイト管理者アクセス ... 535
ローカル ユーザーをサイトに追加する ... 535
Active Directory ユーザーをサイトに追加する ... 537
ローカル ユーザーの削除 ... 538
関連情報 ... 538
ユーザーのサイト ロールの設定 ... 538
ユーザー ライセンス、サイト ロール、コンテンツ パーミッションが連係するしくみ ... 539
ユーザーのサイト ロールの変更 ... 540

各サイトロールで許可されている一般的な機能 541
この記事で扱う内容とここで扱われない内容を探す場所 541
Tableau サイトロール 542
コンテンツをパブリッシュできる人物 545
サイトロールとActive Directory のインポートと同期 546
ユーザーの表示、管理、削除 548
ユーザーの認証タイプをSAML 用に設定する 554
注 555
ユーザーのインポート 555
CSV ファイルからのユーザーの追加 556
ユーザーのサイトロールの割り当てまたは維持方法 558
マルチサイト環境におけるサーバーレベルでのインポート 558
シングルサイト環境へのインポート 560
マルチサイトインポートとシングルサイトインポート 561
CSV インポートファイルのガイドライン 562
CSV ファイル形式の要件 562
CSV ファイルの必須の列構造 563
Tableau Server の列の詳細 564
アイデンティティプール 566
アイデンティティプールにユーザーを一括で追加 566
Tableau Server の CSV インポートの例 566
アイデンティティプールの例 567

tabcmd に渡される大きな CSV ファイルのパフォーマンス改善 568
CSV 設定とサイトロール 569
サイトユーザーの表示状態の管理 570
ユーザーの表示状態の制限 570
ユーザーの表示状態の制限に関するベストプラクティス 572
ユーザーの表示状態を[完全] に戻す 573
ゲストユーザー 573
ゲストユーザー パーミッション 573
ゲストアクセスの有効化または無効化 574
ゲストアカウントのその他の特徴 575
グループ セットの操作 575
グループ セットをオンにする 576
グループ セットを作成する 576
グループ セットのパーミッションを設定する 577
グループ 578
グループへのユーザーの追加 578
ユーザーをグループに追加 (ユーザー ページ) 579
ユーザーをグループに追加 (グループ ページ) 580
ローカル グループの作成 581
アサーションを使用した動的グループ メンバーシップ 582
ステップ 1: 設定を有効にする 583
ステップ 2: グループ メンバーシップの要求がアサーションに含まれていることを確認する 584

Active Directory 経由でのグループの作成 584

はじめる前に 585

AD からインポートしてグループを追加 585

サイト内の外部ディレクトリグループの同期 588

外部ディレクトリグループのユーザーに対して最小限のサイトロールを設定する 588

参照元の外部ディレクトリからユーザーが削除されるとどうなりますか。 590

参照元の外部ディレクトリ内でユーザー名を変更するとどうなりますか? 590

Tableau Server から外部ディレクトリグループが削除されるとどうなりますか? 591

サーバー上のすべての外部ディレクトリグループの同期 591

はじめる前に 591

外部ディレクトリグループのスケジュールに基づく同期 592

外部ディレクトリグループのオンデマンドでの同期 593

同期アクティビティの表示 593

外部ディレクトリグループのユーザーに対して最小限のサイトロールを設定する 594

参照元の外部ディレクトリからユーザーが削除されるとどうなりますか。 595

グループ同期のパフォーマンスの向上 595

すべての Active Directory グループをスケジュールに従って同期する 596

1 同期に対する最小限のサイトロールを設定する 596

2 スケジュールの設定 597

3 オンデマンドでの同期の実行 (オプション) 598

4 同期タスクのステータスを表示する 598

サインイン時にライセンスを付与 599

サインイン時にロールを付与 600
"サインイン時にロールを付与" が有効になっているユーザー ロールを変更する 603
"サインイン時にロールを付与" の影響を受けるユーザーを削除する 603
グループの削除 604
グループを削除する影響 604
グループ セット内のグループ 604
グループ セットの操作 604
グループ セットをオンにする 605
グループ セットを作成する 606
グループ セットのパーミッションを設定する 606
ダッシュボード ベースのカスタム ポータル 608
ポータル デザインのスケッチを描く 608
ロゴおよびナビゲーション エレメント用の画像を収集する 609
ダッシュボードのレイアウト テキスト、イメージ、および選択されたシート 609
ダッシュボード エレメントをコンテンツにリンクさせる 609
ポータルのパブリッシュ、テストおよび微調整 611
コンテンツ アクセスの管理 611
サイトの Web 作成アクセスと機能の設定 611
サイトに対する Web 作成をオンまたはオフにする 612
注 612
どのサイトで Web 作成を許可するかを確認する 612
クロスデータベース結合について 613

コンテンツの Web 編集、保存、およびダウンロード アクセスを設定する 613
ユーザーのサイトでの直接作業を許可する理由 614
Web 作成の是非 614
ユーザーに役立つパーミッションの管理でコンテンツの急激な拡散を防ぐ 615
適切なレベルのアクセス権のための編集、保存権限とサイト ロールを調整する 615
サイト ロール アクセス 615
マネージド セルフサービスのためのプロジェクト、グループ、グループ セット、パーミッションの設定 616
展開を計画する 617
クローズド パーミッション モデルを使用する 617
必要なプロジェクトとグループのタイプを特定する 617
サイト ロールを考慮する 619
グループとグループ セットを作成する 620
複数のグループのメンバーシップ 620
グループ セットの影響 621
あいまいな点が残るパーミッションを削除し、既定のパーミッション パターンを確立する 621
パーミッション ルールを作成する 622
プロジェクトを作成し、パーミッションを調整する 623
コンテンツ パーミッションをロックする 624
可能なプロジェクト構造 624
サーバー上で自由にコラボレーションできる共有されたワークブック 625
編集不可能な共有レポート 626

アナリストが接続する吟味されたデータソース 626
非アクティブなコンテンツ 627
ワークブックテンプレートのソース 627
次のステップ 628
プロジェクトを使用したコンテンツへのアクセスの管理 629
プロジェクトを使用する理由 630
プロジェクト階層を作成する場合 (例) 631
プロジェクトツリーを使用してプロジェクト間を移動する 631
サイトを使用しない理由 632
プロジェクトレベルの管理 632
プロジェクトを追加してそこにコンテンツを移動させる 633
トップレベルまたは子 (ネストされた) プロジェクトを追加する 633
プロジェクトツリーを使用してプロジェクト間を移動する 635
アセットを別のプロジェクトに移動する 635
プロジェクトの移動がパーミッションに与える影響 636
プロジェクトを削除する 637
アセットを移動するための要件 637
必要なサイトロール 638
コンテンツの移動先プロジェクトで必要となるパーミッション 638
コンテンツの移動元プロジェクトで必要となるパーミッション 638
プロジェクトイメージの追加 638
プロジェクトイメージの設定 639

サイトユーザーにコンテンツへのアクセスを要求できるようにする 642
既定の設定 644
プロジェクトパーミッションの構成 645
プロジェクトパーミッションの変更 645
コンテンツのパーミッションの変更 647
コンテンツのパーミッションの設定 647
ビューのパーミッションの設定 649
パーミッション 649
パーミッションの基本 650
パーミッションの設定 651
プロジェクトレベルのパーミッション 652
すべてのコンテンツタイプにプロジェクトパーミッションを設定する 653
アセットのパーミッションの設定を構成する 654
コンテンツレベルのパーミッション 654
アセットのパーミッションの設定 655
ビューのパーミッションの設定 656
パブリッシュのパーミッションの設定 656
すべてのユーザーグループをクリーニングする 657
特定のシナリオでのパーミッション設定 658
保存、パブリッシュ、上書き 658
パブリッシュする場所 658
上書きして所有権を要求する 658

Web 編集とWeb 作成 659
必要なパーミッションの機能設定 660
パブリッシュ済み Tableau データソースでのデータアクセス 660
他のユーザーによる抽出更新を許可する 662
コンテンツの移動 663
メトリクス (2024.2 で廃止) 663
メトリクスは所有者のパースペクティブからデータを表示 664
「データの説明を見る」 665
シートタブの表示または非表示 666
タブ付きビューをオフにして、独立したビューのパーミッションを許可する 667
コレクション 668
パーミッションの機能とテンプレート 668
テンプレート 668
パーミッションのコピーと貼り付け 669
機能 669
プロジェクト 669
ビューのテンプレート 669
パブリッシュのテンプレート 670
ワークブック 670
ビューのテンプレート 670
探索のテンプレート 670
パブリッシュのテンプレート 671

管理のテンプレート 672
ビュー 672
データソース 673
ビューのテンプレート 673
探索のテンプレート 673
パブリッシュのテンプレート 673
管理のテンプレート 674
その他のアセットタイプ 674
プロジェクトを使用したパーミッションの管理 677
プロジェクトの管理 677
特別なプロジェクト 679
プロジェクトリーダーの設定 679
アセットパーミッションのロック 680
アセットパーミッションの設定 (プロジェクトのロック) 682
アセットパーミッションの変更 683
プロジェクトとコンテンツの移動 684
Tableau コンテンツと外部アセットの移動 684
プロジェクトの移動 685
コレクション 686
プライベートコレクション 687
有効なパーミッション 687
パーミッションルールの評価 688

複数レベルで設定されたパーミッションの評価 690
ビューに対するパーミッション 691
パーミッション、サイトロール、ライセンス 692
サイトロールで許可される最大の機能 693
プロジェクト 693
ワークブック 693
データソース 695
データの役割 696
フロー 696
「データに聞く」レンズ 697
メトリクス (2024.2 で廃止) 698
コレクション 699
仮想接続 699
クイックスタート: パーミッション 700
プロジェクトに対してグループのパーミッションルールを作成する 700
1.ユーザーをグループに追加する 700
2.プロジェクトレベルのパーミッション設定にアクセスする 701
3.パーミッションルールを作成する 701
4.ユーザーの有効なパーミッションを表示する 702
サイトロール 702
パーミッションの論理 703
コンテンツ所有権の管理 705

コンテンツ タイプごとに、所有権を変更したり受け取ったりできる人物 705
コンテンツ所有権の変更に関する考慮事項 707
コンテンツ リソースの所有者の変更 708
外部資産でのパーミッションの管理 710
Tableau Catalog によるコンテンツと資産のインデックス化 710
Tableau Catalog のしくみ 712
メタデータに対するパーミッション 712
メタデータへのアクセス 712
Tableau コンテンツに対するパーミッション 713
派生パーミッションを使用する外部資産に対するパーミッション 713
個別の外部アセットに対するパーミッションの設定 716
系列情報にアクセスする 720
実行可能なユーザー 724
データの管理 728
Tableau Server データ ソース 728
データ ソースの管理 729
制限事項 731
.hyper 形式への抽出のアップグレード 732
.tde ファイルのサポート終了 732
Tableau Desktop を使用した .tde 抽出の手動アップグレード 732
ライブ接続を使用した .tde の手動アップグレード 733
サイトのタイム ゾーンを抽出用に設定する 733

Web 上での抽出の作成 733
Web 作成で抽出を作成する 734
Web 作成で埋め込みデータソースを抽出する 734
抽出設定を定義する 735
物理テーブル オプションを使用する条件 736
増分更新を設定する 739
詳細設定を使用する 740
抽出更新時の考慮事項 741
制限事項 741
コンテンツ サーバーで抽出を作成する 742
コンテンツ サーバーでパブリッシュされたデータソースを抽出する 742
コンテンツ サーバーで埋め込みデータソースを抽出する 743
制限事項 743
抽出されたデータを最新に保つ 744
抽出の監視と管理 744
データソース属性の表示 744
名前別にデータソースを表示 745
接続のリストの表示 745
データを最新に保つ 746
更新タスクの管理 746
関連項目 747
定期的なデータ更新 747

クイックスタート: スケジュールに基づく抽出の更新 749
1 サーバー上でのスケジュールの設定 749
2 スケジュールに基づく抽出の更新と失敗メールを有効にする 749
3 抽出を含むワークブックのパブリッシュ 750
4 更新パフォーマンスの監視 751
更新タスクの自動化 752
抽出の更新アラートの処理 752
抽出更新の問題の解決 754
非アクティブなワークブックとデータソースの抽出更新を自動的に一時停止する 755
抽出更新の一時停止機能の構成 756
通知 756
一時停止された抽出更新の再開 756
Tableau Server での接続の編集 757
Google、Salesforce、および WDC データの認証タイプ 759
Google 認証オプション 760
Salesforce.com 認証オプション 760
進行状況の監視 762
キューブ データソース 763
Tableau Server での Web データコネクタ 763
Tableau Server でコネクターを実行する前に 764
セーフリスト内のコネクタの管理 765
WDC セーフリストの更新にはサーバーの再起動が必要 765

コネクタをセーフリストおよび二番目のセーフリストに追加する 765
WDC または WDC 抽出更新の許可または却下 766
セーフリストから1 つ以上の WDC を削除します。 767
セーフリストにあるすべての WDC のリスト 767
コネクターの抽出の更新 767
トラブルシューティング 768
Web データコネクタのテストおよび入念な検査 768
ソースの調査 769
隔離された環境での Web データコネクタのテスト 769
Web データコネクタが生成したトラフィックを監視します。 769
Web データコネクタのパフォーマンスとリソース使用のテスト 770
Tableau Catalog の有効化 770
Catalog を有効にする前 770
必要なバージョン 771
Catalog を有効にする場合の期待値 771
初期取り込み 771
初期取り込み速度 771
メタデータを格納するためのディスク領域 772
非インタラクティブなマイクロサービス コンテナーのメモリ 772
Catalog を有効にするためのベストプラクティス 773
Catalog を有効にする手順の概要 773
Catalog の有効化 774

ステップ 1: Tableau Server 上のコンテンツ量を決定する 774
ステップ 2: 初期取り込みにかかる時間を見積もる 774
ステップ 3: 初期取り込みの時間を短縮する 775
ステップ 4: データ管理 ライセンスを認証する 777
ステップ 5 (オプション): 各サイトの Catalog の機能をオフにする 777
ステップ6:tsm maintenance metadata-services コマンドを実行する 777
ステップ 7: 初期取り込みの進行状況を監視し、その状態を検証する 778
ステップ 8: SMTP セットアップを構成する 778
ステップ 9 (オプション): 各サイトの Catalog 機能をオンにする 778
Catalog のトラブルシューティング 779
「Timeout limit and node limit exceeded (タイムアウト制限およびノード制限を超過しました)」メッセージ 779
コンテンツが見つからない 780
初期取り込み後のパフォーマンス 781
メモリ不足エラー 781
Catalog の無効化 781
各サイトの Catalog 機能をオフにする 781
メタデータのインデックス作成を停止する 782
初期取り込みステータスの取得 782
ステップ 1: REST API を使用して認証する 782
ステップ 2: GET 要求を実行する 782
応答のステータス値 783
応答例: 784

イベントステータスの取得 ... 787
ステップ 1: REST API を使用して認証する ... 788
ステップ 2: GET 要求を実行する ... 788
応答のステータス値 ... 788
応答例: ... 788
インパクト分析での系列の使用 ... 789
系列の移動 ... 789
埋め込みアセットが外部アセットに表示される ... 792
系列とカスタム SQL 接続 ... 793
Catalog ではキューブはサポートされていません ... 793
系列カウントとタブカウントの不一致 ... 794
ワークブックカウントの不一致の例 ... 794
メールを使用して所有者に連絡する ... 795
データラベル ... 795
ラベルを付けることができるアセット ... 796
ラベル名とカテゴリ ... 796
ラベルのカテゴリ ... 797
認定 ... 797
データ品質警告 ... 798
機密度ラベル ... 798
カスタムラベルのカテゴリ ... 799
データラベルが表示される場所 ... 799

データ ラベル ダイアログ 803

アセットのデータ ラベルを操作するために必要なパーミッション 805

データ ラベルとタグの比較 806

ユーザーが信頼できるデータを見つけられるように認証を使用する 807

認証はどのように信頼できるデータをユーザーから見つけやすくするか 807

認定するデータの選出ガイドラインを作成する 809

データを認定できる人物 809

データの認定方法 810

認証をカスタマイズする 812

データ品質に関する警告の設定 812

データ品質に関する警告について 813

データ品質に関する警告が表示される場所 814

視認性 816

サブスクリプションのデータ品質に関する警告 817

データ品質に関する警告の設定方法 817

データ品質に関する警告を削除する 820

品質に関する警告の監視をオンにするには 821

品質に関する警告の監視をオフにするには 822

抽出更新とフロー実行の失敗をサイト全体で監視 823

サイト全体の監視と明示的な監視の関係 823

データ品質に関する警告を設定できるユーザー 824

データ品質に関する警告のカスタマイズ 824

機密度ラベル 824
アセットへの機密度ラベルの設定 825
アセットからの機密度ラベルの削除 826
機密度ラベルが表示される場所 827
視認性 828
メール サブスクリプションの機密度ラベル 829
秘密度ラベルを設定できる人 829
機密度ラベルのカスタマイズ 829
カスタム カテゴリのラベル 830
カスタム カテゴリのラベルをアセットに設定する 831
Tableau Cloud および Tableau Server 2024.2 以降の場合 831
Tableau Server 2023.3 の場合 832
カスタム カテゴリのラベルをアセットから削除する 834
Tableau Cloud および Tableau Server 2024.2 以降の場合 834
Tableau Server 2023.3 の場合 835
カスタム カテゴリのラベルが表示される場所 835
カスタム カテゴリラベルを追加できる人 836
カスタム カテゴリのラベルをカスタマイズする 836
データ ラベルの管理 837
ラベル マネージャー 837
データ ラベルのプロパティ 840
名前 840

カテゴリ ... 841
説明 ... 842
視認性 ... 843
データラベルを作成する ... 843
ラベル作成の制限事項 ... 844
データラベルを編集する ... 844
ラベル編集の制限事項 ... 845
データラベルを削除する ... 845
ラベル削除の制限事項 ... 846
組み込みデータラベルをデフォルトに戻す ... 846
データラベルのカテゴリを作成する ... 846
ラベル カテゴリ作成の制限事項 ... 847
データラベルのカテゴリを編集する ... 847
ラベル カテゴリ編集の制限事項 ... 848
データラベルのカテゴリを削除する ... 848
カスタマイズのシナリオ ... 848
シナリオ: 組み込みデータラベルをカスタマイズする ... 848
シナリオ: カスタムのデータラベルを作成する ... 848
シナリオ: データラベルの新しいカテゴリと、関係づけるデータラベルを作成する 848
Tableau Server でダッシュボードとViz の拡張機能を管理する ... 849
Tableau Server で拡張機能を実行する前に ... 850
Tableau の信頼できる拡張機能 ... 850

ネットワーク対応拡張機能 851
既定の拡張機能設定 851
制御に関する拡張機能の使用 852
サーバーでの拡張機能を有効にするグローバル設定の変更 852
サイトの既定の設定の変更 853
許可リストの使用 853
拡張機能の URL を確認する 854
Tableau Exchange から 854
マニフェストファイルから 854
[詳細] ダイアログ ボックスを使用してダッシュボード拡張機能を確認する 855
許可リストへの拡張機能の追加と拡張機能の構成 856
拡張機能にワークブックまたはビューにアクセスするパーミッションがあることを確認する 857
特定の拡張機能のブロック 857
許可リストの URL で正規表現を使用する 857
ネットワーク対応拡張機能のセキュリティテスト 859
ソースファイルの調査 859
データアクセスの理解 860
隔離された環境での拡張機能のテスト 861
ダッシュボードの拡張機能により作成されたトラフィックの監視 861
分析拡張機能を使用した接続の構成 861
Server SSL 862
分析拡張機能を有効にする 862

分析拡張機能設定の構成 ... 863
分析拡張機能接続の編集または削除 ... 865
クライアント要件: Rserve 外部サービスの中間証明書チェーン ... 865
スクリプトエラー ... 865
分析拡張機能の使用状況の判定 ... 866
テーブル拡張機能 ... 866
利点 ... 866
前提条件 ... 867
テーブル拡張機能の作成 ... 867
テーブル拡張機能と分析拡張機能 ... 869
テーブル拡張機能 ... 869
分析拡張機能 ... 870
Einstein Discovery 統合の構成 ... 870
Einstein Discovery ダッシュボードの拡張機能 ... 870
Einstein Discovery の分析拡張機能 ... 871
Einstein Discovery Tableau Prep の拡張機能 ... 872
外部アクション ワークフロー統合の構成 ... 872
エディション、サイトロール、パーミッションの要件 ... 873
外部アクションの導入要件 ... 873
外部アクションをオンまたはオフにする ... 874
TSM CLI の使用 ... 874
サイトレベルの設定の変更 ... 874

Tableau とSlack ワークスペースの統合 874
Tableau Server サイトを Slack ワークスペースと統合する 875
ステップ 1: Tableau App for Slack を作成する 876
ステップ 2: Tableau サイトに OAuth クライアントを追加する 877
ステップ 3: 接続を完了する 877
Tableau サイトを Slack から切断する 878
Tableau App for Slack を更新する 878
Tableau App for Slack のトラブルシューティング 879
Creators: Web 上のデータへの接続 879
[データに接続] ページを開く 880
Tableau Server 881
[このサイト上] のデータへの接続 881
ファイルへの接続 881
コネクタの使用 882
Tableau Server コネクタ 882
Tableau Catalog サポートされるコネクタ 884
Tableau Cloud 884
[このサイト上] のデータへの接続 884
ファイルへの接続 884
コネクタの使用 884
Tableau Cloud コネクタ 885
Tableau Catalog サポートされるコネクタ 886

ダッシュボード スターターを使用する ... 886
Tableau Public ... 887
ファイルへの接続 ... 887
コネクタの使用 ... 887
Tableau Public コネクタ ... 887
接続後 ... 888
Web 作成でデータを最新に保つ ... 888
初期 SQL の実行 ... 888
初期 SQL を使用するには ... 889
初期 SQL ステートメントのパラメーター ... 891
サーバーに対する実行の保留 ... 892
セキュリティと偽装 ... 893
MySQL および Oracle 接続で「表を作成する」際のトラブルシューティング ... 893
MySQL 接続で、初期 SQL を使用してテーブルを作成してもテーブルが一覧表示されない ... 893
Oracle 接続で、初期 SQL を使用してテーブルを作成すると Tableau が停止する ... 893
Web 上でのフローの作成と操作 ... 894
サイトでのフローの Web 作成をオンまたはオフにする ... 894
リンクされたタスクを有効にする ... 895
フロー パラメーターを有効にする ... 896
Tableau Prep Conductor を有効にする ... 897
[今すぐ実行] を有効にする ... 898

フロー サブスクリプション ... 898
Tableau Prep 拡張機能を有効にする ... 899
自動保存をオフまたはオンにします ... 900
Web 上の Tableau Prep ... 900
インストールと展開 ... 901
サンプル データと処理の制限 ... 901
Web 上で利用可能な機能 ... 902
自動保存と下書き作業 ... 904
Web上でフローをパブリッシュする ... 905
認証資格情報の埋め込み ... 905
フローのパブリッシュ ... 907
実行可能なユーザー ... 907
Web 上でのビューの作成とデータの探索 ... 908
アラートとサブスクリプション ... 908
データ接続のために保存された認証資格情報の管理 ... 908
保存された認証資格情報を使用した接続のテスト ... 909
保存済み認証資格情報の更新 ... 909
保存されたすべての認証資格情報のクリア ... 910
保存済み認証資格情報の削除 ... 910
パーソナル スペースでプライベート コンテンツを作成および編集 ... 911
パーソナル スペースのプライバシー ... 911
Tableau Catalog とパーソナル スペース ... 911

コラボレーション ツール 912
パーソナル スペースでの抽出更新 912
パーソナル スペースのコンテンツを検索 912
パーソナル スペースへのワークブックのパブリッシュ 914
Tableau Server または Tableau Cloud のパーソナル スペースへのワークブックのパブリッシュ 914
Tableau Desktop からパーソナル スペースへのワークブックのパブリッシュ 914
ワークブックをパーソナル スペースに移動 915
パーソナル スペースからワークブックを移動 916
複数テーブルのデータ分析に関係を使用する 916
Tableau データ モデル 920
データ モデルのレイヤー 922
[データモデル] について 923
論理テーブルへの説明の追加 925
新しいモデルの構築 926
複数テーブルのモデル 927
単一テーブルのモデル 928
他のテーブルを含む単一テーブルのモデル 928
対応しているデータ モデルのスキーマ 929
単一テーブル 929
スター スキーマとスノーフレーク スキーマ 930
複数のテーブルでメジャー含むスタースキーマとスノーフレーク スキーマ 931
マルチファクト分析 932

データモデルにおけるリレーションシップの要件 934
関連付けたテーブルの利点が限られる要因 934
関係と結合の違い 935
リレーションシップと結合の特性 936
関係 937
結合 937
リレーションシップの使用要件 937
関連付けたテーブルの利点が限られる要因 938
結合について 938
パフォーマンス オプションを使用してリレーションシップ クエリを最適化する 940
カーディナリティと参照整合性の設定の意味 941
カーディナリティ オプション 941
参照整合性 オプション 941
結合について 942
パフォーマンス オプションの使用に関するヒント 944
用語の定義 945
マルチファクト関係データモデルについて 945
関連性のレベル 946
例 948
フィールド レベルの関連性指標 949
ワークシート上の関連性指標 949
関連性に関する警告ダイアログ ボックス 950

データモデルにおけるテーブル レベルの関連性 951
関連のないテーブル 952
関連テーブル 952
共有テーブル 953
分析におけるフィールドレベルの関連性 953
関連フィールド 953
関連のないフィールド 954
ディメンションのつなぎ合わせ 954
まだ関連付けられていないフィールド 955
あいまいに関連しているフィールド 956
共有テーブルからのあいまいなメジャー 957
フィールド間の不明確な関係を解決する 957
つなぎ合わせ vs 不明瞭さの解決 959
関連性の各レベルにおける結合の使用方法 960
関連ディメンションで内部結合を使用する 963
関連のないディメンションでクロス結合を使用する 964
つなぎ合わされたディメンションで外部結合を使用する 966
中間結果が外部結合される 966
メジャーを保持するための追加の結合 968
関連するメジャー 969
関連のないメジャー 970
トラブルシューティング 971

マルチファクト関係データモデルを扱う際の考慮事項 971
解決された問題 972
2024.2 での既知の問題 973
マルチファクト関係モデルを使用するとき 974
関連のないテーブルをモデル化する機能を設けた理由 975
名前の由来 975
マルチファクト関係データモデルを使用するとき 976
マルチファクト分析 976
その他のシナリオ 978
基底テーブルを特定する 979
基底テーブルと共有テーブルの特徴 980
代わりに追加の基底テーブルを試す 981
マルチファクト関係データモデルのツールヒント 981
フィールドレベルの関連性 981
データモデルを理解する 982
関連のないディメンションとディメンションのペア 984
ディメンションのつなぎ合わせ 984
関連のないディメンションとメジャーのペア 986
まだ関連付けられていない 987
あいまいに関連する 989
共有テーブルからのメジャー 989
すべてのディメンションが常に互いに関連していない 990

一部のディメンションが互いに関連している 990
関連するメジャー 991
フィルター 992
メジャーバリューの計算方法についての補足説明 993
例 994
メジャーの値はディメンション要素に追随します 994
マルチファクト関係データモデルの構築 997
モデルの構築 997
モデルの探索 999
用語 999
関係ツリーの特定 1001
関係の詳細を表示 1001
関係の選択 1002
基底テーブルとの交換 1002
例 1003
基底テーブルの折りたたみ 1003
トラブルシューティング 1004
単一のデータソースの作成 1004
例 1005
循環の解決 1005
データモデルの制限事項 1006
循環 1006

ネストされた共有テーブル 1007
Web 画像をワークシートに動的に追加する 1007
データソースを準備する 1008
データセットの例: 1008
イメージの役割を URL に割り当てる 1009
[データソース] ページから割り当てる: 1009
ワークシートから割り当てる: 1009
ビジュアライゼーションに画像を追加する 1010
ビジュアライゼーションを共有する 1010
画像の接続に関するトラブルシューティング 1011
Viz に画像が表示されない 1011
一部の画像が Viz に表示されない 1012
ワークシートの外に画像が表示されない 1014
「データに聞く」(Ask Data) 機能を使用した自動的なビューの構築 1015
「データに聞く」レンズに移動する 1016
レンズ ページまたはダッシュボード オブジェクトから「データに聞く」 1018
レンズに移動して、データをもっと詳しく知る 1018
テキストを入力してクエリを作成する 1019
提案されたフレーズを追加してクエリを作成する 1021
フィールドとフィルターを追加してクエリを作成する 1022
クエリの要素がどのように適用されているかを確認する 1023
質問のフレーズの変更 1024

Viz タイプの変更 ... 1024
フィールド、フィルター、表示データを変更する ... 1025
日付フィルターの調整 ... 1027
時間の経過による差異の比較 ... 1030
単純計算を適用する ... 1031
他の Viz を含むシートを追加する ... 1032
メール、Slack、またはリンクを介して「データに聞く」(Ask Data) の Viz を共有する 1033
レンズ所有者にフィードバックを送信する ... 1033
クエリを正常に実行するためのヒント ... 1034
特定の対象者に焦点を当てた「データに聞く」のレンズを作成する ... 1034
Tableau サイトでレンズのページを作成または構成する ... 1035
推奨ビジュアライゼーションのリストを変更する ... 1038
推奨ビジュアライゼーションを追加または置換する ... 1038
セクションのタイトルと推奨ビジュアライゼーションの名前を編集するか、推奨を削除する ... 1038
ダッシュボードに「データに聞く」レンズを追加する ... 1039
ダッシュボードの「データに聞く」オブジェクトに別のレンズを適用する ... 1041
レンズの名前、説明、プロジェクトの場所を変更する ... 1041
レンズを通した「データに聞く」の利用状況を確認する ... 1042
ユーザーがレンズに関する質問をメールで送信できるようにします。 ... 1042
レンズのパブリッシュと表示のパーミッション ... 1043
サイトでの「データに聞く」(Ask Data) 機能の無効化または有効化 ... 1044
「データに聞く」(Ask Data) 機能でのデータの最適化 ... 1045

「データに聞く」(Ask Data) 機能でのデータの最適化 1046
データソースまたはレンズ レベルでの設定の変更 1046
フィールドの名前と値に対応する同義語の追加 1047
検索結果から特定のフィールド値を除外する 1047
データソースを最適化する 1048
「データに聞く」(Ask Data) のインデックス作成を最適化する 1048
データ抽出を使用してパフォーマンスを向上させる 1050
ユーザーがデータソースにアクセスできるようにする 1050
サポートされていないデータソース機能に注意する 1050
ユーザーの質問を予測する 1051
データを簡素化する 1051
フィールドの適切な既定の設定 1051
地理的フィールドとカテゴリー フィールドでの階層の作成 1052
Tableau データストーリーを作成する (英語のみ) 1052
データストーリーがデータを扱う方法を理解する 1053
データストーリーの書き方について学ぶ 1053
サイトのデータス トーリーを管理する 1054
ダッシュボードに Tableau データストーリーを追加する 1054
自分の Tableau データス トーリーに合った適切なストーリー タイプを選択する 1059
連続 1059
不連続 1060
全体のパーセント 1061

散布図 1062
Tableau データストーリーの構成設定 1063
Tableau データストーリーの設定: 分析 1063
ストーリー用に分析を設定する 1064
様々なタイプの分析を理解する 1064
相関性 1064
クラスタリング 1064
分布 1065
セグメント 1065
傾向線 1065
変動性 1066
ストーリー生成のために分析を使用する方法 1066
非連続的なストーリーのための分析を理解する 1066
非連続的なストーリーのための分析を理解する 1068
散布図ストーリー用に分析を理解する 1069
ストーリー全体の割合のための分析を理解する 1071
Tableau データストーリー設定の構成: 特性 1071
ディメンションを使用して特性を測定する 1072
メジャーの特性の詳細 1072
書式設定 1072
コンテンツ 1073
並べ替え 1074

Tableau データストーリー設定の構成: 表示 1074

ストーリーの表示を構成する 1074

ストーリー表示設定を使用するタイミングを理解する 1075

Tableau データストーリー設定の構成: ドライバー 1075

ディメンション ドライバーを設定する 1076

ディメンション ドライバーのタイプを理解する 1076

セカンダリ要因を使用する 1076

メトリクス ドライバーを設定する 1077

Tableau データストーリー設定の構成: ナラティブ 1078

詳細度を設定する 1079

ドリルダウンを設定する 1079

ディメンションの用語を追加する 1079

メジャー ラベルを管理する 1080

Tableau データストーリー設定の構成: 関係 1080

連続ストーリーまたは不連続ストーリーの [Actual vs. Benchmark (実際のデータとベンチマーク)] 関係を作成する 1081

[Current/Most Recent vs. Previous Period (現在/最新と前の期間)] 関係を作成する 1081

Tableau データストーリーのカスタマイズ 1082

独自のインサイトを追加する 1082

ヘッダーとフッターを追加する 1083

関数を追加する 1083

条件を追加する 1084

カスタム コンテンツを複製する ... 1085
ドリルダウン セクションにカスタム コンテンツを追加する ... 1086
Tableau データストーリーのカスタマイズ: コンテキスト変数 ... 1087
コンテキスト変数の設定 ... 1087
コンテキスト変数を使用するタイミング: 複数のメジャーの参照 ... 1089
コンテキスト変数を使用するタイミング: 期間ごとの分析 ... 1091
Tableau データストーリーのカスタマイズ: 関数 ... 1093
Average ... 1094
Count ... 1094
Difference ... 1095
DifferenceFromMean ... 1095
Direction ... 1095
Ending Label ... 1095
EndingValue ... 1095
Label ... 1096
LargestNegativeChangeDifference ... 1096
LargestNegativeChangeEndingLabel ... 1096
LargestNegativeChangeEndingValue ... 1096
LargestNegativeChangePercentDifference ... 1096
LargestNegativeChangeStartingLabel ... 1097
LargestNegativeChangeStartingValue ... 1097
LargestNegativePercentChangeDifference ... 1097

LargestNegativePercentChangeEndingLabel 1097
LargestNegativePercentChangeEndingValue 1097
LargestNegativePercentChangePercentDifference 1098
LargestNegativePercentChangeStartingLabel 1098
LargestNegativePercentChangeStartingValue 1098
LargestPositiveChangeDifference 1098
LargestPositiveChangeEndingLabel 1098
LargestPositiveChangeEndingValue 1099
LargestPositiveChangePercentDifference 1099
LargestPositiveChangeStartingLabel 1099
LargestPositiveChangeStartingValue 1099
LargestPositivePercentChangeDifference 1099
LargestPositivePercentChangeEndingLabel 1100
LargestPositivePercentChangeEndingValue 1100
LargestPositivePercentChangePercentDifference 1100
LargestPositivePercentChangeStartingLabel 1100
LargestPositivePercentChangeStartingValue 1100
LongestStreakDifference 1101
LongestStreakDirection 1101
LongestStreakEndingLabel 1101
LongestStreakEndingValue 1101
LongestStreakLength 1101

LongestStreakPercentDifference 1102
LongestStreakStartingLabel 1102
LongestStreakStartingValue 1102
MaxLabel 1102
MaxValue 1102
Median 1103
MinLabel 1103
MinValue 1103
PercentDifference 1103
PercentOfWhole 1103
PeriodLabel 1104
PeriodLabelNewest 1104
PeriodValue 1104
PeriodValueNewest 1104
Range 1104
SortAscendingLabel 1104
SortAscendingValue 1105
SortDescendingLabel 1105
SortDescendingValue 1105
StartingLabel 1105
StartingValue 1105
StartToFinishDifference 1106

StartToFinishPercentDifference 1106
StdDev 1106
Sum 1106
Total 1106
Value 1107
Z-Score 1107
Tableau データストーリーのカスタマイズ: コンテンツの非表示と並べ替え 1107
コンテンツとセクションを非表示にする 1107
セクション内のコンテンツを並べ替える 1108
Tableau データストーリーにデータを追加する 1109
非表示のシートを使用する 1109
ディメンションを連結する 1111
複数のデータストーリーを積み重ねる 1112
ダッシュボードに Tableau ポップアップ データストーリーを追加する 1112
Tableau データストーリーでカスタム メジャー関係を作成する 1114
Tableau データストーリーのパラメーターを更新する 1116
Tableau データストーリーで表計算を使用する 1118
「データの説明を見る」を使用して、インサイトをより早く発見する 1120
「データの説明を見る」へのアクセス 1121
「データの説明を見る」を使用して分析を強化する 1122
「データの説明を見る」を使い始める 1122
ダッシュボード、シート、マークで「データの説明を見る」を実行する 1123

説明を見るために必要な「データの説明を見る」のパーミッション 1125
「データの説明を見る」を使用するためのヒント 1125
説明を掘り下げる 1126
分析したフィールドを表示する 1126
説明での用語と概念 1128
「データの説明を見る」の説明タイプ 1131
基になる値を探索する 1131
基になる特性 1132
極値 1132
違いを視覚化する 1134
Null 値 1135
レコード数 1136
マークの平均値 1137
関連する単一の値 1138
上位の要因 1140
関連するディメンション 1140
関連するメジャー 1142
その他の探索項目 1143
その他の興味深いディメンション 1143
「データの説明を見る」の分析済みフィールド 1144
「データの説明を見る」によって分析されたフィールドを表示する 1145
「データの説明を見る」が統計分析に使用するフィールドを表示するには 1146

統計分析に使用されるフィールドを変更する ... 1148
「データの説明を見る」が統計分析に使用するフィールドを編集するには ... 1149
既定で除外されるフィールド ... 1151
「データの説明を見る」を使用する場合の要件と考慮事項 ... 1152
「データの説明を見る」の使用に最適な Viz とは ... 1153
「データの説明を見る」に最適なデータ ... 1153
「データの説明を見る」が使用できない場合 ... 1154
「データの説明を見る」へのアクセスの制御 ... 1155
「データの説明を見る」にアクセスできるユーザー ... 1155
「データの説明を見る」を使用できるユーザーとユーザーが表示できる内容の制御 ... 1156
編集モード ... 1156
表示モード ... 1157
「データの説明を見る」の説明のダイアログ ボックスを開く ... 1158
「データの説明を見る」の表示に説明タイプを含めるか除外する ... 1158
統計分析に使用するフィールドを含めるか除外する ... 1159
ユーザーがメールや Slack を介して説明を共有できるように Tableau を構成する 1161
「データの説明を見る」のしくみ ... 1162
「データの説明を見る」であるもの (および「データの説明を見る」ではないもの) ... 1162
説明の分析および評価方法 ... 1164
予測範囲とは ... 1164
分析に使用されるモデル ... 1165
サイトに対する「データの説明を見る」の無効化または有効化 ... 1167

ダッシュボードの拡張機能の使用 1168

拡張機能をダッシュボードに追加する 1168

ダッシュボードの拡張機能の構成 1169

ダッシュボードの拡張機能の再読み込み 1170

データセキュリティ、Tableau の信頼できる拡張機能、ネットワーク対応拡張機能 1170

ネットワーク対応拡張機能の許可または拒否 1171

Tableau Desktop で JavaScript が有効になっていることを確認する 1172

拡張機能が Tableau Cloud または Tableau Server で実行していることを確認する 1172

拡張機能にワークブックまたはビューにアクセスするパーミッションがあることを確認する 1173

サンドボックス拡張機能でサポートされる Web ブラウザー 1173

サンドボックス拡張機能でサポートされているバージョンの Tableau Server 1173

サードパーティーおよびパートナーが構築したダッシュボードの拡張機能についてのサポートを受ける 1173

アニメーションの書式設定 1174

同時および連続したアニメーションを理解する 1174

同時アニメーション 1174

連続したアニメーション 1175

ワークブックのビジュアライゼーションをアニメーション化する 1175

ワークブックのアニメーション設定をリセットする 1178

すべてのアニメーションを完全に無効にする 1178

軸のアニメーションの小数点を書式設定する 1179

アニメーションが再生されない理由 ... 1179
サーバー レンダリング ... 1179
サポートされていないブラウザーと機能 ... 1179
数字とNULL 値の書式設定 ... 1180
Tableau Desktop の場合 ... 1180
数値形式を指定する ... 1180
カスタム数値形式を定義する ... 1182
カスタム数値形式の例 ... 1183
カスタム数値形式に特殊文字を含める ... 1184
フィールドに既定の数値形式を設定する ... 1185
メジャーを通貨に書式設定する ... 1185
ロケールを使用して数値形式を指定する ... 1187
NULL 値の書式設定 ... 1188
Tableau Server または Tableau Cloud の場合 ... 1190
数値形式を指定する ... 1190
カスタム日付形式 ... 1193
カスタム日付形式フィールドを見つける方法 ... 1194
ビュー内の日付フィールドの書式設定 (Tableau Desktop) ... 1194
ビュー内の日付フィールドの書式設定 (Tableau Cloud および Tableau Server) ... 1195
[データ] ペイン内の日付フィールドの書式設定 (Tableau Desktop のみ) ... 1195
サポートされている日付形式記号 ... 1196
カスタム日付形式の例 ... 1199

日本の元号に基づいた日付形式のサポート 1200
日付形式でリテラル テキストを使用する 1202
抽出データソースでの DATEPARSE 関数の形式構文 1202
URL アクション 1206
URL アクションで Web ページを開く 1206
URL アクションを使用したメールの作成 1210
URL におけるフィールド値およびフィルター値の使用 1212
集計フィールドを含める 1213
パラメーター値を挿入する 1213
ごみ箱 1214
ごみ箱にはどのようなコンテンツが保存されますか? 1214
コンテンツ: 注意事項とベストプラクティス 1215
プロジェクト: 注意事項とベストプラクティス 1215
ごみ箱からコンテンツを復元する 1218
ごみ箱からコンテンツを削除する 1218
ごみ箱をオンにする 1218
ごみ箱をオフにする 1219
管理者に問い合わせる 1219
ビューまたはワークブックへのサブスクリプションの作成 1219
自分または他のユーザー向けのサブスクリプションの設定 1220
サブスクリプションの更新またはサブスクライブ解除 1223
一時停止したサブスクリプションの再開または削除 1225

関連項目 ... 1225
カスタム ビューの使用 ... 1226
カスタム ビューに関する注意事項 ... 1226
カスタム ビューの作成 ... 1227
カスタム ビューの検索 ... 1227
ビューから ... 1227
ワークブックから ... 1227
デフォルトのカスタム ビューの設定 ... 1228
カスタム ビューの共有 ... 1228
カスタム ビューの削除 ... 1229
削除時の注意 ... 1229
カスタム ビューの管理 ... 1230
カスタム ビューを含むコンテンツの安全な変更 ... 1230
ビューを Salesforce にパブリッシュする ... 1231
前提条件 ... 1231
ビューを Salesforce にパブリッシュする ... 1231
Salesforce にパブリッシュしたビューを表示できるユーザー ... 1232
Tableau Lightning Web コンポーネントとトークン認証によるシングル サインオン (SSO) の設定 ... 1232
信頼できる URL を追加する ... 1233
Tableau LWC のシームレス認証を有効にする ... 1233
Salesforce 設定を構成する ... 1233
Tableau 設定を構成する ... 1235

ユーザー属性の取得を許可する(Tableau Cloud のみ) 1236

ホストマッピングを設定または編集する 1236

新しいホストマッピングを作成する 1236

ホストマッピングを編集する 1237

ユーザー属性クレーム定義を作成する 1237

Lightning アプリケーション ビルダーを使用して Tableau LWC を Lightning ページに追加する 1238

Tableau LWC を Lightning ページに追加する 1239

ページを保存してアクティブ化する 1239

複数の Tableau ビューの埋め込み 1239

モバイル向け Tableau LWC シングル サインオン 1239

トークン認証を使用した Tableau LWC SSO のトラブルシューティング 1240

保存された Tableau ホストマッピングのホストタイプを更新する 1240

Salesforce と Tableau の構成を検証する 1241

JWT トークンを検証する 1241

ページの有効化を確認する 1242

Tableau ビュー LWC がシームレスな認証なしで動作していることを確認する (Tableau ビュー LWC のみ) 1242

エラー: LWC コンポーネントのバージョンはサポートされなくなりました (Tableau View LWC のみ) 1242

エラー: Tableau Pulse LWC を有効にするには、Salesforce 管理者に連絡して Tableau のシームレス認証を設定してください (Tableau Pulse LWC のみ) 1243

関連項目 1243

Tableau Lightning Web コンポーネントでユーザー属性関数 (UAF) を活用する 1243

Salesforce での Tableau UAF クレーム定義 1244
定義の作成 1244
Tableau Lightning Web コンポーネントページに定義を追加する 1245
ホーム ページ、アプリページ、およびレコードLightning ページ 1245
Experience Cloud ページとユーティリティバー 1246
同じページに複数のコンポーネントがある場合の考慮事項 1246
定義の削除 1246
ホーム ページ、アプリページ、およびレコードLightning ページ 1247
Experience Cloud ページとユーティリティバー 1247
Tableau でデータを操作する 1247
どうぞ、クリックしても安全です 1247
1: Tableau サイトとは 1248
2: viz の検索 1248
3: コンテンツの操作 1250
詳細表示とデータの並べ替え 1250
データのフィルター 1251
元に戻す 1252
4: 最新化 1252
バックグラウンドマップの選択 1253
バックグラウンドマップの変更: 1254
Tableau Desktop での既定のバックグラウンドマップの変更 (非推奨の機能) 1254
オフライン バックグラウンドマップの使用 1255

メトリクスの作成とトラブルシューティング (廃止) 1256
従来のメトリクスの廃止 1256
サイト内のメトリクスを検索する 1258
メトリクスのコンポーネント 1259
タイムライン 1260
比較 1261
ステータス 1263
ビューからメトリクスを作成する 1264
マークを選択してメトリクスを定義する 1264
メトリクスの説明と設定 1265
メトリクスを完成させる 1267
メトリクスを上書きする 1268
メトリクスを作成できない場合 1268
メトリクスの構成を編集する 1269
メトリクスの更新方法 1270
失敗した更新の修正 1270
接続済みビューがまだ一覧にある場合 1272
接続されているビューが一覧にない場合 1272
一時停止された更新の再開 1272
Tableau Catalog でのメトリクスの表示 1273
パブリッシュされたデータにアクセスするための認証資格情報を設定する 1275
認証タイプの設定 1276

Dropbox、OneDrive 接続 ... 1277
Tableau データソースへのワークブックの接続 ... 1277
仮想接続 ... 1278
関連項目 ... 1279
データガイドでダッシュボードを探検する ... 1279
データガイドを作成者としてカスタマイズする ... 1279
データガイドをダッシュボードのユーザーとして探索する ... 1280
データガイドをさまざまなレベルで探索する ... 1281
ダッシュボードレベルの詳細を理解する ... 1281
Viz レベルの詳細を理解する ... 1283
マークレベルの詳細を理解する ... 1284
データガイドの表示制御 ... 1286
クエリキャッシュとビューの高速化のデータ鮮度ポリシーの設定 ... 1286
クエリキャッシュのデータの鮮度を理解する ... 1286
ビューの高速化のデータの鮮度を理解する ... 1287
ワークブックに最適なデータを選択する ... 1287
ワークブックのデータ鮮度ポリシーを編集する ... 1287
動的な軸の範囲の使用 ... 1289
サポートされているフィールドタイプ ... 1290
選択内容に基づいた動的な軸範囲の設定 ... 1290
既存のパラメーターを使用した動的な軸の範囲の設定 ... 1291
制限とエッジ ケースについて ... 1291

動的な軸タイトルの使用 ... 1291
サポートされているフィールドタイプ ... 1292
動的な軸タイトルの設定 ... 1292
制限とエッジ ケースについて ... 1292
Dynamic Zone Visibility の使用 ... 1293
サポートされているフィールドタイプ ... 1293
動的なダッシュボードゾーンを構成する ... 1294
制限とエッジ ケースについて ... 1297
Microsoft Teams 用 Tableau アプリのセットアップ ... 1297
関連項目 ... 1297
前提条件 ... 1297
ユーザー マッピング属性 ... 1298
Microsoft Teams SDK の属性 ... 1298
Microsoft Entra プロファイルの属性 ... 1298
Tableau Server の追加の前提条件 ... 1299
Tableau Server 用 Tableau アプリのインストール ... 1299
1. アプリマニフェストをダウンロードして編集する ... 1299
2. アプリマニフェストをアップロードする ... 1300
3. ユーザー用にアプリをインストールしてピン留めする ... 1300
Tableau アプリを設定する ... 1301
1. 接続済みアプリを作成する ... 1301
2. Tableau アプリにサイト設定を追加する ... 1302

サイトの設定を追加または削除する ... 1302
Microsoft Teams での Tableau コンテンツへのアクセス ... 1303
コンテンツを探索する ... 1304
コンテンツを検索する ... 1304
メトリクスをフォローする ... 1304
コンテンツを共有する ... 1305
サイトを切り替える ... 1305
チャネルやミーティングへの Tableau コンテンツの追加 ... 1306
チャネル タブにコンテンツをピン留めする ... 1306
ミーティングに Tableau コンテンツを追加する ... 1308
メッセージで Tableau コンテンツを共有する ... 1309
サーバーの管理 ... 1311
セキュリティ ... 1316
認証 ... 1316
アドオン認証の互換性 ... 1318
クライアント認証の互換性 ... 1320
ユーザー インターフェイス (UI) を通じて処理される認証 ... 1320
認証はプログラムで処理されます ... 1321
ローカル認証 ... 1321
外部認証ソリューション ... 1322
Kerberos ... 1322
SAML ... 1322

OpenID Connect ... 1322
相互 SSL ... 1323
接続済みアプリ ... 1323
直接信頼 ... 1323
EAS または OAuth 2.0 信頼 ... 1323
信頼できる認証 ... 1324
LDAP ... 1324
SCIM ... 1324
その他の認証シナリオ ... 1325
データアクセスとソース認証 ... 1325
ローカル認証 ... 1325
パスワードの保存 ... 1326
パスワード設定の構成 ... 1326
TSM Web インターフェイスの使用 ... 1327
TSM CLI の使用 ... 1327
構成ファイル リファレンス ... 1329
SAML ... 1331
認証の概要 ... 1333
SAML 要件 ... 1334
証明書およびアイデンティティプロバイダー (IdP) の要件 ... 1334
SSL オフロード ... 1336
SAML での SSL 証明書およびキー ファイルの使用 ... 1337

ユーザー管理要件 1337
SAML 互換性についての注意事項と要件 1339
Tableau クライアントアプリケーションでの SAML SSO の使用 1343
認証されたユーザーを Tableau クライアントに再びリダイレクトする 1343
XML データの要件 1344
サーバー全体の SAML の構成 1348
はじめる前に 1349
TSM Web インターフェイスの使用 1349
TSM CLI の使用 1353
はじめる前に 1353
ステップ 1: リターン URL、SAML エンティティ ID を構成し、証明書ファイルとキーファイルを指定する 1354
ステップ 2:Tableau Server メタデータを生成し IdP を構成する 1355
ステップ 3: アサーションを一致させる 1355
オプション: SAML を使用したクライアントのタイプの無効化 1357
オプション: AuthNContextClassRef 値を追加します。 1357
構成のテスト 1358
Tableau Server で Salesforce IdP を使用した SAML を構成する 1358
Salesforce をアイデンティティプロバイダーとして有効にする 1359
Tableau Server で SAML を構成する 1359
Tableau Server を Salesforce の接続アプリケーションとして追加する 1359
Lightning Web コンポーネントを有効にする 1360
Tableau ビューを Salesforce に埋め込む 1360

Tableau Viz Lightning Web コンポーネントの SAML の構成 1360
要件 1361
認証ワークフローの構成 1361
Tableau Server でフレーム内認証を有効にする 1362
Tableau Server のバージョン 1362
SAML IdP を使用してフレーム内認証を有効にする 1363
Salesforce の承認リストのドメイン 1363
Salesforce IdP 1363
Okta IdP 1363
Ping IdP 1363
OneLogin IdP 1364
ADFS とEntraID IdP 1364
Salesforce Mobile アプリ 1364
Tableau Server での Azure AD IdP を使用した SAML の構成 1365
始める前に: 前提条件 1365
ステップ 1: Azure AD への SSL 接続を確認する 1365
ステップ 2: Tableau Server 上で SAML を構成する 1366
ステップ 3: Azure AD クレーム ルールを構成する 1366
ステップ 4: Azure AD メタデータを Tableau Server に提供する 1367
Azure AD アプリプロキシ 1368
トラブルシューティング 1368
Tableau Server での AD FS を使用した SAML の構成 1369

前提条件 ... 1369
ステップ 1: AD FS への SSL 接続を確認する ... 1370
ステップ 2: Tableau Server 上で SAML を構成する ... 1370
ステップ 3: Tableau Server からのサインイン要求を承認するための AD FS の設定 ... 1370
ステップ 4: AD FS メタデータを Tableau Server に提供 ... 1374
Kerberos データベース委任とともに SAML SSO を使用する ... 1376
プロセスの概要 ... 1376
Tableau Server で Kerberos による SAML を構成する ... 1377
サイト固有の SAML の構成 ... 1377
サイト固有 SAML 有効化の前提条件 ... 1377
サイト固有の SAML に関連するサーバー全体の設定 ... 1378
サーバーにおけるサイト固有 SAML のサポートの構成 ... 1379
コマンドについて ... 1380
サイトの SAML の構成 ... 1380
SAML 証明書の更新 ... 1384
サーバー全体の SAML の証明書を更新する ... 1384
サイト固有の SAML 証明書の更新 ... 1386
SAML のトラブルシューティング ... 1387
SAML、および自動ログインの有効化 ... 1387
SAML の構成時に "HTTP Status 500" エラーが発生する ... 1387
コマンド ラインからのサインイン ... 1388
ログインの失敗: ユーザーが見つかりませんでした ... 1388

ログインの失敗: SSL オフロード 1388
SAML エラー ログ 1389
末尾のスラッシュ 1389
接続を確認 1390
複数の定義域 1390
Kerberos 1390
Kerberos のしくみ 1391
Kerberos の要件 1392
一般的な要件 1392
Active Directory の要件 1393
Kerberos 委任 1393
Keytab 要件の理解 1394
Windows Active Directory でのユーザー認証 (SSO) 1394
バッチファイル: SPN を設定してキータブを Active Directory に作成します。 1395
SPN およびキータブ バッチファイル コンテンツ 1396
オペレーティング システム 1402
ディレクトリサービス 1403
データソース委任 1404
Kerberos の構成 1405
TSM Web インターフェイスの使用 1405
TSM CLI の使用 1407
SSO 構成の確認 1407

Kerberos SSO に対する Tableau クライアントのサポート 1408
一般的なブラウザー クライアントのサポート 1408
Tableau Desktop およびブラウザーのクライアント 1408
Tableau Mobile アプリ クライアント 1409
オペレーティング システムとブラウザーに固有の注記 1409
注 1: Windows デスクトップでの Internet Explorer または Chrome 1410
注 2: Windows または Mac OS X デスクトップでの Firefox 1411
注 3: Mac OS X デスクトップでの Chrome 1412
注 4: iOS で使用する Mobile Safari または Tableau Mobile 1413
注 5: Android プラットフォーム 1413
詳細 1413
Kerberos のトラブルシューティング 1413
Tableau Server へのシングル サインオン 1414
クライアント コンピューターでのサインイン エラーに関するトラブルシューティング 1414
サーバーでのサインイン エラーに関するトラブルシューティング 1416
Kerberos 構成スクリプトの確認 1418
データ ソース SSO 1418
委任されたデータ ソース アクセスの失敗 1418
Kerberos 委任マルチドメイン構成 1418
クロス ドメインの制限付き委任 1419
Web 作成 1420
相互 SSL 認証の構成 1420

ユーザー認証セッションの時間制限 ... 1421
証明書の使用 ... 1421
クライアント証明書要件 ... 1422
TSM Web インターフェイスの使用 ... 1423
TSM CLI の使用 ... 1425
ステップ 1: 外部サーバー通信に SSL が必要 ... 1425
ステップ 2: 相互 SSL を構成して有効にする ... 1425
相互 SSL の追加オプション ... 1426
フォールバック認証 ... 1426
ユーザー名マッピング ... 1427
証明書失効リスト(CRL) ... 1427
相互 SSL 認証のしくみ ... 1427
相互認証中にクライアント証明書をユーザーにマッピングする ... 1428
ユーザー名マッピングオプション ... 1429
証明書マッピングの変更 ... 1430
複数ドメイン組織でユーザー名マッピングのあいまいさに対処する ... 1431
OpenID Connect ... 1432
認証の概要 ... 1432
OpenID Connect で Tableau が機能する仕組み ... 1434
OpenID Connect の使用要件 ... 1435
IdP アカウント ... 1435
ローカル アイデンティティストア ... 1436

IdP クレーム - ユーザーのマッピング 1436
既定: メール クレームを使用してユーザーをマッピングする 1437
ドメイン名を無視 1437
カスタム クレームを使用してユーザーをマッピングする 1438
sub クレームを変更する 1439
認証コンテキスト 1440
OpenID Connect に対応するように ID プロバイダーを構成する 1440
リダイレクト URL 1441
署名アルゴリズム 1441
IdP プロセスの例 1441
OpenID Connect に対応するように Tableau Server を構成する 1442
TSM Web インターフェイスの使用 1442
TSM CLI の使用 1444
OpenID Connect を使用して Tableau Server にサインインする 1446
OpenID Connect を使用してサインインする 1446
サーバー管理者にコマンドライン ツールのサインインを制限する 1447
OpenID 接続認証要求パラメーター 1447
scope 値の構成 1448
Tableau Server で OpenID Connect 用に IdP を変更する 1449
プロバイダーの変更 1449
ユーザー識別子のリセット 1449
OpenID Connect のトラブルシューティング 1450

拡張 OpenID ロギングの有効化 1450
コマンドラインからのサインイン 1450
ログインできませんでした 1451
エラー 69: "サインインできません" 1451
OpenID エラー ログ 1451
ユーザーが見つかりません 1452
信頼できる認証 1452
信頼できる認証のしくみ 1453
信頼できるチケットの保存方法 1454
Tableau Server への信頼できる IP アドレスまたはホスト名の追加 1455
TSM Web インターフェイスの使用 1455
TSM CLI の使用 1457
Tableau Server からのチケットの取得 1458
チケット付きビューの表示 1460
Tableau Server ビューの例 1460
埋め込まれたビューの例 1460
オプション: クライアント IP 照合の設定 1462
信頼できる認証のテスト 1463
ステップ 1: テストユーザーの追加 1464
ステップ 2: テスト HTML ページの作成 1464
ステップ 3: Tableau Server からの信頼できるチケットの取得 1466
ステップ 4: 信頼できるチケットを使用したテストアクセス 1467

信頼できる認証のトラブルシューティング 1467

関連項目 1468

Tableau Server から返されたチケットの値 -1 1468

HTTP 401 - 承認されていません 1470

HTTP 404 - ファイルが見つかりません 1470

無効なユーザー (SharePoint または C#) 1471

間違った IP アドレスからチケットを引き換えようとしています 1471

Cookie 制限エラー 1472

サーバーとの通信中にエラーが発生しました (403) 1472

SCIM 1472

Tableau Server との SCIM 統合の構成 1473

ステップ 1: 前提条件を実行する 1473

ステップ 2: サイトレベルの SCIM を構成する 1473

Tableau で生成されたトークンを使用して SCIM を有効にする 1474

外部トークンを使用して SCIM を有効にする 1475

ステップ 3: ユーザーとグループをプロビジョニングする 1476

個人用アクセス トークン 1476

個人用アクセス トークンについて 1478

サーバー管理者の偽装 1478

偽装サインインの要求時に Tableau Server で個人用アクセス トークンが承認されるようにする 1478

個人用アクセス トークンの作成 1479

個人アクセス トークンの有効期限の変更 1479

個人用アクセストークンの失効 ... 1479
個人用アクセストークンの使用状況の追跡と監視 ... 1480
Tableau 接続済みアプリを使用してアプリケーションを統合する ... 1480
直接信頼 ... 1481
OAuth 2.0 信頼 ... 1482
直接信頼を使用して接続済みアプリを設定する ... 1482
Tableau 接続済みアプリが直接的な信頼関係と連携するしくみ ... 1483
接続済みアプリの主要コンポーネント ... 1483
接続済みアプリのワークフロー ... 1484
埋め込みのワークフロー ... 1484
接続済みアプリの作成 ... 1485
ステップ 1: 接続済みアプリを作成する ... 1485
ステップ 2: シークレットを生成する ... 1487
ステップ 3: JWT を構成する ... 1487
JWT の例 ... 1491
ステップ 4: 次のステップ ... 1493
埋め込みのワークフローの場合 ... 1493
REST API 認証ワークフローの場合 ... 1494
メタデータ API ワークフローの場合 ... 1494
接続済みアプリの管理 ... 1494
接続済みアプリの無効化と削除、およびシークレットの削除による影響 ... 1496
アクセスレベル (埋め込みのワークフローのみ) ... 1496

ドメイン許可リストルール (埋め込みのワークフローのみ) ... 1497
ドメイン オプション ... 1497
ドメインの書式設定 ... 1497
動的グループ メンバーシップ (埋め込みワークフローのみ) ... 1498
既知の問題 (埋め込みのワークフローのみ) ... 1499
トラブルシューティング ... 1500
OAuth 2.0 信頼を使用して接続済みアプリを設定する ... 1500
Tableau 接続済みアプリが OAuth 2.0 信頼と連携するしくみ ... 1500
接続済みアプリの主要コンポーネント ... 1500
接続済みアプリのワークフロー ... 1501
埋め込みのワークフロー ... 1501
接続済みアプリを作成する ... 1502
ステップ 1: 始める前に ... 1502
ステップ 2: EAS を Tableau Server に登録する ... 1506
サイトレベルの EAS について ... 1506
サーバー全体の EAS ... 1506
オプション 1: TSM Web UI を使用する ... 1506
オプション 2: TSM CLI を使用する ... 1509
サイトレベルの EAS ... 1509
ステップ 1: 接続済みアプリを有効にする ... 1510
ステップ 2: EAS を登録する ... 1511
ステップ 3: 次のステップ ... 1512

埋め込みのワークフローの場合 1512

埋め込み用のドメイン許可リストを使用してコンテンツを埋め込むことができる場所を制御する 1513

REST API 認証ワークフローの場合 1513

メタデータAPI ワークフローの場合 1513

接続済みアプリの管理 1513

動的グループ メンバーシップ (埋め込みワークフローのみ) 1514

既知の問題 (埋め込みのワークフローのみ) 1514

トラブルシューティング 1515

接続済みアプリのアクセス スコープ 1519

スコープ アクション 1520

スコープの種類 1520

REST API アクセスを認可する方法の概要 1522

例 1522

JWT 認可をサポートするREST API メソッド 1524

ワイルドカード (*) スコープ 1524

クロスカテゴリスコープ 1525

個別スコープ 1526

ラベル 1527

データ ソース 1528

抽出 1530

フロー 1531

ジョブ 1531

メトリクス (廃止) 1531
サブスクリプション 1533
ビュー 1534
ワークブック 1536
パブリッシュ 1537
ダウンロード 1538
ユーザー 1539
グループ 1540
グループ セット 1541
プロジェクト 1542
パーミッション 1543
サイト 1547
トラブルシューティングのスコープ 1548
401001 - サインイン エラー 1548
401002 - 不正アクセス エラー 1549
接続済みアプリをトラブルシューティングする - 直接信頼 1549
データ接続認証 1554
詳細 1554
Kerberos 委任の有効化 1554
サポートされているデータ ソース 1555
要件 1555
Web オーサリングとユーザーの Kerberos 認証 1555

構成プロセス ... 1556
関連項目 ... 1559
JDBC コネクタの Kerberos 委任を有効にする ... 1559
サポートされているデータ ソース ... 1559
JDBC コネクタの Kerberos 実行アカウント認証を有効にする ... 1560
サポートされているデータ ソース ... 1560
OAuth 接続 ... 1560
OAuth プロセスの概要 ... 1561
既定の保存済み認証資格情報コネクタ ... 1562
データ接続のためのアクセス トークン ... 1565
承認済みクライアントからの認証用のアクセス トークン ... 1565
既定の管理対象キーチェーン コネクタ ... 1565
トークンの制限とストレージ ... 1566
未使用のキーチェーン レコードの削除 ... 1566
管理対象キーチェーンでのシナリオの制限 ... 1567
管理対象キーチェーンを保存された認証資格情報へ変換 ... 1567
サイトのカスタム OAuth の設定 ... 1567
保存済みアクセス トークンの許可 ... 1568
認証資格情報の一元管理 ... 1569
関連項目 ... 1570
Salesforce.com OAuth を保存済み認証資格情報に変更する ... 1570
手順の概要 ... 1571

ステップ1: Salesforce に接続されたアプリを作成する 1572
ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する 1574
サイトのカスタム OAuth の構成 1575
1: OAuth クライアント ID、クライアント シークレット値、リダイレクト URL の準備 1575
2: OAuth クライアント ID とクライアント シークレットの登録 1576
3: 保存した認証資格情報の検証と更新 1576
4: 保存した認証資格情報を更新するようにユーザーに通知する 1577
アクセス トークンの管理 1577
OAuth と最新の認証用に Azure AD を構成する 1577
ステップ 1: Azure 用の OAuth クライアントを登録する 1578
ステップ 2: Azure 用に Tableau Server を構成する 1580
Azure Data Lake StorageGen2 用の既定の OAuth クライアントを構成する 1580
Azure Synapse、Azure SQL Database、Azure Databricks 用の既定のクライアントを構成する 1581
OneDrive および SharePoint Online の既定の OAuth クライアントを構成する 1581
SharePoint リスト用 (JDBC) の既定の OAuth クライアントを構成する 1582
OneDrive 用 (非推奨) の既定の OAuth クライアントを構成する 1583
サーバーの再起動シナリオ 1583
複数のコネクタの設定 1584
サイトのカスタム OAuth の構成 1584
1: OAuth クライアント ID、クライアント シークレット値、リダイレクト URL の準備 1585
2: OAuth クライアント ID とクライアント シークレットの登録 1585
3: 保存した認証資格情報の検証と更新 1586

4: 保存した認証資格情報を更新するようにユーザーに通知する 1587
Snowflake 接続用 OAuth の設定 1587
OAuth クライアントを Snowflake に登録する 1588
オプション 1: TSM を使用して Snowflake 接続の OAuth を設定する 1588
オプション 2: サイトごとに Snowflake 接続の OAuth を構成する 1591
1: OAuth クライアント ID、クライアントシークレット、リダイレクト URL の準備 1591
2: OAuth クライアント ID とクライアントシークレットの登録 1592
3: 保存した認証資格情報の検証と更新 1593
4: 保存した認証資格情報を更新するようにユーザーに通知する 1594
Tableau Server を Salesforce Data Cloud に接続する 1594
Tableau Server (バージョン 2023.3 以降) 1595
ステップ 1: Salesforce 接続済みアプリの作成 1595
ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する 1596
サイトのカスタム OAuth の構成 1597
Tableau Server (バージョン 2023.1 以前) 1597
ステップ 1: コネクタを設定する 1598
ステップ 2: 顧客データプラットフォーム JDBC ドライバーをインストールする 1598
ステップ 3: Salesforce 接続済みアプリを作成する 1598
Customer Data Platform で OAuth を使用する 1601
OAuth 設定での TSM コマンドの使用 1602
複数のコネクタの設定 1602
ステップ 1: OAuth クライアント ID とクライアントシークレットの登録 1603

ステップ 2: 保存された認証資格情報の検証と更新 1603
ステップ 3: 保存された認証資格情報を更新するようにユーザーに通知 1604
関連項目 1604
SAP HANA の外部 OAuth 1604
HANA で IdP を設定する 1604
IdP を設定する 1605
HANA に接続する 1606
Okta 1606
Snowflake の外部 OAuth 1607
Snowflake で IDP を設定する 1607
Tableau で IDP を設定する 1607
Snowflake に接続する 1608
Okta 1608
Hyper Query Processing (ベータ版) 1609
Tableau Server で Hyper Query Processing を有効にする 1609
関連項目 1610
Amazon Redshift IAM OAuth の設定 1610
ステップ 1: IDP を構成する 1611
AWS で IDP を構成する 1612
Redshift ユーザーのロールを構成する 1612
Redshift に接続する 1613
トークン 1614

フェデレーションのグループ化 1615
使用上の注意 1615
Okta 1616
ドライバーを更新する 1616
トラブルシューティング 1617
Amazon Redshift IAM ID センター OAuth の設定 1620
ステップ 1: IDP を構成する 1620
ステップ 2: AWS で IDP とロールを構成する 1621
ステップ 3: Redshift に接続する 1621
トークン 1623
Okta 1623
ドライバーを更新する 1624
Redshift IAM IDC OAuth のトラブルシューティング 1624
Dremio 向け OAuth の設定 1627
ステップ 1: OAuth クライアントを Dremio に登録する 1627
ステップ 2: Dremio OAuth 用に Tableau Server を構成する 1627
複数のコネクタの設定 1628
サイトのカスタム OAuth の構成 1629
1: OAuth クライアント ID、クライアント シークレット値、リダイレクト URL の準備 1629
2: OAuth クライアント ID とクライアント シークレットの登録 1630
3: 保存した認証資格情報の検証と更新 1630
4: 保存した認証資格情報を更新するようにユーザーに通知する 1631

Dropbox 向け OAuth の設定 1631
ステップ 1: 新しいアプリを作成する 1631
ステップ 2: Dropbox 用に Tableau Server を構成する 1632
サイトのカスタム OAuth の構成 1632
1: OAuth クライアント ID、クライアントシークレット値、リダイレクト URL の準備 1633
2: OAuth クライアント ID とクライアントシークレットの登録 1633
3: 保存した認証資格情報の検証と更新 1634
4: 保存した認証資格情報を更新するようにユーザーに通知する 1635
Google での OAuth の設定 1635
手順の概要 1636
クライアント ID を取得して Google API を有効にする 1636
Google OAuth のための Tableau Server の構成 1639
サイトのカスタム OAuth の構成 1639
1: OAuth クライアント ID、クライアントシークレット値、リダイレクト URL の準備 1640
2: OAuth クライアント ID とクライアントシークレットの登録 1640
3: 保存した認証資格情報の検証と更新 1641
4: 保存した認証資格情報を更新するようにユーザーに通知する 1642
Google データソースの作成と編集 1642
アクセストークンの管理 1642
Intuit QuickBooks Online 向け OAuth の設定 1642
ステップ 1: Intuit アプリを作成する 1643
ステップ 2: Intuit QuickBooks Online 用に Tableau Server を構成する 1643

アクセス トークンの管理 1644
OAuth 接続のトラブルシューティング 1644
競合エラー 1644
SAP HANA SSO の構成 1645
はじめる前に 1646
SAP HANA での Tableau Server SAML を構成する 1646
キーペア認証のための Snowflake の設定 1648
キーペア認証の詳細を追加する 1648
関連項目 1649
Kerberos サービス アカウントへのアクセスの有効化 1649
実行サービス アカウントを使用したデータ アクセス 1650
推奨事項 1650
要件 1651
構成プロセス 1651
JDBC コネクタの Kerberos 実行アカウント認証を有効にする 1653
サポートされているデータ ソース 1653
SQL Server の偽装 1653
偽装の要件 1654
擬装のしくみ 1655
実行サービス アカウントでの擬装 1656
埋め込み SQL 認証資格情報の擬装 1657
カスタム TSM 管理グループの構成 1659

ステップ 1: 新しいグループを作成する 1660
ステップ 2: Tableau Server の構成 1660
ステップ 3: 新しいグループにユーザーを追加する 1661
認可 1661
サイトロール 1661
パーミッション 1661
データアクセスと外部認可 1662
データセキュリティ 1663
Tableau の行レベルのセキュリティオプションの概要 1665
手動でユーザーフィルターを作成し、ユーザーを値にマッピングする 1666
データのセキュリティフィールドを使用して動的ユーザーフィルターを作成する 1666
データポリシーを使用する 1667
データベース内の既存の RLS を使用する 1667
行レベルのセキュリティオプションの比較 1668
どの行レベルのセキュリティオプションを使用するか? 1669
データソースとワークブックの RLS のベストプラクティス 1670
RLS のワークフロー 1670
資格テーブル 1671
資格表モデル 1671
ユーザーとロール 1672
結合 1672
行レベルのセキュリティを実装する 1673

最も深い粒度 ... 1673
スパース資格 ... 1673
データソースフィルター ... 1674
最も深い粒度を持つすべてのアクセス ... 1674
操作のパフォーマンスと処理順序 ... 1675
ライブ接続 ... 1675
抽出 ... 1676
抽出に関する考慮事項 ... 1676
単一の表の抽出 ... 1677
組み込みの行レベル セキュリティをデータベースで使用する ... 1678
データベースの行レベル セキュリティ ... 1678
偽装 (Microsoft SQL Server) ... 1679
Kerberos と制約付き委任 ... 1679
OLAP キューブ ... 1680
SAML 委任とSAP HANA ... 1680
ユーザー固有のセッションを強制する初期 SQL (Oracle VPD) ... 1680
行レベル セキュリティメソッドの比較マトリックス ... 1681
サーバー シークレットの管理 ... 1683
シークレット ストレージの仕組みについて理解する ... 1684
マスター キーへのアクセス権を持っているのは誰か ... 1684
構成情報のインポートおよびエクスポート ... 1685
インポートおよびエクスポート操作におけるシークレットの保護 ... 1685

クラスター ノード ... 1688
シークレット ストレージ イベントのログ作成 ... 1688
シークレットの管理 ... 1688
シークレットの更新 ... 1689
パスワードの取得 ... 1689
拡張機能のセキュリティ- 展開に関するベストプラクティス ... 1693
Tableau での拡張機能のセキュリティ ... 1693
Tableau の信頼できる拡張機能 ... 1694
サンドボックス化された拡張機能 ... 1695
Tableau が作成した拡張機能 ... 1695
信頼できるパートナーが作成した拡張機能 (オプトイン) ... 1696
ネットワーク対応拡張機能 ... 1696
ローカルに展開されたネットワーク対応拡張機能 ... 1697
サードパーティープロバイダー管理のネットワーク対応拡張機能 ... 1697
ネットワーク対応拡張機能のリスクに関する考慮事項 ... 1697
ネットワーク対応拡張機能に関するセキュリティ上の脅威を軽減する ... 1698
Tableau を使用した拡張機能の管理 ... 1699
Tableau Desktop での推奨事項 ... 1699
展開シナリオの例 ... 1700
Tableau Server および Tableau Cloud での推奨事項 ... 1702
Tableau の信頼できる拡張機能と許可リスト上のネットワーク対応拡張機能を許可する ... 1703
許可リストのチェックリスト: ... 1704

許可リストに拡張機能を追加します。 1704

Tableau Server での特定の拡張機能の実行のブロック 1704

サイトでの拡張機能のオフ 1705

ネットワーク対応拡張機能を実行するためのユーザープロンプトの表示または非表示 1705

Tableau の信頼できる拡張機能のすべてまたは一部をオフにする 1705

Tableau Server のキー管理システム 1706

Tableau Server のローカル KMS 1707

構成のトラブルシューティング 1708

マルチノードの構成ミス 1708

Tableau Server で RMK と MEK を再生成する 1708

保存中の抽出の暗号化 1708

制限事項 1708

パフォーマンスの概要 1709

バックグラウンダー負荷の増加 1709

Viz の読み込み時間とワーカー負荷の増加 1710

バックアップと復元への影響 1710

サイトでの保存中の暗号化の実施 1711

サイトでの保存中の暗号化の有効化 1711

サイトでの保存中の暗号化の無効化 1711

すべてのサイトの抽出暗号化モードの表示 1712

パブリッシュ済みワークブックまたはデータソースでの抽出の暗号化または復号化 1712

複数アイテムの暗号化または復号化 1713
単一項目の暗号化ステータスの表示 1713
暗号化ステータスによるデータソースのフィルター 1713
暗号化ステータスによるワークブックのフィルター 1713
抽出の暗号化または復号化のバックグラウンドタスクのステータスの表示 1714
tabcmd ユーティリティ 1714
サイト作成時の抽出暗号化モードの指定 1714
サイト編集時の抽出暗号化モードの指定 1714
サイトのリスト作成時の抽出暗号化モードの取得 1714
サーバーにワークブック、データソース、または抽出をパブリッシュするときの抽出の暗号化 1714
サイト上のすべての抽出の復号化 1715
サイト上のすべての抽出の暗号化 1715
新しい暗号化キーを使用したサイト上のすべての抽出の再暗号化 1715
Tableau Server Rest API 1715
ネットワークセキュリティ 1716
クライアントからTableau Server へ 1716
インターネットからのクライアントアクセス 1717
クリックジャック保護 1717
Tableau Server からデータベースへ 1717
Tableau Server からインターネットへ 1718
Tableau Server からSMTP サーバーへ 1718
リポジトリとの通信 1718

クラスタ内のサーバー コンポーネント通信 1718
クリックジャック保護 1719
クリックジャック保護の効果 1719
クリックジャック保護の無効化 1720
HTTP 応答ヘッダー 1721
応答ヘッダーの構成 1721
HTTP Strict Transport Security (HSTS) 1721
オプション 1722
Referrer-Policy 1722
オプション 1722
X-Content-Type-Options 1723
オプション 1723
X-XSS-Protection 1723
オプション 1723
コンテンツ セキュリティ ポリシー 1724
CSP の構成と有効化 1724
ステップ 1: 既定のディレクティブの設定 1724
ステップ 2: ディレクティブの追加 (オプション) 1726
ステップ 3: レポート専用ディレクティブの指定 (オプション) 1727
ステップ 4: Tableau Server での CSP の有効化 1727
ステップ 5: tsm pending-changes apply の実行 1728
CSP レポートの表示 1728

SSL ... 1729
Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する ... 1730
SSL 証明書要件 ... 1730
クラスタの SSL を構成 ... 1732
複数ゲートウェイのある SSL ... 1732
Tableau Server クラスター環境の追加構成情報 ... 1732
環境の準備 ... 1733
Tableau Server での SSL の構成 ... 1733
TSM Web インターフェイスの使用 ... 1733
TSM CLI の使用 ... 1734
ポート リダイレクトとロギング ... 1735
SSL ポートのローカル ファイアウォールへの追加 ... 1736
SSL 証明書の変更または更新 ... 1736
例: SSL 証明書 - キーとCSR の生成 ... 1737
キーおよび CSR の作成手順 ... 1738
複数のドメイン名に対する証明書の設定 ... 1738
キーの生成 ... 1738
証明書署名要求を作成して認証局に送信する ... 1739
CSR を認証局に送信して SSL 証明書を取得する ... 1739
キーおよび証明書を使用して Tableau Server を構成する ... 1740
SAN 証明書の場合、OpenSSL 構成ファイルを変更する ... 1740
内部 Postgres 通信用に SSL を設定する ... 1742

TSM Web インターフェイスの使用 1742
TSM CLI の使用 1743
コマンドの実行内容 1744
repository-ssl enable のオプション 1744
クラスタ環境 1745
TSM コントローラーのカスタム SSL 証明書の設定 1745
デフォルトの TSM SSL 機能 1745
Tableau Server v2023.1 SSL カスタム証明書 1746
設定 1747
クライアントからの直接接続を許可するようPostgres SSL を構成する 1747
相互 SSL 認証の構成 1748
ユーザー認証セッションの時間制限 1749
証明書の使用 1749
クライアント証明書要件 1750
TSM Web インターフェイスの使用 1751
TSM CLI の使用 1753
ステップ 1: 外部サーバー通信に SSL が必要 1753
ステップ 2: 相互 SSL を構成して有効にする 1753
相互 SSL の追加オプション 1754
フォールバック認証 1754
ユーザー名マッピング 1755
証明書失効リスト (CRL) 1755

相互認証中にクライアント証明書をユーザーにマッピングする 1755
ユーザー名マッピングオプション 1756
証明書マッピングの変更 1757
複数ドメイン組織でユーザー名マッピングのあいまいさに対処する 1758
外部の LDAP アイデンティティストアへの暗号化チャネルの構成 1758
証明書の要件 1759
Tableau キーストアへの証明書のインポート 1760
暗号化方式 1760
シンプル バインド用の暗号化チャネルの構成 1761
構成するタイミング 1761
Tableau Server の新規インストールの場合 1762
Active Directory 環境での新規インストールの場合 1762
アップグレードシナリオ 1764
デフォルトの暗号化 LDAP チャネルを無効にする 1764
新規インストールで無効にする 1764
アップグレードする前に無効にする 1765
エラー メッセージ 1766
GUI のセットアップの実行時 1766
システム ユーザー、sudo 権限、systemd 1767
権限の分離 1767
sudo 権限 1767
systemd ユーザー サービス 1768

systemctl コマンドの実行 ... 1768
セキュリティ強化チェックリスト ... 1769
セキュリティ更新のインストール ... 1769
1.現在のバージョンへの更新 ... 1769
2.有効な信頼された証明書を使用して SSL/TLS を構成する ... 1770
3.古いバージョンの TLS の無効化 ... 1770
4.内部トラフィック用に SSL 暗号化を構成する ... 1771
5.ファイアウォールによる保護を有効にする ... 1771
6.サーバー コンピューターや重要なディレクトリへのアクセスを制限する ... 1772
7.新しいシークレットとトークンの生成 ... 1772
8.使用していないサービスの無効化 ... 1773
JMX サービス ... 1773
9.セッション存続時間設定の検証 ... 1773
10.ファイルベースのデータ ソース用にサーバーの許可リストを設定する ... 1774
11.Web ブラウザー クライアントでの HTTP Strict Transport Security の有効化 ... 1775
12.ゲスト アクセスの無効化 ... 1776
13.Referrer-Policy HTTP ヘッダーを 'same-origin' に設定する ... 1777
14.TLS による SMTP 接続の構成 ... 1777
15.LDAP の SSL を設定 ... 1778
変更リスト ... 1779
ライセンスの管理 ... 1779
ライセンス発行の概要 ... 1780

アクティブ化 ... 1780
オンラインでのアクティブ化 ... 1780
オフラインでのアクティブ化 ... 1781
アクティブ認証の失敗 ... 1781
ライセンス認証の解除 ... 1781
Tableau Server のライセンス発行と仮想マシン (VM) ... 1782
ログインベースのライセンス管理 ... 1782
ユーザーの追加 ... 1783
ライセンス モデルとプロダクト キーを理解する ... 1783
期間ライセンス モデル ... 1784
ロールベースのライセンス モデル ... 1786
コアベースのライセンス モデル ... 1788
組み込み分析の使用量ベースのモデル ... 1788
永続ライセンス モデル (従来) ... 1789
ライセンス エディション ... 1790
Tableau ライセンス エディション ... 1790
Tableau Enterprise ライセンス エディション ... 1791
プライベート接続ライセンス ... 1791
機能ライセンス ... 1791
データ管理 ... 1792
Advanced Management ... 1793
ログインベースのライセンス管理 ... 1793

更新可能なサブスクリプション ライセンス(USL) 1793
USL の基本を理解する 1794
オンラインまたは接続された環境での USL の有効化 1794
オフラインまたは切断された環境での USL の有効化 1794
オフライン環境でのライセンス資格の更新管理 1795
USL のオフラインでの有効化手順 1796
USL オフライン ライセンス資格の更新 1797
サーバー ライセンスの表示 1798
Tableau Server Web UI からのライセンスの表示 1798
TSM Web インターフェイスの使用 1799
TSM CLI の使用 1799
プロダクト キーの有効期限と属性の更新 1801
はじめる前に 1802
TSM Web インターフェイスの使用 1803
TSM CLI の使用 1803
Tableau Server へのライセンスの追加 1803
TSM Web インターフェイスの使用 1804
TSM CLI の使用 1807
オフラインでの Tableau Server の有効化 1808
オフラインでのアクティブ化とログインベースのライセンス管理 (LBLM) 1809
オフラインでの有効化と更新可能なサブスクリプション ライセンス(USL) 1809
オフライン アクティブ化の概要 1809

オフライン アクティブ化 ファイル名の変更 1810
TSM Web インターフェイスの使用 1811
TSM CLI の使用 1814
ステップ 1.Tableau サービス マネージャーへのログイン 1814
ログインできない場合 1814
ステップ 2.オフライン アクティブ化要求 ファイルを生成する 1814
ステップ 3.オフライン アクティブ化要求を Tableau アクティベーション Web サイトにアップロードする 1815
ステップ 4.ライセンスを初期化または認証します 1815
プロダクトキーの認証解除 1816
はじめる前に 1816
TSM Web インターフェイスの使用 1816
TSM CLI の使用 1817
オフラインでの Tableau Server のライセンス認証解除 1817
ライセンス発行タスクの自動化 1818
ライセンス発行のトラブルシューティング 1819
ライセンスのないサーバーの処理 1819
ライセンスを解除されたユーザーベースのサーバー 1819
ライセンスを解除されたコアベースのサーバー 1819
ライセンスを解除されたサーバー管理者 1820
ロールベース ライセンス発行のトラブルシューティング 1821
ライセンスの有効期限切れのためユーザーまたは管理者がライセンスなしである 1821

Creator ライセンスを使用する場合、サーバー管理者サイトロールは変更されない 1823

ライセンスはすぐには利用できない 1823

Viewer ライセンスを持つユーザーは、Tableau Desktop から Tableau Server または Tableau Cloud のワークブックを開くことができない 1823

コアベース ライセンスからロールベース ライセンスへの移行 1823

ロールベース ライセンスへの移行準備 1823

ロールベース ライセンスへの移行 1824

コアベース ライセンスが設定されているサーバーでロールベース ライセンスを使用する 1825

コアベース ライセンスからの移行完了の例 1826

クイック スタート: Tableau Server でログインベースのライセンス管理を使用する 1826

ステップ 1: Tableau Server のインストール 1826

ステップ 2: Tableau Server への許可されたユーザーの追加 1827

ステップ 3: Tableau Desktop または Tableau Prep Builder のライセンス認証 1827

ログインベースのライセンス管理 1828

ログインベースのライセンス管理の動作方法 1829

ログインベースのライセンス管理 と Tableau Cloud 1830

ログインベースのライセンス管理 と Tableau Server 1830

ログインベースのライセンス管理を使用する 1831

ステップ 1: Tableau Server をインストールする 1831

(オプション) ステップ 2: ログインベースのライセンス管理の設定を変更する 1832

ログインベースのライセンス管理設定 1836

Microsoft Windows ... 1839
MacOS ... 1840
(オプション) ステップ 3: 実行の承認 (ATR) の期間を変更する ... 1840
ステップ 4: Tableau Desktop をライセンス認証する ... 1843
ログインベースのライセンス使用状況の表示 ... 1844
トラブルシューティング ... 1846
ログインベースのライセンス管理 が Tableau Server で有効になっていない ... 1846
ログインベースのライセンス管理 が Tableau Desktop で有効になっていない ... 1847
年間サブスクリプションを購入しても、プロダクトキーの有効期限は変更されません ... 1847
Creator ライセンスがない ... 1847
最大数の PC でライセンス認証した ... 1847
アクティブ化を最大にする ATR トークン期間を短縮するには ... 1848
コンピューターをライセンス認証されていない状態に戻すには ... 1849
Tableau 認証資格情報が無効である ... 1850
PC の時計が現在の時刻と同期されていない ... 1850
認証資格情報でライセンス認証できない ... 1850
ゼロダウンタイム のライセンス発行 ... 1851
Tableau Server を再起動する必要がある場合 ... 1851
Tableau Enterprise について ... 1852
Tableau Enterprise ライセンス ... 1852
Tableau Enterprise 機能一覧 ... 1852
アイデンティティの移行について ... 1853

既存の展開に対する手順の概要 1854
重要な用語 1855
アイデンティティの移行の目的 1855
アイデンティティ移行時の注意点 1856
アイデンティティの移行を実施する 1857
ステップ 1: 始める前に 1857
ステップ 2: 新しいスキーマとサインイン機能を有効にする 1861
ステップ 3: アイデンティティの移行を開始する 1862
ステップ 4: アイデンティティの移行を完了する 1862
ステップ 5: アイデンティティサービスを使用するように Tableau Server を構成する 1864
ステップ 6: アイデンティティプールを有効にする 1865
アイデンティティの移行を管理する 1865
アイデンティティの移行ジョブを管理する 1866
アイデンティティの移行設定を変更する 1869
移行設定 1869
アイデンティティの移行を無効にする 1871
アイデンティティの移行を完了し、アイデンティティサービスを構成します 1872
ステップ 1: アイデンティティの移行を検証して完了する 1872
ステップ 2: アイデンティティサービスを使用するように Tableau Server を構成する 1873
ステップ 3: アイデンティティプールを有効にする 1874
アイデンティティ移行時の競合の解決 1874
ステップ 1: アイデンティティの競合を解決する 1874

クイック リファレンス: アイデンティティの競合 1875
ステップ 2: アイデンティティの移行を完了する 1877
ステップ 3: アイデンティティサービスを使用するように Tableau Server を構成する 1878
ステップ 4: アイデンティティプールが有効になっていることを確認する 1879
アイデンティティの移行に関する問題をトラブルシューティングする 1879
バックアップを復元できません 1879
ステップ 1: legacy-identity-mode を有効にして、バックアップを復元する 1880
ステップ 2: アイデンティティの移行を検証して完了する 1880
ステップ 3: アイデンティティサービスを使用するように Tableau Server を構成する 1882
ステップ 4: アイデンティティプールが有効になっていることを確認する 1882
[アイデンティティの移行] ページの “予期しないエラー” 1882
移行の進行状況が応答しないか動いていない 1883
「アイデンティティの移行が進行中です」というポップアップが表示され続ける 1884
アイデンティティ移行ページが消える 1884
ユーザーがサインインできない 1884
アイデンティティの移行を元に戻す 1885
アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証 1885
ID プールとは 1886
ID プールを使用する場合 1887
ID プールの詳細 1887
初期プール (TSM 構成) と ID プール 1887
ID プールがユーザーのサインイン エクスペリエンスに与える影響 1888

Tableau におけるユーザー名と識別子 1888
ID プールの設定と管理 1889
前提条件 1889
はじめに 1890
ステップ 1: Tableau Server を設定してセッションを確立する 1890
ステップ 2: ID ストアを設定する 1892
例 1893
URI 1893
例 1893
URI 1893
要求本文 (JSON) 1893
レスポンス本文 1894
ステップ 3: 認証を設定する 1894
例 1895
URI 1895
要求本文 (JSON) 1895
レスポンス本文 1896
ステップ 4: ID プールを作成する 1896
例 1897
URI 1897
要求本文 (JSON) 1897
レスポンス本文の例 1897

ステップ 5: 初期プールにユーザーを追加する ... 1898
ID プールのテスト ... 1900
ID プールの管理 ... 1901
ID プールのトラブルシューティング ... 1901
ID プールの制限 ... 1901
Tableau Server のランディング ページに IdP エラーが表示される ... 1901
Tableau Server のランディング ページにアイデンティティプールが表示されない ... 1901
Tableau Server へのユーザーの追加 ... 1902
はじめる前に ... 1902
ユーザーをサイトレベルではなくサーバーレベルで追加する場合 ... 1903
サーバーへのユーザーの追加 ... 1904
ID プールが設定されていない場合 ... 1904
少なくとも 1 つの ID プールが設定されている場合 ... 1907
Tableau Server 管理者エリアへのサインイン ... 1912
サーバー管理者アカウントとパスワードのリセット ... 1914
Tableau Web 環境の管理者エリアの移動 ... 1914
サイトロールとサイト数に基づくアクセス ... 1915
サーバー管理者 ... 1915
サイト管理者 ... 1917
サーバー管理者のタスク ... 1918
サイト管理者のタスク ... 1919
Tableau サービス マネージャーの Web UI へのサインイン ... 1919

要件 ... 1920
TSM の Web UI へのサインイン ... 1921
サーバーのカスタマイズ ... 1922
Tableau Server の言語とロケール ... 1924
サポート対象の言語 ... 1924
既定の設定 ... 1924
言語とロケールの決定のしくみ ... 1924
Tableau Server でのカスタム フォントの使用 ... 1925
サーバー間でのサイトの管理 ... 1926
サイトの概要 ... 1926
サイトとは ... 1926
認証とサインイン認証資格情報 ... 1927
既定サイト ... 1927
サイトを追加する/しない理由 ... 1928
サイトへの管理者レベルのアクセス権 ... 1929
ライセンスとユーザー数の上限 ... 1930
サイトのエクスポートまたはインポート ... 1931
サイト移行オプション ... 1931
サイト移行に関する制限事項 ... 1932
サイトエクスポートで保持される情報と保持情報 ... 1932
サイトエクスポートで保持されない情報 ... 1932
ソースサイトとターゲットサイトの準備 ... 1933

古くなったコンテンツの削除 ... 1933
使用されなくなったユーザーの削除 ... 1933
ターゲットサイトの作成または識別 ... 1933
サイトIDを見つける ... 1933
アイデンティティストアを確認する ... 1934
必要に応じてターゲットサーバーでユーザーを作成する ... 1934
ターゲットサーバーでのサブスクリプション配信の構成 ... 1934
スケジュールの確認 ... 1934
ユーザーやスケジュールが少ないターゲットにインポートするためのヒント ... 1935
サイトの移行 ... 1936
ステップ1: サイトをエクスポートする ... 1936
ステップ2 : インポートマッピング ファイルを生成する ... 1936
ステップ3 :サイト設定が正しくマッピングされたことを確認する ... 1937
マッピング ファイルを検証するには ... 1938
ステップ4: 正しくマッピングされたファイルをターゲットサイトにインポートする ... 1938
マッピングファイルの内容リファレンス ... 1939
CSV ファイル名:mappingsDomainMapperForGroups ... 1939
CSV ファイル名:mappingsScheduleMapper ... 1940
CSV ファイル名:mappingsSiteMapper ... 1941
CSV ファイル名:mappingsSystemUserNameMapper ... 1941
CSV ファイル名: MappingsScheduleRecurrenceMapperWithAutoCreation ... 1942
サイトの追加または削除 ... 1943

サイトを追加する 1943
サイトを削除する 1944
サイトの可用性 1944
サイトを有効化または一時停止するには 1944
サイトロールの制限の管理 1945
サイトでロールの制限を作成する 1945
サイトロールの制限に達した場合 1946
ユーザーに対してリビジョン履歴の保存を許可する 1947
注 1947
ユーザーがリビジョン履歴で操作する必要があるパーミッション 1947
リビジョン履歴を有効化し、許可されるリビジョンの数を設定する 1948
すべてのリビジョンをクリア 1948
ワークブックのプレビューおよび復元に対するセキュリティ 1949
関連項目 1949
Tableau Server サイトで AI を有効にする 1949
PII、データセキュリティ、多言語サポート 1950
課金に関する考慮事項 1950
Tableau Server で AI を有効にするための前提条件 1950
Tableau の AI を設定する 1951
Tableau Mobile アプリのセキュリティ設定 1954
セキュリティ設定 1955
抽出更新スケジュール 1956

抽出を更新する前に ... 1956
更新スケジュールの設定 ... 1957
手動での抽出の更新 ... 1958
Tableau Desktop からの抽出の更新 ... 1958
抽出更新スケジュールと失敗通知を有効にする ... 1959
サーバーからスケジュールを管理する ... 1960
更新失敗メールのしくみ ... 1960
管理者以外のユーザーへの更新失敗メールを有効にする ... 1960
前回成功した更新日の決定方法 ... 1961
スケジュールの作成または変更 ... 1961
新しいスケジュールを作成するには ... 1962
既存のスケジュールを変更するには ... 1964
スケジュールの作成または変更のルール ... 1964
関連項目 ... 1965
サブスクリプションのカスタム スケジュールを可能にする ... 1965
カスタム スケジュールを有効にする ... 1965
スケジュールされたサーバー ジョブの優先順位付け方法 ... 1966
ジョブとタスク ... 1966
ジョブの優先度ルール ... 1966
スケジュールされた更新後のワークブック パフォーマンスの設定 ... 1968
パフォーマンスへの影響に対する判断 ... 1969
サーバーでのワークブックのキャッシュ停止 ... 1969

サイトでのワークブックのキャッシュ停止 1970
ワークブックのキャッシュに必要なしきい値の設定 1970
サブスクリプションおよびデータ主導アラートへのアクセスの確保 1970
サブスクリプションのサーバー設定 1971
前提条件: サブスクリプション メールを送信するようサーバーを構成する 1972
サブスクリプションを有効にする 1972
サイトでのサブスクリプションのテスト 1974
すべてのユーザー サブスクリプションの管理 1975
サブスクリプションの一時停止 1975
一時停止したサブスクリプションの再開 1975
関連項目 1976
データ主導アラートの設定 1976
データ主導アラートのメールの設定 1977
1 つのサイトですべてのデータ主導アラートを管理する 1977
サイトのデータ主導アラートの無効化 1977
データ主導アラートの一時停止 1978
一時停止したアラートの再開 1978
サーバーのデータ主導アラートのチェック頻度を制御する 1979
サーバーのアラートチェックプロセスの追跡 1979
失敗したアラートの特定および修正 1980
メトリクスのセットアップ 1982
ユーザーがメトリクスを作成できるようにする 1982

サイトのメトリクスの無効化 ... 1983
サーバーのメトリクスの無効化 ... 1983
メトリクスの更新頻度の構成 ... 1983
メトリクス更新の失敗通知を構成する ... 1984
メトリクス更新が中断されるタイミングを構成する ... 1984
メトリクスの管理 ... 1984
アドレスの失敗および中断されたメトリクス更新 ... 1985
一時停止された更新の再開 ... 1985
管理ビューでメトリクスアクティビティを監視する ... 1986
パブリッシュされたデータソースを編集する ... 1986
変更を編集してテストする ... 1987
ロールバック変更 ... 1988
サポートされている接続を理解する ... 1988
パーミッションについて学ぶ ... 1989
フローによってパブリッシュされたデータソースを編集する ... 1989
Tableau Server でのバックグラウンドジョブの管理 ... 1989
概要 ... 1990
タスクのタイプ ... 1991
フィルター ... 1993
ジョブのキャンセル ... 1993
ステータス ... 1994
Tableau サービス マネージャーのジョブ ... 1997

tsm ジョブのキャンセル 1999
TSM ジョブのキャンセル 1999
進行中のジョブのキャンセル 2000
管理ビュー 2001
管理ビューへの移動 2001
プレビルト管理ビュー 2003
ビューのパフォーマンス 2004
フロー実行のパフォーマンス 2005
ビューへのアクセス量 2006
データソースへのアクセス量 2008
すべてのユーザーによるアクション 2009
特定のユーザーによるアクション 2010
最近のユーザーによるアクション 2011
抽出でのバックグラウンドタスク 2012
このビューを理解する 2013
ステータス 2013
タスクの詳細を見る 2014
抽出以外のバックグラウンドタスク 2014
サムネイルのアップグレードジョブ 2016
トラブルシューティング 2017
バックグラウンドタスクの遅延 2017
ロード時間の統計 2019

領域使用量の統計 2020

サーバーのディスク容量 2022

ログインベースのライセンス使用状況 2024

フィルター 2025

過去<nn>日間に使用されたCreatorシートはどれか? 2026

過去 <nn> 日間使用されていない Creator シートはどれか 2026

デスクトップ ライセンスの使用率 2027

過去 <nn> 日間に Tableau を使用したユーザー 2028

過去 <nn> 日間使用されていないライセンス 2029

デスクトップ ライセンスの有効期限 2029

バックグラウンダーダッシュボード 2031

サマリーとフィルター 2032

詳細 2034

古いコンテンツ 2037

サマリーとフィルター 2038

詳細 2040

古いコンテンツのアーカイブまたは削除 2042

[「データに聞く」(Ask Data) の使用状況] ビュー 2044

ダッシュボードの詳細 2045

データ品質に関する警告履歴 2047

警告の詳細の表示 2047

警告履歴のフィルター処理 2048

時間範囲でフィルター処理 2049
コンテンツ タイプでフィルター処理 2049
データ品質に関する警告履歴データへのアクセス 2049
実行可能なユーザー 2050
カスタム管理ビューの作成 2050
パフォーマンス 2051
Tableau Server パフォーマンスの概要 2052
一般的なパフォーマンスのガイドライン 2052
ハードウェアとソフトウェア 2052
外部リポジトリ 2052
構成 2053
サーバー リソース マネージャー (SRM) 2054
パフォーマンス モニタリングの概要 2055
Tableau Server リポジトリでデータを収集する 2055
Tableau Server リポジトリへのアクセスを有効にする 2056
Tableau Server リポジトリへの接続 2057
PostgreSQL のバージョン 2060
Tableau Server データディクショナリについて 2060
パフォーマンスの調整 2061
ユーザー トラフィックの最適化 2062
ユーザー トラフィックを最適化するタイミング 2062
ビューの読み込み時間が遅い 2062

高いリソース使用量がユーザー トラフィックに対応している 2063
ユーザー トラフィック用の最適化方法 2065
VizQL サーバープロセスの数を調節する 2065
他のプロセスの数を調節する 2065
VizQL セッション タイムアウト制限を調節する 2066
キャッシュの更新頻度を減らす 2066
ビューの応答性の評価 2066
クライアント側レンダリングの構成 2067
サポートされるブラウザー 2067
コンピューターとモバイル デバイス用の複雑性のしきい値を構成する 2067
クライアント側レンダリングを無効にする 2068
URL パラメーターを使用したテスト 2068
抽出の最適化 2069
抽出を最適化するタイミング 2069
高い CPU 使用量が抽出の更新に対応している 2069
抽出に失敗する、または実行速度が遅い 2071
抽出用の最適化方法 2072
抽出更新スケジュールを調整する 2072
特定の抽出の高速化 2073
抽出更新の実行モードを構成する 2073
バックグラウンダープロセスの数を増やす 2074
プロセスの隔離 2074

抽出クエリの負荷が大きい環境用に最適化 ... 2074
この構成を使用するタイミング ... 2074
この構成を使用する利点 ... 2075
この構成を使用しないタイミング ... 2076
設定 ... 2077
ハードウェアのガイダンス ... 2078
その他のパフォーマンスの調整と最適化: ... 2079
ノードの追加と再構成の時期 ... 2083
パフォーマンスの記録 ... 2084
パフォーマンスの記録の作成 ... 2084
サイトのパフォーマンスの記録を開始する ... 2085
ビューのパフォーマンスの記録を開始する ... 2087
パフォーマンスの記録を表示 ... 2087
パフォーマンスの記録の解釈 ... 2087
Performance Summary (パフォーマンス サマリー) ... 2088
タイムライン ... 2088
イベント ... 2088
クエリ ... 2090
Detailed Timeline (詳細なタイムライン) ... 2091
Detailed Views (詳細ビュー) ... 2091
Depth (深度) ... 2091
CPU および Elapsed Time (経過時間) ... 2092

パフォーマンス モニタリング ツール 2093
クライアント側レンダリングの構成 2095
サポートされるブラウザー 2095
コンピューターとモバイル デバイス用の複雑性のしきい値を構成する 2096
クライアント側レンダリングを無効にする 2096
URL パラメーターを使用したテスト 2097
ビューの高速化 2097
ビューの高速化 2098
ビューの高速化が利用できない、停止している、効果がない理由 2100
ビューの高速化を利用できない 2100
ビューの高速化が停止している 2101
ビューの高速化の効果がない 2102
高速化されたビューの更新 2102
高速化されたビューのイベントベースの更新 2102
高速化されたビューのスケジュール ベースの更新 2102
サイトのビューの高速化を管理する 2103
お勧めビューをアクセラレートする 2103
アクセラレーションが推奨されているビューを管理する 2104
リソースを節約するためにアクセラレーションを自動的に一時停止する 2105
高速化されたワークブックの表示と管理 2105
ビューの高速化の通知を管理する 2106
事前に計算を行うためのユーザー コンテキストを理解する 2106

ビューの高速化のコスト 2107
抽出クエリのロードバランシング 2107
Tableau Server のモニタリング 2108
SMTP セットアップの構成 2108
セキュア SMTP 2109
TSM Web インターフェイスの使用 2109
TSM CLI の使用 2110
SMTP CLI 構成リファレンス 2112
TLS 暗号 2115
サーバーイベント通知の構成 2117
TSM Web インターフェイスの使用 2118
TSM CLI の使用 2121
個別に通知の値を設定する 2121
単一の json ファイルを使用してすべての通知の値を設定する 2123
メンテナンス 2124
バックアップと復元 2124
プラットフォームの互換性 2125
バックアップと復元のディスク容量使用率 2126
復元のディスク空き容量の要件 2127
Tableau Server のバックアップに関するベストプラクティス 2127
バックアップファイルの保護 2127
バックアップ効率を最大化する。 2128

Tableau Server の完全なバックアップと復元を実行します 2129

バックアップ データ型 2129

手動プロセスが必要なバックアップ アセット 2131

復旧用 Tableau Server のバックアップ 2132

コア Tableau Server 機能の復元 2134

その他の機能の復元 2138

復元後の抽出の再暗号化 2138

Tableau Server データのバックアップ 2138

バックアップのディスク容量使用率 2139

Tableau Server バックアップの最適化 2141

TSM コマンド ライン インターフェース (CLI) を使用したバックアップの作成 2142

アップグレード前のバックアップの作成 2143

アップグレード中のバックアップ 2143

バックアップのスケジュール設定と管理 2144

バックアップをスケジュールするには、次の手順に従います。 2144

スケジュールされたバックアップを表示するには、次の手順に従います。 2145

スケジュールされたバックアップを更新するには、次の手順に従います。 2145

バックアップ スケジュールを中断または再開するには、次の手順に従います。 2145

バックアップ プロセスのスクリプト作成 2146

ログ ファイルを削除して一時フォルダーをクリアする 2146

バックアップの実行 2146

バックアップ ファイルを別のコンピューターにコピーする 2147

バックアップからの復元 2147

Tableau Server を復元する際の制限事項 2147

バックアップ ファイルからTableau Server を復元する 2148

サーバーのメンテナンス 2150

サーバープロセス ステータスの表示 2150

TSM CLI を使用したプロセス ステータスの表示 2150

Web UI でのプロセス ステータスの表示 2150

Tableau サービス マネージャー (TSM) のステータス ページ 2151

Tableau Server のステータス ページ 2153

外部 ノード 2154

ステータスへのリモート アクセス 2156

プロセス ステータスを XML として取得 2157

XML のステータス値 2158

サーバープロセスのトラブルシューティング 2159

クラスタ コントローラー 2160

ファイル ストア 2161

インデックス作成/検索 サーバー 2162

リポジトリ 2163

VizQL Server 2164

保存されたデータ接続 パスワードのクリア 2165

すべてのサーバー ユーザーの保存されているデータ接続 パスワードをクリアにするには: 2165

サーバー上のすべての外部ディレクトリグループの同期 2166

はじめる前に 2167
外部ディレクトリグループのスケジュールに基づく同期 2167
外部ディレクトリグループのオンデマンドでの同期 2168
同期アクティビティの表示 2169
外部ディレクトリグループのユーザーに対して最小限のサイトロールを設定する 2170
参照元の外部ディレクトリからユーザーが削除されるとどうなりますか。 2171
グループ同期のパフォーマンスの向上 2171
すべてのユーザーの既定のスタートページの設定 2172
すべてのユーザーの既定のスタートページを設定するには 2172
ユーザーが設定したスタートページと階層 2173
接続されているクライアントからサイトにアクセス 2173
自動クライアント認証を無効にする 2174
不必要なファイルの削除 2175
ディスク容量使用率の監視 2175
ディスク容量使用率の削減 2175
サーバー設定 (全般とカスタマイズ) 2176
全般 2176
カスタマイズ 2190
モバイル 2190
Tableau Server PC の停止または再起動 2192
tsm コマンドライン リファレンス 2193
tsm CLI の使用 2193

tsm CLI での認証 2194
Tableau サービスマネージャー CLI へのローカルでのログイン 2194
tsm CLI へのリモートでのログイン 2194
Tableau サービスマネージャー承認グループへのアカウントの表示と追加 2195
Tableau サービスマネージャー CLI でのスクリプト作成と自動化 2195
シェルでヘルプ コンテンツを表示する 2195
シノプシス 2195
コマンド 2196
カテゴリ 2196
tsm authentication 2196
tsm authentication identity-migration configure 2197
シノプシス 2198
オプション 2198
tsm authentication kerberos <commands> 2198
シノプシス 2198
kerberos configure のオプション 2199
tsm authentication legacy-identity-mode <commands> 2199
シノプシス 2199
tsm authentication list 2199
シノプシス 2199
オプション 2199
tsm authentication mutual-ssl <commands> 2200

シノプシス 2200
オプション 2200
tsm authentication open-id <commands> 2201
シノプシス 2201
注 2202
openid configure のオプション 2202
openid map-claims のオプション 2204
注 2205
tsm authentication pat-impersonation <commands> 2205
シノプシス 2205
tsm authentication saml <commands> 2205
利用可能なコマンド 2205
tsm authentication saml configure 2206
シノプシス 2206
オプション 2206
例 2209
tsm authentication saml enable および saml disable 2209
シノプシス 2209
tsm authentication saml export-metadata 2210
シノプシス 2210
オプション 2210
tsm authentication saml map-assertions 2210

シノプシス 2210
オプション 2211
saml map-assertions の例 2211
tsm authentication sitesaml enable および sitesaml disable 2211
シノプシス 2212
tsm authentication sspi <commands> 2212
シノプシス 2212
tsm authentication trusted <commands> 2212
シノプシス 2212
オプション 2213
グローバル オプション 2213
tsm configuration 2214
「不明なキー」の応答 2215
"NULL" 値の応答 2215
tsm configuration get 2215
シノプシス 2215
オプション 2216
tsm configuration list-dynamic-keys 2216
シノプシス 2216
tsm configuration set 2216
シノプシス 2216
オプション 2217

グローバル オプション 2217
tsm configuration set のオプション 2218
tsm CLI を使用する 2219
Tsm 構成キーの基本的な使用方法 2220
構成キーを設定する 2220
構成キーを既定値にリセットする 2220
構成キーの現在の値を確認する 2220
構成キー 2221
adminviews.disabled 2221
api.server.enabled 2221
auditing.enabled 2221
backgrounder.default_run_now_priority 2222
backgrounder.enable_parallel_adsync 2222
backgrounder.externalquerycachewarmup.enabled 2222
backgrounder.externalquerycachewarmup.view_threshold 2222
backgrounder.extra_timeout_in_seconds 2223
backgrounder.default_timeout.run_flow 2223
backgrounder.failure_threshold_for_run_prevention 2223
backgrounder.flow_failure_threshold_for_run_prevention 2223
backgrounder.log.level 2223
backgrounder.querylimit 2224
backgrounder.restrict_serial_collections_to_site_level 2224

backgrounder.notifications_enabled ... 2225

backgrounder.sort_jobs_by_type_schedule_boundary_heuristics_milliSeconds ... 2225

backgrounder.subscription_failure_threshold_for_run_prevention ... 2225

backgrounder.subscription_image_caching ... 2226

backgrounder.timeout_tasks ... 2226

backgrounder.timeout.single_subscription_notify ... 2226

backgrounder.timeout.sync_ad_group ... 2226

backgrounder.vInstances_max_overflow_queue_size ... 2227

backup.zstd.thread_count ... 2227

basefilepath.backuprestore ... 2227

basefilepath.log_archive ... 2228

basefilepath.site_export.exports ... 2228

basefilepath.site_import.exports ... 2228

clustercontroller.log.level ... 2228

clustercontroller.zk_session_timeout_ms ... 2229

dataAlerts.checkIntervalInMinutes ... 2229

dataAlerts.retryFailedAlertsAfterCheckInterval ... 2229

dataAlerts.SuspendFailureThreshold ... 2229

databaseservice.max_database_deletes_per_run ... 2229

dataserver.log.level ... 2230

elasticserver.vmopts ... 2230

excel.shadow_copy_all_remote.enabled ... 2231

extractservice.command.execution.timeout 2231

features.ActiveMQ 2231

features.DeleteOrphanedEmbeddedDatabaseAsset 2232

features.DesktopReporting 2232

features.IdentityMigrationBackgroundJob 2232

features.IdentityPools 2233

features.JWTSupportForSCIM 2234

features.MessageBusEnabled 2234

features.NewIdentityMode 2234

features.PasswordlessBootstrapInit 2235

features.PasswordReset 2235

features.VizDataServiceClientEnable 2235

features.VizqlDataServiceDeployWithTsm 2236

features.VizqlDataServicePermission 2236

filestore.empty_folders_reaper.enabled 2237

filestore_empty_folders_reap.frequency_s 2238

features.Hyper_DisallowTDEPublishing 2238

filestore.log.level 2238

filestore.reapemptyfoldersholdoffms 2238

floweditor.max_datafile_upload_size_in_kb 2239

gateway.external_url 2239

gateway.http.cachecontrol.updated 2239

gateway.http.hsts 2240

gateway.http.hsts_options 2240

gateway.httpd.loglevel 2240

gateway.httpd.shmcb.size 2240

gateway.httpd.socache 2240

gateway.http.request_size_limit 2241

gateway.http.x_content_type_nosniff 2241

gateway.http.x_xss_protection 2241

gateway.log.level 2242

gateway.public.host 2242

gateway.public.port 2242

gateway.slow_post_protection.enabled 2242

gateway.slow_post_protection.request_read_timeout 2243

gateway.timeout 2243

gateway.trusted 2243

gateway.trusted_hosts 2243

hyper.file_partition_size_limit 2243

hyper.global_file_partition_size_limit 2244

hyper.enable_accesspaths_symbolic_canonicalization 2244

hyper.log_queries 2245

hyper.log_query_cpu 2245

hyper.log_timing 2245

hyper.log_troublesome_query_plans 2246
hyper.memory_limit 2246
hyper.memtracker_hard_reclaim_threshold 2246
hyper.memtracker_soft_reclaim_threshold 2247
hyper.network_threads 2247
hyper.objectstore_validate_checksums 2248
hyper.query_total_time_limit 2248
hyper.session_memory_limit 2248
hyper.srm_cpu_limit_percentage 2249
hyper_standalone.consistent_hashing.enabled 2250
hyper_standalone.health.enabled 2250
hyper.temp_disk_space_limit 2250
hyper.hard_concurrent_query_thread_limit 2251
hyper.soft_concurrent_query_thread_limit 2251
hyper.use_spooling_fallback 2252
indexandsearchserver.vmopts 2253
jmx.security.enabled 2253
jmx.ssl.enabled 2253
jmx.ssl.require_client_auth 2254
jmx.ssl.user.name 2254
jmx.ssl.user.password 2254
jmx.user.access 2255

licensing.login_based_license_management.default_requested_duration_seconds ... 2255

licensing.login_based_license_management.enabled ... 2255

licensing.login_based_license_management.max_requested_duration_seconds ... 2256

maestro.app_settings.sampling_max_row_limit ... 2256

maestro.input.allowed_paths ... 2256

maestro.output.allowed_paths ... 2257

maestro.output.write_to_mssql_using_runas ... 2259

maestro.sessionmanagement.maxConcurrentSessionPerUser ... 2259

metadata.ingestor.blocklist ... 2259

metadata.ingestor.pipeline.throttleEventsEnable ... 2260

metadata.ingestor.pipeline.throttleLimit ... 2261

metadata.ingestor.pipeline.throttlePeriodLength ... 2261

metadata.query.limits.time ... 2262

metadata.query.limits.count ... 2262

metadata.query.throttling.enabled ... 2263

metadata.query.throttling.queryCostCapacity ... 2263

metadata.query.throttling.tokenRefilledPerSecond ... 2264

metricsservices.checkIntervalInMinutes ... 2264

metricsservices.enabled ... 2264

metricsservices.failureCountToWarnUser ... 2265

metricsservices.maxFailedRefreshAttempts ... 2265

mobile.deep_linking.on_prem.enabled 2265

monitoring.dataengine.connection_timeout 2265

native_api.allowed_paths 2266

native_api.connection.limit.<connection class> 2266

native_api.connection.globallimit 2266

native_api.ExplainDataEnabled 2267

native_api.force_alternative_federation_engine 2267

native_api.ProtocolTransitionLegacyFormat 2267

native_api.unc_mountpoints 2267

native_api.InitializeQueryCacheSizeBasedOnWeights 2268

native_api.LogicalQueryCacheMaxAllowedWeight 2268

native_api.MetadataQueryCachMaxAllowedWeight 2268

native_api.NativeQueryCacheMaxAllowedWeight 2268

native_api.QueryCacheEntryMaxAllowedInPercent 2268

native_api.QueryCacheMaxAllowedMB 2269

native_api.ResourceLuidsInSnowflakeSessionQueryTagsEnabled 2269

native_api.ResourceLuidsInTeradataSessionQueryBandsEnabled 2269

native_api.UserInfoInGeneratedSQLEnabled 2270

nlp.concepts_shards_count 2270

nlp.values_shards_count 2271

nlp.defaultNewSiteAskDataMode 2272

noninteractive.vmopts 2272

pgsql.port 2272
pgsql.preferred_host 2273
pgsql.ssl.ciphersuite 2273
pgsql.ssl.max_protocol_version 2273
pgsql.ssl.min_protocol_version 2273
pgsql.verify_restore.port 2274
ports.blocklist 2274
recommendations.enabled 2274
recommendations.vizrecs.enabled 2274
redis.max_memory_in_mb 2275
refresh_token.absolute_expiry_in_seconds 2275
refresh_token.idle_expiry_in_seconds 2275
refresh_token.max_count_per_user 2275
rsync.timeout 2275
schedules.display_schedule_description_as_name 2276
schedules.display_schedules_in_client_timezone 2276
schedules.ignore_extract_task_priority 2276
searchserver.connection_timeout_milliseconds 2276
searchserver.index.bulk_query_user_groups 2277
searchserver.javamemopts 2277
searchserver.startup.zookeeper_healthcheck_timeout_ms 2277
searchserver.zookeeper_session_timeout_milliseconds 2278

ServerExportCSVMaxRowsByCols 2278

service.jmx_enabled 2278

service.max_procs 2279

service.port_remapping.enabled 2279

sheet_image.enabled 2279

ssl.ciphersuite 2279

ssl.client_certificate_login.blocklisted_signature_algorithms 2279

ssl.client_certificate_login.min_allowed.elliptic_curve_size 2280

ssl.client_certificate_login.min_allowed.rsa_key_size 2280

ssl.protocols 2280

storage.monitoring.email_enabled 2281

storage.monitoring.warning_percent 2281

storage.monitoring.critical_percent 2281

storage.monitoring.email_interval_min 2281

storage.monitoring.record_history_enabled 2281

subscriptions.enabled 2282

subscriptions.timeout 2282

svcmonitor.notification.smtp.enabled 2282

svcmonitor.notification.smtp.mime_use_multipart_mixed 2282

tabadmincontroller.auth.expiration.minutes 2283

tdsservice.log.level 2283

tomcat.http.maxrequestsize 2283

tomcat.http.proxyHost 2284
tomcat.http.ProxyPort 2284
tomcat.https.proxyHost 2284
tomcat.https.ProxyPort 2284
tomcat.https.port 2284
tomcat.server.port 2284
tomcat.useSystemProxies 2285
tomcatcontainer.log.level 2285
tsm.log.level 2285
tsm.controlapp.log.level 2285
usernotifications.reap_after_days 2285
vizportal.adsync.update_system_user 2286
vizportal.alwaysUseEmbeddedShareLinks 2286
vizportal.art_skip_list 2286
vizportal.commenting.delete_enabled 2287
vizportal.csv_user_mgmt.index_site_users 2287
vizportal.csv_user_mgmt.bulk_index_users 2287
vizportal.enable_art 2287
vizportal.log_art_java 2288
vizportal.log.level 2288
vizportal.oauth.connected_apps.max_expiration_period_in_minutes 2289
vizportal.oauth.external_authorization.enabled 2289

vizportal.oauth.external_authorization_server.blocklisted_jws_algorithms 2290

vizportal.oauth.external_authorization_server.issuer 2290

vizportal.oauth.external_authorization_server.jwks 2290

vizportal.oauth.external_authorization_server.max_expiration_period_in_minutes 2291

vizportal.openid.client_authentication 2291

vizportal.openid.essential_acr_values 2292

vizportal.openid.full_server_request_logging_enabled 2292

vizportal.openid.voluntary_acr_values 2292

vizportal.password_reset 2293

vizportal.rest_api.cors.allow_origin 2293

vizportal.rest_api.cors.enabled 2294

vizportal.site_user_group_count_enabled 2294

vizqlserver.allow_insecure_scripts 2294

vizqldataservice.vmopts 2294

vizqlserver.browser.render 2295

vizqlserver.browser.render_threshold 2295

vizqlserver.browser.render_threshold_mobile 2296

vizqlserver.clear_session_on_unload 2296

vizqlserver.force_maps_to_offline 2296

vizqlserver.geosearch_cache_size 2296

vizqlserver.initialsql.disabled 2297

vizqlserver.log.level 2297

vizqlserver.NumberOfWorkbookChangesBetweenAutoSaves 2297

vizqlserver_<n>.port 2297

vizqlserver.protect_sessions 2298

vizqlserver.querylimit 2298

vizqlserver.RecoveryAttemptLimitPerSession 2298

vizqlserver.session.expiry.minimum 2298

vizqlserver.session.expiry.timeout 2298

vizqlserver.sheet_image_api.max_age_floor 2299

vizqlserver.showdownload 2299

vizqlserver.showshare 2299

vizqlserver.url_scheme_whitelist 2299

vizqlserver.web_page_objects_enabled 2300

vizqlserver.WorkbookTooLargeToCheckpointSizeKiB 2300

vizqlserver.workflow_objects_enabled 2300

webdataconnector.refresh.enabled 2300

webdataconnector.whitelist.fixed 2301

webdataconnector.enabled 2301

webdataconnector.whitelist.mode 2301

wgserver.audit_history_expiration_days 2302

wgserver.authentication.legacy_identity_mode.enabled 2302

wgserver.authentication.identity_pools.default_pool_description 2302

wgserver.authentication.prep_builder_externalbrowser 2303

wgserver.change_owner.enabled 2304
wgserver.clickjack_defense.enabled 2304
wgserver.domain.accept_list 2304
wgserver.domain.ldap.domain_custom_ports 2305
wgserver.domain.password 2305
wgserver.domain.username 2305
wgserver.domain.whitelist 2306
wgserver.extended_trusted_ip_checking 2306
wgserver.ignore_domain_in_username_for_matching 2306
wgserver.restrict_options_method 2307
wgserver.saml.blocklisted_digest_algorithms 2307
wgserver.saml.forceauthn 2308
wgserver.saml.idpattribute.username 2308
wgserver.saml.iframed_idp.enabled 2308
wgserver.saml.maxassertiontime 2309
wgserver.saml.min_allowed.elliptic_curve_size 2309
wgserver.saml.min_allowed.rsa_key_size 2310
wgserver.saml.responseskew 2310
wgserver.saml.sha256 2310
wgserver.session.apply_lifetime_limit 2311
wgserver.session.idle_limit 2311
wgserver.session.lifetime_limit 2311

wgserver.unrestricted_ticket 2311
workerX.gateway.port 2311
workerX.vizqlserver.procs 2312
zookeeper.config.snapCount 2312
tsm customize 2312
シノプシス 2314
オプション 2314
グローバル オプション 2315
tsm data-access 2316
tsm data-access caching list 2317
シノプシス 2317
tsm data-access caching set 2317
シノプシス 2317
オプション 2318
tsm data-access repository-access disable 2318
シノプシス 2318
オプション 2318
tsm data-access repository-access enable 2319
シノプシス 2319
オプション 2319
tsm data-access repository-access list 2320
シノプシス 2320

tsm data-access set-saml-delegation configure 2320
シノプシス 2320
オプション 2320
tsm data-access set-saml-delegation disable 2321
シノプシス 2321
tsm data-access set-saml-delegation enable 2321
シノプシス 2321
tsm data-access web-data-connectors add 2321
シノプシス 2321
オプション 2322
tsm data-access web-data-connectors allow 2323
シノプシス 2323
オプション 2323
tsm data-access web-data-connectors delete 2323
シノプシス 2323
オプション 2324
tsm data-access web-data-connectors list 2324
シノプシス 2324
オプション 2324
グローバル オプション 2324
tsm email 2325
tsm email test-smtp-connection 2325

シノプシス 2326
グローバル オプション 2326
tsm initialize 2327
シノプシス 2327
オプション 2327
グローバル オプション 2328
tsm jobs 2329
tsm jobs cancel 2329
シノプシス 2329
オプション 2329
tsm jobs list 2329
シノプシス 2330
オプション 2330
tsm jobs reconnect 2330
シノプシス 2330
オプション 2330
グローバル オプション 2330
tsm licenses 2331
tsm licenses activate 2332
シノプシス 2332
オプション 2332
tsm licenses atr-configuration get 2333

シノプシス 2333
オプション 2333
tsm licenses atr-configuration set 2333
シノプシス 2333
オプション 2333
tsm licenses deactivate 2333
シノプシス 2334
オプション 2334
tsm licenses get-offline-activation-file 2334
シノプシス 2334
オプション 2335
tsm licenses get-offline-deactivation-file 2335
シノプシス 2335
オプション 2335
tsm licenses list 2336
シノプシス 2337
tsm licenses refresh 2337
シノプシス 2337
グローバル オプション 2337
tsm login 2338
シノプシス 2339
グローバル オプション 2339

tsm logout 2340
シノプシス 2340
グローバル オプション 2340
tsm maintenance 2341
tsm maintenance backup 2342
シノプシス 2343
オプション 2343
例 2346
tsm maintenance cleanup 2347
シノプシス 2347
オプション 2347
例 2350
tsm maintenance jmx disable 2350
シノプシス 2350
オプション 2350
tsm maintenance jmx enable 2351
シノプシス 2351
オプション 2351
tsm maintenance metadata-services disable 2352
シノプシス 2353
オプション 2353
tsm maintenance metadata-services enable 2353

シノプシス 2354
オプション 2354
tsm maintenance metadata-services get-status 2354
シノプシス 2354
tsm maintenance reindex-search 2354
シノプシス 2354
オプション 2355
tsm maintenance reset-searchserver 2355
シノプシス 2355
オプション 2355
tsm maintenance restore 2355
シノプシス 2356
オプション 2356
tsm maintenance send-logs 2357
シノプシス 2357
オプション 2358
tsm maintenance snapshot-backup complete 2358
シノプシス 2359
オプション 2359
tsm maintenance snapshot-backup prepare 2359
シノプシス 2359
オプション 2360

tsm maintenance snapshot-backup restore ... 2360
シノプシス ... 2360
オプション ... 2360
tsm maintenance validate-backup-basefilepath ... 2361
シノプシス ... 2361
オプション ... 2361
tsm maintenance validate-resources ... 2361
シノプシス ... 2361
オプション ... 2362
tsm maintenance ziplogs ... 2362
シノプシス ... 2362
オプション ... 2362
グローバル オプション ... 2366
tsm pending-changes ... 2367
tsm pending-changes apply ... 2367
シノプシス ... 2367
オプション ... 2368
tsm pending-changes discard ... 2368
シノプシス ... 2368
オプション ... 2368
tsm pending-changes list ... 2369
シノプシス ... 2369

オプション ... 2369
グローバル オプション ... 2369
tsm register ... 2371
シノプシス ... 2371
オプション ... 2371
グローバル オプション ... 2371
tsm reset ... 2372
シノプシス ... 2373
オプション ... 2373
グローバル オプション ... 2373
tsm restart ... 2375
シノプシス ... 2375
オプション ... 2375
グローバル オプション ... 2375
tsm schedules ... 2376
tsm schedules delete ... 2377
シノプシス ... 2377
オプション ... 2377
tsm schedules list ... 2377
シノプシス ... 2377
オプション ... 2378
tsm schedules resume ... 2378

シノプシス 2378
オプション 2378
tsm schedules suspend 2379
シノプシス 2379
オプション 2379
tsm schedules update 2379
シノプシス 2379
オプション 2380
グローバル オプション 2380
tsm security 2381
前提条件 2383
tsm security authorize-credential-migration 2383
シノプシス 2384
オプション 2384
例 2385
tsm security cancel-credential-migrations 2385
シノプシス 2385
オプション 2386
tsm security custom-cert add 2386
シノプシス 2386
オプション 2386
tsm security custom-cert delete 2387

シノプシス ... 2387
tsm security custom-cert list ... 2387
シノプシス ... 2387
tsm security custom-indexandsearch-ssl add ... 2387
シノプシス ... 2388
tsm security custom-indexandsearch-ssl list ... 2388
シノプシス ... 2388
tsm security custom-tsm-ssl disable ... 2388
シノプシス ... 2388
tsm security custom-tsm-ssl enable ... 2389
シノプシス ... 2390
tsm security custom-tsm-ssl list ... 2390
シノプシス ... 2390
tsm security external-ssl disable ... 2390
シノプシス ... 2390
tsm security external-ssl enable ... 2390
シノプシス ... 2390
オプション ... 2390
tsm security external-ssl list ... 2392
シノプシス ... 2392
tsm security kms set-mode aws ... 2392
シノプシス ... 2393

オプション 2393
例 2393
tsm security kms set-mode azure 2393
シノプシス 2394
オプション 2394
例 2394
tsm security kms set-mode local 2394
シノプシス 2394
tsm security kms status 2394
シノプシス 2395
tsm security maestro-rserve-ssl disable 2396
tsm security maestro-rserve-ssl enable 2396
シノプシス 2396
オプション 2396
tsm security maestro-tabpy-ssl disable 2397
tsm security maestro-tabpy-ssl enable 2397
シノプシス 2397
オプション 2398
tsm security regenerate-internal-tokens 2398
シノプシス 2399
オプション 2399
tsm security repository-ssl disable 2400

シノプシス 2400
tsm security repository-ssl enable 2400
シノプシス 2400
オプション 2400
tsm security repository-ssl get-certificate-file 2401
シノプシス 2401
オプション 2402
tsm security repository-ssl list 2402
シノプシス 2402
tsm security rotate-coordination-service-secrets 2402
シノプシス 2402
オプション 2402
グローバル オプション 2403
tsm settings 2404
tsm settings clone 2405
シノプシス 2405
オプション 2405
tsm settings export 2405
シノプシス 2406
オプション 2406
tsm settings import 2406
シノプシス 2407

オプション 2407
グローバル オプション 2407
tsm sites 2408
tsm sites export 2409
シノプシス 2410
オプション 2410
tsm sites import 2411
シノプシス 2411
オプション 2411
tsm sites import-verified 2413
シノプシス 2413
オプション 2413
tsm sites unlock 2413
オプション 2414
グローバル オプション 2414
tsm start 2415
シノプシス 2415
オプション 2416
グローバル オプション 2416
tsm status 2417
シノプシス 2417
オプション 2417

グローバル オプション 2419
tsm stop 2420
シノプシス 2420
オプション 2420
グローバル オプション 2421
tsm topology 2422
tsm topology cleanup-coordination-service 2423
シノプシス 2424
オプション 2424
tsm topology deploy-coordination-service 2424
シノプシス 2424
オプション 2424
tsm topology external-services gateway disable 2425
シノプシス 2425
オプション 2425
tsm topology external-services gateway enable 2425
シノプシス 2425
オプション 2426
tsm topology external-services gateway update 2426
シノプシス 2426
オプション 2426
tsm topology external-services list 2426

シノプシス 2427

オプション 2427

tsm topology external-services repository disable -n nodeN 2427

シノプシス 2427

オプション 2427

tsm topology external-services repository enable 2427

シノプシス 2428

オプション 2428

tsm topology external-services repository replace-host 2429

シノプシス 2429

オプション 2430

tsm topology external-services storage disable 2430

シノプシス 2430

オプション 2431

tsm topology external-services storage enable 2431

シノプシス 2431

オプション 2431

tsm topology external-services storage switch-share 2431

シノプシス 2432

オプション 2432

tsm topology failover-repository 2432

シノプシス 2432

オプション 2432

tsm topology filestore decommission 2433

シノプシス 2433

オプション 2433

tsm topology filestore recommission 2434

シノプシス 2434

オプション 2434

tsm topology list-nodes 2435

シノプシス 2435

オプション 2435

tsm topology list-ports 2435

シノプシス 2435

オプション 2435

tsm topology node-nickname list 2436

シノプシス 2436

オプション 2436

tsm topology node-nickname remove 2436

シノプシス 2436

オプション 2436

tsm topology node-nickname set 2437

シノプシス 2437

オプション 2437

tsm topology nodes get-bootstrap-file 2437
シノプシス 2438
オプション 2438
tsm topology remove-nodes 2439
シノプシス 2439
オプション 2439
tsm topology set-node-role 2440
シノプシス 2440
オプション 2440
tsm topology set-ports 2442
シノプシス 2442
オプション 2442
tsm topology set-process 2443
シノプシス 2443
オプション 2444
tsm topology toggle-coordination-service 2444
シノプシス 2444
オプション 2445
グローバル オプション 2445
tsm user-identity-store 2446
tsm user-identity-store get-group-mappings [options] 2446
シノプシス 2447

tsm user-identity-store get-user-mappings [options] 2447
シノプシス 2447
tsm user-identity-store list [options] 2447
シノプシス 2447
オプション 2447
tsm user-identity-store set-connection [options] 2447
シノプシス 2447
オプション 2447
tsm user-identity-store set-group-mappings [options] 2449
シノプシス 2449
オプション 2449
tsm user-identity-store set-user-mappings [options] 2450
シノプシス 2450
オプション 2451
tsm user-identity-store verify-group-mappings [options] 2452
シノプシス 2452
オプション 2452
tsm user-identity-store verify-user-mappings [options] 2452
シノプシス 2453
オプション 2453
グローバル オプション 2453
tsm version 2454

シノプシス 2454
グローバル オプション 2454
tsm ファイル パス 2455
ファイルの既定の場所 2456
現在のファイルの場所を取得する 2457
現在のファイルの場所を変更する 2458
エンティティ定義とテンプレート 2460
構成ファイルの例 2460
エンティティとキー 2461
gatewaySettings エンティティ 2462
ゲートウェイの設定 2462
構成テンプレート 2463
構成ファイル リファレンス 2463
identityStore エンティティ 2465
はじめる前に 2465
構成テンプレート 2466
ローカル 2467
重要 2467
LDAP - Active Directory 2467
OpenLDAP - GSSAPI バインド 2469
OpenLDAP - シンプル バインド 2471
構成テンプレート リファレンス 2472

共有されるアイデンティティストア オプション 2472
LDAP GSSAPI バインド オプション 2472
LDAP シンプル バインドのオプション 2473
LDAPS とサブドメイン 2474
共有 LDAP オプション。 2474
identityStoreSchemaType オプション 2475
JSON ファイルのインポート 2477
kerberosSettings エンティティ 2478
構成テンプレート 2478
構成ファイル リファレンス 2479
mutualSSLSettings エンティティ 2479
構成テンプレート 2480
構成ファイル リファレンス 2481
openIDSettings エンティティ 2484
構成テンプレート 2484
構成ファイル リファレンス 2485
samlSettings エンティティ 2488
テンプレートのカテゴリーと定義 2489
samlSettings 構成テンプレート 2489
SAML 構成エンティティ リファレンス 2490
Tableau Server に構成ファイルを渡す 2495
関連項目 2496

sapHanaSettings エンティティ 2496
SAP HANA SAML の設定 2496
構成テンプレート 2496
構成ファイル リファレンス 2497
shareProductUsageDataSettings エンティティ 2498
構成テンプレート 2499
構成ファイル リファレンス 2499
trustedAuthenticationSettings エンティティ 2500
構成テンプレート 2500
構成ファイル リファレンス 2501
web-data-connector-settings エンティティ 2503
Web データコネクタの設定 2504
構成テンプレート 2504
単一の WDC 2504
複数の WDC 2505
構成ファイル リファレンス 2505
tabcmd 2506
tabcmd のインストール 2507
tabcmd の使用方法 2512
例 2513
ステータス メッセージおよびログ 2514
tabcmd コマンド 2514

addusers group-name 2515
オプション 2516
グローバル オプション 2516
createextracts 2519
オプション 2519
グローバル オプション 2520
creategroup group-name 2523
グローバル オプション 2523
createproject project-name 2525
オプション 2526
グローバル オプション 2526
createsite site-name 2529
オプション 2529
グローバル オプション 2530
createsiteusers filename.csv 2533
ローカル認証 2534
Active Directory 認証 2534
オプション 2534
グローバル オプション 2536
createusers filename.csv 2539
ローカル認証 2539
Active Directory 認証 2539

オプション ... 2540
グローバル オプション ... 2541
decryptextracts ... 2544
グローバル オプション ... 2544
delete workbook-name または delete datasource-name ... 2547
オプション ... 2547
グローバル オプション ... 2548
deleteextracts ... 2550
オプション ... 2550
グローバル オプション ... 2551
deletegroup group-name ... 2554
グローバル オプション ... 2554
deleteproject project-name ... 2557
オプション ... 2557
グローバル オプション ... 2557
deletesite site-name ... 2560
グローバル オプション ... 2560
deletesiteusers filename.csv ... 2563
グローバル オプション ... 2564
deleteusers filename.csv ... 2566
オプション ... 2567
グローバル オプション ... 2567

editdomain 2569

オプション 2570

グローバル オプション 2571

editsite site-name 2573

オプション 2574

グローバル オプション 2575

encryptextracts 2577

グローバル オプション 2578

export 2580

オプション 2584

グローバル オプション 2585

get url 2587

グローバル オプション 2590

initialuser 2593

オプション 2594

グローバル オプション 2594

listdomains 2596

グローバル オプション 2597

listsites 2599

オプション 2599

グローバル オプション 2600

login 2602

オプション ... 2604
グローバル オプション ... 2605
logout ... 2608
filename.twb(x)、filename.tds(x)、またはfilename.hyper のパブリッシュ ... 2608
オプション ... 2609
グローバル オプション ... 2612
publishsamples ... 2615
説明 ... 2615
構文 ... 2615
オプション ... 2615
グローバル オプション ... 2615
reencryptextracts ... 2618
グローバル オプション ... 2618
refreshextracts workbook-name または refreshextracts datasource-name ... 2621
オプション ... 2622
グローバル オプション ... 2624
reset_openid_sub ... 2626
オプション ... 2627
グローバル オプション ... 2627
removeusers group-name ... 2629
オプション ... 2630
グローバル オプション ... 2630

runschedule schedule-name 2633
グローバル オプション 2633
set setting 2636
グローバル オプション 2636
syncgroup group-name 2639
オプション 2640
グローバル オプション 2641
upgradethumbnails 2643
オプション 2644
グローバル オプション 2644
validateidpmetadata 2646
オプション 2647
グローバル オプション 2647
version 2650
グローバル オプション 2650
tabcmd 用のスイッチとプロパティのインストール (Windows) 2653
トラブルシューティング 2657
Linux での Tableau Server のトラブルシューティング 2657
一般的なトラブルシューティングの手順 2658
クリーン インストール 2658
ディスク容量 2658
古いログ ファイルの削除 2658

手動のログ収集 2659

サーバーの再起動 2660

Linux を使用したインストール ファイルおよび構成ファイルの編集 2660

systemd ログの確認 2660

Tableau Server のインストール 2661

ハードウェア要件のためインストールが失敗する 2661

タイムアウトのためインストールが失敗する 2661

"Failed to initialize the instance of the temporary database" と表示されインストールに失敗する 2661

Parallels の仮想マシンでインストールが失敗する 2662

Tableau Server が起動しない 2662

インストール後に Tableau Server を起動できない 2662

複数の Active Directory (AD) ドメインを使用すると初期管理者アカウントを作成できない 2663

フォント 2663

アジア言語の文字セットへのサポート 2664

Tableau Server の初期化 2664

TSM 初期化は、tableau ユーザー アカウントは存在しますが、tableau グループのメンバーではないため失敗します。 2664

未サポートのシステム ロケール上での Tableau Server 初期化エラー 2665

en_US.utf8 がロケール リストに含まれず発生する Tableau Server 初期化エラー 2665

エラー: ステータス 10 - データディレクトリのパスにピリオドが含まれる場合の Tableau Server 初期化 2666

再インストール後の Tableau Server の初期化エラー 2666
Tableau Server のライセンス認証 2666
Tableau Server のライセンス認証の失敗 2666
Tableau Server の検索とブラウズの再インデックス 2667
検索とブラウズの再インデックスによって解決する他の問題 2667
Tableau Server の再起動 2667
Tableau Server の再起動または変更の適用が失敗する 2667
ノードの追加または構成後の Tableau Server の再起動エラー 2668
バックアップ/復元 2668
ファイルの場所 2669
basefilepath を変更しても既存のファイルの場所が変わらない 2669
TSM コマンド 2670
TSM コマンドラインが長期実行されるタスクの進捗を表示しない 2670
ファイアウォール ポートを開く 2670
Ubuntu でファイアウォール ポートを手動で開く 2670
OpenID が初回のサインイン試行で失敗する 2671
管理者ビューが表示されない 2671
ビューでのロケールの変更 2671
ログ ファイルの操作 2671
Tableau Server ログのコンテンツ 2672
Tableau Server の問題の調査 2673
Tableau Server ログとログ ファイルの場所 2675

アクティブなクラスター上の Tableau Server ログ ファイル 2676
動作中の Tableau Server インストールのプライマリログの場所 2676
動作中の Tableau Server インストールの構成ファイルの場所 2677
プライマリの場所に書き込まれないログ 2677
Zip アーカイブ内のサーバー ログ ファイル 2677
ログ ファイルのスナップショット(ログのアーカイブ) 2678
TSM Web インターフェイスの使用 2678
Tableau サポート用のログのスナップショットのアップロード 2680
TSM CLI の使用 2681
Tableau Support のログ アーカイブの送信 2683
ロギング レベルの変更 2683
ロギング レベル 2684
ロギング レベルの変更 2684
動的ログ レベルの構成 2684
ロギング レベルを変更するための設定キー 2685
ロギング レベルのリセット 2688
Tableau Server のインストールとアップグレードのトラブルシューティング 2689
一般的なトラブルシューティングの手順 2689
Tableau Server のインストールに関する全般的な問題 2690
インストール ログの場所 2690
インストールの試行が何度も失敗する 2690
ハードウェア要件のためインストールが失敗する 2691

CPU 要件が原因でインストールまたはアップグレード失敗する 2691

Tableau Server のアップグレードに関する全般的な問題 2692

アップグレードログの場所 2692

アップグレード後にマップが表示されないか、完全に表示されません 2692

アップグレードスクリプトエラー:"Tableau Server Version change validation failed. (Tableau Server のバージョン変更の検証に失敗しました。)" 2692

マルチノードをアップグレードし、追加ノードを初期化すると"Enter your credentials again (認証資格情報を再度入力してください)" というエラーで失敗する 2692

ディスク容量の不足によるアップグレードの失敗 2693

RebuildSearchIndex ジョブでアップグレードが失敗する 2693

2022.1 以降でアップグレードに失敗する 2694

2020.4.0 以降でアップグレードが失敗する 2694

バックアップ/復元ファイルの場所にパーミッションの問題があるため、アップグレードに失敗する 2695

アップグレードは成功したが、パブリッシュ済みデータソースにアクセスできない 2696

影響なし 2697

詳細 2697

共通設定のインポートに関する問題 2697

設定ファイルをインポートすると、サービスが欠落しているために "not present on any node (ノードに存在しません)" 検証エラーが発生する 2697

設定ファイルをインポートすると"configuration value you specified does not match" (指定した構成値が一致しません) エラーが発生する 2698

エラー:「調整サービスのインスタンスを直接変更することはできません」 2699

設定ファイルをインポートした後にこのエラーが表示される場合: 2700

調整サービスのプロセス カウントを手動で設定する際にエラーが表示される場合: 2701

Tableau Server の起動 2701

Tableau Server は完全に起動したことを確認できません 2701

Tableau Server が起動しない 2702

Tableau index および Tableau Server (2023.3 以降) の再構築 2702

インデックス作成/検索サーバーのインデックスを再構築することによって解決できる問題 2702

Tableau Server の検索 & 参照 (2023.1 以前の場合) のインデックス再作成 2702

検索とブラウズの再インデックスによって解決する他の問題 2702

Tableau Server のライセンス認証 2703

Tableau Server のライセンス認証の失敗 2703

ライセンシング サーバーへアクセスできることを確認する 2703

日付と時刻を確認します 2705

プロダクト キーを強制的に再度読み取る 2705

信頼されているストレージのコンテンツを Tableau サポートに送信する 2705

tabcmd のインストールの問題 2706

tabcmd を個別にインストールする 2706

Linux で tabcmd をインストールときの問題 2706

Java がインストールされていない 2706

Java のバージョンが正しくない 2707

systemd ユーザー サービスのエラー 2707

背景 2707

Tableau Server on Linux 10.5 からのアップグレード 2708
新規インストール時に発生したエラーのトラブルシューティング 2708
例 2709
サービス障害によるジョブの失敗に関するトラブルシューティング 2710
サーバーへのサインインに関する問題のトラブルシューティング 2711
シナリオのトラブルシューティング 2712
ライセンス発行のトラブルシューティング 2712
ライセンスのないサーバーの処理 2712
ライセンスを解除されたユーザーベースのサーバー 2713
ライセンスを解除されたコアベースのサーバー 2713
ライセンスを解除されたサーバー管理者 2713
ロールベース ライセンス発行のトラブルシューティング 2714
ライセンスの有効期限切れのためユーザーまたは管理者がライセンスなしである 2715
Creator ライセンスを使用する場合、サーバー管理者サイトロールは変更されない 2716
ライセンスはすぐには利用できない 2716
Viewer ライセンスを持つユーザーは、Tableau Desktop から Tableau Server または Tableau Cloud のワークブックを開くことができない 2717
ライセンスのないサーバープロセスの処理 2717
Tableau サービス マネージャー (TSM) のコマンド タイムアウト 2718
Tableau サービス マネージャー (TSM) のバックアップのトラブルシューティング 2718
サービスが起動しないため、バックアップを開始できない 2718
Cookie 制限エラー 2719

サブスクリプションのトラブルシューティング 2720
"このメール内のビュースナップショットを正しくレンダリングできませんでした。" 2720
メール内の画像を表示できない 2720
サブスクライブできない 2721
サブスクリプションアイコンがない 2721
無効または「壊れた」subscriptions 2722
添付ファイルがない 2722
サブスクリプションの一時停止 2723
一時停止したサブスクリプションの再開 2723
サブスクリプションの頻度を"データ更新時"に設定できない 2723
サブスクリプションが届かない ("メール送信エラー。コマンドをSMTP ホストに送信できない。") 2724
データ品質に関する警告または機密度ラベルが欠落している 2724
サーバー管理者参考資料 2724
Tableau Server プロセス 2724
ライセンスプロセス 2725
プロセスワークフロー 2756
Tableau Server 管理エージェント 2757
Tableau Server 管理コントローラー 2758
管理コントローラープロセスが失敗した場合の処理 2759
管理コントローラーを移動する 2759
管理コントローラーを再起動する 2759
Tableau Server のアプリケーションサーバー 2760

アプリケーション サーバーの問題のトラブルシューティング 2760
Tableau Server で ART データを有効にする 2760
ART を無効にする 2761
Tableau Server バックグラウンダー プロセス 2761
バックグラウンダー リソースの管理 2762
関連コンテンツ 2763
Tableau Server のキャッシュ サーバー 2763
Tableau Server クライアント ファイル サービス 2764
Tableau Server コレクション サービス 2766
サーバー構成 2766
マルチノード構成 2766
コレクション サービスが正常に実行されていない場合の影響 2767
コレクション サービスの 1 つのインスタンス 2767
コレクション サービスの複数のインスタンス 2767
ログ ファイル 2767
Tableau Server コンテンツ探索サービス 2767
サーバー構成 2767
マルチノード構成 2768
パフォーマンスの調整 2768
ログ ファイル 2769
検索精度 2769
インデックスの再作成 2769

Tableau Server 調整サービス 2770
調整サービスの構成 2771
調整サービス クォーラム 2771
使用する調整サービスインスタンス数 2771
ノードの数を減らす場合 2773
調整サービスのステータスの表示 2773
Tableau Server データエンジン 2774
メモリとCPU 使用率 2775
CPU 使用率 2775
メモリ使用率 2776
サーバー構成、スケーラビリティ、パフォーマンス 2776
スケーラビリティ: 2777
パフォーマンス: 2778
パフォーマンスにおけるメリット 2778
Tableau Server の Data Server 2779
Tableau Server データ ソース プロパティ サービス 2780
データ ソース プロパティ サービスのログ ファイル 2780
マルチノード クラスタのデータ ソース プロパティ サービス 2781
Tableau Server ファイル ストア 2781
decommission コマンド 2781
ファイル ストアが管理コントローラーと同じ場所にある場合のコミッション解除 2782
Tableau Server ゲートウェイ プロセス 2782

ポートの割り当て ... 2782
ゲートウェイプロセスのログファイル ... 2783
クラスターのゲートウェイプロセス ... 2783
追加情報 ... 2784
インデックス作成/検索サーバー ... 2784
サーバー構成 ... 2785
マルチノード構成 ... 2785
パフォーマンスの調整 ... 2786
インデックスの再作成 ... 2786
Tableau Server 内部データソースプロパティサービス ... 2786
内部データソースプロパティサービスのログファイル ... 2787
マルチノードクラスタでの内部データソースプロパティサービス ... 2787
Tableau Server メッセージング サービス ... 2787
メッセージング サービスが正常に実行されていない場合の影響 ... 2788
メッセージング サービスの複数のインスタンス (マルチノード、バージョン 2020.1 以降) ... 2788
メッセージング サービスの 1 つのインスタンス ... 2788
マルチノードクラスター内のメッセージング サービス ... 2788
Tableau Server メトリクス サービス ... 2788
メトリクス サービスに障害が発生した場合の影響 ... 2789
マルチノードクラスタ内のメトリクス サービス ... 2789
メトリクス サービスのログファイル ... 2790
Tableau Server マイクロサービス コンテナー ... 2790

マイクロサービス コンテナー ステータスの表示 2791
マイクロサービス コンテナー ステータス 2791
TSM Web インターフェイスの使用 2791
TSM CLI の使用 2792
Tableau Server リポジトリ 2794
優先するアクティブなリポジトリ 2794
failoverrepository コマンド 2795
Tableau Server リソース制限マネージャー 2795
Tableau Server SAML サービス 2796
Tableau Server の検索 & 参照 2797
検索と参照のプロセスのチューニング 2798
最大ヒープメモリの検索と参照 2798
既定の最大ヒープメモリ割り当て 2799
クライアントセッションのタイムアウト 2800
起動時の ZooKeeper 接続のヘルスチェックのタイムアウト 2800
Tableau 統計サービス 2801
サーバー構成 2801
Tableau 統計サービスに障害が発生した場合の影響 2802
パフォーマンス 2802
ログ ファイル 2802
Tableau Server TSM メンテナンス サービス 2802
Tableau Server VizData サービス 2803

推奨事項 2804
Tableau Server VizData ネイティブ サービス 2805
Tableau Server VizQL Data Service 2806
Tableau Server の VizQL Server 2806
Tableau Prep Conductor 2807
パフォーマンスと規模に関する推奨事項 2808
トポロジと構成 2809
Tableau Prep フロー作成 2809
パフォーマンスと規模に関する推奨事項 2810
別個のノードへのフロー作成の分離 2810
リソースの追加 2811
追加ライセンス 2811
トポロジと構成 2812
Tableau Server の動的トポロジの変更 2817
動的な構成変更 2818
シナリオの例 2818
動的トポロジの変更を行う 2818
動的トポロジの変更による影響 2819
ベストプラクティス 2819
動的トポロジの変更の自動化 2819
追加情報 2820
サーバープロセスの制限 2820

Tableau サービス マネージャーのポート 2821
エフェメラル ポートの使用 2821
ファイアウォールの要件 2821
ポートの割り当て 2822
動的ポート割り当て 2823
ポート範囲の変更 2823
範囲内の特定のポートをブロックする 2824
動的ポート割り当ての無効化 2824
手動ポート割り当て 2825
インストール時のポートの構成 2825
インストール後のポートの構成 2827
動的マッピングされないポート 2828
initialize-tsm によるポート リマッピングの制御 2829
動的マッピングされたポート 2830
JMX ポートの有効化 2839
安全な JMX ポートの有効化 2839
JMX ポートの決定のしくみ 2841
ATRDiag.exe コマンド ライン リファレンス 2841
シノプシス 2842
オプション 2842
グローバル オプション 2845
initialize-tsm スクリプトの出力ヒント 2845

出力 ... 2846
関連トピック ... 2849
upgrade-tsm Script の出力ヒント ... 2849
出力 ... 2849
サーバーのバージョン表示 ... 2851
Tableau Server Web UI からのサーバー バージョンの表示 ... 2852
TSM コマンド ラインからのサーバー バージョンとTSM バージョンの表示 ... 2852
短いバージョン、長いバージョン、version_code ... 2853
長いバージョン番号の確認 ... 2853
Einstein Discovery 統合の構成 ... 2854
Einstein Discovery ダッシュボードの拡張機能 ... 2854
Einstein Discovery の分析拡張機能 ... 2855
Einstein Discovery Tableau Prep の拡張機能 ... 2856
Salesforce.com で CORS を構成して、Einstein Discovery を Tableau Server に統合する ... 2856
Einstein Discovery 用の CORS を構成します。 ... 2857
分析拡張機能を使用した接続の構成 ... 2858
Server SSL ... 2859
分析拡張機能を有効にする ... 2859
分析拡張機能設定の構成 ... 2860
分析拡張機能接続の編集または削除 ... 2862
クライアント要件: Rserve 外部サービスの中間証明書チェーン ... 2862
スクリプト エラー ... 2862

分析拡張機能の使用状況の判定 2863
アイデンティティストアの変更 2863
警告 2864
コンテンツとパーミッションの復元方法 2864
ユーザー フィルター 2865
ユーザー名とTableau アイデンティティストア 2866
方法 1: サイトのエクスポートとインポートを使用する 2867
方法 2: フレッシュインストール — ユーザーがコンテンツを再パブリッシュする 2867
バックアップを作成し、削除してから再インストール 2867
ステップ 1: Tableau Server のバックアップ 2868
ステップ 2: Tableau Server の削除 2868
ステップ 3: 新しい認証タイプを使用して Tableau Server を再インストールする 2868
外部のアイデンティティストアの構成リファレンス 2868
構成方法 2869
Active Directory の構成 2871
構成リファレンス テーブル 2871
計算済みの configKeys 2887
サポート対象外の configKeys 2888
製品基礎データ 2888
個々のコンピューターで製品基礎データの共有を無効にする 2889
エンタープライズ レベルで製品基礎データの共有を無効にする 2889
アーカイブ済みコンテンツ 2890

これはアーカイブされたコンテンツです 2890
パブリック クラウド サービスで Tableau Server をセルフホストする 2890
これはアーカイブされたコンテンツです 2890
概要 2891
Tableau Server 上の Tableau Advanced Management について 2892
Advanced Management のライセンス要件 2892
Advanced Management プロダクト キーを有効化する 2897
実行可能なユーザー 2898
Tableau Resource Monitoring Tool について 2899
Resource Monitoring Tool とは 2899
2900
Tableau Resource Monitoring Tool の使用開始 2900
インストール前 2900
Tableau Server との製品互換性 2900
Resource Monitoring Tool サーバー(RMT サーバー) とエージェントの互換性 2901
概念と用語 2901
セットアップ アーキテクチャ 2901
Resource Monitoring Tool の最小ハードウェア要件と推奨事項 2902
Resource Monitoring Tool のインストール前チェックリスト 2903
トラブルシューティング 2903
概念 2903
エージェント 2903

Resource Monitoring Tool サーバー (RMT サーバー) 2903
環境 2904
環境ステータス 2904
OK 2904
警告 2904
重要 2904
サーバー 2905
インストール前チェックリスト - Tableau Resource Monitoring Tool 2905
マシン、ネットワーク、アカウントの要件 2905
実行可能なユーザー 2909
Tableau Resource Monitoring Tool の最小ハードウェア要件と推奨事項 2909
RMT サーバーの最小ハードウェア推奨事項 2910
Resource Monitoring Tool エージェント - リソース使用率 2910
クラウド環境でのインストール 2911
実行可能なユーザー 2911
既定のインストール パーミッション - Tableau Resource Monitoring Tool 2911
Windows インストール 2911
Linux インストール 2915
実行可能なユーザー 2933
Resource Monitoring Tool (RMT) サービス 2933
Tableau Resource Monitoring Tool のインストール 2944
インストールのバージョン履歴 2944

Resource Monitoring Tool をインストールする前に ... 2945
セットアップ アーキテクチャ ... 2947
インストールの概要 ... 2949
外部構成 ... 2949
Linux でのインストール ... 2950
HTTPS ... 2950
SSL 証明書のモードと要件 ... 2951
ファイルとディレクトリの既定の場所: ... 2952
RMT サーバー: ... 2952
エージェント: ... 2952
実行可能なユーザー ... 2952
次のステップ ... 2952
Web インターフェイスを使用した RMT サーバーのインストール ... 2952
実行可能なユーザー ... 2956
次のステップ ... 2957
Web インターフェイスを使用したエージェントのインストール ... 2957
インストールの前に ... 2957
エージェントの bootstrap ファイルをダウンロードするステップ ... 2957
エージェントをインストールするステップ ... 2958
実行可能なユーザー ... 2958
次のステップ ... 2958
コマンド ラインを使用して RMT サーバーをインストール ... 2958

Linux でのインストール 2959

Windows でのインストール 2968

Windows インストールのプロパティとスイッチ 2977

実行可能なユーザー 2980

次のステップ 2980

RMT サーバー初期化スクリプトのオプション 2981

コマンドラインを使用してエージェントをインストールする 2985

インストールの前に 2985

Linux でのインストール 2986

Windows でのインストール 2989

Windows インストールのプロパティとスイッチ 2991

マルチノードTableau Server でのエージェントのインストール 2994

実行可能なユーザー 2994

次のステップ 2994

RMT エージェントの初期化スクリプトのオプション 2994

RMT エージェントの初期化オプション 2994

Tableau Resource Monitoring Tool の外部リポジトリ 2997

Resource Monitoring Tool の新規インストール 2998

既存の Resource Monitoring Tool インストール 3001

アップグレードのベストプラクティス 3001

RMT の新しいバージョンが PostgreSQL のメジャーバージョンのアップグレードを必要とする場合のアップグレードについて 3002

アップグレードの失敗からの復元 3003

RMT とPostgreSQL のバージョンの互換性 3004
実行可能なユーザー 3004
Tableau Resource Monitoring Tool の外部メッセージ キュー サービス (RabbitMQ) 3004
Resource Monitoring Tool の新規インストール 3005
Tableau Resource Monitoring Tool の既存のインストール 3007
アップグレードのベストプラクティス 3007
RabbitMQ の TLS を有効にするアップグレード ステップ 3007
製品の互換性 3008
実行可能なユーザー 3008
Tableau Resource Monitoring Tool 前提条件 - ライセンス 3008
Resource Monitoring Tool のアップグレード 3010
アップグレードに関する注意 3010
Resource Monitoring Tool のアップグレード方法 3011
RabbitMQ の TLS を有効にするアップグレード ステップ 3014
実行可能なユーザー 3014
Resource Monitoring Tool のアンインストール 3015
remove を使用して Resource Monitoring Tool をアンインストール: 3015
tableau-rmt-obliterate スクリプトを使用して Resource Monitoring Tool を削除: 3016
実行可能なユーザー 3017
Tableau Resource Monitoring Tool の構成 3017
Resource Monitoring Tool サーバー構成 3018
インストール後のセットアップ構成 3018

通知 ... 3019
Slack 通知の設定 ... 3020
メール通知 ... 3021
接続障害のトラブルシューティング ... 3025
インシデントのしきい値 ... 3025
セキュリティ ... 3025
データ ... 3026
詳細設定 ... 3026
実行可能なユーザー ... 3026
RMT サーバー構成ファイル ... 3026
データ保持 ... 3027
SMTP 構成 ... 3027
通知構成 ... 3028
ヒストグラム構成 ... 3030
最小 TLS バージョン ... 3032
インシデント構成 ... 3033
RMT ServerLogging ... 3033
エージェント ... 3033
Tableau Server の検出 ... 3033
エージェントログ ... 3034
共通 ... 3035
暗号化されたメッセージング ... 3035

実行可能なユーザー 3036
rmtadmin コマンド ライン ユーティリティ 3036
rmtadmin add-user 3038
シノプシス 3038
オプション 3038
rmtadmin agents 3040
シノプシス 3040
オプション 3040
rmtadmin bootstrap-file 3041
シノプシス 3041
オプション 3041
rmtadmin cleanup 3042
シノプシス 3042
オプション 3042
rmtadmin create-admin-user 3043
シノプシス 3043
オプション 3043
rmtadmin create-env 3044
シノプシス 3044
オプション 3044
rmtadmin data-access 3048
シノプシス 3048

位置パラメーター 3048
オプション 3048
rmtadmin delete-env 3049
シノプシス 3049
位置パラメーター 3049
オプション 3049
rmtadmin delete-env-data 3050
シノプシス 3050
位置パラメーター 3050
オプション 3051
rmtadmin delete-server 3051
シノプシス 3051
オプション 3051
rmtadmin delete-server-data 3052
シノプシス 3052
オプション 3053
rmtadmin delete-user 3053
シノプシス 3054
オプション 3054
rmtadmin の登録解除 3054
シノプシス 3054
オプション 3054

rmtadmin deregister-agent ... 3055
シノプシス ... 3055
オプション ... 3055
rmtadmin environments ... 3056
シノプシス ... 3057
rmtadmin get ... 3057
シノプシス ... 3057
位置パラメーター ... 3057
サポートされる構成キー ... 3057
rmtadmin get-user ... 3061
シノプシス ... 3062
オプション ... 3062
rmtadmin help ... 3062
シノプシス ... 3062
rmtadmin master-setup ... 3063
シノプシス ... 3063
rmtadmin passwd ... 3071
シノプシス ... 3071
位置パラメーター ... 3072
rmtadmin query ... 3072
シノプシス ... 3072
位置パラメーター ... 3072

オプション ... 3072
rmtadmin register ... 3073
シノプシス ... 3073
位置パラメーター ... 3073
オプション ... 3073
rmtadmin restart ... 3074
シノプシス ... 3074
オプション ... 3075
rmtadmin rotate-mq-certificate ... 3076
シノプシス ... 3076
位置パラメーター ... 3076
オプション ... 3076
rmtadmin rotate-mq-certificates ... 3077
シノプシス ... 3077
オプション ... 3077
rmtadmin servers ... 3077
シノプシス ... 3077
位置パラメーター ... 3078
rmtadmin set ... 3078
シノプシス ... 3078
位置パラメーター ... 3078
サポートされる構成キー ... 3079

オプション 3082
rmtadmin start 3082
シノプシス 3082
オプション 3082
rmtadmin status 3083
シノプシス 3084
rmtadmin stop 3084
シノプシス 3084
オプション 3084
rmtadmin test-env 3085
シノプシス 3085
位置パラメーター 3086
rmtadmin update-baseline 3086
シノプシス 3086
オプション 3086
rmtadmin update-env 3087
シノプシス 3087
オプション 3087
rmtadmin users 3090
シノプシス 3090
rmtadmin version 3091
シノプシス 3091

rmtadmin ziplogs ... 3091
シノプシス ... 3091
位置パラメーター ... 3091
オプション ... 3091
グローバル オプション ... 3092
Tableau Resource Monitoring Tool の通信ポート ... 3092
RMT サーバー ... 3092
RMT サーバーの通信 ... 3092
RabbitMQ ... 3093
PostgreSQL データベース ... 3093
エージェント ... 3094
Tableau Resource Monitoring Tool レスポンス ヘッダー ... 3094
レスポンス ヘッダーの表示と更新 ... 3094
無効なヘッダー ... 3095
ユーザーの管理 ... 3095
ローカル ユーザーの追加 ... 3095
委任されたユーザーの追加 ... 3096
ユーザー認証の変更 ... 3096
Tableau Resource Monitoring Tool のサーバー ロール ... 3097
認証の問題のトラブルシューティング ... 3097
Tableau Resource Monitoring Tool - インシデント ... 3097
システム定義のインシデント ... 3098

構成可能なインシデント 3099
環境のダウン インシデント 3100
実行可能なユーザー 3101
エージェント インシデント 3101
エージェントのライセンスなし 3101
互換性のないエージェント バージョン 3102
エージェントのアップグレード: 3103
RMT サーバーのアップグレード 3104
エージェント メッセージ キューの認証資格情報ローテーションの失敗 3106
エージェントダウン 3107
エージェントのポーリングとインシデントの作成時間 3107
実行可能なユーザー 3108
抽出失敗インシデント 3108
実行可能なユーザー 3108
ハードウェア インシデント 3108
RMT サーバーWeb インターフェイスの使用 3109
構成ファイル (config.json) を使用する 3111
実行可能なユーザー 3114
Hyper スプーリング インシデント 3114
実行可能なユーザー 3114
低速クエリインシデント 3114
RMT サーバーWeb インターフェイスの使用 3114

構成ファイル (config.json) を使用する ... 3115
実行可能なユーザー ... 3116
低速ビュー インシデント ... 3116
低速ビュー インシデントのしきい値の構成 ... 3117
暗号化されたデータの収集 ... 3118
バージョン 2022.3 以降 ... 3118
Tableau リポジトリSSL の構成 ... 3118
バージョン 2022.2 以前 ... 3119
RabbitMQ の設定 ... 3119
Tableau Resource Monitoring Tool の設定 ... 3119
実行可能なユーザー ... 3120
RMT サーバーのハードウェアを変更した場合 - PostgreSQL データベースを調整する ... 3120
Tableau Server のトポロジの変更 ... 3121
ノードの追加 ... 3121
ノードの削除 ... 3121
エージェントの再登録 ... 3123
実行可能なユーザー ... 3123
Tableau Resource Monitoring Tool ログ ファイル ... 3124
ログ ファイル ... 3124
コンポーネント ... 3124
ログ レベルの構成 ... 3126
Tableau カスタマー サポートへのログ ファイルの送信 ... 3128

実行可能なユーザー ... 3128
Tableau ログ ファイル ... 3128
Tableau のアップグレード ... 3129
Resource Monitoring Tool で Tableau の新しいバージョンがサポートされていることを確認する ... 3129
エージェントの停止 ... 3129
Tableau のアップグレード ... 3130
Resource Monitoring Tool での Tableau バージョンの更新 ... 3130
エージェントの再起動 ... 3130
実行可能なユーザー ... 3130
Tableau Server のパフォーマンスを監視する ... 3130
Tableau Resource Monitoring Tool を使用して Tableau Server のパフォーマンスを監視する ... 3131
事前に設定されたチャート ... 3131
カスタム チャート ... 3133
実行可能なユーザー ... 3134
Tableau Resource Monitoring Tool のパフォーマンス チャート ... 3134
環境タブ ... 3134
パフォーマンス チャート ... 3135
Tableau プロセス チャート ... 3135
バックグラウンド タスク チャート ... 3136
同時ユーザー数 チャート ... 3136
低速のビュー読み込み要求数チャート ... 3136

ビュー読み込み要求合計数チャート 3137
サーバー タブ 3137
インサイト タブ 3138
最も低速のビュー 3138
最長の抽出更新 3138
ステータス タブ 3138
実行可能なユーザー 3138
関連トピック 3138
Tableau Resource Monitoring Tool のアクティビティページ 3139
VizQL セッション 3140
バックグラウンド タスク 3140
データ クエリ 3140
ビューの読み込み 3140
低速ビュー 3141
実行可能なユーザー 3141
関連トピック 3141
Tableau Resource Monitoring Tool のコンテンツ ページ 3142
サイト 3143
プロジェクト 3143
ワークブック 3143
ビュー 3143
実行可能なユーザー 3143

関連トピック ... 3143
低速ビュー読み込み要求の調査 ... 3144
実行可能なユーザー ... 3147
データの収集で使用されるツール ... 3147
Tableau データソースファイルを使用したモニタリングデータの探索 ... 3148
要件 ... 3149
Resource Monitoring Tool PostgreSQL データベースへのアクセスの有効化 ... 3149
Resource Monitoring Tool バージョン 2022.3 以降: ... 3149
ローカルリポジトリを持つ Resource Monitoring Tool ... 3149
外部リポジトリを持つ Resource Monitoring Tool ... 3150
Resource Monitoring Tool バージョン 2022.2 以前: ... 3150
Tableau Desktop から RMT .tds ファイルへの接続 ... 3151
実行可能なユーザー ... 3151
チャージバックレポート ... 3151
セキュリティ ... 3153
データの生成 ... 3153
実行可能なユーザー ... 3154
Tableau Resource Monitoring Tool の問題のトラブルシューティング ... 3154
ハードウェアパフォーマンスデータが見つからない場合のトラブルシューティング ... 3154
ステップ 1: エージェントの接続ステータスを確認 ... 3155
ステップ 2: エージェントが実行中であることを確認 ... 3155
ステップ 3: エージェントが正しく構成されていることを確認 ... 3155

ステップ 4: エージェントを再起動 3156
ステップ 5: 実行アカウントの構成を確認 3156
ステップ 6: サポートへの問い合わせ 3156
実行可能なユーザー 3157
RMT サーバーのサービス中断のトラブルシューティング 3157
実行可能なユーザー 3157
Tableau Server プロセスのステータスが不明である場合のトラブルシューティング 3157
ステップ 1: Tableau Server 環境の設定の確認 3158
ステップ 2: Tableau Server マシンとプロセスの更新 3158
ステップ 3: マシン名の更新 3159
ステップ 4: サポートへの問い合わせ 3159
VizQL セッションの詳細ページに VizQL プロセスが不明と記載されています .. 3159
実行可能なユーザー 3159
ユーザー認証のトラブルシューティング 3160
RMT ユーザー認証に関する問題のトラブルシューティング 3160
ログを使用して認証に関する問題をトラブルシューティングする 3160
実行可能なユーザー 3161
Web インターフェイスのタイムアウトのトラブルシューティング 3161
実行可能なユーザー 3162
メッセージング テーブルのディスク使用量の警告のトラブルシューティング 3163
実行可能なユーザー 3163
Tableau Content Migration Tool について 3163

Content Migration Tool とは ... 3164

ヘルプとサポート ... 3164

Tableau Content Migration Tool の使用開始 ... 3164

インストール前 ... 3164

インストール要件 ... 3164

Tableau Server との互換性 ... 3165

Tableau Cloud との互換性 ... 3166

Tableau コンテンツとの互換性 ... 3166

インストール後 ... 3167

コンテンツ移行に伴う制限事項 ... 3167

移行計画を作成する ... 3167

Tableau Content Migration Tool のインストール ... 3167

インストール要件 ... 3168

Content Migration Tool のインストール ... 3168

Content Migration Tool のアップグレード ... 3169

コマンドラインからの Content Migration Tool のインストール ... 3169

スイッチのインストール ... 3169

実行可能なユーザー ... 3171

Tableau Content Migration Tool の使用 ... 3171

Tableau Content Migration Tool の使用事例 ... 3171

コンテンツの昇格 ... 3172

顧客向けコンテンツの調整 ... 3174

環境間の移行 ... 3175
外部コンテンツの共有 ... 3176
データベースの移行の検証 ... 3178
コンテンツの地理的な移行 ... 3179
サイトの統合 ... 3180
メンテナンスタスク ... 3181
古くなったコンテンツのタグ付け ... 3181
コンテンツの復元 ... 3182
部分的なバックアップ ... 3182
移行計画の概要 ... 3183
コンテンツ移行に伴う制限事項 ... 3183
暗号化キー ... 3184
移行プロセス ... 3184
ステップ 1: 開始 ... 3184
ステップ 2: 計画の作成 ... 3185
ステップ 3: 移行 ... 3185
パブリッシュされたワークブック ... 3187
パブリッシュされたデータソース ... 3187
出力 ... 3187
エラーと警告 ... 3188
実行可能なユーザー ... 3189
移行に関する制限事項 ... 3189

Tableau コンテンツとの互換性 ... 3189
構成 ... 3189
データ接続 ... 3190
サポートされないコンテンツ ... 3191
移行計画: サイト ... 3194
必要なパーミッションとライセンス ... 3194
ステップ 1: ソース ... 3195
展開元サイトへのサインイン ... 3195
ステップ 2: 展開先 ... 3196
保存された接続 ... 3196
保存されている接続の追加または編集 ... 3197
個人用アクセストークンを使用して保存された接続を追加する ... 3199
ステップ 3: 次のステップに進む ... 3200
実行可能なユーザー ... 3200
移行計画: ソースプロジェクト ... 3200
ステップ 1: ソースプロジェクトを選択する ... 3200
ステップ 2: プロジェクトオプションを選択する ... 3201
ステップ 3: 次のステップに進む ... 3202
実行可能なユーザー ... 3202
移行計画: ワークブック ... 3203
ステップ 1: ワークブックの選択 ... 3203
特定のワークブックの選択 ... 3204

すべて選択 3205
表示: 3205
サムネイル 3205
リスト 3205
ルール ベースの選択 3205
Workbooks in projects 3206
Workbooks tagged with 3206
Workbooks published by 3206
すべてのワークブックの選択 3206
ステップ 2: ワークブックのマッピング 3207
ワークブック名の変更 3208
プロジェクトの変更 3208
Add Project (プロジェクトの追加) 3208
プレフィックスの変更 3209
サフィックスの変更 3210
ステップ 3: ワークブックの変換 3210
アクション URL の置換 3212
例: 3212
パラメーター値の設定 3213
イメージの削除 3213
ツールヒント コマンドの削除 3213
イメージの置換 3213

例: 3214
ズーム コントロールの表示 3214
Web ページ URL の置換 3214
例: 3214
ステップ 4: データ ソースの変換 3215
Set Calculation Formula (計算式の設定) 3218
Set Connection Info (接続情報の設定) 3218
Set Custom SQL (カスタム SQL の設定) 3219
Remove Extract (抽出の削除) 3220
保存済み認証資格情報の適用 3220
ステップ 5: パブリッシュ オプション 3220
Reset Dashboard Selections (ダッシュボードの選択をリセットする) 3221
Overwrite Newer Workbooks (新しい方のワークブックを上書きする) 3221
Copy Workbook Permissions (ワークブックのパーミッションをコピーする) 3221
抽出更新スケジュールのコピー 3221
ワークブックの認証資格情報をコピーする 3222
Copy Workbook Owners (ワークブックの所有者をコピーする) 3222
ユーザー マッピングの適用 3222
オプションの追加 3222
Add Tags (タグの追加) 3224
Remove Tags (タグの削除) 3225
Apply Extract Refresh Schedules (抽出更新スケジュールの適用) 3226

パーミッションの設定 ... 3227
Set Generate Thumbnail As (名前を付けてサムネイルの生成を設定) ... 3228
ステップ 6: 次のステップに進む ... 3229
実行可能なユーザー ... 3229
移行計画: パブリッシュ済みデータソース ... 3229
ステップ 1: 選択 ... 3230
ステップ 2: マッピング ... 3230
削除 ... 3231
名前 ... 3231
プロジェクト ... 3231
展開先名 ... 3231
展開先プロジェクト ... 3232
ステップ 3: データソースの変換 ... 3232
表/スキーマ名の置換 ... 3234
Set Calculation Formula (計算式の設定) ... 3234
Set Connection Info (接続情報の設定) ... 3235
Set Custom SQL (カスタム SQL の設定) ... 3235
Remove Extract (抽出の削除) ... 3236
Tableau Bridge の使用 ... 3236
保存済み認証資格情報の適用 ... 3236
ステップ 4: パブリッシュオプション ... 3236
より新しいデータソースの上書き ... 3237

データソース権限のコピー ... 3237
抽出更新スケジュールのコピー ... 3237
データソースの埋め込みの認証資格情報をコピーする ... 3238
データソース所有者のコピー ... 3238
ユーザー マッピングの適用 ... 3238
オプションの追加 ... 3238
Remove Tags (タグの削除) ... 3240
Add Tags (タグの追加) ... 3240
Apply Extract Refresh Schedules (抽出更新スケジュールの適用) ... 3241
パーミッションの設定 ... 3242
ステップ 5: 次のステップに進む ... 3243
実行可能なユーザー ... 3243
移行計画: パーミッションと所有権 ... 3243
マッピングの制限事項 ... 3244
ステップ 1: マッピングを追加する ... 3244
ドメイン マッピング ... 3245
ユーザー マッピング ... 3245
グループ マッピング ... 3246
CSV ファイルからマッピングをインポート ... 3246
CSV ファイル形式の要件 ... 3246
ユーザー パーミッションのマッピングをインポート ... 3247
CSV インポートの例 ... 3248

ステップ 2: マッピング順序の変更 3250
ステップ 3: 次のステップに進む 3250
実行可能なユーザー 3250
移行計画: 移行スクリプト 3251
ステップ 1: 移行前 3251
Working Directory 3251
Run 3251
Command Executable 3252
Command Parameters 3252
スクリプト 3252
ステップ 2: 移行後 3252
Working Directory 3253
Run 3253
Command Executable 3253
コマンドパラメーター 3253
スクリプト 3253
ステップ 3: 次のステップに進む 3253
実行可能なユーザー 3254
移行計画: 計画のオプション 3254
ステップ 1: オプションの構成 3254
抽出更新の除外 3255
ステップ 2: バージョン管理 3256

ステップ 3: 計画の保存 3257
ステップ 4: 次のステップに進む 3257
実行可能なユーザー 3257
抽出を使用するワークブックとデータソースの移行 3258
抽出を使用するデータ接続の変更 3258
オプション 1: パブリッシュ済みデータソースの使用 3259
オプション 2: 移行中に抽出を削除 3259
オプション 3: 移行後に抽出を更新 3259
実行可能なユーザー 3260
認証資格情報が埋め込まれたワークブックとデータソースの移行 3261
概要 3261
埋め込み認証資格情報の移行を許可する 3262
Tableau Cloud 3262
TSM コマンドライン インターフェイス 3262
Content Migration Tool 3263
トラブルシューティング 3264
埋め込み認証資格情報を移行するオプションがない 3264
埋め込み認証資格情報の移行に失敗した 3264
実行可能なユーザー 3264
Tableau Content Migration Tool コンソール ランナーの使用 3265
計画の実行 3266
使用可能なオプション: 3266

終了コード: 3266
計画の概要を表示 3266
help 3266
version 3266
encryption 3267
改善 3267
例 3267
license 3267
例 3267
script-warning 3268
例 3268
実行可能なユーザー 3268
例: 移行計画のスクリプト作成 3269
実行可能なユーザー 3271
Tableau Content Migration Tool コマンド ライン インターフェイスの使用 3271
migrate 3271
help 3272
例 3272
license 3272
例 3272
update 3272
例 3273

version 3273
実行可能なユーザー 3273
Tableau Content Migration Tool の設定 3273
実行可能なユーザー 3275
Tableau Content Migration Tool ログ ファイル 3276
Content Migration Tool ログ ファイルの場所 3276
実行可能なユーザー 3277
アクティビティ ログ 3277
アクティビティ ログのイベントへのアクセス 3278
アクティビティ ログを使用した権限の監査 3278
ログの形式 3278
例 3279
イベント 3280
アクティビティ ログのサイト イベント タイプのリファレンス 3280
イベント タイプの詳細 3280
共通の属性 3281
add_delete_user_to_group 3282
content_owner_change 3283
create_delete_group 3283
create_permissions 3284
delete_all_permissions 3285
delete_permissions 3285

delete_permissions_grantee ... 3286
display_sheet_tabs ... 3287
move_content ... 3287
project_lock_unlock ... 3288
set_permissions ... 3288
update_permissions ... 3289
update_permissions_template ... 3291
user_create_delete ... 3292
Tableau Server のキー管理システム ... 3292
Tableau Server のローカル KMS ... 3293
構成のトラブルシューティング ... 3294
マルチノードの構成ミス ... 3294
Tableau Server で RMK とMEK を再生成する ... 3294
AWS キー管理システム ... 3294
保管時の暗号化用に使用できるAWS KMS ... 3295
Tableau Server の暗号化された抽出用に AWS KMS を構成する ... 3296
ステップ 1: AWS で CMK を作成して Tableau Server 用のキーポリシーを設定する ... 3296
ステップ 2: AWS 構成パラメーターを収集する ... 3297
ステップ 3: AWS KMS 用に Tableau Server を構成する ... 3297
ステップ 4: 保存中の暗号化を有効にする ... 3298
ステップ 5: インストールを検証する ... 3298
構成のトラブルシューティング ... 3299

マルチノードの構成ミス ... 3299
AWS CMK を更新する ... 3300
Tableau Server で RMK とMEK を再生成する ... 3300
AWS KMS を使用してバックアップと復元を実行する ... 3300
Azure Key Vault ... 3301
保存中の暗号化用の Azure キー コンテナ ... 3302
Tableau Server の暗号化された抽出用に Azure キー コンテナを構成する ... 3302
ステップ 1: Azure で Tableau サーバーのキー コンテナとキーを作成する ... 3303
ステップ 2: Azure 構成パラメーターを収集する ... 3303
ステップ 3: Azure キー コンテナ用に Tableau Server を構成する ... 3303
ステップ 4: 保存中の暗号化を有効にする ... 3304
ステップ 5: インストールを検証する ... 3304
構成のトラブルシューティング ... 3305
マルチノードの構成ミス ... 3305
Azure キーの更新 ... 3305
Azure キー コンテナを使用したバックアップと復元 ... 3305
Tableau Server 外部ファイル ストア ... 3306
外部ファイル ストアを使用する理由 ... 3306
外部ファイル ストアを管理する ... 3307
ライセンス管理 ... 3307
サポートされる移行シナリオ ... 3307
バックアップと復元 ... 3307

アップグレードに関する考慮事項 3308
高可用性に関する考慮事項 3308
トポロジ 3308
次へ 3310
外部ファイル ストアを使用して Tableau Server をインストールする 3310
前提条件 3311
外部ファイル ストアを使用して Tableau Server をインストールする 3312
ステップ 1: ネットワーク共有を構成する 3312
ステップ 2: TSM をダウンロードしてインストールする 3312
ステップ 3: TSM を初期化する 3314
ステップ 4: Tableau Server をライセンス認証して登録する 3314
ステップ 5: 外部ファイル ストアを有効にする 3314
ステップ 6: 最初のノードの設定を構成する 3315
ステップ 7: インストールを完了する 3315
ステップ 8: インストール後のタスク 3316
実行可能なユーザー 3316
次へ 3316
ファイル ストアを再構成する 3316
外部ファイル ストアを使用して Tableau Server を再構成する 3316
前提条件 3316
ステップ 1: Tableau Server をアップグレードする 3317
ステップ 2: Advanced Management ライセンスを認証する 3318

ステップ 3: 外部ストレージを使用するファイル ストアを構成する 3318
ローカル ファイル ストアを使用するように Tableau Server を再構成する 3319
別の外部ストレージを使用するように Tableau Server を構成する 3320
実行可能なユーザー 3321
外部ファイル ストアによるバックアップと復元 3321
バックアップ戦略: 3321
外部ファイル ストアを使用して構成された Tableau Server 3322
スナップショット バックアップの作成 3323
スナップショット バックアップの復元 3324
外部ファイル ストアと外部リポジトリを使用して構成された Tableau Server 3325
リポジトリのバックアップ 3325
オプション 1: ネットワーク共有のスナップショットにリポジトリのバックアップを含める 3326
スナップショット バックアップの作成 3326
スナップショット バックアップの復元 3327
オプション 2: リポジトリを個別にバックアップする 3328
スナップショット バックアップの作成 3329
スナップショット バックアップの復元 3330
実行可能なユーザー 3332
外部ファイル ストアのパフォーマンスに関する考慮事項 3332
実行可能なユーザー 3333
Tableau Server 外部リポジトリ 3333
外部リポジトリに関する考慮事項 3334

クラウドプラットフォーム ... 3334
要件 ... 3335
バージョニング ... 3336
トポロジ ... 3337
外部リポジトリの管理 ... 3338
ライセンス管理 ... 3338
サポートされる移行シナリオ ... 3339
バックアップと復元 ... 3339
SSL 接続 ... 3340
SSL 証明書の更新 ... 3341
高可用性に関する考慮事項 ... 3341
アップグレードに関する考慮事項 ... 3341
リポジトリのステータスの監視 ... 3342
ログの取得 ... 3344
次のステップ ... 3344
AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成 ... 3344
要件と推奨事項 ... 3345
Amazon RDS で PostgreSQL DB インスタンスを作成する ... 3346
ステップ 1: パラメーターグループを作成する ... 3346
ステップ 2: Amazon RDS で PostgreSQL インスタンスを作成する ... 3346
ステップ 3: PostgresSQL DB インスタンスのエンドポイントを取得する ... 3348
ステップ 4: SSL 証明書ファイル (.pem ファイル) をダウンロードする ... 3348

PostgreSQL DB の高可用性の構成 3349

PostgreSQL DB の災害復旧 3349

実行可能なユーザー 3350

次のステップ 3350

Azure で Azure Database PostgreSQL インスタンスを作成する 3350

要件と推奨事項 3350

Azure Database で PostgreSQL インスタンスを作成する 3351

ステップ 1: PostgreSQL 用 Azure Database インスタンスの委任サブネットを作成する 3351

ステップ 2: PostgreSQL 用 Azure Database インスタンスを作成する 3352

ステップ 3: サーバー レベルのファイアウォール ルールを構成する 3353

ステップ 4: PostgreSQL インスタンス用の Azure Database を構成する 3353

ステップ 5: PostgresSQL DB インスタンスのエンドポイントを取得する 3354

ステップ 6: SSL 証明書ファイルをダウンロードする 3354

PostgreSQL DB の高可用性の構成 3354

PostgreSQL DB の災害復旧 3354

実行可能なユーザー 3355

次のステップ 3355

Google Cloud での PostgreSQL インスタンスの作成 3355

要件と推奨事項 3356

Google Cloud での PostgreSQL データベース インスタンスの作成 3356

ステップ 1: 新しい PostgreSQL インスタンスを作成する 3356

ステップ 2: PostgreSQL インスタンスのデータベース フラグを構成する 3357

ステップ 3: PostgreSQL DB インスタンスのエンドポイントを取得する 3357
ステップ 4: SSL 証明書ファイルをダウンロードする 3357
PostgreSQL DB の高可用性の構成 3357
PostgreSQL DB の災害復旧 3357
実行可能なユーザー 3358
次のステップ 3358
PostgreSQL データベースをスタンドアロン インストールで作成 3359
要件と推奨事項 3359
スタンドアロンの PostgreSQL データベース インスタンスの作成 3360
ステップ 1: PostgreSQL をインストールして初期化する 3360
ステップ 2: PostgreSQL インスタンスを設定する 3360
スーパー ユーザーの設定 3360
ネットワークとセキュリティ 3361
データベースに関するオプション 3361
パラメーターの更新 3361
リモート接続の構成 3361
SSL の構成 3361
高可用性とディザスター リカバリ 3363
実行可能なユーザー 3363
PostgreSQL 外部リポジトリでの Tableau Server のインストール 3363
インストールの前に 3364
Tableau Server のインストールと構成 3365

ステップ 1: 構成ファイルの作成 3365
ステップ 2: Tableau Server のインストールと外部リポジトリの構成 3366
ステップ 3: tsm による初期化の完了 3367
ステップ 4: インストールの完了 3368
実行可能なユーザー 3368
Tableau Server リポジトリの再構成 3368
リポジトリをローカルから外部に移動 3368
リポジトリを外部からローカルに移動 3370
実行可能なユーザー 3371
新しいメジャーバージョンの PostgreSQL を外部リポジトリとして使用する Tableau Server のアップグレード 3371
アップグレードする前に 3371
Tableau Server のアップグレード 3373
製品の互換性 3374
実行可能なユーザー 3377
RDS インスタンスのアップグレード 3377
実行可能なユーザー 3378
ノードロールによるワークロード管理 3378
バックグラウンダー ノードロール 3378
バックグラウンダー ノードロールの使用 3379
構成オプション 3379
ライセンスの要件 3380
考慮事項 3381

ファイル ストア ノード ロール 3382

抽出更新ワークロードとバックアップまたは復元ワークロードを最適化するための
ガイドライン 3383

抽出クエリワークロード管理の微調整 3385

構成オプション 3386

ライセンスの要件 3387

ノードロールの表示方法 3387

実行可能なユーザー 3387

Tableau Server の独立したゲートウェイ 3387

独立したゲートウェイを使用する理由 3388

独立したゲートウェイの管理 3388

ライセンス管理 3388

バックアップと復元 3388

高可用性に関する考慮事項 3389

トポロジ 3389

次へ 3389

Tableau Server と独立したゲートウェイのインストール 3389

前提条件 3389

Tableau Server とIndependent Gateway のインストール 3390

ステップ 1: Tableau Server のダウンロードとインストール 3391

ステップ 2: Independent Gateway のダウンロードとインストール 3391

Tableau Server Independent Gateway の初期化 3392

ステップ 3: Tableau Server で独立ゲートウェイを有効化 3393

Independent Gateway JSON ファイルの内容 3394
Independent Gateway の認証シークレット 3394
Independent Gateway の JSON ファイルの例 3394
Tableau Server で Independent Gateway を有効化 3395
ステップ 4: Tableau Server で Independent Gateway を確認 3395
Independent Gateway を使用した Tableau Server の構成 3396
直接接続とリレー接続 3396
直接接続 3396
構成 3397
ポート入力の管理 3397
リレー接続 3397
構成 3398
ハウスキーピング プロトコル 3398
HK ポートの変更 3399
ログ ファイルの場所 3399
トラブルシューティング 3400
Independent Gateway を使用した認証モジュールの構成 3400
認証モジュールの構成例 3401
構成プロパティ 3401
<Location "/tsighk"> ブロック 3402
カスタム認証モジュール構成のトラブルシューティング 3403
独立したゲートウェイで TLS を構成する 3404

TLS 構成の例 ... 3404
TLS 構成の概要 ... 3404
証明書の要件と考慮事項 ... 3405
グローバルな TLS の構成 ... 3405
独立したゲートウェイへの外部 TLS ... 3406
ステップ 1: 独立したゲートウェイのコンピューターにファイルを配布する ... 3407
ステップ 2: 独立したゲートウェイのコンピューターの環境変数を更新する ... 3407
ステップ 3: Tableau Server で TLS 構成プロパティを設定する ... 3408
Tableau Server への独立したゲートウェイ ... 3410
ステップ 1: Tableau Server で TLS を構成し、有効にする ... 3410
ステップ 2: 独立したゲートウェイのコンピューターで証明書ファイルを配布する ... 3411
ステップ 3: Tableau Server で TLS 構成プロパティを設定する ... 3411
ステップ 4: ルート CA 証明書を Tableau Server にアップロードする ... 3415
Tableau Server と独立したゲートウェイの間のハウスキーピング接続 ... 3415
ステップ 1: 独立したゲートウェイのコンピューターにファイルを配布する ... 3416
ステップ 2: 独立したゲートウェイのルート CA 証明書を Tableau Server の信頼ストアにインポートする ... 3417
ステップ 3: 独立したゲートウェイのコンピューターの環境変数を更新する ... 3417
ステップ 4: 独立したゲートウェイで httpd.conf.stub を更新する ... 3418
ステップ 5: TLS 構成プロパティを Tableau Server に設定する ... 3419
ステップ 6: 独立したゲートウェイの JSON 構成ファイルを更新する ... 3422
トラブルシューティング ... 3423
Tableau Server の独立したゲートウェイのアップグレード ... 3423

概要 ... 3423
ステップ 1: 参照用にファイルをコピーする ... 3424
ステップ 2: 独立したゲートウェイを削除する ... 3424
ステップ 3: 独立したゲートウェイをインストールする ... 3424
ステップ 4: (オプション) tsighk-auth ファイルを元のコピーで上書きする ... 3425
ステップ 5: ハウスキーピング TLS 設定を更新する ... 3425
ステップ 6: (オプション) バックエンドの Tableau Server 展開を更新する ... 3426
ステップ 7: tsig-httpd サービスを再起動する ... 3426
Tableau Server の独立したゲートウェイのアンインストール ... 3427
独立したゲートウェイのアンインストール ... 3427
initialize-tsig スクリプトの出力ヒント ... 3428
出力 ... 3428
関連トピック ... 3430
Tableau Server Backgrounde のリソース制限 ... 3430
概要と概念 ... 3430
概要 ... 3430
使用するタイミング ... 3430
要件と推奨事項 ... 3431
用語と概念 ... 3431
実施できること ... 3431
Backgrounder リソース制限の設定方法 ... 3432
サイトのデフォルト制限 ... 3433

サイトのカスタム制限 3433
リソース制限を構成後の動作 3434
リソース制限を調整するタイミング 3434
実行可能なユーザー 3435
コンテナでの動的スケーリング - Tableau Server バックグラウンダー 3435
概要 3435
前提条件 3436
制限事項 3436
Tableau Server およびバックグラウンダー ポッドのイメージの作成 3436
展開ガイド 3437
バックグラウンダー ジョブ 3438
NODE_ROLE_CONFIG 3439
Tableau Server in a Container のポッド 3440
バックグラウンダー ポッド 3440
ログ 3441
バックグラウンダー ポッドの実行中にログを収集する 3441
バックグラウンダー ポッドが終了した (または開始に失敗した) ときにログを収集する 3442
ライブ構成変更 3443
スケーリング戦略 3443
スケジュールされたスケーリング 3443
Kubernetes の構成 3444
新しい環境変数 3444

バックグラウンダー ポッドのポート 3445
共有ネットワークディレクトリ 3445
Kubernetes の構成例 3446
Tableau Server コンテナの構成 3446
バックグラウンダー ポッドの構成 3456
スケジュールされたスケーリングの構成 3463
クローン構成を消去する Kubernetes ジョブ (オプション) 3464
データ管理 について 3465
データ管理 の機能 3466
Tableau Catalog 3467
Tableau Prep Conductor 3469
仮想接続とデータポリシー 3469
データ管理 のライセンス 3470
Tableau Prep Conductor 3470
Tableau Catalog 3471
仮想接続とデータポリシー 3471
データ管理 ライセンスのしくみ 3472
ユーザーベース 3472
コアベース 3472
Tableau Prep Conductor 3475
Tableau Server での Tableau Prep Conductor の有効化 3478
フローのワークスペースについて 3478

フローの [概要] ページ 3478
データ管理 のないフロー概要ページ 3481
フロー接続ページ 3481
フローのスケジュールされたタスクのページ (データ管理必須) 3482
スケジュール ページ 3483
フロー実行履歴 (データ管理必須) 3484
フロー リビジョン履歴 3485
実行可能なユーザー 3485
Tableau Server での Tableau Prep Conductor の有効化および構成 3485
Server のトポロジ 3486
次の手順: 3488
実行可能なユーザー 3488
手順 1 (新規インストール): Tableau Server を Tableau Prep Conductor と一緒にインストール 3489
インストールの前に 3489
Tableau Server のインストールおよび Tableau Prep Conductor の有効化 3489
パブリック ゲートウェイの設定を構成する 3490
Tableau Prep Conductor を有効にする 3490
Tableau Prep Conductor が有効で実行中であることの確認 3491
ノードを Tableau Prep Conductor 専用にする 3493
分散インストール 3494
次の手順 3494
実行可能なユーザー 3494

手順 1 (既存インストール): Tableau Prep Conductor の有効化 3494
アップグレードする前に 3495
アップグレードの準備: 3495
パブリックゲートウェイの設定を構成する 3495
ユーザーベース ライセンスを使用した Tableau Server インストール 3496
Tableau Server の単一ノードインストール 3496
Tableau Server の分散インストール 3499
コアベース ライセンスを使用した Tableau Server インストール 3502
Tableau Server の単一ノードインストール 3503
Tableau Server の分散インストール 3505
次の手順 3509
実行可能なユーザー 3509
手順 2 - Tableau Server のフロー設定の構成 3509
パブリッシュ、スケジュール、および認証資格情報の設定 3510
Tableau Prep Conductor の無効化が意味すること 3511
フローの失敗に関する通知の構成 3512
サーバー全体のメール通知を有効にするには 3512
通知値の設定 3513
サイトの通知を構成するには、次を実行します。 3514
次の手順 3514
実行可能なユーザー 3514
手順 3 - フロー タスクのスケジュールの作成 3515

新しいスケジュールの作成: 3515
次の手順 3516
実行可能なユーザー 3516
手順 4 - 入力場所と出力場所を許可リストに登録する 3516
入力場所と出力場所を許可リストに登録する方法 3517
次の手順 3520
実行可能なユーザー 3520
手順 5 - オプションのサーバー構成 3520
フローのタイムアウト期間を設定する 3520
フロー タスクを中断するしきい値を設定する 3521
実行可能なユーザー 3521
フロー タスクのスケジュール 3522
フロー タスクのスケジュールを設定する 3523
リンクされたタスクのスケジュール 3528
実行可能なユーザー 3538
フロー実行の成功をユーザーに通知する 3539
フロー サブスクリプションのサイト設定を構成する 3539
フローをパブリッシュする 3540
フロー サブスクリプションを追加する 3540
フローのサブスクリプションを解除する 3542
サブスクリプションを表示する 3542
一時停止したフロー サブスクリプションの再開 3543

通知メールからフローデータにアクセスする 3543
実行可能なユーザー 3544
フローの管理 3545
フローの管理 3545
実行可能なユーザー 3547
フローの正常性とパフォーマンスの監視 3557
発生した問題を検出して解決する 3557
フローが失敗したときに通知を受け取る 3557
エラーを表示して解決する 3557
フローの [概要] ページ 3559
[接続] ページ 3559
[スケジュール タスク] ページ 3560
[実行の履歴] ページ 3561
アラート 3562
Tableau Prep Conductor プロセスのステータス 3562
実行可能なユーザー 3563
フローの管理ビュー 3564
実行可能なユーザー 3564
すべてのユーザーによるアクション 3564
特定のユーザーによるアクション 3565
最近のユーザーによるアクション 3565
バックグラウンドタスクの遅延 3566

抽出以外のバックグラウンドタスク 3567
フロー実行のパフォーマンス 3568
領域使用量の統計 3569
実行可能なユーザー 3570
開発者向けリソース - REST API 3570
Tableau Catalog について 3571
Tableau Catalog のしくみ 3572
Tableau Catalog の主要な用語 3573
Tableau Catalog のライセンス 3573
Tableau Catalog の有効化 3573
特長と機能 3573
データディスカバリ 3574
キュレーションと信頼。 3574
系列とインパクト分析 3574
開発者向けリソース 3575
仮想接続とデータポリシーについて 3575
重要な用語 3576
仮想接続とデータポリシーのライセンス 3577
仮想接続とデータポリシーの有効化 3577
パーミッション 3577
パーミッションとデータポリシー 3577
パーミッションとデータポリシーが連携するしくみ 3578

特長と機能 ... 3578
仮想接続エディターのワークフロー ... 3579
次のステップ ... 3580
仮想接続の作成 ... 3581
データへの接続 ... 3581
別の接続の追加 ... 3581
接続に含めるテーブルの選択 ... 3582
テーブルのライブ モードと抽出モードの選択 ... 3582
増分抽出 ... 3583
カスタム SQL への変換 ... 3584
テーブル データの抽出 ... 3585
テーブルの表示状態の設定 ... 3586
テーブルの詳細の確認 ... 3586
データベースからデータを更新する ... 3587
実行可能なユーザー ... 3588
次のステップ ... 3588
関連項目 ... 3588
行レベルのセキュリティに使用するデータ ポリシーの作成 ... 3588
データ ポリシーについて ... 3588
ポリシー テーブルのポリシー列でフィルターリングする ... 3589
ポリシー テーブルのポリシー列を使用した例 ... 3593
資格テーブルのポリシー列でフィルターリングする ... 3593

資格テーブルのポリシー列を使用した例 3596
ポリシー条件を記述する 3596
ポリシー条件の例 3596
ポリシー条件でサポートしているTableau 関数 3597
実行可能なユーザー 3597
次のステップ 3598
リソース 3598
[ユーザーとしてプレビュー] を使用して、行レベルのセキュリティをテストする 3598
実行可能なユーザー 3599
次のステップ 3599
仮想接続のパブリッシュとパーミッションの設定 3599
下書きの保存 3599
進行中の下書き 3599
接続のパブリッシュ 3600
仮想接続でのパーミッションの設定 3600
実行可能なユーザー 3601
次のステップ 3601
仮想接続の抽出更新のスケジュールを設定する 3601
テーブルを抽出する 3602
Tableau Server での抽出更新のスケジュール 3602
更新の抽出の時間制限 3603
実行可能なユーザー 3604

次のステップ 3604
仮想接続を使用する 3604
仮想接続に接続する 3604
仮想接続またはデータポリシーを編集する 3605
参照元スキーマの変更に対応する 3605
仮想接続のリビジョン履歴を操作する 3605
仮想接続リビジョンを復元または削除する 3606
ワークブック内の既存のデータソースを仮想接続に置き換える 3607
実行可能なユーザー 3607

Tableau Server のリリース ノート

このトピックでは、最新リリースの Tableau Server の新機能について説明します。次の Viz を使用して、Tableau Server の新機能を探索します。機能をクリックすると、ツールヒントにその機能の詳細ドキュメントへのリンクが表示されます。フィルターを探索して、検索を絞り込みます。データをダウンロードして、カスタマイズされたリストを作成します。

- **機能で検索**のダッシュボードを使用して、製品またはバージョンの新機能のリストを表示したり、機能がいつリリースされたかを調べたりすることができます。ダッシュボードは現在、最新バージョンの Server の場合は、既定で Server に設定されています。
- **Server をアップグレード**のダッシュボードを使用して、アップグレードに固有の機能のリストを表示します。アップグレード ビューには、新機能または変更された機能を絞り込むことができるフィルターが含まれています。**変更済み**ステータスの下にリストされている機能は、通常、アップグレードに影響を与える可能性がある機能です。変更済みとして分類された機能のリストは、アップグレードの準備とテストに役立つ場合があります。

Tableau Release Navigator - リリース ノート
機能で検索
製品を選...
(All)
製品バー...
最新
オファリ...
Null
Tableau+ のすべての機能を表示するには、Tableau+、データ管理、Advanced Management を選択してください。
Tableau Pulse の機能をお探しですか? こちらをクリッ
ステータス
(All)
機能
機能を選択して詳細を確認する
Server のアップグレ...
Desktop のアップグ...
Prep のアップグレード
製品別とバージョン別の機能リスト
製品
リリース
ステータス
Tableau Cloud
October 2025
変更済み
新規
AI クレジット消費量に関する変更
2025 年 9 月の Tableau Cloud 向..
2025 年 10 月の Tableau Cloud 向..
[データの表示] のダウンロードを構成する
Dremio サーバーのプライベート接続
MariaDB のプライベート接続
OIDC および SAML ワークフローのユ..
Tableau Agent による多言語サポート
Tableau Agent のパーミッション
Tableau Agent のメタデータの改善
Tableau Pulse のダイジェスト サイト..
Tableau の信頼できる拡張機能
Teradata Vantage Cloud のプライベ..
Teradata Vantage オンプレミスのプラ..
フィードバックはありますか? 当社にお知らせください
View on Tableau Public
Share

展開の計画

このセクションの記事には、Tableau Server 展開の計画に関する情報が記載されています。

このセクションに記載されている記事の他に、計画ワークフローの最初のステップとして Tableau Blueprint を参照することをお勧めします。Tableau Blueprint は、データ主導型の組織に変わるための手順を示すガイドです。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

サーバー管理者の概要

Tableau Server on Linux は、ユーザーに独自のセルフサービスデータ分析文化を提供するため、IT インフラストラクチャ内の多数のコンポーネントと統合します。サーバー管理者は、Tableau Server が IT インフラストラクチャにどのように適合するかを理解しておくことが重要です。

このセクションのトピックは、Tableau Server の計画、展開、調整、管理に関する情報を提供します。

組織にデータ主導型の文化をもたらす広範囲な取り組みの一環として Tableau Server を展開する場合は、Tableau Blueprint を参照してください。Tableau Blueprint は、データに基づいた組織に変わるための手順を示すガイドです。組織で最新のセルフサービス分析を初めて使用する場合、

またはこれを既に展開済みで、データの使用をあらゆる分野で推進する必要がある場合のどちらでも役立ちます。

このトピックでは、Tableau Server についての考え方や、既存の IT インフラストラクチャとのやり取りのしくみについて短い概要を提供します。

> Tableau Server on Windows については、「サーバー管理者の概要」を参照してください。

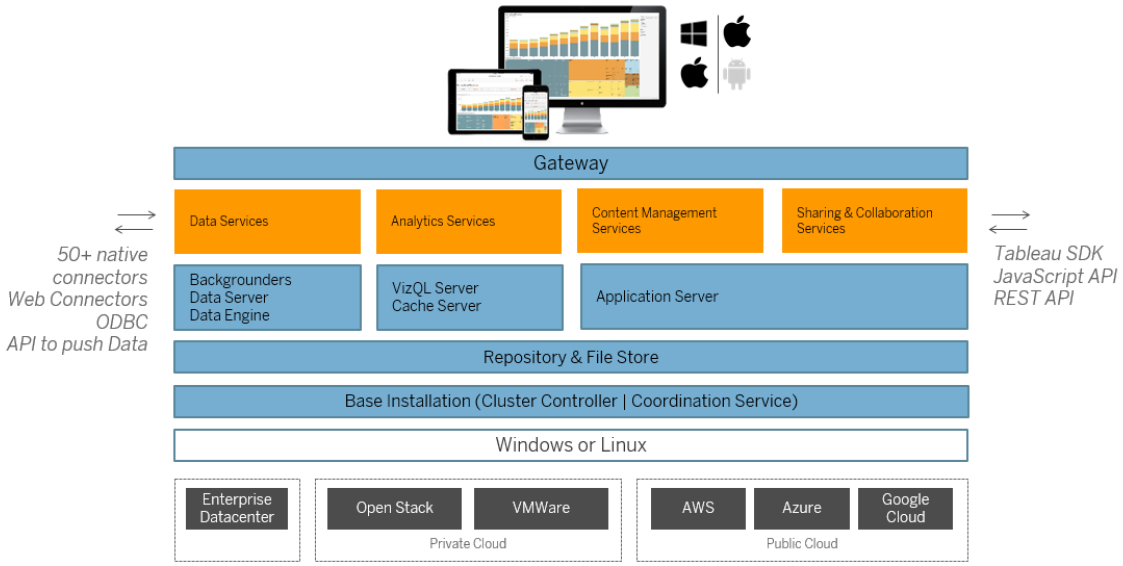
サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

アーキテクチャの概要

Tableau Server は、ユーザー用に完全なセルフサービス分析プラットフォームを提供するために連携するプロセスの集まりです。次の図は、Tableau Server の高度なアーキテクチャビューを示しています。



複数のサーバープロセス(上記では青で示しています) が連携し、さまざまな階層でサービスを提供します。ゲートウェイプロセスは、すべての Tableau クライアントからのトラフィックを、クラスター内の利用可能なサーバー ノードにリダイレクトするコンポーネントです。

データ サービスはデータの鮮度、共有メタデータ管理、管理されたデータ ソース、およびインメモリ データを提供するサービスの論理的なグループ化です。データ サービス、バックグラウンダー、データ サーバーおよびデータ エンジン プロセスに力を与える元となるプロセス。

VizQL やキャッシュ サーバー プロセスで構成されるアナリティクス サービスは、ユーザーが直面している視覚化、アナリティクス サービス、およびキャッシュ サービスを提供します。

共有、コラボレーション、およびコンテンツ管理サービスは、アプリケーション サーバー プロセスによって搭載されています。ユーザー ログイン、コンテンツ管理 (プロジェクト、サイト、パーミッションなど) および管理者アクティビティなどの主な Tableau Server 機能は、アプリケーション サーバー プロセスによって提供されます。

上記のサービスはすべて、メタデータ、パーミッション、ワークブック、データ抽出、ユーザー情報、およびその他のデータなどの構造化されたリレーショナルデータを含む、リポジトリプロセスを使用し、それらに依存しています。ファイル ストア プロセスはクラスター全体でデータ抽出ファイルの冗長性を有効にし、すべてのクラスター ノードでローカルに抽出を利用できるようにします。より高い負荷がか

かる場合、より早く処理およびレンダリングを行うため、抽出ファイルをクラスタ全体でローカルに利用できます。

Tableau のアーキテクチャは柔軟なため、どこからでもプラットフォームを実行することができます。Tableau Server をオンプレミス、プライベートクラウドやデータセンター内、Amazon EC2、Google クラウドプラットフォーム、MS Azure にインストールできます。Tableau アナリティクスプラットフォームは、仮想プラットフォーム上でも実行します。Tableau Server からの最大限のパフォーマンスを得るため、各仮想プラットフォームのベストプラクティスに従うことをお勧めします。

Tableau およびデータ

組織に Tableau Server をインストールすると、Tableau Server はユーザーが必要とするデータに対する分析パイプラインのコアコンポーネントとなります。Tableau Server とビジネスデータのやり取りの方法を理解しておくことが重要です。特に、Tableau Server は組織内にデータの抽出を保存できます。ライブデータソースにも接続できます。Tableau ユーザーへのデータの提供方法をどのように選択するかは、データソースのタイプ、ユーザーシナリオ、パフォーマンスおよびアクセス要件、およびインフラストラクチャの条件という、多数の変数から情報を得ることができます。

Tableau Server は、静的なネイティブデータファイルを格納するデータウェアハウスサーバーとして構築されていません。実際、従来のデータウェアハウスとして Tableau Server を使用している場合、投資をうまく利用できていません。それよりも、データストレージに関しては、Tableau Server で最適化されたデータ抽出をホストすることをお勧めします。データ抽出は多くの場合、組織内の大きなデータソースのサブセットですが、営業時間外に抽出の更新をスケジュール設定することで、営業時間中に過大な負担がかかるデータソースの抽出を作成することもできます。

抽出は、データのモデリングや、高性能の視覚化作成を有効にする場合にも便利です。たとえば、視覚化の作成やインタラクションのパフォーマンスを改善には、指定された部門やプロジェクトに対して重要なフィールドへとソースデータをフィルターリングして抽出を最適化する必要がある場合があります。

また、Tableau Server はライブデータソースへの直接的な認証アクセスを提供するため、ユーザーは接続されているさまざまなデータソースに対してフィルターされた複雑なクエリを構築および実行することができます。このシナリオでは、Tableau は組織内のデータソースやクラウド内のデータソースに対し、高性能のネットワークアクセスを必要とします。大容量の、複雑なデータ操作で

必要な処理負荷を扱うため、Tableau Server およびターゲットデータソースのサイズも適切に設定する必要があります。

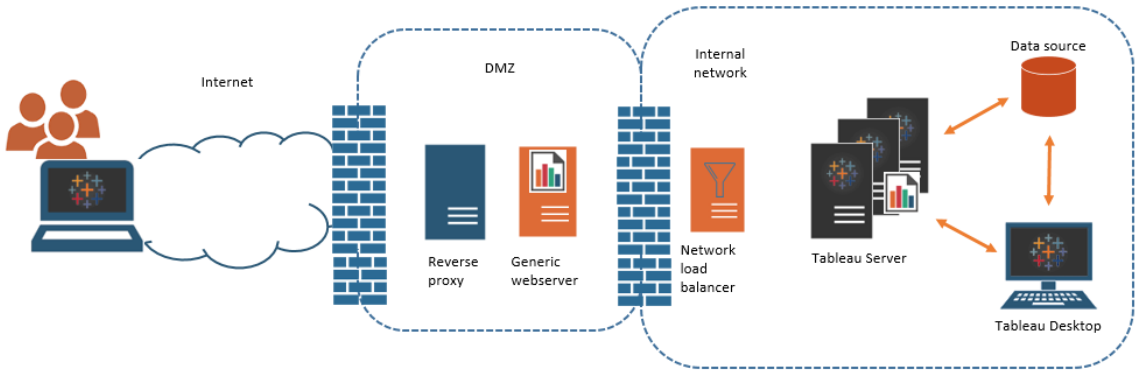
ユーザー アクセス

Tableau Server はユーザーが接続してさまざまなデバイスからデータの視覚化とデータソースを共有、表示、操作する Web ベースのコラボレーション プラットフォームでもあります。つまり、ローカルで保護されているネットワーク内の Tableau ユーザーが、Tableau Server にアクセス可能である必要があります。また、データの視覚化へのアクセスをデスクトップ、モバイル、および組織外の認証 Web ユーザーへ拡大することもできます。

Tableau Server を次のユーザー認証 ソリューションと統合します。Active Directory、SAML、OpenId、および Kerberos。

ネットワークのどこに Tableau Server をインストールすればよいか。

組織が Tableau Server で管理するほとんどのデータの機密性により、および Tableau Server は内部データストアへのアクセスが必要なため、Tableau Server は保護されたネットワーク内で実行する必要があります。インターネットからの認証アクセスは、リバース プロキシや VPN ソリューションを通じて Tableau Server に接続するよう構成されています。



組織によっては公開 Web ページに、または内部ユーザー用として社内ネットワーク上の汎用 Web ブラウザー上に Tableau ビューを埋め込んでいます。

Tableau Server は、認証アクセスまたは匿名アクセスのいずれかを使用したシナリオをサポートするよう構成できます。ユーザーがパーミッションを持つ参照元データのみを表示できる認証アクセスの場合は、汎用 Web サーバーで信頼できるチケットを構成できます。このシナリオでは、Tableau Server は埋め込みビュー内の参照元データへのアクセスを許可します。このスキーマにより、DMZ 内や、保護されているネットワーク外の Web サーバーで、インタラクティブなデータ視覚化をホストすることができます。

埋め込み Tableau ビューへの匿名アクセスでは、"ゲストユーザー" を Tableau Server に対して有効にする必要があります。また、ゲストユーザーには、名前付きユーザー (Interactor) モデルではなく、実行中のコア数に従って Tableau Server ライセンスを提供する必要があります。

サイジングと拡張性

組織の規模と組織内のデータ利用状況に応じて、Tableau Server をスケール アップまたはスケール アウトすることができます。サーバーをスケール アウトする場合は、データのニーズやユーザーのニーズを満たすため、リソースを選択的に割り当てることもできます。

Tableau Server をスケール アップする場合は、1 台のサーバーにハードウェア リソースを追加します。たとえば、Tableau Server を実行するコンピューターのメモリや処理能力を増やすことができます。

Tableau Server をスケール アウトする場合は、コンピューター (または ノード) を追加します。フェールオーバーを使用して高可用性展開を作成するには、3 つ以上のノードが必要です。たとえば、最も CPU を消費するサーバープロセスを 2 つのノード上で実行し、ゲートウェイの 3 番目のノードと、調整 コントローラー サービスを使用している場合があります。

スケール アップとスケール アウトのどちらを行う場合でも、実行するサーバープロセスの数とタイプを構成することにより、リソースを選択的に割り当てることができます。組織に大量のデータがあり、大量の抽出を作成する場合は、抽出の更新および保存専用のプロセスの数を増やすことができます。また、組織でユーザー負荷の増大にともなう最適化を行う場合は、ユーザーの要求に対する応答専用のプロセスの数を増やすことができます。さらに、Tableau Server を業界標準のネットワーク ロード バランサーに統合して、サーバーをユーザーの要求に合わせてより最適化することができます。

Tableau Server 管理モデル

Tableau Server は、サーバー管理者とサイト管理者の、2 つの高度な管理者で管理スキームをサポートするよう作られています。小さな組織では、これらの役割は同じ人物やチームが行うと想定されると想定されますが、大きな組織では多くの場合、分担されます。

このモデルでは、サーバー管理者は異種サーバーソリューションを維持および展開する IT 専門家です。サーバー管理者の重要な分野には、ネットワーキング、ハードウェアの調整やメンテナンス、セキュリティとアクセス、ユーアーやディレクトリサービスの管理が含まれる場合があります。Tableau Server に付属するサーバー管理者用のツールやドキュメントは、これらのコア サーバー IT エリアをサポートします。

一方、サイト管理者は、Tableau Server や Tableau Cloud の展開に固有の管理的役割です。Tableau サイト管理者は、基本的にデータのコンテンツに関心を持っています。サイト管理者はユーザーの管理や、プロジェクト、ワークブック、データ ソースへのユーザーのアクセスの管理を行います。サイトについて、およびサイト用の展開計画の方法の詳細については、サイトとはを参照してください。

管理者の役割

小規模企業の中には、1 人の管理者が Tableau Server 全体を管理している場合があります。しかし、大規模な企業組織になると、Tableau Server の大規模な管理を行うには最低でも管理者の役割が 3 つ必要となります。

Tableau Server 管理者

Tableau Server 管理者は、サイトの作成と編集、ユーザーの追加、役割の設定、Tableau Server のインストール完了後に発生する多くのコンテンツ関連タスクを処理するための管理ページにアクセスできます。また、Tableau Server 管理者は、サイト、ユーザー グループ、プロジェクトを管理する他のサーバー管理者やサイト管理者も作成し、管理します。

Tableau Server 管理者として Tableau Server へサインインするには、Tableau Server 管理者エリアへのサインインを参照してください。

TSM 管理者

Tableau サービス マネージャー (TSM) は、サーバー管理者に Tableau Server のインストール、アップグレード、構成、保守を行うためのコマンドラインと Web ベースのオプションを提供するツールです。TSM 管理者は、サーバーをインストールしたり、サーバー関連の管理タスク (サーバー データのバックアップ、バックアップの復元、ログ アーカイブの作成、マルチノード クラスタの管理など) を実行したりします。

TSM 管理者は、ローカル コンピュータの管理者である必要があります。Tableau サービス マネージャーの Web UI へのサインインを参照してください。

TSM 管理者が実行する一般的なタスクには、次のものがあります。

- インストール後の Tableau Server の初期構成
- 設定の編集やサーバー トポロジの変更を含む継続的な構成管理
- バックアップ、復元、ログの圧縮などの管理タスクの実行

TSM の詳細については、Tableau Services Manager の概要を参照してください。

Tableau ポータル管理者

Tableau Server の展開における重要な管理上の役割を担うのが、Tableau カスタマー ポータル管理者です。ポータル管理者は、Tableau の展開に必要なライセンスの発行や関連するキーの管理を行います。ポータル管理者としての最初のステップは、Tableau カスタマー ポータルでライセンスを購入することです。ライセンスを購入すると、ポータルから対応するプロダクト キーが返されます。ライセンスを更新するには、Tableau の更新 Web ページにアクセスしてください。

Tableau には、多数の製品 (Desktop、Server、Prep Builder など) があります。Tableau 製品を使用するには、Tableau カスタマー ポータルで購入・保存したプロダクト キーを使用して Tableau ソフトウェアを更新し、ライセンスをアクティブにする必要があります。Tableau のライセンス認証を担当する管理者は、ライセンスとキーの関係を理解することが重要です。ライセンス モデルとプロダクト キーを理解するを参照してください。

管理ツール

Tableau Server には、システムを管理するためのさまざまなツールセットが含まれています。

- **Tableau Server 管理者ページ**:各 Tableau Server インスタンスにインストールされている、Web ベースの管理サイトです。管理者ページで実行されるタスクは、サーバー管理者とサイト管理者の両方にとって日常タスクです。サーバー関連のタスクにはサイトやサイト管理者アカウントの作成、ユーザーのインポート(オプション)、ディレクトリサービスとの同期、抽出の更新スケジュールの設定、サーバーのパフォーマンスと利用状況の監視、およびその他のグローバル設定が含まれます。

  サイト関連のタスクにはコンテンツの管理やパーミッションの割り当て、抽出の更新の実行、グループやプロジェクトの作成、サイトアクティビティの監視、ユーザーの追加 (オプション)、およびその他のコンテンツ関連のタスクが含まれます。

  Tableau Web 環境の管理者エリアの移動を参照してください。

  Tableau Server 管理者ページに必要なパーミッションはサイトロールによって決まります。サイトロールは Tableau Server によって生成および管理されます。

- tsm コマンドライン リファレンス - サーバー全体の構成用プライマリインターフェイス。TSM の CLI を使用して作成された構成の多くは、最初の構成後に改訂されることはほとんどありません。例は次のとおりです。SSL、サブスクリプション、データのキャッシュ、サービス アカウント、SMTP アラート、ユーザー認証、シングルサインオン構成はすべて TSM の CLI を使用して実行されます。
- また、Tableau サービス マネージャーの Web UI へのサインインすることもできます。
- tabcmd: Windows コンピューターまたは Linux コンピューターで tabcmd コマンドライン ユーティリティを使用してスクリプトを作成し、Tableau Server サイトの管理タスクを自動で実行できます。たとえば、ユーザー、プロジェクト、およびグループの作成や削除には tabcmd を使用します。
- REST API:Tableau Server REST API では、HTTP を介してプログラムによって Tableau Server リソースを管理および変更できます。API は、データソース、プロジェクト、ワークブック、サイトユーザーまたは Tableau サーバー上のユーザーの陰にある機能へのシンプルなアクセスを提供します。このアクセスを使って、独自のカスタム アプリケーションを作成したり、Tableau Server リソースとの相互作用をスクリプトしたりできます。

セキュリティ

機密性が高いデータへ接続する可能性があるアプリケーション サーバーとして、Tableau Server ではさまざまな業界セキュリティ標準をサポートし、実装しています。サーバー管理者ドキュメントには、ユーザー認証、認可、データ セキュリティ、およびネットワーク セキュリティのベストプラクティスと

実装が含まれています。既定のインストールは設計で安全になっていますが、展開をさらにロックダウンするため、セキュリティ強化チェックリストに従うことをお勧めします。

セキュリティ監査コンプライアンス、脆弱性報告、およびその他のセキュリティリソースの詳細については、http://www.tableau.com/ja-jp/security にアクセスしてください。

Tableau Services Manager の概要

この記事では、Tableau Server の構成および管理に使用する Tableau Services Manager (TSM) の概要について説明します。

- 機能
- コンポーネント
- 認証
- 接続

機能

TSM には、Tableau Server を構成して維持するためのサーバー管理者コマンドラインおよび Web ベースのオプションがあります。これには、サーバーデータのバックアップ、バックアップの復元、ログアーカイブの作成、マルチノードクラスタの管理などの管理タスクを実行することが含まれます。たとえば、TSM を使用して次のタスクを実行できます。

- インストール後の Tableau Server の初期構成
- 設定の編集やサーバートポロジの変更を含む継続的な構成管理
- バックアップ、復元、ログの圧縮などの管理タスクの実行

以前のバージョンの Tableau Server に慣れている管理者にとって、TSM は以前のバージョンの Tableau Server にあった以下のツールに取って代わるものとなります。

- Tableau Server 構成ユーティリティ
- tabadmin コマンドラインユーティリティ
- Tableau Server Monitor

コンポーネント

TSM は*サービス*(このドキュメントでは*プロセス*と呼ばれる) と*クライアント*で構成されます。TSM プロセスは、Tableau Server プロセスを管理する管理サービスです。TSM を初期化した後は、他の Tableau Server がオフラインの場合でも TSM プロセスが継続的に実行されます。

Tableau Server が停止する場合でも実行する TSM プロセスには以下のものがあります。

- 管理エージェント
- 管理コントローラー
- クライアントファイル サービス
- 調整サービス (Apache Zookeeper™ に基づく)
- サービス マネージャー
- ライセンス発行サービス

TSM プロセスと Tableau Server プロセスの詳細については、Tableau Server プロセスを参照してください。

TSM 認証

TSM Web UI、コマンド ライン インターフェイス、または TSM API を使用する場合も、管理タスクを実行するには Tableau Server への認証が必要です。このユーザー アカウントは、Tableau Server 管理者やサイト管理者などの Tableau Server ユーザー アカウントとは区別されます。

TSM は動作しているオペレーティング システムにユーザーの認証を委任します。Linux の場合、認証はプラグ可能な認証モジュール (PAM) を使用して処理されます。PAM は、Tableau Server がサポートされているすべての Linux ディストリビューションの規格です。組織でディレクトリサービス (Active Directory、LDAP) の認証に PAM を構成している場合、TSM へのアクセスをそのディレクトリサービスからユーザーに許可できます。このシナリオでは、`tsmadmin` グループのメンバーである PAM の認証ユーザーは、TSM へのアクセスを許可されます。

注: TSM 管理者ユーザー名がドメインに対して認証されるとき、`sAMAccountName` 形式のみがサポートされます。Active Directory 認証を処理する PAM で Tableau Server on Linux を使用する場合、Tableau では「@」などの特殊文字を処理できないため、UPN (userPrincipalName) 形式 (メール アドレスや user@domain.com など) はサポートされません。

2019.1 リリースでは、TSM 認証プロセスは PAM を直接使用し、PAM にエラーが発生する場合やディレクトリサービスを使用して構成されていない場合には*代替ユーザー*(su) を使用して認証スキームを構成します。PAM でディレクトリサービスが構成されていない場合、ローカル アカウントを Linux コンピューターで管理する必要があります。こうした場合、TSM は su の認証方法を使用し、ユーザーが提供した認証資格情報を渡して /bin ディレクトリで true コマンドを実行します。このコマンドが成功すると、認証が検証されます。そのため、ユーザーが tsmadmin グループのメンバーの場合、認証されたユーザーは TSM へのアクセスを付与されます。

カスタム PAM サービスの定義

TSM は認証に標準的な PAM *ログイン*サービスを使用します。さらに、/etc/pam.d に tableau PAM サービス ファイルを作成することで、TSM 認証の動作をカスタマイズすることができます。このファイルが存在する場合、PAM ログイン サービスの代わりにファイルが確認されます。

TSM 認可グループ

Tableau Server コンピュータ上に存在するユーザーを TSM に対して認証します。この TSM ユーザー アカウントでは、パスワードベースの認証を使用する必要があります。既定では、TSM ユーザー アカウントは Tableau Server が実行されているコンピューターの tsmadmin グループのメンバーである必要があります。または、TSM 管理用に別の認可グループを指定することもできます。インストール プロセスで別の既定グループのを指定する場合は、initialize-tsm スクリプトの出力ヒントを参照してください。Tableau Server をインストールした後に別の認可グループを指定するには、*カスタム TSM 管理グループの構成*」を参照してください。

TSM クライアントの接続

セキュリティ対策として、HTTPS を経由するクライアント(CLI、Web UI、Rest API) を使用してのみ TSM に接続できます。セキュリティ対策として、HTTPS を使用しないとTSM には接続できません。

TSM クライアントを使用して接続している場合、TSM 管理コントローラー サービスを実行している Tableau Server インスタンスに接続する必要があります。

セキュリティのベストプラクティスとして、TSM ポート(既定では 8850) をインターネットに公開しないでください。

> **注**: TSM CLI ツールでは、一部のシナリオに管理者の認証資格情報は必要ありません。特に TSM 承認グループのメンバーとしてログインしているアカウントの場合、tsm CLI をローカルで実行する際にコマンドを実行するために認証資格情報を指定する必要はありません。詳細については、tsm CLI での認証を参照してください。

TSM HTTPS 接続は、Tableau Server インストーラーによって生成される自己署名証明書に依存します。この証明書は、Tableau が HTTP 経由のトラフィックの暗号化のために作成する SSL 証明書に署名する、Tableau インストール CA 証明書です。Tableau インストール CA 証明書は、TSM 管理コントローラーに接続するシステムが信頼する必要があります。

TSM CLI クライアントは、TSM Web UI が使用する以外のストアによる証明書の信頼を検証します。TSM CLI クライアントはローカル Java キーストアの信頼されているストアを参照し、CA 証明書の信頼を検証します。TSM Web UI は Web ブラウザーとの接続を確立する必要があるため、オペレーティング システムの信頼されるキーストアを使用して信頼が検証されます。CA 証明書の保存方法の差は、以下に概説する信頼の構成シナリオの違いによって決まります。

- Tableau Server での TSM CLI の通信では、証明書の信頼はインストール、ノード bootstrap、アップグレードプロセスの一部として既定で構成されます。Tableau インストール CA 証明書は、Java キーストアで信頼されるストアに追加されます。これにより、追加構成を行わなくても、クラスタの任意のコンピューターより TSM に CLI を使用してアクセスできます。しかし、TSM Web UI にアクセスする際には、ブラウザーにより TSM 管理コントローラー サービスを実行中のホストを信頼するようにメッセージが表示されます。
- リモート コンピューターからの TSM CLI 接続では、TSM 管理コントローラーを実行中の Tableau Server に初めて接続するときに、Tableau インストール CA 証明書を信頼するようにメッセージが表示されます。CA 証明書を信頼するよう選択できますが、この場合には、証明書の有効期限が切れるまで (既定は 4 年間)、そのコンピューターにメッセージは表示されません。または、`--trust-admin-controller-cert` フラグとともに TSM コマンドを実行することで、ワンタイムの信頼を使用して接続できます。
- リモート コンピューターからの TSM Web UI 接続では、ブラウザーにより TSM 管理コントローラー サービスを実行中のホストを信頼するようにメッセージが表示されます。

インフラストラクチャ計画

Tableau Server をインストールする前に、ディスク要件、推奨される構成、ユーザー アカウント、セキュリティ、およびネットワーク要件を確認する必要があります。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

インストールの前に...

注: Tableau Web サイトの Tableau Server の技術仕様に関するその他の情報は、こちらで見つけることができます。

このトピックでは、Tableau Server を稼働環境にインストールする前に考慮する必要のある要件および推奨事項について説明します。

- 初心者が組織に Tableau Server を展開する場合には、最初にテスト環境の単一サーバーとして Tableau Server を展開することをお勧めします。単一サーバーを最も簡単にインストールするには、「インストールのジャンプスタート」のステップに従います。
- エンタープライズ対応の 4 ノードのリファレンス アーキテクチャを階層型データ センターに導入する方法を説明するエンドツーエンドの手順については、「企業環境への Tableau Server 導入ガイド」を参照してください。
- 分散クラスターに Tableau Server を展開している場合は、このトピックに記載された要件および推奨事項に加えて分散環境の要件も確認してください。

- Tableau Server on Windows から Tableau Server on Linux に移行しようとしている場合には、Tableau Server の Windows から Linux への移行を参照してください。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

本番環境インストールに関するハードウェアの推奨事項

次のリストは、本番環境で使用するために Tableau Server を単一ノードにインストールする場合の最小ハードウェア推奨事項です。

重要: これらの推奨事項は最低限のものであり、ご利用のインストールおよび組織の要件を反映していない場合があります。たとえば、抽出やフローをパブリッシュするかどうか、Tableau Server にパブリッシュするワークブックの数など、ディスク領域の要件に影響を及ぼす要因がいくつかあります。空きディスク領域の要件に影響を与える可能性のあるものの詳細については、ディスク容量の要件を参照してください。

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
シングルノード	• 64 ビット (x86_64 チップセット) • SSE4.2 および POPCNT 命令セットをサポートする必要があります	8 コア (16vCPU)、2.0 GHz 以上	バージョン 2022.3 以降: • 128 GB	50 GB

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
	• ARM ベースのプロセッサーはサポートされていません		バージョン 2021.4.0 からバージョン 2022.1.x: • 64 GB バージョン 2021.3.x 以前: • 32 GB	
	Tableau Server インストールに Tableau Prep Conductor を追加する場合、2 番目のノードを追加し、これを Tableau Server Prep Conductor の実行専用にすることをお勧めします。このノードには最低 4 コア (8 vCPU)、RAM 16 GB が必要です。			
	推奨されるスワップ メモリ: システム RAM の平方根または 8 GB のいずれか大きい方。			
マルチノードおよびエンタープライズ展開	技術的ガイダンスについては Tableau にお問い合わせください。 ノードは最小ハードウェア推奨事項を満たしているか、それを超える必要があります。ただし、次の場合は除きます。 • バックグラウンダーのインスタンスを最大 2 つ実行している専用バックグラウンダー ノード: 4 コアは許容可能。 • Tableau Prep Conductor の専用ノード: 最低 4 コア (8 vCPU)、RAM 16 GB。 • 独立したゲートウェイの専用ノード: 最低 2 コア (4 vCPU)、8 GB の RAM、および 100 GB の空きディスク領域。			

重要:ディスク領域の要件は TSM を初期化するまでは確認できません。十分な領域がない場合、Tableau Server パッケージのインストールが終了するまでその旨を通知されません。

50 GB のディスク空き領域。少なくとも 15GB が `/opt` ディレクトリに割り当てられ、残りがデータストレージ用として `/var` ディレクトリに割り当てられます。

- 空きディスク容量は、Tableau Server セットアッププログラムの解凍後に計算されます。セットアッププログラムでは約 1 GB の容量を使用します。抽出を使用するかどうかなどさまざまな要因に応じて、追加ディスク領域の割り当てが必要になる場合があります。

  Tableau Server の中核部分は、15 GB 以上の空きディスク領域があるディレクトリにインストールする必要があります。十分なスペースがないコンピューターに Tableau Server をインストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行することができません。既定のインストール場所は `/opt` ディレクトリです。RHEL ディストリビューションで Tableau Server のインストール パスを変更することができます。

  抽出を頻繁に使用する予定がある場合は、追加ディスク領域の割り当てが必要になる場合があります。インストール時に、データ(抽出)を保存するための別のディレクトリを指定することができます。

- **外部ファイル ストアに対するネットワーク接続ストレージ領域の要件:** 外部ファイル ストアを使用した Tableau Server を構成する場合は、ネットワーク接続ストレージ専用のストレージ領域の容量を見積もる必要があります。

  ストレージ サイズの見積もり: パブリッシュと抽出更新に必要なストレージの容量を考慮する必要があります。さらに、トピックオプション 2: リポジトリを個別にバックアップするで説明されているように、リポジトリのバックアップを個別に実行するオプションを特に選択しない限り、リポジトリのバックアップ サイズも考慮する必要があります。

  - 抽出:
    - Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。

    - 抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。
  - リポジトリのバックアップ:
    - リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。
    - リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、`workgroup.pg_dump` ファイルのサイズを確認します。
- コア数は "物理" コアに基づいています。物理コアは実際のサーバー ハードウェアまたは仮想マシン (VM) 上のコアを表す場合があります。コア数のカウントの目的でのハイパー スレッドは無視されます。
- 表示されている RAM はシングル ノード インストールでの最小限の推奨です。たとえば、アクティビティ、ユーザー数、バックグラウンド ジョブなどに応じて、RAM が増えるとインストールの機能が向上する場合があります。

推奨事項の全リストと最小要件を確認するには、Tableau Server の最小ハードウェア要件と推奨事項を参照してください。Tableau がスケーラビリティのテストのために内部で使用するハードウェア仕様については、「本番環境インストールに関するハードウェアの推奨事項」を参照してください。

アマゾン ウェブ サービスおよび Google Cloud Platform でのパブリック クラウド展開の場合、その "vCPU" は完全な CPU コアではなく、実際には CPU ハイパースレッドである場合があります。クラウドのインスタンスのサイズを変更する場合、Tableau Server の CPU コア要件として指定されるサイズの 2 倍の vCPU が必要になります (トライアルの最小インストールで 8 vCPU、単一ノードの推奨インストールでは 16 vCPU)。

オペレーティング システム要件

重要: Linux で Tableau Server バージョン 2023.3.1 - 2023.3.17、2024.2.1 - 2024.2.10、および 2025.1.0 - 2025.1.1 を実行している場合に、初期化またはライセンスの有効化が失敗

> する既知の問題があります。この問題は、2025 年 5 月のメンテナンス リリース以降、バージョン 2023.3.18、2024.2.11、2025.1.2 で修正されています。これを解決するには、このナレッジベースの記事を参照してください。

次の Linux ディストリビューションがサポートされています。

	2022.3.-0 以前 2022.3.-3	2022.3.-4 以降	2023.1.-0 以前 2023.1.-7	2023.1.-8 以降	2023.3.-0	2023.3.1 - 2024.2.-11	2024.2.-12 以降	2025.1.-x
AlmaLinux 8.x						✓	✓	✓
AlmaLinux 9.x						✓	✓	✓
Amazon Linux 2	✓	✓	✓	✓	✓	✓	✓	
Amazon Linux 2023						✓	✓	✓
CentOS 7.9 以降 (8.x を除く)	✓	✓	✓	✓	✓	✓	✓	
CentOS Stream 8.x						✓	✓	✓
CentOS Stream 9.x						✓	✓	✓
Oracle	✓	✓	✓	✓	✓	✓	✓	

	2022.3.-0 以前 2022.3.-3	2022.3.-4 以降	2023.1.-0 以前 2023.1.-7	2023.1.-8 以降	2023.3.-0	2023.3.1 - 2024.2.-11	2024.2.-12 以降	2025.1.-x
Linux 7.3 以降								
Oracle Linux 8.x						✓	✓	✓
Oracle Linux 9.x						✓	✓	✓
RHEL 7.3 以降	✓	✓	✓	✓	✓	✓	✓	
RHEL 8.3 以降	✓	✓	✓	✓	✓	✓	✓	✓
RHEL 9.x					✓	✓	✓	✓
Rocky Linux 8.x						✓	✓	✓
Rocky Linux 9.x						✓	✓	✓
Ubuntu 16.04 LTS	✓	✓	✓	✓	✓	✓	✓	
Ubuntu 18.04 LTS	✓	✓	✓	✓	✓	✓	✓	
Ubuntu 20.04 LTS		✓	✓	✓	✓	✓	✓	✓

	2022.3.-0 以前 2022.3.-3	2022.3.-4 以降	2023.1.-0 以前 2023.1.-7	2023.1.-8 以降	2023.3.-0	2023.3.1 - 2024.2.-11	2024.2.-12 以降	2025.1.-x
Ubuntu 22.04 LTS				✓		✓	✓	✓
Ubuntu 24.04 LTS							✓	✓

Tableau Server in a Container の Linux ディストリビューションの要件については、「構築でサポートされているディストリビューション」を参照してください。

重要: 最新のセキュリティおよび機能パッチを確実に入手できるように、サポートされている最新バージョンの Linux ディストリビューションを使用して Tableau を展開することを強くお勧めします。Tableau は通常、サポートされているディストリビューションのメジャーバージョンの最新マイナーバージョンでテストおよび検証します。

Linux ディストリビューションに関するその他の注意事項:

- Red Hat Enterprise Linux (RHEL)、CentOS、Oracle Linux、Amazon Linux ディストリビューションは、このドキュメントでは総称して RHEL ライクと呼ばれます。
- CentOS 7、RHEL 7、Oracle Linux 7、Ubuntu 16.04 LTS、Ubuntu 18.04 LTS、Amazon Linux 2 の各ディストリビューションは、バージョン 2025.1.0 以降サポートされなくなりました。
- 2022 年 7 月の時点で、Debian ディストリビューションはサポートされなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください。
- Ubuntu の LTS 以外のリリースはサポートされていません。
- Tableau Server バージョン 2025.1.0 で Ubuntu バージョン 24.04 のサポートが追加されました。

- Ubuntu バージョン 20.04 のサポートは、Tableau Server バージョン 2023.1.0 および Server メンテナンス リリース 2022.1.12 および 2022.3.4 で追加されました。それ以前のバージョンではサポートされていません。

- カスタム カーネルはサポートされていません。

Tableau Server のマルチノード インストールでは、Tableau をインストールしているすべてのコンピューター ノードが Linux と同じディストリビューションの Linux を稼働している必要があります。

インストール ディレクトリ

Tableau Server の中核部分は、既定では `/opt` ディレクトリにインストールされます。

- Tableau Server のインストール先ディレクトリには、それに割り当てられている最低 15 GB の空きディスク領域が必要です。十分なスペースがないコンピューターに Tableau Server をインストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行することができません。
- RHEL ライクシステムでは規定以外のインストール ディレクトリを指定できますが、Ubuntu では場所を変更できません。
- RHEL ライクシステム上で規定以外のインストール場所を指定する場合は、シンボリック リンクまたはネットワーク ファイル システム (NFS) ボリューム上のディレクトリ位置を指定しないでください。

データディレクトリ

Tableau Server は既定で、Tableau が管理するすべてのコンテンツおよび抽出用のデータディレクトリを作成します。ディレクトリは `/var/opt/tableau/tableau_server` に作成されます。

インストール時にデータ(抽出) を保存するための別のディレクトリを指定することができます。別のディレクトリを使用する計画がある場合は、ディレクトリを作成しないでください。代わりに、Tableau Server のセットアップによりディレクトリを作成してください。データディレクトリには、インストール プロセスの際に設定される特定のパーミッションが必要です。

データディレクトリを変更するには、`initialize-tsm` スクリプトを実行するときに、データディレクトリパスと共にフラグを渡す必要があります。initialize-tsm スクリプトの出力ヒントを参照してください。

既定のデータディレクトリを変更する場合:

- ネットワークファイル システム (NFS) ボリューム上でシンボリック リンクやデータディレクトリの場所を指定することはしないでください。
- パスにピリオドやスペースが含まれているデータディレクトリの場所を指定することはしないでください。パスにピリオドやスペースが含まれていると、初期化が失敗します。
- データディレクトリは、インストール ディレクトリとは別のディレクトリにインストールする必要があります。

重要: `initialize-tsm` を実行した後でデータディレクトリの場所を変更することはできません。データディレクトリの場所は、以降のアップグレードを含め、展開の寿命が終わるまで保持されます。

Tableau Prep Conductor

Tableau Prep Conductor は Tableau Server でのプロセスの 1 つです。フローの実行、接続の認証資格情報の確認、フローが失敗した場合のアラート送信を行います。Tableau Prep Conductor は Tableau Server のスケジュール調整機能および追跡機能を活用しているため、データが変更になる際に、Tableau Prep Builder にログインして個々のフローを手動で実行するのではなく、フロー実行を自動化してフロー出力を更新することができます。

Tableau Prep Conductor は個別にライセンスされ、データ管理 ライセンスを通じて利用できます。Tableau Prep Conductor のライセンス発行の詳細については、データ管理 のライセンスを参照してください。

専用ノードで Tableau Prep Conductor を有効にすることをお勧めします。詳細情報:

- 新しい Tableau Server をインストールする場合は、「手順 1 (新規インストール): Tableau Server を Tableau Prep Conductor と一緒にインストール」を参照してください。
- Tableau Prep Conductor を既存の Tableau Server インストールに追加する場合は、「手順 1 (既存インストール): Tableau Prep Conductor の有効化」を参照してください。

その他の要件

ご使用の環境が次の追加の要件も満たしていることを確認してください。

ホスト名

- Tableau Server は、ドメイン名サーバー (DNS) または Tableau Server を実行しているコンピューター上のローカル ホスト ファイルを使用し、IP アドレスに対するホスト名を解決できる必要があります。既定では、ホスト ファイルは `/etc/hosts` に保存されています。
- セットアップ プロセスの際に Tableau サービス マネージャーを起動した後はサーバーのホスト名を変更しないでください。たとえば、cloud-init パッケージを使用して仮想マシンを初期化し、その仮想マシンに Tableau Server をインストールする場合にこのようになる可能性があります。
- アンダースコア (_) を含むホスト名は、Tableau Server でサポートしていません。

静的 IP アドレス

Tableau Server をインストールするコンピューターには静的な IPv4 または IPv6 アドレスが必要です。

データベース ドライバー

特定のデータ ソースに接続するには、Tableau Server をインストールするコンピューターに正しいデータベース ドライバーがインストールされている必要があります。詳細については、データベース ドライバーを参照してください。

利用可能なポート

TSM と Tableau Server はいずれもアクセスするために利用可能な TCP ポートが必要です。TSM は 8850 番ポートが既定で、Tableau Server ゲートウェイ サービスは 80 番ポートが既定です。Tableau Server をインストールする前に、8850 番ポートと 80 番ポートをお使いのシステムで使用しないことを強くお勧めします。これらのポートが利用できないと、TSM とゲートウェイのポートは別のポート番号に動的に再マッピングされる場合がありますが、どのポートに再マッピングされたかを表示するインターフェイスは現時点で存在しません。

Tableau サービス マネージャーのポートを参照してください。

ローカル ファイアウォール構成

Tableau Server をインストールするコンピューターでファイアウォールを実行している場合は、Tableau Server トラフィック用に次の既定のポートを開く必要があります。443 を除くすべてのポート番号を変更できます。

ポート	TCP/UDP	使用...	インストールのタイプ	
			すべて	分散/高可用性
80	TCP	ゲートウェイ	X	
443	TCP	SSL。Tableau Server で SSL を構成している場合、アプリケーション サーバーはこのポートに要求をリダイレクトします。このポートは変更しないでください。	X	
8850	TCP	Tableau Services Manager。	X	
8060	TCP	PostgreSQL データベース。	X	
8061	TCP	PostgreSQL バックアップ検証ポート	X	
8000-9000	TCP	Tableau プロセスの動的マッピング用に既定で予約されているポートの範囲		X
27000-27009	TCP	Tableau Server がライセンス サービスで使用するポートの範囲。この範囲は、ライセンス サービスを実行しているノードで開いている必要があり、また他のノードからアクセス可能である必要があります。既定では、最初のノードはライセンス サービスを実行します。	X	

Tableau サービス マネージャーのポートおよびローカル ファイアウォールの構成を参照してください。

システム ユーザーとグループ

Tableau Server on Linux では、1 人の権限なしユーザーと2 つのグループを使用して適切な操作を行います。Tableau では設定中に既定のアカウントおよびグループが作成されます。または、既存のアカウントを指定することもできます。システム ユーザーとグループ
およびTSM 認可グループを参照してください。

sudo および root アクセス

Tableau Server のすべてのインストール タスクおよび管理タスクは root として実行する必要があります。多くの場合は sudo コマンドを使用して実現できますが、root ユーザーとしてコマンドを直接実行することも可能です。

root アカウントを使用して Tableau Server をインストールするには、インストール時にユーザー アカウントを指定する必要があります。アカウントは TSM の管理に使用されます。`-a` オプションを使用して initialize-tsm スクリプトを実行することによりアカウントを指定します。initialize-tsm スクリプトの出力ヒントを参照してください。

アカウントのパスワード

Tableau Server をインストールして管理するユーザー アカウントは、パスワードで認証できるアカウントである必要があります。これは、このユーザーが認証に別の方法 (パブリック キー認証など) を使用してはならないということを意味します。

Tableau Server のインストールと初期化に使用するアカウントにパスワードがない場合は、次のように `passwd` コマンドで設定できます。

```
sudo passwd $USER
```

ポート アクセス要件

SSH などを使用して Tableau Server をリモートからインストールする場合は、次のポートが開かれていることを確認します。番 ポート。

- `8850`。Tableau サービス マネージャー (TSM) の Web インターフェイスに使用するポートです。このインターフェイスを使用して Tableau Server を構成できます。
- `80`。番 ポート。主要な Tableau Server Web インターフェイスに使用するポートです。

Tableau Server インストーラーはインストール プロセス中にこれらのポートを開こうとしますが、firewalld ファイアウォールに対してこれらのポートのみを開くことができます。コンピューターが別のファイアウォールを実行している場合は、インストール前にポートを開く必要があります。

仮想コンテナー環境

バージョン 2021.2 以降、Tableau Server on Linux の特定の構成をコンテナーで実行できるようになりました。サポートされる構成の詳細については、Tableau Server in a Containerを参照してください。

Tableau Server on Linux の以前のバージョンとサポートされていない構成はテストされておらず、Docker などの仮想コンテナー環境ではサポートされていません。このような環境で Tableau Server on Linux をインストールしても、期待どおりには機能しません。

パッケージ要件

Systemd

Tableau Server ではサービスを管理する systemdが必要です。このパッケージは既定で CentOS 7 以降および Ubuntu 16 以降にインストールされます。これらの配布の修正バージョンで Tableau Server をテストすると決定した場合は、次のコマンドを実行して systemd がインストールされていることを確認できます。

```
whereis systemd
```

systemd がインストールされていると、インストールの場所が表示されます。たとえば、次のような出力が表示されます。

```
systemd: /usr/lib/systemd /etc/systemd
/usr/share/systemd /usr/share/man/man1/systemd.1.gz
```

systemd をインストールしているものの、Tableau インストーラーで systemd の要件確認に失敗する場合には、systemd が実行中でない可能性があります。systemd が実行していることを確認するには、次のコマンドを実行します。

```
ls /run/systemd
```

ファイルとディレクトリの一覧が出力されます。systemd が実行中の場合には、出力に `system` が含まれます。`system` が出力に含まれていない場合には systemd は実行していません。

ウイルス対策ソフトウェア

Tableau Server で使用されるディレクトリをスキャンするウイルス対策ソフトウェアは、Tableau Server のインストールや現在の使用に干渉する可能性があります。場合によっては、これによってインストール エラーや Tableau Server の起動の問題が発生したり、パフォーマンスへの影響が生じたりすることがあります。Tableau Server を実行しているコンピュータでウイルス対策ソフトウェアを実行する予定がある場合は、ナレッジ ベースの推奨事項に従ってください。

次の手順、Tableau Server のインストールと構成に進んでください。

ディスク容量の要件

一般的に、Tableau Server で日常的な使用量を割り当てるための追加ディスク容量を推定するときには、抽出を Tableau Server にパブリッシュするかどうか、および Tableau Server へのパブリッシュを想定しているワークブック数を考慮に入れる必要があります。抽出の使用を想定している場合、Tableau では数百ギガバイト(GB) を開始容量とすることを推奨しています。抽出の使用を想定していない場合、使用量ニーズを満たすのに 50 GB 程度しか必要ないと考えられます。ドライブの空き領域アラートを設定する方法については、「サーバー イベント通知の構成」を参照してください。

> Tableau Server on Windows については、「ディスク容量の要件」を参照してください。

Tableau Server では十分なディスク空き容量を確保することが重要です。Tableau Server のインストール中にノードのディスク空き容量が不足すると、Tableau Server または TSM Web UI にアクセスできないなど、パフォーマンスが不安定になる可能性があります。トラブルシューティングの手順については、を参照してください。

ディスク空き容量の要件に影響を与える要因と、Tableau Server のインストール先として選択できる場所を以下に示します。

Tableau Server への抽出のパブリッシュ

Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。

抽出の更新

抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。

多数のワークブックの作成

ワークブックを使用している場合、Tableau Server にパブリッシュするワークブックの数を考慮に入れます。個々のワークブックが専有するディスク容量は少量の傾向があります。とはいえ、数千ものワークブックのパブリッシュを想定している場合には、これらのワークブックに対応できるよう追加のディスク容量を割り当てる必要があるかもしれません。

ロギング

日々の管理とトラブルシューティングをサポートするため、Tableau Server は通常動作の一環としてログ ファイルを作成しています。ロギングの構成レベルによって異なりますが、Tableau Server コンピューターで必要なディスク容量に大きく影響する可能性があります。

バックアップと復元のプロセス

バックアップの作成に必要な空きディスク領域は、Tableau Server リポジトリやファイル ストア サービス内のデータの量、tabadmincontroller サービスとの併置によって異なります。バックアップ中は、古い抽出をクリーンアップするためのバックグラウンド タスクが一時的に一時停止されます。つまり、バックアップの期間中は抽出を更新しても抽出ファイルがそのまま残るため、ディスク領域の使用率が増えます。バックアップに長い時間がかかったり、組織で定期的に更新される抽出が多数使用されていたりする場合は、一時ディスク領域の使用率が大幅に増加します。これらの一時ファイルは、バックアップが完了すると削除されます。

次の表に、ノードがリポジトリ、ファイル ストア、コントローラー、またはそれらを組み合わせてホストするかどうかに基づいた、バックアップに必要なディスク領域を示します。マルチノードの Tableau Server 環境では、各ノードで必要なディスク空き容量を見積もる必要があります。

リポジトリ	ファイル ストア	コントローラー	必要なディスク領域
✓			リポジトリデータ x 3 + 250 MB リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。 リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、workgroup.pg_dump ファイルのサイズを使用します。
	✓		ファイル ストア データ x 1.5 ファイル ストア データ(抽出、フローなど)の推量を取得するには、`<data directory>/dataengine` ディレクトリのサイズを確認します。
		✓	リポジトリデータ x 3 + 250 MB + ファイル ストア データ x 2.5
✓	✓		リポジトリデータ x 3 + 250 MB + ファイル ストア データ x 1.5
	✓	✓	リポジトリデータ x 3 + 250 MB + ファイル ストア データ x 1.5
✓		✓	リポジトリデータ x 3 + 250 MB + ファイル ストア データ x 2.5

✓	✓	✓	リポジトリデータx 3 + 250 MB + ファイル ストアデータx 1.5

復元のディスク空き容量の要件

データベースの復元プロセスを正常に実行するには、十分なディスク空き容量が必要です。

Tableau Server を復元するには:

- コントローラー ノードでは、バックアップ アーカイブのサイズ以上の空き領域が必要です。
- リポジトリ ノードでは、バックアップ アーカイブ内にリポジトリデータのサイズの少なくとも 3 倍 + 250 MB + pgsql データディレクトリに相当する空き領域が必要です。
- ファイル ストア ノードでは、バックアップ アーカイブ内にある dataengine フォルダーのサイズの少なくとも 2 倍の空き領域が必要です。

サイトのエクスポートとインポート

サイトのエクスポートとインポートには、バックアップと復元のプロセスと同様に、十分なディスク空き容量が必要です。少なくとも、エクスポートされるファイル用に十分な空き容量が必要です。サイトのエクスポートとインポートを行うときに必要なディスク容量のガイドラインとして、バックアップと復元の要件を使用できます。

このファイルは TSM の `basefilepath.site_export.exports` 変数に定義されているディレクトリに生成されます。詳細については、「tsm sites export」を参照してください。

ファイル パスとその変更方法の詳細については、「tsm ファイル パス」を参照してください。

推奨されるベースライン構成

Tableau Server 展開のトポロジ (ノード数、Tableau Server プロセス数) を決定する際には、利用環境、データのソース、セルフサービス データ アクセスを提供するための管理、作業負荷、および使用率などの変数について考慮する必要があります。しかし、Tableau Server を初めて展開する

場合、これらの変数について十分な情報がないことがあります。このトピックでは、Tableau Server インストールの出発点として使用できる3つのベースライン アーキテクチャについて説明します。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

本番環境インストールに関するハードウェアの推奨事項

以下の本番環境の Tableau Server インストールに関するハードウェア推奨事項は、Tableau チームが Tableau Server のスケーラビリティのテストに使用するハードウェアに基づいています。これらの推奨事項を、本番環境展開の出発点として使用することをお勧めします。概念実証 (PoC) の展開では、Tableau Cloud を使用することをお勧めします。詳細については、「インストールの最小ハードウェア要件」を参照してください。

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
シングルノード	• 64 ビット (x86_64 チップセット) • SSE4.2 および POPCNT 命令セットをサポートする必要があります • ARM ベースのプロセッ	8 コア (16vCPU)、2.0 GHz 以上	バージョン 2022.3 以降: • 128 GB バージョン 2021.4.0 から バージョン 2022.1.x: • 64 GB	500 GB ~ 1 TB

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
	サーはサポートされていません		バージョン 2021.3.x 以前: • 32 GB	
	Tableau Server インストールに Tableau Prep Conductor を追加する場合、2 番目のノードを追加し、これを Tableau Server Prep Conductor の実行専用にすることをお勧めします。このノードには最低 4 コア (8 vCPU)、RAM 16 GB が必要です。			
	推奨されるスワップ メモリ: システム RAM の平方根または 8 GB のいずれか大きい方。			
マルチノードおよびエンタープライズ展開	技術的ガイダンスについては Tableau にお問い合わせください。 ノードは最小ハードウェア推奨事項を満たしているか、それを超える必要があります。ただし、次の場合は除きます。 • バックグラウンダーのインスタンスを最大 2 つ実行している専用バックグラウンダー ノード: 4 コアは許容可能。 • Tableau Prep Conductor の専用 ノード: 最低 4 コア (8 vCPU)、RAM 16 GB。 • 独立したゲートウェイの専用 ノード: 最低 2 コア (4 vCPU)、8 GB の RAM、および 100 GB の空きディスク領域。			

注: 仮想マシンを使用する展開の場合、専用 CPU アフィニティを適用することをお勧めします。仮想環境で Tableau Server を実行している場合は、VM ホスト上の物理 CPU コア数と関連して、vCPU 割り当てに対して VM ホストのベストプラクティスを使用します。通常、Tableau Server の場合は 2 vCPU = 1 物理コアです。たとえば、AWS インストールでは、4 コアの最小推奨事項が 8 AWS vCPU に相当します。同様に、Tableau Server が適切な計

> 算、メモリ、およびデータ リソースに確実にアクセスできるように、使用する仮想インフラストラクチャのプロバイダーから提供されるベストプラクティスに従います。仮想環境またはクラウドベース展開で Tableau Server をインストールしている場合は、このトピックで後述する仮想マシンおよびパブリック クラウド展開のセクションを参照してください。

ディスク領域の推定

抽出をパブリッシュするかどうか、フロー、または Tableau Server にパブリッシュするワークブックの数が、ディスク領域要件に影響を及ぼす要因としてあげられます。詳細については、ディスク容量の要件を参照してください。

ベースライン構成

- 単一サーバー インストール
- 2 ノード インストール - 抽出負荷の高い環境用
- 2 ノード インストール - フロー環境用
- 高可用性インストール (HA)

単一サーバー インストール

推奨事項

初めての展開では、単一のマシンを使用して Tableau Server をインストールし、基幹業務以外で限定的に使用することをお勧めします。単一サーバー インストールは、作業負荷が増大したときに、複数 ノード インストールに展開することもできます。

単一サーバー インストールが適さない可能性がある例を以下に示します。

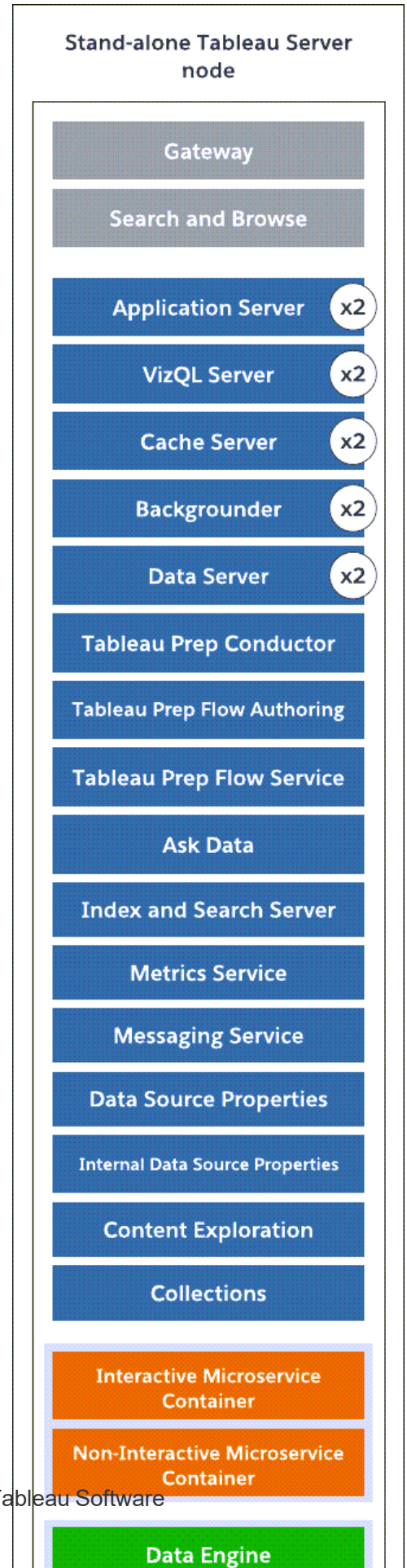
- 使用するシステムが必要不可欠なシステムであり、高い可用性が必要とされている場合。高可用性とは、システム ダウンタイムを最小限に抑えることです。それには、単一障害点をなくし、信頼性の高いフェールオーバー メカニズムを備える必要があります。Tableau Server は、冗長性の提供と単一障害点の排除のために、少なくとも 3 ノードの構成が必

要です。このことが、複数ノード構成に移行する主な理由の1つです。

- 多数のアクティブユーザーがいて、多数の抽出更新が発生する場合は、マシン上の同一リソースに対してこの2種類の負荷が競合する可能性があります。そのようなシナリオにおいては、異なる作業負荷を分離するために追加の専用ノードが必要になることがあるため、単一サーバー構成が適切な選択肢ではない可能性があります。

注: アクティブユーザーとは、Tableau Server に対して行われるインタラクティブな同時要求のことを表します。ノートパソコンやモバイルデバイスでのダッシュボードの使用、Web 作成、パブリッシュされたデータソースへの接続およびクエリなどが含まれます。

サーバー構成


Stand-alone Tableau Server node
Gateway
Search and Browse
Application Server x2
VizQL Server x2
Cache Server x2
Backgrounder x2
Data Server x2
Tableau Prep Conductor
Tableau Prep Flow Authoring
Tableau Prep Flow Service
Ask Data
Index and Search Server
Metrics Service
Messaging Service
Data Source Properties
Internal Data Source Properties
Content Exploration
Collections
Interactive Microservice Container
Non-Interactive Microservice Container
Data Engine

- すべてのプロセスが 1 台のマシン上にインストールされたスタンドアロンの単一サーバー ノード。
- 以下は、8 コア マシン用のプロセス数です。
    - VizQL Server: 2 インスタンスに設定します (物理コア数を 4 で割る、最大 4)。
    - バックグラウンダー、キャッシュ サーバー、およびデータ サーバー: 2 インスタンスに設定します。
    - その他のすべてのプロセスは、ハードウェアに関係なく、プロセスのインスタンスを 1 つだけインストールします。

注: サーバー上で Data Management プロダクト キーがアクティブ化されている場合、Tableau Prep Conductor の 1 つのインスタンスが自動的に構成されます。ただし、Tableau Prep Conductor の専用ノードを設けることをお勧めします。Tableau Server にフローを設定する予定がある場合は、2 つ以上のノードを使用し、そのノードのうち 1 つをフローのみの実行専用にすることをお勧めします。単一ノード サーバーであるため、上記の構成例には Tableau Prep Conductor は含まれていません。

マルチノード インストール

複数のマシン上で Tableau Server を実行することを、マルチノード インストールまたはクラスターと言います。さまざまな理由でマルチノード インストールが必要になることがあります。たとえば、抽出負荷の高い環境では、一部のハードウェア リソースがバックグラウンダー プロセス専用に使用される可能性があります。高い可用性を必要とするシステムの場合、少なくとも 3 つのノードを備えたマルチノード環境を構築する必要があります。

2 ノード インストール - 抽出負荷の高い環境用

推奨事項

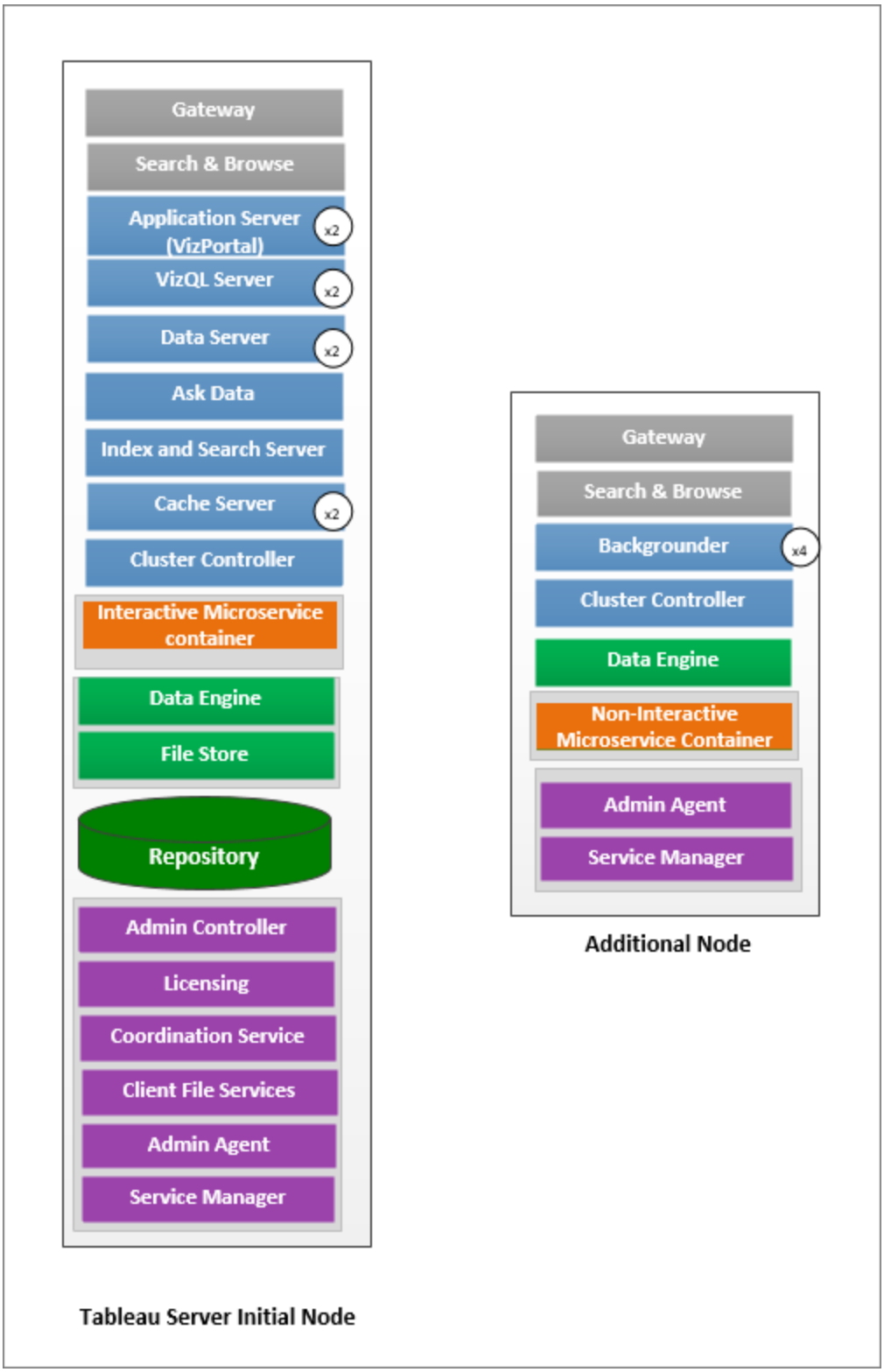
以下の条件が当てはまる場合は、2 ノード構成から開始してください。

- **抽出負荷の高い環境:** データソースの大半を抽出が占めます。少数の極めて大きな抽出を持っているだけで、非常に多数の小さな抽出を持っているのと同じように、展開がこのカテゴリに該当する場合があります。

- **頻繁な抽出の更新:** 抽出を更新するタスクは、CPU を大量に使用するタスクです。抽出が頻繁に更新される(1 日の営業時間内に数回など) 展開は、更新タスクを処理するバックグラウンドプロセスをさらに強化することで改善されることがよくあります。

重要: 2 ノード構成は、高可用性のための最小要件を満たしません。可用性の高いシステムが必要な場合は、高可用性インストール (HA)を参照してください。

サーバー構成


Gateway
Search & Browse
Application Server (VizPortal)
x2
VizQL Server
x2
Data Server
x2
Ask Data
Index and Search Server
Cache Server
x2
Cluster Controller
Interactive Microservice container
Data Engine
File Store
Repository
Admin Controller
Licensing
Coordination Service
Client File Services
Admin Agent
Service Manager
Tableau Server Initial Node
Gateway
Search & Browse
Backgrounder
x4
Cluster Controller
Data Engine
Non-Interactive Microservice Container
Admin Agent
Service Manager
Additional Node

- 最初のノードで、バックグラウンダーを除くすべてのプロセスをインストールします。以下は、8 コア マシン用のプロセスのインスタンス数です。

    - VizQL Server: 2 インスタンスに設定します(既定の計算: 物理コア数を 4 で割る、最大 4)。

    - キャッシュサーバーおよびデータサーバー: 2 インスタンスに設定します。「データに聞く」(Ask Data) の 1 つのインスタンスが、Data Server があるノードで自動的に構成されます。

    - Index and Search Server: `indexandsearchserver.vmopts` TSM 構成オプションを使用して、パフォーマンスが向上するように Index and Search Server のメモリを構成できます。詳細については、tsm configuration set のオプションを参照してください。

    - その他のすべてのプロセスは、ハードウェアに関係なく、プロセスのインスタンスを 1 つだけインストールします。インタラクティブなマイクロサービス コンテナーの 1 つのインスタンスがアプリケーション サーバーが有効になっているノードにインストールされ、インタラクティブでないマイクロサービス コンテナーの 1 つのインスタンスがバックグラウンダーが有効になっているノードにインストールされます。

- 追加ノードでバックグラウンダーを隔離します。このノードで実行するバックグラウンダープロセスの最小数を計算するには、コンピューターの物理コアの合計数を 4 で除算します。最大数を計算する場合は、コンピューターの物理コアの合計数を 2 で除算します。上記の例では、どちらのノードも 8 物理コアを備えたマシン上にあります。バックグラウンダーをインストールするとき、Tableau Server によりデータエンジンの 1 つのインスタンスが自動的にインストールされます。

注: この構成では、Tableau Server で Tableau Prep Conductor が有効になっていないことを前提としています。Tableau Prep Conductor を使用してフローのスケジュールと管理を行っており、抽出負荷の高い環境の場合、または負荷の高いフローに対応する必要がある場合、最低 3 つのノードがあり、このトピックで後述する 3 ノード構成を使用することをお勧めします。

パフォーマンスと利用状況に関するデータを監視および収集しながら、これらのプロセスのインスタンス数を微調整および構成できます。たとえば、バックグラウンダーを実行する専用のノードで、最初はバックグラウンダーの数を推奨最小値 (コアの合計数を4で割る) に設定しておき、後で次のような状況が発生したら、バックグラウンダープロセスの数を増やします。

- 抽出の更新の完了に時間がかかっている
- サブスクリプションおよびアラートが時間どおりに完了していない

専用のバックグラウンダー ノードまたは Tableau Prep Conductor ノードの場合、ワークロードとシステム リソースに応じて、バックグラウンダー インスタンスまたは Tableau Prep Conductor インスタンスの数を増減する必要がある場合があります。これらのインスタンスの数を調整すると、ノードのパフォーマンスに良い影響を与えたり、悪い影響を与えたりすることがあります。お客様には、RAM や CPU のリソースや Tableau Server のその他の側面を注意深く監視し、お使いの環境や負荷に最適な構成を決定する責任があります。ここで概説した推奨事項は、平均的なユースケースを処理するために設計された出発点であり、特定のニーズには適さない可能性があります。さらにサポートが必要な場合は、Tableau プロフェッショナル サービスにお問い合わせください。

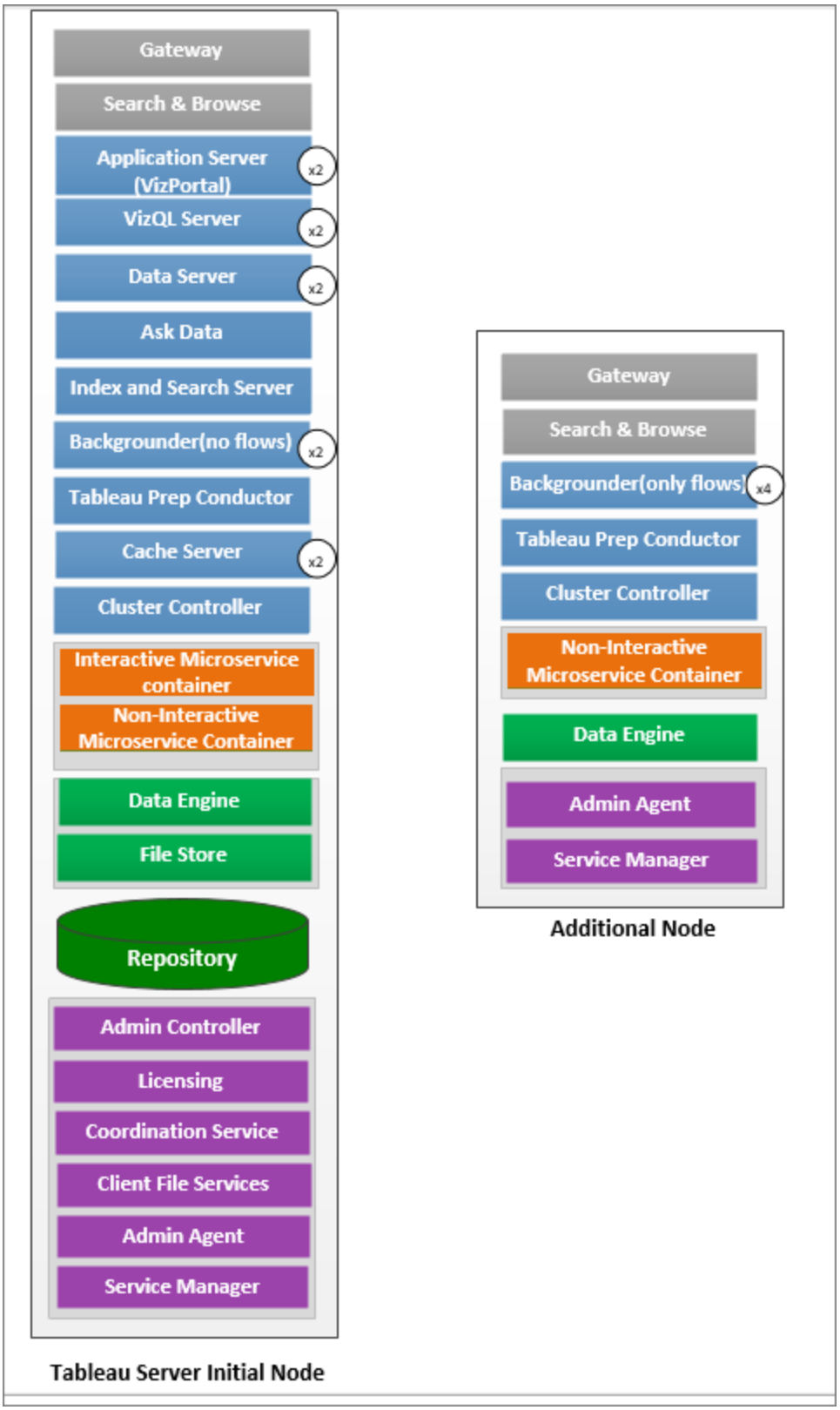
パフォーマンス調整の詳細については、パフォーマンスの調整トピックを参照してください。

2 ノード インストール - フロー環境用

Tableau Server でフローのパブリッシュ、スケジュール、管理を計画している場合、2 ノード構成から開始します。

> **重要:** 2 ノード構成は、高可用性のための最小要件を満たしません。可用性の高いシステムが必要な場合は、高可用性インストール (HA)を参照してください。

サーバー構成


Gateway
Search & Browse
Application Server (VizPortal)
x2
VizQL Server
x2
Data Server
x2
Ask Data
Index and Search Server
Backgrounder(no flows)
x2
Tableau Prep Conductor
Cache Server
x2
Cluster Controller
Interactive Microservice container
Non-Interactive Microservice Container
Data Engine
File Store
Repository
Admin Controller
Licensing
Coordination Service
Client File Services
Admin Agent
Service Manager
Tableau Server Initial Node
Gateway
Search & Browse
Backgrounder(only flows)
x4
Tableau Prep Conductor
Cluster Controller
Non-Interactive Microservice Container
Data Engine
Admin Agent
Service Manager
Additional Node

- 最初のノードで、すべてのプロセスをインストールします。以下は、8 コア マシン用のプロセスのインスタンス数です。

    - VizQL Server: 2 インスタンスに設定します(既定の計算: 物理コア数を 4 で割る、最大 4)。

    - キャッシュサーバーおよびデータサーバー: 2 インスタンスに設定します。「データに聞く」(Ask Data) の 1 つのインスタンスが、Data Server があるノードで自動的に構成されます。

    - バックグラウンダー: 最小 2、最大 4。上の図は、8 コア ノード用の最大数が表示されています。Tableau Prep Conductor では、バックグラウンダーがインストールされているノードで自動的に構成されます。最初のノードで `tsm topology set-node-role` の tsm 構成を使用し、フローを含むすべてのジョブ タイプを実行するようにバックグラウンダー ノードの役割を設定します。詳細については、tsm topology set-node-role を参照してください。

    - Index and Search Server: `indexandsearchserver.vmopts` TSM 構成オプションを使用して、パフォーマンスが向上するように Index and Search Server のメモリを構成できます。詳細については、tsm configuration set のオプションを参照してください。

    - その他のすべてのプロセスは、ハードウェアに関係なく、プロセスのインスタンスを 1 つだけインストールします。インタラクティブなマイクロサービス コンテナーの 1 つのインスタンスがアプリケーション サーバーが有効になっているノードにインストールされ、インタラクティブでないマイクロサービス コンテナーの 1 つのインスタンスがバックグラウンダーが有効になっているノードにインストールされます。

- 2 つ目のノードで、フローをサポートするためにバックグラウンダーと Tableau Prep Conductor を設定します。

    - 追加のノードにバックグラウンダーを分離し、そのノードではフローのみを実行するように設定します。この設定を構成するには、`tsm topology set-node-role` の

tsm コマンドを使用します。詳細については、tsm topology set-node-roleを参照してください。

注: 抽出負荷の高い環境でサーバー上のフローのスケジュールと管理を行う場合、または、負荷の高いワークフローに対応する必要がある場合は、以下の 3 のノード構成を使用することをお勧めします。

高可用性インストール (HA)

推奨事項

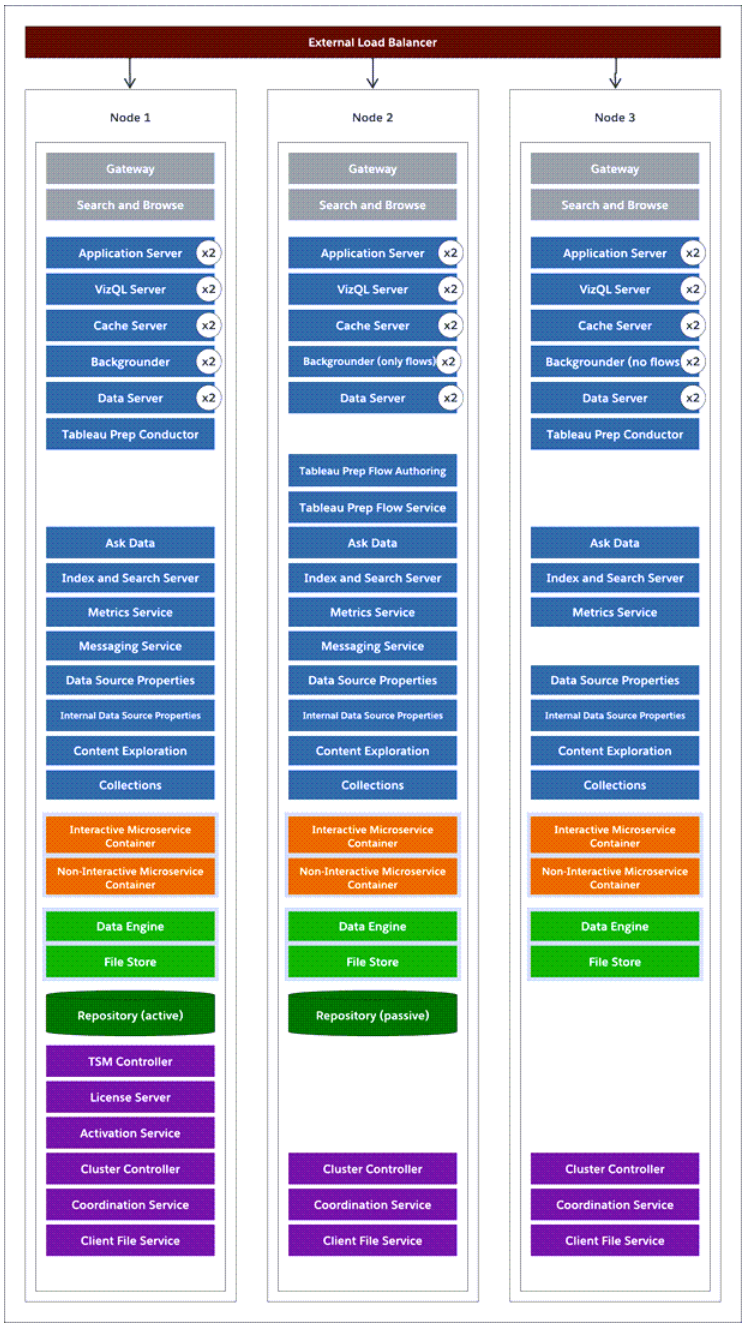
Tableau Server で可用性の高いインストールは、Tableau Server の可用性を最大化するように設計された分散インストールです。高可用性とは、基本的に、システムでダウンタイムを最小限に抑えながら利用可能な状態を保つことです。リポジトリ、ファイルの冗長性、フェールオーバーなど HA に関連するアイテムの冗長性を組み込むには、**最低でも 3 つのノード**が必要です。ダウンタイムに対する許容度は各組織によってさまざまであり、組織で確立されている SLA によって異なります。

高可用性を実現するには、単一障害点をなくし、障害を検出し、信頼性の高いフェールオーバーシステムを構築する必要があります。Tableau Server の HA は、主に次の要素によって実現されます。

- 複数のファイル ストア/データ エンジン インスタンスによるファイルの冗長性。
- 2 つのノードにまたがるアクティブ/パッシブ リポジトリ。
- 3 つのノードすべてにまたがる Index and Search Server
- 外部ロードバランサーの追加。それにより、インストール環境がゲートウェイ障害に対して堅牢になります。また、機能しているゲートウェイ プロセスのみに要求が送られるようになります。

サーバー構成

3 ノード構成:



- 冗長性を組み込むには、リポジトリやファイル ストア/データエンジンプロセスのホストインスタンスに追加ノードを追加する必要があります。ノード上のプロセスの複数インスタンスを含め、その他のプロセスのインスタンスを追加することができます。

- バックグラウンドジョブのタイプで冗長性を持たせるには、いずれかのノード(この例では最初のノード) ですべてのタイプのジョブを実行します。バックグラウンダーは既定ですべてのタイプのジョブを実行します。追加ノードのいずれかでフローのみを実行するようにバックグラウンダーを設定し、他の追加ノードではフロー以外のすべてのジョブを実行するように設定します。

- Tableau Server が正常に機能するかは、調整サービスが正常に機能しているかに依存します。3 ノード以上のサーバーインストールでは、新しい調整サービスアンサンブルを展開することによって、調整サービスの別のインスタンスを追加することをお勧めします。これにより、調整サービスのインスタンスの 1 つに問題が発生した場合に冗長性および向上された可用性が提供されます。詳細については、調整サービスアンサンブルの展開を参照してください。

- Index and Search Server のメモリは、冗長性確保のために 3 つのノードすべてに追加されているため、`indexandsearchserver.vmopts` TSM 構成オプションを使用して、パフォーマンスが向上するように構成することができます。詳細については、「indexandsearchserver.vmopts」を参照してください。

- システムの脆弱性を緩和するために、複数ゲートウェイおよび一部のサーバープロセスの追加インスタンスを実行することができます。この構成を達成するには最低 3 台のコンピューターが必要です。

- リポジトリも最初のノードから追加ノードの 1 つに移動され、2 番目のパッシブインスタンスが他の新しいノードに追加されました。

- インタラクティブなマイクロサービスコンテナーの 1 つのインスタンスがアプリケーションサーバーが有効になっているノードにインストールされ、インタラクティブでないマイクロサービスコンテナーの 1 つのインスタンスがバックグラウンダーが有効になっているノードにインストールされます。

  専用のバックグラウンダーノードまたは Tableau Prep Conductor ノードの場合、ワークロードとシステムリソースに応じて、バックグラウンダーインスタンスまたは Tableau Prep Conductor インスタンスの数を増減する必要がある場合があります。ノード上のこれらのインスタンスの数を調整すると、ノードのパフォーマンスに良い影響を与えたり、悪い影響を与えたりするこ

とがあります。お客様には、RAM や CPU のリソースや Tableau Server のその他の側面を注意深く監視し、お使いの環境や負荷に最適な構成を決定する責任があります。ここで概説した推奨事項は、平均的なユースケースを処理するために設計された出発点であり、特定のニーズには適さない可能性があります。さらにサポートが必要な場合は、Tableau プロフェッショナル サービスにお問い合わせください。

注: 特定の状況で、最初のノードで実行するプロセスを制限したい場合があるかもしれません。たとえば、ノード上でプロセスするリクエストを制限するために、実行するプロセス数をできるだけ少なくしたい場合などです。また、コアベースのライセンスがあり、最初のノードのコアでコアの使用をカウントされたくない場合、ライセンスされた Tableau Server プロセスをノードから削除することがあるかもしれません。Tableau Server ライセンス プロセスの詳細については、ノードのTableau Server プロセスを参照してください。

仮想マシンおよびパブリック クラウド展開

一般的に、このトピックで説明されている考慮事項と推奨事項は、仮想環境およびクラウド展開に適用されます。

仮想環境で Tableau Server を実行している場合は、VM ホスト上の物理 CPU コア数と関連して、vCPU 割り当てに対して VM ホストのベストプラクティスを使用します。通常、Tableau Server の場合は 2 vCPU = 1 物理コアです。たとえば、AWS インストールでは、4 コアの最小推奨事項が 8 AWS vCPU に相当します。

クラウドベース展開の詳細については、以下を参照してください。

- パブリック クラウド サービスで Tableau Server をセルフホストする

ベースライン以上の構成

ここで説明した制限を超える構成のシステムを計画している場合は、Tableau プロフェッショナル サービスにお問い合わせください。

高 VizQL のピーク使用に関する考慮事項

Tableau Server のパフォーマンスを最適化するには、展開内の専用ノードでリポジトリを分離することをお勧めします。Advanced Management ライセンスをお持ちの場合は、リポジトリを外部データベースとして実行することを検討してください。

組織のピーク負荷が 1 時間あたり1000 VizQL セッションを超える場合は、Tableau Server on Linux を実行することもお勧めします。このシナリオでは、VizQL セッションは、Tableau Server からビジュアライゼーションを表示または生成するユーザー アクションのことを指します。

詳細については、「Tableau Server 外部リポジトリ」を参照してください。

災害復旧の考慮事項

可用性の高い構成によってダウンタイムは減少しますが、それでも災害が起きたりハードウェアが故障したりした場合には、障害が発生する可能性があります。上記の考慮事項に加えて、組織において災害復旧の重要性を評価し、災害復旧の目標と目的を満たすことができる展開を計画する必要があります。

Tableau 環境で災害復旧 (DR) の計画を立てる場合、考慮すべき重要な要素が 2 つあります。

- 目標復旧時間 (RTO)。業務において完全復旧までの許容できるダウンタイム時間の指標。
    - バックアップの代替クラスターへの復元頻度とインフラへの投資額に影響します。
- 目標復旧時点 (RPO)。業務において許容できるデータ損失量の指標。
    - システムのバックアップの取得頻度に影響します。
    - Tableau Server の場合、RPO は、サーバーのフル バックアップを完了するのにかかる時間よりも短くすることはできません。

以下の図は、さまざまな RTO 要件に対応するための計画の立て方を示しています。

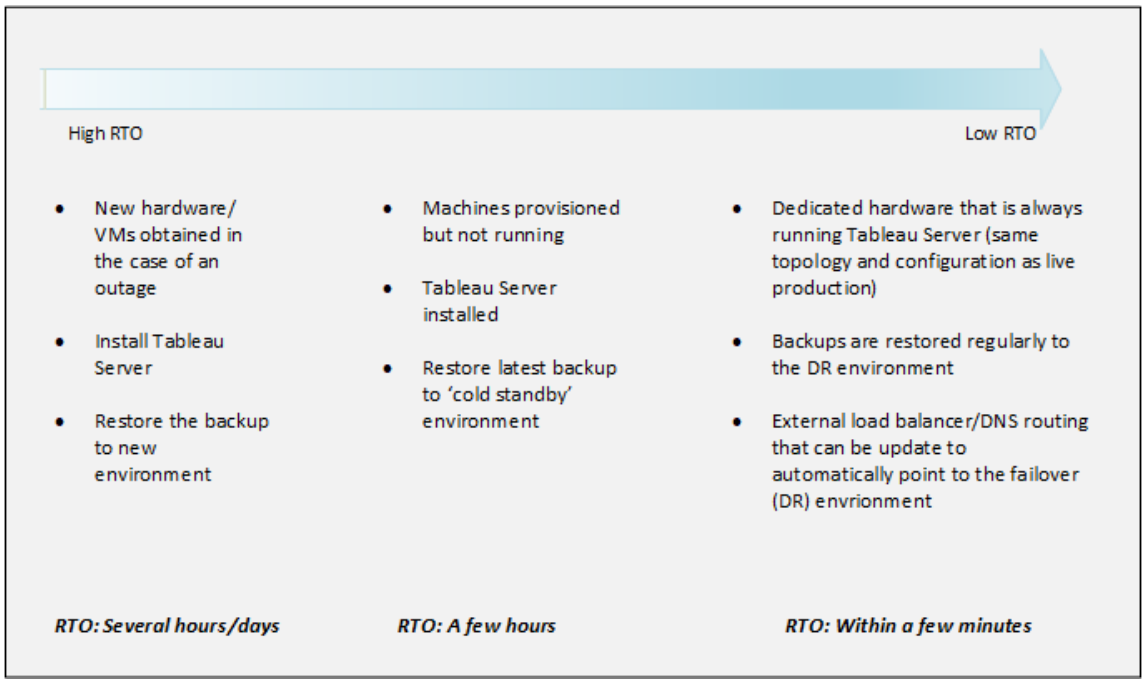


Tableau Server のスケーラビリティ

ニーズは変化したり増加したりするので、これらのベースライン構成では十分でない可能性があり、これらの構成以上に Tableau Server を拡張する必要が生じる場合があります。他のエンタープライズ プラットフォームと同様に、Tableau Server は、プロセッサー、メモリ、およびディスクを既存のノードに追加することでスケール アップできます。また、クラスターにノードをさらに追加することでスケール アウトすることもできます。ただし、スケーラビリティおよびパフォーマンスは、外部システムやユーザー アクティビティに大きく依存します。Tableau Server の構成は、以下のような要件や変数によって異なります。

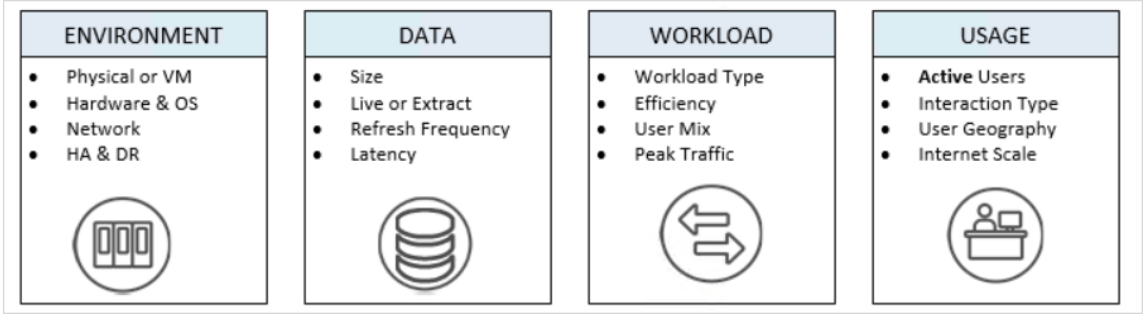


Tableau Server のスケーラビリティおよびスケーラビリティに影響を及ぼす変数の詳細については、ホワイトペーパー「Tableau Server のスケーラビリティ」(英語) を参照してください。

アイデンティティストア

Tableau Server にはユーザーやグループの情報を格納するためのアイデンティティストアが必要です。アイデンティティストアには、ローカルと外部の 2 種類があります。Tableau Server をインストールする場合は、ローカルまたは外部のアイデンティティストアを構成する必要があります。

アイデンティティストアの構成オプションについては、identityStore エンティティおよび外部のアイデンティティストアの構成リファレンスを参照してください。シングル アイデンティティストア モデルの柔軟性を向上させる方法については、「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。

ローカル アイデンティティストア

ローカルのアイデンティティストアで Tableau Server を構成する場合は、すべてのユーザー情報およびグループ情報は Tableau Server リポジトリに格納されて管理されます。ローカル アイデンティティストアのシナリオでは、ユーザーおよびグループ用の外部のソースはありません。

外部のアイデンティティストア

外部のストアで Tableau Server を構成する場合は、すべてのユーザー情報およびグループ情報は外部ディレクトリサービスによって格納され管理されます。Tableau Server は、ローカルのユーザーとグループのコピーが Tableau Server リポジトリに存在できるように外部のアイデンティティストアと同期する必要があります。しかし、ユーザーとグループのすべてのデータの信頼できるソースは外部のアイデンティティストアです。

Tableau Server アイデンティティストアを構成して外部 LDAP ディレクトリと通信する場合は、Tableau Server に追加するすべてのユーザー (初期の管理アカウントを含む) にディレクトリのアカウントを指定する必要があります。

Tableau Server が外部の LDAP ディレクトリを使用するように構成されている場合、外部ディレクトリから Tableau Server リポジトリへ、ユーザー ID をシステム ユーザーとして最初にインポートする必要があります。その後、ユーザーが Tableau Server にサインインすると、認証資格情報はユーザーの認証を処理する外部ディレクトリに渡されます。Tableau Server はこの認証を実行しません。ただし、アイデンティティストアに保存されている Tableau ユーザー名は Tableau Server の権限やパーミッションと関連付けられています。したがって、認証が検証された後、Tableau Server は Tableau リソースへのユーザー アクセス (認可) を管理します。

外部のユーザー ストアの一例となるのが Active Directory です。Tableau Server は Active Directory とのインターフェイスになるように最適化されています。たとえば、初期ノード設定の構成を使用して、Active Directory ドメインに参加している PC に Tableau Server をインストールする場合、ほとんどの Active Directory 設定はセットアップによって検出されて構成されます。一方、TSM CLI を使用して Tableau Server をインストールする場合は、すべての Active Directory 設定を指定する必要があります。この場合は、LDAP - Active Directory テンプレートを使用してアイデンティティストアを構成してください。

Active Directory にインストールする場合は、展開の前に外部 ID ストアを使用した展開におけるユーザー管理を確認することをお勧めします。

他のすべての外部ストアには、Tableau Server はアイデンティティストアとの通信を行う一般的な方法として LDAP をサポートします。たとえば、OpenLDAP は柔軟なスキーマを備えた LDAP サーバー実装の一種です。Tableau Server は、OpenLDAP サーバーにクエリを実行するように構成することができます。このためには、ディレクトリ管理者はスキーマに関する情報を提供する必要があります。セットアップ時に、初期ノード設定の構成 を使用して他の LDAP ディレクトリへの接続を設定します。

LDAP バインド

LDAP を使用してユーザー ストアにクエリを実行するクライアントには、セッションの認証と確立が必要です。これを行うのがバインディングです。バインディングにはいくつかの方法があります。シンプル バインドはユーザー名とパスワードを使用する認証です。シンプル バインドで Tableau Server に接続する組織は、SSL 暗号化接続を構成することをお勧めします。そうでないと認証資格情報が平文の状態でネットワーク上に拡散します。Tableau Server がサポートするもう一つのバインド

方法は GSSAPI です。GSSAPI は Kerberos を使用して認証します。Tableau Server の場合、Tableau Server がクライアントで外部のユーザー ストアが LDAP サーバーです。

GSSAPI (Kerberos) バインドを使用した LDAP

キータブ ファイルを使用して LDAP サーバーへの認証を行う場合は、GSSAPI を使用して LDAP ディレクトリにバインドすることをお勧めします。Tableau Server サービス専用のキータブ ファイルが必要です。また、SSL/TLS を使用して LDAP サーバーでチャネルを暗号化することもお勧めします。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

Active Directory にインストールしており、Tableau Server のインストール先のコンピューターが既にドメインに結合している場合、コンピューターには既に構成ファイルおよび keytab ファイルがある可能性があります。このような場合、Kerberos ファイルはオペレーティング システムの機能と認証のためのものです。厳密にはこれらのファイルを GSSAPI バインドに使用することはできますが、その使用を推奨していません。代わりに、Active Directory の管理者に連絡して Tableau Server サービス専用のキータブを要求してください。Keytab 要件の理解を参照してください。

お使いのオペレーティングシステムに定義域の認証用に適切に設定された keytab がある場合、Tableau Server のベース インストール二必要なものは GSSAPI バインド用の Kerberos キーファイルだけです。ユーザー向けに Kerberos 認証の使用を計画している場合は、インストール完了後にユーザー認証向けに Kerberos を構成 および データソースへの Kerberos 委任 を行います。

SSL 経由の LDAP

既定では、任意の LDAP サーバーへのシンプル バインドの LDAP は暗号化されません。LDAP サーバーとのバインド セッションの確立に使用されるユーザー認証資格情報は、Tableau Server と LDAP サーバー間でプレーンテキストで通信されます。Tableau Server と LDAP サーバー間のチャネルを暗号化することを強くお勧めします。

バージョン 2021.2 以降、Linux 上の Tableau Server では、Active Directory を ID ストアとして使用する場合、暗号化された LDAP チャネルが必要です。2021.2 以降にインストールまたはアップグレードする前に、有効な SSL/TSL 証明書をインストールする必要があります。推奨されていませんが、既定の暗号化 LDAP チャネルを無効にすることもできます。Active Directory およびその他の

LDAP サーバーの暗号化を有効または無効にする方法の詳細については、外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

システム ユーザーとグループ

Tableau Server on Linux では、1 人のユーザーと2 つのグループを使用して適切な操作を行います。ユーザーとグループはローカルに設定することも LDAP ディレクトリサービスから設定することも可能です。

ユーザー

Tableau Server ではサービス アカウントが必要です。このアカウントは、通常のログイン権限を持つ権限なしのユーザーです。既定では、Tableau Server のインストールでは、サービス アカウント用のローカル ユーザー `tableau` が作成されます。

Tableau Server サービス アカウントに既存のユーザー アカウントを使用する場合は、インストール中にアカウントの作成を無効にする必要があります。

特に、initialize-tsm スクリプトを実行するときは `--disable-account-creation` オプションを設定する必要があります。また、`--unprivileged-user`オプションを使用してアカウント名を指定する必要があります。指定したアカウントが存在しない場合は、初期化 tsm スクリプトによって作成されます。詳細については、initialize-tsm スクリプトの出力ヒントを参照してください。

既存のアカウントをこの `--unprivileged-user`オプションで指定する場合は、ユーザー アカウントが通常のログイン権限を持つ非特権ユーザーであることを確認します。次の特性を持つアカウントを構成します。

- シェルを `/bin/bash` に設定します。
- 利便性のため、データディレクトリパスに対するホーム ディレクトリの設定を検討してください。アカウントには、ホーム ディレクトリに対する所有権と書き込み権限がある必要があります。

セットアップ中に別の特権のないアカウントを指定する場合は、同じユーザーを systemd-journal グループに手動で追加する必要があります。Tableau Server が tsm maintenance ziplogs コマンドの実行時に「データに聞く」(Ask Data) などの一部のサービスからログを収集できるようにするに

は、特権のないユーザーは systemd-journal グループのメンバーである必要があります。特権を持たないユーザーがグループのメンバーでない場合、ziplog には影響を受けるサービスからのログは含まれません。

グループ

Tableau Server では操作に 2 つのグループが必要です。

既定インストールでは、ローカルの `tableau` サービスアカウントは `tableau` という名前のプライマリグループに属しています。ただし、インストール時に権限のない代替ユーザーを使用する場合は、その代替アカウントのプライマリグループが使用されます。必要に応じて、いずれかのアカウントをこのグループに追加し、Tableau Server ログファイルを読み取るようにすることができます (ルートにはなりません)。

2 番目のグループは、Tableau サービスマネージャー (TSM) の認証を許可されたユーザーを承認するために使用されます。このグループのユーザーは、コマンドを TSM に送信して Tableau Server 管理者に制限する必要があります。既定のグループ名は `tsmadmin` です。

既定の名前を使用しない場合は、初期化 tsm を実行する際に `--tsm-authorized-group` オプションを使用してグループ名指定する必要があります。詳細については、initialize-tsm スクリプトの出力ヒントを参照してください。

クライアントの認証

Tableau Server の基本のユーザー認証は、ローカルと外部のユーザーストアの両方において、ユーザー名とパスワードによるサインインで行います。ローカルの場合、ユーザーのパスワードはハッシュ化されたパスワードとしてリポジトリに保存されます。外部の場合は、Tableau Server から外部のユーザーストアに認証資格情報が渡され、これらが有効かどうかの回答があるまで待機します。外部のユーザーストアは、Kerberos やなど他の種類の認証も処理できますが、コンセプトはこの場合も同じで、Tableau Server が認証資格情報またはユーザーを外部のストアに委任して応答を待機します。

ユーザー名とパスワードによるサインインを無効化するように Tableau Server を構成することができます。このようなシナリオでは、信頼できる認証、OpenID、または SAML など、他の認証方法を使用できます。認証を参照してください。

場合によっては、LDAP 外部ディレクトリを更新して、Tableau Server からのユーザー名 + DN 形式でのバインド操作を許可する必要があります。サインイン時のユーザー バインド動作参照してください。

外部 ID ストアを使用した展開におけるユーザー管理

このトピックでは、外部の ID ストアを使用して Tableau Server のユーザーを管理する場合、熟知しておく必要がある技術的に重要な詳細について説明します。Tableau Server は、LDAP を使用した外部ディレクトリへの接続に対応しています。このシナリオでは、Tableau Server は外部 LDAP ディレクトリから Tableau Server リポジトリにユーザーをシステム ユーザーとしてインポートします。

任意の LDAP ディレクトリ

Tableau のシステム ユーザー名は、LDAP 設定の一部として設定した属性です ("cn" など)。これは、個々のユーザーのインポートとグループの同期機能の両方に当てはまります。外部のアイデンティティストアの構成リファレンスを参照してください。

サインイン時のユーザー バインド動作

DN が追加されたユーザー名を使用してバインドできるように、LDAP 構成を更新する必要がある場合があります。具体的には、Tableau Server が任意の LDAP ディレクトリを使用して構成されている場合は、LDAP 構成を更新する必要があります。(例: OpenLDAP) ユーザー名として UPN または メール アドレスを使用します。

Tableau Server は、サインイン時に指定されたユーザー名に基づいて特定のユーザーを検索します。その後、Tableau Server は、DN が付加されたユーザー名を使用してバインドを試みます。Tableau Server が GSSAPI を使用して構成されている場合は、username @ REALM (ドメイン名) が使用されます。

Active Directory

このトピックの以下のコンテンツでは、読者が Active Directory のユーザー管理および基本的な Active Directory スキーマやドメインの概念を理解していることを想定しています。

注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

Active Directory ユーザー認証と Tableau Server

Tableau Server はすべてのユーザー名を、リポジトリによって管理される Tableau Server アイデンティティストアに保存します。Tableau Server が認証で Active Directory を使用するように構成されている場合、最初にユーザー ID を Active Directory からアイデンティティストアにインポートする必要があります。その後、ユーザーが Tableau Server にサインインすると、認証資格情報はユーザーの認証を処理する Active Directory に渡されます。Tableau Server はこの認証を実行しません。(既定では、認証に NTLM が使用されていますが、シングル サインオン機能のための Kerberos または SAML を有効にすることができます。ただし、これらのすべての場合では、認証は引き続き Active Directory のままとなります。)ただし、アイデンティティストアに保存されている Tableau ユーザー名は Tableau Server の権限やパーミッションと関連付けられています。したがって、認証が検証された後、Tableau Server は Tableau リソースへのユーザー アクセス (認可) を管理します。

Active Directory ユーザー名属性と Tableau Server

Active Directory は、さまざまな属性を使用してユーザー オブジェクトを一意的に識別します。(詳細については、MSDN Web サイトの「ユーザー命名属性」を参照してください。)Tableau Server は 2 つの Active Directory ユーザー命名属性に依存しています。

- `sAMAccountName`。この属性は、古いバージョンの Windows で使用するために作成されたログオン名を指定します。多くの組織では、この名前は、`example\jsmith` のような形式を使用して、認証で使用する NetBIOS 名と組み合わせて作成されます。`example` が NetBIOS 名で、`jsmith` が `sAMAccountName` 値です。Windows のオリジナル設計により、`sAMAccountName` 値は 20 文字未満にする必要があります。

  Windows の **[Active Directory ユーザーとコンピューター]** 管理コンソールでは、この値はユーザー オブジェクトの **[アカウント]** タブで **[ユーザー ログオン名 (Windows 2000 以前)]** というラベルの付いたフィールドにあります。

- `userPrincipalName` (UPN). この属性は「`jsmith@example.com`」などの形式でユーザー名を指定します。ここで "`jsmith`" は UPN プレフィックス、"`@example.com`" は UPN

サフィックスです。

Windows の **[Active Directory ユーザーとコンピューター]** 管理コンソールで、UPN は、ユーザー オブジェクトの **[アカウント]** タブの 2 つのフィールド (**[ユーザー ログオン名]** フィールドとその隣にあるドメイン ドロップダウン リスト) を連結したものです。

Active Directory からユーザーを追加する

サーバー環境でそれらを入力するか、CSV ファイルを作成し、ユーザーをインポートすることで、Active Directory からユーザーを個別に追加できます。Active Directory 経由でグループを作成し、グループのユーザー全員をインポートして Active Directory ユーザーを追加することもできます。結果は、使用しているアプローチによって異なる場合があります。

UPN プレフィックスをユーザー名としてインポートする

UPN 全体をユーザー名としてインポートすることはできません。

ほとんどの場合、Tableau Server がアイデンティティストアにインポートするユーザー名は sAMAccountName 値になります。この動作の例外については、ナレッジ ベースの記事「Active Directory を使用した非標準シナリオで UPN プレフィックスをユーザー名としてインポートする」を参照してくださいTableau。

ユーザー グループの追加

Active Directory ユーザー グループをインポートした場合、Tableau は `sAMAccountName` を使用してグループからすべてのユーザーをインポートします。

Active Directory からユーザーを削除する際の動作の同期

Active Directory 同期操作を通じて Tableau Server からユーザーを自動削除することはできません。Active Directory のグループから無効化または削除されたユーザーは引き続き Tableau Server に残っているため、ユーザーのアカウントを完全に削除する前にユーザーのコンテンツを監視して再び割り当てることができます。

ただし、Tableau Server は Active Directory におけるそのユーザーのステータスがどのように変化したかに基づいて、異なる方法でユーザー オブジェクトを操作します。Active Directory でユーザーを

削除/無効化するか、Active Directory の同期されたグループからユーザーを削除するという2 つのシナリオがあります。

ユーザーを Active Directory から削除または無効化し、Tableau Server でそのユーザーのグループと同期すると、以下の事柄が生じます。

- ユーザーは、同期した Tableau Server グループから削除される。
- ユーザーに設定されているロール: "ライセンスなし" に設定されます。
- ユーザーは、引き続き [すべてのユーザー] グループに属します。
- ユーザーが Tableau Server にサインインできない。

Active Directory のグループからユーザーから削除し、Tableau Server でそのグループを同期すると、以下の事柄が生じます。

- ユーザーは、同期した Tableau Server グループから削除される。
- ユーザーの役割は保持され、「ライセンスなし」には設定されません。
- ユーザーは、引き続き [すべてのユーザー] グループに属します。
- ユーザーは引き続き Tableau Server に対するパーミッションとともに、[すべてのユーザー] グループが使用するパーミッションを供与されているすべてのコンテンツへのアクセス権を持ちます。

いずれの場合でも、Tableau Server からユーザーを削除するには、サーバー管理者は Tableau Server の [サーバー ユーザー] ページからユーザーを削除する必要があります。

ドメインのニックネーム

Tableau Server では、ドメイン ニックネームは Windows NetBIOS ドメイン名と同等です。Windows の Active Directory フォレストでは、完全修飾ドメイン名 (FQDN) に任意の NetBIOS 名を付けることができます。NetBIOS 名は、ユーザーが Active Directory にログインしたときにドメイン識別子として使用されます。

たとえば、FQDN `west.na.corp.lan` は「`SEATTLE`」というNetBIOS 名 (ニックネーム) で構成できます。そのドメイン内にいるユーザー「`jsmith`」は次のユーザー名のいずれかで Windows にログオンできます。

- `west.na.corp.example.com\jsmith`
- `SEATTLE\jsmith`

ユーザーに FQDN ではなくNetBIOS 名を使用して Tableau Server にサインインさせる場合、ユーザーがログインする各ドメインに対してニックネーム値が設定されていることを検証する必要があります。各ドメインでのニックネーム値の表示および設定方法の詳細については、editdomain を参照してください。

複数ドメインのサポート

次の場合、Tableau Server コンピューターのドメインとは異なるドメインからユーザーやグループを追加できます。

- サーバーが使用しているドメインとユーザーのドメインの間に双方向の信頼関係が設定されています。
- サーバーのドメインはユーザーのドメインを信頼します (1 方向の信頼)。Active Directory 展開のドメイン信頼要件を参照してください。

サーバー以外のドメインからユーザーやグループ初めて追加する場合、ユーザー名/グループ名と共に完全修飾ドメイン名を指定する必要があります。そのドメインから追加されたユーザーまたはグループは、ニックネームが NetBIOS 名と一致している場合は、ドメインのニックネームを使用して追加されます。Tableau Server が複数のドメインに接続する場合は、Tableau サービス マネージャーで `wgserver.domain.whitelist` (2020.3 以前のバージョン) または `wgserver.domain.accept_list` (2020.4 以降のバージョン) のオプションを設定して Tableau Server が接続する他のドメインも指定する必要があります。詳細については、wgserver.domain.whitelistまたはwgserver.domain.accept_listを参照してください。

重複する表示名

ユーザーの表示名が複数のドメインで一意ではない場合、Tableau で同じ表示名を使用してユーザーを管理すると混乱を招く可能性があります。Tableau Server は 2 人のユーザーに同じ名前を表示します。たとえば、2 つのドメイン (example.lan とexample2.lan など) を使用している組織について考えます。両方のドメインにユーザー "John Smith" が存在する場合、そのユーザーをグループやその他の管理タスクに追加すると、Tableau Server で混乱を招きます。このシナリオでは、アカウントを差別化するため、いずれかのユーザーの Active Directory での表示名を更新することを検討してください。

NetBIOS 名を使用して Tableau Server にサインインする

ユーザーはドメインのニックネーム (NetBIOS 名) を使用して Tableau Server にサインインできます。例: `SEATTLE\jsmith`。

Tableau Server は指定された FQDN の NetBIOS 名をクエリできません。その結果、Tableau はネームスペースへの最初の入力に従って指定された FQDN のニックネームを設定します。たとえば、FQDN `west.na.corp.lan` の場合、Tableau はニックネームを `west` に設定します。

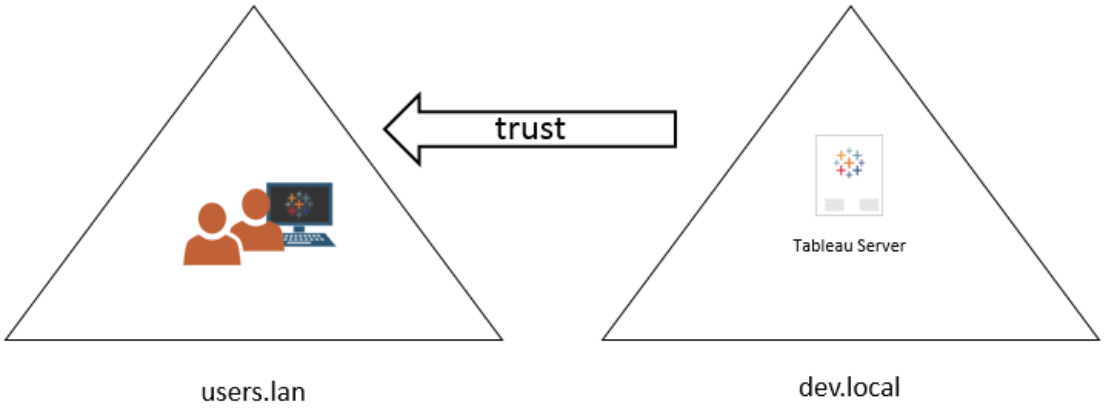
そのため、ユーザーがニックネームを使用しサインインする前に、Tableau Server でドメインのニックネームを更新する必要が生じることがあります。ニックネームを更新しない場合、完全修飾ドメイン名を使用してサインインする必要があります。詳細については、Tableau ナレッジ ベースの「新しいドメインからのユーザーがログインできず、ユーザー リストに表示されない」を参照してください。

Active Directory 展開のドメイン信頼要件

Tableau Server を複数のドメインにまたがる Active Directory 環境 (同じ Active Directory フォレスト内また異なるフォレスト内のいずれか) で使用している場合、一部の Tableau 機能はドメイン間の信頼関係に依存します。たとえば、一部の管理者が Tableau Server などのサーバー アプリケーションが展開されている場所とは別のドメイン内のユーザーを管理しています。他の組織では、Tableau Server の展開が、外部パートナーや、組織内の異なるパートナーと共有されている場合があります。最後に、Tableau Server が接続している SQL Server、MSAS、Oracle などの Windows で認証されたデータ ソースも、他のドメイン内に存在している場合があります。

実現可能な場合は、Tableau Server とやり取りするすべてのドメイン間で双方向の信頼関係を構成することをお勧めします。これが不可能な場合は、一方向信頼関係が構成されているユーザー認証をサポートするように Tableau Server を構成できます。このケースでは、Tableau Server がインストールされているドメインがユーザー アカウントの存在するドメインを信頼するように構成されている場合に、ドメイン間の一方向信頼関係がサポートされます。

次の図は、Tableau Server がインストールされているドメインとユーザー アカウントが存在するドメインとの間の一方向信頼関係を示しています。



このシナリオでは、Tableau Server は dev.local ドメイン内にあり、users.lan Active Directory ドメインのユーザーが Tableau Server にインポートされます。このシナリオでは一方向信頼関係が必要です。特に、dev.local ドメインは、users.lan ドメインを信頼するように構成されています。users.lan ドメイン内のユーザーは、通常の Active Directory 認証資格情報で dev.local 内の Tableau Server にアクセスできます。ただし、ユーザーがニックネームを使用してログオンする前に、Tableau Server でドメインのニックネームを更新する必要が生じることがあります。詳細については「Tableau ナレッジ ベース」をご覧ください。

このシナリオで Tableau Server を構成する場合は、インストール時にプライマリユーザードメインを指定します。初期ノード設定の構成を参照してください。Tableau Server が他の Active Directory ドメインに接続できるようにするには TSM で `wgserver.domain.accept_list` オプションを設定して Tableau Server が接続する他のドメインも指定する必要があります。詳細については、wgserver.domain.accept_list を参照してください。

ドメインの信頼のためにバインドアカウントを複製する

Tableau Server on Linux は、シンプル バインドを使用して Active Directory による認証を行う JDK の LDAP 実装に依存しています。シンプル バインドはドメイン対応ではないため、クロスドメイン バインドはサポートされません。最初のアイデンティティストアをセットアップするときは、Active Directory への認証に使用するバインドアカウントを指定する必要があります。

クロスドメイン信頼とディレクトリルックアップを有効にするには、すべてのターゲットドメインでこのバインドアカウントを複製する必要があります。各ドメインの各バインドアカウントは、同じユーザー名 (`sAMAccountName` または `dn`) とパスワードを使用する必要があります。

Kerberos シングル サインオンは、この一方向信頼関係シナリオでサポートされています。

外部 ID ストアを使用した展開におけるユーザー管理を見直し、複数のドメイン、ドメイン命名、NetBIOS、および Active Directory ユーザー名の形式が Tableau ユーザー管理にどのように影響するかを理解してください。

一方向信頼関係シナリオでのライブデータへの接続

一方向信頼関係シナリオでは、Tableau Server に接続しているユーザーは、クラウド内、または Windows 認証に依存していないオンプレミスの他のデータソースでホストされているライブデータに接続できます。

Windows 認証が必要なデータソースは、シナリオを複雑化する追加の認証要件を伴っていたり、Tableau Server ユーザーの接続を妨げたりすることがあります。これは、Tableau Server でこのようなデータソースの認証にサービス アカウントを使用するためです。Tableau Server を Windows 認証を使用するデータソースとは異なるドメインで実行している場合は、Tableau Server 用に使用するサービス アカウントがデータソースにアクセスできることを確認してください。

インターネットとの通信

多くの企業において、Tableau Server はインターネットと通信する必要があります。Tableau Server は保護された内部ネットワーク内で動作するように設計されました。Tableau Server をインターネット上や DMZ で直接設定しないでください。お使いのネットワークとインターネットとの間の通信は、プロキシ サーバーを使用して仲介するようお勧めします。Tableau Server を実行しているコンピューターがインターネットに直接アクセスできない場合は、ネットワーク内部からインターネット上のターゲットへのトラフィックを仲介するためのフォワードプロキシ サーバーを展開する必要が生じる場合があります。Tableau Server では、パススルー認証または手動プロキシ認証はサポートされていません。

インバウンド トラフィックの場合は、Tableau Server をリバースプロキシ サーバーの背後で実行することを推奨します。

Tableau がインターネットと通信する方法

Tableau Server は、インターネットへのアウトバウンドのアクセスを必要としています。その目的は以下のとおりです。

- マップの操作。Tableau は外部でホストされているマップデータを使用します。

  Tableau Server でマップを使用するには、ポート 443 を使用して次のインターネット上の場所に接続できる必要があります。

  - mapsconfig.tableau.com
  - api.mapbox.com

  Tableau がこれらの場所に接続できないと、マップの読み込みに失敗する場合があります。

  接続をテストするには、ブラウザーからアドレス `https://mapsconfig.tableau.com/v1/config.json` および `https://api.mapbox.com/` にアクセスし、json ファイルのダウンロードを求めるメッセージが表示されることを確認します。

  インターネットへの接続にプロキシを使用していて `api.mapbox.com` へ接続できない場合は、Mapbox Web サイトの「ファイアウォールの操作」を参照してください。

  Tableau Server バージョン 2019.1 以前のバージョンについては、Tableau ヘルプでお使いのバージョンのドキュメントを参照してください。

- Tableau 送信ログ サーバーへの接続。

  サポートを利用するとき、Tableau にログ ファイルをアップロードできます。`tsm maintenance send-logs` を参照してください。ファイルを正常に Tableau へアップロードするには、Tableau Server がポート 443 で送信ログ サーバーと通信できる必要があります。

  - report-issue.tableau.com:443
  - crash-artifacts-747369.s3.amazonaws.com

    - s3-us-west-2-w.amazonaws.com
    - s3-w-a.us-west-2.amazonaws.com

- 基本的な製品データの送信。

  ドメイン prod.telemetry.tableausoftware.com は、プロセスの起動とシャットダウンに関する基本的な製品データを Tableau が受信するために使用されます。また、一般的な製品利用状況データにも使用されます。

  このドメインへのトラフィックは、ポート 80 (製品データ クライアントの初期登録用) とポート 443 (後続のすべてのトラフィック) で発生します。

  prod.telemetry.tableausoftware.com:80

  prod.telemetry.tableausoftware.com:443

- ライセンスの発行。Tableau 製品は、プロダクト キーをライセンス認証するため、インターネットに接続します。オフライン ライセンス認証 ツールで Tableau ソフトウェアをライセンス認証する場合を除いて、すべての Tableau 製品では、ライセンスを検証するためにインターネットへのアクセスが必要になります。具体的には、ライセンス認証、ライセンス認証の解除、メンテナンス日の更新といったライセンス操作の際にインターネットへのアクセスが必要です。これらの操作の詳細については、「ライセンスの管理」を参照してください。

  Tableau Server は、プロダクト キーを有効化したり、製品を登録したり、Tableau Cloud にサインインしたりするときに、以下のインターネット サイトに接続する必要があります。

    - atr.licensing.tableau.com:443
    - licensing.tableau.com:443
    - register.tableau.com:443
    - o.ss2.us
    - s.ss2.us

- crt.rootca1.amazontrust.com
- crt.sca1b.amazontrust.com
- crt.sca0a.amazontrust.com
- crt.sca1a.amazontrust.com
- crt.sca2a.amazontrust.com
- crt.sca3a.amazontrust.com
- crt.sca4a.amazontrust.com
- *.digicert.com
- ocsp.*.amazontrust.com
- crl.*.amazontrust.com
- crt.rootg2.amazontrust.com

上記のドメインへのリクエストは、ポート80 またはポート443 で行われることがあります。ポート80 は、証明書の検証 (失効、証明書チェーンなど) に使用されます。ポート443 は SSL 接続に使用されます。

`ocsp.*.amazontrust.com` および `crl.*.amazontrust.com` ドメインへのリクエストは、証明書失効に関する情報であるため、Amazon によって管理されます。詳細については、「ACM certificate characteristics」を参照してください。Tableau を実行しているコンピューターの証明書信頼ストアに Amazon ルート証明書をインストールすることをお勧めします。Amazon ルート証明書をダウンロードしてインストールする方法については、Amazon Trust Services Web サイトの「認証局」を参照してください。

Tableau Server は、ライセンス認証を試行した際に接続できない場合、オフラインでライセンス認証を実行するよう要求されます。

Tableau のライセンス サーバーへの接続を診断するには、以下の URL をブラウザーまたは Tableau Server コンピューターの curl コマンドプロンプトに貼り付けます。

```
https://atr.licensing.tableau.com/_status/healthz
```

Tableau Server がライセンス サーバーにアクセスできた場合、「OK」メッセージが表示されます。そうでない場合は、「このページにアクセスできません」などのエラーが表示されます。この問題を解決するには、ネットワーク チームと協力して、Tableau Server コンピューター上の atr.licensing.tableau.com:443 へのアクセス制限を解除してください。

- 外部またはクラウド ベースのデータの操作。

  Tableau Server で Anaplan、Box、Dropbox、Google Drive、Google Sheets、OneDrive、Snowflake などのサービスを使用するには、次のインターネット上の場所に接続する必要があります。

  ```
  galop.connectors.tableau.com:443
  ```

- Tableau ダッシュボード拡張機能の操作

  サンドボックス化されたダッシュボードの拡張機能を使用するには、Tableau Server を次のインターネットの場所に接続する必要があります。

  ```
  extensions.tableauusercontent.com: 443
  ```

  詳細については、「Tableau Server でダッシュボードと Viz の拡張機能を管理する」を参照してください。

- Slack の操作

  Tableau を Slack ワークスペースに統合する場合は、特定の URL を Tableau の許可リストに追加するなど、多くのステップを実行する必要があります。このステップは、こちらに一覧表示されています。これを行う方法については、「Tableau と Slack ワークスペースの統合」を参照してください。

Tableau Server はインターネット アクセスなしで実行できます。インターネットへのアクセスなしで Tableau Server を組織内に展開する方法の詳細については、Tableau Server を未接続の (エアギャップ) 環境にインストールするを参照してください。

多くの企業において、ユーザーはネットワーク外 (つまり、インターネット) からも Tableau Server にアクセスする必要があります。たとえば、多くの企業の場合、ユーザーは、サーバーに格納されているビューを操作するために、モバイル デバイスから Tableau Server に到達できるようにします。インターネットまたはモバイル デバイスから Tableau Server へのアクセスを構成するには、リバース プロキシを使用することをお勧めします。Tableau Server のプロキシとロード バランサーの設定を参照してください。

セキュリティのベストプラクティスとして、TSM ポート (既定では 8850) をインターネットに公開しないでください。

Tableau Server のプロキシとロード バランサーの設定

多くの企業において、Tableau Server はインターネットと通信する必要があります。Tableau Server は保護された内部ネットワーク内で動作するように設計されました。Tableau Server をインターネット上や DMZ で直接設定しないでください。お使いのネットワークとインターネットとの間の通信は、プロキシ サーバーを使用して仲介するようお勧めします。フォワード プロキシ サーバーは、ネットワーク内部からインターネット上のターゲットへのトラフィックを仲介します。リバース プロキシ サーバーとロード バランサーは、インターネットからネットワーク内のターゲットへのトラフィックを仲介します。

誰がこの記事を読むべきか

この記事は、ネットワーク、ロード バランサー、ゲートウェイ プロキシの一般的なソリューションの経験がある IT プロフェッショナル向けです。この記事では、Tableau でいつ、どのようにインターネット アクセスが必要になるかを説明し、インターネットとのアクセスにプロキシ サーバーとロード バランサーを使用できるように、ご利用のネットワークと Tableau を設定する方法について説明します。多くのサード パーティ プロキシ ソリューションを使用できるので、この記事のいくらかのコンテンツは必然と一般的なものになります。

プロキシ サーバーを構成する前に、インターネットとの通信を参照してください。

フォワードプロキシ サーバーの構成

Tableau Server からインターネットへの通信を有効にするには、フォワードプロキシ サーバーの背後で Tableau Server を展開します。Tableau Server がインターネットへのアクセスを必要とする場合、要求を直接インターネットに送ることはしません。代わりに、フォワードプロキシに要求を送ることで、要求が転送されます。フォワードプロキシは、管理者がインターネットへのトラフィックで、ロードバランシング、サイトへのアクセスのブロックなどのタスクを管理するのをサポートします。

フォワードプロキシを使用する場合、トラフィックをフォワードプロキシに送るために、ネットワーク内で Tableau Server を実行するコンピューターを構成する必要があります。Tableau Server では、パススルー認証または手動プロキシ認証はサポートされていません。

フォワードプロキシ ソリューションで OpenID 認証を実行している場合は、追加の構成が必要です。「OpenID Connect に対応するように Tableau Server を構成する」を参照してください。

フォワードプロキシを使用するための Tableau Server on Linux の構成

インストール プロセスの一部としてフォワードプロキシ ソリューションを使用するように Tableau Server を構成することをお勧めします。具体的には、TSM のインストールと初期化の説明に従い、`./initialize-tsm` を実行時に Tableau Server を設定します。

以下の手順では、Linux 上の Tableau Server 用のフロントエンドプロキシ構成ファイルを作成する方法について説明します。

この構成ファイルは、次のディレクトリに保存されています。

```
~<unprivileged_user>/.config/systemd/tableau_server.conf.d
```

既定では、Tableau Server によって、権限なしユーザー、`tableau` が作成されます。したがって、構成ディレクトリへのデフォルトのパスは次のとおりです。

```
~tableau/.config/systemd/tableau_server.conf.d
```

このトピックおよび以下の構成ファイルのプロキシ構成ファイルは、`20-proxy.conf` と呼ばれます。このファイルには独自の規則に従って名前を付けることができますが、.conf 拡張子を使用する

必要があります。ファイル名に応じて、tableau_server.conf.d ディレクトリに格納されているファイルを語彙的オーダーで処理します。

1. tsm stop コマンドを実行します。
2. 権限なしのユーザーとしてセッションを開始します。既定では、インストール時に Tableau Server は tableau という権限なしユーザーを作成します。次のコマンドを実行します。

   ```
   sudo su -l tableau
   ```

3. tableau_server.conf.d ディレクトリに 20-proxy.conf ファイルを作成するか、開きます。セットアップ中にプロキシを構成した場合、20-proxy.conf ファイルは既に作成されています。

   - ファイルの作成。次のコマンドを実行します。

     ```
     touch ~tableau/.config/systemd/tableau_server.conf.d/20-proxy.conf
     ```

   - テキストエディターで 20-proxy.conf ファイルを開きます。

4. *プロキシ構成ファイルの内容* をファイルにコピーします。既存のファイルを編集している場合は、構成を削除しないように注意してください。*プロキシ構成ファイルの内容* には、フォワードプロキシ構成の設定方法に関する説明が含まれています。ファイルを編集して保存したら、ステップ 5 に進みます。

プロキシ構成ファイルの内容

```
# Always edit this file on Tableau Server as the unprivileged
user. By default, tableau, is the unprivileged user created by
Tableau Server during installation.
# Set environment variables http_proxy and https_proxy to point
to your proxy host.
# For example, to set the proxy to example-host for ports 80
```

```
and 443, run the following commands:
#
http_proxy=http://example-host:80/
https_proxy=http://example-host:443/
#
# Take care to use 'http' when you specify the URL for the
https_proxy environmental variable.
# Do not specify the 'https' protocol for the value of  the
https_proxy environmental variable.
#
# To bypass the proxy server, specify exceptions in the no_
proxy variable. Use this variable if your proxy server does not
route internal addresses.
# You must also add exceptions to this proxy configuration to
guarantee that all communications within a local Tableau Server
cluster (if you have one now or will have one later) do not
route to the proxy server.
# Enter both the host name and the IP address for each
computer. Additionally, include the cannonical host name
(localhost) and IP address (127.0.0.1) for the local computer.
# For example, to specify exceptions for a three-node cluster:
#
no_
proxy="localhost,127.0.0.1,hostname1,hostname2,hostname3,IP1,IP-
2,IP3"
#
# Where "hostname1" is the actual hostname of node 1, and "IP1"
is the actual IP address of node 1, etc.
```

5. Tableau シェルを終了します。次のコマンドを実行します。

   ```
   exit
   ```

6. TSM ビジネス サービスを再起動します。次のスクリプト コマンドを実行します。

```
sudo /opt/tableau/tableau_
server/packages/scripts.<version>/stop-administrative-services

sudo /opt/tableau/tableau_
server/packages/scripts.<version>/start-administrative-services
```

7. TSM を再起動します。

```
tsm restart
```

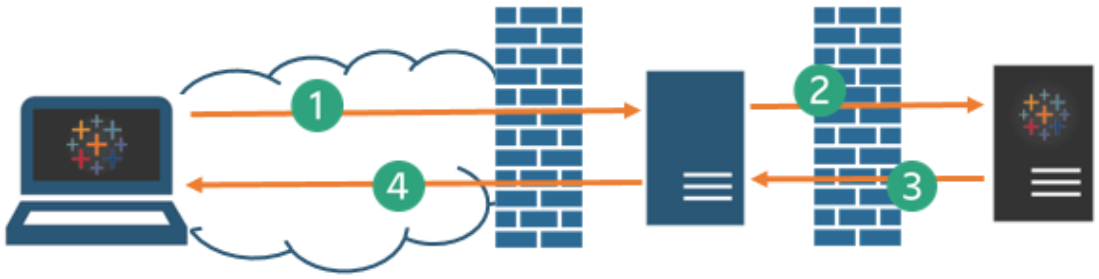
サーバー クラッシュ レポーター

組織でプロキシ サーバーを使用してインターネットに接続する場合は、Tableau Server のクラッシュレポーターでプロキシを使用するように構成する必要があります。Tableau Server でプロキシを使用するように既に構成している場合でも、別途、サーバー クラッシュレポーターも構成する必要があります。サーバー クラッシュレポーターでプロキシを構成するには、サーバー クラッシュレポーターの構成を参照してください。

リバース プロキシとロード バランサーが Tableau Server と連携するしくみ

リバースプロキシとロード バランサーは、外部 (インターネット) のクライアントから要求を受信し、Tableau Server に転送するサーバーです。これらのソリューションは、特定の Tableau Server の個別 IP アドレスをインターネットに公開しなくても、Tableau Server をインターネットで使用可能にします。また、社外のユーザーがアクセスできる場所にデータを格納しないよう、認証デバイスおよびパススルーデバイスとして動作させます。PCI や、HIPAA、SOX などのさまざまなプライバシー規制の対象となる組織にとって、この要件が重要となる場合があります。

次の図は、リバースプロキシやロード バランサー (LB) を使用するように設定されている Tableau Server に対し、クライアントが要求を行った場合の通信パスを示しています。



1. 外部クライアントは、Tableau Server への接続を開始します。クライアントは、リバースプロキシ サーバーや LB 用に設定されているパブリック URL (`https://tableau.example.com` など) を使用します (クライアントはリバースプロキシや LB にアクセスしていることを認識しません)。

2. 代わりに要求されるリバースプロキシ マップは、Tableau Server へのリクエストを渡します。シナリオによっては、Tableau Server に要求を渡す前提条件として、クライアントを (SSL/TLS を使用して) 認証するようリバースプロキシを構成することもできます。

3. Tableau Server は要求を取得し、リバースプロキシや LB に応答を送信します。

4. リバースプロキシや LB はコンテンツをクライアントに送り返します。クライアントに関しては、Tableau Server とやり取りを行ったにすぎないため、通信が中間サーバーによってパススルーされたことを知り得ません。

TLS/SSL

ゲートウェイのシナリオによっては、自社ネットワークの外部からのトラフィックに対して TLS/SSL を使用するようリバースプロキシとロード バランサーを設定する必要があります。これにより、プライバシー、コンテンツの整合性、および認証を確保するのに役立ちます。また、その他のセキュリティ対策を展開してインターネット ゲートウェイと Tableau Server の間のトラフィックを保護している場合を除き、ゲートウェイ プロキシと Tableau Server の間に SSL を構成することをお勧めします。内部証明書または自己署名した証明書を使い、Tableau Server と他の内部コンピューターとの間のトラフィックを暗号化することができます。

モバイル アクセス

Tableau Server は Tableau Mobile セッションのすべての HTTP 応答に X ヘッダーを追加します。既定では、大半のプロキシ ソリューションが X ヘッダーを維持します。ゲートウェイ ソリューションが X ヘッダーを維持しない場合は、モバイル クライアント セッションのすべての HTTP 応答でヘッダー `X-Tableau: Tableau Server` を維持するようにプロキシ サーバーとロード バランサーを設定する必要があります。

ゲートウェイで認証を設定している場合は、プロキシ サーバーや LB は Tableau Mobile の HTTP 要求に HTTP 302 で応答する必要があります。302 には、アイデンティティ プロバイダーのログイン

ページへのリダイレクトが含まれている必要があります。302 の認証シーケンスが示されている図を確認するには、Tableau コミュニティの「Tableau Mobile Authentication Sequence」(英語) を参照してください。

リバースプロキシ、ロードバランサー、ユーザー認証

Tableau Server は必ずユーザーを認証します。これは、組織用ゲートウェイでのインバウンド接続を認証する場合でも、Tableau Server がユーザーを認証することを示します。

ただし、すべてのクライアントがゲートウェイ ソリューションによるユーザー認証をサポートしているとは限りません。

- サポートしている Web ブラウザーでは、リバースプロキシや LB で SAML、OpenID Connect、Kerberos、Trusted Tickets、または手動の認証を使用できます。
- Tableau Mobile は、リバースプロキシや LB で SAML または手動の認証をサポートしています。iOS バージョンの Tableau Mobile では、リバースプロキシや LB を使用した Kerberos もサポートしています。上記と同様の推奨事項が適用されます。
- Tableau Prep では、リバースプロキシやロードバランサーを使用した認証はサポートしていません。リモートアクセスの場合は、VPN ソリューションを使用するか、Tableau Prep から Tableau Server に直接トラフィックをルーティングするようにゲートウェイ サービスを設定して認証を行います。
- Tableau Desktop では、トラフィックが最終認証のために Tableau Server にルーティングされる前に、認証モジュールがリバースプロキシで事前認証を実行している場合、リバースプロキシを使用した認証がサポートされます。詳細については、「*企業環境への Tableau Server 導入ガイド*」の「パート 5 - Web 層の構成」および「Independent Gateway を使用した認証モジュールの構成」を参照してください。

組織が Active Directory で認証している場合:

- Tableau Server を Kerberos 用に構成する前に、Tableau Server をリバースプロキシ用に構成する必要があります。詳細については、Kerberos の構成を参照してください。

リバースプロキシ サーバーやロードバランサーと連携させるための Tableau Server の設定

Tableau Server を構成する前に、プロキシ サーバー構成に関する次の情報を収集する必要があります。Tableau Server を構成するには、`tsm configuration set` コマンドを使用します。収集する必要のある情報は、`tsm` 実行の際に必要となるオプションに対応します。

以下の tsm オプションの大半も、ロードバランサーの背後で動作する Tableau Server 展開を設定するために使用できます。詳細については、ロードバランサーの追加を参照してください。

アイテム	説明	対応する **tsm configuration set** オプション
IP アドレスまたは CNAME	このオプションでは、IP アドレスまたは CNAME のいずれかを入力できます。 プロキシ サーバーとロードバランサーのパブリック IP アドレス。IP アドレスは、`203.0.113.0` などの IPv4 形式になります。また、静的 IP である必要があります。 静的 IP を提供できない場合や、クラウドプロキシまたは外部ロードバランサーを使用している場合は、クライアントが Tableau Server への接続に使用する CNAME (Canonical Name = 正規名) DNS 値を指定できます。この CNAME 値は、リバースプロキシ ソリューションで Tableau Server と通信するように設定する必要があります。	`gateway.trusted`
FQDN	Tableau Server へのアクセスに使用される完全修飾ドメイン名	`gateway.public.host`

アイテム	説明	対応する **tsm configuration set** オプション
	(tableau.example.com など)。Tableau Server は、このオプションのコンテキストの切り替えをサポートしていません。たとえば、以下の URL はサポートされていません。example.com/tableau。	
非完全修飾ドメイン名	プロキシサーバーや LB のサブドメイン名。tableau.example.com の例では、サブドメイン名は tableau です。	gateway.trusted_hosts
別名	プロキシ サーバーや LB の任意のパブリック代替名。たいていの場合、別名は CNAME 値を使って定義されます。例として、プロキシ サーバー bigbox.example.com および ftp.example.com や www.example.com の CNAME エントリがあります。	gateway.trusted_hosts
ポート	クライアントからリバースプロキシ サーバーへのトラフィックのポート番号。	gateway.public.port

Tableau Server の分散インストールを使用している場合は、クラスタ内の初期ノードで次の tsm コマンドを実行します。

1. 次のコマンドを入力して、クライアントがプロキシ サーバーや LB を通じて Tableau Server にアクセスするために使用する FQDN を設定します。ここで、*name* は FQDN です。

   ```
   tsm configuration set -k gateway.public.host -v 'name'
   ```

   たとえば、ブラウザーに https://tableau.example.com と入力して Tableau Server に到達した場合、次のコマンドを入力します。

```
tsm configuration set -k gateway.public.host -v
'tableau.example.com'
```

2. 次のコマンドを入力して、プロキシ サーバーや LB のアドレスまたは CNAME を設定します。ここで、*server_address* は IPv4 アドレスまたは CNAME 値です。

   ```
   tsm configuration set -k gateway.trusted -v 'server_ip_address'
   ```

   組織が複数のプロキシ サーバーや LB を使用している場合は、複数の IPv4 アドレスをコンマで区切りながら入力します。IP 範囲はサポートされていません。Tableau Server のスタートアップと初期化を改善するために、gateway.trusted のエントリー数を最小限に抑えます。

3. 次のコマンドを入力し、完全修飾ドメイン名や非完全修飾ドメイン名、別名など、プロキシ サーバーや LB の代替名を指定します。名前が複数ある場合は、名前をコンマで区切ります。

   ```
   tsm configuration set -k gateway.trusted_hosts -v 'name1,
   name2, name3'
   ```

   例:

   ```
   tsm configuration set -k gateway.trusted_hosts -v
   'proxy1.example.com, proxy1, ftp.example.com, www.example.com'
   ```

4. プロキシ サーバーがインターネットとの通信に SSL を使用している場合、次のコマンドを実行します。これは、リバースプロキシ サーバーがポート 80 の代わりにポート 443 を使っていることを Tableau に伝えます。

   ```
   tsm configuration set -k gateway.public.port -v 443
   ```

   > **注**: プロキシ サーバーが Tableau Server との通信に SSL を使用している場合、Tableau Server で SSL を設定して有効化する必要があります。

5. 次のコマンドを入力し、構成の変更をコミットします。

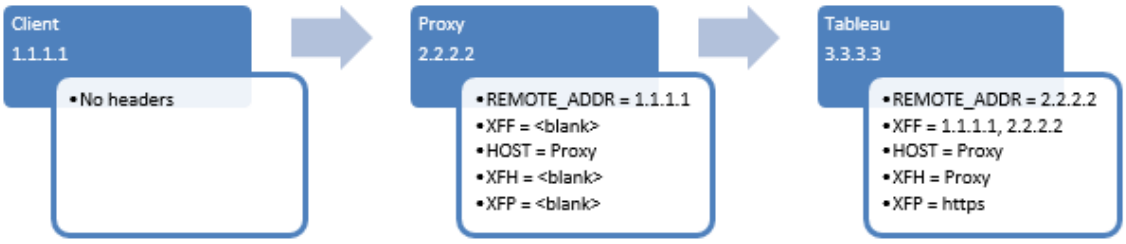
```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

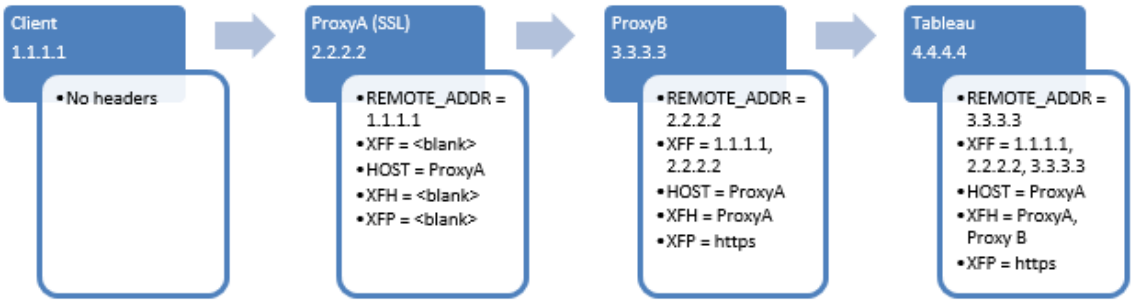
Tableau Server と連携させるためのリバースプロキシ サーバーやロード バランサーの設定

クライアントがリバースプロキシやロード バランサーを通して Tableau Server にアクセスする場合、特定のメッセージ ヘッダーを保持 (または追加) する必要があります。具体的には、メッセージ チェーン内のすべてのサーバーを `gateway.trusted` および `gateway.trusted_hosts` の設定で示す必要があります。

次の図はシングル ホップのメッセージ チェーン用のヘッダー例を表しています。ここでは、プロキシ サーバーが直接 Tableau Server と通信しています。



次の図はマルチ ホップのメッセージ チェーン用のヘッダー例を示しています。ここでは、メッセージが Tableau Server に接続する前に 2 つのプロキシ サーバーを横断しています。



次の表は、これらのヘッダーについて、また Tableau Server 設定の構成にどう関連しているかについて説明します。

ヘッダー	説明	関連した Tableau Server 設定
REMOTE_ADDR と X-FORWARDED-FOR (XFF)	Tableau Server は、要求元の IP アドレスを特定するために、これらのヘッダーを必要としています。X-FORWARDED-FOR ヘッダーでは、IP アドレスのチェーンを接続が起きた順に Tableau Server に示す必要があります。	gateway.trusted に設定した IP アドレスは REMOTE_ADDR で表される IP と一致しなくてはなりません。gateway.trusted に複数のアドレスを送信した場合、そのうち 1 つが REMOTE_ADDR で表される IP に一致しなくてはなりません。
HOST と X-FORWARDED HOST (XFH)	これらのヘッダーは、Tableau Server がクライアントに返信する際、Tableau Server への絶対リンクを生成するために使用されます。X-FORWARDED-HOST ヘッダーでは、ホスト名を接続が起きた順に Tableau Server に示す必要があります。	X-FORWARDED-HOST ヘッダーに表示されるホスト名は gateway.trusted_hosts で指定されるホスト名に含まれている必要があります。
X-FORWARDED-PROTO (XFP)	クライアントからプロキシへのトラフィックで SSL が有効化され、プロキシから Tableau Server へのトラフィックでは有効化されていない場合、このヘッダーが必要です。	リバースプロキシのポート構成 (クライアントからのインバウンド接続と Tableau Server へのアウトバウンド接続) は、

	メッセージ ルートの各ホップで HTTP や HTTPS が保持されないシナリオでは、 X-FORWARDED-PROTO ヘッダーが重要です。たとえばリバース プロキシが外部要求に SSL を必要としているが、リバース プロキシと Tableau Server の間のトラフィックが SSL を使用するよう構成されていない場合は、X-FORWARDED-PROTO ヘッダーが必要です。プロキシ ソリューションによって、X-FORWARDED-PROTO ヘッダーが自動的に追加される場合と追加されない場合があります。最後に、プロキシ ソリューションによっては、ポート転送を構成し、要求をポート 443 からポート 80 へ変換しなければならない場合があります。 関連する KB 記事:「"Unable to Sign In" and "Invalid username or password" Error With SAML After Upgrading」。	対応するパラメーター: gateway.public.port で指定する必要があります。これはプロキシへの接続に使用されるポート クライアントです。 プロキシ サーバーが Tableau Server との通信に SSL を使用している場合、Tableau Server で SSL を設定して有効化する必要があります。

リバース プロキシとロード バランサーの設定を検証する

ゲートウェイから Tableau Server への設定を検証するには、Tableau Server の Web 作成または Tableau Desktop を使用して、ワークブックとデータソースをパブリッシュします。Web ブラウザーを使用してインターネットから Tableau Server に接続している場合、推奨ブラウザーを使用していることを確認してください。既存のデータソースおよびパブリッシュしたデータソースを使用するワークブックをパブリッシュして表示します。以下のリンクを参照して、エンドユーザーによる Tableau Server の接続について理解してください。

タスク	ドキュメント
Web 作成の概要。	Web での Tableau の使用
Tableau Desktop または Web ブラウザーから Tableau Server にログインする。	Tableau Server または Online へのサインイン
ワークブックを Tableau Server にパブリッシュする。	ワークブックのパブリッシュ
データ ソースをパブリッシュする。	データ ソースのパブリッシュ
Tableau Server からワークブックを開く。	サーバーからワークブックを開く
サーバーからログアウトする (Desktop で)。	Tableau Server または Online へのサインイン
Web ブラウザーからワークブックをダウンロードする。	ワークブックのダウンロード
tabcmd (非サーバー クライアントから) が正常に動作しているかを確認する。	tabcmd

関連トピック

- Tableau Desktop インターネット アクセス要件
- ロード バランサーの追加

展開

このセクションのトピックでは、Tableau Server on Linux のインストール、構成、アップグレードの詳細に関して取り上げます。

> Tableau Server on Windows については、「Tableau Server のインストールと構成」を参照してください

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

Tableau Server のインストールと構成

このページの下部で参照されているトピックは、Tableau Server のインストールおよび構成を行うためのステップを説明しています。分散展開 (クラスター) をインストールする場合、このトピックの手順を使用して最初のノードをインストールし、次に追加ノードをインストールするために分散型高可用性 Tableau Server インストールを参照してください。

インストールの実行後、ライセンス認証、Tableau Server の登録、認証などのさまざまな設定を行い、セットアップを続行する必要があります。

その他のインストール方法

Tableau Server のインストールに使用できる代わりの方法がいくつかあります。

- 本番以外の環境で Tableau Server をインストールするためのクイックスタート手順が必要な場合は、インストールのジャンプスタートを参照してください。
- エンタープライズ対応の 4 ノードのリファレンスアーキテクチャを階層型データセンターに導入する方法を説明するエンドツーエンドの手順については、「企業環境への Tableau Server 導入ガイド」を参照してください。
- インターネットへの接続がない環境で Tableau Server をしている場合は、Tableau Server を未接続の (エアギャップ) 環境にインストールするを参照してください。
- Tableau Server をさまざまなクラウドプラットフォームにインストールすることもできます。パブリック クラウド サービスで Tableau Server をセルフホストするを参照してください。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

はじめる前に

Tableau Server をインストールするには、ハードウェア要件を満たすコンピューターが必要です。使用するコンピューターが最小要件を満たしているが、推奨される最小要件を満たしていない場合、情報メッセージが表示されます。この場合は、コンピューターのハードウェアは Tableau のトライアル版をインストールできますが、これは本番環境には適切ではありません。詳細については、インストールの前に...を参照してください。

インストール手順

次の手順では、1 台のコンピューターで Tableau Server をインストールする方法について説明します。この手順を使用して、単一サーバー展開に Tableau Server をインストールします。この手順を使用して、マルチノード Tableau Server 展開に最初のノードをインストールします。手順を連続して実行します。

1. TSM のインストールと初期化
2. Tableau Server のライセンス認証と登録
3. 初期ノード設定の構成
4. 管理者アカウントの追加
5. インストールの検証

インストールの前に...

注: Tableau Web サイトの Tableau Server の技術仕様に関するその他の情報は、こちらで見つけることができます。

このトピックでは、Tableau Server を稼働環境にインストールする前に考慮する必要のある要件および推奨事項について説明します。

- 初心者が組織に Tableau Server を展開する場合には、最初にテスト環境の単一サーバーとして Tableau Server を展開することをお勧めします。単一サーバーを最も簡単にインストールするには、「インストールのジャンプスタート」のステップに従います。
- エンタープライズ対応の 4 ノードのリファレンス アーキテクチャを階層型データ センターに導入する方法を説明するエンドツーエンドの手順については、「企業環境への Tableau Server 導入ガイド」を参照してください。
- 分散クラスターに Tableau Server を展開している場合は、このトピックに記載された要件および推奨事項に加えて分散環境の要件も確認してください。

- Tableau Server on Windows から Tableau Server on Linux に移行しようとしている場合には、Tableau Server の Windows から Linux への移行を参照してください。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

本番環境インストールに関するハードウェアの推奨事項

次のリストは、本番環境で使用するために Tableau Server を単一ノードにインストールする場合の最小ハードウェア推奨事項です。

重要: これらの推奨事項は最低限のものであり、ご利用のインストールおよび組織の要件を反映していない場合があります。たとえば、抽出やフローをパブリッシュするかどうか、Tableau Server にパブリッシュするワークブックの数など、ディスク領域の要件に影響を及ぼす要因がいくつかあります。空きディスク領域の要件に影響を与える可能性のあるものの詳細については、ディスク容量の要件を参照してください。

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
シングル ノード	• 64 ビット (x86_64 チップセット) • SSE4.2 および POPCNT 命令セットをサポートする必要があります	8 コア (16vCPU)、2.0 GHz 以上	バージョン 2022.3 以降: • 128 GB	50 GB

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
	• ARM ベースのプロセッサーはサポートされていません		バージョン 2021.4.0 からバージョン 2022.1.x: • 64 GB バージョン 2021.3.x 以前: • 32 GB	
	Tableau Server インストールに Tableau Prep Conductor を追加する場合、2 番目のノードを追加し、これを Tableau Server Prep Conductor の実行専用にすることをお勧めします。このノードには最低 4 コア (8 vCPU)、RAM 16 GB が必要です。			
	推奨されるスワップ メモリ: システム RAM の平方根または 8 GB のいずれか大きい方。			
マルチノードおよびエンタープライズ展開	技術的ガイダンスについては Tableau にお問い合わせください。 ノードは最小ハードウェア推奨事項を満たしているか、それを超える必要があります。ただし、次の場合は除きます。 • バックグラウンダーのインスタンスを最大 2 つ実行している専用バックグラウンダー ノード: 4 コアは許容可能。 • Tableau Prep Conductor の専用ノード: 最低 4 コア (8 vCPU)、RAM 16 GB。 • 独立したゲートウェイの専用ノード: 最低 2 コア (4 vCPU)、8 GB の RAM、および 100 GB の空きディスク領域。			

重要:ディスク領域の要件は TSM を初期化するまでは確認できません。十分な領域がない場合、Tableau Server パッケージのインストールが終了するまでその旨を通知されません。

50 GB のディスク空き領域。少なくとも 15GB が /opt ディレクトリに割り当てられ、残りがデータストレージ用として /var ディレクトリに割り当てられます。

- 空きディスク容量は、Tableau Server セットアッププログラムの解凍後に計算されます。セットアッププログラムでは約 1 GB の容量を使用します。抽出を使用するかどうかなどさまざまな要因に応じて、追加ディスク領域の割り当てが必要になる場合があります。

  Tableau Server の中核部分は、15 GB 以上の空きディスク領域があるディレクトリにインストールする必要があります。十分なスペースがないコンピューターに Tableau Server をインストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行することができません。既定のインストール場所は /opt ディレクトリです。RHEL ディストリビューションで Tableau Server のインストール パスを変更することができます。

  抽出を頻繁に使用する予定がある場合は、追加ディスク領域の割り当てが必要になる場合があります。インストール時に、データ(抽出)を保存するための別のディレクトリを指定することができます。

- **外部ファイル ストアに対するネットワーク接続ストレージ領域の要件:** 外部ファイル ストアを使用した Tableau Server を構成する場合は、ネットワーク接続ストレージ専用のストレージ領域の容量を見積もる必要があります。

  ストレージ サイズの見積もり: パブリッシュと抽出更新に必要なストレージの容量を考慮する必要があります。さらに、トピックオプション 2: リポジトリを個別にバックアップするで説明されているように、リポジトリのバックアップを個別に実行するオプションを特に選択しない限り、リポジトリのバックアップ サイズも考慮する必要があります。

  - 抽出:
    - Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。

- 抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。

- リポジトリのバックアップ:

  - リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。
  - リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、`workgroup.pg_dump` ファイルのサイズを確認します。

- コア数は "物理" コアに基づいています。物理コアは実際のサーバー ハードウェアまたは仮想マシン (VM) 上のコアを表す場合があります。コア数のカウントの目的でのハイパー スレッドは無視されます。

- 表示されている RAM はシングル ノード インストールでの最小限の推奨です。たとえば、アクティビティ、ユーザー数、バックグラウンド ジョブなどに応じて、RAM が増えるとインストールの機能が向上する場合があります。

推奨事項の全リストと最小要件を確認するには、Tableau Server の最小ハードウェア要件と推奨事項を参照してください。Tableau がスケーラビリティのテストのために内部で使用するハードウェア仕様については、「本番環境インストールに関するハードウェアの推奨事項」を参照してください。

アマゾン ウェブ サービスおよび Google Cloud Platform でのパブリック クラウド展開の場合、その "vCPU" は完全な CPU コアではなく、実際には CPU ハイパースレッドである場合があります。クラウドのインスタンスのサイズを変更する場合、Tableau Server の CPU コア要件として指定されるサイズの 2 倍の vCPU が必要になります (トライアルの最小インストールで 8 vCPU、単一ノードの推奨インストールでは 16 vCPU)。

オペレーティング システム要件

重要: Linux で Tableau Server バージョン 2023.3.1 - 2023.3.17、2024.2.1 - 2024.2.10、および 2025.1.0 - 2025.1.1 を実行している場合に、初期化またはライセンスの有効化が失敗する既知の問題があります。この問題は、2025 年 5 月のメンテナンス リリース以降、バージョ

ン2023.3.18、2024.2.11、2025.1.2 で修正されています。これを解決するには、このナレッジベースの記事を参照してください。

次の Linux ディストリビューションがサポートされています。

	2022.3.-0 以前 2022.3.-3	2022.3.-4 以降	2023.1.-0 以前 2023.1.-7	2023.1.-8 以降	2023.3.-0	2023.3.-1 - 2024.2.-11	2024.2.-12 以降	2025.1.-x
AlmaLinux 8.x						✓	✓	✓
AlmaLinux 9.x						✓	✓	✓
Amazon Linux 2	✓	✓	✓	✓	✓	✓	✓	
Amazon Linux 2023						✓	✓	✓
CentOS 7.9 以降 (8.x を除く)	✓	✓	✓	✓	✓	✓	✓	
CentOS Stream 8.x						✓	✓	✓
CentOS Stream 9.x						✓	✓	✓
Oracle Linux 7.3	✓	✓	✓	✓	✓	✓	✓	

	2022.3.-0 以前 2022.3.-3	2022.3.-4 以降	2023.1.-0 以前 2023.1.-7	2023.1.-8 以降	2023.3.-0	2023.3.-1 - 2024.2.-11	2024.2.-12 以降	2025.1.-x
以降								
Oracle Linux 8.x						✓	✓	✓
Oracle Linux 9.x						✓	✓	✓
RHEL 7.3 以降	✓	✓	✓	✓	✓	✓	✓	
RHEL 8.3 以降	✓	✓	✓	✓	✓	✓	✓	✓
RHEL 9.x					✓	✓	✓	✓
Rocky Linux 8.x						✓	✓	✓
Rocky Linux 9.x						✓	✓	✓
Ubuntu 16.04 LTS	✓	✓	✓	✓	✓	✓	✓	
Ubuntu 18.04 LTS	✓	✓	✓	✓	✓	✓	✓	
Ubuntu 20.04 LTS		✓	✓	✓	✓	✓	✓	✓
Ubuntu				✓		✓	✓	✓

	2022.3.-0 以前 2022.3.-3	2022.3.-4 以降	2023.1.-0 以前 2023.1.-7	2023.1.-8 以降	2023.3.-0	2023.3.-1 - 2024.2.-11	2024.2.-12 以降	2025.1.-x
22.04 LTS								
Ubuntu 24.04 LTS							✓	✓

Tableau Server in a Container の Linux ディストリビューションの要件については、「構築でサポートされているディストリビューション」を参照してください。

重要: 最新のセキュリティおよび機能パッチを確実に入手できるように、サポートされている最新バージョンの Linux ディストリビューションを使用して Tableau を展開することを強くお勧めします。Tableau は通常、サポートされているディストリビューションのメジャーバージョンの最新マイナーバージョンでテストおよび検証します。

Linux ディストリビューションに関するその他の注意事項:

- Red Hat Enterprise Linux (RHEL)、CentOS、Oracle Linux、Amazon Linux ディストリビューションは、このドキュメントでは総称して RHEL ライクと呼ばれます。
- CentOS 7、RHEL 7、Oracle Linux 7、Ubuntu 16.04 LTS、Ubuntu 18.04 LTS、Amazon Linux 2 の各ディストリビューションは、バージョン 2025.1.0 以降サポートされなくなりました。
- 2022 年 7 月の時点で、Debian ディストリビューションはサポートされなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください。
- Ubuntu の LTS 以外のリリースはサポートされていません。
- Tableau Server バージョン 2025.1.0 で Ubuntu バージョン 24.04 のサポートが追加されました。

- Ubuntu バージョン 20.04 のサポートは、Tableau Server バージョン 2023.1.0 および Server メンテナンス リリース 2022.1.12 および 2022.3.4 で追加されました。それ以前のバージョンではサポートされていません。
- カスタム カーネルはサポートされていません。

Tableau Server のマルチノード インストールでは、Tableau をインストールしているすべてのコンピューター ノードが Linux と同じディストリビューションの Linux を稼働している必要があります。

インストール ディレクトリ

Tableau Server の中核部分は、既定では `/opt` ディレクトリにインストールされます。

- Tableau Server のインストール先ディレクトリには、それに割り当てられている最低 15 GB の空きディスク領域が必要です。十分なスペースがないコンピューターに Tableau Server をインストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行することができません。
- RHEL ライク システムでは規定以外のインストール ディレクトリを指定できますが、Ubuntu では場所を変更できません。
- RHEL ライク システム上で規定以外のインストール場所を指定する場合は、シンボリック リンクまたはネットワーク ファイル システム (NFS) ボリューム上のディレクトリ位置を指定しないでください。

データディレクトリ

Tableau Server は既定で、Tableau が管理するすべてのコンテンツおよび抽出用のデータディレクトリを作成します。ディレクトリは `/var/opt/tableau/tableau_server` に作成されます。

インストール時にデータ(抽出) を保存するための別のディレクトリを指定することができます。別のディレクトリを使用する計画がある場合は、ディレクトリを作成しないでください。代わりに、Tableau Server のセットアップによりディレクトリを作成してください。データディレクトリには、インストール プロセスの際に設定される特定のパーミッションが必要です。

データディレクトリを変更するには、`initialize-tsm` スクリプトを実行するときに、データディレクトリパスと共にフラグを渡す必要があります。initialize-tsm スクリプトの出力ヒントを参照してください。

既定のデータディレクトリを変更する場合:

- ネットワーク ファイル システム (NFS) ボリューム上でシンボリック リンクやデータ ディレクトリの場所を指定することはしないでください。
- パスにピリオドやスペースが含まれているデータ ディレクトリの場所を指定することはしないでください。パスにピリオドやスペースが含まれていると、初期化が失敗します。
- データ ディレクトリは、インストール ディレクトリとは別のディレクトリにインストールする必要があります。

重要: `initialize-tsm` を実行した後でデータ ディレクトリの場所を変更することはできません。データ ディレクトリの場所は、以降のアップグレードを含め、展開の寿命が終わるまで保持されます。

Tableau Prep Conductor

Tableau Prep Conductor は Tableau Server でのプロセスの 1 つです。フローの実行、接続の認証資格情報の確認、フローが失敗した場合のアラート送信を行います。Tableau Prep Conductor は Tableau Server のスケジュール調整機能および追跡機能を活用しているため、データが変更になる際に、Tableau Prep Builder にログインして個々のフローを手動で実行するのではなく、フロー実行を自動化してフロー出力を更新することができます。

Tableau Prep Conductor は個別にライセンスされ、データ管理 ライセンスを通じて利用できます。Tableau Prep Conductor のライセンス発行の詳細については、データ管理 のライセンスを参照してください。

専用ノードで Tableau Prep Conductor を有効にすることをお勧めします。詳細情報:

- 新しい Tableau Server をインストールする場合は、「手順 1 (新規インストール): Tableau Server を Tableau Prep Conductor と一緒にインストール」を参照してください。
- Tableau Prep Conductor を既存の Tableau Server インストールに追加する場合は、「手順 1 (既存インストール): Tableau Prep Conductor の有効化」を参照してください。

その他の要件

ご使用の環境が次の追加の要件も満たしていることを確認してください。

ホスト名

- Tableau Server は、ドメイン名サーバー(DNS) または Tableau Server を実行しているコンピューター上のローカル ホスト ファイルを使用し、IP アドレスに対するホスト名を解決できる必要があります。既定では、ホスト ファイルは `/etc/hosts` に保存されています。
- セットアップ プロセスの際に Tableau サービス マネージャーを起動した後はサーバーのホスト名を変更しないでください。たとえば、cloud-init パッケージを使用して仮想マシンを初期化し、その仮想マシンに Tableau Server をインストールする場合にこのようになる可能性があります。
- アンダースコア (_) を含むホスト名は、Tableau Server でサポートしていません。

静的 IP アドレス

Tableau Server をインストールするコンピューターには静的な IPv4 または IPv6 アドレスが必要です。

データベース ドライバー

特定のデータ ソースに接続するには、Tableau Server をインストールするコンピューターに正しいデータベース ドライバーがインストールされている必要があります。詳細については、データベース ドライバーを参照してください。

利用可能なポート

TSM とTableau Server はいずれもアクセスするために利用可能な TCP ポートが必要です。TSM は 8850 番ポートが既定で、Tableau Server ゲートウェイ サービスは 80 番ポートが既定です。Tableau Server をインストールする前に、8850 番ポートと80 番ポートをお使いのシステムで使用しないことを強くお勧めします。これらのポートが利用できないと、TSM とゲートウェイのポートは別のポート番号に動的に再マッピングされる場合がありますが、どのポートに再マッピングされたかを表示するインターフェイスは現時点で存在しません。

Tableau サービス マネージャーのポートを参照してください。

ローカル ファイアウォール構成

Tableau Server をインストールするコンピューターでファイアウォールを実行している場合は、Tableau Server トラフィック用に次の既定のポートを開く必要があります。443 を除くすべてのポート番号を変更できます。

ポート	TCP/UDP	使用...	インストールのタイプ	
			すべて	分散/高可用性
80	TCP	ゲートウェイ	X	
443	TCP	SSL。Tableau Server で SSL を構成している場合、アプリケーション サーバーはこのポートに要求をリダイレクトします。このポートは変更しないでください。	X	
8850	TCP	Tableau Services Manager。	X	
8060	TCP	PostgreSQL データベース。	X	
8061	TCP	PostgreSQL バックアップ検証ポート	X	
8000-9000	TCP	Tableau プロセスの動的マッピング用に既定で予約されているポートの範囲		X
27000-27009	TCP	Tableau Server がライセンス サービスで使用するポートの範囲。この範囲は、ライセンス サービスを実行しているノードで開いている必要があり、また他のノードからアクセス可能である必要があります。既定では、最初のノードはライセンス サービスを実行します。	X	

Tableau サービス マネージャーのポートおよびローカル ファイアウォールの構成を参照してください。

システム ユーザーとグループ

Tableau Server on Linux では、1 人の権限なしユーザーと2 つのグループを使用して適切な操作を行います。Tableau では設定中に既定のアカウントおよびグループが作成されます。または、既存のアカウントを指定することもできます。システム ユーザーとグループ
およびTSM 認可グループを参照してください。

sudo および root アクセス

Tableau Server のすべてのインストール タスクおよび管理タスクは root として実行する必要があります。多くの場合は sudo コマンドを使用して実現できますが、root ユーザーとしてコマンドを直接実行することも可能です。

root アカウントを使用して Tableau Server をインストールするには、インストール時にユーザー アカウントを指定する必要があります。アカウントは TSM の管理に使用されます。`-a` オプションを使用して initialize-tsm スクリプトを実行することによりアカウントを指定します。initialize-tsm スクリプトの出力ヒントを参照してください。

アカウントのパスワード

Tableau Server をインストールして管理するユーザー アカウントは、パスワードで認証できるアカウントである必要があります。これは、このユーザーが認証に別の方法 (パブリック キー認証など) を使用してはならないということを意味します。

Tableau Server のインストールと初期化に使用するアカウントにパスワードがない場合は、次のように `passwd` コマンドで設定できます。

```
sudo passwd $USER
```

ポート アクセス要件

SSH などを使用して Tableau Server をリモートからインストールする場合は、次のポートが開かれていることを確認します。番 ポート。

- `8850`。Tableau サービス マネージャー (TSM) の Web インターフェイスに使用するポートです。このインターフェイスを使用して Tableau Server を構成できます。
- `80`。番 ポート。主要な Tableau Server Web インターフェイスに使用するポートです。

Tableau Server インストーラーはインストール プロセス中 にこれ らのポートを開 こうとしますが、firewalld ファイアウォールに対 してこれ らのポートのみ を開 くことができます。コンピューター が別 のファイアウォールを実 行 している場 合 は、インストール前 にポートを開 く必 要 があります。

仮想 コンテナー環境

バージョン 2021.2 以降、Tableau Server on Linux の特 定 の構 成 をコンテナーで実 行 できるようになりました。サポートされる構 成 の詳 細 については、Tableau Server in a Containerを参照 してください。

Tableau Server on Linux の以 前 のバージョンとサポートされ ていない構 成 はテストされておらず、Docker などの仮 想 コンテナー環 境 ではサポートされ ていません。このような環 境 で Tableau Server on Linux をインストールしても、期 待 どおりには機 能 しません。

パッケージ要 件

Systemd

Tableau Server ではサービスを管 理 する systemdが必 要 です。このパッケージは既 定 で CentOS 7 以 降 および Ubuntu 16 以 降 にインストールされ ます。これ らの配 布 の修 正 バージョンで Tableau Server をテストすると決 定 した場 合 は、次 のコマンドを実 行 して systemd がインストールされ ていることを確 認 できます。

```
whereis systemd
```

systemd がインストールされ ていると、インストールの場 所 が表 示 され ます。たとえば、次 のような出 力 が表 示 され ます。

```
systemd: /usr/lib/systemd /etc/systemd
/usr/share/systemd /usr/share/man/man1/systemd.1.gz
```

systemd をインストールしているものの、Tableau インストーラーで systemd の要 件 確 認 に失 敗 する場 合 には、systemd が実 行 中 でない可 能 性 があります。systemd が実 行 していることを確 認 するには、次 のコマンドを実 行 します。

```
ls /run/systemd
```

ファイルとディレクトリの一覧が出力されます。systemd が実行中の場合には、出力に `system` が含まれます。`system` が出力に含まれていない場合には systemd は実行していません。

ウイルス対策ソフトウェア

Tableau Server で使用されるディレクトリをスキャンするウイルス対策ソフトウェアは、Tableau Server のインストールや現在の使用に干渉する可能性があります。場合によっては、これによってインストール エラーや Tableau Server の起動の問題が発生したり、パフォーマンスへの影響が生じたりすることがあります。Tableau Server を実行しているコンピュータでウイルス対策ソフトウェアを実行する予定がある場合は、ナレッジ ベースの推奨事項に従ってください。

次の手順、Tableau Server のインストールと構成に進んでください。

Tableau Server の最小ハードウェア要件と推奨事項

次の最小ハードウェア要件は、物理ハードウェアおよび仮想マシン (VM) を含め、Tableau Server を実行中のすべてのコンピューターに適用されます。

- **インストールの最小要件**は、Tableau Server をインストールするために、コンピューターに装備されている必要がある最小限のハードウェアです。テストするだけの場合でも、これらの値を備えたサーバーで Tableau Server を実行することはお勧めしません。ライセンスされている使用中の機能によっては、パフォーマンスが低下し、実用的なエクスペリエンスが得られない場合があります。場合によっては、Tableau Server は少なくとも 20 GB のメモリがないと起動しないことがあります。

  プロトタイピングと概念実証 (PoC) テストには、Tableau Cloud を使用することをお勧めします。これにより、適切なサイズのハードウェアで Tableau Server を体験する機会が得られます。

- **本番環境の最小推奨事項**は最小インストール要件よりも厳格であり、ほとんどの本番ノードへのインストールに使用する必要がある最小ハードウェア構成を表しています。コンピューターが最小インストール要件を満たしているが、これらの推奨事項は満たさない場合、セットアップ プログラムによって警告されますが、インストールは続行できます。特定のタスクやバックグラウンダーなどのプロセス専用の特定のノードまたは Prep の場合は、この最小推奨

事項を満たさないサーバーを使用できる場合があります。

ここに記載されている最小推奨事項は一般的なガイダンスとすることを目的としています。お使いの環境での推奨事項は異なる可能性があります。詳細については、「推奨されるベースライン構成」トピックの「ハードウェア推奨事項」セクションを参照してください。

また、Tableau Server は、データベースやアプリケーション サーバーなどのように、リソースを大量に消費するアプリケーションを実行している物理コンピューターや VM インスタンス、あるいは共有リソースを使用している VM インスタンスにはインストールしないでください。

注: コンピューターが最小ハードウェア要件を満たしていながらコア数が 8 未満、システム メモリが 16 GB に満たないコンピューターに Tableau Server をインストールする場合、すべてのプロセスの既定数は設計により各プロセスの 1 つに減らされます。プロセスの詳細については、サーバープロセスの制限を参照してください

Tableau Server on Windows については、「Tableau Server の最小ハードウェア要件と推奨事項」を参照してください。

インストールの最小ハードウェア要件

Tableau Cloud を使用して、概念実証 (PoC) テストまたはプロトタイピングを実行することを強くお勧めします。これにより、十分なリソースを備えたシステムで実行できるようになります。

Tableau Server をインストールまたはアップグレードするコンピューターは、最小ハードウェア インストール要件を満たす必要があります。セットアップ プログラムにより、コンピューターが以下の要件を満たしていないと判断された場合は、Tableau Server をインストールできません。これらの要件を満たすことは、Tableau Server のテストで優れたエクスペリエンスを保証するものではありません。

最小ハードウェア要件

これらの最小要件は、Tableau Server をインストールするためのものです。テストの成功や正常な使用を保証するためのものではありません。本番環境の最小推奨事項については、「本番環境の最小ハードウェア推奨事項」を参照してください。

プロセッサー	*CPU*	*RAM*	*空きディスク領域*
• 64 ビット • SSE4.2 および POPCNT 命令セットをサポートする必要があります • ARM ベースのプロセッサーはサポートされていません	4 コア (8 vCPU)	バージョン 2022.3 以降: • 64 GB - 最初のノード • 16 GB - 追加ノード バージョン 2022.1 以前: • 16 GB - すべてのノード	15 GB

- 空きディスク容量は、Tableau Server セットアッププログラムの解凍後に計算されます。セットアッププログラムでは約 1 GB の容量を使用します。抽出を使用するかどうかなどさまざまな要因に応じて、追加ディスク領域の割り当てが必要になる場合があります。

  Tableau Server の中核部分は、15 GB 以上の空きディスク領域があるディレクトリにインストールする必要があります。十分なスペースがないコンピューターに Tableau Server をインストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行することができません。既定のインストール場所は `/opt` ディレクトリです。RHEL ディストリビューションで Tableau Server のインストール パスを変更することができます。

  抽出を多用する場合は、追加のディスク領域の割り当てが必要になることがあります。インストール時にデータ (抽出) を保存するための別のディレクトリを指定することができます。

- コア数は "物理" コアに基づいています。物理コアは実際のサーバー ハードウェアまたは仮想マシン (VM) 上のコアを表す場合があります。コア数のカウントの目的でのハイパー スレッドは無視されます。

本番環境の最小ハードウェア推奨事項

本番環境において、Tableau Server をインストールまたはアップグレードするコンピューターは、ほとんどの場合、最小ハードウェア推奨事項を満たすか上回る必要があります。推奨事項は一般的なものです。Tableau Server のインストールに必要な実際のシステム要件は、ユーザー数、抽出の数とサイズ、ライセンスされている機能など、多くの要因によって異なります。セットアップ プログラムにより、コンピューターが以下の推奨事項を満たしていないと判断された場合、警告が表示されますが、インストール プロセスは続行できます。詳細については、「本番環境インストールに関するハードウェアの推奨事項」を参照してください。

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
シングル ノード	• 64 ビット (x86_64 チップセット) • SSE4.2 および POPCNT 命令セットをサポートする必要があります • ARM ベースのプロセッサーはサポートされていません	8 コア (16vCPU)、2.0 GHz 以上	バージョン 2022.3 以降: • 128 GB バージョン 2021.4.0 から バージョン 2022.1.x: • 64 GB バージョン 2021.3.x 以前: • 32 GB	50 GB
	Tableau Server インストールに Tableau Prep Conductor を追加する場合、2 番目のノードを追加し、これを Tableau Server Prep Conductor の実行専用にすることを			

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
	お勧めします。このノードには最低 4 コア (8 vCPU)、RAM 16 GB が必要です。			
	推奨されるスワップ メモリ: システム RAM の平方根または 8 GB のいずれか大きい方。			
マルチノードおよびエンタープライズ展開	技術的ガイダンスについては Tableau にお問い合わせください。 ノードは最小ハードウェア推奨事項を満たしているか、それを超える必要があります。ただし、次の場合は除きます。 • バックグラウンダーのインスタンスを最大 2 つ実行している専用バックグラウンダー ノード: 4 コアは許容可能。 • Tableau Prep Conductor の専用ノード: 最低 4 コア (8 vCPU)、RAM 16 GB。 • 独立したゲートウェイの専用ノード: 最低 2 コア (4 vCPU)、8 GB の RAM、および 100 GB の空きディスク領域。			

重要:ディスク領域の要件は TSM を初期化するまでは確認できません。十分な領域がない場合、Tableau Server パッケージのインストールが終了するまでその旨を通知されません。

50 GB のディスク空き領域。少なくとも 15GB が `/opt` ディレクトリに割り当てられ、残りがデータストレージ用として `/var` ディレクトリに割り当てられます。

- 空きディスク容量は、Tableau Server セットアップ プログラムの解凍後に計算されます。セットアップ プログラムでは約 1 GB の容量を使用します。抽出を使用するかどうかなどさまざまな要因に応じて、追加ディスク領域の割り当てが必要になる場合があります。

  Tableau Server の中核部分は、15 GB 以上の空きディスク領域があるディレクトリにインストールする必要があります。十分なスペースがないコンピューターに Tableau Server をインストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行する

ことができません。既定のインストール場所は /opt ディレクトリです。RHEL ディストリビューションで Tableau Server のインストール パスを変更することができます。

抽出を頻繁に使用する予定がある場合は、追加ディスク領域の割り当てが必要になる場合があります。インストール時に、データ(抽出)を保存するための別のディレクトリを指定することができます。

- **外部ファイル ストアに対するネットワーク接続ストレージ領域の要件:** 外部ファイル ストアを使用した Tableau Server を構成する場合は、ネットワーク接続ストレージ専用のストレージ領域の容量を見積もる必要があります。

  ストレージ サイズの見積もり: パブリッシュと抽出更新に必要なストレージの容量を考慮する必要があります。さらに、トピックオプション 2: リポジトリを個別にバックアップするで説明されているように、リポジトリのバックアップを個別に実行するオプションを特に選択しない限り、リポジトリのバックアップ サイズも考慮する必要があります。

  - 抽出:
    - Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。

    - 抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。

  - リポジトリのバックアップ:

    - リポジトリデータのサイズは、<data directory>/pgsql/data/base ディレクトリのサイズを確認することで見積もることができます。

    - リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、workgroup.pg_dump ファイルのサイズを確認します。

- コア数は "物理" コアに基づいています。物理コアは実際のサーバー ハードウェアまたは仮想マシン(VM)上のコアを表す場合があります。コア数のカウントの目的でのハイパー スレッ

ドは無視されます。

- 表示されているRAMはシングルノードインストールでの最小限の推奨です。たとえば、アクティビティ、ユーザー数、バックグラウンドジョブなどに応じて、RAMが増えるとインストールの機能が向上する場合があります。

推奨事項の全リストと最小要件を確認するには、Tableau Serverの最小ハードウェア要件と推奨事項を参照してください。Tableauがスケーラビリティのテストのために内部で使用するハードウェア仕様については、「本番環境インストールに関するハードウェアの推奨事項」を参照してください。

クラウドでのTableau Serverのハードウェア推奨事項については、以下を参照してください。

- AWS Cloud上のTableau Server on Linux管理者ガイドの「AWSインスタンスの種別とサイズを選択する」
- 「Google Cloud Platform上のTableau Server on Linux管理者ガイド」の「Google Compute Engine仮想マシンの種別とサイズを選択する」
- Microsoft Azure上のTableau Server on Linux管理者ガイドの「Microsoft Azure仮想マシンの種別とサイズを選択する」

TSMのインストールと初期化

このトピックでは、Tableau Serverのインストールと Tableau サービス マネージャー (TSM) を初期化するプロセスについて説明します。

重要: 本番環境にはベータバージョンのTableau Serverをインストールしないでください。また、ベータバージョンのバックアップを使用して、本番Tableau Serverインストールを復元しないでください。

前提条件

続行する前に、インストールの前に...のトピックを確認してください。

オプションの初期化パラメーターの確認

TSM をインストールして初期化する前に、初期化操作の一環としてオプションとして設定する場合があるパラメーターを確認することが重要です。`initialize-tsm` は 1 回のみ実行可能なため、組織が必要とするすべてのオプションを使用して実行していることを確認してください。既定以外のシステム ユーザーやグループなど、一部のオプションは初期化操作の一環でのみ構成可能です。フォワードプロキシやダイナミック ポートの設定などの他の構成は、初期化の実行後に手動で設定することができますが、はるかに労力が求められるプロセスです。

オプションのパラメーターの完全なリストについては、initialize-tsm スクリプトの出力ヒントを参照してください。

オプションの初期化パラメーターが使用されている、一般的なシナリオには次のものがあります。

- 転送プロキシ サーバーを使用するために Tableau Server を構成している。インストール時に Tableau Server を構成するには、このトピックで後述するオプション: 一般的な initalize-tsm パラメーターを参照してください。インストール後に Tableau Server を構成することもできます。フォワードプロキシを使用するための Tableau Server on Linux の構成を参照してください。
- 動的ポート マッピングを指定している。既定では、Tableau Server で必要なほとんどのポートは事前定義されたポート範囲から動的に割り当てられます (マッピングされます)。ポート割り当ては、インストール時に各サービスまたはプロセスに対して行われます。ポート マッピングを変更する場合、インストール時にこの構成を行うことをお勧めします。initialize-tsm によるポート リマッピングの制御を参照してください。
- 既定以外のシステム ユーザーまたはグループを指定している。この構成の変更は、初期化の際にのみ実行可能です。システム ユーザーとグループを参照してください。
- 既定の以外のデータディレクトリを指定している。この構成の変更は、初期化の際にのみ実行可能です。データディレクトリを参照してください。

セキュリティのベストプラクティスとして、TSM ポート (既定では `8850`) をインターネットに公開しないでください。

Tableau Server のインストール

ディストリビューションのパッケージ マネージャーを使用して Tableau Server をインストールしてから、スクリプトを実行して TSM を初期化します。スクリプトはインストールされたパッケージに含まれ

ています。

> **重要**: TSM の起動後はサーバーのホスト名を変更できません。たとえば、cloud-init パッケージを使用して仮想マシンを初期化し、その仮想マシンに Tableau Server をインストールする場合にこのようになる可能性があります。

Tableau Server パッケージのインストール

既定では、Tableau Server は `/opt` ディレクトリにインストールされます。RHEL ライクなディストリビューションでは、別のインストール場所を指定できます。

1. Tableau Server をインストールするコンピューターに `sudo` アクセス権を持つユーザーとしてログオンします。

   > **注:** 複雑な問題を避けるため、特殊文字 (たとえば、非ASCII 文字、"+"、"-" など) を含まないユーザー アカウントを使用することをお勧めします。特殊文字を使用すると、環境の構成によっては、Tableau Server の完全インストールの失敗などの問題が発生する可能性があります。

2. `.rpm` または `.deb` インストーラー パッケージを「Tableau Server のダウンロードおよびリリース ノート」ページからダウンロードします。

3. `.rpm` または `.deb` パッケージをコピーしたディレクトリに移動します。

4. パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

- CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。

  - **既定の場所**: 既定の場所 (`/opt/tableau/tableau_server`) にインストールするには次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

    ```
    sudo yum update

    sudo yum install tableau-server-<version>.x86_64.rpm
    ```

  - **既定以外の場所**: 既定以外の場所にインストールするには、`rpm -i` を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

    次のコマンドを実行します。

    ```
    sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
    ```

    注: Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で `yum` の使用が禁止されているため、`rpm -i` を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

  ```
  sudo apt-get update
  sudo apt-get upgrade
  ```

```
sudo apt-get -y install gdebi-core
sudo gdebi -n tableau-server-<version>_amd64.deb
```

TSM の初期化

Tableau を既定の場所にインストールしたかに関わらず、データ(抽出および抽出メタデータ) ストレージには別の場所を指定できます。

1. scripts ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

2. 次のスクリプトを実行して TSM を起動します。

   ```
   sudo ./initialize-tsm --accepteula --activation-service --
   <optional_parameters>
   ```

   initialize-tsm スクリプトで必要な唯一のパラメーターは --accepteula です。このパラメーターは、Tableau Server のエンドユーザー ライセンス契約 (EULA) に同意するために含める必要があります。EULA は次の場所にあります。

   ```
   /opt/tableau/tableau_server/packages/docs.<version_
   code>/Commercial_EULA.txt
   ```

   ただし、スクリプトの実行前に以下の一般的なパラメーターは確認してください。

オプション: 一般的な initalize-tsm パラメーター

initialize-tsm を実行するときに設定できるパラメーター (*フラグ*とも呼ばれる) は数多くあります。一般的なフラグは下記のとおりです。完全なリストを表示するには、コマンド sudo ./initialize-tsm -h を実行するか、initialize-tsm スクリプトの出力ヒントを参照してください。

- Tableau 実行承認 (ATR) サービスを使用して Tableau Server をアクティブ化するには、`--activation-service` オプションを使用します。Tableau Server 2021.4 以降では、このオプションは Microsoft Windows 上の Tableau Server、Linux 上の Tableau Server、コンテナ内の Tableau Server で既定となっています。サーバー ATR は、新規インストールのみに使用でき、アップグレードには使用できません。このオプションは、クラウドベースまたは仮想環境で使用するのが最適ですが、オンラインで Tableau Server のコピーをアクティブ化できるユーザーであれば使用可能です。製品のアクティブ化で ATR を選択すると、永続的な選択となるため、後で取り消すことができません。詳細については、「実行承認 (ATR) サービスを使用した Tableau Server のライセンス認証」を参照してください。サーバー ATR を使用して Tableau Server をライセンス認証しない場合は、`--no-activation-service` オプションを使用します。

- `-a` フラグ: このスクリプトを実行しているユーザーではなく、tsmadmin グループまたは tableau グループに追加されるユーザーを指定します。root アカウントを使用してインストールしている場合は、`-a` フラグを指定する必要があります。

- `--unprivileged-user` フラグ: 別のサービス アカウントを設定します。既定では、`tableau` という名前で新規ユーザーが作成されます。このアカウントは権限が付与されていないサービス アカウントで、ほとんどの Tableau プロセスはこのアカウントで実行されます。`tableau` ユーザー アカウントがコンピューターに既存する場合にのみ、別のユーザーを作成することをお勧めします。

- `-d` フラグ:「データディレクトリ」(Tableau Server が抽出、抽出の情報、その他を格納するディレクトリ) の既定以外の場所を指定します。

  既定では、Tableau Server は、次の場所をそのディレクトリ用に使用します。

  ```
  /var/opt/tableau/tableau_server
  ```

  このフラグを設定して `intialize-tsm` スクリプトを実行すると、指定したディレクトリが作成され、パーミッションが適用されます。既定のディレクトリパスの変更に適用される重要な制限があります。データディレクトリを参照してください。

- インストール時に Tableau Server でフォワードプロキシ ソリューションを構成することをお勧めします。

  これを行うには、--http_proxy フラグや --https_proxy フラグを含め、フォワードプロキシ サーバーを指定します。ポートを含む URL を指定します。たとえば、以下のようにします。

  ```
  --http_proxy=http://proxy.exampe.lan:80/
  ```

  ```
  --https_proxy=http://1.2.3.4:443/
  ```

  https_proxy 環境変数で URL を指定する場合は、注意して http を使用してください。https_proxy 環境変数の値には https プロトコルを指定しないでください。

  Tableau Server でフォワードプロキシを回避するように構成するには、--no_proxy フラグを含めます。また、このプロキシ構成に例外を追加することで、ローカル Tableau Server クラスタ(現在設定している、または今後設定する場合) 内の通信すべてがプロキシ サーバーに送られないようにする必要もあります。例:

  ```
  --no_
  proxy=localhost,127.0.0.1,localaddress,.localdomain.com。
  ```

- TSM および Tableau Server プロセスのポート割り当てを手動で管理する場合は、initialize-tsm.とともに 1 つまたは複数のポート関連のスイッチを使用する必要があることがあります。詳細については、initialize-tsm によるポート リマッピングの制御を参照してください。

- --debug フラグ: トラブルシューティングのため。このオプションは、コマンドの実行により問題のトラブルシューティングが簡単になる場合に、スクリプトに各コマンドを表示します。このオプションを使用すると、画面に広範囲に表示されます。

  注: バージョン 2021.3 以降、このオプションは削除され、スクリプト出力は表示されずにログ /var/tmp/ に記録されます。

3. Tableau Server を構成する前に、ターミナルからログオフしてもう一度ログオンしてください。

再びログオンした時点で新しいセッションが作成され、このときにグループのメンバーシップ変更が反映されます。この新しいセッションは、`initialize-tsm` スクリプトによって追加された環境変数へのアクセス権もあります。

次のステップ

- Tableau Server のライセンス認証と登録

Tableau Server のライセンス認証と登録

Tableau Server を使用するには、アクティブ化して登録する必要があります。

Tableau Server には、サーバーをライセンス認証し、ユーザーに割り当て可能なライセンス レベルの数を指定するプロダクト キーが少なくとも 1 つ必要です。 カスタマー ポータルからプロダクト キーにアクセスできます。

オフラインのコンピューターで Tableau Server をライセンス認証する必要がある場合は、オフラインでの Tableau Server の有効化を参照してください。追加のプロダクト キーをライセンス認証して既存の Tableau Server インストールに容量を追加する必要がある場合は Tableau Server へのライセンスの追加を参照してください。

前提条件

このトピックの手順を実行する前に、Tableau Server のインストールと構成の指示に従って、次の前提条件を満たしてください。

- TSM のインストールと初期化
- TSM はポート 8850 を使用します。ローカル ファイアウォールを実行している場合は、ポート 8850 を開きます。ローカル ファイアウォールの構成を参照してください。

TSM Web インターフェイスの使用

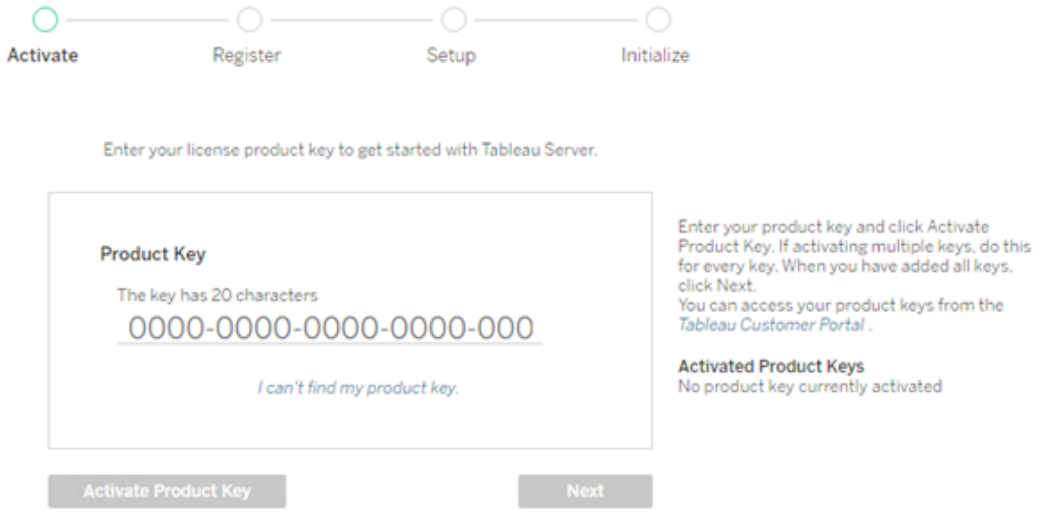
1. Tableau サービス マネージャーの Web UI へのサインイン。

   使用するアカウントには、TSM をインストールするコンピューターに対する管理者権限が必要です。

2. **[ライセンス認証]** ページで、プロダクトキーを入力するか貼り付け、**[Activate Product Key (プロダクト キーのアクティブ化)]** をクリックします。

   > **注:** 2023.3.0 より前のバージョンでは、試用版ライセンスをアクティブ化するオプションが表示されます。このオプションはバージョン 2023.3.0 から廃止されました。Tableau Server を試すためのオプションについては、アカウント担当者にお問い合わせください。

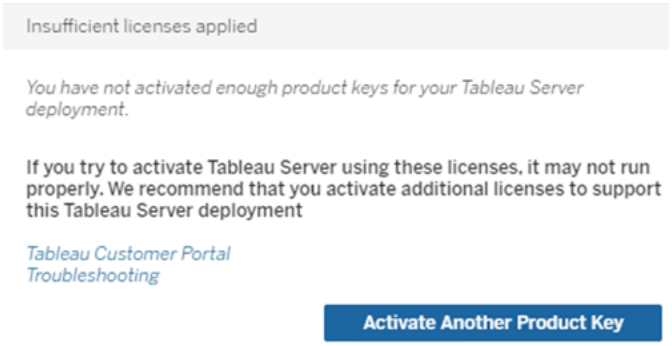
   プロダクトキーをアクティブにすると、**[Activated Product Keys (アクティブ化されたプロダクト キー)]** の下にプロダクトキーが表示されます。



3. 別のプロダクトキーをアクティブにするには、今入力したキーを上書きして新しいプロダクトキーを追加し、**[Activate Product Key (プロダクト キーのアクティブ化)]** をクリックします。プロダクトキーをアクティブにすると、**[Activated Product Keys (アクティブ化されたプロダクト キー)]** の下にプロダクトキーが表示されます。完了するまで、この方法でプロダクトキーを追加していきます。

4. プロダクトキーをアクティブにしたら、**[次へ]** をクリックします。

注: アクティブにしたプロダクトキーに必要な容量がない場合 (コアが足りない場合や、Viewer ロールのプロダクトキーのみの場合など) は、Tableau サービス マネージャーのダイアログ ボックスに**「不十分なライセンスが適用されました」**というメッセージが表示されます。**[Activate Another Product Key (別のプロダクトキーのアクティブ化)]** をクリックして、別のプロダクトキーを追加し、Tableau Server インストールの容量を増やします。



5. **[登録]** ページで情報をフィールドに入力し、**[登録]** をクリックします。

Register with Tableau. All fields are required.

Contact Information

First Name

Last Name

Phone Number

Email

Company Information

Organization

Industry

Company Size

Department

Job Role

Region Information

City

Postal Code

Country/Region

State/Province

TSM CLI の使用

Tableau Server をライセンス認証する

本番環境用に Tableau Server をライセンス認証するには、プロダクトキーが必要です。

プロダクトキーをライセンス認証するには、次のコマンドを実行します。

```
tsm licenses activate -k <product key>
```

2 週間のトライアル版をライセンス認証するには、次のコマンドを実行します。

```
tsm licenses activate -t
```

Tableau をライセンス認証できない場合は、次のようなエラー メッセージが表示されます。

```
License Server not available
```

オフラインでの Tableau Server の有効化を参照してください。

TSM コマンドを実行しようとするとアクセス拒否エラーが表示される場合はどうすればいいですか?

残りのインストールの構成に使用するアカウントは、初期化中に作成された tsmadmin グループのメンバーでなければなりません。tsmadmin グループのユーザー アカウントを表示するには、次のコマンドを実行します。

```
grep tsmadmin /etc/group
```

ユーザー アカウントがグループにない場合、次のコマンドを実行してユーザーを tsmadmin グループに追加します。

```
sudo usermod -G tsmadmin -a <username>
```

Tableau Server の登録

ライセンス認証ができたら、Tableau Server を登録する必要があります。登録するには、登録ファイルを作成して tsm register コマンドを使用してオプションとしてこれを渡します。

1. 次のコマンドを実行して編集可能なテンプレートを生成します。

   ```
   tsm register --template > /path/to/<registration_file>.json
   ```

2. 完成した登録ファイルを作成するために、テンプレートファイルを編集します。

   次の例は、必要な形式で作成された登録ファイルです。

```
{
    "first_name" : "Andrew",
    "last_name" : "Smith",
    "phone" : "311-555-2368",
    "email" : "andrew.smith@mycompany.com",
    "company" : "My Company",
    "industry" : "Finance",
    "company_employees" : "500",
    "department" : "Engineering",
    "title" : "Senior Manager",
    "city" : "Kirkland",
    "state" : "WA",
    "zip" : "98034",
    "country" : "United States",
    "opt_in" : "false",
    "eula" : "accept"
}
```

   注:

   - `opt_in` フィールドは、Tableau からのマーケティングに関するお知らせを受信するための環境設定を示します。多くの国では、既定値は `false` に設定されています。国固有の法律によって、このフィールドの既定値が影響を受ける可能性があります。詳細については、Salesforce Web サイトの「プライバシー情報」を参照してください。

   - `eula` フィールドは、ユーザーがエンドユーザーライセンス契約 (EULA) に同意することを確認するためのものです。既定値は `accept` です。Tableau Server を使用するには、EULA に同意する必要があります。同意しない場合、Tableau Server を実行できません。

3. ファイルの変更を保存したら、`--file` オプションでこのファイルを渡し、Tableau Server を登録します。

```
tsm register --file /path/to/<registration_file>.json
```

例:

```
tsm register --file /usr/share/tableau-reg-file.json
```

Data Management または Advanced Management のプロダクトキーがある場合、追加機能を使用するために、そのキーをアクティブ化する必要があります。これらのライセンスのプロダクトキーは、サーバーで少なくとも 1 つの容量のプロダクトキーをアクティブにした後にのみアクティブにする必要があります。

- Tableau データ管理 のライセンス情報については、データ管理 のライセンスを参照してください。
- Tableau Advanced Management のライセンス情報については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

次の手順

- 初期ノード設定の構成

実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する

サーバー実行承認 (ATR) サービスを使用すると、ライセンス契約を途切れさせることなく、オンプレミス、クラウド、コンテナ、仮想環境などに展開された Tableau Server をアクティブ化できます。これを実現するために、ATR サービスでは、構成可能な期間 (ATR 期間) 中に短期リースが提供されます。短期リースによって、プロダクトキーの有効期限が切れるまでライセンスとデバイスが関連付けられます。ATR は、基盤となるハードウェアが変更された場合でも、最大限のアクティブ化エラーが発生しないように、アクティブ化にかかる容量を追跡します。このオプションは、すべての Tableau Server インストールで推奨されます。

Tableau Server をオンラインでアクティブ化する場合 (これがデフォルト)、Tableau Server はライセンス取得のためにさまざまなインターネットの場所に接続します。詳細については、インターネットとの通信を参照してください。

Tableau Server をオフラインでアクティブ化する場合、ATR サービスを使用してアクティブ化できますが、オフライン アクティブ化での ATR 期間の動作には若干の違いがあります。詳細については、オフライン アクティベーションの ATR 期間を参照してください。

サーバーのライセンス発行を簡素化するには、サーバー実行承認 (ATR) サービスを使用することをお勧めします。Server ATR を使用しないことを選択した場合は、Server ATR の動的管理機能を提供しない従来のライセンス テクノロジを使用することになります。

選択したアクティベーション タイプは、Tableau Server のこのインストールに対して永続的になります。後で変更するには、Tableau Server をバックアップし、削除してからもう一度インストールする必要があります。

Tableau Server ATR の仕組み

ATRサービスが有効になっている場合、Tableau Server は、Tableau がホストする実行承認 (ATR) サービスに定期的に接続し、ライセンスと承認ウィンドウの長さ (ATR 期間 または リース) に基づいて、Tableau の実行が承認されていることを確認します。この通信が成功している限り、Tableau はユーザーに影響することなく実行されます。

Tableau Server がオフラインでアクティブ化され、ATR が有効になっている場合、Tableau Server は ATR サービスに定期的に接続して、Tableau の実行が許可されていることを確認できません。代わりに、Tableau Server は、プロダクト キーが最初にオフラインでアクティブ化されたときからの ATR 期間を内部的に追跡します。

ATR 期間

デフォルトでは、検証のために ATR サービスに正常に接続できるように、Tableau Server のインスタンスには 5 日間の ATR リース (期間) が付与されています。これは、最初の認証後、アクティベーションの有効期限が切れるまで、ネットワーク接続なしで Tableau Server を 5 日間使用できることを意味します。Tableau Server と ATR サービス間の承認チェックは定期的に試され、チェックが成功するたびに、ATR 期間は最大限の長さにリセットされます。

承認チェックの頻度は変化し、ATR 期間によって異なります。

ATR 期間	**承認チェックの頻度**

< 4 時間	15 分ごと
< 24 時間	1 時間ごと
< 7 日間	12 時間ごと
> 7 日間	24 時間ごと

例: ATR 期間が 48 時間の場合、Tableau Server は 12 時間ごとに ATR サービスに接続して承認チェックを実施し、承認チェックが成功するたびに ATR 期間は 48 時間にリセットされます。ATR 期間はカウントダウンを開始し、0 になると次の承認チェックを行います。Tableau Server マシンがシャットダウンされたり、インターネットアクセスがなくなったりすると、Tableau Server は ATR サービスに接続できなくなります。そのようなことが発生した場合、ATR 期間は 48 時間にリセットされず、0 までカウントダウンされ続けます。ATR 期間が 0 に達する前に Tableau Server が起動していないか、ATR サービスと通信できない場合、ライセンスは期限切れになるため、ライセンスをもう一度アクティブ化する必要があります。

注: Tableau Server は可能な限り実行し続けてください。Tableau Server が ATR 期間内に承認チェックを正常に完了できない場合、ATR サービスはライセンスリースを再利用するため、ご利用のライセンスをもう一度アクティブ化する必要があります。

ATR 期間を表示するには、「tsm licenses atr-configuration get」を参照してください。

サーバー ATR 期間を設定または変更する

サーバーの実行承認 (ATR) を使用して Tableau Server をライセンス認証する場合、ATR 期間をデフォルトの 432,000 秒 (5 日) とは異なる設定に変更できます。有効期間が短いテストサーバーまたは仮想マシン (VM) を作成する場合は、ATR 期間を短くすることができます。同様に、長期間利用する予定のサーバーの場合は、ATR の期間を長くすることができます。

新しい VM を頻繁に起動する場合は、ATR 期間を短縮すると、古い VM の容量が解放され、新しい VM で使用できるようになります。一方、ATR 期間を長くすると、更新サイクルは長くなりますが、容量が解放される頻度は低くなります。

ATR 期間を変更するには、`tsm licenses atr-configuration set --duration <value_in_seconds>` コマンドを使用します。詳細については、「tsm licenses atr-configuration set」を参照してください。

ATR 期間 (秒)	最小	最大	既定
Tableau Server (コンテナーを使用する場合)	3,600 (1 時間)	2,593000 (30 日間)	14,400 (4 時間)
Tableau Server (コンテナーを使用しない場合)	3,600 (1 時間)	7,776,000 (90 日間)	432,000 (5 日間)

Tableau Server on Linux

ATR 期間を設定し、アクティブ化サービスを手動で起動するには、次のコマンドを実行します。

1. 最初のノードでターミナル セッションを開き、次のコマンドを使用します。
2. `tsm licenses atr-configuration set --duration <value_in_seconds>`
3. `tsm pending-changes apply`
4. `sudo su -l tableau`
5. `systemctl --user stop activationservice_0`
6. `systemctl --user start activationservice_0`
7. `tsm status -v` コマンドを使用して、アクティブ化サービスが実行中であることを確認します。Tableau Server アクティブ化サービスは "実行中" と表示されます。アクティブ化サービスが起動されない場合は、次のコマンドを実行します。

   ```
   systemctl --user restart activationservice_0
   ```

Tableau Server in a Container

ATR 期間を設定し、アクティブ化サービスを手動で起動するには、次のコマンドを実行します。

1. 最初のノードでターミナル セッションを開き、次のコマンドを使用します。
2. `tsm licenses atr-configuration set --duration <value_in_seconds>`
3. `tsm pending-changes apply`
4. `sudo su -l tableau`
5. `supervisorctl stop activationservice_0`
6. `supervisorctl start activationservice_0`
7. `tsm status -v` コマンドを使用して、アクティブ化サービスが実行中であることを確認します。Tableau Server アクティブ化サービスは "実行中" と表示されます。アクティブ化サービスが起動されない場合は、次のコマンドを実行します。

   ```
   supervisorctl restart activationservice_0
   ```

注: コンテナで作動している Tableau Server の場合、TTL 開始値と TTL 終了値は、Tableau Server が現在使用しているリースを反映します。コンテナのリースは、1 日より短い場合 1 時間ごとに更新されますが、1 日より長い場合は更新に最大 24 時間かかる場合があります。

オフライン アクティベーションの ATR 期間

Tableau Server をオフラインでアクティブ化する場合、ATR 期間は、プロダクト キーがオフラインでアクティブ化された日から 1 年、またはプロダクト キーの更新日のいずれか早い方に設定されます。ATR 期間が設定された日付に達すると、プロダクト キーは非アクティブ化されます。Tableau はオフライン環境では ATR サービスと通信できないため、承認チェックはありません。

`tsm licenses atr-configuration get` コマンドを使用して、オフライン アクティベーションの ATR 期間を表示すると、結果が 0 になります。オフライン アクティベーションの ATR 期間を表示するには、代わりに `ATRDiag -product "Tableau Server"` コマンドを実行します。詳細に

ついては、「ATRDiag.exe コマンド ライン リファレンス」を参照してください。結果の出力では、`TTL End` 日 が ATR 期間の終了日です。

オフライン環境での ATR 期間の更新

オフライン環境では認可のチェックを行えないため、ATR 期間は上記のように固定値に設定されます。サーバーの予定外のダウンタイムを回避するには、インストール時の ATR 期間に注意し、有効期限が切れる前に ATR 期間を更新するように計画してください。これを行う方法は、USL キーを持っているか、USL キー以外のキーを持っているかによって異なります。

- **USL キー:** USL キーを持っている場合は、「USL オフライン ライセンス資格の更新」に記載されている手順に従ってください。
- **USL 以外のキー:** USL 以外のキーを持っている場合は、以下を行います。
  - 既存のライセンス キーを無効にします。「オフラインでの Tableau Server のライセンス認証解除」を参照してください。
  - 新しいキーを有効にします。「オフラインでの Tableau Server の有効化」を参照してください。

サーバー ATR プロダクト キーを別の Tableau Server に移行する

既存の Tableau Server でプロダクト キーを使用するのやめて、新しい Tableau Server のインストールで使用したい場合、サーバー ATR プロダクト キーを移行できます。プロダクト キーは次のような場合に移行することができます。

- 開発環境と実稼働前環境の切り替え。
- Tableau Server をアップグレードされたハードウェアに移行。
- Tableau Server をクラウド インフラストラクチャに移行。
- 仮想マシン (VM) やコンテナなどで Tableau Server を一時的に使用。

別の Tableau Server で再利用するためにプロダクトキーを非アクティブ化する

バージョン 2021.4 以降でのプロダクトキーの非アクティブ化

サーバー ATR を使用してライセンス認証したプロダクトキーを削除することにより、ある Tableau Server インストールで使用したライセンスを別のインストールで再利用できます。プロダクトキーを再利用する方法についての詳細は、「プロダクトキーの認証解除」を参照してください。

バージョン 2021.3 以前でのプロダクトキーの非アクティブ化

ATR サービスを使用して Tableau Server ライセンスをアクティブ化している場合、ライセンスを手動で非アクティブ化することはできません。ライセンスごとに 1 つの本番インストールと 2 つの非本番インストールを行うことができます。ライセンスに未使用のインストールがある場合は、別の Tableau Server で同じライセンスをアクティブ化できます。アクティベーションを使いきっている場合でも、以下の手順で ATR リースが切れた後にライセンスをアクティブ化できます。

1. 既存の Tableau Server がライセンス認証のリースを更新しないように無効化します。これは、次のいずれかの方法で実行できます。
    - 既存の Tableau Server をシャットダウンする。
    - 既存の Tableau Server をアンインストールする。
    - イーサネットケーブルを抜くか、wifi を切断して、インターネットから接続し、既存の Tableau Server の接続を切断する。
2. ATR リースの有効期限が切れた後、別の Tableau Server でライセンスを再利用できます。
3. 新しいコンピューターに Tableau Server をインストールします。

    プロンプトが表示されたら、同じライセンスを使用して Tableau Server を認証します。

オフラインでの Tableau Server の有効化

Tableau Server をインストールする場合は、少なくとも 1 つのプロダクトキーをアクティブ化する必要がありますが、Tableau カスタマーポータルにあるすべての Tableau Server ライセンスをアクティブ化することをお勧めします。これにより、サーバーがアクティブになり、ユーザーに割り当てることが

できるライセンス レベルの数が指定されます。ATR を使用した Tableau Server でオフライン アクティベーションを行う場合、Tableau カスタマー ポータルの [Offline Activation Id (オフライン アクティベーション ID)] フィールドに記載されているプロダクト キーをアクティブ化する必要があります。適切なキーを見つける方法の詳細については、ナレッジ記事「Tableau Server で有効化するための正しいキーを見つける」を参照してください。

また、サーバーに容量を追加したり、新しいプロダクト キーを取得したりする場合は、Tableau Server のインストール後にライセンス認証が必要になる場合もあります。プロダクト キーをお持ちでない場合は、「Tableau Customer Account Center」からプロダクト キーを入手できます。

> **注:** Tableau Server を起動した後にプロダクト キーをライセンス認証するには、Tableau Server を再起動して変更を有効にする必要があります。

ほとんどの場合、Tableau サービス マネージャー (TSM) ライセンス ページを使用して、インストール中やインストール後に Tableau Server から直接キーをライセンス認証できますが、これを実行できない状況もあります。たとえば、コンピューターがインターネットに接続されていない場合や、イントラネット外へのアクセスを制限するファイアウォールがある場合などです。このような場合は、オフラインでライセンス認証を実行する必要があります。
Tableau Server in a Container では、サーバー ATR を使用したライセンス認証のみをサポートします。Server ATR を使用したオフライン アクティベーションは、2023.1 以降でサポートされています。この機能はコンテナで利用できますが、追加の手順と承認が必要です。エアギャップ環境またはオフライン環境のコンテナで Tableau Server を実行する必要がある場合、詳細についてはアカウント担当者にお問い合わせください。

オフラインでのアクティブ化とログインベースのライセンス管理 (LBLM)

Tableau Server バージョン 2023.1.0 以降、サーバーが実行承認 (ATR) サービスを使用するように構成されている場合、LBLM でオフラインでのライセンス認証を行うことができます。ATR サービスを使用するように Tableau Server を構成できるのは、新規インストール時のみです。既存のサーバー インストールをアップグレードするお客様は、バージョン 2023.1.0 以降の新しい Tableau Server のインスタンスをインストールし、既存のインストールのバックアップをその新しいインスタンスに復元する必要があります。

このプロセスの詳細については、「Tableau Server のアップグレードにブルー/グリーン アプローチを使用する」を参照してください。

ATR サービスの詳細については、「実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する」を参照してください。

オフラインでの有効化と更新可能なサブスクリプション ライセンス (USL)

更新可能なサブスクリプション ライセンスのオフラインでの有効化には特別な手順が必要です。詳細については、オフラインまたは切断された環境での USL の有効化を参照してください。

オフラインでのライセンス認証が必要となる場合、次の 2 つのシナリオがあります。

- インストール時にオフラインでライセンス認証する - Tableau Server のインストール時にオフラインでのライセンス認証を実施します。
- インストール後にオフラインでライセンス認証する - サーバーをインストールして実行した後にオフラインでのライセンス認証を実施します。

オフライン アクティブ化の概要

Tableau Server のオフラインでのアクティブ化は、アクティブ化の方法 (実行承認サービス (ATR) を使用する場合としない場合) によって異なります。

ATR *を使用しない*オフラインでのアクティブ化では、次のステップが含まれます。

1. オフライン アクティブ化要求ファイルを生成します。
2. オフラインでのアクティブ化要求ファイルを、インターネットにアクセスできるコンピューターにコピーします。
3. オフラインでのアクティブ化要求ファイルを、Tableau ライセンス認証 Web サイトにアップロードします。
4. 作成されたオフラインでのアクティブ化応答ファイルを Web サイトからダウンロードします。このファイルを使用して Tableau Server をアクティブ化します。

ATR を使用したオフラインでのアクティブ化では、オフラインでのアクティブ化 ID ではなく、プロダクトキー名を使用する必要があります。

オフライン アクティブ化ファイル名の変更

Tableau Server バージョン 2023.1 以降、Tableau ライセンス発行システムでは 2 つの基本的なライセンス発行テクノロジーがサポートされています。管理上の観点から見ると、2 つのシステムの構成の違いは、オフラインでのアクティブ化で生成および使用されるファイルの種類のみです。ライセンス発行テクノロジーは、Tableau Server の初期インストール時に決定され、インストール後に変更することはできません。

ライセンス発行テクノロジーのレガシー バージョンは、現在もサポートされており、FlexNet と呼ばれます。このテクノロジーの最新バージョンは、ATR と呼ばれます。詳細については、「実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する」を参照してください。次の表は、各テクノロジーのファイル名の命名規則について説明したものです。また、一般的なリファレンスも含まれています。

汎用ファイル名	サーバー ATR ファイル名	FlexNet ファイル名
OfflineActivationRequest	OfflineActivationRequestFile_yyyyMMdd.hhmmss.json	TableauOfflineActivationRequest.tlq
OfflineActivationResponse	OfflineActivationLicensingAtrs.zip	activation.tlf

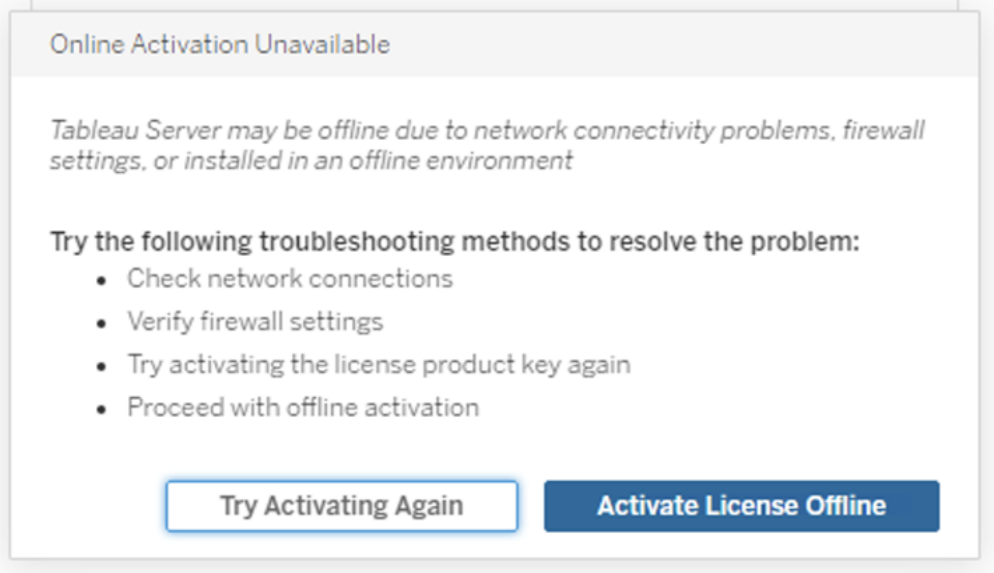
注: このドキュメントは Tableau Server の複数のバージョンに対応しているため、このトピックの残りの部分では一般的なファイル名のリファレンス (OfflineActivationRequest と OfflineActivationResponse) を使用します。Tableau Server のインストールで使用されるライセンス発行テクノロジーは次のステップで生成されるファイルの種類によって特定できます。

TSM Web インターフェイスの使用

TSM のライセンス ページからプロダクト キーを認証しようとした際にオンラインでのライセンス認証を実行できないことを示すダイアログが表示された場合は、キーをオフラインで認証できます。オフライ

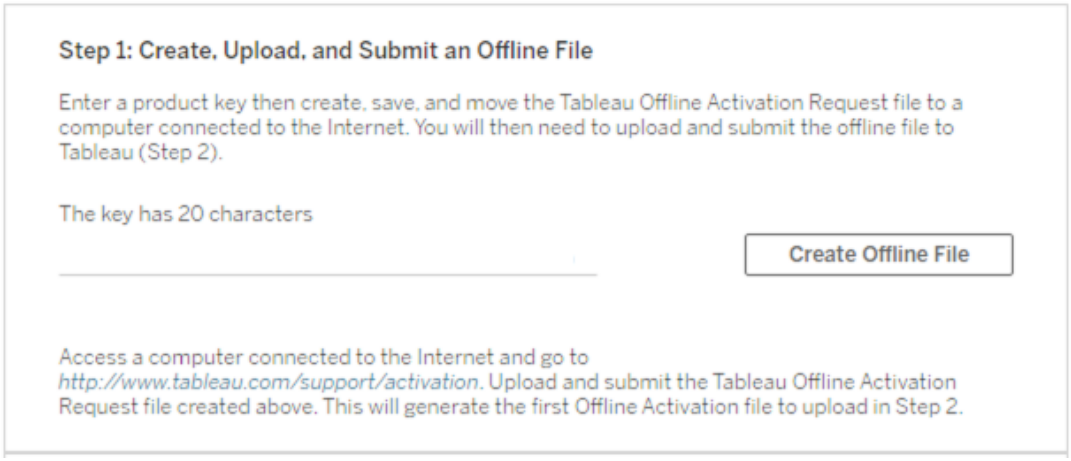
ンでのライセンス認証プロセスは、プロダクトキーごとに 1 回ずつ実施する必要があります。

1. **[オフラインでライセンス認証]** をクリックします。



2. プロダクトキーのオフラインでのライセンス認証要求ファイル (OfflineActivationRequest) を作成します。

   Tableau ライセンス認証 Web サイトにアップロードする OfflineActivationRequest ファイルを作成します。プロダクトキーがフォームに事前に入力されていない場合は、キーを入力して **[オフライン ファイルの作成]** をクリックし、ローカル コンピューター上に OfflineActivationRequest ファイルを生成します。



OfflineActivationRequest ファイルをインターネットにアクセスできるコンピューターにコピーします。ライセンス認証応答ファイルを生成するには、このファイルを Tableau ライセンス認証 Web サイトにアップロードする必要があります。

3. OfflineActivationRequest ファイルをアップロードして送信します。

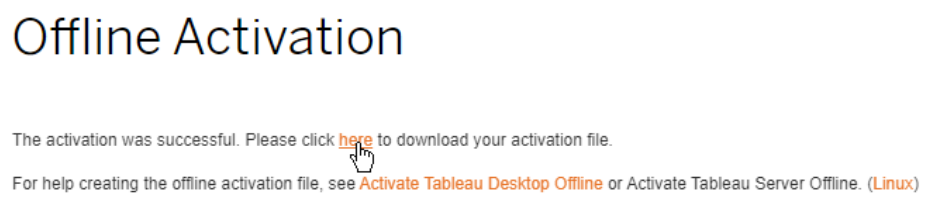
   OfflineActivationRequest ファイルをアップロードして Tableau ライセンス認証 Web サイトに送信します。これにより、ライセンス認証応答ファイル (OfflineActivationResponse) が自動的に生成されます。これを Tableau Server コンピューターにダウンロードしてコピーします。

   a. OfflineActivationRequest ファイルをコピーしたコンピューターで、ブラウザーを開き http://www.tableau.com/ja-jp/support/activation に移動して、Tableau サポートの [ライセンス認証] ページを開きます。

   b. [オフラインでのライセンス認証] ページで、**[ファイルの選択]** をクリックして [OfflineActivationRequest] ファイルを選択します。

   c. **[ライセンス認証ファイルのアップロード]** をクリックして、ファイルを Tableau ライセンス認証 Web サイトに送信します。

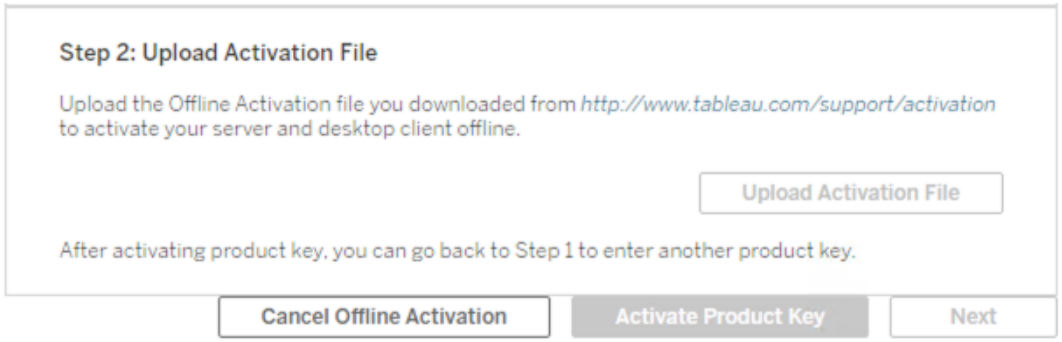
   d. `here` のリンクをクリックして OfflineActivationResponse ファイルをご利用のコンピューターにダウンロードします。



e. OfflineActivationResponse ファイルを、Tableau Server がインストールされているコンピューターにコピーします。

4. OfflineActivationResponse ファイルをアップロードします。

   Tableau Server コンピューターで **[ライセンス認証ファイルのアップロード]** をクリックし、OfflineActivationResponse ファイルを Tableau Server にアップロードします。これを正常に実行すると、**[プロダクト キーのアクティブ化]** ボタンが有効になります。



5. **[プロダクト キーのアクティブ化]** をクリックしてオフラインでのアクティブ化を完了します。

6. (Tableau Server を初めてインストールする場合は、この手順をスキップします)。

   ライセンスの変更を有効にするには、Tableau Server を再起動します。

TSM CLI の使用

ステップ 1.Tableau サービス マネージャーへのログイン

手順を進めるには、Tableau サービス マネージャー (TSM) にログインする必要があります。TSM にログインするには、次のコマンドを実行します。

```
tsm login -u <username>
```

ログインできない場合

残りのインストールの構成に使用するアカウントは、初期化中に作成されたtsmadminグループのメンバーでなければなりません。tsmadminグループのユーザー アカウントを表示するには、次のコマンドを実行します。

```
grep tsmadmin /etc/group
```

ユーザー アカウントがグループにない場合、次のコマンドを実行してユーザーをtsmadminグループに追加します。

```
sudo usermod -G tsmadmin -a <username>
```

ユーザーをtsmadminグループに追加した後に、tsm loginコマンドを実行します。

ステップ2.オフライン アクティブ化要求ファイルを生成する

1. 最初のノードで、ターミナル セッションを開きます。
2. オフラインのライセンス認証ファイルを取得するには、次のコマンドを入力します。

   ```
   tsm licenses get-offline-activation-file -k <product-key> -o
   <target-directory>
   ```

   プロダクト キーは「Tableau カスタマー ポータル」で取得できます。ターゲット ディレクトリが既に存在している必要があります。
3. オフライン アクティブ化ファイル (OfflineActivationRequest) をターゲット ディレクトリからコピーして、インターネットにアクセスできるコンピューターに配置します。

ステップ3.オフライン アクティブ化要求をTableau アクティベーション Web サイトにアップロードする

1. インターネットにアクセスできるコンピューターで、Tableau の「アクティブ化」ページに移動します。
2. 手順を完了し、OfflineActivationRequest ファイルをアップロードします。

これにより、アクティブ化応答ファイル (OfflineActivationResponse) が作成されます。

3. Tableau アクティベーション Web サイトから OfflineActivationResponse ファイルをダウンロードします。

ステップ 4. ライセンスを初期化または認証します

1. OfflineActivationResponse ファイルを Tableau Server コンピューターからアクセスできる場所にコピーします。

2. 次のコマンドを実行します。

   ```
   tsm licenses activate -f <path-and-activation-file>
   ```

   注意: ATR を使用して Tableau Server を有効化する場合、`<path-and-activation-file>` は、パッケージ化された OfflineActivationResponse `.zip` ファイルを指す必要があります。このコマンドを実行する前に、OfflineActivationResponse ファイルを解凍しないでください。

3. (Tableau Server を初めてインストールする場合は、この手順をスキップします)。

   ライセンスの変更を有効にするには、Tableau Server を再起動します。

   ```
   tsm restart
   ```

4. (オプション) すべてのライセンスがアクティブ化されていることを検証するには、次のコマンドを実行します。

   ```
   tsm licenses list
   ```

   上記のステップを完了すると、成功メッセージが表示されます。

   ```
   Activation successful.
   ```

   Tableau Server はライセンス認証されています。さらにサポートが必要な場合は、Tableau 技術サポートにお問い合わせください。

初期ノード設定の構成

このトピックでは、Tableau Server の初期インストール プロセスの一環として重要なサーバー設定を構成する方法について説明します。

前提条件

このトピックの手順を実行する前に、Tableau Server のインストールと構成の指示に従って、次の前提条件を満たしてください。

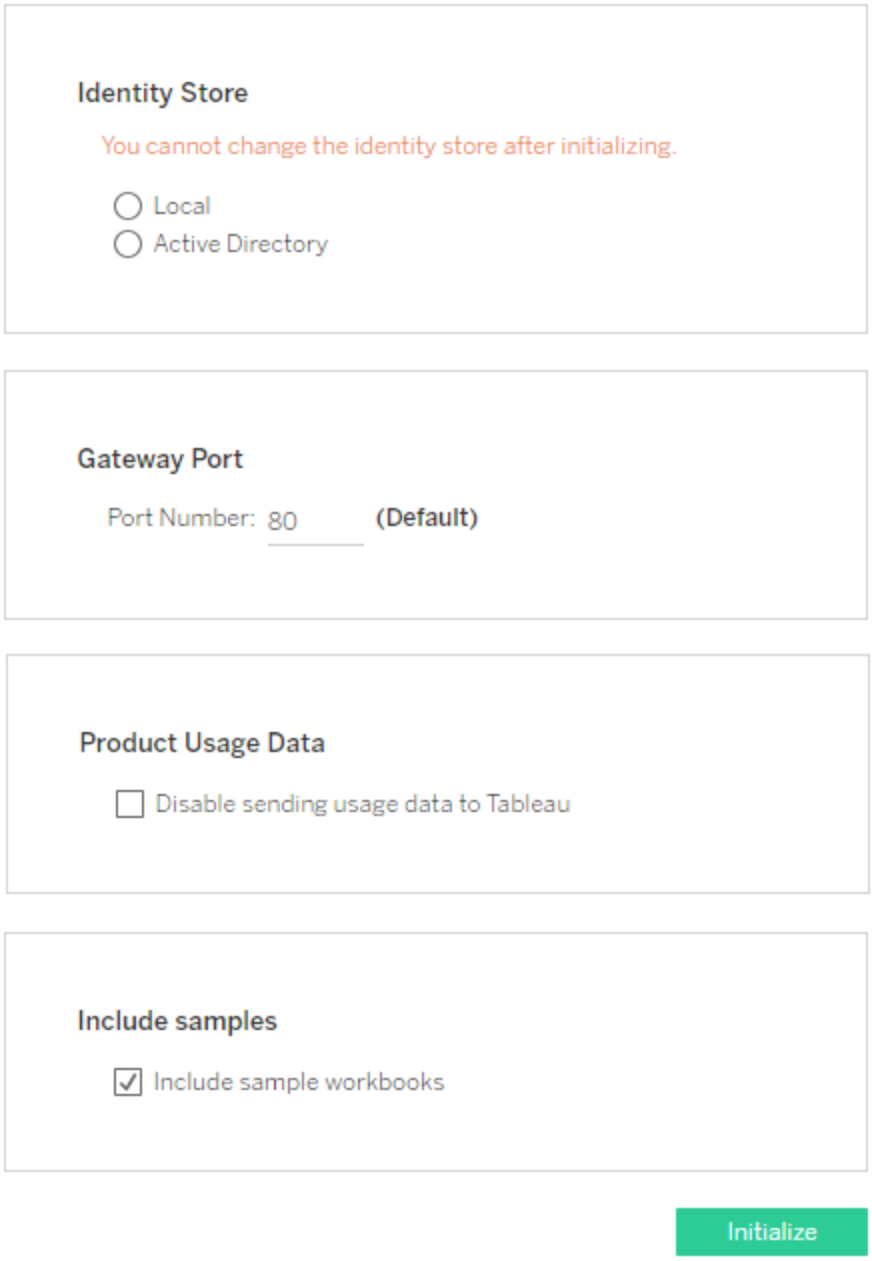
- TSM のインストールと初期化
- Tableau Server のライセンス認証と登録

また、Tableau Server トラフィックのローカル ファイアウォールを構成する必要もあります。ローカル ファイアウォールの構成を参照してください。

TSM Web インターフェイスの使用

Tableau Server のライセンス認証および登録が完了したら、セットアップ ページがインストール プログラムに表示されます。

> **注**: Tableau Server を Active Directory ではない LDAP ディレクトリに接続するよう構成する必要がある場合は、TSM CLI を使用する必要があります。



アイデンティティストア設定

Tableau Server コンピューターのアイデンティティストアを構成しなくてはなりません。アイデンティティストアは Tableau Server アカウントを管理します。アイデンティティストアを構成して、外部ディレクトリ(OpenLDAP や Active Directory など) と同期したり、Tableau Server のアカウントの管理

や保存を行うことができます。シングル サインオン ソリューション (OpenID、SAML、Kerberos など) を使用している場合、アイデンティティストアを構成する前に以下のトピックを確認してください。

- アイデンティティストア
- 認証

重要: アイデンティティストアの設定を構成および適用した後は、変更できません。

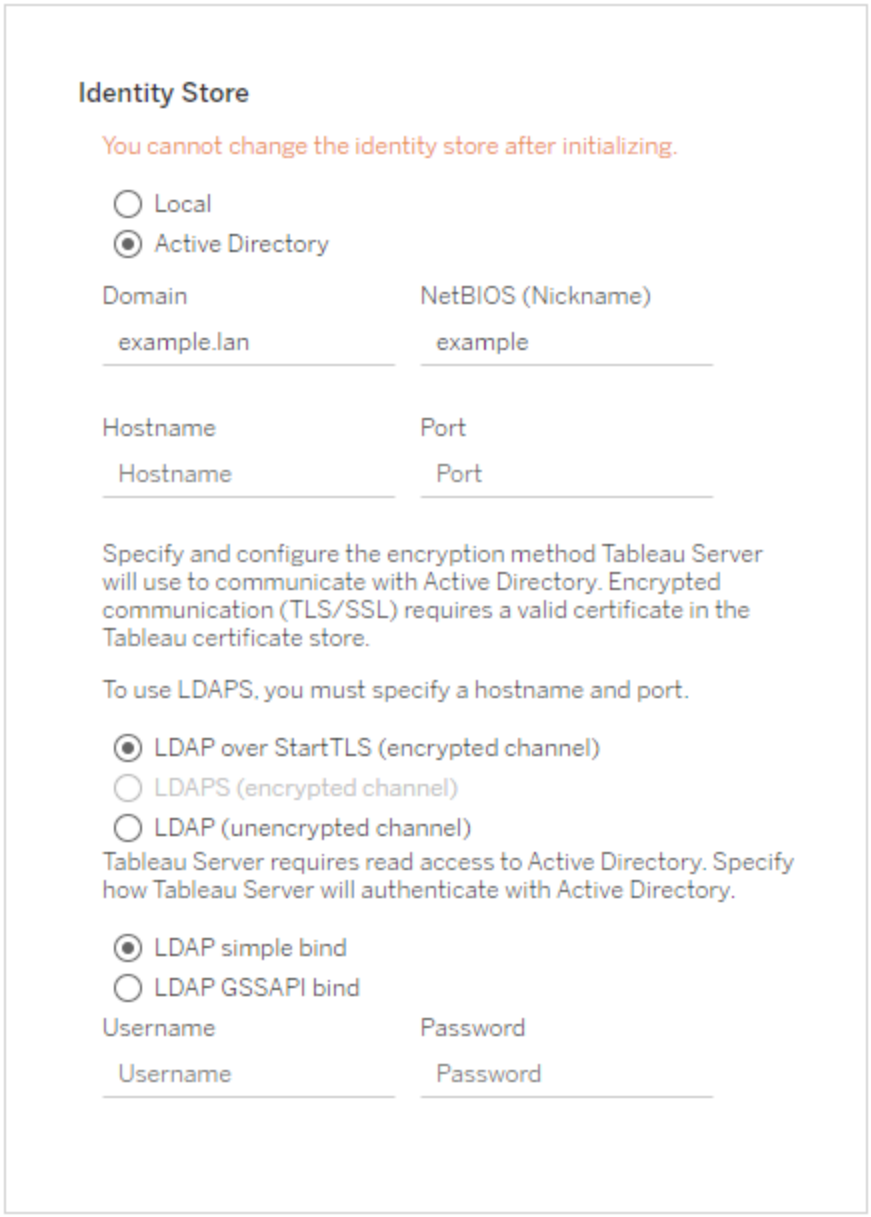
[Active Directory] を選択する場合、Tableau Server はセットアップを実行中の PC から **[ドメイン]** フィールドと **[NetBIOS]** フィールドに入力します。セットアップでこれらの属性が表示されない場合があります。Tableau Server とActive Directory との接続および通信の詳細については、外部 ID ストアを使用した展開におけるユーザー管理を参照してください。

Tableau Server では Active Directory への読み取りアクセスが必要です。

シンプル バインドまたは GSSAPI バインドを使用し、Active Directory によりTableau Server を認証できます。Tableau Server がインストール先のドメイン外部のドメインへのアクセスを必要とする場合、複製バインド アカウントを作成する必要があります。ドメインの信頼のためにバインド アカウントを複製するを参照してください。

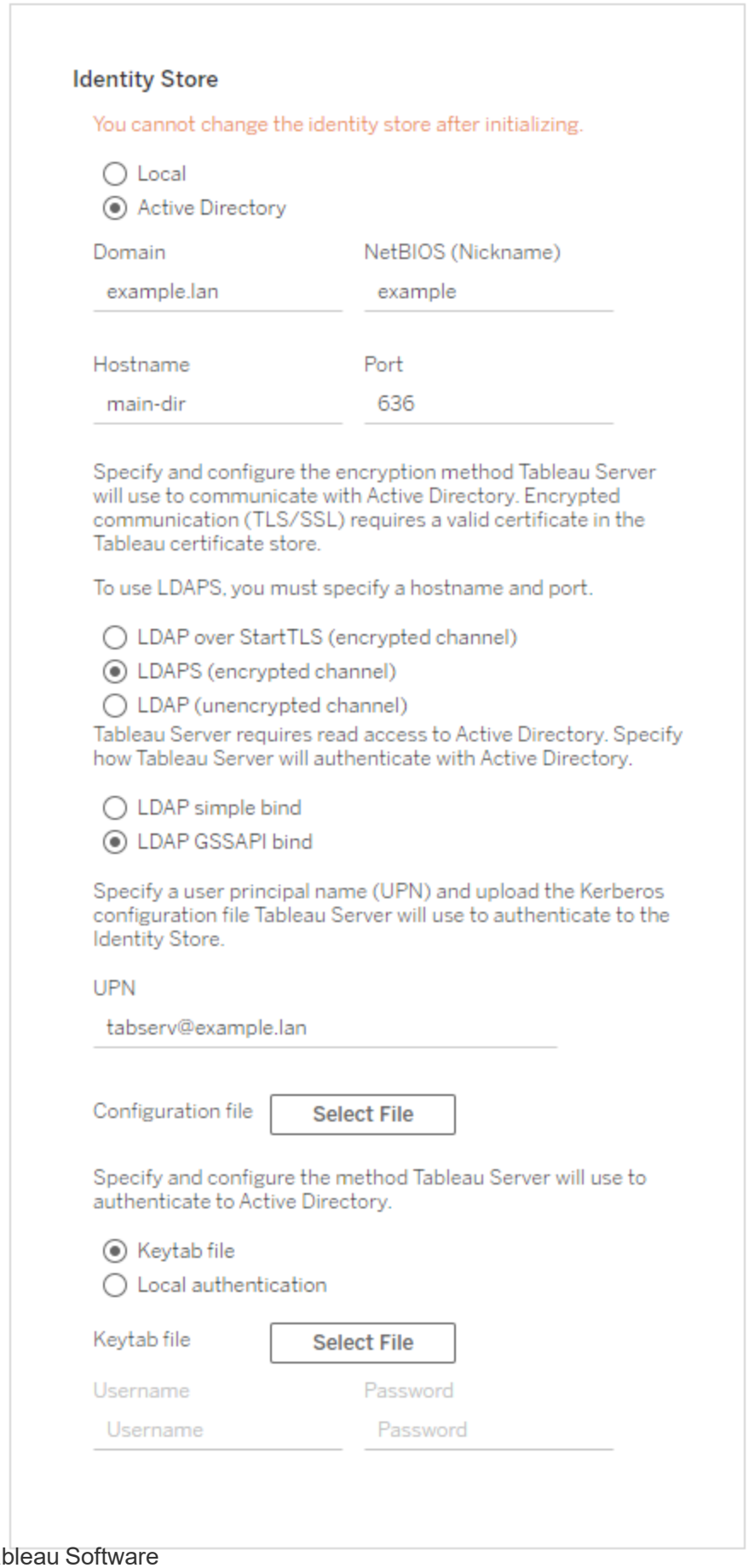
LDAP の暗号化チャネルを設定することをお勧めします。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

LDAP シンプル バインド



Active Directory による認証にシンプル バインドを使用している場合、ドメイン アカウントとパスワードを入力します。

LDAP GSSAPI バインド

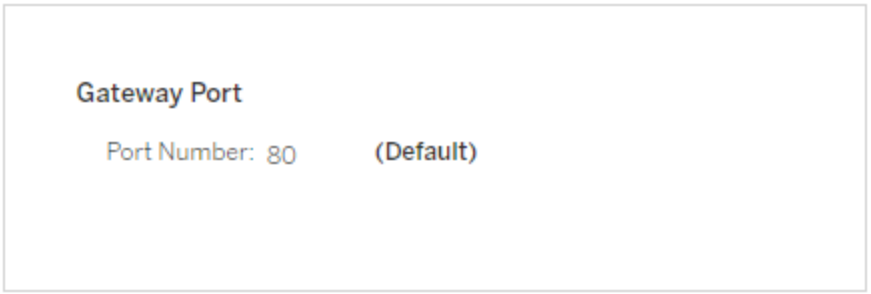

Identity Store
You cannot change the identity store after initializing.
Local
Active Directory
Domain
example.lan
NetBIOS (Nickname)
example
Hostname
main-dir
Port
636
Specify and configure the encryption method Tableau Server will use to communicate with Active Directory. Encrypted communication (TLS/SSL) requires a valid certificate in the Tableau certificate store.
To use LDAPS, you must specify a hostname and port.
LDAP over StartTLS (encrypted channel)
LDAPS (encrypted channel)
LDAP (unencrypted channel)
Tableau Server requires read access to Active Directory. Specify how Tableau Server will authenticate with Active Directory.
LDAP simple bind
LDAP GSSAPI bind
Specify a user principal name (UPN) and upload the Kerberos configuration file Tableau Server will use to authenticate to the Identity Store.
UPN
tabserv@example.lan
Configuration file
Select File
Specify and configure the method Tableau Server will use to authenticate to Active Directory.
Keytab file
Local authentication
Keytab file
Select File
Username
Password
Username
Password

GSSAPI を使用してバインドするには、認証資格情報またはキータブ ファイルを使用できます。キータブ ファイルを使用している場合は、Tableau Server サービス専用のキータブを作成することをお勧めします。Keytab 要件の理解を参照してください。

ゲートウェイ ポート

Tableau Server への Web アクセス (HTTP 経由) の既定のポートは 80 です。Tableau Server を最初にインストールする際、ポート 80 が使用中であるとインストール プログラムによって判断された場合、代替ポート (8000 など) が使用され、ポート番号ボックスに表示されます。

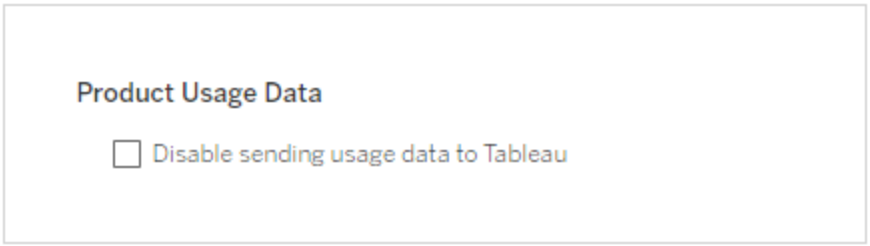
他のネットワークのニーズ用にポート番号の変更が必要となる場合があります。たとえば、Tableau Server ホストの前にハードウェア ファイアウォールやプロキシがある場合、ポート 80 でバックエンド システムを実行するのは望ましくありません。



製品利用状況データ

既定では、Tableau Server が Tableau と利用状況データを共有します。これにより、製品の使用方法についての理解が深まり、全体的な操作性が向上して、Tableau の機能をさらに改善する非常にインテリジェントな機能を構築できるようになります。

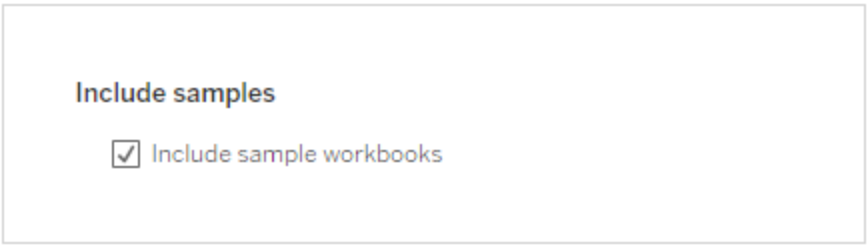
利用状況データを Tableau に送信しない場合は、このオプションをオフにします。



この設定は、インストール後に TSM Web UI の [TSM メンテナンス] タブまたは TSM CLI を使用して変更することもできます。詳細については、サーバーの利用状況データを参照してください。

サンプル ワークブックのインストール

既定では、Tableau Server はサーバーが初期化されるとサンプル ワークブックを既定のサイトにインストールします。

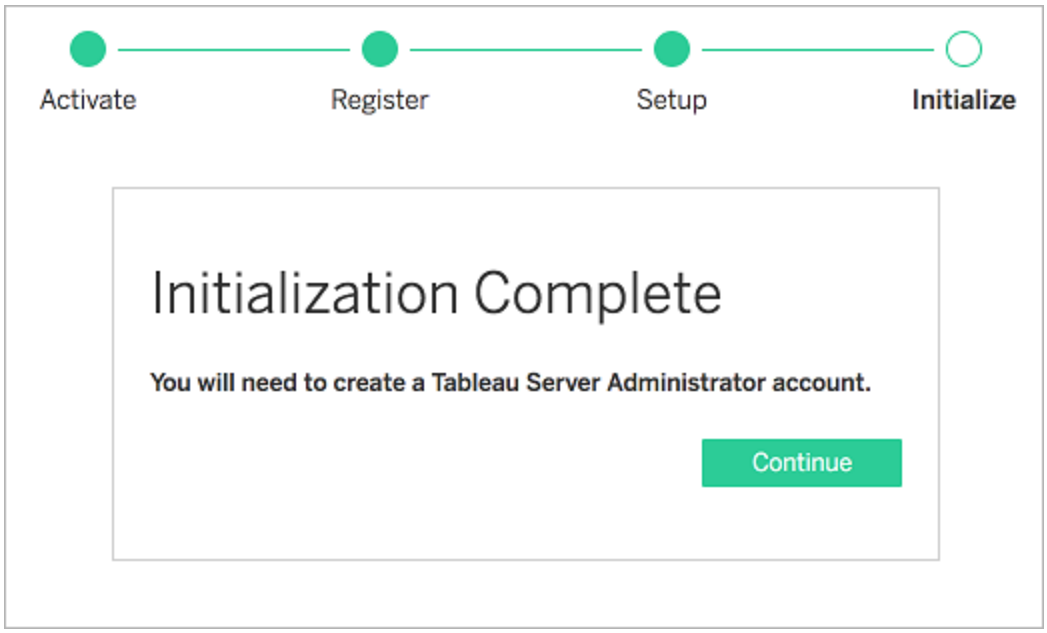


または、publishsamples tabcmd コマンドを使用して、インストール後にサンプルをパブリッシュできます。

インストールの初期化

このページのオプションを構成したら、**[初期化]** をクリックします。

初期化プロセスにはしばらく時間がかかる場合があります。初期化が完了したら、以下のページが表示されます。



TSM CLI の使用

まず、アイデンティティストア、ゲートウェイ設定、およびサンプル ワークブックのインストールを構成します。その後、変更を適用し、LDAP 接続を検証 (オプション) してから、Tableau Server を初期化します

アイデンティティストア設定の構成

Tableau Server コンピューターのアイデンティティストアを構成しなくてはなりません。

> **重要**: アイデンティティストアの設定を構成および適用した後は、変更できません。

identityStore エンティティの json テンプレートを使用して json ファイルを作成します。適切な値でオプションに入力した後、次のコマンドで json ファイルを渡します。

```
tsm settings import -f path-to-file.json。
```

ゲートウェイ設定の構成 (オプション)

ネットワーク構成により、Tableau Server コンピューターのゲートウェイ設定を構成する必要があります。例として、SSL を有効化しているか、リバースプロキシでの Tableau Server へのアクセスを構成している場合、ゲートウェイ設定を構成する必要が生じることがあります。詳細については、gatewaySettings エンティティを参照してください。

gatewaySettings エンティティの json テンプレートを使用して json ファイルを作成します。適切な値でオプションに入力した後、次のコマンドで json ファイルを渡します。

```
tsm settings import -f path-to-json-file.json。
```

製品利用状況データの構成 (オプション)

既定では、Tableau Server が Tableau と利用状況データを共有します。これにより、製品の使用方法についての理解が深まります。また、全体的な操作性が向上して、Tableau の機能をさらに改善する非常にインテリジェントな機能を構築できるようになります。

Tableau では操作性の向上を唯一の目的として行動データと利用状況データのみを収集し、ユーザーの機密データベース値を収集することはありません。また、利用状況データの共有や販売は一切行いません。

製品利用状況データを共有しない場合は、shareProductUsageDataSettings エンティティの json テンプレートを使用して json ファイルを作成し、`false` 値を指定します。次に、以下のコマンドを使用して json ファイルを渡します。

```
tsm settings import -f path-to-json-file.json.
```

この設定は、インストール後に、[TSM メンテナンス] タブまたは TSM CLI を使用して変更することもできます。詳細については、サーバーの利用状況データを参照してください。

サンプル ワークブックのインストールの構成 (オプション)

既定では、Tableau Server はサーバーが初期化されるとサンプル ワークブックを既定のサイトにインストールします。

インストール中にサンプル ワークブックをインストールしたくない場合は、次のコマンドを実行します。

```
tsm configuration set -k install.component.samples -v false
```

publishsamples tabcmd コマンドを使用して、インストール後にサンプルを発行できます。

保留中の構成の変更を適用する

これで初期構成の作成と設定が完了したので、次はこれを適用する必要があります。構成の変更が適用されると、それがコミットされる前に、行った設定が tsm で検証されます。

構成の変更を Tableau Server に適用するには、次のコマンドを実行します。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

このコマンドが完了した時点では TSM プロセスが実行中で、Tableau Server は構成されていますが、まだ起動していません。

LDAP 構成の確認 (オプション)

アイデンティティストアで LDAP が使用される場合、LDAP 接続を確認してから手順を進めることをお勧めします。

これを行うには、サーバーを初期化する前に次のコマンドを実行します。

```
tsm user-identity-store verify-user-mappings -v <user name>
```

```
tsm user-identity-store verify-group-mappings -v <group name>
```

ユーザー名とグループ名は、接続先の LDAP サーバーに実在する有効な名前にする必要があります。LDAP 接続が正しく設定されると、ユーザーまたはグループの属性がシェルに返されます。接続の設定が正しくない場合は、エラーが返されます。

Tableau Server の初期化と起動

- Tableau Server を初期化して開始するには、次のように --start-server オプションを使用します。

  ```
  tsm initialize --start-server --request-timeout 1800
  ```

  これにより、初期化後にサーバーの稼働が開始されるので、時間を節約できます。

- 初期化後に Tableau Server を再構成する場合は、--start-server オプションをオフのままにします。

  ```
  tsm initialize --request-timeout 1800
  ```

  これにより、初期化後にサーバーが停止されます。

Tableau Server を起動します。初期化中に --start-server オプションを使用せずに、Tableau Server の構成が完了すると、次のコマンドを使用してサーバーが起動します。

```
tsm start --request-timeout 900
```

注: Tableau Server のインストールまたは構成中にタイムアウトが発生する場合、タイムアウトの時間を長めに指定する必要があります。詳細については、タイムアウトのためインストールが失敗するを参照してください。

次のステップ

初期化が完了したら、Tableau Server の管理者ユーザー アカウントを作成します。管理者アカウントの追加を参照してください。

構成ファイルの例

この記事では、エンティティgatewaySettings および identityStore を指定した完全な JSON 構成ファイルの例を提供します。さらに、構成キーによりゲートウェイ タイムアウトを 900 秒に設定します。

設定が必要なオプションによって表示される構成ファイルは異なります。

インストール時に、複数の .json 構成ファイルを設定できます。Tableau Server で各ファイルの値を設定するには、各構成ファイルごとに次のコマンドを 1 回ずつ実行します。

```
tsm settings import -f path-to-file.json
```

構成ファイルの設定後、tsm pending-changes apply を実行し、設定したすべての .json ファイルから変更を適用します。

```
{
   "configEntities": {
      "gatewaySettings": {
         "_type": "gatewaySettingsType",
         "port": 80,
         "publicHost": "localhost",
         "publicPort": 80
      },
      "identityStore": {
         "_type": "identityStoreType",
         "type": "local",
         "domain": "example.lan",
         "nickname": "EXAMPLE"
      }
    },
     "configKeys": {
        "gateway.timeout": "900"
     }
}
```

エンティティとキー

上記の例のとおり、configEntities とconfigKeys の 2 クラスの構成パラメーターがあります。

configEntities

アイデンティティストアやゲートウェイ構成など、特定のシナリオにマッピングするエンティティセットを通じて、特定のタイプの構成が行われます。`tsm settings import -f path-to-`

`file.json` コマンドで configEntities のセットを渡すと、TSM が構成を検証します。渡された値が無効である場合、TSM がエラーを表示します。これにより、初期化または実行時に構成が失敗することなく、構成プロセス中に変更を加えることができます。

configEntities ブロックを .json ファイルに含めることでのみ、エンティティを設定できます。

> **重要**: configEntities で参照されているすべてのファイルがローカル コンピューター上に配置されている必要があります。UNC パスは指定しないでください。

configKeys

エンティティは、設定可能な構成値のほんの一部しかカバーしていません。.yml ファイルに格納されているパラメーターには数百個のキーが対応します。Tableau Server は、これらのパラメーターを使用して、すべてのサービスのすべての構成情報を格納します。

tsm configuration コマンドを使用して、個々のキーを設定することができます。しかし、展開時に上記の方法で、JSON ファイル内の他の構成シナリオと共にキーを設定する方が便利です。

configEntities とは異なり、configKeys は検証されません。

> **注**:tsm configuration set のオプションに記載されていないパラメーターを設定することはお勧めしません。

サーバーの利用状況データ

Tableau Server 管理者は、Tableau Server からの利用状況データを Tableau に送信するかどうかを制御できます。既定ではこのオプションは有効になっており、初期インストール時、または Tableau Server のインストール後に、TSM Web UI またはコマンド ラインを使用して、無効にできます。この利用状況データの詳細については、「Tableau 製品の利用状況データ」を参照してください。

製品利用状況データに加えて、製品基礎データも Tableau に送信されます。このデータは、製品利用状況データの送信を無効にしたかどうかに関係なく送信されます。製品基礎データの送信は、個別に無効にすることができます。詳細については、製品基礎データを参照してください。

利用状況データの共有を無効にする

利用状況データの共有は、Tableau Server のインストール時、またはインストール後にいつでも無効にできます。

インストール時に利用状況データの共有を無効にする

Tableau Server のインストール時に Tableau で利用状況データの共有を無効にするには、サーバーの初期設定中に、このオプションをオフにします。詳細については、製品利用状況データを参照してください。

インストール後に利用状況データの共有を無効にする

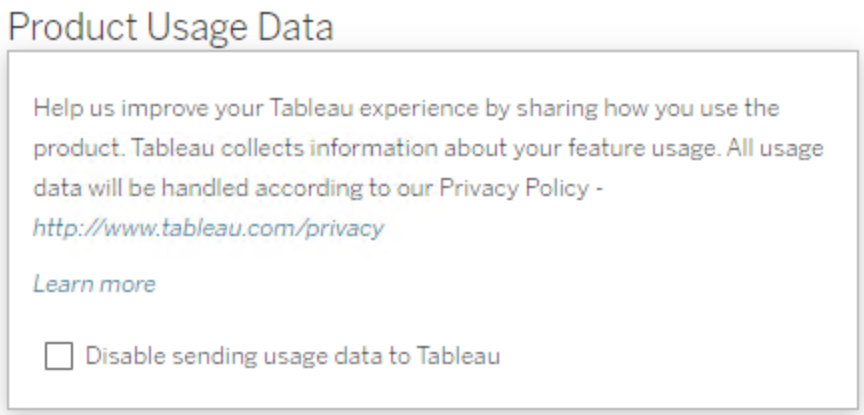
TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[メンテナンス]** タブをクリックします。

3. [サーバーの利用状況データ] の [その他のメンテナンス タスク] で、**[Tableau の機能を改善するために使用状況データを送信する]** をオフにします。



4. 完了したら、**[変更を保留中]** をクリックしてから**[変更を適用して再起動]** をクリックします。

TSM CLI の使用

製品の利用状況データを共有しない場合は、次の tsm configuration コマンドを使用してオプションを無効にします。

```
tsm configuration set -k shareproductusagedata.enabled -v false

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

製品基礎データ

既定では、Tableau 製品から製品利用状況データが Tableau に送信されるため、当社はお客様がソフトウェアをどのように使用しているかを理解でき、さらに、お客様がどこで成功していて、どこで問題に遭遇する可能性があるかのインサイトを得ることで、問題の対処に役立てることができま

す。たとえば、このデータを使用すると、アップグレードが失敗しやすい箇所を把握できるため、それらの問題に対処するよう製品を変更したり、特定のバージョンの Tableau Server に適用されるセキュリティ問題に関する情報がどのユーザーベースで必要とされているかを特定したりすることができます。このデータの送信は、インストール時またはそれ以降に無効にすることができます。これを行う方法の詳細については、Tableau Desktop または Tableau Server の手順を参照してください。

製品利用状況データの送信を無効にしても、特定の製品基礎データは Tableau に送信されます。この製品基礎データには、製品とそのプロセスに関する情報 (実行している製品やプロセス、起動時刻、実行しているオペレーティング システム、ライセンス情報、データを送信したマシンまたはクラスタ(一意の仮名識別子を使用)、製品が製品利用状況データを送信するように構成されているかどうかなど) が含まれます。

製品基礎データの送信をマシン レベルまたはエンタープライズ レベルで無効にするには、**prod.telemetry.tableausoftware.com** に送信されるトラフィックをブロックします。

個々のコンピューターで製品基礎データの共有を無効にする

> **重要:** この手順では、ローカル `hosts` ファイルを変更します。これが何であるかわからない場合は、変更しないでください。この変更は、ファイルに変更を加えることによる影響とファイルの変更方法を理解していて、かつ、安全対策としてファイルのバックアップが作成してある場合にのみ行ってください。

`hosts` ファイルを変更すると、コンピューターのネットワーク動作が変わります。`hosts` ファイルの変更に関する詳細な手順は、Microsoft、Apple、Linux ディストリビューションなどのオペレーティング システム プロバイダーによって提供されます。

1. 既存のホスト ファイルのコピーを作成し、それを Tableau コンピューターではないコンピューターに保存します。これは、変更を元に戻す必要がある場合に備えるためのバックアップです。バックアップ コピーを作成するまで、ファイルの変更を開始しないでください。
2. 次の行を含むようにコンピューターの `hosts` ファイルを変更します。

```
# Stops sending Product Usage to Tableau
(prod.telemetry.tableausoftware.com).
# Learn more here: http:\\tableau.com\derived-data
127.0.0.1    prod.telemetry.tableausoftware.com
```

最初の行と2 番目の行には 3 行目を説明するコメントを入力します。

3 行目の入力は、prod.telemetry.tableausoftware.com (http://prod.telemetry.tableausoftware.com/) へのすべてのトラフィックを内部ホストループバックアドレスに送信することで、ローカル マシンから発信されなくなるようにするものです。データはコンピューターの外部に送信されません。

エンタープライズ レベルで製品基礎データの共有を無効にする

エンタープライズ レベルで製品基礎データの送信を無効にするには、ネットワーク ファイアウォールを変更して、prod.telemetry.tableausoftware.com にアウトバウンド トラフィックが送信されないようにします。

このドメインは、Tableau がプロセスの起動とシャットダウンに関する製品基礎データを受信するために使用されます。また、一般的な製品利用状況データにも使用されます。このドメインへのトラフィックをブロックすると、どちらのタイプのデータも送信されなくなります。

このドメインへのトラフィックは、ポート 80 (製品データ クライアントの初期登録用) とポート 443 (後続のすべてのトラフィック) で発生します。製品データの送信を完全に停止するには、このドメインへのすべてのトラフィックをブロックします。

ネットワーク ファイアウォールの構成方法の詳細については、ベンダーまたは社内の IT 部門に問い合わせてください。Tableau ではこれらの手順を提供することはできません。

管理者アカウントの追加

Tableau Server のライセンス認証の最終ステップは、初期管理者アカウントの追加です。管理者には、ユーザー、グループ、プロジェクトの管理機能など、サーバーに対するすべてのアクセス権が与えられます。

初期管理ユーザーを作成する場合は、サーバーが実行中になっている必要があります。

Tableau Server のアイデンティティストアを構成して LDAP または Active Directory を使用している場合は、指定した初期管理ユーザーをディレクトリのアカウントにする必要があります。通常、初期管理ユーザーは、tsm の実行に使用する Tableau Server コンピューターのユーザー アカウントと区別します。

ただし、Tableau Server のアイデンティティストアを構成して LDAP または Active Directory を使用している場合、および初期管理ユーザーが Tableau Server コンピューターで tsmadmin グループのメンバーになっている場合は、これらのアカウントを同一にすることができます。

前提条件

このトピックの手順を実行する前に、Tableau Server のインストールと構成の指示に従って、次の前提条件を満たしてください。

- TSM のインストールと初期化
- Tableau Server のライセンス認証と登録
- 初期ノード設定の構成

Web UI を使用する

Tableau Server で初期化が完了すると、インストール プログラムにより Tableau Server 管理者の作成ページが表示されます。

- セットアップ中にローカル アイデンティティストアを構成した場合、使用する名前とパスワードを指定します。
- セットアップ中に LDAP または Active Directory のアイデンティティストアを構成した場合は、ディレクトリのメンバーであるユーザー アカウントを指定する必要があります。
- ユーザー名のサフィックスが Tableau Server のプライマリドメインと一致しない限り、username 値に @ マークを含めることはできません。たとえば、Tableau Server がドメイン "myco.com" に接続する場合、"user@example.com@myco.com" というユーザー名は使用できません。

リモートでインストールを進めている場合には、Tableau Server をインストールしている物理 PC で Tableau サービス マネージャーにサインインする必要があります。または、PC にリモートでアクセスし、シェルよりtabcmd initialuser コマンドを実行することもできます。

tabcmd CLI を使用する

Tableau Server 用の初期管理アカウントを作成する必要があります。

- セットアップ中にローカル アイデンティティストアを構成した場合、使用する名前とパスワードを指定します。
- セットアップ中に LDAP または Active Directory のアイデンティティストアを構成した場合は、ディレクトリのメンバーであるユーザー アカウントを指定する必要があります。
- ユーザー名のサフィックスが Tableau Server のプライマリドメインと一致しない限り、`username` 値に @ マークを含めることはできません。たとえば、Tableau Server がドメイン "myco.com" に接続する場合、"user@example.com@myco.com" というユーザー名は使用できません。

最初のユーザーを作成するには、次の `tabcmd` コマンド実行します。

```
tabcmd initialuser --server http://localhost --username '<new-admin-username>'
```

例:

```
tabcmd initialuser --server http://localhost --username 'tableau-admin'
```

HTTP プロトコルを 80 以外のポートで実行している場合は、ホスト名の後にポートを指定します (例: `--server http://localhost:8080`)。

このコマンドを実行した後、管理者パスワードを入力するようシェルから求められます。

次のステップ

Tableau Server 管理者アカウントの作成後は、インストール後のタスクの構成トピックに従って作業し、展開を続行します。

重要: 組み込みの管理ビューを使用する場合は、PostgreSQL ドライバーをインストールする必要があります。サポート対象であるすべてのコネクタのドライバーのリンクおよびインストール手順については、ドライバーのダウンロードページで確認できます。

インストールの検証

Tableau Server がインストールされて適切に稼働していることを検証し、組み込みの管理ビューを確認するには、PostgreSQL ドライバーをインストールする必要があります。

前提条件

このトピックの手順を実行する前に、Tableau Server のインストールと構成の指示に従って、次の前提条件を満たしてください。

- TSM のインストールと初期化
- Tableau Server のライセンス認証と登録
- 初期ノード設定の構成
- 管理者アカウントの追加

PostgreSQL ドライバーのインストールとインストールの検証

インストールを検証するには、次のようにします。

1. 「ドライバーのダウンロード」ページから PostgreSQL ドライバーをダウンロードしてください。
2. .jar ファイルをこのフォルダーにコピーします (手動で作成する必要がある場合があります):
   `/opt/tableau/tableau_driver/jdbc`
3. TSM を再起動します:

   ```
   tsm restart
   ```

4. ドライバーがインストールされていることを検証するには、Tableau Server で管理ビューに移動します。

最初のノードのインストールの既定

既定では、Tableau Server インストーラーは、検出されたハードウェアに基づいて Tableau Server が実行するプロセス インスタンスの数を設定します。既定の構成はシングルサーバー インストール、およびマルチノード インストールの初期サーバーに適用されます。

各プロセスの次のルールに基づいて既定の構成を計算できます。ここで、コアの数は物理プロセッサーの数を示します。

プロセス名	プロセス数
VizQL Server	コアの数を 4 で割った数となり、最大プロセス インスタンスは 4 です。
バックグラウンダー	コアの数が 8 以上の場合は 2 に設定します。
キャッシュ サーバー	コアの数が 8 以上の場合は 2 に設定します。
データ サーバー	コアの数が 8 以上の場合は 2 に設定します。

その他すべてのプロセス タイプの場合、ハードウェアに関係なく、プロセス インスタンスの数は常に 1 つに設定されます。

16 コアを搭載した PC 用の既定の構成例は次のとおりです。

プロセス名	プロセス数
VizQL Server	4
アプリケーション サーバー	1
バックグラウンダー	2
キャッシュ サーバー	2

プロセス名	プロセス数
データサーバー	2
ファイル ストア	1
データエンジン	1

インストールのジャンプスタート

このトピックでは、コマンドラインを使用して基本的な、Tableau Server のクイック スタート インストールを実行するために必要なステップを提示します。ここに説明されている構成の目的とは、サポートされている CentOS または Linux の Ubuntu ディストリビューションを実行中のコンピューター上に Tableau Server をインストールする際の最も迅速でシンプルな方法を提示することです。この手順は、Tableau Server インストールと管理の練習、実際のサーバー展開前のユーザー体験のために使用してください。

重要: 本番環境に Tableau Server を展開する際は、この手順を唯一のリソースとしては使用しないでください。Tableau Server を本番環境に展開するには、Tableau Server のインストールと構成のコンテンツを参照してください。

このトピック内の手順は次の特徴を持つ Tableau Server for Linux のインスタンスをインストールします。

- オペレーティング システム: サポート対象の RHEL ライクな Linux ディストリビューションである Ubuntu(2022 年 7 月の時点で、Tableau は Debian ディストリビューションをサポートしなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください)。
- アイデンティティ ストア: ローカル認証
- ゲートウェイ ポート: 80
- Tableau Server 管理者アカウント: admin

はじめる前に

インストールの前に...のトピックを確認してください。以下の手順は、要件に準拠するハードウェアに、そのトピックで指定された環境要件に従って Linux をインストールしていることを前提とします。

インストールするコンピューターはTableau Server の最小ハードウェア要件と推奨事項に指定されている最小ハードウェア要件を満たさなくてはならないことに注意してください。セットアップ プログラムは、最小ハードウェア要件を満たさないシステムに Tableau Server をインストールしません。

ステップ 1: Tableau Server パッケージをインストールして Tableau Services Manager を起動する

Tableau Server を、お使いのディストリビューションのパッケージ マネージャーでインストールしてから、Tableau Services Manager (TSM) を初期化するスクリプトを実行します。Tableau Services Manager は、Tableau サービスをインストール、校正および管理する管理ツールセットです。

初期化スクリプトはインストール パッケージに含まれています。詳細については、TSM のインストールと初期化を参照してください。

1. Tableau Server をインストールするコンピューターに `sudo` アクセス権を持つユーザーとしてログオンします。
2. Tableau Server のインストール パッケージをコピーしたディレクトリに移動します。
3. パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションで、次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

     ```
     sudo yum update

     sudo yum install tableau-server-<version>.x86_64.rpm
     ```

     注: Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。所属の組織で `yum` の使用が禁止されているため、

> rpm -i を使用しなくてはならない場合は、すべての依存するパッケージを個別にインストールしなくてはなりません。

- Ubuntu で次のコマンドを実行します (<version> の書式設定は「メジャー-マイナー-メンテナンス」(例: 2019-2-5))。

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get -y install gdebi-core
sudo gdebi -n tableau-server-<version>_amd64.deb
```

4. scripts ディレクトリへ移動します。

```
cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
```

5. 次のスクリプトを実行して TSM を起動します。

```
sudo ./initialize-tsm --accepteula
```

6. 初期化が完了した後、ターミナル セッションを閉じます。

```
exit または logout
```

ステップ 2: Tableau Server のライセンス認証と登録

Tableau Server を構成する前に、ライセンス認証して登録する必要があります。詳細については、Tableau Server のライセンス認証と登録を参照してください。

1. Bash セッションを開き、initialize-tsm の実行に以前使用したのと同じアカウントでサインインします。
2. Tableau Server をライセンス認証します。次のコマンドを実行します。

```
tsm licenses activate -k <license_key>
```

3. Tableau Server を登録します。次のコマンドを実行して編集可能なテンプレートを生成します。

   ```
   tsm register --template > /path/to/registration_file.json
   ```

4. テキスト編集プログラムを開き、登録ファイルに入力して保存し、次のコマンドで渡します。

   ```
   tsm register --file /path/to/registration_file.json
   ```

ステップ 3: ローカル アイデンティティ ストアの構成

アイデンティティ ストア設定を構成しなくてはなりません。この手順では、アイデンティティ ストアをローカル認証に設定することでインストールを簡易化します。ワークブックの例は既定でインストールされています。これらの既定設定のカスタマイズについて詳細は、初期ノード設定の構成を参照してください。

- 次のコマンドで構成ファイルを渡します。

  ```
  tsm settings import -f /opt/tableau/tableau_
  server/packages/scripts.<version_code>/config.json
  ```

ステップ 4: インストールの仕上げ

インストールの最終ステップでは、変更を適用し、TSM を初期化および起動して、管理者アカウントを作成します。これらのステップについての詳細は、初期ノード設定の構成に説明されています。

1. これ以前のステップで作成した構成を適用します。次のコマンドを実行します。

   ```
   tsm pending-changes apply
   ```

2. Tableau Server を初期化して起動します。次のコマンドを実行します。

   ```
   tsm initialize --start-server --request-timeout 1800
   ```

3. Tableau Server 管理者アカウントを作成します。次のコマンドを実行します。

```
tabcmd initialuser --server 'localhost:80' --username 'admin' --password '<password>'
```

`'<password>'` には強力なパスワードが入ります。パスワードとその他の引数を一重引用符で囲みます。

作成した `admin` アカウントを使用して Tableau Server admin ウェブページにアクセスします。Tableau Server 管理者エリアへのサインインを参照してください。

ステップ 5: PostgreSQL ドライバーのインストール

Tableau Server がインストールされて適切に稼働していることを検証し、組み込みの管理ビューを確認するには、PostgreSQL ドライバーをインストールする必要があります。

1. 「ドライバーのダウンロード」ページから PostgreSQL ドライバーをダウンロードしてください。
2. .jar ファイルをこのフォルダーにコピーします (手動で作成する必要がある場合があります): `/opt/tableau/tableau_driver/jdbc`
3. TSM を再起動します:

   ```
   tsm restart
   ```

4. ドライバーがインストールされていることを検証するには、Tableau Server で管理ビューに移動します。

ローカル ファイアウォールの構成

このトピックは Tableau Server を実行しているコンピューター上でファイアウォールを構成する方法について説明します。

シングル ノード展開およびマルチノード展開では、ローカルのファイアウォールをオペレーティング システムで有効化して Tableau Server を保護する必要があります。Tableau Server の分散 (マルチノード) インストールでは、ノード間の通信は安全な通信を使用しません。そのため、Tableau Server をホストするコンピューター上でファイアウォールを有効化することをお勧めします。

外部トラフィックからアクセス可能なポートは gateway ポートとtabadmincontroller ポートの 2 つだけになるようファイアウォールを構成することをお勧めします。既定では、これらのポートはそれぞれ 80 および 8850 です。さらに分散環境で実行している場合は、ポート範囲 27000 ~ 27009 を開き、ノード間でライセンス通信が行えるようにする必要があります。

gateway ポートは Tableau Server への HTTP 接続に使用されます。gateway ポートには SSL を使用することをお勧めします。SSL を使用している場合は、ポートを 443 に設定する必要があります (Tableau Server では SSL に対しその他のポートがサポートされていないため)。以下の手順では、gateway ポートでのファイアウォールの構成方法について説明します。Tableau Server ゲートウェイを構成して (初期ノード設定の構成)、ここで設定するポートと一致させます。

下記の例では、RHEL/CentOS ディストリビューションで実行される Tableau Server のシングルおよびマルチノード展開でのファイアウォール構成方法を説明します。例では CentOS の規定ファイアウォールである Firewalld が使用されています。

シングル ノード構成

1. bash shell を開いて次の TSM コマンドを実行し、tabadmincontroller ポートのポート番号を取得します。

   ```
   tsm topology list-ports
   ```

   tabadmincontroller ポートを書き留めます。既定では、このポートは 8850 です。

2. Firewalld を起動します。

   ```
   sudo systemctl start firewalld
   ```

3. 既定のゾーンが public などの高セキュリティゾーンになっていることを確認します。そうでない場合、高セキュリティゾーンに変更することを推奨します。

   ```
   sudo firewall-cmd --get-default-zone
   ```

   ```
   sudo firewall-cmd --set-default-zone=public
   ```

4. gateway ポートとtabadmincontroller ポート用のポートを追加します。以下の例では、既定のポート(80 および 8850) を使用しています。

   ```
   sudo firewall-cmd --permanent --add-port=80/tcp

   sudo firewall-cmd --permanent --add-port=8850/tcp
   ```

5. ファイアウォールをリロードして、設定を検証します。

   ```
   sudo firewall-cmd --reload

   sudo firewall-cmd --list-all
   ```

マルチノード クラスタ構成

マルチノード クラスタでファイアウォールを構成するには、ポートの有効化に加え、ノード同士で通信できるようにする追加の手順が必要です。

はじめる前に

クラスタ内の各 ノードの IP アドレスが必要です。この例では、最初のノードの IP アドレスに <node1IP>、2 台の追加 ノードの IP アドレスに <node2IP> と<node3IP> をプレースホルダとして使用します。

ステップ 1: 最初のノードの構成

1. bash shell を開いて次の TSM コマンドを実行し、tabadmincontroller ポートのポート番号を取得します。

   ```
   tsm topology list-ports
   ```

   tabadmincontroller ポートを書き留めます。既定では、このポートは 8850 です。

2. 次のコマンドを実行して、TSM が動的に選択する可能性のあるポート番号の範囲を調べます。この手順の後半で、この範囲を指定します。ポートの範囲を書き留めます。

   ```
   tsm configuration get -k ports.range.min
   ```

```
tsm configuration get -k ports.range.max
```

標準的な範囲は 8000 から9000 です。

3. Firewalld を起動します。

```
sudo systemctl start firewalld
```

4. 既定のゾーンがpublic などの高セキュリティゾーンになっていることを確認します。そうでない場合、高セキュリティゾーンに変更することを推奨します。

```
firewall-cmd --get-default-zone
```

```
sudo firewall-cmd --set-default-zone=public
```

5. gateway ポートとtabadmincontroller ポート用のポートを追加します。以下の例では、既定のポート(80 および 8850) を使用しています。ノード間でライセンス通信が行えるようにするために、ポート範囲 27000-27010 も追加する必要があります。

```
sudo firewall-cmd --permanent --add-port=80/tcp
```

```
sudo firewall-cmd --permanent --add-port=8850/tcp
```

```
sudo firewall-cmd --permanent --add-port=27000-27010/tcp
```

6. クラスタ内の他ノードからのトラフィックを許可するようにファイアウォールを構成します。ports オプションには、ステップ 2 で書き留めた範囲を指定します。クラスタ内の各追加ノードでコマンドを実行します。例:

```
sudo firewall-cmd --permanent --add-rich-rule='rule family=ipv4
source address=<node2IP>/32 port port=8000-9000 protocol=tcp
accept'
```

```
sudo firewall-cmd --permanent --add-rich-rule='rule family=ipv4
source address=<node3IP>/32 port port=8000-9000 protocol=tcp
accept'
```

7. ファイアウォールをリロードして、設定を検証します。

   ```
   sudo firewall-cmd --reload

   firewall-cmd --list-all
   ```

ステップ 2: 追加ノードの構成

クラスタ内の各ノードは最初のノード、そして他のノードと通信できなくてはなりません。

この手順をクラスタ内の各追加ノードで実行します。この例では、IP アドレス<node2IP> のノードが <node1IP> にある最初のノードと<node3IP> にある第三のノードと通信します。

1. Firewalld を起動します。

   ```
   sudo systemctl start firewalld
   ```

2. 既定のゾーンがpublic などの高セキュリティゾーンになっていることを確認します。そうでない場合、高セキュリティゾーンに変更することを推奨します。

   ```
   firewall-cmd --get-default-zone

   sudo firewall-cmd --set-default-zone=public
   ```

3. クラスタ内の他ノードからgateway とtabadmincontroller へのアクセスを許可するようにファイアウォールを構成します。例:

   ```
   sudo firewall-cmd --permanent --add-rich-rule='rule family=ipv4
   source address=<node1IP>/32 port port=80 protocol=tcp accept'

   sudo firewall-cmd --permanent --add-rich-rule='rule family=ipv4
   source address=<node1IP>/32 port port=8000-9000 protocol=tcp
   accept'

   sudo firewall-cmd --permanent --add-rich-rule='rule family=ipv4
   source address=<node3IP>/32 port port=80 protocol=tcp accept'
   ```

```
sudo firewall-cmd --permanent --add-rich-rule='rule family=ipv4
source address=<node3IP>/32 port port=8000-9000 protocol=tcp
accept'
```

この例では、tabadmincontroller ポート(8850) がポート範囲に含まれているため、コマンド内で明示的に指定されていません。

4. ファイアウォールをリロードして、設定を検証します。

```
sudo firewall-cmd --reload

firewall-cmd --list-all
```

Tableau Server を未接続の (エア ギャップ) 環境にインストールする

外部ネットワークアクセスが全くない未接続の環境でも Tableau Server をインストールできます。そのような未接続の環境は、しばしば「エア ギャップ環境」と呼ばれ、データの漏洩を防止し、ハッキングから保護するために高いレベルのセキュリティが必要とされるときに使用されます。エア ギャップ環境にはインターネット接続がなく、外部ネットワーク アクセスも、外部ワイヤレス アクセスなどもありません。エア ギャップ環境にソフトウェアやデータを出し入れする唯一の方法は、USB メモリや書き込み可能な光学 CD または DVD などのリムーバブル メディアを使用することです。

エア ギャップ環境に Tableau Server をインストールする作業は高度なタスクであり、セキュリティの懸案事項、ベストプラクティス、エア ギャップ環境にソフトウェアをインストールする際の落とし穴について良く知っている IT 管理者の仕事です。

エア ギャップ環境では、以下の Tableau Server の機能が利用できなくなるか、機能性が制限されます。

- **マップ** – Tableau Server は外部でホストされたマップ データを既定で使用します。バージョン 2020.4.0 以降では、オフライン マップを使用するように Tableau Server を構成できます。以前のバージョンの Tableau では、エア ギャップ環境にマップ サーバーをインストールしない限

り、エア ギャップ環境でマップは利用できません。詳細については、エア ギャップ環境でマップを表示するを参照してください。

- **ライセンス発行** – Tableau Server はプロダクト キーをライセンス認証するためにインターネットに接続する必要があります。ただし、プロダクト キーは手動でライセンス認証できます。
- **外部データ**– エア ギャップ環境の外に位置するデータは利用できません。

前提条件

エア ギャップ環境に Tableau Server をインストールするためには、以下の条件が必要とされます。

- Tableau Server に必要とされるインストール パッケージとリソースをダウンロードするために使用できる、インターネットへの制限されたアクセス手段のある信頼できるコンピューター。信頼できるコンピューターは、使用前にウイルスおよびマルウェアのスキャンと駆除がされています。
- エア ギャップ環境にダウンロードされたソフトウェアを転送するために使用できる、信頼できるリムーバブル メディア。信頼できるリムーバブル メディアとは、新しく、今まで未使用で、信頼できるまたは有名な販売元から来たものです。信頼できるリムーバブル メディアは、スキャンされ、ウイルスおよびマルウェアを含まない事が確認されています。
- Tableau Server をインストールする要件を満たすコンピューターとストレージを備えたエア ギャップ環境

Tableau Server を Linux を実行するエア ギャップ環境にインストールする

エア ギャップ環境にあるコンピューター上に Tableau Server をインストールする一番簡単な方法は、コンピューターがエア ギャップ環境に配置される前にインストールを実行することです。これが不可能であれば、必要なパッケージをエア ギャップ環境外の信頼できるコンピューターにダウンロードする必要があります。

1. エアギャップ環境外のインターネットアクセスできる信頼できるコンピューター上で、Tableau Server のインストールパッケージをダウンロードします。

2. 依存パッケージのリストを抽出します。

   Ubuntu の場合:

   `dpkg --field <debfile> Depends` (ここでの`<debfile>` とは、Tableau からダウンロードした .deb パッケージ名です)。

   コマンド入力例:

   ```
   dpkg --field tableau-server-linux-1.deb Depends
   ```

   ### 出力例

   ```
   ca-certificates, fontconfig, net-tools, bash-completion, ca-certificates-java, freeglut3, libegl1-mesa, libfreetype6, libgssapi-krb5-2, libxcomposite1, libxrender1, libxslt1.1, lsb-core
   ```

   RHEL および RHEL ライクな Linux 配布:

   `yum -q deplist <RPM file>` (ここでの`<RPM file>` とは、Tableau からダウンロードした .rpm パッケージ名です)。

   コマンド入力例:

   ```
   yum -q deplist tableau-server-linux_1.rpm
   ```

   ### 出力例

```
package: tableau-server-10400.17.0703.1600.x86_64 10400-
17.0703.1600
dependency: /bin/sh
provider: bash.x86_64 4.2.46-21.el7_3
dependency: bash-completion
provider: bash-completion.noarch 1:2.1-6.el7
dependency: ca-certificates
provider: ca-certificates.noarch 2017.2.14-70.1.el7_3
dependency: fontconfig
provider: fontconfig.x86_64 2.10.95-10.el7
provider: fontconfig.i686 2.10.95-10.el7
dependency: freeglut
provider: freeglut.x86_64 2.8.1-3.el7
provider: freeglut.i686 2.8.1-3.el7
dependency: freetype
provider: freetype.x86_64 2.4.11-12.el7
provider: freetype.i686 2.4.11-12.el7
dependency: krb5-libs
provider: krb5-libs.x86_64 1.14.1-27.el7_3
provider: krb5-libs.i686 1.14.1-27.el7_3
dependency: libXcomposite
provider: libXcomposite.x86_64 0.4.4-4.1.el7
provider: libXcomposite.i686 0.4.4-4.1.el7
dependency: libXrender
provider: libXrender.x86_64 0.9.8-2.1.el7
provider: libXrender.i686 0.9.8-2.1.el7
dependency: libxslt
provider: libxslt.x86_64 1.1.28-5.el7
provider: libxslt.i686 1.1.28-5.el7
dependency: mesa-libEGL
provider: mesa-libEGL.x86_64 11.2.2-2.20160614.el7
provider: mesa-libEGL.i686 11.2.2-2.20160614.el7
dependency: net-tools
provider: net-tools.x86_64 2.0-0.17.20131004git.el7
dependency: redhat-lsb-core
```

```
provider: redhat-lsb-core.x86_64 4.1-27.el7.centos.1
provider: redhat-lsb-core.i686 4.1-27.el7.centos.1
```

3. 各依存パッケージをダウンロードします。

   Ubuntu の場合:

   ```
   apt-get download <package1> <package2>…
   ```

   RHEL および RHEL ライクな Linux 配布:

   ```
   yumdownloader <package1> <package2>…
   ```

4. パッケージをリムーバブル メディアに転送します。

5. エア ギャップ環境のコンピューター上で、Tableau Server のインストールおよび依存パッケージが入っているリムーバブル メディアを挿入し、インストーラーを実行します。

6. インストールの完了後、Tableau Server プロダクト キーをライセンス認証できます。詳細については、Tableau Server をエア ギャップ環境でライセンス認証するを参照してください。

Tableau Server をエア ギャップ環境でライセンス認証する

エア ギャップ状態のコンピューターはインターネットに接続していないため、手動で Tableau Server のライセンス認証プロセスを実行する必要があります。

オフライン アクティブ化の概要

Tableau Server のオフラインでのアクティブ化は、アクティブ化の方法 (実行承認サービス (ATR) を使用する場合としない場合) によって異なります。

ATR *を使用しない*オフラインでのアクティブ化では、次のステップが含まれます。

1. オフライン アクティブ化要求ファイルを生成します。
2. オフラインでのアクティブ化要求ファイルを、インターネットにアクセスできるコンピューターにコピーします。
3. オフラインでのアクティブ化要求ファイルを、Tableau ライセンス認証 Web サイトにアップロードします。
4. 作成されたオフラインでのアクティブ化応答ファイルを Web サイトからダウンロードします。このファイルを使用して Tableau Server をアクティブ化します。

ATR *を使用した*オフラインでのアクティブ化では、オフラインでのアクティブ化 ID ではなく、プロダクトキー名を使用する必要があります。

オフライン アクティブ化ファイル名の変更

Tableau Server バージョン 2023.1 以降、Tableau ライセンス発行システムでは 2 つの基本的なライセンス発行テクノロジーがサポートされています。管理上の観点から見ると、2 つのシステムの構成の違いは、オフラインでのアクティブ化で生成および使用されるファイルの種類のみです。ライセンス発行テクノロジーは、Tableau Server の初期インストール時に決定され、インストール後に変更することはできません。

ライセンス発行テクノロジーのレガシー バージョンは、現在もサポートされており、FlexNet と呼ばれます。このテクノロジーの最新バージョンは、ATR と呼ばれます。詳細については、「実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する」を参照してください。次の表は、各テクノロジーのファイル名の命名規則について説明したものです。また、一般的なリファレンスも含まれています。

汎用ファイル名	サーバー ATR ファイル名	FlexNet ファイル名
OfflineActivationRequest	OfflineActivationRequestFile_yyyyMMdd.hhmmss.json	TableauOfflineActivationRequest.tlq
OfflineActivationResponse	OfflineActivationLicensingAtrs.zip	activation.tlf

注: このドキュメントは Tableau Server の複数のバージョンに対応しているため、このトピックの残りの部分では一般的なファイル名のリファレンス (OfflineActivationRequest と OfflineActivationResponse) を使用します。Tableau Server のインストールで使用されるライセンス発行テクノロジーは次のステップで生成されるファイルの種類によって特定できます。

ステップ 1.Tableau サービス マネージャーへのログイン

- Tableau サービス マネージャー(TSM) にログインするには、次のコマンドを実行します。

```
tsm login -u <username>
```

ログインできない場合

残りのインストールの構成に使用するアカウントは、初期化中に作成された `tsmadmin` グループのメンバーでなければなりません。`tsmadmin` グループのユーザー アカウントを表示するには、次のコマンドを実行します。

```
grep tsmadmin /etc/group
```

ユーザー アカウントがグループにない場合、次のコマンドを実行してユーザーを `tsmadmin` グループに追加します。

```
sudo usermod -G tsmadmin -a <username>
```

ユーザーを `tsmadmin` グループに追加した後に、`tsm login` コマンドを実行します。

ステップ 2.Tableau Server ライセンスの種類を確認する

Tableau Server のライセンスを認証する方法は、実行しているライセンスの種類によって異なります。次のコマンドを実行して、Tableau Server 展開を構成しているライセンスの種類を特定します。

```
tsm configuration get -k serverauthorizationtorun.enabled
```

このコマンドで `true` が返された場合、展開を構成しているライセンスの種類はサーバー ATR です。

このコマンドで `false` が返された場合、展開を構成しているライセンスの種類はレガシーです。

ライセンスの種類に応じたステップ 3 に進みます。

ステップ 3 (サーバー ATR ライセンス タイプ) json コンテンツを生成し、アクティベーション ページにコピーします。

Tableau Server 展開がサーバー ATR ライセンス タイプで構成されている場合は、次の手順に従います。サーバーがレガシー ライセンス タイプで構成されている場合は、スキップして次のセクションに進んでください。

1. エア ギャップ環境にある Tableau Server 上で、TSM を使用してオフラインでのアクティブ化ファイルを取得します。コマンド プロンプトで次を入力します。

   ```
   tsm licenses get-offline-activation-file -k <product-key> -o
   <target-directory>
   ```

   `<target-directory>` は存在していなくてはなりません。プロダクト キーは Tableau カスタマー ポータルで取得できます。

2. JSON ファイルの内容をコピーします。

3. インターネットに接続されたコンピューターから、「Tableau オフライン アクティベーション」Web サイトに移動し、**[オプション B - アクティベーション ファイルから情報を手動で入力する]** を選択し、要求されたフィールドに JSON コンテンツをコピーして、**[送信]** をクリックします。

4. ウェブサイトでは、次が表示されるでしょう。`The activation was successful. Please click here to download your activation file.`

   OfflineActivationResponse ファイルを Tableau からダウンロードし、ステップ 4 に進みます。

ステップ 3 (レガシー ライセンス タイプ) エア ギャップ状態のシステムからアクティベーション要求テンプレートにデータを転記します。

Tableau Server 展開がレガシー ライセンス タイプで構成されている場合は、次の手順に従います。サーバーがサーバー ATR ライセンス タイプで構成されている場合は、前述の手順を実行します。

1. エアギャップ環境にあるTableau Server上で、TSMを使用してオフラインでのアクティブ化ファイルを取得します。コマンドプロンプトで次を入力します。

   ```
   tsm licenses get-offline-activation-file -k <product-key> -o
   <target-directory>
   ```

   <target-directory>は存在していなくてはなりません。プロダクトキーはTableauカスタマーポータルで取得できます。

2. オフラインのリクエストファイル(OfflineActivationRequest)をターゲットディレクトリからインターネットにアクセスできるコンピューターにコピーできる場合は、ステップ5に進みます。

   または、セキュリティ上の理由でファイルを別のコンピューターにコピーできない場合は、ステップ3に進みます。

3. インターネットにアクセス可能な信頼できるコンピューターに`server_linux.tlq`ファイルをダウンロードして、Notepad++などのXMLテキストエディターで開きます。

   ステップ4にリストされている値をオフライン テンプレート(`server_linux.tlq`)にコピーできるように、値をエアギャップ環境のコンピューターから書き留めておく必要があります。

4. 適切な`server_linux.tlq`ファイルの以下のXML要素を、以下に示すエアギャップ環境のコンピューターの同じ要素の値に更新します。

   .tlqファイルにあるすべてのマシン/ハッシュ値は16進値です。有効な文字は0-9とA-Fだけです。すべて大文字を使用してください。

   スペースや改行制御は追加せず、テンプレートにある“X”文字のみを変更してください。ファイルの形式は変更しないでください。

   ライン 2 - <EntitlementId>

   ライン 5 - <ClientVersion>

ライン 5 - <RevisionType> (この値は server_linux.tlq ファイルに存在します。)

ライン 5 - <MachineIdentifier> (この値は server_linux.tlq ファイルに存在します。)

ライン 11 - <Value> (値が存在しない場合、"X" のプレース ホルダーを削除し、<Value></Value> のまま残します。)

ライン 12 - <Value> (値が存在しない場合、"X" のプレース ホルダーを削除し、<Value></Value> のまま残します。)

ライン 13 - <Value> (値が存在しない場合、"X" のプレース ホルダーを削除し、<Value></Value> のまま残します。)

ライン 15 - <SequenceNumber>

ライン 61 - <Hash>

5. オフライン リクエスト ファイル (OfflineActivationRequest) または編集したテンプレート ファイル (server_linux.tlq) を Tableau のオフラインでのアクティブ化 Web サイトにアップロードします。

6. ウェブサイトでは、次が表示されるでしょう。The activation was successful. Please click here to download your activation file.

   OfflineActivationResponse ファイルをダウンロードして、Tableau Server に転送します。ステップ 4 に進みます。

ステップ 4. ライセンスを初期化または認証します

1. 信頼できるリムーバブル メディアを使用して、OfflineActivationResponse ファイルをエアギャップ環境のコンピューターに移動します。

2. 次のコマンドを実行します。

   ```
   tsm licenses activate -f <path-and-activation-file>
   ```

Tableau Server がアクティブ化されていることを示す、"Activation successful."というメッセージが表示されます。

エアギャップ環境でマップを表示する

エアギャップ環境では、インターネットアクセスがないため、既定で Tableau Server 内のマップが利用できなくなります。代わりに、以下の手順を使用して、エアギャップ環境でローカル マッピングを使用するように Tableau を構成できます。

オフライン マッピング用に Tableau Server を構成する

1. 管理者としてコマンドプロンプトを開きます。
2. ローカルで使用可能なオフライン マップを使用するように Tableau を構成します。

```
tsm configuration set -k vizqlserver.force_maps_to_offline -v
true

tsm pending-changes apply
```

Tableau Server のクローンを作成

Tableau Server のバージョン 2022.3 以降では、新しい tsm コマンドを使用すると、Tableau Server 展開の構成とトポロジのコピーを作成し、それを使用して元の展開の正確なレプリカを作成できます。コマンド`tsm settings clone` は、「クローン ペイロード」と呼ぶ一連のファイルを作成します。これには、設定、シークレット、構成、トポロジ (使用ポートを含む)、および外部サービスに関する情報が含まれます。

セキュリティに関する考慮事項

`tsm settings clone` コマンドで生成される「クローン ペイロード」と呼ぶ一連のファイルには、構成の際にサーバー管理者が提供したキーペアと証明書を含むキーストアなどの情報だけでなく、Tableau Server によって生成されたすべてのシークレットも含まれます。クローン ペイロードとその出力場所を安全に保つことは、セキュリティ上非常に重要です。これを容易にするために、以下が行われます。

- 出力ディレクトリが存在しない場合、`tsm settings clone` コマンドを実行しているユーザーにアクセスが制限されたディレクトリが tsm によって作成されます。
- コマンド実行時に出力ディレクトリが存在する場合、コマンドを実行しているユーザーがそのディレクトリを所有していること、およびアクセス許可がそのユーザーのみに制限されていることを tsm が確認します。ディレクトリに必要な権限がない場合は、次のメッセージが表示されます。

  ```
  The output directory '<path/to/directory>' exists, but must be restricted to owner only.
  ```

クローン ペイロードの制約

- 複製される Tableau Server のバージョンは、作成されたバージョンと一致する必要があります。古いバージョンのクローン ペイロードを使用して、新しいバージョンのサーバーをインストールすることはできません。

clone コマンドを使用して Tableau Server のコピーを作成する

Tableau Server の複製を作成するプロセスは複数の手順からなりますが、基本的な手順は次の 2 つです。

1. 複製する Tableau Server インストールからクローン ペイロードを作成します。
2. クローン ペイロードを使用して、Tableau Server のデプロイメントをもう一つインストールします。2 つのインストールは、トポロジを一致させ、元のデプロイメントと一致するようにノードを追加する必要があります。

クローン ペイロードを作成する

`tsm settings clone` コマンドを使用して、Tableau Server の構成とトポロジの設定を含む一連のファイル (クローン ペイロード) を生成します。このコマンドが取る引数は 1 つであり、一連のファイルを保存する出力ディレクトリです。

```
tsm settings clone --output-directory <output_directory>
```

作成したクローン ペイロードを使用すると、Tableau Server の新しいインスタンスを元のインストールと同じ構成とトポロジでインストールできます。

クローン ペイロードを使用して Tableau Server のコピーを作成する

1. 最初のノードに Tableau Server パッケージをインストールします。Tableau Server は初期化しないでください。Tableau Server パッケージのインストール

2. `initialize-tsm` スクリプトを実行し、`tsm settings clone` コマンドで作成したクローン ペイロードへのパスを指定します。

   ```
   sudo /opt/tableau/tableau_
   server/packages/scripts.<version>/initialize-tsm --accepteula -
   -clone-artifact-dir=<path-to-clone-directory>
   ```

   > **注:** Tableau Server は特権のない tableau ユーザーとして実行されるため、管理者は tableau ユーザーがクローンのディレクトリとファイルへの読み取りアクセス権を持っていることを確認する必要があります。ほとんどの場合、クローン ディレクトリのコンテンツに対する読み取り権限と、すべての親ディレクトリに対する実行権限 (トラバーサル用) を「others」に付与する必要があります。

3. (オプション) Tableau Server の元の (複製元の) インストールとノード数が一致するように、追加のノードをインストールします。元のインストールでノードを追加した場合、この手順は必須です。元のインストールと同数のノードのみを追加してインストールします。

   a. 最初のノード上で bootstrap ファイルを生成します。

   b. 追加した各ノードに Tableau Server をインストールし、スクリプト initialize-tsm を実行します。

      追加のノードのインストールと構成

4. 最初のノードで、初期化を完了します。

   ```
   tsm initialize
   ```

コンテナ

クローン機能は、コンテナで Tableau Server を使用する場合、特に外部のリポジトリやストレージで使用する場合に便利です。clone コマンドを使用すると、管理者は Tableau Server 環境をすばやく再構築し、外部サービスを再接続できます。

> **注:** Tableau Server クラスタを再作成する手順は、使用しているコンテナ技術 (docker、docker compose、Kubernetes など) によって異なる場合があります。

以下の情報は、docker を直接使用する場合に固有のものです。最初のノードのコンテナを起動するときに、CLONE_ARTIFACT_DIR 環境変数を使用してクローン ペイロードの場所を指定する必要があります。

例:

```
docker run \
-v <path-to-clone-directory>:/docker/custom-clone-path \
-e CLONE_ARTIFACT_DIR=/docker/custom-clone-path \
-e LICENSE_KEY=<key> \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

Tableau Server のマルチノード展開をコンテナで再作成する

マルチノードの Tableau Server 展開から生成されたクローン ペイロードの場合、最初のコンテナは追加のノードがクラスタに参加するのを待ちます。

クローンを使用するときに追加のノードをクラスタにセットアップすることは、通常の展開でノードを追加することと同じであり、以下が必要です。

- ノード間で bootstrap ファイルを共有するためのボリュームのマウント
- 環境変数 ALWAYS_WRITE_BOOTSTRAP_FILE および BOOTSTRAP_INSTALL の指定

コンテナ展開でマルチノード環境を作成する詳細な手順については、「マルチノードの Tableau Server in a Container」を参照してください。

注: クローンペイロードは、クラスタ内の最初のノードにのみマウントして使用する必要があります。追加のノードにクローンペイロードをマウントする必要はありません。

Tableau Server in a Container

概要

Tableau Server in a Container は、Tableau 初のコンテナーベースのサーバー製品です。Tableau Server in a Container は、Linux Docker コンテナ内で実行されるオールインワンの Tableau Server インスタンスです。つまり、Tableau Server in a Container のイメージは、自己完結型の Tableau Server アプリケーション全体を実行する Docker イメージです。Tableau Server in a Container は、コンテナーベースの環境で Tableau Server の実行をサポートする数多くのステップの中の最初のステップとなります。

Tableau Server in a Container の概念は、Tableau Server が事前にインストールされた VM のようなものと考えると、簡単に理解することができます。イメージは UBI 8 イメージに基づいており、コンテナー内で (`systemd` の代わりに) `supervisord` を実行します。コンテナーが `supervisord` を開始すると、直ちに Tableau Server を初期化して起動しようとします。ここにあるドキュメントの多くは、Docker 環境に合わせて Tableau Server を実行できるように、構成を提供して自動化の活用方法を説明するためのものです。

Tableau Server in a Container のイメージのセットアップツールを使用すると、カスタムパッケージとアーティファクトを含むようにコンテナーイメージを作成してカスタマイズすることができます。このツールの主な機能の 1 つは、コンテナーイメージを構築し、カスタムデータコネクタをインストールすることです。

Tableau Server in a Container に関する制限事項

- Tableau Server in a Container では、サーバー ATR を使用したライセンス認証のみをサポートします。Server ATR を使用したオフライン アクティベーションは、2023.1 以降でサポートされています。この機能はコンテナで利用できますが、追加の手順と承認が必要です。エアギャップ環境またはオフライン環境のコンテナで Tableau Server を実行する必要がある場合、詳細についてはアカウント担当者にお問い合わせください。
- Tableau Server in a Container では現在、Resource Monitoring Tool (RMT) エージェントはサポートされていません。
- Kerberos は、Tableau Server in a Container ではサポートされていません。

概念実証シナリオで Tableau Server in a Container のイメージをすばやくテストするには、Tableau Server in a Container のクイック スタートを参照してください。

Tableau Server in a Container の基本的なワークフロー

Tableau Server in a Container の基本的なワークフローは以下のとおりです。リンクの各ステップの詳細な手順を確認できます。

1. セットアップ ツールを使用して、Tableau Server in a Container のカスタム イメージを作成します。Tableau Server in a Container - セットアップ ツールの使用を参照してください。
2. 作成したイメージを実行して、Tableau Server in a Container を起動して使用します。Tableau Server in a Container - イメージの使用を参照してください。

Tableau Server の機能に関する考慮事項

コンテナーでは、Tableau Server の一部の機能の動作が異なります。このセクションでは、コンテナー環境で特別な考慮事項または異なる考慮事項がある特定の機能について説明します。

Active Directory

AD ドメイン コントローラーを設定する

Tableau Server Web ページおよびサイトのアイデンティティ ストアとして Active Directory を使用することを計画している場合は、追加の考慮事項があります。Linux 環境で実行している Tableau Server は、IP サブネットを調べることによって、通信する AD ドメイン コントローラーを動的に決定

します。コンテナーには任意の IP アドレスを割り当てることができます。この場合、Tableau Server は必ずしもその IP アドレスを使用して適切なドメイン コントローラーを見つけることができるわけではありません。そのため、Tableau Server が通信する特定のドメイン コントローラー/ホスト名を構成する必要がある場合があります。これを行うには、次のステップに従ってください。

1. Tableau Server で使用するドメイン コントローラーを決定し、ホスト名を取得します。
2. 標準の Tableau Server 管理者構成オプションを使用して、構成キー `wgserver.domain.ldap.hostname` をホスト名に設定します。
   - json 構成ファイル `CONFIG_FILE` に値を設定します。
   - TSM 構成コマンドを使用します

     ```
     tsm configuration set -k wgserver.domain.ldap.hostname -v
     <hostname>
     ```

AD 証明書を Tableau Server キーストアにインポートする

既定では、Tableau Server in a container は、シンプル バインドが使用されるたびに StartTLS を介して AD と通信します。したがって、コンテナーをこの構成で実行する場合は、AD サーバー証明書を Tableau Server キーストアにインポートする必要があります。そうしなければ、サーバーの初期化が失敗します。これを行うには、次のステップに従ってください。

1. `pre-init-command` スクリプトを作成します (初期化前のスクリプト セクションを確認してください)。次の行を追加して、AD 証明書を Tableau Server キーストアに追加します。

   ```
   ${INSTALL_DIR}/packages/repository.${SERVICE_
   VERSION}/jre/bin/keytool -importcert -noprompt -alias
   startTlsCert -file <mounted-certificate-path> -storetype JKS -
   storepass changeit -keystore ${DATA_
   DIR}/config/tableauservicesmanagerca.jks
   ```

2. `pre-init-command` スクリプトの `-file` パラメーターに提供されたファイル パスに AD サーバー証明書をマウントします。

または、StartTLS を介して AD と通信する既定の設定を無効にすることもできます。`wgserver.domain.ldap.starttls.enabled` を `false` に設定し、StartTLS を無効にします。ただし、これは推奨されていません。

注: AD を使用した構成エンティティの例については、「LDAP - Active Directory」を参照してください。

展開構成の例

Docker

Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key>
-p 8080:8080
-d <Tableau Server in a Container image ID or tag>
```

自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key> \
-e TABLEAU_USERNAME=<myadmin> \
-e TABLEAU_PASSWORD_FILE=/etc/tableau-admin-secret \
-v <full-path-to-pw-file>:/etc/tableau-admin-secret \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

TSM 専用モード

```
docker run \
-e TSM_ONLY=1 \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

マルチノードの基本的な使用例

最初のノード

オプション 1: サーバー構成 (CONFIG_FILE) でマルチノードトポロジが指定される場合にこれを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-v <full-path-to-config-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

オプション 2: サーバー構成でマルチノードトポロジが指定されていない場合でも、マルチノード展開が必要な場合は、これを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e LICENSE_KEY=<key> -e ALWAYS_WRITE_BOOTSTRAP_FILE=1 \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

追加ノード

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e BOOTSTRAP_INSTALL=1 \
-p 8080:8080 -p 8800-9000:8800-9000 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

データ使用の外部化

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
```

```
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Init コンテナーの基本的な使用例

Init コンテナー

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
-e INIT_CONTAINER=1 \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

コンテナーの実行

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

シングルノードバックアップからの基本的な復元

```
docker run \
-v <full-path-to-backup-file>:/docker/config/backup/backup-
file.tsbak \
-v <full-path-to-config-only-file>:/docker/config/config.json:ro \
-v <full-path-to-identity-store-config-only-
file>:/docker/config/identity-store-config.json \
-e LICENSE_KEY=<key> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Docker の構成

```
version: '3.2'
services:
    tableau-server:
         hostname: localhost
         volumes:
```

```
            - <your-tsm-command-file>:/docker/config/tsm-
commands:ro
            - <your-config-file >:/docker/config/config.json:ro
        ports:
            - "8080:8080"
        image: ${IMAGE_NAME}
        environment:
            - LICENSE_KEY=<license-key>
```

Tableau Server in a Container - セットアップ ツールの使用

概要

Tableau Server in a Container は、Tableau 初のコンテナーベースのサーバー製品です。Tableau Server in a Container は、Linux Docker コンテナ内で実行されるオールインワンの Tableau Server インスタンスです。つまり、Tableau Server in a Container のイメージは、自己完結型の Tableau Server アプリケーション全体を実行する Docker イメージです。Tableau Server in a Container は、コンテナーベースの環境で Tableau Server の実行をサポートする数多くのステップの中の最初のステップとなります。Tableau Server in a Container の概念は、Tableau Server が事前にインストールされた VM のようなものと考えると、簡単に理解することができます。イメージは UBI 8 イメージに基づいており、コンテナー内で (`systemd` の代わりに) `supervisord` を実行します。コンテナーが `supervisord` を開始すると、直ちに Tableau Server を初期化して起動しようとします。ここにあるドキュメントの多くは、Docker 環境に合わせて Tableau Server を実行できるように、構成を提供して自動化の活用方法を説明するためのものです。

Tableau Server in a Container のイメージのセットアップ ツールを使用すると、カスタム パッケージとアーティファクトを含むようにコンテナー イメージを作成してカスタマイズすることができます。このツールの主な機能の 1 つは、コンテナー イメージを構築し、カスタム データ コネクタをインストールすることです。

Tableau Server in a Container に関する制限事項

- Tableau Server in a Container では、サーバー ATR を使用したライセンス認証のみをサポートします。Server ATR を使用したオフライン アクティベーションは、2023.1 以降でサポートされています。この機能はコンテナで利用できますが、追加の手順と承認が必要です。エアギャップ環境またはオフライン環境のコンテナで Tableau Server を実行する必要がある場合、詳細についてはアカウント担当者にお問い合わせください。
- Tableau Server in a Container では現在、Resource Monitoring Tool (RMT) エージェントはサポートされていません。
- Kerberos は、Tableau Server in a Container ではサポートされていません。

Tableau Server in a Container のセットアップ ツール

Tableau Server in a Container のセットアップ ツールで `build-image` を実行すると、Tableau `.rpm` インストーラーと提供されている構成ファイルから Tableau Server in a Container のカスタム イメージが構築されます。

セットアップ ツールでは、Tableau Server インストーラー、ドライバー、およびアーティファクトを入力として受け取り、Docker イメージを作成します。`build-image` ツールが適切に使用されると、新しく生成されたイメージによって目的のアーティファクトがインストールされます。

構築でサポートされているディストリビューション

Tableau Server in a Container の Docker イメージの構築は、RHEL ベースの Linux システム (RHEL、CentOS、または Amazon Linux 2) でのみサポートされます。他の Linux ディストリビューションでの構築は可能ですが、現在はテストが行われておらず、サポートもされていません。macOS でのイメージの構築はサポートされていません。作成されたイメージは、UBI 8 イメージ (バージョン 2022.1 以前は CentOS 7.x) に基づいています。

コンテナー イメージを構築するには、Docker バージョン 18.09 以降がホストにインストールされている必要があります。一般的に、Docker の最新の安定したバージョンを使用することが推奨されています。一部の Linux ディストリビューションでは、ソフトウェア リポジトリに古いバージョンの Docker のみが提供されています。その場合、別のソースから Docker をインストールする必要がある場合があります。バージョン 18.09 より前のバージョンの Docker には、Tableau Server in a Container に必要な機能が含まれていません。

必要なファイルをダウンロードする

セットアップ ツールを使用するには、ツールおよび互換性のあるサーバー インストーラー .rpm ファイルの両方をダウンロードする必要があります。インストーラー ファイルはバージョン 2021.2.0 以降である必要があります。どちらのファイルも Tableau Server ページからダウンロードできます。

1. サーバー インストーラー ファイル tableau-server-<version>.rpm バージョン 2021.2.0 以降をダウンロードします。
2. Server in a Container のセットアップ ツール tableau-server-container-setup-tool-<version>.tar.gz.をダウンロードします。

インストール

Tableau Server in a Container のセットアップ ツールは、tarball として提供されます。圧縮ファイルの内容は抽出する必要があります。次に示すのは、Tableau Server in a Container のセットアップ ツールのアーカイブが現在のディレクトリにあると想定した例です。

```
tar -xzf tableau-server-container-setup-tool-<VERSION>.tar.gz
```

これにより、新しいディレクトリtableau-server-container-setup-tool-<VERSION> がツールの実行に使用するbuild-image スクリプトとともにが作成されます。

登録フォームを完成させる

エンドユーザー ライセンス契約に従い、登録ファイルを編集して、Tableau Server を登録するために必要な一意の識別情報を入力します。ファイル reg-info.json は、一意に識別可能な必須登録情報のテンプレートとして使用でき、Tableau Server in a Container のセットアップ ツールの最上位ディレクトリにあります。このファイルは、イメージで実行されている Tableau Server インスタンスを登録するために使用します。正確な情報を提供することで、登録プロセスを適切に完了させることができます。

注:

- `eula` フィールドの値には「accept」が事前に入力されており、エンドユーザー ライセンス契約 (EULA) に同意していることを示します。EULA は、ビルドツールの EULA ディレクトリで表示できます。EULA で概説されているように、Tableau Server をアクティブ化するときは、一意に識別可能なユーザー登録を送信する必要があります。登録ファイルの編集が完了すると、他のフィールドに一意の情報を反映する値が入力されます。このファイルは、イメージで実行されている Tableau Server インスタンスを登録するために使用します。正確な情報を入力して登録プロセスを完了させ、登録を送信すると、ライセンス付与の要件が満たされます。

  > **注:** Tableau Server を使用するには、EULA に同意する必要があります。EULA に同意しない場合、Tableau Server を実行できません。

- `opt_in` フィールド値の値は、Salesforce からのマーケティング資料の送付を希望しないことを示すために、多くの国ではデフォルトで「false」になっています。

  国別の法律によって、このフィールドの既定値が変わる場合があります。詳細については、Salesforce Web サイトの「プライバシー情報」を参照してください。

編集前の登録ファイル テンプレート `reg-info.json`:

```
{
"zip" : "[value]",
"country" : "[value]",
"city" : "[value]",
"last_name" : "[value]",
"industry" : "[value]",
"eula" : "[value]",
"title" : "[value]",
"company_employees" : "[value]",
"phone" : "[value]",
"company" : "[value]",
"state" : "[value]",
"opt_in" : "[value]",
"department" : "[value]",
"first_name" : "[value]",
```

```
"email" : "[value]"
}
```

ツールの使用方法

セットアップ ツールには、build-image という実行可能なスクリプトがあります。これを -h オプションを使用して実行すると、以下のヘルプが表示されます。

```
./src/build-image -h
Usage: build-image --accepteula -i [INPUT_IMAGE_NAME] -o [OUTPUT_
IMAGE_NAME] [optional arguments]
Creates new Tableau Server image with pre-installed database
drivers, configuration, and instructions.
REQUIRED

--accepteula              Indicate that you have accepted the End
User License Agreement (EULA).

The EULA is available in the EULA directory of this tool.

-i installer              Path to the Tableau Server installer.

OPTIONAL

-o output name            Tag of target generated Tableau Server
image.

-e environment file       Optional environment file to configuration
overrides to docker image at build time.

-v setup tool version     Prints the Setup tool version.


-b base image version     Prints the default base image URL.
```

基本的な使用例:

```
./build-image --accepteula -i tableau-server-image.rpm
```

アーティファクトやセットアップ スクリプトを提供しなくてもコマンドは実行できますが、追加のリソースは元の Tableau Server in a Container のイメージにコピーされず、インストールもされないため、何も実行されません。

オーケストレーション

Tableau では、Linux で実行されている Tableau Server コンテナーのドキュメントとサポートのみが提供されます。Tableau では、Kubernetes や Docker Swarm などのコンテナー オーケストレーション システムのドキュメントやサポートは提供されていません。ただし、Kubernetes を使用して Tableau Server in a Container を展開することができます。Kubernetes で Tableau コンテナーを展開するためのリソースとガイダンスについては、コミュニティでサポートされている GitHub プロジェクト https://github.com/tableau/tableau-server-in-kubernetes を参照してください。

イメージをカスタマイズする

構築時に環境変数を設定する

イメージをカスタマイズできる環境変数のサブセットは、イメージの構築時にのみ設定できます。これには、コンテナー内でルート権限を必要とするユーザー、グループ、およびその他のプロパティが含まれます。既定では、ルート アクションは実行時に使用できません。さらに、構築時に特定の環境変数をイメージにベイク処理しておくと、イメージを実行するたびに設定する必要がないため便利です。これらの環境変数はすべて、環境ファイルを build-image スクリプトに渡すことにより設定できます。

環境ファイル

環境ファイルは、-e 引数を使用して build-image スクリプトに渡すことができます。ファイルは次の形式に準拠している必要があります。

```
<KEY>=<VALUE>
<KEY>=<VALUE>
```

使用例:

正しい形式で環境ファイルを作成します。

```
UNPRIVILEGED_TABLEAU_UID=1012
UNPRIVILEGED_TABLEAU_GID=1020
TABLEAU_USERNAME=myuser
TABLEAU_PASSWORD=pw
```

ファイルをイメージ ビルダーに渡す

```
./build-image --accepteula -i tableau-server-2020-3.x86_64.rpm -e
<path-to-env-file>
```

環境変数

環境変数はすべて環境ファイルで設定できます。完全なリストを確認するには、[環境変数] セクションを参照してください。

構築時の環境変数は、次のスクリプトを実行してイメージを構築する場合にのみ設定できます。

環境名	既定	説明
BASE_IMAGE_URL	ビルド ツール コマンドbuild-image -b を使用します。	build-image ツールとDockerfile で指定された既定のイメージは、公式にサポートされている唯一のベース イメージです。このパラメーターを使用して、この特定のベース イメージのコピーをカスタム Docker イメージ リポジトリからプルするか、カスタム ベース イメージを定義できます。カスタム定義のベース イメージを使用する場合 (詳細については、「カスタム ベース イメージの定義」を参照) は、UBI 8 に基づいており、Tableau Server を適切に実行するために必要なリソースが含まれていることを確認する必要があります。
PRIVILEGED_TABLEAU_GID	997	特権 Tableau グループの GID。
UNPRIVILEGED_TABLEAU_GID	998	特権のない Tableau グループの GID。
UNPRIVILEGED_TABLEAU_UID	999	Tableau プロセスを実行するユーザーの UID (シングルユーザー展開)。

環境名	既定	説明
UNPRIVILEGED_USERNAME	tableau	特権のないユーザーの文字列名。
UNPRIVILEGED_GROUP_NAME	tableau	特権のないグループの文字列名。
PRIVILEGED_GROUP_NAME	tsmadmin	特権グループの文字列名。
LANG	en_US.UTF-8	ロケール設定

ドライバー、証明書、その他のファイル

Tableau Server イメージには、データコネクタまたはドライバーが事前にインストールされていません。Bash セットアップスクリプトを作成して、build-image スクリプトに Tableau Server に必要なデータコネクターをインストールするよう指示する必要があります。以下のステップに従います。

1. Tableau Server in a Container のセットアップツールが正しくインストールされていることを確認する
2. Tableau ドライバーページ https://www.tableau.com/en-us/support/drivers からドライバーをダウンロードする
3. ダウンロードしたドライバーファイルを、Tableau Server in a Container のセットアップツールの customer-files ディレクトリにコピーします。

   customer-files ディレクトリが Docker イメージにコピーされます。コンテナー内の次のパスに配置されます。/docker/customer-files
4. Tableau Server in a Container のセットアップツール customer-files/setup-script でファイルを編集し、Docker にドライバーのインストール方法を指示します。

   このスクリプトは、Docker イメージの構築時に実行される実行可能な bash ファイルです。(このファイルは、技術的には、環境のセットアップ、設定、ドライバーの更新や追加など、イメージ内の任意のアクションを実行するために使用できます。)

このスクリプトはコンテナ行されるため、Tableau Server in a Container のイメージで実行するときにコマンドが機能する必要があることに注意してください。

たとえば、"mydriver.rpm" という名前のドライバーをインストールする場合は、これを `setup-script` で次のように記述します。

```
yum install -y /docker/customer-files/mydriver.rpm
```

Tableau 管理ビュー

Tableau 管理ビューでは、PostgreSQL ドライバーがインストールされている必要があります。これらのビューを使用する場合は、上記のステップに従って PostgreSQL ドライバーをインストールする必要があります。管理ビューの詳細については、管理ビュー を参照してください。

カスタム ベース イメージを定義する

Tableau Server コンテナーの構築に使用される既定のベース イメージは、Docker Hub から提供される UBI 8 イメージです。場合によっては、別の Docker イメージ リポジトリからイメージをプルするように build image ツールを構成したほうがよいこともあります。たとえば、社内で Docker リポジトリを管理していて、`build-image` をパブリック リポジトリからプルしたくない場合などです。このベース イメージのパスをカスタマイズするには、次のステップに従います。

1. 次のコマンドを使用して、現在の `build-image` ツールの既定のベース イメージ名とバージョンを表示します

   ```
   ./build-image -b
   ```

2. 完全に一致するベース イメージ名とバージョンを Docker Hub から取得し、お気に入りのイメージ リポジトリに保存/キャッシュします (会社のイメージ ポリシーに従います)

3. build image ツールに戻ります。既存の環境ファイルを作成または変更して、値が次の新しい Docker イメージ レジストリ パスに設定されている `BASE_IMAGE_URL` 環境キーを含めます。

   ```
   BASE_IMAGE_URL=<custom-image-registry-path>
   ```

4. 次の環境ファイルを使用してイメージを構築します。

```
./build-image --accepteula -i <rpm> -e <path-to-env-file>
```

これらのステップを行うことにより、完全に異なるベースイメージを指定できます。この機能は、UBI 8 ベースのイメージでのみサポートされており、お客様がより安全なイメージを作成できるようにするために提供されています。

> Tableau で指定された既定以外のベースイメージを使用すると、正常に起動または機能しないイメージが生成されるリスクがあります。カスタム ベースイメージを使用する場合は、そのベースイメージで Tableau Server を正しく実行できるようにする必要があります。セキュリティ上の懸念などの理由で組織から許可されない場合以外は、既定のベースイメージを使用することをお勧めします。
>
> カスタム ベースイメージは、UBI 8 に基づいている必要があります。他のディストリビューションを使用すると、サポートされていないイメージになります。

yum とpip の内部リポジトリを使用する

コンテナー イメージ内の Tableau Server は、デフォルトの yum リポジトリとpip リポジトリを使用して依存パッケージをプルするように構成されています。リポジトリを編集、削除、追加する必要がある場合 (たとえば、セキュリティを向上させるために内部リポジトリを使用する場合など)、イメージ設定ツールの初期化スクリプトを変更する必要があります。

内部リポジトリを使用するには、ソースコードの `<setup_tool>/src/image/init/setup_default_environment.bash` スクリプトを変更します。内部リポジトリに必要なリポジトリファイルはすべて `<setup_tool>/src/image/init/` ディレクトリに格納します。このディレクトリが Docker イメージにコピーされます。

ベースイメージとセキュリティ

多くのお客様は、生成された Tableau Server Dockerイメージに対してコンテナー スキャン ツール (AquaScan や TwistLock など) を実行します。これらのセキュリティツールによって、潜在的なセ

キュリティの脆弱性 (または共通脆弱性識別子(CVE)) のレポートが生成されます。Tableau Server in a Container のイメージに関連する脆弱性には次の 2 つのタイプがあります。

- Tableau Server に関連付けられている CVE または依存関係のあるライブラリに関連付けられている CVE。
- 基盤となる Linux ディストリビューションに関連付けられている CVE。

Tableau は、Tableau Server に直接関連付けられている CVE の責任を担います。セキュリティチームはこれらのレポートを分析して影響と重要度を判断し、問題を解決する優先順位を付けます。ベースラインの修復の優先順位とタイムラインは、元の CVSS 重要度スコアによって決定されます。サードパーティコンポーネントのセキュリティ更新プログラムは、脆弱性を明らかにする実行可能なコードパスがない限り、通常は、古いリリースにはバックポートされません。

コンテナー化されたディストリビューション モデルを使用するお客様は、OS レイヤーの脆弱性に関するさまざまな課題に直面しています。従来の Tableau Server では、Tableau は安全なアプリケーションの提供に注力し、お客様はオペレーティング システム (Linux または Windows) の管理の責任を担っていました。しかし、コンテナー化されたことで、OS (バージョン 2023.3 以降は UBI 8) とアプリケーションが 1 つのコンテナーに一緒にパッケージ化されました。Tableau は、Tableau Server に関連付けられている CVE に対する責任と、サードパーティライブラリの CVE がお客様に影響を与えるかどうかを判断する責任を担います。ただし、OS レイヤーに関する問題の責任はお客様が負う必要があります。OS レイヤーのセキュリティ問題に対処するため、お客様は、ベース OS レイヤーを独自のバージョン (バージョン 2022.3 以降は UBI-8 ベース) に置き換えることができます。その際、Tableau Server が正常に動作していることを検証する必要があります。Tableau Server が依存しているライブラリを削除すると、セキュリティ上の問題により、Tableau Server のインスタンスが機能しなくなる可能性があります。ベース OS レイヤーを変更した場合、お客様はそれが機能することを検証する責任を担います。

Tableau Server in a Container を強化する

Tableau Server in a Container に標準的な強化ステップを適用できます。Tableau Server の強化の詳細については、セキュリティ強化チェックリストを参照してください。

展開構成の例

Docker

Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key>
-p 8080:8080
-d <Tableau Server in a Container image ID or tag>
```

自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key> \
-e TABLEAU_USERNAME=<myadmin> \
-e TABLEAU_PASSWORD_FILE=/etc/tableau-admin-secret \
-v <full-path-to-pw-file>:/etc/tableau-admin-secret \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

TSM 専用モード

```
docker run \
-e TSM_ONLY=1 \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

マルチノードの基本的な使用例

最初のノード

オプション 1: サーバー構成 (`CONFIG_FILE`) でマルチノードトポロジが指定される場合にこれを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-v <full-path-to-config-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<name-of-host-machine> \
-d <Tableau Server in a Container image ID or tag>
```

オプション 2: サーバー構成でマルチ ノードトポロジが指定されていない場合でも、マルチノード展開が必要な場合は、これを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e LICENSE_KEY=<key> -e ALWAYS_WRITE_BOOTSTRAP_FILE=1 \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<name-of-host-machine> \
-d <Tableau Server in a Container image ID or tag>
```

追加ノード

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e BOOTSTRAP_INSTALL=1 \
-p 8080:8080 -p 8800-9000:8800-9000 \
--hostname=<name-of-host-machine> \
-d <Tableau Server in a Container image ID or tag>
```

データ使用の外部化

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
--hostname=localhost \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Init コンテナーの基本的な使用例

Init コンテナー

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
-e INIT_CONTAINER=1 \
--hostname=localhost \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

コンテナーの実行

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
--hostname=localhost \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

シングルノードバックアップからの基本的な復元

```
docker run \
-v <full-path-to-backup-file>:/docker/config/backup/backup-
file.tsbak \
-v <full-path-to-config-only-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Docker の構成

```
version: '3.2'
services:
    tableau-server:
         hostname: localhost
         volumes:
              - <your-tsm-command-file>:/docker/config/tsm-
commands:ro
              - <your-config-file >:/docker/config/config.json:ro
         ports:
              - "8080:8080"
         image: ${IMAGE_NAME}
         environment:
              - LICENSE_KEY=<license-key>
```

Tableau Server in a Container - イメージの使用

概要

Tableau Server in a Container は、Tableau 初のコンテナーベースのサーバー製品です。Tableau Server in a Container は、Linux Docker コンテナ内で実行されるオールインワンの Tableau Server インスタンスです。つまり、Tableau Server in a Container のイメージは、自己完結型の Tableau Server アプリケーション全体を実行する Docker イメージです。Tableau Server in a Container は、コンテナーベースの環境で Tableau Server の実行をサポートする数多くのステップの中の最初のステップとなります。Tableau Server in a Container の概念は、Tableau Server が事前にインストールされた仮想マシン (VM) のようなものと考えると、簡単に理解することができます。イメージは UBI 8 イメージに基づいており、コンテナー内で (`systemd` の代わりに) `supervisord` を実行します。コンテナーが `supervisord` を開始すると、直ちに Tableau Server を初期化して起動しようとします。ここにあるドキュメントの多くは、Docker 環境に合わせて Tableau Server を実行できるように、構成を提供して自動化の活用方法を説明するためのものです。

Tableau Server in a Container のイメージのセットアップ ツールを使用すると、カスタム パッケージとアーティファクトを含むようにコンテナー イメージを作成してカスタマイズすることができます。このツールの主な機能の 1 つは、コンテナー イメージを構築し、カスタム データ コネクタをインストールすることです。

概念実証シナリオで Tableau Server in a Container のイメージをすばやくテストするには、Tableau Server in a Container のクイック スタートを参照してください。

Tableau Server in a Container に関する制限事項

- Tableau Server in a Container では、サーバー ATR を使用したライセンス認証のみをサポートします。Server ATR を使用したオフライン アクティベーションは、2023.1 以降でサポートされています。この機能はコンテナで利用できますが、追加の手順と承認が必要です。エアギャップ環境またはオフライン環境のコンテナで Tableau Server を実行する必要がある場合、詳細についてはアカウント担当者にお問い合わせください。
- Tableau Server in a Container では現在、Resource Monitoring Tool (RMT) エージェントはサポートされていません。
- Kerberos は、Tableau Server in a Container ではサポートされていません。

Tableau Server in a Container のイメージ

Tableau Server in a Container のイメージは、すべての Tableau Server を含む Docker イメージです。イメージは、Tableau Server in a Container のセットアップ ツールを使用して作成されます。作成時、イメージには Tableau Server が含まれていますが、まだ初期化されていません。Tableau Server in a Container のイメージの既定のユーザーは、ルート権限のないユーザーです。

前提条件

configure-container-host スクリプトを実行する

Tableau Server をコンテナーなしでインストールすると、インストール プロセスの一部として特定のリソース制限とコア ダンプ プロパティが変更されます。これにより、Tableau Server のパフォーマンスを最適化しやすくなります。Tableau Server in a Container のイメージには、ホスト マシンでこれらの変更を行う機能がないため、Tableau Server コンテナーのセットアップ ツールで提供される configure-container-host スクリプトを、コンテナー内の Tableau Server in a Container のイメージを実行する任意のマシンで実行することをお勧めします。これにより、Tableau Server in a Container のイメージのパフォーマンスがコンテナー外部のイメージのパフォーマンスと同等になります。

`configure-container-host` スクリプトを実行するには:

1. Tableau Server in a Container のセットアップ ツールの最上位ディレクトリで、スクリプト `configure-container-host` を見つけます。
2. それを Tableau Server の実行を計画している環境にコピーします。
3. Tableau Server in a Container のイメージの既定ユーザーとして実行される特権のないユーザー アカウント/uid を決定します。このユーザーはホスト マシンに存在していて、Tableau Server コンテナー内の環境変数 `UNPRIVILEGED_TABLEAU_UID` に設定されている UID と一致している必要があります。Docker イメージの作成時にこれを設定しなかった場合、コンテナー内の権限のない既定ユーザー ID は 999 です。Docker ユーザー マッピングを使用している場合、この UID はホスト マシンに存在するユーザーに対応している必要があります。

4. 次のスクリプトを root として実行します。

```
sudo ./configure-container-host -u <uid>
```

イメージを実行する

Tableau Server in a Container の Docker イメージを実行するために、Tableau Server in a Container のイメージを実行させる最も簡単なコマンドは次のとおりです。

```
docker run \
-e LICENSE_KEY=<key>
-p 8080:8080
-d <Tableau Server in a Container image ID or tag>
```

これにより、ドッカーがバックグラウンドで実行され、しばらくすると、Tableau Server のインスタンスが完全にインストールされます。Tableau Server は、イメージを実行しているコンピューターのハードウェアに応じて、完全に起動するまでに 10 分から 20 分かかります。`docker ps` コマンドを入力すると、コンテナが実行されていることを確認できます。Tableau Server が操作可能になったら、最初の Tableau Server の管理者アカウントを作成する必要があります。このステップは自動化できます。詳細については、Tableau Server 初期管理者の作成の自動化を参照してください。

基本的な実行引数の概要

docker run コマンドで使用されるオプションはすべて必要なものであり、多くの場合、イメージ内のさまざまな機能を利用できるように、さらに多くのオプションが提供されます。ここでは、Tableau Server in a Container の最も単純な docker run コマンドで使用される引数だけを詳しく見てみましょう。

引数	説明
-e LICENSE_KEY=<key>	Tableau Server はライセンスを取得する必要があります。この環境変数には、サーバーのライセンス認証に使用されるキーが格納されます。これは初期化プロセスの必須コンポーネントです。コンマで区切った複数のライセンス名を入力できます。
-p 8080:8080	これにより、Docker は、コンテナー内のポート 8080 を公開し、それ

引数	説明
	をホストマシン上のポート8080 にバインドするように指示されます。最初の8080 値は構成可能です。これを変更すると、ホストにマッピングされているポートが変更されます。Tableau Server では既定で、コンテナー内のポート8080 でユーザートラフィックを受信することが想定されていますが、このポートを別のホストポートで公開するか、公開しないかを選択することができます。

Tableau Server 初期管理者の作成の自動化

Tableau Server を初めて起動する場合は、Tableau Server へのリモートネットワーク接続が許可される前に、初期管理者ユーザーを作成する必要があります。これを行うには、コンテナー内で `tabcmd initialuser -s localhost:8080 -u <username> -p <password>` コマンドを実行します。環境変数を介して管理者の認証資格情報を設定することもできます。`TABLEAU_USERNAME` および `TABLEAU_PASSWORD` または `TABLEAU_PASSWORD_FILE` (推奨) は、初期管理者の認証資格情報を渡すために設定できる環境変数です。パスワード管理の詳細については、パスワード管理を参照してください。

`tabcmd initialuser` コマンドの詳細については、initialuserを参照してください。

例

```
docker run \
-e LICENSE_KEY=<key> \
-e TABLEAU_USERNAME=<myadmin> \
-e TABLEAU_PASSWORD_FILE=/etc/tableau-admin-secret \
-v <full-path-to-pw-file>:/etc/tableau-admin-secret \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

ライセンス発行

コンテナーでのライセンス発行

Tableau Server in a Container のライセンス発行では、実行承認 (ATR) サービスを使用し、ライセンス契約を途切れさせることなく、クラウド、コンテナ、仮想環境などに展開された Tableau

Server をライセンス認証できます。ATR サービスでは、プロダクトキーの有効期限が切れるまで、構成可能な期間 (ATR 期間) 中に短期リースが提供されます。ATR はハードウェア基盤の変更から Tableau のライセンス発行を抽象化します。これはコンテナー展開の基本的な側面です。サーバー ATR では、コンテナーが Tableau によってホストされている ATR サービスに到達できる必要があるため、コンテナーにはインターネットアクセスが必要です。Tableau Server in a Container は、オフラインまたは手動によるライセンス認証はサポートされていません。詳細については、実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証するを参照してください。

> **重要:** 環境変数は `LICENSE_KEY` または `LICENSE_KEY_FILE` のいずれかを入力する必要があります (1 つのみを設定)。

Tableau Server in a container をアップグレードするとき、ライセンス数をすでに最大限使用している場合は、ATR 期間 (既定では 4 時間/14,400 秒) が経過するまで Tableau Server を開始できません。ATR 期間の設定や変更の詳細については、実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証するを参照してください。

ライセンス環境変数

Tableau Server in a Container は、環境変数を使用したライセンスキーの設定をサポートしています。`LICENSE_KEY` には、コンマ区切りリストから 1 つ以上のキー (-e LICENSE_KEY="<key1>、<key2>") を含めることができます。

例

```
docker run \
-e LICENSE_KEY="<key1>, <key2>" \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

ライセンス ファイル

Tableau Server in a Container では、ファイルを使用した設定プロダクトキーがサポートされています。ファイルをコンテナー内の既定のプロダクトキー ファイルの場所 (`/docker/config/license_file`) にマウントするか、環境変数 `LICENSE_KEY_FILE` で指定された場所にマウントします。

例

```
docker run \
-v <full-path-to-license-file>:/docker/config/license_file \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

要求されたライセンスのリース期間

環境変数 REQUESTED_LEASE_TIME を設定すると、Tableau Server コンテナー内の ATR ライセンスのリース期間を指定できます。リース期間は秒単位で指定する必要があり、最小期間は 3,600 秒 (または 1 時間) です。Tableau Server を実験およびテストするときは、ライセンス承認の最大リース制限に達する可能性を低減するために、リース期間を短くすることをお勧めします。本番環境の展開では、REQUESTED_LEASE_TIME パラメーターを設定しないことを強くお勧めします (既定値を使用)。これにより、Tableau で理想的なリース期間が決定されるようになります。

例

```
docker run \
...
-e REQUESTED_LEASE_TIME=<time-in-seconds> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

初期化されていないイメージを実行する

Tableau Server には 2 つのフェーズのインストールがあり、最初に Tableau サービス マネージャー (TSM) サービスがインストールされます。一般的なオンプレミスインストールでは、この手順は、サーバー管理者がサーバーを登録し、ライセンスをアクティブ化し、サーバーが望む動作をするように構成する時間です。インストールの 2 番目のフェーズでは、Tableau Server プロセスを設定および開始します。このプロセスでは、エンドユーザー トラフィックおよび関連するビジネス ロジックが処理されます。

Tableau Server in a Container のイメージの既定の動作では、すべてのインストール ステップが自動化され、docker run コマンドによって最終的に完全に機能するサーバーになります。ただし、Tableau Server in a Container のイメージを起動し、TSM サービスのみ (サーバー管理者が initialize-tsm のみを実行した場合に期待する内容) を実行する場合、環境変数として TSM_ONLY フラグを渡すことで、これを行うことができます。

例:

```
docker run \
-e TSM_ONLY=1 \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

イメージを操作する

Tableau Server in a Container のイメージを実行している場合は、TSM コマンドとtabcmd コマンドを直接呼び出すことができます。これらのツールは、pid 1 ユーザー(この時点ではルート) の環境パスに直接追加されます。つまり、TSM コマンドとtabcmd コマンドを次のように呼び出すことができます。

```
docker exec -it <container> tsm status -v
```

```
docker exec -it <container> tabcmd initialuser -s localhost -u
<admin> -p <secret>
```

さらに、コンテナ内でシェルを開いて、より一般的な操作を実行することもできます。これは通常、デバッグの目的以外は推奨されません。

```
docker exec -it <container> bash
```

TSM Web UI とリモートCLI

既定では、TSM Web UI とリモートCLI にはアクセスできません。これは、認証にはユーザー名とパスワードが必要であり、デフォルトではコンテナーで tableau Server プロセスを実行しているユーザーにはパスワードが提供されないためです。これはセキュリティ上の理由によります。(イメージに既定のパスワードを同梱することは、リモートアクセスが許可されてしまうため、推奨されていません。)場合によっては、TSM Web UIと、TSM CLI を使用したリモートアクセスの呼び出しが役立つことがあります。これらの機能を使用する場合は、以下に概説するステップに従って、リモートアクセス ユーザーアカウントを作成する必要があります。

TSM Web UI および CLI の詳細については、Tableau サービス マネージャーの Web UI へのサインインを参照してください。

TSM リモートユーザーを作成する

セットアップツールを使用して Tableau Server in a Container のイメージを構築するときの環境変数 TSM_REMOTE_UID および TSM_REMOTE_USERNAME を指定します。イメージに TSM 対応アカウントを作成するには、実行時に使用できないイメージのアクセス権限が必要です。そのため、これは、Docker イメージが Tableau Server in a Container のセットアップツール (build-image) で構築されている場合にのみ実行できます。

Tableau Server in a Container のセットアップツール environment ファイルの例:

```
TSM_REMOTE_UID=1010
TSM_REMOTE_USERNAME=myuser
```

TSM リモートユーザーのパスワードを設定する

Tableau Server in a Container のイメージの実行時には、アカウントのパスワードが必要です。このアカウントのパスワードを設定する方法は 2 つあります。

シークレットファイル (推奨)

remote-user-secret という名前の付いたファイルを作成します。パスワードをファイルに書き込み、実行時にそれをコンテナーにマウントします。TSM_REMOTE_PASSWORD_FILE は、期待される場所を決定します (既定の場所は、コンテナー内のシークレットファイルの /docker/config/remote-user-secret) です。

remote-user-secret ファイルの例:

```
mypassword
```

docker run コマンドの例:

```
docker run \
-e LICENSE_KEY=<key>
-v {absolute-path}/remote-user-secret:/docker/config/remote-user-
secret
-p 8080:8080 \
-p 8850:8850 \
-d <Tableau Server in a Container image ID or tag>
```

環境変数

Docker イメージを開始するときは、環境変数 TSM_REMOTE_PASSWORD を設定することもできます。

docker run コマンドの例:

```
docker run \
-e LICENSE_KEY=<key>
-e TSM_REMOTE_PASSWORD=<password>
-p 8080:8080 \
-p 8850:8850 \
-d <Tableau Server in a Container image ID or tag>
```

セキュリティに関する注意事項

- TSM 要求トラフィックを受信するには、ポート8850 を公開する必要があります。
- 実行時にパスワードがイメージで適切に設定されていない場合、コンテナーはすぐに終了します。
- TSM は、イメージの Linux ユーザー アカウント システムに依存しています。この場合、イメージ内のアカウントには制限が設けられています。つまり、アカウントのシェルが制限され、実行できるのは、次の 2 つのコマンド /bin/true および passwd に限定されます。

TSM リモート ユーザーのパスワードをローテーションする方法

TSM リモート ユーザーのアカウントのパスワードをローテーションする場合は、以下のオプションのいずれかを使用します。

Tableau Server in a Container を新たに起動する

アカウントのパスワードは、コンテナーが起動するたびに設定されます。コンテナーの外部で Tableau データを永続化する場合は、新しいパスワードを使用して新しいイメージを開始すると、パスワードが効果的にローテーションされます。

1. 実行中のイメージをシャットダウンして削除する
2. イメージ構成の環境変数 (上記参照) TSM_REMOTE_PASSWORD または TSM_REMOTE_PASSWORD_FILE のどちらかに新しいパスワード値を設定します。
3. イメージを再起動する

実行中のコンテナー内でパスワードを手動でローテーションする

イメージをシャットダウンしたくない場合は、パスワードを手動でローテーションすることができます。

1. コンテナー内でシェルを開く
2. `su` コマンドを使用して、リモートユーザーアカウントとしてログインします。
3. `passwd` コマンドを実行して、パスワードを変更します。

> **警告:** これらの手動ローテーションは、コンテナーインスタンスの書き込みレイヤーが残されている間は保持されます。コンテナーを削除すると、新しいコンテナーを起動したときに、この手動による変更は適用されません。

初期構成のオプション

必要な Tableau Server の動作を取得するには、Tableau Server in a Container を構成することが不可欠です。Tableau Server in a Container は Tableau Server のクリーンインストールであるため、コンテナーの外部で Tableau Server を構成する場合と同じ情報をコンテナーに提供する必要があります。

ランタイム環境変数

以下のランタイム環境変数を設定すると、Tableau Server in a Containerr に Tableau Server の展開方法を指示することができます。これらのサブセットについて詳しく説明します。

これらの値はすべて上書きが可能なため、より柔軟に構成することができます。

環境名	既定	説明
ACCEPT-EULA	`0`	Tableau Server in a Container のセットアップツールを使用してイメージを構築すると、自動的に 1 に設定されます。
LICENS-E_KEY		サーバーのライセンス認証に使用するプロダクトキーに設定します。コンマで区切られた複数のライ

環境名	既定	説明
		センスを使用できます。
LICENS-E_KEY_FILE	/docker/config/lice-nse_file	ライセンス ファイルへのファイル パス。ライセンス ファイルの形式は、1 行につき 1 つのライセンス キーにする必要があります。
REGIST-RATIO-N_FILE	/docker/config/tabl-eau_reg.json	イメージ内の登録ファイルへのファイル パス。既定では、これには、Tableau Server in a Container が構築されたときに提供された登録情報が含まれます。これは実行時に上書きできます。詳細については、Tableau Server のライセンス認証と登録を参照してください。
REGIST-RATIO-N_DATA		登録情報は、別の方法で実行時に上書きすることもできます。この環境変数は、Tableau Server 登録ファイルにあるものと同じ登録情報を含むシリアル化された JSON 文字列に設定する必要があります。詳細については、Tableau Server のライセンス認証と登録を参照してください。
TABLEA-U_USERNA-ME		これは Tableau Server の初期管理者アカウントを指します。これは推奨されていますが、使用は任意です。このユーザーが設定されていない場合、Tableau Server の初期管理者アカウントは tabcmd を使用して設定する必要があります。この変数を値に設定する場合は、パスワードも必要です。これは、Tableau Server が初めて初期化されるときにのみ使用されます。この値を設定すると、Tableau Server in a Container は自動的にユーザーの初期化を試みるように指示されます。詳細については、管理者アカウントの追加を参照してください。
TABLEA-U_PASSW-		tableau ユーザーのプレーン テキスト パスワード。これは Tableau Server の初期管理者アカウントを指します。これは、TABLEAU_USERNAME が指定

環境名	既定	説明
ORD		されている場合に必要です。詳細については、管理者アカウントの追加を参照してください。
TABLEA-U_ PASSW-ORD_ FILE		Tableau ユーザーのパスワードテキストのみを含むファイルへのファイル パス。これは Tableau Server の初期管理者アカウントを指します。これは、TABLEAU_USERNAME が指定されている場合に必要です。詳細については、管理者アカウントの追加を参照してください。
CONFI-G_FILE	/docker/config/conf-ig.json	既定の TSM 構成ファイルへのファイルパス。このファイルは、Tableau Server の構成に使用されます。詳細については、構成ファイルの例を参照してください。 CONFIG_FILE を使用している場合は、CONFIG_DATA を設定しないでください
CONFI-G_DATA		これは、CONFIG_FILE の代わりに使用できます。外部ファイルをマウントせずにサーバーに構成を提供する場合は、この環境変数を TSM 構成ファイルのシリアル化された同等の内容に設定します。 例: CONFIG_DATA="{\"configEntities\":{\"identityStore\":{\"_type\":\"identityStoreType\",\"type\":\"local\"}}}" 詳細については、構成ファイルの例を参照してください。 CONFIG_DATA を使用している場合は、CONFIG_FILE を設定しないでください
IGNOR-	0	0 または 1。1 に設定されている場合、コンテナー

環境名	既定	説明
E_ TOPOLO-GY_ CONFIG		は、CONFIG_FILE で指定された構成ファイルに存在するトポロジ関連の構成をすべて無視します。
BACKU-P_FILE	/docker/config/back-up/backup-file.tsbak	Tableau Server バックアップ ファイル (.tsbak) へのファイル パス。初期化中に指定された場合、サーバーは復元を試みます。IDENTITY_ STORE_FILE も提供する必要があります。
IDENTIT-Y_ STORE_ FILE	/docker/config/iden-tity-store-config.json	バックアップ ファイルから復元中に構成ファイルを識別します。
INIT_ CONTAI-NER	0	0 または 1。1 に設定されている場合、Tableau Server は TSM とTableau Server の初期化のみを試み、初期化が完了するとコンテナーは終了します。
TSM_ ONLY	0	0 または 1。Tableau Server rpm をインストールし、初期化-tsm を実行することと同等です。TSM (Tableau サービス マネージャー) サービスのみが開始されます。コンテナーを初めて初期化する場合にのみ機能します (以前に初期化したサーバー ディレクトリで Tableau Server in a Container を起動している場合は機能しません)。
BOOTST-RAP_ INSTALL	0	0 または 1。サーバーが最初のノードか追加ノードかを示します。1 に設定されている場合、コンテナーは、bootstrap ファイルが $BOOTSTRAP_FILE で指定された場所に存在するまで無期限に待機します。
ALWAY-S_	0	0 または 1。1 に設定すると、コンテナーは BOOTSTRAP_FILE で指定された場所に

環境名	既定	説明
WRITE_BOOTSTRAP_FILE		bootstrap ファイルを書き込みます。
WAIT_FOR_BOOTSTRAP_FILE	1	0 または 1。1 (既定) に設定されていて、それがワーカー インストール (BOOTSTRAP_INSTALL=1) であることをコンテナーが検出した場合、コンテナーは、BOOTSTRAP_FILE で設定されたパスに存在するファイルが検出されるまで無期限に待機します。起動プロセスの実行時に 0 に設定すると、この待機はスキップされます。これは、デバッグの一部のケースで役立ちます。
BOOTSTRAP_FILE	/docker/config/bootstrap/bootstrap.json	bootstrap ファイルへのファイル パス。ワーカーコンテナにのみ適用されます。このファイルはブートストラップファイルのみを指します。この一般的な使用方法として、ターゲット ファイルのディレクトリ(既定は /docker/config/bootstrap) をホストにマウントします。
BOOTSTRAP_DATA		これをBOOTSTRAP_FILE の代わりに使用できます。外部ファイルをマウントせずに bootstrap ファイルを提供する場合は、この環境変数をシリアル化された TSM bootstrap ファイルと同じ内容に設定します。BOOTSTRAP_FILE を使用する場合は、BOOTSTRAP_DATA を設定しないでください。
PORT_RANGE_MIN	8800	パフォーマンス上の理由から、Docker で 1000 ポートを公開するとDocker イメージの起動時間に悪影響を与える可能性があるため、Tableau Server オンプレミスの既定の 8000 ~ 9000 ポート範囲ではなく、200 ポートのみ (8800 ~ 9000) を公開することをお勧めします。詳細については、以下のライセンス発行 ポートとTSM ポートの公開参照してください。

環境名	既定	説明
PORT_RANGE_MAX	9000	Docker で 1000 ポートを公開すると Docker イメージの起動時間に悪影響を与える可能性があるため、Tableau Server オンプレミスの既定の 8000 ~ 9000 ポート範囲ではなく、200 ポートのみ (8800 ~ 9000) を公開することをお勧めします。詳細については、以下のライセンス発行ポートと TSM ポートの公開参照してください。
HTTP_PROXY		http 要求をプロキシ サーバーに転送するには、プロキシ ホストをポイントするようにこの環境変数を設定します。たとえば、ポート 8080 の example-host にプロキシを設定するには、`HTTP_PROXY=http://example-host:8080/` のように設定します。
HTTPS_PROXY		https 要求をプロキシ サーバーに転送するには、プロキシ ホストをポイントするようにこの環境変数を設定します。たとえば、プロキシをポート 443 の example-host に設定するには、`HTTPS_PROXY=http://example-host:443/` のように設定します。環境変数 `HTTPS_PROXY` の URL を指定するときは、必ず 'http' を使用してください。
NO_PROXY		プロキシ サーバーを回避するには、no_proxy 変数で例外を指定します。この変数は、プロキシ サーバーが内部アドレスを経由しない場合に使用してください。また、このプロキシ構成に例外を追加することで、ローカル Tableau Server クラスタ (現在設定している、または今後設定する場合) 内の通信すべてがプロキシ サーバーに送られないようにする必要もあります。各コンピューターのホスト名と IP アドレスの両方を入力し、コンテナーのホスト名を追加します。さらに、ローカル コンピューターの正規ホスト名 (localhost) と IP アドレス

環境名	既定	説明
		(127.0.0.1) を含めます。たとえば、3 ノード クラスタの例外を指定するには、次のようにします。 `NO_PROXY="localhost,127.0.0.1,hostname1,hostname2,hostname3,IP1,IP2,IP3"`
COORDINATION_SERVICE_CLIENT_PORT	`PORT_RANGE_MIN` と `PORT_RANGE_MAX` の間の任意のポート	コーディネーション サービスのクライアント ポート。
COORDINATION_SERVICE_PEER_PORT	`PORT_RANGE_MIN` と `PORT_RANGE_MAX` の間の任意のポート	コーディネーション サービスのピア ポート。
COORDINATION_SERVICE_LEADER_PORT	`PORT_RANGE_MIN` と `PORT_RANGE_MAX` の間の任意のポート	コーディネーション サービスのリーダー ポート。
LICENSE_SERVICE_VENDOR_DAEMO-	`PORT_RANGE_MIN` と `PORT_RANGE_MAX` の間の任意のポート	ライセンス発行サービスのベンダー デーモン ポート。

環境名	既定	説明
N_PORT		
AGENT_FILE_TRANSFER_PORT	PORT_RANGE_MIN と PORT_RANGE_MAX の間の任意のポート	エージェントサービスのファイル転送ポート。
CONTROLLER_PORT	PORT_RANGE_MIN と PORT_RANGE_MAX の間の任意のポート	コントローラー サービスの https ポート。
REQUESTED_LEASE_TIME	既定では、現在 4 時間に設定されています。	サーバー ATR のライセンス認証で要求されるリース期間を設定します。時間値は秒単位で指定する必要があり、最小期間は 14400 秒 (または 4 時間) です。通常、この値を変更することは、本番環境の展開では推奨されていません。ただし、Tableau Server in a Container を使用して開発またはプロトタイピングする場合は、ライセンス認証の損失を最小限に抑えるために、この値を最小値に設定することをお勧めします。

読み取り専用環境変数

これらは、Tableau Server in a Container のイメージの基本的なプロパティの一部を記述する環境プロパティです。これらの値を上書きすることは推奨されていません。

環境名	既定	説明
PRE_INIT_COMMAND_SCRIPT	${DOCKER_CONFIG}/customer-files/pre_init_command	Tableau Server の初期化の前に Tableau Server で実行されるユーザー カスタム bash/ 実行可能ファイルへのパス。**注:** ファイルにすべてのユーザーの実行パーミッションがあることを確認するか、chmod +rx

環境名	既定	説明
		<path-to-pre-init-command-file> を実行してください。
POST_INIT_COMMAND_SCRIPT	${DOCKER_CONFIG}/customer-files/post_init_command	サーバーが完全に機能して実行中になった後に Tableau Server で実行されるユーザーカスタム bash/実行可能ファイルへのパス。**注:** ファイルにすべてのユーザーの実行パーミッションがあることを確認するか、chmod +rx <path-to-post-init-command-file> を実行してください。
DATA_DIR	/var/opt/tableau/tableau_server	Tableau Server のビットが書き込まれるデータディレクトリ。
INSTALL_DIR	/opt/tableau/tableau_server	Tableau Server インストールのビットが書き込まれるインストール ディレクトリ。
SERVICE_NAME	Tableau Server	コンテナーで実行されているアプリケーションの名前。
SERVICE_VERSION	N/A	コンテナーにインストールされている Tableau Server のバージョン。
DOCKER_CONFIG	/docker	Tableau 固有の Docker 構成を保存するディレクトリ。
ENV_FILE	${DOCKER_CONFIG}/customer-files/environment	すべてのユーザー環境の上書きを含むファイル。

構築時の環境変数

BASE_IMAGE_URL	ビルドツール コマンド`build-image -b` を使用します。	build-image ツールとDockerfile で指定された既定のイメージは、公式にサポートされている唯一のベースイメージです。このパラメーターを使用して、この特定のベースイメージのコピーをカスタム Docker イメージ リポジトリからプルするか、カスタム ベースイメージを定義できます。カスタム定義のベースイメージを使用する場合は、UBI 8 に基づいており、Tableau Server を適切に実行するために必要なリソースが含まれていることを確認する必要があります。カスタム ベースイメージの詳細については、Tableau Server in a Container - イメージの使用を参照してください。
PRIVILEGED_ TABLEAU_GID	`997`	特権のある Tableau グループの GID。
UNPRIVILEGED_ TABLEAU_GID	`998`	特権のない Tableau グループの GID。
UNPRIVILEGED_ TABLEAU_UID	`999`	Tableau プロセスを実行するユーザーの UID (シングル ユーザー展開)。
UNPRIVILEGED_ USERNAME	`tableau`	特権のないユーザーの文字列名。
UNPRIVILEGED_ GROUP_NAME	`tableau`	特権のないグループの文字列名。
PRIVILEGED_	`tsmadmin`	特権グループの文字列名。

GROUP_NAME		
LANG	en_US.UTF-8	ロケール設定

Tableau Server 構成の上書き

これらの環境変数は、コンテナー内の任意のファイルを指すために Docker によって上書きできます。たとえば、別のマウント ポイントを指定したい場合に、これを行うことができます。

Tableau Server を起動して実行するには、構成ファイルが必要です。

```
CONFIG_FILE=/docker/config/config.json
```

CONFIG_FILE は TSM 構成ファイルを指します。形式と使用法は、構成ファイルの例で説明されている構成ファイルと同じです。

初期化前および初期化後のコマンド

Tableau Server は、サーバーを初期化前の状態から完全に実行中の状態にする自動インストール スクリプトを実行します。しかし、初期化プロセスで独自の自動化コードを追加したい場合もあります。これを行うために、初期化前スクリプトと初期化後スクリプトの 2 つのフックが用意されています。

初期化前スクリプト

このスクリプトは、TSM の基本プロセスが初期化された直後、および他の TSM セットアップ ステップが実行される前に実行されます。これは、Tableau Server を実行する前に TSM 構成コマンドを実行する場合に役立ちます。この時点で行われた構成の変更は、スクリプトの完了後に通常の Tableau Server の自動化によって適用されるため、変更を適用する必要はありません。

初期化後のスクリプト

このスクリプトは、他のすべてのTableau Server の初期化と起動の自動化が完了した後に実行されます。このスクリプトを実行すると、Tableau Server は完全に機能し、実行している状態にな

ります。この時点で行われた構成の変更は適用する必要があります。

手順

イメージ内のこれらのフックの 1 つにカスタム スクリプトを追加するには、次の手順に従います。

1. カスタム スクリプトを書く
2. カスタム スクリプトを Tableau Server in Containers のビルド イメージ ツールの `customer-files` ディレクトリにコピーします。
3. スクリプトを実行するタイミングに応じて (両方のフックは別々に使用できます)、スクリプトの名前を `pre_init_command` または `post_init_command` のいずれかに変更します。
4. スクリプトのパーミッションが他の (`chmod +rx <command-file>`) によって実行可能であることを確認するか、所有権のパーミッションがコンテナー内の特権のないユーザーと一致することを確認します。

ユーザー構成

Tableau Server は、特権のないユーザーを使用してサーバープロセスを実行します。このユーザーは、Tableau Server in Container の初期化中にコンテナー内に作成されました。既定では、このユーザーは `tableau` という名前で、UID は 999 です。マウントを使用してホスト マシン上のデータを外部に保存する Tableau Server in a Container を展開する場合は、UID を変更してホスト コンピューター上の UID にマッピングすることもできます。Docker ユーザーの名前空間を使用して、これを行うこともできます。

Tableau Server in a Container のユーティリティとツール

Tableau Server in a Container のすべてのユーティリティおよびツール関数は、次のディレクトリに配置されます。

`/docker/`

ファイル パーミッションの管理

コンテナーに任意の構成ファイルを渡す場合、コンテナー内で Tableau Server プロセスを実行しているユーザーがファイルにアクセスするためのパーミッションを持っていることを確認する必要があります。コンテナーにマウントされているファイルへのすべてのユーザー アクセスを許可しないようにするには、コンテナー内で Tableau Server を実行しているユーザーの UID や GID を変更して、ホスト上のユーザーまたはグループの所有者と一致させることができます。コンテナー ユーザーは、環境変数

UNPRIVILEGED_TABLEAU_UID (既定: 999) によって決定される UID と、UNPRIVILEGED_TABLEAU_GID (既定: 998) によって決定される GID を持っています。これらの値は、環境変数を上書きすることによって変更することができます。または、Docker ユーザーの名前空間をマッピングすることにより、コンテナー内の UID/GID をホスト上の別の UID/GID に関連付けることもできます。

パスワード管理

特定の機能とオプションでは、ユーザーの認証資格情報をコンテナーの構成設定として入力する必要があります。Tableau の初期管理の認証資格情報は、任意の認証資格情報の例であり、これを入力することで追加機能を有効にすることができます。この場合、パスワードの設定方法は 2 つあります。1 つ目の方法では、パスワードを含むファイルを指定し、環境変数へのファイル パスを指定します。2 つ目の方法では、パスワードを直接保存する環境変数を設定します。

より安全な推奨オプションとして、パスワードをコンテナーへのファイル パスとして指定することができます。ファイルにシークレットを含めることは、Docker、Docker Swarm、Kubernetes、およびその他のコンテナ オーケストレーション システムで十分にサポートされているパターンです。パスワードを環境変数に直接保存することは一般的なパターンであり、サポートされていますが、通常はパスワードの安全性は低くなります。

例

TABLEAU_USERNAME 資格情報を見てみましょう。ユーザーのパスワードは、TABLEAU_PASSWORD または TABLEAU_PASSWORD_FILE のいずれかで指定できます。Tableau Server in a Container のイメージを実行する場合、パスワードを指定するためにいずれかの環境変数を入力することができます。

パスワード ファイルの環境変数では、コンテナー内のファイル パスが有効なシークレット ファイルであると想定されています。シークレットファイルは、シークレットを含む 1 行にする必要があります。

シークレット ファイルの使用例

```
docker run \
...
-e TABLEAU_USERNAME=admin \
-e TABLEAU_PASSWORD_FILE=/etc/admin-secret \
```

```
-v <full-path-to-pw-file>:/etc/admin-secret \
-d <Tableau Server in a Container image ID or tag>
```

シークレットファイルの内容の例

```
mypassword23879172
```

または、パスワードをパスワード環境変数にプレーンテキストで直接保存することもできます。この方法の安全性は低いですが、コンテナーで使用する場合は便利で一般的なパターンです。

例

```
docker run \
...
-e TABLEAU_USERNAME=admin \
-e TABLEAU_PASSWORD=password \
-d <Tableau Server in a Container image ID or tag>
```

Tableau Server を実行した後に構成する

Tableau Server が初期化され実行中になったら、サーバーを操作する最適な方法として、TSM CLI ツールを使用します。これは、管理タスクを実行するための従来の Tableau Server ツールです。将来的には、実行間に提供される環境変数 CONFIG_FILE の静的構成の変更に対応する Tableau Server がサポートされる予定です。ただし、現時点では、Tableau Server が初期化された後は、従来のツールを使用してサーバーを操作する必要があります。

tsm コマンドラインの詳細については、tsm コマンドライン リファレンスを参照してください。

ステータス

イメージで提供される Tableau Server には、2 つのステータス チェックが用意されています。これらは、サーバーの稼働状態と準備状況を確認するために使用できます。

稼働状態チェック

稼働状態チェックは、TSM サービスが実行されているかどうかを示します。これは、Tableau Server のオーケストレーション サービスが動作していて機能しているかどうかを示します。このチェックは以下を使用して呼び出すことができます。

```
/docker/alive-check
```

もう1 つのオプションは、Web ブラウザーを介して管理機能を提供するために Tableau Controller サービスが実行するポート8850 を公開することです。TCP ヘルス チェックを使用してサービスの正常性を定期的にチェックできます。

準備完了チェック

準備状況チェックは、Tableau Server が実行されているかどうか、およびビジネス サービスがトラフィックを受信する準備ができているかどうかを示します。これは、以下のスクリプトを使用して判断できます。

```
/docker/server-ready-check
```

もう1 つのオプションは、ポート8080 (または Tableau Server がトラフィックを受信するためにバインドされているポート) に対して tcp ヘルス チェックを使用することです。サーバーの準備完了チェックは TSM に報告されたサービス ステータスに基づいているため、この種類の tcp ヘルス チェックはサーバーの準備完了チェックよりも信頼性が高く、サービス状態が更新されると遅延する場合があります。

データの永続化

多くのユーザーが、コンテナーを使用する際は、重要な情報を失うことなく、コンテナーをシャットダウンして電源を入れ直すことができるようにしたいと考えています。Tableau Server in a Container のイメージではこの点のサポートが万全であり、特定のディレクトリをコンテナーの外部にマウントできるため、コンテナー インスタンスを完全に破棄または削除してもデータを保持できます。このデータを使用して、別のコンテナー インスタンスを開始し、前のコンテナーが中断した場所から再開できます。

次のセクションでは、さまざまな種類の管理状態について説明します。

Tableau Server データ

サーバー データはすべてデータ ディレクトリに保存されます。データ ディレクトリには、すべてのユーザー関連データとサービス ランタイム メタデータが保存されます。このデータを外部に配置すると、Tableau Server in a Container を完全に削除した後でもユーザーのデータを永続化できます。

このデータは転送可能であり、AWS EBS ボリュームなどのクラウド管理のブロック ストレージ システムで使用できます。

Tableau Server in a Container を外部ファイルストアと組み合わせて使用する場合、データディレクトリは EBS 上にある必要があります。データディレクトリにはネットワーク ファイルシステム (NFS など) を使用しないでください。外部ファイルストア ディレクトリは、NFS ボリューム上に配置できます。

静的ホスト名

Tableau Server では動的ホスト名の変更は処理できないため、コンテナーの内部ホスト名を指定して、コンテナーの実行間で一貫性を保つ必要があります。コンテナ内のホスト名は任意であり、任意の値に設定できます。この `--hostname` オプションを使用すると、コンテナの内部ホスト名を指定できます。**同じ永続データを使用する後続のコンテナーが、同じホスト名の値を使用して実行されていることを確認します。**

これをマルチノード サーバーのインストールと混同しないでください。これらの場合は、追加ノードにそれぞれ異なるホスト名を割り当てる必要があります。重要なのは、単一のコンテナーを再起動する場合、そのインスタンスに対して同じ永続データを使用する後続のコンテナーが同じホスト名を持つ必要があります。

完全な例

これは、データディレクトリがコンテナーの外部にマウントされている例です。

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

バックアップと復元

Tableau Server in a Container では、バックアップ ファイル (.tsbak) からの Tableau Server の作成と復元がサポートされています。最初のステップでは、Tableau Server in a Container のイメージを実行し、バックアップ ファイル (.tsbak) をイメージにマウントし、バックアップ ファイルへのファイル パスを使用して環境変数 `BACKUP_FILE` を設定します。また、環境変数 `CONFIG_FILE` では、バック

アップ json 構成ファイルを提供する必要があります。Tableau Server コンテナーでは、マルチノード展開の場合でも復元プロセスが自動化されます。この自動化のある時点でシステムを完全に復元できない場合は、コンテナーを使用しない展開と同じように、TSM コマンドなどの従来の Tableau Server ツールやプロセスでフォールバックして、Tableau Server を操作することができます。

標準の Tableau Server インスタンスのバックアップと復元の実行方法の詳細については、Tableau Server の完全なバックアップと復元を実行しますを参照してください。

Tableau Server コンテナーでのバックアップ

1. Tableau Server in a Container のバージョン A 内にあるシェルを開きます。リポジトリバックアップ、トポロジ、および構成バックアップ ファイルを作成します。

   ```
   docker exec -it my-server bash

   # Just providing filename automatically produces the backup
   file at /var/opt/tableau/tableau_
   server/data/tabsvc/files/backups/
   tsm maintenance backup -f <repository-backup>.tsbak -d

   # Any filepath where current user(UNPRIVILEGED USER) can write.
   tsm settings export -f /var/opt/tableau/tableau_
   server/data/tabsvc/files/backups/<topology-conf-backup>.json
   ```

2. 前のステップで作成したファイルをホストマシンにコピーします。ファイルのパーミッションを変更して、両方のファイルに "すべて読み取り" のパーミッション セットを設定します。

   ```
   docker cp my-server:/var/opt/tableau/tableau_
   server/data/tabsvc/files/backups/<repository-backup>.tsbak
   ./<repository-backup>.tsbak
   docker cp my-server:/var/opt/tableau/tableau_
   server/data/tabsvc/files/backups/<topology-conf-backup>.json
   ./<topology-conf-backup>.json
   ```

```
chmod a+r ./<repository-backup>.tsbak ./<topology-conf-
backup>.json
```

3. バックアップ アーティファクトを安全な場所に保存します。必要に応じて、以下の復元ステップに従ってください。

Tableau Server コンテナー内で復元する

サポートされている Tableau Server バージョン (コンテナーを使用する場合、およびコンテナーを使用しない場合) からのバックアップは、Tableau Server コンテナー内で復元できます。

前提条件

- Tableau Server のバックアップ ファイル。
- 構成情報とトポロジー情報の両方を含む構成 json ファイル。
- アイデンティティストア情報を含む構成 JSON ファイル。
- **注意**: "すべて読み取り" のパーミッション セットを含むように、バックアップ ファイルを変更する必要があります。バックアップ ファイルは通常、ファイルを作成したユーザーにロックされており、このユーザーはたいていコンテナーで実行している Tableau ユーザーとは異なります。

```
docker run \
-v <full-path-to-backup-file>:/docker/config/backup/backup-
file.tsbak \
-v <full-path-to-config-only-file>:/docker/config/config.json:ro \
-v <full-path-to-identity-store-config-only-
file>:/docker/config/identity-store-config.json \
-e LICENSE_KEY=<key> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

注:

- マルチノード システムを復元する場合、復元の自動化を機能させるには、他のノードも起動する必要があります。詳細については、このドキュメントのマルチノードの Tableau Server in a Container セクションを参照してください。バックアップ ファイル、バックアップ構成 ファイル、およびライセンスが必要となるのは、最初のノードのみです。
- バックアップ ファイルは、コンテナーを初めて実行するときにのみ提供する必要があります。サーバーが初期化されたら、バックアップ ファイルにマウントし続ける必要はありません。

Tableau Server から Tableau Server in a Container に移行する

標準の Tableau Server インストールから Tableau Server in a Container に移行するには、バックアップと復元の手法を使用する必要があります。サポートされている Tableau Server バージョン (コンテナーを使用する場合、およびコンテナーを使用しない場合) からのバックアップは、Tableau Server コンテナー内で復元できます。詳細については、上記の Tableau Server コンテナー内で復元するのセクションを参照してください。

Tableau Server バージョンをアップグレードする

Tableau Server をアップグレードする方法は 2 つあります。このセクションにリストされているアップグレード イメージ メソッドは、推奨されるソリューションです。ただし、フォールバックとして、バックアップ/復元を使用して Tableau Server をアップグレードすることもできます。

アップグレード イメージ メソッドによるアップグレード

アップグレード イメージは、Tableau Server in a Container のセットアップ ツールからの `build-upgrade-image` スクリプトを使用して構築できる Docker イメージです。このイメージの目的は、現在実行している Tableau Server in a Container をアップグレードすることだけです。

> **重要:** アップグレード イメージ メソッドを使用している場合でも、上記のバックアップと復元の説明に従って Tableau Server を完全にバックアップすることを強くお勧めします。これにより、何らかの理由でアップグレードに失敗した場合の選択肢が得られます。

アップグレードを行うには、以下のステップに従ってください。

1. `build-upgrade-image` スクリプトを使用してアップグレード イメージを作成します。このコンテナーを構築するには、新しいバージョンの Tableau Server rpm が必要です。
2. 現在 Tableau Server を実行しているコンテナーをシャットダウンします。
3. 前のステップでシャットダウンしたコンテナーから同じデータ ディレクトリをマウントして、アップグレード イメージ を開始します。
4. アップグレード プロセスにはしばらく時間がかかりますが、Tableau Server がアップグレードされます。アップグレード プロセスの更新については、Docker ログを確認してください。コンテナーは、アップグレード プロセス後にシャットダウンします。

5. 新しいバージョンの Tableau Server in a Container を新たに起動します。前のステップから同じディレクトリをマウントします。

例:

Tableau Server を実行している Tableau Server in a Container があるとします。この例の前提条件は以下のとおりです。

- 貴重なデータがあるため、アップグレードプロセス中にそのデータを失いたくありません。データディレクトリは、コンテナーの外部で永続化する必要があります。
- コンテナーの名前は my-server です。Docker イメージの名前は tableau-server: version A です。
- my-server で現在使用しているサーバー バージョンは、バージョン A です。
- それをバージョン B にアップグレードしたいと考えています。

1. バージョン B の Tableau Server rpm を取得します。アップグレードイメージを作成します。

```
# For all the options available in the script
./build-upgrade-image -h

# Frequently used command to create a upgrade-image
./build-upgrade-image --installer=<path to the tableau server
version B> -i tableau-server:versionA -o tableau-server-
upgrade:versionAB
```

2. my-server コンテナーを停止します。

```
docker stop my-server -t 120
```

3. 新しく作成されたイメージ tableau-server-upgrade:version AB を開始します。先ほど停止したコンテナーから同じデータディレクトリをマウントします。コンテナーはバージョン B へのアップグレードプロセスを開始します。

```
docker run --name my-upgrade-server \
-v <data-dir mount from previous step>:/var/opt/tableau \
...
tableau-server-upgrade:versionAB
```

4. アップグレードが完了すると、コンテナーは停止します。Docker ログでアップグレードプロセスのログを確認し、アップグレードプロセスが成功したことを確認します。Docker コンテナーの終了コードをチェックして、アップグレードプロセスが正常に完了したことを確認することもできます。

```
# The log file /var/opt/tableau/tableau_server/logs/upgrade-
console.log is created after 3-4 mins into the start of upgrade
container. When the upgrade completes successfully, "upgrade is
complete" log will be # seen.
docker logs my-upgrade-server
...
...
Verifying licensing state.
Tableau Server has been upgraded to version near.20.0801.1050.
>> The upgraded Tableau binary directory will be added to PATH
for new shells. To get the
>> updated path, either start a new session, or for bash users
run:
>> source /etc/profile.d/tableau_server.sh
Starting service...
Starting service...
Job id is '12', timeout is 30 minutes.
Service was started successfully.
Status: RUNNING
Tableau Server is Running
upgrade is complete
```

5. my-upgrade-server コンテナーを停止します。Tableau Server in a Container のイメージの新しいバージョン B を起動し、停止した my-upgrade-server コンテナーからデータディレクトリをマウントします

```
# Stop the server.
docker stop my-upgrade-server -t 120


# Run the new version Hu
```

```
docker run --name my-upgraded-server \
-v <data-dir mount from previous step>:/var/opt/tableau \
...
...
tableau-server:versionB
```

バックアップ復元メソッドによるアップグレード

このドキュメントのバックアップと復元セクションのステップに従います。バックアップと復元の操作をアップグレード操作に変更するために必要となる調整は、新しいバージョンの Tableau Server でバックアップを復元することだけです。

マルチノードの Tableau Server in a Container

マルチノードの Tableau Server in a Container とは、複数のノードに分散された Tableau Server の単一の展開を指します。このコンテキストのマルチノードは、容量を増やしたり、処理能力を高めたりするために他のノードで特定のプロセスを実行できる Tableau Server マルチノードと同じものです。これは、各コンテナーが独立したサーバーで、独自のデータが含まれる、複数の Tableau Server in a Container を個別に起動する場合とは異なります。

マルチノードの Tableau Server in a Container は、コンテナーを使用しない Tableau Server の分散インストールとほとんど同じように機能し、同じ基本メカニズムを使用します。コンテナーを使用しない Tableau Server の分散インストールのセットアップの概要については、分散型高可用性 Tableau Server インストールを参照してください。

以下に例を示します。

マルチノードの基本的な使用例

最初のノード

オプション 1: サーバー構成 (`CONFIG_FILE`) でマルチノードトポロジが指定される場合にこれを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-v <full-path-to-config-file>:/docker/config/config.json:ro \
```

```
-e LICENSE_KEY=<key> \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

オプション 2: サーバー構成でマルチ ノードトポロジが指定されていない場合でも、マルチノード展開が必要な場合は、これを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e LICENSE_KEY=<key> -e ALWAYS_WRITE_BOOTSTRAP_FILE=1 \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \

--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

追加ノード

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e BOOTSTRAP_INSTALL=1 \
-p 8080:8080 -p 8800-9000:8800-9000 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

ライセンス発行ポートとTSM ポートの公開

ワーカー ノードがプライマリインスタンスと通信するには、追加のポートを開く必要があります。次のポート範囲で、プライマリTableau Server in a Container インスタンス上の他のノードからのトラフィックを許可する必要があります。

```
Service Ports: 8800-9000
Postgres Port: 8060
Licensing Ports: 27000-27010
```

開くポートの数に注意してください。Docker で 1000 ポートを公開するとDocker イメージのパフォーマンスと起動時間に悪影響を与える可能性があるため、Tableau Server の既定の 8000 ~ 9000 ポート範囲ではなく、200 ポート (8800 ~ 9000) のみを公開することをお勧めします。

Tableau Server トポロジの複雑さに応じて、より小さいポート範囲または大きなポート範囲を使用できます。一般的に、100 ポート未満のポートを公開すると、クラスタが特定のサービスでやり取りができなくなるサービス リスクがあるため、推奨されていません。独自のポート範囲を指定する場合は、ポート 8850 を公開してください (これは 8800-9000 に暗黙的に含まれています)。ポート範囲は、環境変数 `PORT_RANGE_MIN` および `PORT_RANGE_MAX` を設定して指定します。

追加ノードではサービス ポート範囲 (8800 ~ 9000) を公開する必要がありますが、ライセンス発行ポートの範囲は公開する必要はありません。これらのポート範囲は、Tableau Server のプロセス間の通信を許可するだけのために使用することに注意してください。これらのポートは、同じマルチノード クラスタの Tableau Server in a Container を実行しているコンピューター以外のユーザーまたは他のマシンには公開しないでください。

これらのポート規則は、Tableau Server のファイアウォール ドキュメントと一致しています。詳細については、ローカル ファイアウォールの構成を参照してください。

ホスト名の解決

Tableau Server ではホスト名の動的な変更が処理されないため、Tableau Server in a Container の複数のノードは一貫性のあるホスト名で実行する必要があります。Tableau Server マルチノードを実行する場合、それらのノードは相互に通信する必要があります。Tableau Server ノードは、マルチノードの Tableau Server in a Container を使用するように構成されているホスト名を使用して、相互に接続しようとします。たとえば、ノードを "initial" というホスト名で実行すると、追加ノードは "initial" というホストにトラフィックを送信しようとします。ホスト名を他のイメージに解決するようにイメージを構成するには複数の方法があります。各コンテナー内の `/etc/hosts` ファイルを使用して、任意のコンテナーのホスト名 ("initial") を他のコンテナーを実際に実行している IP アドレスにマッピングします。

追加ノードをブートストラップする

クラスタの一部として実行する最初の Tableau Server コンテナーは、後続の追加ノードをクラスタに結合するために使用する必要がある bootstrap ファイルを生成します。追加ノードがクラスタのトポロジーに登録されたら、そこで実行する Tableau Server プロセスの割り当てを開始できます。このプロセスは完全に自動化できます。マルチノード トポロジを指定する Tableau Server 構成ファイルを提供した場合 (通常、構成ファイルを `CONFIG_FILE` で指定されるファイル パスにマウントする

ことによって提供されます。既定のパス: `/docker/config/config.json`)、最初のノードは、すべての追加ノードが登録されるまで自動的に待機します。追加ノードが登録されると、マルチノードトポロジはクラスタ全体に適用されます。

Tableau Server in a Container の最初のノードで Tableau Server を完全に実行したら、追加ノードの bootstrap ファイルを生成できます。

```
docker exec -it <container-name> tsm topology nodes get-bootstrap-file -f $BOOTSTRAP_FILE
```

このコマンドは `ALWAYS_WRITE_BOOTSTRAP_FILE` の値を 1 に設定すると自動的に呼び出されます。

セキュリティに関する考慮事項

bootstrap ファイルには、最初のノードとの TSM セッションを確立するためのサーバー シークレットが含まれています。つまり、悪意のあるユーザーがファイルを取得したら、サーバーに TSM コマンドを一定期間送信できるということです。ファイル自体には、サーバー構成の秘密の解読を可能にするデータも含まれています。このファイルは、機密性の高いファイルとして扱う必要があり、マルチノード展開の確立に直接関連するサービスおよびシステムのみがアクセス可能である必要があります。

Bootstrap の有効期限

bootstrap ファイルは、有効期限が 2 時間のセッションを実行します。そのウィンドウでは、追加ノードは、追加ノードとして結合するために最初のノードに認証資格情報を提供する必要はありません。セッションの有効期限が切れた後に bootstrap ファイルを使用することは可能ですが、最初のノードに認証資格情報を提供する必要があります。

bootstrap ファイルを転送する

bootstrap ファイルは、ワーカー ノードの Tableau Server in a Container で使用する必要があります。bootstrap ファイルは、この展開のワーカー ノードとして、必要となる他のすべてのノードの Tableau Server in a Container と共有する必要があります。共有は、さまざまな方法で行うことができます。

セキュリティで保護されたネットワーク経由でファイルを転送する

最初のノードの自動化の一部では、追加ノードに直接ファイルを送信する必要があります。その際、いくつかの安全なファイル転送クライアント/ツールを使用します。これは、最初のノードの有効期間中に複数の bootstrap ファイルが生成される可能性がある(後で追加ノードを追加する可能性がある) シナリオでより役に立ちます。

ネットワーク ファイル マウントを使用する

特定の配置内のすべてのコンテナで共有されるネットワーク ファイル マウントも、別のオプションです。

その他

最終目標は、あるコンテナーで作成されたファイルを安全に転送し、それを他のコンテナーの特定のセットに転送することです。したがって、これを達成し、安全である方法があれば十分です。

追加ノードを起動する

Tableau Server in a Container の追加ノードを起動するには、環境変数 `BOOTSTRAP_INSTALL` を 1 に設定してコンテナーを起動します。

これにより、環境変数 `BOOTSTRAP_FILE` (これも設定可能です) で指定されたパスに bootstrap ファイルが存在するまでスリープするように Tableau Server in a Container に指示が送られます。既定のファイル パスを表示するには、環境変数テーブルを参照してください。つまり、Tableau Server in a Container のイメージを [additional node mode (追加ノードのモード)] で実行すると、コンテナーでは、`supervisord` や、bootstrap ファイルが存在するかどうかを 5 秒ごとに確認する pid 1 として実行している bash スクリプト以外のプロセスは開始されません。ファイルが存在すると、Tableau Server in a Container は追加ノードとしての初期化に進みます。

追加ノードを構成する

特定のトポロジを実行するように追加ノードを構成すると、標準の Tableau Server の展開の場合と同じように機能します。また、同じ要件も伴うため、ノードに新しいプロセスを追加する際に、クラスタ全体の再起動が必要になる場合があります。詳細については、ノードの構成を参照してください。

Tableau Server の機能に関する考慮事項

コンテナーでは、Tableau Server の一部の機能の動作が異なります。このセクションでは、コンテナー環境で特別な考慮事項または異なる考慮事項がある特定の機能について説明します。

Active Directory

AD ドメイン コントローラーを設定する

Tableau Server Web ページおよびサイトのアイデンティティストアとして Active Directory を使用することを計画している場合は、追加の考慮事項があります。Linux 環境で実行している Tableau Server は、IP サブネットを調べることによって、通信する AD ドメイン コントローラーを動的に決定します。コンテナーには任意の IP アドレスを割り当てることができます。この場合、Tableau Server は必ずしもその IP アドレスを使用して適切なドメイン コントローラーを見つけることができるわけではありません。そのため、Tableau Server が通信する特定のドメイン コントローラー/ホスト名を構成する必要がある場合があります。これを行うには、次のステップに従ってください。

1. Tableau Server で使用するドメイン コントローラーを決定し、ホスト名を取得します。
2. 標準の Tableau Server 管理者構成オプションを使用して、構成キー `wgserver.domain.ldap.hostname` をホスト名に設定します。
    - json 構成ファイル `CONFIG_FILE` に値を設定します。
    - TSM 構成コマンドを使用します

      ```
      tsm configuration set -k wgserver.domain.ldap.hostname -v
      <hostname>
      ```

AD 証明書を Tableau Server キーストアにインポートする

既定では、Tableau Server in a container は、シンプル バインドが使用されるたびに StartTLS を介して AD と通信します。したがって、コンテナーをこの構成で実行する場合は、AD サーバー証明書を Tableau Server キーストアにインポートする必要があります。そうしなければ、サーバーの初期化が失敗します。これを行うには、次のステップに従ってください。

1. pre-init-command スクリプトを作成します (初期化前のスクリプトセクションを確認してください)。次の行を追加して、AD 証明書を Tableau Server キーストアに追加します。

   ```
   ${INSTALL_DIR}/packages/repository.${SERVICE_VERSION}/jre/bin -importcert -noprompt -alias startTlsCert -file <mounted-certificate-path> -storetype JKS -storepass changeit -keystore ${DATA_DIR}/config/tableauservicesmanagerca.jks
   ```

2. pre-init-command スクリプトの -file パラメーターに提供されたファイル パスに AD サーバー証明書をマウントします。

または、StartTLS を介して AD と通信する既定の設定を無効にすることもできます。wgserver.domain.ldap.starttls.enabled を false に設定し、StartTLS を無効にします。ただし、これは推奨されていません。

展開構成の例

Docker

Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key>
-p 8080:8080
-d <Tableau Server in a Container image ID or tag>
```

自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key> \
-e TABLEAU_USERNAME=<myadmin> \
-e TABLEAU_PASSWORD_FILE=/etc/tableau-admin-secret \
-v <full-path-to-pw-file>:/etc/tableau-admin-secret \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

TSM 専用モード

```
docker run \
-e TSM_ONLY=1 \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

マルチノードの基本的な使用例

最初のノード

オプション 1: サーバー構成 (CONFIG_FILE) でマルチノード トポロジが指定される場合にこれを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-v <full-path-to-config-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

オプション 2: サーバー構成でマルチ ノードトポロジが指定されていない場合でも、マルチノード展開が必要な場合は、これを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e LICENSE_KEY=<key> -e ALWAYS_WRITE_BOOTSTRAP_FILE=1 \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

追加ノード

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e BOOTSTRAP_INSTALL=1 \
-p 8080:8080 -p 8800-9000:8800-9000 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

データ使用の外部化

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
```

```
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Init コンテナーの基本的な使用例

Init コンテナー

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
-e INIT_CONTAINER=1 \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

コンテナーの実行

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

シングルノードバックアップからの基本的な復元

```
docker run \
-v <full-path-to-backup-file>:/docker/config/backup/backup-
file.tsbak \
-v <full-path-to-config-only-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Docker の構成

```
version: '3.2'
services:
    tableau-server:
         hostname: localhost
         volumes:
              - <your-tsm-command-file>:/docker/config/tsm-
commands:ro
```

```
      - <your-config-file >:/docker/config/config.json:ro
ports:
      - "8080:8080"
image: ${IMAGE_NAME}
environment:
      - LICENSE_KEY=<license-key>
```

Tableau Server in a Container - トラブルシューティング

概要

Tableau Server in a Container は、Tableau 初のコンテナーベースのサーバー製品です。Tableau Server in a Container は、Linux Docker コンテナ内で実行されるオールインワンの Tableau Server インスタンスです。つまり、Tableau Server in a Container のイメージは、自己完結型の Tableau Server アプリケーション全体を実行する Docker イメージです。Tableau Server in a Container は、コンテナーベースの環境で Tableau Server の実行をサポートする数多くのステップの中の最初のステップとなります。Tableau Server in a Container の概念は、Tableau Server が事前にインストールされた VM のようなものと考えると、簡単に理解することができます。イメージは UBI 8 イメージに基づいており、コンテナー内で (`systemd` の代わりに) `supervisord` を実行します。コンテナーが `supervisord` を開始すると、直ちに Tableau Server を初期化して起動しようとします。ここにあるドキュメントの多くは、Docker 環境に合わせて Tableau Server を実行できるように、構成を提供して自動化の活用方法を説明するためのものです。

Tableau Server in a Container のイメージのセットアップ ツールを使用すると、カスタム パッケージとアーティファクトを含むようにコンテナー イメージを作成してカスタマイズすることができます。このツールの主な機能の 1 つは、コンテナー イメージを構築し、カスタム データ コネクタをインストールすることです。

概念実証シナリオで Tableau Server in a Container をすばやくテストするには、Tableau Server in a Container のクイック スタートを参照してください。

制限事項

- Tableau Server in a Container では、サーバーATR を使用したライセンス認証のみがサポートされます。これを行うには、コンテナーからインターネットにアクセスできる必要があります。したがって、エアギャップ環境におけるオフラインのライセンス認証はできません。
- Tableau Server in a Container では現在、Resource Monitoring Tool (RMT) エージェントはサポートされていません。
- Kerberos は、Tableau Server in a Container ではサポートされていません

トラブルシューティング

Tableau Server の実行中に問題が発生した場合、解決策を追求するための方法がいくつかあります。このセクションでは、ログの場所やログの意味など、Tableau Server のトラブルシューティングに関する一般的なアドバイスについて説明します。また、いくつかの特定の既知のシナリオと軽減パスについても説明します。

Tableau Support を使用して問題をデバッグしている場合は、以下を提供すると役立つ場合があります。

- Tableau Server ログ (これらのログの収集については、以下で説明します)。
- Docker コンテナー `stdout` ログ。
- Tableau Server の Dockerfile (カスタマイズが行われている場合)。

- 以下を含む展開構成:
    - `Kubeconfig` (または同等の展開構成)。
    - Tableau Server コンテナーを構成する静的構成ファイル。

インストールと初期化の失敗

Tableau Server を初めて初期化する場合、またはコンテナー内で新規インストールを実行した場合、サーバーはコンテナーを再起動するだけでは回復しません。インストールを試みるときは必ず、クリーンなデータディレクトリを使用する必要があります。これにより、以前のコンテナーの実行から永続的なボリューム データが削除される場合もあります。これを行う場合は、デバッグに役立つ可能性のあるログと情報を必ず保存してください。

失敗したインストールをデバッグする

Tableau Server コンテナーは、インストールの失敗が発生すると終了するように設計されています。このパターンによって、インストールの失敗がいつ発生したのかを特定する作業を簡単に自動化することができます。しかし、コンテナーが終了すると、調査可能なランタイム状態が残されないため、デバッグが困難になる可能性があります。初期化中に失敗した実行中のコンテナー内でデバッグ セッションを実行する場合は、次のステップに従います。

1. コンテナー展開で新しい Tableau Server を準備します。
2. コンテナーを環境変数 `TSM_ONLY=1` を使用して実行するように構成します。環境変数 `TSM_ONLY=1` は、TSM のみを初期化するように Tableau Server に指示します。これは、コンテナーを使用していない標準インストールの `initialize-tsm` スクリプトを実行している場合と同じ操作です。
3. Tableau Server container を実行します。
4. コンテナー内でシェルを開きます。
5. Tableau Server が初期化されていなくても、TSM コマンドを実行できるようになりました。初期化中に通常行われる自動化を再開するには、`tsm-commands` スクリプト: `"${DOCKER_CONFIG}"/config/tsm-commands` を実行します

Tableau Support とKubernetes

Tableau Server in a Container は、Kubernetes を使用して実行できますが、必須条件ではありません。今後、大半のお客様が、Kubernetes または関連するマネージド クラウド環境 (EKS、AKS、または GKS) のいずれかを使用して、Tableau Server in a Container を実行および管理していくことになるでしょう。

Kubernetes では、実行やデバッグするための環境が複雑になる可能性があり、多くの場合、個々の企業のインフラストラクチャとセットアップに対する依存関係が含まれています。このため、Tableau Support は、Tableau Server in a Container の実行に関連する Kubernetes (またはインフラストラクチャの展開) の問題を解決するための支援ができません。ただし、Docker コンテナー内の Tableau Server の実行はサポートされています。したがって、Kubernetes を使用して Tableau Server in a Container 実行する際に問題が発生した場合、Tableau Support は、Docker コンテナー自体が正しく機能しているかの検証のみを行います。

Kubernetes を使用して Tableau Server in a Container を実行する方法については、次の Github サイトを参照してください https://github.com/tableau/tableau-server-in-kubernetes。

コンテナ内の Tableau Serverのタイムゾーンの変更

コンテナ内の Tableau Server で特定のタイムゾーンを使用するには、/etc/localtime から目的のタイムゾーン ファイルへのシンボリック リンクを作成することで設定できます。

```
ln -sf /usr/share/zoneinfo/Asia/Tokyo /etc/localtime
```

```
export TZ="Asia/Tokyo"
```

管理プロセスを含む Tableau Server を再起動して、新しいタイムゾーンを認識させます。

```
$ tsm configuration set -k service.log.timezone -v "Asia/Tokyo"
$ tsm pending-changes apply
$ cd /opt/tableau/tableau_server/packages/scripts.{version}/
$ ./stop-administrative-services
$ ./start-administrative-services
```

tsm status コマンドが「STOPPED」ステータスを返すまで待ってから、Tableau を開始します。

```
$ tsm start
```

ログ

ログは、Tableau Server の問題を見つけ、理解し、解決するために不可欠なリソースです。サポート チームはこれらを使用することで、発生した問題の根本原因を見つけることができます。ログは、独自のデバッグやトラブルシューティングにも役立つ場合があります。

すべてのログを抽出する

デバッグをさらに進めたり、サポート チームに送信したりするためにすべてのログを抽出する必要がある場合は、いくつかの方法でこの情報を取得することができます。

Ziplogs

TSM は、関連するすべてのサーバー ログを含む圧縮アーカイブを作成できます。これをトリガーするには、tsm maintenance ziplogs コマンドを実行します。コマンドが完了すると、ログ アーカイ

ブのファイル パスが報告されます。アーカイブは、状況に最も適したファイルの転送方法を使用してコピーする必要があります。Ziplogs の詳細については、tsm maintenance ziplogsを参照してください。

コンテナ内で実行されるコマンドの例:

```
tsm maintenance ziplogs
```

手動の tar コマンド

ziplogs コマンドを実行できない場合、たとえば、サーバーが一貫性のある状態に到達できない場合でも、コンテナー内で tar コマンドを実行するとログを取得できます。アーカイブは、状況に最も適したファイルの転送方法を使用してコピーする必要があります。

コンテナ内で実行されるコマンドの例 (tar をコンテナのデータディレクトリの一時ディレクトリに書き込みます):

```
tar -zcvf /var/opt/tableau/tableau_server/temp/<archive_
name>.tar.gz \
/var/opt/tableau/tableau_server/data/tabsvc/logs/. \
/var/opt/tableau/tableau_server/supervisord/ \
/var/opt/tableau/tableau_server/data/tabsvc/config/ \
/docker/.metadata.conf \
--exclude='*keystores' --exclude='*.jks' --exclude='*.tks' \
--exclude='*asset_keys.yml' --exclude='*.ks' --exclude='*.ts' \
--exclude='*.crt' --exclude='*cacerts' --exclude='*.key'
```

ログの操作とデバッグのヒント

Tableau Server のほぼすべての問題を診断できる一般的なステップがあります。サーバー ログを確認することを検討している場合は、サーバー ライフサイクルのどこでエラーが発生したかに応じて、検索する情報を分類することができます。

コンテナーを起動する (初期/インストール)

コンテナーがすぐにクラッシュしたり、インストールや初期化に失敗したりする場合は、次のリソースを確認してください。

コンテナーの stdout

Dockerコンテナーの `stdout` を調べます。stdout にはたいてい、コンテナー オーケストレーション システム (Kubernetes など) によって収集されたコンテナー出力を確認することでアクセスできます。Tableau Server はコンテナ内で実行されるマルチプロセス システムであるため、起動時に壊滅的な障害が発生しない限り、stdout はそれほど有用ではなく、問題の根本原因も報告されません。Tableau Server ログをさらに掘り下げる前に、問題が発生しているコンテナーの `stdout` をチェックすることをお勧めします。

例:

```
docker logs <container-name>
```

Tableau Server コンテナーの起動ログ

Tableau Server Container の起動ログは、Tableau Server の初期化、構成、および起動を行う自動化からの出力を取得します。コンテナーを初めて起動または実行しているときに問題が発生していることがわかった場合、最初に以下のログを確認します。

```
/var/opt/tableau/tableau_server/supervisord/run-tableau-server.log
```

ログの下部をチェックして、報告された障害があるかどうかを確認します。ログのエラー報告によって、問題がすぐに明らかになる場合もあります。ログでエラーが明らかになっていない場合は、根本的な原因がステージ固有またはサービス固有のログ ファイルにのみ表示されている可能性があります。このような可能性に該当するのは、以下にリストされているログです。

Tableau Server インストール ログ

起動ログに、TSM の初期化ステージを処理する自動化に問題があったことが示されている場合は、次のログを確認してください。

```
/var/opt/tableau/tableau_server/logs/app-install.log
```

Tableau Server コントローラー ログ

起動ログにサーバー ステージ (CLI のみ) の初期化と起動に問題があったことが示されている場合は、tabadmincontroller サービス ログを確認してください。

```
/var/opt/tableau/tableau_
server/data/tabsvc/logs/tabadmincontroller/tabadmincontroller_
node1-0.log
```

このログ ファイルは、tabadmincontroller と呼ばれる特定のサービス向けのものです。Tabadmincontroller は、サーバーの初期化および起動機能の調整を行います。このログは複雑で冗長になる可能性があります。このログ ファイルのエラーは、まだ根本的な原因を示していない可能性があります。エラーは、tabadmincontroller が特定のタスクを完了するために依存しているサービスが原因で発生する場合があります。詳細については、以下のサーバー ランタイムのセクションを確認してください。

サービスログ - サーバー ランタイム

Tableau Server で通常の実行時に問題が発生した場合、またはサービスがタスクを完了できないか、ダウンしている場合は、サービス ログで詳細を確認できます。Tableau Server の一部として実行されているすべてのサービスには、サービス ログ ファイルがあります。調べるサービスがわかっている場合は、次の一般的なディレクトリでそのサービスのログを見つけることができます。

```
/var/opt/tableau/tableau_server/data/tabsvc/logs/<service_name>
```

ファイル パスの引数 <service_name> にサービスの名前を入力します。どのサービスでも、複数のタイプのログ ファイルを書き込むことができます。また、同じサービスを複数 (複数のインスタンス) 実行している場合は、すべてのサービス ログが同じサービス ディレクトリに書き込まれます。

一般的なサービス固有のログ ファイルの分類

この表では、Tableau Server サービスの最も一般的なサービス ログファイル名、タイプ、および説明について説明します。[Failure types (障害のタイプ)] 列は、特定の障害シナリオで役立つ可能性が高いログ ファイルを示します。

名前	ファイル名の形式	説明	障害のタイプ	例
Control-App	control_ <service_ name>_	サービスのインストールと構成を行うcontrol-app プロセスの情報が	インストール、構成、ステータス	control_ backgrounder_ node1-0.log

名前	ファイル名の形式	説明	障害のタイプ	例
	`<node_id>-<instance_id>.log`	含まれています。これは多くの場合、サービス関連の情報が書き込まれる最初のログです。サービスのインストールと構成の失敗については、最初に以下を参照してください。		
サービス ログ	`<service_name>_<node_id>-<instance_id>.log`	実行中のサービスのプライマリログ。多くの場合、このログには、spring/java アプリケーション レイヤーからの出力が含まれています。	開始、ランタイム、ステータス	`backgrounder_node1-1.log`
Stdout ログ	`stdout_<service_name>_<instance_id>.log`	サービスの stdout 出力が含まれています。ほとんどのサービスは stdout にそれほど多くのコンテンツを出力しません。代わりに、プライマリログに書き込みます。このログには、サービスの終了時に役立つ情報が含まれている場合があります。	起動、停止	`stdout_backgrounder_0.log`
Native API ログ	`nativeapi_<service_name>_<instance_id>.txt`	一部のサービスは、ネイティブ コード レイヤーを実行します。このログは、アプリケーションのランタイムのその部分を取得します。	ライセンス発行、開始、ランタイム、ステータス	`nativeapi_backgrounder_1-1_2021_05_10_00_00_00.txt`

名前	ファイル名の形式	説明	障害のタイプ	例
Tomcat ログ	`tomcat_<service_name>_<node_id>-<instance_id>.log`	これは、Tomcat コンテナ内で実行されて Tomcat ログを含むサービスのみに使用するログです。このログから、サービス障害に関する情報が提供されることはほとんどありません。これは、ネットワークの問題をデバッグする際に役に立ちます。	ネットワーク、開始	`tomcat_backgrounder_node1-0.2021-05-10.log`

停止したコンテナー

コンテナーが停止している場合、またはコマンドの実行が難しい場合でも、サーバーのデータディレクトリがマウントされたボリュームに外部化されていれば、ログを確認できます。それ以外の場合は、コンテナーの `stdout` は、コンテナーのオーケストレーション システムで調査できますが、多くの場合、根本的な原因は含まれていません。

認証プロパティの設定に失敗しました

アイデンティティストアを最初に設定せずに Tableau Server で認証プロパティを設定することに問題があるようです。この問題を回避するには、初期化前のフックにアイデンティティストアを設定します。

1. Tableau Server イメージ ビルド ツールの customer-files ディレクトリで `./customer-files/pre_init_command` というファイルを作成し、以下を含むように編集します。

   ```
   #!/bin/bash
   tsm configuration set -k wgserver.authenticate -v local --force-keys
   ```

2. スクリプトを実行可能に設定します。

```
chmod +x ./customer-files/pre_init_command
```

3. イメージを構築して実行します。

新しい起動時のエラー(Tableau Server が起動しない理由など)

- Tableau Server の初期化または起動に関する問題が発生する場合は、問題の発見に役立つトラブルシューティングオプションが多数あります。
- コンテナーをまったく起動できない場合は、`docker logs <container-name>` コマンドを使用して PID 1 プロセスから stdout をチェックします。
- コンテナーが実行中で、Tableau Server が初期化されていないか、正しく実行されていない場合、エラーを確認する 2 番目の場所は次のファイルです。

```
${DATA_DIR}/supervisord/run-tableau-server.log
```

例:

```
docker exec -it <container-name> bash -c 'cat $DATA_
DIR/supervisord/run-tableau-server.log'
```

このログ ファイルには、Tableau Server の起動を処理する、さらに、コンテナーに提供した任意のセットアップスクリプトまたはカスタム設定を実行する、tableau コンテナー初期化サービスによって調整されたすべてのイベントが含まれます。ほとんどのスタートアップ エラーは、ここで問題を報告します。エラーが TSM または Tableau Server プロセスに関連している場合は、詳細な情報を確認する別のログ ファイルが提案される場合があります。

既存のデータを使用してコンテナーを再起動または起動する際の障害

サーバーが Postgres (または他のプロセス) を起動しない

データがコンテナーの外部に永続化され、その古いデータを使用して別の Tableau Server in a Container イメージ のンスタンスを開始する場合、新しいコンテナーの内部ホスト名が、永続化されたデータを初期化したコンテナーのホスト名と一致する必要があることに注意してください。Tableau Server では動的ホスト名の変更をうまく処理しないため、内部ホスト名が異なる新しいコンテナーを起動すると、そのシナリオが実際に発生します。

この問題を解決するには、単に、コンテナーのホスト名が、そのデータで以前に実行されていたコンテナーと同じ値に設定されていることを確認します。これはマルチノードと混同しないように、ワーカーは互いに異なるホスト名を持つことができます(そしておそらく必要があります)。重要なのは、特

定のコンテナが再起動されるか、後続のコンテナが先行コンテナと同じホスト名を持つ必要がある場合です。

展開構成の例

Docker

Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key>
-p 8080:8080
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key> \
-e TABLEAU_USERNAME=<myadmin> \
-e TABLEAU_PASSWORD_FILE=/etc/tableau-admin-secret \
-v <full-path-to-pw-file>:/etc/tableau-admin-secret \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

TSM 専用モード

```
docker run \
-e TSM_ONLY=1 \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

マルチノードの基本的な使用例

最初のノード

オプション 1: サーバー構成 (`CONFIG_FILE`) でマルチノードトポロジが指定される場合にこれを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
```

```
-v <full-path-to-config-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<name-of-host-machine> \
-d <Tableau Server in a Container image ID or tag>
```

オプション 2: サーバー構成でマルチ ノードトポロジが指定されていない場合でも、マルチノード展開が必要な場合は、これを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e LICENSE_KEY=<key> -e ALWAYS_WRITE_BOOTSTRAP_FILE=1 \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<name-of-host-machine> \
-d <Tableau Server in a Container image ID or tag>
```

追加ノード

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e BOOTSTRAP_INSTALL=1 \
-p 8080:8080 -p 8800-9000:8800-9000 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

データ使用の外部化

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
---hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Init コンテナーの基本的な使用例

Init コンテナー

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
```

```
-e LICENSE_KEY=<key> \
-e INIT_CONTAINER=1 \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

コンテナーの実行

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

シングルノードバックアップからの基本的な復元

```
docker run \
-v <full-path-to-backup-file>:/docker/config/backup/backup-
file.tsbak \
-v <full-path-to-config-only-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Docker の構成

```
version: '3.2'
services:
    tableau-server:
         hostname: localhost
         volumes:
              - <your-tsm-command-file>:/docker/config/tsm-
commands:ro
              - <your-config-file >:/docker/config/config.json:ro
         ports:
              - "8080:8080"
         image: ${IMAGE_NAME}
         environment:
              - LICENSE_KEY=<license-key>
```

Tableau Server in a Container のクイック スタート

概要

Tableau Server in a Container は、Tableau 初のコンテナーベースのサーバー製品です。Tableau Server in a Container は、Linux Docker コンテナ内で実行されるオールインワンの Tableau Server インスタンスです。つまり、Tableau Server in a Container のイメージは、自己完結型の Tableau Server アプリケーション全体を実行する Docker イメージです。Tableau Server in a Container は、コンテナーベースの環境で Tableau Server の実行をサポートする数多くのステップの中の最初のステップとなります。Tableau Server in a Container の概念は、Tableau Server が事前にインストールされた VM のようなものと考えると、簡単に理解することができます。イメージは UBI 8 イメージに基づいており、コンテナー内で (`systemd` の代わりに) `supervisord` を実行します。コンテナーが `supervisord` を開始すると、直ちに Tableau Server を初期化して起動しようとします。ここにあるドキュメントの多くは、Docker 環境に合わせて Tableau Server を実行できるように、構成を提供して自動化の活用方法を説明するためのものです。

Tableau Server in a Container のイメージのセットアップ ツールを使用すると、カスタム パッケージとアーティファクトを含むようにコンテナー イメージを作成してカスタマイズすることができます。このツールの主な機能の 1 つは、コンテナー イメージを構築し、カスタム データ コネクタをインストールすることです。セットアップ ツールの使用の詳細については、Tableau Server in a Container - セットアップ ツールの使用を参照してください。

Tableau Server in a Container に関する制限事項

- Tableau Server in a Container では、サーバー ATR を使用したライセンス認証のみをサポートします。Server ATR を使用したオフライン アクティベーションは、2023.1 以降でサポートされています。この機能はコンテナで利用できますが、追加の手順と承認が必要です。エアギャップ環境またはオフライン環境のコンテナで Tableau Server を実行する必要がある場合、詳細についてはアカウント担当者にお問い合わせください。
- Tableau Server in a Container では現在、Resource Monitoring Tool (RMT) エージェントはサポートされていません。
- Kerberos は、Tableau Server in a Container ではサポートされていません。

Tableau Server in a Container - クイック スタート ガイド

このトピックでは、コマンド ラインを使用して Tableau Server in a Container のイメージの基本的なクイック スタート展開を実行するために必要なステップを説明します。ここで説明する構成は、コンテナー内で実行している Tableau Server on Linux をインストールするための最も迅速でシンプルな方法を提供することを目的としています。Tableau Server in a Container で提供される内容を確認するには、Tableau Server in a Container のセットアップ ツール または Tableau Server in a Container のイメージ までスキップしてください。

重要: 本番環境に Tableau Server を展開する際は、次のクイック スタート手順をスタンドアロン リソースとして使用しないでください。

はじめる前に

Tableau Server in a Container を使用するには、次の 2 つの基本的なステップが必要です。

1. Docker イメージを構築する

   Tableau Server in a Container の Docker イメージの構築は、RHEL ベースの Linux システム (RHEL、CentOS、または Amazon Linux 2) でのみサポートされます。他の Linux ディストリビューションでの構築は可能ですが、現在はテストが行われておらず、サポートもされていません。macOS でのイメージの構築はサポートされていません。作成されたイメージは、UBI 8 イメージ (バージョン 2022.1 以前は CentOS 7.x) に基づいています。

   コンテナー イメージを構築するには、Docker バージョン 18.09 以降がホストにインストールされている必要があります。一般的に、Docker の最新の安定したバージョンを使用することが推奨されています。一部の Linux ディストリビューションでは、ソフトウェア リポジトリに古いバージョンの Docker のみが提供されています。その場合、別のソースから Docker をインストールする必要がある場合があります。バージョン 18.09 より前のバージョンの Docker には、Tableau Server in a Container に必要な機能が含まれていません。

2. Docker イメージを実行する

Tableau Server in a Container の本番環境での使用は、Linux でのみサポートされています。探索的なプロトタイピングやテスト作業を行う場合、Linux ベースの Docker イメージを実行できるシステムはすべて、Tableau Server in a Container のイメージを実行できる必要があります (インストールの前に...で概説されているハードウェアとオペレーティング システムの要件を満たしていることを前提としています)。

Tableau Server in a Container のイメージを実行するには、Tableau Server 自体と同じハードウェア リソースが必要です。本番環境に展開する場合は、本番環境の最小ハードウェア推奨事項に記載されている推奨事項に従ってください。探索的作業をする場合は、インストールの最小ハードウェア要件に記載されている推奨事項に従ってください。パブリック クラウド環境に展開する場合は、インスタンスのサイズ設定の推奨事項が記載されている Tableau Server の最小ハードウェア要件と推奨事項ページ下部のリンクを参照してください。コンテナーの既定のリソース制限は最小要件をはるかに下回る値に設定されているため、Docker Desktop を Windows または macOS で探索作業に使用している場合は、コンテナーを正常に実行するために制限値を増やす必要があることに注意してください。

注: Tableau Support は、サポートされていないプラットフォームや構成での問題を解決することはできません。

以下のステップは、Docker がインストールされた Linux システムがあり、Tableau Server in a Container のセットアップ ツールおよび互換性があるバージョンの Tableau Server RPM インストーラーをダウンロードしたことを前提としています。

1. Tableau Server in a Container のセットアップ ツールを解凍する

   Tableau Server in a Container のセットアップ ツールは、tarball として提供されます。tar アーカイブを解凍するには、次のコマンドを使用します。

   ```
   tar -xzf tableau-server-container-setup-tool-<VERSION>.tar.gz
   ```

2. エンド ユーザー ライセンス契約に従い、登録ファイルを編集して、Tableau Server を登録するために必要な一意の識別情報を入力します。ファイル `reg-info.json` は、一意に識

別可能な必須登録情報のテンプレートとして使用でき、Tableau Server in a Container のセットアップ ツールの最上位ディレクトリにあります。このファイルは、イメージで実行されている Tableau Server インスタンスを登録するために使用します。正確な情報を提供することで、登録プロセスを適切に完了させることができます。

`eula` フィールド値には「accept」が事前に入力されており、エンドユーザー ライセンス契約 (EULA) に同意していることを示します。EULA は、ビルド ツールの EULA ディレクトリで表示できます。EULA で概説されているように、Tableau Server をアクティブ化するときは、一意に識別可能なユーザー登録を送信する必要があります。登録ファイルの編集が完了すると、他のフィールドに一意の情報を反映する値が入力されます。このファイルは、イメージで実行されている Tableau Server インスタンスを登録するために使用します。正確な情報を入力して登録プロセスを完了させ、登録を送信すると、ライセンス付与の要件が満たされます。

> **注:** Tableau Server を使用するには、EULA に同意する必要があります。EULA に同意しない場合、Tableau Server を実行できません。

編集前の登録ファイル テンプレート `reg-info.json`:

```
{
"zip" : "<value>",
"country" : "<value>",
"city" : "<value>",
"last_name" : "<value>",
"industry" : "<value>",
"eula" : "accept",
"title" : "<value>",
"phone" : "<value>",
"company" : "<value>",
"state" : "<value>",
"department" : "<value>",
"first_name" : "<value>",
```

```
"email" : "<value>"
}
```

3. Tableau Server in a Container のイメージを構築する

   Tableau Server in a Container のセットアップ ツールで、build-image スクリプトを実行します

   ```
   ./build-image --accepteula -i <Tableau Server Installer>.rpm
   ```

4. Tableau Server in a Container のイメージを実行する

   a. すべての引数を入力して、docker run コマンドを実行します。コマンド ラインに次の情報を入力する必要があります。

      - プロダクト キーまたはプロダクト キー ファイル
      - 新しい初期管理者アカウントのユーザー名とパスワード(Tableau Server の起動後にサインインするために使用します)
      - コンピューターの静的 (内部) ホスト名
      - Tableau Server in a Container のイメージの名前

   b. この情報を収集したら、最小仕様のホストで次の docker コマンドを実行します。

      ```
      docker run \
      -e LICENSE_KEY=<key> \
      -e TABLEAU_USERNAME=<username> \
      -e TABLEAU_PASSWORD=<password> \
      --hostname=<static (internal) name of host machine> \
      -p 8080:8080 -d <Tableau Server in a Container image ID or
      tag>
      ```

   c. 10~20分後に、Tableau Server が初期化されます。

5. Tableau Server にアクセスする

   Tableau Server が完全に初期化されたら、Web ブラウザーを開いて、サーバーが正しく実行されていることを確認し、 `http://<hostname>:8080` に移動します。`<hostname>` はイメージを実行しているマシンのホスト名です。

この時点で、Docker コンテナー内の Tableau Server インスタンスを実行する必要があります。より高度でカスタマイズされた展開については、以下のドキュメントを参考にしてください。

展開構成の例

Docker

Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key>
-p 8080:8080
-d <Tableau Server in a Container image ID or tag>
```

自動化された初期管理ユーザーによる Tableau Server in a Container の基本的な使用例

```
docker run \
-e LICENSE_KEY=<key> \
-e TABLEAU_USERNAME=<myadmin> \
-e TABLEAU_PASSWORD_FILE=/etc/tableau-admin-secret \
-v <full-path-to-pw-file>:/etc/tableau-admin-secret \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

TSM 専用モード

```
docker run \
-e TSM_ONLY=1 \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

マルチノードの基本的な使用例

最初のノード

オプション 1: サーバー構成 (`CONFIG_FILE`) でマルチノードトポロジが指定される場合にこれを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-v <full-path-to-config-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
```

```
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

オプション 2: サーバー構成でマルチノードトポロジが指定されていない場合でも、マルチノード展開が必要な場合は、これを使用します。

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e LICENSE_KEY=<key> -e ALWAYS_WRITE_BOOTSTRAP_FILE=1 \
-p 8080:8080 -p 8800-9000:8800-9000 -p 27000-27010:27000-27010 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

追加ノード

```
docker run \
-v <network-shared-directory>:/docker/config/bootstrap \
-e BOOTSTRAP_INSTALL=1 \
-p 8080:8080 -p 8800-9000:8800-9000 \
--hostname=<static (internal) name of host machine> \
-d <Tableau Server in a Container image ID or tag>
```

データ使用の外部化

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Init コンテナーの基本的な使用例

Init コンテナー

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
-e LICENSE_KEY=<key> \
-e INIT_CONTAINER=1 \
```

```
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

コンテナーの実行

```
docker run \
-v <empty-data-dir>:/var/opt/tableau \
--hostname=<static (internal) name of host machine> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

シングルノードバックアップからの基本的な復元

```
docker run \
-v <full-path-to-backup-file>:/docker/config/backup/backup-
file.tsbak \
-v <full-path-to-config-only-file>:/docker/config/config.json:ro \
-e LICENSE_KEY=<key> \
-p 8080:8080 -d <Tableau Server in a Container image ID or tag>
```

Docker の構成

```
version: '3.2'
services:
    tableau-server:
         hostname: localhost
         volumes:
              - <your-tsm-command-file>:/docker/config/tsm-
commands:ro
              - <your-config-file >:/docker/config/config.json:ro
         ports:
              - "8080:8080"
         image: ${IMAGE_NAME}
         environment:
              - LICENSE_KEY=<license-key>
```

インストール後のタスク

Tableau Server をインストールしたら、セキュリティ強化チェックリストの確認、サーバー イベント通知の構成、ビューのデータ キャッシュの構成、サーバー クラッシュ レポートの構成など、他の構成タスクを実行できます。

セキュリティ強化チェックリスト

次のリストは、セキュリティの向上 (「強化」) を Tableau Server インストールで行うための推奨事項を提供します。

> Tableau Server on Windows については、「セキュリティ強化チェックリスト」参照してください。

セキュリティ更新のインストール

セキュリティ更新は、Tableau Server の最新バージョンおよびメンテナンス リリース (MR) に含まれています。セキュリティ更新をパッチとしてインストールすることはできません。代わりに、最新のセキュリティ修正で Tableau Server を更新するには、最新バージョンまたは MR にアップグレードする必要があります。

アップグレード後は、このトピックの最新バージョンを常に参照してください。最新バージョンには、トピックの URL に `/current/` が含まれています。

たとえば、米国バージョンの URL は https://help.tableau.com/current/server/ja-jp/security_harden.htm です。

1.現在のバージョンへの更新

常に最新バージョンの Tableau Server を実行することをお勧めします。さらに、Tableau では、既知のセキュリティ脆弱性に対する修正を含む Tableau Server のメンテナンス リリースを定期的にパブリッシュします。(セキュリティの脆弱性に関する既知の情報は、Tableau の「セキュリティ文書」

ページ、およびSalesforce の「セキュリティアドバイザリ」ページにあります。)メンテナンス リリース通知を確認し、インストールするかどうかを決定することをお勧めします。

Tableau Server の最新バージョンまたはメンテナンス リリースを取得するには、カスタマー ポータルにアクセスしてください。

2.有効な信頼された証明書を使用して SSL/TLS を構成する

重要: バージョン 2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキーペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

Tableau Server との通信のセキュリティを保護するには、セキュア ソケット レイヤー (SSL/TLS) が不可欠です。Tableau Desktop、モバイル デバイス、および Web クライアントが安全な接続を介してサーバーに接続できるように、有効で信頼できる証明書 (自己署名証明書以外) を使用して Tableau Server を構成します。詳細については、SSLを参照してください。

3.古いバージョンの TLS の無効化

Tableau Server は TLS を使用して、コンポーネントと外部クライアントの間の多くの接続を認証および暗号化します。ブラウザーなどの外部クライアント、Tableau Desktop、Tableau Mobile は TLS over HTTPS を使用して Tableau に接続します。トランスポート レイヤー セキュリティ (TLS) は、SSL の向上バージョンです。実際、古いバージョンの SSL (SSL v2 および SSL v3) は適切に保護された通信標準とは見なされなくなりました。その結果、Tableau Server 接続に SSL v2 または SSL v3 プロトコルを使用している外部クライアントの使用を許可しません。

外部クライアントが TLS v1.3 および TLS v1.2 を使用して Tableau Server に接続するのを許可することが推奨されています。

TLS v1.2 は依然として安全なプロトコルと見なされており、多くのクライアント (Tableau Desktop を含む) では TLSv1.3 はまだサポートされていません。

サーバーで TLS v1.2 がサポートされている場合でも、TLSv1.3 対応のクライアントは TLSv1.3 をネゴシエートします。

次の tsm コマンドは TLS v1.2 および v1.3 ("all" パラメーターを使用する) を有効にし、SSL v2、SSL v3、TLS v1、および TLS v1.1 を (指定されたプロトコルの先頭にマイナス [-] 文字を付けることにより) 無効にします。TLS v1.3 では、Tableau Server のコンポーネントでまだ対応していないものがあります。

```
tsm configuration set -k ssl.protocols -v 'all -SSLv2 -SSLv3 -TLSv1
-TLSv1.1'
```

```
tsm pending-changes apply
```

Tableau Server PostgreSQL リポジトリの SSL を管理するプロトコルを変更するには、pgsql.ssl.ciphersuite を参照してください。

また、Tableau Server が SSL/TLS セッションで使用する暗号スイートの既定リストを変更することもできます。詳細については、tsm configuration set のオプションの *ssl.ciphersuite* セクションを参照してください。

4.内部トラフィック用に SSL 暗号化を構成する

Postgres リポジトリと他のサーバー コンポーネントの間のすべてのトラフィックを暗号化するために SSL を使用するようTableau Server を設定します。デフォルトでは、SSL はサーバー コンポーネントとリポジトリとの間の通信について無効になっています。単一サーバー インストールであっても、Tableau Server のすべてのインスタンス用に内部 SSL を有効化することをお勧めします。マルチノード展開では特に、内部 SSL を有効にすることが重要です。内部 Postgres 通信用に SSL を設定するを参照してください。

5.ファイアウォールによる保護を有効にする

Tableau Server は保護された内部ネットワーク内で動作するように設計されました。

重要: Tableau Server または Tableau Server のどのようなコンテンツもインターネット上または DMZ 内で実行しないでください。Tableau Server はインターネット ファイアウォールで保護された会社ネットワーク内で実行する必要があります。Tableau Server に接続する必要があるインターネット クライアント用にリバース プロキシ ソリューションを構成することをお勧めします。Tableau Server のプロキシとロード バランサーの設定を参照してください。

シングル ノード展開およびマルチノード展開では、ローカルのファイアウォールをオペレーティング システムで有効化して Tableau Server を保護する必要があります。Tableau Server の分散 (マルチノード) インストールでは、ノード間の通信は安全な通信を使用しません。そのため、Tableau Server をホストするコンピューター上でファイアウォールを有効化することをお勧めします。ローカル ファイアウォールの構成を参照してください。

受動的な攻撃者によってノード間の通信が傍受されるのを防ぐため、分離された仮想 LAN または他のネットワーク レイヤー セキュリティ ソリューションを設定します。

Tableau Server で必要なポートとサービスを理解するには、Tableau サービス マネージャーのポートを参照してください。

6.サーバー コンピューターや重要なディレクトリへのアクセスを制限する

Tableau Server 構成ファイルおよびログ ファイルには、攻撃者に対して脆弱な情報が含まれる可能性があります。そのため、Tableau Server を実行しているマシンへの物理アクセスを制限します。さらに、認証され、信頼されたユーザーのみが `/var/opt/tableau/tableau_server/` ディレクトリの Tableau Server ファイルにアクセスできるようにします。

7.新しいシークレットとトークンの生成

リポジトリまたはキャッシュ サーバーと通信する Tableau Server サービスは、シークレット トークンを使用して最初に認証する必要があります。シークレット トークンは、Tableau Server セットアップ時に生成されます。セットアップ時には、Postgres リポジトリへのトラフィックを暗号化するために内部 SSL が使用する暗号化キーも生成されます。

Tableau Server をインストールした後、展開用に新しい暗号化キーを作成することをお勧めします。

これらのセキュリティアセットは、tsm security regenerate-internal-tokens コマンドで再生成できます。

次のコマンドを実行します。

```
tsm security regenerate-internal-tokens

tsm pending-changes apply
```

8.使用していないサービスの無効化

Tableau Server の攻撃表面を最小限に抑えるため、不要な接続ポイントを無効化します。

JMX サービス

JMX は、既定で無効化されています。有効化されているにもかかわらず使用していない場合は、次を実行して無効化する必要があります。

```
tsm configuration set -k service.jmx_enabled -v false

tsm pending-changes apply
```

9.セッション存続時間設定の検証

既定では、Tableau Server に絶対的なセッション タイムアウトはありません。つまり、Tableau Server の非アクティブ タイムアウトを超えない場合、ブラウザーベースのクライアント セッション (Web 作成) を開いたままにすることができます。既定の非アクティブ タイムアウトは、240 分です。

セキュリティポリシーで要求される場合は、絶対的なセッション タイムアウトを設定できます。絶対的なセッション タイムアウトは、時間がかかる抽出のアップロードまたはワークブックのパブリッシュ操作が組織で実行できる範囲で設定してください。セッション タイムアウトの設定が短すぎると、操作に時間がかかりすぎて抽出やパブリッシュが失敗する場合があります。

セッション タイムアウトを設定するには、次のコマンドを実行します。

```
tsm configuration set -k wgserver.session.apply_lifetime_limit -v
true
```

`tsm configuration set -k wgserver.session.lifetime_limit -v` *value*、ここで *value* は分数です。既定では、1440 (24 時間) です。

`tsm configuration set -k wgserver.session.idle_limit -v` *value*、ここで *value* は分数です。既定では、240 です。

```
tsm pending-changes apply
```

接続されたクライアント (Tableau Desktop、Tableau Mobile、Tableau Prep Builder、Bridge、および個人用アクセス トークン) のセッションでは、OAuth トークンを使用して、セッションを再確立することによりユーザーをログインしたままにします。すべての Tableau クライアント セッションを、上記のコマンドによって制御されるブラウザーベースのセッション制限のみによって管理する場合は、この動作を無効にできます。詳細については、自動クライアント認証を無効にするを参照してください。

10. ファイルベースのデータ ソース用にサーバーの許可リストを設定する

2023 年 10 月の Tableau Server リリース以降、デフォルトのファイルベースのアクセス動作が変わりました。以前は、Tableau Server は許可された Tableau Server ユーザーに対し、サーバー上のファイルをファイル ベースのデータ ソース (スプレッドシートなど) として使用するワークブックの作成を許可していました。2023 年 10 月のリリースでは、Tableau またはリモート共有に保存されているファイルへのアクセスは、ここで説明する設定を使用して Tableau Server で特別に構成する必要があります。

この設定により、`tableau`システム アカウントによるアクセスを、指定したディレクトリのみに制限できます。

共有ファイルへのアクセスを構成するには、許可リスト機能を構成する必要があります。これにより、データ ファイルをホストしているディレクトリ パスのみに `tableau` アカウントのアクセスを制限できます。

1. Tableau Server を実行しているコンピューターで、データ ソース ファイルをホストするディレクトリを識別します。

   重要: この設定で指定したファイル パスが存在し、システム アカウントでアクセス可能であることを確認してください。

2. 次のコマンドを実行します。

   `tsm configuration set -k native_api.allowed_paths -v "path"`、ここで、*path* は許可リストに追加するディレクトリです。指定したパスのすべてのサブディレクトリは、許可リストに追加されます。指定したパスの末尾にバックスラッシュを追加する必要があります。複数のパスを指定する場合は、この例のようにセミコロンで区切ります。

   ```
   tsm configuration set -k native_api.allowed_paths -v
   "/datasources;/HR/data/"

   tsm pending-changes apply
   ```

11.Web ブラウザー クライアントでの HTTP Strict Transport Security の有効化

HTTP Strict Transport Security (HSTS) は、Tableau Server などの Web アプリケーション サービスで構成されるポリシーです。適合するブラウザーに HSTS を実行している Web アプリケーションがある場合、サービスとのすべての通信が安全な接続 (HTTPS) を介して行われる必要があります。HSTS は主要なブラウザーでサポートされています。

HSTS のしくみとサポートされるブラウザーの詳細については、Open Web Application Security Project Web ページの「HTTP Strict Transport Security チート シート (HTTP Strict Transport Security Cheat Sheet)」を参照してください。

HSTS を有効にするには、Tableau Server で次のコマンドを実行してください。

```
tsm configuration set -k gateway.http.hsts -v true
```

既定では、HSTS ポリシーが 1 年に設定されています (31536000 秒)。ブラウザーが HTTPS 経由でサーバーにアクセスする期間を指定します。HSTS の初期ロールアウト時は最長期間を短く設定することを検討する必要があります。この期間を変更するには、`tsm configuration set -k gateway.http.hsts_options -v max-age=<seconds>` を実行します。たとえば、HSTS ポリシーの期間を 30 日に設定し、`tsm configuration set -k gateway.http.hsts_options -v max-age=2592000` と入力します。

```
tsm pending-changes apply
```

12.ゲストアクセスの無効化

Tableau Server のコアベースのライセンスには、ゲストユーザー オプションが含まれます。このオプションでは、組織のすべてのユーザーに対し、Web ページに埋め込まれた Tableau ビューの表示や操作を許可できます。

コアベースのライセンスを使用して展開された Tableau Server では、ゲストユーザー アクセスが既定で有効になっています。

ゲストアクセスによって、ユーザーは埋め込みビューを表示できます。ゲストユーザーは Tableau Server インターフェイスを参照できません。また、ビューを操作するサーバー インターフェイス エレメントは表示されません (ユーザー名、アカウント設定、コメントなど)。

組織でコア ライセンスを使用して Tableau Server を展開しており、ゲストアクセスが必要ない場合は、ゲストアクセスを無効にします。

ゲストアクセスをサーバー レベルまたはサイトレベルで無効にできます。

サーバー レベルまたはサイトレベルでゲストアカウントを無効にするには、サーバー管理者である必要があります。

ゲスト アクセスをサーバー レベルで無効にするには、次の手順を行います。

1. サイト メニューで、**[すべてのサイトを管理] > [設定] > [全般]** の順にクリックします。
2. **[ゲスト アクセス]** で、**[ゲスト アカウントを有効にする]** チェック ボックスをクリアします。

3. **[保存]** をクリックします。

サイトのゲスト アクセスを無効にするには、次の手順を行います。

1. サイト メニューで、サイトを選択します。
2. **[設定]** をクリックし、[設定] ページで、**[ゲスト アカウントを有効化]** チェック ボックスをクリアします。

詳細については、ゲスト ユーザーを参照してください。

13.Referrer-Policy HTTP ヘッダーを 'same-origin' に設定する

2019.2 から、Tableau Server には Referrer-Policy HTTP ヘッダーの動作を構成する機能が含まれています。このポリシーは、すべての "secure as" 接続で起点の URL を含める既定の動作 (`no-referrer-when-downgrade`) を指定して有効になっています。この動作では、同種の接続 (HTTP から HTTP) またはよりセキュリティの高い接続 (HTTP から HTTPS) にのみ、起点の参照元に関する情報が送信されます。

この値を、参照元の情報が同じサイトの起点にのみ送信される `same-origin` に設定することをお勧めします。サイトの外部からの要求は参照元に関する情報を受信しません。

Referrer-Policy を `same-origin` に更新するには、次のコマンドを実行します。

```
tsm configuration set -k gateway.http.referrer_policy -v same-origin

tsm pending-changes apply
```

セキュリティ向上を目的とした追加のヘッダー構成の詳細については、HTTP 応答ヘッダーを参照してください。

14.TLS による SMTP 接続の構成

2019.4 から、Tableau Server には TLS による SMTP 接続を構成する機能が含まれています。Tableau Server では、STARTTLS (便宜的または明示的な TLS) のみがサポートされています。

オプションで、メール サーバーに接続するように Tableau Server を構成することができます。SMTP を構成したら、システム障害についてサーバー管理者にメールを送信し、サブスクライブしたビューおよびデータ主導アラートについてサーバー ユーザーにメールを送信するようTableau Server を構成することができます。

TLS によるSMTP を構成するには、次の手順を実行します。

1. 互換性のある証明書をTableau Server にアップロードします。tsm security custom-cert add を参照してください。
2. TSM CLI を使用してTLS 接続を構成します。

   次のTSM コマンドを実行して、SMTP サーバーへのTLS 接続を有効化および強制し、証明書の検証を有効にします。

   ```
   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   enabled -v true

   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   required -v true

   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   check_server_identity -v true
   ```

   既定では、Tableau Server は TLS バージョン 1、1.1、および 1.2 をサポートしますが、SMTP サーバーがサポートする最も高いTLS バージョンを指定することをお勧めします。

   次のコマンドを実行してバージョンを設定します。有効な値は、`SSLv2Hello`、`SSLv3`、`TLSv1`、`TLSv1.1`、および `TLSv1.2` です。次の例では、TLS バージョンをバージョン 1.2 に設定します。

   ```
   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   versions -v "TLSv1.2"
   ```

   TLS 構成オプションの詳細については、SMTP セットアップの構成 を参照してください。

3. 変更を反映するには、Tableau Server を再起動します。次のコマンドを実行します。

```
tsm pending-changes apply
```

15.LDAP の SSL を設定

汎用 LDAP 外部 ID ストアを使用するように Tableau Server の展開が構成されている場合は、Tableau Server とLDAP サーバー間の認証を保護するように SSL を構成することをお勧めします。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

Active Directory を使用するように Tableau Server の展開が構成されている場合は、認証トラフィックを保護するために Kerberos を有効にすることをお勧めします。Kerberosを参照してください。

変更リスト

Date	Change
May 2018	Added clarification: Do not disable REST API in organizations that are running Tableau Prep.
May 2019	Added recommendation for referrer-policy HTTP header.
June 2019	Removed recommendation to disable Triple-DES. As of version 2019.3, Triple-DES is no longer a default supported cipher for SSL. See 変更箇所 - アップグレード前に知っておく事柄.
January 2020	Added recommendation to configure TLS for SMTP.
February 2020	Added recommendation to configure SSL for LDAP server.
May 2020	Added TLS v1.3 to the disabled list of TLS ciphers. Added clarification to introduction about topic versioning.
October 2020	Added TLS v1.3 as a default supported cipher.
January 2021	Added clarification: All products enabled by the Data Management license require REST API.

February 2021	Removed recommendation to disable REST API. The API is now used internally by Tableau Server and disabling it may limit functionality.

SMTP セットアップの構成

Tableau Server は、システム障害についてサーバー管理者にメールを送信し、サブスクライブしたビューおよびデータ主導アラートについてサーバー ユーザーにメールを送信することができます。ただし、最初に Tableau Server がメールの送信に使用する SMTP サーバーを設定する必要があります。SMTP を構成した後は手順を完了して通知を構成しますが (サーバー イベント通知の構成)、その後サーバーを起動または再起動するとメール通知が起動し、これで通知が正しく設定されたことが確認されます。

SMTP の構成には Tableau Server サービスの再起動が必要です。

セキュア SMTP

SMTP で TLS を有効にして構成するには、このトピックの説明に従って TSM CLI を使用する必要があります。Tableau Server では、STARTTLS (便宜的または明示的な TLS) のみがサポートされています。

組織が TLS 接続の検証にパブリック証明書を使用しない場合は、プライベート証明書を Tableau Server にアップロードして信頼できる接続を検証することができます。詳細については、tsm security custom-cert add コマンドを参照してください。

また、証明書の検証プロセスを無効にすることによってのみ、SMTP TLS で暗号化を構成することもできます。詳細については、下記の「*TSM CLI の使用*」タブの「*構成ファイル リファレンス*」セクションを参照してください。

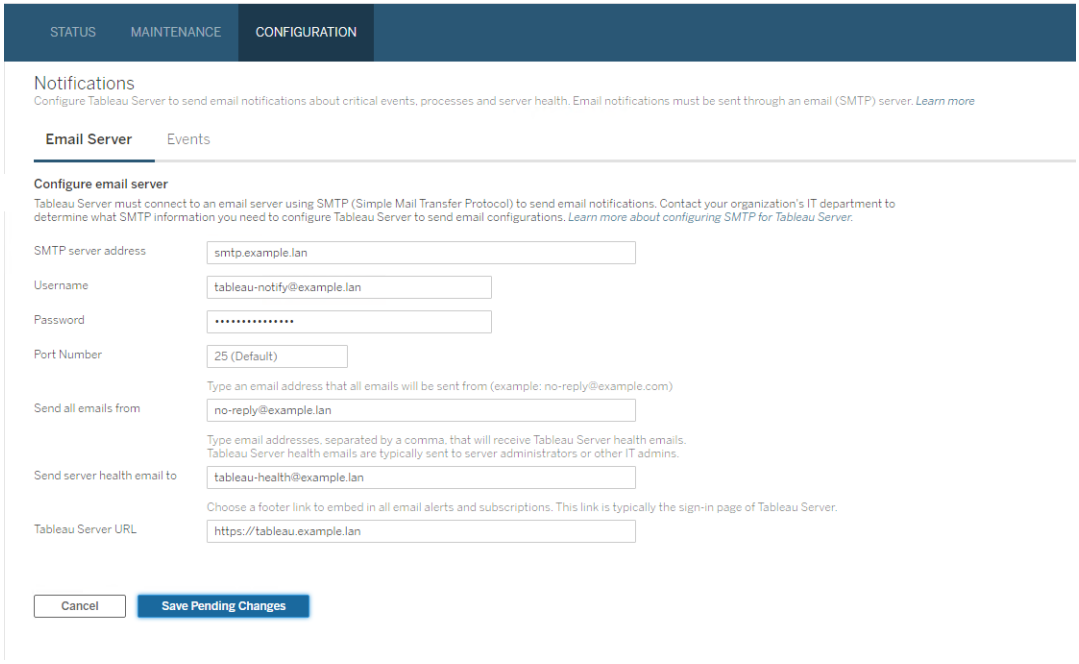
TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[通知]** をクリックし、**[メール サーバー]** をクリックします。

3. 自分の組織の SMTP 構成情報を次のように入力します。



4. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

5. ページ上部の **[変更を保留中]** をクリックします。

6. **[変更を適用して再起動]** をクリックします。

7. `tsm email test-smtp-connection` を実行し、接続構成を表示して検証します。tsm email test-smtp-connection を参照してください。

TSM CLI の使用

SMTP の初期の構成では、次の構成ファイル テンプレートを使用して json ファイルを作成することをお勧めします。また、tsm configuration set で説明されている構文と共に以下の単一構成キー

を設定することもできます。

1. 以下の json テンプレートをファイルにコピーします。

   重要: 以下のテンプレートには、ほとんどの展開に共通するオプションが含まれています。テンプレートをテキストファイルにコピーした後、SMTP サーバー要件のオプション値を編集する必要があります。オプションの削除や追加が必要になる場合があります。サポートされているすべての SMTP キーオプションの詳細については、後続のリファレンス セクションを参照してください。

   ```
   {
   "configKeys": {
          "svcmonitor.notification.smtp.server": "SMTP server host name",
          "svcmonitor.notification.smtp.send_account": "SMTP user name",
          "svcmonitor.notification.smtp.port": 443,
          "svcmonitor.notification.smtp.password": "SMTP user account password",
          "svcmonitor.notification.smtp.ssl_enabled": true,
          "svcmonitor.notification.smtp.from_address": "From email address",
          "svcmonitor.notification.smtp.target_addresses": "To email address1,address2",
          "svcmonitor.notification.smtp.canonical_url": "Tableau Server URL"
          }
   }
   ```

2. `tsm settings import -f file.json` を実行し、適切な値を指定した json ファイルを Tableau サービス マネージャーに渡して Tableau Server の SMTP を構成します。Tableau サービス マネージャーによってエンティティの値が検証されます。

3. `tsm pending-changes apply` コマンドを実行して変更を適用します。tsm pending-changes apply を参照してください。

4. `tsm email test-smtp-connection` を実行し、接続構成を表示して検証します。tsm email test-smtp-connection を参照してください。

SMTP CLI 構成リファレンス

次の表は、TSM の CLI で SMTP を構成する際に使用できるすべてのオプションをリストしたものです。

オプション	説明
`svcmonitor.notification.smtp-.server`	SMTP サーバーのアドレス。 例: `"svcmonitor.notification.smtp.server": "mail.example.com"`
`svcmonitor.notification.smtp-.send_account`	SMTP アカウントのユーザー名。
`svcmonitor.notification.smtp-.port`	SMTP サーバーのポート番号。既定は 25 です。
`svcmonitor.notification.smtp-.password`	SMTP サーバー アカウントのパスワード。 例: `"svcmonitor.notification.smtp.password":"password"`
`svcmonitor.notification.smtp-.ssl_enabled`	SMTP サーバーへの接続が暗号化されるかどうかを指定します。既定は false です。
`svcmonitor.notification.smtp-.ssl_required`	有効にすると、Tableau Server は TLS を使用しない SMTP サーバーへの接続を拒否します。`svcmonitor.notification.smtp.ssl_enabled`オプションも true に設定する必要があります。

オプション	説明
	既定は false です。
`svcmonitor.notification.smtp-.ssl_check_server_identity`	true に設定すると、Tableau Server は RFC 2595 で指定されている SMTP サーバー ID を確認します。サーバーの証明書の内容に基づきこの追加確認を行うのは、中間者攻撃を防ぐことが目的です。 既定は false です。
`svcmonitor.notification.smtp-.ssl_trust_all_hosts`	TLS を使用する場合、すべてのメール サーバーからの証明書を信頼し、証明書の信頼チェーンの有効性を無視します。このキーを true に設定すると、TLS は SMTP ホストへのトラフィックの暗号化にのみ使用されます。 既定は false です。
`svcmonitor.notification.smtp-.ssl_ciphers`	暗号スイートの既定とサポート対象のセットは、Tableau Server でインストールされている JDK のバージョンによって定義されます。サポート対象の暗号と既定の暗号のリストについては、以下の TLS 暗号セクションを参照してください。 Tableau Server が SMTP TLS 接続に使用する暗号スイートを更新するには、この値に暗号スイートを空白で区切ったリストを入力します。たとえば、`"TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256 TLS_DHE_DSS_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"` のように使用します。
`svcmonitor.notification.smtp-.ssl_versions`	このバージョンの Tableau Server で有効な既定の TLS バージョンは TLSv1、TLSv1.1、TLSv1.2、

オプション	説明
	TLSv1.3 です。 TLS バージョンのサポートは、Tableau Server でインストールされている JDK のバージョンによって定義されます。 サポートされている TLS のバージョンは `SSLv2Hello, SSLv3, TLSv1, TLSv1.1, TLSv1.2, TLSv1.3` です。 Tableau Server が SMTP TLS 接続に使用するバージョンを更新するには、この値にバージョンを空白で区切ったリストを入力します。たとえば、`"TLSv1.2 TLSv1.3"` のように使用します。
`svcmonitor.notification.smtp-.from_address`	システム障害が発生した場合に通知を送信する電子メール アドレス。電子メール アドレスは有効な構文 (ITalerts@bigco.com や noreply@mycompany など) でなければなりませんが、Tableau Server 上の実際のメール アカウントである必要はありません。(一部の SMTP サーバーでは、実際のメール アカウントが必要となる場合があります)。 **注**: サイトごとにシステム全体のメール アドレスを上書きできます。詳細については、サイトとはを参照してください。 例: `"svcmonitor.notification.smtp.from_address": "donot-reply@example.com"`

オプション	説明
`svcmonitor.notification.smtp-.target_addresses`	通知を受信する電子メール アドレス。メール通知が有効化されている場合、1 つ以上のアドレスを含める必要があります。複数のアドレスはコンマで区切ります。 例: `"svcmonitor.notification.smtp.target_addresses": "iluvdata@example.com"`
`svcmonitor.notification.smtp-.canonical_url`	Tableau Server の URL。`http://` または `https://` を入力してから Tableau Server の名前または IP アドレスを入力します。サブスクリプション メールのフッターで使用します。 例: `"svcmonitor.notification.smtp.canonical_url": "http://myserver.example.com"`

TLS 暗号

以下は、Tableau Server に含まれる JDK でサポートされている TLS 暗号のリストです。このバージョンの Tableau Server では、これらの暗号はすべて既定で有効になっています。上記の表の説明に従い、`svcmonitor.notification.smtp.ssl_ciphers` オプションを空白で区切ったリストを入力することで、SMTP 構成の暗号のカスタム リストを指定することができます。

TLS_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDH_RSA_WITH_AES_256_CBC_	TLS_ECDHE_ECDSA_WITH_AES_128_

SHA384	CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_128_GCM_SHA256	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256	TLS_DHE_DSS_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_256_GCM_SHA384	TLS_EMPTY_RENEGOTIATION_INFO_SCSV
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	TLS_DHE_DSS_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_SHA	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_RSA_WITH_AES_256_	TLS_ECDH_RSA_WITH_AES_128_CBC_

CBC_SHA384	SHA
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	TLS_DHE_DSS_WITH_AES_128_CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
TLS_RSA_WITH_AES_128_CBC_SHA	TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	TLS_AES_256_GCM_SHA384
TLS_AES_128_GCM_SHA256	

TSM におけるファイルとパーミッション

このトピックは Tableau サービス マネージャー (TSM) に必要とされるパーミッション要件について取り上げます。この情報はサーバー管理者向けのものです。このトピックは、Tableau Server 内でのユーザーおよびコンテンツの管理 (コンテンツおよびユーザーに対するパーミッション) に使用されるパーミッションについては*取り扱いません*。これらのパーミッションに関する情報については、パーミッションを参照してください。

TSM および Tableau Server のインストール中に、権限なしユーザー (*tableau*) が作成され、サーバー認証グループ (*tableau*) に追加されます。このユーザー アカウントは、TSM および Tableau

Server プロセスによって作業が実行されることを可能にします。インストール中に、このユーザーおよびグループを変更できます。詳細については、アイデンティティストアを参照してください。

TSM でのパーミッションの要件は両方のファイル、およびそれらのファイルが配置されているディレクトリに適用されます。TSM がファイルを作成および管理するとき、ファイルは特定の既定の場所に、必要なパーミッションと共に配置されるため、パーミッションを設定することについて心配する必要はありません。ユーザーが自らファイルを作成、コピーまたは移動する、または既定の場所以外の場所にファイルを配置するときは、対象ファイルに TSM が適切にアクセスできるよう、パーミッションの要件について注意が必要です。一般的なケース (既定の場所以外の場所を使用することについての情報は、tsm ファイル パスを参照してください) 。

パーミッションと TSM についての一般的なルールとは以下の通りです。

- ファイル—*tableau* グループがあるファイルにアクセスする権限を持つ場合 (それがグループオーナーでファイルへの読み取りアクセス権限を持つ場合)、グループに属するユーザーもファイルへのアクセス権限を持ちます。もう1 つのアプローチとは、"other" 読み込みアクセスを付与することです。

- ディレクトリ—*tableau* グループにあるファイルを含むディレクトリおよびそのディレクトリの親ディレクトリすべてへの読み込みおよび実行アクセスの権限を持つ場合、グループに属するユーザーはそのファイルへのアクセス権限を持ちます。

パーミッションの調整を必要とする状況には、サーバー バックアップ ファイルとサイト インポート アーカイブを異なるコンピューターから既定の場所以外の場所にコピーした場合、ロゴや画像などのカスタマイズ用ファイル、および SSL 証明書などのセキュリティ証明書が含まれます。

たとえば、Windows 上の Tableau Server から Linux 上の Tableau Server に移行する場合、Windows 内で作成されたバックアップを使用してお使いの Linux サーバーにデータを復元します。このバックアップ ファイルは TSM に作成されたものでないため、復元プロセスがそのファイルにアクセスするための適切なパーミッションを持たない場合があります。バックアップ ファイルとそれをコピーする先のディレクトリ構造に、適切なパーミッションがあることを確認する必要があります。同様に、証明書のようなファイルをクラスタ内の追加のノードにコピーしようとする場合、ファイルとそれをコピーする先のディレクトリにアクセスするために *tableau* ユーザーが必要とするパーミッションがあるようにする必要があります。

個別のファイルにパーミッションを設定する

TSM に作成された既定の場所の 1 つからコピーしたファイルを使用している場合、*tableau* ユーザーに読み取りアクセスを付与することで、ファイルの所有権とパーミッションが TSM のアクセスを許可するようにする必要があります。これは次の 2 つの方法で行うことができます。

- `chgrp` および `chmod` コマンドでファイルに読み取りおよび実行アクセス権限を *tableau* グループ(既定のインストール内の) に付与することで*tableau* ユーザーに読み取りアクセスを付与できます。例:

  ```
  chgrp tableau <backup>.tsbak

  chmod g+rx <backup>.tsbak
  ```

- そのかわりに、ファイルの読み取りおよび実行アクセス権限を全体に付与することもできます。

  ```
  chmod o+rx <backup>.tsbak
  ```

ディレクトリのパーミッション設定

ファイル自体に適切なパーミッションを設定することに加え、TSM はファイルを含むディレクトリ、およびすべての親ディレクトリへのパーミッションも必要としています。TSM がアクセスする予定のファイルを既定の場所以外の場所に置いている場合、親ディレクトリまたはファイルを含むディレクトリへのパーミッションが読み取りおよび実行アクセスを許可するようにする必要があります。

次の 2 つの方法でこの問題を解決できます。

- ディレクトリの所有権を *tableau* グループに変更し、グループにディレクトリの読み取りおよび実行パーミッションを追加する。これにより、*tableau* ユーザーがディレクトリ内のファイルを利用できるようになります。

  ```
  chgrp tableau <directory-name>

  chmod g+rx <directory-name>
  ```

- そのかわりに、ディレクトリの読み取りおよび実行アクセス権限を全体に追加することもできます。これにより、システム上のすべてのユーザーがディレクトリ内のファイルを利用できるようになります。このアプローチはディレクトリ内の他のファイルのセキュリティを確保するために追加の手順を必要とする場合があります。たとえば、ディレクトリ内の他のファイルを他のユーザーが読み取れないように、全体から読み取り可能にしないようにする場合があります。

  ```
  chmod o+rx <directory-name>
  ```

ヒント: `namei -mo` コマンドを使用して、パーミッション ツリー全体をリスト化できます。これにより、*tableau* グループからのアクセスを許可するためにどのディレクトリのパーミッションが必要なのかを確認しやすくなります。詳細については、インターネットを参照してください。

サーバー イベント通知の構成

Tableau サービス マネージャー (TSM) 管理者は、次のイベントの通知を許可するように Tableau Server を構成できます。

- コンテンツの更新
  - 抽出失敗 (既定では有効)
  - ユーザーのサブスクリプション ビュー (既定では無効)
- サーバーの健全性モニタリング
  - サーバー ステータスの変化 (既定では無効)
  - デスクトップ ライセンス レポート (既定では無効)
- ドライブの空き容量
  - ディスク空き容量が事前に設定したしきい値以下になった場合に送信するメール アラート (デフォルトでは無効)
  - 使用履歴の記録 (既定では有効)

注: サブスクリプションまたは通知を設定する前に、SMTP を設定する必要があります。詳細については、SMTP セットアップの構成を参照してください。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   `https://<tsm-computer-name>:8850`。

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[通知]** をクリックし、**[イベント]** をクリックします。

3. 組織に合わせた通知設定を構成します。

   - コンテンツの更新

     - **抽出更新の失敗の際にメールを送信する**

       このオプションが有効 (既定) の場合、サーバー管理者は、抽出更新の失敗の際にメール通知を送信するように構成できます。これらのメッセージはサイト レベルで構成されるため、このオプションが有効になっていても、サイトで **[スケジュールされた更新に失敗した場合、データ ソースおよびワークブックの所有者にメールを送信する]** オプションが有効になっていない限り(既定では有効)、メッセージは送信されません。詳細については、抽出更新スケジュールと失敗通知を有効にするを参照してください。

     - **ユーザーに登録しているビューに関するメール受信を許可**

       このオプションが有効になっている場合 (既定では無効)、サーバー管理者はサブスクリプション メールを送信するようにサイトを構成できます。これらのメール メッセージはサイト レベルで構成され、このオプションが有効な場合にのみ構成できます。詳細については、サブスクリプションのサーバー設定を参照してください。

       ユーザーがワークブックやビューをサブスクライブすると、ビューのスナップショットがそのユーザーにメールで送信されるため、ユーザーは Tableau Server にサインインせずに最新の更新を確認できます。

ユーザーがサブスクリプション メールに PDF レンダリングを添付できるようにするには、**[サブスクライブ中のビューへの添付ファイルの追加をユーザーに許可する]** を選択します。

- サーバーの健全性モニタリング
  - **Tableau Server プロセス イベント (実行中、ダウン、フェールオーバー) に関するメールを送信する**

    データエンジン、ファイル ストア、ゲートウェイ、またはリポジトリサーバープロセスが停止または再開するとき、または最初の Tableau Server ノードが停止または再開するときに、Tableau Server がメール メッセージを送信します。

    単一サーバーへのインストールを実行している場合 (同じコンピューター上にすべてのプロセスが存在する場合)、健全性に関するアラートは、Tableau Server の稼働中にのみ送信されます。「DOWN (ダウン)」アラートは送信されません。フェールオーバーのために構成された分散インストールを実行している場合、DOWN (ダウン) アラートはアクティブなリポジトリまたはデータエンジンのインスタンスに障害が発生していることを意味し、後続の UP (アップ) アラートはプロセスのパッシブ インスタンス (リポジトリ) または 2 番目のインスタンス (データエンジン) が引き継いだことを意味します。

    > **注:** Tableau Server は、自己修正するように設計されています。サービスまたはプロセスが応答を停止するかダウンした場合、Tableau Server はそれを再起動しようとします。これを完了するまでに 15 ～ 30 分かかる場合があります。そのため、サービスまたはプロセスのアラートにすぐに対応しても、特に再起動中に要求を処理できる冗長サービスを備えたインストールでは、非生産的なアクションになる可能性があります。

  - **Tableau Desktop ライセンス レポートの有効化**

ライセンス レポートは Tableau Desktop で生成され、Tableau Server に送信されます。このオプションが有効になっている場合、Tableau Server はデスクトップ ライセンス レポートの管理レポートを生成して表示します。レポートの詳細については、デスクトップ ライセンスの使用率を参照してください。

- ドライブの空き容量

  Tableau Server のディスク空き容量についての通知 (アラート) を有効にします。

  - **未使用のドライブ空き容量がしきい値を下回っている場合にメールを送信する**

    いずれかのノードでディスク空き容量がしきい値を下回った場合や、継続して下回っている場合にメール通知を送信するように、Tableau Server を設定できます。しきい値の通知を送信する頻度も設定できます。

    [警告しきい値] と **[重要しきい値]** の 2 つを設定する必要があります。しきい値は、残っているディスク空き容量のパーセンテージで表されます。クリティカルしきい値は、警告しきい値よりも小さくなければなりません。

    [Send threshold alert every (しきい値アラートを送信する頻度)] オプションも指定します。これにより、警告や緊急の通知を送信する間隔 (分) を決定します。既定値は 60 分です。

  - **カスタム管理ビュー用のディスク空き容量の使用率に関する情報としきい値違反を記録する**

    ディスク空き容量の使用率を記録するようTableau Server を設定している場合、ディスク空き容量に関する情報はリポジトリに保存され、管理ビューを使用して使用履歴を表示できます。

4. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

5. ページ上部の **[変更を保留中]** をクリックします。

6. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

tsm configuration set コマンドを使用し、前述のさまざまな通知の値を個別に設定できます。または、json ファイルを作成し、1 回の操作ですべての構成値を渡すこともできます。どちらの方法も本セクションで説明しています。

個別に通知の値を設定する

以下の表は、本トピックで前述した通知イベントにマッピングするキー/値のペアを示しています。単一のキー/値のペアを設定するには、以下の構文で tsm configuration set コマンドを使用します。

```
tsm configuration set -k <config.key> -v <config_value>
```

たとえば、ジョブ失敗の通知を有効にするには、以下のコマンドを実行します。

```
tsm configuration set -k backgrounder.notifications_enabled -v true
```

通知オプション	キー	値
抽出失敗またはフロー実行失敗	`backgrounder.notifications_enabled`	`true`|`false`
ユーザーのサブスクリプションビューの有効化	`subscriptions.enabled`	`true`|`false`
サブスクリプションの PDF 添付ファイルの有効化	`subscriptions.attachments_enabled`	`true`|`false`

サブスクリプション通知の最大添付ファイル サイズ (MB)	`subscriptions.max_attachment_size_megabytes`	整数値。既定は 150 です
サーバー ステータスの変化	`svcmonitor.notification.smtp.enabled`	`true|false`
ライセンスのレポート	`features.DesktopReporting`	`true|false`
残っている容量のしきい値: メール通知の有効化	`storage.monitoring.email_enabled`	`true|false`
残っている容量のしきい値: 警告を行うパーセンテージ	`storage.monitoring.warning_percent`	20 などの整数値
残っている容量のしきい値: クリティカルなパーセンテージ	`storage.monitoring.critical_percent`	15 などの整数値
メール間隔の設定	`storage.monitoring.email_interval_min`	25 などの整数値 (分)
使用履歴の記録	`storage.monitoring.record_history_enabled`	`true|false`

値を設定したら、以下のコマンドを実行する必要があります。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバー

が停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

単一の json ファイルを使用してすべての通知の値を設定する

単一の構成ですべての通知設定を作成するには、json ファイルを渡します。

次のテンプレートをコピーして編集し、構成用のファイルを作成します。

```
{
  "configKeys": {
    "backgrounder.notifications_enabled": true,
    "subscriptions.enabled": true,
    "subscriptions.attachments_enabled": true,
    "subscriptions.max_attachment_size_megabytes": 150,
    "svcmonitor.notification.smtp.enabled": true,
    "features.DesktopReporting": true,
    "storage.monitoring.email_enabled": true,
    "storage.monitoring.warning_percent": 20,
    "storage.monitoring.critical_percent": 15,
    "storage.monitoring.email_interval_min": 25,
    "storage.monitoring.record_history_enabled": true
  }
}
```

ファイルを保存したら、それを以下のコマンドを使用して渡します。

```
tsm settings import -f <path-to-file.json>
```

変更を適用するには、次のコマンドを実行します。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプ

ションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

データキャッシュの構成

Tableau Server にパブリッシュされたビューはインタラクティブであり、データベースとライブ接続されている場合があります。Web ブラウザーでユーザーとしてビューを操作すると、クエリされたデータがキャッシュに格納されます。それ以降のアクセスでは、キャッシュを使用できる場合、そのキャッシュのデータが使用されます。既定では、Tableau Server はできるだけ長い間データをキャッシュして再利用します。`tsm data-access caching set` コマンドを使用してキャッシュオプションを構成することで、この動作を変更できます。

1. 以下のコマンドを実行します。

   ```
   tsm data-access caching set -r <value>
   ```

   <value> には次のいずれかのオプションが入ります。

   - **low または空の文字列 ("")。**これは既定値であり、Tableau Server がキャッシュを構成し、キャッシュに保存されたデータが使用可能な場合は常にそれを使用することを示します。
   - <value>。"<value>" はデータがキャッシュされる最長時間を分数で指定します。
   - **always** または **0** (ゼロ)。これらの値は、Tableau Server が常に最新のデータを取得することと、ページが再読み込みされるたびにキャッシュが更新されることを示します。

2. `tsm pending-changes apply` コマンドで変更を適用します。これにより Tableau Server が再起動されます。

データベース ドライバー

Tableau コネクタにはデータベースとやり取りするためのドライバーが必要です。Tableau Server からデータソースに接続できるようになる前に、接続するデータソース用のドライバーをインストールする必要があります。Tableau Server on Linux がサポートするデータソースの詳細については、Tableau Server の技術仕様ページで確認できます。サポート対象であるすべてのコネクタのドライバーのリンクおよびインストール手順については、ドライバーのダウンロードページで確認できます。

> **重要:** 組み込みの管理者ビューを使用する場合は、PostgreSQL ドライバーをインストールする必要があります。このドライバーは、ドライバーのダウンロードページで見つけることができます。

クラスタでのドライバーのインストール

データソースのドライバーは Tableau Server クラスタの最初のノードにインストールする必要があります。複数のノードに Tableau Server をインストールする場合は、次のいずれかのプロセスを実行するすべてのノードにドライバーをインストールする必要もあります。

- アプリケーション サーバー (Vizportal)
- バックグラウンダー
- データ サーバー
- VizQL Server

サーバー クラッシュ レポーター

Tableau Server 管理者は、サーバーの問題によりクラッシュが発生した場合、ログを作成し関連ファイルを Tableau に送信することを許可するオプションを有効にできます。これらのファイルは、Tableau によって、クラッシュの原因を識別および解決するために使用されます。既定では、このオプションは無効になっています。データプライバシーに関する規制の対象とならない組織でのみ有効化してください。

重要: データがプライバシー規制の対象となる場合は、クラッシュレポートを有効化しないでください。

Tableau Server の問題によりクラッシュが発生した場合、ログ ファイルとダンプ ファイルが生成されます。クラッシュデータのアップロード機能が有効になっている場合、これらのファイルは自動収集および暗号化パッケージに圧縮され、スケジュールされた時刻にバックグラウンドで送信されます。ネットワーク パフォーマンスへの影響を制限するため、暗号化されたパッケージは小さなパッケージで送信されます。一度にパッケージ化およびアップロードできるクラッシュレポートは 1 つのみで (新しいクラッシュレポートは、前のパッケージがアップロードされるまでパッケージ化されません)、"先入れ先出し" の順で送信されます。ユーザーへの影響をさらに抑えるため、使用量の少ない時間枠に送信するようスケジュール設定できます。

暗号化されたパッケージは、次を含むクラッシュダンプ ファイルおよびログで構成されます。

- クラッシュ/ コア ダンプ ファイル
- クラッシュに関連するエラー ログ ファイル
- クラッシュに関連するマニフェスト ファイル

ファイルには、次のようなデータが含まれる場合があります:

- マシン固有の情報 (例: ハードウェア、オペレーティング システム、ドメイン)。
- クラッシュ時のメモリーのコンテンツのスナップショット (データ接続に関する情報などのアプリケーション アクティビティの詳細、Tableau でユーザーが実施すべきアクション、Tableau で作業するデータを含む)。
- Tableau 情報には、顧客を識別可能な情報が含まれています。

サーバー クラッシュレポーターの構成

サーバー クラッシュレポートは既定で無効化されています。このトピックでは、サーバー クラッシュレポートを有効化および構成する方法について説明します。クラッシュレポートは、暗号化されて Tableau に送られます。詳細については、サーバー クラッシュレポーターを参照してください。

組織でプロキシ サーバーを使用してインターネットに接続する場合は、サーバー クラッシュ レポーターでプロキシを使用するように構成する必要があります。Tableau Server でプロキシを使用するように既に構成している場合でも、別途、サーバー クラッシュ レポーターも構成する必要があります。サーバー クラッシュ レポーターでプロキシを構成するには、このトピックでの説明に従って TSM CLI の手順を行ってください。

重要: データがプライバシー規制の対象となる場合は、クラッシュ レポートを有効化しないでください。

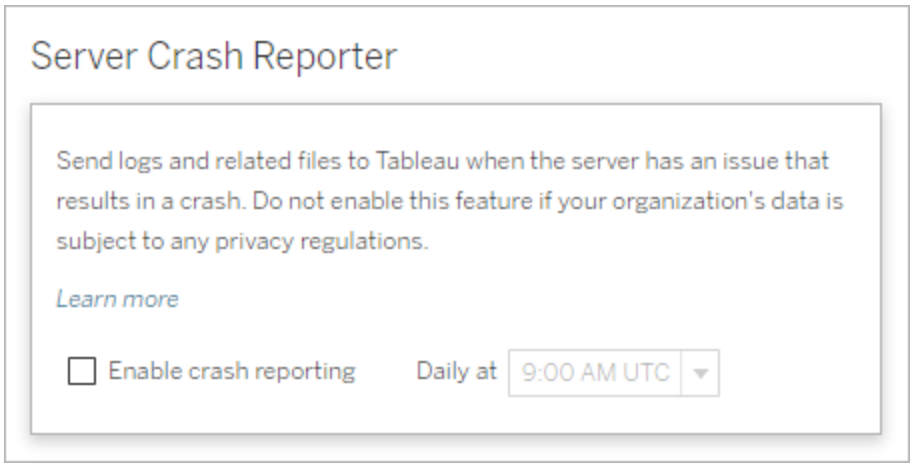
TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[メンテナンス]** タブをクリックします。

3. [その他のメンテナンス タスク] の [サーバー クラッシュ レポーター] で、**[クラッシュ レポートの有効化]** を選択します。



4. スケジュールされた時刻を指定して、クラッシュ レポートを Tableau にアップロードします。

5. 完了したら、**[変更を保留中]** をクリックしてから **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

既定値: `tsm settings import -f path-to-file.json`

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

構成ファイルでは configKey クラスが使用されているため、configEntities クラスを使用する場合とは異なり、渡した値が TSM によって検証されません。tsm configuration コマンドを使用することにより、個々のオプションを確認して設定することができます。

クラッシュレポーターの設定

次のテンプレート内のクラッシュレポーターの設定では、Tableau Server がクラッシュレポートを Tableau に送るように構成するためのさまざまなオプションが指定されています。

構成テンプレート

このテンプレートを使用してゲートウェイの設定を構成します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

```
{
       "configKeys": {
        "servercrashupload.enabled": "true",
        "servercrashupload.scheduled_time": "1:00:00 UTC",
        "servercrashupload.proxy_server_host": "",
        "servercrashupload.proxy_server_port": "",
        "servercrashupload.proxy_server_username": "",
```

```
        "servercrashupload.proxy_server_password": "",
        "servercrashupload.preserve_upload_packages": "false",
        "servercrashupload.delete_completed_dumps": "false"
        }
}
```

構成ファイル リファレンス

この表には、クラッシュレポートを構成するために設定するキーが含まれています。

`servercrashupload.enabled`

既定: `false`

`true` に設定するとクラッシュレポートが有効になります。

`servercrashupload.scheduled_time`

既定: `1:00:00 UTC`

クラッシュアップロードの開始をスケジュールする時刻を指定します。24 時間形式で時刻を入力します。

`servercrashupload.proxy_server_host`

組織がプロキシ サーバーを使用してインターネットと通信している場合、ホスト名を指定します。

`servercrashupload.proxy_server_port`

組織がプロキシ サーバーを使用してインターネットと通信している場合、ポート番号を指定します。

`servercrashupload.proxy_server_username`

プロキシ サーバーで認証が必要な場合、このキーでユーザー名を指定します。

`servercrashupload.proxy_server_password`

プロキシ サーバーで認証が必要な場合、このキーでパスワードを指定します。

`servercrashupload.preserve_upload_packages`

既定: `false`

クラッシュレポート用に作成されるパッケージをすべて保存するには、このキーを `true` に設定します。

既定では、パッケージは `/var/opt/tableau/tableau_server/data/tabsvc/clustercontroller/tabcrashreporter` に保存されます。

`servercrashupload.delete_completed_dumps`

既定: `false`

すべてのダンプを送信後に削除するには、このキーを `true` に設定します。

Tableau Web 環境の管理者エリアの移動

Tableau Server または Tableau Cloud の管理者は、他のユーザーが利用できない管理者設定にアクセスし、サイト、ユーザー、プロジェクトの構成や他のコンテンツ関連のタスクを行うことができます。

この記事の設定では、Tableau Web 環境について記載しています。適切な認証資格情報が設定されている Tableau Server 管理者は、プロセッサー、キャッシュ、認証、分散展開、関連する構成など、TSM Web 環境を使用してサーバー設定を変更することもできます。詳細については、Tableau サービス マネージャーの Web UI へのサインインを参照してください。

サイト ロールとサイト数に基づくアクセス

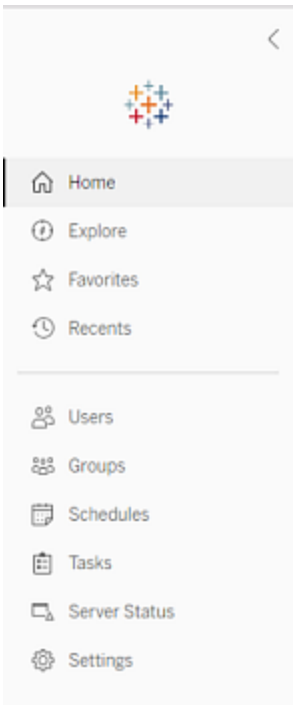
Tableau Server または Tableau Cloud にサインインすると利用できるメニューは、以下の条件によって変わります。

- サイト管理者やサーバー管理者かどうか。

  サイト管理者アクセスは Tableau Cloud および Tableau Server で利用できます。サーバー管理者アクセスは Tableau Server のみで利用できます。

- アクセスできるサイトは 1 つのみ、または複数なのか。

サーバー管理者

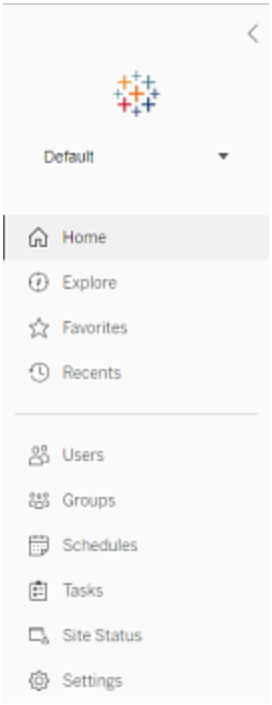
単一サイトサーバーではサイトセレクターが表示されず、他のすべてのメニューは同様です。	

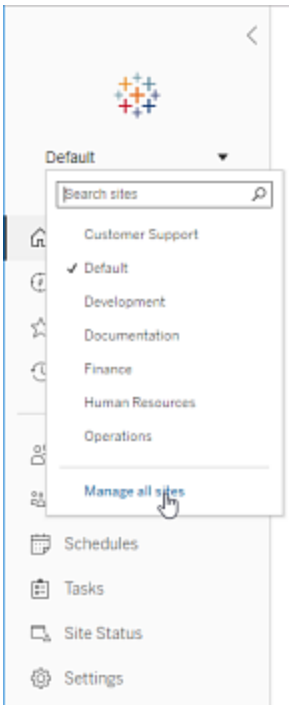
複数サイト環境では、左側のメニューにより、特定のサイトやすべてのサイトを修正したり、ユーザー、グループ、スケジュール、タスク、サーバー設定を構成したりすることができます。

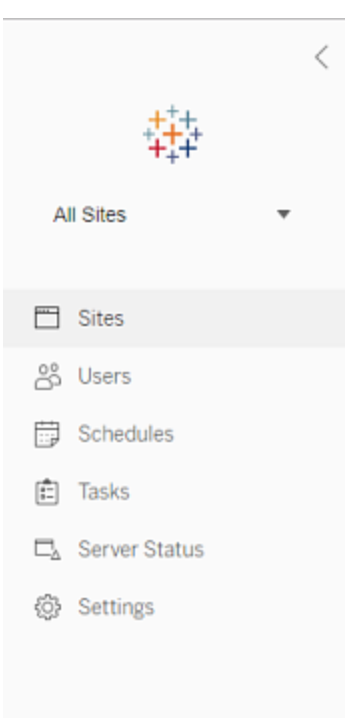
すべてのサイトに影響するサーバー管理者設定にアクセスするには、現在のサイト名の横にある矢印をクリックしてサイト メニューを開き、**[すべてのサイトを管理]** を選択します。

[コンテンツ] タブと**[グループ]** タブがなくなるほか、サイト メニューのテキストが **[すべてのサイト]** に変わり、サーバー全体の設定を管理していることを示します。また、**[サーバー ステータス]** などのオプションにサーバー全体の視点が反映されます。

サイト管理者 メニューに戻るには、**[すべてのサイト]** を選択して管理するサイトを選択します。





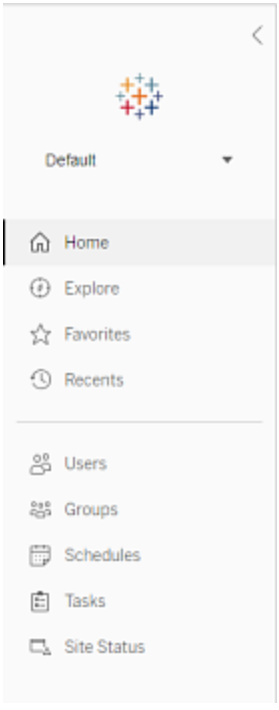


サイト管理者

Tableau Cloud または Tableau Server のサイト管理者であり、複数サイトへのアクセス権を持っている場合は、管理するサイトを選択するメニューや、そのサイトのコンテンツ、ユーザー、グループ、スケジュール、タスクを管理するメニュー、およびそのステータスをモニタリングするメニューが利用できます。

サイトセレクターには、現在のサイト名が表示されます。別のサイトに移動するには、サイトメニューを選択してからサイト名を選択します。

1 つのサイトのみへのアクセス権を持っている場合は、サイトセレクターは表示されませんが、他のすべてのメニューは同様です。



サーバー管理者のタスク

サーバー管理者 (Tableau Server Enterprise でのみ利用可能) は、次の操作を実行できます。

- サーバーのステータスとアクティビティを監視する。
- ログ ファイルを生成する。
- サイトを追加し、サイト設定を編集する。サーバー管理者のみが、サーバーにサイトを追加できます。
- サーバーにユーザーを追加し、ユーザーをサイトに割り当てる。
- サイト グループを追加および管理する。
- アイデンティティ プールにユーザーを追加/削除します。

特定サイトでのみ設定を管理するには、まずはそのサイトに移動する必要があります。各サイト内では、次の操作を実行できます。

- 管理者 コンテンツ: プロジェクトを作成する、あるプロジェクトから別のプロジェクトにコンテンツを移動する、パーミッションを割り当てる、コンテンツ リソースの所有権を変更するなど。
- 抽出更新とサブスクリプションのスケジュールを管理する。
- サイトの活動を監視し、ワークブック パフォーマンス メトリクスを記録する。
- ユーザーがパブリッシュするコンテンツ用に、保存スペースの制限を管理する。
- Web 作成を許可する。
- リビジョン履歴を有効にする。
- サイト管理者に、ユーザーの追加および削除を許可する。
- サイトが各ライセンス タイプ (Creator、Explorer、Viewer) に対して使用できるライセンスの最大数を設定します。

- ユーザーにワークブックおよびビューのサブスクライブを許可し、コンテンツの所有者に他のメンバーにコンテンツのサブスクライブを許可する。
- お気に入り用のオフライン スナップショットを有効にする(iOS のみ)。

サイト管理者のタスク

Tableau Cloud または Tableau Server のサイト管理者は、次の操作を実行できます。

- 管理者コンテンツ: プロジェクトを作成する、あるプロジェクトから別のプロジェクトにコンテンツを移動する、パーミッションを割り当てる、コンテンツ リソースの所有権を変更するなど。
- 抽出更新およびサブスクリプションのスケジュールを表示、管理、および手動で実行する。
- サイトユーザーを追加および管理する(サーバー管理者によって許可されている場合。サイト設定 リファレンスを参照してください)。
- サイトグループを追加および管理する。
- サイトアクティビティを監視する。

Tableau Server を別のドライブに移動させる

Tableau Server を別のドライブに移動する必要がある場合(たとえば、新しいポリシーによりシステムドライブにアプリケーションデータを保有できない場合、または元のドライブの領域が不足している場合など)、以下の手順を実行できます。ステップは Tableau Server を同じコンピューター上のあるドライブから別のドライブに移動させる場合の例であり、お使いのインストールや構成を正確に反映していない場合があります。新しいコンピュータに Tableau Server を移動する手順については、「新しいハードウェアへの移行」を参照してください。

はじめる前に

開始前に、次を確認します。

- 既存のインストールからのデータとアセットの現在のバックアップと、設定のエクスポートがある。新しいドライブでの Tableau Server のインストールを復元するためにこれが'必要になりま

す。バックアップ作成の詳細については、Tableau Server の完全なバックアップと復元を実行しますを参照してください。

稼働中のサーバーと同じタイプのアイデンティティストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。

- tsmadmin グループのメンバーアカウントを使用して、Tableau Server コンピューターにアクセスできる。ここで作業を行うため、tsmadmin グループのメンバーアカウントを使用して、このコンピューターにログオンする必要があります。
- 移動する Tableau Server のバージョンのインストールプログラムのコピーを持っている。Tableau Server を新しいドライブにインストールする必要があります。
- 現在のインストールで使用している認証方法を知っている。例えば、Tableau Server 例: が SSL, SAML、または Kerberos 用に構成されている場合、関連証明書またはキータブファイルのバックアップを個別に作成し、Tableau Server を再インストールした後でそれらを新しいドライブにコピーする必要があります。
- 現在のインストールにおける初期ノード設定と構成を知り、理解している。

Tableau Server を別のドライブに移す準備が整い、データとすべてのアセットを完全にバックアップし、これらのファイルを別のコンピュータ上の安全な場所に保存する準備 ができたら、

1. tsmadmin グループのメンバーアカウントを使用して、最初のノードでターミナル セッションを開きます。
2. スクリプト tableau-server-obliterate を実行します。

   ```
   sudo /opt/tableau/tableau_server/packages/scripts.<version_
   code>/tableau-server-obliterate -a -y -y -y
   ```

   Tableau Server をマルチノードに (分散) インストールしている場合は、クラスタ内の各ノードでスクリプト tableau-server-obliterate を実行します。

3. `tableau-server-obliterate` スクリプトを実行したコンピューターを再起動します。
4. 新しいディレクトリに Tableau Server をインストールして、初期化します。詳細については、「TSM のインストールと初期化」を参照してください。
5. Tableau Server のライセンス認証と登録。
6. (オプション)。ローカル ファイアウォールの構成。
7. (オプション)。LDAP を確認します。
8. Tableau Server を初期化します。初期ノード設定の構成を参照してください。
9. バックアップ ファイル *.tsbak* を `basefilepath.backuprestore` 変数で指定したフォルダーの場所にコピーします。既定の場所は、`/var/opt/tableau/tableau_server/data/tabsvc/files/backups/` です。

   > **注:** バックアップ ファイルの場所は変更することができます。詳細については、tsm ファイル パスを参照してください。

10. Tableau Server を停止します。

    ```
    tsm stop
    ```

11. 以前に保存したTableau Serverデータバックアップとアセットを復元します。これには、データベースの復元、および手動で復元する必要がある資産の復元が含まれます。

    詳細については、コア Tableau Server 機能の復元を参照してください。

12. Tableau Server を起動します。

    ```
    tsm start
    ```

Tableau Server 製品のプロダクト キーのライセンス認証

Tableau Server の同じプロダクト キーを最大で 3 回ライセンス認証できます。これにより、サンドボックス環境や QA 環境で Tableau Server をテストでき、実稼働環境で Tableau を使用すること

もできます。ライセンス認証を最大限に行うには、同じコンピューターに Tableau を再インストールしない限り、Tableau Server をコンピューターから削除する際にプロダクトキーのライセンス認証を解除する必要があります。これにより、別のコンピューターでライセンス認証を行うことができます。たとえば、あるコンピューターまたは VM から別のコンピューターに Tableau Server を移動する場合は、プロダクトキーのライセンス認証を解除します。その後、元のコンピューターから Tableau を削除します。新しいコンピューターに Tableau をインストールするとき、競合を発生させずにキーをライセンス認証できます。ロールベースライセンスを使用する場合は、Tableau への管理者アクセス権を失う可能性がありますので、必ず Creator キーまたは Explorer キーをアクティブにしてください。同じコンピューターに再インストールするために Tableau Server を削除する場合は、キーのライセンス認証を解除する必要はありません。再インストールされた時点で Tableau はこのキーを使用します。たとえば、コンピューター上のあるドライブから同じコンピューターの別のドライブに Tableau を移動する場合などです。プロダクトキーのライセンス認証を解除する方法の詳細については、tsm licenses deactivate を参照してください。

Obliterate スクリプトを使用して Tableau Server を削除する場合は、スクリプトを無効化し、"`-l` スイッチを使用して Tableau プロダクトキー情報を削除することができます。これにより、インストールしている場合は Tableau Desktop を含むすべての Tableau ライセンス情報が無効化され、削除されます。キーのライセンス認証を解除する場合は、スクリプトを実行する*前*にキーのライセンス認証を解除することをお勧めします。詳細については、コンピュータから Tableau Server を削除するを参照してください。

分散型高可用性 Tableau Server インストール

このトピックでは、Tableau Server インストールの異なる展開タイプについて説明します。

サーバー展開計画の検証

組織に新しい Tableau Server 展開をインストールする前に、オプションを注意深く評価する必要があります。ほとんどの組織の場合、Tableau Cloud は、セルフホストしている Tableau Server よりも信頼性、パフォーマンス、コスト効率が高い分析ソリューションを提供します。組織に対する Tableau Cloud の有効性に関する詳細については、ブログ投稿「分析を Cloud に移行する必要がありますか (英語)」を参照してください。

すでに Tableau Server を実行していて、Tableau Cloud への移行を検討している場合は、「Tableau Cloud 手動移行ガイド」を参照してください。

インストールのタイプ

Tableau Server を実行する最も基本的な方法はそれを単一のノードにインストールすることです。この種のインストール環境では、完全に機能する Tableau Server と、すべての Tableau サービスマネージャー (TSM) および Tableau Server プロセスがその単一のノード上で実行されますが、これは Tableau Server を使用する最適な方法ではないかもしれません。所属組織のニーズおよびお手持ちのリソースに従い、Tableau のインストール方法を決定できます。ノードを追加したり、高可用性向けに Tableau を構成したりすることもできます。インストール オプションには次のものが含まれます。

- **シングル ノード インストール**—この種のインストール環境は、冗長性に欠けるため、テスト、トライアルの実行、時折のダウンタイムおよびシステムの可用性を許容できる環境に向いています。すべてのサーバープロセスは単一のマシン上で実行されています。サーバープロセスの 1 つに問題が発生した場合の冗長性および予防措置が少なくなっています。また、Tableau Server をインストールするコンピューターに、プロセスおよびユーザーとデータの要求を取り扱うために適切なリソースを確保する必要もあります。

- **分散インストール**—この種のインストールはマルチノード インストールとも呼ばれ、複数のコンピューターを必要として、それらの分散ノード上にサーバープロセスをインストールおよび実行します。サーバープロセスを複数ノード上に広げることで、冗長性とさらなるコンピューティングパワーを提供し、Tableau Server の信頼性と効率性を拡張できます。正しい構成をすれば、分散インストールは自動リポジトリフェールオーバー機能も提供できます。詳細については、リポジトリフェールオーバーを参照してください。

- **高可用性 (HA) インストール**—Tableau Server の HA インストールとは、3 つ以上のノードと主要プロセス (リポジトリ、ファイル ストア/データ エンジン (Hyper)、調整サービス、クライアント ファイル サービス) の複数インスタンスが異なるコンピューター上に含まれる特殊なマルチノード インストールです。HA インストールでは、それらの主要プロセスに、複数ファイル ストアおよび自動リポジトリフェールオーバーを含む組み込まれた冗長性があります。目的は、単一障害点をなくし、可能な場合はフェールオーバーの失敗を検出可能にして、システム ダウンタ

イムを最小限に抑えることです。

最初のノードに障害が発生した場合、またはアプリケーション サーバー (VizPortal) を実行しているノードが障害から回復中の場合は、ダウンタイムが発生する可能性があります。ダッシュボードとビューの読み込みが予想よりも遅くなる場合があり、システムの構成方法と使用方法によってはタイムアウトが発生する可能性があります。最初のノードの障害については、以下の最初のノードに障害が発生した場合を参照してください。

Tableau を最初にインストールするコンピューターである「初期ノード」には、固有の特徴があります。初期ノード上で実行できるのは、ライセンス サービス (ライセンス マネージャー)、ライセンス認証サービス、TSM コントローラー (管理コントローラー) の 3 つのプロセスのみで、エラーが発生した場合を除き、これらのプロセスを他のノードに移動させることはできません。Tableau Server には、これらのプロセスを他のいずれかの既存ノードに移動するのを自動化するスクリプトが含まれています。これにより、TSM に戻るための完全なアクセス権を取得し、Tableau Server の実行を維持できるようになります。

他の 2 つのプロセスはまず初期ノードに含められますが、追加のノード、CFS (Client File Service)、調整サービスに追加・移動させることができます。インストールが CFS および調整サービスでどのように構成されていたかによって、これらを再デプロイする手順も必要になる場合があります。

ライセンス サービスおよび TSM コントローラーを最初のノードから別のノードに移動する方法については、下の最初のノードの障害を復旧するを参照してください。

外部リポジトリ

Tableau Server のパフォーマンスを最適化するには、展開内の専用ノードでリポジトリを分離することをお勧めします。Advanced Management ライセンスをお持ちの場合は、リポジトリを外部データベースとして実行することを検討してください。

組織のピーク負荷が 1 時間あたり 1000 VizQL セッションを超える場合は、Tableau Server on Linux を実行することもお勧めします。このシナリオでは、VizQL セッションは、Tableau Server からビジュアライゼーションを表示または生成するユーザー アクションのことを指します。

詳細については、「Tableau Server 外部リポジトリ」を参照してください。

前提条件

ここに示す指示はお使いのクラスタが分散環境の要件を満たしていることを前提にしています。

マルチノード クラスタ内のすべてのノードには、同じタイプのオペレーティング システムと、そのオペレーティング システムと同じメジャー バージョンが必要です。たとえば、すべての RHEL 9 ノードなどです。

Linux ノードとWindows ノードの組み合わせて使用している場合、Tableau Server のマルチノード インスタンスをインストールすることはできません。

ライセンス発行

有効な Tableau Server プロダクト キーが必要です。使用しているライセンスの種類によって、Tableau をインストールできるノードの数が決まる場合があります。ライセンス発行の詳細については、ライセンス発行の概要を参照してください。

分散 Tableau Server インストール環境の作成

Tableau Server の分散インストール環境を作成するための一般的な手順を示します。

1. 最初のノード上に Tableau Server をインストールすることから始めます。

   詳細については、Tableau Server のインストールと構成を参照してください。

2. 最初のノード上でノード構成 (bootstrap) ファイルを生成します。

   詳細については、ノードbootstrap ファイルの生成を参照してください。

3. bootstrap ファイルを使用して、Tableau Server を追加のノードにインストールします。

   詳細については、追加ノードのインストールと初期化を参照してください。

4. 追加のノードを、その上で実行したいプロセスで構成します。

   詳細については、追加のノードを構成するを参照してください。

5. インストールしたい追加のノードでステップ 3 および 4 を繰り返します。

6. 新しい調整サービス アンサンブルを展開します。

   詳細については、調整サービス アンサンブルの展開を参照してください。

7. 調整サービスを実行しているすべてのノードにクライアント ファイル サービスを追加します。

   詳細については、クライアント ファイル サービスの構成を参照してください。

高可用性 (HA) Tableau Server インストール環境の作成

高可用性 Tableau Server インストール環境とは、特殊な分散型インストール環境であり、完全なサーバー機能性を失うことなく主要なサーバー コンポーネントでの障害に対応するように設計されています。HA インストール環境を作成するための手順は、分散型展開を作成するために行う内容と同様のものですが、展開に高可用性を持たせるための追加の手順も含まれます。このような追加の手順には少なくとも 2 台の追加のノードを加えること(クラスタ内の最小ノード数は 3 台であるため)、リポジトリに 2 番目のインスタンスを追加すること、追加のゲートウェイ プロセスを加えること、および調整サービス アンサンブルを展開することが含まれます。さらに、ロード バランサーを追加してゲートウェイカンで要求を割り振ることもできます。

高可用性の Tableau Server インストール環境を作成するための手順の概要を以下に示します。

1. 最初のノード上に Tableau Server をインストールすることから始めます。

   詳細については、Tableau Server のインストールと構成を参照してください。

2. 最初のノード上でノード構成 (bootstrap) ファイルを生成します。

   詳細については、ノード bootstrap ファイルの生成を参照してください。

3. ノードの bootstrap ファイルを使用して、Tableau Server を追加 2 台のノードにインストールします。

   詳細については、追加ノードのインストールと初期化を参照してください。

4. 各追加ノードを、その上で実行したいプロセスで構成します。これらプロセスには 2 番目の Tableau Server リポジトリのコピー、2 番目のデータエンジンおよびファイル ストア、さらにゲートウェイの追加インスタンスが含まれなくてはなりません。

   詳細については、追加のノードを構成するを参照してください。

5. 調整サービス アンサンブルを展開します。

   詳細については、調整サービス アンサンブルの展開を参照してください。

6. 調整サービスを実行しているすべてのノードにクライアント ファイル サービスを追加します。

   詳細については、クライアント ファイル サービスの構成を参照してください。

7. (オプション) ロード バランサーを構成します。

   詳細については、ロード バランサーの追加を参照してください。

3 ノード HA インストール環境を作成する方法の詳細については、例: 3 ノード HA クラスタのインストールおよび構成を参照してください。

最初のノードに障害が発生した場合

初期ノードに問題があり、他のノードに冗長プロセスがある場合、Tableau Server が引き続き実行できるという保証はありません。

- Tableau Server は、第一ノードに障害が発生してから最大 72 時間、ライセンス サービスの欠如によって他のプロセスが影響を受け始める前まで、実行を継続することができます。初期ノードでエラーが発生しても、ユーザーは引き続きサインインしてコンテンツを表示および使用することはできますが、管理コントローラーへのアクセス権がなくなるため、Tableau Server の再構成ができなくなります。
- ATR 用に構成された Tableau Server 2021.4.2 (またはそれ以前) のバージョンを実行している場合、第一ノードに問題が発生すると、すべてのサーバー機能が使用できなくなります。これは、初期ノードに問題がある場合でも、意図的に初期ノードを停止している (たとえば、システムレベル パッチを適用するために) 場合でも当てはまります。

冗長プロセスで構成されている場合でも、*初期ノードに障害が発生すると、Tableau Server が機能し続けることができなくなる可能性があります。*これは、インストールが高可用性用に構成されている場合でも当てはまります。つまり、できるだけ早く、2 つの固有のプロセスを別の実行中のノードに移動させる必要があります。相対的に時間が足りず復元できないという理由で初期のノードに失敗が起きる場合 (修正できるハードウェアのエラーなど)、まずは以下の手順を実行せずにノードのバックアップを試行する必要があります。

最初のノードの障害を復旧する

Tableau Server をインストールすると、最初のノードにはそのノードだけにインストールされる 2 つのサービス (ライセンス サービスとTSM コントローラー) が含まれています。最初のノードで問題が発生した場合、Tableau Server は、高可用性を実現するように構成されていたとしても、機能を継続できない場合があります。TSM コントローラーとライセンス サービスを既存の構成されたノードに移動することで、最初のノードに障害が起きた状況から復旧する事ができます。これにより、クラスタ内に既に所有しているリソースを使って障害から復旧できます。最初のノードに障害が発生したときに備えて、スタンバイの最初のノードを構成しておかないですみます。

最初のノードの障害から復旧する方法について詳細は、初期ノードの障害からの回復を参照してください。

追加のノード上で調整サービス アンサンブルを構成する

複数ノードに調整サービスを構成すると、プロセスの複製が追加されるため、調整サービス ノードのいずれかの問題に起因してサーバーダウンタイムが生じる可能性が低くなります。クラスタに調整サービス アンサンブルを展開する方法の詳細については、調整サービス アンサンブルの展開を参照してください。

追加のノードにクライアント ファイル ・サービス (CFS) を追加する

Tableau Server には、クライアント ファイル ・サービス (CFS) の少なくとも 1 つのインスタンスが必要です。CFS の追加のインスタンスを追加すると、プロセスの複製が追加されるため、いずれかの CFS ノードの問題に起因するサーバーダウンタイムが生じる可能性が低くなります。調整サービス

を展開する各ノードで CFS のインスタンスを構成することをお勧めします。他のノードで CFS を構成する方法の詳細については、クライアント ファイル サービスの構成を参照してください。

Tableau Server サービス ライセンス チェック

Tableau Server をインストールする際には、多数のプロセスがインストールされます。これらのプロセスには、有効な Tableau Server ライセンスを必要としているものと、必要としていないものがあります。有効な Tableau Server ライセンスが必要な Tableau Server のサブセットは、"ライセンス プロセス" と見なされます。

ライセンス プロセスが開始または再開するとき、プロセスは初期ノードのTableau Serverライセンス マネージャーに有効なライセンスがあるかどうかを確認します。ライセンス マネージャーがライセンスを検証する際、プロセスは完全に機能し、他の Tableau Server プロセスからの要求に対応できます。ライセンス マネージャーからライセンス プロセスに確認が届いたら、プロセスは 72 時間、またはプロセスが再開するまでライセンスを再確認する必要はありません。プロセスがライセンスを検証できない場合 (たとえば、プライマリ ノードが使用できない場合など) は実行することはできますが、ライセンスが確認されるまで引き続き有効なライセンスのチェックを行います。ライセンス確認が最後にいつ行われたかを調べるには、`/var/opt/tableau/tableau_server/data/tabsvc/logs/licenseservice`ディレクトリのログファイルを参照してください。ライセンス プロセスについての詳細情報は、ライセンス プロセスを参照してください。

分散環境の要件

Tableau Server クラスタの構成を開始する前に、次の要件を満たしていることを確認してください。

ハードウェア

クラスタで使用するコンピューターが インストールの前に... で説明されている要件を満たしている必要はありますが、全く同じである必要はありません。

高可用性を得るためのハードウェアのガイドライン

これらは、フェールオーバーと高可用性 を得るために使用するシステムのいくつかのガイドラインです。

- **フェールオーバー — 3 台のコンピューター**: ファイル ストア プロセスとリポジトリプロセスにフェールオーバー サポートを提供するクラスタを構成するには、最低 3 台のコンピューターまたは VM が必要です。1 台は最初の Tableau Server ノード用で、もう2 台は追加ノード用です。

- **複数のゲートウェイ — 3 台のコンピューターおよびロード バランサー**: Tableau Server のインストールに複数のゲートウェイのプロセスを追加し、ロードバランサーを使用して自動的に要求をそれらのゲートウェイに分配すると、Tableau の信頼性をさらに向上できます。フェールオーバーおよび複数ゲートウェイをサポートするクラスタを構成するには、3 ノードのクラスタの前にロードバランサーを 1台追加する必要があります。

- **フェールオーバーおよび複数のゲートウェイのサポート—3 台のコンピューターおよびロード バランサー**:上記に加えて複数のゲートウェイのサポートを提供するクラスタを構成するには、最低 3 台のコンピューターまたは VM、およびクラスタの前面にロード バランサーが必要です。

- **高可用性 — 3 台のコンピューターとロード バランサー**: 高可用性が得られるよう構成するには、上記のリソースが必要です。

- **最初のコンピューター**: 高可用性を得るための構成では、最初の Tableau Server ノードで実行する Tableau Server プロセスを少なくするか、またはなしにする必要があります。このため、最初のノードとして機能するコンピューターには、追加ノードを実行するコンピューターほど多くのコアは必要ありません。ただし、バックアップおよび復元プロセスの途中に最初のコンピューターが使用されるため、バックアップのために十分なディスク領域が必要です。バックアップ ファイルに必要な領域の容量に加えて、一時ディスク領域が必要です。ディスク容量の要件の詳細については、バックアップと復元のディスク容量使用率 を参照してください。

マルチノード サーバーのソフトウェア要件

- 分散インストールでは、すべてのノードが同じバージョンの Tableau Server を実行している必要があります。

- マルチノード クラスタ内のすべてのノードには、同じタイプのオペレーティング システムと、そのオペレーティング システムと同じメジャー バージョンが必要です。たとえば、すべての RHEL 9 ノードなどです。

  Linux ノードと Windows ノードの組み合わせて使用している場合、Tableau Server のマルチノード インスタンスをインストールすることはできません。

インストール場所

次の要件と制限に注意が必要です。

- Tableau Server をインストールする場所とデータディレクトリの場所は、クラスタ内のすべてのノードで同じである必要があります。
- RHEL ライクのディストリビューションではデフォルト以外のインストール ディレクトリを指定できますが、Ubuntu では場所を変更できません。詳細については、インストール ディレクトリを参照してください。
- Tableau を初期化すると、データディレクトリにデフォルト以外の場所を指定できます。詳細については、データディレクトリを参照してください。

ネットワークとポート

- **ポート**:あらゆる分散システムに該当することですが、使用するコンピューターまたは VM は、相互に通信できる必要があります。Tableau サービス マネージャーによるポート マッピングの処理方法の詳細については、Tableau サービス マネージャーのポートを参照してください。

- **遅延**: サーバー ノード間のネットワーク遅延は Tableau Server のパフォーマンスに影響を与える可能性があります。特にパフォーマンスの問題が発生した場合は、発生しうる遅延の問題に注意してください。ネットワーク遅延を減らすために、Tableau Server に近接するゲートウェイやデータ ソースを見つけるなどの手順を実行できます。

- **静的 IP アドレス**: 単一サーバーのインストールまたはクラスタの一部であっても、Tableau Server を実行するいずれのコンピューターも、静的 IP アドレスを保持している必要があります。

- **検出可能**: ノード内の各クラスタは、DNS またはローカル ホスト ファイルを使用して他のノード コンピューターから検出できる必要があります。
- **タイムゾーンと時刻**: クラスタ内の各ノードは同じタイムゾーンに属し、それぞれのシステム クロックが同期されている必要があります。これは自動的に行われる場合もあります。たとえば、ノードがすべて Active Directory ドメインに属する場合は、通常、ドメイン コントローラーがこれを処理します。クラスタがこの要件を満たしているかわからない場合は、内部の IT 専門家に問い合わせてください。

ベストプラクティス

インストールと構成を開始する前に念頭に置いておくべきことをいくつか挙げます。

- **IP アドレスまたはコンピューター名**: 前述のように、クラスタの各コンピューターは静的 IP アドレスを使用する必要があります。
- **バックアップ**:ベストプラクティスとして、システムに大幅な変更を加える前に、バックアップを作成しておくことをお勧めします。手順については、Tableau Server データのバックアップを参照してください。

SSL

複数のゲートウェイおよびロード バランサーを使用する高可用性 Tableau Server クラスタのために SSL を構成する予定であれば (詳細情報)、使用する SSL 証明書がロード バランサーのホスト名で発行されていることを確認してください。他の詳細については、Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成するを参照してください。

分散インストールの推奨事項

Tableau Server インストールにノードを追加するときは、各コンピューター上で実行するプロセスの数を決定する必要があります。このページでは、開始点のみを対象とする一般的な推奨事項をいくつか提供します。

これらの一般的な推奨事項に加えて、以下のことにも従う必要があります。

- 組織によるTableau Server の使用方法を理解し、使用事例 (ユーザー応答や抽出の更新用に最適化する場合など) に合わせて設定を調整します。
- 徹底的なパフォーマンステストを実施し、プロセスの設定を調整するのに最適な場所を特定します。

Tableau Server インストールを組織のニーズに合わせて調整する方法についての詳細は、「パフォーマンスの調整」を参照してください。

分散インストールの要件、および追加ノードの構成の詳細については、分散型高可用性 Tableau Server インストールを参照してください。

すべてのインストールでの推奨事項

Tableau Server クラスターを構成するコンピューターのハードウェアは同じでなくても問題ありませんが、すべてが最小システム要件を満たしている必要があります。このページのすべての推奨事項は、Tableau Server をインストールしたコンピューターにコアが 8 つ以上あることを想定しています。

次の推奨事項は、すべてのサーバー設定に適用されます。

- バックグラウンダープロセスを専用コンピューターで実行します (抽出を頻繁に更新する予定がある場合)。一般的にバックグラウンドプロセスは CPU を大量に使用し、同じコンピューター上の他のプロセスの速度が低下する場合があります。
- 抽出を頻繁に更新する予定の場合や、大きな抽出を更新する予定の場合は、バックグラウンダープロセスのプロセス数を増やしてください。
- VizQL プロセスをバックグラウンダープロセス以外のコンピューターで実行します。それらを同じマシン上で保持すると、抽出の更新がユーザー ビューに影響することがあります。
- ファイル ストアがインストールされているノードにインストールされたデータ エンジンのインスタンスは、ビューの要求のデータをクエリするために使用されます。バックグラウンダー タスクによるユーザー ビューへの影響を最小化するには、ファイル ストア プロセスをバックグランダープロセスから分離することを検討してください。
- **トポロジの構成を使用した最適化:**

- 管理コントローラーと同じノード上にファイル ストアを配置すると、バックアップ プロセス中にノード間でデータを転送する必要性が減少するか排除されるため、Tableau Server のバックアップにかかる時間を短縮できます。これは、組織で多数の抽出を使用している場合に特に当てはまります。
- リポジトリ(pgsql) を管理コントローラー ノードと同じ場所に配置すると、バックアップ時間は短縮できますが、ファイル ストアの場合ほど時間の節約は顕著ではありません。

最初のノードでエラーがあった場合や、コントローラーを別のノードに移動した場合を除き、管理コントローラーは通常最初のノード上にあります。

注: 3 ノード以上の分散インストールでは、最大 2 つのリポジトリ(アクティブおよびパッシブ) のインスタンスを持つことができます。また、1 つのリポジトリで Tableau Server を実行することもできますが、この操作を行うとそのリポジトリで利用可能なフェイルオーバーはありません。詳細については、Tableau Server リポジトリを参照してください。

追加のノードのインストールと構成

1台のコンピューター (または ノード) に Tableau Server をインストールした後、サーバーは機能していて仕様の準備が完了していますが、冗長性がありません。プロセスまたはコンピューター自体に問題が発生した場合、Tableau Server が利用できなくなる可能性があります。さらに、すべてのプロセスが1台のコンピューター上で実行されているため、そのコンピューター上でリソースの輻輳が発生するかもしれません。

追加のノードに Tableau を追加することで、分散インストールの状態を作り、Tableau Serverのインストールを拡張できます。この記事は、最初のノードに既に Tableau をインストールしてあるという前提で、追加のノードに Tableau Server をインストールする一般的な手順を説明します。最初のノードに Tableau をインストールすることについての詳細は、TSM のインストールと初期化を参照してください。

複数のノードに Tableau Server をインストールしている場合は、ノードを一度に 1 つずつインストールおよび構成する必要があります。このようにした方が、問題が発生したときに解決しやすくなります。

重要: 保留中の変更を適用することによりプロセスをすべて完了する場合、追加ノードを追加して構成する必要があります。保留中の変更を適用して終了せずにノードを追加すると、ユーザーが Tableau Server にログインできない可能性があります。

インストール場所

次の要件と制限に注意が必要です。

- Tableau Server をインストールする場所とデータディレクトリの場所は、クラスタ内のすべてのノードで同じである必要があります。
- RHEL ライクのディストリビューションではデフォルト以外のインストール ディレクトリを指定できますが、Ubuntu では場所を変更できません。詳細については、インストール ディレクトリを参照してください。
- Tableau を初期化すると、データディレクトリにデフォルト以外の場所を指定できます。詳細については、データディレクトリを参照してください。

TSM Web インターフェイスの使用

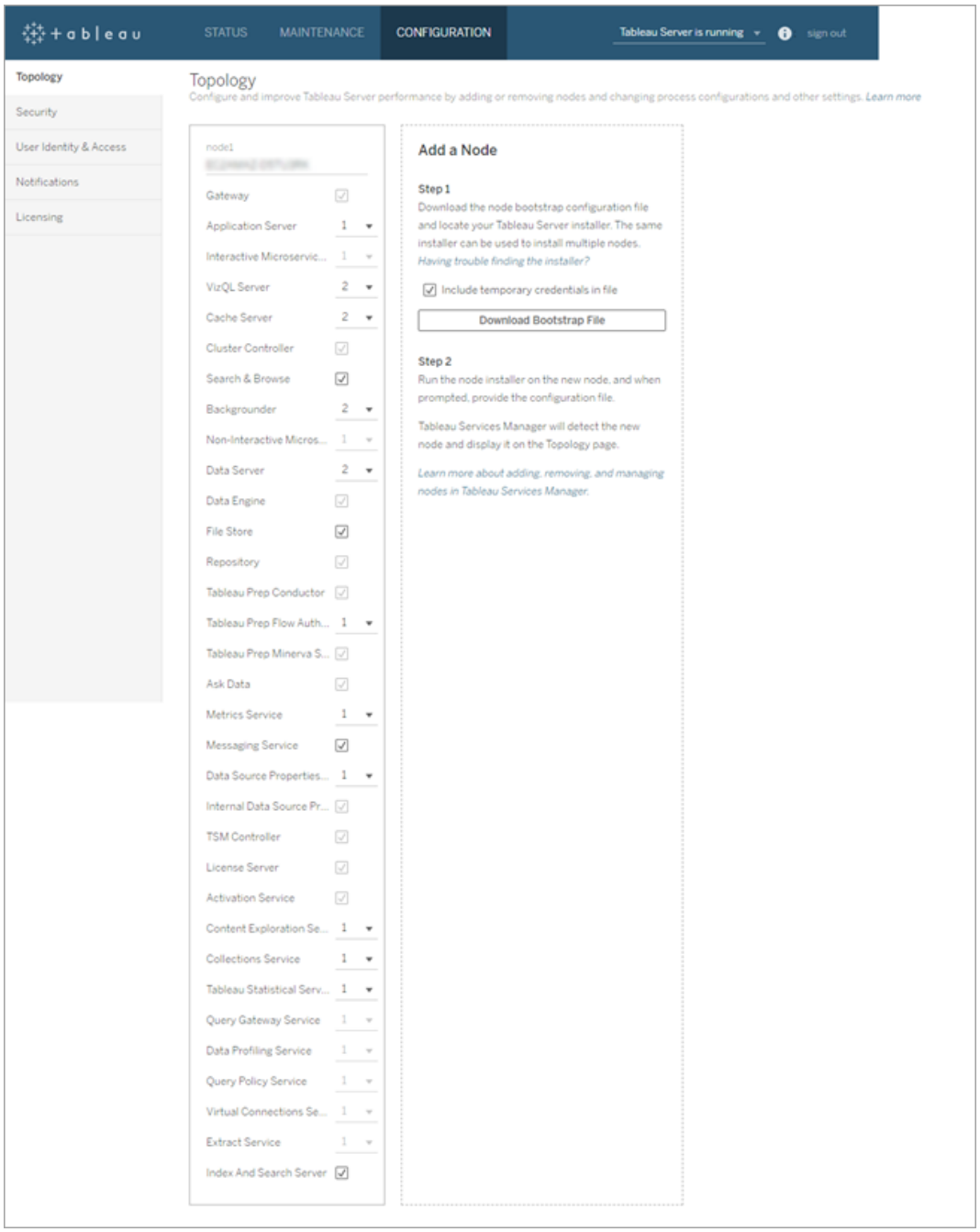
ノード bootstrap ファイルの生成

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックし、[ノードの追加] ボックスで **[Bootstrap ファイルをダウンロード]** をクリックします。



bootstrap ファイルが作成され、ローカル コンピューターにコピーされます。

既定では埋め込みの認証資格情報は bootstrap ファイルに含まれています。認証資格情報を bootstrap ファイルに埋め込まない場合は、**[ファイルに一時的な認証資格情報を含**

める] オプションをオフにします。埋め込み認証資格情報をノードの bootstrap ファイルに含める機能を完全に無効にする場合は、サーバーの構成オプションを設定できます。詳細については、features.PasswordlessBootstrapInit を参照してください。

追加ノードのインストールと初期化

開始前に、ノード bootstrap ファイルが最新であることを確認してください。たとえば、bootstrap ファイルの生成後に `tsm security regenerate-internal-tokens` を実行した場合は初期化に失敗します。

1. 1 番目のコンピューターで使用した元のインストーラーと、生成した bootstrap ファイルをコピーして、Tableau Server を追加する新しいコンピューターからアクセスできる場所に貼り付けます。この場所とは、据え付けられたネットワーク共有ディスクや、直接新しいコンピューター上である可能性があります。

2. ローカル ファイアウォールを実行している場合は、クラスタに含まれるすべてのノードに対してファイアウォールのルールを構成する必要があります。詳細については、ローカル ファイアウォールの構成を参照してください。

3. 新しいノードで、Tableau Server セットアップ プログラムを実行します。

   パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。

     - **既定の場所**: 既定の場所 (`/opt/tableau/tableau_server`) にインストールするには次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

       ```
       sudo yum update
       ```

```
sudo yum install tableau-server-<version>.x86_64.rpm
```

- **既定以外の場所**: 既定以外の場所にインストールするには、`rpm -i` を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

  次のコマンドを実行します。

  ```
  sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
  ```

> **注:** Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で `yum` の使用が禁止されているため、`rpm -i` を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

  ```
  sudo apt-get update
  sudo apt-get upgrade
  sudo apt-get -y install gdebi-core
  sudo gdebi -n tableau-server-<version>_amd64.deb
  ```

4. 新しいノードと最初のノードの間の通信を初期化するには、Tableau Server のインストール時にインストールした initialize-tsm スクリプトを実行します。

   `scripts` ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

5. initialize-tsm スクリプトを実行します。

```
sudo ./initialize-tsm -b /path/to/<bootstrap>.json --accepteula
```

- -b フラグを使用して、コンピューターにコピーした bootstrap ファイルへのパスを指定します。bootstrap ファイルを暗号化した場合、インポートおよびエクスポート操作におけるシークレットの保護の説明に従って、ファイルをパイプする必要があります。

- 埋め込み認証資格情報を使用せずに bootstrap ファイルが生成された場合は、-u フラグを使用して最初のノードで管理ユーザーのユーザー名を指定します。これは Tableau Server 管理者ではなく、コンピュータの管理ユーザー名です。ユーザー パスワードを入力するように求めるプロンプトが表示されます。詳細については、tsm topology nodes get-bootstrap-file を参照してください。

  ```
  sudo ./initialize-tsm -b /path/to/<bootstrap>.json -u
  <admin-user-on-first-node> --accepteula
  ```

  > **重要:** 最初のノードでインストール処理を行った際に使用したものと同じユーザーの認証資格情報を入力する必要があります。

- --accepteula フラグによって、Tableau Server End User License Agreement (EULA) に同意します。EULA は次の場所で入手できます: エンド ユーザー ライセンス契約。

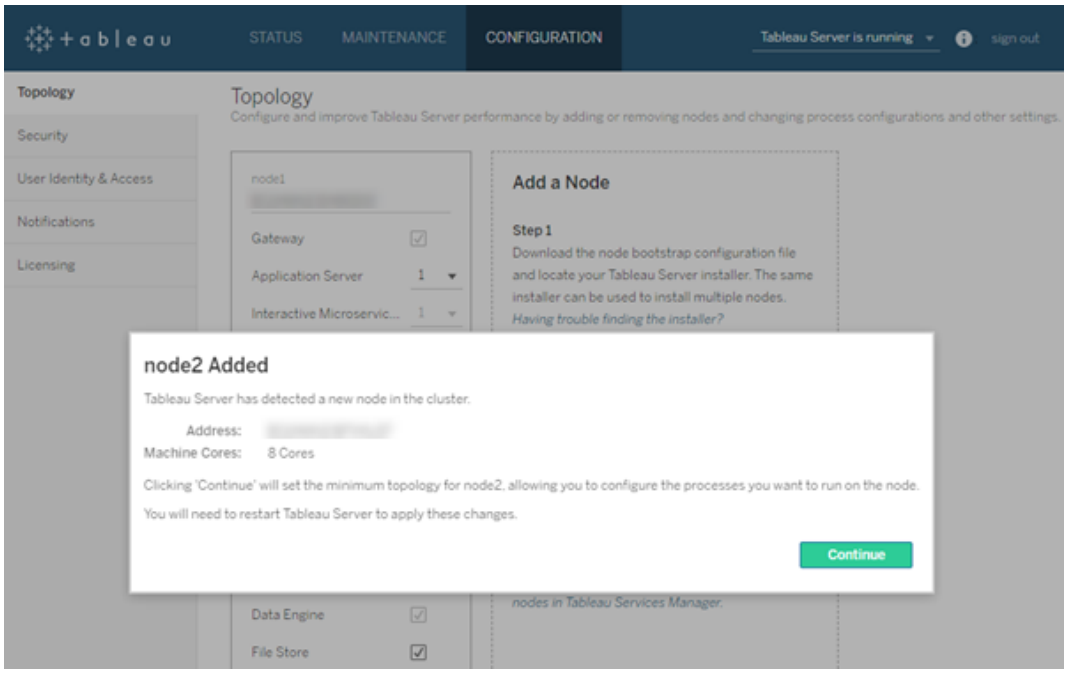
クラスタ コントローラーを使用した追加ノードの構成

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックします。新しいノードが追加されたことがメッセージで知らされます。



[続行] をクリックしてメッセージを閉じます。

3. クラスター コントローラーは最小トポロジの一部であり、既に選択されています。各ノードはクラスター コントローラーを実行する必要があります。

   ノードの最小トポロジを設定する必要しかない場合は、以下のステップ 4 に進んでください。これを行う可能性があるのは、追加ノードを追加しようとしており、このノードを最小以上に構成する準備が整っていない場合です。

   追加プロセスを最小トポロジに追加する必要がある場合、ノードで実行する必要があるプロセスを指定してください。これを行うには、必要なプロセスを選択するか、ノードで実行する必要があるプロセスの数を指定します。

   たとえば、ゲートウェイ 1 つとバックグラウンダーのインスタンス 2 つをノードに追加するには、以下を実行します。

   a. **[ゲートウェイ]** を選択します。

   b. **[バックグラウンダー]** のカウントを 2 に設定します。

バックグラウンダーをノードに追加すると、まだノードにない場合、データエンジンのインスタンスも追加します。

設定する特定のプロセスおよびプロセス カウントは、組織の環境やニーズによって異なります。一部のプロセスは、別のプロセスを追加すると自動的に追加されます。詳細については、ノードの構成とTableau Serverプロセスを参照してください。

> **注:** TSM の Web UI はインスタンス数の選択を許可するプロセスを最大 8 インスタンスに制限します。これより多くのインスタンスを構成するには、コマンド ラインと`tsm topology set-process` コマンドを使用してください。詳細については、tsm topology set-processを参照してください。

4. ページ上部の **[変更を保留中]** をクリックします。

3 ノード以上のクラスタを構成している場合、調整サービス アンサンブルの警告が表示されます。続行できます。別のステップで、調整サービス アンサンブルを展開します。

5. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

Tableau Server の再起動後、ノードは必要な最小トポロジおよび構成した追加プロセスに含まれています。

TSM CLI の使用

ノードbootstrap ファイルの生成

追加のコンピューターに Tableau Server をインストールするには、最初のノードで使用したものと同じインストーラーと共に、最初のノードで生成した "bootstrap" ファイルを使用します。

重要: bootstrap ファイルには、構成シークレットの暗号化に使用されるマスター キーストア ファイルのコピーが含まれています。また、このファイルには事前定義された時間で有効となる埋め込み認

証資格情報も含まれており(tabadmincontroller.auth.expiration.minutes を参照)、セッション Cookie として機能します。booststrap ファイルを保護するために追加措置を講じること強くお勧めします。

次のコマンドセットは、booststrap ファイルの出力を暗号化する方法の例です。この方法は、インポートおよびエクスポート操作におけるシークレットの保護で詳しく説明した暗号化プロセスに似ています。

ただし、この方法では、次のような末尾演算子 `&& \` を使用して別々の引数として渡す必要があります。

```
mkfifo -m 600 /tmp/secure1 && \

tsm topology nodes get-bootstrap-file --file /tmp/secure1 && \

gpg --symmetric --batch --yes --passphrase-file
~/.secrets/pgppassphrase.txt --cipher-algo AES256 --output
encrypted.enc < /tmp/secure1 && \

rm /tmp/secure1
```

1. Tableau Server を最初のノードにインストールしたら、ノード bootstrap ファイルを生成します。
2. 最初のノードで、ターミナル セッションを開きます。
3. 次のコマンドを入力して、bootstrap ファイルを生成します。

   ```
   tsm topology nodes get-bootstrap-file --file <path\file>.json
   ```

   既定では埋め込みの認証資格情報は bootstrap ファイルに含まれています。bootstrap ファイルを認証資格情報に含めない場合は、次のように `--no-embedded-credentials` または `-nec` オプションを使用します。

   ```
   tsm topology nodes get-bootstrap-file --file <path\file>.json --no-embedded-credentials。
   ```

埋め込み認証資格情報をノードの bootstrap ファイルに含める機能を完全に無効にする場合は、サーバーの構成オプションを設定できます。詳細については、features.PasswordlessBootstrapInitを参照してください。

追加ノードのインストールと初期化

1. 1 番目のコンピューターで使用した元のインストーラーと、生成した bootstrap ファイルをコピーして、Tableau Server を追加する新しいコンピューターからアクセスできる場所に貼り付けます。この場所とは、据え付けられたネットワーク共有ディスクや、直接新しいコンピューター上である可能性があります。

2. ローカル ファイアウォールを実行している場合は、クラスタに含まれるすべてのノードに対してファイアウォールのルールを構成する必要があります。詳細については、ローカル ファイアウォールの構成を参照してください。

3. 新しいノードで、Tableau Server セットアップ プログラムを実行します。

   パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。

     - **既定の場所**: 既定の場所 (`/opt/tableau/tableau_server`) にインストールするには次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

       ```
       sudo yum update

       sudo yum install tableau-server-<version>.x86_64.rpm
       ```

- **既定以外の場所**: 既定以外の場所にインストールするには、`rpm -i` を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

  次のコマンドを実行します。

  ```
  sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
  ```

  注: Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で `yum` の使用が禁止されているため、`rpm -i` を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

  ```
  sudo apt-get update
  sudo apt-get upgrade
  sudo apt-get -y install gdebi-core
  sudo gdebi -n tableau-server-<version>_amd64.deb
  ```

4. 新しいノードと最初のノードの間の通信を初期化するには、Tableau Server のインストール時にインストールした initialize-tsm スクリプトを実行します。

   新しいノードで、

   `scripts` ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

5. initialize-tsm スクリプトを実行します。

```
sudo ./initialize-tsm -b /path/to/<bootstrap>.json --accepteula
```

- -b フラグを使用して、コンピューターにコピーした bootstrap ファイルへのパスを指定します。bootstrap ファイルを暗号化した場合、インポートおよびエクスポート操作におけるシークレットの保護の説明に従って、ファイルをパイプする必要があります。

- 埋め込み認証資格情報を使用せずに bootstrap ファイルが生成された場合は、-u フラグを使用して最初のノードで管理ユーザーのユーザー名を指定します。これは Tableau Server 管理者ではなく、コンピュータの管理ユーザー名です。ユーザー パスワードを入力するように求めるプロンプトが表示されます。詳細については、tsm topology nodes get-bootstrap-file を参照してください。

  ```
  sudo ./initialize-tsm -b /path/to/<bootstrap>.json -u
  <admin-user-on-first-node> --accepteula
  ```

  重要: 最初のノードでインストール処理を行った際に使用したものと同じユーザーの認証資格情報を入力する必要があります。

- --accepteula フラグによって、Tableau Server End User License Agreement (EULA) に同意します。EULA は次の場所にあります。

  ```
  /opt/tableau/tableau_server/packages/docs.<version_
  code>/EULA.rtf
  ```

追加のノードを構成する

注: この基本的な例は、ノード上でのトポロジの設定方法を示しています。稼働中のマルチノードの詳細な例については、例: 3 ノード HA クラスタのインストールおよび構成を参照してください。

最初 (元) のノードで、新しく追加するノードのトポロジを設定します。トポロジでは、ノードで実行するプロセスおよび各プロセスで実行するインスタンスの数を指定します。ノードのトポロジは、お使いの環境と組織のニーズによって異なります。次に示すのは、トポロジの設定の一例です。

1. 最初 (元) のノードで、ターミナル セッションを開きます。

2. 次の新しいノードの node-id を取得します。

   ```
   tsm topology list-nodes -v
   ```

   -v オプションは、ノードと現在実行中のプロセスをリストします。新しく追加されたノードにはプロセスがないので、それにより判別できます。

3. このノードで実行する必要があるプロセスを個別に指定します。

   これを次のコマンドで実行します。

   ```
   tsm topology set-process -n <nodeID> -pr <processname> -c <n>
   ```

   各ノードにクラスタ コントローラーのインスタンスを追加しなくてはなりません。

   例えば、クラスタ コントローラー、バックグラウンダーの 2 つのインスタンス、node2 へのゲートウェイを追加するには次のコマンドを実行します。

   ```
   tsm topology set-process -n node2 -pr clustercontroller -c 1
   tsm topology set-process -n node2 -pr backgrounder -c 2
   tsm topology set-process -n node2 -pr gateway -c 1
   ```

   設定する特定のプロセスおよびプロセス カウントは、組織の環境やニーズによって異なります。一部のプロセスは、別のプロセスを追加すると自動的に追加されます。詳細については、ノードの構成とTableau Server プロセスを参照してください。

4. ノード構成を適用します。保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

   ```
   tsm pending-changes apply
   ```

3 または 5 ノードクラスタを展開する場合には、調整サービス アンサンブルを展開しているという警告が表示されます。この警告しか表示されていないのであれば、警告に関わらず、--ignore-warnings オプションを使用して安全に上書きし、構成を変更できます。

```
tsm pending-changes apply --ignore-warnings
```

ドライバーのインストール

Tableau Serverがデータに接続して、抽出を実行できるようにドライバーをインストールする必要があります。次のプロセスのいずれかを実行しているすべてのノードにこれらのドライバーをインストールします。

- VizQL Server (vizqlserver)
- アプリケーション サーバー (vizportal)
- データ サーバー (dataserver)
- バックグラウンダー (backgrounder)

ドライバーと管理ビュー

Tableau Serverで内蔵の管理ビューを使用したい場合は、上記のプロセスのいずれかを実行しているすべてのノードに PostgreSQL ドライバーもインストールする必要があります。

詳細については、データベース ドライバーを参照してください。

データベース ドライバー

Tableau コネクタにはデータベースとやり取りするためのドライバーが必要です。Tableau Server からデータ ソースに接続できるようになる前に、接続するデータ ソース用のドライバーをインストールする必要があります。Tableau Server on Linux がサポートするデータ ソースの詳細については、Tableau Server の技術仕様ページで確認できます。サポート対象であるすべてのコネクタのドライバーのリンクおよびインストール手順については、ドライバーのダウンロード ページで確認できます。

重要: 組み込みの管理者ビューを使用する場合は、PostgreSQL ドライバーをインストールする必要があります。このドライバーは、ドライバーのダウンロード ページで見つけることができます。

クラスタでのドライバーのインストール

データソースのドライバーは Tableau Server クラスタの最初のノードにインストールする必要があります。複数のノードに Tableau Server をインストールする場合は、次のいずれかのプロセスを実行するすべてのノードにドライバーをインストールする必要もあります。

- アプリケーション サーバー (Vizportal)
- バックグラウンダー
- データサーバー
- VizQL Server

例: 3 ノードHA クラスタのインストールおよび構成

Tableau Server の高可用性マルチノードインストールは、Tableau Server の効率性および可用性を最大化する上で有効です。マルチノードの Tableau Server 展開を構成する際は、実行するすべての手順は、冗長性を組み込むことで起こりうるサーバーシステムの停止時間の低減を助けるように設計されています。サーバープロセスを追加のノードに移動または追加することで効率性を向上するのに加え、次の要件を満たすことで Tableau Server の高可用性 (HA) インストールを実現できます。

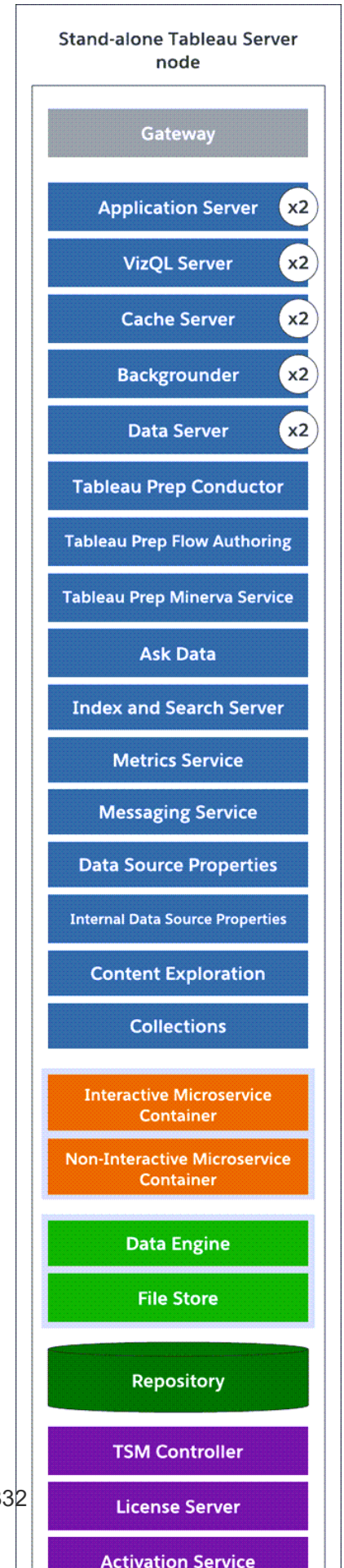
- 少なくとも合計 3 台のノードを追加する
- 調整サービスのアンサンブルを少なくとも 3 台のノードで展開する
- クライアント ファイル サービス (CFS) のインスタンスを少なくとも 1 つのノードに追加する (調整サービスを実行している各ノードにインスタンスを追加することをお勧めします)
- 追加ノードのうちの 1 台にファイル ストアの第二インスタンスを追加 (既にノード上にない場合はデータエンジンが自動的にインストールされます)
- 追加ノードのうちの 1 台にリポジトリの第 2 インスタンス (pgsql) を追加

このような追加を含む Tableau Server のインストールには、冗長性が組み込まれ、リポジトリに問題が発生した場合のフェールオーバーをサポートします。この例は、このやり方などを示します。

単一サーバー システム

最初のノードに Tableau Server をインストールすると、機能する必要があるすべてを実行するシステムがあります。そのシステムには、全てのサーバープロセスの少なくとも一つのインスタンスがあり、

Tableau Serverの最も基礎的な構成です。冗長性はありません。サーバートポロジは次のようになります (一部のTSM固有プロセスは表示されません)。


Stand-alone Tableau Server node
Gateway
Application Server
x2
VizQL Server
x2
Cache Server
x2
Backgrounder
x2
Data Server
x2
Tableau Prep Conductor
Tableau Prep Flow Authoring
Tableau Prep Minerva Service
Ask Data
Index and Search Server
Metrics Service
Messaging Service
Data Source Properties
Internal Data Source Properties
Content Exploration
Collections
Interactive Microservice Container
Non-Interactive Microservice Container
Data Engine
File Store
Repository
TSM Controller
License Server
Activation Service

3 ノードシステム

冗長性を組み込むには、リポジトリやデータエンジンやファイル ストア プロセスのホストインスタンスに追加 ノードを追加する必要があります。ノード上のプロセスの複数インスタンスを含め、その他のプロセスのインスタンスを追加することができます。さらに、システムの脆弱性を緩和するため、複数ゲートウェイおよび一部のサーバープロセスの追加インスタンスを実行することができます。この構成を達成するには最低 3 台のコンピューターが必要です。

下の図では、ファイル ストア プロセスが両方の追加 ノードに追加されています。2 番目のリポジトリのパッシブ インスタンスも他の新規 ノードの 1 つに追加されています。最後に、サーバープロセス (青色で表示) が、冗長性を提供するために追加 ノードに追加されました。

構成の手順

この手順は、上の図のように、3 ノードの HA Tableau Server クラスタを、追加 ノード上の 2 つのリポジトリインスタンスおよび 2 つのファイル ストア/データエンジン インスタンスと構成する方法について説明しています。

はじめる前に

追加ノードに Tableau Server をインストールする前に、各追加ノードが分散環境の要件を満たしていることを確認してください。詳細については、分散環境の要件 を参照してください。

TSM Web インターフェイスの使用

注: この操作には、TSM コマンドラインを使用して実行する必要のある手順が含まれています。

ステップ 1: 最初のノードをインストールする

Tableau Server のインストールと構成を参照してください。

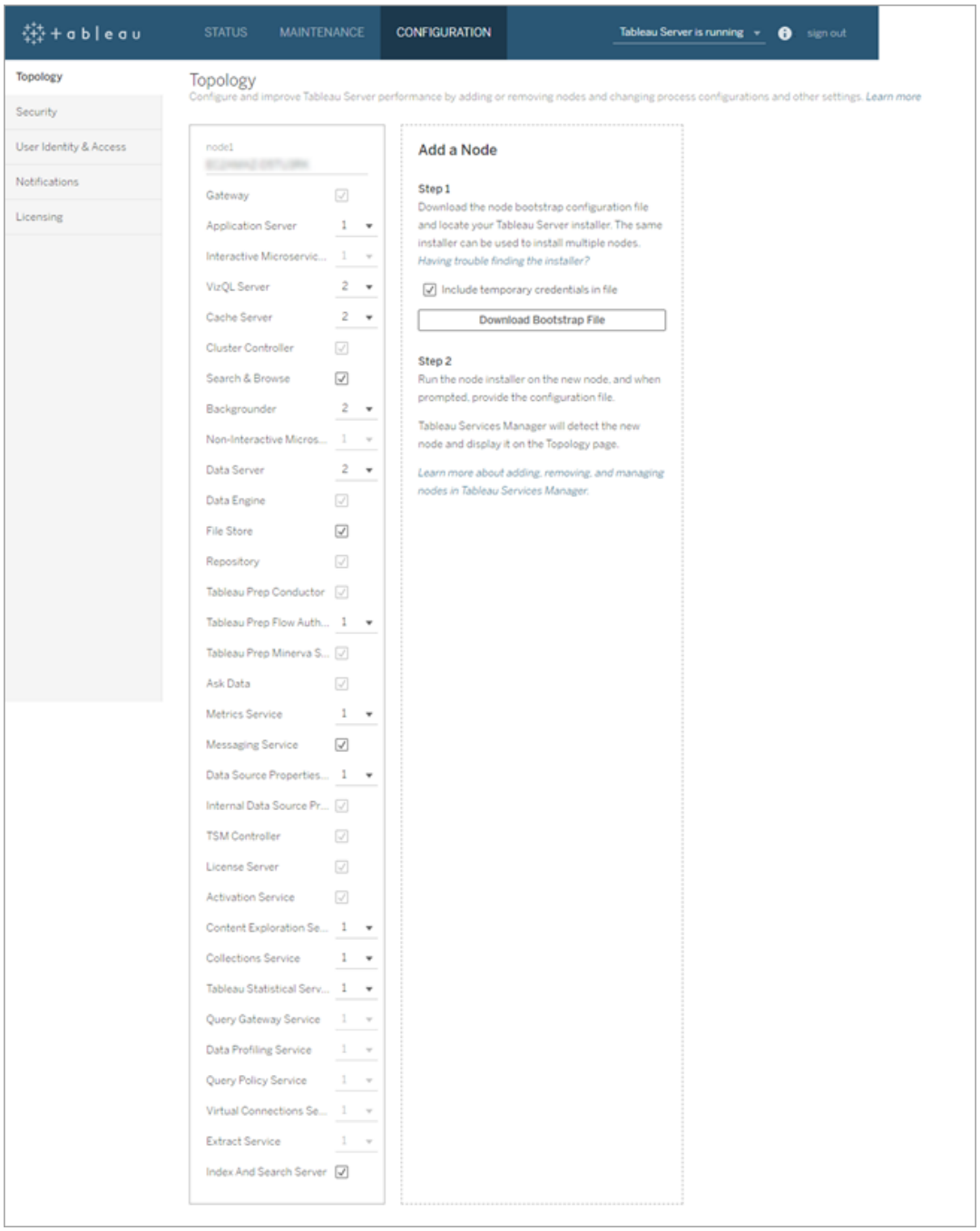
ステップ 2: 追加ノード用の bootstrap ファイルを生成する

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックし、[ノードの追加] ボックスで **[Bootstrap ファイルをダウンロード]** をクリックします。



bootstrap ファイルが作成され、ローカル コンピューターにコピーされます。

既定では埋め込みの認証資格情報は bootstrap ファイルに含まれています。認証資格情報を bootstrap ファイルに埋め込まない場合は、**[ファイルに一時的な認証資格情報を含め**

る] オプションをオフにします。埋め込み認証資格情報をノードの bootstrap ファイルに含める機能を完全に無効にする場合は、サーバーの構成オプションを設定できます。詳細については、features.PasswordlessBootstrapInit を参照してください。

ステップ 3: ノード 2 をインストールして初期化する

1. 1 番目のコンピューターで使用した元のインストーラーと、生成した bootstrap ファイルをコピーして、Tableau Server を追加する新しいコンピューターからアクセスできる場所に貼り付けます。この場所とは、据え付けられたネットワーク共有ディスクや、直接新しいコンピューター上である可能性があります。

2. ローカル ファイアウォールを実行している場合は、クラスタに含まれるすべてのノードに対してファイアウォールのルールを構成する必要があります。詳細については、ローカル ファイアウォールの構成を参照してください。

3. 新しいノードで、Tableau Server セットアップ プログラムを実行します。

   パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。

     - **既定の場所**: 既定の場所 (`/opt/tableau/tableau_server`) にインストールするには次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

       ```
       sudo yum update

       sudo yum install tableau-server-<version>.x86_64.rpm
       ```

- **既定以外の場所**: 既定以外の場所にインストールするには、`rpm -i` を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

  次のコマンドを実行します。

  ```
  sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
  ```

  注: Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で `yum` の使用が禁止されているため、`rpm -i` を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

  ```
  sudo apt-get update
  sudo apt-get upgrade
  sudo apt-get -y install gdebi-core
  sudo gdebi -n tableau-server-<version>_amd64.deb
  ```

4. 新しいノードと最初のノードの間の通信を初期化するには、Tableau Server のインストール時にインストールした initialize-tsm スクリプトを実行します。

   `scripts` ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

5. initialize-tsm スクリプトを実行します。

   ```
   sudo ./initialize-tsm -b /path/to/<bootstrap>.json --accepteula
   ```

- -b フラグを使用して、コンピューターにコピーした bootstrap ファイルへのパスを指定します。bootstrap ファイルを暗号化した場合、インポートおよびエクスポート操作におけるシークレットの保護の説明に従って、ファイルをパイプする必要があります。

- 埋め込み認証資格情報を使用せずに bootstrap ファイルが生成された場合は、-u フラグを使用して最初のノードで管理ユーザーのユーザー名を指定します。これは Tableau Server 管理者ではなく、コンピュータの管理ユーザー名です。ユーザー パスワードを入力するように求めるプロンプトが表示されます。詳細については、tsm topology nodes get-bootstrap-file を参照してください。

  ```
  sudo ./initialize-tsm -b /path/to/<bootstrap>.json -u
  <admin-user-on-first-node> --accepteula
  ```

  > **重要:** 最初のノードでインストール処理を行った際に使用したものと同じユーザーの認証資格情報を入力する必要があります。

- --accepteula フラグによって、Tableau Server End User License Agreement (EULA) に同意します。EULA は次の場所で入手できます: エンド ユーザー ライセンス契約。

ステップ 4: ノード 3 をインストールして初期化する

上記のステップ 3 を繰り返します。

ステップ 5: 最小トポロジを使用して新しいノードを構成する

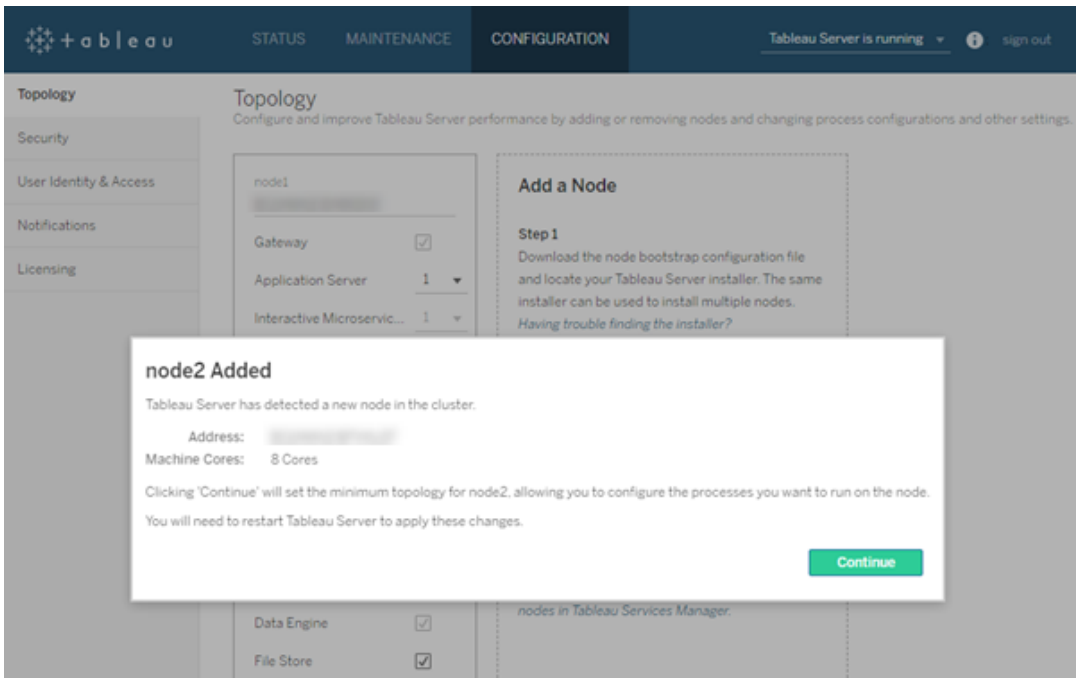
クラスタへの新規ノードの追加プロセスを完了するには、最小限のトポロジでノードを構成する必要があります。最小限のトポロジの場合、保留中の変更はクラスタ コントローラーの追加だけであり、これを各ノードで行う必要があります。ノードに他のプロセスが必要な場合は、そのほとんどを同時に追加できます。この例では、クラスタ コントローラーでノードの構成のみを行っています。

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックします。新しいノードが追加されたことがメッセージで知らされます。



[続行] をクリックしてメッセージを閉じます。

3. ページ上部の **[変更を保留中]** をクリックします。

3 ノード以上のクラスタを構成している場合、調整サービス アンサンブルの警告が表示されます。続行できます。別のステップで、調整サービス アンサンブルを展開します。

4. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

Tableau Server が再起動すると、ノードは必要な最小限のトポロジで組み込まれます。

ステップ 6: 調整サービス アンサンブルを展開する

合計 3 台以上のノードをインストールした場合、調整サービス アンサンブルも展開してください。これを展開しないと、サーバーの構成やトポロジに変更を加えるたびに警告メッセージが表示されるようになります。このメッセージは無視できますが、ベストプラクティスはマルチノードの調整サービス アンサンブルを展開することです。

Tableau Server をインストールすると、調整サービスのインスタンスが最初のノードに 1 つインストールされます。TSM とTableau Server は調整サービスが正しく機能することに依存しているため、冗長性を提供し、マルチノード インストールでの可用性を確保するため、調整サービス アンサンブルを展開することで調整サービスの追加のインスタンスを構成します。調整サービス アンサンブルは 1、3、または 5 個の調整サービス インスタンスでインストールされます。Tableau Server の 3 ノード インストールでは、推奨される調整サービスのインスタンス数は 3、つまり各ノードに 1 つです。

保留中の他の変更がある場合は、調整サービス アンサンブルの展開を試行しないでください。新しい調整サービス アンサンブルを展開する前に、保留中の変更を破棄または適用します。

注: この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

以下の手順は、既存の 3 ノードTableau Server クラスタ上に新しい調整サービス アンサンブルを展開し、旧アンサンブルをクリーンアップする方法を説明しています。

1. 最初のノードで、ターミナル セッションを開きます。
2. Tableau Server を停止します。

   ```
   tsm stop
   ```

   メッセージが表示されたら、TSM 管理者としてサインインします。

   管理コントローラーや管理エージェントなど、継続して実行されるTSM プロセスもあります。
3. 保留中の変更がないことを確認します。

```
tsm pending-changes list
```

保留中の変更がある場合は、変更を破棄するか、変更を適用する必要があります。保留中の変更の適用には少し時間がかかります。

- 変更を破棄します。

  ```
  tsm pending-changes discard
  ```

  または

- 変更を適用します。

  ```
  tsm pending-changes apply
  ```

コマンドが完了し、システム プロンプトに戻るまで待ちます。

4. クラスタ内の各 *ノード*の*ノード*ID を取得します。

   ```
   tsm topology list-nodes -v
   ```

5. `tsm topology deploy-coordination-service` コマンドを使用して調整サービスを指定された*ノード*に追加することで、新しい調整サービス アンサンブルを追加します。各*ノード*を識別するために実際の*ノード*ID を使用して、調整サービスを追加する*ノード*を指定する必要があります。展開が失敗しなければ、このコマンドにより、新しいアンサンブルが "本番環境" のアンサンブル (使用中のアンサンブル) になり、古いアンサンブルが削除されます。このような場合は、以下のステップ 6 を参照してください。

   注: サーバーの再起動が行われることを確認する「y/n (はい / いいえ)」のプロンプトが表示されます。入力せずにコマンドを実行するには、`--ignore-prompt` オプションを含めます。

   たとえば、調整サービスを 3 *ノード* クラスタの 3 *ノード* (node1、node2、node3) すべてに展開します。

   ```
   tsm topology deploy-coordination-service -n node1,node2,node3
   ```

   コマンドが完了し、システム プロンプトに戻るまで待ちます。

6. (オプション) 展開が失敗した場合は、`tsm topology cleanup-coordination-service` コマンドを実行して、配置に失敗したアンサンブルを削除する必要があります。コマンドの実行の詳細については、tsm topology cleanup-coordination-service を参照してください。

7. Tableau Server を起動します。

   ```
   tsm start
   ```

新しい調整サービス アンサンブルを展開する方法の詳細については、調整サービス アンサンブルの展開を参照してください。

ステップ 7: 追加のノードでクライアント ファイル ・サービス (CFS) を構成する

CFS を追加のノードに追加します。調整サービスを実行しているすべてのノードに CFS を追加することをお勧めします。

注: この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

1. 最初のノードで、ターミナル セッションを開きます。

2. CFS を追加しようとしているノードのノード ID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. ノード、プロセス、および単一のインスタンスを指定して、ノードに CFS を追加します。

   たとえば、次のコマンドは、CFS の 1 つのインスタンスを node2 に追加します。

   ```
   tsm topology set-process -n node2 -pr clientfileservice -c 1
   ```

   CFS を使用して既に構成されているノードに CFS のインスタンスを追加しようとすると、エラー メッセージが表示され、ノードに既にインスタンスが存在していることが通知されます。

CFS をさらに他のノードに追加するには、ノードごとにこの手順を繰り返します。

4. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

ステップ 8: ノード 2 のプロセスを構成する

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックします。

3. 1 台目の追加ノードで実行されるべきプロセスとインスタンスの数を指定します。

   この例では、次のようになります。

   a. **[ゲートウェイ]** を選択します。

   b. **アプリケーション サーバー** (vizportal) カウントを 2 に設定。

   c. **VizQL Server** カウントを 2 に設定。

   d. **キャッシュサーバー** カウントを 2 に設定。

   e. **[バックグラウンダー]** のカウントを 2 に設定します。

      バックグラウンダーをノードに追加すると、まだノードにない場合、データエンジンのインスタンスも追加します。

   f. **Data Server** カウントを 2 に設定。

   g. **ファイル ストア** を選択します。

   h. **リポジトリ** (pgsql) を選択します。

i. **[メトリクス サービス]** を選択します。(従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。)

j. **[メッセージング サービス]** を選択します。

k. **[データ ソース プロパティ]** を選択します。

l. **[コンテンツ探索 サービス]** を選択します。

m. **[コレクション サービス]** を選択します。

n. **[Index and Search Server]** を選択します。

設定する特定のプロセスおよびプロセス カウントは、組織の環境やニーズによって異なります。一部のプロセスは、別のプロセスを追加すると自動的に追加されます。詳細については、ノードの構成とTableau Server プロセスを参照してください。

> **注:** TSM の Web UI はインスタンス数の選択を許可するプロセスを最大 8 インスタンスに制限します。これより多くのインスタンスを構成するには、コマンド ラインと `TSM topology set-process` コマンドを使用してください。詳細については、tsm topology set-processを参照してください。

ステップ 9: ノード 3 のプロセスを構成する

1. TSM の、**構成** タブで、2 台目の追加ノードで実行されるべきプロセスとインスタンスの数を指定します。

   この例では、次のようになります。

   a. **[ゲートウェイ]** を選択します。

   b. **アプリケーション サーバー** (vizportal) カウントを 2 に設定。

   c. **VizQL Server** カウントを 2 に設定。

d. **キャッシュサーバー**カウントを 2 に設定。

e. **[バックグラウンダー]** のカウントを 2 に設定します。

バックグラウンダーをノードに追加すると、まだノードにない場合、データエンジンのインスタンスも追加します。

f. **Data Server** カウントを 2 に設定。

g. **ファイル ストア**を選択します。

h. **[メトリクス サービス]** を選択します。(従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。)

i. **[データ ソース プロパティ]** を選択します。

j. **[コンテンツ探索 サービス]** を選択します。

k. **[コレクション サービス]** を選択します。

l. **[Index and Search Server]** を選択します。

2. ページ上部の **[変更を保留中]** をクリックします。

[変更を保留中] のリストが表示されます。

3. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

TSM CLI の使用

ステップ 1: 最初のノードをインストールする

Tableau Server のインストールと構成を参照してください。

ステップ 2: 追加ノード用の bootstrap ファイルを生成する

追加のコンピューターに Tableau Server をインストールするには、最初のノードで使用したものと同じインストーラーと共に、最初のノードで生成した "bootstrap" ファイルを使用します。

重要: bootstrap ファイルには、構成シークレットの暗号化に使用されるマスター キーストア ファイルのコピーが含まれています。また、このファイルには事前定義された時間で有効となる埋め込み認証資格情報も含まれており(tabadmincontroller.auth.expiration.minutes を参照)、セッション Cookie として機能します。booststrap ファイルを保護するために追加措置を講じること強くお勧めします。

次のコマンド セットは、booststrap ファイルの出力を暗号化する方法の例です。この方法は、インポートおよびエクスポート操作におけるシークレットの保護で詳しく説明した暗号化プロセスに似ています。

ただし、この方法では、次のような末尾演算子 `&& \` を使用して別々の引数として渡す必要があります。

```
mkfifo -m 600 /tmp/secure1 && \

tsm topology nodes get-bootstrap-file --file /tmp/secure1 && \

gpg --symmetric --batch --yes --passphrase-file
~/.secrets/pgppassphrase.txt --cipher-algo AES256 --output
encrypted.enc < /tmp/secure1 && \

rm /tmp/secure1
```

1. Tableau Server を最初のノードにインストールしたら、ノード bootstrap ファイルを生成します。
2. 最初のノードで、ターミナル セッションを開きます。
3. 次のコマンドを入力して、bootstrap ファイルを生成します。

   ```
   tsm topology nodes get-bootstrap-file --file <path\file>.json
   ```

既定では埋め込みの認証資格情報は bootstrap ファイルに含まれています。bootstrap ファイルを認証資格情報に含めない場合は、次のように `--no-embedded-credentials` または `-nec` オプションを使用します。

`tsm topology nodes get-bootstrap-file --file <path\file>.json --no-embedded-credentials`。

埋め込み認証資格情報をノードの bootstrap ファイルに含める機能を完全に無効にする場合は、サーバーの構成オプションを設定できます。詳細については、features.PasswordlessBootstrapInit を参照してください。

ステップ 3: ノード 2 をインストールして初期化する

1. 1 番目のコンピューターで使用した元のインストーラーと、生成した bootstrap ファイルをコピーして、Tableau Server を追加する新しいコンピューターからアクセスできる場所に貼り付けます。この場所とは、据え付けられたネットワーク共有ディスクや、直接新しいコンピューター上である可能性があります。

2. ローカル ファイアウォールを実行している場合は、クラスタに含まれるすべてのノードに対してファイアウォールのルールを構成する必要があります。詳細については、ローカル ファイアウォールの構成を参照してください。

3. 新しいノードで、Tableau Server セットアップ プログラムを実行します。

   パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。

- **既定の場所**: 既定の場所 (/opt/tableau/tableau_server) にインストールするには次のコマンドを実行します (<version> の書式設定は「メジャー-マイナー-メンテナンス」(例: 2019-2-5))。

  ```
  sudo yum update

  sudo yum install tableau-server-<version>.x86_64.rpm
  ```

- **既定以外の場所**: 既定以外の場所にインストールするには、rpm -i を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

  次のコマンドを実行します。

  ```
  sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
  ```

> **注:** Tableau Server のインストールに yum を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で yum の使用が禁止されているため、rpm -i を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (<version> の書式設定は「メジャー-マイナー-メンテナンス」(例: 2019-2-5))。

  ```
  sudo apt-get update
  sudo apt-get upgrade
  sudo apt-get -y install gdebi-core
  sudo gdebi -n tableau-server-<version>_amd64.deb
  ```

4. 新しいノードと最初のノードの間の通信を初期化するには、Tableau Server のインストール時にインストールした initialize-tsm スクリプトを実行します。

   新しいノードで、

   scripts ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

5. initialize-tsm スクリプトを実行します。

   ```
   sudo ./initialize-tsm -b /path/to/<bootstrap>.json --accepteula
   ```

   - -b フラグを使用して、コンピューターにコピーした bootstrap ファイルへのパスを指定します。bootstrap ファイルを暗号化した場合、インポートおよびエクスポート操作におけるシークレットの保護の説明に従って、ファイルをパイプする必要があります。

   - 埋め込み認証資格情報を使用せずに bootstrap ファイルが生成された場合は、-u フラグを使用して最初のノードで管理ユーザーのユーザー名を指定します。これは Tableau Server 管理者ではなく、コンピュータの管理ユーザー名です。ユーザー パスワードを入力するように求めるプロンプトが表示されます。詳細については、tsm topology nodes get-bootstrap-file を参照してください。

     ```
     sudo ./initialize-tsm -b /path/to/<bootstrap>.json -u
     <admin-user-on-first-node> --accepteula
     ```

     重要: 最初のノードでインストール処理を行った際に使用したものと同じユーザーの認証資格情報を入力する必要があります。

   - --accepteula フラグによって、Tableau Server End User License Agreement (EULA) に同意します。EULA は次の場所にあります。

     ```
     /opt/tableau/tableau_server/packages/docs.<version_
     code>/EULA.rtf
     ```

ステップ 4: ノード 3 をインストールして初期化する

Tableau Server をノード 3 にインストールする

1. 1 番目のコンピューターで使用した元のインストーラーと、生成した bootstrap ファイルをコピーして、Tableau Server を追加する新しいコンピューターからアクセスできる場所に貼り付けます。この場所とは、据え付けられたネットワーク共有ディスクや、直接新しいコンピューター上である可能性があります。

2. ローカル ファイアウォールを実行している場合は、クラスタに含まれるすべてのノードに対してファイアウォールのルールを構成する必要があります。詳細については、ローカル ファイアウォールの構成を参照してください。

3. 新しいノードで、Tableau Server セットアップ プログラムを実行します。

   パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。

     - **既定の場所**: 既定の場所 (`/opt/tableau/tableau_server`) にインストールするには次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

       ```
       sudo yum update

       sudo yum install tableau-server-<version>.x86_64.rpm
       ```

     - **既定以外の場所**: 既定以外の場所にインストールするには、`rpm -i` を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

       次のコマンドを実行します。

```
sudo rpm -i --prefix /preferred/install/path tableau-
server.rpm
```

> **注:** Tableau Server のインストールに yum を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で yum の使用が禁止されているため、rpm -i を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (<version> の書式設定は「メジャー-マイナー-メンテナンス」(例: 2019-2-5))。

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get -y install gdebi-core
sudo gdebi -n tableau-server-<version>_amd64.deb
```

4. 新しいノードと最初のノードの間の通信を初期化するには、Tableau Server のインストール時にインストールした initialize-tsm スクリプトを実行します。

   新しいノードで、

   scripts ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

5. initialize-tsm スクリプトを実行します。

   ```
   sudo ./initialize-tsm -b /path/to/<bootstrap>.json --accepteula
   ```

   - -b フラグを使用して、コンピューターにコピーした bootstrap ファイルへのパスを指定します。bootstrap ファイルを暗号化した場合、インポートおよびエクスポート操作におけるシークレットの保護の説明に従って、ファイルをパイプする必要があります。

- 埋め込み認証資格情報を使用せずに bootstrap ファイルが生成された場合は、-u フラグを使用して最初のノードで管理ユーザーのユーザー名を指定します。これは Tableau Server 管理者ではなく、コンピュータの管理ユーザー名です。ユーザー パスワードを入力するように求めるプロンプトが表示されます。詳細については、tsm topology nodes get-bootstrap-file を参照してください。

  ```
  sudo ./initialize-tsm -b /path/to/<bootstrap>.json -u
  <admin-user-on-first-node> --accepteula
  ```

  重要: 最初のノードでインストール処理を行った際に使用したものと同じユーザーの認証資格情報を入力する必要があります。

- --accepteula フラグによって、Tableau Server End User License Agreement (EULA) に同意します。EULA は次の場所にあります。

  ```
  /opt/tableau/tableau_server/packages/docs.<version_
  code>/EULA.rtf
  ```

ステップ 5: プロセスを追加ノードに追加する

1. 最初のノードで、次の各追加ノードのクラスタ コントローラー インスタンスを構成します。

   ```
   tsm topology set-process -n <nodeID_second-node> -pr
   clustercontroller -c 1
   ```

   ```
   tsm topology set-process -n <nodeID_third-node> -pr
   clustercontroller -c 1
   ```

2. ノード構成の変更を適用します。

   ```
   tsm pending-changes apply --ignore-warnings
   ```

   3 ノード クラスタを展開したため、調整サービス アンサンブルの展開に関する警告が表示されます。警告を無視し、--ignore-warnings オプションを使用して変更の構成を適用します。次のステップで、新しい調整サービス アンサンブルを展開します。

ステップ 6: 調整サービス アンサンブルを展開する

合計 3 台以上のノードをインストールした場合、調整サービス アンサンブルも展開してください。これを展開しないと、サーバーの構成やトポロジに変更を加えるたびに警告メッセージが表示されるようになります。このメッセージは無視できますが、ベストプラクティスはマルチノードの調整サービス アンサンブルを展開することです。

Tableau Server をインストールすると、調整サービスのインスタンスが最初のノードに 1 つインストールされます。TSM と Tableau Server は調整サービスが正しく機能することに依存しているため、冗長性を提供し、マルチノード インストールでの可用性を確保するため、調整サービス アンサンブルを展開することで調整サービスの追加のインスタンスを構成します。調整サービス アンサンブルは 1、3、または 5 個の調整サービス インスタンスでインストールされます。Tableau Server の 3 ノード インストールでは、推奨される調整サービスのインスタンス数は 3、つまり各ノードに 1 つです。

保留中の他の変更がある場合は、調整サービス アンサンブルの展開を試行しないでください。新しい調整サービス アンサンブルを展開する前に、保留中の変更を破棄または適用します。

> **注:** この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

以下の手順は、既存の 3 ノード Tableau Server クラスタ上に新しい調整サービス アンサンブルを展開し、旧アンサンブルをクリーンアップする方法を説明しています。

1. 最初のノードで、ターミナル セッションを開きます。
2. Tableau Server を停止します。

   ```
   tsm stop
   ```

   メッセージが表示されたら、TSM 管理者としてサインインします。

   管理コントローラーや管理エージェントなど、継続して実行される TSM プロセスもあります。
3. 保留中の変更がないことを確認します。

```
tsm pending-changes list
```

保留中の変更がある場合は、変更を破棄するか、変更を適用する必要があります。保留中の変更の適用には少し時間がかかります。

- 変更を破棄します。

  ```
  tsm pending-changes discard
  ```

  または

- 変更を適用します。

  ```
  tsm pending-changes apply
  ```

コマンドが完了し、システム プロンプトに戻るまで待ちます。

4. クラスタ内の各 *ノード*の*ノード*ID を取得します。

   ```
   tsm topology list-nodes -v
   ```

5. `tsm topology deploy-coordination-service` コマンドを使用して調整サービスを指定された*ノード*に追加することで、新しい調整サービス アンサンブルを追加します。各*ノード*を識別するために実際の*ノード*ID を使用して、調整サービスを追加する*ノード*を指定する必要があります。展開が失敗しなければ、このコマンドにより、新しいアンサンブルが "本番環境" のアンサンブル (使用中のアンサンブル) になり、古いアンサンブルが削除されます。このような場合は、以下のステップ 6 を参照してください。

   注: サーバーの再起動が行われることを確認する「y/n (はい / いいえ)」のプロンプトが表示されます。入力せずにコマンドを実行するには、`--ignore-prompt` オプションを含めます。

   たとえば、調整サービスを 3 *ノード* クラスタの 3 *ノード* (node1、node2、node3) すべてに展開します。

   ```
   tsm topology deploy-coordination-service -n node1,node2,node3
   ```

   コマンドが完了し、システム プロンプトに戻るまで待ちます。

6. (オプション) 展開が失敗した場合は、`tsm topology cleanup-coordination-service` コマンドを実行して、配置に失敗したアンサンブルを削除する必要があります。コマンドの実行の詳細については、tsm topology cleanup-coordination-service を参照してください。

7. Tableau Server を起動します。

   ```
   tsm start
   ```

新しい調整サービス アンサンブルを展開する方法の詳細については、調整サービス アンサンブルの展開を参照してください。

ステップ 7: 追加のノードでクライアント ファイル ・サービス (CFS) を構成する

CFS を追加のノードに追加します。調整サービスを実行しているすべてのノードに CFS を追加することをお勧めします。

> **注:** この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

1. 最初のノードで、ターミナル セッションを開きます。

2. CFS を追加しようとしているノードのノード ID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. ノード、プロセス、および単一のインスタンスを指定して、ノードに CFS を追加します。

   たとえば、次のコマンドは、CFS の 1 つのインスタンスを node2 に追加します。

   ```
   tsm topology set-process -n node2 -pr clientfileservice -c 1
   ```

   CFS を使用して既に構成されているノードに CFS のインスタンスを追加しようとすると、エラー メッセージが表示され、ノードに既にインスタンスが存在していることが通知されます。

CFS をさらに他のノードに追加するには、ノードごとにこの手順を繰り返します。

4. 変更を適用します。

```
tsm pending-changes apply
```

ステップ 8: ノード 2 のプロセスを構成する

1. 最初のノードで、ノード 2 のトポロジを設定します。トポロジでは、ノードで実行するプロセスおよび各プロセスで実行するインスタンスの数を指定します。ノード構成の詳細については、ノードの構成を参照してください。その他のプロセスを追加すると、データエンジンが自動的に追加されます。データエンジンがいつ追加されるかについての詳細は、Tableau Server プロセスを参照してください。

   a. 次の新しいノードの node-id を取得します。 `tsm topology list-nodes -v`

   b. 新しいノードで実行するプロセスを次のように個別に指定します。

```
tsm topology set-process -n <node-id> -pr gateway -c 1

tsm topology set-process -n <node-id> -pr vizqlserver -c 2

tsm topology set-process -n <node-id> -pr vizportal -c 2

tsm topology set-process -n <node-id> -pr backgrounder -c
2

tsm topology set-process -n <node-id> -pr cacheserver -c 2

tsm topology set-process -n <node-id> -pr
indexandsearchserver -c 1

tsm topology set-process -n <node-id> -pr dataserver -c 2

tsm topology set-process -n <node-id> -pr filestore -c 1

tsm topology set-process -n <node-id> -pr pgsql -c 1

tsm topology set-process -n <node-id> -pr metrics -c 1
```

```
tsm topology set-process -n <node-id> -pr activemqserver -c
1

tsm topology set-process -n <node-id> -pr tdsservice -c 1

tsm topology set-process -n <node-id> -pr
contentexploration -c 1

tsm topology set-process -n <node-id> -pr collections -c 1
```

2. ノード構成の変更を適用します。Tableau Server が再起動するというメッセージが表示されます。

   ```
   tsm pending-changes apply
   ```

ステップ 9: ノード 3 のプロセスを構成する

最初のノードで、ノード 3 のトポロジを設定します。トポロジでは、ノードで実行するプロセスおよび各プロセスで実行するインスタンスの数を指定します。その他のプロセスを追加すると、データエンジンが自動的に追加されます。データエンジンがいつ追加されるかについての詳細は、Tableau Server プロセスを参照してください。

1. 次の新しいノードの node-id を取得します。

   ```
   tsm topology list-nodes -v
   ```

2. 新しいノードで実行するプロセスを次のように個別に指定します。

   ```
   tsm topology set-process -n <node-id> -pr gateway -c 1

   tsm topology set-process -n <node-id> -pr vizqlserver -c 2

   tsm topology set-process -n <node-id> -pr vizportal -c 2

   tsm topology set-process -n <node-id> -pr backgrounder -c 2

   tsm topology set-process -n <node-id> -pr cacheserver -c 2
   ```

```
tsm topology set-process -n <node-id> -pr indexandsearchserver
-c 1

tsm topology set-process -n <node-id> -pr dataserver -c 2

tsm topology set-process -n <node-id> -pr filestore -c 1

tsm topology set-process -n <node-id> -pr metrics -c 1

tsm topology set-process -n <node-id> -pr tdsservice -c 1

tsm topology set-process -n <node-id> -pr contentexploration -c
1

tsm topology set-process -n <node-id> -pr collections -c 1
```

3. ノード構成を適用します。Tableau Server が再起動するというメッセージが表示されます。

   ```
   tsm pending-changes apply
   ```

4. サーバーを起動します。

   ```
   tsm start
   ```

ステップ 10: ファイアウォールのルールを構成する (オプション)

ローカル ファイアウォールを実行している場合は、クラスタに含まれるすべてのノードに対してファイアウォールのルールを構成する必要があります。詳細については、「ローカル ファイアウォール構成」と「ローカル ファイアウォールの構成」をご覧ください。

ロード バランサーの追加

この時点で 3 つすべてのノードにはゲートウェイがあり、利用できるサーバープロセスに要求を送るのに使用されます。全てのゲートウェイはアクティブですが、クラスタ内のダウンタイムをさらに削減するには、ロードバランサーを構成できます。詳細については、ロード バランサーの追加を参照してください。

ロードバランサーの追加

Tableau Server の信頼性を高めるには、複数のノードでゲートウェイを実行し、ロードバランサーを構成してゲートウェイ間で要求を分散します。アクティブまたはパッシブであるリポジトリプロセスとは異なり、すべてのゲートウェイプロセスはアクティブです。クラスタの 1 台のゲートウェイが使用できなくなると、ロードバランサーによりゲートウェイへの要求送信が停止します。選択したロードバランサーアルゴリズムによって、ゲートウェイがクライアント要求を送る方法が決まります。

- **Kerberos**: Kerberos 認証を使用する場合、Kerberos 用に Tableau Server を構成する前に、ロードバランサー用に Tableau Server を構成する必要があります。詳細については、Kerberos の構成を参照してください。
- **テストされたロードバランサー**: 複数のゲートウェイを使用する Tableau Server クラスタは、Apache および F5 ロードバランサーでテストされました。

  Apache ロードバランサーを使用してカスタム管理ビューを作成している場合、Tableau Server リポジトリに直接接続する必要があります。ロードバランサーを介して接続することはできません。
- **Tableau Server URL**: ロードバランサーが Tableau Server クラスタの前面にある場合、Tableau Server ユーザーがアクセスする URL は、初期 Tableau Server ノードではなくロードバランサーに属します。
- **単一ロードバランサー エンドポイント**: 単一の URL エンドポイントに対してロードバランサーを構成する必要があります。同じ Tableau Server の展開にリダイレクトするように、異なるエンドポイントホストを構成することはできません。Tableau Server を構成するときに、1 つの外部 URL が `gateway.public.host` で定義されます。詳細については、Tableau Server のプロキシとロードバランサーの設定を参照してください。
- **信頼できるホストの設定**: ロードバランサーを実行しているコンピューターは、信頼できるホストとして Tableau Server と認識される必要があります。詳細については、Tableau Server のプロキシとロードバランサーの設定を参照してください。

ロードバランサーを使用するための Tableau Server の構成

ロードバランサーを Tableau Server と識別するために使用する設定は、リバースプロキシ サーバーを識別するために使用する設定と同じです。Tableau Server クラスタにプロキシ サーバーとロードバランサーの両方が必要な場合、どちらも `gateway.public.host` に定義された単一の外部 URL を使用する必要があり、すべてのプロキシ サーバーとロードバランサーが `gateway.trusted` および `gateway.trusted_hosts` に指定されている必要があります。Tableau Server のプロキシとロードバランサーの設定を参照してください。

調整サービス アンサンブルの展開

調整サービス アンサンブルはオープン ソース プロジェクトである Apache ZooKeeper 上に構築され、サーバー上の活動を調整し、障害発生時にはクォーラムを保証し、サーバー トポロジー、校正、状態に関する「真実」の参照元としての役割を担います。このサービスは最初の Tableau Server ノードに自動的にインストールされますが、他のノードを追加しても追加のインスタンスはインストールされません。Tableau Server が順調に機能を発揮するには、調整サービスが適切に機能している必要があるため、3 台以上のサーバー インストール環境では、新しい調整サービス アンサンブルを展開することで調整サービスの追加のインスタンスを加えることを推奨します。これにより、調整サービスのインスタンスの 1 つに問題が発生した場合に冗長性および向上された可用性が提供されます。

- ハードウェア要件
- 調整サービス クォーラム
- 新規調整サービス アンサンブルの展開

ハードウェア要件

Tableau Server に使用するハードウェアは、調整サービスの正常な実行にある程度の影響を及ぼします。特に:

- **メモリ**。調整サービスは メモリ内に状態情報を保持します。設計上 メモリのフットプリントは小さく、一般的に、サーバーの全体的なパフォーマンスの要因ではありません。

- **ディスク速度**。サービス ストアはディスク上の情報を示すため、個々のノード コンピューター上のディスク速度が高速であるというメリットがあります。
- ノード間の**接続速度**。サービスは クラスタ ノード間で継続的に通信します。ノード間の接続速度が速いと、効率的に同期を実行できます。

調整サービスは I/O に負荷をかけるため、Tableau Server を最小ハードウェア要件をぎりぎりまたは少し超える程度のコンピューター上で実行している場合は、他のサーバープロセスに使用されていないノードにサービスを配置するように調整サービス アンサンブルを構成したほうが良いかもしれません。これにより、サーバープロセス間の I/O 接続による遅延の可能性を削減します。調整サービスのみの専用ノード上でアンサンブルを展開する方法についての情報は、下記の調整サービス専用ノードで高可用性用を Tableau Server に設定するを参照してください。

調整サービス クォーラム

調整サービスが適切に動作することを確認するには、サービスに クォーラム、すなわち最低限の数のサービスのインスタンスが必要です。つまり、インストールに含まれるノードの数は、アンサンブルで構成する必要がある調整サービスのインスタンスの数に影響を与えます。

使用する調整サービスインスタンス数

Tableau Server 上のアンサンブルで一度に使用できる調整サービルの最大数は、実装中の Tableau Server ノードの数に依存します。調整サービスアンサンブルをこれらのガイドラインに基づいて構成します。

サーバー ノードの合計数	**アンサンブル内の調整サービスノード推奨数 (1、3または5でなければなりません)**	**注**
1-2 ノード	1 ノード	これが既定であり、調整サービスを最初のノードから追加のノードに移動したい場合以外は設定を変更する必要はありません。
3-4	3 ノード	3 ノード調整サービス アンサンブルを使用すると、いずれかの

サーバーノードの合計数	アンサンブル内の調整サービスノード推奨数 (1、3または5でなければなりません)	注
ノード		アンサンブル ノードが失敗しても、Tableau Server で障害を発生することはありません。
5 以上のノード	3 ノード または 5 ノード	インストールできる調整サービスインスタンスの最大数は5台です。3 ノード調整サービス アンサンブルを使用すると、いずれかのアンサンブル ノードが失敗しても、Tableau Server で障害を発生することはありません。5 ノードのアンサンブルを使用すると、2 つのアンサンブル ノードが失敗しても Tableau Server で障害が発生することはありません。 ほとんどのインストールでは、3 つの調整サービス ノードが適切です。調整サービスでは I/O に負荷がかかるため、この構成にすることで、パフォーマンスが最も高くなります。 高可用性が絶対的な優先事項である場合は、5 ノードの調整サービス アンサンブルを展開することを検討してください。これにより、1 つ以上のノードで障害が発生したにもかかわらず、より多くのシステム リソースが必要になる場合であっても、最も高い冗長性を確保できます。(ノード上の他のサービスが、機能しているノードにも存在している限り)、最大 2 つのアンサンブル ノードが Tableau Server に影響を与えることなく失敗する可能性があります。 パフォーマンスへの影響を軽減するには、他のサービスをより少ない数で実行しているノードに調整サービスを配置するか、調整サービス専用 ノードを使用することを検討してください。詳細については、調整サービス専用 ノードで高可用性用を Tableau Server に設定するを参照してください。

新規調整サービス アンサンブルの展開

合計 3 台以上のノードをインストールした場合、調整サービス アンサンブルも展開してください。これを展開しないと、サーバーの構成やトポロジに変更を加えるたびに警告メッセージが表示されるようになります。このメッセージは無視できますが、ベストプラクティスはマルチノードの調整サービス アンサンブルを展開することです。

Tableau Server をインストールすると、調整サービスのインスタンスが最初のノードに 1 つインストールされます。TSM とTableau Server は調整サービスが正しく機能することに依存しているため、冗長性を提供し、マルチノード インストールでの可用性を確保するため、調整サービス アンサンブルを展開することで調整サービスの追加のインスタンスを構成します。調整サービス アンサンブルは 1、3、または 5 個の調整サービス インスタンスでインストールされます。Tableau Server の 3 ノード インストールでは、推奨される調整サービスのインスタンス数は 3、つまり各 ノードに 1 つです。

保留中の他の変更がある場合は、調整サービス アンサンブルの展開を試行しないでください。新しい調整サービス アンサンブルを展開する前に、保留中の変更を破棄または適用します。

> **注:** この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

以下の手順は、既存の 3 ノードTableau Server クラスタ上に新しい調整サービス アンサンブルを展開し、旧アンサンブルをクリーンアップする方法を説明しています。

1. 最初のノードで、ターミナル セッションを開きます。
2. Tableau Server を停止します。

   ```
   tsm stop
   ```

   メッセージが表示されたら、TSM 管理者としてサインインします。

   管理コントローラーや管理エージェントなど、継続して実行されるTSM プロセスもあります。
3. 保留中の変更がないことを確認します。

```
tsm pending-changes list
```

保留中の変更がある場合は、変更を破棄するか、変更を適用する必要があります。保留中の変更の適用には少し時間がかかります。

- 変更を破棄します。

  ```
  tsm pending-changes discard
  ```

  または

- 変更を適用します。

  ```
  tsm pending-changes apply
  ```

コマンドが完了し、システム プロンプトに戻るまで待ちます。

4. クラスタ内の各 *ノード*の*ノード*ID を取得します。

   ```
   tsm topology list-nodes -v
   ```

5. `tsm topology deploy-coordination-service` コマンドを使用して調整サービスを指定された*ノード*に追加することで、新しい調整サービス アンサンブルを追加します。各*ノード*を識別するために実際の*ノード*ID を使用して、調整サービスを追加する*ノード*を指定する必要があります。展開が失敗しなければ、このコマンドにより、新しいアンサンブルが "本番環境" のアンサンブル (使用中のアンサンブル) になり、古いアンサンブルが削除されます。このような場合は、以下のステップ 6 を参照してください。

   注: サーバーの再起動が行われることを確認する「y/n (はい / いいえ)」のプロンプトが表示されます。入力せずにコマンドを実行するには、`--ignore-prompt` オプションを含めます。

   たとえば、調整サービスを 3 *ノード* クラスタの 3 *ノード* (node1、node2、node3) すべてに展開します。

   ```
   tsm topology deploy-coordination-service -n node1,node2,node3
   ```

コマンドが完了し、システム プロンプトに戻るまで待ちます。

6. (オプション) 展開が失敗した場合は、`tsm topology cleanup-coordination-service` コマンドを実行して、配置に失敗したアンサンブルを削除する必要があります。コマンドの実行の詳細については、tsm topology cleanup-coordination-service を参照してください。

7. Tableau Server を起動します。

   ```
   tsm start
   ```

クライアント ファイル サービスの構成

複数ノードのクラスタ内でほとんどの共有ファイルを管理するのが Client File Service (CFS) です。例えば、認証関連の証明書、キー、ファイル (OpenID、相互 SSL、SAML、Kerberos)、カスタマイズ ファイルなどが CFS によって管理されます。

Tableau サービス マネージャー (TSM) とTableau Server を初めてインストールすると、最初のノードで単一インスタンスの CFS が構成されます。他のノードで追加のインスタンスを構成できる必要があります。

クラスターでは、CFS の唯一のインスタンスを実行しているノードに障害が発生すると、CFS によって管理されているファイルが失われるため、証明書とカスタム イメージをもう一度インポートし、関連する構成を変更して、CFS のこれらのファイルをもう一度設定する必要があります。CFS によって管理されるファイルのリストについては、Tableau Server クライアント ファイル サービスを参照してください。

これらの制限事項と推奨事項に留意してください。

- Tableau Server インストールに最低 1 インスタンスの CFS が必要です。CFS のインスタンスをすべて削除することはできません。
- いずれのノードにも 1 つを超えるインスタンスを構成することはお勧めしません。追加インスタンスを構成してもメリットは一切なく、リソースの浪費となります。

- 調整サービスを展開する各ノードで CFS インスタンスを構成することをお勧めします。これに

より冗長性が提供され、ノードに障害が発生した場合に予期しないエラーからTableau Server を保護できます。

この記事では、CFS のインスタンスを追加または削除する方法を説明します。CFS の構成は Web UI から行うことはできません。CFS の追加や削除には、CLI を使用する必要があります。この手順はプロセスをノードに追加するのに使用する手順と同じです。

追加ノードでの CFS の構成

1. 最初のノードで、ターミナル セッションを開きます。
2. CFS を追加しようとしているノードのノードID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. ノード、プロセス、および単一のインスタンスを指定して、ノードに CFS を追加します。

   たとえば、次のコマンドは、CFS の 1 つのインスタンスを node2 に追加します。

   ```
   tsm topology set-process -n node2 -pr clientfileservice -c 1
   ```

   CFS を使用して既に構成されているノードに CFS のインスタンスを追加しようとすると、エラー メッセージが表示され、ノードに既にインスタンスが存在していることが通知されます。

   CFS をさらに他のノードに追加するには、ノードごとにこの手順を繰り返します。

4. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

リポジトリフェールオーバー

Tableau Server のインストールにおいて、リポジトリ(pgsql) データベースはキーが必要なプロセスの 1 つです。Tableau Server リポジトリには、Tableau Server ユーザー、グループおよびグループ割り当て、パーミッション、プロジェクト、データ ソース、および抽出メタデータと更新情報に関する情報が格納されています。これはサーバーが機能するために重要なものであるため、一定の要件

を満たす分散環境にサーバーがインストールされるときに、リポジトリの自動「フェールオーバー」が Tableau Server に組み込まれます。

自動リポジトリフェールオーバー

自動リポジトリフェールオーバーとは、アクティブな Tableau Server リポジトリに問題がある場合に、パッシブ リポジトリを使用するようにサーバーが自動的に切り替えることです。リポジトリの一時的な問題によっては、スイッチの切り替えが適切でない場合もあるため、フェールオーバーはすぐには実行されませんが、リポジトリが 1 ~ 5 分以上使用できない場合は、フェールオーバーが発生します。たとえば、基盤となる PostgresSQL サービスに障害が発生した場合、約 1 分以内にフェイルオーバーが発生します。ただし、他の構成の問題によっては、最大 5 分間障害が発生しない場合があります。

自動リポジトリフェールオーバーが機能するには、Tableau Server インストールで次の条件が満たされる必要があります。

- ノードが 3 つ以上ある
- リポジトリの 2 つのインスタンスがインストールされている

  オプションであるが強く推奨:

- マルチノード調整サービス アンサンブルの展開

これらの条件が満たされていると、プロセスに関する問題や、プロセスが実行されているノードに関する問題によってアクティブなリポジトリが使用できなくなったときに、リポジトリフェールオーバーが発生します。元のリポジトリが再び利用可能になったら(たとえば、ノードが再起動され、すべてのプロセスが正常に発生する状態)、元のリポジトリがパッシブ リポジトリになり、必要に応じてフェールオーバー用に利用できるようになります。

手動リポジトリフェールオーバー

フェールオーバーの発生後に元のリポジトリに戻したい理由が生じる場合があります。理由の 1 つとして、リポジトリのインスタンスが、リソースの多いコンピューターにインストールされている場合があります。これを行うには、`tsm topology failover-repository` コマンドを使用して手動で元のリポジトリに戻します。詳細については、tsm topology failover-repository を参照してください。

優先するアクティブなリポジトリ

Tableau Server を構成するとき、優先アクティブ リポジトリとしてノードを指定することができます。Tableau Server がリポジトリフェールオーバー用に構成されている場合、この優先アクティブ リポジトリ ノードがアクティブ リポジトリ用に使用されます。このステップはオプションです。優先アクティブ リポジトリ ノードを指定しない場合、Tableau Server は起動時にアクティブ リポジトリ ノードを選択します。

優先アクティブ リポジトリを構成するには、`tsm configuration set` コマンドを使用して、`pgsql.preferred_host` オプションを次のように構成します。

```
tsm configuration set -k pgsql.preferred_host -v "<host_name>"
```

注: `host_name` は大文字と小文字が区別され、`tsm status -v` の出力に表示されるノード名と一致する必要があります。

Tableau Server が起動時に特定のノードを選択するようにする場合は、優先するアクティブなリポジトリを設定します。アクティブなリポジトリ(たとえば、ディスク容量やメモリの大きいコンピューター)に特定のサーバーを使用する場合や、カスタムの管理ビューを使用している場合、この操作が必要となる場合があります。カスタム管理ビューには、ビューを作成したリポジトリを参照する接続情報が埋め込まれています。Tableau Server リポジトリへの接続の詳細については、Tableau Server リポジトリでデータを収集するを参照してください。

初期ノードの障害からの回復

Tableau を最初にインストールするコンピューターである「初期ノード」には、固有の特徴があります。初期ノード上で実行できるのは、ライセンス サービス (ライセンス マネージャー)、ライセンス認証サービス、TSM コントローラー (管理コントローラー) の 3 つのプロセスのみで、エラーが発生した場合を除き、これらのプロセスを他のノードに移動させることはできません。Tableau Server には、これらのプロセスを他のいずれかの既存ノードに移動するのを自動化するスクリプトが含まれています。これにより、TSM に戻るための完全なアクセス権を取得し、Tableau Server の実行を維持できるようになります。

他の 2 つのプロセスはまず初期ノードに含められますが、追加のノード、CFS (Client File Service)、調整サービスに追加・移動させることができます。インストールが CFS および調整サービスでどのように構成されていたかによって、これらを再デプロイする手順も必要になる場合があります。

最初のノードに障害が発生した場合

初期ノードに問題があり、他のノードに冗長プロセスがある場合、Tableau Server が引き続き実行できるという保証はありません。

- Tableau Server は、第一ノードに障害が発生してから最大 72 時間、ライセンスサービスの欠如によって他のプロセスが影響を受け始める前まで、実行を継続することができます。初期ノードでエラーが発生しても、ユーザーは引き続きサインインしてコンテンツを表示および使用することはできますが、管理コントローラーへのアクセス権がなくなるため、Tableau Server の再構成ができなくなります。
- ATR 用に構成された Tableau Server 2021.4.2 (またはそれ以前) のバージョンを実行している場合、第一ノードに問題が発生すると、すべてのサーバー機能が使用できなくなります。これは、初期ノードに問題がある場合でも、意図的に初期ノードを停止している (たとえば、システムレベルパッチを適用するために) 場合でも当てはまります。

冗長プロセスで構成されている場合でも、*初期ノードに障害が発生すると、Tableau Server が機能し続けることができなくなる可能性があります*。これは、インストールが高可用性用に構成されている場合でも当てはまります。つまり、できるだけ早く、2 つの固有のプロセスを別の実行中のノードに移動させる必要があります。相対的に時間が足りず復元できないという理由で初期のノードに失敗が起きる場合 (修正できるハードウェアのエラーなど)、まずは以下の手順を実行せずにノードのバックアップを試行する必要があります。

注: この記事の手順ではサーバーのダウンタイムが必要となり、業務に支障が出る可能性があります。したがって、最初のノードに非常に重大な障害が発生した場合にのみ実行する必要があります。最初のノードを再稼働できない場合は、次の手順を使用して主要な TSM プロセスをクラスタ内の別のノードに移動してください。

一般的な要件

Tableau Server バージョン 2020.1 では復元機能が強化されました。このトピックの手順は、Tableau Server 2021.1 用に作成されています。

障害が発生したノードを以前のバージョンの Tableau Server から復元しようとしている場合は、そのバージョン用の手順に従う必要があります。Tableau ヘルプのアーカイブ バージョンを表示するには、Tableau ヘルプを参照してください。

- マルチノード Tableau Server インストールをセットアップするプロセスの一環として、調整サービス アンサンブルを展開してあるはずです。下記のプロセスは、最初のノードで問題が発生する前に調整サービス アンサンブルが展開されていることを前提にしています。新しい調整サービス アンサンブルを展開する方法について詳細は、調整サービス アンサンブルの展開を参照してください。
- このプロセスでは、調整サービスを実行しているすべてのノードでクライアント ファイル サービス (CFS) のインスタンスを構成していることを前提にしています。CFS のインスタンスを追加しなかった場合は、最初のノードのみにインスタンスがあるため、CFS の少なくとも 1 つのインスタンスを別のノードに追加する必要があります。CFS を再読み込みする必要もあります。Tableau Server には最低 1 インスタンスの CFS が必要です。詳細については、クライアント ファイル サービスの構成および Tableau Server クライアント ファイル サービスを参照してください。

注: この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

TSM コントローラー、ライセンス サービス、アクティブ化サービスを別のノードに移動する

最初のノードで問題が発生した場合、TSM コントローラー、ライセンス発行サービス、アクティブ化サービスを別のノードで起動する必要があります。次のステップに従い、提供された `move-tsm-controller` スクリプトを使用して、TSM コントローラー、ライセンス発行サービス、アクティブ化サービスを別のノードで動作させます。

1. まだ動作しているノード上で、コントローラー復旧スクリプトを実行します。動作中ノード上のターミナルのプロンプトで、次のコマンドを入力します。

   ```
   sudo /opt/tableau/tableau_server/packages/scripts.<version_
   code>/move-tsm-controller -n <nodeID>
   ```

   "nodeID" には TSM コントローラーを実行したいノードの ID が入ります。例:

   ```
   sudo /opt/tableau/tableau_
   server/packages/scripts.10400.17.0802.1319/move-tsm-controller
   -n node2
   ```

2. 管理コントローラーがノード上で実行されていることを確認します。

   ```
   tsm status -v
   ```

3. Tableau Server を停止します。

   この手順の残りの部分には、--ignore-node-status オプションを使用する一部のコマンドが含まれています。--ignore-node-statusオプションを使用してコマンドを実行すると、指定されたノードの状況を考慮せずにコマンドが実行されます。--ignore-node-statusを使用するには、障害が発生したノードを指定します。

   ```
   tsm stop --ignore-node-status <nodeID>
   ```

   たとえば、node1 に障害が発生した場合は、次のコマンドを実行します。

   ```
   tsm stop --ignore-node-status node1
   ```

4. ノードにライセンス サービスを追加します。

   ```
   tsm topology set-process -pr licenseservice -n <nodeID> -c 1
   ```

5. 元のノードから旧ライセンス サービスを削除します。"nodeID" は障害の発生した元の最初のノードです。

   ```
   tsm topology set-process -pr licenseservice -n <nodeID> -c 0
   ```

6. 次のバージョンのいずれかを実行している場合

    - 2023.3.0 以降
    - 2023.1.3 以降
    - 2022.3.7 以降
    - 2022.1.15 以降

    または、以前のバージョンを実行していてATR を使用している場合は、有効化サービスを新しいノードに追加します。

    ```
    tsm topology set-process -pr activationservice -n <nodeID> -c 1
    ```

7. 次のバージョンのいずれかを実行している場合

    - 2023.3.0 以降
    - 2023.1.3 以降
    - 2022.3.7 以降
    - 2022.1.15 以降

    または、以前のバージョンを実行していてATR を使用している場合は、元のノードから古い有効化サービスを削除します。「nodeID」は障害が発生した最初のノードです。

    ```
    tsm topology set-process -pr activationservice -n <nodeID> -c 0
    ```

    重要: クラスターでは、CFS の唯一のインスタンスを実行しているノードに障害が発生すると、CFS によって管理されているファイルが失われるため、証明書とカスタム イメージをもう一度インポートし、関連する構成を変更して、CFS のこれらのファイルをもう一度設定する必要があります。CFSによって管理されるファイルのリストについては、Tableau Server クライアント ファイル サービスを参照してください。

8. 最初のノードでメッセージング サービスのみを実行していた場合は、このノードにメッセージング サービスを追加します。

   ```
   tsm topology set-process -pr activemqserver -n node2 -c 1
   ```

9. (オプション) 最初のノードで実行されていて、このノードでは実行されていなかったその他のプロセスも追加できます。たとえば、キャッシュ サーバーを追加するには次を入力します。

   ```
   tsm topology set-process -pr cacheserver -n node2 -c 1
   ```

10. 変更を適用します。

    ```
    tsm pending-changes apply --ignore-node-status <nodeID>
    ```

    保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

11. TSM 管理コントローラーを (*tableau* システム アカウントとして) を再起動します。

    ```
    sudo su -l tableau -c "systemctl --user restart
    tabadmincontroller_0.service"
    ```

    > **注:** tabadmincontroller の再起動には数分かかる場合があります。コントローラーを完全に再起動させる前に次の手順で保留中の変更を適用しようとする場合、TSM はコントローラーに接続できなくなります。`tsm status -v` コマンドを使用して、コントローラーが実行されていることを確認できます。Tableau Server 管理コントローラーが「実行中」として表示されます。

12. 保留中の変更を適用します (変更は何もないように見えますが、このステップは必要です)。

```
tsm pending-changes apply --ignore-node-status <nodeID>
```

13. Tableau Server ライセンスを新しいコントローラー ノード上で認証します。

    ```
    tsm licenses activate -k <product-key>
    ```

14. ライセンスが適切に認証されていることを確認します。

    ```
    tsm licenses list
    ```

15. 最初のノードで調整サービスを実行していた場合は、そのノードに含まれていない新しい調整サービス アンサンブルを展開する必要があります。ノード クラスタが 3 つあり、最初のノードで調整サービスを実行していた場合、新しい単一インスタンスの調整サービス アンサンブルを別のノードに展開し、古いアンサンブルをクリーンアップする必要があります。次の例では、単一インスタンスの調整サービスが 2 番目のノードに展開されています。

    ```
    tsm topology deploy-coordination-service -n node2 --ignore-
    node-status node1
    ```

16. 最初のノードでファイル ストア インスタンスを実行していた場合は、このインスタンスを削除する必要があります。

    ```
    tsm topology filestore decommission -n <nodeID> --delete-
    filestore
    ```

    ここで nodeID は障害が発生した最初のノードです。

17. 上記で展開した新しい調整サービス アンサンブルが単一ノードのアンサンブルの場合は、--ignore-warnings フラグを使用して保留中の変更を適用します。

    ```
    tsm pending-changes apply --ignore-node-status node1 --ignore-
    warnings
    ```

18. 最初のノードを削除します。nodeID は障害が発生した最初のノードです。

    ```
    tsm topology remove-nodes -n <nodeID>
    ```

19. 上記で展開した新しい調整サービス アンサンブルが単一ノードのアンサンブルの場合は、--ignore-warnings フラグを使用して保留中の変更を適用します。

    ```
    tsm pending-changes apply --ignore-warnings
    ```

20. Tableau Server を起動します。

    ```
    tsm start
    ```

    この時点でサーバーが起動し、TSM を使用して構成を行うことができます。次のステップでは最初のノードを置換し、クラスタに元のノード数が含まれるようにします。この実行方法は、エラーが発生したノードを再利用するかどうかによって変わります。エラーが発生した理由を特定できない場合にはそのノードのみを再利用し、エラーが再発しないように対策を取ることをお勧めします。

21. 元のノードの再利用を計画している場合は、まずはそこからTableau を完全に削除する必要があります。これを行うには tableau-server-obliterate スクリプトを実行します。この実行の詳細については、コンピュータからTableau Server を削除するを参照してください。

22. 新しいコンピューター、または Tableau を完全に削除した後の元のコンピューターで、元のセットアップ プログラムと、現在管理コントローラーおよびライセンス発行サービスを実行中のノードから生成された bootstrap ファイルを使用して Tableau をインストールします。これにより、クラスターの一部として構成できる追加のノードが作成されます。ノード追加方法の詳細については、追加のノードのインストールと構成を参照してください。

    ベストプラクティスは、元のノードでエラーが発生したときに失ったプロセスを構成し、クラスタでの十分な冗長性を確保することです。新しい最初のノードから、新しく追加した追加ノードにプロセスを移動し、元の構成を複製することが必要な場合があります。たとえば、最初のノードがゲートウェイとファイル ストアのみを実行していた場合、同様の方法で新しい最初のノードの構成が必要となる場合があります。

23. 必要な方法でノードを起動して実行したら、新しい調整サービス アンサンブルの再展開も必要です。詳細は、調整サービス アンサンブルの展開を参照してください。

24. 最後に、まだ実施していない場合は、調整サービスを実行しているすべてのノードに CFS のインスタンスを追加します。詳細については、クライアント ファイル サービスの構成を参照してください。クライアント ファイル サービスの構成

    クラスターでは、CFS の唯一のインスタンスを実行しているノードに障害が発生すると、CFS によって管理されているファイルが失われるため、証明書とカスタム イメージをもう一度インポートし、関連する構成を変更して、CFS のこれらのファイルをもう一度設定する必要があります。CFS によって管理されるファイルのリストについては、Tableau Server クライアント ファイル サービスを参照してください。

ノードの障害からの回復

サーバー ノードのいずれかに問題があり、他のノードに冗長プロセスがある場合、Tableau Server は引き続き実行可能です。ユーザーは引き続きサインインし、ノード障害発生後にコンテンツを表示して使用することはできますが、ノード障害のためにパフォーマンスが低下する場合があります。また、不適切なノードが冗長性のなくなったプロセスを実行している場合、サーバーが致命的な障害にさらされるリスクが大きくなります。つまり、不適切なノードを取り除き、可能な限り早くそのノードを置き換える必要があります。相対的に時間が足りず復元できないという理由でノードに障害が発生する場合 (修正できるハードウェアのエラーなど)、まずは以下の手順を実行せずにノードのバックアップを試行する必要があります。

注: 障害が発生したノードが*最初のノード*の場合、Tableau Server インストールには大きな影響があります。たとえば、Prep Web Authoring が機能するには、設定済みの最も番号の小さいノードでキャッシュ サーバーが実行されている必要があります。キャッシュ サーバーがノード 2、3、および 4 に構成されているが、ノード 2 でダウンしている場合、Prep Web Authoring は機能しません。最初のノードの障害から復旧する方法について詳細は、初期ノードの障害からの回復を参照してください。キャッシュサーバーがダウンしているときに Prep Web Authoring が動作しない場合の詳細については、ナレッジベースの「キャッシュ サーバーが最も小さい番号のノードで実行されない」を参照してください。

一般的な要件

Tableau Server バージョン 2020.1 では回復機能が強化されました。このトピックの手順は、Tableau Server 2020.1 用に作成されています。

障害が発生したノードを以前のバージョンの Tableau Server から復元しようとしている場合は、そのバージョン用の手順に従う必要があります。Tableau ヘルプのアーカイブ バージョンを表示するには、Tableau ヘルプを参照してください。

- ファイル ストアのインスタンスが存在する機能しているノードが 1 つ以上あること。
- リポジトリが存在する機能しているノードが 1 つ以上あること。
- クライアント ファイル サービス (CFS) が存在する機能しているノードが 1 つ以上あること。

注: この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

障害が発生したノードの削除

Tableau Server クラスタから障害が発生したノードを削除するには:

1. 障害が発生したノードを特定します。

   ```
   tsm status -v
   ```

   障害が発生したノードのステータスは "ERROR (エラー)" となり、プロセスは使用不可と表示されます。ノード ID は "node<n>" と記載され、その後にマシン名が続きます。たとえば、node3 では以下のようになります。

   ```
   node3: WIN-OO915SFASVH
                                                    Status: ERROR
                                  'Tableau Server Gateway 0' status is una
   ```

2. Tableau Server を停止します。

この手順の残りの部分には、--ignore-node-status オプションを使用する一部のコマンドが含まれています。--ignore-node-statusオプションを使用してコマンドを実行すると、指定されたノードの状況を考慮せずにコマンドが実行されます。--ignore-node-statusを使用するには、障害が発生したノードを指定します。

```
tsm stop --ignore-node-status <nodeID>
```

たとえば、node3 に障害が発生した場合は、次のコマンドを実行します。

```
tsm stop --ignore-node-status node3
```

3. ノード上で実行されていた主要なプロセスを判断します。

   - 障害が発生したノードがメッセージング サービスを実行していた場合は、障害が発生したノードからサービスを削除し、それを機能しているノードに追加する必要があります。

     障害が発生したノードから削除します。

     ```
     tsm topology set-process -pr activemqserver -n <nodeID> -c
     0
     ```

     次の機能しているノードに追加します。

     ```
     tsm topology set-process -pr activemqserver -n <nodeID> -c
     1
     ```

   - 障害が発生したノードが調整サービスを実行していた場合、そのノードを削除する前に、新しいアンサンブルを展開する必要があります。

     ```
     tsm topology deploy-coordination-service -n <good_nodeID>
     --ignore-node-status <failed_nodeID>
     ```

   - 障害が発生したノードがクライアント ファイル サービス (CFS) のインスタンスのみを実行していた場合、機能するノードで CFS の新しいインスタンスを作成する必要があります。調整サービスを実行しているすべてのノードに CFS を設定することをお勧め

します。詳細については、クライアント ファイル サービスの構成を参照してください。

- 障害が発生したノードがファイル ストアを実行していた場合、ノードを削除する前に、強制的にファイル ストアをコミッション解除して削除する必要があります。

  ```
  tsm topology filestore decommission -n <nodeID> --delete-filestore
  ```

  保留中の変更を適用します (3 ノード クラスタと単一の調整サービス インスタンスがある場合は、`--ignore-warnings` オプションを使用します):

  ```
  tsm pending-changes apply --ignore-warnings --ignore-node-status <nodeID>
  ```

4. クラスタが 3 ノード クラスタであり、動作している残りのノードにリポジトリがある場合、1 つのリポジトリを削除するか、新しいノードを追加する必要があります。ノード数が 3 未満の場合、ユーザーは単一インスタンスのリポジトリに制限されるためです。

   1 つのリポジトリを削除するには、次を実行します。

   ```
   tsm topology set-process -n <nodeID> -pr pgsql -c 0
   ```

5. コマンドを実行し、障害が発生したノードを削除します。これにより、保留中の変更のリストに変更が追加されます。

   ```
   tsm topology remove-nodes -n <nodeID>
   ```

6. ノードの削除が保留中であることを確認してください。

   ```
   tsm pending-changes list
   ```

7. 保留中の変更を適用し、ノードを削除します。

   ```
   tsm pending-changes apply
   ```

8. Tableau Server を起動します。

   ```
   tsm start
   ```

9. 新しいノードに Tableau Server をインストールし、このノードにおいて、障害が発生した古いノードで実行していたプロセスを構成します。

10. 新しいコンピューターまたは Tableau を完全に削除した後の元のコンピューターで、元のセットアップ プログラムと、最初のノードから生成された bootstrap ファイルを使用して Tableau をインストールします。この実行方法の詳細については、追加のノードのインストールと構成を参照してください。

    ベストプラクティスは、元のノードでエラーが発生したときに失ったプロセスを構成し、クラスタでの十分な冗長性を確保することです。

11. 必要な方法でノードを起動して実行したら、新しい調整サービス アンサンブルの再展開も必要です。詳細は、調整サービス アンサンブルの展開を参照してください。

12. 最後に、まだ実施していない場合は、調整サービスを実行しているすべてのノードに CFS のインスタンスを追加します。詳細については、クライアント ファイル サービスの構成を参照してください。クライアント ファイル サービスの構成

ノードの構成

Tableau サービス マネージャー (TSM) の Web UI または CLI コマンドを使用してノードのトポロジを構成します。最初のノードは既定どおりに構成され、この構成には TSM と Tableau Server で使用されるすべてのプロセスが含まれます。ノードを追加する場合は、これらのノードで実行するプロセスを指定する必要があり、プロセスのインスタンスを何種類実行するかも指定する必要があります。また、最初のノードのトポロジを変更する場合もあります。既存のプロセスのインスタンスを追加したり、これらのプロセスのいくつかを追加のノードに移動したりできます (Tableau Server の分散インストールを設定する場合に一般的)。

注: クラスタにそのプロセスのインスタンスがこれしかない場合、ファイル ストアまたは リポジトリ (pgsql) を削除できません。Tableau Server のインストールの各プロセスで、少なくとも 1 つのインスタンスが必要です。さらに、元のノード上のプロセスを削除するには、事前に 2 番目のイ

ンスタンスを追加して、そのインスタンスと最初のインスタンスとの同期を許可する必要があります。詳細については、リポジトリプロセスの移動およびファイル ストアのプロセスの移動を参照してください。

TSM Web インターフェイスの使用

ほとんどの場合、複数の変更をサーバー構成に加えることが可能です。ファイル ストア インスタンスまたは リポジトリを移動または削除する場合は例外です。

Tableau Server ノードを TSM web インターフェイスを使用して構成するには、以下を行います。

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

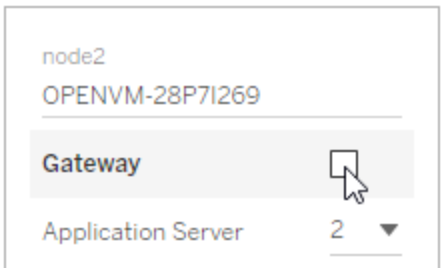
2. **[構成]** タブをクリックします。

   次のステップは、行いたい構成の変更によります。

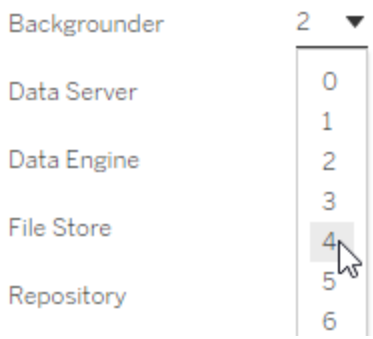
   - **プロセスの追加**—ノードに必要なインスタンスの数を指定するか、プロセスのボックスを選択してノードにプロセスを追加します。

     例えば、バックグラウンダーの 4 つのインスタンスと、ゲートウェイをノードに追加します。

     a. **ゲートウェイ** ボックスを選択します。



b. **バックグラウンダー**カウントを4 に設定。



バックグラウンダーをノードに追加すると、まだノードにない場合、データエンジンのインスタンスも追加します。

> **注:** TSM の Web UI はインスタンス数の選択を許可するプロセスを最大8 インスタンスに制限します。これより多くのインスタンスを構成するには、コマンドラインと`TSM topology set-process` コマンドを使用してください。詳細については、tsm topology set-processを参照してください。

- **プロセスカウントの変更**—新しい数字を指定するか、またはプロセスのボックスを選択してノードでプロセス数を変更します。
- **プロセスを完全に削除する**—プロセスのボックスをクリアするか、カウントをゼロに設定してノードからプロセスのインスタンスをすべて削除します。

ほとんどの場合、あるノードから別のノードにプロセスを移すには、最初のノード上のプロセス インスタンスのカウントを0 に設定し、2 番目のノードのカウントにゼロ以外の値を設定します。

許容されていない構成を作成しようとすると(例えば、使用停止になっていないファイル ストアを削除しようとする)、それを知らせるメッセージが表示されます。

3. 右上の **[変更を保留中]**、**[変更を適用して再起動]** をクリックし、変更を確定して Tableau Server を再起動します。

TSM CLI の使用

- ノードにプロセスを追加する
- ノードのプロセス数を変更する
- ノードからプロセスのインスタンスをすべて削除する
- あるノードから別のノードにプロセスのすべてのインスタンスを移す

ノードを構成するには、最初のノードからコマンドを実行し、構成しているノードをノード ID で指定します。ノード ID を特定するには、`tsm topology list-nodes` コマンドを使用します。ノードにプロセスを追加したり、プロセスを更新または削除するには、`tsm topology set-process` コマンドを使用します。ここでは、構成するノード、追加、更新、または削除するプロセス、およびプロセスのインスタンスの数を指定する必要があります。ノードのトポロジの設定が終了したら、Tableau Server に変更を適用する必要があります。

`tsm pending-changes apply` コマンドを使用して変更を適用します。変更の適用後、Tableau Server はコマンドを実行する前の状態に戻ります。つまり、稼働中であった場合は再起動され、停止中だった場合は、保留中の変更の適用後、停止した状態のままになります。ほとんどの場合、保留中の変更を適用するときに Tableau Server が稼働中の場合、変更を適用するためにサーバーがいったん停止され、再起動されます。ただし、既存のノード上のバックグラウンダー、、または VizQL Server のインスタンス数を変更する場合は例外です。既存のノード上でこれらのプロセスを変更する際は、Tableau Server が稼働中であっても停止する必要がありません。

ノードを構成するには、ノードのノード ID が必要です。ノード ID を特定するには次のコマンドを使用します。

```
tsm topology list-nodes -v
```

注: この例で示しているのは一部のプロセス名です。完全なリストについては、Tableau Server プロセスを参照してください。

ノードにプロセスを追加する

プロセスをノードに追加するには `tsm topology set-process` コマンドを使用します。コマンドには構成するノード、追加するプロセス、プロセスのインスタンス数を指定する必要があります。

1. 最初のノードで、ターミナル セッションを開きます。
2. 変更しようとしているノードのノード ID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. プロセスおよびインスタンス数を指定し、ノードにプロセスを追加します。

   たとえば、次のコマンドは、バックグラウンダーの 2 つのインスタンスを node1 に追加します。

   ```
   tsm topology set-process -n node1 -pr backgrounder -c 2
   ```

4. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

ノードのプロセス数を変更する

ノードのプロセス数を変更するには、構成済みのプロセスを指定してから、インスタンス数の新しい値を指定します。

1. 最初のノードで、ターミナル セッションを開きます。
2. 変更しようとしているノードのノード ID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. ノードのプロセス数を変更するには、構成済みのプロセスを指定してから、インスタンス数の新しい値を指定します。

たとえば、バックグラウンダーをすでに実行しているノード(node1) で、次のコマンドはインスタンスの数を4 に変更します。

```
tsm topology set-process -n node1 -pr backgrounder -c 4
```

4. 変更を適用します。

```
tsm pending-changes apply
```

ノードからプロセスのインスタンスをすべて削除する

1. 最初のノードで、ターミナル セッションを開きます。
2. 変更しようとしているノードのノードID を確認します。

```
tsm topology list-nodes -v
```

3. ノードからプロセスを削除するには、対象となるノードのプロセスに対してカウント0 のインスタンスを指定します。

たとえば、次のコマンドは node1 からバックグラウンダーのプロセスを削除します。

```
tsm topology set-process -n node1 -pr backgrounder -c 0
```

4. 変更を適用します。

```
tsm pending-changes apply
```

あるノードから別のノードにプロセスのすべてのインスタンスを移す

ほとんどの場合、あるノードから別のノードにプロセスを移すには、最初のノード上のプロセス インスタンスのカウントを0 に設定し、2 番目のノードのカウントにゼロ以外の値を設定します。

1. 最初のノードで、ターミナル セッションを開きます。
2. 変更しようとしているノードのノードID を確認します。

```
tsm topology list-nodes -v
```

3. あるノードから別のノードにプロセスを移動するには、最初のノードでそのプロセスのインスタンス数に 0 を指定し、2 つ目のノードに 1 以上の数を指定します。

   たとえば、次のコマンドは、node1 からバックグラウンダーを削除し、そのバックグラウンダーの 2 つのインスタンスを node2 に追加します。

   ```
   tsm topology set-process -n node1 -pr backgrounder -c 0

   tsm topology set-process -n node2 -pr backgrounder -c 2
   ```

4. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

ノード ロールによるワークロード管理

ノード ロールを使用すると、Tableau Server インストールで特定のタイプのワークロードが処理される場所を構成できます。ノード ロール機能を使用すると、リソースを特定のワークロード専用にしたり、ワークロードに合わせて拡張したりできます。バックグラウンダーとファイル ストアのノード ロールを構成できます。

バックグラウンダー ノード ロールは、ノードで実行する必要があるバックグラウンド タスクのタイプを指定しますが、ファイル ストア ノード ロールは、ノードで実行する必要がある抽出のワークロードのタイプを指定します。どちらのノード ロールも、ノード レベルで指定されます。これらのノード ロールは独立して機能して選択したワークロードを最適化できますが、2 つのノード ロールを組み合わせて使用すると、サーバー ノードを、選択したワークロードを優先的に実行し、抽出負荷の大きいワークロードでパフォーマンスを最適化するための専用のノードにすることができます。この組み合わせについては、後ほど「ファイル ストア ノード ロール セクション」で詳しく説明します。

バックグラウンダー ノード ロール

バックグラウンダー プロセスでは、抽出の更新、サブスクリプション、フロー タスク、「今すく実行」タスク、*tabcmd* から開始したタスクなどの Tableau Server タスクを実行します。これらのタスクをすべて実行すると、多くのマシン リソースが使用される可能性があります。クラスタ内に複数のバックグラウンダー ノードがある場合は、バックグラウンダー ノード ロール機能を使用してバックグラウンダーが

ノードで実行できるタスクのタイプを指定することにより、バックグラウンダー ワークロードを管理できます。

現在、この構成オプションは TSM CLI コマンド経由でのみ使用でき、マルチノード クラスタでのみ有用です。ノードが 1 つしかない場合、バックグラウンダーは既定ですべてのタスクを実行するように設定され、これは変更できません。

バックグラウンダー ノード ロールの使用

バックグラウンダー ノード ロール機能は、Tableau Server インストールで特定のタイプのバックグラウンダー ワークロードが処理される場所をより細かく制御して、リソースを特定のワークロード専用にしたりワークロードに合わせて拡張できるようにすることを意図しています。

たとえば、展開で抽出が大量に実行され、ユーザーが多くの抽出更新や暗号化ジョブを実行している場合は、あるノードを抽出更新専用にすると便利です。サブスクリプションの場合も同様で、Tableau Server インストールで大量のサブスクリプションが処理される場合、他のジョブがサブスクリプション用のリソースを取得しないようにするには、あるノードをサブスクリプション専用にすることができます。これらの場合は、他のバックグラウンダー ノードを抽出更新やサブスクリプション以外のワークロード専用にすることもできます。

高可用性をサポートするため、複数のノードをそれぞれ特定のワークロード専用にすることをお勧めします。たとえば、あるノードを抽出更新専用にする場合は、抽出更新ワークロードを処理する 2 番目のノードも構成するとよいでしょう。このようにすると、抽出更新専用のノードが使用できなくなった場合でも、抽出更新を他のノードで引き続き処理できます。

構成オプション

構成	ジョブ
all-jobs (既定値)	すべての Tableau Server ジョブ。
flows	フロー実行ジョブ。
no-flows	フロー以外のすべてのジョブ。
extract-	次の処理用に作成されたジョブ。

refreshes	増分更新、完全更新、フロー出力が作成する抽出を含むすべての抽出の暗号化と復号化。
subscriptions	サブスクリプション ジョブ。
system	他の Tableau Server プロセスと連携するシステム メンテナンス ジョブ。たとえば、クラッシュしたジョブのクリーニング、データベース イベントの取得、Active Directory の同期などがあります。
extract-refreshes-and-subscriptions	抽出更新、フロー出力が生成する抽出を含むすべての抽出の暗号化と復号化、およびサブスクリプション ジョブ。
no-extract-refreshes	抽出更新、フロー出力から作成された抽出を含むすべての抽出の暗号化と復号化以外のすべてのジョブ。
no-subscriptions	サブスクリプション以外のすべてのジョブ。
no-extract-refreshes-and-subscriptions	抽出更新、フロー出力から作成された抽出を含むすべての抽出の暗号化と復号化、サブスクリプション以外のすべてのジョブ。
no system	システム メンテナンス ジョブを除くすべてのジョブ。

tsm コマンドを使用してノード ロールを設定する方法の詳細については、tsm topologyを参照してください。

注: ノードの役割を構成するには、サーバーを再起動する必要があり、いくらかのダウンタイムが必要になります。詳細については、tsm pending-changesを参照してください。

ライセンスの要件

フロー、抽出更新、サブスクリプションなどの特定のタイプのタスクのみを実行するようにノードを構成するには、Tableau Server で次のいずれかのライセンスがアクティブ化されている必要があります。

- フローを実行するようにノードを構成するには、有効なデータ管理 ライセンスがサーバーで有効になっており、Tableau Prep Conductor がそのノード上で実行されている必要があります。Tableau Prep Conductor の詳細については、Tableau Prep Conductorを参照してください。

- 抽出更新、サブスクリプション、および抽出やサブスクリプションに関連するあらゆる組み合わせを実行するようにノードを設定するには、Tableau Server 上で Advanced Management 機能を有効化している必要があります。ライセンスの有効期限が切れた場合、または非アクティブになっている場合は、サーバー構成に変更を加えるたびにエラーが表示されます。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

重要!
フロー、抽出更新、サブスクリプションはコストがかかり、リソースを大量に消費する可能性がありますが、専用のリソースが必要なジョブはこれだけではありません。**all-jobs** グループには、ワークブックのサムネイル生成など、バックグラウンダーが実行するさまざまなシステム ジョブがあります。抽出更新、サブスクリプション、フロー以外のジョブを実行するノードに十分なマシン リソースがあることを確認してください。

TSM コマンドを使用したノード ロールの構成の詳細については、tsm topology set-node-roleを参照してください。

考慮事項

バックグラウンダー ノード ロールを構成する際は、以下のようないくつかの規則を考慮する必要があります。

- ノードに対して一度に設定できるノード ロール構成は 1 つだけです。1 つのノードで複数のノード ロールを構成することはできません。

- ノード ロールを構成するには、そのノードに少なくとも 1 つのバックグラウンダー プロセスが存在する必要があります。

- バックグラウンダー ノードが 1 つしかない場合は、このノードによってすべてのジョブを実行するように構成する必要があります。これは既定の構成であり、追加のライセンスは必要ありま

せん。

- 複数のバックグラウンダー ノードがある場合は、その組み合わせによってすべてのジョブを処理するように構成する必要があります。これは、次の方法で実現できます。

    - all-jobs オプションを使用して、すべてのジョブを実行するようにいずれかのノードを構成します。これは最も簡単な方法です。

    - いずれかのノードで以下のいずれかの例外構成を使用します。

        - no-flows
        - no-subscriptions
        - no-extract-refreshes
        - no-extract-refreshes-and-subscriptions

たとえば、3 つのバックグラウンダーがあるクラスタでは、1 台のノードをフローの実行用に構成し、1 台のノードをサブスクリプションと抽出更新の実行用に構成し、1 台のノードをフロー、サブスクリプション、抽出更新を除くすべてのジョブの実行用に構成できます。

注: フロー、フロー以外のすべてのジョブ、またはすべてのジョブの実行用のノード ロールを指定する機能は、2019.1 で導入されました。

ファイル ストア ノード ロール

Tableau Server ファイル ストアは、抽出のストレージを管理します。抽出に依存するワークロードには、大きく分けて 3 つのカテゴリーがあります。

抽出のワークロード	実行サービス
更新	バックグラウンダー
クエリ	データ エンジン
バックアップ/復元	バックアップ/復元

ファイル ストア ノード ロールの管理とバックグラウンダー ノード ロールの管理を組み合わせると、サーバー管理者はサーバー ノードを、選択したワークロードを優先的に実行し、抽出負荷の大きいすべてのカテゴリーのワークロードでパフォーマンスを最適化する専用のノードとすることができます。

スタンドアロンのデータエンジン ノードのみを持つトポロジを使用し、抽出クエリワークロードを実行する専用のノードとすることができます。詳細については、抽出クエリの負荷が大きい環境用に最適化を参照してください。とはいえ、これはバックグラウンダー ノードによって実行される抽出更新ワークロードを犠牲にして成り立つものです。トポロジベースの分離アプローチを使用すると、バックグラウンダー ノードのいずれもファイル ストアがなく、すべての抽出更新トラフィックがネットワーク経由で流れるため、抽出更新の負荷が大きいバックグラウンダー ワークロードは遅くなる可能性があります。

ファイル ストア ノード ロールの構成オプションを使用すると、抽出クエリを処理する特定のサーバー ノードを、それが可能なサーバー ノードのリストから優先的に選択するように指定できます。これは、サーバー管理者がバックグラウンダー サーバー ノード上でファイル ストアを有効にするのを許可し、これらのノードで抽出クエリを実行できないようにすることで、バックアップや抽出更新などのワークロードを高速化するのに役立ちます。この機能は、抽出負荷の大きいクエリワークロードと更新ワークロードがあり、抽出クエリと更新パフォーマンスを最適化する場合に便利です。

抽出更新ワークロードとバックアップまたは復元ワークロードを最適化するためのガイドライン

専用のデータエンジン ノードを含むトポロジから開始します 抽出クエリの負荷が大きい環境用に最適化を参照)。

注: 以下の図と手順では、ノード1は初期ノード、ノード2は追加ノード1、ノード3は追加ノード2、ノード4は追加ノード3です。

Process	Initial Node	Additional Node 1	Additional Node 2	Additional Node 3
Cluster Controller	✓	✓	✓	✓
Gateway	✓	✓		
Application Server	✓	✓		
VizQL Server	✓ ✓	✓ ✓		
Cache Server	✓ ✓	✓ ✓		
Search & Browse	✓	✓		
Backgrounder	✓ ✓ ✓ ✓	✓ ✓ ✓ ✓		
Data Server	✓ ✓	✓ ✓		
Data Engine	✓	✓	✓	✓
File Store			✓	✓
Repository	✓	✓		

トポロジ 1 - 専用データエンジン ノード

1. ファイル ストアをノード1 に追加します。

   ```
   tsm topology set-process -n node1 -pr filestore -c 1
   ```

2. 抽出クエリのワークロードを優先的に実行するようにノード3 とノード4 を指定します。

   ```
   tsm topology set-node-role -n node3, node4 -r extract-queries
   ```

3. 非抽出更新のワークロードを優先的に実行するようにノード1 を指定します。

   ```
   tsm topology set-node-role -n node1 -r extract-refreshes
   ```

4. 非抽出更新のワークロードを優先的に実行するようにノード2 を指定します。

   ```
   tsm topology set-node-role -n node2 -r no-extract-refreshes
   ```

5. 保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

Process	Initial Node	Additional Node 1	Additional Node 2	Additional Node 3
Cluster Controller	✓	✓	✓	✓
Gateway	✓	✓		
Application Server	✓	✓		
VizQL Server	✓ ✓	✓ ✓		
Cache Server	✓ ✓	✓ ✓		
Search & Browse	✓	✓		
Backgrounder	✓ ✓ ✓ ✓	✓ ✓ ✓ ✓		
Data Server	✓ ✓	✓ ✓		
Data Engine	✓	✓	✓	✓
File Store	✓		✓	✓
Repository	✓	✓		

トポロジ 2 - 余分のファイル ストア ノード

注: Tableau Server 展開では、既存ノードにファイル ストア ロールを追加すると、新しいファイル ストアを同期しながら、すべてのファイル ストア ノード間のネットワーク I/O が一時的に増加します。この動作期間は、ファイル ストア上のデータ量とネットワーク帯域幅の容量によって変わります。同期のステータスは、TSM Web GUI を使用して監視できます。展開に複数のファイル ストアを追加する場合、それを連続で追加し、各ファイル ストアの追加の間に最初の同期が完了するのを待機することをお勧めします。

抽出クエリワークロード管理の微調整

ユーザーが抽出ベースの Viz をインタラクティブに表示しているときに、メール登録とメトリクス アラートの抽出クエリがそれと同時に実行されると、Viz の読み込み時間が通常よりも遅くなることがあります。次のノード ロールを使用して、これらのワークロードの優先順位を微調整します。

使用するノード ロール	抽出クエリ ワークロードのタイプ	例
`extract-queries`	スケジュール済み	メール登録とメトリクス アラート
`extract-queries-interactive`	インタラクティブ	抽出ベースの Viz を表示するユーザー

サーバーの展開でメール登録とメトリクスアラートの増加が確認される場合は、ノードを追加して extract-queries ノードロールを割り当てることができます。そうすると、メール登録とメトリクスアラートを処理するためのノードの可用性が増大します。

サーバーの展開で抽出ベースの Viz を表示するユーザーの増加が確認される場合は、ノードを追加して extract-queries-interactive ノードロールを割り当てることができます。そうすると、ノードでインタラクティブな抽出クエリに優先順位を付けて、抽出ベースの Viz の読み込み時間を短縮できます。extract-queries-interactive ノードロールは、厳密に分離されるものではなく、優先されるロールです。つまり、クエリは extract-queries-interactive ノードロールが割り当てられているノードにルーティングされます。extract-queries-interactive ロールを持つ複数のノードがある場合、クエリはノードの正常性に基づいてルーティングされます。

たとえば、ノードを追加し、extract-queries-interactive ワークロードを優先的に実行するように指定します。

- `tsm topology set-node-role -n node4 -r extract-queries-interactive`

構成オプション

構成	ジョブ
all-jobs (既定値)	すべての Tableau Server ジョブ。
extract-queries	抽出クエリ用に作成されるジョブ。選択されたノードは all-jobs として実行され、抽出クエリの処理を優先して実行します。
extract-queries-interactive	抽出クエリ用に作成されるジョブ。選択されたノードは all-jobs として実行され、インタラクティブな抽出クエリの処理を優先して実行します。これらのクエリは、ユーザーが画面を見ていて、抽出ベースのダッシュボードの読み込みを待機しているときに実行されるクエリのようなものです。これは高度な設定であるため、クラスターのサブスクリプションおよびアラートジョブのワークロードの負荷が大きく、スケジュールされたロードと同じ時間に実行される Viz の読み込み時間のパフォーマンスが低下する場合にのみ使用してください。

TSM コマンドを使用したノードロールの構成の詳細については、tsm topology set-node-roleを参照してください。

ライセンスの要件

ノードを抽出クエリの実行用に構成するには、有効な Advanced Management ライセンスが Tableau Server でアクティブ化されている必要があります。

ノードロールの表示方法

次のコマンドを使用して、Tableau Server で現在構成されているノードロールを表示します。

```
tsm topology list-nodes -v
```

実行可能なユーザー

Tableau Server 管理者は、ノードロールを構成し、必要なプロダクトキーを認証することができます。

2 ノードクラスターに Tableau Server をインストールする

2 ノードクラスターで Tableau Server をインストールする場合、1 つまたは両方のノードでサーバープロセスをインストールする必要があります。2 ノードクラスターは複数のマシン上で作業を実行するため、Tableau Server のパフォーマンスを向上させることができます。

2 ノードクラスターについては、次に注意してください。

- 2 ノードクラスターは、フェールオーバーの提供や高可用性のサポートは行いません。
- 2 ノードクラスタにリポジトリの 1 つ以上のインスタンスをインストールすることはできません。また、リポジトリは最初のノード上にある必要があります。

フェイルオーバーや高可用性が必要な場合、またはリポジトリに 2 つ目のインスタンスが必要な場合は、少なくとも 3 台のコンピューターに Tableau Server をインストールする必要があります。3 ノード以上を含むクラスターで、リポジトリの 2 つのインスタンスを設定し、クラスター フェールオーバー機能を利用できます。

複数ノードTableau Server コンピューターの再起動

Tableau Server の分散インストールを実行しているコンピューターを再起動するには、いくつかの特定のステップに従う必要があります。

Tableau Server の分散インストールを実行しているコンピューターを再起動するには:

1. Tableau Server を停止します。これを実行する手順は以下のとおりです。
    a. 最初のノードで、ターミナル セッションを開きます。
    b. 以下の tsm コマンドを実行します。

        ```
        tsm stop
        ```

    c. Tableau Server が停止するまで待機します。
2. 追加ノードのコンピューターを再起動します。これらは、TSM コントローラを実行しているコンピューターを除く他のすべてのコンピューターです。
3. 各追加ノードが完全に再起動するまで待機します (そこにインストールされている Tableau サービス マネージャー (TSM) も含む)。
4. 第一ノードのコンピューター (TSM コントローラを実行しているコンピューター) を再始動します。ノードが完全に再起動され、Tableau Serverが実行されると、各追加ノードへの接続が復元されます。

分散環境の管理

最初のノードと分散型のインストール環境用に追加する1 台以上のノードを設定すると、TSM CLI を使用して最初のノードから、またはブラウザーおよび TSM Web インターフェイスを使用してすべてのコンピューターから以降のすべての構成および更新を実行できるようになります。

追加ノードをインストールする場合、コンピューター名で追加されます。ノードのコンピューター名が変更になる場合は、ノードを削除して再インストールする必要があります。ノード削除の詳細については、ノードの削除を参照してください。

Tableau Server クラスタのステータスは、[TSM ステータス] ページで監視できます。詳細については、サーバープロセス ステータスの表示を参照してください。

分散環境を維持するために必要となる可能性のある追加のアクションは、次のとおりです。

リポジトリプロセスの移動

Tableau Server は PostgreSQL リポジトリに依存してサーバー データを保存しています。Tableau Server インストールにはリポジトリの有効なインスタンスが常に少なくとも 1 つ必要で、インストールに少なくとも 3 つのノードがある場合は、最大で 2 つのインスタンス(1 つアクティブ、1 つパッシブ) を使用できます。唯一のインスタンスの場合は、リポジトリインスタンスを削除できません。

つまり、リポジトリの唯一のインスタンスを 1 つのノードから他のノードに移動したい場合、2 番目のインスタンスを追加して新しいリポジトリと古いリポジトリを同期してから、古いリポジトリを削除する必要があります。サーバーを起動してリポジトリインスタンスを同期します。新しいリポジトリを追加したら、既存のインスタンスと自動的に同期します。

サーバークラスタからノードを削除するときにそのノードがリポジトリの唯一のインスタンスをホストしている場合は、ノードを削除する前に、リポジトリの 2 番目のインスタンスを追加してそのインスタンスを同期する必要があります。

また、ファイル ストアを移動する場合は、同時にリポジトリも移動できます。ファイル ストアのプロセスの移動を参照してください。

リポジトリに変更を加える前に、Tableau Server のフル バックアップを作成します。詳細については、tsm maintenance backupを参照してください。

重要: 同じステップで、2 番目のリポジトリインスタンスの追加と、1 番目のインスタンスの削除はできません。両方を実行させて、元のインスタンスを削除する前に、1 番目のインスタンスのコンテンツを 2 番目と同期させる必要があります。

リポジトリの削除ステップは以下の通りです。

1. リポジトリの新しいインスタンスを別のノードに追加し、サーバーを起動し、1 番目のリポジトリと同期するのを待ちます。
2. リポジトリのインスタンスを元のノードから削除します。

TSM Web インターフェイスの使用

リポジトリを移動するには、まず 2 番目のインスタンスを 2 番目のノードに追加し、次に 2 つのインスタンスが元のリポジトリにあるデータ全てを同期した後で、元のインスタンスを削除します。上記ステップは、2 つのインスタンス間でのコンテンツ同期を可能にするために別々に行う必要があります。

リポジトリの新しいインスタンスの追加

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックします。
3. リポジトリに追加するノードでは以下を行います。

   リポジトリ(pgsql)を選択します。

4. ページ上部の **[変更を保留中]** をクリックします。

[変更を保留中] のリストが表示されます。

3 または 5 ノードのクラスタを構成していて、調整サービス アンサンブルをまだ展開していない場合、警告が表示されます。継続して、調整サービス アンサンブルを別のステップで展開します。新しい調整サービス アンサンブルを展開する方法についての詳細は、調整サービス アンサンブルの展開を参照してください。

5. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。
6. Tableau Server が起動したら、**ステータス** タブで、すべてのプロセスが有効であることを検証します。

リポジトリのインスタンスの削除

1. TSM の **ステータス** タブで、すべてのプロセスが有効であることを検証します。両方のリポジトリが有効として表示されたら、1 番目を削除できます。
2. **[構成]** タブをクリックします。
3. リポジトリを削除しようとしているノードで、**リポジトリ ボックス**をクリアします。
4. ページの上部にある**変更を保留中**をクリックします。

   3 または 5 ノードのクラスタを構成していて、調整サービス アンサンブルをまだ展開していない場合、警告が表示されます。継続して、調整サービス アンサンブルを別のステップで展開します。新しい調整サービス アンサンブルを展開する方法についての詳細は、調整サービス アンサンブルの展開を参照してください。

5. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

TSM CLI の使用

リポジトリを移動するには、まず 2 番目のインスタンスを 2 番目のノードに追加し、次に 2 つのインスタンスが元のリポジトリにあるデータ全てを同期した後で、元のインスタンスを削除します。上記ス

テップは、2 つのインスタンス間でのコンテンツ同期を可能にするために別々に行う必要があります。

リポジトリの新しいインスタンスの追加

1. リポジトリ(pgsql) を別のノードに追加します。

   ```
   tsm topology set-process -n <nodeID> -pr pgsql -c 1
   ```

2. 変更を適用します。保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

   ```
   tsm pending-changes apply
   ```

3. 2 番目のノードの新しいリポジトリが最初のノードのリポジトリと同期するまで待機します。

   ```
   tsm status -v
   ```

   新しいリポジトリのステータスが [パッシブ] として表示されるまで待機します。

リポジトリのインスタンスの削除

リポジトリの新しいインスタンスが完全に同期され、[パッシブ] として表示されたら、元のインスタンスを削除できます。

1. プロセス カウントを 0 (ゼロ) に設定して、1 番目のノードからリポジトリを削除します。

   ```
   tsm topology set-process -n <nodeID> -pr pgsql -c 0
   ```

2. 変更を適用します。保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されな

いようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

```
tsm pending-changes apply
```

ファイル ストアのプロセスの移動

このトピックでは、Tableau Server でローカルに実行されるようにファイル ストア プロセスを構成した場合について説明します。ただし、ファイル ストアはローカルだけでなくTableau Server の外部でも実行できます。Tableau Server 外部ファイル ストアの詳細については、Tableau Server 外部ファイル ストアを参照してください。

ファイル ストアのインスタンスがサーバーの唯一のインスタンスの場合は、これを削除できません。ファイル ストアの少なくとも 1 つのインスタンスがないとTableau Server を実行できません。つまり、ファイル ストアを移動する必要がある場合、またはファイル ストアの唯一のインスタンスをホストするサーバー ノードを削除する場合は、最初にファイル ストアを別のノードに移動させる必要があります。

ファイル ストアの移動は、次の 2 部プロセスで行います。

- ファイル ストアの 2 番目のインスタンスを追加する (2 番目のインスタンスがまだない場合)。
- ファイル ストアのオリジナル インスタンスをコミッション解除および削除する

この記事では Tableau Server が初期ノードおよび少なくとも一つの追加ノードにインストールされていることを前提としています。ノードの Tableau Server に追加する方法の詳細については、追加のノードのインストールと構成を参照してください。

TSM Web インターフェイスの使用

ファイル ストアの 2 番目のインスタンスを追加する

この手順は追加のノードを追加済であることを前提としています。ノードの Tableau Server に追加する方法の詳細については、追加のノードのインストールと構成を参照してください。

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850

詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックします。

3. ファイル ストアを追加したいノードで、**ファイル ストア**を選択します。

   ファイル ストアをノードに追加すると、まだノードにない場合、データエンジンのインスタンスも追加されます。

4. ページ上部の **[変更を保留中]** をクリックします。

   [変更を保留中] のリストが表示されます。

   3 ノード クラスタを構成しているので、調整サービス アンサンブル警告が表示されます。続行できます。次のステップで、調整サービス アンサンブルを展開します。

5. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

ファイル ストアのインスタンスをコミッション解除および削除する

1. TSM の **ステータス** タブで、すべてのプロセスが有効であることを検証します。

2. **[構成]** タブをクリックします。

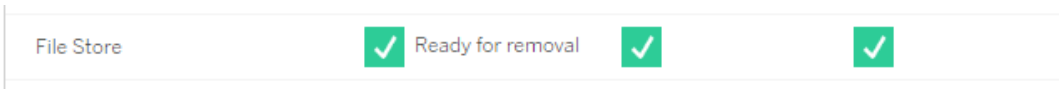
3. ファイル ストアを削除しようとしているノードで、**ファイル ストア ボックス**をクリアします。

   これにより、削除する前に、ファイル ストアのコミッション解除についての警告が生成されます。ファイル ストアのインスタンスは、コミッション解除されていない限り、削除できません。**OK** をクリックして使用停止し、ファイル ストアを削除します。これらのプロセスのうちいずれかのインスタンスがノード (VizQL Server、アプリケーション サーバー (Vizportal)、データ サーバー、バックグラウンダー) にインストールされていない場合は、データ エンジンも削除されます。

4. TSM 管理コントローラー(通常は初期ノード)もあるノードからファイル ストアを削除する場合、サーバーのバックアップ パフォーマンスに対する影響について警告が表示されます。

   [続行] をクリックし、ファイル ストアをコミッション解除します。

5. **ステータス** タブをクリックして、コミッション解除のステータスを確認します。ファイル ストアのインスタンスが「削除準備完了」とマークされていれば、続行できます。

6. ページの上部にある**変更を保留中**をクリックします。

   注:3 または 5 ノードのクラスタを構成している場合、調整サービス アンサンブル警告が表示されます。保留中の変更の適用を継続して、調整サービス アンサンブルを別のステップで展開します。新しい調整サービス アンサンブルを展開する方法についての詳細は、調整サービス アンサンブルの展開を参照してください。

7. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

TSM CLI の使用

ファイル ストアの 2 番目のインスタンスを追加する

1. Tableau Server の完全なバックアップを作成します。詳細については、Tableau Server データのバックアップを参照してください。

2. 2 番目のノードにファイル ストアを追加します。

   ```
   tsm topology set-process -n <nodeID> -pr filestore -c 1
   ```

   ファイル ストアが自動的に追加されます。データ エンジンがノードにまだない場合は、一緒に追加されます。

   構成の変更を次のように適用します。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

3. 新しいファイル ストア インスタンスの状態を確認します。

   ```
   tsm status -v
   ```

   最初のファイル ストアと新しいノードのファイル ストアが同期するまで待機します。同期が完了すると、新しいファイル ストアのステータスが [同期中] ではなく[実行中] になります。

ファイル ストアのインスタンスをコミッション解除および削除する

ファイル ストアの 2 番目のインスタンスをインストールおよび同期したら、元のインスタンスをコミッション解除および削除できます。元のインスタンスは削除前にコミッション解除する必要があります。これにより、ファイル ストア ノード上の一意のファイルが別のファイル ストア ノードに複製されます。

1. 元のファイル ストアを使用停止にします。

   ```
   tsm topology filestore decommission -n <nodeID> --override
   ```

2. decommission コマンドが完了したら、保留中の構成変更を適用してノードからファイル ストアを削除します。ファイル ストアが自動的に削除されます。これらのプロセスのうちいずれかのインスタンスがノード (VizQL Server、アプリケーション サーバー (Vizportal)、データ サーバー、バックグラウンダー) にインストールされていない場合は、データ エンジンも削除されます。

   ```
   tsm pending-changes apply
   ```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

メッセージング サービス プロセスの移動

この記事では Tableau Server が初期ノードおよび少なくとも一つの追加ノードにインストールされていることを前提としています。ノードの Tableau Server に追加する方法の詳細については、追加のノードのインストールと構成を参照してください。

メッセージング サービスがサーバーの唯一のインスタンスの場合は、これを削除できません。Tableau Server はメッセージング サービスのインスタンスが 1 つないと実行できません。つまり、メッセージング サービスを移動する必要がある場合、またはメッセージング サービスの をホストしているサーバーノードを削除する場合は、最初にサービスを別のノードに移動させる必要があります。

メッセージング サービスの移動は、3 つの手順からなる簡単なプロセスです。

- メッセージング サービスの元のインスタンスを削除します。
- メッセージング サービスの新しいインスタンスを追加します。
- 保留中の変更を適用します。

TSM Web インターフェイスの使用

メッセージング サービスの の移動

1. TSM の **ステータス** タブで、すべてのプロセスが有効であることを検証します。
2. **[構成]** タブをクリックします。

3. メッセージング サービスを削除しようとしているノードで、**メッセージング サービス** ボックスをクリアします。

   これにより、**[保留中の変更]** ボタンがアクティブになりますが、[保留中の変更] でエラーが表示され、メッセージング サービス (activemqserver) がどのノードにも存在しないことが示されます。別のノードに追加するまで、保留中の変更を適用することはできません。

4. メッセージング サービスを追加するノードの **[メッセージングサービス]** ボックスを選択します。

5. ページの上部にある**変更を保留中**をクリックします。

6. **[変更を適用して再起動]**、**[確認]** をクリックし、Tableau Server の再起動を確認します。

TSM CLI の使用

メッセージング サービスの の移動

1. 最初のノードで、ターミナル セッションを開きます。

2. 変更しようとしているノードのノード ID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. 1 つのノードからメッセージング サービスを削除します。

   ```
   tsm topology set-process -n <nodeID> -pr activemqserver -c 0
   ```

4. 別のノードにメッセージング サービスを追加します。

   ```
   tsm topology set-process -n <nodeID> -pr activemqserver -c 1
   ```

5. 構成の変更を次のように適用します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--

`ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

ノードの削除

Tableau Server のインストールにもう必要ないノードが含まれている場合、それを削除することでインストールを簡素化するとともにそのノード上のハードウェア リソースを解放することができます。

ノードを削除するための前提条件

Tableau Server クラスタからノードを削除する前に満たす必要がある前提条件があります。ノードに多数ある制限のうちいずれかが該当する場合には、ノードの削除前にそれに対処する必要があります。制限としては、ノードが少なくとも 1 つのプロセスで追加されていることの確認、およびノードに別のノードにもインストールされていないプロセスが含まれていないことの確認が含まれます。

構成制限のいずれかが該当する場合には、ノードを削除する前に対応を取る必要があります。

- ノードが Web UI を使用して追加されたばかりの場合は、削除する前に保留中の変更を適用する必要があります。コマンド ラインを使用して追加した場合は、削除する前に少なくとも 1 つのプロセスで構成する必要があります。
- ノードにリポジトリ インスタンスのみが含まれている場合は、リポジトリを別のノードに移動する必要があります。リポジトリプロセスの移動を参照してください。
- ノードでファイル ストアのインスタンスのみが実行されている場合は、ファイル ストアを別のノードに移動する必要があります。ファイル ストアのプロセスの移動を参照してください。
- ノードが調整サービスのインスタンスを実行している場合は、ノードを含まない新しい調整サービス アンサンブルを展開する必要があります。調整サービス アンサンブルの展開を参照してください。
- ノードでメッセージング サービスの が実行されている場合は、メッセージング サービスを別のノードに移動する必要があります。メッセージング サービス プロセスの移動を参照してください。

重要: `tableau-server-obliterate` スクリプトを使用してノードを削除しないでください。まず、TSM の UI または tsm topology remove-nodes コマンドを使用してノードを削除します。このようにすると、クラスタの残りは良い状態で保たれます。後でノードをクラスタにもう一度追加する必要がある場合は、ノード上で `tableau-server-obliterate` スクリプトを実行して Tableau を完全に削除しますノードからTableau を削除した後、コンピューターを再起動してから、ノードを追加および構成する通常のステップで、ノードを再び追加します。スクリプト実行の詳細については、コンピュータからTableau Server を削除するを参照してください。

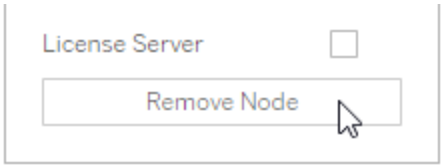
TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブをクリックし、削除したいノードの下で **[ノードの削除]** をクリックします。



   構成の制限によりノードを削除できない場合 (たとえば、最初にコミッション解除する必要のあるファイル ストアが含まれている場合)、そのことを知らせるメッセージが表示されます。詳細については、上記の**前提条件**を参照してください。

   > **注:** ノードを追加しただけで構成していない場合、ノードを削除する前に保留中の変更を適用する必要があります。

3. 右上の **[変更を保留中]** をクリックし、**[変更を適用して再起動]** をクリックします。

TSM CLI の使用

`tsm topology remove-nodes` コマンドを使用して、クラスターからノードを削除します。

クラスタからノードを削除する際は、過去のどこかの時点でそのノード上にプロセスが構成されていたはずです。CLI を使用して追加した後でプロセスを構成していないノードを削除する場合は、ノードにプロセスを追加し、`tsm pending-changes apply` コマンドを実行してから、ノードを削除する必要があります。たとえば、ノードに 1 インスタンスのクラスター コントローラーを追加する場合があります: `tsm topology set-process -n <nodeID> -pr clustercontroller -c 1`.

Web UI を使用して追加したノードを削除する場合、クラスタ コントローラー プロセスは自動的に追加されるため、ノードを削除する前に追加する必要はありません。

1. 最初のノードで、ターミナル セッションを開きます。
2. 変更しようとしているノードのノード ID を確認します。

   ```
   tsm topology list-nodes -v
   ```

3. remove-nodes コマンドを使用してノードを削除します。

   たとえば、以下の方法で既存のクラスターから `node2` を削除します。

   ```
   tsm topology remove-nodes --node-names "node2"
   ```

   構成の制限によりノードを削除できない場合 (たとえば、最初にコミッション解除する必要のあるファイル ストアが含まれている場合)、そのことを知らせるメッセージが表示されます。詳細については、上記の**前提条件**を参照してください。

4. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

調整サービス専用ノードで高可用性用を Tableau Server に設定する

調整サービス アンサンブルはオープン ソース プロジェクトである Apache ZooKeeper 上に構築され、サーバー上の活動を調整し、障害発生時にはクォーラムを保証し、サーバー トポロジー、校

正、状態に関する「真実」の参照元としての役割を担います。このサービスは最初の Tableau Server ノードに自動的にインストールされますが、他のノードを追加しても追加のインスタンスはインストールされません。Tableau Server が順調に機能を発揮するには、調整サービスが適切に機能している必要があるため、3 台以上のサーバー インストール環境では、新しい調整サービス アンサンブルを展開することで調整サービスの追加のインスタンスを加えることを推奨します。これにより、調整サービスのインスタンスの 1 つに問題が発生した場合に冗長性および向上された可用性が提供されます。

調整サービスはサーバーの他のコンポーネントと通信すつため大量の I/O を生成する場合があります。そのため、最小ハードウェア要件を満たすかわずかに上回るコンピューターで Tableau Server を実行している場合、調整サービス専用ノードを使用する構成での Tableau Server のインストールが必要になる場合があります。つまり、他のサーバープロセスを実行していないノードに調整サービスをインストールし、他のサーバープロセスを実行中のノードから調整サービスを削除します。この手順はこの方法について説明します。調整サービス アンサンブルを他の Tableau Server プロセスを実行している同じノードで実行することも可能です。詳細は、調整サービス アンサンブルの展開を参照してください。

前提条件

このトピックの手順を進める前に、以下の前提条件を確認してください。

- Tableau Server のインストールと構成 - Tableau を初期ノードにインストール。
- 追加のノードのインストールと構成 - Tableau を少なくとも 2 つの追加のノードにインストール。

注: この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

調整サービス専用ノードでアンサンブルを展開する

調整サービスの高い I/O 不可に対応する方法の 1 つとして、アンサンブルを調整サービスとクラスタ コントローラーのみを実行するノード上に展開する方法があります。以下の手順は、既存のマルチノード Tableau Server クラスタ上に調整サービス アンサンブルを展開する方法を説明しています。

注: コアベースの Tableau Server ライセンスでは、調整サービス専用ノードは、ライセンスを持つコアの合計数に対してカウントされません。

1. 追加のノードをクラスタに追加します。

   追加のノードのインストールと構成を参照してください。

2. 新しノードを TSM CLI を使用して追加した場合、それらのノードをクラスター コントローラーで構成する必要があります(このステップは、TSM Web UI を使用してノードを追加した場合は不要です。なぜならノードを Web UI で追加する時にクラスター コントローラーは自動的に追加されるからです) 。

   最初のノードで、ターミナル セッションを開きます。

3. クラスタの最初のノードから、一連の新しいノードにクラスタ コントローラーのインスタンスが実行されるように構成します。

   ```
   tsm topology set-process -pr clustercontroller -n <node4> -c 1

   tsm topology set-process -pr clustercontroller -n <node5> -c 1

   tsm topology set-process -pr clustercontroller -n <node6> -c 1
   ```

   メッセージが表示されたら、TSM 管理者としてサインインします。

4. 構成の変更を適用します。保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

   ```
   tsm pending-changes apply
   ```

マルチ ノード クラスタを展開しているため、調整サービス アンサンブルを展開しているという警告が表示されます。この警告しか表示されていないのであれば、警告に関わらず、--ignore-warnings オプションを使用して安全に上書きし、構成を変更できます。

```
tsm pending-changes apply --ignore-warnings
```

5. すべてのノードが起動され実行中であることを確認します。

   ```
   tsm status -v
   ```

6. クラスタの最初のノードでターミナル セッションを開き、次のコマンドを入力して Tableau Server を停止します。

   ```
   tsm stop
   ```

7. クラスタ内の各ノードのノード ID を取得します。

   ```
   tsm topology list-nodes -v
   ```

8. tsm topology deploy-coordination-service コマンドを使用して調整サービスを指定されたノードに追加することで、新しい調整サービス アンサンブルを追加します。調整サービスを追加する先のノードを指定しなくてはなりません。このコマンドにより、新しいアンサンブルが「本番」アンサンブル (使用中のアンサンブル) になり、古いアンサンブルが削除されます。

   注: サーバーの再起動が行われることを確認する「y/n (はい / いいえ)」のプロンプトが表示されます。入力せずにコマンドを実行するには、--ignore-prompt オプションを含めます。

   たとえば、調整サービスを 6 ノード クラスタの 3 台のノードに展開します。

   ```
   tsm topology deploy-coordination-service -n <node4,node5,node6>
   ```

   コマンドが完了し、システム プロンプトに戻るまで待ちます。

9. Tableau Server を起動します。

```
tsm start
```

ロードバランサーの追加

Tableau Server の信頼性を高めるには、複数のノードでゲートウェイを実行し、ロードバランサーを構成してゲートウェイ間で要求を分散します。アクティブまたはパッシブであるリポジトリプロセスとは異なり、すべてのゲートウェイプロセスはアクティブです。クラスタの 1 台のゲートウェイが使用できなくなると、ロードバランサーによりゲートウェイへの要求送信が停止します。選択したロードバランサーアルゴリズムによって、ゲートウェイがクライアント要求を送る方法が決まります。

- **Kerberos**: Kerberos 認証を使用する場合、Kerberos 用に Tableau Server を構成する前に、ロードバランサー用に Tableau Server を構成する必要があります。詳細については、Kerberos の構成を参照してください。
- **テストされたロードバランサー**: 複数のゲートウェイを使用する Tableau Server クラスタは、Apache および F5 ロードバランサーでテストされました。

  Apache ロードバランサーを使用してカスタム管理ビューを作成している場合、Tableau Server リポジトリに直接接続する必要があります。ロードバランサーを介して接続することはできません。
- **Tableau Server URL**: ロードバランサーが Tableau Server クラスタの前面にある場合、Tableau Server ユーザーがアクセスする URL は、初期 Tableau Server ノードではなくロードバランサーに属します。
- **単一ロードバランサー エンドポイント**: 単一の URL エンドポイントに対してロードバランサーを構成する必要があります。同じ Tableau Server の展開にリダイレクトするように、異なるエンドポイントホストを構成することはできません。Tableau Server を構成するときに、1 つの外部 URL が `gateway.public.host` で定義されます。詳細については、Tableau Server のプロキシとロードバランサーの設定を参照してください。
- **信頼できるホストの設定**: ロードバランサーを実行しているコンピューターは、信頼できるホストとして Tableau Server と認識される必要があります。詳細については、Tableau Server のプロキシとロードバランサーの設定を参照してください。

ロードバランサーを使用するための Tableau Server の構成

ロードバランサーを Tableau Server と識別するために使用する設定は、リバースプロキシ サーバーを識別するために使用する設定と同じです。Tableau Server クラスタにプロキシ サーバーとロードバランサーの両方が必要な場合、どちらも `gateway.public.host` に定義された単一の外部 URL を使用する必要があり、すべてのプロキシ サーバーとロードバランサーが `gateway.trusted` および `gateway.trusted_hosts` に指定されている必要があります。Tableau Server のプロキシとロードバランサーの設定を参照してください。

Tableau Server のアップグレードの概要

このセクションの記事は、Tableau Server on Linux の既存インストールをアップグレードする際に役立ちます。実際にアップグレードを実行する*前*の計画とテストに関して推奨されるステップを説明しています。ベストプラクティスのほか、実際のアップグレードの実行準備が整っている場合のシングル ノード サーバーおよびマルチノード インストールのアップグレード ステップに関する情報が記載されています。可能であれば考えられる落とし穴について説明し、ユーザーがこれらを回避できるようサポートします。

注: 現在のバージョンの Tableau を新しいバージョンと比較するときは、Tableau リリース ナビゲーター を使用して、Tableau Desktop、Server、Prep の機能を検索するか、機能の完全なリストを表示します。

Tableau Server on Windows については、「Tableau Server のアップグレードの概要」を参照してください。

コンピューターのオペレーティング システムのアップグレード

重要: メジャー バージョン オペレーティング システムや Linux システム カーネルのアップグレード (たとえば、、RHEL 7.x から RHEL 8.x など) は、ブルー/グリーンのガイドラインに従って実行す

る必要があります。**Tableau Server をすでに実行しているマシンでは、メジャー バージョンのアップグレードを直接実行しないでください。**以下の手順に従っても、Tableau Server は起動しても不安定な状態になります。

Tableau Server を実行しているマシンで直接 OS のメジャー バージョンのアップグレードを行うと、Tableau Server のインストールはサポートされなくなります。

ブルー/グリーン更新の詳細については、Tableau Server のアップグレードにブルー/グリーン アプローチを使用するを参照してください。

Tableau Server コンピューターのオペレーティング システムまたはクラスター内のノードをアップグレードする場合は、必ずアップグレードのベストプラクティスを確認してそれに従ってください。**メジャー アップグレード (たとえば、、RHEL 7.x から RHEL 8.x など) は、ブルー/グリーンのガイドラインに従って実行する必要があります。**詳細については、Tableau Server コンピューターの更新に関するベストプラクティスおよびTableau Server のアップグレードにブルー/グリーン アプローチを使用するを参照してください。

アップグレード パスを選択する

青/緑のアップグレード

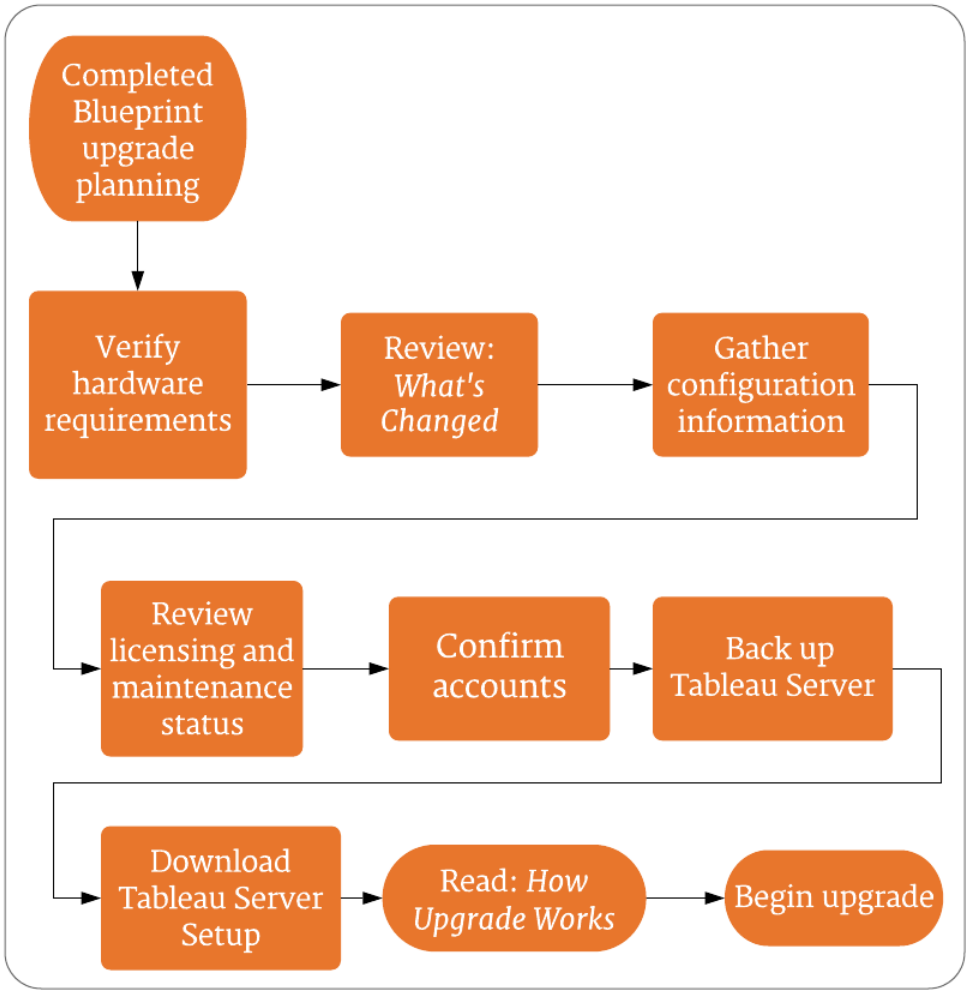
青/緑のアップグレードは、ダウンタイムを最小限に抑える特殊なタイプのアップグレード アプローチですが、実装には知識の豊富な IT チームとリソースが必要です。これらはすべての組織に適しているわけではありませんが、適切な技術知識とスタッフを備えた組織にとっては適切なアップグレード ソリューションになる可能性があります。詳細については、「Tableau Server のアップグレードにブルー/グリーン アプローチを使用する」を参照してください。

インプレース アップグレード

Linux で Tableau Server をアップグレードする手順については、2018.1 以降からのアップグレード (Linux)を参照してください。

アップグレードの準備

フロー チャートとリンク先のトピックに従って、Tableau Server をアップグレードするための環境を準備します。



リリース ナビゲーター

注: 現在のバージョンの Tableau を新しいバージョンと比較するときは、Tableau リリース ナビゲーター を使用して、Tableau Desktop、Server、Prep の機能を検索するか、機能の完全

なリストを表示します。

サーバーのアップグレード- 最小ハードウェア推奨事項

Tableau Server をアップグレードして稼働環境で使用するコンピューターは、ハードウェアが最低限の推奨事項を満たすか上回る必要があります。推奨事項は一般的なものです。Tableau Server インストールに必要な実際のシステム要件は、ユーザー数、抽出の数とサイズなど、多くの要因によって異なります。セットアッププログラムにより、コンピューターが次の要件を満たしていないと判断された場合、警告が表示されますが、セットアッププロセスは続行できます。ここに記載されている最小推奨事項は一般的なガイダンスとすることを目的としています。お使いの環境での推奨事項は異なる可能性があります。詳細については、「推奨されるベースライン構成」トピックの「ハードウェア推奨事項」セクションを参照してください。

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
シングルノード	• 64 ビット (x86_64 チップセット) • SSE4.2 および POPCNT 命令セットをサポートする必要があります • ARM ベースのプロセッサーはサポートされていません	8 コア (16vCPU)、2.0 GHz 以上	バージョン 2022.3 以降: • 128 GB バージョン 2021.4.0 から バージョン 2022.1.x: • 64 GB バージョン 2021.3.x 以前: • 32 GB	50 GB
	Tableau Server インストールに Tableau Prep Conductor を追加する場合、2 番目			

インストールの種類	*プロセッサー*	*CPU*	*RAM*	*空きディスク領域*
	のノードを追加し、これを Tableau Server Prep Conductor の実行専用にすることをお勧めします。このノードには最低 4 コア (8 vCPU)、RAM 16 GB が必要です。			
	推奨されるスワップ メモリ: システム RAM の平方根または 8 GB のいずれか大きい方。			
マルチノードおよびエンタープライズ展開	技術的ガイダンスについては Tableau にお問い合わせください。 ノードは最小ハードウェア推奨事項を満たしているか、それを超える必要があります。ただし、次の場合は除きます。 • バックグラウンダーのインスタンスを最大 2 つ実行している専用バックグラウンダー ノード: 4 コアは許容可能。 • Tableau Prep Conductor の専用ノード: 最低 4 コア (8 vCPU)、RAM 16 GB。 • 独立したゲートウェイの専用ノード: 最低 2 コア (4 vCPU)、8 GB の RAM、および 100 GB の空きディスク領域。			

重要: ディスク領域の要件は TSM を初期化するまでは確認できません。十分な領域がない場合、Tableau Server パッケージのインストールが終了するまでその旨を通知されません。

50 GB のディスク空き領域。少なくとも 15GB が `/opt` ディレクトリに割り当てられ、残りがデータストレージ用として `/var` ディレクトリに割り当てられます。

- 空きディスク容量は、Tableau Server セットアップ プログラムの解凍後に計算されます。セットアップ プログラムでは約 1 GB の容量を使用します。抽出を使用するかどうかなどさまざまな要因に応じて、追加ディスク領域の割り当てが必要になる場合があります。

  Tableau Server の中核部分は、15 GB 以上の空きディスク領域があるディレクトリにインストールする必要があります。十分なスペースがないコンピューターに Tableau Server をイン

ストールしようとすると、Tableau Server パッケージはインストールされますが、設定を続行することができません。既定のインストール場所は /opt ディレクトリです。RHEL ディストリビューションで Tableau Server のインストール パスを変更することができます。

抽出を頻繁に使用する予定がある場合は、追加ディスク領域の割り当てが必要になる場合があります。インストール時に、データ(抽出) を保存するための別のディレクトリを指定することができます。

- **外部ファイル ストアに対するネットワーク接続ストレージ領域の要件:** 外部ファイル ストアを使用した Tableau Server を構成する場合は、ネットワーク接続ストレージ専用のストレージ領域の容量を見積もる必要があります。

  ストレージ サイズの見積もり: パブリッシュと抽出更新に必要なストレージの容量を考慮する必要があります。さらに、トピックオプション 2: リポジトリを個別にバックアップするで説明されているように、リポジトリのバックアップを個別に実行するオプションを特に選択しない限り、リポジトリのバックアップ サイズも考慮する必要があります。

  - 抽出:
    - Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。

    - 抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。

  - リポジトリのバックアップ:

    - リポジトリデータのサイズは、<data directory>/pgsql/data/base ディレクトリのサイズを確認することで見積もることができます。

    - リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、workgroup.pg_dump ファイルのサイズを確認します。

- コア数は "物理" コアに基づいています。物理コアは実際のサーバー ハードウェアまたは仮想マシン (VM) 上のコアを表す場合があります。コア数のカウントの目的でのハイパー スレッドは無視されます。

- 表示されている RAM はシングル ノード インストールでの最小限の推奨です。たとえば、アクティビティ、ユーザー数、バックグラウンド ジョブなどに応じて、RAM が増えるとインストールの機能が向上する場合があります。

推奨事項の全リストと最小要件を確認するには、Tableau Server の最小ハードウェア要件と推奨事項を参照してください。Tableau がスケーラビリティのテストのために内部で使用するハードウェア仕様については、「本番環境インストールに関するハードウェアの推奨事項」を参照してください。

クラウドでの Tableau Server のハードウェア推奨事項については、以下を参照してください。

- AWS Cloud 上の Tableau Server on Linux 管理者ガイドの「AWS インスタンスの種別とサイズを選択する」

- 「Google Cloud Platform 上の Tableau Server on Linux 管理者ガイド」の「Google Compute Engine 仮想マシンの種別とサイズを選択する」

- Microsoft Azure 上の Tableau Server on Linux 管理者ガイドの「Microsoft Azure 仮想マシンの種別とサイズを選択する」

続けて、「サーバーのアップグレード- 変更された内容の確認」を行います。

サーバーのアップグレード- 変更された内容の確認

このトピックには、Tableau Server 2019.1 以降のバージョンでの重要な製品変更のリストが含まれています。このトピックにリストされている変更によって、アップグレードプロセス自体やアップグレード後の機能に影響が出る可能性があります。これらの変更を注意深く読み、必要な変更と修正手順をメモしてください。これらの修正手順は、アップグレードプロセスまたはアップグレード後の構成計画の一部として含められています。

このリストは累積的であるため、以前のバージョン (2019.1 など) からアップグレードする場合は、お使いのバージョンとアップグレード先のバージョンとの間にあるすべてのバージョンの変更点のリストをお読みください。リストが含まれたワークブックをダウンロードして、独自のカスタマイズされたチェックリストを作成することもできます。

より堅牢な検証とテスト計画を開発するには、アップグレードする前に新機能も確認することをお勧めします。すべての新機能と変更された機能の完全なリストを表示するには、**[ステータス]** フィルターにリストされているすべての項目を選択します。

サーバーのアップグレード- 構成の詳細の収集に進みます。

サーバーのアップグレード- 構成の詳細の収集

インプレース アップグレードを実行する場合 (つまり、アップグレードの一環としてハードウェアを更新しない場合)、ほぼすべての構成データが保持されます。厳密に言うと、この場合、すべての構成情報を収集する必要はありませんが、このトピックで説明している情報を収集することをお勧めします。最悪のシナリオでは、アップグレードが失敗した場合、必要な復元に備えて、すべての構成情報が記録されます。いずれの場合も、収集した構成の詳細を使用して、完了したアップグレードを検証できます。

スクリーン ショットの取得

構成の基本を比較的簡単にキャプチャする方法は、Tableau サービス マネージャーの Web インターフェイス ページと Tableau Server 管理者エリアのスクリーン ショットを取得することです。

表示されているすべてのページをクリックして、スクリーン ショットを取得します。

- Tableau サービス マネージャーの Web UI へのサインイン
- Tableau Server 管理者エリアへのサインイン

オブジェクト数の記録

Tableau Server 管理者エリアにサインインしているときに、各サイトで次の項目をカウントして記録します。

- プロジェクト、ワークブック、ビュー、データ ソース
- ユーザーとグループ

ファイアウォール構成の記録

Linux 上で Tableau Server 用のローカル ファイアウォールを構成している場合は、記録のために構成をコピーすることをお勧めします。

設定に関するドキュメントでは、Firewalld を使用して、RHEL/CentOS ディストリビューションで実行される Tableau Server のシングルおよびマルチノード展開でファイアウォールを構成する方法を説明しています。「ローカル ファイアウォールの構成」を参照してください。

次のコマンドを実行して、ファイアウォール構成を取得します。

```
sudo firewall-cmd --list-all
```

Tableau サービス マネージャー コントローラー証明書の有効期限の確認

Tableau サービス マネージャー コントローラー証明書が引き続き有効であることを確認してください。

Tableau サービス マネージャー コントローラーの SSL 証明書の有効期限を確認するには、次の手順を実行します。

1. ターミナル セッションを開きます。
2. 次のコマンドを入力して、証明書が有効な日付を表示します。

   ```
   openssl s_client -connect <tsm_servername>:8850
   ```

   ```
   echo | openssl s_client -connect <tsm_servername>:8850
   2>/dev/null | openssl x509 -noout -dates
   ```

3. 証明書の有効期限が切れている場合は、サポートチームに問い合わせて事例を開始し、サポートを受けてください。

アセット ファイルの収集

Tableau Server にアップロードするサポート ファイル (証明書、IdP メタデータ、ロゴなど) の多くは、アップロード後に Tableau サービス マネージャーを使用してアクセスできません。具体的には、クライアント ファイル サービスによってアップロードおよび管理されるファイルは、展開全体に配布される前に名前が変更され、難読化されます。このプロセスでは、Tableau サービスに必要なファイル属性もパラメーター化されます。その結果、ファイルは、以降のファイルのファイル システム上でファイルの単一の場所にマッピングされません。以降のファイルをアップロードした場合、Tableau Server を実行している PC からそれらのファイルのコピーを保存してください。

- SAML 証明書ファイル
- SAML キー ファイル
- SAML IdP メタデータ ファイル
- OpenID.static.file
- Kerberos.keytab ファイル
- LDAP Kerberos keytab ファイル
- LDAP Kerberos conf ファイル
- 相互 SSL 証明書ファイル
- 相互 SSL 失効ファイル

- カスタマイズ ヘッダー ロゴ ファイル
- カスタマイズ サインイン ロゴ ファイル
- カスタマイズ コンパクト ロゴ ファイル

カスタムの構成情報の収集

一部の構成情報は、Tableau サービス マネージャーまたは Tableau Server の Web ページに表示されません。このセクションでは、Tableau の展開をカスタマイズした方法に応じて収集が必要になる場合がある構成の詳細を説明します。

セキュア SMTP

Tableau Server 用に TLS を構成している場合は、Tableau サービス マネージャーの Web UI の **電子メール サーバー**構成に含まれていない TLS 関連の構成を記録する必要があります。

TLS 関連の構成を収集するには、次のキー値を指定して `tsm configuration get` コマンドを実行する必要があります。

- `svcmonitor.notification.smtp.ssl_enabled`
- `svcmonitor.notification.smtp.ssl_required`
- `svcmonitor.notification.smtp.ssl_check_server_identity`
- `svcmonitor.notification.smtp.ssl_trust_all_hosts`
- `svcmonitor.notification.smtp.ssl_ciphers`
- `svcmonitor.notification.smtp.ssl_versions`

たとえば、SMTP TLS 用に構成されている暗号のリストを取得するには、次のコマンドを実行します。

```
tsm configuration get -k svcmonitor.notification.smtp.ssl_ciphers
```

上記の各キーの詳細については、「SMTP セットアップの構成」の「Tableau サービス マネージャー CLI」セクションを参照してください。

分析拡張機能

分析拡張機能 (以前は「外部サービス」と呼ばれていました) を構成している場合は、構成情報を記録する必要があります。Tableau Server 2020.2 以降にアップグレードすると、この機能のすべての構成が削除されます。

Tableau Server バージョン 2019.1 ~ 2020.1 から分析拡張機能の構成を取得するには、次のコマンドを実行します。

```
tsm security vizql-extsvc-ssl list
```

分析拡張機能の接続用に保存されているパスワードを取得するには (存在する場合)、次の tsm コマンドを実行します。

```
tsm configuration get -k vizqlserver.rserve.password
```

SSL を構成している場合は、分析拡張機能の証明書のコピーも必要です。

外部リポジトリ

これは、Tableau Server で外部リポジトリ構成を使用している場合にのみ適用されます。適用されるかどうかがわからない場合は、Tableau Server 外部リポジトリを参照してください。

外部リポジトリを使用している場合は、アップグレード時に追加の手順が必要になる場合があります。

- **バージョンの変更なし** - PostgreSQL にバージョンの変更がない場合、特別なアクションは必要ありません。
- **マイナー バージョンの変更** - PostgreSQL にマイナー バージョンの変更がある場合、Tableau Server をアップグレードする前に外部リポジトリをアップグレードする必要があります。ほとんどの場合、アップグレードを行うためのインプレース メソッドがあります。使用するメソッドはリポジトリの場所によって異なり、このドキュメントの対象外です。

- **メジャー バージョンの変更** - PostgreSQL にメジャー バージョンの変更がある場合、「新しいメジャー バージョンの PostgreSQL を外部リポジトリとして使用する Tableau Server のアップグレード」で説明されている手順に従う必要があります。

  次のステップが含まれます。

  1. PostgreSQL DB の新しいインスタンスを作成します。詳細については、以下を参照してください。
     - AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成
     - 詳細については、Azure で Azure Database PostgreSQL インスタンスを作成するを参照してください。

- Google Cloud での PostgreSQL インスタンスの作成
- PostgreSQL データベースをスタンドアロン インストールで作成

2. 構成ファイルを作成し、ステップ 1 で作成した新しいインスタンスの SSl 証明書ファイルをダウンロードします。

アップグレード中に、構成ファイルを使用して新しいインスタンスに Tableau Server を指定する必要があります。アップグレードプロセスによって、コンテンツが現在の外部リポジトリから新しいインスタンスに移行します。詳細については、新しいメジャー バージョンの PostgreSQL を外部リポジトリとして使用する Tableau Server のアップグレードを参照してください。

外部ファイル ストア

外部ファイル ストアを使用して構成された Tableau Server をアップグレードする場合、特別な手順、アクション、構成は必要ありません。通常の手順を使用して Tableau Server をアップグレードできます。

ポートのカスタマイズ

Tableau Server の動的ポート範囲を変更した場合、または手動でポートを構成した場合は、これらの変更を記録してください。

次のコマンドを実行します。

```
tsm topology list-ports
```

詳細については、「Tableau サービス マネージャーのポート」を参照してください。

サーバーのアップグレード- ライセンス ステータスの確認」に進みます

サーバーのアップグレード- ライセンス ステータスの確認

Tableau Server をアップグレードする前に、現在インストールされているプロダクト キーを確認し、Tableau カスタマー ポータルに記載されているプロダクト キー、メンテナンスの有効期限、およびライセンスの有効期限終了日と比較します。

アップグレード前に、プロダクト キーに関連付けられている有効期限と終了日がすべて最新の状態に設定されている必要があります。

ステップ 1: ライセンスの有効期限情報を表示します。

次のコマンドを実行して、Tableau Server インストールにインストールされているすべてのライセンスを表示します。

```
tsm licenses list
```

この tsm licenses list コマンドは、すべてのライセンスを返します。各ライセンスには多数のフィールドが含まれています。アップグレードする場合、次の 2 つのフィールドをメモする必要があります。

- サブスクリプション ライセンスごとに、LIC EXP フィールドの日付をメモします。LIC EXP には、ライセンスの有効期限が切れ、Tableau Server が動作を停止する日付が示されています。
- レガシー永続キーごとに、MAINT EXP フィールドをメモします。MAINT EXP には、Tableau Server 展開に対するメンテナンス契約の有効期限が切れる日付が表示されます。

ステップ 2: Tableau サービス マネージャーのライセンス有効期限と Tableau カスタマー ポータル**に記載されているライセンスの有効期限を比較します。**

ステップ 3: 必要に応じて、ライセンスを更新します。

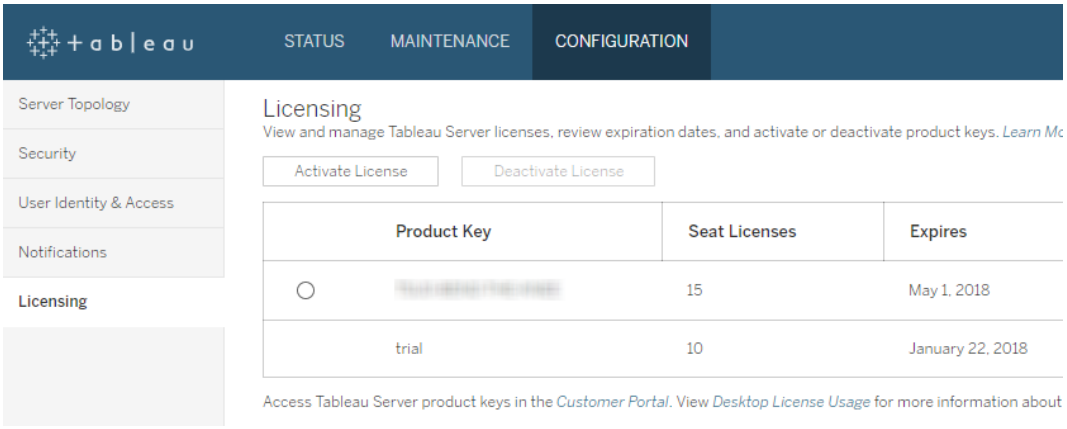
Tableau サービス マネージャーのメンテナンス日が最新ではない場合、有効期限が間もなく終了する場合、または有効期限が Tableau カスタマー ポータルに記載されている日付と同じでない場合は、ライセンスを更新します。

ライセンスを更新するには:

1. ブラウザーで TSM を開きます。

   http://<tsm-computer-name>:8850

2. **[構成]** および **[ライセンス発行]** をクリックし、**[すべて更新]** をクリックします。



上記の手順を実行しても、メンテナンスまたは有効期限の終了日が更新されず、Tableau カスタマー ポータルに現在の日付に近い終了日が表示される場合は、Tableau テクニカル サポートにお問い合わせください。

アップグレードプロセスの一環としてプロダクト キーのライセンスが再認証されます。

「サーバーのアップグレード- アカウントの確認」に進みます。

サーバーのアップグレード- アカウントの確認

Tableau Server をアップグレードする前に、Tableau Server のセットアップの実行に使用するアカウントに `sudo` アクセス権があることを確認してください。

組織が外部アイデンティティ ストア (LDAP または Active Directory) を使用している場合は、LDAP とのバインドに使用するアカウントの認証資格情報またはキータブ ファイルを使用することをお勧めします。

「サーバーのアップグレード- Tableau Server のバックアップ」に進みます。

サーバーのアップグレード- Tableau Server のバックアップ

アップグレードプロセスを開始する前に Tableau Server のインストールのバックアップを作成することを強くお勧めします。アップグレードを開始する前に作成されたバックアップには、アップグレードさ

れた環境のテストバージョンを設定するために必要なデータと、アップグレードプロセスが失敗した場合の復元機能も含まれています。バックアップはアップグレード中のインクルードステップではありません。ただし、アップグレードに PostGRES へのマイナーバージョンアップグレードが含まれていて、アップグレード中に内部で使用するための pg-only アップグレードが作成される場合を除きます。

注:

- バックアップを作成する直前に本番環境でサブスクリプションおよびスケジュールを無効にし、バックアップの完了後に再度有効にすることをお勧めします。この操作は、テスト環境でバックアップを復元したときに、ユーザーが重複するサブスクリプションやメール メッセージを受け取るのを回避するのに役立ちます。
- 大規模なインストールや多数の抽出がある場合は、フル バックアップにしばらく時間がかかることがあります。
- バックアップの作成時からアップグレードの実行時までの間に加えられた変更はバックアップに含まれないため、失われます。
- 不必要なファイルの削除

サーバー構成データをバックアップするには、`tsm settings` コマンドを使用します。

`tsm maintenance backup` コマンドを使用すると、バックアップ ファイルに現在の日付が追加されます。

```
tsm maintenance backup -f <backup_file> -d
```

詳細については、「tsm maintenance backup」を参照してください。

続けて、「サーバーのアップグレード- セットアップのダウンロード」を行います。

サーバーのアップグレード- セットアップのダウンロード

Tableau Server セットアップ プログラムをダウンロードして、Tableau Server を実行しているコンピューターにコピーする必要があります。Tableau Server をマルチノードで展開して実行している場合は、クラスタ内の各ノードに対してセットアップ プログラムをコピーする必要があります。

最新バージョンの Tableau Server ソフトウェアを取得するには、カスタマー ポータルに移動してください。Tableau を購入すると、カスタマー ポータルへのサインイン用ユーザー名とパスワードを入手できます。

重要: アップグレードする先のバージョンは、常に最新のメンテナンス リリースをダウンロードしてインストールしてください。

続いて、「Tableau Server のアップグレードの仕組み」を行います。

Tableau Server のアップグレードの仕組み

2018.2 以降の Tableau Server をアップグレードする場合は、以前のバージョンをアンインストールしないでください。アップグレードプロセスは、新しいバージョンを既存のバージョンと同じ場所にインストールし、前のバージョンよりも新しいものに切り替えるように設計されています。

TSM から TSM へのアップグレード (バージョン 2018.2.x からそれ以降のバージョン) を実行する場合、既存バージョンはそのままにしておき、アップグレード スクリプトを使用してアップグレードする直前まで実行しておく必要があります。アップグレードする前に既存のバージョンをアンインストールすると、アップグレードできなくなります。エンドユーザーとしては、既存バージョンの実行中に新しいバージョンをインストールするため、ダウンタイムが減るという効果があります。サーバーが実行されない時間は、アップグレード スクリプトにより実際に新しいバージョンにアップグレードしている間だけです。

新規バージョンのインストールとアップグレード後、以前のバージョンをいつでもアンインストールできます。古いバージョンを無限にインストールしたままにしておくことも可能ですが、古いバージョンに固有のファイルが使用しているディスク容量を空けるために、アンインストールを選択することもできます。Tableau Server のアンインストールと、Tableau Server の全アスペクトの完全な削除との違いについては、「Tableau Server のアンインストール」を参照してください。

Tableau Server をアップグレードするには、アップグレードプロセスの一部としてサーバーの停止と起動が必要です。停止から起動までの間は Tableau Server を利用できません。

変更箇所 - アップグレード前に知っておく事柄

バージョン 2020.2 以降、このトピックは「Tableau Server の新機能」と統合し、Tableau Server の新機能や変更された機能と動作について 1 か所で提供されるようになりました。このバージョンと以前のバージョンで導入された新機能や変更については、Tableau Server のリリース ノートを参照してください。

注: 現在のバージョンの Tableau を新しいバージョンと比較するときは、Tableau リリース ナビゲーター を使用して、Tableau Desktop、Server、Prep の機能を検索するか、機能の完全なリストを表示します。

Tableau Server コンピューターの更新に関するベストプラクティス

このトピックでは、Tableau Server を実行しているコンピューターでマイナー バージョンのアップグレードまたはパッチを実行する手順について説明します。

重要: メジャー バージョン オペレーティング システムや Linux システム カーネルのアップグレード (たとえば、、RHEL 7.x から RHEL 8.x など) は、ブルー/グリーンのガイドラインに従って実行する必要があります。**Tableau Server をすでに実行しているマシンでは、メジャー バージョンのアップグレードを直接実行しないでください。**以下の手順に従っても、Tableau Server は起動しても不安定な状態になります。

Tableau Server を実行しているマシンで直接 OS のメジャー バージョンのアップグレードを行うと、Tableau Server のインストールはサポートされなくなります。

ブルー/グリーン更新の詳細については、Tableau Server のアップグレードにブルー/グリーン アプローチを使用するを参照してください。

Tableau Server を実行しているコンピューターでオペレーティング システムまたは Linux カーネルのマイナー バージョンの更新またはパッチを実行するには、次の手順を実行します。

1. TSM を使用して Tableau Server を停止します。

   コマンドプロンプトで `tsm stop` と入力し、すべてのサービスが完全に停止するまで待ちます。

2. スクリプト `stop-administrative-services` を実行して TSM 管理サービスを停止します。既定では次の場所にあります。

   ```
   /opt/tableau/tableau_server/packages/scripts.<version_
   code>/stop-administrative-services
   ```

3. クラスター内のすべてのノードに対して、IT チームまたはオペレーティング システムのベンダーの指示に従ってオペレーティング システムにパッチを適用します。

4. パッチを適用するためにオペレーティング システムの再起動が必要な場合は、Tableau Server を再起動する前に、Tableau Server クラスター内のすべてのホストの再起動が完了していることを確認してください。

5. TSM を使用して Tableau Server を起動します。

   コマンドプロンプトで `tsm start` と入力し、すべてのサービスが完全に開始するまで待ちます。

2018.1 以降からのアップグレード (Linux)

以下のフロー チャートとリンクされたトピックに従って、Tableau Server のインプレース アップグレードを実行します。

アップグレードを開始する前に、アップグレードの準備が完了していることを確認してください。詳細については、「アップグレードの準備」を参照してください。

アップグレードの一環として Tableau Server を新しいハードウェアに移行している場合は、新しいハードウェアへの移行を参照してください。

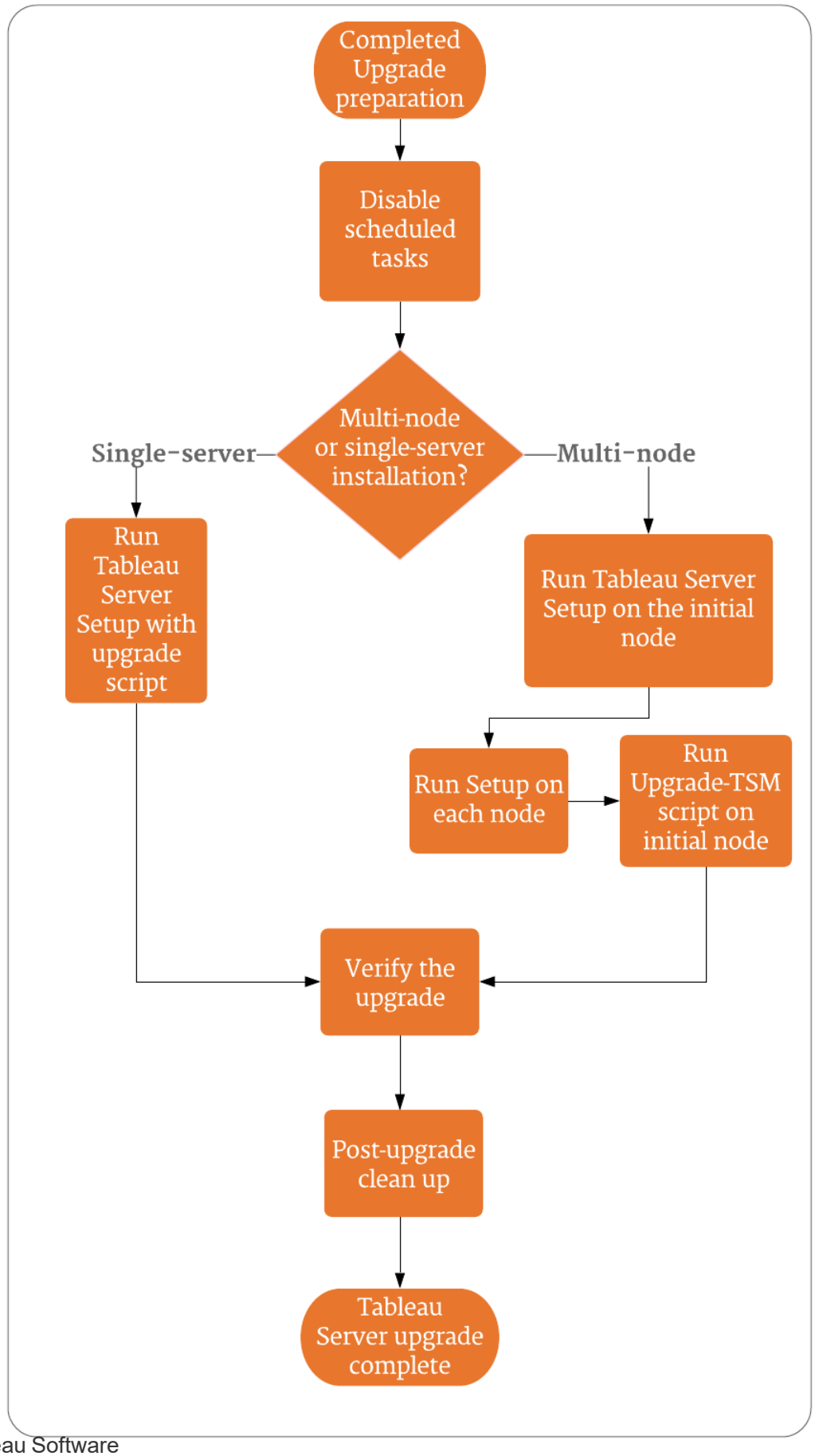

Completed Upgrade preparation
Disable scheduled tasks
Multi-node or single-server installation?
Single-server
Multi-node
Run Tableau Server Setup with upgrade script
Run Tableau Server Setup on the initial node
Run Setup on each node
Run Upgrade-TSM script on initial node
Verify the upgrade
Post-upgrade clean up
Tableau Server upgrade complete

Tableau Server のアップグレードに役立つサポートとサービス

Tableau Server のアップグレードには、計画とテストが必要です。アップグレードは、通常、ユーザーが Tableau Server を使用していない時間帯に行われます。週末にアップグレードを計画し、Tableau テクニカル サポートが必要になると予想している場合は、Tableau Web サイトの「テクニカル サポートプログラム」の情報を参照してください。対応可能な範囲は、ご利用のサポートのレベルによって異なります。Tableau Server のインストールの計画およびアップグレードに支援が必要な場合、詳細については、「Tableau グローバル サービス: サーバーのアップグレード」(英語) ページを参照してください。

サーバー アップグレード - スケジュールされたタスクの無効化

重要: アップグレードを実行する前に、スケジュールされたタスクを無効にすることをお勧めします。これにはデータ コンテンツのすべての更新も含まれ、アップグレード前のバックアップを作成する前に実行する必要があります。これには、REST API ベースの抽出更新や tabcmd の使用によって開始されるジョブなど、Tableau Server の外部でトリガーされるジョブの無効化が含まれる場合があります。

スケジュールされた抽出更新、フロー、サブスクリプションをすべて無効にするには、次の手順を実行します。

1. Tableau Server 管理者として Tableau Server 管理者エリアへのサインイン。
2. サーバー全体の [スケジュール] ページに移動します。
   - (既定の) サイトが 1 つしかない Tableau Server で、左ペインの **[スケジュール]** をクリックします。
   - 複数のサイトがある Tableau Server では、左側のペインの上部にあるドロップダウン メニューで **[すべてのサイトの管理]** をクリックし、**[スケジュール]** をクリックします。
3. [スケジュール] ページで、**[すべて選択]** をクリックします。
4. **[アクション]** メニューで **[無効にする]** を選択し、表示されるプロンプトで **[無効にする]** をクリックします。

続けて、「単一サーバーのアップグレード-- セットアップの実行」または「マルチノード アップグレード-- セットアップの実行」を行います。

単一サーバーのアップグレード-- セットアップの実行

セットアップの実行

次の手順に従って、Tableau Serverバージョン 2018.2 以降のシングル ノード インストールをアップグレードします。

1. アップグレードしようとしているコンピューターに sudo アクセス権を持つユーザーとしてログオンします。
2. `.rpm` または `.deb` Tableau Server パッケージをコピーしたディレクトリに移動します。
3. パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。
     - **既定の場所**: 既定の場所 (`/opt/tableau/tableau_server`) にインストールするには次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

       ```
       sudo yum update

       sudo yum install tableau-server-<version>.x86_64.rpm
       ```

     - **既定以外の場所**: 既定以外の場所にインストールするには、`rpm -i` を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

次のコマンドを実行します。

```
sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
```

> **注:** Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で `yum` の使用が禁止されているため、`rpm -i` を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get -y install gdebi-core
sudo gdebi -n tableau-server-<version>_amd64.deb
```

4. アップグレードスクリプトを実行します。

   スクリプトの実行時に Tableau Server が停止していない場合、スクリプトによりサーバーが停止していないことが通知され、サーバーの停止が提案されます。スクリプトを実行する前に、`tsm stop` コマンドを使用してサーバーを停止することもできます。アップグレードを完了するには、Tableau Server を停止する必要があります。

   含める必要があるオプションは、アップグレードするバージョンによって異なります。

   - バージョン 2019.3 以降:

```
sudo /opt/tableau/tableau_server/packages/scripts.<version_
code>/upgrade-tsm --accepteula
```

ここで、<version_code> はアップグレードしている新しいバージョンの正式名です (例: scripts.20183.18.1128.2033)。

バージョン 2019.3.0 以降では、2019.2.x 以降からアップグレードすると、ログインしているアカウントを使用してスクリプトが実行されます。プロンプトが表示される場合は、パスワードを入力します。詳細については、変更箇所 - アップグレード前に知っておく事柄を参照してください。-u オプションを使用し、最初のノードがインストールされているコンピューターで管理パーミッションを持つユーザーを指定すると、管理パーミッションを持つ別のユーザーを指定できます。管理ユーザーのパスワードを求めるプロンプトが表示されます。

- バージョン 2018.1 からバージョン 2019.2.x:

```
sudo /opt/tableau/tableau_server/packages/scripts.<version_
code>/upgrade-tsm -u <system_admin> --accepteula
```

ここで、<version_code> は、アップグレードしている新しいバージョンの正式名です (例: scripts.20183.18.1128.2033)。また、<system_admin> は、最初のノードがインストールされているコンピューターで管理パーミッションを持つユーザーです。管理ユーザーのパスワードを求めるプロンプトが表示されます。

この -u オプションは 2018.1 以降追加されました。詳細については、変更箇所 - アップグレード前に知っておく事柄を参照してください。

upgrade-tsm スクリプトで使用できるすべてのオプションを表示するには、-h オプションを使用します。例は次のとおりです。

```
sudo /opt/tableau/tableau_server/packages/scripts.<version_
code>/upgrade-tsm -h
```

5. アップグレードが完了したら、ターミナル セッションを終了し、もう一度ログインします。これにより、セッションで更新された TSM バージョンが使用されるようになります。

6. Tableau Server を起動します。

```
tsm start
```

続けて、「Tableau Server のアップグレードの確認」を行います。

マルチノードアップグレード -- セットアップの実行

セットアップの実行

1. sudo アクセス権を持つユーザーとしてクラスター内の最初のノードにログインします。
2. .rpm または .debTableau Server パッケージをコピーしたディレクトリに移動します。
3. パッケージ マネージャーを使用して、Tableau Server パッケージをインストールします。

   新しいバージョンを既存のバージョンと同じ場所にインストールする必要があります。インストール場所は、すべてのノードで同じである必要があります。シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。

   - CentOS を含む RHEL ライクなディストリビューションでは、既定以外の場所に Tableau をインストールすることができます。
     - **既定の場所**: 既定の場所 (/opt/tableau/tableau_server) にインストールするには次のコマンドを実行します (<version> の書式設定は「メジャー-マイナー-メンテナンス」(例: 2019-2-5))。

       ```
       sudo yum update

       sudo yum install tableau-server-<version>.x86_64.rpm
       ```

     - **既定以外の場所**: 既定以外の場所にインストールするには、rpm -i を使用する必要があります。また、依存するすべてのパッケージをインストールする必要があります。下記の注を参照してください。

       次のコマンドを実行します。

       ```
       sudo rpm -i --prefix /preferred/install/path tableau-server.rpm
       ```

> **注:** Tableau Server のインストールに `yum` を使用した場合、すべての依存するパッケージが自動的にダウンロード、インストールされます。これが Tableau のインストール推奨方法です。既定以外の場所にインストールする場合、または所属の組織で `yum` の使用が禁止されているため、`rpm -i` を使用しなくてはならない場合は、依存するすべてのパッケージを個別にインストールしなくてはなりません。依存パッケージのインストールの詳細については、Tableau Server を Linux を実行するエアギャップ環境にインストールするを参照してください。

- Ubuntu で次のコマンドを実行します (`<version>` の書式設定は「メジャー-マイナー-メンテナンス」(例: `2019-2-5`))。

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get -y install gdebi-core
sudo gdebi -n tableau-server-<version>_amd64.deb
```

続けて、「マルチノードアップグレード-- 各ノードでセットアップを実行する」を行います。

マルチノードアップグレード-- 各ノードでセットアップを実行する

セットアップの実行

各追加ノードで、Tableau Server セットアッププログラムをコピーしたフォルダーに進み、最初のノードで行ったようにセットアッププログラムを実行します。これにより、各ノードで新しいバージョンの Tableau Server が既存の実行中のバージョンと同じ場所にインストールされます。

続けて、「マルチノードアップグレード-- アップグレードスクリプトの実行」を行います。

マルチノードアップグレード-- アップグレードスクリプトの実行

アップグレードスクリプトの実行

1. クラスター内のすべてのノードに新しいバージョンをインストールしたら、最初のノードでアップグレードスクリプトを実行しします。

   スクリプトの実行時に Tableau Server が停止していない場合、スクリプトによりサーバーが停止していないことが通知され、サーバーの停止が提案されます。スクリプトを実行する前に、`tsm stop` コマンドを使用してサーバーを停止することもできます。アップグレードを完了するには、Tableau Server を停止する必要があります。

   含める必要があるオプションは、アップグレードするバージョンによって異なります。

   - バージョン 2019.3 以降:

     ```
     sudo /opt/tableau/tableau_
     server/packages/scripts.<version_code>/upgrade-tsm --
     accepteula
     ```

     ここで、`<version_code>` はアップグレードしている新しいバージョンの正式名です (例: `scripts.20183.18.1128.2033`)。

     バージョン 2019.3.0 以降では、2019.2.x 以降からアップグレードすると、ログインしているアカウントを使用してスクリプトが実行されます。プロンプトが表示される場合は、パスワードを入力します。詳細については、変更箇所 - アップグレード前に知っておく事柄を参照してください。`-u` オプションを使用し、最初のノードがインストールされているコンピューターで管理パーミッションを持つユーザーを指定すると、管理パーミッションを持つ別のユーザーを指定できます。管理ユーザーのパスワードを求めるプロンプトが表示されます。

   - バージョン 2018.1 からバージョン 2019.2.x:

```
sudo /opt/tableau/tableau_
server/packages/scripts.<version_code>/upgrade-tsm -u
<system_admin> --accepteula
```

ここで、<version_code> は、アップグレードしている新しいバージョンの正式名です (例: scripts.20183.18.1128.2033)。また、<system_admin> は、最初のノードがインストールされているコンピューターで管理パーミッションを持つユーザーです。管理ユーザーのパスワードを求めるプロンプトが表示されます。

この -u オプションは 2018.1 以降追加されました。詳細については、変更箇所 - アップグレード前に知っておく事柄を参照してください。

upgrade-tsm スクリプトで使用できるすべてのオプションを表示するには、-h オプションを使用します。例は次のとおりです。

```
sudo /opt/tableau/tableau_server/packages/scripts.<version_
code>/upgrade-tsm -h
```

2. アップグレードが完了したら、ターミナル セッションを終了し、もう一度ログインします。これにより、セッションで更新された TSM バージョンが使用されるようになります。

3. Tableau Server を起動します。

```
tsm start
```

続けて、「Tableau Server のアップグレードの確認」を行います。

Tableau Server のアップグレードの確認

このトピックのセクションで説明している手順に従って、サーバーのアップグレードが成功したかどうかを確認します。

Tableau Service プロセスの確認

検証プロセスを開始するには、コマンド ラインを使用して Tableau サービス マネージャーにサインインし、Tableau Server プロセスのステータスを確認します。

1. tsmadmin グループのメンバー アカウントでコマンドプロンプトを開きます。

   tsm コマンド ラインの使用については、tsm CLI の使用を参照してください。

2. コマンドtsm status -v. を実行して、すべての Tableau Server プロセスのステータスを表示します。

TSM コマンドを実行しようとするとアクセス拒否エラーが表示される場合はどうすればいいですか?

残りのインストールの構成に使用するアカウントは、初期化中に作成されたtsmadmin グループのメンバーでなければなりません。tsmadmin グループのユーザー アカウントを表示するには、次のコマンドを実行します。

```
grep tsmadmin /etc/group
```

ユーザー アカウントがグループにない場合、次のコマンドを実行してユーザーをtsmadmin グループに追加します。

```
sudo usermod -G tsmadmin -a <username>
```

Tableau サービス マネージャーのグローバル設定の確認

アップグレードの準備の一環として、Tableau サービス マネージャーの Web UI 設定のスクリーンショットを取得している必要があります。Tableau サービス マネージャーの Web UI へのサインイン、設定とスクリーン ショットを比較します。

サブスクリプションとスケジュールの有効化

1. Tableau Server 管理者として Tableau Server 管理者エリアへのサインイン。

2. サーバー全体の [スケジュール] ページに移動します。

   - (既定の) サイトが 1 つしかない Tableau Server で、左ペインの **[スケジュール]** をクリックします。
   - 複数のサイトがある Tableau Server では、左側のペインの上部にあるドロップダウン メニューで **[すべてのサイトの管理]** をクリックし、**[スケジュール]** をクリックします。

3. [スケジュール] ページで、**[すべて選択]** をクリックします。
4. **[アクション]** メニューで、**[有効]** をクリックしてから、表示されるプロンプトで **[有効]** をクリックします。

ユーザー アクセスの確認

ユーザー アカウントで、Tableau Desktop とブラウザーを使用して Tableau Server にサインインします。

Tableau Server が自分の組織外から利用できる場合は、インターネットからの接続を確認します。

組織が Tableau Server へのモバイル接続を許可している場合は、モバイル デバイスを使用してサーバーにサインインします。

パブリッシュされたワークブックの表示

ユーザー アカウントで Tableau Server にサインインし、パブリッシュされたワークブックを参照します。ワークブックを開き、ブラウザーで表示します。

ワークブックとデータ ソースのパブリッシュの確認

Creator ライセンスがあるアカウントを使用して、Web 作成機能を検証してデータに接続し、ワークブックとデータ ソースを作成してパブリッシュします。

- Web 上のデータへの接続
- Web でデータ ソースをパブリッシュする
- Tableau サイトへのワークブックのアップロード

確認 Tableau Prep Builder

組織が Tableau Prep Builder を実行している場合は、データに接続したり、フローを開いたりできることや、その他の Prep タスクを確認します。「Tableau Prep: データへの接続」を参照してください。

Tableau オブジェクトの数の確認

アップグレードの準備の一環として、展開内のプロジェクト、ワークブック、ビュー、データ ソース、ユーザー、およびグループの数を確認する必要があります。すべてのオブジェクトが引き続き表示されることを確認します。

Tableau Server 管理者として Tableau Server 管理者エリアへのサインイン。

- プロジェクト、ワークブック、ビュー、データソースの数を確認します
- ユーザーとグループの数を確認します

API 機能の確認

組織が Tableau API を使用してコンテンツを展開または開発している場合は、それらの API が動作可能であることを確認します。

REST API

「入門用チュートリアル: 概要とセットアップ」のサンプル cURL を実行して、REST API のアクセスと認証を確認します。

```
curl "https://MY-SERVER/api/3.8/auth/signin" -X POST -d @signin.xml
```

signin.xml の内容:

```
<tsRequest>
  <personalAccessTokenName="MY_PAT_NAME"
       <credentials name="username" password="password" >
        <site contentUrl="MarketingSite" />
  </credentials>
</tsRequest>
```

応答例:

```
<tsResponse version-and-namespace-settings>
  <credentials token="12ab34cd56ef78ab90cd12ef34ab56cd">
       <site id="9a8b7c6d-5e4f-3a2b-1c0d-9e8f7a6b5c4d"
       contentUrl="MarketingSite"/>
  </credentials>
</tsResponse>
```

互換性テスト

開発された他の機能については、互換性テストを実行して、アップグレードされたバージョンの Tableau Server が既存のソリューションで期待どおりに動作することを確認します。

- ダッシュボードの拡張機能
- 埋め込みの Tableau 用の JavaScript API

- コネクタ(Web データコネクタ、Taco)

「アップグレード後のクリーニング」に進みます。

アップグレード後のクリーニング

「Tableau Server のアップグレードの仕組み」で説明したように、Tableau Server のアップグレードプロセスでは、既存のバージョンと同じ場所に新しいバージョンがインストールされます。アップグレードが完了し、検証が完了したので、古いバージョンの Tableau Server を削除してディスク領域を解放できます。この手順はオプションです。

以前のバージョンをアンインストールする

この手順は、新しいバージョンの Tableau Server にアップグレードした後に、以前の Tableau Server バージョンのパッケージをアンインストールしてディスク容量を空けるために使用します。

1. environment.bash ファイルで、現在使用している Tableau Server のバージョンを確認します。コマンドプロンプトで、次のように入力します。

   ```
   grep TABLEAU_SERVER_DATA_DIR_VERSION /etc/opt/tableau/tableau_
   server/environment.bash
   ```

2. コンピュータにインストールされる Tableau Server パッケージのバージョンを決定します。

   - CentOS を含む RHEL ライクなディストリビューションで、次のコマンドを実行します。

     ```
     yum list installed tableau-server"*"
     ```

   - Ubuntu で、次のコマンドを実行します。

     ```
     apt list --installed tableau-server"*"
     ```

3. パッケージ マネージャーを使用して Tableau Server パッケージを削除します。

   - CentOS を含む RHEL ライクなディストリビューションで、次のコマンドを実行します。

     ```
     sudo yum remove tableau-server-<version>.x86_64
     ```

- Ubuntu で次のコマンドを実行します。

```
sudo apt-get purge tableau-server-<version>
```

Tableau Server のアップグレードにブルー/グリーン アプローチを使用する

多くの組織は Tableau Server のインプレース アップグレードを実行することで十分なサービスを受けていますが、Tableau をミッションクリティカルに使用している他の組織は、より堅牢なエンタープライズ グレードのアップグレード アプローチを必要とする場合があります。このアプローチでは、追加の労力とリソースを費やして、リスクと影響を軽減する必要があります。このような場合、Tableau では、その結果を達成するために「ブルー/グリーン」アップグレード アプローチを推奨しています。このアプローチの特徴は次のとおりです。

- 新しく構成された非本番ホスト環境にアプリケーションを信頼性の高い方法 (インプレース アップグレードと比べて) で展開する
- 一般的なユース ケースをテストして、組織に影響を与える可能性のある動作の変更を検出することなど、非本番環境で新しいバージョンの実稼働前テストを実行して、ビジネス クリティカルな機能を検証する
- 統合とリソース要件を検証するために、本番環境と同一の環境でテストする
- 以前のバージョンにすばやくロールバックできる

この大まかなガイダンスでは、Tableau Server (およびその他のミッション クリティカルなアプリケーション) を自信を持ってアップグレードするために、Tableau の最大のお客様の多くが成功裏に使用しているブルー/グリーン アップグレード アプローチについて説明します。ブルー/グリーン アップグレードは、長年の業界慣行です。多くのバリエーションがあり、以下に示す手順は 1 つの可能な手法の例です。組織で行う次のアップグレードのために、このアプローチをカスタマイズすることに関して具体的なガイダンスが必要な場合は、プレミアム サポートのテクニカル アカウント マネージャーに相談するか、サービス契約についてアカウント エグゼクティブまたはデリバリー パートナーと話し合ってください。

ブルー/グリーン アプローチでは、新しいバージョンの Tableau Server を、既存の本番環境 (「ブルー」インストール) を反映した非本番環境 (「グリーン」インストール) にインストールします。当該のライセンス契約で詳しく説明しているように、Tableau は、Tableau ライセンスの目的上、展開における単一の本番環境をサポートするために、2 つの非本番環境に対して標準的な権限を付与

することに留意してください。Tableau ナレッジ ベース記事「Tableau Server テスト環境のライセンス発行」で説明されているように、これらの非本番環境は、テスト環境の使用目的と一貫したこの方法を実行するために停止および使用できるようにする必要があります。

コンテンツを本番環境から非本番環境 (テスト環境など) のグリーン インストールにコピーすることで、Tableau Server のアップグレードされたインスタンスを作成できます。この場合、ダウンタイムを最小限に抑えながら、必要に応じて元の本番インストールにフォールバックできるセーフティネットを確保できます。グリーン環境のすべての重要なコンテンツが機能していることを確認したら、ユーザーを新しい環境 (グリーン) に切り替えます。グリーン環境が本番環境になり、次のアップグレードをサポートするために、元のインストール (ブルー環境) を非本番環境として維持できます。

以下は、Tableau Server をアップグレードするためのアプローチとしてブルー/グリーンを使用するために必要な手順の概要です。これらのステップでは、ユーザー トラフィックをあるインストールから別のインストールにリダイレクトできるロード バランサーまたは DNS があり、アップグレードする Tableau の新しいバージョンをダウンロードしていることを前提としています。

重要: ブルー/グリーン インストールは、本番環境として扱う必要があります。これらは、災害復旧または一般的なテスト環境として使用することはできません。災害復旧の詳細については、Tableau Blueprint の「災害復旧」を参照してください。

1. Tableau Server の本番インストールを可能な限り反映した、サーバーの 2 番目のインストールを作成します。この 2 番目の環境は、グリーン インストール、つまり非本番環境です。グリーン インスタンスはアップグレード後に本番インスタンスになるため、グリーンには、本番環境 (「ブルー」) と同様の容量、リソース、構成が必要です。
2. Tableau Server のブルー インストールの初期バックアップと設定のエクスポートを作成します。これらは、グリーンで初期のユーザー受け入れテスト (UAT) に使用します。後で、実際にブルーからグリーンに切り替える際に使用する最終的な最新のバックアップとエクスポートを作成します。
    a. バックアップ: バックアップの作成に関する詳細については、Tableau Server データのバックアップを参照してください。

b. 設定のエクスポート: バックアップとともに、ブルー インストールから設定をエクスポートする必要もあります。一部の設定はグリーン環境で手動により作成し直す必要があります。詳細については、手動プロセスが必要なバックアップ アセットを参照してください。

3. 新しいバージョンの Tableau Server をグリーンにインストールします。ブルーで使用するのと同じアイデンティティ ストア タイプと実行サービス アカウントを指定する必要があります。

4. 初期のブルー バックアップと設定のエクスポートを使用して、グリーン インストールを更新します。

   a. ブルー環境のバックアップをグリーン インストールで復元します。

   b. ブルー インストールからエクスポートした設定をインポートします。エクスポートした .json ファイルは、常に直接インポートできるわけではありません。おそらく、セクションを編集または削除する必要があります。メジャー バージョンのアップグレードとアップグレードの間では、手動による変更が増えることが予想されます。たとえば、トポロジ情報が不完全であるため、Tableau Server の構成が機能しなくなる場合があります。コア Tableau Server 機能の復元を参照してください。

5. テスト中に重複メッセージを送信したり、データベース リソースが過負荷になったりしないように、グリーンのスケジュールされたタスクを無効にします。詳細については、サーバー アップグレード- スケジュールされたタスクの無効化を参照してください。

6. グリーン インストールをテストして、期待どおりに機能していることを確認します。ユーザー受け入れテスト(UAT) を実行し、必要に応じて負荷テストをシミュレートします。スクリプトまたは API を Tableau と統合している場合は、これらもテストする必要があります。テストが完全であればあるほど、元のブルー インストールと新しいグリーン インストール間の不一致に対処するための準備が整います。

7. ブルー インストールを準備し、バックアップを作成してグリーンの本番環境に使用する設定をエクスポートできるようにします。これには、Tableau へのアクセスを制限するか、コンテンツに変更を加えないようにユーザーに指示したり、バックアップの作成後に行った変更が失わ

れることをユーザーに警告したりするメッセージを送信することが含まれる場合があります。

8. ブルーの最終バックアップを作成します。これは、グリーンを最新の本番コンテンツにするために使用するバックアップです。定期的なバックアップを行う場合は、バックアップの作成後にブルーで行われた変更や更新が失われることに注意して、最新の本番バックアップを使用できます。このバックアップを作成した後、本番環境で「変更の凍結」を行うことをお勧めします。これにより、新しいワークブックやデータソースが追加されたり、既存のコンテンツが変更されたりすることはなくなります。

9. 手動で取り扱う必要があるアセットに注意しながら、ブルー環境の設定をエクスポートします。詳細については、コア Tableau Server 機能の復元を参照してください。

10. ブルーの最終バックアップを復元し、ブルーの設定ファイルをグリーンにインポートします。必要に応じて手動で構成を変更します。

11. グリーンコンテンツの UAT/サニティーチェックを実行します。

12. 上記のステップ 5 で無効にしたジョブ、サブスクリプション、通知のスケジュールをもう一度有効にします。

13. ロードバランサーまたは DNS を使用してユーザーをグリーンに切り替え、ブルーへのアクセスをブロックします。

14. グリーンサーバーが期待どおりに機能していることを確認し、本番負荷の問題を監視します。Advanced Management がある場合は、Resource Monitoring Tool を使用してパフォーマンスを監視できます。

15. (オプション) 今後のアップグレードのためにブルー環境を維持しない場合は、ブルー環境のプロダクトキーを非アクティブ化してください。

    - ATR でアクティブ化された Tableau Server 2021.4 以降の場合は、「プロダクトキーの認証解除」を参照してください。
    - ATR でアクティブ化されていない、すべてのバージョンの Tableau Server の場合は、「プロダクトキーの認証解除」を参照してください。
    - ATR でアクティブ化された Tableau Server 2021.3 以前の場合は、「サーバー ATR ライセンスを別の Tableau Server ライセンスに移動する」を参照してください。

16. (オプション) 今後のアップグレードのためにブルー環境を維持しない場合は、ブルー環境を停止して、コミッション解除の準備をしてください。コミッション解除の詳細については、「コンピュータからTableau Server を削除する」を参照してください。

アップグレードのテスト

Tableau Server のアップグレードが現在の環境に与える影響を把握する最適な方法では、それをテストすることです。アップグレードがユーザーやサーバーにどのような影響を与えるかを把握することにより、ユーザーが驚くことのないよう、実際のアップグレードの前に計画や連絡を行うのに役立ちます。

Tableau Server テスト環境がある場合、アップグレードのテストに最適です。

Tableau Server アップグレードをテストする際には、次のシーケンスを推奨します。

1. テスト環境の準備
2. テスト環境のアップグレード
3. 既存の機能が作動することを確認する
4. パフォーマンスおよびユーザー受け入れテスト
5. 新機能のテスト
6. アップグレードに関する通信

テスト環境の準備

最初に、できるだけ本番環境を厳密に反映するテスト環境を作成します。アップグレード対象の実際の環境とテスト環境が近ければ近いほど、アップグレードによる影響をより正確に確認することができます。これには、同じまたは同等のハードウェアとオペレーティング システム、同じ認証オプションとネットワーク アクセスが含まれます。

テスト コンピューターまたは仮想マシンの準備が整ったら、次の手順に従ってテスト環境を作成します。

1. 既存の本番環境で、`tsm maintenance backup` コマンドを使用して Tableau Server のバックアップを作成します。

   詳細については、アップグレード前のバックアップの作成を参照してください。

2. テスト環境で、本番環境と同じバージョンの Tableau Server のコピーをインストールします。

   > **注:** 代替ダウンロードサイトから最新バージョンのセットアッププログラムをダウンロードできます。

3. `tsm maintenance restore` コマンドを使用して既存のデータベースデータを復元します。

   詳細については、バックアップからの復元を参照してください。

4. 既存の Tableau Server 構成を手動で複製します。

   環境の特定の側面を手動で構成する必要があるのは、Tableau データベースを復元する際に構成の詳細とカスタマイズが含まれないためです。

テスト環境のアップグレード

テスト環境をアップグレードするための適切な手順に従います。

- 2018.1 以降からのアップグレード(Linux)

すべてが期待どおりに機能していることを確認する

Tableau Server の新しいバージョンをテスト環境でインストールおよび構成したら、テストの準備が整います。基本的な機能に加え、組織が依存しているサーバーの特別な側面をテストすることをお勧めします。たとえば、組織が依存している重要なサブスクリプションがある場合は、それらをテストする必要があります。

検討が必要なテスト対象エリアがいくつかあります。

- **サーバー プロセス**。サーバー管理者として Tableau Server にサインインしてから、[サーバー ステータス] ページを開き、すべてのサービスとプロセスが期待どおりに実行していることを確認します (分散型インストールの場合はすべての追加ノードを含む)。

- **ユーザー アクセス**。Tableau Server ユーザーがサインインできることを確認します。通常のユーザー サインイン プロセスをテストします。一部のユーザーにテストに参加してもらい、期待どおりサインインでき、本番環境で彼らがアクセス権を持っている一部のコンテンツにアクセスできることを確認します。

- **ワークブックとデータ ソースのパブリッシュ**。ユーザーに Tableau Desktop からワークブックとデータ ソースをパブリッシュしてもらい、期待どおりに動作することを確認します。

- **パブリッシュされたワークブックの表示**。コンテキストを熟知したユーザーに、パブリッシュされたワークブックを確認させ、それらが期待どおりに表示されていることを確認してもらいます。Web ページに埋め込まれたビューをテストします (SharePoint ページなど)。

- **サブスクリプションと抽出の更新**。いくつかの抽出更新を手動で実行し、正常に完了することを確認します。スケジュールされた重要な抽出更新をいくつか手動で実行し、それらが期待どおりに完了することを確認します。

- **パーミッション**。ユーザーやコンテンツに対するパーミッションがまだ期待どおりに設定されていることを確認します。

- **コマンドライン ユーティリティと API**。該当する場合は、コマンドライン ユーティリティ (tsm および tabcmd)、および API を介したプログラム的アクセスをテストします。

パフォーマンスおよびユーザー受け入れテスト

Tabjolt、Replayer、および Scout のようなツールを使用して、テスト環境でパフォーマンスおよびユーザー受け入れテストを行います。これらのツールおよびその他のパフォーマンス テスト ツールの詳細については、パフォーマンス モニタリング ツールを参照してください。

新機能のテスト

アップグレード後のバージョンに付属する新しい機能や、現在持っているバージョンと新しいバージョンの間に追加された機能を確認します。環境に適用された機能のメリットに対するお客様の

理解を助けるにはどうすればよいかを検討してください。

新機能の詳細については、「Tableau Server ヘルプの新機能」を参照してください。

アップグレードに関する通信

アップグレードをスムーズに進める最適な方法は、アップグレードについて、およびアップグレードによる影響について事前に組織に知らせておくことです。ユーザーにテストを手伝ってもらった場合、テスト中に彼らが目にした変更を連絡してもらうことにより、ユーザーの体験を活用できます。また、実際にアップグレードを行う前にアップグレード後のバージョンを確認する必要がある重要な人物がいる場合、テスト環境へのユーザー アクセスを提供することもできます。

Tableau Server のインストールとアップグレードのトラブルシューティング

Tableau Server の一般的な問題を解決するには、このトピックの提案に従ってください。ステータスページに表示されるプロセス ステータスに基づいた追加のトラブルシューティングの手順については、サーバープロセスのトラブルシューティングを参照してください。

一般的なトラブルシューティングの手順

多くの Tableau Server の問題は、幾つかの基本的な手順で対応することができます。

1. Tableau Server を実行している各コンピューターで十分なディスク容量があることを確認します。ディスク容量が限られている場合、インストールの失敗、アップグレードの失敗、または Tableau Server の実行で問題が生じる可能性があります。

2. Tableau Server を再起動します。完全に開始していないプロセスに関連する問題は、Tableau Server を制御された方法で再起動することによって解決できます。Tableau Server を再起動するには、`tsm restart` コマンドを使用します。このコマンドでは、Tableau Server に関連付けられたすべてのプロセスを停止して、再起動します。

3. Tableau Server インデックスの再作成を実行します。インデックス作成に関連する問題は、Tableau Server インデックスの再作成を行うことによって解決できます。Tableau Server インデックスを再作成するには、`tsm maintenance reindex-search` コマンドを使用します。詳細については、下のTableau index および Tableau Server (2023.3 以降) の再構築を参照してください。

4. Tableau Server が実行されている PC を再起動します。データソース接続に関連するような問題は、サーバー PC を再起動して解決できます。

Tableau Server のインストールに関する全般的な問題

インストール ログの場所

インストール ログである `app-install.log` は `/var/opt/tableau/tableau_server/logs` にあります。

アップグレード ログである `app-upgrade.log` は `/var/opt/tableau/tableau_server/logs` にあります。

インストールの試行が何度も失敗する

Tableau Server のインストールを試みて、インストールに失敗する場合、`tableau-server-obliterate` スクリプトを実行して Tableau をコンピューターからクリーンアップしない限り、その後インストールを試みても失敗する可能性が高くなります。

インストールの試行に失敗すると、コンピューターはその後の試行も失敗する状態となり、前回のインストール試行に直接は関係がないように見えるエラーが残る可能性があります。考えられるエラーの 1 つは以下です。

```
Enabling and starting all services
+ services=(appzookeeper* tabadmincontroller* tabsvc*
licenseservice* fnplicenseservice* tabadminagent*
clientfileservice*)
+ systemctl_user enable appzookeeper_0.service
'tabadmincontroller*' 'tabsvc*' 'licenseservice*'
```

```
fnplicenseservice_0.service 'tabadminagent*' 'clientfileservice*'
++ id -ru a_tabadminpoc
+ local unprivileged_uid=222954
+ su -l a_tabadminpoc -c 'XDG_RUNTIME_DIR=/run/user/222954 systemctl
--user enable appzookeeper_0.service tabadmincontroller* tabsvc*
licenseservice* fnplicenseservice_0.service tabadminagent*
clientfileservice*'
Failed to execute operation: No such file or directory
```

この問題を修正するには、tableau-server-obliterate スクリプトを実行し、前回インストールを試みたときに残ったものをすべてクリーンアップしてから、コンピューターを再起動します。詳細については、tableau-server-obliterate スクリプトの実行を参照してください。

重要: 保持する Tableau (<file>.tsbak) のバックアップを作成 (新しいインストール場所に復元するなど) したら、そのファイルを別のコンピューターの安全な場所にコピーして、Tableau コンピューターをクリーンアップしても削除されないようにします。

ハードウェア要件のためインストールが失敗する

Tableau Server は、インストール先のコンピュータがハードウェアの最低要件を満たしていない場合はインストールできません。この要件は、Tableau Server をインストールしているすべてのコンピューターに適用されます。ハードウェアの最低要件の詳細については、Tableau Server の最小ハードウェア要件と推奨事項を参照してください。

CPU 要件が原因でインストールまたはアップグレード失敗する

バージョン 2020.4.0 以降、Tableau Server では、SSE4.2 および POPCNT 命令セットをサポートする CPU が必要です。これらの命令セットをサポートしていない CPU が搭載されているコンピューターでは、Tableau Server 2020.4.0 以降をインストールまたはアップグレードすることはできません。

新規インストール時、または既存のインストールのアップグレードの準備中に、次のエラーメッセージが表示されることがあります。

```
Your computer's processor doesn't meet the minimum requirements
that Tableau requires to install the software. If you are using a
VM, make sure Processor compatibility mode is off.
```

SSE4.2 および POPCNT 命令セットは 10 年以上にわたって使用されている一般的なセットであり、ほとんどの新しい CPU がサポートしていますが、仮想マシン (VM) に Tableau Server をインストールまたはアップグレードしようとしたときにプロセッサーの最小要件に関連するエラーが発生した場合は、プロセッサー互換モードが VM で有効になっている可能性があります。Tableau を VM に正常にインストールまたはアップグレードするには、プロセッサー互換モードがオフになっていることを確認してください。

Tableau Server のアップグレードに関する全般的な問題

アップグレードログの場所
アップグレードログである `app-upgrade.log` は `/var/opt/tableau/tableau_server/logs` にあります。

アップグレード後にマップが表示されないか、完全に表示されません

Tableau バージョン 2019.2 以降、マップのインターネットアクセス要件が変更されました。バージョン 2019.1.x 以前からバージョン 2019.2.x 以降にアップグレードしているときに、マップが期待どおりに表示されない場合は、お使いの環境がポート 443 で `mapsconfig.tableau.com` および `api.mapbox.com` にアクセスできるように構成されていることを確認してください。

バージョン 2019.1.x 以前では、`maps.tableausoftware.com` へのアクセスが必要でした。

インターネットアクセスの要件の詳細については、インターネットとの通信を参照してください。

アップグレードスクリプトエラー:"Tableau Server Version change validation failed. (Tableau Server のバージョン変更の検証に失敗しました。)"

アップグレード時に、以前のバージョンの `scripts.<version_code>` ディレクトリから `upgrade-tsm` スクリプトを実行すると、次のエラーが発生し、アップグレードが失敗します。

```
Tableau Server Version change validation failed.
Tableau Server <version> is already installed.
```

このエラーが発生した場合は、インストールしたバージョンの `scripts.<version_code>` ディレクトリに変更し、そこからスクリプトを実行します。

マルチノードをアップグレードし、追加ノードを初期化すると"Enter your credentials again (認証資格情報を再度入力してください)" というエラーで失敗する

Tableau Server のアップグレード時に追加ノードを初期化しようとすると、次のエラーが表示されます。

```
Enter your credentials again. The credentials you enter must provide
administrative access to the computer where you generated the
configuration file.
```

これは、ノードが接続できないか、最初のノードと通信していることを示します。これには以下の複数の理由が考えられます。

- 入力した認証資格情報が有効でないか、入力ミスがある。認証資格情報は、Tableau Server を最初にインストールしたコンピューターの管理パーミッションを持つユーザーの情報でなければなりません。bootstrap ファイルを作成したユーザーの認証資格情報を使用する必要はありませんが、これが確実に有効な認証資格情報となります。
- 追加しようとしているコンピューターのローカル ファイアウォールで最初のノードへの通信が許可されていない。詳細については、ローカル ファイアウォール構成を参照してください。

ディスク容量の不足によるアップグレードの失敗

Tableau Server のセットアップ プログラムを実行およびアップグレードするためのディスク容量が不足している場合、インストールは失敗します。必要なディスク容量は、リポジトリデータベースのサイズおよび抽出の数とサイズによって異なります。

ディスク容量を空けるには:

1. `tsm maintenance ziplogs` コマンドを使用してログのアーカイブのスナップショットを作成します。

   圧縮されたログ ファイルを作成した後、Tableau Server インストールとは別の安全な場所に保存します。

2. `tsm maintenance cleanup` コマンドを使って不必要なファイルをクリーンアップします。詳細については、不必要なファイルの削除を参照してください

RebuildSearchIndex ジョブでアップグレードが失敗する

バージョン 2020.1.x 以降、アップグレードの最後のステップで検索インデックスを再構築します。この時点ですべてのサービスはアップグレードされているため、このジョブが失敗した場合は、サーバーを手動でリセットできます。最初からやり直す必要はありません。

エラーは次のようになります。

```
An error occurred while rebuilding search index.
```

検索サーバーをリセットするには、以下を実行します。

1. 最初のノードで、ターミナル セッションを開きます。

   アップグレード スクリプトで新しいバージョンのシステム環境に更新するため、新しいのターミナル セッションである必要があります。

2. tsm maintenance reindex-search コマンド (2023.3 以降) または tsm maintenance reset-searchserver コマンド (2023.1 以前) を使用して、検索インデックスを再構築します。

2022.1 以降でアップグレードに失敗する

Tableau Server 2022.1 (またはそれ以降) をアップグレードした後、アップグレードプロセスの一部として Tableau Server のバックアップを復元すると、次のエラーが発生する可能性があります。

"Tableau Server は既定で新しいアイデンティティサービスのテーブルを使用するため、バックアップを復元できません。"

この問題は、Tableau Server 2022.1 (およびそれ以降) がバックアップで使用されるアイデンティティスキーマとは異なるアイデンティティスキーマを使用するために発生します。この問題を解決するには、「アイデンティティの移行に関する問題をトラブルシューティングする」を参照してください。

2020.4.0 以降でアップグレードが失敗する

バージョン 2020.4.0 以降では、Checkpoint Upgrade 機能を使用して、失敗したアップグレードを再試行できます。一般に、これは、Tableau Server ログ ファイルに慣れていて、ログを検索することをいとわない経験豊富なサーバー管理者や IT プロフェッショナルにとって役立ちます。この機能を使用すると、upgrade-tsm スクリプトを再実行できるので、失敗したすべてのアップグレードに役立ちます。スクリプトは最後に成功したステップから実行するため、時間を節約できます。経験のある人にとっては、ディスク領域の問題やアクセス権限の問題などを特定して修正し、アップグレードを再実行することができます。

バージョン 2020.4.0 以降へのアップグレードに失敗した場合は、次の手順でアップグレードを完了できるかもしれません。

- `upgrade-tsm` スクリプトをもう一度実行します。アップグレードの失敗は、アップグレードプロセス中にタイムアウトが発生した結果である場合があり、スクリプトを再実行すると、断続的または偶発的なタイミングの問題から抜け出せる可能性があります。これは、安全で簡単に実行できる手順です。スクリプトを再実行しても害はありません。最悪の場合、アップグレードは同じ時点で失敗しますが、その前の手順を実行する必要はありません。

  スクリプトは `\scripts` ディレクトリにあります:

  ```
  opt/tableau/tableau_server/packages/scripts.<version_
  code>/upgrade-tsm
  ```

  upgrade-tsm スクリプトを再実行しても Tableau Server のアップグレードに成功せず、Tableau Server のログにも問題がない場合は、さらに次のトラブルシューティングの手順を実行します。

- コマンド画面でスクリプトの出力を確認します。エラー メッセージは、アップグレードの失敗の原因を特定し、問題を解決する方法のアイデアを得るのに役立ちます。

- `app-upgrade.log` ファイルを確認します。コマンド ラインに表示されるエラーは `app-upgrade.log` ファイルにも表示され、多くの場合、より詳細に表示されます。

- tabadmincontroller.log ファイルを確認します。アップグレードの問題を上記の 2 つのインスタンスで簡単に識別できない場合は、ジョブに問題がある可能性があります。tabadmincontroller.log ファイルには、問題の診断に役立つ詳細情報が含まれている場合があります。

  注: ログ ファイルの場所については、Tableau Server ログとログ ファイルの場所を参照してください。

バックアップ/復元ファイルの場所にパーミッションの問題があるため、アップグレードに失敗する

2022.1.0 より前のバージョンの Tableau Server では、バックアップ/復元ファイルのファイルの場所に正しいパーミッションがない場合、アップグレード スクリプトは失敗し、バックアップ ファイルを読み取れない、または リポジトリを復元できないというエラーが発生します。

バージョン 2022.1 以降、アップグレード スクリプトは、アップグレードを開始する前に、バックアップ/復元ファイルのファイルの場所のパーミッションを確認します。これにより、新しいバージョンの Tableau Server へのアップグレード中に、その場所からのファイルの書き込みおよび読み取りを実行できます。

エラーは次のようになります。

```
The tableau user does not have permission to read the backup file:
<backup/restore basefilepath>.

Repository restore failed.
An error occurred during installation.
An error occurred while restoring repository.
```

TSM がバックアップと復元に使用する場所は、basefilepath.backuprestore 構成キーによって定義され、既定によりインストール プログラムが正しいパーミッションを使用して設定されますが、これらは組織の IT ルールによって影響を受けたり、自分で作成した場所に変更した場合に影響を受けたりする可能性があります。2022.1 以降で使用可能な新しいコマンドを使用すると、ファイルの作成直後にバックアップ/復元ファイルの場所に関するパーミッションを確認して、パーミッションに関連する問題を回避することができます。このコマンドの詳細については、「tsm maintenance validate-backup-basefilepath」を参照してください。

バックアップ/復元のファイル パスの詳細については、「tsm ファイル パス」を参照してください。

アップグレードは成功したが、パブリッシュ済みデータ ソースにアクセスできない

限定的な特定のシナリオでは、Tableau Server をバージョン 2021.3 から 2023.1 または 2023.3 の初期バージョンにアップグレードした後、既存のパブリッシュ済みデータ ソースへ接続または更新しようとすると、次のエラーで失敗します。

```
java.io.FileNotFoundException: Unable to fetch data from any other
host. This may indicate a lost or invalid folder.
```

これは次の場合に発生する可能性があります。

1. かつてバージョン 2021.3.x であった Tableau Server のインストールをアップグレードしようとしている (現在 2021.3 を実行しているか、以前に 2021.3 から 2022.x バージョンにアップグレードした場合が考えられます)

   および

2. そのインストールを 2023.1 または 2023.3 の初期バージョンにアップグレードしようとしている

影響なし

以下の状況では問題はありません。

- 2021.3 *以外からのアップグレード パス*
- 2023.1 または 2023.3 以外へのアップグレード パス
- 2023.1 および 2023.3 のすべての新規インストール

詳細

2024 年 9 月 16 日現在、問題のあるバージョンはすべてダウンロードサイトから削除されています。バージョン 2023.1.x または 2023.3.x にアップグレードする必要がある場合は、メンテナンス リリース バージョン 2023.1.16 以上、または 2023.3.9 以上にアップグレードしてください。

この問題の詳細については、「既知の問題」を参照してください。

共通設定のインポートに関する問題

設定ファイルをインポートすると、サービスが欠落しているために "not present on any node (ノードに存在しません)" 検証エラーが発生する

Tableau Server の新しいバージョンをインストールして、以前のバージョンから設定ファイルをインポートしてアップグレードする場合、`tsm settings import` コマンドの実行時にトポロジ検証エラーが発生する可能性があります。

これは、古いバージョンの Tableau Server から設定ファイルをエクスポートして新しいバージョンにインポートする際に、Tableau の 2 つのバージョン間で新しいサービスが追加されている場合に発生する可能性があります。

次のようなエラーが発生します (特定のサービスが異なる場合があります)。

```
>tsm settings import -f 20183-export.json

Pending topology set.
There are 1 topology validation errors/warnings.

Service 'elasticserver' is not present on any node in the cluster.
Service: Elastic Server
```

この問題を解決するには、欠落しているサービスを Tableau Server に追加します。

1. 検証エラーを生成したサービスの場合、インスタンス数が 1 のサービスを追加します。

   たとえば、Elastic Server がクラスタに存在しない場合は、検証エラー メッセージの最初の行に表示されるサービス名を使用して、プロセス インスタンス数を 1 に設定します。

   ```
   tsm topology set-process -n node1 -pr elasticserver -c 1
   ```

   エラーが発生した各サービスに対してこの手順を繰り返します。

2. 警告やエラーが表示されなくなると、保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

これで、設定が正常にインポートされるはずです。

設定ファイルをインポートすると"configuration value you specified does not match" (指定した構成値が一致しません) エラーが発生する

Tableau Server の新しいバージョンをインストールし、以前のバージョンから設定ファイルをインポートすると、`tsm settings import` コマンドの実行時に構成検証エラーが発生する場合があります。Tableau から削除された構成値が設定ファイルに含まれている場合、これらのエラーが発生する可能性があります。

次のようなエラーが発生します (構成キーが異なる場合があります)。

```
>tsm settings import -f 20183-export.json
Configuration error: At least one configuration value you specified
does not match a known configuration key. This applies to the
following keys: '[features.TsmConfigFileService]'
Use this parameter to override unknown key error: --force-keys
```

この問題を解決するには、インポートする設定ファイルを編集して、エラー内にある構成キーへの参照を削除します。

1. JSON 設定ファイルをコピーし、バックアップ用にコピーを保存します。
2. プレーンテキストエディターで JSON 設定ファイルを開きます。
3. キーを含む行全体を見つけて削除します。この例では、キーを含む行は `features.TsmConfigFileService` です。

   ```
   "configKeys" : {
     "config.version" : 19,
     "tabadmincontroller.port" : "8850",
     "endpoints.enabled" : false,
     "endpoints.health.enabled" : true,
     "features.TsmConfigFileService" : true,
     "tableau_projects.language" : "en",
   ```

   上記のコードは、エクスポートされた設定ファイルの小さなセクションの例であり、ファイルの内容全体を表すものではありません。
4. 設定ファイルを保存し、インポートし直します。

トポロジ検証に関連する追加のエラーが発生する場合があります。これらのエラーを解決する方法の詳細については、上記の設定ファイルをインポートすると、サービスが欠落しているために "not present on any node (ノードに存在しません)" 検証エラーが発生するを参照してください。

エラー:「調整サービスのインスタンスを直接変更することはできません」

このエラーは、次の 2 つの状況で発生する可能性があります。

- Tableau Server 設定ファイルを、設定ファイルのとは異なる調整サービス トポロジ持つインストールにインポートする場合
- `tsm topology set-process` コマンドを使用して調整サービスを構成しようとする場合

設定ファイルをインポートした後にこのエラーが表示される場合:

Tableau Server 設定ファイルには、ターゲット サーバーのとは異なる調整サービス トポロジがあります。これは、新しいバージョンをインストールし、以前のバージョンの設定ファイルをインポートすることによって Tableau Server をアップグレードすると、発生する可能性があります。ターゲット サーバーに調整サービス アンサンブルを明示的に展開していない場合、そのサーバーの最初のノードには調整サービスの単一のインスタンスが含まれます。

このエラーを修正するには、コマンド ラインから不一致を修正するか、設定のインポート ファイルを編集します。保留中の変更をすべて破棄して、インポート ファイルの設定に一致するようにターゲット コンピューターに調整サービスを展開し、設定ファイルをもう一度インポートすることもできます。

コマンド ラインからの不一致を修正するには、エラーが発生する各ノードに対して、tsm topology set-process コマンドを使用して、調整サービスのインスタンス数を元に戻します。

1. `tsm pending-changes list` コマンドを実行します。出力には、変更が加えられているノードが表示されます。
2. 調整サービスのカウントが変更された単一のノードまたは複数のノードを見つけます。

たとえば、設定ファイルでは node2 上に調整サービス インスタンスが存在しているものの、ターゲット システムではそのノード上に調整サービス インスタンスがない場合、node2 のカウントでは、設定ファイルのインポートによって 0 から 1 に変更されたことが表示されます。

```
C:\Windows\system32>tsm pending-changes list
Configuration
There are no pending configuration changes.
Topology
node2:
            Coordination Service
                                New Instance Count:1
                                Old Instance Count:0
```

3. `tsm topology set-process` コマンドを使用して、カウントを"古いインスタンス" 値に戻します。

   上記の例では、次のようになります。

   ```
   tsm topology set-process -n node2 -c 0 -pr "Coordination
   Service"
   ```

4. 変更された調整サービス インスタンスのカウントをリセットしたら、保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

調整サービスのプロセス カウントを手動で設定する際にエラーが表示される場合:

このエラーは、調整サービスを管理するための `tsm topology` コマンドではなく、`tsm topology set-process` コマンドを使用して、コーディネーション サービスを直接更新しようとした場合にも発生します。以下を試した場合:

1. 保留中の変更を破棄するには、`tsm pending-changes discard` コマンドを使用します。
2. 調整サービスを構成するための正しいコマンドを使用します。詳細については、調整サービス アンサンブルの展開を参照してください。

Tableau Server の起動

Tableau Server は完全に起動したことを確認できません

Tableau Server はスタートアップ時にすべてのコンポーネントが正常に起動したことを確認できないと報告する場合があります。次のメッセージが表示されます。"サービスのすべてのコンポーネントが正常に起動したことを確認できませんでした。"

起動後にこのメッセージが表示された場合、`tsm status -v` コマンドを使用して Tableau Server が正常に実行されていることを確認します。

ステータスが実行中 ("Status: RUNNING") であることを示している場合、サーバーは正常に起動されており、メッセージを無視することができます。ステータスが DEGRADED または STOPPED の場合は、次のセクションの「Tableau Server が起動しない」を参照してください。

Tableau Server が起動しない

Tableau Server が開始しないか機能が低下した状態で実行されている場合、コマンドプロンプトから`tsm restart` コマンドを実行します。これによって実行中のすべてのプロセスをシャットダウンして、Tableau Server を再起動します。

Tableau index および Tableau Server (2023.3 以降) の再構築

インデックス作成/検索サーバーのインデックスを再構築することによって解決できる問題

再構築が必要になるのは、以下のようなインデックスです。

- ユーザーがログインを試みるとサイトのリストが空になっている
- ユーザーがプロジェクトの選択を試みるとプロジェクトのリストが空になっている
- コンテンツがない (ワークブック、ビュー、ダッシュボード)
- 予期しない、または不正確なアラート (たとえば、"更新の失敗" 抽出を含まないワークブックのアラート)

このような状態になっている場合は、`tsm maintenance reindex-search` コマンドを使用して、インテックス作成/検索サーバーのインデックスをリセットして再構築します。

Tableau Server の検索 & 参照 (2023.1 以前の場合) のインデックス再作成

検索とブラウズの再インデックスによって解決する他の問題

再構築が必要なインデックスの症状には次が含まれます。

- ユーザーがログインを試みるとサイトのリストが空になっている
- ユーザーがプロジェクトの選択を試みるとプロジェクトのリストが空になっている
- コンテンツがない (ワークブック、ビュー、ダッシュボード)
- 予期しない、または不正確なアラート (たとえば、"更新の失敗" 抽出を含まないワークブックのアラート)

これらの症状が見られる場合、`tsm maintenance reset-searchserver` コマンドを使用して検索とブラウズインデックスを再構築します。

Tableau Server のライセンス認証

Tableau Server のライセンス認証の失敗

一部のインスタンスの Tableau Server のライセンス認証に失敗する場合があります。さまざまなエラー メッセージがありますが、中でも一般的なメッセージは次のとおりです。

- `An error has occurred`

また、より具体的なメッセージの例を以下に挙げます。

- `Function flxActCommonLicSpcPopulateFromTS returned error 50030, 71521,`
- `No license found for 'Tableau Server'`

この問題を解決するには、以下の解決方法を記載された順に試してください。

ライセンシング サーバーへアクセスできることを確認する

Tableau のライセンス発行サービスは、2018 年 10 月 6 日に新しいデータセンターに移行しました。つまり、licensing.tableau.com にアクセスするために特別な構成を必要とする (静的 IP の許

可リストなど) 環境では、Tableau プロダクトキーをライセンス認証、更新、または認証解除する前にアップデートが必要となります。

アクセスをテストするには、ブラウザーにライセンスサーバーの URL とポートを入力します。

```
https://licensing.tableau.com:443
```

および:

```
https://atr.licensing.tableau.com/_status/healthz
```

サーバーにアクセスできる場合は、最初のサーバーに対して "テスト成功" メッセージが表示され、2 番目のサーバーに対して "OK" というメッセージが表示されます。

Tableau Server は、ライセンス発行の目的で、次のインターネット上の場所に接続する必要があります。

- atr.licensing.tableau.com:443
- licensing.tableau.com:443
- register.tableau.com:443
- o.ss2.us
- s.ss2.us
- crt.rootca1.amazontrust.com
- crt.sca1b.amazontrust.com
- crt.sca0a.amazontrust.com
- crt.sca1a.amazontrust.com
- crt.sca2a.amazontrust.com
- crt.sca3a.amazontrust.com
- crt.sca4a.amazontrust.com

- *.digicert.com
- ocsp.*.amazontrust.com
- crl.*.amazontrust.com
- crt.rootg2.amazontrust.com

上記のドメインへのリクエストは、ポート80 またはポート443 で行われることがあります。ポート80 は、証明書の検証 (失効、証明書チェーンなど) に使用されます。ポート443 は SSL 接続に使用されます。

`ocsp.*.amazontrust.com` および `crl.*.amazontrust.com` ドメインへのリクエストは、証明書失効に関する情報であるため、Amazon によって管理されます。詳細については、「ACM certificate characteristics」を参照してください。Tableau を実行しているコンピューターの証明書信頼ストアに Amazon ルート証明書をインストールすることをお勧めします。Amazon ルート証明書をダウンロードしてインストールする方法については、Amazon Trust Services Web サイトの「認証局」を参照してください。

日付と時刻を確認します

最初の Tableau Server コンピューター上の日付と時刻が正しいことを確認してください。時計が現在の日付より早い日時に設定されていると、Tableau Server は起動できません。

プロダクトキーを強制的に再度読み取る

1. 最初の Tableau Server コンピューターで、sudo アクセス権を持つユーザーとしてログオンします。
2. Tableau Server の bin ディレクトリに移動します。既定では、次になります。

   ```
   /opt/tableau/tableau_server/packages/bin.<version_code>/
   ```

3. 次のコマンドを入力します。

   ```
   tsm stop
   ```

```
./lmreread
tsm start
```

信頼されているストレージのコンテンツを Tableau サポートに送信する

FlexNet ライセンス発行サービスがインストールされ、実行中で、エラーが表示されていない場合、Tableau プロダクトキー情報に問題がある可能性があります。この問題を解決するため、次の手順を完了して、信頼されているストレージに格納されているキー情報のファイルを作成します。

1. 最初の Tableau Server コンピューターで、sudo アクセス権を持つユーザーとしてログオンします。
2. 次のコマンドを入力します。

   ```
   serveractutil -view > <machine_name>-LicResults.txt
   ```

   これにより、現在のディレクトリに `<machine_name>-LicResults.txt` ファイルが作成されます。ログ ファイルの場所に対する書き込み権限がなく、エラーが表示される場合は、ファイルを作成する権限がある場所に変更してから、コマンドをもう一度実行します。
3. Tableau サポート (http://www.tableau.com/ja-jp/support/request) に連絡し、作成した `<machine_name>-LicResults.txt` ファイルを含めます。

tabcmd のインストールの問題

tabcmd を個別にインストールする

tabcmd は、Tableau Server をインストールするときに最初の Tableau Server ノードに自動的にインストールされますが、別のコンピューターで実行する場合は、tabcmd を個別にダウンロードしてインストールする必要があります。詳細については、tabcmd のインストールを参照してください。

Linux で tabcmd をインストールときの問題

Tabcmd を正しく実行するには、Java11 が必要です。RHEL のようなシステムでは、tabcmd のインストール時に依存関係としてインストールされます。Debian のようなシステムでは、Java 11 がまだインストールされていない場合は個別にインストールする必要があります。

2022 年 7 月の時点で、Debian ディストリビューションはサポートされなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください。

Java がインストールされていない

tabcmd をインストールするときに次のようなエラーが表示される場合は、Linux コンピューターに Java 11 がインストールされていることを確認します。

```
Cannot find 'java' in your PATH. Install 'java' and make sure it is
in your PATH to continue.
```

Java のバージョンが正しくない

次のようなエラーが表示された場合は、Java 11 がインストールされていることを確認します。

```
Exception in thread "main" java.lang.UnsupportedClassVersionError:
com/tableausoftware/tabcmd/Tabcmd : Unsupported major.minor version
52.0
```

または

```
*** Uncaught exception NoClassDefFoundError:
javax/xml/bind/JAXBException
*** See the logs for the stacktrace.
```

Tableau Server のアンインストール

アップグレードを行う前に Tableau をアンインストールしないでください。アップグレードの詳細については、「2018.1 以降からのアップグレード(Linux)」を参照してください。

同時に複数のバージョンの Tableau Server をインストールできるようになりました。これにより、既存のバージョンを実行しながらほとんどのアップグレードを実行できるようになるため、ユーザーへのダウンタイムおよび影響が軽減します。アップグレードが完了したら、以前のバージョンをアンインストールできます。これにより、ディスク領域を解放します。以前のバージョンをアンインストールする必要はありません。

この記事では、新しいバージョンへのアップグレード後に以前のバージョンをアンインストールする方法について説明しています。

Tableau Server のアンインストールと完全な削除

Tableau Server on Linux がサポートする主な「アンインストール」シナリオは次の 2 種類です。

- **Tableau Server のアンインストール:** Tableau Server を新しいバージョンに*アップグレードしたら*、以前のバージョンをアンインストールしてディスク領域を解放することができます。引き続き、Tableau のアンインストールに関する情報を参照してください。
- **Tableau Server の削除:** Tableau Server を完全にコンピューターから削除したい場合は、Tableau 提供のスクリプトを使用して Tableau Server およびすべての関連するファイルを削除できます。*これにより、全データおよびサーバー コンポーネントが削除されるので、PC を Tableau 前の状態にリセットしたい場合にだけ行ってください。*テクニカルサポートがインストールに関する問題のトラブルシューティングを行う際に、このステップが推奨される場合は実行する必要があるかもしれません。Tableau を削除する前に、データのバックアップを作成することをお勧めします。バックアップ ファイルは、コンピューター上の Tableau のインストールとは異なる安全な場所に保存してください。Tableau Server の完全な削除では、事前に他のバージョンをアンインストールすることはありません。スクリプトはコンピューターに存在するすべての既存のバージョンをアンインストールします。既存のバージョンをすでにアンインストール済で、Tableau を完全に削除したい場合、そうするためのスクリプトを一時的な場所で検索可能です。詳細については、コンピュータから Tableau Server を削除するを参照してください。

Tableau Server パッケージのアンインストール

この手順は、新しいバージョンの Tableau Server にアップグレードした後に、以前の Tableau Server バージョンのパッケージをアンインストールしてディスク容量を空けるために使用します。

1. environment.bash ファイルで、現在使用している Tableau Server のバージョンを確認します。コマンドプロンプトで、次のように入力します。

   ```
   grep TABLEAU_SERVER_DATA_DIR_VERSION /etc/opt/tableau/tableau_
   server/environment.bash
   ```

2. コンピュータにインストールされる Tableau Server パッケージのバージョンを決定します。

- CentOS を含む RHEL ライクなディストリビューションで、次のコマンドを実行します。

  ```
  yum list installed tableau-server"*"
  ```

- Ubuntu で、次のコマンドを実行します。

  ```
  apt list --installed tableau-server"*"
  ```

3. パッケージ マネージャーを使用して Tableau Server パッケージを削除します。

   - CentOS を含む RHEL ライクなディストリビューションで、次のコマンドを実行します。

     ```
     sudo yum remove tableau-server-<version>.x86_64
     ```

   - Ubuntu で次のコマンドを実行します。

     ```
     sudo apt-get purge tableau-server-<version>
     ```

誤ってアンインストールした Tableau Server パッケージの再インストール

現在実行中のバージョンの Tableau Server のパッケージはアンインストール*しないで*ください。サーバーが使用できなくなります。Tableau Server とそのファイルすべてを完全に削除するには、コンピュータから Tableau Server を削除するを参照してください。

Tableau Server パッケージの現在の Tableau Server インスタンスをアンインストールすると次の処理が行われます。

- /opt/tableau/tableau_server 以下のすべてのファイルが削除される。これらは変更されないインストール ファイルです。
- Tableau Server の各種サービスが停止および無効化される
- Tableau Server のすべてのサービスのサービス ファイルは保持される
- データ ファイルはそのまま残る

現在実行中のバージョンの Tableau Server のパッケージを誤ってアンインストールした場合は、この手順に従い、元の状態に戻してください。

実行中の Tableau Server のインスタンスをアンインストールした後に再インストールするには、以下を実行します。

1. 誤ってアンインストールしたバージョンのパッケージを再インストールします。
2. `initialize-tsm` を実行します。

コンピュータからTableau Server を削除する

警告: 以下の手順を行うと、Tableau Server on Linux は*完全に削除され*、`initialize-tsm` で作成されたユーザーおよびグループ、すべての関連データ、構成情報も削除されます。これには、`/etc/opt/tableau/tableau_server/environment.bash` で権限あり/権限なしユーザー(既定では `tsmagent` および `tableau`) と設定されているユーザーが所有する `/tmp` または `/var/tmp` 内のファイルが含まれます。以下のコマンドを実行していて、コンピューターがインターネットに接続されているときに、`-l` オプションを削除しなければ、Tableau Server のライセンスも非アクティブになります。

ディスク領域を空けるために特定の Tableau Server パッケージをアンインストールする場合は (アップグレード後など)、Tableau Server のアンインストールを参照してください。

Tableau Server の通常インストールの一環として、Tableau と関連ファイルをコンピューターから完全に削除する方法を示したスクリプトがインストールされます。このスクリプトは Tableau データ、構成、ログ ファイルが不要な場合にのみ、または Tableau テクニカル サポートと連携しており、インストールやアップグレードの試行に失敗して Tableau Server を再インストールする必要がある場合にのみ使用してください。obliterate スクリプトでは、Tableau Server で使用するドライバーを個別にインストールしても、そのドライバーが削除されることはありません。

`tableau-server-obliterate` スクリプトは、コンピューターから Tableau Server を完全に削除する場合に使用します。これは次のようないくつかの理由により必要となることがあります。

- コンピューターにインストールした Tableau Server が必要なくなった場合。`tableau-server-obliterate` スクリプトを使用して Tableau Server を完全に削除します。この状況で、コンピューターがインターネットに接続されている場合は、`-l` オプションを含めて Tableau Server のライセンを非アクティブにすることができます。

- Tableau Server インストール中の問題に対するトラブルシューティング - Tableau のインストール中に問題が発生した場合は、再インストールする前に tableau-server-obliterate スクリプトを使用してコンピューターから Tableau Server を完全に削除する必要があります。これにより、古い設定や状態 (/etc/opt/tableau/tableau_server/environment.bash ファイルなど) がクリーンアップされ、「クリーンな」コンピューター上に再インストールできるようになります。再インストールを実行すると、-l オプションを中止してコンピューターにライセンス情報を保持することができます。-l オプションを省略すると、Tableau Server を再インストールするときにライセンス認証する必要はありません。

tableau-server-obliterate の実行内容

tableau-server-obliterate スクリプトの目的は、コンピューターから Tableau Server を完全に削除することです。これにはファイル、システム設定、構成が含まれますが、指定している場合にはライセンス情報も含まれます。

tableau-server-obliterate を実行する場合、次の手順を実行します。

- インストールされているすべてのバージョンの Tableau Server でアンインストールが実行されます (yum erase または apt-get remove).
- データディレクトリの大部分の内容が削除されます (既定は /var/opt/tableau/tableau_server)。既定では、バックアップ ファイルとログ ファイルは保持されます。以下の「Tableau Server のバックアップ ファイルとログ ファイルを保持する」セクションを参照してください。
- すべてのセマフォと共有メモリセグメントが削除されます。
- "tableau" ユーザーが所有するすべての一時ファイルが /tmp および /var/tmp から削除されます。
- インストール中に作成されたすべてのユーザーおよびグループが削除されます。
- /etc/opt/tableau が削除されます。
- 信頼できる証明書が /etc/pki/ca-trust/source/anchors/TableauServer および /usr/share/ca-certificates/tableau から削除されます。
- 構成ファイルが次の場所から削除されます。
    - /etc/sysctl.d/99-tableau-server.conf
    - /etc/profile.d/tableau_server*
    - /etc/security/limits.d/99-tableau_server*
    - /etc/systemd/logind.conf.d/tableau_server*
    - /usr/share/bash-completion/completions/tsm

    - etc/bash_completion.d/tsm
    - /usr/share/bash-completion/completions/tabcmd
    - /etc/bash_completion.d/tabcmd
    - /run/tableau
    - /usr/lib/tmpfiles.d/tableau-server.conf
- コンピューターがインターネットに接続されている状態で `-l` オプションを使用すると、すべてのサーバー ライセンスが非アクティブになります。このオプションはオフラインでは機能しません。

Tableau Server のバックアップとログファイルを保持する

2020.1 以前のバージョンの Tableau Server では、`tableau-server-obliterate` スクリプトを実行するとデータディレクトリの内容がすべて削除されました。バージョン 2020.1 以降では、`tableau-server-obliterate` スクリプトの既定の動作が変更されました。Tableau Server のバックアップ ファイルとログ ファイルが `logs-temp` ディレクトリにコピーされて保存されます。logs-temp ディレクトリの既定の場所は、`/var/opt/tableau/logs-temp` です。この新しい機能は、スクリプトでオプションを設定することによって無効にすることもできます。

既定の動作を変更してバックアップファイルまたはログファイルを削除するには、`tableau-server-obliterate` スクリプトの実行時に次のいずれかのオプションを含めます。

- `-k` logs-temp ディレクトリにバックアップをコピーしません。
- `-g` logs-temp ディレクトリにログをコピーしません。
- `-a` logs-temp ディレクトリに何もコピーしません。

tableau-server-obliterate スクリプトの実行

ライセンス情報を保持するか、ライセンス情報をその他のすべての Tableau Server の側面とともに削除して、Tableau Server をコンピューターから完全に削除できます。Tableau Server を同じコンピューターに再インストールする場合は、ライセンスを保持しておくこともできます。

以前のバージョンの `tableau-server-obliterate` では、新しいバージョンの Tableau Server のファイルが認識されない場合があります。コンピューターにインストールされている Tableau Server のバージョンに対応した obliterate スクリプトを常に実行してください。

サーバー ライセンスを削除せずに Tableau Server を完全に削除するには

この手順のスクリプト例には、Tableau Server のバックアップ ファイルとログ ファイルを削除する -a オプションも含まれています。

> **注:** Tableau Server を再インストールして、実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する場合は、Server ATR を使用して Tableau Server を再インストールおよびアクティブ化する前に、ライセンス情報を削除することをお勧めします。

1. 最初のノードで、ターミナル セッションを開きます。
2. スクリプト `tableau-server-obliterate` を実行します。

   ```
   sudo /opt/tableau/tableau_server/packages/scripts.<version_code>/tableau-server-obliterate -a -y -y -y
   ```

   > **注:** Tableau Server をアンインストール済みでこれから削除する場合は、`/var/tmp` ディレクトリにある `tableau-server-obliterate` スクリプトのコピーを実行します。スクリプトがこのディレクトリに存在しない場合は、Tableau Server パッケージを再インストールしてから `tableau-server-obliterate` スクリプトを実行してください。

   Tableau Server をマルチノードに (分散) インストールしている場合は、クラスタ内の各ノードでスクリプト `tableau-server-obliterate` を実行します。
3. `tableau-server-obliterate` スクリプトを実行した各コンピューターを再起動します。

Tableau Server とライセンスを完全に削除するには

この手順のスクリプト例には、Tableau Server のバックアップ ファイルとログ ファイルを削除する -a オプションも含まれています。

1. 最初のノードで、ターミナル セッションを開きます。
2. ライセンス認証されているプロダクト キーの認証を解除します。

```
tsm licenses deactivate -k <product_key>
```

3. スクリプトtableau-server-obliterate を実行します。

   ```
   sudo /opt/tableau/tableau_server/packages/scripts.<version_
   code>/tableau-server-obliterate -a -y -y -y -l
   ```

   コンピューターがインターネットに接続されているときに、-l このオプションを選択すると、コンピューターからすべてのライセンス ファイルが削除されます。スクリプトでは最初にすべてのアクティブなライセンスの認証解除を試行しますが、ライセンス認証の解除が正常に実行されたかどうかにかかわらず、すべてのライセンス情報が削除されます。このため、obliterate スクリプトを実行する前に tsm licenses deactivate コマンドを実行することが推奨されます。

   > **注:** Tableau Server をアンインストール済みでこれから削除する場合は、/var/tmp ディレクトリにある tableau-server-obliterate スクリプトのコピーを実行します。スクリプトがこのディレクトリに存在しない場合は、Tableau Server パッケージを再インストールしてから tableau-server-obliterate スクリプトを実行してください。

   Tableau Server をマルチノードに (分散) インストールしている場合は、クラスタ内の各ノードでスクリプトtableau-server-obliterate を実行します。追加ノードでライセンス認証を解除する必要はありません。

4. tableau-server-obliterate スクリプトを実行した各コンピューターを再起動します。

tableau-server-obliterate スクリプトのヘルプ出力

以下のヘルプ コンテンツは、次のコマンドを実行したときに出力されます。

```
sudo ./tableau-server-obliterate -h
```

./tableau-server-obliterate スクリプトは、/opt/tableau/tableau_server/packages/scripts.<version_code>/ にインストールされています。

出力

```
Remove Tableau Server from this computer.

This script will stop and remove all Tableau Services from this
computer. It also removes data and configuration files. It leaves
licensing in place. It also preserves logs and backup files, which
are moved to a temp directory under the Tableau data folder. You can
force removal of these files, and licensing, using optional
parameters.

This script is destructive and not reversible. It should only
be used to clean Tableau Server from a computer. For multi-node
installations, you must run the script separately on each node.

This script must be run as the root user.


-y              Required. Yes, remove Tableau Server from this
computer.
                Must be specified three times to confirm.
-l              Optional. Delete licensing files and data. This command
                will attempt to deactivate licenses before deleting
                licensing data. Internet access is required for license
                deactivation. Offline deactivation is not supported.
                To deactivate license before removing Tableau Server,
                run 'tsm licenses deactivate' before running this script.
-k              Optional. Do not copy backups to logs_temp directory.
-g              Optional. Do not copy logs to logs_temp directory.
-a              Optional. Do not copy anything to logs_temp directory.
```

移行

このセクションのトピックでは、Tableau Server から Tableau Cloud に移行する方法について説明します。このトピックでは、Tableau Server から Tableau Server への移行シナリオについても取り上げます。

> Tableau Server on Windows については、「移行」を参照してください。

Tableau Server から Tableau Cloud への移行 **480**

サーバーからサーバーへの移行 **491**

Tableau Server から Tableau Cloud への移行

小規模 (100 人未満のユーザー) の Tableau Server からクラウドに移行する方法については、「Tableau Cloud 手動移行ガイド」を参照してください。

Tableau Cloud への移行計画については、「Tableau Blueprint: Tableau のクラウド移行」を参照してください。

Tableau Server から Tableau Cloud に移行する際の技術的な考慮事項

このトピックでは、Tableau Server から Tableau Cloud への移行時に発生する可能性のある技術的な考慮事項を取り上げます。この記事は、現在 Tableau Server を管理していて、Tableau Cloud への移行を検討している管理者を対象としています。このトピックを移行計画の最初のステップとして考察することをお勧めします。また、アカウントチームと協力して、移行が組織にとって妥当な計画であるかどうかについても確認してください。

このトピックは、Tableau Server 2023.3 の製品機能を反映したものです。

追加情報を確認し、Tableau Cloud への移行を開始するには、Tableau.com の Tableau Cloud への移行 サイトにアクセスしてください。

概要

このトピックは、Tableau 管理者にとって馴染みのある 4 つの主要な領域で構成されています。次の表は、各主要領域における Tableau Cloud と Tableau Server の機能比較の概要を示しています。各機能の詳細については、このトピックの後半のセクションを参照してください。

セキュリティ、管理、ガバナンス		
	Tableau Cloud	**Tableau Server**
コンプライアンス	Salesforce Compliance ポータルを参照してください。このサイトにアクセスするには、アカウントマネージャーの許可が必要になる場合があります。	お客様による管理
ホスティングとアップグレード	Tableau による管理	お客様による管理
サイトレベルの管理	単一サイト(既定)	お客様による管理
テレメトリデータ	制限あり	お客様による管理
可観測性データ	管理者インサイトとアクティビティログ	リポジトリへのアクセス権を持ち、アクティビティログを使用するお客様による管理

データ接続		
	Tableau Cloud	**Tableau Server**
ファイル	サポート済み	サポート済み
アプリケーションとデータベース	サポート済み (データベース固	サポート済み

	有の制限あり)	
カスタム接続	限定サポート	サポート済み
キューブデータソース	サポートされていない	サポート済み

ユーザー管理とライセンス発行		
	Tableau Cloud	**Tableau Server**
ライセンス発行	内部: ロールベース 外部: ロールと使用ベース	内部: ロールベースとコアベース 外部: ロールベースとコアベース
SSO	サポート済み	サポート済み
Active Directory、Kerberos、LDAP ディレクトリ	サポートされていない	サポート済み
ユーザーとグループの自動管理	Okta、OneLogin、Azure Ad、カスタム ツール	Active Directory、LDAP

拡張性と外部インテグレーション		
	Tableau Cloud	**Tableau Server**
API を介した自動ツール	サポート済み	サポート済み
R とPython の統合	サポート済み	サポート済み
埋め込み	サポート済み	サポート済み
カスタマイズ	限定サポート	サポート済み

セキュリティ、管理、ガバナンス

セキュリティとコンプライアンス

Tableau Cloud セキュリティ認証が組織の要件を満たしていることを確認してください。Tableau Cloud は ISO 27001/27017/27018 および SOC 2/3 に準拠しており、GDPR で概説されているデータプライバシー要件を遵守しています。認証の完全な最新リストを参照するには、Salesforce Compliance ポータルにアクセスしてください。このサイトにアクセスするには、アカウントマネージャーの許可が必要になる場合があります。

また、Tableau Cloud とSalesforce は、地域のデータセキュリティとプライバシーに関する最も厳格な基準を遵守しています。お客様のデータが、ホストされている地域から外部に送信されることはありません。Salesforce は責任共有モデルを採用しています。このモデルでは、当社はお客様と協力して、Salesforce とお客様の組織で適切なセキュリティおよびコンプライアンス管理がすべて有効になっていることを確認します。

ホスティングとアップグレード

Tableau は、お客様のために Tableau Cloud をホストおよび管理します。Tableau Cloud は常に最新バージョンの Tableau を実行しています。メジャーアップグレードはダウンタイムなしで完了します。ダウンタウンを最小限に抑えるために、定期的なメンテナンス (アップグレード、バックアップ、パフォーマンス調整などのアクティビティ) が実行されます。ダウンタイムが必要な場合は、事前に通知された時間帯にメンテナンスを行います。Tableau のメンテナンス スケジュールの詳細と、Tableau Trust からのメンテナンス通知にサインアップする方法については、「Tableau Cloud のシステム メンテナンス」を参照してください。

Tableau Cloud は Amazon Web Services で実行されます。サイトは、こちらにリストされている AWS リージョンにデプロイできます。

Tableau は、災害復旧に備えて環境のバックアップを毎日作成します。ただし、これらのバックアップはシステム全体を復元するためのものです。Tableau Cloud からファイルが削除または消去されている場合、復元するファイルのバックアップの維持はお客様の責任で行います。

サイトレベルの管理

Tableau Cloud の既定の導入では、Tableau をユーザーに導入するため単一のサイトが提供されます。単一サイトモデルによって、環境の管理が簡素化され、Blueprint で推奨されている包括的なガバナンスシナリオを実現できます。

プラットフォーム全体で一貫したパフォーマンスを維持するため、Tableau Cloud はサイトに容量制限を設けてサイトを制限します。容量のタイプと関連する許容量のリストについては、「Tableau Cloud サイトの容量」を参照してください。容量制限には、ストレージ全体の制限と、サイトで実行できるタスクの制限が含まれます。Tableau Server リポジトリに接続し、使用状況をサイトの容量制限と比較することで、ユースケースがこれらの制限の影響を受けるかどうかを評価できます。リポジトリへの接続の詳細については、「Tableau Server リポジトリでデータを収集する」を参照してください。

可観測性データ

Tableau Cloud では、Tableau Server リポジトリデータベースに直接接続することはできません。代わりに、サイト管理者は、事前に構築された可観測性ダッシュボードとデータソースが含まれる管理者インサイトと呼ばれるプロジェクトにアクセスすることができます。「管理者インサイトを使用したカスタム ビューの作成」を参照してください。管理者インサイトを使用すると、使用状況、パフォーマンス、ライセンス発行、ユーザー管理などを容易に把握できるようになります。また、提供されたデータソースを使用してカスタムレポートを作成したり、このデータに基づいて構築されたアクセラレーターを活用したりすることもできます。たとえば、ダッシュボードの読み込み時間アクセラレーターを参照してください。管理者インサイトの既定のデータ保持期間は 90 日間です。Advanced Management を使用すると、保存期間が 365 日延長されるほか、アクティビティログにアクセスして、より詳細な可観測性データを取得することができます。

ビューの読み込みパフォーマンス

ビューの読み込みパフォーマンスは、サーバー環境の構成やリソースの可用性の違いにより、Tableau Cloud と Tableau Server で異なる場合があります。このパフォーマンスは、次のような多くの要因の影響を受けます。

- **マルチテナンシー:** Tableau Cloud は、リソースがポッド上のすべてのサイトで共有されるマルチテナント環境です。

- **最適化されたサーバー構成:** Tableau Server のインフラストラクチャと構成はコンテンツのパフォーマンスに合わせて調整できますが、Tableau Cloud の設定の大部分は事前に設定されています。
- **リソース容量:** 使用可能な CPU、RAM、およびストレージの読み取り/書き込みの速度は、解析、クエリ、レンダリングにかかる時間に影響を与えます。
- **ビューへのアクセス量:** 要求の量は、リソースの使用率とビューの読み込みに関連するプロセスの負荷に直接影響を与えます。
- **同僚ユーザーによるアクティビティ:** 同じポッド上で他のユーザーがリソースを大量に消費するタスクを実行すると、パフォーマンスに影響を与える可能性があります。

データ接続

Tableau Cloud で次の 2 つの接続オプションのうちどちらを使用するかは、データの保存場所によって決まります。

- Tableau Cloud は、以下にリストされているクラウドデータソースに直接接続することができます。直接接続の場合、一部のデータベースでは、接続前に Tableau Cloud からこれらのデータソースにアクセスするための承認が必要になります。「Tableau Cloud にパブリッシュされたクラウドデータへのアクセスの承認」を参照してください。

サポート済み Tableau Cloud コネクタ

Alibaba AnalyticsDB for MySQL‡	Dropbox*‡	OData‡
Alibaba Data Lake Analytics‡	Esri コネクタ‡	OneDrive*‡
Amazon Athena‡	Exasol‡	Oracle‡
Amazon Aurora for MySQL‡	Google BigQuery*‡	Pivotal Greenplum Database‡
Amazon EMR Hadoop Hive‡	Google Cloud SQL (MySQL 互換)‡§	PostgreSQL‡
Amazon Redshift‡	Google ドライブ‡	Presto‡
Apache Drill‡	Hortonworks Hadoop Hive	Qubole Presto‡

Azure Data Lake Storage Gen2‡	Impala‡	Salesforce‡
Azure Synapse Analytics (SQL Server 互換)	Kyvos‡	SAP HANA (仮想接続のみ)‡
Box‡	MariaDB‡	SharePoint Lists‡
Cloudera Hadoop‡	Microsoft Azure SQL データベース‡	SingleStore (旧 MemSQL)‡
Databricks‡	Microsoft Azure Synapse Analytics‡	Snowflake‡
Datorama by Salesforce‡	Microsoft SQL Server‡	Spark SQL‡
Denodo‡	MongoDB BI Connector‡	Teradata**‡
Dremio by Dremio‡	MySQL‡	Vertica‡

*Tableau Cloud で Google BigQuery、OneDrive、Dropbox 接続に OAuth 2.0 標準を使用する方法の詳細については、「OAuth 接続」を参照してください。

** Teradata Web 作成では、現在クエリバンディング機能はサポートされていません。詳細については、「Teradata」を参照してください。

‡データ管理 をお持ちの場合は仮想接続をサポートします。詳細については、Tableau Cloud ヘルプの「仮想接続とデータポリシーについて」を参照してください。

§Tableau Cloud は、Google Cloud SQL を使用した SSL をサポートしていません。

- オンプレミスまたはプライベートクラウドネットワークに保存されているデータの場合は、Tableau Bridge をデプロイできます。Tableau Bridge でサポートされているコネクタとサポートされていないコネクタのリストについては、「Bridge を使用した接続」を参照してください。

更新に 10 分以上かかる埋め込みデータソース抽出がワークブックに含まれている場合は、各データソースをワークブックとは別にパブリッシュしてから、個別に更新することをお勧めします。こうすることで、実行時間の長いクエリのタイムアウトを回避できます。詳細については、「Bridge の更新パフォーマンスの最適化」および「データソースのパブリッシュ」を参照してください。

ファイル

Tableau Cloud ではさまざまなファイル タイプがサポートされています。Google Drive、Dropbox、Box、S3 などのクラウドプラットフォームでホストされているファイルには、Tableau Cloud ダイレクト コネクタを使用でき、ライブ接続と抽出接続の両方をサポートすることができます。Tableau Bridge では、抽出を行うためのファイルベースのデータ接続がサポートされていますが、ファイルベースのデータへのライブ接続はサポートされていません。

アプリケーションとデータベース

Tableau Cloud では、Tableau Server でサポートされているものとほぼ同じコネクタがサポートされていますが、いくつかの違いがあります。一部のデータベースでは、Tableau Cloud への直接接続を有効にするための特定の構成が必要です。「クラウドプラットフォームでホストされているデータへの直接接続の許可」を参照してください。

Tableau Cloud とTableau Bridge では、Kerberos を使用したデータベースへの認証がサポートされていません。

カスタム接続

Tableau Cloud は、すべてのお客様が同じデータベース コネクタを使用するように設計されているため、お客様固有の接続ソリューションはサポートされていません。ただし、Hyper API とTableau Bridge は例外で、この 2 つは、接続を拡張またはカスタマイズするために使用できます。Hyper API は、Tableau Cloud にパブリッシュできる抽出を作成できます。Tableau Bridge を使用すると、ODBC、JDBC、Web データ コネクタなどの一部の接続オプションを拡張できます。「Bridge を使用した接続」を参照してください。

Tableau Cloud または Tableau Bridge で Connector SDK (.taco) を使用して接続をカスタマイズすることはできません。

データ管理製品の一部として仮想接続を使用している場合は、必要なコネクタに固有の違いが生じる可能性があります。Tableau Cloud または Tableau Bridge がシナリオをサポートしていることを確認します。

キューブ データ ソース

Tableau Cloud ではキューブ データ ソースはサポートされていません。代わりに、キューブが構築されている参照元データベースに直接接続して、分析の柔軟性を高めることをお勧めします。

データ準備

Tableau Bridge では、Prep Conductor を使用した Tableau Prep フローの実行はサポートされていません。代わりに、KB 記事「Tableau Cloud でオンプレミス データを使用して Tableau Prep Conductor のフローを実行する方法」のプロセスに従って、データ準備を自動化します。

ライセンス発行とユーザー管理

ライセンス発行

Tableau Cloud では、サブスクリプションのロールベース プランによるライセンス発行がサポートされています。詳細については、「Tableau の価格」ページを参照してください。主な管理タスクを排除するために、Tableau Cloud は既定で、ログインベースのライセンス管理 (LBLM) を使用してユーザーにライセンスを発行します。従来の永久ライセンスとサーバー コア ベースのライセンス発行はサポートされていません。

Tableau Cloud では「ゲスト」ユーザーは提供されません。ただし、接続済みアプリを使用して、組織内の各種用途に利用できるコンテンツを提供することができます。「Tableau 接続済みアプリを使用してアプリケーションを統合する」を参照してください。

組織外のユーザーに Tableau Cloud のライセンスを発行している場合は、外部の使用事例に対して、使用ベースによるライセンス発行を利用できます。「Tableau 埋め込み分析の使用ベースによるライセンス発行で、収益を上げ、コストを削減する」を参照してください。

シングル サインオン

Tableau Cloud では、SAML と OIDC の両方によるシングル サインオン (SSO) がサポートされています。Okta、Azure AD、One Login、Salesforce、PingOne などとの直接インテグレーションがサポートされています。

Tableau Cloud にアクセスするすべてのユーザーは、多要素認証 (MFA) を利用して Tableau Cloud への認証を行う必要があります。

SSO プロバイダーを使用していない場合は、Tableau Cloud の組み込み認証タイプである TableauID with MFA を使用できます。SAML を使用してカスタム ソリューションを作成したり、接続済みアプリを使用して埋め込みソリューションにアクセスしたりすることができます。

詳細については、Tableau Cloud ヘルプの「認証」を参照してください。

Active Directory、Kerberos、LDAP

Tableau Cloud では、Active Directory との直接統合はサポートされていません。ただし、Azure AD で SAML を構成すると、Tableau Cloud と統合することができます。

さらに、ほとんどの IdP には Active Directory が統合されているため、Tableau Cloud での認証を円滑に行うことができます。例については、「Active Directory 統合 (Okta)」を参照してください。

Tableau Cloud では、Kerberos 認証や LDAP をアイデンティティストアとして使用することはできません。

ユーザーとグループの自動管理

Tableau Cloud は、ユーザー アイデンティティ情報の交換を自動化するために、クロスドメイン ID 管理システム (SCIM) を使用しています。Tableau Cloud では、いくつかの IdP 固有の構成を使用できます。「外部 ID プロバイダーを介したユーザープロビジョニングおよびグループ同期の自動化」を参照してください。

Tableau REST API および tabcmd 2.0 は、多くのユーザーとグループの自動化タスクをサポートします。

Tableau Cloud では、ユーザーやグループのプロビジョニングを行うための Active Directory との直接インテグレーションはサポートされていません。

拡張性と外部インテグレーション

自動化ツール

Tableau Cloud では、REST API、JavaScript API、メタデータ API、ダッシュボード API、Webhook、および認証済みユーザーの HTML ページへの埋め込みがサポートされています。詳細については、「Tableau 開発者用ツール」を参照してください。

移行前に使用事例を確認することをお勧めします。

Tableau Cloud の自動化ツール戦略を計画する際に注意すべき相違点がいくつかあります。

- Tableau Server では、サーバーレベルの管理を行うための REST API がサポートされています。Tableau Cloud ではサイトレベルの管理がサポートされています。
- Tableau Server では、Web データコネクタのグローバルリストがサポートされています。Tableau Cloud では、Bridge を使用して Web データコネクタの抽出更新を実行する必要があります。
- Tableau Exchange のコネクタは、Tableau Server でのみサポートされています。

Tableau Cloud でサポートされている API は、Tableau 個人用アクセストークン (PAT) による認証に最適化されています。Tableau Cloud への移行の一環として、PAT 認証を使用するようにツールをリファクタリングすることを計画します。「個人用アクセストークン」を参照してください。

拡張性

Tableau Cloud では、R および Python を使用する分析拡張機能がサポートされています。「分析拡張機能を使用した接続の構成」を参照してください。ただし、Tableau Bridge では分析拡張機能がサポートされていないため、これらの拡張機能は Tableau Bridge を必要としないデータソースとのみ互換性があります。

Tableau Cloud では、Tableau Cloud に作成またはパブリッシュされた Prep フロー内の R または Python のスクリプト手順はサポートされていません。

埋め込み

Tableau Cloud では、他のアプリケーションへの Tableau の埋め込みがサポートされています。埋め込み API、さまざまな SSO オプション、ユーザーとコンテンツ管理のための Tableau REST API がサポートされています。

Tableau Cloud での埋め込みと Tableau Server での埋め込みには、主に 2 つの違いがあります。

- Tableau Cloud では、埋め込みシナリオに対して、サブスクリプション ロールベースのライセンス モデルと使用ベースのライセンス モデルの両方がサポートされています。
- Tableau Cloud はシングルテナント モデルを使用します。データを分離する際には、さまざまな方法を使用できます。どのデータ分離戦略を採用するかは、ビジネス ニーズと埋め込みソリューションによって決まります。詳細については、以下の Tableau Cloud ヘルプ トピックを

参照してください。

- プロジェクトを使用したコンテンツへのアクセスの管理
- Tableau 接続済みアプリを使用してアプリケーションを統合する
- サイトユーザーの表示状態の管理
- パーミッション

カスタマイズ

Tableau Cloud では、言語、ロケール、カスタム ロゴ、カスタム プロジェクト画像などのユーザー エクスペリエンスの一部をカスタマイズすることができます。「サイト設定とコンテンツ設定のカスタマイズ」を参照してください。

Tableau Cloud では、カスタム URL、カスタム フォント、カスタム ウェルカム バナーはサポートされていません。

サーバーからサーバーへの移行

このセクションのトピックでは、Tableau Server のハードウェア、オペレーティング システム、およびクラウドプラットフォーム間の移行について取り上げます。

サイトを 1 つの Tableau Server から別の Tableau Server に移行するには、「サイトのエクスポートまたはインポート」を参照してください。

Tableau Content Migration Tool を使用して Tableau Server のプロジェクト間でコンテンツをコピーまたは移行するには、Tableau Content Migration Tool についてを参照してください。

新しいハードウェアへの移行

Tableau Server をコンピューター間で移行するには、次の手順を行います。これは、Tableau が実行されているコンピューターハードウェアをアップグレードする場合に行います。

次の手順は、Tableau Server のデータを、稼働中のコンピューターから Tableau Server がインストールされている新しいコンピューターに移動する方法を示しています。始める前に、バックアップの作成やマニュアル操作に必要なアセットの収集など、「アップグレードの準備」の手順を終えている

ことを確認します。詳細については、Tableau Server の完全なバックアップと復元を実行しますを参照してください。Tableau Server のデータと構成を復元するには、これらが必要です。

稼働中のサーバーと同じタイプのアイデンティティストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。

注: この操作には、TSM コマンドラインを使用して実行する必要のある手順が含まれています。

1. Tableau Server の元のインストールでプロダクトキーのライセンス認証を解除します。新しいコンピューターでライセンス認証できるよう、これを行う必要があります。詳細については、以下の「Tableau Server プロダクトキーのライセンス認証」を参照してください。
2. 新しいコンピューターへの Tableau Server のインストール、ライセンスのアクティブ化、初期設定の構成、および管理ユーザーの作成を行います。詳細については、Tableau Server のインストールと構成を参照してください。
3. バックアップファイル *.tsbak* を `basefilepath.backuprestore` 変数で指定したフォルダーの場所にコピーします。既定の場所は、`/var/opt/tableau/tableau_server/data/tabsvc/files/backups/` です。

   注: バックアップファイルの場所は変更することができます。詳細については、tsm ファイルパスを参照してください。

4. 次に、Tableau Server を停止します。

   ```
   tsm stop
   ```

5. 稼働中のデータを、`tsm maintenance restore` コマンドを使用して新しい Tableau Server インストールに復元します。

   ```
   tsm maintenance restore -f <filename>
   ```

   ここで、`<filename>` はステップ 3 でコピーした *.tsbak* ファイルの名前です。

   Tableau Server データの復元の詳細については、バックアップからの復元を参照してください。

6. サーバーを起動します。

   ```
   tsm start
   ```

7. **分散インストールのみ**: Tableau Server は、新しい Tableau Server クラスターに追加する新しいコンピューターにインストールします。手順については、追加のノードのインストールと構成を参照してください。

8. 古い PC でプロダクトキーのライセンス認証を解除していない場合、それを行う前に、新しい Tableau Server インストールをテストし、本番環境用の準備が整っていることを確認してください。詳細については、以下の「Tableau Server プロダクトキーのライセンス認証」を参照してください。

   > **注**: インターネットに接続していない場合、ライセンス認証解除プロセスを完了するために、オフラインでのライセンス認証ファイルを作成するよう求めるプロンプトが表示されます。手順については、オフラインでの Tableau Server の有効化を参照してください。

Tableau Server 製品のプロダクトキーのライセンス認証

Tableau Server の同じプロダクトキーを最大で 3 回ライセンス認証できます。これにより、サンドボックス環境や QA 環境で Tableau Server をテストでき、実稼働環境で Tableau を使用することもできます。ライセンス認証を最大限に行うには、同じコンピューターに Tableau を再インストールしない限り、Tableau Server をコンピューターから削除する際にプロダクトキーのライセンス認証を解除する必要があります。これにより、別のコンピューターでライセンス認証を行うことができます。たと

えば、あるコンピューターまたは VM から別のコンピューターに Tableau Server を移動する場合は、プロダクトキーのライセンス認証を解除します。その後、元のコンピューターからTableau を削除します。新しいコンピューターに Tableau をインストールするとき、競合を発生させずにキーをライセンス認証できます。ロールベースライセンスを使用する場合は、Tableau への管理者アクセス権を失う可能性がありますので、必ず Creator キーまたは Explorer キーをアクティブにしてください。同じコンピューターに再インストールするために Tableau Server を削除する場合は、キーのライセンス認証を解除する必要はありません。再インストールされた時点で Tableau はこのキーを使用します。たとえば、コンピューター上のあるドライブから同じコンピューターの別のドライブに Tableau を移動する場合などです。プロダクトキーのライセンス認証を解除する方法の詳細については、tsm licenses deactivateを参照してください。

Tableau Server の Windows から Linux への移行

Tableau Server on Windows を実行しているお客様は、既存の Tableau のインストールをバックアップして新しい Linux へのインストールで復元することで、Linux へ移行できます。このトピックでは、この移行に必要な手順について説明します。Tableau Server のベータ版から公式にリリースされたバージョンへの移行やアップグレードは実行できません。

Tableau Server on Windows からTableau Server on Linux への基本的な移行手順は、次のとおりです。

1. ステップ 1: 移行の計画を立てる - 移行を正常に完了するために必要なすべての情報の収集など、移行についての計画を立てます。この手順では、Tableau Server on Windows と Tableau Server on Linux の間で起こり得る違いについて把握する必要があります。

2. ステップ 2: バックアップを作成する - Tableau Server on Windows のバックアップを作成します。

3. ステップ 3: Tableau Server on Linux をインストールして Windows のバックアップを復元する - テスト環境で Tableau Server on Linux の新しいインスタンスをインストールします。これにより、移行をテストしてからWindows のバックアップを復元できるようになります。Windows のバックアップを復元することで Tableau のコンテンツ (ユーザー、プロジェクト、サイト、ワークブック、データソース) が復元されますが、カスタマイズ内容は復元されないため、Tableau

Server on Linux を構成して組織で必要とされる設定と一致させる必要があります。マルチノードインストールの場合は、ノードを追加して個別に構成する必要があります。

4. ステップ 4: Tableau Server on Linux のテストを行う- Tableau Server on Linux を試行して、コンテンツが期待通りになっているか、およびユーザーが Windows で実行しているすべての操作を実行できるかどうかを確認します。この手順では、Tableau Server on Windows と Tableau Server on Linux の間で異なる可能性があると特定された変更点について詳しく見ていきます。通知を活用できるか、今後の変更について通知できるかどうかをテストする際は、重要な利害関係者を含めるようにします。

5. ステップ 5: Tableau Server on Linux を本番環境にインストールし、Windows のバックアップを復元する- Tableau Server on Linux に必要な機能が十分含まれていることがわかったら、Linux の本番環境に Tableau をインストールし、Windows のバックアップを復元します。

ステップ 1: 移行の計画を立てる

Windows から Linux への移行を正常に完了するには、いくつかの事前準備が必要です。次の要件を満たす必要があります。

- **アイデンティティストア**: 稼働中のサーバーと同じタイプのアイデンティティストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。アイデンティティストアをローカル アイデンティティストアから外部のアイデンティティストアに変更するには、アイデンティティストアの変更を参照してください。

- **サーバー管理者**: Tableau Server ソース (Windows) 展開から、サーバー管理者権限を持つ少なくとも 1 つのユーザー アカウントのパスワードが必要です。このユーザーを、Linux に移行された展開でサーバー管理者として使用します。

- **Tableau Server on Linux のクリーン インストール**: Tableau Server on Linux をこのトピックの後でインストールする場合は、Windows で使用しているアイデンティティストアのタイプと同じタイプを必ず使用するようにしてください。また、ユーザーやコンテンツは作成しないでくだ

さい。Windows バックアップ ファイルを Tableau Server の Linux 展開に復元すると、すべてのユーザー データとコンテンツが Windows バックアップ ファイルから置き換えられます。

- **Windows と Linux の違いについて**: Tableau Server on Windows と Tableau Server on Linux の違いを確認し、理解してテスト環境で調べられるようにします。

    - カスタム フォントを使用する場合は、Linux コンピューターにインストールする必要が生じる場合があります。また、Windows とは表示が異なる場合があります。

    - Linux の接続オプションは、Tableau Server on Windows で使用できるオプションのサブセットです。利用可能な接続タイプと、必要な接続タイプがサポートされていることを確認します。

ステップ 2: バックアップを作成する

TSM を使用してバックアップを作成します。

Tableau Server on Windows から移行する場合は、次の手順を実行します。

1. Tableau Server on Windows を実行しているコンピューターにログオンします。
2. 管理者としてコマンドプロンプトを開きます。
3. 次のコマンドを実行します。

    ```
    tsm maintenance backup -f <filename> -d
    ```

    バックアップ ファイル名に日付を含めるには、-d フラグを追加します。

詳細については、Tableau Server on Windows のヘルプで、「Tableau Server データのバックアップ」を参照してください。

ステップ 3: Tableau Server on Linux をインストールして Windows のバックアップを復元する

テスト環境で、Tableau Server on Linux をインストールします。

- Tableau Server のインストールと構成の手順に従って、Tableau Server on Linux をインストールします。Windows の展開と同じアイデンティティ ストアを使用します。また、ユーザー

やコンテンツは作成しないでください。

Tableau Server on Windows のバックアップを復元します。

1. Tableau Server on Linux が稼動しているコンピューターに Windows バックアップ ファイルをコピーします。既定では、復元プロセスは次の場所でファイルを検索します。

   ```
   /var/opt/tableau/tableau_server/data/tabsvc/files/backups/
   ```

   場所を変更することができます。詳細については、tsm ファイル パスを参照してください。

2. 次のコマンドを実行して Tableau Server を停止します。

   ```
   tsm stop
   ```

3. 次のコマンドを実行してバックアップ ファイルからの復元を実行します。

   ```
   tsm maintenance restore -f <filename.tsbak>
   ```

   ここで、<filename.tsbak> はバックアップ ファイルの名前です。たとえば、tabserver-2017-10-20.tsbak のようになります。

   詳細については、「tsm maintenance restore」を参照してください。

4. 次のコマンドを実行して Tableau Server を起動します。

   ```
   tsm start
   ```

Tableau のコンテンツを復元した後で、Tableau Server を構成する必要が生じる場合があります。たとえば、マルチノード インストールから移行を実行している場合は、追加ノードを追加して構成する必要があります。また、画像やロゴ ファイルのコピーなど、Windows 上でカスタマイズを行っている場合はそのカスタマイズを構成する必要があります。

ステップ 4: Tableau Server on Linux のテストを行う

Tableau Server on Linux のインストールとWindows バックアップの復元が完了したら、新バージョンの Tableau Server をテストすることができます。復元プロセスでは Tableau のコンテンツのみが復元されているため、構成、トポロジ (追加ノードの追加など)、および Windows のインストール時に行ったカスタマイズを更新する必要が生じる場合があります。

Windows とLinux の間の違いについてや潜在的な問題の領域について理解しておく必要もあります。基本的な機能に加え、組織が依存しているサーバーの特別な側面をテストします。たとえば、組織が Tableau で使用する重要なデータソースがある場合があります。これらをテストして、期待通りに表示されることを確認します。

検討が必要なテスト対象エリアがいくつかあります。

- **ユーザー アクセス。**管理者を含む Tableau Server のユーザーがサインインできることを確認します。通常のユーザー サインイン プロセスをテストします。一部のユーザーにテストに参加してもらい、期待どおりサインインでき、本番環境で彼らがアクセス権を持っている一部のコンテンツにアクセスできることを確認します。

- **組み込み管理ビューの表示。**Tableau Server on Linux の今回のリリースでは、PostgreSQL ドライバーを手動でインストールする必要があります。また、ドライバーによっては管理ビューもインストールします。組み込みの管理ビューにアクセスし、必要なドライバーがインストールされていることを確認します。詳細については、管理ビューを参照してください。

- **データ ソースの可用性。**Tableau Server on Linux では、Tableau Server on Windows のデータ ソースのサブセットがサポートされています。組織で使用されるデータ ソースが Linux でサポートされていることを確認し、必要なドライバーをインストールする必要があります。サポートされるデータ ソースの詳細については、Tableau Server 技術仕様を参照してください。ドライバーのインストールの詳細については、データベース ドライバーを参照してください。

- **共有 ドライバーでのファイルベースのデータ ソースへのアクセス。**ネットワーク ドライブ上のデータ ソース (Excel ファイルなど) には、特別な操作を行って Linux からアクセスできるようにする必要があります。管理者または IT 部門がドライブをマウントし、これらのデータ ソースを使用してワークブックを更新する必要があります。ネットワーク ドライブへの接続の詳細については、Tableau コミュニティを参照してください。

- **パブリッシュされたワークブックの表示。**コンテンツを熟知したユーザーに、パブリッシュされたワークブックでそれらが期待どおりに表示されていることを確認してもらいます。Web ページに埋め込まれたビューをテストします (SharePoint ページなど)。特に、Windows とLinux で

どのようにフォントが異なるかを理解し、場合によっては一部のカスタム フォントを Linux コンピューターにインストールするか、Linux で使用できない場合は他のフォントに置き換える必要があります。フォントが異なるため、ダッシュボードのレイアウトも異なって表示される可能性があります。

- **サーバー プロセス**。サーバー管理者として Tableau Server にサインインしてから、[サーバー ステータス] ページを開き、すべてのサービスとプロセスが期待どおりに実行されていることを確認します。バージョン 10.5 での変更による意図的な変更点についても注意します。詳細については、Tableau Server データエンジンを参照してください。

- **ワークブックとデータ ソースのパブリッシュ**。ユーザーに Tableau Desktop からワークブックとデータ ソースをパブリッシュしてもらい、期待どおりに動作することを確認します。ドライバーをインストールして使用中のデータ ソース接続をサポートする必要が生じる場合があります。データベース ドライバーを参照してください。

- **サブスクリプションと抽出の更新**。いくつかの抽出更新を手動で実行し、正常に完了することを確認します。スケジュールされた重要な抽出更新をいくつか手動で実行し、それらが期待どおりに完了することを確認します。

- **パーミッション**。ユーザーやコンテンツに対するパーミッションがまだ期待どおりに設定されていることを確認します。

- **コマンドライン ユーティリティと API**。該当する場合は、コマンドライン ユーティリティ (tsm および tabcmd)、および API を介したプログラム的アクセスをテストします。

ステップ 5: Tableau Server on Linux を本番環境にインストールし、Windows のバックアップを復元する

テストが完了し、追加の変更が必要な領域を特定してユーザーに通知したら、本番環境に Tableau Server on Linux をインストールし、Windows のバックアップを復元する準備は完了です。これには、上記と同じ手順を実行します。

Tableau Server のオンプレミスのコンピューターからクラウド内の VM への移行

Tableau Server をデータセンター内のコンピューターからクラウド内の仮想マシン (VM) に移行できます。この移行の一環として、Tableau Server のさまざまなデータや構成設定をオンプレミスのコンピューターから、Tableau Server がインストールされているクラウド内の VM に移動する必要があります。

注: この操作には、TSM コマンドラインを使用して実行する必要のある手順が含まれています。

前提条件

Tableau Server をオンプレミスのコンピューターからクラウド内の VM に移行する前に、以下の前提条件を確認してください。

- アカウントがあり、クラウドプロバイダーで環境が設定されていること。詳細については、以下を参照してください。
    - Amazon Web Services に Tableau Server をインストールする
    - Google Cloud Platform 上での Tableau Server のインストール
    - Microsoft Azure に Tableau Server をインストールする
    - Alibaba Cloud 上での Tableau Server のインストール
- 移行の準備を行うには、「アップグレードの準備」および「Tableau Server データのバックアップ」をお読みください。

Tableau Server を Cloud 内の VM に移行するには

1. 抽出更新やワークブックのパブリッシュなど、実行中のバックグラウンドタスクがないことを確認します。Tableau Server がアイドル状態の場合、使用されているすべてのデータソース接続タイプに注意することで、クラウド内の VM の新しい Tableau Server に適切なドライバーをインストールできます。

2. Tableau Server を停止します。コマンドプロンプトで、次のように入力します。

   ```
   tsm stop
   ```

3. Tableau Server が停止したら、「アップグレードの準備」および「Tableau Server データのバックアップ」のステップに従い、完全なバックアップを実行します。バックアップは、Tableau Server データの復元に使用します。クラウド内の VM に Tableau Server データを復元するには、このバックアップを使用します。
4. バックアップをサーバー外に移動し、安全に保管します。たとえば、Amazon S3、Google ドライブなどのクラウドストレージにバックアップを保存できます。
5. オンプレミスサーバーで SSL を使用している場合は、SSL フォルダーのコピーを作成します。インフラストラクチャのさまざまな要素とクラウド内の VM 上にある新しい Tableau Server との間に通信の問題が発生しないように、ファイアウォール ルールやポートなどのセキュリティルールに注意するようにしてください。
6. オンプレミスの Tableau Server の IP アドレスを使用して DNS エントリをすべて削除し、クラウド内の VM 上にある Tableau Server に再割り当てできるようにします。
7. Tableau Server の推奨事項を満たすクラウドに VM を起動します。
8. Tableau Server をクラウド内の VM にインストールします。詳細については、以下を参照してください。

   - Amazon Web Services に Tableau Server をインストールする
   - Google Cloud Platform 上での Tableau Server のインストール
   - Microsoft Azure に Tableau Server をインストールする
   - Alibaba Cloud 上での Tableau Server のインストール

9. インストールが完了したら、SSL ファイルをクラウド内の VM 上にある Tableau Server にコピーし、新しいサーバーでバックアップを復元します。
10. クラウド内の VM 上で Tableau Server を構成し、SSH やポートに関連するものすべてをインストールします。

> **注:** データベースの IP アドレス/ポートを変更する場合、接続情報をすべて更新することが必要な場合があります。

11. バックアップ ファイル *.tsbak* を `basefilepath.backuprestore` 変数で指定したフォルダーの場所にコピーします。既定の場所は、`/var/opt/tableau/tableau_server/data/tabsvc/files/backups/` です。

    注: バックアップ ファイルの場所は変更することができます。詳細については、tsm ファイル パスを参照してください。

12. 次に、Tableau Server を停止します。コマンドプロンプトで、次のように入力します。

    ```
    tsm stop
    ```

13. 構成情報のない稼働中のデータを、新しい Tableau Server インストールに復元します。コマンドプロンプトで、次のように入力します。

    ```
    tsm maintenance restore -f <filename>
    ```

    ここで、`<filename>` はステップ 11 でコピーした *.tsbak* ファイルの名前です。

    Tableau Server データの復元の詳細については、バックアップからの復元を参照してください。

14. サーバーを起動します。コマンドプロンプトで、次のように入力します。

    ```
    tsm start
    ```

15. **分散インストールのみ**: Tableau Server は、クラウド内の新しい Tableau Server クラスタに追加する新しい VM にインストールします。詳細については、以下を参照してください。

    - 追加のノードのインストールと構成
    - Tableau Server を AWS に分散環境で自己展開
    - Tableau Server を分散環境の Google Cloud Platform で自己展開
    - Tableau Server on Microsoft Azure を分散環境で自己展開
    - Tableau Server を分散環境の Alibaba Cloud で自己展開

16. クラウド内の VM 上にある Tableau Server をテストし、期待通りに動作することを確認します。Tableau Server がクラウドで正常に動作している場合は、オンプレミスの Tableau Server のプロダクトキーを非アクティブ化し、同じプロダクトキーを使用してクラウド内の VM 上にある Tableau Server をアクティブ化できます。

    注: インターネットに接続していない場合、ライセンス認証解除プロセスを完了するために、オフラインでのライセンス認証ファイルを作成するよう求めるプロンプトが表示されます。詳細については、オフラインでの Tableau Server のライセンス認証解除を参照してください。

17. 同じ Tableau Server プロダクトキーをアクティブ化できるのは 3 回までです。稼働環境、テスト環境、および QA 環境でそれぞれ 1 回ずつです。Tableau Server の新規インストールをテストし、稼働準備が整ったことを確認したら、Tableau Server の以前の稼働バージョンを非アクティブ化し、アンインストールする必要があります。以前のバージョンを非アクティブにするには、tsm licenses deactivate を参照してください。

    注: インターネットに接続していない場合、ライセンス認証解除プロセスを完了するために、オフラインでのライセンス認証ファイルを作成するよう求めるプロンプトが表示されます。手順については、オフラインでの Tableau Server のライセンス認証解除を参照してください。

アイデンティティストアの変更

インフラストラクチャやビジネスの変更により、Tableau Server でのアイデンティティストアの変更が必要となる場合があります。アイデンティティストアには、ローカルと外部の 2 種類があります。Tableau Server をインストールした場合は、ローカルまたは外部のアイデンティティストアを構成しました。

ローカルのアイデンティティストアで Tableau Server を構成する場合は、すべてのユーザー情報およびグループ情報は Tableau Server リポジトリに格納されて管理されます。ローカル アイデンティティストアのシナリオでは、ユーザーおよびグループ用の外部のソースはありません。

外部のストアで Tableau Server を構成する場合は、すべてのユーザー情報およびグループ情報は外部ディレクトリサービスによって格納され管理されます。Tableau Server は、ローカルのユーザーとグループのコピーが Tableau Server リポジトリに存在できるように外部のアイデンティティストアと同期する必要があります。しかし、ユーザーとグループのすべてのデータの信頼できるソースは外部のアイデンティティストアです。外部のアイデンティティストアには、OpenLDAP や Active Directory などがあります。

Tableau アイデンティティストアの詳細については、アイデンティティストアを参照してください。

ローカル ストアから外部 ストアに変更したり、外部 ストアからローカル ストアに変更したりできます。いずれの場合も、アイデンティティストアのタイプを変更するには、次の手順を完了します。

1. Tableau Server をアンインストールし、再インストールします。完全なアンインストールとクリーン インストールの手順についてはこのトピックの最後に説明します。

2. コンテンツとパーミッションを復元します。

   これらの手順における"復元" という用語は、作成したバックアップを `TSM maintenance restore` コマンドによって復元することを指してはいません。ターゲットの Tableau Server とは異なるアイデンティティストアを使用する Tableau Server インスタンスで作成したバックアップ (`.tsbak`) を復元することはできません。バックアップは、元の Tableau Server 構成に戻す必要が生じた場合に使用できる最善の保護策です。

警告

Tableau Server でのインストール タイプの変更は複雑で、時間のかかるプロセスになる場合があります。ユーザーのデータのコンテンツの紛失や孤立を防ぐため、このプロセスを慎重に計画する必要があります。いずれの場合でも、ワークブックおよびデータ ソースに適用されるユーザー フィルターは、変更後に手動で更新する必要があります。

最も重要な点として、Tableau Server を再インストールした後にコンテンツとパーミッションを新しいアイデンティティストアに移行する方法を決定します。

コンテンツとパーミッションの復元方法

次のリストは、Tableau Server を再インストールした後にコンテンツとパーミッションを復元するための 2 つの方法について説明しています。環境要件に最も適した方法を選択してください。

- **方法 1: サイトのエクスポートとインポートを使用する —** この方法では、最初に、既存の展開で各サイトをエクスポートします。次に、新しいサーバーをインストールして、新しいアイデンティティストアのタイプ用に構成します。次に、新しいサーバーの既定サイトで新しいユーザーを作成します。最後に、元のサイトをすべてインポートします。インポート段階で、既定サイトで作成した新しいユーザーに元の ID をマッピングできます。

  > **注:** Tableau Server のインスタンス間でサイトを移行する場合、移行先サイトは、移行元サイトの Tableau Server のバージョンと同じかそれ以降の Tableau Server のバージョン上にある必要があります。移行元サイトと移行先サイトの両方が、サポートされているバージョンの Tableau Server からのものである必要があります。

  この方法はすべてのコンテンツとパーミッションを各サイトでエクスポートするため、組織にとっての優れた方法は、アイデンティティストアの変更が完了した後でコンテンツとパーミッションの厳密なレプリカを要求することです。組織によっては、認証を変更した結果としてアイデンティティストアの変更が必要となる場合があります。こうした場合、別のユーザー名構文の設定が新しいモデルでの要件となることが多くなります。この方法には元のユーザー名を新しいユーザー名にマッピングするプロセスが含まれ、それぞれのシナリオに柔軟性を提供します。

- **方法 2: フレッシュインストール、ユーザーがコンテンツを再パブリッシュする —** この方法では、Tableau Server の新しいバージョンをインストールし、設定中に新しいアイデンティティストアを選択します。また、新しいサイトを作成します。次に、ユーザーを作成して彼らにアクセス権を供与し、ユーザーはワークブックやデータソースを再パブリッシュします。他の方法とは異なり、この方法では、既存の Tableau Server インフラストラクチャを再使用しません。

  この方法は、非常に自動化された小さな展開とデータに精通したユーザーに最適です。管理上の観点から、この方法はコンテンツを積極的に移植しないため、最もシンプルです。た

だし、コンテンツのパブリッシュは完全にユーザーに依存しているため、コンテンツの一元化された監督が必要となる大きな組織では、この方法は成功しない場合があります。

ユーザー フィルター

ユーザー フィルターはドメイン固有のものです。そのため、Tableau Server のドメインが変わったり、認証タイプが変わったりすると、フィルターが期待どおりに機能しなくなります。ユーザー フィルターは Tableau Server で生成されますが、ユーザーが設定を行った後は、ワークブックおよびデータ ソースに保存されます。これらの方法のどれでアイデンティティストアを変更しても、ワークブックまたはデータ ソースのコンテンツは変更されません。

アイデンティティストアの変更を計画する際は、Tableau Desktop ですべてのワークブックおよびデータ ソースのユーザー フィルターを修正する最終作業を必ず計画に含めてください。

ユーザー名とTableau アイデンティティストア

方法 1 を使用している場合は、Tableau Server が Tableau アイデンティティストアでユーザー名をどのように保存するかを知っておくと役立ちます。Tableau は、Tableau Server のさまざまなサービスを使用してコンテンツ パーミッションやサイト メンバーシップを調節するリポジトリで、すべてのユーザー ID を保存します。一般的に、Active Directory 向けに構成されているアイデンティティストアではユーザー名が `domain\username` の形式で保存されています。一部の組織は UPN (`jsmith@domain.lan`) を使用しています。

一方、ローカルのアイデンティティストアを使用して Tableau Server を構成する組織は通常、標準の切り詰められたユーザー名 (`jsmith` など) を作成します。

すべての場合において、これらのユーザー名は Tableau アイデンティティストア内で一意のリテラル文字列にする必要があります。1 つのアイデンティティストア タイプから別のタイプに変更する場合は、ターゲットの認証である SSO またはユーザー プロビジョニング ソリューションで特定のユーザー名形式が必要となる場合があります。

そのため、すべてのパーミッション、コンテンツ、およびユーザーの実行可能性を保つため、アイデンティティストア タイプの変更後に次のいずれかが当てはまる必要があります。

- 新しいユーザー名は元のユーザー名と一致している必要があります。または
- 新しい書式設定に一致する用、元のユーザー名を更新する必要があります。

認証の変更によりアイデンティティストアの変更が引き起こされる場合、ターゲット認証スキームにより元のユーザー名と異なるユーザー名構文が使用される可能性があります。メソッド1には、元のユーザー名を新しいユーザー名にマッピングできるプロセスが含まれます。

元のユーザー名の書式設定が新しい認証タイプで機能する可能性があります。たとえば、ローカルアイデンティティストア展開でUPN名を使用している場合、Active Directory展開で同じユーザー名を使用できる場合があります。また、ユーザーがTableau Serverへのサインインに引き続き`domain\username`形式を使用している場合は、その形式をローカルアイデンティティストアに使用することもできます。

ローカルアイデンティティストアを外部Active Directoryストアに変更する場合は、計画プロセスの一部として、外部IDストアを使用した展開におけるユーザー管理トピックを見直してください。

方法 1: サイトのエクスポートとインポートを使用する

エクスポート操作とインポート操作には、同じバージョンのTableau Serverを使用する必要があります。

1. すべてのサイトをサーバーにエクスポートします。サイトのエクスポートまたはインポートを参照してください。
2. バックアップを作成し、削除してから再インストール
3. Tableau Serverで新規ユーザーを作成します。元のサーバーの各ユーザーに対応する新しいユーザーが必要です。
4. ステップ1でエクスポートしたサイトをインポートします。サイトのエクスポートまたはインポートを参照してください。インポート時、新しいユーザーを元のユーザーにマッピングするよう要求されます。

方法 2: フレッシュインストール – ユーザーがコンテンツを再パブリッシュする

アイデンティティストアの変更の一部としてコンテンツの移植を計画しない場合であっても、サーバーをバックアップすることをお勧めします。

1. バックアップを作成し、削除してから再インストール。
2. ユーザー、サイトおよびグループを作成します。
3. 新しいTableau Serverについてユーザーに通知し、新しい認証資格情報を提供してからコンテンツのパブリッシュを許可します。

バックアップを作成し、削除してから再インストール

どちらの方法にも、次のステップが含まれます。

1. Tableau Server のバックアップ
2. Tableau Server の削除。
3. 新しいアイデンティティストア タイプを使用した Tableau Server の再インストール。

ステップ 1: Tableau Server のバックアップ

ベストプラクティスとして、続行前にサーバーをバックアップする必要があります。

TSM コマンド ライン インターフェース (CLI) を使用したバックアップの作成の手順に従ってください。`-d` オプションを使用して `backup` コマンドを実行します。`-d` オプションによりデータスタンプが追加されます。

完了したら、バックアップ ファイル (`.tsbak`) を Tableau Server インストールの一部ではない安全な場所にコピーします。

ステップ 2: Tableau Server の削除

コンピューターから Tableau Server を完全に削除する必要があります。コンピュータから Tableau Server を削除するを参照してください。

ステップ 3: 新しい認証タイプを使用して Tableau Server を再インストールする

1. Tableau のカスタマー ポータルにサインインし、Tableau のユーザー名とパスワードを使ってサイン インしたら、Tableau Server をダウンロードします。
2. Tableau Server をインストールします。詳細については、Tableau Server のインストールと構成を参照してください。インストール時、新しいアイデンティティストア タイプを選択します。初期ノード設定の構成を参照してください。

個別のサイトの管理

Tableau でのサイトの計画に加えて、ユーザーとグループの管理、プロジェクトの管理、コンテンツへのアクセスの制御、サイトデータの管理、Web 上のビューの操作も行うことができます。

サイトとは 509

サイトの計画 512

サイト設定 リファレンス 515

ユーザーとグループの管理 534

ダッシュボード ベースのカスタム ポータル 608

コンテンツ アクセスの管理 611

サイトとは

これまで、サイトを「接続されているコンピューターの集まり」または「Web サイト」の略語として使用していた方もいるでしょう。Tableau では、サイトという言葉を、Tableau Server の同じインスタンス上の他のグループおよびコンテンツから壁で守られている、ユーザー、グループ、およびコンテンツ (ワークブック、データ ソース) の集まりを意味するために使用します。つまり、Tableau Server はマルチテナント機能をサポートしており、サーバー管理者はサーバー上で複数セットのユーザーやコンテンツ向けの複数のサイトを作成できます。

すべてのサーバー コンテンツは、サイト単位でパブリッシュ、アクセス、および管理されます。各サイトには専用の URL と専用のユーザー セットがあります (各サーバー ユーザーを複数のサイトに追加できる場合でも)。各サイトのコンテンツ (プロジェクト、ワークブック、およびデータ ソース) は、他のサイトのコンテンツから完全に隔離されます。

あなたがTableau Server展開のサーバー管理者なら、サイト、(プロジェクトに対して) いつサイトを使用するかなどについての詳細を、「**Manage Server** (サーバー管理)」セクションの「サイトの概要」で学ぶことができます。

注: この記事は、Tableau Server 展開上でのサイトの構成に関する内容を記載しています。Tableau Cloud については、「サイト管理者のロールとタスク」を参照してください。

サイト管理者のタスク

サーバー管理者サイトロールがユーザーに Tableau Server 展開全体への無制限のアクセスを与える場合、サイト管理者サイトロールは、ユーザーにサイトレベルで無制限または最小限の制限があるアクセスを与えます。サイト管理者 Creator とサイト管理者 Explorer の違いは、データ接続およびパブリッシュのアクセスレベルにあります。いずれのサイトロールもサイト自体の管理とサイトユーザーの管理を許可しています。詳細については、ユーザーのサイトロールの設定を参照してください。

サーバー管理者は、サーバーレベルとサイトレベルの両方で作業できますが、2 つのタスクレベルは区別されています。サイト管理者は、通常、組織内の Tableau ユーザーがデータソースおよびワークブックのパブリッシュ、共有、管理、およびそれらへの接続ができるようにするフレームワークを作成および管理する責任を担う担当者として定義されています。そのため、サイト管理者のタスクには以下のような作業が含まれます (どちらのサイトロールもこのレベルのアクセスを許可しています)。

- サイトのデータソースおよびワークブックを整理するための、プロジェクト階層の作成。

  これには、プロジェクトレベルの管理をプロジェクトリーダーに委託することを含めることもできます。

- ユーザーに必要なコンテンツのみへのアクセスを許可するための、グループの作成およびパーミッションの割り当て。

- ユーザーの追加と削除、サイトロールの割り当て。

  これは、サイトで既定で許可されています。ただし、サーバー管理者は、このアクセスをサーバーレベルのみに制限することができます。

- サイトの抽出およびサブスクリプション スケジュールの管理。
- サイト アクティビティの監視。

サーバー管理者とサイト管理者の違いについての詳細は、「**Manage Server (サーバー管理)**」セクションの サイトへの管理者レベルのアクセス権を参照してください。

サイトの設定手順

次の表は、サイト設定の手順の概略と、詳細情報が入手できるトピックへのリンクを示しています。組織にとって有意義な任意の順序で手順を完了できます。

ただし、サイトを構成する手順を実行する前に、このセクションの記事に多少時間を割いて、サイトの認証、サイト ロール、プロジェクト、およびパーミッションについて学習することをお勧めします。理想的には、プロジェクト、グループおよび全体のパーミッション戦略について計画を立案しドキュメント化してください。そして、いくつかのプロジェクトを設定して、予備的なユーザーのセットを追加し、計画をテストして、残りのユーザーを追加する前に問題を解決します。ユーザーがサイトで作業を開始した後でも多くのサイト設定を変更できますが、本稼働後の変更は最小限に留めるように心がけてください。

計画	この表の上の推奨事項に加えて、どのようにサイト コンポーネントが一緒に機能しているかについての概要をサイトの計画で確認してください。
アクセスの構成	サーバー管理者と作業し、ユーザーのサイトへのサインイン方法、サイトの適切な構成方法を決定します。 例えば、サーバーが SAML を使用したシングル サインオンで構成されている場合、サイトレベルでも SAML 認証を構成したいかもしれません。
プロジェクトの作成とパーミッション構造	プロジェクトは、コンテンツの整理、プロジェクト レベルのコンテンツ管理の委託、パーミッションの効果的な管理に役立ちます。始めるには、プロジェクトを使用したコンテンツへのアクセスの管理を参照してください。

ユーザーの追加	サイトにサインインできるユーザーを決定します。サイトへのユーザーの追加を参照してください。
データを Tableau Server に送る	プロジェクトとパーミッション構造作成後、吟味されたデータソースをサイト上の適切なプロジェクトにパブリッシュし管理するために、承認済みのユーザーを指定します。 組織によっては、1 人で複数の Tableau ロールを務める場合があります。通常、サイト管理者はデータ案内人でもあります。つまり、サイト管理者は Tableau データ接続の作成、パブリッシュ、および管理を行います。ご自分がサイト管理者である場合は、サイト管理者 Creator サイトロールが割り当てられていることを確認してください。 サイトにコンテンツがパブリッシュされたら、接続情報 (認証資格情報、アクセス トークン) を管理しスケジュールを更新することができます。詳細については、定期的なデータ更新を参照してください。
サイトの使用率とパフォーマンスの分析	使用、パフォーマンス、その他のメトリクスを監視します。管理ビューを参照してください。

サイトの計画

サイトにユーザーとコンテンツを追加する前に、サイトの次の側面を計画することをお勧めします。

- プロジェクト
- ユーザーとグループ

- サイトロールとパーミッション
- 抽出更新スケジュール

後続のセクションは、これらのサイトコンポーネントについては、精通していることを想定して飛ばしています。

> **注:** この記事とセクションは、オンプレミスまたはクラウドで使用している自己管理型の Tableau Server 展開のみに適用します。Tableau Cloud を使用する場合は、「コンテンツアクセスの管理」を参照してください。

プロジェクト

サイト上で、関連するコンテンツ資産 (データソースおよびワークブックなど) を整理するコンテナーの役割を果たすプロジェクトを作成することができます。例えば、組織内でミッションクリティカルな決定に使用する、認定済みデータソースとワークブックすべてを含むプロジェクトを設定することができます。または、部門ごとにプロジェクトを設定することもできます。

プロジェクトは、パーミッションの管理にも便利です。ユーザーのコンテンツへのアクセスのニーズがどのようなものかが分かれば、それらのコンテンツの種類に基づいて、プロジェクトは作りやすくなり、プロジェクトレベルでパーミッションを管理しやすくなります。

各サイトには、**既定**という名前の既定プロジェクトがあります。プロジェクトを作成すると、新しいプロジェクトは既定プロジェクトからパーミッションの初期セットを取得します。事実上、既定プロジェクトは新しいプロジェクトのテンプレートです。関連する記事で説明したとおり、殆どの環境で、既定プロジェクトはパーミッションのテンプレートとしてのみ使用することを推奨します。パブリッシュされたコンテンツのコンテナーとして使用することは推奨しません。

詳細は、プロジェクトを使用したコンテンツへのアクセスの管理を参照してください。

ユーザーとグループ

コンテンツをパブリッシュしたり、パブリッシュされたコンテンツにアクセスしたりするユーザーは、そのサイトにサインインできる必要があります。ユーザーが既にそのサーバーのアカウントを持っている場合

は、そのユーザーを適切なサイトに追加する必要があります。一人のユーザーを一つ以上のサイトに追加することができます。ユーザーがまだ存在しない場合は、ユーザー アカウントを作成する必要があります。いずれにしても、各サイトにサインインする必要があるユーザーのリストを作成します。

注: サーバー ライセンスによって、許容可能なユーザー数や持つことができるアクセス権のレベルが制限される場合があります。自分のすべてのユーザー用のアカウントを持っていることをサーバー管理者に確認します。

一般的に、サーバー上でグループを作成してから、それらのグループにユーザーを追加することをお勧めします。これによって、パーミッションの管理がずっと簡単になります。パーミッションをグループに割り当て、グループ内のすべてのユーザーにパーミッションを与えることができます。(次のセクションを参照してください。)

典型的な戦略は、コンテンツの使用方法が類似したユーザーのグループを作成することです。たとえば、ワークブックをパブリッシュする "Sales" 部門のすべてのユーザー用に「SalesWBPublishers」という名前のグループを作成し、データ ソースをパブリッシュする "Sales" 部門のすべてのユーザー用に「SalesDSPublishers」という名前の別のグループを作成できます。これらのユーザーのセットは、ニーズに合ったグループになるように、それぞれ独自の機能のセットが必要です。

サイト ロールとパーミッション

各ユーザーには、サイト上で持つことができる最大のパーミッションを決定する*サイト ロール*があります。サイト計画の一部として、各ユーザーのサイト ロールを決定する必要があります。非常に限定的なサイト ロールを持つユーザーは、必要な操作を実行できない場合があります。同じトークンで、セキュリティが最も高いプラクティスは、ユーザーの機能を仕事をするために必要なものだけに限定することです。これは、*最小権限*の原則と呼ばれます。

あなたか、あなたがこのタスクを委託するサイト管理者が、コンテンツで作業するためにユーザーが必要なパーミッションも決定する必要があります。各コンテンツ資産 (ワークブック、データ ソース、プロジェクト) が*機能*のセットをサポートします。たとえば、ワークブックの**閲覧**または**コメントの追加**ができます。ユーザーがワークブックでタスクを実行する前に、彼らのパーミッションでそれらの機能を

許可する必要があります。サーバーに設定する前に、Tableau 外部のユーザーへのパーミッションのマッピング概略を描いておくことを推奨します。

パーミッションは、*サイトロールのコンテキスト内*でユーザーが何をできるかを決定します。サイトロールが**Viewer**のユーザーは、パーミッションを付与するとしても、サイトにパブリッシュすることはできません。サイトロールが**Creator**のユーザーは、ワークブックを保存および表示するパーミッションが付与されている場合のみ、サイトにワークブックをパブリッシュできます。

抽出更新スケジュール

ユーザーが抽出を含むデータソースやワークブックをパブリッシュする場合、通常は、抽出を更新し、抽出に最新のデータが含まれるようにします。ユーザーは手動で抽出を更新できますが、抽出が大きく、更新に長い時間がかかる場合、これは必ずしも良い考えではありません。代わりに、抽出を更新するスケジュールを設定できます。したがって、サイト管理者が行うもう1 つの計画タスクは、抽出を更新する時期について考え、スケジュール設定に取り組むことです。

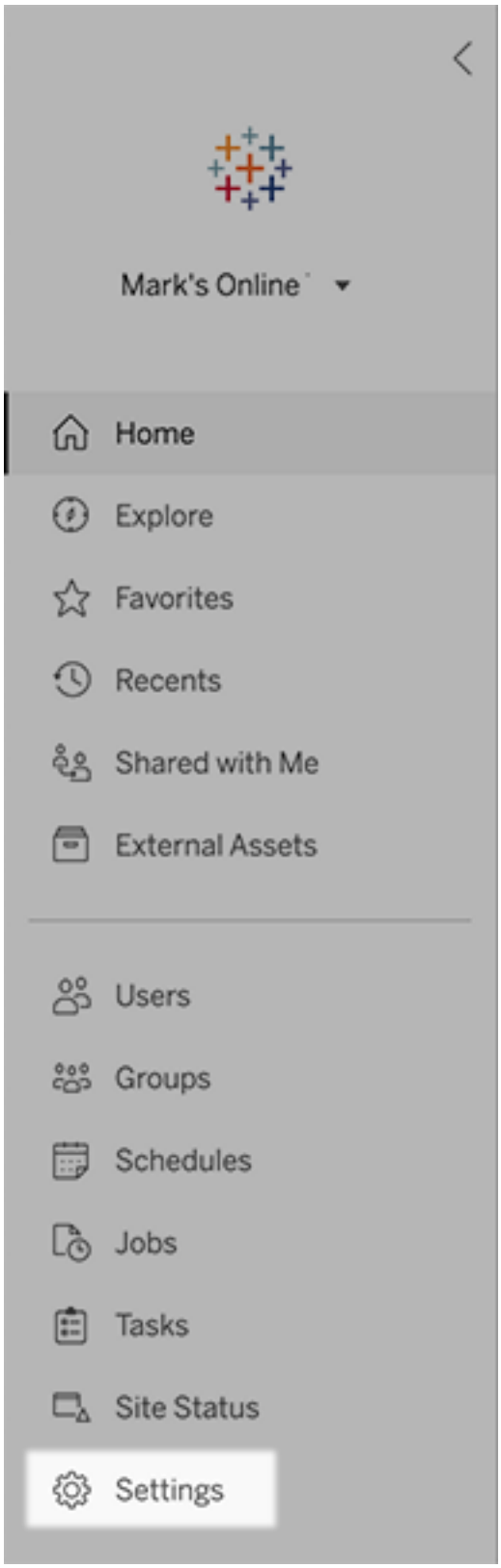
サイト設定リファレンス

以下の設定を使用して、自分の組織に合うようにサイトをカスタマイズします。サイト設定を表示および編集するには、Tableau Cloud のサイト管理者または Tableau Server のサーバー管理者である必要があります。

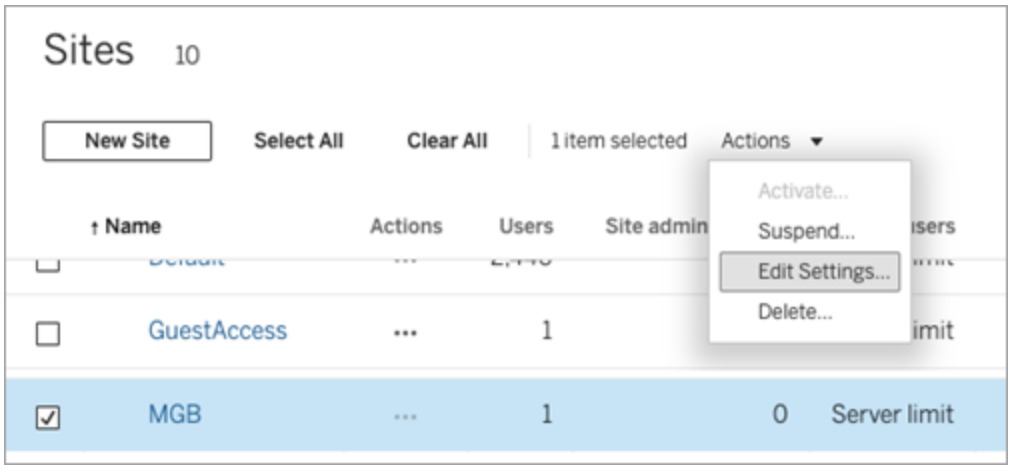
サイト設定の表示は、Tableau Cloud とTableau Server で異なります。使用できる設定は、サイトの構成と、Tableau Cloud とTableau Server のどちらを使用するかによって異なります。Ctrl+F (Windows) または Command+F (macOS) を押して設定を検索すると、以下の特定の設定を簡単に見つけることができます。

サイト設定へのアクセス

- (Tableau Cloud) ホームページで、サイドペインを展開し、下部にある **[設定]** をクリックします。


Mark's Online
Home
Explore
Favorites
Recents
Shared with Me
External Assets
Users
Groups
Schedules
Jobs
Tasks
Site Status
Settings

- (Tableau Server) 既存のサイトを編集している場合は、[サイト] ページでそのサイトを選択し、**[設定の編集]** を選択します。または、単一サイトの展開で、サイドペインの下部にある **[設定]** をクリックします。



[全般] タブ

設定	説明
[サイト名とID] (Tableau Server 管理者のみ)	ユーザー インターフェイスに表示されるサイト名と、サイトの URL に表示される ID を指定します。([既定] サイトを編集している場合は、ID を変更できません)。 URL の "#/site" の部分を変更することはできません (例: http://localhost/#/site/sales)。マルチサイト サーバー環境では、これらの設定が既定サイト以外のサイトの URL に表示されます。
[ストレージ] (Tableau Server 管理者のみ)	**[サーバーの上限]** または **[GB]** のいずれかを選択します。[GB] を選択した場合は、パブリッシュされたワークブックや抽出などのデータ ソースに適用するストレージ容量の制限に必要な GB 数を入力します。 サーバーの上限を設定してサイトがそれを超えた場合、パブリッシャーはサイトが制限内に戻るまで、新しいコンテンツをアップロードできません。サーバー管理者は、[サイト] ページの [最大容量] および [使用容量] 列を使用して、容量に対するサイトの使用量を管理できます。

[リビジョン履歴] (Tableau Server 管理者のみ)	サーバーに保存されているワークブック、フロー、およびデータソースの以前のバージョンの数を指定します。
[Tableau Prep Conductor]	適切なパーミッションを持つユーザーがフローをスケジュール設定および監視できるかどうかを制御します。Tableau Prep Conductor は データ管理 の一部です。詳細については、「Tableau Prep Conductor」を参照してください。
Web 作成	サイトのブラウザー ベースの作成を有効にするかどうかを制御します。ワークブックの Web 作成が無効になっている場合、ユーザーはパブリッシュされたワークブックをサーバー Web 環境から作成または編集することはできませんが、代わりに、Tableau Desktop を使用してワークブックをもう一度パブリッシュする必要があります。フローの Web 作成が無効になっている場合、ユーザーはパブリッシュされたフローをサーバー Web 環境から作成または編集することはできませんが、代わりに、Tableau Prep Builder を使用してフローをもう一度パブリッシュする必要があります。 詳細については、Tableau Cloud ヘルプの「サイトの Web 作成アクセスと機能の設定」を参照してください。
[ユーザーの管理] (Tableau Server 管理者のみ)	サーバー管理者のみがユーザーの追加と削除、およびサイトロールの変更を実行できるようにするか、サイト管理者にも権限を付与するかを指定します。 サイト管理者にユーザーの管理権限を与える場合、サイト管理者がサイトに追加できるユーザー数を、次のいずれかを選択して指定します。 • [サーバーの上限] では、利用可能なサーバー接続ライセンス数を追加します。コアベースのライセンス設定を使用するサーバーについては、制限はありません。 • [サイトの上限] では、サイト管理者が、指定された上限までユーザーを追加できます。 • [サイトロールの上限] では、サイト管理者が、サイトに指定されたライセンス上限まで各サイトロールのユーザーを追加できます。 詳細については、「サーバー ライセンスの表示」を参照してください。

[ゲストアクセス] (Tableau Server 管理者のみ)	Tableau Server アカウントを持たないユーザーが、ゲストアクセス パーミッションが設定されているビューを表示できます。 **注:** Tableau Server を使用している場合、管理者が [ゲストアクセス] を無効にすることができます。
[Tableau Catalog]	Tableau Server または Tableau Cloud サイトが データ管理 でライセンス設定されている場合は、Catalog 機能をオフにします。詳細については、「Catalog の無効化」を参照してください。
[スケジュールされた更新後のワークブック パフォーマンス] (Tableau Server 管理者のみ)	最近表示したワークブックをスケジュールされた更新で事前に計算し、それらのワークブックをより速く開きます。詳細については、「スケジュールされた更新後のワークブック パフォーマンスの設定」を参照してください。
[ワークブック パフォーマンス メトリクス] (Tableau Server 管理者のみ)	サイトユーザーがワークブックの実行方法 (読み込みの速度など) に関するメトリクスを収集できるようにします。記録を開始するには、ワークブックの URL にパラメーターを追加する必要があります。詳細については、「パフォーマンスの記録の作成」を参照してください。
管理対象キーチェーンのクリーンアップ (Tableau Server 管理者のみ)	サイト管理者は、サイトで OAuth 接続用に保存された認証資格情報キーチェーンを管理できます。詳細については、「OAuth 接続」を参照してください。
[抽出更新タスクを自動的に一時停止する]	Tableau ではリソースを節約するために、非アクティブなワークブックの抽出更新タスクを自動的に中断することができます。この機能は、毎週またはより頻繁に実行される更新スケジュールのみに適用されます。詳細については、

	Tableau Cloud ヘルプの「非アクティブなワークブックの抽出更新を自動的に中断する」を参照してください。
リンクされたタスク (Tableau Server およびサイト管理者のみ)	サーバー管理者は、ユーザーがスケジュールを設定してフロー タスクを順に実行できるようにします。また、ユーザーが **[今すぐ実行]** を使用して、スケジュールされたフロー タスクをトリガーして実行できるようにすることも可能です。 この設定は、サーバー レベルで適用して、Tableau Server 上のすべてのサイトを含めることができます。この設定は、サイト レベルで無効にして、特定のサイトのみを含めることができます。 リンクされたタスクをスケジュールした後にサーバー設定をオフにした場合、実行中のタスクはすべて完了しますが、スケジュール済みのリンクされたタスクは非表示になり、**[スケジュールされたタスク]** タブに表示されなくなります。 詳細については、「リンクされたタスクのスケジュール」を参照してください。
[メール設定] (Tableau Server 管理者のみ)	アラートとサブスクリプションの自動 メールに表示される送信元アドレスとメッセージ フッターを指定します。
[サイトの招待通知] (Tableau Cloud のみ)	シングルサインオン認証を使用するサイトの場合、新しいユーザーがサイトに追加されたときに招待 メールを送信します。
[サイトのロゴ] (Tableau Cloud のみ)	サイト名とともに表示される画像を指定します。
スタート ページ	ユーザーがサインインしたときに表示されるサイト ページを制御します。既定では、ホーム ページが表示されますが、代わりに、[すべてのプロジェクト]、[すべてのワークブック]、または他のページを開くことができます。詳細については、Tableau Cloud ヘルプの「すべてのユーザーの既定のスタート ページの設定」を参照してください。

> **注:** Tableau Server を使用している場合、管理者がこのサイト設定を上書きすることができます。

Tableau Pulse - 展開 (Tableau Cloud のみ)	Tableau Pulse をすべてのユーザーに対して利用可能にするか、ユーザーのグループに対して利用可能にするか、どのユーザーに対しても利用可能にしないかを制御します。詳細については、「Tableau Pulse 用にサイトをセットアップする」を参照してください。
Tableau Pulse - ダイジェスト - ユーザーがメールと Slack のダイジェストとアラートを受信できるようにする (Tableau Cloud のみ)	ユーザーが別のチャネルを介して Tableau Pulse のダイジェストとアラートを受信できるかどうかを制御します。詳細については、「Tableau Pulse 用にサイトをセットアップする」を参照してください。
Tableau Pulse - ダイジェスト - ダイジェストの開始時刻 (Tableau Cloud のみ)	Tableau Pulse がダイジェストの生成を開始する時刻を決定します。詳細については、「Tableau Pulse 用にサイトをセットアップする」を参照してください。
Tableau Pulse - ダイジェスト - 週単位および月単位のスケジュール (Tableau Cloud のみ)	週単位および月単位のダイジェストを生成する曜日と日付を決定します。詳細については、「Tableau Pulse 用にサイトをセットアップする」を参照してください。
Tableau の AI (Tableau	このセクションには、Tableau 機能で生成 AI を有効にするかどうかを制御す

Cloud)	る複数の設定が含まれています。 一部の生成 AI 機能では、Tableau+ と、Einstein Generative AI がセットアップされた Salesforce 組織への接続が必要です。Tableau の AI 機能を有効にする方法については、「サイトで Tableau の AI を有効にする」を参照してください。 Tableau の AI の詳細については、「Tableau の AI」を参照してください。
Tableau の AI (Tableau Server)	このセクションには、Tableau 機能で生成 AI を有効にするかどうかを制御する複数の設定が含まれています。 バージョン 2025.3 以降で利用できるこの機能を使用すると、独自の大規模言語モデル (LLM) プロバイダーに接続して、Tableau Server で Tableau Agent を有効にし、Tableau Web 作成と Tableau Prep に活用することができます。詳細については、「Tableau Server サイトで AI を有効にする」を参照してください。
パーソナライズされたインサイトランキング (Tableau Cloud のみ)	ユーザーが個々のインサイトに対する賛成または反対のフィードバックを提供できるかどうかを制御します。この設定がオンになっていて、ユーザーがフィードバックを提供すると、Tableau Pulse インサイトプラットフォームではそのフィードバックを基に、ユーザーに表示されるインサイトのタイプをさらにパーソナライズしてランク付けします。 この設定は、Tableau Pulse を展開するための設定とは別のものです。パーソナライズされたインサイトランキングがオフになっている場合、ユーザーは個々のインサイトに対する賛成または反対のフィードバックを送信できなくなります。詳細については、「Tableau Pulse 用にサイトをセットアップする」の「パーソナライズされたインサイトランキングをオフにする」を参照してください。
[ユーザーの表示状態]	他のユーザーに表示されるユーザー名とグループ名を制御します。詳細については、Tableau Cloud ヘルプの「サイトユーザーの表示状態の管理」を参照してください。
「データに聞く」	データソースに対して「データに聞く」レンズを有効にするか無効にするかを制

の可用性	御します。「データに聞く」では、ユーザーは会話言語を使用してデータを照会し、自動的に Viz を表示できます。詳細については、Tableau ユーザーヘルプの「「データに聞く」機能を使用した自動的なビューの構築」を参照してください。
[「データの説明を見る」の可用性]	適切なパーミッションを持つサイトユーザーが「データの説明を見る」を実行できるかどうか、および作成者が「データの説明を見る」の設定にアクセスできるかどうかを制御します。詳細については、「データの説明を見る」へのアクセスの制御を参照してください。「データの説明を見る」の詳細については、「データの説明を見る」を使用して、インサイトをより早く発見するを参照してください。
[データベースおよびテーブルに関するメタデータへの自動アクセス]	派生パーミッションを使用して、外部アセットに対する特定の機能をユーザーに自動的に付与します。詳細については、Tableau Cloud ヘルプの「派生パーミッションをオフにする」を参照してください。
[系列の機密データ]	関連するメタデータに対する適切なパーミッションをユーザーが持っていない場合に、系列の機密データを難読化するか、またはフィルターリングするかを指定します。詳細については、「系列の機密データ」を参照してください。
[クロスデータベース結合]	複数のソースからのデータを結合するときに、結合プロセスが実行される場所を指定します。詳細については、Tableau ユーザーヘルプの「異なるデータベースのテーブルを組み合わせる」を参照してください。
[保存中の抽出の暗号化] (Tableau Server 管理者のみ)	Tableau Server に保存中の .hyper 抽出を暗号化できます。Server 管理者は、サイト上のすべての抽出の暗号化を実施したり、特定のパブリッシュ済みワークブックやデータソースに関連付けられたすべての抽出をユーザーが暗号化するのを許可したりできます。詳細については、「保存中の抽出の暗号化」を参照してください。
[Tableau サポートアクセス] (Tableau Cloud のみ)	Tableau サポート技術者がサイトにアクセスして、サポートケースのトラブルシューティングを支援できるようにします。既定では、この機能は無効化されています。詳細については、「サポートアクセスの有効化」を参照してください。

[共有]	ユーザーが他のユーザーとアイテムを直接共有できるようにします。アイテムが共有されると、受信者は通知を受け取り、アイテムは受信者の[自分と共有]ページに追加されます。この設定が有効でない場合、ユーザーは共有するリンクのみをコピーできます。詳細については、Tableau ユーザー ヘルプの「Web コンテンツの共有」を参照してください。
[コメント]	ユーザーが各ビューの[コメント]サイドペインにコメントを追加し、@mentions で他の Tableau ユーザーにメールで通知できるかどうかを制御します。詳細については、Tableau ユーザー ヘルプの「ビューのコメント」を参照してください。
[データドリブン アラート]	データが重要なしきい値に到達したときに、ユーザーが自動でメールを受信できるようにします。詳細については、Tableau ユーザー ヘルプのデータドリブン アラートの送信に関するページを参照してください。
[サブスクリプション]	サイト ユーザーがビューをサブスクライブし、それらのビューに関するメールを定期的に受信できるようにします。Tableau Server では、これらのオプションは、最初にサブスクリプション設定を構成している場合にのみ表示されます。
ビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベル	サブスクリプションに関連するアップストリームの視認性の高いデータ品質の警告と機密度ラベルをメールに含めるかどうかを制御します。Tableau Server では、これらのオプションは、最初にサブスクリプションを有効にして構成している場合にのみ表示されます。データ品質に関する警告の詳細については、データ品質に関する警告の設定を参照してください。機密度ラベルの詳細については、機密度ラベルを参照してください。 以前のタイトルは **[サブスクリプションのデータ品質に関する警告]** でした。 **注:** データ品質の警告と機密度ラベルは、データ管理の一部である Tableau Catalog の機能です。
[タグ付け]	ユーザーがアイテムに追加できるタグの数を指定します。既定の制限は 50 タグで、最大は 200 タグです。詳細については、「タグの使用」を参照してください。
[ビューの推奨	推奨事項をサイトに表示するかどうか、および推奨アイテムを見たユーザーの

事項]

名前を推奨ツールヒントに表示するかどうかを制御します。

> **注:** Tableau Server を使用している場合、管理者が推奨を無効にすることができます。

[アクセス権の要求]

ユーザーはコンテンツまたはプロジェクトの所有者にアクセス権の要求を送信できます。詳細については、Tableau Cloud ヘルプの「サイトユーザーにコンテンツへのアクセスを要求できるようにする」を参照してください。

[メトリクス コンテンツ タイプ]

サイトでメトリクスを使用できるかどうかを制御します。メトリクスをオンにすると、ユーザーはビューからメトリクスを作成でき、メトリクスはコンテンツ タイプとして表示されます。オフにすると、メトリクスはサイトに表示されなくなり、同期は続行されません。メトリクスをもう一度オンにすると、これらのメトリクスが再表示され、更新が再開されます。詳細については、Tableau Cloud ヘルプまたは Tableau Server ヘルプの「メトリクスのセットアップ」を参照してください。

従来のメトリクス機能の廃止

Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

Web ページ オブジェクトと Web イメージ

これらの Web ページ オブジェクトとイメージ オブジェクトでターゲット URL を表示できるかどうかを制御します。詳細については、Tableau ユーザー ヘルプの Web ページ オブジェクトのセキュリティに関するページを参照してください。

[パーソナル スペース]

Creator および Explorer サイトのユーザーがコンテンツを作成してプライベート パーソナル スペースに保存できるようにします。パーソナル スペースがオンになっている場合、ユーザーのストレージ制限を設定できます。詳細について

	は、パーソナル スペースでプライベート コンテンツを作成および編集を参照してください。
コレクション	サイトでコレクションを使用できるかどうかを制御します。コレクションをオンにすると、ユーザーはコンテンツを整理するためのコレクションを作成したり、他のユーザーが利用できるコレクションを閲覧したりすることができます。詳細については、「コレクション内のアイテムを整理する」を参照してください。
[サイトにおける抽出のタイム ゾーン]	サイトにおける抽出ベースのデータ ソースの既定タイム ゾーンは、協定世界時 (UTC) です。サイト管理者は、別のタイム ゾーンを設定できます。詳細については、Tableau Server ヘルプの「サイトのタイム ゾーンを抽出用に設定する」または Tableau Cloud ヘルプの「サイトのタイム ゾーンを抽出用に設定する」を参照してください。
[抽出容量制限の通知]	抽出ジョブ容量の問題が原因で抽出更新ジョブがキャンセルされた場合、すべてのサイト管理者にメール アラートを送信します。
フロー パラメーター (Tableau Server およびサイト管理者のみ)	ユーザーは、パラメーターを含むフローをスケジュールして実行できます。 管理者は、フロー パラメーターで任意の値が受け入れられるようにすることもできます。このオプションを有効にすると、フロー ユーザーはパラメーターに任意の値を入力できるため、ユーザーがアクセスすべきではないデータがパブリッシュされる可能性があります。 パラメーターは、ファイル名、パス、テーブル名のインプット ステップ、またはカスタム SQL クエリを使用する場合は、ファイル名、パス、テーブル名のアウトプット ステップ、およびフィルターまたは計算値の任意のステップ タイプに入力できます。 フロー パラメーターの設定は、サーバー レベルで適用して、Tableau Server 上のすべてのサイトを含めることができます。この設定は、サイト レベルで無効にして、特定のサイトのみを含めることができます。 パラメーターを使用する方法についての詳細は、Tableau Prep ヘルプの「フローでパラメーターを作成して使用する」を参照してください。

[今すぐ実行]	Web、Rest API、および Tabcmd で [今すぐ実行] オプションを使用して、ジョブを手動で実行できるユーザーを制御します。既定では、このオプションはユーザーが手動でジョブを実行できるように設定されています。管理者だけがジョブを手動で実行できるようにする場合は、このチェックボックスをオフにします。 **注:** Tableau Server を使用している場合、管理者がこのサイト設定を無効にすることができます。
[通知の管理]	抽出ジョブ、フローの実行、または他のユーザーによるコンテンツやコメント内のメンションの共有などのイベント通知を、サイトユーザーが受信する方法を制御します。通知は、通知センターを介して Tableau サイトに表示したり、メールで送信したり、Slack ワークスペースに送信したりできます。通知が有効になっている場合、ユーザーは [アカウント設定] ページで通知設定を構成できます。 **注:** Tableau Server を使用している場合、サーバー管理者がこのサイトの設定を無効にすることができます。
メール通知のカスタマイズ (Tableau Cloud のみ)	データに基づくアラートおよびワークブックとビューのサブスクリプションの通知メールを、Tableau メール サーバーと独自の SMTP サーバーのどちらを使用して送信するかを制御します。独自の SMTP サーバーを使用する場合、メール送信者の名前や送信者のメール アドレスで使用されるドメイン、およびメール通知内のリンクのドメインをカスタマイズすることができます。2024 年 2 月の Tableau Cloud で追加されました。 SMTP 設定とは別に、データに基づくアラートとサブスクリプションの通知メールをユーザーに送信する際に、リンクを含めるかどうかを制御します。これらのリンクは、ユーザーを Tableau サイトに誘導して、コンテンツを表示したり、アラートやサブスクリプションを管理したりできるようにします。この設定がオンになっているかどうかに関係なく、通知メールにはサブスクリプションを解除するためのリン

	クが必ず含まれています。2023 年 6 月の Tableau Cloud で追加されました。
フロー サブスクリプション	フローの所有者がフローの出力データを含むメールを、自分自身や他のユーザー宛てにスケジュールに基づいて送信できるかどうかを制御します。フロー サブスクリプションを許可する場合、フロー出力データをサブスクリプション メールに含めるかどうか、およびフロー出力ファイルをメールに添付するかどうかを制御することができます。詳細については、「フロー実行の成功をユーザーに通知する」を参照してください
[OAuth クライアント レジストリ]	コネクタのサブセットの場合、サイトのカスタム OAuth クライアントを登録すると、サーバー用に構成された OAuth クライアントを上書きできます。カスタム OAuth クライアントを登録することにより、新規および既存の接続で、サーバー全体の OAuth クライアントの代わりにサイトレベルの OAuth クライアントを使用できるようになります。詳細については、「カスタム OAuth の構成」を参照してください。
ビューの高速化	Creator および Explorer のサイト ユーザーが、Tableau Server ワークブックのビューをアクセラレートして読み込み時間を短縮できるかどうかを制御します。ビューの高速化を許可する場合、高速化するビューの最大数を設定し、高速化タスクが繰り返し失敗するビューの高速化を自動的に停止するように選択できます。詳細については、「ビューの高速化」を参照してください。
グループ メンバーシップのアサーション	Tableau Server に接続するユーザーを認証するときにグループ メンバーシップを動的にアサートすることによって、SSO IdP または Tableau の接続済みアプリを通じてローカル グループ メンバーシップを制御および管理できるようにします。SAML アサーション、OIDC アサーション、または JSON Web トークン (JWT) で追加の設定が必要です。詳細については、アサーションを使用した動的グループ メンバーシップを参照してください。 **重要**: このサイトレベルの設定は、まずサーバー全体の設定が有効になっている場合にのみ有効にできます。
グループ セット	**[グループ セット]** ページと、グループ セットを作成する機能を有効にします。グループ セットは、特定のユーザー (サーバー管理者、サイト管理者、プロジェクト所有者、コンテンツ所有者) が使用できます。ここで適用されるパーミッショ

	ンルールでは、グループセットを基にしてパーミッションが設定されたコンテンツにユーザーがアクセスするには、そのユーザーがグループセット内のすべてのグループのメンバーである必要があります。詳細については、「グループセットの操作」を参照してください。
ごみ箱	サイトのごみ箱をオンにして、Explorer (公開可能) または Creator ライセンスを持つ管理者と作成者が、選択した期間、以前に削除したプロジェクト、ワークブック、およびデータソースを復元できるようにするかどうかを制御します。詳細については、「ごみ箱」を参照してください。
クロスドメインアイデンティティ管理 (SCIM) (Tableau Server)	サーバー管理者は、サイト固有の SCIM を有効にして、ユーザーとグループのプロビジョニングと同期を自動化できるようになります。詳細については、「SCIM」を参照してください。**注:** この機能を有効にするには、サイト固有の SAML 認証を構成する必要があります。

[認証] タブ (Tableau Cloud)

設定	説明
[認証タイプ]	ユーザーがサイトにサインインする方法、および初回のサインイン後にユーザーがサイトにアクセスする方法を指定します。認証では、ユーザーの ID を検証します。詳細については、「認証」を参照してください。
[埋め込みビュー用の既定の認証タイプ]	ユーザーが埋め込みビューにサインインする方法を指定します。既定では、Tableau 認証が選択されています。
認証ワークフローでのユーザーアクセスの制御	コンテンツで使用されるユーザー属性関数が、接続済みアプリ、OIDC クレーム、SAML XML 応答を経由して JSON Web トークン (JWT) から渡されるユーザー属性を受け入れるようにします。ユーザー属性は Tableau に渡され、ランタイムにユーザーに表示できるデータをカスタマイズおよび制御できるようにします。詳細については、「埋め込み API v3」のヘルプ、「OIDC」、または「SAML」のトピックを参照してください。

クロスドメイン アイデンティティ管理 (SCIM)	サードパーティアイデンティティプロバイダー (IdP) を通じてサイト上のユーザーを管理できるようにします。有効にすると、IdP の SCIM 構成で使用する値がベース URL とシークレット ボックスに設定されます。詳細については、「外部 ID プロバイダーを介したユーザープロビジョニングおよびグループ同期の自動化」を参照してください。
[接続されたクライアント]	ユーザーが初めてサインイン認証資格情報を提供した後、Tableau Mobile や Tableau Bridge などの Tableau クライアントがサーバーに対して認証されたままになることを許可します。オフにすると、ユーザーは Tableau Cloud にアクセスするたびに明示的にサインインする必要があります。詳細については、「接続されたクライアントからサイトへのアクセス」を参照してください。

[Bridge] タブ (Tableau Cloud)

設定	説明
[クライアントが通知を実行していない]	クライアントがサイトから切断されているように見える場合、データ ソース所有者にメール アラートを送信します。
プール	ライブ クエリを分散し、Bridge プール内のすべてのクライアントでジョブを更新します。詳細については、Tableau Cloud ヘルプの「Bridge クライアントのプールの構成と管理」を参照してください。
プライベート ネットワークの許可リスト	Tableau Cloud に代わって、プライベート ネットワーク データへの専用 Bridge プール アクセスを可能にするドメインを追加および管理します。

[拡張機能] タブ

設定	説明
ダッシュボードとViz の拡張機能	ダッシュボードとViz の拡張機能を管理および制御します。ダッシュボードの拡張機能は、カスタム ダッシュボード ゾーンで実行される Web アプリケーションであり、ダッシュボードの残りの部分を操作できます。Viz の拡張機能は、新しい Viz タイプをサポートする Web アプリケーションです。詳細については、Tableau

	Cloud ヘルプまたは Tableau Server ヘルプの「ダッシュボードとViz の拡張機能の管理」を参照してください。
[分析拡張機能]	R とPython との統合用として分析拡張機能に式を渡すために使用できる関数セットを有効にします。詳細については、Tableau Cloud ヘルプまたは Tableau Server ヘルプの「分析拡張機能を使用した接続の構成」を参照してください。
[Tableau Prep 拡張機能]	Web でフローを作成する場合、ユーザーは Einstein Discovery を活用したモデルをフローに適用して、データの予測を一括でスコアリングできます。 詳細については、Tableau Server ヘルプの「Einstein Discovery 統合の設定」を参照してください。

[統合] タブ

設定	説明
[Slack 接続]	Slack ワークスペースとTableau サイト間の接続を表示します。接続すると、Tableau サイトユーザーは、接続された Slack ワークスペースで Tableau 通知を確認できます。 **注**: Tableau Server では、Tableau Server 管理者が Oauth クライアントを追加して Slack に接続する前に、Slack 管理者がプライベートSlack アプリを作成して Slack ワークスペースにインストールする必要があります。 Tableau Server では、プライベートSlack アプリケーションの Oauth クライアント情報を追加してから、[Connect to Slack (Slack に接続)] を選択して接続を完了することができます。詳細については、Tableau とSlack ワークスペースの統合を参照してください。

[分析拡張機能]	R とPython との統合用として分析拡張機能に式を渡すために使用できる関数セットを有効にします。詳細については、Tableau Cloud ヘルプまたは Tableau Server ヘルプの「分析拡張機能を使用した接続の構成」を参照してください。
Salesforce へのパブリッシュ (Tableau Cloud および Tableau Server のベータ版)	サイトユーザーは Salesforce アプリにビューをパブリッシュすることができます。 ビューを Salesforce にパブリッシュすると、選択したアプリにアクセスできるユーザーは誰でも、そのコンテンツの存在を確認することができます。ただし、ビューを表示できるのは、既存の Tableau パーミッションを使用してサインインしているユーザーのみです。詳細については、「ビューを Salesforce にパブリッシュする (ベータ版)」を参照してください。

[接続済みアプリ] タブ

設定	説明
接続済みアプリ	Tableau 接続済みアプリの作成と管理、または Tableau Server (サーバー全体およびサイトレベル) とカスタム アプリケーション間の明示的な直接信頼関係または OAuth 2.0 信頼関係の作成と管理を行い、JSON Web トークン (JWT) を使用してユーザーに代わって Tableau REST API へのアクセスをプログラムで承認します。詳細については、Tableau Cloud ヘルプの「Tableau 接続済みアプリをアプリケーションの統合に使用する」を参照してください。

[Mobile] タブ

設定	説明
アプリロック	ユーザーがこのサイトを Tableau Mobile で開くには、生体認証方式またはデバイスのパスコードが必要です。詳細につい

	ては、「Tableau Mobile 展開ガイド」の「アプリロックを有効にしてセキュリティを強化する」を参照してください。
オフラインプレビュー	ユーザーが Tableau Mobile のサイトにアクセスしたときに、オフラインプレビューを生成して表示するかどうかを制御します。詳細については、「Tableau Mobile 展開ガイド」の「デバイスでの Tableau Mobile データの管理」を参照してください。
モバイル セキュリティ ポリシー	一部のセキュリティポリシーは自動的に有効になり、無効にすることはできません。モバイル セキュリティポリシーは、Tableau Mobile の MAM バージョンでは利用できません。
ジェイルブレイクの検出	「ジェイルブレイク」または「ルート化」されたデバイスを使用している Tableau Mobile アプリユーザーが Tableau のコンテンツにアクセスできるかどうかに加えて、ジェイルブレイクまたはルート化されたデバイスが検出されたときに発生する応答のレベルを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。
マルウェア検出 (Android のみ)	モバイル デバイスでマルウェア検出を有効にするかどうかに加えて、マルウェアが検出されたときに発生する応答のレベルを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。 2025 年 9 月に、Tableau Mobile はマルウェア検出設定のサポートを廃止します。マルウェア検出設定は、2025 年 9 月の Tableau Cloud 設定ページと、Tableau Server バージョン 2025.2.3 の Tableau Server 設定ページから削除されました。Tableau Server の以前のバージョンでマルウェア検出を設定しても、Tableau Mobile アプリには影響しません。
ポリシーの更新を行わないオフラインの最大日数	モバイル デバイスがオフラインの状態でアプリを使用できる最大日数を設定するかどうかを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。

デバッグの防止	モバイル デバイスでデバッガーを防止するかどうかを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。
画面共有とスクリーンショット (Android のみ)	Tableau Mobile ユーザーがアプリ内でスクリーンショットを撮ったり、画面共有を使用したりできるかどうかを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。

ユーザーとグループの管理

Tableau サイトにユーザーを追加して、各ユーザーのアクセス レベルを決定するサイト ロールを設定できます。さらに、ユーザーのグループを作成して、サイトへのゲスト アクセスを有効にすることもできます。

サイトへのユーザーの追加

Tableau Server にアクセスする必要がある人は、その目的が参照、パブリッシュ、コンテンツの編集、サイトの管理のいずれであっても、ユーザーとして追加される必要があります。管理者はユーザーを追加する以下のオプションを使用できます。

- この記事で後述されているとおり、ローカルのユーザー アカウントまたは Active Directory のユーザー アカウントを追加します。

  Active Directory グループをインポートしてユーザーを追加することもできます。Active Directory 経由でのグループの作成を参照してください。

  Tableau Server on Linux の場合は、すべての外部ディレクトリ通信が LDAP アイデンティティストアによって構成され管理されます。ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアが構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

- CSV インポートファイルのガイドラインを使用して作成するCSV ファイルからユーザーのインポート。

ユーザー管理へのサイト管理者アクセス

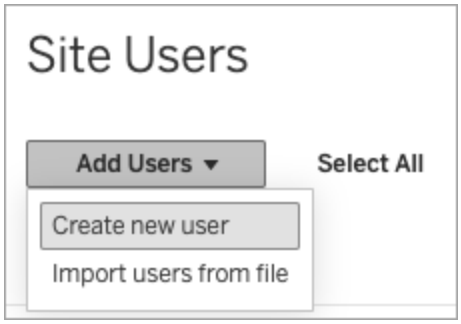
既定ではサイト管理者はサイト上のユーザーの追加と削除を行うことができます。サーバー管理者はサイトの [設定] ページで、その権限を取り消し、サーバー管理者のみがサイトのユーザーの管理を行えるようにできます。

サイト管理者がそのユーザーがメンバーとなっているすべてのサイトに対するアクセス権を持っている場合のみ、既存のローカル ユーザー アカウントを編集できます。たとえば、User1 がサイトA とサイトB のメンバーなら、サイトB のみを管理する管理者は、User1 のフル ネームを変更することもパスワードをリセットすることもできません。

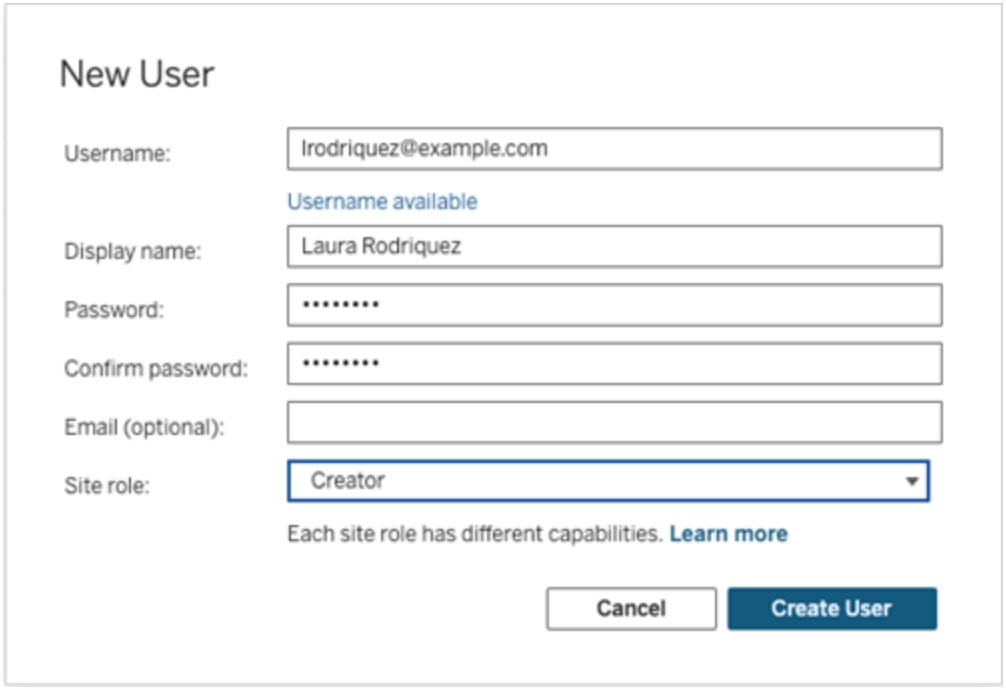
注: ローカル認証を使用する際に、サイト管理者がユーザーを追加または削除できれば、ユーザー名が Tableau Server のサイトでユーザーとして構成されているかどうかがわかります。

ローカル ユーザーをサイトに追加する

1. Tableau Server に管理者としてサインインし、該当する場合にはサイトを選択します。
2. **[ユーザー]** を選択します。[ユーザー] ページで、**[ユーザーの追加]**、**[新しいユーザーの作成]** の順にクリックします。



3. ユーザー名を入力します。ローカル認証を使用する場合、電子メール アドレスをユーザー名として使用するのがユーザー名の競合を避けるための最適な方法です (たとえば、*lrodriguez* ではなく*lrodriguez@example.com*)。



ユーザー名は大文字と小文字を区別しません。ユーザー名で許可されていない文字はセミコロン (;) とコロン (:) です。

次のフィールドにも情報を入力します:

- **表示名** - ユーザーの表示名を入力します (*Laura Rodriguez* など)。
- **パスワード** - ユーザーのパスワードを入力します。
- **パスワードの確認** - パスワードを再入力します。
- **メール** - これは任意です。ユーザープロフィール設定から後で追加できます。

4. サイトロールを選択します。

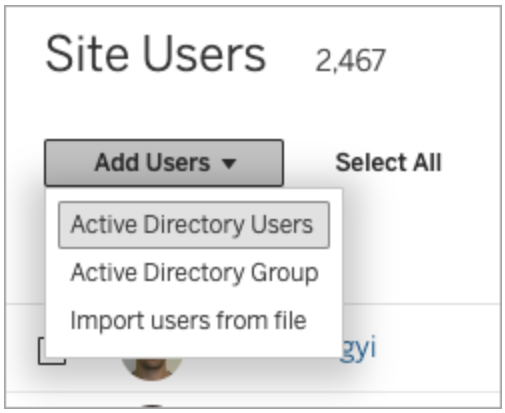
サイトロールの定義については、ユーザーのサイトロールの設定を参照してください。

5. **[ユーザーの作成]** をクリックします。

Active Directory ユーザーをサイトに追加する

ユーザーをサイトに追加する前に、外部 ID ストアを使用した展開におけるユーザー管理を見直し、複数のドメイン、ドメイン命名、NetBIOS、Active Directory ユーザー名の形式が Tableau ユーザー管理にどのように影響するかについて理解してください。

1. サイトで、**[ユーザー]**、**[ユーザーの追加]**、**[Active Directory ユーザー]** を順にクリックします。



2. 1 つまたは複数のユーザー名を入力します(セミコロンで区切る)。

   たとえば、「*tdavis; jjohnson; hwilson*」のように入力します。

   サーバーを実行している同じ Active Directory ドメインのユーザーを追加する場合は、ドメインなしで AD ユーザー名を入力するだけです。サーバーのドメインが使用されます。

   > **注**: このフィールドにはユーザーのフル ネームを入力しないでください。インポートプロセスでエラーの原因となる可能性があります。

3. サイト ロールを選択します。

サイトロールの定義については、ユーザーのサイトロールの設定を参照してください。

4. **[ユーザーのインポート]** をクリックします。

ローカル ユーザーの削除

1. 管理者として Tableau Server にサインインし、サイトを選択して **[ユーザー]** ページを開きます。

2. ユーザー名の横にあるチェック ボックスを選択して、**[アクション]** メニューから、**[削除]** を選択します。

   ユーザーが現在のサイトのみのメンバーであり、コンテンツを所有していない場合、ユーザーはサーバーから削除されます。

   現在のサイトから削除するユーザーが、サーバー上の他のサイトのメンバーである場合、それらのサイトでのステータスに変更はありません。

関連情報

- サイトに所属させることなく Tableau Server へのユーザーの追加することもできます。
- Tableau Server を 2018.1 以前のバージョンからバージョン 2018.1 以降にアップグレードし、ユーザーベースのライセンスを認証しない場合、2018.1 サーバー バージョンで **Viewer (ビューアー)** サイトロールを割り当てられたユーザーには影響が及びます。

  詳細については、ライセンス発行の概要の「ユーザーベースのライセンス」およびユーザーのサイトロールの設定を参照してください。

ユーザーのサイトロールの設定

Tableau Server または のサイトにユーザーを追加する場合、ライセンス タイプに関わらず、それらのユーザーに *サイト ロール*を適用する必要があります。サイトロールはサイトに対してユーザーが持つことができる最大アクセス レベルを示します。コンテンツ パーミッションと併せて、サイトロールにより、誰にパブリッシュと操作を許可するのか、パブリッシュされたコンテンツの表示のみを許可するのか、サイト自体でサイトのユーザーと管理者を誰が管理するのかが決定されます。

> Tableau Server on Windows については、「ユーザーのサイト ロールの設定」を参照してください。

ユーザー ライセンス、サイト ロール、コンテンツ パーミッションが連係するしくみ

ユーザーのライセンス タイプ、サイト ロール、コンテンツ パーミッションの交点により、ユーザーが Tableau サイトで持つアクセス レベルが決まります。

> **注:** 一部のユーザーはサイト間で異なるロールを持っているため、**[サーバー ユーザー]** タブのヘッダーの上部にあるライセンス レベルのカウントは、**[Max User Role (最大ユーザー ロール)]** フィルターのカウントとは異なる場合があります。

1. ライセンス タイプはユーザーに関連付けられています。ユーザーに割り当てるサイト ロールにより、必要なライセンス タイプが決まります。

   Tableau Server のマルチサイト環境では、ユーザーのライセンスは、ユーザーがメンバーであるすべてのサイトに適用されます。

2. サイト ロールもユーザー レベルに設定されています。マルチサイト環境では、各サイトにサイト ロールを割り当てます。たとえば、同じユーザーが、あるサイトではサイト管理者 Creator サイト ロール、別のサイトでは Viewer サイト ロールを持つことができます。

   サイト ロールは、ユーザーが持つことができる最大の権限を定義します。

3. サイト ロールの最大の権限をユーザーが利用できるかどうかは、コンテンツ リソース (プロジェクト、データ ソース、ワークブック) に対して設定されているパーミッションにより決まります。

たとえば、あるユーザーはサイトに対して次のアクセス権を持つとします。

- Creator ライセンス (別のサイトへのアクセスのため)
- Explorer サイト ロール (このサイト)
- (このサイトの) プロジェクトに対する保存パーミッション機能

このシナリオでは、ライセンスにより、Web 編集環境または Tableau Desktop で新しいデータソースへの接続とデータソースの作成が許可されるほか、パーミッション ルールにより、プロジェクトへの新しいデータソースの保存が許可されます。ただし、サイト ロールは保存することを許可しません。つまり、有効なパーミッションには保存機能が含まれていません。したがって、ユーザーはコンテンツをサイトにパブリッシュできません。

ユーザーが Creator ライセンスと Creator サイト ロールを持っている場合でも、少なくとも 1 つのプロジェクトで保存機能がなければサイトにパブリッシュすることは一切できません。

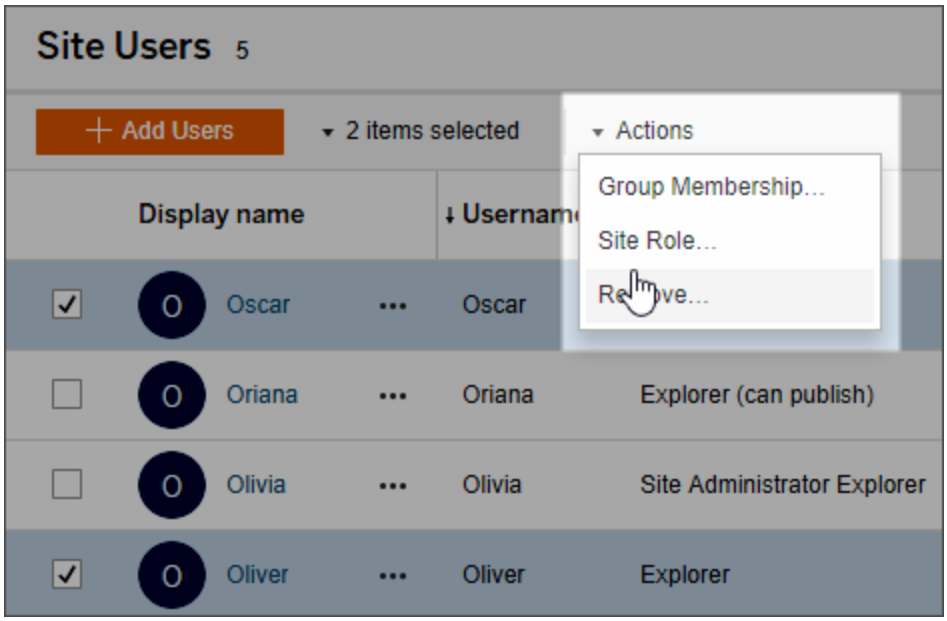
詳細については、パーミッションを参照してください。

ユーザーのサイト ロールの変更

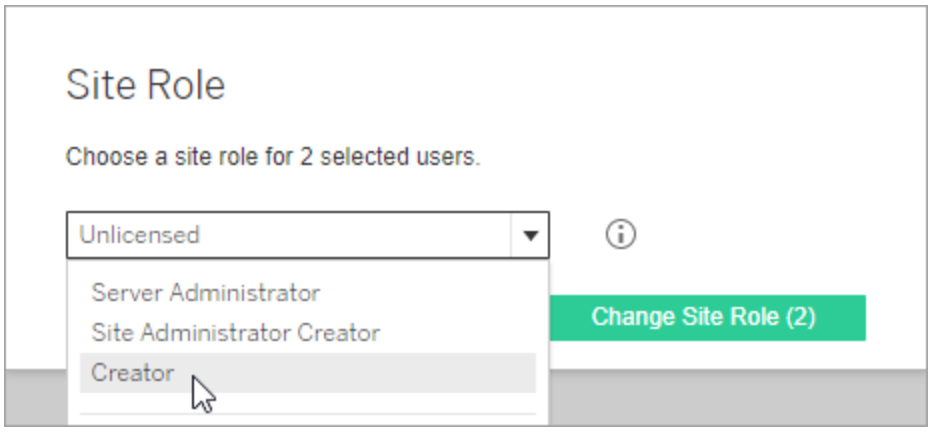
1. サーバーまたはサイト管理者としてサイトにサインインし、**[ユーザー]** エリアに移動します。

   サイト管理者であるときに、**[ユーザー]** エリアが表示されない場合、サーバー管理者に、サイト管理者のユーザー管理権限を拒否したかどうかを確認してください。

2. ユーザーを選択してから、**[アクション] > [サイト ロール]** を選択します。



3. 新しいサイト ロールを選択し、**[サイト ロールの変更]** をクリックします。



ポインターを情報アイコンに重ね、各サイトロールが許可する一般的な機能の最大レベルを示すマトリクスを表示することができます。詳細は、各サイトロールで許可されている一般的な機能に続行してください。

各サイトロールで許可されている一般的な機能

以下の表は、ライセンスタイプ、各タイプで許可される最高レベルのサイトロールを一覧にし、各サイトロールで許可される最大の権限を要約しています。

この記事で扱う内容とここで扱われない内容を探す場所

- この情報は *サイトロール*に重点を置き、より一般化されています。*ライセンスロール*ごとに利用可能な共通の特定のタスクのリストについては、Tableau 価格ページのチームおよび組織向け タブにあるマトリックスを参照してください。
- コアベースのライセンスがユーザーベースのライセンスにどのように関連しているかや、その他の特定のライセンス発行への移行シナリオについての詳細情報は、まず以下のトピックを参照してください。

  コアベース ライセンスからロールベース ライセンスへの移行

  ライセンス発行のトラブルシューティング

  コアベースのライセンス モデル (ライセンス モデルとプロダクト キーを理解する)

Tableau サイトロール

サイトロール名	このサイトロールで許可される最大の権限
Creator ライセンスを使用するサイトロール	
—これらのサイトロールを持つユーザーには、Tableau Prep、Tableau Desktop、Tableau Bridge、Tableau Mobile などの Tableau クライアントに対するアクセス権があります。	
サーバー管理者	Tableau Server でのみ利用可能。Tableau Cloud には適用されません。 このサイトロールは常に Creator および Explorer 間のサーバーで認証された最高のライセンスとなります。Tableau Server ブラウザ環境の構成設定、サーバー上のすべてのサイト、ユーザーとグループのほか、フロー、プロジェクト、データソース (接続情報を含む)、ワークブックなどすべてのコンテンツ資産に対して無制限のアクセス権を許可します。 ブラウザ、Tableau Desktop、または Tableau Prep から、Tableau のパブリッシュされたデータソースまたは外部データに接続できます。新しいデータソースを作成してパブリッシュできます。ワークブックを作成してパブリッシュできます。
サイト管理者 Creator	Tableau Cloud で最高のアクセスレベルです。 上記のとおりコンテンツに対する無制限のアクセス権がありますが、サイトレベルでのアクセス権は除きます。ブラウザ、Tableau Desktop、または Tableau Prep で Tableau または外部データに接続できます。新しいデータソースを作成できます。コンテンツを作成してパブリッシュできます。 Tableau Server では、サーバー管理者が、サイト管理者にユーザーの管理およびサイトロールとサイトメンバーシップの割り当てを許可するかどうかを決定できます。Tableau Server では既定で、また Tableau Cloud では常に、サイト管理者にはこれらの機能が許可されます。

サイトロール名	このサイトロールで許可される最大の権限
Creator	以前のパブリッシャーサイトロールに類似していますが、新機能が使用可能です。このサイトロールでは、管理者以外は最大レベルのコンテンツアクセスが得られます。 ブラウザで Tableau または外部データへの接続、フロー、データソース、ワークブックの作成とパブリッシュ、Dashboard Starters へのアクセス、パブリッシュされたビューの操作機能の使用を行うことができます。 Tableau Prep または Tableau Desktop からデータに接続し、フロー、ワークブック、データソースのパブリッシュ(アップロード/保存)およびダウンロードも行うことができます。
Explorer ライセンスを使用するサイトロール	
—これらのサイトロールを持つユーザーは、ブラウザまたは Tableau Mobile からサーバーにアクセスできます。	
サーバー管理者	Tableau Server でのみ利用可能。Tableau Cloud には適用されません。 新しいサーバー管理者ユーザーが作成されたときに、サーバーでアクティブ化される最高のライセンスタイプが Explorer である場合、ユーザーのサイトロールはサーバー管理者です。ただし、そのユーザーは、Creator ライセンスのみに付属する完全な接続およびパブリッシュ機能を利用できません。 Explorer ライセンスがあると、サーバー管理者には Tableau Server ブラウザ環境の構成設定、サーバー上のすべてのサイト、ユーザーとグループのほか、プロジェクト、フロー、データソース(接続情報を含む)、ワークブックなどすべてのコンテンツ資産に対して無制限のアクセス権が許可されます。 ただし、Explorer ライセンスがあっても、サーバー管理者はブラウザから外部データに接続して新しいデータソースを作成することはできませ

サイト ロール名	このサイト ロールで許可される最大の権限
	ん。Tableau Desktop からはワークブックとデータ ソースを作成またはパブリッシュできます。パブリッシュに関しては、Explorer (パブリッシュ可能) サイト ロールと同じ機能が使えます。Tableau Prep フローをパブリッシュすることはできません。
サイト管理者 Explorer	サイトおよびユーザーの構成に対するアクセス権はサイト管理者 Creator と同じですが、Web 編集環境からは外部データまたは仮想接続に接続できません。 Tableau のパブリッシュされたデータ ソースに接続して新しいワークブックを作成したり、既存のワークブックの編集や保存を行うことができます。Tableau Prep フローをパブリッシュすることはできません。
Explorer (パブリッシュ可能)	既存データ ソースを使用して Web からワークブックをパブリッシュしたり、パブリッシュしたビューの閲覧や操作を行ったり、すべての操作機能を使用したりできます。 Web 編集環境では、既存のワークブックを編集して保存できます。ワークブックに埋め込まれたデータ接続から新しいスタンドアロン データ ソースを保存したり、外部データまたは仮想接続に接続したり、新しいデータ ソースを作成したりすることはできません。Tableau Prep フローをパブリッシュしたり実行したりすることはできません。
Explorer	パブリッシュされたビューを閲覧および操作できます。コンテンツをサブスクライブし、データ ドリブン アラートを作成したり、Tableau のパブリッシュされたデータ ソースに接続したり、アドホック クエリのために Web 作成環境でワークブックを開いたりすることはできますが、作業は保存できません。仮想接続に接続することはできません。Tableau Prep フローをパブリッシュしたり実行したりすることはできません。
Viewer ライセンスを使用するサイト ロール	
Viewer	他のユーザーが作成した、パブリッシュされたビューを表示し、ほとんどの

サイトロール名	このサイトロールで許可される最大の権限
	操作機能を使用することができます。ビューにサブスクライブしたり、イメージデータまたは概要データとしてダウンロードしたりできます。データへの接続、コンテンツの作成、編集、パブリッシュ、データアラートの設定は行うことができません。 具体的な権限のリストについては、Tableau の価格ページのマトリックスの **Viewer** 列を参照してください。
その他のサイトロール	
ライセンスなし	ライセンスなしのユーザーは、Tableau Server または Tableau Cloud にサインインできません。次の状況では、ユーザーには「ライセンスなし」のロールが割り当てられます。 • CSV ファイルからユーザーをインポートします。ライセンスレベルはライセンスなしに設定されます。 • ユーザーの追加またはインポート時に、利用可能なライセンス数に達しました。 • サイト上のコンテンツを所有するユーザーを削除します。ユーザーは引き続きコンテンツを所有しますが、何も行うことはできません。 • プロダクトキーの有効期限が切れています。プロダクトキーの有効期限と属性の更新を参照してください。

コンテンツをパブリッシュできる人物

以下のサイトロールではパブリッシュについて指定されたレベルのアクセス権が許可されます。

- **サーバー管理者** (Tableau Server のみ)、**サイト管理者 Creator**、**Creator** では接続およびパブリッシュについて完全なアクセス権が許可されます。

これにはデータへの接続のほか、Tableau Desktop や Web 編集環境からの新しいフロー、新しいワークブック、新しいデータソースのパブリッシュが含まれます。このサイトロールでは、既存のパブリッシュされたワークブックの編集や保存、または既存のデータソースに対する更新のパブリッシュも許可されます。

- **Explorer (パブリッシュ可能)** および**サイト管理者 Explorer** では、各サイトロールで許可されている一般的な機能で説明されているように、パブリッシュ権限に制限があります。
- **Explorer**、**Viewer**、**読み取り専用**、および**ライセンスなし**ではパブリッシュが許可されません。

サイトロールとActive Directory のインポートと同期

Active Directory などの外部ディレクトリからユーザーをインポートする場合は、サイトロールを指定できます。ユーザーがサーバーの任意のサイトのメンバーでない場合、ユーザーは割り当てられたロールでサイトに追加されます。外部ディレクトリからグループを同期すると、サイトロールは **[グループ - 詳細]** ページの **[最低限のサイトロール]** 設定を通じて適用されます。

注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

ユーザーが Tableau Server サイトに既に存在する場合、インポートまたは同期プロセスで割り当てられたサイトロールは、ユーザーによるサイトへのアクセスが増える場合にのみ適用されます。AD ユーザーとグループをインポートまたは同期すると、ユーザーのサイトロールのレベルが上げることができますが、ユーザーのサイトロールのレベルが下がることはありません。

ユーザーがパブリッシュする機能をすでに持っている場合、その機能は保持されます。

以下のマトリックスでは、インポートのサイトロールに適用されるルールを示しています。

注: [サイトロールのインポート] 行の略称ヘッダーは、インポートのために指定したサイトロールを示します。**[現在のサイトロール]** 列ヘッダーは、ユーザーの現在のサイトロールを示しま

す。表の値は、結果のサイトロールの略語を示します。

- サイト管理者: SA
- サイト管理者 Creator: SC
- サイト管理者 Explorer: SE
- Creator: C
- Explorer: E
- Explorer (パブリッシュ可能): EP
- Viewer: V
- ライセンスなし: U

	現在のサイトロール						
サイトロールのインポート	**SC**	**C**	**SE**	**EP**	**E**	**V**	**U**
サイト管理者 Creator (SC)	SC	SC	SC	SC	SC	SC	SC
サイト管理者 Explorer (SE)	SC	SE	SE	SE	SE	SE	SE
Creator (C)	SC	C	SE	C	C	C	C
Explorer (パブリッシュ可能) (EP)	SC	C	SE	EP	EP	EP	EP
Explorer (E)	SC	C	SE	EP	E	E	E
Viewer	SC	C	SE	EP	E	V	V

	現在のサイトロール						
サイトロールのインポート	**SC**	**C**	**SE**	**EP**	**E**	**V**	**U**
(V)							
ライセンスなし **(U)**	SC	C	SE	EP	E	V	U

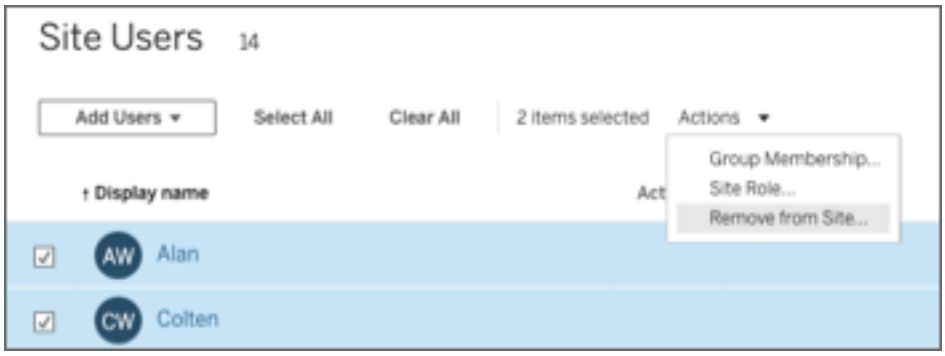
ユーザーの表示、管理、削除

管理者は、ユーザーの追加や削除、ユーザーが属するグループの設定、サイトロールの設定など、サイトのユーザー管理を行うことができます。サーバー管理者は **[すべてのサイト]** ページで一度に複数サイトのユーザーを管理できます。

サイトのユーザーの表示と管理

サイトに管理者としてサインインし、**[ユーザー]** を選択します。このページでは、次のいずれかの操作でユーザーを管理できます。

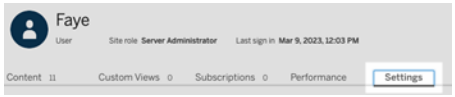
- グループ メンバーシップやサイトロールを設定したり、サイトからユーザーを削除したりします。サイトで SAML シングル サインオンを構成した場合、選択したユーザーの認証タイプを設定できます。サーバーが、パスワード ポリシーとアカウント アクセス ロックアウト用に構成されている場合は、ロックアウトされたユーザー アカウントへのアクセスのロックを解除できます。サーバーが 1 つ以上のアイデンティティプールを使用するように構成されている場合、ユーザーを追加または削除することができます。



- ユーザーの名前を選択し、それらのユーザーの所有コンテンツ、サブスクライブしているビュー、およびアカウント設定などの詳細を表示します。

  ユーザーの **[設定]** ページは、以下の条件に該当する場合に使用できます。

  - ユーザーが、管理者が管理するサイトのみのメンバーになっている
  - サイト管理者は、既定でユーザーを管理することができます。Tableau Server 管理者は、サイト管理者のこのアクセス権限を変更できます。



  サーバーが内部ユーザー管理システム (ローカル認証) を使用するよう構成されている場合、ユーザーを追加した後に**表示名**、**メール**、および**パスワード**を編集できます。多くの変更を行う場合は、CSV ファイルから変更をインポートする方が簡単な場合があります。詳細については、ユーザーのインポートおよびCSV インポート ファイルのガイドラインを参照してください。

サーバー ユーザーの表示および編集

Tableau Server にサーバー管理者としてサインインします。サイト メニューで、**[すべてのサイトを管理]**、**[ユーザー]** の順に選択します。

ユーザーのサイト メンバーシップの管理

サーバー管理者とサイト管理者は既定で、個々のサイトレベルでユーザーを管理できます。また、サーバー管理者は複数サイトのユーザーやサイトロールの管理を行うこともできます。この管理は **[すべてのサイト]** レベル (サーバーレベル) で実行します。

1. サイトメニューで、**[すべてのサイトを管理]**、**[ユーザー]** の順に選択します。
2. [サーバーユーザー] ページで、ユーザーの隣にあるチェックボックスを選択し、**[アクション] > [サイト メンバーシップ]** を選択します。
3. 1 つ以上のサイトと各サイトのサイトロールを選択してから、**[保存]** をクリックします。

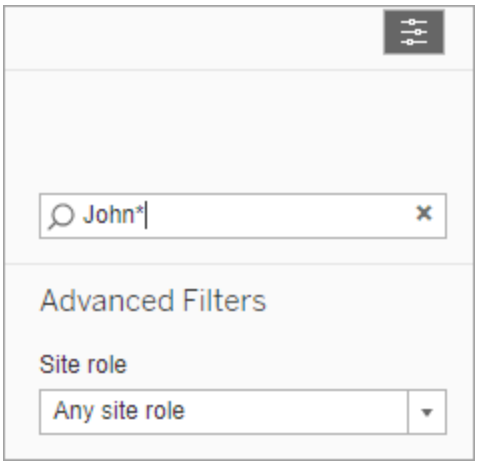
ユーザー (またはグループやサイト) の検索

特定のユーザー (またはグループやサイト) を検索するには、右上のフィルターの切り替えを使用して、検索ボックスとサイトロール フィルターを表示します。次に、検索ボックスまたはフィルターを使用して、目的のユーザー (またはグループやサイト) を検索します。検索オプションは、表示名とユーザー名属性を確認します。

検索ボックスは、ワイルドカード (*) 文字をサポートしています。たとえば、`John*` を検索すると、*John* で始まるすべての名前が返されます。

追加情報:

- Tableau Server 2021.4.1 以降、ワイルドカード文字 (*) を特殊文字と一緒に使用して、特殊文字を含む名前を検索することができます。たとえば、`sync-*` や `*sync-*` などです。
- Tableau Server 2022.1.13 以降、発音区分符号を含む名前を検索する場合は、名前に正確な発音区分符号を入力して、関連する結果が返されるようにする必要があります。たとえば、*José* を検索するには `José` と入力します。`Jose` と検索すると、結果が返されません。
- Tableau Server 2022.3.5 以降、ユーザー名をフィルタリングする際に、ワイルドカード文字 (*) に AND または OR 条件を含めて使用することができます。たとえば、`*aw* AND John*` と検索すると、ユーザー名に *aw* が含まれていて、かつ、名前が *John* で始まるすべてのユーザーが返されます。
    - AND または OR を使用したワイルドカードは、ユーザー名をフィルタリングする場合にのみ使用でき、表示名には使用できないことに注意してください。



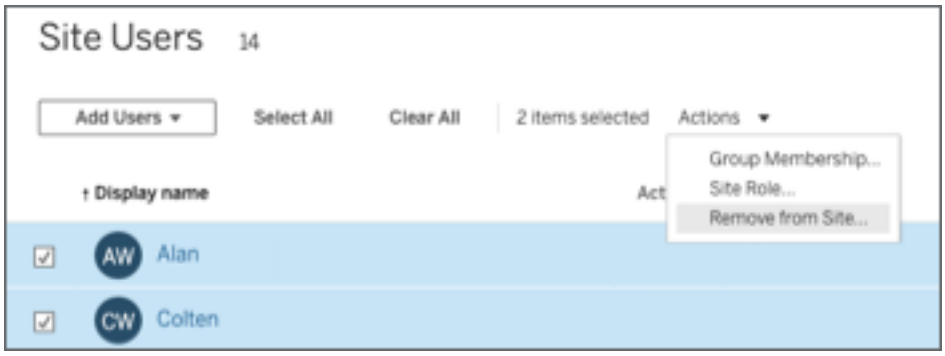
サイトからユーザーを削除

ユーザーを削除できるのは、そのユーザーがコンテンツ (プロジェクト、ワークブック、ビュー、データソース、コレクション、データアラートなど) を所有していない場合のみです。コンテンツを所有しているユーザーを削除しようとすると、そのユーザーのサイトロールが **[Unlicensed (ライセンスなし)]** に設定されますが、ユーザーは削除されません。

既定の [すべてのユーザー] グループで [Grant site role on sign in (サインイン時にサイトロールを付与)] が有効になっている場合、そのユーザーのコンテンツを別のユーザーに再割り当てするか削除してから、ライセンスの取り消しや削除を行う必要があります。サインイン時にロールを付与する方法については、「"サインイン時にロールを付与" の影響を受けるユーザーを削除する」を参照してください。コンテンツ所有権の変更の詳細については、「コンテンツ所有権の管理」を参照してください。コンテンツ所有権の管理

> **注:** Tableau Server で管理者がサイトからユーザーを削除すると (また、ユーザーが 1 つのサイトのみに属している)、ユーザーもサーバーから削除されます。

1. サイトに管理者としてサインインし、**[ユーザー]** ページに移動します。1 つまたは複数の削除するユーザーを選択し、**[アクション]**、**[サイトから削除]** の順に選択します。



2. 確認ダイアログ ボックスで **[ユーザーの削除]** をクリックします。

サーバーからユーザーを削除

ユーザーがコンテンツ (プロジェクト、ワークブック、ビュー、またはデータ ソース) を所有していない場合のみ、ユーザーを削除できます。コンテンツを所有しているユーザーを削除しようとすると、ユーザーのサイト ロールはライセンスなしに設定されますが、ユーザーは削除されません。

ユーザーが複数のサイトのメンバーで、いずれかのサイトでコンテンツを所有している場合、ユーザーはコンテンツを所有しないサイトから削除されます。ユーザーはコンテンツを所有しているサイトでは引き続きメンバーとなりますが、ライセンスなしのサイト ロールに引き下げられます。

1. サイト メニューで、**[すべてのサイトを管理]**、**[ユーザー]** の順にクリックします。シングルサイト環境では、**[ユーザー]** をクリックします。

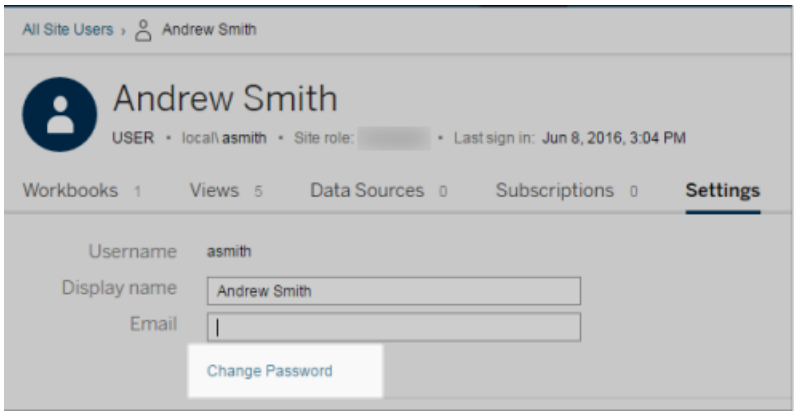
   削除するユーザーを 1 人または複数選択し、**[アクション] > [削除]** をクリックします。

2. 確認ダイアログ ボックスで **[削除]** をクリックします。

シングル サイトでのユーザーのパスワードの変更

シングル サイトへのメンバーシップを持つユーザーのパスワードを変更するには、サイト管理者またはサーバー管理者として Tableau Server にサインインします。

1. 正しいサイトがメニューで選択されていることを確認します。
2. **[ユーザー]** をクリックします。
3. ユーザーの表示名をクリックします。
4. **[設定]** をクリックします。
5. **[パスワードの変更]** リンクをクリックしてパスワードを編集してから、**[パスワードの保存]** をクリックします。

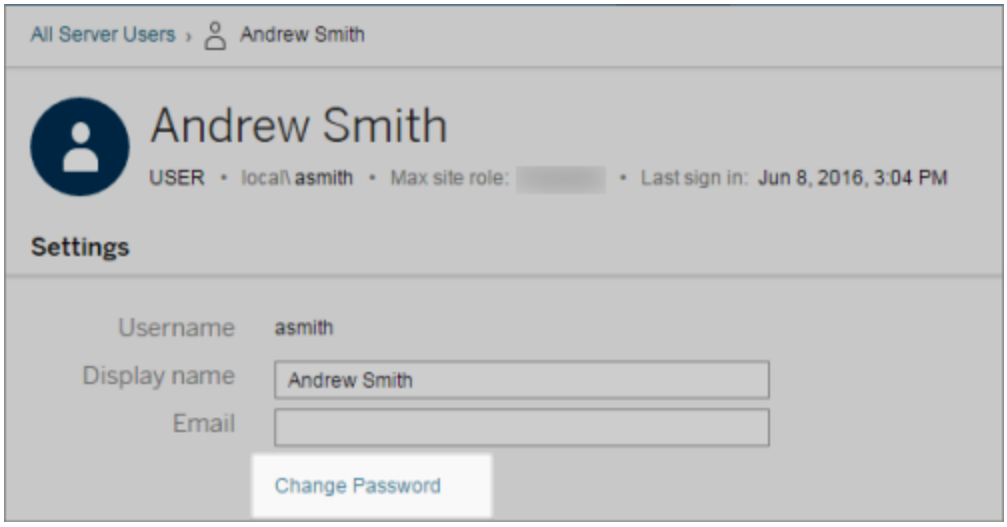


複数サイトのユーザーのパスワードの変更

複数サイトへのメンバーシップを持つユーザーのパスワードを変更するには、サーバー管理者として Tableau Server にサインインします。

1. サイト メニューで、**[すべてのサイトを管理]** をクリックします。
2. **[ユーザー]** をクリックします。
3. ユーザーの表示名をクリックします。

4. **[パスワードの変更]** リンクをクリックしてパスワードを編集してから、**[パスワードの保存]** をクリックします。

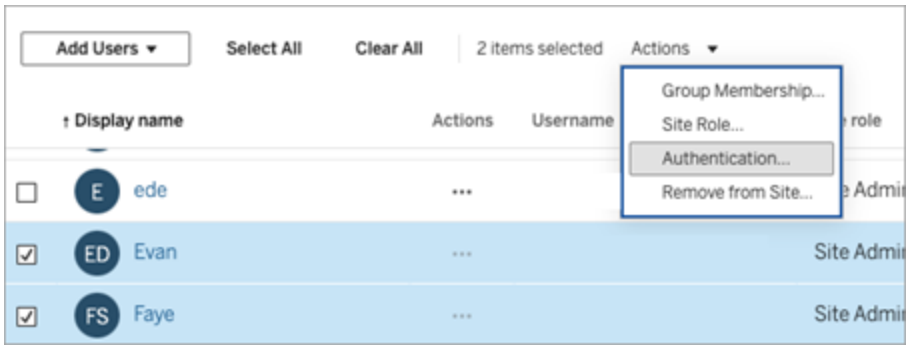


ユーザーの認証タイプを SAML 用に設定する

サイト固有の SAML 用に構成されているサイトでは、管理者がユーザーの認証タイプを指定できます。たとえば、Tableau Server がサイト固有の SAML およびサーバー全体の SAML がどのように構成したかによって、管理者はサイト固有の SAML を使用して認証するユーザー、およびサーバー全体の SAML を使用して認証するユーザーを指定できます。

認証方式は、Tableau Server へのユーザーの追加時やその後の任意のタイミングで割り当てることができます。

1. Tableau Server サイトにサインインし、**[ユーザー]** を選択します。
2. **[サイト][ユーザー]** ページで、認証タイプを割り当てたり変更したりするユーザーの隣にあるチェックボックスを選択します。
3. **[アクション]** メニューで、**[認証]** を選択します。



4. [認証] ダイアログ ボックスで **[サイト SAML]** または **[サーバー既定]** を選択します。

注

- サイト固有の SAML を使用して認証するユーザーは、1 つのサイトにのみ所属できます。ユーザーが複数のサイトに所属する必要がある場合、認証タイプをサーバーの既定に設定します。サーバー管理者によってサイト固有の SAML がどのように構成されたかによって、サーバーの既定はローカル認証またはサーバー全体の SAML のいずれかになります。
- ユーザーの認証をサイト固有の SAML に変更すると、そのユーザーが次にサインインする際、認証資格情報を入力するためにアイデンティティプロバイダーのサイトにリダイレクトされます。

ユーザーのインポート

ユーザー情報を含む CSV ファイルを作成し、そのファイルをインポートすることによって、サイトへのユーザーの追加プロセスを自動化することができます。

サイト管理者は、特定のサイトにユーザーをインポートできます。サーバー管理者 (Tableau Server のみ) は、サーバー レベルでユーザーをインポートし、後で複数のサイトに追加できます。

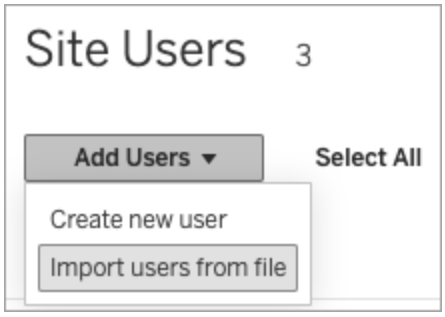
注: このトピックでは、既に CSV ファイルが作成済みであることを前提として、インポートの手順を説明します。まだファイルを作成していない場合は、CSV インポート ファイルのガイドラインでファイルの書式の要件とインポート オプションの一覧を参照してください。

CSV ファイルからのユーザーの追加

次の手順では、ユーザーをサイトまたはサーバーに追加する方法について説明します。イメージは、サイトレベルでのユーザーの追加を示しています。

1. 次のいずれかを実行します。

   - サイトレベルでユーザーを追加するには、**[ユーザー]** の次に **[ユーザーの追加]** を選択します。

   - **シングルサイト** サーバー上でサーバーレベルでユーザーを追加するには、**[ユーザー]** の次に **[ユーザーの追加]** を選択します。

   - **マルチサイト** サーバー上でサーバーレベルでユーザーを追加するには、サイトのリストを開き、**[すべてのサイトの管理]** を選択します。**[ユーザー]**、**[ユーザーの追加]** の順に選択します。

2. **[ファイルからユーザーをインポート]**、**[参照]** の順にクリックしてファイルに移動し、**[ユーザーのインポート]** をクリックします。

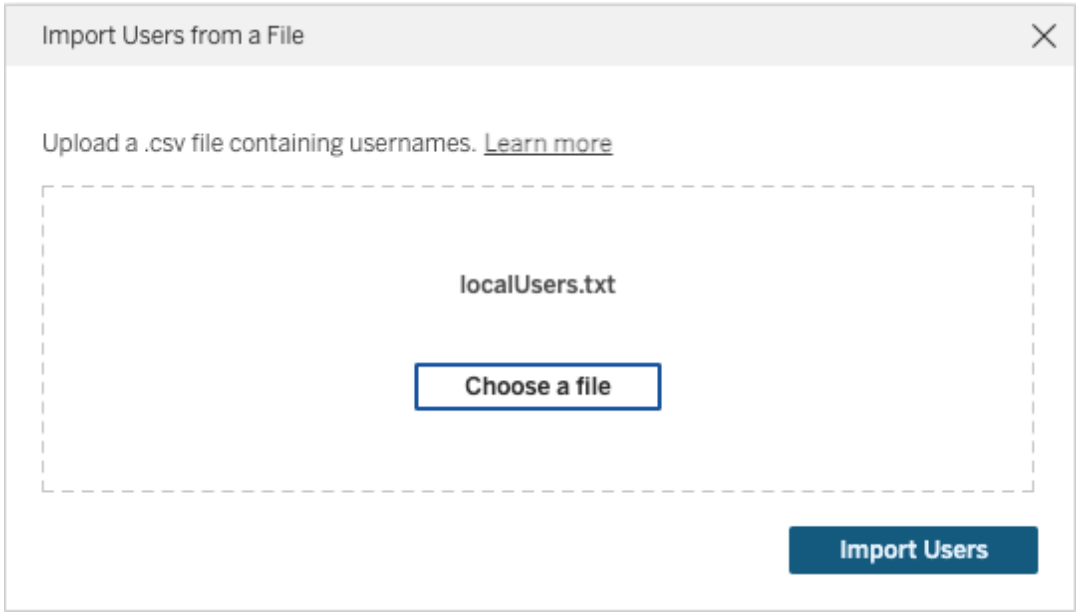
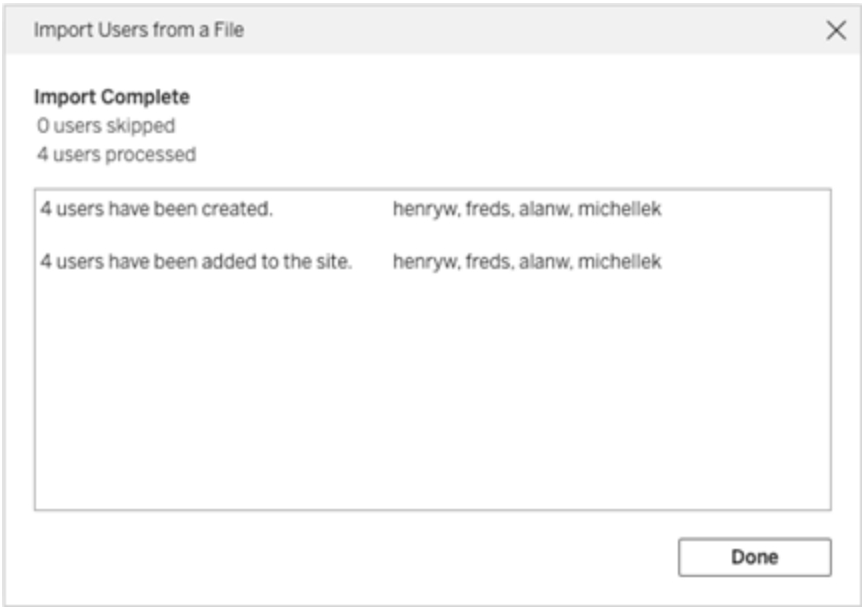


Tableau はインポートプロセスの結果を表示します。



3. **[完了]** をクリックします。

ユーザーのサイトロールの割り当てまたは維持方法

サイトレベルまたはシングルサイトサーバー上で tabcmd を使用してインポートする場合、CSV ファイル内のすべてのユーザーにサイトロールを指定することができます。ユーザーが Tableau Server サイトに既に存在する場合、インポートプロセス時に割り当てられたサイトロールの制限がユーザーの既存のサイトロールより厳しい場合でも適用されます。例外として、サーバー管理者のサイトロールに影響を与えることはできません。

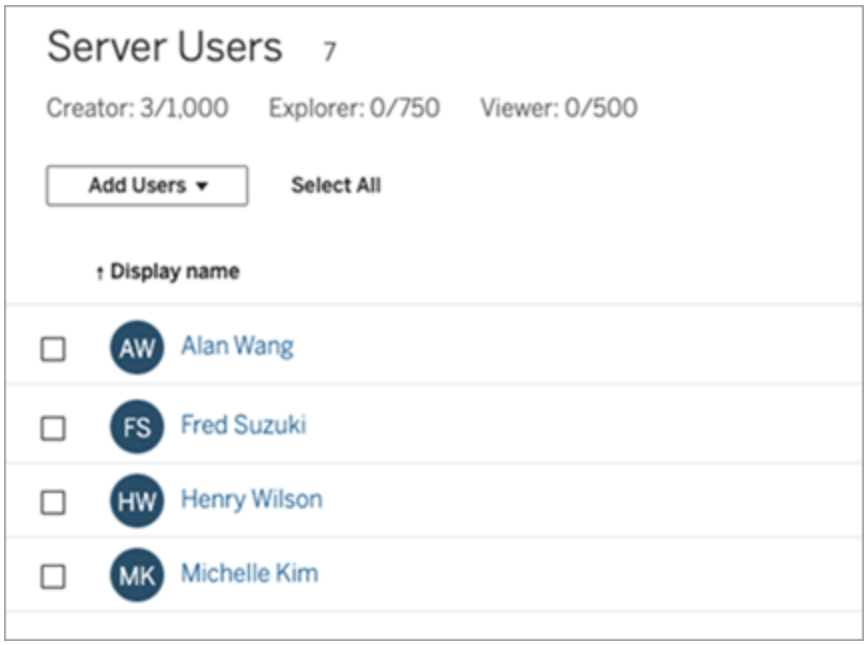
マルチサイトサーバーの場合、[サーバー ユーザー] ページでユーザーをインポートすると、サイトに所属しないサーバーユーザーが作成されます。これらのユーザーはサイトに属さないため、サイトロールを持つことができません。ユーザーがサーバー レベルで持つことのできるサイトロールは、**「ライセンスなし」**または**「サーバー管理者」**のみです。

ユーザーにサイト メンバーシップを割り当てるときに、ユーザーのサイトロールも指定することができます。詳細については、ユーザーのサイト メンバーシップの管理を参照してください。

マルチサイト環境におけるサーバー レベルでのインポート

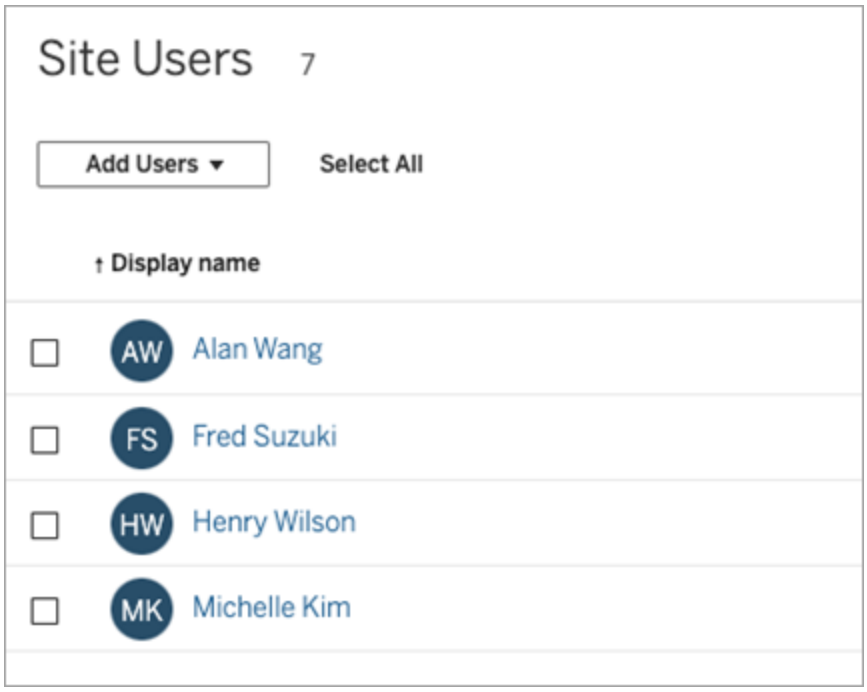
サーバーで複数のサイトが実行されており、ユーザーがサーバー管理者である場合、2 つの場所から CSV ファイルをインポートできます。既存のユーザー アカウントが関係する場合、それぞれの場所は権限が異なります。

- **[サーバー ユーザー]** ページはマルチサイト環境で表示されます。サーバー管理者のみがこのページにアクセスできます。



新しいユーザーの追加だけではなく、既存のユーザー アカウントを更新する場合は、ここから CSV ファイルをインポートすることができます。たとえば、既存ユーザーの新しいパスワードを入れたファイルをインポートすると、既存ユーザーのパスワードがリセットされます。

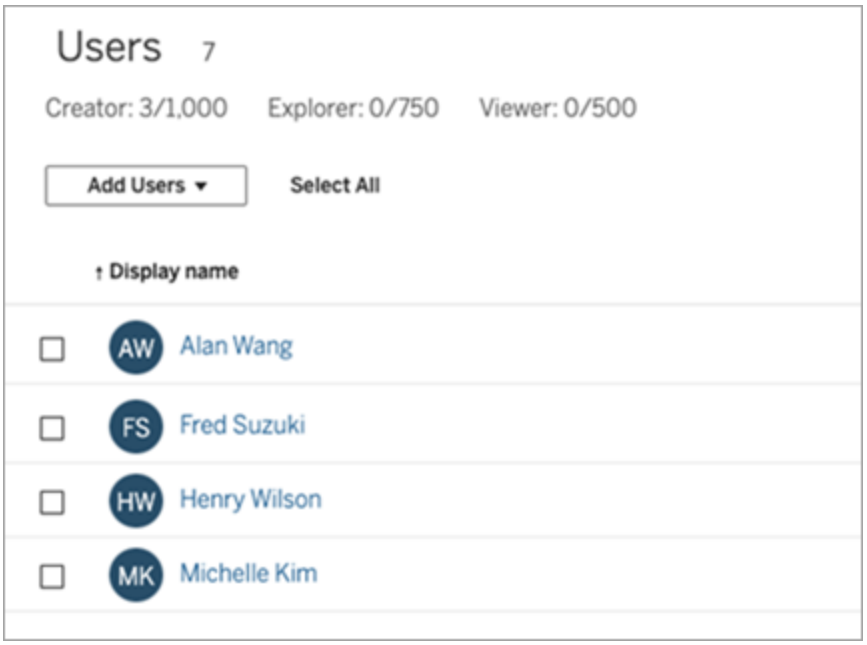
- **[サイト ユーザー]** ページです。



サーバー管理者は、CSV インポートにより新しいユーザー アカウントを追加することができます。CSV ファイルに既存のユーザーが含まれている場合、**[パスワード]** フィールドと**[表示名]** フィールドが既存の内容と一致しているか、または空白になっている必要があります。新しいパスワードまたはフルネームが使用されると、インポートは失敗します。

シングルサイト環境へのインポート

シングルサイト サーバーのサーバー管理者およびサイト管理者は、CSV ユーザー インポートをサイトの **[ユーザー]** ページから実行します。



マルチサイト インポートとシングルサイト インポート

ユーザーは同じサーバーの複数のサイトに属することができますが、各サイトで同じ認証資格情報を使用する必要があります。これは、ユーザーをサイトに追加し、それらのユーザーが既に別のサイトのメンバーの可能性がある場合に重要です。既に存在するユーザーをインポートしようとしていて、CSV ファイル内のユーザーの認証資格情報が既存の認証資格情報と一致しない場合、そのユーザーのインポートは失敗します。

ユーザーをあるサイトにインポートしているものの、そのユーザーがサーバー上に既に存在する可能性がある場合、試しに CSV ファイルの **Password** フィールドを空白のまま残すことができます (フィールドの区切り文字は含めます)。CSV で定義されたユーザーが別のサイトに既に存在する場合、そのユーザーはインポートしようとしているサイトに追加されます。ただし、そのユーザーがサーバー上に存在しない場合は、ユーザーが作成され、CSV インポート ウィンドウに新しいユーザーのパスワードがありませんというアラートが表示されます。その場合は、サーバー管理者のページを使用して、パスワードを持たないユーザーにパスワードを割り当てることができます。

注: サーバーが認証に Active Directory を使用するよう構成されている場合、ユーザー パスワードは Active Directory によって管理されるため、CSV ファイルのパスワード フィールドを空欄のままにすることができます。

CSV インポート ファイルのガイドライン

ユーザー情報を格納したコンマ区切りの値 (CSV) ファイルを作成し、それをインポートすることによって、ユーザーを一括で追加できます。CSV ファイルには、ライセンス レベルやパブリッシュ アクセスなど、インポートと同時にユーザーに適用する属性を含めることができます。

このトピックでは、CSV ファイル自体の詳細について説明します。CSV を使用してユーザーをインポートするには、次の方法があります。

- サーバーまたはサイトの管理ページを使用します。詳細については、「ユーザーのインポート」を参照してください。
- `tabcmd` ユーティリティを使用します。詳細については、「createsiteusers filename.csv」を参照してください。

ユーザーはサイトまたはサーバーのレベルでインポートできます。ユーザーをサーバーに (特定サイトにではない) インポートする場合、ユーザーはサイトに割り当てされず、Unlicensed としてインポートされます。

注: 特に明記されていない限り、このトピックで指定されているガイドラインは、アイデンティティ プールを構成時に使用したかどうかにかかわらず、Tableau Server に適用されます。

CSV ファイル形式の要件

ユーザーをインポートするための CSV ファイルには、次の書式要件があります。

- 列の見出しはありません。Tableau Server ファイル内の各行を 1 人のユーザーと見なします。
- バイト オーダー マーク (BOM) 付きの UTF-8 形式は、サイトにインポートする場合にのみサポートされます。Tableau Cloud Manager にインポートする場合は、UTF-8 形式を使用し、ファイルから BOM を削除してください。

- BIG-5 などの文字エンコーディングは UTF-8 に変換する必要があります。この操作を行うには、テキストエディターでファイルを開き、**[名前を付けて保存]** コマンドを使用します。
- @ 文字がドメインセパレーター以外の**ユーザー名**フィールドに存在する場合は、16 進形式 (\0x40) にする必要があります。たとえば、user@fremont@mycompany.com は user\0x40fremont@mycompany.com にする必要があります。

CSV ファイルの必須の列構造

ファイルにはヘッダー行を含めないでください。すべての列に入力する必要はありませんが、すべてのコンマ区切り記号をそのまま含めることによって、必須の列構造を保持する必要があります。

列構造は、製品やインポートツールによって異なります。環境に適した構造を使用していることを確認します。

- **Tableau Cloud:** ユーザー名、パスワード、表示名、ライセンスレベル、管理者レベル、パブリッシュ機能、メールアドレス
- **Tableau Cloud Manager (TCM) を使用する Tableau Cloud:** テナントまたはサイト URI、ユーザー名、パスワード、表示名、ライセンスレベル、管理者レベル、パブリッシュ機能、メールアドレス、認証タイプ
- **Tableau Server:** ユーザー名、パスワード、表示名、ライセンスレベル、管理者レベル、パブリッシュ権限、メールアドレス
- **アイデンティティプールを使用する Tableau Server:** ユーザー名、パスワード、表示名、ライセンスレベル、管理者レベル、パブリッシュ機能、メールアドレス、アイデンティティプール名、識別子

各列に指定できる値の詳細については、以下の「列の詳細」セクションを参照してください。

重要: 列の順序には重要な意味があります。列のコンテンツに関係なく、最初の列はユーザー名、2 番目の列はパスワード、3 番目の列は表示名などと表示されます。フィールドの値を省略する場合でも、フィールドのコンマ区切り記号は含める必要があります。

Tableau Server の列の詳細

> **注**: 特定のサイトにサインインしておらず、ユーザーをサーバー レベルでインポートしている場合は、サーバー管理者とUnlicensed のサイト ロールのみを割り当てることができます。

CSV ファイルの列構造: `username, password, display name, license level, admin level, publishing capability, email address`

列の詳細:

Username: *必須*。ユーザーの名前。

- サーバーが Active Directory を使用するように設定されている場合、この値は、Active Directory で定義されたユーザーと一致する必要があります。ユーザー名がドメイン全体で一意でない場合、ユーザー名の一部としてドメインを含めます (例: `example\Adam` または `adam@example`)。
- アイデンティティプールにユーザーを追加する場合は、以下を確認してください。
  - AD をアイデンティティストアとして使用するアイデンティティプールにユーザーを追加する場合は、必ずユーザー名に AD sAMAccountName 値を使用してください。
  - アイデンティティストアとして LDAP を使用するアイデンティティプールにユーザーを追加する場合は、必ずユーザー名に LDAP ユーザー名の値を使用してください。

Password: ユーザーのパスワード。この列が必須かどうかは、サーバーの構成によって異なります。

- ローカル認証: *必須*。Tableau Server 2024.2 より、必須パスワード フィールドの適用が開始されました。詳細については、「Tableau Server 2024.2 で tabcmd 経由でユーザーを追加しようとすると、予期しない「errorCode=134」が発生する」のナレッジ記事を参照してください。
- Active Directory: *無視されます*。ベストプラクティスとして、この列は空白のままにします。

Display name: 表示名は、サーバー上のユーザーを識別するのに使用される情報の一部です。この列が使用されるかどうかは、サーバーの構成によって異なります。

- ローカル認証:ユーザーの表示名がすでに使用されている場合、Tableau Server は、CSV ファイル内の設定で既存のユーザー情報を更新します。

- Active Directory: *無視されます*。ベストプラクティスとして、この列は空白のままにします。

License level: Creator、**Explorer**、**Viewer**、または**Unlicensed**のいずれかです。

- ライセンスレベルが Creator の場合、パブリッシュ機能は Yes/True/1 である必要があります。
- ユーザーベースのサーバーがインストールされていて、ユーザーを追加するとライセンスで許可されるユーザー数を超える場合は、ユーザーはライセンスなしのユーザーとして追加されます。
- `tabcmd` を使用し、ライセンスを指定しているものの、インポートするユーザーがライセンスの制限を超過する場合は、ユーザーはライセンスなしのユーザーとしてインポートされます。

Administrator level: System、**Site**、または**None**のいずれかであり、ユーザーが管理者としてインポートされるかどうかを決定します。

- System = サーバー管理者、Site = サイト管理者、None = 管理者ではない、を示します。
- 特定のサイトにサインインしておらず、ユーザーをサーバーレベルでインポートしている場合は、サーバー管理者とライセンスなしのみを割り当てることができます。
- Web UI を使用してユーザーをインポートする場合、サーバー (全サイト) レベルでファイルのインポートを実行する場合にのみ、管理者サイトロールを System に設定できます。特定のサイトにサインインしている場合、および CSV ファイル内のユーザーの管理者列が System に設定されている場合は、Tableau Server はユーザーをサイト管理者としてインポートします。

Publishing capability: Yes/True/1 または **No/False/0** のいずれかです。Web UI を使用している場合、特定のサイトにサインイン中にインポートする場合にのみパブリッシュ機能が使用されます。

- ライセンスレベルが Creator の場合、パブリッシュ機能は Yes/True/1 である必要があります。
- ライセンスレベルが Explorer の場合、パブリッシュ機能の値によって Explorer (パブリッシュ可能) と Explorer のサイトロールが区別されます。
- ライセンスレベルが Viewer またはライセンスなしの場合、この列の値に関係なく、ユーザーはパブリッシュすることはできません。

Email address: メールアドレスは、サーバー上のユーザーを識別するのに使用される情報の一部です。電子メールアドレスがすでに使用されている場合、Tableau Server は、CSV ファイル内の設定で既存のユーザー情報を更新します。

空白のままの列は、列構造を保持するためにコンマで区切る必要があります。

アイデンティティプール

アイデンティティプールにユーザーを追加する場合は、標準の列に加えて以下の列が必要です。

CSV ファイルの列構造: `username, password, display name, license level, admin level, publishing capability, email address, identity pool name, identifier`

追加の列の詳細:

アイデンティティプール名: ユーザーの追加先のアイデンティティプールの名前。アイデンティティプールにユーザーを追加する際にアイデンティティプール名を指定しない場合、ユーザーは初期プール (TSM で構成済み) に追加されます。これは Tableau Server のセットアップ中に TSM でプロビジョニングされたユーザーのセットです。

識別子: 追加するユーザーの識別子。識別子は、アイデンティティ照合のためだけに使用されます。詳細については、「Tableau におけるユーザー名と識別子」を参照してください。

- ローカル認証: *オプション*
- Active Directory または LDAP: *必須*

アイデンティティプールにユーザーを一括で追加

CSV インポートプロセスを使用すると、次のことができます。

- ユーザーを追加のアイデンティティプールに一括で追加します。**注:** CSV インポートプロセスを使用して、ユーザーがすでに属しているアイデンティティプールを別のアイデンティティプールに置き換えることはできません。異なるアイデンティティプール値を持つ既存のユーザーを追加すると、そのユーザーは追加のアイデンティティプールに追加されます。
- まだ識別子を持っていないユーザーに識別子を一括で追加します。**注:** 同じプール内のユーザーに別の識別子を追加しても、そのユーザーの既存の識別子は置き換えられません。代わりに、そのユーザーに対して新しい識別子レコードが作成されます。

Tableau Server の CSV インポートの例

ヒント: 構造は `username, password, display name, license level, admin level, publishing capability, email address` であることを忘れないでください。

次の例は、複数のユーザー用の情報を含む CSV ファイルを示しています。

```
henryw,henrypassword,Henry
Wilson,Creator,None,yes,henryw@example.com
freds,fredpassword,Fred Suzuki,Viewer,None,no,freds@example.com
alanw,alanpassword,Alan Wang,Explorer,Site,yes,alanw@example.com
michellek,michellepassword,Michelle
Kim,Creator,System,yes,michellek@example.com
```

- サイトを管理中にこのファイルをインポート場合、そのサイトに 4 人のユーザーが追加されます。ユーザー Michelle の `Administrator` 設定は `System` です。ただし、サイトにユーザーをインポートしているため、Tableau Server は Michelle をサイト管理者 Creator サイトロールに設定します。ユーザーのうち 3 人はパブリッシュが許可されています。
- サーバーを管理中にこのファイルをインポート場合、そのサーバーに 4 人のユーザーが追加されますが、それらのユーザーはサイトには追加されません。1 ユーザーだけがサーバー管理者としてインポートされ、残りはライセンスなしに設定されます。

アイデンティティプールの例

ヒント: 構造は `username, password, display name, license level, admin level, publishing capability, email address, identity pool name, identifier`であることを忘れないでください。

次の例は、アイデンティティプールに追加された 2 人のユーザーの情報を含む CSV ファイルを示しています。

```
henryw,henrypassword,Henry
Wilson,Viewer,None,yes,hwilson@myco.com,General Contractors,hwilson
freds,fredpassword,Fred
Suzuki,Creator,None,no,fsuzuki@myco.com,General Contractors,fsuzuki
```

次の例は、追加のアイデンティティプールに追加された 2 人のユーザーの情報を含む CSV ファイルを示しています。

```
henryw,henrypassword,Henry
Wilson,Viewer,None,yes,hwilson@myco.com,General Contractors
2,hwilson
freds,fredpassword,Fred
```

```
Suzuki,Creator,None,no,fsuzuki@myco.com,General Contractors
2,fsuzuki
```

次の例は、既存の識別子を持たない 2 人のユーザーの情報を含む CSV ファイルを示しています。

```
janes,janepassword,Jane
Smith,Viewer,None,yes,jsmith@myco.com,General Contractors,jwang
laurar,laurapassword,Laura
Rodriguez,Creator,None,no,lrodriguez@myco.com,General
Contractors,jrodriguez
```

tabcmd に渡される大きな CSV ファイルのパフォーマンス改善

> **注:** これらの設定は、Tableau Server バージョン 2022.1 以前に適用されます。影響を受ける検索およびインデックス サービスは、バージョン 2022.3 以降で非推奨になり、2023.3 で廃止 (完全に削除) されました。

サーバー管理者は tabcmd コマンドを使用して、大きな CSV ファイルのインポート時にパフォーマンスを改善できるサーバー設定を有効にできます。これは `tsm configuration set` コマンドを次のオプションとともに使用して可能です。

- `vizportal.csv_user_mgmt.index_site_users`
- `vizportal.csv_user_mgmt.bulk_index_users`
- `searchserver.index.bulk_query_user_groups`

基本的に、これらのオプションを 1 つずつサーバーのデータベースに追加していくのではなく、CSV ファイルを 1 つ処理した後にインデックスを作成します。これにより、データベースを呼び出す回数も、ファイルの処理に必要となるメモリも減ります。これらの `tsm configuration set` オプションは、`tabcmd createsiteusers`、`deletesiteusers`、`addusers`、および `removeusers` コマンドに適用されます。

これらの設定の説明については、tsm configuration set のオプションを参照してください。

CSV 設定とサイトロール

ユーザーのライセンスレベル、管理者、およびパブリッシュ設定が、ユーザーのサイトロールがインポートプロセス中にどのように設定されるかを決定します。次に、サイトロールをCSV に取り込む方法を示します。

サイトロール	ライセンス レベル	管理者レベル	Publishing capability
サーバー管理者 *このロールは、サーバーを管理しているとき(つまり、特定のサイトにサインインしていないとき) にユーザーをインポートする場合にのみ有効です。*	ただし、Creator ライセンスが利用可能な場合は、常にそのライセンスがサーバー管理者サイトロールに適用されます。利用できる Creator ライセンスがない場合は、ライセンス発行のトラブルシューティング を参照して Tableau Server がこれをどのように処理するかについて学んでください。	System	True
サイト管理者 Creator または **サイト管理者 Explorer** *これらのロールは、特定のサイトへのサインイン中にユーザーをインポートする場合にのみ有効です。*	Creator または Explorer	Site	True
Creator	Creator	None	True
Explorer (パブリッシュ可能)	Explorer	None	True
Explorer	Explorer	None	False

Viewer	Viewer	None	False
Unlicensed	Unlicensed	None	False

サイトユーザーの表示状態の管理

既定では、パーミッションにより許可されている場合にはすべてのサイトユーザーが別名、プロジェクトの所有権、他のユーザーによるコメントを表示することができます。[ユーザーの表示状態] 設定を使用すると、管理者は、Viewer (ビューアー) や Explorer のサイトロールを持つユーザーにサイト上の他のユーザーやグループが表示されるかを管理することができます。これは複数のクライアントにより使用されているサイトで重要な場合があります。サイトロールの詳細については、ユーザーのサイトロールの設定を参照してください。

ユーザーの表示状態の制限

[ユーザーの表示状態] を **[Limited (制限)]** に設定すると、特定のコラボレーション ツールに影響が及び、Tableau Cloud および Tableau Server のユーザー情報が非表示になります。ユーザーの表示状態を制限すると、Viewer (ビューアー) および Explorer (サイト管理者 Explorer を除く) の機能が無効になるか、他の領域からユーザー情報が削除されます。[ユーザーの表示状態] が [Limited (制限)] に設定されている場合でも、Creator と管理者にはユーザー情報が表示されます。

Explorers とViewer (ビューアー) (サイト管理者 Explorer を除く) でのユーザーの表示状態を制限するには、以下の操作を行います。

- サイトの **[設定]** ページに移動します
- **[ユーザーの表示状態]** 設定で **[Limited (制限)]** を選択します

[ユーザーの表示状態] を [Limited (制限)] に設定する場合に影響があるサイト領域のリストを次に示します。この機能がすべてのユーザーで無効になるという記載がなければ、管理者以外の Explorer または Viewer (ビューアー) のみが影響を受けます。

領域	影響
検索	ユーザー情報が表示されない

コンテンツ所有者	ユーザー情報が表示されない (Explorer とViewer (ビューアー) は自分を表示できないが、[マイ コンテンツ] でコンテンツを確認できる)
プロフィール写真	ユーザー情報が表示されない
サブスクリプション	ユーザー情報が表示されない
推奨事項	類似ユーザーが表示されない (すべてのユーザー)
タグの追加/編集	Explorer とViewer (ビューアー) はタグを表示できるが、削除や変更ができない
"このビューを表示したユーザー"	無効
「データに聞く」(Ask Data) の使用状況分析	無効
パーミッション ダイアログ	無効
名前付き共有	無効 (すべてのユーザー)
アラート	無効 (すべてのユーザー) 既存のアラートが一時停止する
コメント	無効 (すべてのユーザー)
パブリック カスタム ビュー	無効 (すべてのユーザー) 既存のパブリック カスタム ビューがプライベートとして表示される
アクセス権の要求	無効 (すべてのユーザー)
Tableau Desktop	Desktop でのワークブックのパブリッシュが無効になっている

	ユーザー フィルターにユーザー情報が表示されない
Tableau Pulse	フォロワーを表示および管理するためのボタンがメトリクスに表示されない
Tableau Catalog (データ管理 に付属)	ユーザー情報が表示されない

[ユーザーの表示状態] を [制限つき] に設定すると、Tableau Server の REST API およびメタデータAPI の呼び出しは上記の表の説明に従って動作します。既知のユーザーID を参照するAPI 呼び出しを行うことは引き続き可能です。

サイトのユーザーは、ビューを操作して、フィルターの適用などのビューを変更できます。そのユーザーが変更したビューを他のユーザーと共有する場合、またはユーザーがその変更されたビュー (指標またはプライベートカスタムビューなど) から何かを作成した場合、そのユーザーの名前が URL に表示されます。この変更されたビューの URL は、そのユーザーの名前を表示できるユーザーにのみ配布されるようにしてください。

注: ユーザーが複数のサイトのメンバーである場合、Tableau Cloud のサインインページにメールを入力すると、ユーザーがメンバーになっているすべてのサイトの名前が返されます。

ユーザーの表示状態の制限に関するベストプラクティス

管理者は、次の方法でユーザーとグループの情報が表示されないことも確認できます。

- 適切な関係者にコンテンツの提供のみを行うパーミッションを構成する。詳細については、パーミッションを参照してください。
  - ユーザーの表示状態が制限されていると、ユーザーID 情報は検索しても表示されませんが、検索しているユーザーにコンテンツに対する表示パーミッションがある場合には、所有者の名前で検索を行うとユーザーがパブリッシュしたコンテンツが返される場合があります。
  - 同じプロジェクト内でタイトルが重複したワークブックをパブリッシュするユーザーには、そのタイトルのワークブックが既に存在するという警告が表示されることがあります。
- 必要に応じて行レベルのセキュリティを適用します。
- ダッシュボード内のメタデータにユーザー情報が含まれていないことを確認します。

- ユーザーがアクセスできる計算にユーザー メタデータが含まれていないことを確認します (ユーザー フィルターなど)。

ユーザーの表示状態を [完全] に戻す

管理者が [ユーザーの表示状態] を [完全] に戻したとき、[ユーザーの表示状態] を [Limited (制限)] に設定してすべてのユーザーに対して無効にした機能 (コメントやアラートなど) はオフのままです。管理者は、サイトの [設定] ページでこれらの機能をもう一度有効にすることができます。

[ユーザーの表示状態] を [完全] に設定したとき、以前の機能設定は保持されず、影響を受ける機能は自動的にオンになりません。

ゲスト ユーザー

Tableau Server のコアベースのライセンスには [ゲスト ユーザー] オプションが含まれています。これにより、ユーザーがサーバーのアカウントを持っていなくても Tableau のビューにアクセスできるようになります。

Tableau Server がコアベースのライセンスでインストールされている場合、ゲスト ユーザーのアクセスが既定で有効になっています。ユーザーベースのライセンスでは利用できません。ゲスト ユーザーのアクセスを使用しない場合は、無効にする必要があります。

ゲスト アクセスは、ユーザーに対して Tableau ビューの表示および操作のみを許可します。ゲスト ユーザーは Tableau Server インターフェイスを参照できません。また、ビューを操作するサーバー インターフェイス エレメントは表示されません (ユーザー名、アカウント設定、コメントなど)。ライセンスの詳細については、ライセンスの管理を参照してください。

> **ヒント:** ビューをゲスト ユーザーと共有するには、Web ページに URL リンクまたは埋め込みビューを指定します。詳細については、「Tableau ユーザー ヘルプ」を参照してください。

ゲスト ユーザー パーミッション

ゲスト ユーザーには次の最大の権限が与えられる可能性があります。

- **ワークブックおよびビュー**: 表示、イメージのエクスポート、サマリー データ、コメントの表示、フィルター、フル データ、Web 編集、ダウンロード (ローカル コピーを保存するため)
- **データ ソース**: 表示およびダウンロード

コンテンツ リソースに対するより高いレベルのアクセス権があるグループにゲスト ユーザーを追加する場合、ゲスト ユーザーのアクセス権が上記の権限を上回ることはありません。ですが、ゲスト ユーザー アカウントは、制限がより厳しい方のパーミッション設定に従います。

ゲスト ユーザーは、認証資格情報が埋め込まれているデータ、または認証資格情報が不要なデータにのみアクセスできます。ゲスト ユーザーは、ユーザーに認証資格情報の入力が必要なデータ ソースにはアクセスできません。「パブリッシュされた Tableau データ ソースに対するデータ アクセス」を参照してください。

ゲスト アクセスの有効化または無効化

サーバー レベルまたはサイト レベルでゲスト アカウント設定を変更するには、サーバー管理者である必要があります。

> **注:** サイトのゲスト ユーザーを有効にすると、潜在的な同時ビューアーの数が予期していたユーザー リストの数を超える可能性があります。管理ビューの **[ステータス] > [ビューへのアクセス量]** は、このアクティビティを判断する際に役立ちます。

1. サイト メニューで、**[すべてのサイトを管理] > [設定] > [全般]** の順にクリックします。
2. **[ゲスト アクセス]** で、**[ゲスト アクセスを有効にする]** をオンまたはオフにします。
3. **[保存]** をクリックします。

これにより、すべてのサイトでゲスト ユーザーが有効になります。その後、特定のサイトの同じ設定に移動できます。サイトのゲスト アクセスを却下するには:

1. サイト メニューで、サイトを選択します。
2. **[設定]** をクリックし、[全般] タブで、**[このサイトのゲスト アクセスを有効にする]** チェック ボックスをオフにします。

一部またはすべてのサイトでゲスト アカウントが有効化されている場合、サーバー レベルでゲスト アクセスをオフにすると、すべてのサイトでもオフになります。

注: ゲストユーザーのアクセスをサーバーレベルまたはサイトレベルでオフにすることはできますが、ユーザーを削除することはできません。それで、サーバーにサインインせずにデータやビューにアクセスすることは誰にもできませんが、ゲストユーザーは [サイトユーザー] リストおよびゲストユーザーの追加先であるグループのグループリストに表示されます。

ゲストアカウントのその他の特徴

他にも、ゲストユーザーは次の点で一意です。

- 単一ユーザーアカウントとして、Tableau ビューにアクセスする認証されていないすべてのユーザーを代表します。
- 有効になっている場合は、[すべてのユーザー] グループのメンバーです。
- ゲストユーザーをサイト上の他のグループのメンバーとして追加できます。
- コンテンツリソースの所有者として編集または選択することはできません。
- このアカウントでは、カスタムビューの保存は許可されていません。
- ユーザーフィルターでゲストを使用することはできません。
- アカウントを削除することはできませんが、上記のステップで説明したチェックボックスをクリアすることで、アクセスをオフにすることができます。

グループセットの操作

Tableau Server 2024.2 以降、グループセットを使用してグループのコンテナを作成できます。グループセットには 1 つまたは複数のグループを含めることができ、グループセットを基にしたコンテンツのパーミッションに、より詳細なルールを適用できます。グループセットを基にして機能を有効にする場合、グループセットに属するグループのユーザーが、機能が評価されるすべてのグループのメンバーである必要があります。このように、グループセットは AND ロジックを適用します。

グループセットの利点:

- 同期されたグループとローカルグループをパーミッションルールで組み合わせて、より動的なアクセス制御のシナリオを実現できます。
- パーミッションルールのグループに AND ロジックを使用すると、シナリオによってはアクセス制御を簡素化できます

注:

- グループ セットのパーミッション ルールは、ユーザー ルールとグループ ルールの後に評価されます。これらのルールの詳細については、「パーミッション ルールの評価」を参照してください。
- グループ セットは、サーバー管理者のみが作成できます。
- グループ セットをグループのグループとして使用しないでください。

グループ セットをオンにする

グループ セットをパーミッションに使用するには、グループ セット設定を有効にする必要があります。

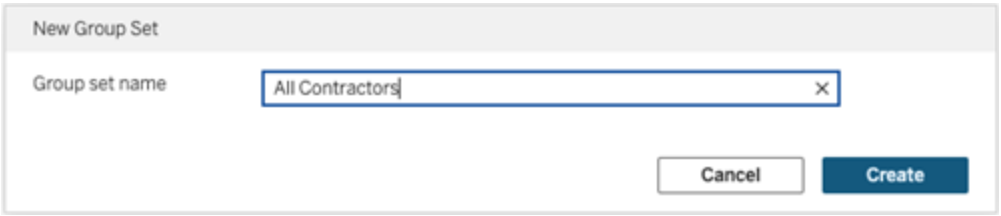
1. Tableau Server にサーバー管理者としてサインインします。
2. **[設定]** ページへ移動します。
3. [グループ セット] セクションで、**[グループ セットを許可する]** チェック ボックスをオンにします。

グループ セットを有効にすると、ナビゲーション ペインに専用の **[グループ セット]** ページが表示されます。

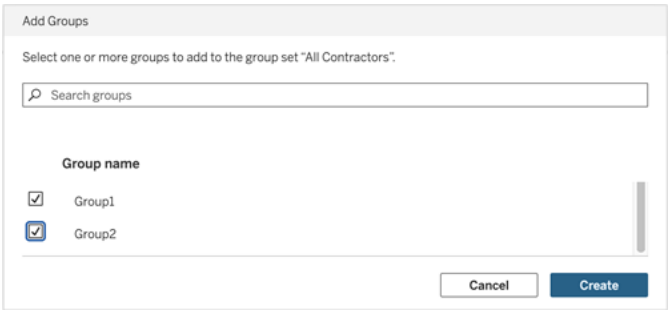
グループ セットを作成する

グループ セットを作成するには、[グループ セット] ページに移動し、グループの場合と同じようにグループ セットを作成します。

1. Tableau Server にサーバー管理者としてサインインします。
2. [グループ セット] ページに移動し、**[新しいグループ セット]** ボタンをクリックします。
3. グループ セット名を入力し、**[作成]** をクリックします。



4. [グループ セット] テーブルで、作成したグループ セット名をクリックし、**[グループの追加]** ボタンをクリックします。
5. 利用可能なグループのリストから、グループ セットに追加するグループを選択し、**[追加]** ボタンをクリックします。



グループ セットのパーミッションを設定する

サイト管理者、プロジェクト リーダー、またはコンテンツ所有者がグループ セットを使用するには、グループ セットを使用するコンテンツのパーミッションを追加または編集します。

たとえば、あなたが「打者」ワークブックの所有者であるとします。グループ セットを基にしたパーミッションを適用するには、次の手順を実行します。

1. ワークブックに移動し、アクション メニューから**[パーミッション]** を選択します。
2. [パーミッション] ダイアログ ボックスで、**[グループ/ユーザー ルールの追加]** ボタンをクリックし、次の操作を行います。
   a. テキスト ボックスに、「すべての契約者」などのグループ セット名を入力します。
   b. 必要な機能をテンプレートで選択します。
   c. **[保存]** をクリックします。

グループ セット モデルを使用してパーミッションを適用する場合は、グループ セット内のすべてのグループのメンバーシップを持つユーザーに影響を及ぼすパーミッションを作成します。

たとえば、ユーザーが所属する地域グループに基づいて、異なるワークブック ビューへのアクセスを制限することができます。これらのワークブック ビューにアクセスする、契約社員、契約社員 マネージャー、フルタイム社員、フルタイム マネージャーのユーザーがいるとします。

ワークブックを表示するための既定のパーミッションを持つグループ: すべて、北部、南部、東部、西部、フルタイム、およびマネージャー。

- 北部地域の詳細 ビュー:
  - パーミッションはグループ セットに基づいています: 北部地域
  - グループ セット内のグループ: フルタイム、北部

  結果: [北部地域の詳細] ビューでデータを表示できるのは、北部地域のフルタイム社員とフルタイム マネージャーのみです。
- 全地域の詳細 ビューの場合:
  - グループ セット名はマネージャーです
  - グループ セット内のグループ: すべて、マネージャー

  結果: [全地域の詳細] ビューでデータを表示できるのは、契約社員 マネージャーとフルタイム マネージャーのみです。

パーミッションの詳細については、マネージド セルフサービスのためのプロジェクト、グループ、グループ セット、パーミッションの設定を参照してください。

グループ

ユーザー グループの作成と削除、グループへのユーザーの追加、および Active Directory とのグループの同期を行うことができます。

グループへのユーザーの追加

Tableau Server ユーザーをグループに整理して、複数のユーザーを管理しやすくできます。グループはサーバーでローカルに作成することも、Active Directory からインポートすることもできます。

Active Directory などの外部のアイデンティティ ストアを使用してユーザーを管理している場合は、外部のアイデンティティ ストア自体を介してユーザーをグループに追加します。ユーザーが外部のアイデンティティ ストアのグループに追加されると、Tableau Server は、外部のアイデンティティ ストア

のユーザー グループを Tableau Server のユーザー グループと同期することにより、ユーザーを更新できます。

たとえば、Active Directory グループのメンバーシップを最新の状態に保つには、次を確認することをお勧めします。

- サイト管理者は、サイト内の選択したグループをオンデマンドで同期できます。詳細については、「サイトでの Active Directory グループの同期」を参照してください。
- サーバー管理者は、定期的またはオンデマンドでサーバー上のすべての Active Directory グループを同期させることができます。詳細については、「サーバー上のすべての Active Directory グループの同期」を参照してください。

注:

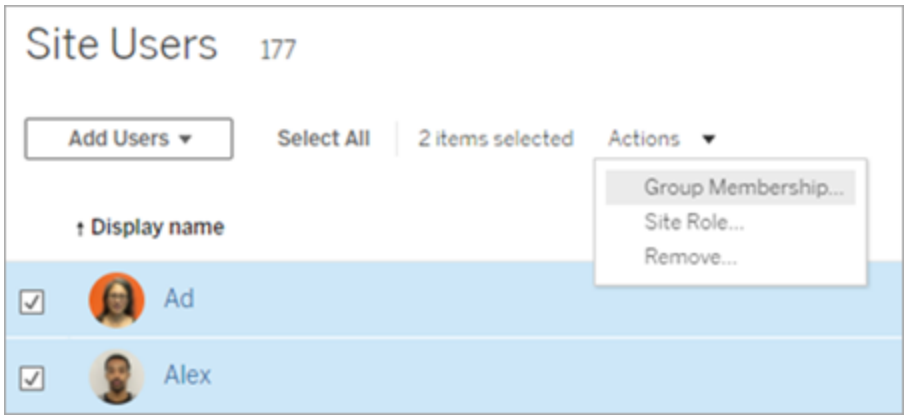
- ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアが構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。アイデンティティプールが構成されている場合、グループをアイデンティティプールに追加することはできません。
- グループをアイデンティティプールに追加することはサポートされていません。アイデンティティプールに追加できるのは、個々のユーザーのみです。

ローカル アイデンティティストアを使用しているユーザーを管理している場合は、以下に説明する手順を使用して、ユーザーをグループに追加します。

グループにユーザーを追加するには、グループが既に存在している必要があります。

ユーザーをグループに追加 (ユーザー ページ)

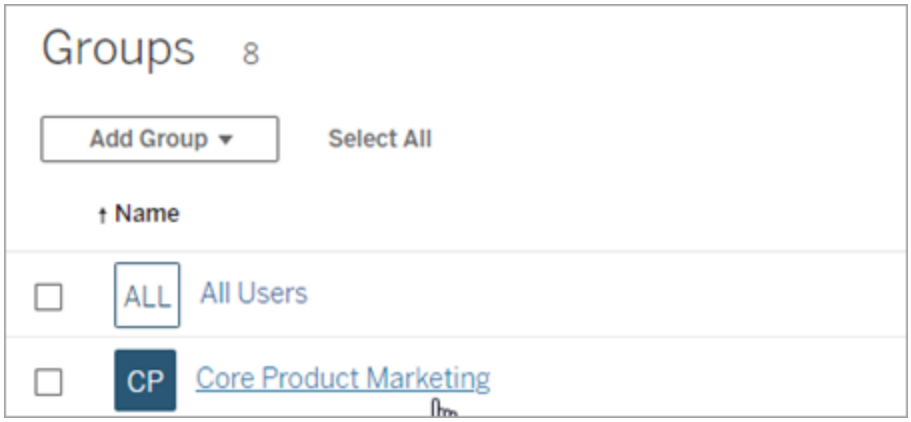
1. サイトで、**[ユーザー]** をクリックします。
2. グループに追加するユーザーを選択し、**[アクション] > [グループ メンバーシップ]** をクリックします。



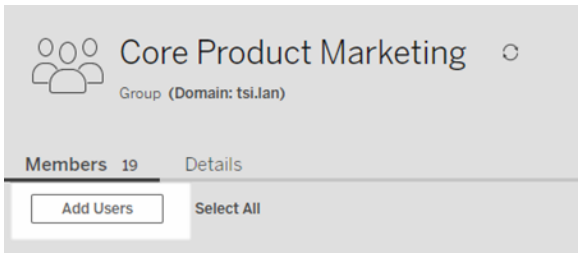
3. [グループ メンバーシップ] ダイアログ ボックスで、グループを選択し、**[保存]** をクリックします。

ユーザーをグループに追加 (グループ ページ)

1. サイトで、**[グループ]** をクリックし、グループの名前をクリックします。



2. グループ ページで、**[ユーザーの追加]** をクリックします。

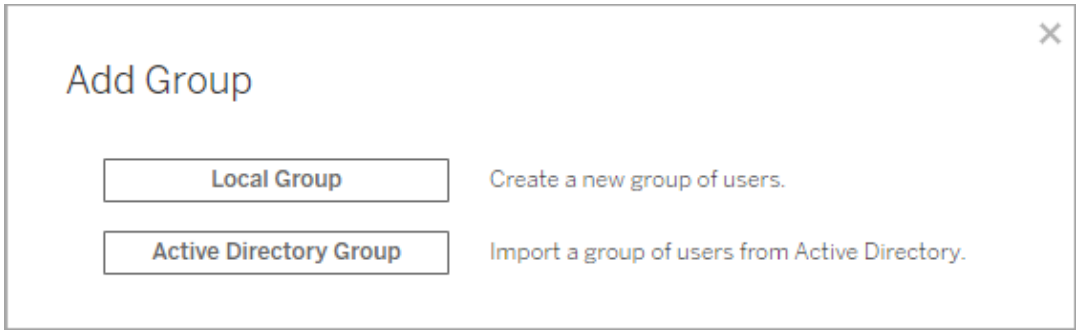


3. [ユーザーの追加] ダイアログ ボックスで、追加するユーザーを選択し、**[ユーザーの追加]** をクリックします。

ローカル グループの作成

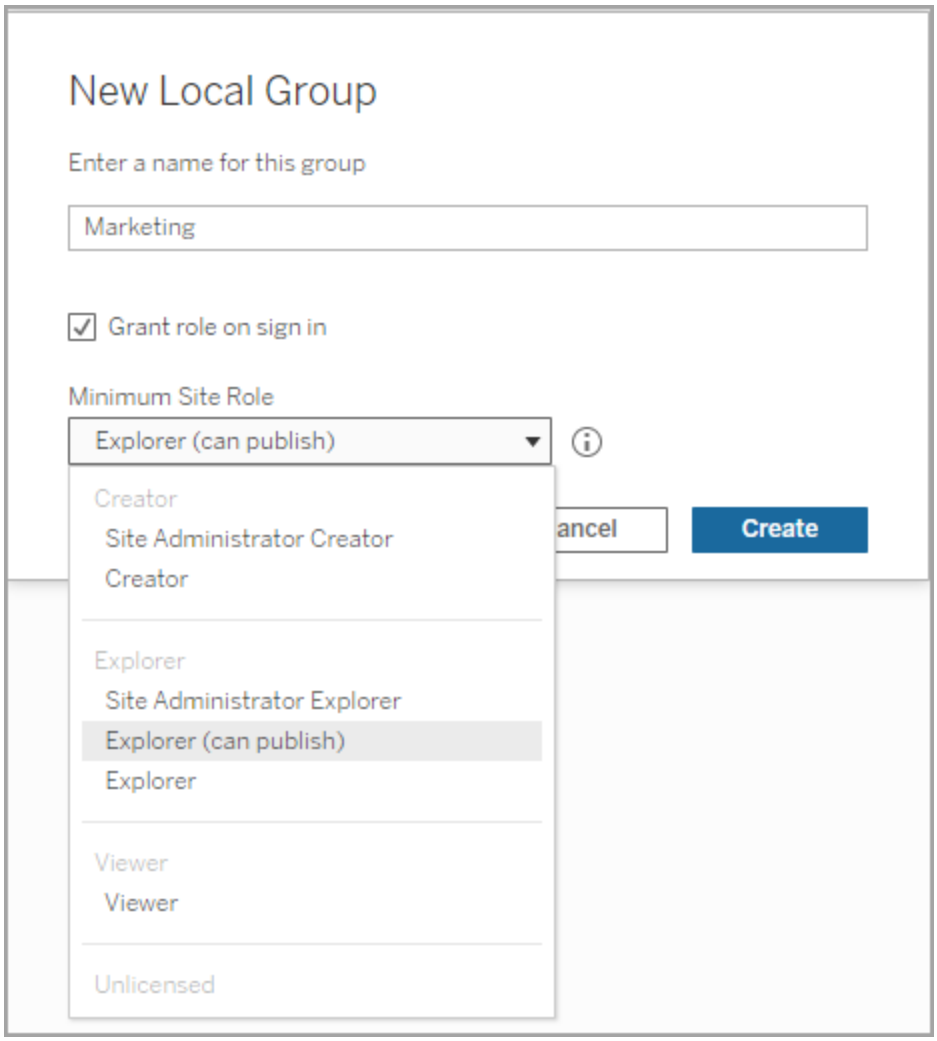
ローカル グループは、Tableau Server 内部ユーザー管理システムを使用して作成されたグループです。グループを作成したら、ユーザーの追加や削除、グループ内のユーザーがサインインするときに付与される最小限のサイト ロールの設定ができます。

1. サイトで **[グループ]** をクリックし、**[ローカル グループ]** をクリックします。



2. グループの名前を入力します。

3. グループの最小限のサイト ロールを設定するには、**[Grant site role on sign in (サインイン時にサイト ロールを付与)]** を選択し、ドロップダウン リストから最小限のサイト ロールを選択します。



4. **[作成]** をクリックします。

アサーションを使用した動的グループ メンバーシップ

Tableau Server 2024.2 以降、SAML 認証が設定されている場合、またはワークフローを埋め込むために Tableau 接続アプリを使用している場合は、アサーションを通じてグループ メンバーシップを動的に制御できます。設定すると、Tableau はユーザー認証中の実行時にアサーションを受信し、グループのメンバーシップを評価するとともに、それらのグループを基にパーミッションのあるコンテンツを評価します。

アサーションを通じてグループ メンバーシップを動的に制御するプロセスでは、1) 設定を有効にし、2) グループ メンバーシップの要求がアサーションに含まれていることを確認する必要があります。

ステップ 1: 設定を有効にする

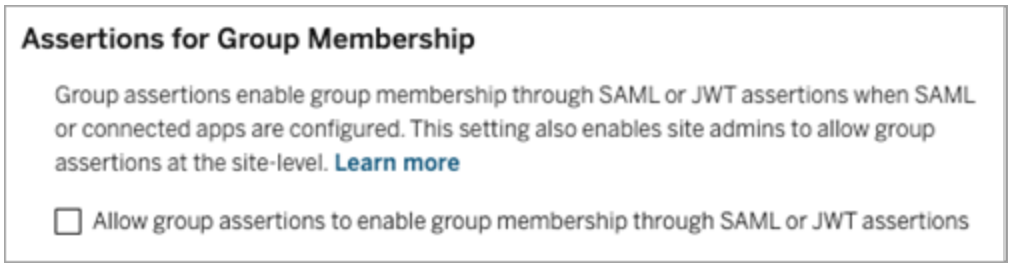
この機能には、サーバー全体の設定とサイトレベルの設定の 2 つの設定があります。サイトレベルの設定は、サーバー全体の設定が最初に有効になっている場合にのみ有効にできます。サイトの SAML または接続済みアプリが設定されている場合は、サイトレベルの設定を有効にすることを検討してください。

セキュリティ上の理由から、設定が有効になっている場合にのみ、グループ メンバーシップが認証ワークフローで検証されます。

1. Tableau サーバーにサインインし、**[設定]** ページに移動します。

   注: マルチサイト サーバーの場合は、すべてのサイトの **[設定]** ページに移動します。

2. [グループ メンバーシップのアサーション] ヘッダーの下で、**[グループ アサーションを許可して、SAML または JWT のアサーションを通じてグループ メンバーシップを有効にする]** チェック ボックスをオンにします。



3. (オプション) サイトレベルでサイト SAML または接続済みアプリが設定されている場合は、サイトに移動して [設定] ページに移動し、[グループ メンバーシップのアサーション] ヘッダーの下にある **[グループ アサーションを許可して、SAML または JWT のアサーションを通じてグループ メンバーシップを有効にする]** チェック ボックスをオンにします。

設定の詳細については、次のいずれかのトピックを参照してください。

- サーバーワイド- サーバー設定 (全般とカスタマイズ)
- サイトレベル - サイト設定 リファレンス

ステップ 2: グループ メンバーシップの要求がアサーションに含まれていることを確認する

グループ メンバーシップを指定するには、SAML、OIDC、または JWT のアサーションにそれぞれ 2 つのカスタム グループ メンバーシップの要求を含める必要があります。2 つのカスタム グループ メンバーシップの要求は次のとおりです。

- グループ: `https://tableau.com/groups`
- グループ名: これらの名前は、Tableau Server のローカル グループ名と正確に一致している必要があります。

アサーションの例については、次のいずれかのセクションを参照してください。

- SAML アサーションを使用した動的グループ メンバーシップ:
- 接続済みアプリ- 直接信頼: 動的グループ メンバーシップ (埋め込みワークフローのみ)
- 接続済みアプリ- OAuth 2.0 信頼: 動的グループ メンバーシップ (埋め込みワークフローのみ)

Active Directory 経由でのグループの作成

Active Directory (AD) のグループをインポートすると、AD と一致するグループを Tableau Server に作成したり、AD グループの各メンバーに対応する、Tableau Server にまだ存在しないユーザーを作成することができます。

注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

各ユーザーには、重要なプロセスの一部としてサイト ロールが割り当てられます。Tableau Server 内に既に存在しているユーザーをインポートする場合、インポートプロセスの間に割り当てられたサイト ロールは、ユーザーに対してサーバーへのアクセス権を追加で与えた場合にのみ適用されます。ユーザーをインポートしても、サイト ロールは設定されません。

はじめる前に

グループをインポートする前に、外部 ID ストアを使用した展開におけるユーザー管理を見直し、複数のドメイン、ドメイン命名、NetBIOS、および Active Directory ユーザー名の形式が Tableau ユーザー管理にどのように影響するかについて理解してください。

AD からインポートしてグループを追加

Tableau Server にユーザーグループを追加する一環として、Active Directory (AD) からグループをインポートします。グループをインポートするときは、最初にグループ名を入力してグループを検索します。

デフォルトでは、グループ名を入力すると、Tableau Server は AD (または LDAP) に対してワイルドカードクエリ(*marketing* など) を実行して、検索結果を最大にします。大規模な AD (または LDAP) ID ストアに対して操作すると、インポートするグループを正常に選択する前にタイムアウトが発生する可能性があります。

時間を節約し、インポート中にタイムアウトが発生する可能性を回避するには、次のいずれかの方法でグループ名を検索することを検討してください。

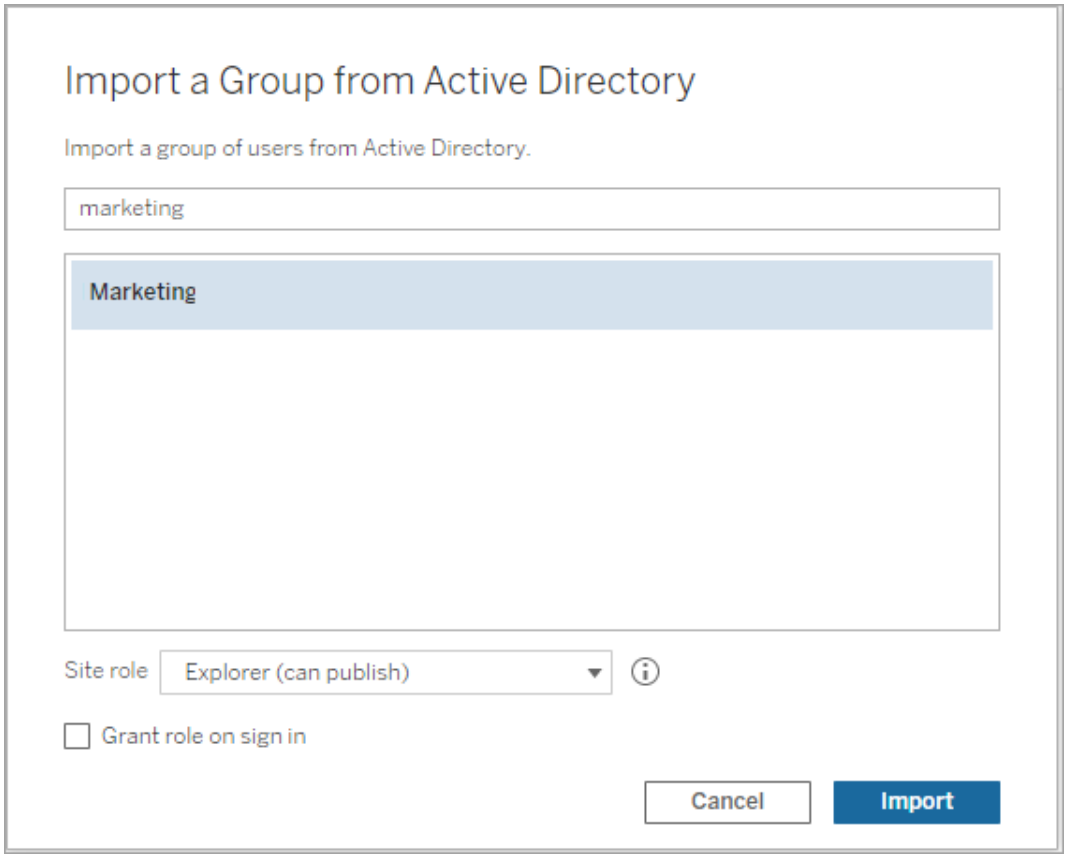
- **完全一致:** グループ名を検索するための最も効率的な方法です。入力した文字列の前後に引用符 (") を付けて、完全なグループ名を入力します。たとえば、**"marketing"** などです。
- **部分一致:** グループ名の一部を入力し、入力する文字列の前後にアスタリスク(*) を含めます。たとえば、***ket*** のように入力します。
- **前方一致:** グループ名の最初の部分に続いてアスタリスク(*) を入力します。たとえば、**market*** などです。
- **後方一致:** アスタリスク(*) の後に、グループ名の末尾部分を入力します。たとえば、***ing** などです。

注: これらの方法は、Tableau Server でユーザーを検索する際にも適用されます。

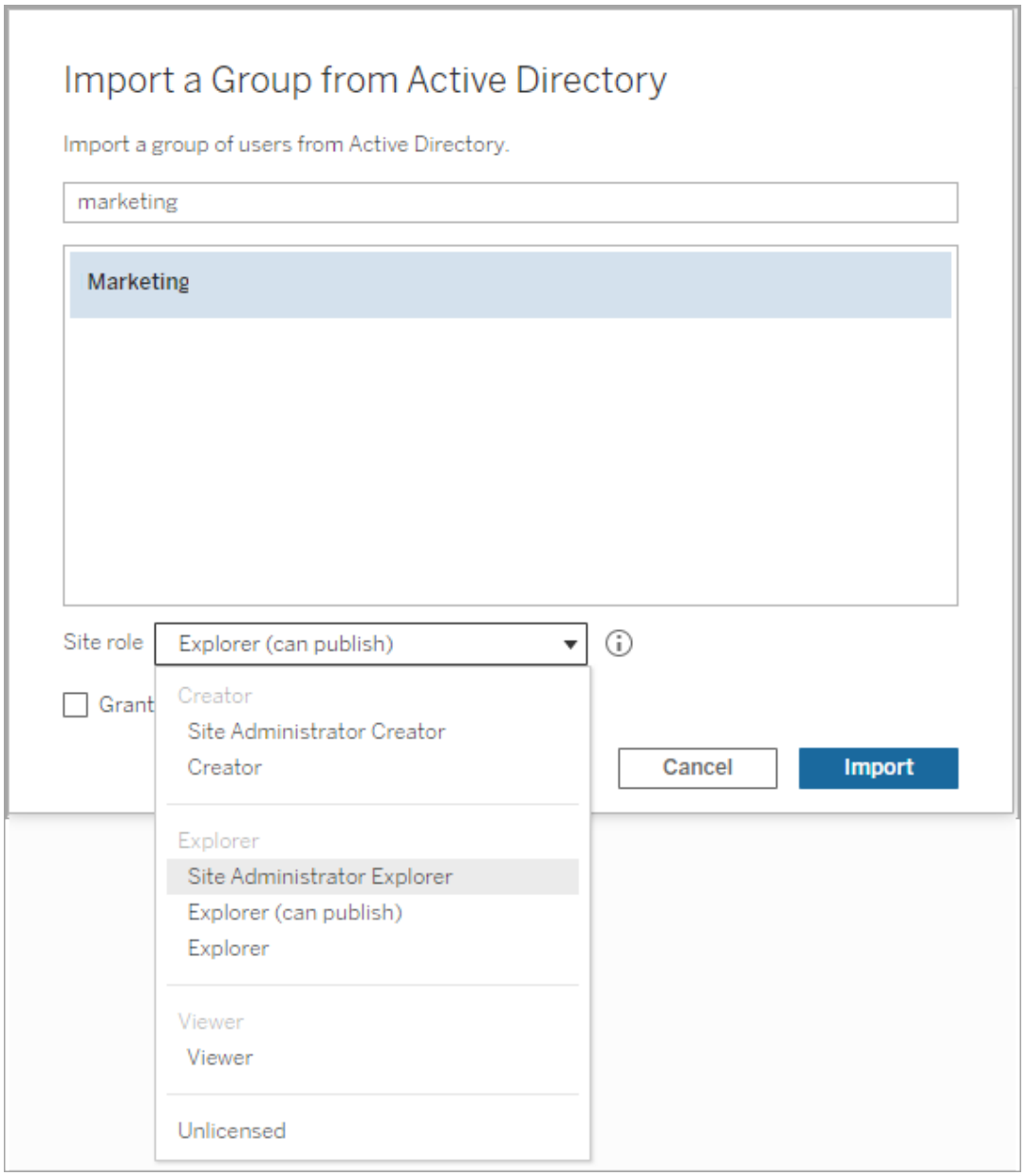
1. サイトで **[グループ]** をクリックし、**[グループの追加]** をクリックします。

2. インポートするActive Directory グループの名前を入力し、結果リスト内でグループ名を選択します。上記のフィルタリング方法のいずれかを使用して、パフォーマンスを向上させます。

   サーバーを実行している同じAD ドメインのグループをインポートする場合は、ドメインなしでADグループ名を入力できます。サーバーのドメインが使用されます。



3. ユーザーの最小のサイトロールを選択します。



4. (オプション) グループ ユーザーがサインインするときに新しいサイト ロールとライセンスをプロビジョニングするには、**[サインイン時にロールを付与]** を選択します。詳細については、サインイン時にライセンスを付与を参照してください。

5. **[インポート]** ボタンをクリックします。

注: Active Directory からインポートしたグループの名前は変更できません。グループ名は Active Directory でのみ変更できます。

サイト内の外部ディレクトリグループの同期

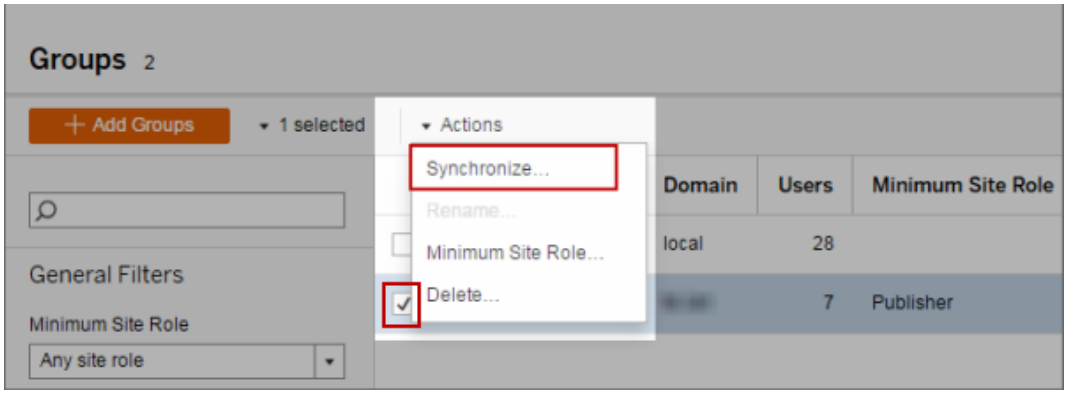
外部ディレクトリグループ (Active Directoryなど) は、いつでもTableau Server と同期でき、外部ディレクトリの新規ユーザーはTableau Serverにも追加されます。グループを 1 つずつ同期することも、複数のグループを一度に同期することもできます。

注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成されたTableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成されたLDAP ディレクトリソリューションとシームレスに連動します。

1. サイト内で **[グループ]** をクリックします。

   [グループ] ページで 1 つまたは複数のグループを選択します。

2. **[アクション] > [同期]** をクリックします。



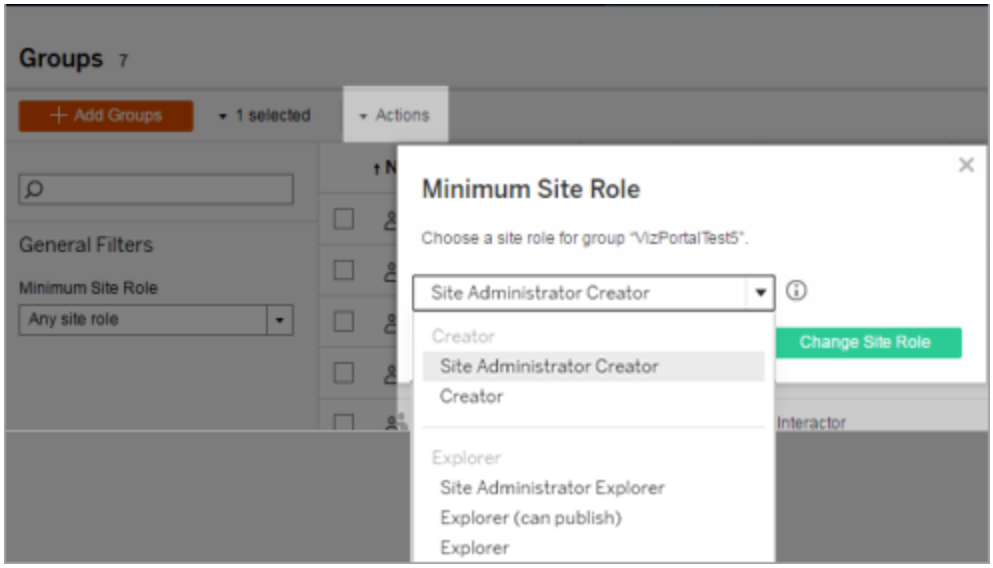
外部ディレクトリグループのユーザーに対して最小限のサイトロールを設定する

[グループ - 詳細] ページでは、管理者は同期中に適用されるグループユーザー向けの最小限のサイトロールを設定できます。

この設定では同期は実行されませんが、同期が実行されるたびにグループに適用される最小限のサイトロールが設定されます。外部ディレクトリグループを同期すると、最小限のサイトロールが割り当てられた新しいユーザーがサイトに追加されます。既にユーザーが存在する場合、ユーザーによるサーバーへのアクセスが増える場合にのみ最小限のサイトロールが適用されます。最小限のサイトロールを設定しない場合、新しいユーザーは既定で**ライセンスなし**として追加されます。

注: ユーザーのサイトロールのレベルが上がりますが、最小限のサイトロール設定に基づいてレベルが下がることはありません。ユーザーがパブリッシュする機能をすでに持っている場合、その機能は引き続き維持されます。最小限のサイトロールの詳細については、「サイトロールと Active Directory のインポートと同期」を参照してください。

1. サイト内で **[グループ]** をクリックします。
2. [グループ] ページで 1 つのグループを選択し、**[アクション] > [最小サイトロール]** の順に選択します。
3. 最小限のサイトロールを選択してから、**[サイトロールの変更]** をクリックします。



参照元の外部ディレクトリからユーザーが削除されるとどうなりますか。

外部ディレクトリの同期操作を通じて Tableau Server からユーザーを自動的に削除することはできません。外部ディレクトリグループからユーザーが無効または削除されても、Tableau Server では引き続き有効になっているため、管理者はユーザーのアカウントを完全に削除する前にユーザーのコンテンツを監視して再び割り当てることができます。詳細については、Active Directory からユーザーを削除する際の動作の同期を参照してください。

参照元の外部ディレクトリ内でユーザー名を変更するとどうなりますか?

既定では、Tableau Server は、対応するアカウントが Tableau Server で作成された時点の初期同期以降、ユーザー表示名やメール アドレスへの変更内容は同期しません。たとえば、表示名が John Smith でユーザー名として jsmith が使用されている場合、外部ディレクトリ内の表示名が Joe Smith に変更されても Tableau Server 内の対応する jsmith ユーザーには同期されません。同様に、ユーザーのメール アドレスが外部ディレクトリ内で変更された場合も、Tableau Server は変更を同期しません。

`vizportal.adsync.update_system_user` を `true` に設定すると、参照元の外部ディレクトリ内で名前と電子メールのプロパティが変更されたときに、名前と電子メールのプロパティを更新するように Tableau Server を構成できます。

この動作を変更するには、以下の tsm コマンドを実行します。

```
tsm configuration set -k vizportal.adsync.update_system_user -v
true

tsm pending-changes apply
```

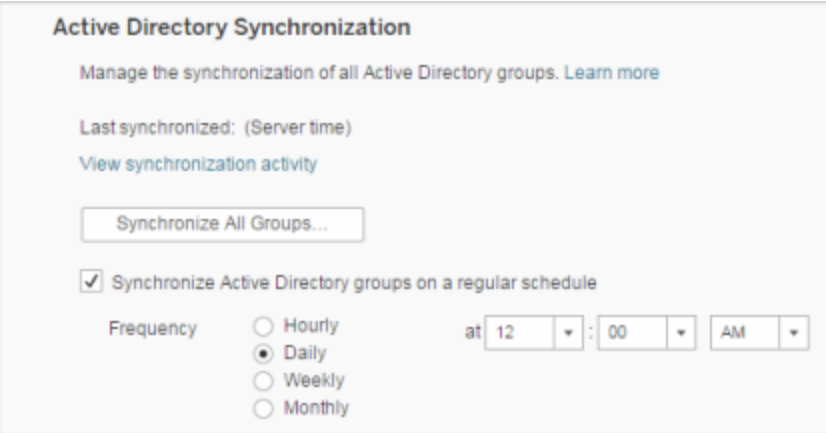
保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

Tableau Server から外部ディレクトリグループが削除されるとどうなりますか?

多くの Tableau 管理者が外部ディレクトリグループを使用してユーザーをインポートおよび作成しています。Tableau Server にユーザーをインポートした後、管理者は Tableau Server 内のグループを削除できます。グループを削除しても、グループ内のユーザーは削除されません。

サーバー上のすべての外部ディレクトリグループの同期

サーバー管理者は、サーバーの **[設定]** ページの **[全般]** タブで、(Tableau Server で設定済みの) すべての外部ディレクトリグループ (Active Directory など) をスケジュールに基づいて、またはオンデマンドで同期できます。



注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

はじめる前に

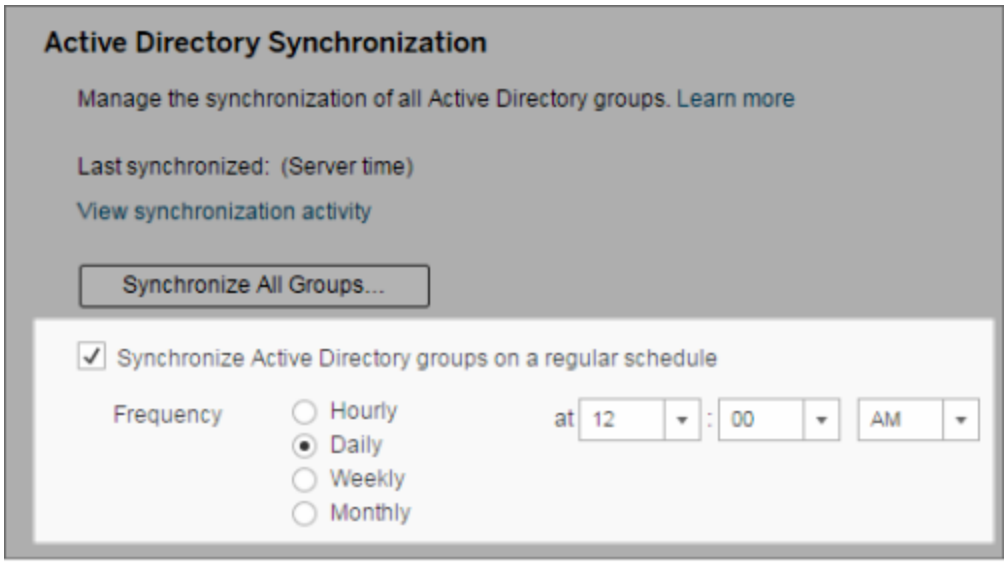
このトピックの説明に従ってグループを同期する前に、まず外部ディレクトリグループを Tableau Server にインポートする必要があります。Active Directory 経由でのグループの作成を参照してください。

外部ディレクトリグループのスケジュールに基づく同期

1. **単一のサイト**: **[設定] > [全般]** をクリックします。

   マルチサイト: サイト メニューで、**[すべてのサイトを管理]**、**[設定] > [全般]** の順にクリックします。

2. ページを下にスクロールして **[Active Directory の同期]** に移動し、**[スケジュール上の Active Directory グループの同期]** を選択します。



3. 同期の頻度と時間を選択します。

4. **[保存]** をクリックします。

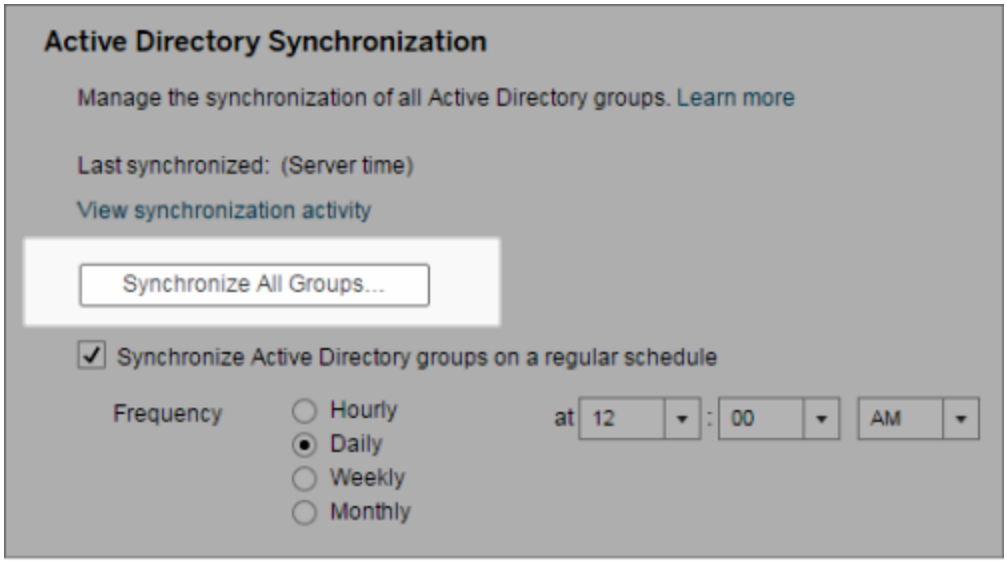
注: バージョン 2021.1.23、2021.2.21、2021.3.20、2021.4.15、2022.1.11、2022.3.3、2023.1 以降では、スケジュールされたグループ同期がキャンセルされるまでの待機時間が、既定の時間制限 (4 時間) になります。スケジュールされた同期が非常に大きなグループの同期である場合、または既定よりも長い時間がかかっている場合は、サーバー管理者はこの制限時間を変更することができます。詳細については、「すべての Active Directory グループをスケジュールに従って同期する」および「backgrounder.timeout.sync_ad_group」を参照してください。

外部ディレクトリグループのオンデマンドでの同期

外部ディレクトリグループ (Active Directory など) は、いつでもTableau Serverと同期でき、外部ディレクトリの新規ユーザーと変更はTableau Server上のすべての外部ディレクトリグループに反映されます。

1. **単一のサイト:** **[設定] > [全般]** をクリックします。

   マルチサイト: サイト メニューで、**[すべてのサイトを管理]**、**[設定] > [全般]** の順にクリックします。



2. **[Active Directory の同期]** で、**[すべてのグループの同期]** をクリックします。

同期アクティビティの表示

[抽出以外のバックグラウンド タスク] 管理ビューに同期ジョブの結果を表示できます。**[Active Directory グループ同期のキューイング]** とは、キューイングを行うタスクで、実行される **[Active Directory グループの同期]** タスクの数を示します。

1. **単一のサイト:** **[状態]** をクリックします。

   マルチサイト: サイト メニューで、**[すべてのサイトを管理]**、**[ステータス]** の順にクリックします。

2. **[抽出以外のバックグラウンド タスク]** リンクをクリックします。

3. **[タスク]** フィルターに **[Active Directory グループ同期のキューイング]** および **[Active Directory グループの同期]** を設定します。

サーバーの **[設定]** ページの **[同期アクティビティの表示]** リンクをクリックすると、この管理ビューに素早く移動できます。

外部ディレクトリグループのユーザーに対して最小限のサイトロールを設定する

[グループ - 詳細] ページでは、Active Directory の同期中に適用されるグループユーザー向けの最小限のサイトロールを設定できます。

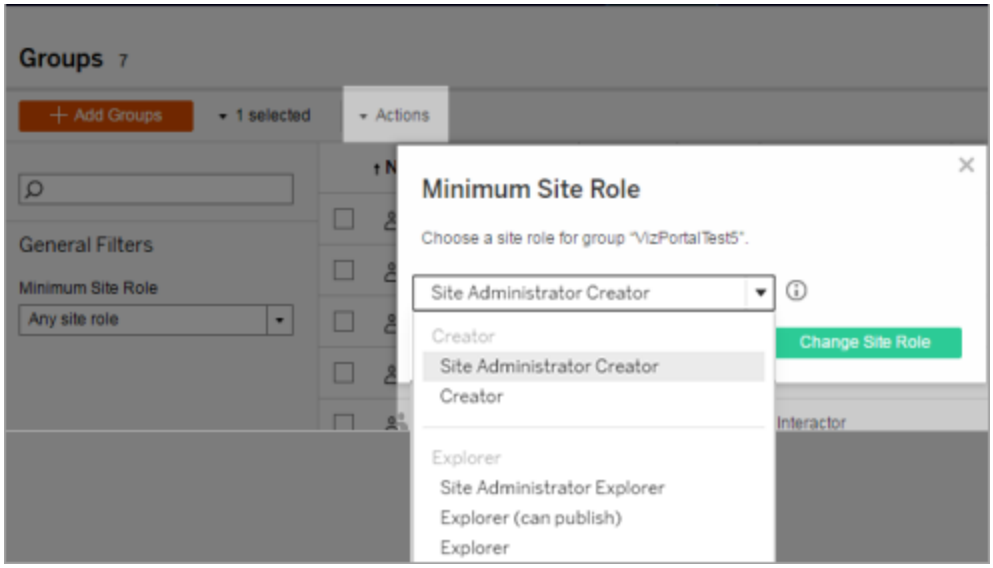
この設定では同期は実行されません。代わりに、同期が実行されるたびにグループに適用する最小限のサイトロールを設定します。その結果、外部ディレクトリグループを同期すると、最小限のサイトロールが割り当てられた新しいユーザーがサイトに追加されます。既にユーザーが存在する場合、ユーザーにとってサイト内でのアクセス範囲が増える場合にのみ最小限のサイトロールが適用されます。最小限のサイトロールを設定しない場合、新しいユーザーは既定で**ライセンスなし**として追加されます。

注: ユーザーのサイトロールのレベルが上がりますが、最小限のサイトロール設定に基づいてレベルが下がることはありません。ユーザーがパブリッシュする機能をすでに持っている場合、その機能は引き続き維持されます。最小限のサイトロールの詳細については、「サイトロールと Active Directory のインポートと同期」を参照してください。

1. サイト内で **[グループ]** をクリックします。
2. [グループ] ページでグループを選択します。

   [アクション] > [最小限のサイトロール] をクリックします。

3. 最小限のサイトロールを選択してから、**[サイトロールの変更]** をクリックします。



参照元の外部ディレクトリからユーザーが削除されるとどうなりますか。

外部ディレクトリの同期操作を通じて Tableau Server からユーザーを自動的に削除することはできません。外部ディレクトリグループからユーザーが無効または削除されても、Tableau Server では引き続き有効になっているため、管理者はユーザーのアカウントを完全に削除する前にユーザーのコンテンツを監視して再び割り当てることができます。詳細については、Active Directory からユーザーを削除する際の動作の同期を参照してください。

グループ同期のパフォーマンスの向上

外部ディレクトリの同期はバックグラウンダープロセスによって実行されます。このバックグラウンダープロセスは抽出の管理や作成に使用されるものと同じプロセスであり、サブスクリプション コンテンツの生成にも使用されます。動的なグループ メンバーシップと大量の抽出が使用される大規模な組織では、外部ディレクトリのグループ同期プロセスによって業務に支障が出る可能性があります。業務時間外にグループ同期を実行することをお勧めします。

既定では、バックグラウンダープロセスは逐次操作で同期を実行します。つまり、各グループが 1 つのバックグラウンダープロセスで 1 つずつ同期されます。単一の Tableau Server や分散展開でバックグラウンダープロセスの複数のインスタンスを実行している場合は、外部ディレクトリの同期で並列処理を有効にすることを検討してください。並列バックグラウンダー処理が有効になっていると、パ

フォーマンスを向上させるためにグループ同期が複数のバックグラウンダープロセスに分散されます。

グループ同期で並列バックグラウンダー処理を有効にするには、TSM CLI を開いて次のコマンドを入力します。

```
tsm configuration set -k backgrounder.enable_parallel_adsync -v
true

tsm pending-changes apply
```

すべての Active Directory グループをスケジュールに従って同期する

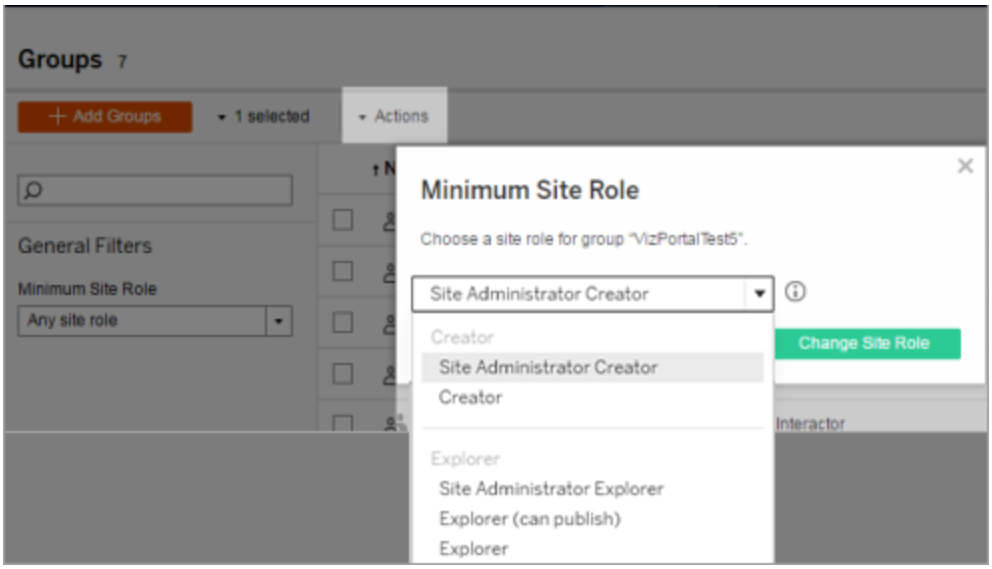
Tableau Server で Active Directory グループをインポートした後、スケジュールを設定することで、Tableau Server 内での同期を保つことができます。また、いつでもオンデマンドでサーバー上のすべての Active Directory グループを同期することもできます。ユーザーが同期されている間、グループに対する最小限のサイトロール設定が適用されます。

注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

> バージョン 2021.1.23、2021.2.21、2021.3.20、2021.4.15、2022.1.11、2022.3.3、2023.1 以降では、スケジュールされたグループ同期がキャンセルされるまでの最大時間制限を設定することができるようになりました。既定の制限時間は 4 時間です。詳細については、「backgrounder.timeout.sync_ad_group」を参照してください。

1 同期に対する最小限のサイトロールを設定する

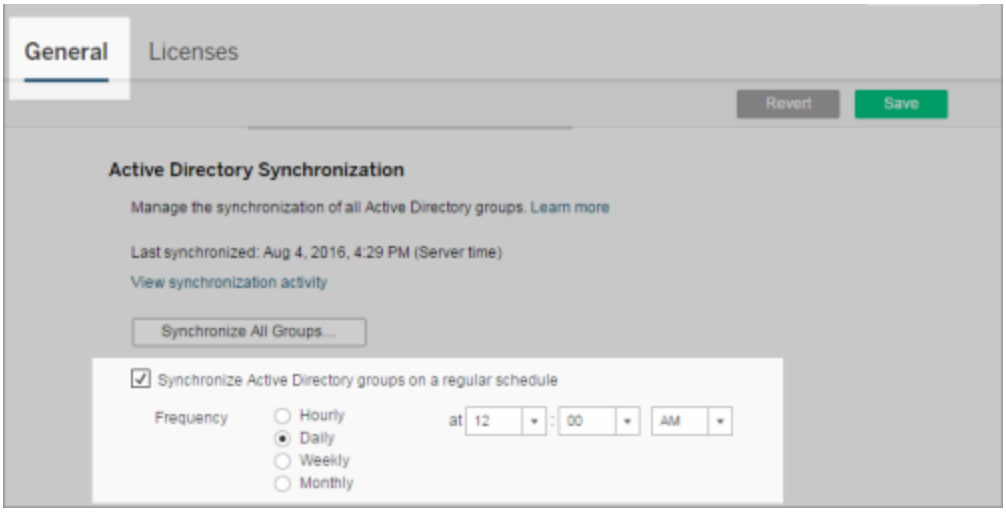
サイト内で **[グループ]** をクリックします。グループを選択してから、**[アクション] > [最小限のサイトロール]** をクリックします。最小限のサイトロールを選択してから、**[サイトロールの変更]** をクリックします。サーバーとサイト管理者はグループユーザーに対し、Active Directory 同期の間に適用される最小限のサイトロールを設定できます。最小限のサイトロールを設定しない場合、新しいユーザーは**ライセンスなし**として追加されます。



同期すると、ユーザーのサイトロールのレベルが上がりますが、ユーザーのサイトロールのレベルが下がることはありません。

2 スケジュールの設定

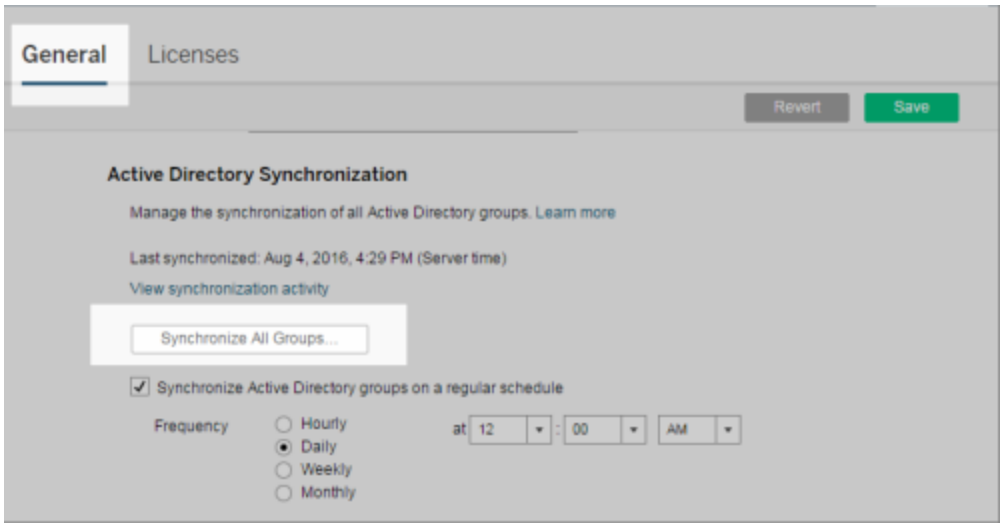
サーバー管理者は、サーバーの **[設定]** ページの **[全般]** タブ上ですべての Active Directory グループに対して同期を有効化できます。同期を有効にし、頻度設定を選択してから、**[保存]** をクリックします。



サーバー上のすべての Active Directory グループは同じスケジュールにしたがって同期されます。

3 オンデマンドでの同期の実行 (オプション)

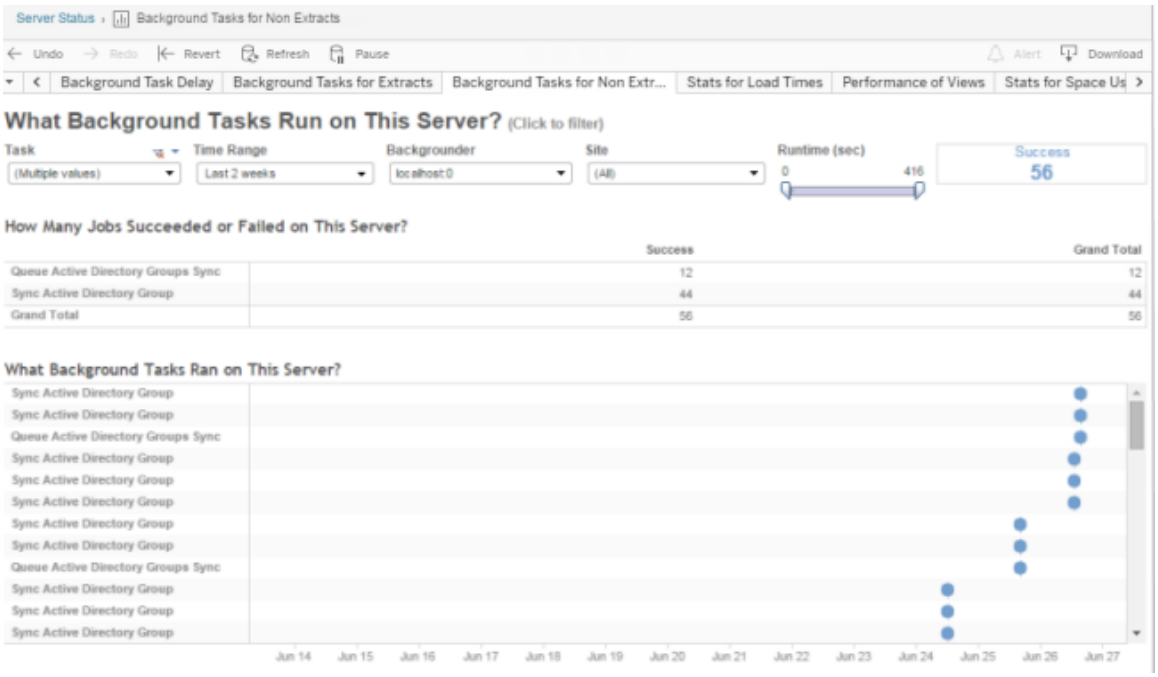
[設定] ページの **[全般]** タブで、**[すべてのグループを同期]** をクリックして Tableau Server 上のすべての Active Directory グループを直ちに同期します。新しいユーザーと変更がサーバー上のすべての Active Directory グループに適用されていることを確認するには、いつでもこのボタンをクリックします。



スケジュール外のサーバー上ですべての Active Directory グループを同期するには、**[すべてのグループを同期]** をクリックします。

4 同期タスクのステータスを表示する

サーバーとサイト管理者は、**[抽出以外のバックグラウンド タスク]** 管理ビューの Active Directory 同期ジョブの結果を表示できます。サーバー上またはサイト内で、**[ステータス]** をクリックします。**[分析]** の下で、**[Active Directory グループ同期のキュー]** および **[Active Directory グループの同期]** タスクで、**[抽出でのバックグラウンド タスク]** およびフィルターをクリックします。



[Active Directory グループ同期のキュー] は実行する **[Active Directory グループの同期]** タスクをキューに追加します。

サインイン時にライセンスを付与

サインイン時にライセンスを付与 (サインイン時にロールを付与) すると、ライセンスのない特定グループのユーザーがライセンスを取得して Tableau サイトにサインインすることができます。これにより、管理者がライセンスを付与する作業が合理化され、ユーザーは Tableau を使用するためにライセンスを申請する必要がなくなります。

サイトロール機能と最小サイトロールの詳細については、ユーザーのサイトロールの設定を参照してください。

たとえば、100 人のユーザーがいる Marketing という Active Directory のグループがあり、そのうち 25 人のユーザーだけが Tableau Server にアクセスする必要がある場合を考えます。サイト管理者またはサーバー管理者は、Active Directory グループ Marketing のすべてのユーザーをインポートし、Explorer に対する最小サイトロールをこのグループに設定して、**[サインイン時にロールを付与]** を選択します。Marketing にいる Tableau ユーザーの誰かが Tableau サイトにサインインすると、

Explorer ライセンスが付与されます。Tableau Server を必要としないユーザーは、サインインしない限りライセンスを取得しません。

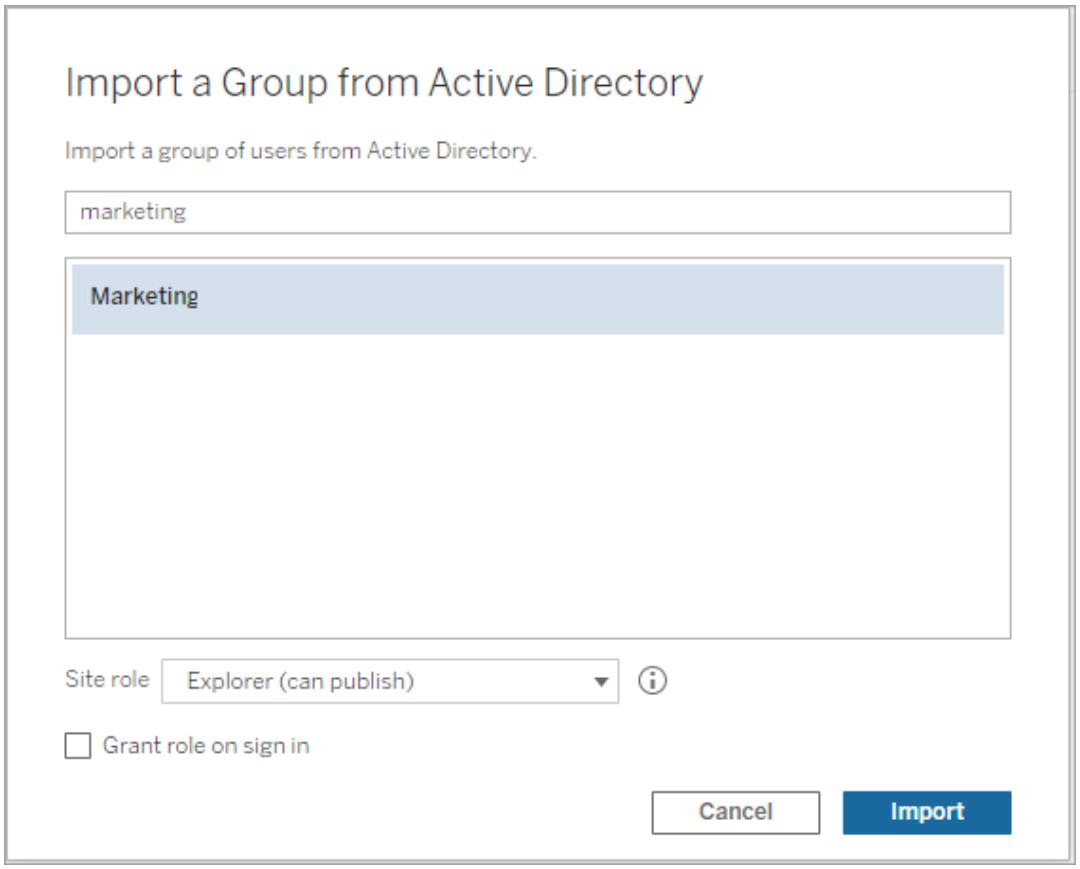
> **注**: 利点とベストプラクティスの詳細については、Tableau のデータ主導型組織向けの計画ツールである Tableau Blueprint にある「サインイン時にロールを付与」を参照してください。

サインイン時にロールを付与

新規または既存のグループで [サインイン時にロールを付与] を有効にできます。以下の手順では、サインイン時にロールを付与する方法について説明します。ライセンスを使用する資格のある対象者ではあるものの、使用しない可能性のある新しいユーザーを追加することができます。会社に多数の対象ユーザーがいるものの、Tableau ライセンスの数が限られている場合が例として考えられます。

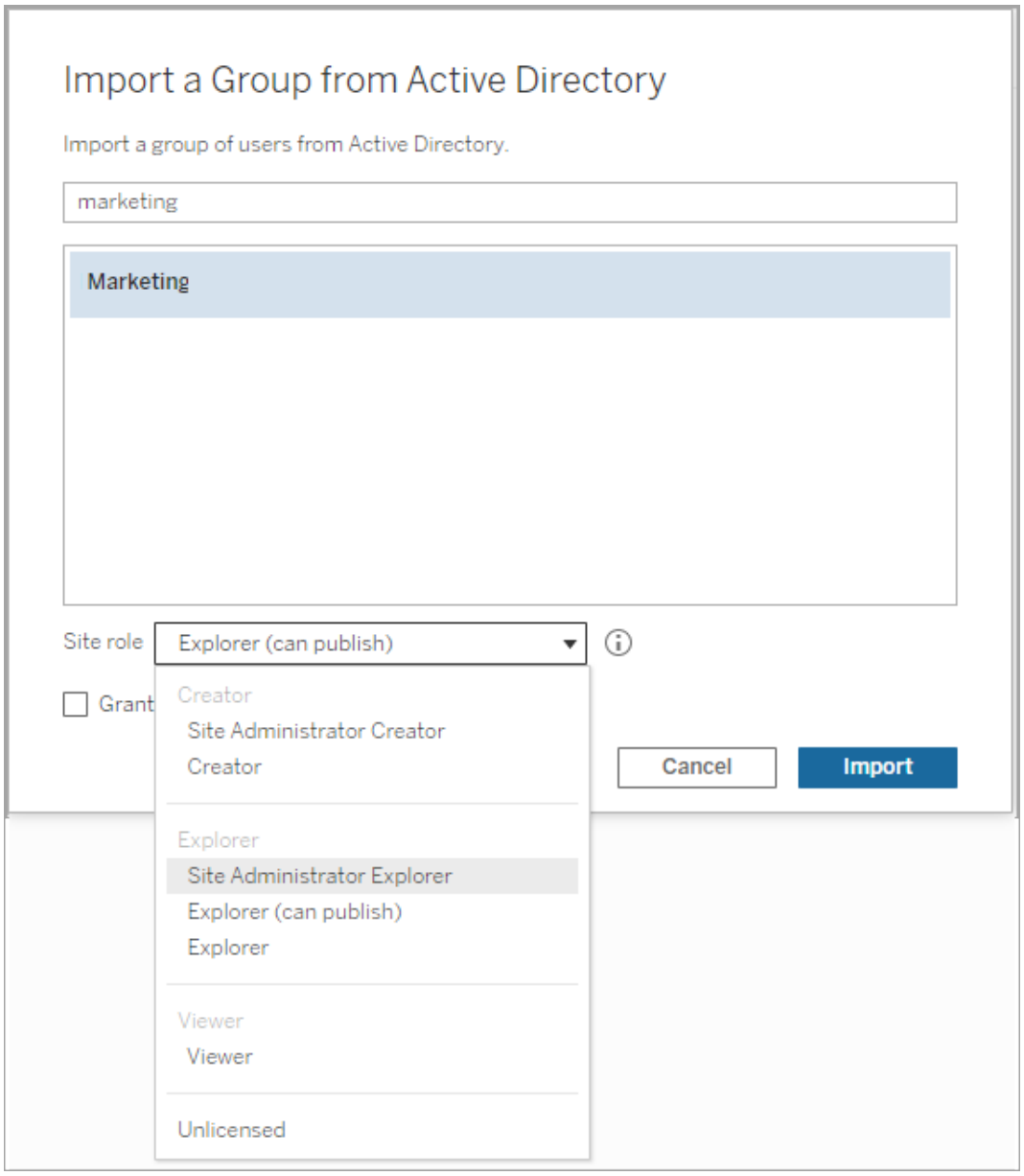
1. サイトで **[グループ]** をクリックしてから、**[グループの追加]** をクリックします。

   Active Directory のグループをインポートして新しいユーザーを追加します。インポートするグループの名前を入力し、表示されるリスト内でそのグループ名を選択します。



2. ユーザーの最小サイトロールを選択し、**[サインイン時にロールを付与]** を選択します。

   選択した Active Directory グループのすべてのユーザーが、ライセンスのないユーザーとしてインポートされます。グループに設定した最小サイトロールは、Tableau Server にサインインするグループユーザーにのみ付与されます。



3. **[インポート]** をクリックします。

注: [サインイン時にロールを付与] をローカル グループで有効化することもできます。グループのメンバーが Tableau Server にサインインするときに、最小のサイト ロールを付与できるようになります。詳細については、ローカル グループの作成を参照してください。

"サインイン時にロールを付与" が有効になっているユーザー ロールを変更する

"サインイン時にロールを付与" を使用しているグループにユーザーが含まれる場合、そのユーザー ロールは、サインインの有無にかかわらず、ライセンスなしに設定したり、そのグループに設定された最小サイト ロールよりも低いロールにダウングレードしたりできません。ただし、管理者はユーザーのサイト ロールを手動でアップグレードできます。

ユーザーのサイト ロールをダウングレードするか、サイトからユーザーをライセンス解除するには、"サインイン時にロールを付与" が有効になっているグループからユーザーを削除します。

エンド ユーザー ライセンス契約の条項に従って、承認ユーザー ベースで付与されたライセンスは、新しいユーザーに恒久的に再割り当てすることができます。ユーザーは、より高いロールでサーバー ソフトウェアにアクセスすることを恒久的に停止する場合にのみ、下位のサイト ロール (ライセンスなしを含む) にダウングレードできます。

"サインイン時にロールを付与" の影響を受けるユーザーを削除する

ユーザーがコンテンツを所有していない場合のみ、サイトからユーザーを削除できます。コンテンツを所有しているユーザーを削除しようとすると、ユーザーのサイト ロールはライセンスなしに設定され、すべてのグループから削除されますが、ユーザーはサイトから削除されません。コンテンツの所有者を削除するには、"サイト ロールの付与" が有効になっているグループから所有者を削除するか、コンテンツ所有権を別のユーザーに再割り当てします。詳細については、ユーザーの表示、管理、削除ヘルプ トピックの「サイトからユーザーを削除」を参照してください。

既定の "すべてのユーザー" グループで "サイト ロールの付与" が有効になっている場合、コンテンツを所有しているユーザーをサイトから削除したり、ライセンスなしに設定したりできません。これらのユーザーを削除またはライセンス解除するには、コンテンツ所有権を別のユーザーに再割り当てしてから、ユーザーを削除またはライセンス解除します。

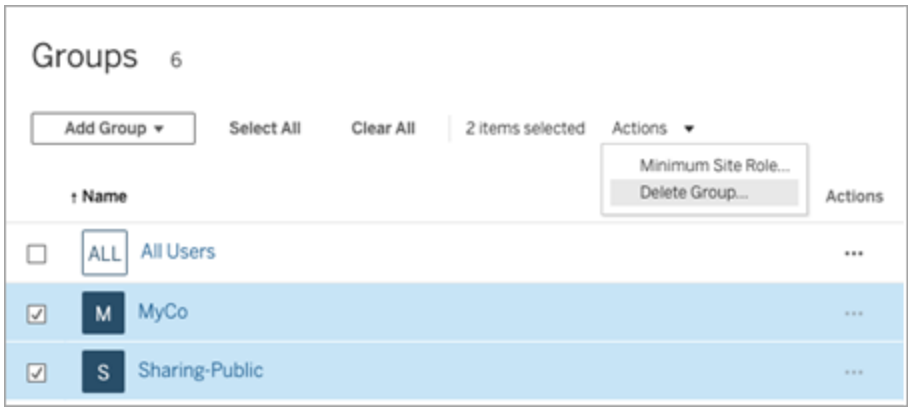
Tableau REST API を使用して、ワークブックのコンテンツ所有権を再度割り当てることができます。詳細については、REST API ヘルプの「ワークブックを更新する」を参照してください。REST API を使用して、サイトからユーザーを削除し、コンテンツ所有権を別のユーザーに転送することもできます。詳細については、REST API ヘルプの「サイトからユーザーを削除する」を参照してください。

Tableau Server でのコンテンツ所有権の変更に関する詳細については、コンテンツ所有権の管理を参照してください。

グループの削除

[すべてのユーザー] グループを除く任意のグループをサーバーから削除できます。グループを削除すると、ユーザーはグループから削除されますが、サーバーからは削除されません。

1. 管理者としてTableau Server サイトにサインインします。
2. 左側のナビゲーション ペインで、**[グループ]** をクリックします。
3. [グループ] ページで、削除する1 つまたは複数のグループを選択します。
4. **[アクション] > [削除]** を選択します。



グループを削除する影響

グループ セット内のグループ

Tableau Server 2024.2 以降、グループ セットにグループを追加できます。コンテンツのパーミッションがグループ セットに基づいている場合、ユーザーがグループ セット内のすべてのグループに属しているときにコンテンツの機能が評価されます。コンテンツのパーミッションがグループ セットに基づいている場合、グループ セットに属するグループが削除されると、Tableau コンテンツへのユーザーのアクセス権が変更される可能性があります。

グループ セットの操作

Tableau Server 2024.2 以降、グループ セットを使用してグループのコンテナを作成できます。グループ セットには 1 つまたは複数のグループを含めることができ、グループ セットを基にしたコンテン

ツのパーミッションに、より詳細なルールを適用できます。グループセットを基にして機能を有効にする場合、グループセットに属するグループのユーザーが、機能が評価されるすべてのグループのメンバーである必要があります。このように、グループセットは AND ロジックを適用します。

グループセットの利点:

- 同期されたグループとローカルグループをパーミッションルールで組み合わせて、より動的なアクセス制御のシナリオを実現できます。
- パーミッションルールのグループに AND ロジックを使用すると、シナリオによってはアクセス制御を簡素化できます

注:

- グループセットのパーミッションルールは、ユーザールールとグループルールの後に評価されます。これらのルールの詳細については、「パーミッションルールの評価」を参照してください。
- グループセットは、サーバー管理者のみが作成できます。
- グループセットをグループのグループとして使用しないでください。

グループセットをオンにする

グループセットをパーミッションに使用するには、グループセット設定を有効にする必要があります。

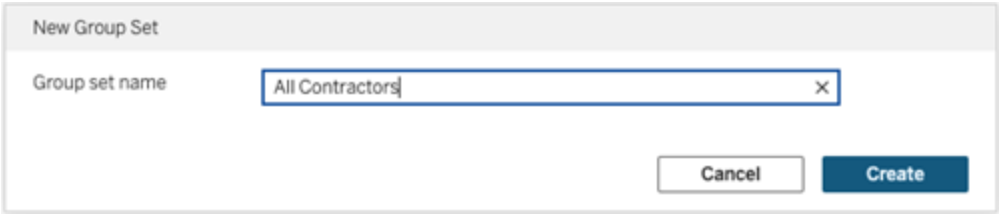
1. Tableau Server にサーバー管理者としてサインインします。
2. **[設定]** ページへ移動します。
3. [グループセット] セクションで、**[グループセットを許可する]** チェックボックスをオンにします。

グループセットを有効にすると、ナビゲーションペインに専用の **[グループセット]** ページが表示されます。

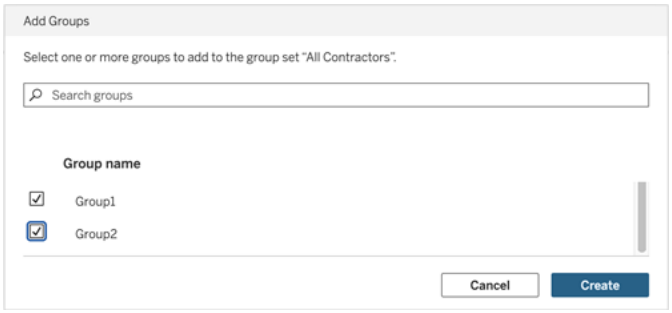
グループ セットを作成する

グループ セットを作成するには、[グループ セット] ページに移動し、グループの場合と同じようにグループ セットを作成します。

1. Tableau Server にサーバー管理者としてサインインします。
2. [グループ セット] ページに移動し、**[新しいグループ セット]** ボタンをクリックします。
3. グループ セット名を入力し、**[作成]** をクリックします。



4. [グループ セット] テーブルで、作成したグループ セット名をクリックし、**[グループの追加]** ボタンをクリックします。
5. 利用可能なグループのリストから、グループ セットに追加するグループを選択し、**[追加]** ボタンをクリックします。



グループ セットのパーミッションを設定する

サイト管理者、プロジェクト リーダー、またはコンテンツ所有者がグループ セットを使用するには、グループ セットを使用するコンテンツのパーミッションを追加または編集します。

たとえば、あなたが「打者」ワークブックの所有者であるとします。グループ セットを基にしたパーミッションを適用するには、次の手順を実行します。

1. ワークブックに移動し、アクション メニューから**[パーミッション]** を選択します。
2. [パーミッション] ダイアログ ボックスで、**[グループ/ユーザー ルールの追加]** ボタンをクリックし、次の操作を行います。
   a. テキスト ボックスに、「すべての契約者」などのグループ セット名を入力します。
   b. 必要な機能をテンプレートで選択します。
   c. **[保存]** をクリックします。

グループ セット モデルを使用してパーミッションを適用する場合は、グループ セット内のすべてのグループのメンバーシップを持つユーザーに影響を及ぼすパーミッションを作成します。

たとえば、ユーザーが所属する地域グループに基づいて、異なるワークブック ビューへのアクセスを制限することができます。これらのワークブック ビューにアクセスする、契約社員、契約社員 マネージャー、フルタイム社員、フルタイム マネージャーのユーザーがいるとします。

ワークブックを表示するための既定のパーミッションを持つグループ: すべて、北部、南部、東部、西部、フルタイム、およびマネージャー。

- 北部地域の詳細ビュー:
  - パーミッションはグループ セットに基づいています: 北部地域
  - グループ セット内のグループ: フルタイム、北部

  結果: [北部地域の詳細] ビューでデータを表示できるのは、北部地域のフルタイム社員とフルタイム マネージャーのみです。

- 全地域の詳細ビューの場合:
  - グループ セット名はマネージャーです
  - グループ セット内のグループ: すべて、マネージャー

  結果: [全地域の詳細] ビューでデータを表示できるのは、契約社員 マネージャーとフルタイム マネージャーのみです。

パーミッションの詳細については、マネージドセルフサービスのためのプロジェクト、グループ、グループセット、パーミッションの設定を参照してください。

ダッシュボードベースのカスタム ポータル

注: この概要は、Tableau のビジョナリである Mark Jackson の作品の影響を受けています。プロセスの詳細については、マークのブログをチェックしてください。

Tableau Cloud または Tableau Server の標準のインターフェイスは多くの組織に適していますが、会社ブランドの体験を作成したいが API 開発者スキルを持っていないという場合は、Tableau ダッシュボードに基づいたカスタム ポータルの作成を検討してください。カスタム ポータルを使用することで特定の部門やジョブのロールに関するコンテンツを整理したり、データをより完全に操作するのに役立つ研修を組み込むこともできます。Tableau コンテンツの量が増えるにつれ、カスタム ポータルがユーザーを必要とするデータへ直接導き、簡単に詳しく探索できるよう、関連するビューや Tableau Server 検索へのアクセスを提供します。

ポータル デザインのスケッチを描く

Tableau の外で、紙やワイヤーフレーム作成アプリケーションを使用して始めます。組織の構造と、各ユーザーのグループに適用される Tableau ビューの数を考慮してください。コンテンツにすぐにたどり着く、1 レベルのナビゲーションを単純に必要としていますか? または、別のユーザーやレポートタイプのグループへのナビゲーション リンクを提供するメイン ページから開始し、次に、関連ダッシュボードやビューが入った 2 番目のレベルへと進める必要がありますか?

重要な利害関係者やデータコンシューマーからデザインについて同意が得られたら、次の段階に進むことができます。

ロゴおよびナビゲーション エレメント用の画像を収集する

最初のモックアップを微調整しながらイメージについて考え始め、次に、それらをクリップアートのライブラリや承認済みブランドグラフィックから収集するか、Photoshop or SnagIt などのアプリケーションでゼロから作成します。一般的なグラフタイプのナビゲーション サムネイルを作成する予定がある場合は、PowerPoint も優れたソースです。

ダッシュボードのレイアウト テキスト、イメージ、および選択されたシート

異なるデバイスでのエレメントの予測可能な配置や拡大縮小を行うため、タイル レイアウトを使用して、ポータル用にダッシュボードを含むワークブックを作成します。次に、テキストおよび画像オブジェクト、ポータルの最前線でハイライトさせるデータビューシートの追加を開始します。これらのダッシュボード エレメントの間の隙間を調節するため、空白オブジェクトを挿入してサイズを調節します。詳細については、「ダッシュボードの作成」を参照してください。

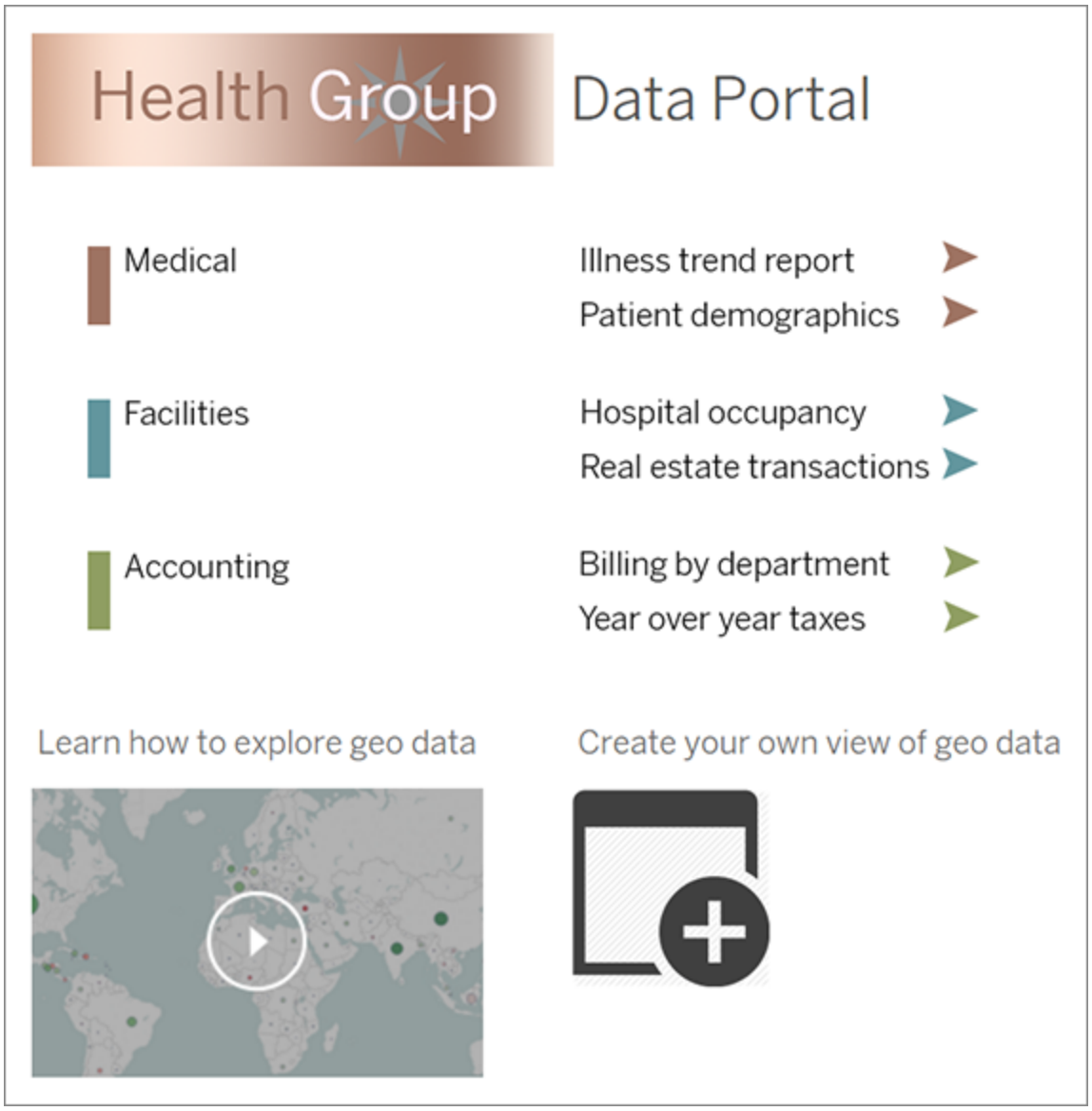
> **ヒント:** 人気のあるコンテンツや最近作成されたコンテンツを示す動的なエレメントについては、カスタム管理ビューに基づいてシートを含めます。

ダッシュボード エレメントをコンテンツにリンクさせる

ポータルに 2 番目のナビゲーション レベルがある場合は、フィルター アクションを使用して、メインのダッシュボードからセカンダリダッシュボードを示します。データビューを直接開くリンクを作成するに

は、イメージ オブジェクトを右 クリックして [URL の設定] を選択します(以下の例では、色の付いた矢印および付随するテキストは、ビュー URL へのリンクです)。データ ソースが事前に読み込まれた空のビューをポイントして、ユーザーに Web 作成ワークスペースで新しい Tableau コンテンツを作成するよう促すこともできます。

ヒント: テキスト オブジェクトにハイパーリンクを追加するには、フル URL を含めます (`http://www.tableau.com` など)。



ポータルのパブリッシュ、テストおよび微調整

ワークブックをサーバーにパブリッシュして、ダッシュボードURL をユーザーに配布します。ポータル デザインは、データ分析と同様に、周期的プロセスです。この時点でポータルはまだ未開のため、体験を継続的に改善できるよう、ユーザー フィードバックの収集を開始します

> **ヒント:** 仕上げに、Tableau ツールバーを非表示にして、ポータルにカスタムの外観を与えます。ダッシュボードURL の最後のクエスチョン マークの後に、
> 「`:embed=y&:toolbar=n`」を追加します

コンテンツ アクセスの管理

サイトのコンテンツにアクセスできるユーザーを管理し、コンテンツの所有権を制御するパーミッションを設定できます。

サイトの Web 作成アクセスと機能の設定

Tableau Server 管理者は、パブリッシュされたビューをユーザーがWeb 環境で編集し、他の Web 作成機能を構成できるかどうかを、サイト レベルで指定できます。

既定では、Web 作成機能はすべてのサイトで有効になっています。**Web 編集**の権限を持つユーザーは、サーバーで直接ワークブックを作成して編集できます。ユーザーに、パブリッシュされたワークブックの表示と操作を許可しながら、コア情報は変更できないように設定する場合は、Web 作成をオフにします。

以下の手順は、サイト全体に対してWeb 作成および他の関連する機能を設定する方法について示しています。どのユーザーにWeb 編集を許可するかの詳しい制御を行うために、プロジェクト、グループ、パーミッションを使用できます。「Web 編集、保存、およびコンテンツのダウンロード アクセスの設定」を参照してください。

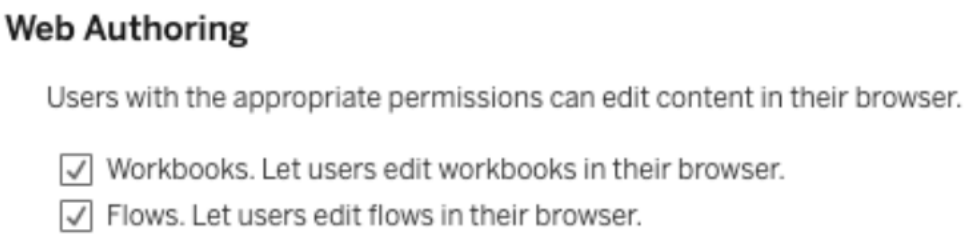
Web 上のフローの作成を有効にする方法については、「Web上でのフローの作成と操作」を参照してください。

サイトに対するWeb作成をオンまたはオフにする

1. Webブラウザーで管理者としてサーバーにサインインし、Web作成を有効化するサイトに移動します。そのサイトで、**[設定]** をクリックします。

2. **[Web作成]** セクションで、**[ワークブック]** を選択します。**ユーザーがブラウザーでワークブックを編集して**、機能を有効にすることができるようにします。

   チェックボックスをクリアし、そのサイトのWeb作成をオフにします。

3. サイトがすでに稼働中で、すぐに変更を有効にする場合は、サーバーを再起動します。

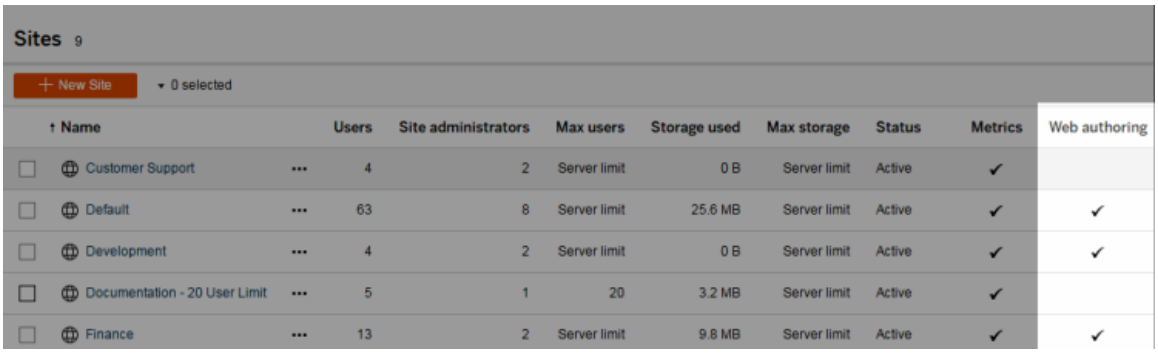
   再起動しない場合、変更内容はサーバーセッションのキャッシュ期限が切れた後、または次回ユーザーがサインアウト後に再びサインインしたときに有効になります。

注

- Web作成を有効にする場合は、適切なワークブックまたはビューで、ユーザーまたはグループに対するパーミッションルールが**Web編集**機能を許可していることを確認します。
- 本番サイトでWeb作成をオフにし、最後の手順でサーバーを再起動していない場合は、ユーザーはセッションキャッシュの期限が切れるかサインアウトするまで引き続き作成アクセス権を持つ場合があります。

どのサイトでWeb作成を許可するかを確認する

どのサイトでWeb作成が許可されているかを確認するには、最上部のサイト選択メニューで **[すべてのサイトを管理]**、**[サイト]** ページの順に選択します。



クロスデータベース結合について

クロスデータベース結合のパフォーマンスを向上させるために、Tableau はデフォルトで、Hyper を使用して Tableau 内で結合を実行するか、接続されたライブデータベースにデータを一時テーブルとして移動してそこで結合を実行するかを決定するようになりました。

各サイトの **[設定]** でクロスデータベース結合を構成するためのオプションは引き続き表示されますが、デフォルトから変更することはできなくなりました。

この機能の詳細については、「クロスデータベース結合でのパフォーマンスの向上」を参照してください。

コンテンツの Web 編集、保存、およびダウンロードアクセスを設定する

Web 作成機能をサイトで有効にしている場合は、サイトでこの機能にアクセスできるユーザーを正確に設定することができます。コンテンツレベルでのサイトロールとパーミッションルールを使用して、プロジェクト、ワークブック、データソースに対する **Web 編集**、**保存**、または**ダウンロード**機能のパーミッションを付与または拒否できます。

注: 本ドキュメントでは、パーミッション ルールで機能名を指定するために *Web 編集*というフレーズを使用します。また、*web 作成*とは、サーバー上でワークブックを作成したり変更したりする一般的な機能を指します。ただし、これらのフレーズが入れ替わって使用される場合もあります。

ユーザーのサイトでの直接作業を許可する理由

管理者として、人々がサイトにコンテンツを投稿するのを表面的にでも無条件に許可することには疑問を持たれるかもしれません。しかし、少し制御するだけでその実行場所を管理することができ、また、コンテンツを一元管理することは管理者とユーザーの双方に大きなメリットがあります。

Web 作成の是非

パブリッシャーやビジネス ユーザーにとって、Web 作成には以下のメリットがあります。

- 入力を提供する中心的な場所と共に、共同で作業に参加するアナリスト チームが提供される。
- Tableau Desktop を持たないユーザーがデータ ソースに接続してワークブックを作成できるようになる。
- Tableau Desktop コンピューターや VPN から離れた場所にいるユーザーが、コンピューターとハンドヘルド デバイスのどちらを使用していても、コンテンツにアクセスできるようになる。
- Tableau レポート全般の一貫性を保つことができるフレームワークが提供される。ワークブックのテンプレートをサイト上で利用可能にすることにより、アナリストは新しいワークブックをダウンロードしたり、接続、ブランディング、書式設定がすでに完了しているデータを使用して新しいワークブックを作成したりできます。

管理者にとっては、次のようなメリットがあります。

- 管理とサポートのために展開する Tableau Desktop が少なくなる。
- データベース ドライバーをインストールする必要があるコンピューターが少なくなる。
- コンテンツを管理する権限。
- ユーザーが Tableau で何を行うかより正確に監視できる。

Web 編集には、次のような不利な点があります。

- アナリストにとって、Web 編集は Tableau Desktop と同程度には機能が豊富ではない (同程度に近づきつつはあります)。

- 管理者にとって、サーバー上で作業する人が多いと、システムのアップグレードの必要が生じる。
- パブリッシュのガイドラインを設けることなくサイト上でコンテンツを拡散できる可能性がある。これは、パブリッシュされている Tableau のダッシュボードとデータソースに依存する人々を混乱させるだけでなく、結局はサーバーのパフォーマンスやデータの品質を低下させる可能性があり、データのセキュリティにも影響を及ぼしかねません。

ユーザーに役立つパーミッションの管理でコンテンツの急激な拡散を防ぐ

コンテンツのサイト上での急激な拡散を防ぐようにユーザーを支援するために、Tableau 管理者の多くは、コンテンツに対する多様なレベルのアクセス権をプロジェクトで許可しています。たとえば、あるプロジェクトでは、すべてのユーザーにワークブックの編集と保存を許可し、別のプロジェクトでは、認可されたパブリッシャーのみ新しいコンテンツを保存できるように構成できます。

この方法がどのように機能するかの詳細については、次のリソースを参照してください。

- マネージドセルフサービスのためのプロジェクト、グループ、グループセット、パーミッションの設定
- 「大規模な管理セルフサービス」、Tableau ホワイトペーパー、Rupali Jain
  PDF を表示するには Tableau Web サイトの認証資格情報が必要になる場合があります。これはコミュニティフォーラムで使用したり、サポート事例の送信で使用するものと同じです。

適切なレベルのアクセス権のための編集、保存権限とサイトロールを調整する

ワークブックを編集、保存、およびダウンロードするには、これらのアクションを許可するサイトロールに加えて、ユーザーは編集関連のアクセス権を付与または拒否するパーミッションルールで定義されている権限を持っている必要があります。

サイトロール アクセス

- 適切なパーミッションがコンテンツレベルで設定されている場合、**Creator** または **Explorer (パブリッシュ可能)** のサイトロールでは **[保存]** (上書き) および **[名前を付けて保存]/[ダウンロード]** の両方が許可されています。

  [ファイル] > [保存] はワークブック所有者のみが使用できます。プロジェクトおよびワークブックレベルで **[保存]** パーミッション機能が付与されている場合、所有者以外のユーザーは、**[ファ**

イル] > [名前を付けて保存] を選択し、同じワークブック名を使用して、Web 作成で既存のワークブックを上書きできます。これにより、既存のコンテンツが上書きされ、所有者になってコンテンツへのフル アクセス権が得られます。

- **Explorer** サイト ロールには、**[Web 編集]** 機能と**[名前を付けて保存]/[ダウンロード]** 機能を付与できますが、保存することはできません (既存のワークブックを上書きしたり、新しいワークブックへの変更を保存したりすることはできません)。

詳細については、Web 編集とWeb 作成を参照してください。

マネージド セルフサービスのためのプロジェクト、グループ、グループ セット、パーミッションの設定

Tableau Cloud とTableau Server へのパブリッシュは簡単です。組織によっては、少し*簡単すぎる*くらいの作業です。有用な方法として、作成者が独自のコンテンツをパブリッシュできるようになる前に、制御されたフレームワークを作成します。

コンテンツを整理して、ユーザーが適切なコンテンツを見つけてアクセスできるようにするには、マネージド セルフサービスのサイトを構成すると便利です。つまり、コンテンツが整理され、検出可能で、安全であることを保証するためのガイドラインと設定を用意することで、パブリッシュプロセスのボトルネックを解消することができます。

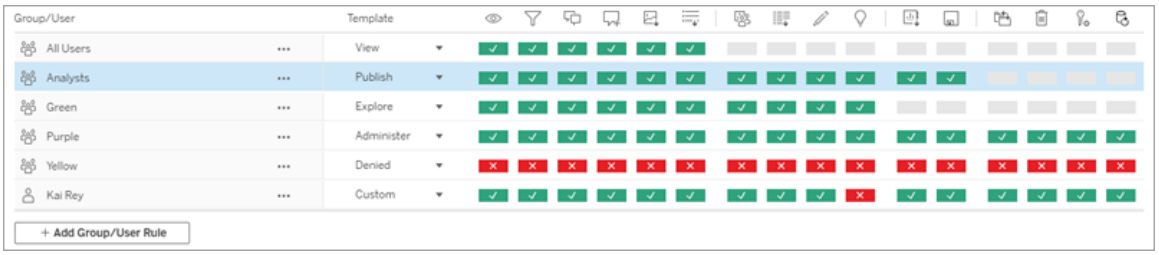
この記事では、、サイト管理者がマネージド セルフサービスのサイトを設定するために活用できる手法を示します。

1. 必要なグループとプロジェクトのタイプを特定する
2. グループとグループ セットを作成する
3. あいまいな点が残るパーミッションを削除し、既定のパーミッション パターンを確立する
4. プロジェクトの作成
5. プロジェクト パーミッションをロックする

注: ここで提供する情報は、Tableau Visionaries および自身の経験を共有してくれた顧客のプラクティスを基に変更を加え、簡略化しています。

展開を計画する

Tableau のパーミッションは、グループまたはユーザーのコンテンツ (プロジェクトやワークブックなど) に適用されるルールで構成されます。これらのパーミッション ルールは、特定の機能を許可または拒否することによって作成されます。



プロジェクト、グループ、パーミッション ルールの包括的な計画を立てることは、新規に開始する場合でも変更を加える場合でも役立ちます。詳細は自分で決められますが、すべての環境で推奨されている 2 つの重要なプラクティスがあります。

- 個々のコンテンツではなく、プロジェクトでパーミッションを管理します。
- 個々のユーザーではなく、グループにパーミッションを割り当てます。

個々のユーザー レベルおよび各コンテンツ資産に対してパーミッションを設定していると、すぐに管理不可能になります。

クローズド パーミッション モデルを使用する

パーミッション設定の一般的なモデルは *オープン*または *クローズ*です。オープン モデルでは、ユーザーは高レベルなアクセス権を取得し、管理者は明示的に権限を拒否します。クローズド モデルでは、ユーザーは自分たちの *ジョブ*を実行するために必要なアクセス権のみを取得します。これは、セキュリティの専門家が提唱するモデルです。このトピックの例は、クローズド モデルに準じています。

Tableau パーミッションの評価方法の詳細については、「有効なパーミッション」を参照してください。

必要なプロジェクトとグループのタイプを特定する

コンテンツ (プロジェクト) とユーザーのカテゴリ (グループ)、またはグループのカテゴリ (グループ セット) を収容する構造を設計することは、サイトを立ち上げる上で最も難しい部分ですが、この設計により継続的な管理がはるかに簡単になります。

プロジェクト: プロジェクトは、パーミッションを管理するためのユニットとしても、組織的およびナビゲーション的なフレームワークとしても機能します。ユーザーが期待するコンテンツの検索方法と論理的なパーミッション付与のバランスがとれているプロジェクト構造を作成します。

グループまたはグループ セット: グループを作成する前に、ユーザーがコンテンツをどのように操作するかについて、共通のテーマを見つけると役に立ちます。グループまたはグループ セットの作成に使用できるパターンを特定し、個々のユーザーに対する1回限りのパーミッションは避けます。

例 1: プロジェクトとグループの構造

たとえば、すべての人がアクセスできる会社全体のコンテンツと、制限を設ける必要があるいくつかの HR コンテンツがある環境を考えてみましょう。

プロジェクトには以下が含まれます。

- **Acme Corp 会議**。チケット販売のデータ ソースとワークブック、コンテンツ戦略のダッシュボード、社内会議のプロジェクト計画が含まれます。
- **従業員の業績**。従業員の内部調査に関する匿名化されたデータ ソースとワークブックが含まれます。
- **人事**。HR チームのメンバーのみが利用できる HR データ ソースとワークブックが含まれます。

各グループは従業員が行うべき内容と一致している必要があります。

- **コア コンテンツ作成者**。このグループは、上位レベルのプロジェクトにパブリッシュでき、データ ソースに幅広くアクセスできるが、コンテンツを移動したり管理したりする必要がないユーザーを対象としています。
- **人事 コンテンツ作成者。**このグループは、HR データ ソースにアクセスでき、HR プロジェクトにパブリッシュできるユーザーを対象としています。
- **ビジネス ユーザー。**このグループは、コア コンテンツ作成者によって作成されたコンテンツにアクセスできる必要があるが、HR コンテンツが存在することさえ知らないユーザーを対象としています。

- **人事ユーザー。**このグループは、HR プロジェクトのコンテンツにアクセスする必要があるが、コンテンツを作成またはパブリッシュする権限を持っていないユーザーを対象としています。
- **コア プロジェクト リーダー。**このグループは、HR ではないプロジェクトでプロジェクト リーダーのステータスが付与される必要があるユーザーを対象としています。

例 2: グループとグループセットの構造

Tableau Server 2024.2 以降、グループ セットのレベルでパーミッションを有効にすることで、ユーザーに付与 (または拒否) する機能をさらに制御できます。グループ セットのレベルで権限が設定されている場合、ユーザーは評価されるグループ セット内のすべてのグループに属している必要があります。

注: グループ セットのパーミッション ルールは、ユーザー ルールとグループ ルールの後に評価されます。

たとえば、上記の例 1 でユーザーのニーズに合わせてグループを作成したとします。次のグループ セットを作成すると、人事アクセスをさらに制限できます。

- **人事リーダー。**このグループ セットは、人事コンテンツ作成者とコア プロジェクト リーダーで構成されます。このグループ セットのユーザーが両方のグループに属している場合にのみ、プロジェクト リーダーのステータス、機密性の高い人事データ ソースへのアクセス権、および人事プロジェクトへのパブリッシュの権限が付与されます。

サイト ロールを考慮する

パーミッションは、グループやユーザーではなく、コンテンツに関連付けられていることに注意してください。グループに **[探索]** パーミッションを単独で付与することはできません。グループに付与できるのは、プロジェクトとそのコンテンツの **[探索]** パーミッションです。ただし、サイト ロールが特定のユーザーに付与されると、ユーザーが持つことができるパーミッションが定義または制限される場合があります。ライセンス、サイト ロール、パーミッションを関連付ける方法の詳細については、「パーミッション、サイト ロール、ライセンス」を参照してください。

グループとグループ セットを作成する

必要な情報を特定したらすぐにグループやプロジェクトを作成したくなるかもしれませんが、特定の順序で作成することが重要です。

プロジェクト: プロジェクトは、[既定] プロジェクトを適切に構成してから作成してください (次のセクションを参照)。これは、最上位のプロジェクトが [既定] プロジェクトをパーミッション ルールのテンプレートとして使用するためです。

グループ: グループを作成してから、パーミッション ルールを作成する必要があります。ユーザーはまだグループに追加する必要はありませんが、追加することはできます。グループの作成とユーザーの追加の詳細については、グループおよびグループへのユーザーの追加を参照してください。

グループ セット: パーミッション ルールの作成に使用するには、まず、グループを作成する必要があります。ユーザーはまだグループに追加する必要はありませんが、追加することはできます。詳細については、「グループ セットの操作」を参照してください。

ヒント: 複数のグループとプロジェクトを作成して、パーミッションを手動で設定するのは、面倒な作業です。これらのプロセスを自動化し、将来のアップデートで繰り返し実行できるようにするために、これらのタスクを REST API コマンドを使用して実行できます。tabcmd コマンドは、単一のプロジェクトやグループの追加や削除、ユーザーの追加などのタスクに使用できますが、パーミッションの設定には使用できません。

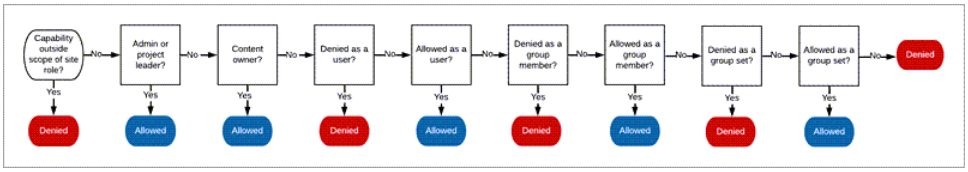
複数のグループのメンバーシップ

[HR コンテンツ作成者] グループと [HR ユーザー] グループのユーザーを [ビジネス ユーザー] グループに含めることができます。これにより、大半のコンテンツのパーミッションを、コア コンテンツ ユーザーとビジネス ユーザーに簡単に割り当てることができます。ただし、そのシナリオでは、[ビジネス ユーザー] グループが人事フォルダーの機能を拒否されると、HR ユーザーも拒否されることになります。代わりに、[ビジネス ユーザー] グループは未指定のままにしておく必要があります。これにより、特定の [HR コンテンツ作成者] グループと [HR ユーザー] グループに適切な機能が付与されます。

これは、Tableau パーミッションに制限が設けられているためです。[ビジネス ユーザー] グループが特定の機能を拒否された場合、その拒否によって、両方のグループのユーザーに対して許可されている別のパーミッション ルールが上書きされてしまいます。

グループ セットの影響

割り当てられたパーミッションがグループ セットのレベルで有効になっている場合、機能を許可するには、グループ セット内のすべてのグループに対して、パーミッションが指定されていたり、拒否されていたりしてはなりません。



グループ メンバーシップの割り当て方法を決定する際は、パーミッション ルールの評価方法を理解することが重要です。詳細については、「有効なパーミッション」を参照してください。

あいまいな点が残るパーミッションを削除し、既定のパーミッション パターンを確立する

すべてのサイトには、**[すべてのユーザー]** グループと**[既定]** プロジェクトがあります。

[すべてのユーザー] グループ: サイトに追加されたすべてのユーザーは自動的に [すべてのユーザー] グループのメンバーになります。複数のグループに設定されたパーミッション ルールとの混同を避けるために、[すべてのユーザー] グループからパーミッションを削除することをお勧めします。

[既定] プロジェクト: [既定] プロジェクトは、サイト内の新しいプロジェクトのテンプレートとして機能します。すべての新しい最上位レベル プロジェクトは、[既定] プロジェクトからパーミッション ルールを取得します。[既定] プロジェクトでベースラインのパーミッション パターンを確立すると、新しいプロジェクトの予測可能な開始点になります(ネストされたプロジェクトは、[既定] プロジェクトではなく、親プロジェクトからパーミッション ルールを継承することに注意してください)。

[既定] プロジェクトで [すべてのユーザー] グループのパーミッション ルールを削除します。

1. **[探索]** を選択し、サイトで最上位レベル プロジェクトを表示します。
2. **[既定]** プロジェクトの **[アクション]** (…) メニューで、**[パーミッション]** を選択します。
3. **[すべてのユーザー]** グループ名の横にある **[…]** を選択し、**[Delete Rule… (ルールを削除…)]** を選択します。

これにより、すべてのユーザーに競合するパーミッションを割り当てることなく、完全に制御できるグループに対してパーミッション ルールを確立できます。複数のルールを評価して、有効なパーミッションを決定する方法の詳細については、「有効なパーミッション」を参照してください。

パーミッション ルールを作成する

すべての新しい最上位レベル プロジェクトが継承する [既定] プロジェクトの基本的なパーミッション パターンを設定できるようになりました。[既定] プロジェクトのパーミッション ルールを空のままにして、新しい最上位レベル プロジェクトごとにパーミッションを個別に作成することもできます。ただし、大半のプロジェクトに適用する必要のあるパーミッション ルールがある場合は、[既定] プロジェクトでそれらのルールを設定すると便利です。

プロジェクトのパーミッション ダイアログには、コンテンツのタイプごとにタブが含まれています。**プロジェクト レベルでコンテンツのタイプごとにパーミッションを設定する必要があります**。設定しない場合、ユーザーはそのコンテンツ タイプへのアクセスを拒否されます(ユーザーに明示的に許可されている場合にのみ、機能が付与されます。機能を [未指定] にすると、その機能は拒否されます。詳細については、「有効なパーミッション」を参照してください)。

ヒント: プロジェクト レベルでパーミッション ルールを作成するたびに、すべてのコンテンツ タイプ タブを確認してください。

必要に応じて、パーミッション ルールを作成します。

1. **[+ グループ/ユーザー ルールの追加]** をクリックし、入力を開始してグループ名を検索します。
2. 各タブで、ドロップダウンから既存のテンプレートを選択するか、または、機能をクリックし、カスタム ルールを作成します。
3. 完了したら、[保存] をクリックします。

パーミッションの設定の詳細については、「パーミッションの設定」を参照してください。

例: 各コンテンツタイプに対するプロジェクトレベルのパーミッション

この例では、ほとんどのユーザーがプロジェクトの大部分を利用できるようにする必要があります。[既定]プロジェクトでは、パーミッション ルール テンプレートを使用して、コア コンテンツ作成者にパブリッシュ権限を付与し、他のすべてのユーザーにはワークブックを操作する権限のみを付与します。

グループ	プロジェクト	ワークブック	データ ソース	(その他のコンテンツ)
コア コンテンツ作成者	パブリッシュ	パブリッシュ	パブリッシュ	ビュー
HR コンテンツ作成者	ビュー	探索	ビュー	なし
ビジネス ユーザー	ビュー	探索	ビュー	なし
HR ユーザー	ビュー	探索	ビュー	なし
コア プロジェクト リーダー	プロジェクト リーダーとして設定	N/A	N/A	N/A

このパターンはクローズド モデルに従い、ほとんどのユーザーに対して、大半のコンテンツの基本的なパーミッションのみを使用するよう制限を設けます。新しい最上位レベル プロジェクトが作成されると、これらのルールは既定で継承されますが、パーミッション ルールは必要に応じてプロジェクトごとに変更できます。**[人事]**プロジェクトでは、これらのパーミッションを削除し、独自のパターンを確立する必要があることに注意してください。

プロジェクトを作成し、パーミッションを調整する

[既定]プロジェクトをカスタム パーミッション テンプレートを使用して設定したら、残りのロジェクトを作成できます。各プロジェクトで、必要に応じて既定のパーミッションを調整できます。

プロジェクトを作成するには、次の手順を実行します。

1. **[探索]** を選択し、サイトで最上位レベル プロジェクトを表示します。
2. **[新規]** ドロップダウンから、**[プロジェクト]** を選択します。
3. プロジェクトに名前を付け、必要に応じて説明を追加します。

命名規則を設定すると便利です。たとえば、基本的な構造を <DepartmentPrefix><Team> - <ContentUse> (例: DevOps - モニタリング) という形式にできます。

プロジェクトのサムネイルと**[プロジェクト詳細]** ページにカーソルを合わせると、説明が表示されます。適切な説明があることで、操作が正しいことをユーザーに知らせることができます。

4. 必要に応じてパーミッションを調整します。
    a. 新しいプロジェクトを開きます。
    b. [アクション] メニュー (...) から、[パーミッション] を選択します。
    c. 必要に応じて、パーミッション ルールを変更します。*すべてのコンテンツ タブを確認することを忘れないでください。*

コンテンツ パーミッションをロックする

パーミッション ルールに加えて、プロジェクトにはコンテンツ パーミッションの設定があります。この設定は、**[ロック]** (推奨) または **[カスタマイズ可能]** の 2 つの方法のいずれかで構成できます。

プロジェクトをロックすることで、一貫性が維持され、プロジェクト内のすべてのコンテンツに (コンテンツ タイプごとに) 統一されたパーミッションが設定されます。カスタマイズ可能なプロジェクトでは、権限を持つユーザーがコンテンツに個別のパーミッション ルールを設定できます。詳細については、「コンテンツ パーミッションをロックする」を参照してください。

コンテンツ パーミッションの設定に関係なく、パーミッションは常にコンテンツに適用されます。

可能なプロジェクト構造

組織によっては、特定の目的に応じたプロジェクトを作成すると便利な場合があります。以下にプロジェクトの例とその使用目的を示します。これらはサンプル テンプレートであるため、必ずお使いの環境で構成をテストしてください。

各コンテンツタイプのパーミッションルールテンプレートに含まれる機能については、パーミッションの機能を参照してください。

例: 特定の目的のためのパーミッション設定

サーバー上で自由にコラボレーションできる共有されたワークブック

同じ部門にいる誰もが公開されたコラボレーションプロジェクトにパブリッシュできます。コンテンツは開発中の段階です。同僚はサーバーの Web 編集機能を使用してコラボレーションできます。この機能は Sandbox と呼ばれたり、ステージングと呼ばれたりします。このプロジェクトでは、Web の編集、保存、ダウンロードなどを実行できます。

共同で作業できるだけでなく、Tableau Desktop を持っていないユーザーでもフィードバックが寄せられるようになります。

グループ	プロジェクト	ワークブック	データソース	(その他のコンテンツ)
データ案内人	パブリッシュ	パブリッシュ	パブリッシュ	*TBD*
アナリスト	パブリッシュ	パブリッシュ	探索	*TBD*
ビジネスユーザー	パブリッシュ	パブリッシュ	探索	*TBD*

パブリッシュテンプレートの一部の機能 (上書きなど) は、その機能が許可されている場合でも、ユーザーのサイトロールによって制限される可能性があることに注意してください。

注:「*TBD*」は、これらのパーミッションルールがこのシナリオでは簡単に判断できないことを示しますが、特定の環境に適している場合は設定できます。

編集不可能な共有レポート

これは、ワークブックおよびデータソースを作成するユーザー(アナリストおよびデータ案内人)が、「真似」されたり修正されたりすることがないと確信できるコンテンツをビジネスユーザーに対して閲覧できるようにする場合にパブリッシュできるプロジェクトです。

このタイプのプロジェクトでは、サーバーのデータを再利用するために編集または取得できる機能をすべて拒否します。許可するのは機能の閲覧です。

グループ	プロジェクト	ワークブック	データソース	(その他のコンテンツ)
データ案内人	パブリッシュ	*TBD*	パブリッシュ	*TBD*
アナリスト	パブリッシュ	パブリッシュ	ビュー	*TBD*
ビジネスユーザー	ビュー	ビュー	なし	なし

アナリストが接続する吟味されたデータソース

これは、すべてのデータ要件を満たし、組織にとって「信頼できるソース」となるデータソースをデータ案内人がパブリッシュする場所になります。このプロジェクトのプロジェクトリーダーは、これらのデータソースを承認して検索結果の高ランクに位置付けたり、お勧めのデータソースに含めたりできます。

認可されたアナリストがワークブックをこのプロジェクトのデータソースに接続することは許可しますが、ダウンロードや編集はできないようにします。[ビジネスユーザー] グループへのビュー機能を拒否するため、このグループのユーザーはこのプロジェクトを表示することもできません。

グループ	プロジェクト	ワークブック	データソース	(その他のコンテンツ)
データ案内人	パブリッシュ	*TBD*	パブリッシュ	*TBD*
アナリスト	ビュー	なし	ビュー	なし
ビジネスユーザー	なし	なし	なし	なし

非アクティブなコンテンツ

サイトの管理ビューが一定期間表示されていないワークブックおよびデータソースを隔離することもできです。コンテンツをサーバーから削除する前にコンテンツ所有者に時間制限を与えます。

これを行うか、ワーキングプロジェクトから直接削除してしまうかは、組織で決定します。アクティブな環境にある場合は、使用されていないコンテンツが削除される心配はありません。

グループ	プロジェクト	ワークブック	データソース	(その他のコンテンツ)
データ案内人	なし	なし	なし	なし
アナリスト	ビュー	ビュー	*TBD*	*TBD*
ビジネスユーザー	なし	なし	なし	なし

ワークブックテンプレートのソース

これは、ユーザーがダウンロードできるがパブリッシュや保存はできないプロジェクトです。認可されたパブリッシャーまたはプロジェクトリーダーによってワークブックのテンプレートが利用できるようになります。組織で許可されたフォント、色、画像、および内蔵されたデータ接続を使用したテンプレートによって、作成者は大幅な時間短縮を実現し、レポートのスタイルに一貫性を持たせることができます。

グループ	プロジェクト	ワークブック	データソース	(その他のコンテンツ)
認可された作成者	パブリッシュ	パブリッシュ	パブリッシュ	*TBD*
データ案内人	なし	なし	なし	なし
アナリスト	ビュー	*テンプレート:* 探索 +	ビュー	なし

		機能: ワークブックのダウンロード/コピーの保存			
ビジネス ユーザー	なし	なし		なし	なし

次のステップ

プロジェクト、グループ、パーミッションに加え、その他のデータ管理のテーマとして次の内容があります。

ユーザーの教育

すべてのTableau ユーザーが優れたデータ案内人になれるようにします。最も成功を収めているTableau 組織では、Tableau のユーザー グループを作成して定期的にトレーニング セッションなどを開催しています。

ユーザーがサイトを正しく理解できるようになるための一般的な手法については、ダッシュボードベースのカスタム ポータルを参照してください。

パブリッシュおよびデータ認証のヒントについては、以下のトピックを参照してください。

- ユーザーが信頼できるデータを見つけられるように認証を使用する
- ワークブックのパブリッシュの準備 (Tableau ヘルプ)
- パブリッシュされたデータ ソースのベストプラクティス (Tableau ヘルプ)

抽出の更新およびサブスクリプションのアクティビティの最適化

Tableau Server を使用している場合は、抽出の更新およびサブスクリプション スケジュールに関するポリシーを作成して、サイトのリソースに影響を与えないようにします。Wells Fargo および Sprint によるTC カスタマーのプレゼンテーションで、これらの課題について詳しく取り上げています。また、「パフォーマンス調整」のトピックも参照してください。

Tableau Cloud を使用する場合は、次のトピックを参照して抽出の更新を実行する方法についてよく理解しておいてください。

- データを最新に保つ
- Tableau Bridge を使用したデータの鮮度に関するオプションの拡張

監視

管理ビューを使用して、サイトのパフォーマンスおよびコンテンツの使用を監視します。

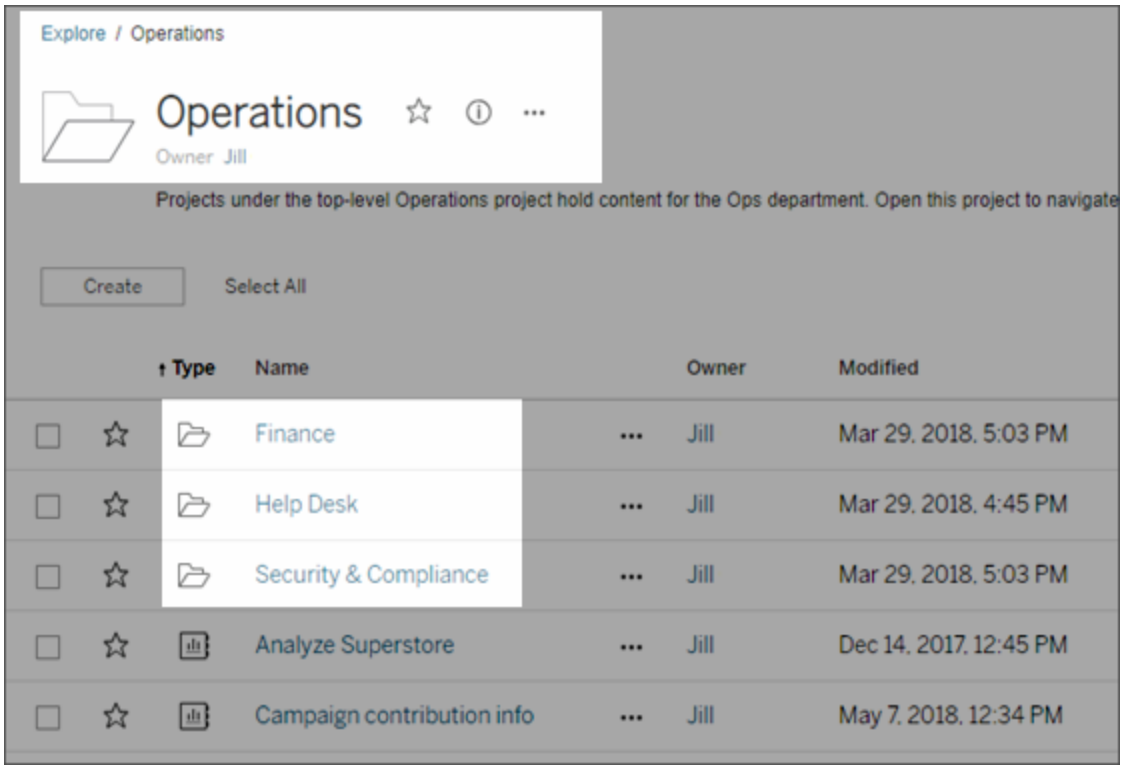
管理ビュー

プロジェクトを使用したコンテンツへのアクセスの管理

Tableau Desktop ユーザーが Tableau Server のサイトにコンテンツをパブリッシュするときに、パブリッシュ先の*プロジェクト*を選択できます。

プロジェクトは、ワークブック、データソース、レンズ、ネストされたプロジェクトなどのアセットのナビゲーション、整理、およびアクセス管理に使用できます。Tableau Cloud 2022 年 10 月/Server 2022.3 以降では、データ管理のライセンスが付与され、Catalog が有効になっている場合、プロジェクトにデータベースなどの外部アセットも含めることができます。

以下のイメージでは、Web 作成環境で "Operations (運営)" のトップレベルプロジェクト内のコンテンツが表示されています。"Operations (運営)" プロジェクトにはいくつかのネストされたプロジェクト (ハイライト表示) とパブリッシュされたワークブックが含まれています。プロジェクトには、他のアセットタイプを含めることもできます。



プロジェクトを使用する理由

プロジェクトは、Tableau Server にパブリッシュされたコンテンツへのアクセスを管理するために、拡張可能なプロセスを作成するのに役立ちます。プロジェクトの利点:

- 管理者は、より密接にコンテンツを使用するプロジェクト リーダーにコンテンツ管理を委任できます。サイトまたはサーバー設定への管理者アクセス権限をプロジェクト リーダーに付与する必要はありません。
  - プロジェクト リーダーはネストされたプロジェクトをトップレベル プロジェクトの下に作成でき、単一階層内にチームのコンテンツを維持することが可能となります。
  - **注:** プロジェクト所有者は、自分が所有する上位レベルのプロジェクトを削除できます。プロジェクト リーダーはトップレベル プロジェクトを削除できません。
- セルフサービス ユーザーがサイトを参照しやすくなります。
  - Tableau Server サイトをエリアに分割します。これらのエリアは、エリアにパブリッシュされたデータの使用方法、または使用する Tableau ユーザー グループに基づいて、ユーザーにアクセス権を付与します。

    - プロジェクトを使用する必要がないグループにはプロジェクトを非表示にしたり、区別可能なプロジェクトの命名規則を作成したり、プロジェクトの説明を活用してプロジェクトの使用方法を明確にしたりできます。
- パーミッションを効果的に追跡できます。
    - グループ内のユーザーが必要とするコンテンツアクセス権のレベルに基づいてグループを作成し、プロジェクトに既定のパーミッションを作成できます。これにより、新しいユーザーが既定で取得する機能がどれか、またこれと同様に、新しいプロジェクトの作成時にすべてのユーザーが取得する機能がどれかを正確に把握することができます。

プロジェクト階層を作成する場合 (例)

多くの組織には、それぞれ独特な Tableau ユーザーグループがあり、それぞれに独自の優先順位やリーダーが存在します。これらのグループは、組織全体に関わるコンテンツを共有する (もしくは、組織全体のデータソースプールからコンテンツを取り出す) こともありますが、主に、自分のチーム専用のデータやレポートを使用します。このようなまたはこれに近いシナリオの場合、プロジェクト階層の使用例は次のようになります。

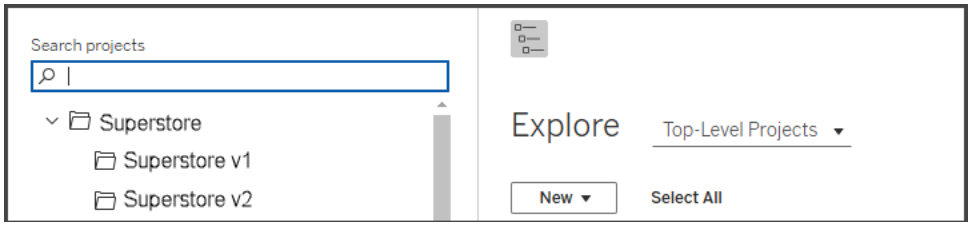
1. サイトまたはサーバー管理者は、異なる Tableau チームごとにトップレベルプロジェクトを作成することができます。
2. 各トップレベルプロジェクトごとに、チームリーダーにプロジェクトリーダーのステータスを割り当て、プロジェクトの所有権を変更します。プロジェクトリーダーは実質的にはコンテンツ管理者であるため、Tableau におけるパーミッションのしくみと、Tableau のコンテンツ管理のベストプラクティスを理解する必要があります。
3. 各プロジェクトリーダーは自分のプロジェクトを管理し、チームにとって都合の良い構造をプロジェクト内に作成することができます。つまり、チームメンバーがどのように協力し、どのようにデータやレポートを共有するかを考慮して、必要な子プロジェクトを作成することができます。

サイト管理者にとってのメリットは、システムの健康状態に集中できることです。Tableau ユーザーにとってのメリットは、Tableau およびデータを使用するためのベストプラクティスを理解した人物がチームのためにこれらを管理できることです。パーミッションの変更やプロジェクトの追加を行うために、IT 部門にリクエストを提出する必要はありません。

プロジェクトツリーを使用してプロジェクト間を移動する

既定では、[探索] ペインにはプロジェクトツリーが表示されるので、プロジェクト間を簡単に移動できます。プロジェクトツリーを使用して、プロジェクトフォルダー内のサブプロジェクトを見つけることが

できます。プロジェクト ツリーを使用してプロジェクトを検索することもできます。



[探索] ペインにあるプロジェクト ツリーのアイコンをクリックすると、プロジェクト ツリーを非表示にしたり、もう一度表示したりできます。

注: プロジェクト ツリーが表示されるようにするには、[探索] ページでブラウザー ウィンドウの幅を 1441 ピクセル以上に維持します。ウィンドウが狭い (1440px 以下) か、ディスプレイの解像度が低いと、プロジェクト ツリーは表示されません。

サイトを使用しない理由

すべての段階においてコンテンツが完全に別々のままで、なおかつユーザーの重複がほとんどない場合は、サイトがうまく機能します。複数のサイトを使用する良い (一般的な) 例は、コンサルタントまたはベンダーとして、複数の外部クライアントのパブリッシュされたコンテンツを管理している状況で、そのクライアントごとにサイトを作成する場合です。

プロジェクトにより、共有されたデータやレポート、複数のグループに属するユーザーを柔軟に管理することができます。開発からステージング環境、さらに本番環境へとコンテンツを進めていく上で、プロジェクトはサイトよりもうまく機能します。

プロジェクト レベルの管理

プロジェクトの管理の詳細については、「プロジェクトを使用したパーミッションの管理」を参照してください。

プロジェクトを追加してそこにコンテンツを移動させる

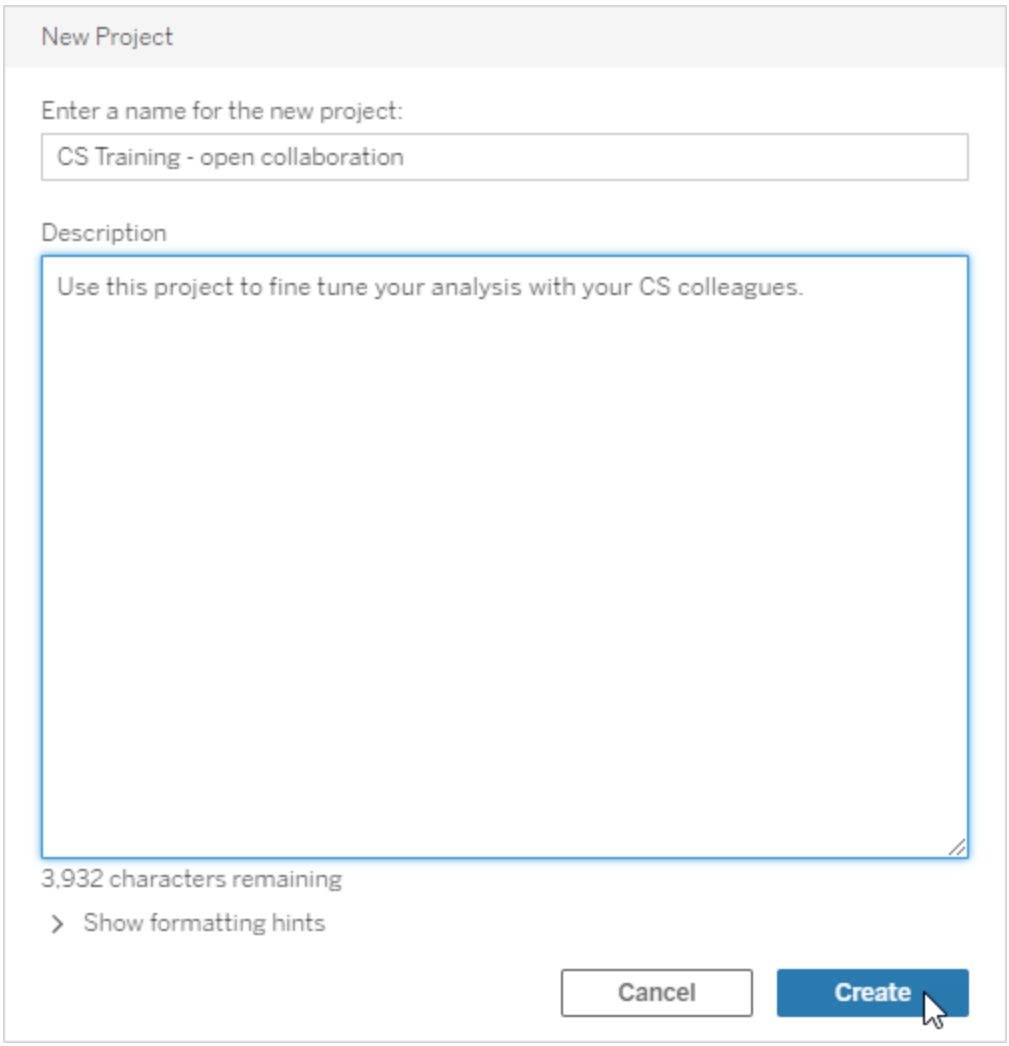
Tableau コンテンツ (ワークブック、データソースなど) がプロジェクトに含まれている必要があります。Tableau Server 2022.3 および Tableau Cloud 2022 年 10 月以降では、データ管理のライセンスが付与され、Catalog が有効になっていれば、外部アセット (データベースやテーブルなど) もプロジェクトに含めることができます。サーバーおよびサイト管理者はサイト上のトップレベルプロジェクトを追加または削除し、パブリッシュされているコンテンツを 1 つのプロジェクトから別のプロジェクトに移動できます。適切なサイトロールを持つプロジェクトリーダーは、子プロジェクトを追加または削除し、コンテンツをプロジェクトリーダーとしてのアクセス権限があるプロジェクト間で移動できます。

この記事はプロジェクトの作成と移動の手順を含みます。以下の関連コンテンツも理解しておくことをお勧めします。

- プロジェクトについて学び、使用するタイミングと理由を知るには、プロジェクトを使用したコンテンツへのアクセスの管理を参照してください。
- プロジェクト階層を作成する前に、パーミッションの内容を読み、理解しておいてください。
- 完全なプロジェクトリーダーアクセス権を許可する特定のサイトロールの詳細については、プロジェクトレベルの管理を参照してください。

トップレベルまたは子 (ネストされた) プロジェクトを追加する

1. 管理者またはプロジェクトリーダーとして Tableau Server にサインインして **[探索]** を選択し、以下のいずれかを実行します。
    - **[New (新規)] > [プロジェクト]** を選択し、新しいトップレベルプロジェクトを作成します (これを実行できるのは管理者のみです)。
    - サブプロジェクトを作成するプロジェクトに移動して、それを開き、**[New (新規)] > [プロジェクト]** を選択します。子プロジェクトがある場所がわからない場合は、**[探索]** の横にあるドロップダウンメニューから **[すべてのプロジェクト]** を選択するか、右上のフィルターを使用します。
2. プロジェクトの名前と説明を入力してから、**[作成]** をクリックします。



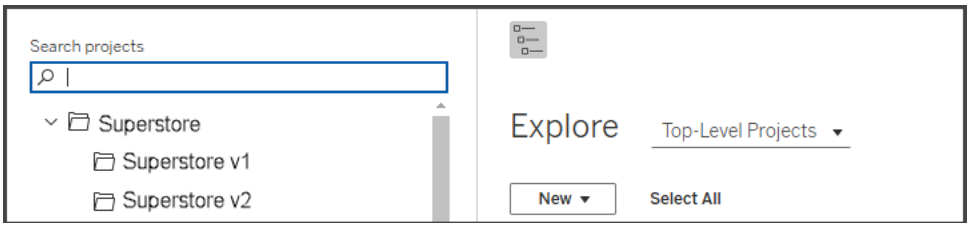
プロジェクトの説明に、書式設定やハイパーリンクを含めることができます。構文については、**[書式設定のヒントを表示]** を選択してください。

プロジェクトの説明を入力するときは、「PROJECT (a)」のように、大文字と括弧の間にスペースを入れて、括弧内の内容を表示します。このスペースを省略すると、プロジェクトの説明で表示上の問題が発生します。

注: プロジェクトの説明を後から編集するには、選択して開き、その名前の隣にある情報アイコンを選択し、**[編集]** をクリックします。

プロジェクト ツリーを使用してプロジェクト間を移動する

既定では、[探索] ペインにはプロジェクト ツリーが表示されるので、プロジェクト間を簡単に移動できます。プロジェクト ツリーを使用して、プロジェクト フォルダー内のサブプロジェクトを見つけることができます。プロジェクト ツリーを使用してプロジェクトを検索することもできます。

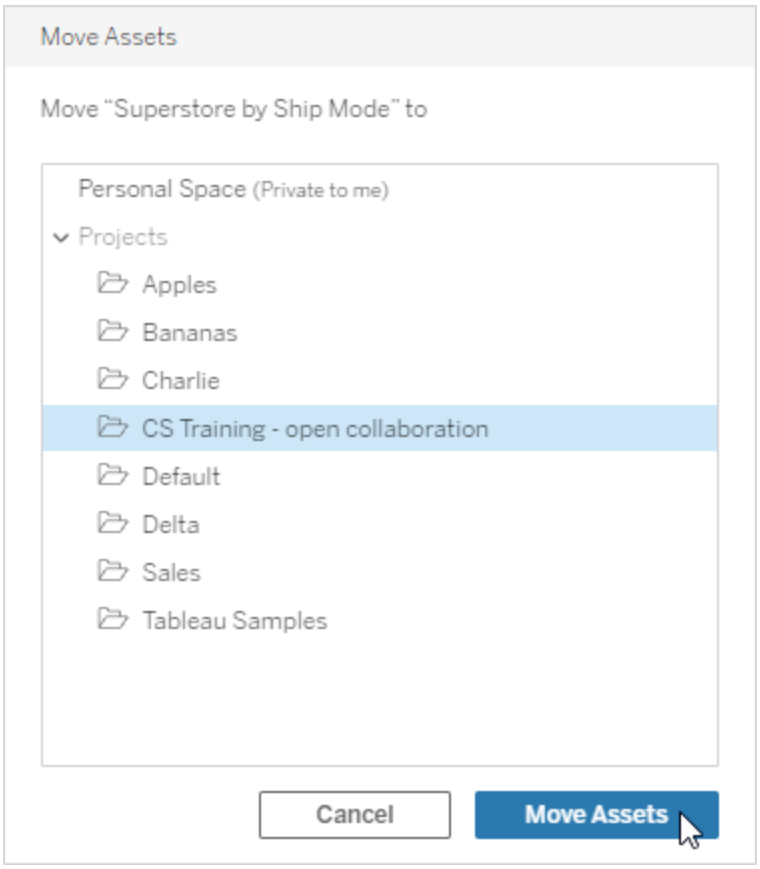


[探索] ペインにあるプロジェクト ツリーのアイコンをクリックすると、プロジェクト ツリーを非表示にしたり、もう一度表示したりできます。

注: プロジェクト ツリーが表示されるようにするには、[探索] ページでブラウザー ウィンドウの幅を 1441 ピクセル以上に維持します。ウィンドウが狭い (1440px 以下) か、ディスプレイの解像度が低いと、プロジェクト ツリーは表示されません。

アセットを別のプロジェクトに移動する

1. **[探索]** セクションで、移動するアセットを見つけます。右上のフィルターを使用して検索するか、プロジェクト階層内を検索することができます。
2. ワークブックの **[アクション]** (**...**) メニューで **[移動]** を選択します。
3. ワークブックから新しいプロジェクトを選択し、**[コンテンツを移動]** をクリックします。



プロジェクトを移動すると、子プロジェクトとそれらのアセットを含めた、プロジェクト内のすべてのコンテンツが移動します。

プロジェクトの移動がパーミッションに与える影響

プロジェクトを移動すると、プロジェクト リーダーのパーミッションは新しいプロジェクト環境に適応します。

- 対象プロジェクト階層が**ロック**された状態の場合、以前のプロジェクト リーダー パーミッションが削除され、新しいプロジェクト リーダー パーミッションが対象階層のトップ レベルに設定されている内容に従って付与されます。
- 対象プロジェクト階層が**カスタマイズ可能**な状態の場合、以前に黙示的に付与されていたプロジェクト リーダー パーミッションが削除され、明示的に設定されていたパーミッションは

保持され、新しいプロジェクト リーダー パーミッションが対象階層のトップ レベルに設定されている内容に従って付与されます。

プロジェクトとアセットを移動すると、パーミッションが影響を受ける可能性があります。詳細については、パーミッションを参照してください。

プロジェクトを削除する

プロジェクトを削除すると、そのプロジェクトに含まれているすべての Tableau コンテンツも削除されます。プロジェクトを削除し、コンテンツは削除しない場合は、コンテンツを別のプロジェクトに移動させてから、プロジェクトを削除します。

データベースやテーブルなどの外部アセットは削除されませんが、**外部アセットの既定プロジェクト**に移動されます。(Tableau Server 2022.3 以前では、アセットは **[外部アセット]** にあります)

重要

- プロジェクトの削除はやり直しできません。
- プロジェクトを削除すると、子プロジェクトとそれらのコンテンツを含む、そのプロジェクト内のすべての Tableau コンテンツが削除されますが、外部アセットは削除されません。
- **既定**プロジェクトや**外部アセットの既定プロジェクト**は削除できません。

プロジェクトを削除するには

1. **[探索]** セクションで、削除するプロジェクトを見つけます。プロジェクトがある場所がわからない場合は、**[探索]** の横にあるドロップダウン メニューから **[すべてのプロジェクト]** を選択するか、右上のフィルターを使用します。
2. プロジェクトの **[アクション]** (**...**) メニューで **[削除]** を選択します。
3. プロジェクトを削除することを確認します。

アセットを移動するための要件

アセットの移動は、アセットを 1 つのプロジェクトから削除して別のプロジェクトへパブリッシュすることと実際に似ています。管理者以外の場合は、移動元のプロジェクトと移動先のプロジェクトで必要と

なるパーミッションが異なります。

必要なサイトロール

アセットを移動するには、ユーザーに次のサイトロールのいずれかが必要です。

- サーバー管理者 (Tableau Server のみ)
- サイト管理者 Creator またはサイト管理者 Explorer
- Creator または Explorer (パブリッシュ可能)

サーバー管理者またはサイト管理者のサイトロールを持つユーザーには、追加機能は必要ありません。

コンテンツの*移動先*プロジェクトで必要となるパーミッション

管理者以外は、移動先プロジェクトに対する**パブリッシュ**パーミッション機能が必要です。

コンテンツの*移動元*プロジェクトで必要となるパーミッション

管理者以外のユーザーは、次の要件を満たす必要があります。

- 元のプロジェクトのプロジェクト所有者、プロジェクトリーダー、またはコンテンツ所有者である必要があります。

  または

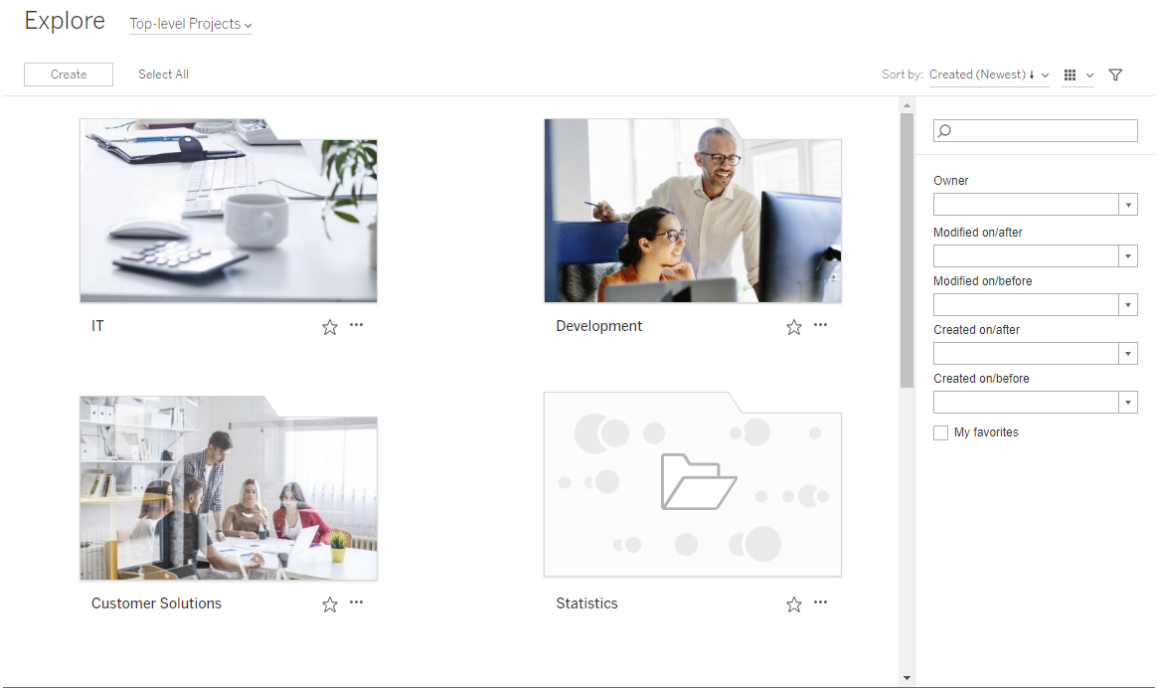
- コンテンツに対する**移動**パーミッション機能がなければなりません (データソースの場合、そのデータソースの所有者)。データベースとそのテーブルを移動する場合、ユーザーはデータベースとそのテーブルの両方に対する移動機能を持っている必要があります。

アセットの移動の詳細については、「コンテンツの移動」を参照してください。

プロジェクトイメージの追加

Tableau Server で管理するプロジェクトの区別 (およびユーザーによるプロジェクトの検索) に役立つよう、表示される画像をサムネイルに追加できます。イメージは次の要件を満たしている必要があります。

- イメージには HTTPS プロトコルを使用してアクセスできるようにする必要があります。共有ネットワークディレクトリおよび関連プロトコル (UNC、SMB、AFP、NFS など) はサポートされていません。プロジェクトイメージの HTTP プロトコルは、Google Chrome でサポートされていません。
- プロジェクトにアクセスするすべてのユーザーに、ターゲットイメージに対する「読み取り専用」以上のパーミッションが必要です。
- イメージは、一般的なインターネット形式 (.jpg、.png、または .gif) にする必要があります。

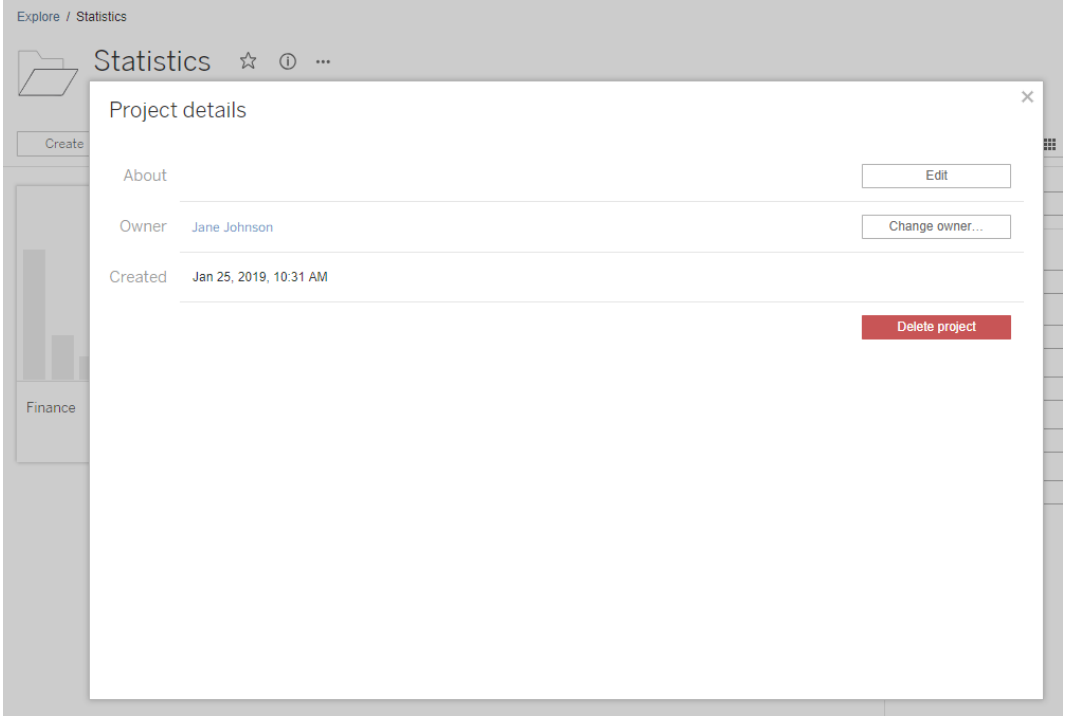


プロジェクトイメージの設定

1. Tableau Server でサイトへサインインします。アクセス権がある **[最上位のプロジェクト]** のリストで、更新するプロジェクトを選択するか、これに移動します。この例では、"Statistics (統計)" プロジェクトフォルダーにイメージを追加します。

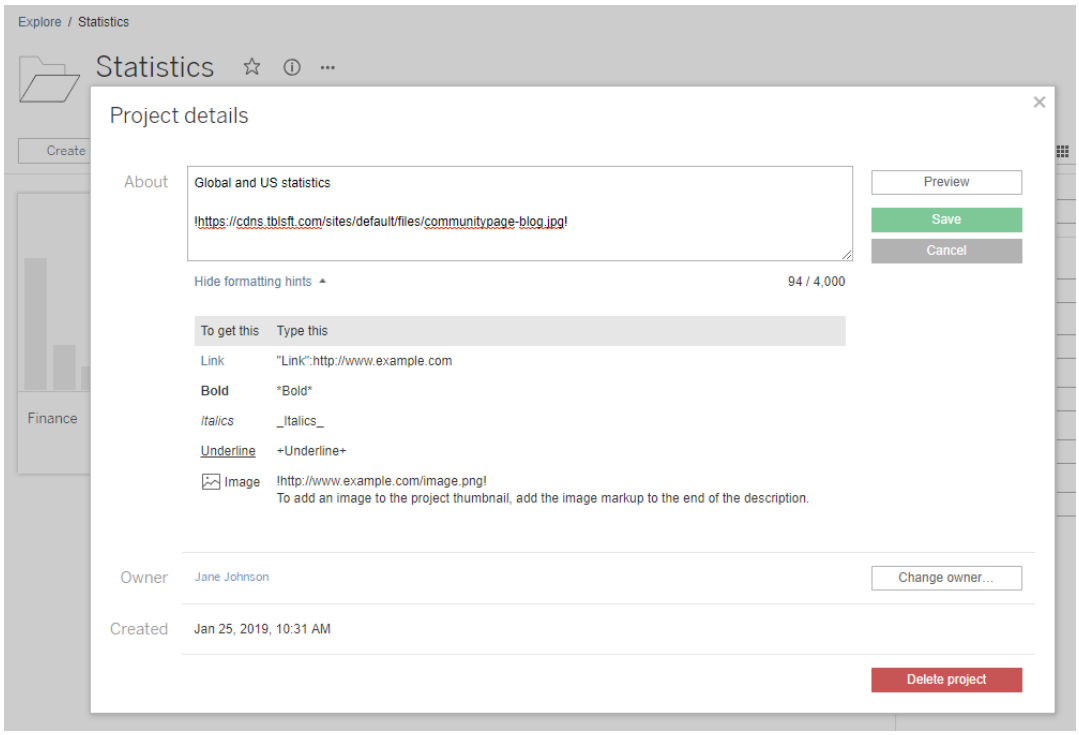
   子プロジェクトをどこで見つければよいかわからない場合は、**[検索]** ドロップダウンリストを使用し、**[すべてのプロジェクト]** を選択します。

2. **[詳細]** アイコン (i) をクリックし、**[プロジェクトの詳細]** ダイアログボックスを開いてから、**[編集]** をクリックします。



3. **[詳細]** フィールドには「Global and US statistics (世界規模および米国の統計)」など、プロジェクトの説明を入力することができます (オプション)。プロジェクトの説明の末尾で、次の構文を使用してイメージの URL を追加します。

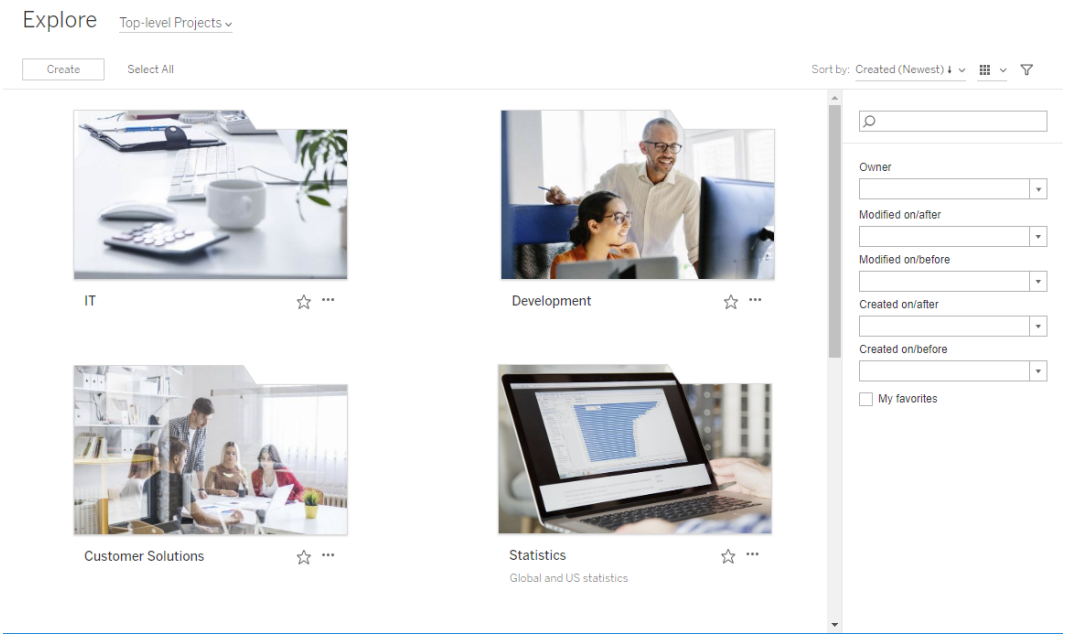
```
!http://www.example.com/image.png!
```



[書式設定のヒントを表示] を選択して、説明文を書式設定できる方法を確認します。

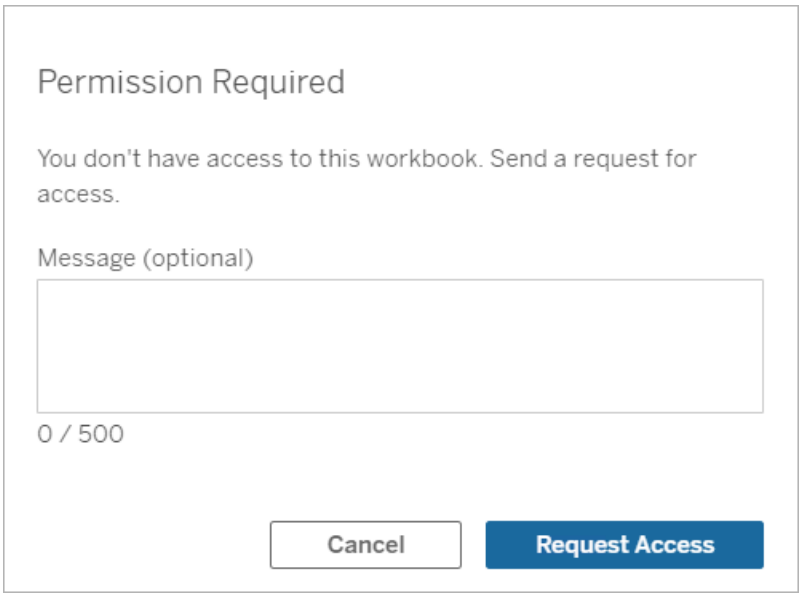
> **注:** プロジェクトの説明に埋め込まれたイメージはサイズ変更や配置を行うことができません。推奨サイズは 300 x 184 ピクセルです。300 x 184 ピクセルではないイメージは、サムネイルの幅に合うように拡大、縮小、またはトリミングできます。さらに、プロジェクトの説明の末尾に追加し、**!** (感嘆符) で囲む必要があります。そのようにしないと、サムネイルとして表示されません。

4. **[保存]** をクリックします。



サイトユーザーにコンテンツへのアクセスを要求できるようにする

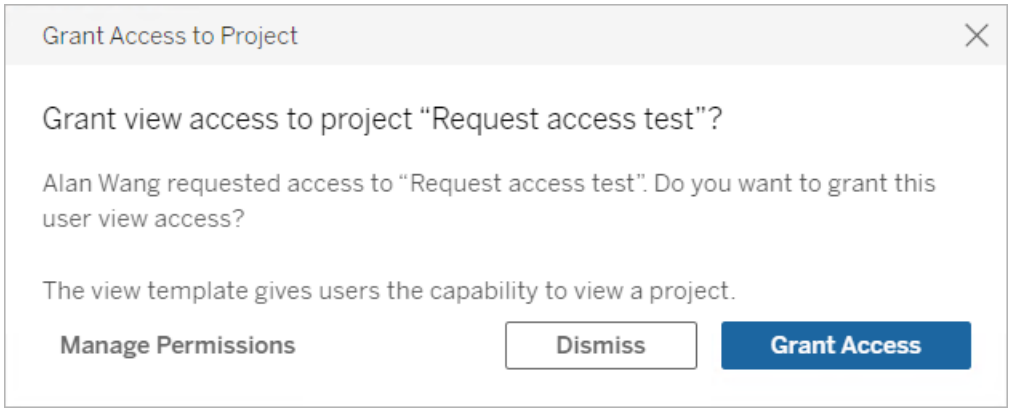
パーミッションによって、ユーザーがワークブック、ビュー、またはプロジェクト内の他のコンテンツにアクセスして表示できるかどうかが決まります。ユーザーがアクセス権を持っていないコンテンツやプロジェクトをクリックした場合、そのコンテンツのパーミッションを管理する所有者にアクセス要求を送信することができます。



誰かがアクセスを要求すると、そのコンテンツのパーミッションを管理する所有者 (プロジェクトレベルまたはワークブックレベル) は、要求者の名前とメール アドレス、要求されたコンテンツまたはプロジェクト、およびコンテンツへのアクセスを許可するリンクが記載されたメールを受信します。Tableau Server バージョン 2022.3 以前では、所有者はアクセス権を直接付与するリンクではなく、パーミッションを管理するためのコンテンツへのリンクを受信します。

1. メール通知で、**[Grant Access (アクセス権の付与)]** を選択します。
2. 表示されるダイアログで、ビューのパーミッション テンプレートを付与するには、**[Grant Access (アクセス権の付与)]** を選択します。ビューのテンプレート以外のアクセス権を付与

するには、**[パーミッションの管理]** を選択します。



ユーザーがワークブックへのアクセス権を要求し、コンテンツ パーミッションがプロジェクトにロックされている場合、プロジェクトの所有者がその要求を受信します。同様に、ユーザーがワークブックへのアクセスを要求し、プロジェクトのパーミッションがワークブックの所有者によって管理されている場合、ワークブックの所有者はその要求を受け取ります。

パーミッションを付与したら、所有者は要求者にプロジェクトやワークブックを表示する機能が付与されたことをメールで通知できます。

既定の設定

新しいサイトでは、アクセス要求の設定が既定で有効になっています。無効になっている場合に、設定を有効にするには:

1. サイトの [設定] ページの [全般] タブに移動します。
   - サイトが 1 つである場合は、サイドナビゲーションで **[設定]** をクリックし、**[全般]** をクリックします。
   - 複数のサイトがある場合は、構成するサイトを選択してから、**[設定]** をクリックし、**[全般]** をクリックします。
2. [全般] タブで、[アクセスの要求] までスクロールダウンし、**[ユーザーがプロジェクト、ワークブック、およびビューへのアクセスを要求できるようにする]** を選択します。
3. **[保存]** をクリックします。

プロジェクト パーミッションの構成

プロジェクトのコンテンツ パーミッションを調整することで、誰がアクセス要求を受信するかを制御することができます。コンテンツのパーミッションが次の場合:

- プロジェクトにロック: プロジェクトの所有者が要求を受け取ります。
- 所有者によって管理: ワークブックの所有者は、要求を受け取ります。

プロジェクトを使用してコンテンツアクセスを管理するには、プロジェクトを使用したコンテンツへのアクセスの管理 および パーミッション を参照してください。

パーミッション ルールの評価方法の詳細については、パーミッション: パーミッション ルールの評価」を参照してください。

プロジェクト パーミッションの変更

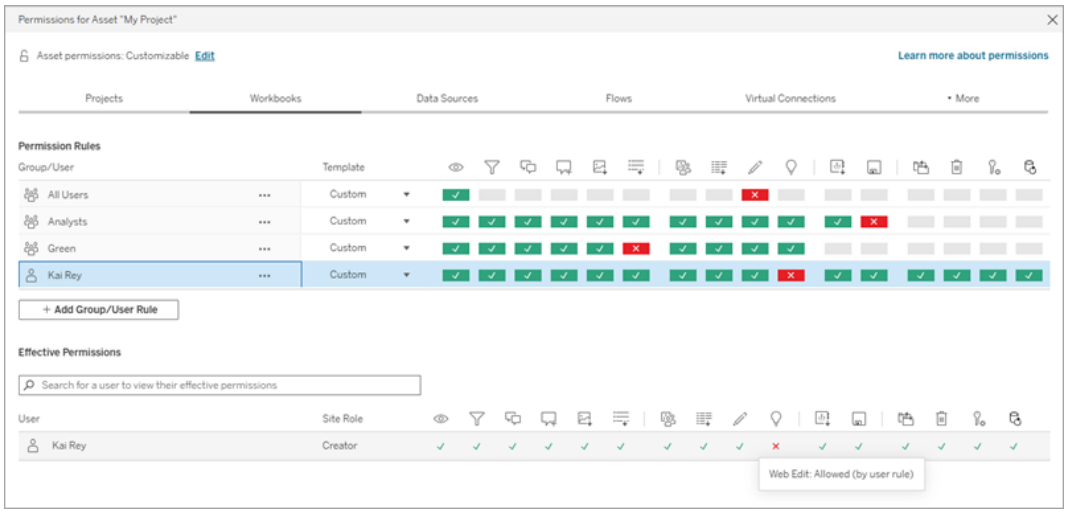
管理者、プロジェクト リーダーの場合

プロジェクト自体とそのプロジェクト内のコンテンツの両方で、パーミッションをプロジェクト レベルで設定できます。たとえば、プロジェクト レベルでワークブック パーミッションを構成すると、そのプロジェクトにパブリッシュされたすべてのワークブックはそれらの既定のパーミッションを継承します。ただし、Creator がパブリッシュ時にパーミッションを変更することも、特定のユーザーがパブリッシュ済みコンテンツに対するパーミッションを変更することもできます。プロジェクト レベルで設定されたパーミッションを実行するために、**コンテンツ パーミッション**をプロジェクトにロックできます。詳細については、「アセット パーミッションのロック」を参照してください。

プロジェクト レベルでパーミッションを設定するには:

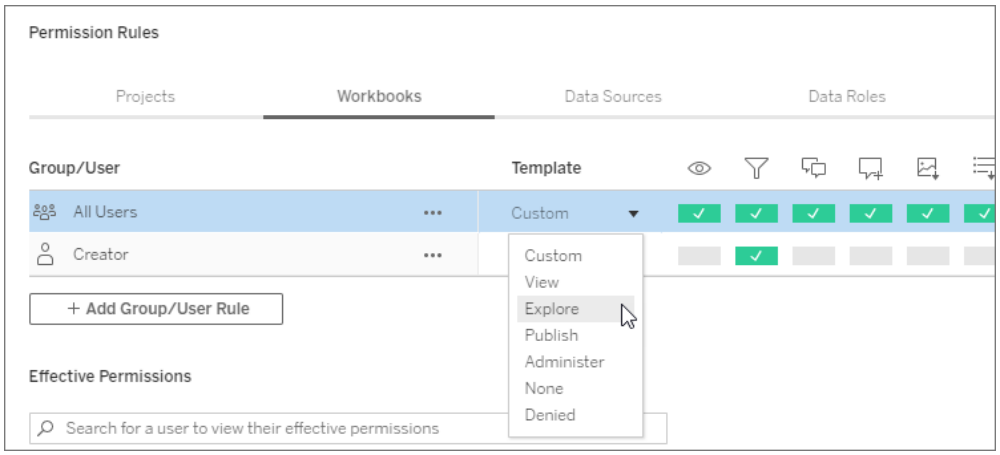
1. プロジェクトに移動します。
2. [アクション] メニュー (...) を開いて **[パーミッション]** をクリックします。[パーミッション] ダイアログ ボックスが開きます。

   このダイアログ ボックスには、上部のパーミッション ルールと下部の有効なパーミッション グリッドの 2 つの主要な領域があります。タブを使用して、コンテンツのタイプ間を移動します。



上部で行を選択すると、有効なパーミッション グリッドへの入力が行われます。これを使用してパーミッションを確認します。機能を表す指標にカーソルを合わせると、その特定のユーザに対して機能が許可または拒否される理由が表示されます。

3. 既存のパーミッション ルールを変更するには、ルールを選択し、機能ボックスをクリックして、許可/拒否/未指定を切り替えます。
4. 新しいルールを作成するには、
   a. **[+ グループ/ユーザー ルールの追加]** を選択します。
   b. ドロップダウン ボックスからグループまたはユーザーを選択します。これにより、パーミッション ルールを構成できる行が作成されます。
5. パーミッション ルールの行
   a. 各コンテンツ タイプ タブのドロップダウン ボックスから既存のパーミッション ロールのテンプレートを選択します。



b. または、[コンテンツ タイプ] タブに移動して機能をクリックし、カスタム ルールを作成します。機能は 1 回クリックすると**[許可]**、2 回クリックすると**[拒否]** に設定され、3 回目のクリックでは選択が解除されます (**[指定されていません]**)。

6. 完了したら、**[保存]** をクリックします。

コンテンツのパーミッションの変更

管理者、プロジェクト リーダー、コンテンツ所有者の場合

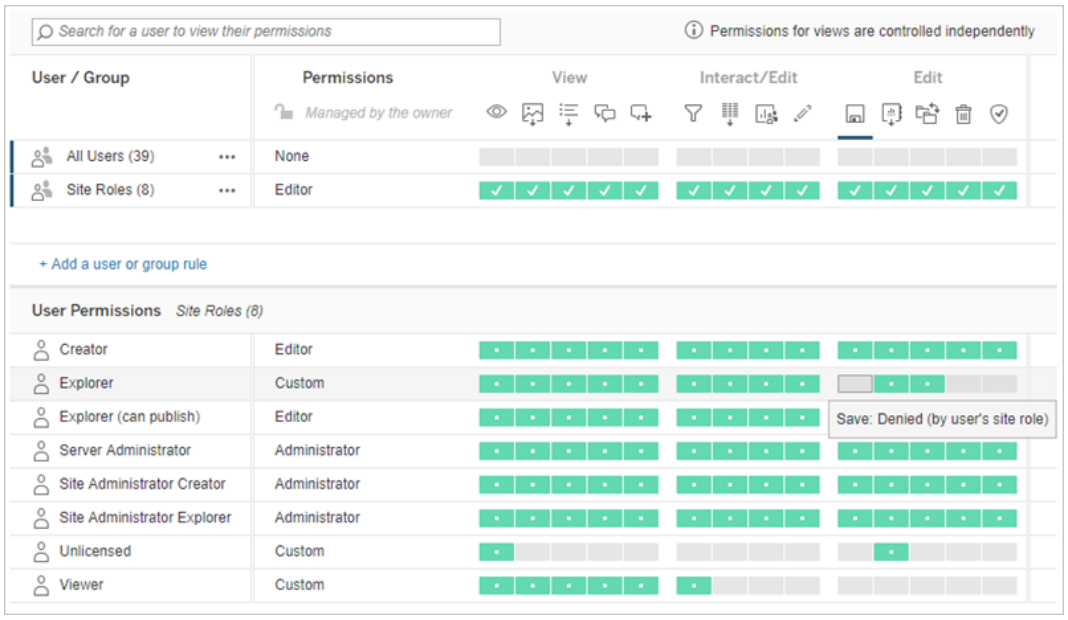
プロジェクトのパーミッションがロックされていない場合、個々のコンテンツのパーミッションを変更できます。

警告: Tableau では、Tableau サイト内のプロジェクト レベルでパーミッションを管理することを推奨しています。これらの手順は、所有者がパーミッションを管理しているプロジェクトのコンテンツにのみ関連します。

コンテンツのパーミッションの設定

1. コンテンツ (ワークブック、データ ソース、フロー、データの役割) に移動します。
2. [アクション] メニュー (...) を開いて **[パーミッション]** をクリックします。[パーミッション] ダイアログ ボックスが開きます。

   このダイアログ ボックスには、上部のパーミッション ルールと下部の有効なパーミッション グリッドの 2 つの主要な領域があります。



上部で行を選択すると、有効なパーミッション グリッドへの入力が行われます。これを使用してパーミッションを確認します。機能を表す四角にマウス ポインターを置くと、特定のユーザーでその機能が許可または拒否されている理由についての情報が表示されます。

3. 既存のパーミッション ルールを変更するには、その行の [アクション] メニュー (..) を開いて **[編集]** をクリックします。
4. 新しいルールを作成するには、
   a. **[+ ユーザーまたはグループ ルールの追加]** を選択します。
   b. 必要に応じて、右側のドロップダウン ボックスを使用して、グループとユーザーを切り替えます。
   c. ドロップダウン ボックスからグループまたはユーザーを選択します。これにより、パーミッション ルールを構成できる行が作成されます。
5. パーミッション ルールの行で、ドロップダウン ボックスから既存のパーミッション ロール テンプレートを選択するか、機能をクリックしてカスタム ルールを作成します。

   機能は 1 回クリックすると **[許可]**、2 回クリックすると **[拒否]** に設定され、3 回目のクリックでは選択が解除されます (**[指定されていません]**)。
6. 完了したら、**[保存]** をクリックします。

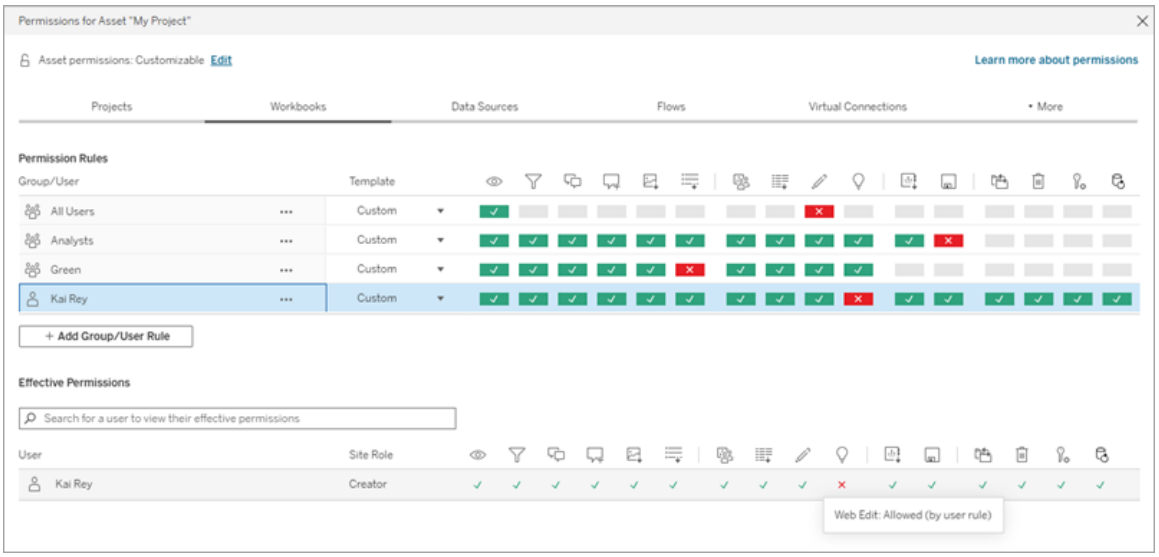
ビューのパーミッションの設定

状況によっては、ビューを含むワークブックとは別にビューのパーミッションを指定することが有益な場合があります。パブリッシュされたビューにパーミッションを設定するには、パブリッシュされたワークブック内のビューに移動し、上記の手順を行います。

警告: ワークブック内でビュー レベルのパーミッションを設定することは可能ですが、できる限りプロジェクト(またはワークブック) レベルでパーミッションを管理することを強くお勧めします。ビューがパーミッションを継承するには、プロジェクトをロックするか、**[シートをタブとして表示]** でワークブックをパブリッシュする必要があります。詳細については、サイト ユーザーにコンテンツへのアクセスを要求できるようにするを参照してください。

パーミッション

パーミッションは、ユーザーがワークブックやデータ ソースなどのコンテンツを操作する方法を決定します。パーミッションは、パーミッション ダイアログ内で設定するか、または REST API から設定します。ダイアログの上部で、パーミッション ルールはグループまたはユーザーの機能を構成します。下部のパーミッション グリッドには、ユーザーの有効なパーミッションが表示されます。



パーミッションの検討、設定、および管理の方法について説明している、相互に関連するトピックがいくつかあります。主なトピックは次のとおりです。

- このトピック: 基本事項、プロジェクトやその他のコンテンツに対してパーミッション ルールを設定する方法、および特定のシナリオに対するパーミッションの考慮事項について説明します。
- パーミッションの機能とテンプレート: パーミッション ルールの作成に使用されるさまざまな機能を詳細に説明します。
- プロジェクトを使用したパーミッションの管理: プロジェクトを使用したパーミッションの管理と、ネストされたプロジェクトやロックされたプロジェクトがどのようにパーミッションに影響を与えるかについて説明します。
- 有効なパーミッション: パーミッション ルールの評価方法と最終的なパーミッションの決定方法を説明します。
- パーミッション、サイト ロール、ライセンス: サイトでユーザーが行える操作を決定するために、パーミッションがどのようにサイト ロールおよびライセンスと相互作用するかについて説明します。

また、データ管理がライセンスされている場合、外部アセットに対するパーミッションには追加の考慮事項があります。詳細については、「外部資産でのパーミッションの管理」を参照してください。

パーミッションの基本

プロジェクトとグループ

Tableau サイトは、コンテンツの整理に*プロジェクト*を、ユーザーの整理に*グループ*を使用します。パーミッション ルールが次の場合は、パーミッションの管理が容易になります。

- 個々のコンテンツではなく、プロジェクト レベルで設定します。
- 個人ではなくグループに対して設定されます。

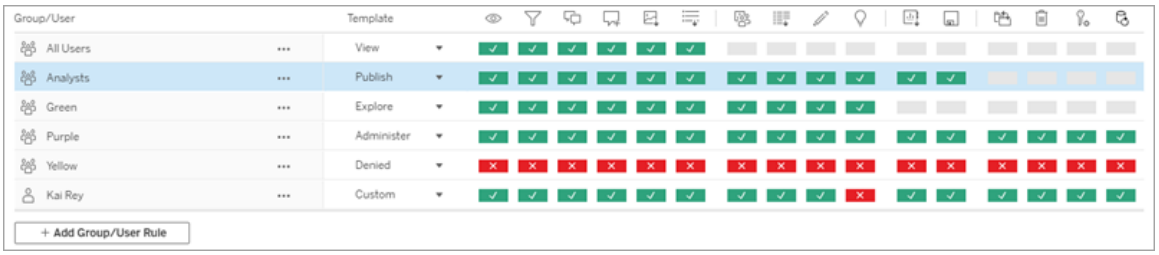
パーミッションは、既に存在するユーザー、グループ、プロジェクト、またはアセットに対してのみ設定できます。ユーザーおよびグループの作成、プロジェクトの作成、コンテンツのパブリッシュの詳細については、「ユーザーとグループの管理」、「プロジェクトを使用したコンテンツへのアクセスの管理」、「データ ソースとワークブックのパブリッシュ」を参照してください。

機能とパーミッションのルール

パーミッションは、コンテンツの表示、Web 編集、データ ソースのダウンロード、コンテンツの削除といった操作を実行できる*機能*で構成されます。*パーミッション ルール*は、アセットに対し、どの機能がユーザーやグループに許可または拒否されるかを決定します。

機能とパーミッション ルールのテンプレートの詳細については、「パーミッションの機能とテンプレート」を参照してください。

注: 一般にパーミッションについて語るとき、「ユーザーには削除*権限*が必要」という表現をよく見かけます。これは幅広いコンテキストで理解しやすい文言です。ですが、この記事のように、技術的なレベルでパーミッションを扱う場合は、「削除*機能*」と言う方が正確です。このトピックでは、より正確な用語である*機能*を使用しますが、他の箇所では*パーミッション*と書かれている場合があることに注意してください。



機能アイコンとその意味については、「パーミッションの機能とテンプレート」を参照してください。

ライセンス レベル、サイト ロール、および複数のパーミッション ルール間での相互作用は、ユーザーが実行できる操作とできない操作の最終的な決定に関係します。各ユーザーに対して、これが*有効なパーミッション*になります。詳細については、「有効なパーミッション」を参照してください。ブラウザーから新しいワークブックを作成したり(Web 作成)、コンテンツを移動したりするなどの一部のタスクは、1 つの機能に包括されるとは限らず、複数の機能を具体的に構成することが必要な場合があります。詳細については、「特定のシナリオでのパーミッション設定」を参照してください。

パーミッションの設定

パーミッション ルールは、プロジェクト レベルとコンテンツ レベルで、または Tableau Desktop からコンテンツをパブリッシュする場合、設定がそれぞれ異なります。

注: 「プロジェクトのパーミッション」という語句には、2 つの意味があります。プロジェクト自体のパーミッション機能 (ビューとパブリッシュ) があり、これはユーザーがプロジェクトと対話する方法

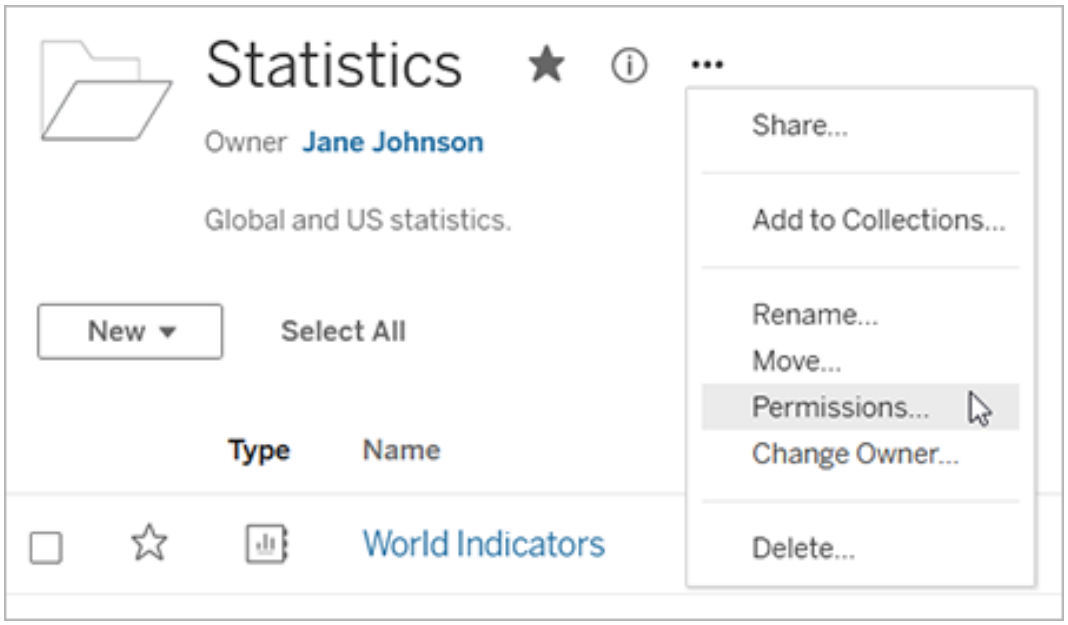
を制御します。また、他のコンテンツ タイプに対するプロジェクト レベルのパーミッション ルールの概念もあります。この記事の「プロジェクト レベルのパーミッション」とは、プロジェクトのパーミッション ダイアログで構成されたワークブック、データ ソース、およびその他のアセットに対するパーミッション ルールを意味します。これは、特定のワークブックやデータ ソースなどに設定できる「コンテンツ レベル」のパーミッション ルールとは対照的です。

プロジェクト レベルのパーミッション

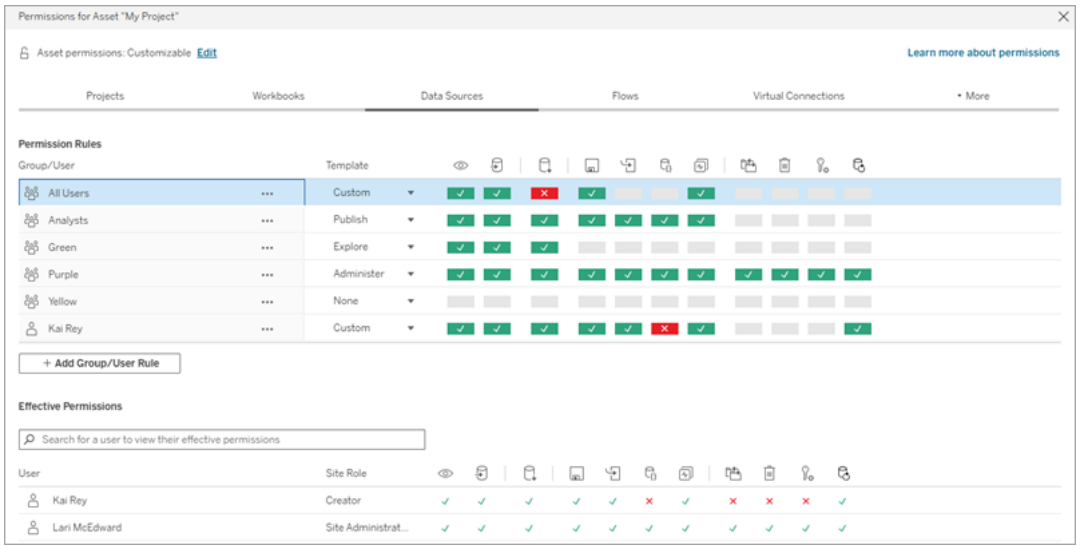
管理者、プロジェクト所有者およびプロジェクト リーダーの場合

プロジェクト レベルでパーミッションを設定するには:

1. プロジェクトに移動します。
2. [アクション] メニュー (...) を開いて **[パーミッション]** をクリックします。



[パーミッション] ダイアログが開きます。このダイアログには、上部のパーミッション ルールと下部の有効なパーミッション グリッドの 2 つの主要な領域があります。各コンテンツ タイプにはタブがあります。下の図は、[データ ソース] タブを示しています。



上部で行を選択すると、有効なパーミッション グリッドへの入力が行われます。これを使用してパーミッションを確認します。カーソルを合わせると、特定のユーザーで機能が許可または拒否されている理由についての情報が得られます。

3. 既存のパーミッション ルールを変更するには、そのコンテンツ タイプに適したタブを選択し、機能をクリックします。
4. ルールを作成するには、**[+ グループ/ユーザー ルールの追加]** をクリックし、入力を開始してグループまたはユーザーを検索します。各タブで、ドロップダウン ボックスから既存のテンプレートを選択するか、機能をクリックして、カスタム ルールを作成します。

   機能は 1 回クリックすると**[許可]**、2 回クリックすると**[拒否]** に設定され、3 回目のクリックでは選択が解除されます (**[指定されていません]**)。

5. 完了したら、**[保存]** をクリックします。
   - [なし] というテンプレートが選択されている場合、ボタンには [Delete Rule (ルールの削除)] と表示されます。

すべてのコンテンツ タイプにプロジェクト パーミッションを設定する

プロジェクトのパーミッション ダイアログには、コンテンツのタイプごとにタブが含まれています。**プロジェクト レベルでコンテンツのタイプごとにパーミッションを設定する必要があります。設定しない場合、ユーザーはそのコンテンツ タイプへのアクセスを拒否されます。**機能は明示的に許可された場合にのみユーザーに付与されます。機能を [未指定] にすると、その機能は拒否されます。

ヒント: プロジェクト レベルでパーミッション ルールを作成するたびに、すべてのコンテンツ タイプ タブを確認してください。

アセットのパーミッションの設定を構成する

プロジェクト レベルで設定されたパーミッション ルールは、そのプロジェクトに保存されたコンテンツと、そのプロジェクトに含まれるネストされたプロジェクトの既定のルールとして機能します。これらのプロジェクト レベルの既定のルールを統一するか、または編集可能かどうかは、**アセットのパーミッション**の設定によって決まります。この設定は、**[ロック]** または **[カスタマイズ可能]** の 2 つの方法のいずれかで構成できます。詳細については、「アセット パーミッションのロック」を参照してください。

コンテンツ レベルのパーミッション

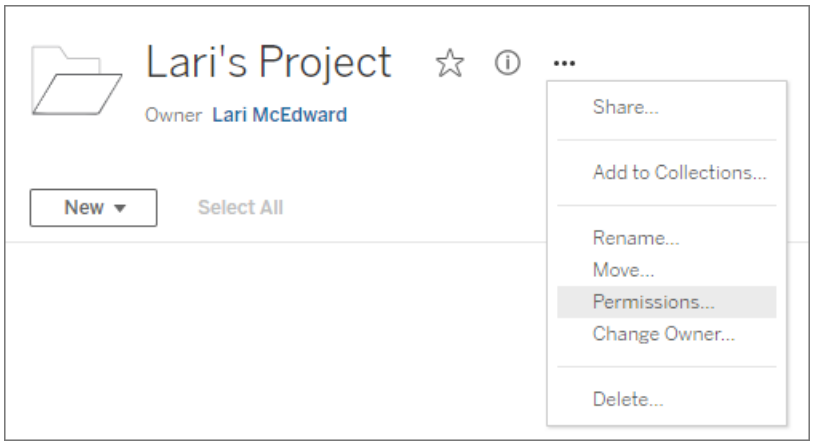
管理者、プロジェクト リーダー、コンテンツ所有者の場合

プロジェクトの**アセットのパーミッション**が**カスタマイズ可能**な場合、個々のアセットのパーミッションを変更することができます。以下の情報は、ロックされたプロジェクトのアセットには関係ありません。詳細については、「アセット パーミッションのロック」を参照してください。

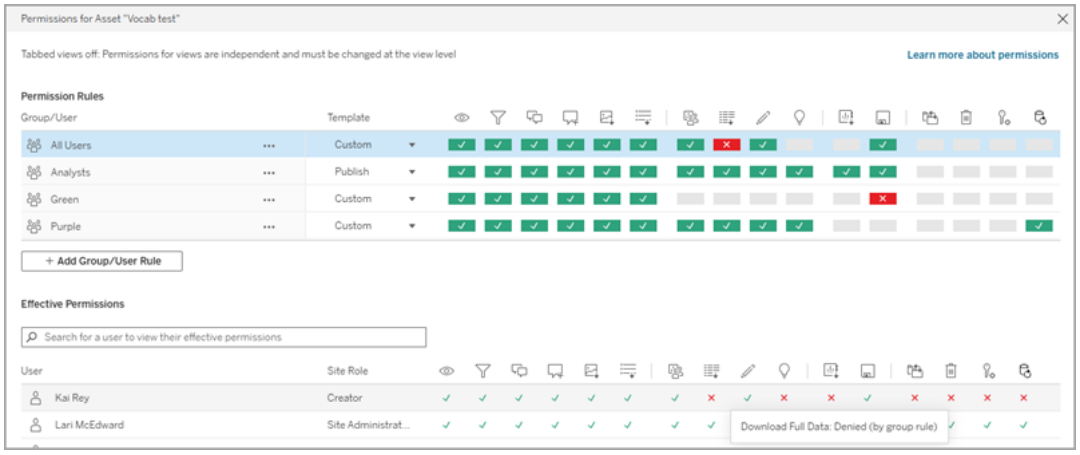
ヒント: **カスタマイズ可能**なプロジェクトの個々のアセットに対するパーミッションを設定することは可能ですが、プロジェクト レベルでパーミッションを管理することを推奨します。

アセットのパーミッションの設定

1. アセット(ワークブック、データソース、またはフローなど)に移動します
2. [アクション]メニュー(...)を開いて**[パーミッション]**をクリックします。



[パーミッション]ダイアログが開きます。このダイアログには、上部のパーミッションルールと下部の有効なパーミッショングリッドの2つの主要な領域があります。(上部にタブがないことに注意してください。アセットレベルでのパーミッションダイアログにはタブがありません)。



上部で行を選択すると、有効なパーミッショングリッドへの入力が行われます。これを使用してパーミッションを確認します。機能を表す四角にマウスポインターを置くと、特定のユーザーでその機能が許可または拒否されている理由についての情報が表示されます。

3. 既存のパーミッションルールを変更するには、機能をクリックします。

4. ルールを作成するには、**[+ グループ/ユーザールールの追加]** をクリックし、入力を開始してグループまたはユーザーを検索します。ドロップダウンから既存のテンプレートを選択するか、機能をクリックしてカスタム ルールを作成します。

   機能は 1 回クリックすると**[許可]**、2 回クリックすると**[拒否]** に設定され、3 回目のクリックでは選択が解除されます (**[指定されていません]**)。

5. 完了したら、**[保存]** をクリックします。
   - [なし] というテンプレートが選択されている場合、ボタンには [Delete Rule (ルールの削除)] と表示されます。

ビューのパーミッションの設定

ヒント: ワークブック内でビュー レベルのパーミッションを設定することは可能ですが、プロジェクト (または必要に応じてワークブック) レベルでパーミッションを管理することを強くお勧めします。

ワークブックが **[シートをタブとして表示]** チェック ボックスをオンにしてパブリッシュされた場合、そのワークブックのビューは、そのワークブックに設定されているすべてのパーミッションを継承します。ビューのパーミッション ダイアログは読み取り専用になります。

状況によっては、ビューを含むワークブックとは別にビューのパーミッションを指定することが有益な場合があります。**[シートをタブとして表示]** をオフ (シート タブを非表示) にしてワークブックがパブリッシュされた場合、ビューはワークブックのパーミッションで開始されますが、その後は独立して設定することができます。これは、ワークブックのパーミッション ルールが変更された場合、それらの変更はビューに適用されず、各ビューのパーミッションを個別に管理する必要があることを意味する点に注意してください。

詳細については、「シート タブの表示または非表示」を参照してください。

パブリッシュのパーミッションの設定

コンテンツ パブリッシャーの場合

プロジェクトの**アセットのパーミッション**が**カスタマイズ可能**な場合は、Tableau Desktop からパブリッシュするときに個々のアセットのパーミッションを設定できます。以下の情報は、ロックされたプロジェ

クトのコンテンツには関係がありません。詳細については、「アセット パーミッションのロック」を参照してください。

ヒント: **カスタマイズ可能**なプロジェクトの個々のアセットに対するパーミッションを設定することは可能ですが、プロジェクト レベルでパーミッションを管理することを推奨します。

1. [パブリッシュ] ダイアログ ボックスで、**[パーミッション]** の [編集] リンクをクリックします。
   [編集] リンクが使用できない場合はパーミッションがプロジェクトにロックされており、プロジェクト所有者、プロジェクト リーダー、管理者を除き変更することができません。
2. [パーミッションの追加/編集] ダイアログには、既存のパーミッション ルールが表示されます。**[追加]** をクリックしてパーミッション ルールを追加するか、**[編集]** をクリックして既存のパーミッション ルールを変更します。
   a. 左側のペインからグループまたはユーザーを選択します。グループを展開すると、含まれるユーザーを確認できます。
   b. 右側のペインの上部にあるセレクターを使用して既存テンプレートを選択するか、ラジオ ボタンを使用してカスタム ルールを作成します。

   パブリッシュを行うダイアログから有効なパーミッションを調査することはできない点に注意してください。
3. 完了したら、**[OK]** をクリックして、パブリッシュを再開します。

> **注**: Tableau Prep Builder のフローをパブリッシュしているときにパーミッションを設定することはできません。フローに対するパーミッションを設定するには、「プロジェクト レベルのパーミッション」または「コンテンツ レベルのパーミッション」の手順を参照してください。

すべてのユーザー グループをクリーニングする

既定ではすべてのユーザーが、コンテンツに対して基本的なパーミッションを持つ「すべてのユーザー」グループに追加されます。独自のパーミッション ルールを構築するときに何もない状態から開始するため、「すべてのユーザー」のルールを完全に削除するか、または編集してすべてのパーミッションを削除することをお勧めします(パーミッション ロール テンプレートは [なし] に設定します) 。これにより、特定のユーザーに適用されるルールの数を減らし、今後あいまいさが生じないようにするのに役立ち、有効なパーミッションを理解しやすくなります。

特定のシナリオでのパーミッション設定

特定のアクションでは、パーミッション機能に加え、場合によってはサイトロールの組み合わせが必要になります。以下はいくつかの一般的なシナリオと必要なパーミッションの構成を示しています

保存、パブリッシュ、上書き

パーミッションでは、保存とは基本的にパブリッシュのことです。したがって、**[上書き]** および **[コピーを保存]** の機能は、パブリッシュを許可する、管理者、Creator、または Explorer (パブリッシュ可能) のサイトロールを持つユーザーにのみ付与できます。Explorer や Viewer サイトロールはパブリッシュ、上書き、またはコピーを保存を行うことができません。

- プロジェクトの **パブリッシュ** 機能を使用すると、ユーザーはそのプロジェクトにコンテンツをパブリッシュできます。
- コンテンツの **[上書き]** 機能を使用すると、ユーザーは既存のコンテンツを上書きして保存することができます。コンテンツを上書きして保存すると、ユーザーはそのコンテンツの所有者になります。また、ユーザーは [上書き] 機能を使用して、メトリクスの説明やデータの役割の同義語など、既存のコンテンツの軽微な側面を編集することができます。この方法で既存のコンテンツを編集しても、コンテンツの所有者は変更されません。
- コンテンツの **[ダウンロード/コピーを保存]** 機能または **[コピーを保存]** 機能を使用すると、ユーザーは **[名前をつけてパブリッシュ]** メニューオプションを使用してコンテンツの新しいコピーを保存することができます。これは通常、Web 作成と同時に行われ、ユーザーが変更を保存できることを意味しています。
    - ワークブックのコピーを保存することは、変更を下書きとして保持する自動保存とは異なります。

パブリッシュする場所

すべてのコンテンツはプロジェクトにパブリッシュされる必要があるため、少なくとも 1 つのプロジェクトで **[パブリッシュ]** 権限がない限り、ユーザーはコンテンツを [パブリッシュ] または [名前を付けてパブリッシュ] できないことに注意が必要です。プロジェクトレベルで **[パブリッシュ]** 機能がない場合、そのコンテンツをパブリッシュすることはできません。

上書きして所有権を要求する

Web 編集では、[ファイル] メニューの **[パブリッシュ]** オプションは、コンテンツ所有者にのみ表示されます。所有者ではないが **[上書き]** 権限がある (コンテンツを保存できる) ユーザーの場合は、**[ファ**

イル] の **[名前を付けて保存]** を使用して、ワークブックに全く同じ名前を付ける必要があります。既存コンテンツを上書きしようとしている旨の警告メッセージが表示されます (実際に上書きできます)。逆に、**[コピーを保存]** 機能しかないユーザーが同じ名前を使用しようとすると、既存コンテンツを上書きするパーミッションがない旨のエラーが表示されます。

そのコンテンツ所有者ではないユーザーがコンテンツを上書きする場合、そのユーザーは所有者になり、所有者に伴うすべてのパーミッションを持ちます。その後、コンテンツに対する元の所有者のアクセス権は、所有者ではなくユーザーとしてのパーミッションで決まります。

> **注**: **[ダウンロード/コピーを保存]** はワークブックに使用する結合機能です。Explorer はこの権限を持つことはできますが、**[名前を付けてパブリッシュ]** を使用したコピーの保存ではなく、ワークブックのダウンロードのみを行うことができます。この権限を Explorer (パブリッシュ可能)、Creator、または管理者のサイトロールに付与すると、ワークブックのダウンロードと、[ファイル] メニューのパブリッシュオプションを使用したコピーの保存の*両方*を行うことができます。

Web 編集と Web 作成

Web 編集と Web 作成の機能によって、ユーザーはブラウザーでワークブックを直接編集または作成することができます。

パーミッション機能は ***Web 編集***と呼ばれ、サイト設定は ***Web 作成***と呼ばれます。このセクションでは、Web ベースの編集操作やパブリッシュ操作のすべてを ***Web 作成***と呼びます。

この機能を有効にするためのいくつかの要件があります。

- **サイト設定**: Web 作成は Tableau サイト全体で有効にする必要があります。サイトの Web 作成アクセスの設定 を参照してください。この設定を有効にしなければ、どのユーザーも *Web 編集機能を持っていても*、ブラウザーでワークブックの作成や既存のワークブックの編集を行うことはできません。
- **ユーザーのサイトロール**: ユーザーは適切なサイトロールを持っている必要があります。
  - Viewer は Web 編集を行うことができません。
  - Explorer には Web 編集機能を付与できますが、パブリッシュは行えません。基本的に、Web 編集を使用し、既存コンテンツに基づいて疑問に対する答えをその場で見つけることはできますが、編集内容を保存することはできません。

  - Explorer (パブリッシュ可能) またはサイト管理者 Explorer はパブリッシュできますが、サイトに既にパブリッシュ済みのデータを使用することしかできません。
  - Creator、サイト管理者 Creator、サーバー管理者は、データソースをパブリッシュして作成できます。
- **パーミッション機能**: ユーザーは必要な機能に基づき、必要なパーミッション機能を持っている必要があります。

必要なパーミッションの機能設定

必要な機能	最小限のサイトロール	Web 編集	ダウンロード/コピーを保存	上書き (ワークブック)	パブリッシュ (プロジェクト)	接続 (データソース)
保存を行えない Web 作成	*Explorer*	許可	拒否	拒否	オプション	許可
Web 作成と新しいコンテンツとして保存	*Explorer (パブリッシュ可能)*	許可	許可	拒否	許可	許可
Web 作成とコンテンツの保存 (上書き)	*Explorer (パブリッシュ可能)*	許可	許可	許可	許可	許可
新しいデータを使用した Web 作成と新しいコンテンツの保存	*Creator*	許可	オプション	オプション	許可	オプション

オプションは、この機能が必要な機能とは関係ないことを示しています

パブリッシュ済み Tableau データソースでのデータアクセス

Tableau サイトにパブリッシュされたデータソースには、Tableau 環境内にパーミッションだけでなく、ネイティブ認証を含めることができます。

データソースを Tableau サイトにパブリッシュする場合、パブリッシャーは パブリッシュされたデータにアクセスするための認証資格情報を設定する方法 を選択することができます。この記事では、

データソースの認証資格情報を取り扱う方法について記載しています (データベースへのログインや Google スプレッドシートでの認証資格情報の入力をユーザーに求めるなど)。この認証は、どのテクノロジーによりデータを保持するかによって制御されます。データソースがパブリッシュされるときに、これを埋め込むことができます。またはデータソースのパブリッシャーがユーザーにデータソースの認証資格情報を求めるメッセージを表示するようにすることもできます。詳細については、「データソースのパブリッシュ」を参照してください。

また、Tableau でユーザーに表示 (**ビュー**) や、パブリッシュ済みデータソースへの接続 (**接続**) を許可または拒否するデータソース機能もあります。これらの機能は、Tableau の他のパーミッションと同様に設定されます。

パブリッシュ済みデータソースを使用するワークブックがパブリッシュされると、作成者は、ワークブックを使用するユーザーで Tableau 認証の動作方法を制御することができます。作成者はパブリッシュ済みデータソースに対するワークブックのアクセス権を **[Embed password (パスワードの埋め込み)]** (作成者のデータソースに対する接続アクセス権を使用) または **[Prompt users (ユーザーにメッセージを表示)]** (ワークブックを表示しているユーザーの接続アクセス権を使用) のいずれかに設定します。データソース認証も必要となる場合があります。

- ワークブックが **[パスワードの埋め込み]** に設定されている場合、ワークブックを閲覧するすべてのユーザーには、データソースに対する作成者のアクセス権に基づいてデータが表示されます。
- ワークブックが **[Prompt users (ユーザーにメッセージを表示)]** に設定されている場合、Tableau が制御するアクセス権がデータソースで確認されます。ワークブックを使用するユーザーがデータを表示するには、パブリッシュ済みデータソースで接続機能を持つ必要があります。パブリッシュ済みデータソースも [Prompt user (ユーザーにメッセージを表示)] に設定されている場合、ビューアーはデータソース自体の認証資格情報も入力する必要があります。

データソースに対するワークブックの認証	データに対するデータソースの認証	ワークブックを使用するユーザーでデータアクセスを評価する方法
パスワードの埋め込み	パスワードの埋め込み	ユーザーがワークブックの作成者であるかのようにデータが表示される
パスワードの埋	ユーザーに	ユーザーがワークブックの作成者であるかのようにデータが表

め込み	メッセージを表示	示される(ユーザーではなく、作成者にデータソース認証でメッセージが表示される)
ユーザーにメッセージを表示	パスワードの埋め込み	ユーザーがパブリッシュ済みデータソースに対して独自の**接続**機能を持っている必要がある
ユーザーにメッセージを表示	ユーザーにメッセージを表示	ユーザーがパブリッシュ済みデータソースに対して独自の**接続**機能を持っている必要があり、参照元データの認証資格情報を求めるメッセージが表示される

これは Web 編集ではなく、ワークブックの使用に適用されます。Web 編集を行うには、ユーザーが独自の接続機能を持っている必要があります。

仮想接続を使用するデータソースやワークブックなどの Tableau コンテンツをパブリッシュするときにパスワードを埋め込む方法については、Tableau Server ヘルプの「仮想接続」を参照してください。

他のユーザーによる抽出更新を許可する

Explorer (パブリッシュ可能) 以上のサイトロールを持つユーザーには、ワークブックまたはデータソースの抽出更新機能が付与されます。この機能により、ユーザーはアセットの所有者と同じインターフェイスを使用して抽出更新をトリガーできます。抽出はパブリッシュされたデータソースであるか、ワークブックに対してローカルである場合は認証資格情報が埋め込まれている必要があります。

抽出更新は大量のリソースを消費し、すでにスケジュールされている可能性があるため、抽出更新機能を割り当てる際には次のベストプラクティスを考慮してください。

- **使用頻度の低い抽出**: オンデマンドで抽出更新する権限をユーザーに付与し、冗長または不要なスケジュールされた更新をオフにすることを検討してください。
- **定期的に使用する抽出**: 抽出更新をスケジュールし、オンデマンド更新の抽出更新機能を制限することを検討してください。
    - 抽出更新のパーミッションを広範に付与するのではなく、アセットの所有者や管理者に依存するボトルネックを回避するために、抽出更新機能をバックアップとして割り当てます。

コンテンツの移動

アイテムを移動するには、[アクション] メニュー (...) を開き、**[移動]** をクリックします。アイテムで新しいプロジェクトを選択し、**[アセットを移動]** をクリックします。**移動**が使用できない場合、または使用可能な移動先プロジェクトがない場合は、適切な条件を満たしていることを確認します。

- 管理者はいつでもアセットやプロジェクトを任意の場所に移動できます。
- プロジェクトリーダーとプロジェクト所有者は、アセットとネストされたプロジェクトをプロジェクト間で移動できます。
  - 管理者以外は、プロジェクトを上位プロジェクトになるように移動できないことに注意してください。
- 他のユーザーは、次の 3 つの要件がすべて満たされている場合にのみアセットを移動できます。
  - Creator または Explorer (パブリッシュ可能) のサイトロール。
  - 移動先プロジェクトのパブリッシュ権限 (**[ビュー]** および **[パブリッシュ]** 機能)。
  - コンテンツの所有者である、またはワークブックやフローの場合には **[移動]** 機能がある。

データベースとそのテーブルを移動する場合、ユーザーはデータベースとそのテーブルの*両方*に対する**移動**機能を持っている必要があります。

コンテンツやプロジェクトを移動するときのパーミッションの処理方法については、「プロジェクトとコンテンツの移動」を参照してください。

メトリクス (2024.2 で廃止)

ヒント: このセクションでは、従来のメトリクス機能について説明しています。この機能は、Tableau Cloud では 2024 年 2 月に廃止され、Tableau Server ではバージョン 2024.2 で廃止されました。Tableau Pulse のメトリクスに対するパーミッションの詳細については、「Tableau Pulse 用にサイトをセットアップする」を参照してください。

従来のメトリクス機能の廃止

Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエク

スペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

メトリクスは、パブリッシュされたワークブックのビューから作成されます。ユーザーは、次の場合にメトリクスを作成できます。

- Creator または Explorer (パブリッシュ可能) のサイトロールを持つ
- プロジェクトで **[パブリッシュ]** 機能を持つ
- 関連するワークブックの **[Create/Refresh Metric (メトリクスの作成/更新)]** 機能を持つ

詳細については、「メトリクスの作成とトラブルシューティング (廃止)」と「メトリクスのセットアップ」を参照してください。

注: 2021.3 よりも前のバージョンでは、ビューでメトリクスを作成する機能は、[すべてのデータのダウンロード] 機能によって制御されていました。

メトリクスは独立したアセットであるため、メトリクスのパーミッションは、作成元のビューとは独立して管理されることに注意してください(これは、データドリブン アラートやサブスクリプションの場合とは異なります。アラートやサブスクリプションのコンテンツは、ユーザーがビュー自体の正しいパーミッションを持っている場合にのみ表示されます)。

メトリクスの機能は単純ですが、**ビュー**機能については、慎重に検討する必要があります。パーミッションが制限されているワークブックは、よりオープンなパーミッションが設定されるメトリクスの基準になる可能性があります。機密データを保護するために、特定のワークブックに対してメトリクスを作成できないようにすることもできます。

メトリクスは所有者のパースペクティブからデータを表示

メトリクスを作成するときは、そのビューからデータのパースペクティブをキャプチャします。これは、メトリクスにアクセスできるすべてのユーザーに、表示されたとおりのデータが表示されることを意味します。ビュー内のデータが認証資格情報に基づいてフィルターされている場合、表示されるデータは、他のユーザーが同じビューにアクセスしたときに表示されるデータとは異なる場合があります。データのパースペクティブを公開することに懸念がある場合は、メトリクスの**表示**機能を制限してください。

「データの説明を見る」

「データの説明を見る」が使用可能な場合、ユーザーはビューでマークを選択し、マークのツールヒントメニューで [Run Explain Data (「データの説明を見る」を実行)] をクリックできます。「データの説明を見る」を編集モードと表示モードで使用できるようにするには、設定の組み合わせを有効にする必要があります。

作成者が編集モードで「データの説明を見る」を実行したり、「データの説明を見る」の設定を編集したりするための要件は次のとおりです。

- サイト設定: **[「データの説明を見る」の可用性]** が **[有効]** に設定されている。デフォルトで有効に設定されています。
- サイトロール: Creator または Explorer (パブリッシュ可能)。
- パーミッション: **[Run Explain Data (「データの説明を見る」を実行)]** が **[許可]** に設定されている。既定では指定されていません。このパーミッションを使用したワークブック (Tableau バージョン 2022.1 以前) を Tableau バージョン 2022.2 以降で開く場合は、[「データの説明を見る」を実行] を [許可] にリセットする必要があります。

注: Creator または Explorer (パブリッシュ可能) の **[すべてのデータのダウンロード]** 機能は、極値の説明の [フルデータの表示] オプションを表示するかどうかを制御します。Viewer の場合、[すべてのデータのダウンロード] 機能が常に拒否されます。ただし、「データの説明を見る」の設定で極値の説明のタイプが有効になっている場合は、すべてのユーザーがレコードレベルの詳細を表示できます。

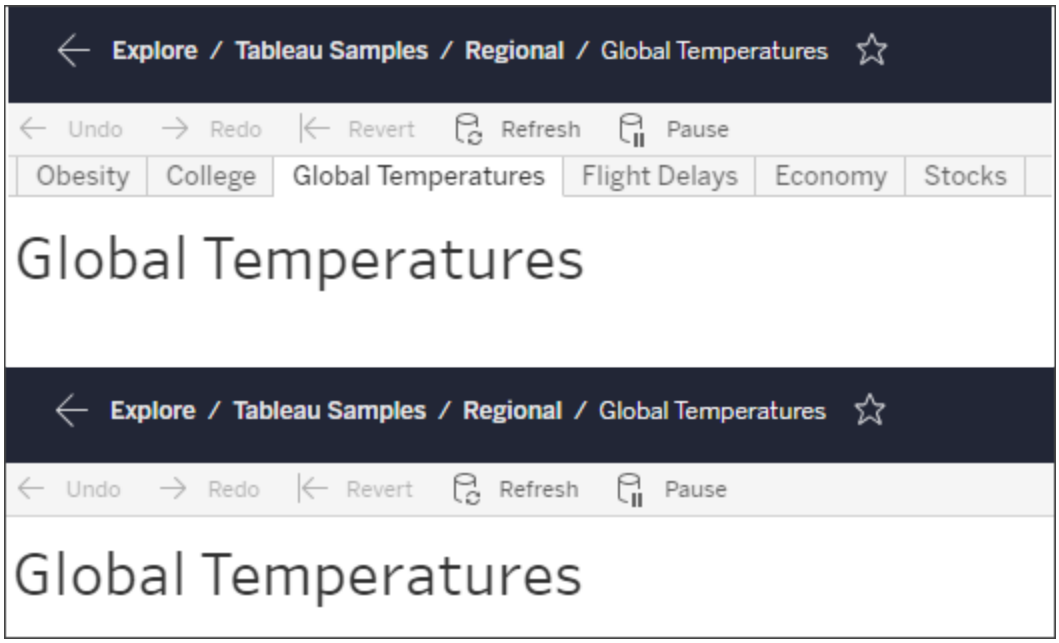
すべてのユーザーが「データの説明を見る」を表示モードで実行するための要件は次のとおりです。

- サイト設定: **[「データの説明を見る」の可用性]** が **[有効]** に設定されている。デフォルトで有効に設定されています。
- サイトロール: Creator、Explorer、または Viewer。
- パーミッション: **[Run Explain Data (「データの説明を見る」を実行)]** が **[許可]** に設定されている。既定では指定されていません。このパーミッションを使用したワークブック (Tableau バージョン 2022.1 以前) を Tableau バージョン 2022.2 以降で開く場合は、[「データの説明を見る」を実行] を [許可] にリセットする必要があります。

シートタブの表示または非表示

パブリッシュされたコンテンツのコンテキストでは、シートタブ(タブ付きビューとも呼ばれる)は Tableau Desktop のシートタブとは異なる概念です。Tableau Desktop でのシートタブの表示と非表示を行う場合は、オーサリング環境でシートを非表示にすることを意味します。詳細については、「ダッシュボードとストーリーでのシートの管理」を参照してください。

パブリッシュされたコンテンツのシートタブの表示と非表示 (タブ付きビューのオン/オフ) は、パブリッシュされたワークブックのナビゲーションを指します。シートタブが表示されている場合、パブリッシュされたコンテンツには、それぞれのビューの上部に並んでナビゲーション シートタブが表示されます。



この設定は、パーミッションの機能方法とセキュリティに影響を与える可能性もあります (注を参照)。

注: ビューを含むワークブックまたはプロジェクトの**ビュー**機能がなくても、あるビューの**ビュー**機能を持つことができます。通常、ユーザーがプロジェクトとワークブックのビュー機能を持たない場合、それらのアセットが存在することを認識できません。しかし、ユーザーがビューのビュー機

能を持つ場合、ナビゲーション階層リンクなどで、ビューを表示するときにプロジェクトとワークブックの名前を表示できる場合があります。これは、想定内の許容される動作です。

タブ付きビューをオフにして、独立したビューのパーミッションを許可する

一般的な方法としては推奨されませんが、ビューを含むワークブックとは独立してビューに対するパーミッションを設定するのが便利な場合があります。これを行うには、次の 3 つの条件を満たす必要があります。

1. ワークブックをパブリッシュする必要があります。パブリッシュ中にビュー パーミッションを設定する方法はありません。
2. ワークブックは、カスタマイズ可能なプロジェクトに含まれる必要があります。
3. ワークブックではシートをタブとして表示できません (タブ付きビューは非表示にする必要があります)。

ワークブックでシートがタブとして表示される場合、すべてのビューはワークブック パーミッションを継承し、ワークブック パーミッションに対する変更はすべてのビューに影響します。**カスタマイズ可能なプロジェクトのワークブックにタブ付きビューが表示されない場合、すべてのビューでは、パブリッシュ時にワークブック パーミッションが前提とされますが、ワークブックのパーミッション ルールに対するその後の変更はビューによって継承されません。**

パブリッシュ済みワークブックでシートをタブとする構成を変更すると、パーミッション モデルにも影響があります。[タブの表示] を使用すると、既存のビューレベルのパーミッションが上書きされ、すべてのビューのワークブックレベルのパーミッションが回復されます。[タブの非表示] を使用すると、ワークブックとそのビューとの関係が壊れます。

- パブリッシュ済みワークブックでシートをタブとして構成するには、ワークブックの [アクション] メニュー (...) を開き、**[タブ付きビュー]** を選択します。必要に応じて、**[タブの表示]** または **[タブの非表示]** を選択します。
- パブリッシュ時にシートをタブとして構成するには、「シートをタブとして表示」を参照してください。
- ビューレベルのパーミッションを設定するには、「アセットのパーミッションの設定」を参照してください。

重要: カスタマイズ可能なプロジェクトでは、ナビゲーション シート タブが非表示の場合 (タブ付きビューがオフ)、ワークブックレベルのパーミッションに対する変更が適用されません。パーミッションに対する変更は、個々のビューに対して行う必要があります。

コレクション

コレクションでのパーミッションの管理については、「コレクション」を参照してください。

パーミッションの機能とテンプレート

パーミッションは機能で構成されています。機能とは、コンテンツに対する特定のアクション (ビュー、フィルター、ダウンロード、または削除) を実行する能力のことです。ダイアログの [パーミッション ルール] 領域の各行は、*パーミッション ルール*です。パーミッション ルールは、その行のグループまたはユーザーの各機能 (許可、拒否、または未指定) を設定するものです。パーミッション ルールには、機能をすばやく割り当てやすくするための*テンプレート*があります。パーミッション ルールはコピーして貼り付けることもできます。

> **注**: プロジェクトのパーミッション ダイアログには、それぞれのコンテンツタイプ (**プロジェクト**、**ワークブック**、**データ ソース**、**データの役割**、**フロー**、**「データに聞く」レンズ**、**メトリクス**) があります。— (データ管理—**仮想接続**、**データベース**、**テーブル**がある場合)。(Tableau Server 2021.4 とTableau Cloud 2021 年 12 月で仮想接続が追加されました。Tableau Server 2022.3、Tableau Cloud 2022 年 10 月でデータベースとテーブルが追加されました)。パーミッション ルールを追加すると、すべてのコンテンツ タイプに対するすべての機能の既定値は [未指定] になります。それぞれのコンテンツ タイプの機能を許可または拒否するには、それぞれのタブに順番に移動する必要があります。特定のコンテンツのパーミッションダイアログには、そのコンテンツのみに適用されるタブおよびパーミッション ルールが存在しません。

テンプレート

テンプレートは、一般的なユーザー シナリオ、**表示**、、**探索**、**パブリッシュ**および **管理** に基づいて、一緒に割り当てられることが多い機能のセットをグループ化します。テンプレートを割り当てると、それに含まれている機能が **[許可]** に設定され、残りは **[未指定]** になります。テンプレートは累積的であるため、探索テンプレートには、ビュー テンプレートに加えて追加の機能がすべて含まれ

ています。また、すべてのコンテンツには、**[なし]** (すべての機能が未指定に設定されます) と**[拒否]** (すべての機能が拒否に設定されます) のテンプレートもあります。

テンプレートは、開始点を表しており、適用後に調整できます。また、機能はテンプレートをまったく使わずに個別に付与または拒否することもできます。どちらの場合も、テンプレート列には **[カスタム]** と表示されます。

パーミッションのコピーと貼り付け

複数のグループまたはユーザーに割り当てる必要があるパーミッション ルールがある場合は、あるルールから別のルールにコピーして貼り付けることができます。プロジェクト リーダーのステータスを含むルールからコピーしたり、貼り付けたりすることはできません。

1. コピー元の既存ルールのアクション メニュー (..) を開き、**[パーミッションのコピー]** を選択します。これは、ルールが編集モードでない場合にのみ使用できます。
2. 貼り付ける既存のルールを選択します。**[+ グループ/ユーザールールを追加]** をクリックして、グループまたはユーザーを選択して、新しいルールを作成することもできます。
3. アクション メニュー (...) を開いて、**[パーミッションの貼り付け]** を選択します。

機能

各コンテンツ タイプには、次のような特定の機能があります。

プロジェクト

プロジェクトには 2 つの機能と2 つのテンプレートしかありません。プロジェクト リーダーの詳細と割り当て方法については、「プロジェクトの管理」を参照してください。

ビューのテンプレート

[ビュー] では、ユーザーはプロジェクトを表示することができます。ユーザーにビュー機能が付与されていない場合、そのユーザーにプロジェクトは表示されません。プロジェクトにビュー機能を付与しても、プロジェクト内のコンテンツをユーザーが表示できるわけではなく、単にプロジェクト自体が表示されます。

パブリッシュのテンプレート

[パブリッシュ] では、ユーザーは Tableau Desktop または Tableau Prep Builder からプロジェクトにコンテンツをパブリッシュすることができます。コンテンツをプロジェクトに移動したり、Web 作成からプロジェクトにコンテンツを保存したりするには、パブリッシュ機能も必要です。

ワークブック

ビューのテンプレート

[ビュー] では、ユーザーはワークブックまたはビューを表示することができます。ユーザーにビュー機能が付与されていない場合、そのユーザーに.twbは表示されません。

[フィルター] では、ユーザーは保持のみや除外フィルターを含め、ビュー内のフィルターを操作することができます。この機能がないユーザーは、ビューにフィルター コントロールが表示されません。

[コメントの表示] では、ユーザーはワークブック内のビューに関連付けられたコメントを表示することができます。

[コメントの追加] では、ユーザーはワークブック内のビューに対してコメントを追加することができます。

[イメージ/PDF のダウンロード] では、ユーザーは各ビューを PNG、PDF または PowerPoint としてダウンロードすることができます。

[サマリー データのダウンロード] では、ユーザーはビュー内や選択したマーク内の集計データを表示したり、そのデータを (CSV として) ダウンロードしたりできます。

探索のテンプレート

[カスタマイズの共有] では、ユーザーはワークブックに表示される“その他のビュー” のリストにカスタム ビューを追加できます。

- この機能が拒否された場合、ユーザーはカスタム ビューを作成するときに [他のユーザーにも表示する] オプションは表示されません。詳細については、「カスタム ビューの使用」を参照

して ください。この機能は、カスタム ビューを共有ダイアログを使用して共有したり、リンクをコピーしたりする機能には影響しません。

[完全なデータのダウンロード] では、ユーザーはビュー内や選択したマーク内の参照元データを表示したり、そのデータを (CSV として) ダウンロードしたりできます。

[Web 編集] では、ユーザーはブラウザーベースの作成環境でビューを編集できます。

- ブラウザーでの新規コンテンツの作成や、Web 編集インターフェイスからのビューの保存には、特定の機能の組み合わせが必要となります。詳細については、Web 編集とWeb 作成を参照してください。
- Web 編集機能はサイト全体に対しても有効にする必要があります。そうしなければ、この機能を持つユーザーであっても Web 編集を行うことができません。詳細については、「サイトの Web 作成アクセスの設定」を参照してください。

[「データの説明を見る」の実行] では、ユーザーはマーク上で「データの説明を見る」を編集モードおよび表示モードで実行できます。

- ユーザーがワークブック内のマークを選択したときに「データの説明を見る」をオプションとして表示するには、この機能をサイト設定として有効にする必要もあることに注意してください。「データの説明を見る」を表示モードで使用できるようにするには、作成者が「データの説明を見る」設定のワークブック内からこの機能を許可する必要もあります。詳細については、「「データの説明を見る」へのアクセスの制御」を参照してください。

パブリッシュのテンプレート

[ワークブックのダウンロード/コピーの保存] では、ユーザーは (TWBX として) パッケージドワークブックをダウンロードすることができます。ユーザーは Web 編集インターフェイスからのコピーを新しいワークブックとして保存 (パブリッシュ) できます。

[上書き] では、ユーザーはサーバー上のコンテンツやアセットを上書き (保存) することができます。

- 許可された場合、ユーザーはワークブック、データソース、またはフローをもう一度パブリッシュしたり、ワークブックやフローを Web 作成に保存したりできるため、所有者としてすべてのパー

ミッションへのアクセス権を得ることになります。所有者の変更後、ワークブックに対する元の所有者のアクセス権は、他のユーザーと同様にパーミッションで判断されます。

[メトリクスの作成/更新] では、ユーザーはワークブック内のビューにメトリクスを作成でき、それらのビューから作成したメトリクスを更新することもできます。従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

管理のテンプレート

[移動] では、ユーザーはプロジェクト間でワークブックを移動することができます。詳細については、コンテンツの移動を参照してください。

[削除] では、ユーザーはワークブックを削除することができます。

[パーミッションの設定] では、ユーザーはワークブックのパーミッション ルールを作成することができます。

[抽出の更新] では、ユーザーはそのワークブックの抽出更新をトリガーできます。

ビュー

ワークブックがロックされているプロジェクトに含まれておらず、かつ、ワークブックにシートがナビゲーションのタブとして表示されていない場合、ワークブックのパーミッションはパブリッシュ時にビュー (シート、ダッシュボード、ストーリー) に継承されますが、パーミッション ルールの変更は個々のビューで行う必要があります。ビュー機能はワークブックでの機能と同じですが、例外として、**[上書き]**、**[ワークブックのダウンロード/コピーの保存]**、**[移動]** の機能はワークブック レベルでのみ利用できます。

できる限りナビゲーション シート タブを表示して、ワークブックのパーミッションがビューに引き続き継承されるようすることをお勧めします。

データ ソース

ビューのテンプレート

[ビュー] では、ユーザーはサーバー上のデータ ソースを表示することができます。

[接続] では、ユーザーは Tableau Desktop、Tableau Prep Builder、「データに聞く」または Web 編集でデータ ソースに接続することができます。

- ワークブック作成者がパブリッシュ済みワークブックのパブリッシュ済みデータ ソースに認証資格情報を埋め込むと、基本的にそれらの**接続**機能が埋め込まれます。このため、ユーザーはそのデータ ソースに対する独自の**接続**機能とは関係なく、ワークブック内のデータを表示することができます。ワークブック作成者が認証資格情報をパブリッシュ済みデータ ソースに埋め込まない場合、ユーザーはそのワークブックを使用するために、データ ソースに対する独自の**接続**機能が必要です。詳細については、「パブリッシュ済み Tableau データ ソースでのデータ アクセス」を参照してください。
- ユーザーは、「データに聞く」レンズを使用するために、データ ソースの**接続**機能を持っている必要があります。詳細については、「サイトおよびデータ ソースでの「データに聞く」機能の有効化」を参照してください。

探索のテンプレート

[データ ソースのダウンロード] では、ユーザーはサーバーからデータ ソースを (TDSX として) ダウンロードすることができます。

- Microsoft Analysis Services や Oracle Essbase 接続用などのキューブ データ ソースは、ローカルで使用する必要があります。パブリッシュされたデータ ソースを Tableau Desktop にダウンロードするには、ユーザーが [ダウンロード] 機能を持っている必要があります。詳細については、「キューブ データ ソース」を参照してください。

パブリッシュのテンプレート

[上書き] では、ユーザーはデータ ソースをサーバーにパブリッシュし、サーバー上のデータ ソースを上書きします。

API アクセスを使用すると、VizQL データサービスを使用してデータソースをクエリできます。詳細については、「VizQL データサービス」を参照してください。

[メトリクス定義の作成] を使用すると、ユーザーはデータソースに接続して Tableau Pulse のメトリクス定義を作成できます。詳細については、「Tableau Pulse でメトリクスを作成する」を参照してください。

管理のテンプレート

[削除] では、ユーザーはデータソースを削除することができます。

[パーミッションの設定] では、ユーザーはデータソースのパーミッション ルールを作成して編集することができます。

[抽出の更新] では、ユーザーはデータソースの抽出更新をトリガーできます。

その他のアセット タイプ

	ビューのテンプレート	探索のテンプレート	パブリッシュのテンプレート	管理のテンプレート
フロー	**[ビュー]** では、ユーザーはフローを表示することができます。	**[フローのダウンロード]** では、ユーザーはフローを (.tflx として) ダウンロードすることができます。	**[実行]** では、ユーザーはフローを実行することができます。 **[上書き]** では、ユーザーはフローをパブリッシュし、パブリッシュされたフローを上書きできます。	**[移動]** では、ユーザーはプロジェクト間でアセットを移動することができます。詳細については、コンテンツの移動を参照してください。 **[削除]** では、ユーザーはアセット

を削除することができます。

[パーミッションの設定] では、ユーザーはアセットのパーミッションルールを作成することができます。

データの役割	**[ビュー]** では、ユーザーはデータの役割を表示することができます。	N/A	**[上書き]** では、ユーザーはデータの役割をパブリッシュし、パブリッシュされたデータの役割を上書きして、パブリッシュされたデータの役割の同義語を編集できます。
メトリクス(廃止)	**[ビュー]** では、ユーザーはメトリクスを表示することができます。	N/A	**[上書き]** では、ユーザーはメトリクスを上書きし、メトリクスの詳細を編集できます。
「データに聞く」レンズ	**[ビュー]** では、ユーザーはレンズを表示することができます。	N/A	**[上書き]** では、ユーザーはレンズを編集することができます。
仮想接続	**[ビュー]** では、ユーザーは仮想接続を表示できます。 **[接続]*** では、ユーザーは仮想接続を使用してデータに接続できます。	N/A	**[上書き]** では、ユーザーは仮想接続を編集できます。
データ	**[ビュー]** では、	N/A	**[上書き]** で

ベース	ユーザーはデータベースを表示することができます。		は、ユーザーはデータベースのメタデータを編集できます。	
テーブル	**[ビュー]** では、ユーザーはテーブルを表示することができます。	N/A	**[上書き]** では、ユーザーはテーブルのメタデータを編集できます。	
コレクション	**[ビュー]** では、ユーザーはコレクションを表示することができます。	N/A	N/A	N/A

仮想接続の既定のカスタム テンプレートでは、表示は [許可] に設定されていますが、接続機能の許可は設定されていません。ユーザーが仮想接続を使用して接続できるように、接続機能を [許可] に設定してください。

プロジェクトを使用したパーミッションの管理

プロジェクトでは、ネストされたプロジェクト、プロジェクトの可視性、管理者以外のプロジェクト リーダー、権限のロックなどの機能を使用すると、パーミッションの管理を簡素化できます。

ヒント: プロジェクト レベルでパーミッションをどのように設定するかは、特にデフォルト プロジェクトの場合に重要です。新しい上位プロジェクトが作成されると、既定のプロジェクトから (すべてのコンテンツ タイプで) 既定のパーミッション ルールが継承されます。別のプロジェクト内でネストされている新しいプロジェクトが作成されると、子プロジェクトは親プロジェクトから既定のパーミッション ルールを継承します。

プロジェクトの管理

プロジェクトは、コンテンツへのアクセスを整理、管理するために使用されるコンテナです。プロジェクトを管理するために管理者以外の権限を付与することで、特定のコンテンツ管理タスクをプロジェ

クトレベルで処理することができます。

プロジェクト リーダー: プロジェクトには、**プロジェクト リーダー** として設定されたユーザーであるプロジェクト リーダーがいます。この設定は、そのプロジェクトおよびそのプロジェクトのすべてのコンテンツに対し、サイト ロールに応じてユーザーに最大の機能を自動的に付与します。Explorer (パブリッシュ可能) 以上のサイト ロールを持つプロジェクト リーダーは、すべての機能を使用できます。プロジェクト リーダーは、基本的に、サイトやサーバー設定へのアクセス権がない、プロジェクトのローカル管理者です。

階層: 上位プロジェクトを作成できるのは管理者だけです。プロジェクト所有者とプロジェクト リーダーは、プロジェクト内でネストされたプロジェクトを作成できます。

プロジェクト所有者とプロジェクト リーダーは、プロジェクトとそのコンテンツに加え、そのプロジェクトに含まれるすべてのネストされたプロジェクトに対する完全な管理アクセス権を持ちます。階層では、プロジェクト リーダーにはすべての子コンテンツへのプロジェクト リーダー アクセスが黙示的に付与されます。[プロジェクト リーダー] アクセス権を削除するには、ロールが明示的に割り当てられた階層内のレベルで、削除を行う必要があります。

所有権: プロジェクトには複数のプロジェクト リーダーを設定することができますが、各プロジェクトの所有者は 1 人です。既定では、プロジェクトは作成したユーザーが所有します。

プロジェクトの所有者は、既存の所有者または管理者が変更できます。(プロジェクト リーダーは、プロジェクトの所有権を変更できません。コンテンツの所有権のみを変更できます)。プロジェクトは、Explorer (パブリッシュ可能)、Creator、管理者のサイト ロールを持つユーザーが所有できます。プロジェクトの所有権は、プロジェクトがロックされている場合でも変更できます。

削除: ほとんどのコンテンツは、プロジェクトにのみ存在できます。上位プロジェクトを作成および削除できるのは管理者だけですが、プロジェクト リーダーはネストされたプロジェクトの作成や削除を行えます。

プロジェクトを削除すると、そこに含まれるすべての Tableau コンテンツとネストされたプロジェクトも削除されます。プロジェクトを削除する際にコンテンツを失わないようにするには、まずコンテンツを別のプロジェクトに移動します。プロジェクトの削除を元に戻すことはできません。

外部アセットの扱いは異なります。プロジェクト内になくてもかまいません。外部アセットのプロジェクトが削除されても、**外部アセット**に引き続き表示される場合、外部アセットは削除されません。詳細については、「プロジェクト内に存在しない外部アセット」を参照してください。

プロジェクトの管理に関する詳細については、「プロジェクトを使用したコンテンツへのアクセスの管理」および「プロジェクトを追加してそこにコンテンツを移動させる」を参照してください。

特別なプロジェクト

Default: 「Default」という名前のプロジェクトは特別なプロジェクトです。他の最上位レベルプロジェクトが作成されると、それらのプロジェクトは Default プロジェクトをテンプレートとして使用し、Default プロジェクトからすべてのパーミッション ルールをコピーします (ただし、**アセット パーミッション**の設定はコピーしません)。**デフォルト**プロジェクトを削除、移動、名前変更することはできませんが、その説明は変更できます。デフォルトでは、このプロジェクトの所有者はいませんが、所有者を割り当てることができます。

外部アセットの既定プロジェクト: Tableau Cloud および Tableau Server 2023.1 以降では、Catalog が有効になっているデータ管理 ライセンスがある場合、Catalog で新規または既存の外部アセットをプロジェクトに移動する必要があるときに、「External Assets Default Project (外部アセットの既定プロジェクト)」という名前のプロジェクトが表示されます。Catalog は、新しい外部アセットと、削除されたプロジェクトからの外部アセットを、**外部アセットのデフォルト プロジェクト**に配置します。デフォルトでは、このプロジェクトにはパーミッション ルールがないため、パーミッションを追加しない限り、このプロジェクトを表示できるユーザーはサーバー管理者とサイト管理者だけになります。このプロジェクトを削除、移動、名前変更することはできませんが、その説明は変更できます。デフォルトでは、このプロジェクトの所有者はいませんが、所有者を割り当てることができます。

プロジェクト リーダーの設定

プロジェクト リーダーとは、特定のプロジェクトまたはプロジェクト階層に対して管理者と同等なアクセス権を持つユーザーです。

プロジェクト リーダーのステータスをグループまたはユーザーに割り当てるには

1. 適切なプロジェクトのパーミッション ダイアログを開きます。
2. 既存のパーミッション ルールを選択するか、または **[+ グループ/ユーザー ルールの追加]** をクリックして、目的のグループまたはユーザーを選択します。

3. そのパーミッション ルールのアクション メニュー (...) を開き、**[プロジェクト リーダーを設定...]** を選択します。

注: アクション メニューに **[「プロジェクト リーダーを設定」を有効にする]** オプションがある場合は、グループまたはユーザーをプロジェクト リーダーとして設定する前に、このオプションを選択する必要があります。このオプションは、そのグループまたはユーザーがプロジェクト リーダー機能 (2020.1 以前) を拒否された場合にのみ表示されます。その拒否された機能は、プロジェクト リーダーとして設定する前に削除する必要があります。

パーミッション ルールでプロジェクト リーダーを設定すると、すべての機能がプロジェクト リーダーに許可されるため、パーミッション ルールのテンプレートと機能は編集できなくなります。ネストされたプロジェクトを含むプロジェクトでプロジェクト リーダーを設定すると、ネストされたすべてのプロジェクトとそのコンテンツに対してプロジェクト リーダーのステータスが継承されます。

プロジェクト リーダーのステータスは、常にプロジェクト階層全体を通じて下方に適用され、設定されたレベルからのみ削除できます。プロジェクト リーダーのステータスを削除するには、同じ手順に従い、アクション メニューから **[プロジェクト リーダーから削除]** を選択します。グループまたはユーザーがプロジェクト リーダーから削除されると、パーミッション ルールのすべての機能は [未指定] に設定されます。これは、グループまたはユーザーにコンテンツへのパーミッションを与える他のパーミッション ルールがない場合、そのプロジェクトへのパーミッションと機能がなくなってしまうことを意味する場合があります。プロジェクトとそのコンテンツへのアクセス権を維持するには、他のグループやユーザーと同様に機能を設定する必要があります。

注: プロジェクト リーダーは、ほとんどの状況でプロジェクトの抽出を更新できます。ネストされたプロジェクト (最上位プロジェクトではなく) の唯一のプロジェクト リーダーであり、最上位プロジェクトが *ロックされている (ネストされたプロジェクトを含む)* 場合、抽出を更新することはできません。

アセット パーミッションのロック

プロジェクト レベルで設定されたパーミッション ルールは、そのプロジェクトに保存されたコンテンツと、そのプロジェクトに含まれるネストされたプロジェクトの既定のルールとして機能します。これらの

プロジェクトレベルの既定のルールが適用されるのか、または単に予備的なものとするのかは、**[Asset permissions (アセット パーミッション)]** の設定によって決まります。この設定は、**[ロック]** (推奨) または **[カスタマイズ可能]** の 2 つの方法のいずれかで構成できます。プロジェクトをロックすると、コンテンツ所有者がコンテンツのパーミッション ルールを変更する機能が削除されます。パーミッションをロックすると、ネストされたプロジェクトに適用することも、親プロジェクト自体だけに適用することもできます。

- **[Asset permissions (アセット パーミッション)]** が **[ロック]** されている場合 (ネストされたプロジェクトを含む)、そのプロジェクトレベルで設定されているパーミッション ルールは、プロジェクト内のすべてのアセットとネストされたすべてのプロジェクトに対して適用されます。
- **[アセット パーミッション]** が **ロック** されている場合 (ネストされたプロジェクトを*含まない*)、そのプロジェクトレベルで設定されているパーミッション ルールは、プロジェクト内のアセットに対して適用されます。ネストされたプロジェクトは、独自のパーミッション ルールを使用して個別に設定することができ、ロックまたはカスタマイズ可能として設定できます。
- **[Asset permissions (アセット パーミッション)]** が **[カスタマイズ可能]** な場合、そのプロジェクトレベルで設定されたパーミッション ルールは、既定でプロジェクト内のすべてのアセットに適用されます。ただし、パブリッシュ中またはパブリッシュ後に、個々のアセットに対してパーミッション ルールを変更することはできます。

注: パーミッション ルールがロックされているかカスタマイズ可能であるかにかかわらず、コンテンツに対するパーミッションは常に適用されます。*ロック済*および*カスタマイズ可能*な場合は、プロジェクトレベルのパーミッションがプロジェクトのコンテンツにどのように継承され、それらを変更できるユーザーだけを参照します。カスタマイズ可能なパーミッションを持つプロジェクトでも、特定のユーザーのみがパーミッション (コンテンツまたはプロジェクトの所有者、プロジェクト リーダー、管理者、または [権限の設定] 機能を持つユーザー) を変更できます。

ロックされたプロジェクト内:

- コンテンツ タイプごとのプロジェクト パーミッション ルールは、すべてのアセットに適用されます。
- パーミッションを変更できるのは、管理者、プロジェクト所有者およびプロジェクト リーダーだけです。
- コンテンツ所有者は、パーミッションの設定機能を失いますが、コンテンツに対する他の機能はすべて保持します。
- プロジェクト内にあるコンテンツすべてのパーミッションは予測可能です。

カスタマイズ可能なプロジェクト内:

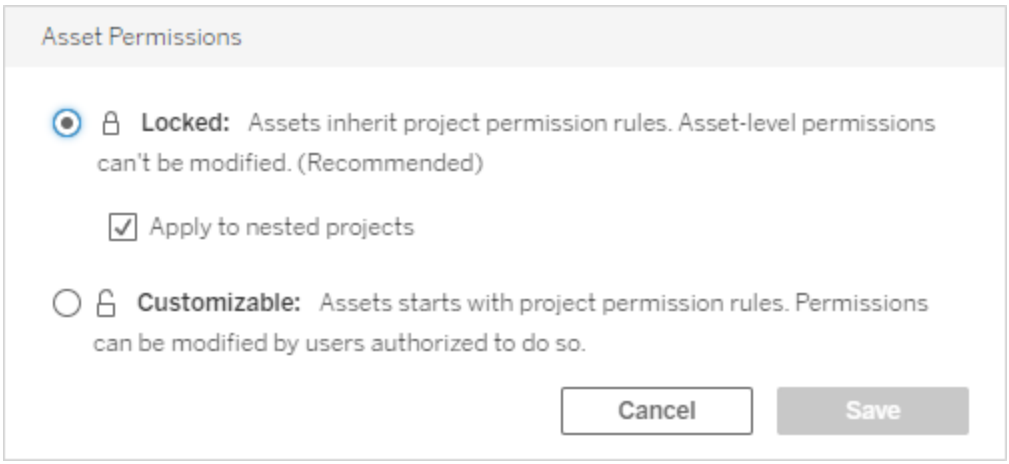
- このパーミッション ルールは、コンテンツがプロジェクトにパブリッシュされるとき、またはネストされたプロジェクトが作成されるときに既定で適用されますが、パーミッションはパブリッシュ中またはコンテンツの作成後に変更することができます。
- パーミッションの設定機能を持つユーザーは、そのコンテンツのパーミッション ルールを変更できます。
- コンテンツ所有者は、コンテンツに対するすべての機能を持っています。
- パーミッションは、プロジェクトのコンテンツ全体で異なる場合があります。
- カスタマイズ可能な親プロジェクトのパーミッションを変更しても、ネストされたプロジェクトには適用されません。

アセット パーミッションの設定 (プロジェクトのロック)

新しい上位レベルのプロジェクトは、既定プロジェクトからすべての初期パーミッション ルールを継承しますが、**[カスタマイズ可能]** に設定されている **[Asset permissions (アセット パーミッション)]** 設定は継承されません。必要に応じて、設定を **[ロック]** に変更することができます。

[Asset permissions (アセット パーミッション)] を構成するには、次の手順を実行します。

1. サイトに管理者、プロジェクト所有者、またはプロジェクト リーダーとしてログインする必要があります。
2. プロジェクトで [パーミッション] ダイアログを開きます
3. 左上の **[Asset permissions (アセット パーミッション)]** の横にある **[編集]** リンクをクリックして、**[Asset permissions (アセット パーミッション)]** ダイアログで必要なオプションを選択します。



注: 上の手順 3 で左上隅に **[編集]** リンクが表示されない場合は、次のいずれのパーミッション ダイアログが表示されます。(a) ネストされたプロジェクト、またはロックされているプロジェクト内のコンテンツ。この場合、このリンクを使うと管理プロジェクトに移動できます。(b) カスタマイズ可能なプロジェクトのコンテンツの一部。これは何も表示されません。または (c) ビューのアクセス パーミッションがそのワークブックにどのように関連付けられているかを示すビュー。ビューとワークブックのパーミッションの相互作用に関する詳細については、「シート タブの表示または非表示」を参照してください。

アセット パーミッションの変更

プロジェクトの **[Asset permissions (アセット パーミッション)]** 設定を変更すると、その新しい設定に応じて結果が異なります。ロックされている階層でのパーミッション ルールに対する変更は、管理プロジェクトのレベルで行う必要があります。

変更元	変更先	結果
ロック済み (ネストされたプロジェクトを含む)	ロック済み	既存のパーミッション ルールは変更しません。 ネストされたプロジェクトは、カスタマイズ可能になります。
	カスタマイズ可能	既存のパーミッション ルールは変更しませんが、カスタマイズ可能になります。 ネストされたプロジェクトは、カスタマイズ可能になります。
ロック済み	ロック済み (ネストされたプロジェクトを含む)	ネストされたすべてのプロジェクトとそのコンテンツに対して、既存のカスタム パーミッション ルールを上書きします。これは元に戻すことはできません。
	カスタマイズ可能	既存のパーミッション ルールは変更しませんが、カスタマイズ可能になります。

		ネストされたプロジェクトは、コンテンツのパーミッション設定とパーミッション規則を保持します。
カスタマイズ可能	ロック済み (ネストされたプロジェクトを含む)	プロジェクト内のコンテンツ、およびすべてのネストされたプロジェクトとそのコンテンツに対して、既存のカスタム パーミッション ルールを上書きします。これは元に戻すことはできません。
	ロック済み	プロジェクト内のコンテンツに関する既存のカスタム パーミッション ルールを上書きします。これは元に戻すことはできません。 ネストされたプロジェクトは、パーミッション ルールを保持し、カスタマイズ可能な状態を維持します。

プロジェクトとコンテンツの移動

Tableau コンテンツと外部アセットの移動

パーミッション設定が異なるプロジェクト間で *Tableau コンテンツ*または*外部アセット*を移動する場合、**[Asset permissions (アセット パーミッション)]** 設定によって、パーミッションを適用する方法のロジックが決まります。

- ロックされたプロジェクトにアセットを移動すると、既存のパーミッション ルールが上書きされ、移動先のパーミッションが適用されます。
- **カスタマイズ可能なプロジェクトにアセットを移動すると、アセットに対する既存のパーミッション ルールが維持されます。**

注: Tableau Server 2022.3 および Tableau Cloud 2022 年 6 月より前は、外部アセットをプロジェクトに含めることができませんでした。また、テーブルに対するパーミッションは、親データベースの **[テーブルのパーミッション]** 設定を介して管理されていました。Tableau Server 2022.3 および Tableau Cloud 2022 年 6 月以降では、外部アセットをプロジェクトに含めることができます。データベースまたはテーブルをプロジェクトに移動した場合、データベースを介し

てテーブルのパーミッションを管理していた古い設定は無視され、データベースまたはテーブルのパーミッションは、他のアセットのロジックに従うようになります。

プロジェクトの移動

*プロジェクト*を別のプロジェクトに移動する場合、移動先のプロジェクトの範囲がネストされたプロジェクトを含むように範囲が設定されていない限り、移動するアイテムのパーミッション設定は維持されます。(この場合のプロジェクト パーミッションとは、プロジェクト自体の表示とパブリッシュの機能を意味します。)

- 移動先のプロジェクトが **[ロック (ネストされたプロジェクトを含む)]** に設定されている場合、移動するプロジェクトとコンテンツのパーミッションは上書きされます。
- 移動先のプロジェクトが **[ロック]** (ネストされたプロジェクトを含まない) に設定されている場合、移動するプロジェクトのパーミッションは上書きされません。移動したプロジェクトがロックまたはカスタマイズ可能であるかは、元の設定から維持されます。
- 移動先のプロジェクトが**カスタマイズ可能**に設定されている場合、移動するプロジェクトのパーミッションは上書きされませんが、編集可能になります。

  移動するプロジェクトが、*ロック (ネストされたプロジェクトを含む) された親の下で以前ネストされていた場合*、移動時に、そのプロジェクトは *ロック (ネストされたプロジェクトを含む) の設定を引き継ぎ*、そのプロジェクトに含まれる任意のプロジェクトの管理プロジェクトになります。
  注: これは、プロジェクトが最上位プロジェクトに移動した場合と同じ結果になります。

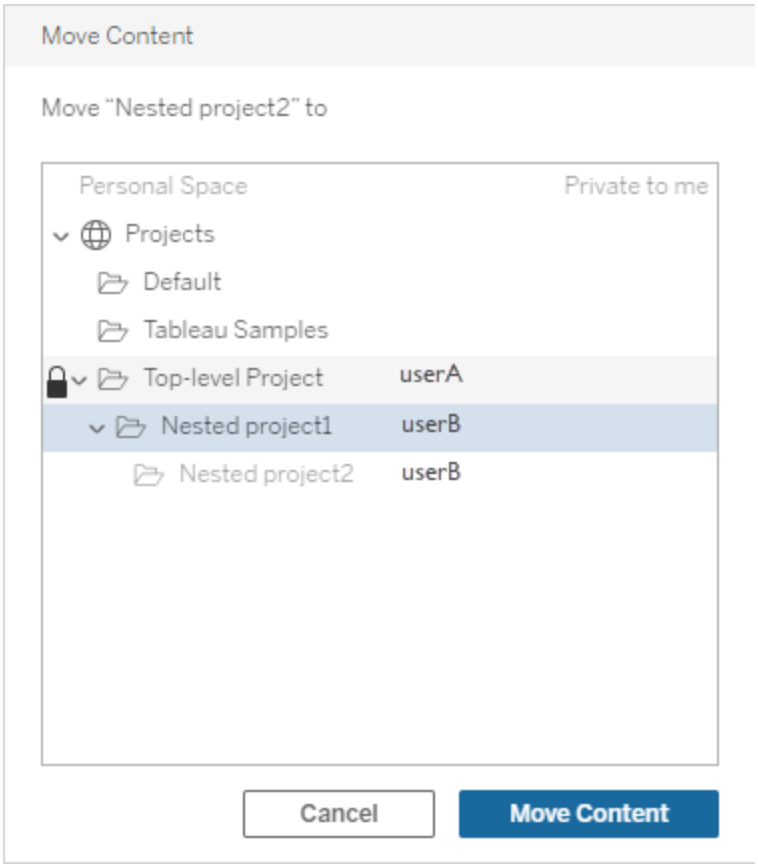
ロックされ、ネストされたプロジェクトの移動には注意してください

ロックされた (ネストされたプロジェクトを含む) 環境内で、ネストされたプロジェクトを移動するのは難しい場合があります。ユーザーが再び移動できないような状況にプロジェクトを移動させる可能性があります。

ネストされたプロジェクトが管理プロジェクトとは異なるユーザーによって所有されており、管理プロジェクトが *ロックされている (ネストされたプロジェクトを含む)* 場合、管理者以外はネストされたプロジェクトを移動できなくなる可能性があります。

たとえば、ユーザーA が所有するロックされた(ネストされたプロジェクトを含む)最上位プロジェクトと、ユーザーB が所有する2 つのネストされたプロジェクトについて考えます。ユーザーB が一方のネストされたプロジェクトを他方のプロジェクト内に移動すると、元に戻すことはできなくなります。ユーザーA も戻せません。

- ユーザーB は、移動先として**最上位プロジェクト**の権限を移動する権限を持っていないため、**ネストされたプロジェクト2** を移動できません。
- ユーザーA は、移動権限がないため、**ネストされたプロジェクト2** を移動できません。
- **最上位プロジェクト**のプロジェクトリーダーは、ネストされたプロジェクトにトリクルダウンしても、そのプロジェクトを移動できません。
- この状況では、**ネストされたプロジェクト2** を移動できるのは管理者だけです。



コレクション

コンテンツを含むプロジェクトとは異なり、コレクションはコンテンツへのリンクのリストと考えることができます。プロジェクトのパーミッションはプロジェクトのコンテンツにより継承されますが、コレクションの

パーミッションはコレクションに追加されるコンテンツに影響を与えません。つまり、アイテムを表示するためのパーミッションに応じて、ユーザーごとにコレクション内のアイテム数が異なる可能性があります。ユーザーがコレクション内のすべてのアイテムを表示できるようにするには、それらのアイテムのパーミッションを個別に調整します。

コレクションのパーミッションは、パーミッション ダイアログを使用して変更するか、管理者またはコレクションの所有者である場合は、コレクションの共有時にアクセス権を許可することによって変更できます。詳細については、「コレクションへのアクセス権限の管理」を参照してください。

プライベート コレクション

コレクションを作成すると、既定でプライベートに設定されます。プライベート コレクションは所有者の [My Collections (マイ コレクション)] ページに表示されますが、サイトのすべてのコレクションのリストには表示されません。プライベート コレクションは、パーミッション ルールが追加されていない単なるコレクションです。他のタイプのコンテンツとは異なり、コレクションには [すべてのユーザー] グループが既定で追加されていません。コレクションにパーミッション ルールを追加すると、プライベートのフラグが付かなくなります。コレクションをプライベートの状態に戻すには、パーミッション ルールを削除します。

プライベート コレクションは、コレクションの所有者と管理者が表示でき、そのサイト ロールによって、すべてのコレクションを表示するための有効なパーミッションが付与されます。

有効なパーミッション

パーミッション ルールでは、影響を受けるユーザー (グループ セット、グループまたはユーザー) と、そのユーザーによる使用が**許可**または**拒否**されている、あるいは、**指定されていない**機能を設定します。パーミッション ルールを単純に設定してそれを全体に適用するのは簡単なことのように思われますが、メンバーシップが複数グループに含まれていることや、サイト ロールおよび所有権がパーミッション ルールと相互に作用しているため、ユーザーが機能を使用できるかどうかが明確でない場合があります。

複数の要因は特定の順序で評価され、コンテンツに対する*有効なパーミッション*が付与されます。

ヒント: できるだけ簡単な状態に保てるよう、(1) ユーザーではなくグループに対してパーミッション ルールを設定すること、(2) 個々のコンテンツに対してパーミッションを設定するのではなく、プロジェク

トレベルでロックされているパーミッションを管理すること、(3) [すべてのユーザー] グループのパーミッション ルールを削除するか、すべての機能を [なし] に設定することをお勧めします。

次の 3 つの条件がすべて満たされた場合にのみ、ユーザーに対して機能が許可されます。

- その機能がサイト ロールの範囲内に含まれている。
- ユーザーはその機能を持っている:
  - 特定のユーザー シナリオに基づく(コンテンツ所有者やプロジェクト リーダーである、または管理者サイト ロールであるなど)、
    または
  - ユーザーとして機能が許可されている、
    または
  - 機能を許可されているグループに含まれており、ユーザーまたは別のグループのメンバーとして機能を拒否するルールがない。
- 優先度の高い別のコンテンツ レベルに競合するパーミッション設定がない。

他の状況ではユーザーで機能が拒否されます。

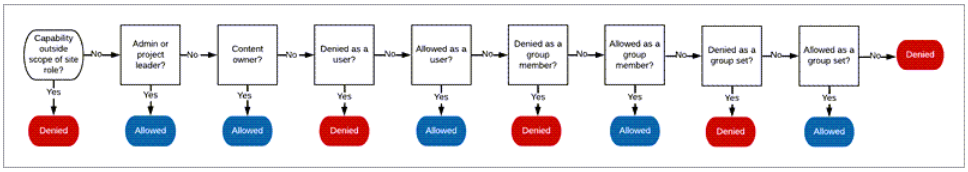
機能にカーソルを合わせると、有効なパーミッションを説明するツールヒントが表示されます。有効なパーミッション (実際に何を実行でき、何を実行できないか) が、指定したパーミッション ルールの記載とは異なって表示される場合がある理由について、いくつかの例を示します。

- ユーザーの機能が、サイト ロールに含まれているが (管理者)、パーミッション ルールで拒否されている可能性がある。
- ユーザーの機能が、ユーザー シナリオで許可されているが (コンテンツ所有者である、またはプロジェクト所有者やリーダーであるため)、パーミッション ルールで拒否されている可能性がある。
- ユーザーの機能が、パーミッション ルールで許可されているが、サイト ロールで許可されていない可能性がある。
- ユーザーの機能が、パーミッション ルールで許可されているが、競合するグループまたはユーザーのルールにより拒否されている可能性がある。
- ユーザーの機能が、あるレベルのコンテンツ (ワークブックなど) のパーミッション ルールで許可されているが、別のレベルのコンテンツ (ビューなど) により拒否されている可能性がある。

パーミッション ルールの評価

Tableau のパーミッションはより制限的です。ユーザーに機能が付与されない限り、パーミッションは拒否されます。次のロジックでは、個々のユーザーで機能が許可されるか拒否されるかを評価し

ます。



1. **サイト ロール:** サイト ロールで機能が許可されない場合、ユーザーは拒否されます。ユーザーのサイト ロールで機能が許可されている場合は、特定のユーザー シナリオが評価されます。
    - たとえば、Viewer サイト ロールでは Web 編集はできません。各サイト ロールで実行可能な内容の詳細については、「各サイト ロールで許可されている一般的な機能」を参照してください。
2. **特定のユーザー シナリオ:**
    - ユーザーが管理者の場合、すべてのコンテンツに対してすべての機能を持っています。
    - ユーザーがプロジェクト所有者またはプロジェクト リーダーの場合、プロジェクトに含まれるすべてのコンテンツに対してすべての機能を持っています。
    - ユーザーがコンテンツ所有者の場合、それらのコンテンツに対してすべての機能*を持っています。
    - これらのシナリオがユーザーに適用されない場合、ユーザー ルールが評価されます。

    *例外: コンテンツ所有者は、パーミッションがロックされているプロジェクトでは **[パーミッションの設定]** 機能を持っていません。管理者、プロジェクト所有者、プロジェクト リーダーのみがロックされたプロジェクトでパーミッション ルールを設定できます。
3. **ユーザー ルール:** ユーザーが機能を拒否されている場合は拒否されます。機能が許可されている場合は許可されます。機能が指定されていない場合は、グループ ルールが評価されます。
4. **グループ ルール:** ユーザーが機能を拒否されているグループのいずれかに含まれている場合は拒否されます。ユーザーが機能を許可されているグループに所属している場合 (かつ、その機能が拒否されているグループに所属していない場合) は許可されます。
    - つまり、ユーザーが 2 つのグループに所属しており、一方では機能が許可され、他方では同じ機能が拒否されている場合、拒否が優先されるため、そのユーザーは拒否されます。
5. **グループ セット ルール:** ユーザーがグループ セット内のグループのメンバーである場合、グループ セット内のいずれかのグループで機能が拒否されると、そのユーザーは拒否されます。

6. 上記の条件がいずれも適用されない場合は、ユーザーはその機能を拒否されます。つまり実際には、指定されていない機能は拒否されます。

このため、次の 3 つの状況では、最終的に有効なパーミッションが**許可**になります。

- サイトロール (サーバー管理者、サイト管理者 Creator、サイト管理者 Explorer) により許可される
- ユーザーがコンテンツ所有者、プロジェクト所有者、またはプロジェクトリーダーであるため許可される
- グループルール、グループセットルール、またはユーザールールにより許可される (かつ、優先度の高いルールにより拒否されない)

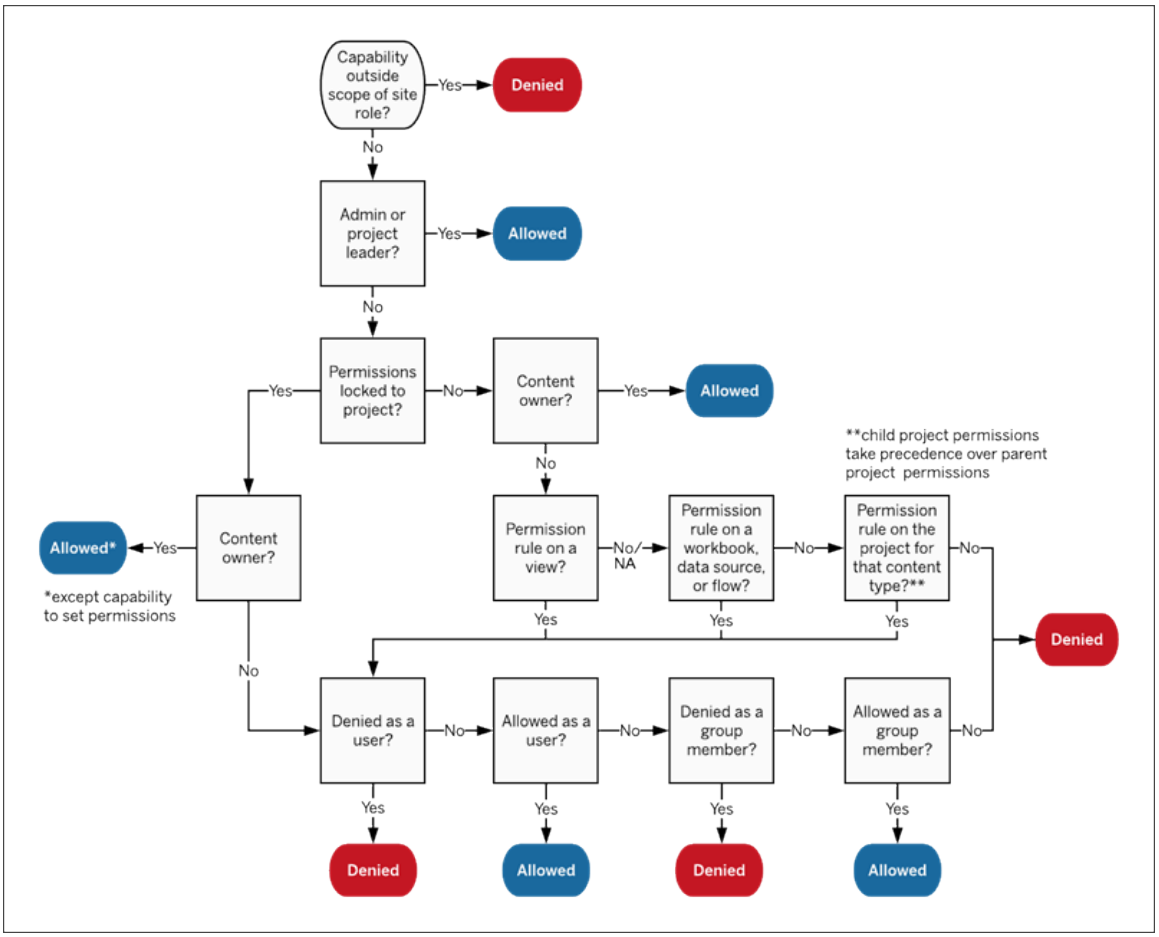
次の 3 つの状況では、**拒否**になります。

- サイトロールにより拒否される
- ルールにより拒否される (かつ、優先度の高いルールにより許可されない)
- いずれのルールによっても付与されない

複数レベルで設定されたパーミッションの評価

アセットパーミッションが**カスタマイズ可能**に設定されている場合、複数の場所でパーミッションルールを構成することができます。どのパーミッションをコンテンツに適用するかを決める特定のルールがあります。

- ネストされたプロジェクトがある場合、子レベルで設定されたパーミッションは、親レベルで設定されたパーミッションよりも優先されます。
- パーミッションの変更をプロジェクトレベルで行っても、既存のコンテンツには適用されません。
- パブリッシュ中またはパブリッシュ後にコンテンツ (ワークブック、データソース、またはフロー) に設定されたパーミッションがある場合は、これがプロジェクトレベルで設定されたルールより優先されます。
- ワークブックにナビゲーション シート タブが表示されない場合、ワークブックレベルのパーミッションに対する変更はビューによって継承されず、パーミッションの変更はビューで行う必要があります。
- ナビゲーション シート タブを表示するようにワークブックを構成すると、既存のビューレベルのパーミッションを上書きし、それらをワークブックレベルのパーミッションと同期します。シートタブの表示または非表示を参照してください。



この画像は、複数のレベルのコンテンツで機能がどのように評価されるかを示しています。

ビューに対するパーミッション

ワークブックがロックされているプロジェクトに含まれておらず、かつ、ワークブックにシートがナビゲーションのタブとして表示されていない場合、ワークブックのパーミッションはパブリッシュ時にビュー(シート、ダッシュボード、ストーリー)に継承されますが、パーミッションルールの変更は個々のビューで行う必要があります。ビュー機能はワークブックでの機能と同じですが、例外として、**[上書き]**、**[ワークブックのダウンロード/コピーの保存]**、**[移動]** の機能はワークブックレベルでのみ利用できます。

できる限りナビゲーション シート タブを表示して、ワークブックのパーミッションがビューに引き続き継承されるようすることをお勧めします。詳細については、シート タブの表示または非表示を参照してください。

パーミッション、サイトロール、ライセンス

ユーザーをTableau Cloud に追加するには、使用可能なライセンスが必要です。(ユーザーをライセンスなしで追加および構成して、最初にサインインしたときにのみライセンスを使用するようにすることもできます。詳細については、サインイン時にライセンスを付与を参照してください)。ユーザーは、所属するサイトごとに 1 つのサイトロールを持っており、そのライセンスによって制限が設けられています。ユーザーにはサイト上のコンテンツに対するパーミッションがあり、サイトロールで許可される内容により制限されます。

ライセンスとサイトロールはユーザーに適用されます。パーミッション機能はコンテンツに適用されます。

ライセンスは、Tableau Server または Tableau Cloud サイトで作成されるとき (または初めてサインインするとき) にユーザーに割り当てられます。ユーザーは **Creator**、**Explorer**、または **Viewer** としてライセンスが付与されます。

- ライセンスレベルは、ユーザーがそのサーバーに対して持つ最大の*サイトロール*に基づいて消費されます。
    - サーバー管理者、サイト管理者 Creator、Creator のサイトロールは、Creator ライセンスを使用します。
    - サイト管理者 Explorer、Explorer (パブリッシュ可能)、Explorer のサイトロールには、少なくとも Explorer ライセンスを使用します。
    - Viewer サイトロールは、少なくとも Viewer ライセンスを使用します。
    - ライセンスのないユーザーはサーバーに存在することはできますが、サインイン時にサイトロールが付与されて追加されない限りログインすることはできません。
- Tableau Server では、ユーザーが複数のサイトに所属する場合でも、サーバーごとに 1 つのライセンスのみを使用します。ユーザーが複数のサイトに所属する場合、必要なライセンスレベルは最も高いサイトロールによって決まります(たとえば、ユーザーが 1 つのサイトで Creator サイトロールを持っており、他の 2 つのサイトで Viewer サイトロールを持っている場合、Creator ライセンスが消費されます)。

サイトロールは、ユーザーがメンバーとなっているサイトごとに割り当てられます。

- サイトロールによって、ユーザーがそのサイトで持つ最大の機能が決まります(たとえば、Viewer のサイトロールを持つユーザーは、特定のデータソースをダウンロードする機能が明示的に付与されている場合でも、データソースをダウンロードすることはできません)。

- サイトロールは元々、そのロールに含まれている機能を付与するわけではありません。ただし、管理者サイトロールは例外です。管理者は常に、そのライセンスレベルに適用されるすべての機能を持ちます。

パーミッションは、プロジェクトへの保存、ワークブックの Web 編集、データソースへの接続などの*機能*で構成され、特定のコンテンツ (プロジェクト、データソース、ワークブック、ビュー、またはフロー) のグループまたはユーザーに適用されます。

- パーミッションの機能は、他とは独立してグループまたはユーザーに付与されるのではなく、コンテンツに照らして付与されます。ユーザーは、異なるコンテンツ資産に対して異なる機能を持つことができます。
- パーミッションは、ユーザーのサイトロールとそのユーザーまたは所属しているグループのパーミッション ルールの相互作用に基づいて評価されます。
- Web 作成などの一部のアクションでは、機能の組み合わせが必要になる場合があります。詳細については、特定のシナリオでのパーミッション設定を参照してください。

サイトロールで許可される最大の機能

これらのテーブルは、各サイトロールで使用できる機能を示しています。サイトロールを持つユーザーが他の方法で同様のアクションを実行できる場合もあります。たとえば、Viewer には **[カスタマイズの共有]** 機能が付与されていないため、カスタム ビューをワークブック上の他のユーザーに表示することができませんが、ビューの URL をコピーしてカスタム ビューを共有することはできます。各サイトロールで実行可能な内容の詳細については、「各サイトロールで許可されている一般的な機能」を参照してください。

プロジェクト

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
パブリッシュ	✓	✓	✗	✗

ワークブック

機能	Creator	Explorer (パブ	Explorer	Viewer

	リッシュ可能)			
表示	✓	✓	✓	✓
フィルター	✓	✓	✓	✓
コメントの表示	✓	✓	✓	✓
コメントの追加	✓	✓	✓	✓
画像/PDF のダウンロード	✓	✓	✓	✓
サマリーデータのダウンロード	✓	✓	✓	✓
「データの説明を見る」の実行 †	✓	✓	✓	✓
カスタマイズの共有	✓	✓	✓	✗
すべてのデータのダウンロード	✓	✓	✓	✗
Web 編集	✓	✓	✓	✗
ワークブックのダウンロード/コピーを保存	✓	✓	✓	✗

上書き	✓	✓	✗	✗
メトリクスの作成/更新 ‡	✓	✓	✗	✗
移動	✓	✓	*	✗
削除	✓	✓	✗	✗
パーミッションの設定	✓	✓	✗	✗
抽出の更新	✓	✓		✗

† Tableau 2021.3 より前は、「データの説明を見る」の可用性は、tsm configuration set オプションの ExplainDataEnabled を使用してサーバー レベル ✗ でのみ制御されていました。2021.3 以降では、「データの説明を見る」の可用性は、「データの説明を見る」の実行機能を使用してサイト設定およびワークブックで制御できます。表示モードでの「データの説明を見る」の可用性は、[「データの説明を見る」の設定] ダイアログ ボックスのワークブックで制御されます。

‡ Tableau 2021.3 より前は、メトリクスの作成/更新機能は、[フル データのダウンロード] 機能によって制御されていました。

データ ソース

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
接続	✓	✓	✓	✓
データ ソースのダウンロード	✓	✓	✓	✗

上書き	✓	✓	✗	✗
メトリクス定義の作成	✓	✓	✗	✗
削除	✓	✓	✗	✗
パーミッションの設定	✓	✓	✗	✗
抽出の更新	✓	✓	✗	✗

データの役割

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
上書き	✓	✓	✗	✗
移動	✓	✓	*	✗
削除	✓	✓	✗	✗
パーミッションの設定	✓	✓	✗	✗

フロー

スケジュールに従ってフローを実行するには、データ管理 ライセンスが必要です。フロー設定の詳細については、「Web 上でのフローの作成と操作」を参照してください。Explorer ライセンスのユーザーは、Tableau Cloud でフローを実行できます。

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
フローのダウンロード	✓	✓	✓	✗
Web 編集	✓	✗	✗	✗
フローの実行	✓	✗	✗	✗
上書き	✓	✗	✗	✗
移動	✓	✓	*	✗
削除	✓	✓	✗	✗
パーミッションの設定	✓	✓	✗	✗

「データに聞く」レンズ

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
上書き	✓	✓	✗	✗
移動	✓	✓	*	✗
削除	✓	✓	✗	✗

パーミッションの設定	✓	✓	✗	✗

メトリクス (2024.2 で廃止)

ヒント: このセクションでは、従来のメトリクス機能について説明しています。この機能は、Tableau Cloud では 2024 年 2 月に廃止され、Tableau Server ではバージョン 2024.2 で廃止されました。Tableau Pulse のメトリクスに対するパーミッションの詳細については、「Tableau Pulse 用にサイトをセットアップする」を参照してください。

従来のメトリクス機能の廃止

Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
上書き	✓	✓	✗	✗
移動	✓	✓	*	✗
削除	✓	✓	✗	✗
パーミッションの設定	✓	✓	✗	✗

コレクション

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓

仮想接続

仮想接続には、データ管理 ライセンスが必要です。詳細については、「データ管理 について について」を参照してください。

機能	Creator	Explorer (パブリッシュ可能)	Explorer	Viewer
表示	✓	✓	✓	✓
接続	✓	**	**	**
上書き	✓	✗	✗	✗
移動	✓	✓	*	✗
削除	✓	✓	✗	✗
パーミッションの設定	✓	✓	✗	✗

* Explorer ロールに **[移動]** 機能を付与することはできますが、プロジェクトの **[パブリッシュ]** 機能を付与することはできないため、コンテンツはどこにも移動できません。そのため、Explorer サイトロールでは、**[移動]** 機能は使用できないものと認識する必要があります。

** Explorer (パブリッシュ可能) ロールは仮想接続の**接続する**機能を利用できますが、仮想接続を含むあらゆる種類の新しいデータソースを作成できるのは、Creator サイトロールを持つユーザーのみです。同様に、Explorer ロールおよび Viewer ロールのユーザーは、新しいデータソースまたは既

存のデータソースに接続するための UI にアクセスできません。**接続する**機能は、Creator 以外のロールでは使用できないと考える必要があります。

クイックスタート: パーミッション

パーミッション ルールは、ワークブック、プロジェクト、データソースなどのコンテンツに対するグループまたはユーザーのアクセス権を定義する一連の機能です。

パーミッションを効率よく管理するには:

- さらに多くのグループを作成する前に、**[すべてのユーザー]** グループからパーミッションを削除する
- さらに多くのプロジェクトを作成する前に、**既定の**プロジェクトでテンプレートのパーミッションを構成する
- ユーザーではなく、グループのパーミッションを管理する
- コンテンツではなく、プロジェクトのパーミッションを管理する

プロジェクトに対してグループのパーミッション ルールを作成する

以下のステップの詳細については、パーミッションに関する主な記事を参照してください。このクイックスタート ガイドは概要であるため、パーミッションやパーミッションの管理に関する多くの重要な詳細は記載されていません。

1.ユーザーをグループに追加する

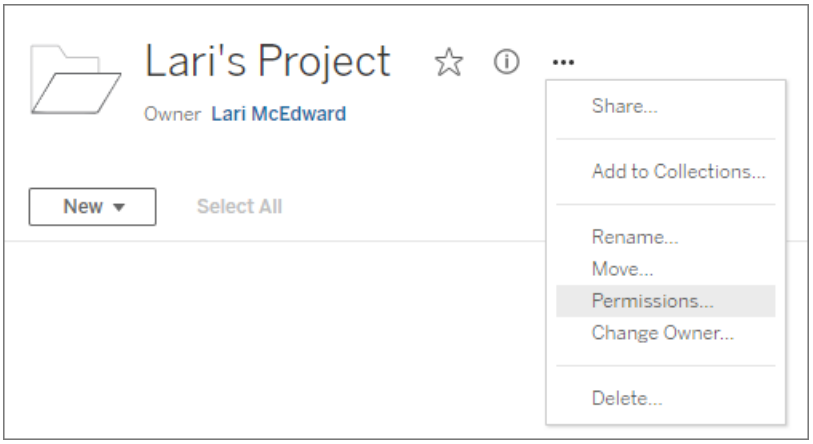
パーミッションを管理する一般的な方法として、同じパーミッションを持つユーザーに対してグループを使用します。

1. 必要に応じて、サイトにユーザーを追加します。
2. サイト内で **[グループ]** を選択します。
3. 必要に応じて、**[グループの追加]** オプションを使用して、グループを作成します。
4. [グループ名] をクリックして開き、**[ユーザーの追加]** ボタンを使用して、既存のユーザーをグループに追加します。

2.プロジェクトレベルのパーミッション設定にアクセスする

[探索] ページには、サイト上のコンテンツが表示されます。ドロップダウンを使用して、[上位レベルのプロジェクト] または [すべてのプロジェクト] を表示します (ネストされたプロジェクトも表示されます)。

更新するプロジェクトに移動し、**[アクション]** (**...**) メニューを開き、**[パーミッション]** を選択します。

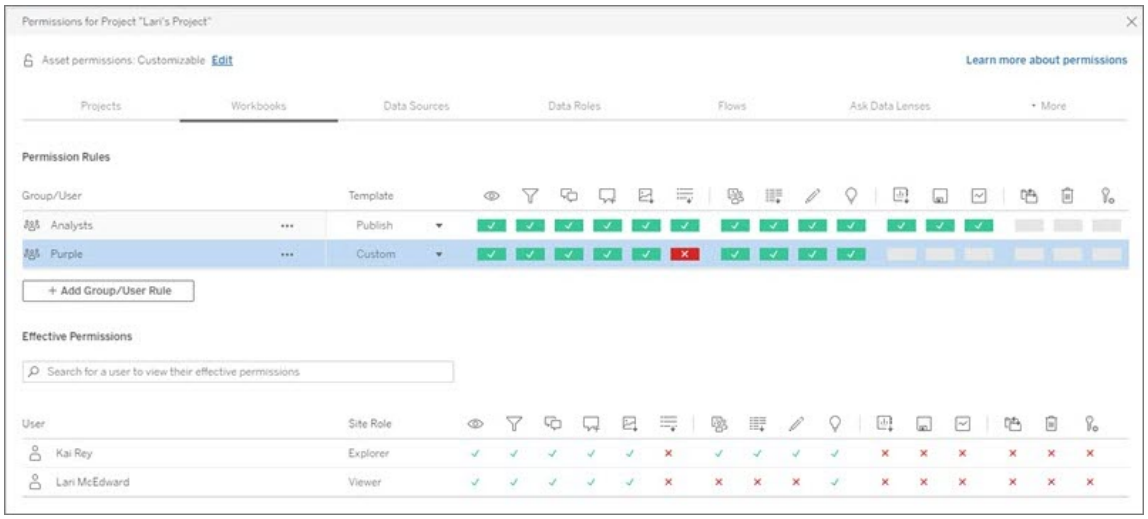


3.パーミッション ルールを作成する

新しいパーミッション ルールを作成するには、**[+ Add Group/User Rule (+グループ/ユーザー ルールの追加)]** を選択します。

[テンプレート] ドロップダウンには、グループの初期機能セットを適用するためのショートカットが用意されています。

必要であれば、機能をクリックして **[許可]** または **[拒否]** に設定するか、**[未指定]**.のままにして、パーミッション ルールをカスタマイズします。



ユーザーがパーミッションを設定できるかどうかは、サイトロール、コンテンツの所有権、および**パーミッションの設定機能**の設定方法によって決まります。

4.ユーザーの有効なパーミッションを表示する

グループのパーミッション ルールを保存したら、各ユーザーの有効なパーミッションを表示することができます。グループ名をクリックし、グループのユーザーおよびパーミッションを表示します。権限ボックスの上にマウスのポインタを置き、その権限が許可されているか拒否されているかの詳細に関するツールヒントを確認します。

サイトロール

ユーザーのサイトロールは、ユーザーが取得できる最大のパーミッション数を決定します。

- サーバー管理者とサイト管理者は、すべてのサイトコンテンツにアクセスし、それに対してアクションを実行できます。
- 所有者は、彼らがパブリッシュしたコンテンツに対して常にフル アクセス権を持ちます。親プロジェクトのパーミッションがロックされていない場合は、所有者はパブリッシュされたコンテンツのパーミッションを変更することができます。

詳細については、ユーザーのサイトロールの設定およびプロジェクトを使用したコンテンツへのアクセスの管理を参照してください。

パーミッションの論理

- **拒否**は**許可**より優先されます。
- **未指定**の場合、他にパーミッションが指定されていないと**拒否**になります。
- コンテンツで具体的なユーザー パーミッションが設定されている場合、この設定が、コンテンツに対するグループ パーミッションより優先されます。つまり、ユーザー パーミッションはグループ パーミッションより優先されます。

詳細については、「有効なパーミッション」を参照してください。

コンテンツ所有権の管理

Tableau Server でデータ ソース、ワークブック、フローをパブリッシュするか、プロジェクトを作成すると、その*所有者*となります。コンテンツ所有者、適切なサイト ロールを持つプロジェクト リーダー、または管理者がコンテンツ資産の所有権を変更できます。所有権を再割り当てすると、元の所有者にはコンテンツ アイテムへの特別な接続がありません。アイテムにアクセスする機能は、プロジェクトまたはその特定アイテムに対するパーミッションによって決定されます。

コンテンツ タイプごとに、所有権を変更したり受け取ったりできる人物

所有権を変更したり受け取ったりできるかどうかは、次の表に記載しているように、パーミッションとコンテンツ資産との関係によって異なります。

注: 完全なプロジェクト リーダー アクセス権は一部のサイト ロールでのみ利用可能です。詳細については、プロジェクト レベルの管理を参照してください。

コンテンツ資産のタイプ	所有権を変更できる人物	所有権を受け取ることのできる人物
上位プロジェクト	• サーバー管理者[1] • サイト管理者	• サーバー管理者 • サイト管理者 • Creator • Explorer (パブリッシュ可能)
子 (ネストされた) プロジェクト	• サーバー管理者 • サイト管理者 • プロジェクト所有者	Explorer および Viewer を除く管理者または所有者
ワークブックとデータ ソース	• サーバー管理者 • サイト管理者 • ワークブックまたはデータ ソースの所有者 • このアイテムを含むプロジェクトのプロジェクト リーダーまた	• サーバー管理者 • サイト管理者 • Creator • Explorer

	は所有者	
メトリクス (従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。)	• サーバー管理者 • サイト管理者 • メトリクスの所有者 • このアイテムを含むプロジェクトのプロジェクトリーダーまたは所有者	Explorer および Viewer を除く、サイトの管理者またはユーザー
フロー	• サーバー管理者 • サイト管理者 • フロー所有者 (2025.2 以降)	• サーバー管理者 • サイト管理者 • Creator **注**: バージョン 2025.1 以前では、サーバー管理者およびサイト管理者は所有者を自分自身にのみ変更できます。
データの役割	• サーバー管理者 • サイト管理者 • データの役割の所有者 • このアイテムを含むプロジェクトのプロジェクトリーダーまたは所有者	サイトの管理者またはユーザー (Explorer および Viewer を除く)
コレクション	• サーバー管理者 • サイト管理者 • コレクションの所有	• サーバー管理者 • サイト管理者 • Creator

	者	• Explorer • Viewer
仮想接続[2]	• サーバー管理者 • サイト管理者 • 仮想接続の所有者	• サーバー管理者 • サイト管理者 • Creator

[1] サーバー管理者サイトロールは、Tableau Server のみに適用され、Tableau Cloud には適用されません。

[2] 仮想接続には データ管理 が必要です。詳細については、「データ管理 について について」を参照してください。仮想接続を編集するには、データベースの認証資格情報が必要であることに注意してください。

コンテンツ所有権の変更に関する考慮事項

- Tableau Server からユーザーを削除する前に、それらのユーザーはコンテンツ資産を所有していないことを確認します。
  - ユーザーがコンテンツを所有する場合、まずそれらの資産の所有権を再割り当てすると、そのユーザーを削除できます。そのようにしないと、、サイトロールは **ライセンスなし** に設定されますが、それらのユーザーは削除されず、管理者のみがそのコンテンツに対する特定の対応を取ることができます。ユーザーのサイトロールを「ライセンスなし」に設定する前、またはユーザーを削除する前に、埋め込みの認証資格情報があるワークブックまたはデータソースの所有権を再割り当てします。
  - 参照元データへの接続に使用する埋め込みの認証資格情報を含むワークブック、データソース、フローの所有権を変更した場合、埋め込みの認証資格情報は削除され、接続するにはもう一度認証し直す必要があります。パブリッシュされたデータソースへの接続は、元のフローの所有者を使用して認証され、パーミッションに基づいて承認されます。新しい所有者は、そのデータソースに対する読み取り/書き込みパーミッションが必要です。そうでなければ、フローでデータソースへの接続またはデータソースへの書き込みを試行しても失敗します。
  - フローへの入力または出力データソースの所有権を新しい所有者に移すか、データソースに対するパーミッションを更新する必要があります。
  - Tableau Server の接続情報を編集すると埋め込み認証資格情報を更新できます。詳細については、「接続の編集」を参照してください。フローの場合、Tableau

Prep の Web 作成でフローを開いて編集し、新しい認証資格情報を使用してフローをもう一度パブリッシュできます。

- 所有権を移せるのはパブリッシュされたフローのみです。フローの下書きの所有権は変更できません。
- パーミッションをプロジェクトにロックしない場合、コンテンツの所有権の供与対象であるユーザーが、所有権の変更の際に、パーミッションのガイドラインやアカウントのパーミッションを理解していることを確認してください。ロックしたプロジェクトでは既定で、コンテンツ所有者はコンテンツにパーミッションを設定できます。詳細については、パーミッションを参照してください。
- メトリクスの所有者を Viewer または Explorer のサイトロールを持つユーザーに変更することは可能ですが、それを行うと、メトリクスの更新が一時停止されるため、推奨されていません。メトリクスを更新、上書き、または削除するには、Creator または Explorer (パブリッシュ可能) のサイトロールが必要です。

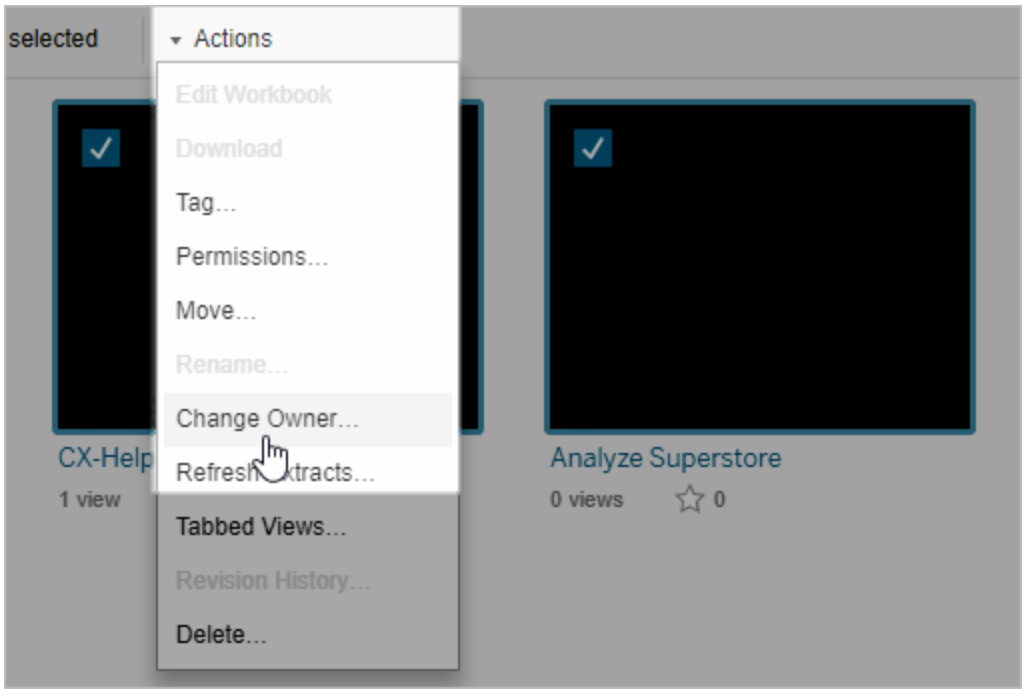
コンテンツ リソースの所有者の変更

1. Tableau Server Web 環境にサインインし、ナビゲーション メニューから **[探索]** を選択します。
2. 別の誰かに割り当てるコンテンツに移動します。
   - 複数のワークブックなど、複数の同じタイプのコンテンツを再割り当てする場合、ドロップダウン メニューからコンテンツ タイプを選択します。
   - 同一プロジェクト内で複数のアイテムを再割り当てする場合、そのプロジェクトに移動します。

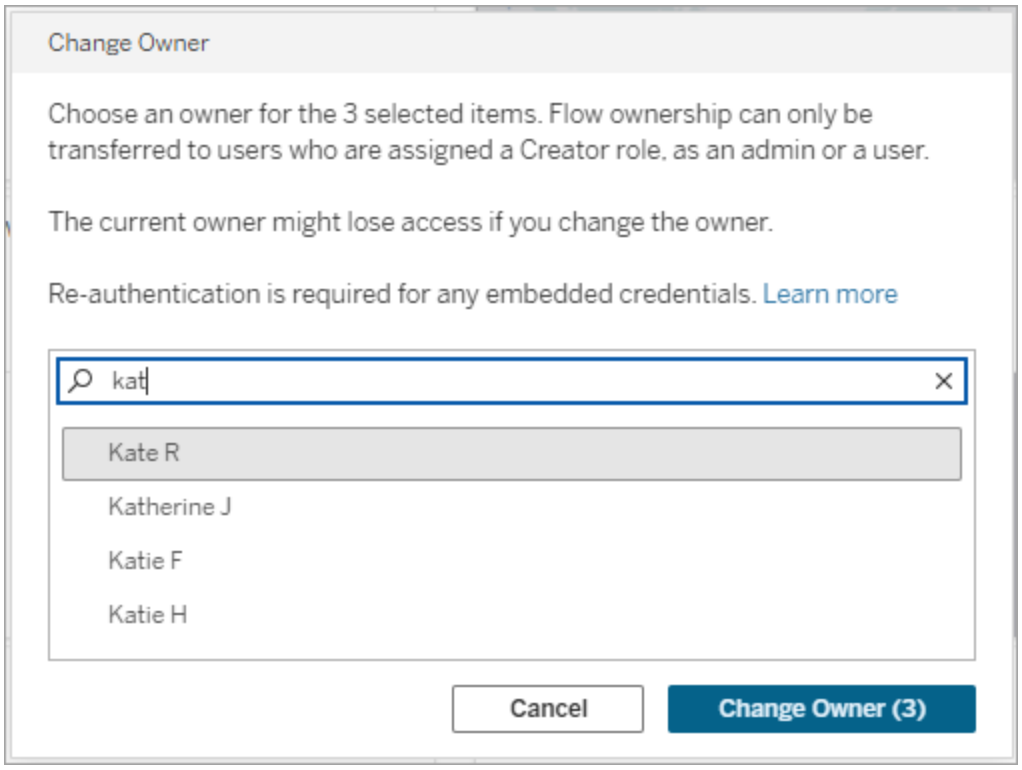
     子プロジェクトの場所がわからない場合は、フィルターを表示し、**[すべてのプロジェクトを表示]** を選択します。

   - 同じ所有者を持つ複数のコンテンツ アイテムを再割り当てする場合は、**[ユーザー]** ページでユーザーを検索します。
3. 再割り当てするアイテムを選択し、**[アクション] > [所有者の変更]** を選択します。

   他の表示されるメニュー コマンドは、コンテンツ タイプによって変わります。


selected
Actions
Edit Workbook
Download
Tag...
Permissions...
Move...
Rename...
Change Owner...
Refresh Extracts...
Tabbed Views...
Revision History...
Delete...
CX-Help
1 view
Analyze Superstore
0 views
0

4. ユーザー名を入力するか、リストからユーザーを選択します。



5. **[所有者の変更]** を選択します。

外部資産でのパーミッションの管理

Tableau Cloud および Tableau Server では、パブリッシュ済みコンテンツにアクセスして管理する領域が提供されます。Tableau Cloud または Tableau Server のライセンスをデータ管理 で設定している場合は、Tableau Catalog にアクセスできます。Tableau Catalog を使用すると、サイト全体で補助的な領域と一連の機能が追加され、サイトにパブリッシュされたコンテンツで使用される外部資産のメタデータと系列を追跡して管理することができます。

Tableau Catalog によるコンテンツと資産のインデックス化

Catalog は、Tableau Cloud または Tableau Server にパブリッシュするコンテンツのメタデータを検出、追跡、保存します。

Catalog は次のものについてメタデータをインデックス化します。

- **Tableau コンテンツ:** ワークブック、データ ソース、フロー、プロジェクト、メトリクス、仮想接続、仮想接続テーブル、ユーザー、およびサイト(従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。)。

- **外部資産:** Tableau コンテンツに関連付けられたデータベース (データ ファイルを含む) およびテーブル (テーブルのようなオブジェクトを含む)。

  Catalog は Tableau 環境の外部に由来する外部資産のメタデータを分類します。外部資産は、データベース サーバーやローカルの .json ファイルなど、多くの異なる形式で保存できます。

  Catalog は外部データのメタデータのみを追跡し、いずれの形式 (未加工や集計) の参照元データも追跡することはありません。

Catalog のメタデータには次のものが含まれます。

- **系列情報** またはアイテム間のリレーションシップ。たとえば、[Sales (売上高)] テーブルには [Superstore (スーパーストア)] データ ソースと [Superstore (スーパーストア)] サンプル ワークブックの両方とのリレーションシップがあります。

- **スキーマ情報** いくつかの例としては、次のものがあります:
  - テーブル名、列名、および列タイプ。たとえば、テーブル A には列 A、B、および C が含まれ、タイプは INT、VARCHAR、VARCHAR です。
  - データベース名とサーバーの場所。たとえば、Database_1 は SQL Server データベース (http://example.net) です。
  - データ ソース名およびデータ ソースに含まれるフィールドの名前とタイプ。たとえば、"Superstore (スーパーストア)" データ ソースには、フィールド AA、BB、および CC があります。フィールド CC は、フィールド AA とフィールド BB の両方を戻って参照する計算フィールドです。

- **ユーザーがキュレーション**、追加、または管理する情報。たとえば、アイテムの説明、認証、ユーザー連絡先、データ品質に関する警告などがあります。

Tableau Catalog のしくみ

Tableau Catalog は Tableau Cloud または Tableau Server にパブリッシュされたすべてのコンテンツをインデックス化し、系列およびスキーマのメタデータを追跡します。たとえば、メタデータは、ワークブック、パッケージ化されたワークブック、データソース、および Tableau Server や Tableau Cloud のリポジトリに由来します。

インデックス作成プロセスの一環として、パブリッシュ済みコンテンツで使用される外部アセット (データベース、テーブル、その他のオブジェクト) に関する系列やスキーマのメタデータもインデックス化されます。

注: Tableau Cloud または Tableau Server から Catalog にアクセスできるほか、Tableau メタデータ API および Tableau Server REST API からもインデックス化されたメタデータにアクセスできます。Tableau メタデータ API や REST API でのメタデータメソッドの詳細については、Tableau Server REST API の Tableau メタデータ API および メタデータ メソッドをそれぞれ参照してください。

メタデータに対するパーミッション

パーミッションは、外部アセットの表示および管理が許可されるユーザーや、系列で表示されるメタデータを制御します。

注: Tableau Cloud または Tableau Server のライセンスをデータ管理 で設定していない場合、デフォルトでは、管理者のみが Tableau メタデータ API を使用してデータベースとテーブルのメタデータを表示できます。

メタデータへのアクセス

Catalog (または メタデータ API) を使用してメタデータにアクセスするのに使用するパーミッションは、Tableau Cloud または Tableau Server よりコンテンツにアクセスするパーミッションと同じように機能しますが、系列によって公開できる機密データと外部アセットに対して付与された機能に関しては他のいくつかの考慮事項があります。

Tableau コンテンツに対するパーミッション

Catalog では、既存の Tableau コンテンツによって既に使用されているビュー機能と管理機能を使用し、Tableau コンテンツで表示できるメタデータを制御します。これらの機能の一般的な情報については、パーミッションを参照してください。

派生パーミッションを使用する外部資産に対するパーミッション

Tableau Cloud または Tableau Server のライセンスをデータ管理 で設定していない場合、Catalog はデフォルトで *派生したパーミッション* を使用し、外部アセット機能を自動的に付与します。派生パーミッションは、外部資産に対するパーミッションであり、外部資産が関連付けられている Tableau コンテンツから派生します。

[表示] 機能の場合:

- ワークブック、データソース、またはフローの所有者である場合は、そのワークブック、データソース、またはフローで *直接* 使用するデータベースと表のメタデータを表示することができます。「系列に関するその他の注意事項」を参照してください。
- プロジェクト所有者またはプロジェクト リーダーの場合、プロジェクトにパブリッシュしたコンテンツで使用されるすべてのデータベースとテーブルのメタデータを表示できます。
- 埋め込みファイルでは、外部アセット (データベースまたはテーブル) の派生パーミッションではなく、ソース コンテンツ (ワークブック、データソース、フローなど) のパーミッションを使用します。たとえば、埋め込みファイルを含むワークブックを表示できる場合は、そのワークブックで使用されている埋め込みファイルとそのメタデータを表示できます。

[上書き] と **[パーミッションの設定]** 機能の場合:

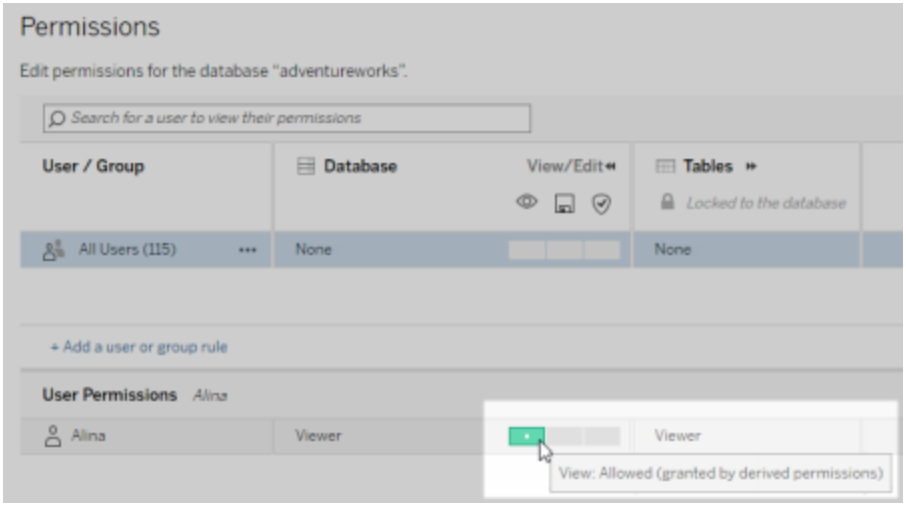
- フローの所有者は、フロー出力で使用するデータベースとテーブルのメタデータのパーミッションの編集や管理が行えます。

注: フローの場合、上記の機能は、フローの現在の所有者の下で少なくとも 1 回正常に実行された後にのみ適用されます。

パーミッションを確認する

管理者やアセットに対するパーミッションを設定する権限を持っているユーザーは、派生パーミッションを持っているユーザーを以下の手順で検証できます。

1. Tableau Cloud または Tableau Server にサインインします。
2. 左側のナビゲーション ペインで、**[外部アセット]** をクリックします。
3. ドロップダウン メニューから **[Database and Files (データベースおよびファイル)]** または **[テーブルとオブジェクト]** を選択します。
   注: .json ファイルや .csv ファイルのようなローカル ファイルは **[データベース]** で外部アセットとしてグループ化されます。
4. パーミッションを変更するデータベースまたはテーブルの横のチェック ボックスを選択し、**[アクション] > [パーミッション]** を選択します。
5. [パーミッション] ダイアログ ボックスで **[+ Add Group/User Rule (グループ/ユーザー ルールの追加)]** をクリックして入力を開始すると、グループまたはユーザーが検索されます。
6. パーミッション ルール内のグループ名またはユーザー名をクリックして、下に表示される有効なパーミッションを確認することにより、パーミッションを検証します。



外部アセットに対する派生パーミッションの優先順位

Tableau Cloud サイトまたは Tableau Server で派生したパーミッションを構成する場合、外部資産に対する各ユーザーのアクセス レベルは、関連付けられた Tableau コンテンツと、Tableau がそのコンテンツに対して使用するルールの優先順によって決まります。

Tableau では、[拒否] と評価される限り、以下のルールを順に評価していきます。いずれかのルールで [許可] と評価されると、その権限が許可され、Tableau は評価を停止します。このルール リストは、パーミッションに基づいています。

ビュー機能の場合:

1. 管理者ロール
2. ライセンス
3. プロジェクト リーダー (Tableau コンテンツ)
4. プロジェクト所有者 (Tableau コンテンツ)
5. コンテンツ所有者 (Tableau コンテンツ)
6. *派生パーミッション* (外部アセットとビュー機能にのみ適用)
    a. 管理者ロール
    b. ライセンス
    c. プロジェクト リーダー (外部アセット)
    d. プロジェクト所有者 (外部アセット)
    e. コンテンツ所有者 (外部アセット)
7. 明示的なパーミッション

上書きと**パーミッションの設定**機能の場合:

1. 管理者ロール
2. ライセンス
3. プロジェクト リーダー (Tableau コンテンツ)
4. プロジェクト所有者 (Tableau コンテンツ)
5. コンテンツ所有者 (Tableau コンテンツ)
6. 明示的なパーミッション (Tableau コンテンツ)
7. *派生パーミッション* (外部アセット、上書き、フロー出力のパーミッション機能の設定にのみ適用)

a. 管理者ロール
b. ライセンス
c. プロジェクトリーダー(外部アセット)
d. プロジェクト所有者 (外部アセット)
e. コンテンツ所有者 (外部アセット)

派生パーミッションをオフにする

管理者は、明示的なパーミッションをデータベースとテーブルに手動で付与するために、サイトでの派生パーミッションの既定の設定をオフにできます。

1. Tableau Cloud または Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[設定]** をクリックします。
3. **[全般]** タブの **[データベースおよびテーブルに関するメタデータへの自動アクセス]** で **[権限のあるユーザーにデータベースおよびテーブルに関するメタデータへのアクセス権を自動的に付与する]** チェック ボックスをオフにします。

   注: 派生パーミッションを使用してユーザーに表示される、データベースやテーブルについてのデータ品質の警告メッセージは、このチェック ボックスがオンになっていない場合でも、引き続きそれらのユーザーに表示されます。

個別の外部アセットに対するパーミッションの設定

他のユーザーが外部アセットの表示、編集 (上書き)、管理を行えるように追加のパーミッションを与えるには、ユーザーまたはグループに個々のデータベースまたはテーブルに対して管理者が権限を明示的に付与できます。

Tableau Server 2022.3 および Tableau Cloud September 2022 からは、プロジェクトで外部アセットを整理することができます。外部アセットに対する権限の継承は、「パーミッション」のトピックで説明されているように、Tableau コンテンツと同じように機能し、パーミッションの管理を簡素化することができます。

パーミッションの機能の概要

次の表に、外部アセットに設定できる機能を示します。

機能	説明	テンプレート
ビュー	データベースまたはテーブルのアセットを参照してください。	ビュー
上書き	データベースまたはテーブルのアセットデータの品質に関する警告と説明を追加または編集します。バージョン 2020.1 より前のバージョンでは、上書き機能は保存機能と呼ばれていました。	パブリッシュ
移動	データベースまたはテーブルのアセットを移動します。	管理
パーミッションの設定	データベースまたはテーブルのアセットに対してパーミッションを付与または拒否します。	管理

データベースまたはテーブルに対するパーミッションを設定する

データベースまたはテーブルに対するパーミッションを設定するには、次の手順を行います。

1. 管理者または「パーミッションの設定」権限を付与されたユーザーとして Tableau Cloud または Tableau Server にサインインします。
2. データベースまたはテーブルを検索します。データベースまたはテーブルの現在の場所がわかっている場合は、**[探索]** (Tableau Server 2022.3 および Tableau Cloud September 2022 以降) から行うことができ、すべてのデータベース、テーブル、ファイルのリストを見る場合は、**[外部アセット]** から行うことができます。
    - **[探索]** - 左側のナビゲーション ペインで **[探索]** をクリックし、データベースまたはテーブルが存在するプロジェクトの場所を特定します。
    - **[外部アセット]** - 左側のナビゲーション ペインで **[外部アセット]** をクリックします。ドロップダウン メニューから **[データベースとファイル]** または **[テーブルとオブジェクト]** を選択し

ます (注: .json ファイルや .csv ファイルなどのローカル ファイルは、データベースの外部アセットとしてグループ化されます)。

3. パーミッションを変更するデータベースまたはテーブルの横のチェック ボックスを選択し、**[アクション] > [パーミッション]** を選択します。

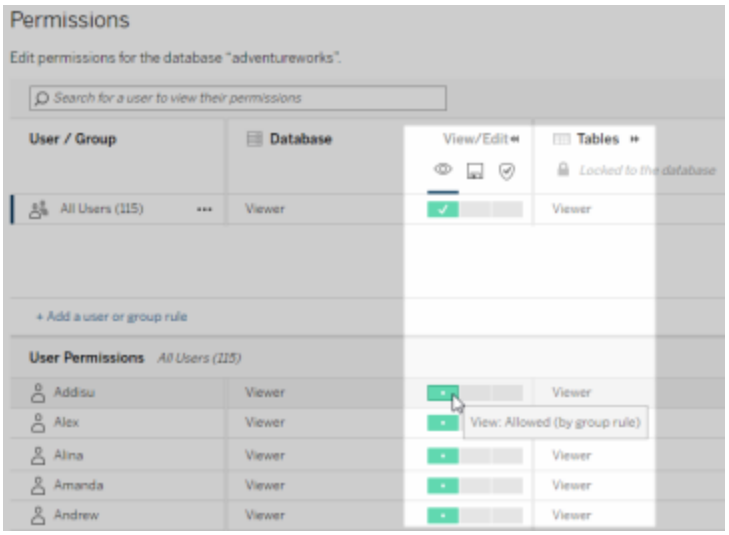
4. [パーミッション] ダイアログ ボックスで **[+ Add Group/User Rule (グループ/ユーザー ルールの追加)]** をクリックして入力を開始すると、グループまたはユーザーが検索されます。

5. グループまたはユーザーの初期機能セットに適用するパーミッション ロール テンプレートを選択してから、**[保存]** をクリックします。使用できるテンプレートは、[ビューア]、[パブリッシュ]、[管理]、[なし]、[拒否] です。

6. ルールをさらにカスタマイズするには、ルール内の機能をクリックし、[許可] または [拒否] に設定するか、指定しないままにします。操作が終了したら、[保存] をクリックします。

7. 他のグループまたはユーザー用に希望する追加ルールを構成します。

8. パーミッション ルール内のグループ名またはユーザー名をクリックして、下に表示される有効なパーミッションを確認することにより、パーミッションを検証します。



プロジェクトにない外部アセット

シナリオによっては、外部アセットがプロジェクトにない場合があります。

- **外部アセットの既定のプロジェクト**が存在する前 (Tableau Cloud 2022 年 12 月 / Server 2023.1) に Catalog によって検出された外部アセットは、それ以降にプロジェクトに移動されていない限り、プロジェクトには含まれません。
- **外部アセットの既定のプロジェクト**が存在する前 (Tableau Cloud 2022 年 12 月 / Server 2023.1) にプロジェクトが削除された外部アセットは、それ以降にプロジェクトに移動されていない限り、プロジェクトには含まれません。
- Tableau Server 2022.1 以前では、外部アセットをプロジェクト移動することはできません。

外部アセットがプロジェクトにない場合、外部アセットに対するパーミッションは、Tableau Server 2022.1 および Tableau Cloud June 2022 以前と同様に機能します。つまり、データベースとテーブルのパーミッションは、プロジェクト内のコンテンツとは別に制御され、テーブルのパーミッションは、データベースのパーミッションを通じて管理することができます。アクセス パーミッションは、このようにデータベース レベルで設定されている場合、そのデータベースで新しく検出され、インデックス付けされた子テーブルに対するテンプレートとして機能します。また、子のテーブルが常にデータベース レベルで設定したパーミッションを使用するように、データベース パーミッションをロックすることもできます。

> 注: データベースがプロジェクト内にある場合、データベースへのパーミッションをロック (またはロックを解除) することはできません。

データベースに対するパーミッションをロック (またはロックを解除) するには、次の手順を使用します。

1. 管理者または「パーミッションの設定」権限を付与されたユーザーとして Tableau Cloud または Tableau Server にサインインします。
2. 左側のナビゲーション ペインで、**[外部アセット]** をクリックします。既定では、[外部アセット] ページにデータベースとファイルのリストが表示されます。
3. パーミッションをロックするデータベースの隣にあるチェック ボックスをオンにして **[アクション] > [パーミッション]** を選択し、[Table Permissions (テーブルのパーミッション)] で **[編集]** リンクをクリックします。

4. [データベースのテーブルのパーミッション] ダイアログ ボックスで、**[Locked (ロック)]**、**[保存]** の順にクリックします。

5. パーミッションをロック解除するには、もう一度 **[編集]** をクリックし、**[Customized (カスタマイズ)]** を選択します。

系列情報にアクセスする

Catalog (およびメタデータAPI) を使用すると、リレーションシップと依存関係のメタデータ(*系列*とも呼ばれます) をTableau Cloud やTableau Server 上のTableau コンテンツや外部アセットの間で公開できます。系列は、主に次の3 つのことを示しています。

- 項目が相互に関連している方法 (直接的または間接的)
- 相互に関連付けられている項目の数
- (適切なパーミッションがある場合) 系列内の項目に関する機密データ

系列の機密データ

場合によっては、データ品質に関する警告メッセージ、コンテンツやアセットの名前、または関連するアイテムやメタデータなどの機密データが系列に含まれることがあります。

既定では、すべてのユーザーに系列の完全な情報が表示されますが、機密データは適切なビュー機能を持たない特定のユーザーに対してブロックされます。機密データをブロックするこの概念は、難読化と呼ばれます。

難読化により、適切なビュー機能を持たない特定のユーザーに対して機密データをブロックしながら、系列内 のすべてのメタデータを表示できます。この既定の動作では、完全なインパクト分析に依存するワークフローを使用できます。

組織において系列内の機密データを難読化するだけでは不十分な場合は、その機密データを含む系統の特定の部分をフィルターすることができます。

フィルターすることによって、その機密データに対して適切な**表示**機能を持たない、特定のユーザーの系列 (およびデータの詳細などの系統に関連する部分) が除かれます。フィルターによって系列の一部が省略されるため、この場合は完全なインパクト分析に依存するワークフローが防止されます。

機密データの処理方法を変更するには、次の手順を実行します。

1. Tableau Cloud または Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[設定]** をクリックします。
3. [全般] タブの **[系列の機密情報]** で、Tableau Cloud サイトや Tableau Server のすべてのユーザーに対して系列情報を最も適切に処理するラジオ ボタンを選択します。

系列に関するその他の注意事項

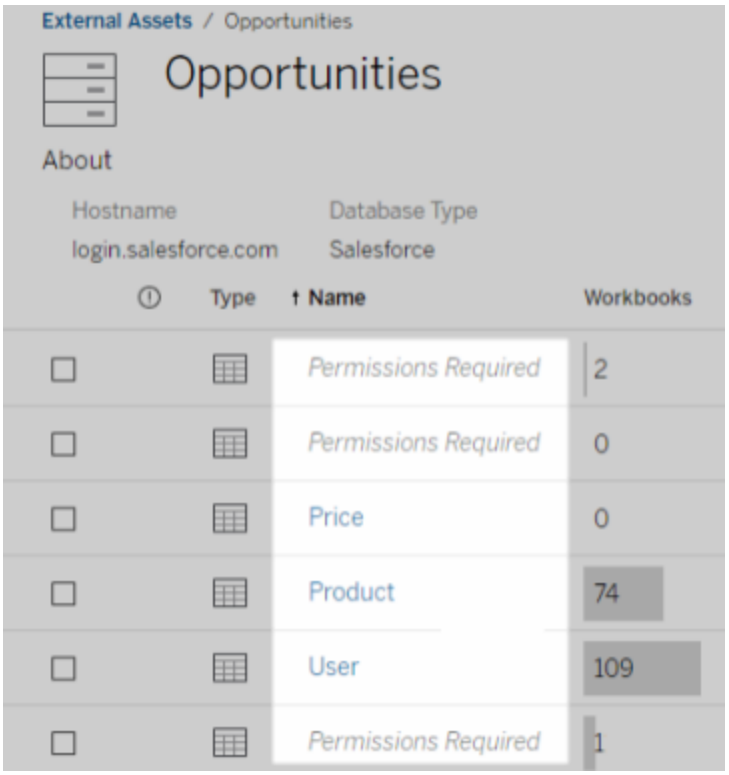
- **関連アセットに対するビュー機能がある場合**、いつどのアセットやコンテンツが相互に関連付けられるか、およびその機密メタデータを表示できます。

  たとえば、1) 関連するアップストリーム データベースとテーブルの名前、データ品質に関する警告、合計数、2) 評価対象のアセットのダウンストリーム ワークブックの系列に含まれるシート (表示および非表示) の合計数を表示できます。

- **関連アセットに対するビュー機能がない場合**、いつアセットが相互に関連付けられるかを常に表示できます。

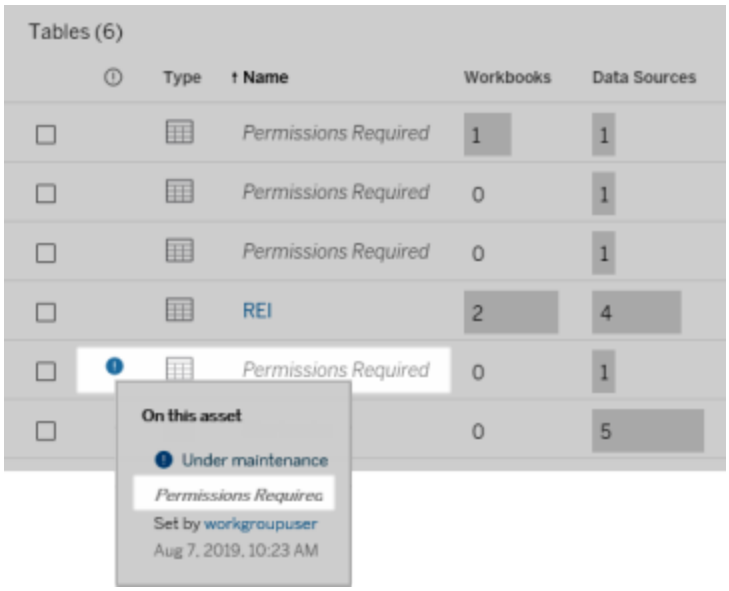
  たとえば、1) 関連するアップストリームのデータベースとテーブルが系列に存在するかどうか、および 2) 評価対象のアセットに関連するデータベースの合計数またはテーブルの合計数を表示することができます。

  ただし、ビュー機能がない場合、これらのアセットに関連付けられているメタデータを表示することはできません。パーミッションが制限されているためにメタデータがブロックされる場合や、アセットがパーソナル スペースにある場合、**[必要なパーミッション]** が表示されます。



- **関連アセットに対するビュー機能がない場合**、アセットが認証されているかどうかを常に表示できます。

  ただし、表示機能がない場合、関連するデータベースやテーブルの名前など、認証に関連する機密情報を表示することはできません。パーミッションが制限されているためにメタデータがブロックされる場合や、アセットがパーソナル スペースにある場合、**[必要なパーミッション]** が表示されます。



系列の詳細については、インパクト分析での系列の使用を参照してください。

系列データを通じて検出可能なタグに関するその他の注意事項

Tableau コンテンツに加えて、外部アセットにタグを付けることもできます。タグは常に表示されますが、系列データを通じて表示されるタグ付けされたアイテムは、難読化 (既定) にするか、このトピックで前述されているようにフィルターすることができます。

タグ付けされたアイテムが難読化されている場合:

- **タグ付けされたアイテムの表示機能がある場合**は、タグ付けされたアイテムとタグ付けされた関連アイテム、およびすべてのメタデータを表示できます。
- **タグ付けされたアイテムの表示機能がない場合**:
    - タグ付けされたアイテムとタグ付けされた関連アイテムのタイプは表示できますが、アイテムに関する機密性の高いメタデータは表示できません。たとえば、タグ フィルターを使用して、“注目に値する” というタグが付いたアイテムを表示するとします。“注目に値する” というタグが付いたデータベースのアイテムは表示されますが、タグ付けされたデータベースの名前は表示されません。

- タグ付けされた関連アイテムの数が表示されます。たとえば、“注目に値する” というタグのクエリを実行するとします。クエリは、タグが付いたデータベースを 5 つ返します。

タグ付けされたアイテムがフィルターされている場合、タグ付けされたアイテムとタグ付けされた関連アイテムが表示されるのは、アイテムの表示機能がある場合のみに限定されます。

タグの詳細については、Tableau ユーザー ヘルプの「タグ アイテム」を参照してください。

資産の結果とコンテンツの結果が不一致になる可能性

Catalog に系列情報が表示される場合、Tableau コンテンツと外部アセットの情報が提供されます。Catalog の系列には常に、関連付けられたアイテムの真の数または結果が表示されます。ただし、サイトの他の領域では、表示される項目が少なくなる場合があります。これは、利用している**表示**機能が原因であるかもしれません。カタログ以外では、自分のパーミッションで許可されているコンテンツしか表示されません。

たとえば、[Superstore (スーパーストア)] データ ソースを見ているとします。「スーパーストア」データ ソースの系列には、データ ソースが接続する上流の参照元のテーブルがいくつあるか、データ ソースに依存する下流のワークブックがいくつあるかが表示されます。ただし、これらの下流にあるワークブックをすべて表示する権限がない可能性があるため、「カタログ」系列の関連するワークブックの数 (実際の合計) は、**[接続されたワークブック]** タブのワークブックの数 (表示する権限があるもの) よりも多くなる可能性があります。

パーミッションとは別の理由によって、アセットの数とコンテンツの数の不一致が生じる可能性があります。詳細については、インパクト分析での系列の使用を参照してください。

実行可能なユーザー

このトピックで説明するタスクを実行できるのは、以下のユーザー タイプです。

Tableau Cloud サイトまたは Tableau Server の管理者

データ管理	機能	要件
ライセンス認証済み	アセットとそのメタデータを表示する	なし
	アセットとそのメタデータを編集する	なし
	アセットとそのメタデータに対するパーミッションを変更する	なし
	アセットとそのメタデータを表示する権限をユーザーに付与する	**デフォルト:** "派生パーミッション" が有効なユーザーは、所有するコンテンツの外部アセットのメタデータや、ユーザーがプロジェクトリーダーまたは所有者になっているプロジェクトにパブリッシュされたコンテンツのメタデータを表示できます。 **アドホック:** 指定した外部アセットに対して、明示的な**ビュー**パーミッションを構成できます。
	アセットとそのメタデータを編集する権限をユーザーに付与する	指定した外部アセット(ユーザーがフロー所有者であるために自動的に付与されない場合) に対して、"書き込み" または**上書き**の明示的なパーミッションを構成できます。
	アセットとそのメタデータ対するパーミッションを変更する権限をユーザーに付与する	指定した外部アセット(ユーザーがフロー所有者であるために自動的に付与されない場合) に対して、明示的な「編集」または**パーミッションの設定**を構成できます。

データ管理	機能	要件
ライセンス認証なし	アセットとそのメタデータをすべて表示する	**メタデータ API にのみ適用する**
	アセットとそのメタデータを編集する	データ管理 が必要です
	アセットとそのメタデータに対するパーミッションを変更する	データ管理 が必要です
	アセットとそのメタデータを表示する権限をユーザーに付与する	**メタデータ API にのみ適用する:** 派生パーミッションは、上記の手順で有効にできます。"派生パーミッション" が有効なユーザーは、所有するコンテンツの外部アセットのメタデータや、ユーザーがプロジェクトリーダーまたは所有者になっているプロジェクトにパブリッシュされたコンテンツのメタデータを表示できます。
	アセットとそのメタデータを編集する権限をユーザーに付与する	データ管理 が必要です
	アセットとそのメタデータ対するパーミッションを変更する権限をユーザーに付与する	データ管理 が必要です

Creator または Explorer のライセンスを持つユーザー

データ管理	機能	要件
ライセンス認証済み	アセットとそのメタデータを表示する	**デフォルト:** Tableau Cloud サイト管理者や Tableau Server 管理者が「派生したパーミッション」を有効にすると、所有するコンテンツの外部ア

データ管理	機能	要件
		セットのメタデータや、プロジェクトリーダーまたは所有者になっているプロジェクトにパブリッシュされたコンテンツのメタデータを表示できるようになります。 **アドホック:** 明示的な**表示**パーミッションが付与されている外部アセットのメタデータを表示できます。
	アセットとそのメタデータを編集する	"書き込み" または**上書き**の明示的なパーミッションが付与されている外部アセットのメタデータを編集できます。(ユーザーがフロー所有者であるために自動的に付与されない場合)
	アセットとそのメタデータに対するパーミッションを変更する	明示的な "編集" または**パーミッションの設定**が付与された外部アセットに対するパーミッションを変更できます。(ユーザーがフロー所有者であるために自動的に付与されない場合)
	アセットとそのメタデータを表示するパーミッションを他のユーザーに付与する	明示的な "編集" または**パーミッションの設定**が付与された外部アセットに対するパーミッションを変更できます。(ユーザーがフロー所有者であるために自動的に付与されない場合)
ライセンス認証なし	アセットとそのメタデータを表示する	**メタデータ API にのみ適用する:** Tableau Cloud サイト管理者や Tableau Server 管理者が「派生したパーミッション」を有効にすると、所有するコンテンツの外部アセットのメ

データ管理	機能	要件
		タデータや、プロジェクト リーダーまたは所有者になっているプロジェクトにパブリッシュされたコンテンツのメタデータを表示できるようになります。
	アセットとそのメタデータを編集する	データ管理 が必要です
	アセットとそのメタデータに対するパーミッションを変更する	
	アセットとそのメタデータを表示するパーミッションを他のユーザーに付与する	

データの管理

Tableau で使用するデータに接続して管理できます。

Tableau Server データ ソース

Tableau ユーザー同士で定義したデータ接続を共有する場合は、データ ソースを Tableau Server にパブリッシュできます。データ ソースをサーバーにパブリッシュすると、他のタイプのデータと同様に他のユーザーが独自のワークブックから接続できるようになります。Tableau データ ソースのデータが更新されると、これに接続するすべてのワークブックがこの変更を取得します。

Tableau Server on Windows については、「Tableau Server データ ソース」を参照してください。

Tableau Server データ ソースは、次を説明するメタデータで構成されます。

- **接続情報**: データがライブ データベースにあるか抽出かを定義します。また、どのデータを Tableau に取り込むかについても定義します。
- **カスタマイズとクリーンアップ**: データの効率的な使用を促進するための情報が含まれます(計算、セット、グループ、ビン、パラメーター、カスタム フィールドの書式設定など)。
- **データへのアクセスや更新の手順**: 参照元となるデータベース サーバーの場所 (オンプレミスまたはクラウド内)、ファイルベースのデータのネットワーク パス、認証資格情報やアクセス トークンなどのセキュリティ情報、および関連情報が含まれます。

ユーザーが一貫した信頼性の高いデータを作成する際に便利なだけでなく、Tableau データ ソースを使用すると、管理者にもメリットがあります。複数のワークブックを 1 つのデータ ソースに接続できるため、データ ソースの急増を最小限に抑え、ストレージ容量と処理時間を節約できます。また、Tableau データ ソースに接続し、抽出接続も使用するワークブックをダウンロードすると、その抽出がサーバー上に残るため、ネットワークのトラフィック量が減ります。最後に、接続にデータベース ドライバーが必要な場合は、各ユーザーのコンピューター上ではなくサーバー上にのみドライバーをインストールして保持する必要があります。Tableau Cloud を使用している場合は、サポートされるすべてのドライバーをサイトにパブリッシュされたデータ ソースに使用できます。

データ ソースの管理

次のいずれかのアクセス レベルがある場合は、データ ソース上の一部またはすべての管理タスクを実行できます。

- サイトまたはサーバー管理者
- データ ソースをパブリッシュするプロジェクトのプロジェクト リーダーまたは所有者

  完全なプロジェクト リーダー アクセス権は一部のサイト ロールでのみ利用可能です。詳細については、プロジェクト レベルの管理を参照してください。

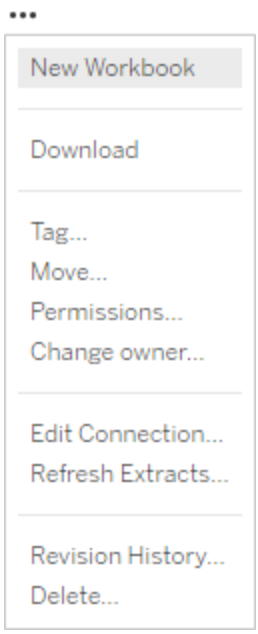
- データ ソースの所有者

サイトまたはサーバー管理者でない場合は、すべての管理タスクに対するアクセス権がありません。たとえば、[パーミッション] メニューが使用できない場合、プロジェクトのパーミッションがロックされている可能性があり、個々のワークブックおよびデータ ソース上でパーミッションの設定が拒否されます。

ヒント: すべてのデータソースを同一のガイドラインの下で維持できるよう、ベストプラクティスは任意のプロジェクトまたはサイト全体にパブリッシュされたすべてのデータソースを管理するユーザーまたはチームを指定することです。

アクセス権のある管理タスクを実行するには、次を実行します。

1. サイトにサインインし、**[コンテンツ]** タブで **[検索] > [データソース]** を選択します。
2. データソースで **[アクション]** (**...**) メニューを選択します。



- **新しいワークブックまたはダウンロード:** このデータソースに接続するブラウザー環境で新しいワークブックを作成します。または、ローカルで使用するデータソースをダウンロードします。
- **タグ:** キーワードタグを追加または削除します。タグには、単一の語句、または複数の語句をコンマで区切って含めることができます。
- **移動:** データソースをあるプロジェクトから別のプロジェクトに移動します。これを実行するには、プロジェクトごとに個別の設定が必要です。詳細については、アセットを移動するための

要件を参照してください。

- **パーミッション:** どのユーザーまたはグループがデータソースの接続、修正、ダウンロードを行えるかを指定するパーミッションを表示または更新します。このセクションの冒頭に記載されているように、このアクションが利用できない場合はプロジェクトパーミッションがロックされる場合があり、プロジェクトリーダーまたは管理者のみがパーミッションを変更できます。
- **所有者の変更:** 誰かを所有者にすると、それに対する完全なアクセス権を付与することになります。
- **抽出の更新:** データソースに抽出が含まれている場合は、この抽出を更新スケジュールに割り当てることができます。

  詳細については、定期的なデータ更新を参照してください。
- **データソースのリビジョン履歴の表示**
- **削除:** データソースを削除すると、そのデータソースに接続するワークブックに影響を与えます。データソースを削除する前に、そのデータソースに接続されているワークブックがないことを確認するか、別のデータソースを使用するようワークブックを編集します。

さらに、プロキシ接続のデータソースの場合、管理者はデータベースに対するユーザー認証がどのように行われるか、および適切なドライバーがインストールされているかどうかを常に把握できます。詳細については、データベースドライバーおよびデータセキュリティを参照してください。

制限事項

パブリッシュされたデータソースは、多くの場合、キュレーションされた信頼できるデータソースとして機能します。そのため、変更方法や使用方法に関して制限事項があります。

別名と計算は編集できません。

- 新しい別名も作成できません。フィールドを複製して、そのコピーに別名を付けることができます。
- 新しい計算を作成できます。既存の計算もコピーして編集することができます。

関係と結合は編集できません。

パブリッシュされたデータソースは結合でも関係でも使用することはできません。

- パブリッシュされたデータソースを結合する必要がある場合は、ブレンドを使用します。

.hyper 形式への抽出のアップグレード

Tableau バージョン 2018 では、Tableau 抽出に..hyper 形式を導入し、従来の .tde 形式と置き換えました。.hyper 形式は、2018 年以降、Tableau で抽出を作成する際の標準形式となっており、抽出の大部分は..hyper ファイルで作成されます。2023 年初頭から、Tableau は Tableau Cloud および Tableau Public での .tde 形式のサポートを終了します。この非推奨化の詳細については、Tableau コミュニティの投稿を参照してください。

.tde ファイルのサポート終了

2023 年以降、Tableau 抽出の .tde 形式は非推奨になりました。この形式は 2018 年に .hyper 形式に置き換えられましたが、2023 年 3 月までにアップロードされたファイルには引き続き有効です。

- この変更は、2023 年 3 月時点の Tableau Cloud および Tableau Public に対して行われました。
- Tableau Server バージョン 2023.1.0 以降、.tde 形式のファイルのアップロードは利用できなくなりました。
- 2024.2 は、.tde ベースのワークブック、データソース、ブックマークをサポートする Tableau Desktop の最後のバージョンです。バージョン 2024.3 以降では、.hyper 形式のみがサポートされます。

Tableau Desktop を使用した .tde 抽出の手動アップグレード

抽出をローカルで管理している場合、Tableau Desktop を使用して、.tde 抽出を .hyper 抽出に手動でアップグレードすることができます。

注: この機能は、Tableau Desktop のバージョン 2024.2 以前でのみ使用できます。

1. Tableau Desktop で、.tde 抽出を使用するワークブックを開きます。
2. **[データ]** メニューから抽出データソースを選択し、**[抽出]**、**[アップグレード]** の順に選択します。
3. **[ファイル]**、**[保存]** の順に選択すると、ワークブックが保存され、抽出のアップグレードも完了します。

ライブ接続を使用した .tde の手動アップグレード

.tde ファイルがライブ接続 (抽出ではない) を使用している場合は、.hyper ファイルへの更新に関する Tableau コミュニティの投稿の手順に従ってファイルをアップグレードする必要があります。ライブ接続を使用した .tde ファイルを Tableau Desktop を使用して更新することはできません。

サイトのタイムゾーンを抽出用に設定する

サイトでは、抽出ベースのデータソースの既定タイムゾーンは、協定世界時 (UTC) です。サーバー管理者は、別のタイムゾーンを設定できます。

抽出のためにサイトのタイムゾーンを設定するには次を実行します。

1. Tableau に管理者としてサインインします。
2. 設定するサイト上で、[Settings] をクリックします。
3. [Site Time Zone for Extracts] セクションで、タイムゾーンを選択して [Save] をクリックします。

計算フィールドでは、NOW() や TODAY() などの関数がタイムゾーンを調べています。抽出の詳細については、Tableau Desktop ヘルプの「データの抽出」を参照してください。

タイムゾーン設定は、抽出ベースのデータソースに使用されるだけでなく、内部抽出にも影響します。たとえば、Tableau がテキストファイルなどのファイルベースのデータソースに接続すると、抽出が内部で自動的に作成されます。Tableau は、自動的に作成された内部抽出を使用して異なるソースからデータを統合します。

Web 上での抽出の作成

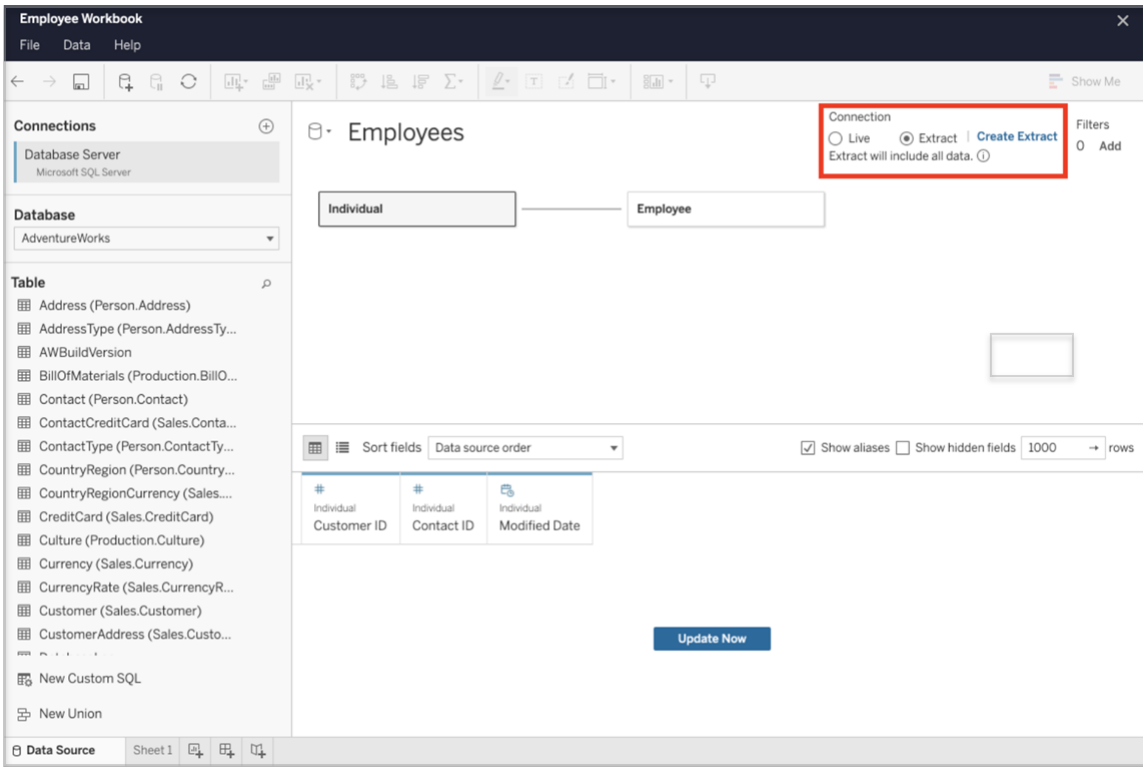
Web でデータソースを抽出して (Tableau Desktop を使用せずに)、データソースのパフォーマンスを向上させたり、追加の分析機能をサポートしたりできます。データソースを抽出すると、Tableau は、データをリモートデータストアから Tableau Server または Tableau Cloud にコピーします。デー

タを抽出するメリットの詳細については、「データの抽出」を参照してください。Web では、Web 作成またはコンテンツ サーバーで抽出を実行できます。

Web 作成で抽出を作成する

既定の抽出設定を使用して、Web 作成で抽出を直接作成できます。

Web 作成で埋め込みデータ ソースを抽出する



Web 作成で抽出を作成するには、次の手順を実行します。

ヒント: 抽出を作成する前にデータ モデルを確定することをお勧めします。抽出の作成には時間がかかる場合があり、新しい論理テーブルの追加など、データ モデルに変更を加えると、抽出が無効になります。

1. Web 作成ペインの左下隅にある **[データ ソース]** タブを選択します。新しいワークブックの場合は、**[データ ソース]** タブから開始します。
2. 右上隅で、接続タイプを **[ライブ]** から **[抽出]** に変更します。

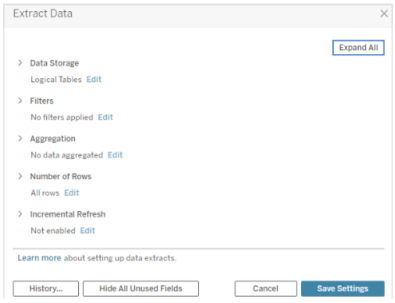
3. **[抽出の作成]** を選択します。**[抽出を作成]** ダイアログ ボックスが表示されます。

抽出の作成には時間がかかる場合があり、抽出の作成中に作成セッションを終了できます。抽出の作成が失われないようにするには、ダイアログ ボックスで **[完了時に通知する]** を選択して、抽出したワークブックを保存する場所を指定します。抽出が成功すると、ワークブックは指定した場所に保存され、Web 作成セッションを続行できることが通知されます。

抽出の作成に失敗した場合は、抽出を作成できなかったことが通知されるので、Web 作成で元のワークブックをもう一度開いて、保存されていない変更を復元できます。

抽出設定を定義する

次のいずれか 1 つまたは複数のオプションを設定すると、Tableau での保存方法、フィルター定義、および抽出のデータ量制限を設定できます。



抽出データの保存方法を決定する

2 つの構造 (スキーマ)、つまり論理テーブル (非正規化されたスキーマ) または物理テーブル (正規化されたスキーマ) のいずれかを使用し、Tableau でデータを抽出に保存させることもできます。論理および物理テーブルの詳細については、「Tableauデータ モデル」を参照してください。選択するオプションは、目的に応じて異なります。

論理テーブル

論理テーブルは、データ ソース内の各論理テーブルに対して 1 つの抽出テーブルを使用してデータを格納します。論理テーブルを定義する物理テーブルは、その論理テーブルとマージされて保管されます。たとえば、データソースが 1 つの論理テーブルで構成されている場合、データは 1 つのテーブルに格納されます。

データソースが 3 つの論理テーブル (それぞれに複数の物理テーブルを含む) で構成されている場合、抽出データは 3 つのテーブル (論理テーブルごとに 1 つずつ) に格納されます。

抽出フィルター、集計、上位 N、または非正規化データを必要とするその他の機能などの追加の抽出プロパティを使用して、抽出内のデータの量を制限したい場合は**論理テーブル**を選択します。また、データがパススルー関数 (RAWSQL) を使用する場合にも使用します。抽出データの保存に Tableau ではこの構造が既定で使用されます。

抽出に結合が含まれるときにこのオプションを使用する場合は、抽出の作成時に結合が適用されます。

物理テーブル

物理テーブルは、データソース内の各物理テーブルに対して 1 つの抽出テーブルを使用してデータを格納します。

抽出が 1 つまたは複数の等価結合と組み合わされたテーブルで構成されており、以下の物理テーブル オプションを使用する条件を満たす場合は、**[物理テーブル]** を選択します。このオプションを使用する場合、結合はクエリ時に実行されます。

このオプションにより、パフォーマンスが向上し、抽出ファイルのサイズが小さくなる可能性があります。Tableau が推奨する物理テーブル オプションの使用については、「物理テーブル オプションを使用する際のヒント」を参照してください。

このオプションは、行レベルのセキュリティの回避策として使用できる場合もあります。Tableau を使用した行レベルのセキュリティに関する詳細は、Tableau Desktop ヘルプの「データ行レベルでのアクセスの制限」を参照してください。

物理テーブル オプションを使用する条件

物理テーブルオプションを使用する抽出を保存するには、抽出に含まれるデータが以下の条件すべてを満たしている必要があります。

- 物理テーブルの結合がすべて等価 (=) 結合である
- 関係または結合に使用される列のデータ型は同一である
- パススルー関数 (RAWSQL) が使用されていない
- 増分更新が構成されていない

- 抽出フィルターが構成されていない
- 上位 N やサンプリングが構成されていない
- 抽出が物理テーブルとして保存される場合、抽出にデータを追加することはできない
- 論理テーブルでは、複数の論理テーブルが含まれる抽出にデータを追加することはできない。

注: [論理テーブル] オプションと[物理テーブル] オプションのどちらも、抽出のデータの保存方法にしか影響しません。このオプションを使用しても、[データソース] ページでの抽出の表の表示方法には影響しません。

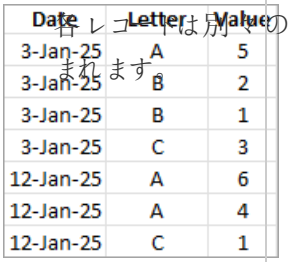
抽出するデータの量を決定する

[追加] を選択して 1 つまたは複数のフィルターを定義し、フィールドとその値に基づいて、抽出するデータ量を制限します。

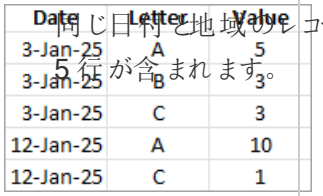
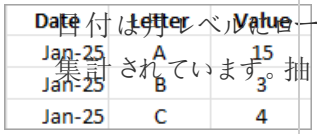
抽出のデータを集計する

[表示可能なディメンションのデータの集計] を選択し、既定の集計を使用してメジャーを集計します。データの集計により、行を統合し、抽出ファイルのサイズを最小限に抑え、パフォーマンスを向上させることができます。

データの集計を選択する場合、年、月などの指定した日付レベルに **[日付のロールアップ]** を選択することもできます。次の例は、選択できる各集計オプションでどのようにデータが抽出されるかを示します。

元のデータ	 各レコードは別々の行として表示されます。データには 7 行が含まれます。	

Date	Letter	Value
3-Jan-25	A	5
3-Jan-25	B	2
3-Jan-25	B	1
3-Jan-25	C	3
12-Jan-25	A	6
12-Jan-25	A	4
12-Jan-25	C	1

表示可能なディメンションのデータの集計 *(ロールアップなし)*	 	同じ日付と地域のレコードが 1 行に集計されています。抽出には 5 行が含まれます。
表示可能なディメンションのデータの集計 *(日付を月にロールアップ)*	 	日付は月レベルにロールアップされ、同じ地域のレコードが 1 行に集計されています。抽出には 3 行が含まれます。

抽出する行を選択する

抽出する行数を選択します。

すべての行または上位 N 行を抽出できます。Tableau では、最初にフィルターと集計が適用され、次にフィルターおよび集計の結果から指定の行数が抽出されます。行数のオプションは、抽出元のデータソースのタイプによって異なります。

注:

- サンプリングをサポートしていないデータソースもあります。そのため、[データの抽出] ダイアログ ボックスに [サンプリング] オプションが表示されない場合があります。
- [データ ソース] ページまたはシート タブで最初に非表示にしたフィールドはすべて抽出から除外されます。

増分更新設定の構成

ほとんどのデータソースで増分更新がサポートされています。抽出全体を更新する代わりに、前回データを抽出して以降の新しい行のみを追加するように抽出を構成することができます。

たとえば、新しい販売トランザクションで毎日更新されるデータソースがあるとします。その場合、抽出全体を毎日再構築する代わりに、その日に発生した新しいトランザクションのみを追加できます。更新をスケジュールするときにオプションとして増分更新を使用するには、まず設定を定義する必要があります。

定期的に完全更新を行い、最新のデータを維持することをお勧めします。

> **注:** ソースデータのデータ構造が変更された場合 (たとえば、新しい列が追加された場合)、増分更新を再び開始する前に、完全な抽出更新を実行する必要があります。

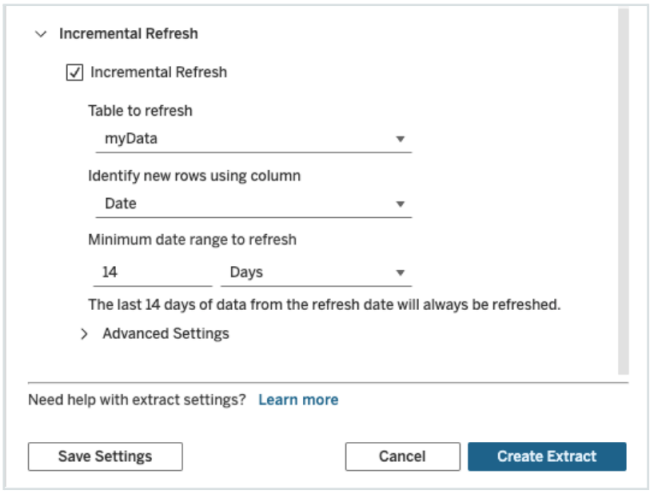
増分更新を設定する

抽出が増分更新されるように設定するには、次の手順を行います。**[行数]** セクションの **[すべての行]** を選択したことを確認します。

増分更新は、データベースのすべての行を抽出する場合にのみ定義できます。サンプル抽出を増分更新することはできません。

1. **[増分更新]** セクションを拡大します。
2. **[増分更新]** ボックスをチェックします。
3. 更新するテーブルを選択します。
4. 新しい行を識別するために使用する列を選択します。
5. (オプション) 時間の量と時間の単位を **[更新する最小の日付範囲]** フィールドに入力し、前回の更新以降に追加された新しいデータを抽出し、抽出内の既存データのサブ範囲を置き換えて、データソースで発生した可能性のある変更を取得します。このフィールドでは、秒から年まで、複数の方法で時間の長さを定義するオプションがあります。たとえば、「2」と「月」を入力すると、抽出更新ですべての新しい行が取得され、さらに過去 2 か月分のデータが更新されます。

> **注**: 抽出の更新と更新する最小の日付構成は、理解しにくい場合があります。詳細については「抽出の増分更新の設定」を参照してください。



詳細設定を使用する

[詳細設定] を展開すると、新しい行を取得する方法を設定できます。

> **注:** 更新する最小日付範囲を設定している場合は、詳細設定機能を利用できません。

[詳細設定] を使用すると、最後に記録された値以上の値を更新することによって最後に追加された行を置き換えたり、最後に記録された値よりも大きい値を含む抽出のみを更新することによって最後に追加された行を保持したりすることができます。

最初のアプローチでは、ユーザーは [日付]、[日時]、[ID] などの一意でないキー列を使用して抽出を段階的に更新することができます。

このメソッドでは、増分更新を実行するときに新しいステップが追加されます。Tableau はまず、抽出結果の行の中から、前に表示された最高値と等しい行を削除します。次に、前に表示された最高値以上のすべての行をクエリし、削除されたすべての行と新しい行をすべて取得します。

逆に、最後に追加された行を置き換えずに、最後に記録された値より大きい値を持つ行だけを追加することもできます。

終了するには、**[抽出の作成]** を選択します。

抽出更新時の考慮事項

抽出の編集:

既存の抽出を編集する場合は、前回の更新が表示されるため、正しいデータで抽出を更新していることを確認できます。

完全更新:

完全更新では、抽出を更新するたびに、すべての行が元のデータソースのデータに置き換えられます。完全更新には時間がかかるため、データベースにかかるコストが高くなる可能性があります。

日付/時刻の精度と抽出:

データエンジンは、Tableau が抽出を作成するために使用する基本的なメカニズムであり、時刻値を小数点以下 3 桁までの精度で保存します。

[列を利用して新しい行を識別] に日時またはタイムスタンプ列を指定し、データベースが Tableau よりも高い精度を使用している場合は、増分更新の後に行が重複する場合があります。

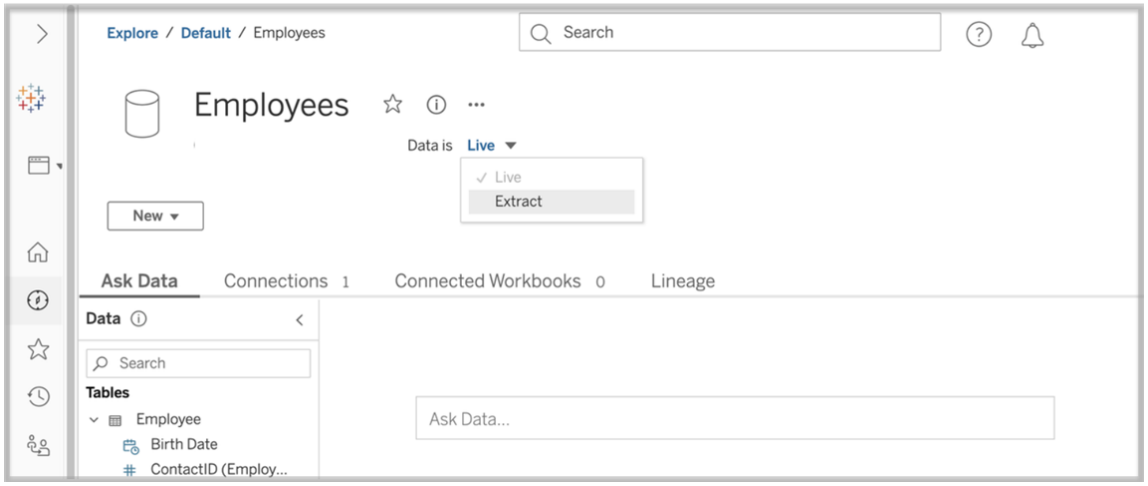
たとえば、データベースに日時の値として 2015-03-13 17:30:56.502352 と2015-03-13 17:30:56.502852 という2 つの行がある場合、Tableau は 2015-03-13 17:30:56.502 という値で両方の行を保存するため、重複行が作成されます。

制限事項

- パブリッシュされたデータソースを参照する埋め込みデータソースの抽出を作成することはできません。回避策として、パブリッシュされたデータソースで直接抽出を作成します。詳細については、「コンテンツ サーバーでパブリッシュされたデータソースを抽出する」を参照してください。
- ファイルベースのデータソースの抽出を作成することはできません。ファイルベースのデータソースには既に特別なパフォーマンス機能があり、抽出を追加してもパフォーマンス上の利点はありません。
- この機能は、Tableau Cloud のブリッジベースのデータソースには適用されません。
- カスタム SQL の制限: カスタム SQL クエリは詳細設定ではサポートされていません。カスタム SQL に依存しているユーザーが増分更新に詳細設定を使用したい場合は、アプローチを調整する必要があります。

コンテンツ サーバーで抽出を作成する

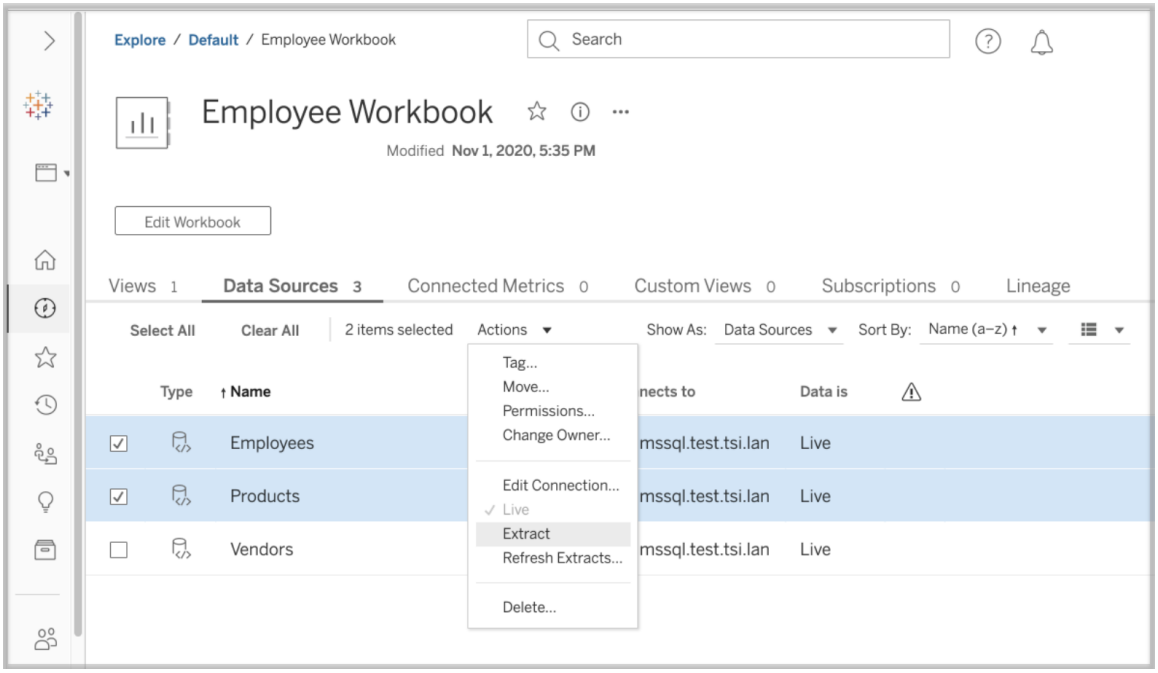
コンテンツ サーバーでパブリッシュされたデータ ソースを抽出する



パブリッシュされたデータ ソースを抽出するには、次の手順を実行します。

1. 管理者またはデータ ソースの所有者としてサインインします。
2. [コンテンツ] タブで **[探索]** を選択し、**[データ ソース]** を選択します。
3. データ ソース名をクリックして、データ ソースを選択します。
4. 画面上部の [データ ソース名] の下で、**[ライブ]** というドロップダウン メニューを選択します。
5. 接続タイプを **[ライブ]** から **[抽出]** に変更します。サイトで保存時の抽出の暗号化機能が有効になっている場合は、**[暗号化]** または **[暗号化なし]** を選択します。
6. 埋め込みの認証資格情報に関するエラー メッセージが表示された場合は、認証資格情報をデータ ソースに埋め込みます。これを行うには、**[接続の編集]** を選択します。[接続の埋め込みパスワード] を選択し、**[保存]** を選択します。

コンテンツ サーバーで埋め込みデータ ソースを抽出する



パブリッシュされたワークブックに埋め込まれている 1 つまたは複数のデータ ソースを抽出するには、次の手順を実行します。

1. 管理者またはデータ ソースの所有者としてサインインします。
2. パブリッシュされたワークブックに移動します。
3. データ ソース タブに移動します。
4. 1 つ以上のデータ ソースを選択します。
5. **[アクション]** ボタンを選択します。
6. **[抽出]** を選択します。サイトで保存時の抽出の暗号化機能が有効になっている場合は、**[暗号化]** または **[暗号化なし]**を選択します。

制限事項

- 接続認証資格情報は、データ ソースに埋め込む必要があります。
- パブリッシュされたデータソースを参照する埋め込みデータ ソースの抽出を作成することはできません。回避策として、パブリッシュされたデータ ソースで直接抽出を作成します。

- ファイルベースのデータソースの抽出を作成することはできません。ファイルベースのデータソースには既に特別なパフォーマンス機能があり、抽出を追加してもパフォーマンス上の利点はありません。
- バージョン 2023.1 以降では、ライブ接続をコンテンツサーバー上の抽出に変更すると、以前に Tableau Desktop または Web 作成で作成された抽出フィルターが削除されます。
- この機能は、Tableau Cloud のブリッジベースのデータソースには適用されません。

抽出されたデータを最新に保つ

データを抽出した後、必要に応じて抽出更新スケジュールを設定して、データを最新に保つことができます。詳細については、定期的なデータ更新を参照してください。

抽出の監視と管理

サーバー管理者は、**[抽出でのバックグラウンドタスク]** 管理ビューで抽出の作成を監視できます。詳細については、抽出でのバックグラウンドタスクを参照してください。

サーバー管理者は、[ジョブ] ページで抽出を管理できます。詳細については、Tableau Server でのバックグラウンドジョブの管理を参照してください。

抽出の作成ジョブ (抽出更新ジョブなど) には、タイムアウトする前の最大クエリ制限があります。これは、ジョブが永久に実行され、無制限のサーバーリソースを使用することを防ぐためです。抽出クエリ制限のタイムアウトは、サーバー管理者が TSM コマンドラインインターフェイスの構成設定 `backgrounder.querylimit` を使用して設定できます。詳細については、tsm configuration set のオプションを参照してください。

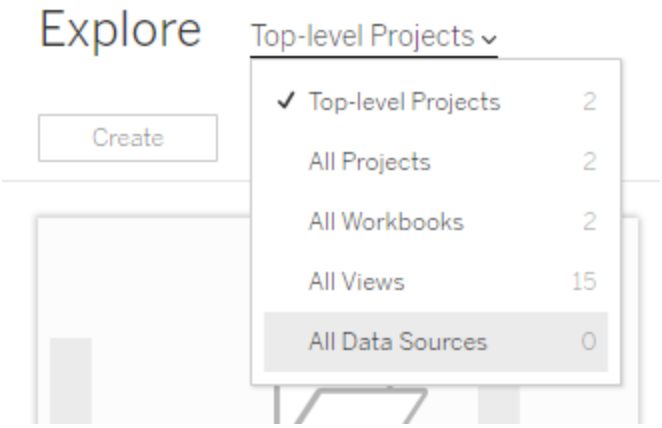
サーバー管理者は、Web 作成を管理できます。詳細については、サイトの Web 作成アクセスと機能の設定を参照してください。

データソース属性の表示

Tableau Server の Web 作成環境の **[コンテンツ]** エリアでは、ビューをフィルターすることにより、データソースまたは接続とその属性のみを表示することができます。

名前別にデータソースを表示

データソース名別にフィルターするには、**[検索]** で **[データソース]** を選択します。

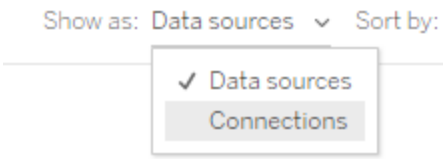


データソース名リストの属性には、次が含まれます。

- **タイプ -** データソース名の横のアイコンは、そのデータソースがライブ接続としてパブリッシュされているか()、抽出接続か()、またはワークブックに埋め込まれているか() を示します。
    - パブリッシュされたデータソースの名前を選択してそのデータソースページとタブを開き、接続や接続元のワークブックを表示します。
    - 埋め込みデータソースの名前を選択して関連付けられたワークブックを開き、そのワークブックに接続されているその他のデータを確認するタブを表示します。
- **接続先** — データベースサーバーまたはデータファイルの名前を示します。これは、Tableau Server 以外のデータベース、抽出、またはパブリッシュされたデータソースです。
- **ライブまたは前回の抽出 -** この列は、データへの接続がライブか、抽出への接続か、抽出が最後に更新されたのはいつかを表示します。

接続のリストの表示

接続タイプ別にフィルターするには、**[表示]** リストで **[接続]** を選択します。



接続属性には、次が含まれます。

- **接続先** — データベース サーバーまたはデータ ファイルの名前を示します。これは、Tableau Server 以外のデータベース、抽出、またはパブリッシュされたデータ ソースです。
- **接続タイプ** — データ型を表示します。**Tableau Server** は、接続先がサイト上でパブリッシュされたデータ ソースであることを示します。**Tableau Data Engine** とは、データ ソースが抽出を Tableau データ サーバーに抽出を保存したことを意味します。

データを最新に保つ

このセクションのトピックでは、データを手動で更新する方法と、データ更新をスケジュールする方法について説明します。

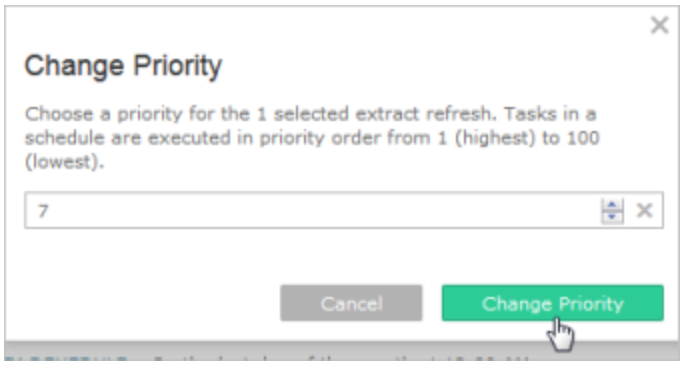
更新タスクの管理

管理者は、他のサーバー タスクに対するスケジュールされた抽出更新の優先度の変更、手動での抽出の更新、またはスケジュールの削除を実行できます。

1. 管理対象のスケジュールがあるサイトにサインインし、**[タスク]** をクリックします。
2. スケジュールされた抽出の更新を 1 つ以上選択します。
3. **[アクション]** メニューから以下のいずれかを実行します。
    - **[スケジュールの変更]** を選択し、リストから新しいスケジュールを選択します。
    - **[今すぐ実行]** を選択して手動で更新を実行します。

注: スケジュールされたタスクが抽出にない場合、[データ接続] ページからオンデマンドで更新できます。

- **[優先順位の変更]** を選択し、1 から 100 の数字を入力して、抽出を優先順位リスト内で上下に移動させます。



- **[削除]** をクリックし、選択されたデータソースのスケジュールを完全に削除します。

関連項目

抽出更新スケジュールと失敗通知を有効にする

定期的なデータ更新

パブリッシュされた抽出データソースや、抽出に接続するパブリッシュされたワークブックについては、更新タスクのスケジュールを設定できます。新しいスケジュールは、Tableau Server 管理者が **[スケジュール]** ページで作成できます。詳細については、「スケジュールの作成または変更」を参照してください。

フロー出力を更新する方法については、「フロー タスクのスケジュール」を参照してください。

不要なスケジュールされた更新を回避するには、抽出をオンデマンドで更新するパーミッションをユーザーに付与することを検討してください。詳細については、「ワークブックまたはデータソースの抽出の更新」を参照してください。

1. Tableau Server にサインインしたら、左側のナビゲーション ペインから **[検索]** を選択し、更新するコンテンツのタイプに応じてドロップダウン メニューから **[すべてのワークブック]** または **[す**

べてのデータ ソース] を選択します。

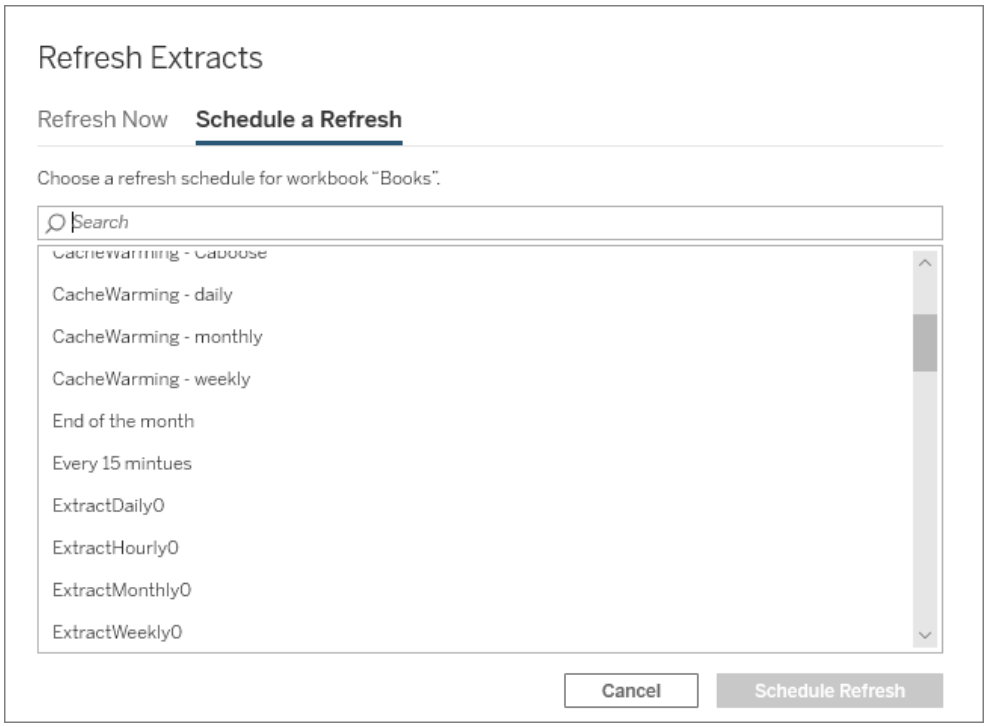
2. 更新するワークブックまたはデータ ソースのチェック ボックスを選択し、**[アクション] > [抽出の更新]** を選択します。

3. [抽出の更新] ダイアログで **[更新のスケジュール]** を選択し、次の手順を実行します。

   - 必要なスケジュールを選択します。
   - 利用可能な場合は、完全更新または増分更新を指定します。

     既定では、完全更新が実行されます。増分更新を使用できるのは、抽出をパブリッシュする前に Tableau Desktop または Tableau Server の Web 作成で増分更新を設定している場合のみです。詳細については、Tableau ヘルプの「抽出の更新」を参照してください。

   - **[更新のスケジュール]** をクリックします。



クイック スタート: スケジュールに基づく抽出の更新

データ抽出に接続しているパブリッシュされたワークブックの場合、接続されているすべてのワークブックで常に最新のデータを表示するため、反復的なスケジュールで抽出を更新するようサーバーを設定できます。

スケジュール更新するには、管理者またはデータ所有者パーミッションが必要です。

> **注:** このトピックは、Tableau Server にパブリッシュされる抽出に適用されます。Tableau Cloud での抽出の更新方法は、接続する参照元データによって異なります。詳細については、「データを最新に保つ」を参照してください。

1 サーバー上でのスケジュールの設定

サーバーにサインインし、**[スケジュール]** ページに移動してから、**[新規 スケジュール]** をクリックします。

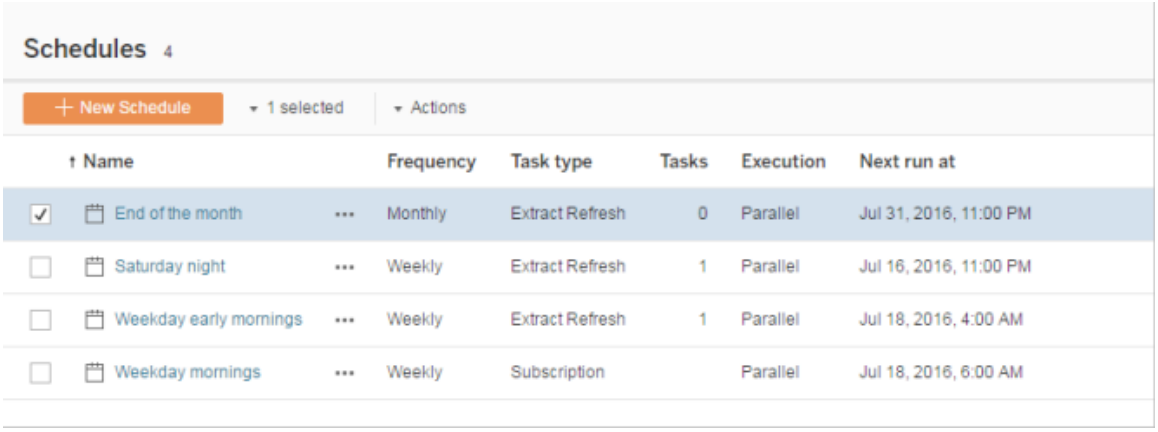


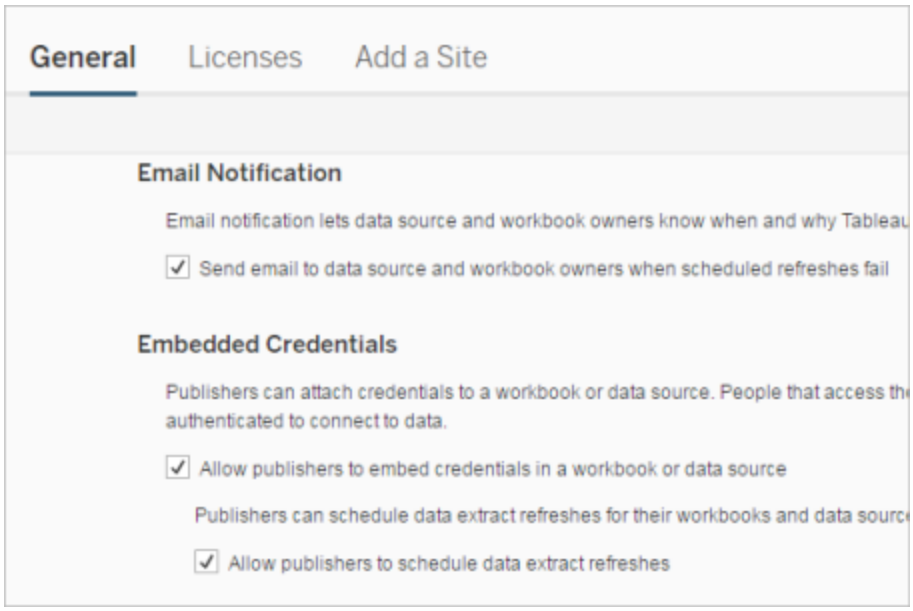
Tableau はいくつかの更新スケジュールを提供します。必要な追加スケジュールを作成します。

2 スケジュールに基づく抽出の更新と失敗メールを有効にする

サーバー管理者またはサイト管理者として、スケジュールや、抽出更新が失敗した場合のメール通知を有効にすることができます。

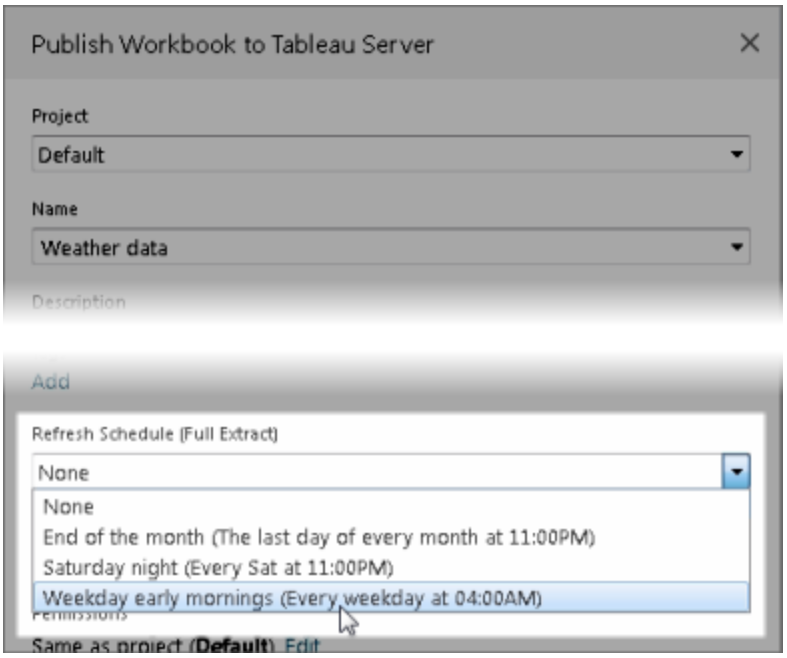
[設定] を選択してから、**[全般]** ページに移動します。

- [メール通知] で **[スケジュールされた更新に失敗した場合、データ ソースおよびワークブックの所有者にメールを送信する]** を選択します。
- **[埋め込み認証資格情報]** の下で、両方のチェック ボックスをオンにし、パブリッシャーに対して認証資格情報の埋め込みと抽出更新のスケジュールを許可します。



3 抽出を含むワークブックのパブリッシュ

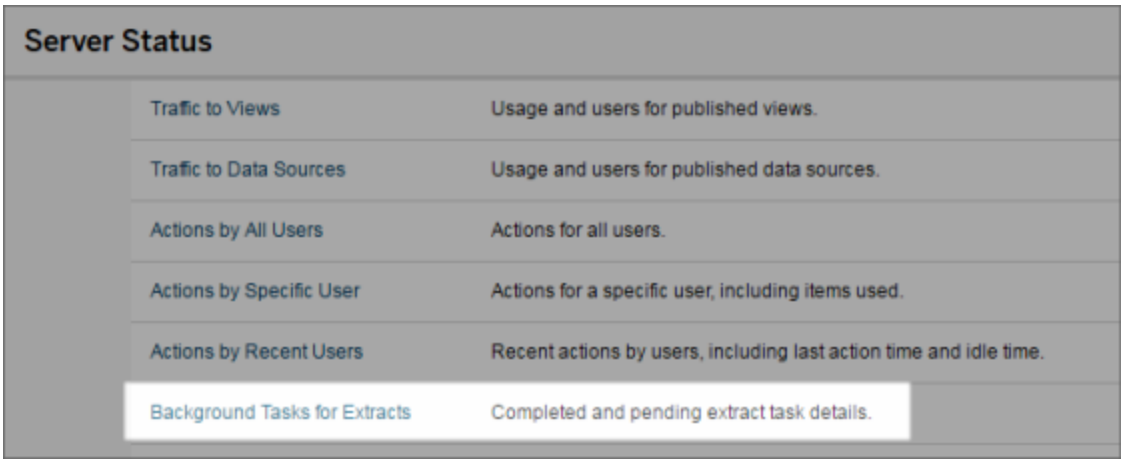
Tableau Desktop で、**[サーバー] > [ワークブックのパブリッシュ]** を選択します。まだの場合はサーバーにサインインします。**[Tableau Server にワークブックをパブリッシュ]** ダイアログ ボックスで、**[スケジュールと認証]** をクリックします。**[抽出スケジュール]** で、リストからスケジュールを選択します。



元のデータで認証が必要な場合、それらのデータへのアクセス方法も選択する必要があります。

4 更新パフォーマンスの監視

[ステータス] ページの **[抽出のバックグラウンド タスク]** を表示してスケジュールされたタスクを監視できます。



更新タスクの自動化

抽出更新タスクを Tableau Server のスケジュールに関連付けて、抽出の更新を自動化できます。また、tabcmd (Tableau Server と共に使用するためダウンロードできるコマンド ライン ユーティリティ) を使用して抽出の更新を自動化することもできます。特に、独自のスクリプト内の他のコマンドと組み合わせて、`refreshextracts` コマンドを使用できます。例は次のとおりです。

```
tabcmd login - http://mytabserver -u jsmith -p P@ssw0rd!
refreshextracts --datasource salesq4
```

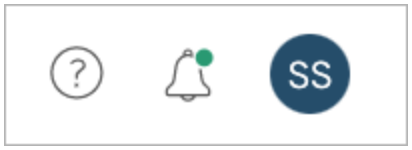
注: 抽出の更新が Tableau 10.4 以前で作成された抽出 (つまり、.tde 抽出) で実行されると、抽出は自動的に .hyper 抽出にアップグレードされます。.hyper 抽出にアップグレードする利点は多数ありますが、ユーザーは Tableau Desktop の以前のバージョンでは抽出を開けなくなります。詳細については、.hyper 形式への抽出のアップグレードを参照してください。

tabcmd ユーティリティのダウンロードに関する詳細については、tabcmd を参照してください。

抽出の更新アラートの処理

Tableau Server がスケジュールされた更新を完了できない場合は、更新が失敗したことを示すアラートが表示されます。スケジュールされた更新が 5 回連続して失敗すると、Tableau Server は更新を一時停止します。更新が一時停止されると、ユーザーが障害の原因を修正するアクションを実行するまで、Tableau Server は再試行しません。

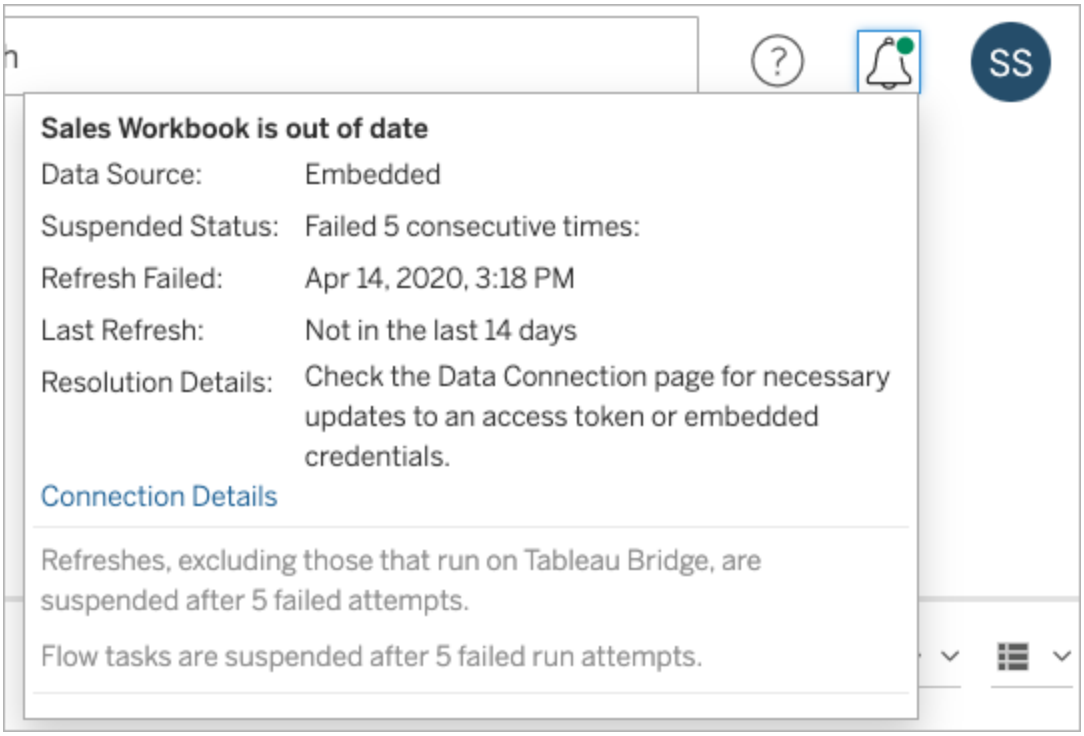
注: 更新の連続失敗回数は既定で 5 回に設定されていますが、Tableau Server 管理者によって、`backgrounder.failure_threshold_for_run_prevention` オプションを使用して変更できます。詳細については、tsm configuration set のオプションを参照してください。



[アラート] メニューは、抽出更新に失敗し、ユーザーが次の場合にのみ表示されます。

- システム管理者またはサイト管理者。
- 更新できなかったワークブックまたはデータソースの作成者。
- 更新できなかったデータソースに接続するワークブックの作成者。

[アラート] メニューを開くと、更新失敗について、次のような詳細情報が表示されます。



データソースが**埋め込み**として列挙されている場合、データソースの定義 (データソース認証資格情報またはデータベース名のようなものを含む) が、Tableau Desktop に作成された元のワークブック自体の中に埋め込まれている、またはあることを表します。

データソース名またはワークブック名が**データソース** (たとえば、**データソース: sales_data**) として列挙されている場合、データソースが Tableau Server データソースであることを表します。データソースの定義は Tableau Server にあります。

Tableau Desktop の [Data] ペインでは、データ ソースが Tableau Server 上にあるか、ローカルかどうかを判断できます。データ ソースがサーバー上にある場合、データベース アイコンの代わりに Tableau のアイコンがデータ ソース名の横に表示されます。

抽出更新の問題の解決

更新に関する問題を解決するため、アラートに示された原因に基づいて、次のアクションを実行できます。

- **アクセス トークンの検証またはユーザー認証資格情報に関するエラー**

  アラート内の **[接続の詳細]** をクリックし、いくつかの抽出更新の問題を解決することができます。問題があるデータ ソースの横にあるチェック ボックスを選択し、**[アクション] > [接続の編集]** をクリックしてから、不足している情報を入力します。完了したら **[保存]** をクリックします。接続情報が更新されると、Tableau Server は更新スケジュールを再起動します。

  ワークブックまたはデータ ソースを Tableau Desktop からパブリッシュしたときに、元々認証資格情報またはその他のデータ接続情報が埋め込まれている場合は、ワークブックまたはデータ ソースを再度パブリッシュすることもできます。パブリッシュ プロセスの一部として、新しい更新スケジュールを設定することができます。新しいスケジュールを選択していない場合、Tableau Server は既存のスケジュールを再起動します。

- **データベースが到達不能であることを示すエラー**

  データベースがオンラインで、サインインしてデータにアクセスできることを確認します。アラート内の **[再試行]** リンクを使用して、更新スケジュールを再起動できます。

- **ユーザー フィルターまたは偽装を使用している場合のエラー**

  Tableau ナレッジ ベースを参照してください。

データ接続を編集しても問題を修正することができない場合、Tableau Desktop で問題を解決しワークブックを再度パブリッシュする必要があります。

ヒント: 管理者は **[データ接続]** ページでいつでもデータ接続を編集できます。このページには、**[コンテンツ]** タブと[データ接続] をクリックして各サイトからアクセスできます。

非アクティブなワークブックとデータソースの抽出更新を自動的に一時停止する

リソースを節約するために、Tableau では、非アクティブなワークブックとパブリッシュされたデータソースの抽出更新を自動的に一時停止できます。この機能は、完全抽出更新にのみ適用され、増分抽出更新には適用されません。また、週に 1 回以上実行される更新タスクのみに適用されます。毎週または週 1 回未満の更新タスクには適用されません。

サーバー管理者は、抽出更新タスクが一時停止されるまでのワークブックの非アクティブ時間の長さを構成できます。詳細については、以下の「抽出更新の一時停止機能の構成」を参照してください。

注: データソースの抽出更新の自動一時停止は、Tableau Server バージョン 2023.3 からサポートされています。

ワークブックで以下のイベントのいずれかが発生した場合、ワークブックの非アクティブ カウントダウンタイマーがリセットされます:

- ワークブック シートの表示
- ワークブックにデータドリブン アラートまたはサブスクリプションが設定されている
- ワークブックのダウンロード
- ワークブックの場所の移動または所有者の変更

パブリッシュされたデータソースの場合、データソースからデータをフェッチするイベントが発生すると、非アクティブ カウントダウン タイマーがリセットされます。これらのイベントには以下が含まれます。

- データソースに接続されているワークブック ビューの読み込み
- データソースの [データに聞く] ページへのアクセス
- Tableau Desktop からデータソースへの接続

抽出更新の一時停止機能の構成

1. Tableau Server にサーバー管理者としてサインインします。
2. サイトの [設定] ページの [全般] タブに移動します。
   - サイトが 1 つの場合は、ブラウザー ウィンドウの上部にある **[設定]** をクリックしてから、**[全般]** をクリックします。
   - 複数のサイトがある場合は、構成するサイトを選択してから、**[設定]** をクリックし、**[全般]** をクリックします。
3. **[全般]** ページで、次の手順を実行します。
   - **[Automatically Suspend Extract Refresh Tasks (抽出更新タスクを自動的に一時停止する)]** で、**[Automatically suspend extract refresh tasks for inactive workbooks and datasources (非アクティブなワークブックとデータ ソースの抽出更新タスクを自動的に一時停止する)]** チェック ボックスをオンにします。
   - 抽出更新タスクを一時停止する前に、ワークブックを非アクティブにする日数 (7 ~ 100 日) を指定します。デフォルトは 32 日です。
   - **[保存]** をクリックします。

通知

抽出更新スケジュールが一時停止される 3 日前に、メール通知が送信されます。

抽出更新スケジュールが一時停止されると、別のメール通知が送信されます。

一時停止された抽出更新の再開

一時停止された抽出更新は、誰かがワークブックを使用すると自動的に再開されるわけではありません。サーバー管理者またはサイト管理者が手動で再開する必要があります。

一時停止された抽出更新を表示および再開するには、次の手順を実行します。

1. 管理者としてサイトにサインインし、**[タスク]** をクリックします。
2. **[抽出更新]** タブをクリックします。
3. 1 つまたは複数のアイテムを選択します。

4. **[アクション]** メニューで、**[再開]** を選択します。

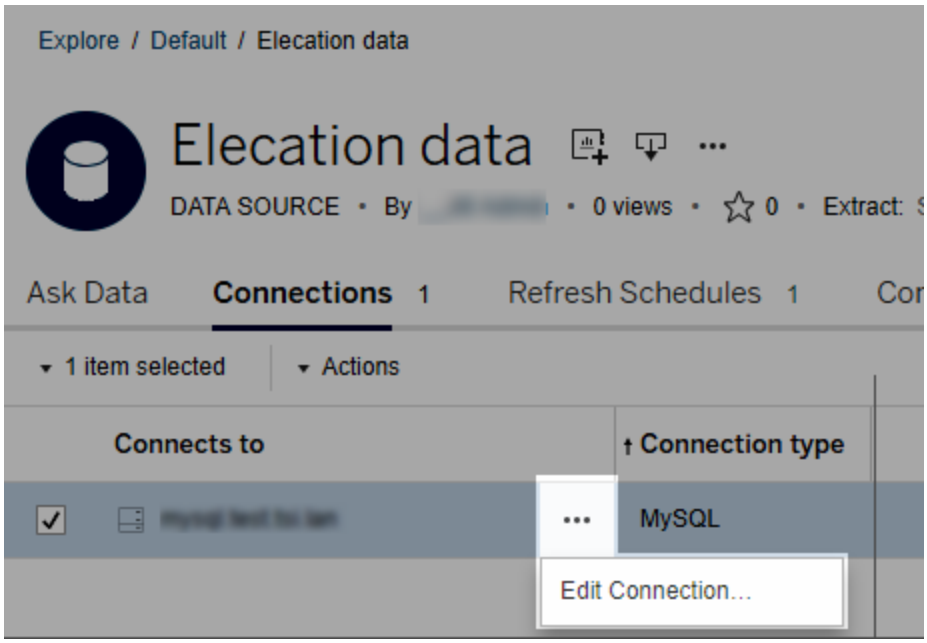
Tableau Server での接続の編集

管理者とデータソース所有者は、どのようにしてパブリッシュされたデータソースを参照元データに接続するかを説明する情報を管理できます。この情報には、データベース サーバー名またはアドレス、サーバー ポート、データベース ユーザー名、およびパスワードが接続に埋め込まれているかが含まれます。

注: 接続を編集できるかどうかは主に、データソースにおけるパーミッションではなくサイト ロールによって決定します。接続を編集するには、サイト ロールが**サーバー管理者**、**サイト管理者**、または **Creator** である必要があります。サイト ロールが **Creator** の場合、データソースの所有者でもある必要があります。

1. 変更対象のデータソースがあるサイトにサインインし、**[コンテンツ]** タブで **[検索] > [データソース]** を選択します。
2. 接続を更新するデータソースの名前を選択します。

   フィルターを表示し、データソースを検索する、またはデータソース リストの範囲を狭めます。**[サーバー名]** および **[データベース ユーザー名]** フィールドに入力した値は、正規表現として処理されます。
3. **[接続]** ビューでデータソースの **[アクション]** (**...**) メニューを選択し、**[接続の編集]** を選択します。

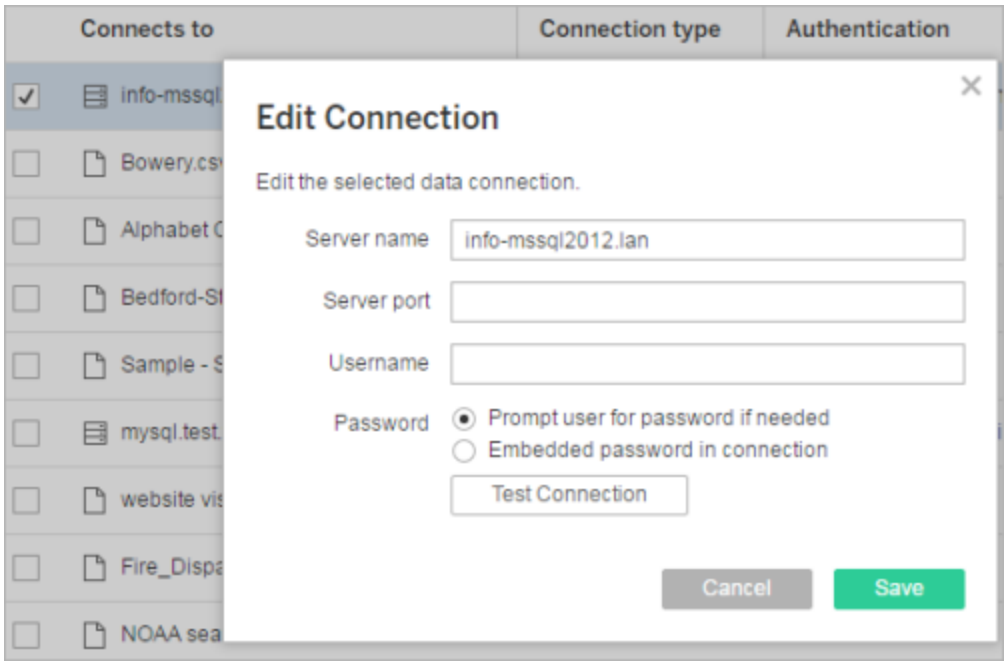


複数のデータソースを編集するには、編集するデータソースを選択し、**[アクション]** メニューと**[接続の編集]**をクリックします。

4. 接続情報を更新します。

   サーバー名では、IP アドレスを使用する場合、データベースやドライバーがその接続タイプをサポートしていることを確認します。そうでない場合は、データベースサーバー名を入力します。

   本トピックの後半のGoogle、Salesforce、および WDC データの認証タイプも参照してください。



注: バージョン 2021.4 以降では、[Prompt user for password if needed (必要に応じてユーザーにパスワードを要求する)] が選択されている場合、**[Test Connection (接続のテスト)]** ボタンは無効になります。

5. **[保存]** をクリックします。
6. ブラウザー ページを更新して変更内容を反映させます。

Google、Salesforce、および WDC データの認証タイプ

Google BigQuery、Google Analytics、Salesforce.com、および多くの Web データ コネクター (WDC) 接続は、“そのまま” のユーザー名とパスワード認証資格情報の代わりにセキュアなアクセス トークンを使用する、OAuth 認証標準を使用します。OAuth では、データベースの認証資格情報は Tableau に保存する必要はありません。そのデータ ソースへの接続するワークブックを作成または編集する Tableau Desktop ユーザーを含め、すべてのユーザーは、このアクセス トークンを介して接続します。

次のセクションでは、Google および Salesforce の接続オプションについて説明します。Web データコネクターのオプションはさまざまですが、アクセス トークンを確立するには、いずれの場合もプロバイダーの Web ベース サインイン フォームでサインインする必要があります。

Google 認証オプション

Google BigQuery または Google Analytics 接続を編集すると、**[接続の編集]** ダイアログ ボックスで次のオプションのいずれかを選択できます。

- 指定されたアカウントを介して認証するために **[Google BigQuery (または Google Analytics) 認証資格情報を接続に埋め込む]** を選択し、リストから既存のアカウントを選択するか、**[今アカウントを認証する...]** を選択して新しいアカウントを追加します。

  新しいアカウントを追加する場合、Google サインイン ページが表示されます。データベースの認証資格情報を提供すると、Google はデータへの Tableau アクセスを確認するようプロンプトを表示します。**[同意する]** をクリックすると、Google はデータへ接続するために使用するアクセス トークンを返します。

  > **注**:Google データ ソースの抽出を作成する場合、更新タスクをスケジュールできるようこの最初のオプションを選択します。

- **[Google BigQuery/Analytics 認証資格情報に関するメッセージをユーザーに表示]** を選択すると、ユーザーは自分の個人用アクセス トークンを介して接続するか、接続するたびに毎回サインインする必要があります。

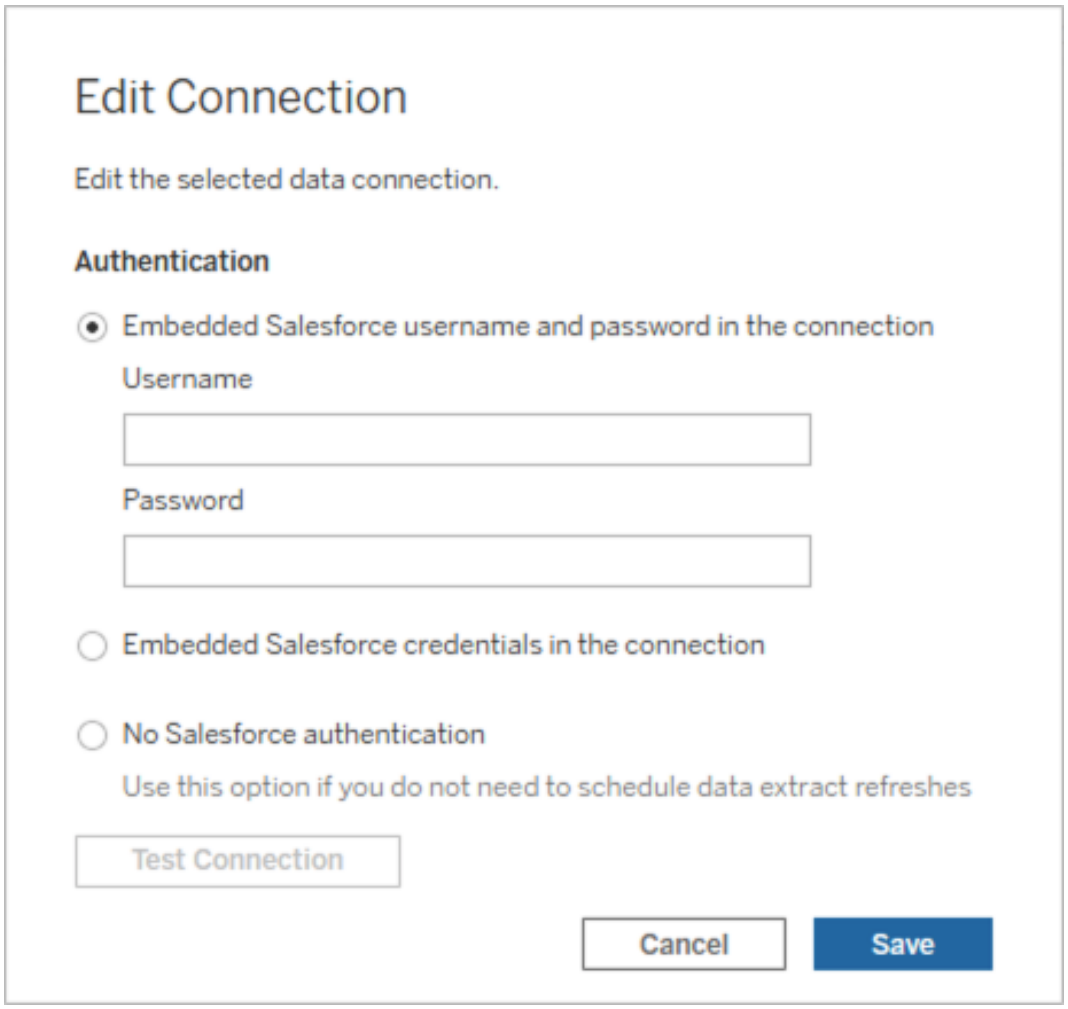
Salesforce.com 認証オプション

> **注:** これは、保存された認証資格情報を使用して Salesforce で Oauth 認証を行うよう Tableau Sever が構成されている場合にのみ適用されます。これを行うようにサーバーが構成されていない場合は、上記の標準プロセスを使用して接続を変更します。保存された認証資格情報を OAuth で使用できるように Tableau Server を構成する方法の詳細については、Salesforce.com OAuth を保存済み認証資格情報に変更するを参照してください。

Salesforce.com 接続を編集すると、[接続の編集] ダイアログ ボックスで次のオプションのいずれかを選択できます。

- 従来の認証方法を使用するには、**[Embedded Salesforce username and password in the connection (Salesforce ユーザー名とパスワードを接続に埋め込む)]**を選択します。
- **[Embedded Salesforce credentials in the connection (Salesforce 認証資格情報を接続に埋め込む)]** を選択して、OAuth 接続およびスケジュール更新タスクを使用してから、リストから既存のアカウントを選択するか、**[Add a Salesforce Account (Salesforce アカウントの追加)]** をクリックして新しいアカウントを追加します。

  新しいアカウントを追加する場合、Salesforce.com サインイン ページが表示されます。データベースの認証資格情報を提供すると、Salesforce.com はデータへの Tableau アクセスを確認するようプロンプトを表示します。Tableau のアクセスを許可すると、Salesforce.com はデータに接続するためのアクセス トークンを作成します。



- ユーザーが接続するたびに Salesforce.com にサインインさせるには、**[Salesforce の認証なし]** を選択します。(このオプションではスケジュールされた抽出の更新を使用できません。)

進行状況の監視

[接続の編集] ダイアログ ボックスでの変更を保存すると、ダイアログは進行状況を表示します。ダイアログ ボックスを閉じると、変更は完了するまでバックグラウンドで継続されます。Tableau Server可能な限り多くの変更が行われます。エラーはスキップされますが、他の変更が行われないということではありません。たとえば、サーバー名を変更し複数の接続にパスワードを追加しようと

した場合、サーバー名は変更され、ワークブックのパスワードは変更されます。しかしながら、データソースにパスワードを追加できないため、データソースのパスワードは変更されません。

これらのタスクの進捗の確認に関する詳細については、抽出でのバックグラウンドタスクを参照してください。

キューブデータソース

キューブ(多次元)データソースには独特な特性があるため、Tableauでユニークなものとなっています。

キューブデータソースはパススルー接続をサポートしていません。つまり、キューブデータソースがパブリッシュされる場合、そのデータソースを使用してTableau Serverからの接続を作成することはできません。また、Tableau Serverでデータソースを使用してワークブックを作成することもできません。

キューブデータソースをTableau Serverにパブリッシュすると、サーバーにデータソースを保存することができます。ただし、データソースを使用するには、データソースをTableau Desktopにダウンロードし、それをローカルで使用する必要があります。パブリッシュされたデータソースをダウンロードするには、次が必要です。

- データソースに対する**[ダウンロード/名前を付けて保存]**パーミッション。詳細については、パーミッションを参照してください。
- Tableau Desktopを実行しているコンピューターで正しいドライバーがインストールされ、ポートが開かれていること。

Tableau Desktopでのキューブデータソースの使用の詳細については、「キューブデータソース」を参照してください。

Tableau Serverでの Webデータコネクタ

Webデータコネクタ(WDCs)とは、Tableau内でコネクタをまだ持たないデータソースに対してHTTP経由でアクセス可能なデータ接続を提供するWebページです。WDCを使用すると、ユーザーはWeb上でアクセス可能なほぼすべてのデータに接続し、ワークブック用に抽出を作成できま

す。WDC 用のデータ ソースには、内部 Web サービス、JSON データ、REST API、HTTP または HTTPS 経由で利用可能な他のソースを含めることができます。ユーザーは独自の WDC を作成するか、他のユーザーが作成したコネクタを使用できます。

Tableau Desktop での WDC の使用方法の詳細については、Tableau Desktop ドキュメントの「Web データ コネクタ」を参照してください。

WDC の作成方法の詳細については、Github の「Web データ コネクタのドキュメント」を参照してください。

Tableau Server でコネクターを実行する前に

このトピックで説明しているように、セキュリティ対策として、コネクタを承認しない限り、Tableau Server は WDC を実行しません。

> **注**: WDC の Tableau Server の使用は、サーバー管理者が承認する必要があります。

WDC には実行可能コードが含まれており、通常、サードパーティの Web サイトへの要求を行うため、管理者による承認が必要です。ユーザーが Tableau Server とともに WDC を使用する前に、コネクタが使用するドメインとポートをセーフリストに追加し、コネクタがリクエストを送受信できるドメインを 2 番目のセーフリストに含める必要があります。これを行う前に、コネクターの挙動およびどのようなサイトに接続するかについて理解できるようにコネクターを入念に検査し、テストすることをお勧めします。詳細については、Web データ コネクタのテストおよび入念な検査を参照してください。

コネクタをセーフリストに追加する場合は、コネクタがホストされている特定の URL への接続およびコネクタが照会できる URL からの接続を許可するようTableau Server を構成します。これが、Tableau Server で WDC の実行を許可する唯一の方法です。その後、コネクターは組織のファイアウォール内にあるサーバー上か、外部ドメインにあるサーバー上でホストできます。Tableau Server では、WDC のインポートはサポートされていません。

セーフリスト内のコネクタの管理

WDC をセーフリストに追加するには、`tsm data-access web-data-connectors add` コマンドを使用します。以下に説明されている、このコマンドと関連コマンドを使用すると、以下のタスクを実行できます。

- WDC をセーフリストおよび 2 番目のセーフリストに追加します。
- すべての WDC、または WDC 更新を許可または却下します。
- セーフリストから 1 つ以上のコネクタを削除します。
- セーフリストおよび 2 番目のセーフリストにあるすべての WDC をリストします。

WDC セーフリストの更新にはサーバーの再起動が必要

WDC に変更を加えるすべてのコマンドの実行後、`tsm pending-changes apply` コマンドを使用して保留中の変更を適用する必要があります。

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

コネクタをセーフリストおよび二番目のセーフリストに追加する

コネクタをセーフリストおよび 2 番目のセーフリストに追加するには、tsm data-access web-data-connectors add コマンドを使用し、名前、セーフリストの URL、2 番目のセーフリストの URL を指定します。web-data-connector-settings エンティティを使用して WDC を管理することもできます。このコマンドを実行した後、`tsm pending-changes apply` コマンドを使用して保留中の変更を適用する必要があります。メッセージが表示され、このコマンドにより Tableau Server が再起動することが警告されます。Tableau Server が現在実行中の場合には再起動されますが、停止している場合には再起動することなく停止状態のままになります。

```
tsm data-access web-data-connectors add --name "USGS Earthquakes" -
--url
https://tableau.github.io:443/webdataconnector/Examples/html/earthq-
uakeUSGS.html --secondary
https://tableau.github.io/.*,https://earthquake.usgs.gov/.*,https:/-
/maxcdn.bootstrapcdn.com/.*,https://ajax.googleapis.com/.*,https://-
connectors.tableau.com/.*
```

書式設定のための注

- カールした引用符や「スマート」引用符ではなく、必ず直線の引用符 (" および ') を使用して WDC の名前を囲ってください。名前にスペースが含まれる場合は、二重引用符 (") を使用します。
- 多くの WDC では `--url` オプションで指定されているポートは 443 または 80 ですが、Tableau Server のデータソースの詳細を確認することで、コネクタの値を確認することができます。WDC が SSL (HTTPS) を使用している場合は、URL の一部としてポート番号を指定する必要があります。たとえば、HTTPS の既定のポートを使用する場合、URL は `https://example.com:443/WDC/` のようになります。
- WDC がアクセスする必要のあるライブラリやソースを提供するドメインを示す `--secondary` オプションには、URL またはカンマ区切りの URL リストを指定する必要があります。WDC がセカンダリドメインを使用する場合以外は、このオプションを省略したり、空のままにしたりすることはできません。WDC がセカンダリドメインを使用しているかどうか、またはセカンダリドメインが何であるかがわからない場合は、WDC の開発者に問い合わせる必要があります。また、`http://.*` や `https://.*` のようなワイルドカード URL を使用してすべてのドメインを許可することもできます。しかし、セキュリティを強化するため、より具体的な URL を使用することを強くお勧めします。
- 2 番目のセーフリストにドメイン全体を追加するには、ドメイン URL の末尾をワイルドカードの式にします。ドメイン全体を示すには、次の例のように、ワイルドカードとして `.*` を使用します。例: `https://example.com/.*`

WDC または WDC 抽出更新の許可または却下

WDC または WDC 抽出更新を許可または却下するには、`-r` または `-t` オプションを付けた `tsm data-access web-data-connectors allow` コマンドを使用します。既定では、WDC と

WDC 抽出更新は許可されています。この設定を変更する場合は、`tsm pending-changes apply` コマンドを実行する必要があります。コマンドとコマンドオプションの実行の詳細については、`tsm data-access web-data-connectors allow` を参照してください。

セーフリストから1つ以上の WDC を削除します。

承認リストから1つ以上の WDC を削除するには、`tsm data-access web-data-connectors delete` コマンドを使用します。この設定を変更する場合は、`tsm pending-changes apply` コマンドを実行する必要があります。詳細については、「`tsm data-access web-data-connectors delete`」を参照してください。

セーフリストにあるすべての WDC のリスト

承認リストにあるすべての WDC を一覧表示するには、`tsm data-access web-data-connectors list` コマンドを使用します。詳細については、「`tsm data-access web-data-connectors list`」を参照してください。

コネクターの抽出の更新

ユーザーが WDC を使用するワークブックを作成すると、Tableau Server はそのコネクタが返すデータから抽出を作成します。その後、ユーザーがワークブックをパブリッシュすると、パブリッシュのプロセスがサーバーにワークブックとデータ抽出を送信します。

Tableau は、他の抽出を更新できるのと同様に、WDC によって作成された抽出を更新できます。Web ベースのデータソースにサインインするためにコネクタが認証資格情報を必要とする場合は、認証資格情報がデータソースに埋め込まれており、WDC がサーバーのセーフリストに含まれていることを確認する必要があります。コネクタが認証資格情報を必要とし、それらがデータソースに埋め込まれていない場合、Tableau Server は抽出を更新できません。これは、スケジュールまたは他のバックグラウンド状況で更新が発生する可能性があり、サーバーは認証資格情報を求めるプロンプトを表示できないためです。

現在、Tableau Server から直接データソースを再認証する方法はありません。データソースが期限切れの認証資格情報を保持している、または認証資格情報が埋め込まれずにパブリッシュされた場合は、ワークブックとデータ抽出を新しい埋め込み認証資格情報とともに、再度パブリッシュする必要があります。

更新操作を実行するバックグラウンドプロセスが失敗した場合、この問題を示すアラートとログエントリが作成されます。ユーザーには、抽出のタイムスタンプが変更されないことがわかります。

すべてのWDCで更新を無効にするには、`tsm data-access web-data-connectors allow -r false`コマンドを使用します。

トラブルシューティング

コネクタをセーフリストに追加する際にサーバーに問題が発生した場合は、ログファイルを調べることができます。初期サーバーノードおよびゲートウェイプロセスを実行しているその他のノードのログファイルを必ず確認してください。ログファイルの詳細については、Tableau Serverログとログファイルの場所を参照してください。

WDCによって作成された抽出をTableau Serverが更新できない問題が発生している場合は、`webdataconnector.refresh.enabled`構成設定が`true`に設定されていることを確認してください。`false`に設定されている場合は、以下のコマンドを実行し、サーバー上のすべてのWDCで抽出更新を許可してください。

```
tsm data-access web-data-connectors allow -r true
```

注: セーフリストは、Tableau ServerでWebデータコネクタの実行を許可する唯一の方法です。Webデータコネクタのインポートは、バージョン10.5以降で非推奨になりました。

Webデータコネクタのテストおよび入念な検査

Webデータコネクタ(WDC)は、通常は別のサイト上のデータに接続するJavaScriptを含んでいます。このため、ユーザーがWebデータコネクタをワークブックのデータソースとして使用する前、およびWebデータコネクタをTableau Serverで使用する前に、Webデータコネクタをテストし、入念に検査することをお勧めします。

このトピックには、Webデータコネクタのテストおよび入念な検査に関するいくつかの提案が含まれています。

ソースの調査

Web データコネクタのコードは JavaScript で記述されているため、ファイル (およびコネクタが使用しているすべての外部ファイル) を開いてソースコードを調べることができます。

多くのコネクタは、jQuery ライブラリやサードパーティ用の API ライブラリなどの外部 JavaScript ライブラリを参照します。外部ライブラリの URL がライブラリの信頼できる場所を指していることを検証します。たとえば、コネクタが jQuery ライブラリを参照する場合、標準および安全とみなされているサイト上にライブラリが存在すること確認してください。コネクタのソースコードを変更することが実用的である場合は、HTTPS プロトコル (`https://`) を使用して外部ライブラリを参照し (ソースサイトが HTTPS をサポートする場合)、サイトの信頼性を検証しやすくします。

可能な限り、コードの内容を理解するようにしてください。具体的には、コードが外部サイトへの要求をどのように構築しているか、また要求によりどのような情報が送信されるかを理解するように努めてください。

> **注:** 経験豊富な JavaScript のプログラマーは、ダウンロード用にコードのサイズを小さくするために、しばしばコードを圧縮 (縮小) します。不可解な関数名と変数名を使用するコードがぎっしり詰まったブロックは珍しくありません。これにより、コードの調査がより困難にはなりますが、意図的にコードの理解を困難にしようとして書かれたものではありません。

隔離された環境での Web データコネクタのテスト

可能な場合、本番環境とユーザーのコンピュータから隔離された環境で Web データコネクタをテストします。たとえば、本番環境で使用されていない Tableau Server のバージョンを実行しているテストコンピューターまたは仮想マシンの安全リストに Web データコネクタを追加します。

Web データコネクタが生成したトラフィックを監視します。

Web データコネクターをテストする際には、Fiddler、Charles HTTP プロキシ、または Wireshark などのツールを使用して、コネクタが実行する要求や応答を調査します。コネクタが要求を送るサイトの種類や、コネクタが要求しているコンテンツの内容を理解してください。同様に、コネクタが、その

目的に関連しないデータやコードを読み取っていないことを確認するために、応答とその内容を確認してください。

Web データコネクタのパフォーマンスとリソース使用のテスト

Web データコネクタをテストする際、コネクタによる CPU とメモリの使用状況を監視するツールを使用します。Web データコネクタは、既にプロセスを実行中の環境である Tableau Server 上で実行されます。コネクタがデータをフェッチする際、コネクタがサーバーのパフォーマンスに過度に影響を与えないようにします。

コネクタがディスクに書き込むかどうかを確認します。その場合、消費ディスク容量をチェックし、出力を調べて書き込みの内容と理由を確認します。

Tableau Catalog の有効化

Tableau Catalog は、ワークブック、データソース、シート、フローなど、Tableau Cloud サイトや Tableau Server のすべてのコンテンツを検出してインデックス化します。(従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。) コンテンツ (またはメタデータ)、およびコンテンツのスキーマと系列についての情報を収集するために、インデックス作成が使用されます。その後メタデータより、Catalog は Tableau Cloud サイトまたは Tableau Server 上のコンテンツが使用するすべてのデータベース、ファイル、表を特定します。

Catalog は データ管理 ライセンスで入手可能です。詳細については、データ管理 についてを参照してください。

Catalog に加え、メタデータ メソッドを使用して Tableau メタデータ API や Tableau REST API からコンテンツについてのメタデータにアクセスすることもできます。

Catalog を有効にする前

Tableau Server 管理者は、Tableau Server 環境で Catalog の最適なパフォーマンスを確保するために、Catalog を有効にする前および有効化の実行中に考慮する必要がある事項がいくつかあります。

必要なバージョン

Catalog を有効にする前に、次のいずれかのバージョンの Tableau Server を実行していることを確認します。

- 少なくとも Tableau Server 2019.3.4 以降
- 少なくとも Tableau Server 2019.4.2 以降
- 少なくとも Tableau Server 2020.1.0 以降
- 少なくとも Tableau Server 2020.2.15 以降
- Tableau Server 2020.3 以降

上記バージョンが必要な理由の詳細については、「Tableau ナレッジ ベース」を参照してください。

Catalog を有効にする場合の期待値

Catalog を有効にすると、Tableau Server に既に存在するコンテンツが直ちにインデックス化されます。

初期取り込み

インデックス作成プロセスは、2 つの主要コンポーネントで構成され、そのうちの 1 つは初期取り込みと呼ばれます。初期取り込みは、次の 2 つの追加コンポーネントに分割できます。

- コンテンツのバックフィル
- 系列のバックフィル

コンテンツのバックフィルのステータスと系列のバックフィルのステータスは、今後進行状況を監視し、Tableau Server 環境で Catalog が正常に有効になっていて実行されていることを検証する際に注意すべき重要な項目です。

初期取り込み速度

最初に Catalog でコンテンツのインデックス作成にかかる時間は、次の 2 つの要素によって異なります。

- **Tableau Server 上のコンテンツの量:** コンテンツの量は、ワークブック、メトリクス、パブリッシュされたデータ ソース、および Tableau Server にパブリッシュされたフローの合計数によって

測定されます。詳細については、メタデータを格納するためのディスク領域を参照してください。

- **非インタラクティブなマイクロサービス コンテナーの数:** Catalog は、非インタラクティブなマイクロサービス コンテナーを使用して、Tableau Server 上のすべてのコンテンツのインデックスを作成します。詳細については、「非インタラクティブなマイクロサービス コンテナーのメモリ」を参照してください。

初期取り込みに影響を与える要因を理解することは、環境で Catalog を有効にして実行するのにかかる時間を測定するのに役立ちます。

メタデータを格納するためのディスク領域

初期取り込み中にメタデータが生成され、Tableau Server リポジトリ(「関係」PostgreSQL データベース) に格納されます。メタデータを格納するために必要となるディスク領域の量は、リポジトリ(「ワークグループ」PostgreSQL データベース) で現在使用されているディスク領域の約半分です。

たとえば、Catalog を有効にする前にリポジトリで 50 GB のディスク領域を使用すると、Catalog の有効後は最大 75 GB のディスク領域を使用できます。

非インタラクティブなマイクロサービス コンテナーのメモリ

初期取り込みは、非インタラクティブなマイクロサービス コンテナー内で実行されます。非インタラクティブなマイクロサービス コンテナーは、2 つの Tableau Server マイクロサービス コンテナー プロセスのうちのいずれかです。既定では、非インタラクティブなマイクロサービス コンテナーの 1 つのインスタンスが、バックグラウンダー プロセスがインストールされているすべてのノードに追加されます。

既定では、非インタラクティブなマイクロサービス コンテナーの単一インスタンスで行われる初期取り込み時に、バックグラウンダー ノードで最大 4 GB のメモリを使用できます。Tableau Server 上のコンテンツの量が 10,000 を超える場合、非インタラクティブなマイクロサービス コンテナーでは、バックグラウンダー ノードで最大 16 GB のメモリが必要になる場合があります。したがって、Catalog を有効にする場合は、初期取り込みプロセス中に、すべてのバックグラウンダー ノードに、非インタラクティブなマイクロサービス コンテナーをサポートするための使用可能な容量があることを確認します。容量を増やす必要がある場合は、非インタラクティブなコンテナーに対して JVM ヒープ サイズ

を更新し、バックグラウンダー ノードで最大 16 GB のメモリを割り当てる必要があります。詳細については、「noninteractive.vmopts」を参照してください。

非インタラクティブなマイクロサービス コンテナーを追加して初期取り込み時間を短縮する場合は、まず必要なコンテナー数の合計を決定し(以下の「ステップ 2: 初期取り込みにかかる時間を見積もる」を使用)、Tableau Server 環境がすべての非インタラクティブなマイクロサービス コンテナーに対応するのに十分な容量で構成されているかどうかを確認します。Tableau Server 環境の構成方法によっては、初期取り込み時間を短縮するのに必要となる非インタラクティブなマイクロサービス コンテナーをすべて追加することができない場合があります。

Catalog を有効にするためのベストプラクティス

初期取り込みの速度と要件は各 Tableau Server 環境に固有であるため、Catalog を有効にする場合は、次の 1 つまたは複数を実行することをお勧めします。

- Tableau Server リポジトリに、初期取り込みによって生成されたり格納されたりする追加のメタデータに対応するために使用できる十分なディスク容量があることを確認します。一般的に、リポジトリには現在リポジトリが使用しているディスク領域からさらに 50% 追加で必要になります。Tableau Server のディスク容量に関する詳細については、サーバーのディスク容量を参照してください。
- Tableau Server のコンテンツ量に応じて、初期取り込み中に、各バックグラウンダー ノードに非インタラクティブなマイクロサービス コンテナーの各インスタンスに対して、4 ~ 16 GB 以上の使用可能なメモリがあることを確認します。
- 週末にプロセスを実行して、ユーザーが Catalog 機能を使用する前に最初の取り込みが完了できるようにします。
- 最初に本番コンテンツを使用してテスト環境でプロセスを実行します。これは、取り込む必要のあるコンテンツの種類が摂取速度に重要な役割を果たす可能性があるためです。

Catalog を有効にする手順の概要

次のステップは、Tableau Server で Catalog を有効にして実行するプロセスをまとめたものです。ステップは順番に実行する必要があります。

1. Tableau Server 上のコンテンツ量を決定する
2. 最初の取り込み時間を見積もる
3. 最初の取り込み時間を短縮する
4. データ管理 ライセンスの認証
5. Catalog 機能をオフにする
6. tsm maintenance metadata-services コマンドを実行する
7. 初期取り込みの進行状況を監視し、そのステータスを検証する
8. SMTP の構成
9. Catalog 機能をオンにする

注: Tableau Server 上の Tableau コンテンツに関するメタデータのインデックス化は メタデータAPI によって処理されるため、Catalog を実行および使用するには、メタデータAPI を有効にする必要があります。

Catalog の有効化

ステップ 1: Tableau Server 上のコンテンツ量を決定する

Tableau Server 上のコンテンツ量を決定するには、次の手順を実行します。

1. 管理者の認証資格情報を使用して Tableau Server にサインインします。
2. **[探索]** ページに移動します。
3. [最上位のプロジェクト] ドロップダウン メニューをクリックし、**[すべてのワークブック]**、**[すべてのメトリクス]**、**[すべてのデータ ソース]**、および **[すべてのフロー]** の横にまとめて数値を追加します。これが、Tableau Server 上にあるコンテンツの合計量となります。

ステップ 2: 初期取り込みにかかる時間を見積もる

Catalog が Tableau Server で初めてコンテンツを取り込むのにかかる時間を見積もるには、*お使いの* Tableau Server の設定をベースラインの Tableau Server 設定と比較します。

次の設定が含まれる Tableau Server では、最初の取り込みが完了するまでに約 6 時間かかる場合があります。

コンポーネント	ベースライン値
コンテンツ	17,000 個のワークブック、メトリクス、パブリッシュされたデータ ソース、およびフロー
非インタラクティブなマイクロサービス コンテナー	10
取り込み	**最大 6 時間**

お使いの Tableau Server 環境に含まれているコンテンツがだいたい*半量*程度の場合は、最初の取り込みが完了するまでにかかる時間も半分程度となる見込みです。

たとえば、8,500 (ワークブック、メトリクス、パブリッシュされたデータ ソース、およびフロー) + 10 非インタラクティブなマイクロサービス コンテナー = 最大 3 時間 (初期取り込み)

お使いの Tableau Server 環境に含まれているコンテンツがだいたい *2 倍*程度の場合は、最初の取り込みが完了するまでにかかる時間も 2 倍程度となる見込みです。

たとえば、34,000 (ワークブック、パブリッシュされたデータ ソース、およびフロー) + 10 非インタラクティブなマイクロサービス コンテナー = 最大 12 時間 (初期取り込み)

ステップ 3: 初期取り込みの時間を短縮する

一般に、Catalog での初期取り込みにかかる時間は、非インタラクティブなマイクロサービス コンテナーの数に相関します。初期取り込み時間を短縮するために、非インタラクティブなマイクロサービス コンテナーの数を増やすことができます。

非インタラクティブなマイクロサービス コンテナーの数を増やす

既定では、バックグラウンダーを持つすべてのノードに、非インタラクティブなマイクロサービス コンテナーが 1 つ追加されています。Tableau では、初期取り込み時間を短縮するために、`tsm topology set-process` コマンドを使用して、非インタラクティブなマイクロサービス コンテナーの数を増やすことをお勧めします。

1. クラスタの最初のノード (TSM がインストールされているノード) で管理者としてコマンド プロンプトを開きます。

2. `tsm topology set-process --count <process_count> --node <node_ID> --process <process_name>` コマンドを実行します。

   たとえば、第一ノードの非インタラクティブなマイクロサービス コンテナーの数を 4 つに増やすには、次のコマンドを実行します。

   ```
   tsm topology set-process --count 4 --node node1 --process noninteractive
   ```

   コマンドとそのグローバル オプションの実行の詳細については、tsm トポロジーを参照してください。

重要: 非インタラクティブなマイクロサービス コンテナーの数を増やす前に、次の点を確認してください。

- 非インタラクティブなマイクロサービス コンテナーを増やす理由は、非インタラクティブなマイクロサービス コンテナーの合計数を増やすためであって、ノードごとの非インタラクティブなマイクロサービス コンテナーの合計数を増やすためではありません。たとえば、4 つのノードがある場合に、非インタラクティブなコンテナーの数を 8 に増やすとします。`--count` tsm コマンドで使用する値は 2 です。
- 非インタラクティブなマイクロサービス コンテナーを追加するごとに、ノードで 4GB の追加メモリが使用され、Tableau Server リポジトリ(PostreSQL データベース) に負荷がかかるようになります。
  - 非インタラクティブなマイクロサービス コンテナーを一度に 2 つずつ段階的に増やしながら、Tableau Server の環境を注意深く監視して、Tableau Server リポジトリ (PostgreSQL データベース) の CPU 使用率に関する問題を回避することをお勧めします。
  - 非インタラクティブなマイクロサービスを多く追加しすぎると、PostgreSQL データベースの CPU 使用率が急増してフェールオーバーする可能性があることに注意してください。監視する必要がある現象には、vizportal ログの SQLException エラーなどがあります。詳細については、「リポジトリフェールオーバー」トピックを参照してください。

ステップ 4: データ管理 ライセンスを認証する

(データ管理 が必要です)

まだ完了していない場合は、データ管理 をアクティブ化できます。詳細については、データ管理 のライセンスを参照してください。

ステップ 5 (オプション): 各サイトの Catalog の機能をオフにする

(データ管理 が必要です)

データ管理 のアクティブ化の一部として、Catalog の機能は既定で有効になっています。インデックス作成プロセスと完了までにかかる推定時間のため、Catalog 機能を一時的にオフにして、Catalog 機能が完全な状態で提供でき、正確な結果が得られるようになるまで Tableau Server ユーザーがこれらの機能にアクセスできないようにすることを検討してもよいでしょう。

1. 管理者の認証資格情報を使用して Tableau Server にサインインします。
2. 左側のナビゲーション ペインで、**[設定]** をクリックします。
3. [全般] タブで、[Tableau Catalog] の下にある **[Tableau Catalog をオンにする]** チェック ボックスをオフにします。
4. Tableau Server の各サイトで、ステップ 2 ~ 3 を繰り返します。

ステップ 6:tsm maintenance metadata-services コマンドを実行する

`tsm maintenance metadata-services` コマンドを実行して、Tableau メタデータ API を有効にします。コマンドの実行は、初期取り込みを開始します。Tableau Server がデータ管理 でライセンスされている場合、このコマンドを実行すると Catalog 機能もオンになります (以前にオフにされていない場合)。

1. クラスタの最初のノード (TSM がインストールされているノード) で管理者としてコマンド プロンプトを開きます。

2. `tsm maintenance metadata-services enable` コマンドを実行します。

   tsm コマンドの実行の詳細については、「tsm maintenance」を参照してください。

注: このコマンドを実行するときは、次の点に留意してください。

- このコマンドは、Tableau Server で使用されているいくつかのサービスが停止してから起動します。これによって推奨事項などの一部の機能を一時的に使用できなくなります。
- この時点で、メタデータの新しいインデックスが作成されます。このコマンドをそれ以降の時間で実行すると、以前のインデックスが作成され、置き換えられます。

ステップ 7: 初期取り込みの進行状況を監視し、その状態を検証する

上記の tsm コマンドを実行すると、初期取り込みプロセスが開始されます。初期取り込みプロセスが円滑に進むよう、バックフィル API を使用してプロセスの進行状況を監視することができます。詳細については、初期取り込みステータスの取得を参照してください。

ステップ 8: SMTP セットアップを構成する

Tableau Server でまだ設定していない場合は、SMTP セットアップを構成してください。SMTP は、データの変更に関する連絡が必要な所有者へのメール送信をサポートしています。SMTP の構成の詳細については、「SMTP セットアップの構成」を参照してください。

ステップ 9 (オプション): 各サイトの Catalog 機能をオンにする

(データ管理 が必要です)

上記のいずれかの手順で Catalog を有効にする前に Catalog 機能をオフにした場合は、Catalog をオンにして、ユーザーがその機能にアクセスできるようにする必要があります。

1. 管理者の認証資格情報を使用して Tableau Server にサインインします。
2. 左側のナビゲーション ペインで、**[設定]** をクリックします。
3. [Tableau Catalog] の [全般] タブで **[Tableau Catalog をオンにする]** チェック ボックスを選

択します。

4. Tableau Server の各サイトで、ステップ 2 ~ 3 を繰り返します。

Catalog のトラブルシューティング

Catalog の使用時に、ユーザーは次のいずれかの問題に直面する可能性があります。

「Timeout limit and node limit exceeded (タイムアウト制限およびノード制限を超過しました)」メッセージ

大量の結果を返す必要がある Catalog のタスクが Tableau Sever のすべてのシステム リソースを消費してしまわないようにするため、Catalog ではタイムアウト制限とノード制限の両方を設けています。

- **タイムアウト制限**

  Catalog 内のタスクがタイムアウト制限に達すると、管理者およびユーザーに次のメッセージが表示されます。

  *「Showing partial results, Request time limit exceeded.Try again later. (結果の一部を表示しています。要求時間の制限を超過しました。後でもう一度やり直してください。)」*または TIMEOUT_LIMIT_EXCEEDED

  この問題を解決するには、Tableau Server 管理者として、`tsm configuration set -k metadata.query.limits.time` コマンドを使用してタイムアウト制限を増やすことができます。詳細については、「tsm configuration」トピックおよび「tsm configuration set のオプション」トピックを参照してください。

  重要: タイムアウト制限を増やすと、さらに多くの CPU をより長く使用できるため、Tableau Server 上の他のプロセスのパフォーマンスに影響を与える可能性があります。

- **ノード制限**

  Catalog 内のタスクがノード制限に達すると、管理者およびユーザーに次のメッセージが表示されます。

NODE_LIMIT_EXCEEDED

この問題を解決するには、Tableau Server 管理者として、`tsm configuration set -k metadata.query.limits.count` コマンドを使用してノード制限を増やすことができます。詳細については、「tsm configuration」トピックおよび「tsm configuration set のオプション」トピックを参照してください。

重要: タイムアウト制限を増やすと、システムのメモリに影響を与える可能性があります。

コンテンツが見つからない

- 最初の取り込みの後に、Catalog でコンテンツが欠落していると思われる場合は、イベント API を使用してトラブルシューティングを行うことができます。イベント処理では、最初に取り込みした後に Tableau Server でコンテンツのインデックスを作成します。詳細については、イベントステータスの取得を参照してください。

- 埋め込まれた外部アセットとそのダウンストリーム のTableau コンテンツ間の接続が削除されると、その接続は、毎日 22:00:00 UTC (協定世界時) に実行されるバックグラウンダープロセスによって自動的に削除されるまで、Catalog (または Tableau メタデータAPI) に残ります。たとえば、最初に埋め込みテキストファイル A でパブリッシュされたワークブックが、埋め込みテキストファイル B で再びパブリッシュされた場合、ファイル A は、バックグラウンダープロセスでそれを削除できるようになるまで、外部アセットとして表示 (またはクエリを実行可能) のままになります。

  これらのタイプの外部アセットを削除したくない場合、またはこのプロセス専用にしたくないシステム リソースを使用していると思われる場合は、このバックグラウンダープロセスの実行を無効にすることができます。または、削除される外部埋め込みアセットの数を調整することもできます。詳細については、features.DeleteOrphanedEmbeddedDatabaseAsset および databaseservice.max_database_deletes_per_run を参照してください。

  このプロセスは次のいずれかの方法で監視できます。

  - 抽出以外のバックグラウンドタスク管理ビューで**正規化ロジックによってタスクのタイプを変更した後**、ワンタイム ジョブにフィルターを設定すると、既存のデータベース/

テーブル アセットがもう一度正規化されます。

- **「孤立した埋め込みデータベースの削除の完了」**または Tableau Server ログ ファイルの **database_service_canonicalization_change** イベントを参照してください。

初期取り込み後のパフォーマンス

特定のコンテンツが頻繁に更新される(頻度の高いスケジュール、コマンド ライン、API リクエストなどを介して)一部の Tableau Server 環境では、インデックス作成プロセスが飽和状態になる可能性があります。このような場合、サーバー管理者は、Catalog のパフォーマンスを適切に維持するためにイベント スロットルを有効にすることを検討する場合があります。詳細については、metadata.ingestor.pipeline.throttleEventsEnableを参照してください。

注: イベント スロットルが有効になっていると、ユーザーは、Catalog でのコンテンツ変更に意図的な遅延が発生することに気付くことがあります。

メモリ不足エラー

場合によっては、複雑なコンテンツを取り込む際に問題が生じて、Tableau Server でメモリ不足エラーが発生することがあります。Tableau Server のメモリ不足エラーの原因がコンテンツの取り込みであると思われる場合、問題を解決するには、Tableau サポートに連絡して metadata.ingestor.blocklistが取り込まれないようにします。

Catalog の無効化

カタログを無効にするには、2つの方法があります。

各サイトの Catalog 機能をオフにする

(データ管理 が必要です)

Catalog 機能はいつでもオフにできます。Catalog がオフになっている場合、Catalog の機能にアクセスすることができません。この機能には、データ品質に関する警告の追加や、データベースやテーブルのアセットに対するパーミッションの明示的な管理などが含まれます。ただし、Catalog はパブリッシュ済みコンテンツのインデックス化を継続し、メタデータには Tableau メタデータAPI から、また Tableau REST API のメタデータ メソッドからアクセスすることができます。

1. 管理者の認証資格情報を使用して Tableau Server にサインインします。
2. 左側のナビゲーション ペインで、**[設定]** をクリックします。
3. [全般] タブで、[Tableau Catalog] の下にある **[Tableau Catalog をオンにする]** チェック ボックスをオフにします。

メタデータのインデックス作成を停止する

Tableau Server でパブリッシュ済みコンテンツのインデックス化を中止するには、Tableau メタデータ API を無効にします。メタデータ API を無効にするには、`tsm maintenance metadata-services disable` コマンドを実行します。詳細については、「tsm maintenance」を参照してください。

初期取り込みステータスの取得

`tsm maintenance metadata-services` コマンドを使用して Tableau メタデータ API を有効にした後、バックフィル API を使用して初期取り込みの進行状況を監視し、コンテンツと系列のバックフィルのステータス情報を取得できます。

以下に説明する手順は、サーバー管理者が実行する必要があり、「Tableau Catalog の有効化」トピックのステップ 7: 初期取り込みの進行状況を監視し、その状態を検証すると組み合わせて使用することをお勧めします。

ステップ 1: REST API を使用して認証する

バックフィル API にアクセスするには、最初に Tableau Server に対して認証を行い、トークンを取得する必要があります。この操作は、Tableau REST API を使用してで実行できます。詳細については、REST API ヘルプの「サインインとサインアウト (認証)」(英語) を参照してください。

または、管理者の認証資格情報を使用して Tableau Server にサインインできます。

ステップ 2: GET 要求を実行する

次の GET 要求を行うか、URL をブラウザーに貼り付けます。

```
http://my_tableau_server/api/metadata/v1/control/backfill/status
```

リクエストは、最初にコンテンツのバックフィルに関する情報を返します。コンテンツのバックフィルが完了すると、系列のバックフィル情報が返されます。

- コンテンツのバックフィルの場合、リクエストは、Tableau Server で使用可能なコンテンツに応じて、各コンテンツ タイプのステータスの概要と追加のステータス情報を返します。各コンテンツ タイプのインデックス作成は同時に実行されます。
- 系列のバックフィルの場合、リクエストはステータスの概要を返します。

応答のステータス値

以下の値は、バックフィル API によって返されます。

- **contentBackfillTotalDurationSeconds** と **lineageBackfillTotalDurationSeconds** は、それぞれのバックフィル タイプでどれくらい時間が経過したかを秒単位で示します。`backfillComplete` が `true` の場合、`contentBackfillTotalDurationSeconds` と `lineageBackfillTotalDurationSeconds` は、それぞれのバックフィル タイプの完了にかかった合計時間を示します。

コンテンツのバックフィルの場合:

- **contentType** は、コンテンツ タイプ `PublishedDatasource`、`Database`、`DatabaseTable`、`Metric`、`Workbook`、および `Flow` を示します。
- **contentId** は、最後にインデックス化されたアイテムの識別子です。
- **successfullyIngestedCount**は、正常にインデックス化されたアイテムの数です。
- **failedIngestedCount**は、インデックス化できなかったアイテムの数です。
- **durationSeconds**は、コンテンツ タイプのアイテムのインデックス化に費やされた時間 (秒単位) です。
- **totalCount** は、インデックス化するアイテムの総数です。
- **checkpointCreatedTime** は、アイテムのインデックス化が最後に記録された日時 (UTC) です。バックフィル API は、最後にインデックス化されたアイテムを 5 分ごとにチェックします。
- **backfillComplete** が `true` の場合、コンテンツ タイプのすべてのアイテムのインデックス化が完了しています。

系列のバックフィルの場合:

- **totalCount** は、インデックス化する系列関係の総数です。
- **processedCount**は、インデックス化された系列関係の数です。
- **lastLineageConnection**は、最後にインデックス化された系列関係です。

- **backfillComplete** が true の場合、すべての系列関係のインデックス化が完了しています。

応答例:

要求を実行するとJSON テキストが返されます。JSON を読みやすい形式で表示するには、JSON ビューアーまたはブラウザー アドオンを使用します。


```
{
  "contentBackfillTotalDurationSeconds": 362,
  "lineageBackfillTotalDurationSeconds": 14,
  "contentBackfillStatuses": [
    {
        "contentType": "PublishedDatasource",
        "contentId": "sites/1/datasources/-631379806-1912815680",
        "successfullyIngestedCount": 20,
        "failedToIngestCount": 0,
        "durationSeconds": 312,
        "totalCount": 20,
        "checkpointCreatedTime": "2020-07-29T23:50:25.763Z",
        "backfillComplete": true
    },
    {
        "contentType": "Database",
```

```
            "contentId": "sites/1/databases/e1331f9d-4d73-ee04-9edf-
   96fd1c37cb8e",
            "successfullyIngestedCount": 35,
            "failedToIngestCount": 0,
            "durationSeconds": 26,
            "totalCount": 35,
            "checkpointCreatedTime": "2020-04-29T23:50:25.769Z",
            "backfillComplete": true
        },
        {
            "contentType": "DatabaseTable",
            "contentId": "sites/1/tables/d946d084-53a8-09b6-2ad2-93301e6b4b15",
            "successfullyIngestedCount": 64,
            "failedToIngestCount": 0,
            "durationSeconds": 49,
            "totalCount": 64,
            "checkpointCreatedTime": "2020-04-29T23:50:25.774Z",
            "backfillComplete": true
        },
        {
```

```
    "contentType": "Metric",
    "contentId": "sites/1/metrics/metric1",
    "successfullyIngestedCount": 2,
    "failedToIngestCount": 0,
    "durationSeconds": 254,
    "totalCount": 2,
    "checkpointCreatedTime": "2020-04-29T23:50:25.779Z",
    "backfillComplete": true
},
{
    "contentType": "Workbook",
    "contentId": "sites/1/workbooks/6749399-1501801290",
    "successfullyIngestedCount": 10,
    "failedToIngestCount": 0,
    "durationSeconds": 267,
    "totalCount": 10,
    "checkpointCreatedTime": "2020-04-29T23:50:25.784Z",
    "backfillComplete": true
},
{
```

```
            "contentType": "Flow",

            "contentId": "sites/1/flows/4",

            "successfullyIngestedCount": 4,

            "failedToIngestCount": 0,

            "durationSeconds": 195,

            "totalCount": 4,

            "checkpointCreatedTime": "2020-04-29T23:50:25.788Z",

            "backfillComplete": true

        }

       ],

      "lineageBackfillStatus": {

            "totalCount": 45,

            "processedCount": 18,

            "lastLineageConnection": "CloudFile downstreamWorkbooks Workbook",

            "backfillComplete": false

        }

    }
```


イベントステータスの取得

Tableau Server 環境で Tableau Catalog (または Tableau メタデータAPI) を有効にした後、イベントAPI を使用してインデックス作成のパフォーマンスを測定できます。

以下に説明するステップは、サーバー管理者が実行する必要があります。

ステップ 1: REST API を使用して認証する

イベントAPI にアクセスするには、まず Tableau Server に対して認証を行い、トークンを取得する必要があります。この操作は、Tableau REST API を使用してで実行できます。詳細については、REST API ヘルプの「サインインとサインアウト(認証)」(英語) を参照してください。

または、管理者の認証資格情報を使用して Tableau Server にサインインできます。

ステップ 2: GET 要求を実行する

次の GET 要求を行うか、URL をブラウザーに貼り付けます。

```
http://my_tableau_server/api/metadata/v1/control/eventing/status
```

応答のステータス値

次の値は、イベントAPI によって返されます。

- **contentType**は、最後にインデックスが作成されたコンテンツ タイプです。
- **queueSize**は、インデックス作成キュー内のアイテムの数です。キューのサイズが大きいほど、カタログまたは メタデータAPI にアイテムが表示されるまでの時間が長くなります。時間の経過とともにキューのサイズが増加する場合は、インタラクティブなコンテナーとインデックス作成プロセスをサポートできるよう、バックグラウンダーの容量を調整する必要があります。詳細については、非インタラクティブなマイクロサービス コンテナーのメモリを参照してください。
- **checkpointCreatedTime**は、最後にアイテムのインデックス作成が記録された時刻 (UTC) です。イベントAPI では、5 分ごとに最後にインデックスが作成されたアイテムがチェックされます。

応答例:

要求を実行するとJSON テキストが返されます。JSON を読みやすい形式で表示するには、JSON ビューアーまたはブラウザー アドオンを使用します。

```
{

   "contentType": "PublishedDatasource",

   "queueSize": 312,
```

```
    "checkpointCreatedTime": "2020-07-29T23:50:25.763Z"
}
```

インパクト分析での系列の使用

データの由来を把握することは、データを信頼する上で重要です。また、データを使用している他のユーザーを把握しておくと、環境内でデータに対する変更により受ける影響を分析できます。Tableau Catalog の系列機能は、これらの両方を実行できます。

データ管理ライセンスがあり、Tableau Catalog が有効になっている場合は、コンテンツの系列情報にアクセスできます。Tableau Catalog の詳細については、Tableau Server または Tableau Cloud のヘルプの「Tableau Catalog について」を参照してください。

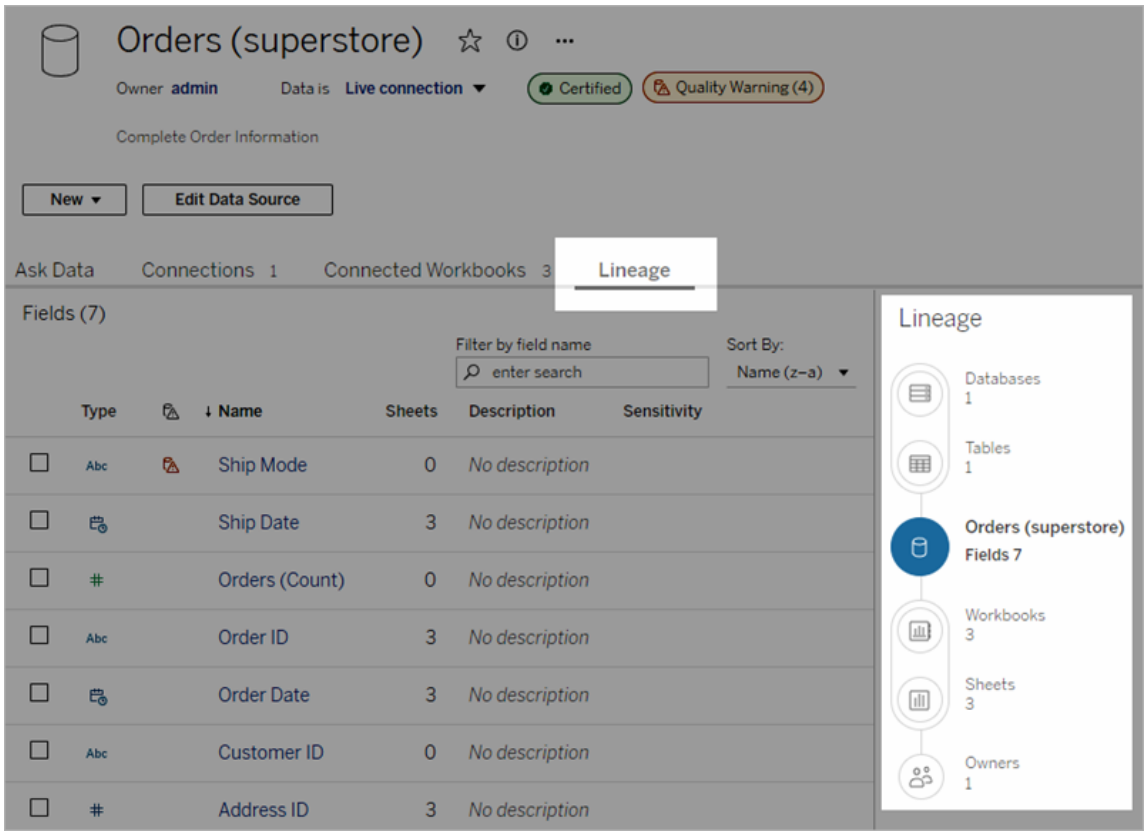
系列の移動

アセットの系列を確認するには、まずアセットのページに移動します。このステップのオプションは次のとおりです。

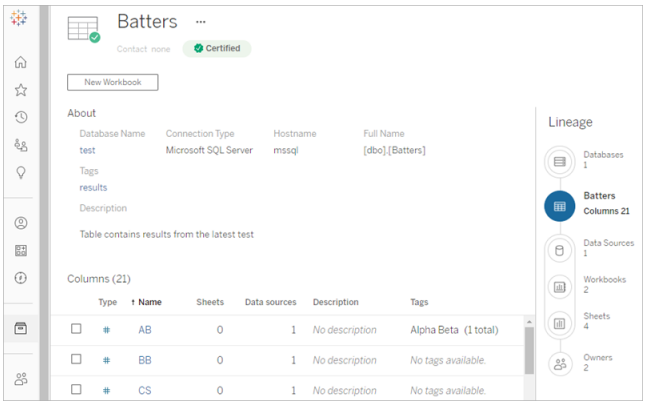
- アセットを検索して選択します。
- **[探索]** からアセットに移動します。
- プロジェクト内にない外部アセット (データベースやテーブルなど) の場合は、**[外部アセット]** からアセットに移動します (このオプションは、プロジェクト内にある外部アセットに対しても機能します)。

次に、**[系列]** タブを選択します。

注: フローにパラメーター値が含まれている場合、フローの系列データは表示されません。フローでパラメーターを使用する方法の詳細については、Tableau Prep ヘルプの「フローでパラメーターを作成して使用する」を参照してください。

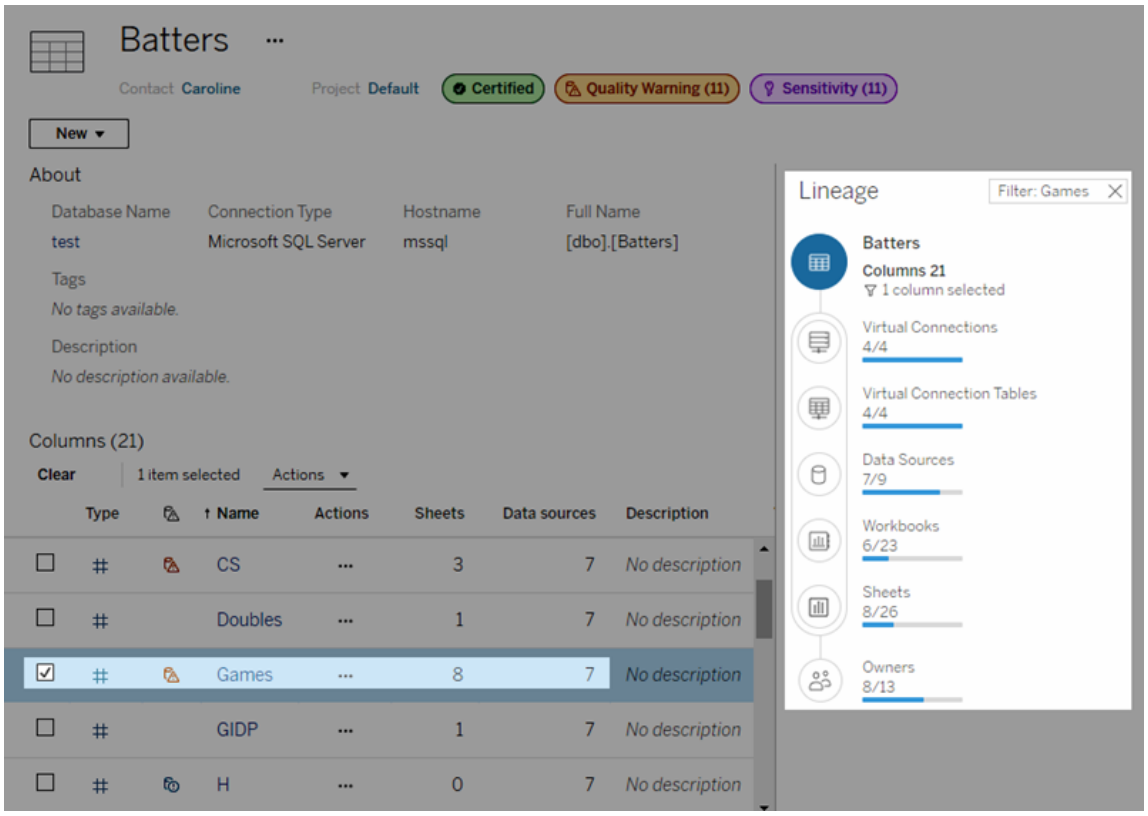


系列には、系列アンカー(選択したアセット)との関係における依存性を示します系列アンカーは、データベース、テーブル、ワークブック、パブリッシュされたデータソース、仮想接続、仮想接続テーブル、またはフローなどです。(アンカーは、上記の図では "Orders (オーダー) (スーパーストア)" データソース、下記の図では "Batters (バッター)" テーブルです)。アンカーの下にあるアセットは、すべてアンカーに直接的または間接的に依存しており、アウトプットまたはダウンストリーム アセットと呼ばれます。アンカーの上にあるアセットは、アンカーが直接的または間接的に依存するアセットで、インプットまたはアップストリーム アセットと呼ばれます。



2023 年 6 月の Tableau Cloud および Tableau Server 2023.3 以降では、データ ソースの系列ページに検索 とフィルター (フィールド リストの右上) が含まれており、関心のあるフィールドや関連性のあるフィールドをすばやく見つけることができます

データ ソースのフィールドまたはテーブルの列を選択すると、「ゲーム」列にフィルターを適用した系列を示す「バッター」テーブルの例のように、フィールド (または列) に依存する、あるいはフィールド (または列) に対するアップストリーム インプットに依存するダウンストリーム アセットのみが表示されるように系列がフィルターされます。



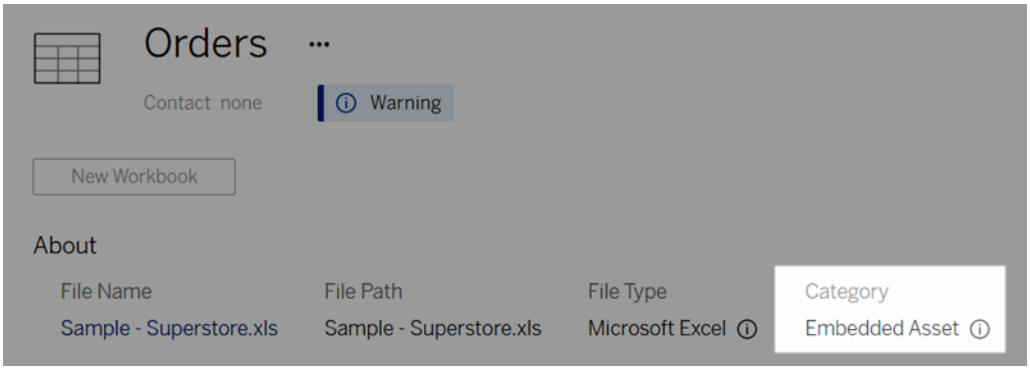
[系列] ペインでアップストリームまたはダウンストリーム アセットを選択して、その詳細を確認することができます。たとえば、[データ ソース] を選択すると、このテーブルに依存するデータ ソースのリストが [系列] ペインの左側に表示されます。

[系列] ペインから、最初に選択したアセットに関連するアセットに移動することができます。この場合は、テーブルから関心のあるリンクへと移動します。

埋め込みアセットが外部アセットに表示される

データ ソースまたはワークブックのパブリッシュ時に **[外部ファイルを含める]** チェック ボックスがオンになっている場合、Tableau Catalog は外部アセットを‘埋め込み’として扱います。外部アセット (データベース、テーブル、またはファイル) がパブリッシュされた Tableau コンテンツ (ワークブック、データ ソース、フロー) に埋め込まれている場合、その外部アセットはコンテンツによって使用されますが、他のユーザーと共有することはできません。埋め込まれた外部アセットは、Tableau コンテンツからアップストリームの系列に表示され、外部アセットに一覧表示されます。

外部アセットが埋め込まれているかどうかを確認するには、外部アセットの詳細ページに移動し、**[カテゴリー]** に [埋め込みアセット] が表示されているかどうかを確認します。



埋め込まれたデータの詳細については、Tableau Desktop および Web 作成のヘルプの「データを個別にパブリッシュするか、ワークブックに埋め込む」を参照してください。

系列とカスタム SQL 接続

カスタム SQL を使用する接続の系列を表示する場合は、次の点に注意してください。

- 系列が完全でない可能性があります。
- Catalog では、カスタム SQL を介してのみ認識されるテーブルの列情報は表示されません。
- フィールドの詳細カードには、接続されている列へのリンクが含まれていない場合や、接続されている列がまったく表示されない場合があります。
- 列の詳細カードには、列を使用するフィールドへのリンクが含まれていない場合や、フィールドがまったく表示されない場合があります。

詳細については、Tableau Desktop と Web オーサリングのヘルプの「カスタム SQL の Tableau Catalog のサポート」を参照してください。

Catalog ではキューブはサポートされていません

キューブ データ ソース (多次元データ ソースまたは OLAP データソースとも呼ばれます) は、Tableau Catalog ではサポートされていません。キューブ データに依存する Tableau コンテンツ (データ ソース、ビュー、ワークブックなど) では、Catalog にキューブ メタデータまたはキューブ系列は表示されません。

系列カウントとタブ カウントの不一致

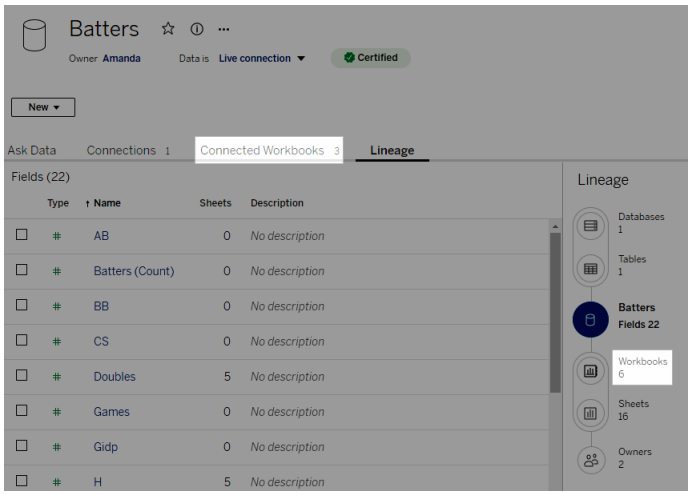
Tableau Catalog の系列ツールとTableau Server や Tableau Cloud のタブの間でアセット数が一致しないことに気付くかもしれません。

数が一致しないことは、それぞれの系列とタブで、アセットが異なる方法でカウントされていることによって説明されます。たとえば、Catalog は特定の時点でインデックス化されたアセットのみをカウントすることができますが、Tableau Server や Tableau Cloud はパブリッシュ済みのアセットをカウントします。カウントに差が出る他の理由には以下のものもあります。

- アセットの "ビュー" パーミッションを持っている。
- アセットが非表示である。
- 任意のフィールドがワークブックで使用されている。
- アセットが直接的または間接的に接続されている。
- アセットがパーソナル スペースにある。

ワークブック カウントの不一致の例

一例として、ワークブックでタブ カウントと系列カウントがどのように決まるかを示します。



[接続されたワークブック] タブでは、次の両方の条件を満たすワークブックをカウントします。

- データ ソースに接続している (任意のフィールドがワークブックで実際に使用されているかどうかは問わない)。
- ユーザーがビューに対するパーミッションを持っている (ワークシート、ダッシュボード、またはストーリーであるかは問わない)。

Tableau Catalog の系列は、次のすべての条件を満たすワークブックをカウントします。

- Tableau Catalog によってインデックスが作成されている。
- データソースに接続しており、データソース内の少なくとも 1 つのフィールドを使用している。
- ワークシートを含むダッシュボードやストーリーなど、データソース内の少なくとも 1 つのフィールドを使用するワークシートを含んでいる。

パーミッションが制限されているためにメタデータがブロックされる場合や、アセットがパーソナル スペースにある場合、Catalog は引き続きワークブックをカウントします。機密メタデータの一部を表示するのではなく、**「必要なパーミッション」**を参照してください。詳細については、系列情報にアクセスするを参照してください。

メールを使用して所有者に連絡する

系列の最後は所有者です。所有者のリストには、系列アンカーのダウンストリームにあるコンテンツの所有者または連絡先として割り当てられたユーザーが含まれます。

所有者にメールを送信して、データの変更を通知できます。(所有者にメールを送信するには、系列アンカー コンテンツの "上書き" (保存) 機能が必要です) 。

1. この系列のデータによって影響を受けるユーザーのリストを表示するには、**[所有者]** を選択します。
2. メッセージを送信する所有者を選択します。
3. **[メールの送信]** をクリックして、メール メッセージのボックスを開きます。
4. テキスト ボックスに件名とメッセージを入力し、**[送信]** をクリックします。

データ ラベル

データ ラベルは、データ アセットに設定できるメタデータです。データ ラベルを使用すると、データを分類し、ユーザーに情報を伝えることができます。例:

- パブリッシュされたデータ ソースは、他の同じような名前のデータ ソースよりも信頼できます。認定データ ラベルを使用すると、どのデータ ソースが推奨されるかをユーザーに知らせることができます。
- データベース内のフィールドには、古い情報が含まれています。警告データ ラベルを使用すると、データが最新ではないことをワークブックの作成者や閲覧者に伝えることができます。

- 従業員の収入のテーブルには、共有すべきではない機密情報が含まれています。機密度データラベルを使用すると、テーブルのデータを使用するときに注意する必要があることをユーザーに通知できます。
- パブリッシュされた一部のデータソースは、それらをパブリッシュした部門に基づいてグループ化できます。カスタム ラベル カテゴリにカスタム ラベルを付けると、データソースの担当部門を特定することができます。

> 注: データラベルは、メタデータを分類する方法に関する、より新しい拡張可能な考え方です。「データラベル」という用語が存在するずっと前からデータ管理 ライセンスの一部であった認定とデータ品質に関する警告は、現在では、2023 年 6 月の Tableau Cloud および Tableau Server 2023.3 でリリースされた機密 ラベルとともに、より広範なデータラベル概念のカテゴリとみなされます。

データ管理 ライセンスは、パブリッシュされたデータソースの認証に関連する操作を除き、すべてのデータラベル操作に必要です。

ラベルを付けることができるアセット

次の Tableau コンテンツおよび外部アセットにラベルを付けることができます。

- データベース
- テーブル
- 列 (認証を除く) *(Tableau Cloud 2022 年 10 月 / Server 2022.3 で導入された列 ラベル)*
- データソース
- フロー
- 仮想接続
- 仮想接続テーブル

ラベル名とカテゴリ

各ラベルには名前とカテゴリがあります。Tableau に組み込まれている名前とカテゴリは次のとおりです。

名前	カテゴリ

認定済み	認定
非推奨	データ品質に関する警告
古いデータ	データ品質に関する警告
メンテナンス中	データ品質に関する警告
警告	データ品質に関する警告
抽出更新の失敗	データ品質に関する警告
フロー実行の失敗	データ品質に関する警告
機密データ[1]	機密度

[1] *2023 年 3 月の Tableau Cloud および Tableau Server 2023.1 以前では、機密データラベルはデータ品質警告カテゴリを使用します。*

2023 年 10 月の Tableau Cloud および Tableau Server 2023.3 以降では、管理者は [データラベル] ページのラベル マネージャーまたは REST API を使用して、組み込みラベルをカスタマイズしたり、新しいラベル名とカテゴリを作成したりできるようになりました (Tableau Cloud 管理者は 2023 年 6 月以降、REST API を使用して一部の組み込みラベルを変更し、特定のカテゴリに他のラベルを追加できるようになりました)。詳細については、「データラベルの管理」を参照してください。

ラベルのカテゴリ

ラベルのカテゴリは、ラベルがどこにどのように表示されるか、ダウンストリームのアセットにラベルが表示されるかどうか、どの部分がカスタマイズ可能であるかなどに影響します。

認定

複数のパブリッシャーと大量のアセットが存在するセルフサービス環境では、推奨コンテンツを見つけるのが難しい場合があります。認定を使用すると、信頼できるアセットに印を付けることができ、Tableau 全体のさまざまな場所でアセットにバッジが表示されます。詳細については、「ユーザーが信頼できるデータを見つけられるように認証を使用する」を参照してください。

データ品質警告

問題のあるデータを特定することは、ユーザーとの信頼関係を築くために重要です。データ品質警告を使用すると、問題があることがわかっているデータアセットに印を付けることができます。データ品質警告をアセットに設定すると、そのアセットと、それを使用するダウンストリームのアセットに警告が表示され、データ利用者はソースデータの問題に気づくことができます。たとえば、データベーステーブルを非推奨としてマークすると、そのテーブルに基づくワークブックを表示しているユーザーには警告が表示されます。

さらに、抽出更新またはフロー実行が失敗したときにデータ品質に関する警告を自動的に設定し、成功したときに再度削除することができます。また、[データラベル] ページまたは REST API を使用すると、管理者は、データ品質に関する警告のラベルをカスタマイズして新しく作成することにより、ユーザーが選択できる警告にニュアンスと具体性を追加できます。詳細については、データ品質に関する警告の設定を参照してください。

機密度ラベル

データによっては、異なる取り扱いが必要なものもあります。機密度ラベルを使用すると、データの機密性についての情報をデータの利用者に伝えることができます。機密度のマークをアセットに付加すると、Tableau Cloud を閲覧しているユーザーには、そのアセットと、それを使用するダウンストリームのアセットにバッジが表示されます。たとえば、テーブルの列を機密としてマークすると、そのテーブルに基づいて新しいワークブックを作成しているユーザーには警告が表示されます。さらに、**[データ ラベル]** ページまたは REST API を使用すると、管理者は、カスタマイズされた機密度ラベルを新しく作成することにより、ユーザーが機密度ラベルを使用するときに選択できる分類の範囲にニュアンスと具体性を追加できます。

> 注: 機密度ラベルは 2023 年 6 月の Tableau Cloud と Tableau Server 2023.3 で導入されました。Tableau Cloud と Tableau Server の以前のバージョンでは、専用の機密度ラベルを使用するのではなく、「機密データ」というデータ品質に関する警告を通じてデータの機密性を伝えます。

詳細については、「機密度ラベル」を参照してください。

カスタム ラベルのカテゴリ

場合によっては、認定、データ品質に関する警告、機密度ラベル以外の方法でデータを分類する必要があることがあります。管理者が定義したカスタム カテゴリを使用すると、ラベルを使用して、組織に適した方法でアセットを分類できます。たとえば、組織の管理者は、営業、マーケティング、その他の部門のラベルが付いた「部門」というカテゴリを作成し、サイト上のアセットに適用できるようにするかもしれません。詳細については、カスタム カテゴリのラベルを参照してください。

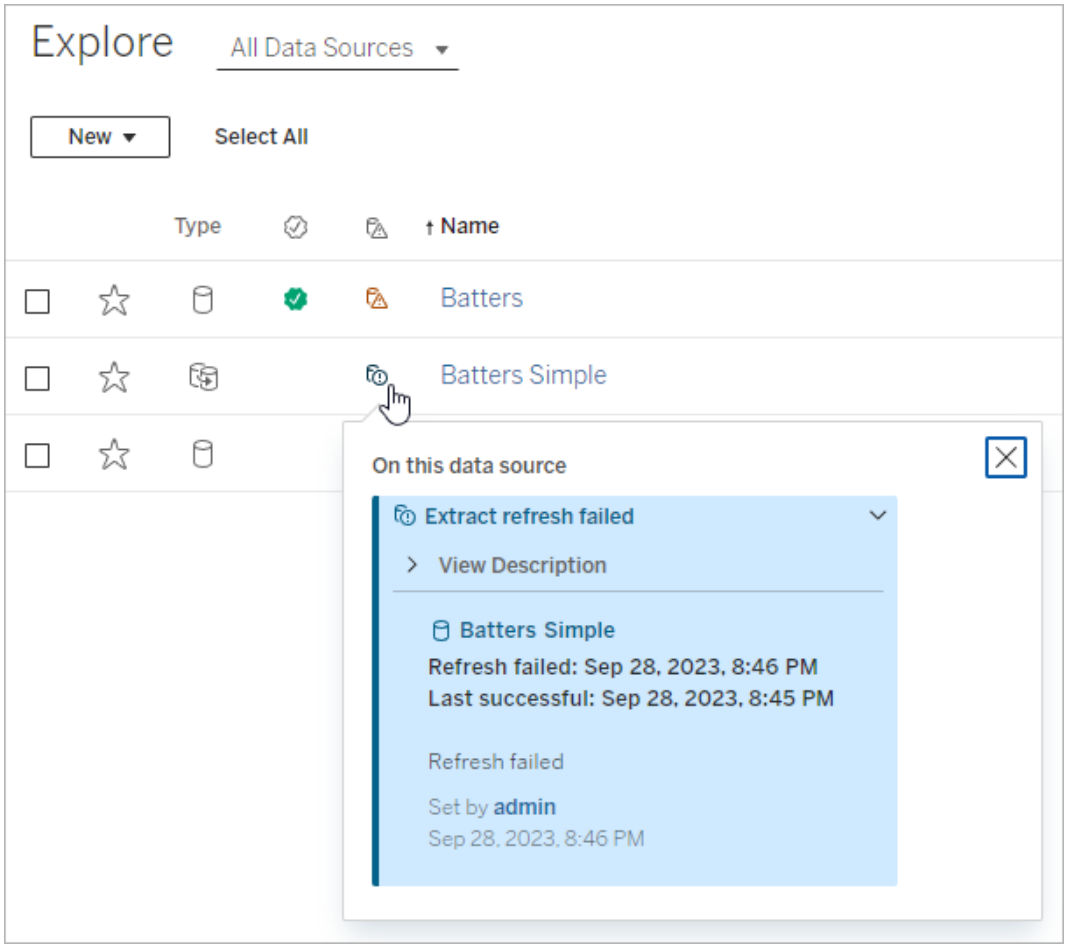
> 注: 管理者がラベル マネージャーを通じてラベル名とカテゴリを作成できる機能は、2023 年 10 月の Tableau Cloud および Tableau Server 2023.3 でリリースされました。Tableau Cloud 管理者は、2023 年 6 月、より限定的な方法で REST API を使用して、組み込みカテゴリを持つカスタム ラベル名を作成できるようになりました。

データ ラベルが表示される場所

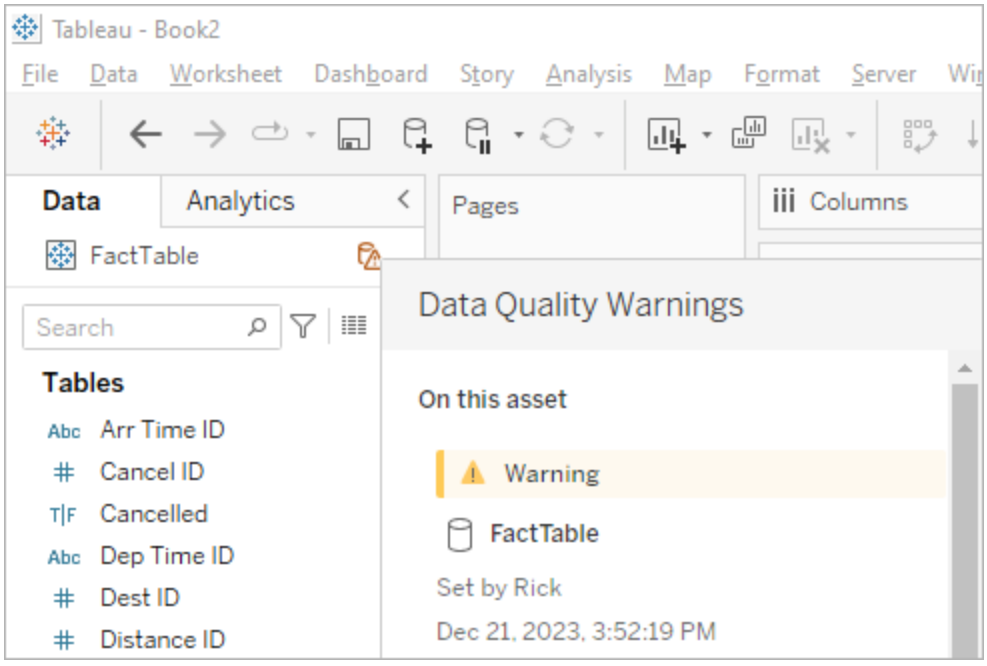
データ ラベルは、次のようなさまざまな場所に表示されます。

- [アセット] ページの上部 (ワークブック、データ ソース、テーブルなど)

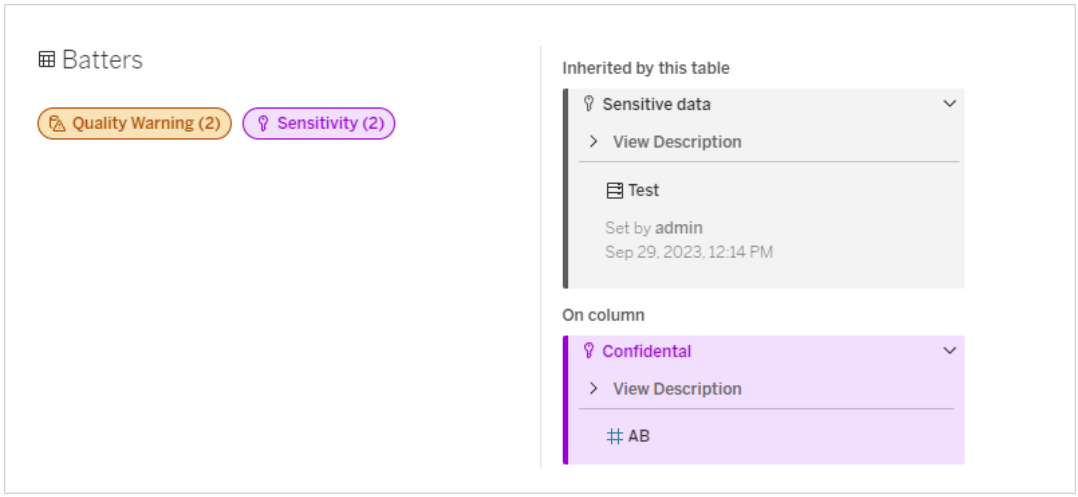
- アセットのリスト (**[探索]** ページ、**[外部 アセット]** ページなど)



- Desktop の **[データ]** ペイン



- Web 作成の **[カタログの詳細]** ウィンドウ



- **[データの詳細]** ペイン

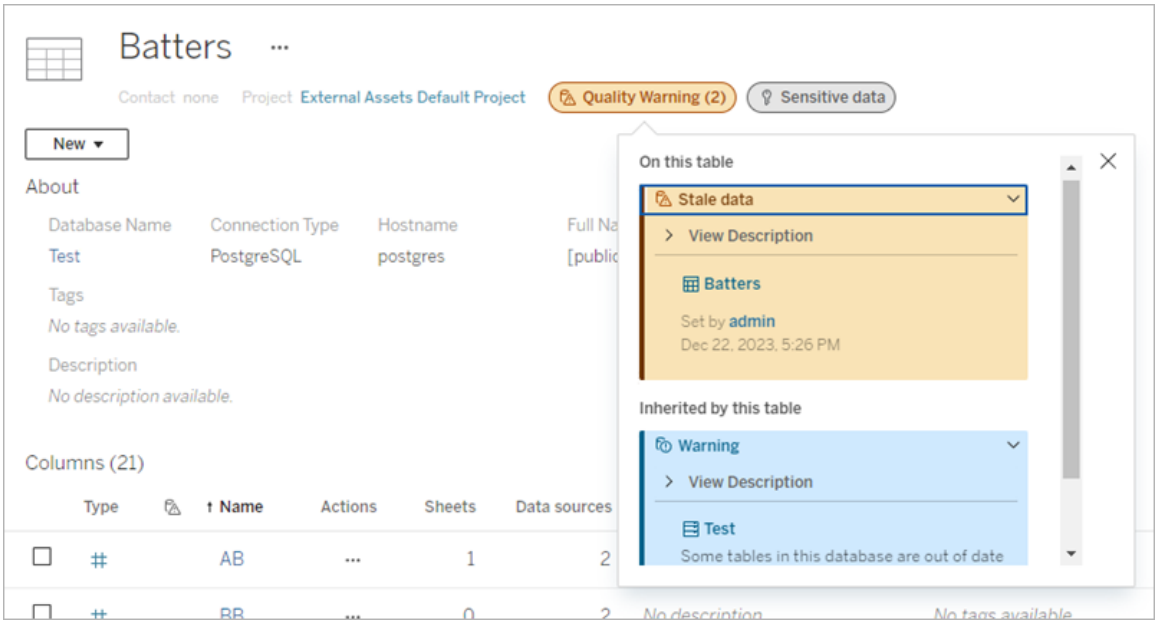
色はラベルのカテゴリと視認性レベルを示します。

- **緑色**は、アセットが認定されていることを示します
- **青色**は、標準的な視認性の品質に関する警告を示します
- **黄色**は、視認性の高い品質に関する警告を示します
- **灰色**は、標準的な視認性の機密度ラベルまたはカスタム カテゴリのラベルを示します
- **紫色**は、視認性の高い機密度ラベルを示します

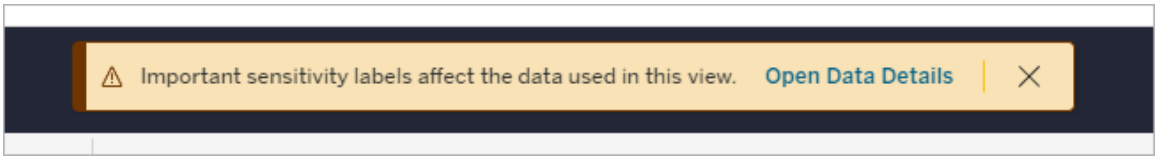
ラベルのバッジを選択すると、アセット上のそのカテゴリのラベルの詳細が表示されます。品質警告ラベルおよび機密度ラベルの場合、詳細にはアップストリームのアセットから継承されたラベルが含まれます。テーブルに関する品質警告ラベルまたは機密度ラベルを調べている場合、詳細にはダウンストリームの列のラベルが含まれます。関連するアセットのページに移動するには、そのアセットを選択します。認証ラベルとカスタム カテゴリ ラベルは、アップストリーム アセットから継承されません。

アセットに付けられているカテゴリ ラベルの数や、アセットによって継承されるラベルの数に関係なく、各ラベル カテゴリに存在するインジケーターは 1 つです。インジケーターが示す 1 つ以上のラベルが視認性の高いラベルである場合、インジケーターには視認性の高いラベルの色が付けられます。たとえば、テーブルに標準的な視認性の品質に関する警告が表示され、テーブルのアップストリーム データベースには視認性の高い品質に関する警告が表示されているとします。このとき、黄

色の**データ品質に関する警告 (2)** インジケーターが表示されます。これは、このインジケーターが 2 つの品質に関する警告を表しており、そのうちの 1 つが視認性が高い警告であるためです。



視認性の高いデータ品質に関する警告と、視認性の高い機密度ラベルがビューや Web 作成セッションに影響を与える場合、アラートが表示されます。

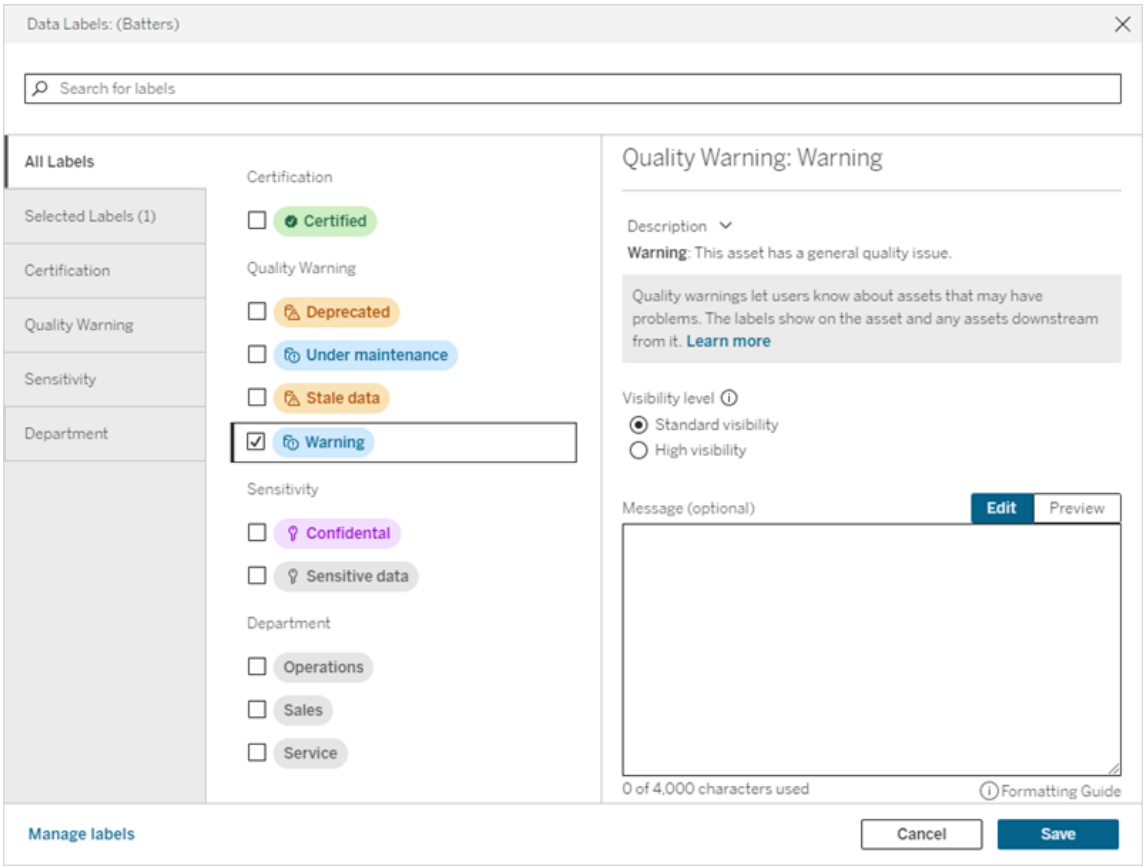


これらのアラートは、表示されているデータを慎重に扱う必要があることをユーザーに通知します。[データの詳細を開く] リンクを選択すると、データに影響を与えるラベルに関する詳細情報が表示されます。

データ ラベル ダイアログ

2024 年 2 月以降の Tableau Cloud では、統合された**データ ラベル** ダイアログを使用して、アセットのデータ ラベルを追加、削除、および変更できます。(抽出更新の監視とフローの更新の監視は、引き続き別のダイアログで制御されます。)

アセットの**データ ラベル**ダイアログを開くには、アセットの横にあるアクション (...) メニューを選択してから、**[データ ラベル]** を選択します。**[Certification (認証)]**、**[データ品質に関する警告]**、**[機密度 ラベル]**、または **[すべてのラベル]** を選択して、適切なタブが表示された**データ ラベル**ダイアログを開きます。



ダイアログの左側にある垂直タブは、リスト上部の **[すべてのラベル]** タブと**[選択したラベル]** タブを除き、データ ラベル カテゴリに対応します。

- **[すべてのラベル]** タブには、カテゴリ全体におけるサイトのすべてのデータ ラベルが一覧表示されます。アセットに対して選択された各ラベルがチェックされます。これらは、ダイアログを開いたときにアセットに対して選択したデータ ラベルや、それ以降に選択したラベルなどです。
- **[選択したラベル]** タブには、アセットに対して選択したすべてのデータ ラベルが一覧表示されます。これらは、ダイアログを開いたときにアセットに対して選択したデータ ラベルや、それ以降に選択したラベルなどです。

- 他のタブはデータラベル カテゴリに対応します。これらのタブには、カテゴリに関連付けられたすべてのデータラベルが一覧表示されます。アセットに対して選択された各ラベルがチェックされます。これらは、ダイアログを開いたときにアセットに対して選択したデータラベルや、それ以降に選択したラベルなどです。

ダイアログの上部にある検索バーは、指定した検索語に一致するデータラベルを返します。結果から必要なラベルを選択または選択解除できます。

アセットのデータラベルを変更するには:

1. 検索バーまたは垂直タブを使用してラベルに移動し、ラベルを見つけます。
2. データラベルをアセットに追加するには、その横にあるチェックボックスをオンにします。
3. 必要に応じて、視認性レベル (該当する場合) とメッセージを追加します。
4. アセットからデータラベルを削除するには、その横にあるチェックボックスをオフにします。
5. これらのステップを繰り返して、アセットのラベルを追加、削除、または変更します。
6. アセットのデータラベルへの変更をコミットしてダイアログを閉じるには、**[保存]** をクリックします。または、ダイアログが開いてから加えたすべての変更を破棄する場合は、**[キャンセル]** をクリックして、変更の破棄を確定します。

> 注: データラベルの横にあるチェックボックスの代わりにデータラベルを選択すると、チェックボックスのステータスを変更せずにラベルの詳細を表示することができます。この操作は、アセットのラベルのステータスを変更せずに、ラベルの説明を確認したり、メッセージを変更したりする場合に便利です。

統合されたデータラベル ダイアログは Tableau Server では使用できません。

特定のカテゴリのラベルの詳細については、該当するトピックを参照してください。

- ユーザーが信頼できるデータを見つけられるように認証を使用する
- データ品質に関する警告の設定
- 機密度ラベル
- カスタム カテゴリのラベル

アセットのデータラベルを操作するために必要なパーミッション

アセットのラベルを表示、追加、更新、削除するには、次のパーミッションが必要です。

- データラベルを表示するには、関連するアセットに対する**読み取り**権限が必要です。
- 認定ラベル以外のデータラベルを追加、更新、削除するには、関連するアセットに対する**書き込み**権限が必要です。
- 認定ラベルを追加、更新、削除するには、管理者であるか、アセットが含まれているプロジェクトのプロジェクトリーダーまたは製品所有者である必要があります。
- プロジェクトにない外部アセットの認定ラベルを追加、更新、削除するには、関連するアセットに対する**変更**権限が必要です。

データラベルとタグの比較

Tableau Cloud とTableau Server には、アセットを分類するための別のソリューションである、タグも搭載されています。データラベルとタグは大きく異なります。

領域	**データラベル**	**タグ**
構造と制御	管理者がデータラベルの範囲を制御します	ユーザーが追加するタグの範囲を制御して管理することはできません
パーミッション	アセットのパーミッションで、データラベルの追加/更新/削除の機能を制御します	Explorer とCreator は、表示できるあらゆるアセットにタグを付けることができます
外観	データラベルのアイコンは見やすく、カテゴリと視認性レベルごとに色分けされています	タグが表示される場所はデータラベルよりも少なく、そのアイコンもありません
継承	一部のデータラベル (警告ラベルや機密ラベルなど) は、下流のアセットに表示されます	継承はありません
検索/フィルタリング	認証とデータ品質に関する警告は、一部のアセットリストでフィルターとして使用できます	検索結果は一致するタグを持つアセットを返し、タグは一部のアセットリストでフィルターとして使用できます
API でのアクセス可否	REST API とメタデータAPI 経由でアクセス可能	REST API とメタデータAPI 経由でアクセス可能

ライセンスの要件	データ管理ライセンスが必要です (パブリッシュされたデータソースの認証を除く)	ライセンス要件はありません
使用	ユーザーによるデータの使用に影響を与える可能性のある、重要な情報の提供に重点を置いた構造化された分類	アセットを分類するための無制限の方法

タグの詳細については、Tableau Desktop とWeb 作成ヘルプの「タグの使用」を参照してください。

ユーザーが信頼できるデータを見つけられるように認証を使用する

複数のパブリッシャーが存在するセルフサービス環境では、通常、似たような名前のコンテンツ、同じまたは類似した参照元データに基づくコンテンツ、あるいは説明情報がないままパブリッシュされているコンテンツなど、さまざまなコンテンツが Tableau Server 上のプロジェクトに含まれています。その場合、アナリストは使用するデータについて自身を持てない場合があります。

ユーザーが自分の分析タイプに対して推奨される信頼できるデータを簡単に見つけられるように、組織のデータ基準に準拠するデータを*認証*することができます。

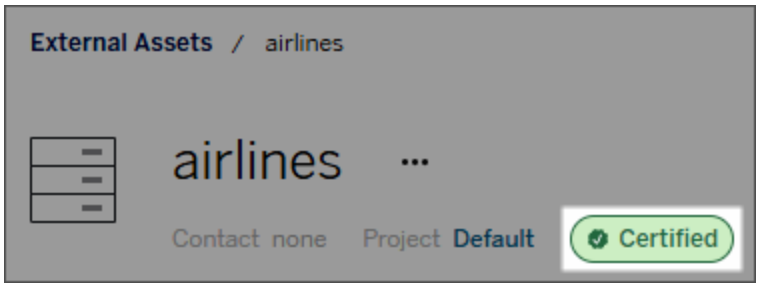
認定はデータをキュレーションを通して推進する方法を提供することで推奨トレーニングのスケジュール機能を補完します。

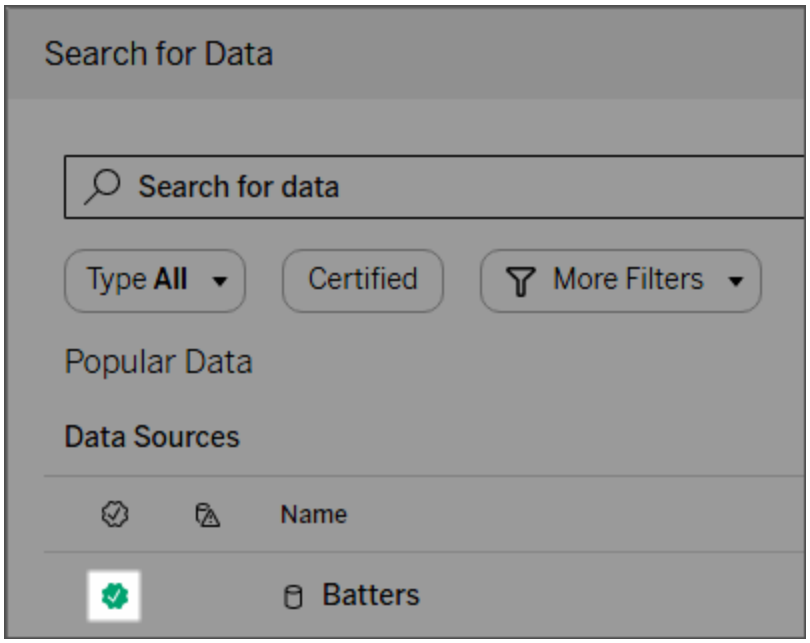
パブリッシュされたデータソースの認証に加えて、Tableau Server または Tableau Cloud のデータ管理 がある場合:

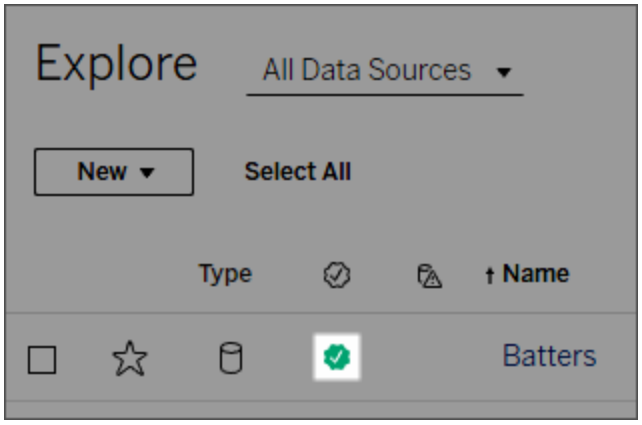
- Tableau Catalog が有効になっている場合、Tableau コンテンツに関連付けられているデータベースとテーブルを認証できます。(Tableau Catalog の詳細については、Tableau Server または Tableau Cloud ヘルプの「Tableau Catalog について」を参照してください。)
- Tableau 2022.1 以降では、仮想接続と仮想接続テーブルを認証できます。

認証はどのように信頼できるデータをユーザーから見つけやすくするか

アセットを認証する場合、アセットが表示されている場所に応じて、ユーザーに緑色のバッジまたは緑色のチェックマークが表示されます。

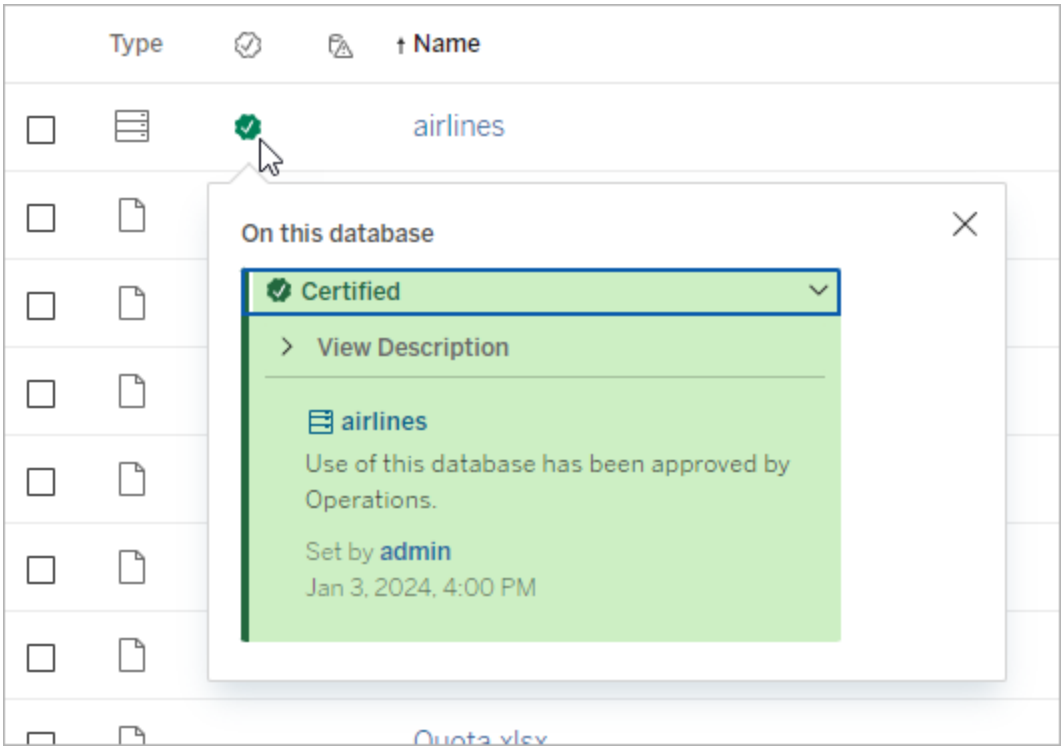






認定されたデータソースは検索結果でより上位にランク入りし、推奨されるデータソースに加えられます。

さらに、認証状態についての注記を設定できます。この注記は、ユーザーがバッジをクリックしたときに表示されるか、Web 作成や Tableau Desktop でデータソースアイコンにカーソルを合わせたときにツールヒントに表示されます。また、データソースが誰によって認証されているかについての情報も表示されます。



認定するデータの選出ガイドラインを作成する

その他の Tableau 機能と同様、認定には柔軟性があります。アセットの認定を決定する条件を自分のオーガナイゼーションで定義することができます。その際、ガイドラインを文書化して共有します。ガイドラインがあることで、自分や他の管理者、またはプロジェクトリーダーが認定対象を選択する際に、一貫性を持たせることができます。ガイドラインは、ユーザーが認定の意味を理解するためにも役立ちます。

全プロジェクトで同じ認定条件を使用していても、各プロジェクトで固有の条件を定義していても、重要なのは認定が今いる環境で何を意味するのかです。

データを認定できる人物

データソースを認定するには、次のことが必要です:

- サーバー管理者またはサイト管理者である、*または*
- **Explorer (パブリッシュ可能)** または **Creator** のサイトロールを持っており、かつ、認証するデータを含むプロジェクトのプロジェクト所有権または**プロジェクト リーダー**権限を持っている必要があります。

仮想接続と仮想接続テーブルを認証するには、使用する環境にデータ管理 ライセンスがある必要があります。また、

- サーバー管理者またはサイト管理者である、*または*
- **Explorer (パブリッシュ可能)** または **Creator** のサイトロールを持っており、かつ、認証するデータを含むプロジェクトのプロジェクト所有権または**プロジェクト リーダー**権限を持っている必要があります。

データベースまたはテーブルを認証するには、使用する環境で Tableau Catalog が有効になっており、

- サーバー管理者またはサイト管理者である、*または*
- データベースまたはそのデータベース内のテーブルを認証するために、データベースに対して**パーミッションを設定する**権限を持っている必要があります。

データの認定方法

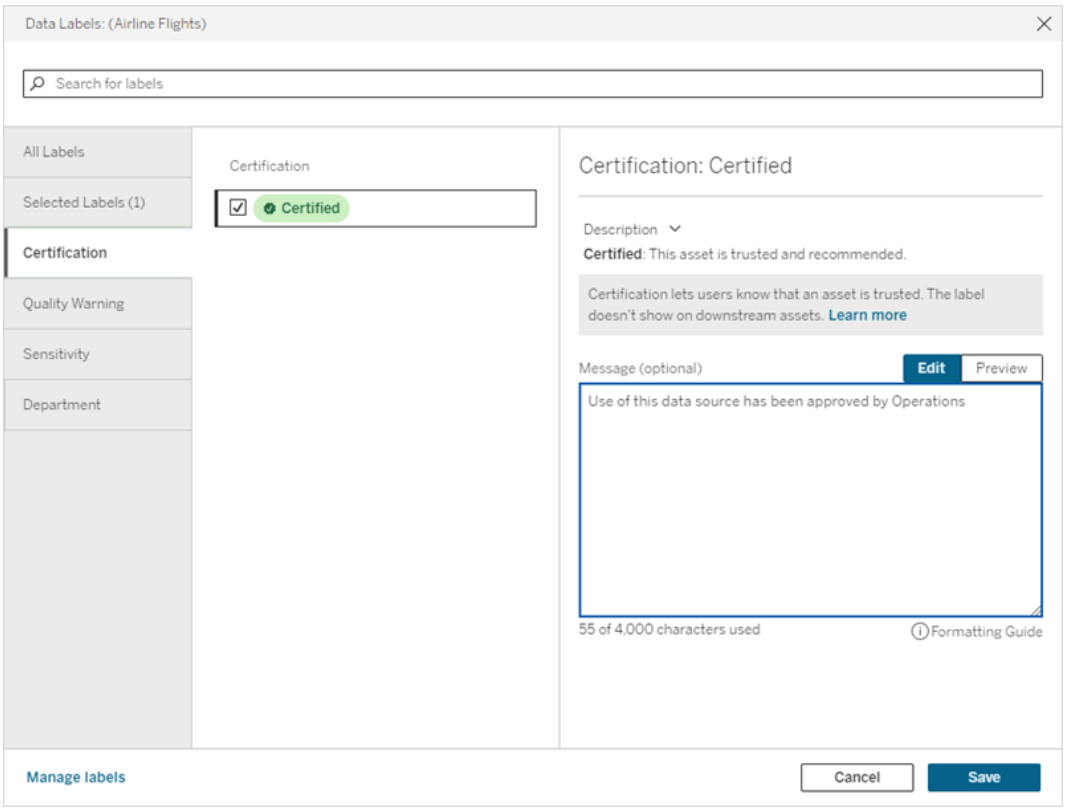
認定できるデータは、付与されているパーミッションや、データ管理 ライセンスを持っているかどうか、および Tableau Catalog がお使いの環境で有効になっているかどうかによって変わります。

- パーミッションを持つすべてのユーザーが、データソースを認定できます。
- データ管理 ライセンスがある場合、パーミッションを持つユーザーは、仮想接続および仮想接続テーブルを認定することもできます。
- データ管理 ライセンスを持っていて、Tableau Catalog が有効になっている場合、パーミッションを持つユーザーはデータベース、テーブル、ファイルを認定することもできます。

アセットを認定するには:

> 注: Tableau Cloud 2024 年 2 月以降、および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータ ラベル ダイアログを使用して証明書の追加と削除を行うようになりました。データ ラベル ダイアログの詳細については、「データ ラベル ダイアログ」を参照してください。

1. アセットを検索するか、アセットに移動します。移動するステップは、認定するアセットのタイプによって異なります。
    - データソースまたは仮想接続 - **[探索]** ページで、**[すべてのデータ ソース]** または **[すべての仮想接続]** を選択します。
    - 仮想接続テーブル - **[探索]** ページで、**[すべての仮想接続]** を選択し、認証する仮想接続テーブルを含む仮想接続を選択します。次に、仮想接続テーブルを選択します。
    - データベースまたはテーブル - **[探索]** ページで、データベースまたはテーブルに移動します。または **[外部アセット]** ページで、**[データベースとファイル]** または **[テーブルとオブジェクト]** を選択します。
2. ページで、認定するアセット名の隣にある [その他のアクション] メニュー (...) を選択します。
3. **[データ ラベル] > [Certification (認証)]** (または Tableau Server 2023.1 以前では **[Edit Certification (認証の編集)]**) を選択します。
4. **[認定済み]** チェックボックスをオンにします。(以前のバージョンの Tableau Server では、切り替えを使用します)。
5. 必要に応じてメッセージを追加します。メッセージを通じて、認証ステータス、データの使用目的、その他の有用な情報をユーザーに提供します。**[Message (メッセージ)]** セクションに追加する情報は、「認証はどのように信頼できるデータをユーザーから見つけやすくするか」に前述されている認証バッジまたはツール ヒントに表示されます。メッセージのテキストの書式を太字、下線、斜体に設定したり、リンクや画像を含めたりすることもできます。テキストの書式設定のヒントを表示するには、**[保存]** ボタンの上にある情報 (i) アイコンをクリックします。(Tableau Cloud 2024 年 2 月以降、メッセージの使用は任意です。以前のバージョンの Tableau Cloud および Tableau Server では必須でした。)



6. **[保存]** を選択します。

認証をカスタマイズする

Tableau Cloud 2023 年 6 月および Tableau Server 2023.3 以降、管理者は [データ ラベル] ページのラベル マネージャーまたは REST API を使用して、証明書ダイアログでユーザーに表示される証明書の説明を変更できるようになりました。詳細については、「データ ラベルの管理」を参照してください。

データ品質に関する警告の設定

データ品質の警告は、Tableau Server および Tableau Cloud のデータ管理 オファリングの一部である、Tableau Catalog の機能です。Tableau Catalog の詳細については、Tableau Server または Tableau Cloud ヘルプの「Tableau Catalog について」を参照してください。

お使いの環境で Tableau Catalog が有効になっている場合、データアセットにデータ品質の警告を設定して、データのユーザーが問題を認識できるようにすることができます。たとえば、ユーザーに

対して、データソースが廃止されていること、または抽出データソースの更新が失敗したことを知らせる必要があるとします。

データ品質の警告は、データソース、データベース、テーブル、フロー、仮想接続、仮想接続テーブル、および列に対して設定できます。

データソース、データベース、テーブル、およびフローのデータ品質の警告は、バージョン 2019.3 の Tableau Cloud と Tableau Server で導入されました。仮想接続と仮想接続テーブルのデータ品質の警告は、2022 年 3 月の Tableau Cloud と Tableau Server 2022.1 で追加されました。列のデータ品質の警告は、2022 年 10 月の Tableau Cloud と Tableau Server 2022.3 で追加されました。

データ品質に関する警告について

データ品質に関する警告には次の 2 種類があります。1 つはユーザーが設定する品質に関する警告であり、もう1 つは 抽出を更新するときまたはフロー実行が失敗したときの品質に関する警告 (品質に関する警告の監視とも呼ばれます) で、Tableau が設定します。

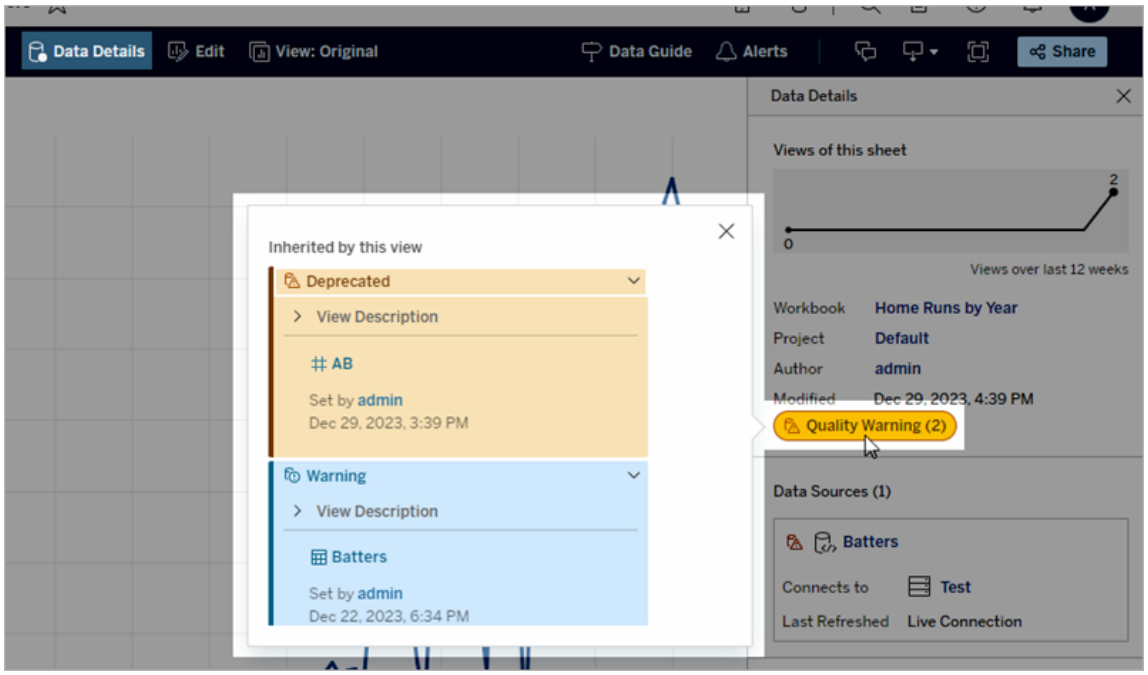
- **ユーザーが設定する品質に関する警告**: ユーザーが設定する品質に関する警告は、ユーザー自身または別のユーザーが削除するまでユーザーに表示されます。
- **品質に関する警告の監視**: 抽出またはフロー アセットの品質に関する警告の監視を有効にすると、抽出を更新するときまたはフロー実行が失敗したときに、Tableau によって品質に関する警告が自動的にアセットに追加されます。その後、抽出更新またはフロー実行が成功すると、Tableau は品質に関する警告を自動的に削除します。

2023 年 10 月の Tableau Cloud と Tableau Server 2023.3 以降では、アセットレベルで監視警告を設定することに加えて、抽出更新とフロー実行の監視をサイト全体で一度にオンまたはオフにすることもできます。サイト全体の監視の詳細については、抽出更新とフロー実行の失敗をサイト全体で監視を参照してください。

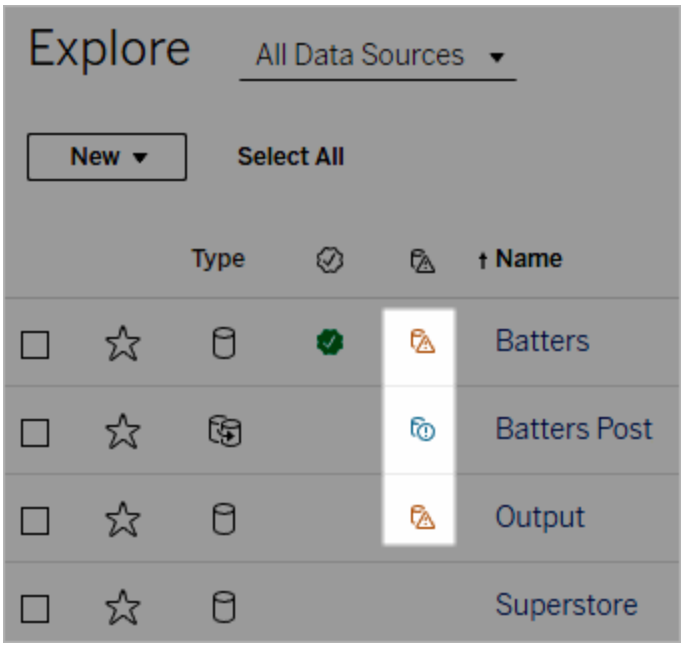
注: REST API を使用して、両方の種類のデータ品質に関する警告を有効または変更することができます。詳細については、Tableau REST API ヘルプの「メタデータの メソッド」でラベルとデータ品質に関する警告を参照してください。

データ品質に関する警告が表示される場所

Tableau Cloud および Tableau Server では、データ ソース、フロー、データベース、テーブル、列、仮想接続、または仮想接続テーブルに警告を設定すると、その警告は、アセットおよびそのアセットからダウンストリームにあるアセットのユーザーに表示されます。たとえば、あるテーブルに設定されている警告セットは、そのテーブルに依存しているダッシュボードを見ているユーザーに表示されます。ユーザーには、ダッシュボードの [データの詳細] タブに警告アイコンが表示され、そのペインを開くと詳細を確認できます。



リスト ビューでいくつかのタイプのコンテンツを探索すると、データ品質に関する警告が表示されます。



データ品質に関する警告は、アセットページの上部にも表示されます。

Tableau Desktop では、次の場合に **[データ]** ペインのデータソースの横にアイコンが表示されます。

- ワークブックで使用しているデータソースに警告がある
- ワークブックで使用しているデータソースのアップストリームに警告がある

注: 列および仮想接続のデータ品質に関する警告は、Tableau Desktop には表示されません。

警告の詳細を表示するには、警告アイコンを選択します。

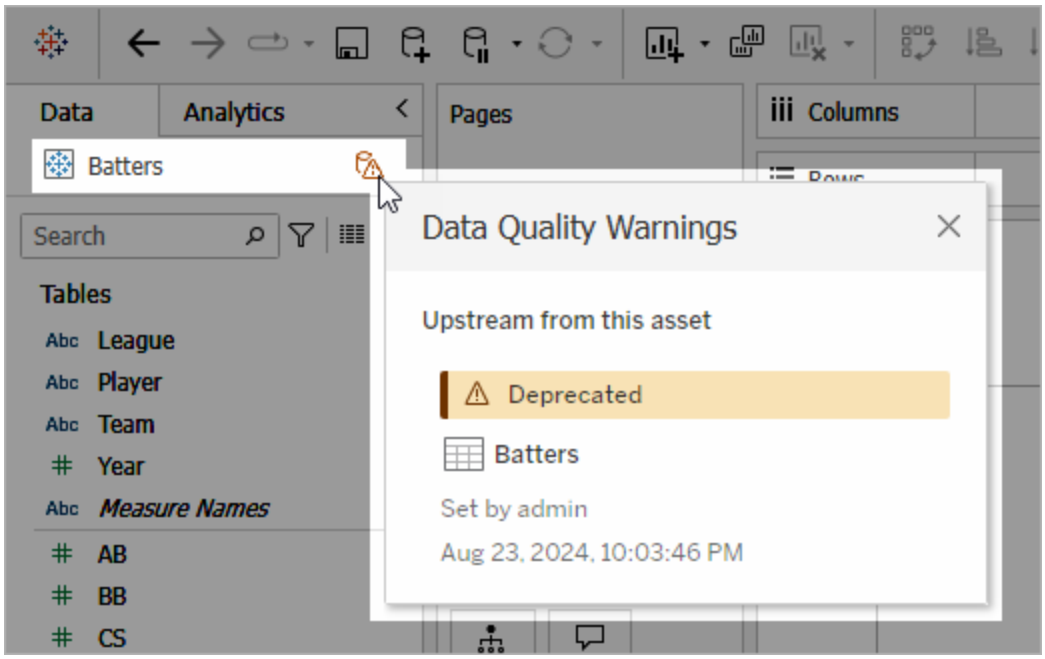
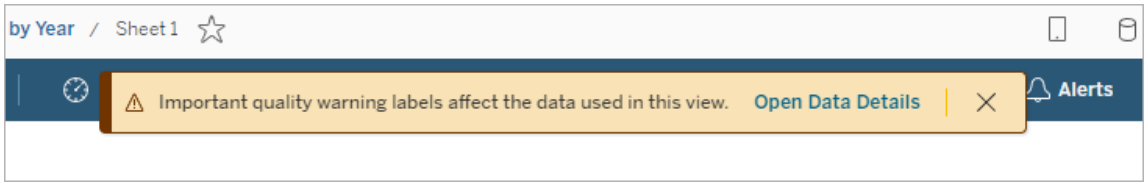


Tableau Cloud の Web 作成では、データ接続またはフィールドで **[カタログの詳細]** を選択することで、[データ] ペイン (データ接続またはフィールド) 内のオブジェクトのすべてのラベルのプロパティを確認できます。

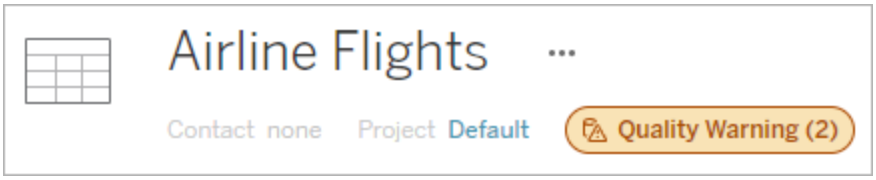
視認性

データ品質に関する警告は、視認性の高い警告として分類できます。視認性の高い品質に関する警告は、より緊急性が高く、より多くの場所に表示されます。たとえば、データソースに関する視認性の高い警告は、誰かがそのデータソースに依存するビューを開いたときに通知を生成します。



視認性の高い品質に関する警告の場合、アイコンとバッジは黄色またはオレンジ色ですが、標準的な視認性の警告は青色です。

1 つ以上の品質に関する警告がアセットに適用される場合 (アセットに品質に関する警告が複数あるため、またはアップストリームの警告を継承するため)、バッジには番号が含まれ、色は最も視認性の高いラベルの色になります。たとえば、品質に関する警告が 2 つアセットに適用されていて、1 つが標準的な視認性の警告で、もう1 つが視認性の高い警告の場合、バッジは黄色またはオレンジ色になります。



サブスクリプションのデータ品質に関する警告

管理者は、サブスクリプションでデータ品質に関する警告をオンにすることができます。この機能がオンになっている場合、ユーザーが受信するメールには、そのビューに関する視認性の高いデータ品質に関する警告と、次のリンクが含まれます。

- **[データの詳細]** ペインが開いている状態の関連するビューまたはワークブックへのリンク
- データ ソース、テーブル、データベースなど、関連するアップストリーム アセットへのリンク

管理者は、Tableau Server または Tableau Cloud サイトの設定ページで **[ビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベル]** オプション (以前は **[サブスクリプションのデータ品質に関する警告]** オプションと呼ばれていました) を選択することにより、メール サブスクリプションのデータ品質に関する警告をオンにできます。詳細については、サイト設定 リファレンスのビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベルを参照してください。

データ品質に関する警告の設定方法

アセットには複数の異なるデータ品質に関する警告を設定できます。2023 年 6 月の Tableau Cloud と Tableau Server 2023.3 以降では、管理者はデータ ラベルをカスタマイズすることで、利用可能なデータ品質に関する警告のリストに追加できるようになりました

2023 年 6 月の Tableau Cloud と Tableau Server 2023.3 以降では、「機密データ」はデータ品質に関する警告ではなくなり、代わりに機密度ラベルになりました。詳細については、「機密

> 度ラベル」を参照してください。Tableau Server 2023.1 以前のバージョンでは、「機密データ」はデータ品質に関する警告のままです。

データ品質に関する次の警告が組み込まれています。

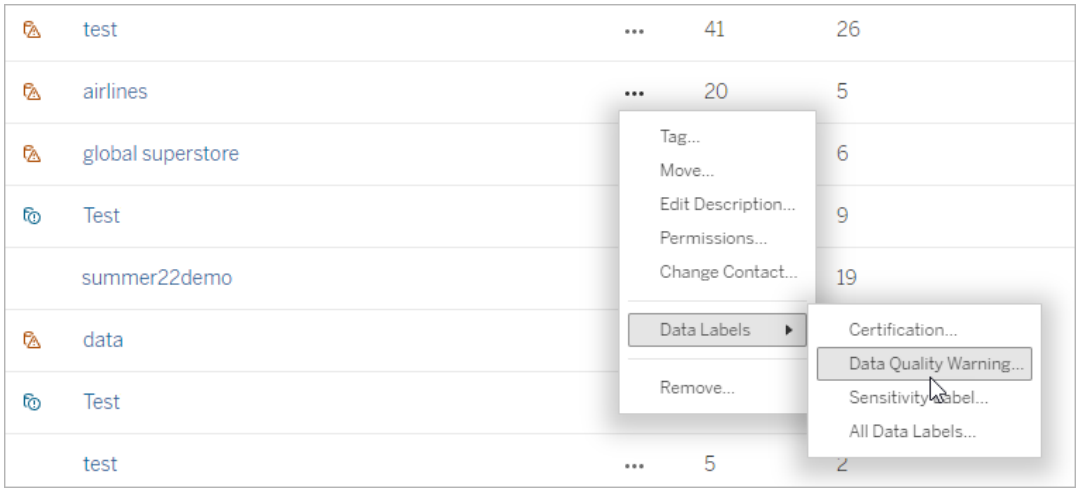
- 警告
- 非推奨
- 古いデータ
- メンテナンス中

アセットにデータ品質に関する警告を設定するには、次の操作を行います。

> 注: Tableau Cloud 2024 年 2 月および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータ ラベル ダイアログを使用して、品質に関する警告の追加と削除を行うようになりました。データ ラベル ダイアログの詳細については、「データ ラベル ダイアログ」を参照してください。

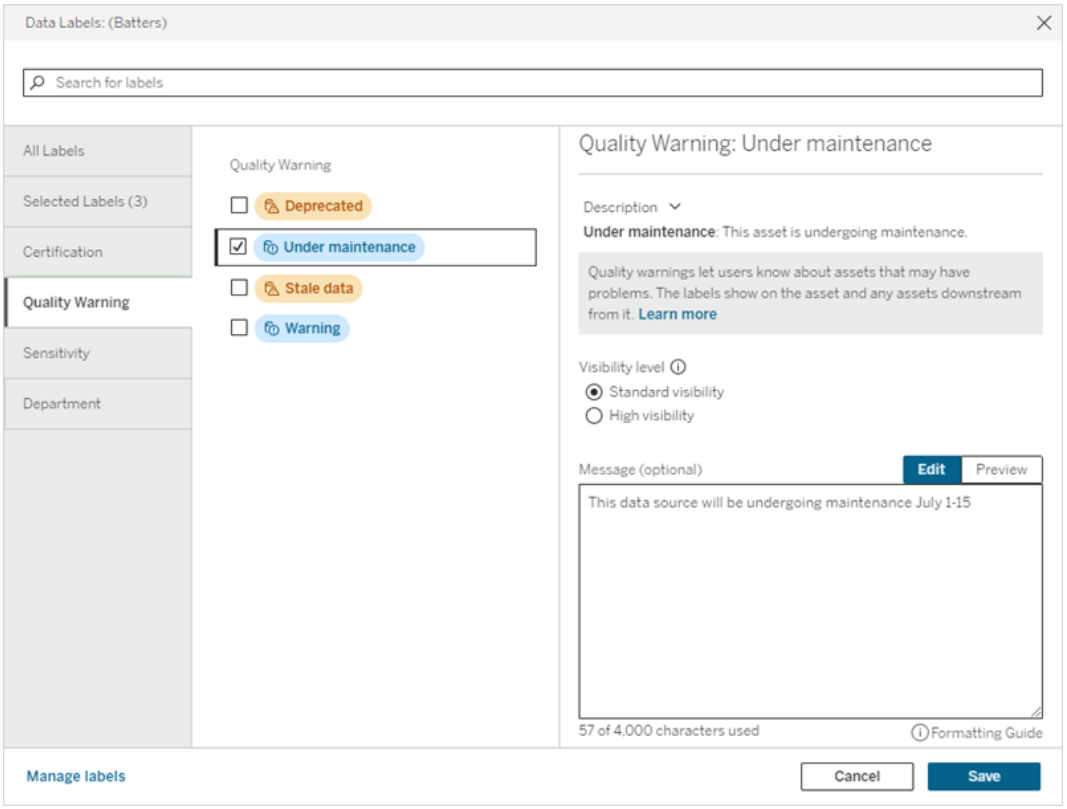
1. アセットを検索するか、アセットに移動します。移動するステップは、データ品質に関する警告を追加するアセットのタイプによって異なります。
   - データ ソースまたは仮想接続 - **[探索]** ページで、**[すべてのデータ ソース]** または **[すべての仮想接続]** を選択します。
   - 仮想接続テーブル - **[探索]** ページで、**[すべての仮想接続]** を選択し、認証する仮想接続テーブルを含む仮想接続を選択します。次に、仮想接続テーブルを選択します。
   - データベースまたはテーブル - **[探索]** ページで、データベースまたはテーブルに移動します。または **[外部アセット]** ページで、**[データベースとファイル]** または **[テーブルとオブジェクト]** を選択します。
   - 列 - **[探索]** ページで、テーブルに移動します。または **[外部アセット]** ページで **[テーブルとオブジェクト]** を選択して、テーブルに移動します。次に、リスト内で列を見つけます。
2. アセットの横にある [アクション メニュー(...) を選択し、**[データ ラベル] > [データ品質に関する警告]** の順に選択します。(Tableau Server 2022.3 以前の列の場合は、代わりに列を選

択してから、アクションのドロップダウンをクリックして **[品質に関する警告]** を選択します)。



3. アセットにアタッチする品質に関する警告の横にあるチェックボックスをオンにします。品質に関する警告の名前がわかっている場合は、ダイアログの上部で名前を検索し、その横にあるチェックボックスを選択できます。(Tableau Server 2023.3 以前では、各アセットにアタッチできる品質に関する警告は 1 つだけです。**[警告の表示]** スイッチ、または **[警告の有効化]** チェックボックスを使用して、そのアセットのデータ品質に関する警告をオンにし、ドロップダウンリストから目的の警告を選択します)。
4. 視認性レベルを設定します。
5. 必要に応じて、ユーザーに表示するメッセージを入力します。(Tableau Server 2023.3 以前では、メッセージが必要です。) メッセージのテキストの書式を太字、下線、斜体に設定したり、リンクや画像を含めたりすることもできます。テキストの書式設定のヒントを表示するに

は、**[保存]** ボタンの上にある情報 (i) アイコンをクリックします。



6. **[保存]** を選択します。

データ品質に関する警告を削除する

警告が適用されなくなった場合は、その警告を使用しているデータアセットに移動して削除できます。

> 注: Tableau Cloud 2024 年 2 月および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータラベル ダイアログを使用して、品質に関する警告の追加と削除を行うようになりました。データラベル ダイアログの詳細については、「データラベル ダイアログ」を参照してください。

1. アセットの横にある [アクション メニュー(...)] を選択してから、**[データ品質に関する警告]** を選択します(Tableau Server 2022.3 以前の列の場合は、代わりに列を選択してから、アクションのドロップダウンをクリックして **[品質に関する警告]** を選択します)。

2. アセットから削除したい品質に関する警告の横にあるチェックボックスをオフにします。(Tableau Server 2023.3 以前では、**[警告の表示]** スイッチ、または **[警告の有効化]** チェックボックスを使用して、そのアセットのデータ品質に関する警告をオフにします)。
3. 警告をオフにします。
4. **[保存]** を選択します。

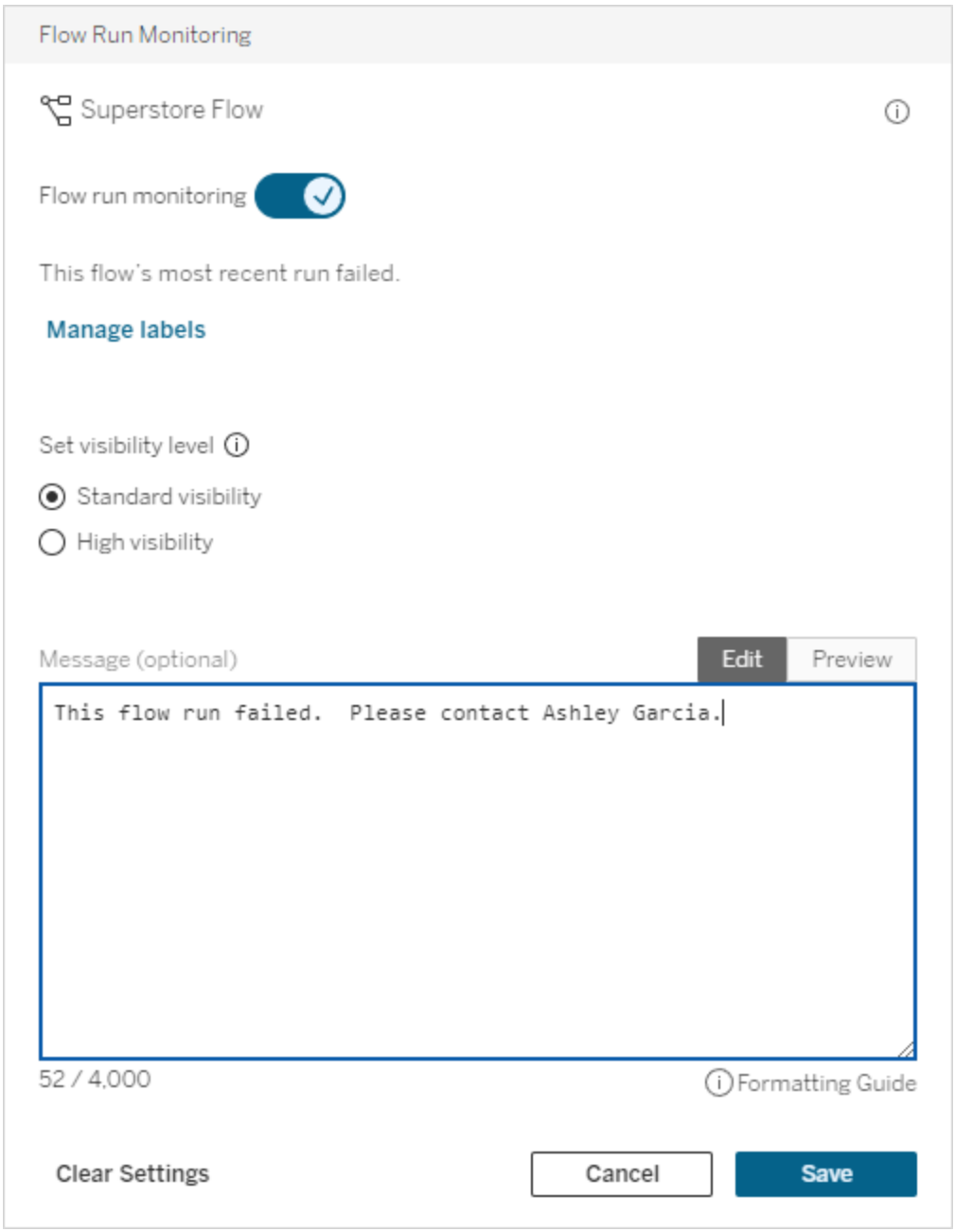
品質に関する警告の監視をオンにするには

抽出データソースの更新失敗とフロー実行の失敗の 2 つのイベントを監視するように Tableau を設定できます。イベントが発生すると、Tableau によってデータ品質に関する警告が生成され、手動で表示する場合と同じ場所に表示されます。

抽出またはフローで明示的に監視を有効にすることも、2023 年 10 月の Tableau Cloud と Tableau Server 2023.3 以降で、すべての抽出更新とフロー実行の失敗に対してサイト全体の監視を有効にすることもできます。サイト全体の監視の詳細については、抽出更新とフロー実行の失敗をサイト全体で監視を参照してください。

抽出更新またはフロー実行の失敗を明示的に監視するには、次の操作を行います。

1. 警告を作成する抽出データソースまたはフローの横にあるアクション メニュー (...) を選択し、次の中から適切なオプションを選択します。
    - Tableau Cloud および Tableau Server 2023.3 以降の場合:
        - **[データ ラベル] > [抽出更新の監視]**
        - **[データ ラベル] > [フロー実行の監視]**
    - Tableau Server 2023.1 以前の場合:
        - **[データ品質に関する警告] > [抽出更新の監視]**
        - **[データ品質に関する警告] > [フロー実行の監視]**
2. 警告を有効にします。
3. 視認性レベルを設定します。(古いバージョンのダイアログには、視認性を高めるためのチェックボックスがあります)。
4. 必要に応じて、抽出更新またはフローの実行に失敗した場合に、ユーザーに表示されるメッセージを警告の詳細に入力します。メッセージのテキストの書式を太字、下線、斜体に設定したり、リンクや画像を含めたりすることもできます。テキストの書式設定のヒントを表示するには、**[保存]** ボタンの上にある情報 (i) アイコンをクリックします。
5. **[保存]** をクリックします。



品質に関する警告の監視をオフにするには

抽出更新またはフロー実行の失敗のいずれかの監視をオフにするには、次の操作を行います。

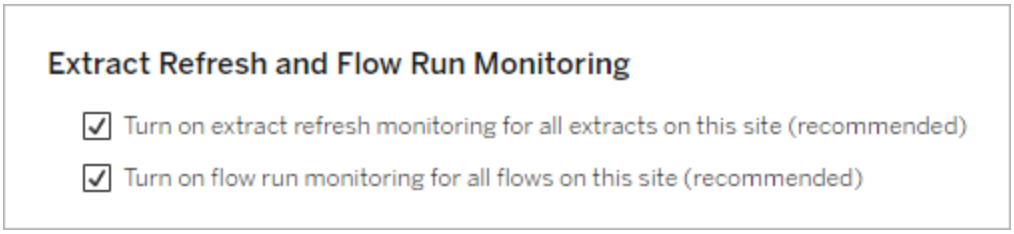
1. 警告を作成する抽出データソースまたはフローの横にあるアクション メニュー (...) を選択し、次の中から適切なオプションを選択します。

- Tableau Cloud および Tableau Server 2023.3 以降の場合:
  - **[データ ラベル] > [抽出更新の監視]**
  - **[データ ラベル] > [フロー実行の監視]**
- Tableau Server 2023.1 以前の場合:
  - **[データ品質に関する警告] > [抽出更新の監視]**
  - **[データ品質に関する警告] > [フロー実行の監視]**

2. 警告をオフにします。
3. **[保存]** をクリックします。

抽出更新とフロー実行の失敗をサイト全体で監視

2023 年 10 月の Tableau Cloud と Tableau Server 2023.3 以降、管理者はサイト全体の監視を有効にして、抽出更新の失敗やフロー実行の失敗に対するデータ品質に関する警告を追加または削除できるようになりました。この機能は、[設定] ページの [抽出更新とフロー実行の監視] セクションで制御できます。



これらの設定は、すべての新しいサイトで既定でオンになっています。変更前に存在していたサイトでは設定がオフになりますが、管理者は設定をオンにすることができます。

注: データ品質に関する警告の通知は、Tableau Bridge を使用する抽出更新では表示されません。

サイト全体の監視と明示的な監視の関係

アセットの監視を明示的に行う設定とサイト全体で行う設定の関係は次のとおりです。

- アセットの監視が明示的に有効になっており、かつ、サイト全体でも有効になっている場合、そのアセットの明示的な設定がサイト全体の設定より優先されます。これらの設定には、視認性レベルやメッセージなどのプロパティが含まれます。

- サイト全体の監視をオフにすると、次のようになります。
  - 監視が明示的に有効になっているアセットは変更されません。
  - 監視が明示的に有効になっていないアセットは、抽出更新またはフロー実行の失敗の監視を停止し、それらのアセットで抽出更新やフロー実行の失敗から以前発生した警告は削除されます。
  - 警告ラベルが付けられなくなったアセットをカタログがもう一度取り込むため、カタログ取り込みのパフォーマンスが一時的に低下する可能性があります。

サイト全体の監視は、2023 年 10 月の Tableau Cloud と Tableau Server 2023.3 でリリースされました。以前のバージョンでは、明示的な監視とサイト全体の監視に相互関係はありません。

データ品質に関する警告を設定できるユーザー

データ品質に関する警告を設定するには、以下のいずれかの条件を満たす必要があります。

- サーバー管理者またはサイト管理者である
- アセットに**上書き**できる権限がある

データ品質に関する警告のカスタマイズ

2023 年 6 月の Tableau Cloud と Tableau Server 2023.3 以降では、管理者は [データ ラベル] ページのラベル マネージャーまたは REST API を使用して、[データ品質に関する警告] ダイアログでユーザーに表示されるデータ品質に関する警告を変更したり、新しい警告を作成したりすることができます。詳細については、「データ ラベルの管理」を参照してください。

機密度ラベル

データによっては、より慎重に扱わなければならないものがあります。信頼とセキュリティを確保するには、それがどのデータなのかをユーザーが認識できることが重要です。2023 年 6 月の Tableau Cloud および Tableau Server 2023.3 以降、データ管理 ライセンスがあれば Tableau の新しいデータ ラベルのカテゴリ*「機密度 ラベル」*を利用できます。機密度 ラベルを使用すると、ビューの作成時や情報の共有時に注意すべき取り扱いのレベルを示すことができます。さらに、機密度 ラベルは、同じアセット上で、認定やデータ品質警告などの他のラベルと共存できます。管理者は、**[データ ラベル]** ページのラベル マネージャーまたは REST API を使用して、新たな機密度 ラベルを組織のニーズに合わせて作成することができます。

> 注: 2023 年 3 月の Tableau Cloud および Tableau Server 2023.1 以前では、データの機密度はデータ品質に関する警告「機密データ」を使用して表現されていました。2023 年 6 月の Tableau Cloud および Tableau Server 2023.3 へのアップグレードにより、データ品質に関する警告「機密データ」は機密度ラベルに移行されました。

機密度ラベルを設定できるアセットの種類は、他のデータ ラベルを付加できるアセットの種類と同じです。

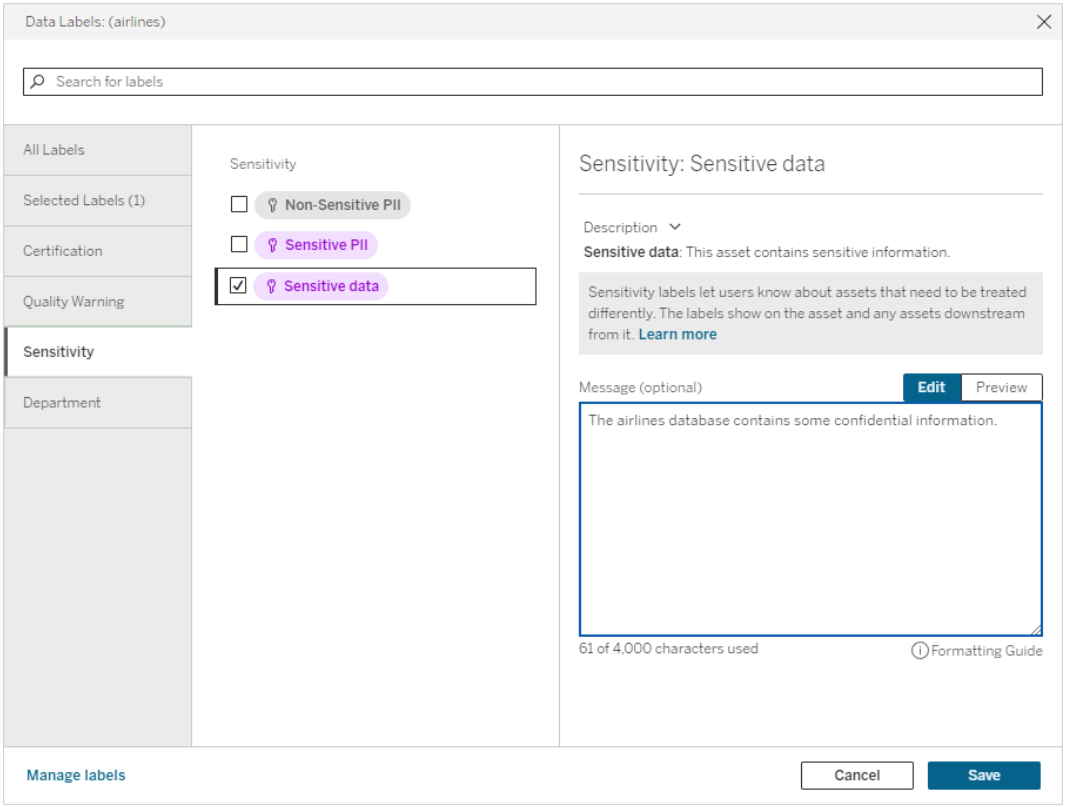
アセットへの機密度ラベルの設定

アセットに機密度ラベルを設定するには、次の操作を行います。

> 注: 2024 年 2 月の Tableau Cloud、および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータ ラベル ダイアログを使用して機密度ラベルの追加と削除を行うようになりました。データ ラベル ダイアログの詳細については、「データ ラベル ダイアログ」を参照してください。

1. アセットを検索するか、アセットに移動します。移動するステップは、機密度ラベルを追加するアセットのタイプによって異なります。
   - データ ソースまたは仮想接続 - **[探索]** ページで、**[すべてのデータ ソース]** または **[すべての仮想接続]** を選択します。
   - 仮想接続テーブル - **[探索]** ページで、**[すべての仮想接続]** を選択し、認証する仮想接続テーブルを含む仮想接続を選択します。次に、仮想接続テーブルを選択します。
   - データベースまたはテーブル - **[探索]** ページで、データベースまたはテーブルに移動します。または **[外部アセット]** ページで、**[データベースとファイル]** または **[テーブルとオブジェクト]** を選択します。
   - 列 - **[探索]** ページで、テーブルに移動します。または **[外部アセット]** ページで **[テーブルとオブジェクト]** を選択して、テーブルに移動します。次に、リスト内で列を見つけます。
2. アセットの横にある [アクション メニュー(...)] を選択し、**[データ ラベル]**、**[機密度ラベル]** の順に選択します。

3. アセットにアタッチする機密度ラベルの横にあるチェックボックスを選択します。機密度ラベルの名前がわかっている場合は、ダイアログの上部で名前を検索し、その横にあるチェックボックスを選択できます。(Tableau Server 2023.3 以前では、各アセットにアタッチできる機密度ラベルは 1 つだけです。**[ラベルを表示]** 切り替えを使用してそのアセットの機密度ラベルをオンにし、ドロップダウン リストから目的の機密度ラベルを選択します)。
4. 必要に応じて、ユーザーに表示するメッセージを入力します。メッセージのテキストの書式を太字、下線、斜体に設定したり、リンクや画像を含めたりすることもできます。テキストの書式設定のヒントを表示するには、**[保存]** ボタンの上にある情報 (i) アイコンにカーソルを合わせます。
5. **[保存]** を選択します。



アセットからの機密度ラベルの削除

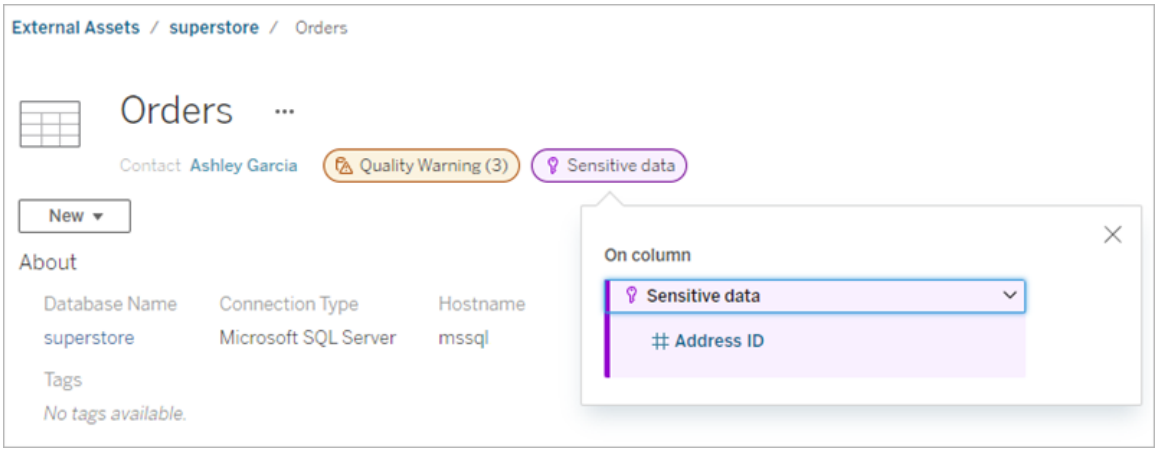
アセットから機密度ラベルを削除するには、次の操作を行います。

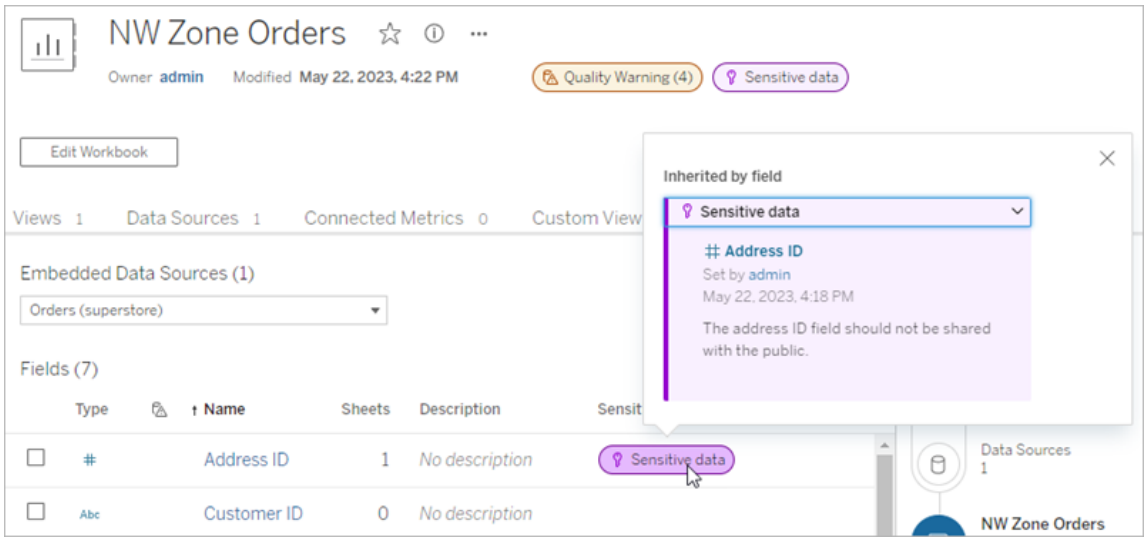
> 注: 2024 年 2 月の Tableau Cloud、および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータ ラベル ダイアログを使用して機密度ラベルの追加と削除を行うようになりました。データ ラベル ダイアログの詳細については、「データ ラベル ダイアログ」を参照してください

1. アセットの横にある [アクション メニュー(...)] を選択し、**[データ ラベル]**、**[機密度 ラベル]** の順に選択します。
2. アセットから削除する機密度ラベルの横にあるボックスのチェックボックスをオフにします。(Tableau Server 2023.3 以前では、**[ラベルを表示]** 切り替えを使用してラベルをオフにします。)
3. **[保存]** を選択します。

機密度ラベルが表示される場所

機密度ラベルは、Tableau Server を操作するときにアセット上に表示されます。データ品質警告と同様に、機密度ラベルは、ラベルが設定されているアセットの下流のアセットにも表示されます。たとえば、列の機密度ラベルは、テーブル ページの列の各行、テーブル ページの上部、およびデータベース ページに表示されます。





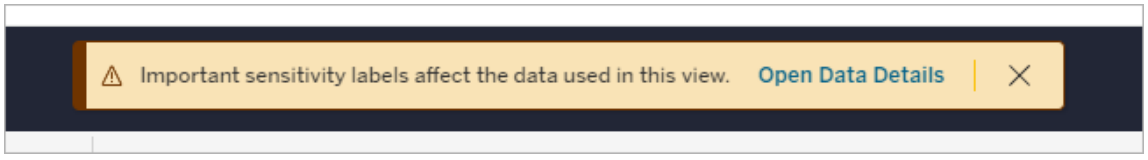
Web 作成では、次の場合に **[データ]** ペインのデータソースの横にアイコンが表示されます。

- ワークブックで使用しているデータソースに機密度ラベルがアタッチされている
- ワークブックで使用しているデータソースのアップストリームに警告がある

警告の詳細を表示するには、警告アイコンの上にカーソルを合わせます。または、Tableau Cloud の Web 作成では、データソースまたは列を選択してから、**[カタログの詳細]** を選択すると、そのラベルをすべて表示できます。

視認性

視認性の高い機密度ラベルは、より緊急性が高く、より多くの場所に表示されます。たとえば、テーブル上の視認性の高い機密度ラベルは、誰かがビューを作成するか、そのビューに依存するパブリッシュされたビューを開いたときに通知を生成します。



視認性の高い機密度ラベルのアイコンとバッジは紫色ですが、標準的な視認性のラベルの場合は灰色です。既定では、「機密データ」という名前の組み込み機密度ラベルは高視認性です。

複数の機密度ラベルがアセットに適用される場合 (アセットに複数の機密度ラベルがある、またはアップストリームのアセットに継承される)、バッジには番号が含まれ、色は最も視認性の高いラベルの色になります。たとえば、2 つの機密度ラベルがアセットに適用され、1 つが標準的な視認性のラベルで、もう1 つが視認性の高いラベルの場合、バッジは紫色になります。

メール サブスクリプションの機密度ラベル

管理者は、メール サブスクリプションの機密度ラベルをオンにして、ユーザーがビューをサブスクライブしたときに、そのビューに関連付けられた視認性の高い機密度ラベルを受信するメールに含めるようにすることができます。視認性の高い機密度ラベルが付いているメールには、次の内容が含まれます。

- **[データの詳細]** ペインが開いている状態の関連するビューまたはワークブックへのリンク
- データ ソース、テーブル、データベースなど、関連するアップストリーム アセットへのリンク

管理者は、Tableau Server または Tableau Cloud サイトの設定ページで **[ビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベル]** オプション (以前は **[サブスクリプションのデータ品質に関する警告]** オプションと呼ばれていました) を選択することにより、メール サブスクリプションの機密度ラベルをオンにできます。詳細については、サイト設定 リファレンスのビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベルを参照してください。

秘密度ラベルを設定できる人

機密度ラベルを設定するには、以下のいずれかである必要があります。

- サーバー管理者またはサイト管理者
- アセットに**上書き**できる権限がある。

機密度ラベルのカスタマイズ

組み込みの機密度ラベルは、*機密データ*のみです。2023 年 6 月の Tableau Cloud および Tableau Server 2023 以降、管理者は、**[データ ラベル]** ページのラベル マネージャーまたは REST

API を使用して、機密度ラベル値を作成したり、既存の機密度ラベルの名前や説明を変更したりできるようになりました。一般的には、次のような名前と説明を追加します。

- **公開**: 一般に公開されます。
- **社外秘**: 社員と契約社員に限定されます。このデータは社外に公開してはいけませんが、機密保持契約 (NDA) に基づいて顧客やパートナーと共有できます。
- **機密**: 認められた一部の社員と契約社員が利用できます。このデータは、法律、規制、または会社のマスター サービス契約 (MSA) による制約はありません。NDA に基づいて顧客、パートナーと共有できます。
- **制限付き**: 認められた一部の社員と契約社員のみが利用できます。このデータは、法律、規制、NDA、または会社の MSA による制約を受ける可能性があります。
- **極秘**: 認められたごく一部の社員と契約社員のみが利用できます。第三者にアクセスを許可する場合、より厳しい契約要件を課す可能性があります。このデータは、ほとんどの場合、法律、規制、NDA、または会社の MSA による制約を受けます。

詳細については、「データ ラベルの管理」を参照してください。

カスタム カテゴリのラベル

2025 年 10 月 14 日以降、「Data Cloud」は「Data 360」というブランド名に変わりました。この変更の途上では、アプリケーションやドキュメントに「Data Cloud」と表示されることがあります。名前は新しいですが、機能と内容は変更されていません。

Tableau は、データを分類するさまざまな方法をカバーするいくつかのデータ ラベル (認定ラベル、品質警告ラベル、機密度ラベル) を提供しています。ところが、ユースケースに合った、その他のラベルやカテゴリが必要になる場合もあります。2023 年 10 月の Tableau Cloud および Tableau Server 2023.3 以降、管理者が定義したカスタム カテゴリのラベルを使用してアセットを分類できるようになりました。たとえば、管理者は「部門」というカテゴリを作成して、販売、サービス、運用という部門のラベルを登録し、アセットに適用できるようにできます。

カスタム カテゴリのラベルには、Tableau Catalog が有効になっているデータ管理 ライセンスが必要であり、他のデータ ラベルと同じ種類のアセットに設定できます。ただし、カスタム カテゴリのラベルは、データ品質警告ラベルと秘密度ラベルと異なり、ダウンストリームのアセットには表示されません。

> 注: 管理者がカスタムのカテゴリとラベルを作成する場合は、データ ラベルの管理を参照してください。

カスタム カテゴリのラベルをアセットに設定する

> 注: 2024 年 2 月の Tableau Cloud、および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータ ラベル ダイアログを使用してカスタム カテゴリのラベルの追加と削除を行うようになりました。データ ラベル ダイアログの詳細については、「データ ラベル ダイアログ」を参照してください。

カスタム カテゴリのラベルをアセットに設定するには次の操作を行います。

Tableau Cloud および Tableau Server 2024.2 以降の場合

1. アセットを検索するか、アセットに移動します。移動するステップは、ラベルを追加するアセットのタイプによって異なります。
   - データ ソースまたは仮想接続 - **[探索]** ページで、**[すべてのデータ ソース]** または **[すべての仮想接続]** を選択します。
   - 仮想接続テーブル - **[探索]** ページで、**[すべての仮想接続]** を選択し、認証する仮想接続テーブルを含む仮想接続を選択します。次に、仮想接続テーブルを選択します。
   - データベースまたはテーブル - **[探索]** ページで、データベースまたはテーブルに移動します。または **[外部アセット]** ページで、**[データベースとファイル]** または **[テーブルとオブジェクト]** を選択します。
   - 列 - **[探索]** ページで、テーブルに移動します。または **[外部アセット]** ページで **[テーブルとオブジェクト]** を選択して、テーブルに移動します。次に、リスト内で列を見つけます。
2. アセットの横にあるアクション メニュー (...) を選択し、**[データ ラベル]**、**[すべてのデータ ラベル]** の順に選択します。
3. ダイアログの左側にある、カスタム ラベル カテゴリに対応する垂直タブを選択します。ラベルの名前がわかっている場合は、ダイアログの上部でラベルを検索できます。
4. アセットにアタッチするラベルの横にあるチェックボックスを選択します。
5. 必要に応じて、ユーザーに表示するメッセージを入力します。メッセージのテキストの書式を太字、下線、斜体に設定したり、リンクや画像を含めたりすることもできます。テキストの書

式設定のヒントを表示するには、**[保存]** ボタンの上にある情報 (i) アイコンにカーソルを合わせます。

6. 追加するラベルごとにステップ 3 ~ 5 を繰り返します。
7. **[保存]** を選択します。

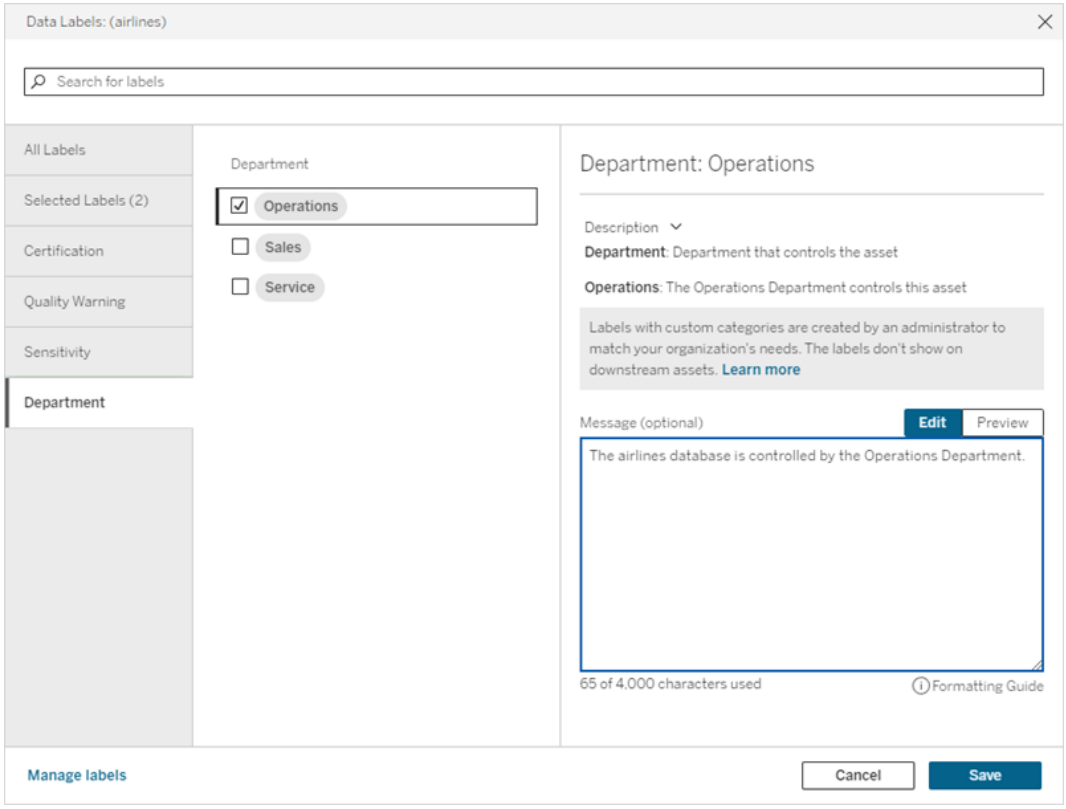
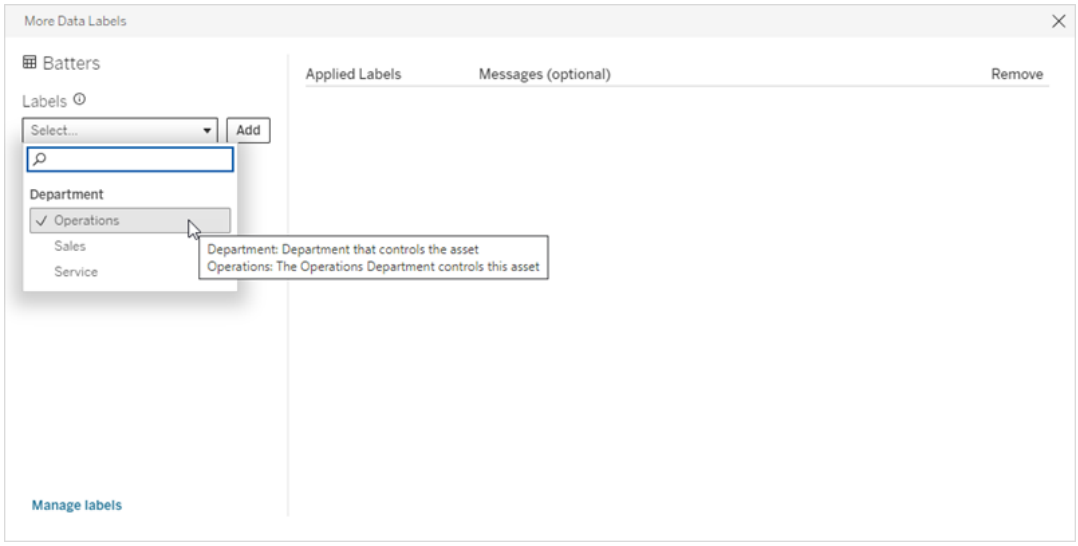


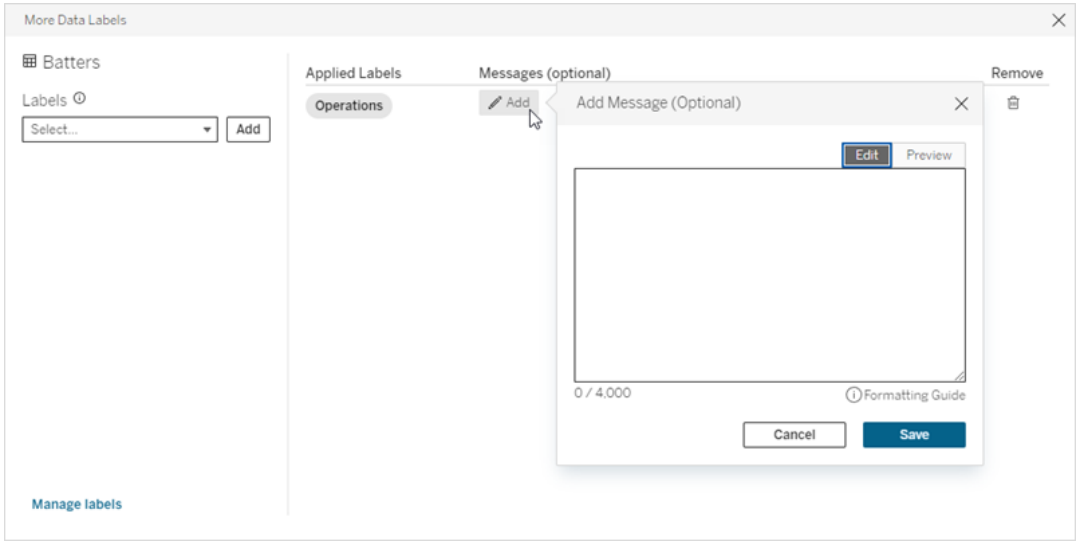
Tableau Server 2023.3 の場合

1. アセットの横にあるアクション メニュー (...) を選択し、**[データ ラベル]**、**[データ ラベルをさらに表示]** の順に選択します。

2. **[ラベル]** ドロップダウンを選択します。ラベル名はラベル カテゴリごとに並べられており、スクロールすることができます。



3. アタッチするラベルを選択し、**[ラベル]** ドロップダウンの横にある **[追加]** を選択します。ダイアログの右側の **[適用されたラベル]** リストにラベルが表示されます。
4. この特定のアセットにオプションのラベル メッセージを追加するには、**[メッセージ (オプション)]** 列の鉛筆を選択し、**[保存]** を選択します。



5. 追加するラベルごとに手順 2 から 5 を繰り返します。
6. ラベルの追加が完了したら、ダイアログを閉じます (ダイアログ ボックスの右上隅にある [X] を選択するか、ダイアログ ボックスの外側を選択してダイアログ ボックスを閉じます)。

カスタム カテゴリのラベルをアセットから削除する

> 注: 2024 年 2 月の Tableau Cloud、および Tableau Server 2024.2 以降、ラベルのタイプごとに個別のダイアログを使用するのではなく、統合されたデータ ラベル ダイアログを使用してカスタム カテゴリのラベルの追加と削除を行うようになりました。データ ラベル ダイアログの詳細については、「データ ラベル ダイアログ」を参照してください。

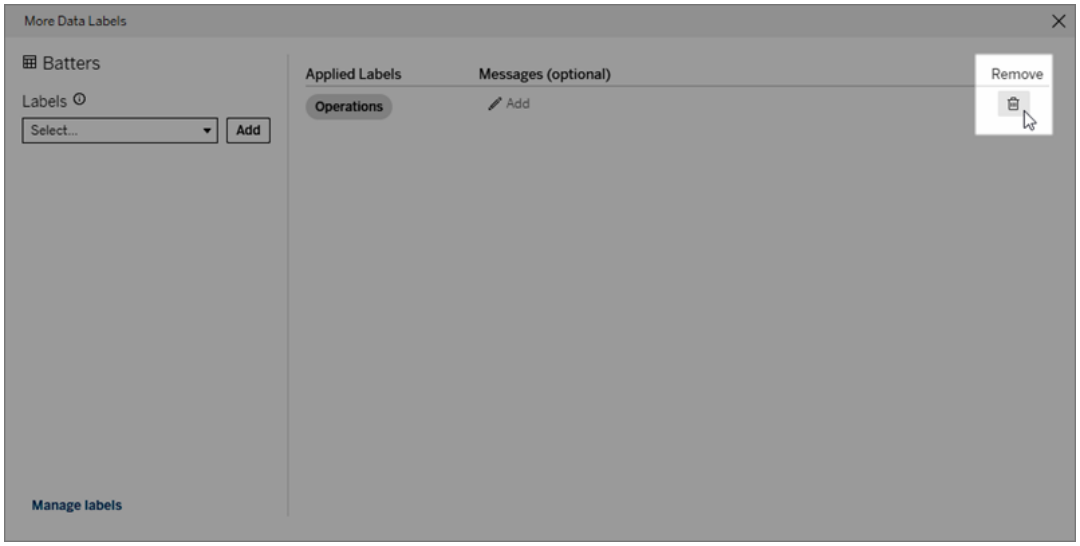
カスタム カテゴリのラベルをアセットから削除するには次の操作を行います。

Tableau Cloud および Tableau Server 2024.2 以降の場合

1. アセットを検索するか、アセットに移動します。移動するステップは、ラベルを追加するアセットのタイプによって異なります。
   - データ ソースまたは仮想接続 - **[探索]** ページで、**[すべてのデータ ソース]** または **[すべての仮想接続]** を選択します。
   - 仮想接続テーブル - **[探索]** ページで、**[すべての仮想接続]** を選択し、認証する仮想接続テーブルを含む仮想接続を選択します。次に、仮想接続テーブルを選択します。
   - データベースまたはテーブル - **[探索]** ページで、データベースまたはテーブルに移動します。または **[外部アセット]** ページで、**[データベースとファイル]** または **[テーブルとオブジェクト]** を選択します。
   - 列 - **[探索]** ページで、テーブルに移動します。または **[外部アセット]** ページで **[テーブルとオブジェクト]** を選択して、テーブルに移動します。次に、リスト内で列を見つけます。
2. アセットの横にあるアクション メニュー (...) を選択し、**[データ ラベル]**、**[すべてのデータ ラベル]** の順に選択します。
3. ダイアログの左側にある、カスタム ラベル カテゴリに対応する垂直タブを選択します。必要に応じて、**[選択したラベル]** 垂直タブを使用して、アセットにアタッチされているすべてのラベルを表示します。または、ラベルの名前がわかっている場合は、ダイアログの上部でラベルを検索できます。
4. アセットから削除するラベルの横にあるチェックボックスの選択を解除します。
5. 削除するラベルごとにステップ 3 と 4 を繰り返します。
6. **[保存]** を選択します。

Tableau Server 2023.3 の場合

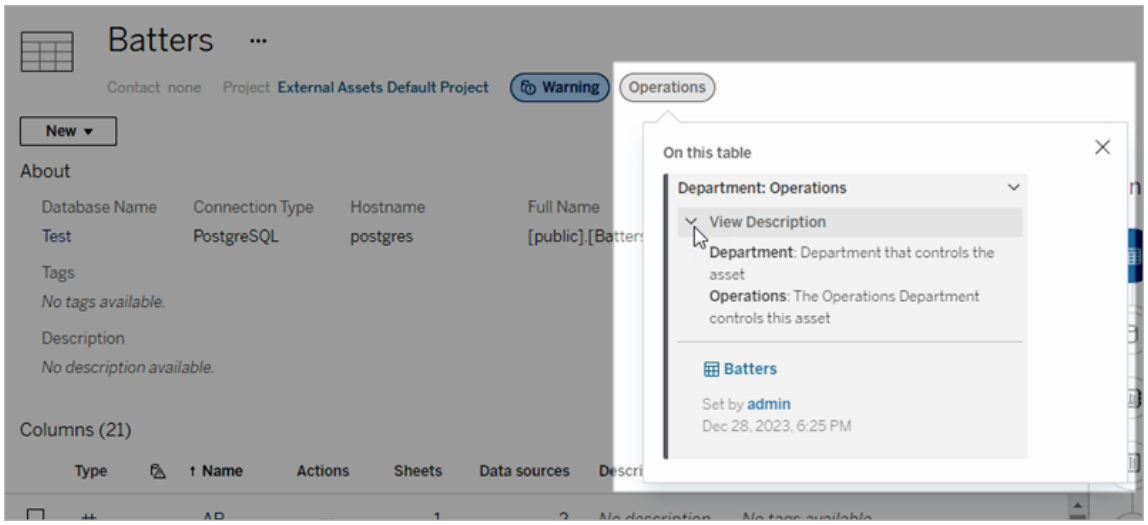
1. アセットの横にあるアクション メニュー (...) を選択し、**[データ ラベル]**、**[データ ラベルをさらに表示]** の順に選択します。
2. ダイアログの **[適用されたラベル]** セクションで、削除するラベルの行にあるゴミ箱アイコンを選択します。



3. 削除するラベルごとに手順 2 を繰り返します。
4. ラベルの削除が完了したら、ダイアログを閉じます (ダイアログ ボックスの右上隅にある [X] を選択するか、ダイアログ ボックスの外側を選択してダイアログ ボックスを閉じます)。

カスタム カテゴリのラベルが表示される場所

カスタム ラベルは、Tableau Cloud と Tableau Server を操作するときにアセット上に表示されます。



Web 作成では、データ ソースまたは列を選択してから、**[カタログの詳細]** を選択すると、そのラベルをすべて表示できます。

品質警告ラベルや秘密度ラベルとは異なり、カスタム カテゴリのラベルは、ラベルがアタッチされているアセットのダウンストリームには表示されません。たとえば、組織に「部門」という名前のカスタム ラベル カテゴリがあり、「販売」という名前のカスタム ラベルが属するとします。「販売」ラベルを「注文」というテーブルに設定すると、ラベルは「注文」テーブルにのみ表示され、そのダウンストリームのワークブックには表示されません。

カスタム カテゴリ ラベルを追加できる人

カスタム カテゴリのラベルをアセットに追加するには (またはアセットからラベルを削除するには)、次のいずれかである必要があります。

- サーバー管理者またはサイト管理者
- アセットに上書きできる権限がある

カスタム カテゴリのラベルをカスタマイズする

[データ ラベルをさらに表示] ダイアログに表示されるカスタム カテゴリとラベルを管理者が作成または編集する方法については、「データ ラベルの管理」を参照してください。

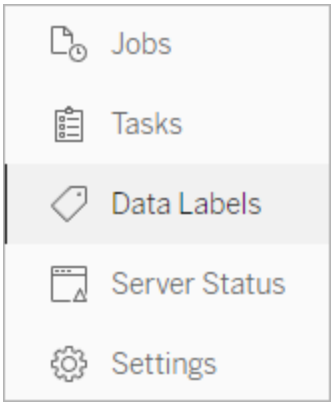
データ ラベルの管理

2023 年 10 月の Tableau Cloud および Tableau Server 2023.3 以降、データ管理 ライセンスがあり、管理者である場合は、ラベル マネージャーを使用してラベル名とラベル カテゴリを作成および編集できます。これらのカスタマイズは、ユーザーがラベルを操作するときに Tableau 全体でラベルが表示される方法に影響します。

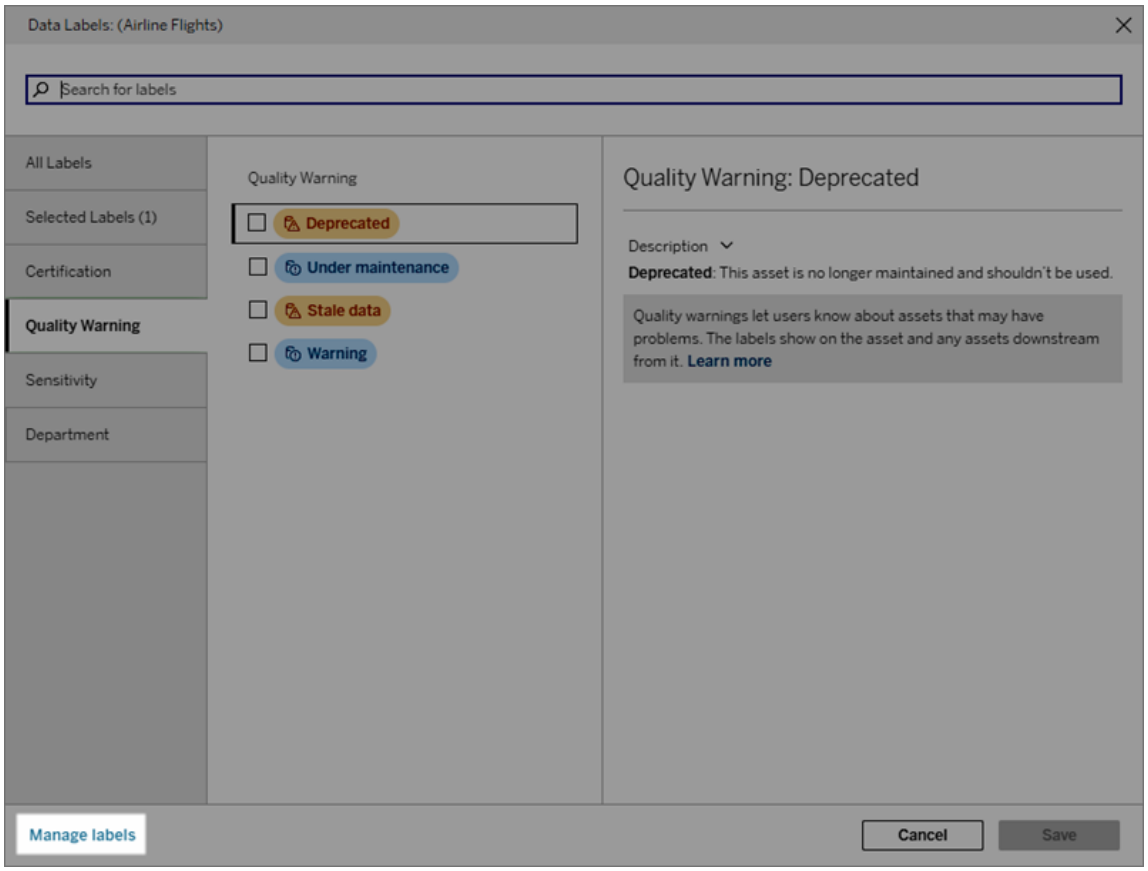
> 注: REST API の labelValues メソッドと labelCategories メソッドを使用して、ラベルとラベル カテゴリを作成および編集することもできます (Tableau Cloud 管理者は、2023 年 6 月の Tableau Cloud 以降、labelValues メソッドを使用してラベルの名前と説明を作成および変更できるようになりました)。詳細については、「REST API リファレンス」のメタデータ メソッドを参照してください。

ラベル マネージャー

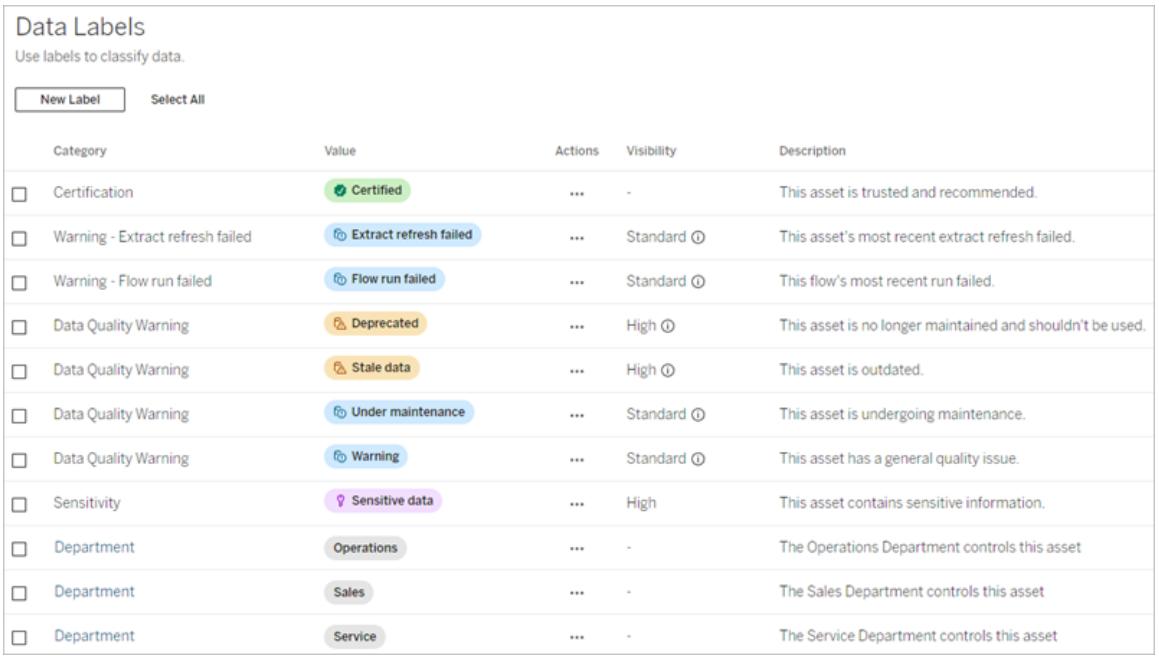
ラベル マネージャーを使用するには、管理者としてログインし、左側のナビゲーション ペインから **[データ ラベル]** を選択します。



あるいは、管理者としてログインし、**[データ ラベル]** ダイアログを開いてアセットにラベルをアタッチすると、左下隅に **[ラベルの管理]** リンクが表示されます。**[ラベルの管理]** リンクをクリックすると、**[データ ラベル]** ページに移動します。(Tableau Server 2023.3 以前では、個々の認証、データ品質に関する警告、機密度ラベル、カスタム ラベル ダイアログのラベル選択のドロップダウンに **[ラベルの管理]** リンクが表示されます。)



ラベル マネージャーのページには、ラベル カテゴリごとに並べ替えられた各ラベルの行が表示されます。各行には、ラベルのカテゴリ、名前 (ここでは値と呼ばれます)、そのラベルに対してアクションを実行するための **[アクション]** メニュー (...)、視認性、および説明が含まれます。



ラベル マネージャーを使用して次のことを行います。

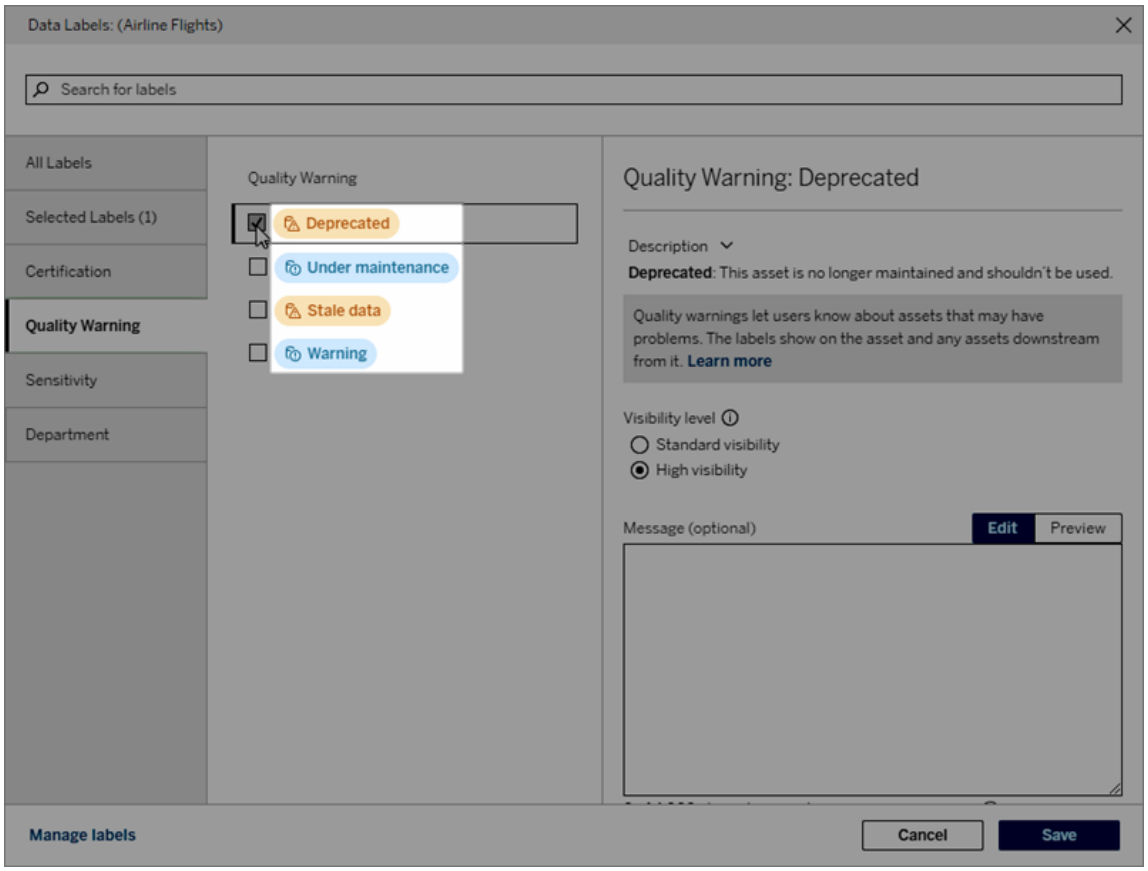
- **組み込みラベルを編集する**
  - 例: 組み込みの「メンテナンス中」という品質警告の名前を「メンテナンス モード」に変更します。
  - 例: 抽出更新監視ラベルの視認性を標準の視認性から高い視認性に変更して、ビューに表示されるようにします。
- **既存の組み込みカテゴリを使用する新しいラベルを作成する**
  - 例: 「機密」という新しい機密度ラベルを追加します。
- **組み込みラベルをデフォルトの名前、説明、視認性に戻す**
  - 例: 品質警告名を「古いデータ」から「期限切れ」に変更しましたが、デフォルト名に戻します。
- ***カスタム カテゴリを作成する***
  - 例: さまざまな事業部門のラベルを追加するために、「部門」という新しいラベル カテゴリを作成します。
- ***カスタム カテゴリに新しいラベルを作成する***
  - 例: 新しく作成した「部門」カテゴリに、「販売」、「サービス」、「運用」というラベルを新しく作成します。

データ ラベルのプロパティ

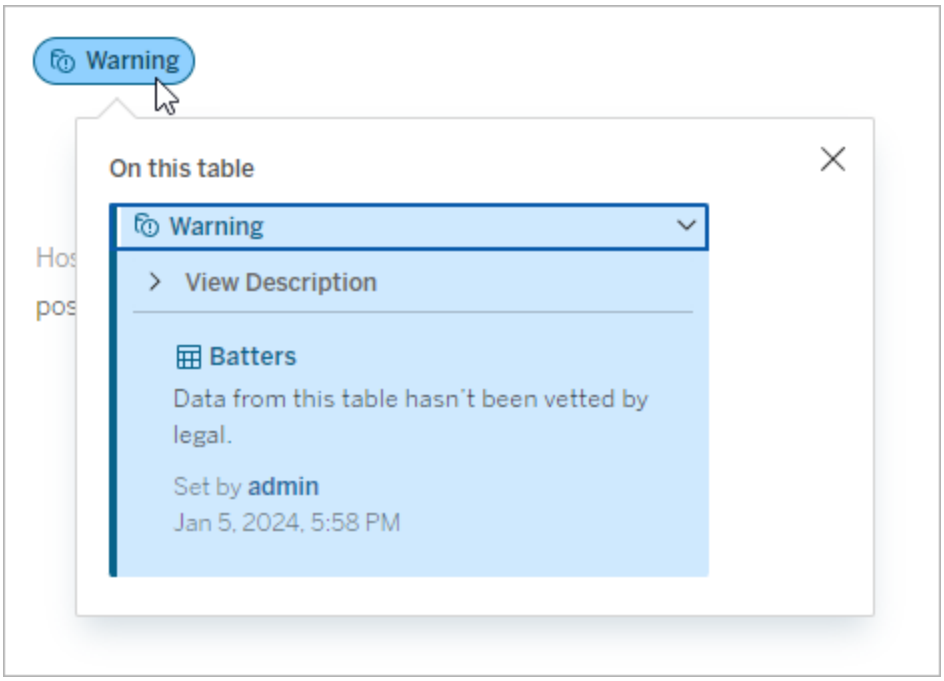
ラベルには名前、カテゴリ、および説明があります。品質警告または機密度カテゴリのラベルには視認性レベルもあります。

名前

ラベル名は、さまざまな場所に表示されるラベルの通称です。たとえば、ここでは、**[データ ラベル]** ダイアログの **[品質に関する警告]** タブでラベル名 [Deprecated (非推奨)] が選択されています。



ここでは、ラベル名「警告」が「バッター」テーブル ページの上部に表示され、[ラベルの詳細] にも表示されています。

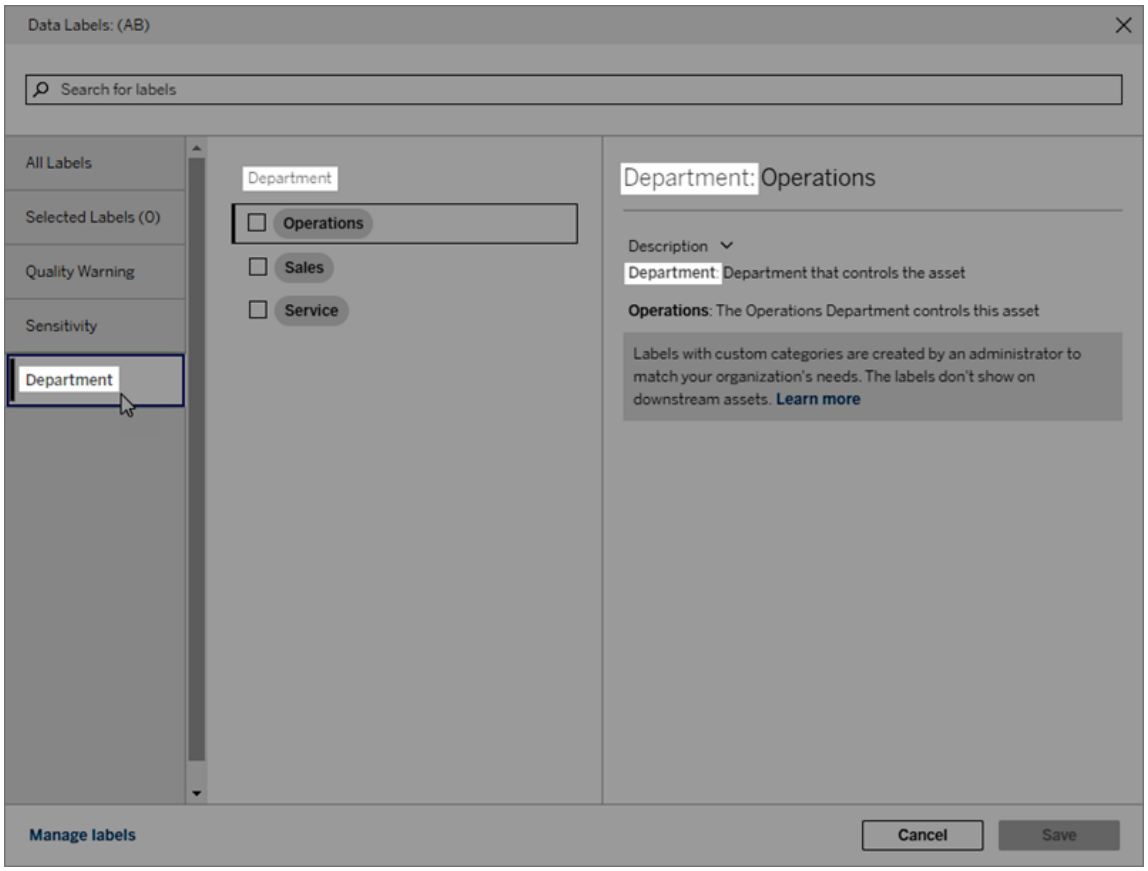


カテゴリ

ラベル *カテゴリ*は、ラベルがどこにどのように表示されるか、ラベルを設定したアセットのダウンストリームのアセットにラベルが表示されるかどうか、どの部分がカスタマイズ可能であるかなどに影響します。たとえば、品質警告 ラベルと機密度 ラベルはダウンストリームのアセットに表示されますが、他のカテゴリのラベルは表示されません。別の例: 認定 ラベルの説明は変更できますが、名前は変更できません。

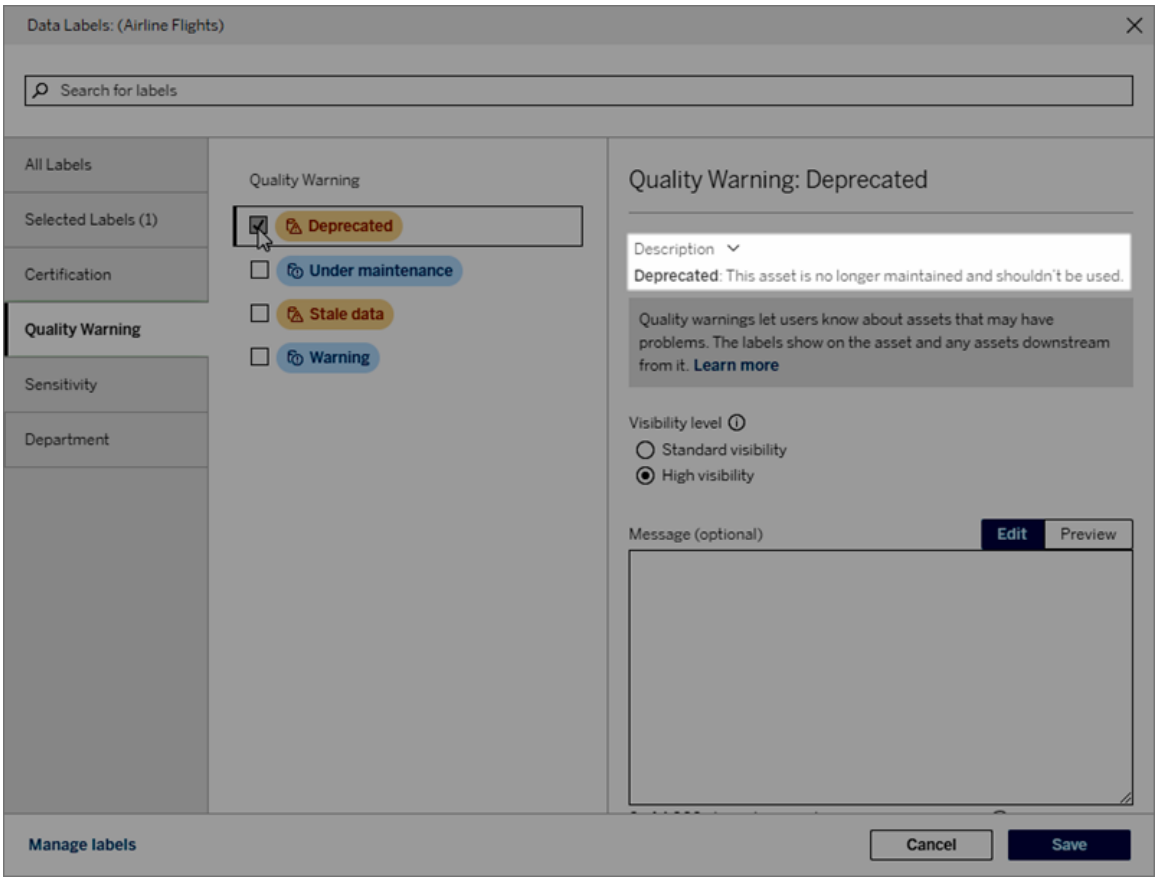
組み込み *カテゴリ*は、**認定**、**品質警告**、および**機密度**です。

*カスタム カテゴリ*の場合は、**[データ ラベル]** ダイアログの垂直 *カテゴリ* タブなどに *カテゴリ* 名が表示されます。たとえば、ここでは *カテゴリ* 名「部門」が垂直 *カテゴリ* タブやラベル名リストの上部などに表示されます。



説明

ラベルの説明は、**[データ ラベル]** ダイアログを含むさまざまな場所に表示されるため、ユーザーはラベルの用途を簡単に理解することができます。たとえば、この品質に関する警告のラベルの説明には、「このアセットは保守されなくなったので、使用しないでください」と表示されています。



視認性

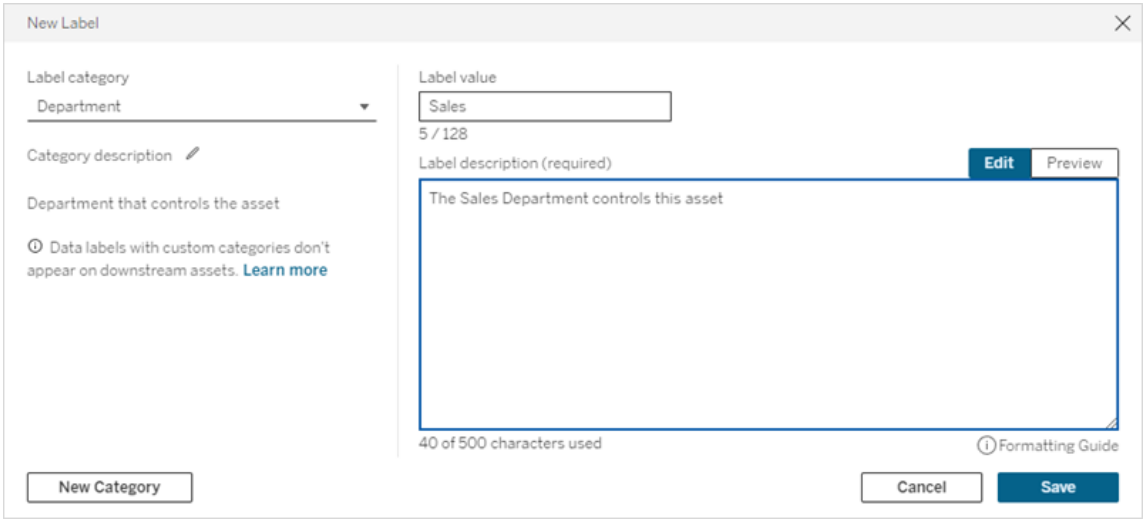
ラベルの外観は視認性に応じて変わります。視認性の高いラベルはより多くの場所に表示されるため、緊急性が高いラベルとして認識される可能性があります。視認性のレベルは、品質警告ラベルまたは機密度ラベルにのみ設定できます。さらに、ラベルに品質警告のカテゴリがある場合、権限を持つユーザーは、品質警告ラベルを設定する各アセットのデフォルトの視認性を上書きできます。詳細については、データ品質に関する警告と機密度ラベルの設定に関するトピックを参照してください。

データラベルを作成する

ラベルを作成するには次の操作を行います。

1. **[データ ラベル]** ページで、**[新しいラベル]** を選択します。
2. **[ラベル]** カテゴリのドロップダウンからカテゴリを選択します。
3. **[ラベル値]** フィールドにラベル名を入力します。

4. **[ラベルの説明]** フィールドにラベルの説明を入力します。テキストの書式を太字、下線、斜体などに設定したり、リンクや画像を含めたりすることもできます。テキストの書式設定のヒントを表示するには、**[保存]** ボタンの上にある情報 (i) アイコンにカーソルを合わせます。
5. ラベルに品質警告または機密度のカテゴリがある場合は、視認性レベルを設定します。詳細については、「視認性」を参照してください。
6. **[保存]** を選択します。



ラベル作成の制限事項

- 認定カテゴリではラベルを作成できません。認定カテゴリでは、単一の組み込みラベルのみが許可されます。
- 新しい監視警告を作成することはできません。ただし、「ラベルの編集」セクションで説明されているように、抽出更新の失敗とフロー実行の失敗の警告は、限定的な方法で編集することができます。
- Tableau Cloud とTableau Server 2024.2 以降では、ラベル名の最大長は 128 文字です。Tableau Server 2023.3 以前では、ラベル名の最大長は 24 文字です。
- ラベルの説明の最大長は 500 文字です。

データラベルを編集する

既存のラベルを編集するには次の操作を行います。

1. **[データ ラベル]** ページで、ラベルの行の **[アクション]** (...) メニューを選択します。または、左側のチェックボックスで行を選択し、ラベル リストの上部にある **[アクション]** ドロップダウンをクリックします。

2. **[編集]** を選択します。
3. (オプション) **[ラベル値]** フィールドをでラベル名を変更します。
4. (オプション) **[ラベルの説明]** フィールドをでラベルの説明を変更します。
5. (オプション) ラベルに品質警告または機密度のカテゴリがある場合は、視認性レベルを設定します。詳細については、「視認性」を参照してください。
6. **[保存]** を選択します。

ラベル編集の制限事項

- 既存のラベルのカテゴリを変更することはできません。
- Tableau Cloud とTableau Server 2024.2 以降では、ラベル名の最大長は 128 文字です。Tableau Server 2023.3 以前では、ラベル名の最大長は 24 文字です。
- ラベルの説明の最大長は 500 文字です。

ラベルのカテゴリによってラベル編集のレベルを変えることができます。次の表に、指定されたカテゴリのラベルの編集可能なプロパティを示します。

ラベルのカテゴリ	ラベルのカテゴリを編集できる	ラベルの名前を編集できる	ラベルの説明を編集できる	ラベルの視認性を編集できる
認定	不可	不可	可	N/A
品質に関する警告	不可	可[1]	可	可[2]
機密度	不可	可	可	可
カスタム	不可	可	可	N/A

[1] *抽出更新またはフロー実行の監視警告のラベル名 (ラベル値) は編集できません。*
[2] *品質警告に対して設定した視認性レベルがデフォルトの視認性です。権限のあるユーザーは、アセットに品質警告を設定するときに、デフォルトの視認性を上書きできます。詳細については、「視認性」を参照してください。*

データラベルを削除する

既存のラベルを削除するには次の操作を行います。

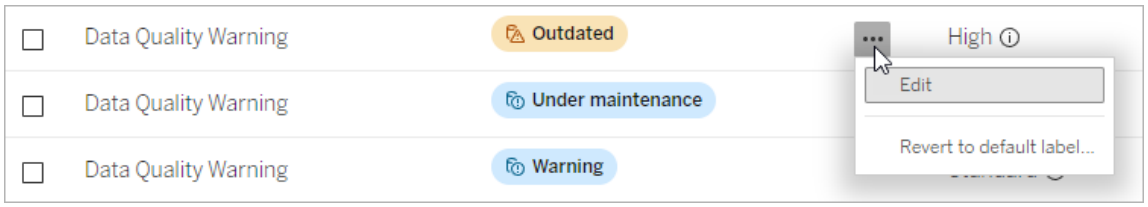
1. **[データ ラベル]** ページで、ラベルの行の **[アクション]** (...) メニューを選択します。または、左側のチェックボックスで行を選択し、ラベル リストの上部にある **[アクション]** ドロップダウンをクリックします。
2. **[削除]** を選択します。

ラベル削除の制限事項

- 組み込みラベルは削除できません。組み込みラベルは、Tableau Server のデフォルトのラベルです。

組み込みデータ ラベルをデフォルトに戻す

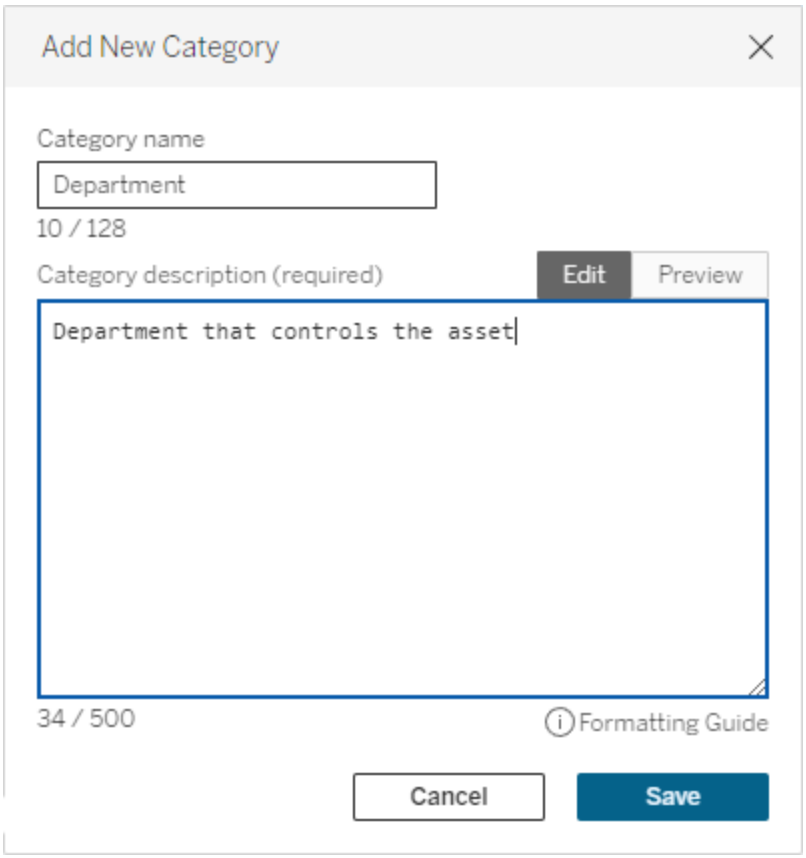
管理者が以前に組み込みラベルを編集していた場合、そのラベルの **[アクション]** メニューには **[デフォルト ラベルに戻す]** が表示されます。ラベルをデフォルトに戻すと、ラベルの名前 (値)、説明、および視認性が組み込みのデフォルトに戻ります。



データ ラベルのカテゴリを作成する

ラベル カテゴリを作成するには次の操作を行います。

1. **[データ ラベル]** ページで、**[新しいラベル]** を選択します。
2. **[新しいラベル]** ダイアログで、**[新しいカテゴリ]** を選択します。
3. **[カテゴリ名]** フィールドにカテゴリ名を入力します。
4. **[カテゴリの説明]** フィールドにカテゴリの説明を入力します。テキストの書式を太字、下線、斜体などに設定したり、リンクや画像を含めたりすることもできます。テキストの書式設定のヒントを表示するには、**[保存]** ボタンの上にある情報 (i) アイコンにカーソルを合わせます。
5. **[保存]** を選択します。



ラベル カテゴリ作成の制限事項

- Tableau Cloud とTableau Server 2024.2 以降では、カテゴリ名の最大長は 128 文字です。Tableau Server 2023.3 以前では、カテゴリ名の最大長は 24 文字です。
- カテゴリの説明の最大長は 500 文字です。

データ ラベルのカテゴリを編集する

ラベル カテゴリを編集するには次の操作を行います。

1. **[データ ラベル]** ページからラベルのカテゴリを選択し、鉛筆アイコンを選択します。または、**[新しいラベル]** ダイアログまたは **[ラベルの編集]** ダイアログの **[ラベル カテゴリ]** ドロップダウンでカテゴリを選択し、**[カテゴリの説明]** の横にある鉛筆アイコンを選択します。
2. (オプション) **[カテゴリ名]** フィールドを使用して、カテゴリ名を変更します。
3. (オプション) **[カテゴリの説明]** フィールドを使用して、カテゴリの説明を変更します。
4. **[保存]** を選択します。

ラベル カテゴリ編集の制限事項

- 組み込みカテゴリは編集できません。
- Tableau Cloud とTableau Server 2024.2 以降では、カテゴリ名の最大長は 128 文字です。Tableau Server 2023.3 以前では、カテゴリ名の最大長は 24 文字です。
- カテゴリの説明の最大長は 500 文字です。

データ ラベルのカテゴリを削除する

現在、通常の Tableau Server インターフェイスでラベル カテゴリを削除することはできません。REST API を使用したカテゴリの削除の詳細については、「REST API リファレンス」の「ラベル カテゴリの削除 メソッド」を参照してください。

カスタマイズのシナリオ

シナリオ: 組み込みデータ ラベルをカスタマイズする

「警告」というデータ品質警告をより具体的にできると判断したとします。管理者は、ラベル名をデフォルトの「警告」から、組織にとってより実用的と思われる名前「未許可」に変更します。ユーザーがラベルを選択しているときに、ラベル名「未許可」が [ラベル] ダイアログに表示されるようになりました。

あるいは、ラベルの説明を変更して、ユーザーが [ラベル] ダイアログの警告をよりよく理解できるようにすることもできます。例: 「このアセットは、マーケティング部門が求める品質基準を満たしていません。」

シナリオ: カスタムのデータ ラベルを作成する

アセットの機密度の分類を、ユーザーがより細かく管理できるようにしたいとします。「パブリック」と「個人情報 (PII)」という名前の 2 つの新しい機密度 ラベルを作成します。カスタムのラベル名「パブリック」と「個人情報 (PII)」が、組み込みの機密度 ラベルと並んで、[ラベル] ダイアログのドロップダウンと説明に表示されるようになりました。

シナリオ: データ ラベルの新しいカテゴリと、関係づけるデータ ラベルを作成する

アセットを担当する業務部門を特定する方法が必要だとします。「部門」という名前の新しいラベル カテゴリを作成します。次に、その「部門」カテゴリに関連する「販売」、「サービス」、「運用」の 3

つのラベルを作成します。「部門」カテゴリに関係づけられた3つのラベルが、ユーザーがアセットに設定できる**[その他のデータ ラベル]**ダイアログに表示されるようになりました。

Tableau Server でダッシュボードとViz の拡張機能を管理する

Tableau 拡張機能は、Tableau ワークブックで提供される分析機能を拡張できるWeb アプリケーションです。拡張機能は、Tableau Extensions API を使用して、Tableau ワークブックの残りの部分とシームレスに統合し、連携します。

ダッシュボードの拡張機能は、カスタム ダッシュボード ゾーンで実行され、ダッシュボード残りの部分と連携します。ダッシュボードの拡張機能を使用すると、ユーザーは他のアプリケーションをTableau で直接操作できます。Viz 拡張機能はワークシート内で実行され、ユーザーはカスタム Viz タイプを使用できます。Tableau ユーザーは、ダッシュボードの拡張機能にダッシュボード オブジェクトからアクセスし、Viz 拡張機能にはワークシートのマーク カードを介してアクセスします。

注: サーバー管理者は、サーバーでダッシュボードとViz の拡張機能を有効にしたり、特定の拡張機能の実行をブロックしたりできます。管理者は、拡張機能を許可リストに追加し、拡張機能がアクセス可能なデータのタイプを制御できます。サーバー管理者は、拡張機能を追加または表示する際に、サイト上のユーザーにプロンプトが表示されるかどうかを構成することもできます。また、サーバー管理者には、特定の拡張機能の使用をすべてのサイトでブロックできる権限があります。拡張機能のセキュリティと推奨される導入オプションについては、「拡張機能のセキュリティ- 展開に関するベストプラクティス」を参照してください

Tableau でダッシュボードの拡張機能を使用する方法の詳細については、「ダッシュボードの拡張機能の使用」を参照してください。

Viz の拡張機能を使用する方法の詳細については、「ワークシートに Viz の拡張機能を追加する」を参照してください。

Tableau Cloud については、「Tableau Cloud でダッシュボードの拡張機能を管理する」を参照してください。

Tableau Server で拡張機能を実行する前に

拡張機能はオープンプラットフォームですが、一般的に考慮すべき標準形式が 2 つあります。

- Tableau の信頼できる拡張機能は、厳密な審査を受け、Tableau が管理するホストで展開される拡張機能です。これらの拡張機能には、Tableau が開発および管理するソリューション、Tableau Exchange パートナーが構築し、Tableau の審査およびホストサービスを通じて展開されるソリューション、および従来のサンドボックス拡張機能ホスト内のソリューションが含まれます。

- ネットワーク対応拡張機能: サードパーティーのホストでホストされる拡張機能です。ネットワーク対応拡張機能を使用すると、開発者は Tableau を介さずに拡張機能の配信を管理できます。これらには、より優れた本番環境サポートを提供するためにパートナーが直接管理する拡張機能、ローカル ネットワーク内でホストおよび展開される拡張機能、および Tableau 外部のサードパーティー サービスとの通信を必要とする拡張機能が含まれます。ネットワーク対応拡張機能には、Web へのフル アクセス権があります。

> **注:** Tableau は Einstein Discovery ダッシュボードの拡張機能を介した Einstein Discovery との統合をサポートします。これは、Salesforce.com のデータにアクセスできる特別な拡張機能であり、既定で許可されています。これは、ネットワーク対応拡張機能またはサンドボックス拡張機能とは見なされません。Einstein Discovery の統合の詳細については、Tableau Server のリリース ノートを参照してください。

Tableau の信頼できる拡張機能

2025.3 以降、Tableau の信頼できる拡張機能は、使用のために昇格された Tableau Exchange 拡張機能のサブセットを表します。信頼できる拡張機能は Tableau ホストでホストされ、ユーザー クライアントと Tableau との間の通信を制限するために必要となります。これらの拡張機能には、展開前にネットワーク通信のテストと脆弱性のスキャンが実施されています。この拡張機能の標準形式は、拡張機能へのより管理されたアクセスを提供するために使用されます。この標準形式には以下が含まれます。

- サンドボックス拡張機能は Tableau によってホストされ、コンテンツ セキュリティ ポリシー (CSP) などの W3C 標準を使用します。また、この拡張機能では、Tableauサンドボックス

ホスト以外でネットワーク呼び出しを行うことはできません。サンドボックス拡張機能ではデータをクエリできますが、サンドボックス環境ではそのデータと外部との通信を明示的にブロックします。サンドボックス拡張機能は、Tableau 2019.4 以降でサポートされています。既定では、サイトの拡張機能が有効になっていればサンドボックス拡張機能を実行できます。

- Tableau が作成した拡張機能は、Tableau 開発チームによって開発され、積極的にメンテナンスされているソリューションです。Tableau が作成した拡張機能は、ユーザーデータを外部にクエリするのではなく、ユーザークライアントに通信を向けるように明示的に設計されています。これにより、データ流出やデータ保持の設計上のリスクが制限されます。また、Tableau が作成した拡張機能は、Salesforce ソフトウェア開発ライフサイクル (SDLC) 標準も満たしています。既定では、サイトの拡張機能が有効になっていれば、Tableau が作成した拡張機能を実行できます。

- 信頼できるパートナーが作成した拡張機能は、Tableau Exchange に貢献している、認定された Tableau および Salesforce パートナーによって開発され、積極的にメンテナンスされているソリューションです。Tableau 開発者は、これらのサードパーティーソリューションを Tableau の信頼できる拡張機能として展開する前に、手動テストを通じて検証します。Tableau 開発者は、専用の Tableau ホストに展開する前に、拡張機能の機能テスト、ネットワーク通信テスト、自動コードスキャンを実施します。信頼できるパートナーが作成した拡張機能は、Tableau 管理者が有効にして使用可能にします。

ネットワーク対応拡張機能

ネットワーク対応拡張機能は Web アプリケーションなので、Web サーバーとしてセットアップされた任意の PC で実行することができます。これにはローカル PC、ドメイン内の PC、サードパーティーの Web サービスが含まれます。ネットワーク対応拡張機能はサードパーティーのサーバー上でホストされ、ワークブックのデータにアクセスできる可能性があるため、信頼できる拡張機能のみを許可する必要があります。ネットワーク対応拡張機能のセキュリティテストを参照してください。

既定の拡張機能設定

管理者は、Tableau Server の拡張機能の設定を使用して、実行が許可される拡張機能を制御したり、制限したりすることができます。

- 既定では、明示的に無効にしない限り、拡張機能の実行が有効になっています。この設定は、サーバー管理者がすべてのサイトに対して一括管理するか、サイト管理者が特定のサイトで無効にすることができます。

- Tableau の信頼できる拡張機能の使用を許可することをお勧めします。既定では、サンドボックス拡張機能とTableau が作成した拡張機能は、サイトで有効になっていれば実行できます。信頼できるパートナーが作成した拡張機能は、管理者が有効にすることができます。

- 既定では、ネットワーク対応拡張機能は明示的に許可リストに追加されていない限り許可されません。HTTPS プロトコルを使用する拡張機能のみが許可され、この場合、データの送受信用の暗号化チャネルが使用されます (唯一の例外は `http://localhost` です)。一部の拡張機能には、フル データ パーミッション (参照元データへのアクセス) が必要です。明示的にセーフリストに拡張機能を追加し、拡張機能にフル データへのアクセス権を付与しない限り、これらの拡張機能を Tableau Server で実行することはできません。

制御に関する拡張機能の使用

サーバー管理者は、サーバー上のすべてのサイトで拡張機能を許可するように全体設定を制御できます。また、サーバー管理者は、サンドボックス拡張機能などの拡張機能が実行されないように全体の拒否リストに入れることもできます (特定の拡張機能のブロックを参照してください)。既定では、すべてのサンドボックス拡張機能がサーバーで有効になっていますが、サイト管理者によって既定の拡張機能を上書きし、サイトのサンドボックス拡張機能を禁止することもできます。

サーバーでの拡張機能を有効にするグローバル設定の変更

1. サーバーでこの設定を変更するには、**[すべてのサイトを管理] > [設定] > [拡張]** に移動します。サーバーに 1 つのサイトしかない場合は、全体のコントロールがサイトの設定ページに表示されます。

2. [Dashboard and Viz Extensions (ダッシュボードとViz の拡張機能)] で、**[このサーバーで拡張機能を実行することを許可する]** のチェック ボックスをオンまたはオフにします。このオプションをオンにしないと拡張機能の実行は許可されません。このグローバル設定は、各サイトでの **[このサイトで拡張機能を実行することを許可する]** 設定よりも優先されます。

サイトの既定の設定の変更

サーバー管理者は、サイトに対して拡張機能を有効にするか、サイトで Tableau の信頼できる拡張機能を許可するかを制御できます。つまり、サーバーで拡張機能が有効になっている場合、既定のサイト設定によりサイトでのサンドボックス拡張機能の実行が許可されます。ただし、拡張機能がサーバー上で明示的にブロックされている場合を除きます。既定のサイト設定では、ネットワーク対応拡張機能がサイトの許可リストに登録されている場合は、実行が許可されます。拡張機能を明示的にセーフリストに追加すると、その拡張機能がサイトで使用可能になります。

1. サイトでこれらの設定を変更するには、**[設定] > [拡張]** に移動します。
2. [Dashboard and Viz Extensions (ダッシュボードとViz の拡張機能)] で、次のオプションを設定します。
    - **このサイトで拡張機能を実行することを許可する**
    - **このサイトで信頼できる拡張機能を実行することを許可する**

サーバー管理者はサイトの許可リストで拡張機能を追加または削除できます。許可リストに拡張機能を追加する場合、拡張機能にフル データへのアクセス許可するパーミッションを設定するかどうかを制御できます。「許可リストへの拡張機能の追加と拡張機能の構成」を参照してください。

許可リストの使用

許可リストを使用して、サイトで特定の拡張機能を実行することを許可します。許可リストは、主にネットワーク対応拡張機能に使用されます。既知の Tableau ホストから提供され、既定で実行が許可されている Tableau の信頼できる拡張機能とは異なり、ネットワーク対応拡張機能は、明示的に許可リストに追加されていない限り、使用が許可されません。許可リストを使用すると、拡張機能の実行を許可するパーミッションをユーザーに求めるかどうか、および拡張機能がワークブックのフル データ(参照元データ) にアクセスできるかどうかを制御することもできます。

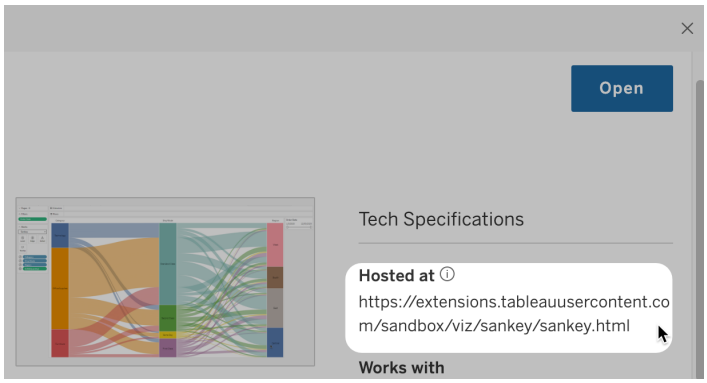
既定では、Tableau の信頼できる拡張機能の実行が許可されています。これには、サンドボックス拡張機能と Tableau が作成した拡張機能が含まれます。既定の設定が変更され、許可されなくなった場合を除き、それらを許可リストに追加する必要はありません。信頼できるパートナーが作成した拡張機能も同様であり、許可されている場合は、明示的に許可リストに追加する必要はありません。

拡張機能の URL を確認する

拡張機能は Web アプリケーションとして URL と関連付けられます。この URL を使用して、拡張機能のテストや検証を行うことができます。また、この URL は拡張機能を許可リストに追加してフルデータアクセスを許可したり、拒否リストに追加してアクセスを禁止したりするのにも使用します。
ネットワーク対応拡張機能またはその他の拡張機能を許可するには、拡張機能の URL を検索し、その URL を許可リストに追加する必要があります。URL を許可リストに追加すると、Web アプリケーションとの通信が承認されます。拡張機能の URL を特定するには、いくつかの方法があります。

Tableau Exchange から

Tableau Exchange からダウンロードした拡張機能を使用する場合は、Exchange のリストページで拡張機能の URL を見つけることができます。拡張機能のタイルを開き、[Tech Specifications (技術仕様)] の下にある **[Hosted at (ホスト先)]** という見出しの下の URL を探します。



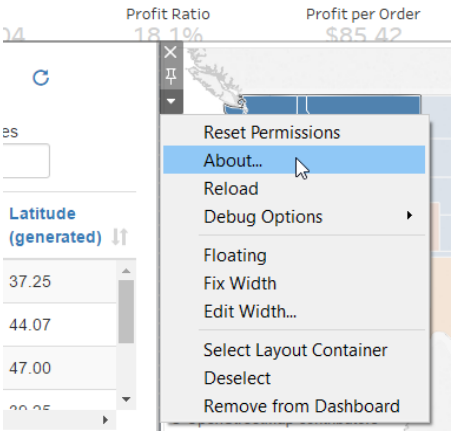
マニフェストファイルから

拡張マニフェストファイル (`.trex`)、つまり拡張機能のプロパティを定義する XML ファイルがある場合、URL を `<source-location>` 要素から見つけることができます。マニフェストファイルは、Web ブラウザーを使用して Tableau Exchange から拡張機能をダウンロードすると取得できます。テキストエディターでマニフェストファイルを開きます。

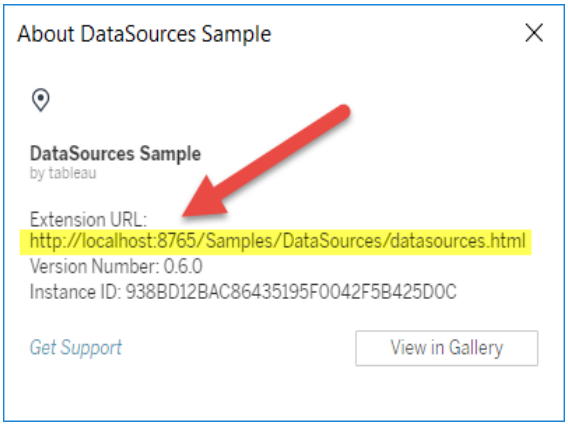
```
<source-location>
    <url>https://www.example.com/myExtension.html</url>
</source-location>
```

[詳細] ダイアログ ボックスを使用してダッシュボード拡張機能を確認する

ダッシュボードの拡張機能の場合、URL はダッシュボードの拡張機能のコンテキスト メニューにある拡張機能のプロパティから見つけることができます。**[その他のオプション]** メニューで **[詳細]** をクリックします。



[詳細] ダイアログ ボックスには、拡張機能の名前、作成者、作成者の Web サイト、拡張機能の URL がリストされています。



許可リストへの拡張機能の追加と拡張機能の構成

ユーザーがネットワーク対応拡張機能にアクセスできるようにするには、サイトの許可リストに拡張機能を追加します。既定では、許可リストに拡張機能を追加すると、その拡張機能は概要 (または集計) データにのみアクセスできます。また、拡張機能によるデータへのアクセスを許可するように求めるプロンプトをユーザーに表示するかどうかを制御することもできます。ユーザーにプロンプトを表示するかどうかを構成できるようにするために、拡張機能を許可リスト(たとえば、サンドボックス拡張機能) に追加する場合もあります。ユーザーにプロンプトが表示されないようにする場合は、拡張機能をすぐに実行できます。

1. **[設定] > [拡張]** に移動します。
2. **[特定の拡張機能の有効化]** で拡張機能の URL を追加します。拡張機能の URL を確認するを参照してください。

   ヒント: ピリオドとアスタリスク (`.*`) を URL 内のワイルドカードとして使用して、特定のドメインまたは場所ですべての拡張子を許可することができます。たとえば、ポート 8080 を使用する example.com のドメインですべての拡張機能を許可するには、URL `https://example.com:8080/.*` を追加します。詳細については、「許可リストの URL で正規表現を使用する」を参照してください。
3. 拡張機能の**フル データ アクセス**に関し、**[許可]** または **[拒否]** を選択してください。

   フル データ アクセスとは、単なるサマリー データや集計データではなく、ビューの参照元データへのアクセスです。フル データ アクセスには、接続、フィールド、表の名前など、データ ソースについての情報も含まれます。通常、使用したい拡張機能を許可リストに追加する場合、必要に応じて、拡張機能がフル データにアクセスできるようにすることもできます。許可リストに拡張機能を追加する前に、ネットワーク対応拡張機能のセキュリティテストするようにしてください。
4. **[ユーザー プロンプト]** の **[表示]** または **[非表示]** を選択します。

   ダッシュボード拡張機能をダッシュボードに追加している場合、Viz 拡張機能をワークシートに追加している場合、または拡張機能が含まれたビューを操作している場合は、既定でユーザーにプロンプトが表示されます。このプロンプトには、拡張機能の詳細および拡張機

能がフル データにアクセスできるかどうかが記載されています。このプロンプトによりユーザーには、拡張機能の実行を許可または拒否する権限が与えられます。ユーザーにプロンプトが表示されないようにし、拡張機能がすぐに実行されるようにできます。

拡張機能にワークブックまたはビューにアクセスするパーミッションがあることを確認する

ダッシュボードまたは Viz 拡張機能には、ビュー内の表示データへのアクセス権が必要です。表示データは、ワークシートまたはダッシュボードに表示されるサマリー集計データです。ワークブックまたはビューで **[サマリー データをダウンロードする]** パーミッションが許可されていることを確認します。拡張機能でデータがダウンロードされません。拡張機能には、ビュー内のデータを操作するためのパーミッションが必要です。

特定の拡張機能のブロック

既定のグローバル ポリシーでは、Tableau での拡張機能の使用が許可されています。既定では、サンドボックス拡張機能とTableau が作成した拡張機能の実行が許可されています。サイトでは、信頼できるパートナーが作成した拡張機能と、許可リストに表示されるネットワーク対応拡張機能を有効にすることができます。サーバー管理者は、サーバーの拒否リストに特定の拡張機能を追加することで、それが実行されないようにすることができます。拡張機能が全体の拒否リストに入っている場合、サイトの許可リストにある拡張機能の設定が上書きされます。

1. 拡張機能をサーバーで拒否リストに追加するには、**[すべてのサイトを管理] > [設定] > [拡張]** に移動します。単一サイトのインストールでは、拒否リストはサイトの **[拡張]** 設定ページにあります。

2. **[特定の拡張機能のブロック]** で拡張機能の URL を追加します。拡張機能の URL を確認するを参照してください。

許可リストの URL で正規表現を使用する

一般的に、拡張機能を許可リストに追加する場合は、拡張機能の特定の URL を使用する必要があります。ただし、同じドメインと場所からホストされている複数の拡張機能を許可したい場合もあります。この場合、URL でワイルドカードを使用すると便利です。拡張機能の設定では、正規表現の使用がサポートされています。

正規表現	説明
.	ピリオド(.) は、任意の文字と一致させるために使用できるワイルドカードです。URL にワイルドカードの代わりにピリオド(.) を指定する必要がある場合は、バックスラッシュ(\ .) を使用して文字をエスケープすることができます。
*	アスタリスク(*) は、前の文字の 1 つまたは複数のインスタンスを指定する量指定子です。

ワイルドカードを使用する場合は、許可リストの制限を緩和しすぎたり、アクセス不可の拡張機能へのアクセスを誤って許可したりしないように注意してください。

次の表は、URL で正規表現を使用する場合の例を示しています。これらの例では、拡張機能のプロトコルと完全な URL は示されていないことに注意してください。HTTPS プロトコルを使用する拡張機能のみが許可されます (http://localhost を除く)。

指定する対象	例	指定する内容
ドメインの範囲	.*\.example.com	example.com にあるすべてのサブドメイン。
すべてのポート	example.com:.*	example.com のすべてのポートからのアクセスが拡張機能に許可されます。
ドメイン、ポート、パスのすべての拡張機能	example.com:8080/xyz/.*	ポート 8080 を使用し、zyz に配置されているドメイン example.com にあるすべての拡張機能にアクセスが許可されます。

ドメインの範囲のすべてのポート	.*\.example.com:.*	example.com にあるすべてのサブドメインのすべてのポートで拡張機能へのアクセスを許可します。
パターンに一致するドメインとパスにあるすべての拡張機能	example.com/tc/.*	たとえば、tic、tac、toc など、パターン t.c に一致するフォルダーにある example.com で実行されている拡張機能へのアクセスを許可します。

ネットワーク対応拡張機能のセキュリティテスト

拡張機能とは、拡張 API を使用して Tableau のデータを操作する Web アプリケーションです。Tableau によって直接審査および管理されないネットワーク対応拡張機能を許可する場合、ユーザーは、これらの拡張機能がネットワークの呼び出しを行い、インターネット上のリソースにアクセスできる可能性があることに注意する必要があります。ネットワーク対応拡張機能を導入して広く展開する前に、ネットワーク対応拡張機能をテストして理解することをお勧めします。

ソース ファイルの調査

拡張機能には、さまざまな HTML、CSS、JavaScript ファイルや、拡張機能でプロパティを定義する XML マニフェスト ファイル (*.trex) が含まれています。ネットワーク対応拡張機能を審査する際は、拡張機能の開発者に直接連絡して、審査するためのアクセス権とパーミッションを取得することをお勧めします。Tableau は、他の開発者のソース ファイルの配布やアクセス権の提供は行いません。

拡張機能のコードは GitHub で公開されているため、そこで調査したり、ダウンロードしたりできます。マニフェスト ファイル (`*.trex`) ではソースの場所または URL を確認でき、ここでは拡張機能がホストされている場所、作成者の名前、サポートの問い合わせ先となる作成者または会社の Web サイトが示されています。`<source-location>` 要素は URL を指定し、`<author>` 要素はサポートの問い合わせ先である組織名と Web サイトを指定します (`website="SUPPORT_URL"`)。Web サイトは、ダッシュボード拡張機能の **[詳細]** ダイアログ ボックスでユーザーに表示される **[サポートを受ける]** のリンクです。

拡張機能は、jQuery ライブラリやサードパーティ—向けの API ライブラリなどの外部 JavaScript ライブラリを参照します。外部ライブラリの URL がライブラリの信頼できる場所を指していることを検証します。たとえば、拡張機能が jQuery ライブラリを参照する場合、ライブラリが標準および安全と見なされるサイト上に配置されていること確認してください。

すべての拡張機能では、拡張機能のホストに HTTPS プロトコル (`https://`) を使用することが求められています。拡張機能のソース ファイルを調査し、外部ライブラリへの参照が HTTPS をやはり使用していること、または拡張機能と同じ Web サイトでホストされていることを確認します。HTTPS の要件に関する 1 つの例外は、拡張機能が Tableau (`http://localhost`) と同じ PC でホストされているかどうかです。

可能な限り、コードの内容を理解するようにしてください。具体的には、コードが外部サイトへの要求をどのように構築しているか、また要求によりどのような情報が送信されるかを理解するように努めてください。特に、クロスサイト スクリプティングを防ぐために、ユーザーにより提供されたデータが検証されているかどうかを確認してください。

データ アクセスの理解

Tableau 拡張機能は主に、ワークシートまたはダッシュボードに表示されるサマリー集計データを操作します。ただし、Tableau 拡張 API は、拡張機能がフル データ アクセスの必要性を宣言した場合に、参照元データにアクセスできるメソッドを提供します。これらのメソッドを使って、データ ソースのアクティブなテーブルやフィールドの名前、データ ソース接続のサマリーの説明、ワークシートの参照元データにアクセスできます。ビューで拡張機能がこれらのメソッドのいずれかを使用している場合、拡張機能の開発者は、拡張機能にはマニフェスト ファイル (`.trex`) にフル データのパーミッションが必要であることを宣言しなければいけません。宣言は次のようになります。

```
<permissions>
   <permission>full data</permission>
</permissions>
```

Tableau はこの宣言を使用して、実行時にこのアクセスを許可するオプションを示すプロンプトを表示します。拡張機能でこれらのメソッドのいずれかが使用されているものの、マニフェストファイルでフルデータパーミッションが宣言されていない場合、拡張機能は読み込まれますが、メソッドの呼び出しは失敗します。

拡張機能がダッシュボードからデータにアクセスする方法、および使用されている JavaScript メソッドの詳細については、Tableau 拡張 API の「参照元データへのアクセス (英語)」を参照してください。データに関して拡張機能で確認できる内容についてさらに理解するには、データソースサンプルのダッシュボードの拡張機能 (Tableau 拡張 API GitHub リポジトリより入手可能) を使用し、`getDataSourcesAsync()` メソッドが呼び出されるときにどのデータが表示されるかを確認できます。

隔離された環境での拡張機能のテスト

可能な場合は、本番環境とユーザーの PC から隔離された環境で拡張機能をテストすることをお勧めします。たとえば、本番環境で使用されていない Tableau Server のバージョンを実行しているテスト PC または仮想マシンの許可リストにダッシュボードまたは Viz の拡張機能を追加します。

ダッシュボードの拡張機能により作成されたトラフィックの監視

ネットワーク対応拡張機能が使用する可能性のあるネットワークの呼び出しを評価するには、Fiddler、Charles HTTP プロキシ、または Wireshark などのツールを使用して、拡張機能で実行する要求や応答を調査します。拡張機能が要求しているのがどのコンテンツかを把握するようにしてください。拡張機能が、その目的に直接関連しないデータやコードを読み取っていないことを確認するために、トラフィックを確認してください。

分析拡張機能を使用した接続の構成

Tableau では、分析拡張機能に式を渡して、R や Python、Einstein Discovery と統合できるようにする関数セットがサポートされています。

注: R および Python スクリプトを使用すると、Tableau Prep フローで複雑なクリーニング操作を実行することができますが、サポートされる構成と機能が異なる場合があります。詳細については、Tableau Prep ヘルプの「フローでの R および Python スクリプトの使用」を参照してください。

このトピックでは、拡張機能を使用してサイトを Tableau Server で構成する方法について説明します。

Tableau Server は認証機構を提供しているため、Tableau Desktop より Tableau Server の方がより安全に分析拡張機能をユーザーに表示できます。

ユーザーシナリオの詳細と Tableau Desktop または Tableau Cloud の構成については、*Tableau Desktop と Web 作成のヘルプ*の「分析拡張機能を使用した式の受け渡し」を参照してください。

この記事の構成ステップは、ワークブックに固有のものです。R および Python スクリプトを使用して予測モデリングデータをフローに組み込む方法については、*Tableau Prep* ヘルプの「フローでの R および Python スクリプトの使用」を参照してください。

Server SSL

分析拡張用に SSL を構成するには、Tableau Server を実行しているコンピューターに有効な証明書をインストールする必要があります。証明書は Tableau Server を実行しているコンピューターによって信頼されている必要があります。証明書の [件名] フィールドや SAN エントリの 1 つが、分析拡張機能サービス構成の URI と完全に一致する必要があります。

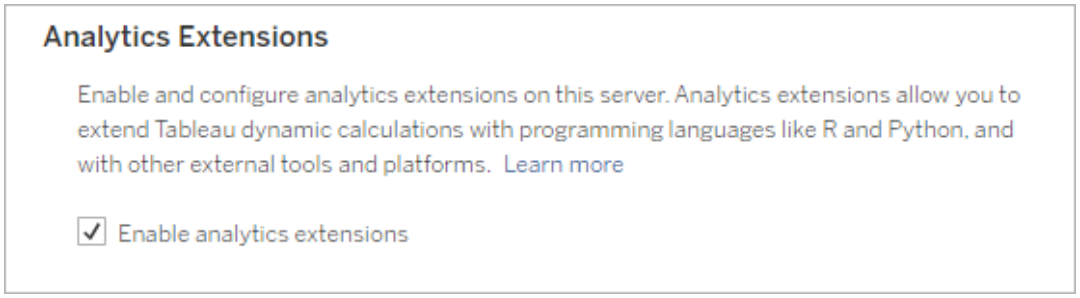
分析拡張機能を有効にする

拡張機能を構成する前に、サーバー全体で分析拡張機能を有効にする必要があります。

1. Tableau Server 管理者エリアへのサインインします。
    - サーバー上にサイトが 1 つしかない場合は (既定)、**[設定]** をクリックし、ステップ 2 に進みます。

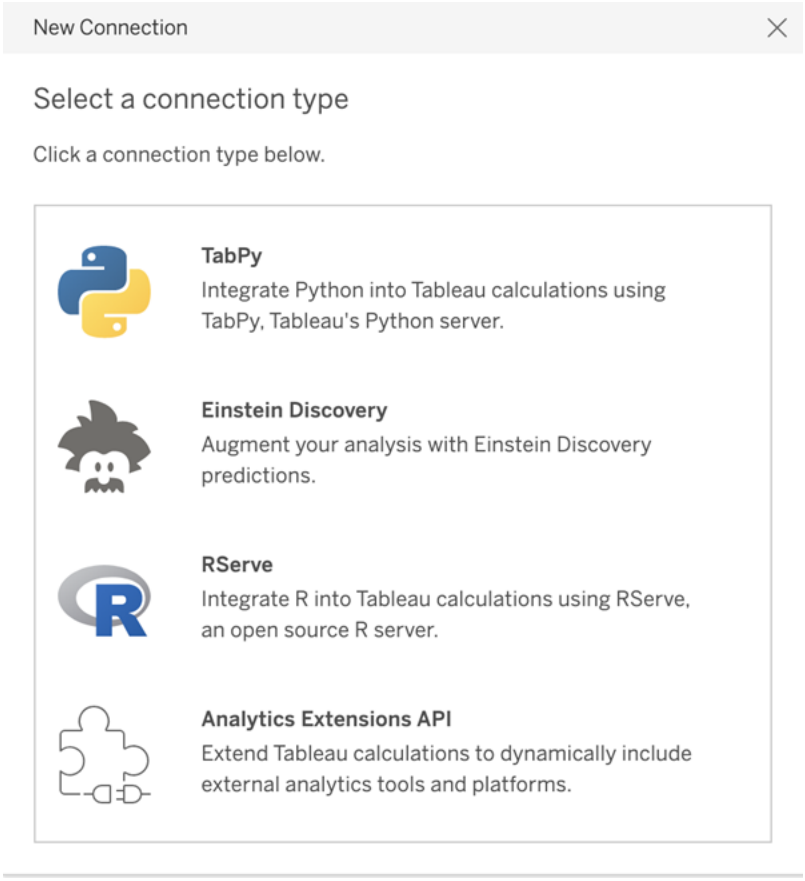
- サーバーに複数のサイトがある場合は、
  a. **[すべてのサイト]** で、**[すべてのサイトを管理]** をクリックします。
  b. **[拡張]** タブをクリックします。

2. **[分析拡張機能]** までスクロールし、**[Enable analytics extension (分析拡張機能を有効にする)]** を選択して **[保存]** をクリックします。



分析拡張機能設定の構成

1. Tableau Server 管理者エリアへのサインインします。
2. [設定] ページで **[拡張機能]** タブをクリックし、**[分析拡張機能]** までスクロールします。(Tableau Server のマルチサイト展開で、分析拡張機能を構成するサイトに移動し、**[設定] > [拡張機能] をクリック**します。)
3. **マルチサイト展開のみ:** 各サイトで分析拡張機能を有効にする必要があります。[分析拡張機能] で、**[Enable analytics extensions for site (サイトの分析拡張機能を有効にする)]** を選択します。
4. [分析拡張機能] で、**[新しい接続の作成]** をクリックします。
5. **[新しい接続]** ダイアログで、追加する接続タイプをクリックし、分析サービスの構成設定を入力します。



6. 構成する必要があるオプションは、選択した接続タイプによって異なります。

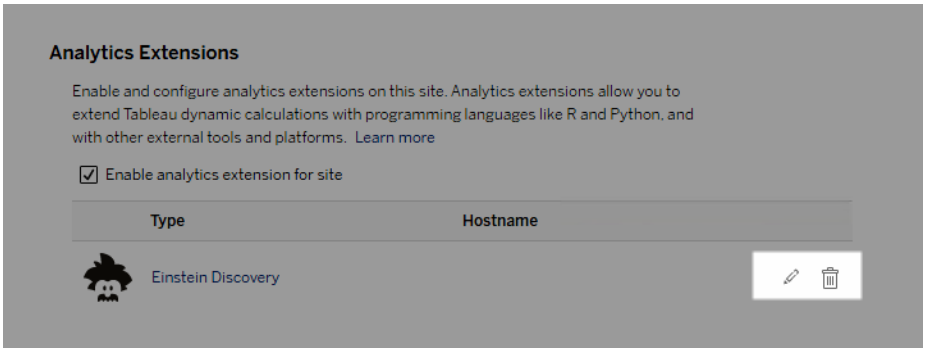
   - Einstein Discovery 接続の場合は、**[有効]** をクリックします。
   - TabPy、RServer、および分析拡張機能 API 接続の場合は、次の情報を入力します。
     - **接続名**: (必須) 接続するサーバーのタイプを指定します。RSERVE は、RServe パッケージを使用した R への接続に対応します。TABPY は、TabPy を使用した Python への接続やその他の分析拡張機能への接続に対応します。
     - **SSL を必須にする** (推奨): 分析サービスへの接続を暗号化するには、このオプションを選択します。**[ホスト名]** フィールドで `HTTPS` URL を指定する場合は、このオプションを選択する必要があります。
     - **ホスト名**: (必須) 分析サービスを実行しているコンピューター名や URL を指定します。このフィールドでは大文字と小文字を区別します。
     - **ポート**: (必須) サービスのポートを指定します。

- **ユーザー名 およびパスワードを指定してサインイン**(推奨): 分析サービスの認証に使用するユーザー名とパスワードを指定するには、このオプションを選択します。

7. **[作成]** をクリックします。

分析拡張機能接続の編集または削除

構成を編集または削除するには、サイトの **[拡張機能]** タブにある **[分析拡張機能]** に移動します。



[編集] アイコンまたは **[削除]** アイコンをクリックし、プロンプトに従って構成を変更します。

クライアント要件: Rserve 外部サービスの中間証明書チェーン

Tableau Server 経由で Rserve 外部接続に接続している Tableau Desktop コンピューター (Windows と Mac) に完全な証明書チェーンをインストールする必要があります。この要件は、Rserve がセキュア接続でハンドシェイクを管理する方法が原因です。

Tableau Desktop にルート証明書をインポートするだけでは不十分であり、証明書チェーン全体をクライアント コンピューターにインポートする必要があります。

スクリプト エラー

Tableau では、分析拡張機能を使用するワークブックが Tableau Server に適切にレンダリングされることを確認できません。必須の統計ライブラリがユーザーのコンピューター上では使用できても Tableau Server で使用している分析拡張機能インスタンス上では使用できない場合があります。

分析拡張機能を使用するビューがワークブックに含まれている場合、ワークブックをパブリッシュするときに警告が表示されます。

> このワークシートには外部サービス スクリプトが含まれており、管理者が外部サービス接続を構成するまでターゲット プラットフォーム上では表示できません。

分析拡張機能の使用状況の判定

バージョン 2021.2 以降、分析拡張機能の設定はワークブック レベルで反映されます。これにより、管理者はカスタム ビューを使用して Tableau リポジトリにクエリを実行することができ、どのワークブックでどの拡張機能が使用されているか、それらはどのくらいの頻度で使用されているかを判断できます。

これを行うには、ワークブック接続テーブルをワークブックの使用状況を示すテーブルに結合する必要があります。カスタム管理ビューの作成と使用の詳細については、「Tableau Server リポジトリでデータを収集する」および「カスタム管理ビューの作成」を参照してください。

テーブル拡張機能

テーブル拡張機能を使用すると、分析拡張機能スクリプトで新しいデータ表を作成できるようになります。カスタムの TabPy または Rserve スクリプトを記述したり、オプションで 1 つまたは複数の入力表を追加したりできます。テーブル拡張機能は、Tableau Cloud、Tableau Server、Tableau Desktop でサポートされています。このドキュメントは、Tableau Server に焦点を合わせています。

> **注:** ワークブックを開いたり、データ ソースを更新したりするたびにデータが更新されます。

利点

テーブル拡張機能は、新規ユーザーと経験豊富なユーザーの両方に次の利点を提供します。

- より高速なデータ処理
- ローコード エディター
- 「データに聞く」と「データの説明を見る」の統合

- TabPy や Rserve との統合
- 結果を使用してダッシュボードや Viz の作成が可能

前提条件

テーブル拡張機能を使用する前に、次の操作を行う必要があります。

- 分析拡張機能の構成
  - 分析拡張機能の接続を構成する手順については、「分析拡張機能を使用した接続の構成」を参照してください。
- ワークブックをパブリッシュする

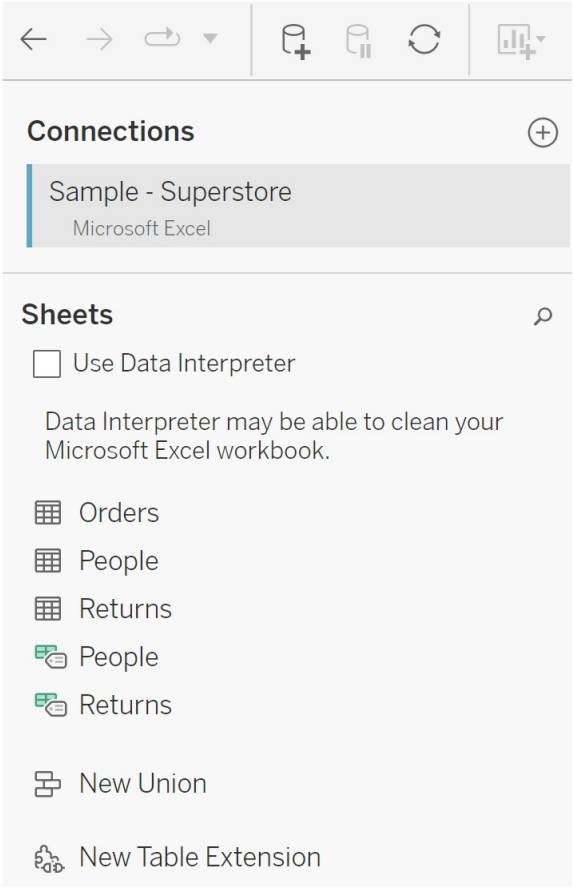
テーブル拡張機能の作成

新しいテーブル拡張機能を作成するには、次の手順を実行します。

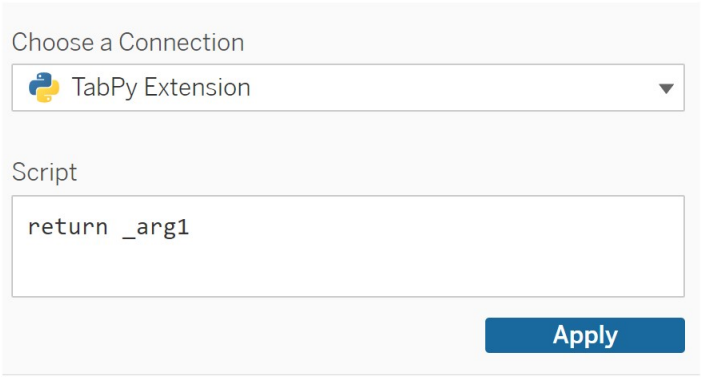
1. パブリッシュされたワークブックを開きます。

> **注:** テーブル拡張機能を追加する前に、ワークブックをパブリッシュする必要があります。

2. **[シート]** の下にある **[新しいテーブル拡張機能]** を選択します。

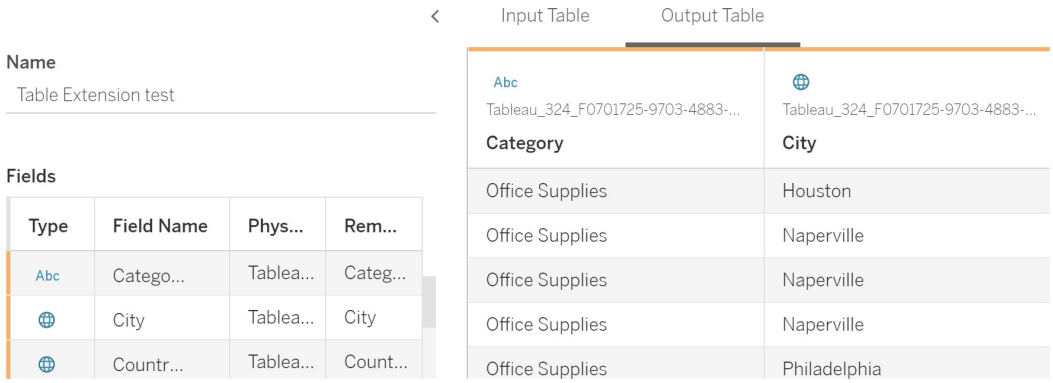


3. (オプション) シートを [テーブル拡張機能] ペインにドラッグします。
4. **[接続の選択]** で、分析拡張機能を選択します。



5. **[スクリプト]** にスクリプトを入力します。

6. **[適用]** を選択します。
7. **[今すぐ更新]** を選択すると、**[出力テーブル]** タブに結果が表示されます。



8. **[名前]** フィールドに、新しいテーブル拡張機能の一意の名前を入力します。
9. [シート] タブに移動して、ワークブックをパブリッシュして保存します。

> **注:** 入力テーブルを編集した場合は、更新された出力テーブルデータを表示または使用する前に、**[適用]** をもう一度押す必要があります。

トラブルシューティングのヒント: テーブル拡張機能にエラーが発生した場合は、**[保存]** ボタンの横にある円形の **[データソースの更新]** ボタンを使用してみてください。

テーブル拡張機能と分析拡張機能

テーブルには、名前に「拡張機能」が付いた機能がいくつかあります。一部の機能は相互に関連していませんが、テーブル拡張機能と分析拡張機能は関連しています。テーブル拡張機能は、分析拡張機能との接続を介して機能します。それぞれの機能を詳しく見てみましょう。

テーブル拡張機能

テーブル拡張機能では、ワークブックの計算を作成し、分析拡張機能にデータと処理スクリプトを送信することができます。返された結果は、[データソース] タブのテーブルとして、およびワークブックのメジャーとディメンションとして表示されます。

分析拡張機能

分析拡張機能では、Python などのプログラミング言語、外部ツール、外部プラットフォームを使って Tableau の動的な計算を拡張することができます。分析拡張機能への接続を作成すると、計算フィールドを介して外部サーバーと通信できるようになります。詳細については、「分析拡張機能を使用した接続の構成」を参照してください。

Einstein Discovery 統合の構成

バージョン 2021.1.0 以降の Tableau Server では Einstein Discovery と統合することができます。ワークブックやダッシュボードの作成者と閲覧者が Einstein Discovery の予測を利用できるようになりました。バージョン 2021.2.0 以降では、Web 上でフローを作成するときに Einstein Discovery 予測も利用できるようになりました。

Tableau の Einstein Discovery は salesforce.com が提供しています。適用条件については、salesforce.com との契約を確認してください。

Tableau Server とEinstein Discovery を統合するには、いくつかの構成手順を実施する必要があります。手順のいくつかは Tableau Server 側で必要であり、手順のいくつかは Einstein Discovery を実行している Salesforce 組織側で必要です。ここでは、ダッシュボードの拡張機能、分析の拡張機能、および Tableau Prep 拡張機能の手順の概要を説明し、サーバーの構成手順が記載されている特定のトピックへのリンクを示します。

Tableau における Einstein Discovery 予測の使用方法 (ライセンス発行とパーミッションの要件を含む) については、Tableau Desktop および Web 作成ヘルプの「Tableau の Einstein Discovery 予測の統合」を参照してください。予測をフローに追加する方法については、「Einstein Discovery 予測をフローに追加する」を参照してください。

Einstein Discovery ダッシュボードの拡張機能

Einstein Discovery のダッシュボード拡張機能を使用すると、ワークブックの作成者は Tableau で予測をリアルタイムに表示できます。ダッシュボード拡張機能は、Tableau ワークブックのソースデータと、Salesforce にデプロイされた Einstein Discovery を利用したモデルを使用して、インタラクティブにオンデマンドで予測を提供します。

Einstein Discovery ダッシュボードの拡張機能を使用するように Tableau Server を構成するには、以下を行う必要があります。

1. Tableau Server で次の手順を実行します。
    a. Tableau Server のデータ接続と拡張機能に対して OAuth の保存済みトークンを有効にします。保存済みアクセス トークンの許可
    b. サーバーのダッシュボード拡張機能を有効にします。「Tableau Server でダッシュボードと Viz の拡張機能を管理する」を参照してください。
2. Salesforce で、Einstein Discovery を実行している組織において次の手順を実行します。
    a. Salesforce.com で CORS を構成して、Einstein Discovery を Tableau Server に統合する。
    b. Salesforce で、Tableau CRM を実行している組織において接続アプリを作成します。「ステップ 1: Salesforce に接続されたアプリを作成する」を参照してください。
3. Tableau Server で、接続アプリからの情報を使用して、Salesforce の保存済み OAuth 認証資格情報用にサーバーを構成します。「ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する」

Einstein Discovery の分析拡張機能

Einstein Discovery 分析拡張機能を使用すると、Tableau の計算フィールドに予測を直接埋め込むことができます。Salesforce にデプロイされたモデルからの予測をテーブル計算スクリプトで要求するには、関連する予測 ID とモデルに必要な入力データを渡します。Salesforce の Model Manager を使用して Tableau のテーブル計算スクリプトを自動生成し、そのスクリプトを Tableau ワークブックで使用できるように計算フィールドに貼り付けます。

Einstein Discovery の分析拡張機能を使用するように Tableau Server を構成するには、以下を行う必要があります。

1. Tableau Server で次の手順を実行します。
    a. Tableau Server のデータ接続と拡張機能に対して OAuth の保存済みトークンを有効にします。保存済みアクセス トークンの許可
    b. サーバーの分析拡張機能を有効にし、接続タイプを構成します。分析拡張機能を使用した接続の構成を参照してください
2. Salesforce で、Einstein Discovery を実行している組織の接続アプリを作成します。ステップ 1: Salesforce に接続されたアプリを作成するを参照してください。

3. Tableau Server で、接続アプリからの情報を使用して、Salesforce の保存済み OAuth 認証資格情報用にサーバーを構成します。「ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する」

Einstein Discovery Tableau Prep の拡張機能

バージョン 2021.2.0 以降の Tableau Server および Tableau Cloud でサポートされています

Einstein Discovery Tableau Prep の拡張機能を使用すると、ユーザーは Web 上でフローを作成するときに、Einstein 予測をフローに直接埋め込むことができます。

Einstein Discovery Tableau Prep の拡張機能を使用するように Tableau Server または Tableau Cloud を構成するには、次を行う必要があります。

1. Tableau Server で次の手順を実行します。
   a. Tableau Server のデータ接続と拡張機能に対して OAuth の保存済みトークンを有効にします。保存済みアクセストークンの許可を参照してください。
   b. サーバーに対して Tableau Prep 拡張機能を有効にします。Tableau Prep 拡張機能を有効にする参照してください。
2. Salesforce で、Einstein Discovery を実行している組織の接続アプリを作成します。ステップ 1: Salesforce に接続されたアプリを作成するを参照してください。
3. Tableau Server で、接続アプリからの情報を使用して、Salesforce の保存済み OAuth 認証資格情報用にサーバーを構成します。「ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する」

外部アクション ワークフロー統合の構成

注: Tableau の外部アクションは、Salesforce フローから提供される機能を使用しています。この機能は、選択したデータを、別個の Salesforce インフラストラクチャで実行されている Salesforce フローに送信します。Salesforce フローとその他の Salesforce 製品およびサービスの使用には、Salesforce との契約が適用されます。

外部アクションの使用方法の詳細については、「外部アクションの統合」を参照してください。

エディションロール、パーミッションの要件

外部アクションワークフローを構成して使用する場合、ワークフローのすべてのユーザーは、外部アクションをサポートするSalesforceおよびTableauエディションのサイトロールとパーミッションを持つ必要があります。

製品	エディション	サイトロールとパーミッション
Tableau	Tableau Cloud、Tableau Desktop、またはTableau Serverバージョン2022.3以降	**ワークフローを作成または編集する場合:** CreatorまたはExplorer (パブリッシュ可能) サイトロールと、ワークブックを編集および保存するパーミッション (Linux | Windows) **データを送信する場合**: 概要データをダウンロードする パーミッション機能 **ワークフローを使用する場合:** すべてのサイトロール
Salesforce	Essentials、Professional、Enterprise、Performance、Unlimited、またはDeveloperエディション (「Salesforceエディション」を参照してください)。	**フローを作成または編集する場合:** フローを管理するパーミッション **フローを使用する場合:** フローを実行するパーミッション、ユーザーの詳細ページにあるフローユーザー設定、または*フローノードで既定の動作を上書きし、有効化されたプロフィールやパーミッションセットへのアクセスを制限する*設定 (「フローセキュリティの仕組み」を参照してください)

外部アクションの導入要件

外部アクションワークフロー拡張機能を使用する場合、Salesforce管理者は、SalesforceでTableau Server用の接続アプリケーションを作成する必要があります。

また、Tableau Server または Tableau Cloud サイトのドメイン (拡張機能が展開される) を Salesforce クロス オリジン リソース共有 (CORS) の許可リストに追加する必要があります。

外部アクションをオンまたはオフにする

Tableau 2022.3 より、Tableau 外部アクション機能は既定でオンになっています。この機能は、Tableau サービスマネージャー (TSM) コマンド ライン インターフェイス (CLI) を使用するか、サイトレベルの設定を構成してオフにすることができます。

TSM CLI の使用

TSM CLI を使用して、`vizqlserver.workflow_objects_enabled` 構成キーの値を `tsm configuration set -k vizqlserver.workflow_objects_enabled -v false` に設定します。

構成をキーの値を変更した後、`tsm pending-changes apply` コマンドを使用して、変更を適用する必要があります。保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージを表示しないようにすることができます。ただし、このオプションにより、再起動に関する動作が変わることはありません。変更を適用する際に、サーバーの再起動が必要ない場合は、プロンプトは表示されずに変更が適用されます。詳細については、「tsm pending-changes apply」を参照してください。

サイトレベルの設定の変更

Tableau Server でサイトレベルの設定をカスタマイズする方法の詳細については、「サイト設定リファレンス」を参照してください。

Tableau と Slack ワークスペースの統合

Tableau Server と Tableau Cloud は、バージョン 2021.3 以降、Tableau App for Slack との統合をサポートするようになりました。これにより、Slack ワークスペースで作業をしているチームメイトは、まさにその作業環境で、Tableau スナップショットを共有し、Tableau コンテンツを検索し、Tableau データに関する通知を受信することができます。

Tableau App for Slack を使用すると、Tableau サイトを Slack ワークスペースに接続できます。これを有効にすると、Tableau ユーザーは次のことができます。

- コンテンツがチームメイトに共有されたとき、コメントでメンションされたとき、データドリブンアラートでデータが所定のしきい値に達したときに、Slack で通知を受信できます。Tableau Cloud のサイト管理者または Tableau Server のサーバー管理者がサイトで通知を有効にしていれば、ユーザーはアカウント設定を構成することで、どの通知を Slack で受信するかを制御できます。
- Tableau の URL を Slack に貼り付けると Viz のプレビューが表示されるため、ユーザーはデータ関連のコンテンツを Slack のコンテキストで直接共有できます。
- Slack の DM およびチャネルで、Tableau ビューまたはワークブックを検索できます。
- Tableau App for Slack から最近使用したアイテムとお気に入りにアクセスできます。

詳細については、「Tableau App for Slack を使用した通知の受信、検索、共有」を参照してください。

> **注:**: サイトで機能がオフになっている場合、一部の通知設定は使用できない場合があります。例えば、[ユーザーの表示状態] が [制限付き] に設定されている場合、通知はオフになります。詳細については、「サイト設定リファレンス」および「サイトユーザーの表示状態の管理」を参照してください。

Tableau サイトと Slack を統合するには、いくつかの設定手順を実施する必要があります。手順のいくつかは Tableau サイト側で必要であり、いくつかは接続先の Slack ワークスペース側で必要です。この概要では、Tableau Cloud のサイト管理者または Tableau Server のサーバー管理者と、Slack ワークスペースの管理者の両方の手順を説明します。

要件

Slack で Tableau を有効にするには、Slack ワークスペースの管理者と、Tableau Cloud のサイト管理者または Tableau Server のサーバー管理者の両方が必要です。

Tableau Server サイトを Slack ワークスペースと統合する

Tableau Server の管理者は、Slack ワークスペースを 1 つの Tableau Server サイトに接続できます。Tableau サイトを Slack ワークスペースに接続するには 3 つタスクを実施します。

- **Slack ワークスペース管理者**: Slack API プラットフォームでプライベートの Slack アプリケーションを作成します。
- **Tableau server 管理者**: アプリ情報を使用して OAuth クライアントを追加します。
- **Tableau Server 管理者**: Tableau サイトを Slack に接続します。

Tableau サイトごとに、1 つの Slack ワークスペースに接続することができます。Slack アプリの作成については、Slack の「Slack プラットフォームに関するベストプラクティスとガイドライン」を参照してください。

注: プロキシ サーバーを使用している場合は、開始する前に「Tableau Server のプロキシとロードバランサーの設定」のガイダンスに従っていることを確認してください。

Windows の場合:

- Windows 環境変数に `http_proxy` と `https_proxy` がすでに指定されていることを確認してください。環境変数を指定する手順については、「Tableau Server のプロキシとロードバランサーの設定」を参照してください。
- このリストの Slack ドメインを許可リストに追加します。詳細については、「インターネットとの通信」を参照してください。

ステップ 1: Tableau App for Slack を作成する

Slack ワークスペース管理者

1. Slack の「API ドキュメント」に移動し、**[新しいアプリの作成]** を選択します
2. **[最初から]** を選択し、アプリ名とアプリが存在するワークスペースを追加します。**[新しいアプリの作成]** を選択します。
3. アプリに名前を付けて、Slack ワークスペースを選択します。
4. アプリの基本情報設定が表示されます。ここで、アプリの権限や説明などを変更できます。
5. [機能の追加] セクションで **[ボット]** を選択し、**[追加するスコープの確認]** を選択します。
6. [ボット トークン スコープ] で、次のスコープを追加します。
    - `chat:write`
    - `files:read`
    - `files:write`
    - `users:read`
    - `users:read.email`

7. ナビゲーション メニューで **[OAuth とパーミッション]** を選択します。

   重要: Tableau App for Slack のトークン ローテーションをオプトインすると、通知が Slack で機能しなくなります。トークン ローテーションを追加した後は削除できません。

8. **[新しいリダイレクト URL の追加]** を選択します。
9. 完全修飾 URL `https://<Tableau Server URL>/auth/add_oauth_token` を追加します。
10. ナビゲーション メニューで **[基本情報]** を選択します。
11. クライアント ID、クライアント シークレット、およびリダイレクト URL を Tableau Server 管理者に伝えます。
12. [設定] メニューから [基本情報] を選択して、Tableau App for Slack を Slack ワークスペースに追加し、**インストール**します。

注: 管理者は、配布オプションを有効にすることで、他の Slack ワークスペースとアプリを共有できます。詳細については、Slack のドキュメントの「配布の管理」を参照してください。

ステップ 2: Tableau サイトに OAuth クライアントを追加する

Tableau Server 管理者:

Tableau Server 管理者は、Slack ワークスペース管理者から入手した Oauth クライアントを追加し、Tableau サイトを Slack に接続します。

1. Slack に接続しようとしているサイトにサインインします。サイトの [設定] ページで、**[統合]** タブを選択します。

2. [Slack 接続] で、**[OAuth クライアントの追加]** を選択します。
3. Slack ワークスペース管理者から入手した**クライアント ID**、**クライアント シークレット**、**リダイレクト URL** を追加し、ダイアログで **[OAuth クライアントの追加]** を選択します。
4. 接続タイプとクライアント ID がテーブルに表示されます。

ステップ 3: 接続を完了する

Tableau Server 管理者:

OAuth クライアントが [Slack 接続] に追加されたら、以下を実施します。

1. **[Slack に接続]** を選択します。
2. プロンプトに従って、Slack ワークスペースにサインインします。
3. **[許可]** を選択して、Tableau サイトが Slack ワークスペースにアクセスできるようにします。

これで、Tableau サイトと Slack ワークスペースが接続されました。Slack ワークスペースで、誰かが Tableau コンテンツを共有したとき、データ駆動型アラートがトリガーされたとき、ビューやワークブックのコメントで誰かに @ メンションされたときなどに、ライセンスのある Tableau ユーザーは Slack 通知を受け取ることができます。

Tableau サイトを Slack から切断する

サイト管理者は、サイト設定の **[統合]** タブで **[Slack から切断]** を選択することで、Slack ワークスペースから Tableau サイトを切断できます。ユーザーはしばらくの間、引き続き通知を受け取ります。ステップ 2 で追加した OAuth クライアントの情報は保持され、必要に応じて新しいワークスペースに接続するために使用できます。

Slack ユーザーは、Tableau App for Slack の **[ホーム]** タブから **[Tableau から切断]** を選択することで、Slack から切断できます。

Tableau App for Slack を更新する

Tableau App for Slack の新しいバージョンが利用可能になった場合は、アプリケーションを更新してアプリのパフォーマンスを維持し、新しい機能を使用することをお勧めします。

Tableau App for Slack を更新するには、以下を実施します。

1. サイトの **[設定]** ページから、**[統合]** タブを選択します。

2. **[Slack 接続]** で、**[Slack に接続]** を選択します。
3. **[更新]** を選択します。

> **注:** Tableau 管理者によって適用されたアプリの更新は、同じワークスペースに接続されているすべての Tableau サイトに影響します。

Tableau App for Slack のトラブルシューティング

Slack ワークスペースのアプリ承認を管理することがベストプラクティスです。ただし、Slack ワークスペースで管理者以外のユーザーがアプリを削除できる場合、ユーザーがサイトの Slack 用 Tableau アプリを削除できる可能性があります。このシナリオでは、他のユーザーには Tableau App for Slack が表示されますが、その機能は期待どおりに動作しません。この問題を解決するには、Slack 管理者に Tableau App for Slack をアンインストールしてから再インストールしてもらいます。次に、管理者に手順に従って Slack を Tableau サイトに接続してもらいます。

Tableau リンクのプレビューが Slack で表示されない場合は、Tableau アプリで **files:read** トークンが Slack パーミッション スコープの**ボット**を許可していることを確認してください。

Creators: Web 上のデータへの接続

ワークブックを作成し、Web 上でビューを構築してデータを分析する前に、データに接続する必要があります。Tableau では、Tableau Desktop からパブリッシュされた Web 上のデータ ソースへの接続や、Tableau Cloud、Tableau Server、Tableau Public を介したデータへの直接接続がサポートされています。

2019.3 以降、Tableau Catalog は Tableau Server と Tableau Cloud のデータ管理の一部として使用できます。Tableau Catalog がお使いの環境で有効な場合は、[探索] からデータに移動して接続するだけでなく、Tableau Catalog から、データベースやテーブルなどのさまざまなタイプのデータに移動して接続することもできます。Tableau Catalog の詳細については、Tableau Server または Tableau Cloud のヘルプの「Tableau Catalog について」を参照してください。2021.4 以降、データ管理には、データへの中心的なアクセス ポイントである仮想接続が含まれています。詳細については、Tableau Server または Tableau Cloud ヘルプの「仮想接続とデータ ポリシーについて」を参照してください。

[データに接続] ページを開く

Web では接続するデータにアクセスするのに [データに接続] ページを使用します。Tableau Server または Tableau Cloud にサインインした後、次の 2 つの方法でこのページを開くことができます。

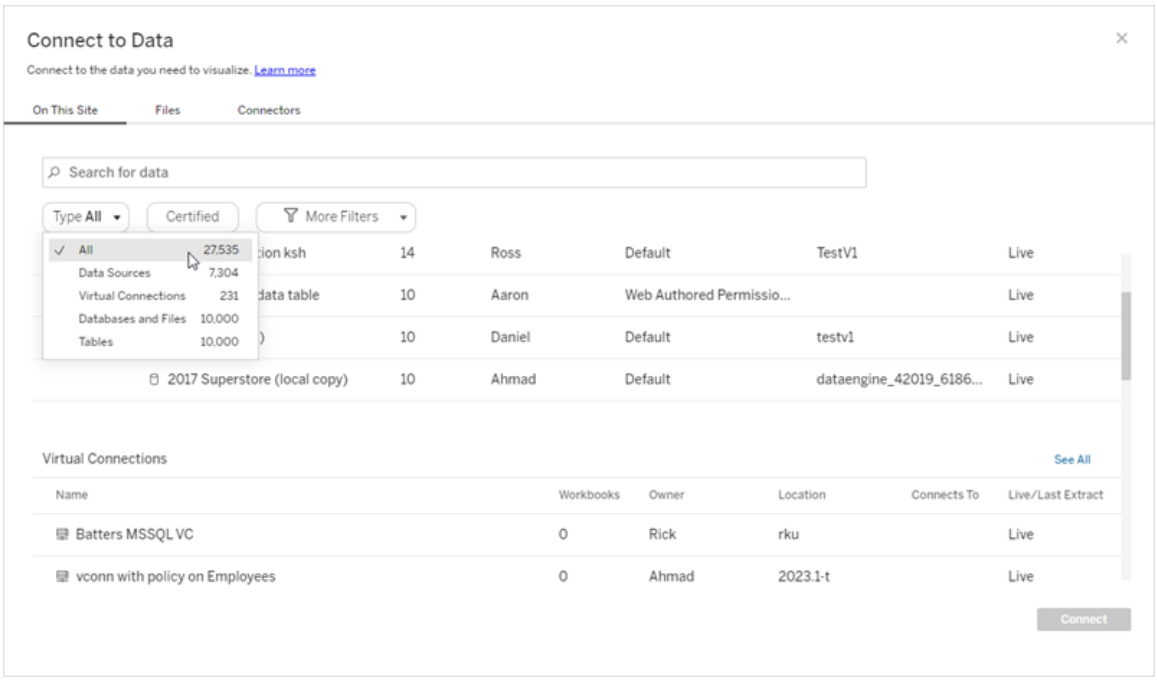
- **[ホーム] > [新規作成] > [ワークブック]**
- **[探索] > [新規作成] > [ワークブック]**

Tableau Public を使用している場合は、作成者プロフィールからこのページを開くことができます。

- **[マイ プロフィール] > [Viz の作成]**

[データに接続] ダイアログには、人気のある混合コンテンツのスクロール可能なリストが表示されます。データ管理 ライセンスを持っている場合は、仮想接続を使用してデータに接続することができます。また、データ管理 を持っていて Tableau Catalog を有効にしている場合は、データベース、ファイル、テーブルなどの外部アセットに接続することができます。

レスポンシブ検索フィールドには、テキストを入力すると更新される候補のリストが表示されます。データのタイプ、認証状況、または選択したデータのタイプに依存するその他のフィルターを使用して結果を絞り込むことができます。たとえば、データのタイプによっては、タグ、接続タイプ、データ品質に関する警告、またはその他の基準に基づいてフィルターできる場合があります。旧バージョンのダイアログの外観と機能は若干異なりますが、全体的な機能はほぼ同じです。



[データに接続] ページに表示されるタブは、使用している製品によって異なります。

Tableau Server

Tableau Server では、[このサイト上]、[ファイル]、[コネクタ] のタブから選択してデータへの接続を行います。

[このサイト上] のデータへの接続

1. **[このサイト上]** は、パブリッシュされたデータソースの参照や検索を行う場合に選択します。
2. **[名前]** よりデータソースを選択し、**[接続]** ボタンをクリックします。

> **注:** データソースへの接続に加えてデータ管理がある場合は、**[このサイト上]** を選択すると仮想接続を使用してデータに接続できます。Tableau Catalog を有効にすると、データベース、ファイル、およびテーブルにも接続できます。

ファイルへの接続

Tableau は、Excel、テキストベースのデータソース (.xlsx、.csv、.tsv)、およびブラウザーで直接 1 つのファイル (.kml、.geojson、.topojson、.json、.zip にパッケージ化された Esri シェープファイルおよび

Esri File Geodatabases) のみを必要とする空間ファイル形式のアップロードをサポートしています。**[データに接続]** ペインの **[ファイル]** タブで、ファイルをフィールドにドラッグ アンド ドロップするか、**[コンピューターからアップロード]** をクリックして、ファイルに接続します。アップロードできるファイルの最大サイズは 1 GB です。

コネクタの使用

[コネクタ] タブより、クラウドデータベースやエンタープライズのサーバーに格納されているデータに接続できます。作成する各データ接続に接続情報を入力する必要があります。たとえば、ほとんどのデータ接続では、サーバー名とサインイン情報を提供する必要があります。

「サポートされるコネクタ」には、Tableau をこれらの各コネクタタイプに接続してデータソースを設定する方法についての情報が記載されています。[コネクタ] タブに必要なコネクタが表示されない場合、Tableau Desktop からデータに接続し、Web 作成用に Tableau Cloud や Tableau Server にデータソースをパブリッシュできます。Tableau Desktop で「データソースをパブリッシュする」方法の詳細をご覧ください。

Tableau でデータへの接続に成功すると[データソース] ページが開かれ、分析用データの準備を行ったり、ビューの構築を開始できます。詳細については、「Creators: Web 上のデータの準備」を参照してください。

Tableau Server コネクタ

- Actian Matrix*
- Alibaba AnalyticDB for MySQL‡
- Alibaba Data Lake Analytics‡
- Alibaba MaxCompute‡
- Amazon Athena‡
- Amazon Aurora for
- Google BigQuery**‡
- Google BigQuery JDBC**‡
- Google Cloud SQL‡
- Google ドライブ‡
- Impala‡
- Kognito*
- Kyvos‡
- OData‡
- OneDrive‡
- Oracle‡
- Pivotal Greenplum Database‡
- PostgreSQL‡
- Progress OpenEdge*
- Presto‡

MySQL‡

Amazon EMR Hadoop Hive‡

Amazon Redshift‡

Apache Drill‡

Aster Database*

Azure Data Lake Storage Gen2‡

Box‡

Cloudera Hadoop‡

Databricks‡

Datorama by Salesforce‡

Denodo‡

Dremio by Dremio‡

Dropbox‡

Esri コネクタ‡

Exasol‡

Hortonworks Hadoop Hive

IBM BigInsights

IBM DB2‡

IBM PDA (Netezza)*

Kyvos‡

MariaDB‡

MarkLogic*

Microsoft Azure SQL データベース‡

Microsoft Azure Synapse Analytics‡

Microsoft SQL Server‡

MonetDB*

MongoDB BI Connector‡

MySQL‡

Qubole Presto‡

SAP HANA (仮想接続のみ)‡

SAP Sybase ASE*

SAP Sybase IQ*

Salesforce‡

SharePoint Lists‡

SingleStore (旧 MemSQL)‡

Snowflake‡

Spark SQL‡

Teradata***‡

Vertica‡

* Linux サーバーでは利用できません。

** Google BigQuery では、Web からデータソースを作成するときに OAuth が必要です。サーバー管理者が Google 用に OAuth を設定する方法の詳細をご覧ください。

*** Teradata Web 作成では、現在クエリバンディング機能はサポートされていません。詳細については、「Teradata」を参照してください。

‡データ管理 をお持ちの場合は仮想接続をサポートします。詳細については、Tableau Server ヘルプの「仮想接続とデータポリシーについて」を参照してください。

Tableau Catalog サポートされるコネクタ

Tableau Catalog は、Tableau Server がサポートする一部のデータコネクタとの接続をサポートしています。データソース、データベース、ファイル、または表がグレー表示されている場合は、Tableau Server から接続することができません。ただし、適切なパーミッションがある場合は、Tableau Desktop の **[接続]** ペインから接続することができます。

Tableau Cloud

Tableau Cloud では、[このサイト上]、[ファイル]、[コネクタ]、[ダッシュボードスターター] のタブから選択してデータへの接続を行います。

[このサイト上] のデータへの接続

1. **[このサイト上]** は、パブリッシュされたデータソースの参照や検索を行う場合に選択します。
2. **[名前]** よりデータソースを選択し、**[接続]** ボタンをクリックします。

> **注:** データソースへの接続に加えてデータ管理がある場合は、**[このサイト上]** を選択すると仮想接続を使用してデータに接続できます。Tableau Catalog を有効にすると、データベース、ファイル、およびテーブルにも接続できます。

ファイルへの接続

Tableau はブラウザーからの Excel またはテキストベースのデータソース (.xlsx、.csv、.tsv) の直接アップロードをサポートしています。[データに接続] ペインの **[ファイル]** タブで、ファイルをフィールドにドラッグ アンド ドロップするか、**[コンピューターからアップロード]** をクリックして、Excel ファイルまたはテキストファイルに接続します。アップロードできるファイルの最大サイズは 1 GB です。

コネクタの使用

[コネクタ] タブより、クラウドデータベースやエンタープライズのサーバーに格納されているデータに接続できます。作成する各データ接続に接続情報を入力する必要があります。たとえば、ほとんどの

データ接続では、サーバー名とサインイン情報を提供する必要があります。

「サポートされるコネクタ」には、コネクタを使用してデータを Tableau に接続する方法についての情報が記載されています。[コネクタ] タブに必要なコネクタが表示されない場合、Tableau Desktop からデータに接続し、Web 作成用に Tableau Cloud や Tableau Server にデータソースをパブリッシュできます。Tableau Desktop で「データソースをパブリッシュする」方法の詳細をご覧ください。

注: Tableau Cloud のデータに接続できない場合は、データベースが公にアクセス可能かどうかを確認してください。Tableau Cloud は、パブリックインターネットからアクセス可能なデータにのみ接続できます。データがプライベートネットワークの背後にある場合、Tableau Bridge を使用して接続可能です。詳細については、「パブリッシャー: Tableau Bridge を使用して Tableau Cloud のデータを最新の状態に維持する」を参照してください。

Tableau Cloud コネクタ

- Alibaba AnalyticsDB for MySQL‡
- Alibaba Data Lake Analytics‡
- Amazon Athena‡
- Amazon Aurora for MySQL‡
- Amazon EMR Hadoop Hive‡
- Amazon Redshift‡
- Apache Drill‡
- Azure Data Lake Storage Gen2‡
- Azure Synapse Analytics (SQL Server 互換)
- Box‡
- Dropbox*‡
- Esri コネクタ‡
- Exasol‡
- Google BigQuery*‡
- Google Cloud SQL (MySQL 互換)‡§
- Google ドライブ‡
- Hortonworks Hadoop Hive
- Impala‡
- Kyvos‡
- MariaDB‡
- OData‡
- OneDrive*‡
- Oracle‡
- Pivotal Greenplum Database‡
- PostgreSQL‡
- Presto‡
- Qubole Presto‡
- Salesforce‡
- SAP HANA (仮想接続のみ)‡
- SharePoint Lists‡

Cloudera Hadoop‡

Databricks‡

Datorama by Salesforce‡

Denodo‡

Dremio by Dremio‡

Microsoft Azure SQL データベース‡

Microsoft Azure Synapse Analytics‡

Microsoft SQL Server‡

MongoDB BI Connector‡

MySQL‡

SingleStore (旧 MemSQL)‡

Snowflake‡

Spark SQL‡

Teradata**‡

Vertica‡

*Tableau Cloud で Google BigQuery、OneDrive、Dropbox 接続に OAuth 2.0 標準を使用する方法の詳細については、「OAuth 接続」を参照してください。

** Teradata Web 作成では、現在クエリバンディング機能はサポートされていません。詳細については、「Teradata」を参照してください。

‡データ管理 をお持ちの場合は仮想接続をサポートします。詳細については、Tableau Cloud ヘルプの「仮想接続とデータポリシーについて」を参照してください。

§Tableau Cloud は、Google Cloud SQL を使用した SSL をサポートしていません。

Tableau Catalog サポートされるコネクタ

Tableau Catalog は、Tableau Cloud がサポートする一部のデータコネクタとの接続をサポートしています。データソース、データベース、ファイル、または表がグレー表示されている場合は、Tableau Cloud から接続することができません。ただし、適切なパーミッションがある場合は、Tableau Desktop の **[接続]** ペインから接続することができます。

ダッシュボード スターターを使用する

Tableau Cloud でダッシュボード スターターを使用して LinkedIn Sales Navigator、Oracle Eloqua、Salesforce、ServiceNow ITSM、QuickBooks Online のデータを作成および分析することができます。**[ダッシュボード スターター]** タブの事前作成されたデザインのリストよりオプションを選択し、**[ダッシュボードを使用]** をクリックします。詳細については、「クラウドベースのデータ ソースのダッシュボード スターター」を参照してください。

Tableau Public

Tableau Public では、サポートされているファイルをアップロードしてデータに接続できます。

ファイルへの接続

Tableau はブラウザーからの Excel またはテキストベースのデータソース (.xlsx、.csv、.tsv) の直接アップロードをサポートしています。[データに接続] ペインの **[ファイル]** タブで、ファイルをフィールドにドラッグ アンド ドロップするか、**[コンピューターからアップロード]** をクリックして、Excel ファイルまたはテキスト ファイルに接続します。アップロードできるファイルの最大サイズは 1 GB です。

データセットがない場合は、Tableau Public Web サイトで、無料のサンプル データセットを確認してください。

コネクタの使用

[コネクタ] タブから、クラウド データベースに格納されているデータに接続できます。作成する各データ接続に接続情報を入力する必要があります。たとえば、ほとんどのデータ接続では、サインイン情報を入力する必要があります。

「サポートされるコネクタ」には、コネクタを使用してデータを Tableau に接続する方法についての情報が記載されています。[コネクタ] タブに必要なコネクタが表示されない場合は、Tableau Desktop からデータに接続し、データ抽出を作成できます。

> **注**: Tableau Public のデータに接続できない場合は、データベースが公にアクセス可能かどうかを確認してください。Tableau Public は、パブリック インターネットからアクセス可能なデータにのみ接続できます。

Tableau Public コネクタ

Google ドライブ

OData

接続後

Tableau でデータへの接続を行うと[データソース] ページが開かれ、分析用データの準備を行ったり、ビューの構築を開始できます。詳細については、「Creators: Web 上のデータの準備」を参照してください。

Web 作成でデータを最新に保つ

Tableau Cloud または Tableau Server のアップロード済みファイルを更新する: Web 作成用のファイル (Excel またはテキスト) を手動でアップロードする場合、Tableau ではそのファイルを自動的に更新できません。データを更新するには、[接続の編集] を選択して新しいバージョンのファイルをアップロードします。

Tableau Public で Vz に移動し、**[更新の要求]** をクリックします。Tableau Desktop Public Edition で [Keep this data in sync (このデータを同期)] を選択して、データを自動的に最新の状態に保つこともできます。

Tableau Cloud のファイルベースのパブリッシュ済みデータソースを更新する: ファイルベースのデータを使用するパブリッシュ済みのデータソース (Tableau Desktop よりパブリッシュ) が Tableau Cloud にある場合は、Tableau Bridge を使用して最新に保つことができます。詳細については、「Tableau Bridge を使用したデータの鮮度に関するオプションの拡張」を参照してください。

初期 SQL の実行

> **注**: Tableau Prep Builder バージョン 2019.2.2 以降では、初期 SQL の使用がサポートされていますが、Tableau Desktop でサポートされているのと同じオプションがすべてサポートされているわけではありません。Tableau Prep Builder での初期 SQL の使用の詳細については、Tableau Prep Builder Salesforce のヘルプの「初期 SQL を使用した接続のクエリ」を参照してください。

一部のデータベースに接続する場合、データベースへの接続時に実行またはキャッシュされた値を使用する初期 SQL コマンドを指定できます。データベースへの接続は、ワークブックを開いたとき、

抽出を更新したとき、Tableau Server へにサインインしたとき、Tableau Server にパブリッシュしたときなどに行われます。

> **注:** 初期 SQL はカスタム SQL 接続とは異なります。カスタム SQL 接続はクエリの発行対象となる関係 (表) を定義します。詳細については、「カスタム SQL クエリへの接続」を参照してください。

このコマンドを使用すると、次を実行できます。

- セッション中に使用する一時表の設定。
- カスタム データ環境の設定。

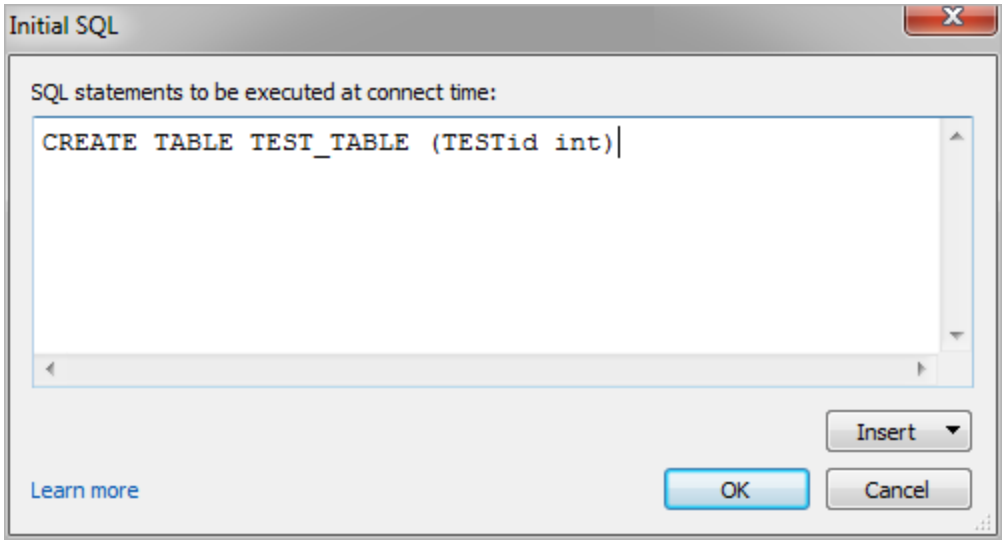
[サーバー接続] ダイアログ ボックスまたは [データ ソース] ページでは初期 SQL コマンドを追加するオプションを利用できます。

> **注:** データ ソースが初期 SQL ステートメントの実行をサポートしている場合、[サーバー接続] ダイアログ ボックスの左下隅に **[初期 SQL]** リンクが表示されます。データ ソースの詳細については、「サポートされるコネクタ」を参照してください。

初期 SQL を使用するには

1. [サーバー接続] ダイアログ ボックスで、**[初期 SQL]** をクリックします。または [データ ソース] ページで、接続先のデータベースに応じ、**[データ] > [初期 SQL]** か、**[データ] > [クエリ バンディングおよび初期 SQL]** を選択します。
2. [初期 SQL] ダイアログ ボックスに SQL コマンドを入力します。**[挿入]** ドロップダウン メニューを

使用すると、データソースにパラメーターを渡すことができます。



注: Tableau はステートメントのエラーを調べません。この SQL ステートメントは、接続時にデータベースに送信されます。

ソフトウェア ライセンスにより、接続への初期 SQL の使用が制限される場合があります。Tableau Server にパブリッシュする場合、初期 SQL ステートメントを許可するようにサーバーが設定されている必要があります。既定では、ワークブックを Web ブラウザーに読み込んだときにこれらのステートメントを実行できるようにサーバー ソフトウェアが設定されています。

管理者は、`tsm configuration set` コマンドを使用して、初期 SQL ステートメントを無視するようにサーバーを設定することができます。

```
tsm configuration set -k vizqlserver.initialsql.disabled -v true
```

サーバーで初期 SQL ステートメントが許可されていない場合、ワークブックは開きますが初期 SQL コマンドは送信されません。

`tsm configuration set` コマンドの詳細については、「Tableau Server ヘルプ」を参照してください。

初期 SQL ステートメントのパラメーター

初期 SQL ステートメントのデータ ソースにパラメーターを渡すことができます。次のリストに、初期 SQL ステートメントでパラメーターを使用するいくつかの利点を示します。

- **TableauServerUser** または **TableauServerUserFull** パラメーターを使用して偽装を設定できます。
- データ ソースでパラメーターがサポートされている場合は、ユーザーが表示権限を持つデータのみが表示されるように、行レベルのセキュリティ(たとえば、Oracle VPD や SAP Sybase ASE など) を設定できます。
- ログに、Tableau バージョンやワークブック名などの詳細を提供できます。

次のパラメーターは初期 SQL ステートメントでサポートされます。

パラメーター	説明	返される値の例
TableauServerUser	現在のサーバー ユーザーのユーザー名。サーバーで擬装を設定する際に使用します。ユーザーが Tableau Server にサインインしていない場合は、空の文字列を返します。	jsmith
TableauServerUserFull	現在のサーバー ユーザーのユーザー名およびドメイン。サーバーで擬装を設定する際に使用します。ユーザーが Tableau Server にサインインしていない場合は、空の文字列を返します。	domain.lan\jsmith
TableauApp	Tableau アプリケーションの名前。	Tableau Desktop Professional Tableau Server
TableauVersion	Tableau アプリケーションのバージョン。	9.3
WorkbookName	Tableau ワークブックの名前。埋め込み	財務分析

データ ソースが入 ったワークブックでのみ使用 します。

警告 : Tableau Desktop にはドメインが含 まれ ていません。委任 を使用 しておらず、tsm 設定「set -k DelegationUseFullDomainName=-v true--force-keys」を設定 している場合 は、これ を含 めることができます

次 の例 は、初期 SQL ステートメントでパラメーターを使用 できるさまざまな方法 を示 しています。

- この例 では、Microsoft SQL Server でセキュリティ コンテキストを設定 します。

  ```
  EXECUTE AS USER = [TableauServerUser] WITH NO REVERT;
  ```

- この例 では、DataStax データ ソースでパラメーターを使用 して、ログに詳細 を追加 したり、データを追跡 するためのセッション変数 を設定 したりする方法 を説明 しています。

  ```
  SET TABLEAUVERSION [TableauVersion];
  ```

- この例 を使用 すると、Oracle VPD に低 レベルのセキュリティを設定 する際 に役立 ちます。

  ```
  begin

   DBMS_SESSION.SET_IDENTIFIER([TableauServerUser]);

  end;
  ```

  注 : Oracle PL/SQL ブロックでは、ブロックを終了 するために、末尾 にセミコロンを付 ける必要 があります。正 しい構文 については、Oracle ドキュメントを参照 してください。

サーバーに対 する実行 の保留

初期 SQL ステートメントを遅延 させて、サーバー上 でのみ実行 することができます。サーバーへの実行 を保留 する理由 の 1 つに、偽装 を設定 したコマンドを実行 するパーミッションを持 っていない場合 があります。サーバー上 でのみ実行 されるコマンドを囲 うには <ServerOnly></ServerOnly> タグを使用 します。

例:

```
CREATE TEMP TABLE TempTable(x varchar(25));
INSERT INTO TempTable VALUES (1);
<ServerOnly>INSERT INTO TempTable Values(2);</ServerOnly>
```

セキュリティと偽装

初期 SQL ステートメントで **TableauServerUser** パラメーターまたは **TableauServerUserFull** パラメーターを使用すると、他のユーザーと共有できない専用接続が作成されます。これによりキャッシュの共有も制限でき、セキュリティが向上しますが、パフォーマンスが低下する可能性があります。

MySQL および Oracle 接続で「表を作成する」際のトラブルシューティング

MySQL 接続で、初期 SQL を使用してテーブルを作成してもテーブルが一覧表示されない

MySQL に接続して初期 SQL ステートメントを実行すると、Tableau でのクエリの構築方法が原因でテーブルが表示されない場合があります。

```
CREATE TABLE TestV1.testtable77(testID int);
```

この問題を解決するには、次のように `IF NOT EXISTS` を SQL ステートメントに追加します。

```
CREATE TABLE IF NOT EXISTS TestV1.TestTable(testID int);
```

Oracle 接続で、初期 SQL を使用してテーブルを作成すると Tableau が停止する

Oracle に接続して初期 SQL ステートメントを実行すると、Tableau でのクエリの構築方法が原因で Tableau が車輪のマークが回転した状態のまま停止します。

```
CREATE TABLE TEST_TABLE (TESTid int)
```

この問題を解決するには、次の SQL ステートメントを使用します。

```
BEGIN
EXECUTE IMMEDIATE 'create table test_table(testID int)';
EXCEPTION
WHEN OTHERS THEN NULL;
END;
```

Web 上でのフローの作成と操作

バージョン 2020.4 以降、Tableau Server でフローを作成および操作して、データのクリーンアップと準備を行うことができます。データへの接続、新しいフローの作成、および既存のフローの編集を行うことができ、作業は数秒ごとに自動的に保存されます。自分だけが利用できる下書きフローを作成するか、フローをパブリッシュして他のユーザーが利用できるようにします。データ管理がライセンスされている場合、Web から個別のフローを実行するか、Tableau Prep Conductor を使用してスケジュールに従ってフローを自動的に実行します。詳細については、「Web 上の Tableau Prep」を参照してください。

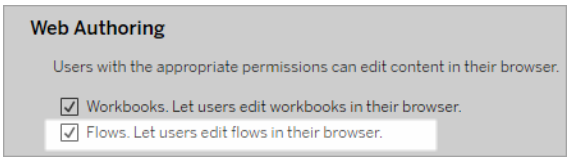
Tableau Server での Tableau Prep のフロー作成プロセスの設定の詳細については、Tableau Prep フロー作成を参照してください。

この記事で説明されているタスクを完了して、フローの Web 作成とその他のフロー機能をWeb 上で有効にします。

サイトでのフローの Web 作成をオンまたはオフにする

このオプションは既定で有効になっており、ユーザーが Tableau Server または Tableau Cloud のどちらでフローを作成および編集できるかどうかを制御します。

1. Web ブラウザーで管理者としてサーバーにサインインし、サイトに移動して **[設定]** をクリックします。
2. **[Web 作成]** セクションで、**[フロー]** をクリアまたは選択します。**ユーザーがブラウザーでフローを編集して**、機能をオフまたはオンにできるようにします。



3. 変更内容をすぐに有効にする場合は、サーバーを再起動します。再起動しない場合、変更内容はサーバー セッションのキャッシュ期限が切れた後、または次回ユーザーがサインアウト後に再びサインインしたときに有効になります。

リンクされたタスクを有効にする

Tableau Cloud および Tableau Server バージョン 2021.3 以降でサポートされています。

[リンクされたタスク] オプションを使用して、最大 20 件のフローを次々に実行するようにスケジュールできます。リンクされたタスクは、**[リンクされたタスク]** オプションが選択されているスケジュールでのみ実行できます。リンクされたタスクの詳細については、「リンクされたタスクのスケジュール」を参照してください。

バージョン 2022.1 以降、**リンクされたタスク**は、既定で **[サーバー設定]** および作成する新しいフロー スケジュールで有効になっています。以前のバージョンでは、管理者は最初にオプションを有効にする必要があります。

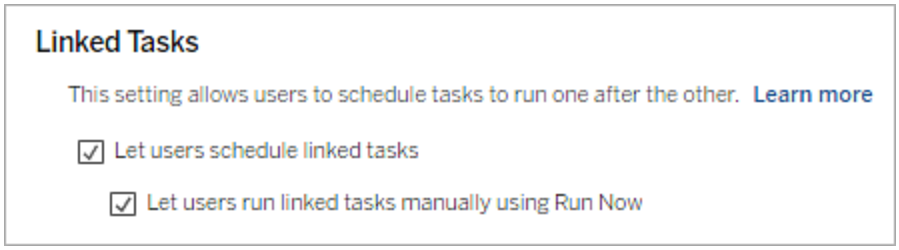
複数のサイトがある場合は、以下のチェック ボックスをオフにすることで、個々のサイトの **[リンクされたタスク]** をオフにできます。

リンクされたタスクをスケジュールした後にサーバー設定をオフにした場合、実行中のタスクはすべて完了しますが、スケジュール済みのリンクされたタスクは非表示になり、**[スケジュールされたタスク]** タブに表示されなくなります。

リンクされたタスクを有効にする (バージョン 2021.4 以前)

1. Web ブラウザーで管理者としてサーバーにサインインし、フローの **[リンクされたタスク]** を有効にするサイトに移動します。そのサイトで、**[設定]** をクリックします。
2. **[リンクされたタスク]** セクションで、**[リンクされたタスクをスケジュールできるようにする]** を選択して、管理者が、リンクされたタスクを実行するスケジュールを設定できるようにします。
3. **[リンクされたタスクを [今すぐ実行] を使用して手動で実行することを許可する]** を選択して、ユーザーが **[今すぐ実行]** を使用して、リンクされたフロー タスクを実行できるようにしま

す。



フロー パラメーターを有効にする

ユーザーがパラメーターを含むフローをスケジュールして実行できるようにします。パラメーターを使用すると、ユーザーはフローを一度構築してから、さまざまなデータシナリオに対応するようにパラメーター値を変更することで、フローをスケーリングできます。

パラメーターは、ファイル名、パス、テーブル名のインプット ステップ、またはカスタム SQL クエリを使用する場合は、ファイル名、パス、テーブル名のアウトプット ステップ、およびフィルターまたは計算値の任意のステップ タイプに入力できます。

Tableau Prep Builder および Tableau Cloud バージョン 2023.2 以降では、システム パラメーターをフロー出力名に追加して、フローの実行開始日時を自動的に含めることができます。

フロー パラメーターの設定は、サーバー レベルで適用して、Tableau Server 上のすべてのサイトを含めることができます。この設定は、サイト レベルで無効にして、特定のサイトのみを含めることができます。

フローでパラメーターを使用する方法の詳細については、Tableau Prep ヘルプの「フローでパラメーターを作成して使用する」を参照してください。

1. Web ブラウザーで管理者としてサーバーにサインインし、**[フロー パラメーター]** を有効にするサイトに移動します。そのサイトで、**[設定]** をクリックします。
2. **[フロー パラメーター]** セクションで、**[パラメーターを使用するフローをユーザーが実行およびスケジュールできるようにする]** を選択して、機能を有効にします。
3. (バージョン 2023.2 以降) **[Allow system generated parameters like timestamps to be applied to output names (タイムスタンプなどのシステム生成パラメーターを出力名に**

適用できるようにする)] を選択すると、ユーザーがファイルおよびパブリッシュされたデータソース出力タイプの実行時に、日付またはタイムスタンプをフロー出力名に追加できるようになります。

4. **[Allow parameters that can accept any input (任意の入力を受け入れるパラメーターを許可する)]** を選択して、フローを実行しているすべてのユーザーが実行時にフローに任意のパラメーター値を入力できるようにします。

   重要: このオプションを設定すると、フローのユーザーはパラメーターに任意の値を入力できるようになり、ユーザーがアクセスすべきでないデータが公開される可能性があります。

   このオプションが選択されていない場合、ユーザーは事前定義されたパラメーター値のリストのみから選択でき、任意の値を受け入れるパラメーターを含むフローは実行されないか、実行するようにスケジュールされなくなります。

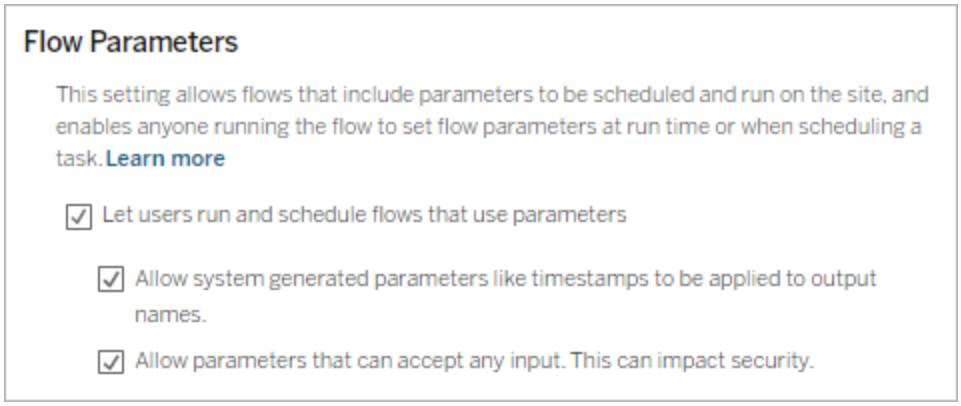


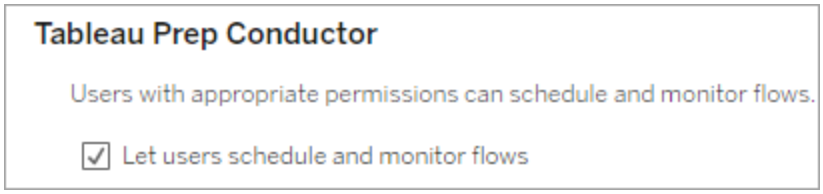
Tableau Prep Conductor を有効にする

データ管理がライセンスされている場合は、このオプションを有効にして、ユーザーが Tableau Server および Tableau Cloud でフローのスケジュール設定と追跡ができるようにします。Tableau Prep Conductor の追加の構成要件については、「Tableau Prep Conductor」参照してください。

1. Web ブラウザーで管理者としてサーバーにサインインし、Tableau Prep Conductor を有効にするサイトに移動します。そのサイトで、**[設定]** をクリックします。

2. **Tableau Prep Conductor** セクションで、**[Let users schedule and monitor flows (ユー**

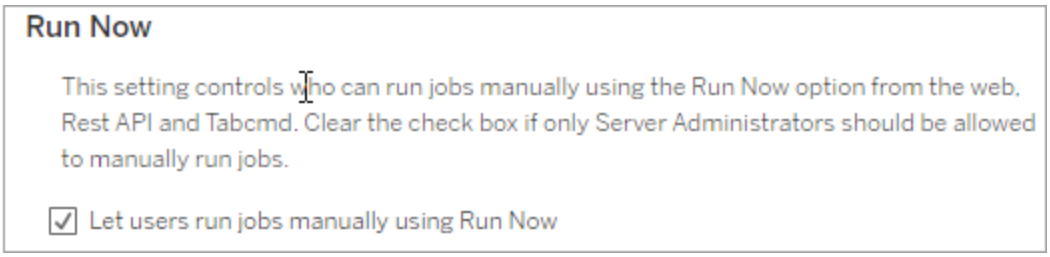
ザーにフローのスケジュール設定と監視を許可する)] を選択して、機能を有効にします。



[今すぐ実行] を有効にする

[今すぐ実行] オプションを使用して、ユーザーまたは管理者のみがフローを手動で実行できるかどうかを制御します。フローを手動で実行するには、データ管理 は必要ありません。

1. Web ブラウザーで管理者としてサーバーにサインインし、フローの [今すぐ実行] を有効にするサイトに移動します。そのサイトで、**[設定]** をクリックします。
2. **[今すぐ実行]** セクションで、**[ユーザーに [今すぐ実行] を使用してジョブを手動で実行するのを許可する]** を選択して、機能を有効にします。



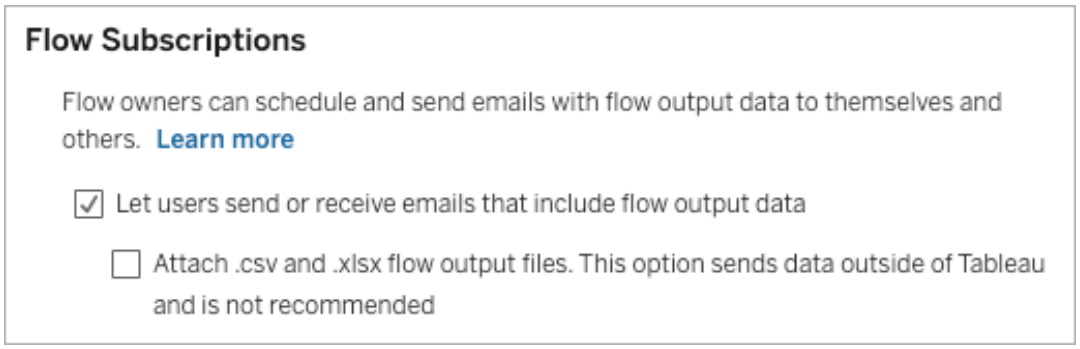
サーバー管理者のみがフローを手動で実行できる場合は、チェック ボックスをオフにします。

フロー サブスクリプション

スケジュールされたタスクでフローの実行が成功したことについて、ユーザーがフロー通知を受け取るかどうかを制御します。この通知を有効にするには、データ管理 が必要です。

1. Web ブラウザーで管理者としてサーバーにサインインし、フロー サブスクリプションを有効にするサイトに移動します。そのサイトで、**[設定]** をクリックします。

2. **[フロー サブスクリプション]** セクションで、**[Let users send or receive emails that include flow output data (ユーザーがフロー出力データが含まれるメールを送受信できるようにする)]** を選択して、機能を有効にします。



> **注**: .csv または .xlsx ファイル タイプのいずれかをメールに添付するオプションは、オンプレミス環境のみで使用できます。

Tableau Prep 拡張機能を有効にする

バージョン 2021.2.0 以降の Tableau Server および Tableau Cloud でサポートされています

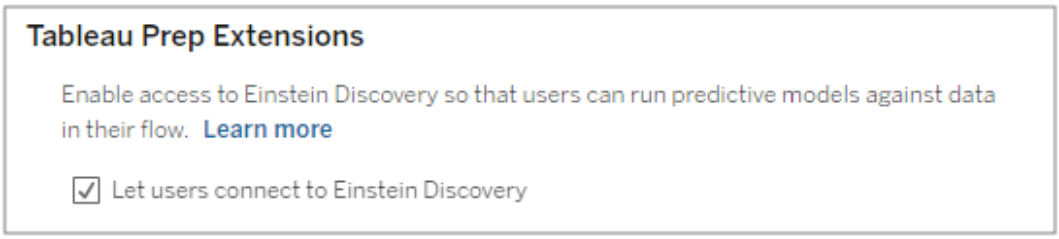
このオプションを設定して、ユーザーが Einstein Discovery に接続して、フロー内のデータに対して予測モデルを実行できるかどうかを制御します。

Einstein Discovery を活用したモデルを使用して、Web 上でフローを作成するときに、フロー内のデータの予測を一括でスコアリングできます。予測を行うことで、より適切な情報に基づいて意思決定を行い、ビジネス成果を向上させるためのアクションを実行できます。

フローに予測を含めるには、追加の設定を構成する必要があります。詳細については、「Einstein Discovery 予測をフローに追加する」および「Einstein Discovery 統合の設定」を参照してください。

1. Web ブラウザーで管理者としてサーバーにサインインし、Tableau Prep 拡張機能を有効にするサイトに移動します。そのサイトで、**[設定] > [Extensions (拡張機能)]** の順にクリックします。

2. **[Tableau Prep 拡張機能]** セクションで、**[Let users connect to Einstein Discovery (ユーザーに Einstein Discovery への接続を許可する)]** を選択して機能を有効にします。



自動保存をオフまたはオンにします

この機能は既定で有効になっており、ユーザーのフロー作業を数秒ごとに自動的に保存します。

管理者は、Tableau Server REST API メソッドの「サイトの更新」と `flowAutoSaveEnabled` 設定を使用して、サイトの自動保存を無効にすることができますが、これは推奨されていません。詳細については、「Tableau Server REST API サイトのメソッド: サイトの更新」を参照してください。Web 上での自動保存の詳細については、自動保存をオフまたはオンにしますを参照してください。

サイト設定の構成については、Tableau Server ヘルプの「サイト設定 リファレンス」を参照してください。

Web 上の Tableau Prep

Windows 上の Internet Explorer 11 および Internet Explorer の互換モードはサポートされていません。

バージョン 2020.4 以降、Tableau Prep ではフローの Web 作成がサポートされています。これにより、Tableau Prep Builder、Tableau Server、または Tableau Cloud を使用して、データのクリーニングと準備を行うフローを作成できます。また、Web 上でフローを手動で実行することができ、データ管理は必要ありません。

Web 上でも Tableau Prep Builder の同じ機能の大部分がサポートされていますが、フローを作成して操作する場合には、いくつかの相違点があります。

重要: Web 上でフローを作成および編集するには、Creator ライセンスが必要です。ただし、データ管理 は、Tableau Prep Conductor を使用するか API を経由して、スケジュールに従ってフローを実行する場合にのみ必要です。Tableau Prep Conductor を使用して構成および使用する方法については、Tableau Server または Tableau Cloud ヘルプの「Tableau Prep Conductor」を参照してください。

インストールと展開

ユーザーが Web 上でフローを作成および編集できるようにするには、サーバーでいくつかの設定を構成する必要があります。これらの各設定の詳細については、「Web 上でのフローの作成と操作」を参照してください。

- **Web 作成**: このオプションは既定で有効になっており、ユーザーが Tableau Server または Tableau Cloud のどちらでフローを作成および編集できるかどうかを制御します。
- **今すぐ実行**: **[今すぐ実行]** オプションを使用して、ユーザーまたは管理者のみが手動でフローを実行できるかどうかを制御します。Web 上でフローを手動で実行するには、データ管理 は必要はありません。
- **Tableau Prep Conductor**: データ管理 がライセンスされている場合、このオプションを有効にすると、ユーザーはフローをスケジュールおよび追跡できます。
- **Tableau Prep 拡張機能** (バージョン 2021.2.0 以降): ユーザーが Einstein Discovery に接続してフロー内のデータに対して予測モデルを適用および実行できるかどうかを制御します。
- **自動保存**: この機能は既定で有効になっており、数秒ごとにユーザーのフロー作業を自動的に保存します。

Tableau Server では、管理者は Tableau Prep のフロー作成プロセスの構成を微調整できます。詳細については、「Tableau Prep のフローの作成」を参照してください。

サンプル データと処理の制限

Web 上のフローを操作する際のパフォーマンスを維持するために、フローに含めることができるデータの量に制限が適用されます。

以下の制限が適用されます。

- ファイルに接続する場合、最大ファイル サイズは 1 GB です。
- すべてのデータを含めるデータサンプリング オプションは使用できません。既定のサンプル データの制限は 100 万行です。

データサンプルの設定については、Tableau Prep ヘルプの「データサンプルのサイズの設定」を参照してください。

Web 上で利用可能な機能

Web 上でフローを作成および編集すると、ナビゲーションおよび特定の機能の可用性に多少の相違点があることがわかります。ほとんどの機能はすべてのプラットフォームで利用できますが、一部の機能は Tableau Server または Tableau Cloud で制限が設けられているか、まだサポートされていません。次の表に相違点がある機能を示します。

メジャー エリア	例外	Tableau Prep Builder	Tableau Server	Tableau Cloud
データへの接続	一部のコネクタは、Web 上でサポートされていない場合があります。サーバーの **[接続]** ウィンドウを開いて、サポートされているコネクタを確認します。	✅	✅	✅
Tableau Agent を使用して Tableau Prep フローを構築する	バージョン 2025.2 以降の Tableau Prep Builder および Tableau Cloud でサポートされています。Tableau+ が必要です	✅	🚫	✅
フローの構築と編成		✅	✅	✅
データサンプルのサイズの設定	Tableau Server および Tableau Cloud では、データサンプルのサイズは管理者が設定した制限に従います	✅	✅	✅
インプット ステップのファイルおよびデータベース表のユニオン作成	インプット ユニオンは、Tableau Server や Tableau Cloud で編集または作成することはできません。Tableau Prep Builder のみ。	✅	✅	✅

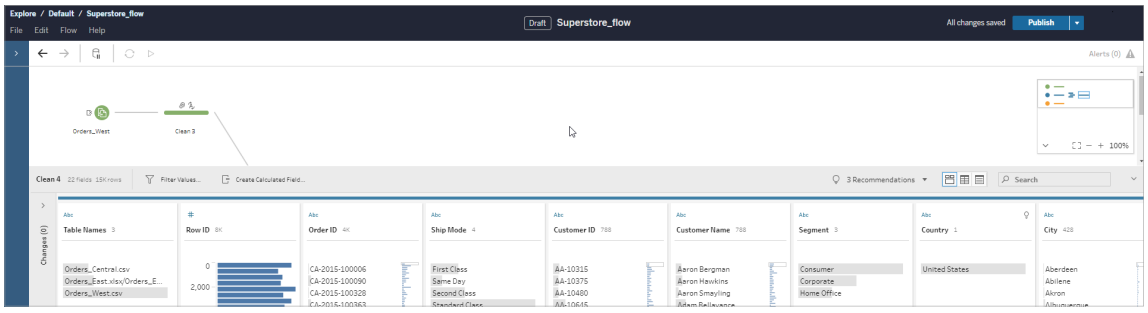
データのクリーニングと形成		✓	✓	✓
データグリッドの値のコピー	Tableau Prep Builder と Tableau Server ではバージョン 2022.3.から、Tableau Cloud では 2022.2 (8月) から利用可能です。	✓	✓	✓
データの集計、結合またはユニオン		✓	✓	✓
フローでの R および Python スクリプトの使用	Tableau Cloud でフローを作成または編集する場合、スクリプトステップを追加することはできません。これは現在、Tableau Prep Builder および Tableau Server でのみサポートされています。	✓	✓	⊘
再利用可能なフローステップの作成		✓	⊘	⊘
Web 上でフローを自動的に保存する		該当しません	✓	✓
ファイルの自動復元		✓	該当しません	該当しません
Tableau Desktop でフロー出力を表示する		✓	⊘	⊘
抽出ファイルを作成する		✓	⊘	⊘

Microsoft Excel ワークシートへの抽出を作成する		✓	✗	✗
カスタム SQL クエリへの接続		✓	✓	✓
パブリッシュされたデータソースを作成する		✓	✓	✓
フロー出力を外部データベースに保存する		✓	✓	✓
Einstein Discovery 予測をフローに追加する		✓	✓	✓

自動保存と下書き作業

サーバー上でフローを作成または編集すると、数秒ごとに作業が下書きとして保存されるため、クラッシュが発生した場合や、誤ってタブを閉じた場合でも、作業が失われることはありません。

下書きは、サインインしているサーバーとプロジェクトに保存されます。下書きを別のサーバーに保存またはパブリッシュすることはできませんが、**[ファイル] > [Publish As (名前を付けてパブリッシュ)]** メニュー オプションを使用すると、そのサーバー上の別のプロジェクトにフローを保存できます。



下書きのコンテンツは、パブリッシュするまで自分以外のユーザーには表示できません。変更内容をパブリッシュした後にそれを元に戻す必要がある場合は、**[リビジョン履歴]** ダイアログを使用して以前パブリッシュしたバージョンを表示し、元に戻すことができます。Web 上でのフローの保存の詳細については、「Web 上でフローを自動的に保存する」を参照してください。

Web上でフローをパブリッシュする

Web 上でフローをゼロから作成する場合でも、既存のフローを編集する場合でも、フローを実行する前にパブリッシュする必要があります。

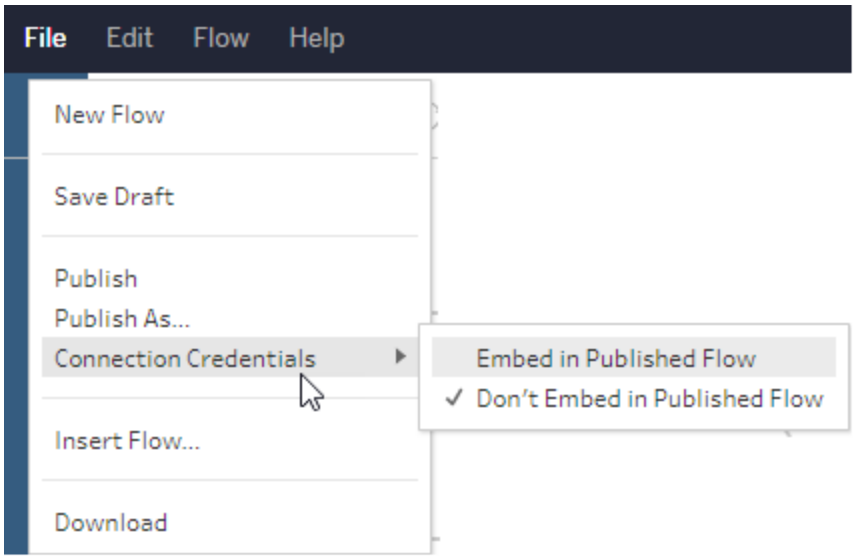
- フローの下書きはサインインするサーバーにのみパブリッシュできます。
- **[ファイル]** メニューの **[Publish As (名前を付けてパブリッシュ)]** をクリックすると、下書きを別のプロジェクトにパブリッシュできます。
- フローのデータベース接続に認証資格情報を埋め込むことで、フローの実行時に認証資格情報を手動で入力しなくても、フローを実行できます。フローを開いて編集する場合は、認証資格情報を再入力する必要があります。

認証資格情報の埋め込み

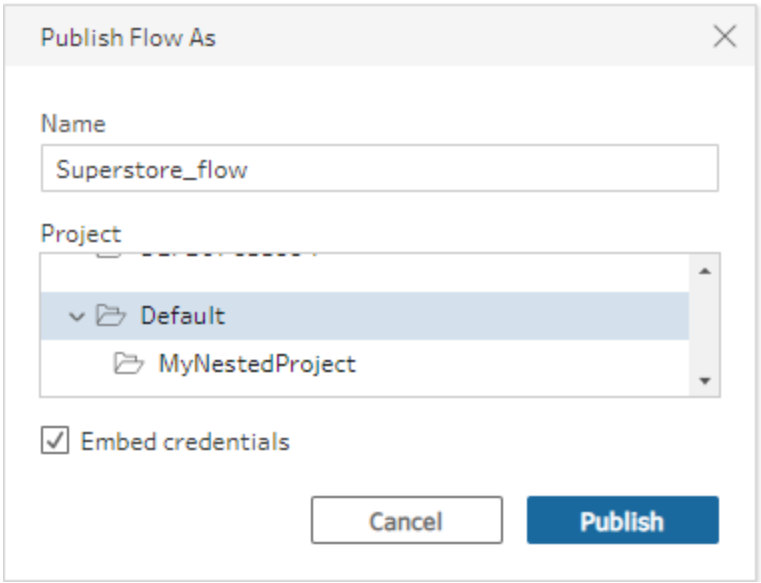
注: 認証資格情報の埋め込みは、サーバー上でフローを実行する場合にのみ適用されます。Tableau Cloud バージョン 2025.2 以降では、すべてのデータベース接続がフローに埋め込まれているか、ユーザープロフィールに保存されている場合、フロー所有者は、それらの接続を再認証する必要なしに、Web 作成でフローを編集できます。Tableau Server を使用している場合、データベースに接続されているフローを編集する際に、認証資格情報を手動で入力する必要があります。認証資格情報の埋め込みは、サーバーレベルまたはサイトレベルではなく、フローレベルでのみ設定できます。

次のいずれかを実行します。

- トップ メニューから、**[ファイル] > [接続の認証資格情報] > [Embeded in Published Flow (パブリッシュされたフローへの埋め込み)]** の順に選択します。



- フローをパブリッシュする場合は、**[認証資格情報の埋め込み]** チェックボックスをオンにします。このオプションは、**[Publish As (名前を付けてパブリッシュ)]** を選択して新しいプロジェクトに初めてフローをパブリッシュしたとき、または他のユーザーが最後にパブリッシュしたフローを編集するときに表示されます。

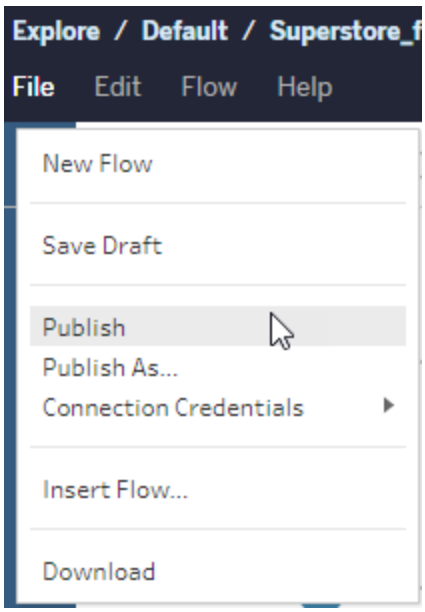


フローのパブリッシュ

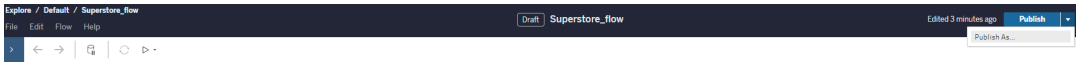
フローをパブリッシュすると、それが現在のバージョンのフローになり、プロジェクトにアクセスできる他のユーザーが実行および表示できます。パブリッシュされていないフローやフローの下書きに対して行った変更内容は、フローをパブリッシュするまで自分以外のユーザーには表示できません。フローのステータスの詳細については、「Web 上でフローを自動的に保存する」を参照してください。

フローをパブリッシュするには、次のいずれかを実行します。

- トップメニューから、**[ファイル] > [パブリッシュ]** または **[ファイル] > [Publish As (名前を付けてパブリッシュ)]** を選択します。



- 上部のバーで **[パブリッシュ]** ボタンをクリックするか、ドロップダウン矢印をクリックして **[Publish As (名前を付けてパブリッシュ)]** を選択します。



実行可能なユーザー

- サーバー管理者、サイト管理者 Creator、および Creator は、完全な接続およびパブリッシュアクセスを許可します。
- Creator は、Web 作成タスクを実行できます。

Web 上でのビューの作成とデータの探索

ビューを Tableau Server で作成して操作できます。詳細については、Tableau ヘルプで以下のユーザー向けトピックを参照してください。

Web での Tableau の使用

Tableau サイトの概要

Web でのビューの編集

データの結合

ダッシュボードの作成

ストーリーの作成

Web ページでビューおよびダッシュボードを埋め込む

バージョン間でワークブックの互換性があるようにする

アラートとサブスクリプション

サブスクリプションのトラブルシューティング

Tableau Cloud または Tableau Server からのデータ主導アラートの送信

データ接続のために保存された認証資格情報の管理

保存された認証資格情報は、認証資格情報の入力を要求されずにデータソースに接続できるようにします。接続のために保存された認証資格情報には OAuth アクセストークン、またはユーザー名とパスワードなどのその他の認証資格情報があります。保存された認証資格情報は [アカウント設定] ページで管理できます。

Tableau Server では、サーバー管理者によって認証資格情報の保存が許可されている場合、**[アカウント設定]** ページの **[保存された認証資格情報]** セクションで管理できます。**[保存された**

認証資格情報] セクションが表示されない場合は、認証資格情報の保存の許可について管理者にお問い合わせください。詳細については、「保存済みアクセストークンの許可」を参照してください。

> **注**: Web 上で Tableau Prep フローを編集する場合、再認証を求められる場合があります。

保存された認証資格情報を使用した接続のテスト

コネクターがテスト機能をサポートする場合、接続を保存された認証資格情報でテストできます。

1. Tableau Server または Tableau Cloud にサインインし、[アカウント設定] ページを表示します。
2. [保存済み認証資格情報] セクションで、テストしたい保存済み接続の横にある **[テスト]** リンクを選択します。

このテストにより Tableau Cloud または Tableau Server がこの対応する保存された認証資格情報を使用してアカウントにアクセスできることを確認できます。テテストが成功したにもかかわらず、この接続管理を通じてデータにアクセスできない場合は、入力した認証資格情報でデータにアクセスできることを確認してください。

たとえば、誤って個人の Gmail アカウントを使用して接続を作成している場合や、異なるアカウントを使用して Google Analytics データベースにアクセスしている場合は、保存済み認証資格情報を削除して、適切な Gmail アカウントを使用してサインインする必要があります。

保存済み認証資格情報の更新

サイトにカスタム OAuth クライアントを構成した後は、既存の Tableau コンテンツからのデータアクセスが中断されないようにするために、保存済み認証資格情報を更新することをお勧めします。保存済み資格情報を更新するには、以前保存された特定のコネクタの認証資格情報を削除してから、もう一度追加します。

保存済み資格情報をもう一度追加すると、新しい Tableau コンテンツと既存の Tableau コンテンツの両方で、サーバー管理者によって構成されたカスタム OAuth クライアントを使用してデータにア

クセスできるようになります。カスタム OAuth クライアントの詳細については、「サイトのカスタム OAuth の設定」を参照してください。

1. Tableau Server にサインインし、**[マイ アカウントの設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。
    1. コネクタの保存済み認証資格情報の横にある **[削除]** をクリックします。
    2. 同じコネクタの横にある **[追加]** をクリックし、プロンプトに従って 1) サイト管理者から通知されたカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

保存されたすべての認証資格情報のクリア

[保存されたすべての認証資格情報の削除] を選択すると、ユーザー アカウントから以下の項目が削除されます。

- アカウントに保存されているすべての接続用の認証資格情報。

  > **注意:** これらの保存された認証資格情報のいずれかがパブリッシュ済みワークブックやデータ ソースで保存されている場合、これらのトークンを削除するとデータ ソースへのアクセスもそれらの場所から削除されます。事実上、これはまるで影響を受ける保存された認証資格情報が使用されるあらゆる場所で「鍵を変える」ようなものです。

- パブリッシュされたデータ抽出やそれらに接続するワークブックへのアクセスに使用しているパスワード。

保存済み認証資格情報の削除

データへの Tableau アクセスを削除するには、アカウントからそのデータへの接続に関連する保存された認証資格情報を削除します。認証資格情報を削除したら、次回アクセスする際にデータにサインインする必要があります。これにより、保存済み認証資格情報が新たに作成されます。

管理者は、すべてのユーザーが同じ共有認証資格情報を使用してデータ ソースに接続するよう選択できます。この場合、保存された認証資格情報はすべてのユーザーのデータ接続に関連付

けられます。また、認証資格情報は [アカウント設定] ページの [保存済み認証資格情報] には表示されません。

> **注:** Tableau Server ユーザーであるものの、保存済み認証資格情報を削除できない場合は、サーバー設定で [ユーザーにデータソース アクセス トークンの保存を許可] オプションをクリアしたかどうかについて、管理者に問い合わせてください。

パーソナル スペースでプライベート コンテンツを作成および編集

パーソナル スペースは、すべての Explorer および Creator が Tableau サイトで作業するときにコンテンツを保存するためのプライベートな場所です。パーソナル スペースに保存されたコンテンツは、他のユーザーと共有することはできませんが、他のユーザーに見せる準備ができたらプロジェクトに移動できます。パーソナル スペース内で、新しいワークブックを作成したり、ワークブックを個別のコピーとしてパーソナル スペースに保存したりできます。自分が所有する既存のコンテンツをパーソナル スペースに移動して編集し、後でプロジェクトに戻すこともできます。Explorer は、ワークブックに含まれるすべてのデータを含めて、ワークブックをパーソナル スペースにダウンロードできます。

パーソナル スペースのプライバシー

パーソナル スペースに保存されたコンテンツは、自分とサイト管理者だけが閲覧できます。サイト管理者は、ユーザーのパーソナル スペースに直接アクセスしたり、誰かのパーソナル スペースのコンテンツを編集したりすることはできませんが、パーソナル スペースのワークブックを表示および管理することができます。パーソナル スペースのワークブックは、管理者の検索結果に表示されるほか、[Explore] ページでワークブックの場所として表示されます。また、ワークブックはプライベートであるため、ワークブックがパーソナル スペースにある場合、[パーミッション] メニューは使用できません。

Tableau Catalog とパーソナル スペース

> 2019.3 以降、Tableau Catalog は、Tableau Server と Tableau Cloud に対するデータ管理で使用できます。詳細については、Tableau Server または Tableau Cloud ヘルプの「Tableau Catalog について」を参照してください。

作業環境で Tableau Catalog が有効になっている場合、パーソナル スペースに保存したワークブックに関する情報は Catalog によってインデックス化されます。これらのワークブックは系列カウントに含まれますが、ワークブックを表示できるのは自分だけです。さらに、系統ツールを閲覧するユーザーには、パーソナル スペース内のワークブックに関する情報の代わりに、必要なパーミッションが表示されます。

コラボレーション ツール

ワークブックがパーソナル スペースにある場合、共有、メトリクス、コメント、アラート、サブスクリプションなど、一部の機能が無効になります。自分への既存のアラートとサブスクリプションは引き続き実行されますが、コンテンツがプライベートになっているため、他のユーザーへのアラートとサブスクリプションは失敗します。メトリクスは、パーソナル スペースでは作成できませんが、接続されたワークブックがパーソナル スペースに移動されれば引き続き機能します。(従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。)

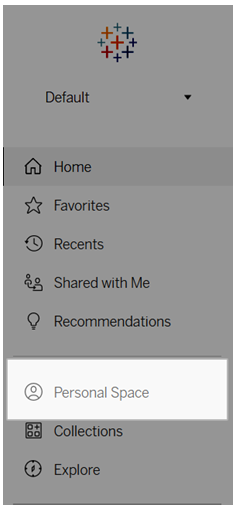
これらの制限は、ワークブックを別の場所に移動または保存すると取り除かれます。たとえば、ワークブックにコメントが含まれていてパーソナル スペースに移動した場合、既存のコメントは非表示になります。ワークブックを別の場所に移動すると、コメントが復元されます。

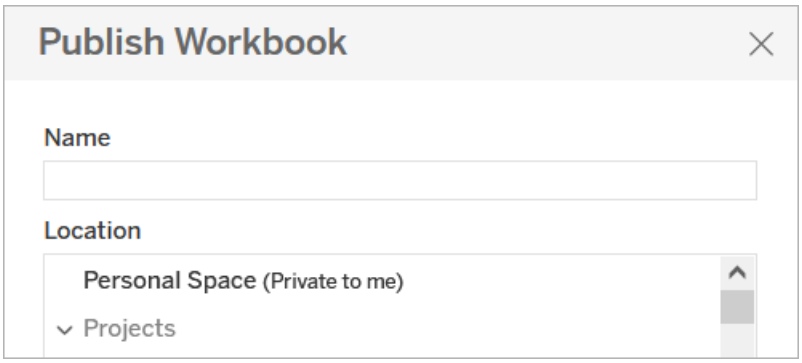
パーソナル スペースでの抽出更新

リソースの消費を制限するために、既存の抽出更新は、スケジュールされている場合は引き続き実行されますが、ワークブックがパーソナル スペースにある間は、新しい抽出更新をスケジュールすることはできません。

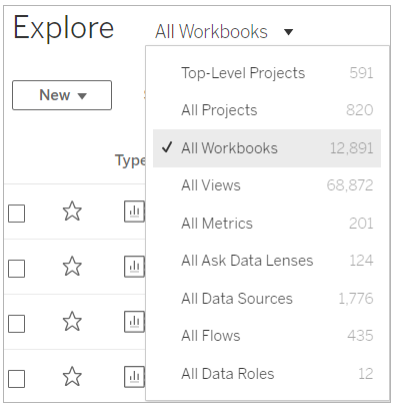
パーソナル スペースのコンテンツを検索

左側のナビゲーション メニューからパーソナル スペースにアクセスすると、パーソナル スペースのすべてのコンテンツを表示したり、新しいワークブックを作成したりできます。また、サイトの任意の場所で作成または編集したワークブックを、パーソナル スペースに保存することもできます。





[すべてのワークブック] を選択すると、[探索] ページからパーソナル スペースのワークブックを表示することができ、フィルターでパーソナル スペースのコンテンツに絞り込むこともできます。



パーソナル スペースへのワークブックのパブリッシュ

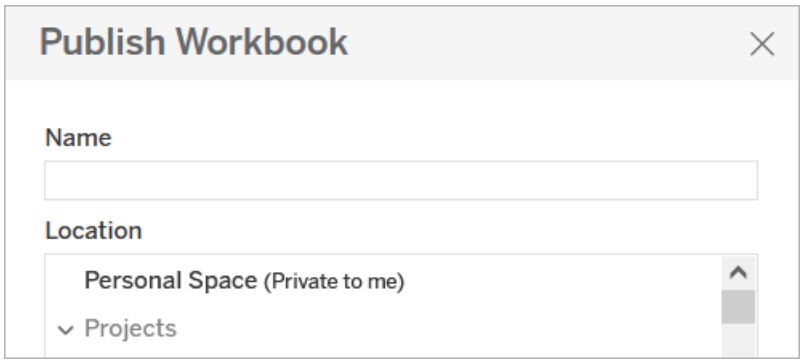
パーソナル スペースは、Tableau Cloud、Tableau Server、Tableau Desktop から新規または既存のワークブックをパブリッシュする、プライベートプロジェクトのように機能します。

パーソナル スペースが有効な場合、どのプロジェクトで新しいワークブックが作成されるかに関係なく、下書きはパーソナル スペースに自動的に保存されます。

パーソナル スペースが*有効になっていない*場合、新しいワークブックの下書きはプロジェクト内に保存されます。ただし、最上位プロジェクトで新しいワークブックが作成された場合、下書きはどこにも保存されません。

Tableau Server または Tableau Cloud のパーソナル スペースへのワークブックのパブリッシュ

1. ワークブックを開いた状態で、**[ファイル]**、**[名前を付けてパブリッシュ]** の順に選択します。
2. [場所] の下で、**[パーソナル スペース]** を選択します。



注: Explorer はワークブックをパーソナル スペースに保存できるだけなので、場所の選択ダイアログが表示されない場合があります。

Tableau Desktop からパーソナル スペースへのワークブックのパブリッシュ

2023.1 以降、ワークブックを Tableau Desktop からパーソナル スペースにパブリッシュできます。

1. パブリッシュするワークブックを Tableau Desktop で開いた状態で、**[サーバー]**、**[ワークブックのパブリッシュ]** の順に選択します。
2. [プロジェクト] で、**[パーソナル スペース]** を選択します。
3. [データ ソース] で、**[編集]** を選択します。
4. [パブリッシュ タイプ] の [データ ソースの管理] ポップアップで、すべてのデータ ソースについて **[ワークブックに埋め込まれている]** を選択します。パーソナル スペースにデータ ソースを個別にパブリッシュすることはできないため、Tableau Desktop からパブリッシュする場合はデータ ソースを埋め込む必要があります。
5. 通常どおり、パブリッシュ オプションの残りの部分を入力します。詳細については、「ワークブックをパブリッシュするための包括的な手順」を参照してください。

ワークブックをパーソナル スペースに移動

ワークブックの所有者であり、パーソナル スペースに空きがある場合は、既存のワークブックをパーソナル スペースに移動できます。パーソナル スペースのストレージ制限は、管理者によって設定されます。

ワークブックをパーソナル スペースに移動するには、次の手順を実行します。

- ワークブックを選択し、**[アクション]** ドロップダウン メニューをクリックします。
- **[移動]** を選択します。
- [場所] の下で、**[パーソナル スペース]** を選択します。

> **注:** Explorer はワークブックをパーソナル スペースに保存できるだけなので、移動アクションまたは場所の選択ダイアログが表示されない場合があります。

詳細については、「Web コンテンツの管理」ヘルプ トピックの「アクションの実行」を参照してください。

既存のワークブックまたはデータ ソースをパーソナル スペースに移動すると、共有、アラート、サブスクリプションなどのツールが非表示になります。既存の抽出更新は、スケジュールされている場合は

引き続き実行されますが、ユーザーはパーソナル スペース内で新しい抽出更新をスケジュールすることはできません。

既存のサブスクリプションとアラートも継続されますが、パーソナル スペースから編集することはできず、他のユーザーが受信者である場合は失敗します。既存の接続済みメトリクスは引き続き更新されますが、接続済みビューは他のユーザーに表示されません。

パーソナル スペースからワークブックを移動

パーソナル スペースからワークブックを移動すると、共有、アラート、サブスクリプションなどのコラボレーション ツールが表示され、既存のコメントが再び表示されます。

複数テーブルのデータ分析に関係を使用する

このキャンバスにドラッグしたテーブルでは、リレーッションシップを使用します。リレーッションシップは、Tableau で複数テーブルを分析するためにデータを組み合わせる柔軟な方法です。

関係は、2 つのテーブル間の契約と考えることができます。これらのテーブルのフィールドを使って Viz を構築する場合、Tableau は、その契約を使用してこれらのテーブルからデータを取り込み、適切な結合を使用してクエリを作成します。

関係によってデータの準備と分析がより簡単かつ直感的に行えるようになるため、データを結合する際の最初のアプローチとして関係を使用することをお勧めします。結合は、必要不可欠な場合にのみ使用してください。この 5 分間のビデオで、リレーッションシップを作成するための基本について学んでください。

> **注**: このビデオで示されている関係を編集するためのインターフェースは、現在のリリースとは少し異なりますが、同じ機能を備えています。

リレーションシップのしくみの詳細については、Tableau ブログ投稿を参照してください。

- 関係 (パート1): Tableau に新しいデータモデリングを導入する
- 関係 (パート2): ヒントとテクニック
- 関係 (パート3): 複数の関連するテーブルを横断して質問する

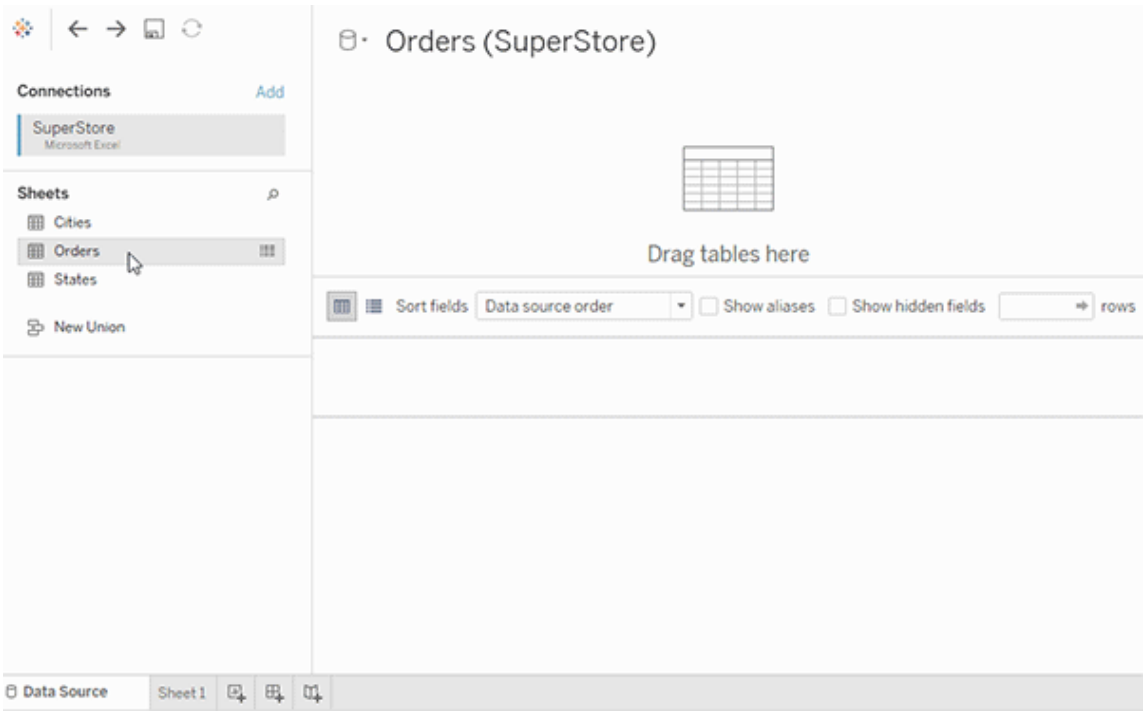
また、「Tableau が関係を発明した理由」など、Action Analytics からの関係に関するビデオ ポッドキャストも参照してください。詳細については、ライブラリの [ビデオ ポッドキャスト] をクリックしてください。

Tableau バージョン 2024.2 以降では、Tableau データモデルは、マルチファクト関係を通じてマルチファクト分析と共有ディメンションをサポートします。詳細については、「マルチファクト関係データモデルについて」、「マルチファクト関係モデルを使用するとき」、「マルチファクト関係データモデルの構築」を参照してください。

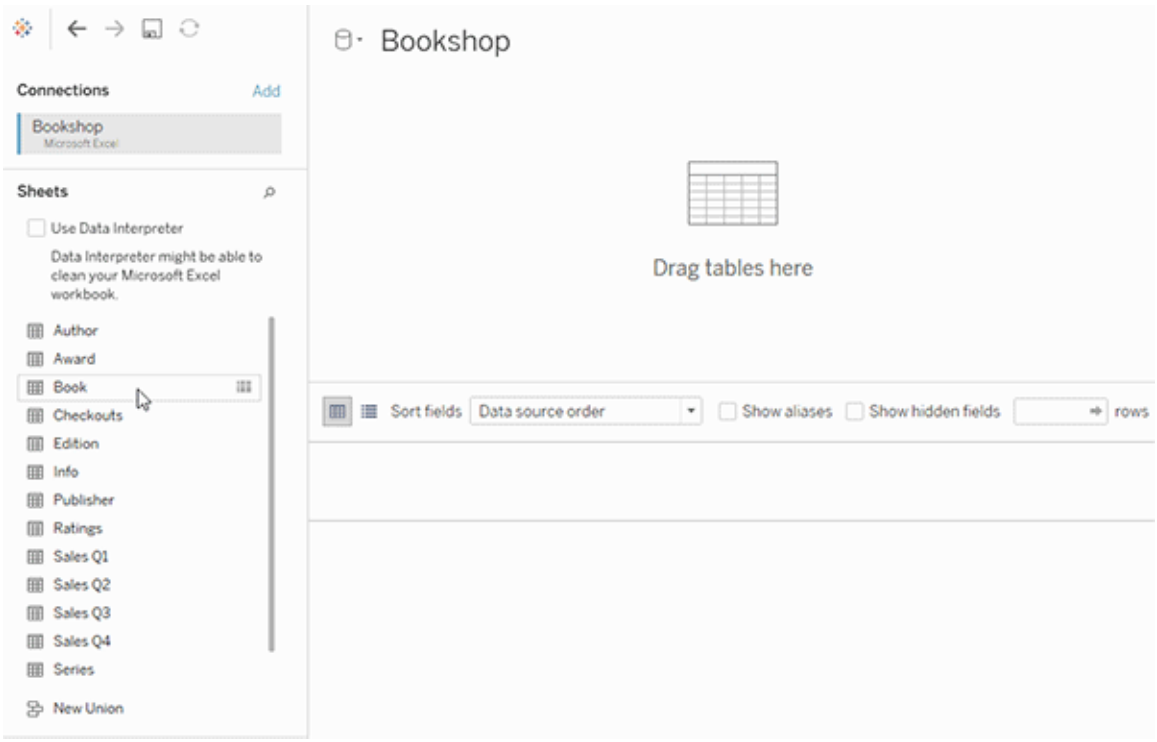
新しいデータソースとワークブックを作成していますか?

データソースの構築を開始するには、テーブルを [データソース] ページ キャンバスにドラッグします。

データソースで、分析に必要なすべてのディメンション フィールドとメジャー フィールドを含む単一のテーブルを作成することができます。



また、複数テーブルのデータソースを作成するには、さらに多くのテーブルをドラッグして、それらのリレーションシップを定義します。



リレーッションシップの使用を開始する方法については、この 1 分間のビデオをご覧ください。

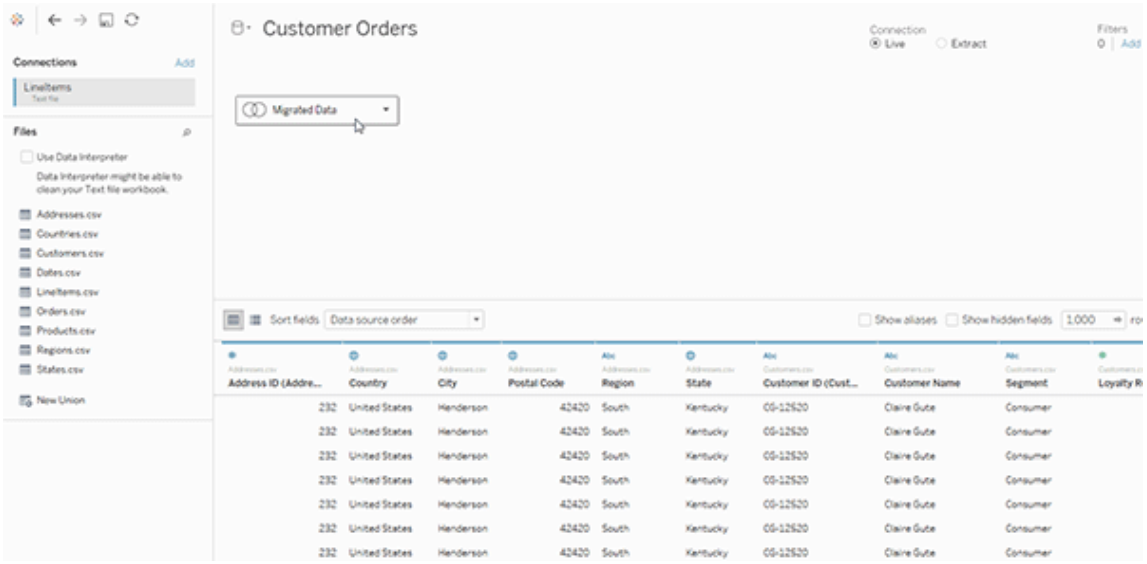
注: このビデオで示されている関係を編集するためのインターフェースは、現在のリリースとは少し異なりますが、同じ機能を備えています。

関係の使用に関する詳細については、「データの関連付け」、「関係と結合の違い」、「Tableau データモデル」、「関係の作成と定義」を参照してください。

Tableau 2020.2 でのデータソースと分析で変更された詳細については、「2020.2 のデータソースと分析で変更された内容」と「2020.2 の関係、データモデル、データソースに関する質問」を参照してください。

古いワークブックやデータソースを開いていますか?

2020.2 より前のワークブックやデータソースを 2020.2 で開くと、データソースがキャンバス内で単一の論理テーブルとして表示され、「Migrated Data (移行済みのデータ)」という名前か元のテーブル名が付けられます。データは保持され、ワークブックはこれまでどおり使用できます。



単一の論理テーブルを構成する物理テーブルを表示するには、その論理テーブルをダブルクリックして、物理レイヤーで開きます。結合とユニオンを含む、基となる物理テーブルが表示されます。

Tableau 2020.2 でのデータソースと分析で変更された詳細については、「2020.2 のデータソースと分析で変更された内容」と「2020.2 の関係、データモデル、データソースに関する質問」を参照してください。

Tableau データモデル

Tableau で作成したすべてのデータソースには、データモデルがあります。データモデルとは、接続したデータベース テーブルでデータをどのようにクエリするかを Tableau に伝える図のようなものです。

[データ ソース] ページでキャンバスに追加したテーブルによって、そのデータモデルの構造が作成されます。データモデルには、単一テーブルのように単純なものもあります。また、複数のテーブルを

含む複雑なモデルには、リレーションシップ、結合、ユニオンなどのさまざまな組み合わせが使用されています。

データモデルには、次の 2 つのレイヤーがあります。

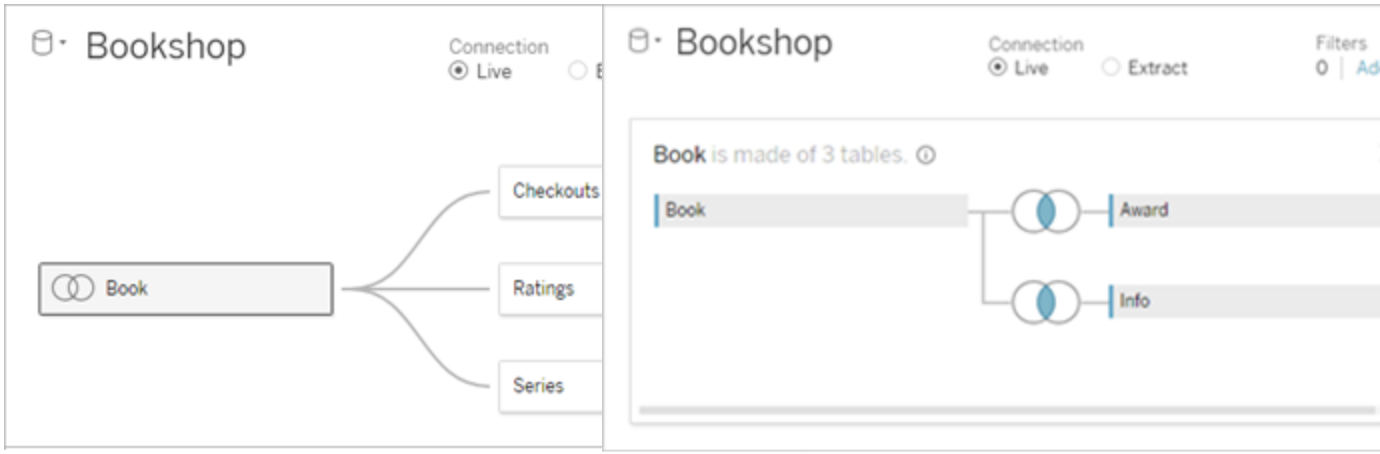
- [データソース] ページのキャンバスで最初に表示される既定のビューは、そのデータソースの*論理*レイヤーです。論理レイヤーでは、リレーションシップ (ヌードル) を使用してデータを組み合わせます。このレイヤーは、[データソース] ページのリレーションシップのキャンバスと考えてください。詳細については、「複数テーブルのデータ分析に関係を使用する」を参照してください。
- 次のレイヤーは物理レイヤーです。物理レイヤーでは、結合とユニオンを使用してテーブル間でデータを組み合わせます。このレイヤーでは、それぞれの論理テーブルに少なくとも 1 つの物理テーブルが含まれています。物理レイヤーは、[データソース] ページの結合/ユニオンのキャンバスと考えてください。論理テーブルをダブルクリックして、結合とユニオンを表示または追加します。

論理レイヤー

ヌードル = リレーションシップ

物理レイヤー

ベン図 = 結合



複数の関連するテーブルがあるデータソースの最上位ビュー。これが論理レイヤーになります。論理テーブルは、リレーションシップ (ヌードル) を使用して組み合わせることができます。結合タイプは使用されません。論理テーブルは物理テーブルのコンテナの

論理テーブルをダブルクリックで開き、物理テーブルを表示します。物理テーブルは、結合またはユニオンを使用して組み合わせることができます。この例では、書籍の論理テーブルが、結合された 3 つの物理テーブル (書籍、賞、情報) で構成されています。

論理レイヤー	物理レイヤー
ように機能します。	

論理レイヤー	物理レイヤー
[データソース] ページのリレーションシップのキャンバス	[データソース] ページの結合/ユニオンのキャンバス
ここにドラッグしたテーブルは論理テーブルと呼ばれます	ここにドラッグしたテーブルは物理テーブルと呼ばれます
論理テーブルは他の論理テーブルに関連付けることができます	物理テーブルは、他の物理テーブルに結合またはユニオンすることができます
論理テーブルは物理テーブルのコンテナのように機能します	論理テーブルをダブルクリックすると、その物理テーブルが表示されます。
詳細レベルは、論理テーブルの行レベルにあります。	詳細レベルは、マージされた物理テーブルの行レベルにあります。
論理テーブルは、データソースにマージされず、行の重複が削除された(正規化)状態のままになります。	物理テーブルは、論理テーブルを定義する単一のフラットなテーブルにマージされます。

データモデルのレイヤー

データソースで表示されるトップレベルビューは、データモデルの論理レイヤーです。このレイヤーでは、結合ではなくリレーションシップを使用してテーブルを組み合わせるため、リレーションシップのキャンバスとみなすこともできます。

複数のテーブルのデータを組み合わせる場合、論理レイヤーのキャンバスにドラッグするテーブルには別のテーブルとのリレーションシップが存在する必要があります。関係の結合タイプを指定する必要はありません。Tableau では、ワークシート内の分析のフィールドとコンテキストに基づいて、適切な結合タイプが自動的に選択されます。

データモデルの物理レイヤーでは、結合とユニオンを使用してデータを組み合わせることができます。このキャンバスではピボットのみを使用できます。このレイヤーは、結合/ユニオンのキャンバスと考えることができます。以前のバージョンの Tableau では、データモデル内で物理レイヤーが唯一のレイヤーでした。それぞれの論理テーブルには、1 つまたは複数の*物理テーブル*を含めることができます。

重要: Tableau では引き続き、結合とユニオンを使用する単一テーブルのデータソースを作成することができます。Tableau で単一テーブルの分析を行う動作は変更されていません。アップグレードしたワークブックは、2020.2 以前と同じように動作します。

詳細情報: 関係を使用したデータの組み合わせに関する関連情報については、次のトピックとブログ投稿も参照してください。

- 関係と結合の違い
- 複数テーブルのデータ分析に関係を使用する
- データの関連付け
- 関係 (パート 1): Tableau に新しいデータモデリングを導入する
- 関係 (パート 2): ヒントとテクニック
- 関係 (パート 3): 複数の関連するテーブルを横断して質問する

また、「Tableau が関係を発明した理由」など、Action Analytics からの関係に関するビデオポッドキャストも参照してください。詳細については、ライブラリの [ビデオポッドキャスト] をクリックしてください。

[データモデル] について

Tableau の以前のバージョン (2020.2 より前) では、データは物理レイヤーのみをモデル化していました。Tableau 2020.2 以降では、データモデルには論理レイヤーと物理レイヤーがあります。これにより、分析に合わせてスキーマを使用してデータを結合するためのオプションが増えます。

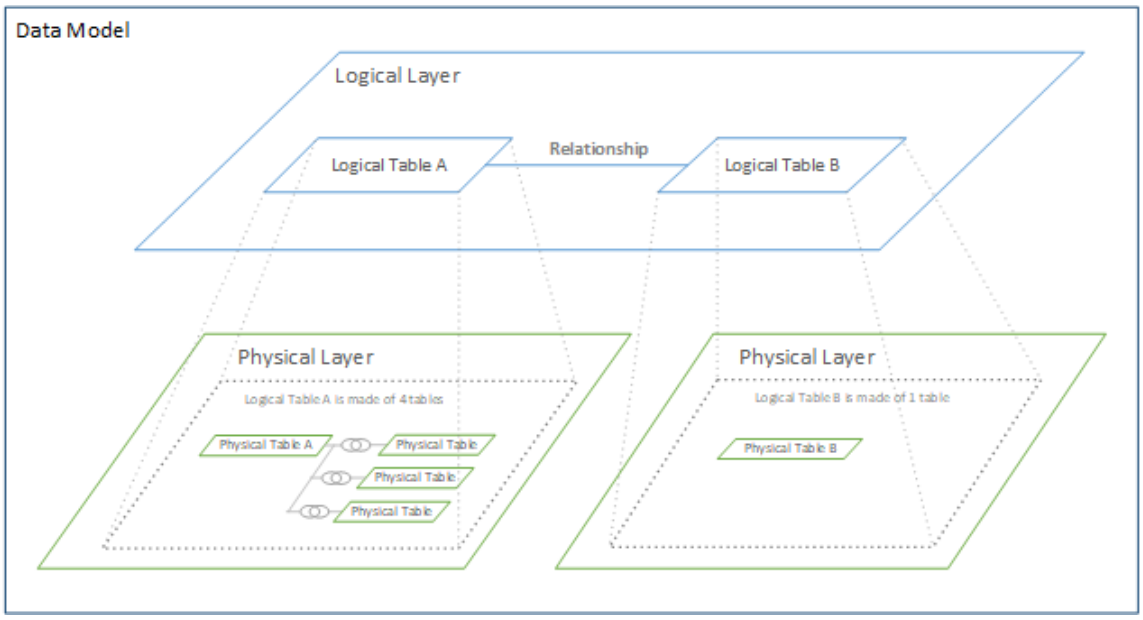
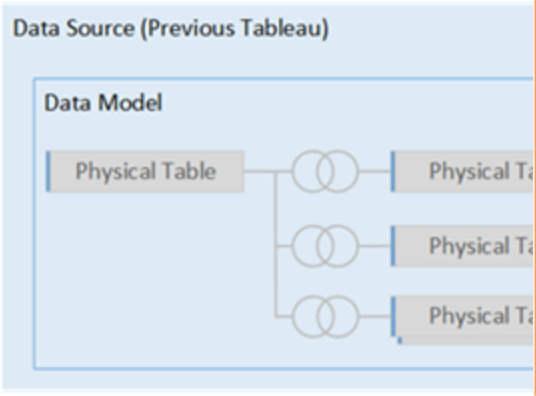


Tableau 2020.2 以降では、データソースに論理レイヤーが追加されました。論理テーブルには、物理レイヤーの中に物理テーブルが含まれています。

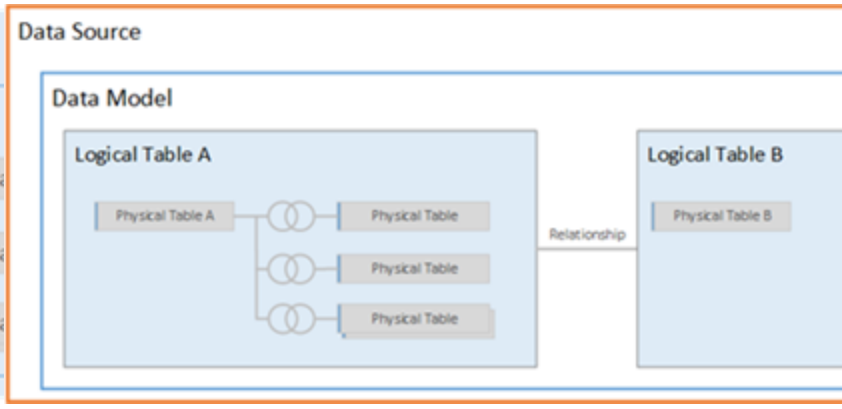
Tableau の以前のバージョン (2020.2 より前) では、データソースにあるデータモデルは、結合とユニオンを指定できる単一の物理レイヤーで構成されていました。結合やユニオンを使用して物理レイヤーに追加されたテーブルによって、分析に使用できる単一の平坦化された (非正規化済みの) テーブルが作成されていました。

以前のバージョン



2020.2 より前のバージョンの Tableau

2020.2 以降



2020.2 以降では、データモデルには論理レイ

以前のバージョン	2020.2 以降
では、データモデルには物理レイヤーだけが含まれています。	ヤーと物理レイヤーの 2 つのレイヤーがあります。

Tableau 2020.2 以降、データソースのデータモデルには、物理レイヤーの上に新しいセマンティックレイヤー(論理レイヤー)が含まれ、複数のテーブルを追加して相互に関連付けることができます。論理レイヤーにあるテーブルは、データソースにマージされず、独立した(正規化された)状態のままで、ネイティブの詳細レベルが維持されます。

論理テーブルは、マージされた物理テーブルのコンテナーのように機能します。論理テーブルには、単一の物理テーブルを含めることができます。または、結合またはユニオンを使用してマージされた複数の物理テーブルを含めることができます。

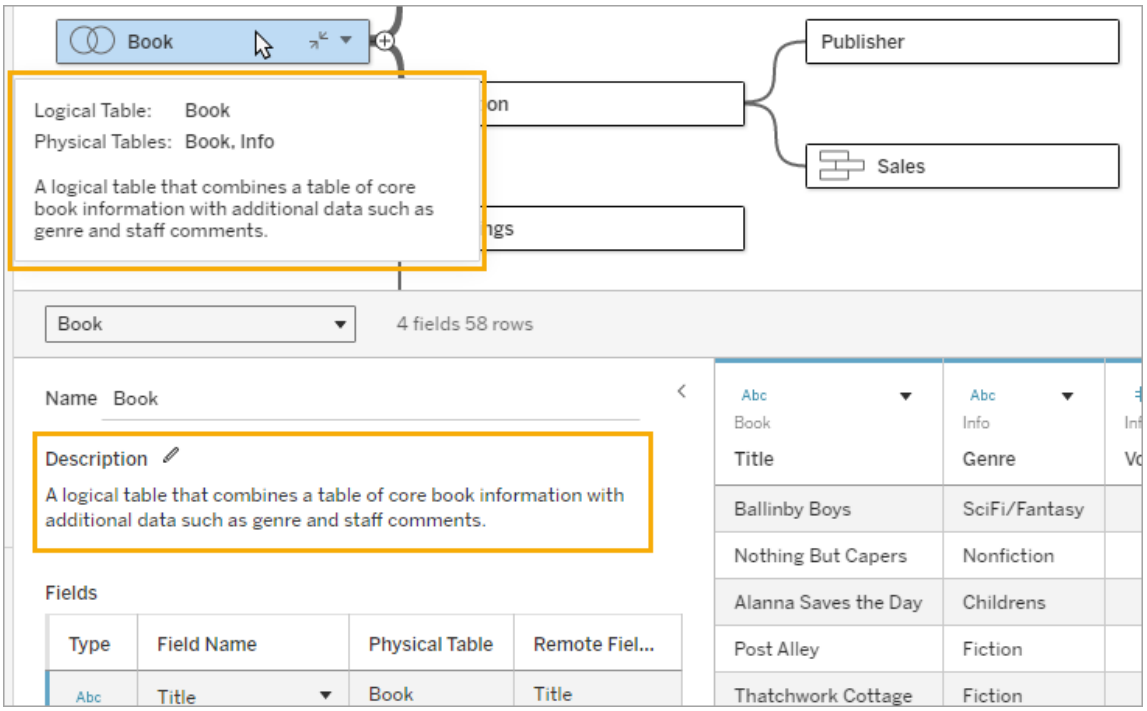
論理テーブルへの説明の追加

論理テーブルには、ツールヒントに表示される説明を含めることができます。これらのツールヒントは、[データソース] タブにのみ表示されます。

説明を追加するには、次の操作を行います。

1. キャンバス内のテーブルを選択する
2. 下のペインで **[説明]** の横にある編集アイコン(鉛筆)をクリックする
3. リッチテキストエディターに説明を入力する

注: Web 作成の場合のみ、説明内のリンクは生きていて、クリックして直接開くことができます。



新しいモデルの構築

基本的に、論理レイヤーに 1 つまたは複数のテーブルを追加すると、データソースのデータモデルを構築できます。単一の論理テーブルからデータソースを作成するか、複数のテーブルをキャンバスにドラッグして複雑なモデルを作成できます。

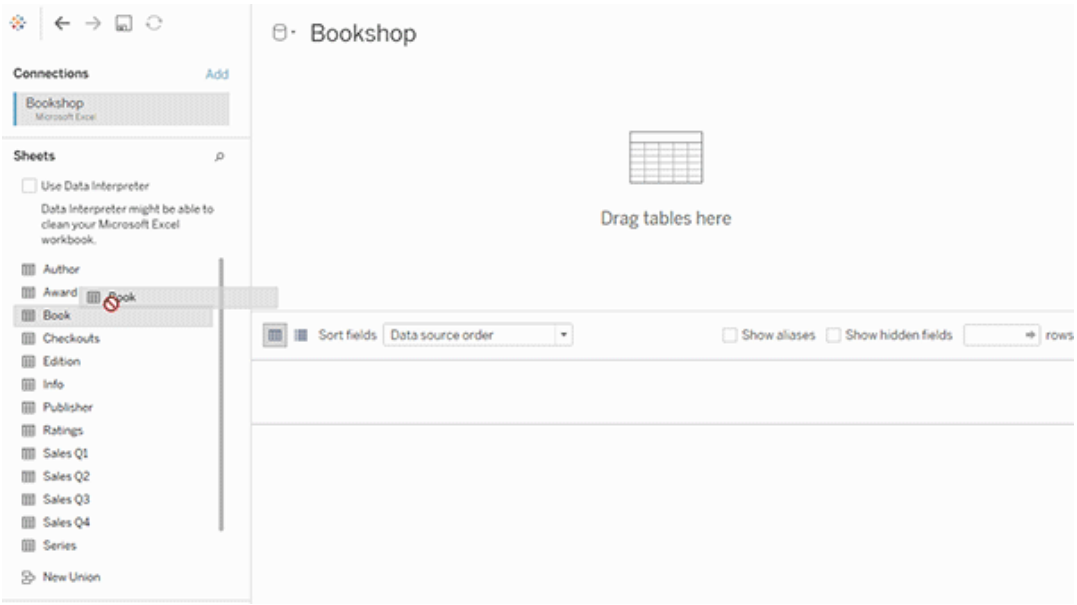
- キャンバスにドラッグした最初のテーブルは、データソース内のデータモデルのルートテーブルになります。
- ルートテーブルをドラッグした後、任意の順序でさらに別のテーブルをドラッグできます。相互に関連付ける必要があるテーブルに加えて、各関係に対して定義する、一致したフィールドペアを考慮する必要があります。
- スター スキーマを作成する場合は、最初にファクトテーブルをドラッグしてから、ディメンションテーブルをそのテーブルに関連付けると便利です。
- キャンバス内のテーブルを削除すると、関連する子孫も自動的に削除されます。ルートテーブルを削除すると、モデル内の他のすべてのテーブルも削除されます。
- 各関係は、少なくとも 1 つの一致するフィールドペアで構成する必要があります。複数のフィールドペアを追加すると、複合リレーションシップを作成できます。一致するペアは同じデータ型である必要があります。[データソース] ページでデータ型を変更しても、この要件は変更されません。Tableau では、クエリの参照元データベースのデータ型が引き続き使用されます。

- リレーションシップは計算フィールドに基づきます。
- リレーションシップを定義する際は、演算子を使用して、リレーションシップで使用するフィールドの比較方法を指定できます。

関係の詳細については、「データの関連付け」の「関係の作成と定義」を参照してください。

複数テーブルのモデル

- 複数テーブルのモデルを作成するには、テーブルを[データソース] ページのキャンバスの論理レイヤーにドラッグします。



[データソース] ページのキャンバスの論理レイヤーにドラッグするテーブルは、相互に関連している必要があります。追加のテーブルを論理レイヤーのキャンバスにドラッグすると、Tableau では、既存のキー制約と一致するフィールドに基づいて、リレーションシップを自動的に作成し、そのリレーションシップの定義を試みます。一致するフィールドを判別できない場合は、ユーザーがそれらのフィールドを選択する必要があります。

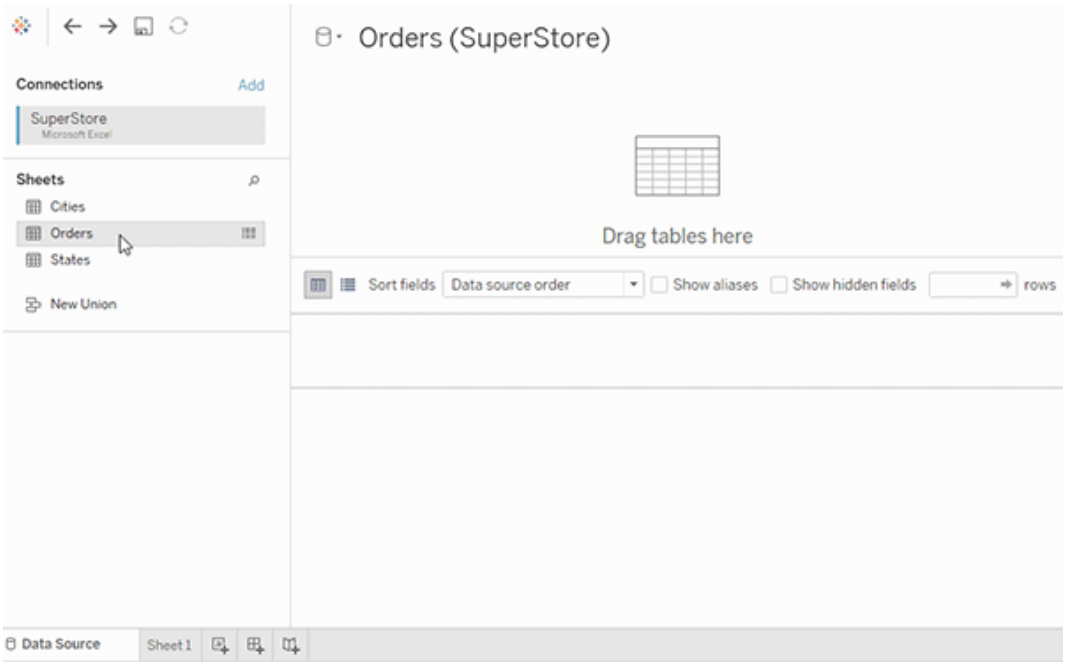
制約が検出されない場合は、**多対多**のリレーションシップが作成され、参照整合性が **[Some records match (一部のレコードが一致)]** に設定されます。これらの既定の設定は安全な選択肢であり、データソースに対して最大の柔軟性を提供します。既定の設定では完全外部結合に対応しており、分析する際はテーブルデータを集計してから結合を形成するとクエリを最適化できます。分析には、各テーブルのすべての列データと行データを使用できます。

テーブルをダブルクリックすると、任意の論理テーブル内にデータを追加できます。これにより、[データソース] ページのキャンバスの物理レイヤーが開きます。結合やユニオンを使用する必要がある場合は、結合したりユニオンしたりするテーブルを物理レイヤーキャンバスにドラッグできます。物理テーブルは論理テーブルにマージされます。

複数のテーブルを組み合わせるには、「リレーションシップの作成と定義」の手順に従ってください。

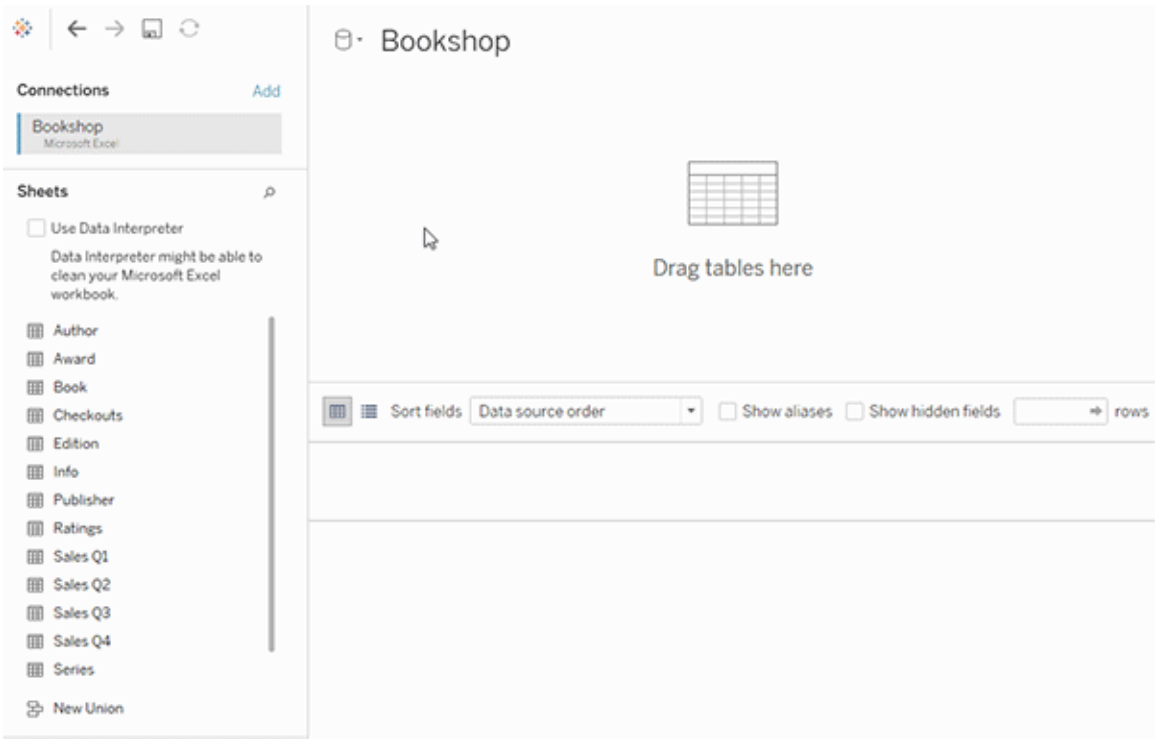
単一テーブルのモデル

- 単一テーブルのモデルを作成するには、[データソース] ページの論理レイヤーのキャンバスにテーブルをドラッグします。その後、[データ] ペインのテーブルのフィールドを分析に使用できます。



他のテーブルを含む単一テーブルのモデル

テーブルをダブルクリックすると、単一の論理テーブル内にデータを追加できます。これにより、[データソース] ページのキャンバスの物理レイヤーが開きます。結合やユニオンを使用する必要がある場合は、結合したりユニオンしたりするテーブルを物理レイヤーキャンバスにドラッグできます。物理テーブルは論理テーブルにマージされます。



この例では、データ ソースの関係キャンバス (論理レイヤー) に [Book] テーブルが表示されます。[Book] 論理テーブルをダブルクリックすると、結合/ユニオン キャンバス (物理レイヤー) が開きます。

この例では、結合によって賞と情報のテーブルが書籍のテーブルにマージされます。この場合、賞の詳細レベルでは、書籍と賞との結合は一対多になります。これにより、書籍と情報のメジャー バリューが重複します。重複を避けるため、書籍の論理テーブル内で賞と情報を結合する代わりに、それらを書籍に関連付けることができます。

対応しているデータ モデルのスキーマ

バージョン 2020.2 で導入された Tableau のデータ モデリング機能は、スター型やスノーフレーク型のデータ モデルを含む、一般的な複数テーブルのデータ シナリオの分析を容易に行えるように設計されています。Tableau のデータ ソースでは、次のモデル タイプが対応しています。

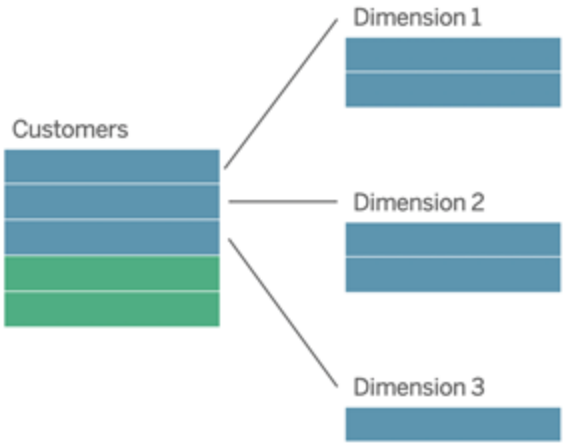
単一テーブル

ディメンションとメジャーの混合を含む単一の論理テーブル上の分析は、Tableau 2020.2 以前と同様に機能します。結合、ユニオン、カスタム SQL などの組み合わせを使用して論理テーブルを作

成できます。

スター スキーマとスノーフレーク スキーマ

エンタープライズ データ ウェアハウスでは、メジャーが中央のファクト テーブルに含まれ、ディメンションが独立したディメンション テーブルに個別に格納される、スター スキーマまたはスノーフレーク スキーマでデータが構造化されるのが一般的です。このデータの編成では、ロールアップやドリルダウンなど、多くの一般的な分析フローがサポートされます。



これらのモデルは、Tableau 2020.2 以降で利用できるデータ モデリング機能で、関係を使用して直接表現できます。

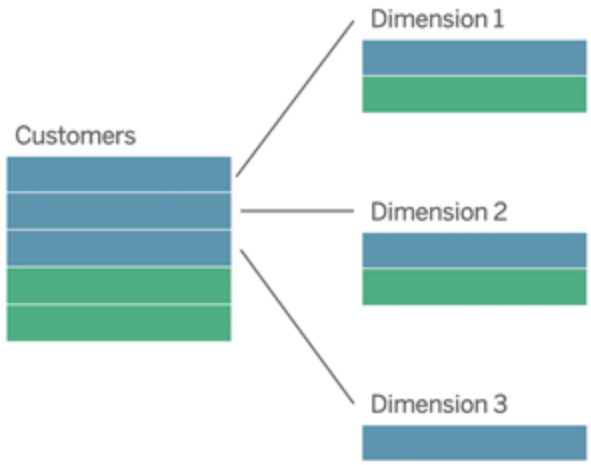
まず、ファクト テーブルをモデルにドラッグしてから、ディメンション テーブルをファクト テーブル (スター スキーマ内) または他のディメンション テーブル (スノーフレーク内) に関連付けます。

通常、適切にモデルされたスター スキーマまたはスノーフレーク スキーマでは、ファクト テーブルとディメンション テーブルの間のリレーションシップは多対 1 になります。この情報がデータ ウェアハウスにエンコードされている場合、Tableau は自動的にこれを使用してリレーションシップのパフォーマンス オプションを設定します。設定されていない場合は、この情報を自分で設定できます。詳細については、「パフォーマンス オプションを使用して関係のクエリを最適化する」を参照してください。

適切にモデル化したスター スキーマやスノーフレーク スキーマでは、ファクト テーブルのすべての行に、各ディメンション テーブルの一致するエントリが含まれます。これが true でデータ ウェアハウスの整合性制約にキャプチャされた場合、Tableau は自動的にこの情報を使用して、[パフォーマンス オプション] の参照整合性設定を設定します。一部のファクト テーブル行がディメンション テーブル内に一致する行を持たない場合 ("遅延到着ディメンション" または "早期到着ファクト" とも呼ばれます) は、メジャーの計算時にすべての行が保持されますが、ディメンション ヘッダーを表示するときに値が削除される可能性があります。詳細については、「パフォーマンス オプションを使用して関係のクエリを最適化する」を参照してください。

複数のテーブルでメジャー含むスタースキーマとスノーフレーク スキーマ

一部のスター スキーマまたはスノーフレーク スキーマでは、分析のすべてのメジャーがファクト テーブルに含まれています。ただし、多くの場合、分析のディメンション テーブルに関連する追加のメジャーが関係している可能性があります。ディメンション テーブルにメジャーが含まれていない場合でも、分析でディメンションの値をカウントしたり集計するのが一般的です。このような場合は、ファクト テーブルとディメンション テーブルはあまり明確に区別されていません。データ モデルをわかりやすく表示するには、まず最も細かいテーブルをデータ ソースのキャンバスに追加してから、そのテーブルに他のすべてのテーブルを関連付けることをお勧めします。



これらのテーブルを結合して 1 つの論理テーブルにする場合、LOD 計算または COUNT DISTINCT を使用して値を重複しないようにする予防措置を講じなければ、ディメンション テーブル内のメジャーがレプリケートされ、集計が歪んでしまいます。ただし、これらのテーブル間にリレーションシップを作成する場合、Tableau は、結合を実行する前にメジャーを集計し、不要な重複の問題を回避します。これにより、測定の詳細レベルを慎重に追跡する必要が解消されます。

マルチファクト分析

バージョン 2024.2 以降、Tableau のデータモデリング機能は、マルチファクト関係を使用したマルチファクト分析をサポートします。マルチファクト関係データモデルを作成する方法の詳細については、以下を参照してください。

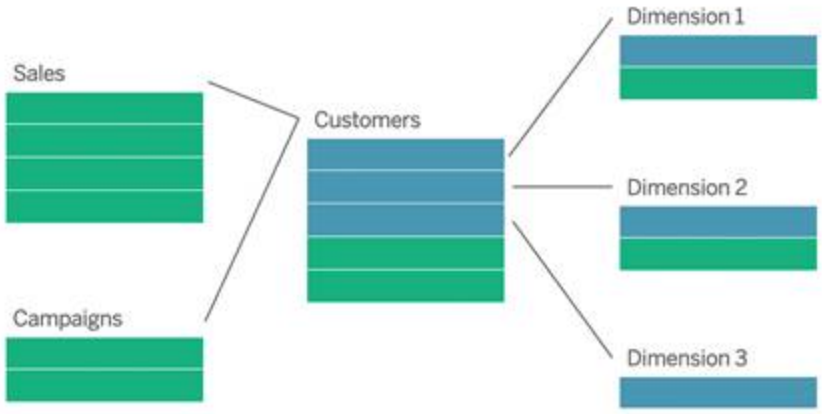
- マルチファクト関係データモデルについて
- マルチファクト関係モデルを使用するとき
- マルチファクト関係データモデルの構築

マルチファクト関係モデル (複数の基底テーブルを持つデータモデル) では、共有テーブルがモデル内に存在するとき、関連のないテーブルがモデル内に存在することも許されます。分析の間、共有テーブルのフィールドは、共通に持つ共有ディメンション (同じ場所や同じ日付に発生するなど) に基づいて、関連のないデータテーブルを「つなぎ合わせ」ます。各テーブルの粒度、つまりネイティブの詳細レベルの保持を含め、関係のすべての利点が維持されます。

基底テーブルが 1 つしかないデータモデルと同様に、Tableau は、Viz の構造に基づいて、バックグラウンドで使用する最適な結合タイプを決定します。ただし、マルチファクト関係モデルでは、結合オプションを拡張して外部結合とクロス結合を含むようになり、さまざまなレベルの関係を処理できます。詳細については、「マルチファクト関係データモデルについて」を参照してください。

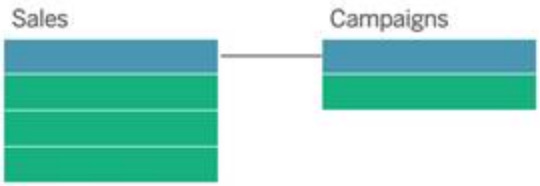
注: バージョン 2020.2 から 2024.1 では、ファクトテーブル (メジャーを含む) が単一のディメンションテーブルに関連付けられている場合にのみ、スターモデルとスノーフレークモデルに追加できます。

たとえば、Customer 360 などの分析のように、複数のファクトテーブル テーブルまとめてから共有ディメンションを分析できます。これらのファクトテーブルは、ディメンション テーブルとは異なる詳細レベルにしたり、それぞれを異なる詳細レベルにすることができます。また、ディメンション テーブルと多対多のリレーションシップを持つことができます。このようなシナリオの場合、Tableau では集計の前に値がレプリケートされないようにします。



ファクトテーブルを関連付ける共有ディメンション テーブルがない場合は、カスタム SQL を使用するか、他のディメンション テーブルの結合またはユニオンを使用して、ディメンション テーブルを動的に作成できる場合があります。

ファクトテーブルが 2 つある場合は、共通ディメンション上で相互に直接関連付けることができます。このタイプの分析は、ファクトテーブルの 1 つに共通ディメンションのスーパーセットが含まれている場合に最適です。



基底テーブルが 1 つしかないデータソースではなく、複数の基底テーブルを使用してマルチファクト関係モデルを構築する必要があるシナリオはさまざまです。

- **円環関係**。円環関係は現在サポートしていません。循環のあるデータソースを構築しようとしている場合は、マルチファクト関係を使用し、ダウンストリーム テーブルを別の基底テーブルにします。
- **適合ディメンション**と**文脈的 OR 関係**。同じ関係句のセット(日付と場所など) に基づいて関連付けられている一連のテーブルがある場合、それらのディメンションを抽出して共有テーブルにする必要があります。
  - そうすると便利であるのは、複数の関係句がすべて真 (論理的には AND) であることが、それらのレコードにテーブルを関連付けるために必要であるからです。
  - 代わりに、一度に真になる可能性があるのは 1 つであるレコード(文脈的 OR) を分析する場合は、共有ディメンション テーブルを使用してデータモデルを設定することで、この柔軟性が実現されます。
- **同等のブレンド**。データソースをプライマリとセカンダリに区別せず、同等にブレンドしたい場合、ブレンドのデータソースと、共有テーブル内のリンク フィールドを結合するデータモデルを構築します。

データモデルにおけるリレーションシップの要件

- テーブルを関連付ける際は、リレーションシップを定義するフィールドが同じデータ型である必要があります。[データソース] ページでデータ型を変更しても、この要件は変更されません。Tableau では、クエリの参照元データベースのデータ型が引き続き使用されます。
- リレーションシップは地理的フィールドに基づいて定義することはできません。
- データモデルでは循環関係に対応していません。
- パブリッシュされたデータソース間の関係を定義することはできません。

関連付けたテーブルの利点が限られる要因

- テーブル内にダーティデータがある(つまり、適切に構造化されたモデルを考慮して作成しておらず、メジャーとディメンションが複数のテーブルに混在している) 場合、複数テーブルの分析がさらに複雑になることがあります。

- データソースフィルターを使用すると、データ内で結合選択を行うTableauの機能が制限されます。結合選択とは、Tableauで不要な結合を削除してクエリを簡略化する方法のことです。
- 関係全体を通じて一致しない値が多く含まれるテーブル。
- バージョン2020.2から2024.1では、複数のファクトテーブルと複数のディメンションテーブルを相互に関連付けた場合(共有ディメンションや適合ディメンションのモデル化を試みた場合)。バージョン2024.2以降では、マルチファクト関係を使用してこれらのケースに対処できます。

関係と結合の違い

関係は、複数のテーブルのデータを分析用に組み合わせる動的で柔軟な方法です。リレーションシップの結合タイプは定義しないため、リレーションシップを作成するときにはベン図が表示されません。

関係は、2つのテーブル間の契約と考えることができます。これらのテーブルのフィールドを使ってVizを構築する場合、Tableauは、その契約を使用してこれらのテーブルからデータを取り込み、適切な結合を使用してクエリを作成します。

- **結合タイプが不要**。必要な操作は、一致するフィールドを選択して関係を定義することだけです(結合タイプは定義しません)。Tableauでは、既存のキー制約と一致するフィールド名に基づいて、リレーションシップの作成を試みます。次に、それらが使用するフィールドであることを確認するか、フィールドペアを追加して、テーブルを関連付ける方法をさらに明確に定義します。
- **コンテキストに応じた自動処理。**関係では、分析時にコンテキストが発生するまで結合が行われません。ビジュアライゼーションで使用されているフィールドに基づいて結合タイプが自動的に選択されます。分析中は、結合タイプがインテリジェントに調整され、ネイティブの詳細レベルがデータ内で保持されます。元となる結合について考えずに、Vizのフィールドの詳細レベルで集計を見ることができます。FIXEDなどのLOD式を使用して、関連付けられたテーブル内でデータが重複しないようにする必要はありません。
- **柔軟**。関係は多対多にすることができ、完全外部結合を使用できます。リレーションシップを使用してテーブルを組み合わせるのは、全データがワークブックの単一データソースに入っている、すべてのViz用の柔軟なカスタムデータソースを作成するようなものです。Tableauでは、ビジュアライゼーションのフィールドとフィルターに基づいて必要なテーブルのみがクエリされるため、さまざまな分析フローに使用できるデータソースを構築できます。

詳細については、「データの関連付け」と「関係を敬遠しないで」を参照してください。

結合は、データを組み合わせるためのオプションとして引き続き使用できます。論理テーブルをダブルクリックして、結合キャンバスに移動します。詳細については、結合についてを参照してください。

ビデオを視聴する: Tableau で関係を使用する方法については、この 5 分間のビデオを参照してください。

注: このビデオで示されている関係を編集するためのインターフェースは、現在のリリースとは少し異なりますが、同じ機能を備えています。

また、「Tableau が関係を発明した理由」など、Action Analytics からの関係に関するビデオ ポッドキャストも参照してください。詳細については、ライブラリの [ビデオ ポッドキャスト] をクリックしてください。

リレーションシップ クエリのしくみの関連情報については、Tableau の次のブログ投稿を参照してください。

- 関係 (パート 1): Tableau に新しいデータ モデリングを導入する
- 関係 (パート 2): ヒントとテクニック
- 関係 (パート 3): 複数の関連するテーブルを横断して質問する

リレーションシップと結合の特性

関係は、複数のテーブルのデータを分析用に組み合わせる動的で柔軟な方法です。関係によってデータの準備と分析がより簡単かつ直感的に行えるようになるため、データを結合する際の最初のアプローチとして関係を使用することをお勧めします。結合は、必要不可欠な場合にのみ使用してください。

リレーションシップを使用してテーブルを組み合わせると、次のような利点があります。

- データ ソースの定義、変更、再利用が容易になります。
- 正確な詳細レベル (LOD) での複数テーブルにまたがるデータ分析が容易になります。
- 詳細レベルが異なる分析では、LOD 式または LOD 計算を使用する必要はありません。
- 現在のビジュアライゼーションで使用されているフィールドを持つテーブルのデータに対してのみ、クエリが実行されます。

関係

- 論理テーブル間に柔軟性の高いヌードルとして表示されます。
- 2 つの論理テーブル間で一致するフィールドを選択する必要があります。
- 結合タイプを選択する必要はありません。
- 関連付けられたテーブルのすべての行データと列データをデータソースでも使用できるようにします。
- データソースでは分析中も、各テーブルの詳細レベルを維持します。
- 複数の詳細レベルで独立したドメインを作成します。テーブルはデータソースにはマージされません。
- 分析では、使用しているフィールドに基づいて適切な結合が自動的に作成されます。
- 集計値を重複させない (パフォーマンス オプションを [多対多] に設定している場合)
- 一致しないメジャー バリューを保持する (パフォーマンス オプションを [Some Records Match (一部のレコードが一致)] に設定している場合)

結合

結合とは、データの静的に組み合わせる方法です。分析を行う前に、物理テーブル間の結合を事前に定義する必要があり、定義を変更すると、そのデータソースを使用するすべてのシートに影響が及びます。結合したテーブルは常に単一のテーブルにマージされます。その結果、結合したデータに不一致の値が欠落するか、集計値が重複する場合があります。

- 物理テーブルごとにベン図アイコンが表示されます。
- 結合タイプと結合句を選択する必要があります。
- 結合した物理テーブルは、データの組み合わせが固定された単一の論理テーブルにマージされます。
- 不一致のメジャー バリューをドロップする可能性があります。
- フィールドが異なる詳細レベルである場合、集計値が重複する可能性があります。
- 抽出フィルターや集計など、データの単一テーブルが必要なシナリオに対応できます

リレーションシップの使用要件

- テーブルを関連付ける際は、リレーションシップを定義するフィールドが同じデータ型である必要があります。[データソース] ページでデータ型を変更しても、この要件は変更されません。Tableau では、クエリの参照元データベースのデータ型が引き続き使用されます。
- リレーションシップは地理的フィールドに基づいて定義することはできません。
- データモデルでは循環関係に対応していません。
- パブリッシュされたデータソース間の関係を定義することはできません。

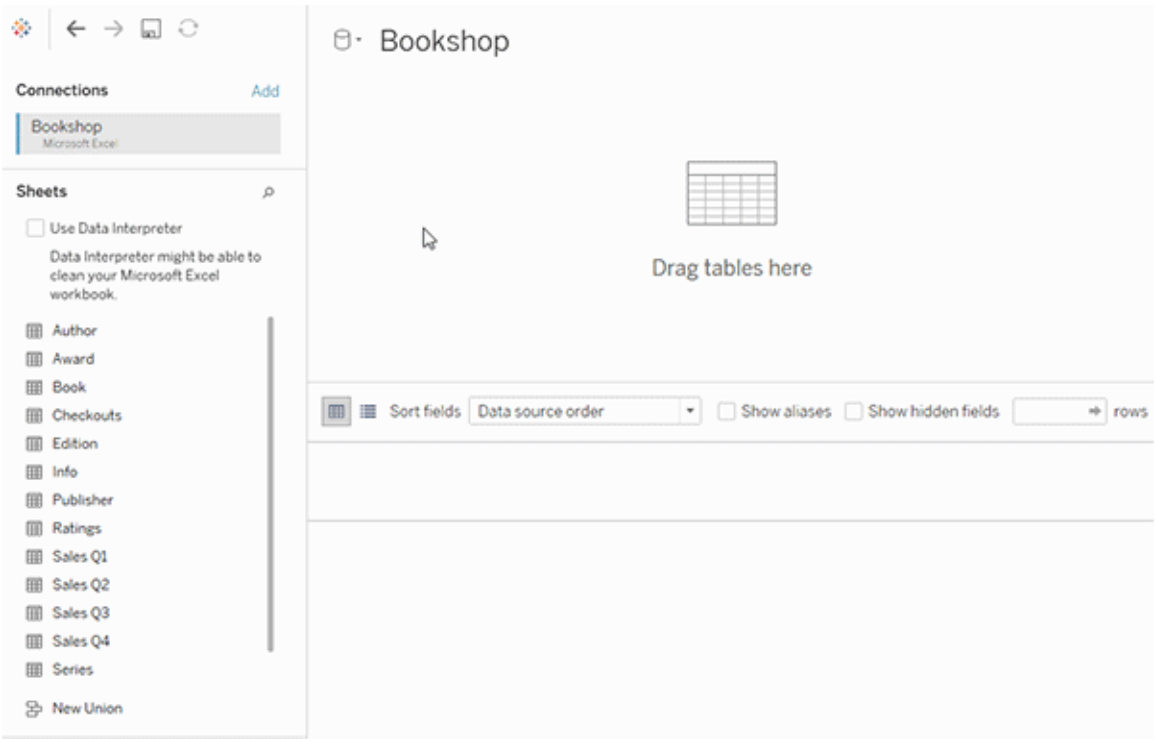
関連付けたテーブルの利点が限られる要因

- テーブル内にダーティデータがある(つまり、適切に構造化されたモデルを考慮して作成しておらず、メジャーとディメンションが複数のテーブルに混在している)場合、複数テーブルの分析がさらに複雑になることがあります。
- データソースフィルターを使用すると、データ内で結合選択を行うTableauの機能が制限されます。結合選択とは、Tableauで不要な結合を削除してクエリを簡略化する方法のことです。
- 関係全体を通じて一致しない値が多く含まれるテーブル。
- バージョン2020.2から2024.1では、複数のファクトテーブルと複数のディメンションテーブルを相互に関連付けた場合(共有ディメンションや適合ディメンションのモデル化を試みた場合)。バージョン2024.2以降では、マルチファクト関係を使用してこれらのケースに対処できます。詳細については、「関係を使用したマルチファクト分析」および「マルチファクト関係データモデルについて」を参照してください。

結合について

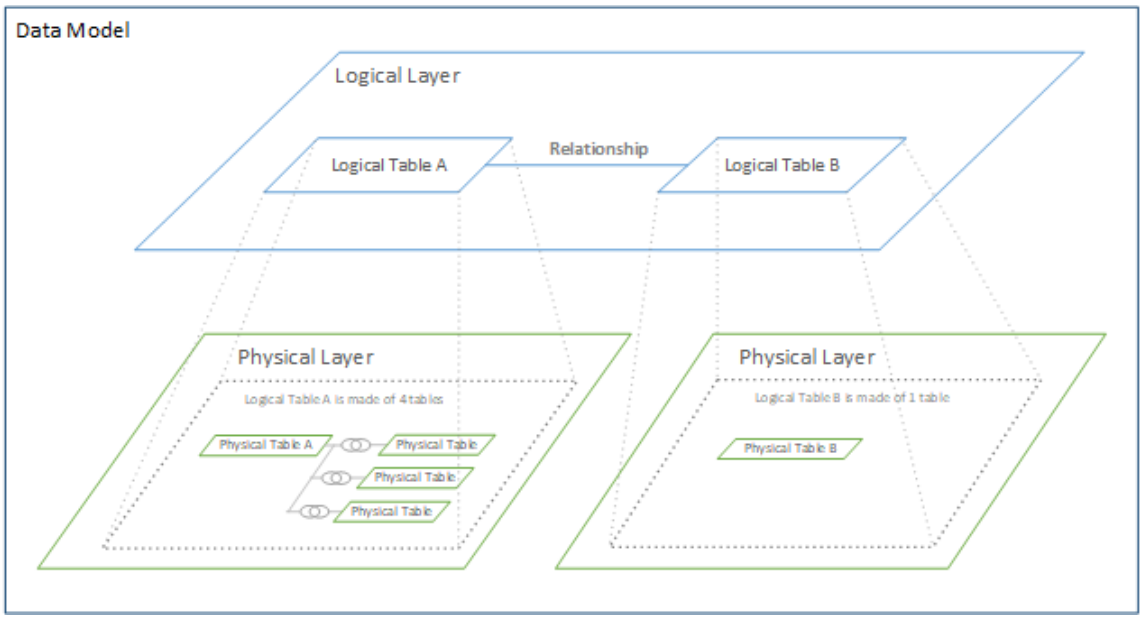
データソースの物理レイヤー内のテーブル間では引き続き結合を指定できます。論理テーブルをダブルクリックして、物理レイヤーの結合/ユニオンのキャンバスに移動し、結合またはユニオンを追加します。

すべての最上位の論理テーブルには、少なくとも1つの物理テーブルが含まれています。論理テーブルを開くと、その物理テーブル間の結合を表示、編集、作成できます。論理テーブルを右クリックし、**[開く]**をクリックします。テーブルをダブルクリックしても開くことができます。



作成したデータソースには 2 つのレイヤーがあります。最上位のレイヤーは、データソースの論理レイヤーです。論理レイヤーでは、関係を使用して表間でデータを組み合わせます。

次のレイヤーは、データソースの物理レイヤーです。物理レイヤーでは、結合を使用してテーブル間でデータを組み合わせます。詳細については、「データモデルの構造」を参照してください。



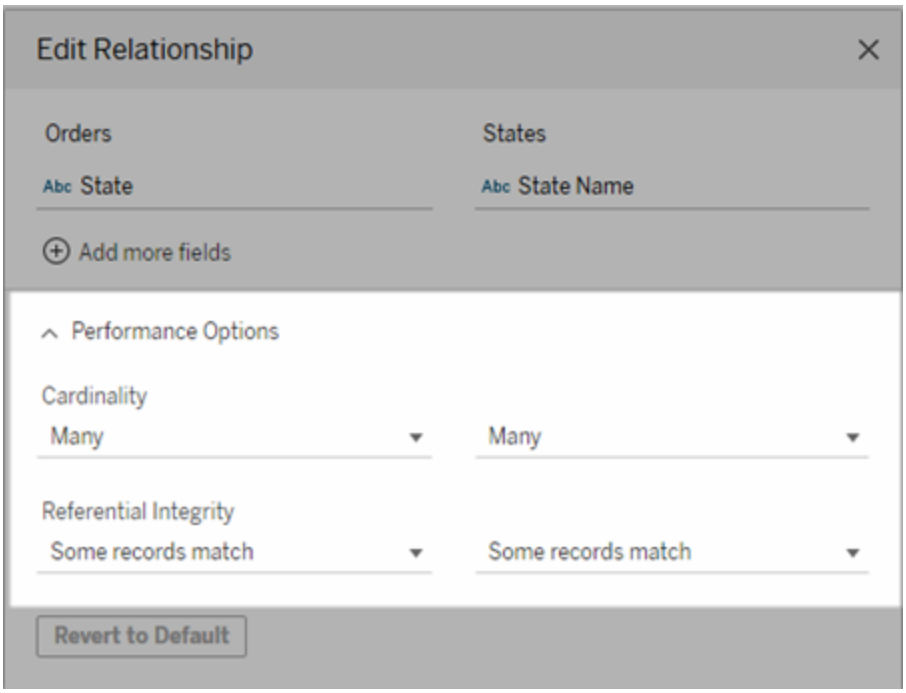
パフォーマンス オプションを使用してリレーションシップ クエリを最適化する

パフォーマンス オプションは、リレーションシップ内の 2 つのテーブル間のカーディナリティ(一意性) と参照整合性(一致する記録) を定義するオプションの設定です。これらの設定は、Tableau での分析中にクエリを最適化するのに役立ちます。

- **適切な選択肢がわからない場合は**、Tableau で表示される既定の推奨設定を使用してください。既定値は安全に使用でき、分析中に正しい集計と結合が自動的に生成されます。基数や参照整合性がわからない場合は、これらの設定を変更する必要はありません。
- **データの形状がわかっている場合は**、必要に応じてこれらの設定を変更し、2 つの表のレコード間の一意性と一致を表すことができます。

多くの分析シナリオでは、関係の既定の設定を使用すると、分析に必要なすべてのデータを得ることができます。シナリオによっては、パフォーマンス オプションの設定を調整してデータをより正確に記述できます。リレーションシップを使用してデータを組み合わせて分析する詳細については、「データの関連付け」とTableau ブログ記事「リレーションシップ、パート 1: Tableau に新しいデータモデリングを導入する」を参照してください。

カーディナリティと参照整合性の設定の意味



カーディナリティオプション

カーディナリティの設定は、分析中にデータを自動的に結合する際の前後に表データを集計するかどうかを決定します。

- フィールドの値が一意でない場合や一意かどうかわからない場合は、**[Many (多)]** を選択します。分析中に結合を形成する前に、関連データが集計されます。
- フィールド値が一意の場合は、**[One (1)]** を選択します。分析中、集計前に関連データが結合されます。このオプションを設定すると、リレーションシップ内のフィールド値が一意の場合にワークブック内のクエリが適切に最適化されます。ただし、フィールド値が一意でない場合に **[One (1)]** を選択すると、ビューに重複する集計値が表示される場合があります。

注: **[1]** を選択すると、各キー値が一意で、NULL 値を持つ行が 1 つだけの場合と同様にレコードが扱われます。

参照整合性オプション

参照整合性の設定では分析中にメジャーのディメンション値を取得するために使用する結合のタイプを決定できます。

- フィールドの一部の値が他のテーブルの値と一致しない場合や、一致しているかどうかが不明な場合は、**[一部のレコードが一致する]** を選択します。分析中、Tableau では外部結合を使用してメジャーのディメンション値が取得されます。一致しないメジャーを含むすべてのメジャー値がビューに表示されます。
- フィールドの値が他のテーブルの値と確実に一致する場合は、**[すべてのレコードが一致する]** を選択します。この設定により、分析時に生成される結合は少なく、またシンプルになり、クエリが最適化されます。この表に一致しない値がある場合、分析中に表示される結果が一貫しなくなる(一致しない値が削除されたりビューに表示されなかったりする) ことがあります。

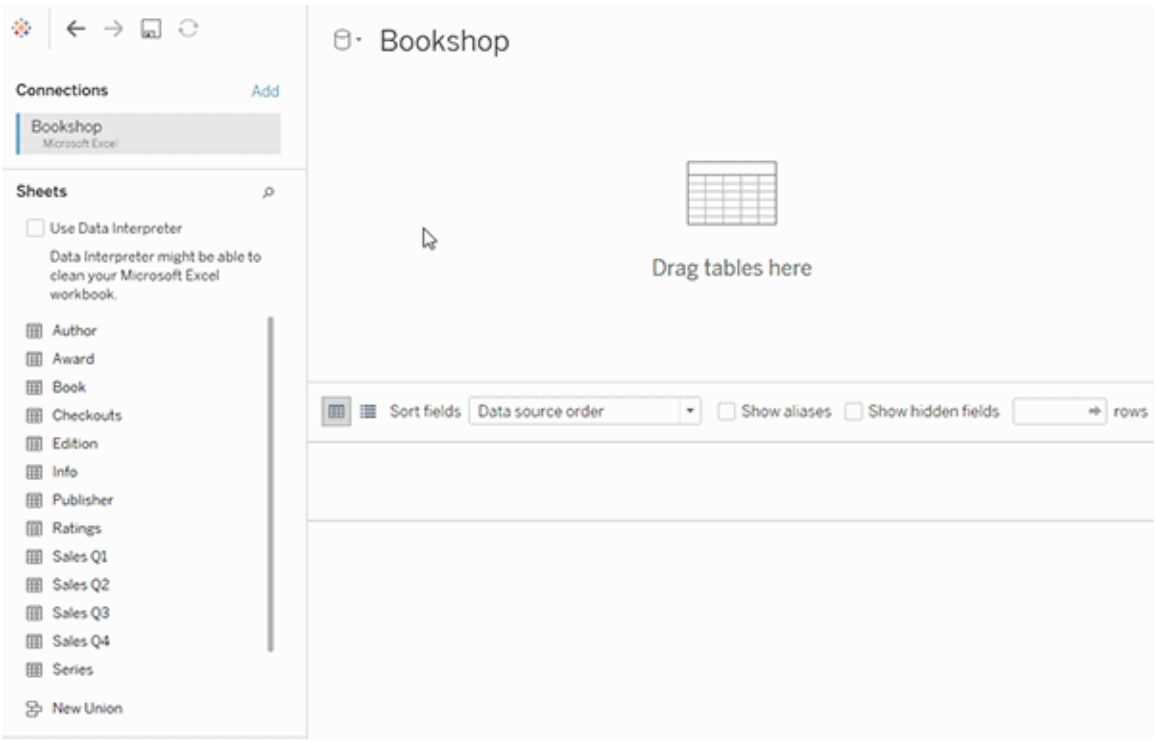
注: **[すべてのレコードが一致する]** を選択すると、リレーションシップに使用されるフィールドに NULL 値が存在しない場合と同様にレコードが処理されます。分析中には、Tableau は内部結合を使用してメジャーの値が取得されます。既定では、Tableau は NULL キーを結合しません。

カーディナリティと参照整合性の概念の詳細については、「カーディナリティと参照整合性」を参照してください。

結合について

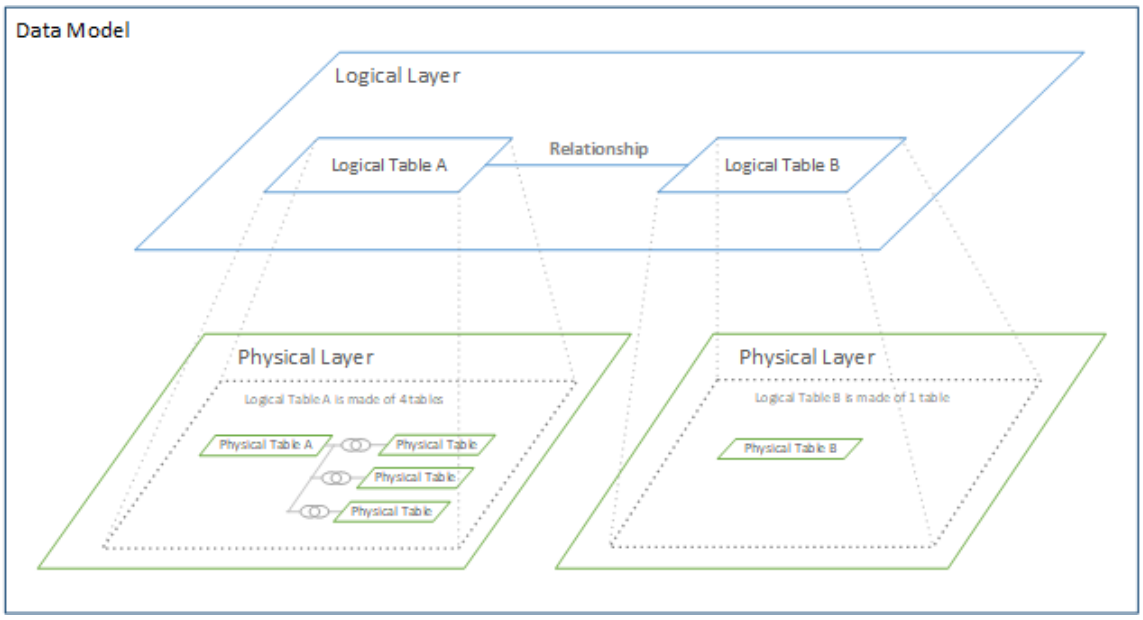
データソースの物理レイヤー内のテーブル間では引き続き結合を指定できます。論理テーブルをダブルクリックして、結合キャンバスに移動します。

すべての最上位の論理テーブルには、少なくとも 1 つの物理テーブルが含まれています。論理テーブルを開くと、その物理テーブル間の結合を表示、編集、作成できます。論理テーブルを右クリックし、**[開く]** をクリックします。テーブルをダブルクリックしても開くことができます。



作成したデータソースには 2 つのレイヤーがあります。最上位のレイヤーは、データソースの論理レイヤーです。論理レイヤーでは、関係を使用して表間でデータを組み合わせます。

次のレイヤーは、データソースの物理レイヤーです。物理レイヤーでは、結合を使用してテーブル間でデータを組み合わせます。詳細については、「データモデルの構造」を参照してください。



パフォーマンス オプションの使用に関するヒント

データの形状がわかっている場合は、[パフォーマンス オプション] のオプション設定を使用して、テーブルのカーディナリティを互いに (1 対 1、1 対多、多対多など) 確立し、参照整合性を示すことができます (1 つのテーブルの値は常に他のテーブルに一致があります)。

[パフォーマンスオプション] の各設定の意味は、「はい」と「いいえ」ではなく、「はい」と「わからない」と考えることができます。テーブルの値が一意であることが確実である場合は、**[1]**を選択します。一方のテーブルの各レコードが、もう一方のテーブルの 1 つ以上のレコードと一致することを確認する場合は、**[すべてのレコードが一致する]** チェック ボックスをオンにします。それ以外の場合は、既定の設定をそのまま使用します。

データの形状が不明な場合は、既定の設定を使用します。Tableau のデータ内でこれらの設定を検出できない場合の既定の設定は、次のとおりです。

- カーディナリティ:多対多
- 参照整合性: 一部のレコードが一致す

Tableau がデータ内の主要なリレーションシップや参照整合性を検出すると、それらの設定が使用され、「検出済み」として表示されます。

既定の設定を再適用するには、**[既定値に戻す]** をクリックします。

用語の定義

*カーディナリティ*とは、フィールド(列) やフィールドの組み合わせに含まれるデータの一意性を指します。分析する表に多くのデータ行が含まれる場合、クエリに時間がかかる場合があります (および全体的なデータソースのパフォーマンスが影響されます)。そのため、Tableau では、表間の関連列のカーディナリティに基づいてデータを組み合わせる方法を選択することをお勧めします。

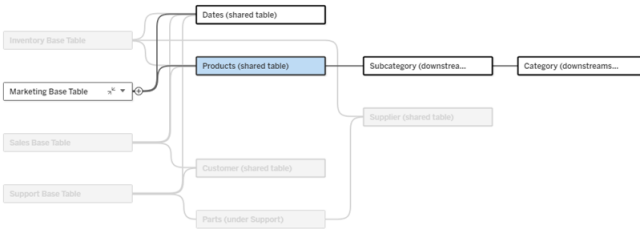
- 低カーディナリティー: 関連する列に多くの繰り返しデータがある場合。たとえば、[Products (製品)] というテーブルには、[Furniture (家具)]、[Office Supplies (オフィス用品)]、[Technology (テクノロジー)] という3 つの値を含む [カテゴリ] 列が含まれる場合があります。
- 高基数: 関連する列に非常に一意なデータがある場合。たとえば、[Orders (注文)] というテーブルには、商品の注文ごとに一意の値を含む [Order ID (注文 ID)] 列が含まれる場合があります。

*参照整合性*とは、一方のテーブルの行に一致する行が常に他方のテーブルにも存在することを意味します。たとえば、売上表の行には常に製品カタログの表に一致する行があります。

マルチファクト関係データモデルについて

マルチファクト関係を使用すると、複数の基底テーブルを持つデータソースを構築できます。データモデルで複数の基底テーブルを使用すると、Tableau でマルチファクト分析を実行できます。

基底テーブルをルートとするテーブルのツリーを確立することで、異なる概念ドメインを持つデータ構造をモデル化し、それらの共有特性を使用してデータを接続することができます。このタイプの分析は、多くの場合、マルチファクト分析、適合ディメンション、または共有ディメンションと呼ばれます。Tableau では、関係を使用してモデルを構築することから、これをマルチファクト関係データモデルと呼びます。マルチファクト関係データモデルには、常に複数の基底テーブルが含まれます。基底テーブルは、データモデル内にある一番左のテーブルです。基底テーブルとして使用するテーブルを決定する方法については、「マルチファクト関係モデルを使用するとき」を参照してください。



1 つの基底テーブルのツリーが強調表示された複数の基底テーブル データモデル

関連性のレベル

複数の基底テーブルを持つデータモデルでは、データの断片を互いにどのように関連付けられるか (または関連付けないようにするか) についての柔軟性が高くなります。

注: 関連性の適用範囲はどのレベルにおいても、複数の基底テーブルを持つデータモデルのみです。マルチファクト関係データモデルが登場する前は、すべてが関連している (単一のデータソース内) か、何も関連していない (複数のデータソースにまたがってブレンドしている) かのいずれかでした。

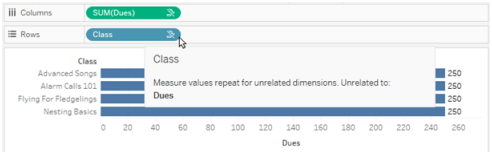
データモデルにおける関連性

テーブルは、データモデルの構造に基づいて、関連があったり、関連がなかったり、共有されたりします。データソースでは、テーブルの関連性は一定です。簡単な概要は次のとおりです。

- *関連テーブル*は同じツリー内にあります。
    - 2024.2 より前は、すべてのデータソースが単一のツリーで構成される単一の基底テーブル データソースであり、単一の基底テーブル データソースではすべてのテーブルが関連していました。
- *関連のないテーブル*は異なるツリー内にあります。基底テーブルは常に互いに関連していません。また、ただ 1 つの基底テーブルのダウンストリームにあるテーブルも、他のツリー内のテーブルとは関連していません。
- *共有テーブル*は複数の入力関係を持ち、複数のツリーに属します。
    - 複数の入力関係を持つテーブルのダウンストリームにあるテーブルも共有テーブルと見なされます。

分析中の関連性

フィールドは、関連していたり、まだ関連していなかったり、あいまいに関連していたりすることがあり、さらに、つなぎ合わせるフィールドとして機能することもあります。フィールドのグループ間の関連性は、データモデルの構造や、どのフィールドがアクティブに使用されているか (つまり、ピルとしてシェルフに並んでいるか)、およびそれらのフィールドがディメンションであるかメジャーであるかに基づいて、シートごとに決定されます。



複数のテーブルのフィールドを使用してビジュアライゼーションを作成するには、Tableau はバックグラウンドで結合を実行して値を計算する必要があります。使用する結合のタイプは、フィールドの関連性によって異なります。簡単な概要は次のとおりです。

- Viz で*関連フィールド*が使用されている場合、ディメンションは内部結合され、メジャー バリューはディメンションごとに分類されます。
    - 実際はもう少し複雑であり、メジャー バリューがドロップされないようにするために、バックグラウンドでの追加の結合が必要になる場合があります。しかし、ディメンションのみの Viz では、関連するディメンションが内部結合されており、それがここでの主な概念となります。
    - これは、単一基底テーブル モデルと同じ動作です。
- Viz で*関連のないフィールド*が使用されている場合、ディメンションはクロス結合されます。メジャー バリューはテーブル スコープ (つまり、テーブル全体の単一の値にローカルに集計) され、繰り返されます。
    - また、フィールドが*まだ関連付けられていないか、あいまいに*関連している可能性もあります。つまり、アクティブなフィールドの組み合わせでは、テーブル間の関係を解決する方法が複数あるということです。Tableau は不明瞭なフィールドに遭遇すると、それらを関連のないフィールドとして処理します。
- フィールドが共有フィールドに基づいて*つなぎ合わされる*場合、ディメンションは外部結合されます。メジャー バリューは、ディメンションごとの分類レベルで集計され、繰り返される場合があります。
    - *ディメンションのつなぎ合わせは*、データブレンドにおけるフィールドのリンクに似ています。関連するフィールドのペアごとに結果が計算され、関連のない値は、それらの間で共有されるディメンションの共有値全体でつなぎ合わされます。

これらの概念と定義については、このトピックの後半で詳しく説明します。

ディメンションとメジャー関する補足説明

Tableau では、*メジャー*は*集計*であり、ビュー内の*ディメンション*によって設定される*粒度*まで集計されます。したがって、メジャーの値はディメンションのコンテキストによって異なります。たとえば、「シリアルの箱の数」は、在庫総数を意味するのか、ブランドごとの箱数を意味するのかによって異なります。

*ディメンション*は通常、国やブランドなどのカテゴリフィールドです。Tableau では、ディメンションによってビューの粒度、つまり*詳細レベル*が設定されます。通常、いくつかのカテゴリの組み合わせごとにマークを使用してデータをグループ化します。ビューを構築するために使用するディメンションによって、マークの数が決まります。

メジャーがディメンションなしで使用される場合、それは、*テーブルスコープ*と呼ばれます。つまり、その値はテーブル全体の完全な集計値ということです。Viz でブランドなどのディメンションを使用するとすぐに、メジャーがより細かく分類されます。シリアルの箱の総数がブランドごとに表示されるようになります。

*集計*は、データがどのように組み合わせられているかを指すものです。Tableau の規定の集計は SUM です。集計は、平均値、中央値、個体数、最小値など、さまざまなオプションに変更できます。*粒度*は、メジャーがどの程度細かく分類されているかを指し、ディメンションによって制御されます。メジャーの粒度が行レベル (つまり非集計) でない限り、その値を集計する必要があります。

例

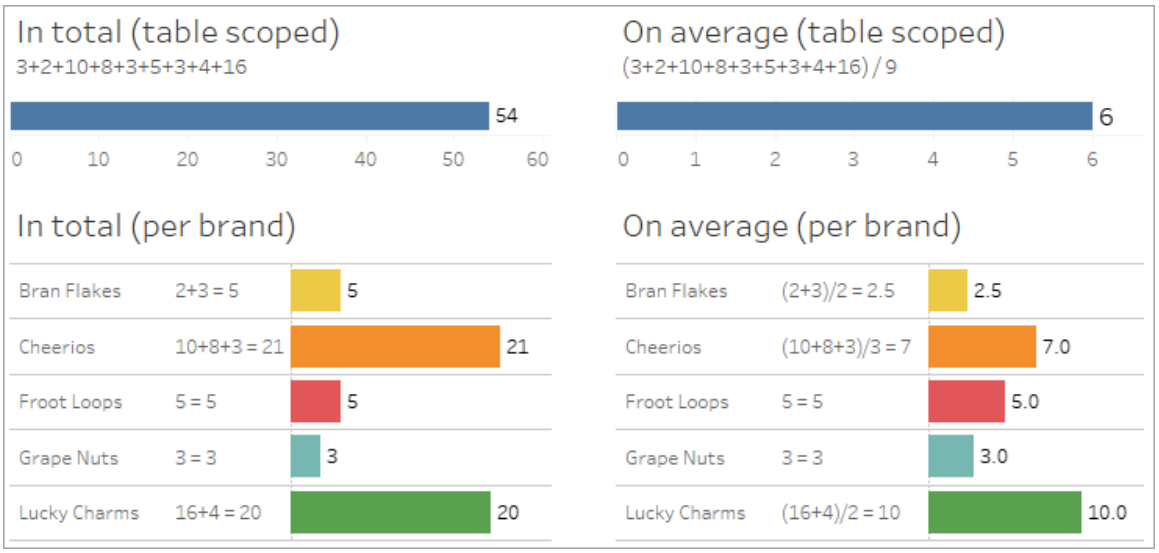
Raw data

	Mini	Normal	ValuePak
Bran Flakes		3	2
Cheerios	10	8	3
Froot Loops		5	
Grape Nuts		3	
Lucky Charms		4	16

「シリアルの箱の数」の値は何ですか?

この値は、集計タイプと、ディメンションによって設定される粒度によって異なります。

- 集計:
  - Sum (合計)
  - Average
- 粒度:
  - テーブルスコープ/完全集計 (図の例では青いバー)
  - **ブランド**ディメンションごとに分類 (図の例では色付きのバー)

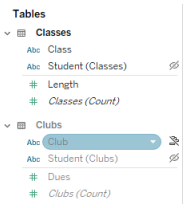


フィールドレベルの関連性指標

分析で使用しているフィールドの関連性の程度を理解しやすくするための視覚的な手がかりがいくつかあります。

ワークシート上の関連性指標

- **関連のないアイコン**: Tableau は関連のないアイコンを使用して、ビュー内のすべてが関連しているわけではないことを示します。ビューまたはデータペイン内のピルに関連のないアイコンが表示されている場合は、アイコンの上にカーソルを合わせると、詳細情報が表示されます。
    - 関連ありアイコンは、フィールドが関連のないフィールドとつなぎ合わされていることを示します。
- **薄い灰色のフィールド名**: フィールド名がシェルフで使用されている*フィールド*と関連していない場合、[データペイン] に薄い灰色のテキストで表示されます。これらのフィールドは引き続きその Viz の分析に使用できますが、分析では、関連のないフィールドは関連するフィールドとは異なる方法で評価されます。また、これらのフィールドにカーソルを合わせると、関連のないアイコンが表示されます。

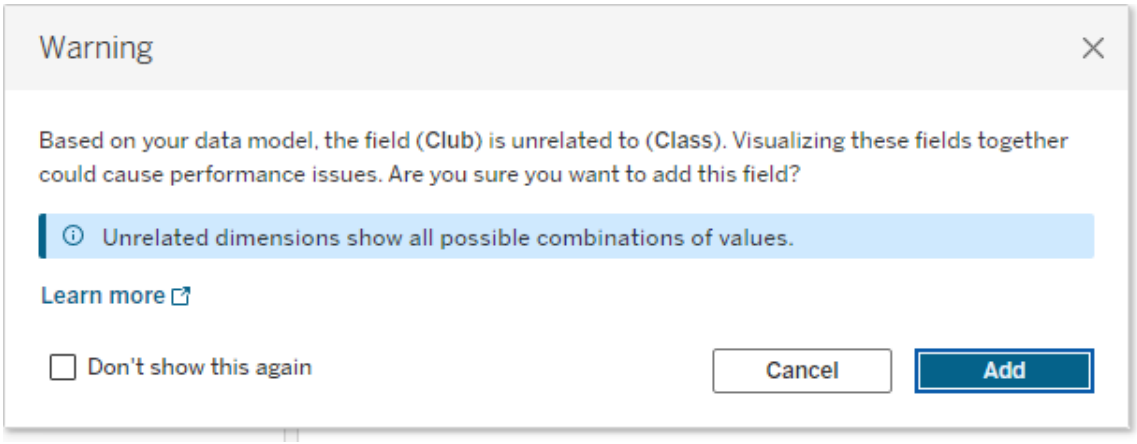


注: Tableau の以前のバージョンでは、薄い灰色のフィールド名はフィールドが非表示で、**[非表示フィールドの表示]** が選択されていることを示していました。非表示のフィールドは、クリック可能な目のアイコンとともに表示されるようになりました。

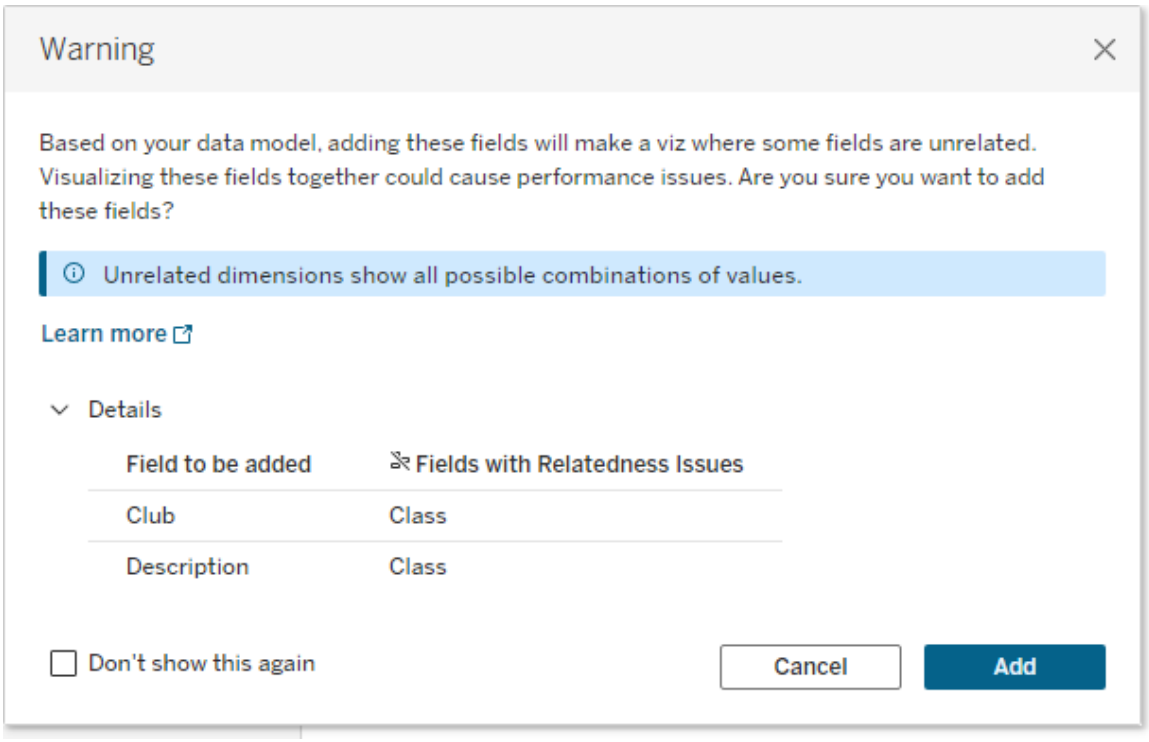
関連性に関する警告ダイアログ ボックス

関連のないフィールドが Viz 内で一緒に使用されている場合、Tableau はフィールドが関連していないことを知らせる警告ダイアログ ボックスを表示します。この警告は、パフォーマンスに影響を与える可能性のある偶発的なクロス結合を防ぐために、関連のないフィールドが追加されるたびに表示されます。

- 関連のないフィールドをつなぎ合わせずに使用する場合は、**[追加]** をクリックして Viz へのフィールドの追加を続行します。
- 関連のないフィールドをつなぎ合わせる場合は、ベストプラクティスとして、つなぎ合わせるフィールドを関連のないフィールドの前に配置します。つなぎ合わせるフィールドがすでに使用されている場合、ダイアログは表示されません。つなぎ合わせによってクロス結合を防止する方法については、「関連性の各レベルにおける結合の使用方法」をご覧ください。



複数のフィールドが追加されている場合、またはそれらが既にビューに存在する場合は、ダイアログに **[詳細]** 領域が表示されます。これを展開すると、使用中のすべてのフィールドの関連性に関する詳細情報が表示され、関連がないという問題がどこで発生しているかを特定することができます。



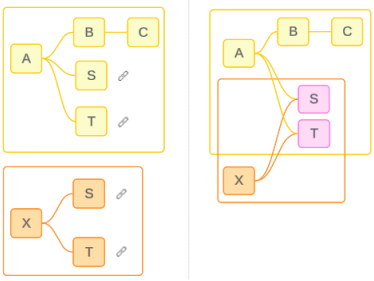
警告メッセージが表示されないようにするには、**[これを次から表示しない]** オプションを選択します。これらの警告メッセージは、次の手順に従って再度オンにすることで、いつでも有効にすることができます。

- Tableau Desktop で、**[ヘルプ]** メニュー > **[設定とパフォーマンス]** > **[無視したメッセージをリセット]** の順にクリックします。
- ブラウザーでキャッシュされたデータを消去します。たとえば Chrome では、**3 つの点が縦に並んでいるアイコン**をクリックし、**[閲覧履歴データを削除]** > **[キャッシュされた画像とファイル]** > **[データを削除]** の順にクリックします。

データモデルにおけるテーブル レベルの関連性

複数の基底テーブルを持つデータモデルでは、各基底テーブルは関連しているテーブルのセットを正義し、概念ツリーを形成します。全体のデータソースが単一のエンティティになるように、これらのツリーを少なくとも 1 つの共有テーブルによって接続する必要があります。

これにより、以前はリンクフィールドを使用してブレンドすることができた 2 つのデータソースが、これらの共通フィールドを含む共有テーブルによって接続される、2 つのツリーを持つ単一のデータソースになります。



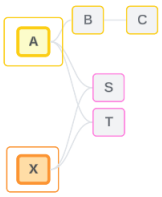
Tip: データモデル内でテーブルがどのように関連付けられるかは、分析でフィールドがどのように関連付けられるかに影響します。分析中に[データモデルの表示]ダイアログを使用すると、テーブルが全体的なデータモデルにどのように適合するかを確認するのに役立ちます。

このサンプル データソースを使用して、どのテーブルが関連しているか、または関連していないか、あるいは共有されているかを確認してみましょう。2 つのツリーがあり、1 つは基底テーブル A によって確立され、もう1 つは基底テーブル B によって確立されています。

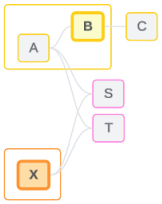
関連のないテーブル

各基底テーブルは基本的に関連していません。同様に、単一のツリーにのみ存在するテーブルは、他のツリー内のテーブルとは関連していません。

テーブル A とテーブル X は関連していません



テーブル B とテーブル X は関連していません

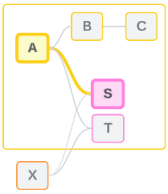


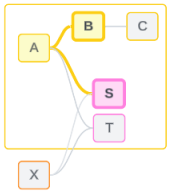
関連テーブル

同じツリー内のテーブルは関連していると見なされます。

テーブル A とテーブル S は関連しています

テーブル B とテーブル S は (テーブル A を介して) 関連しています

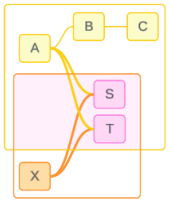




共有テーブル

各共有テーブルには複数の入力関係があります。これらのテーブルは複数のツリーに属しており、そのツリー間で共有されます。

テーブル S とテーブル T は共有されます。



分析におけるフィールドレベルの関連性

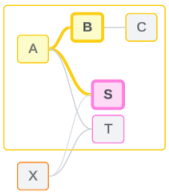
フィールド間の関連性は、データモデルの構造や、どのフィールドがアクティブに使用されているか(つまり、どのフィールドがシェルフ上のピルとして Viz に表示されているか)、およびそれらのフィールドがディメンションであるかメジャーであるかに基づいて、シートごとに決定されます。フィールドの関連性が Viz の結果にどのように影響するかについては、次のセクションで説明します。

同じサンプル データ ソースを使用していくつかのシナリオを見てみましょう。各フィールドの名前は、テーブル B の FieldB のように、どのテーブルからのフィールドであるかを示します。特に明記されていない限り、フィールドはディメンションまたはメジャーになります。

関連フィールド

大まかに言えば、Tableau が単一のツリー内の関係パスに基づいてフィールドをまとめて評価する方法を明確に決定できる場合、それらのフィールドは互いに関連しています。

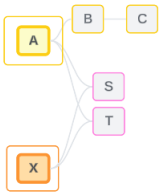
たとえば、FieldB (テーブル B から) と FieldS (テーブル S から) は関連しています。



関連のないフィールド

大まかに言えば、関連がなければ、フィールドは互いに関連していません。これは、2 つの基底テーブルのフィールドを使用しているなど、フィールドが関連のないテーブルからのものであることが原因である可能性があります。この場合、異なる基底テーブルのフィールドは基本的に関連していません。

たとえば、FieldA と FieldX は関連していません。



または、あいまいに関連している場合や、まだ関連付けられていない場合など、ある時点においてフィールドが関連がないものとして扱われることもあります。ほとんどの場合、関連性指標を使用すると、Viz のコンテキストでフィールドが関連していない場合に警告が表示されます。

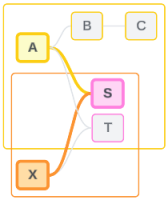
ディメンションのつなぎ合わせ

つなぎ合わせは、Tableau が分析中にマルチファクトデータモデル内の関連のないテーブルのフィールドを評価できるようにするためのものです。Viz で共有テーブルのディメンションを使用すると、関連のないフィールドがつなぎ合わされ、同じ Viz で同時に評価できるようになります。これは、共有するディメンションに基づいて 2 つのツリーの結果が並置されるということです。

たとえば、FieldA と FieldX を使用して Viz が構築されている場合、これら 2 つのフィールドには関連がありません。DimensionS を追加すると、つなぎ合わせるフィールドが導入されます。

- FieldA と DimensionS は一緒に評価されます。
- FieldX と DimensionS は一緒に評価されます。

- これらの中間結果は、DimensionS の値に基づいてまとめられます。
- FieldA と FieldX がつなぎ合わされました。



ヒント: ベストプラクティスは、関連のないフィールドを表示する前に、Viz でつなぎ合わせるフィールドを使用することです。たとえば、FieldA、FieldX、DimensionS の順にドラッグするのではなく、DimensionS を最初にドラッグするか、FieldA、DimensionS、FieldX の順にドラッグします。最初につなぎ合わせるフィールドを追加すると、Tableau は常に関係の評価方法を認識するようになるため、関連のないディメンションがクロス結合によってまとめて評価されることで起こりうるパフォーマンスの問題を回避することができます。

つなぎ合わせを行うには、共有テーブルのディメンションが Viz でアクティブになっている必要があります。フィルター シェルフまたは マーク カードのツールヒントプロパティに配置されたフィールドは、つなぎ合わせを行う際にはアクティブとは見なされません。

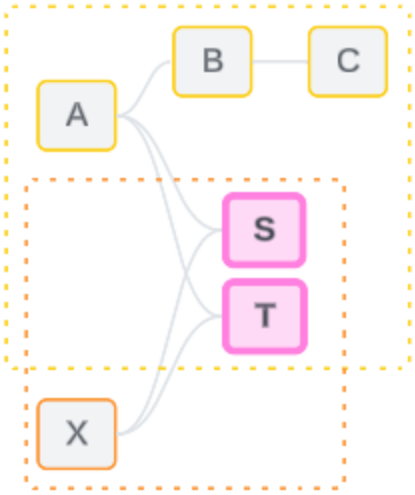
まだ関連付けられていないフィールド

フィールドに複数の関係が存在する可能性があるものの、まだ関連付けられていないという場合もあります。これは、2 つの共有テーブル (またはダウンストリームの共有テーブル) 間に複数の関係が存在する場合に発生します。

FieldS と FieldT について考えてみましょう。両者のテーブルは、基底テーブル A によって定義されたツリーと、基底テーブル X によって定義されたツリーの両方を通じて相互に関連付けられています。

FieldS と FieldT だけが示されている Viz では、それらを関連付けるためにどのツリーを使用すべきかについての情報がありません。追加情報がなければ、Tableau はこれらのフィールドを基底テーブル A のツリーを介して関連付けるか、あるいは基底テーブル B のツリーを介して関連付けるかを評価することができません。

複数の潜在的な関係が存在しているものの、FieldS と FieldT は関連のないフィールドとして扱われます。



関係が存在する可能性があるものの、まだ関連付けられていないフィールドは、Tableau が関係パスを明確に判断できないため、関連のないフィールドとして評価されます。つなぎ合わせが可能なまったく関連のないフィールドとは異なり、まだ関連付けられていないフィールドは解決可能であり、フィールドを直接関連付けることができます。

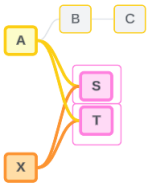
あいまいに関連しているフィールド

フィールドはあいまいに関連していることもあります。これは、共有テーブル (またはダウンストリームの共有テーブル) 間にアクティブな関係が複数存在する場合に発生します。まだ関連付けられていないフィールドは、関連性が低い、または関連性が不足していると考えられるのに対し、あいまいに関連するフィールドは、関連性が高い、または関連性が過剰であると考えられます。

FieldS と FieldT について考えてみましょう。両者のテーブルは、基底テーブル A によって定義されたツリーと、基底テーブル X によって定義されたツリーの両方を通じて相互に関連付けられています。

FieldA、FieldX、FieldS、FieldT が示されている Viz では、情報が多すぎるため、フィールドを関連付けるためにどのツリーを使用すべきかを判断することができません。情報をトリミングしなければ、Tableau はこれらのフィールドを基底テーブル A のツリーと基底テーブル B のツリーのどちらを介して関連付けるべきかを評価することができません。

複数の*アクティブ*な関係が存在しているものの、FieldS と FieldT は関連のないフィールドとして扱われます。

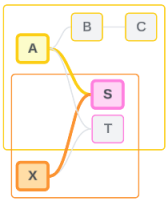


これらのあいまいに関連するフィールドは、Tableau が関係パスを明確に判断できないため、関連のないフィールドとして評価されます。つなぎ合わせが可能なまったく関連のないフィールドとは異なり、あいまいに関連するフィールドは解決可能であり、フィールドを直接関連付けることができます。

共有テーブルからのあいまいなメジャー

共有テーブルからディメンションが使用されると、関連のないアップストリーム テーブルからフィールドがつなぎ合わされます。ただし、メジャーをつなぎ合わせることはできないため、メジャーの値は関連するディメンションによって決まります。

DimensionA とDimensionX を含む Viz では、これら2 つのディメンションには関連がありません。MeasureS がテーブル S から使用される場合、DimensionA とDimensionX の組み合わせとは関連がありません。どちらか一方に独立して関連付けることはできますが、同じ Viz 内で両方に同時に関連付けることはできません。



共有メジャーは、あいまいに関連するもの、または過剰に関連するものとしてみなされ、同じように解決されます。

フィールド間の不明確な関係を解決する

フィールドを関連付ける方法が不明瞭である場合は、Tableau は恣意的な決定を下さずに、それらを関連のないフィールドとして扱います。多くの場合、どのツリーを使用するかに関して不明瞭な点を明確にしてから、これらのフィールドを関連付ける方が適切です。

まだ関連付けられていないフィールドを解決するには、使用するツリーを確立するためのフィールドを追加します。あいまいに関連しているフィールドを解決するには、どのツリーを使用するかを確定できるようにフィールドを削除します。

例:

まだ関連付けられていないフィールドを解決する: フィールドを追加

- FieldS とFieldT の Viz で、テーブル A、B、または C のフィールドを Viz に追加すると、基底テーブル A のツリーがアクティブになり、FieldS とFieldT 間の目的のパスが解決されます。
- あるいは、テーブル X のフィールドを使用すると、基底テーブル X のツリーに対する FieldS と FieldT 間の目的のパスが解決されます。

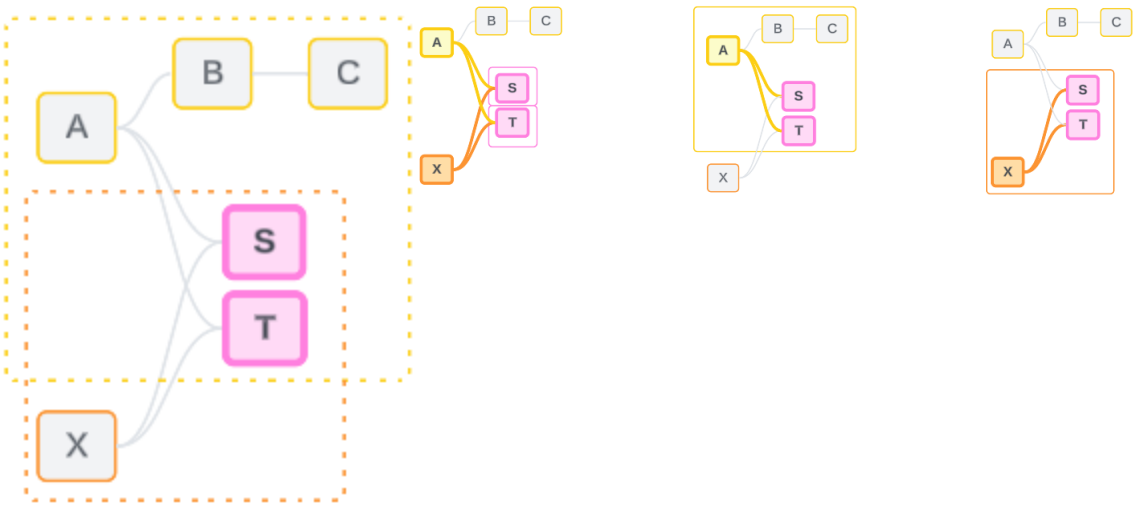
曖昧な関係を解決する: フィールドを削除

- FieldA、FieldX、FieldS、および FieldT の Viz で、FieldX を削除すると、基底テーブル A のツリーのみがアクティブになり、FieldS とFieldT 間の目的のパスが解決されます。
- あるいは、FieldA を削除すると、Base Table X のツリーを介して FieldS とFieldT 間の目的のパスが解決されます。

共有メジャーを解決する: フィールドを削除

- DimensionA、DimensionX、MeasureS が示されている Viz で、DimensionX を削除すると、基底テーブル A のツリーのみがアクティブになり、DimensionA とMeasureS 間の目的のパスが解決されます。
- あるいは、DimensionA を削除すると、基底テーブル X のツリーを介して DimensionX と MeasureS 間の目的のパスが解決されます。

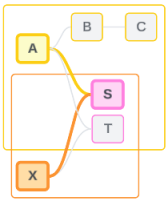
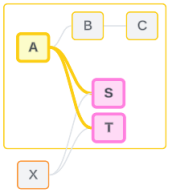
まだ関連付けられていない	あいまいに関連する	関連性が一本のツリーに解決される	
		基底テーブル A を介して関連付け	基底テーブル X を介して関連付け



不明瞭さを解決することは、FIXED 詳細レベル (LOD) の式を使用することと似ています。FIXED LOD 式では、ディメンション宣言を定義して、どの詳細レベルに集計するかを Tableau に指示します。不明瞭さは、Viz の構造を変更して 1 つのツリーのみをアクティブにすることで解決され、分析を実行するために考慮できる関係パスが Tableau に伝えられます。

つなぎ合わせ vs 不明瞭さの解決

つなぎ合わせと不明瞭さの解決はどちらも無関連性に対処する方法ですが、結果はそれぞれ異なります。

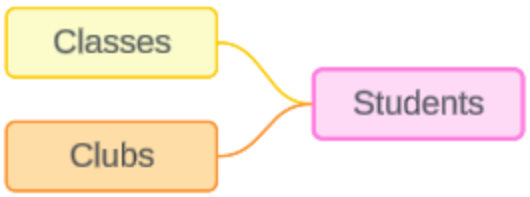
つなぎ合わせ	不明瞭さの解決
	
関連のない FieldA と FieldX が DimensionS によってつなぎ合わされる	FieldS と FieldT は基底テーブル A で定義されたツリーを介して評価される
共通の属性に基づいて関連のないフィールドを並置する	複数のオプションがある場合 (あいまいさや共有メジャーがある場合) は使用する関係パスを絞り込

	み、関係パスが存在しない (まだ関連付けられていない) 場合は関係パスを確立します。
複数の基底テーブル ロジックを使用して結果を計算する	単一の基底テーブル ロジックを使用して結果を計算する
分析に関連のないテーブルが含まれる	分析に共有テーブルが含まれる

関連性の各レベルにおける結合の使用方法

フィールドレベルの関連性が決定されたら、Tableau は結果を評価して実際のビジュアライゼーションを作成する必要があります。Viz に表示される値を計算するために使用されるクエリは結合に依存します。フィールドが関連しているか、関連していないか、またはつなぎ合わされているかによって、どの結合を実行するかへの影響が異なります。このコンテキストでは、あいまいに関連するフィールドとまだ関連付けられていないフィールドは、関連のないフィールドとして扱われることに注意してください。

関連性と結合を説明するために、このセクションではテーブルとそのフィールド、およびそれらのフィールドの値について説明します。2 つの基底テーブル (「クラス」と「クラブ」) と共有テーブル (「生徒」) を含む次のデータモデルを検討してください。



クラス	クラブ	生徒

Classes
7 rows 3 fields

Abc Classes Class	Abc Cla: Stu
Nesting Basics	Rol
Advanced Songs	Sp:
Flying For Fledgelings	Rol
Nesting Basics	Sp:
Advanced Songs	Fin
Nesting Basics	Fin
Alarm Calls 101	

Clubs
7 rows 3 fields

Abc Clubs Club	Abc Clubs Student ...
Photography	Finch
Travel	Cardinal
Juggling	Sparrow
Art	Finch
Art	Cardinal
Art	Sparrow
First Aid	Robin

Students
5 rows 3 fields

Abc Students Bus Rider	Abc Students Student	# Students Age
yes	Finch	3
yes	Cardinal	4
no	Sparrow	6
yes	Robin	3
no	Jay	8

フィールド:

- **クラス**:「巣作りの基本」、「高度な歌」、「ひな鳥の飛行」、「警戒音101」の値を持つディメンション
- **長さ** - メジャー
- **生徒**:「生徒」テーブルに関連付けるために使用されるディメンション

フィールド:

- **クラブ**:「写真」、「旅行」、「ジャグリング」、「美術」、「応急処置」の値を持つディメンション
- **会費**: メジャー
- **生徒**:「生徒」テーブルに関連付けるために使用されるディメンション

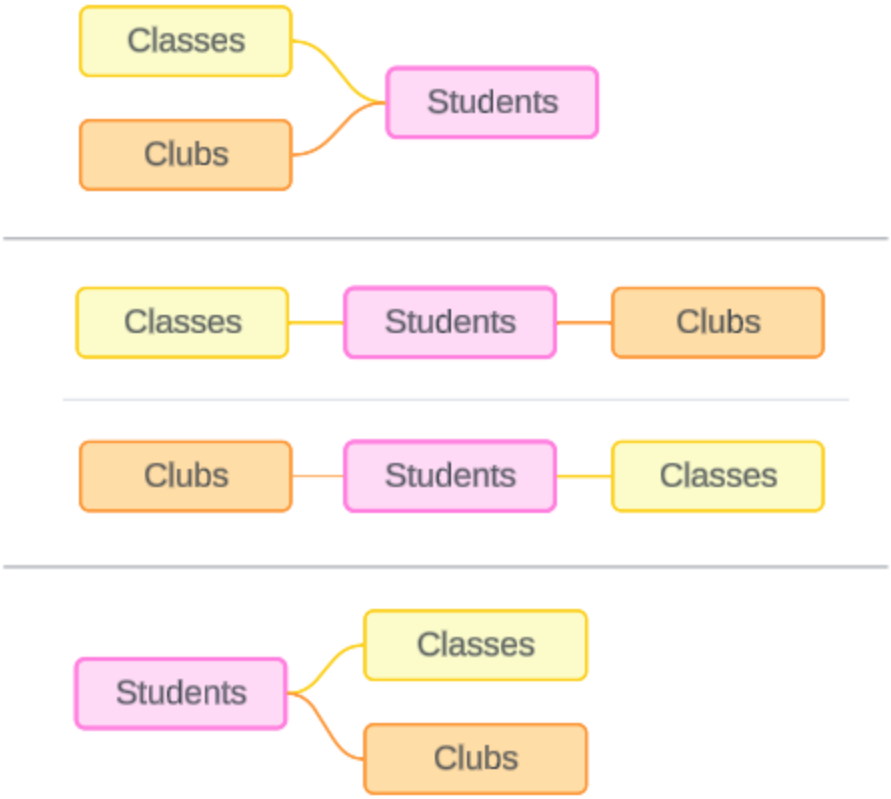
フィールド:

- **バス利用者**:「はい」または「いいえ」の値を持つディメンション
- **生徒**:「スズメ」、「コウカンチョウ」、「ヒメドリ」、「コマドリ」、「カラス」の値を持つディメンション他の2つのテーブルと関連付けるために使用される
- **年齢**: メジャー

この非常に単純なモデルは、マルチファクト関係データモデルに対して大まかなの結合ロジックがどのように計算されるかを示しています。関係に基づいて構築された単一の基底テーブル データモデルで使用される結合の基本については、「関係を使用する複数テーブルのデータソースでの分析動作」を参照してください。

この例は、複数の基底テーブルを持つデータモデルである必要がありますか?

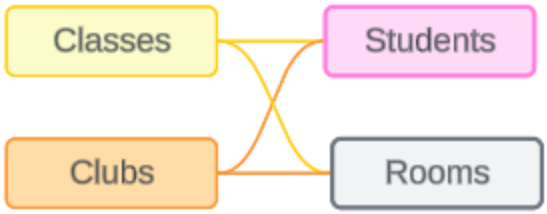
この 3 つのテーブルのデータモデルの場合、「クラス」-「生徒」-「クラブ」、または「クラブ」-「生徒」-「クラス」として、または「生徒」を基底テーブルとして、単一の基底テーブル モデルを設定したいと思うかもしれません。原則として、マルチファクト関係データモデルは、特定の種類のデータスキーマまたは分析シナリオを対象としています。データモデルにマルチファクト関係データモデルに最適な特性がある場合は、そのように設定して、基底テーブルを概念的に関連のない状態に保ちます。ただし、データにこのタイプの構造が必要ない場合は、単一の基底テーブル モデルを使用する方が簡単です。



これらの 3 つのテーブルに対して構築できるモデル: (1)「クラス」と「クラブ」を基底テーブルとして、「生徒」を共有テーブルとして使用するモデル、(2)「クラス」または「クラブ」のいずれかから線形に

開始するモデル、(3)「生徒」を単一の基底テーブルとして、「クラス」と「クラブ」をダウンストリームテーブルとして使用するモデル。

この特定の例では、これらのテーブル、データ、またはモデルには、実際に複数の基底テーブルを必要とするものは何もありません。このモデルは、結合ロジックに重点を置くことができるようシンプルにまとめた例として使用しています。あるいは、議論が複雑になりすぎないようにするために単に除外していた、別の関連テーブル「教室」があると考えることもできます。



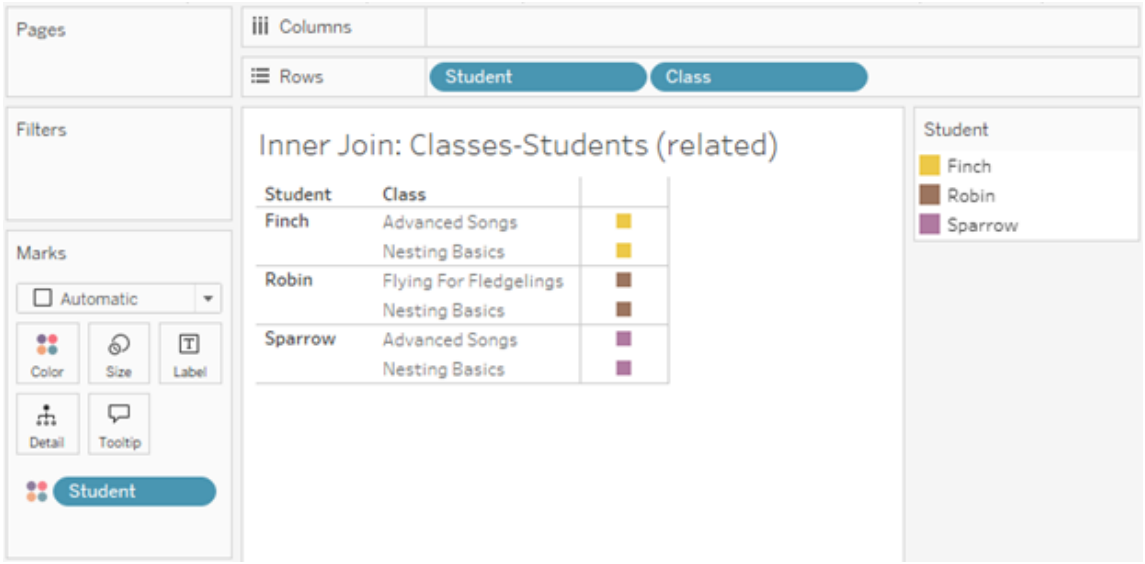
ただし、ベストプラクティスとして、データに必要な場合にのみマルチファクト関係モデルを使用してください。

関連ディメンションで内部結合を使用する

関連するディメンションは内部結合されます。内部結合では、両方のテーブル間で共有されていないディメンション値がすべてドロップされます。

- Tableau は、メジャーバリューが失われないようにするために追加のロジックを使用します。このセクションでは、ディメンションのみを使用して、Tableau が関連するディメンションに内部結合を適用する方法の基本を説明します。

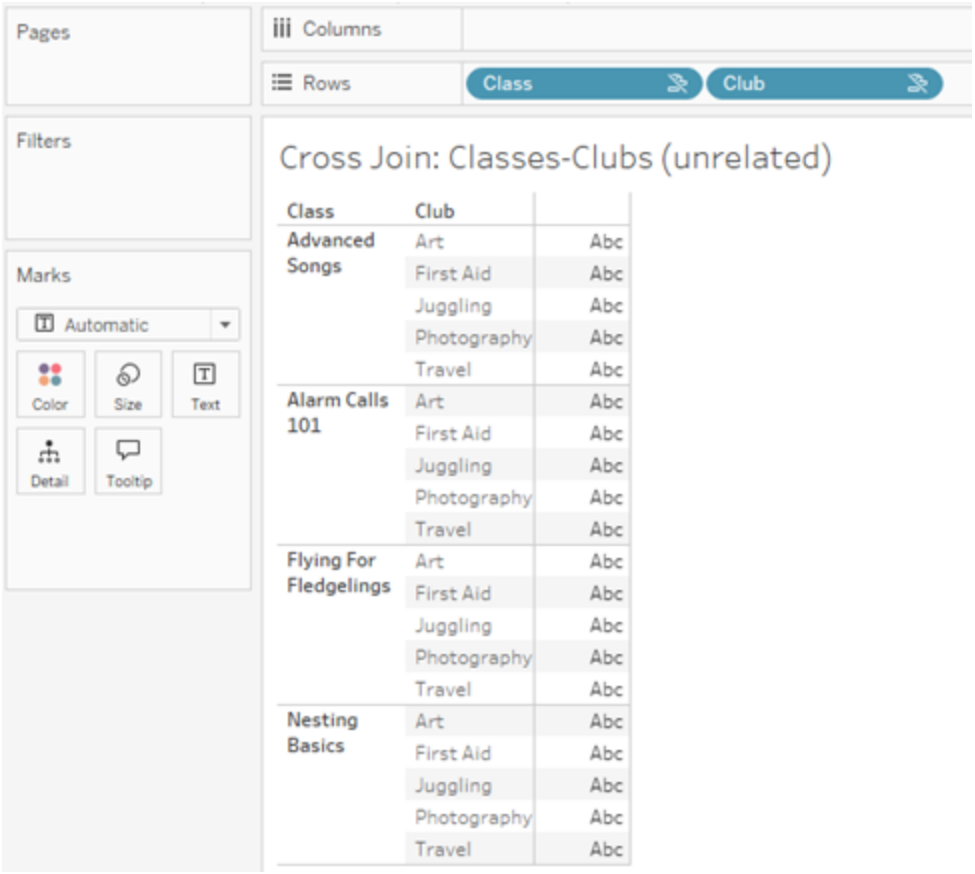
次の例は、関連するディメンションがデータ内に存在する行のみを返す方法を示しています。「警戒音101」クラスには生徒がいないため、結果には表示されません。「コウカンチョウ」と「カラス」はどのクラスにも所属していないため、結果には表示されません。



関連のないディメンションでクロス結合を使用する

関連のないディメンション (つなぎ合わせるディメンションなしで単独で存在するディメンション) は、クロス結合されます。

クロス結合では、結果の組み合わせがデータ内に実際に存在しない場合でも、1 つのディメンションのすべての値が他のディメンションのすべての値と結合されます。この例では、クロス結合により、Class と Club の可能な組み合わせごとに行が追加されます。



分析中にクロス結合が発生しているかどうかを認識することが重要です。クロス結合の結果テーブルには「高度な歌 + 応急処置」の行がありますが、実際にこのアクティビティの組み合わせに参加している生徒はいません (次のセクションのつなぎ合わせの例でこの事実を確認します)。

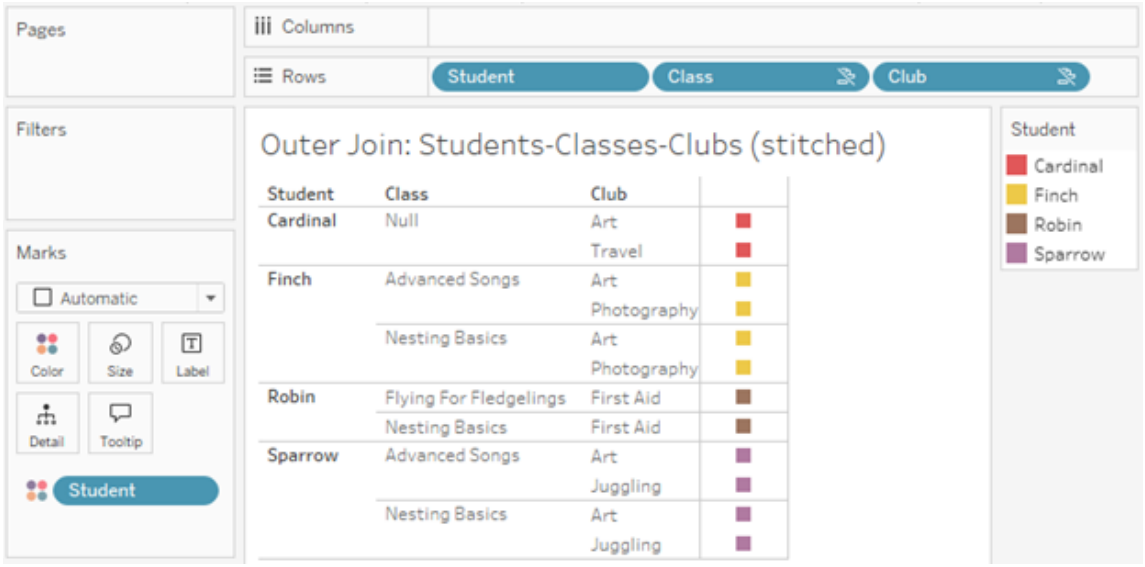
すべてのクロス結合結果がデータに基づいているわけではないことを認識するのが重要なのはなぜでしょうか? クラスやクラブ活動のスケジュールを、生徒のスケジュールとぶつからないように作成しようとしていると想像してください。「高度な歌」と「応急処置」には生徒がいないので、この結果を無視して、そのクラスとクラブを同時にスケジュールすることができます。クロス結合は、データ内に実際に存在する値の組み合わせを表すものではありません。

さらに、カーディナリティが高い場合 (一意の値の数が多い場合) のクロス結合はパフォーマンスに影響を与える可能性があります。連絡先にあるすべての電話番号とすべてのメール アドレスをクロス結合することを想像してみてください。組み合わせが爆発的に増え、操作にかかるコストも上昇します。

つなぎ合わされたディメンションで外部結合を使用する

つなぎ合わせるディメンションが存在する場合、関連のないディメンションは外部結合されます。

この例では、「クラス」テーブルと「クラブ」テーブルの両方が共有「生徒」テーブルに関連付けられていますが、相互に関連付けられていないため、「クラス」フィールドと「クラブ」フィールドには関連がありません。「生徒」ディメンションを追加すると、Tableau は分析で「クラス」のどの値と「クラブ」のどの値を並置する必要があるかを認識することができます。この外部結合の動作を*「つなぎ合わせ」*と呼びます。



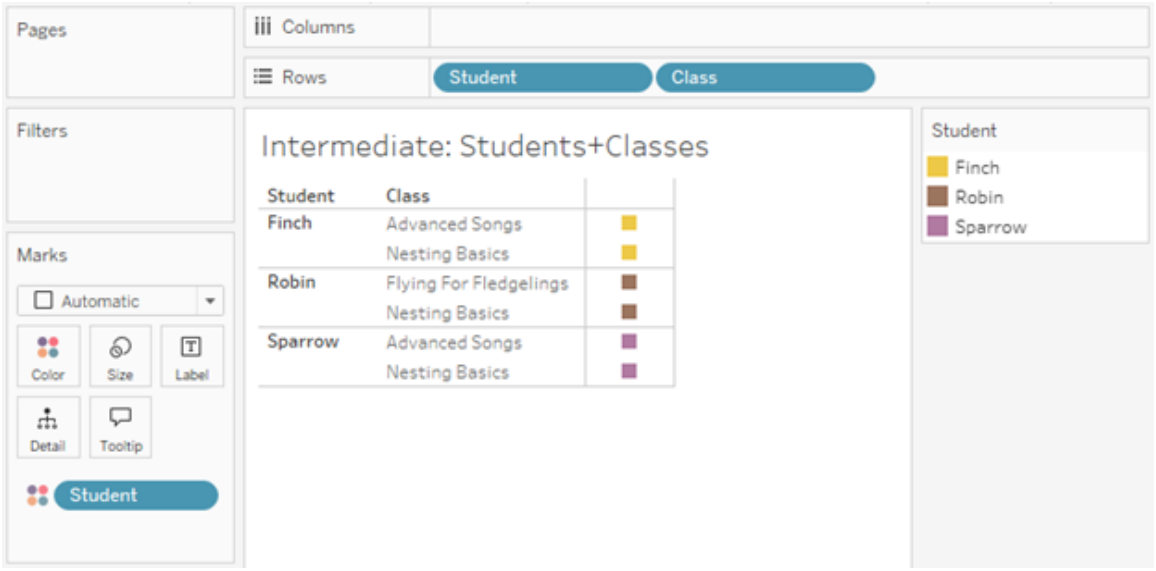
つなぎ合わせは、全体的な結果を得るための中間的な結果があるという点で、データブレンドに似ています。ただし、ブレンドとは異なり、つなぎ合わせは左結合ではなく外部結合であり、どちらの側からも値がドロップされることはありません。すべてが 1 つのデータソースである場合、プライマリデータソースまたはセカンダリデータソースの概念はないため、関連のない両方のフィールドに同等の優先順位が付けられます。

中間結果が外部結合される

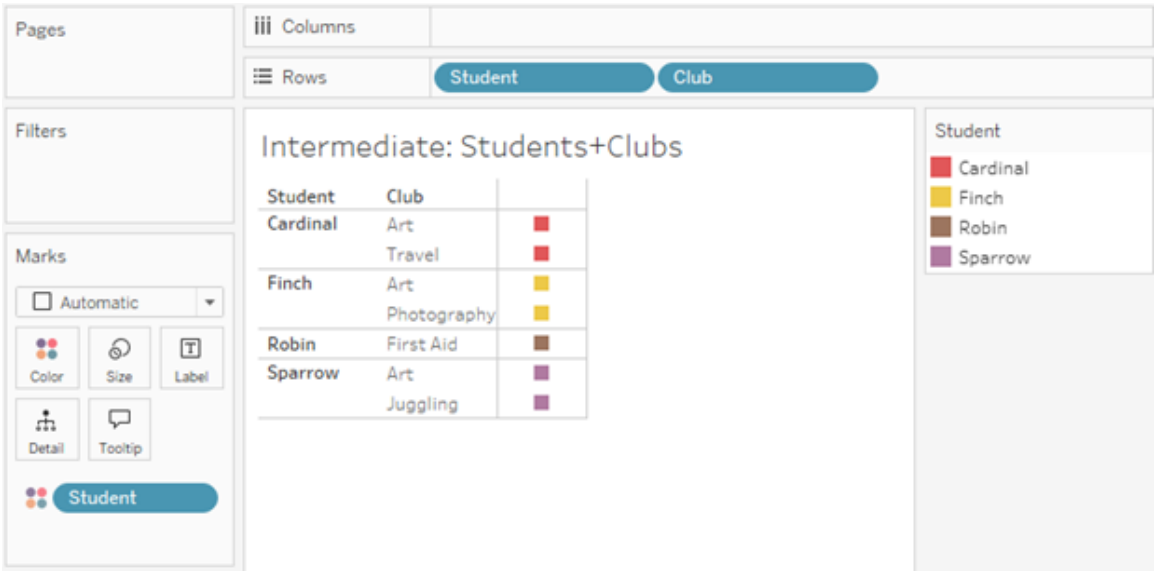
つなぎ合わされたフィールドの外部結合には何が含まれますか? つなぎ合わせるフィールドと、関連のない各フィールドに対して中間的な内部結合が順番に計算され、それらの中間的な結果が、つなぎ合わせるディメンションの値に基づいて外部結合されます。

例

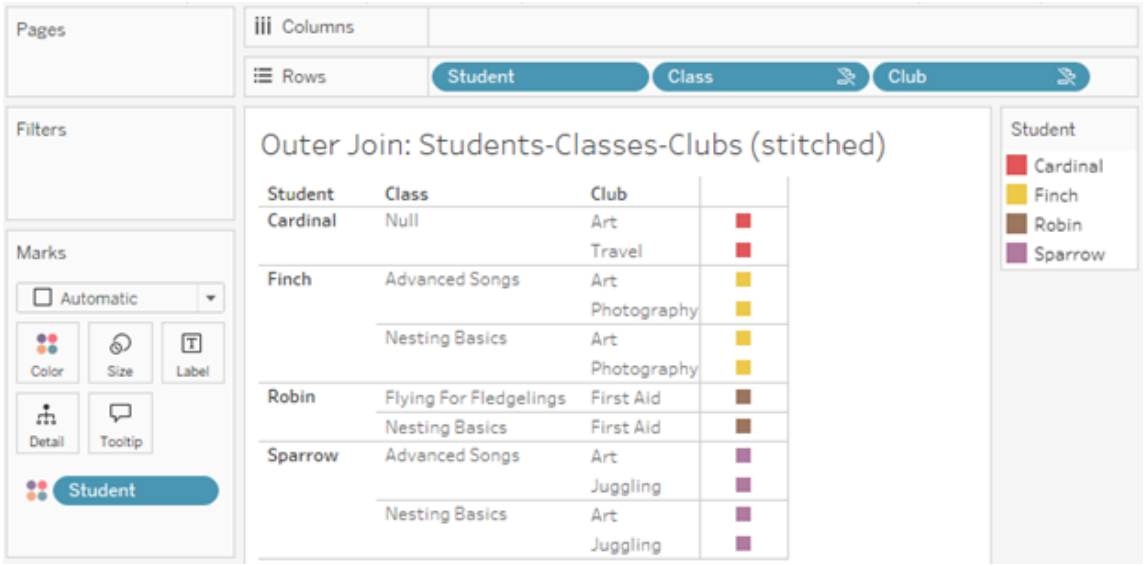
「生徒」と「クラス」の内部結合...



...「生徒」と「クラブ」の内部結合...



...「生徒」で外部結合されます。



メジャーを保持するための追加の結合

ディメンションの結合ロジックに加えて、メジャーでは追加の結合を導入できます。Tableau で関係が初めて導入されたとき、中核となる原則の 1 つは、メジャー バリューが失われないことでした。これは、マルチファクト関係データモデルでも維持されます。

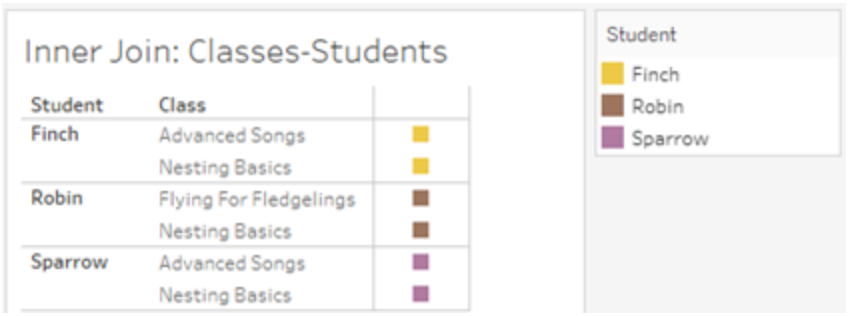
重要な詳細は次のとおりです。

- メジャー バリューは関連するディメンションによってのみ分類されます。
- 関連のないディメンションのメジャー バリューは繰り返されます。
- ディメンションのみの Viz では削除されるディメンション値は、関連するメジャー バリューが関連付けられている場合は返されることがあります。

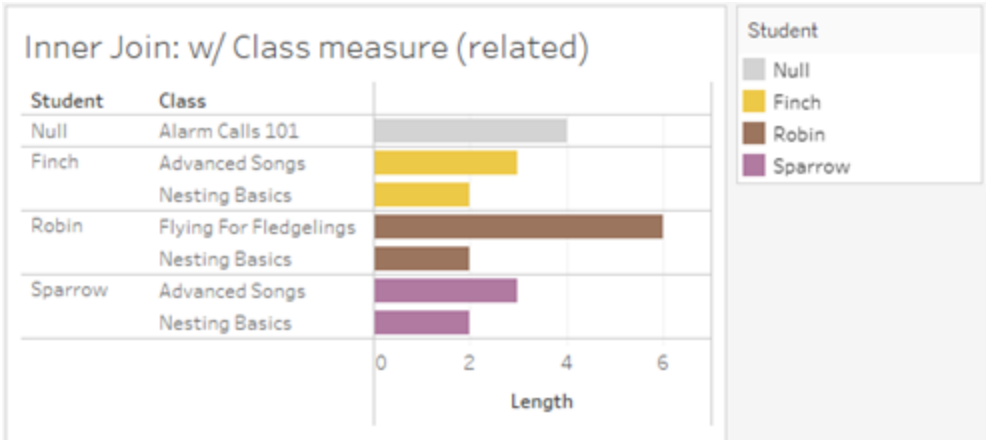
注: メジャーは集計であり、Viz 内のディメンションの組み合わせによって設定された詳細レベル (粒度) で計算されることに注意してください。これは、メジャーがディメンションごとに*分類*されることを意味します。メジャーがディメンションなしで使用される場合、それは*テーブル スコープ*であると言われます。つまり、メジャーの値は完全に集計された値になります。Viz でディメンションを使用するとすぐに、メジャーはディメンション値に基づいてより細かく分類されます。したがって、分析におけるメジャーの値は、ディメンションのコンテキストによって異なります。

関連するメジャー

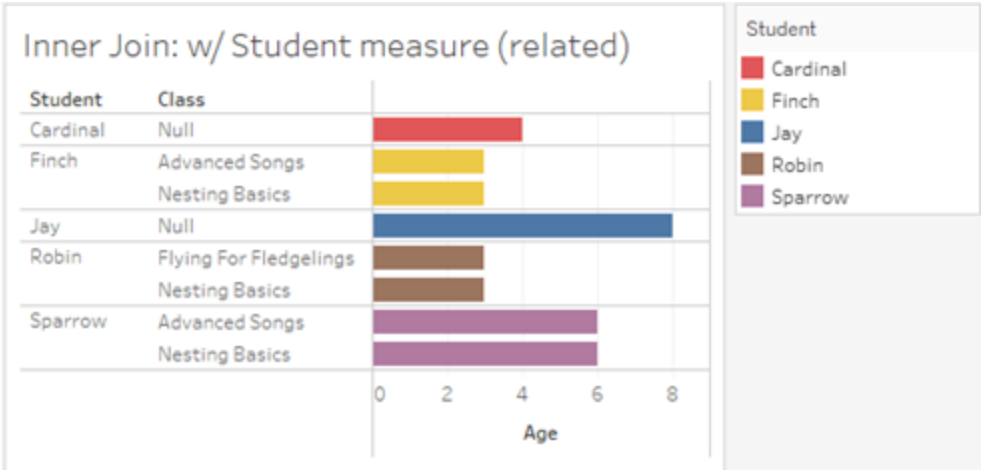
関連するディメンション**「生徒」**と**「クラス」**の内部結合に対して返されるディメンション値のサブセットを検討します。「生徒」の値には「スズメ」、「ヒメドリ」、「コマドリ」の *3* つがあり、「クラス」の値には「高度な歌」、「巣作りの基本」、「ひな鳥の飛行」の *3* つがあります。



「クラス」テーブルから**「長さ」**メジャーを追加すると、*4* つのクラスすべてが表示され、「生徒」には NULL 値があることがわかります。各クラスの**「長さ」**は、**「クラス」**のレベルで表示されます。



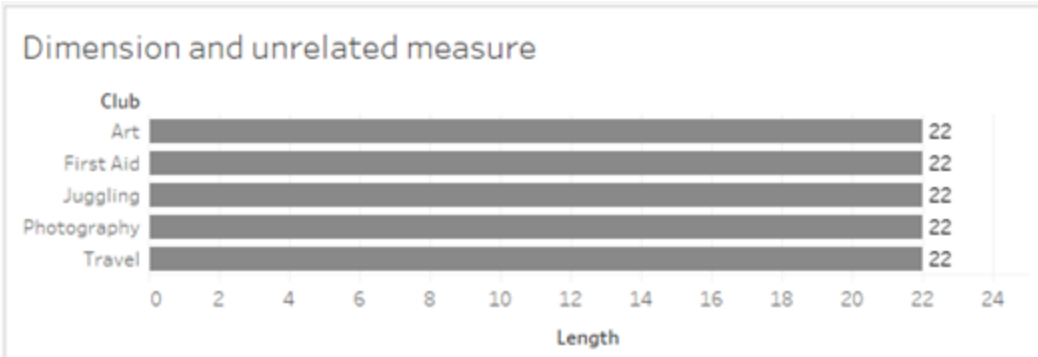
代わりに、「生徒」テーブルから**「年齢」**メジャーを追加すると、*5* 人の生徒全員が表示され、「クラス」に 2 つの NULl 値があることがわかります。結果には、クラスにいない生徒も含め、すべての生徒が保存されます。各生徒の**「年齢」**が、**「生徒」**レベルで表示されます。



関連のないメジャー

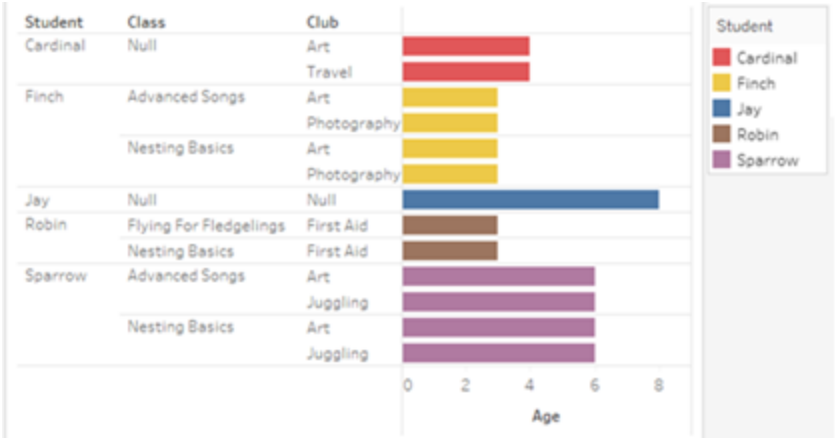
関連のないディメンション値に対してメジャー バリューが繰り返されます。

「クラス」テーブルの**「長さ」**メジャーおよび関連のない**「クラブ」**ディメンションを見ると、メジャーがテーブル スコープになっており、**「クラブ」**のすべてのディメンション値全体で繰り返されています。



つなぎ合わせるディメンションが存在する場合、メジャーを分類したり繰り返したりすることができます。

この**「年齢」**というメジャーは「生徒」テーブルからのもので、生徒のレベルまで分類されます。**「クラス」**と**「クラブ」**のディメンションに基づいて生徒が繰り返されるたびに、**「年齢」**の値が繰り返されます。



トラブルシューティング

マルチファクト関係データモデルを扱う際の考慮事項

テーブルごとの抽出フィルター

マルチファクト関係データモデル抽出のすべての抽出フィルターはテーブルごとに行われます (全体的ではありません)。このため、ライブ接続と抽出接続ではフィルタリング結果が異なる場合があります。

行レベルの計算

行レベルの計算では、同じアップストリームの基底テーブルを共有するフィールドのみを参照できます。つまり、行レベルの計算をツリー全体で実行することはできません。

組み合わせフィールド

組み合わせフィールド内のすべてのフィールドは、アップストリーム テーブルを共有する必要があります。つまり、異なるツリーにあるフィールドを使用して組み合わせフィールドを作成することはできません。

セット

セットは、同じアップストリームの基底テーブルを共有するフィールドを含む定義を使用する場合にのみ作成できます。ただし、Viz では、セットの定義に使用されるフィールドとは関連のないフィールドによってマークが定義されている場合、そのマークから[セットに追加] オプションを使用できる場合

があります。[セットに追加] を選択すると、Tableau は関連するフィールドのみをセット定義に追加します。これは、単一の基底テーブル データ ソースでの [セットに追加] の動作とは異なります。この [セットに追加] では、マークを定義するすべてのものが追加されます。

INCLUDE 詳細レベルの式の検証

関連のないフィールド間では、INCLUDE LOD 式を評価することはできません。フィールド間の関連性はシートごとに評価されるため、データペインまたは計算エディターでは有効な LOD 式であっても、特定の Viz のコンテキストでは無効になる可能性があります (関連のないディメンションが存在する場合)。これが起こると、LOD ピルが赤くなります。LOD 式を更新して、関連のないフィールドの競合を削除するか、Viz の構造を変更するか、あるいは Viz から LOD 式を削除することができます。

パブリッシュされたデータ ソースの更新

ベストプラクティスとして、接続されているすべてのワークブックで新しいデータ モデルが必要であるわけではない場合に、既存のパブリッシュされたデータ ソースをマルチファクト関係データ モデルに変更する予定があるときは、既存のパスリッシュされたデータ ソースのコピーを作成します。すべてのワークブックに新しいテーブルが必要となる場合以外は、データ ソースの既存のバージョンを更新しないでください。変更したデータ ソースを新しいデータ ソースとしてパブリッシュし、そこから新しいワークブックを作成します。これにより、既存のワークブックが必要としていないときに、データ サーバーの代わりに VDS を使用するように変換されることがなくなり、パフォーマンスが低下する可能性を防ぐことができます。

解決された問題

解決された問題	修正日
抽出 *ローカル データ ソース (ワークブック内):* マルチファクト関係データ ソースを抽出しようとすると、「そのようなテーブルはありません」というエラーが表示されます。 *パブリッシュされたデータ ソース:* パブリッシュされたマルチファクト関係 データ ソースの抽出は成功したよ	• Tableau Cloud: 7 月中旬のアップデートで解決済み これは、public.tableau.com にも適用されます。 • Tableau Desktop: 2024 年 7 月 24 日 リリースのメンテナンス リリース 2024.2.1 で解決済み • Tableau Server: 2024 年 7 月

うに見えますが、フィールド値が入れ替わっている可能性があります。

EXCLUDE 詳細レベル表現

関連のないフィールドが存在する場合は、INCLUDE LOD のみを検証する必要があります。ただし、EXCLUDE LOD も同じ条件下で誤って「無効」とマークされる可能性もあります。

ネストされたユーザー計算

ネストされたユーザー計算は、マルチファクト関係データモデルを持つパブリッシュされたデータソースでは使用できません。

24 日 リリースのメンテナンス リリース 2024.2.1 で解決済み

これらの問題が Tableau Desktop または Tableau Server で引き続き発生する場合は、2024 年 7 月 24 日以降のバージョンにアップグレードしてください。

2024.2 での既知の問題

複数のマークカードによる関連性指標

行シェルフまたは列シェルフに複数のメジャーを含む Viz が構築されている場合、各メジャーには独自のマークカードが割り当てられます。どのマークカードが開いているかによって、関連性指標 (関連のないアイコン、ツールヒントのテキスト、関連性に関する警告ダイアログボックス) を決定するために使用されるロジックから期待どおりの結果が得られない場合があります。ただし、Viz 自体は、各フィールドのペアの関連性に基づいて正しく計算されます。この動作については修正が予定されています。

BatchQueryProcessor

マルチファクト関係データモデルをサポートするには、BatchQueryProcessor を有効にする必要があります。これは予想される動作であり、現在修正は予定されていません。

Tableau Pulse

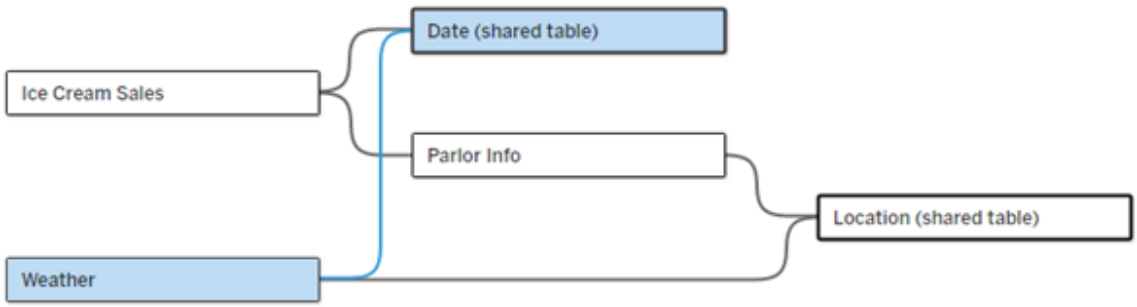
Tableau Pulse は、マルチファクト関係データモデルでは動作しない可能性があります。メトリクス定義を作成できなかったり、作成されたメトリクスが空白になったりする場合があります。これは予期される動作ではありませんが、現時点では修正の予定はありません。

マルチファクト関係モデルを使用するとき

マルチファクト関係モデルは、単一のデータソースに関連のないテーブルを追加し、ビジュアル分析の際に関連しているフィールドを使用して、コンテキストに基づいてテーブルを本質的につなぎ合わせることができるデータモデルです。ブレンディングとは異なり、データは単一のデータソース内に存在します。プライマリデータソースとセカンダリデータソースの概念は適用されず、左結合からデータが欠落することはありません。単一テーブルのデータモデルとは異なり、複数の基底テーブルは、共有してるテーブルに関して独自のコンテキストを維持します。マルチファクト関係データモデルを使用すると、Tableau でマルチファクト分析を実行するためのオプションがさらに増えます。

天気とアイスクリームの売上は、傾向がどのように連動して変化するかを分析したいとします。天気とアイスクリームの売上は、どちらも特定の日付と特定の場所で発生しますが、アイスクリームの売上と天気の間には直接的な関係はありません。これらは、日付と場所という共通の概念には関連しているものの、関連のないデータです。

この問題は、マルチファクト関係モデルを作成するのに役立ちます。アイスクリームの売上と天気はそれぞれ基底テーブルとして追加でき、共有テーブルである日付と場所に関連付けることができます。



関連のない 2 つのテーブル (「アイスクリームの売上」と「天気」) と、2 つの共有テーブル (「日付」と「場所」) を含む複数の基底テーブル データモデル。「アイスクリームの売上」と「場所」の間には、「パーラー情報」という中間テーブルがあります。

関連のないテーブルをモデル化する機能を設けた理由

分析では、直接的な関係はないものの、同じ共通の情報 (たとえば、日付や場所) に関連しているデータのテーブルを組み合わせて扱うことがよくあります。マルチファクト関係モデルでは、関連性の概念と、関連のない複数の基底テーブルを使用してデータモデルを構築する機能を導入することで、緩やかなセマンティック結合をサポートします。

- セマンティック結合とは、データがどれだけ密接に結合されているかを表すために使用される用語です。*結合*または*ユニオン*は、密接なセマンティック結合であり、複数のテーブルを 1 つの新しい物理テーブルにまとめ、単一のテーブルとして機能します。関係は、テーブルを論理的に結び付け、別々のテーブルとしての状態を明確に維持する、テーブル間の*疎結合*です。セマンティック結合の世界のさらに先にあるのが*データブレンディング*です。これは、別々のデータソースからの結果が、両者の間で共有される要素に基づいて視覚的に結合されるものです。*マルチファクト関係モデルは、セマンティック結合の世界でもほとんどデータブレンディング*に近いところにありますが、データソース間ではなく単一のデータソース内に作成します。

マルチファクト関係モデル (複数の基底テーブルを持つデータモデル) では、共有テーブルがモデル内に存在する限り、関連のないテーブルがモデル内に存在することも許されます。分析の間、共有テーブルのフィールドは、共通に持つ共有ディメンション (同じ場所や同じ日付に発生するなど) に基づいて、関連のないデータテーブルを「つなぎ合わせ」ます。各テーブルの粒度、つまりネイティブの詳細レベルの保持を含め、関係のすべての利点が維持されます。

基底テーブルが 1 つしかないデータモデルと同様に、Tableau は、Viz の構造に基づいて、バックグラウンドで使用する最適な結合タイプを決定します。ただし、マルチファクト関係モデルでは、結合オプションを拡張して外部結合とクロス結合を含むようになり、さまざまなレベルの関係を処理できます。詳細については、マルチファクト関係データモデルについてを参照してください。

名前の由来

マルチファクト関係という名前は、マルチファクト分析に由来しています。データウェアハウスモデルでは、データはディメンションテーブルに囲まれた中央のファクトテーブルに保存されます。その文脈では、*ファクト*は測定値またはメトリクスを指し、データに関する事実を捉えたデータの数値フィール

ド、つまりTableau のメジャーを指します。ディメンション テーブルには、これらのファクトに関する属性が含まれます。

ファクト テーブルに基づくスキーマは、ディメンション テーブルの構成に応じて、スター型またはスノーフレーク型として構造化されることがよくあります。ファクト テーブル全体で分析を行う必要がある場合、これをマルチファクト分析と呼びます。分析は、共有ディメンションまたは適合ディメンションと呼ばれる共通ディメンション テーブルのコンテキストで実行されます。Tableau では、関係を使用してこれらのデータ モデルを構築するため、この一連の機能をマルチファクト関係と名付けました。

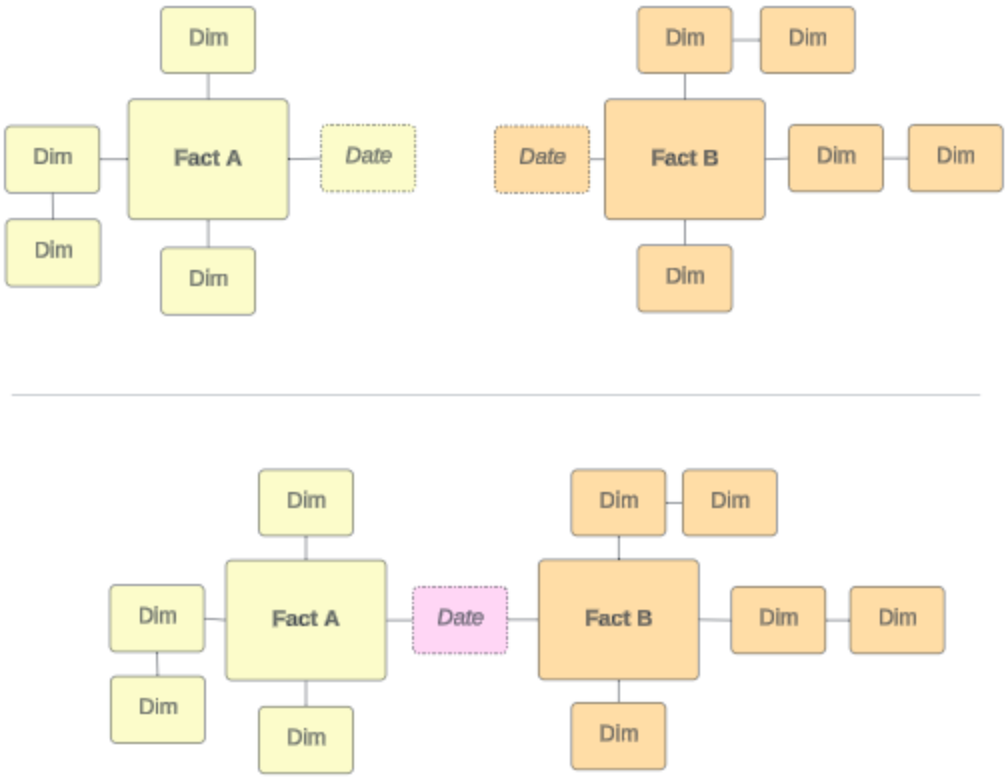
マルチファクト関係データ モデルを使用するとき

データが相互に関連するテーブルで構成されている場合、関係で構築された、基底テーブルが 1 つしかないデータ ソースを使用できます。データがさまざまな概念にまたがり、複数のファクト テーブルや、関連のない別々のコンテキストの形式を取る場合は、マルチファクト関係 モデルが必要になります。

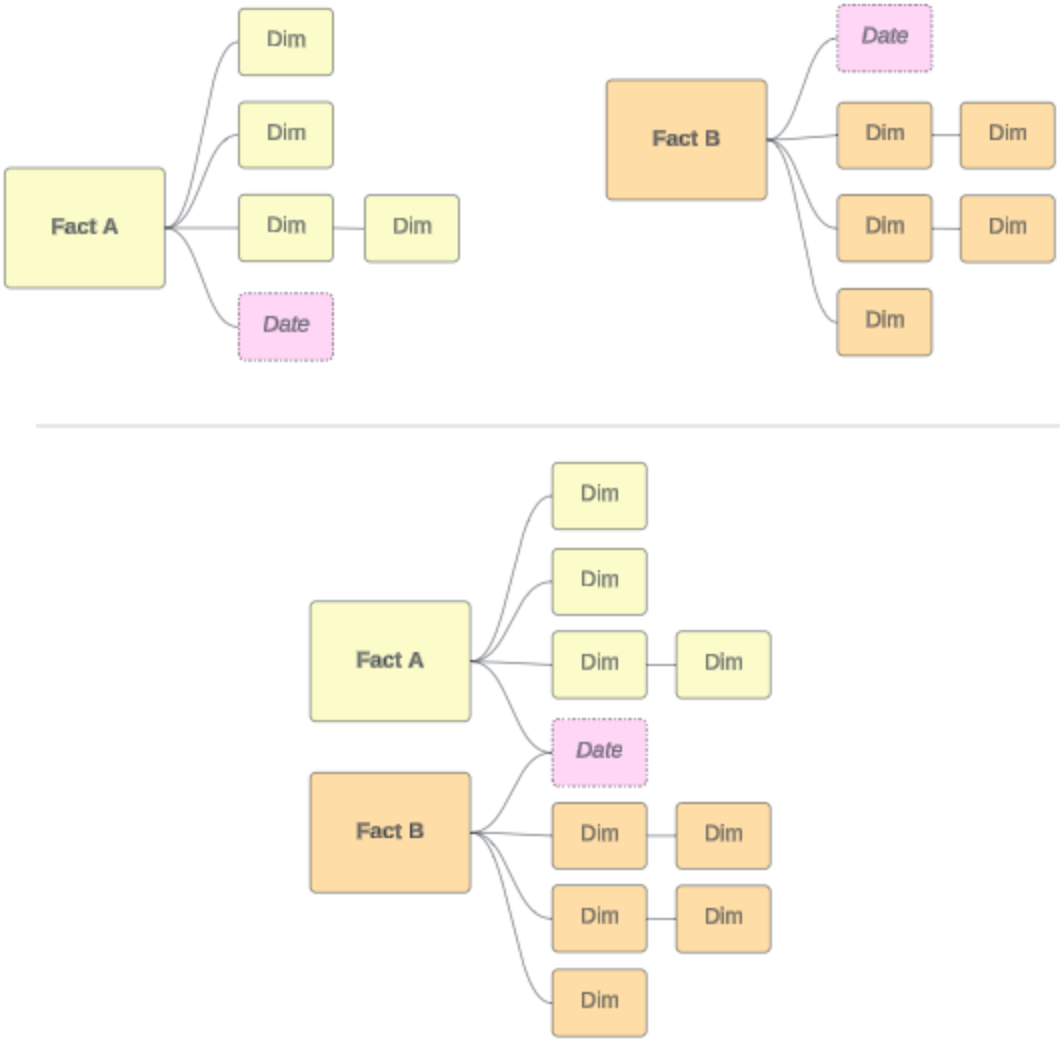
可能な限り、単一の基底テーブルを使用してデータ ソースを構築します。基底テーブルが 1 しかないデータ モデルでは、すべてのテーブルが関連しているため、関連性の度合いを考慮する必要はありません。データ モデル構造が必要とされる場合にのみ、マルチファクト関係を使用します。

マルチファクト分析

マルチファクト分析は、Tableau におけるマルチファクト関係の中心的な使用例です。この例では、「ファクト A」と「ファクト B」がテーブル「日付」を共有しています。



これを Tableau でモデル化するには、ファクトテーブルが基底テーブルになり、共有ディメンションテーブルに対して複数の入力関係が確立されます。



その他のシナリオ

ただし、マルチファクト関係データモデルは、マルチファクト分析のためだけのものではありません。Tableau では、ファクトテーブルまたはディメンション テーブルの厳密な定義は必要ありません。どのテーブルでも基底テーブルにすることができます (ただし、基底テーブルの特性に適合している必要があります)。基底テーブルが複数あるデータソースが役立つシナリオとしては、次のようなものがあります。

- **ステージの移動。**たとえば、学生のテーブルを共有する、出願、成績証明書、卒業生イベントの基底テーブル。

- **同じ出来事でも異なる文脈。**たとえば、医師または患者の文脈で設定したテーブルを共有する、診療予約と請求書の出来事についての基底テーブル。
- **相関する可能性のあるさまざまな領域。**たとえば、日付と場所のテーブルを共有して相関する、アイスクリームの売上と天気のように、以前はデータブレンディングで最適に処理されていたシナリオ。

マルチファクト関係がどのような場合に役に立つかについては、次の Tableau ブログ投稿「Tableau でマルチファクト関係を使用するタイミングとその方法 (英語)」を参照してください。

基底テーブルを特定する

マルチファクト関係モデルでは、方向性が重要です。つまり、モデルの左側にある基底テーブルがどれで、下流で共有されるテーブルがどれかによって、分析結果を返すために関係がどのように評価されるかが決まります。

請求書、予約、医師、患者の概念的な蝶ネクタイを考えてみましょう。

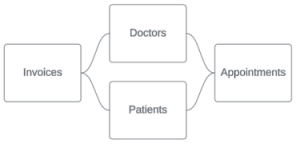
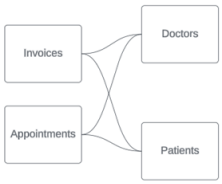
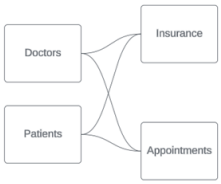


Tableau でデータモデルを構築する正しい方法は、請求書と予約を基底テーブルとして使用し、医師と患者を共有テーブルとして使用することです (医師と患者を基底テーブルとして使用しません)。

正解: 請求書と予約を基底テーブルとする



誤り: 医師と患者を基底テーブルとする



概念的には、患者 (または医師) は、予約のイベントと請求書のイベントをつなぎ合わせるエンティティです。

データモデルが逆の場合 (予約と請求書ではなく、医師と患者を基底テーブルとして使用するなど)、外部結合のつなぎ合わせる動作はあまり役に立ちません。分析では、テーブル スコープのメジャーやあいまいさが多数表示される可能性があります。予期していなかったあいまいな関連フィールドが見つかった場合は、基底テーブルとして使用しているテーブルを再評価し、データモデルを逆にする必要があるかどうかを確認します。

基底テーブルと共有テーブルの特徴

マルチファクト分析を実行する場合、ファクト テーブルが基底テーブルになり、共有ディメンションのテーブルが共有テーブルになります。Tableau では、ファクト テーブルとディメンション テーブルの特性を厳密に遵守する必要はありません。ただし、どのテーブルを基底テーブルにし、どのテーブルを共有テーブルにするかを識別するのに役立つ特定の属性があります。

基底テーブル	**共有テーブル**
データウェアハウス スキーマのファクト テーブル	データウェアハウス スキーマの共有または適合ディメンション テーブル
文脈や分析に特有	さまざまな文脈にまたがる一貫した概念
(フライト情報、エネルギー使用量)	(日付、場所)
重いメジャー	主にディメンション
より頻繁に更新される/トランザクション	より安定的/永続的
(医療予約、処方箋、バイタルサイン)	(医師、患者)
外部キーのフィールドがある	主キーのフィールドがある
イベント ベース	エンティティ ベース
(授業スケジュール、課題の成績)	(生徒、教室)

基底テーブルと共有テーブルの間に中間テーブルがある場合は、データモデルを根本的に変更しなくても、どちらを基底テーブルにするかを切り替えることができます (最初の例では、パーラー情報

やアイスクリームの売上など)。重要なのは、共有テーブルの上流にあるのはどのテーブルで、共有されているのはどのテーブルかということです。

代わりに追加の基底テーブルを試す

基底テーブルが 1 つしかないデータソースではなく、複数の基底テーブルを使用してマルチファクト関係モデルを構築する必要があるシナリオはさまざまです。

- 循環のあるデータソースを構築しようとしている場合は、ダウンストリーム テーブルを別の基底テーブルにする必要があります。
- 同じ関係句のセット(日付と場所など) に基づいて関連付けられている一連のテーブルがある場合、それらのディメンションを抽出して共有テーブルにする必要があります。
    - そうすると便利であるのは、複数の関係句がすべて真 (論理的には AND) であることが、それらのレコードにテーブルを関連付けるために必要であるからです。
    - 代わりに、一度に真になる可能性があるのは 1 つであるレコード(文脈的 OR) を分析する場合は、共有ディメンション テーブルを使用してデータモデルを設定することで、この柔軟性が実現されます。
- データソースをプライマリとセカンダリに区別せず、同等にブレンドしたい場合、ブレンドのデータソースと、共有テーブル内のリンク フィールドを結合するデータモデルを構築します。

マルチファクト関係データモデルのツールヒント

> **注**: 単一テーブルのデータソース、または基底テーブルが 1 つしかないデータソースでは、すべてのテーブルが関連しています。このページでは、基底テーブルが複数あるデータソースについて説明します。

フィールドレベルの関連性

基底テーブルが複数あるデータモデルでは、テーブルを相互に関連付ける(または関連付けない)ことを柔軟に実施できます。*テーブル*の関連性は、データモデルに基づいて一定です。ただし、Viz 内の *フィールド*の関連性は、アクティブなフィールド(つまり、ワークシートのシェルフでピルとして使用されているフィールド) によって決まります。単一の Viz レベルでは、Tableau はアクティブなフィールドをペアで評価し、それらがどのように相互に関連しているかを判断します。

フィールドに表示される関連のないアイコンは、そのフィールドがViz内の他のどのフィールドとも関連がないことを意味します。このアイコンは、シェルフ上のピルまたはデータペインに表示されることがあります。(場合によっては、関連ありアイコンが表示されることがあります。)アイコンの上にカーソルを置くと、詳細情報を示すツールヒントが開きます。フィールドの関連性のタイプに応じて、次のような異なるメッセージが表示されます。

- 関連のないディメンションとディメンションのペア
- 関連のないディメンションとメジャーのペア
- ディメンションのつなぎ合わせ
- 関連のないフィルターのペア

メッセージは、フィールドがシェルフで使用されているか、データペインで使用されているかによっても若干異なります。

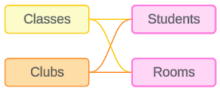
- **シェルフ上**: シェルフ上のピルのツールヒントには、Viz内のフィールドが互いにどのように関連しているか、またそれがTableauの計算結果にどのような影響を与えるかについての情報を提供します。
- **データペイン内**: データペインのフィールドのツールヒントには、そのフィールドがVizに追加された場合に何が起きる*可能性*があるかを示します。データペインのフィールドがViz内のどのフィールドとも関連がない場合は、それらを薄い灰色のテキストで強調表示しないようにすることもできます。

ヒント: Tableauでは、集計されたディメンション(ATTR(ディメンション)やMIN(ディメンション)など)はメジャーとして機能します。

データモデルを理解する

マルチファクト関係を含むデータモデルを操作するときは、モデルを定期的に参照すると有用です。[データモデルの表示]オプション(**[ワークシート]**メニュー > **[データモデルの表示]**)を使用して、シートのタブを表示中に参照できるデータモデルが含まれたダイアログを開きます。

このトピックの例は、2つの基底テーブル(「クラス」と「クラブ」)と2つの共有テーブル(「生徒」と「ルーム」)を含むデータモデルを基にしています。



クラス　クラブ　生徒　ルーム

Classes
7 rows 4 fields

Abc Classes Class
Nesting Basics
Advanced Songs
Flying For Fledgelings
Nesting Basics
Advanced Songs
Nesting Basics
Alarm Calls 101

Clubs
7 rows 4 fields

Abc Clubs Club	Abc Clubs Room
Photography	Cedar
Travel	Cedar
Juggling	Outsi
Art	Cedar
Art	Cedar
Art	Cedar
First Aid	Fir

Students
5 rows 3 fields

Abc Students Bus Rider	Abc Stude Stud
yes	Fincl
yes	Card
no	Spar
yes	Robi
no	Jay
Sparrow	10
Robin	0

Rooms
5 rows 2 fields

Abc Rooms Room	# Rooms Branches
Cedar	20
Alder	37
Maple	44
Fir	14
Madrone	23

フィールド:

- **クラス**:「巣作りの基本」、「高度な歌」、「ひな鳥の飛行」、「警戒音 101」の値を持つディメンション
- **ルーム**:「ルーム」テーブルに関連付けるために使用されるディメンション
- **生徒**:「生徒」テーブルに関連付けるため

フィールド:

- **クラブ**:「写真」、「旅行」、「ジャグリング」、「美術」、「応急処置」の値を持つディメンション
- **ルーム**:「ルーム」テーブルに関連付けるために使用されるディメンション
- **生徒**:「生徒」テーブルに関連付けるため

フィールド:

- **バス利用者**:「はい」または「いいえ」の値を持つディメンション
- **生徒**:「スズメ」、「コウカンチョウ」、「ヒメドリ」、「コマドリ」、「カラス」の値を持つディメンション基底テーブルと関連付けるために使用

フィールド:

- **ルーム**:「ヒマラヤスギ」、「ハンノキ」、「カエデ」、「モミ」、「マドローネ」の値を持つディメンション
- **枝**: メジャー

に使用されるディメンション
- **長さ**: メジャー

に使用されるディメンション
- **会費**: メジャー

- **年齢**: メジャー

関連のないディメンションとディメンションのペア

関連のないディメンションはクロス結合される可能性があり、その結果、ヘッダー全体のディメンション要素の組み合わせが生じ、参照元テーブル内の実際のデータの組み合わせが反映されない可能性があります。

関連のないディメンションとディメンションのペアのメッセージは次のとおりです。

- **シェルフ上**: 関連のないディメンションには、値のすべての可能な組み合わせが表示されます。関連なし: <*ディメンションのリスト*>
- **データペイン内**: 使用した場合、このディメンションは、関連のないディメンションとのすべての可能な値の組み合わせを表示します。<*ディメンションのリスト*>
- **データペイン内でグレー表示**: このディメンションは、Viz 内のどのディメンションとも関連がありません。このディメンションを使用した場合、ディメンションには、関連のない他のディメンションと値のすべての可能な組み合わせが表示されます。

関連のないフィルターのメッセージは次のとおりです。

- **Viz 内**: このフィールドは、関連のないフィルターによってフィルタリングされません。<*filters*>
- **データペイン内**: 使用した場合、このフィールドは、関連のないフィルターによって無視されます。<*フィルター*>

クロス結合の操作には高額なコストがかかり、パフォーマンスに悪影響を与える可能性があります。このため、Tableauでは関連のないディメンションをViz に追加した場合に関連性に関する警告ダイアログ ボックスが表示されます。

ディメンションのつなぎ合わせ

関連のないディメンションのみを視覚化することには分析上適切な理由がありますが、複数の基底テーブルを使用した分析を行うときは、最も一般的な方法として、つなぎ合わせるディメンションを追加して使用します。つなぎ合わせるディメンションが存在する場合、関連のないディメンションはクロス結合されなくなり、代わりに外部結合されます。外部結合では依然としてNULL が発生する可能性がありますが、ディメンション要素のヘッダーは、すべての可能な組み合わせから、外部

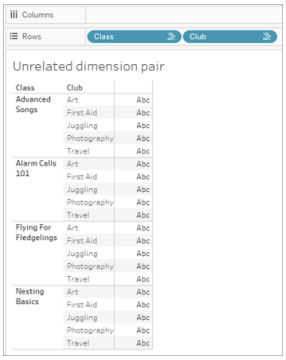
結合の少なくとも 1 つの側に関連する組み合わせにまで絞り込まれます。また、クロス結合の場合と同程度のパフォーマンスへの悪影響が発生する可能性もありません。関連なしアイコンを必要とする関連性に関するその他の問題がない場合は、つなぎ合わせるディメンションによって、関連ありアイコンが表示されます。

つなぎ合わせるディメンションのメッセージは次のとおりです。

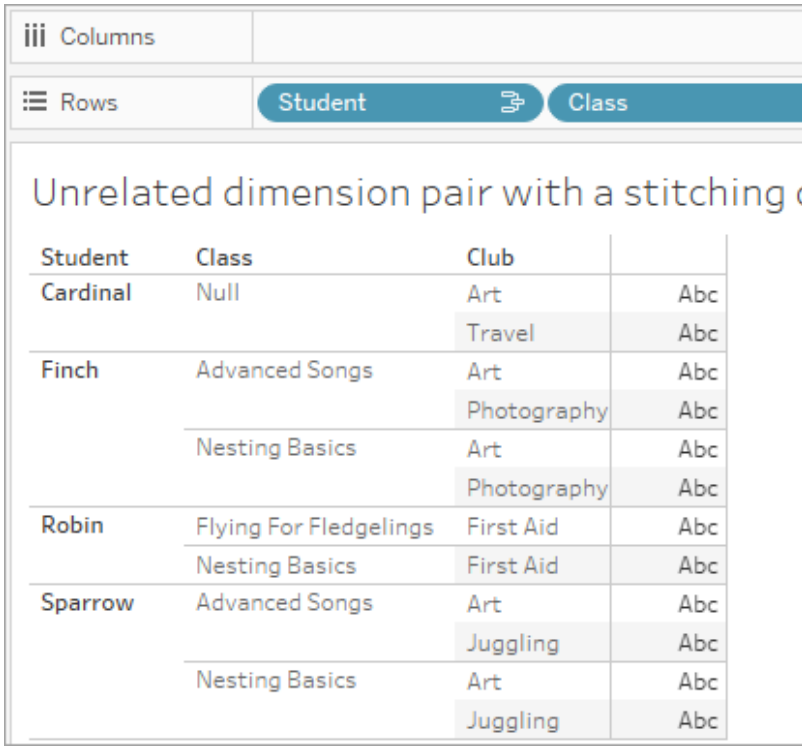
- **シェルフ上**: このディメンションは、次のフィールドをつなぎ合わせます。<*フィールドのリスト*>
- **データペイン内**: 使用した場合、このディメンションは、次のフィールドをつなぎ合わせます。<*フィールドのリスト*>
- **データペイン内でグレー表示**: *適用されません。つなぎ合わせは Viz 内でのみ発生します*

関連のないディメンションとつなぎ合わされたディメンションを比較する

関連のない場合: クロス結合	つなぎ合わされた場合: 中間内部結合の外部結合



「クラス」と「クラブ」のクロス結合を示す Viz。行には「高度な歌」「警戒音 101」「ひな鳥の飛行」「巣作りの基礎」と「美術」「応急処置」「ジャグリング」「写真」のすべての組み合わせが表示されています。



「生徒」と「クラス」の内部結合と「生徒」と「クラブ」の内部結合の外部結合の結果を示す viz。「クラス」と「クラブ」のすべての組み合わせが表示されるわけではなく、「クラス」のない「生徒」や「クラブ」の行もあります。

関連のないディメンションとメジャーのペア

メジャー バリューの計算方法とディメンションの関連性が重要な理由については、「メジャー バリューの計算方法についての補足説明」を参照してください。

メジャーのメッセージは次のとおりです。

- **シェルフ上**: このメジャーは、関連のないディメンションごとに分類することはできません。<*ディメンションのリスト*>
- **データ ペイン内**: 使用した場合、このメジャーは、関連のないディメンションごとに分類されません。<*ディメンションのリスト*>

- **データペイン内でグレー表示**: このメジャーは、Viz 内のどのディメンションとも関連がありません。メジャーを使用した場合、メジャーは分類されません。

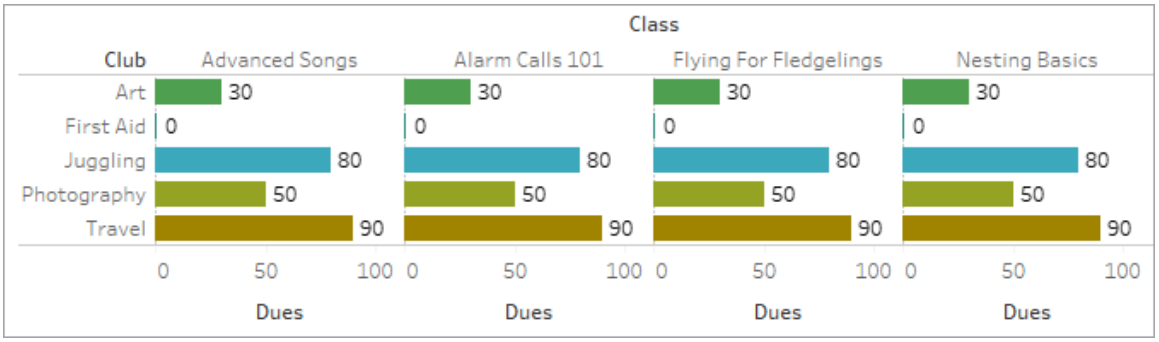
ディメンションのメッセージは次のとおりです。

- **シェルフ上**: このディメンションは、関連のないメジャーを分類できません。*<メジャーのリスト>*
- **データペイン内**: 使用した場合、このディメンションは、関連のないメジャーを分類しません。*<メジャーのリスト>*
- **データペイン内でグレー表示**: このディメンションは、Viz 内のどのメジャーとも関連がありません。このディメンションを使用した場合、ディメンションは メジャー バリューを分類しません。

関連のないフィルターのメッセージは次のとおりです。

- **Viz 内**: このフィールドは、関連のないフィルターによってフィルタリングされません。*<filters>*
- **データペイン内**: 使用した場合、このフィールドは、関連のないフィルターによって無視されます。*<フィルター>*

Viz の結果は、関連のないディメンションの値全体にわたってメジャーの値が繰り返されることになります。この動作は、LOD 式を使用して、Viz のネイティブの粒度とは異なる詳細レベルでメジャーの集計レベルを設定する場合と似ています。関連のないディメンションは、基本的にメジャーの集計値の計算から除外されます。



まだ関連付けられていない

潜在的な関係が存在するフィールドは、まだ関連付けられていないと見なされ、Viz 内で関連のないフィールドとして評価されます。まだ関連付けられていないフィールドの解決は、真に関連のないフィールド(つなぎ合わせのみ可能)の解決とは異なる方法で行われるため、ツールヒントには、関連がないと見なされる理由が表示されます。

ディメンションのメッセージは、ディメンションが別のディメンション、メジャー、またはフィルターにまだ関連付けられていないかどうかによって異なります。

- ディメンションとディメンションのペアの場合、ディメンションのメッセージは次のとおりです。
  - **シェルフ上**: 関連のないディメンションには、値のすべての可能な組み合わせが表示されます。このディメンションはまだ関連付けられていないため、次のディメンションとは関連のないディメンションとして扱われます。<*ディメンションのリスト*>
  - **データペイン内**: 使用した場合、このディメンションは、まだ次のディメンションに関連付けられていないため、これらのディメンションとは関連のないディメンションとして扱われます。<*ディメンションのリスト*>
- ディメンションとメジャーのペアの場合、ディメンションのメッセージは次のとおりです。
  - **シェルフ上**: このディメンションは、まだ次のメジャーと関連付けられていないため、これらのメジャーを分類できません。<*メジャーのリスト*>
  - **データペイン内**: 使用した場合、このディメンションは、まだ次のメジャーに関連付けられていないため、これらのメジャーを分類しません。<*メジャーのリスト*>
- ディメンションとフィルターのペアの場合、ディメンションのメッセージは次のとおりです。
  - **シェルフ上**: このフィールドは、まだ関連付けられていないため次のフィルターによって無視されます。<*フィルターのリスト*>
  - **データペイン内**: 使用した場合、このフィールドは、まだ関連付けられていないため次の関連のないフィルターによって無視されます。<*フィルターのリスト*>

メジャーのメッセージは、メジャーがディメンションまたはフィルターにまだ関連付けられていないかどうかによって異なります。

- メジャーとディメンションのペアの場合、メジャーのメッセージは次のとおりです。
  - **シェルフ上**: このメジャーは、まだ次のディメンションに関連付けられていないため、これらのディメンションごとに分類できません。<*ディメンションのリスト*>
  - **データペイン内**: 使用した場合、このメジャーは、まだ次のディメンションに関連付けられていないため、これらのディメンションごとに分類されません。<*ディメンションのリスト*>
- メジャーとフィルターのペアの場合、メジャーのメッセージは次のとおりです。
  - **シェルフ上**: このフィールドは、まだ次のフィルターに関連付けられていないため、これらのフィルターによって無視されます。<*フィルターのリスト*>
  - **データペイン内**: 使用した場合、このフィールドは、まだ次のフィルターに関連付けられていないため、関連のないこれらのフィルターによって無視されます。<*フィルターのリスト*>

メジャー バリューの計算方法とディメンションの関連性が重要な理由については、「メジャー バリューの計算方法についての補足説明」を参照してください。

あいまいに関連する

潜在的にアクティブな関係が複数存在するフィールドは、あいまいに関連していると見なされ、Viz 内で関連のないフィールドとして評価されます。あいまいに関連しているフィールドの解決は、真に関連のないフィールド(つなぎ合わせのみ可能)の解決とは異なる方法で行われるため、ツールヒントには、関連がないと見なされる理由が表示されます。

ディメンションのメッセージは次のとおりです。

- **シェルフ上**: このディメンションは、次のメジャーへの関連付けがあいまいなため、これらのメジャーを分類できません。<*メジャーのリスト*>
- **データ ペイン内**: 使用した場合、このディメンションは、次のメジャーへの関連付けがあいまいなため、これらのメジャーを分類しません。<*メジャーのリスト*>

メジャーのメッセージは、あいまいさがディメンションまたはフィルターに付随するかどうかによって異なります。

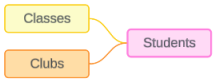
- メジャーとディメンションのペアの場合、メジャーのメッセージは次のとおりです。
  - **シェルフ上**: このメジャーは、次のディメンションへの関連付けがあいまいなため、これらのディメンションごとに分類できません。<*ディメンションのリスト*>
  - **データ ペイン内**: 使用した場合、このメジャーは、次のディメンションへの関連付けがあいまいなため、これらのディメンションごとに分類されません。<*ディメンションのリスト*>
- メジャーとフィルターのペアの場合、メジャーのメッセージは次のとおりです。
  - **シェルフ上**: このメジャーは、次のフィルターへの関連付けがあいまいなため、これらのフィルターによって無視されます。<*フィルターのリスト*>
  - **データ ペイン内**: 使用した場合、このメジャーは、次のフィルターへの関連付けがあいまいなため、次のフィルターによって無視されます。<*フィルターのリスト*>

メジャー バリューの計算方法とディメンションの関連性が重要な理由については、「メジャー バリューの計算方法についての補足説明」を参照してください。

共有テーブルからのメジャー

共有テーブル(「学生」など)からのディメンションを使用すると、関連のないテーブル(「クラス」や「クラブ」など)のディメンションがつなぎ合わされます。しかし、ディメンションの代わりに「生徒」テーブ

ルからのメジャーを使用するとどうなるでしょうか? メジャーをつなぎ合わせることはできません。



関連のないディメンションが一緒に視覚化されている場合、メジャーをそれらのディメンションごとに同時に分類することはできません。(メジャー バリューの計算方法とディメンションの関連性が重要な理由については、「メジャー バリューの計算方法についての補足説明」を参照してください。)この場合、メジャーが任意のディメンションに個別に関連していたとしても、ディメンションの組み合わせとは関連がないメジャーとして扱います。

こうした種類のあいまいさを解決し、メジャーにテーブル スコープが設定されないようにするには、メジャーを集計するための明確な関係パスができるまで、関連のないディメンションをつなぎ合わせるか、1 つ以上のディメンションを削除します。

すべてのディメンションが常に互いに関連していない

関連のないディメンション間で共有されるメジャーのメッセージは次のとおりです。

- **シェルフ上**: このメジャーは、次のいずれかのディメンションごとでのみ一度に分類できます。<*ディメンションの箇条書きリスト*>
- **データ ペイン内**: 使用した場合、このメジャーは、次のいずれかのディメンションごとでのみ一度に分類できます。<*ディメンションの箇条書きリスト*>

各ディメンションのメッセージは次のとおりです。

- **シェルフ上**: このディメンションは、次のディメンションとの組み合わせでメジャー <*メジャー*> を分類できません。<*ディメンションの箇条書きリスト*>
- **データ ペイン内**: 使用した場合、このディメンションは、次のディメンションとの組み合わせでメジャー <*メジャー*> を分類できません。<*ディメンションの箇条書きリスト*>

一部のディメンションが互いに関連している

いくつかのディメンションが相互に関連している場合は、関連のないディメンションが存在しなくても、関連するディメンションによってメジャーが細分化される可能性があります。どのディメンションが相互に関連しているかどうかを知っておくと便利です。(例:関連するディメンション Dim1 および

Dim2 と、それらの両方と関連のないディメンション DimA を使用して Viz が作成されている場合、共有テーブルのメジャーは、3 つのディメンションすべてに同じように関連しているわけではありません。Dim1 と Dim2 は互いに競合しませんが、どちらも DimA と競合します。フィールドを削除して解決するには、DimA のみを削除するか、Dim1 と Dim2 の両方を削除します。)

この場合、ディメンションは、ツールヒントメッセージにグループとしてリストされます。

関連のないディメンションのグループ全体で共有されるメジャーのメッセージは次のとおりです。

- **シェルフ上**: このメジャーは、次のいずれかのディメンションのグループごとでのみ一度に分類できます。<*ディメンションの箇条書きリスト*>
- **データペイン内**: 使用した場合、このメジャーは、次のいずれかのディメンションのグループごとでのみ一度に分類できます。<*ディメンションの箇条書きリスト*>

各ディメンションのメッセージは次のとおりです。

- **シェルフ上**: このディメンションは、次のディメンションのグループとの組み合わせでメジャー <*メジャー*> を分類できません。<*ディメンションの箇条書きリスト*>
- **データペイン内**: 使用した場合、このディメンションは、次のディメンションのグループとの組み合わせでメジャー <*メジャー*> を分類できません。<*ディメンションの箇条書きリスト*>

関連するメジャー

メジャーが Viz 内の一部のディメンションに関連しているが、他のディメンションには関連していない場合、メジャーのツールヒントに、どのディメンションに関連しているかを説明する追加メッセージが表示されることがあります。この情報から、メジャーどのように集計されているかがわかります。このメッセージは、メジャーが Viz 内のディメンションにも関連していない場合にのみ表示されます。それ以外の場合は、標準的な動作として、メジャーは関連するメジャーの詳細レベルに集計されます。

メジャーのメッセージは次のとおりです。

- **シェルフ上**: このメジャーは、関連するディメンションの詳細レベルに集計されます。<*このメジャーが関連する viz 内のディメンション*>
- **データペイン内**: 使用した場合、このメジャーは、関連するディメンションの詳細レベルに集計されます。<*このメジャーが関連する viz 内のディメンション*>

このメッセージは、メジャー バリューが計算されるときに考慮されるディメンションを特定しやすくするためのものです。メジャー バリューの計算方法とディメンションの関連性が重要な理由については、「メジャー バリューの計算方法についての補足説明」を参照してください。

フィルター

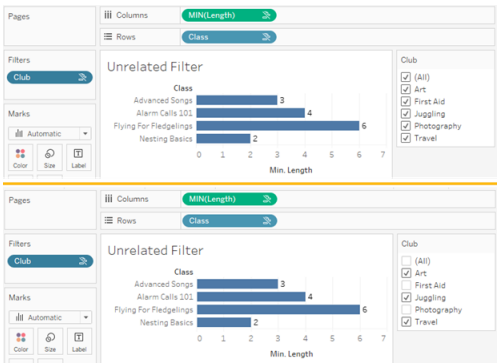
フィルター シェルフ上のフィールドの関連性も、Viz 内でアクティブな他のフィールドと比較して評価されます。

フィルターが Viz 内のどのフィールドとも関連がない場合、アイコンとツールヒントが表示されます。フィルターのフィールドと Viz 内のフィールドの両方にツールヒントがあります。

フィルターのメッセージは次のとおりです。

- **関連のないフィルター:** このフィルターは、関連のないフィールドに適用されません。<*フィールド*>
- **あいまいなフィルター**: このフィルターは、次のメジャーへの関連付けがあいまいなため、これらのメジャーを無視します。<*メジャーのリスト*>
- **まだ関連していないフィルター**: このフィルターは、次のフィールドにまだ関連付けられていないため、これらのフィールドを無視します。<*フィールドのリスト*>

フィルターの動作は、Viz 内の他のフィールドとの関連性によっても異なります。フィルターは、関連性のないフィールドの値には影響しません。フィルターが値なし (すべてを除外するか、何も含めない) に設定されていない限り、フィルターと関連性のないフィールドについては、Viz は変更されません。ただし、フィルター内のすべてのオプションの選択を解除すると、空の Viz が返されます。



関連のないフィルターがある Viz の 2 つのスクリーンショット。インタラクティブ フィルター コントロールでオプションの選択を解除しても Viz に影響がないことを示しています。

関連しているフィールドは期待どおりにフィルタリングされます。関連しているフィールドと関連のないフィールドの組み合わせを含むより複雑な Viz (つなぎ合わされたコンテキストなど) では、フィルターはフィルター フィールドに関連する値にのみ影響します。

メジャー バリューの計算方法についての補足説明

メジャーがディメンションに関連付けられていない場合、そのディメンション要素ごとにメジャーを分類することはできません (つまり、クラブにクラスの長さがない場合、クラブごとの平均クラスの長さを分類することはできません)。代わりに、メジャーはビュー内のディメンション要素のヘッダーとは異なるレベルで集計されます。

ディメンションとメジャーの用語

Tableau では、*メジャー*は*集計*であり、ビュー内の*ディメンション*によって設定される*粒度*まで集計されます。したがって、メジャーの値はディメンションのコンテキストによって異なります。たとえば、「シリアルの箱の数」は、在庫総数を意味するのか、ブランドごとの箱数を意味するのかによって異なります。

*ディメンション*は通常、国やブランドなどのカテゴリフィールドです。Tableau では、ディメンションによってビューの粒度、つまり*詳細レベル*が設定されます。通常、いくつかのカテゴリの組み合わせごとにマークを使用してデータをグループ化します。ビューを構築するために使用するディメンションによって、マークの数が決まります。

*集計*は、データがどのように組み合わせられているかを指すものです。Tableau の規定の集計は SUM です。集計は、平均値、中央値、個体数、最小値など、さまざまなオプションに変更できます。*粒度*は、メジャーがどの程度細かく分類されているかを指し、関連ディメンションによって制御されます。メジャーの粒度が行レベル (つまり非集計) でない限り、その値を集計する必要があります。

例

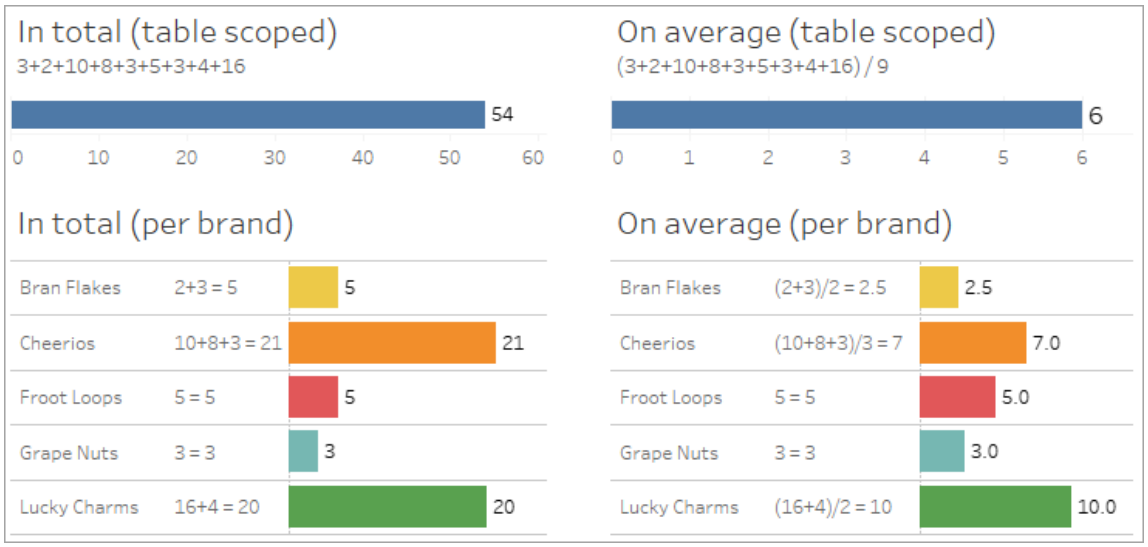
Raw data

	Mini	Normal	ValuePak
Bran Flakes		3	2
Cheerios	10	8	3
Froot Loops		5	
Grape Nuts		3	
Lucky Charms		4	16

「シリアルの箱の数」の値は何ですか?

この値は、集計タイプと、ディメンションによって設定される粒度によって異なります。

- 集計:
  - Sum (合計)
  - Average
- 粒度:
  - テーブルスコープ/完全集計 (図の例では青いバー)
  - **ブランド**ディメンションごとに分類 (図の例では色付きのバー)

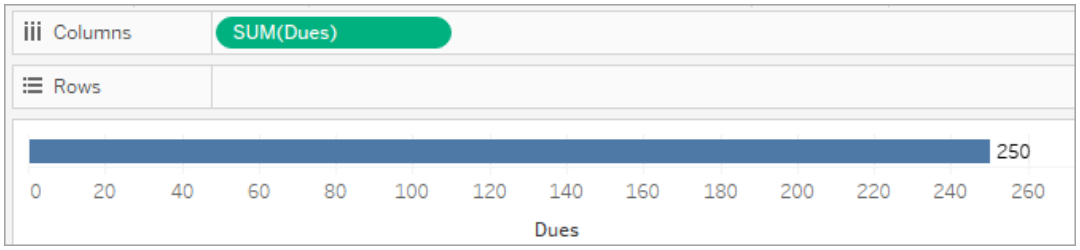


メジャーの値はディメンション要素に追随します

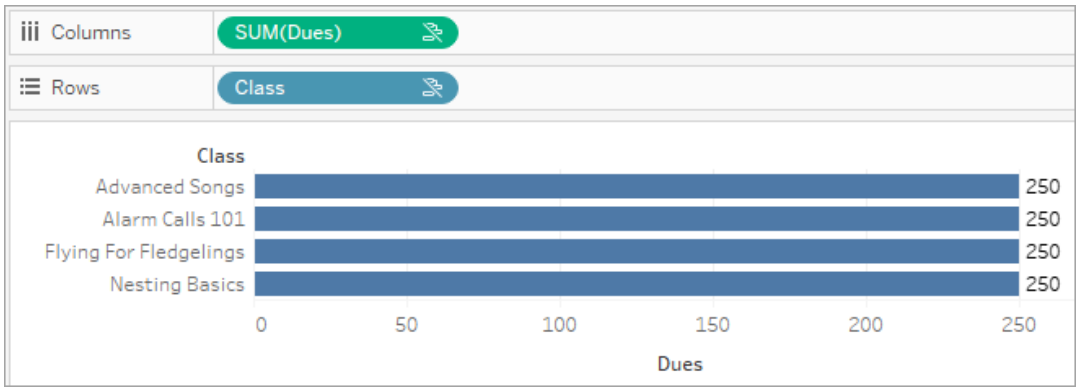
メジャーの値は、関連するディメンションによって決まります。関連するディメンションのないメジャーにはテーブル スコープが設定されます。関連するディメンションを持つメジャーは、関連するディメンション要素ごとに分類されます (つまり、メジャーの値はディメンション要素ごとに計算されます)。関

連のないディメンションが存在するために関連するディメンション要素が繰り返される場合、メジャーの値はディメンション要素に基づいて繰り返されます。

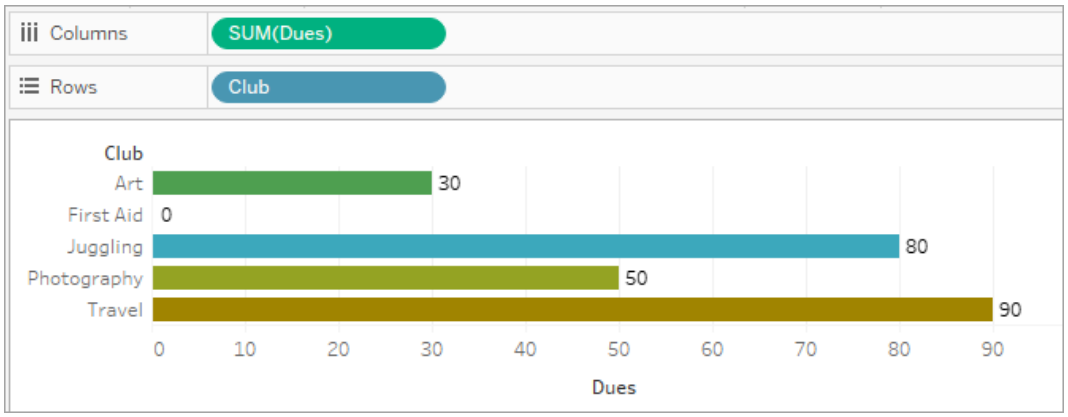
1. ディメンションのないメジャーは、その全体的な値にテーブル スコープが設定されます。



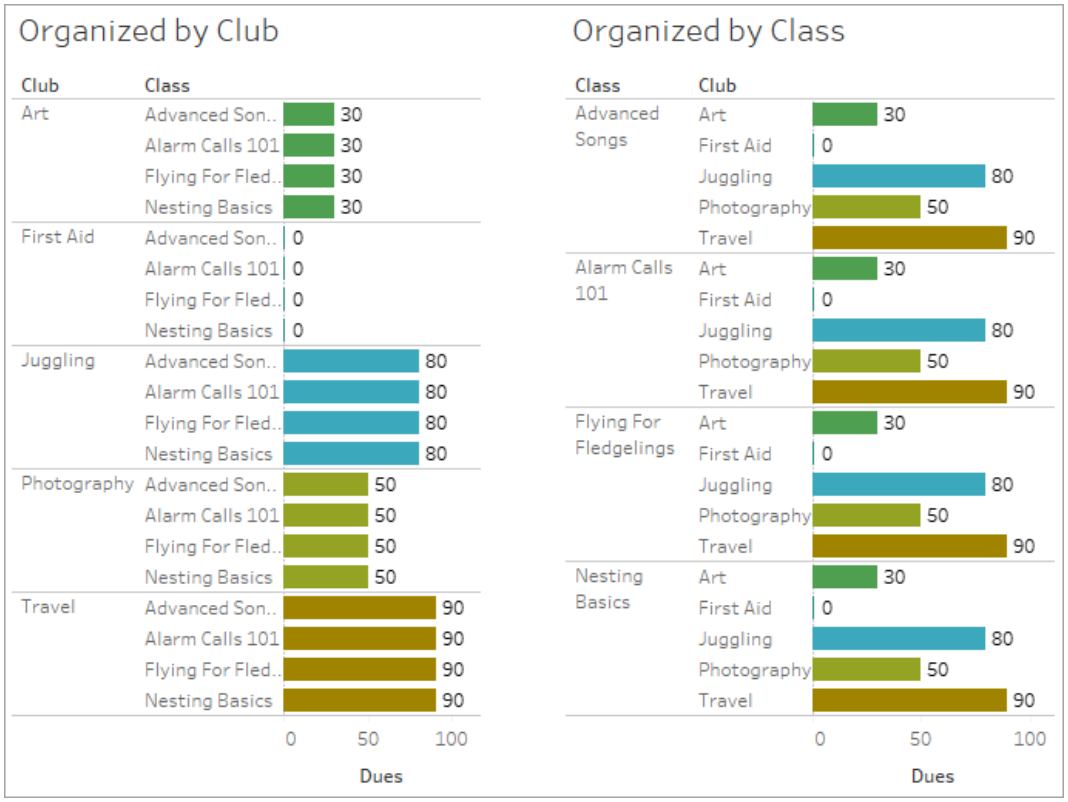
2. 関連のないディメンションが 1 つのみ存在する場合、メジャーにはテーブル スコープが設定され、その関連のないディメンション要素に対して繰り返されます。



3. 関連するディメンションが存在する場合、メジャーはより細かく分類され、その値は関連ディメンション要素ごとに計算されます。



4. 関連のないディメンションと関連するディメンションが存在する場合、メジャーは関連するディメンションによって分類されます。関連するディメンション要素が関連のないディメンションに対して繰り返される場合、メジャー バリューは常に関連するディメンション要素に追随します。



会費はクラブごとに発生するため、クラブが繰り返されるたびに各クラブの会費の金額も繰り返されます。

マルチファクト関係データモデルの構築

分析では、互いに直接的な関係はないものの、日付や場所などの同じ共通情報に関連しているデータのテーブルをまとめることがよくあります。このタイプの分析は、共有ディメンションを持つマルチファクト分析と呼ばれることもあります。

Tableau でこの種の分析を実行するには、共有テーブルによって接続されている複数の基底テーブルを使用するデータソースを作成する必要があります。

- 基底テーブルは、[データソース] タブのデータモデル内にある一番左のテーブルです。基底テーブルとして使用するテーブルを決定する方法については、「マルチファクト関係モデルを使用するとき」を参照してください。
- *共有テーブル*は、複数の入力関係を持つダウンストリーム テーブルです。これらのテーブルには、Viz での分析中に関連のないフィールドをつなぎ合わせるために使用できるフィールドが含まれています。日付と場所は、一般的に共有されるテーブルの例です。

モデルの構築

マルチファクト関係データモデルの構築は、関係を使用する他のデータソースの作成と基本的に同じですが、追加の基底テーブルと共有テーブルへの複数の入力関係という2 つの要素が追加されます。

1. データに接続します。テーブルがすべて同じデータベース内にない場合は、複数のデータ接続を使用できます。
2. 最初の基底テーブルを作成するには、テーブルをキャンバスにドラッグします。
3. 左側のペインから別のテーブルを **[New Base Table (新しい基底テーブル)]** ドロップ領域にドラッグします。

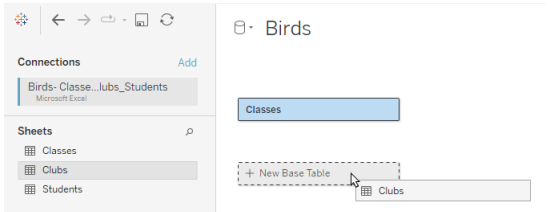
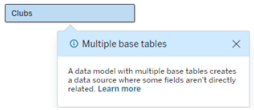
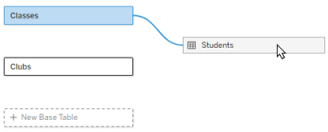


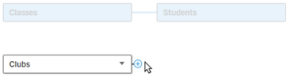
Tableau では、複数の基底テーブルを含むデータモデルを構築していることを示す警告が表示されます。データに必要な場合にのみ、複数の基底テーブル モデルを設定することをお勧めします。それ以外の場合は、複数の基底テーブル モデルに伴う複雑さを回避するために、単一の基底テーブル モデルを使用してください。



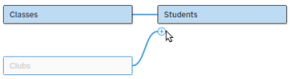
4. 別のフィールドをキャンバスにドラッグし、それを基底テーブルの 1 つに関連付けます。必要に応じて各関係を構成します。



5. まだ関連付けられていない基底テーブルにカーソルを合わせると、プラス記号 (愛称は「ミートボール」) が表示されます。



6. プラス記号アイコンを共有テーブルにドラッグして、新しい入力関係 (「ヌードル」とも呼ばれます) を作成します。



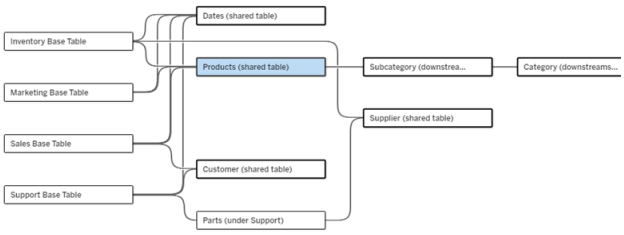
7. *オプション:* ダウンストリーム テーブルと基底テーブルの追加を続行します。必要に応じて、プラス記号アイコンをクリックしてドラッグし、共有テーブルを構築します。

注: 基底テーブルから始めるのではなく、最初に単一の基底テーブル モデルを構築し、後で基底テーブルを追加することもできます。

モデルの探索

複数の基底テーブルを持つデータモデルは関係を使用して構築されますが、すべてのテーブルが相互に同じ程度の関係を持つわけではありません。各基底テーブルはツリーを定義します。ツリーには、その基底テーブルに関連するすべてのテーブルが、直接関連しているか、関連テーブルのダウンストリームにあるかに関係なく含まれます。共有テーブルは複数のツリーに存在します。関連性の詳細については、「マルチファクト関係データモデルについて」を参照してください。

複数の基底テーブルを含むモデルを表示する場合、データモデルを探索および管理するためのさまざまなオプションがあります。これらのオプションは、データモデルが複雑な場合に特に役立ちます。



4 つの基底テーブル、複数の共有テーブル、および基底テーブル間で共有されている/共有されていないダウンストリーム テーブルの両方を含む複雑なデータモデルの例。

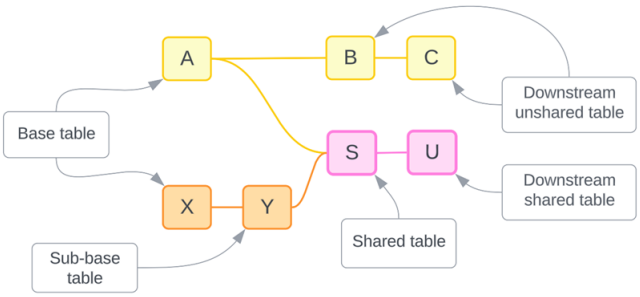
2024.2 では、データモデルの新しいレイアウトの詳細がいくつか導入されました。マルチファクト関係データモデルでは、関係がバンドルされるため、テーブルに含まれる入力関係の数を追跡しやすくなります。また、共有テーブル (およびダウンストリームの共有テーブル) のアウトラインは、共有されていないテーブルよりも太く表示されます。

用語

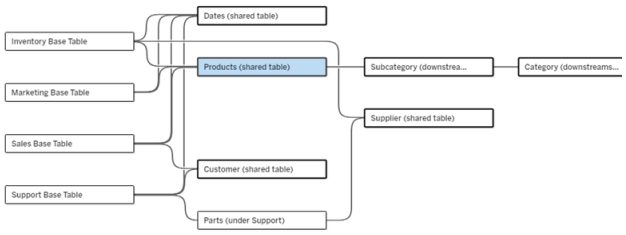
複数の基底テーブル データモデル内のテーブルには、特定の役割があります。マルチファクト関係データモデルに存在する必要があるテーブルは、基底テーブルと共有テーブルのみです。(複数の基底テーブルがない場合、それはマルチファクト関係データモデルではありません。基底テーブルを接続する共有テーブルがなければ、有効なデータモデルではありません。)

これらのモデルは複雑な性質を持つことが多いため、他のタイプのテーブルがデータモデルにどのように適合するかについて説明する際には、共通の用語を使用すると便利です。

- *基底テーブル* - 一番左にあり、入力関係を持ちません。
  - この例では、Inventory (在庫)、Marketing (マーケティング)、Sales (売上)、および Support (サポート) が基底テーブルです。
- *サブ基底テーブル* - 基本テーブルと共有テーブルの間にあります。
  - この例では、Parts (部品) がサブ基底テーブルです。
- *共有テーブル* - 複数の入力関係を持ちます。
  - この例では、Products (製品)、Dates (日付)、Customer (お客様)、Supplier (サプライヤー) が共有テーブルです。
- *ダウンストリームの共有テーブル* - 入力関係を 1 つのみ持ち、そのアップストリームのどこかに共有テーブルがあります。
  - この例では、Subcategory (サブカテゴリ) と Category (カテゴリ) がダウンストリームの共有テーブルです。
- *ダウンストリームの非共有テーブル* - 入力関係を 1 つのみ持ち、そのアップストリームには共有テーブルはありません。
  - この例では、ダウンストリームの非共有テーブルはありません。



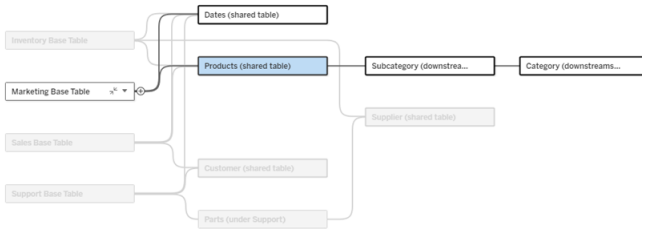
理解度をテストする: サンプル データ ソース内のテーブル タイプを識別する



- **基底テーブル**: Inventory (在庫)、Marketing (マーケティング)、Sales (売上)、Support (サポート)
- **サブ基底テーブル:** Parts (部品)
- **共有テーブル:** Products (製品)、Dates (日付)、Customer (お客様)、Supplier (サプライヤー)
- **ダウンストリームの共有テーブル:** Subcategory (サブカテゴリ)、Category (カテゴリ)
- **ダウンストリームの非共有テーブル:** なし

関係ツリーの特定

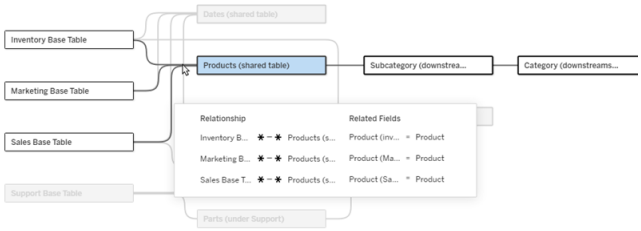
テーブルの上にカーソルを合わせると、関連するテーブルが強調表示されます。Tableau では、そのテーブルのツリーが強調され、関連のないテーブルは強調されません。



Marketing 基底テーブル ツリーは、2 つの共有テーブル (Dates (日付)、Products (製品)) と、製品のダウンストリームにある Subcategory (サブカテゴリ) テーブルと Category (カテゴリ) テーブルで構成されます。

関係の詳細を表示

関係または関係のバンドルの上にカーソルを合わせると、ツールヒントに詳細が表示されます。



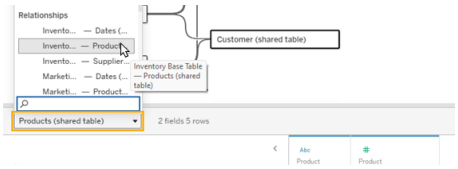
Product テーブルへの入力関係には、基底テーブル (Inventory、Marketing、Sales) への 3 つの個別の関係が含まれます。

関係の選択

関係を選択すると、データモデル キャンバスで青色に強調表示され、その詳細がデータモデル キャンバスの下の [テーブルの詳細] ペインに表示されます。[テーブルの詳細] ペインでは、関係句を検査または変更できます。

関係を選択するには、以下のようにいくつかの方法があります。

- キャンバス内の関係線 (ヌードル) をクリックします。すべての関係には、そのヌードルだけを選択するクリック可能な領域があります。
- キャンバス内のテーブルを右クリックするか、Control キーを押しながらクリックしてメニューを開きます。**[Selet Relationship (関係の選択)]** オプションを選択し、必要なテーブルの関係を選択します。
- キャンバス内の関係のバンドルをクリックすると、永続的なツールヒントが表示されます (バンドルの上にカーソルを合わせるとツールヒントが表示されます。永続化するにはクリックする必要があります)。次に、ツールヒントの詳細で行を選択して、モデル内のその関係を強調表示します。
- [テーブルの詳細] ペインのツールバーのメニューを開き、目的の関係を選択します。このメニューを使用して特定のテーブルを選択し、そのプレビューをペインに表示することもできます。



基底テーブルとの交換

基底テーブルと共有テーブル間の関係内にある中間テーブルでは、ダウンストリーム テーブルを基底テーブルと交換するオプションが提供されます。これは概念の理解しやすくするための単なる視覚的な変更であり、データモデルの構造を変更するものではありません。

ダウンストリーム テーブルを右クリックするか、Control キーを押しながらクリックし、**[Swap with base table (table name) (基底テーブル (テーブル名) と交換)]** を選択します。交換オプションは、ダウンストリーム テーブルにのみ存在し、基底テーブルには存在しません。

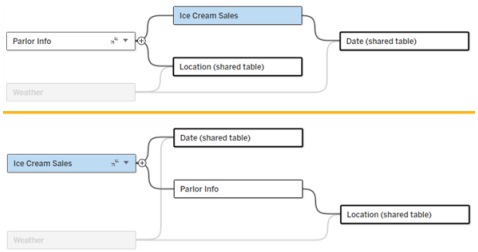
交換オプションは、共有テーブルやダウンストリームの共有テーブルなど、交換によってデータモデルが変更されるテーブルでは使用できません。基底テーブルと交換できるのは、ダウンストリームの非共有テーブルまたはサブ基底テーブルのみです。

例

この例では、データモデルの基本構造を変更せずに、**Parlor Info** テーブルと**Ice Cream Sales** テーブルを交換できます。他のテーブルは交換できません。

- Ice Cream Sales は、Parlor Info とDate (共有テーブル) の両方に関連しています。
- Parlor Info は、Ice Cream Sales とLocation (共有テーブル) の両方に関連しています。
- Weather は、Date とLocation の両方の共有テーブルに関連しています。

これらの 2 つのモデルは概念的には同等です。



基底テーブルの折りたたみ

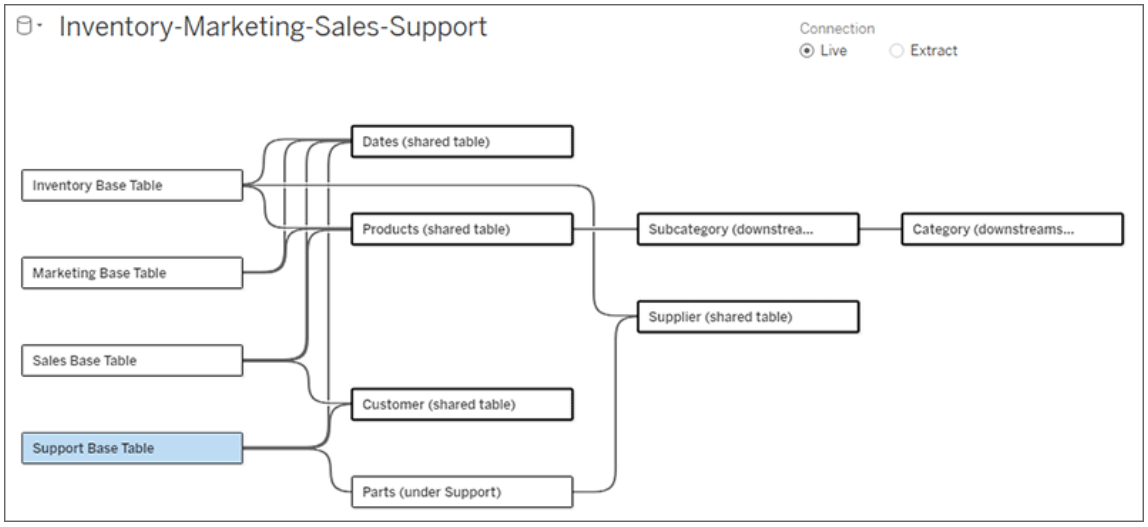
関係のパスまたはツリーをその基底テーブルにのみに折りたたんで、データモデルのビューを一時的に簡素化することもできます。

[Collapse (折りたたむ)] をクリックするか、基底テーブル上の [Expand (展開)] ボタンをクリックして、ツリー全体を折りたたむか、展開します。または、ダウンストリーム テーブルを右クリックまたは Ctrl キーを押しながらクリックし、**[Collapse this path (このパスを折りたたむ)]** または **[Collpase other paths (他のパスを折りたたむ)]** を選択します。このオプションは、共有テーブルまたは共有テーブルのダウンストリームのテーブルでは使用できません。

ツリーを基底テーブルに折りたたむことは、単なる視覚的な操作であるため、関連のないテーブルに関するアラートはトリガーされません。パスが折りたたまれると、積み重ねられたテーブルおよび展開ボタンを持つ基底テーブルが表示されます。折りたたみは、そのツリーにのみ関連するすべてのテー

ブルと関係に影響するものであるため、折りたたまれていないパスと共有されているテーブルは非表示になりません。

基底テーブルとツリー内のすべてをもう一度開くには、[展開]ボタンをクリックします。



トラブルシューティング

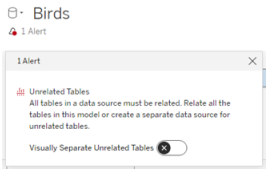
単一のデータソースの作成

すべてのテーブルは、何らかの形でデータソース全体に関連付けられている必要があります。全体的なデータモデルに関連していないテーブルがある場合は、アラートが表示されます。アラートは、データモデルの残りの部分から完全に分離されているテーブルまたはツリーがなくなるまで表示されます。アラートがアクティブな場合は、データソースをパブリッシュすることはできず、分析でデータソースを使用することもできません。

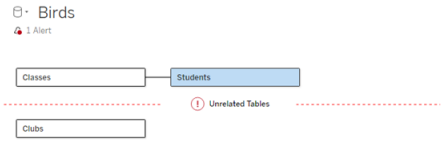
アラートを開いて、[Visually Separate Unrelated Tables (関連のないテーブルを視覚的に分離)]を設定すると、アラートをトリガーしたテーブルを識別できます。このオプションは、複雑なモデルを持っていて、どのテーブルがデータモデルの残りの部分にまだ関連付けられていないかを特定する必要がある場合に役立ちます。

例

「モデルの構築」の手順では、2 番目の基底テーブルを接続するための関係が追加される前に、ステップ 4 でアラートが表示されます。

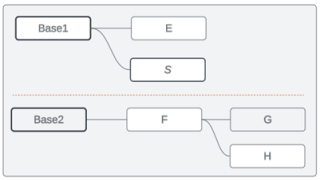


[Visually Separate Unrelated Tables (関連のないテーブルを視覚的に分離)] オンに設定すると、**クラブ** テーブルが **[Unrelated Tables (関連のないテーブル)]** 行の下に移動します。**Clubs** を **Students** に関連付けると、アラートが解決されます。

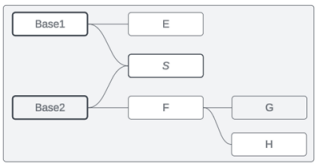


循環 の解決

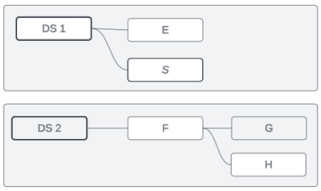
一部のテーブルが互いに直接関連していなくても、データモデル全体がまとまったものである必要があります。この例では、各基底テーブルはツリーを定義しますが、それら接続する共有テーブルがありません。これは有効な分析モデルではありません。



関連テーブルの 2 つのグループを共有テーブルを介して結合する必要があります...



または、データモデルを2つの別個のデータソースとして作成する必要があります。

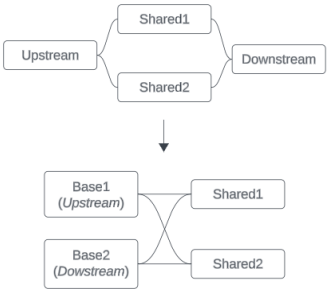


データモデルの制限事項

テーブル間の一部の関係パスは、複数の基底テーブル モデルでサポートされていません。テーブルに2番目の入力関係を作成しようとしたときにミートボールをドロップできない場合は、作成しようとしている構造がTableauでサポートされていることを確認してください。サポートされていないモデルの例は次のとおりです。

循環

循環 (アップストリーム テーブルから同じダウンストリーム テーブルへの関係パスが複数ある) はサポートされていません。このサポートされていない構造は、蝶ネクタイと呼ばれることもあります。Tableauでテーブル間のこのような関係をモデル化するには、ダウンストリーム テーブルを別のベース テーブルに変換して、蝶ネクタイではなく複数の基底テーブルを使用します。



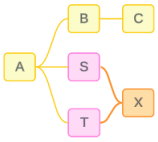
データモデルは、*有向非巡回グラフ*である必要があります。つまり、テーブルへのすべての入力関係は、アップストリームの別の基底テーブルまで追跡可能である必要があります。

サポートされていません。テーブル X には 2 つの入力関係があり、それらは共に基底

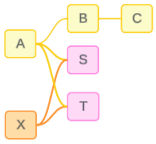
サポートされます。テーブル S と T には両方とも複数の入力関係があるが、それぞ

サポートされます。視覚的には循環があるように見えるが、テーブル S とテーブル T の両方

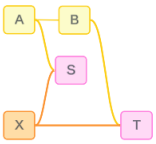
テーブル A のダウンストリームテーブルからのものである。



れが異なる基底テーブルからのものである。

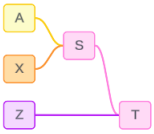


の入力関係は、異なる基底テーブルへと追跡することができる。



ネストされた共有テーブル

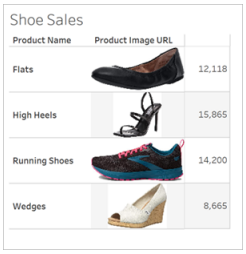
ネストされた共有テーブルはサポートされていません。共有テーブル (複数の入力関係を持つテーブル) のダウンストリーム テーブルが持つことがきる入力関係は、1 つだけです。



サポートされていません。テーブル T は共有テーブルのダウンストリームであり、追加の入力関係を受け取ることができません。

Web 画像をワークシートに動的に追加する

Tableau では、画像はデータを分析する際の強力なツールです。靴の月別売上高の Viz を想像してみてください。データは、ウェッジ ヒールよりもハイ ヒールのほうが売れていることを示していますが、この 2 つのタイプの靴の違いを想像することはできません。そこで、イメージの役割が登場します。Web 画像をワークシートに動的に追加し、それらをヘッダーで使用して視覚的な詳細を追加することができます。



データソースを準備する

イメージの役割は、Web 画像を指す URL を含む個別のディメンション フィールドに割り当てることができます。データを準備するには、画像フィールドが、イメージの役割を割り当てるための Tableau によって設定されている要件を満たしていることを確認します。

- URL で移動する先が、.png、.jpeg、.jpg、.svg、.webp、.jfif、.ico、.gif のいずれかの画像ファイルであることを確認します。
- 各 URL が http または https で始まっていることを確認します。トランスポートプロトコルが含まれていない場合、Tableau は https を想定します。
- データセットで使用される画像の数を最適化します。通常は、フィールドごとに最大 500 個の画像を読み込むことができます。
- 各画像ファイルが 200 KB 未満であることを確認します。

注: Tableau 23.1 以前を使用している場合、URL は、ファイル拡張子が .jpg、.jpeg、または .png の画像ファイルに移動する必要があります。

Tableau 23.2 以降では、.gif ファイルがサポートされていますが、.gif *アニメーション*は、クライアント側レンダリングが有効になっている Tableau Cloud および Tableau Server でのみ表示されます。サーバー側レンダリングが有効になっている Tableau Desktop および Tableau Server では、.gif ファイルが静的画像として表示されます。

Viz の複雑さによっては、Tableau がデフォルトでサーバー側のレンダリングを使用する場合があります。その場合、フィールドあたりの画像数は 100 に制限されます。複雑性の設定とサーバー側のレンダリングの詳細については、「クライアント側レンダリングの構成」を参照してください。

データセットの例:

製品名	製品画像 URL	製品売上高
フラット	https://img.example.com/flats.png	12,118
ハイヒール	https://img.example.com/highheels.png	15,865

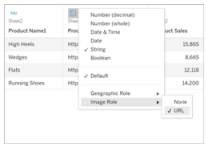
ランニング シューズ	https://img.example.com/runningshoes.png	14,200
ウェッジ	https://img.example.com/wedges.png	8,665

イメージの役割を URL に割り当てる

データ ソースに接続したら、[データ ソース] ページまたはワークシートの [データ] ペインからイメージの役割を URL に割り当てることができます。

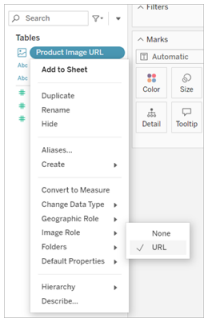
[データ ソース] ページから割り当てる:

1. 画像の URL がある列を見つけます。
2. 列の左上にあるアイコンを右クリック (Mac では Control を押しながらクリック) し、**[イメージの役割] > [URL]** を選択します。
   アイコンが画像アイコンに変わり、画像を使用できるようになります。



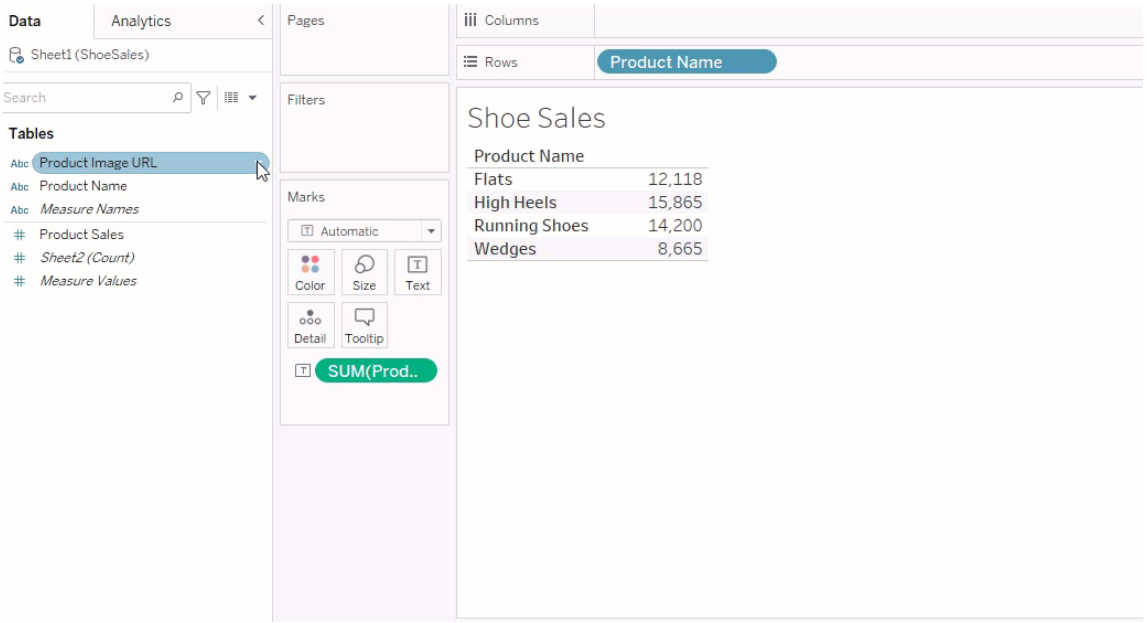
ワークシートから割り当てる:

1. 新しいワークシートを開きます。
2. 画像の URL がある個別のディメンション フィールドを見つけます。
3. ディメンション フィールドを右クリック (Mac では Control を押しながらクリック) し、**[イメージの役割] > [URL]** を選択します。
   アイコンが画像アイコンに変わり、画像を使用できるようになります。



ビジュアライゼーションに画像を追加する

ワークシートから、[イメージの役割] フィールドを [行] または [列] シェルフにドラッグします。これで Viz 上で画像と関連データを表示できるようになりました。



ビジュアライゼーションを共有する

画像を含むワークブックはエクスポートして共有することができます。Viz で使用する画像は、Viz を共有するすべてのユーザーが閲覧できるようにしてください。たとえば、ファイアウォールの内部にある内部サーバーでホストされている画像を使用している場合は、Viz を共有するユーザー全員が画像にアクセスするための十分な権限を持っていることを確認します。この例では、同じサーバーに接続して Viz を表示しているユーザーは、問題な く画像を表示しています。ただし、Viz を Tableau Cloud サーバー上の PDF にエクスポートしているユーザーは、画像を表示できない場合があります。

エクスポートされたワークブックを Viewer が Tableau 2022.3 以前で開こうとすると、画像を表示することができません。

注: ベストプラクティスとして、スクリーン リーダーやその他のアクセシビリティ ソフトウェアがコンテンツにアクセスできるように、イメージの役割フィールドは常にテキストの説明と一緒に配置します。

画像の接続に関するトラブルシューティング

フィールドごとに許可されている画像の数を超えてる場合や、複雑な Viz に大きな画像ファイルがある場合、またはモバイル デバイスで Viz を表示している場合に、画像が表示されないことがあります。このセクションは、このようなエラーのトラブルシューティングに役立ちます。

Viz に画像が表示されない

Viz 内の画像が多すぎる

Viz の複雑さにもよりますが、通常、フィールドごとに 500 個の画像を読み込むことができます。複雑な Viz の場合は、既定でサーバー側レンダリングになることがあります。サーバー側レンダリングでは、フィールドごとに 100 個の画像を読み込むことができます。

Viz 内の画像が多すぎるというエラー メッセージが表示された場合は、画像をフィルターで除外してからやり直してください。

Tableau が画像にアクセスできない

画像の代わりに破損した画像アイコンが表示される場合は、まず画像を表示するための十分な権限があることを確認してください (または、画像がファイアウォールの内側にある場合など)。Tableau は画像にアクセスできなければならず、画像を表示するために別の認証を要求することはできません。

画像を表示する権限があり、サイズ要件を満たしている場合は、設定ページで Web 画像が有効になっていることを確認してください。

Tableau Desktop の場合:

1. ワークブックから、ツールバーの **[ヘルプ]** をクリックします。
2. **[設定とパフォーマンス] > [ダッシュボードWeb 表示セキュリティ]** の順に選択します。
3. **[Web ページ オブジェクトとWeb 画像を有効にする]** がオンになっていることを確認します。

Tableau Cloud の場合:

1. ナビゲーション パネルで、**[設定]** をクリックします。
2. [全般] で、下にスクロールして **[Web ページ オブジェクト] と[Web 画像]** を見つけます。
3. **[Web ページ オブジェクトとWeb 画像を有効にする]** がオンになっていることを確認します。

一部の画像が Viz に表示されない

画像ファイルが大きすぎる

レンダリングするには、各画像ファイルを200 KB 未満にする必要があります。画像ファイルのサイズを確認して、もう一度やり直してください。

以前のバージョンの Tableau を使用している

Tableau 23.1 以前を使用している場合、.png、.jpeg、および .jpg 画像ファイルのみがサポートされます。Tableau のバージョンをアップグレードするか、使用している Tableau のバージョンでサポートされている種類のファイルを使用してください。

画像の URL は http または https で始まる必要があります

各画像の URL は http または https で始まる必要があります。Tableau は現在、FTP/SMTP 呼び出しをサポートしていません。URL の形式を確認して、もう一度やり直してください。

画像ファイルは URL キーである必要があります。

イメージの役割は、.png、.jpeg、または .jpg、.svg、.webp、.jfif、.ico、.bmp、または .gif の画像ファイルに移動する URL にのみ割り当てることができます。URL の形式を確認して、もう一度やり

直してください。

画像ファイルの種類がサポートされていない

イメージの役割は、.png、.jpeg、または .jpg、.svg、.webp、.jfif、.ico、.bmp、または .gif の画像ファイルに移動する URL にのみ割り当てることができます。URL の形式を確認して、もう一度やり直してください。

Tableau 23.1 以前を使用している場合、.png、.jpeg、および .jpg 画像ファイルのみがサポートされます。Tableau のバージョンをアップグレードするか、使用している Tableau のバージョンでサポートされている種類のファイルを使用してください。
Tableau 23.2 以降では、.gif ファイルがサポートされていますが、.gif *アニメーション*は、クライアント側レンダリングが有効になっている Tableau Cloud および Tableau Server でのみ表示されます。サーバー側レンダリングが有効になっている Tableau Desktop および Tableau Server では、.gif ファイルが静的画像として表示されます。

画像ファイルに不適切な文字が含まれている

次の文字が含まれる URL にはイメージの役割を割り当てることはできません。

```
< > & \ ^ '
```

または以下の文字シーケンス:

```
..\\.\r \n \t
```

URL にこれらの文字または文字列が含まれていないことを確認してから、もう一度やり直してください。

ワークシートの外に画像が表示されない

ツールヒントViz に画像が表示されない

ツールヒントの Viz はサーバー側レンダリングで処理されるため、フィールドごとに最大 100 個の画像を読み込むことができます。フィールドごとの画像が 100 個未満であることを確認してから、もう一度やり直してください。

サーバー側とクライアント側のレンダリングの詳細については、「クライアント側レンダリングの構成」を参照してください。

ツールヒントViz の詳細については、「ツールヒント内でビューを作成する(ツールヒントViz)」を参照してください。

[サムネイルで表示] に画像が表示されない

[サムネイルで表示] はサーバー側レンダリングで処理されるため、フィールドごとに最大 100 個の画像を読み込むことができます。フィールドごとの画像が 100 個未満であることを確認してから、もう一度やり直してください。

サーバー側とクライアント側のレンダリングの詳細については、「クライアント側レンダリングの構成」を参照してください。

ワークブックをエクスポートすると画像が表示されない

すべてのエクスポートおよびエクスポート関連の機能は、サーバー側レンダリングで処理されます。これにより、フィールドごとに最大 100 個の画像を読み込むことができます。フィールドごとの画像が 100 個未満であることを確認してから、もう一度やり直してください。

モバイルデバイスで画像が表示されない

モバイル デバイスはコンピューターに比べて複雑性のしきい値が低いため、サーバー側レンダリングで処理を完了します。そのため、フィールドごとに最大 100 個の画像を読み込むことができますフィールドごとの画像が 100 個未満であることを確認してから、もう一度やり直してください。

モバイル デバイスで複雑性の設定を変更できます。複雑性のしきい値の詳細については、「コンピューターとモバイル デバイスの複雑性のしきい値を構成する」を参照してください。

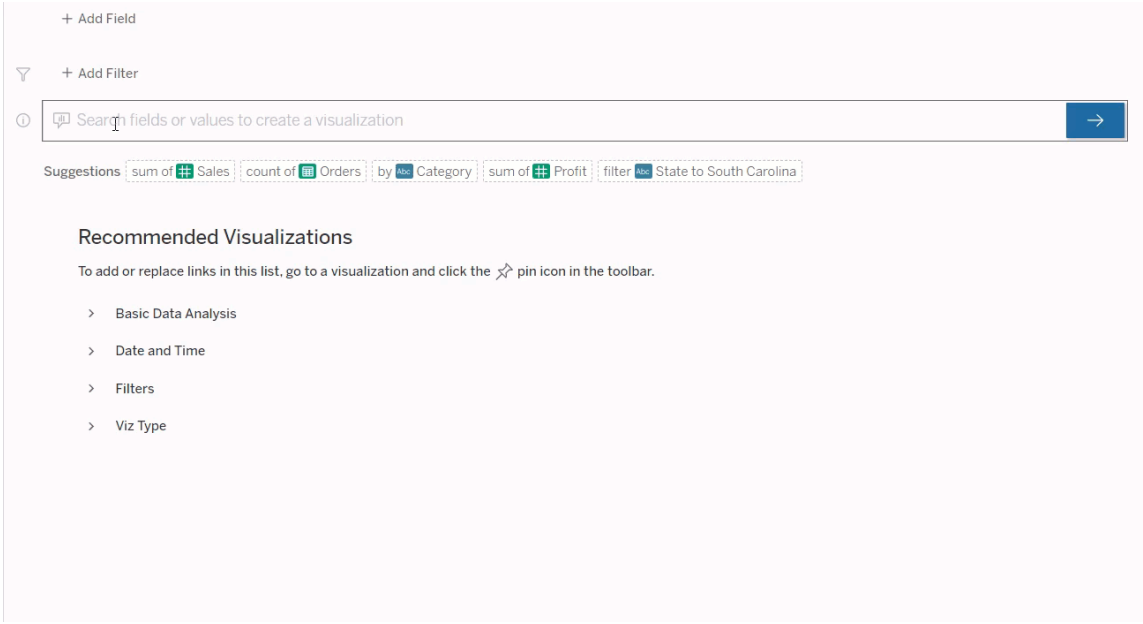
「データに聞く」(Ask Data) 機能を使用した自動的なビューの構築

「データに聞く」とメトリクスへの重要な変更点

Tableau の「データに聞く」機能とメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「*Tableau AI とTableau Pulse によるデータエクスペリエンスの再構築*」を参照してください。

「データに聞く」を使用すると、一般的な言葉で質問を入力し、Tableau で即座に回答を得ることができます。回答は自動的なデータビジュアライゼーションの形で得られ、フィールドを手動でドラッグ アンド ドロップしたり、データ構造のニュアンスを理解したりする必要はありません。

「データに聞く」を使用すると、時系列分析や空間分析のような主要な分析概念をサポートし、「去年」や「一番人気の」といった会話調のフレーズを理解することで、高度な質問を自然に行うことができます。

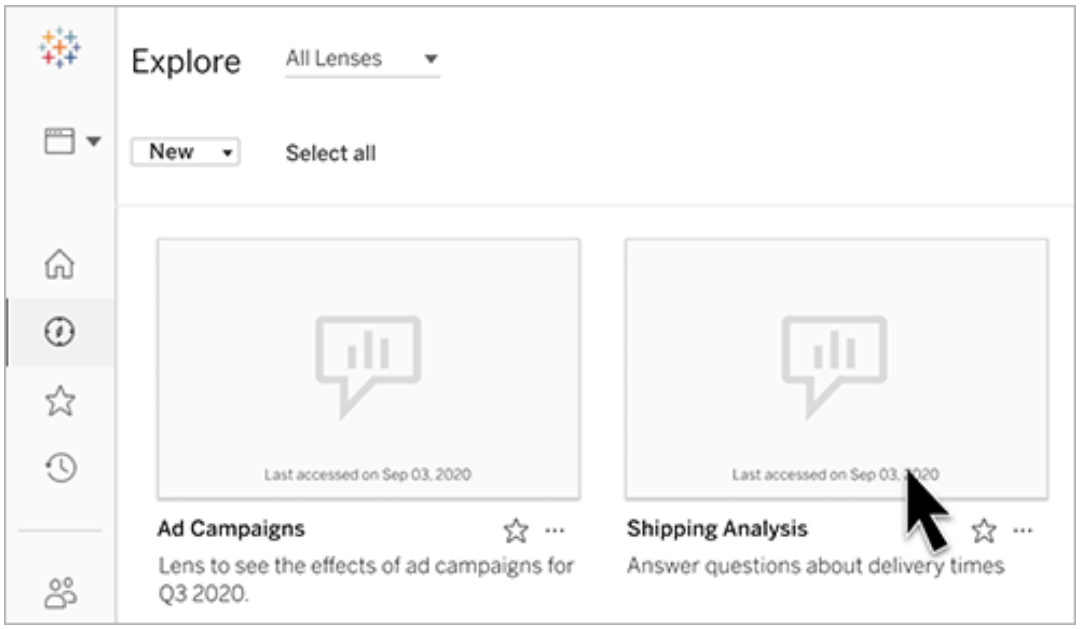


「データに聞く」レンズに移動する

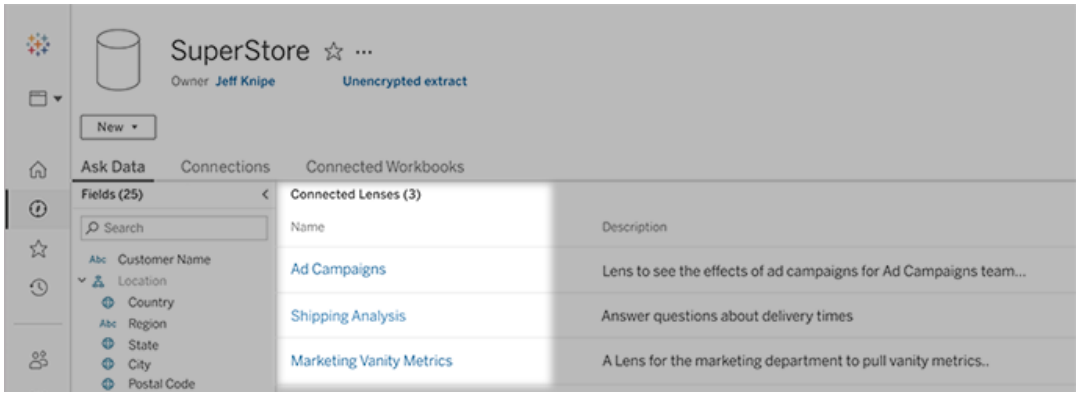
「データに聞く」(Ask Data) を使用してデータソースのクエリを実行する前に、Tableau 作成者は、まずレンズを作成して、レンズで使用するデータフィールドのサブセットを指定する必要があります。

Tableau で「データに聞く」レンズにアクセスできる場所は次のとおりです。

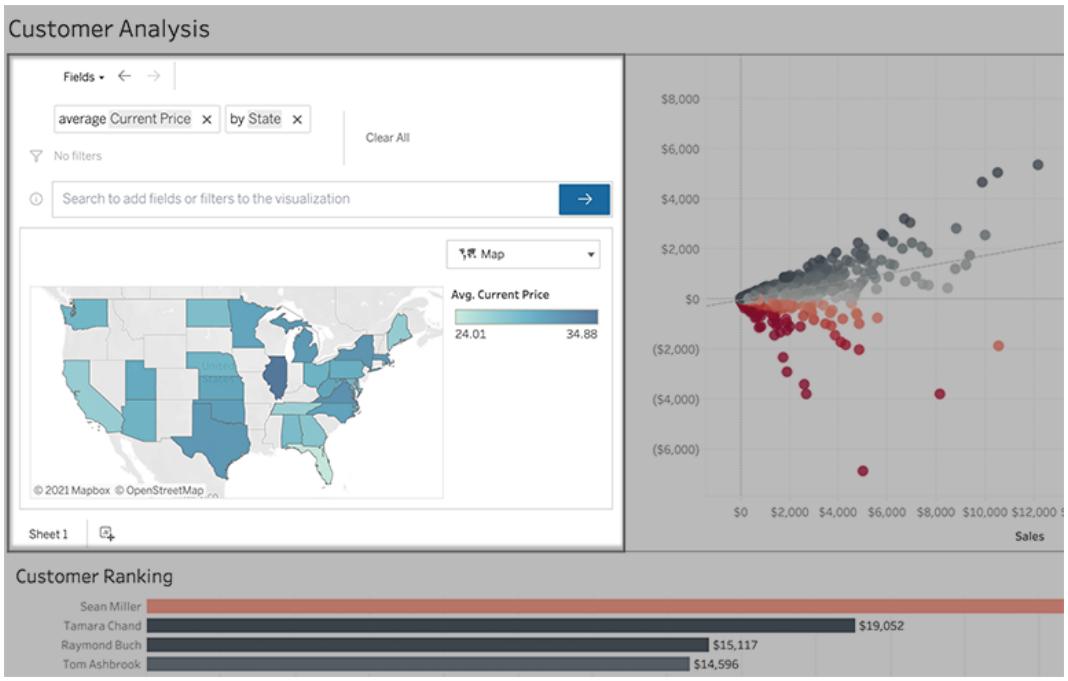
- Tableau Cloud または Tableau Server サイトのトップレベルにある [すべてのレンズ] ページ。



- レンズを作成したデータソースの「データに聞く」タブ。



- ダッシュボードの「データに聞く」オブジェクト。



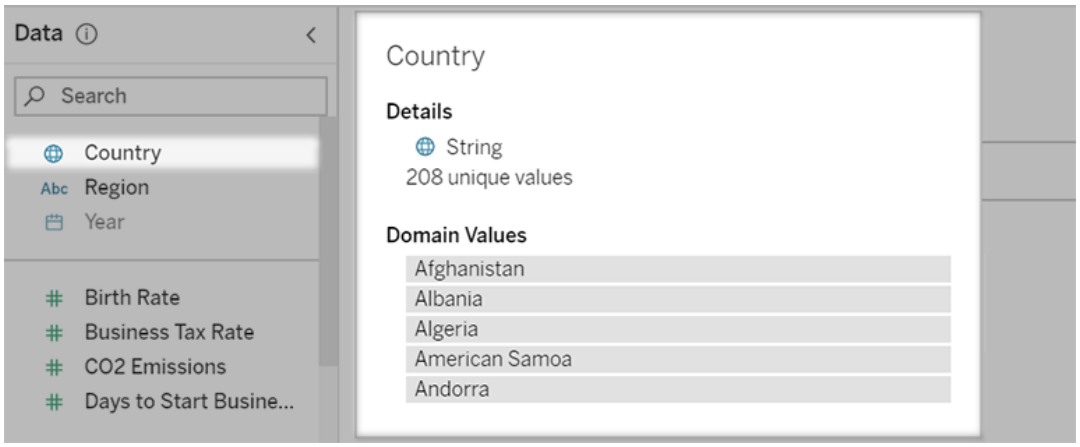
レンズ ページまたはダッシュボード オブジェクトから「データに聞く」

レンズに移動して、データをもっと詳しく知る

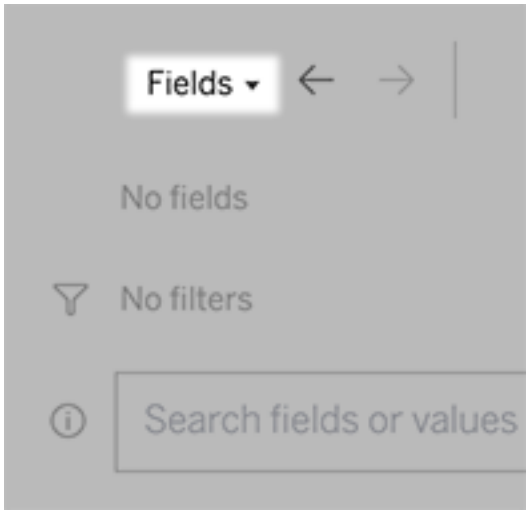
1. レンズに移動するには、Tableau サイトの [すべてのレンズ] ページ、データソースの [データに聞く] タブ、ダッシュボードの [データに聞く] オブジェクトなどを選択します。
2. (オプション) **[Recommended Visualizations (ビジュアライゼーションの候補)]** で、エントリをクリックすると、レンズの作成者が組織用に作成した視覚化がすばやく表示されます。

   推奨事項が現在のデータ分析のニーズに対応していない場合は、クエリを作成して独自の質問を作成してください。

3. 左側の [データ] ペインで、各フィールドにカーソルを合わせると、そこに含まれるデータの詳細が表示されます。

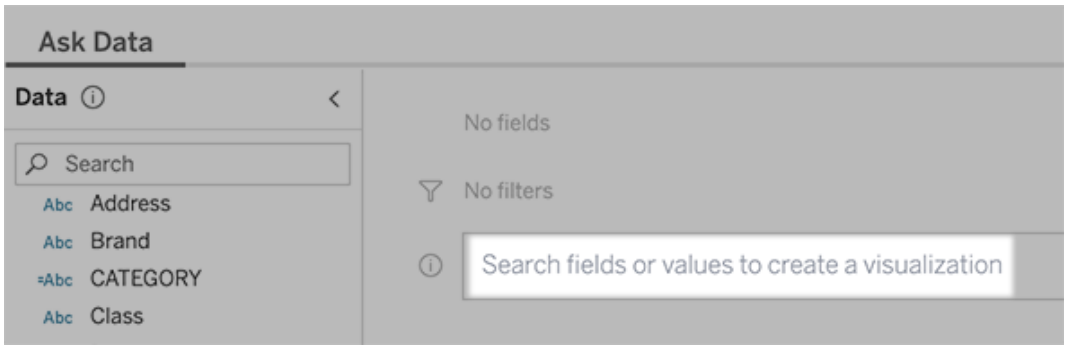


幅が狭いダッシュボードオブジェクトでは、[データ] ペインが非表示になっている場合がありますが、**[フィールド]** ドロップダウン メニューをクリックすると同様の情報を表示できます。

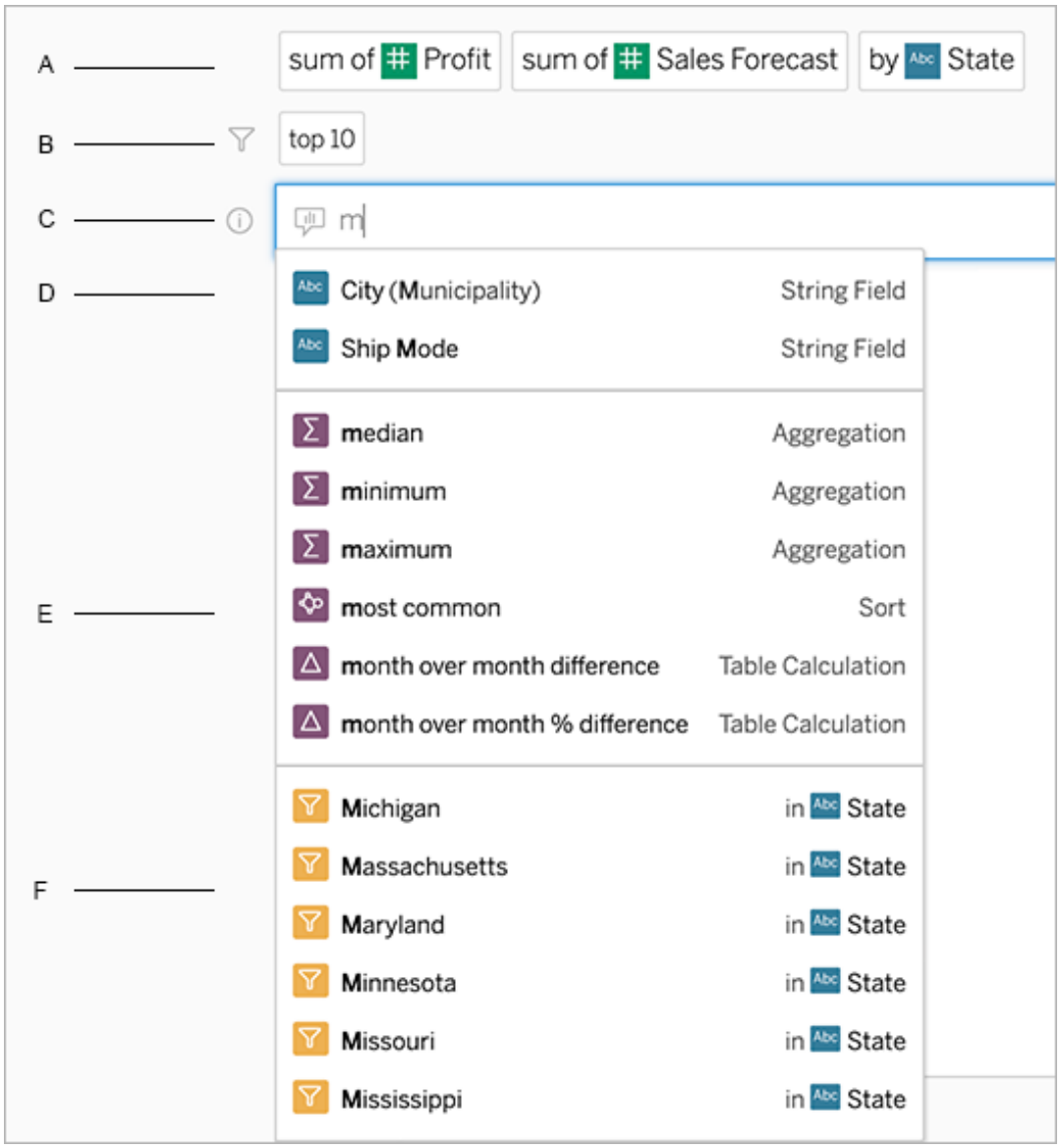


テキストを入力してクエリを作成する

1. **[フィールドまたは値を検索してビジュアライゼーションを作成する]** というボックスに入力します。



2. 入力すると、「データに聞く」(Ask Data) はデータフィールド、関数、および文字列値を検索し、結果をドロップダウン リストに表示します。リスト内のアイテムをクリックして、検索ボックスの上に表示されている現在のエントリに追加します。現在のエントリを使用して Viz を自動的に作成するには、任意の時点で **Enter キー**を押します。



フィールドと分析関数を検索してクエリを変更する

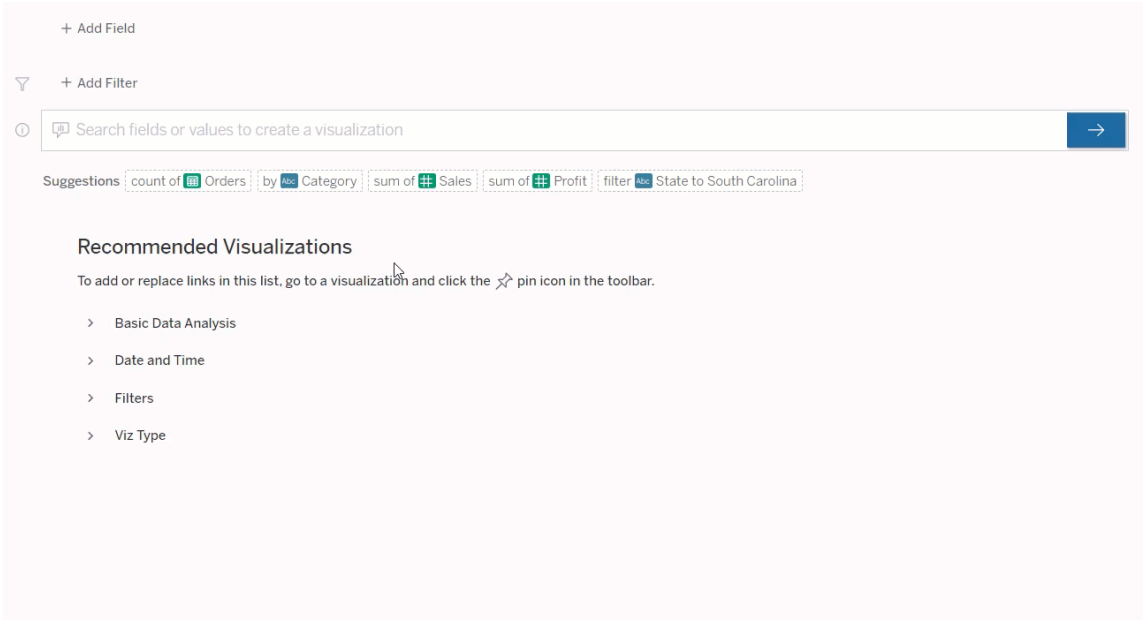
A 現在のエントリB.現在のフィルターC.検索ボックスD.返されたフィールドE.返された分析関数 F.返されたフィールド値

提案されたフレーズを追加してクエリを作成する

Tableau は、あなたのレンズや組織内の他のユーザーから尋ねられた、よくある問い合わせに基づいてフレーズを提案します。レンズを開くと、よくある質問にすばやく答えられるような提案が表示さ

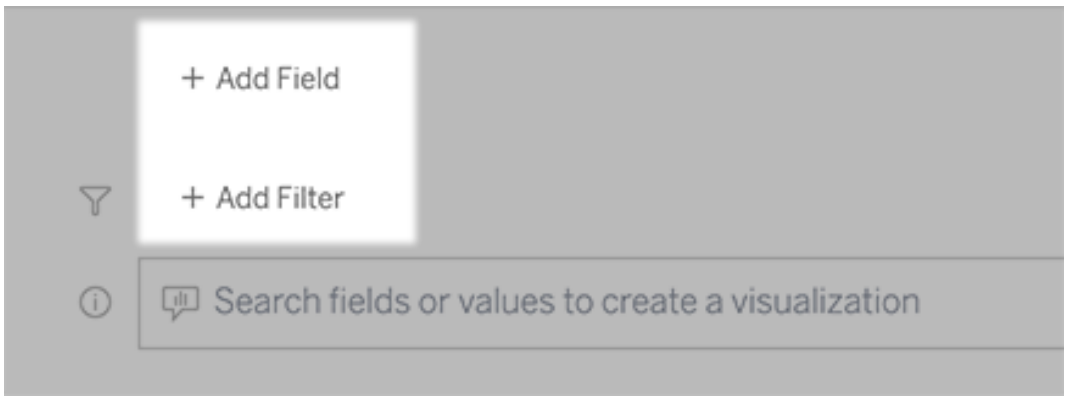
れます。

これらの提案をクリックすると、クエリに追加できます。クエリにフレーズを追加すると、より関連性の高いフレーズで提案が動的に更新されます。選択するたびに、自動的にビューが構築されます。

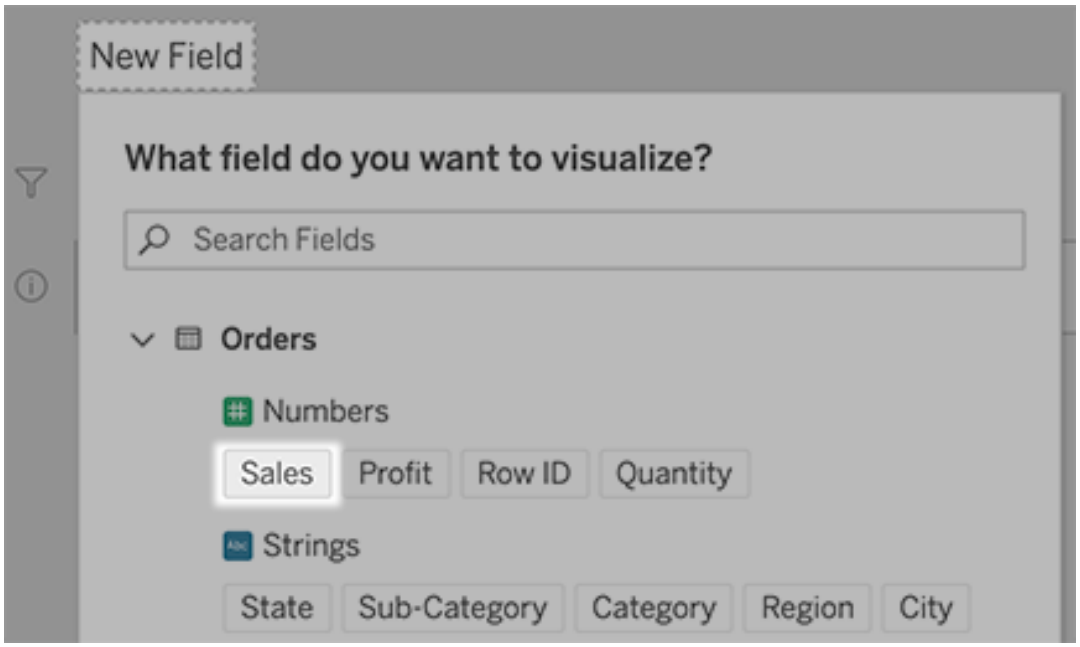


フィールドとフィルターを追加してクエリを作成する

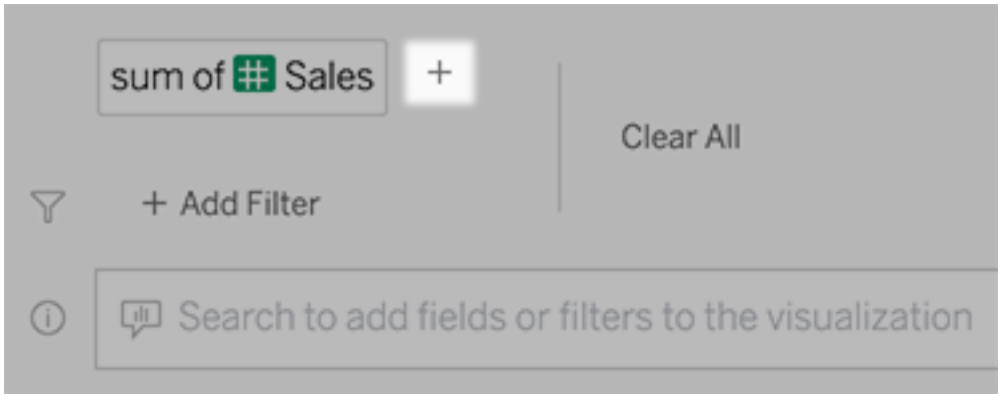
1. **[フィールドの追加]** または **[フィルターの追加]** をクリックします。



2. 目的のフィールドをクリックします。(長いリストを絞り込むには、最初に **[検索フィールド]** ボックスに入力します。)

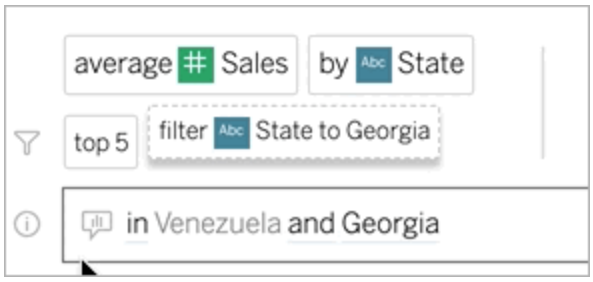


3. サブオプション (数値フィールドの集計タイプ、文字列フィールドと日付フィールドのグループ化など) を設定します。
4. フィールドまたはフィルターをさらに追加するには、プラス記号をクリックします。



クエリの要素がどのように適用されているかを確認する

クエリの要素がどのように適用されているかを確認するには、テキストボックスまたはその上の解釈にカーソルを合わせます。使用されていない単語はグレー表示されているため、「データに聞く(Ask Data)」に伝わりやすくなるようにクエリのフレーズを変更できます。



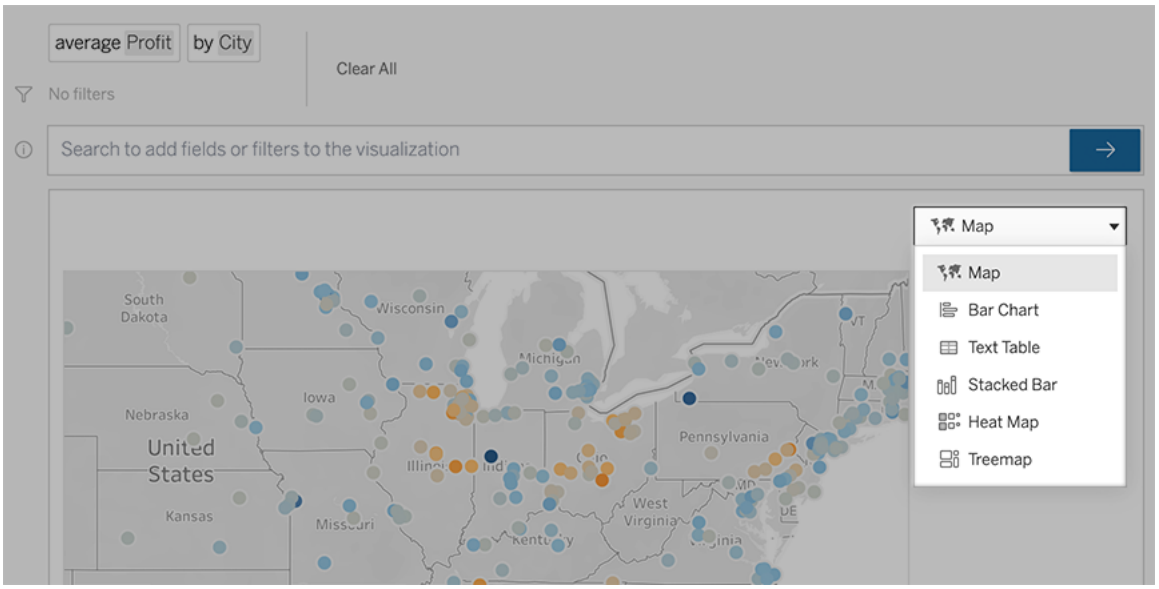
質問のフレーズの変更

ユーザー インターフェイスでオプションやデータ フィールド、フィルターなどをクリックすることによって、質問のフレーズを変更することもできます。

Viz タイプの変更

既定の Viz からデータの十分な情報が取得できない場合は、右上隅にあるメニューをクリックし、以下のサポートされている Viz タイプから選択します。

- 棒グラフ
- ガント棒
- ヒートマップ
- ヒストグラム
- 折れ線グラフ
- マップ
- 円グラフ
- 散布図
- 積み上げ棒グラフ
- テキストテーブル
- ツリーマップ

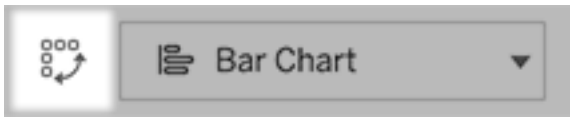


注: 特定の Viz タイプを自動的に作成するため、「データに聞く」がエントリに [レコード数] などのフィールドを追加する場合があります。

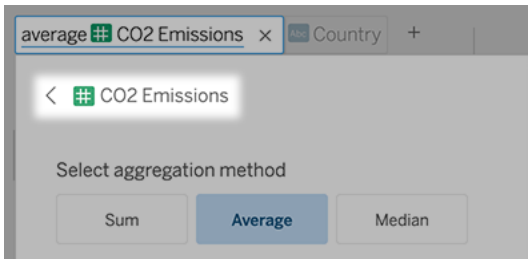
フィールド、フィルター、表示データを変更する

「データに聞く」(Ask Data) により、フィールド値の表示方法をいくつかの方法で微調整できます。

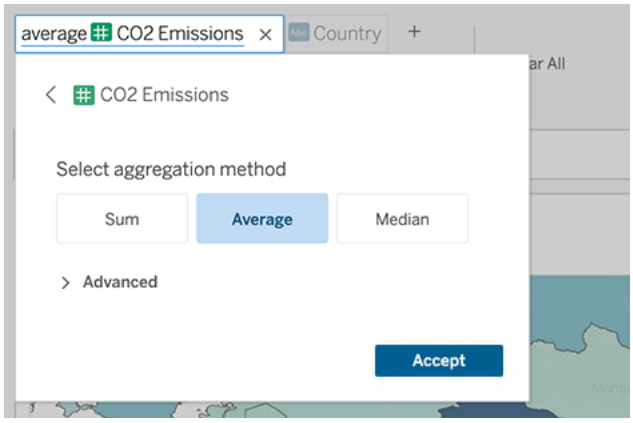
- 縦軸と横軸で使用するフィールドを切り替えるには、Viz 選択メニューの左にある [軸の交換] ボタン をクリックします。

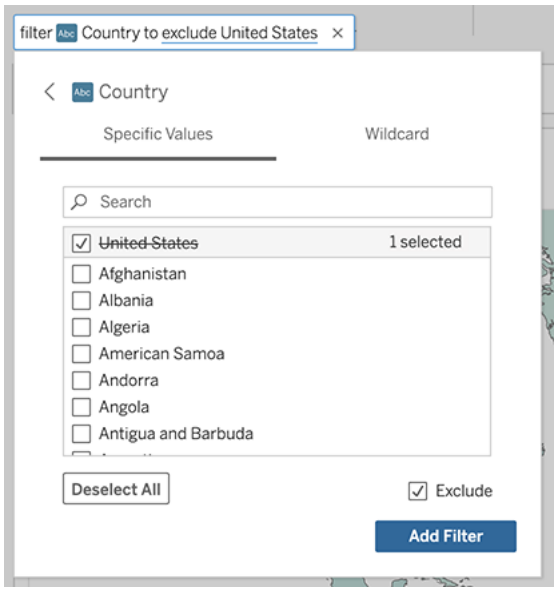
- フィールドを変更するには、最初にクエリエントリでフィールドをクリックしてから、下のフィールド名をクリックします。(差の計算に使用されるフィールドを変更するには、「時間の経過による差異の比較」を参照してください。)



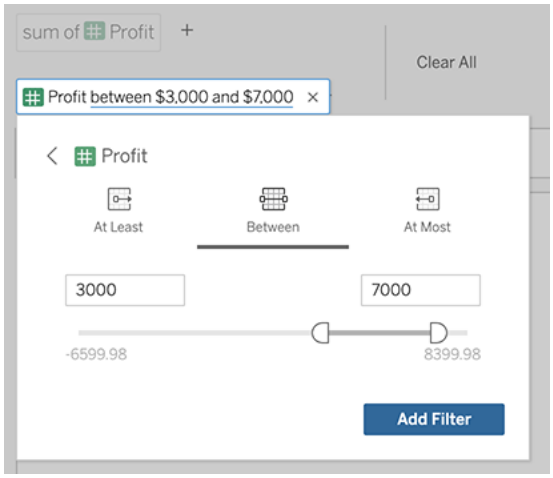
- フィールドの集計タイプまたはグループ化タイプを変更 (平均から合計に変更するなど) するには、テキスト ボックスのフィールド名をクリックしてから、別の集計タイプまたはグループ化タイプを選択します。



- カテゴリフィルターの場合は、以下の例の「米国を除外」のように、値をクリックして特定の値を変更するか、ワイルドカード パラメーターを入力します。



- 数値範囲を調整するには、「高い」、「中間」、「安い」などの単語をクリックします。



- フィールドやフィルターを削除するには、カーソルを合わせて「**X**」をクリックします。

日付フィルターの調整

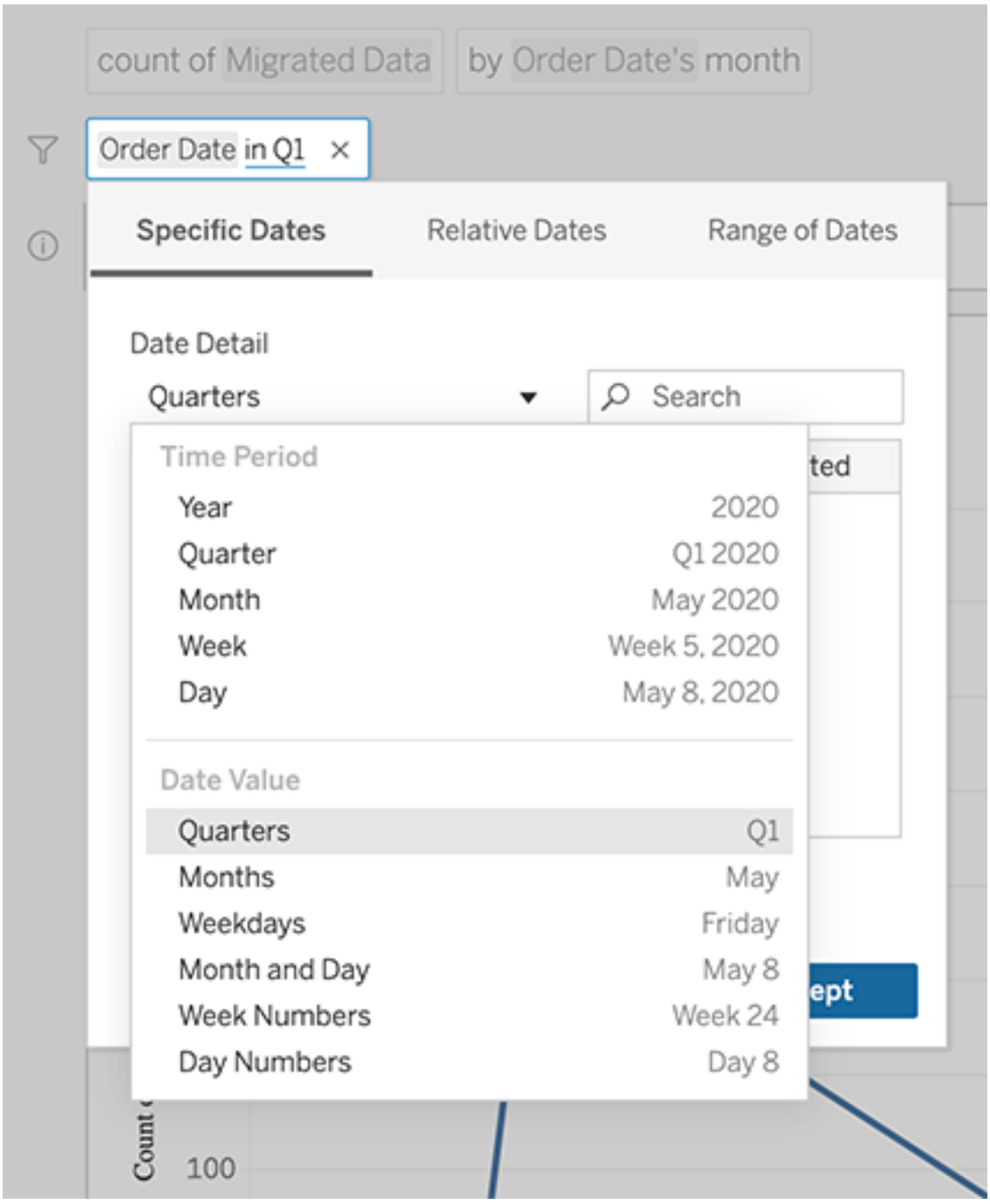
日付フィルターを調整するには、「最後の」、「前の」などの単語をクリックします。その後、次のいずれかをクリックします。

- **[特定の日付]** をクリックして、特定の期間または日付の値を入力します。
- **[相対日付]** をクリックして、現在の日付を基準にした日付範囲を表示します。

- **[日付範囲]** をクリックして、特定の開始点および終了点を入力します。

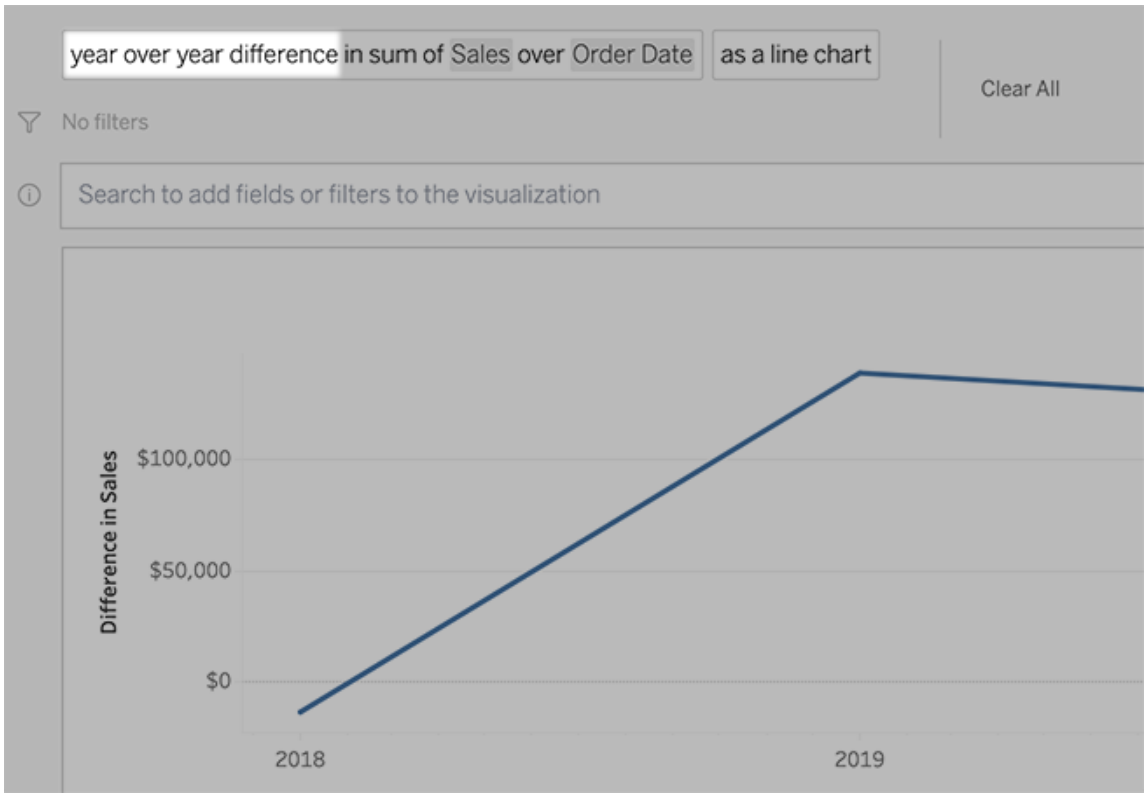
[特定の日付] では、**[日付の詳細]** メニューで次のような固有のオプションが提供されます。

- **[期間]** オプションでは、連続した日付の範囲が 1 つ表示されます。
- **[日付の値]** オプションでは、複数の期間で繰り返すことができる複数の範囲が表示されます。たとえば、複数年にわたる第 1 四半期の売上実績を組み合わせる場合は、[日付の値] で [四半期] を選択します。

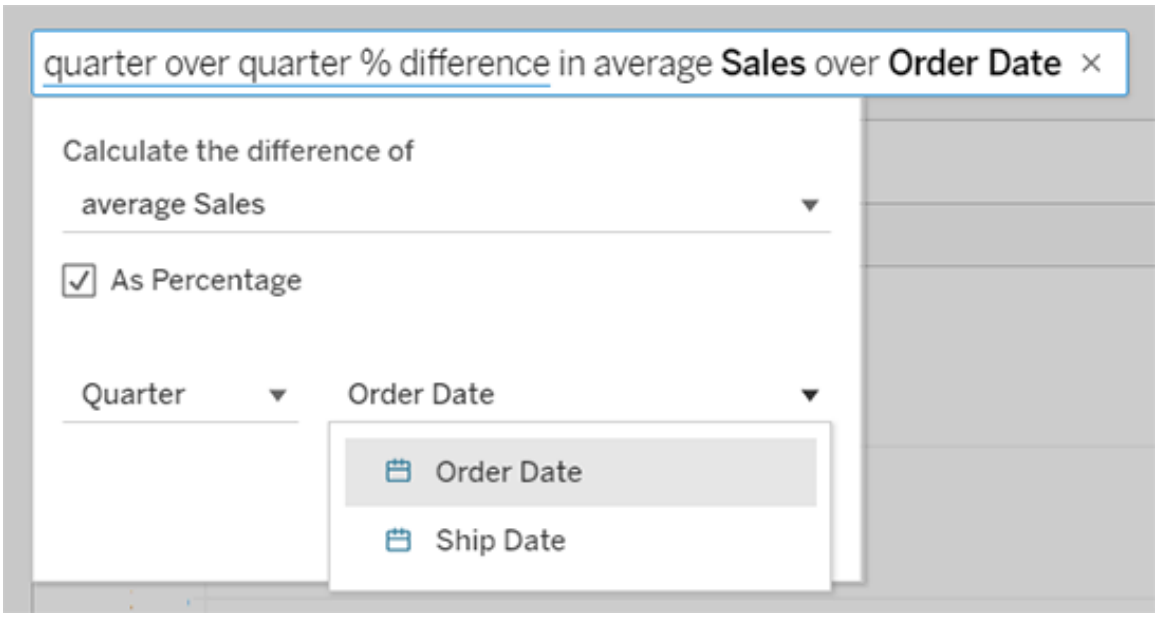

count of Migrated Data
by Order Date's month
Order Date in Q1
Specific Dates
Relative Dates
Range of Dates
Date Detail
Quarters
Search
Time Period
Year 2020
Quarter Q1 2020
Month May 2020
Week Week 5, 2020
Day May 8, 2020
Date Value
Quarters Q1
Months May
Weekdays Friday
Month and Day May 8
Week Numbers Week 24
Day Numbers Day 8
100

時間の経過による差異の比較

「データに聞く」を使用すると、「前年比」や「四半期比」などの語句を使用して期間を比較できます。結果は、「データに聞く」から保存したワークブックの差分または差の割合表計算として表示されます。



テキストボックスで差分計算をクリックして、他のフィールド、集計方法、および期間を選択します。



単純計算を適用する

「データに聞く」(Ask Data) では 2 つのメジャー間の単純計算がサポートされており、以下の記号を使用して適用できます。

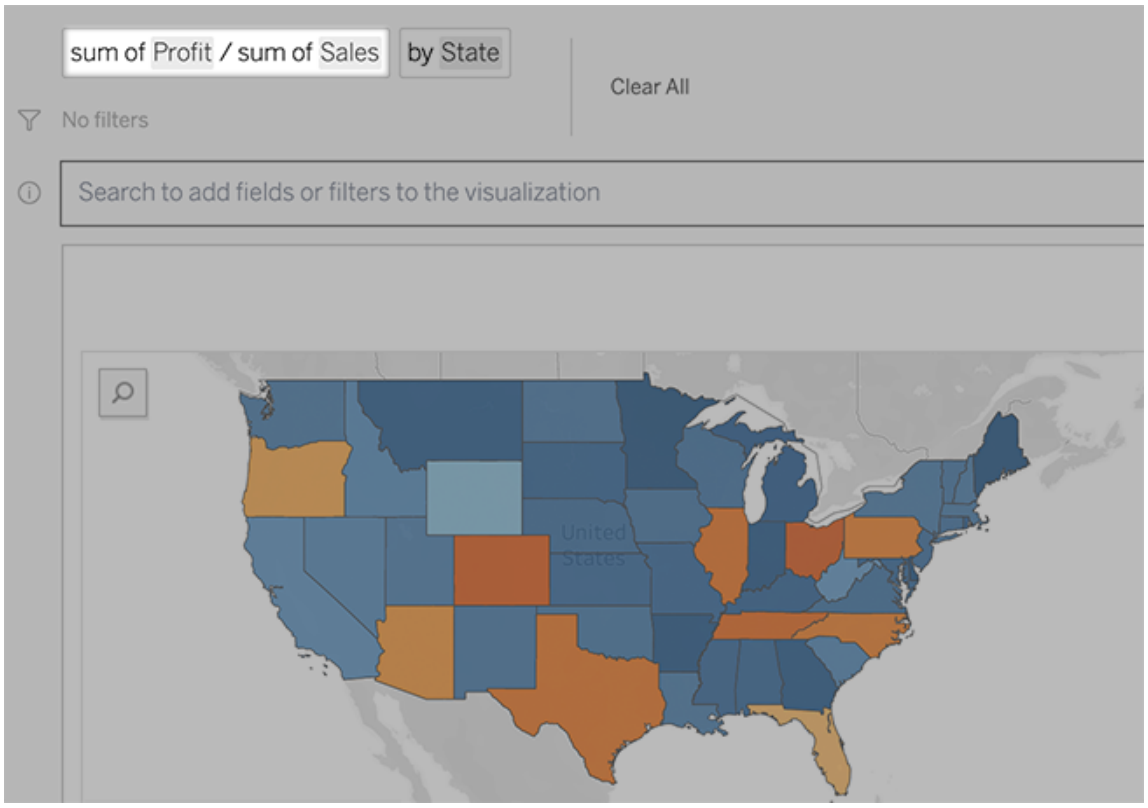
+ メジャーを合計する

- メジャーの差を計算する

***** 乗算を行う

/ 除算を行う

「データに聞く」(Ask Data) で保存されたワークブックでは、これらの計算は計算フィールドになる代わりに [列]、[行]、[マーク] シェルフでアドホック計算になります。

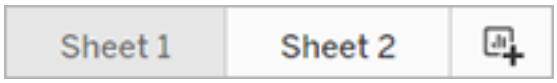


他の Viz を含むシートを追加する

レンズから複数の異なる Viz をすばやく作成するには、「データに聞く」(AskData) でシートを追加します。

Web ページの一番下で、次のいずれかを実行します。

- 名前付きシートの右側にある **[Add Sheet (シートの追加)]** アイコンをクリックします。

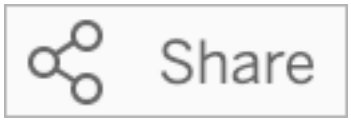
- シート名を右クリックし、**[Duplicate (複製)]** または **[削除]** を選択します

  (「データに聞く」(Ask Data) からシートの名前を変更するには、新しいワークブックに保存する必要があります)。

メール、Slack、または リンクを介して「データに聞く」(Ask Data) の Viz を共有する

「データに聞く」(AskData) の Viz は、レンズにアクセスできる人なら誰とでもすばやく共有できます。

1. ブラウザーの右上隅にある [Share (共有)] アイコンをクリックします。

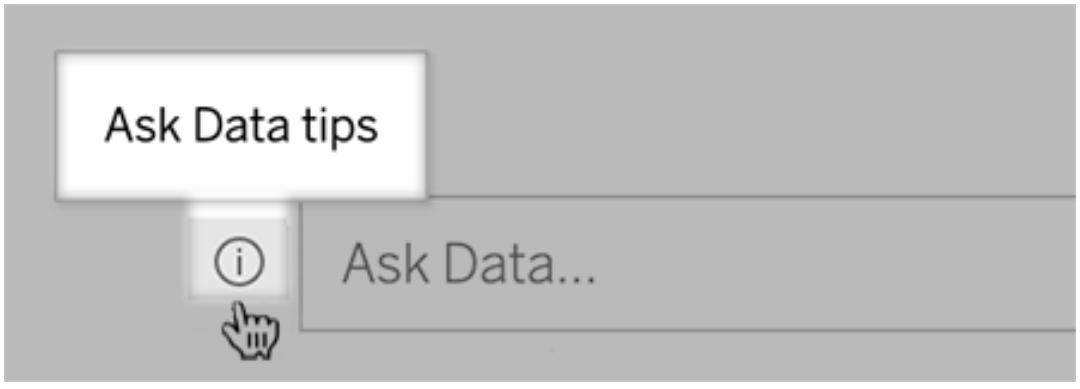


2. 次のいずれかを実行します。
   - メールまたは Slack で Viz を共有するには、テキストボックスに特定のユーザー名を入力します。(メールとSlack の統合は、Tableau 管理者によって事前に構成されている必要があります。)
   - カスタム メールやその他のメッセージに貼り付けることができるURL をコピーするには、**[リンクのコピー]** をクリックします。

レンズ所有者にフィードバックを送信する

レンズの構造や「データに聞く」の活用方法について質問がある場合、作者に直接フィードバックを送信できます。(このオプションは既定で有効になっていますが、レンズ作成者はそれを無効にすることができます)。

1. 「データに聞く」のクエリボックスの左側にある **[「データに聞く」のヒント]** のアイコンをクリックします。



2. ヒントダイアログの下部にある **[レンズ作成者に連絡]** をクリックします。

クエリを正常に実行するためのヒント

「データに聞く」(Ask Data) に対する質問を組み立てるときに、次のヒントを活用すると、より良い結果が得られます。

- **キーワードを使用する -** たとえば、「これらの空港がある国をすべて表示したい」ではなく「空港と国」で試してみてください。
- **フィールド名や値の正確な文字列を使用する —** たとえば、レンズに "Airport Code (空港コード)"、"Airport Name (空港名)"、"Airport Region (空港の地域)" フィールドが含まれる場合、これらを名前で指定します。
- **ランク付けされたリストを表示する -** 「データに聞く」では、「最高」や「最低」といった用語はそれぞれ上位 1 位と下位 1 位にマッピングされています。より広い範囲のランキングを表示する場合は、代わりに「高」や「低」を使用します。たとえば、「低価格の家」と入力します。
- **表計算のクエリ —** 表計算フィールドのクエリ式では、「前年差」をフィルター、制限、または含めることはできません。
- **通常よりも長い値を引用符で囲む —** 行の改行、タブ、11 個以上の単語を含む長いフィールド値を分析する場合は、それらを引用符で囲みます。パフォーマンスを向上させるため、「データに聞く」(Ask Data) は、その長さのフィールド、または最初の 200,000 の一意のフィールド値を超えるフィールドにはインデックスを作成しません。

特定の対象者に焦点を当てた「データに聞く」のレンズを作成する

「データに聞く」とメトリクスへの重要な変更点

Tableau の「データに聞く」機能とメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「*Tableau AI と Tableau Pulse によるデータエクスペリエンスの再構築*」を参照してください。

多くの人は、データソースにある全ての情報を必要としているわけではなく、自分の職務、たとえば、販売、マーケティング、サポートなどに関連するデータについての視覚化を望んでいます。

Tableau 作成者は、このようなさまざまな対象者の「データに聞く」(Ask Data)を最適化するため、選択したフィールドのサブセットに対してクエリを実行する個別の「データに聞く」(Ask Data) レンズを作成します。選択したフィールドには、フィールドの名前や値の同義語を指定することができます。たとえば「ロサンゼルス」を「LA」と言うように、レンズの利用者が一般的な言葉で使う用語を織り込んでおくことができます。レンズ作成者は、「データに聞く」(Ask Data) のクエリボックスの下に表示されるビジュアライゼーションの候補をカスタマイズすることもできます。ビジュアライゼーションの候補を利用すると、ユーザーは 1 回のクリックで答えを得ることができます。

注: 「データに聞く」レンズは、Tableau サイトに個別にパブリッシュされたデータソースに対してのみ作成できます。ワークブックに埋め込まれたデータソースや仮想接続を使用するデータソースに対しては、レンズを作成することはできません。

Tableau サイトでレンズのページを作成または構成する

Tableau サイトには、レンズごとに個別のページがあります。そのページで、ユーザーは「データに聞く」に問い合わせることができ、作成者はレンズのフィールドや同義語、質問の候補などを構成できます。

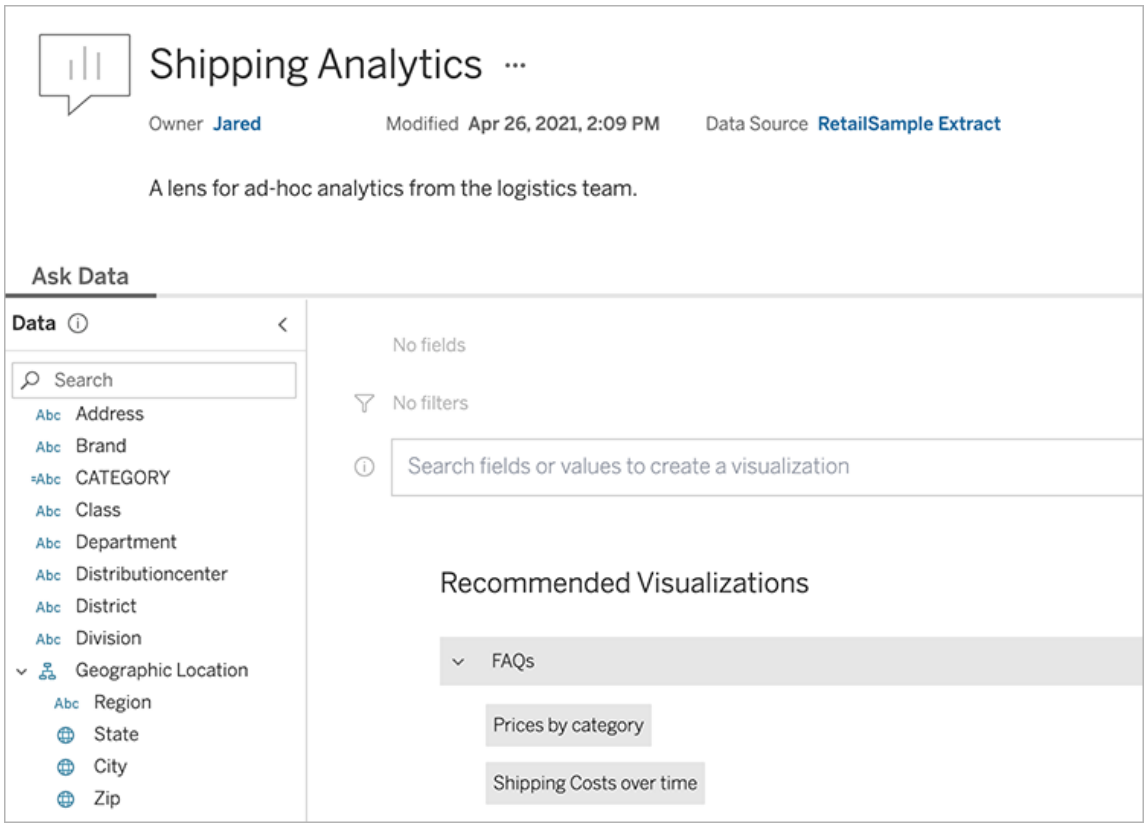
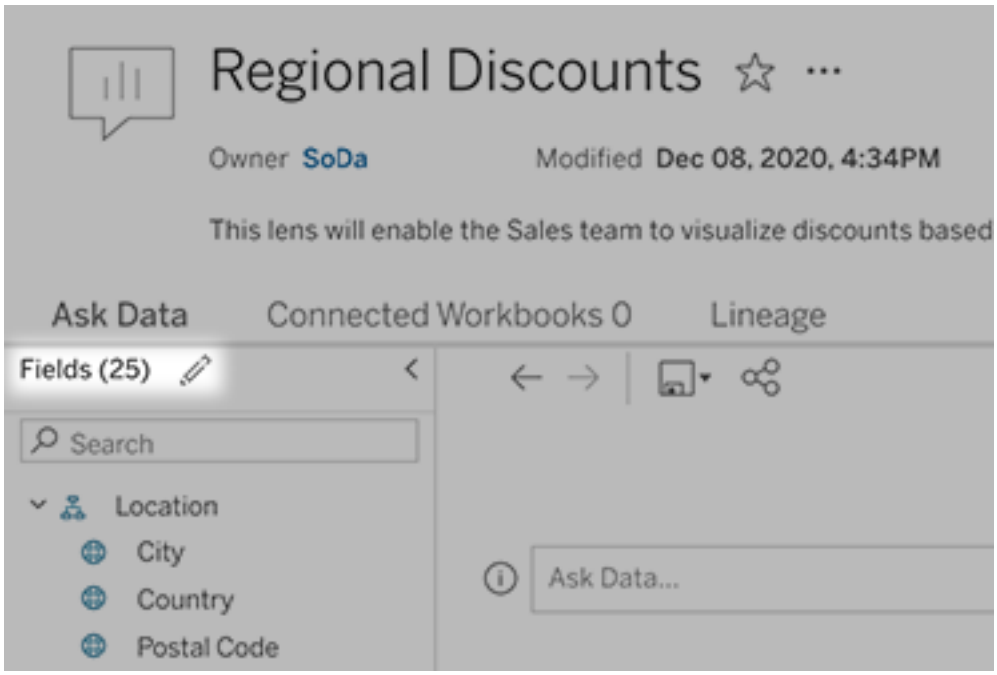


Tableau サイトのレンズ ページ

1. Tableau サイトにレンズ ページを作成するには、データ ソース ページに移動し、**[新規] > [「データに聞く」(Ask Data) レンズ]** を選択します。

   既存のレンズを構成するには、サイトのレンズ ページに移動します。(または、ダッシュボードの「データに聞く」(Ask Data) オブジェクトで上隅のポップアップ メニューをクリックし、**[レンズ ページに移動]** を選択します。)

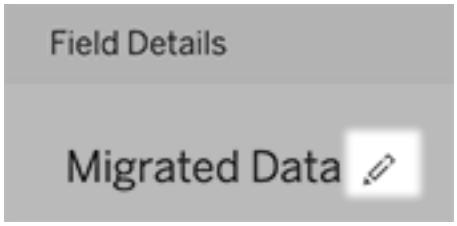
2. 新しいレンズを作成する場合は、名前、説明、プロジェクトの場所を入力して、**[レンズのパブリッシュ]** をクリックします。
3. 左側の [フィールド] ペインの上部で、鉛筆のアイコンをクリックします。レンズのユーザーに関係のあるフィールドを選択し、**[保存]** をクリックします。



4. 左側で、個々の表またはフィールドにカーソルを合わせ、鉛筆アイコンをクリックします。

その後、次のいずれかを実行します。

- 右側の鉛筆アイコンをクリックして、より代表的な名前を付けます。



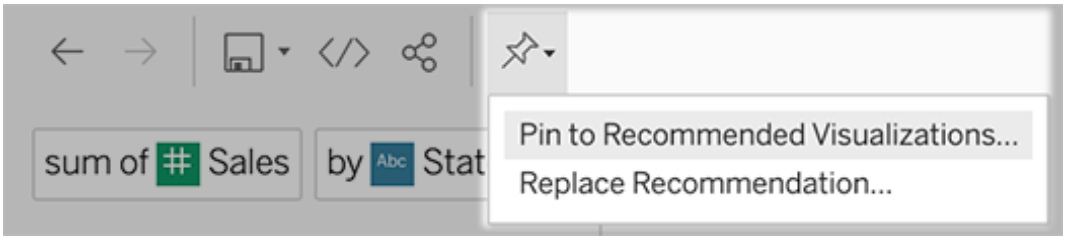
- レンズ ユーザーがクエリに入力できるフィールド名と値の一般的な同義語を追加します。
- ユーザーがフィールドにカーソルを合わせたときに表示される説明を編集します。

推奨ビジュアライゼーションのリストを変更する

レンズ ユーザーからの一般的なクエリに対処するために、クエリボックスの下に表示される推奨ビジュアライゼーションをカスタマイズできます。

推奨ビジュアライゼーションを追加または置換する

1. テキストボックスにクエリを入力し、Enter キーまたは Return キーを押します。
2. ビジュアライゼーションが表示されたら、ツールバーのピン アイコンから、**[Pin to Recommended Visualizations (推奨ビジュアライゼーションにピン留めする)]** または **[Replace Recommendation (推奨を置換する)]** を選択します。

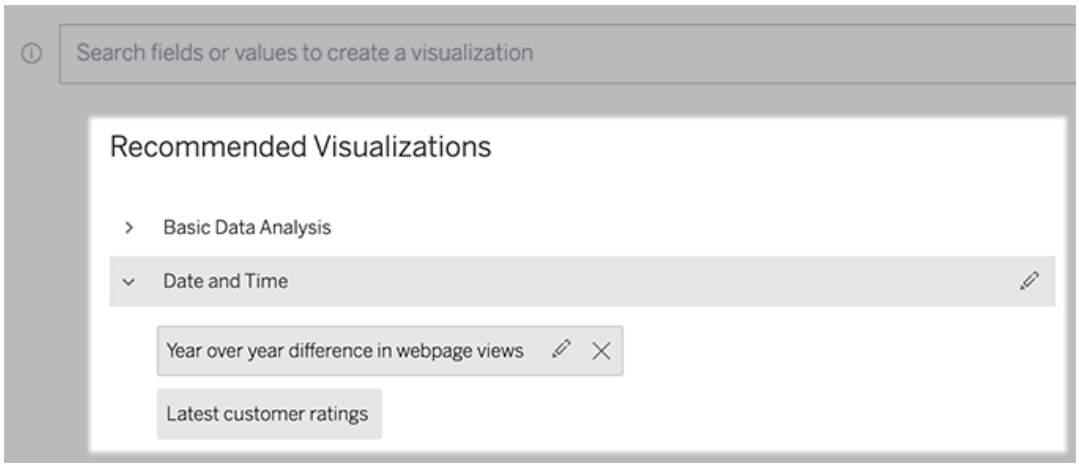


3. 推奨を新たに追加する場合は、名前を入力し、それを表示するセクションを選択します。推奨を置換する場合は、上書きする既存の推奨を選択してください。

セクションのタイトルと推奨ビジュアライゼーションの名前を編集するか、推奨を削除する

- セクション タイトルを編集するには、タイトルの右にある鉛筆アイコン をクリックします。
- 既存の推奨を変更するには、カーソルを合わせて鉛筆アイコン をクリックします。推奨

を削除するには、[X] をクリックします。

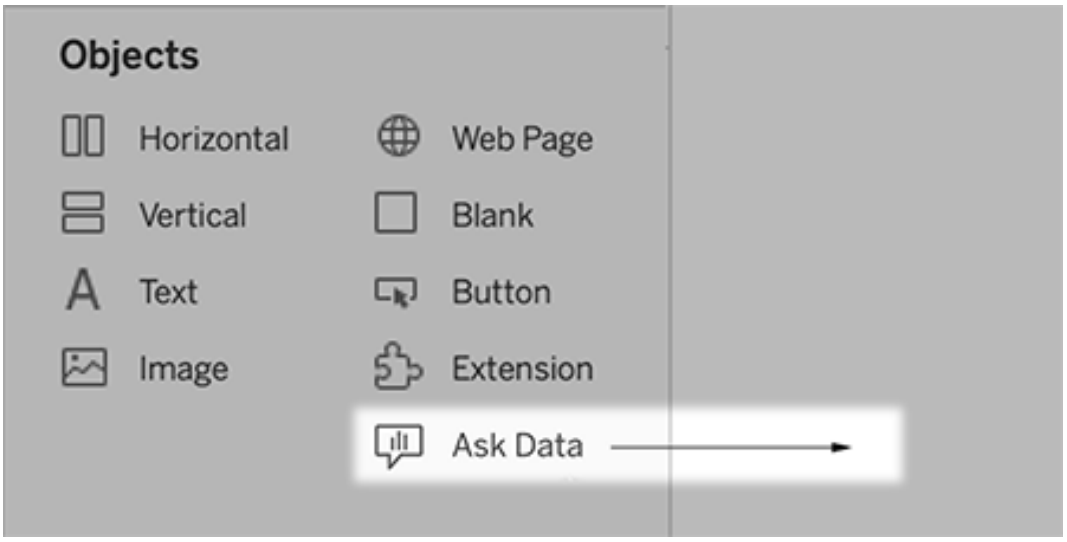


ダッシュボードに「データに聞く」レンズを追加する

ダッシュボードに「データに聞く」のオブジェクトを追加して、パブリッシュされたデータソースを Tableau サイトのレンズでクエリできるようにします。

1. Tableau Cloud または Tableau Server でダッシュボードを編集しているときに「データに聞く」のオブジェクトをキャンバスにドラッグします。

> **注:** Tableau Desktop でも、「データに聞く」のオブジェクトをキャンバスにドラッグして、配置することもできます。ただし、レンズを選択するには、Tableau Cloud または Tableau Server にパブリッシュし、そこでオブジェクトを編集する必要があります。

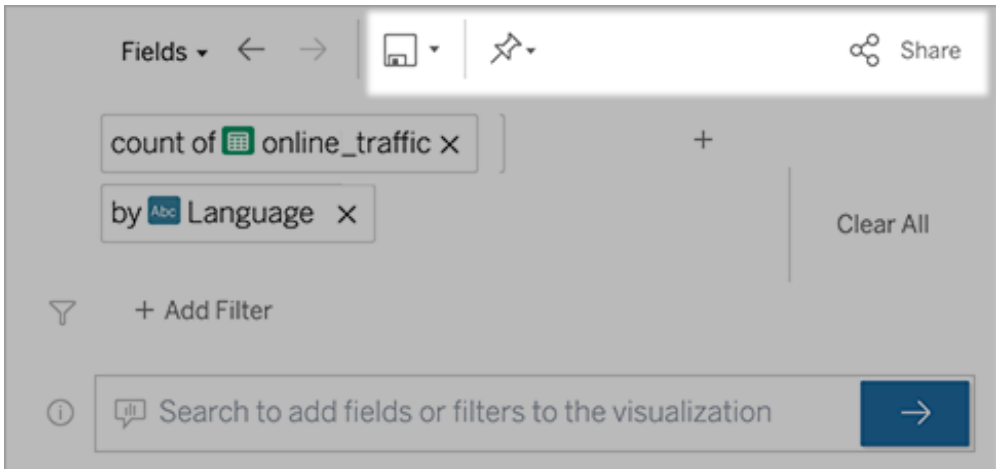
2. 

3. 以前にワークブックに接続されたパブリッシュ済みのデータソースを選択します。
4. 既存のレンズを使用するには、レンズを選択して、**[レンズを使用]** をクリックします。

   新しいレンズを作成するには、次のいずれかの操作を行います。

   - データソースのレンズがない場合は、**[データ ソースのページに移動]** をクリックします。
   - レンズがすでにある場合は、ダイアログの下部にあるデータソース名をクリックします。
5. (新しいレンズのみ) Tableau サイトでレンズのページを作成または構成するの手順を実施します。
6. **[Toolbar Options for Lens Users (レンズ ユーザーのツールバー オプション)]** で、ユーザーが使用できるボタンを選択します。
   - **[Add Visualization to Pins (ビジュアライゼーションをピンに追加)]** を使用すると、ユーザーはクエリボックスのすぐ下に表示される推奨ビジュアライゼーションのリストに追加できます。
   - **[Publish as Workbook (ワークブックとしてパブリッシュ)]** を使用すると、ユーザーは Tableau サイトにビジュアライゼーションをワークブックシートとして保存できます。
   - **[Share Visualization (ビジュアライゼーションの共有)]** を使用すると、ユーザーはメール、Slack、またはリンクを介して共有できます。

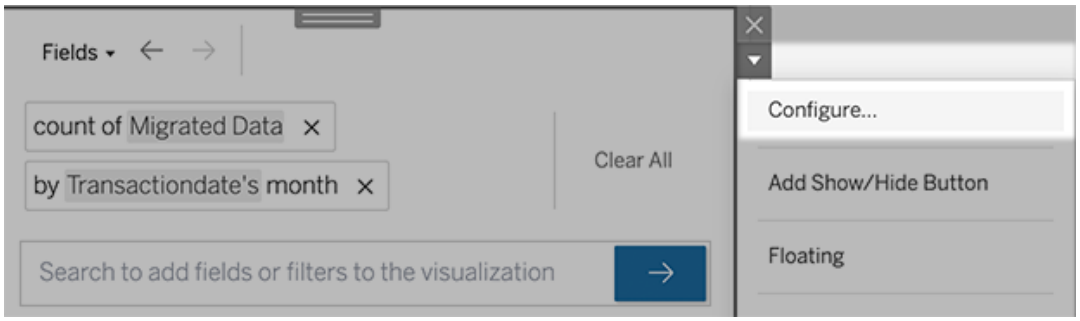
     レンズでは、パブリッシュ(保存アイコン)、ピン留め、共有のオプションが右上隅に表示されます。



7. (新しいレンズのみ) レンズを作成したら、ダッシュボードの [レンズ] オブジェクトに戻り、**[更新]** をクリックします。次に、新しいレンズを選択し、**[レンズを使用]** をクリックします。

ダッシュボードの「データに聞く」オブジェクトに別のレンズを適用する

1. オブジェクトの上部のポップアップ メニューから、**[構成]** を選択します。



2. ダッシュボードに「データに聞く」レンズを追加するに移動し、手順 2 以降を繰り返します。

レンズの名前、説明、プロジェクトの場所を変更する

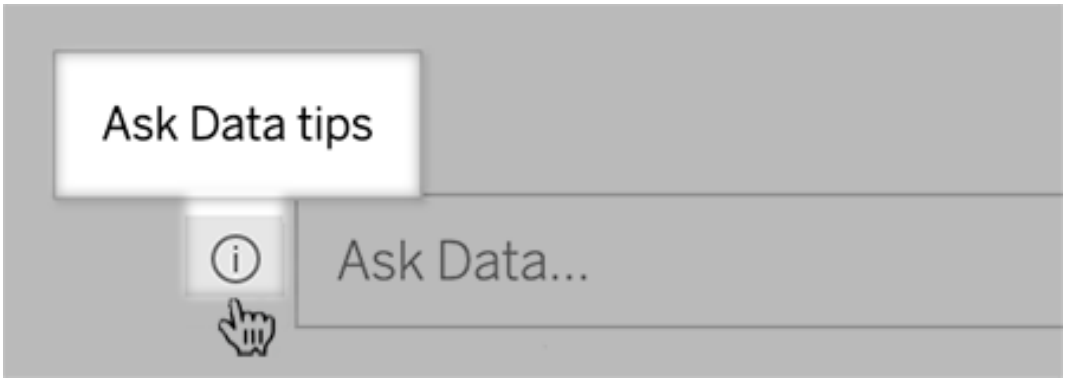
1. Tableau Cloud または Tableau Server でレンズのページに移動します。
2. ページ上部のレンズ名の右側にある 3 つのドット (...) をクリックし、**[ワークブックの編集]** を選択します。
3. **[レンズの詳細を編集]** をクリックします。

レンズを通した「データに聞く」の利用状況を確認する

データソースの所有者またはレンズ作成者である場合は、「データに聞く」(Ask Data) のダッシュボードで、よく利用されるクエリやフィールド、ビジュアライゼーションがクリックされた回数などの有用な情報を確認できます。フィルターを使用して、特定のユーザーや時間範囲にデータを絞り込むことができます。これらの統計情報を活用してレンズをさらに最適化すると、ユーザーの体験を向上できます。

注: Tableau Server を使用している場合は、Tableau Server リポジトリでこのデータにアクセスすることができ、カスタム ダッシュボードを作成できます。

1. Tableau Server または Tableau Cloud で、レンズのページに移動します。
2. 「データに聞く」(Ask Data) テキスト ボックスの左側にある [Ask Data tips (「データに聞く」(Ask Data) のヒント)] アイコンをクリックします。



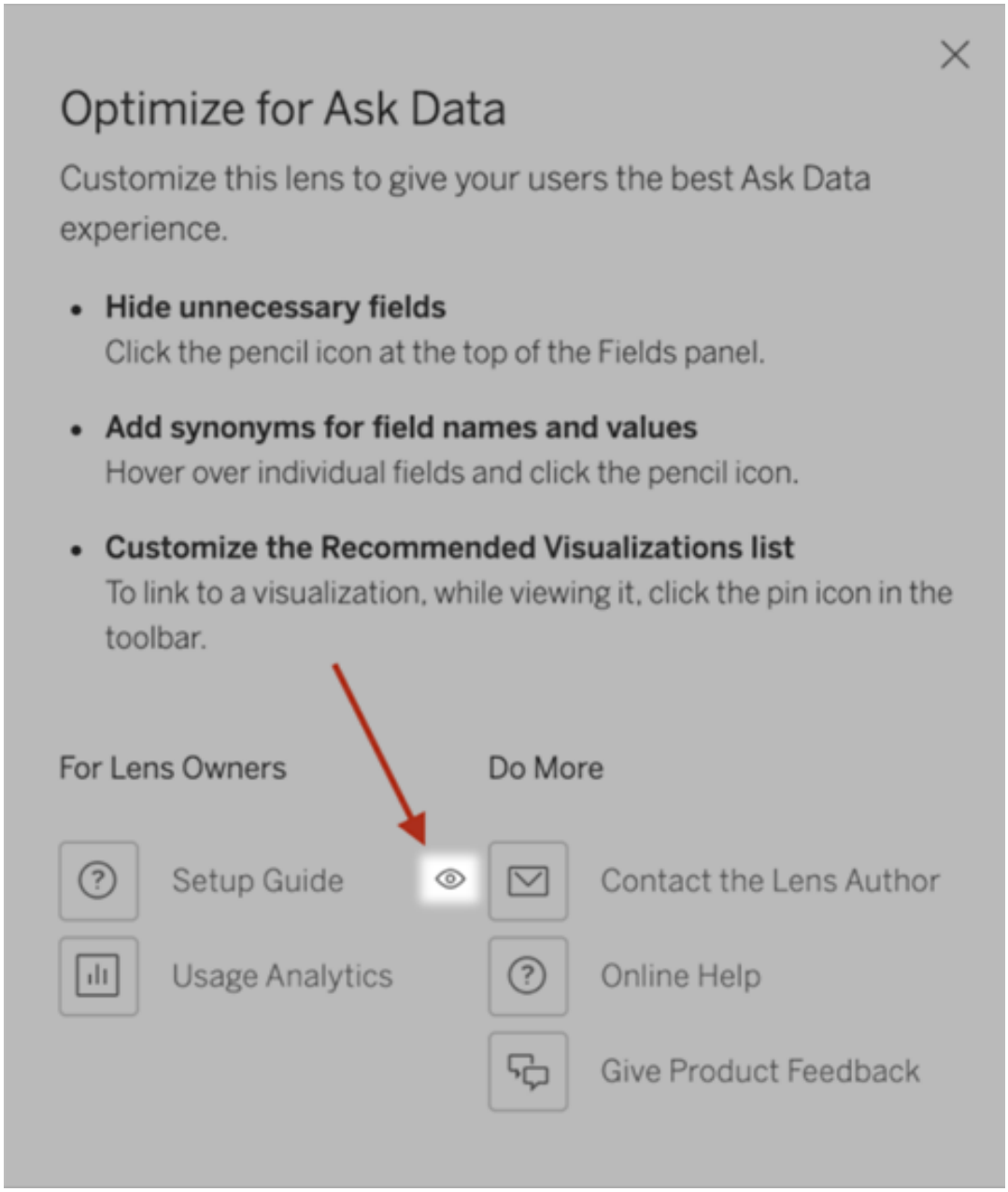
3. ヒントダイアログの左下隅にある **[使用状況分析]** をクリックします。

ユーザーがレンズに関する質問をメールで送信できるようにします。

レンズ所有者は、ユーザーがデータ構造や期待される結果などに関する質問をメールで送信できるようにすることもできます。このオプションは既定でオンになっていますが、以下のステップを使用してオフにすることができます。

1. Tableau Server または Tableau Cloud で、レンズのページに移動します。
2. 上記のレンズを通した「データに聞く」の利用状況を確認する] に表示されている、「データに聞く」(Ask Data) テキスト ボックスの左側の [i] をクリックします。

3. ヒントダイアログの下部で、[Contact the Lens Author (レンズ作成者に連絡)] の横にある目のアイコンをクリックして、フィードバックを有効または無効にします。



レンズのパブリッシュと表示のパーミッション

ダッシュボードの「データに聞く」オブジェクトについては、パーミッションを変更する必要はありません。デフォルトで、既存のワークブック作成者がレンズを作成でき、既存のダッシュボード利用者がレン

ズを表示できます。ただし、データソースページを介したダッシュボードと直接アクセスについては、必要なレンズのパーミッションについて、概要を参考までに以下に示します。

レンズの作成とパブリッシュに必要な権限:

- 作成者またはエクスプローラーのユーザーロール
- データソースに対するレンズ作成パーミッション(デフォルトでは接続パーミッションから継承されます)
- レンズをパブリッシュする親プロジェクトに対する書き込みパーミッション

パブリッシュされたレンズへのアクセスと操作に必要な権限:

- 閲覧者以上のロール
- データソースに対する接続パーミッション
- レンズに対する閲覧パーミッション

注: デフォルトでは、閲覧などのレンズのパーミッションは、ワークブックに対するプロジェクトのパーミッションが反映されます。Tableau 管理者がデフォルトのレンズのパーミッションを変更したい場合、各プロジェクトに対して個別に変更することも、パーミッション API を使用して一括で変更することもできます。

サイトでの「データに聞く」(Ask Data) 機能の無効化または有効化

「データに聞く」とメトリクスへの重要な変更点

Tableau の「データに聞く」機能とメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「*Tableau AI と Tableau Pulse によるデータエクスペリエンスの再構築*」を参照してください。

既定で「データに聞く」(Ask Data) はサイトに対して有効になっていますが、Tableau 管理者はそれを無効にすることができます。

1. **[全般]** サイト設定に移動します。
2. (Tableau Server のみ) **[Web 作成]** セクションで、**[Let users edit workbooks in their browser (ユーザーが自分のブラウザーでワークブックを編集するのを許可する)]** を選択します。
3. **[Availability of Ask Data (「データに聞く」の可用性)]** セクションで、次のオプションから選択します。
    - **[有効]** にすると、パブリッシュされたすべてのデータソースに対して「データに聞く」(Ask Data) のレンズを作成できます。
    - **[無効]** にすると、サイト全体で「データに聞く」(Ask Data) が非表示になりますが、以前作成したレンズに関する情報は保持されるため、「データに聞く」(Ask Data) を再度有効にする場合はレンズを復元できます。

注: バージョン 2019.4.5 以降、Tableau サーバー管理者は「データに聞く」(Ask Data) を既定で有効にするかどうかを設定できます。

「データに聞く」(Ask Data) 機能でのデータの最適化

「データに聞く」とメトリクスへの重要な変更点

Tableau の「データに聞く」機能とメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「*Tableau AI と Tableau Pulse によるデータエクスペリエンスの再構築*」を参照してください。

データソースを管理してパブリッシュする際、以下のいくつかのヒントは「データに聞く」(Ask Data) をユーザーがさらに上手に使用するのに役立ちます。このプロセスに少し時間を割くと、組織の幅広い人員がデータ分析を使用し始めることができ、自主的に答えを得たり、より深い洞察を得たりするのに役立ちます。

「データに聞く」(Ask Data) 機能でのデータの最適化

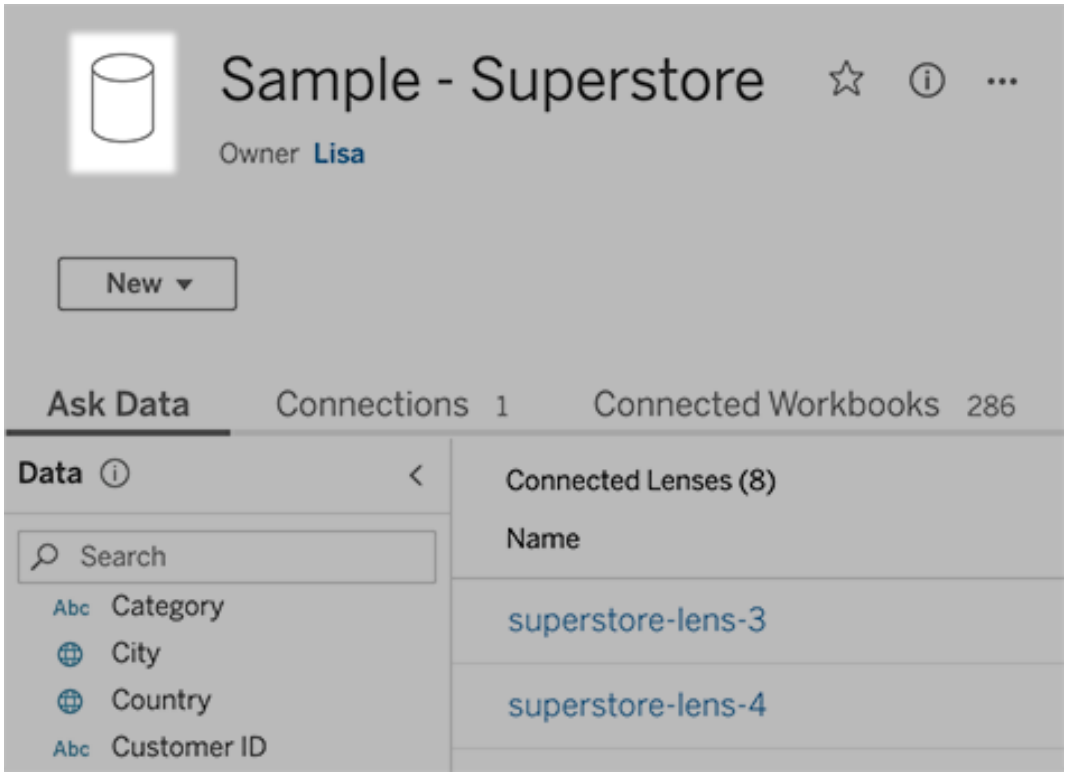
データソースの所有者は、「データに聞く」(Ask Data) インターフェイスの左側にある [データ] ペインで、フィールドの同義語を追加したり、無関係な値を除外したりすることができます。

データソースまたはレンズレベルでの設定の変更

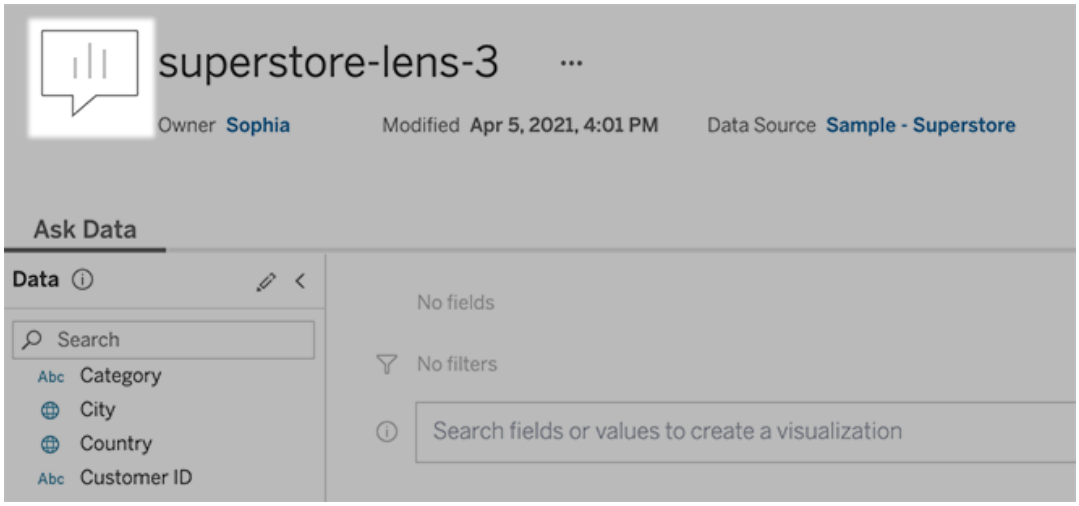
「データに聞く」(Ask Data) の [データ] ペインで設定を変更するときは、データソースレベルとレンズレベルのどちらで変更するのかをしっかり確認してください。(詳細については、「特定の対象者に焦点を当てた「データに聞く」(Ask Data) のレンズを作成する」を参照してください。)

- データソースレベルでは、左上隅に円筒形のデータソースアイコンが表示されます。[データ] ペインで行った変更内容は、既定で、その後に作成されるすべてのレンズに適用されます。

> **注:** 抽出の場合、シリンダーが 2 つ表示されます。



- 個々のレンズについては、左上隅に引用符アイコンが表示されます。[データ] ペインで行った変更内容は、このレンズにのみ適用されます。



フィールドの名前と値に対応する同義語の追加

データソースにある用語と同じ用語を使用しないユーザーもいるかもしれません。このため、データソース所有者や Tableau 管理者は、特定のデータフィールドの名前と値に対応する同義語を追加できます。入力する同義語は組織全体が使用するため、全員がデータ分析をすばやく簡単に行えるようになります

検索結果から特定のフィールド値を除外する

「データに聞く」(Ask Data) における検索結果の有用性を向上させるために、特定のフィールド値をインデックス作成から除外することができます。「データに聞く」(Ask Data) では、インデックスが作成されていない値は検索結果に追加されませんが、関連する値は引き続きビジュアライゼーションに表示されます。たとえば、検索結果に不要な詳細が追加されないようにするため、"製品" フィールドの値からインデックスを作成しない場合でも、「データに聞く」(Ask Data) では結果データのビジュアライゼーションに "iPhone 12" などの値が表示されます。また、ユーザーは、インデックスが作成されていない値を引用符で囲んでクエリに手動で追加できます ("iPhone 12" を含む製品の売上など)。

注意: データソースのインデックス作成設定の値が [無効] に設定されている場合、このフィールドレベルの設定は無視されます。フィールド名と関連する同義語には、常にインデックスが作成されます。

1. データソースまたは個々のレンズの [「データに聞く」(Ask Data)] タブに移動します。
2. 左側のソースフィールドにカーソルを合わせ、**[フィールドの詳細を編集]** アイコン (鉛筆) をクリックします。

3. **[Index field values (フィールド値のインデックス作成)]** の選択を解除します。

   表示されるテキストボックスをクリックしてデータソースのインデックスをもう一度作成するか、通常のインデックス作成スケジュールに基づいてインデックスをもう一度作成します。

データソースを最適化する

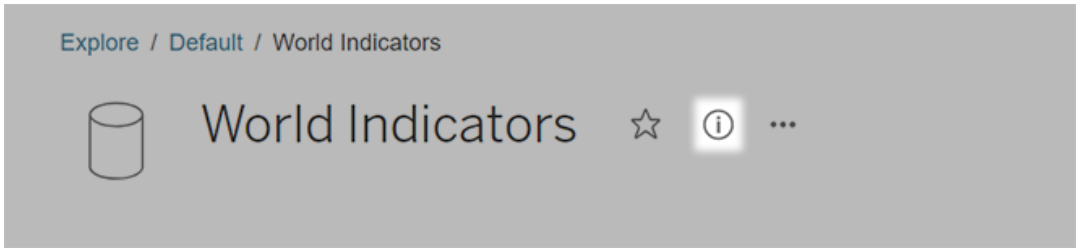
「データに聞く」(Ask Data) で最適なエクスペリエンスが得られるようにするには、元のデータソースを最適化します。

注: 「データに聞く」では、多次元キューブデータソースや、Google アナリティクスのようなリレーショナルデータソースではないデータソース、仮想接続を使用するデータソースはサポートされていません。

「データに聞く」(Ask Data) のインデックス作成を最適化する

データソースの所有者は、「データに聞く」(Ask Data) のフィールド値のインデックス作成の頻度を変更して、システムパフォーマンスを最適化することができます。

1. [データソース] ページの上部で [詳細] アイコンをクリックします。



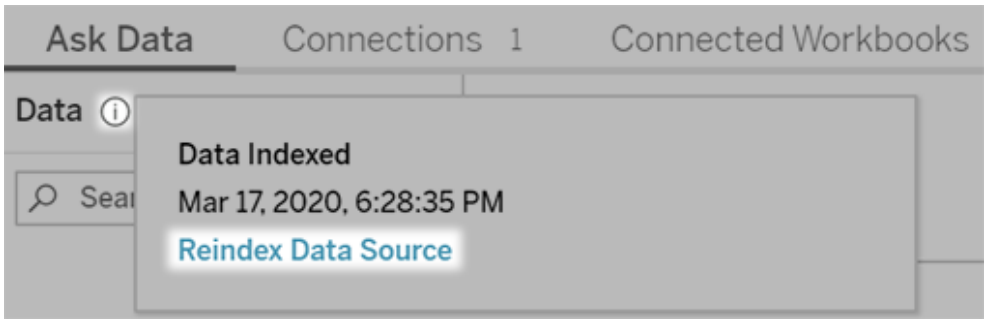
2. 「データに聞く」(Ask Data) セクションで、**[編集]** をクリックします。

3. フィールド値のインデックス作成のオプションを選択します。

   - **[自動]** にすると、ライブの場合、更新された抽出がある場合、または再パブリッシュされた場合に、変更があるかを24時間ごとに確認し、データソースを分析します。このオプションは、「データに聞く」(Ask Data) を使用することが多いデータソースで選択し、ユーザーがクエリを発行する前に使用できるようにしてください。

   - **[手動]** にすると、Tableau 作成者がデータソースページにインデックス作成をトリガーする場合にのみ、データソースを分析します。このオプションは、データソースに頻繁に変更があるものの、ユーザーがたまにしか「データに聞く」(Ask Data) を使用してクエリを発行しない場合に選択してください。

     手動のインデックス作成をトリガーするには、[データソース] ページに移動して、左側の [データ] ペインで円で囲まれた "i" をクリックし、**[Reindex Data Source (データソースのインデックス再作成)]** をクリックします。



- **[無効]** にすると、値ではなくフィールド名のみが分析されます。

4. **[保存]** をクリックします。

データ抽出を使用してパフォーマンスを向上させる

パフォーマンスを向上させ、大規模なデータセットをサポートするには、ライブ データ ソースではなく、パブリッシュされた抽出と共に「データに聞く」を使用します。詳細については、「抽出の作成」を参照してください。

ユーザーがデータ ソースにアクセスできるようにする

ユーザーが「データに聞く」(Ask Data) を使用するには、個々のデータ ソースに接続するためのパーミッションが必要です。データ ソースに行レベルのパーミッションがある場合、それらのパーミッションは「データに聞く」(Ask Data) にも適用されますが、安全な値は認識されず、関連する統計的な推奨事項もありません。

サポートされていないデータ ソース機能に注意する

「データに聞く」(Ask Data) では、以下を除くすべての Tableau データ ソース機能がサポートされています。データ ソースにこれらが含まれている場合、「データに聞く」(Ask Data) のユーザーは関連フィールドのクエリを実行できません。

- セット
- 組み合わせフィールド
- パラメーター

ユーザーの質問を予測する

ユーザーがしそうな種類の質問を予測し、以下の技術を使用して、これらの質問に合わせてデータソースを最適化します。

- Tableau Prep や類似ツールでデータをクリーニングまたは加工します。
- データを結合し、ある表でユーザーが質問を持つかもしれないすべてのフィールドを含め、パフォーマンスを向上させます。
- ユーザーのよくある質問に回答する計算フィールドを追加します。
- ヒストグラムまたはビニングされた別の形状としてユーザーが表示する可能性が高い定量的変数に対し、適切なサイズのビンを作成します。

データを簡素化する

ユーザーと「データに聞く」(Ask Data) の両方がデータを理解しやすくなるよう、データ準備プロセスで可能な限りデータソースを簡素化します。

1. パフォーマンスを向上させるため、不要なフィールドは削除します。
2. 各フィールドに固有の意味のある名前を付けます。
    - たとえば、冒頭に “Sales … (売上高 …)” が付くフィールド名が 5 つある場合、「データに聞く」(Ask Data) が “sales (売上高)” という用語を適切に解釈できるよう、適切な区別を付けておきます。
    - “Number of records (レコード数)” という名前も、より分かりやすいものに変更します。たとえば、各レコードが地震であるデータソースでは、“Number of earthquakes (地震数)” を使用します。
    - フィールド名が数字、日付、ブール (“true” または “false”) 値にならないようにします。
    - “Sales in 2015 (2015 年の売上高)” や “Average Products Sold (平均販売製品数)” などの分析表現に類似した名前は使わないようにします。
3. 意味のあるフィールド値の別名を作成し、ユーザーが会話で用いる用語を反映させます。

フィールドの適切な既定の設定

「データに聞く」(Ask Data) がデータを正しく分析できるよう、既定のフィールド設定が各フィールドのコンテンツを反映するようにしてください。

- テキスト、時刻、日付、地理的値、他の値に対してデータ型を設定します。
- 適切なデータの役割を割り当てます: ディメンションまたはメジャー、連続または不連続。
- Tableau Desktop で各メジャーに対し、色、並べ替え順序、数値形式 (パーセンテージ、通貨など)、集計関数などの適切な既定の設定を割り当てます。たとえば、"Sales (売上) に

は SUM が適切ですが、"Test Score (テストスコア)" には AVERAGE が既定としてより適している場合があります。

> **ヒント:** 既定のコメントではユーザーが「データに聞く」(Ask Data) でフィールドにカーソルを合わせると説明情報が表示されるため、各フィールドで既定のコメントを設定すると特に役立ちます。

地理的フィールドとカテゴリー フィールドでの階層の作成

時刻データでは、Tableau は自動的に階層を作成するため、ユーザーは簡単に Viz をドリルアップおよびドリルダウンすることができます (日から週、月など)。ただし、地理的データおよびカテゴリーデータでは、カスタム階層を作成して、フィールド間の関係を反映するビジュアライゼーションを「データに聞く」(Ask Data) が生成しやすくすることをお勧めします。「データに聞く」(Ask Data) ではデータペインに階層が表示されません。

「データに聞く」(Ask Data) は階層名のインデックスを作成せず、階層内のフィールド名でのみインデックスを作成します。たとえば、[Location (場所)] という名前の地理的階層に [Country (国)] フィールドと [City (市区町村)] フィールドが含まれている場合、ユーザーは「データに聞く」(Ask Data) の質問に、「Country (国)」と「City (市区町村)」を入力する必要があります。

Tableau データストーリーを作成する (英語のみ)

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

Tableau ダッシュボードのエグゼクティブ サマリーを書いたことのある方は、それが時間のかかる作業であることをご存じでしょう。どのインサイトを共有するか選ぶだけでも時間がかかり、データが更新されるたびに要旨を書き直す必要もあります。Tableau データストーリーを利用するとダッシュボード内に物語風のインサイトを自動的に生成することができ、時間を節約できるだけでなく、関連す

るインサイトを明らかにできます。ダッシュボード内の Viz を探索すると、データストーリーによって書かれたストーリーが調整され、データをより深く掘り下げて、重要なインサイトをより迅速に特定できるようになります。

Tableau で作業しているとき、すばやくその場所で**データストーリー**オブジェクトをダッシュボードに追加できます。また、ストーリーで使用する用語や指標をカスタマイズできるので、ビジネスで使用していることばを使ってデータストーリーに語らせることができます。

現在は、Tableau を使用している場所であればどこからでも Tableau データストーリーを書いたり見たりすることができます。データストーリーを作成すると、Tableau Mobile で表示できるようになります。ただし、ダッシュボードを PDF などにエクスポートする場合、データストーリーは含まれません。

データストーリーがデータを扱う方法を理解する

データストーリーを書くために、Tableau は Tableau Cloud または Tableau Server 環境でホストされるサービスを使用します。ダッシュボードに Tableau データストーリーを追加するしたり、ダッシュボードからデータストーリーを表示したりすると、Tableau は、Cloud のセキュリティおよび Tableau Server のセキュリティで概説されているセキュリティ標準を使用して、関連するワークシートのデータをログインしている環境 (つまり、Tableau Cloud サイトまたは Tableau Server インスタンス) に送信します。データストーリーは、Tableau を使用している場所であればどこからでも書いたり見たりすることができます。

データストーリーの書き方について学ぶ

Tableau データストーリーは、ルールベースのテンプレート化された自然言語生成 (NLG) を搭載しています。データストーリーは自動分析を実行して、基本的な計算からより高度な統計に至るまで、参照元データについて関連性のある正確な事実を判断します。ストーリーを書くために、データストーリーは事前定義された言語テンプレートのライブラリを使用して、これらの事実を自然言語のインサイトに合成します。データストーリーは、接続されている Tableau ワークシートからの最新の概要データを使用して、ランタイムにこれらのテンプレートを処理します。カスタム言語機能を活用して、独自の言語テンプレートの生成、機能の追加、ビジネスルールの定義を行うことで、より関連性が高く状況に応じたデータストーリーを構築することができます。

注: データストーリーでは、インサイトやストーリーを作成する際に、生成 AI、大規模言語モデル (LLM)、または機械学習は使用しません。

サイトのデータス ストーリーを管理する

Tableau 管理者は、サイトで Tableau データストーリーを使用するかどうかを選択できます。データストーリーは既定でオンになっています。

1. Tableau サイトにサインインします。
2. 左側のペインで、**[設定]** を選択します。
3. **[全般]** タブで、**[Availability of Data Stories (データストーリーの可用性)]** セクションまでスクロールします。
4. データストーリーを**オン**にするか**オフ**にするかを選択します。

注: データストーリーがオフになっている場合、機能をオンに戻すと、ダッシュボードに既に存在していたデータストーリーが復元されます。

ダッシュボードに Tableau データストーリーを追加する

Tableau データストーリーの重要な変更

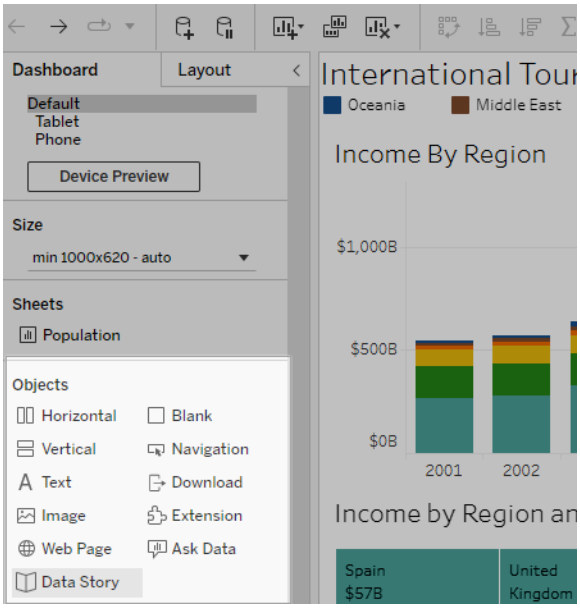
Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

ダッシュボードを作成した後、**データ ストーリー** オブジェクトをダッシュボードに追加すると、viz について自然言語で書かれた洞察を表示できます。現在、Tableau データストーリーは英語のみで書かれており、Tableau Cloud、Tableau Server (バージョン 2023.1 以降)、および Tableau Desktop で利用できます。**データ ストーリー**を作成する際、データサイズに制限はありません。ただし、大量のデータを分析しようとすると、45 秒でストーリーの生成がタイムアウトしてしまいます。

データ ストーリーは、データ ポイントが 1,000 以下のビジュアライゼーションで使用することをお勧めします。

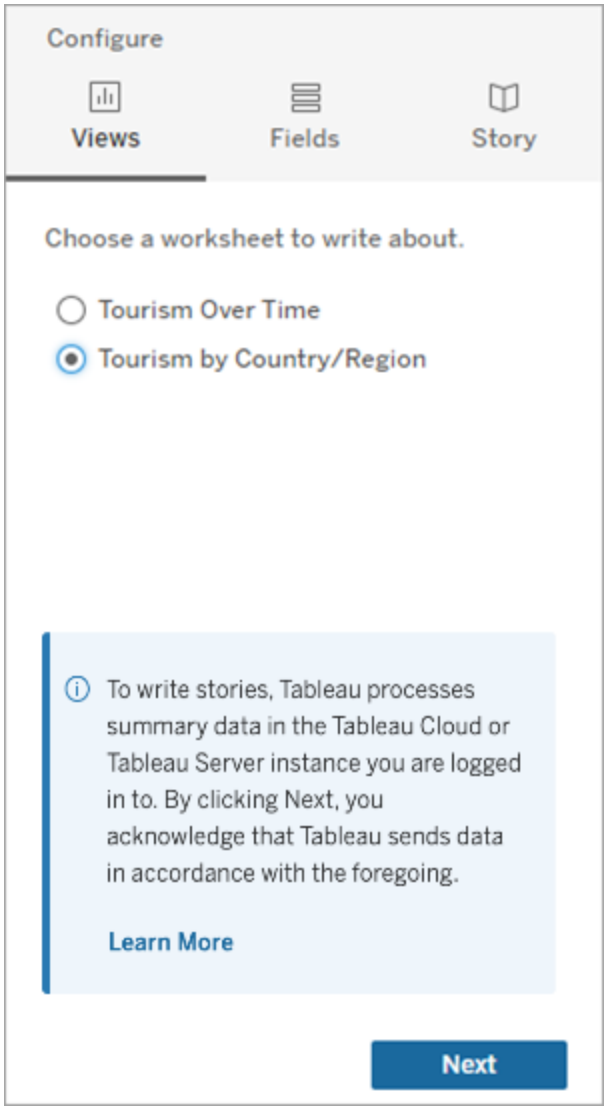
> **注:** Tableau データ ストーリーはポップアップ ウィンドウで開くため、必ずポップアップを許可してください。全画面モードを使用している場合は、データ ストーリーを新しいタブで開くことができます。

1. **データ ストーリー** オブジェクトを Tableau ダッシュボードにドラッグします。まだ行っていない場合は、ダッシュボードにシートを追加して Tableau データ ストーリーを使用します。

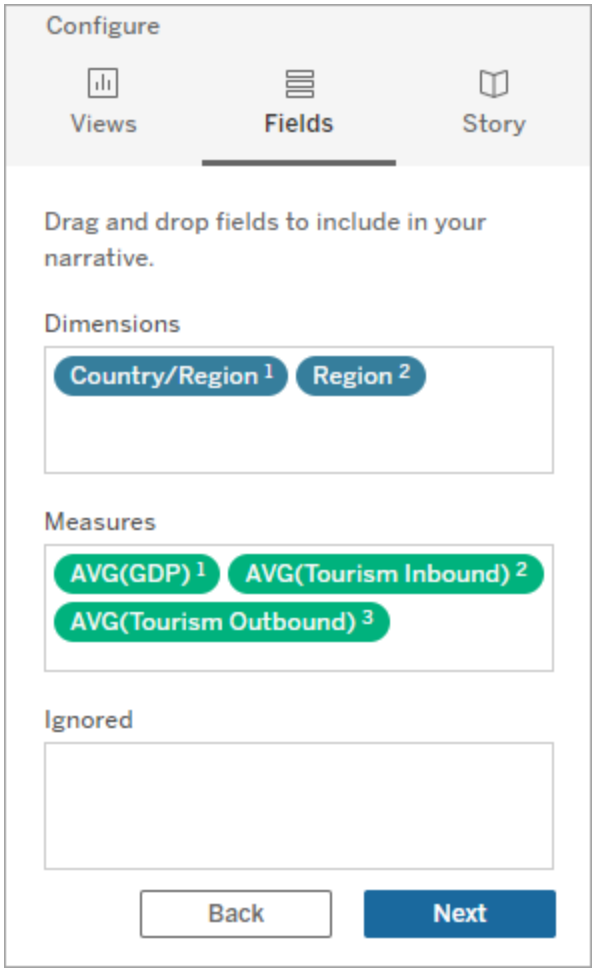


2. [データ ストーリー] ダイアログ ボックスで、説明する対象のワークシートを選択してストーリーを設定します。**[次へ]** をクリックすると、Tableau は関連するすべてのワークブック データを、

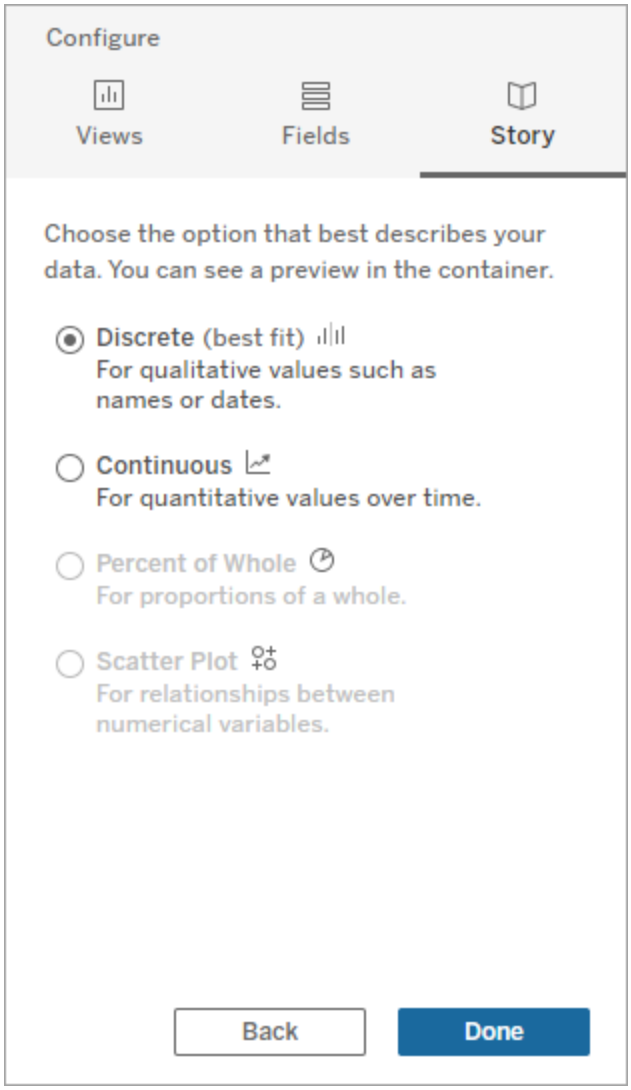
ユーザーがログインしている Tableau Cloud または Tableau Server に送信します。



3. ストーリーに含めるディメンションとメジャーを選択します。



4. データを最もよく説明するストーリーのタイプを選択します。



[不連続] は、名前や日付などの定性的な値に適しています (横棒グラフや縦棒グラフなど)。
[連続] は、時間の経過に伴う定量的な値に適しています (折れ線グラフなど)。
[全体に占める割合] は、全体に対する比率に適しています (円グラフなど)。
[散布図] は、数値間の関係に適しています (散布図など)。

5. **[完了]** をクリックします。

ビジュアライゼーションのメニューを開いて **[フィルターとして使用]** をクリックすると、ビジュアライゼーションのさまざまなセクションをクリックして、データストーリーをフィルタリングすることができます。

ストーリーを生成後、**[データ ストーリー]** オブジェクトの上部にある **[設定]** をクリックすると、ストーリーのパーソナライズやコンテキストの設定に役立つガイドが表示されます。詳細については、「Tableau データストーリーの構成設定」を参照してください。

> **注:** Tableau データストーリーに不一致が発生した場合 (たとえば、数値がストーリーとビジュアライゼーションとで異なる場合)、ビジュアライゼーションの設定方法が原因である可能性があります。別のシートで新しいビジュアライゼーションを作成し、[非表示のシートを使用する] 手法を活用して新しいデータストーリーを追加し、問題の原因を明らかにしてください。

自分の Tableau データストーリーに合った適切なストーリー タイプを選択する

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

ダッシュボードに Tableau データストーリーを追加するときは、データに適したタイプのストーリーを選択することが重要です。時間の経過に伴う傾向についてのストーリーが必要ですか? それとも、比較している 2 つの値についてのストーリーが必要ですか? このトピックでは、適切なストーリーを伝えられるようにするために、各ストーリー タイプの例を含め、さまざまなタイプのストーリーについて説明します。

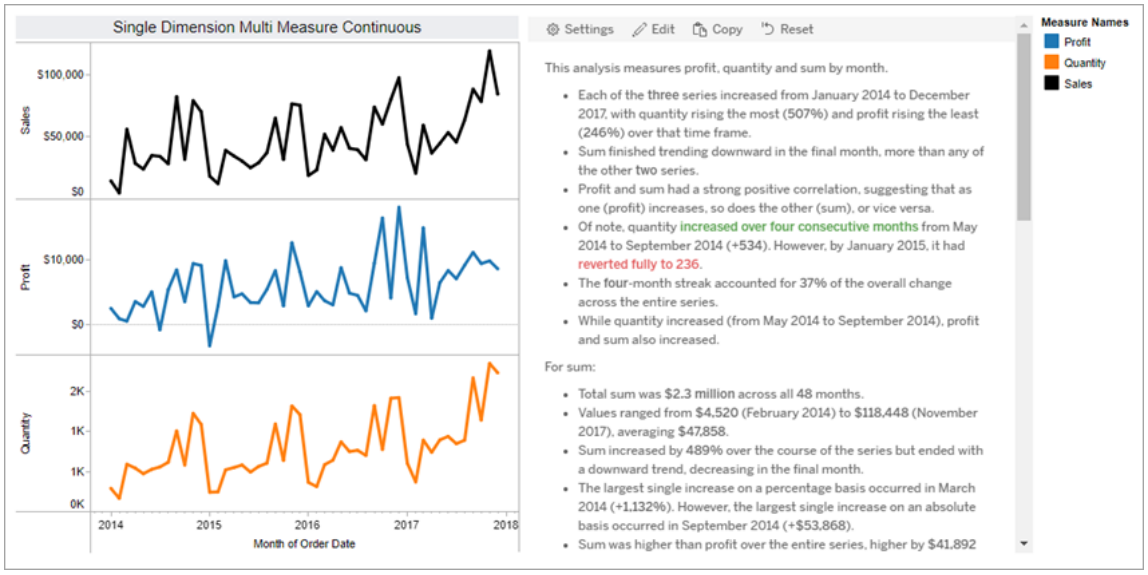
連続

連続ストーリーは、時間の経過に伴う傾向や進行状況を分析するのに最適です。

連続ストーリーを作成すると、ストーリーには、パフォーマンス、セグメント、ボラティリティ、傾向線のコンテンツが含まれます。また、ストーリーには、複数のディメンションを使用するストーリーの貢献度分析と相関性も含まれます。連続ストーリーを使用するには、ワークシートに次のものが必要です。

- 1 ～ 10 個のメジャーを持つ 1 つのディメンション
- 2 つのディメンションと最大 3 つのメジャー

次の例は、単一のディメンションと複数のメジャーを持つ折れ線グラフの連続ストーリーです。



不連続

不連続ストーリーは、値を比較し、各値のデータの分布を把握するのに最適です。不連続ストーリーを作成すると、ストーリーには、データ全体の分布とグループまたはクラスタに関するコンテンツが含まれます。また、ストーリーには、複数のディメンションを使用するワークシートの貢献度分析が含まれます。

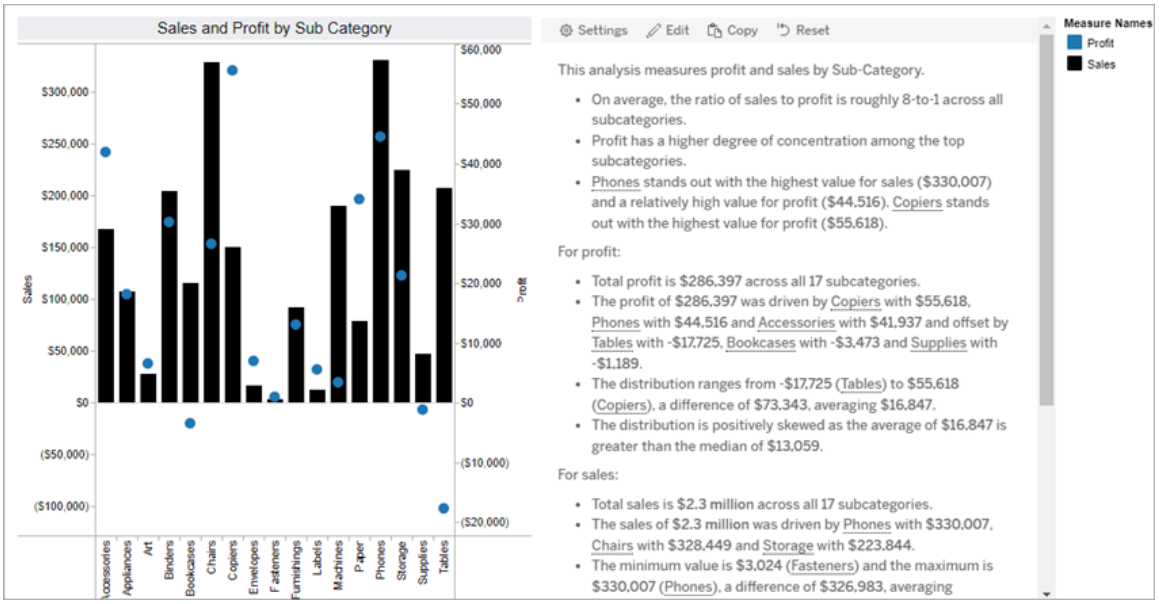
次の場合は、不連続ストーリーを使用することを検討してください。

- 売上レポートの重要業績評価指標 (KPI) のドライバーを把握する。
- データディスカバリ中に外れ値をすばやく特定して把握する。
- 監査を実行するときに、ビジュアルで簡単に監視できない傾向を特定する。
- 地理的分析のために、複雑な使用率のインサイトを即座に引き出す。
- 売上と利益の間などの重要な関係を特定してコールアウトする。

不連続ストーリーを使用するには、ワークシートに次のものが必要です。

- 1 ～ 10 個のメジャーを持つ 1 つのディメンション
- 2 つのディメンションと最大 3 つのメジャー

次の例は、単一のディメンションと2 つのメジャーを持つ棒グラフの不連続ストーリーです。

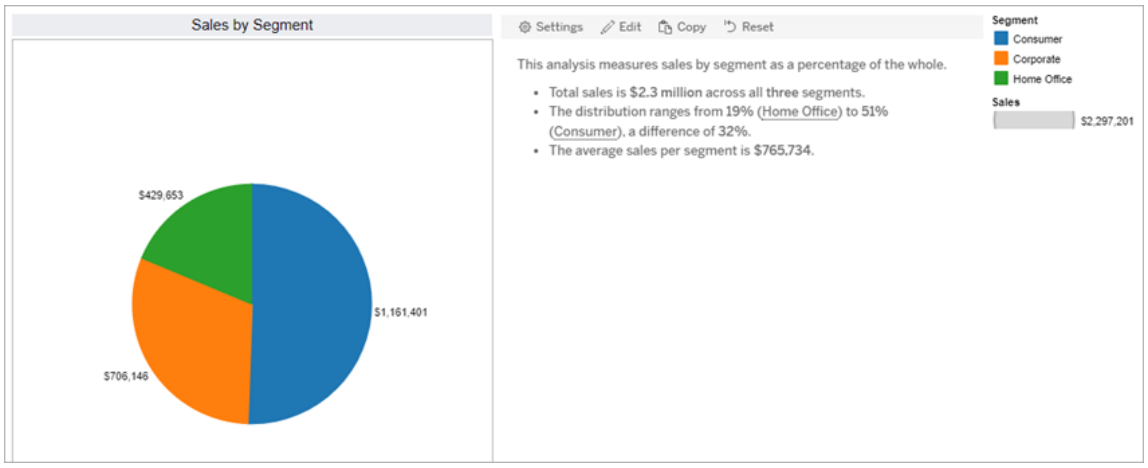


全体のパーセント

全体のパーセントストーリーには、円グラフが最適です。全体のパーセントストーリーを使用するには、ワークシートに次のものが必要です。

- 1 つのディメンション
- 1 つのメジャー

次の例は、単一のディメンションと単一のメジャーを持つ円グラフを使用した、全体のパーセントストーリーです。



散布図

散布図ストーリーは、2 つのメジャー間の関係を把握するのに最適です。散布図ストーリーを作成すると、ストーリーには、2 つのメジャー間の関係 (回帰) に関するコンテンツが含まれます。また、ストーリーには、データ内にグループ (クラスタ) が存在する場合、それらに関するコンテンツが含まれます。

次の場合は、散布図ストーリーを使用することを検討してください。

- 2 つのメジャー間の関係をコールアウトして、影響を特定する (回帰分析)。
- 定義されたしきい値を上回っている、または下回っている外れ値を特定して把握する。
- データがどのように分布しているかを分析する。

散布図ストーリーを使用するには、ワークシートに次のものが必要です。

- 1 つのディメンション
- 2 つまたは 3 つのメジャー

注: 散布図ストーリーを作成した場合、選択した最初のメジャーは独立変数として、2 番目のメジャーは従属変数として扱われます。

次の例は、単一のディメンションと2 つのメジャーを持つ散布図を使用した散布図ストーリーです。

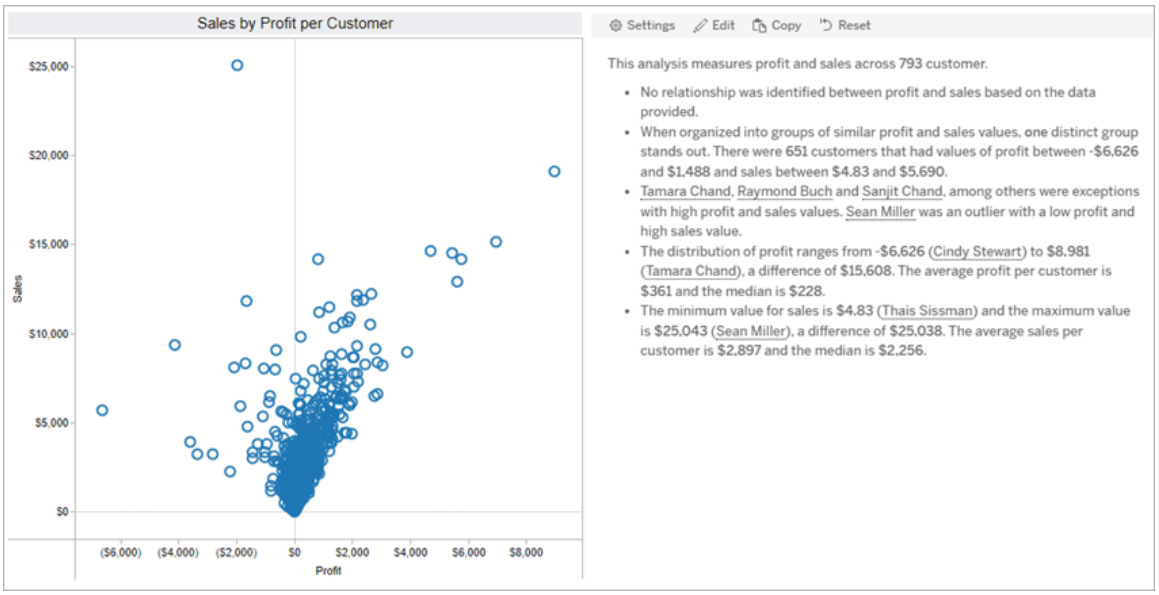


Tableau データストーリーの構成設定

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

ダッシュボードに Tableau データストーリーを追加すると、Tableau データストーリーをニーズに合わせて構成および編集できます。データに固有の言語を使用したり、記述する分析を指定したり、Tableau データストーリーの表示方法をカスタマイズしたりすることができます。

Tableau データストーリーの設定: 分析

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細に

ついては、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

Tableau データストーリー内で、どの分析について記述するのか、いつの分析について記述するかを選択できます。ストーリーのタイプと、ストーリーに含まれるディメンションとメジャーの数に応じて、様々なタイプの分析を利用できます。ただし、散布図のストーリータイプに対する分析は現在サポートされていません。詳細については、「自分の Tableau データストーリーに合った適切なストーリータイプを選択する」を参照してください。

ストーリー用に分析を設定する

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、**データストーリー** オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[分析]** タブをクリックします。
4. スイッチをクリックして、様々なタイプの分析を有効にします。
5. **[セグメント]** と **[傾向線]** については **[設定]** を展開し、分析を行うしきい値を設定します。
6. **[保存]** をクリックします。

様々なタイプの分析を理解する

相関性

相関姓を使用すると、2 つの系列間で真の統計的相関姓を特定できます。系列が 3 つ以上ある場合は、すべての組み合わせについて相関姓を分析します。たとえば、**相関姓**をオンにすると、2 つの製品が一緒に購入されることが多い時期を特定できます。

クラスタリング

クラスタリングを使用すると、性質が異なるデータポイントのグループ (クラスター) を単一の統計分析を使用して特定できます。たとえば、**クラスタリング**をオンにすると、ある製品が特定の地域で非常に人気が出る時期を特定できます。

分布

分布を使用すると、平均、中央値、歪みなどの非統計的観測値を使用してデータポイントを相互にランク付けできます。たとえば、**分布**をオンにすると、利益率が最も高い製品を特定できます。

セグメント

セグメントを使用すると、系列内のデータポイントに対する注目すべき変化を強調表示できます。まず、記述したいセグメント内の変化の最小割合を設定します。定義したしきい値を下回る変化については記述されません。たとえば、60% を超える変化をセグメントのしきい値として設定した場合、時系列上で 30% の減少が見られる谷間の期間があってもストーリーは記述しません。

しきい値を設定したら、**フォーマットを適用**するかどうかを選択し、フォーマットしたい変化の最小割合を設定します。

傾向線

傾向線を使用すると、線形最適線を計算し、定義された信頼度の範囲内にあるデータを特定できます。変動性の高いデータは、一貫性のあるデータに比べて信頼レベルが低く、その信頼レベルは傾向線が記述されるかどうかに影響します。傾向線は、ディメンションが 1 つでメジャーが 1 つのストーリーで使用することも、ドリルダウンで使用することもできます。ドリルダウンの詳細については、「Tableau データストーリー設定の構成: ナラティブ」を参照してください。

傾向線の信頼度の最小割合を設定します。しきい値を 95% に設定すると、90% の信頼度で傾向線を描画できる場合でも、ストーリーは傾向線について記述しません。しきい値を設定したら、**フォーマットを適用**するかどうかを選択します。次に、フォーマットする変化の最小割合を設定します。

傾向線に関するTableau データストーリーは、ある期間にわたる絶対的な変化を伝えます。傾向線について記述されたストーリーは、ストーリーに設定した冗長性のレベルによって異なります。ストーリーが高い冗長性を使用している場合、ストーリーは R 二乗値について記述します。これは、データが傾向線にどの程度適合しているかを定量化する統計的概念です。冗長性設定の詳細については、「Tableau データストーリー設定の構成: ナラティブ」を参照してください。

傾向線の設定では、ストーリーに予測を記述する将来の期間を選択することもできます。予測を使用する場合、ストーリーは傾向線の傾きと切片を使用して、将来の期間の予測値を計算します。予測の信頼度に応じて、傾向線に設定した信頼度のしきい値に上限と下限を追加します。予測は、ストーリーにデータポイントが少なくとも 30 あり、それらが線形である場合に使用できます。

変動性

変動性を使用すると、時間の経過に伴う標準偏差を分析できます。たとえば、データの平均的な範囲から外れた値についてストーリーに記述させたい場合、**変動性**を使用します。

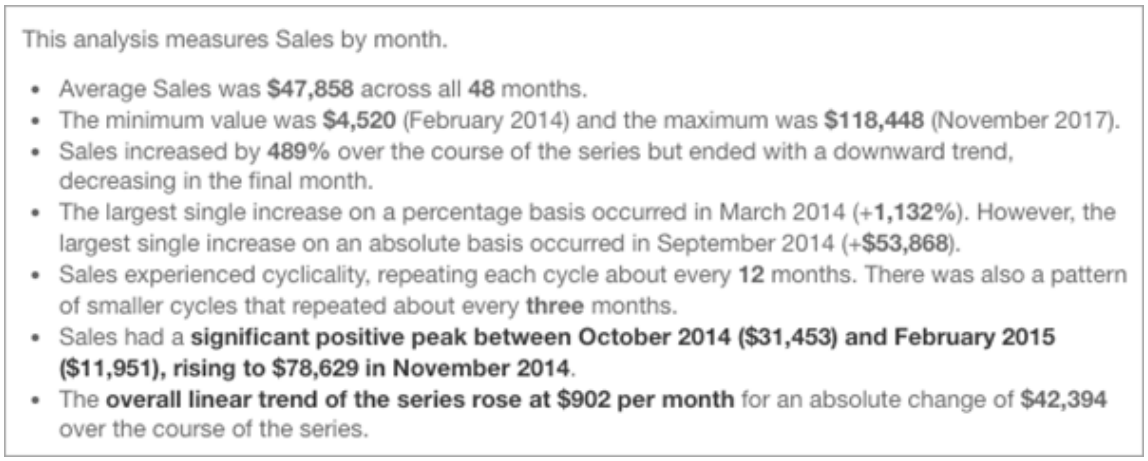
ストーリー生成のために分析を使用する方法

この時点で、様々なタイプのストーリー用の分析が、どのように行われるのか気になるところでしょう。各タイプのストーリーの例を見て、ストーリーの各文を分解してみましょう。

非連続的なストーリーのための分析を理解する

連続的なストーリーでは時間の経過に伴う傾向を測定するため、データストーリーは、パフォーマンス、進行、平均、合計、傾向、変動性、セグメント、予測について記述します。

次の連続的なストーリーの例は、月々の売上に関するものです。


This analysis measures Sales by month.

- Average Sales was **$47,858** across all **48** months.
- The minimum value was **$4,520** (February 2014) and the maximum was **$118,448** (November 2017).
- Sales increased by **489%** over the course of the series but ended with a downward trend, decreasing in the final month.
- The largest single increase on a percentage basis occurred in March 2014 (+**1,132%**). However, the largest single increase on an absolute basis occurred in September 2014 (+**$53,868**).
- Sales experienced cyclicality, repeating each cycle about every **12** months. There was also a pattern of smaller cycles that repeated about every **three** months.
- Sales had a **significant positive peak between October 2014 ($31,453) and February 2015 ($11,951), rising to $78,629 in November 2014**.
- The **overall linear trend of the series rose at $902 per month** for an absolute change of **$42,394** over the course of the series.

ストーリーの例	ストーリーの内容

• 全 **48** か月を通じて平均売上は **47,858 ドル**でした。 • 最小値は **4,520 ドル** (2014 年 2 月) で、最大値は **118,448 ドル** (2017 年 11 月) でした。	最初の 2 つの文は、平均関数と範囲関数を使用して、分析期間の平均値、最大値、最小値について記述しています。
• 系列全体で売上は **489%** 増加しましたが、最終月には減少に転じ、減少傾向で終了しました。	3 番目の文は、期間中のメジャーの全体的なパフォーマンスに関するものです。たとえば、特定の期間に売上が増加したか、減少したか、傾向が変わったか、といった文が考えられます。
• 単独で最も割合が増加したのは、2014 年 3 月 (**+1,132%**) でした。ただし、単独で最も絶対数が増加したのは、2014 年 9 月 (**+53,868 ドル**) でした。	4 番目の文は、進行分析を使用します。この文は、割合ベースと絶対数ベースの両方で、そのメジャーに基づく期間中の最大の増減について記述しています。
• **3 つ**の系列のうち、最も強い相関を示したのは企業とホームオフィスであり、中程度の正の相関があります。一般的に、一方 (企業) が増加すると他方 (ホームオフィス) も増加すること(その逆も同様) を示唆しています。	この文は、**相関関係**のインサイトです。このタイプの分析インサイトは、データ内の異なる系列間の顕著な相関関係について記述しています。
• 売上には周期性があり、約 **12** か月ごとに繰り返される周期がありました。約 **3** か月ごとに繰り返される小さな周期のパターンもありました。 • 売上は、**2014 年 10 月 (31,453 ドル) から 2015 年 2 月 (11,951 ドル) の間に大幅なプラスのピークを示し、2014 年 11 月には 78,629 ドルまで上昇しました。**	この文は、**セグメント**のインサイトです。このタイプの分析インサイトは、時間の経過とともに注目すべき増減について記述しています。
• **系列全体の線形傾向は月 902 ドルの上昇**であり、系列全体では **42,394 ドル**の絶対的な変化がありました。この	この文は、**傾向線**のインサイトです。このタイプのインサイトは、傾向線がデータにどの程度適合しているかを一定の信頼度で記述していま

傾向が次の 1 か月間続くと、売上は**約 69,958 ドルになると予測されます。**	す。傾向線を使用すると、過去の傾向に基づいて予測を立てることができます。

非連続的なストーリーのための分析を理解する

非連続的なストーリーでは値を比較してデータの分布を把握できるため、ストーリーは、分布、平均、合計、データのグルーピングやクラスターについて記述しています。

次の非連続的なストーリーの例は、製品ごとの売上に関するものです。

This analysis measures Sales by product.

- Total Sales is **$2.3 million** across all **17** products.
- The Sales of **$2.3 million** was driven by Phones with **$330,007**, Chairs with **$328,449** and Storage with **$223,844**.
- The distribution ranges from **$3,024** (Fasteners) to **$330,007** (Phones), a difference of **$326,983**, averaging **$135,129**.
- The distribution is positively skewed as the average of **$135,129** is greater than the median of **$114,880**.
- Sales is somewhat concentrated with **eight** of the **17** products (**47%**) representing **78%** of the total.
- The top **two** products represent over a quarter (**29%**) of overall Sales.
- Phones (**$330,007**) is more than **two** times bigger than the average across the **17** products.

ストーリーの例	ストーリーの内容
• 総売上は、全 **17** 製品で合計 **230 万ドル**です。	最初の文は、メジャーの合計値を計算しています。
• 売上 **230 万ドル**を牽引したのは、電話機 が **330,007 ドル**、椅子 が **328,449 ドル**、ストレージ が **223,844 ドル**でした。	2 番目の文は、ディメンションのドライバーについて記述しています。この例では、ディメンションのドライバーは、総売上に最も貢献した製品です。
• 平均値 **135,129 ドル**は中央値 **114,800 ドル**よりも大きいため、分布は正に偏っています。 • 売上は比較的集中しており、**17** 製品中 **8** 製品 **(47%)** が全体の **78%** を占めています。	3 番目と4 番目の文は、データの分布を分析しています。平均、中央値、データの集中度 (存在する場合)、およびデータがどのように歪んでいるかを分析します。これらのグループ化された変数が互いにどの程度バランスが取れているかを特定するのに役立ちます。

• 上位 **2 つ**の製品で、全体の売上の 4 分の 1 以上 (**29%**) 以上を占めています。	この文では、**クラスタリング**を使用して、グループ化できるメジャーについて記述しています。データの中に目立つ明確なグループがあるかどうかを特定するのに役立ちます。
• 電話機 (**330,007 ドル**) は、**17** 製品の平均の 2 倍以上です。	最後の文は、注目すべき外れ値について記述しています。

散布図ストーリー用に分析を理解する

散布図のタイプのストーリーは、2 つのメジャー間の関係を理解するために最もよく使用されます。そのため、散布図ストーリーには 2 ~ 3 のメジャーが必要です。散布図分析は、2 つのメジャー間の関係 (回帰) について記述し、データ内にグループ (クラスター) が存在する場合はそれについて記述しています。

次の散布図ストーリーの例は、ディメンション全体の利益と売上に関するものです。

This analysis measures profit, quantity and sales across 793 customer.

- As quantity increased and profit increased, sales increased based on the data provided. Specifically, when quantity increased by 1, sales increased $49.55, and when profit increased by $1.00, sales increased $1.20.
- Few customers deviated from this general relationship, indicating a good fit.
- When organized into groups of similar profit, quantity and sales values, one distinct group stands out. There were 651 customers that had values of profit between -$6,626 and $1,488, quantity between 2 and 122 and sales between $4.83 and $5,690.
- Tamara Chand, Raymond Buch and Sanjit Chand, among others were outliers with high profit and sales values. Sean Miller stood out with a low profit and high sales value.
- The minimum value for profit is -$6,626 (Cindy Stewart) and the maximum value is $8,981 (Tamara Chand), a difference of $15,608. The average profit per customer is $361 and the median is $228.
- The minimum value for quantity is 2 (Anthony O'Donnell) and the maximum value is 150 (Jonathan Doherty), a difference of 148. The average quantity per customer is 47.76 and the median is 44.
- The distribution of sales ranges from $4.83 (Thais Sissman) to $25,043 (Sean Miller), a difference of $25,038. The average sales per customer is $2,897 and the median is $2,256.

ストーリーの例	ストーリーの内容

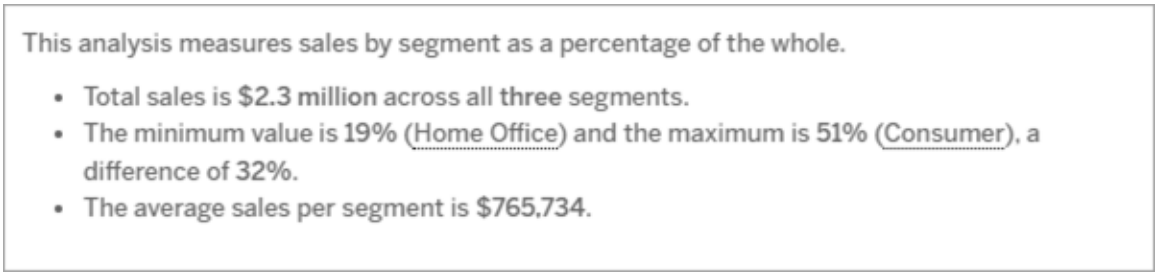
• 提供されたデータに基づくと、数量の増加と利益の増加に伴い、売上が増加しています。具体的には、数量が **1** 増えると売上が **49.55** ドル増え、利益を **1.00** ドル増やすには売上を **1.20** ドル増やしました。 • この一般的な関係から逸脱した顧客はほとんどなく、この関係はよく当てはまることを示しています。	最初の 2 つの文は、回帰分析に基づいています。回帰は、あるメジャーが別のメジャーにどのように影響するかを示します。最初の文で、ストーリーが利益と売上の関係を特定していることに注意してください。
• 利益、数量、売上が同程度のグループに整理すると、ある特徴的なグループが目立ちます。利益が **-6,626** ドルから **1,488** ドル、数量が **2** から **122**、売上が **4.83** ドルから **5,690** ドルの顧客が **651** 人いました。	3 番目の文は、クラスタリングから導出しています。クラスタリング分析は、データ内のすべての変数にまたがって、主要なグループやクラスターの特定を試みます。
• Tamara Chand、Raymond Buch、Sanjit Chandは、利益と売上の値が他に比べて高い外れ値でした。Sean Miller は、低い利益と高い売上で目立っていました。	4 番目の文は、平均を大幅に上回ったり下回ったりする外れ値について記述しています。
• 利益の最小値は **-6,626** ドル (Cindy Stewart)、最大値は **8,981** ドル (Tamara Chand) であり、その差は **15,608** ドルです。顧客あたりの利益の平均値は **361** ドルで、中央値は **228** ドル です。 • 数量の最小値は **2** (Anthony O'Donnell)、最大値は **150** (Jonathan Doherty) であり、その差は **148** です。顧客あたりの数量の平均値は **47.76** で、中央値は **44** です。 • 売上の分布は **4.83** ドル (Thais Sissman) から **25,043** ドル (Sean Miller) までで、その差は **25,038** ドルで	散布図ストーリーの残りの文は、範囲と平均の分析を使用してインサイトを記述しています。

す。顧客あたりの売上の平均値は **2,897** ドルで、中央値は **2,256** ドルです。	

ストーリー全体の割合のための分析を理解する

ストーリー タイプ全体で割合を示すと、ディメンションやメジャーが全体のどれだけを占めるかをよく把握できます。

次の全体の割合の例は、セグメント別の売上に関するものです。



ストーリーの例	ストーリーの内容
• 総合計 SUM(売上) は、3 つのエンティティすべてで 230万 です。	最初の文は、メジャーの合計値を計算しています。
• SUM(売上) 230万の内訳は、消費者が 120万、企業が 706,146、ホームオフィスが 429,653 でした。	2 番目の文は、ドライバーについて記述しています。この例では、ドライバーは総売上に最も貢献したセグメントです。
• 最小値は 429,653 (ホームオフィス)、最大値は 120万 (消費者) であり、その差は 731,748、平均は 765,734 です。	最後の文は、データの分布を分析しています。

Tableau データストーリー設定の構成: 特性

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

Tableau データストーリー内で、特性の設定を構成してデータにコンテキストを追加すると、より多くのインサイトが含まれたストーリーを作成できます。たとえば、ストーリー内の売上のコンテキストでは、数値が大きいほど適切であることを指定できます。ただし、顧客苦情のコンテキストでは、数値が大きいほど不適切になります。

ディメンションを使用して特性を測定する

ストーリーと数値の書式設定は、メジャーが何であるかに基づいて調整されます。既定では、ストーリーはすべてのメジャー バリューを数値として記述し、追加の計算を実行したり、特別なレンダリング ルールを適用したりしません。

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[Characteristics (特性)]** タブをクリックします。
4. 数値タイプ、小数点以下の桁数、負の値など、書式設定を構成します。
5. **[保存]** をクリックします。

メジャーの特性の詳細

書式設定

値が **[Percentages (パーセンテージ)]** として書式設定されている場合、ストーリーは、数値の場合のように変化率ではなく、パーセントポイントの差について記述します。**[Characteristics (特性)]** タブで数値を **[Percentages (パーセンテージ)]** として書式設定すると、**データストーリー**は数値に 100 を掛けて、ストーリーに表示されるパーセンテージを作成します。

値が **[Currency (通貨)]** として書式設定されている場合は、希望する通貨を指定できます。大きな値 (100 万を超える数値) をどのように書式設定するかを指定することもできます。たとえば、1,300,000.00 ドルではなく、130 万ドルと書式設定できます。

[Numbers (数値)] と**[Currency (通貨)]** の両方で、大きな値とマイナスの値をどのように記述するかを指定できます。マイナスの値を括弧で囲んで記述する場合、マイナスの値が括弧で囲まれたフレーズ内にあると、入れ子になった括弧がストーリーに表示されることがあります。

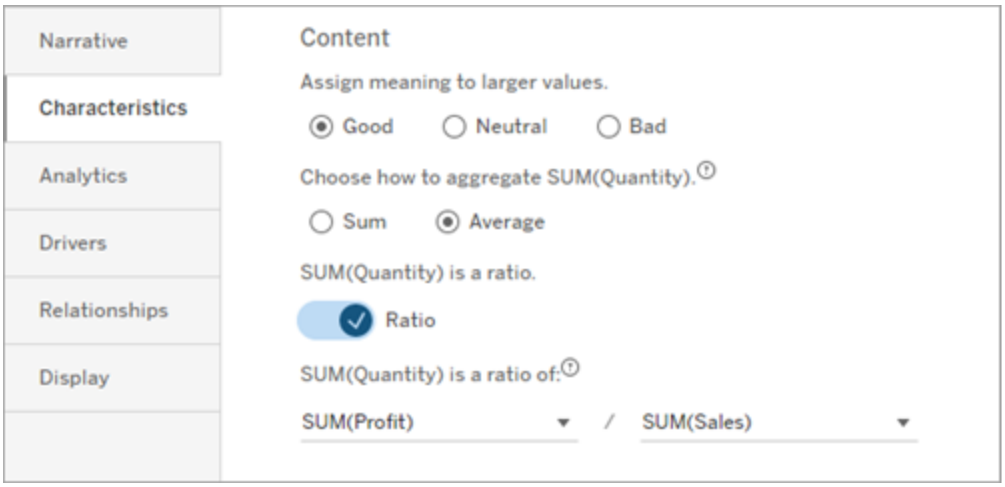
[Number (数値)] の書式設定を選択した場合、ストーリーで (数値を使用するのではなく) 10 以下の数値をスペルアウトするかどうかを指定することもできます。

小数点以下の桁数の場合、**[Dynamic (動的)]** が既定のオプションです。これは、数値の大きさに応じて、ストーリーでは、小数点以下のさまざまな桁数に丸めることを意味します。パーセント値が 10 未満の場合、数値には小数点以下 2 桁が含まれます。パーセント値が 10 より大きい場合、数値は最も近い整数に丸められます。使用する小数点以下の桁数を指定することもできます。これは、ストーリー全体で一貫して使用されます。

コンテンツ

大きな値に意味を割り当てることができます。たとえば、売上の値は大きいほど適切ですが、損失の値は大きいほど不適切です。

さらに、合計または平均で値を集計する方法を選択できます。Viz で使用しているのと同じ集計方法を選択することをお勧めします。比率メジャーの場合は、**[Average (平均)]** を選択し、その比率メジャーのコンポーネントメジャーを選択して比率を定義します。比率のコンポーネントであるメジャーは、合計可能でなければなりません。



累積メジャー(連続ストーリーのみで使用可能)の場合は、**[Sum (合計)]** を選択してから、メジャーがすでに累積であることを指定します。連続ストーリーは、シリーズ全体のメジャーの合計について記述します。

並べ替え

ディメンション値を並べ替えるには、上矢印/下矢印アイコンをクリックして、ディメンション内の最も古いまたは最も新しい時間値に基づいて並べ替えます。

> **注:** ディメンション値の並べ替えは、連続ストーリーのみで使用できます。

Tableau データストーリー設定の構成: 表示

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

フォントの色やサイズなど、Tableau データストーリー内のテキストの表示方法を構成できます。ストーリーで箇条書きまたは段落を使用するかどうかを選択することもできます。

ストーリーの表示を構成する

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[表示]** タブをクリックします。
4. ストーリー構造を**箇条書き**にするか**段落**にするかを選択します。
5. **フォント サイズ**を選択します。
6. 良い変化と悪い変化を表す色を選びます (連続したストーリーで利用可能)。
7. **動的順序付け**を使用するかどうかを選択します。
8. **圧縮ビュー**を使用するかどうかを選択します。
9. **[保存]** をクリックします。

ストーリー表示設定を使用するタイミングを理解する

色を使用するには、ストーリーが連続している必要があります。色を使用する場合、パレットから色を選択して、良い変化と悪い変化を表すことができます。ストーリーが良い変化と悪い変化を判断できるようにするには、**[特性]** タブで大きな値に意味を割り当てる必要があります。詳細については、「Tableau データストーリー設定の構成: 特性」を参照してください。表示を構成した後、傾向線またはセグメント分析のしきい値に基づいて、スタイルと色がストーリーに適用されます。

[Dynamic Ordering (動的順序付け)] をオンにすると、ストーリー内のメジャーのインサイトが、平均値の最高値から最低値の順に動的に順序付けられます。維持したい順序がすでにある場合は、**[Dynamic Ordering (動的順序付け)]** をオフにします。

[Condensed View (圧縮ビュー)] をオンにすると、ストーリーから追加のスペースが削除されます。これは、ダッシュボードに余計なスペースがあまりない場合、またはダッシュボードに複数のデータストーリー オブジェクトがある場合に役立ちます。

Tableau データストーリー設定の構成: ドライバー

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

前月比の売上が大幅に増加したとします。何が売り上げを伸ばしたのでしょうか? そして、それらの売上げの増加を損なった (相殺した) のは何でしょうか? データストーリーにドライバーを設定すると、これらの質問に答えることができます。

データストーリーでは、ドライバーは合計値に貢献します。オフセッターは合計値を損ないます。不連続ストーリーと連続ストーリーで、ドライバーとオフセッターに関するインサイトを引き出します。これらのインサイトにより、データで何が起こっていて、その原因は何かについて正確に把握できるようになります。

ディメンション ドライバーを設定する

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[Drivers (ドライバー)]** タブをクリックします。
4. **[Dimension Drivers (ディメンション ドライバー)]** セクションで、分析に最大の影響を与えるドライバーのタイプを選択します。
   [Count (カウント)] には、要因とオフセッターの最大数を設定します。
   [Individual % (個別 %)] には、個別の要因とオフセッターについて記述する際のしきい値を設定します。
   [Cumulative % (累積 %)] には、集合的な価値に基づいて要因とオフセッターについて記述する際のしきい値を設定します。
5. **[保存]** をクリックします。

ディメンション ドライバーのタイプを理解する

- **[Count (カウント)]** は、ストーリーで呼び出されるエンティティ(要因とオフセッター) の数を指定します。たとえば、**[Count (カウント)]** を使用して、データの上位 3 つの要因とオフセッターを確認できます。
- **[Individual % (個別 %)]** はしきい値を設定し、そのしきい値よりも高い値がストーリーに含まれます。たとえば、**[Individual % (個別 %)]** を使用すると、合計値の 5% 以上を占めるエンティティのみについて記述することを指定できます。
- **[Cumulative % (累積 %)]** は、含まれるエンティティが集合的に占める、合計値のパーセンテージしきい値を設定します。たとえば、**[Cumulative % (累積 %)]** を使用すると、その合計値の少なくとも 90% に貢献したエンティティについて記述することを指定できます。この例では、エンティティの累積値が合計値の 90% を占めるまでの大きさの順にエンティティについて記述しています。

セカンダリ要因を使用する

セカンダリ要因を使用するには、時間ではない 2 番目のディメンションが必要です。セカンダリ要因を使用する場合、記述される各ドライバーには、セカンダリ コントリビューターの詳細とドライバーも含まれます。たとえば、店舗の売上を分析している場合、セカンダリ要因は部門内のクラスになります。セカンダリ要因は、より深い分析を可能にします。また、セカンダリ要因に多くの情報を含めて、ストーリーの 1 つの文に収めることができます。

メトリクス ドライバーを設定する

他のサブカテゴリー メジャーで構成されるメジャーの場合、ドライバー分析により、各メジャーがトップレベルの値に与えた影響を説明できます。たとえば、材料費と運用コストが総コストの要因になります。

メトリクス ドライバーを使用するには、メトリクス分析に複数のメジャーを含める必要があります。次に、各メジャー間の関係を指定できます。

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[Drivers (ドライバー)]** タブをクリックします。
4. **[Metric Drivers (メトリクス ドライバー)]** セクションで、最初に別のメジャーのサブカテゴリーであるメジャーを選択します。
5. 次に、プライマリ カテゴリーであるメジャーを選択します。
6. **[保存]** をクリックします。

ヒント: 詳細度設定はドライバーにも適用されます。ストーリーの詳細度設定を変更することで、インサイトの記述方法を調整できます。高い詳細度を使用すると、さらに多くの情報が括弧内に表示されます。低い詳細度を使用すると、ドライバーについて簡潔に記述されたインサイトが得られます。詳細については、「Tableau データストーリー設定の構成: ナラティブ」を参照してください。

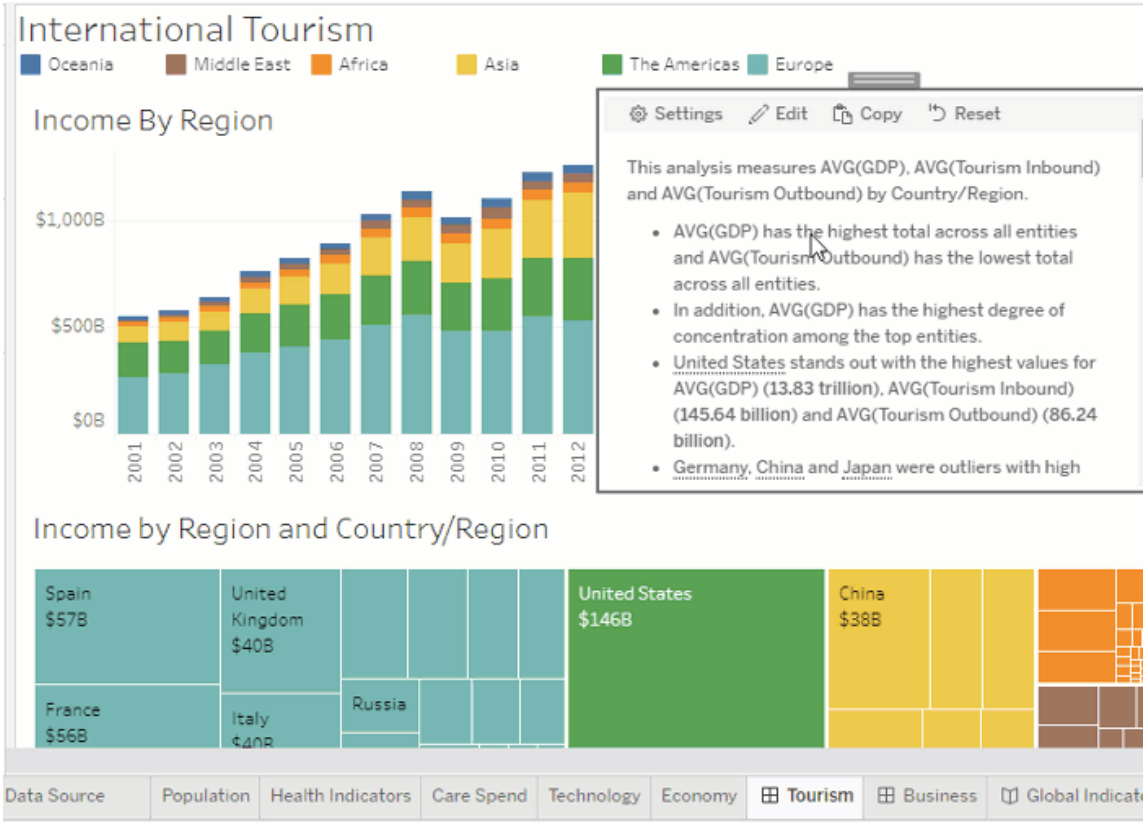


Tableau データストーリー設定の構成: ナラティブ

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

Tableau データストーリーのナラティブ設定をカスタマイズできます。具体的には、ストーリーの詳細度と分析の詳細レベルを指定したり、データに固有の用語を追加したりできます。これらの設定は、適切な言語と詳細レベルで記述されたストーリーをダッシュボードに作成するのに役立ちます。

詳細度を設定する

詳細度は、ストーリーの長さと記述される分析の詳細を指定します。高い詳細度を選択した場合、ストーリーは、より分析的な長いインサイトを提供します。低い詳細度を選択した場合、ストーリーはより簡潔になり、詳細が少なくなります。

閲覧者に詳細度の変更を許可すると、パブリッシュされたダッシュボードの閲覧者は、データストーリー オブジェクトの詳細度レベルを変更できます。これは、ストーリーにさまざまな詳細レベルを求めている閲覧者がいる幅広いユーザーがダッシュボードを使用する場合に役立ちます。

ドリルダウンを設定する

ドリルダウンには 2 つのディメンションが含まれます。ドリルダウンは、ダッシュボードの各ディメンションに関連付けられた数値の説明を提供します。

製品カテゴリーごとの月間売上高を示すダッシュボードがあるとします。ストーリーは、**時間**と**カテゴリー**の両方のディメンションについて記述するように構成されています。この場合、このストーリーには、**時間**ディメンション内およびその全体のパフォーマンスを説明する、各カテゴリーに関するインサイトが含まれます。

ドリルダウンの最大数を設定することで、ストーリーに含まれるインサイトの数を制御できます。ドリルダウン インサイトも、メジャーの特性に割り当てられた意味に基づいてランク付けされます。ドリルダウンの数を減らすと、パフォーマンスが最も低いメジャー (悪い意味が割り当てられたメジャー) が削除されます。

ディメンションの用語を追加する

用語を追加することで、ストーリーで各メジャーとディメンションにラベルを付けて参照する方法を定義できます。

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[ナラティブ]** タブをクリックします。
4. ディメンションを展開して、単数と複数の両方の形式でどのように記述されるかを確認します。

5. **[Add Term (用語の追加)]** をクリックして、ディメンションの説明に使用するストーリーの別のバリエーションを追加します。
6. **[保存]** をクリックします。

ストーリーでは、ディメンションについて記述するときに追加した用語を (ランダムに) 使用します。

メジャー ラベルを管理する

ディメンションと同様に、ストーリーのメジャーに使用されるラベルを管理できます。

1. **[ナラティブ]** タブで、メジャーを展開してラベルを確認します。
2. メジャーに使用する新しいラベルを入力します。
3. **[保存]** をクリックします。

Tableau データストーリー設定の構成: 関係

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

[関係] の設定を使用すると、複数のメジャーがある Tableau データストーリーのメジャー間の関係を割り当てることができます。関係を構成するには、ストーリーに 1 つのディメンションと複数のメジャーが必要です。詳細については、「Tableau データストーリーでカスタム メジャー関係を作成する」を参照してください。

関係には次の 2 つのタイプがあります。

- Actual vs. Benchmark (実際のデータとベンチマーク)
- Current/Most Recent vs Previous Period (現在/最新と前の期間)

割り当てレポートを実行する場合など、パフォーマンスベンチマークを上回っているか下回っているかを知りたい場合は、**[Actual vs. Benchmark (実際のデータとベンチマーク)]** を使用します。また、このタイプの関係は、ベンチマークを大幅に上回ったり下回ったりしているため追加の分析が必

要なデータポイントを特定するのに役立ちます。**[Actual vs. Benchmark (実際のデータとベンチマーク)]** を使用するには、比較するメジャーの値のタイプが同じである必要があります。

主要業績評価指標 (KPI) が時間の経過とともに増加、減少、または一貫性を維持しているかどうかを確認する場合は、**[Current/Most Recent vs. Previous Period (現在/最新と前の期間)]** を使用します。**[Current/Most Recent vs. Previous Period (現在/最新と前の期間)]** を使用するには、不連続ストーリータイプを使用する必要があります。詳細については、「自分の Tableau データストーリーに合った適切なストーリータイプを選択する」を参照してください。

[Actual vs. Benchmark (実際のデータとベンチマーク)] と**[Current/Most Recent vs. Previous Period (現在/最新と前の期間)]** を同時に使用することもできます。構成された関係の一部ではない Viz のメジャーは、別の段落に記述されます。

連続ストーリーまたは不連続ストーリーの [Actual vs. Benchmark (実際のデータとベンチマーク)] 関係を作成する

1 つのメジャーが他のメジャーのベンチマークである場合は、**[Actual vs. Benchmark (実際のデータとベンチマーク)]** 関係を使用します。たとえば、実際の売上高と売上目標を比較して、目標を上回ったか下回ったかについてのインサイトをストーリーで記述できます。このタイプの関係を使用した場合、ストーリーでは不要なコンテンツが削除され、最も重要なこと、つまり、メトリクスとその関連するベンチマークとの比較に焦点が当てられます。

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリーオブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログボックスで **[関係]** タブをクリックします。
4. **[Actual vs. Benchmark (実際のデータとベンチマーク)]** のボックスをオンにします。
5. 最初に、ベンチマークとなるメジャーを選択します。
6. 次に、ベンチマークと比較するメジャーを選択します。
7. **[保存]** をクリックします。

[Current/Most Recent vs. Previous Period (現在/最新と前の期間)] 関係を作成する

[Current/Most Recent vs. Previous Period (現在/最新と前の期間)] 関係を使用して、一定期間にわたる 2 つのメジャーのパフォーマンスを比較します。たとえば、2 つの製品を比較して、昨年最も多くの収益を上げた製品を確認できます。

1. ダッシュボードに Tableau データストーリーを追加する。
2. ダッシュボードで、データストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
3. [データストーリー] ダイアログ ボックスで **[関係]** タブをクリックします。
4. **[Current/Most Recent vs. Previous Period (現在/最新と前の期間)]** のボックスをオンにします。
5. 最初に、前の期間のメジャーを選択します。
6. 次に、現在の期間のメジャーを選択します。
7. 測定している期間のラベルを入力します (例 : 年)。
8. 測定する期間の数を選択します。
9. **[保存]** をクリックします。

Tableau データストーリーのカスタマイズ

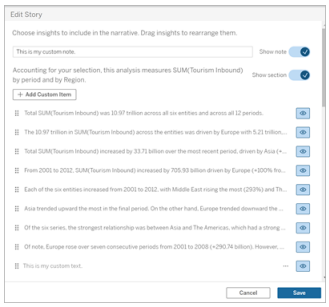
Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

閲覧者に合わせてカスタマイズした言葉を使用して、ビジネスに特化したインサイトで Tableau データストーリーを補足できます。閲覧者にとって重要なデータストーリーから分析とデータを特定し、独自の言語を使用して影響力のあるストーリーを作成します。全体的な Tableau データストーリーと同様に、カスタム コンテンツで使用されるデータと変数は動的であり、ダッシュボードに合わせて調整されます。

独自のインサイトを追加する

1. **データストーリー**で **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. 記述したいセクションを見つけて、**[カスタム アイテムの追加]** をクリックします。



3. 表示されるフィールドにカスタム テキストを入力します。
4. **[保存]** をクリックします。

ヘッダーとフッターを追加する

データ ストーリーの上部と下部にカスタム テキストを挿入できます。ヘッダーとフッターを使用すると、ストーリーに独自の定性分析を追加したり、データの傾向について追加の説明を含めたり、法的およびプライバシーに関する免責事項を追加したりできます。

1. **データ ストーリー**で **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. **[メモの表示]** スイッチをクリックします。
3. カスタム メモを入力します。

4. **[保存]** をクリックします。

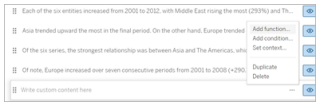
これで、メモが**データ ストーリー**の上部または下部に表示されます。

関数を追加する

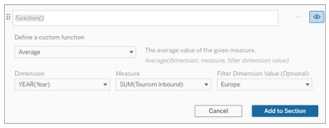
データ ストーリーで関数を使用すると、ストーリーをカスタマイズし、あなたやあなたのビジネスにとって最も重要なインサイトを見つけることができます。

たとえば、ある期間の観光収入の平均合計を知りたい場合は、関数として [平均] を選択し、メジャーとして SUM(インバウンド観光) を指定します。これにより、インバウンド観光の平均合計が返されます。

1. **データ ストーリー**で **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. ボックスの右側にあるメニューをクリックして、**[関数の追加]** を選択します。



3. [データストーリー関数] を選択し、必須フィールドに入力します。



4. **[セクションに追加]** をクリックします。
5. **[保存]** をクリックします。

これで、カスタム コンテンツがデータ ストーリーに表示されるようになりました。

条件を追加する

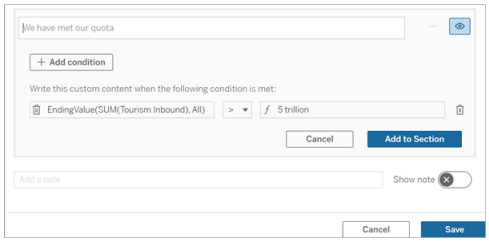
作成するカスタム文ごとに、その文が**データ ストーリー**に表示されるかどうかを決定する条件を追加できます。条件が満たされると、カスタム文がストーリーに表示されます。条件が満たされない場合、カスタム文は表示されません。

各カスタム文には複数の条件を適用することができ、[条件の追加] ダイアログ ボックス内で **[任意]** または **[すべて]** のボタンを使用すると条件を組み合わせることができます。

条件文は数値比較で最もよく使用されますが、この関数は、等しい (=) または等しくない (!=) 記号を使用した文字列照合もサポートしています。

1. データ ストーリーで **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. カスタム文を入力します。

3. ボックスの右側にあるメニューをクリックして、**[条件の追加]**を選択します。
4. 条件付きロジックを通知するために使用するカスタム関数を定義します。この例では、「We have met our quota (ノルマ達成)」という文は、インバウンド観光の合計が 5 兆を超えると表示されます。



5. **[セクションに追加]** をクリックします。
6. **[保存]** をクリックします。

これで、条件が満たされた場合にのみ、カスタム文がデータストーリーに表示されるようになりました。

カスタム コンテンツを複製する

データストーリーに追加されたカスタム コンテンツは簡単に複製できるため、様々なバリエーションの文を簡単に作成できます。しきい値を適用したり、言語のバリエーションを組み込んだり、様々な論理的なバリエーションを作成したりする場合は、完全に構築されたカスタム文をコピーすることをお勧めします。

1. データストーリーで **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. 必要に応じて、関数と条件を完全に備えたカスタム文を作成します。
3. 完全な文のボックスで、ボックスの右側にあるメニューをクリックし、**[複製]** を選択します。

4. 複製した文をクリックして、必要に応じて更新し、**[セクションに追加]** をクリックします。
5. **[保存]** をクリックします。

文をコピーすると、すべての関数と条件文もコピーされます。コピーされた箇条書きは、同じセクションの元の箇条書きのすぐ下に表示されます。

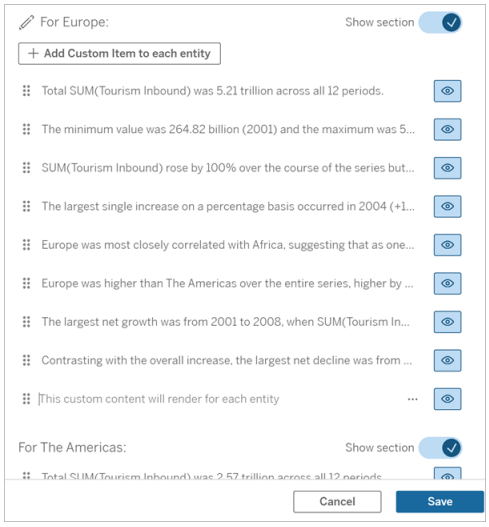
注: コピーされた箇条書きは、元の箇条書きが含まれているセクションにのみ追加できます。

ドリルダウン セクションにカスタム コンテンツを追加する

2 次元のストーリーの場合、最初のセクションの後の各セクションはドリルダウン セクションと呼ばれます。ドリルダウン セクションでは、個々のプライマリ ディメンション エンティティ(次の例では南北アメリカ) に焦点を当てています。

ドリルダウン セクションのコンテンツ構造は同じであるため、最初のセクション (ヨーロッパ) で追加されたカスタム コンテンツは、追加の各セクション (南北アメリカ) に適用されます。コンテンツを作成または編集できるのは、最初のドリルダウン セクションのみです。

1. データ ストーリーで **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. 最初のドリルダウン セクションで、各エンティティに対して **[カスタム アイテムの追加]** をクリックします。
3. カスタム コンテンツを入力します。
4. **[保存]** をクリックします。



ドリルダウン セクションのカスタム コンテンツには、現在のカテゴリ値 (動的) というコンテキスト変数がすでに含まれています。これにより、現在のカテゴリ値 (動的) というディメンション値オプションが作成されます。これは、ドリルダウン セクションが含まれるセクションを常に表します。

ヒント: 単一の特定のドリルダウン セクションにのみ表示されるコンテンツを作成するには、条件付きロジックを使用して、適切な場所にのみ書き込むようにすることができます。

Tableau データストーリーのカスタマイズ: コンテキスト変数

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータ エクスペリエンスを刷新」を参照してください。

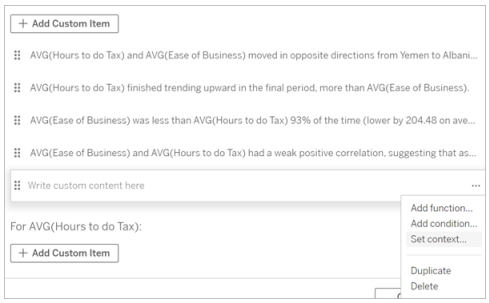
コンテキスト変数は、他の関数から参照できる関数です。つまり、コンテキスト変数を使用すると、関数を他の関数内にネストできます。

コンテキスト変数を定義すると、新しい関数を Tableau データストーリーに追加するときに使用できる関数のように表示されます。

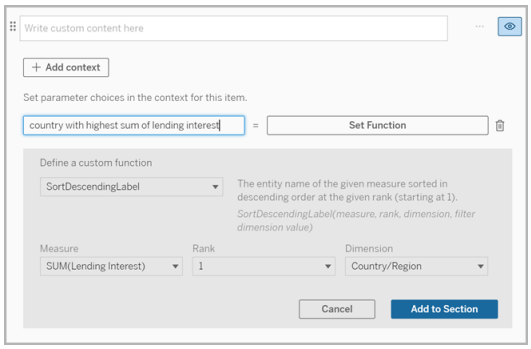
> **注:** 各カスタム文に複数のコンテキスト変数を設定できますが、カスタム コンテンツごとに各コンテキスト変数を個別に定義する必要があります。

コンテキスト変数の設定

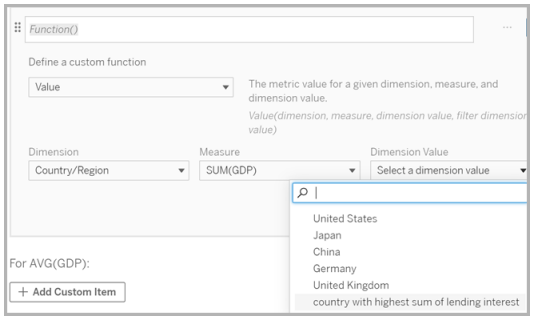
1. **データ ストーリー**で **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. **[カスタム アイテムの追加]** をクリックします。
3. カスタム コンテンツ ボックスの右側にあるメニューをクリックし、**[コンテキストの設定]** を選択します。



4. **[コンテキストの追加]** をクリックします。
5. コンテキスト変数に名前を付けて、**[関数の設定]** をクリックします。
6. カスタム関数を定義し、ディメンションを選択します。



7. **[セクションに追加]** をクリックします。
8. コンテキスト変数を設定した文にクリックして戻ります。
9. 手順に従って [関数を追加する] します。



これで、関数を追加するときに、コンテキスト変数が **[ディメンション値]** ドロップダウン リストのオプションとして一覧表示されるようになります。

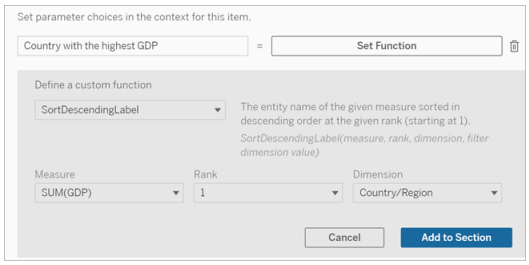
コンテキスト変数を使用するタイミング: 複数のメジャーの参照

データ ストーリーの 1 つの分析文で、複数のメジャーを参照したいと思います。コンテキスト変数がなければ、1 つの文に記述できるメジャーは一度に 1 つだけでしょう。ところが、コンテキスト変数を使用すると、1 つの文で複数のメジャーを参照できます。

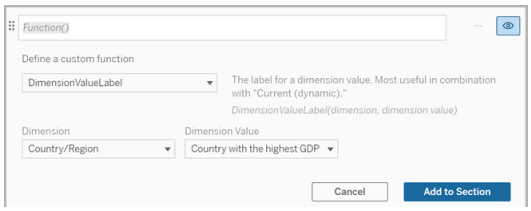
コンテキスト変数を使用して複数のメジャーを参照するには、**データ ストーリー**に次のものが必要です。

- 1 つのディメンション
- 2 つ以上のメジャー

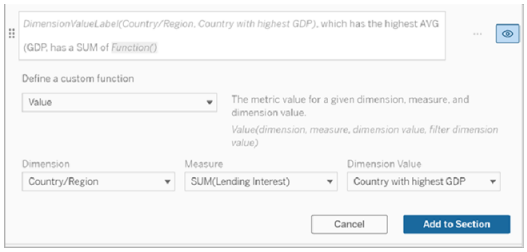
1. **データ ストーリー**で **[編集]** をクリックし、[編集] ダイアログ ボックスを開きます。
2. **[カスタム アイテムの追加]** をクリックします。
3. カスタム コンテンツ ボックスの右側にあるメニューをクリックし、**[コンテキストの設定]** を選択します。
4. **[コンテキストの追加]** をクリックします。
5. コンテキスト変数に名前を付けて、**[関数の設定]** をクリックします。
6. カスタム関数を定義し、ディメンションを選択します。
7. **[セクションに追加]** をクリックします。



8. コンテキスト変数を設定した文にクリックして戻ります。
9. 1 番目の関数を追加し、必要なフィールドに入力します。この例では、**[DimensionValueLabel]** を選択し、次に **[ディメンション]** から [国/地域] を選択し、次に **[ディメンション値]** から GDP が最も高い国 (コンテキスト変数) を選択しました。



10. [セクションに追加] をクリックします。
11. 2 番目の関数を追加し、必要なフィールドに入力します。この例では、[値] を選択し、次に [ディメンション] から [国/地域] を選択し、[メジャー] から SUM(貸出金利) を選択し、[ディメンション値] から GDP が最も高い国 (コンテキスト変数) を選択しました。



12. [セクションに追加] をクリックします。
13. [保存] をクリックします。

データストーリーは、私たちが関心を持っている国 (GDP が最も高い国) の副次メジャー (貸付金利) についての洞察を与える文を書いています。

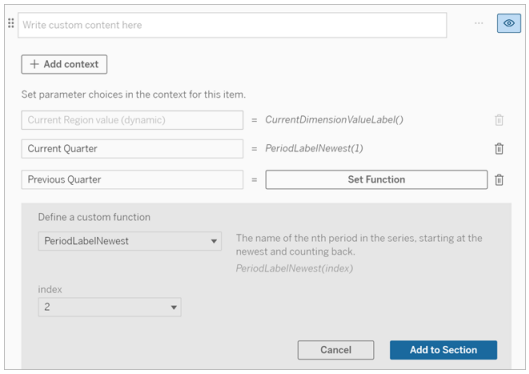
コンテキスト変数を使用するタイミング: 期間ごとの分析

コンテキスト変数は、**データ ストーリー**で 2 つの異なる期間のパフォーマンスを分析する場合に役立ちます。カスタム文を作成すると、ドリルダウン セクションに表示されるメジャーについて記述し、メジャーを年ごとや月ごとなどの様々な期間で比較することができます。

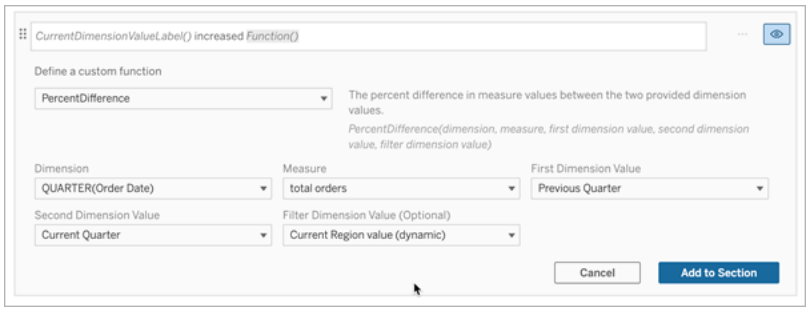
期間ごとの分析を設定するには、データストーリーに次のものが必要です。

- 2 つのディメンション: 1 つの期間ディメンション (プライマリ) と 1 つの非期間ディメンション (セカンダリ)
- 1 つから 3 つのメジャー

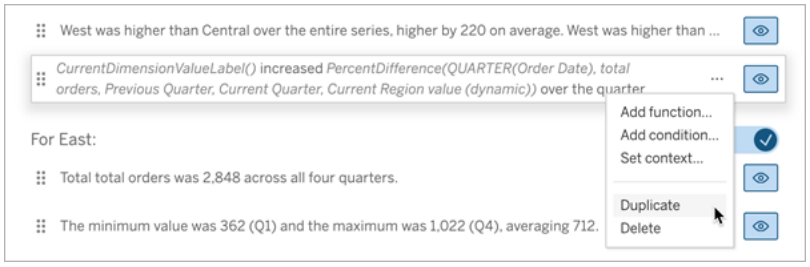
1. **データ ストーリー**を作成します。
2. [フィールド] ダイアログ ボックスで、期間ディメンションが最初に順序付けられていることを確認し、**[次へ]** をクリックします。
3. [ストーリー] ダイアログ ボックスで **[連続]** を選択し、**[完了]** をクリックします。
4. [編集] ダイアログ ボックスを開き、最初のドリルダウン領域で **[カスタム アイテムの追加]** を選択します。
5. 期間を表すコンテキスト変数を 2 つ作成します。たとえば、「現在の四半期」と「前の四半期」を作成します。



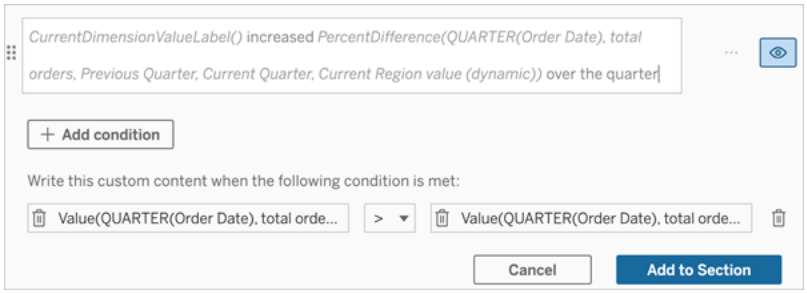
6. カスタムの言葉と関数を追加して、「*増加した*」文を作成します。この文は、「[CurrentDimensionValueLabel] は四半期で [X%] 増加しました」というコンテンツになります。
7. カスタム テキスト ボックスに、関数に続けて「増加しました」という単語を入力します。
8. 期間中のメジャーの変化率を返す関数を追加します。



9. [セクションに追加] をクリックします。
10. カスタム コンテンツを複製するし、「増加」という単語を「減少」に置き換えて、減少バージョンを作成します。関数は同じままです。



11. データに応じて 1 つだけが書き込まれるように、各文に条件を追加するします。



12. 「増加した」文について、次の例に従って条件を設定します。

    左の引数 = 値の関数

    ディメンション = Quarter(オーダー日)。期間ディメンションを選択します

    メジャー = SUM(オーダー合計)。計算に使用したメジャーを選択します

ディメンション値 = 現在の四半期。コンテキスト変数の 1 つ

フィルターディメンション値 = 現在の地域の値 (動的)。これはプリセットされたコンテキスト変数です

中央の引数 = > (より大きい)

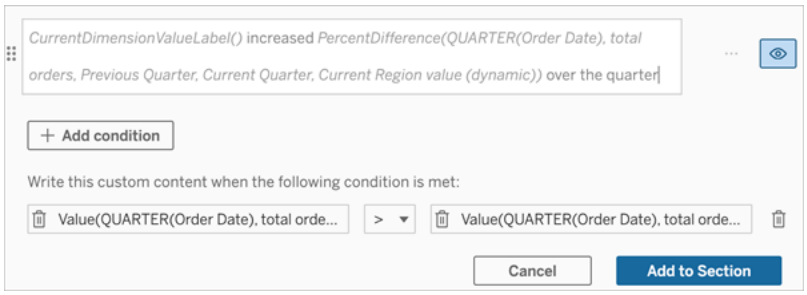
右の引数 = 値の関数

ディメンション = Quarter(オーダー日)。期間ディメンションを選択します

メジャー = Total(オーダー合計)。計算に使用したメジャーを選択します

ディメンション値 = 前の四半期。コンテキスト変数の 1 つ

フィルターディメンション値 = 現在の地域の値 (動的)。これはプリセットされたコンテキスト変数です



13. 「*減少した*」文について、同じ条件を設定しますが、**>** 記号 (より大きい) を **<** 記号 (より小さい) に置き換えます。右の引数と左の引数は同じままです。
14. **[保存]**をクリックすると、**データストーリー**に 2 つの期間の分析から得られたインサイトを含む文が記述されます。

Tableau データストーリーのカスタマイズ: 関数

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

関数を使用して Tableau データストーリーをカスタマイズする場合は、使用できる関数、各関数の機能、各関数に必要なディメンションとメジャーを知っていると役立ちます。

データストーリーに関数を追加するする方法を説明します。

文字をクリックして、その字で始まる関数を参照します。その字で始まる関数がない場合、アルファベットの次の字で始まる関数が表示されます。Ctrl+F (Mac では Command-F) を押して検索ボックスを開き、ページ内で特定の関数を検索できます。

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Average

説明: 指定されたメジャーの平均値。

構文: Average(ディメンション, メジャー, filterDimensionValue)

Count

説明: 指定されたディメンション内のディメンション値の個数。

構文: Count(ディメンション)

Difference

説明: 指定された2つのディメンション値のメジャー値間の差異。

構文: Difference(ディメンション, メジャー, firstDimensionValue, secondDimensionValue, filterDimensionValue)

DifferenceFromMean

説明: 指定されたディメンション値の平均値とメジャー値の差異。

構文: DifferenceFromMean(ディメンション, メジャー, firstDimensionValue, filterDimensionValue)

Direction

説明: 指定された2つのディメンション値のメジャー値間の方向 (たとえば、増加または減少) を説明する言葉。

構文: Direction(ディメンション, メジャー, firstDimensionValue, secondDimensionValue, filterDimensionValue, フレーズ)

Ending Label

説明: 系列の最後の期間の名前。

構文: Label(メジャー)

EndingValue

説明: 指定されたメジャーの系列における最後の期間の値。

構文: EndingValue(メジャー, filterDimensionValue)

Label

説明: 指定されたメジャーのラベル。

構文: Label(メジャー)

LargestNegativeChangeDifference

説明: 指定された系列における、期間ごとの差異の負の最大値 (絶対値ベース)。

構文: LargestNegativeChangeDifference(メジャー, filterDimensionValue)

LargestNegativeChangeEndingLabel

説明: 指定された系列において、期間ごとの差異が負の最大値 (絶対値ベース) となる終了期間の名前。

構文: LargestNegativeChangeEndingLabel(メジャー, filterDimensionValue)

LargestNegativeChangeEndingValue

説明: 指定された系列において、期間ごとの差異が負の最大値 (絶対値ベース) となる時の終了値。

構文: LargestNegativeChangeEndingValue(メジャー, filterDimensionValue)

LargestNegativeChangePercentDifference

説明: 指定された系列において、期間ごとの差異が負の最大値 (絶対値ベース) となる時の変化率。

構文: LargestNegativeChangePercentDifference(メジャー, filterDimensionValue)

LargestNegativeChangeStartingLabel

説明: 指定された系列において、期間ごとの差異が負の最大値 (絶対値ベース) となる開始期間の名前。

構文: LargestNegativeChangeStartingLabel(メジャー, filterDimensionValue)

LargestNegativeChangeStartingValue

説明: 指定された系列において、期間ごとの差異が負の最大値 (絶対値ベース) となる時の開始値。

構文: LargestNegativeChangeStartingValue(メジャー, filterDimensionValue)

LargestNegativePercentChangeDifference

説明: 指定された系列における、期間ごとの差異の負の最大値 (割合ベース)。

構文: LargestNegativePercentChangeDifference(メジャー, filterDimensionValue)

LargestNegativePercentChangeEndingLabel

説明: 指定された系列において、期間ごとの差異が負の最大値 (割合ベース) となる終了期間の名前。

構文: LargestNegativePercentChangeEndingLabel(メジャー, filterDimensionValue)

LargestNegativePercentChangeEndingValue

説明: 指定された系列において、期間ごとの差異が負の最大値 (割合ベース) となる時の終了値。

構文: LargestNegativePercentChangeEndingValue(メジャー, filterDimensionValue)

LargestNegativePercentChangePercentDifference

説明: 指定された系列において、期間ごとの差異が負の最大値 (割合ベース) となる時の変化率。

構文: LargestNegativePercentChangePercentDifference(メジャー, filterDimensionValue)

LargestNegativePercentChangeStartingLabel

説明: 指定された系列において、期間ごとの差異が負の最大値 (割合ベース) となる開始期間の名前。

構文: LargestNegativePercentChangeStartingLabel(メジャー, filterDimensionValue)

LargestNegativePercentChangeStartingValue

説明: 指定された系列において、期間ごとの差異が負の最大値 (割合ベース) となる時の開始値。

構文: LargestNegativePercentChangeStartingValue(メジャー, filterDimensionValue)

LargestPositiveChangeDifference

説明: 指定された系列における、期間ごとの差異の正の最大値 (絶対値ベース)。

構文: LargestPositiveChangeDifference(メジャー, filterDimensionValue)

LargestPositiveChangeEndingLabel

説明: 指定された系列において、期間ごとの差異が正の最大値 (絶対値ベース) となる終了期間の名前。

構文: LargestPositiveChangeEndingLabel(メジャー, filterDimensionValue)

LargestPositiveChangeEndingValue

説明: 指定された系列において、期間ごとの差異が正の最大値 (絶対値ベース) となる時の終了値。

構文: LargestPositiveChangeEndingValue(メジャー, filterDimensionValue)

LargestPositiveChangePercentDifference

説明: 指定された系列において、期間ごとの差異が正の最大値 (絶対値ベース) となる時の変化率。

構文: LargestPositiveChangePercentDifference(メジャー, filterDimensionValue)

LargestPositiveChangeStartingLabel

説明: 指定された系列において、期間ごとの差異が正の最大値 (絶対値ベース) となる開始期間の名前。

構文: LargestPositiveChangeStartingLabel(メジャー, filterDimensionValue)

LargestPositiveChangeStartingValue

説明: 指定された系列において、期間ごとの差異が正の最大値 (絶対値ベース) となる時の開始値。

構文: LargestPositiveChangeStartingValue(メジャー, filterDimensionValue)

LargestPositivePercentChangeDifference

説明: 指定された系列における、期間ごとの差異の正の最大値 (割合ベース)。

構文: LargestPositivePercentChangeDifference(メジャー, filterDimensionValue)

LargestPositivePercentChangeEndingLabel

説明: 指定された系列において、期間ごとの差異が正の最大値 (割合ベース) となる終了期間の名前。

構文: LargestPositivePercentChangeEndingLabel(メジャー, filterDimensionValue)

LargestPositivePercentChangeEndingValue

説明: 指定された系列において、期間ごとの差異が正の最大値 (割合ベース) となる時の終了値。

構文: LargestPositivePercentChangeEndingValue(メジャー, filterDimensionValue)

LargestPositivePercentChangePercentDifference

説明: 指定された系列において、期間ごとの差異が正の最大値 (割合ベース) となる時の変化率。

構文: LargestPositivePercentChangePercentDifference(メジャー, filterDimensionValue)

LargestPositivePercentChangeStartingLabel

説明: 指定された系列において、期間ごとの差異が正の最大値 (割合ベース) となる開始期間の名前。

構文: LargestPositivePercentChangeStartingLabel(メジャー, filterDimensionValue)

LargestPositivePercentChangeStartingValue

説明: 指定された系列において、期間ごとの差異が正の最大値 (割合ベース) となる時の開始値。

構文: LargestPositivePercentChangeStartingValue(メジャー, filterDimensionValue)

LongestStreakDifference

説明: 指定された系列において、増加または減少が連続した最長ストリークにおける差異。

構文: LongestStreakDifference(メジャー, filterDimensionValue)

LongestStreakDirection

説明: 指定された系列において、増加または減少が連続した最長ストリークの方向 (正または負)。

構文: LongestStreakDirection(メジャー, filterDimensionValue)

LongestStreakEndingLabel

説明: 指定された系列において、増加または減少が連続した最長ストリークの終了期間の名前。

構文: LongestStreakEndingLabel(メジャー, filterDimensionValue)

LongestStreakEndingValue

説明: 指定された系列において、増加または減少が連続した最長ストリークにおける終了値。

構文: LongestStreakEndingValue(メジャー, filterDimensionValue)

LongestStreakLength

説明: 指定された系列において、増加または減少が連続した期間の最大数。

構文: LongestStreakLength(メジャー, filterDimensionValue)

LongestStreakPercentDifference

説明: 指定された系列において、増加または減少が連続した最長ストリークにおける割合の差異。

構文: LongestStreakPercentDifference(メジャー, filterDimensionValue)

LongestStreakStartingLabel

説明: 指定された系列において、増加または減少が連続した最長ストリークの開始期間の名前。

構文: LongestStreakStartingLabel(メジャー, filterDimensionValue)

LongestStreakStartingValue

説明: 指定された系列において、増加または減少が連続した最長ストリークにおける開始値。

構文: LongestStreakStartingValue(メジャー, filterDimensionValue)

MaxLabel

説明: 指定されたメジャーの最大値を持つエンティティの名前。

構文: MaxLabel(ディメンション, メジャー, filterDimensionValue)

MaxValue

説明: 指定されたメジャーの最大値。

構文: MaxValue(メジャー)

Median

説明: 指定されたメジャーの中央値。

構文: Median(ディメンション, メジャー, filterDimensionValue)

MinLabel

説明: 指定されたメジャーの最小値を持つエンティティの名前。構文: MinLabel(ディメンション, メジャー, filterDimensionValue)

MinValue

説明: 指定されたメジャーの最小値。

構文: MinValue(ディメンション, メジャー, filterDimensionValue)

PercentDifference

説明: 指定された2つのディメンション値のメジャー値間の割合差。

構文: PercentDifference(ディメンション, メジャー, firstDimensionValue, secondDimensionValue, filterDimensionValue)

PercentOfWhole

説明: 指定されたディメンション値のメジャー値が、そのディメンションのメジャー値合計に占める割合。

構文: PercentOfWhole(ディメンション, メジャー, dimensionvalue, filterDimensionValue)

PeriodLabel

説明: 系列の n 番目の期間 (1 から開始) の名前。

構文: PeriodLabel(インデックス)

PeriodLabelNewest

説明: 系列の n 番目の期間 (最新から開始しカウントバック) の名前。

構文: PeriodLabelNewest(インデックス)

PeriodValue

説明: 系列の n 番目の期間 (1 から開始) における指定されたメジャーの値。

構文: PeriodValue(メジャー, インデックス, filterDimensionValue)

PeriodValueNewest

説明: 系列の n 番目の期間 (最新から開始しカウントバック) における指定されたメジャーの値。

構文: PeriodValueNewest(メジャー, インデックス)

Range

説明: 指定されたメジャーの最大値と最小値の差。

構文: Range(ディメンション, メジャー, filterDimensionValue)

SortAscendingLabel

説明: 所定のランク (1 から開始) で降順に並べ替えられた指定されたメジャーのエンティティ名。

構文: SortAscendingLabel(メジャー, ランク, ディメンション, filterDimensionValue)

SortAscendingValue

説明: 所定のランク(1 から開始) で昇順に並べ替えられた指定されたメジャーの値。

構文: SortAscendingValue(メジャー, ランク, ディメンション, filterDimensionValue)

SortDescendingLabel

説明: 所定のランク(1 から開始) で降順に並べ替えられた指定されたメジャーのエンティティ名。

構文: SortDescendingLabel(メジャー, ランク, ディメンション, filterDimensionValue)

SortDescendingValue

説明: 所定のランク(1 から開始) で降順に並べ替えられた指定されたメジャーの値。

構文: SortDescendingValue(メジャー, ランク, ディメンション, filterDimensionValue)

StartingLabel

説明: 系列の最初の期間の名前。

構文: StartingLabel()

StartingValue

説明: 指定されたメジャーの系列における最初の期間の値。

構文: StartingValue(メジャー, filterDimensionValue)

StartToFinishDifference

説明: 指定された系列の最初と最後の期間の値の差。

構文: StartToFinishDifference(メジャー, filterDimensionValue)

StartToFinishPercentDifference

説明: 指定された系列の最初と最後の期間の値の割合の差。

構文: StartToFinishPercentDifference(メジャー, filterDimensionValue)

StdDev

説明: 指定されたメジャーの標準偏差の値。

構文: StdDev(ディメンション, メジャー, filterDimensionValue)

Sum

説明: 指定された 2 つのディメンション値のメジャー値の合計。

構文: Sum(ディメンション, メジャー, firstDimensionValue, secondDimensionValue, filterDimensionValue)

Total

説明: 指定されたメジャーの合計値。

構文: Total(ディメンション, メジャー, filterDimensionValue)

Value

説明: 指定されたディメンション、メジャー、ディメンション値のメトリクス値。

構文: Value(ディメンション, メジャー, ディメンション値, filterDimensionValue)

Z-Score

説明: 指定されたメジャーの Z スコア。

構文: Z-Score(ディメンション, メジャー, firstDimensionValue, filterDimensionValue)

Tableau データストーリーのカスタマイズ: コンテンツの非表示と並べ替え

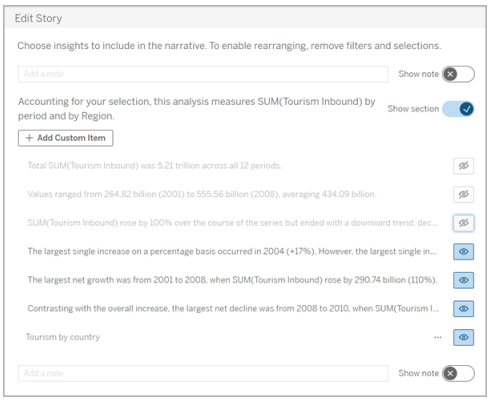
Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

ストーリー内のコンテンツを非表示にしたり並べ替えたりすることで、自分やオーディエンスにとって最も重要なコンテンツで、Tableau データストーリーをカスタマイズすることができます。

コンテンツとセクションを非表示にする

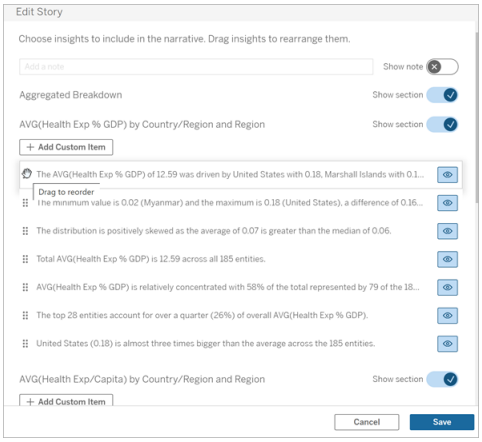
1. **データストーリー**を作成し、**[編集]** をクリックして [編集] ダイアログ ボックスを開きます。
2. セクションの表示/非表示を設定するには、**[Show section (セクションの表示)]** スイッチをクリックしてオン/オフを切り替えます。
3. 個々の文の右側にある青いボックスにカーソルを合わせ、ボックスをクリックすると表示/非表示が切り替わります。
4. **[保存]** をクリックします。



[表示] に設定されている文とセクションのみがデータストーリーに表示されます。

セクション内のコンテンツを並べ替える

1. **データストーリー**を作成し、**[編集]** をクリックして [編集] ダイアログ ボックスを開きます。
2. 並べ替えるコンテンツの左側のメニューにカーソルを合わせます。カーソルが手のアイコンに変わります。
3. カーソルでアイテムをクリックし、同じセクション内の任意の場所にドラッグします。
4. **[保存]** をクリックします。



設定した順序で**データストーリー**に文章が表示されます。

注: 現在、コンテンツは同じセクション内でのみ移動できます。セクション全体の移動はまだサポートされていません。

Tableau データストーリーにデータを追加する

Tableau データストーリーの重要な変更

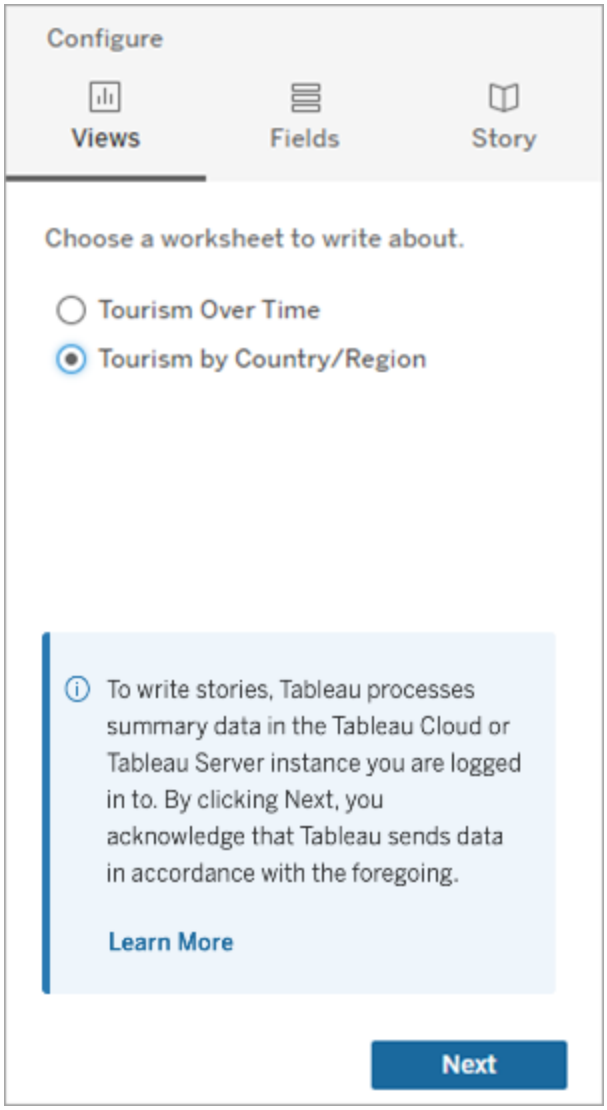
Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

データストーリーは、現在、2 つのディメンションと1 つのメジャー、または 1 つのディメンションと最大 10 個のメジャーが含まれるストーリーをサポートしています。ダッシュボードに表示する必要のないデータについて記述する場合は、、非表示のシートを使用してダッシュボードを簡素化します。ストーリーに 3 つ以上のディメンションを追加する場合は、ディメンションを連結するか、複数のデータストーリーを作成して積み重ねます。

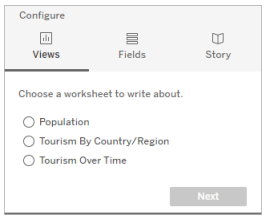
非表示のシートを使用する

Tableau データストーリーに含めたいデータがあり、インサイトの獲得をサポートしたすべてのデータを表示する必要がない場合は、非表示のシートを使用して、ダッシュボードを煩雑にすることなく、ストーリーに追加のメジャーとディメンションを取り込むことができます。

1. **データストーリー** オブジェクトをダッシュボードにドラッグして、[データストーリー] ダイアログ ボックスに記述できるワークシートを確認します。この例では、記述できる 2 つのワークシートがあります。

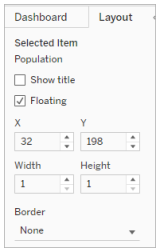


2. 左側のメニューに移動して、**[Floating (フローティング)]** を選択し、そのシートをダッシュボードにドラッグすることにより、「人口」などの別のデータ ソースを取り込みます。



[データ ストーリー] ダイアログ ボックスが、利用可能な新しいデータ ソースで更新されます。

3. **[レイアウト]** をクリックし、サイズを 1 x 1 に調整してシートを非表示にしますが、参照元データは ストーリーに残します。



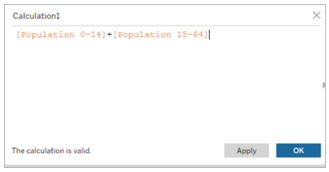
これで、この非表示のシートを使用してストーリーを構成できます。

> **注:** シートを非表示のままにするには、追加の凡例要素の選択を解除する必要がある場合があります。

ディメンションを連結する

データに 3 つのディメンションと1 つのメジャーが含まれていて、データが **[Discrete Story (不連続ストーリー)]** である場合、計算フィールドを作成することにより、これらのディメンションのうちの 2 つを連結 (リンク) できます。

1. ストーリーで使用するワークシートで、**[分析]** をクリックし、**[計算フィールドの作成]** を選択します。
2. 計算フィールドに名前を付け、次の式を使用して計算を作成し、+ 記号を使用してディメンションを結合します。
   [Dimension 1] + [Dimension 2]



ヒント: ディメンションを **[計算フィールド]** ボックスにドラッグして、式に配置します。

3. **[OK]** をクリックします。
4. 新しい計算フィールドを **[Detail (詳細)]** ペインにドラッグして、データストーリーでアクセスできるようにします。

複数のデータストーリーを積み重ねる

複数の**データストーリー**を作成し、ダッシュボードに垂直または水平に積み重ねて、さらに多くのメジャーとディメンションについて記述します。

たとえば、実際の収益とベンチマーク収益についてのストーリーを作成したい場合は、2 つの異なるストーリー (1 つは実際の収益と最初のベンチマーク、もう1 つは実際の収益と2 番目のベンチマークについてのストーリー) を作成して、これらのストーリーを比較することができます。

ダッシュボードに Tableau ポップアップ データストーリーを追加する

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

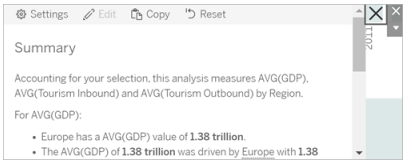
ポップアップ ウィンドウに Tableau データストーリーを配置できます。ユーザーは、ストーリーを開いて読み、完了したら閉じることができます。これは、すでに確立されているダッシュボードで使用されているスペースを節約したり、ダッシュボード上の乱雑な情報の量を減らしたりするための優れた方法です。

1. ダッシュボードに Tableau データストーリーを追加する。
2. メニューをクリックして、**[Floating (フローティング)]** を選択し、コンテナーを **[Floating (フローティング)]** に設定します。

ヒント: コンテナーを **[Floating (フローティング)]** に設定するもう1 つの方法は、Shift キーを押しながらコンテナーをダッシュボードにドラッグすることです。

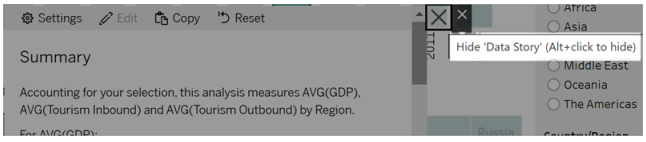
3. 左側の列の **[レイアウト]** タブに移動し、背景色を白に設定します。

4. 浮動コンテナーに関連付けられているメニューをクリックして設定を表示し、**[表示/非表示ボタンの追加]** を選択します。これにより、ストーリーを表示または非表示にできる [X] アイコンが作成されます。



注: ストーリーが選択されている場合、ボタンは、データストーリーオブジェクトの一部であるサイドバーオプションによって部分的に隠されています。ボタンを表示するには、ストーリーの外側をクリックします。

5. [X] アイコンにカーソルを合わせると、ストーリーを開いたり閉じたりする手順が表示されます。この例では、Alt キーを押しながら [X] アイコンをクリックするように求められます。



6. プロンプトに示されているキーを押しながら [X] アイコンをクリックして、ストーリーを折りたたみます。

ストーリーは折りたたまれますが、ユーザーが必要に応じてストーリーを展開できるように、メニューはダッシュボードに残ります。必要に応じて、折りたたみ可能なフローティング ストーリーをダッシュボード内で移動できます。

Tableau データ ストーリーでカスタム メジャー関係を作成する

Tableau データ ストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

Tableau データストーリーで関係ストーリーを作成すると、データを別のデータセットと比較して確認することができます。関係ストーリーを作成するには、少なくとも 2 つのメジャーと1 つのディメンションが必要です。多くの場合、比較したいデータはある列のデータでグループ化されています。たとえば、「年」や「月」などの名前の列で、「2022」や「3 月」などの値が設定されています。

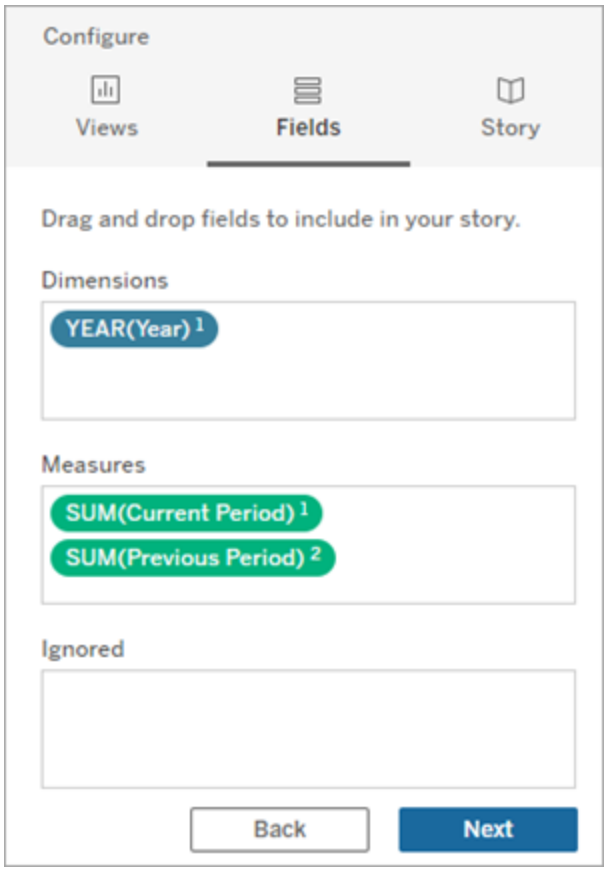
簡単な計算フィールドを作成して「2022」と「2021」あるいは「2 月」と「3 月」で分離すると、2 つの期間を関係ストーリーで比較できます。

1. **データ ストーリー**で使用するシートから始めます。
2. **[分析]** をクリックして **[計算フィールドの作成]** を選択します。
3. 「現在の期間」などの名前の計算フィールドを作成します。

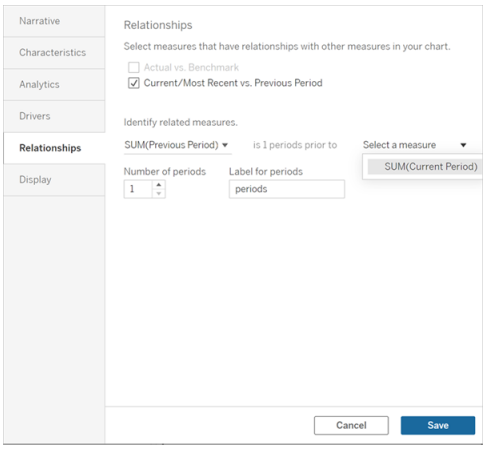
ヒント: 計算フィールドはここに示した構造に従いますが、ディメンションやメジャーの名前 (オレンジ色のテキスト) はご利用の名前に置き換えてください。

4. 「前の期間」などの名前の計算フィールドを作成します。

5. 新しいメジャーを **[詳細]** マークにドラッグします。
6. ダッシュボードでストーリーをクリックし、2 つの新しいメジャーを**データ ストーリー**に追加します。



7. ダッシュボードで、データ ストーリー オブジェクトの左上隅にある **[設定]** アイコンをクリックします。
8. [データ ストーリー] ダイアログ ボックスで **[関係]** タブをクリックします。
9. 2 つのカスタム メジャーを使用した関係ストーリーを設定します。



10. **[保存]** をクリックします。

これで、カスタム メジャーを比較する文がストーリーに記述されるようになりました。

Tableau データストーリーのパラメーターを更新する

Tableau データストーリーの重要な変更

Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

ダッシュボードにパラメーターを追加し、データストーリーを使用している場合、パラメーターをクリックするとビジュアライゼーションが更新されますが、ストーリーは更新されません。これは、パラメーターがフィルターのように参照元データを更新しないためです。

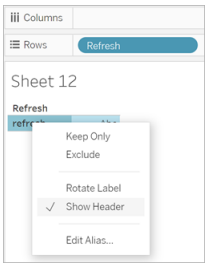
パラメーター データを使用して Tableau データストーリーを更新するには、ダッシュボードに [更新] ボタンを追加して、パラメーターに合わせてストーリーを更新します。

1. ワークブックに新しいシートを作成します。
2. 次の情報を使用して、新しいシートに**計算フィールド**を作成します。

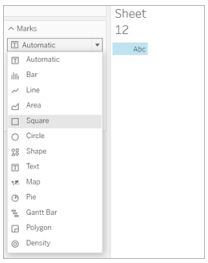
   名前: 更新

コンテンツ: "更新"

3. **[OK]** をクリックし、新しい計算フィールド(更新) を新しいシートにドラッグします。
4. フィールドを右クリックし、**[ヘッダーの表示]** をクリックしてヘッダーを非表示にします。



5. ボタンの形状を選択します。



6. ダッシュボードに戻り、更新ボタンを含むシートをパラメーターの横のダッシュボードにドラッグします。
7. ボタンにカーソルを合わせ、**[その他のオプション]** を選択し、**[タイトル]** をクリックしてタイトルを非表示にします。
8. ボタンにカーソルを合わせ、**[フィルターとして使用]** をクリックします。

9. パラメーターを調整してから、新しい更新ボタンをクリックします。ボタンをクリックすると、パラメーターに合わせてストーリーが更新されます。

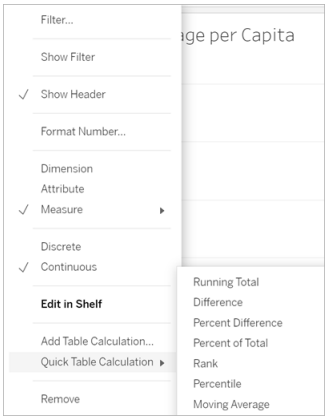
Tableau データストーリーで表計算を使用する

Tableau データストーリーの重要な変更

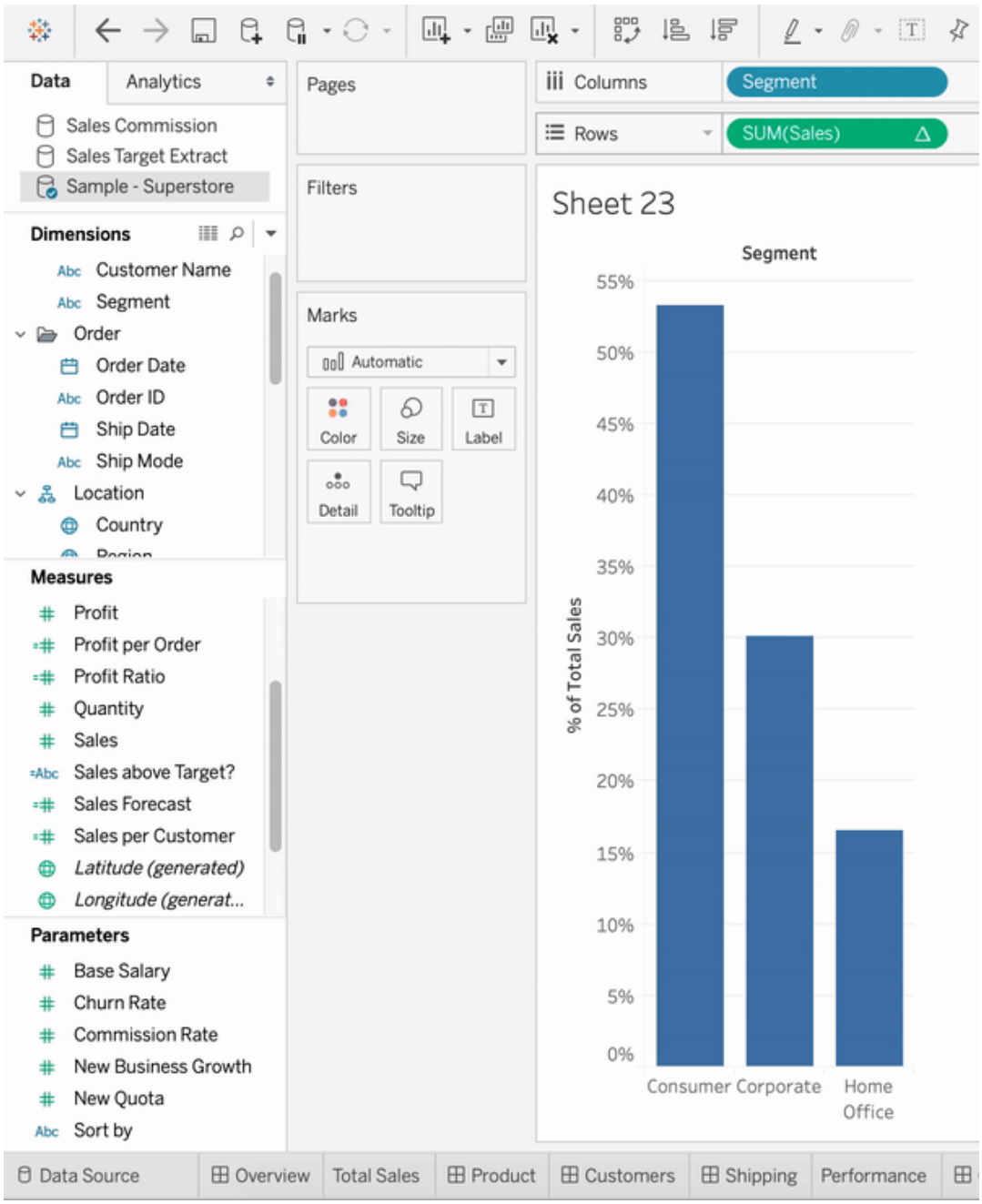
Tableau データストーリーは、2025 年 1 月 (2025.1) の Tableau Desktop、Tableau Cloud、Tableau Server で廃止されます。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「Tableau AI を搭載した Tableau Pulse がデータエクスペリエンスを刷新」を参照してください。

表計算の作成に使用したメジャーに加えて、Tableau データストーリーのメジャーとして**表計算**を使用することができます。

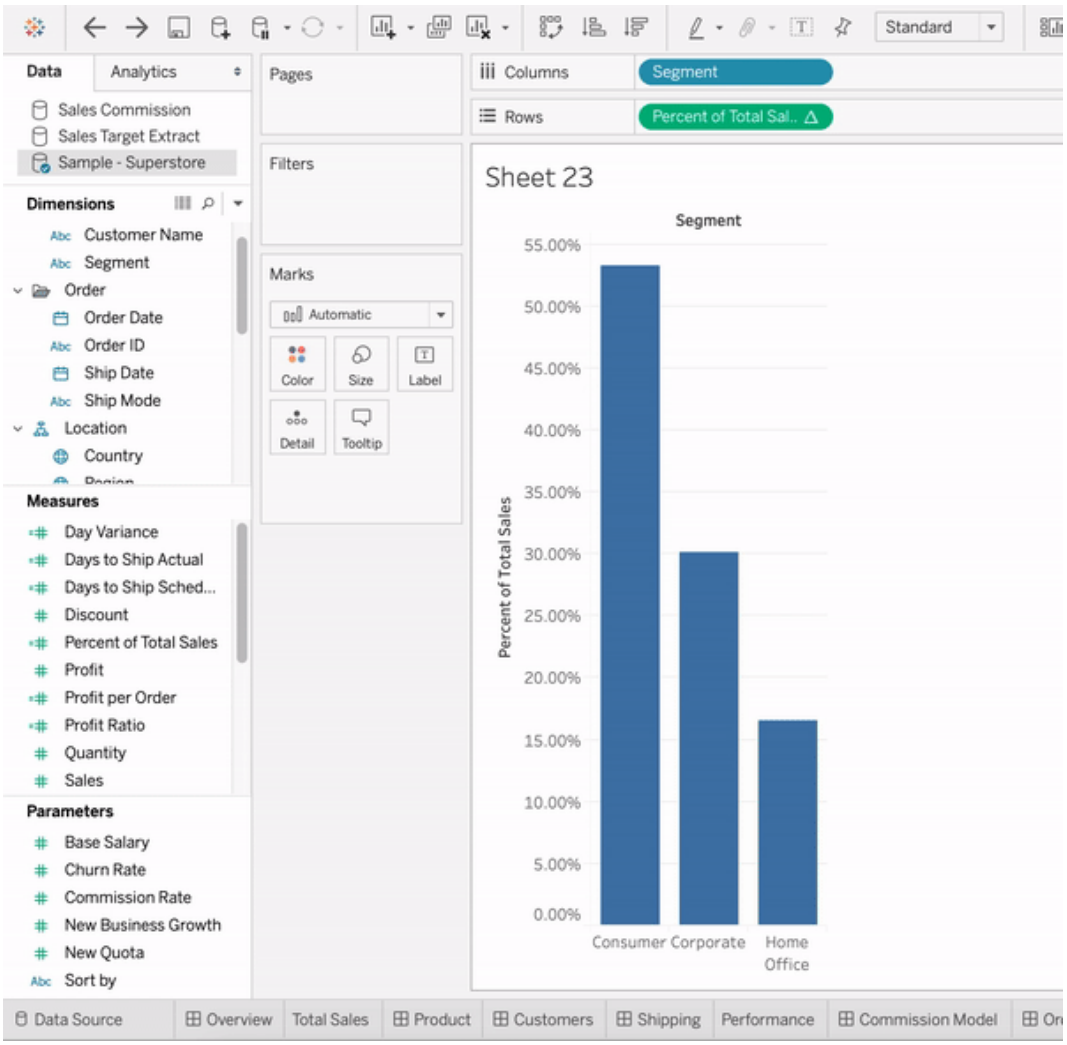
1. **マーク**カードから、フィールドの右側をクリックしてメニューを開き、**[簡易表計算]** をクリックします。



2. 表計算を作成したら、それを**[データ]** ペインの**[メジャー]** にドラッグして名前を変更します。



3. 元のメジャー (例: 売上高) を**[行]** シェルフにドラッグして戻し、新しい計算 (例: 売上高の合計) を**[ツールヒント]** にドラッグします。ビジュアライゼーションは同じままでも、新しいメジャーにアクセスできるようになりました。



4. ダッシュボードに移動し、**データ ストーリー** オブジェクトをダッシュボードにドラッグします。ストーリーを作成すると、両方のメジャーが表示されます。

「データの説明を見る」を使用して、インサイトをより早く発見する

データガイドの「データの説明を見る」を使用すると、データを検索する際に、Viz 内のマークを調査しながら新たな発見を得て、データを深く掘り下げることができます。「データの説明を見る」を使用して、ダッシュボード、シート、または選択したマークを分析し、参照元データに異常値や相関関係がある可能性があるかどうかを確認することができます。「データの説明を見る」を実行す

ると、統計モデルが構築され、現在のビューで使用されていないデータソースの潜在的な関連データなど、Viz 内の個々のマークについて考えられる説明が提供されます。

「データの説明を見る」の実行と説明の探索については、「データの説明を見る」を使い始めるを参照してください。

> **注**: このトピックでは、Tableau 2021.2 以降のバージョンで「データの説明を見る」がどのように機能するかについて説明します。以前のバージョンの Tableau を使用している場合は、「データの説明を見る」のヘルプのバージョン 2021.1 でこのトピックをお読みください。

さまざまなビューを構築するときに「データの説明を見る」を出発点として使用すると、データをさらに深く検索してより良い質問をするのに役立ちます。詳細については、「データの説明を見る」を使用して分析を強化する参照してください。「データの説明を見る」で使用するデータソースをより興味深くするための特性については、「データの説明を見る」を使用する場合の要件と考慮事項を参照してください。

「データの説明を見る」へのアクセス

デフォルトでは、「データの説明を見る」はサイトレベルで有効になっています。サーバー管理者 (Tableau Server) とサイト管理者 (Tableau Cloud) は、サイトで「データの説明を見る」を使用できるかどうかを制御できます。詳細については、サイトに対する「データの説明を見る」の無効化または有効化を参照してください。

ワークブックの編集が可能で、かつ、ワークブックの「データの説明を見る」の実行権限を持っている作成者は、編集モードで「データの説明を見る」を実行できます。「データの説明を見る」の実行権限をを持つすべてのユーザーは、Tableau Cloud および Tableau Serverの表示モードで「データの説明を見る」を実行できます。

サイト管理者によって許可されている場合は、他の Tableau Cloud や Tableau Server ユーザーとメールや Slack を介して説明を表示モードで共有することができます。詳細については、「ユーザーがメールや Slack を介して説明を共有できるように Tableau を構成する」を参照してください。

作成者は、「データの説明を見る」の設定を使用して、[データガイド] ペインに表示される説明のタイプを制御できます。

「データの説明を見る」、説明タイプ、およびフィールドへのアクセスの制御については、「データの説明を見る」へのアクセスの制御を参照してください。

「データの説明を見る」を使用して分析を強化する

「データの説明を見る」は、データ内の関係を明らかにし、説明するツールです。関係の原因やデータの解釈方法は教えてくれません。**あなたこそがデータのエキスパートです。**ドメインに関する知識や直感を持つことで、さまざまなビューを使用しながらさらに深く探索できる興味深い特性を決定することができます。関連情報については、「データの説明を見る」のしくみ および「データの説明を見る」を使用する場合の要件と考慮事項 を参照してください。

「データの説明を見る」のしくみと、分析を強化するために「データの説明を見る」を使用する方法に関する詳細情報については、Tableau Conference のプレゼンテーションを参照してください。

- アナリストから統計学者へ: 実際の「データの説明を見る」(1 時間)
- 「データの説明を見る」の活用 (45 分)

「データの説明を見る」を使い始める

Viz のマークを詳しく調べるときは、分析のフローで「データの説明を見る」を使用します。「データの説明を見る」は、[Data Guide (データガイド)] ペインが開いているときに自動的に実行され、現在の選択 (ダッシュボード、シート、またはマーク) に基づいて更新されます。

「データの説明を見る」を使用する

- ダッシュボード、シート、マークで「データの説明を見る」を実行する
- 説明を掘り下げる
- 分析したフィールドを表示する
- 説明での用語と概念
- 説明のタイプ

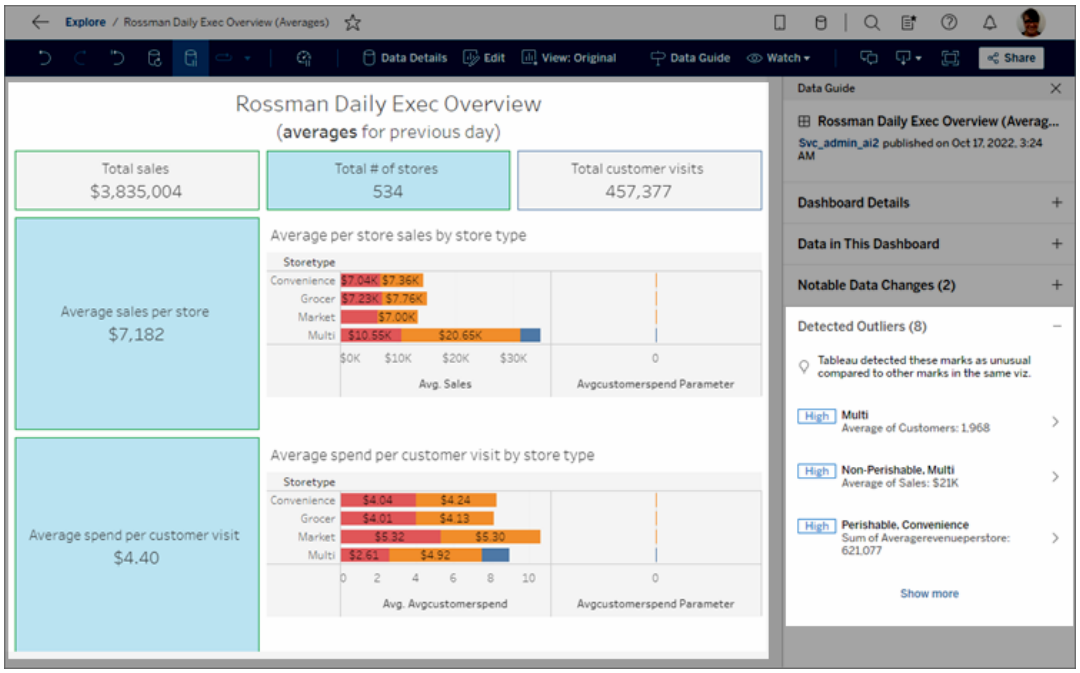
ワークブックの作成とアクセスの制御

- 「データの説明を見る」を使用する場合の要件と考慮事項
- 「データの説明を見る」設定の変更 (作成者のみ)
- 「データの説明を見る」へのアクセスの制御
- サイトに対する「データの説明を見る」の無効化または有効化
- 「データの説明を見る」のしくみ

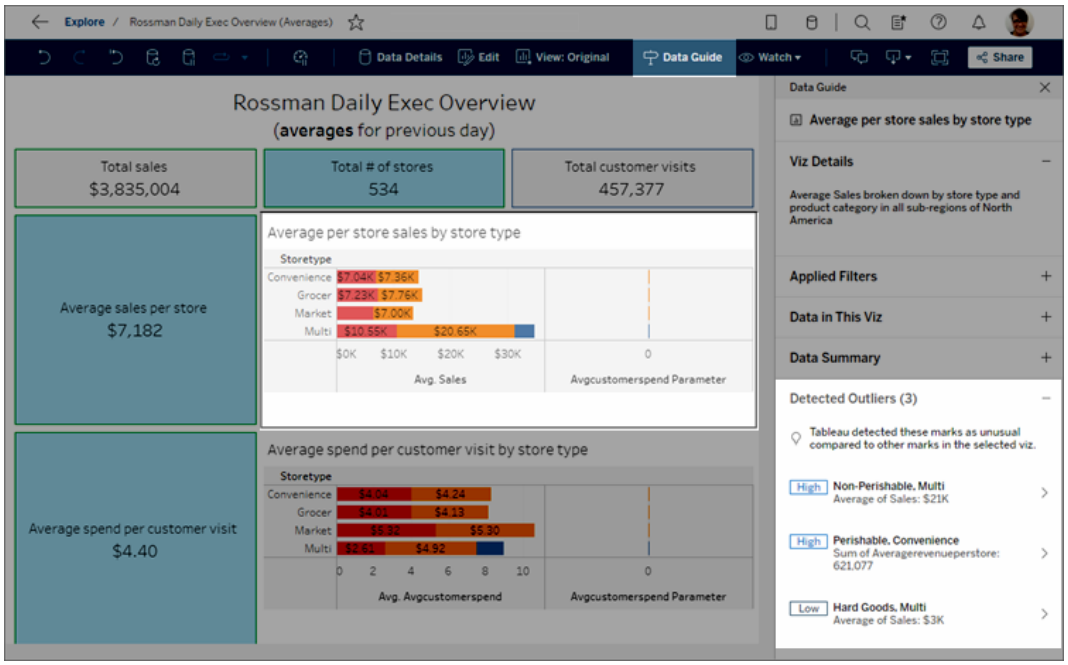
ダッシュボード、シート、マークで「データの説明を見る」を実行する

Tableau Desktop、Tableau Cloud、Tableau Server で「データの説明を見る」を実行するための基本的な手順は次のとおりです。

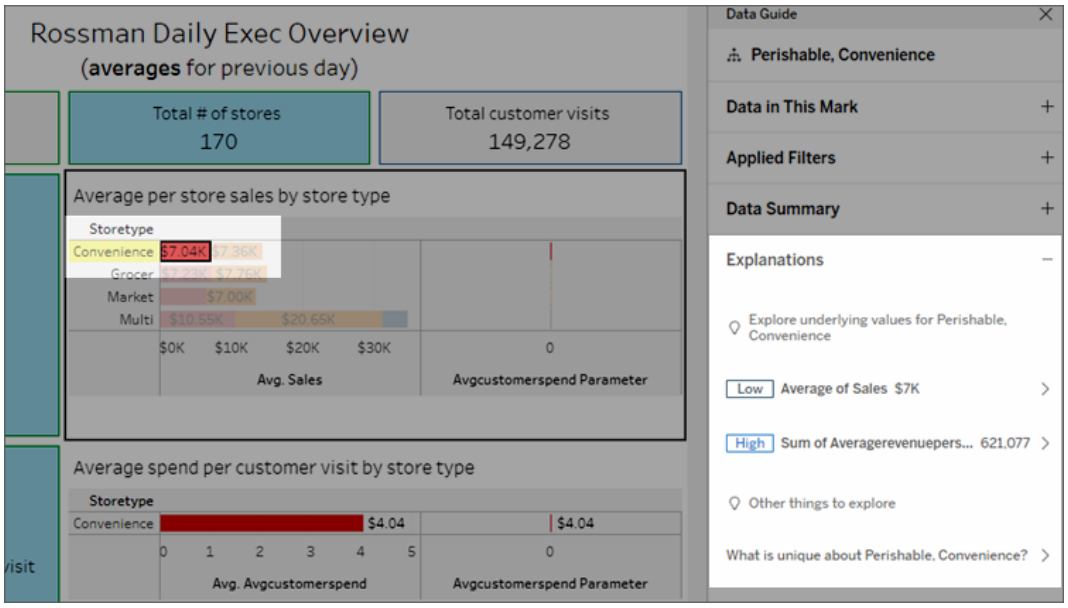
1. ワークブックでダッシュボードまたはシートを開きます。
2. ナビゲーション ツールで、**[データ ガイド]** を選択し、[データ ガイド] ペインを開きます。
3. ダッシュボードが開いている場合は、「データの説明を見る」が外れ値があるかどうかを分析します。



ダッシュボードでシートを選択している場合、「データの説明を見る」がそのシートのマークを分析し、外れ値を検出します。



ダッシュボードでマークを選択している場合、「データの説明を見る」がそのマークだけを分析して説明します。



説明されているマークは、対応する説明を選択するとViz で強調表示されます。

必要に応じて、Viz 内のマークを選択し、マークにカーソルを合わせることができます。ツールヒント メニューで、**[データ ガイド]** を選択します。

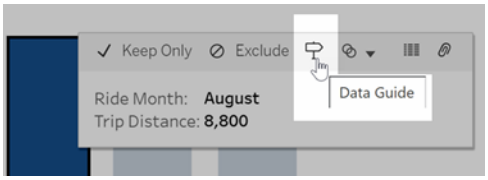
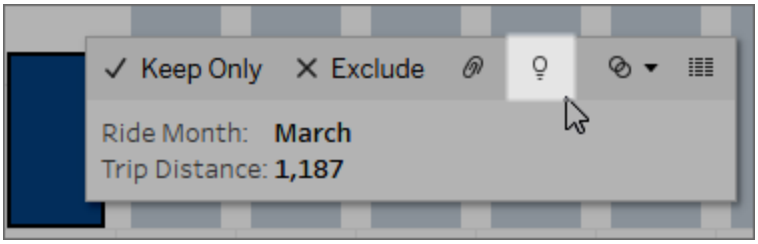


Tableau Public で、ツールヒント メニューの電球を選択して、「データの説明を見る」を実行します。



分析されたマークの値について考えられる説明は、[Data Guide (データ ガイド)] ペインに表示されます。別の説明名を選択して詳細を展開し、探索を始めます。

説明を見るために必要な「データの説明を見る」のパーミッション

[検出された外れ値] が「Viz の所有者に連絡してください」という注記とともに表示される場合、これらのタイプの説明を表示するにはパーミッションが必要になります。所有者名を選択して、所有者のメール アドレスが記載された Tableau コンテンツ ページに移動します。所有者に連絡して、ワークブックまたはビューの「データの説明を見る」のパーミッションを付与するよう依頼してください。

ワークブックの所有者である場合、パーミッションの設定の詳細については、「「データの説明を見る」を使用できるユーザーとユーザーが表示できる内容の制御」を参照してください。

「データの説明を見る」を使用するためのヒント

- 複数のマークを選択して、相互に比較することができます。
- ビューには、SUM、AVG、COUNT、COUNTD、AGG などの計算フィールドを使用して集計されたマークを含める必要があります。

- 「データの説明を見る」では、選択されたマークのタイプを分析できない場合、その理由を示すメッセージが表示されます。詳細については、「「データの説明を見る」が使用できない状況」を参照してください。
- データは、単一のプライマリデータソースから取得する必要があります。「データの説明を見る」は、ブレンドされたデータソース、またはキューブデータソースでは機能しません。
- 「データの説明を見る」で使用するデータソースをより興味深くするための特性については、「データの説明を見る」を使用する場合の要件と考慮事項を参照してください。

説明を掘り下げる

1. [データガイド] ペインで、説明名を選択して詳細を表示します。

   説明を選択すると、詳細を展開したり、折りたたんだりすることができます。

2. スクロールすると、説明の詳細が表示されます。

3. 解説のチャートにカーソルを合わせると、各データポイントの詳細が表示されます。**[開く]** アイコンを選択すると、Viz の拡大版を表示できます。

   Creator または Explorer が編集のためにビューを開いた場合は、**[開く]** アイコンを選択すると、新しいワークシートで Viz が開き、より詳しくデータを調べることができます。

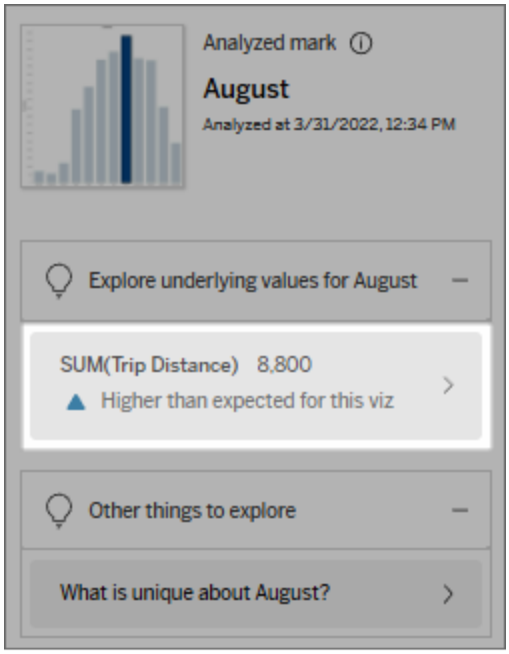
   > **注**: 編集権限を持つ Creator と Explorer は、「データの説明を見る」の設定を制御することもできます。詳細については、「データの説明を見る」へのアクセスの制御を参照してください。

4. ヘルプアイコン ⓘ にカーソルを合わせると、説明のためのツールヒントのヘルプを参照できます。ヘルプアイコン ⓘ を選択して、ツールヒントを開いたままにします。**[詳細]** リンクを選択すると、関連するヘルプトピックが開きます。

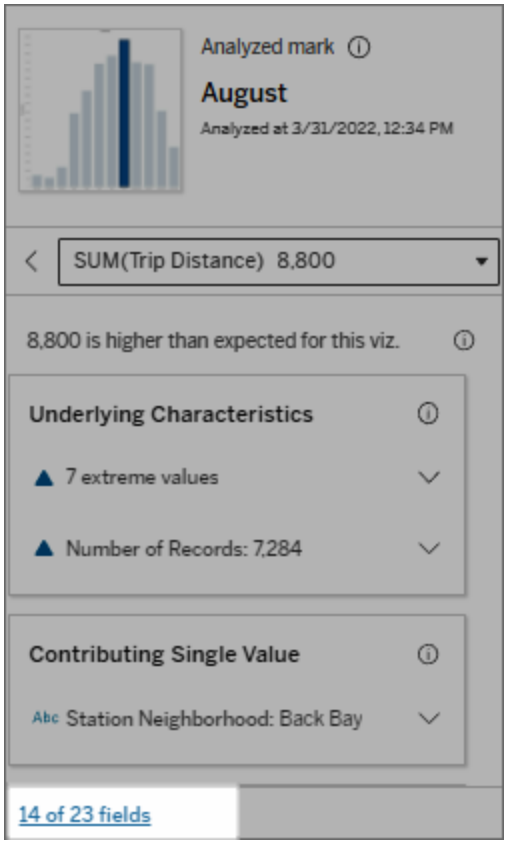
分析したフィールドを表示する

1. ダッシュボード、シート、マークで「データの説明を見る」を実行する。

2. [データガイド] ペインの **[Explore underlying values for (基になる値の探索)]** で、対象のメジャー名を選択します。



3. ペインの下部にある *[number-of-fields (フィールド数)]* リンクを選択します。



作成者は、「データの説明を見る」の設定を開いて、分析に含めるフィールドを制御することもできます。詳細については、「統計分析に使用するフィールドの変更」を参照してください。

説明での用語と概念

説明では、以下の用語や概念が頻繁に登場します。「データの説明を見る」におけるこれらの意味を理解しておくと役立つ場合があります。

マークとは

マークとは、参照元のデータの中からいくつかのレコードの値を集約したものであり、選択可能なデータポイントとして表示されます。1 つのマークは、1 つのレコード、または複数のレコードをまとめ

て作られています。Tableau では、線、図形、棒、セル テキストなどのさまざまな方法でマークを表示できます。

Tableau は、ビュー内のフィールドの交差に基づいて、マークを作り上げるレコードを取得します。

分析されたマークとは、「データの説明を見る」で分析された、ダッシュボードやシートのマークを指します。

マークの詳細については、「マーク」を参照してください。

予測とは

マークの予測値は、Viz の参照元データの予測範囲の中央値です。予測範囲は、分析されたマークに対して統計モデルが予測する 15 パーセンタイルから 85 パーセンタイルまでの値の範囲です。Tableau は、選択したマークに対して統計分析を実行するたびに、予測範囲を算出します。

期待値の概要で、マークが*予想よりも低いか予想よりも高い*と説明されている場合、集計済みのマーク値が、統計モデルがマークで予測している値の範囲外にあることを意味します。予測値の概要で、マークが*予測よりもわずかに低いか予測よりもわずかに高い*と説明されているか、*自然変動幅の範囲内にある*と説明されている場合は、集約されたマーク値は予測範囲内にあるが、中央値よりも低いか高いことを意味します。

詳細については、「予測範囲とは」を参照してください。

ディメンションとメジャーとは

データベースの各列名はフィールドです。たとえば、製品名と売上は、それぞれフィールドです。Tableau では、製品名などのように、データを分類するフィールドはディメンションと呼びます。いっぽう、売上などのように、定量化できるデータを持つフィールドはメジャーと呼びます。Tableau は、メジャーをビューにドラッグすると、デフォルトでメジャーを集計します。

いくつかの説明では、参照元のレコードの値とそれらの値を集計したものが、分析されたマークの値にどのように影響しているかを説明しています。別の説明では、分析されたマークのディメンション全体で、値がどのように分布しているかを説明しています。

マーク上で「データの説明を見る」を実行すると、ビューに表示されていない、データソースのディメンションとメジャーについても分析で考慮されます。それらのフィールドは、非視覚化ディメンションや非視覚化メジャーと呼びます。

ディメンションとメジャーの詳細については、「ディメンションとメジャー」を参照してください。

集計や集約とは

集計や集約とは、要約したり、合計したりした値です。Tableau では、メジャーを行、列、マークカードオプション、ビューなどにドラッグするたびに、SUM や AVG などの集計値を自動的に計算します。たとえば、メジャーが SUM(売上) や AVG(売上) などと表示されていると、メジャーがどのように集計されているかを示しています。

「データの説明を見る」を使用するには、SUM、AVG、COUNT、COUNTD、AGG などで集計されたメジャーをビジュアライゼーションで使用している必要があります。

集計の詳細については、「Tableau でのデータ集計」を参照してください。

レコードの値とは

レコードは、データベーステーブルの行です。行には、各フィールドに対応する値が含まれています。この例では、カテゴリ、製品名、売上などがフィールド(または列)です。家具、フロアランプ、96 ドルなどが値です。

カテゴリ	製品名	売上
家具	フロアランプ	$96.00

分布とは

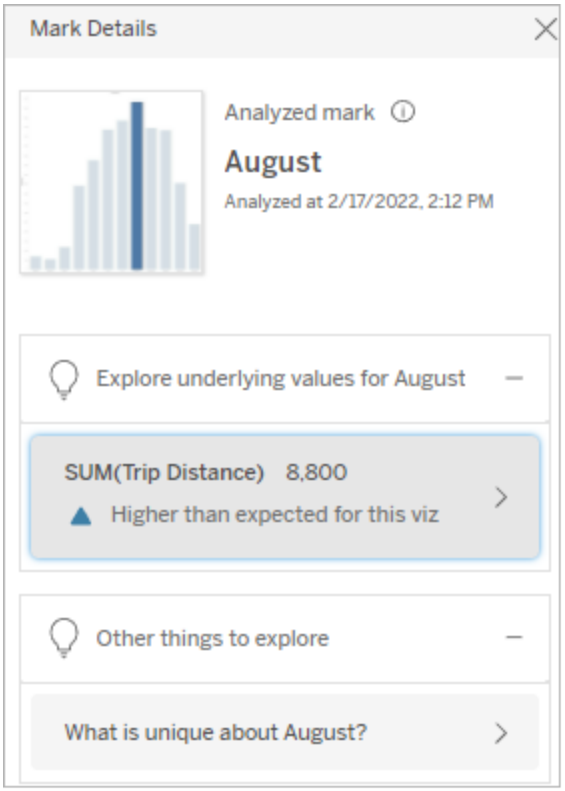
分布とは、データが取り得るすべての値 (または間隔) のリストです。各値が発生する頻度 (発生頻度) も示します。

「データの説明を見る」の説明タイプ

Tableau は、viz やダッシュボードで新しいマークを選択して「データの説明を見る」を実行するたびに、そのマークとワークブックの参照元データを使用して新たな統計分析を行います。考えられる説明は、[データガイド] ペインの展開可能なセクションに表示されます。「データの説明を見る」が説明を分析して評価するしくみについては、「「データの説明を見る」のしくみ」を参照してください。

基になる値を探索する

このセクションには、説明可能な各メジャー (*ターゲットメジャー*と呼ばれます) の説明が一覧表示されます。一覧表示される各説明は、分析されたマーク上で調べたターゲットメジャーの値との関係を説明しています。実世界でそのデータが実用的にどのような意味を持つかに基づいて、「データの説明を見る」で検出された関係に意味があり、確認する価値があるかどうかを判断します。



この例では、走行距離がターゲット メジャーです

基になる特性

この説明では、ビュー内のマークの参照元レコードが、説明されているメジャーの集計値にどのように影響している可能性があるかを解説しています。マークの属性には、そのマークの極値、Null 値、レコード数、平均値などがあります。

注: 説明で使用する一般的な用語の定義については、「説明での用語と概念」を参照してください。

極値

この説明タイプでは、1 つまたは複数のレコードの値が、ほとんどのレコードと比べて大幅に高いか低いかを説明します。説明しているモデルがある場合は、分析されたマークのターゲットメジャーに極値が影響を与えていることを示します。

マークに極値が含まれていても、ただちに、それらの値は外れ値である、または、ビューからそれらのレコードを除外する必要がある、という意味ではありません。選択すべき内容は分析に応じて異なります。説明は、単にマークの極値について指摘しているだけです。たとえば、バナナの価格が 10 セントではなく10 ドルであるレコードであれば、誤って入力された値だとわかるかもしれません。または、特定の営業担当者がある四半期に優れた成績を収めていたことが明らかになるかもしれません。

注: パブリッシュされたワークブックの表示モードで説明を表示するには、作成者がその説明の表示を有効にする必要があります。詳細については、「データの説明を見る」へのアクセスの制御を参照してください。

この説明では以下を示します。

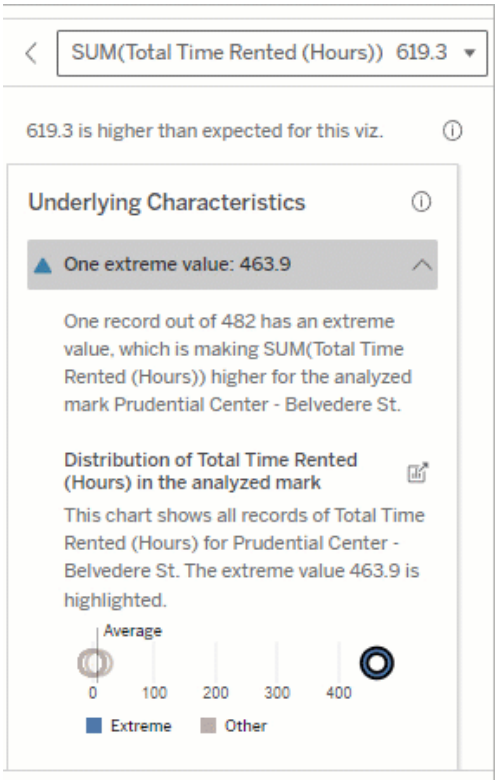
- 分析されたマークの参照元レコードの数。
- ターゲットメジャーの値に影響する 1 つまたは複数の極値。
- マーク内の値の分布。
- 各値の分布に対応するレコードの詳細。

探索オプションを以下に示します。

- グラフの円にカーソルを合わせると、対応する値を表示できます。
- 詳細リストの下にある左矢印または右矢印を選択すると、レコードの詳細をスクロールできます。
- 利用可能な場合は、**[フル データを表示]** を選択してから、**[フル データ]** タブを選択すると、テーブル内にあるすべてのレコードを表示できます。
- **[開く]** アイコンを選択すると、Viz の拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- レコード数が少ない場合は、それらの値を極値と比較して調べてください。
- 極値が他のレコード値よりも大幅に高いまたは低い場合は、極値を除外すると、分析されたマークの値がどのように変化するか確認してください。
- 極値がある場合とない場合のデータを検討するときは、そのデータにつ



この例では、463 時間というレンタル時間の 1 つの極値が、予測よりも高い 613 時間という総レンタル時間に影響しています。

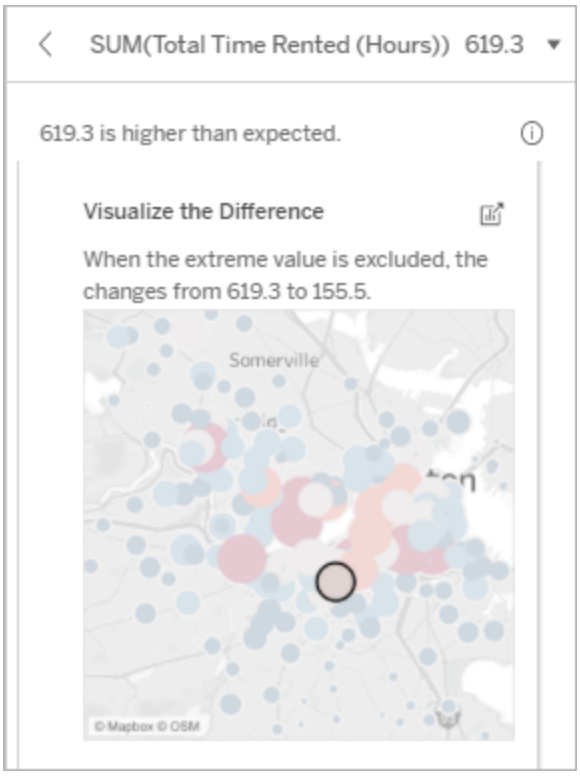
この高い値の理由としては、誰かが自転車を返却したときにドッキングするのを忘れた可能性が考えられます。このような場合は、作成者は将来の分析のためにこの値を除外することをお勧めします。

いての実用面での知識を活用する必要があります。

違いを視覚化する

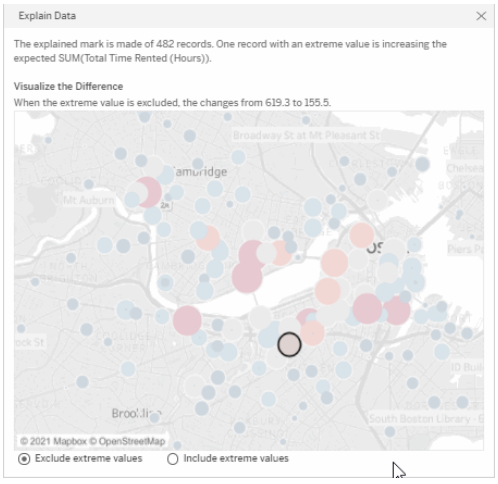
このセクションでは次のことを示しています。

- 極値を除いた場合の分析されたマーク値の変化。



探索オプションを以下に示します。

- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。
- 極値がある場合とない場合の違いを調べます。
- 作成者は、ビューを新しいシートとして開き、フィルターを適用して極値を除外できます。



さらなる分析のための手順を以下に示します。

この例では、483 という極値を除外する

- 極値が他のレコード値よりも大幅に高いまたは低い場合は、極値を除外すると、分析されたマークの値がどのように変化するか確認してください。
- 極値がある場合とない場合のデータを検討するときは、そのデータについての実用面での知識を活用する必要があります。

と、分析されたマークは、ビュー内の他のマークと比較してとくに高くないことがわかります。他のマークが目立つようになりました。自転車のレンタル時間が他のいくつかの場所で長く、作成者はその理由を確認しようとして、他のマークを調べたいと考えるでしょう。

Null 値

Null 値の説明タイプは、想定よりも多くの欠落データがマーク内にあることを示しています。これは、Null であるターゲットのメジャー バリューの割合と、Null 値がそのメジャーの集計値にどのように寄与しているかを示しすものです。

この説明では以下を示します。

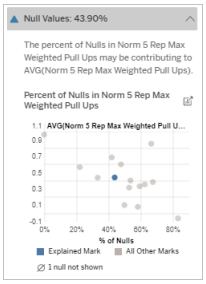
- 分析されたマーク(青色の円)のターゲット メジャーの Null 値の割合

探索オプションを以下に示します。

- 散布図の各円にカーソルを合わせると、詳細を表示できます。
- スクロールすると、グラフをさらに表示できます。
- **[開く]** アイコンを選択すると、Viz の拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- 必要に応じて、マークの Null 値を除外し、分析を進めます。



この例では、ターゲット メジャーの Null 値の割合が青色の円で示されています。

レコード数

この説明タイプでは、参照元レコードの数が合計に影響を与えている場合を説明します。この分析では、マークに集計されているレコードの数と、マークの実際の値との関係を確認します。

当たり前に思えるかもしれませんが、この説明タイプを利用すると、分析されたマークの値が、レコードの値に影響を受けているのか、レコードの数に影響を受けているのかを確認できます。

この説明では以下を示します。

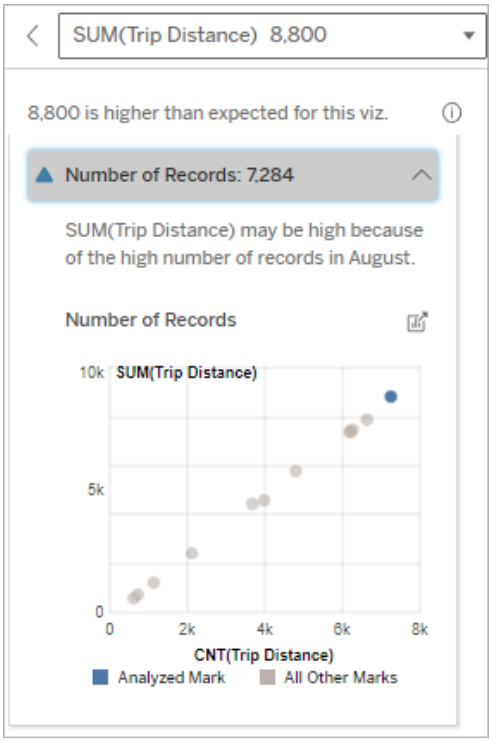
- 分析されたマーク(濃い青色のバー)のターゲットメジャーのレコード数。
- ソースのビジュアライゼーションの他のマーク(薄い水色のバー)のターゲットメジャーのレコード数。

探索オプションを以下に示します。

- 各バーにカーソルを合わせると、詳細を表示できます。
- スクロールすると、グラフをさらに表示できます。
- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- 個々のレコードの値が低いか高いか、または分析されたマークのレコード数が少ないか多いかを比較します。
- レコードの数が多すぎるとわかった場合は、作成者はデータを正規化する必要があるかもしれません。



この例では、元のビジュアライゼーションのディメンションである乗車月ごとに、走行距離のレコード数が一覧表示されています。8月は総走行距離の値が最も高くなっています。

8月に走行距離の値が最も高くなっているのは、8月には多くの人が乗車したからか、それとも一部の人が長く走行したから

かを確認することができます。

マークの平均値

この説明タイプでは、メジャーの平均が合計に影響を与えている場合を説明します。平均値が低いか高いか、またはレコード数が少ないか多いかを比較できます。

この説明では以下を示します。

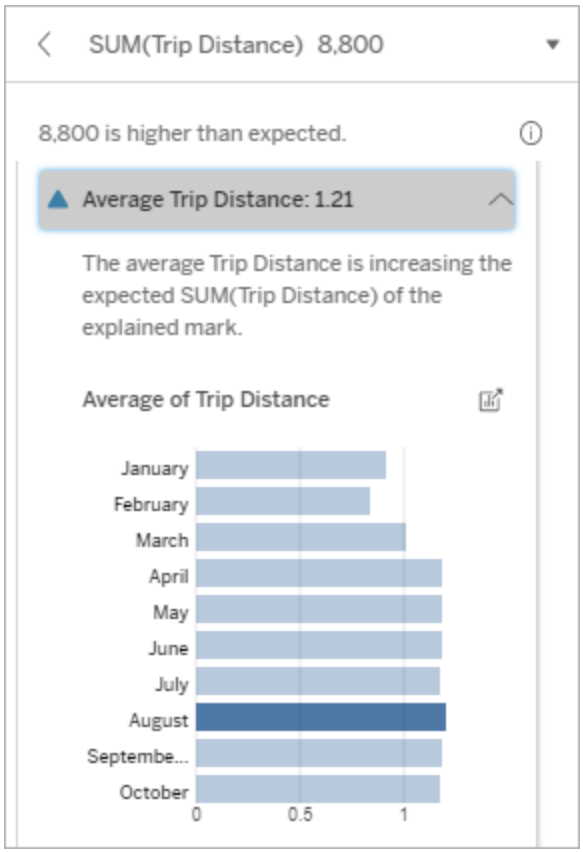
- ソースのビジュアライゼーションで使用しているディメンションの各値のターゲットメジャーの平均。

探索オプションを以下に示します。

- 各バーにカーソルを合わせると、詳細を表示できます。
- スクロールすると、グラフをさらに表示できます。
- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- 平均値が低いか高いか、またはレコード数が少ないか多いかを比較できます。たとえば、利益が大きかったのは、たくさんの商品を販売したからか、または高価な商品を販売したからか。
- 分析されたマークの平均値が大幅に高いまたは低い理由を確認します。



この例では、8月の平均走行距離は、ほかの月と比べて大幅に高くも低くもありません。これは、8月は乗車時間が長い人がいたからではなく、乗車回数が多かったため、8月の総走行距離が長かったことを示しています。

関連する単一の値

この説明を使用すると、分析されたマークを構成するレコード値の構成を確認できます。

この説明タイプでは、視覚化されていないディメンションの単一の値が、分析されたマークの集計値に影響している可能性がある場合を説明します。視覚化されていないディメンションとは、データソース内に存在するが、現在ビューでは使用されていないディメンションを指します。

この説明では、ディメンションの参照元レコードがすべて同じ値である場合や、分析されたマークに対して多くのレコードまたは少数のレコードが同じ値であるためにディメンション値が目立つ場合について示しています。

注: 説明で使用する一般的な用語の定義については、「説明での用語と概念」を参照してください。

この説明では以下を示します。

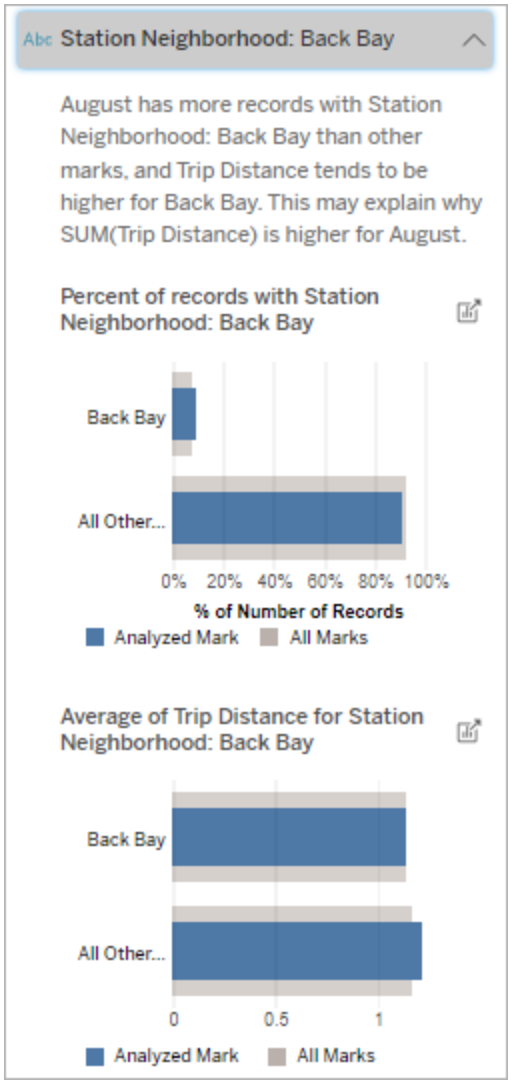
- ソースのビジュアライゼーションのすべてのマーク(灰色のバー)に対する分析されたマーク(青いバー)について、ディメンションの単一値のレコード数のパーセント。
- ソースのビジュアライゼーションのすべてのマーク(灰色のバー)に対する分析されたマーク(青いバー)について、ディメンションの他のすべての値のレコード数のパーセント。
- 分析されたマーク(青いバー)とすべてのマーク(灰色のバー)について、ディメンションの単一値のターゲットメジャーの平均。
- ソースのビジュアライゼーションのすべてのマーク(灰色のバー)に対する分析されたマーク(青いバー)について、ディメンションの他のすべての値のターゲットメジャーの平均。

探索オプションを以下に示します。

- 各バーにカーソルを合わせると、詳細を表示できます。
- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- この説明を使用すると、分析されたマークを構成するレコード値の構成を確認できます。
- この説明で明らかになった視覚化されていないディメンションを分析する



この例では、統計分析により、多くの乗車がバック湾の駅周辺から行われていることが明らかになりました。「駅周辺」は視覚化されていないディメンションであり、ビジュアライゼーションの参照元データの走行距離と何らかの関係があることがわかります。

ために、作成者は新しいビジュアライゼーションを作成したいと思うかもしれません。

上位の要因

この説明を使って、分析されたマークの中で最も大きな割合を占める値を確認してください。

COUNT 集計の場合、上位の要因には最も多くのレコードを持つディメンジョン値が表示されます。SUM の場合、この説明には、部分和が最も大きいディメンションの値が表示されます。

関連するディメンション

この説明を使用すると、分析されたマークを構成するレコード値の構成を確認できます。

この説明タイプでは、視覚化されていないディメンションの分布が、分析されたマークの集計値に影響している可能性があることを説明します。この説明タイプは、ターゲットメジャーの合計、カウント、平均に使用します。視覚化されていないディメンションとは、データソース内に存在するが、現在ビューでは使用されていないディメンションを指します。

注: 説明で使用する一般的な用語の定義については、「説明での用語と概念」を参照してください。

この説明では以下を示します。

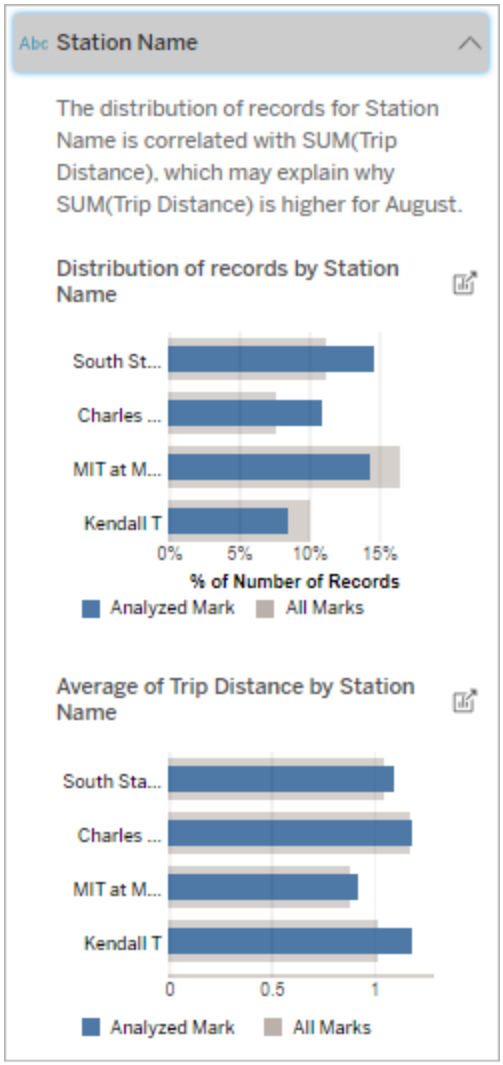
- ソースのビジュアライゼーションのすべてのマーク(灰色のバー)のディメンションのすべての値に対する、分析されたマーク(青いバー)のディメンションのすべての値のレコード数のパーセント。
- すべてのマーク(灰色のバー)のディメンションのすべての値に対する、分析されたマーク(青いバー)のディメンションのすべての値のターゲットメジャーの平均。

探索オプションを以下に示します。

- 各バーにカーソルを合わせると、詳細を表示できます。
- スクロールすると、グラフをさらに表示できます。
- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- この説明を使用すると、分析されたマークを構成するレコード値の構成を確認できます。
- この説明で明らかになった視覚化されていないディメンションを分析するために、作成者は新しいビジュアライゼーションを作成したいと思うかもしれません。



この例では、統計分析により、全体的な乗車のマークと比較して、サウス駅とMITからの乗車が多く、チャールズサークル駅とケンダル駅からの乗車が少ないことが明らかになりました。

「駅名」は視覚化されていないディメンションであり、ビジュアライゼーションの参照元データの走行距離と何らかの関係がある

ことがわかります。

関連するメジャー

この説明タイプでは、視覚化されていないメジャーの平均が、分析されたマークの集計値に影響している可能性があることを説明します。視覚化されていないディメンションとは、データソース内に存在するが、ビューでは現在使用していないディメンションを指します。

この説明により、視覚化されていないメジャーとターゲットメジャーの間の線形関係または二次関係が明らかになります。

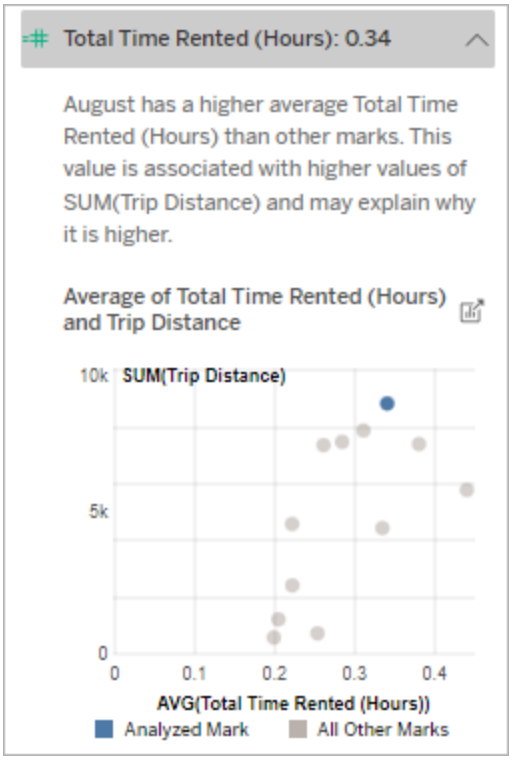
> **注**: 説明で使用する一般的な用語の定義については、「説明での用語と概念」を参照してください。

この説明では以下を示します。

- ビュー内の分析されたマーク(青い円)とすべてのマーク(灰色の円)について、ターゲットメジャーの合計と視覚化されていないメジャーの平均との関係。
- 視覚化されていないメジャーの平均値が高いまたは低いために、ターゲットメジャーの合計が高いまたは低い場合。

探索オプションを以下に示します。

- 各円にカーソルを合わせると、詳細を表示できます。
- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。



さらなる分析のための手順を以下に示します。

- この説明で明らかになった視覚化されていないメジャーを分析するために、作成者は新しいビジュアライゼーションを作成したいと思うかもしれません。

この例では、走行距離が長い理由の1つとして、平均合計レンタル時間が長いことが考えられます。

その他の探索項目

このセクションには、分析されたマークが一意的または異常であると考えられる理由が示されています。これらの説明については、以下の点に注意してください。

- このマークがなぜその値であるかは説明していません。
- ソースの視覚化におけるメジャーの値とは関係がありません。
- ターゲットメジャーを考慮していません。

その他の興味深いディメンション

この説明を使用すると、分析されたマークを構成するレコード値の構成を確認できます。

分析されたマーク内の視覚化されていないディメンションの分布は、ビュー内の他のすべてのマークの値の分布と比較して異常です。視覚化されていないディメンションとは、データソース内に存在するが、現在ビューでは使用されていないディメンションを指します。

注: 説明で使用する一般的な用語の定義については、「説明での用語と概念」を参照してください。

この説明では以下を示します。

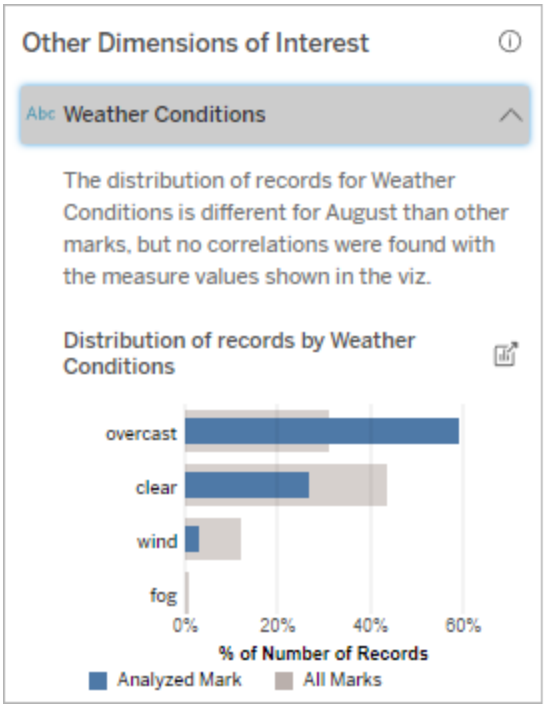
- ソースのビジュアライゼーションのすべてのマーク(灰色のバー)のディメンションのすべての値に対する、分析されたマーク(青いバー)のディメンションのすべての値のレコード数のパーセント。

探索オプションを以下に示します。

- 各バーにカーソルを合わせると、詳細を表示できます。
- スクロールすると、グラフをさらに表示できます。
- **[開く]** アイコンを選択すると、Vizの拡大版を表示できます。

さらなる分析のための手順を以下に示します。

- この説明を使用すると、分析されたマークを構成するレコード値の構成を確認できます。
- この説明で明らかになった視覚化されていないディメンションを分析するために、作成者は新しいビジュアライゼーションを作成したいと思うかもしれません。

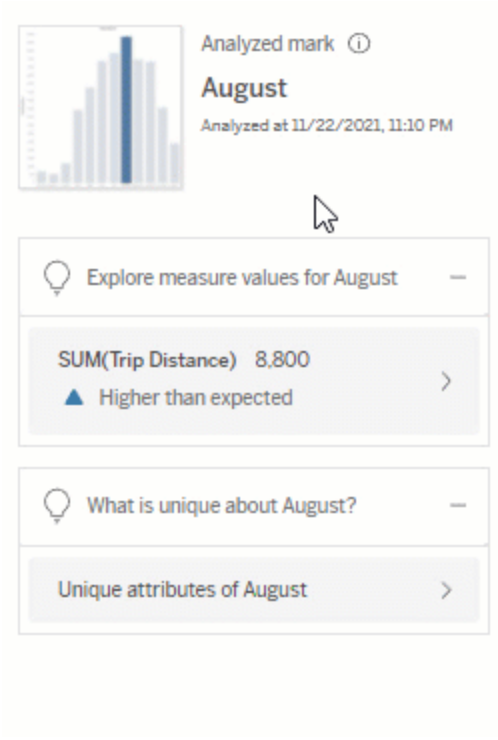


この例では、高い割合のレコードが曇りの天気に関連付けられています。データはボストンでのレンタサイクルに関するものであり、分析されたマークは 8 月の走行距離であるため、通常、天気は暖かくて湿度が高いと推測できます。暑さを避けるために、曇りの日には自転車がより頻繁に借りられたかもしれません。8 月は曇りの日が多かった可能性もあります。

「データの説明を見る」の分析済みフィールド

「データの説明を見る」は、ダッシュボードやシート上で統計分析を行い、異常値となっているマークや、選択した具体的なマークを検索します。また、現在のビューには表示されていない、データソースの関連するデータポイントの可能性も考慮されます。

「データの説明を見る」の分析では、データソースのすべての列が含まれるわけではありません。多くの場合、特定の種類のフィールドは自動的に分析から除外されます。詳細については、既定で除外されるフィールドを参照してください。

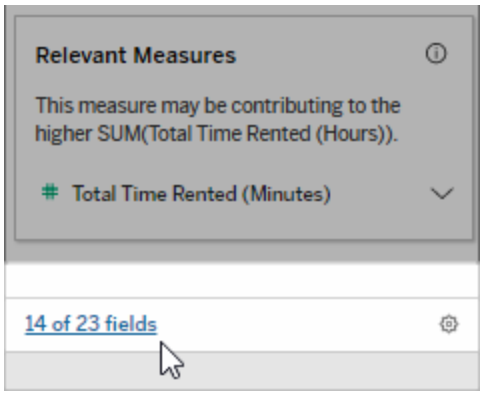


注: 500 を超える一意の値を持つディメンションは、分析の対象とは見なされません (「データの説明を見る」の設定で作成者によって許可されている場合を除く)。

すべてのユーザーは、現在の分析に含まれるフィールドまたは除外されるフィールドに関する情報を表示できます。編集パーミッションを持つ Creator と Exploror は、「データの説明を見る」で使用するフィールドを編集して、統計分析に使用することができます。

「データの説明を見る」によって分析されたフィールドを表示する

マークの値に寄与するメジャーの説明を展開すると、分析で考慮されるフィールド数を示すリンクが [データガイド] ペインの下部に表示されます。

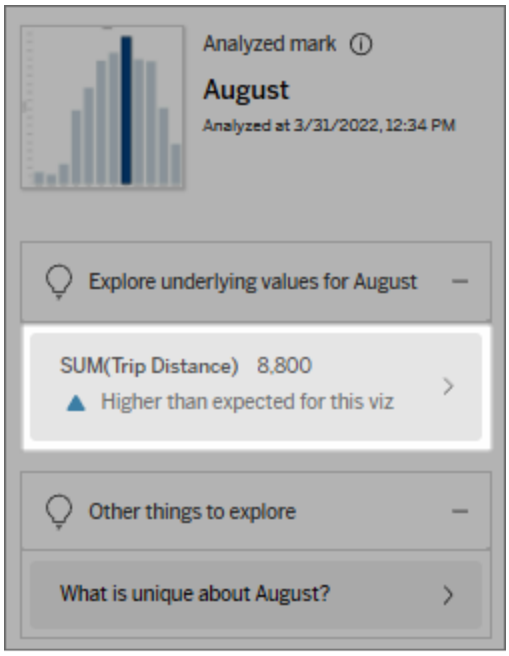


リンクをクリックして、現在の統計分析に含まれるフィールドのリスト、または現在の統計分析から除外されるフィールドのリストを表示します。

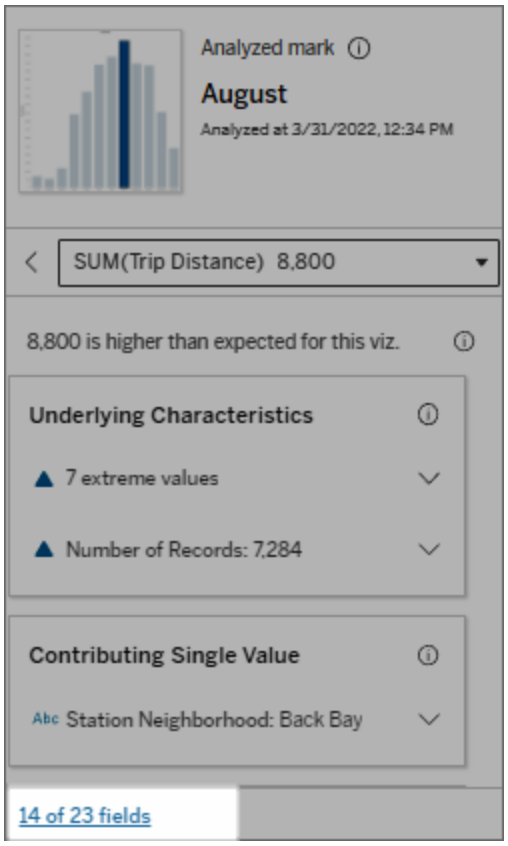
データソースに含まれるディメンションやメジャーのうちビジュアライズされていないものが 1,000 個を超える場合、「データの説明を見る」でさらにフィールドを考慮するかどうかを尋ねるアラートが表示されることがあります。**[すべて説明]** をクリックして、フィールドをさらに含めて分析を実行します。分析が完了するまで時間がかかる場合があります。

「データの説明を見る」が統計分析に使用するフィールドを表示するには

1. ダッシュボード、シート、マーク上で「データの説明を見る」を実行します。
2. [データガイド] ペインの **[値への寄与]** で、メジャー名をクリックします。

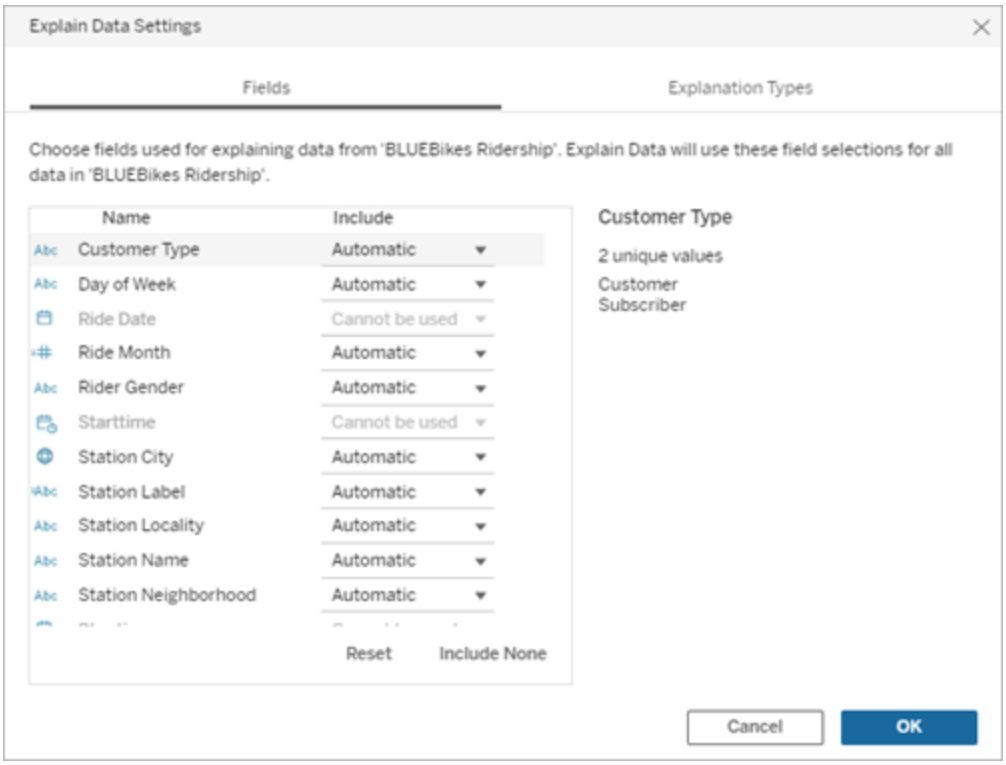


3. ペインの下部にある *[number-of-fields (フィールド数)]* リンクをクリックします。



統計分析に使用されるフィールドを変更する

編集パーミッションを持つCreatorとExplorerは、[「データの説明を見る」] ダイアログ ボックスの [フィールド] タブで、統計分析に含めるフィールドまたは統計分析から除外するフィールドを選択できます。

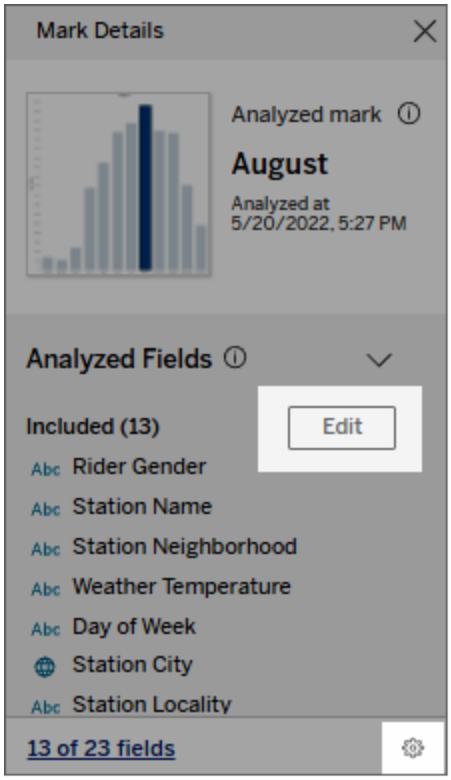


データ ソース内 のディメンション に一意 の値 が多数 含まれる場合 (500 個まで)、それらのフィールドは分析対象とは見なされません。

「データの説明を見る」が統計分析に使用するフィールドを編集するには

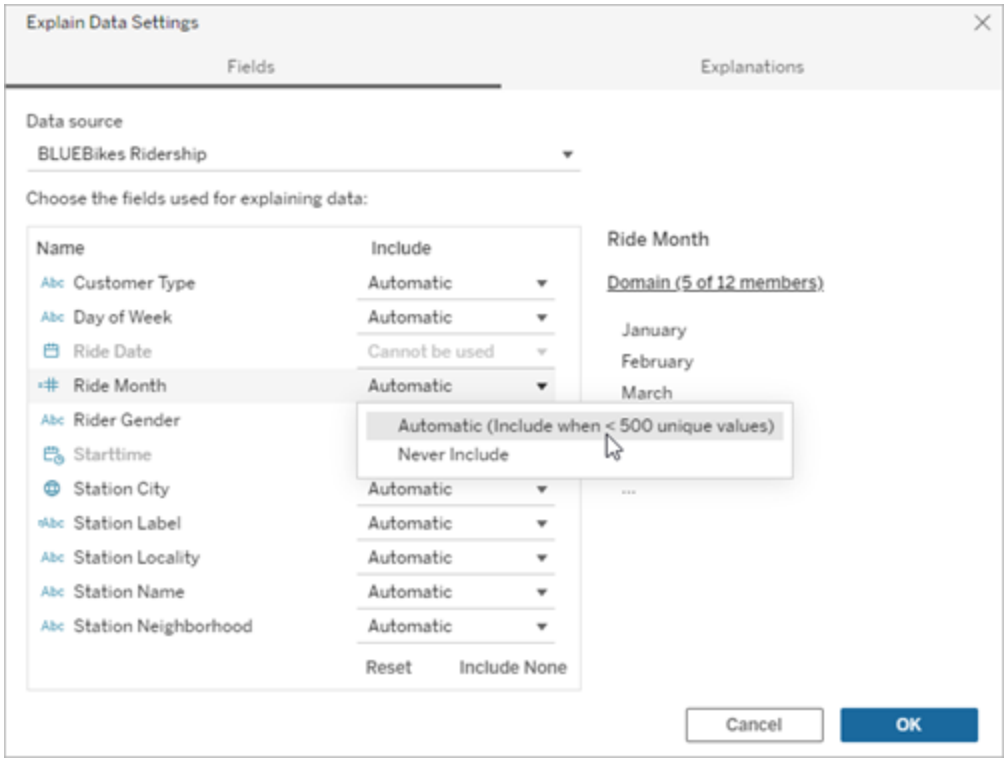
分析済みフィールドの設定は、データ ソース レベルで適用されます。

1. ビューを編集するときに、マーク上で「データの説明を見る」を実行します。
2. [データ ガイド] ペインで、ペインの下部にある設定のアイコンをクリックします。または、[Analyzed Fields (分析済みフィールド)] ビュー (分析済みフィールドを開く方法) の **[編集]** ボタンをクリックします。



3. [「データの説明を見る」の設定] ダイアログ ボックスで **[フィールド]** タブをクリックします。
4. フィールド名の横にあるドロップダウン矢印をクリックし、**[Automatic (自動)]** または **[Never Include (含めない)]** を選択して、**[OK]** をクリックします。

   分析に含めるには、フィールドの一意の値が 500 個未満である必要があります。



既定で除外されるフィールド

既定で除外されるフィールド	除外の理由
データソースに 1,000 を超えるメジャーがあるときの、すべてのビジュアライズされていないメジャー。	1,000 個を超えるビジュアライズされていないメジャーやディメンションの説明を計算すると、計算に時間がかかり、場合によっては数分かかることがあります。これらのフィールドは初期分析では規定により除外されますが、さらに分析するためにそれらを含めるよう選択することができます。
データソースに 1,000 を超えるディメンションがあるときの、すべてのビジュアライズされていないディメンション。	このような場合は、「データの説明を見る」で複数のフィールドを考慮するかどうかを確認するアラートが表示されることがあります。詳細を表示するには、アラートのリンクをクリックします。**[すべて説明]** をクリックして、フィールドをさらに含めて分析を実行します。

既定で除外されるフィールド	除外の理由
ジオメトリ、緯度、または経度を使用するフィールド	ジオメトリ、緯度、経度だけでは説明にはなりえません。緯度や経度を説明として呼び出す説明は、偽の相関関係によるものであり、確実な説明ではない可能性が高いです。
濃度の高いディメンション (500 人を超えるメンバーを持つディメンション)	高濃度のディメンションは、計算に時間がかかります。ディメンションに 500 個を超える一意の値が含まれる場合は、分析対象として見なされません。
グループ、ビン、またはセット	現在サポートされていません。
表計算	表計算がビューとは異なる詳細レベルにある場合は、表計算を分析することはできません。
平均化できないビジュアライズされていないメジャー	平均化できないビジュアライズされていないメジャーには、計算式に集計が含まれる計算フィールドであるメジャーが含まれます (シートに追加するとAGG() フィールドとして表示されます)。
非連続メジャーと連続ディメンション	現在サポートされていません。
非表示のフィールド	利用できません。
エラーのある計算フィールド	分析する値がありません。

「データの説明を見る」を使用する場合の要件と考慮事項

Tableau Desktop の作成者は、いつでも「データの説明を見る」を使用できます。

Tableau Cloud および Tableau Server では、「データの説明を見る」がサイトに対して有効になっている場合、適切なパーミッションを持つ Creator と Explorer は、ワークブックを編集する際に「データの説明を見る」を実行できます。適切な権限を持つすべてのユーザーは、パブリッシュされたワークブックの表示モードで「データの説明を見る」を実行できます。詳細については、「データの説明を見る」へのアクセスの制御を参照してください。

「データの説明を見る」の使用に最適な Viz とは

「データの説明を見る」は、要約されたデータを伝達するインフォグラフィックスタイルの説明的な Viz ではなく、より深い調査と分析を必要とするビジュアライゼーションで使用するのが最も適しています。

- 「データの説明を見る」がデータのモデルを作成し、説明を生成するには、行レベルのデータが必要です。参照元データの行レベルのデータを含む Viz で、視覚化されていないフィールドに関係が存在する可能性がある場合は、「データの説明を見る」を実行する絶好の機会です。
- 事前に集計されたデータに基づくViz で、行レベルのデータにアクセスできない場合は、「データの説明を見る」によって実行される統計分析には適していません。

「データの説明を見る」に最適なデータ

ワークシートで「データの説明を見る」を使用する場合は、「データの説明を見る」は以下と連携することを覚えておいてください。

- **単一のマークのみ**:「データの説明を見る」は単一のマークを分析します。複数のマークの分析はサポートされていません。
- **集計データ**: ビューには、SUM、AVG、COUNT、または COUNTD を使用して集計された 1 つ以上のメジャーを含める必要があります。少なくとも 1 つのディメンションがビューに存在する必要があります。
- **単一のデータ ソースのみ**: データは単一のプライマリデータ ソースから抽出する必要があります。「データの説明を見る」は、ブレンドされたデータ ソース、またはキューブ データ ソースでは機能しません。

ワークブックのデータ ソースを準備する際、分析時に「データの説明を見る」を使用する場合は、次の点に注意してください。

- 十分に広域な参照元データを含むデータ ソースを使用します。理想的なデータセットには、説明を行う1 つまたは複数の集計メジャーに加えて、10 ~ 20 以上の列が含まれています。
- 列 (フィールド) にわかりやすい名前を付けます。

- 冗長な列とデータ準備アーティファクトを排除します。詳細については、統計分析に使用されるフィールドを変更するを参照してください。
- データソース内の視覚化されていない列は破棄しないでください。「データの説明を見る」では、マークを分析するときに、参照元データのフィールドが考慮されます。
- カーディナリティディメンションが低い方が適切に機能します。カテゴリーディメンションの説明は、基数が高すぎない方が解釈しやすくなります (20 カテゴリー未満)。ディメンションに 500 個を超える一意の値が含まれる場合は、分析対象として見なされません。
- 原則として、データは事前に集計しないでください。ただし、データソースが大規模な場合は、データを適切な詳細レベルに事前に集約することを検討してください。
- ライブデータソースに対しては抽出を使用します。抽出は、ライブデータソースよりも速く実行されます。ライブデータソースを使用すると、説明を作成するプロセスで多数のクエリ (各候補の説明ごとに約 1 つずつ) が作成されるため、説明の生成により長い時間がかかってしまいます。

「データの説明を見る」が使用できない場合

データソースやビューの特性によっては、「データの説明を見る」を選択したマークで使用できない場合があります。「データの説明を見る」で選択したマークを分析できない場合、「データの説明を見る」アイコンとコンテキスト メニュー コマンドは使用できません。

「データの説明を見る」は、以下を使用するビューでは実行できません。	• マップの座標フィルター • ブレンドされたデータソース • パラメーターを含むデータソース • Access などの COUNTD 構文または COUNT (DISTINCT...) 構文をサポートしないデータソース。 • 集計メジャーのフィルター • 非集計メジャー
「データの説明を見る」は、以下を選択した場合は実行できません。	• 複数のマーク • 軸 • 凡例 • 総計 • 傾向線またはリファレンス ライン • 非常に少ない数のマークを含むビュー内のマーク
「データの説明を見る」は、以下の	• SUM、AVG、COUNT、COUNTD を使用して集

説明に使用されるメジャーでは実行できません。	計されません • 表計算です • メジャーバリューで使用されます
「データの説明を見る」は、以下の場合にディメンションの説明を提供できません。	• 計算フィールド • パラメーター • メジャーネームとメジャーバリューで使用される場合 • 500 個を超える一意の値が含まれるフィールド。ディメンションに 500 個を超える一意の値が含まれる場合は、分析対象として見なされません。

「データの説明を見る」へのアクセスの制御

「データの説明を見る」へのアクセスは、サイトロールととコンテンツパーミッションによって異なります。Tableau Desktop の作成者は、いつでも「データの説明を見る」を使用できます。適切なパーミッションを持つ作成者は、Tableau Cloud および Tableau Server の編集モードで「データの説明を見る」を実行できます。

また、作成者は、パブリッシュされたワークブックの表示モードで「データの説明を見る」を表示するかどうか、およびどの説明タイプを表示するかを制御できます。

「データの説明を見る」では、ビューに表示されていないデータソースのディメンションとメジャーから値を表示することもできます。作成者は、「データの説明を見る」を実行し、結果として表示される説明を確認して、公開されたワークブックに機密データが公開されていないことをチェックする必要があります。

「データの説明を見る」にアクセスできるユーザー

デフォルトでは、「データの説明を見る」はサイトレベルで有効になっています。サーバー管理者 (Tableau Server) とサイト管理者 (Tableau Cloud) は、サイトで「データの説明を見る」を使用できるかどうかを制御できます。詳細については、サイトに対する「データの説明を見る」の無効化または有効化を参照してください。

モード	アクセスできるユーザー
表示モード	Tableau の **Viewer**、**Explorer**、**Creator** ([Run Explain Data (「データの説明を見る」を実行)] パーミッション機能を持つ) は、「データの説明を見る」の説明を表示モードで実行および探索できます。
編集モード	Tableau の **Creator** は、Tableau Desktop、Tableau Cloud、Tableau Server でビューを編集するときに「データの説明を見る」を実行できます。[Run Explain Data (「データの説明を見る」を実行)] パーミッション機能と編集権限を持つ **Explorer** は、Tableau Cloud または Tableau Server でワークブックを編集するときに「データの説明を見る」を実行できます。 Creator と編集権限を持つ Explorer は、新しいワークシートを開いてさらに分析することができます。 また、「データの説明を見る」の設定を使用して、「データの説明を見る」を使用できるユーザーと表示内容を制御することもできます。

「データの説明を見る」を使用できるユーザーとユーザーが表示できる内容の制御

Tableau Cloud および Tableau Server の編集モードと表示モードで「データの説明を見る」を使用できるようにするには、設定の組み合わせを有効にする必要があります。

編集モード

作成者が編集モードで「データの説明を見る」を実行したり、「データの説明を見る」の設定を編集したりするための要件は次のとおりです。

- サイト設定: **[「データの説明を見る」の可用性]** が **[有効]** に設定されている。デフォルトで有効に設定されています。
- サイトロール: Creator または Explorer (パブリッシュ可能)。
- パーミッション: **[Run Explain Data (「データの説明を見る」を実行)]** が **[許可]** に設定されている。既定では指定されていません。このパーミッションを使用したワークブック (Tableau

バージョン2022.1以前)をTableauバージョン2022.2以降で開く場合は、[Run Explain Data (「データの説明を見る」を実行)]を[許可]にリセットする必要があります。

注: Creatorまたは Explorer (パブリッシュ可能)の**[すべてのデータのダウンロード]**機能は、極値の説明の[フルデータの表示]オプションを表示するかどうかを制御します。Viewerの場合、[すべてのデータのダウンロード]機能が常に拒否されます。ただし、「データの説明を見る」の設定で極値の説明のタイプが有効になっている場合は、すべてのユーザーがレコードレベルの詳細を表示できます。

編集権限と[Run Explain Data (「データの説明を見る」を実行)]パーミッション機能を持つCreatorとExplorerは、以下を制御するためのオプションを提供する**[「データの説明を見る」の設定]**にアクセスできます。

- [データガイド]ペインに表示される説明タイプ
- 統計分析に含めるフィールドと除外するフィールド。

これらのオプションはワークブック全体に対する設定であり、[「データの説明を見る」の説明]ダイアログボックスでのみ設定できます。

表示モード

すべてのユーザーが「データの説明を見る」を表示モードで実行するための要件は次のとおりです。

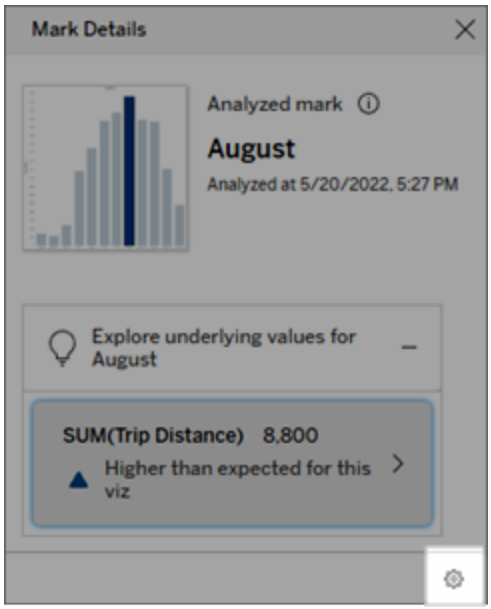
- サイト設定: **[「データの説明を見る」の可用性]**が**[有効]**に設定されている。デフォルトで有効に設定されています。
- サイトロール: Creator、Explorer、または Viewer。
- パーミッション: **[Run Explain Data (「データの説明を見る」を実行)]**が**[許可]**に設定されている。既定では指定されていません。このパーミッションを使用したワークブック(Tableauバージョン2022.1以前)をTableauバージョン2022.2以降で開く場合は、[Run Explain Data (「データの説明を見る」を実行)]を[許可]にリセットする必要があります。

注: データガイドで検出された外れ値の説明を表示するには、Vizのユーザーは、ワークブックまたはビューに対して許可された「データの説明を見る」のパーミッションを持っている必要があります。ワークブックの所有者は、Tableau ServerまたはTableau Cloudでこのワークブックの

パーミッション設定を開き、そのユーザーに「データの説明を見る」のパーミッションを許可する必要があります。

「データの説明を見る」の説明のダイアログ ボックスを開く

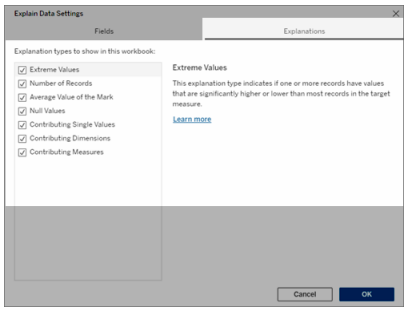
1. **[分析]** メニューから、**[「データの説明を見る」の設定]** を選択します。または、[データ ガイド] ペインで、設定のアイコン (右下) をクリックします。



「データの説明を見る」の表示に説明タイプを含めるか除外する

Creator と編集権限を持つ Explorer は、すべてのワークブック ユーザーに表示する説明タイプを、除外するか含めるか選択できます。

1. [「データの説明を見る」の設定] ダイアログ ボックスで、**[説明のタイプ]** タブをクリックします。



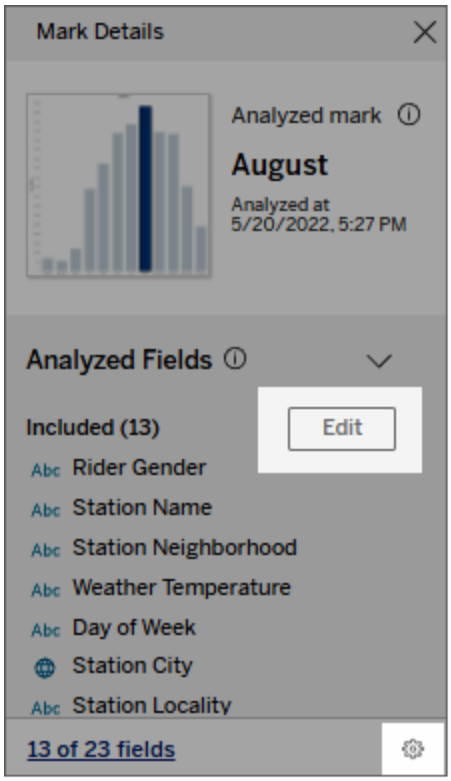
2. 説明タイプのリストで、説明タイプを選択またはクリアします。
3. **[OK]** をクリックします。

公開されたワークブックを保存して閉じ、ワークブックから表示モードでビューを開くことにより、設定をテストします。通常、極値の説明があるマークを選択し、「データの説明を見る」を実行して説明の結果を確認します。

統計分析に使用するフィールドを含めるか除外する

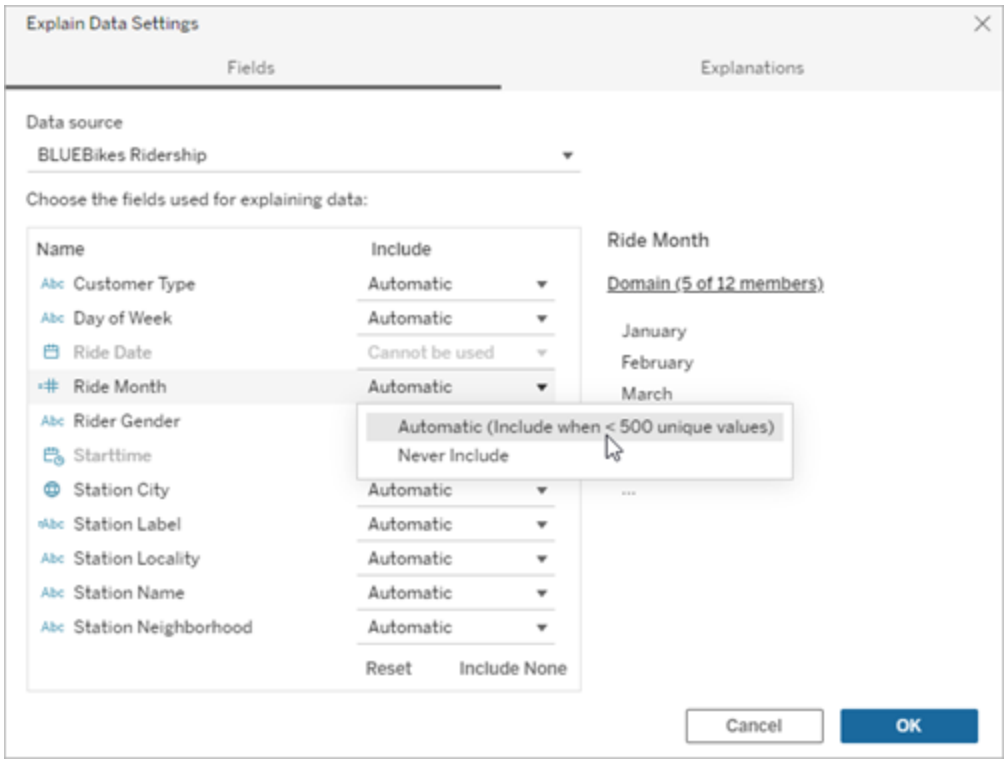
Creator と編集権限を持つ Explorer は、分析の対象とするフィールドを、除外するか含めるか選択できます。

1. [データガイド] ペイン (右下) で、設定のアイコンをクリックします。または、[分析フィールドビュー] の **[編集]** ボタンをクリックします。



2. [「データの説明を見る」の設定] ダイアログ ボックスで、**[フィールド]** タブをクリックします。
3. **[含める]** の下のフィールドのリストで、ドロップダウン矢印をクリックして **[自動]** を選択し、そのワークブックで「データの説明を見る」が実行されるたびに適格なフィールドが含まれるようにします。

   分析に含めるには、フィールドの一意の値が 500 個未満である必要があります。



特定のフィールドを明示的に除外するには、**[含めない]** を選択します。
フィールドを考慮せずにデータの統計分析を実行するには、**[含めるものがない]**を選択します。
デフォルト設定に戻すには、**[リセット]**を選択します。

4. **[OK]** をクリックします。

公開されたワークブックを保存して設定をテストします。マークを選択して「データの説明を見る」を実行し、説明の結果を確認します。

ユーザーがメールや Slack を介して説明を共有できるように Tableau を構成する

Tableau 管理者は、メールまたは Slack を介して説明を表示モードで他の Tableau ユーザーと共有できるかどうかを制御できます。

次の手順に従って、Tableau Cloud または Tableau Server でメールおよび Slack を介して通知や共有を行うことを許可します。

1. **[設定]** をクリックします。
2. [全般] タブで、**[通知の管理]** セクションまでスクロールします。
3. コラボレーションの場合、**[Tableau 上]**、**[メール]**、および **[Slack]** を介した **[共有]** を選択します。

Slack を介して説明を共有するには、Tableau アプリが Slack ワークスペースに設定されている必要があります。Tableau Cloud では、Slack を介した説明の共有が既定で有効になっています。

Tableau Server では、管理者が Tableau Slack アプリを設定する必要があります。詳細については、「Tableau と Slack ワークスペースの統合」を参照してください。

「データの説明を見る」のしくみ

「データの説明を見る」は、データをより詳細に探索するための増分出発点として使用します。データが生成する説明によって、ビュー内の分析済みマークを構成する値や関連する値など、さまざまな値を表示できるようになりますこれにより、データソース内のデータポイントの特性、および統計モデリングを使用したデータの関連付け方法 (相関性) について説明できるようになります。これらの説明では、データを調査したり次に探索する内容についての興味深いヒントを見つけたりするための別のツールが提供されます。

注: データの説明を見るは、データ内の関係を明らかにし、説明するツールです。関係の原因やデータの解釈方法は教えてくれません。**あなたこそがデータのエキスパートです。**さまざまなビューを使用してさらに深く掘り下げていこうと興味がわく特性を判断する鍵となるのは、ドメインに関するあなたの知識や直感です。

「データの説明を見る」のしくみと、分析を強化するために「データの説明を見る」を使用する方法に関する関連情報については、Tableau Conference のプレゼンテーションを参照してください。

- アナリストから統計学者へ: 実際の「データの説明を見る」(1 時間)
- 「データの説明を見る」の活用 (45 分)

「データの説明を見る」であるもの (および「データの説明を見る」ではないもの)

「データの説明を見る」とは次のものを指します。

- ドメインの専門知識を活用するツールとワークフロー。
- データの関係を明らかにし、次にどこを見るべきかを推奨するツール。

- データ分析を迅速化しより広範なユーザーがデータ分析にアクセスしやすくするためのツールとワークフロー。

「データの説明を見る」とは次のものを指しません。

- 統計テストツール。
- 仮説を証明または反証するツール。
- データの因果関係について、回答を与えたり、何かを教えるツール。

マークで「データの説明を見る」を実行するときは、次の点に注意してください。

- **データの形状、サイズ、およびカーディナリティを考慮する。**「データの説明を見る」は小さなデータセットで使用できますが、十分な幅と、モデルを作成するための十分なマーク(粒度)が含まれている必要があります。

- **因果関係を想定しない。**相関関係は因果関係ではありません。説明はデータのモデルに基づいていますが、因果関係を説明するものではありません。

  相関関係があるとは、いくつかのデータ変数の間 (たとえば A とB の間) に関係があることを意味します。データの関連性を見ただけで、A が原因で B になった、B が原因で A になった、またはもっと複雑な何かが実際に起きていることは断定できません。データのパターンは、それぞれの場合で全く同じであり、アルゴリズムによりそれぞれの場合の違いを示すことはできません。2 つの変数が一緒に変化すると思われるというだけで、必ずしも一方が他方の変化の原因であるとは限りません。3 つ目の要素がその両方の変化の原因の可能性もあれば、それが偶然で全く因果関係がない場合もあります。

  とはいえ、どういうことかを特定するのに役立つ、データに含まれていない外的知識がある場合があります。一般的な類の外的知識と言えば、実験でデータを収集した状況があります。コイン投げで B が選択されたことを知っているなら、A の違いの一貫したパターン(単なるランダム ノイズではない) は B が原因でなければいけません。これらの概念の詳細な説明については、Hal Varian の記事「Causal inference in economics and marketing」を参照してください。

説明の分析および評価方法

「データの説明を見る」は、ダッシュボードやシート上で統計分析を行い、異常値となっているマークや、選択した具体的なマークを検索します。また、現在のビューには表示されていない、データソースの関連するデータポイントの可能性も考慮されます。

「データの説明を見る」では、最初にビジュアライゼーションに存在するデータのみを使用してマークの値を予測します。次に、データソース内のデータ(現在のビューではない) が考慮され、モデルに追加されます。モデルによって、予測されるマーク値の範囲が決まりますが、これは予測値の 1 標準偏差以内です。

予測範囲とは

マークの予測値は、Viz の参照元データの予測範囲の中央値です。予測範囲は、分析されたマークに対して統計モデルが予測する 15 パーセンタイルから 85 パーセンタイルまでの値の範囲です。Tableau は、選択したマークに対して統計分析を実行するたびに、予測範囲を算出します。

考えられる説明は、統計モデルを使用して説明能力が評価されます。説明ごとに、期待値と実際の値とが比較されます。

値	説明
予測より高い / 予測より低い	期待値の概要で、マークが*予想よりも低いか予想よりも高い*と説明されている場合、集計済みのマーク値が、統計モデルがマークで予測している値の範囲外にあることを意味します。予測値の概要で、マークが*予測よりもわずかに低いか予測よりもわずかに高い*と説明されているか、*自然変動幅の範囲内にある*と説明されている場合は、集約されたマーク値は予測範囲内にあるが、中央値よりも低いか高いことを意味します。
予測値	マークに予測値がある場合、その値が、そのマークの統計モデルが予測している予測値の範囲内にあることを意味します。
ランダムな変動	分析されたマークのレコード数が少ない場合、「データの説明を見る」で使用できる十分なデータがないため、統計的に有意な説明が作成されない可能性がありますマークの値が予測範囲外の場合、この予期しない値がランダムな変動によるものなのか、参照元レコードの意

値	説明
	味のある違いによるものなのかを「データの説明を見る」で判断できません。
説明なし	分析されたマーク値が予想範囲外で、「データの説明を見る」で使用する統計モデルに適合しない場合、説明は生成されません。

分析に使用されるモデル

「データの説明を見る」はビュー内のデータに関するモデルを作成してマークの値を予測し、そのモデルを踏まえ、マークが予測より高いか低いかを判断します。次に、データソースからビューへの列の追加やレコードレベルの外れ値のフラグ付けといった追加の情報を潜在的な説明として検討します。潜在的な説明ごとに、「データの説明を見る」で新しいモデルを適合させ、新しい情報を踏まえてマークがどの程度想定外であったかを評価します。説明が必要となる変動性の大きさに対して複雑性 (データソースから追加される情報の量) をトレードオフすることで、説明が採点されます。わかりやすい説明とは、変化のある説明よりも簡単であることです。

説明のタイプ	評価
極値	極値は、視覚化されたマークのモデルに基づく、外れ値となる集計マークです。選択したマークは、レコード値がデータで期待値分布の端にある場合に極値を含むと見なされます。 極値は、極値がある場合とない場合の集計マークを比較することによって決定されます。値を削除してもマークがあまり減らなければ、高い得点が得られます。 マークに極値が含まれていても、ただちに、それらの値は外れ値である、または、ビューからそれらのレコードを除外する必要がある、という意味ではありません。選択すべき内容は分析に応じて異なります。説明は単にマーク内の興味深い極値を示しています。たとえば、バナナの価格が 10 セントではなく10 ドルであるレコードであれば、誤って入力された値だとわかるかもしれません。または、特定の営業担当者がある四

説明のタイプ	評価
	半期に優れた成績を収めていたことが明らかになるかもしれません。
レコード数	レコード数は、個数を集計する観点から集計の合計をモデル化して説明したものです。いっぽう、レコードの平均値は、平均を集計する観点からモデル化しています。モデルが合計をより適切に説明するほど、スコアは高くなります。 ここでは、数が多いまたは少ないか、平均が高いまたは低いことが原因で合計が興味深いかどうかについて説明します。
マークの平均値	このタイプの説明は、合計を表す集計マークで使用されます。集計する個数または平均の観点から、SUM(X) = COUNT(X) * AVG(X) の関係を踏まえて、そのマークが他のマークと一致しているかどうかを説明します。 ここでは、数が多いまたは少ないか、平均が高いまたは低いことが原因で合計が興味深いかどうかについて説明します。
関連するディメンション	この説明は、視覚化されていないディメンションのカテゴリ間の分析という観点から、分析されたマークのターゲットメジャーをモデル化します。この分析により、モデルの複雑さとマークの説明のバランスを取ることができます。 *視覚化されていないディメンション*とは、データソース内に存在するが、現在ビューでは使用されていないディメンションを指します。このタイプの説明は、合計、カウント、平均に使用されます。 非視覚化ディメンションのモデルは、説明する列の分類値に従ってマークを分割してから、ソースのビジュアライゼーションに含まれるすべてのデータポイントが含まれた値を使用してモデルを構築し、作成されます。行ごとに、モデルが各マークを形成する個々のコンポーネントの復元を試みます。分析では、視覚化されていないディメンションの値が不明なモデルを使用した場合と比べて、視覚化されていないディメンショ

説明のタイプ	評価
	ンに対応するコンポーネントをモデル化して合計すると、モデルがマークをより適切に予測するかどうかを示します。 集計ディメンションの説明では、マークの値が条件を指定せずにどの程度詳しく説明されているかを調べます。次に、潜在的な説明となる各列の値にモデルが条件を付けます。説明の列の分布に条件を付けることによって、より優れた予測が得られます。
関連するメジャー	この説明は、視覚化されていないメジャーの観点から、視覚化されたディメンション全体で平均を集計してマークをモデル化します。*視覚化されていないディメンション*とは、データソース内に存在するが、ビューでは現在使用していないディメンションを指します。 関連するメジャーの説明によって、視覚化されていないメジャーとターゲットメジャーの間の線形関係または二次関係が明らかになります。

サイトに対する「データの説明を見る」の無効化または有効化

既定で「データの説明を見る」はサイトに対して有効になっていますが、Tableau 管理者はそれを無効にすることができます。

1. **[全般]** サイト設定に移動します。
2. (Tableau Server のみ) **[Web 作成]** セクションで、**[Let users edit workbooks in their browser (ユーザーが自分のブラウザーでワークブックを編集するのを許可する)]** を選択します。
3. **[Availability of Explain Data (「データの説明を見る」の可用性)]** セクションで、次のオプションから選択します。
    - **有効にする**と、適切なパーミッションを持つ Creator と Explorer が「データの説明を見る」を編集モードで実行できるようになります。表示モードが有効になっている場合、適切な権限を持つすべてのユーザーが「データの説明を見る」を実行できます。

- **無効にする**と、すべてのユーザーが「データの説明を見る」を実行したり、ワークブックの「データの説明を見る」設定にアクセスしたりできなくなります。

4. Tableau Cloud および Tableau Server 2023.3 以降で「データの説明を見る」を使用するには

    - **[データガイドの可用性]** セクションで、**[表示]** を選択します。データガイドの詳細については、「データガイドでダッシュボードを探検する」を参照してください。

ダッシュボードの拡張機能の使用

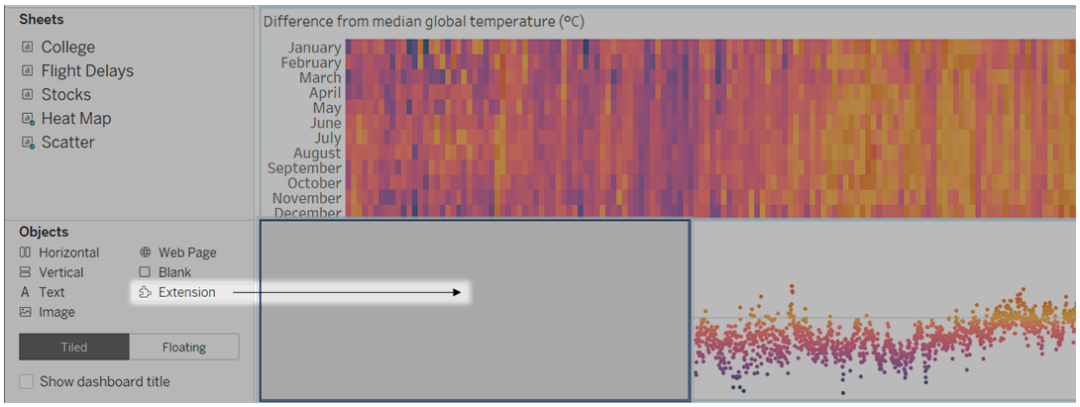
拡張により、固有の機能をダッシュボードに追加したり、Tableau 以外のアプリケーションに直接統合できます。拡張の追加は簡単です。他のダッシュボードオブジェクトと同じようにダッシュボードのレイアウトに組み込みます。

拡張によりダッシュボードの機能は、サードパーティー開発者により作成された Web アプリケーションで改善しました。開発者が独自の拡張機能を作成したい場合は、GitHub のドキュメンテーション「Tableau 拡張機能 API」を参照してください。

注: Tableau 管理者は、Tableau Desktop、Tableau Server、および Tableau Cloud のダッシュボードの拡張機能をオフにすることができます。

拡張機能をダッシュボードに追加する

1. Tableau ワークブックで、ダッシュボードシートを開きます。
2. **[オブジェクト]** セクションから、**[拡張機能]** をダッシュボードにドラッグします。



3. [Add an Extension (拡張機能の追加)] ダイアログ ボックスで、次のいずれかを実行します。

   - 拡張機能を検索して選択します。
   - **[Access Local Extensions (ローカルの拡張機能にアクセス)]** をクリックして、先ほどダウンロードした .trex ファイルを開きます。

4. プロンプトが表示されたら、ワークブックにあるデータへのダッシュボードの拡張機能のアクセスを許可または拒否します。詳細については、データ セキュリティ、Tableau の信頼できる拡張機能、ネットワーク対応拡張機能を参照してください。

   アクセスを許可する場合は、画面上の指示に従って拡張機能を設定します。

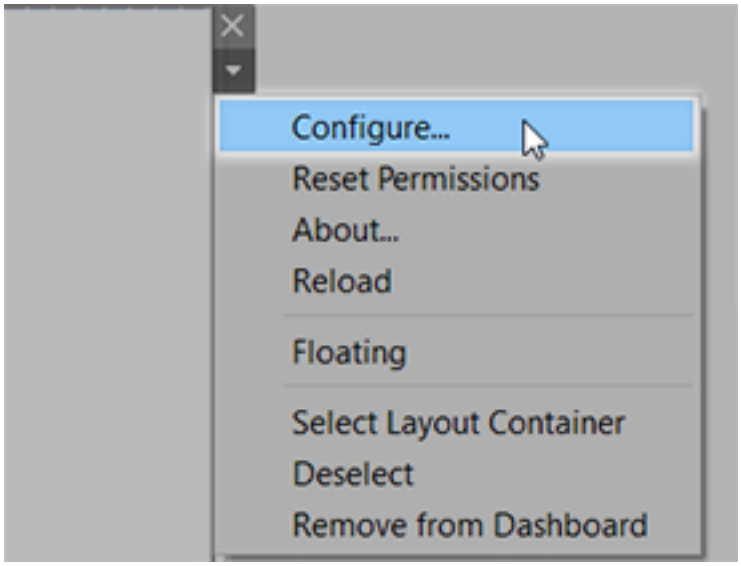
   > **注:** Tableau Server または Tableau Cloud を使用している場合、拡張機能オブジェクトは、印刷物、PDF、およびダッシュボードの画像 (登録メールの画像を含む) で空白表示されます。

ダッシュボードの拡張機能の構成

機能をカスタマイズできる構成オプションを提供するダッシュボードの拡張機能もあります。

1. 拡張機能をダッシュボードで選択し、右上隅のドロップダウン メニューから**[構成]**をクリックします。

2. 画面上の指示に従って拡張機能を構成します。



ダッシュボードの拡張機能の再読み込み

ダッシュボードの拡張機能が応答しなくなったら、再読み込みする必要があります。これは、ブラウザーでWeb ページを更新するのと似ています。

1. 拡張機能をダッシュボードで選択し、右上隅のドロップダウンメニューから**[再読み込み]**をクリックします。

   ダッシュボードの拡張機能が更新され、その元々の状態に設定されます。

2. 拡張機能を再読み込みしても使用できる状態に復元できない場合は、ダッシュボードから削除して再度追加してみてください。

データセキュリティ、Tableau の信頼できる拡張機能、ネットワーク対応拡張機能

拡張機能は、次の 2 つの標準形式で提供される Web アプリケーションです。

- *Tableau の信頼できる拡張機能* は、厳密に審査され、Tableau が管理するホスト上で展開されます。これらには、Tableau が開発および管理する拡張機能、Tableau Exchange

パートナーが作成し、Tableau の拡張機能審査およびホストサービスを通じて展開される拡張機能、および従来のサンドボックス拡張機能ホスト内の拡張機能が含まれます。

- *ネットワーク対応拡張機能*は、サードパーティーのホストに展開されています。これらの拡張機能を使用すると、開発者は Tableau を介さずに拡張機能の配信を管理できます。これらには、より優れた本番環境サポートを提供するためにパートナーが直接管理する拡張機能、ローカル ネットワーク内でホストおよび展開される拡張機能、および Tableau 外部のサードパーティー サービスとの通信を必要とする拡張機能が含まれます。

ネットワーク対応拡張機能を追加する前に、機能をホストする Web サイトが信頼できるサイトであることを確認します。

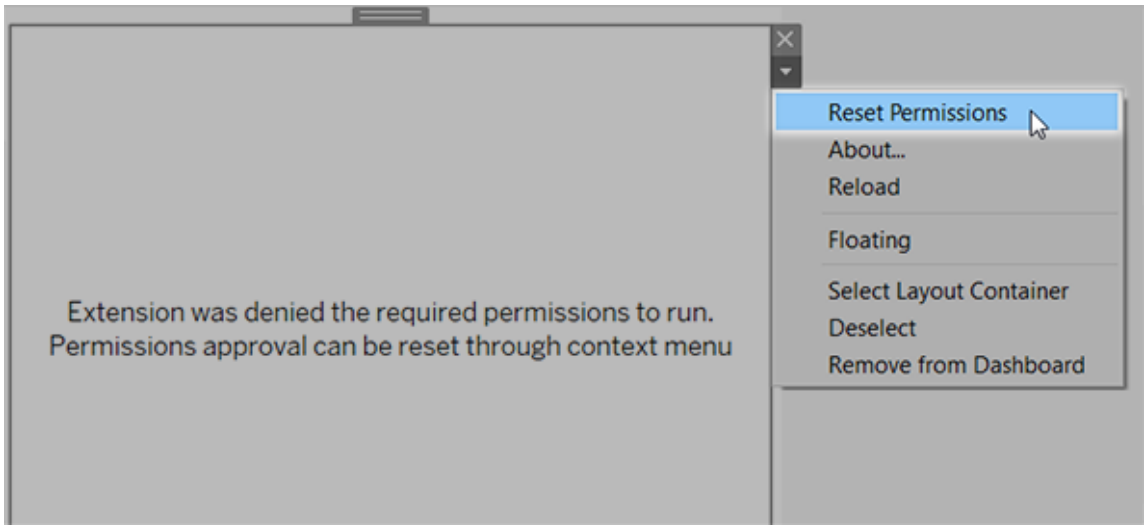
既定では、拡張機能は HTTPS プロトコルを使用します。これにより、データの送受信用の暗号化チャネルが保証され、ある程度のプライバシーとセキュリティが保証されます。ダッシュボードの拡張機能を使用する際のデータ セキュリティの詳細については、「拡張機能のセキュリティ- 展開に関するベストプラクティス」を参照してください。

ネットワーク対応拡張機能の許可または拒否

既定では、ネットワーク対応拡張機能を実行するには、パーミッションを要求する必要があります。このパーミッション ダイアログには、ホスト元のサードパーティー サーバーと、拡張機能に必要なデータ アクセス レベルに関する情報が表示されます。拡張機能の実行を許可する前に、この情報を確認することをお勧めします。

拡張機能は、通常、ビュー内の表示データにアクセスします。拡張機能が「フル データ アクセス」を要求する際に、ワークブック内のより深いデータへのアクセスを要求することがあります。これには、参照元データへのフル アクセス、データ ソースのテーブル名とフィールド名、データ ソース接続に関する情報が含まれます。ワークブックで拡張機能の実行を許可する前に、拡張機能のパーミッション ダイアログで宣言された情報を確認し、拡張機能の実行とこのデータへのアクセスを許可するか拒否するかを決定する必要があります。

フルデータ アクセスを必要とするがアクセスが拒否されている拡張機能を含むダッシュボードを表示している場合は、拡張機能の代わりにメッセージが表示されます。拡張機能を信頼しこれを使用したい場合は、パーミッションをリセットして拡張機能の実行を許可します。



1. ダッシュボードで拡張機能を選択し、右上隅のドロップダウン メニューから**[パーミッションのリセット]** をクリックします。
2. **[許可]** をクリックして拡張機能を実効しデータにアクセスするか、または **[拒否]** をクリックして拡張機能の実行を防ぎます。

Tableau Desktop で JavaScript が有効になっていることを確認する

ダッシュボードの拡張機能は JavaScript ライブラリである Tableau 拡張 API ライブラリを使用してデータを操作します。拡張機能を使用する場合は、ダッシュボードのセキュリティ設定で JavaScript が有効になっていることを確認します。

[ヘルプ]、**[設定とパフォーマンス]**、**[ダッシュボード Web 表示セキュリティの設定]**、**[JavaScript の有効化]** の順に選択します。

拡張機能が Tableau Cloud または Tableau Server で実行していることを確認する

拡張機能を、Tableau Desktop からパブリッシュするワークブックに追加したり、Tableau Cloud および Tableau Server の Web 作成モードに直接追加したりすることが可能です。Tableau 管理者は、拡張のサイトでの実行を可能にし、ネットワーク対応拡張を安全なリストに追加する必要があります。管理者は、テスト済みで信頼できる拡張だけを許容すべきです。

ダッシュボードの拡張機能を Tableau Cloud または Tableau Server で使用する場合は、管理者に「ダッシュボードの拡張機能を Tableau Cloud で管理」または「ダッシュボードの拡張機能を Tableau Server で管理」を参照するように指示します。

拡張機能にワークブックまたはビューにアクセスするパーミッションがあることを確認する

拡張機能を含むワークブックをパブリッシュする場合は、ビュー内の表示可能なデータに拡張機能がアクセスできることを確認してください。表示可能なデータは、サマリー データとも呼ばれます。Tableau Server または Tableau Cloud でこのアクセス権を付与するには、ワークブックまたはビューで **[サマリー データをダウンロード]** するパーミッションが許可されていることを確認します。拡張機能でデータがダウンロードされません。拡張機能には、ビュー内のデータを操作するためのパーミッションが必要です。

ワークブックをパブリッシュしたときにシートをタブとして表示するよう選択したかどうかに応じて、ワークブック レベルまたはビュー レベルのいずれかで **[サマリー データをダウンロード]** するパーミッションを設定します。シートがタブとして表示されない場合は、ビュー レベルでパーミッションを設定しています。

ワークブックまたはビューを Tableau Public にパブリッシュする場合、拡張機能には表示可能なデータへのアクセス権も必要です。パーミッションを付与するには、[設定] を開いて **[アクセスを許可]** を選択します。

サンドボックス拡張機能でサポートされる Web ブラウザー

サンドボックス拡張機能は、Internet Explorer 11 を除くすべてのブラウザーでサポートされている Tableau Server と Tableau Cloud で実行されます。

サンドボックス拡張機能でサポートされているバージョンの Tableau Server

サンドボックス拡張機能は、Tableau Server 2019.4 以降で使用できます。

サードパーティーおよびパートナーが構築したダッシュボードの拡張機能についてのサポートを受ける

パートナーおよびサードパーティーの拡張機能は、Tableau では直接サポートされていません。ただし、サポートが必要な場合は、詳しい情報を確認できる方法がいくつかあります。

- Tableau Exchange の一部のパートナーページには、**サポート**を直接受けることができるリンクが用意されています。
- Tableau Exchange の**開発者 Web サイトのリンク**から、コネクタの作成者に連絡できます。

拡張機能 API に関する質問については、DataDev Slack チャネルを通じて質問を共有してサポートを求めるか、拡張機能 API に関する問題を GitHub から送信することをお勧めします。

アニメーションの書式設定

ビジュアライゼーションをアニメーション化すると、データの変化パターンをより良く強調表示して、スパイクや外れ値を明らかにしたり、データポイントのクラスタリングや分離の様子を確認したりできます。

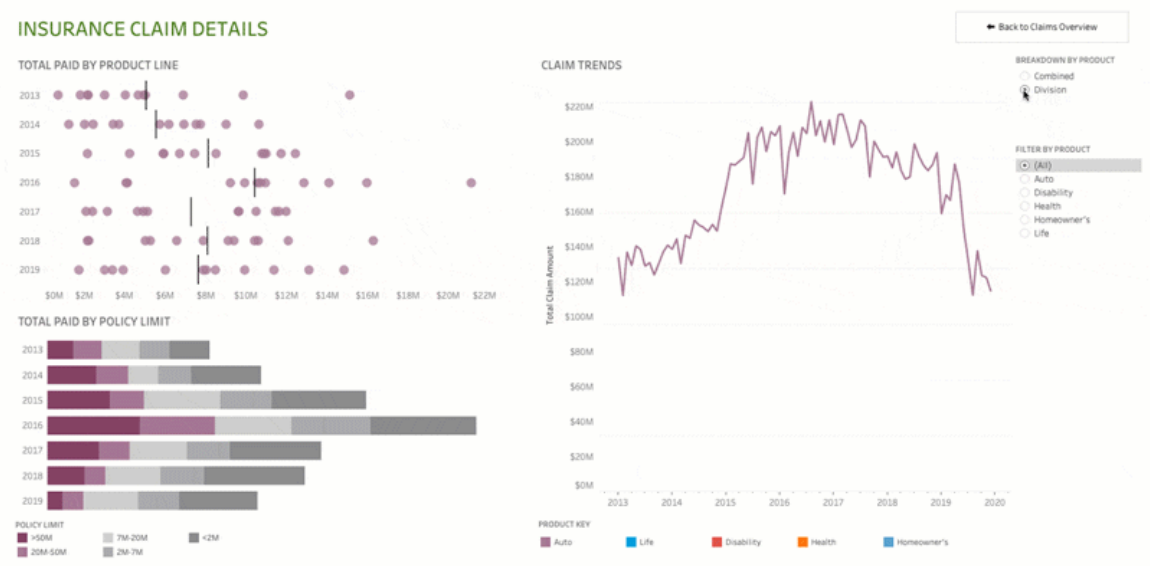
アニメーションは、フィルター、並べ替え、ズームの設定、さまざまなページ、ならびにフィルター、パラメーター、設定アクションに対する変更をビジュアルで移行します。これらの変化に応じてビジュアライゼーションがアニメーション化するため、ユーザーはデータの違いをより明確に認識して、情報に基づいた意思決定を行うことができます。

同時および連続したアニメーションを理解する

アニメーションを作成する場合、2 つの異なるスタイル (同時または連続するスタイル) を選択可能です。それぞれの種類の例を次に示します。

同時アニメーション

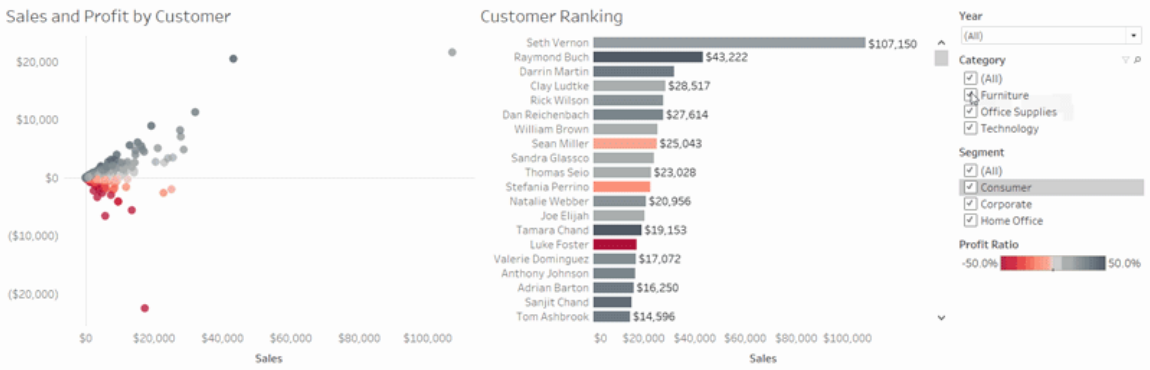
デフォルトの同時アニメーションは、より簡単なチャートやダッシュボードで値の変化を示すときに高速になり、正しく機能します。



上の画像をクリックするとアニメーションが再生されます。

連続したアニメーション

連続したアニメーションは時間がかかりますが、複雑な変更を段階的に表示することで明確にします。



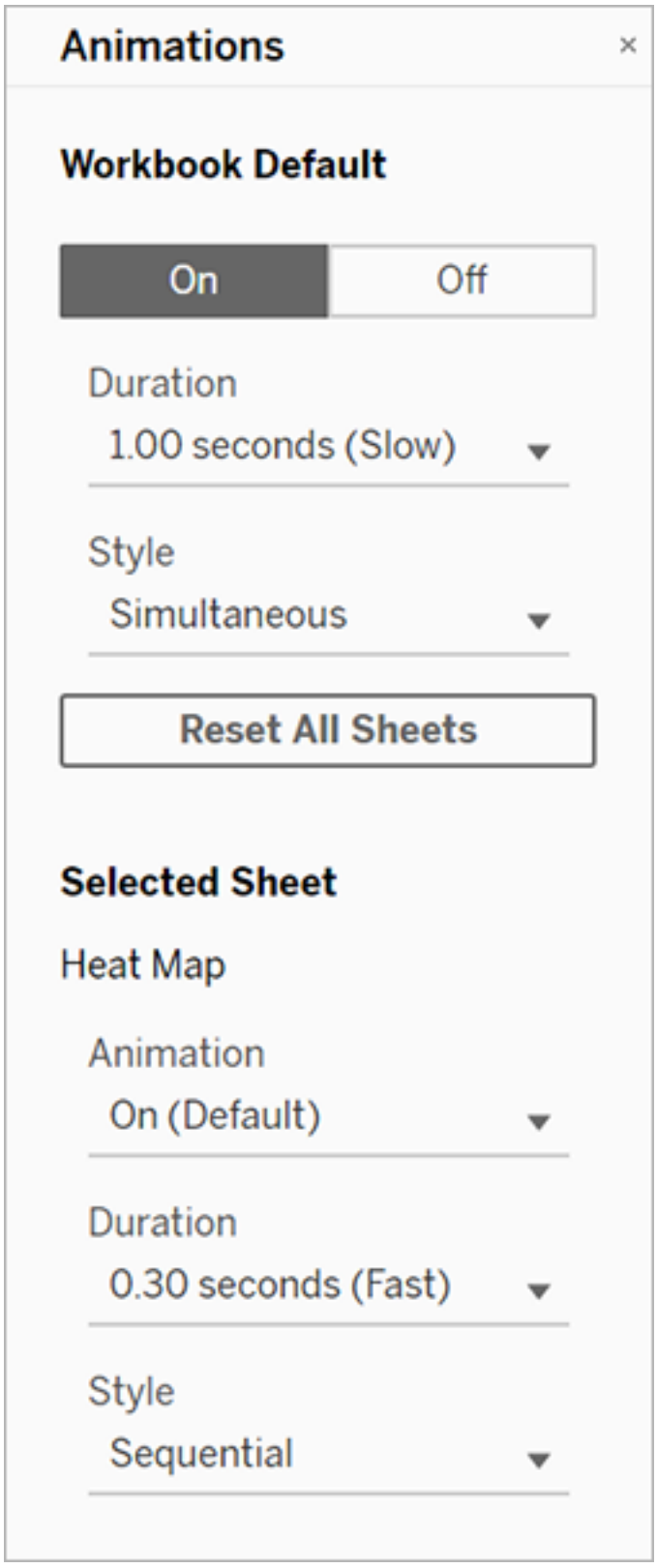
上の画像をクリックするとアニメーションが再生されます。

ワークブックのビジュアライゼーションをアニメーション化する

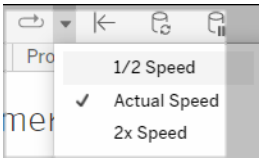
新しいワークブックを作成すると、Tableau では既定で Viz のアニメーションが有効になります。ユーザーおよびワークブックのレベルでアニメーションのオンとオフを切り替えることができます。

1. **[書式設定] > [アニメーション]** を選択します。
2. すべてのシートをアニメーション化する場合は、**[ワークブックの既定値]** で **[オン]** をクリックします。その後、次を実行します。
   - **[継続時間]** で、プリセットを選択するか、最大 10 秒のカスタム時間を指定します。
   - **[スタイル]** では、**[同時]** を選択してすべてのアニメーションを一度に再生するか、**[連続]** を選択してマークをフェードアウトし、アニメーションを移動して並べ替えてからフェードインします。
3. 特定シートに対するワークブック規定値を無効にするには、**[選択したシート]** の設定を変更します。

> **注:** [選択したシート]セクションで、「(デフォルトが)」関連するワークブックのデフォルト設定を自動的に反映する設定を示します。


Animations
Workbook Default
On
Off
Duration
1.00 seconds (Slow)
Style
Simultaneous
Reset All Sheets
Selected Sheet
Heat Map
Animation
On (Default)
Duration
0.30 seconds (Fast)
Style
Sequential

アニメーションを再生するには、ツールバーの **[再生]** ボタンをクリックします。**[再生]** ボタンから、アニメーションを再生する速度 (実際の速度、2 倍速、または 1/2 速度) を選択することができます。



ワークブックのアニメーション設定をリセットする

アニメーションをリセットして、ワークブック全体を既定のアニメーション設定に戻すことができます。これを行うとアニメーションは既定でオフになる点に注意してください。

1. **[書式設定] > [アニメーション]** を選択します。
2. **[アニメーション]** ペインの中央にある **[すべてのシートをリセット]** をクリックします。

すべてのアニメーションを完全に無効にする

既定では、新しいワークブックを作成すると、Viz のアニメーションが有効になります。ビジュアライゼーションを表示しているときにアニメーションが気が散る場合は、それらが再生されないように完全に無効にすることができます。(これはシステム全体の設定ではなく、それぞれのユーザーが個別に適用する必要があります。)

- Tableau Desktop で、**[ヘルプ] > [設定とパフォーマンス]** を選択し、**[アニメーションを有効にする]** の選択を解除します。
- Tableau Cloud または Tableau Server で、ブラウザーの右上隅にあるお使いのプロファイル画像またはイニシャルをクリックし、**[マイ アカウント設定]** を選択します。次に、ページの一番下までスクロールし、**[アニメーションを有効にする]** をオフにして、**[変更の保存]** をクリックします。

注: アニメーションが無効になっている場合、オーサリングモードで **[フォーマット] > [アニメーション]** を選択して設定を調整することはできますが、効果はありません。

軸のアニメーションの小数点を書式設定する

メジャーの小数点が既定で設定されている場合、軸のアニメーションで表示される小数点以下の桁数が、アニメーションの再生中に変動する可能性があります。これを回避するには、メジャーに表示される小数点以下の桁数を書式設定します。詳細については、「数値とNull値の書式設定」を参照してください。

アニメーションが再生されない理由

サーバー レンダリング

Vizがサーバーでレンダリングされている場合、アニメーションは再生されません。Vizがクライアント PC やモバイル デバイスでレンダリングされるようにするには、次の方法を使用します。

- Vizの作成者は、Vizの複雑さを減らします。
- Tableau Server管理者は、クライアント側レンダリングの複雑さのしきい値を大きくします。

> **注:** 処理能力の低いコンピューターでは、アニメーションが途切れることがありますが、ユーザーは応答性を低下させることなく、引き続きビジュアライゼーションを操作できます。

サポートされていないブラウザーと機能

アニメーションは、Internet Explorer を除くすべての Web ブラウザーでサポートされています。

次の Tableau 機能はアニメーション化を行いません。

- Web ブラウザーのマップ、多角形、密度マーク
- 円グラフとテキスト マーク
- ヘッダー
- 予測、傾向、および参照ライン
- ページ履歴のトレイル (Vizにこれらが含まれている場合は、予期しない動作を避けるためにアニメーションをオフにします)。

数字とNULL 値の書式設定

viz に表示される、メジャー、ディメンション、パラメーター、計算フィールド、軸ラベルなどの数値の形式を指定できます。数値形式を指定する場合、数値、通貨、科学、パーセンテージなどの一連の標準形式から選択できます。特殊文字を含めるオプションでカスタム数値形式を定義することもできます。

メジャーに null 値が含まれている場合は、null 値をゼロに置き換えたり非表示にしたりすることなど、書式設定を使用して null 値を別の方法で処理できます。

Tableau Desktop の場合

数値形式を指定する

1. ビュー内の数値を右クリック(Mac では Control を押しながらクリック)し、**[書式設定]** を選択します。
2. **[書式設定]** ペインで、**[数値]** ドロップダウン メニューをクリックします。
3. 数値形式を選択します。

   形式によっては追加の設定が必要です。たとえば、**[指数]** を選択した場合、小数点の位置も指定する必要があります。

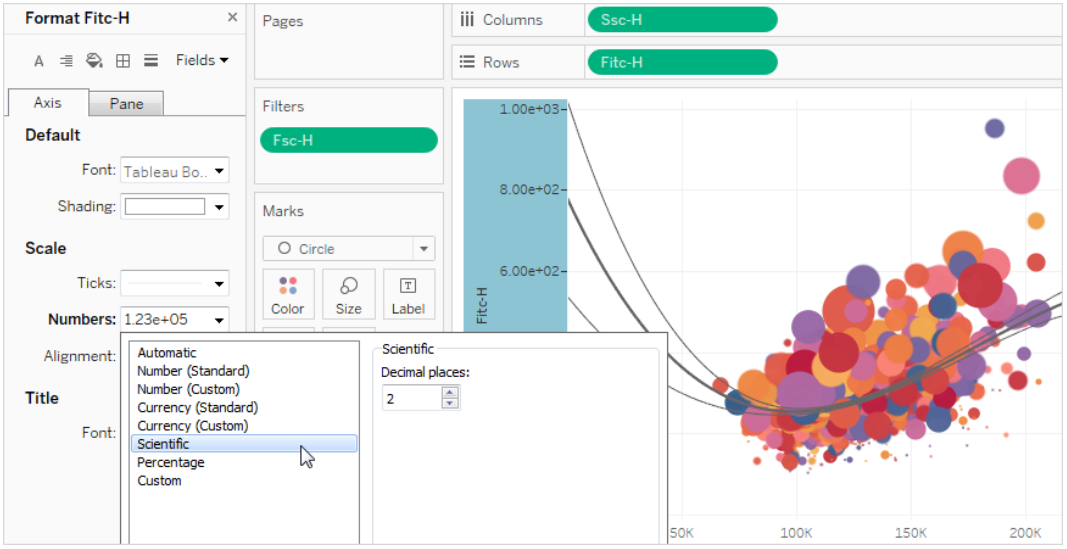


Tableau で利用可能な数値の形式および関連オプションを、以下に示します。

数値形式	形式のオプション
自動: データソースによって指定された形式、またはフィールドに含まれるデータのいずれかに基づいて、形式が自動的に選択されます。	なし。
数値 (標準): 形式は選択されたロケールに基づきます。	**ロケール**: 数値形式は、選択された地理的位置に応じて変わります。
数値 (カスタム): 必要に応じて希望どおりに書式設定をカスタマイズします。	**小数点**: 表示する小数点以下の桁数。 **負の値**: 負の値の表示方法。 **単位**: 指定された単位で数値を表示します。たとえば、数値が 20,000、単位が千の場合、数値は 20K と表示されます。 **プレフィックス/サフィックス**: 表示される各数値の先頭に付ける文字と末尾に付ける文字。 **千の桁区切りを含める**: 数値に千の桁区切りを表示するかどうかを指定します (例: 100,000 または 100000)。
通貨 (標準): 形式および通貨記号は選択されたロケールに基づきます。	**ロケール**: 通貨形式は、選択された地理的位置に基づきます。
通貨 (カスタム): 必要に応じて形式および通貨記号をカスタマイズします。	**小数点**: 表示する小数点以下の桁数。 **負の値**: 負の値の表示方法。 **単位**: 指定された単位で数値を表示します。たとえば、数値が 20,000、単位が千の場合、数値は

	20K と表示されます。 **プレフィックス/サフィックス**: 表示される各数値の先頭に付ける文字と末尾に付ける文字。 **千の桁区切りを含める**: 数値に千の桁区切りを表示するかどうかを指定します (例: 100,000 または 100000)。
指数: 科学的表記法で数値を表示します。	**小数点**: 表示する小数点以下の桁数。
パーセンテージ: 数値をパーセンテージとしてパーセント記号付きで表示します。値 1 は 100%、値 0 は 0% と解釈されます。	**小数点**: 表示する小数点以下の桁数。
カスタム: 形式は、形式のオプションで指定したものに完全に基づきます。	**カスタム**: 使用する形式を入力します。オプションで特殊文字を含めます。詳細については、このトピックの「カスタム数値形式を定義する」を参照してください。

カスタム数値形式を定義する

viz にカスタム数値形式を適用するには、次の操作を行います。

1. ビュー内の数値を右クリック (Mac では Control を押しながらクリック) し、**[書式設定]** を選択します。
2. **[書式設定]** ペインで **[数値]** ドロップダウン メニューをクリックし、**[カスタム]** を選択します。
3. **[書式]** フィールドで、次の構文 `Positive number format;Negative number format;Zero values` を使用して書式設定を定義します。

数値形式のコードを定義するときは、次の点に注意してください。

- 数値形式は、正の数、負の数、ゼロの順序に最大 3 タイプを指定できます。
- 数値の各タイプはセミコロン (;) で区切る必要があります。
- 数値タイプを 1 つだけ指定すると、そのタイプの形式がすべての数値に適用されます。

- 数値タイプを2つ指定すると、最初のタイプの形式が正の数とゼロに適用され、2番目のタイプの形式が負の数に適用されます。
- 数値形式のコードでタイプをスキップするときは、欠落するタイプごとにセミコロン(;)を含める必要があります。

カスタム数値形式の例

viz で使用できる一般的なカスタム数値形式のコードの例については、次の表を参照してください。

構文には、セミコロンで区切られた3つの部分 `<positive number format>;<negative number format>;<zero format>` があります。

使用事例	数値コード構文	結果の例
正の値のみを表示	#,##; ; (2、3番目のセミコロンの後の空白に注意)	正の値: 1,234 負の値: (空白部分のみ表示) ゼロ値: (空白部分のみ表示)
負の値のみを表示	;-#,##;	正の値: (何も表示されません) 負の値: -1,234 ゼロ値: (何も表示されません)
ゼロ値のみを表示	;;0;	正の値: (何も表示されません) 負の値: (何も表示されません) ゼロ値: 0
ゼロ値を非表示	#,###;-#,###;;	正の値: 1,234 負の値: -1,234 ゼロ値: (何も表示されません)
負の値を括弧内に表示	#,###;(#,###);;	正の値: 1,234

		負の値: (1,234) ゼロ値: (何も表示されません)
値に文字のプレフィックスを追加	$#,###.##;-$#,###.##;$0;	正の値: $1,234.56 負の値: -$1,234.56 ゼロ値: $0
値に文字のサフィックスを追加	#%;-#%;0%;	正の値: 12% 負の値: -34% ゼロ値: 0%
テキスト記述子を追加	"$"#,#" Surplus";"$"-##" Shortage";"$"0;	正の値: $1,234 過剰 負の値: $-1,234 不足 ゼロ値: $0

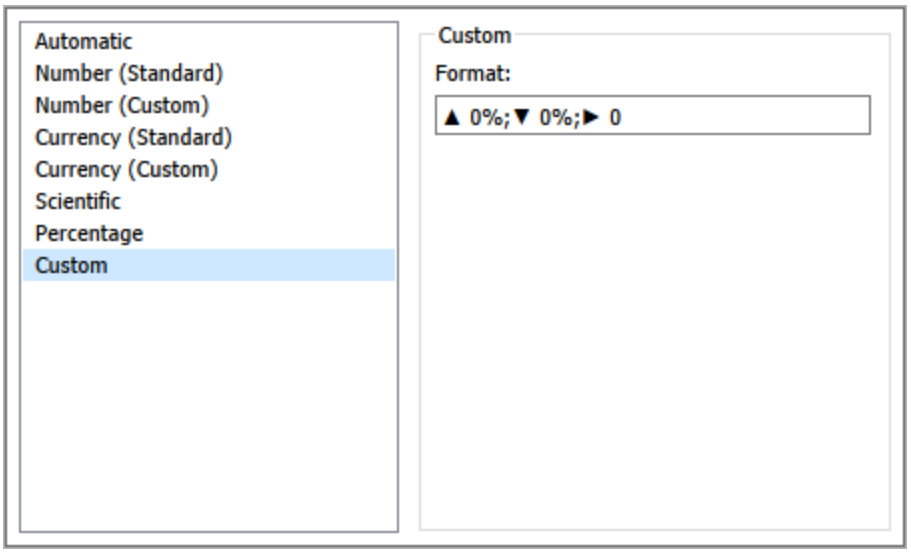
ニーズに合わせて数値形式をカスタマイズする方法はいくつかあります。コードの説明と事例の詳細については、Microsoft ナレッジベースの「数値形式をカスタマイズするためのレビューガイドライン」を参照してください。

注: Microsoft が説明しているオプションと、Tableau で使用可能なオプションの間にはわずかな違いがあります。たとえば、列内のテキストを揃えるカスタム数値形式は、Tableau には関係ありません。また、マークカードを使用してテキストに色を付けることができるため、テキストをカラーコード化するカスタム数値形式は適用できません。(色の適用については、「ビュー内のマークの表示を制限する」を参照してください。)Tableau で適用できるカスタム数値形式のみを使用してください。

カスタム数値形式に特殊文字を含める

カスタム数値形式の利点の 1 つは、特殊文字や記号を含めることができることです。記号を使用すると、メジャーと計算の比較を viz ですばやく理解しやすくなります。

たとえば、会社で最も売れている3つの製品の利益を月ごとに比較したいとします。標準ラベルを使用してこれらの製品の利益が先月から+5%、-2%、0%変化したと表示するのではなく、カスタム数値形式を使用してこれらの変化を▲5%、▼2%、►0と表示することができます。



フィールドに既定の数値形式を設定する

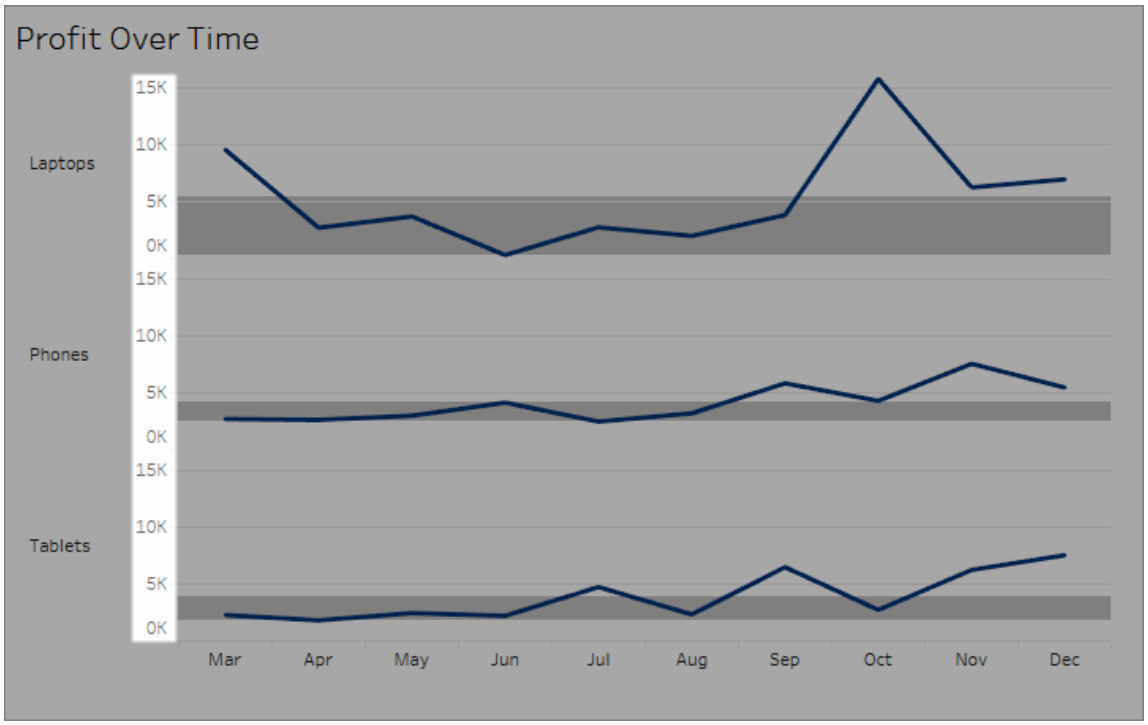
[データ] ペインでフィールドを右クリック(Macで Control を押しながらクリック)し、**[既定のプロパティ] > [数値書式]** を選択します。

次に表示されるダイアログ ボックスで、フィールドをビューに追加するときに常に使用される数値形式を指定します。既定の数値形式は、ワークブックと共に保存されます。接続情報をエクスポートするときは、既定の数値形式もエクスポートされます。

> **注: [書式設定]** ペインを使用して数値を書式設定すると、別の箇所に適用した可能性がある任意の数値の書式がオーバーライドされます。

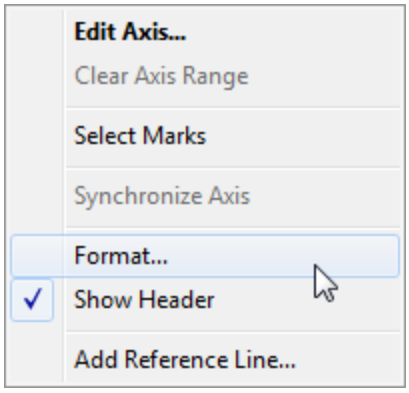
メジャーを通貨に書式設定する

次の画像のビューは、一定の期間における収益を示しています。縦軸の収益の数字は、通貨として書式設定されません。



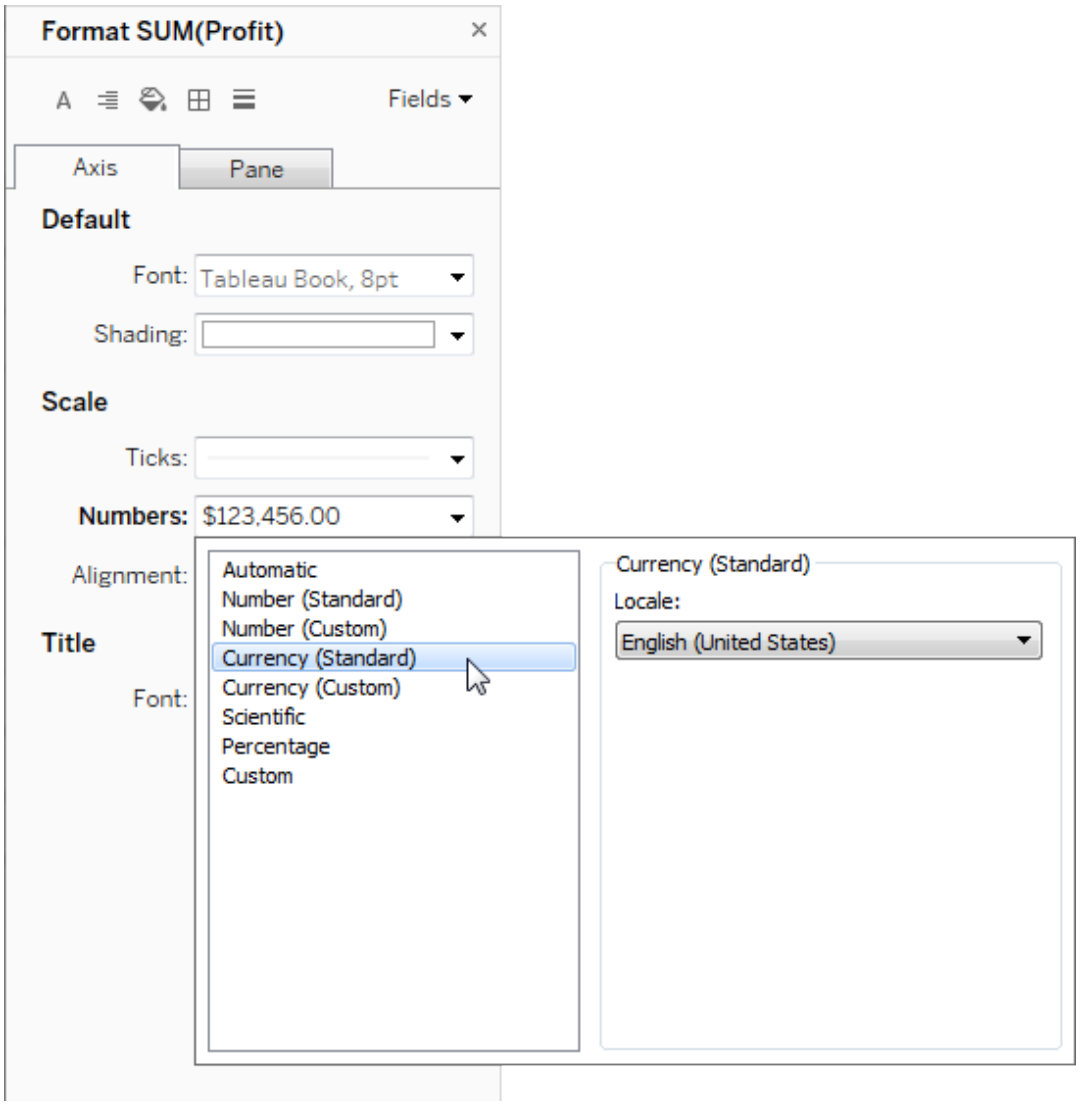
数字を通貨として書式設定する方には:

1. **"Profit (収益)"** 軸を右クリックして **[書式設定]** を選択します。



2. **[書式設定]** ペインの **[軸]** タブで、**[スケール]** から **[数字]** ドロップダウン リストを選択した後、以下のいずれかを選択します。

   通貨 (標準): ドル記号や小数点以下 2 桁を数字に追加します。



通貨 (カスタム): 小数点以下の桁数位、負の値の処理方法、単位、プレフィックスまたはサフィックスを含めるかどうか、千の桁区切りを含めるかどうかを指定します。

ロケールを使用して数値形式を指定する

既定では、Tableau はお使いのコンピューターのロケールと言語設定を使用して数字を書式設定します。ただし、**[書式設定]** ペインから明示的に別のロケールを設定することもできます。

次の手順では、以前のセクションと同じビューを使用してスイス (ドイツ語圏) の通貨を設定する方法を示します。

1. **"Profit (収益)"** 軸を右クリックして **[書式設定]** を選択します。
2. **[軸]** タブの **[スケール]** から **[数字]** ドロップダウン リストを選択した後、**[通貨 (標準)]** を選択します。
3. **[ロケール]** ドロップダウン リストで、アイテムは **[言語 (国)]** 形式に表示されます。この例では、**"German (Switzerland) (ドイツ (スイス))"** を選択します。ビューが更新されて売上高をスイス フランで、ドイツ語用に書式設定して表示します。

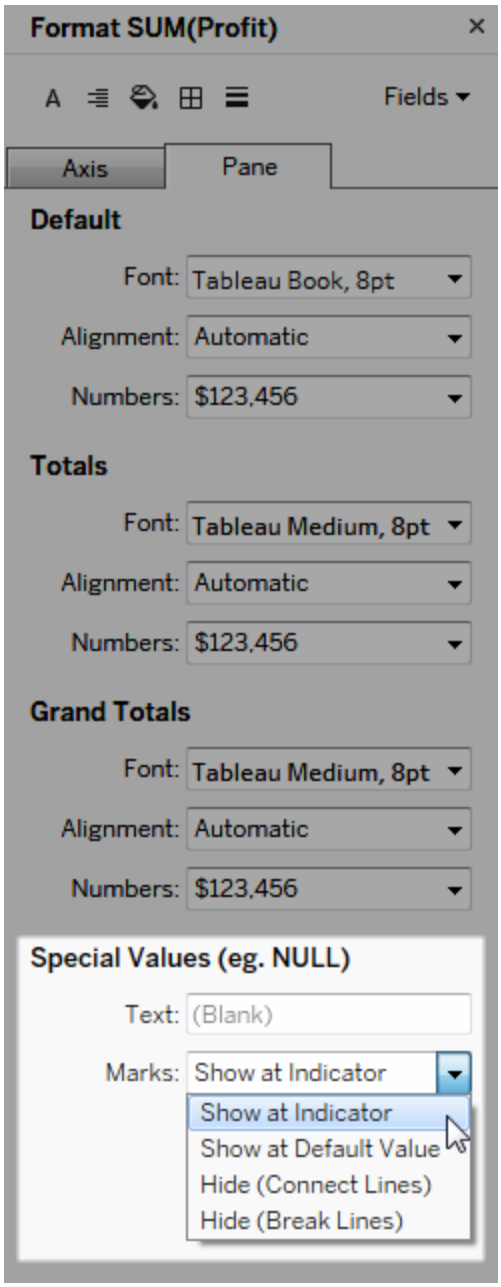
ヒント: 既定の通貨設定を変更し、**"Profit (収益)"** メジャーをビューまでドラッグするたびに希望する設定を使用します。**[データ]** ペインで **"Profit (収益)"** (またはその他の通貨メジャー) を右クリックし、**[既定のプロパティ] > [数値形式]** を選択します。次に、フィールドを上記のように書式設定します。

NULL 値の書式設定

通常、メジャーに含まれている NULL 値は、ビュー内ではゼロとしてプロットされます。ただし、これによりビューが変わることがあるので、NULL 値を完全に非表示にする方が良い場合もあります。NULL 値を独自の方法で処理するように各メジャーを書式設定できます。

特定のフィールドの NULL 値を書式設定するには:

1. NULL 値を含むビューでフィールドを右クリック (Mac では Control を押しながらクリック) し、**[書式設定]** を選択します。
2. **[ペイン]** タブに移動します。
3. **[特殊な値]** エリアでは、ビューの右下隅のインジケーターを使用して、NULL 値を表示するかどうかを必要に応じて指定し、既定値 (数値フィールドのゼロなど) でこれらの値をプロットします。また値を非表示にして行をつなぐか、NULL 値が存在することを示すために値を非表示にして改行するかを指定します。



4. **[テキスト]** フィールドでテキストを指定すると、マークラベルがオンの場合に、NULL 値に対するテキストがビューに表示されます。「マークラベルの表示と非表示」を参照してください。

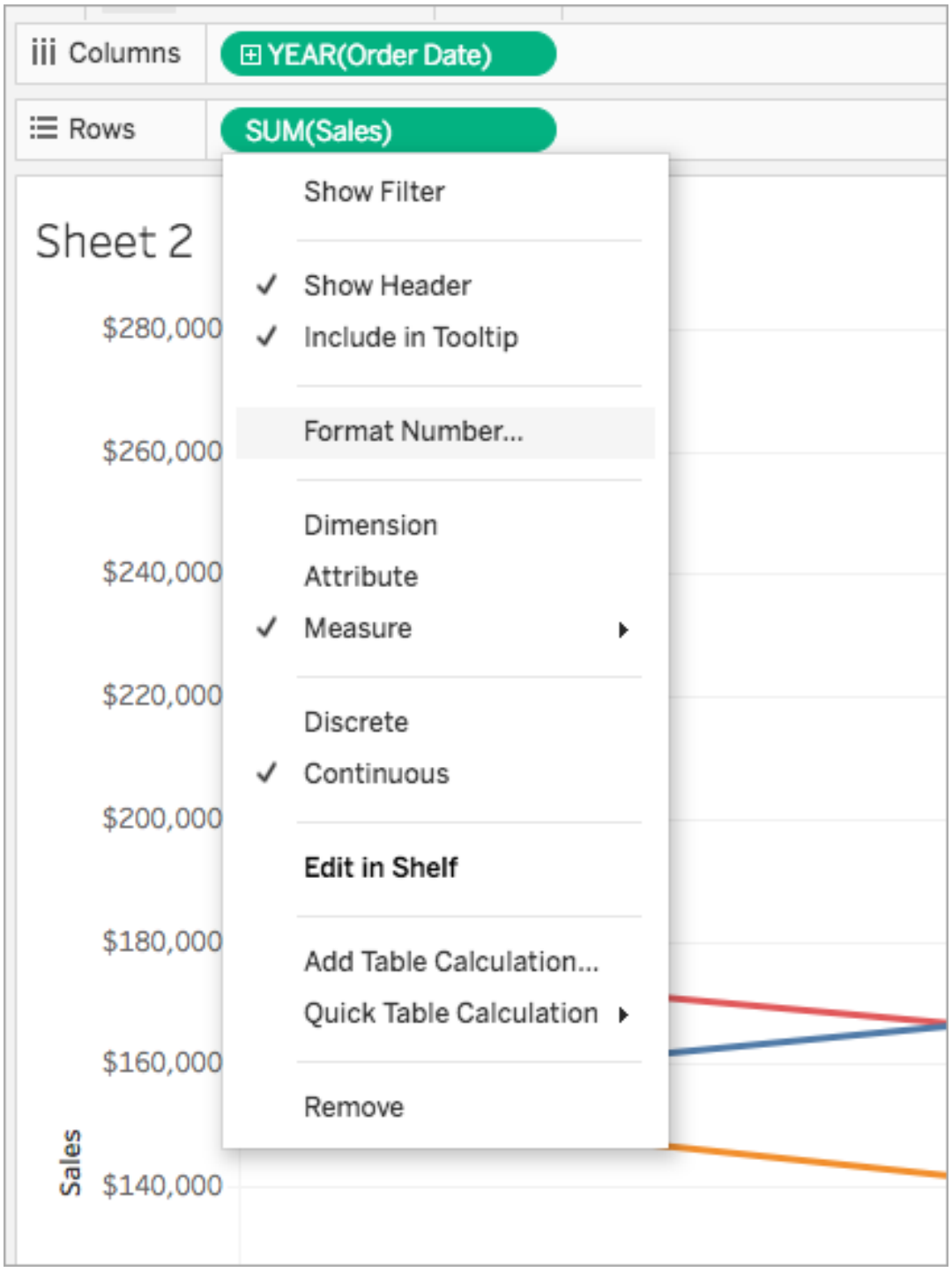
注: [特殊な値] エリアはディメンションや不連続メジャーには使用できません。

Tableau Server または Tableau Cloud の場合

数値形式を指定する

Web でビューを作成している場合は、ビューで使用されるフィールドの数値形式を指定できます。

1. Web 編集モードで、ビュー内のメジャーを右クリックし、**[数値の書式設定]** を選択します。



2. 表示されたダイアログ ボックスで、数値形式を選択します。

   形式によっては追加の設定があります。たとえば、**[通貨]** を選択する場合は、小数点以下の桁数、単位、およびコンマなどの桁区切りを含めるかどうかも指定する必要があります。

   この例では、"Sales (売上高)" は、小数点以下 0 桁の 1000 (k) 単位の通貨として書式設定されています。ビュー内の "Sales (売上高)"の数値は、これらの設定で更新されます。同様に、ラベルとツールヒントも更新されます。

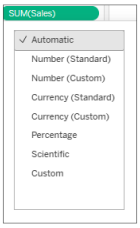


Tableau で利用可能な数値の形式および関連オプションを、以下に示します。

数値形式	形式のオプション
自動: データ ソースによって指定された形式、またはフィールドに含まれるデータのいずれかに基づいて、形式が自動的に選択されます。	なし。
数値 (カスタム): 必要に応じて希望どおりに書式設定をカスタマイズします。	**小数点**: 表示する小数点以下の桁数。 **単位**: 指定された単位で数値を表示します。たとえば、数値が 20,000、単位が千の場合、数値は 20K と表示されます。 **桁区切りを含める**: 数値に千の桁区切

	りを表示するかどうかを指定します (例: 100,000 または 100000)。
通貨 (カスタム): 必要に応じて形式および通貨記号をカスタマイズします。	**小数点**: 表示する小数点以下の桁数。 **単位**: 指定された単位で数値を表示します。たとえば、数値が 20,000、単位が千の場合、数値は 20K と表示されます。 **桁区切りを含める**: 数値に千の桁区切りを表示するかどうかを指定します (例: 100,000 または 100000)。
パーセンテージ (カスタム): 数値をパーセンテージとしてパーセント記号付きで表示します。値 1 は 100%、値 0 は 0% と解釈されます。	**小数点**: 表示する小数点以下の桁数。
指数 (カスタム): 科学的表記法で数値を表示します。	**小数点**: 表示する小数点以下の桁数。

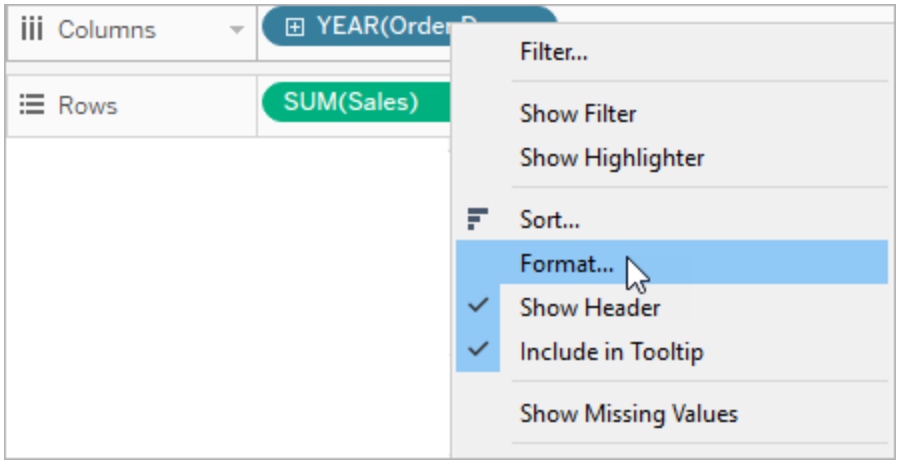
カスタム日付形式

この記事では、カスタム日付形式フィールドを使用してビュー内の日付を書式設定する方法を説明します。Tableau で日付が処理される方法の概要については、「日付と時刻」または「日付レベルの変更」を参照してください。**データ ソース**の日付プロパティを設定する方法については、「データ ソースの日付プロパティ」を参照してください。

カスタム日付形式フィールドを見つける方法

ビュー内の日付フィールドの書式設定 (Tableau Desktop)

Tableau Desktop でビュー内の日付フィールドを書式設定するには、フィールドを右クリック (Mac では Control を押しながらクリック) し **[書式設定]** を選択します。



これにより、ビューの左側に **[書式設定]** パネルが開きます。**[日付]** フィールドを選択します。

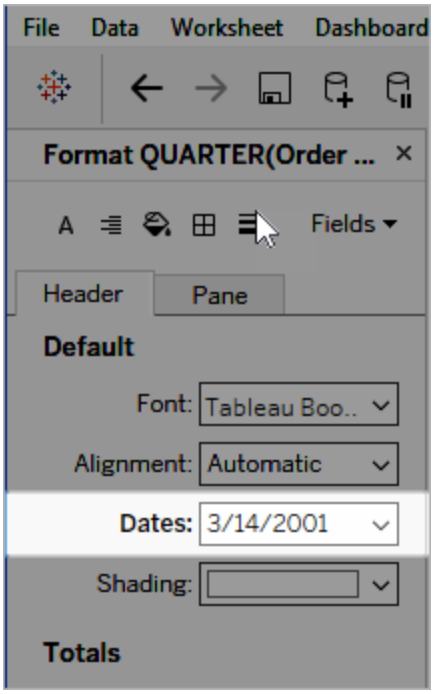
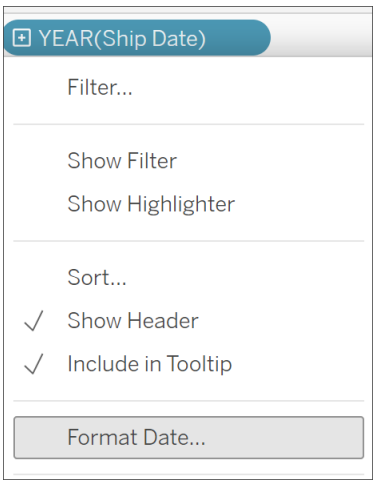


Tableau は利用可能な書式設定の一覧を表示します。不連続な日付部分を扱っていると、その日付部分に関連するオプションが表示されます (**YEAR([オーダー日])** のオプションとして [自動]、[2 桁]、[4 桁] など)。連続または正確な日付の不連続日付では、リストの最後の項目は **[カスタム]** です。「サポートされている日付形式記号」の表に記載されている形式記号を単独で使用するか、組み合わせて使用し、カスタム日付を指定できます。

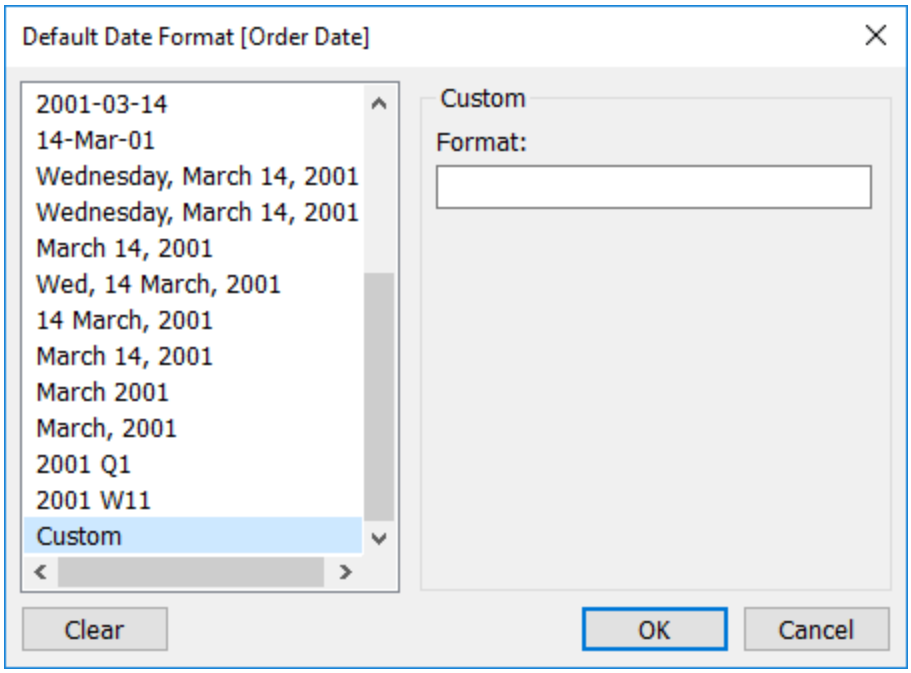
ビュー内の日付フィールドの書式設定 (Tableau Cloud および Tableau Server)

Tableau Cloud および Tableau Server でビュー内の日付フィールドを書式設定するには、フィールドを右クリック (Mac では Control を押しながらクリック) し **[書式設定]** を選択します。



[データ] ペイン内の日付フィールドの書式設定 (Tableau Desktop のみ)

[データ] ペインの日付フィールドを書式設定するには、フィールドを右クリックして **[既定のプロパティ] > [日付形式]** を選択します。



表の日付形式は、ワークブックが Tableau 抽出に接続されている場合、または日付形式をサポートするデータ ソースへのライブ接続がある場合にサポートされます。(希望する日付形式がサポートされていることを確認するには、データ ソースのドキュメントを参照してください。)

Tableau はデータ ソースから日付形式を取得します。また、Tableau Server は、Tableau Server を実行しているサーバーの実行ユーザー アカウントから日付形式を取得することもできます。

注: 次の日付形式は、Type Conversion 関数で使用されるものと異なる場合があります。詳細については、「文字列を日付フィールドに変換する」を参照してください。

サポートされている日付形式記号

次の記号を使用して、カスタム日付形式を作成します。

記号	説明
(:)	時間区切り文字。一部のロケールでは、時間区切り文字を表すため、別の文字列が使用されます。時間区切り文字は、時間の値を書式設定したときに時間、分、秒を区切ります。書式設定された出力で時間区切り文字として使用

	される実際の文字は、コンピュータのシステム設定によって決定されます。
(/)	日付区切り文字。一部のロケールでは、日付区切り文字を表すために別の文字が使用されます。日付区切り文字は、日付の値を書式設定したときに日、月、年を区切ります。書式設定された出力で日付区切り文字として使用される実際の文字は、コンピュータのシステム設定によって決定されます。
c	日付を ddddd、 時間を ttttt としてこの順で表示します。日付シリアル番号に分数部分がない場合は日付情報のみを表示し、整数部分がない場合は時間情報のみを表示します。
d	日付を先頭にゼロを付けずに数字 (1 ~ 31) で表示します。
dd	日付を先頭にゼロを付けて数字 (01 ~ 31) で表示します。
ddd	曜日を略語 (Sun、Sat) で表示します。
dddd	曜日を完全な名前 (Sunday、Saturday) で表示します。
ddddd	日付をお使いのシステムの短い日付形式設定に従って書式設定された完全な日付 (日、月、年を含む) で表示します。既定の短い日付形式は m/d/yy です。
dddddd	日付のシリアル番号を、システムで認識されている長い日付設定に従って書式設定された完全な日付 (日、月、年を含む) で表示します。既定の長い日付形式は mmmm dd, yyyy です。
aaaa	dddd と同じですが、文字列のローカライズされたバージョンです。
w	曜日を数字 (日曜日は 1、土曜日は 7 など) で表示します。
ww	週 (年間) を数字 (1 ~ 54) で表示します。
M	月の先頭にゼロを付けずに数字 (1 12) で表示します。「m」の直後に「h」または「hh」が来る場合、月ではなく分が表示されます。
MM	月を先頭にゼロを付けて数字 (01 ~ 12) で表示します。「m」の直後に「h」または「 hh」が来る場合、月ではなく分が表示されます。
MMM	月を略語 (Jan ~ Dec) で表示します。

MMMM	月を完全な名前 (January ~ December) で表示します。
MMMMM	月を1文字の略語 (J ~ D) で表示します。
oooo	MMMM と同じですが、ローカライズされます。
q	年の四半期を数字 (1 ~ 4) で表示します。
y	年の通算日を数字 (1 ~ 366) で表示します。
yy	年を2桁の数字 (00 ~ 99) で表示します。
yyyy	年を4桁の数字 (100 ~ 9999) で表示します。
h	時間を先頭にゼロを付けずに数字 (0 ~ 23) で表示します。
Hh	時間を先頭にゼロを付けて数字 (00 ~ 23) で表示します。
N	分の先頭にゼロを付けずに数字 (0 59) で表示します。
Nn	分の先頭にゼロを付けて数字 (00 59) で表示します。
S	秒の先頭にゼロを付けずに数字 (0 59) で表示します。
Ss	秒の先頭にゼロを付けて数字 (00 59) で表示します。
000	ミリ秒を表示します。ミリ秒を指定する前の区切り文字として、ピリオドを使用します。
ttttt	時間をシステムで認識される時間形式で定義された時間区切り文字を使用した形式の、完全な時刻 (時、分、秒を含む) で表示します。先頭にゼロを付けるオプションが選択され、時刻が 10:00 A.M. または P.M. より前の場合、先頭にゼロが表示されます。既定の時間形式は `h:mm:ss` です。
AM/PM	12時間形式を使用し、正午より前の時刻には大文字の「AM」、正午から11:59 P.M. の時刻には大文字の「PM」を付けて表示します。
am/pm	12時間形式を使用し、正午より前の時刻には小文字の「AM」、正午から11:59 P.M. の時刻には小文字の「PM」を付けて表示します。
A/P	12時間形式を使用し、正午より前の時刻には大文字の「A」、正午から11:59 P.M. の時刻には大文字の「P」を付けて表示します。

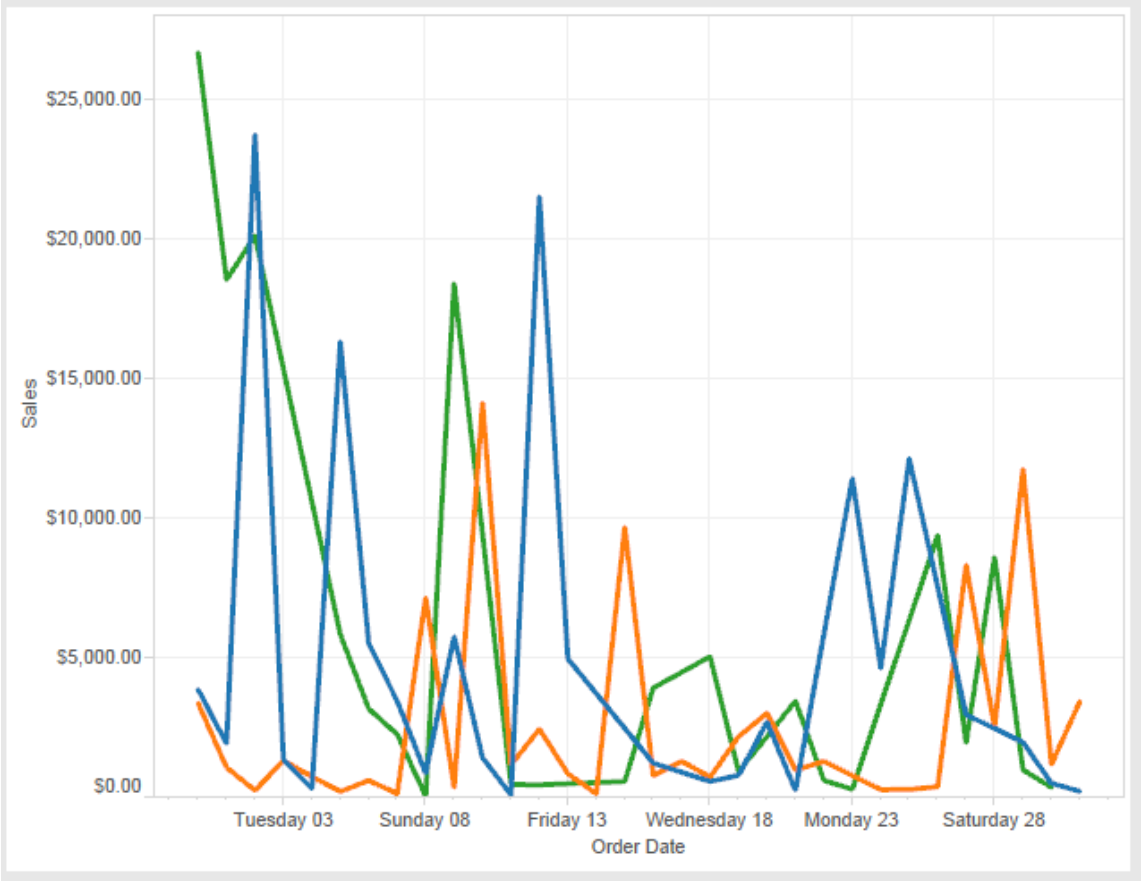
a/p	12 時間形式を使用し、正午より前の時刻には小文字の「A」、正午から 11:59 P.M. の時刻には小文字の「P」を付けて表示します。
AMPM	12 時間形式を使用し、システムで定められたとおり、正午より前の時刻には AM 文字列リテラル、正午から 11:59 P.M. の時刻には大文字の PM 文字列リテラルが表示されます。AMPM には大文字か小文字のいずれかを使用できますが、文字列の大文字/小文字はシステム設定で定義された文字列と同じになります。既定の形式は AM/PM です。

カスタム日付形式の例

上記の表の日付形式記号は、単独で使用するか、組み合わせて使用することができます。

カスタム形式 yyyy-MM-dd HH:mm:ss.000 を指定すると、2015-05-10 11:22:16.543 という形式で日付が生成されます。このような形式は科学的データに適している場合があります。

カスタム形式 DDDD DD を指定すると、以下のように曜日と日付を表示するデータが生成されます。



カスタム形式 `yy-mm-dd (dddd)` を指定すると、形式 **18-01-04 (木曜日)** の日付が生成されます。

カスタム形式 `"Q"1 YYYY` を指定すると、**Q1 2018** を示す日付が生成されます。

日本の元号に基づいた日付形式のサポート

Tableau は日本の元号に基づいた日付 (和暦) 形式をサポートします。元号に基づいた日付形式をビューのフィールドに適用する方法は、次のとおりです。

1. ワークブックのロケールを日本に設定します。
2. 日付形式を設定するビューのフィールドを右クリックします。

3. **[書式設定]** を選択します。

4. **[書式設定]** ペインの、**[日付]** ドロップダウン リストから、形式を選択します。

希望する形式が一覧にない場合は、独自の日付形式を作成できます。これを行うには、**[日付]** ボックスで **[カスタム日付]** を選択してから、Tableau の日付プレースホルダーを使用して書式を入力します。使用できる元号に基づいたプレースホルダーは、次のとおりです。

記号	説明
g	元号の省略形 (平成の場合は「H」など)。
gg	元号名 (「平成」など)。
ggg	元号の長い名前 (日本語の場合、この名前は標準的な元号名と同じです)。
e	元号に基づいた年 (例:元号の最初の年は 1)。
ee	元号に基づいた年 (例:元号の最初の年は 01)。1 桁しかない場合、元号に基づいた年の先頭にゼロが 1 つ追加されます。

ワークブックのロケールが日本語ではない場合、次のように、カスタム日付形式を作成してから形式の前に言語コード「!ja_JP!」を挿入します。

```
!ja_JP! gg ee"年"mm"月"dd"日"
```

言語コードは、日付を強制的に日本の日付として処理します。

元号に基づいた日付は、Tableau Server ブラウザ ビューでは完全にはサポートされていません。特に、インタラクティブ フィルターを含むワークブックをパブリッシュする場合、e および g プレースホルダーは入力されません。

この問題を回避するため、ワークブックをブラウザーで表示する場合は、元号をベースにした日付をインタラクティブ フィルターで表示しないでください。

日付形式でリテラル テキストを使用する

日付形式に、言葉や語句を含める場合があります (**yyyy 年度の第 q 四半期** など)。ただし、テキストを Tableau の書式設定ボックスに直接入力すると、文字は日付として処理される場合があります。

Quarter of Order Date
Fi01/1/2010al 1uarter 1 of 2010
Fi04/1/2010al 2uarter 2 of 2010
Fi07/1/2010al 3uarter 3 of 2010
Fi010/1/2010al 4uarter 4 of 2010

Tableau がこれを行うのを防ぐため、日付として処理しない文字と単語を二重引用符で囲みます。`"Fiscal Quarter" q "of" yyyy`

引用されたセクションの内側にリテラル引用を使用する場合は、このコード `"\""` を挿入します。たとえば、`"Fiscal "\"" Quarter"` という形式は、**第 “四半期** として書式設定されます。

抽出データ ソースでの DATEPARSE 関数の形式構文

抽出で DATEPARSE 関数を使用している場合は、Unicode Consortium (ユニコード コンソーシアム) が定義する構文を使用します。

次表に DATEPARSE 関数の形式パラメーターに示すことができるフィールド タイプが一覧されています。各フィールドの種類をクリックすると、記号、フィールドのパターン、例、説明が表示された Unicode Consorttium の Web サイトが表示されます。

時間の単位	**注**
Era (元号)	N/A
Year (年)	.hyper 抽出では "U" を除くすべての記号がサポートされています。 **注:**

時間の単位	注
	• マイナスの値は、紀元前 (BC) を表します。たとえば、`DATEPARSE('y','-10')` では紀元前 11 世紀の最初の 1 月を返し、`DATEPARSE('y','-0')` では紀元前 1 世紀の最初の 1 月を返します。 • 暦年に "y" を使用している場合、パターン "yy" にはその年の下 2 桁が必要です。70 超の数値の場合、DATEPARSE 関数は 2000+x 年を返します。70 以下の数値の場合、DATEPARSE 関数は 1900+x 年を返します。 • カレンダー日に基づいて "ISO 暦週日付" に "Y" を使用している場合は、年の移行は週の境界で行われ、暦年の移行とは異なる場合があります。指定した "Y" は ISO 年 - 週 カレンダーのパターン文字 "w" と併せて使用されます。ISO 暦週日付システムは、日付と時刻の表記に関する国際規格である ISO 8601 に含まれるうるう週のカレンダー システムに効果的です。"y" と同様に、"Y" のマイナス値は紀元前 (BC) を表します。
Month (月)	.hyper 抽出では "l" を除くすべての記号がサポートされています。 **注:** • 月の指定は、日付を表す "d" とともに使用されます。 • ICU とは異なり、.hyper 抽出では 1 ~ 12 の値が許可されます。それ以外の値を使用するとエラーが発生します。
Week (週)	.hyper 抽出では "W" を除くすべての記号がサポートされてい

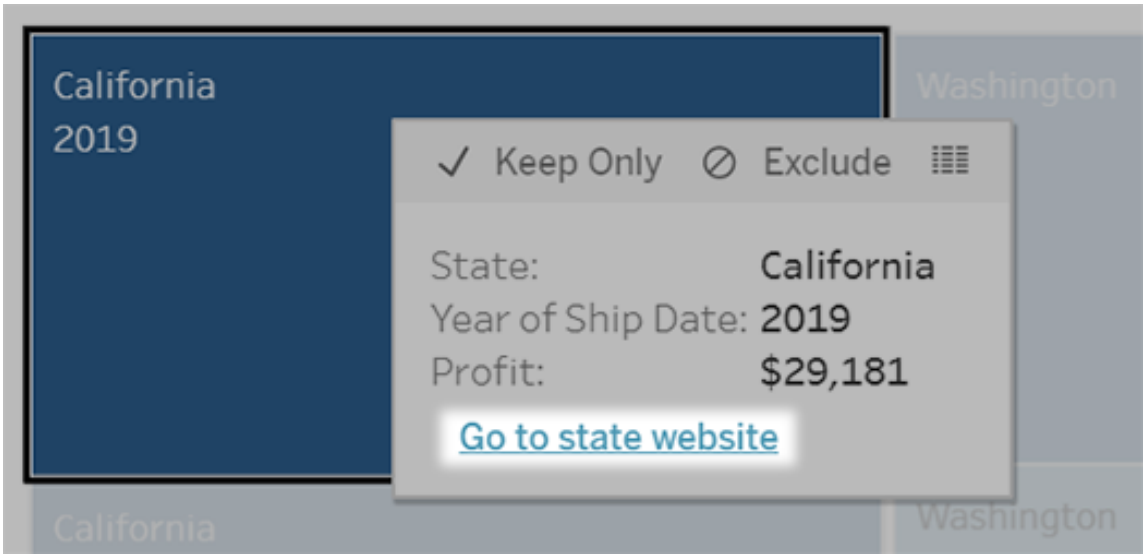
時間の単位	注
	ます。 **注:** • ICU とは異なり、"w" を使用している場合は .hyper 抽出で有効な週のみが許可されます。1 年は 52 週または 53 週です (ISO 8601)。DATEPARSE 関数で入力値が検証されます。たとえば、2016 年に 53 週目はないため、2016 年の 53 週目ではエラーが発生します。 • ICU では "W" の使用はサポートされていませんが、9 月の第 1 月曜日などの日付を指定する場合に便利です。
Day (日)	**注:** • ICU とは異なり、"d" を使用している場合は .hyper 抽出で有効な日の数値のみが許可されます。たとえば、2 月 31 日の場合はエラーが発生します。 • ICU とは異なり、"D" を使用している場合は .hyper 抽出で有効な日の数値のみが許可されます。たとえば、2017 年の 366 日目ではエラーが発生します。
Hour (時)	.hyper 抽出では、記号 "h" と"H" のみがサポートされています。 **注:** • "h" を使用している場合、.hyper 抽出のフィールドにはマイナスの値を使用できません。マイナスの値を使用するとエラーが発生します。

時間の単位	注
	• "H" を使用している場合、.hyper 抽出のフィールドにはマイナスの値を使用できません。マイナスの値を使用するとエラーが発生します。
Minute (分)	**注:** ICU とは異なり、.hyper 抽出のフィールドにはマイナスの値を使用できません。マイナスの値を使用するとエラーが発生します。
Second (秒)	**注:** • ICU とは異なり、.hyper 抽出のフィールドにはマイナスの値を使用できません。マイナスの値を使用するとエラーが発生します。 • "S" を使用している場合、`DATEPARSE ('ss.SSSS', '12.3456')` では 1990-01-01 00:00:12:3456 AD を返します。
Quarter (四半期)	**注:** ICU とは異なり、.hyper 抽出では 1 ~ 4 の値のみ許可されます。それ以外の値を使用するとエラーが発生します。
Weekday (曜日)	**注:** • ICU とは異なり、"e" および "ee" を使用している場合は .hyper 抽出で 1 ~ 7 の値のみが許可されます。それ以外の値を使用するとエラーが発生します。 • ICU とは異なり、"c..cc" を使用している場合は .hyper 抽出で 1 ~ 7 の値のみが許可されます。それ以外の値を使用するとエラーが発生します。
Period (期間)	N/A

URL アクション

URL アクションは、Tableau の外部にある Web ページ、ファイル、または他の Web ベース リソースを指すハイパーリンクです。URL アクションを使用すると、メールのほか、データに関する追加情報へのリンクを作成することができます。データに基づいてリンクをカスタマイズする場合は、URL にフィールド値をパラメーターとして自動的に入力できます。

ヒント: URL アクションも、ダッシュボードの Web ページ オブジェクトで開くことができます。詳細については、「アクションとダッシュボード」を参照してください。



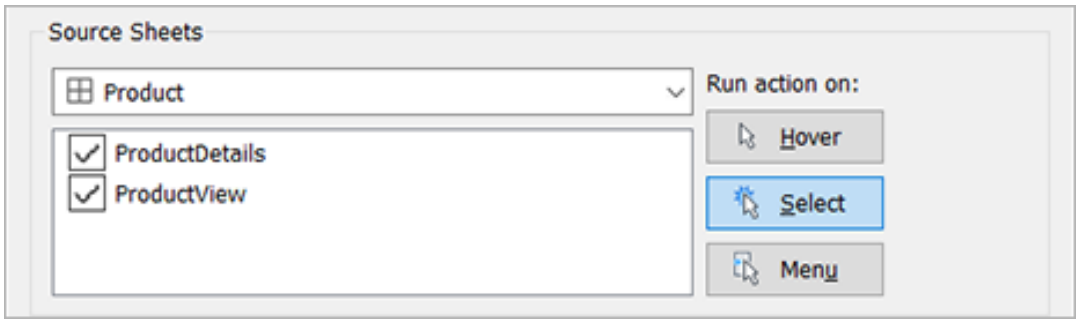
URL アクションは、ツールヒント メニューから実行されます。リンクには、ターゲット URL ではなく、アクション名が反映されます。

URL アクションで Web ページを開く

1. ワークシートで **[ワークシート] > [アクション]** を選択します。ダッシュボードから **[ダッシュボード] > [アクション]** を選択します。
2. [アクション] ダイアログ ボックスで、**[アクションの追加]** をクリックして、**[URL に移動]** を選択します。
3. 次のダイアログ ボックスで、アクションの名前を入力します。名前にフィールド変数を入力するには、**[名前]** ボックスの右側にある **[挿入]** メニューをクリックします。

> **注:** ツールヒントのリンクテキストは、URL ではなく、アクションの名前なので、そのアクションにはわかりやすい名前を付けます。たとえば、製品の詳細情報にリンクしている場合、「詳細を表示する」などが適切な名前になります。

4. ドロップダウン リストを使用して、ソース シートまたはデータ ソースを選択します。データ ソースまたはダッシュボードを選択する場合、それに含まれる個々のシートを選択できます。



5. ユーザによるアクションの実行方法を選択します。

このオプションを選択した場合	ユーザーが次の操作を行ったときにアクションが実行される
ポイントする	ビューのマークをポイントする。このオプションは、ダッシュボード内のハイライト アクションに対して最適に機能します。
選択する	ビュー内のマークをクリックする。このオプションは、すべてのタイプのアクションに対して有効に機能します。
メニュー	ビュー内の選択したマークを右クリック (Mac では Control を押しながらクリック) して、ツールヒント (メニュー) のオプションを選択します。このオプションは、URL アクションに対して特に有効に機能します。

6. URL ターゲットでは、リンクで以下を開く場所を指定します。
   - **Web ページ オブジェクトが存在しない場合の新しいタブ** — Web ページ オブジェクトが存在しないシート上で、その URL がブラウザで開くようにします。これは [ソース シート] が [すべて] またはデータ ソースに設定されている場合に良い選択肢となります。
   - **新しいブラウザ タブ** — デフォルトのブラウザで開きます。

- **Web ページ オブジェクト** — (Web ページ オブジェクトを含むダッシュボードでのみ使用可能) 選択した Web ページ オブジェクトで開きます。

> **注**: 新しいタブ オプションでは、アクションが初めて実行されると新しいタブが開きます。その後、アクションで追加のタブが開くのではなく、同じタブが開きます。

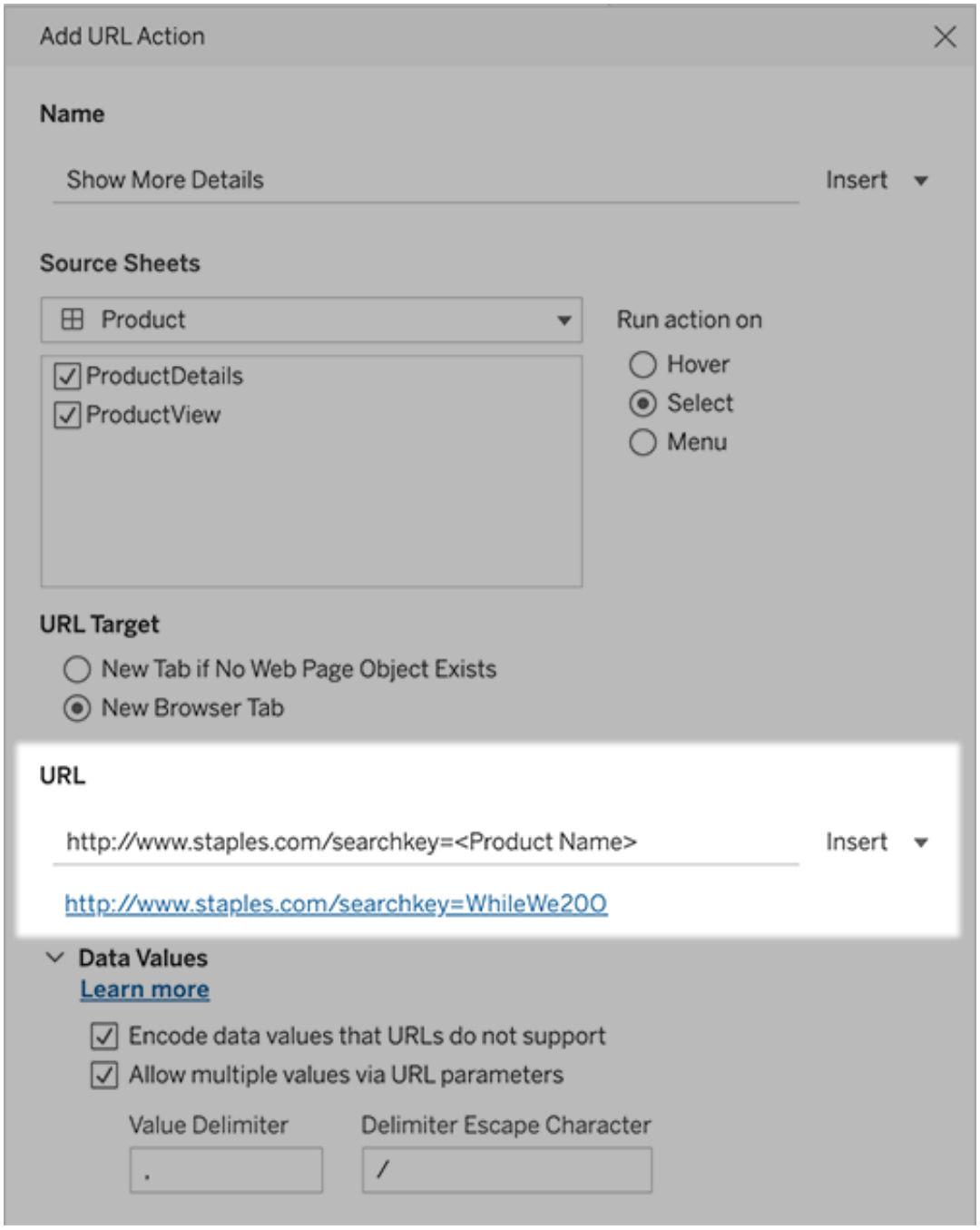
7. URL を入力します。
   - URL は次のいずれかのプレフィックスで開始する必要があります:`http`, `https`, `ftp`, `mailto`, `news`, `gopher`, `tsc`, `tsl`, `sms`, or `tel`

   > **注:** プレフィックスが入力されていない場合、先頭に `http://` が自動的に追加され、Tableau Desktop で URL アクションが機能するようになります。ただし、プレフィックスがない URL アクションが Tableau Server または Tableau Cloud にパブリッシュされると、ブラウザで URL アクションが失敗します。ダッシュボードがパブリッシュされる場合は、アクションに完全修飾 URL を常に指定してください。

   > **注**: FTP アドレスの指定は、ダッシュボードに Web オブジェクトが含まれていない場合にのみ行うことができます。Web オブジェクトが存在する場合、FTP アドレスは読み込まれません。

   - Tableau Desktop は、`C:\Example folder\example.txt` のようなローカル パスやファイル URL アクションもサポートしています。

- URL にフィールド値とフィルター値を動的な値として入力するには、URL の右側にある **[挿入]** メニューをクリックします。ビューで参照フィールドを使用する必要があることに注意してください。詳細については、URL におけるフィールド値およびフィルター値の使用を参照してください。



入力したURL の下にはハイパーリンクが付いている例があり、クリックしてテストすることができます。

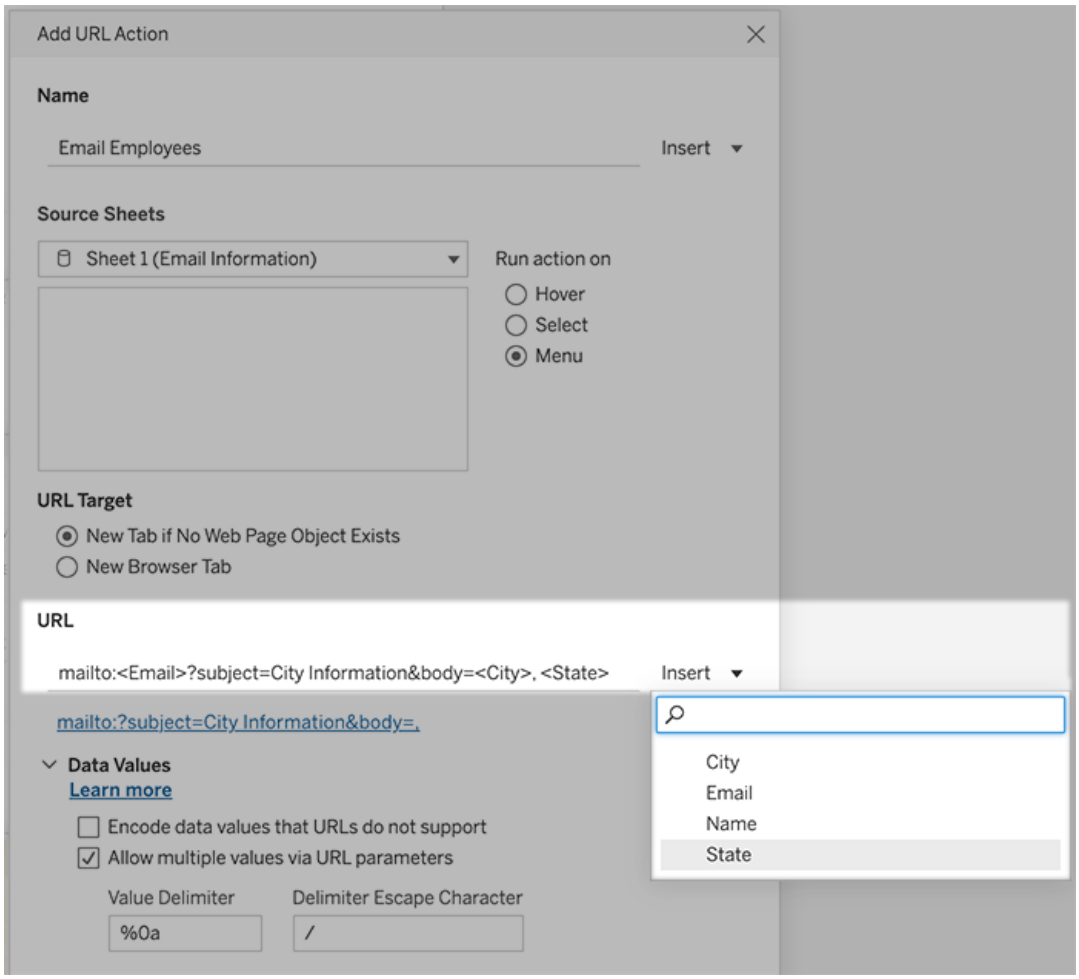
8. (オプション) [データ値] セクションで、次のオプションのいずれかを選択します。
   - **URL でサポートしていないデータ値をエンコードする -** ブラウザが URL で許可していない文字の値がデータに含まれている場合、このオプションを選択します。たとえば、"売上高 & 金融" など、データ値の 1 つにアンパサンドが含まれている場合、アンパサンドは、ブラウザが認識する文字に変換する必要があります。
   - **URL パラメーターで複数の値を許可する -** 値のリストを URL 内のパラメーター経由で受け取ることのできる Web ページにリンクしている場合、このオプションを選択します。たとえば、ビューでいくつかの製品を選択し、Web ページでホストされている各製品の詳細を表示するとします。サーバーが識別子 (製品 ID または製品名) のリストに基づいて複数の製品の詳細を読み込むことができる場合、複数選択を使用して識別子のリストをパラメーターとして送信できます。

     複数の値を許可した場合は、区切り文字となるエスケープ文字も定義する必要があります (コンマなど)。この文字は、リスト内の各項目を区切る文字であり、コンマなどが使用されます。また、区切り文字エスケープも定義する必要があります。区切り文字がデータ値で使用されている場合にこれを使用します。

URL アクションを使用したメールの作成

1. ワークシートで **[ワークシート] > [アクション]** を選択します。ダッシュボードから **[ダッシュボード] > [アクション]** を選択します。
2. [アクション] ダイアログ ボックスで、**[アクションの追加]** をクリックして、**[URL に移動]** を選択します。
3. [ソース シート] ドロップダウン リストで、送信先の電子メール アドレスが入ったフィールドが含まれるシートを選択します。
4. [URL] ボックスで、次の操作を実行します。
   - 「**mailto:**」と入力し、右側の **[挿入]** メニューをクリックして、電子メール アドレスを含むデータ フィールドを選択します。
   - 「**?subject=**」と入力し、[件名] 行のテキストを入力します。
   - 「**&body=**」と入力し、右側の **[挿入]** メニューをクリックして、メールの本文に含める情報のフィールドを選択します。

次の例では、"Email (メール)" フィールドにメール アドレスが含まれ、件名は "City Information (市区町村の情報)" であり、メールの本文はメール アドレスに関連した市区町村と州のデータで構成されています。



:

5. (オプション) ワークブックからのデータを、メールの本文で既定の横方向リストの代わりに縦方向リストとして表示できます。たとえば、"シカゴ、パリ、バルセロナ" という横方向リストを、次のように縦方向に表示すると想定します。

   シカゴ

   パリ

   バルセロナ

リストを垂直方向にするには、[データ値] セクションで次を実行します。

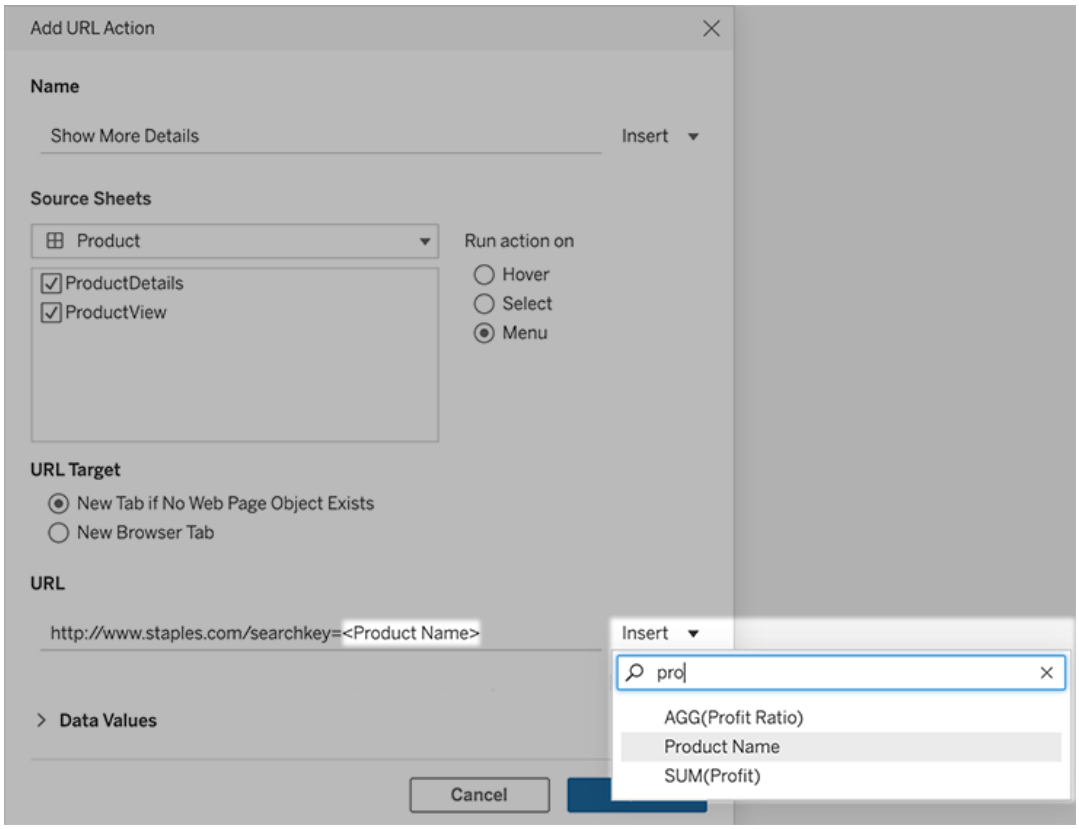
- **[URL でサポートしていないデータ値をエンコードする]** の選択をクリアします
- **[URL パラメーターで複数の値を許可する]** を選択します。
- **[値の区切り文字]** テキストボックスに「**%0a**」と入力して、リスト内の各項目の間に改行を追加します(これは改行を表す URL エンコード文字です)。

URL におけるフィールド値およびフィルター値の使用

ユーザーが選択したマークから URL アクションをトリガーすると、Tableau は フィールド値、フィルター値、およびパラメーター値を URL 内の変数として送信できます。たとえば、URL アクションがマッピング Web サイトにリンクしている場合、アドレス フィールドを挿入して、現在選択されているアドレスを Web サイトで自動的に開くことができます。

1. [URL アクションの編集] ダイアログ ボックスで、リンクの URL を入力します。
2. フィールド値、パラメーター値、またはフィルター値を挿入する場所にカーソルを置きます。
3. テキスト ボックスの右側にある **[挿入]** メニューをクリックして、挿入するフィールド、パラメーター、フィルターなどを選択します。変数は山括弧内に表示されます。必要な数だけ変数を追加できます。

> **注:** ビューで参照フィールドを使用する必要があります。そうでない場合、[リンクのテスト] をクリックしたときにリンクが機能したとしても、リンクが Viz に表示されません。



集計フィールドを含める

使用可能なフィールドのリストには、非集計フィールドのみが含まれます。集計フィールド値をリンクパラメーターとして使用するには、まず関連する計算フィールドを作成し、そのフィールドをビューに追加します(ビジュアライゼーションに計算フィールドが必要ない場合は、計算フィールドを[マーク] カードの [詳細] にドラッグします)。

パラメーター値を挿入する

パラメーター値を挿入すると、URL アクションは既定で [表示名] 値を送信します。代わりに、実際の値を送信するには、パラメーター名の後に文字 `~na` を追加します。

たとえば、IP アドレスを含むパラメーターがあり、その実際の値の文字列が 10.1.1.195 で、表示名の文字列が `Computer A (10.1.1.195)` などのわかりやすい値だとします。実際の値を送信するには、URL のパラメーターを `http://<IPAddress~na>/page.htm` のように修正します。

ごみ箱

ごみ箱を使用すると、サイト管理者、サーバー管理者、Explorer (パブリッシュ可能) または Creator ライセンスを持つ作成者は、以前に削除したプロジェクト、ワークブック、データソース、およびフローを復元できます。

ごみ箱には Tableau サイトのホームページからアクセスできます。サイドペインを展開し、**[ごみ箱]** を選択します。ごみ箱が表示されない場合は、管理者がこの機能をオンにする必要があります。

重要: Tableau Cloud バージョン 2025.1 のごみ箱のリリースに伴い、テクニカル サポート チームは、誤って削除されたコンテンツの復元に関するサポートのリクエストを受け付けなくなりました。これには、ごみ箱から削除されたコンテンツも含まれます。

ごみ箱にはどのようなコンテンツが保存されますか?

ごみ箱がオンになっている場合、次のコンテンツ タイプは削除後に一時的にごみ箱に移動されます。

- ワークブック (変更履歴、ビュー、埋め込みの認証資格情報、埋め込みデータソース、抽出を含む)
- データソース (抽出および埋め込みの認証資格情報を含む)
- プロジェクト (ワークブックやデータソースを含む)
- Tableau Prep フロー (注: Tableau Prep フローの下書きは完全に削除され、ごみ箱には移動しません)

ただし、ワークブック、データソース、プロジェクトを削除して復元すると、元のコンテンツの以下の要素が失われます。

- カスタム ビュー
- サブスクリプション
- データ主導アラート
- パーミッション

- 削除されたアイテムに関する集計統計 (ビューおよび使用量カウンター)
- タグ
- お気に入り情報 (アイテムがお気に入りに登録されている場合)
- コメント
- フローのスケジュール
- 抽出更新スケジュール

Tip: ごみ箱に入っているはずのものが見つからない場合は、正しい **コンテンツ タイプ** を選択していることを確認します。

コンテンツ: 注意事項とベストプラクティス

- サポートされているコンテンツ タイプの場合、コンテンツが削除された後も元の名前が保持されます。復元先のプロジェクトに同じ名前のアイテムが既に存在する場合、復元アクションは失敗します。この問題を解決するには、別のプロジェクトに復元し、復元したコンテンツの名前を変更して、コンテンツを目的の場所に移動します。
- ごみ箱に保存されたワークブックとデータ ソースは、引き続きサイトのスペースを消費します。ワークブックまたはデータ ソースで抽出が使用されている場合、その抽出のサイズはサイトのクォータに対してカウントされます。
- サポートされているコンテンツ タイプの場合、コンテンツが削除された後も所有者は変わりません。元々コンテンツを所有していたユーザーが引き続き所有者とみなされます。
- コンテンツを復元すると、復元されたアイテムは、復元先のコンテナに設定された既定のアクセス許可を取得します。重要なコンテンツを復元する場合、操作ユーザーは必要に応じて権限が設定されていることを確認する必要があります。復元操作では、元のコンテンツの権限は復元されません。

プロジェクト: 注意事項とベストプラクティス

プロジェクトには、さまざまなコンテンツやネストされたプロジェクトを含めることができます。ごみ箱がオンになっている場合のプロジェクトでは、次の点に注意してください。

注: ごみ箱では、元の削除アクションが実行されたときとコンテンツの復元時に、ネストされたプロジェクトの階層が考慮されます。たとえば、ルートプロジェクトからプロジェクト分岐全体を削除して、ルートとすべてのネストされたプロジェクトをごみ箱に送信し、その後、ルートプロジェクトを復元すると(ネストされたプロジェクトはすべてごみ箱に残っています)、プロジェクト分岐全体をごみ箱から復元できます。ただし、ごみ箱からコンテンツを削除するときに階層は考慮されません。ごみ箱から元のプロジェクト分岐のすべてのプロジェクトを削除するには、各プロジェクトを個別に削除する必要があります。

プロジェクトの例: プロジェクトA には、2 つのワークブック(A とB)、1 つのデータソース、1 つのフローが含まれています。

- プロジェクトが最初に削除され、ごみ箱に移動すると、ごみ箱と互換性のあるそのプロジェクトのすべてのコンテンツが個別に保存されます。プロジェクトA の場合、ごみ箱にはプロジェクト、ワークブックA、ワークブックB、データソースが表示されます。フローなどの他のタイプのコンテンツはすぐに削除され、復元することはできません。
- プロジェクトの互換性のあるコンテンツはすべて、個別に復元または削除できます。たとえば、プロジェクトA からワークブックA を完全に削除して、プロジェクト、ワークブックB、およびデータソースだけをごみ箱に残すことができます。
- プロジェクトを復元すると、その時点でごみ箱に保存されている互換性のあるコンテンツもすべて復元されます。プロジェクトA の場合、ワークブックA を既に削除してからプロジェクトA を復元すると、プロジェクトはワークブックB とデータソースのみを含めて復元されます。
- ごみ箱からプロジェクトを削除すると、プロジェクトのみが削除されます。たとえば、プロジェクトA をごみ箱から削除した場合、ワークブックB とデータソースは、個別に削除または復元されるか、保存期間が終了するまでごみ箱に残ります。

プロジェクト例: プロジェクトA には、ワークブックA、ワークブックB、データソース、およびワークブックC を含むネストされたプロジェクトB が含まれています。

- 削除されたプロジェクトにネストされたプロジェクトが含まれている場合、上位プロジェクトを復元すると、その時点でまだごみ箱に保存されているネストされたプロジェクトも復元されます。たとえば、ルートプロジェクトA を復元すると、上位プロジェクトA、ワークブックB、データソース、プロジェクトB、およびワークブックC が選択した復元場所に復元されます。

- ごみ箱内のネストされたプロジェクトは、上位プロジェクトから独立して復元できます。ネストされたプロジェクトを個別に復元してから上位レベルのプロジェクトを復元すると、上位レベルのプロジェクトはネストされたプロジェクトのコンテンツを含めずに復元されます。たとえば、プロジェクトB を個別に復元してからプロジェクトA を復元すると、プロジェクトA は上位プロジェクト、ワークブックB、およびデータソースを含めて復元されます。復元されたプロジェクトB はプロジェクトA 内にネストされなくなります。

プロジェクトの例プロジェクト1 (P1) は、4 つのネストされたプロジェクト(P2-P3-P4-P5) を含むプロジェクトの分岐です。

シナリオ: 最初にP1 を削除し、分岐全体 (ルートP1 とネストされたP2-P3-P4-P5) をごみ箱に送ります。

- ごみ箱から、元の階層 (P1-P2-P3-P4-P5) の個々のプロジェクトを削除できます。

- ネストされたプロジェクトをごみ箱から削除すると、元のプロジェクト階層が破壊されます。

    - P3 を削除すると、P1、P2、P4、P5 はごみ箱に残りますが、P4 は P2 に接続されなくなります。元の単一のプロジェクトの分岐は、P1-P2 とP4-P5 の 2 つの分岐に分割されました。

    - P3 を削除してからP1 を復元すると、復元アクションによってP2 も復元されますが、P4 とP5 はごみ箱に残ります。

    - P3 が削除された状態でP4 を復元すると、P4 がP5 のルートとみなされているため、復元アクションによってP5 も復元されます。

    - 元のネストされていたプロジェクトは、個別に復元できます。

- ごみ箱からプロジェクト階層全体を削除することはできません。各プロジェクトを個別に削除する必要があります。

ごみ箱からコンテンツを復元する

管理者が設定した保存期間内であれば、ごみ箱のコンテンツをプロジェクトに復元できます。

1. ホーム ページからサイドペインを展開し、**[ごみ箱]** を選択します。
2. **[コンテンツ タイプ]** で **[ワークブック]**、**[プロジェクト]**、**[データ ソース]** または **[フロー]** を選択します。
3. 復元したいコンテンツの **[その他のアクション]** メニューを選択し、**[Restore (復元)]** を選択します。
4. 復元場所としてプロジェクトを選択します。
5. **[復元]** を選択します。

ごみ箱からコンテンツを削除する

管理者が設定した保存期間が終了する前に、ごみ箱からコンテンツを完全に削除できます。このアクションは元に戻すことはできません。

1. ホーム ページからサイドペインを展開し、**[ごみ箱]** を選択します。
2. **[コンテンツ タイプ]** で **[ワークブック]**、**[プロジェクト]**、**[データ ソース]** または **[フロー]** を選択します。
3. 完全に削除したいコンテンツの **[その他のアクション]** メニューを選択し、**[削除]** を選択します。
4. **[削除]** を選択します。

ごみ箱をオンにする

管理者はサイト設定で、Tableau サイトのごみ箱をオンにすることができます。

1. [全般] タブで、[ごみ箱] セクションまでスクロールします。
2. **[Allow Recycle Bin for this site (このサイトのごみ箱を許可する)]** のチェックボックスをオンにします。
3. コンテンツをごみ箱に保存する日数を選択します。1 日、7 日、30 日のいずれかを選択できます。

   重要: 選択した日数が経過すると、コンテンツは完全に削除されます。
4. **[保存]** を選択します。

ごみ箱をオフにする

警告: 管理者がごみ箱をオフにすると、ごみ箱に保存されているコンテンツはすべて完全に削除されます。

管理者はサイト設定で、Tableau サイトのごみ箱をオフにすることができます。

1. [全般] タブで、[ごみ箱] セクションまでスクロールします。
2. **[Allow Recycle Bin for this site (このサイトのごみ箱を許可する)]** のチェックボックスをオフにします。
3. **[保存]** を選択します。

管理者に問い合わせる

ごみ箱をオンにして設定できるのは、Tableau サイト管理者またはサーバー管理者のみです。管理者の連絡先情報を確認するには、ホームページから **[Show help menu (ヘルプメニューを表示)]** アイコンを選択し、**[Contact Tableau Administrator (Tableau 管理者に問い合わせる)]** を選択します。

ビューまたはワークブックへのサブスクリプションの作成

サブスクリプションにより、ビューやワークブックのイメージまたは PDF スナップショットが定期的にメールで送信されます。Tableau Server にサインインする必要はありません。

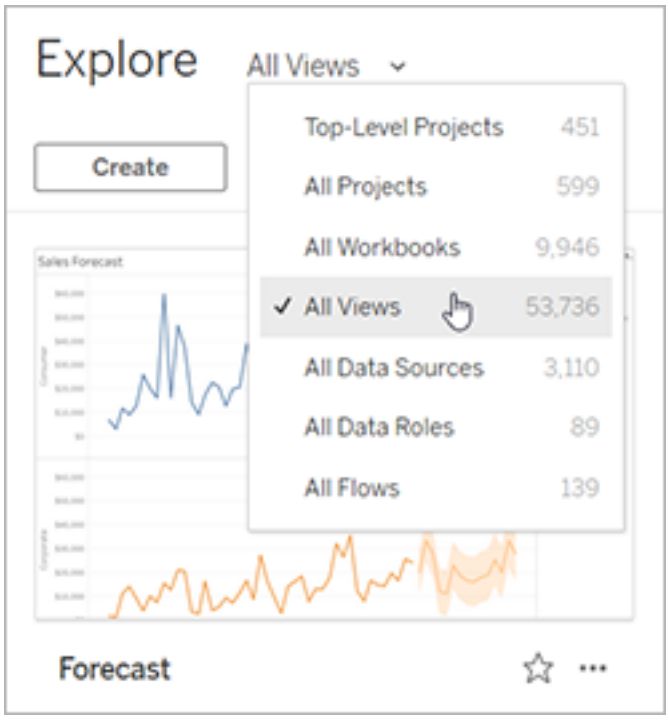
注: 管理者が、サブスクリプションをサイトに対して有効にするかどうかを判断します。

Tableau Catalog がサイトに対して有効になっている場合、管理者は関連するアップストリームのデータ品質に関する警告をサブスクリプション メールに含めるかどうかも判断することができます。Tableau Catalog は データ管理 製品に含まれています。詳細については、「Tableau Catalog について」を参照してください。

自分または他のユーザー向けのサブスクリプションの設定

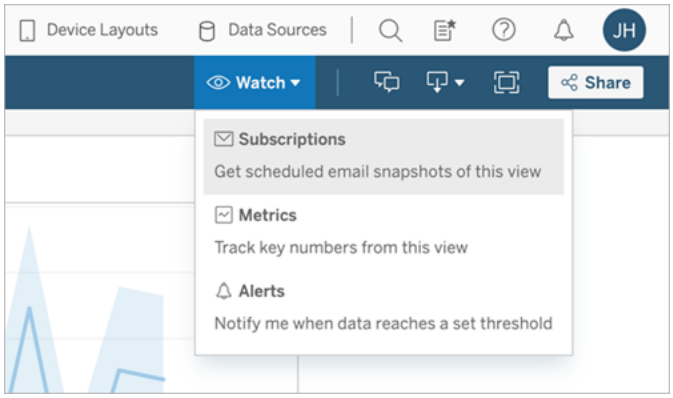
ビューを Tableau Server で開いたときに、ツールバーにサブスクリプション アイコン (✉+) が表示される場合は、そのビューまたはワークブック全体をサブスクライブできます。ワークブックを所有している場合、適切なサイト ロールを持つプロジェクト リーダーの場合、または管理者の場合は、コンテンツを表示するパーミッションを持つ他のユーザーをサブスクライブできます。

1. サイトの [検索] セクションから、**[すべてのワークブック]** または **[すべてのビュー]** を選択するか、サブスクライブするビューが含まれるプロジェクトを開きます。



2. ビューを直接、または含まれているワークブックを開いてから開きます。ビューに適用するフィルターは、サブスクリプションには反映されません。フィルタリングされたビューにユーザーをサブスクライブするには、それらのフィルターを含むカスタム ビューを保存してから、ユーザーをそのビューにサブスクライブします。カスタム ビューの作成の詳細については、「カスタム ビューの使用」を参照してください。

3. ビューのツールバーで、**[Watch (視聴)] > [サブスクリプション]** を選択します。



4. サブスクリプションを受信するTableau ユーザーまたはグループを追加します。サブスクリプションを受け取るには、ユーザーは画像/PDF を表示およびダウンロードするための権限を持っている必要があり、アカウントには メール アドレスも必要です。

   ワークブックを所有している場合は、**[自分をサブスクライブする]** を選択します。

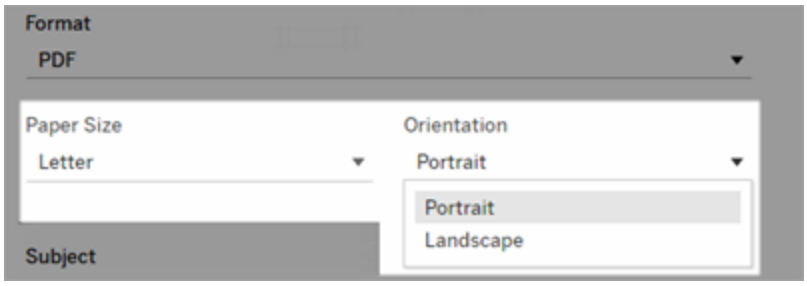
   注:

   - グループをサブスクライブする場合、各ユーザーはサブスクリプションの作成時に個別に追加されます。もっと多くのユーザーを後でグループに追加する場合は、サブスクリプションを受信するために、それらの新しいユーザーでグループを再度サブスクライブする必要があります。同様に、グループから後で削除されるユーザーでは、サブスクライブしているビューへのパーミッションが削除されない限り、サブスクリプションは自動的に削除されません。
   - グループセットをサブスクライブすることはできません。

5. サブスクリプション メールに現在のビューを含めるか、ワークブック全体を含めるかを選択します。高優先度の情報が存在するときにのみビューにデータを含める場合は、**[ビューが空の**

時は送信しない] を選択します。

6. スナップショットの形式 (PNG 画像、PDF 添付ファイル、またはその両方) を選択します。
   - PDF の場合は、出力する用紙サイズと向きを選択します。



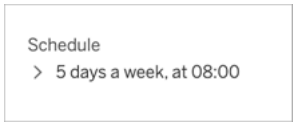
7. サブスクリプション メールをわかりやすくするために、件名行をカスタマイズしてメッセージを追加します。

> **注:** サブスクリプション メッセージを更新するには、既存のサブスクリプションの登録を解除し、別のメッセージを使用して新しいサブスクリプションを作成する必要があります。詳細については、「サブスクリプションの更新またはサブスクライブ解除」を参照してください。

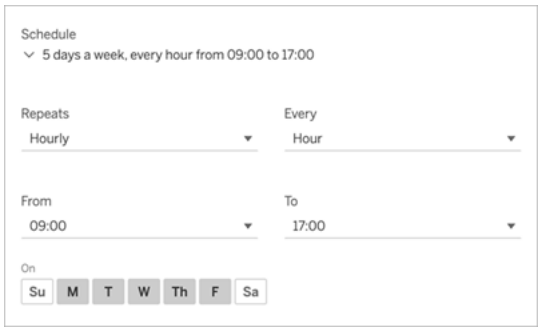
8. ワークブックでパブリッシュされた接続で 1 つのデータ抽出を使用する場合、その頻度を選択できます。
   - **データ更新時**: ビューまたはワークブックのデータが、更新スケジュールの実行によって更新されたときにのみ送信されます。
   - **選択したスケジュール**: サブスクリプションのスケジュールを選択します。

9. 頻度が [When Data Refreshes (データ更新時)] に設定されていない場合は、スケジュールを選択します。

- 管理者が確立したサブスクリプション スケジュールから選択します。
- カスタム スケジュールが有効になっているサイトの場合は、現在の設定の右にあるドロップダウン矢印をクリックします。



サブスクリプション メールをいつでも好きなときに送信できるカスタム スケジュールを指定します。(サーバー負荷が高い場合は、厳密な配信時刻は異なる場合があります)。



タイム ゾーンを変更するには、「タイム ゾーン」リンクをクリックしてアカウント設定ページに移動してください。

10. **[サブスクライブ]** をクリックします。

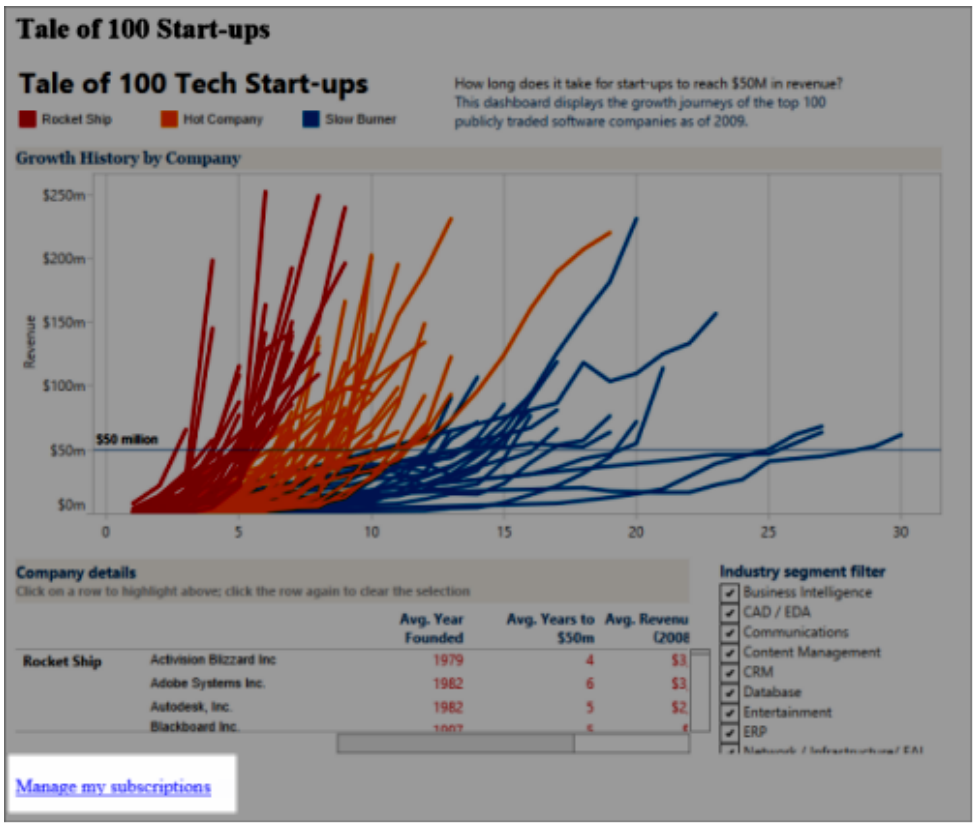
サブスクリプション メールを受信し、イメージ (または PDF サブスクリプションのメッセージ本文に含まれるリンク) を選択すると、Tableau Server のビューまたはワークブックに移動することができます。

サブスクリプションの更新またはサブスクライブ解除

既存サブスクリプションをサブスクライブ解除したり、サブスクリプションの形式、スケジュール、件名、または空のビュー モードを変更することができます。

1. 次のいずれかの方法を使用して、Tableau Server のアカウント設定にアクセスします。

   - サブスクリプション メールの下にある **[自分のサブスクリプションの管理]** をクリックします。



   - Tableau Server にサインインします。ページ上部で自分のユーザー アイコンを選択

してから、**[マイ コンテンツ]** を選択します。

2. **[サブスクリプション]** をクリックします。

3. サブスクライブを解除するビューの横にあるチェックボックスを選択し、**[アクション]**、**[サブスクライブ解除]** の順にクリックするか、変更したいサブスクリプション オプションを選択します。

一時停止したサブスクリプションの再開または削除

ブックに問題があるか、ビューの読み込みで問題が発生しているために、サブスクリプションが失敗する場合があります。サブスクリプションの失敗が 5 回を超えると、サブスクリプションが一時停止した旨の通知メールを受信します。サブスクリプション所有者または管理者は、いくつかの方法で、一時停止したサブスクリプションを再開することができます。

- Tableau Web ページの [マイ コンテンツ] エリアの [最終更新] 列に、サブスクリプションが一時停止していることを示すアイコンが表示されます。**[...]> [サブスクリプションの再開]** を選択して再開します。

- 影響が出ているワークブックの [サブスクリプション] タブで、[前回更新] 列に、サブスクリプションが一時停止になっていることを示すアイコンが表示されます。**[...]> [サブスクリプションの再開]** を選択して再開します。

サブスクリプションが再び機能している場合、通知メールが送信されます。

関連項目

Tableau Desktop とWeb 作成のヘルプの「サブスクリプション設定の変更」。

Tableau Cloud ヘルプの「プロジェクトレベルの管理」では、どのサイトロールで完全なプロジェクトリーダー権限が許可されるかを確認できます。

カスタム ビューの使用

カスタム ビューは、フィルターの選択や並べ替えなど、公開済み Viz に対する特定の操作状態へのショートカットです。カスタム ビューは、参照元のコンテンツには影響を与えません。Viz を見るたびに同じフィルターを設定したり、同じデータに注目して閲覧したりするのであれば、カスタム ビューは良い選択肢になります。

カスタム ビューは、Web 編集とは異なり、参照元の公開済みコンテンツ自体に変更を加えません。「Web での Tableau ビューの編集」を参照してください。

カスタム ビューがフィルター設定専用である場合は、共有 URL にフィルター パラメーターを埋め込むことを検討してください。「データスクール」の「URL を編集して公開済みダッシュボードをフィルタリングする」を参照してください。

カスタム ビューに関する注意事項

- カスタム ビューは、元のコンテンツを変更しません。
- 元のコンテンツを削除すると、そのカスタム ビューも削除されます。
- 元のコンテンツが更新または再パブリッシュされると、カスタム ビューも更新されます。

  ヒント: 元のコンテンツに変更を加えると、カスタム ビューが壊れる可能性があります。カスタム ビューを使用したコンテンツの変更に関するベストプラクティスについては、「カスタム ビューを使用したコンテンツの維持」を参照してください。

- ユーザーがサイトから削除されると、ユーザーが所有していた共有カスタム ビューも失われます。
- カスタム ビューに基づくサブスクリプションとデータドリブンなアラートは、元のコンテンツに基づくサブスクリプションよりも脆弱になる可能性があります。

注: 2022.3 リリースから、Tableau はカスタム ビュー URL のユーザー名を ID に置き換えました。ブックマークされた URL は引き続き機能しますが、新しい URL スキーマにリダイレクトされます。今回の変更は、企業やユーザーのデータ保護をより強化するためのものです。

カスタム ビューの作成

まず、個別のビューに移動します。マークの選択、データのフィルタリング、並べ替えの変更など、カスタム ビューで取り込みたい変更を加えます。

1. 変更内容をカスタム ビューとして保存する準備ができたら、ツールバーから**[カスタム ビューの保存]**を選択します。

   > **注**: 現在のビューに変更を加えると、**[カスタム ビューの保存]** ボタンがツールバーに表示されます。

2. **[カスタム ビューの保存]** ダイアログで、カスタム ビューの名前を入力します。
3. (オプション) **[デフォルトにする]** を選択します。
4. (オプション) **[他の人に見えるようにする]** を選択します。これにより、元のコンテンツを表示できるすべてのユーザーがカスタム ビューを利用できるようになります。しかし、以下のように、このオプションが使えない場合もあります。
   - ユーザーのサイト ロールが Viewer である。
   - サイトの [ユーザーの表示状態] が [Limited (制限)] に設定されている。
   - パーミッションの機能 [カスタマイズの共有] がワークブックで拒否されました。
5. **[保存]** をクリックします。

カスタム ビューの検索

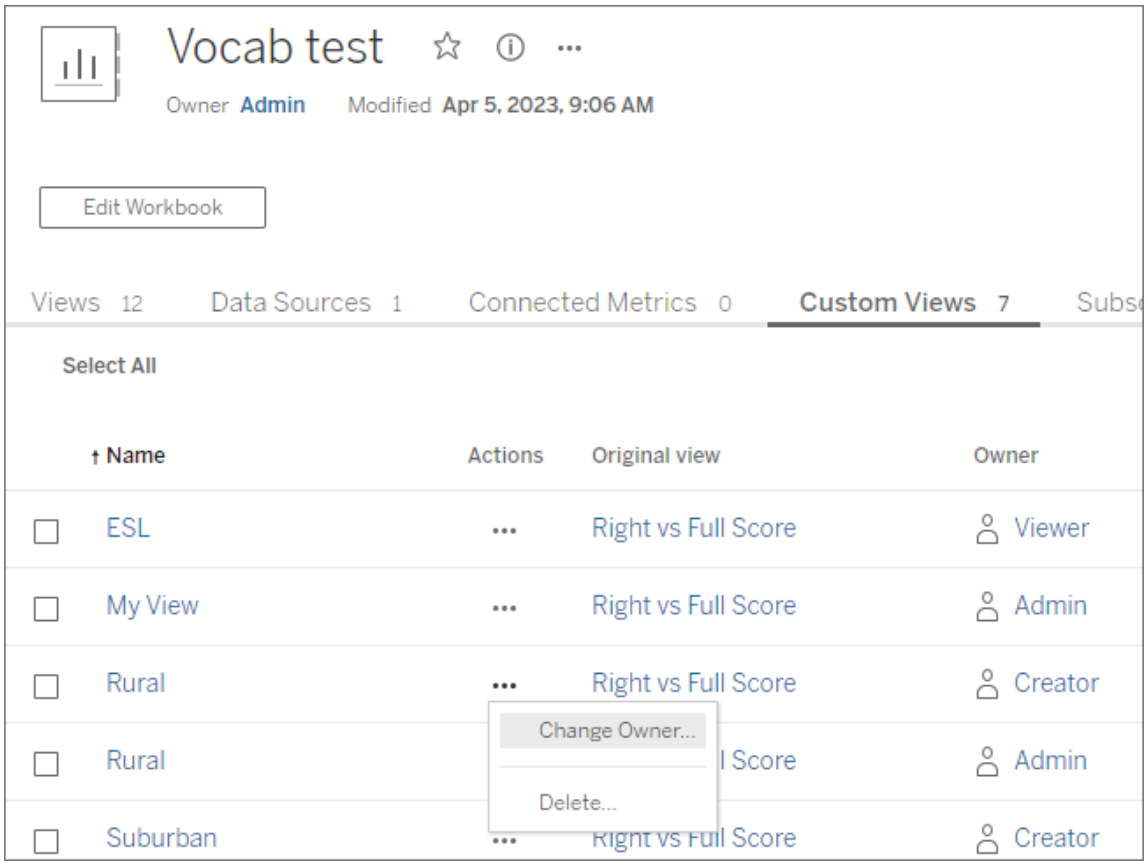
ビューから

Viz を表示しているときに、ツールバーの [表示] アイコンを選択すると、別のカスタム ビューに変更できます。ツールバーに空きがある場合は、閲覧しているカスタム ビューの名前が表示されます。

自分が作成したすべてのカスタム ビューと、他のユーザーが作成したカスタム ビューのうち表示できるものが、リストに表示されます。

ワークブックから

ワークブック レベルでコンテンツを表示している場合は、**[カスタム ビュー]**タブを使用すると、そのワークブックで使用可能なすべてのカスタム ビューを表示できます。



デフォルトのカスタム ビューの設定

検索したり作成したりしたカスタム ビューは、Viz を開いたときに表示されるデフォルトのビューに設定できます。

1. ツールバーの **[表示]** アイコンを選択します。
2. **[Set this view a your default (このビューを既定に設定する)]** オプションをオンにします。
3. ダイアログを閉じて保存します。

次回その Viz を開くと、そのカスタム ビューが表示されます。

カスタム ビューの共有

デフォルトでは、カスタム ビューはプライベートであり、カスタム ビューを作成したユーザーにのみ表示されます。

> **注**: Viewer のサイトロールを持つユーザーは、カスタム ビューを他のユーザーに表示させることはできません。ただし、URL をコピーして共有することで、カスタム ビューを共有できます。

Explorer 以上のサイトロールを持つユーザーは、カスタム ビューを他のユーザーに表示するように設定できます。この設定により、元のコンテンツにアクセスできるすべてのユーザーがそのカスタム ビューを表示できるようになります。

既存のプライベート カスタム ビューを他のユーザーに表示されるように変更するには (または表示されているビューをプライベートにするには)、次の操作を行います。

1. ツールバーの **[表示]** アイコンを選択します。
2. 共有するビューの目のアイコンを目的の状態に切り替えます。
3. ダイアログを閉じて保存します。

スラッシュのある目 は、ビューがプライベートであることを示します。目のアイコン は、ビューが他のユーザーに表示できることを示します。

カスタム ビューの削除

カスタム ビューを削除するには、次の操作を行います。

1. ツールバーの **[表示]** アイコンを選択します。
2. 削除するビューのゴミ箱アイコンを選択します。
3. ビューを削除することを確認します。

削除時の注意

他のユーザーに表示されるカスタム ビューの所有者である場合は、カスタム ビューを削除すると、すべてのユーザーに対して削除されることに注意が必要です。

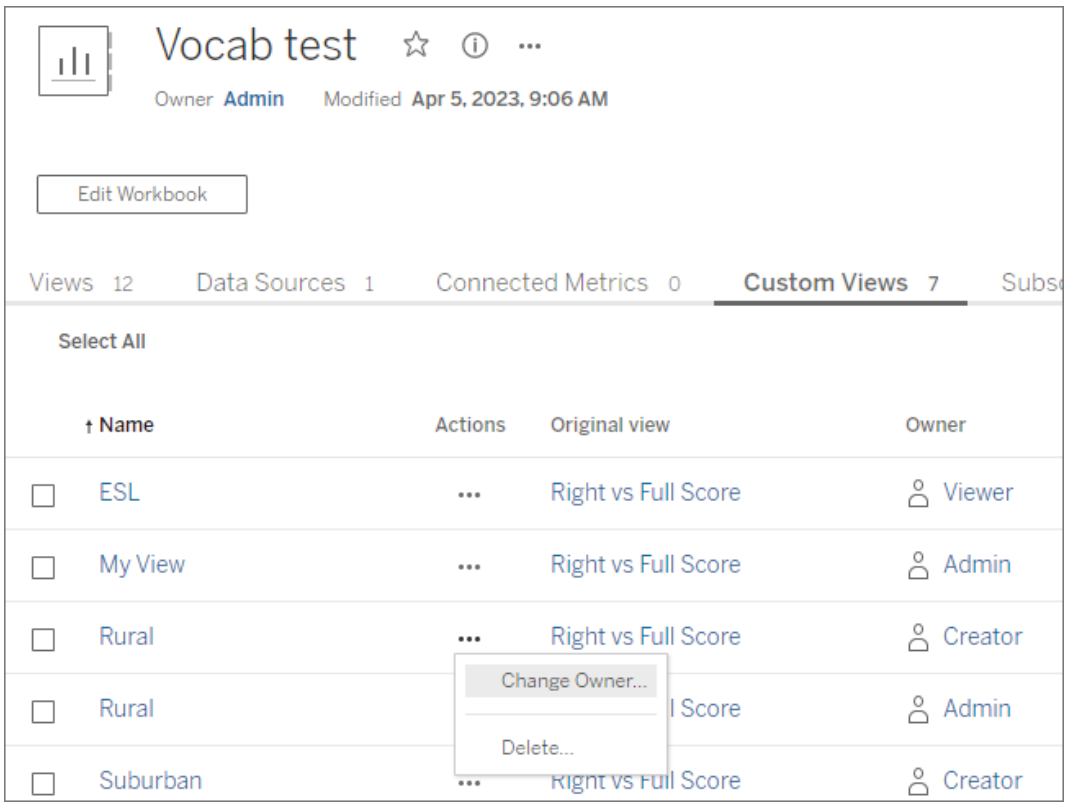
カスタム ビューを削除すると、そのカスタム ビューに基づくサブスクリプションやデータドリブンなアラートも削除されます。

カスタム ビューの管理

管理者は、カスタム ビューのオーナーシップを変更したり、他のユーザーが作成したカスタム ビューを削除したりできます。

カスタム ビューは、コンテンツまたは特定のユーザーに対して管理できます。

1. ワークブックまたはユーザーの [カスタム ビュー] タブに移動します。
2. アクション メニューを使用して、所有者を変更するか、カスタム ビューを削除します。



ヒント: ユーザーに属するカスタム ビューをサイトから削除する前に、カスタム ビューの所有権を変更することをお勧めします。ユーザーを削除すると、他のユーザーが使用しているパブリック ビューを含むカスタム ビューも削除されます。

カスタム ビューを含むコンテンツの安全な変更

カスタム ビューを含むビュー (またはビューの基となっているデータ ソース) を変更する必要がある場合、変更の内容によってはカスタム ビューが壊れる可能性があることに注意が必要です。詳細に

ついては、「カスタムビューを含むコンテンツの維持」を参照してください。

ビューを Salesforce にパブリッシュする

ビューを CRM Analytics アプリまたは Salesforce Lightning ページにパブリッシュすることにより、ビューを Tableau Cloud または Tableau Server から Salesforce エコシステムに直接取り込むことができます。

詳細については、Salesforce ヘルプの「Tableau コンテンツを CRM Analytics にパブリッシュする」を参照してください。

前提条件

必要なライセンス、アカウント設定、パーミッションなど、前提条件の完全なリストについては、Salesforce ヘルプを参照してください。

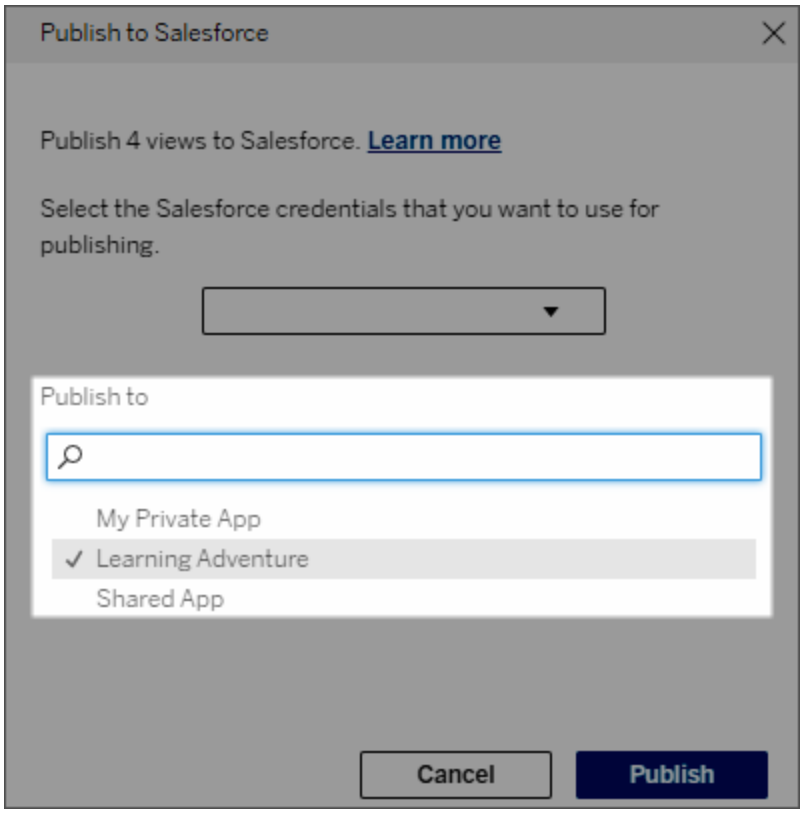
ビューを Salesforce にパブリッシュする

ダッシュボード、シート、ストーリーなど、1 つ以上のビューを選択します。次に、編集または管理するアクセス権を持つ CRM Analytics アプリのリストから宛先を選択します。

1. Salesforce にパブリッシュするビューを選択します。

   > **注:** Salesforce にパブリッシュするビューは、一度に最大 25 個まで選択できます。

2. **[アクション]** で、**[Publish to Salesforce (Salesforce にパブリッシュ)]** を選択します。
3. パブリッシュに使用する Salesforce 認証資格情報を選択します。
4. パブリッシュする宛先アプリを選択します。表示できるのは、Salesforce ユーザーとしてサインインして編集または管理できるアプリのみです。



5. **[パブリッシュ]** をクリックします。

Salesforce にパブリッシュしたビューを表示できるユーザー

ビューを Salesforce にパブリッシュすると、選択した CRM Analytics アプリまたは Lightning ページにアクセスできるユーザーは誰でもコンテンツが存在することを確認できます。ただし、ビューを表示できるのは、既存の Tableau パーミッションを使用してサインインしているユーザーのみです。

Tableau Lightning Web コンポーネントとトークン認証によるシングル サインオン (SSO) の設定

Tableau Lightning Web コンポーネント (LWC) を使用すると、Salesforce の顧客は、Tableau ビューと Tableau Pulse メトリクスを Salesforce Lightning ページにドラッグ アンド ドロップできます。

- Tableau ビュー コンポーネントを使用すると、Tableau Cloud または Tableau Server から埋め込みビューを追加できます。

- Tableau Pulse コンポーネントを使用すると、Tableau Cloud から Tableau Pulse 埋め込みメトリクスを追加できます。

Tableau LWC のシームレスな認証を使用すると、サインインせずに、接続済みアプリの信頼できるトークンを使用して Tableau コンテンツを表示できます。シームレスな認証は、Tableau ビュー LWC ではオプションですが、Tableau Pulse LWC では必須です。

重要:
• Salesforce コンソール アプリは Tableau Lightning Web コンポーネントの使用をサポートしていません。
• ケース レコード ページでは、Chatter メールでの Tableau Lightning Web コンポーネントの使用をサポートしていません。

信頼できる URL を追加する

Lightning ページに追加する Tableau ビューまたは Pulse URL は、信頼できる URL として追加する必要があります。

1. Salesforce アプリで、右上隅にある歯車を選択し、**[設定]** を選択します。
2. 左側のナビゲーション ペインで、**[Quick Find (クイック検索)]** 検索バーに「Trusted URLs (信頼できる URL)」と入力します。
3. **[Trusted URLs (信頼できる URL)]** 設定ページを選択します。
4. **[New Trusted URL (新しい信頼できる URL)]** を選択します。
5. 設定ページの指示に従って、**[API 名]** と **[URL]** を入力します。**注**: URL は https:// で始まる必要があります。
6. [CSP Context (CSP コンテキスト)] では、**[すべて]** を選択します。
7. [CSP Directives (CSP ディレクティブ)] では、すべてのボックスをオンにします。
8. **[保存]** を選択します。

Tableau LWC のシームレス認証を有効にする

Tableau View および Tableau Pulse LWC は、設定なしで Lightning App Builder で使用できます。ただし、Tableau Pulse LWC が機能するにはトークン認証が必要です。

Salesforce 設定を構成する

次の手順は、Salesforce 管理者が 1 回だけ完了する必要があります。

1. Salesforce アプリで、右上隅にある歯車を選択し、**[設定]** を選択します。
2. 左側のナビゲーション ペインで、**[Quick Find (クイック検索)]** 検索バーに「Tableau」と入力します。
3. **[Tableau Embedding (Tableau 埋め込み)]** 設定ページを選択します。
4. **[トークンベースのシングル サインオン認証を有効にする]** のチェックボックスを選択してください。

   > **注**: Tableau Pulse LWC を設定するには、このボックスをオンにする必要があります。Tableau View LWC の場合、トークン認証を設定せず、代わりにコンポーネントの読み込み時に手動でサインインすることを選択できます。
   >
   > Tableau View LWC をモバイルで使用するには、トークンベースの認証をオンにし、シームレスな認証を設定する必要があります。

5. **[Select Tableau User Identity field (Tableau ユーザー アイデンティティ フィールドの選択)]** では、組織レベルのユーザー フィールドを設定して、Tableau でユーザーを認証できるようにします。Tableau ユーザー名に対応する Salesforce ユーザー フィールドを選択する必要があります。ドロップダウンには現在のユーザーのフィールド値が表示され、値が定義されていない場合は Null が表示されます。Tableau ユーザー名と一致するユーザー フィールドがない場合は、空のフィールド (フェデレーション ID やカスタム フィールドなど) を選択します。次に、空のフィールドにユーザーの Tableau ユーザー名を入力します。

   > **注**: Tableau ユーザー アイデンティティ フィールドの設定はすべてのユーザーに適用され、個別に設定する必要はありません。

6. 変更内容を保存します。
7. Tableau 管理者の場合は、次のセクションで Tableau 設定を構成する間、Salesforce 設定の **[Tableau Embedding (Tableau 埋め込み)]** タブを開いたままにしておきます。管理者ではない場合は、**発行 URL** と **JWKS URI** を Tableau 管理者と共有します。

ヒント: Tableau ユーザー アイデンティティ フィールドの設定はすべてのユーザーに適用され、個別に設定する必要はありません。

1 つのタブで、Salesforce 組織の [Tableau Embedding (Tableau 埋め込み)] 設定ページを開きます。別のタブで、Tableau サイトに移動し、次の手順に従って接続済みアプリを設定します。

Tableau Server の場合は、以下の手順を実行します。

1. Tableau Server 管理者として、Tableau サービス マネージャー (TSM) の Web インターフェイスにサインインします。
2. **[ユーザー ID とアクセス]** に移動し、**[認可サーバー]** タブを選択します。
3. **[埋め込みコンテンツの OAuth アクセスを有効にする]** チェックボックスをオンにします。
4. **[発行者 URL]** と **[JWKS URI]** を入力します。これらは、Salesforce 組織の [Tableau 埋め込み] 設定ページにあります。Salesforce 組織の設定ページにある **[コピー]** ボタンを使用して、発行者 URL の値をコピーし、それを TSM Web インターフェイスに貼り付けます。JWKS URI の値に対してもこのプロセスを繰り返します。

   > **注**: TSM Web インターフェイスでは JWKS URI フィールドはオプションとしてマークされていますが、この値は Tableau LWC のシームレスな認証を使用するために必要です。シームレスな認証を成功させるには、Tableau Server が JWKS URI に到達できる必要があります。

5. **[保留中の変更を保存]** を選択します。
6. ページの右上隅にある **[保留中の変更]** を選択し、**[変更を適用して再起動]** を選択して、Tableau Server を停止してから再起動します。

詳細については、「EAS を Tableau Server に登録する」を参照してください。

Tableau Cloud の場合は、以下の手順を実行します。

1. Tableau の **[設定]** ページを開き、**[接続済みアプリ]** タブを選択します。
2. [新しい接続済みアプリ] のドロップダウンから **[OAuth 2.0 信頼]** を選択します。
3. [接続済みアプリの作成] ダイアログで、**[発行者 URL]** と **[JWKS URI]** を入力します。これらの URL と URI は、Salesforce 組織の [Tableau ビュー埋め込み] 設定ページにあります。Salesforce 組織の設定ページにある **[コピー]** ボタンを使用して、発行元 URL の値をコピーし、それを Tableau 設定ページに貼り付けます。JWKS URI の値に対してもこのプロセスを繰り返します。

重要: [接続アプリの作成] ダイアログでは、JWKS URI フィールドがオプションとして示されていますが、この値は Tableau LWC のシームレスな認証を使用するために必要です。

4. **[接続済みアプリを有効にする]** のチェックボックスをオンにします。
5. **[作成]** を選択します。

> **注:** 接続済みアプリには、外部認証サーバーという名前が付けられます。

詳細については、「EAS を Tableau Cloud に登録する」を参照してください。

このサイトのホストマッピングを作成する場合は、[接続済みアプリ] タブを開いたままにしておきます。このページの URL と[サイト ID のコピー] ボタンを使用して、次のセクションのホストマッピングフィールドに値を入力できます。

ユーザー属性の取得を許可する (Tableau Cloud のみ)

Tableau LWC でユーザー属性関数を活用する場合は、認証ワークフローのアクセスを有効にする必要があります。

1. Tableau の **[設定]** ページを開き、**[認証]** タブを選択します。
2. **[Control User Access in Authentication Workflows (認証ワークフローのユーザーアクセス制御)]** セクションに移動します。
3. **[Enable capture of user attributes in authentication workflows (認証ワークフローでユーザー属性の取得を有効にする)]** チェックボックスをオンにします。

ホストマッピングを設定または編集する

ホストマッピングを作成または編集するには、次の手順に従います。

新しいホストマッピングを作成する

1. Salesforce アプリで、右上隅にある歯車を選択し、**[設定]** を選択します。
2. 左側のナビゲーションペインで、**[Quick Find (クイック検索)]** 検索バーに「Tableau」と入力します。
3. **[Tableau Embedding (Tableau 埋め込み)]** 設定ページを選択します。

4. [Tableau Host Mapping (Tableau ホストマッピング)] セクションで、**[Create New (新規作成)]** を選択します。
5. ホストマッピングの詳細を入力します。
   a. Tableau サイトURL: マッピングするTableau サイトの URL を入力します。既定のサイトを使用するオンプレミスインストールでない限り、URL にはサイト名が含まれている必要があります。**注**: このサイトのホストマッピングを作成する場合は、[接続済みアプリ] タブを開いたままにしておきます。このページの URL と[サイトID のコピー] ボタンを使用して、次のセクションのホストマッピング フィールドに値を入力できます。
   b. Tableau サイトID: マッピングするTableau サイトのサイトID を入力します。[接続済みアプリ] 設定ページまたは [共有] ダイアログで [サイトID のコピー] ボタンを使用できます。
   c. Tableau サイトホストタイプ: **Tableau Cloud**、**Tableau Server (サイトレベルの EAS)**、**Tableau Server (サーバー全体の EAS)** のいずれかを選択します。
6. **[保存]** を選択します。または、保存せずに [Tableau Embedding (Tableau 埋め込み)] 設定ページに戻る場合は、**[キャンセル]** を選択します。

ホストマッピングを編集する

> **注**: バージョン 254.10 より前のホストマッピングを保存している場合は、Tableau ホストのタイプを更新する必要があります。詳細情報と修復手順については、「保存された Tableau ホストマッピングのホストタイプを更新する」を参照してください。

既存のマッピングのサイトID とホストタイプを更新できます。サイトの URL を変更する必要がある場合は、既存のマッピングを削除し、正しい URL で新しいマッピングを作成します。

1. Salesforce アプリの [Tableau Embeddings (Tableau 埋め込み)] 設定ページで、既存のホストマッピングの横にある [編集] を選択します。
2. 必要に応じて、**[Tableau site ID (Tableau サイトID)]** フィールドまたは **[Tableau site host type (Tableau サイトのホストタイプ)]** フィールドを編集します。
3. **[保存]** を選択します。または、保存せずに [Tableau Embedding (Tableau 埋め込み)] 設定ページに戻る場合は、**[キャンセル]** を選択します。

ユーザー属性クレーム定義を作成する

Tableau ビューおよび Pulse Lightning Web コンポーネント (LWC) でユーザー属性関数 (UAF) を活用して、Viz を見ているユーザーの属性に基づいて特定のデータを表示できます。詳細について

は、「Tableau Lightning Web コンポーネントでユーザー属性関数 (UAF) を活用する」を参照してください。

Lightning アプリケーション ビルダーを使用して Tableau LWC を Lightning ページに追加する

Tableau LWC は、アプリページ、ホーム ページ、レコードLightning ページのみで使用できます。Lightning ページ タイプとLightning アプリケーション ビルダーの使用の詳細については、Salesforce ヘルプの「Lightning アプリケーション ビルダー」を参照してください。

> **注:** [ケース レコード] ページでは、Chatter メールでの Tableau Lightning Web コンポーネントの使用をサポートしていません。

Tableau ビューまたは Tableau Pulse LWC を既存の Lightning ページに追加するには、次の手順に従います。

1. 編集するLightning ページに移動します。
2. 右上にある歯車アイコンを選択します。
3. [ページの編集] を選択します。
4. 以下の **[Add a Tableau LWC to a Lightning page (Tableau LWC を Lightning ページに追加する)]** セクションに進みます。

Tableau View または Tableau Pulse LWC を新しい Lightning ページに追加するには、次の手順に従います。

1. Salesforce アプリで、右上隅にある歯車を選択し、**[設定]** を選択します。
2. ナビゲーション ペインで、**[Quick Find (クイック検索)]** 検索バーに「Lightning アプリケーション ビルダー」と入力します。
3. **[Lightning アプリケーション ビルダー]** 設定ページを選択します。
4. **[新規]** を選択します。
5. 作成するページ タイプを選択します。Tableau LWC は、アプリページ、ホーム ページ、レコードページで使用できます。
6. **[次へ]** を選択します。
7. 名前を入力して、新しいページのレイアウトを選択し、**[完了]** を選択します。

Tableau LWC を Lightning ページに追加する

1. ページの左側にあるコンポーネント リストから、Tableau View コンポーネントまたは Tableau Pulse コンポーネントをページにドラッグ アンド ドロップします。
2. IDP を構成します。
   - Tableau ビュー Lightning Web コンポーネントの構成
   - Tableau Pulse Lightning Web コンポーネントの構成

ページを保存してアクティブ化する

1. Tableau ビューまたは Tableau Pulse LWC の追加と構成が完了したら、**[保存]** を選択します。
2. 新しいページを作成した場合は、ユーザーに表示されるようにページをアクティブ化するよう求められます。**[有効化]** を選択します。
3. [アクティベーション] ページの **[Page Settings (ページの設定)]** タブで、名前を入力してアイコンを選択し、表示設定を選択します。
4. (オプション) [アクティベーション] ページの **[Lightning エクスペリエンス]** タブで、ページをさまざまな Lightning エクスペリエンス アプリに追加できます。
5. (オプション) [アクティベーション] ページの **[Mobile Navigation (モバイル ナビゲーション)]** タブで、ページをモバイル ナビゲーション メニューに追加できます。
6. **[保存]** を選択します。

複数の Tableau ビューの埋め込み

すべてのビューが同じサイトからのものである限り、Salesforce Lightning ページに複数の Tableau ビューを埋め込むことができます。Tableau は 1 つのセッションのみをサポートしており、そのセッションはサイトに固有です。最後に許可されたセッションにより、以前のセッションは消去されます。

複数のサイトからの Tableau ビューを埋め込むには、サイト固有の別の Lightning ページを作成する必要があります。

モバイル向け Tableau LWC シングル サインオン

> **注:** Tableau View および Pulse LWC は、iOS 17.2.1 以降で利用できます。

モバイル ユーザーで問題が起こるのを防ぐために、次のベストプラクティスを検討してください。

- Lightening ページのタイプ: モバイル ユーザーはアプリページとレコードページにアクセスできますが、ホーム ページにはアクセスできません。

  > **注:** レコードページは、特定のタイプのレコードに関連付ける必要があります。

- デスクトップとモバイルの両方のユーザーに同じページを使用する場合は、**[有効化]** を選択して、Lightning ページがデスクトップとモバイルの両方のフォーム ファクターの組織の既定として設定されていることを確認します。
- カスタマイズされたビジュアル体験を提供するために、デスクトップとモバイル用に個別の Lightning ページを作成することを検討してください。Tableau ビュー コンポーネントの高さは固定されており、さまざまな画面サイズに合わせて動的に調整されません。
- スクロール バーをビューに追加するには、Tableau ビュー コンポーネントのプロパティペインから **[ツールバーの表示]** を選択します。
- アプリページのタイプでは、**[有効化]** を選択して **[Lightning エクスペリエンス]** タブを選択します。ページを LightningBolt リストに追加すると、モバイルでページが見つけやすくなります。
- モバイルでは、iOS は既定でサイト間のトラフィックをブロックします。モバイルの設定を開き、Salesforce 設定を選択して、**[Web サイト間のトラッキングを許可する]** をオンにします。詳細については、「Web サイト間のトラッキングの有効化」を参照してください。

トークン認証を使用した Tableau LWC SSO のトラブルシューティング

保存された Tableau ホスト マッピングのホスト タイプを更新する

バージョン 254.10 より前のホスト マッピングを保存している場合は、Tableau ホストのタイプを更新する必要があります。

1. **[Tableau 埋め込み]** 設定ページで、保存したホスト マッピングに、**Tableau Cloud**、**Tableau Server (サーバー全体の EAS)**、**Tableau Server (サイトレベルの EAS)** のいずれかの Tableau ホスト タイプが含まれていることを確認します。

2. マッピングに他の Tableau ホスト タイプ (TABLEAU_CLOUD や TABLEAU_SERVER など) がある場合は、**[編集]** を選択してホスト マッピングを編集します。

3. [Tableau ホストマッピングの編集] ページで、Tableau サイトホストタイプのドロップダウンから適切な設定を選択し、**[保存]** を選択します。

4. 必要に応じて、保存したすべてのマッピングに対して手順 1 ~ 3 を繰り返します。

Salesforce とTableau の構成を検証する

1. **発行者 URL** と**JWKS URI** の値が Salesforce 設定とTableau 設定の両方で一致し、JWKS URI が **id/keys** で終わっていることを確認します。

    - Tableau Cloud の場合は、1 つのタブで [Salesforce 設定] の [Tableau 埋め込み] ページを開きます。別のタブで、[Tableau 設定] の [接続済みアプリ] タブを開きます。[接続済みアプリ] タブで、**[外部認証サーバー]** を選択し、**[編集]** を選択します。**[発行者 URL]** と**[JWKS URI]** の値が一致し、JWKS URI が id/key で終わっていることを確認します。
    - Tableau Server の場合は、1 つのタブで [Salesforce 設定] の [Tableau 埋め込み] ページを開きます。次に、Tableau サービス マネージャー (TSM) Web インターフェイスにサインインし、**[ユーザー ID とアクセス]** に移動して、**[認可サーバー]** タブを開きます。

2. ホストマッピングの確認: ホストマッピングを保存した場合は、そのマッピングに正しいサイト ID とホストタイプがあることを確認します。

JWT トークンを検証する

Tableau ビュー Lightning Web コンポーネントのプロパティエディターで、[デバッグ モード] を選択し、JWT トークンが期待どおりに動作していることを確認します。

1. コンソール ログを開き、トークンをコピーします。
2. jwt.io Web サイトに移動し、トークンを **[エンコード済み]** フィールドに貼り付けます。
3. 次のことを確認してください。
    - 件名 (「sub」) は Tableau ユーザー名と一致しています。
    - Tableau Cloud および Tableau Server (サイトレベルの EAS) の場合、オーディエンス (「aud」) は「tableau: {サイト ID}」です。
      Tableau Server (サーバー全体の EAS) の場合、オーディエンス (「aud」) は「tableau」です。
    - スコープ (「scp」) には、「tableau:views:embed」と「tableau:insights:embed」の両方が含まれます。
    - 発行者 (「iss」) EAS サーバーは正確です。

ページの有効化を確認する

場合によっては、ユーザーが Lightning ページを作成しても、有効化されていない、またはどこにも割り当てられていないため、ユーザーがそれを見つけることができないことがあります。**[有効化]** を選択して、Lightning ページが目的のフォーム ファクターの組織の既定として設定されていることを確認します。

ヒント: デバッグするときは、リッチ テキストのコンポーネントをページにドラッグ アンド ドロップすると便利です。ページ タイプと埋め込もうとしているビュー URL の簡単な説明を追加します。これにより、エンドユーザーが表示しているページが管理者が編集しているページであることを確認できます。

Tableau ビュー LWC がシームレスな認証なしで動作していることを確認する (Tableau ビュー LWC のみ)

1. Lightning ページの [Tableau ビュー] ペインで、**[既定の認証トークン]** のチェックボックスをオフにし、変更を保存します。
2. Tableau Server で、**[サイト ID]** フィールドが空であることを確認します。
3. 別のタブで Tableau にサインインしている場合は、サインアウトします。ビュー URL に移動すると、Tableau サインイン ページにリダイレクトされることを確認します。サインインしないでください。
4. Lightning ページに移動します。Tableau ビュー LWC には **[Tableau にサインイン]** ボタンが表示されます。
5. **[Tableau にサインイン]** を選択し、Tableau の認証資格情報を入力してサインインします。
   注: ビューが読み込まれない場合は、Tableau への認証に広範な問題があることを示しています。

エラー: LWC コンポーネントのバージョンはサポートされなくなりました (Tableau View LWC のみ)

このエラーを解決するには、次の手順に従ってください。

1. [コンポーネント] リストで「Tableau」を検索し、新しい **[Tableau ビュー]** コンポーネントをページにドラッグ アンド ドロップします。
2. 古いコンポーネントの [Tableau ビュー] ペインからすべてのプロパティをコピーして、新しいコンポーネントに上書きします。
3. 古いコンポーネントの削除アイコンを選択します。

エラー: Tableau Pulse LWC を有効にするには、Salesforce 管理者に連絡して Tableau のシームレス認証を設定してください (Tableau Pulse LWC のみ)

このエラーを解決するには、このページの手順に従って、**Tableau LWC のシームレス認証を有効にしてください**。

関連項目

接続済みアプリのトラブルシューティング

EAS を登録して埋め込みコンテンツの SSO を有効にする

Tableau Lightning Web コンポーネントでユーザー属性関数 (UAF) を活用する

Tableau ビューおよび Pulse Lightning Web コンポーネント (LWC) でユーザー属性関数 (UAF) を活用して、Viz を見ているユーザーの属性に基づいて特定のデータを表示できます。ユーザー属性関数は行レベルのセキュリティを提供して、Viz を見るユーザーに自分に関連するデータのみが表示されるようにします。UAF の詳細については、「ユーザー属性関数でパーソナライズされた分析のパワーを解放する」を参照してください。

前提条件

- Tableau Lightning Web コンポーネントには、トークン認証を使用したシングル サインオン (SSO) が設定されています。「Tableau Lightning Web コンポーネントとトークン認証によるシングル サインオン (SSO) の設定」を参照してください。
- Tableau は、認証ワークフローでユーザー属性を取得するように構成されています。「ユーザー属性の取得を許可する」を参照してください。
- 埋め込まれる Tableau コンテンツには、ユーザー属性関数が含まれます。Tableau ワークブックでユーザー属性関数を使用する方法の例については、「コンテンツ作成者がユーザー属性関数を含めていることを確認する」を参照してください。

Salesforce での Tableau UAF クレーム定義

Tableau UAF クレーム定義を使用すると、JWT にユーザー属性のクレームを含めることができます。これらの属性の値は、Salesforce のアカウントオブジェクト、商談オブジェクト、またはケースオブジェクトから取得できます。Tableau コンテンツ内のユーザー属性関数と組み合わせて使用すると、指定された属性に一致するデータのみが表示されます。

たとえば、Tableau ダッシュボードは、[State/Province (都道府県/州)] データ列の値が属性「state」の JWT クレームの値の 1 つと一致する売上データのみを表示するように構成できます。

構文: USERATTRIBUTEINCLUDES('attribute_name', expected value)

例: USERATTRIBUTEINCLUDES('state', [State/Province])

Salesforce でクレーム定義を作成すると、JWT にユーザー固有の適切な情報を含めることができます。この例では、Billing State/Province フィールドから値を取得することにより、現在のユーザーがアカウント所有者であるアカウントの値が属性「state」に入力されます。

クレーム定義の例

- 定義名: state_Account_BillingState
- 属性名: state
- ターゲットオブジェクト: Account
- ターゲットオブジェクトフィールド: Billing State/Province
- 閲覧者とターゲットオブジェクトの関係: 現在の閲覧者はアカウント所有者

ワシントン州、オレゴン州、カリフォルニア州にアカウントを持つ営業担当者の場合、JWT には [WA, OR, CA] という値を持つ「state」のクレームが含まれます。Tableau ダッシュボードには、[State/Province] 値が WA、OR、または CA に一致するデータのみが表示されます。

定義の作成

ユーザー属性の定義を作成するには、次の手順を実行します。

1. Salesforce アプリで、右上隅にある歯車を選択し、**[設定]** を選択します。
2. 左側のナビゲーション ペインで、**[Quick Find (クイック検索)]** 検索バーに「Tableau」と入力します。
3. **[Tableau UAF Claims Definition (Tableau UAF クレーム定義)]** 設定ページを選択します。
4. **[Create New (新規作成)]** を選択します。
5. 定義の値を入力します。
    a. 定義名: 一意の名前を入力します。(例: state_Account_BillingState)
    b. 属性名: Tableau のユーザー属性関数で指定された属性名を入力します。(例: state)
    c. ターゲット オブジェクト: 値を取り出したいターゲット オブジェクトを選択します。Tableau LWC は現在、User オブジェクトとAccount オブジェクトをサポートしています。
    d. ターゲット オブジェクト フィールド: 値を取り出すターゲット オブジェクトのフィールドを選択します。(例: Billing State/Province)
    e. 閲覧者とターゲット オブジェクトの関係: 現在の閲覧者とターゲット オブジェクトの関係を選択します。(例: **現在の閲覧者はアカウント所有者**)。
6. **[保存]** を選択します。または、保存せずに [Tableau UAF Claims Definition (Tableau UAF クレーム定義)] 設定ページに戻る場合は、**[キャンセル]** を選択します。

Tableau Lightning Web コンポーネント ページに定義を追加する

ホーム ページ、アプリページ、およびレコードLightning ページ

定義を追加するには、次の手順を実行します。

1. **[Select… (選択…)]** を選択します。
2. 矢印を使用して、利用可能な定義を**[Selected (選択済み)]** フィールドに移動します。

3. **[OK]** を選択します。

4. 同じページのすべての Tableau Lightning Web コンポーネントに対してこの手順を繰り返します。詳細については、「同じページに複数のコンポーネントがある場合の考慮事項」を参照してください。

Experience Cloud ページとユーティリティバー

UAF クレーム定義名のコンマ区切りリストを入力します。定義名をコピーするには、Salesforce アプリから **[Tableau UAF Claims Definition (Tableau UAF クレーム定義)]** 設定ページに移動し、**[コピー]** を選択します。

同じページに複数のコンポーネントがある場合の考慮事項

複数の Tableau View または Tableau Pulse Lightning Web コンポーネントを同じページに追加する場合、各コンポーネントには同じ UAF 定義名のセットが必要です。コンポーネントの 1 つのみが Tableau に対して認証され、そのセッションが両方のコンポーネントの読み込みに使用されます。つまり、認証されるコンポーネントがどちらであっても、そのセッションに両方のコンポーネントに関連するユーザー属性情報が存在している必要があります。

例:

- 2 つの Tableau ビュー コンポーネントがあり、1 つが定義 A のユーザー属性情報を必要とし、もう1 つが定義 B のユーザー属性情報を必要とする場合、両方のコンポーネントに定義 A とB の両方が含まれている必要があります。
- Tableau ビューのコンポーネントが定義 A を必要としているものの、同じページ上の Tableau Pulse コンポーネントがユーザー属性情報を必要としない場合は、両方のコンポーネントに定義 A を含める必要があります。

定義の削除

ユーザー属性の定義を削除するには、次の手順を実行します。

1. Salesforce アプリで、右上隅にある歯車を選択し、**[設定]** を選択します。
2. 左側のナビゲーション ペインで、**[Quick Find (クイック検索)]** 検索バーに「Tableau」と入力します。
3. **[Tableau UAF Claims Definition (Tableau UAF クレーム定義)]** 設定ページを選択します。
4. 削除する定義に対して **[削除]** を選択します。

ホーム ページ、アプリページ、およびレコードLightning ページ

Tableau LWC にすでに追加されている定義を削除する場合は、Lightning アプリケーション ビルダーでページを更新して、使用可能な定義を更新します。削除された定義は自動的に取り除かれます。

Experience Cloud ページとユーティリティバー

削除された定義は、Tableau LWC コンポーネント フィールドから自動的に取り除かれません**[UAF Definition Names (UAF 定義名)]** コンポーネント フィールドの削除された定義は問題を引き起こしませんが、不要かつ乱雑な状態を作り出します。Tableau LWC コンポーネント フィールドから定義を削除するには、[UAF Definition Names (UAF 定義名)] フィールドを選択し、使用されなくなった名前を手動で削除します。

Tableau でデータを操作する

このチュートリアルでは、Tableau Server におけるデータ ビジュアライゼーション (ビュー) の表示と操作の基本について説明します。

Tableau をツールとして使用すると、パブリッシュされたビジュアライゼーションを操作して、インサイトを探索したり、質問を投げかけたり、データを把握したりすることができます。使い方は次のとおりです。

どうぞ、クリックしても安全です

Tableau は操作されることを考慮して構築されています。ビジュアライゼーションを操作しても、差し当たり、自分のビジュアライゼーションの外観しか変わりません。

他の人には、オリジナルのビジュアライゼーションが表示されています。ビジュアライゼーションの表示に使用したデータも、同じ状態に保たれています。

1: Tableau サイトとは

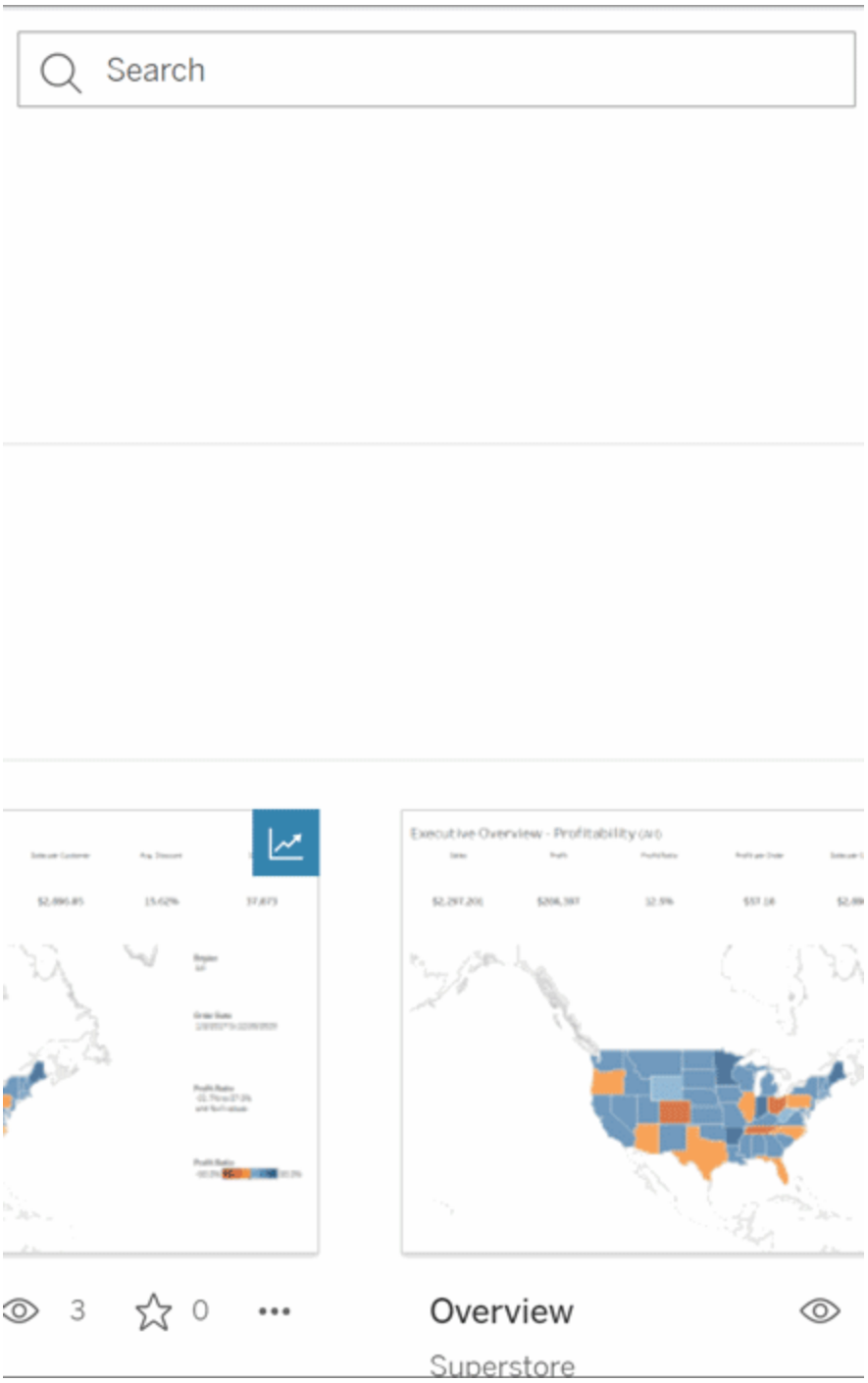
Tableau サイトは、データやデータビジュアライゼーションをチームで互いに共有できる場所です。みんながパブリッシュして利用できるようにしたものを探索できます。

Tableau サイトにサインインすると、ホーム ページが表示されます。

2: viz の検索

Tableau サイト上のビジュアライゼーションをビューと呼びます。ワークブックは、複数のビューを 1 つのファイルにまとめたものです。検索機能で、ビューまたはワークブックを検索します。

検索結果には、クエリに応じて、さまざまなコンテンツ タイプがすべて表示されます。



探していたものがクイックサーチのビューにない場合は、[すべて表示] を選択してすべての検索結果を表示するか、[探索] ページで閲覧して回ります。Tableau サイトでホストできるさまざまな種類のコンテンツがすべて表示されています。

3: コンテンツの操作

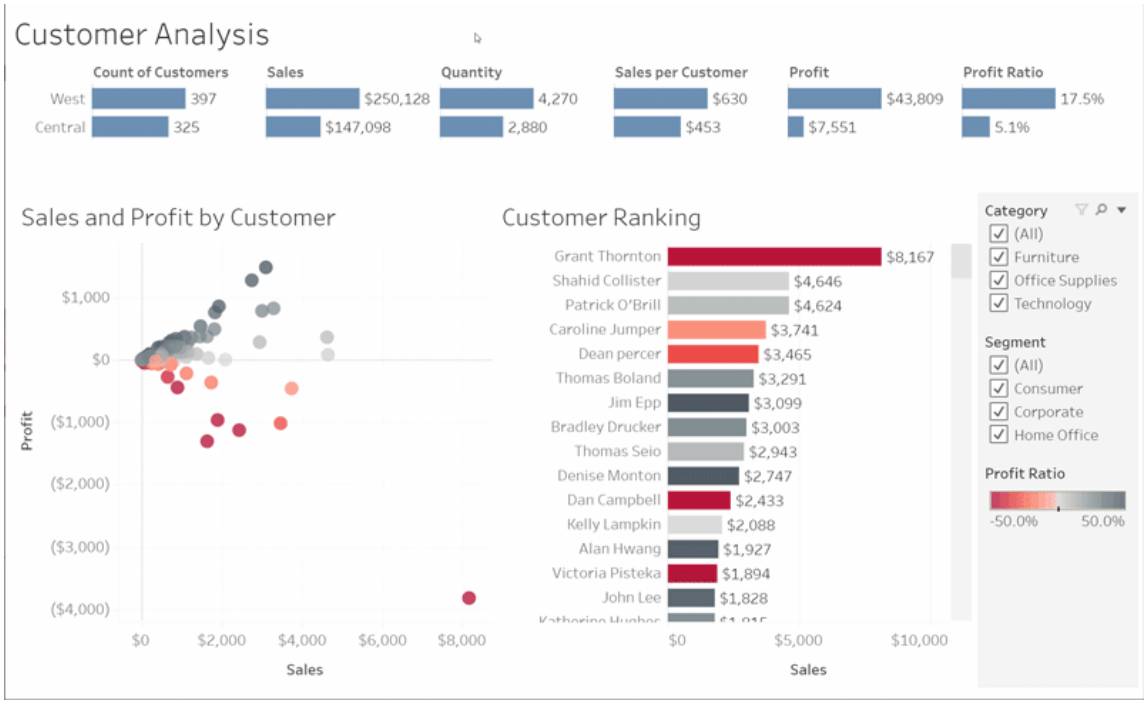
パブリッシュされたビューをキャンバスとして使用すると、データを操作して理解することができます。参照元のデータを破損したり変更したり、他の人に表示されているものを変更したりすることはありません。

ツールボックスには以下のようなツールがあり、データからインサイトを得るために利用できます。

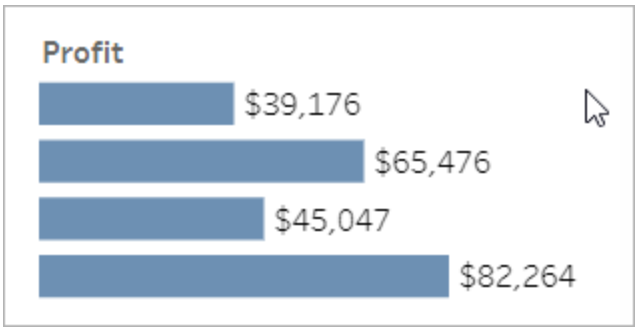
詳細表示とデータの並べ替え

データをクリックできることがわかったので、チェックしてみます。

ビューの中でマウスを移動すると、各データポイント(マーク)の詳細を示すツールヒントが表示されるでしょう。複数のマークを選択することもできます。

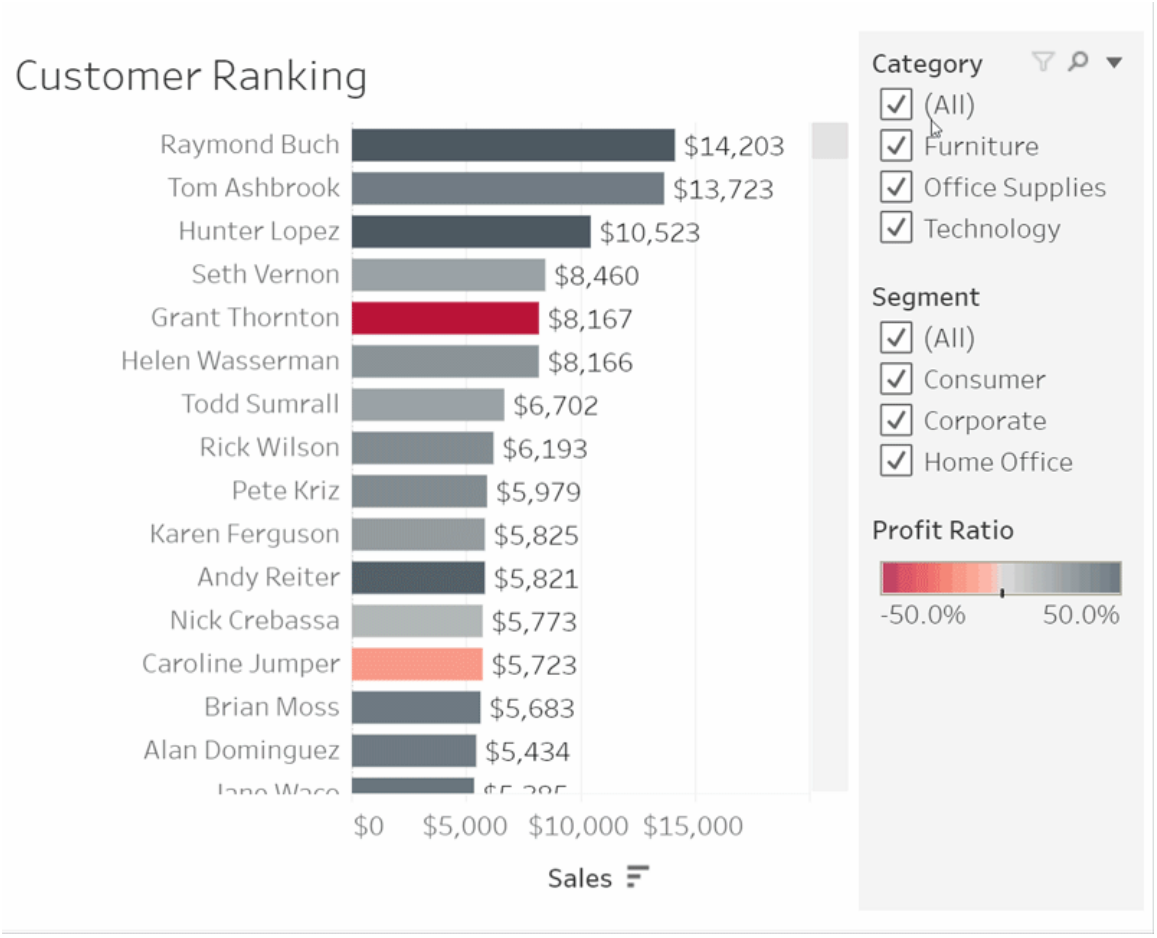


表をアルファベット順や数値順に並べ替えるには、列のヘッダーにカーソルを合わせて並べ替えのアイコンをクリックします。



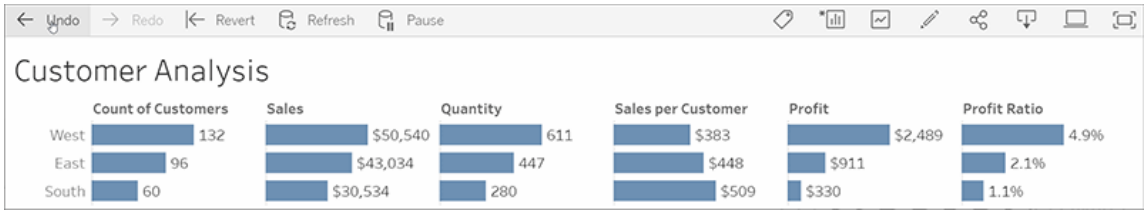
データのフィルター

表示されるデータを特定の領域、日付、カテゴリなどにトリミングしたり制限したりすることができます。



元に戻す

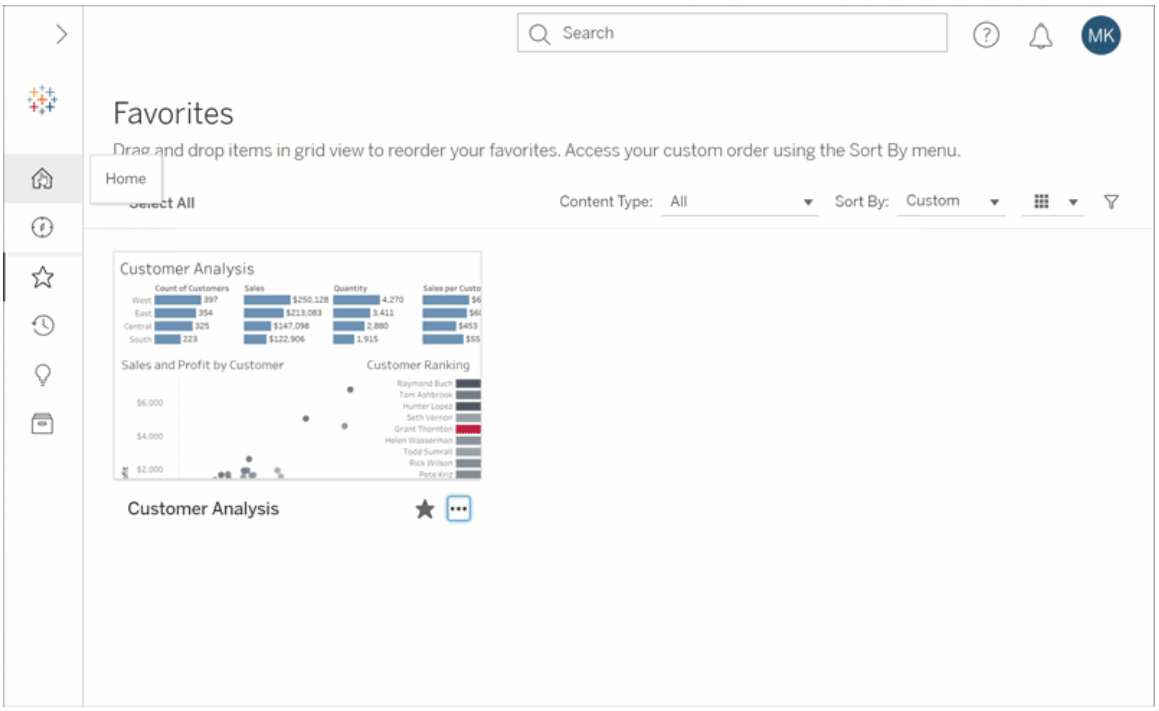
すべてを除外するのではなく、1 つの領域だけを除外するつもりだった、ということがあるかもしれません。[ひとつ前に戻す (Udno)] をクリックすると最後の変更を取り消し、[元に戻す (Revert)] を使用するとすべての選択を元に戻すことができます。



4: 最新化

ビューは新しいデータで自動的に更新されるため、新しいチャートを検索して最新の情報を取得する必要はありません。星形のアイコンをクリックしてお気に入りに追加して、ビューを手元に置いておくことができます。

すべてのお気に入りは、ナビゲーション パネルの [お気に入り] ページに追加されます。最近調べたダッシュボードやビューもホームページに表示され、次回も利用できます。



これらは基本機能の一部であり、Tableau Server でできることはたくさんあります。詳細については、「Tableau Web ビューでできること」を参照してください。

探索してみてください。

バックグラウンドマップの選択

Tableau には、マップビューを作成するためにアクセスできるバックグラウンドマップのセットがインストールされています。

既定では、Tableau マップは、Mapbox マップによって提供される**明るい**バックグラウンドレイヤーに表示されます。

次の表には、Tableau にインストールされている 8 つのバックグラウンドマップについての説明が記載されています。

バックグラウンドマップ	説明
明るい	マークに重点を置いた淡色のマップ。地理的なコンテキストを提供します。データのない領域はすべて、白または薄い灰色で表示されます。
標準	明るいマップに類似した汎用マップ。データのない陸地は白または薄い灰色で表示され、水域は薄い青で表示されます。
暗い	マークに重点を置いた淡色のマップ。地理的なコンテキストを示します。明るいマップを反転したもの。データのない領域は黒または濃い灰色で表示されます。
ストリート	主要な道路や交通網を含む汎用マップ。
アウトドア	水域や公園などの地形や自然の特徴を含む汎用マップ。
サテライト	地球全体の衛星画像を含む定型化されたマップ。
オフライン	インターネットに接続していないときでも使用できるマップ。このバックグラウンドマップは、マップを構成するイメージをマシンのキャッシュに保存します。これにより、パフォーマンスが向上し、オフラインでアクセスできるようになります。詳細については、オフラインマップのセクションを参照してください。
なし	マップ以外の Viz タイプで緯度と経度の間のデータを表示するビジュアライゼーション。

バックグラウンドマップの変更:

Tableau で、**[マップ] > [バックグラウンド マップ]** の順に選択し、さらに使用するバックグラウンドマップを選択します。

Tableau Desktop での既定のバックグラウンドマップの変更 (非推奨の機能)

注: 既定のバックグラウンドマップの変更は、Tableau Desktop 上の WMS マップでのみ機能するレガシー機能です。この機能の使用はお勧めしません。

既定のバックグラウンド マップを Web Map Service (WMS) マップまたはオフライン マップに設定できます。WMS マップの使用の詳細については、「Web Map Service (WMS) サーバーの使用」を参照してください。

既定のバックグラウンド マップを指定するには:

1. Tableau Desktop で、**[マップ] > [バックグラウンド マップ] > [Add WMP Map (WMP マップの追加)]** または **[オフライン]** の順に選択します。
2. [マップ] > [バックグラウンド マップ] > [既定値として設定] の順に選択し、選択したバックグラウンド マップを既定に設定します。

バックグラウンド マップは Tableau マップ ソース (.tmsd) ファイルとして自動的に保存され、マイ Tableau リポジトリの [マップ ソース] フォルダーに配置されます。

これで、すべての新しいワークシートに対する既定のバックグラウンド マップが設定されました。

オフライン バックグラウンド マップの使用

Tableau Desktop に付属のオフライン バックグラウンド マップを使用して、マップ ビュー内のデータをオフラインで作成および検査できます。

オフライン バックグラウンド マップを使用するには:

- Tableau で **[マップ] > [バックグラウンド マップ] > [オフライン]** の順に選択します。

注: オフライン バックグラウンド マップは、マシンに保存されているマップ イメージを使用します。これらのイメージは次の場所で見つけることができます。

- Windows の場合: `C:\Program Files\Tableau\<Tableau Version>\Local\Maps`
- Mac の場合: `//Applications/<Tableau Version>.app/Contents/install/local/maps`

ただし、保存されていないマップ イメージを Tableau で取得する必要がある操作もいくつかあります。新しいマップがマシンに保存されていない場合、Tableau に付属のオンライン マップに再接続するまでマップを読み込むことはできません。

次の 1 つまたは複数の操作を実行する場合、オンライン マップへの再接続が必要になる場合があります。

- **レイヤーのオン/オフの切り替え -** キャッシュに保存されていないレイヤーをオンにする場合、必要な情報を取得するために Tableau をオンラインに接続する必要があります。
- **ズームイン/ズームアウト -** 地図上でのズームインまたはズームアウトには異なるマップ イメージが必要です。指定されたズーム レベルのイメージがキャッシュにない場合、Tableau は更新された地図を取得する必要があります。
- **パン -** パンを行うと、新しいマップ イメージが必要になる場合があります。オフラインで作業中に必要なマップ イメージと凡例がキャッシュに保存されていない場合、新しいイメージと凡例は読み込まれません。

Tableau マップに再接続するには:

- Tableau Desktop で、**[マップ] > [バックグラウンド マップ] > [Tableau]** の順に選択します。

注: マップをオフラインに設定してからワークブックをパブリッシュした場合、パブリッシュされたワークブックはオフラインの保存済みマップを使用し、上記のオフライン マップの機能と制限事項をすべて使用します。

メトリクスの作成とトラブルシューティング (廃止)

従来のメトリクスの廃止

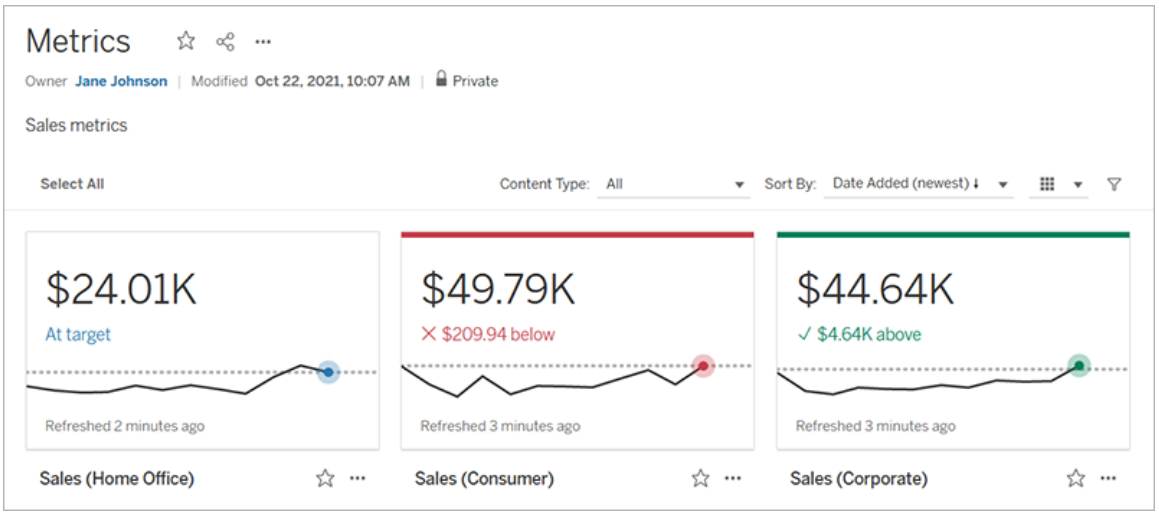
この記事は、Tableau の従来のメトリクス機能について説明しています。この機能は、Tableau Cloud では 2024 年 2 月に廃止され、Tableau Server ではバージョン 2024.2 では廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。

Tableau Pulse では、メトリクスを追跡する新しい方法が導入されています。Tableau Pulse では、作成したメトリクスを使用してデータに関するインサイトを生成します。これらのデータのインサイトは、メトリクスをフォローするユーザーに直接送信されるため、ユーザーは仕事の流れの中で

データの変化を知ることができます。詳細については、「Tableau Pulse でメトリクスを作成する」を参照してください。

引き続き利用したい従来のメトリクスがある場合は、それらのメトリクスのデータソース、メジャー、時間ディメンションをメモし、Tableau Pulse で再作成します。従来のメトリクスは Tableau Pulse に自動的に移行されません。

メトリクスを利用すると、データについての情報をいつも入手していることができます。メトリクスは自動的に更新され、コンテンツのグリッドビューやリストビューに現在の値が表示されるため、気になる主要な数値をすべて数秒で確認できます。



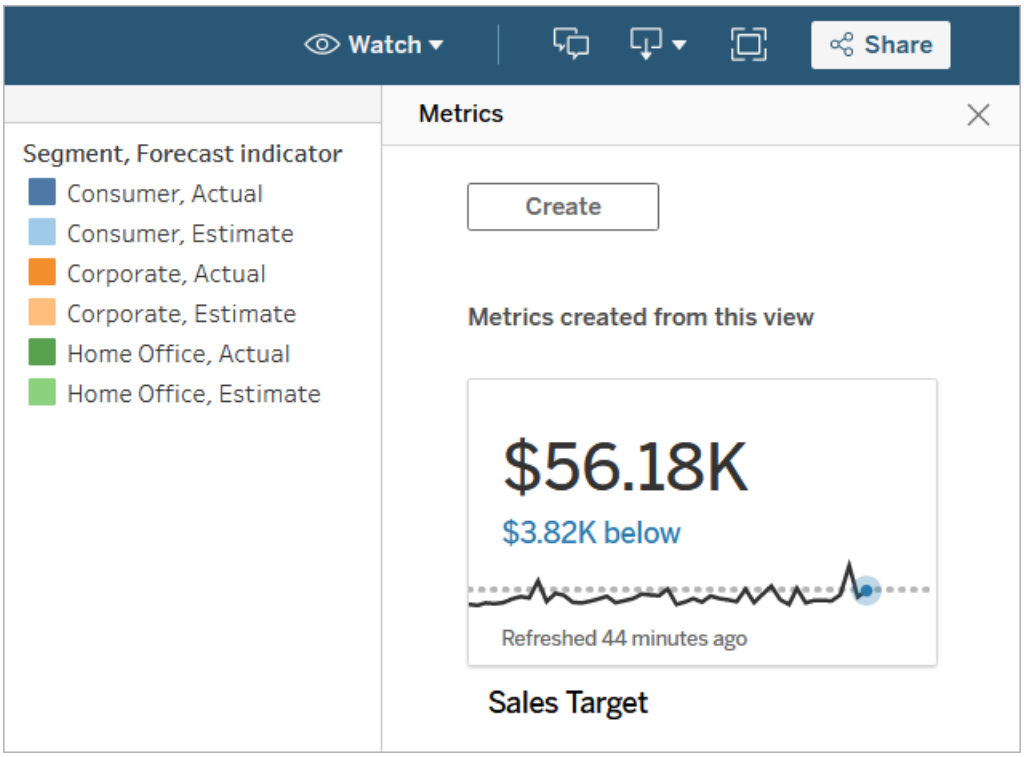
基本的なレベルのメトリクスでは、売上の合計など、メジャーの集計値を示します。より複雑なメトリクスでは、タイムライン、比較、ステータスなどを含めることができます。これにより、前の時点や定義した値と比較して、パフォーマンスの状況をわかりやすく示すことができます。

定期的にチェックする一連のダッシュボードがある場合、監視したい数値のメトリクスを作成して、お気に入りやコレクションに追加したり、同じプロジェクトで作成したりしておくと、まとめて追跡することができます。そうすれば、データを深く掘り下げたいとき以外は、ダッシュボードを読み込んでフィルターリングする必要がなくなります。

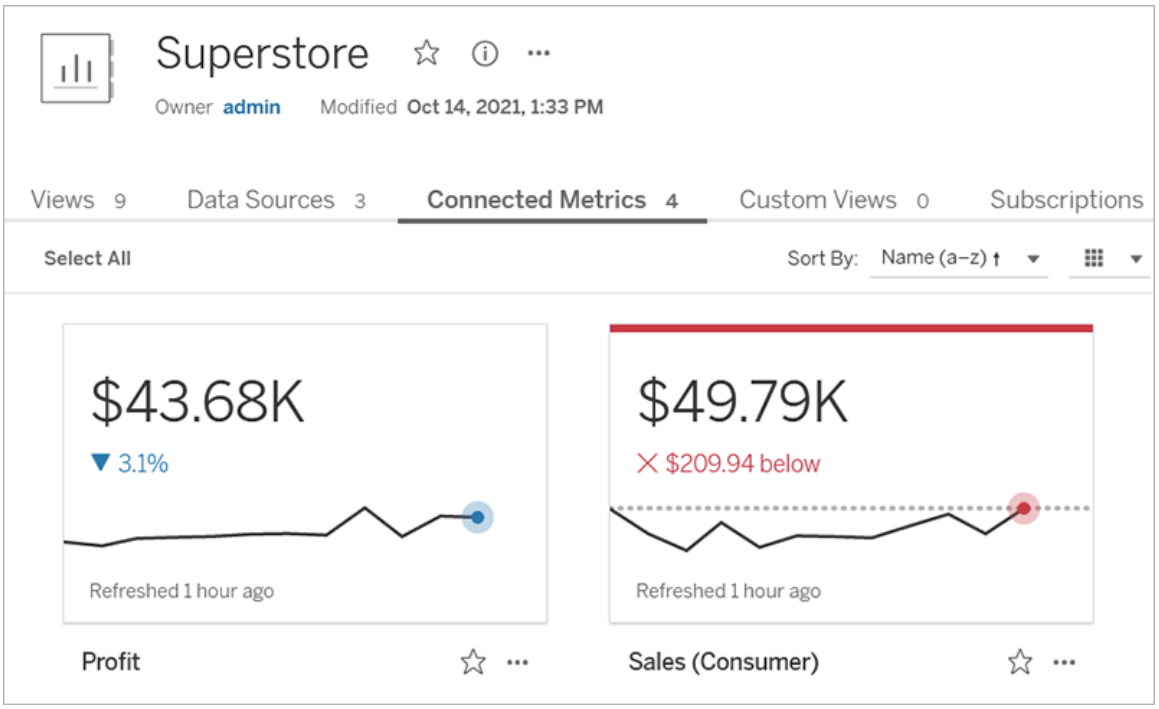
サイト内のメトリクスを検索する

Tableau サイトでメトリクスを見つけるには、いくつかの方法があります。閲覧権限があるすべてのメトリクスを参照するには、[探索] ページに移動し、コンテンツ タイプ メニューから**[すべてのメトリクス]** を選択します。

特定のビューまたはワークブックに関連するメトリクスを探している場合は、そのコンテンツの接続済みメトリクスを確認します。ビューの接続済みメトリクスを表示するには、ビューを開き、ビューのツールバーの **[Watch (視聴)] > [メトリクス]** をクリックします。表示されるメトリクスは、最新の作成日から最も古い作成日の順に並べられます。



ワークブック内のすべてのビューの接続済みメトリクスを表示するには、ワークブックに移動し、**[接続済みメトリクス]** タブをクリックします。[並べ替え] メニューを使用すると、これらのメトリクスを並べ替えることができます。

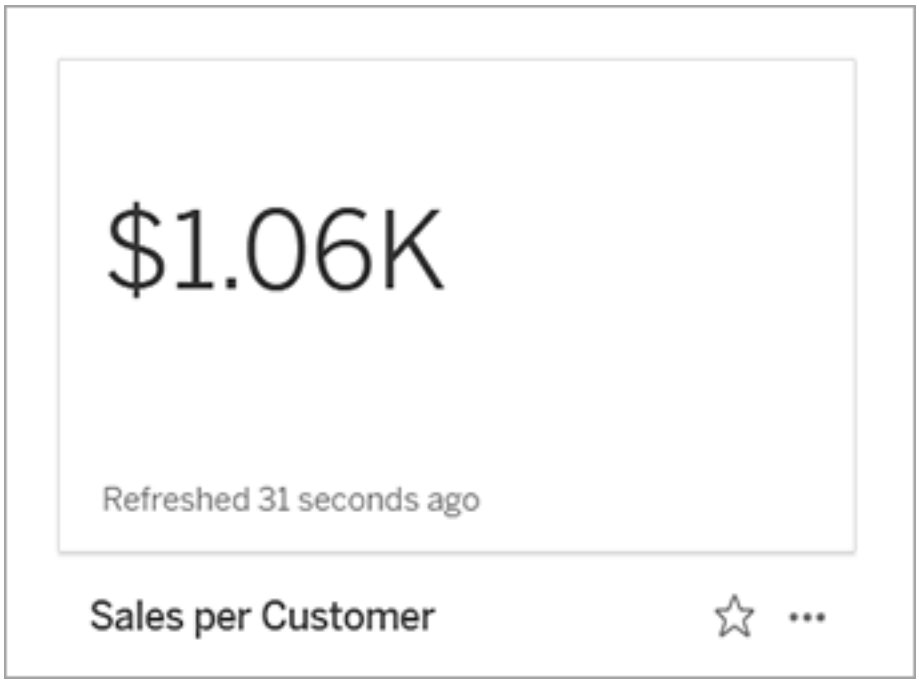


メトリクスのコンポーネント

メトリクスを定義するために必要なデータは、メジャーの集計だけです。メトリクスはビュー内のマークから作成され、そのマークに関連付けられたメジャーでメトリクスを定義します。集計されていないマークは時間が経過しても変化しないため、メジャーは集計されている必要があります。Tableau のディメンションとメジャーについては、「ディメンションとメジャー、青と緑」を参照してください。

メトリクスは、オプションで日付ディメンションによって定義することもでき、メトリクスの比較とステータスを設定することもできます。これらの各コンポーネントは、メトリクス カードに表示されるデータにコンテキストを追加します。

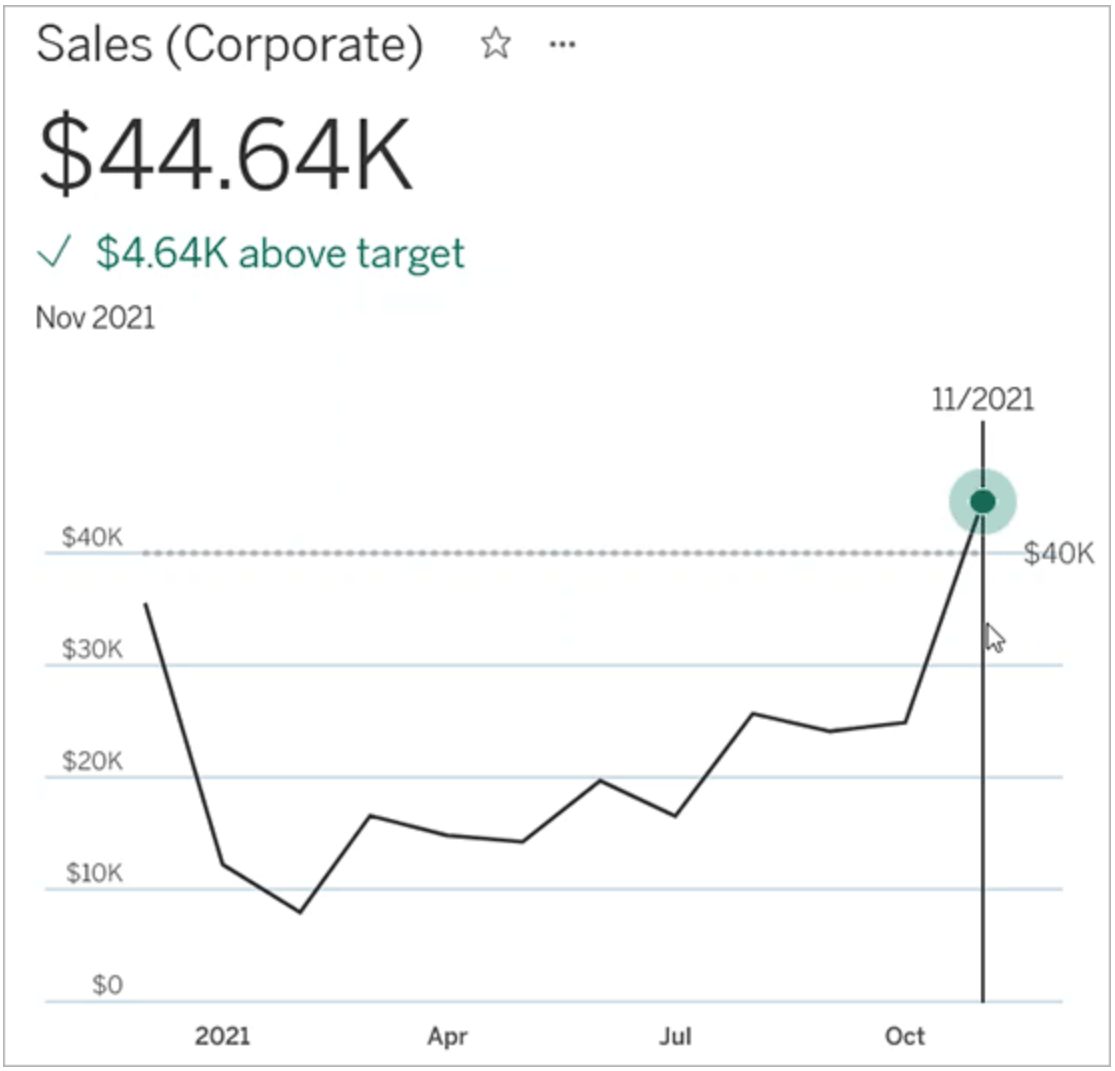
メトリクスを定義するメジャーのみを持つメトリクスは、単一の数値として表示されます。この数値はデータが更新されると更新されますが、カードにタイムラインは表示されません。



タイムライン

マークを選択してメトリクスを定義するとき、そのマークに日付ディメンションが関連付けられている場合、そのディメンションはメトリクスの定義の一部になります。日付ディメンションを持つメトリクスはタイムラインを表示し、メトリクスの履歴比較を設定できます。デフォルトでは、履歴比較は前のマークと比較されます。

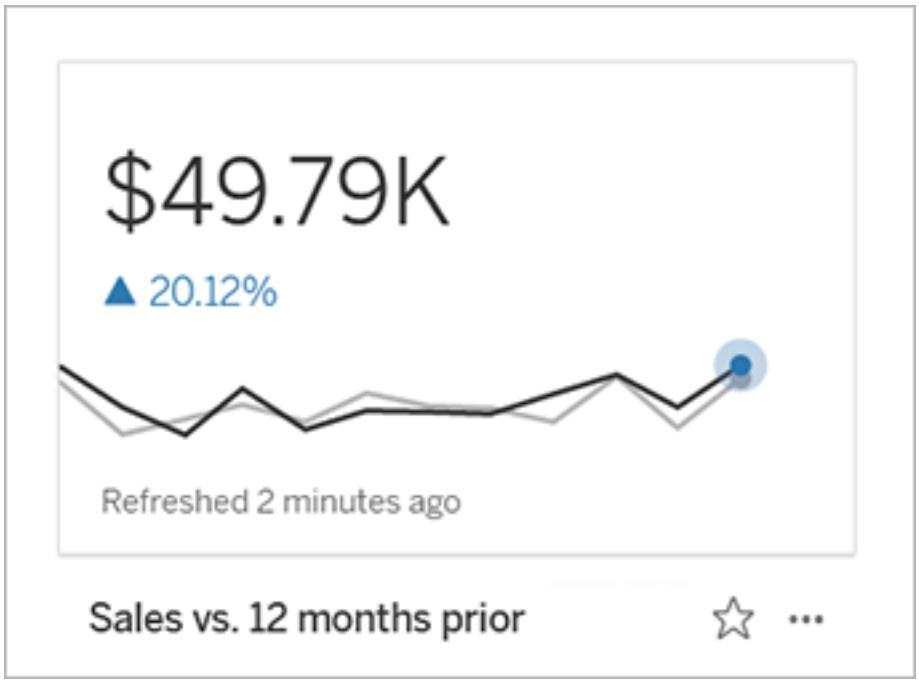
メトリクスの詳細ページを開くと、タイムラインは、例えば日次売上や月間ユーザーなど、日付ディメンションの粒度に基づくメジャーの値を表示します。履歴の値を表示するには、タイムライン上の点にカーソルを合わせます。



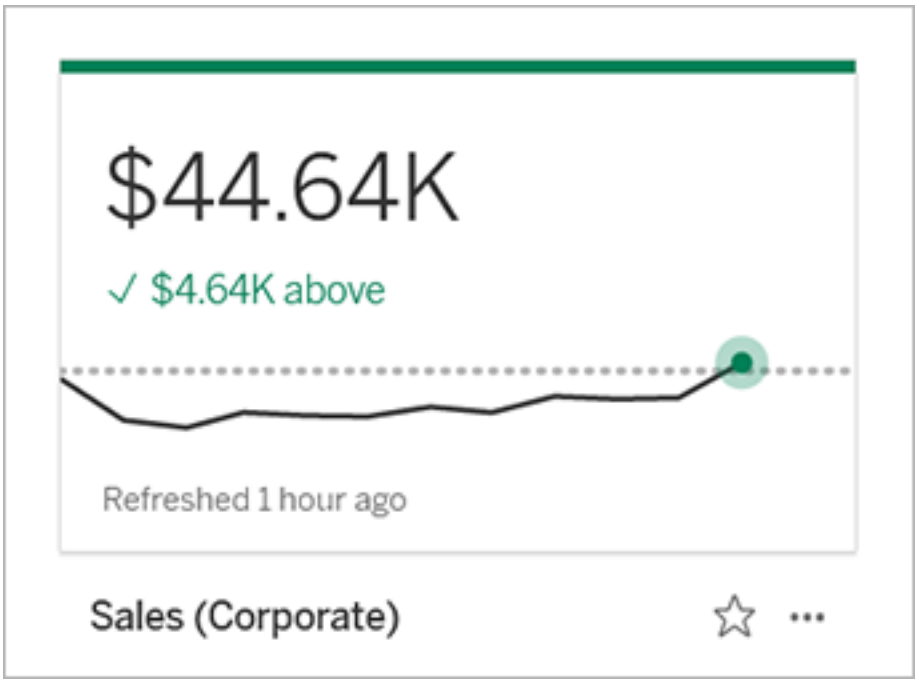
比較

メトリクスに対して設定できる比較には、履歴比較と定数比較の 2 種類があります。履歴比較は、メトリクスに日付ディメンションが関連付けられている場合にのみ設定できますが、定数比較は、任意のタイプのメトリクスに設定できます。

履歴比較は、現在の値と、数時間前、数日前など、指定された時間単位の以前の値との相対的な比較です。たとえば、月間売上の現在の値を 12 か月前の値と比較するように設定できます。履歴比較は、データがメトリクスに追加されるたびに、新しいデータの日時に基づいて修正されます。

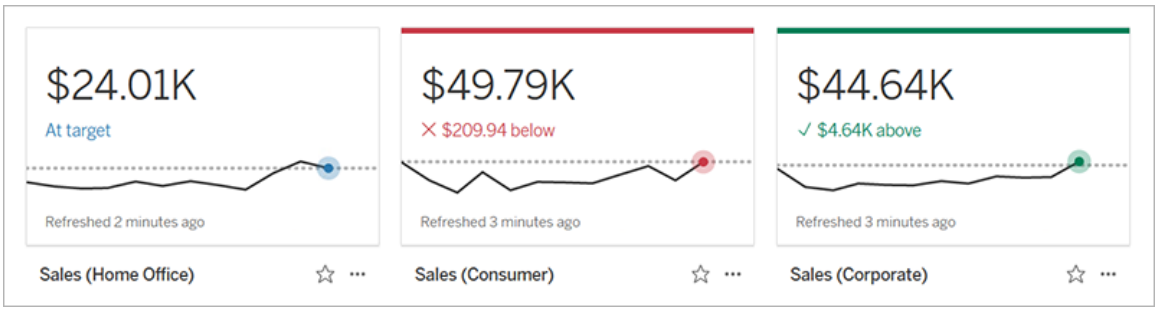


定数比較は、新しいデータが追加されても変化しない単一の値に対する比較です。たとえば、納期達成率 90% を維持する必要がある場合など、あるしきい値を上回ることがわかるように比較を設定できます。または、毎月の月間売上目標など、目指している累積的な目標を定めてもよいでしょう。



ステータス

定数比較を行うメトリクスの場合、比較値より上、下、またはその値であることが、良いこと、悪いこと、または中立であるのかを定義できます。「良い」ステータスのメトリクスは、比較値の横にチェックマークが表示され、メトリクス カードの上部に緑色の帯が表示されます。「悪い」ステータスのメトリクスは、比較値の横に X が表示され、メトリクス カードの上部に赤い帯が表示されます。「中立」ステータスのメトリクスは、ステータス インジケーターのないメトリックと同じように表示されます。アイコンや色はカードに適用されません。



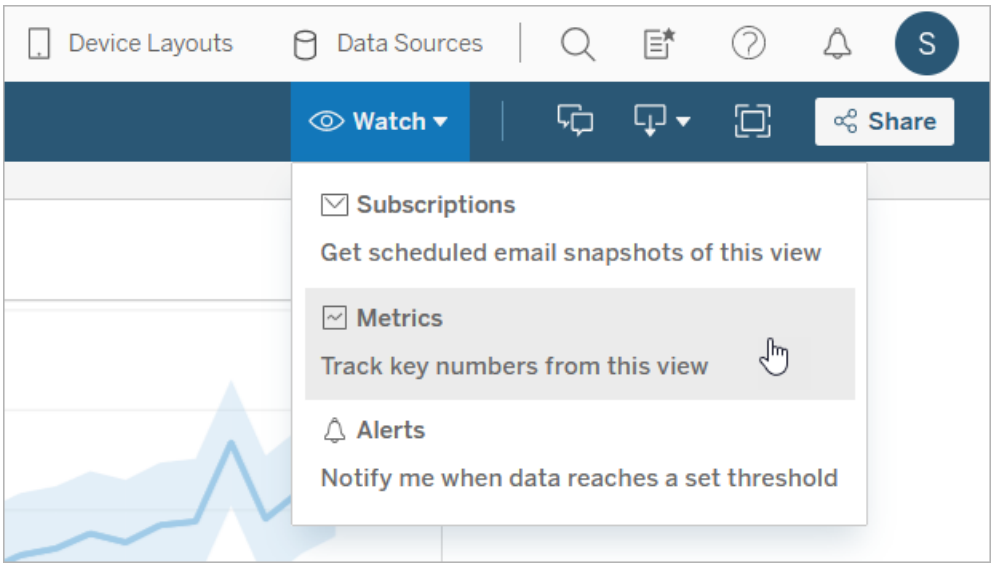
ビューからメトリクスを作成する

Creator または Explorer (パブリッシュ可能) のサイトロールを持っているユーザーや、関連するワークブックで [Create/Refresh Metric (メトリクスの作成/更新)] 機能を使用できるユーザーは、Tableau Cloud または Tableau Server でメトリクスを作成できます。

メトリクスを作成する前に、ビューの接続されたメトリクスをチェックして、作成しようとしているメトリクスがすでに存在していないことを確認してください。メトリクスを重複して作成する代わりに、既存のメトリクスを開いてお気に入りとして追加します。

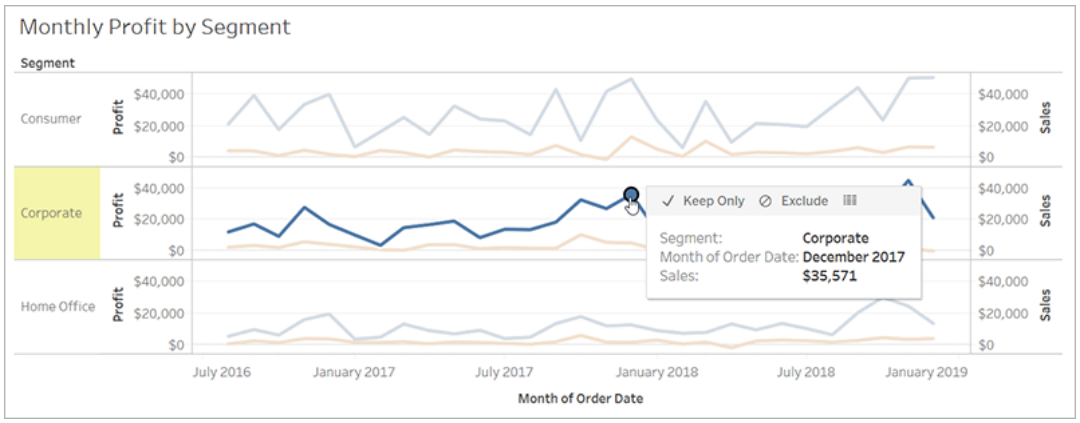
マークを選択してメトリクスを定義する

1. メトリクスを作成するビューに移動します。
2. ビューのツールバーで、**[Watch (視聴)] > [メトリクス]** を選択します。



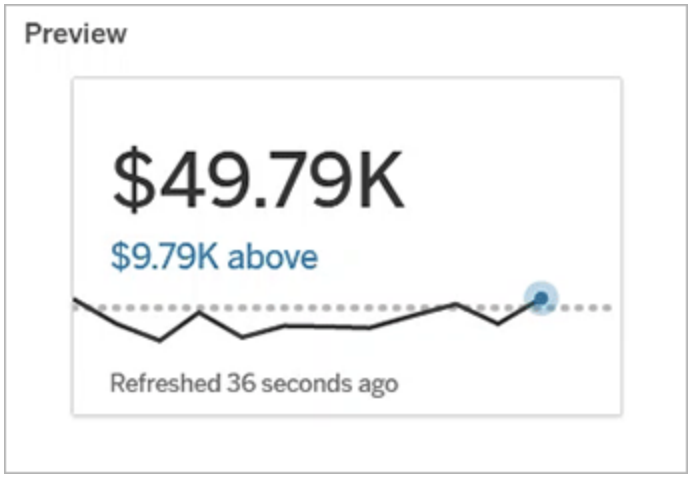
メトリクス ペインが開きます。

3. 接続済みメトリクスがペインに表示されている場合は、**[作成]** ボタンを選択すると、オーサリング モードに入ります。
4. マークを選択します。エラーが発生したら、メトリクスを作成できない場合を参照してください。



このマークに関連付けられたメジャーで、メトリクスが定義されます。このマークに適用されるフィルターは、メトリクスにも適用されます。このマークに日付ディメンションが関連付けられている場合、その日付ディメンションでもメトリクスが定義され、メトリクスにはタイムラインが表示されます。

メトリクス ペインにメトリクスのプレビューが表示されます。プレビューの値はメトリクスの最新の値であり、選択したマークの値とは、時系列で最新ではない場合、異なる可能性があります。プレビューは、さまざまな設定を試すと更新されます。



メトリクスの説明と設定

メトリクスの設定に使用できるオプションは、選択したマークや比較のタイプによって異なります。

1. **名前** フィールドは選択したマークに基づいて事前に入力されています。メトリクスに別の名前を付けることもできます。メトリクスは、それが属するプロジェクト内で一意となる名前をつけなければなりません。

2. **[説明]** には、他のユーザーがメトリクスを理解するのに役立つオプションのメッセージを入力します。たとえば、メトリクスに適用されるフィルターを説明したり、メトリクスで使用されるデータソースを示したりします。

3. **日付範囲** (日付ディメンションを持つメトリクスの場合のみ) には、デフォルトのオプションから 1 つ選択するか、カスタム範囲を設定します。メトリクスにマークが大量にある場合、日付範囲に制限を設けるとタイムラインが読みやすくなります。

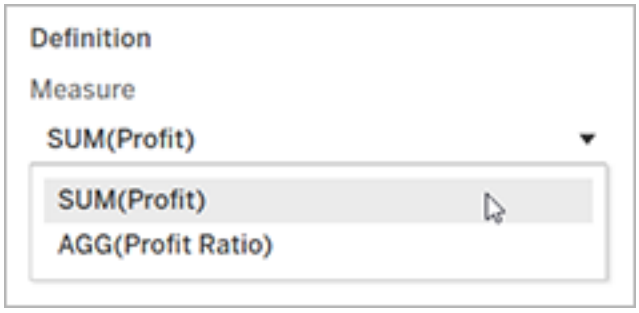
4. メトリクスの**比較タイプ** (履歴または定数) を選択します。

5. **履歴**比較の場合:
   - どれくらい前と比較したいかを入力します。比較するための時間の単位は、時間や月数など、データの粒度と同じです。
   - **[比較行の表示]** を選択すると、タイムライン上に比較期間を示す 2 行目が表示されます。

6. **定数**比較の場合:
   - 比較する値を入力します。このフィールドにはコンマや記号を含めないでください。パーセンテージを入力するは、パーセント記号を除いた数値を入力します。たとえば、目標が 25% の場合は 0.25 ではなく25 と入力します。有効なターゲット値を入力すると、プレビューが更新され、現在の値がターゲット値よりどれだけ大きいか、または小さいかが表示されます。
   - 比較の**ステータス**を設定して、比較値より上、下、またはその値であることが、良いこと、悪いこと、または中立であるのかを定義します。デフォルトでは、ステータスは中立に設定されています。メトリクスをプレビューすると、ステータスの違いによってメトリクスがどのように変わるか確認できます。

7. **[定義] > [メジャー]** で、定義に使用するメジャーをドロップダウンリストから選択します。このオプションは、選択したマークに複数のメジャーが関連付けられている場合にのみ表示され

ます。



メトリクスを完成させる

1. **[プロジェクト]** で **[場所の変更]** を選択すると、メトリクスの場所を変更できます。既定では、メトリクスはビューが属するプロジェクトに追加されます。

   プロジェクト内のすべてのメトリクスには、一意に名前を付ける必要があります。メトリクスの名前とプロジェクトは最初に選択したマークに基づいて設定されるため、別のユーザーがすでにそのマークからメトリクスを作成している場合、メトリクスを保存しようとすると競合が発生する可能性があります。別のプロジェクトや名前をメトリクスに選択するか、既存のメトリクスを上書きする場合は「メトリクスの上書き」を参照してください。

2. **[作成]** ボタンをクリックします。

   追加したプロジェクトのメトリクスへのリンクを含むメッセージが表示されます。

3. [パーミッションの設定] のガイドに従って、メトリクスのパーミッションが正しいことを確認します。

   デフォルトでは、メトリクスは作成されたプロジェクトの権限を継承します。メトリクスにアクセスできるユーザーは、接続されたビューまたはデータソースにアクセスするパーミッションがなくても、メトリクスのデータを表示できます。

メトリクスを作成すると、Tableau サイト上の他の独立したコンテンツを管理するのと同じ方法でメトリクスを管理できるようになります。メトリクスはビューから作成されますが、データ駆動型のアラートやサブスクリプションとは異なり、そのビューとは独立して存在します。接続されたビューを移動せずに、メトリクスを別のプロジェクトに移動できます。Tableau サイトでのコンテンツの管理については、「Web コンテンツの管理」を参照してください。

メトリクスを上書きする

メトリクスを一度作成したら、メトリクスの名前、説明、構成を変更することはできますが、メトリクスの定義方法を変更することはできません。メトリクスが使用するデータを変更する場合は、メトリクスを上書きする必要があります。メトリクスを上書きするには、メトリクスの所有者であるか、操作できる適切なパーミッションが付与されている必要があります。

1. メトリクスを上書きするには、上書きするメトリクスと同じ名前のメトリクスを同じプロジェクトに作成します。

   [メトリクスの上書き] ダイアログボックスが表示されます。

2. **[上書き]** ボタンをクリックします。

メトリクスを上書きしても、そのメトリクスをお気に入りに追加したユーザーに対して引き続き表示され、以前のメトリックのアクセス許可に加えられた変更は新しいメトリクスに適用されます。

メトリクスを作成できない場合

メトリクスに対応していないチャートでマークを選択すると、メトリクスを作成できない理由を説明するエラー メッセージが表示されます。次の表は、これらのシナリオをまとめたものです。

理由	シナリオ
正しいパーミッションがありません。	• ワークブックの所有者または管理者が、[Create/Refresh Metric (メトリクスの作成/更新)] 機能を拒否しました。詳細については、「パーミッション」を参照してください。
完全なデータにアクセスできません。	• 行レベルのセキュリティやユーザー フィルターにより、表示できるデータが制限されています。詳細については、「データ行レベルでのアクセスの制限」を参照してく

理由	シナリオ
	ください。
ワークブックのデータソースのパスワードが埋め込まれていないか、有効でなくなりました。	• ワークブックがパスワードを要求します。詳細については、「パブリッシュされたデータにアクセスするための認証資格情報を設定する」を参照してください。
データの粒度が正しくありません。	• グラフのデータは集計されません。メトリクスは、合計や平均などの集計を使用します。詳細については、「Tableau でのデータ集計」を参照してください。 • データブレンドの結果として、データのセルごとに複数の値があります。詳細については、「データブレンドのトラブルシューティング」を参照してください。
日付ディメンションはサポートされていません。	• チャートに日付部分と日付値の両方が含まれています。詳細については、「日付レベルの変更」を参照してください。 • 日付ディメンションでは、標準のグレゴリオ暦ではなく ISO 8601 暦を使用します。詳細については、「週ベースの ISO-8601 暦」を参照してください。 • 日付ディメンションは、月/年または月/日/年のカスタムレベルで集計されます。詳細については、「カスタム日付」を参照してください。

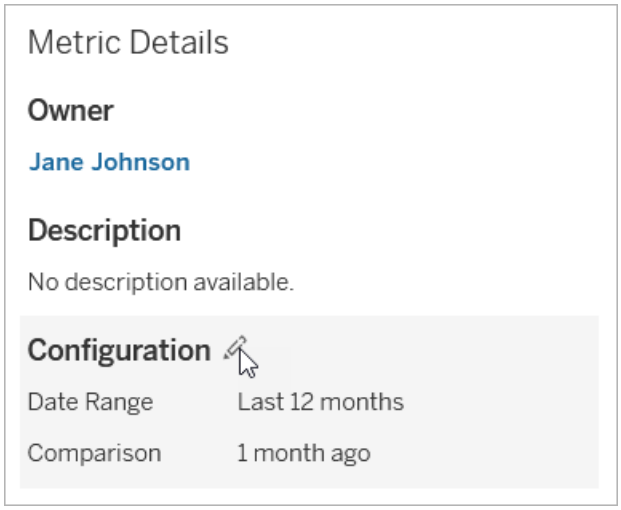
メトリクスの構成を編集する

2022.2 から、メトリクスの構成を変更できます。メトリクスの構成には、比較、日付範囲、およびステータス指標が含まれます。使用可能な構成オプションは、メトリクスのタイプによって異なります。タイムラインを使用したメトリクスを使用すると、履歴比較または定数比較を設定できます。単一数値のメトリクスでは、定数比較のみがサポートされます。

メトリクスの構成には、メトリクス定義 (メトリクス値を生成するメジャーと日付ディメンション) は含まれていません。定義を変更する場合は、メトリクスを新しいメトリクスで上書きします。

メトリクスの構成を編集するには、メトリクスの上書き機能が必要です。

1. 編集するメトリクスのメトリクス詳細ページを開きます。
2. 構成セクションにカーソルを合わせます。セクションの任意の場所をクリックすると、編集モードになります。



3. タイムライン メトリクスの場合、メトリクス カードと詳細に表示する日付範囲を設定します。
4. 比較タイプを選択します。定数比較の場合は、比較値とステータスを設定します。履歴比較の場合は、どれくらい前の値と比較するかを設定し、タイムライン上に比較線を表示するかどうかを選択します
5. **[保存]** をクリックします。構成の変更内容は、メトリクスを閲覧するすべてのユーザーに表示されます。

メトリクスの更新方法

メトリクスの更新時には、接続されたビュー (メトリクスの作成元のビュー) に新しいデータがないかどうかがチェックされます。データが変更されていない場合もあるため、更新の実行時にメトリクスの値が更新されるとは限りません。

メトリクスは、抽出の更新スケジュールに基づく頻度 (あるいは、ライブ データの場合は、60 分ごと) で更新されます。最後の更新時刻がメトリクスに表示されます。

失敗した更新の修正

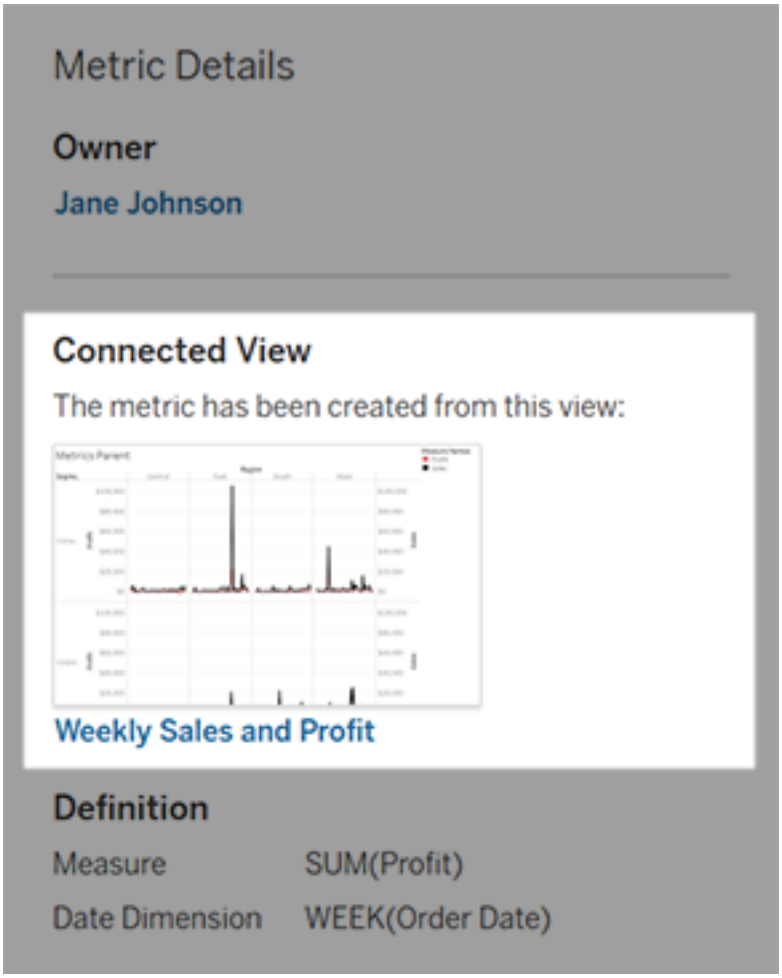
メトリクスが接続されているビューやその参照元データにアクセスできない場合、更新は失敗します。メトリクスの更新に失敗すると、失敗の時刻と影響を受けたメトリクスについての通知を受信

します。

メトリクスの更新は、次のいずれかの理由で失敗する場合があります。

- 接続されているビューが削除または変更されました。
- 接続されているビューのパーミッションが変更されました。
- データソースのパスワードが埋め込まれなくなったか、有効でなくなりました。
- メトリクスの所有者がメトリクスの更新に必要なサイトロールを持っていません。Creator または Explorer (パブリッシュ可能) のサイトロールが必要です。
- 一時的な接続の問題が起きています。この場合は時間が過ぎると解決されます。

障害の原因を特定するには、[メトリクスの詳細] を調べてください。メトリクスの所有者がメトリクスの更新に必要なサイトロールを持っていることを確認してください。次に、**[接続済みビュー]** を調べます。



接続済みビューがまだ一覧にある場合

ビューを開いて、エラーの原因を調査します。

ビューが読み込まれる場合は、メトリクスを定義するメジャーと(オプションの)日付ディメンションがビューにまだ存在することを確認します。

- ビューが変更されていないようである場合は、ビューからメトリクスを更新するパーミッションがない可能性があります。コンテンツ所有者または Tableau 管理者が [Create/Refresh Metric (メトリクスの作成/更新)] パーミッション機能を変更できます。詳細については、「パーミッション」を参照してください。
- メジャーが存在しない場合は、ビューが変更されているため、更新に必要なデータにメトリクスが接続できません。コンテンツ所有者または Tableau 管理者がリビジョン履歴を確認し、以前のバージョンを復元できます。詳細については、「リビジョン履歴の表示」を参照してください。

ビューが読み込まれず、代わりにパスワードの入力を求められるか、データソースへの接続時にエラーが表示される場合は、データソースのパスワードが埋め込まれていないか、無効になっています。コンテンツ所有者または Tableau 管理者がデータソース接続を編集し、パスワードを埋め込むことができます。詳細については、「接続の編集」を参照してください。

接続されているビューが一覧にない場合

ビューが削除されたか、ビューにアクセスするためのパーミッションがなくなりました。支援を受けるには、Tableau 管理者にお問い合わせください。

一時停止された更新の再開

更新の失敗が一定の回数に達すると、更新は一時停止されます。メトリクスの更新が一時停止されると、通知を受信します。

メトリクスの更新が一時停止されると、Tableau ではそのメトリクスについて新しいデータが取得されなくなります。更新が一時停止されたメトリクスでは、引き続き過去のデータが表示されます。

失敗の原因が修正された場合は、更新を再開できます。

1. 影響を受けたメトリクスを開きます。
2. 警告メッセージで、**[更新の再開]** をクリックします。

Tableau で更新が試行されます。この試行が成功すると、通知が送信され、更新はスケジュールに従って実行されるようになります。試行が成功しなかった場合、更新は一時停止されたままになります。

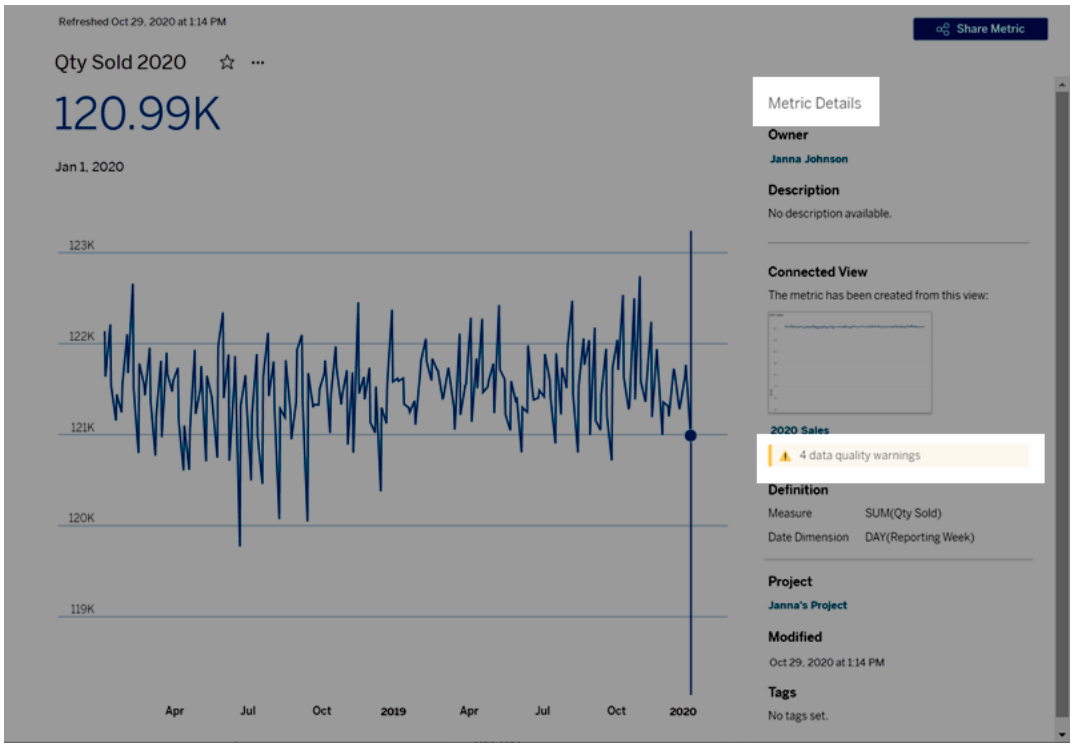
接続されているビューがまだ使用可能な場合は、メトリクスを上書きしてみてください。詳細については、「メトリクスを上書きする」を参照してください。それ以外の場合は、過去のデータの参照用にメトリクスを保持するか、メトリクスを削除できます。

> **注:** 更新に必要なサイトロールがないためメトリクスの更新が一時停止されている場合、更新を再開したり、メトリクスを削除したりすることはできません。

Tableau Catalog でのメトリクスの表示

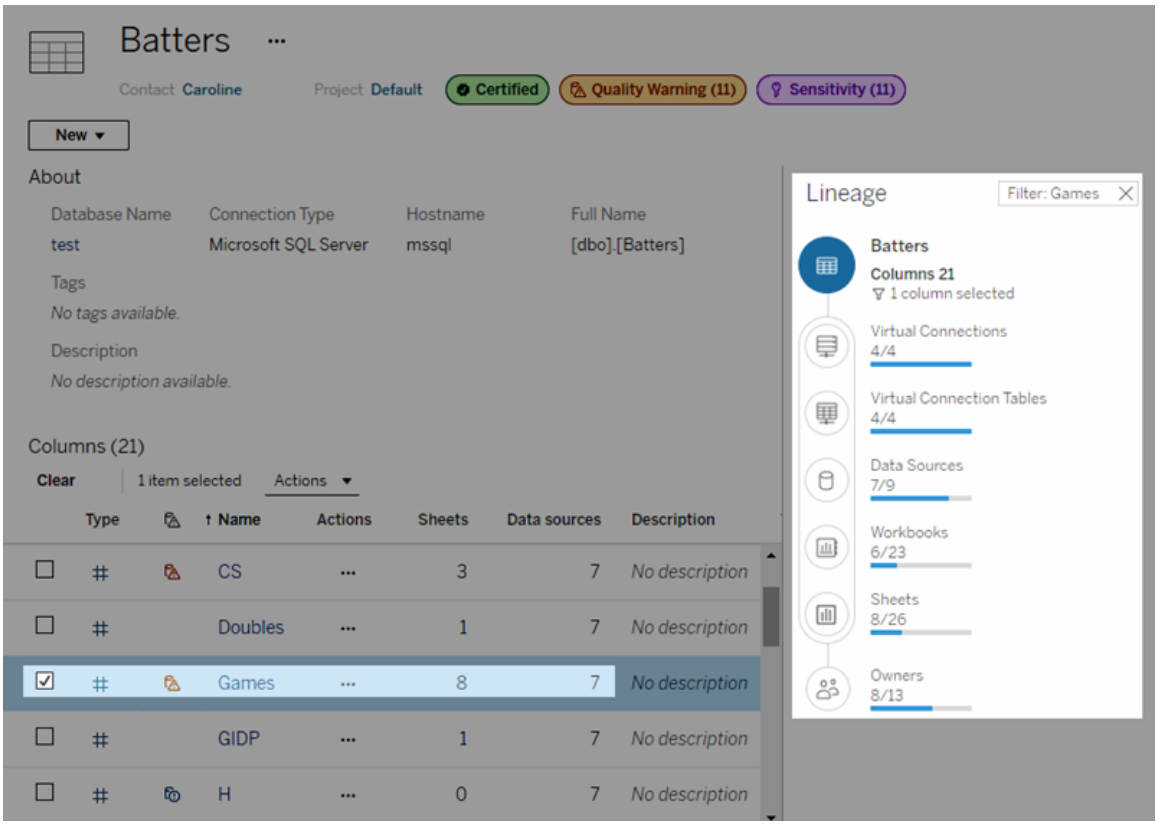
> 2019.3 以降、Tableau Catalog は、Tableau Cloud と Tableau Server に対するデータ管理で使用できます。Tableau Catalog が環境で有効になっている場合、Catalog の系列ツールでメトリクスを確認でき、データ品質に関する警告の影響を受けるメトリクスにそれらの警告が表示されます。Tableau Catalog の詳細については、Tableau Server または Tableau Cloud のヘルプの「Tableau Catalog について」を参照してください。

監視する数値に対してメトリクスを定義している場合は、そのメトリクスが依存するデータが何らかの影響を受けているかどうかを知ることが重要です。これを知るために Catalog を使用する方法がいくつかあります。まず、メトリクスに基づくデータに設定されたデータ品質に関する警告は、メトリクスに表示されます。これらの警告は、Tableau Mobile でメトリクスを開くと表示されるほか、Tableau Server および Tableau Cloud で、次に示すように、グリッドビューおよびメトリクスの詳細ページでメトリクスにカーソルを合わせると表示されます。



詳細については、Tableau Server または Tableau Cloud のヘルプの「データ品質に関する警告の設定」を参照してください。

Tableau Catalog の系列ツールを使用して、メトリクスが依存しているアップストリーム ソースを確認することもできます。インパクト分析を行うときに、特定の列またはテーブルが変更されているか非推奨になっている場合、または特定のワークブックが削除されている場合に影響を受けるメトリクスを確認できます。メトリクスを系列に含めると、Catalog では、変更が環境内のアセットに及ぼす影響の全体像が示されます。



詳細については、Tableau Server または Tableau Cloud のヘルプの「インパクト分析での系列の使用」を参照してください。

パブリッシュされたデータにアクセスするための認証資格情報を設定する

ワークブックを Tableau Cloud または Tableau Server にパブリッシュするときに、接続するデータソースをワークブックの一部 (ワークブックに*埋め込む*) として、または別のスタンドアロンデータソースとしてパブリッシュできます。また、パブリッシュするデータソースで認証が必要な場合、認証資格情報の取得方法をカスタマイズできます。

データソースに対する認証のタイプは、Tableau Server または Tableau Cloud サイトへのユーザーのサインイン方法からは独立しています。たとえば、ワークブック内のデータへの直接アクセスを許可

するには、データベースに、データ ソースの接続に対するユーザーの認証資格情報を埋め込みます。ただし、ワークブックを表示する人物がワークブックを開くには、Tableau Cloud または Tableau Server サイトにサインインできる必要があります。

このトピックでは、パブリッシュ プロセスの一部としてデータ接続で認証を設定する方法について説明します。

注: このトピックは、認証を必要としない接続 (テキスト ファイルや Excel ファイルなど) には適用されません。

認証タイプの設定

多くの接続タイプでは、データベースのユーザー名とパスワードを埋め込んだり、シングル サインオン (SSO) を使用したりできます。具体的な拡張子については、このトピックで後述します。

次の手順は、データ ソースまたはワークブックのパブリッシュの一部として認証を設定する方法について説明します。データ ソースの各接続に対してこれを実行できます。

1. [ワークブックのパブリッシュ] ダイアログ ボックスで、ワークブックの接続をリストしている **[データ ソース]** エリアに行き、**[編集]** を選択します。
2. **[データ ソースの管理]** ポップアップで、データ ソースを別にパブリッシュするか、ワークブックの一部としてパブリッシュするかを決定した後、データ ソースの各接続に対して認証タイプを選択します。使用できる認証タイプは接続タイプに依存し、次の項目が 1 つ以上含まれることがあります。
    - **ユーザーにメッセージを表示**: ビューまたはワークブックを読み込む際にユーザーがパブリッシュされたデータにアクセスするには、自分のデータベース認証資格情報を入力する必要があります。
    - **埋め込みパスワード**:接続に使用した認証資格情報は接続とともに保存され、自分がパブリッシュするデータ ソースやワークブックにアクセスするすべてのユーザーによって使用されます。
    - **サーバーの実行アカウント**: ユーザーの認証に単一の Kerberos サービス アカウントが使用されます。Windows の場合、これが Tableau Server の実行アカウントになります。Linux の場合、任意の Kerberos アカウントを使用できます。
    - **Viewer (ビューアー) 認証資格情報**: ビューアーの認証資格情報は、SSO (通常は Kerberos) を使用してデータベースに渡されます。

- **[Impersonate with embedded account (埋め込みアカウントを使用して偽装)]** または **[Impersonate with server Run As service account (実行サービス アカウントを使用して偽装)]**: 埋め込みの認証資格情報を使用した偽装を使用すると、埋め込みの認証資格情報に接続し、ビューアーの ID に切り替えます (これをサポートするデータベースのみ)。実行サービス アカウントを使用する偽装は類似していますが、ビューアーの ID に切り替える前にまず Kerberos サービス アカウントに接続します。
- **[更新を有効にしない]** または **[更新のアクセスを許可]**: これらのオプションは Salesforce などのクラウドデータの抽出をパブリッシュするときに表示され、参照元のデータにアクセスするにはデータベースの認証資格情報が必要です。**[更新のアクセスを許可]** は、接続に認証資格情報を埋め込み、定期的なスケジュールでその抽出の更新を設定できるようにします。

重要: 抽出したデータの鮮度をどのように保つかも要素となります。

- 自動更新スケジュールを設定する場合は、接続にパスワードを埋め込む必要があります。
- クラウドデータ接続を Tableau Cloud にパブリッシュするときに、Tableau Cloud をデータプロバイダーの承認済みリストに追加する必要がある場合は、パブリッシュ ステップでアラートが表示されます。
- Kerberos 委任され、行レベルでセキュリティで保護されたデータ ソースより作成した抽出をパブリッシュすることはできません。

Dropbox、OneDrive 接続

Dropbox と OneDrive の場合、Tableau でデータ ソースやワークブックをパブリッシュして **[埋め込みパスワード]** を選択すると、保存した認証資格情報が作成され、そのデータ ソースやワークブックに埋め込まれます。

Tableau データ ソースへのワークブックの接続

Tableau Cloud または Tableau Server データ ソースに接続したワークブックをパブリッシュする場合は、参照元のデータ ソースにアクセスするための認証資格情報を設定する代わりに、接続先のパブリッシュされたデータ ソースにワークブックがアクセスできるかどうかを設定します。元のデータ型に関係なく、サーバー データ ソースの選択肢を常に**埋め込みパスワード**とするか、**ユーザーに要求**します。

ユーザーにプロンプトを表示するよう選択している場合、ワークブックを開くユーザーがデータを表示するには、データ ソースの **[表示]** および **[接続]** パーミッションが必要です。パスワードの埋め込みを

選択すると、ユーザーは表示または接続パーミッションを持っていなくてもワークブックの情報を表示できます。

仮想接続

Tableau Cloud、Tableau Server 2022.3、および Tableau Desktop 2022.4 以降、仮想接続を使用するデータソースやワークブックなどの Tableau コンテンツをパブリッシュして、**[埋め込みパスワード]** または **[埋め込み認証資格情報]** を選択すると、コンテンツのビューアーは、仮想接続に接続したり、仮想接続のクエリを実行したりするための権限を持つことになります。ただし、仮想接続に関連付けられているデータポリシーは、自身の ID ではなく、ビューアーの ID を使用して常に評価されます。

たとえば、仮想接続を使用するワークブックをパブリッシュしたとします。ワークブックのビューアーが仮想接続を介してデータに接続したり、データを照会したりできるようにするには、仮想接続に接続したり、仮想接続を照会したりするためのパーミッションを埋め込みます。そうすると、仮想接続に関連付けられているデータポリシーによって、ワークブックのビューアーが機密データにアクセスすることが防止されます。

仮想接続のテーブルを表示したり、そのテーブルにアクセスしたりできるかどうかの評価には、コンテンツ作成者の ID が使用されます。ただし、仮想接続のテーブルに関連付けられているデータポリシーを評価するときは、ビューアーの ID が使用されます。また、コンテンツ作成者は、仮想接続に対して、編集パーミッションではなく、接続パーミッションのみを埋め込むことができます。

パーミッションを埋め込まない場合は、ワークブックまたはデータソースにアクセスするパーミッションを持つユーザーと、仮想接続に接続するパーミッションを持つユーザーのみがワークブックまたはデータソースにアクセスできます。

Tableau Cloud 2022.2, Tableau Server 2022.1、および Tableau Desktop 2022.3 以前では、仮想接続の [埋め込みパスワード] および [埋め込み認証資格情報] オプションは機能しません。2022.3 (Tableau Cloud および Tableau Server) または 2022.4 (Tableau Desktop) にアップグレードする前にこれらのオプションを選択した場合、オプションは、アップグレードした後に適切に機能するようになります。その後、仮想接続を照会するためのパーミッションを埋め込むことが可能になります。

関連項目

- Tableau Server にパブリッシュする場合は、Tableau Server ヘルプの「接続の編集」を参照してください。
- Tableau Cloud にパブリッシュし、ワークブックを Salesforce、Google Analytics、Google スプレッドシート、Google BigQuery、OneDrive、Dropbox、および QuickBooks Online データに接続する場合は、Tableau Cloud ヘルプの「保存した認証資格情報を使用してデータを更新する」を参照してください。
- Tableau Server 管理者が認証の詳細をお探しの場合には Tableau Server のヘルプトピック「認証」(Windows | Linux) と「データ接続認証」(Windows | Linux) を参照してください。

データガイドでダッシュボードを探検する

Tableau の新しいダッシュボードを見つけたとき、データの意味や使い方に疑問を持ったことはありませんか? また、新しいダッシュボードをパブリッシュしたとき、ダッシュボードの使い方について説明を加えられないかと思ったことはありませんか?

データガイドは、ダッシュボードについての有用な情報と、その背後にあるデータに関する洞察を提供します。データガイドは、ダッシュボードの作成者がエンドユーザーに対して、説明やリソースへのリンクなど、より分かりやすいコンテキストをダッシュボードに直接提供できるようにするためのものです。また、データガイドは、データの説明を見るによって提供されるインサイトを自動的に表示するため、ユーザーは外れ値を見つけたり、マークの説明について学習したりすることができます。「データの説明を見る」を利用した「Viz の説明を見る」は、異常値を示すメジャーとその背後にある潜在的なキードライバーを特定します。

これらのコンテキストに関連する詳細を確認すると、ダッシュボードのユーザーは、新しいダッシュボードをより簡単に操作することができ、より速くインサイトを見つけ、正しいデータを見ていることを確信し、データにコンテキストを持たせ、自信を持って Viz を理解できるようになります。

データガイドを作成者としてカスタマイズする

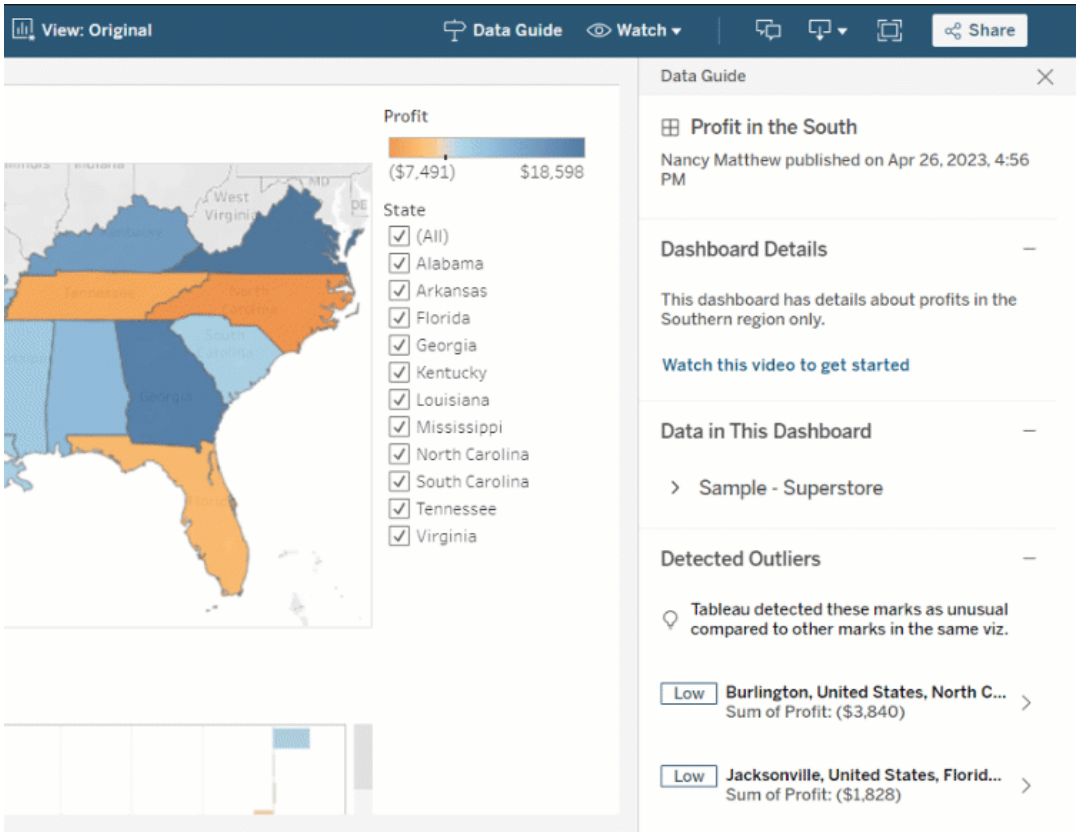
1. ダッシュボードから**[編集]** ボタンを選択します。
2. ツールバーの **[データガイド]** を選択します。
3. [データガイド] ペインに、エンドユーザーがダッシュボードの目的を理解するのに役立つ説明を入力します。
4. **[リンクの追加]** を選択し、リンクテキストとURL の説明ラベルを入力します。

5. **[OK]** を選択します。
6. Viz (ダッシュボードのシート) を選択します。データガイドは自動的に更新されるので、ダッシュボードの各 Viz に関連する説明やリソースを追加することができます。
7. 説明を入力し、その Viz に関連するリンクを追加します。
8. **[OK]** を選択します。

ヒント: データガイドを使用してカスタム代替テキストを記述し、Viz のアクセシビリティを向上させるには、「より多くのテキストを表示して分かりやすいビューにする」参照してください。

データガイドをダッシュボードのユーザーとして探索する

1. ダッシュボードで、ツールバーの **[データガイド]** を選択します。
2. ダッシュボードの説明を読み、ダッシュボードの作成者が提供するリソースを検索します。
3. **[このダッシュボードのデータ]** と **[検出された外れ値]** を展開し、ダッシュボードで使用されている参照元のデータについて詳細を表示します。
4. Viz (ダッシュボード内のオブジェクト) を選択します。
5. Viz の説明を読み、ダッシュボードの作成者が提供するリソースを探索します。
6. **[データサマリー]** と **[検出された外れ値]** を展開すると、その Viz で使用されているデータの詳細が表示されます。
7. **[このマークのデータ]** と **[適用フィルター]** に関する情報を表示するには、グラフの棒やマップの地域など、マーク (または複数のマーク) を選択します。単一のマークを選択すると、その値について考えられる **[説明]** が表示されます。



データガイドをさまざまなレベルで探索する

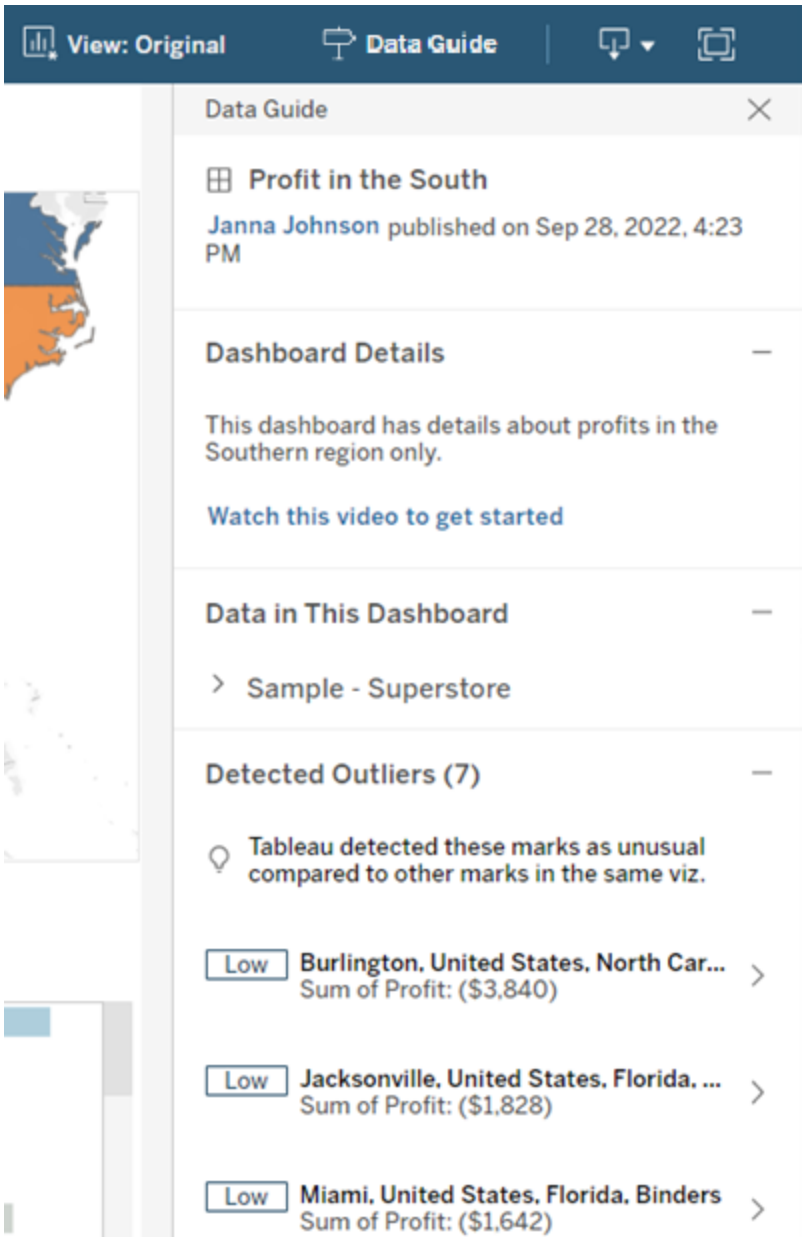
既定では、ワークブックを開いたとき、データガイドは閉じられています。データガイドは、ツールバーでを開くことができます。[データガイド] ペインの上部には、選択したダッシュボードまたは Viz の名前が表示されます。ダッシュボードの作成者は、ダッシュボードと、ダッシュボードを構成する個々の Viz の両方に説明を書くことができます。

ダッシュボードレベルの詳細を理解する

ダッシュボードレベルでは、データガイドで次を行うことができます。

- ダッシュボードの名前、作成者、最終パブリッシュ日を表示する。
- ダッシュボードの作成者が書いた説明や、ビデオや Wiki ページなどの関連リソースへのリンクを含める。

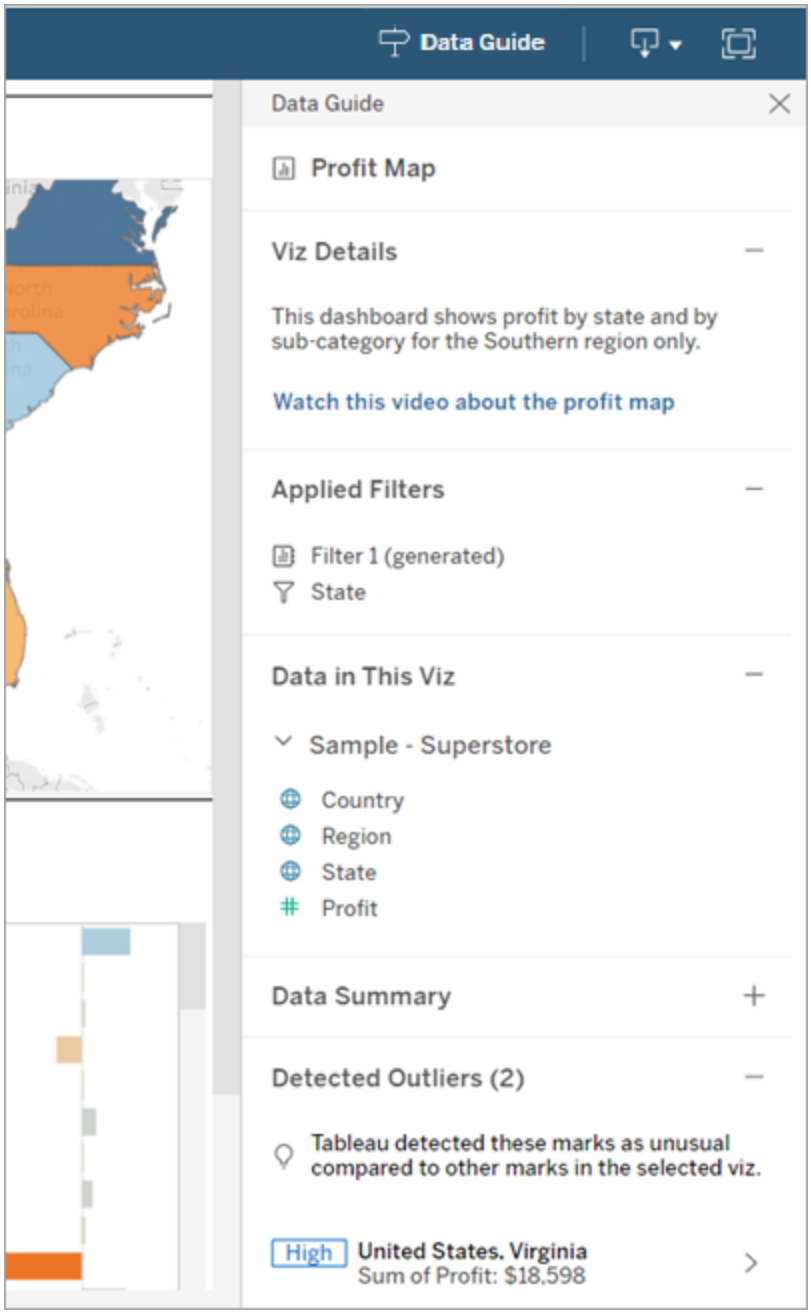
- ダッシュボードが使用するデータソースや、どのディメンションやメジャーが使用されているかなど、データに関する詳細を一覧表示する。
- 「データの説明を見る」で特定された、ダッシュボード内の検出された外れ値を明らかにする。詳細については、「「データの説明を見る」を使い始める」を参照してください。



Viz レベルの詳細を理解する

ユーザーが特定の Viz (ダッシュボードオブジェクト) を選択すると、データガイドは以下を行います。

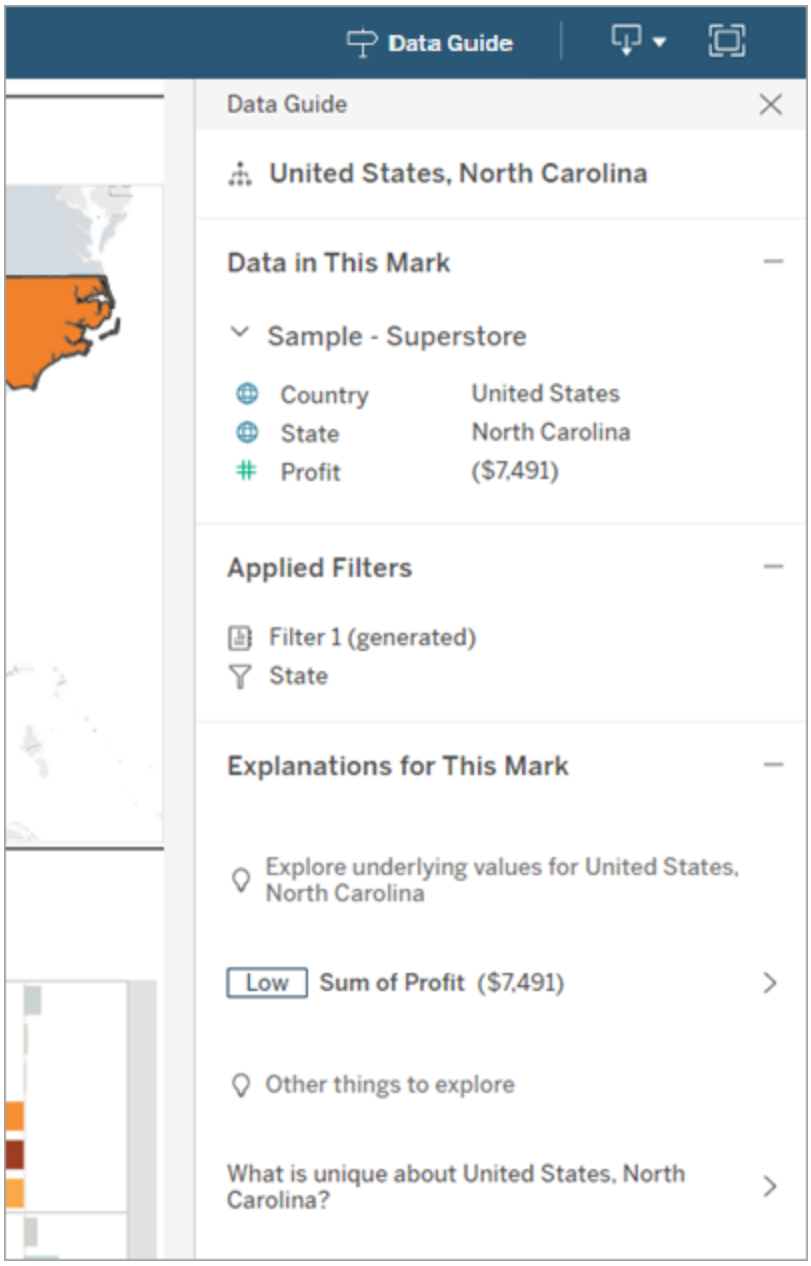
- Viz の名前を表示する。
- ダッシュボードの作成者が書いた説明や、ビデオや Wiki ページなどの関連リソースへのリンクを含める。
- ダッシュボードで使用するデータソース (使用するディメンションとメジャーを含む) を一覧表示する。
- 適用されたフィルターと Viz で使用するデータを一覧表示にすることで、ユーザーが操作するフィルターによってデータがどのように影響されるかを特定する。ユーザーが変更できるフィルターのみが表示されます。
- データポイントの数や並べ替えの詳細など、Viz で使用されるデータの概要を表示する。
- 「データの説明を見る」で特定された、Viz で検出された外れ値を明らかにする。詳細については、「「データの説明を見る」を使い始める」を参照してください。



マークレベルの詳細を理解する

ユーザーがマップ上で地域などのマークを 1 つまたは複数選択すると、データガイドは以下を行います。

- マークの名前を表示する。
- 適用されたフィルターとViz で使用されたデータを一覧表示し、何が含まれ、何が除外されたかがわかるようにする。
- 「データの説明を見る」で特定された、Viz で検出された外れ値を明らかにする。詳細については、「「データの説明を見る」を使い始める」を参照してください。



データガイドの表示制御

既定では、ワークブックを開いたとき、データガイドは閉じられています。データガイドを開いたり閉じたりしても、サイトではすべてのワークブックがこの状態になります。2023.1 以降、サーバー管理者は、すべてのユーザーに対してデータガイドを非表示にすることができます。これを選択すると、サイト上のすべてのワークブックの Tableau ツールバーから[データガイド] ボタンが削除されます。データガイドを非表示にするには:

1. 左側のペインで、**[設定]** を選択します。
2. **[全般]** タブで **[Availability of Data Guide (データガイドの利用可否)]** にスクロールします。
3. **[非表示]** を選択します。
4. **[保存]** を選択します。

クエリキャッシュとビューの高速化のデータ鮮度ポリシーの設定

クエリキャッシュのデータの鮮度を理解する

ワークブックを作成し、チームがそのワークブックを気に入っているとします。しかし、最新のデータを Viz に表示するには、[更新] ボタンをクリックする必要がある場合があります。ライブ接続を使用してワークブックを作成したのに、なぜデータを更新する必要があるのでしょうか。その答えはパフォーマンスです。

パフォーマンスを向上させるために、Tableau はデータのフェッチに使用されたクエリの結果をキャッシュします。これにより、以降のアクセスで、キャッシュされたデータをより迅速に再利用して返すことができます。[更新] ボタンをクリックすると、更新されたデータを取得できますが、パフォーマンスコストが増大する可能性があります。

データのパフォーマンスと鮮度のバランスをとるには、ワークブックのデータ鮮度ポリシーを設定します。データ鮮度ポリシーを設定すると、指定した時間にデータが更新されます。Tableau は、キャッシュに保存されたデータが設定した鮮度ポリシーを満たしていない場合、そのデータを視覚化しません。

ビューの高速化のデータの鮮度を理解する

ビューの高速化機能では、Tableau は選択したワークブックを事前に計算してビューを生成し、読み込み時間を大幅に短縮します。事前計算スケジュールは、選択したワークブックに設定したデータ鮮度ポリシーまたは抽出スケジュールに基づいて作成され、パフォーマンスと鮮度の両方を備えたデータを提供します。ビューの高速化は Tableau Server でのみ利用できます。

リソースの消費を少なくするために、実行できる事前計算ジョブの数は 1 日あたり12 ジョブに制限されています。たとえば、データ鮮度ポリシーが 2 時間未満に設定されている場合、ビューの高速化によるパフォーマンスの利点が得られるのは 1 日の最初の 12 回の更新に限られます。

ワークブックに最適なデータを選択する

データをキャッシュに保存せずに、常に最新のデータを表示することを望むユーザーがいる一方で、オーバーヘッドを削減してワークブックのパフォーマンスを向上させるために大きなキャッシュを望むユーザーもいます。データ鮮度ポリシーを設定する最初のステップは、ビジネスに適したデータを決定することです。

Tableau Cloud では、キャッシュに保存されたデータが既定で 12 時間ごとに更新され、ワークブックの所有者はワークブックレベルでデータ鮮度ポリシーを設定できます。

Tableau Server では、サーバー管理者はサーバー上のすべてのサイトに既定のキャッシュポリシーを設定でき、ワークブックの所有者はワークブックレベルでデータ鮮度ポリシーを設定できます。

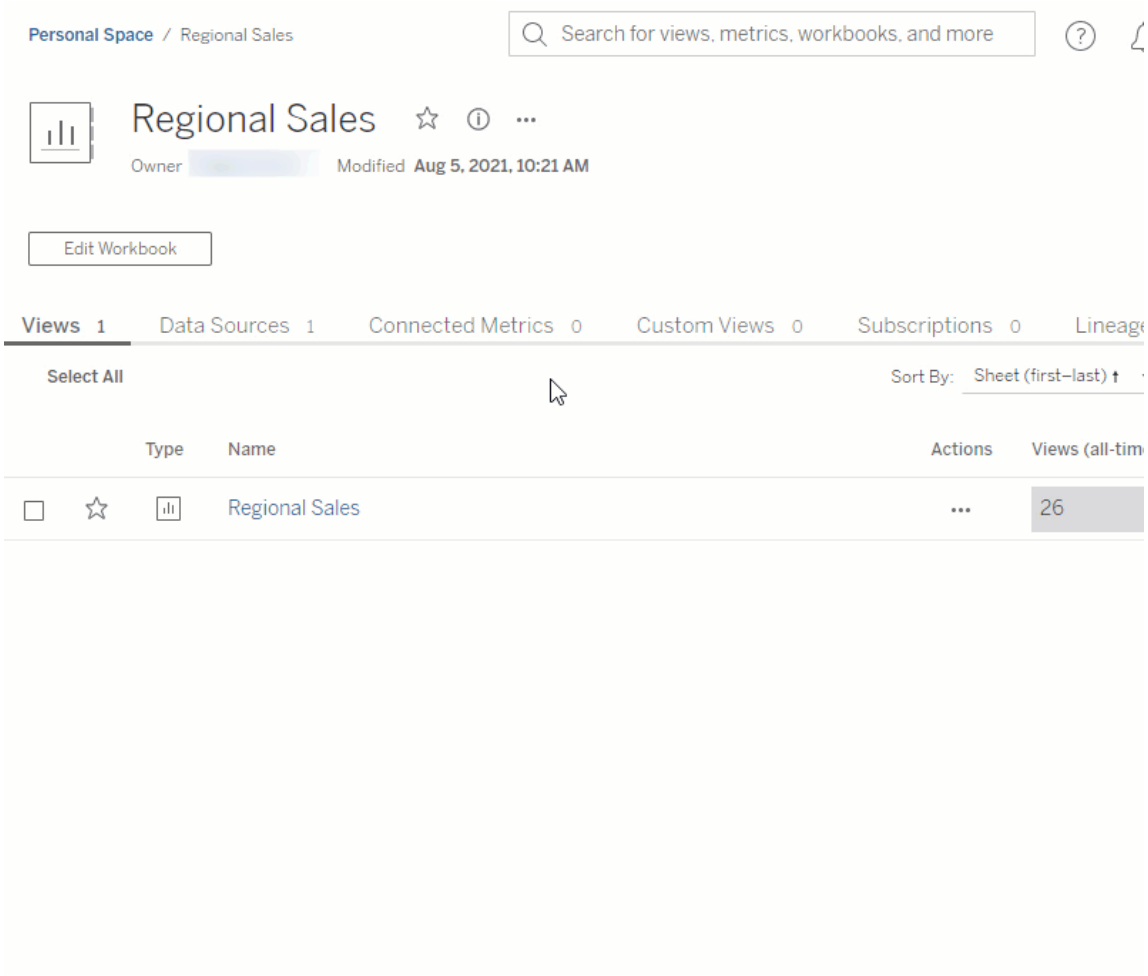
> **注:** データ鮮度ポリシーは、Tableau Desktop または抽出とファイルベースのデータソースを使用するワークブックでは使用できません。

ワークブックのデータ鮮度ポリシーを編集する

ワークブックのデータ鮮度ポリシーを編集するには、ワークブックの所有者であることに加えて、データソースへのライブ接続がワークブックに必要です。

1. Tableau Cloud または Tableau Server でサイトにサインインします。
2. [ホーム] または [探索] ページから、ポリシーを設定するワークブックに移動します。

3. 詳細アイコン ⓘ をクリックします。
4. [ワークブックの詳細] ダイアログで、**[Edit Data Freshness Policy (データ鮮度ポリシーの編集)]** をクリックします。
5. 次のいずれかのオプションを選択します。
   Site default (12 hours) (サイトの既定値 (12時間))
   Always live (Tableau will always get the latest data) (常にライブ (Tableau は常に最新のデータを取得します))
   Ensure data is fresh every (データの更新頻度)
   Ensure data is fresh at (データの更新日時)
6. **[OK]** をクリックします。



[Site default (サイトの既定値)] では、12 時間ごとにデータが更新されます。オーディエンスがダッシュボードを定期的に使用しているが、最新のデータ鮮度を必要としない場合に最適なオプションです。

[Always live (常にライブ)] では、常に最新のデータが提供されるため、読み込み時間が長くなる可能性があります。

[Ensure data is fresh every (データの更新頻度)].では、データを更新する頻度を、分、時間、日、または週単位で指定できます。

[Ensure data is fresh at (データの更新日時)] では、データを更新する日時をスケジュールできます。毎週月曜日、水曜日、金曜日の午前 9 時 (太平洋時間) に重要な会議がある場合は、毎週月曜日、水曜日、金曜日の午前 8 時 45 分にデータを更新するように設定すると、会議の開始時に最新のデータを用意することができます。

動的な軸の範囲の使用

データを正しく分析できるように、Viz で表される範囲をユーザーが理解することは重要です。特に、ダッシュボード内の複数の Viz や複数のワークシートを同時に分析する場合、Viz によって軸の範囲が異なるとユーザーがデータを誤解しやすくなります。たとえば、2 つの棒グラフが隣り合って表示される場合、両方のグラフの棒は同じサイズに見えることがあります。しかし、軸の範囲が大きく異なる場合があり、グラフが誤解を招く可能性があります。

ユーザーが軸の範囲を理解しやすいように、2023.3 以降、作成者は、軸の範囲の最小値と最大値を数値パラメーターまたは日付パラメーターで設定できる「動的な軸の範囲」を使用することができます。その後、ユーザーが Viz 間を移動すると、軸が同期して更新されます。これにより、ユーザーは複数の Viz にわたるデータを簡単かつ正確に分析できるようになります。また、データの範囲を制限することで、参照元のデータをフィルタリングしたり、データの移動平均に影響を与えたりすることなく、データのサブセットを表示できます。

動的な軸の範囲を使用すると、参照元データをフィルタリングすることなく軸の範囲を広くしたり狭くしたりすることもできます。これは、目標に対する進捗状況を示したり、移動平均を示したりするのに最適です。

サポートされているフィールドタイプ

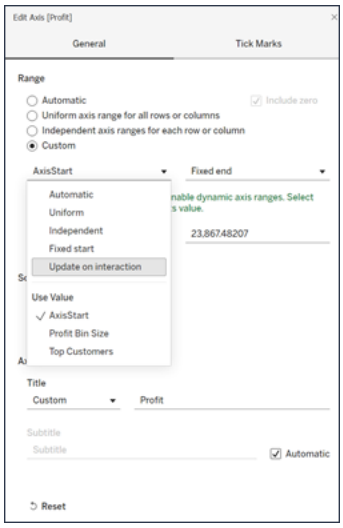
動的な軸の範囲は、選択した連続軸と互換性のあるパラメーターに対応しています。次に例を示します。

- 数値パラメーター
- 時間パラメーター(日付または日時)

選択内容に基づいた動的な軸範囲の設定

Tableau 25.3 以降では、領域の選択に基づいて軸範囲を自動的に更新できるため、パラメーターとパラメーター アクションのワークフローを個別に作成する必要がなくなりました。

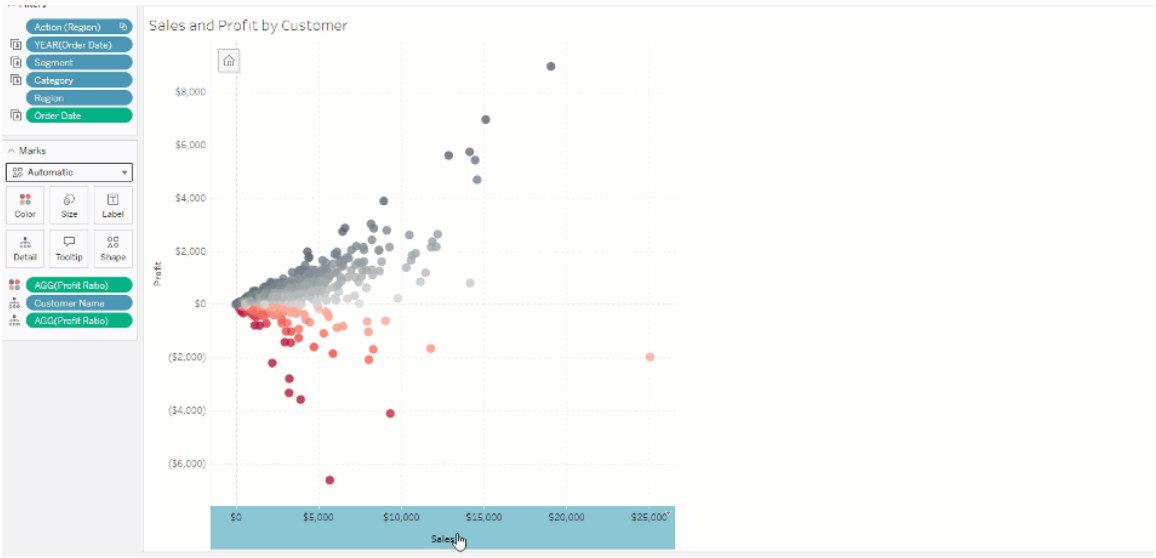
1. ワークシートから軸を右クリックし、**[軸の編集]** を選択します。
2. [範囲] で、**[カスタム]** を選択します。
3. ドロップダウン メニューを展開し、開始範囲と終了範囲のポイントに対して **[操作時に更新]** を選択します。
4. ダイアログを閉じます。
5. ビジュアライゼーションに戻り、Vz 上の領域を選択し、選択した範囲ポイントに対応する値を設定します。



既存のパラメーターを使用した動的な軸の範囲の設定

動的な軸の範囲は、既存のパラメーターに関連付けて構成することもできます。

1. ワークシートで、軸のパラメーターを作成します。
2. 軸を右クリックし、**[軸の編集]** を選択します。
3. [範囲] で、**[カスタム]** を選択します。次に、開始軸範囲または終了軸範囲に対して作成したパラメーターを選択します。
4. ダイアログ ボックスを閉じます。
5. ダッシュボードで複数のシートを使用している場合は、異なるシートの軸に対してこれらの手順を繰り返します。



制限とエッジ ケースについて

動的な軸の範囲は、ストーリー内では更新されません。軸に使用されているパラメーターが削除された場合、最新のパラメーター値が軸範囲として残ります。次回軸を編集するときに、使用する新しいパラメーターを選択するように求めるエラー メッセージが表示されます。

動的な軸タイトルの使用

Tableau の柔軟性により、作成者はさまざまなデータセットをすばやく分析し、データを別の方法で視覚化してより深い分析を行うことができます。ただし、その柔軟性には、Viz で使用されるデータを正確に伝える責任が伴います。たとえば、パラメーター値に応じて変化する測定単位を伝えるこ

とは重要です。天気予報の Viz で、25 度という気温が表示がされているとします。その予報によると、華氏では、素晴らしいスキー旅行になるかもしれません。しかし、摂氏では、海水浴へ行くことになるかもしれません。

作成者は、動的な軸タイトルを使用すると、パラメーターの値または単一値フィールド(LOD 計算など) に基づいて軸タイトルを更新できます。パラメーターのスワップを使用すると、取り扱っているデータに応じて軸のタイトルを更新できます。

サポートされているフィールドタイプ

動的な軸タイトルとして使用するには、フィールドが次の条件を満たす必要があります。

- 単一値と固定 LOD 計算
- パラメーター
- 定数の計算フィールド
- 上位 1 セット

動的な軸タイトルの設定

1. Tableau シートから、連続フィールドをシェルフにドラッグします。
2. 軸をダブルクリックして **[軸の編集]** ダイアログを開きます。
3. **[軸のタイトル]** セクションで、軸のタイトルに使用するフィールドをリストから選択します。
4. **[軸の編集]** ダイアログを閉じます。

より複雑なユースケースの場合は、まず「例: パラメーターを使用したメジャーのスワップ」の手順に従います。次に、以下の手順を実行します。

1. X 軸をダブルクリックして **[軸の編集]** ダイアログを開きます。
2. **[軸のタイトル]** セクションで **[パラメーター]** を選択し、[プレースホルダー 2] セレクターを選択します。
3. **[軸の編集]** ダイアログを閉じます。
4. プレースホルダー 1 を使用し、これらの手順を Y 軸に対して繰り返します。

これで、パラメーターが変更され、軸のタイトルが表示データに応じて更新されます。

制限とエッジケースについて

動的な軸のタイトルは、[表示形式] を使用して Viz タイプを変更するとクリアされます。また、サブタイトルを使用する場合、カスタム タイトルの場合と同様に、サブタイトルは動的な軸タイトルの後

に表示されます。自動サブタイトルは、少なくとも 2 つの期間を持つ単一の年にフィルター処理された、連続日付値の軸がある場合にのみ設定されます。動的な軸のタイトル機能は、ストーリー (情報を伝えるために連携して機能する一連のビジュアライゼーション) でワークシートを使用する場合は機能しません。

Dynamic Zone Visibility の使用

ダッシュボードスペースは、特にデータに関するインサイトを累進的に明らかにしたい場合に重要になります。Dynamic Zone Visibility を使用すると、フィールドやパラメーターの値に基づいてゾーン (ダッシュボードのタイル エレメントまたは浮動エレメント) を隠したり、見せたりできます。ダッシュボードの操作に伴って、ダッシュボード上のゾーンが表示または非表示になります。その結果、望ましいレイアウトが損なわれない動的なダッシュボードが実現します。

ダッシュボードでボタンをクリックすることで、オブジェクトを表示または非表示にすることもできますが、Dynamic Zone Visibility を使用すると、オブジェクトが自動的に表示または非表示になります。この動作は、さまざまなユーザーグループがダッシュボードを使用する場合に最適です。たとえば、ダッシュボードにアクセスしてきたユーザーグループに応じて、異なるゾーンを表示したい場合があるかもしれません。

その場合は、Dynamic Zone Visibility でパラメーター アクションを使用することもできます。たとえば、ユーザーが Viz のマークをクリックすると、それまで非表示になっていたゾーンが表示されます。この機能を使用すると、より深いレベルのデータを表示するタイミングを選択できるようになるため、複雑なダッシュボードに最適です。

サポートされているフィールド タイプ

Dynamic Zone Visibility でフィールドを使用するには、フィールド パラメーターは次の要件を満たす必要があります。

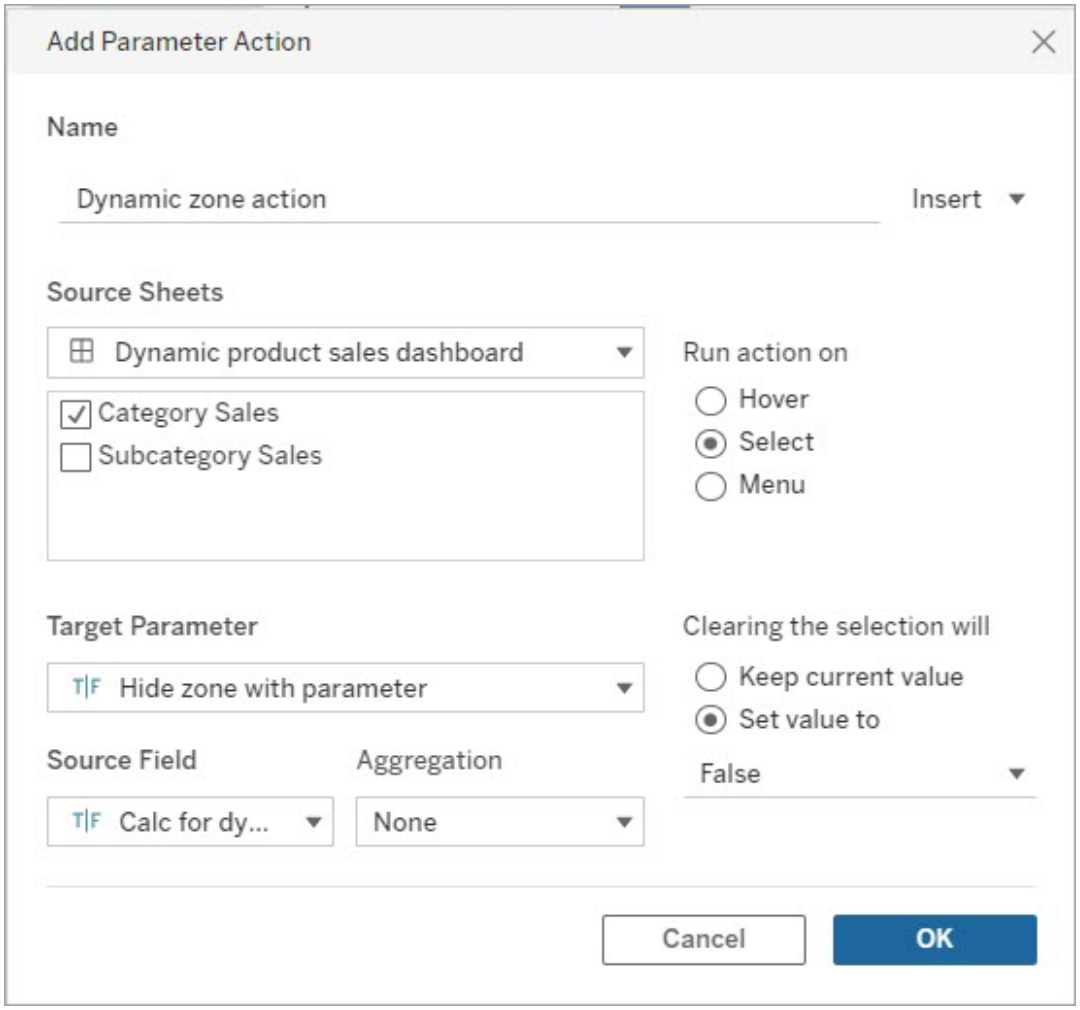
- ブール値。
- 1 つの値。
- Viz と無関係。つまり、フィールドは、固定された詳細レベル (LOD) の計算など、Viz の構造とは関係がない定数値を返します。

動的なダッシュボードゾーンを構成する

次の例には、Superstore (スーパーストア) データを使用する 2 つのシートがあります。最初のシートには、カテゴリ別の売上が示された棒グラフが、2 つ目のシートには、サブカテゴリ別の売上が示された棒グラフがあります。Dynamic Zone Visibility を使用すると、[Sales by Category (カテゴリ別の売上)] ゾーンのマークがクリックされた後にのみ、2 つ目のシートが表示されます。この例では、パラメーター アクションのソース フィールドとして使用されるブール型の計算フィールドを利用しています。パラメター アクションのソース フィールドとして計算を使用するには、計算を [マーク] カードに追加する必要があります。

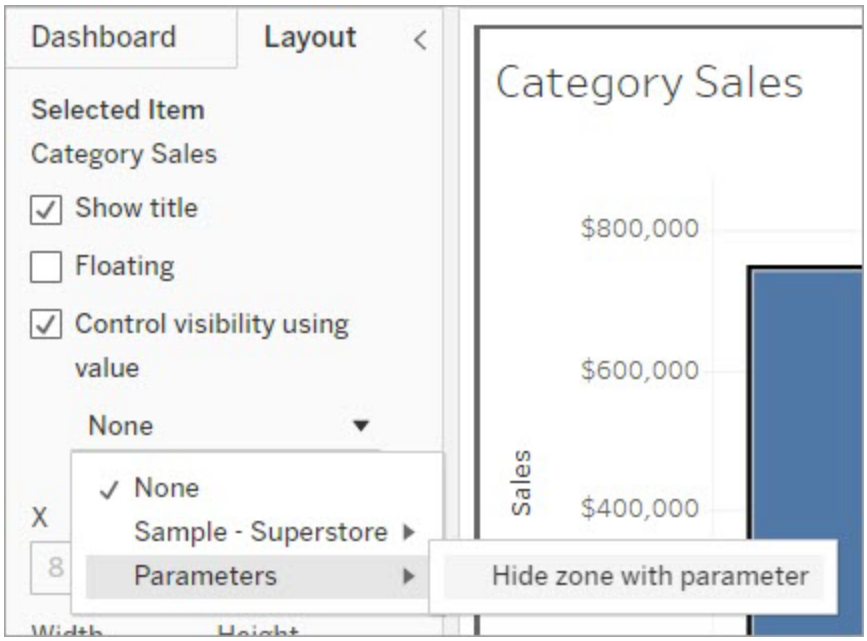
1. [Sales by Category (カテゴリ別の売上)] シートから、パラメーターを作成します。この例では、パラメーターの**データ型**を**ブール値**に設定する必要があります。
2. [Sales by Category (カテゴリ別の売上)] シートから、計算フィールドを作成します。この例では、次の計算を使用します。`True`
3. [Sales by Category (カテゴリ別の売上)] シートで、作成した計算を [マーク] カードの **[Details (詳細)]** にドラッグします。
4. ダッシュボードを作成します。
5. 常に表示するシートをダッシュボードにドラッグします。この例では、[Sales by Category (カテゴリ別の売上)] を常に表示します。
6. ダッシュボードで、[Sales by Category (カテゴリ別の売上)] ゾーン (ダッシュボード オブジェクト) をクリックします。次に、ダッシュボードの [ワークシート] メニューから、パラメーター アクションを作成します。この例では、次の **[Change Parameter (パラメーターの変更)]** アクションを使用しています。
   ソース シートが、作成したダッシュボードと [Sales by Category (カテゴリ別の売上)] シートを使用するように設定されました。
   [ターゲット パラメーター] は、作成したパラメーターです。

[ソース フィールド] は、作成した計算です。

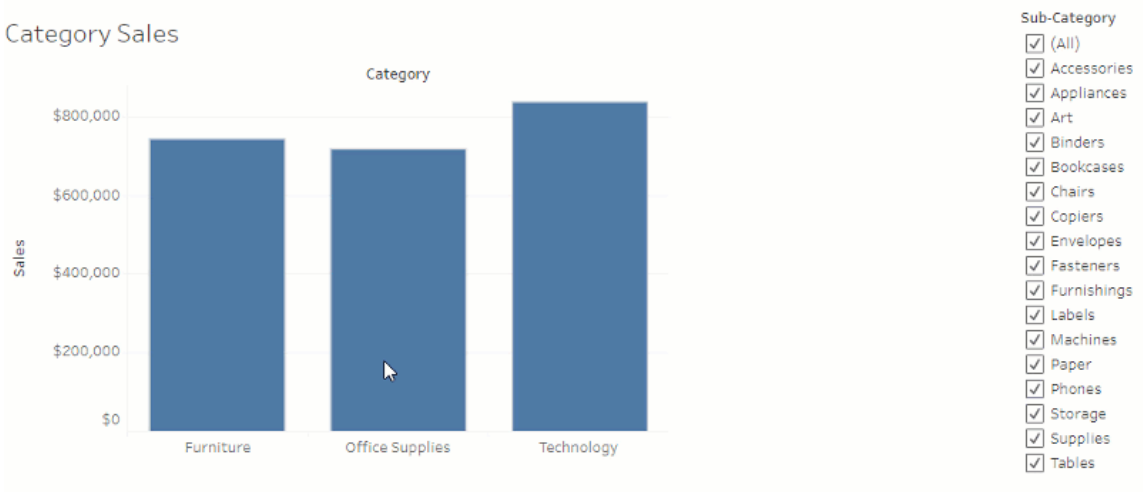


7. [Sales by Sub-Category (サブカテゴリ別の売上)] ワークシートをダッシュボードにドラッグします。
8. [Sales by Category (カテゴリ別の売上)] ゾーンをクリックします。右上のドロップダウン矢印をクリックして、**[フィルターとして使用]** を選択します。
9. [Sales by Sub-Category (サブカテゴリ別の売上)] ゾーンをクリックしてから、**[レイアウト]** タブをクリックします。
10. **[Control visibility using value (値を使用して表示を制御)]** ボックスをオンにします。
11. ドロップダウンから、ゾーンの表示を制御するために作成したパラメーターを選択します。

注: [Control visibility using value (値を使用して表示を制御)] のオプションに目的のフィールドが含まれていない場合は、フィールドが、サポートされているフィールドタイプであることを確認します。



これで、[Sales by Category (カテゴリ別の売上)] ゾーンのカテゴリマークをクリックすると、ダッシュボードに [Sales by Sub-Category (サブカテゴリ別の売上)] ゾーンが表示されるようになりました。



制限とエッジ ケースについて

動的ゾーン表示は、ストーリー(情報を伝えるために連携して機能する一連のビジュアライゼーション)でワークシートを使用する場合は機能しません。

Microsoft Teams 用 Tableau アプリのセットアップ

Microsoft Teams 用 Tableau アプリを使用すると、ユーザーは仕事の流れを妨げることなく、Tableau ビュー(ビジュアライゼーション)とメトリクスを操作できます。Tableau 管理者は、アプリをインストールして設定し、既存の Tableau サイトを Microsoft Teams の環境と連携させます。Tableau Cloud 用アプリは、Microsoft Apps Marketplace から直接インストールできます。Tableau Server 用アプリをインストールするには、アプリをダウンロードしてサイドロードします。

Microsoft Teams 用 Tableau アプリの詳細については、「Microsoft Teams での Tableau コンテンツへのアクセス」を参照してください。この記事では、Tableau アプリのセットアップについて説明します。

関連項目

- GitHub の Tableau アプリに関する FAQ
- GitHub のリリース ノート
- Youtube の入門チュートリアル

前提条件

Microsoft Teams 用 Tableau アプリを設定するには、次のアプリとアカウントが必要です。

- Tableau Cloud または Tableau Server バージョン 2023.1 以降
- Tableau Cloud のサイト管理者アカウントまたは Tableau Server のサーバー管理者アカウント
- Microsoft 365 for Enterprise サブスクリプション
- Microsoft Teams
- Microsoft 365 管理者アカウント(組織のポリシーでアプリをインストールするために管理者アカウントが必要な場合)
- ユーザー マッピング属性のいずれかに基づいて相互に関連付けられた、Tableau と Microsoft Teams の両方のユーザー ライセンス

ユーザー マッピング属性

Tableau でのシングル サインオンを可能にするには、Microsoft Teams のユーザー アカウントを Tableau のユーザー アカウントにマッピングします。このマッピングは、Microsoft Teams SDK または ユーザーの Microsoft Entra プロファイルの属性を使用して設定できます。

Microsoft Teams SDK の属性

属性	説明
Preferred_Username	Microsoft Teams ユーザーのメール アドレス。この値は、Microsoft Entra で定義されているメールと異なる場合があります。
User Principal Name	ユーザーが Microsoft Entra にログインする主な方法。

Microsoft Entra プロファイルの属性

属性	説明
Primary Email	`user.mail` 属性に対応し、ユーザーのメール アドレスを表します。
Mail Nickname	`user.mailNickname` 属性に対応し、ユーザーのエイリアスを表します。
Employee ID	`user.employeeId` 属性に対応し、組織によって割り当てられた従業員 ID を表します。
On-Premise Distinguished Name	`user.onPremiseDistinguishedName` 属性に対応し、オンプレミスの Active Directory から同期された識別名 (DN) を表します。
On-Premise User Principal Name	`user.onPremiseUserPrincipalName` 属性に対応し、オンプレミスの Active Directory から同期された userPrincipalName を表します。
On-premise SAM Account Name	`user.onPremiseSamAccountName` 属性に対応し、オンプレミスの Active Directory から同期された samAccountName を表します。

Extension Attribute X	Microsoft Entra では、ユーザーの Entra プロファイルに最大 15 個の属性を追加できます。Tableau ユーザー名が Microsoft Entra に存在しない場合は、拡張機能属性を使用して各 Entra ユーザーの Tableau ユーザー名を格納します。

Tableau Server の追加の前提条件

Tableau アプリは API 呼び出しを使用して Tableau Server と通信し、インタラクティブなコンテンツを埋め込みます。これらの呼び出しは、Tableau Server に到達できる必要があります。

- Tableau Server がファイアウォールの内側にある場合、4.152.0.0/15 および 20.2.0.0/16 の IP 範囲からの内向きの HTTP 要求を許可します。
- ユーザーが Microsoft Teams にアクセスするのと同じ方法で Tableau Server にアクセスできる必要があります。
  - ユーザーが VPN を使用して Microsoft Teams にアクセスする必要がある場合、同じ VPN を使用して Tableau Server にアクセスできる必要があります。
  - ユーザーがパブリック インターネットから Microsoft Teams にアクセスできる場合、Tableau Server はパブリック インターネットからアクセスできる必要があります。
  - ユーザーがモバイル アプリ経由で Microsoft Teams にアクセスできる場合、Tableau Server がモバイル デバイスからアクセスできる必要があります。

Tableau Server 用 Tableau アプリのインストール

Tableau Server 用 Tableau アプリをインストールするには、アプリマニフェストを編集してホスト名をカスタマイズしてから、アプリマニフェストを Microsoft Teams にアップロードします。

> Tableau Cloud 用 Tableau アプリをインストールするには、「Tableau Cloud セットアップ ガイド」を参照してください。

1. アプリマニフェストをダウンロードして編集する

1. GitHub から、「アプリマニフェストをダウンロード」します。
2. マニフェスト ファイル `tableau-server-app-for-teams.zip` からファイルを抽出します。

3. テキストエディターで manifest.json ファイルを開きます。

4. manifest.json で *.online.tableau.com を検索し、そのドメインを Tableau Server 環境のホスト名に置き換えます。Tableau Server の URL が https://analytics.company.com の場合、ホスト名は analytics.company.com になります。

5. 変更内容を manifest.json に保存します。

6. tableau-app-for-teams-server フォルダー内のファイルを圧縮して、変更内容を含む zip ファイルを作成します。

2. アプリマニフェストをアップロードする

組織のアプリポリシーによって、Microsoft Teams アプリをインストールできるかどうかが決まります。アプリをインストールするには、Microsoft 365 管理者アカウントが必要になる場合があります。

1. 「Microsoft Teams 管理センター」を開きます。

2. ナビゲーション パネルで、**[Teams アプリ]**、**[アプリの管理]** の順に選択します。

3. **[アクション]**、**[新しいアプリのアップロード]** の順に選択します。

4. 編集した tableau-server-app-for-teams.zip ファイルをアップロードします。アプリを管理するためのリンクが記載された確認メッセージが表示されます。

3. ユーザー用にアプリをインストールしてピン留めする

アプリをインストールすると、すべてのユーザーまたは一部のユーザーが利用できるようにすることができます。アプリをピン留めして、ユーザーのアプリバーに表示することもできます。

1. 「Microsoft Teams 管理センター」を開きます。

2. ナビゲーション パネルで、**[Teams アプリ]**、**[セットアップ ポリシー]** の順に選択します。

3. **[インストール済みアプリ]** で **[アプリの追加]** を選択し、Tableau アプリを追加します。

4. 必要に応じて、**[ピン留めされたアプリ]** で **[アプリの追加]** を選択し、Tableau アプリを追加します。

5. 変更内容を保存します。

Tableau アプリを追加した後、ユーザーがアプリを利用できるようになるまでに最大 24 時間かかる場合があります。

Tableau アプリを設定する

Tableau アプリをインストールすると、アプリを設定するように Microsoft Teams から求められます。アプリを Tableau サイトに接続するためのサイト情報を設定します。

1.接続済みアプリを作成する

Microsoft Teams 用 Tableau アプリは、接続済みアプリを使用して、シングル サインオンでユーザーを認証します。Tableau Cloud または Tableau Server でこの接続済みアプリを作成します。

1. Tableau Cloud または Tableau Server で、直接信頼の接続済みアプリを作成します。詳細については、「Tableau Cloud で直接信頼を使用して接続済みアプリを設定する」または「Tableau Server で直接信頼を使用して接続済みアプリを設定する」を参照してください。
    - 接続済みアプリのドメイン許可リストを、指定したドメインのみに制限する場合は、次のドメインを許可リストに追加します。
        - `tableau-ms-teams-prod-gycea7csh5hsbfh5.a02.azurefd.net`
        - `teams.microsoft.com`
        - `*.teams.microsoft.com`
    - ユーザーが他の Microsoft プラットフォームからアプリにアクセスする場合は、「Microsoft 365 URL」の一覧からその他のドメインを追加します。

2. 接続済みアプリを有効にします。

3. 接続済みアプリのシークレットを生成します。

4. 接続済みアプリのクライアント ID、シークレット ID、シークレット値をコピーします。これらの値を保存します。これらは、Microsoft Teams で Tableau サイトを設定する際に必要になります。

2.Tableau アプリにサイト設定を追加する

Tableau アプリを開いたとき、まだアプリが設定されていない場合は、セットアップ画面が表示されます。

1. **[ラベル]** で、サイトに名前を付けます。この名前は、Microsoft Teams アプリのユーザーに表示されます。ラベルを入力しない場合、アプリはサイト名を表示します。このフィールドはオプションです。

2. **[Tableau サイト名]** に、サイトの URL に表示されているとおりの名前を入力します。たとえば、URL が `https://us-west-2b.online.tableau.com/#/site/mysitename/home` の場合、「`mysitename`」と入力します。アプリを Tableau Server に接続していて、既定のサイトを使用する場合は、このフィールドを空白のままにします。

3. **[ベース URL]** には、サイトの URL の `tableau.com` で終わる部分を入力します (例: `https://us-west-2b.online.tableau.com`)。

4. **[ユーザー マッピング属性]** で、シングル サインオンでのユーザーの認証に使用する Microsoft Entra 属性を選択します。

5. **[接続済みアプリ ID]**、**[接続済みアプリ シークレット ID]**、**[接続済みアプリ シークレット値]** に、接続済みアプリを作成したときにコピーした値を貼り付けます。

6. **[サイト設定の追加]** を選択します。

設定の詳細を保存する前に、Tableau アプリが接続をテストします。アプリが Tableau に接続できない場合、サイト設定は追加されません。その場合、接続済みアプリが認証できない、接続済みアプリが有効になっていない、サイト情報が正しくない、または使用しているアカウントが Tableau 管理者アカウントではないことが原因である可能性があります。また、アプリがファイアウォールの背後にあるため、Tableau Server に到達できない場合もあります。

サイトの設定を追加または削除する

サイト設定を追加した後、その他のサイト設定を追加したり、既存のサイト設定を削除したりできます。

1. Microsoft Teams 用 Tableau アプリで、**[設定]** タブを選択します。
2. [設定] で **[Tableau サイトの設定]** タブを選択します。このタブの設定にアクセスできるのは Tableau 管理者のみです。
3. サイトを追加するには、**[サイトの詳細の追加]** を選択します。サイトを削除するには、そのサイトの削除アイコンを選択します。

Microsoft Teams での Tableau コンテンツへのアクセス

Microsoft Teams 用 Tableau アプリを使用すると、ビュー (ビジュアライゼーション) とメトリクスを操作し、メッセージ、チャネル、ミーティングで共有できます。Microsoft Teams で Tableau アプリを使用できるようにするには、まず自分か管理者が Tableau アプリをインストールする必要があります。Tableau アプリのインストールとセットアップの詳細については、「Microsoft Teams 用 Tableau アプリのセットアップ」を参照してください。

アプリには、[Tableau]、[Pulse]、[設定] の 3 つのタブがあります。[Tableau] タブには、お気に入りとして追加したビューが表示されます。[Pulse] タブには、フォローしているメトリクスと、それらのメトリクスに関するインサイトの概要が表示されます。[Pulse] タブは Tableau Cloud サイト用にのみ表示されることに注意してください。[設定] タブでは、コンテンツを検索する際に Microsoft Teams がチェックするサイトを設定できます。また、Tableau 管理者の場合はサイト構成を追加および削除できます。

お気に入りのビューやフォローしているメトリクスがない場合、アプリにコンテンツは表示されません。ビューをお気に入りとして追加したり、メトリクスをフォローしたりするには、Tableau サイトに直接サインインします。詳細については、「お気に入りとしてマークする」および「メトリクスをフォローする」を参照してください。他のタイプのコンテンツをお気に入りとして追加することは可能ですが、ビュー以外のお気に入りのコンテンツは Tableau アプリに表示されません。ワークブックのすべてのビューを表示するには、個々のビューをお気に入りに追加します。

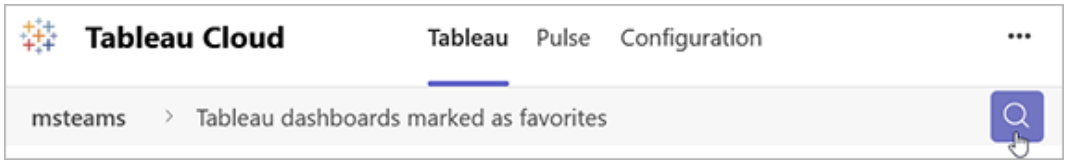
コンテンツを探索する

Tableau または Pulse タブで、ビューまたは メトリクスのカードをクリックすると、完全なインタラクティブ バージョンが開きます。インタラクティブなビューまたは メトリクスを使用すると、データの詳細を確認したり、さまざまなディメンションや時間範囲に基づいてデータをフィルタリングしたりすることができます。

コンテンツを検索する

ビューまたは メトリクスの名前がわかっている場合は、Tableau アプリで検索できます。検索バーにビューの URL を入力して、特定のコンテンツやカスタム ビューにアクセスすることもできます。

1. [Tableau] タブまたは [Pulse] で検索アイコンを選択します。



2. ビューまたは メトリクスの名前を入力します。

3. ビューまたは メトリクスの名前の隣にある [開く] ボタンを選択して、インタラクティブ バージョンに移動します。

メトリクスをフォローする

各 メトリクスは、固有のフィルターの組み合わせを表しています。既存の メトリクスをフィルタリングするときに、そのフィルターの組み合わせを持つ メトリクスがまだ存在しない場合は、新しい メトリクスを作成します。新しい メトリクスをフォローすると、[Pulse] タブとインサイトのサマリーに表示されます。

1. メトリクスで、フォロー (このメトリクスをサブスクライブする) アイコンを選択します。

メトリクスのフォローを解除するには、ボタンをもう一度選択します。

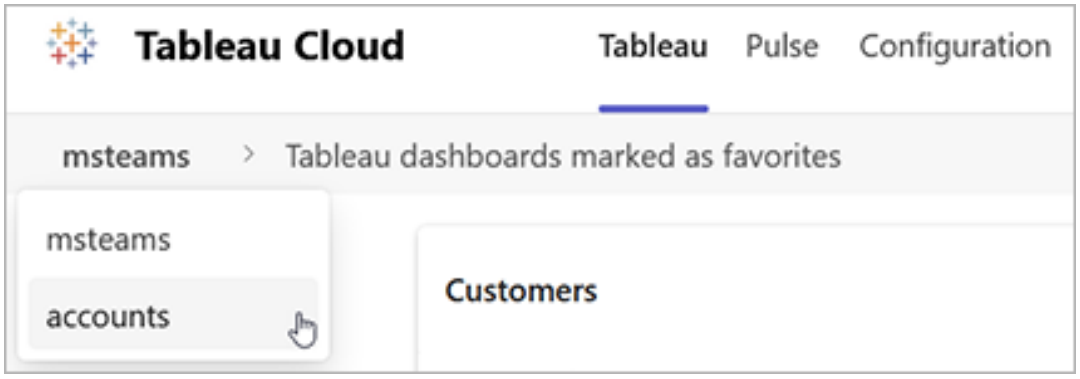
コンテンツを共有する

1. ビューまたはメトリクスで、共有アイコンを選択します。

2. 共有するユーザー、グループ チャット、またはチャネルを選択します。
3. **[共有]** を選択します。

サイトを切り替える

1. [Tableau] タブまたは [Pulse] タブで、パンくずリストからサイト名を選択します。Tableau アプリに複数のサイトが接続されている場合は、サイトのリストが表示されます。

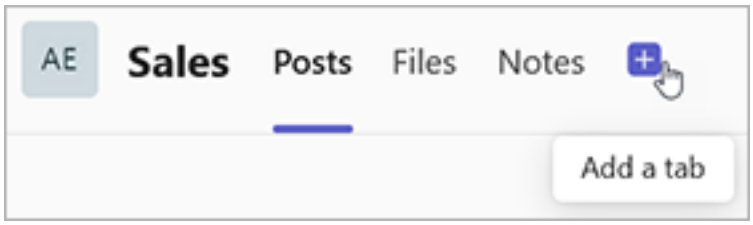


2. 切り替えるサイトを選択します。

チャネルやミーティングへの Tableau コンテンツの追加

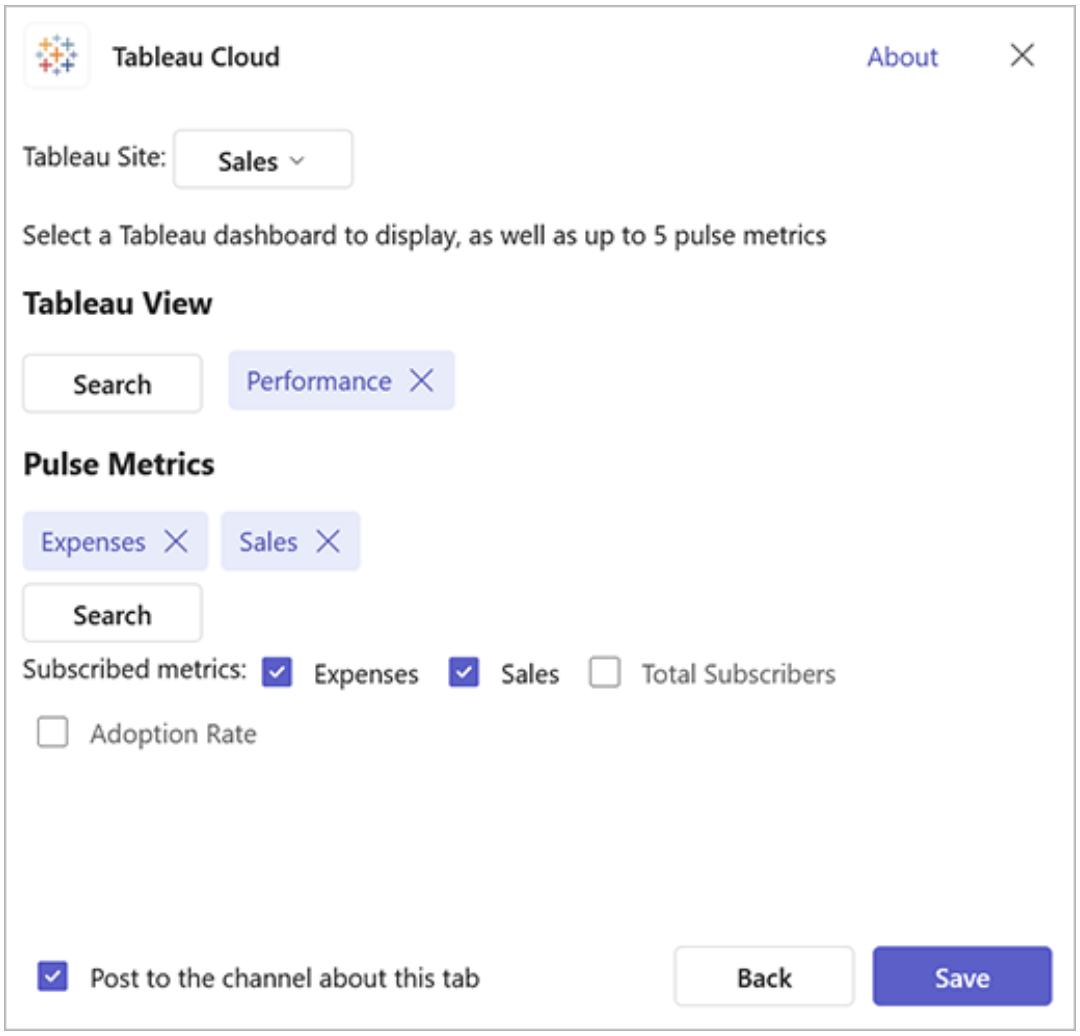
チャネル タブにコンテンツをピン留めする

1. Tableau コンテンツをピン留めするチャネルで、**[+]** アイコンを選択してタブを追加します。



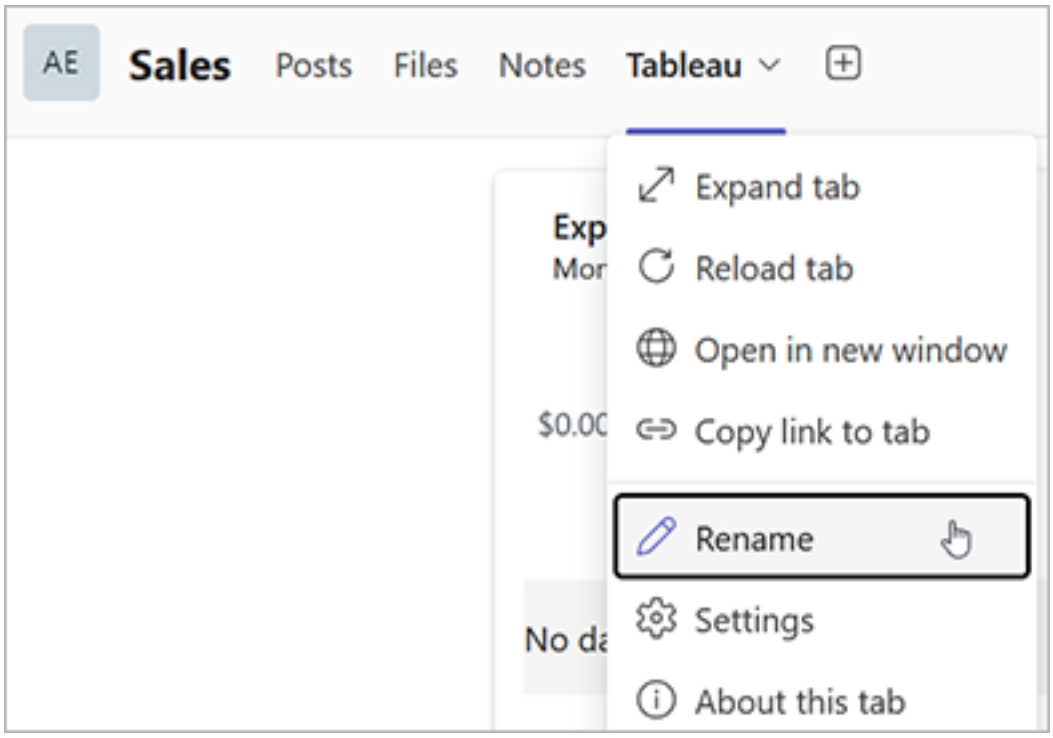
2. アプリのリストから、[Tableau アプリ] を選択します。

3. ピン留めするコンテンツを選択します。カスタム ビューをピン留めするには、検索 ボタンを選択してビューの URL を入力します。タブごとに 1 つのビューと最大 5 つのメトリクスをピン留めできます。さらに多くの Tableau コンテンツをピン留めするには、2 つ目のタブを追加します。



4. **[保存]** を選択します。

5. タブの名前を変更する必要がある場合は、タブのドロップダウン メニューを選択して **[名前の変更]** を選択します。



ミーティングに Tableau コンテンツを追加する

最初にミーティングを作成してから、Tableau コンテンツを追加する必要があります。ミーティングの作成中にサードパーティ製アプリを追加することはできません。

1. カレンダー アプリで、Tableau コンテンツを追加するミーティングを見つけ、**[編集]** を選択します。
2. **[その他のオプション]** を選択します。
3. ツールバーから **[アプリ]** を選択します。
4. アプリのリストから、[Tableau アプリ] を選択します。
5. ピン留めするコンテンツを選択します。
6. **[保存]** を選択します。
7. ミーティングへの変更を保存します。

8. ミーティングを開始したら、[Tableau] タブを選択します。これにより、Tableau コンテンツがサイドパネルで開き、自分だけが表示できるようになります。

9. サイドパネルで [ステージングに共有] ボタンを選択します。これにより、ミーティングの他のユーザーにコンテンツが表示されるようになります。共有したコンテンツは、画面共有のしくみと同様に、自分が見たとおりに他の人にも表示されることに注意してください。データに行レベルのセキュリティがある場合、ミーティングの参加者全員にデータのビューが表示されます。

メッセージで Tableau コンテンツを共有する

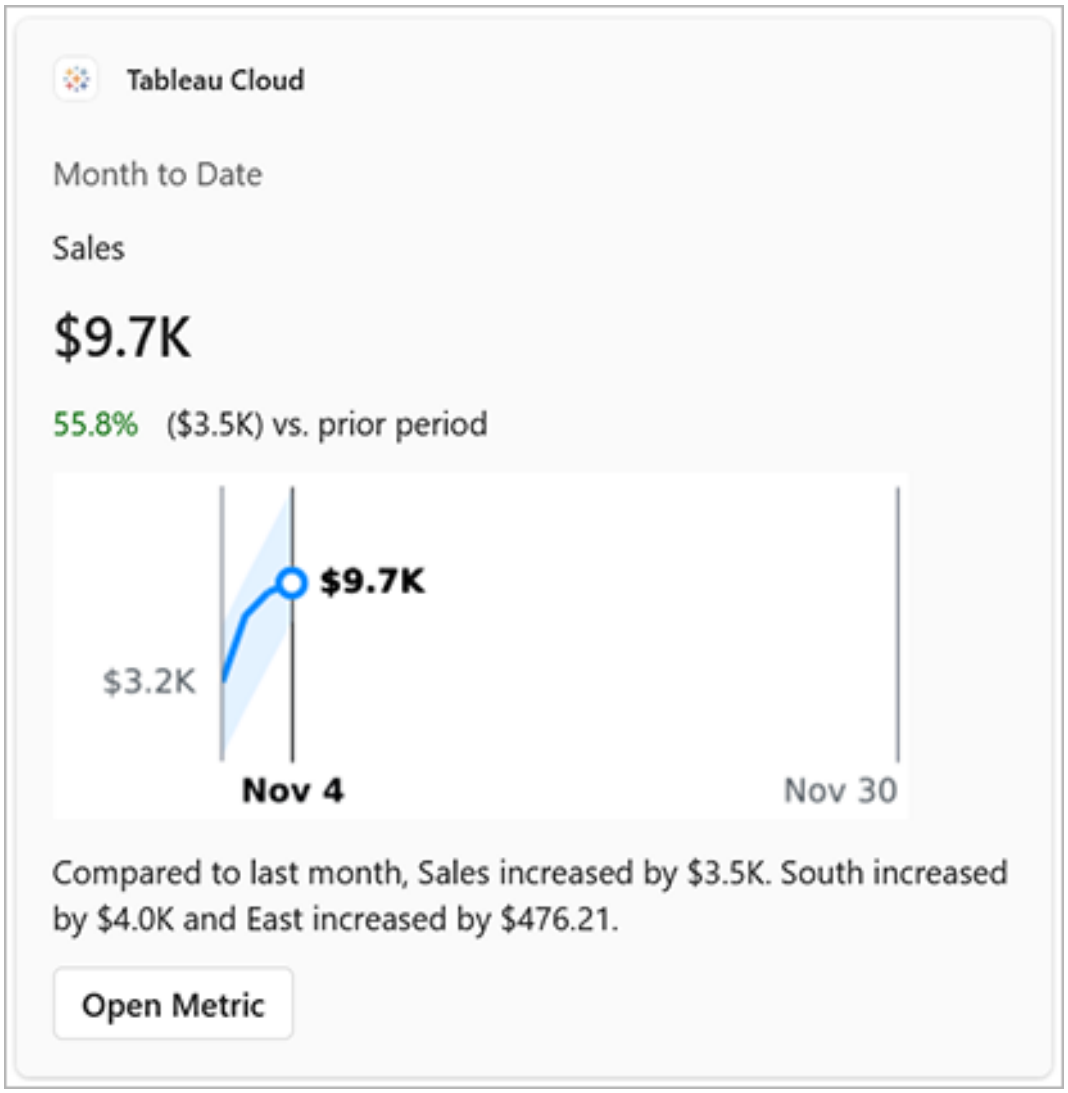
ビューまたはメトリクスの URL がわかっている場合は、その URL をメッセージに貼り付けることで、そのコンテンツをすばやく共有できます。Tableau メッセージ拡張機能を使用すると、リンクがカードに変換され、コンテンツのタイトルとサムネイルが表示され、ボタンを使用して完全なインタラクティブビューまたはメトリクスを開くことができます。送信するコンテンツの URL がわからない場合は、共有するコンテンツを検索できます。

1. メッセージで、**[+]** (アクションとアプリ) アイコンを選択します。

2. [Tableau アプリ] を選択します。

3. 共有するビューまたはメトリクスを Tableau サイトで検索します。

4. ビューまたはメトリクスを選択します。このコンテンツのカードを生成するには少し時間がかかります。



5. メッセージを送信します。

サーバーの管理

Tableau Server をインストールすると、サーバーのカスタマイズや管理を行うことができます。たとえば、セキュリティ、ライセンス、サイト、サブスクリプション、データドリブン アラートなどを管理できます。

セキュリティ 1316

サポートされているデータ ソース 1559

認証資格情報の一元管理 1569

関連項目 1570

ステップ 1: Salesforce に接続されたアプリを作成する 1572

ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する 1574

サイトのカスタム OAuth の構成 1575

アクセス トークンの管理 1577

ステップ 1: Azure 用の OAuth クライアントを登録する 1578

ステップ 2: Azure 用に Tableau Server を構成する 1580

サイトのカスタム OAuth の構成 1584

OAuth クライアントを Snowflake に登録する 1588

オプション 1: TSM を使用して Snowflake 接続の OAuth を設定する 1588

オプション 2: サイトごとに Snowflake 接続の OAuth を構成する 1591

Tableau Server (バージョン 2023.3 以降) 1595

サイトのカスタム OAuth の構成 1597

Tableau Server (バージョン 2023.1 以前) 1597

Customer Data Platform で OAuth を使用する 1601

HANA で IdP を設定する 1604

IdP を設定する 1605

HANA に接続する 1606

Okta 1606

Snowflake で IDP を設定する 1607

Tableau で IDP を設定する 1607

Snowflake に接続する 1608

Okta 1608

Tableau Server で Hyper Query Processing を有効にする 1609

ステップ 1: IDP を構成する 1611

AWS で IDP を構成する 1612

Redshift ユーザーのロールを構成する 1612

Redshift に接続する 1613

トークン 1614

フェデレーションのグループ化 1615

Okta 1616

ドライバーを更新する 1616

トラブルシューティング 1617

ステップ 1: IDP を構成する 1620

ステップ 2: AWS で IDP とロールを構成する 1621

ステップ 3: Redshift に接続する 1621

トークン 1623

Okta 1623

ドライバーを更新する 1624

Redshift IAM IDC OAuth のトラブルシューティング 1624

ステップ 1: OAuth クライアントを Dremio に登録する 1627

ステップ 2: Dremio OAuth 用に Tableau Server を構成する 1627

サイトのカスタム OAuth の構成 1629

ステップ1: 新しいアプリを作成する 1631

ステップ 2: Dropbox 用に Tableau Server を構成する 1632

サイトのカスタム OAuth の構成 1632

クライアント ID を取得してGoogle API を有効にする 1636

Google OAuth のための Tableau Server の構成 1639

サイトのカスタム OAuth の構成 1639

Google データ ソースの作成と編集 1642

アクセス トークンの管理 1642

ステップ 1: Intuit アプリを作成する 1643

ステップ 2: Intuit QuickBooks Online 用に Tableau Server を構成する 1643

アクセス トークンの管理 1644

競合エラー 1644

サポートされているデータ ソース 1653

ライセンスの管理 1779

アイデンティティの移行について 1853

アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証 1885

Tableau Server へのユーザーの追加 1902

Tableau Server 管理者エリアへのサインイン 1912

Tableau サービス マネージャーの Web UI へのサインイン 1919

サーバーのカスタマイズ 1922

サーバー間でのサイトの管理 1926

抽出更新スケジュール 1956

Tableau Server でのバックグラウンド ジョブの管理 1989

Tableau サービス マネージャーのジョブ 1997

管理ビュー 2001

パフォーマンス 2051

Tableau Server のモニタリング 2108

メンテナンス 2124

tsm コマンド ライン リファレンス 2193

エンティティ定義とテンプレート 2460

tabcmd 2506

トラブルシューティング 2657

サーバー管理者参考資料 2724

アーカイブ済みコンテンツ 2890

Tableau Server 上の Tableau Advanced Management について 2892

実行可能なユーザー 3101

エージェントのライセンスなし 3101

互換性のないエージェント バージョン 3102

エージェント メッセージ キューの認証資格情報ローテーションの失敗 3106
エージェントダウン 3107
実行可能なユーザー 3108
実行可能なユーザー 3108
RMT サーバー Web インターフェイスの使用 3109
構成ファイル (config.json) を使用する 3111
実行可能なユーザー 3114
実行可能なユーザー 3114
RMT サーバー Web インターフェイスの使用 3114
構成ファイル (config.json) を使用する 3115
実行可能なユーザー 3116
低速ビュー インシデントのしきい値の構成 3117
環境タブ 3134
サーバー タブ 3137
インサイト タブ 3138
ステータス タブ 3138
実行可能なユーザー 3138
関連トピック 3138
VizQL セッション 3140
バックグラウンド タスク 3140
データ クエリ 3140
ビューの読み込み 3140

低速ビュー 3141
実行可能なユーザー 3141
関連トピック 3141
サイト 3143
プロジェクト 3143
ワークブック 3143
ビュー 3143
実行可能なユーザー 3143
関連トピック 3143
実行可能なユーザー 3147
データ管理 について 3465

セキュリティ

Tableau Server の管理の一環として、認証、データセキュリティ、ネットワークセキュリティを構成できます。

認証

認証では、ユーザーの ID を検証します。Tableau Server にアクセスする必要がある人は、その目的がサーバーの管理やコンテンツのパブリッシュ、参照、管理のいずれであっても、ユーザーとして Tableau Server リポジトリに追加される必要があります。認証方法は Tableau Server (「ローカル認証」) によって実行されるか、外部プロセスによって認証が実行される場合があります。後者では、Kerberos、SAML、OpenID などの外部認証技術を使用するように Tableau Server を構成する必要があります。いずれの場合でも、認証がローカルと外部のどちらで実行されるかに関係なく、Tableau Server リポジトリに各ユーザーの ID を示す必要があります。リポジトリではユーザー ID の認証メタデータが管理されます。

> Tableau Server on Windows については、「認証」を参照してください。

すべてのユーザー ID は最終的に Tableau Server リポジトリに表示され、保存されますが、Tableau Server のアカウントはアイデンティティストアで管理する必要があります。LDAP とローカルという、2 つの相互排他的なアイデンティティストア オプションがあります。Tableau Server では任意の LDAP ディレクトリがサポートされますが、Active Directory の LDAP 実装で最適化されます。または、LDAP ディレクトリを実行していない場合は、Tableau Server のローカル アイデンティティストアを使用できます。詳細については、アイデンティティストアを参照してください。

次の表に示されているように、実装するアイデンティティストアのタイプの一部によって、認証オプションが決まります。

アイデンティティストア	認証方法								
	基本	SAML	サイト SAML	Kerberos	(Windows のみ) Windows ログオン (Microsoft SSPI)	OpenID 接続	接続済みアプリ	信頼できる認証	相互 SSL
ローカル	X	X	X			X	X	X	X
Active Directory	X	X		X	X		X	X	X
LDAP	X	X					X	X	X

アクセスや管理のパーミッションは、サイト ロールを通じて実装されます。サイト ロールは、どのユーザーが管理者であり、どのユーザーがサーバー上のコンテンツ消費者やパブリッシャーであるかを定

義します。管理者、サイトロール、グループ、ゲストユーザー、ユーザー関連の管理タスクの詳細については、「ユーザー」および「ユーザーのサイトロール」を参照してください。

注: 認証において、ユーザーはサーバー上にアカウントを持っているという理由で Tableau Server 経由での外部データソースへのアクセスを許可されていないと理解することが重要です。つまり、既定の構成では、Tableau Server は外部データソースに対するプロキシとしての役割を果たしません。このようなアクセスには、Tableau Server でデータソースの追加構成を行うか、ユーザーが Tableau Desktop からの接続したときにデータソースで認証を行う必要があります。

アドオン認証の互換性

一部の認証方法は組み合わせて使用できます。次の表に、組み合わせ可能な認証方法を示します。"X" でマークされたセルは、組み合わせ可能な認証セットを示します。空白セルは、互換性のない認証セットを示します。

	接続済みアプリ	信頼できる認証	サーバー全体の SAML	サイト SAML	Kerberos	(Windows のみ) 自動ログオン (Microsoft SSPI)	相互 SSL	OpenID Connect
Tableau 連携アプリ	N/A		X	X	X		X	X
信頼できる認証		N/A	X	X	X		X	X
サーバー全体の SAML	X	X	N/A	X				

サイト SAML	X	X	X	N/A				
Kerberos	X	X			N/A			
自動ログオン (Microsoft SSPI)						N/A		
相互 SSL	X	X					N/A	
OpenID Connect	X	X						N/A
パーソナルアクセストークン (PAT)	*	*	*	*	*	*	*	*

* PAT は、設計上、これらの列にリストされている認証メカニズムと直接連携して REST API に対して認証を行うことはありません。代わりに、PAT は Tableau Server ユーザー アカウントの認証資格情報を使用して REST API に対する認証を行います。

クライアント認証の互換性

ユーザー インターフェイス (UI) を通じて処理される認証

クライアント	認証方法									
	基本	SAML	サイト SAML	Kerberos	(Windows のみ) Windows ログオン (Microsoft SSPI)	OpenID 接続	接続済みアプリ	信頼できる認証	相互 SSL	パーソナル アクセス トークン (PAT)
Tableau Desktop	X	X	X	X	X	X			X	
Tableau Prep Builder	X	X	X	X	X	X			X	
Tableau Mobile	X	X	X	X (iOS のみ *)	X **	X			X	
Web ブラウザー	X	X	X	X	X	X	X ***	X	X	

* Kerberos SSO は Android ではサポートされていませんが、ユーザー名とパスワードへのフォールバックは可能です。詳細については、注 5: Android プラットフォームを参照してください。

** SSPI は Tableau Mobile アプリの Workspace ONE バージョンとは互換性がありません。

*** 埋め込みのワークフローのみ。

認証はプログラムで処理されます

クライアント	認証方法									
	基本	SAML	サイト SAML	Kerbero-s	(Window-s のみ) Windows ログオン (Microso-ft SSPI)	OpenI-D 接続	接続済みアプリ	信頼できる認証	相互 SSL	パーソナルアクセストークン (PAT)
REST API	X						X			X
tabcm-d 2.0	X									X
tabcm-d	X									

ローカル認証

サーバーがローカル認証を使用するよう構成されている場合、Tableau Server でユーザーを認証します。ユーザーが Tableau Desktop、tabcmd、API、Web クライアントのいずれかでサインインして認証資格情報を入力すると、Tableau Server は認証資格情報を検証します。

このシナリオを有効にするには、最初に各ユーザーの ID を作成する必要があります。ID を作成するには、ユーザー名とパスワードを指定します。サーバー上のコンテンツにアクセス、またはコンテンツを操作するには、ユーザーにサイトロールが割り当てられている必要があります。tabcmd コマンドまたは REST API を使用して、Tableau Server のサーバー UI にユーザーアイデンティティを追加できます。

また、大きな関連ユーザーグループセット(例:「マーケティング」)の役割の管理と割り当てに役立つよう、Tableau Server 内でグループを作成することもできます。

Tableau Server をローカル認証で構成する場合は、パスワードによるログイン試行の失敗に対するパスワードポリシーおよびアカウントのロックアウトを設定できます。ローカル認証を参照してください。

注: 多要素認証を備えた Tableau (Tableau with MFA) は、Tableau Cloud のみで使用できます。

外部認証ソリューション

Tableau Server を構成して、多くの外部認証ソリューションを使用して動作できるようにします。

Kerberos

Kerberos を使用するように Active Directory で Tableau Server を構成することができます。Kerberosを参照してください。

SAML

Tableau Server を構成して SAML (Security Assertion Markup Language) 認証を使用できます。SAML を使用すると、外部 ID プロバイダー(IdP) はユーザーの認証資格情報を認証、セキュリティアサーションを Tableau Server に送信し、ユーザー ID に関する情報を提供します。

詳細については、「SAML」を参照してください。

OpenID Connect

OpenID Connect (OIDC) は、ユーザーが Google などの ID プロバイダー(IdP) にサインインできるようにする標準認証プロトコルです。ユーザーは IdP に正常にサインインした後、自動的に

Tableau Server へサインインされます。Tableau Server で OIDC を使用するには、ローカル アイデンティティストアを使用するようにサーバーを構成する必要があります。Active Directory または LDAP アイデンティティストアは OIDC ではサポートされていません。詳細については、OpenID Connectを参照してください。

相互 SSL

相互 SSL を使用すると、Tableau Desktop、Tableau Mobile、およびその他の承認済み Tableau クライアントのユーザーに Tableau Server への安全な直接アクセスを提供することができます。相互 SSL では、有効な SSL 証明書を持つクライアントが Tableau Server に接続するときに、Tableau Server でクライアント証明書の存在を確認し、クライアント証明書内のユーザー名に基づいてユーザーを認証します。クライアントに有効な SSL 証明書がない場合、Tableau Server は接続を拒否することができます。詳細については、相互 SSL 認証の構成を参照してください。

接続済みアプリ

直接信頼

Tableau 接続済みアプリを使用すると、Tableau Server サイトとTableau コンテンツが埋め込まれている外部アプリケーションとの間に明示的な信頼関係を構築できるため、シームレスで安全な認証エクスペリエンスを実現できます。接続済みアプリを使用すると、JSON Web トークン (JWT) を使用して Tableau REST API へのアクセスをプログラムで承認することもできます。詳細については、「Tableau 接続済みアプリを使用してアプリケーションを統合する」を参照してください。

EAS または OAuth 2.0 信頼

Tableau Server に外部認証サーバー (EAS) を登録すると、OAuth 2.0 標準プロトコルを使用して、Tableau Server とEAS の間に信頼関係を構築できます。この信頼関係により、IdP を介して、埋め込み Tableau コンテンツへのシングル サインオンのエクスペリエンスをユーザーに提供できます。EAS を登録すると、JSON Web トークン (JWT) を使用して Tableau REST API へのアクセスをプログラムで承認することもできます。詳細については、「OAuth 2.0 信頼を使用して接続済みアプリを設定する」を参照してください。

信頼できる認証

信頼できる認証 (「信頼できるチケット」とも呼ばれます) を使用すると、Tableau Server と1 つ以上の Web サーバーとの間で信頼関係を設定できます。Tableau Server が信頼できる Web サーバーから要求を受け取ると、Tableau Server は Web サーバーで必要とされる認証は既に処理されていると見なします。Tableau Server は、引き替え可能なトークンまたはチケットを含む要求を受け取り、ユーザー ロールおよびパーミッションを考慮したカスタマイズされたビューをユーザーに表示します。詳細については、「信頼できる認証」を参照してください。

LDAP

ユーザー認証に LDAP を使用するように Tableau Server を構成することもできます。ユーザーは認証資格情報を Tableau Server に送信して承認されてから、ユーザーの認証資格情報を使用して LDAP インスタンスへのバインドを試行します。バインドが機能したら、認証資格情報が有効になり、Tableau Server よりユーザーにセッションが付与されます。

「バインディング」は、クライアントが LDAP サーバーへのアクセスを試行するときに発生するハンドシェイク/認証手順です。Tableau Server では、さまざまな非認証関連クエリ(ユーザーおよびグループのインポートなど) を作成する場合に実行します。

ユーザーの認証資格情報を確認するときに、Tableau Server で使用するバインドのタイプを設定できます。Tableau Server では、GSSAPI およびシンプル バインドがサポートされています。シンプル バインドでは、認証資格情報を LDAP インスタンスに直接渡します。SSL を構成してバインド通信を暗号化することをお勧めします。このシナリオの認証は、ネイティブ LDAP ソリューションまたは SAML などの外部プロセスで受ける可能性があります。

LDAP の計画および構成の詳細については、アイデンティティストアおよび外部のアイデンティティストアの構成 リファレンスを参照してください。

SCIM

Tableau Server の SCIM (クロスドメイン アイデンティティ管理) は、外部のクラウドベースの ID プロバイダー (IdP) からユーザーやグループをプロビジョニングしたり管理したりすることを自動化するための標準です。SCIM は認証方法ではありませんが、ユーザーをプロビジョニングします。その後、

ユーザーは Tableau Server 用に構成された方法を使用して認証を行います。この機能は、サイト固有の SAML 認証と連携します。詳細については、「SCIM」を参照してください。

その他の認証シナリオ

- REST API: サインインとサインアウト(認証)

  注: REST API は、SAML のシングル サインオン(SSO) をサポートしていません。

- モバイル デバイス認証: Tableau Mobile へのシングル サインオン
- TSM クライアントの証明書の信頼: TSM クライアントの接続
- TSM 管理のための PAM 統合: TSM 認証

データアクセスとソース認証

多種のデータソースに対応する多数の異なる認証プロトコルをサポートするように、Tableau Server を設定できます。データ接続認証は、Tableau Server 認証とは独立している場合があります。

たとえば、Tableau Server に対するユーザー認証はローカル認証を使用して設定し、特定のデータソースに対しては OAuth、または SAML 認証を設定することができます。データ接続認証を参照してください。

ローカル認証

サーバーがローカル アイデンティティストアを使用するよう構成されている場合、Tableau Server でユーザーを認証します。ユーザーが Tableau Desktop、tabcmd、API、Web クライアントのいずれかでサインインして認証資格情報を入力すると、Tableau Server は認証資格情報を検証します。アイデンティティストアに保存されている Tableau ユーザー名は Tableau Server の権限やパーミッションと関連付けられています。認証が検証された後、Tableau Server は Tableau リソースへのユーザー アクセス(認可) を管理します。

ローカル認証を使用するには、セットアップ時にローカル ID で Tableau Server を構成する必要があります。Tableau Server が外部の ID ストア(LDAP、Active Directory など)で構成されている場合、ローカル認証を使用することはできません。

注: アイデンティティプールは、組織で必要となる追加のユーザープロビジョニングと認証オプションを補完およびサポートするように設計されたツールで、OpenID Connect (OIDC) 認証のみをサポートします。詳細については、「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。

パスワードの保存

ローカル認証を使用すると、ユーザーのソルト付きおよびハッシュ化されたパスワードはリポジトリに格納されます。パスワードは直接には格納されず、ソルトされハッシュ化されたパスワードの結果が保存されます。サーバーは、HMAC SHA512 でハッシュ化された PBKDF2 の派生関数を使用します。

パスワード設定の構成

ローカル認証を使用して Tableau Server をインストールした後、Tableau Server Manager(TSM) を使用して、次のようにして、複数のパスワード関連設定を構成できます。

- パスワードポリシー: これらのポリシーは、長さ、文字タイプ、その他の要件などのパスワード構造の要件を定義します。
- パスワードの有効期限: 有効にしてパスワードの有効期限を指定します。
- ログインレート制限: ユーザーが誤ったパスワードを5回入力した後、サインイン試行の間に時間がかかります。ユーザーは、もう一度サインインを試行するまでに数秒待機する必要があります。ユーザーが誤ったパスワードを継続して入力した場合は、サインイン試行間飛躍的にの時間が長くなります。既定では、サインイン試行の最大時間は60分です。

  アカウントアクセスの失敗が多すぎるとアカウントはロックされます。ロックアウトされるまでにユーザーが試行できるパスワードの入力回数を指定できます。ロックされたアカウントへのアクセスのロックを解除する方法については、サイトのユーザーの表示と管理を参照してください。

- ユーザーのパスワードリセット: ユーザーがパスワードをリセットできるようにします。パスワードのリセットを有効にすると、サインインページにリンクが表示されるように Tableau Server を構成します。パスワードを忘れたユーザーやパスワードをリセットするユーザーは、リンクをク

リックしてパスワードのリセットワークフローを開始することができます。パスワードのリセットは、以下で説明するように、TSM CLI を使用して構成する必要があります。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[ユーザー アイデンティティとアクセス]** をクリックし、次に **[認証方法]** をクリックします。

3. ドロップダウンメニューから[ローカル認証]を選択して、パスワード設定を表示します。

4. パスワード設定を指定し、**「保留中の変更を保存」**をクリックします。

5. ページ上部の **[変更を保留中]** をクリックします。

6. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

パスワードポリシーの初期の構成では、次の構成ファイル テンプレートを使用して json ファイルを作成することをお勧めします。また、tsm configuration set で説明されている構文と共に以下の単一構成キーを設定することもできます。

1. 以下の json テンプレートをファイルにコピーします。ユーザーのパスワードポリシー構成でキー値を埋めます。キー オプションの詳細については、後続の参照セクションを参照してください。

   ```
   {
   "configKeys": {
   ```

```
 "wgserver.localauth.policies.mustcontainletters.enabled":
false,

  "wgserver.localauth.policies.mustcontainuppercase.enabled":
false,

  "wgserver.localauth.policies.mustcontainnumbers.enabled":
false,

  "wgserver.localauth.policies.mustcontainsymbols.enabled":
false,

  "wgserver.localauth.policies.minimumpasswordlength.enabled":
false,

  "wgserver.localauth.policies.minimumpasswordlength.value": 8,

  "wgserver.localauth.policies.maximumpasswordlength.enabled":
false,

  "wgserver.localauth.policies.maximumpasswordlength.value":
255,

  "wgserver.localauth.passwordexpiration.enabled": false,

  "wgserver.localauth.passwordexpiration.days": 90,

  "wgserver.localauth.ratelimiting.maxbackoff.minutes": 60,

  "wgserver.localauth.ratelimiting.maxattempts.enabled": false,

  "wgserver.localauth.ratelimiting.maxattempts.value": 5,

  "vizportal.password_reset": false

                }

}
```

2. `tsm settings import -f file.json` を実行し、適切な値を指定した json ファイルを Tableau サービス マネージャーに渡して Tableau Server を構成します。

3. `tsm pending-changes apply` コマンドを実行して変更を適用します。tsm pending-changes apply を参照してください。

このセクションは、パスワードポリシーの構成に使用できるすべてのオプションをリストしたものです。

`wgserver.localauth.policies.mustcontainletters.enabled`

既定値: `false`

パスワードには少なくとも文字が1つ必要です。

`wgserver.localauth.policies.mustcontainuppercase.enabled`

既定値: `false`

パスワードには少なくとも1つの大文字が必要です。

`wgserver.localauth.policies.mustcontainnumbers.enabled`

既定値: `false`

少なくとも1つの数字が必要です。

`wgserver.localauth.policies.mustcontainsymbols.enabled`

既定値: `false`

パスワードには少なくとも1つの特殊文字が必要です。

`wgserver.localauth.policies.minimumpasswordlength.enabled`

既定値: `false`

最小長さのパスワードを設定します。

`wgserver.localauth.policies.minimumpasswordlength.value`

既定値: `8`

最小文字数のパスワードには。4 と255とを含むその間の数字を入力します。この値を適用するには、wgserver.localauth.policies.minimumpasswordlength.enabled をtrue に設定する必要があります。

wgserver.localauth.policies.maximumpasswordlength.enabled

既定値: false

最大長さのパスワードを設定します。

wgserver.localauth.policies.maximumpasswordlength.value

既定値: 255

最大文字数のパスワードには。8 と225とを含むその間の数字を入力します。この値を適用するには、wgserver.localauth.policies.maximumpasswordlength.enabled をtrue に設定する必要があります。

wgserver.localauth.passwordexpiration.enabled

既定値: false

パスワードの有効期限を実施します。

wgserver.localauth.passwordexpiration.days

既定値: 90

パスワードが期限切れになるまでの日数。1 と365とを含むその間の数字を入力します。この値を適用するには、wgserver.localauth.passwordexpiration.enabled を true に設定する必要があります。

wgserver.localauth.ratelimiting.maxbackoff.minutes

既定値: 60

ユーザーが複数回間違ったパスワードを入力した後の、サインイン試行の最大時間。5 と 1440 とを含むその間の数字を入力します。

`wgserver.localauth.ratelimiting.maxattempts.enabled`

既定値: `false`

パスワードが 5 回誤って入力された後に、アカウントロックを施行する。アカウントロックをトリガする誤ったパスワードの数を変更するには、`wgserver.localauth.ratelimiting.maxattempts.value` を設定します。

`wgserver.localauth.ratelimiting.maxattempts.value`

既定値: `5`

アカウントロックをトリガするためにユーザーが入力する誤ったパスワードの回数。5 と 100 とを含むその間の数字を入力します。この値を適用するには、`wgserver.localauth.ratelimiting.maxattempts.enabled` を `true` に設定する必要があります。

vizportal.password_reset

既定値: `false`

ユーザーがパスワードをリセットできるようにします。この機能を動作させるには、電子メールを送信するように Tableau Server を構成する必要があります。SMTP セットアップの構成を参照してください。

SAML

SAML (セキュリティアサーション マークアップ ランゲージ) は、セキュアな Web ドメインがユーザー認証および認可データを交換できる XML 規格です。Tableau Server を構成し、外部アイデンティティプロバイダー (IdP) を使用して SAML 2.0 を介したユーザーの認証を実行できます。ユーザー認証資格情報が Tableau と共に保存されることはなく、SAML を使用すると、Tableau を組織のシングル サインオン環境に追加できます。

SAML はサーバー全体で使用するか、個々のサイトで構成できます。これらのオプションの概要を以下に示します。

- **サーバー全体の SAML 認証**。単一の SAML IdP アプリケーションにより、すべての Tableau Server ユーザーの認証が処理されます。この場合、お使いのサーバーにデフォルトサイトしかない場合、サイト固有の SAML を構成する必要がないため、このオプションを使用します。マルチサイト環境ではサーバー全体の SAML を使用することもできますが、ユーザーはすべてのサイトに対して単一の IdP に制限されます。

- **サーバー全体のローカル認証とサイト固有の SAML 認証**。マルチサイト環境において、サイトレベルで SAML 認証に対応していないユーザーがローカル認証を使用してサインインできます。

- **サーバー全体の SAML 認証とサイト固有の SAML 認証**。マルチサイト環境において、サイトレベルで構成された SAML IdP を使用してすべてのユーザーが認証します。複数のサイトに所属するユーザー用に、サーバー全体の既定 SAML IdP を指定します。

サイト固有の SAML を使用するには、個々のサイトを構成する前にサーバー全体の SAML を構成する必要があります。サイト固有の SAML が機能するためにサーバー側 SAML を有効にする必要はありませんが、構成する必要があります。

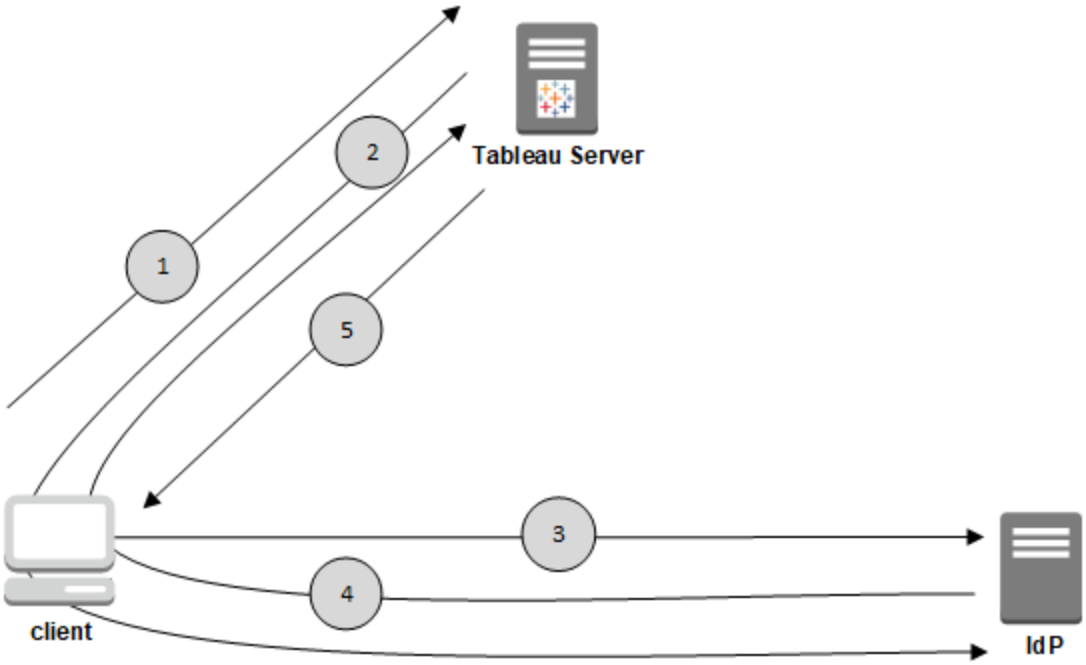
SAML によるユーザー認証は、データソースやワークブックなどの Tableau Server コンテンツに関するパーミッションや認可には適用されません。また、ワークブックやデータソースが接続する参照元データへのアクセスがこれによって制御されることはありません。

注:

- Tableau Server ではサービスプロバイダーによって開始された SAML および IdP によって開始された SAML の両方が、ブラウザーと Tableau Mobile アプリでサポートされています。Tableau Desktop からの SAML 接続は、サービスプロバイダーが開始する必要があります。
- アイデンティティプールは、組織で必要となる追加のユーザープロビジョニングと認証オプションを補完およびサポートするように設計されたツールで、OpenID Connect (OIDC) 認証のみをサポートします。詳細については、「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。

認証の概要

次のイメージは、一般的なサービスプロバイダーによる開始フローで、ユーザーがシングル サインオンを使用して認証するための手順を示します。



1. ユーザーは Tableau Server サインイン ページに移動するかパブリッシュされたワークブックの URL をクリックします。
2. 構成された IdP にクライアントをリダイレクトして、Tableau で認証プロセスを開始します。
3. IdP はユーザーのユーザー名とパスワードを要求します。ユーザーが有効な認証資格情報を送信すると、IdP はユーザーを認証します。
4. IdP は、SAML 応答の形式で成功した認証をクライアントに返します。クライアントは SAML 応答を Tableau Server. に渡します。
5. Tableau Server は、SAML 応答のユーザー名が、Tableau Server リポジトリに保存されているライセンス ユーザーと一致するかどうかを検証します。一致が確認されると、Tableau Server が要求されたコンテンツでクライアントに応答します。

SAML 要件

Tableau Server で SAML を構成する前に、使用している環境が要件を満たしていることを確認してください。

重要: SAML 構成は、IdP とTableau Server の両方で、大文字と小文字が区別されます。たとえば、IdP とTableau Server で構成された URL は完全に一致する必要があります。

証明書およびアイデンティティプロバイダー (IdP) の要件

SAML 向けに Tableau Server を構成するには、次が必要です。

- **証明書ファイル。.crt** 拡張子が付いた PEM 暗号化 x509 証明書ファイル。このファイルは IdP ではなくTableau Server によって使用されます。SSL 証明書を持っている場合、ある状況では同じ証明書を SAML で使用できます。詳細については、この記事の後の部分の SAML での SSL 証明書およびキーファイルの使用を参照してください。

  Tableau Server では、IdP に送信される要求に署名するために証明書キーペアが必要です。これにより、署名された要求をスプーフィングすることが難しくなるため、中間者攻撃の脅威が軽減されます。さらに、Tableau Server は、信頼できる IdP から受信した AuthNResponse を検証します。Tableau Server は、IdP によって生成された署名を使用して AuthNResponse を検証します。IdP 証明書のメタデータは、最初の SAML 構成プロセスの一部として Tableau Server に提供されます。

  署名付き要求は、すべての IdP に必ずしも必要ではありません。既定では、Tableau Server には署名付きの要求が必要です。IdP を使用した通信伝送の安全性をさらに高めるために、この構成をお勧めします。署名付き要求を無効にする必要があるかどうかを理解するために、IdP チームと共同作業してください。署名付き要求を無効にするには、samlSettings エンティティを参照してください。

- **署名アルゴリズム**証明書には、SHA-256 などの安全な署名アルゴリズムを使用する必要があります。SHA-1 署名ハッシュを使用する証明書を使用して Tableau Server に SAML を構成しようとすると、Tableau Server は証明書を拒否します。tsm wgserver.saml.blocklisted_digest_algorithms 構成キーを設定することにより、安全性の低い SHA-1 ハッシュを受け入れるように Tableau Server を構成できます。

- **RSA キーと ECDSA 曲線のサイズ**Tableau Server 証明書の RSA キー強度は 2048 である必要があり、IdP 証明書は RSA キー強度が 2048 であるか ECDSA 曲線サイズが 256 である必要があります。

  それぞれの構成キー wgserver.saml.min_allowed.rsa_key_size および wgserver.saml.min_allowed.elliptic_curve_size を設定することにより、安全性の低いサイズを受け入れるように Tableau Server を構成できます。

- **証明書キー ファイル。**拡張子 **.key** が付いた、RSA または DSA 秘密キー ファイル。RSA キーは PKCS#1 または PKCS#8 形式にする必要があります。

  パスワードの保護要件は次のとおりです。

  - PKCS#1 RSA キー ファイルはパスワードで保護することができません。
  - パスワードで保護されているキー ファイルを使用するには、RSA PKCS#8 ファイルで SAML を構成する必要があります。**注:** NULL のパスワードを含む PKCS#8 ファイルはサポートされていません。
  - サイト固有の SAML 展開では、パスワードで保護されたキー ファイルはサポートされていません。

  サポートの概要

キー ファイル形式	サーバー全体の SAML サポート	サイト レベルの SAML サポート
PKCS＃8 RSA	あり	なし
PKCS＃8 (なし/null のパスワード)	なし	なし
PKCS＃1 RSA	あり	あり
PKCS＃1 RSA (パスワード)	なし	なし
PKCS＃1 DSA (パスワー	なし	なし

キー ファイル形式	サーバー全体の SAML サポート	サイト レベルの SAML サポート
ド)		

- **IdP は、安全な署名アルゴリズムを使用して SAML アサーションに署名する必要があります。**既定では、Tableau Server は SHA-1 アルゴリズムを使用して署名された SAML アサーションを拒否します。tsm wgserver.saml.blocklisted_digest_algorithms 構成キーを設定することにより、安全性の低い SHA-1 ハッシュを使用して署名されたアサーションを受け入れるように Tableau Server を構成できます。
- **SAML 2.0 以降をサポートする IdP アカウント。**外部アイデンティティプロバイダーでのアカウントが必要です。いくつかの例として PingFederate、SiteMinder、および Open AM があります。
- **XML メタデータのインポート/エクスポートをサポートする IdP プロバイダー。**手動で作成したメタデータ ファイルでも動作する可能性はありますが、Tableau テクニカル サポートでは、手動でのファイル生成やそれらのトラブルシューティングをサポートできません。
- **ユーザー名:** 必須。IdP 構成には "username" 属性またはクレームが含まれている必要があり、Tableau Server 上の対応する SAML 構成の属性も "username" に設定されている必要があります。

SSL オフロード

組織が認証要求を Tableau Server に送信する前にプロキシ サーバーで IdP よりSSL 接続を終了する場合は、プロキシ構成を行う必要がある場合があります。このシナリオでは、SSL はプロキシ サーバーで "オフロード" されています。つまりhttps 要求がプロキシ サーバーで終了し、http 経由で Tableau Server に転送されます。

Tableau Server は、IdP から返される SAML 応答メッセージを検証します。SSL はプロキシでオフロードされるため、Tableau Server は受信したプロトコルで検証されます (http)。しかし、IdP 応答は https により書式設定されるため、プロキシ サーバーに `https` に設定された X-Forwarded-Proto ヘッダーが含まれていない限り検証は失敗します。リバースプロキシ サーバーやロードバランサーと連携させるための Tableau Server の設定を参照してください。

SAML での SSL 証明書およびキーファイルの使用

SSL で PEM エンコードされた x509 証明書ファイルを使用している場合、同じファイルを SAML で使用できます。SSL では証明書ファイルはトラフィックを暗号化するために使われます。SAML では証明書は認証のために使われます。

上記の証明書およびアイデンティティプロバイダー (IdP) の要件に記載されている要件に加え、SSL と SAML の両方で同じ証明書を使用するために、その証明書が SAML で機能するように以下の条件を満たす必要もあります。

- 証明書には Tableau Server に適用する証明書のみが含まれ、他の証明書やキーが含まれていないことを確認します。

  これを実行するには、証明書ファイルのバックアップコピーを作成してから、テキストエディターでコピーを開いてそのコンテンツを確認します。

ユーザー管理要件

SAML を有効化すると、ユーザー認証が Tableau の外部で IdP を使用して実行されます。ただし、ユーザー管理はアイデンティティストア (外部アイデンティティストア (Active Directory または LDAP) または Tableau Server のローカル アイデンティティストア) によって実行されます。Tableau Server でのユーザー管理の計画の詳細については、アイデンティティストアを参照してください。

セットアップ時にアイデンティティストアを構成する場合は、SAML の使用方法を反映するオプションを選択する必要があります。サイト固有の SAML を使用するには、個々のサイトを構成する前にサーバー全体の SAML を構成する必要があることに注意してください。

- **サイト固有の SAML の場合:** Tableau Server 上に複数のサイトがあり、特定の IdP または IdP アプリケーション用に各サイトをセットアップする場合 (または、SAML を使用しないように一部のサイトを構成する場合) は、ローカル アイデンティティストアを使用してユーザーを管理するように Tableau Server を構成します。サイト固有の SAML に関しては、Tableau Server では認証を IdP に依存しているため、パスワードは使用しません。

- **サーバー全体の SAML の場合:** 単一の IdP を使用するサーバー全体の SAML を構成する場合は、ローカル アイデンティティストアまたは外部アイデンティティストアを使用するように

Tableau Server を構成できます。

- **サーバー全体の SAML 認証およびサイト固有の SAML 認証の場合:**

  - **ローカル アイデンティティ ストアを使用する場合は**、メール アドレス形式のユーザ名を使用することが重要です。完全なメール アドレスを使用すると、メール プレフィックスが同じであるが、異なるメール ドメインを使用しているユーザーが 2 人いる場合でも、Tableau Server におけるユーザー名の一意性を保証するのに役立ちます。アイデンティティの一意性を保証するには、両方のシステムで完全なメール アドレス形式を使用するか、Tableau Server をバージョン 2022.1.x 以降にアップグレードして、アイデンティティ移行バックグラウンド ジョブを実行します。

  - **マルチサイト環境では**、すべてのユーザーがサイト レベルで構成された SAML IdP を使用して認証を行います。このシナリオでは、複数のサイトに属するユーザー用に、サーバー全体の既定の SAML IdP を指定します。このシナリオを構成するには、Tableau Server をローカル アイデンティティ ストアで構成する必要があります。

  - **SAML ユーザー名属性を照合するときにドメインを無視します**。Tableau Server バージョン 2021.4.21、2022.1.17、2022.3.9、2023.1.5 以降では、ID プロバイダー (IdP) のユーザー名と Tableau Server のユーザー アカウントを照合するときに、ユーザー名属性のドメイン部分を無視するように Tableau Server を構成できます。たとえば、IdP のユーザー名属性は `alice@example.com` となる可能性があり、これは、Tableau Server の `alice` という名前のユーザーと一致します。ユーザー名属性のプレフィックス部分に一致するが、ユーザー名属性のドメイン部分には一致しないユーザーを Tableau Server ですでに定義していれば、ユーザー名属性のドメイン部分を無視すると役立つ場合があります。

    重要: ドメイン名を無視する際は、細心の注意を払ってください。特に、IdP で作成した構成済みドメイン全体で、ユーザー名が一意であることを確認してください。ドメイン名を無視するように Tableau Server を構成すると、意図しないユーザー サインインが発生する可能性があります。複数のドメイン (例: `example.com` と `tableau.com`) に対して IdP が構成されている場合について考えてみてください。

名が同じであるが、ユーザー アカウントが異なる (例: `alice@tableau.com` と `alice@example.com`) 2 人のユーザーが組織にいる場合、マッピングの不一致が発生する可能性があります。

IdP からのユーザー名属性のドメイン名を無視するように Tableau Server を構成するには、`wgserver.ignore_domain_in_username_for_matching` を `true` に設定します。詳細については、「wgserver.ignore_domain_in_username_for_matching」を参照してください。

注:

- このコマンドは、`legacy-identity-mode` になっている Tableau Server 展開か、アイデンティティの移行を通じて更新されていない (アイデンティティサービスを使用しない) Tableau Server 展開のみで機能します。
- ユーザー名属性のドメイン名を無視するように tsm コマンドを変更する場合は、Tableau Server 内のすべてのユーザー名にドメイン名が必要です。

注: REST API および tabcmd では、SAML シングル サインオン (SSO) はサポートされていません。サインインするには、サーバーで作成されたユーザーの名前とパスワードを指定する必要があります。ユーザーは、Tableau Server の構成方法に応じてローカル アイデンティティストアまたは外部アイデンティティストアによって管理できます。REST API または tabcmd 呼び出しで、サインインしているユーザーのパーミッションを取得します。

SAML 互換性についての注意事項と要件

- **ユーザー名の一致:** Tableau Server に保存されているユーザー名が、SAML アサーションの IdP によって送信される構成済みのユーザー名属性と一致する必要があります。既定では、Tableau Server は、そのユーザーの情報を含む "username" という属性が着信アサーションに含まれていることを想定しています。たとえば、Jane Smith のユーザー名が **jsmith** として PingFederate に保存されている場合、Tableau Server でも **jsmith** として保存されている必要があります。

  認証中に SAML を構成する場合

最初の Tableau Server のセットアップ時に SAML を構成する場合、使用する予定のアカウントが IdP に存在することを確認してからセットアップを実行してください。Tableau Server のセットアップ時に、サーバー管理者のアカウントを作成します。

複数のドメインを実行している場合

Active Directory または LDAP による外部アイデンティティストアを使用しており、複数のドメインが存在する (ユーザーが複数のドメインに所属しているか、Tableau Server のインストールに複数のドメインが含まれている) 場合は、SAML アサーションのユーザーに対して、IdP でユーザー名属性とドメイン属性の*両方*を送信する必要があります。これらのユーザー名属性とドメイン属性はどちらも、Tableau Server に保存されているユーザー名とドメインと完全に一致する必要があります。次のいずれか ***1*** つを実行します。

- ユーザー名フィールドに `domain\username` を設定する
- 定義域フィールドに定義域を設定し、ユーザー名フィールドにユーザー名を設定する

定義域属性を設定するときは、完全修飾ドメイン名 (FQDN) または短縮名を使用できます。

定義域が指定されていない場合は、既定の定義域と見なされます。

詳細については、複数ドメインのサポートと、サーバー全体の SAML の構成の**「TSM CLI の使用」**タブにある「アサーションの一致」セクションを参照してください。

- **署名アルゴリズム**: Tableau Server は、SHA256 署名アルゴリズムを使用します。
- **シングル ログ アウト (SLO)**: Tableau Server では、サーバー全体の SAML とサイト固有の SAML の両方で、サービスプロバイダー (SP) によって開始された SLO とアイデンティティプロバイダー (IdP) によって開始された SLO の両方がサポートされています。
- **外部認証タイプ**: Tableau Server は外部認証タイプの同時利用をサポートしています。
- **相互 SSL** : Tableau Server は相互 SSL (双方向 SSL) と SAML の併用をサポートしていません。相互 SSL を使用する場合は IdP で構成できます。
- **アサーション エンコーディング**: アサーションは UTF-8 でエンコードされている必要があります。

- **暗号化とSAML アサーション**:
  - **サーバー全体の SAML**: Tableau Server がサーバー全体の SAML 用に構成されている場合、Tableau Server は IdP からの暗号化されたアサーションをサポートします。暗号化アサーションは、サーバー全体の SAML の初期構成の一部としてアップロードする証明書によって有効になります。SAML の要求と応答は、HTTP または HTTPS 経由で送信できます。
  - **サイト固有の SAML**: Tableau Server がサイト固有の SAML 用に構成されている場合、Tableau Server は IdP からの暗号化されたアサーションをサポートしません。ただし、IdP との通信のセキュリティを確保するために、すべての SAML の要求と応答は HTTPS 経由で送信されます。HTTP の要求と応答はサポートされていません。
- **tabcmd ユーザー用の Tableau Server 内のユーザー ID:** 上記のユーザー管理要件のセクションで説明されているとおり、tabcmd を使用するには、サーバーで定義されているユーザーとしてサインインする必要があります。**tabcmd** では SAML アカウントを使用できません。
- **Tableau Desktop での SAML SSO の使用**: 既定では、Tableau Desktop では SP で開始された SAML 認証ができます。

  IdP でこの機能がサポートされていない場合、以下のコマンドを使用して Tableau Desktop 向けの SAML サインインを無効にできます。

  ```
  tsm authentication saml configure --desktop-access disable
  ```

  詳細については、tsm authentication saml <commands> を参照してください。
- **分散インストール**: TSM バージョンの Tableau Server (2018.2 以降) では、クライアント ファイル サービスによってマルチノード クラスタ内のファイルが共有されます。クラスタの最初のノードで SAML を構成した後、証明書とキー ファイルはクライアント ファイル サービスによって他のノードに配布されます。
- **ログイン URL**:ユーザーがサインインできるように、次の URL に POST 要求を送信する SAML ログイン エンドポイントを IdP に構成する必要があります。

`https://<tableauserver>/wg/saml/SSO/index.html`。

- **ログアウト URL:** ユーザーが SAML を使用してサインインした後にサインアウト (シングル ログアウト、SLO) できるように、POST 要求を次の URL に送信する SAML ログアウト エンドポイントを使用して IdP を構成する必要があります。

  `https://<tableauserver>/wg/saml/SingleLogout/index.html`。

  注: Tableau Server では、サーバー全体の SAML とサイト固有の SAML の両方で、サービス プロバイダー (SP) によって開始された SLO とアイデンティティ プロバイダー (IdP) によって開始された SLO の両方がサポートされています。

- **ログアウト後のリダイレクト先 URL:** 既定では、ユーザーが Tableau Server からサインアウトすると、サインイン ページが表示されます。

  サインアウト後に別のページを表示するには、`tsm authentication saml configure` コマンドで `-su` または `--signout-url` オプションを使用します。

  - 絶対 URL を指定するには、次の例のように `http://` または `https://` で始まる完全修飾 URL を使用します。

    ```
    tsm authentication saml configure -su https://example.com
    ```

  - Tableau Server ホストの相対 URL を指定するには、/ (スラッシュ) で始まるページを使用します。

    ```
    tsm authentication saml configure -su /ourlogoutpage.html
    ```

- **Active Directory Federation Service (AD FS)**: SAML を使用して Tableau 認証用の追加属性を返すよう AD FS を構成する必要があります。**Name ID** および **username** 属性は、いずれも次の同じ AD 属性にマッピングできます。**SAM-Account-Name。**

- **AuthNContextClassRef**: AuthNContextClassRef はオプションの SAML 属性であり、IdP によって開始されるフローで特定の認証の "コンテキスト" を検証します。この属性のコンマ区切り値は TSM を使用して設定できます。この属性が設定されると、Tableau Server は、リストされている値の少なくとも 1 つが SAML 応答に含まれていることを検証します。

SAML 応答に構成済みの値のいずれかが含まれていない場合、ユーザーが IdP によって正常に認証された場合でも、認証は拒否されます。

このオプションの属性を空白のままにすると、既定の動作になります。SAML 応答が認証に成功すると、ユーザーに Tableau Server 内のセッションが付与されます。

この値はサーバー全体の SAML でのみ評価されます。サイト SAML が構成されている場合、`AuthNContextClassRef` 属性は無視されます。

TSM Web インターフェイスでこの値を設定するには、サーバー全体の SAML の構成を参照してください。

この値を tsm configuration set を使用して設定するには、`wgserver.saml.authcontexts` キーを使用して、値のコンマ区切りリストを設定します。

JSON 構成ファイルを使用してこの値を設定するには、samlSettings エンティティを参照してください。

Tableau クライアントアプリケーションでの SAML SSO の使用

Tableau Server ユーザーが SAML 認証資格情報を持っている場合、Tableau Desktop または Tableau Mobile アプリからサーバーにサインインできます。完全な互換性を実現するため、サーバーと同一バージョンの Tableau クライアントアプリケーションを使用することをお勧めします。サイト固有の SAML を使用して接続するには、ユーザーがバージョン 10.0 以降の Tableau クライアントアプリケーションを実行している必要があります。

Tableau Desktop または Tableau Mobile から Tableau Server への接続には、サービスプロバイダー (SP) によって開始された接続が使用されます。

認証されたユーザーを Tableau クライアントに再びリダイレクトする

ユーザーが Tableau Server にサインインすると (SP 開始ワークフローを使用)、Tableau Server は SAML 要求 (`AuthnRequest`) を IdP に送信します。これには Tableau アプリケーションの **RelayState** 値が含まれます。ユーザーが Tableau Desktop や Tableau Mobile などの Tableau ク

ライアントからTableau Server にサインインしている場合は、ユーザーを特定の Tableau リソースに誘導するために、IdP の SAML 応答内で RelayState 値を Tableau に返すことが重要です。

ユーザーが IdP から直接 Tableau Server にサインインするプロセスを開始すると(IdP 開始ワークフローを使用)、IdP は SAML 応答に **RelayState** 値を含めて、ユーザーを特定のTableau リソースに誘導できます。たとえば、RelayState の値を `/#/site/[site-url]/views/[view-name]/[workbook-tab-name]` に設定できます。

これらのシナリオで RelayState 値が適切に返されない場合、サインイン元のアプリケーションにリダイレクトするのではなく、ユーザーは Web ブラウザーで Tableau Server のホームページに移動します。

ID プロバイダーおよび社内 IT チームと協力し、この値が IdP の SAML 要求の一部として含まれること、またその後に IdP とTableau Server の間に存在する任意のネットワーク アプライアンス (プロキシやロード バランサーなど) により保持されることを確認してください。

XML データの要件

SAML 構成の一環として、Tableau Server とIdP 間で XML メタデータを交換します。この XML メタデータは、ユーザーが Tableau Server のサインインプロセスを開始する場合に、ユーザーの認証情報を検証するのに使用します。

Tableau Server とIdP はそれぞれ独自のメタデータを生成します。各セットのメタデータには、以下のリストに記載される情報を含める必要があります。いずれかのセットで情報が不足している場合、SAML の構成時またはサインインの試行時にエラーが発生する可能性があります。

- **HTTP POST**: サーバー全体の SAML の場合、Tableau Server は、SAML 通信に対して HTTP POST 要求のみをサポートします。IdP によってエクスポートされた SAML メタデータ XML ドキュメントでは、`Binding` 属性が HTTP-POST に設定されている必要があります。
- `Binding` 属性を `HTTP-POST` に設定する場合、Tableau Server および IdP がそれぞれエクスポートする SAML メタデータには以下の要素を含める必要があります。
    - 正常に認証された後、IdP がリダイレクトする宛先の URL を指定する要素。これは、アイデンティティプロバイダー メタデータでなくサービス プロバイダー メタデータで必

要です。

```
<md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://<tableau-server>/wg/saml/SSO/index.html
index="0" isDefault="true"/>
```

サイトの SAML では、Location エンドポイントは /samlservice/public/sp/metadata?alias=<site alias> です。

- ログアウトのエンドポイント要素は Tableau Server メタデータに含まれ、IdP が Tableau Server のログアウトエンドポイントに使用する URL を指定します。この要素が IdP メタデータに含まれていない場合、Tableau Server は IdP を使用してログアウトエンドポイントをネゴシエートできず、SAML ログアウト機能は Tableau Server で使用できません。

```
<md:SingleLogoutService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://SERVER-NAME:9031/idp/slo"
```

- IdP からのメタデータ XML に **SingleSignOnService** 要素が含まれていることを確認します。これは、次の例のようにバインドを HTTP-POST に設定する要素です。

```
<md:SingleSignOnService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://SERVER-NAME:9031/idp/SSO.saml2"/>
```

- この要素は IdP メタデータに含まれる必要があり、Tableau Server が IdP のログアウトエンドポイントに使用する URL を指定します。

```
 <md:SingleLogoutService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://SERVER-NAME:9031/idp/slo"/>
```

- ***username* という属性**: username 属性を含むアサーションを saml:AttributeStatement 要素に返すように IdP を構成する必要があります。アサーションの属性タイプは xs:string にする必要があります (xs:any と入力しないでください)。

  次の例のようになります。

```
<saml:Assertion assertion-element-attributes>
  <saml:Issuer>issuer-information</saml:Issuer>
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    ...
  </Signature>
  <saml:Subject>
    ...
  </saml:Subject>
  <saml:Conditions condition-attributes >
    ...
  </saml:Conditions>
  <saml:AuthnStatement authn-statement-attributes >
    ...
  </saml:AuthnStatement>

  <saml:AttributeStatement>
    <saml:Attribute Name="username"
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">
          user-name
    </saml:AttributeValue>
```

```
    </saml:Attribute>
  </saml:AttributeStatement>
</saml:Assertion>
```

既定で Tableau Server は Idp から返される AuthNResponse の `username` 属性を読み取ります。ただし、一部の IdP は、ユーザーを識別することを目的とした別の属性を返す場合があります。

`username` 値を渡す SAML 属性を変更するには、次の TSM コマンドを実行します。

```
tsm authentication saml map-assertions --user-name <USER-NAME>。
```

tsm authentication を参照してください。

- **SAML アサーションを使用した動的グループ メンバーシップ:**

  Tableau Server 2024.2 以降、SAML (またはサイト SAML) を設定して機能が (サーバー全体またはサイトレベルで) 有効になっている場合、ID プロバイダー (IdP) から送信される SAML XML 応答に含まれるカスタム クレームによって、グループ メンバーシップを動的に制御できます。

  設定すると、ユーザー認証の際、IdP は、ユーザーをアサートするグループ (`https://tableau.com/groups`) とグループ名 (「Group1」と「Group2」など) という、2 つのカスタム グループ メンバーシップ クレームを含む SAML アサーションを送信します。Tableau はアサーションを検証し、グループと、そのグループにパーミッションが依存するコンテンツへのアクセスを有効にします。

  詳細については、アサーションを使用した動的グループ メンバーシップを参照してください。

  SAML XML 応答の例

  ```
  <saml2p:Response
    xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
      .....
      .....
    <saml2:Assertion
  ```

```
        .....
        .....
        xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
        <saml2:AttributeStatement
                    xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
                    <saml2:Attribute
                    Name="https://tableau.com/groups"
                            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
    format:unspecified">
                            <saml2:AttributeValue
                                    xmlns:xs="http://www.w3.org/2001/XMLSchema"
                                    xmlns:xsi="http://www.w3.org/2001/XMLSchema-ins
                                    xsi:type="xs:string">Group1
                            </saml2:AttributeValue>
                            <saml2:AttributeValue
                                    xmlns:xs="http://www.w3.org/2001/XMLSchema"
                                    xmlns:xsi="http://www.w3.org/2001/XMLSchema-ins
                                    xsi:type="xs:string">Group2
                            </saml2:AttributeValue>
           <saml2:Attribute>
        </saml2:AttributeStatement>
      </saml2:Assertion>
    </saml2p:Response>
```

サーバー全体の SAML の構成

サーバー全体の SAML の構成は、Tableau Server のすべてのシングル サインオン (SSO) ユーザーに対し、単一の SAML アイデンティティプロバイダー (IdP) を使用して認証するように設定する場合、またはマルチサイト環境でサイト固有の SAML を構成する最初のステップとして実行します。

サーバー全体の SAML を構成しており、サイトを構成する準備が整っている場合は、サイト固有の SAML の構成を参照してください。

SAML 構成手順では次を前提にしています。

- SAML トピックで説明されているとおり、Tableau Server で SAML 認証を構成する際のオプションを熟知している。
- お使いの環境が SAML 要件を満たすことを確認し、それらの要件で説明されている SAML 証明書ファイルを取得してある。

はじめる前に

災害復旧計画の一環として、証明書と IdP ファイルのバックアップを Tableau Server 外部の安全な場所に保管することをお勧めします。Tableau Server にアップロードする SAML アセットファイルは、クライアントファイル サービスによって他のノードに格納および配布されます。ただし、これらのファイルは復元可能な形式では格納されません。Tableau Server クライアントファイル サービスを参照してください。

> **注:** SSL に同じ証明書ファイルを使用する場合は、代わりに既存の証明書の場所を使用して SAML を構成することも、ファイルをダウンロードして後でこの手順を実行する場合に IdP メタデータファイルをディレクトリに追加しておくこともできます。詳細については、SAML 要件の SAML での SSL 証明書およびキー ファイルの使用を参照してください。

クラスタ内で Tableau Server を実行している場合は、SAML の有効化時に、SAML 証明書、キー、および メタデータファイルがノードに自動的に配布されます。

TSM Web インターフェイスの使用

この手順では、SAML 証明書を TSM にアップロードして、証明書がサーバー構成で適切に保存および配布されるようにする必要があります。SAML ファイルは、この手順で TSM Web インターフェイスを実行しているローカル コンピューター上のブラウザーで使用できる必要があります。

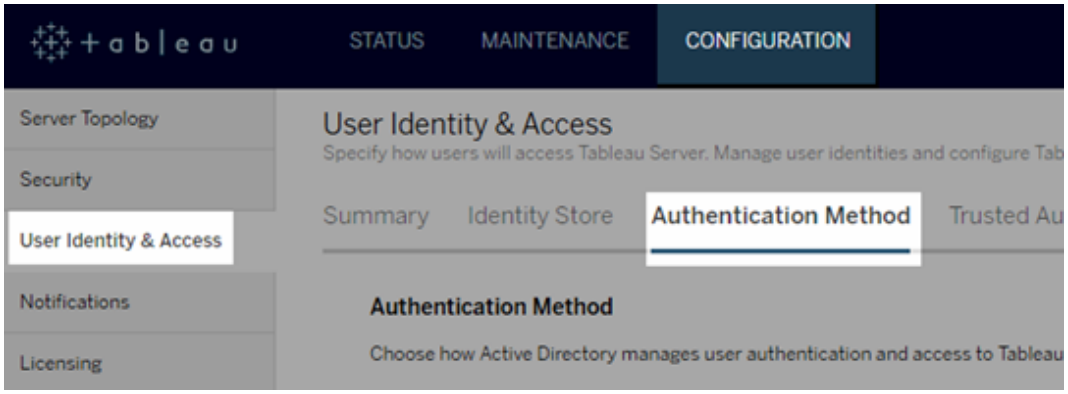
前のセクションで推奨されているように SAML ファイルを収集して Tableau Server に保存した場合は、ファイルをコピーした Tableau Server コンピューターから TSM Web インターフェイスを実行します。

別のコンピューターから TSM Web インターフェイスを実行している場合は、続行する前にすべての SAML ファイルをローカルにコピーする必要があります。以下の手順に従ってローカル コンピューター上のファイルを参照し、TSM にアップロードします。

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、Tableau サービス マネージャーの Web UI へのサインインを参照してください。

2. **[構成]** タブで **[ユーザー ID とアクセス]** を選択してから**[認証方法]** タブを選択します。



3. **[認証方法]** で、**[SAML]** を選択します。

4. SAML セクションが表示されるので、以下の設定を入力して GUI のステップ 1 を完了します (まだチェック ボックスをオンにしてサーバーで SAML を有効にしないでください)。

   - **Tableau Server リターン URL** - Tableau Server のユーザーがアクセスされる http://tableau_server などの URL です。

     https://localhost または末尾にスラッシュがある URL (http://tableau_server/ など) の使用はサポートされていません。

   - **SAML エンティティ ID** — Tableau Server のインストールを IdP に対して一意に識別するエンティティ ID。

     ここで Tableau Server URL を再度入力できます。サイト固有の SAML を後ほど有効にする場合、この URL は各サイトの固有の ID のベースとしても役立ちます。

- **SAML 証明書とキー ファイル — [ファイルの選択]** をクリックして、各ファイルをアップロードします。

  PKCS#8 パスフレーズで保護されたキー ファイルを使用している場合、TSM CLI を使用してパスフレーズを入力する必要があります。

  ```
  tsm configuration set -k wgserver.saml.key.passphrase -v
  <passphrase>
  ```

  GUI のステップ 1 で必要な情報を入力後、GUI のステップ 2 の **[XML メタデータ ファイルのダウンロード]** ボタンが使用できるようになります。

5. GUI のステップ 1 の上にある **[サーバーの SAML 認証を有効化]** チェック ボックスをオンにします。

6. 残りの SAML 設定を完了します。

   a. GUI のステップ 2 および 3 では、メタデータを Tableau Server と IdP の間で交換します(IdP のドキュメントをチェックする必要がある場合があります)。

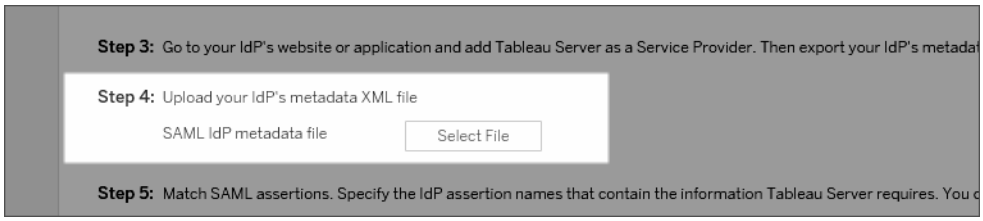
      [XML メタデータ ファイルをダウンロード] を選択して、ファイルの場所を指定します。

      その他の IdP の場合は、IdP アカウントに進み Tableau Server をそのアプリケーションに追加し (サービス プロバイダーとして)、適宜 Tableau メタデータを入力します。

      IdP の Web サイトまたはドキュメントに記載の指示に従って IdP のメタデータをダウンロードします。.xml ファイルを、SAML 証明書およびキー ファイルと同じ場所に保存します。例:

      ```
      /var/opt/tableau/tableau_server/data/saml/idp-metadata.xml
      ```

   b. TSM Web UI に戻ります。GUI のステップ 4 では、IdP メタデータ ファイルへのパスを入力し、次に **[ファイルの選択]** をクリックします。



c. GUI のステップ 5: Tableau Server 構成でアサーション値を変更し、IdP により渡されるアサーション名と一致させる必要がある場合があります。

   アサーション名は、IdP の SAML 構成で確認できます。異なるアサーション名が IdP から渡される場合は、同じアサーション値を使用するように Tableau Server を更新する必要があります。

   ヒント: "アサーション" は主要な SAML コンポーネントであり、アサーション マッピングのコンセプトは最初は扱いにくい場合があります。これを表データコンテキストにすると便利な場合があります。そこでは、アサーション (属性) 名を表の列ヘッダーと同じにします。その列に表示される値の例ではなく、"ヘッダー" 名を入力します。

d. GUI のステップ 6 では、ユーザーがシングル サインオンできる Tableau アプリケーションを選択します。

   > **注:** Tableau Mobile アプリのバージョン 19.225.1731 以上を実行するデバイスでは、モバイル アクセスを無効化するオプションは無視されます。当該バージョンを実行するデバイスで SAML を無効にするには、Tableau Server でクライアント ログイン オプションとしての SAML を無効にする必要があります。

e. SAML サインアウト リダイレクト用に、IdP がシングル ログアウト (SLO) に対応している場合は、サインアウト後にユーザーをリダイレクトするページを、Tableau Server リターン URL に入力したパスからの相対パスとして入力します。

f. (オプション) GUI のステップ 7 で、以下を実行します。

- `AuthNContextClassRef` 属性にコンマ区切りの値を追加します。この属性の使用方法の詳細については、SAML 互換性についての注意事項と要件を参照してください。
- ユーザー名の一部として定義域を送信しない場合は、定義域属性を指定します (例: `domain\username`)。詳細については、「複数のドメインを実行している場合」を参照してください。

7. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。
8. ページ上部の **[変更を保留中]** をクリックします。

9. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

はじめる前に

開始する前に、以下を行ってください。

- IdP の Web サイトまたはアプリケーションから移動し、IdP のメタデータ XML ファイルをエクスポートします。

  IdP からの XML メタデータに **SingleSignOnService** 要素が含まれていることを確認します。これは、次の例のようにバインドを `HTTP-POST` に設定する要素です。

  ```
  <md:SingleSignOnService
  Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
  Location="https://SERVER-NAME:9031/idp/SSO.saml2"/>
  ```

- 証明書ファイルを収集し、Tableau Server 上に配置します。

  Tableau Server フォルダーに SAML という名前の新しいフォルダーを作成し、そのフォルダーに SAML 証明書ファイルのコピーを保存します。例は次のとおりです。

  ```
  /var/opt/tableau/tableau_server/data/saml
  ```

ステップ 1: リターン URL、SAML エンティティ ID を構成し、証明書ファイルとキー ファイルを指定する

1. コマンドプロンプト シェルを開き、サーバーの SAML 設定を構成します (プレースホルダー値を環境パスとファイル名で置き換えます)。

   ```
   tsm authentication saml configure --idp-entity-id
   https://tableau-server --idp-metadata /var/opt/tableau/tableau_
   server/data/saml/<metadata-file.xml> --idp-return-url
   https://tableau-server --cert-file /var/opt/tableau/tableau_
   server/data/saml/<file.crt> --key-file
   /var/opt/tableau/tableau_server/data/saml/<file.key>
   ```

   詳細については、「`tsm authentication saml configure`」を参照してください。

2. パスフレーズで保護されている PKCS#8 キーを使用している場合は、パスフレーズを以下の通りに入力してください。

   ```
   tsm configuration set -k wgserver.saml.key.passphrase -v
   <passphrase>
   ```

3. SAML がまだ Tableau Server 上で有効でない場合、たとえば、初回設定時や、それを無効にしている場合は、ここで SAML を有効にします。

   ```
   tsm authentication saml enable
   ```

4. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

ステップ2:Tableau Server メタデータを生成しIdP を構成する

1. 次のコマンドを実行してTableau Server のために必要な XML メタデータファイルを生成します。

   ```
   tsm authentication saml export-metadata -f <file-name.xml>
   ```

   ファイル名を指定するか、-f パラメーターを省略して、samlmetadata.xml という名前の既定のファイルを作成することができます。

2. IdP の Web サイトまたはアプリケーションで次のように実行します。

   - Tableau Server をサービスプロバイダーとして追加します。

     追加の方法は IdP のマニュアルを参照してください。Tableau Server をサービスプロバイダーとして構成するプロセスの一環として、export-metadata コマンドで生成したTableau Server メタデータファイルをインポートします。

   - IdP がユーザー検証のために**ユーザー名**を属性要素として使用することを確認します。

ステップ3: アサーションを一致させる

Tableau Server 構成でアサーション値を変更し、IdP により渡されるアサーション名と一致させる必要がある場合があります。

アサーション名は、IdP の SAML 構成で確認できます。異なるアサーション名が IdP から渡される場合は、同じアサーション値を使用するように Tableau Server を更新する必要があります。

ヒント: “アサーション” は主要な SAML コンポーネントであり、アサーション マッピングのコンセプトは最初は扱いにくい場合があります。これを表データ コンテキストにすると便利な場合があります。そこでは、アサーション (属性) 名を表の列ヘッダーと同じにします。その列に表示される値の例ではなく、“ヘッダー” 名を入力します。

次の表は、既定のアサーション値と、その値を保存する構成キーを示しています。

アサーション	既定値	キー
ユーザー名	`username`	`wgserver.saml.idpattribute.username`
表示名	`displayName`	Tableau はこの属性タイプをサポートしていません。
メール	`email`	Tableau はこの属性タイプをサポートしていません。
ドメイン	(既定ではマッピングされない)	`wgserver.saml.idpattribute.domain`

指定した値を変更するには、適切なキーと値のペアを使用し、`tsm configuration set` コマンドを実行します。

たとえば、`username` アサーションを `name` という値に変更するには、次のコマンドを実行します。

```
tsm configuration set -k wgserver.saml.idpattribute.username -v
name

tsm pending-changes apply
```

または、`tsm authentication saml map-assertions` コマンドを使用して、指定された値を変更します。

たとえば、ドメイン アサーションを `domain` という値に設定するには、その値を「example.myco.com」として指定し、次のコマンドを実行します。

```
tsm authentication saml map-assertions --domain example.myco.com

tsm pending-changes apply
```

オプション: SAML を使用したクライアントのタイプの無効化

既定では、Tableau Desktop とTableau Mobile アプリの両方で SAML 認証を使用できます。

IdP でこの機能がサポートされていない場合、以下のコマンドを使用して Tableau クライアント向けの SAML サインインを無効にできます。

```
tsm authentication saml configure --desktop-access disable
```

```
tsm authentication saml configure --mobile-access disable
```

> **注**: Tableau Mobile アプリのバージョン 19.225.1731 以上を実行するデバイスでは、`--mobile-access disable` オプションは無視されます。当該バージョンを実行するデバイスで SAML を無効にするには、Tableau Server でクライアントログイン オプションとしての SAML を無効にする必要があります。

```
tsm pending-changes apply
```

オプション: AuthNContextClassRef 値を追加します。

`AuthNContextClassRef` 属性にコンマ区切りの値を追加します。この属性の使用方法の詳細については、SAML 互換性についての注意事項と要件を参照してください。

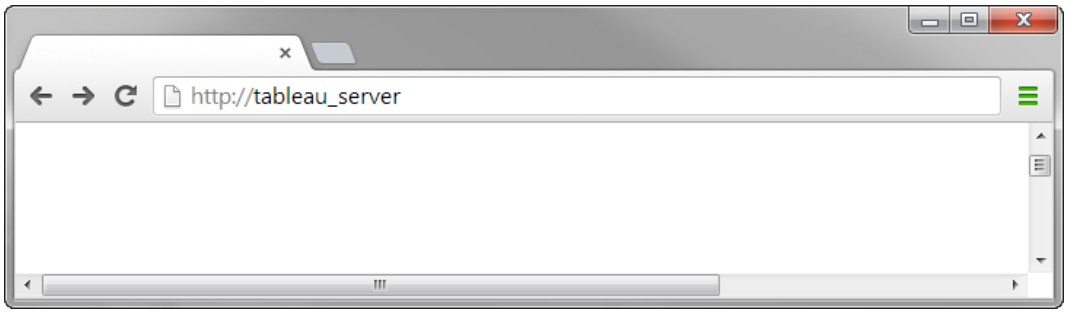
この属性を設定するには、次のコマンドを実行します。

```
tsm configuration set -k wgserver.saml.authcontexts -v <value>
```

```
tsm pending-changes apply
```

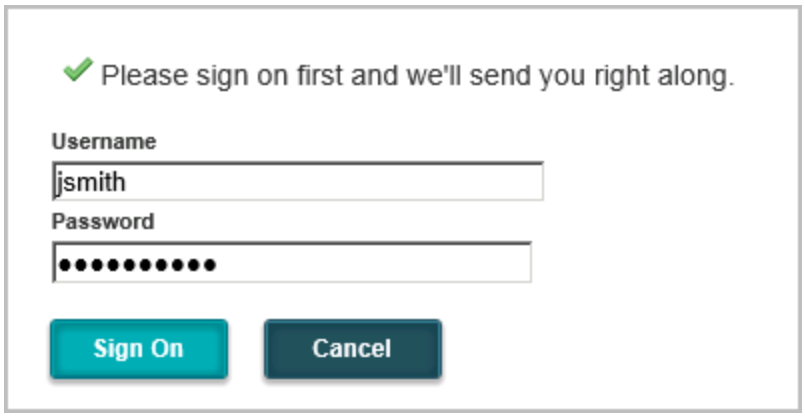
構成のテスト

1. Web ブラウザーで新しいページまたはタブを開き、Tableau Server の URL を入力します。



ブラウザーによって IdP のサインイン フォームにリダイレクトされます。

2. シングル サインオンのユーザー名とパスワードを入力します。



IdP で認証資格情報が検証され、リダイレクトされて Tableau Server のスタートページに戻ります。

Tableau Server で Salesforce IdP を使用した SAML を構成する

このトピックでは、Saleforce IdP を使用して Tableau Server で SAML を構成する方法を示すロードマップのステップを説明します。

このトピックでは、Lightning Web Component (LWC) を有効にする方法についても説明します。LWC を使用すると、Salesforce 管理者は Tableau ビジュアライゼーションを Lightning ページに埋め込むことができます。Tableau Server で Tableau Viz LWC 用に SSO が構成されている場

合、ユーザー エクスペリエンスはシームレスになります。ユーザーが Salesforce にサインインした後、Tableau の埋め込みビューは、Tableau Server へのそれ以上の認証が行われなくても機能します。

Salesforce をアイデンティティプロバイダーとして有効にする

Salesforce を IdP としてまだ設定していない場合は、Salesforce ヘルプ サイトの「SAML ID プロバイダとしての Salesforce の有効化」の手順に従います。

Salesforce を SAML IdP として有効にするプロセス中に、証明書を指定するか、Salesforce が SAML で使用するための自己署名証明書を生成します。この証明書 (.crt ファイル) とそれに関連するプロバイダー メタデータ ファイル (.xml) をダウンロードします。これらのアセットは次のステップで必要になります。

Tableau Server で SAML を構成する

前のステップでダウンロードした証明書ファイルとメタデータ ファイルを使用して、サーバー全体の SAML の構成の手順に従います。

構成プロセスの一部として、Tableau Server の SAML エンティティ ID とログイン URL を生成します。これらのアセットは次のステップで必要になります。

(オプション) サーバー全体の SAML を構成したら、Tableau Server でサイト固有の SAML を構成できます。サイト固有の SAML の構成を参照してください。

Tableau Server を Salesforce の接続アプリケーションとして追加する

Salesforce ヘルプサイトの「SAML 2.0 を使用した接続アプリケーションとしてサービス プロバイダを統合」の手順に従います。

このプロセスでは、新しい接続アプリ (Tableau Server) を作成します。いくつかの重要な詳細は次のとおりです。

- [SAML の有効化] を選択します
- 前のセクションで Tableau を構成するときに生成したエンティティ ID とログイン URL を入力します。デフォルトでは、ログイン URL は `https://<tableauserver>/wg/saml/SSO/index.html` です。

- IdP 証明書については、Salesforce を SAML プロバイダーとして有効にしたときに指定または生成したものと同じ証明書を必ず選択してください。
- ユーザーが組織から Tableau Server にサインインできるようにするには、適切なプロファイルまたはパーミッション セットを割り当てて、接続アプリへのアクセスを管理します。

Lightning Web コンポーネントを有効にする

Tableau Server で SAML SSO の LWC を有効にするには、フレーム内認証を有効にする必要があります。LWC を有効にする前に、Tableau Server の最新のメンテナンス リリースにアップグレードしてください。追加バージョンの次の2 つの重要事項に留意してください。

- 最新のメンテナンス リリースを実行していないときに、ユーザーが Chrome ブラウザーを実行して Salesforce Lightning にアクセスしている場合は、Tableau の KB 記事「Chrome を 80 以降に更新すると、埋め込みビューの読み込みに失敗する」を参照してください。
- Tableau Server でサイト固有の SAML を構成して LWC を使用している場合は、Tableau Server 2020.4 以降を実行している必要があります。

Tableau Server で SAM L を構成したら、次の Tableau サービス マネージャー コマンドを実行してフレーム内認証を有効にします。

```
tsm configuration set -k wgserver.saml.iframed_idp.enabled -v true

tsm pending-changes apply
```

Tableau ビューを Salesforce に埋め込む

Tableau Server を SSO 用に構成したら、Salesforce org に LWC をインストールし、Tableau ビューを埋め込むことができます。「Tableau ビューを Salesforce に埋め込む」を参照してください。

Tableau Viz Lightning Web コンポーネントの SAML の構成

Tableau には、Salesforce Lightning ページ内に Tableau ビジュアライゼーションを埋め込むための Lightning Web コンポーネント(LWC) があります。

このトピックでは、Salesforce Lightning ページで Tableau の埋め込みビジュアライゼーションの SSO エクスペリエンスを有効にする方法を説明します。Tableau Viz LWC シナリオの SSO には SAML 構成が必要です。Tableau 認証に使用する SAML IdP は、Salesforce インスタンスで使用する Salesforce IdP または同じ IdP のいずれかである必要があります。

このシナリオでは、Salesforce 管理者が Tableau Viz LWC を Lightning ページにドラッグ & ドロップして、ビジュアライゼーションを埋め込むことができます。埋め込みビューへの URL を入力すると、管理者が Tableau Server で閲覧できるビューがダッシュボードに表示されます。

Tableau Server で Tableau Viz LWC 用にシングル サインオン (SSO) が構成されている場合、ユーザー エクスペリエンスはシームレスになります。ユーザーが Salesforce にサインインした後、Tableau の埋め込みビューは、Tableau Server へのそれ以上の認証が行われることなく機能します。

SSO が構成されていない場合、Tableau Server からの埋め込みビジュアライゼーションを表示するには、ユーザーは Tableau Server でもう一度認証を行う必要があります。

要件

- Tableau 認証に使用する SAML IdP は、Salesforce インスタンスで使用する Salesforce IdP または同じ IdP のいずれかである必要があります。Tableau Server で Salesforce IdP を使用した SAML を構成するを参照してください。
- SAML が Tableau Server 上で構成されている必要があります。サーバー全体の SAML の構成またははサイト固有の SAML の構成を参照してください。
- SAML を Salesforce 用に構成する必要があります。
- Tableau Viz Lightning Web コンポーネントをインストールします。「Tableau ビューを Salesforce に埋め込む」を参照してください。

認証ワークフローの構成

Tableau の埋め込みビューを使用して Lightning にアクセスするユーザーのサインイン エクスペリエンスを最適化するには、追加の構成が必要になる場合があります。

シームレスな認証を行うことができるユーザー エクスペリエンスを重要視する場合は、いくつかの追加の構成を行う必要があります。このコンテキストでは、“シームレス” とは、Tableau Viz LWC SSO が有効になっている Salesforce Lightning ページにアクセスするユーザーが、Tableau の埋め込みビューを表示するためのアクションを実行する必要がないことを意味します。シームレスなシナリオでは、ユーザーが Salesforce にログインすると、Tableau の埋め込みビューは追加のユーザー アクションなしで表示されます。このシナリオは、*フレーム内認証*によって有効になります。

シームレスなユーザー エクスペリエンスを実現するには、Tableau Server とIdP でフレーム内認証を有効にする必要があります。以下のセクションでは、フレーム内認証の構成方法について説明します。

一方、Lightning ページを操作するユーザーが、Tableau の埋め込みビューを表示するために [サインイン] ボタンをクリックしなければならないシナリオがあります。このシナリオでは、ユーザーが Tableau の埋め込みビューを表示するためにアクションをもう1 つ実行する必要があり、「ポップアップ認証」と呼ばれます。

フレーム内認証を有効にしていない場合、ポップアップ認証が既定のユーザー エクスペリエンスになります。

Tableau Server でフレーム内認証を有効にする

Tableau Server でフレーム内認証を有効にする前に、Tableau Server で SAML を構成し、有効にしておく必要があります。

次の TSM コマンドを実行してフレーム内認証を有効にします。

```
tsm configuration set -k wgserver.saml.iframed_idp.enabled -v true

tsm pending-changes apply
```

注: Tableau Server では、クリックジャック攻撃防止機能が既定で有効になっています。フレーム内認証を有効にすると、フレーム内認証のセッションでクリックジャック攻撃防止機能が一時的に無効になります。クリックジャック攻撃防止機能を無効にするリスクを評価する必要があります。「クリックジャック保護」を参照してください。

Tableau Server のバージョン

最高のユーザー エクスペリエンスを実現するために、Tableau Server の最新のメンテナンス リリースを実行してください。

最新のメンテナンス リリースを実行していないときに、ユーザーが Chrome ブラウザーを実行して Salesforce Lightning にアクセスしている場合は、Tableau の KB 記事「Chrome を 80 以降に更新すると、埋め込みビューの読み込みに失敗する」を参照してください。

SAML IdP を使用してフレーム内認証を有効にする

前述のように、Salesforce Mobile で認証をシームレスに行うためのユーザ エクスペリエンスには、IdP でフレーム内認証がサポートされている必要があります。この機能は、IdP では “iframe 埋め込み” または “iframe 保護“ とも呼ばれています。

Salesforce の承認リストのドメイン

場合によっては、IdP ではドメイン別のフレーム内認証のみを有効にできます。このような場合は、フレーム内認証を有効にするときに、次の Salesforce ワイルドカードのドメインを設定します。

```
*.force

 *.visualforce
```

Salesforce IdP

Salesforce IdP では既定でフレーム内認証がサポートされています。Salesforce 構成でフレーム内認証を有効にしたり、構成したりする必要はありません。ただし、上記の説明に従って Tableau Server で TSM コマンドを実行する必要があります。

Okta IdP

Okta ヘルプ センターのトピック一般的なカスタマイズのオプションで「*Okta を iframe に埋め込む*」を参照してください。

Ping IdP

Ping サポートのトピックPing Federate で "X-Frame-Options=SAMEORIGIN" ヘッダーを無効にする方法を参照してください。

OneLogin IdP

One Login ナレッジ ベース記事 アカウント所有者のアカウント設定の「*フレーム保護*」を参照してください。

ADFS とEntraID IdP

Microsoft では、すべてのフレーム内認証がブロックされており、有効にすることはできません。代わりに、Microsoft では 2 番目のウィンドウでのポップアップ認証のみがサポートされます。その結果、いくつかのブラウザーでポップアップ動作がブロックされることがあり、その場合はユーザーが `force.com` および `visualforce.com` サイトのポップアップを受け入れる必要があります。

Salesforce Mobile アプリ

ユーザーが主に Salesforce Mobile アプリで Lightning を操作する場合は、次のシナリオに注意する必要があります。

- Salesforce Mobile アプリでは、Tableau の埋め込みビューが表示されるように SSO/SAML を構成する必要があります。
- Salesforce Mobile アプリには フレーム内認証が必要です。ポップアップ認証は機能しません。代わりに、Salesforce Mobile アプリのユーザーには Tableau のサインイン ボタンが表示されますが、Tableau にサインインすることはできません。
- ADFS および Azure AD IdP は Mobile アプリに対応していません。
- Mobile アプリでは、OAuth トークンを使用して SSO を有効にします。OAuth トークンによりユーザーが更新およびログアウトされるため、ユーザーは再度ログインしなければならないことがあります。詳細については、Tableau の KB 記事「Salesforce Mobile アプリ上の Tableau Viz Lightning Web コンポーネントによりサインインが求められる」を参照してください。
- SSO の動作は、Salesforce Mobile アプリ(iOS 対 Android) のバージョンとIdP によって異なります。

IdP	Mobile OS	SSO の動作
Salesforce IdP	Android	SSO は最初は機能しますが、ユーザーはしばらくしてからサインインする必要があります。
	iOS	

外部 IdP	Android	SSO が機能しません。ユーザーは手動でサインインする必要があります。(ユーザーが Tableau の埋め込みビューにアクセスできるように SSO を構成する必要があります)。
	iOS	SSO は最初は機能しますが、ユーザーはしばらくしてからサインインする必要があります。

Tableau Server での Azure AD IdP を使用した SAML の構成

Azure AD を SAML アイデンティティプロバイダー (IdP) として構成し、サポートされているシングル サインオン (SSO) アプリケーションに Tableau Server を追加できます。Azure AD を SAML および Tableau Server と統合する場合、ユーザーは標準的なネットワーク認証資格情報を使用して Tableau Server にサインインできます。

始める前に: 前提条件

Tableau Server および SAML を Azure AD で構成する前に、お使いの環境には以下が必要です。

- SHA-2 (256 または 512 ビット) 暗号化を使用して暗号化され、以下のセクションに記載された追加要件を満たす SSL 証明書。
    - SSL 証明書要件
    - SAML 証明書およびアイデンティティプロバイダー (IdP) の要件
- ユーザーが既定のドメインではないドメインからサインインしている場合は、外部 ID ストアを使用した展開におけるユーザー管理と SAML 要件 を参照して、ドメイン属性値が設定および定義されていることを確認し、後でサインインの問題が発生しないようにします。

ステップ 1: Azure AD への SSL 接続を確認する

Azure AD には SSL 接続が必要です。まだこのようにしていない場合は、上記の要件を満たす証明書を使用し、Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成するのステップを完了してください。

または、SSL がリバースプロキシまたはロードバランサーで終端されるように Tableau Server が構成されている場合 (一般には SSL オフロードと呼ばれます)、外部 SSL を構成する必要はありません。

組織で Azure AD アプリプロキシを使用している場合は、以下のセクション「Azure AD アプリプロキシ」を参照してください。

ステップ 2: Tableau Server 上で SAML を構成する

XML ファイルの Tableau Server メタデータをダウンロードし、サーバー全体の SAML の構成のステップを完了してください。その時点でここに戻り、次のセクションを続行します。

ステップ 3: Azure AD クレーム ルールを構成する

マッピングでは大文字と小文字が区別され、スペルが完全に同じになっている必要があります。入力した内容はダブルチェックしてください。次の表に、一般的な属性とクレーム マッピングを示します。Azure AD の具体的な設定を使用してその属性を検証する必要があります。

LDAP 属性	**出力するクレーム タイプ**
onpremisessamaccountname	username
Given-Name	firstName **注:** これはオプションです。
Surname	lastName **注:** これはオプションです。
netbiosname	domain **注:** これは、ユーザーが既定のドメインではないドメインからサインインしている場合にのみ必要です。

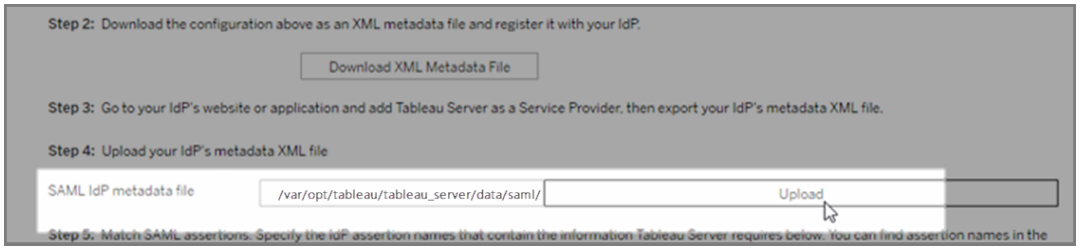
一部の組織では、Azure AD を SAML IdP として使用し、Active Directory を Tableau Server のアイデンティティストアとして使用しています。この場合、通常、`username` は sAMAccountName

の名前です。Azure AD 内の sAMAccountName 属性を識別して、username 属性にマップする方法については、マイクロソフトのドキュメントを参照してください。

ステップ 4: Azure AD メタデータを Tableau Server に提供する

1. TSM の Web UI に戻り、**[構成] > [ユーザー ID とアクセス] > [認証方法]** タブに移動します。

2. SAML 設定画面のステップ 4 で、Azure AD からエクスポートした XML ファイルの場所を入力し、**[アップロード]** を選択します。



3. サーバー全体の SAML の構成で指定されている残りの手順 (アサーションのマッチングとクライアントタイプアクセスの指定) を完了します。変更を保存して適用します。

4. SAML を構成するのが今回が初めてではない場合、次のステップを実行してください。

   a. Tableau Server を停止し、TSM CLI を開いて、次のコマンドを実行します。

   ```
   tsm configuration set -k wgserver.saml.sha256 -v true

   tsm authentication saml configure -a -1
   ```

   b. 変更を適用します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示さ

れないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

Azure AD アプリプロキシ

Tableau Server の手前で Azure AD アプリプロキシを実行していて、SAML が有効になっている場合は、Azure AD アプリプロキシに追加の設定が必要です。

Tableau Server は、SAML が有効になっている場合にのみ、1 つの URL からトラフィックを受け入れることができます。ただし、既定では、Azure AD アプリプロキシは外部 URL と内部 URL を設定します。

これらの値の両方をカスタム ドメイン内の同じ URL に設定する必要があります。詳細については、Microsoft のドキュメント「Azure AD アプリケーション プロキシでカスタム ドメインを構成する」を参照してください。

トラブルシューティング

Azure AD アプリプロキシ

場合によっては、ビューへのリンクは内部でレンダリングされますが、トラフィックが Azure AD アプリプロキシを通過する場合は外部でのレンダリングに失敗します。この問題は、URL にナンバー記号 (#) があり、ユーザーがブラウザーでリンクにアクセスしている場合に発生します。Tableau Mobile アプリは、ポンド記号がある URL にアクセスできます。

ユーザー セッションのタイムアウトが無視されています

Tableau Server が SAML 用に構成されている場合、IdP の最大認証期間が Tableau の最大認証期間より大きい値に設定されているため、ユーザーがサインインするときにエラーが発生する可能性があります。この問題を解決するには、tsm configuration set のオプション `wgserver.saml.forceauthn` を使用することにより、Tableau が認証要求をリダイレクトするたびに、ユーザーの IdP セッションがまだアクティブであってもユーザーを再認証するよう IdP に要求することができます。

たとえば、Azure AD 設定 `maxInactiveTime` が Tableau Server の設定 `maxAuthenticationAge` よりも大きい場合、Tableau は認証要求を IdP にリダイレクトし、IdP はその後、ユーザーがすでに認証されているというアサーションを Tableau に送信します。しかし、ユーザーは Tableau Server の `maxAuthenticationAge` の範囲外で認証されているため、Tableau はユーザー認証を拒否します。このような場合は、次のいずれかを実行します。

- `wgserver.saml.forceauthn` オプションを有効にして、Tableau が認証要求をリダイレクトするたびにユーザーを再認証するよう IdP に要求します。詳細については、wgserver.saml.forceauthn を参照してください。
- Tableau Server の `maxAuthenticationAge` の設定値を大きくします。詳細については、tsm authentication タイプの “a, --max-auth-age <max-auth-age>” を参照してください。

AppID の不一致

vizportal.log ファイルを確認すると、*「対象のオーディエンスが受信者と一致しません」*というエラーが表示される場合があります。

この問題を解決するには、appID が送信された ID と一致していることを確認してください。Azure は、使用中のアプリでアプリケーション ID を使用するときに、appID に "SPN" を自動的に追加します。Tableau SAML 設定で、アプリケーション ID に "SPN:" プレフィックスを追加して値を変更することができます。

例: SPN:myazureappid1234

Tableau Server での AD FS を使用した SAML の構成

Active Directory Federation Services (AD FS) を SAML アイデンティティプロバイダーとして構成し、サポートされているシングル サインオン アプリケーションに Tableau Server を追加できます。AD FS を SAML および Tableau Server と統合する場合、ユーザーは標準的なネットワーク認証資格情報を使用して Tableau Server にサインインできます。

前提条件

Tableau Server または および SAML を AD FS で構成する前に、お使いの環境には以下が必要です。

- AD FS 2.0 (またはそれ以降) および IIS がインストールされた Microsoft Windows Server 2008 R2 (またはそれ以降) を実行するサーバー。
- AD FS サーバーを保護する (例: リバースプロキシの使用) ことをお勧めします。ファイアウォールの外から AD FS サーバーにアクセス可能な場合、Tableau は AD FS がホストするサインインページにユーザーをリダイレクトできます。
- SHA-2 (256 または 512 ビット) 暗号化を使用して暗号化され、以下のセクションに記載された追加要件を満たす SSL 証明書。
    - SSL 証明書要件
    - SAML 証明書およびアイデンティティプロバイダー (IdP) の要件

ステップ 1: AD FS への SSL 接続を確認する

AD FS には SSL 接続が必要です。まだこのようにしていない場合は、上記の要件を満たす証明書を使用し、Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成するのステップを完了してください。

または、SSL が終了されるリバースプロキシまたはロードバランサー (一般には SSL オフロードという) を使用するように Tableau Server が構成されている場合、外部 SSL を構成する必要はありません。

ステップ 2: Tableau Server 上で SAML を構成する

XML ファイルの Tableau Server メタデータをダウンロードし、サーバー全体の SAML の構成のステップを完了してください。その時点でここに戻り、次のセクションを続行します。

ステップ 3: Tableau Server からのサインイン要求を承認するための AD FS の設定

注: これらの手順は、サードパーティのアプリケーションを反映しており、通知なしに変更されることがあります。

Tableau Server のサインイン要求を許容するように AD FS を構成することは、複数のステップで構成されるプロセスであり、Tableau Server の XML メタデータファイルを AD FS にインポートすることから始まります。

1. 次のいずれかの手順を実行して **[Relying Party Trust の追加] ウィザード**を開きます。

   Windows Server 2008 R2 の場合:

   a. **[スタート]**、**[管理ツールへ]> [AD FS 2.0]** の順に選択します。

   b. **[AD FS 2.0]** の **[Trust のリレーションシップ]** で **[Relying Party Trusts]** フォルダーを右クリックし、**[Relying Party Trust の追加]** をクリックします。

   Windows Server 2012 R2 の場合:

   a. **Server Manager** を開き、**[ツール]** メニューで **[AD FS の管理]** をクリックします。

   b. **[AD FS 管理]** の **[アクション]** メニューで、**[依存先との信頼関係の追加]** をクリックします。

2. **[依存先との信頼関係の追加] ウィザード**で、**[スタート]** をクリックします。

3. **[データ ソースの選択]** ページで **[ファイルから Relying Party に関するデータをインポートする]** を選択し、**[参照]** をクリックして Tableau Server XML メタデータファイルを指定します。既定のファイル名は、**samlspmetadata.xml** です。

4. **[次へ]** をクリックして、**[表示名の指定]** ページの **[表示名]** および **[メモ]** ボックスに Relying Party Trust の名前と説明を入力します。

5. [次へ] をクリックして **[多要素認証を今すぐ設定する]** ページにスキップします。

6. [次へ] をクリックして **[発行承認規則の選択]** ページにスキップします。

7. [次へ] をクリックして **[Trust の追加準備完了]** ページにスキップします。

8. **[完了]** ページで **[ウィザードを閉じるときに Relying Party Trust に対するクレーム ルールの編集ダイアログを開く]** チェック ボックスをオンにし、**[閉じる]** をクリックします。

次に、**[クレーム ルールの編集]** ダイアログで作業して AD FS から送信されるアサーションが Tableau Server が期待するアサーションと一致するようにするルールを追加します。最低限、Tableau Server では メール アドレスが必要となります。ただし、メールに加えて姓と名を含めることで、Tableau Server に表示されるユーザー名が AD アカウントと同じになります。

1. **[クレーム ルールの編集]** ダイアログ ボックスで、**[ルールの追加]** をクリックします。
2. **[ルール タイプの選択]** ページの **[クレーム ルール テンプレート]** で、**[LDAP 属性をクレームとして送信]** を選択してから、**[次へ]** をクリックします。
3. **[クレーム ルールの構成]** ページの **[クレーム ルール名]** ボックスで、わかりやすい任意の名前を入力します。
4. **[属性の保存]** では、**[Active Directory]** を選択し、以下のようにマッピングを行ってから **[完了]** をクリックします。

   マッピングでは大文字と小文字が区別され、スペルが完全に同じになっている必要があります。入力した内容はダブルチェックしてください。次の表に、一般的な属性とクレーム マッピングを示します。特定の Active Directory 構成の属性を確認します。

LDAP 属性	出力するクレーム タイプ
SAM-Account-Name	名前 ID
SAM-Account-Name	username
Given-Name	firstName
Surname	lastName

AD FS 2016 以降を実行している場合は、ルールを追加してすべてのクレーム値をパス スルーする必要があります。以前のバージョンの AD FS を実行している場合は、次の手順に進み、AD FS メタデータをエクスポートします。

1. **[Add Rule (ルールの追加)]** をクリックします。
2. **[Claim rule template (クレーム ルール テンプレート)]** で、**[Pass Through or Filter an Incoming Claim (受信 クレームのパス スルーまたはフィルター)]** を選択します。
3. **[Claim rule name (クレーム ルール名)]** に、「Windows」と入力します。

4. **[Edit Rule - Windows (ルールの編集 - Windows)]** ポップアップで、以下の操作を実行します。
    - **[Incoming claim type (受信するクレーム タイプ)]** で、**[Windows account name (Windows アカウント名)]** を選択します。
    - **[Pass through all claim values (すべてのクレーム値をパス スルー)]** を選択します。
    - **[OK]** をクリックします。

後で Tableau Server にインポートする予定の AD FS メタデータをエクスポートします。また、Tableau Server 向けに適切に構成およびエンコードされていることを確認して、お使いの SAML 構成の他の AD FS 要件を検証します。

1. AD FS Federation メタデータを XML ファイルにエクスポートしてから、そのファイルを **https://*<adfs server name>*/federationmetadata/2007-06/FederationMetadata.xml** からダウンロードします。

2. Sublime Text または Notepad++ のようなテキスト エディタで メタデータ ファイルを開き、BOM なしの UTF-8 で適切にエンコーディングされていることを確認します。

   ファイルがその他のエンコーディング タイプを示した場合、テキスト エディタから正しいエンコーディングで保存します。

3. AD FS がフォームベースの認証を使用していることを検証します。サインインはブラウザー ウィンドウで行われるため、このタイプの認証を既定にするには、AD FS が必要です。

   c:\inetpub\adfs\ls\web.config を編集し、タグを検索してから行を移動すると、リストで最初に表示されます。ファイルを保存して、IIS が自動的に再読み込みできるようにします。

   > **注: c:\inetpub\adfs\ls\web.config** ファイルが表示されない場合は、IIS がインストールされておらず、AD FS サーバーにも設定されていません。

4. (オプション) このステップは AD FS がサイト特有の SAMLの IDP として構成されている場合のみ必要です。このステップは AD FS がサーバー全体の SAML の IDP として構成されている場合は不要です。

追加の AD FS Relying Party 識別子を構成します。これにより、システムは SAML ログアウトを使用して AD FS の問題を回避できるようになります。

次のいずれかを実行します。

Windows Server 2008 R2 の場合:

a. **AD FS 2.0** で、事前に Tableau Server 用に作成した Relying Party を右クリックして **[プロパティ]** をクリックします。

b. **[識別子]** タブの **[Relying Party 識別子]** ボックスに、「**https://<tableauservername>/public/sp/metadata**」と入力してから **[追加]** をクリックします。

Windows Server 2012 R2 の場合:

a. **[AD FS 管理]** の **[Relying Party Trusts]** リストで、事前に Tableau Server 用に作成した Relying Party を右クリックして **[プロパティ]** をクリックします。

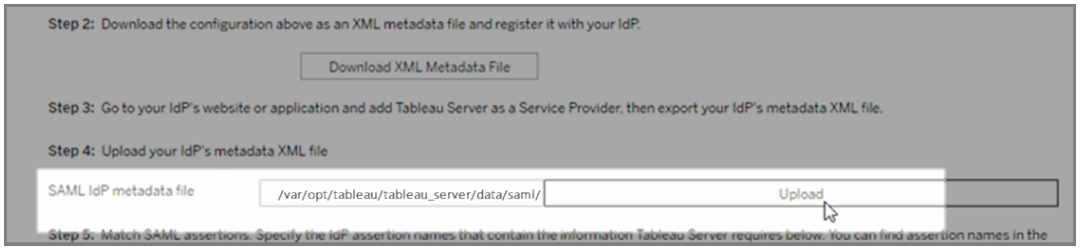
b. **[識別子]** タブの **[Relying Party 識別子]** ボックスに、「**https://<tableauservername>/public/sp/metadata**」と入力してから **[追加]** をクリックします。

> **注**: AD FS は、同じインスタンスに対して単一の Relying Party を使用する Tableau Server で使用できます。AD FS では同じインスタンスに複数の Relying Party (複数サイトの SAML サイトやサーバー全体、およびサイトの SAML 構成など) を使用することはできません。

ステップ 4: AD FS メタデータを Tableau Server に提供

1. TSM の Web UI に戻り、**[構成] > [ユーザー ID とアクセス] > [認証方法]** タブに移動します。

2. SAML 構成ウィンドウのステップ 4 で、AD FS からエクスポートした XML ファイルの場所を入力し、**[アップロード]** を選択します。



3. サーバー全体の SAML の構成で指定されている残りの手順 (アサーションのマッチングとクライアントタイプ アクセスの指定) を完了します。

4. 変更を保存して適用します。

5. SAML を構成するのが今回が初めてではない場合、次のステップを実行してください。

   a. Tableau Server を停止し、TSM CLI を開いて、次のコマンドを実行します。

   ```
   tsm configuration set -k wgserver.saml.sha256 -v true

   tsm authentication saml configure -a -1
   ```

   b. 変更を適用します。

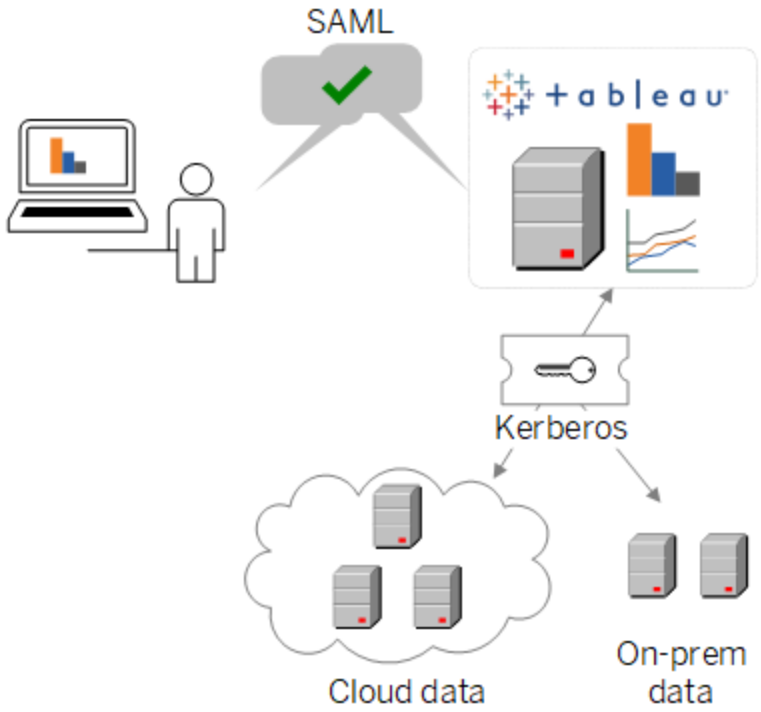
   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

Kerberos データベース委任とともに SAML SSO を使用する

Windows Active Directory (AD) 環境では、Tableau Server への SAML シングル サインオン (SSO) を、Kerberos データベース委任と合わせて可能にできます。これにより、認証ユーザーが Tableau Server のほか、パブリッシュされたワークブックやデータ ソースで定義されている参照元データに直接アクセスできます。

プロセスの概要



一般的なシナリオの場合:

1. Tableau アナリストの 1 人が Tableau Server にダッシュボードをパブリッシュします。そのダッシュボードには、たとえば Hadoop クラスタへの接続が含まれており、これは Kerberos の認証資格情報を受け入れるように構成されます。

   その後、ワークブック パブリッシャーが確認用に同僚にリンクを送信します。

2. 同僚がリンクをクリックすると、Tableau Server は SAML SSO プロセス経由でユーザーを認証します。ユーザーの認可スキームを確認し、許可されている場合には、Tableau Server のキータブを使用してユーザーの代理で参照元データベースにアクセスします。これにより、ユーザーに表示権限がある Hadoop データがダッシュボードに入力されます。

Tableau Server で Kerberos による SAML を構成する

Kerberos による SAML は本質的に、それぞれを個別に有効化するプロセスを完了する際に機能します。

1. サーバー全体の SAML の構成の説明に従って Tableau Server で SAML を構成します。
2. Kerberos 委任の有効化および関連記事の説明に従って、Tableau Server と参照元データベースで Kerberos の認証資格情報を受け入れるように構成します。

サイト固有の SAML の構成

マルチサイト環境でサイト固有の SAML はシングルサインオンの有効化を望み、同時に複数の SAML アイデンティティプロバイダー (IdP) または IdP アプリケーションを使用している場合に使用します。サイト SAML を有効化すると、各サイトの IdP または IdP アプリケーションを指定したり、一部のサイトで SAML を使用し、他のサイトでは既定のサーバー全体の認証方法を使用するように構成できます。

全サーバーユーザーに SAML を使用して同じ IdP アプリケーションを通してサインインしてほしい場合は、サーバー全体の SAML の構成を参照してください。

サイト固有 SAML 有効化の前提条件

サイトレベルで SAML のシングルサインオンを有効化する前に、以下の要件を満たしてください。

- Tableau Server アイデンティティストアがローカル アイデンティティストア用に構成されている必要があります。

  Tableau Server が Active Directory や OpenLDAP などの外部アイデンティティストアを使用して構成されている場合、サイト固有 SAML を構成することはできません。

- お使いの環境と IdP が一般的な SAML 要件を満たしていることを確認してください。

以下の一部の機能 (但し、これらに限定されるものではありません) は、サーバー全体の SAML 展開でのみサポートされています。

- パスワードで保護されたキーファイルは、サイト固有の SAML 展開ではサポートされていません。

- サイト固有の SAML を構成する前に、サーバー全体の SAML を構成する必要があります。サーバー全体の SAML を有効にする必要はありませんが、サイト固有の SAML にはサーバー全体の構成が必要です。サーバー全体の SAML の構成を参照してください。

- SAML 証明書ファイルの格納場所をメモしておいてください。サーバーにおけるサイト固有 SAML のサポートの構成場合にこの情報を提供することになります。

  詳細情報については、サーバー全体の SAML を構成するトピックの「メタデータと証明書ファイルを配置する」を参照してください。

- Tableau Server をサービスプロバイダーとして IdP に追加する。この情報は IdP プロバイダーのドキュメントに掲載されています。

- サイト SAML IdP をホストしている PC と Tableau Server をホストしている PC とのシステムクロックが 59 秒以内であることを確認します。Tableau Serer には、Tableau Server PC と IdP 間の応答スキュー (時間差) を調整するための構成オプションはありません。

サイト固有の SAML に関連するサーバー全体の設定

リターン URL およびエンティティ ID: サイト固有 SAML を構成している設定では、Tableau はこれらの設定に基づいてサイト固有のリターン URL とエンティティ ID を提供します。サイト固有のリターン URL とエンティティ ID は変更できません。これらの構成は、サーバー全体の SAML の構成に記載されているように、TSM によって設定されます。

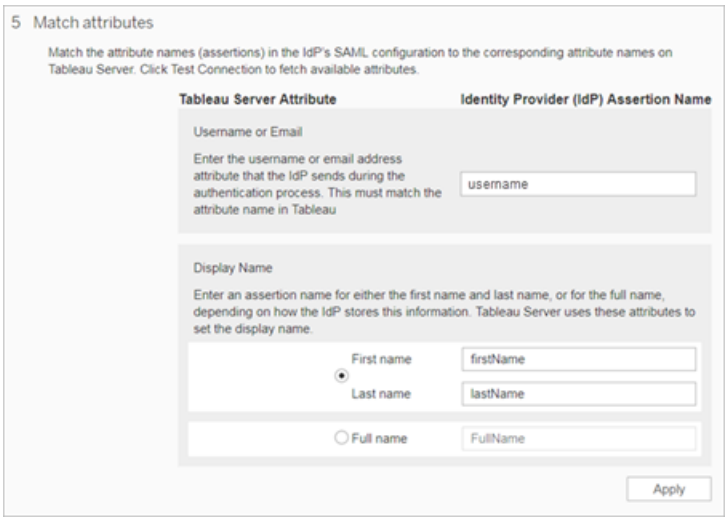
認証期間および応答スキュー: サーバー全体の設定、最大認証期間、および応答スキューは、サイト固有の SAML には適用されません。以下の構成はハードコードされています。

- 最大認証期間とは、IdP から発行された認証トークンが有効である期間を指します。サイト固有の SAML のハードコードされた最大認証期間は 24 日間です。

- 応答スキューは、Tableau Server の時刻と、まだメッセージを処理できるアサーション作成 (IdP サーバーの時刻に基づく) 時刻の最大秒数の差です。この場合のハードコードされたサイト固有の値は 59 秒です。

ユーザー名: 必須。サーバー全体の SAML 構成属性に加えて、サイト固有の SAML 構成属性を "username" に設定する必要があります。

> **注:** サイト固有の SAML にサーバー全体の SAML の既定を使用して正常に動作させるには、wgserver.saml.idpattribute.username 構成キーを使用してサーバー全体の SAML に "username" という名前のユーザー名属性を設定する必要があります。サーバー全体の SAML に使用される IdP は、"username" という名前の属性でユーザー名を提供する必要があります。



HTTP POST と HTTP REDIRECT: サイト固有の SAML の場合、Tableau Server は HTTP-POST、HTTP-REDIRECT、および HTTP-POST-SimpleSign をサポートします。

サーバーにおけるサイト固有 SAML のサポートの構成

上記の前提条件を満たした後、以下のコマンドを実行してサーバーがサイト固有 SAML をサポートするように構成できます。

1. サーバー全体の SAML の構成。少なくとも次の TSM コマンドを実行する必要があります (既にサーバー全体の SAML が構成済みである場合は、ステップ 2 に進みます)。

```
tsm authentication saml configure --idp-entity-id <tableau-
server-entity-id> --idp-return-url <tableau-server-return-url>
--cert-file <path-to-saml-certificate.crt> --key-file <path-to-
saml-keyfile.key>
```

2. サイトの SAML を有効にします。次のコマンドを実行します。

```
tsm authentication sitesaml enable
```

```
tsm pending-changes apply
```

コマンドについて

`sitesaml enable` コマンドは、Tableau Server Web UI の各サイトの **[設定]** ページで **[認証]** タブを表示します。サイトの SAML をサポートするようにサーバーを構成したら、サイトの SAML の構成に進み、**[認証]** タブの設定作業ができます。

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

`pending-changes apply` の実行時に実行されるコマンドと設定を確認する場合は、まず以下のコマンドを実行します。

```
tsm pending-changes list --config-only
```

サイトの SAML の構成

このセクションでは、Tableau Server または の設定ページの **[認証]** タブで表示されている設定手順について説明します。セルフホステッド Tableau Server インストールでは、このページは、サイト固有の SAML のサポートがサーバー レベルで有効になっている場合にのみ表示されます。

注: このプロセスを完了するには、IdP が提供するドキュメンテーションも必要です。SAML 接続用のサービスプロバイダーの構成や定義、またはアプリケーションの追加に関するトピックをご覧ください。

ステップ 1: Tableau からメタデータをエクスポート

Tableau Server と IdP との間に SAML 接続を作成するには、2 つのサービスとの間に必要なメタデータを交換する必要があります。Tableau Server からメタデータを取得するには、次の方法のいずれかを選択します。正しいオプションを確認するには、IdP の SAML 構成ドキュメントを参照してください。

- **[メタデータのエクスポート]** ボタンを選択して、Tableau Server SAML エンティティ ID、Assertion Consumer Service (ACS) URL、および X.509 証明書を含む XML ファイルをダウンロードします。

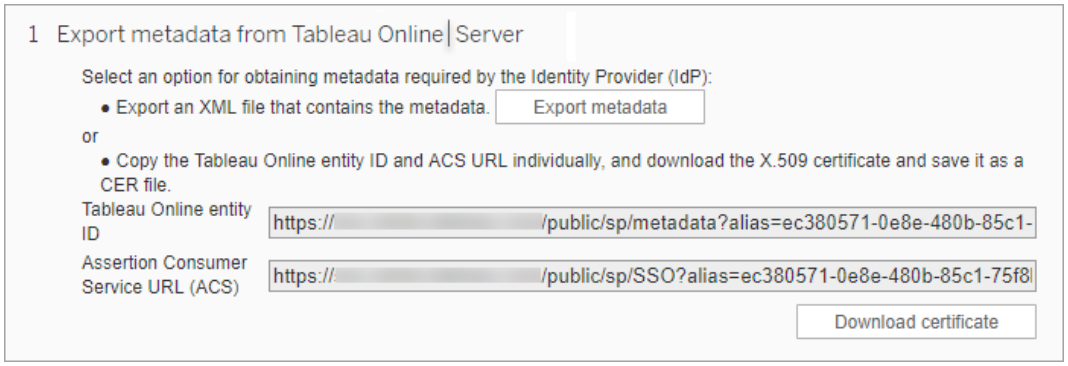
  エンティティ ID はサイト固有であり、サーバー上でサイト SAML を有効化したときに指定したサーバー全体のエンティティ ID に基づいています。例えば、`https://tableau_server` を指定した場合、サイトのエンティティ ID が以下のように表示される可能性があります。

  ```
  https://tableau_
  server/samlservice/public/sp/metadata?alias=48957410-9396-430a-
  967c-75bdb6e002a0
  ```

  Tableau が生成するサイト固有のエンティティ ID または ACS URL を変更することはできません。

- IdP が必要な情報を別の方法で求めている場合、**[証明書のダウンロード]** を選択します。たとえば、Tableau Server エンティティ ID、ACS URL、および X.509 証明書を別の場所で入力する場合です。

次の図は、これらの設定が Tableau Cloud と Tableau Server で同じであることを表示するため編集しています。



ステップ 2 とステップ 3: 外部ステップ

ステップ 2 では、ステップ 1 でエクスポートしたメタデータをインポートするために、IdP アカウントにサインインし、IdP のドキュメンテーションが提供している手順を使って Tableau Server メタデータを提出します。

ステップ 3 では、IdP の文書はメタデータをサービスプロバイダーに対して提供する方法も示しています。メタデータファイルをダウンロードするように指示するか、XML コードが表示されます。XML コードが表示される場合は、コードをコピーして新しいテキストファイルに貼り付け、ファイルに .xml 拡張子を付けて保存します。

ステップ 4: Tableau サイトへの IdP メタデータのインポート

Tableau Server の**認証**ページで、IdP からダウンロードしたか、それが提供する XML から手動で構成したメタデータファイルをインポートします。

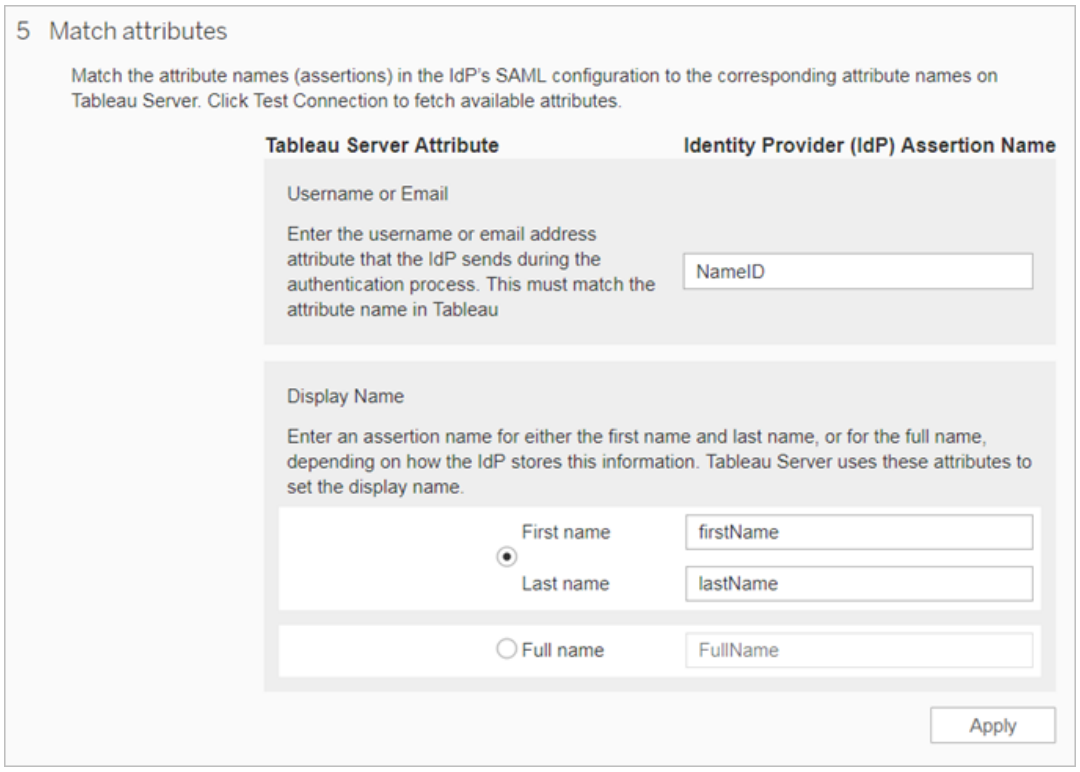
注: 構成を編集する場合は、Tableau が正しい IdP エンティティ ID と SSO サービス URL を使用できるようにメタデータファイルをアップロードする必要があります。

ステップ 5: 属性をマッピング

属性は、認証、認可、およびユーザーに関する他の情報を含みます。**[アイデンティティプロバイダー (IdP) アサーション名]** 列内で、Tableau Server が必要とする情報を含む属性を入力します。

- **ユーザー名 または メール:** (必須) ユーザー名 または メールアドレスを格納する属性の名称を入力します。
- **表示名:** (オプション) IdP の中には、名と姓に別の属性を使用しているものや、フル ネームを 1 つの属性に保存するものがあります。ローカル認証で SAML を使用している場合、表示名の属性は SAML IdP と同期されません。

  IdP が名前を保存する方法に対応するボタンを選択します。たとえば、IdP が名と姓を 1 つの属性で組み合わせている場合は、**[表示名]** を選択して、属性名を入力します。



ステップ 6: ユーザーの管理

既存の Tableau Server ユーザーを選択するか、シングル サインオンを承認する新しいユーザーを追加します。

ユーザーを追加またはインポートするとき、その認証タイプも指定します。ユーザーの認証タイプは、ユーザーの追加後にいつでも [ユーザー] ページで変更できます。

詳細については、サイトへのユーザーの追加、ユーザーのインポートおよびユーザーの認証タイプを SAML 用に設定するを参照してください。

重要: サイト固有の SAML を使用して認証するユーザーは、1 つのサイトにのみ所属できます。ユーザーが複数のサイトにアクセスする必要がある場合、認証タイプをサーバーの既定に設定します。サーバー管理者によってサイト固有の SAML がどのように構成されたかによって、サーバーの既定はローカル認証またはサーバー全体の SAML のいずれかになります。

ステップ 7: トラブルシューティング

[認証] ページで提案されているトラブルシューティングの手順を開始します。それらの手順を実行しても問題が解決しない場合は、SAML のトラブルシューティングを参照してください。

SAML 証明書の更新

SAML 認証を構成した後は、証明書を定期的に更新する必要があります。場合によっては、IT 環境の運用上の変更に合わせて証明書の変更が必要になることもあります。どちらの場合も、TSM または [サイト認証] ページを使用して、設定済みの SAML 証明書を更新する必要があります。

以下は、サーバー全体の SAML 実装およびサイト固有の SAML 実装において、証明書とキー ファイルを更新する手順です。

サーバー全体の SAML の証明書を更新する

サーバー全体の SAML の証明書と、必要に応じて対応するキー ファイルを変更または更新するには、次の手順に従います。

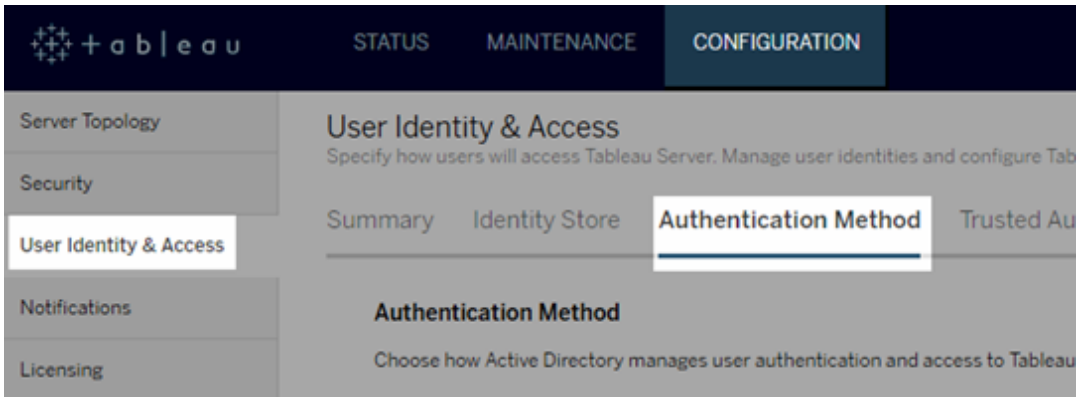
1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. Tableau Server を停止します。

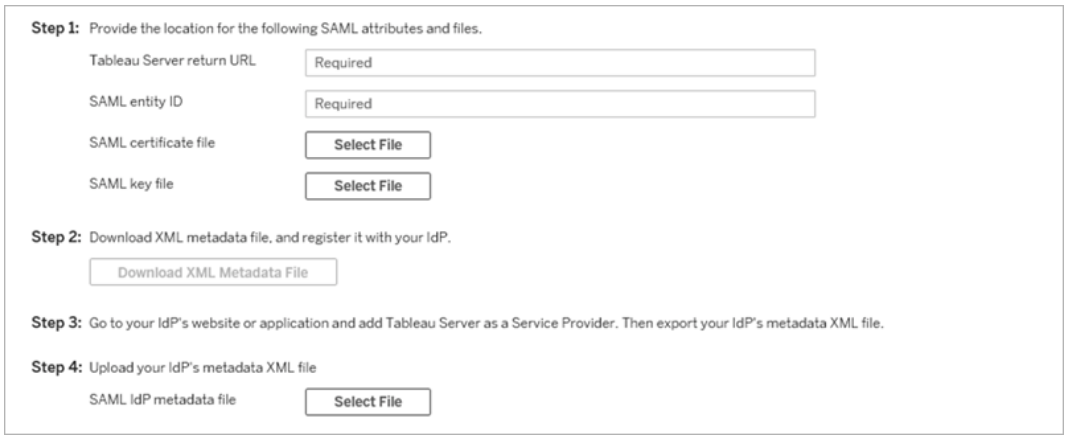
   TSM Web UI から、**[Tableau Server が実行中]** をクリックして **[Tableau Server の停止]** を選択します。あるいは、コマンドラインから`tsm stop` コマンドを実行します。

3. **[構成]** タブで **[ユーザー ID とアクセス]** を選択してから **[認証方法]** タブを選択します。



4. **[認証方法]** で、**[SAML]** を選択します。

5. ステップ 1 からステップ 4 を GUI で実行して SAML 証明書ファイルを更新し、Tableau Server と IdP 間でメタデータを交換します。



6. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

7. ページ上部の **[変更を保留中]** をクリックします。

8. **[変更を適用して再起動]** をクリックします。

証明書を変更した後、`tsm pending-changes apply` を実行して、Tableau Server サービスを再起動する必要があります。この SSL 証明書を使用するコンピューター上の他のサービスも再起動することをお勧めします。オペレーティング システムでルート証明書を変更する場合は、コンピュータを再起動する必要があります。

サイト固有の SAML 証明書の更新

Tableau サイトのメタデータに使用する証明書は Tableau から提供され、構成を変更できません。サイト固有の SAML 証明書を更新するには、IdP に新しい証明書をアップロードし、メタデータを Tableau Server と交換し直す必要があります。

1. サーバーまたはサイト管理者としてサイトにサインインし、**[設定]**、**[認証]** の順に選択します。

2. [認証の種類] で **[接続の編集]** を選択して展開します。

3. 新しいタブまたは新しいウィンドウを開き、IdP アカウントにサインインします。

4. IdP のドキュメントに記載されている手順に従って、新しい SAML 証明書をアップロードします。
5. 新しい XML メタデータファイルをダウンロードして、Tableau Server に提供します。
6. Tableau Server の **[認証]** ページに戻り、UI のステップ 4 で、IdP からダウンロードしたメタデータファイルをインポートします。
7. **[適用]** ボタンをクリックします。

SAML のトラブルシューティング

このトピックでは、SAML 認証を設定する際に発生する可能性のある問題の解決に関する情報を示します。

SAML、および自動ログインの有効化

SAML を使用していて、Tableau Server がユーザー管理に Active Directory を使用するよう構成されている場合、**[自動ログインを有効にする]** を選択しないでください。**[自動ログインを有効にする]** とSAML の両方を同じサーバー インストールで使用することはできません。

SAML の構成時に "HTTP Status 500" エラーが発生する

環境によっては、HTTP ステータス 500 エラーが発生し、SAML を有効にしてブラウザーで Tableau Server URL へ移動した後に、次のエラーが表示される場合があります。

```
org.opensaml.saml2.metadata.provider.MetadataProviderException: User
specified binding is not supported by the Identity Provider using
profile urn:oasis:names:tc:SAML:2.0:profiles:SSO:browser
```

このエラーを解決するには、次の内容を確認してください。

- SAML タブで指定した SSO プロファイルの IdP URL が正しい。
- IdP のサービスプロバイダーを作成中に入力した SSO プロファイルの IdP URL が正しい。

- IdP は HTTP-POST 要求を使用するよう構成されます。(リダイレクトおよび SOAP はサポートされません。)

これらの設定のいずれかが間違っている場合、適切な更新を行ってからもう一度 SAML 構成ステップを実行します (Tableau Server から XML メタデータドキュメントを生成し、エクスポートするところから始める)。

これらの設定が正しいにもかかわらずエラーが表示される場合は、SAML 要件に記載されているとおり、メタデータ XML が Tableau Server および IdP によって作成されていることを確認します。

コマンドラインからのサインイン

Tableau Server で SAML を使用するよう構成されていても、tabcmd や Tableau データ抽出コマンドライン ユーティリティ (Tableau Desktop に付属) を使用して Tableau Server にサインインした場合、SAML は認証に使用されません。これらのツールは、Tableau Server が最初にインストールされたときに (ローカル認証または AD のいずれかで) 認証を構成しておく必要があります。

ログインの失敗: ユーザーが見つかりませんでした

次のメッセージが表示されてログインに失敗します。

```
>Login failure: Identity Provider authentication successful for
user <username from IdP>. Failed to find the user in Tableau
Server.
```

このエラーは一般的に Tableau Server に保存されているユーザー名と IdP が提供したユーザー名が一致しないことを意味します。これを修正するには、これらのユーザー名が一致していることを確認してください。たとえば、山田太郎のユーザー名が tyamada として IdP に保存されている場合、Tableau Server でも tyamada として保存する必要があります。

ログインの失敗: SSL オフロード

次のメッセージが表示されてログインに失敗します。

```
Unable to Sign In - Invalid username or password.
```

さらに、vizportal ログ (debug モードに設定) に次のメッセージが含まれています。

```
DEBUG com.tableau.core.util.RemoteIP - Found header null in X-
FORWARDED-PROTO
```

> **注:** SAML 関連のイベントを記録するには、`vizportal.log.level` を `debug` に設定する必要があります。詳細については、ロギング レベルの変更を参照してください。

このメッセージの組み合わせは、Tableau Server への接続用に SSL をオフロードする外部プロキシ サーバーの構成ミスを示しています。この問題を解決するには、KB 記事「アップグレード後の SAML エラー "Unable to Sign In" および "Invalid username or password" (英語)」を参照してください。

SAML エラー ログ

SAML 認証は Tableau Server の外で行われるため、認証問題のトラブルシューティングは難しい場合があります。ただし、ログインの試みは Tableau Server によって記録されます。ログ ファイルのスナップショットを作成して、問題のトラブルシューティングに使用できます。詳細については、ログ ファイルのスナップショット (ログのアーカイブ) を参照してください。

> **注:** SAML 関連のイベントを記録するには、`vizportal.log.level` を `debug` に設定する必要があります。詳細については、ロギング レベルの変更を参照してください。

解凍されたログ ファイル スナップショットにある次のファイルで、SAML エラーを確認します。
`\vizportal\vizportal-<n>.log`

アプリケーション プロセス (vizportal.exe) が認証を処理するため、SAML 応答はそのプロセスによってログに記録されます。

末尾のスラッシュ

[SAML] タブで、**Tableau Server リターン URL** が末尾のスラッシュで終わっていないことを確認します。

正解: **http://tableau_server**

間違い: http://tableau_server/

接続を確認

構成しているTableau Server にルーティング可能な IP アドレスまたは NAT のいずれかがファイアウォールにあり、サーバーとの直接的な双方向トラフィックを許可していることを確認します。

テルネットをTableau Server で実行して、SAML IdP への接続を試みることによって、接続をテストすることができます。例: `C:\telnet 12.360.325.10 80`

上のテストでは、IdP の HTTP ポート(80) に接続され、HTTP ヘッダーを受け取ります。

複数の定義域

[SAML] タブで、定義域を空白にして、Tableau Server **定義域**属性が、その定義域をSAML アサーションの `domain\username` 形式で検出することを確認します。

正: <empty>

誤: yourdomain.com

Kerberos

Kerberos は、キー配布センター(KDC) と呼ばれる信頼できるサードパーティの使用に依存した3要素認証プロトコルで、コンピューターの識別子を確認し、チケットの交換を通してコンピューター間の安全な接続を提供しています。これらのチケットによりコンピュータ間またはサービス間の相互認証を行い、一方に他方へアクセスするパーミッションがあることを確認します。

Tableau Server は Active Directory の Kerberos 環境で Kerberos 認証をサポートし、Kerberos が Tableau Server への認証を処理します。

注:

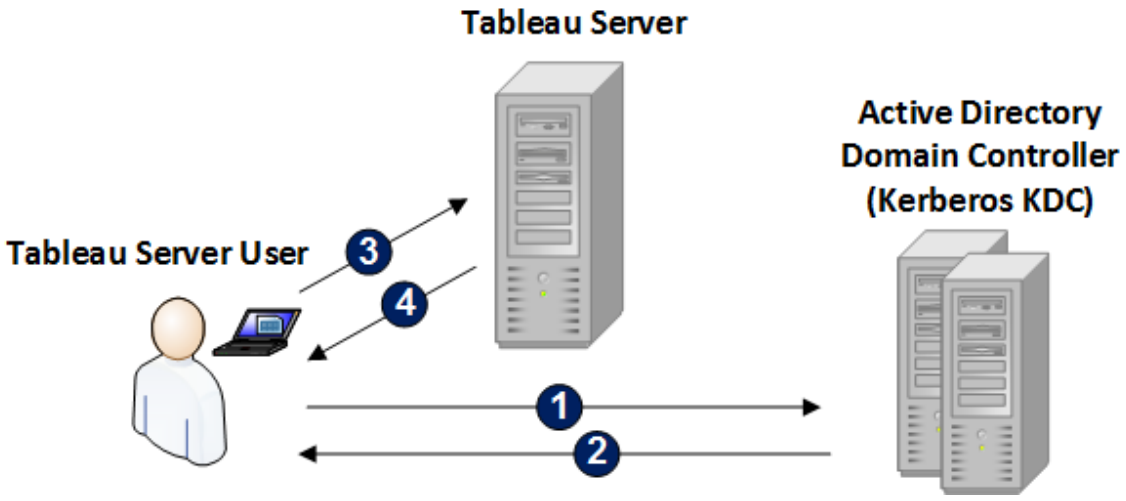
- Tableau Server の Kerberos サポートは、ユーザー認証を行うためのものです。ワークブックなどTableau Server コンテンツに関係する内部許可や認可は処理しません。
- アイデンティティプールは、組織で必要となる追加のユーザープロビジョニングと認証オプションを補完およびサポートするように設計されたツールで、OpenID Connect (OIDC) 認証の

みをサポートします。詳細については、「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。

Kerberos のしくみ

Active Directory (AD) 環境において Tableau Server で Kerberos を構成する場合、AD ドメインコントローラーも Kerberos キー配布センター (KDC) として機能し、そのドメインで他のノードにチケット保証チケットを発行します。KDC により認証されたユーザーは、Tableau Server に接続する際に、さらに認証を行う必要はありません。

次に、認証ワークフロー図を示します。



1. ユーザーは Active Directory ドメインにログインします。
2. Kerberos KDC はユーザーを認証し、チケット保証チケット (TGT) をユーザーのコンピューターに送信します。
3. ユーザーは Tableau Desktop または Web ブラウザーで Tableau Server に接続します。
4. Tableau Server がユーザーを認証します。

Kerberos の要件

Tableau Server で使用する Kerberos の認証は、Active Directory 環境で構成することができます。

一般的な要件

- 外部ロードバランサー/プロキシ サーバー: Tableau Server で Kerberos を使用する場合に、その環境に外部ロードバランサー (ELB) またはプロキシ サーバーがある場合は Tableau Server 構成ユーティリティでの Kerberos 構成の前に、これらを設定する必要があります。Tableau Server のプロキシとロードバランサーの設定を参照してください。

- iOS ブラウザーのサポート: iOS ユーザーは、ユーザーの Kerberos ID を指定する構成プロファイルがインストールされている場合、Mobile Safari で Kerberos 認証を使用することができます。Tableau Mobile ヘルプの「Kerberos サポート用の iOS デバイスの構成」を参照してください。Kerberos SSO のブラウザー サポートの詳細については、Kerberos SSO に対する Tableau クライアントのサポートを参照してください。

- Tableau Server では、データソースへの認証のための制約付きの委任がサポートされています。このシナリオでは、ターゲット データベースの SPN へのアクセス権が Tableau データ アクセス アカウントに専用で付与されています。制約のない委任はサポートされません。

- サポートされているデータ ソース (SQL Server、MSAS、PostgreSQL、Hive/Impala、および Teradata) で Kerberos 認証を構成する必要があります。

- keytab ファイルは、ユーザー認証に使用する Tableau Server のサービス プロバイダー名で構成されます。詳細については、「Keytab 要件の理解」を参照してください。

- Tableau Server 2021.2.25、2021.3.24、2021.4.19、2022.1.15、2022.3.7、および 2023.1.3 (またはそれ以降) では、keytab ファイルが AES-128 暗号または AES-256 暗号を使用して作成されていることを確認してください。RC4 暗号と3DES 暗号はサポートされなくなりました。詳細については、Tableau ナレッジ ベースの「Tableau Server が自分を自動的に認証できない」を参照してください。

Active Directory の要件

Active Directory 環境にある Tableau Server で Kerberos を使用するには次の要件を満たす必要があります。

- Tableau Server では認証に Active Directory (AD) を使用する必要があります。
- ドメインは、Tableau Server に対する Kerberos 接続の AD 2003 以降のドメインである必要があります。
- スマートカードのサポート: スマートカードは、ユーザーがスマートカードでワークステーションにサインインし、それによりユーザーに Kerberos TGT が Active Directory から付与される場合にサポートされます。
- シングル サインオン (SSO): ユーザーがコンピューターにサインインする際、ユーザーに Active Directory からの Kerberos チケット保証チケット (TGT) を付与する必要があります。この動作はドメインで結合した Windows コンピューター、およびネットワーク アカウント サーバーとして AD を使用する Mac コンピューターで標準的です。Mac コンピューターと Active Directory の使用方法の詳細については、Apple ナレッジ ベースの「ネットワーク アカウント サーバに Mac を接続する」を参照してください。

Kerberos 委任

Kerberos 委任シナリオでは、次が必要です。

- ドメインが AD 2003 以降の場合、シングル ドメイン Kerberos 委任がサポートされます。ユーザー、Tableau Server、およびバックエンド データベースは同じドメイン上に存在している必要があります。
- ドメインが AD 2008 の場合は、クロス ドメイン サポートに限定されます。次の条件が満たされている場合は、他のドメインのユーザーに委託できます。Tableau Server とバックエンド データベースが同じドメイン上に存在している必要があり、Tableau Server が存在するドメインとユーザーのドメインとの間に双方向の信頼関係が必要です。

- ドメインが 2012 以降の場合、フル クロスドメイン委任がサポートされます。AD 2012 R2 以外は手動で構成する必要がありますが、AD 2012 R2 には制約付きの委任に対する構成のダイアログがあるため、AD 2012 R2 が推奨されます。

Keytab 要件の理解

Kerberos 認証はキータブファイルと呼ばれる特別に書式設定されたファイルに保存されている認証資格情報に依存します。Tableau Server 展開向けにキータブファイルを生成する必要がある場合があります。このトピックでは、Tableau Server が一般的な組織内での様々なサービスにアクセスする際に使用するキータブファイルについて説明しています。Tableau Server を次のサービスに統合するには、キータブを生成しなければならない場合があります。

- Windows Active Directory でのユーザー認証 (SSO)
- データ ソースの委任
- オペレーティング システム
- ディレクトリサービス

重要: Tableau Server 2021.2.25、2021.3.24、2021.4.19、2022.1.15、2022.3.7、および 2023.1.3 (またはそれ以降) では、keytab ファイルが AES-128 暗号または AES-256 暗号を使用して作成されていることを確認してください。RC4 暗号と3DES 暗号はサポートされなくなりました。詳細については、Tableau ナレッジ ベースの「Tableau Server が自分を自動的に認証できない」を参照してください。

組織内に Id、認証、および/またはセキュリティを扱うIT 専門家がいる場合は、彼らと共に Tableau Server 展開のための適切なキータブの生成計画を作成するべきです。

Windows Active Directory でのユーザー認証 (SSO)

Active Directory を Tableau Server のアイデンティティストアとして使用し、ユーザーに Kerberos SSO で認証することを求める場合は、Tableau Server 用のキータブファイルを生成する必要があります。

Tableau が...で実行されている。	**キータブを手動生成する必要がありますか?**
Active Directory ドメイン内の Windows	はい
Active Directory ドメイン内の Linux	はい

非 Active Directory 環境内の Windows または Linux	Kerberos SSO はサポートされているシナリオではありません。

以下の推奨に従ってください(Windows および Linux バージョンの Tableau Server 向け) :

- サービス アカウントを Tableau Server のディレクトリに作成する。
- Tableau Server アカウント向けにキータブ ファイルを作成します。PC アカウント/OS が認証に使用するキータブファイルを再利用しないでください。同じ Kerberos SSO 向けキータブを、上記シナリオでディレクトリ認証に対して使用するのと同じように使用できます。
- サービスプリンシパル名 (SPN) を Tableau Server サービスの Active Directory に作成する必要があります。
- 次のセクションのバッチファイルを使用して SPN およびキータブファイルを作成します。
- SPN の作成後、キータブファイルを Kerberos の構成に記載されているようにアップロードします。

バッチファイル: SPN を設定してキータブを Active Directory に作成します。

バッチファイルを使用してサービスプリンシパル名 (SPN) を設定しキータブファイルを作成できます。これらの操作は、Active Directory で実行されている (Windows または Linux 上の) Tableau Server 向け Kerberos SSO を有効にするためのプロセスの一部です。

以前のバージョンの Tableau Server(2018.2 以前) 、構成スクリプトは Tableau Server 構成ユーティリティから生成されました。

構成スクリプトを生成するには、以下のバッチファイル コンテンツをコピーしてテキストファイルに貼り付けます。バッチファイルは Tableau Server のサービスプリンシパル名 (SPN) を作成して、ファイルに指定したユーザー向けのキータブファイルを作成します。

ファイル内容の指示に従います。ファイルのカスタマイズ完了後、.bat ファイルとして保存します。

このファイルは定義域管理者によりActive Directory 定義域で実行される必要があります。定義域管理者はファイルで指定したアカウントのサービス アカウント パスワードをプロンプト表示で通知されます。

このバッチ ファイルでは、Windows の set、setspn、および ktpass コマンドを使用します。

> **注**: 以下のバッチ ファイルは自己文書化されたものです。ただし、Kerberos の使用やキータブ ファイルの生成の経験がない場合は、続行する前に Microsoft ブログ投稿「All you need to know about Keytab files」を読むことをお勧めします。組織内の環境設定によっては、ktpass コマンドの追加の構成が必要になる場合があります。たとえば、`/crypto` パラメーターの設定を決定する必要があります。KDC で必要とされる単一の `/crypto` 値を指定することをお勧めします。`/crypto` パラメーターでサポートされている値の完全なリストについては、Microsoft 記事「ktpass」を参照してください。

Active Directory でユーザー認証用のキータブ ファイルを作成する際は、ここで指定した Windows コンピューターで実行する必要があります。Linux コンピューターでのこのキータブ ファイルの作成はサポートされていません。

SPN およびキータブ バッチファイル コンテンツ

Tableau Server 2022.3、2022.1.8、2021.4.12、2021.3.17、2021.2.18、2021.1.20、2020.4.23 以降

```
@echo off
setlocal EnableDelayedExpansion

REM *******

REM This script generates the Service Principal Names (SPNs) and
keytab files required for
REM Kerberos SSO with Apache.
REM This script executes set, setspn, and ktpass commands included
in any Windows Server
```

```
REM Operating System from 2003 on.
REM Before running this script you must enter configuration
information for the setspn and
REM ktpass commands.
REM Elements that require your configuration information are
enclosed in as such:
REM  ! -- and --!.
REM After you customize this file, save it as a .bat file, and run
on a domain-joined
REM computer.
REM This script must be run by a Domain admin.

REM **********

REM The following set command will prompt the domain admin for
credentials of the
REM Tableau Server service account.
REM This account must be a valid domain user account.
REM If the password contains a literal \" (blackslash - double
quote), all backslashes
REM immediately before the double quote must be
REM duplicated when typed for the password to work, e.g. if password
contains
REM  \" replace with \\", if passwords contains \\" replace with
\\\\"

set /p adpass= "Enter password for the Tableau Server service
account."
set adpass=!adpass:"=\"!

REM **********

REM The following setspn commands create the SPN in the domain.
REM More information on setspn can be found here:
REM http://technet.microsoft.com/en-us/library/cc731241(WS.10).aspx
REM Enter the canonical FQDN and the host names for Tableau Server
```

```
followed by the
REM Tableau Server service account name.
REM Use this syntax: HTTP/hostname domain\service_account_name.
REM The example below shows syntax for a computer named "tableau01"
in the "example.lan"
REM domain, with service account, "tab-serv-account":
REM setspn -s HTTP/tableau01 example\tab-serv-account
REM setspn -s HTTP/tableau01.example.lan example\tab-serv-account
REM DNS and AD are not case sensitive, but the keytab files are.
Verify that host names
REM match letter case as stored in DNS.
REM Use Windows Server's DNS Manager utility to verify host name
case.

REM **********

echo Creating SPNs...
setspn -s HTTP/!--replace with canonical host name and service
account --!
setspn -s HTTP/!--replace with canonical FQDN and service account -
-!

REM **********

REM The following commands create the keytab file in the same
directory where the
REM bat file is run. More information on ktpass can be found here:
REM https://docs.microsoft.com/en-us/windows-
server/administration/windows-commands/ktpass
REM Note: keytab files are case-sensitive.
REM The realm following the FQDN should be all uppercase.
REM Syntax is:
REM ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass
!adpass!
REM /pttype KRB5_NTPRINCIPAL /crypto !--cipher--! /out
keytabs\kerberos.keytab
```

```
REM Best practice: specify the /crypto value that is required by
your KDC.
REM Options for /crypto = {DES-CBC-CRC|DES-CBC-MD5|AES256-
SHA1|AES128-SHA1|All}
REM Do not specify /crypto All because it will result in a keytab
that contains ciphers that are not supported
REM and cause errors.
REM When using AES256-SHA1 OR AES128-SHA1, the /mapuser option must
be included
REM in the ktpass command to ensure the keytab file is mapped
properly to the user. For example:
REM ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass
!adpass! /ptype KRB5_NT_PRINCIPAL /mapuser <domain\username> /crypto
AES256-SHA1 /out keytabs\kerberos.keytab
REM The following example shows the ktpass syntax with the
example.lan configuration from above:
REM ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass
!adpass! /ptype KRB5_NT_PRINCIPAL /crypto DES-CBC-CRC /out
keytabs\kerberos.keytab

REM **********

echo Creating Keytab files in %CD%\keytabs
mkdir keytabs
ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass !adpass!
/ptype KRB5_NT_PRINCIPAL /crypto DES-CBC-CRC /out
keytabs\kerberos.keytab
```

Tableau Server の以前のバージョンの場合

```
@echo off
setlocal EnableDelayedExpansion

REM *******

REM This script generates the Service Principal Names (SPNs) and
```

```
keytab files required for
REM Kerberos SSO with Apache.
REM This script executes set, setspn, and ktpass commands included
in any Windows Server
REM Operating System from 2003 on.
REM Before running this script you must enter configuration
information for the setspn and
REM ktpass commands.
REM Elements that require your configuration information are
enclosed in as such:
REM  ! -- and --!.
REM After you customize this file, save it as a .bat file, and run
on a domain-joined
REM computer.
REM This script must be run by a Domain admin.

REM **********

REM The following set command will prompt the domain admin for
credentials of the
REM Tableau Server service account.
REM This account must be a valid domain user account.
REM If the password contains a literal \" (blackslash - double
quote), all backslashes
REM immediately before the double quote must be
REM duplicated when typed for the password to work, e.g. if
password contains
REM  \" replace with \\", if passwords contains \\" replace with
\\\\"

set /p adpass= "Enter password for the Tableau Server service
account."
set adpass=!adpass:"=\"!

REM **********
```

```
REM The following setspn commands create the SPN in the domain.
REM More information on setspn can be found here:
REM http://technet.microsoft.com/en-us/library/cc731241(WS.10).aspx
REM Enter the canonical FQDN and the host names for Tableau Server
followed by the
REM Tableau Server service account name.
REM Use this syntax: HTTP/hostname domain\service_account_name.
REM The example below shows syntax for a computer named "tableau01"
in the "example.lan"
REM domain, with service account, "tab-serv-account":
REM setspn -s HTTP/tableau01 example\tab-serv-account
REM setspn -s HTTP/tableau01.example.lan example\tab-serv-account
REM DNS and AD are not case sensitive, but the keytab files are.
Verify that host names
REM match letter case as stored in DNS.
REM Use Windows Server's DNS Manager utility to verify host name
case.

REM **********

echo Creating SPNs...
setspn -s HTTP/!--replace with canonical host name and service
account --!
setspn -s HTTP/!--replace with canonical FQDN and service account -
-!

REM **********

REM The following commands create the keytab file in the same
directory where the
REM bat file is run. More information on ktpass can be found here:
REM https://docs.microsoft.com/en-us/windows-
server/administration/windows-commands/ktpass
REM Note: keytab files are case-sensitive.
REM The realm following the FQDN should be all uppercase.
REM Syntax is:
```

```
REM ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass
!adpass!
REM /pttype KRB5_NTPRINCIPAL /crypto !--cipher--! /out
keytabs\kerberos.keytab
REM Best practice: specify the /crypto value that is required by
your KDC.
REM Options for /crypto = {DES-CBC-CRC|DES-CBC-MD5|RC4-HMAC-
NT|AES256-SHA1|AES128-SHA1|All}
REM Specifying /crypto All will result in passwords stored with RC4
cipher, which is
REM no longer considered secure.
REM When using AES256-SHA1 OR AES128-SHA1, the /mapuser option must
be included
REM in the ktpass command to ensure the keytab file is mapped
properly to the user. For example:
REM ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass
!adpass! /ptype KRB5_NT_PRINCIPAL /mapuser <domain\username>
/crypto AES256-SHA1 /out keytabs\kerberos.keytab
REM The following example shows the ktpass syntax with the
example.lan configuration from above:
REM ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass
!adpass! /ptype KRB5_NT_PRINCIPAL /crypto DES-CBC-CRC /out
keytabs\kerberos.keytab

REM **********

echo Creating Keytab files in %CD%\keytabs
mkdir keytabs
ktpass /princ HTTP/!--FQDN--!@!--Kerberos_Realm--! /pass !adpass!
/ptype KRB5_NT_PRINCIPAL /crypto DES-CBC-CRC /out
keytabs\kerberos.keytab
```

オペレーティング システム

所属する組織が Kerberos を認証に使用している場合は、Tableau Server が実行されている PC はそれが実行されている Kerberos のアドレス体系で認証される必要があります。

Tableau が...で実行されている。	**キータブを手動生成する必要がありますか?**
Active Directory ドメイン内の Windows	いいえ
Active Directory ドメイン内の Linux	はい
非 Active Directory 環境内の Windows または Linux	はい

Tableau Server on Windows を実行していて、PC が Active Directory に参加している場合は、オペレーティング システム用のキータブ ファイルを管理または生成する必要はありません。

Tableau Server を Linux の Kerberos アドレス体系 (MIT KDC または Active Directory) で実行している場合は、その PC のオペレーティングシステム専用のキータブファイルを生成する必要があります。PC 向けに作成したキータブは OS 認証専用です。OS 認証用と同じキータブファイルを、本トピックの後半で説明するその他サービスに使用しないでください。

ディレクトリサービス

所属する組織が LDAP または Active Directory などのディレクトリサービスを使用して、ユーザー アイデンティティを管理している場合、Tableau Server はそのディレクトリへの読み取り専用アクセスが必要です。

あるいは、Tableau Server をローカルアイデンティティストアとともにインストールしてすべてのアカウントを管理するように構成できます。この場合、キータブは不要です。

以下の表はキータブ要件をまとめています。

Tableau が...で実行されている。	**ディレクトリサービス**	**キータブを手動生成する必要がありますか?**
AD 定義域での Windows	Active Directory	いいえ
Windows	LDAP (GSSAPI バインド)	はい
Linux	Active Directory または LDAP	はい

	(GSSAPI バインド)	
Windows または Linux	Active Directory または LDAP (シンプル バインド)	いいえ
Windows または Linux	ローカル アイデンティティ ストア	キータブは必要ありません。

このシナリオでキータブを手動生成する必要がある場合は、ディレクトリへの GSSAPI バインド用に使用します。以下の推奨に従ってください。

- サービス アカウントを Tableau Server のディレクトリに作成する。
- Tableau Server アカウント向けにキータブ ファイルを作成します。PC アカウント/OS が認証に使用するキータブファイルを再利用しないでください。
- キータブファイルを Tableau Server アイデンティティ ストアの json 構成の一部としてアップロードします。identityStore エンティティを参照してください。

災害復旧計画の一環として、Tableau Server 外部の安全な場所にキータブと conf ファイルのバックアップを保存することをお勧めします。Tableau Server に追加するキータブと conf ファイルは、クライアント ファイル サービスによって他のノードに格納および配布されます。ただし、ファイルは復元可能な形式では格納されません。Tableau Server クライアント ファイル サービスを参照してください。

データソース委任

Kerberos 委任を使用して Active Directory にあるデータソースにアクセスすることも可能です。このシナリオでは、ユーザーは任意のサポートされている認証機構(SAML、ローカル認証、Kerberos など) で Tableau Server に認証されますが、Kerberos により有効化されたデータソースにアクセスできます。

以下の推奨に従ってください。

- Tableau Server (Windows または Linux) のコンピューター アカウントは、Active Directory ドメイン内にある必要があります。

- Kerberos 委任に使用するキータブ ファイルは、Kerberos ユーザー認証 (SSO) に使用する同じキータブであることが可能です。
- キータブは Active Directory での Kerberos 委任のサービス プリンシパルにマップされる必要があります。
- 同じキータブを複数のデータソースに作成できます。

詳細については、次の構成トピックを参照してください。

- Linux での Tableau Server: Kerberos 委任の有効化
- Windows での Tableau Server: Kerberos 委任の有効化

Kerberos の構成

Kerberos を使用するように Tableau Server を構成することができます。これにより、組織のすべてのアプリケーションでシングル サインオン (SSO) が実現します。Tableau Server で Kerberos を構成する前に、お使いの環境が Kerberos の要件を満たしていることを確認してください。

> **注:** Tableau Server への SSO の Kerberos の制約付き委任はサポートされていません。(データソースの制約付き委任はサポートされています)。詳細については、Kerberos の要件の「シングルサインオン (SSO)」を参照してください。

Kerberos を構成するには、最初に Kerberos を有効にしてからユーザー認証用の keytab ファイルを指定する必要があります。指定する keytab ファイルは、Tableau Server のユーザー認証に使用するサービス プロバイダー名で構成する必要があります。データ ソースの認証に Kerberos を使用する場合、これらの認証資格情報を Tableau Server の Kerberos 構成時に指定する単一の keytab ファイルに含める必要があります。

災害復旧計画の一環として、Tableau Server 外部の安全な場所にキータブ ファイルのバックアップを保存することをお勧めします。Tableau Server に追加するキータブ ファイルは、クライアント ファイル サービスによって他のノードに格納および配布されます。ただし、ファイルは復元可能な形式では格納されません。Tableau Server クライアント ファイル サービスを参照してください。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

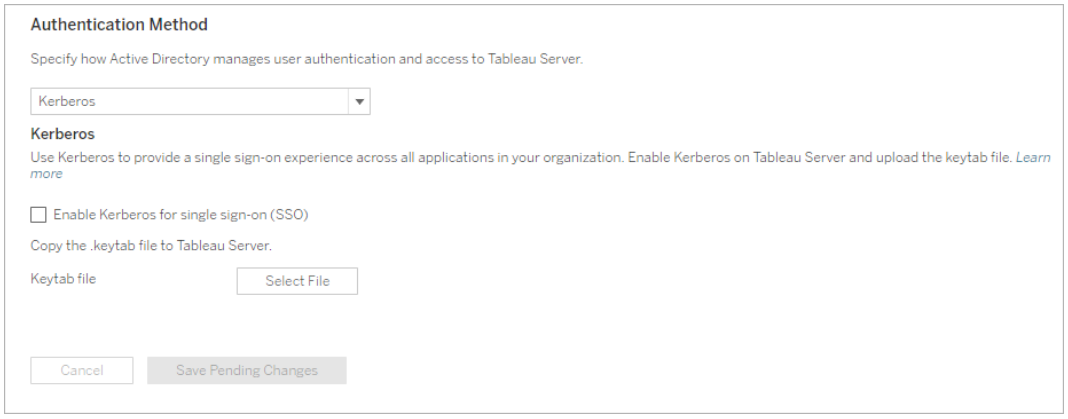
   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[ユーザー アイデンティティとアクセス]** をクリックし、次に **[認証方法]** をクリックします。

3. **[認証方法]** で、ドロップダウン メニューから **[Kerberos]** を選択します。

4. [Kerberos] で、**[シングル サインオン (SSO) 用に Kerberos を有効化]** を選択します。

5. keytab ファイルをサーバーにコピーするには、**[ファイルの選択]** をクリックしてからコンピューター上のファイルを参照します。



6. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

7. ページ上部の **[変更を保留中]** をクリックします。

8. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

1. keytab ファイルを Tableau Server を実行しているコンピューターにコピーし、次のコマンドを実行してファイルにパーミッションを設定します。

   ```
   chmod 644 "/path/keytab_file"
   ```

   分散クラスタを展開する環境で Tableau Server を実行している場合は、keytab ファイルを手動で各ノードに分散し、パーミッションを設定する必要があります。クラスタ内の各ノードで同じディレクトリに keytab ファイルをコピーします。各ノードに keytab ファイルをコピーし、ファイルにパーミッションを設定した後、次の TSM コマンドを 1 つのノードで実行します。構成は各ノードに反映されます。

2. 次のコマンドを入力して keytab ファイルの場所と名前を指定します。

   ```
   tsm authentication kerberos configure --keytab-file <path-to-
   keytab_file>
   ```

3. 次のコマンドを入力して Kerberos を有効にします。

   ```
   tsm authentication kerberos enable
   ```

4. `tsm pending-changes apply` を実行して変更を適用します。

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

SSO 構成の確認

Tableau Server を再起動したら、URL ウィンドウで Tableau Server 名を入力することで、異なるコンピューターの Web ブラウザーから Kerberos 構成をテストします。

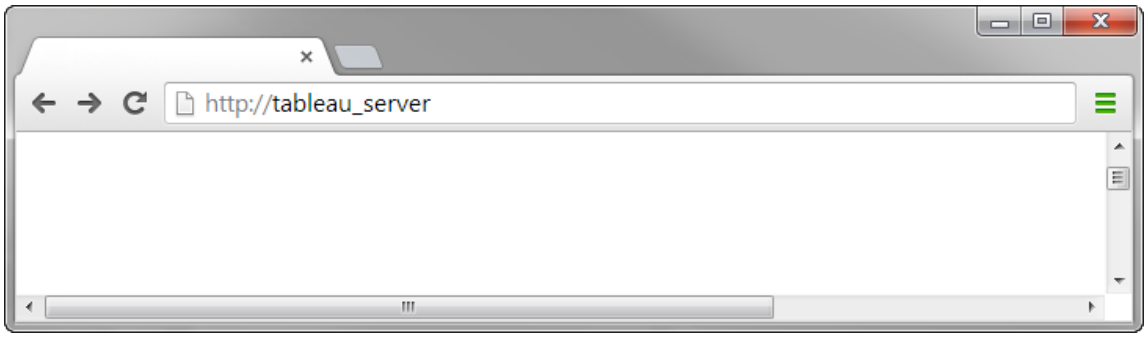


Tableau Server を自動的に認証することをお勧めします。

Kerberos SSO に対する Tableau クライアントのサポート

この記事では、Tableau Server で Kerberos シングル サインオン (SSO) を使用するための要件と特定の Tableau クライアントおよびオペレーティング システムによる微妙な違いについて説明します。ここで対象となる Tableau クライアントには、一般的な Web ブラウザー、Tableau Desktop、Tableau Mobile アプリなどが含まれます。

一般的なブラウザー クライアントのサポート

ブラウザー ベースの Kerberos シングル サインオン (SSO) を使用するには、以下の条件を満たす必要があります:

- Tableau Server 上で Kerberos を有効にする必要があります。
- ユーザーはユーザー名とパスワードを指定して Tableau Server にサインインします。

  注: Kerberos SSO に失敗した場合、フォールバックが設定されていれば、ユーザー名とパスワードの認証資格情報にフォールバックすることができます。

- ユーザーは、クライアント コンピューターまたはモバイル デバイス上で Kerberos による Active Directory への認証を受ける必要があります。具体的には、ユーザーが Kerberos Ticket Granting Ticket (TGT) を付与されている必要があります。

Tableau Desktop およびブラウザーのクライアント

Windows または Mac で、Kerberos SSO を使用して Tableau Desktop の次のバージョンまたはブラウザーから Tableau Server にサインインします。追加構成が必要となる場合もあります。

Windows

- Tableau Desktop 10.3 以降がサポート対象となります。
- Internet Explorer - サポート対象 - 構成が必要となる場合があります。注 1 を参照してください。
- Chrome - サポート対象 - 構成が必要となる場合があります。注 1 を参照してください。
- Firefox - 構成が必要です。注 2 を参照してください。
- Safari - サポート外

Mac OS X

- Tableau Desktop 10.3 以降
- Safari - サポート対象
- Chrome - 注 3 を参照
- Firefox - 注 2 を参照
- Internet Explorer - サポート外

Tableau Mobile アプリクライアント

iOS または Android デバイスでは、次の Tableau Mobile またはモバイル ブラウザーのバージョンを使用して、Tableau Server への Kerberos 認証を行うことができます。

iOS

- Tableau Mobile アプリ- 注 4 を参照してください。
- Safari - 注 4 を参照
- Chrome - サポート外

Android - 注 5 を参照

- Tableau Mobile アプリ
- Chrome

オペレーティング システムとブラウザーに固有の注記

次の注記では、固有のオペレーティング システムおよびクライアントの組み合わせでの構成に関する要件および問題について説明します。

注 1: Windows デスクトップでの Internet Explorer または Chrome

Kerberos SSO は Internet Explorer とChrome のどちらでもサポートされていますが、**Windows の [インターネット オプション]** での設定が必要です。

1. **[統合 Windows 認証]** を有効にします。
2. Tableau Server URL がローカルのイントラネットゾーン内にあることを検証します。

   Internet Explorer でイントラネットゾーンを検出し、この設定を構成できる場合があります。Tableau Server URL が検出および構成されなかった場合は、URL をローカル イントラネットゾーンに手動で追加する必要があります。

統合 Windows 認証を有効にするには:

1. Windows の [コントロール パネル] で、**[インターネット オプション]** を開きます。
2. **[詳細]** タブで、下にスクロールして **[セキュリティ]** セクションに移動します。
3. **[統合 Windows 認証を使用する]** を選択します。
4. **[適用]** をクリックします。

Tableau Server URL を検証またはローカル イントラネットゾーンに追加するには:

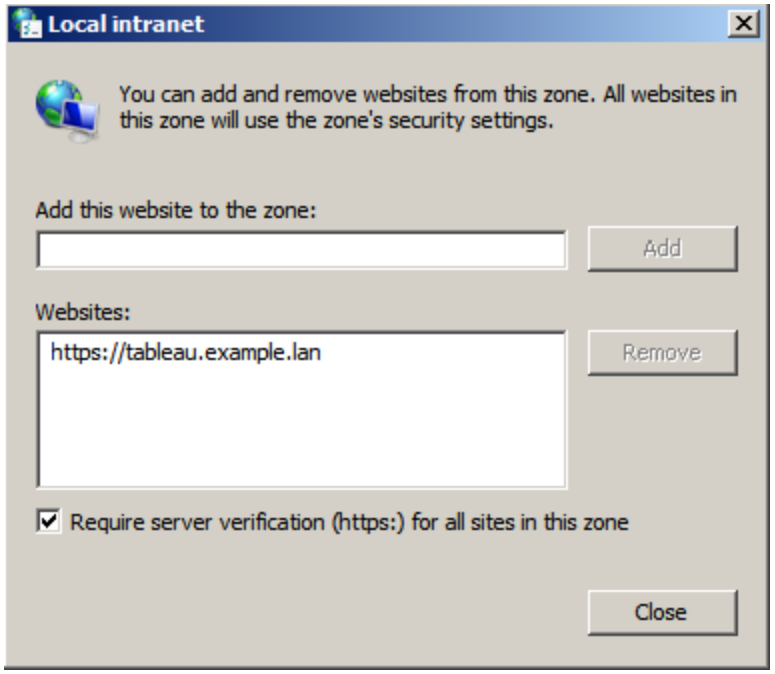
1. Windows の [コントロール パネル] で、**[インターネット オプション]** を開きます。
2. **[セキュリティ]** タブで **[ローカル イントラネット]** を選択し、**[サイト]** をクリックします。
3. **[ローカル イントラネット]** ダイアログ ボックスで、**[詳細設定]** をクリックします。

   [Web サイト] フィールドで、内部 Tableau Server URL を検索します。

   一部の組織では、IT 管理者がワイルドカード (*) を使用して内部 URL を指定します。たとえば、次の URL には、ローカル イントラネットゾーンの内部 `example.lan` 名前空間のすべてのサーバーが含まれます。

`https://*.example.lan`

次のイメージは、`https://tableau.example.lan` の特定の URL を示します。



4. **[Web サイト]** フィールドで Tableau Server URL またはワイルドカード URL が指定されていない場合、**[この Web サイトをゾーンに追加する]** フィールドに Tableau Server 名 (URL) を入力してから、**[追加]**、**[OK]** の順にクリックします。

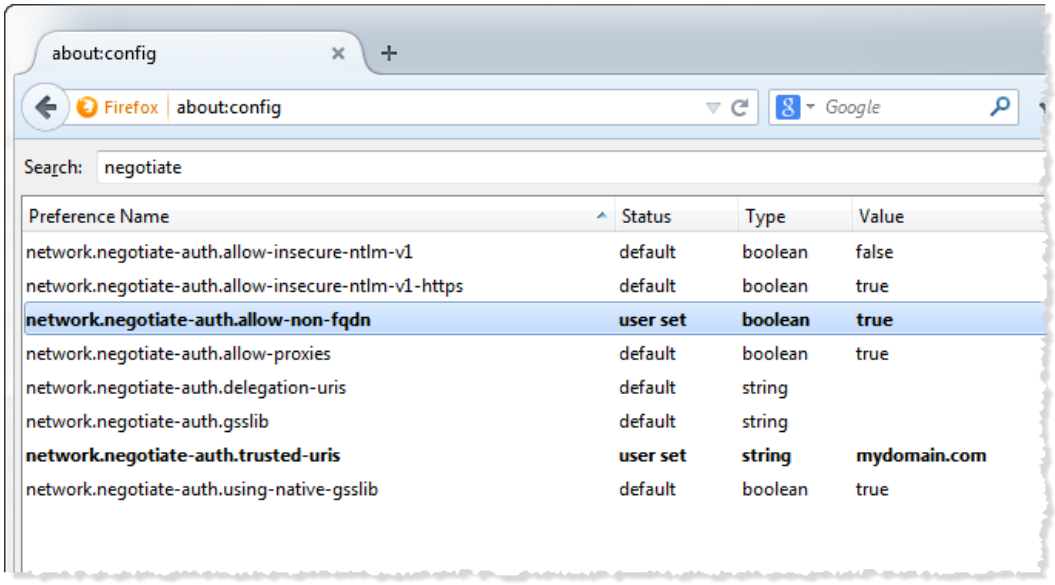
   Tableau Server URL がすでに **[Web サイト]** に記載されている場合は、ダイアログ ボックスを閉じます。

注 2: Windows または Mac OS X デスクトップでの Firefox

Windows または Mac のいずれかの Firefox で、Kerberos SSO を使用して Tableau Server にサインインできます。これには、次の手順を完了して Firefox を設定し、Kerberos をサポートする必要があります。

1. Firefox では、アドレス バーに「`about:config`」と入力します。
2. 詳細設定の変更に関する警告が出たら、**[I'll be careful. I promise!]** をクリックします。

3. **[検索]** ボックスに`negotiate`と入力します。



4. **network.negotiate-auth.allow-non-fqdn** をダブルクリックしてから値を **true** に設定します。

5. **network.negotiate-auth.trusted-urls** をダブルクリックし、Tableau Server の完全修飾ドメイン名 (FQDN) またはドメインを入力します。たとえば、`tableau.example.com` と入力します。

注 3: Mac OS X デスクトップでの Chrome

Chrome マニュアルによれば、ターミナル ウィンドウから次のコマンドで Chrome を起動すると、Mac で Kerberos SSO を使用できます。

```
open -a "Google Chrome.app" --args --auth-server-
whitelist="tableauserver.example.com"
```

ここで、`tableauserver.example.com` は現在の環境における Tableau Server の URL です。

ただし、当社によるテストで一貫性のない結果が検出されました。したがって、Mac で Kerberos SSO を使用する場合は、Safari または Firefox を使用することをお勧めします。詳細については、

The Chromium Projects サイトの「HTTP authentication」にある「*Integrated Authentication*」セクションを参照してください。

> **注**: ユーザーは引き続き Mac OS X で Chrome を使用して Tableau Server にサインインできますが、ログイン時にユーザー名とパスワードを入力するよう求められる場合があります (シングル サインオンは無効)。

注 4: iOS で使用する Mobile Safari または Tableau Mobile

Kerberos を使用するように iOS を構成すれば、Kerberos SSO はサポートされます。iOS デバイスに Kerberos 認証用の構成プロファイルがインストールされていなければなりません。通常、これはエンタープライズ IT グループによって行われます。Tableau サポートは、iOS デバイスを Kerberos 用に設定する作業を支援できません。*Tableau Mobile 導入ガイド*の認証トピックを参照してください。

注 5: Android プラットフォーム

Android オペレーティング システムで使用する Tableau Mobile アプリでは、Kerberos SSO はサポートされていません。SSO でユーザー名とパスワードを認証できない場合に備えて Kerberos にフォールバックが設定されている場合は、Android デバイスと Tableau Mobile アプリまたはサポートされているモバイル ブラウザーを使用して Tableau Server に引き続き接続できます。このシナリオでは、Kerberos を使用して認証するのではなく、Tableau Server にアクセスするときに認証資格情報を入力するよう求められます。

詳細

- *Tableau モバイル導入ガイド*: Tableau Mobile への認証およびアクセスを制御する
- Tableau Server 技術仕様で *Web ブラウザー*を参照してください。

Kerberos のトラブルシューティング

このトピックでのトラブルシューティングのヒントは、サーバーでのシングル サインオン (SSO) に関連する問題と、委任されたデータ ソースの問題に分かれています。

Tableau コミュニティの Wiki ページ「Linux でのデータベース Kerberos の設定のテスト」も参照してください。

Tableau Server へのシングル サインオン

Kerberos SSO 環境では、Web ブラウザーまたは Tableau Desktop から Tableau Server にサインインするユーザーに、Tableau Server ではシングル サインオンを使用して自動的にサインインできないことを示すメッセージが表示される可能性があります。代わりに、Tableau Server のユーザー名とパスワードを指定するよう提案されます。

クライアント コンピューターでのサインイン エラーに関するトラブルシューティング

- **ユーザー名とパスワードを入力する** - Tableau Server へのユーザーの一般的なアクセスを確認するには、ユーザー名とパスワードを入力してサインインします。

  これらの認証資格情報にエラーがある場合、Tableau Server のユーザーでない可能性があります。Kerberos SSO が機能するには、ユーザーが Tableau Server にアクセスできる必

要があります。また、Active Directory によってチケット保証チケット(TGT) が付与される必要もあります (このリストで後述される **TGT** 項目で説明)。

- **その他のユーザーの SSO 認証資格情報を確認する -** 別のユーザー アカウントを使用して、SSO による Tableau Server への接続を試行します。すべてのユーザーが影響を受ける場合、Kerberos 構成に問題がある可能性があります。

- **サーバー コンピューター以外のコンピューターを使用する -** localhost で Tableau Server にサインインする場合、Kerberos SSO が機能しない可能性があります。クライアントは、Tableau Server コンピューター以外のコンピューターから接続する必要があります。

- **IP アドレスではなくサーバー名を使用する -** Tableau Server 名に IP アドレスを入力すると、Kerberos SSO が機能しません。また、にアクセスするために使用するサーバー名は、Kerberos を構成するときに使用した名前と一致する必要があります (以下の「キー テーブル エントリ」を参照してください)。

- **クライアントに TGT があることを確認する -** クライアント コンピューターには Active Directory ドメインによる TGT (チケット保証チケット) が含まれている必要があります。プロキシがチケットを付与する、制約付きの委任はサポートされません。

  TGT がクライアント コンピューターに含まれていることを確認するには、次を実行します。

  - Windows の場合は、コマンド プロンプトを開いて次のように入力します。`klist tgt`
  - Mac の場合は、ターミナル ウィンドウを開いて次のように入力します。`klist`

  アウトプットには、Tableau Server を認証しようとするユーザー/ドメインの TGT が表示される必要があります。

  次の場合、クライアント コンピューターには TGT が含まれていない可能性があります。

- クライアント コンピューターが VPN 接続を使用している。
- クライアント コンピューターがドメインに結合されていない (たとえば、職場で使用している非作業用コンピューター)。
- ユーザーがローカル (非ドメイン) アカウントでコンピューターにサインインした。
- コンピューターが、Active Directory をネットーワーク アカウント サーバーとして使用していない Mac である。

- **ブラウザーのバージョンと設定を確認する** - Web ブラウザーのサインイン時には、Kerberos でサポートされており、必要に応じて適切に設定されていることを確認します。
  - Internet Explorer (IE) とChrome は、Windows ですぐに使用できます。
  - Safari は Mac ですぐに使用できます。
  - Firefox は追加の設定が必要です。

  詳細については、Kerberos SSO に対する Tableau クライアントのサポートを参照してください。

サーバーでのサインイン エラーに関するトラブルシューティング

クライアント コンピューターから問題を解決することができない場合、次の手順として Tableau Server を実行しているコンピューターでトラブルシューティングを行います。管理者は、要求 ID を使用して Tableau Server の Apache ログでサインインの試みを検索することができます。

- **ログ ファイル** - Apache の error.log でサインインに失敗した正確な時間/日付が記されたエラーを確認します。
- **キー テーブル エントリ** - error.log エントリに「キー テーブル エントリが HTTP/<servername>.<domain>.<org>@ と一致しません。」というメッセージが表示される場合。以下に例を示します。

```
[Fri Oct 24 10:58:46.087683 2014] [:error] [pid 2104:tid 4776]
[client 10.10.1.62:56789] gss_acquire_cred() failed:
```

```
Unspecified GSS failure. Minor code may provide more
information (, No key table entry found matching
HTTP/servername.domain.com@)
```

このエラーは、次のいずれかの不一致の結果発生したものです。

- **Tableau Server URL** - クライアントコンピューターがサーバーにアクセスするために使用するURL

  これは、Tableau Desktop またはブラウザーのアドレスバーに入力する名前です。ショートネーム (`http://servername`) または完全修飾ドメイン名 (`http://servername.domain.com`) のいずれかです。

- サーバーIP アドレスの **DNS リバースルックアップ (DNS 逆引き)**

  これは、IP アドレスを使用してDNS 名を検索します。

  コマンドプロンプトで、次のように入力します。

  ```
  ping servername
  ```

  そして、次のように入力して、サーバーに ping して返されたIP アドレスでDNS 逆引きを実行します。

  ```
  nslookup <ip address>
  ```

nslookup コマンドが、IP アドレスのネットワーク情報を返します。応答の *Non-authoritative answer* 部分で、完全修飾ドメイン名 (FQDN) が次の構成値と一致することを確認します。

- Kerberos の .keytab ファイル
- サーバーのサービスプリンシパル名 (SPN)

これらの値の構成に関する詳細については、Keytab 要件の理解を参照してください。

Kerberos 構成スクリプトの確認

環境変数の keytab ファイルの生成に使用した ktpass コマンドの変更が必要になる場合があります。ナレッジ ベース記事のトラブルシューティング手順、「Tableau Server で Kerberos スクリプト構成を生成できない」を確認してください。

データソース SSO

委任されたデータソース アクセスの失敗

vizqlserver ログ ファイルで "ワークグループ認証モード" を確認します。

ログ ファイルで "ワークグループ認証モード" を見つけます。"現状のまま" ではなく"kerberos 偽装" と表示されています。。

Kerberos 委任マルチドメイン構成

Tableau Server は、他の Active Directory ドメインからユーザーを委任できます。データベースが MIT Kerberos を使用している場合、データベース ユーザーのマッピングに合わせて Kerberos プリンシパルを調節する必要が生じる場合があります。具体的には、ユーザーが接続する各 Kerberos 領域の規則を使用して、krb5.conf を更新する必要があります。`[realms]` セクションの `auth_to_local` タグを使用して、マップ プリンシパル名をローカル ユーザー名にマッピングします。

たとえば、ユーザー EXAMPLE\jsmith について検討します。このユーザーの Kerberos プリンシパルは jsmith@EXAMPLE.LAN です。この場合、Tableau Server は委任ユーザーを jsmith@EXAMPLE に指定します。Tableau Server は Active Directory レガシー ドメインの別名を Kerberos Realm として使用します。

ターゲット データベースには、次のように、"jsmith@EXAMPLE.LAN" をデータベース ユーザー "jsmith" にマッピングするなどのルールが既にある場合があります。

```
 EXAMPLE.LAN = {
   RULE:[1:$1@$0](.*@EXAMPLE.LAN)s/@.*//
   DEFAULT
 }
```

委任をサポートするため、jsmith@EXAMPLE をデータベース ユーザーにマッピングする他のルールを追加する必要があります。

```
EXAMPLE.LAN = {
   RULE:[1:$1@$0](.*@EXAMPLE.LAN)s/@.*//
   RULE:[1:$1@$0](.*@EXAMPLE)s/@.*//
   DEFAULT
}
```

詳細については、MIT Kerberos ドキュメントのトピック、「krb5.conf」を参照してください。

クロス ドメインの制限付き委任

KDC が Windows 2012 より前の Windows サーバーで KDC が実行されているクロス ドメイン シナリオでは、委任が失敗する場合があります。発生する可能性のエラーには以下が含まれます。

- SQL サーバー ネットワーク インタフェース: システムはドメイン コントローラーに認証要求に対処するための連絡が取れません。少し後に再試行してください。
- SQL サーバー ネイティブ クライアント: SSPI コンテクストを生成できません。
- ドメイン コントローラーは、次を返します。KRB-ERR-POLICY error with a status STATUS_CROSSREALM_DELEGATION_FAILURE (0xc000040b).

「クロスドメイン」とは、Tableau Server がデータソースとは異なるドメインで、別のサービス アカウントにより実行されているシナリオを指します。例:

- Tableau Server は DomainA 上で DomainA サービス アカウントにより実行されています。
- SQL Server は DomainB 上で DomainB サービス アカウントにより実行されています。

従来の制限された委任は、サーバー両方が同じドメインにある場合のみ機能しました。ユーザーは他のドメインからでも問題ありません。

上記のエラーが表示される場合、このシナリオを有効化するため、Active Directory 管理者に委任するアカウントに構成された従来型の制限された委任を削除してもらってください。委任の削除は Active Directory 管理ツール、または Active Directory プロパティ `msDS-AllowedToDelegateTo.` に関連する値をすべて削除することで実行できます。

既存のシングル ドメイン委任をクロスドメイン委任と共に保持する場合は、双方をリソース ベースの制限された委任を使用して構成しなくてはなりません。

Kerberos と制限された委任の詳細については、Microsoft のトピック「Kerberos Constrained Delegation Overview」を参照してください。

Web 作成

Kerberos 委任をサポートしない Web 作成シナリオは 2 つあります。「Web 上のデータへ接続する」機能と「Web 上でデータベースを作成する」機能はまだ委任をサポートしていません。具体的には、Web 作成で Kerberos を使用するデータソースを作成する場合、データソースは実行サービス アカウントの認証を使用します。Kerberos 委任を使用してデータソースを作成する場合は、Tableau Desktop でパブリッシュする必要があります。実行サービス アカウントの詳細については、「Kerberos サービス アカウントへのアクセスの有効化」を参照してください。

相互 SSL 認証の構成

相互 SSL を使用して、Tableau Desktop、Tableau Mobile、その他の承認済み Tableau クライアントのユーザーに Tableau Server への安全な直接アクセスを提供することができます。相互 SSL を使用すれば、有効な SSL 証明書を持つクライアントが Tableau Server に接続するときに、Tableau Server でクライアント証明書の存在を確認し、クライアント証明書内のユーザー名

に基づいてユーザーを認証します。クライアントに有効な SSL 証明書がない場合、Tableau Server は接続を拒否することができます。

相互 SSL に失敗する場合、ユーザー名/パスワード認証を使用するようTableau Server を構成することもできます。さらに、フォールバック認証が構成されているかどうかに関係なく、REST API でユーザー名とパスワード(存在する場合) を使用してログインできます。

ユーザー認証セッションの時間制限

ユーザーが相互 SSL でログインすると、認証セッションは、Tableau Server のグローバル認証セッションの構成を制御するのと同じ方法で管理されます。

Web ブラウザーを使用して Tableau Server に接続するクライアントの場合、グローバル認証セッションの構成は、*セキュリティ強化チェックリスト*に記載されています。「9.セッション存続時間設定の検証を参照してください。

接続されたクライアント(Tableau Desktop、Tableau Mobile、Tableau Prep Builder、Bridge) のセッションでは、OAuth トークンを使用して、セッションを再確立することによりユーザーをログインしたままにします。既定では、OAuth クライアントトークンは 1 年経過後にリセットされます。クライアントトークンが 14 日間使用されなかった場合には有効期限切れになります。これらの値は、`refresh_token.absolute_expiry_in_seconds` および `refresh_token.idle_expiry_in_seconds` オプションの設定により変更できます。tsm configuration set のオプションを参照してください。

証明書の使用

相互 SSL を有効にして構成する前に、外部 SSL を構成する必要があります。外部 SSL は、Tableau Server をクライアントが認証するものであり、外部 SSL を構成するための証明書とキーを使用してセッションを暗号化します。

相互 SSL では、証明書ファイルがもう1 つ必要です。ファイルは CA 証明書ファイルを連結したものです。ファイルの種類は `.crt`.である必要があります。"CA" は、Tableau Server に接続するクライアントコンピューターに証明書を発行する*認証局*です。CA 証明書ファイルをアップロードするアクションによって信頼が確立されます。これにより、Tableau Server はクライアントコンピューターによって提示される個々の証明書を認証できます。

災害復旧計画の一環として、証明書ファイルと失効ファイル (該当する場合) のバックアップを Tableau Server 外部の安全な場所に保管することをお勧めします。Tableau Server に追加する証明書ファイルと失効ファイルは、クライアント ファイル サービスによって他のノードに格納および配布されます。ただし、ファイルは復元可能な形式では格納されません。Tableau Server クライアント ファイル サービスを参照してください。

RSA キーのサイズと ECDSA 曲線のサイズ

相互 SSL に使用される CA 証明書では、RSA キーの強度が 2048 であるか、ECDSA 曲線サイズが 256 である必要があります。

それぞれの構成キーを設定することによって、安全性の低いサイズを受け入れるように Tableau Server を構成できます。

- ssl.client_certificate_login.min_allowed.rsa_key_size
- ssl.client_certificate_login.min_allowed.elliptic_curve_size

tsm configuration set のオプションを参照してください。

クライアント証明書要件

相互 SSL を使用して Tableau Server に認証するユーザーは、セキュリティの最小要件を満たすクライアント証明書を提示する必要があります。

署名アルゴリズム

証明書には SHA-256 またはそれ以上の署名アルゴリズムを使用する必要があります。

SSL 相互認証用に構成された Tableau Server では、ユーザーが SHA-1 署名アルゴリズムを使用したクライアント証明書を提示すると、認証が拒否されます。

ユーザーが SHA-1 クライアント証明書を使用してログインしようとすると「サインインできません」というエラーが表示され、VizPortal ログには次のエラーが記録されます。

```
Unsupported client certificate signature detected: [certificate
Signature Algorithm name]
```

ssl.client_certificate_login.blocklisted_signature_algorithms 構成オプションを設定することにより、安全性の低い SHA-1 署名アルゴリズムを受け入れるように Tableau Server を構成できます。

RSA キーのサイズと ECDSA 曲線のサイズ

相互 SSL に使用される クライアント証明書では、RSA キーの強度が 2048 であるか、ECDSA 曲線サイズが 256 である必要があります。

これらの要件を満たしていないクライアント証明書から相互認証が要求されると、Tableau Server の認証は失敗します。それぞれの構成キーを設定することにより、安全性の低いサイズを受け入れるように Tableau Server を構成できます。

- ssl.client_certificate_login.min_allowed.rsa_key_size
- ssl.client_certificate_login.min_allowed.elliptic_curve_size

tsm configuration set のオプションを参照してください。

TSM Web インターフェイスの使用

1. Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する。
2. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。
3. **[構成]** タブで **[ユーザー ID とアクセス] > [認証方法]** を選択します。
4. **[認証方法]** のドロップダウン メニューで **[相互 SSL]** を選択します。
5. [相互 SSL] で **[クライアント証明書による相互 SSL と自動サインインを使用]** を選択します。
6. **[ファイルの選択]** をクリックし、認証局 (CA) の証明書ファイルをサーバーにアップロードします。

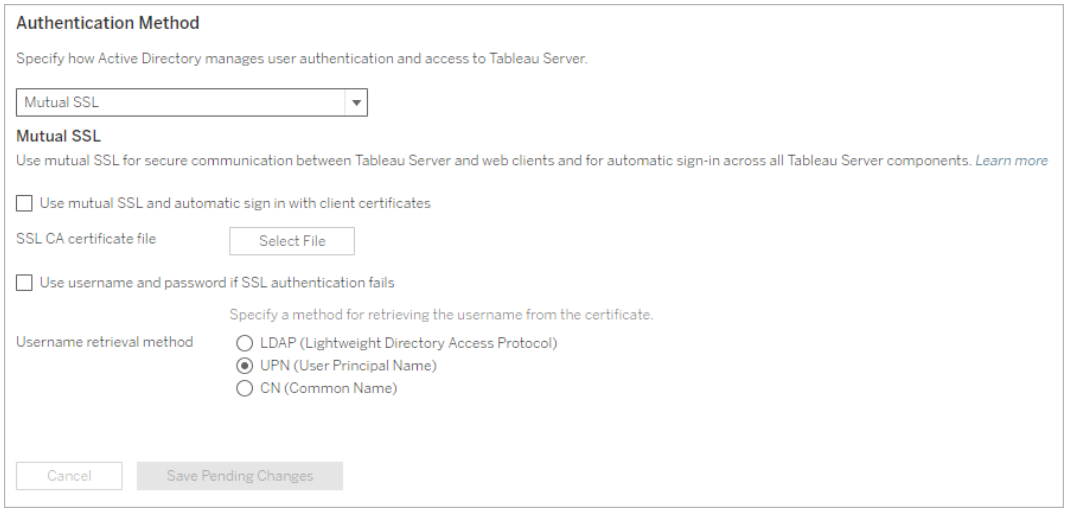
このファイル (.crt) は、クライアント認証に使用する複数の CA 証明書を含んでいるオールインワンのファイルです。アップロードするファイルは、PEM でエンコードされたさまざまな証明書ファイルを優先順に連結したものである必要があります。

7. 自分の組織の残りの SSL 構成情報を入力します。

   ユーザー名の形式: Tableau Server で相互 SSL を構成する場合、サーバーはクライアント証明書からユーザー名を取得し、クライアントユーザーの直接サインインを確立できるようにします。Tableau Server が使用する名前は、Tableau Server がユーザー認証用に設定される方法によって異なります。

   - ローカル認証 — Tableau Server は証明書から UPN (ユーザープリンシパル名) を使用します。
   - **Active Directory (AD)**—Tableau Server は LDAP (ライトウェイトディレクトリアクセスプロトコル) を使用してユーザー名を取得します。

   または、Tableau Server を設定してクライアント証明書から CN (共通名) を使用できます。



8. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

9. ページ上部の **[変更を保留中]** をクリックします。

10. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

ステップ 1: 外部サーバー通信に SSL が必要

Tableau Server とWeb クライアントの間の外部通信に SSL を使用するようTableau Server を構成するには、サーバー証明書の .crt および .key ファイルの名前を指定して、次のように `external-ssl enable` コマンドを実行します。

```
tsm security external-ssl enable --cert-file <file.crt> --key-
file <file.key>
```

- `--cert-file` および `--key-file` には、CA が発行したサーバーの SSL 証明書 (.crt) およびキー (.key) ファイルの保存場所とファイル名を指定します。
- 上記のコマンドは、ユーザーが Tableau Server 上で**サーバー管理者**サイトロールを持つユーザーとしてサインインしていることを前提としています。代わりに、`-u` および `-p` パラメーターを使用して、管理者ユーザーおよびパスワードを指定することもできます。
- 証明書キーファイルにパスフレーズが必要な場合は、`--passphrase` パラメーターと値を含めます。

ステップ 2: 相互 SSL を構成して有効にする

サーバーと各クライアント間の相互認証を追加し、Tableau クライアントユーザーが初回に認証資格情報を提供した後は直接認証されるようにします。

1. 次のコマンドを実行します。

   ```
   tsm authentication mutual-ssl configure --ca-cert <certificate-
   file.crt>
   ```

   `--ca-cert` には、認証局 (CA) 証明書ファイルの場所とファイル名を指定します。

このファイル (.crt) は、クライアント認証に使用する複数の CA 証明書を含んでいるオールインワンのファイルです。アップロードするファイルは、PEM でエンコードされたさまざまな証明書ファイルを優先順に連結したものである必要があります。

2. 次のコマンドを実行して相互 SSL を有効化し、変更を適用します。

   ```
   tsm authentication mutual-ssl enable
   ```

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

相互 SSL の追加オプション

`mutual-ssl configure` を使用して、次のオプションをサポートするようTableau Server を構成することができます。

詳細については、「tsm authentication mutual-ssl <commands>」を参照してください。

フォールバック認証

Tableau Server が相互 SSL 向けに構成されている場合、認証は自動となり、クライアントは有効な証明書を持つ必要があります。フォールバックオプションを許可し、ユーザー名およびパスワード認証を受け付けるようTableau Server を構成することができます。

```
tsm authentication mutual-ssl configure -fb true
```

上記のオプションが `false` に設定されている場合でも、Tableau Server は REST API クライアントからのユーザー名とパスワード認証を受け付けます。

ユーザー名マッピング

Tableau Server が相互 SSL 向けに構成されている場合、サーバーはクライアント証明書からユーザー名を取得することにより、ユーザーを直接認証します。Tableau Serverが使用する名前は、サーバーがユーザー認証用に設定される方法によって異なります。

- **ローカル認証 -** 証明書からUPN (ユーザープリンシパル名) を使用します。
- **Active Directory (AD) -** LDAP (ライトウェイトディレクトリアクセスプロトコル) を使用してユーザー名を取得します。

これらの既定のいずれかを上書きし、Tableau Server が共通名を使用するように設定できます。

```
tsm authentication mutual-ssl configure -m cn
```

詳細については、相互認証中にクライアント証明書をユーザーにマッピングするを参照してください。

証明書失効リスト(CRL)

秘密キーの侵害が疑われる場合、または認証機関 (CA) が証明書を適切に発行しなかった場合、CRL の指定が必要になる場合があります。

```
tsm authentication mutual-ssl configure -rf <revoke-file.pem>
```

相互 SSL 認証のしくみ

相互 (または 2 方向) SSL 認証では、暗号化されたデータストリーム、サーバーおよびクライアントの両方の相互認証、および直接アクセスの利便性を組み合わせて提供します。Tableau Server で相互 SSL を使用するには、以下が必要です。

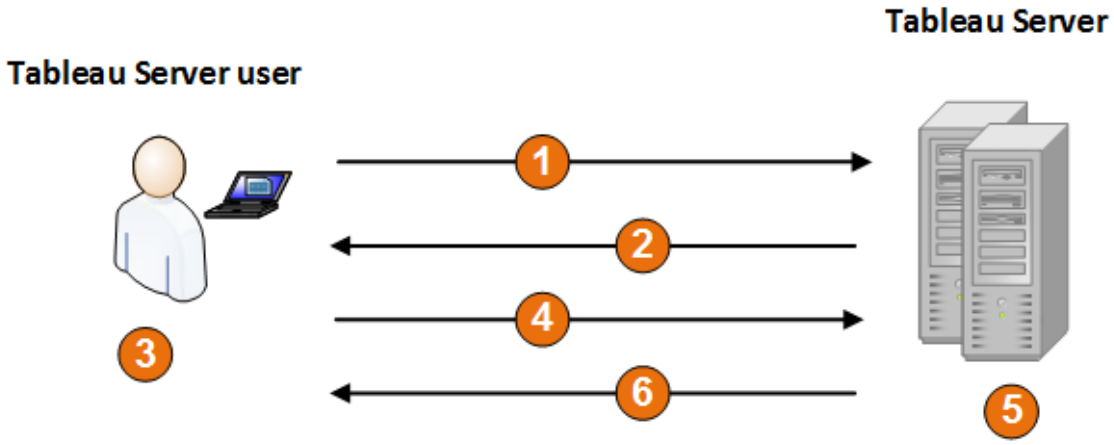
- Tableau Server で構成される外部 SSL
- Tableau Server 用の信頼できるCA により発行されたSSL 証明書。ファイルは CA 証明書ファイルを連結したものです。"CA" は、Tableau Server に接続するクライアントコンピューターに証明書を発行する*認証局*です。CA 証明書ファイルをアップロードするアクションによって信頼が確立されます。これにより、Tableau Server はクライアントコンピューターによって提

示される個々の証明書を認証できます。

- Tableau Server に接続する各クライアントの証明書。
- 相互 SSL を使用するように構成された Tableau Server。

Tableau Server およびクライアントは相互に有効な証明書があることを確認し、Tableau Server はクライアント証明書内のユーザー名に基づいてユーザーを認証します。

次のイメージは、相互 SSL で発生するイベントの流れについてより詳しく示しています。



1. ユーザーは Tableau Server に移動します。
2. Tableau Server が SSL 証明書をクライアント コンピュータに送信します。
3. クライアント コンピュータで Tableau Server 証明書を検証します。
4. クライアント コンピュータにより証明書が Tableau Server に送信されます。
5. Tableau Server がクライアント証明書を認証します。
6. Tableau Server はクライアント証明書内のユーザー名を参照してユーザーを認証します。

相互認証中にクライアント証明書をユーザーにマッピングする

相互 (2 方向) SSL 認証を使用する場合、クライアントは認証プロセスの一環として Tableau Server に証明書を示します。次に、クライアント証明書内のユーザー情報を既知のユーザー ID に

マッピングします。Tableau ServerTableau Server がクライアントのマッピングに使用する方法は、組織のクライアント証明書の内容によって異なります。

このトピックでは、クライアント証明書の情報のユーザーID へのマッピング方法と、Tableau Server でのマッピング実行方法を変更する方法について説明します。マッピングがどのように発生するか、およびそれに対する変更が必要かどうかを理解するには、組織内でクライアント証明書がどのように構造化されているかを知る必要があります。

- ユーザー名マッピングオプション
- 証明書マッピングの変更
- 複数ドメイン組織でユーザー名のあいまいさに対処する

ユーザー名マッピングオプション

Tableau Serverは以下のアプローチの一つをクライアント証明書をユーザーIDにマッピングするため使用します。

- **Active Directory**。ユーザー認証で Active Directory を使用するように Tableau Server が構成された場合、Tableau Server がクライアント証明書を受け取る際に証明書を Active Directory に渡し、その証明書を Active Directory ID にマッピングします。証明書内の明示的なユーザー名情報は無視されます。

  > **注**:この方法では、Active Directory 内のユーザー アカウントに対してパブリッシュされるクライアント証明書が必要です。

- **ユーザープリンシパル名 (UPN)** 。ユーザー名をユーザープリンシパル名 (UPN) フィールドに保管するようにクライアント証明書を構成できます。Tableau Serverは UPN 値を読み取って、Active Directory 内のユーザーまたはローカル ユーザーへマッピングします。

- **Common name (CN)**。ユーザー名を証明書の共通名 (CN) フィールドに保管するようにクライアント証明書を構成できます。Tableau Serverは CN 値を読み取って、Active Directory 内のユーザーまたはローカル ユーザーへマッピングします。

Active Directory認証およびUPNまたはCNユーザー名マッピングをサーバーに構成した場合、ユーザー名を以下の形式の一つで表現してください。

`username`、`domain/username`、または `username@domain`。

例: `jsmith`、`example.org/jsmith`、`jsmith@example.org`

サーバーがローカル認証を使用する場合、UPN または CN フィールドにおける名前の書式はあらかじめ設定されていませんが、フィールド名はサーバーのユーザー名と一致する必要があります。

証明書マッピングの変更

クライアント証明書を Tableau Server のユーザー ID にマッピングするために、tsm authentication mutual-ssl <commands> コマンドを使用します。

```
tsm authentication mutual-ssl configure -m <value>
```

考えられる値は、Active Directory マッピングの場合は `ldap`、UPN マッピングの場合は `upn`、CN マッピングの場合は `cn` です。

Tableau Server を最初にインストールして構成するとき、サーバーは既定のユーザー名マッピングをサーバーの認証タイプに合わせて設定します。

- サーバーが Active Directory を使用するよう構成されされている場合、ユーザーIDに認証をマッピングするためにも Active Directory を使用します。
- サーバーがローカル認証を使用するよう構成されている場合、サーバーは証明書内のUPNフィールドからユーザー名の値を取得します。

Tableau Server がユーザー名を ID にマッピングする方法に関する既定の動作がサーバー構成に対して適切でない場合、次のコマンドセットを実行して CN 値を使用するようマッピングを変更します。

```
tsm authentication mutual-ssl configure -m cn

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

複数ドメイン組織でユーザー名マッピングのあいまいさに対処する

状況によっては、証明書の UPN または CN フィールドのユーザー名があいまいになる可能性があります。このあいまいさにより、ユーザー名をサーバー上のユーザー ID にマッピングしたときに予期しない結果が生じる可能性があります。

例として、Tableau Server がドメインの含まれないユーザー名を取得した場合、サーバーは既定のドメインを使用して、そのユーザー名を ID にマッピングします。これにより、あるユーザーに異なるユーザーのIDとパーミッションを割り当てる可能性があり、不正なユーザー名マッピングを引き起こす可能性があります。

これは、特に次の条件がすべて適用する環境で発生する可能性があります。

- 組織が複数の Active Directory ドメインをサポートしている。
- サーバーが Active Directory 認証を使用するように構成されている。
- サーバーが UPN や CN マッピングを使用するように構成されている。
- 一部のユーザーが同じユーザー名を使っているが、ドメインは異なっている。例えば、`jsmith@example.org` および `jsmith@example.com`。
- 証明書の UPN または CN フィールドにあるユーザー名に、ユーザー名の一部としてドメインが含まれていない (例えば `jsmith` とだけ表示する)。

間違ったユーザー名マッピングを防ぐために、クライアントの証明書に完全修飾ユーザー名とドメインが、`jsmith@example.org` または `example.org/jsmith` 形式で含まれるようにする。

OpenID Connect

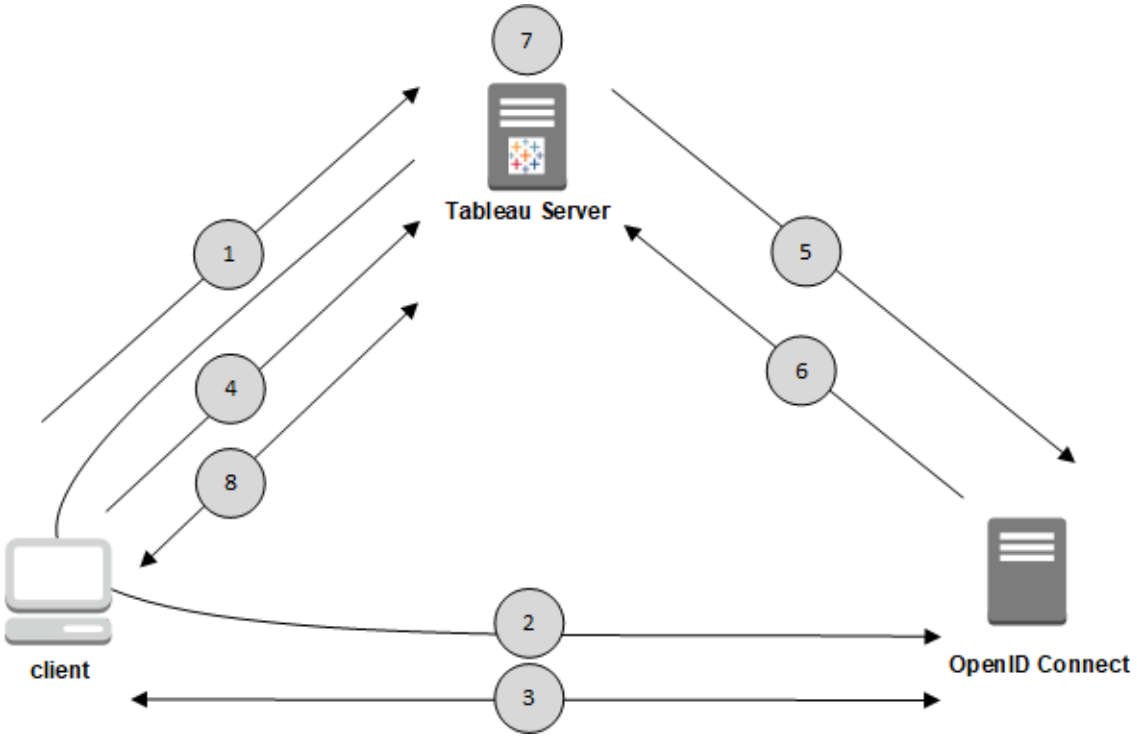
OpenID Connect (OIDC) を使用してシングル サインオン (SSO) を提供するように Tableau Server を構成できます。OIDC は、ユーザーが Google や Salesforce などの ID プロバイダー (IdP) にサインインできるようにする標準認証プロトコルです。ユーザーが IdP に正常にサインインすると、自動的に Tableau Server.にサインインされます。

OIDC の構成には、いくつかのステップがあります。このセクションのトピックでは、OIDC での Tableau Server の使用に関する一般的な情報を提供し、Tableau Server を構成するためのシーケンスについて説明します。

注: 特に明記されていない限り、OIDC 認証に関する情報は、Tableau Server セットアップ中に TSM で構成された OIDC 認証、またはアイデンティティプールを使用して構成された OIDC 認証のどちらにも適用されます。

認証の概要

このセクションでは、Tableau Server での OpenID Connect (OIDC) 認証プロセスについて説明します。



1.ユーザーがクライアントコンピューターからTableau Server にログインを試みます。

2.Tableau Server が認証の要求を IdP ゲートウェイにリダイレクトします。

3.ユーザーに認証資格情報を求めるメッセージが表示され、IdP への認証が正常に完了します。IdP が応答し、Tableau Server に戻るリダイレクトURL を返します。リダイレクトURL には、ユーザーの認可コードが含まれます。

4.クライアントが Tableau Server にリダイレクトされ、認可コードを提示します。

5. Tableau Server がクライアントの認可コードを独自のクライアント認証資格情報と一緒に IdP に提示します。Tableau Server も IdP のクライアントです。この手順は、スプーフィングまたは中間者攻撃を防ぐことを目的としています。

6.IdP は Tableau Server にアクセストークンとID トークンを返します。

- JSON Web Token (JWT) 検証: 既定では、Tableau では IdP JWT の検証が実行されます。検出時には、IdP 構成検出ドキュメントで `jwks_uri` によって指定されている公開キーが取得されます。ID トークンの有効期限が検証された後、JSON Web 署名 (JWS)、発行者 (IdP)、およびクライアントID が検証されます。JWT プロセスの詳細については、OpenID ドキュメントの 10 を参照してください。署名と暗号化 (英語)」および IETF から提案された標準である JSON Web Token を参照してください。お使いの LDP でサポートされていない場合を除き、JWT 検証を有効のままにすることをお勧めします。

- ID トークンは、ユーザー用の属性キーペアのセットです。キーペアは *クレーム*と呼ばれます。ユーザー用の IdP クレームの例:

```
"sub"                          :
"7gYhRR3HiRRCaRcgvY50ubrtjGQBMJW4rXbpPFpg2cptHP62m2sqowM7G1LwjN-
5"
"email"                        : "alice@example.com",
"email_verified"               : true,
"name"                         : "Alice Adams",
"given_name"                   : "Alice",
"family_name"                  : "Adams",
```

```
"phone_number"              : "+359 (99) 100200305",
"profile"                   : "https://tableau.com/users/alice"
```

7. Tableau Server は IdP クレームからユーザーを識別し、ステップ 1 からの認証要求を完了します。Tableau Server は、"sub" (サブジェクト識別子) を一致させることでリポジトリに保存されているユーザーのアカウントレコードを検索し、正しいユーザー アカウントを識別します。sub クレーム値と一緒に保存されているユーザー アカウントがない場合、Tableau Server は、IdP の "email" クレームと一致するリポジトリ内のユーザー名を検索します。ユーザー名の一致が成功すると、Tableau Server は対応する sub クレームをリポジトリ内のユーザーのレコードに保存します。Tableau Server をこのプロセスで別のクレームを使用するように構成することができます。「OpenID Connect の使用要件」「」を参照してください。

8. Tableau Server がユーザーを承認します。

OpenID Connect で Tableau が機能する仕組み

OpenID Connect (OIDC) は、サービスプロバイダー (ここでは Tableau Server) と IdP との間で交換される情報の多くのオプションをサポートする柔軟なプロトコルです。次のリストは、OIDC の Tableau Server 実装の詳細をまとめたものです。これらの詳細は、Tableau Server がどのような種類の情報を送信し、期待するか、および IdP をどのように構成するかを理解するのに役立ちます。

- Tableau Server は、OpenID Connect ドキュメントの「OpenID Connect の最終仕様」(英語) に記載されているように、OpenID 認可コードのフローのみをサポートします。
- Tableau Server は、検出またはプロバイダー URL の使用に依存して OpenID プロバイダー メタデータを取得します。または、静的検出ドキュメントを Tableau Server でホストすることもできます。詳細については、OpenID Connect に対応するように Tableau Server を構成するを参照してください。
- Tableau Server は、`client_secret_basic`、`client_secret_post` クライアント認証、
- Tableau Server は `id_token` 属性の JOSE ヘッダーに `kid` 値を期待します。この値は、JWK セット ドキュメントのいずれかのキーと一致します。JWK セット ドキュメントの URI は、

OpenID 検出ドキュメントの `jwks_uri` 値によって指定されます。JWK セットドキュメントにキーが 1 つしかない場合でも、`kid` 値を示す必要があります。

- Tableau ServerJWK `x5c` パラメーターに対する、および X.509 証明書の使用に対する OpenID サポートを含みます。
- 既定では、Tableau Server はプロキシ設定を無視し、すべての OpenID リクエストを直接 IdP に送信します。

  Tableau Server がフォワードプロキシを使用してインターネットに接続するように設定されている場合は、「OpenID Connect に対応するように Tableau Server を構成する」に記載されている説明に従って、追加の変更を行う必要があります。

OpenID Connect の使用要件

このトピックでは、Tableau Server で OpenID Connect を使用するための要件について説明します。

注: TSM 認証構成コマンドは、Tableau Server のセットアップ中に TSM で構成された OIDC 認証にのみ適用されます。アイデンティティプールの OIDC 認証構成を変更するには、Tableau REST OpenAPI を使用する認証構成の更新エンドポイントを使用します。

要件の概要

- IdP アカウント
- ローカル アイデンティティストア
- IdP クレーム - ユーザーのマッピング
- 認証コンテキスト

IdP アカウント

OpenID Connect (OIDC) プロトコルをサポートするアイデンティティプロバイダー (IdP) に対するアクセス権が必要です。また、IdP のアカウントも必要です。OpenID Connect は多くのアイデンティティプロバイダーがサポートしています。OIDC プロトコルはオープンかつ柔軟な規格であるため、すべて

の環境で同じように実装されるわけではありません。OIDC に対応するようTableau Server を構成する際は、IdP と連携してください。

Google IdP の実装は Tableau Server で十分にテスト済みであり、トピックで説明した構成の IdP モデルになっています。

ローカル アイデンティティストア

Tableau Server で OpenID Connect を使用するには、以下のいずれかが true である必要があります。

- **Tableau Server のセットアップ中に TSM で OIDC を構成する場合**、Tableau Server はローカル アイデンティティストアを使用するように構成されている必要があります。サーバーは、Active Directory などの外部ディレクトリからユーザーをインポートするのではなく、Tableau Server でユーザーを明示的に作成するよう構成する必要があります。OpenID では、外部のアイデンティティストアを使用したユーザーの管理はサポートされていません。
- アイデンティティプール**を使用して OIDC を構成する場合**、OIDC は、1) ローカル アイデンティティストア、または 2) Tableau Server のセットアップ中に TSM で構成されたアイデンティティストアである AD または LDAP を使用して構成することができます。

IdP クレーム - ユーザーのマッピング

Tableau Server に正常にサインインするには、指定されたユーザーを OpenID でプロビジョニングし、Tableau Server のユーザー アカウントにマッピングする必要があります。OpenID は クレームに依存するメソッドを使用して、他のアプリケーションとユーザー アカウント属性を共有します。クレームには、メール、電話番号、名前など、ユーザー アカウント属性が含まれます。Tableau Server が IdP クレームをどのようにユーザー アカウントにマッピングしているかを理解するには、「OpenID Connect」を参照してください。

Tableau Server は IdP クレームに依存して、IdP のユーザーアカウントを.Tableau Server でホストされているユーザーアカウントにマッピングします。既定では、Tableau Server は、IdP から メール クレームが渡されることを期待します。お使いの IdP によっては、別の IdP クレームを使用するよう Tableau Server を構成する必要がある場合もあります。

Google を IdP として使用している場合、既定の `email` クレームを使用して、IdP ID を Tableau Server ユーザー アカウントにマッピングします。Google を IdP として使用していない場合は、お使いの IdP と連携して、Tableau Server を構成する必要があるクレームを決定してください。

既定: メール クレームを使用してユーザーをマッピングする

既定では、Tableau Server のユーザー名は、IdP ID トークンの email クレームと一致する必要があります。そのため、既定の構成では、Tableau Server のユーザー名としてメール アドレス (UPN とも呼ばれます) を使用する必要があります。Google を IdP として使用している場合、Tableau Server のユーザー名はユーザーのGmailアドレス (alice@gmail.com) にする必要があります。完全なメール アドレスを使用することで、2 人のユーザーが同じメールを異なるメールホストを使用している場合でも、Tableau Server におけるユーザー名の一意性を確保することができます。

注: Tableau Server でユーザー アイデンティティを作成するときに、ユーザー名、パスワード、およびメール アドレス (任意) を指定します。既定の構成で OpenID Connect を使用する場合、ユーザー名 (電子メール アドレスとして表す) には IdP のユーザー名と一致する値を使用する必要があります。Tableau Server のユーザー ID で任意で指定したメール アドレスは OpenID 認証には使用されません。

ドメイン名を無視

Tableau Server で IdP email クレームをユーザー アカウントと照合する際にメール アドレスのドメイン部分を無視するように、Tableau を構成することができます。このシナリオでは、IdP の email クレームは alice@example.com となる可能性がありますが、これは、Tableau Server の alice という名前のユーザーと一致します。email クレームのユーザー名部分には一致するが、ドメイン部分には一致しないユーザーをすでに Tableau Server で定義している場合は、ドメイン名を無視することができます。

重要: ユーザー ドメイン名を無視する際は、細心の注意を払ってください。特に、IdP で作成した構成済みドメイン全体で、ユーザー名が一意であることを確認してください。

ユーザーのドメイン名を無視するようTableau Server を設定すると、意図しないユーザーがログオンする結果になる可能性があります。IdP が複数のドメイン (example.com および tableau.com) のドメイン用に構成された事例を考慮します。組織内に、同じ名前で異なるユーザー アカウント (alice@tableau.com とalice@example.com) を使用する 2 人のユーザーがいる場合、OpenID プロビジョニング シーケンスを最初に完了した人物は IdP での sub マッピングを申請しま

す。間違ったユーザーがマッピングされている場合、もう一人のユーザーは関連付けられた sub 値がリセットされるまでログオンできなくなります。

Tableau Server が IdP からのユーザー名のドメイン名を無視するよう構成するには、tsm authentication openid configure --ignore-domain を true に設定します。詳細については、tsm authentication open-id <commands> を参照してください。

tsm authentication openid configure --ignore-domain オプションを変更してユーザー名のドメインを無視するようにすると、Tableau Server のすべてのユーザー名がドメイン名を持たなければならなくなります。

カスタム クレームを使用してユーザーをマッピングする

OpenID Connect で言及したように、sub クレームは IdP クレームに含まれることが多くあります。一般的に、sub クレームは指定されたユーザー アカウントを識別する一意の文字列です。sub クレームを使用することのメリットは、あなたや別の管理者が他のユーザー属性や、そのアカウントと関連付けられている IdP クレーム (メール、電話番号など) を更新していても、このクレームが変化しないことです。既定では、Tableau Server は IdP ID トークン内の sub クレームに従って OpenID ユーザーを識別して検証します。

OpenID sub クレーム値は、対応する Tableau Server のユーザーにマッピングする必要があります。sub クレームは任意の文字列のため、最初のサインイン セッション時、別のクレームを使用してアカウントに関連付けられます。ユーザーが OpenID を使用して初めて Tableau Server にサインインするとき、Tableau は OpenID ユーザー アカウントを、それに対応する Tableau Server.のユーザー アカウントと照合します。既定では、Tableau は IdP クレーム、email を使用して Tableau ユーザーを識別します。Tableau は、OpenID からの sub クレームを使用してユーザーのレコードを更新します。その後のセッションでは、ID トークンに他のクレームとともに sub クレームが常に含まれるため、Tableau は sub クレームのみを使用してユーザーを識別します。

一部の組織では、電子メール アドレスを使用したユーザー名のマッピングは IdP によってサポートされていません。Tableau Server 10.2 以降、任意の IdP クレームのユーザー アカウントを Tableau Server ユーザー名にマッピングできるようになりました。

使用しているIdP クレームは、それに対応するTableau Server ユーザー名に正確にマッピングする必要があります。以下の例では、ユーザー名は `kwilliams` です。

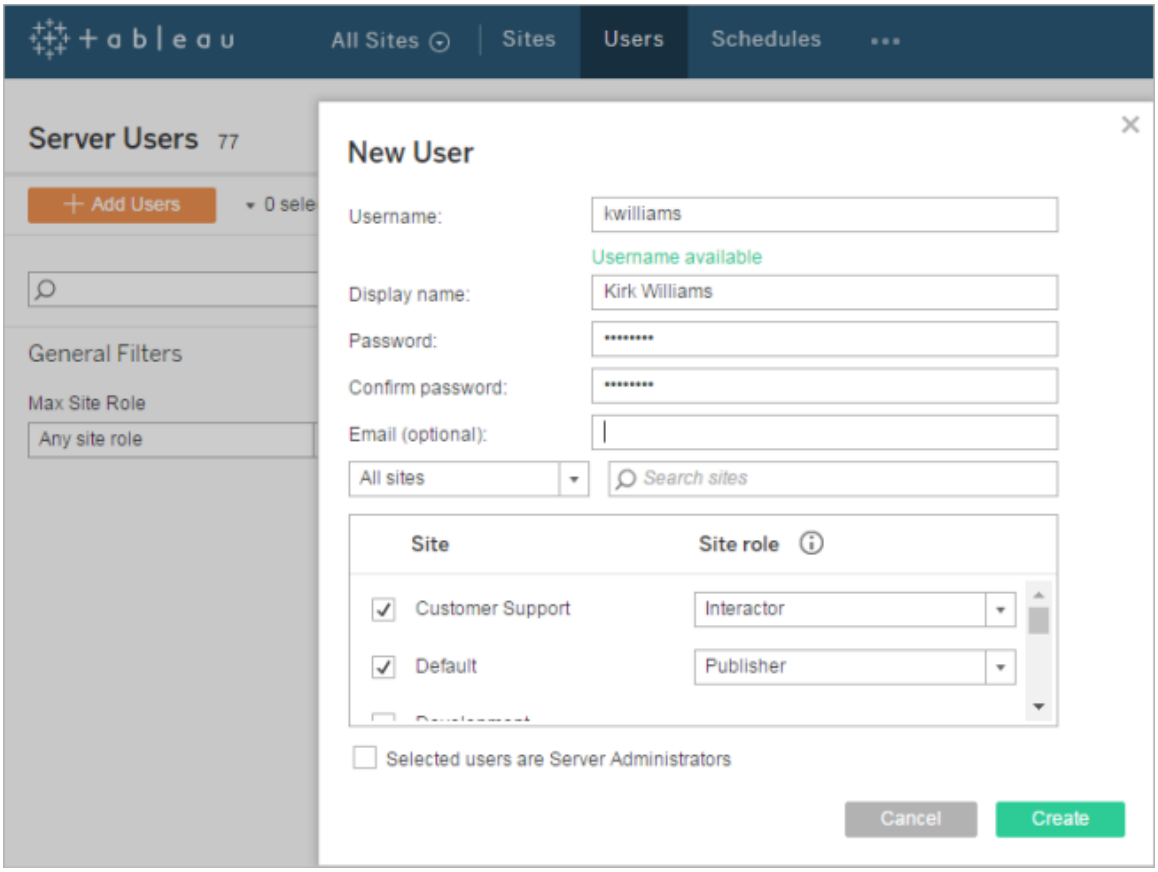


Tableau Server でのID マッピングに使用するIdP クレームを変更するには、`tsm authentication openid map-claims --user-name` コマンドを使用します。詳細については、tsm authentication open-id <commands>を参照してください。

sub クレームを変更する

上記のように、`sub` クレームは、最初のマッピング セッションの後にTableau Server がユーザーの識別に使用する識別子です。`sub` クレームは、それに対応するTableau Server のユーザー アカウントに記述されます。お使いのIdP が `sub` クレームを提供しない場合は、代わりに任意のクレームを指定できます。`sub` のように、指定するクレーム値は一意であり、他のユーザー クレームが更新されたときに変更されない必要があります。

既定の sub クレームに異なる IdP クレームを指定するには、`tsm authentication openid map-claims --id` コマンドを使用します。詳細については、tsm authentication open-id <commands> を参照してください。

ここで、`arbitraryClaim` は `sub` クレームの代わりとして使用する IdP の名前です。

認証コンテキスト

OpenID Connect IdP に特定の認証コンテキストが必要な場合は、`vizportal.openid.essential_acr_values` および `vizportal.openid.voluntary_acr_values` 構成キーを使用して、必須および任意の ACR 値のリストを指定できます。詳細については、tsm configuration set のオプションを参照してください。

OpenID Connect に対応するように ID プロバイダーを構成する

このトピックでは、Tableau Server で OpenID Connect (OIDC) を使用するための ID プロバイダー (IdP) の構成に関する情報を提供します。これは、複数のステップにわたるプロセスの 1 ステップです。次のトピックでは、Tableau での OIDC の構成および使用に関する情報を提供します。

1. OpenID Connect の概要
2. OpenID Connect に対応するように ID プロバイダーを構成する (現在のセクション)
3. OpenID Connect に対応するように Tableau Server を構成する
4. OpenID Connect を使用して Tableau Server にサインインする

Tableau Server で OpenID Connect を使用するには、ID プロバイダー (IdP) で使用するアカウント、および IdP で使用するプロジェクトまたはアプリケーションが必要です。Tableau Server を構成するときには、次の情報を入力する必要があります。

- **クライアント ID:** IdP がアプリケーションに割り当てた識別子です。
- **クライアント シークレット:** これは、Tableau が IdP からの応答の真偽を検証するために使用するトークンです。この値は秘密で、保護しておく必要があります。

- **設定 URL:** これは、OpenID プロバイダーのメタデータを含む、プロバイダー設定を見つける文書の URL です。

リダイレクト URL

IdP によっては、Tableau Server のリダイレクト URL が必要となります。

以下の構文を使用し、IdP の URL を手動で構築できます。

```
<protocol>://<host>/vizportal/api/web/v1/auth/openIdLogin
```

たとえば、`https://tableau.example.com/vizportal/api/web/v1/auth/openIdLogin` とします。

署名アルゴリズム

Tableau Cloud は RS256 (SHA-256 を使用する RSA) をサポートしています。

IdP プロセスの例

次の手順は、プロバイダーに従うステップの概要を示しています。例として、手順ではプロバイダーに Google を使用します。ただし、各プロバイダーの流れは多少異なるため、手順の詳細 (および順序) はプロバイダーによって異なります。

1. プロバイダーの開発者向けサイトに登録し、サインインします。たとえば、Google の場合、URL https://console.developers.google.com で開発者コンソールに移動できます
2. 新しいプロジェクト、アプリケーション、依存するパーティアカウントを作成します。
3. 開発者ダッシュボードで、OAuth 2.0 クライアント ID およびクライアントシークレットを取得する手順に従います。後で使用するためにこれらの値を記録します。

   注: クライアントシークレットは、安全な場所に保管してください。
4. 開発者向けサイトで、IdP が OpenID Connect の検出に使用するエンドポイントの URL を見つけます。たとえば、Google は URL https://accounts.google.com/.well-known/openid-

configuration を使用します。後で使用するためにこの URL を記録します。

または、IdP から静的検出ドキュメントが提供された場合は、後で使用するためにそのファイルを Tableau Server のローカル ディレクトリにコピーします。

OpenID Connect に対応するように Tableau Server を構成する

このトピックでは、シングルサインオン (SSO) に OpenID Connect (OIDC) を使用するようTableau Server を構成する方法を説明します。これは、複数のステップにわたるプロセスの 1 ステップです。次のトピックでは、Tableau Server.での OIDC の構成および使用に関する情報を提供します。

1. OpenID Connect の概要
2. OpenID Connect に対応するように ID プロバイダーを構成する
3. OpenID Connect に対応するように Tableau Server を構成する (現在のセクション)
4. OpenID Connect を使用して Tableau Server にサインインする

注:

- ここで説明するステップを実行する前に、OpenID Connect に対応するように ID プロバイダーを構成するで説明されている OpenID アイデンティティプロバイダー (IdP) を構成する必要があります。
- このトピックで説明する手順は、Tableau Server のセットアップ中に TSM で構成された OIDC 認証に適用されますが、アイデンティティプールを使用して構成された OIDC 認証には適用されません。アイデンティティプールの詳細については、「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。
- Tableau REST API および tabcmd では、OIDC シングル サインオン (SSO) はサポートされていません。tabcmd または REST API を使用するには、ユーザーは TableauID アカウントを使用して Tableau Server にサインインする必要があります。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

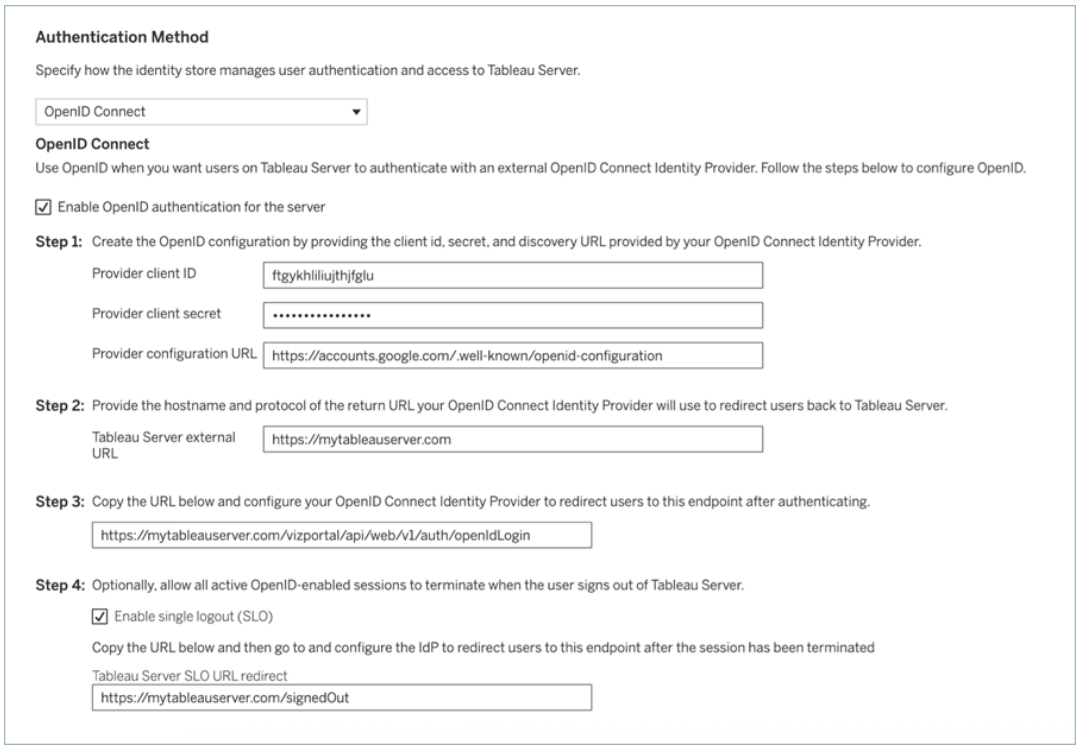
   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[ユーザー アイデンティティとアクセス]** をクリックし、次に **[認証方法]** をクリックします。

3. **[認証方法]** で、ドロップダウン メニューから **[OpenID Connect]** を選択します。

4. OpenID Connect で、**[サーバーの OpenID 認証を有効化]** を選択します。

5. 自分の組織の OpenID 構成情報を次のように入力します。



注:

- **ステップ 3:** プロバイダーが、パブリック URL でホストされたファイルではなくローカル コンピューター上でホストされた構成ファイルに依存している場合は、tsm authentication open-id <commands> を実行してファイルを指定できます。`--metadata-file <file_path>` オプションを使用して、ローカル IdP 構成ファイルを指定します。

- **ステップ 4:** Tableau Server 2025.3 以降では、シングル ログアウトを有効にして、サインアウト後にユーザーをリダイレクトする URL を指定できます。

6. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。
7. ページ上部の **[変更を保留中]** をクリックします。

8. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

このセクションでは、TSM コマンド ライン インターフェースを使用して OpenID Connect を構成する手順を説明します。OpenID Connect の初期構成には、構成ファイルを使用することもできます。openIDSettings エンティティを参照してください。

1. tsm authentication open-id <commands> の `configure` コマンドを使用して以下の必要オプションを設定してください。

   - `--client-id <id>`: IdP がアプリケーションに割り当てているプロバイダー クライアント ID を指定します。たとえば、`"xxxkjwdlnaoiloadjkwha"` のように使用します。

   - `--client-secret <secret>`: プロバイダー クライアント シークレットを指定します。これは、Tableau が IdP からの応答の真偽を検証するために使用するトークンです。この値は秘密で、保護しておく必要があります。たとえば、`"xxxhfkjaw72123="` のように使用します。

   - `--config-url <url>` または `--metadata-file <file_path>`: プロバイダー構成 json ファイルの場所を指定します。プロバイダーがパブリック JSON 検出ファイルをホストしている場合、`--config-url` を使用してください。それ以外の場合、代わりにローカル コンピューター上のパスと `--metadata-file` のファイル名を指定してください。

- `--return-url <url>`: サーバーの URL です。一般的には、これはサーバーのパブリック名 (`"http://example.tableau.com"` など) になります。

たとえば、このコマンドを実行します。

```
tsm authentication openid configure --client-id
"xxxkjwdlnaoiloadjkwha" --client-secret "xxxhfkjaw72123=" --
config-url "https://example.com/openid-configuration" --return-
url "http://tableau.example.com"
```

注:

- openIDSettings エンティティを使用して Open ID Connect に設定できるその他のオプション構成もあります。または、tsm authentication open-id <commands> を使用できます。さらに、IdP クレーム マッピングを構成する必要がある場合は、openid map-claims のオプションを参照してください。
- Tableau Server 2025.3 以降では、tsm authentication open-id <commands> を使用して、必要に応じてシングル ログアウト (SLO) を有効にできます。

2. 次のコマンドを入力して Open ID Connect を有効にします。

   ```
   tsm authentication openid enable
   ```

3. `tsm pending-changes apply` を実行して変更を適用します。

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

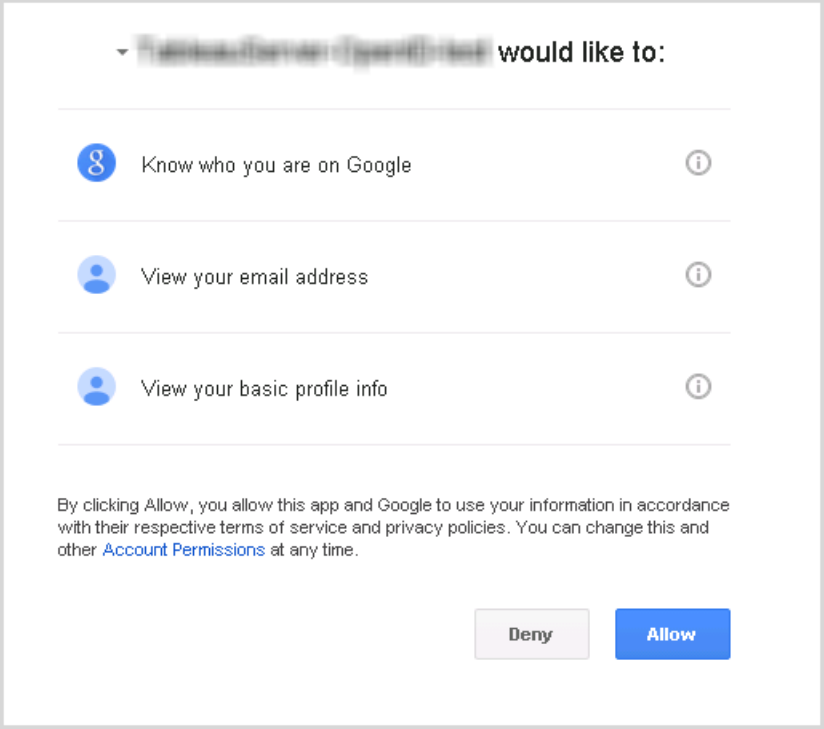
OpenID Connect を使用して Tableau Server にサインインする

このトピックでは、OpenID Connect を使用した Tableau Server へのサインインの詳細を提供します。次のトピックは、Tableau Server での OpenID Connect の構成および使用についての詳細を提供します。

- OpenID Connect
- OpenID Connect に対応するように ID プロバイダーを構成する
- OpenID Connect に対応するように Tableau Server を構成する
- OpenID Connect を使用して Tableau Server にサインインする (現在のセクション)

OpenID Connect を使用してサインインする

Tableau Server が OpenID Connect を使用するよう構成した後、サーバーにアクセスしてサインインが未完了のユーザーは、IdP サイトにリダイレクトされ、そこでサインインを要求されます。ユーザーは IdP での認証資格情報を入力します。多くの場合、ユーザーは次の例のように、Tableau Server と情報を共有するため、IdP を認証するよう求められます。



ユーザーが OpenID Connect を使用してサインインした場合、IdP は Tableau Server にリダイレクトされる情報の一部として一意のユーザー識別子 (OpenID では sub 値) を送信します。この sub 値は、ユーザーの Tableau ユーザー ID と関連付けられます。

サーバー管理者にコマンドライン ツールのサインインを制限する

Tableau Server を操作するためのコマンド ライン ツール (`tabcmd`、TSM、および `tableau.com`) は、OpenID Connect を使用したサインインをサポートしていません。サーバーで OpenID Connect が有効になっている場合、これらのツールは Tableau Server ユーザー名およびパスワードを使用してサインインするよう要求します。

ユーザーが通常 OpenID Connect を使用して認証を行っている場合であっても、各ユーザーは Tableau Server ユーザー名およびパスワードを持っています。つまり、ユーザーは `tabcmd` などのコマンドライン ツールを使用できます。セキュリティ対策として、コマンドライン ツールを使用できるのはサーバー管理者のみであることを確認できます。これを実行するには、`tsm configuration set` を使用して `wgserver.authentication.restricted` を `true` に設定します。この設定が `true` になっている場合、サーバー管理者のみがユーザー名とパスワードを使用して Tableau Server にサインインできます。その他のすべてのユーザーは、OpenID Connect などのシングル サインオン (SSO) オプションを使用してサーバーにサインインする必要があります。その結果、管理者以外のユーザーもコマンドライン ツールを使用できなくなります。これを変更するには、次の TSM コマンドのシーケンスを実行します。

```
tsm configuration set -k wgserver.authentication.restricted -v true

tsm pending-changes apply
```

OpenID 接続認証要求パラメーター

Tableau Server から送信された OpenID 認証要求は、このトピックに記載されている限られたパラメーターのセットを使用して情報を渡します。OpenID IdP が以下のリストに記載されていないパラメーターを要求する場合は、Tableau Server での使用と互換性がありません。

- `scope`。この値は、IdP に対し、返すよう要求するユーザー情報を伝えるプロファイルを指定します。この値は、Tableau Server 管理者が設定できます。既定値は "openid email profile" です。詳細については、本書の後半に記載されている「範囲の値を

設定する」を参照してください。

- `response_type`。OpenID Connect は複数のフローをサポートします。この値は IdP に対し、Tableau Server が期待するフローを伝えます。Tableau では、認可コードフローのみをサポートし、値は常に "code" に設定されます。
- `client_id`。この値は、IdP に要求の発信元を知らせるサーバーの ID ([Tableau Server の構成] ダイアログ ボックスの **[プロバイダー クライアント ID]**) を指定します。サービスの登録時、IdP によって提供されます。値は Tableau Server 管理者が設定できます。
- `redirect_uri`。この値は、OpenID Connect を使用してユーザーが認証された後に IdP がリダイレクトする URL を指定します。URL にはホストとプロトコルを含める必要がありますが (たとえば、`http://example.tableau.com`)、Tableau は URL エンドポイントを提供します。
- `nonce`. Tableau Server は、IdP リダイレクト先のクライアントが、IdP から戻されたエンティティと一致していることを検証する nonce 値を生成します。

scope 値の構成

`scope` 値は、Tableau Server がユーザーに関して要求する情報を IdP に示します。既定では、Tableau Server は値 "openid profile email" を送信します。これは、Tableau が OpenID を使用して認証し (`scope` 属性値のこの部分を常に含める必要があります)、Tableau Server がユーザー認可コードの交換中にユーザープロフィールとメール情報を要求していることを示します。

この既定の範囲が適切でないシナリオでは、Tableau Server にユーザーに関するカスタムの情報を要求させることができます。これを行うには、IdP にカスタムのプロファイル (例: "tableau-scope" など) を設定します。次に、カスタム プロファイルの名前を使用して範囲要求を送信するよう Tableau Server を設定できます。

Tableau Server が要求する範囲の値を変更するには、次の TSM CLI コマンドを使用します。

```
tsm authentication openid configure --custom-scope-name custom-scope-name
```

注:

- Tableau Server は常に "openid" を範囲の値の一部として含めます (`custom_scope` 設定に含まれていない場合であっても含めます)。
- TSM 認証構成コマンドは、Tableau Server のセットアップ中に TSM で構成された OIDC 認証にのみ適用されます。アイデンティティプールの OIDC 認証構成を変更するには、Tableau REST OpenAPI を使用する認証構成の更新エンドポイントを使用します。

Tableau Server で OpenID Connect 用に IdP を変更する

このトピックでは、OpenID Connect を使用するように Tableau Server を設定している場合の、アイデンティティプロバイダー (IdP) の変更の詳細を提供します。

プロバイダーの変更

Tableau Server が使用するよう構成されている IdP を変更するよう決定することができます。これを実行するには、最初の IdP の構成に使用した手順に従い、アカウントを確立、カスタマ ID およびシークレットを取得、その情報を使用して Tableau Server を構成、IdP に Tableau Server 用のリダイレクト URL を提供します。詳細については、OpenID Connect に対応するように Tableau Server を構成するを参照してください。

ユーザー識別子のリセット

ただし、追加のステップも実行する必要があります。既に Tableau Server ユーザーに関連付けられているユーザー識別子 (`sub` 値またはクレーム) をクリアする必要があります。新しい IdP は、各ユーザーに対して異なる `sub` 値を持っています。ユーザーが新しい IdP を使用してサインインしたときに Tableau Server が新しい `sub` 値を保存できるよう、既存の値をクリアする必要があります。

ユーザーの sub 値をクリアするには、`tabcmd reset_openid_sub` コマンドを使用します。次の例のように、個々のユーザーの `sub` 値をリセット (クリア) できます。

```
tabcmd reset_openid_sub --target-username jsmith
```

次のコマンドを使用してすべてのユーザーの sub 値をクリアすることもできます。

```
tabcmd reset_openid_sub --all
```

注: アイデンティティプールのメンバーのユーザー識別子をクリアすることはできません。

OpenID Connect のトラブルシューティング

Tableau Server での OpenID Connect (OIDC) の問題をトラブルシューティングするには、次のトピックを使用します。

OIDC プロトコルは多くの ID プロバイダーがサポートしています。OIDC プロトコルはオープンかつ柔軟な規格であるため、すべての環境で同じように実装されるわけではありません。OIDC に対応するように Tableau Server を構成するときに管理者が直面する問題の多くは、ID プロバイダーによって OIDC の実装方法が異なることが原因で発生しています。Tableau Server で OIDC をセットアップする際にエラーが発生した場合は、IdP と連携して問題を解決することをお勧めします。

拡張 OpenID ロギングの有効化

Tableau Server の OpenID Connect の問題を効率的にトラブルシューティングするには、ログ レベルをデバッグに設定して拡張ログを有効にし、TSM コマンドで `vizportal.openid.full_server_request_logging_enabled` 構成キーを `true` に設定して OpenID の完全なログを有効にします。

```
tsm configuration set -k vizportal.log.level -v debug

tsm configuration set -k vizportal.openid.full_server_request_
logging_enabled -v true

tsm pending-changes apply
```

トラブルシューティングが完了したら、両方の構成キーをデフォルトの値に戻して、ログに収集される情報を制限し、ログ ファイルのサイズを小さくすることをお勧めします。構成キーをデフォルトにリセットする方法の詳細については、構成キーを既定値にリセットするを参照してください。

注: ID プールの拡張ログ機能はサポートされていません。ただし、`vizportal.log.level debug` ロギングはサポートされています。

コマンド ラインからのサインイン

Tableau Server が OIDC を使用するように構成されている場合でも、tabcmd、Tableau REST API、または Tableau Data Extract コマンド ライン ユーティリティ (Tableau Desktop で提供) を使用して Tableau Server にサインインすると、OIDC は使用されません。

ログインできませんでした

次のメッセージが表示されてログインに失敗する:

```
Login failure: Identity Provider authentication unsuccessful for
user <username from IdP>. Failed to find the user in Tableau Cloud.
```

このエラーは通常、Tableau Server に保存されているユーザー名とIdP が提供したユーザー名が一致しないときに発生します。これを修正するには、これらのユーザー名が一致していることを確認してください。たとえば、Jane Smith のユーザー名が `jsmith` として IdP に保存されている場合は、Tableau Server でも `jsmith` として保存する必要があります。

エラー 69: "サインインできません"

Web ブラウザーで Tableau Server にサインインしようとすると、エラー 69 が返され、「サインインできません」というエラー メッセージが表示される場合があります。サインインに失敗しました。Tableau Server 管理者に問い合わせてください。"このメッセージは次の URL から返されます。
`https://example.com/#/error/signin/69?redirectPath=%2`

このエラーが表示される場合は、IDP プロバイダーを確認し、IdP が `client_secret_basic` (Tableau の既定) ではなく`client_secret_post` を想定しているかどうかを検証します。

IdP が `client_secret_post` を想定している場合は、`vizportal.openid.client_authentication` パラメーターを `client_secret_post` に設定する必要があります。

たとえば、このエラーが表示され、Salesforce IdP に OIDC を既に構成している場合は、`vizportal.openid.client_authentication` パラメーターを設定する必要があります。

詳細については、tsm configuration set のオプションを参照してください。

OpenID エラー ログ

OpenID 認証は Tableau Server の外部で行われるため、認証に関する問題のトラブルシューティングが困難になる場合があります。ただし、サインインの試行は Tableau Server によって記録されます。ログ ファイルのスナップショットを作成して、問題のトラブルシューティングに使用できます。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

注: OpenID 関連のイベントを記録するには、tsm configuration set のオプションを使用して、`vizportal.log.level` を `debug` に設定する必要があります。

解凍されたログ ファイル スナップショットにある次のファイルで、OpenID エラーを確認します。

`\vizportal\vizportal-<n>.log`

ユーザーが見つかりません

ユーザーが初めてログインした後に "sub" クレームが変更された場合、「ユーザーが見つかりません」というエラーが返されることがあります。この問題の詳細は、vizportal のログで確認できます。内容は次のとおりです: `Possible conflicting or stale account: <username> A different user already owns this account`。

この問題が引き続き発生する場合は、そのユーザーまたは Tableau Server の全ユーザーの "sub" クレームをリセットしてください。詳細については、「ユーザー識別子のリセット」を参照してください。

信頼できる認証

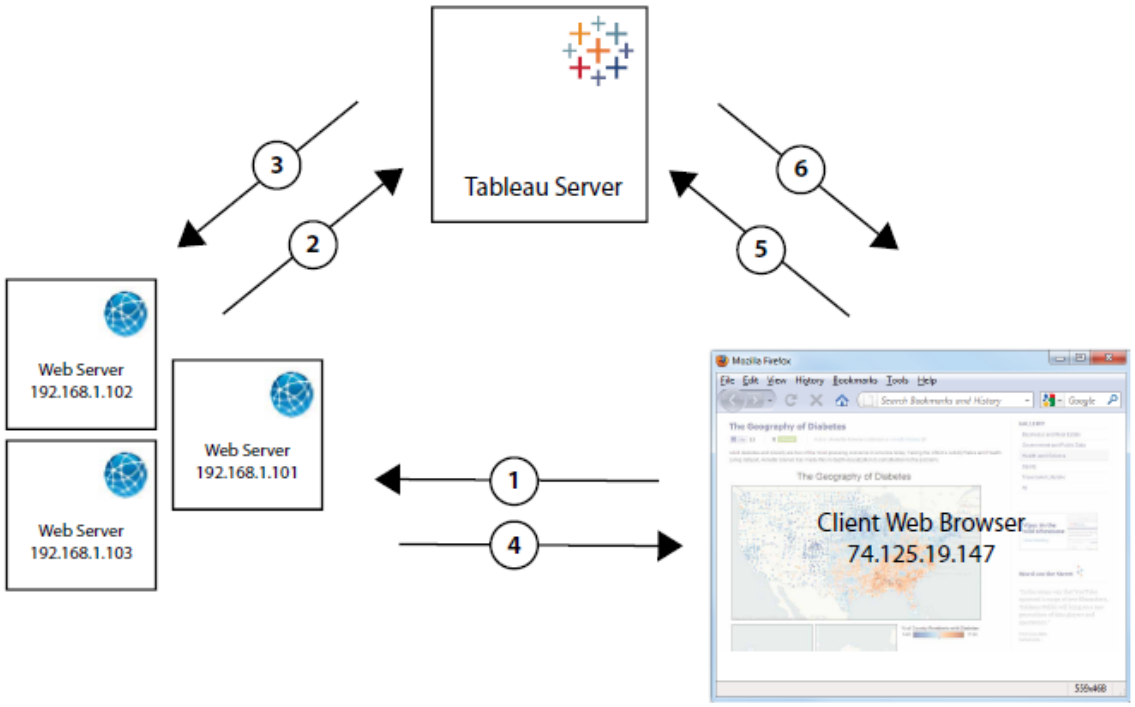
Tableau Server のビューを Web ページに埋め込むときは、そのページにアクセスするすべてのユーザーが Tableau Server のライセンス ユーザーである必要があります。ユーザーがそのページにアクセスすると、ビューが表示される前に Tableau Server へのサインインを求めるプロンプトが表示されます。Web ページ上または Web アプリケーション内でユーザーを認証する手段が既にある場合は、このプロンプトを省略し、信頼できる認証を設定することによりユーザーがサインインを 2 回行わずに済むようにできます。

信頼できる認証とは、Tableau Server と 1 つ以上の Web サーバーとの間で信頼関係が設定されていることを意味します。Tableau Server が信頼できる Web サーバーから要求を受け取ると、Tableau Server は Web サーバーで必要とされる認証は既に処理されていると見なします。

注: 埋め込みビューで信頼できる認証を使用する場合は、サードパーティのクッキーを許可するようにクライアントブラウザーを構成する必要があります。

信頼できる認証のしくみ

下の図は、クライアントの Web ブラウザー、Web サーバー、Tableau Serverの間で信頼できる認証がどのように機能するかを示しています。



ユーザーが Web ページにアクセス: Tableau Server のビューが埋め込まれた Web ページにユーザーがアクセスすると、Web ページにより、そのページの HTML に対する GET 要求が Web サーバーに送信されます。

Web サーバーが Tableau Server に POSTS を送信する。Web サーバーは、POST 要求を信頼できる Tableau Server (たとえば、`https://<server_name>/trusted` ではなく `https://<server_name>`) に送信します。この POST 要求には `username` パラメーターが必要です。`username` の値は、Tableau Server のライセンスユーザーのユーザー名でなければなりません。Tableau Server が複数のサイトをホストし、ビューが "既定" サイト以外のサイトにある場合、POST 要求には `target_site` パラメーターも含める必要があります。

3 **Tableau Server がチケットを作成する**。Tableau Server は、POST 要求を送信した Web サーバーの IP アドレスまたはホスト名 (上図では 192.168.1.XXX) を確認します。Web サーバーが信頼できるホストとしてリストされている場合、Tableau Server は一意の文字列形式でチケットを作成します。チケットは、発行後 3 分以内に引き換える必要があります。Tableau Server はこのチケット付きの POST 要求に応答します。または、エラーが発生してチケットを作成できない場合は、`-1` の値が返されます。サーバーには IPv4 アドレスが必要です。IPv6 アドレスはサポートされていません。詳細については、Tableau Server から返されたチケットの値 -1を参照してください。

4 **Web サーバーがブラウザーに URL を渡す**。Web サーバーがビューの URL を構築し、そのページの HTML に挿入します。チケットが含まれています (例: `https://<server_name>/trusted/<unique_ticket>/views/<view_name>`)。Web サーバーは HTML をクライアントの Web ブラウザーに渡します。

5 **ブラウザーが Tableau Server のビューを要求する**。クライアントの Web ブラウザーは、チケット付きの URL が含まれている GET 要求を、Tableau Server に送信します。

6 **Tableau Server がチケットを引き換える**。Tableau Server はチケットを引き換え、セッションを作成、ユーザーをログインさせて URL からチケットを削除し、埋め込みビューの最終的な URL をクライアントに送信します。

セッションはユーザーに対し、サーバーにログオンしている場合に、そのユーザーが持つ任意のビューへのアクセスを許可します。既定の構成では、信頼できるチケットを使用して認証されたユーザーは、ビューのみが利用可能などの制限アクセス権を持ちます。彼らはワークブック、プロジェクトページ、またはサーバーでホストされるその他のコンテンツにアクセスできません。

この動作を変更するには、tsm configuration set のオプションの `wgserver.unrestricted_ticket` オプションを参照してください。

信頼できるチケットの保存方法

Tableau Server は次のプロセスを使用して、Tableau Server リポジトリに信頼できるチケットを保存します。

1. Tableau Server は 2 パートチケットを作成します。最初の部分は Base64 エンコードされた一意の ID (UUID)、2 つ目の部分は 24 文字のランダムなシークレット文字列です。
2. Tableau Server はシークレット文字列をハッシュし、一意の ID を使用してリポジトリに保存します。ハッシュはシークレット文字列を入力として認識し、アルゴリズムを使用して一意の文字列を計算します。この一意の文字列は、未認証のユーザーからシークレット文字列のセキュリティを保護します。
3. Tableau Server は、クライアントに Base64 UUID と元の 24 文字のランダム化された文字列を送信します。
4. クライアントは、ビューへの要求の一部として、Base64 UUID と元の 24 文字のシークレット文字列を Tableau Server に返します。
5. Tableau Server は Base64 UUID を使用して文字列ペアを見つけ、シークレット文字列をハッシュ化してリポジトリに保存されているハッシュと一致していることを検証します。

このプロセスにより、Tableau Server に保存されている信頼できるチケット コンテンツを、ユーザーの偽装や認証によって保護されているコンテンツへのアクセスに使用できないようにします。ただし、完全に信頼されているチケットは HTTP 経由で Tableau Server とクライアントの間を送信されるため、プロセスは、HTTP データの安全で暗号化された送信に依存します。そのため、信頼できるチケットを、SSL/TLS またはネットワーク暗号化の別のレイヤーでのみ展開することをお勧めします。

Tableau Server への信頼できる IP アドレスまたはホスト名の追加

信頼できる認証を設定するための最初のステップは、Tableau Server を 1 つ以上の Web サーバーからの要求を認識して信頼するように構成することです。

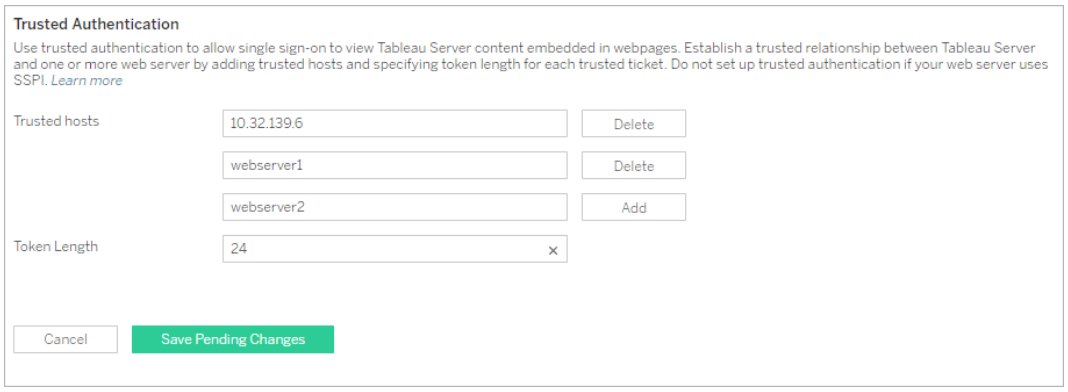
TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<server_name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[ユーザー ID とアクセス]** をクリックしてから、**[信頼できる認証]** をクリックします。

3. **[信頼できる認証]** で信頼できる各ホストに関して名前または IP アドレスを入力し、**[追加]** をクリックします。



注:
指定した値は以前の設定を完全に上書きします。そのため、既存のリストを修正する場合は、ホストの完全なリストを含める必要があります。

静的 IP アドレスが必要: ホスト名を使用する場合でも、指定する Web サーバーには静的 IP アドレスを使用する必要があります。

信頼できるチケットを要求するコンピューター (手順 2 の信頼できる認証 で構成されているコンピューターの 1 台) と Tableau Server の間に 1 台以上のプロキシ サーバーが存在する場合、`tsm configuration set gateway.trusted` オプションを使用してプロキシ サーバーを信頼できるゲートウェイとして追加する必要もあります。手順については、Tableau Server のプロキシとロード バランサーの設定を参照してください。

4. **[トークンの長さ]** に値を入力します (オプション)。

   トークンの長さにより、信頼できるチケットごとの文字数が決まります。既定設定の 24 文字は、ランダム性のある 144 ビットで提供されます。値は、9 から 255 (包含) の間の任意の整数に設定できます。

5. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

6. ページ上部の **[変更を保留中]** をクリックします。

7. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

1. 次のコマンドを入力します。

   ```
   tsm authentication trusted configure -th <trusted IP address or host name>
   ```

   上記のコマンドで、<trusted IP address> は使用している 1 台以上の Web サーバーの IPv4 アドレスまたはホスト名のコンマ区切りリストであり、各ホスト名または IP アドレスを引用符で囲む必要があります。

   > **注**: 指定した値は以前の設定を完全に上書きします。そのため、tsm authentication trusted configure -th コマンドをホストの完全なリストを含める必要があります。(tsm authentication trusted configure -th コマンドを繰り返し実行して、ホストのリストを修正することはできません。)

   例は次のとおりです。

   ```
   tsm authentication trusted configure -th "192.168.1.101", "192.168.1.102", "192.168.1.103"
   ```

   または

   ```
   tsm authentication trusted configure -th "webserv1", "webserv2", "webserv3"
   ```

   > **注:**
   > リスト内の各ホスト名または IP アドレスは二重引用符で囲み、後ろにコンマと各コンマ

> の後にスペースを置く必要があります。
> ホスト名を使用する場合でも、指定するWebサーバーには静的IPアドレスを使用する必要があります。

2. 信頼できるチケットを要求するコンピューター(手順2の信頼できる認証で構成されているコンピューターの1台)とTableau Serverの間に1台以上のプロキシサーバーが存在する場合、`tsm configuration set gateway.trusted`オプションを使用してプロキシサーバーを信頼できるゲートウェイとして追加する必要もあります。手順については、Tableau Serverのプロキシとロードバランサーの設定を参照してください。

   次のコマンドを入力して、すべてのサーバー構成ファイルへの変更を保存します。

   ```
   tsm pending-changes apply
   ```

3.

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply`コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt`オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

   Tableau Serverにjsonファイルを渡すことで作成できるその他の信頼できる認証構成(レガシーサポート、ロギング、およびタイムアウト設定)があります。trustedAuthenticationSettingsエンティティを参照してください。

次に、Web サーバーを Tableau Server からチケットを受け取るように構成する必要があります。

Tableau Server からのチケットの取得

Tableau Server に信頼できる IP アドレスを追加すると、POST 要求を介して Tableau Server からチケットを受け取るように Web サーバーを構成できます (図のステップ 2)。POST 要求は

`http://<server_name>/trusted` に送信される必要があります。たとえば、POST 要求は `http://tabserv` ではなく`http://tabserv/trusted` に送信される必要があります。

> **注:** SSL が有効になっている場合、http の代わりに https を使用する必要があります。たとえば、 `https://tabserv/trusted`。

次に、Tableau Serverへの POST 要求に使用できるデータを示します。

- **`username=<username>`** (必須): Tableau Server のライセンス ユーザーのユーザー名。ローカル認証を使用している場合、ユーザー名は単純な文字列にできます (`username=jsmith` など)。複数のドメインを持つ Active Directory を使用している場合は、ドメイン名とユーザー名を含める必要があります (`username=MyCo\jsmith` など)。

- **`target_site=<site id>`** (既定サイトにビューがない場合にのみ必要): Tableau Server で複数のサイトを実行し、ビューが "既定" サイト以外のサイトにある場合は、ビューがあるサイトを指定します (`target_site=Sales` など)。`<site id>` に使用する値は、サイトが作成されたときに提供された [サイト ID] でなければなりません。この値は大文字と小文字が区別されます。**[サイト ID]** が `SAles`の場合、`target_site=SAles` となります。

- **`client_ip=<IP address>`** (オプション): ビューにアクセスする Web ブラウザーがあるコンピューターの IP アドレスの指定に使用されます (`client_ip=123.45.67.891` など)。これは、Tableau Serverの POST 要求を作成する Web サーバーの IP アドレスではありません。このパラメーターを使用する場合の詳細については、オプション: クライアント IP 照合の設定を参照してください。

POST 要求に対する Tableau Server の応答は、固有の文字列形式になります (チケット)。Tableau Server で要求を処理できない場合は、`-1` が返されます。この修正手順のヒントについては、Tableau Server から返されたチケットの値 -1を参照してください。また、ユーザーが埋め込みビューをクリックしたときにそれを正常に認証するには、サードパーティのクッキーを許可するようにブラウザーを構成する必要もあります。

Tableau Server 10.2 でチケットの書式設定が変更されました。チケットの書式設定は 2 つの部分で構成される文字列になりました。各部分は、クライアントに返される前にエンコード化された 128 ビットの文字列です。最初の部分は Base64 エンコード化されたユニバーサルに一意の ID (UUID v4) です。2 つ目の部分は 24 文字のセキュア ランダム文字列です。これらの部分の連結は Base64(UUIDv4):SecureRandomString として表されます。チケットの例は次のようになります。`9D1ObyqDQmSIOyQpKdy4Sw==:dg62gCsSE0QRArXNTOp6mlJ5`.

次に、ビューの場所とチケットを含むビューの URL の構築を Web サーバーに許可するコードを追加する必要があります。

チケット付きビューの表示

POST 要求を作成したら、ビューの場所と Tableau Server からのチケットを Web サーバーに与えるコードを作成する必要があります。この情報はビューの表示に使用されます。この指定方法は、ビューが埋め込まれているかどうかと、Tableau Server で複数のサイトを実行しているかどうかによって異なります。

Tableau Server ビューの例

次に、ユーザーが Tableau Server を介したアクセスのみを行うビューを指定する方法の例を示します (ビューは埋め込まれていません):

```
http://<server_name>/trusted/<unique_ticket>/views/<workbook_
name>/<view_name>
```

Tableau Server で複数のサイトを実行し、ビューが "既定" サイト以外のサイトにある場合にのみ、`t/<site_id>` をパスに追加する必要があります。例:

```
http://<server_name/trusted/<unique_
ticket>/t/Sales/views/<workbook_name>/<view_name>
```

Tableau Server URL に表示されるのと同じ大文字/小文字の設定を使用します。

埋め込まれたビューの例

次に、埋め込まれたビューの指定方法の例をいくつか示します。埋め込みコードを使用して行うことができる方法は 2 つあるので、両方の方法を以下に示します。どちらを使用するかに関係なく、

信頼できる認証に固有の情報がいくつかあり、これらを指定する必要があります。詳細については、を参照してください「埋め込みコードの書き出し」を Tableau Server ヘルプで検索してください。

> **注:** 以下の例では埋め込みコードのパラメーターを使用します。詳細については、Tableau ヘルプの「埋め込みコード用パラメーター」を参照してください。

Script タグの例

この例では、次のように ticket オブジェクトパラメーターを使用します。

```
<script type="text/javascript"
src="http://myserver/javascripts/api/viz_v1.js"></script>
<object class="tableauViz" width="800" height="600"
style="display:none;">
    <param name="name" value="MyCoSales/SalesScoreCard" />
    <param name="ticket"
value="9D1ObyqDQmSIOyQpKdy4Sw==:dg62gCsSE0QRArXNTOp6mlJ5" />
</object>
```

次に、複数サイトの Tableau Server の場合に上記の例がどのようになるかを示します。ここで、ビューは Sales サイトにパブリッシュされています。

```
<script type="text/javascript"
src="http://myserver/javascripts/api/viz_v1.js"></script>
<object class="tableauViz" width="800" height="600"
style="display:none;">
    <param name="site_root" value="/t/Sales" />
    <param name="name" value="MyCoSales/SalesScoreCard" />
    <param name="ticket"
value="9D1ObyqDQmSIOyQpKdy4Sw==:dg62gCsSE0QRArXNTOp6mlJ5" />
</object>
```

ticket を使用する代わりに path パラメーターを使用すると、ビューの完全パスを明示的に示すことができます。path を使用する場合は、TableauJavaScript 埋め込みコードの必須パラメーターである name パラメーターも使用する必要はありません。

```
<script type="text/javascript"
src="http://myserver/javascripts/api/viz_v1.js"></script>
<object class="tableauViz" width="900" height="700"
style="display:none;">
    <param name="path"
value="trusted/9D1ObyqDQmSIOyQpKdy4Sw==:dg62gCsSE0QRArXNTOp6mlJ5/vi-
ews/MyCoSales/SalesScoreCard" />
</object>
```

次に、複数サイトのサーバーではない場合の同じ例を示します。/t/<site_id> が使用されていることに注意してください。

```
<script type="text/javascript"
src="http://myserver/javascripts/api/viz_v1.js"></script>
<object class="tableauViz" width="900" height="700"
style="display:none;">
     <param name="path"
value="trusted/9D1ObyqDQmSIOyQpKdy4Sw==:dg62gCsSE0QRArXNTOp6mlJ5/t/-
Sales/views/MyCoSales/SalesScoreCard" />
</object>
```

Iframe タグの例

```
<iframe
src="http://tabserver/trusted/9D1ObyqDQmSIOyQpKdy4Sw==:dg62gCsSE0QR-
ArXNTOp6mlJ5/views/workbookQ4/SalesQ4?:embed=yes" width="800"
height="600"></iframe>
```

オプション: クライアントIP 照合の設定

既定では、Tableau Serverはチケットの作成または引き換え時に、クライアントの Web ブラウザーの IP アドレスを考慮しません。これを変更するには、次の 2 つの操作が必要です。チケットを取得する POST 要求で client_ip パラメーターを使用して IP アドレスを指定し、以下の手順に従って Tableau Server がクライアントの IP アドレス照合を実行するように設定します。

1. TSM CLI を開いて次のコマンドを入力します。

   ```
   tsm configuration set -k wgserver.extended_trusted_ip_checking
   -v true
   ```

   次のコマンドを入力します。

   ```
   tsm pending-changes apply
   ```

2.

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

信頼できる認証のテスト

以下の手順は、Web サーバーからの信頼できるチケット取得をテストするのに使用できる方法を示しています。この単純なテストは、Web サーバーとTableau Server の間の接続を評価したり、信頼できる認証が正しく構成されているかどうかを評価したりする際に役立ちます。

重要: このトピックに記載されているテストコードは、Tableau Server で信頼できる認証が正しく構成されていることを視覚的に迅速に検証するため、クライアント側のブラウザーで実行されます。HTML ページを読み込むクライアントブラウザーが信頼できるWeb サーバー上にない場合は、クライアントIP を一時的に信頼する必要があります。実際には、信頼できる認証の構成の一部としてクライアントIP アドレスを信頼することはありません。信頼済みの IP アドレスは、管理者を含む任意のユーザーとしてチケットを要求できます。本番環境では、信頼できる認証のフローはすべてサーバー側 (Tableau Server と信頼できるWeb サーバーの間) で実行する必要があります。

このテストはクライアントブラウザーで実行されるため、本番環境での通信パスを正確に再現したものではありません。このテストを正常に実行した後、最終検証のためにサーバー側で POST 要求を使用してユーザーのチケットを要求するテストを行うことをお勧めします。

または、信頼できるチケットジェネレーターを実行して構成をテストできます。次の URL は、Tableau でサポートされていない信頼できるチケットジェネレーターを参照しています。ただし、このジェネレーターは、信頼できるチケット構成をテストするために多くのお客様によって使用されています: https://github.com/mkannan-tsi/Trusted-Ticket-Generator.

ステップ 1: テストユーザーの追加

Tableau Server で、信頼できるチケットの機能のテストに使用できるユーザーを作成します。Tableau Server へのユーザーの追加を参照してください。そのユーザーをサーバー上のサイトに追加し、ユーザーのサイトロールを **[Explorer]** に設定します。

ステップ 2: テスト HTML ページの作成

テストを実行する Tableau Server マシンに保存した新しい .html ファイルに、次のコードを貼り付けます。ラベルやスタイル属性は好みに応じて変更可能です。

```
<html>
<head>
<title>Trusted Ticket Requester</title>
<script type="text/javascript">
  function submitForm(){
    document.getElementById('form1').action =
    document.getElementById('server').value + "/trusted";
  }
</script>
<style type="text/css">
  .style1 {width: 100%;}
  .style2 {width: 429px;}
  #server {width: 254px;}
</style>
```

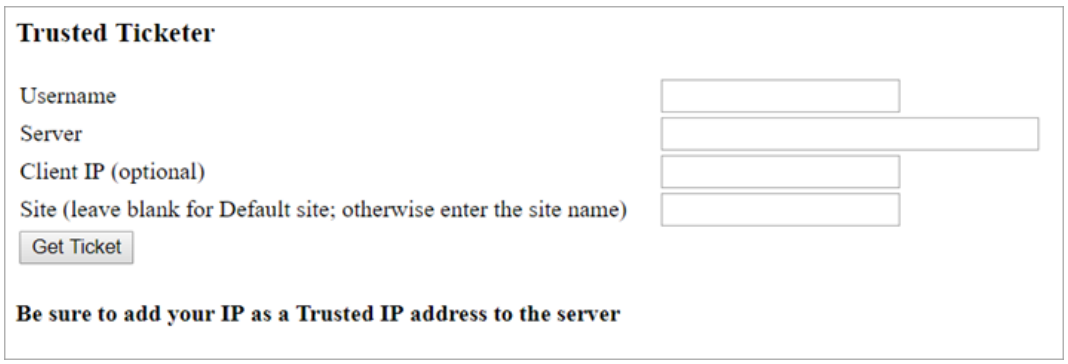
```
</head>
<body>
<h3>Trusted Ticketer</h3>
<form method="POST" id="form1" onSubmit="submitForm()">
  <table class="style1">
    <tr>
      <td class="style2">Username</td>
      <td><input type="text" name="username" value="" /></td>
    </tr>
    <tr>
      <td class="style2">Server</td>
      <td><input type="text" id="server" name="server"
value="https://" /></td>
    </tr>
    <tr>
      <td class="style2">Client IP (optional)</td>
      <td><input type="text" id="client_ip" name="client_ip"
value="" /></td>
    </tr>
    <tr>
      <td class="style2">Site (leave blank for Default site;
otherwise enter the site name)</td>
      <td><input type="text" id="target_site" name="target_site"
value="" /></td>
    </tr>
    <tr>
      <td class="style2"><input type="submit" name="submittable"
value="Get Ticket" /></td>
      <td> </td>
    </tr>
  </table>
```

```
</form>
<h4>Be sure to add your IP as a Trusted IP address to the
server</h4>
</body>
</html>
```

ステップ 3: Tableau Server からの信頼できるチケットの取得

次の手順を実行すると、Tableau Server に信頼できるチケットを返します。

1. 前のステップで作成した Web ページを開きます。



   この操作には Javascript が必要なため、Web ブラウザーにより、スクリプトの実行を許可するよう求めるプロンプトが表示されます。

2. テキストボックスに、次の情報を入力します。

   - **ユーザー名**: ステップ 1 で作成したテストユーザー。
   - **サーバー**: Tableau Server のアドレス (例: `https://<server_name>`)。
   - **クライアント IP (オプション)**: ユーザーのコンピューターの IP アドレス (クライアントの信頼できる IP マッチングに対して構成されている場合)。
   - **サイト**: テストユーザーがメンバーである Tableau Server サイトの名前。

3. **[チケットの取得]** をクリックします。次のうちいずれかが返されます。

   - **一意のチケット**: 信頼できるチケットは、base64 エンコード UUID と 24 文字のランダムな文字列で構成される文字列です (例: `9D1OlxmDQmSIOyQpKdy4Sw==:dg62gCsSE0QRArXNTOp6mlJ5`)。

- **-1**: 値 `-1` が返された場合、構成にエラーが含まれます。Tableau Server から返されたチケットの値 **-1** を参照してください。

ステップ 4: 信頼できるチケットを使用したテストアクセス

これでチケットを入手したため、Tableau Server のコンテンツへのアクセスに使用できます。

前のステップで生成した一意のチケットを使用して URL を構築し、信頼できるチケットを使用したアクセスを検証します。単一サイトを使用して Tableau Server にアクセスする場合と、複数のサイトをホストするサーバーの場合では URL 構文は異なります。

既定のサイトサーバー URL

```
https://<server_name>/trusted/<unique_ticket>/views/<workbook_
name>/<view_name>
```

既定以外のサイトサーバー URL

```
https://<server_name>/trusted/<unique_ticket>/t/<site_
name>/views/<workbook_name>/<view_name>
```

URL 内の変数は山括弧 (< および >) で示されています。その他のすべての構文はリテラルです。

信頼できる認証のトラブルシューティング

このセクションには、信頼できる認証を構成するときに発生する可能性のある一般的な問題とエラーが含まれています。

信頼できる認証のエラーの一般的な原因は、プロキシサーバーまたはロードバランサーの設定ミスです。Tableau Server がリバースプロキシサーバーまたはロードバランサーの背後で動作している場合は、「リバースプロキシサーバーやロードバランサーと連携させるための Tableau Server の設定」および「ロードバランサーの追加」を参照してください。

信頼できる認証情報が `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizqlserver/vizql-*.log` に書き込まれます。

ロギングレベルを `info` から `debug` に上げるには、次のコマンドを実行します。

```
tsm configuration set -k vizqlserver.trustedticket.log_level -v
debug
tsm pending-changes apply
```

信頼できる認証展開をテストするには信頼できる認証のテストを参照してください。

関連項目

特定のエラーのトラブルシューティングの詳細については、以下の**「このセクションの他の記事」**からアクセスできる次のトピックを参照してください。

Web サーバーによるチケットの要求

- Tableau Server から返されたチケットの値 -1
- HTTP 401 - 承認されていません
- HTTP 404 - ファイルが見つかりません
- 無効なユーザー (SharePoint または C#)

閲覧者によるチケットの引き換え

- 間違った IP アドレスからチケットを引き換えようとしています
- Cookie 制限エラー

複数の埋め込みビュー間の移動

- サーバーとの通信中にエラーが発生しました (403)

Tableau Server から返されたチケットの値 -1

Tableau Server は、信頼できる認証プロセスの一環としてチケットを発行できない場合に、チケット値として -1 を返します。このシナリオのトラブルシューティングを行う前に、信頼できる認証のトラブルシューティングで指定されたように、信頼できる認証のログレベルを必ず `debug` に設定してください。

このメッセージの理由は、以下のフォルダー内の `vizqlserver_node*-*.log.*` ファイルに書き込まれます。

```
/var/opt/tableau/tableau_server/data/tabsvc/logs/vizqlserver
```

以下を確認します。

- **すべての Web サーバーのホスト名 または IP アドレスが信頼できるホストに追加されている**

  ログ エラー "`Invalid request host: <ip_address>`" は、POST 要求を送信しているコンピューターの IP アドレスまたはホスト名が Tableau Server の信頼できるホストのリストにないことを示している場合があります。このリストに IP アドレスまたはホスト名を追加する方法については、Tableau Server への信頼できる IP アドレスまたはホスト名の追加を参照してください。

- **IP アドレスが IPv4 である**

  信頼できるホストを指定するために IP アドレスを使用している場合、アドレスは Internet Protocol version 4 (IPv4) 形式である必要があります。IPv4 アドレスは次のようになります。123.456.7.890。IPv6 アドレス (たとえば、fe12::3c4a:5eab:6789:01c%34) は、信頼できるホストを入力する手段としてサポートされません。

- **POST 要求のユーザー名が有効な Tableau Server ユーザーである**

  POST 要求で送信するユーザー名は、Tableau Server ライセンス ユーザーでなければなりません。管理者として Tableau Server にサインインすると、ユーザーの一覧を表示できます。

  次のログ エラーは、ユーザーの POST に関する問題を示しています。

  - "`Missing username and/or client_ip`"
  - "`Invalid user: <username>`"
  - "`Unlicensed user is not allowed: <username>`"

  POST 要求内のユーザー名にドメインが含まれている

  Tableau Server がローカル認証を使用するように構成されている場合は、POST で送信されるユーザー名は単純な文字列です。しかし、サーバーが Active Directory を使用するように構成されている場合は、ドメイン名とユーザー名を含める必要があります (domain\username)。たとえば、username パラメーターは、次のようになります。

`username=dev\jsmith`このシナリオの一般的なエラー ログは、`"Invalid user: <username>"` です。

- **コンテンツのタイプが指定されている**

  ASP.NET または C# アプリケーションの設計を行っている場合、HTTP 要求でコンテンツのタイプを宣言する必要があります。例:

  ```
  http.setRequestHeader("Content-Type","application/x-www-form-urlencoded;charset=UTF-8")
  ```

  コンテンツ タイプを指定しておらず、Tableau Server から -1 が返される場合、ログ ファイルにはエラー `"missing username and/or client_ip"`.が含まれます。

HTTP 401 - 承認されていません

401- Not Authorized error というエラーが表示される場合は、次のいずれかの理由が考えられます。

- Active Directory で SSPI を使用するように Tableau Server を構成した可能性があります。Web サーバーで SSPI が使用されている場合、信頼できる認証を設定する必要はありません。ビューを埋め込むと、ユーザーは Tableau Server のライセンス ユーザーおよび Active Directory のメンバーであればアクセスできるようになります。詳細については、tsm authentication sspi <commands>を参照してください。

  注: SSPI は、Tableau Server for Windows でのみ構成できます。

- または、Tableau Server を展開した後に **302 - リダイレクト エラー**が表示された場合、クライアントの URL を構成するために記述した信頼できるチケット コードが、2 部構成のチケット URL 形式を考慮して更新されていない可能性があります。詳細については、Tableau Server からのチケットの取得を参照してください。

HTTP 404 - ファイルが見つかりません

存在しない Tableau Server URL をプログラム コードが参照したときに、このエラーが表示される可能性があります。Web ページが取得しようとしたときに検出できない無効な URL が Web サーバーによって構築された場合などです。

このエラーのもう1 つの原因は、「Tableau Server への信頼できるIP アドレスまたはホスト名の追加する」で説明しているように、信頼済み Web サーバーを入力しなかった場合です。信頼済み Web サーバーを既に入力している場合は、IP アドレスまたはホスト名が正しいことを確認してください。

無効なユーザー(SharePoint または C#)

Tableau Server で信頼できる認証を構成している場合に、このエラーが発生する可能性があります。

SharePoint .dll のサンプル コードは次の GET 要求を参照しています。

```
SPContext.Current.Web.CurrentUser.Name
```

上記の要求によって、現在の Windows Active Directory ユーザーの表示名が返されます。ログインID を使用する場合は、コードを次のように変更する必要があります。

```
SPContext.Current.Web.CurrentUser.LoginName
```

変更後、SharePoint .dll を再コンパイルします。

間違ったIP アドレスからチケットを引き換えようとしています

信頼できる認証を行うように Tableau Server を構成した場合、チケットの引き換えをする際に問題が発生する可能性があります。T

この問題は、Tableau Server がクライアントIP アドレスの一致を適用するように構成されている場合に発生することがあります。既定では、チケットを引き換えする際は、クライアントの Web ブラウザーの IP アドレスは考慮されないため、Tableau Server への POST 要求で送信されるクライアントの Web ブラウザーの IP アドレスが、ブラウザーが埋め込みビューを取得しようとするときのものと同じであることを確認してください。

たとえば、信頼できる認証の図で、ステップ 3 の POST 要求が `client_ip=74.125.19.147` というパラメーターを送信した場合、ステップ 5 の GET 要求は同じ IP アドレスから送信される必要があります。

クライアントIP アドレスの一致を適用するように Tableau Server を構成する方法についての詳細は、「オプション: クライアントIP 照合の設定」を参照してください。

Cookie 制限エラー

ユーザーが Tableau Server にサインインしたとき、セッション Cookie はローカルのブラウザーに保存されています。保存された Cookie は、ユーザーの Tableau Server へのサインイン認証とサーバーへのアクセス許可を管理する方法です。Cookie はブラウザーのアドレスバーと同じドメインまたはサブドメインで設定されているので、ファーストパーティの Cookie とみなされます。ユーザーのブラウザーがファーストパーティの Cookie をブロックするように設定されている場合、ユーザーは Tableau Server にサインインできません。

埋め込みビュー経由で、または信頼できる認証が設定された環境で、ユーザーが Tableau Server にサインインする場合も、同じことが起こります。cookie は保存されます。しかし、この場合、ブラウザーは Cookie をサードパーティCookie として扱います。これは、Cookie がブラウザーのアドレスバーに表示されているものと異なるドメインで設定されていることが理由です。ユーザーの Web ブラウザーがサードパーティの Cookie をブロックするように設定されている場合、Tableau Serverの認証は失敗します。失敗しないようにするには、Web ブラウザーがサードパーティの Cookie を許可するように構成する必要があります。

サーバーとの通信中にエラーが発生しました (403)

Tableau Server が信頼できる認証用に設定されている場合、ブラウザーで新しいビューを開き、以前に開いたビューに戻ろうとした後に上記のエラーが表示される可能性があります。

Tableau Server では、tsm configuration set のオプション`vizqlserver.protect_sessions` (既定では `true` に設定されています) を通じて、VizQL セッションが無許可で再使用されないようにするための保護を提供します。Tableau Server は信頼できる認証を行うように構成されているため、このレベルの保護は不要な場合があります。このオプションを無効にするには、tsm configuration set のオプションを使用して `vizqlserver.protect_sessions` を `false` に変更します。2024.2.0 以降、Tableau Server は常に、元のユーザーがサインアウトした後に VizQL セッションが再度使用されるのを防止します。

SCIM

SCIM (クロスドメイン アイデンティティ管理) は、ユーザーやグループのプロビジョニングの自動化を標準化するプロトコルであり、Microsoft Entra ID や Okta などのクラウドベースの ID プロバイダーと統合するために使用されます。

バージョン 2025.3 以降、Tableau Server は SCIM をサポートし、ID プロバイダー (IdP) が Tableau Server でユーザーとグループのメンバーシップの管理プロセスを合理化しながら、ユーザー アイデンティティを一元的に管理できるようにします。IdP は SCIM を使用して Tableau でユーザーのライフサイクルを管理し、Tableau Server は IdP のプロビジョニング割り当てと同期し続けます。このタイプの統合により、セキュリティが向上し、Tableau Server のサーバー管理者による手動作業が軽減されます。

Tableau Server の SCIM 機能はサイトレベルで動作し、サイト固有の SAML 認証をサポートするように設計されています。サイト固有の SAML 認証が構成されている場合、SCIM を通じてサイトにプロビジョニングされたユーザーは、サイト固有の SAML 認証を使用してセットアップされます。

Tableau Server との SCIM 統合の構成

ステップ 1: 前提条件を実行する

Tableau Server との SCIM 統合を有効にする前に、以下の要件を満たす必要があります。

- サーバー管理者が Tableau Server にアクセスできる。
- Tableau Server で IdP の SCIM 設定を修正できる。
- サイトで SAML 認証を有効にして構成している。詳細については、「サイト固有の SAML の構成」を参照してください。
- 外部トークンの生成と管理を使用している場合は、必要に応じて、Tableau 接続済みアプリを作成して有効にします。これを行っていない場合は、「Tableau 接続済みアプリを使用してアプリケーションを統合する」を参照してください。

注: Tableau Server のファイアウォールが原因で、SCIM 機能を動作させるために、ID プロバイダー (IdP) にオンプレミス コネクタを設定しなければならない場合があります。たとえば、Okta では、オンプレミス プロビジョニング (OPP) を設定する必要があります。Microsoft Entra ID では、プロビジョニング エージェントを設定する必要があります。

ステップ 2: サイトレベルの SCIM を構成する

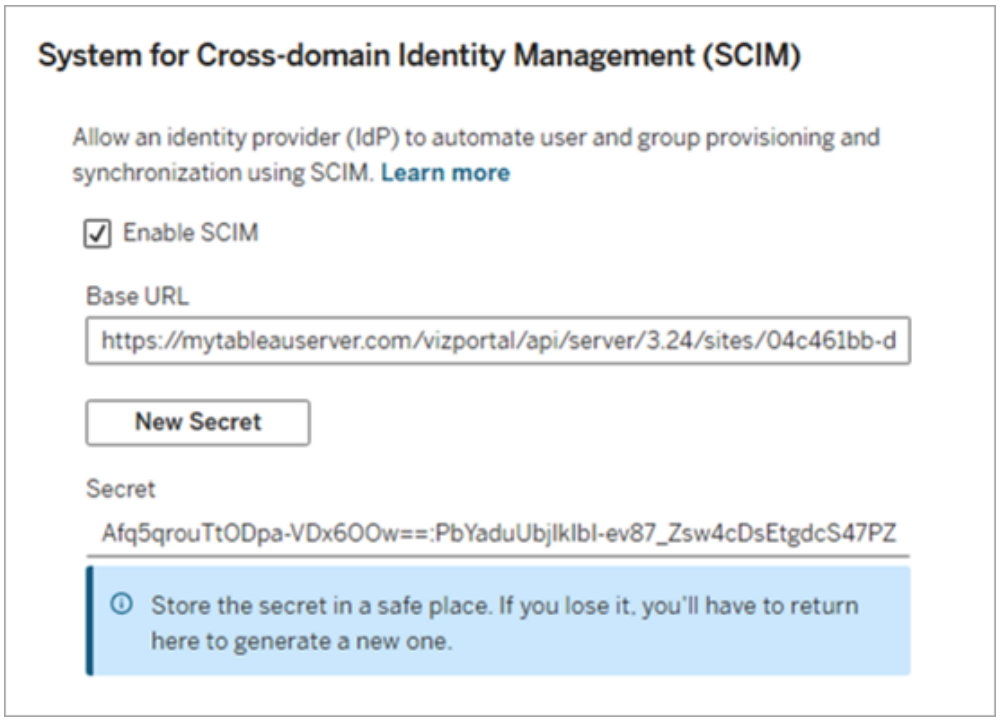
このセクションで説明する手順では、サイト固有の SAML 認証を構成する必要があります。

Tableau で生成された SCIM トークンを使用するオプションがあります。または、Tableau が生成する SCIM トークンをバイパスし、代わりに外部で生成された JWT を使用 (Tableau 接続済みアプリを使用) すると、SCIM 要求をサポートすることができます。

Tableau で生成されたトークンを使用して SCIM を有効にする

1. Tableau Server にサーバー管理者としてサインインします。
2. サイトに移動し、**[設定]** をクリックします。
3. 見出し **[SCIM (クロスドメイン アイデンティティ管理)]** で、**[SCIM の有効化]** チェックボックスをオンにします。これにより、**[ベース URL]** と **[新しいシークレット]** ボタンが設定されます。
4. 次の手順を実行します。
    a. **[ベース URL]** をコピーして、IdP の SCIM 設定で使用します。
    b. **[新しいシークレット]** ボタンをクリックします。
    c. シークレットをコピーして安全な場所に保管し、IdP の SCIM 設定で使用できるようにします。

    重要: シークレット トークンは、生成された直後にのみ表示されます。IdP に適用する前に見失ってしまった場合は、**[新しいシークレット]** を再度クリックできます。



d. [設定] ページの上部または下部にある **[保存]** ボタンをクリックします。

5. Tableau Server で SCIM を有効にしたら、IdP のドキュメントの手順に従って、ID プロバイダー (IdP) を使用した SCIM サポートを有効にします。

外部トークンを使用して SCIM を有効にする

外部トークンを使用するには、1) SCIM の外部トークン機能を有効にし、2) SCIM を有効にする必要があります。

ステップ 1: 外部トークンを有効にする

1. クラスターの初期ノード (TSM がインストールされているノード) で、サーバー管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを実行します。

a. `tsm configuration set -k features.JWTSupportForSCIM -v true`

b. `tsm pending-changes apply`

詳細については、「features.JWTSupportForSCIM」を参照してください。

ステップ 2: SCIM を有効にする

1. Tableau Server にサーバー管理者としてサインインします。
2. サイトに移動し、**[設定]** をクリックします。
3. 見出し **[SCIM (クロスドメイン アイデンティティ管理)]** で、**[SCIM の有効化]** チェックボックスをオンにします。これにより、**[ベース URL]** と **[新しいシークレット]** ボタンが設定されます。
4. 次の手順を実行します。
   a. **[ベース URL]** をコピーして、IdP の SCIM 設定で使用します。
   b. **[新しいシークレット]** ボタンは無視します。
   c. [設定] ページの上部または下部にある **[保存]** ボタンをクリックします。
5. Tableau Server で SCIM を有効にしたら、IdP のドキュメントの手順に従って、ID プロバイダー (IdP) を使用した SCIM サポートを有効にします。

ステップ 3: ユーザーとグループをプロビジョニングする

サイトで SCIM サポートを有効にした後、IdP のドキュメントに従って、ユーザーとグループをプロビジョニングします。

個人用アクセス トークン

個人用アクセス トークン (PAT) を使用すると、管理者と Tableau Server ユーザーが有効期間の長い認証トークンを作成できるようになります。PAT を使用すると、ハードコーディングされた認証資格情報 (ユーザー名とパスワード) や対話型のサインインを必要とすることなく、管理者とユーザーが Tableau REST API にサインインできるようになります。Tableau REST API での PAT の使

用の詳細については、Tableau REST API ヘルプの「サインインとサインアウト(認証)」を参照してください。

Tableau REST API を使用して作成した、自動化されたスクリプトとタスクでは、PAT を作成することをお勧めします。

- **セキュリティの強化**: 個人用アクセス トークンは、認証資格情報が漏えいした場合のリスクを軽減します。Tableau Server がアイデンティティ ストアとして Active Directory または LDAP を使用している場合、自動化されたタスクに個人用アクセス トークンを使用することで、認証資格情報が漏えいする影響を低減することができます。トークンが漏えいしたり、自動化で使用されてエラーやリスクが生じたりした場合、トークンを失効させるだけで済みます。ユーザーの認証資格情報を交換したり、取り消したりする必要はありません。

- **監査と追跡**: 管理者は、Tableau Server ログを確認し、トークンが使用された時期、そのトークンから作成されたセッション、それらのセッションで実行されたアクションを追跡することができます。また、セッションや関連タスクが、トークンから生成したセッションで実行されたか、インタラクティブなサインインから生成したセッションで実行されたかを確認することもできます。

- **自動化の管理**: トークンは実行されるスクリプトやタスクごとに作成できます。これにより、組織全体の自動化タスクをサイロに入れ、確認することができます。さらに、トークンを使用すると、ユーザー アカウントのパスワード リセットやメタデータの変更 (ユーザー名、メールなど) を行っても、認証資格情報がスクリプトにハードコードされている場合と同様、自動化が中断することはありません。

注:

- tabcmd で PAT を使用するには、https://tableau.github.io/tabcmd/ から互換バージョンの tabcmd をインストールします。
- PAT は、一般的なクライアントによる Tableau Server Web UI または TSM へのアクセスには使用されません。
- PAT の有効期限の設定や、UI から PAT を作成できるユーザー権限の無効化や制限は、Tableau Cloud でのみ行うことができます。
- PAT は、ユーザーの認証方法が変更されると自動的に取り消されます。

個人用アクセス トークンについて

作成された個人用アクセス トークン (PAT) は、ハッシュ化され、リポジトリに保存されます。PAT がハッシュ化されて保存されると、PAT シークレットはユーザーに一度だけ表示されますが、ユーザーがダイアログを閉じた後はアクセスできなくなります。そのため、ユーザーは、トークンを安全な場所にコピーし、パスワードと同様に扱うように指示されます。ランタイムに PAT を使用する場合、Tableau Server はユーザーから提示された PAT を、リポジトリに保存されているハッシュ値と比較します。一致すると、認証済みセッションが開始されます。

認可のコンテキストでは、PAT で認証された Tableau Server セッションには、PAT 所有者と同じアクセス権限が与えられます。

注: ユーザーは、1 つの PAT で複数の Tableau Server セッションを同時にリクエストできません。同じサイトであれ、別のサイトであれ、同じ PAT で再度サインインすると、前回のセッションが終了し、認証エラーが発生します。

サーバー管理者の偽装

バージョン 2021.1 以降、Tableau Server の PAT 偽装を有効にできます。このシナリオでは、Tableau REST API を使用する際に、サーバー管理者が作成した PAT をユーザーの偽装に使用できます。偽装は、アプリケーション内にエンドユーザー固有の Tableau コンテンツを埋め込むというシナリオで役に立ちます。具体的には、偽装 PAT を使用すると、認証資格情報をハードコーディングすることなく、特定のユーザーとしてクエリを実行し、そのユーザーが Tableau Server 内でアクセスできるコンテンツを取得するアプリケーションを構築できます。

詳細については、Tableau REST API ヘルプの「ユーザーの偽装」を参照してください。

偽装サインインの要求時に Tableau Server で個人用アクセス トークンが承認されるようにする

デフォルトでは、Tableau Server ではサーバー管理者の PAT 偽装は許可されていません。次のコマンドを実行して、サーバー全体で設定を有効にする必要があります。

```
tsm authentication pat-impersonation enable [global options]

tsm pending-changes apply
```

重要: コマンドを実行すると、サーバー管理者が作成したすべての PAT (既存の PAT を含む) が偽装に使用できるようになります。既存のすべてのサーバー管理者 PAT を一括して失効させるには `DELETE /api/{api-version}/auth/serverAdminAccessTokens` URI をポストします。詳細については、Tableau REST API ヘルプの「ユーザーの偽装」を参照してください。

個人用アクセス トークンの作成

ユーザーは、自分の PAT を作成する必要があります。管理者がユーザーの PAT を作成することはできません。

Tableau Server にアカウントを持つユーザーは、**[マイ アカウント設定]** ページで個人用アクセス トークン (PAT) の作成、管理、失効を行うことができます。詳細については、Tableau ユーザー ヘルプの「アカウント設定の管理」を参照してください。

注: ユーザーは最大 10 個の PAT を持つことができます。

個人アクセス トークンの有効期限の変更

個人用アクセス トークン (PAT) は、15 日間連続して使用されないと失効します。15 日ごとよりも頻繁に使用されても、PAT は 1 年で失効します。1 年後、新しい PAT を作成する必要があります。失効した PAT は、**[マイ アカウント設定]** ページに表示されません。

PAT の有効期限を変更するには、`tsm configuration set` コマンドの `refresh_token.absolute_expiry_in_seconds` オプションを使用します。

個人用アクセス トークンの失効

管理者は、ユーザーの PAT を失効させることもできます。ユーザーも、**[マイ アカウント設定]** ページで自分の個人用アクセス トークン (PAT) を失効させることができます。Tableau ユーザー ヘルプのトピック「アカウントの管理」で説明されている手順を使用します。

1. Tableau Server にサーバー管理者またはサイト管理者としてサインインします。
2. PAT を失効させるユーザーを見つけます。サーバー管理者ページの操作とユーザー検索の詳細については、「ユーザーの表示、管理、削除」を参照してください。
3. ユーザー名前をクリックし、プロフィール ページを開きます。

4. ユーザーのプロフィール ページで、**[設定]** タブをクリックします。
5. 失効させるトークンを **[個人用アクセス トークン]** セクションで特定し、**[失効]** をクリックします。
6. ダイアログ ボックスで **[削除]** をクリックします。

個人用アクセス トークンの使用状況の追跡と監視

個人用アクセス トークン (PAT) に関連したすべてのアクションは、Tableau Server のアプリケーション サーバー (vizportal) サービスに記録されます。PAT 関連のアクティビティを見つけるには、`RefreshTokenService` の文字列を含むログ エントリをフィルタリングします。

PAT は、`Token Guid: <TokenID(Guid)>` の形式で保存されます。TokenID は Base64 エンコード化された文字列です。シークレット値はログに含まれません。

例:

```
Token Guid: 49P+CxmARY6A2GHxyvHHAA== (e3d3fe0b-1980-458e-80d8-
61f1caf1c700)。
```

以下は 2 つのログ エントリのスニペット例です。最初のエントリは、ユーザーが PAT にどのようにマッピングされるかを示します。2 番目のエントリは、同じ PAT の更新イベントを示しています。

```
RefreshTokenService - Issued refresh token to the following user:
jsmith. Token Guid: 49P+CxmARY6A2GHxyvHHAA== (e3d3fe0b-1980-458e-
80d8-61f1caf1c700)
RefreshTokenService - Redeemed refresh token. Token Guid:
49P+CxmARY6A2GHxyvHHAA== (e3d3fe0b-1980-458e-80d8-61f1caf1c700)
```

主要な操作を見つけるには、`OAuthController` の文字列を含むログ エントリをフィルターします。

Tableau 接続済みアプリを使用してアプリケーションを統合する

Tableau Server 2022.1 以降、からは、Tableau 接続済みアプリを使用して、Tableau Server サイトとTableau コンテンツが埋め込まれている外部アプリケーションとの間に明示的な信頼関係を構築することにより、シームレスで安全な認証エクスペリエンスを実現できます。Tableau 接続済

みアプリは に REST API 認証をサポートするように機能が拡張されました。2023 年 10 月の時点では、接続済みアプリを介した REST API 認証は Tableau メタデータ API に対応しています。2025 年 2 月の時点では、接続済みアプリを介した REST API 認証は Tableau VizQL データサービスに対応しています。

注: Tableau 接続済みアプリと Salesforce 接続済みアプリは異なり、提供されている機能も異なります。現在、Tableau 接続済みアプリは、外部アプリケーションに Tableau ビューとメトリクスを埋め込むために最適化されており、Tableau REST API へのアクセスを承認するために使用されます。(メトリクスを埋め込む機能は、2023 年 10 月、Tableau Cloud および Tableau Server バージョン 2023.3 で廃止されました。)。

構成できる接続済みアプリには、直接信頼と OAuth 2.0 信頼の 2 種類があります。

直接信頼

*直接信頼*を使用すると、次のことができます。

- 埋め込み可能なコンテンツへのアクセス、およびそのコンテンツを埋め込みできる場所へのアクセスを制限する
- ユーザーは、アイデンティティプロバイダー (IdP) と統合しなくても、シングルサインオン (SSO) を使用して埋め込みコンテンツにアクセスできる
- ユーザーは、外部アプリケーションから直接認証できる
- JSON Web Token (JWT) を使用して、ユーザーに代わって Tableau REST API および Tableau Metadata API (Tableau Server 2023 年 10 月以降) へのアクセスをプログラムで承認する
- JSON Web Token (JWT) を使用して、ユーザーに代わって Tableau REST API および Tableau VizQL Data Service (Tableau Server 2025 年 2 月以降) へのアクセスをプログラムで承認する
- ユーザーまたはアプリケーションが実行できる Tableau REST API 機能のスコープを設定する

- 次のような追加機能を有効にします。
  - アサーションを使用した動的グループ メンバーシップ (Tableau Server 2024.2 以降)
  - ユーザー属性関数 (Tableau Server 2025.1 以降)

この接続済みアプリの種類の詳細については、「直接信頼を使用して接続済みアプリを設定する」を参照してください。

OAuth 2.0 信頼

*OAuth 2.0 信頼*を使用すると、次のことができます。

- 埋め込み可能なコンテンツへのアクセス、およびそのコンテンツを埋め込みできる場所へのアクセスを制限する
- ユーザーがアイデンティティプロバイダー (IdP) を介したシングルサインオン (SSO) を使用して埋め込みコンテンツにアクセスできるようにする
- OAuth 2.0 標準プロトコルを使用してアクセスを提供する
- Tableau REST API (および Tableau Server 2023 以降は *メタデータ* API) へのアクセスを、ユーザーに代わってプログラムで承認できる
- Tableau REST API および Tableau Server VizQL Data Service (2025 年 2 月以降) へのアクセスをユーザーに代わってプログラムで承認できる
- ユーザーまたはアプリケーションが実行できる Tableau REST API 機能のスコープを設定する
- 次のような追加機能を有効にします。
  - アサーションを使用した動的グループ メンバーシップ (Tableau Server 2024.2 以降)
  - ユーザー属性関数 (Tableau Server 2025.1 以降)

この接続済みアプリの種類の詳細については、「OAuth 2.0 信頼を使用して接続済みアプリを設定する」を参照してください。

直接信頼を使用して接続済みアプリを設定する

Tableau Server バージョン 2022.1 以降、Tableau に接続しているアプリによって Tableau Server サイトと外部アプリケーションとの間に明示的な信頼関係を構築することで、シームレスで安全な

認証エクスペリエンスを実現できるようになりました。

注:

- Tableau Server の UI なしの接続アプリ機能は、Tableau REST API の接続済みアプリのメソッドを介して Tableau Server バージョン 2021.4 で使用できるようになりました。
- 接続済みアプリを介した埋め込みを有効にするには、HTTP トラフィックに対して SSL を使用するように Tableau Server を構成する必要があります。
- セッション トークンを有効にするためには、外部アプリケーションと外部アプリケーションをホストするサーバーの時計が協定世界時 (UTC) に設定されている必要があります。いずれかの時計が異なる規格を使用している場合、接続済みアプリは信頼されません。

Tableau 接続済みアプリが直接的な信頼関係と連携するしくみ

Tableau Server サイトと外部アプリケーション間の信頼関係は、JSON Web Token (JWT) 標準の認証トークンによって確立および検証されます。この認証トークンは、Tableau 接続済みアプリによって提供され、外部アプリケーションによって署名される共有シークレットを使用します。

接続済みアプリの主要コンポーネント

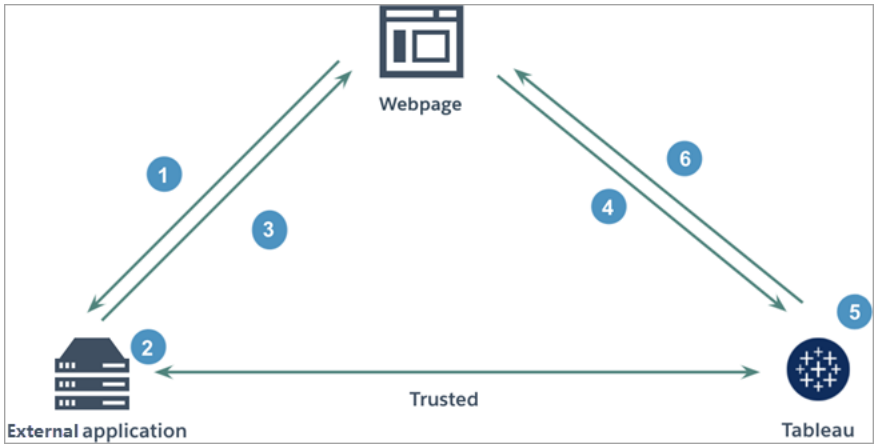
接続済みアプリの次のコンポーネントは、外部アプリケーションの JWT と連携して、ユーザーを認証し、埋め込みコンテンツを表示します。

- **シークレット:** シークレットは、Tableau と外部アプリケーションで共有されるキーです。これらは、JWT を形成する署名で使用されます。接続アプリを使用して認証または REST API 認証を埋め込む場合は、シークレットが必要です。シークレットは接続済みアプリで作成できます。シークレットの有効期限はなく、削除されるまで有効です。
- **ドメイン許可リスト (埋め込みのワークフローのみ):** 各接続済みアプリで許可されているドメインのリストを指定できます。接続済みアプリを介して埋め込まれた Tableau コンテンツは、指定されたドメインでのみ許可されます。これにより、ビジネスで保護および承認されたドメインでコンテンツを公開することができます。
- **アクセス レベル (埋め込みのワークフローのみ):** 接続済みアプリを 1 つのプロジェクトまたはすべてのプロジェクトに関連付けることにより、接続アプリを介して埋め込むことができるコンテンツを指定できます。1 つのプロジェクトを指定した場合、選択したプロジェクトのコンテンツのみを接続済みアプリを介して埋め込むことができます。複数のプロジェクトを指定する場合は、Tableau REST API を使用する必要があります。

接続済みアプリのワークフロー

埋め込みのワークフロー

次の図は、外部アプリケーション(Web サーバーとWeb ページ)とTableau 接続済みアプリ間の認証のしくみを示しています。



1. **ユーザーが Web ページにアクセスします:** ユーザーが Web ページに埋め込まれたコンテンツにアクセスすると、Web ページが外部アプリケーションに GET 要求を送信し、その Web ページの HTML を取得します。
2. **外部アプリケーションが認証トークンを構築します:** 外部アプリケーションが JWT を作成します。この JWT には、接続アプリからのシークレット(追加の JWT 要件については以下のステップ 3 を参照)と、埋め込みコンテンツに対するユーザー アクセスの範囲が含まれます。シークレットは外部アプリケーションによって署名され、後のステップで信頼関係の検証に使用されます。
3. **外部アプリケーションが認証トークンを使用して応答します:** 外部アプリケーションが、Web ページによって呼び出された埋め込みコンテンツの URL に JWT が含まれているページに応答します。
4. **Web ページが Tableau のコンテンツを要求します:** 埋め込みコンテンツを読み込むために、Web ページが埋め込みコンテンツの URL を呼び出し、Tableau に GET 要求が送信されます。
5. **Tableau がトークンを検証します:** Tableau が JWT を受信し、接続済みアプリと JWT で使用される共有シークレットを識別して、外部アプリケーションとの信頼関係を検証します。次に、Tableau はユーザーのセッションを作成します。このセッションでは、JWT で定義

された埋め込み範囲だけでなく、許可されたドメインやプロジェクトなど、接続済みアプリで指定された制限も考慮されます。

6. **Tableau が、埋め込みの制限付きコンテキストに基づいてコンテンツを返します:** 埋め込みコンテンツは、ページが許可されたドメインの下にあり、かつ、コンテンツが許可されたプロジェクトにパブリッシュされている場合にのみ読み込まれます (該当する場合)。認証されたユーザーは、JWT で定義された範囲で埋め込みコンテンツを操作できます。

接続済みアプリの作成

ステップ1: 接続済みアプリを作成する

Tableau Server の [設定] ページから接続済みアプリを作成します。

1. サーバー管理者として、Tableau Server にサインインします。
2. 左側のペインから、**[設定] > [接続済みアプリ]** の順に選択します。
3. [新しい接続済みアプリ] ボタンのドロップダウン矢印を選択して、**[直接信頼]** を選択します。

   注: Tableau Server 2023.3 以前を使用している場合は、**[新しい接続済みアプリ]** ボタンを選択します。

4. [接続済みアプリの作成] ダイアログ ボックスで、次のいずれかの手順を行います。
   - *REST API 認可ワークフローの場合 (認証に REST API を使用するメタデータ API ワークフローを含みます)*、[接続済みアプリ名] テキスト ボックスに接続済みアプリ名を入力し、**[作成]** ボタンを選択します。

     注: REST API とメタデータ API の認可用に接続済みアプリを構成する場合は、**アクセス レベル**と**ドメイン許可リスト**を無視できます。

   - *埋め込みのワークフロー*の場合、次の手順を実行します。
     i. [接続済みアプリ名] テキスト ボックスに、接続済みアプリ名を入力します。
     ii. [適用先] ドロップダウン メニューから、**[すべてのプロジェクト]** または **[1 つのプロジェクトのみ]** を選択して、埋め込むことができるビューまたはメトリクスを制御します。[1 つのプロジェクトのみ] オプションを選択した場合は、スコープを設定する特定のプロジェクトを選択します。これらの 2 つのオプションの詳細については、「アクセス レベル (埋め込みのワークフローのみ)」を参照してください。

        注:

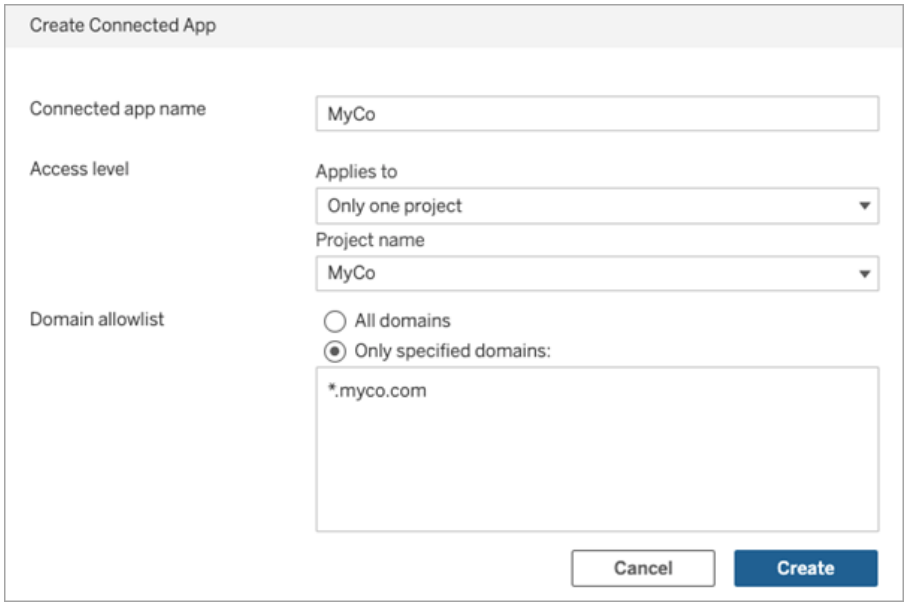
        - メトリクスを埋め込む機能は、Tableau Server 2023.3 で廃止されました。

- Tableau Server 2024.2 以降では、Tableau REST API を使用して複数のプロジェクトを指定できるようになりました。詳細については、REST API ヘルプの「接続済みアプリ」と「接続済みアプリの更新」のメソッドを参照してください。

iii. ドメイン許可リストで、以下の「ドメインの書式設定」に記載されているルールを使用してドメインを指定し、ビューまたはメトリクスを埋め込むことができる場所を制御します。

重要: Tableau コンテンツが許可した場所にのみ埋め込まれるように、セキュリティのベストプラクティスとしてドメイン許可リストを使用することをお勧めします。

iv. 完了したら、**[作成]** ボタンを選択します。



5. 接続済みアプリ名の横にあるアクション メニューを選択し、**[有効]** を選択します。セキュリティ上の理由から、接続済みアプリは作成時に既定で無効に設定されています。

6. 接続済みアプリID (クライアントID とも呼ばれます) をメモして、以下のステップ 3 で使用します。

ステップ 2: シークレットを生成する

接続済みアプリごとに合計 2 つのシークレットを生成できます。2 番目のシークレットは、シークレットのローテーションの目的で使用し、シークレットが侵害された場合の問題からシークレットを保護することができます。

1. ステップ 1 で作成した接続済みアプリの詳細ページで、**[新しいシークレットの生成]** ボタンを選択します。
2. 以下のステップ 3 で使用するシークレット ID とシークレット値をメモします。

ステップ 3: JWT を構成する

シークレットを生成したら、外部アプリケーションから有効な JWT を送信できるようにします。JWT は、2 者間で情報を安全に転送するために使用される標準規格です。JWT は、Tableau Server に情報を安全に送信するために、外部アプリケーションによって署名されています。JWT は、接続済みアプリ、セッションが生成されるユーザー、およびユーザーが持つ必要があるアクセス レベルを参照します。

有効な JWT には、次の情報が含まれます。

- ステップ 1 の接続済みアプリID (クライアントID とも呼ばれます)
- ステップ 2 で生成されたシークレットID とシークレット値
- 登録されたクレームとヘッダー:

クレーム	名前	説明または必要な値
"kid"	シークレットID	必須 (ヘッダー内)接続済みアプリのシークレットキー識別子
"iss"	発行者	必須 (ヘッダー内)信頼できる接続済みアプリとその署名キーを識別する一意の発行元 URI
"alg"	アルゴリズム	必須 (ヘッダー内)JWT 署名アルゴリズム。HS256 のみがサポートされています。
"sub"	件名	認証された Tableau Server ユーザーのユーザー名
"aud"	オーディエンス	値は "tableau" である必要があります。
"exp"	有効期限	JWT の有効期限が切れないようにする必要があります。JWT の有効期限 (UTC) は、設定された最大有効期間の範囲内である必要があります。この最大有効期間は、Tableau サービス マネージャーの vizportal.oauth.connected_apps.max_expiration_period_in_minutes コマンドを使用して構成できます。

"jti"	JWT ID	クレームとして必要です。JWT ID クレームは、JWT に一意の識別子を提供し、大文字と小文字を区別します。
"scp" **重要:** "scope" は使用しないでください。	範囲	*埋め込みのワークフロー*の場合、サポートされる値は次のとおりです。 "tableau:views:embed" "tableau:views:embed_authoring"Tableau Server 2022.3 で追加 "tableau:metrics:embed" (Tableau Server 2023.3 で廃止) "tableau:ask_data:embed" (Tableau Server 2023.1 で廃止) **注:** • 値はリスト型として渡す必要があります。 • tableau:views:embed と tableau:views:embed_authoring の場合、この範囲は、Tableau Server で既に構成されているユーザーのパーミッションを考慮し、元のビューで使用可能な場合は、ユーザーが埋め込みビューのツールを操作できるようにします。 • 埋め込みコードでは、ツールバー パラメーターを除外することをお勧めします。詳細については、以下の「既知の問題 (埋め込みのワークフローのみ)」を参照してください。

		*REST API 承認 ワークフロー*の場合は、「JWT 認可をサポートする REST API メソッド」を参照してください。 *認証に REST API を使用するメタデータ API ワークフロー*の場合、サポートしているスコープは `tableau:content:read` のみです。 *認証に REST API 使用する VizQL データ サービス ワークフロー*の場合、サポートされるスコープは `tableau:viz_data_service:read` のみです。
`https://tableau.com/g-roups`	動的グループ メンバーシップ	*埋め込みのワークフロー*のみ。 値は、Tableau Server の 1 つまたは複数のグループ名に一致する必要があります。詳細については、以下のセクション「動的グループ メンバーシップ (埋め込みワークフローのみ)」を参照してください。
(ユーザー属性)	(ユーザー属性値)	*埋め込みのワークフロー*のみ。 Tableau Server 2025.1 以降では、ユーザー属性を含めることができます。埋め込みコンテンツでユーザー属性関数が使用されると、Tableau は認証されたユーザーのコンテキストをチェックし、ランタイムに表示できるデータを決定します。

		注: • ワークフローに影響を与える可能性のある*既知の問題と制約*については、Tableau 埋め込み API v3 のヘルプを確認してください。 • ユーザー属性を機能させるためには、1) Tableau REST API を使用して、サイトの埋め込み設定を更新するメソッドによってユーザー属性機能を有効にし、必要に応じて、features.VizDataServiceClientEnable TSM オプションを有効にし、2) コンテンツ作成者がユーザー属性関数を作成する必要があります。 • ユーザー属性は、大文字と小文字を区別します。

JWT の例

以下に、Java 言語とPython 言語の両方の JWT の例を示します。Java とPython の例では、それぞれ nimbus-jose-jwt ライブラリとPyJWT ライブラリを使用しています。

Java

```
import com.nimbusds.jose.*;
import com.nimbusds.jose.crypto.*;
import com.nimbusds.jwt.*;

import java.util.*;
```

```
...

String secret = "secretvalue";
       String kid = "connectedAppSecretId";
       String clientId = "connectedAppClientId";
       List<String> scopes = new
ArrayList<>(Arrays.asList("tableau:views:embed"));
       String username = "username";
       JWSSigner signer = new MACSigner(secret);
       JWSHeader header = new
JWSHeader.Builder(JWSAlgorithm.HS256).keyID(kid).customParam("iss",
clientId).build();
       JWTClaimsSet claimsSet = new JWTClaimsSet.Builder()
               .issuer(clientId)
               .expirationTime(new Date(new Date().getTime() + 60 * 1000))
//expires in 1 minute
               .jwtID(UUID.randomUUID().toString())
               .audience("tableau")
               .subject("username")
               .claim("scp", scopes)


               .claim("Region", "East")
               .build();
       SignedJWT signedJWT = new SignedJWT(header, claimsSet);
       signedJWT.sign(signer);
       model.addAttribute("token", signedJWT.serialize());
```

Python

```
import jwt

token = jwt.encode(
       {
               "iss": connectedAppClientId,
               "exp": datetime.datetime.utcnow() + datetime.timedelta
```

```
    (minutes=5),
                    "jti": str(uuid.uuid4()),
                    "aud": "tableau",
                    "sub": user,
                    "scp": ["tableau:views:embed", "tableau:metrics:embed"]

            },
                    connectedAppSecretKey,
                    algorithm = "HS256",
                    headers = {
                    'kid': connectedAppSecretId,
                    'iss': connectedAppClientId
             }
      )
```

JWT を構成した後、外部アプリケーションでコードを実行すると、トークンが生成されます。

ステップ 4: 次のステップ

埋め込みのワークフローの場合

JWT を構成したら、埋め込みコードを外部アプリケーションに追加する必要があります。外部アプリケーションが呼び出す Web コンポーネントに、上記のステップ 3 で構成した有効な JWT が含まれていることを確認します。

Tableau コンテンツの埋め込みの詳細については、次のいずれかまたは両方を参照してください。

- メトリクスの埋め込みについては、Tableau ヘルプの「Web ページへのメトリクスの埋め込み」トピックを参照してください。(Tableau Server 2023.3 で、Tableau はメトリクスを埋め込む機能を廃止しました。)
- Tableau API v3 の埋め込み を使用して、Tableau のビューとメトリクスを埋め込みます。

注: 埋め込みコンテンツにアクセスするユーザーを正常に認証するには、ブラウザーがサードパーティーの Cookie を許可するように設定されている必要があります。

REST API 認証ワークフローの場合

JWT を構成したら、認証アクセスのために、有効な JWT を REST API サインイン要求に追加する必要があります。詳細については、「接続済みアプリのアクセス スコープ」を参照してください。

メタデータ API ワークフローの場合

JWT を構成したら、有効な JWT を REST API サインイン要求に追加する必要があります。詳細については、「接続済みアプリのアクセス スコープ」を参照してください。

接続済みアプリの管理

[接続済みアプリ] ページでは、サイトのすべての接続済みアプリを管理できます。接続済みアプリの作成、削除、無効化などのタスクを実行できるほか、既存のシークレットが侵害されている場合は、新しいシークレットを取り消したり、生成したりすることができます。

1. サーバーTableau Server にサインインします。
2. 左側のペインから、**[設定] > [接続済みアプリ]** の順に選択します。
3. 管理する接続済みアプリの横にあるチェック ボックスを選択し、次のいずれか 1 つまたは複数の操作を実行します。
    - 組織のセキュリティポリシーで指定されたローテーション タイムラインに従って、**新しいシークレットを生成**します。追加のシークレットを生成するには、接続済みアプリの名前をクリックしてから、**[Generate New Secret (新しいシークレットの生成)]** ボタンをクリックします。接続済みアプリは、最大 2 つのシークレットを持つことができます。両方のシークレットは同時にアクティブにすることができ、有効期限はなく、削除されるまで有効です。
    - 接続済みアプリの名前を選択して**接続済みアプリの詳細**を表示し、接続済みアプリの作成日、ID、プロジェクトとドメインの範囲、およびシークレットを確認します。
    - **プロジェクトの範囲またはドメインを変更**し、[アクション] メニューで **[編集]** を選択します。変更を加えて、**[更新]** を選択します。

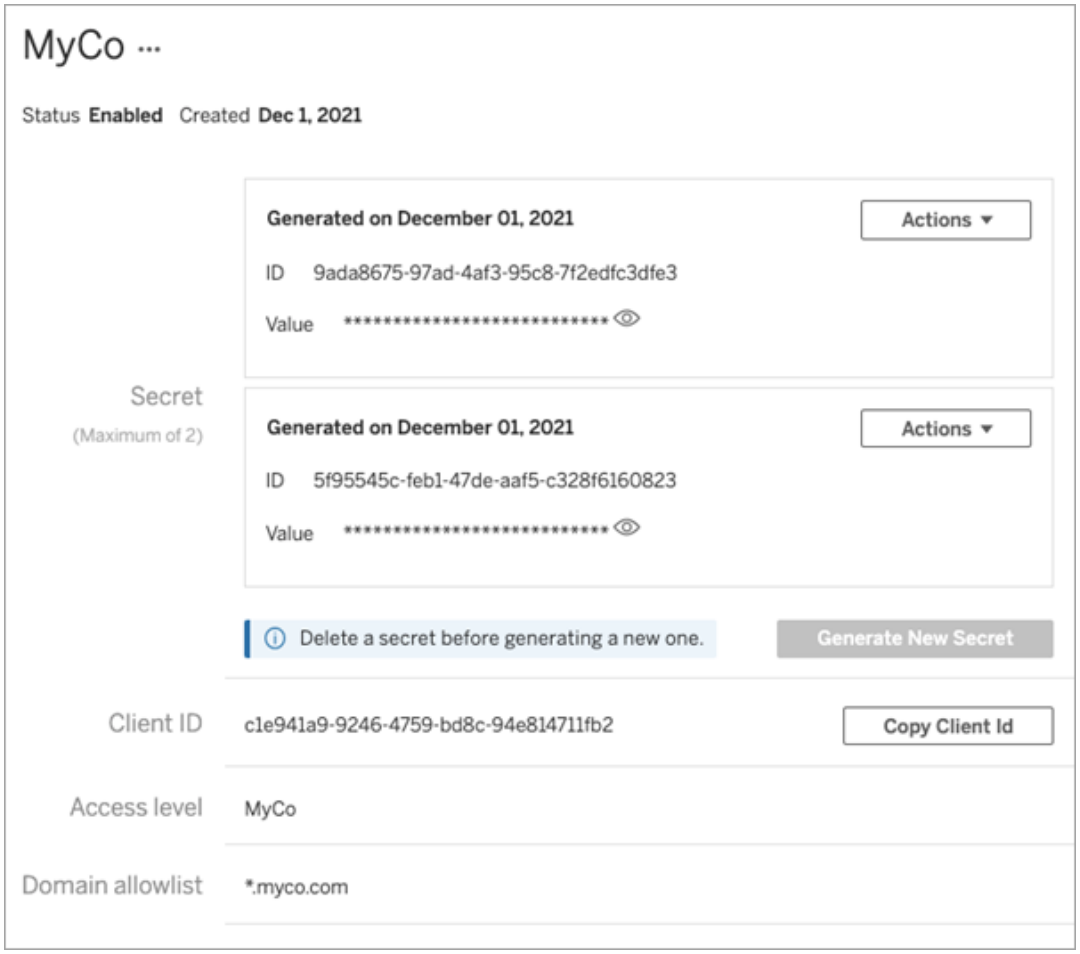
      注: プロジェクトまたはドメインの範囲を変更していて、埋め込みコンテンツが新しいプロジェクトまたは新しいドメインのいずれにも存在しない場合は、埋め込みビューまたはメトリクスを表示することができなくなるため、ユーザーが埋め込みコンテンツにアクセスするとエラーが表示されます。

    - 接続済みアプリの名前を選択して、**シークレットを削除**します。接続済みアプリのページで、シークレットの横にある **[アクション]** をクリックし、**[削除]** を選択します。確

認ダイアログ ボックスで、もう一度 **[削除]** を選択します。

注: 接続済みアプリのシークレットが外部アプリケーションで使用されている場合、シークレットを削除すると、埋め込みビューやメトリクスを表示できなくなります。詳細については、以下の「接続済みアプリの無効化と削除、およびシークレットの削除による影響」を参照してください。

- **接続済みアプリを無効**にし、[アクション] メニューで **[無効]** を選択します。接続済みアプリが外部アプリケーションで使用されている場合、接続済みアプリを無効にすると、埋め込みビューやメトリクスを表示できなくなります。詳細については、以下の「接続済みアプリの無効化と削除、およびシークレットの削除による影響」を参照してください。



接続済みアプリの無効化と削除、およびシークレットの削除による影響

接続済みアプリを介してユーザーに埋め込みコンテンツを表示したり、REST API アクセスを有効にしたりするには、接続済みアプリを有効にして、そのシークレットを生成する必要があります。接続済みアプリが外部アプリケーションで使用されていて、無効化または削除されているか、そのシークレットが削除または置換されている場合、403 エラーが発生します。

この問題を回避するには、接続済みアプリが有効になっていて、JWT で正しいシークレット ID と値が使用されていることを確認します。

アクセス レベル (埋め込みのワークフローのみ)

接続済みアプリのアクセス レベルを構成する際は、次の 2 つのプロジェクト タイプのうちいずれかを選択できます。アクセス レベルは、どのコンテンツを埋め込むことができるかを制御します。

- **すべてのプロジェクト:** このオプションを使用すると、すべてのプロジェクトのコンテンツを埋め込むことができます
- **1 つのプロジェクトのみ:** このオプションを使用すると、指定されたプロジェクト内のコンテンツのみを埋め込むことができます。指定したプロジェクトにネストされたプロジェクトが含まれている場合、そのネストされたプロジェクトにコンテンツを埋め込むことはできません。

複数のプロジェクトについて

Tableau Server 2024.2 以降、Tableau REST API のみを使用して接続済みアプリの複数のプロジェクトでコンテンツを有効にすることができます。プロジェクトを指定するには、接続済みアプリの作成、または接続済みアプリの更新のいずれかのメソッドで「プロジェクト ID」を使用します。

注: 接続済みアプリに複数のプロジェクトが構成されている場合、Tableau は接続済みアプリのアクセス レベルに対応する**複数のプロジェクト**を表示します。**[1 つのプロジェクトのみ]** または **[すべてのプロジェクト]** を選択して接続済みアプリを更新すると、[Multiple projects (複数のプロジェクト)] オプションは表示されなくなります。複数のプロジェクトに対して接続済みアプリをもう一度構成する必要がある場合は、REST API を使用する必要があります。

ドメイン許可リストルール (埋め込みのワークフローのみ)

接続済みアプリのドメイン許可リストを使用すると、埋め込まれた Tableau コンテンツへのアクセスをすべてのドメインまたは一部のドメインに制限するか、一部のドメインを除外するか、またはすべてのドメインをブロックすることができます。

重要: Tableau コンテンツが許可した場所にのみ埋め込まれるように、セキュリティのベストプラクティスとしてドメイン許可リストを使用することをお勧めします。

ドメイン オプション

接続済みアプリのドメイン許可リストを構成する際は、次の 2 つのオプションのいずれかを選択できます。

- **すべてのドメイン:** 既定のオプションとして、このオプションは埋め込みコンテンツへの無制限のアクセスを有効にします。
- **特定のドメインのみ:** このオプションを使用すると、埋め込みコンテンツへのアクセスの範囲を限定できます。このオプションを使用する場合は、次のセクション「ドメインの書式設定」指定されている書式設定ルールに従ってください。

ドメインの書式設定

[domain allowlist (ドメイン許可リスト)] テキストボックスには、以下の書式設定の例を使用して、1 つまたは複数のドメインを入力することができます。

注: ドメインの書式設定のルールは、Tableau REST API で 接続済みアプリのメソッドを使用する場合にも適用されます。

一般的なシナリオに基づく書式設定の例をいくつか示します。

指定する内容	例	埋め込みへのアクセス
ドメインの範囲	*.myco.com	埋め込みコンテンツには、myco.com のすべてのサブドメインからアクセスできます。
すべてのポート	myco.com:*	埋め込みコンテンツには、myco.com のすべてのポートからアクセスできます。
特定のポート	myco.com:8080	埋め込みコンテンツには、myco.com のポー

		ト8080 からのみアクセスできます。
複数の個別のドメイン	myco.com events.myco.com ops.myco.com	埋め込みコンテンツには、3 つのすべてのドメインからアクセスできます。 **注:** 複数のドメインを指定するときは、各ドメインを新しい行に入力するか、ドメインをスペースで区切ってください。REST API の場合は、ドメインをスペースで区切る必要があります。
セキュアなトラフィックのみ	https:	埋め込みコンテンツには、ドメインに関係なく、安全にアクセスできます。
ドメインの範囲のすべてのポートへのセキュアなトラフィック	https:*myco.com:*	埋め込みコンテンツには、myco.com のすべてのサブドメインのすべてのポートから安全にアクセスできます。
ドメインなし	[ドメインなし]	埋め込みコンテンツへのアクセスはブロックされます。

動的グループ メンバーシップ (埋め込みワークフローのみ)

Tableau Server 2024.2 以降、接続済みアプリが設定されて機能が有効になっている場合、外部アプリケーションから送信した JWT に含まれるカスタム クレームを通じてグループ メンバーシップを動的に制御できます。

設定すると、ユーザー認証の際、外部アプリケーションは、ユーザーをアサートするグループ (`https://tableau.com/groups`) とグループ名 (「Group1」と「Group2」など) という、2 つのカスタム グループ メンバーシップ クレームを含む JWT を送信します。Tableau は JWT を検証し、グループと、そのグループにパーミッションが依存するコンテンツへのアクセスを有効にします。

詳細については、アサーションを使用した動的グループ メンバーシップを参照してください。

既知の問題 (埋め込みのワークフローのみ)

接続済みアプリの使用にはいくつかの既知の問題がありますが、これらは今後のリリースで対処される予定です。

- **ツールバー機能:** 埋め込みコンテンツにツールバーのパラメーターが定義されている場合、すべてのツールバー機能が動作するとは限りません。この問題を回避するには、次の例のようにツールバーのパラメーターを非表示にすることをお勧めします。

  ```
  <tableau-viz id='tab-viz' src='https://<your_server>/t/<your_
  site>/...'
         toolbar='hidden'>
  </tableau-viz>
  ```

- **パブリッシュされたデータ ソース:** データベースの認証資格情報の入力を求める **[ユーザーにメッセージを表示]** が設定されているパブリッシュされたデータ ソースは表示されません。この問題を回避するには、可能であれば、データ ソースの所有者がデータベースの認証資格情報を埋め込むことをお勧めします。
- **複数のサイトへの埋め込みビュー:** Tableau Server 2023.1 以前では、同じブラウザーで異なるサイトのビューに切り替えると、エラー**「1008: 接続アプリのシークレットを取得できませんでした」**が発生します。この問題を回避するには、Tableau Server 2023.3 以降にアップグレードしてください。

- **埋め込みダッシュボードの「データに聞く」オブジェクト**: 埋め込みダッシュボードの「データに聞く」オブジェクトは読み込まれません。(Tableau Server 2024.2 では「データに聞く」は廃止されます。)

- **メトリクスとドメイン許可リスト:** 接続済みアプリのドメイン許可リストでアクセス制限が指定されているかどうかにかかわらず、埋め込みメトリクスのビューが表示されます。**注:** 埋め込みビューのツールバーからアクセスされるメトリクス データは、期待どおりに機能します。(Tableau Server 2023.3 で、Tableau はメトリクスを埋め込む機能を廃止しました。)

トラブルシューティング

接続済みアプリに関連する可能性のあるエラーと、推奨されるトラブルシューティングのステップについては、「接続済みアプリをトラブルシューティングする - 直接信頼」を参照してください。

OAuth 2.0 信頼を使用して接続済みアプリを設定する

Tableau Server 管理者は、1 つまたは複数の外部認証サーバー (EAS) を登録し、OAuth 2.0 標準プロトコルを使用して Tableau Server と EAS の間に信頼関係を確立できます。

重要:

- このトピックの一部の手順では、サードパーティーのソフトウェアとサービスを使用して構成する必要があります。Tableau Server で EAS 機能を有効にする手順については最善を尽くして検証していますが、サードパーティのソフトウェアとサービスが変更されたり、ユーザーの組織の方針が異なる場合があります。問題が発生した場合、正式な構成の詳細情報とサポートについては、サードパーティーのドキュメントを参照してください。
- EAS を介した埋め込みを有効にするには、HTTP トラフィックに対して SSL を使用するように Tableau Server を構成する必要があります。
- セッション トークンを有効にするためには、外部アプリケーションと外部アプリケーションをホストするサーバーの時計が協定世界時 (UTC) に設定されている必要があります。いずれかの時計が異なる規格を使用している場合、接続済みアプリは信頼されません。

Tableau 接続済みアプリが OAuth 2.0 信頼と連携するしくみ

Tableau Server サイトと外部アプリケーション間の信頼関係は、JSON Web Token (JWT) 標準の認証トークンを通じて確立および検証されます。

埋め込まれた Tableau コンテンツが外部アプリケーションに読み込まれるときは、認証コードフローの OAuth フローが使用されます。ユーザーが IdP に正常にサインインすると、Tableau Server に自動的にサインインします。以下に説明するステップに従って、EAS を Tableau Server に登録します。

接続済みアプリの主要コンポーネント

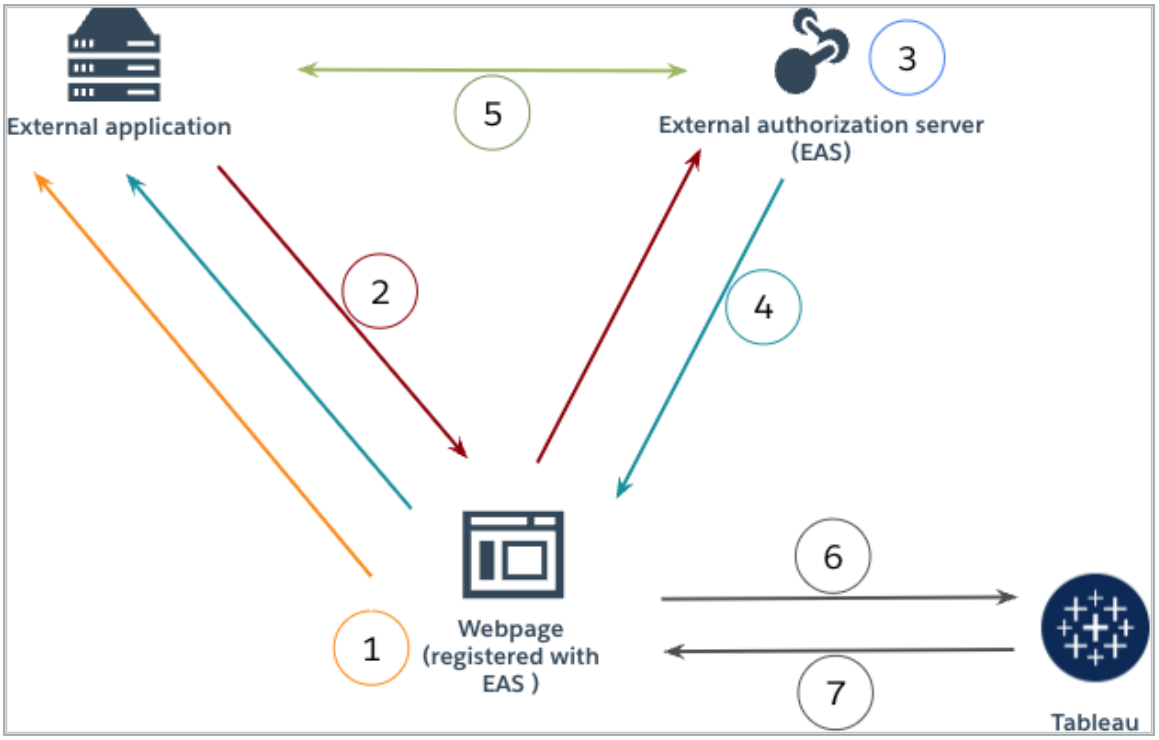
接続済みアプリの次のコンポーネントは、外部アプリケーションの JWT と連携して、ユーザーを認証し、埋め込みコンテンツを表示します。

- **外部認証サーバー (EAS):** ユーザーと外部アプリケーション間のインターフェイスとして機能するサーバー (通常は IdP)。サーバーは、保護された Tableau コンテンツへのユーザー アクセスを認証および承認します。
- **発行者 URL:** EAS インスタンスを一意に識別する URL。

接続済みアプリのワークフロー

埋め込みのワークフロー

次の図は、外部認証サーバー (EAS)、外部アプリケーション (Web サーバーと Web ページ)、Tableau 接続済みアプリ間の認証のしくみを示しています。



1. **ユーザーが Web ページにアクセスします:** ユーザーが Web ページに埋め込まれたコンテンツにアクセスすると、Web ページが外部アプリケーションに GET 要求を送信します。

2. **外部アプリケーションがリクエストを EAS にリダイレクトします:** 外部アプリケーションが、外部認証サーバー (EAS) にリダイレクトする Web ページを使用して応答します。
3. **ユーザーが EAS で認証します:** ユーザーが EAS を使用して認証および承認を行います。
4. **EAS が認証コードを使用して Web ページに応答します:** EAS が認証コードを使用してページに応答し、Web ページにリダイレクトします。
5. **EAS が認証コードを JWT に変換します:** Web ページが EAS を呼び出して認証コードを JWT に変換し、それを埋め込みコンテンツの URL に挿入します。
6. **Web ページが Tableau にコンテンツを要求します:** Web ページが iFrame を読み込み、Tableau に GET リクエストを送信します。
7. **Tableau がトークンを検証します:** Tableau が署名を使用して URL 内の JWT を検証し、コンテンツの要求に応答し、JWT で定義された埋め込みスコープを考慮します。

接続済みアプリを作成する

ステップ 1: 始める前に

EAS を Tableau Server に登録するには、EAS がすでに構成されている必要があります。さらに、EAS は、以下の表にリストされている登録済みのクレームとヘッダーを含む有効な JSON Web Token (JWT) を送信する必要があります。

クレーム	名前	説明または必要な値
"kid"	キー ID	必須 (ヘッダー内)アイデンティティプロバイダーからの一意のキー識別子
"iss"	発行者	必須 (ヘッダー内またはクレームとして)信頼できる接続済みアプリとその署名キーを識別する一意の発行元 URI
"alg"	アルゴリズム	必須 (ヘッダー内)JWT 署名アルゴリズム。サポートしているアルゴリズム名の一覧は、javadoc.io ドキュメントの「Class JWSAlgorithm」ページにあります。この

		署名アルゴリズムは、vizportal.oauth.external_authorization_server.blocklisted_jws_algorithmsのコマンドを使用して設定できます。
"sub"	件名	認証された Tableau Server ユーザーのユーザー名
"aud"	オーディエンス	値は "tableau" である必要があります。
"exp"	有効期限	JWT の有効期限が切れないようにする必要があります。JWT の有効期限 (UTC) は、設定された最大有効期間の範囲内である必要があります。最大有効期間は、vizportal.oauth.external_authorization_server.max_expiration_period_in_minutes コマンドを使用して構成できます。
"jti"	JWT ID	JWT ID クレームは、JWT に一意の識別子を提供し、大文字と小文字を区別します。
"scp"	スコープ	*埋め込みのワークフロー*の場合、サポートされる値は次のとおりです。 "tableau:views:embed" "tableau:views:embed_authoring" (Tableau Server 2022.3 で追加) "tableau:metrics:embed" (Tableau Server 2023.3 で廃止) "tableau:ask_data:embed" (Tableau Server 2023.1 で追加。

		Tableau Server 2024.2 で廃止予定) **注:** • 値はリスト型として渡す必要があります。 • `tableau:views:embed` の場合、この範囲は、Tableau Server で既に構成されているユーザーのパーミッションを考慮し、元のビューで使用可能な場合は、ユーザーが埋め込みビューのツールを操作できるようにします。 • 埋め込みコードでは、ツールバーパラメーターを除外することをお勧めします。詳細については、以下の既知の問題 (埋め込みのワークフローのみ)を参照してください。 *REST API 承認ワークフロー*の場合は、「JWT 認可をサポートする REST API メソッド」を参照してください。 *認証に REST API を使用するメタデータ API ワークフロー*の場合、サポートしているスコープは `tableau:content:read` のみです。 *認証に REST API 使用する VizQL データサービスワークフロー*の場合、サポートされるスコープは `tableau:viz_data_service:read` のみです。
`https://tableau.com/groups`	動的グループ	*埋め込みのワークフロー*のみ。 値は、Tableau Server の 1 つまたは複

	プメンバーシップ	数のグループ名に一致する必要があります。詳細については、以下のセクション「動的グループ メンバーシップ (埋め込みワークフローのみ)」を参照してください。
(ユーザー属性)	(ユーザー属性値)	*埋め込みのワークフローのみ。* Tableau Server 2025.1 以降では、ユーザー属性を含めることができます。埋め込みコンテンツでユーザー属性関数が使用されると、Tableau は認証されたユーザーのコンテキストをチェックし、ランタイムに表示できるデータを決定します。 **注:** • ワークフローに影響を与える可能性のある*既知の問題と制約*については、Tableau 埋め込み API v3 のヘルプを確認してください。 • ユーザー属性を機能させるためには、1) Tableau REST API を使用して、サイトの埋め込み設定を更新するメソッドによってユーザー属性機能を有効にし、必要に応じて、features.VizDataServiceClientEnable TSM オプションを有効にし、2) コンテンツ作成者がユーザー属性関数を作成する必要があります。 • ユーザー属性は、大文字と小文字を区別します。

注: 上記の JWT クレームは、インターネット技術標準化委員会 (IETF) によって配布されたドキュメントの「Registered Claim Names」セクションに文書化されています。

ステップ 2: EAS を Tableau Server に登録する

EAS を Tableau Server に登録することにより、EAS と Tableau Server の間に信頼関係を確立します。これは、ユーザーが外部アプリケーションに埋め込まれた Tableau コンテンツにアクセスすると、IdP で認証するようにリダイレクトされることを意味します。EAS は認証トークンを生成して、検証のために Tableau Server に渡します。信頼関係が検証されると、埋め込まれたコンテンツへのアクセスがユーザーに許可されます。

注: 一部の EAS では、アプリケーションが Tableau コンテンツにアクセスするためにユーザーの承認を求める同意ダイアログを表示するオプションがサポートされています。ユーザーに最高のエクスペリエンスを提供するには、ユーザーに代わって外部アプリケーションの要求に自動的に同意するように EAS を構成することをお勧めします。

サイトレベルの EAS について

Tableau Server 2024.2 以降では、サイトレベルの EAS を構成できます。サイトレベルで EAS を登録するには、Tableau サービスマネージャー (TSM) で接続済みアプリ有効にする必要があります。

サーバー全体の EAS

サーバー全体の EAS を登録するには、TSM Web UI を使用する方法と TSM CLI を使用する方法の 2 つがあります。

EAS を登録すると、確立された信頼関係が Tableau Server 上のすべてのサイトに適用されます。

オプション 1: TSM Web UI を使用する

1. Tableau Server 管理者として、Tableau サービスマネージャー (TSM) の Web UI にサインインします。詳細については、Tableau サービスマネージャーの Web UI へのサインインを参照してください。

2. 次のいずれか *1* つを実行します。

   - Tableau Server 2024.2 以降では、[ユーザー ID とアクセス] ページ > **[接続済みアプリ]** タブに移動します。
   - Tableau Server 2023.3 以前では、[ユーザー ID とアクセス] ページ > **[認可サーバー]** タブに移動します。

3. 次のいずれか *1* つを実行します。

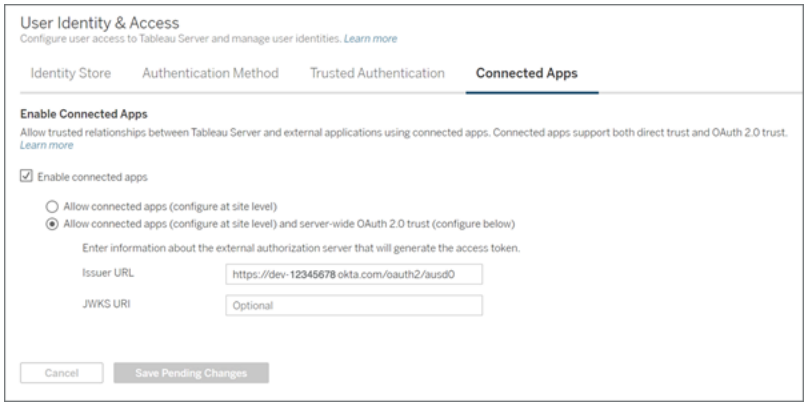
   - Tableau Server 2024.2 以降の場合:

     a. **[Enable connected apps (接続済みアプリを有効化)]** チェックボックスを選択します。

     b. 2 番目のラジオボタン **[接続済みアプリ(サイトレベルで設定) とサーバー全体の OAuth 2.0 信頼 (以下で設定) を許可]** を選択します。

     c. **[Issuer URL (発行元 URL)]** テキストボックスに、EAS の発行元 URL を貼り付けます。

     d. **[保留中の変更を保存]** ボタンをクリックします。

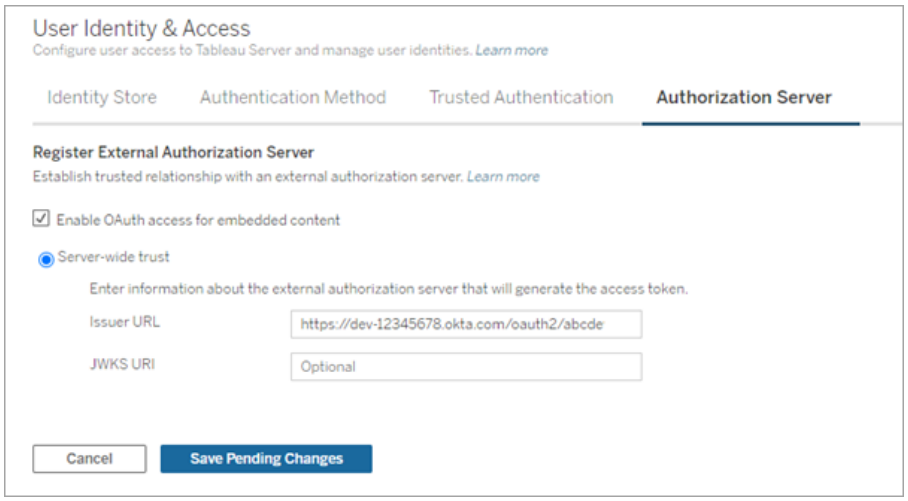
     

   - Tableau Server 2023.3 以前の場合:

a. **[Enable OAuth access for embedded content (埋め込みコンテンツの Oauth アクセスを有効にする)]** チェックボックスをオンにします。

b. **[Issuer URL (発行元 URL)]** テキストボックスに、EAS の発行元 URL を貼り付けます。

c. **[保留中の変更を保存]** ボタンをクリックします。

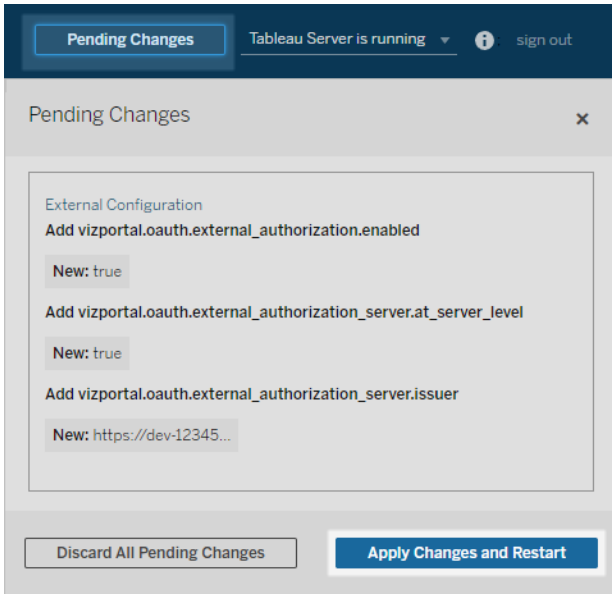


4. 終了したら、次の手順を実行します。

a. ページの右上隅にある **[変更を保留中]** ボタンをクリックします。

b. ページの右下隅にある **[変更を適用して再起動]** ボタンをクリックして、Tableau

Server を停止してから再起動します。



オプション 2: TSM CLI を使用する

1. クラスタの最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

```
tsm configuration set -k vizportal.oauth.external_
authorization.enabled -v true
tsm configuration set -k vizportal.oauth.external_
authorization_server.issuer -v "<issuer_url_of_EAS>"
tsm restart
```

サイトレベルの EAS

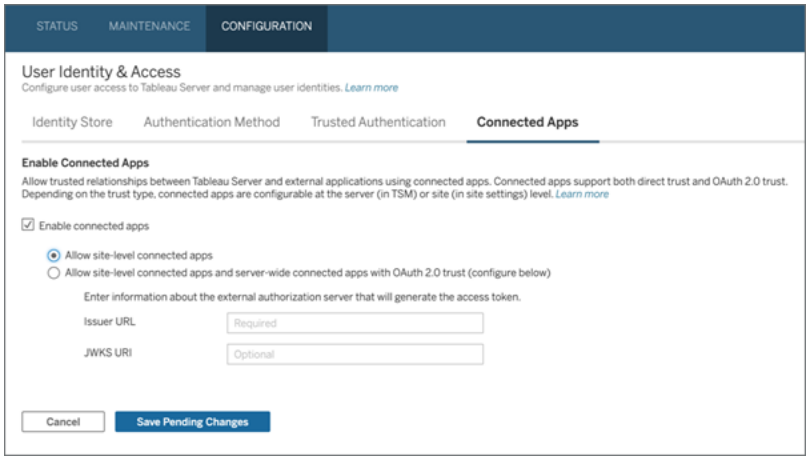
Tableau Server 2024.2 以降では、サイトに 1 つまたは複数の EAS を登録できます。サイトレベルで EAS を登録すると、確立された信頼関係はサイトのみに適用されます。

注: サイトレベルの EAS を構成するための前提条件は、TSM で接続済みアプリが有効になっていることです。

ステップ 1: 接続済みアプリを有効にする

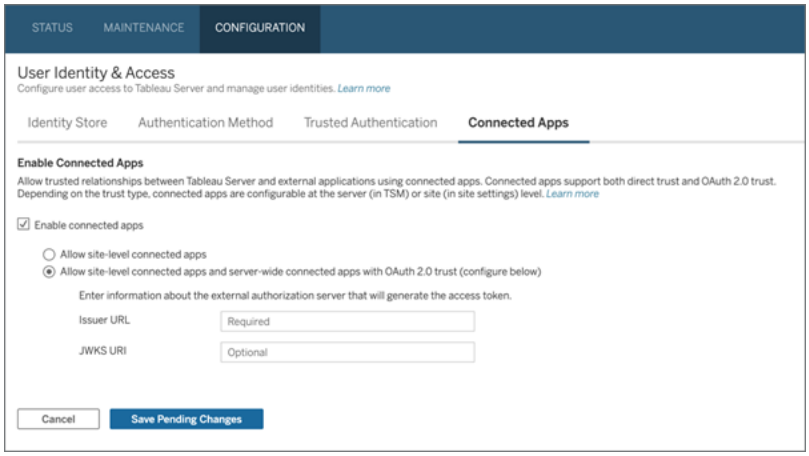
1. Tableau Server 管理者として、Tableau サービス マネージャー (TSM) の Web UI にサインインします。詳細については、Tableau サービス マネージャーの Web UI へのサインインを参照してください。
2. 「ユーザー ID とアクセス」ページ > **[接続済みアプリ]** タブに移動します。
3. **[Enable connected apps (接続済みアプリを有効化)]** チェックボックスを選択します。
4. 次のいずれか *1* つを実行します。

   - 最初のラジオ ボタン **[Allow connected apps (configure at site level) (接続済みアプリ (サイトレベルで構成) を許可)]** を選択して、サイトレベルのみで EAS を登録できるようにします。



   - (既定) 2 番目のラジオ ボタン **[接続済みアプリ (サイトレベルで設定) とサーバー全体の OAuth 2.0 信頼 (以下で設定) を許可]** を選択して、サイトレベルとサーバー全体の両方で EAS を登録できるようにします。このオプションを選択する場合は、サイトレベルで指定された発行者 URL がサーバー全体の発行者 URL と異なること

を確認してください。



5. **[保留中の変更を保存]** ボタンをクリックします。
6. 終了したら、次の手順を実行します。
   a. ページの右上隅にある **[変更を保留中]** ボタンをクリックします。
   b. ページの右下隅にある **[変更を適用して再起動]** ボタンをクリックして、Tableau Server を停止してから再起動します。

ステップ 2: EAS を登録する

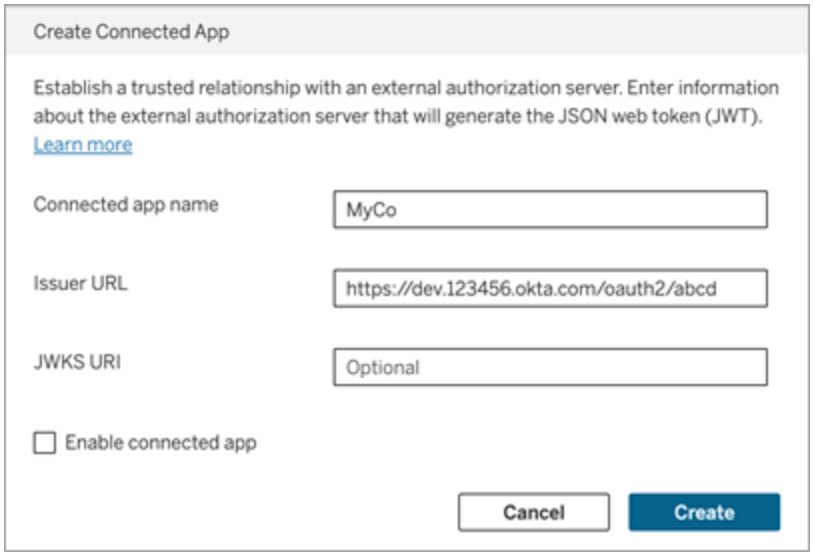
1. Tableau Server 管理者として、Tableau Server にサインインします。
2. 左側のペインから、**[設定] > [接続済みアプリ]** の順に選択します。
3. [New Connected App (新しい接続済みアプリ)] ボタンのドロップダウン矢印をクリックして、**[OAuth 2.0 Trust (OAuth 2.0 信頼)]** を選択します。
4. [Create Connected App (接続済みアプリの作成)] ダイアログ ボックスで、次の手順を行います。

a. **[名前]** テキストボックスに、接続済みアプリ名を入力します。

b. **[Issuer URL (発行元 URL)]** テキストボックスに、EAS の発行元 URL を貼り付けます。

c. **[接続済みアプリを有効にする]** を選択します。セキュリティ上の理由から、接続済みアプリは作成時に既定で無効に設定されています。

d. 完了したら、**[作成]** ボタンをクリックします。



5. 接続済みアプリを作成したら、接続済みアプリのサイト ID をコピーします。サイト ID は、上記のステップ 1 で説明した JWT の "aud" (オーディエンス) クレームに使用されます。

ステップ 3: 次のステップ

埋め込みのワークフローの場合

EAS を使用するように Tableau Server を構成した後、外部アプリケーションに埋め込みコードを追加する必要があります。手順 1 で説明したように、EAS によって生成された有効な JWT が、外部アプリケーションが呼び出す Web コンポーネントに含まれていることを確認してください。

Tableau コンテンツの埋め込みの詳細については、次のいずれかまたは両方を参照してください。

- メトリクスの埋め込みについては、Tableau ヘルプの「Web ページへのメトリクスの埋め込み」トピックを参照してください。(Tableau Server 2023.3 で、Tableau はメトリクスを埋め込む機能を廃止しました。)
- Tableau API v3 の埋め込み を使用して、Tableau のビューとメトリクスを埋め込みます。

注: 埋め込みコンテンツにアクセスするユーザーを正常に認証するには、ブラウザーがサードパーティーの Cookie を許可するように設定されている必要があります。

埋め込み用のドメイン許可リストを使用してコンテンツを埋め込むことができる場所を制御する

Tableau Server 2023.3 より、Tableau REST API の「サイトの埋め込み設定の更新」メソッドを使用して、Tableau コンテンツを制限なしで埋め込むか、または特定のドメインに制限して埋め込むかを制御できるようになります。

既定では、埋め込み用のサイト設定 `unrestrictedEmbedding` は `true` に設定されており、制限なしで埋め込むことができます。または、この設定を `false` に設定し、`allowList` パラメーターを使用して、外部アプリケーションの Tableau コンテンツを埋め込むドメインを指定することができます。

詳細については、次のいずれかまたは両方を参照してください。

- Tableau REST API ヘルプの「サイトの埋め込み設定の更新」
- Tableau 埋め込み API v3 ヘルプの「埋め込みに関する Tableau サイトの設定」

REST API 認証ワークフローの場合

JWT を構成したら、認証アクセスのために、有効な JWT を REST API サインイン要求に追加する必要があります。詳細については、「接続済みアプリのアクセス スコープ」を参照してください。

メタデータ API ワークフローの場合

JWT を構成したら、有効な JWT を REST API サインイン要求に追加する必要があります。詳細については、「接続済みアプリのアクセス スコープ」を参照してください。

接続済みアプリの管理

動的グループ メンバーシップ (埋め込みワークフローのみ)

Tableau Server 2024.2 以降、接続済みアプリが設定されて機能が有効になっている場合、外部アプリケーションから送信した JWT に含まれるカスタム クレームを通じてグループ メンバーシップを動的に制御できます。

設定すると、ユーザー認証の際、外部アプリケーションは、ユーザーをアサートするグループ (`https://tableau.com/groups`) とグループ名 (「Group1」と「Group2」など) という、2 つのカスタム グループ メンバーシップ クレームを含む JWT を送信します。Tableau は JWT を検証し、グループと、そのグループにパーミッションが依存するコンテンツへのアクセスを有効にします。

詳細については、アサーションを使用した動的グループ メンバーシップを参照してください。

既知の問題 (埋め込みのワークフローのみ)

接続済みアプリの使用にはいくつかの既知の問題がありますが、これらは今後のリリースで対処される予定です。

- **ツールバー機能:** 埋め込みコンテンツにツールバーのパラメーターが定義されている場合、すべてのツールバー機能が動作するとは限りません。この問題を回避するには、次の例のようにツールバーのパラメーターを非表示にすることをお勧めします。

  ```
  <tableau-viz id='tab-viz' src='https://<your_server>/t/<your_
  site>/...'
         toolbar='hidden'>
  </tableau-viz>
  ```

- **パブリッシュされたデータ ソース:** データベースの認証資格情報の入力を求める **[ユーザーにメッセージを表示]** が設定されているパブリッシュされたデータ ソースは表示されません。この問題を回避するには、可能であれば、データ ソースの所有者がデータベースの認証資格情報を埋め込むことをお勧めします。
- **複数のサイトへの埋め込みビュー:** Tableau Server 2023.1 以前では、同じブラウザーで異なるサイトのビューに切り替えると、エラー「**1008: 接続アプリのシークレットを取得できませんでした**」が発生します。この問題を回避するには、Tableau Server 2023.3 以降にアップグレードしてください。

トラブルシューティング

埋め込まれたコンテンツが外部アプリケーションに表示されない場合、または Tableau REST API 認証に失敗した場合は、ブラウザーのデベロッパー ツールを使用して、Tableau Server で有効になっている EAS 機能に関連する可能性のあるエラー コードを検査および特定できます。

次の表を参照して、エラー コードの説明と可能な解決策を確認してください。

エラー コード	概要	説明	可能な解決策または説明
5	SYSTEM_USER_NOT_FOUND	Tableau ユーザーが見つかりませんでした	この問題を解決するには、JWT の 'sub' (サブジェクト) クレーム値が、認証済みの Tableau Server の "username" であることを確認します。この値は大文字と小文字が区別されます。
16	LOGIN_FAILED	ログインできませんでした	このエラーは通常、JWT の次のクレームの問題のいずれかが原因で発生します。 • 'exp' (有効期限) が既定の最大有効期間を超えています。この問題を解決するには、有効な JWT に必要な登録済みのクレームを確認し、正しい値を使用するようにします。最大有効期間を変更するには、vizportal.oauth.external_authorization_server.max_expiration_period_in_minutes コマンドを使用します。 • 'sub' (サブジェクト) が、不明

			なユーザーを呼び出しています。この問題を解決するには、'sub' クレームが "username" であることを確認します。この値は大文字と小文字が区別されます。
67	FEATURE_NOT_ENABLED	オンデマンドアクセスがサポートされていません	オンデマンドアクセスは、ライセンス認証されている Tableau Cloud サイトを介してのみ利用できます。
10081	COULD_NOT_RETRIEVE_IDP_METADATA	EAS メタデータエンドポイントがありません	この問題を解決するには、EAS が正しく構成されており、正しい発行元が呼び出されていることを確認します。
10082	AUTHORIZATION_SERVER_ISSUER_NOT_SPECIFIED	発行元がありません	この問題を解決するには、正しい発行元が呼び出されていることを確認します。発行元の URL を変更するには、vizportal.oauth.external_authorization_server.issuer コマンドを使用します。
10083	BAD_JWT	JWT ヘッダーに問題が含まれています	'kid' (シークレット ID) または 'clientId' (発行元) のクレームが JWT ヘッダーにありません。この問題を解決するには、この情報が含まれていることを確認します。
10084	JWT_PARSE_ERROR	JWT に問題が含まれています	この問題を解決するには、次を確認します。 • この JWT で参照される 'aud' (オーディエンス) 値が "tableau" 値を使用しているこ

			とを確認します。この値は大文字と小文字が区別されます。 • 'aud' (オーディエンス) と 'sub' (サブジェクト) は JWT に含まれています。
10085	COULD_NOT_FETCH_JWT_KEYS	JWT がキーを見つけることができませんでした	シークレットを見つけることができませんでした。 この問題を解決するには、正しい発行元が呼び出されていることを確認します。発行元の URL を変更するには、vizportal.oauth.external_authorization_server.issuer コマンドを使用します。
10087	BLOCKLISTED_JWS_ALGORITHM_USED_TO_SIGN	JWT 署名アルゴリズムの問題	この問題を解決するには、署名アルゴリズムを削除します。詳細については、vizportal.oauth.external_authorization_server.blocklisted_jws_algorithmsを参照してください。
10088	RSA_KEY_SIZE_INVALID	JWT 署名要件の問題	この問題を解決するには、EAS または IdP を使用して、JWT が 2048 ビットの RSA キーサイズで署名されていることを確認します。
10091	JTI_ALREADY_USED	一意の JWT が必要です	JWT はすでに認証プロセスで使用されています。この問題を解決するには、EAS または IdP が新しい JWT を生成する必要があります。
10092	NOT_IN_DOMAIN_ALLOW_LIST	埋め込みコンテンツのドメインが指定さ	この問題を解決するには、設定 `unrestrictedEmbedding` が `true` に設定されていること、または `domainAllowlist` パラメーター

		れていません	に、Tableau REST API の「サイトの埋め込み設定の更新」メソッドを使用して Tableau コンテンツを埋め込むドメインが含まれていることを確認してください。
10094	MISSING_REQUIRED_JTI	JWT ID がありません	この問題を解決するには、'jti' (JWT ID) が JWT に含まれていることを確認します。
10096	JWT_EXPIRATION_EXCEEDS_CONFIGURED_EXPIRATION_PERIOD		'exp' (有効期限) が既定の最大有効期間を超えています。この問題を解決するには、有効な JWT に必要な登録済みのクレームを確認し、正しい値を使用するようにします。最大有効期間を変更するには、vizportal.oauth.external_authorization_server.max_expiration_period_in_minutes コマンドを使用します。
10097	SCOPES_MALFORMED	スコープクレームに関する問題	このエラーは、'scp' (スコープ) クレームが JWT にないか、リスト型として渡されていない場合に発生する可能性があります。この問題を解決するには、'scp' が JWT に含まれ、リスト型として渡されていることを確認します。JWT に関するトラブルシューティングのヘルプについては、auth0 サイトの「デバッガー」を参照してください。
10098	JWT_UNSIGNED_OR_ENCRYPTED	JWT が署名されていないか、暗号化されています	Tableau は、署名されていない JWT または暗号化された JWT をサポートしていません。

10099	SCOPES_MISSING_IN_JWT	スコープのクレームがありません	JWT に必要な 'scp' (スコープ) クレームがありません。この問題を解決するには、'scp' が JWT に含まれていることを確認します。JWT に関するトラブルシューティングのヘルプについては、auth0 サイトの「デバッガー」を参照してください。
10100	JTI_PERSISTENCE_FAILED	予期しない JWT ID エラー	'jti' (JWT ID) で予期しないエラーが発生しました。この問題を解決するには、新しい 'jti' で新しい JWT を生成する必要があります。
10103	JWT_MAX_SIZE_EXCEEDED	JWT が最大サイズを超えています	JWT サイズが 8000 バイトを超えている場合、このエラーが発生する可能性があります。この問題を解決するには、必要なクレームのみが Tableau Server に渡されていることを確認してください。

接続済みアプリのアクセス スコープ

Tableau Server バージョン 2022.3 以降、Tableau 接続済みアプリを使用すると、Tableau Server ユーザーの代わりに、カスタム アプリケーションを通じて Tableau REST API をプログラムで呼び出して利用できるようになりました。最初にサインインを要求するときに定義される JSON Web トークン (JWT) を使用することにより、REST API へのアクセスが可能になります。JWT 内のスコープには、カスタム アプリケーションとそのユーザーが連携アプリを通じて利用できる REST API メソッドが定義されている必要があります。

REST API へのアクセスを連携アプリを使用して承認することにより、次を実現します。

- セキュリティの強化 — コンテナ内で .env ファイルを介して管理者ユーザーのパスワードを保存および管理するよりも、JWT をベアラー トークンとして使用する方が本質的に安全です
- 効率の向上 — JWT をベアラー トークンとして使用すると、2 つのリクエストではなく、サインイン エンドポイントへの 1 つのリクエストで偽装を簡単化することができます
- 動的なコンテンツ検索や高度なフィルタリングなど、複雑な Tableau 統合とバックエンド クエリを拡張および自動化します

スコープ アクション

接続済みアプリが使用するスコープは、JWT 認可をサポートする REST API メソッド(下記)を通じて、コンテンツや管理アクションへのアクセス権を付与します。スコープはコロンで区切られた文字列であり、名前空間 `tableau` で始まり、アクセスが許可されている Tableau リソース (`datasources` など) が続き、そのリソースに対して許可されているアクション (`update` など) で終わります。

スコープが実行できるアクションは次のとおりです。

- `create`
- `read`
- `run`
- `update`
- `download`
- `delete`

たとえば、カスタム アプリケーションが UpdateDataSource メソッドを呼び出せるようにするスコープは次のようになります。

```
tableau:datasources:update
```

スコープの種類

使用するスコープの種類は、有効にするコンテンツまたは管理アクションによって異なります。スコープは通常、コンテンツ読み取り、個別、ワイルドカード、クロスカテゴリのいずれかのタイプに分類されます。

- **コンテンツ読み取りスコープ:** コンテンツ読み取りスコープ `tableau:content:read` は、Tableau コンテンツに対してサポートされている GET メソッドを有効にします。このスコープを使用すると、REST API カテゴリ全体でアクションを有効にできます。具体的には、このスコープを使用して、データ ソース、メトリクス、ビュー、ワークブック、プロジェクト、およびサイトに対する GET メソッドを有効にします。Tableau Server 2023.3 以降は、メタデータ API で使用する認証資格情報トークンの作成に使用される JWT でこのスコープも指定します。Tableau Server 2025.1 以降は、VizQL データ サービスで使用する認証資格情報トークンの作成に使用される JWT でもこのスコープを指定します。

注: ユーザーやグループなどの管理アクションに対するGET メソッドを有効にするには、それらの個別スコープを使用できます。

- 個別スコープ: サポートされているコンテンツと管理アクションを有効にするには、それらの個別スコープを使用します。個別スコープは、通常、単一のメソッドとREST API カテゴリに関連付けられています。

  例:

  - データソースアクションのパブリッシュまたは更新を有効にする場合は、それぞれ個別スコープ `tableau:datasources:create` または `tableau:datasources:update` を使用できます。
  - ユーザーの追加または削除などの管理アクションについては、それぞれ個別スコープ `tableau:users:create` または `tableau:users:delete` を使用できます。

  注: REST API のカテゴリをまたがるアクションを有効にできる個別スコープもいくつかあります。例えば、`tableau:views:download` は、データの表示やワークブックの REST API カテゴリでアクションを有効にします。

- ワイルドカード(*) スコープ: 特定のスコープでは、アクションをワイルドカード文字 (*) に置き換えて、特定の REST API カテゴリ内でサポートされているアクションを有効にすることができます。

  例:

  - ワイルドカードスコープ `tableau:projects:*` を使用すると、プロジェクトの REST API カテゴリで create、delete、および update アクションを有効にできます。
  - ワイルドカードスコープ `tableau:users:*` を使用すると、ユーザーの REST API カテゴリで get/list、add、delete、および update アクションを有効にできます。
  - ワイルドカードスコープ `tableau:tasks:*` を使用すると、抽出とサブスクプションの REST API カテゴリで get/list、add、delete、update、および run アクションを有効にできます。また、このスコープは、データソースの更新 (抽出の場合) とワークブックの更新を有効にします。

- クロスカテゴリスコープ: コンテンツ読み取りスコープに加えて、使用された場合、さまざまな REST API カテゴリにまたがってサポートされているアクションを有効にする追加スコープがいくつかあります。

例:

- `tableau:tasks:run` スコープを使用すると、データソースとワークブックの REST API カテゴリでアクションを有効にします。
- 繰り返しになりますが、`tableau:views:download` スコープを使用すると、データの表示とワークブックの REST API カテゴリでアクションを有効にします。
- `tableau:permissions:update` または `tableau:permissions:delete` などのパブリッシュスコープを使用すると、データソース、ワークブック、およびプロジェクトの REST API カテゴリでアクションを有効にします。

REST API アクセスを認可する方法の概要

次のリストは、JWT を介して REST API へのアクセスをリクエストする手順をまとめたものです。

1. 次のいずれかの方法を使用して、**連携アプリを作成**します。
   - 直接信頼を使用して接続済みアプリを設定する
   - OAuth 2.0 信頼を使用して接続済みアプリを設定する
2. **有効な JWT を生成します。**カスタム アプリケーションは、指定したスコープで設定された有効な JWT を実行時に生成します
3. サインイン **リクエストを行います。**カスタム アプリケーションは、JWT を使用してサインインをリクエストし、Tableau 認証資格情報トークンとサイト ID (LUID) を返します。
4. **後続のリクエストで Tableau アクセス トークンを使用します。**後続の REST API 呼び出しでは、1) Tableau 認証資格情報トークンを `X-Tableau-Auth` ヘッダー値として使用し、2) サイト ID (LUID) をリクエスト URI で使用します。

例

直接信頼を使用して連携アプリを作成する場合の例です。直接信頼を使用して REST API を呼び出すカスタム アプリケーションは、連携アプリが生成したクライアント ID とクライアント シークレットを使用して有効な JWT を生成します。

JWT のスコープ

REST API へのアクセスを正常に認可するには、JWT に REST API 機能を定義するスコープが含まれている必要もあります。たとえば、データソース関連のさまざまなメソッドを有効にするには、JWT に次のようなスコープを含めることができます。

```
"tableau:content:read","tableau:datasources:create","tableau:datasou-
rces:update","tableau:datasources:download","tableau:tasks:run"
```

または

```
"tableau:content:read","tableau:datasources:*","tableau:tasks:run"
```

注: スコープの値はリスト型として渡す必要があります。

サインイン リクエストの URI

REST API を呼び出すには、カスタム アプリケーションは最初にサインイン要求を行い、Tableau 認証資格情報のトークンを生成する必要があります。

```
POST https://myco/api/3.17/auth/signin
```

リクエスト本文

JWT を使用して REST API へのアクセスを認可するには、サインイン リクエストの本文に次の例のような有効な JWT が含まれている必要があります。

```
<tsRequest>
   <credentials
jwt="eyJpc3MiOiI4ZTFiNzE3Mi0zOWMzLTRhMzItODg3ZS1mYzJiNDExOWY1NmQiLCJ-
hbGciOiJIUzI1NiIsImtpZCI6ImIwMTE1YmY5LTNhNGItNGM5MS1iMDA5LWNmMGMxNzB-
iMWE1NiJ9.eyJhdWQiOiJ0YWJsZWF1Iiwic3ViIjoicm1vaGFuQHRhYmxlYXUuY29tIi-
wic2NwIjpbInRhYmxlYXU6c2l0ZXM6cmVhZCJdLCJpc3MiOiI4ZTFiNzE3Mi0zOWMzLT-
RhMzItODg3ZS1mYzJiNDExOWY1NmQiLCJleHAiOjE2NDg2Njg0MzksImp0aSI6IjY1ZW-
FmMmYxLTNmZTgtNDc5Ny1hZmRiLTMyODMzZDVmZGJkYSJ9.mUv2o4gtBTrMVLEXY5XTp-
zDQTGvfE2LGi-3O2vdGfT8">
    <site contentUrl="mycodotcom"/>
   </credentials>
</tsRequest>
```

レスポンス本文

サインイン リクエストに対して、Tableau 認証資格情報トークンを含む、次のようなレスポンス本文が生成されます。

```
<tsResponse>
   <credentials token="12ab34cd56ef78ab90cd12ef34ab56cd">
    <site id="9a8b7c6d5-e4f3-a2b1-c0d9-e8f7a6b5c4d" contentUrl=""/>
    <user id="9f9e9d9c-8b8a-8f8e-7d7c-7b7a6f6d6e6d" />
   </credentials>
</tsResponse>
```

Tableau アクセス トークンが生成されたら、後続のすべての REST API リクエストのヘッダーに追加します。

ヘッダー

```
X-Tableau-Auth:12ab34cd56ef78ab90cd12ef34ab56cd
```

Tableau アクセス トークンを使用する後続のすべての REST API リクエストは、JWT のスコープによって制限されます。

JWT 認可をサポートする REST API メソッド

次のスコープを連携アプリに関連付けて、カスタム アプリケーションがユーザーに代わって REST API に対して持つことができるアクセス権とメソッドを定義できます。

注:

- 以下の表にリストアップされていない REST API 機能については、他の認可メカニズムを使用してメソッドにアクセスできます。詳細については、Tableau REST API ヘルプの「認証メソッド」を参照してください。
- サインインとサインアウトのメソッドは、共に JWT 認証でサポートされていますが、Tableau Server 2023.3.以降では、スコープを使用する必要はありません。
- 埋め込み API v3 でサポートされているスコープについては、次のいずれかを参照してください。
    - 直接信頼を使用して接続済みアプリを設定する
    - OAuth 2.0 信頼を使用して接続済みアプリを設定する

ワイルドカード (*) スコープ

ワイルドカード スコープは、特定のアクションではなく、ワイルドカード文字 (*) を使用して、特定の REST API カテゴリ内でサポートされている複数のアクションを有効にします。これらのスコープには

以下が含まれます。

スコープ	有効にされるメソッド
`tableau:datasources:*`	データソースの create、update、および update connection メソッドを有効にします。
`tableau:metrics:*`	メトリクスの query、update、および delete アクションを有効にします。
`tableau:workbooks:*`	ワークブックの publish、update、download、および preview image アクションを有効にします。
`tableau:groups:*`	グループの create、query、update、および delete アクションを有効にします。
`tableau:projects:*`	プロジェクトの create、delete、update メソッドを有効にします。
`tableau:users:*`	ユーザーの get/list、add、delete、および update メソッドを有効にします。
`tableau:tasks:*` **注:** このスコープもクロスカテゴリです。	抽出タスクとサブスクリプション タスクの get/list、add、delete、update、および run メソッドを有効にします。 ワークブックのデータソースの update メソッドを有効にします。

クロスカテゴリスコープ

クロスカテゴリスコープは、複数の REST API カテゴリにまたがってサポートされている複数のアクションを有効にします。これらのスコープには以下が含まれます。

スコープ	有効にされるメソッド
`tableau:content:read`	データソース、メトリクス、ビュー、ワークブック、プロジェクト、サイトなど、Tableau コンテンツの read/list メソッドを有効にします。

スコープ	有効にされるメソッド
`tableau:tasks:run`	データソース、ワークブック、抽出の run メソッドを有効にします。
`tableau:views:download`	データの表示とワークブックの download メソッドを有効にします。
`tableau:tasks:*` **注:** このスコープもワイルドカードです。	抽出タスクとサブスクリプション タスクの get/list、add、delete、update、および run メソッドを有効にします。 ワークブックのデータ ソースの update メソッドを有効にします。

個別スコープ

メソッド	スコープ	説明
(スコープのないメソッド)	(なし)	JWT でスコープが定義されていない場合、REST API へのアクセスは拒否されます。
サインイン	(スコープは不要)	のユーザーとして、Tableau Server の指定されたサイトにサインインします。
サインアウト	(スコープは不要)	現在のセッションからサ

メソッド	スコープ	説明
		インアウトします。
(コンテンツ読み取りスコープ)	`tableau:content:read`	Tableau コンテンツ (データソース、メトリクス、ビュー、ワークブック、プロジェクト) の read/list アクションを有効にします。
ラベル		
Delete Label	`tableau:labels:delete`	LUID で 1 つのデータラベルを削除します。
Delete Labels	`tableau:labels:delete`	1 つまたは複数のアセットのデータラベルを削除します。
Get Label	`tableau:labels:read`	LUID で 1 つのデータラベルを取得します。
Get Labels	`tableau:labels:read`	1 つまたは複数のアセットのデータラベルに関する

メソッド	スコープ	説明
		情報を表示します。
Update Label	`tableau:labels:update`	LUID で 1 つのラベルを更新します。
Update Labels	`tableau:labels:update`	1 つまたは複数のアセットのラベルを作成または更新します。
データ ソース		
(すべての `tableau:datasource-s:` メソッド)	`tableau:datasources:*`	データ ソースの create メソッド、データ ソースの update メソッド、およびデータ ソース接続の update メソッドを有効にします。
データ ソースのパブリッシュ	`tableau:datasources:create`	データ ソースをサイトにパブリッシュします。または、既存のパブリッシュ済みデータ ソースにデータを追加します。

メソッド	スコープ	説明
データソースの照会	`tableau:content:read`	パブリッシュ済みのデータソースに関する情報を取得します。
データソースの照会	`tableau:content:read`	サイトでパブリッシュ済みのすべてのデータソースに関する情報を取得します。
データソース接続の照会	`tableau:content:read`	パブリッシュ済みのデータソースのサーバー アドレス、ポート、ユーザー名、またはパスワード情報を取得します。
データソースの更新	`tableau:datasources:update`	データソースの所有者、プロジェクト、証明書ステータスを更新します。
データソース接続の更新	`tableau:datasources:update`	データソース接続のサーバー アドレ

メソッド	スコープ	説明
		ス、ポート、ユーザー名、またはパスワードを更新します。
データソースを今すぐ更新	`tableau:tasks:run`	抽出の更新を実行します。
抽出		
(すべての `tableau:tasks:` メソッド)	`tableau:tasks:*`	抽出、サブスクリプション、データソースの更新 (抽出を使用するデータソース)、ワークブック メソッドの更新の create、delete、get、list、run、および update refresh アクションを有効にします。
サイトでの抽出更新タスクのリスト	`tableau:tasks:read`	サイトで構成されている抽出更新タスクをリストします。

メソッド	スコープ	説明
抽出更新タスクの実行	`tableau:tasks:run`	抽出更新タスクを実行します。
フロー		
フローのパブリッシュ	`tableau:flows:create`	フローをパブリッシュします。
ジョブ		
クエリジョブ	`tableau:jobs:read`	ジョブを使用して追跡される非同期プロセスに関するステータス情報を取得します。
クエリジョブ	`tableau:jobs:read`	指定されたサイトのアクティブなジョブのリストを取得します。
メトリクス (廃止) **従来のメトリクス機能の廃止** Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、		

メソッド	スコープ	説明
「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。		
(すべての tableau:metrics: メソッド)	tableau:metrics:*	メトリクスの query、update、および delete アクションを有効にします。
(すべての tableau:metrics: メソッド)	tableau:metrics:*	メトリクスの query、update、および delete アクションを有効にします。
メトリクスの取得	tableau:content:read	メトリクスを取得します。
メトリクスの削除	tableau:metrics:delete	メトリクスを削除します。
メトリクスを一覧表示	tableau:content:read	サイトのメトリクスの一覧を取得します。
メトリクス データの照会	tableau:metrics:download	メトリクスの参照元データをカンマ区切り値 (.csv) 形式で取得します。

メソッド	スコープ	説明
メトリクスの更新	`tableau:metrics:update`	所有者、プロジェクト、一時停止ステータス、メトリクス名を更新します。
サブスクリプション		
(すべての `tableau:tasks:` メソッド)	`tableau:tasks:*`	抽出、サブスクリプション、データソースの更新 (抽出を使用するデータソース)、ワークブック メソッドの更新の create、delete、get、list、run、および update refresh アクションを有効にします。
サブスクリプションの作成	`tableau:tasks:create`	サブスクリプションを作成します。
サブスクリプションの削除	`tableau:tasks:delete`	サブスクリプションを削除します。

メソッド	スコープ	説明
サブスクリプションの取得	`tableau:tasks:read`	サブスクリプションの詳細を取得します。
サブスクリプションのリスト表示	`tableau:tasks:read`	サイトでのサブスクリプションをリストします。
サブスクリプションの更新	`tableau:tasks:update`	サブスクリプションを更新します。
ビュー		
カスタム ビューの削除	`tableau:views:update`	指定されたカスタム ビューを削除します。
カスタム ビューの取得	`tableau:content:read`	指定されたカスタム ビューの詳細を取得します。
カスタム ビュー イメージの取得	`tableau:views:download`	指定したカスタム ビューの .png 形式のイメージ ファイルをダウンロードします。
ビューの取得	`tableau:content:read`	ビューに関す

メソッド	スコープ	説明
		る詳細を取得します。
パスでビューを取得	`tableau:content:read`	指定された名前を使用して、サイトのすべてのビューの詳細を取得します。
カスタム ビューのリスト	`tableau:content:read`	サイト上のカスタム ビューのリストを取得します。
ビュー データの照会	`tableau:views:download`	レンダリングされたビューをカンマ区切り値 (.csv) 形式で取得します。
ビュー PDF の照会	`tableau:views:download`	ビューを PDF ファイル (.pdf) で取得します。
ビュー画像の照会	`tableau:views:download`	ビューを画像ファイル (.png) で取得します。
サイトのビューの照会	`tableau:content:read`	サイトのすべてのビューを取得します。

メソッド	スコープ	説明
ワークブックのビューの照会	`tableau:content:read`	指定されたブックのすべてのビューを取得します。
ビューのプレビュー画像の照会	`tableau:views:download`	ビューのサムネイル画像 (.png) を取得します。
カスタム ビューの更新	`tableau:views:update`	既存のカスタム ビューの所有者または名前を変更します。
ワークブック		
(すべての `tableau:workbooks:` メソッド)	`tableau:workbooks:*`	ワークブックの publish、update、download、および preview image アクションを有効にします。
ワークブックのパブリッシュ	`tableau:workbooks:create`	ワークブック (.twb または .twbx) をパブリッシュします。
ワークブックの取得	`tableau:content:read`	指定されたワークブックと

メソッド	スコープ	説明
		その詳細を取得します。
サイトのワークブックの照会	`tableau:content:read`	サイトにパブリッシュされているワークブックのリストを取得します。
ワークブックのプレビュー画像の照会	`tableau:workbooks:download`	ワークブックのサムネイル画像 (.png) を取得します。
ワークブックの更新	`tableau:workbooks:update`	既存のワークブックを変更します。
ワークブック接続の更新	`tableau:workbooks:update`	接続情報を更新します。
今すぐワークブックを更新	`tableau:tasks:run`	スケジュールされたタスクの外でワークブックの更新を開始します。
パブリッシュ		
ファイル アップロードへの追加	`tableau:file_uploads:create`	データのブロックをアップロードし、すでにアップ

メソッド	スコープ	説明
		ロードされているデータに追加します(「ファイルアップロードの開始」メソッドでアップロードが開始された後に使用します)。
ファイル アップロードの開始	`tableau:file_uploads:create`	ファイルのアップロードプロセスを開始します。
ダウンロード		
データ ソースのダウンロード	`tableau:datasources:download`	データ ソース (.tdsx) をダウンロードします。
ビューのクロス集計 Excel のダウンロード	`tableau:views:download`	ビューからクロス集計データを含む Excel ファイル (.xlsx) をダウンロードします。
ワークブックのダウンロード	`tableau:workbooks:download`	ワークブック (.twb または .twbx) をダウンロードしま

メソッド	スコープ	説明
		す。
ワークブック改訂版のダウンロード	`tableau:workbooks:download`	ワークブック (.twb または .twbx) の特定のバージョンをダウンロードします。
ワークブックPDF のダウンロード	`tableau:workbooks:download`	ワークブックのシートの画像を含む PDF ファイル (.pdf) をダウンロードします。
ワークブックPowerPoint のダウンロード	`tableau:views:download`	ワークブックのシートのスライドを含む PowerPoint ファイル (.pptx) をダウンロードします。
ユーザー		
(すべての `tableau:users` メソッド)	`tableau:users:*`	ユーザーの add、query、update、および remove アクションを有効にします。

メソッド	スコープ	説明
グループへのユーザー追加	`tableau:groups:update`	ユーザーをグループに追加します。
サイトへのユーザー追加	`tableau:users:create`	ユーザーを追加し、そのユーザーをサイトに割り当てます。
グループ内ユーザーの取得	`tableau:groups:read`	グループ内のユーザーの一覧を取得します。
サイト上ユーザーの取得	`tableau:users:read`	サイト上のすべてのユーザーを取得します。
サイト上ユーザーの照会	`tableau:users:read`	サイト上の 1 人のユーザーを取得します。
グループからのユーザー削除	`tableau:groups:update`	ユーザーをグループから削除します。
サイトからのユーザー削除	`tableau:users:delete`	ユーザーをサイトから削除します。
グループ		
(すべての `tableau:groups:` メソッ	`tableau:groups:*`	グループの create、

メソッド	スコープ	説明
ド)		query、update、および delete アクションを有効にします。
グループの作成	`tableau:groups:create`	グループを作成します。
グループの削除	`tableau:groups:delete`	グループを削除します。
ユーザーのグループの取得	`tableau:users:read`	ユーザーが属するグループのリストを取得します。
グループの照会	`tableau:groups:read`	サイト上のグループのリストを取得します。
グループの更新	`tableau:groups:update`	グループを更新します。
グループ セット		
(すべての `tableau:groupsets:` メソッド)	`tableau:groupsets:*`	グループ セットの作成、クエリ、更新、削除アクションを有効にします。
グループ セットへのグループの追加	`tableau:groupsets:update`	グループ セットにグループを追加しま

メソッド	スコープ	説明
		す。
グループセットの作成	`tableau:groupsets:create`	グループセットを作成します。
グループセットの削除	`tableau:groupsets:delete`	グループセットを削除します。
グループセットの取得	`tableau:groupsets:read`	グループセットに関する情報を返します。
グループセットの作成一覧表示	`tableau:groupsets:read`	すべてのグループセットを一覧表示します。
グループセットからのグループの削除	`tableau:groupsets:update`	グループセットからグループを削除します。
グループセットの更新	`tableau:groupsets:update`	グループセット名を更新します。
プロジェクト		
(すべての `tableau:projects:` メソッド)	`tableau:projects:*`	プロジェクト create、update、および delete アクションを有効にします。

メソッド	スコープ	説明
プロジェクトの作成	`tableau:projects:create`	プロジェクトを作成します。
プロジェクトの削除	`tableau:projects:delete`	プロジェクトを削除します。
プロジェクトの照会	`tableau:content:read`	プロジェクトのリストを取得します。
プロジェクトの更新	`tableau:projects:update`	プロジェクトの名前、説明、プロジェクト階層を更新します。
パーミッション		
(すべての `tableau:permissions:` メソッド)	`tableau:permissions:*`	パーミッションの add、query、update、および delete アクションを有効にします。
データソース権限の追加	`tableau:permissions:update`	Tableau Server のユーザーまたはグループにデータソースへのアクセス権限を追加します。

メソッド	スコープ	説明
デフォルト権限の追加	`tableau:permissions:update`	プロジェクト内のメトリクス、フロー、ワークブック、データソース、データの役割、レンズなどのリソースに対するデフォルトの権限機能をユーザーまたはグループに追加します。
プロジェクト権限の追加	`tableau:permissions:update`	ユーザーまたはグループにプロジェクトへの権限を追加します。
表示権限の追加	`tableau:permissions:update`	ユーザーまたはグループにビューへの権限を追加します。
ワークブック権限の追加	`tableau:permissions:update`	ユーザーまたはグループに指定されたブックワークへの権限を追加します。
データソース権限の削除	`tableau:permissions:delete`	プロジェクト

メソッド	スコープ	説明
		内のメトリクス、フロー、ワークブック、データソース、データの役割、レンズなどのリソースに対するデフォルトの権限機能をユーザーまたはグループから削除します。
デフォルト権限の削除	`tableau:permissions:delete`	プロジェクト内のメトリクス、フロー、ワークブック、データソース、データの役割、レンズなどのリソースに対するデフォルトの権限機能をユーザーまたはグループから削除します。
プロジェクト権限の削除	`tableau:permissions:delete`	ユーザーまたはグループのプロジェクト権限を削除

メソッド	スコープ	説明
		します。
表示権限の削除	tableau:permissions:delete	ユーザーまたはグループの表示権限を削除します。
ワークブック権限の削除	tableau:permissions:delete	ユーザーまたはグループのワークブック権限を削除します。
データソース権限の照会	tableau:permissions:read	データソースの権限のリストを取得します。
デフォルト権限の照会	tableau:permissions:read	メトリクス、ワークブック、データソースに対するユーザーとグループの既定のパーミッション機能を取得します。
プロジェクト権限の照会	tableau:permissions:read	プロジェクトに対する権限のリストを取得します。
表示権限の照会	tableau:permissions:read	ビューに対する権限のリストを取得します。

メソッド	スコープ	説明
ワークブック権限の照会	`tableau:permissions:read`	ワークブックに対する権限のリストを取得します。
サイト		
(すべての `tableau:sites:` メソッド)	`tableau:sites:*`	サイトの create、query、update、および delete アクションを有効にします。
サイトの作成	`tableau:sites:create`	Tableau Server にサイトを作成します。
最近閲覧したサイトの取得	`tableau:content:read`	サインインユーザーが最近作成、更新、アクセスしたビューとワークブックの詳細を取得します。
複数サイトの照会	`tableau:sites:read`	Tableau Server 上のすべてのサイトを一覧表示します。
サイトのビューの照会	`tableau:content:read`	サイト上のす

メソッド	スコープ	説明
		べてのビューを一覧表示します。
サイトの更新	tableau:sites:update	サイトを更新します。

トラブルシューティングのスコープ

401001 - サインイン エラー

エラー401001 が発生した場合、**サインイン**の応答本文には、追加の接続アプリ固有のエラーコード(16、10084、または 10085) のいずれかが追加されます。

たとえば、次の応答本文では、「10084」は接続アプリのエラー コードであり、REST API 認証に JWT を使用して Tableau Server にサインインする際に起こる問題のトラブルシューティングに役立ちます。

```
<error code="401001">
  "summary": "Signin Error",
  "detail": "Error signing in to Tableau Cloud (10084)"
</error>
```

問題を解決するには、該当するエラー コードとその潜在的な原因の説明を参照してください。

- **16: Cloud でユーザーが見つかりません -** このエラーは、正しくない「sub」(ユーザー名) が指定されると発生する可能性があります。
- **10084: Cloud でアクセス トークンを解析できませんでした -** このエラーは、次の理由で発生する可能性があります。
    - JWT が無効であるか、予期しない問題が発生しました
    - 正しくない [aud] (オーディエンス) が指定されました
    - 直接信頼の場合、シークレットの署名に問題がありました

- **10085: Cloud でクライアント ID の署名を検証するためのシークレットを取得できませんでした -** このエラーは、次の理由で発生する可能性があります。
  - 指定された [iss] のクライアント ID が正しくありません
  - 直接信頼の場合、正しくない [kid] (シークレット ID) が指定されました
  - EAS で、JWKSource からキーを取得できませんでした

401002 - 不正アクセス エラー

エラー 401002 が発生し、リクエストを行う適切なパーミッションがあることを確認した場合は、JWT に含まれるスコープが正しく、作成しようとしているリクエストと一致していることを確認してください。エンドポイントとサポートされるスコープのリストについては、上記の「JWT 認可をサポートする REST API メソッド」セクションを参照してください。

接続済みアプリをトラブルシューティングする - 直接信頼

埋め込みコンテンツがカスタム アプリケーションに表示されない場合、または Tableau REST API 認証が失敗した場合は、ブラウザーの開発者ツールを使用して、埋め込みコンテンツの表示に使用される Tableau 接続済みアプリと関連している可能性のあるエラー コードを調べて、特定することができます。

注: セッション トークンを有効にするためには、外部アプリケーションと外部アプリケーションをホストするサーバーの時計が協定世界時 (UTC) に設定されている必要があります。いずれかの時計が異なる規格を使用している場合、接続済みアプリは信頼されません。

次の表を参照して、エラー コードの説明と可能な解決策を確認してください。

エラー コード	概要	説明	可能な解決策または説明
5	SYSTEM_USER_NOT_FOUND	Tableau ユーザーが見つかりませんでした	この問題を解決するには、JWT の 'sub' (サブジェクト) クレーム値が Tableau Server の "username" であることを確認します。この値は大文字と小文字が区別されます。
16	LOGIN_FAILED	ログインできませんで	このエラーは通常、JWT の次のクレームの問題のいずれかが原因で発生しま

		した	す。 • 'exp' (有効期限) が既定の最大有効期間を超えています。この問題を解決するには、有効な JWT に必要な登録済みのクレームを確認し、正しい値を使用するようにします。最大有効期間を変更するには、vizportal.oauth.connected_apps.max_expiration_period_in_minutes コマンドを使用します。 • 'sub' (サブジェクト) が、不明なユーザーを呼び出しています。この問題を解決するには、'sub' クレームが "username" であることを確認します。この値は大文字と小文字が区別されます。
67	FEATURE_NOT_ENABLED	オンデマンドアクセスがサポートされていません	オンデマンドアクセスは、ライセンス認証されている Tableau Cloud サイトを介してのみ利用できます。
126	CONNECTED_APP_NOT_FOUND	接続されたアプリが見つかりませんでした	この問題を解決するには、接続されたアプリが有効になっていて、正しいクライアント ID (接続アプリ ID とも呼ばれます) が JWT で参照されていることを確認します。
127	CONNECTED_APP_SECRET_NOT_FOUND	接続されたアプリのシークレットが見つかりませんでした	この問題を解決するには、接続されたアプリの正しいシークレット ID とシークレット値が JWT で参照されていることを確認します。

128	CONNECTED_APP_SECRET_LIMIT_EXCEEDED	シークレットの上限に達しました	接続されたアプリには、最大 2 つのシークレットが許可されます。このエラーは、3 番目のシークレットを作成しようとしたときに発生する可能性があります。 この問題を解決するには、新しいシークレットを作成する前に、接続されたアプリからシークレットを削除します。
133	INVALID_CONNECTED_APP_DOMAIN_SAFELIST	ドメイン許可リストに 1 つまたは複数の無効な文字が含まれています	このエラーは、ドメイン許可リストに 1 つまたは複数の無効な文字が含まれている場合に発生する可能性があります。
10083	BAD_JWT	JWT ヘッダーに問題が含まれています	'kid' (シークレット ID) または 'clientId' (発行元) のクレームが JWT ヘッダーにありません。この問題を解決するには、この情報が含まれていることを確認します。
10084	JWT_PARSE_ERROR	JWT に問題が含まれています	この問題を解決するには、次を確認します。 • この JWT で参照される 'aud' (オーディエンス) 値が "tableau" 値を使用していることを確認します。この値は大文字と小文字が区別されます。 • 'aud' (オーディエンス) と 'sub' (サブジェクト) は JWT に含まれています。 • IssueTime を確認するか、あるいは、接続済みアプリをホストしているマシンと Tableau Cloud との間で時刻の不一致がないこと

			を確認します。
10085	COULD_NOT_FETCH_JWT_KEYS	JWT がキーを見つけることができませんでした	シークレットを見つけることができませんでした。 この問題を解決するには、JWT ヘッダーで正しい「kid」(シークレット ID) が使用されていることを確認します。
10087	BLOCKLISTED_JWS_ALGORITHM_USED_TO_SIGN	JWT 署名アルゴリズムの問題	この問題を解決するには、署名アルゴリズムを削除します。詳細については、vizportal.oauth.external_authorization_server.blocklisted_jws_algorithmsを参照してください。
10089	CONNECTED_APP_NOT_FOUND	接続されたアプリが見つかりませんでした	この問題を解決するには、発行元が、正しい接続済みアプリID (クライアント ID とも呼ばれます) を呼び出していることを確認します。
10090	CONNECTED_APP_DISABLED	接続されたアプリが無効になっています	信頼の確認に使用する接続されたアプリが無効になっています。この問題を解決するには、接続されたアプリを有効にします。
10091	JTI_ALREADY_USED	一意の JWT が必要です	JWT はすでに認証プロセスで使用されています。この問題を解決するには、新しい JWT を生成する必要があります。
10092	NOT_IN_DOMAIN_ALLOW_LIST	埋め込みコンテンツのドメインが指定されていません	この問題を解決するには、設定 `unrestrictedEmbedding` が `true` に設定されていること、または `domainAllowlist` パラメーターに、Tableau REST API の「サイトの埋め込み設定の更新」メソッドを使用して Tableau コンテンツを埋め込むドメインが含まれていることを確認してください。

10094	MISSING_REQUIRED_JTI	JWT ID がありません	この問題を解決するには、'jti' (JWT ID) が JWT に含まれていることを確認します。
10096	JWT_EXPIRATION_EXCEEDS_CONFIGURED_EXPIRATION_PERIOD	有効期限の問題	'exp' (有効期限) が既定の最大有効期間を超えています。この問題を解決するには、有効な JWT に必要な登録済みのクレームを確認し、正しい値を使用するようにします。最大有効期間を変更するには、vizportal.oauth.external_authorization_server.max_expiration_period_in_minutes コマンドを使用します。
10097	SCOPES_MALFORMED	スコープクレームに関する問題	このエラーは、'scp' (スコープ) クレームが JWT にないか、リスト型として渡されていない場合に発生する可能性があります。この問題を解決するには、'scp' が JWT に含まれ、リスト型として渡されていることを確認します。JWT に関するトラブルシューティングのヘルプについては、auth0 サイトの「デバッガー」を参照してください。
10098	JWT_UNSIGNED_OR_ENCRYPTED	JWT が署名されていないか、暗号化されています	Tableau は、署名されていない JWT または暗号化された JWT をサポートしていません。
10099	SCOPES_MISSING_IN_JWT	スコープのクレームがありません	必要な 'scp' (スコープ) のクレームが JWT にありません。この問題を解決するには、'scp' が JWT に含まれていることを確認します。JWT に関するトラブルシューティングのヘルプについては、auth0 サイトの「デバッガー」を参照してください。

10100	JTI_PERSISTENCE_FAILED	予期しない JWT ID エラー	予期しない 'jti' (JWT ID) エラーが発生しました。この問題を解決するには、新しい 'jti' で新しい JWT を生成する必要があります。
10103	JWT_MAX_SIZE_EXCEEDED	JWT が最大サイズを超えています	このエラーは、JWT のサイズが 8000 バイトを超えている場合に発生します。この問題を解決するには、必要なクレームのみが Tableau Server に渡されていることを確認してください。
10105	ORIGIN_HEADER_NOT_A_VALID_URI	無効なオリジンヘッダー	このエラーは、1) URL がドメイン許可リストに指定されている、および 2) オリジンヘッダーに有効な URL が含まれていないことが原因で発生する可能性があります。

データ接続認証

Kerberos、OAuth、およびシングル サインオンを使用してデータ接続認証を構成できます。

詳細

- Tableau Server on Linux - 共有ディレクトリへの接続 (英語) (Tableau コミュニティ)
- TNSNames.ora または LDAP.ora を使用するための Oracle 接続の設定 (Tableau サポート)

Kerberos 委任の有効化

Kerberos 委任を使用すると、Tableau Server はワークブックまたはビューを参照しているユーザーの Kerberos 認証資格情報を使用してそのユーザーのためにクエリを実行できます。これは、次のような場合に役立ちます。

- 誰がデータにアクセスしているかを知る必要があります (ビューアーの名前はデータ ソースのアクセス ログに表示されます)。

- データソースのセキュリティは行レベルですが、別のユーザーは別の行へのアクセス権があります。

サポートされているデータソース

Tableau では次のデータソースを使用する Kerberos 委任がサポートされています。

- Cloudera: Hive/Impala
- Denodo
- Hortonworks
- Oracle
- PostgreSQL
- Spark
- SQL Server
- Teradata
- Vertica

MSAS は Linux プラットフォームではサポートされていません。

要件

Kerberos 委任には Active Directory が必要です。

- Tableau Server のアイデンティティストアが、Active Directory を使用するように構成されている必要があります。
- Tableau Server がインストールされているコンピューターを Active Directory のドメインに結合する必要があります。
- MIT Kerberos はサポートされていません。

Web オーサリングとユーザーの Kerberos 認証

特定のターゲットに対してデータへの接続を構成する場合、優先認証方法として統合認証または Windows 認証を選択できます。ただし、Web 作成シナリオのデフォルトの動作では、Kerberos サービス アカウント(「実行」アカウント) が代わりに使用されます。

Kerberos 委任を使用した Web 作成シナリオでユーザーの認証資格情報を有効にするには、TSM を使用して追加の構成を行う必要があります。次のコマンドを実行します。

```
tsm configuration set -k native_
api.WebAuthoringAuthModeKerberosDelegation -v true

tsm pending-changes apply
```

この構成を行うと、Web オーサリングとの統合認証を選択した場合、Kerberos 委任がデフォルトの操作になります。ただし、この設定によってコンテンツ作成者がサービス アカウントにアクセスすることは妨げられません。作成者は、Tableau Desktop またはその他の方法を使用して、実行サービス アカウントに接続するコンテンツをパブリッシュすることができます。

実行サービス アカウントの詳細については、「Kerberos サービス アカウントへのアクセスの有効化」を参照してください。

構成プロセス

このセクションでは、Kerberos 委任を有効にするためのプロセスの例を示します。また、シナリオには各構成要素間のリレーションシップを説明する際に便利なサンプル名が含まれています。

1. Tableau Server では、データベースへの呼び出しを開始するユーザーの委任に Kerberos サービス チケットが必要となります。指定したデータベースへの委任に使用するドメイン アカウントを作成する必要があります。このアカウントは実行サービス アカウントと呼ばれます。このトピックで委任/実行アカウントとして構成されたサンプル ユーザーは、`tabsrv@example.com` となります。

   このアカウントは、ユーザー ドメインに接続されている Windows Server 上の Active Directory のユーザーおよび PC を使用して構成する必要があります。

   - 実行サービス アカウントの **[プロパティ]** ページを開き、**[委任]** タブをクリックして **[指定サービスの委任用のみ、このユーザーを信頼する]** および **[任意の認証プロトコルを使用する]** を選択します。

2. 実行サービス アカウントのキータブ ファイルを作成します。

   たとえば、ktutil ツールを使用して、次のコマンドでキータブ (`tabsrv-runas.keytab`) を作成します。

   ```
   sudo ktutil
   ```

```
ktutil:  addent -password -p tabsrv@EXAMPLE.COM -k 2 -e
<encryption scheme>
```

このコマンドの暗号化スキームには、RC4-HMAC、aes128-cts-hmac-sha1-96、aes256-cts-hmac-sha1-96 などがあります。環境およびデータソースに適した暗号化スキームについては、IT チームにお問い合わせください。

```
ktutil:  wkt tabsrv-runas.keytab
```

Tableau Server では、実行サービス アカウントおよび関連付けられたキータブを使用して認証を行い、データベースへ直接接続します。

3. Tableau Server データディレクトリにキータブをコピーし、適切な所有権とパーミッションを設定します。マルチノード展開を実行している場合は、クラスター内の各ノードで以下のコマンドを実行する必要があります。

```
mkdir /var/opt/keytab
sudo cp -p tabsrv-runas.keytab /var/opt/keytab
sudo chown $USER /var/opt/keytab/tabsrv-runas.keytab

chgrp tableau /var/opt/keytab/tabsrv-runas.keytab

chmod g+r /var/opt/keytab/tabsrv-runas.keytab
```

4. 次の TSM コマンドを実行して Kerberos 委任を有効にし、委任サービス アカウントを設定してキータブ ファイルとサービス アカウントを関連付けます。

```
tsm configuration set -k wgserver.delegation.enabled -v true
tsm configuration set -k native_api.datasource_impersonation_
runas_principal -v tabsrv@EXAMPLE.COM
tsm configuration set -k native_api.datasource_impersonation_
runas_keytab_path -v /var/opt/keytab/tabsrv-runas.keytab
tsm configuration set -k native_api.protocol_transition_a_d_
short_domain -v false
```

```
tsm configuration set -k native_api.protocol_transition_
uppercase_realm -v true
```

場合によっては、TSM で `--force-keys` と記載されるエラーが返されることがあります。このエラーが発生する場合は、引数に `--force-keys` パラメーターに追加してコマンドを再度実行してください。

5. 次の TSM コマンドを実行して Tableau Server へ変更を適用します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

6. データ接続で委任を有効にします:

   - **SQL Server**—Tableau コミュニティの「SQL Server 用に Kerberos 委任を有効にする」を参照してください。
   - **PostgreSQL**—Tableau コミュニティの「PostgreSQL 用に Kerberos 委任を有効にする」を参照してください。
   - **Teradata**—Tableau コミュニティの「Teradata 用に Kerberos 委任を有効にする」を参照してください。
   - **Denodo**—Tableau コミュニティの「Linux 上の Denodo 用に Kerberos 委任を有効にする」を参照してください。
   - **Oracle**—Tableau コミュニティの「Oracle 用に Kerberos 委任を有効にする」を参照してください。

- **Cloudera**—Tableau コミュニティの「Hive/Impala 用に Kerberos 委任を有効にする」を参照してください。
- **Teradata**—Tableau コミュニティの「Vertica 用に Kerberos 委任を有効にする」を参照してください。

関連項目

Kerberos のトラブルシューティング

JDBC コネクタの Kerberos 委任を有効にする

バージョン 2020.2 の時点で、Tableau Server は JDBC コネクタの Kerberos 委任をサポートしています。

Kerberos 委任を使用すると、Tableau Server はワークブックまたはビューを参照しているユーザーの Kerberos 認証資格情報を使用してそのユーザーのためにクエリを実行できます。これは、次のような場合に役立ちます。

- 誰がデータにアクセスしているかを知る必要があります (ビューアーの名前はデータソースのアクセスログに表示されます)。
- データソースのセキュリティは行レベルですが、別のユーザーは別の行へのアクセス権があります。

サポートされているデータソース

Tableau では次のデータソースを使用する JDBC Kerberos 実行アカウント認証がサポートされています。

- Oracle
- PostgreSQL

ネイティブと JDBC ベースの両方のコネクタは、Linux 上の Tableau Server で同じ構成を使用します。Kerberos 委任の有効化を参照してください。

JDBC コネクタの Kerberos 実行アカウント認証を有効にする

バージョン 2020.2 の時点で、Tableau Server は JDBC コネクタの Kerberos 認証をサポートしています。

データベースへのアクセスに Kerberos サービス アカウントを使用するように Tableau Server を構成することができます。このシナリオでは、Tableau Server がサービス アカウント(「実行サービス アカウント」とも呼ばれます) を使用してデータベースに接続します。このシナリオは、「実行アカウント認証」と呼ばれます。

Tableau Server で実行アカウント認証を使用するには、まず統合認証を使用する Tableau Desktop にワークブックまたはデータ ソースを作成する必要があります。Tableau Server へのパブリッシュを行う際に、実行アカウント認証のオプションが表示されます。Web 作成を使用してデータ ソースを作成する場合、統合認証を選択した場合の既定の操作は、実行アカウント認証です。

サポートされているデータ ソース

Tableau では次のデータ ソースを使用する JDBC Kerberos 委任がサポートされています。

- Oracle
- PostgreSQL

ネイティブと JDBC ベースの両方のコネクタは、Linux 上の Tableau Server で同じ構成を使用します。実行アカウント認証を構成するには、Kerberos サービス アカウントへのアクセスの有効化 を参照してください。

OAuth 接続

Tableau Server では、さまざまなコネクタの OAuth に対応しています。多くの場合、OAuth 機能を実装するために Tableau Server で追加の設定を行う必要ありません。

Tableau から、OAuth を使用するコネクタでデータにサインインすると、ユーザーは認証プロバイダーのサインイン ページにリダイレクトされます。ユーザーが認証資格情報を提示し、Tableau にデータへのアクセスを許可すると、認証プロバイダーは Tableau とユーザーを一意に識別する**アクセス トークン**を Tableau に送信します。このアクセス トークンは、ユーザーに代わってデータにアクセスするために使用されます。詳細については、次のOAuth プロセスの概要を参照してください。

OAuth ベースの接続を使用すると、次のような利点があります。

- **セキュリティ:** データベースの認証資格情報が Tableau Server に知られたり保存されたりすることはなく、アクセス トークンはユーザーに代わって Tableau のみが使用します。
- **利便性:** 複数の場所にデータ ソース ID とパスワードを埋め込む代わりに、そのデータプロバイダーにアクセスするすべてのパブリッシュ済みワークブックとデータ ソースの特定のデータプロバイダーに対して提供されるトークンを使用できます。

> **注:** さらに、Google BigQuery データへのライブ接続の場合、各ワークブックの Viewer は単一のユーザー名とパスワードの認証資格情報を共有するのではなく、ユーザーを識別する一意のアクセス トークンを持つことができます。

OAuth プロセスの概要

次の手順では、OAuth プロセスを呼び出す Tableau 環境におけるワークフローについて説明します。

1. ユーザーが取る行動は、クラウドベースのデータ ソースへのアクセスを必要とします。

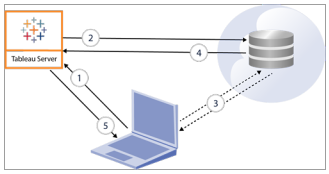
   たとえば、Tableau Serverにパブリッシュされているワークブックを開きます。

2. Tableau は、ユーザーをクラウド上のデータプロバイダーのサインイン ページに誘導します。データプロバイダーに送信される情報により、Tableau は要求サイトとして識別されます。

3. ユーザーがデータにサインインすると、プロバイダーは Tableau Server のデータへのアクセスの認可を確認するようにユーザーに要求します。

4. ユーザーの確認時に、データプロバイダーはアクセス トークンを Tableau Server に送り返します。

5. Tableau Server は、ワークブックとデータをユーザーに提示します。



注: OAuth 接続の 1 回のみ使用可能な更新トークン (1 回限りの更新トークン、ローリング更新トークン、または更新トークンのローテーションと呼ばれることもあります) のサポートは、2025.2 (2025 年夏) リリースで Tableau Cloud に追加され、データソースプロバイダーの IDP でサポートされている場合は、2025.3 リリースの Tableau Server および Tableau Bridge で追加されました。接続に外部の OAuth 構成を使用する場合の詳細については、GitHub の「OAuth の構成と使用方法に関するドキュメント」を参照してください。

次のユーザー ワークフローで OAuth プロセスを使用できます。

- Tableau Desktop または Tableau Server からワークブックを作成してデータソースに接続する。
- Tableau Desktop からデータソースをパブリッシュする。
- Tableau Mobile や Tableau Desktop などの承認済み *クライアント*から Tableau Server にサインインします。

既定の保存済み認証資格情報コネクタ

*保存済み認証資格情報*とは、Tableau Server が OAuth 接続用のユーザー トークンを保存する機能を指します。これにより、ユーザーは OAuth 認証資格情報を Tableau Server 上のユーザープロフィールに保存できます。認証資格情報を保存すると、その後にコネクタにアクセスするときに、パブリッシュ、編集、更新のプロンプトが表示されなくなります。

> **注**: Web 上で Tableau Prep フローを編集する場合、再認証を求められる場合があります。

次のコネクタでは、保存した認証資格情報が既定で使用され、Tableau Server で追加の設定を行う必要はありません。

- Anaplan
- Box
- Dropbox
- Esri ArcGIS サーバー
- Google 広告
- LinkedIn Sales Navigator

- Marketo
- OneDrive および SharePoint Online (2022.3 以降、追加の設定が必要です)
- Oracle Eloqua
- ServiceNow ITSM
- Snowflake - 「プライベート リンク」を使用するには、追加の構成が必要です。詳細については、Snowflake Web サイトの「パートナー アプリケーション用 Snowflake OAuth の構成」および「Snowflake 接続用 OAuth の設定」を参照してください。

次のコネクタは、サーバー管理者が構成を追加することにより、保存された認証資格情報を使用できます。

- Azure Data Lake Storage Gen2、Azure Synapse、Azure SQL データベース、Databricks、OneDrive とSharePoint Online、SharePoint リスト (JDBC)

  詳細については、「OAuth と最新の認証用に Azure AD を構成する」を参照してください。

- Dremio

  詳細については、「Dremio 向け OAuth の設定」を参照してください。

- Google アナリティクス、Google BigQuery、Google ドライブ

  詳細については、「Google での OAuth の設定」を参照してください。

  > **注:** Tableau Server が Google 管理コンソールの [アクセスしたアプリ] リストに表示されていない場合は、クライアント ID を使用して新しいアプリを手動でリストに追加できます。詳細については、「Google OAuth を保存済み認証資格情報に変更する」を参照してください。

- Intuit QuickBooks Online

  詳細については、「Intuit QuickBooks Online 向け OAuth の設定」を参照してください。

  OneDrive および SharePoint Online (2022.3 以降)

  詳細については、「サイトのカスタム OAuth の構成」を参照してください。

- Salesforce

  詳細については、「Salesforce.com OAuth を保存済み認証資格情報に変更するを参照してください。

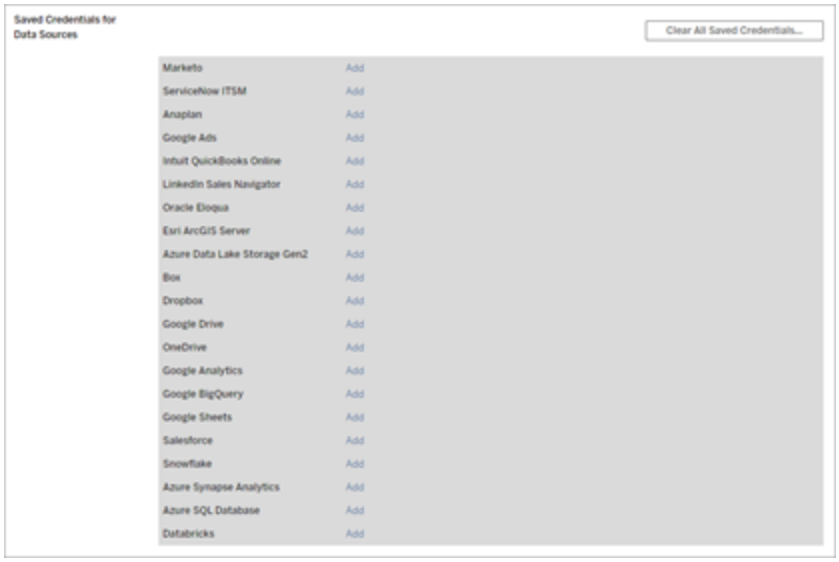
- Salesforce CDP

  詳細については、「Tableau Server を Salesforce Data Cloud に接続する」を参照してください。

- Snowflake

  バージョン 2024.2 以降。詳細については、「Snowflake OAuth を保存された認証資格情報に変更する」を参照してください。

サポートされるすべてのコネクタは、Tableau Server のユーザーの **[マイ アカウント設定]** ページにある **[データ ソースの保存済み認証資格情報]** に一覧表示されます。各コネクタのための保存済み認証資格情報はユーザーが管理します。



データ接続のためのアクセス トークン

初期認証プロセスの後にデータへの直接アクセスを可能にするために、アクセス トークンに基づいて認証資格情報をデータ接続と共に埋め込むことができます。アクセス トークンは Tableau Server ユーザーが削除するか、データプロバイダーが取り消すまで有効です。

データ ソースプロバイダーが許可するアクセス トークンの数を超過する可能性があります。そのような場合、データプロバイダーは トークンがユーザーによって作成される際に、前回のアクセスからの時間の長さを使用して、新しいトークンのためのスペースを空けるために無効にするトークンを決定します。

承認済みクライアントからの認証用のアクセス トークン

既定では、Tableau Server サイトで、ユーザーは初回サインイン時に認証資格情報を入力した後、承認済み Tableau クライアントからサイトに直接アクセスできます。また、このタイプの認証は、OAuth アクセス トークンを使用して、ユーザーの認証資格情報を安全に保存します。

詳細については、「自動クライアント認証を無効にする」を参照してください。

既定の管理対象キーチェーン コネクタ

*管理対象キーチェーン*とは、OAuth トークンがプロバイダーによって Tableau Server に対して生成され、同じサイト内のすべてのユーザーによって共有される機能を指します。まずユーザーがデータ ソースをパブリッシュすると、Tableau Server からデータ ソースの認証資格情報を求めるメッセージが表示されます。Tableau Server は、データ ソースプロバイダーに認証資格情報を送信します。プロバイダーは、使用する Tableau Server の OAuth トークンをユーザーに代わって返します。その後のパブリッシュ操作では、同じクラスとユーザー名に対して Tableau Server に保存された OAuth トークンが使用されるため、ユーザーに OAuth 認証資格情報の入力を求めるプロンプトが表示されることはありません。データ ソースのパスワードが変更された場合、前のプロセスが繰り返され、古いトークンが Tableau Server 上の新しいトークンに置き換えられます。

次の既定の管理対象キーチェーン コネクタでは、Tableau Server で追加の OAuth を設定する必要はありません。

- Google アナリティクス、Google BigQuery、Google スプレッドシート(は 2022 年 3 月にで非推奨)
- Salesforce

トークンの制限とストレージ

Google では、クライアント アプリケーションごとにユーザーあたり50 トークンの制限があります (このシナリオでは、Tableau Server がクライアント アプリケーションです)。OAuth トークンは Tableau Server に格納され、ユーザーによって再利用されるため、ユーザーがトークンの制限を超える可能性は低いです。

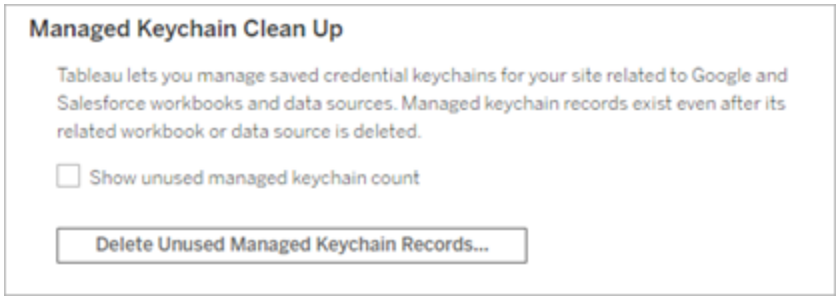
すべてのユーザー トークンは、Tableau Server に保存されるときに暗号化されます。詳細については、サーバー シークレットの管理を参照してください。

未使用のキーチェーン レコードの削除

管理対象のキーチェーン レコードには、dbClass、ユーザー名、および OAuth シークレット属性などの接続属性が含まれます。特定のサイトで管理対象のキーチェーン レコードはすべてマージおよび暗号化されてから、PostgreSQL に保存されます。

レコードは、ワークブックやデータ ソースが削除された場合でも保持されます。これらのレコードは時間の経過とともにサイズが拡大するため、問題を引き起こす可能性があります。

定期的なメンテナンス タスクとして、使用されていないキーチェーン レコードを定期的に削除することをお勧めします。各サイトに保存されているレコード数と未使用レコード数を表示できます。未使用のレコードを削除することもできます。



[Managed Keychain Clean Up (管理対象のキーチェーンのクリーン アップ)] にアクセスするには、Tableau Server 管理ページにサインインして、未使用のレコードを削除するサイトに移動し、**[設定]** をクリックします。

管理対象キーチェーンでのシナリオの制限

Tableau Server で管理対象キーチェーンの OAuth を使用する場合、次の 3 つのシナリオには対応しません。

- ライブ接続で Oauth 認証資格情報の入力を求めるプロンプトを表示する。ユーザーは、ライブ接続時に管理キーチェーン OAuth を使用して認証資格情報を埋め込む必要があります
- Tableau Server での OAuth データソース接続の編集
- Web 作成

管理対象キーチェーンを保存された認証資格情報へ変換

各コネクタの OAuth クライアント ID とシークレットを使用して Tableau Server を構成することで、管理キーチェーンを使用するコネクタを変換し、保存済み認証資格情報を使用することができます。これらのコネクタを保存済み認証資格情報に変換することで、ユーザーは Tableau Server の **[マイアカウント設定]** ページで各コネクタのタイプに対応する認証資格情報を管理できます。さらに、ライブ接続に関するプロンプト、接続の編集、Web 作成にも対応しています。

サイトのカスタム OAuth の設定

コネクタのサブセットの場合、カスタム OAuth クライアントを構成することでサイトレベルの OAuth を構成できます。詳細については、次のいずれかを参照してください。

- Azure Data Lake Storage Gen2、Azure SQL データベース、Azure Synapse、Databricks、OneDrive と SharePoint、SharePoint リスト (JDBC) については、サイトのカスタム OAuth の構成を参照してください。
- Dremio については、「Dremio 向け OAuth の設定」を参照してください。

- Google アナリティクス、Google BigQuery、Google スプレッドシート(Tableau バージョン 2022.1 で非推奨) については、「サイトのカスタム OAuth の構成」を参照してください。
- Salesforce については、「サイトのカスタム OAuth の構成」を参照してください。
- Salesforce CDP については、「Tableau Server を Salesforce Data Cloud に接続する」を参照してください。
- Snowflake については、「オプション 2: サイトごとに Snowflake 接続の OAuth を構成するを参照してください。

保存済みアクセス トークンの許可

OAuth 用の Tableau Server を構成したら、OAuth 認証資格情報の管理をユーザーに許可するか、または一元的に管理するかを決定できます。ユーザーに自分の認証資格情報の管理を許可する場合は、ユーザープロフィール設定をサーバーから有効化する必要があります。

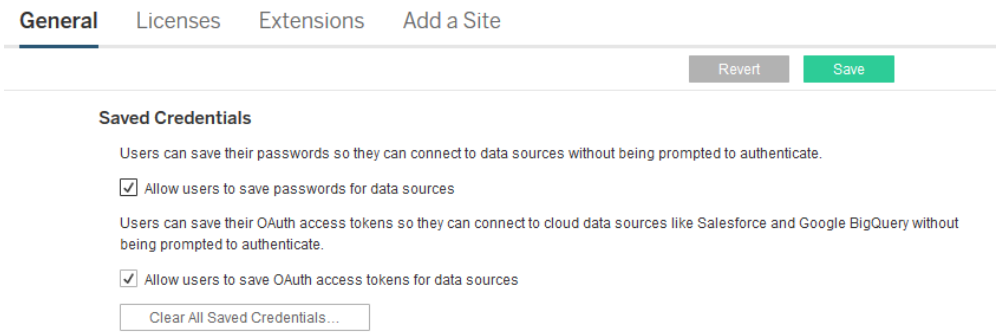
注: OAuth データ接続を有効にするようにサーバーを構成してない場合は、次の関連するトピックを参照してください。

注: 現在 Tableau への OAuth 接続では、使い捨ての更新トークンはサポートされていません。ほとんどの場合、代わりにローリング更新トークンを使用するように ID プロバイダー (Okta や Redshift IDC など) を設定できます。詳細については、プロバイダーの OAuth ドキュメントを参照してください。

1. Tableau Server にサーバー管理者としてサインインします。
2. **単一のサイト:[設定] > [全般]** をクリックします。

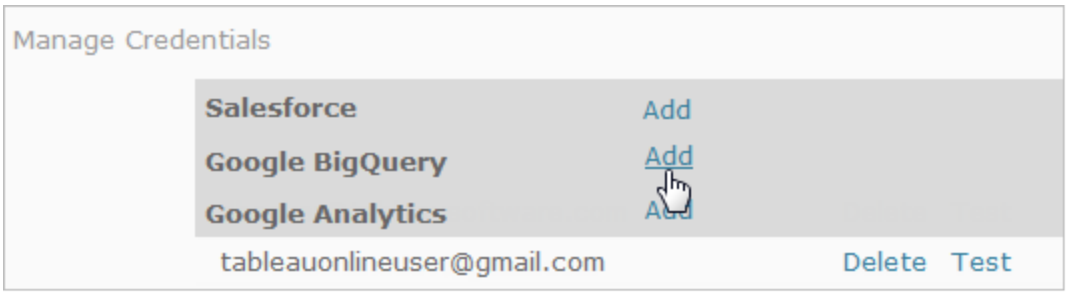
   マルチサイト:サイト メニューで、**[すべてのサイトを管理] > [設定] > [全般]** の順にクリックします。
3. **[保存された認証資格情報]** セクションで、以下を選択します。

- **[データ ソースのパスワードの保存をユーザーに許可]** (ユーザーは、個別の認証資格情報をデータ ソースとともに保存できます)。
- **ユーザーがデータ ソースの OAuth アクセス トークンを保存できるようにします**



4. **[保存]** をクリックします。

これらのチェック ボックスを選択すると、ユーザーのプロファイル設定には **[認証資格情報の管理]** セクションが表示され、ここでユーザーは OAuth データ接続用のアクセス トークンを追加できます。



認証資格情報の一元管理

サーバー管理者は、OAuth 認証資格情報を一元的に管理することもできます。これは、たとえば、複数のユーザーが同じデータから操作を行い、データプロバイダー向けの専用のユーザー アカウントがある場合に適しています。

認証資格情報を一元管理するには、次の手順を行います。

- 前の手順で説明したチェックボックスをオフにします。
- データソースをパブリッシュしたあとに、データ情報を編集します。

  接続を編集するときは、個々のユーザー名とパスワードではなく、OAuth アクセストークンを使用する認証資格情報を埋め込みます。

パスワードとアクセストークンの保存の設定が有効でない場合、ユーザーのプロファイル設定に [認証資格情報の管理] セクションは表示されません。

関連項目

Google での OAuth の設定

Salesforce.com OAuth を保存済み認証資格情報に変更する

Snowflake 接続用 OAuth の設定

Intuit QuickBooks Online 向け OAuth の設定

Salesforce.com OAuth を保存済み認証資格情報に変更する

既定では、Salesforce.com のコネクタでは、データプロバイダーにより Tableau Server 用に生成された、同一サイト内ですべてのユーザーが共有する OAuth トークンの管理キーチェーンを使用します。保存されたクライアント ID とクライアントシークレットを使用して Tableau Server を構成できます。このような場合、3 つのシナリオが考えられます。

- **Salesforce データコネクタ**- Salesforce コネクタを使用する場合は、OAuth クライアントの ID とシークレットを使用して Tableau Server を構成できるため、保存済みの認証資格情報をコネクタに使用できます。
- **CRM Analytics への書き込み**-Tableau Prep のフローデータを Salesforce CRM Analytics (バージョン 2022.3 以降) に書き込む場合は、OAuth クライアント ID とシークレットを使用して Tableau Server を構成し、保存した認証資格情報を使用して Tableau Server でフローを実行できるようにします。
- **Einstein Discovery** - Einstein Discovery 拡張機能を Tableau Server と統合する場合は、OAuth クライアントの ID とシークレットを設定する必要があります。Einstein Discovery

とTableau Server を統合する機能は、バージョン2021.1.0 で追加されました。詳細については、「Einstein Discovery 統合の構成」を参照してください。

このトピックでは、OAuth で保存済み認証資格情報を使用するために、Salesforce.com のデータソースとEinstein Discovery の拡張機能を設定する方法を説明します。各Tableau Serverインスタンスでこれらのステップを実行します。

管理キーチェーンと保存済み認証資格情報の詳細については、「OAuth 接続」を参照してください。

注:

- Salesforce のコネクタには、管理キーチェーン(既定)、サーバー全体のOAuth、またはサイト固有のOAuth のいずれかが必要です。
- 保存済み認証資格情報をサイトに使用するには、最初にサーバー全体のOAuth を構成する必要があります。
- サーバー全体のOAuth は、サイト全体のOAuth が構成されているかどうかに関係なく使用できます。
- サイト固有のOAuth を使用する場合は、各サイトを個別に構成する必要があります。
- ライブ接続プロンプト、接続の編集、およびWeb 作成をサポートするには、管理キーチェーンを保存済み認証資格情報に変換して、エラーを回避します。

手順の概要

OAuth を設定するには、次の3 つの一般的なステップを実行します。

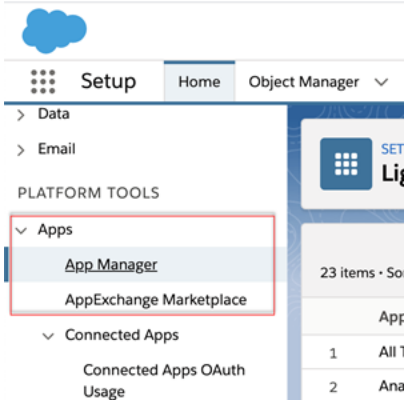
1. Salesforce で接続アプリを作成します。
2. ステップ1 で取得した情報を使用してTableau Server を構成します。
3. (オプション) サイト固有のOAuth を構成します。

ステップ1: Salesforce に接続されたアプリを作成する

注: この手順では、Salesforce Lightning でのプロセスについて説明します。従来のインターフェイスを使用している場合、ナビゲーションは異なる場合がありますが、設定は同じです。

1. Salesforce.com 開発者アカウントにサインインし、右上のユーザー名をクリックしてから**[設定]** を選択します。

2. 左のナビゲーション列の **[アプリ]** で、**[アプリマネージャー]** を選択します。



3. [接続済みアプリ] セクションで、**[新しい接続済みアプリ]** をクリックします。

4. **[基本情報]** でアプリに名前を付け、[api] フィールドをタブで移動して正しい形式で自己入力されるようにし、アプリの連絡先メールアドレスを入力します。

5. **API の [OAuth 設定を有効にする]** セクションで、**[OAuth 設定を有効にする]** を選択します。

6. 表示された新しい OAuth 設定で、**[コールバック URL]** に `https` プロトコルを使用してサーバーの完全修飾ドメイン名を入力し、URL に `auth/add_oauth_token` を追加します。

   例は次のとおりです。

   ```
   https://www.your_tableau_server.com/auth/add_oauth_token
   ```

7. 次の項目を **[利用可能な OAuth スコープ]** から **[選択した OAuth スコープ]** に移動します。

   - **アイデンティティ URL サービス (Id、プロフィール、メール、住所、電話番号) にアクセスする**
   - **API を介してユーザー データを管理する (api)**
   - **いつでもリクエストを実行できる (refresh_token、オフライン アクセス)**

8. **[保存]** をクリックします。

アプリを保存すると、Tableau Server の構成に使用する次の ID が Salesforce の API セクションに表示されます。

- **コンシューマー キー**
- **コンシューマー シークレット**
- **コールバック URL**

ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する

接続済みアプリが Salesforce で作成され、カスタマー キー、カスタマー シークレット、およびコールバック URL を取得したら、Salesforce データの接続と出力、および Einstein Discovery 用に Tableau Server を構成できます。

1. Tableau Server コンピューターのコマンドプロンプトで、次のコマンドを実行します。

   ```
   tsm configuration set -k oauth.salesforce.client_id -v <your_customer_key>
   ```

   ```
   tsm configuration set -k oauth.salesforce.client_secret -v <your_customer_secret>
   ```

   ```
   tsm configuration set -k oauth.salesforce.redirect_uri -v <your_redirect_URL>
   ```

2. (任意) 既定のログイン サーバーを変更するには、次のコマンドを入力してください。

   ```
   tsm configuration set -k oauth.salesforce.server_base_url -v <URL>
   ```

3. 次のコマンドを入力して変更を適用します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

サイトのカスタム OAuth の構成

サイトに対してカスタム Salesforce OAuth クライアントを構成できます。

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

カスタム OAuth クライアントが構成されている場合、サイトレベルの構成はサーバー側の構成よりも優先され、作成されたすべての新しい OAuth 認証資格情報では既定でサイトレベルの OAuth クライアントが使用されます。構成を有効にするために、Tableau Server を再起動する必要はありません。

重要: カスタム OAuth クライアントを構成する前に確立された既存の OAuth 認証資格情報は一時的に使用できますが、データアクセスが中断されないように、サーバー管理者とユーザーの双方が保存された認証資格情報を更新する必要があります。

1: OAuth クライアント ID、クライアントシークレット値、リダイレクト URL の準備

カスタム OAuth クライアントを構成する前に、以下の情報が必要です。この情報を準備したら、サイトのカスタム OAuth クライアントを登録できます。

- **OAuth クライアント ID とクライアントシークレット値:** 最初に OAuth クライアントをデータプロバイダー (コネクタ) に登録して、Tableau Server 用に生成されたクライアントの ID とシークレット値を取得します。
- **リダイレクト URL:** 正しいリダイレクト URL であることを確認します。これは、以下の**ステップ 2** の登録プロセスで必要になります。

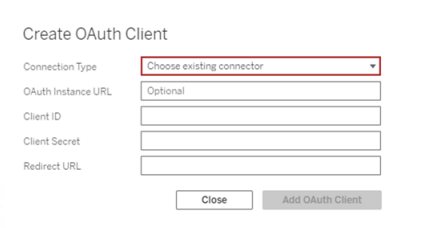
  https://<your_server_name> .com / auth / add_oauth_token

  例: https://example.com/auth/add_oauth_token

2: OAuth クライアント ID とクライアント シークレットの登録

以下の手順に従って、カスタム OAuth クライアントをサイトに登録します。

1. 管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
2. [OAuth Clients Registry (OAuth クライアント レジストリ)] で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックします。
3. 上記の**ステップ 1** の情報を含む、必要な情報を入力します。
   a. **[接続タイプ]** で、カスタム OAuth クライアントを設定するコネクタを選択します。
   b. 複数の OAuth クライアントを登録する場合は、**[OAuth インスタンス URL]** に情報を入力する必要があります。それ以外の場合は、任意で入力してください。
   c. **クライアント ID**、**クライアント シークレット**、**リダイレクト URL** には、上記の**ステップ 1** で準備した情報を入力します。
   d. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックして、登録プロセスを完了します。



4. (オプション) サポートされているすべてのコネクタに対して、ステップ 3 を繰り返します。
5. [設定] ページの下部または上部にある **[保存]** ボタンをクリックして、変更を保存します。

3: 保存した認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイト ユーザー) は、以前に保存した認証資格情報を削除してから、サイトのカスタム Oauth クライアントを使用できるよう

に再度追加する必要があります。

1. **[マイ アカウント設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。
   a. 上記の**ステップ 2** でカスタム OAuth クライアントを構成したコネクタの既存の、保存済み認証資格情報の横にある**[削除]** をクリックします。
   b. コネクタ名の横にある**[追加]** をクリックし、プロンプトに従って 1) 上記の**ステップ 2** で構成したカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

4: 保存した認証資格情報を更新するようにユーザーに通知する

上記の**ステップ 2** で構成したカスタム OAuth クライアントのコネクタの保存済み認証資格情報を更新するようにサイト ユーザーに通知します。サイトのユーザーは、保存済み認証資格情報の更新で説明されている手順を使用して、保存された認証資格情報を更新できます。

アクセス トークンの管理

OAuth 用のサーバーを構成したら、プロファイル設定でのアクセス トークンの管理をユーザーに許可するか、トークンを一元的に管理できます。詳細については、保存済みアクセス トークンの許可を参照してください。

OAuth と最新の認証用に Azure AD を構成する

Azure Synapse、Azure SQL Database、Azure Databricks、Azure Data Lake Gen2、OneDrive、SharePoint Online、SharePoint Lists (JDBC) のコネクタは、Tableau Server の OAuth クライアントを設定することで、Azure AD を介した認証を行うことができます。

注: 現時点では、Tableau への OAuth 接続では、1 回のみ使用可能な更新トークン (ローリング更新トークンまたは更新トークンのローテーションと呼ばれることもあります) はサポートされていません。これらのトークンは今後のリリースでサポートされる予定です。

注: Azure AD の OAuth サポートは、Microsoft SQLServer ドライバー 17.3 以降でのみサポートされています。

ステップ 1: Azure 用の OAuth クライアントを登録する

以下の手順に従って、特定の Azure テナントで Azure 用の OAuth アプリケーションを登録および設定します。

1. Azure ポータルにサインインします。
2. 複数のテナントにアクセスできる場合は、アプリケーションを登録するテナントを選択します。
3. **Azure Active Directory** を検索して選択します。
4. **[管理]** で **[アプリの登録]** を選択し、**[新規登録]** を選択します。
5. 「Tableau Server Oauth」または同様の値を **[名前]** として入力します。
6. アプリ登録ページの **[サポートされているアカウントの種類]** フィールドで、このアプリケーションを使用できるユーザーを選択します。

   注: アプリケーションのクライアント ID とクライアント シークレット値を別のテナントのアカウントに使用する場合は、2 番目のオプション (マルチテナント) を選択してください。

7. **[リダイレクト URI]** (オプション) フィールドで **[Web]** を選択し、サーバーのインターネット アドレスに文字列 `/auth/add_oauth_token` を追加して入力します。

   例: `https://your_server_url.com/auth/add_oauth_token`

8. **[登録]** を選択します。登録が完了すると、Azure ポータルに、アプリケーション (クライアント) ID を含むアプリ登録の [概要] ウィンドウが表示されます。*クライアント ID* とも呼ばれるこの値は、Microsoft ID プラットフォームでアプリケーションを一意に識別します。
9. この ID をコピーして、以下のステップで `[your_client_id]` フィールドとして使用します。
10. 左側のバーで **[証明書とシークレット]** を選択し、**[新しいクライアント シークレット]** を選択します。
11. シークレットの説明を追加します。
12. **[クライアント シークレットの有効期間]** を選択します。

13. **[追加]** を選択し、シークレット値をコピーします。このシークレット値は、以下のステップで `[your_client_secret_value]` として使用します。
14. 左側のバーで **[API パーミッション]** を選択します。
15. **[パーミッションの追加]** を選択します。
16. **[Microsoft グラフ]** を選択します。
17. **[委任されたパーミッション]** を選択します。
18. **[パーミッションの選択]** で、すべての OpenId 権限 (email、offline_access、openid、profile) を選択します。
19. **[パーミッションの追加]** を選択します。
20. その他のパーミッションを追加します。有効にするコネクタについて、次の手順を行います。
    - Azure SQL データベース
      a. **[パーミッションの追加]** をクリックします。
      b. **[マイ API]** を選択します。
      c. **[Azure SQL データベース]** を選択し、**[委任されたパーミッション]** を選択します。
      d. **[user_impersonation]** を選択し、**[パーミッションの追加]** をクリックします。
    - OneDrive および SharePoint Online
      a. **[パーミッションの追加]** をクリックします。
      b. **[Microsoft グラフ]** を選択します。
      c. **[委任されたパーミッション]** をクリックします。
      d. **[パーミッションの選択]** のフィルター検索ボックスに、次のパーミッションを入力して追加します。
         - Files.Read.All
         - Sites.Read.All
         - User.Read
    - SharePoint リスト (JDBC)
      a. **[パーミッションの追加]** をクリックします。
      b. **[Microsoft グラフ]** を選択します。
      c. **[委任されたパーミッション]** をクリックします。
      d. **[パーミッションの選択]** のフィルター検索ボックスに User.Read パーミッションを入力して追加します。
      e. もう一度 **[パーミッションの追加]** をクリックします。
      f. **[SharePoint]** を選択します。
      g. **[委任されたパーミッション]** をクリックします。
      h. **[AllSites]** セクションを展開し、AllSites.Manage パーミッションを選択して追加します。

ステップ 2: Azure 用に Tableau Server を構成する

Tableau Server を構成するには、Tableau Server Manager (TSM) コマンドを実行する必要があります。Azure Data Lake Storage Gen2 では、Azure Synapse、Azure SQL Database、または Azure Databricks などで実行する一般的なコマンドとは異なるコマンドセットが必要です。

Azure Data Lake StorageGen2 用の既定の OAuth クライアントを構成する

Data Lake Storage Gen2 用に Tableau Server を構成するには、次の構成パラメーターが必要です。

- **Azure OAuth クライアント ID:** このクライアント ID はステップ 1 の手順で生成されます。1 番目の tsm コマンドで、この値を `[your_client_id]` にコピーします。
- **Azure OAuth クライアント シークレット値:** このクライアント シークレット値は、ステップ 1 の手順で生成されます。この値を 2 番目の tsm コマンドの `[your_client_secret_value]` にコピーします。
- **Tableau Server URL**: Tableau Server URL を入力します。例: `https://myco.com`3 番目の tsm コマンドで、この値を `[your_server_url]` にコピーします。

次の tsm コマンドを実行して、Azure Data Lake Storage Gen2 用に Tableau Server の OAuth を構成します。

- `tsm configuration set -k oauth.azuredatalake_storage_gen2.client_id -v [your_client_id] --force-keys`
- `tsm configuration set -k oauth.azuredatalake_storage_gen2.client_secret -v [your_client_secret_value] --force-keys`
- `tsm configuration set -k oauth.azuredatalake_storage_gen2.redirect_uri -v [your_server_url]/auth/add_oauth_token --force-keys`
- `tsm pending-changes apply`

Azure Synapse、Azure SQL Database、Azure Databricks 用の既定のクライアントを構成する

Tableau Server を構成するには、次の構成パラメーターが必要です。

- **Azure OAuth クライアント ID:** ステップ 1 の手順で生成されます。この値を tsm コマンドの `[your_client_id]` にコピーします。
- **Azure OAuthClient シークレット値:** ステップ 1 の手順で生成されます。この値を 2 番目の tsm コマンドの `[your_client_secret_value]` にコピーします。
- **Tableau Server URL:** お使いの Tableau Server の URL を入力してください。例: `https://myco.com`
- **構成 ID:** 以下の tsm コマンドのパラメーター `oauth.config.id` の値です。有効な値は次のとおりです。
    - Azure Synapse: `azure_sql_dw`
    - Azure SQL Database: `azure_sqldb`
    - Databricks: `databricks`

次の tsm コマンドを実行して、Azure Synapse、Azure SQL Database、または Azure Databricks 用に Azure AD を構成します。たとえば、Azure Synapse を設定する場合は、次のようになります。

```
tsm configuration set -k oauth.config.clients -v "
[{\"oauth.config.id\":\"azure_sql_dw\", \"oauth.config.client_
id\":\"[your_client_id]\", \"oauth.config.client_secret\":\"[your_
client_secret_value]\", \"oauth.config.redirect_uri\":\"[your_
server_url]/auth/add_oauth_token\"}]" --force-keys

tsm pending-changes apply
```

OneDrive および SharePoint Online の既定の OAuth クライアントを構成する

OneDrive および SharePoint Online 用に Tableau Server を構成するには、次の構成パラメーターが必要です。

- **Azure OAuth クライアント ID:** このクライアント ID はステップ 1 の手順で生成されます。1 番目の tsm コマンドで、この値を [your_client_id] にコピーします。

- **Azure OAuth クライアント シークレット値:** このクライアント シークレット値は、ステップ 1 の手順で生成されます。この値を 2 番目の tsm コマンドの [your_client_secret_value] にコピーします。
- **Tableau Server URL:** お使いの Tableau Server の URL を入力してください。例: `https://myco.com`3 番目の tsm コマンドで、この値を [your_server_url] にコピーします。

次の tsm コマンドを実行して、OneDrive および SharePoint Online 用に Tableau Server の OAuth を構成します。

- `tsm configuration set -k oauth.onedrive_and_sharepoint_online.client_id -v [your_client_id] --force-keys`
- `tsm configuration set -k oauth.onedrive_and_sharepoint_online.client_secret -v [your_client_secret_value] --force-keys`
- `tsm configuration set -k oauth.onedrive_and_sharepoint_online.redirect_uri -v [your_server_url]/auth/add_oauth_token --force-keys`
- `tsm pending-changes apply`

SharePoint リスト用 (JDBC) の既定の OAuth クライアントを構成する

SharePoint リスト (JDBC) 用に Tableau Server を構成するには、次の構成パラメーターが必要です。

- **Azure OAuth クライアント ID**: このクライアント ID はステップ 1 の手順で生成されます。1 番目の tsm コマンドで、この値を [your_client_id] にコピーします。
- **Azure OAuth クライアント シークレット値:** このクライアント シークレット値は、ステップ 1 の手順で生成されます。この値を 1 番目の tsm コマンドの [your_client_secret_value] にコピーします。
- **Tableau Server URL:** お使いの Tableau Server の URL を入力してください。例: `https://myco.com`1 番目の tsm コマンドで、この値を [your_server_url] にコピーします。

次の tsm コマンドを実行して、SharePoint リスト (JDBC) 用に Tableau Server の OAuth を構成します。

- `tsm configuration set -k oauth.config.clients -v "[{\"oauth.config.id\":\"cdata_sharepoint\", \"oauth.config.client_id\":\"[your_client_id]\", \"oauth.config.client_secret\":\"[your_client_secret_value]\", \"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_token\"}]" --force-keys`
- `tsm pending-changes apply`

OneDrive 用 (非推奨) の既定の OAuth クライアントを構成する

OneDrive (非推奨) 用に Tableau Server を構成するには、次の構成パラメーターが必要です。

- **Azure OAuth クライアント ID:** このクライアント ID はステップ 1 の手順で生成されます。1 番目の tsm コマンドで、この値を [your_client_id] にコピーします。
- **Azure OAuth クライアント シークレット値:** このクライアント シークレット値は、ステップ 1 の手順で生成されます。この値を 2 番目の tsm コマンドの [your_client_secret_value] にコピーします。
- **Tableau Server URL:** お使いの Tableau Server の URL を入力してください。例: `https://myco.com`3 番目の tsm コマンドで、この値を [your_server_url] にコピーします。

次の tsm コマンドを実行して、OneDrive用 (非推奨) に Tableau Server の OAuth を構成します。

- `tsm configuration set -k oauth.onedrive.client_id -v [your_client_id] --force-keys`
- `tsm configuration set -k oauth.onedrive.client_secret -v [your_client_secret_value] --force-keys`
- `tsm configuration set -k oauth.onedrive.redirect_uri -v [your_server_url]/auth/add_oauth_token --force-keys`
- `tsm pending-changes apply`

サーバーの再起動シナリオ

既定の OAuth クライアントを構成した後に、次のシナリオが発生する可能性があります。

- 保留中の変更を適用する際に、サーバーの再起動が必要になる場合は、再起動プロンプトが表示されます。

- `--ignore-prompt` オプションを使用してプロンプトを表示しないようにすることができます。ただし、これを行っても、再起動は停止されません。
- 変更を適用する際に、サーバーの再起動が必要ない場合は、プロンプトは表示されずに変更が適用されます。詳細については、「tsm pending-changes apply」を参照してください。

複数のコネクタの設定

複数のコネクタを設定する場合は、すべてを 1 つのコマンドにまとめる必要があります。例:

```
tsm configuration set -k oauth.config.clients -v "
[{\"oauth.config.id\":\"azure_sql_dw\", \"oauth.config.client_
id\":\"[your_client_id]\", \"oauth.config.client_secret\":\"[your_
client_secret_value]\", \"oauth.config.redirect_uri\":\"[your_
server_url]/auth/add_oauth_token\"}, {\"oauth.config.id\":\"azure_
sqldb\", \"oauth.config.client_id\":\"[your_client_id]\",
\"oauth.config.client_secret\":\"[your_client_secret_value]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}, {\"oauth.config.id\":\"databricks\",
\"oauth.config.client_id\":\"[your_client_id]\",
\"oauth.config.client_secret\":\"[your_client_secret_value]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}]" --force-keys

tsm pending-changes apply
```

サイトのカスタム OAuth の構成

サイトに対して、カスタム Azure Data Lake Storage Gen2、Azure Synapse、Azure SQL Database、Databricks OAuth、OneDrive および Sharepoint オンライン、Sharepoint リスト (JDBC) のクライアントを構成できます。

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

カスタム OAuth クライアントが構成されている場合、サイトレベルの構成はサーバー側の構成よりも優先され、作成されたすべての新しい OAuth 認証資格情報では既定でサイトレベルの OAuth クライアントが使用されます。構成を有効にするために、Tableau Server を再起動する必要はありません。

重要: カスタム OAuth クライアントを構成する前に確立された既存の OAuth 認証資格情報は一時的に使用できますが、データアクセスが中断されないように、サーバー管理者とユーザーの双方が保存された認証資格情報を更新する必要があります。

1: OAuth クライアント ID、クライアント シークレット値、リダイレクト URL の準備

カスタム OAuth クライアントを構成する前に、以下の情報が必要です。この情報を準備したら、サイトのカスタム OAuth クライアントを登録できます。

- **OAuth クライアント ID とクライアント シークレット値:** 最初に OAuth クライアントをデータプロバイダー (コネクタ) に登録して、Tableau Server 用に生成されたクライアントの ID とシークレット値を取得します。
- **リダイレクト URL:** 正しいリダイレクト URL であることを確認します。これは、以下の**ステップ 2** の登録プロセスで必要になります。

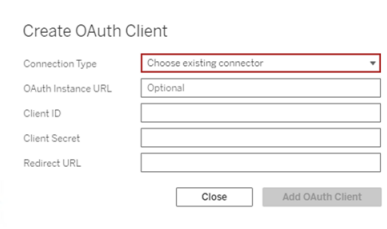
  https://<your_server_name> .com / auth / add_oauth_token

  例: https://example.com/auth/add_oauth_token

2: OAuth クライアント ID とクライアント シークレットの登録

以下の手順に従って、カスタム OAuth クライアントをサイトに登録します。

1. 管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
2. [OAuth Clients Registry (OAuth クライアントレジストリ)] で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックします。
3. 上記の**ステップ 1** の情報を含む、必要な情報を入力します。
   a. **[接続タイプ]** で、カスタム OAuth クライアントを設定するコネクタを選択します。
   b. 複数の OAuth クライアントを登録する場合は、**[OAuth インスタンス URL]** に情報を入力する必要があります。それ以外の場合は、任意で入力してください。
   c. **クライアント ID**、**クライアント シークレット**、**リダイレクト URL** には、上記の**ステップ 1** で準備した情報を入力します。
   d. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックして、登録プロセスを完了します。



4. (オプション) サポートされているすべてのコネクタに対して、ステップ 3 を繰り返します。
5. [設定] ページの下部または上部にある **[保存]** ボタンをクリックして、変更を保存します。

3: 保存した認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイトユーザー) は、以前に保存した認証資格情報を削除してから、サイトのカスタム Oauth クライアントを使用できるように再度追加する必要があります。

1. **[マイ アカウント設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。
    a. 上記の**ステップ 2** でカスタム OAuth クライアントを構成したコネクタの既存の、保存済み認証資格情報の横にある **[削除]** をクリックします。
    b. コネクタ名の横にある **[追加]** をクリックし、プロンプトに従って 1) 上記の**ステップ 2** で構成したカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

4: 保存した認証資格情報を更新するようにユーザーに通知する

上記の**ステップ 2** で構成したカスタム OAuth クライアントのコネクタの保存済み認証資格情報を更新するようにサイト ユーザーに通知します。サイトのユーザーは、保存済み認証資格情報の更新で説明されている手順を使用して、保存された認証資格情報を更新できます。

Snowflake 接続用 OAuth の設定

使用している Tableau のバージョンと更新するサイトの数に応じて、Snowflake 接続用に OAuth を設定する方法は複数あります。このトピックでは、利用可能な各オプションの設定について説明します。

- Tableau バージョン 2024.1 までのすべてのバージョンでは、Tableau Snowflake コネクタは既定で、共通のクライアント ID とシークレットを使用する AWS (GALOP) でホスティングされている OAuth プロキシを使用します。
- Tableau 2020.4以降では、オプションで Tableau Server を設定して、Tableau のインスタンスと同じ場所で実行される新しい OAuth サービスを使用できます。これには、カスタム OAuth と呼ばれる独自のクライアント ID とシークレットを提供する必要があります。
- 2024.2 以降、GALOP プロキシは非推奨となり、認証にローカル OAuth サービスが使用されるようになるため、カスタム OAuth を設定する以下の手順が**必須**になります。
- Tableau Desktop および Tableau Cloud のバージョン 2024.3 以降では、Snowflake にサードパーティの IdP (外部 OAuth) を構成できます。詳細については、「Snowflake の外部 OAuth」を参照してください。

カスタム OAuth によって得られる利点は次のとおりです。

- セキュリティの向上
- OAuth プロキシ (GALOP) に接続できない隔離された環境でも OAuth を使用可能
- AWS Private Link や Azure Private Link の VPC で OAuth フローを実行するために、GALOP の IP アドレスを許可リストに登録する必要がない

OAuth クライアントを Snowflake に登録する

Tableau Server でカスタム OAuth 設定を使用するには、まず OAuth クライアントを登録し、クライアント ID とクライアントシークレットを取得してから、設定を実施する必要があります。Tableau Server バージョン 2024.2 以降では、使用している設定オプションに関係なく、この手順は必須です。Snowflake でカスタム OAuth クライアントを登録するには、「カスタム クライアント用に Snowflake Oauth を設定する」の手順に従います。

登録後、次の Snowflake パラメーターを使用して Tableau Server を構成します。

- アカウントインスタンス URL
- クライアント ID
- クライアントシークレット
- リダイレクト URL

注: リダイレクト URL は、Snowflake 側と Tableau 側の両方で入力した場合と同じになります。形式は次のとおりです。
https://your_server_url.com/auth/add_oauth_token
例: https://example.com/auth/add_oauth_token

オプション 1: TSM を使用して Snowflake 接続の OAuth を設定する

複数のサイトを一度に更新する必要がある場合は、このオプションを使用することをお勧めします。

注: この設定オプションは、Tableau Cloud では使用できません。

1. (バージョン 2024.1 以前) Tableau Server コンピューターで次のコマンドを実行し、Snowflake OAuth サービスを有効にします。

   ```
   tsm configuration set -k native_api.enable_snowflake_
   privatelink_on_server -v true
   ```

   注: バージョン 2024.2 以降では、Snowflake プライベート接続を使用しているかどうかに関係なく、手順 1 をスキップします。

2. テキストエディターで、次のコマンドをコピーして貼り付け、カスタマイズします。

   注: Tableau Server 2021.1 以降でこの設定を変更する場合、oauth.snowflake.client 値の形式が変更されていることに注意してください。

   ```
   tsm configuration set -k oauth.snowflake.clients -v "
   [{\"oauth.snowflake.instance_
   url\":\"https://account.snowflakecomputing.com\",
   \"oauth.snowflake.client_id\":\"client_id_string\",
   \"oauth.snowflake.client_secret\":\"client_secret_string\",
   \"oauth.snowflake.redirect_uri\":\"http://your_server_
   url.com/auth/add_oauth_token\" }]"
   ```

   `oauth.snowflake.clients` キーはキーペアの配列を受け取ります。キーペアの各要素は二重引用符でカプセル化する必要があります。二重引用符は、`\"` としてエスケープする必要があります。

   複数のアカウントインスタンス URL を指定するには、次の例のように、中かっこ ({}) で囲まれた追加の各 OAuth クライアントをコンマ (,) で区切ります。

```
tsm configuration set -k oauth.snowflake.clients -v "
[{\"oauth.snowflake.instance_
url\":\"https://account.snowflakecomputing.com\",\"oauth.snowf-
lake.client_id\":\"client_id_
string1\",\"oauth.snowflake.client_secret\":\"client_secret_
string1\",\"oauth.snowflake.redirect_uri\":\"http://your_
server_url.com/auth/add_oauth_token\" },
{\"oauth.snowflake.instance_
url\":\"https://account2.snowflakecomputing.com\",\"oauth.snow-
flake.client_id\":\"client_id_
string2\",\"oauth.snowflake.client_secret\":\"client_secret_
string2\",\"oauth.snowflake.redirect_uri\":\"http://your_
server_url.com/auth/add_oauth_token\" }]"
```

次に示すように、各キーの値を置き換えます。

- アカウントインスタンスURL: `oauth.snowflake.instance_url`
- クライアントID: `oauth.snowflake.client_id`
- クライアントシークレット: `oauth.snowflake.client_secret`
- リダイレクトURL: `oauth.snowflake.redirect_uri`

> **注**: コマンドを実行する前に、構文を注意深く検証してください。TSMはこの入力を検証しません。

コマンドをTSM CLIにコピーして、コマンドを実行します。

3. 次のコマンドを入力して変更を適用します。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply`コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--

`ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

オプション 2: サイトごとに Snowflake 接続の OAuth を構成する

注: このオプションは、バージョン 2024.2 以降の Tableau Cloud でも使用できます。

Tableau Server UI を使用して、サイトレベルでカスタムの Snowflake OAuth クライアントを設定できます。

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

カスタム OAuth クライアントが構成されている場合、サイトレベルの構成はサーバー側の構成よりも優先され、作成されたすべての新しい OAuth 認証資格情報では既定でサイトレベルの OAuth クライアントが使用されます。構成を有効にするために、Tableau Server を再起動する必要はありません。

重要: カスタム OAuth クライアントを構成する前に確立された既存の OAuth 認証資格情報は一時的に使用できますが、データアクセスが中断されないように、サーバー管理者とユーザーの双方が保存された認証資格情報を更新する必要があります。

1: OAuth クライアント ID、クライアントシークレット、リダイレクト URL の準備

カスタム OAuth クライアントを構成する前に、以下の情報が必要です。この情報を準備したら、サイトのカスタム OAuth クライアントを登録できます。詳細については、**「Snowflake に OAuth クライ**

アントを登録する]のセクションを参照してください。

- **OAuth クライアント ID とクライアント シークレット:** 最初に OAuth クライアントをデータプロバイダー (コネクタ) に登録して、Tableau Server 用に生成されたクライアント ID とシークレットを取得します。
- **リダイレクト URL:** 正しいリダイレクト URL であることを確認します。これは、以下の**ステップ 2** の登録プロセスで必要になります。

  https://<your_server_name> .com / auth / add_oauth_token

  例: https://example.com/auth/add_oauth_token

2: OAuth クライアント ID とクライアント シークレットの登録

以下の手順に従って、カスタム OAuth クライアントをサイトに登録します。

1. (バージョン 2024.1 以前) Tableau Server コンピューターで次の 2 つのコマンドを実行し、Snowflake OAuth サービスを有効にします。

   ```
   tsm configuration set -k native_api.enable_snowflake_
   privatelink_on_server -v true

   tsm pending-changes apply
   ```

   注: バージョン 2024.2 以降では、Snowflake プライベート接続を使用しているかどうかに関係なく、手順 1 をスキップします。

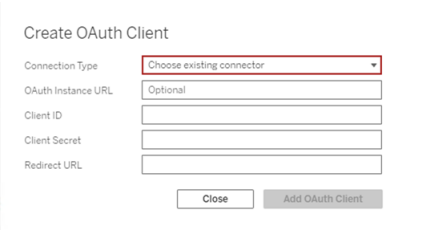
2. 管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
3. [OAuth Clients Registry (OAuth クライアント レジストリ)] で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックします。
4. 上記の**ステップ 1** の情報を含む、必要な情報を入力します。

a. **[接続タイプ]** で、カスタム OAuth クライアントを設定するコネクタを選択します。

b. 複数の OAuth クライアントを登録する場合は、**[OAuth インスタンス URL]** に情報を入力する必要があります。それ以外の場合は、任意で入力してください。

c. **クライアント ID**、**クライアント シークレット**、**リダイレクト URL** には、上記の**ステップ 1** で準備した情報を入力します。

d. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックして、登録プロセスを完了します。



5. (オプション) サポートされているすべてのコネクタに対して、ステップ 3 を繰り返します。
6. [設定] ページの下部または上部にある **[保存]** ボタンをクリックして、変更を保存します。

3: 保存した認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイト ユーザー) は、以前に保存した認証資格情報を削除してから、サイトのカスタム Oauth クライアントを使用できるように再度追加する必要があります。

1. **[マイ アカウント設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。

a. 上記の**ステップ 2** でカスタム OAuth クライアントを構成したコネクタの既存の、保存済み認証資格情報の横にある **[削除]** をクリックします。

b. コネクタ名の横にある **[追加]** をクリックし、プロンプトに従って 1) 上記の**ステップ 2** で

構成したカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

4: 保存した認証資格情報を更新するようにユーザーに通知する

上記の**ステップ 2** で構成したカスタム OAuth クライアントのコネクタの保存済み認証資格情報を更新するようにサイトユーザーに通知します。サイトのユーザーは、保存済み認証資格情報の更新で説明されている手順を使用して、保存された認証資格情報を更新できます。

Tableau Server を Salesforce Data Cloud に接続する

2025 年 10 月 14 日以降、「Data Cloud」は「Data 360」というブランド名に変わりました。この変更の途上では、アプリケーションやドキュメントに「Data Cloud」と表示されることがあります。名前は新しいですが、機能と内容は変更されていません。

Salesforce Data Cloud コネクタは、Tableau Desktop および Tableau Prep 向けに 2023.2 で、Tableau Cloud 向けに 2023 年 6 月に、Tableau Server 向けに 2023.3 でリリースされました。このコネクタは、Tableau を Data Cloud にシームレスに接続し、Tableau Desktop、Tableau Cloud、Tableau Server、Tableau Prep で利用できます。古い CDP コネクタと比べて、Salesforce Data Cloud コネクタは接続が簡単で、API 名の代わりにオブジェクトラベルが表示され、クエリが高速化されています。

注: Data Cloud は、以前は Customer Data Platform (CDP) と呼ばれていました。CDP コネクタは Tableau Server 2023.3 で非推奨になり、Tableau Server 2024.2 では使用できなくなりました。Tableau Server 2025.1 以降では完全に削除されます。CDP コネクタを使用する既存のアセットは、Salesforce Data Cloud コネクタを使用してもう一度作成する必要があります。

2023.1 以前の Tableau Server を使用している場合は、「Tableau Server (バージョン 2023.1 以前)」を参照してください。

Tableau Server (バージョン 2023.3 以降)

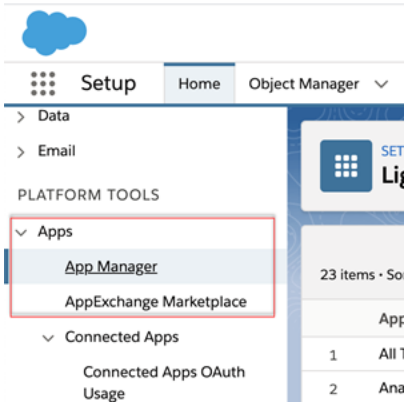
ステップ 1: Salesforce 接続済みアプリの作成

Salesforce Data Cloud コネクタを使用するには、Salesforce 接続済みアプリを作成する必要があります。

Salesforce Data Cloud コネクタと Salesforce コネクタの両方で同じ接続済みアプリを使用できます。そのため、すでに Salesforce コネクタを使用している場合は、ステップ 7 にリストされている 3 つの Customer Data Platform のスコープを既存の接続済みアプリに追加するだけで構いません。

1. Salesforce.com 開発者アカウントにサインインし、右上のユーザー名をクリックしてから **[設定]** を選択します。

2. 左のナビゲーション列の **[アプリ]** で、**[アプリマネージャー]** を選択します。



3. [接続済みアプリ] セクションで、**[新しい接続済みアプリ]** をクリックします。

4. **[基本情報]** でアプリに名前を付け、[API] フィールドをタブで移動して正しい形式で自動的に入力されるようにし、アプリの連絡先メールアドレスを入力します。
5. API の **[OAuth 設定を有効にする]** セクションで、**[OAuth 設定を有効にする]** を選択します。

6. 表示された新しい OAuth 設定で、**[コールバック URL]** に `https` プロトコルを使用してサーバーの完全修飾ドメイン名を入力し、URL に `auth/add_oauth_token` を追加します。例:
   `https://www.your_tableau_server.com/auth/add_oauth_token`
7. 次の項目を **[利用可能な OAuth スコープ]** から **[選択した OAuth スコープ]** に移動します。
   - **アイデンティティ URL サービス (Id、プロフィール、メール、住所、電話番号) にアクセス**
   - **API を介してユーザー データを管理 (api)**
   - **いつでもリクエストを実行 (refresh_token、オフライン アクセス)**
   - **Customer Data Platform のデータに対して ANSI SQL クエリを実行 (cdp_query_api)**
   - **Customer Data Platform のプロファイル データを管理 (cdp_profile_api)**
   - **Customer Data Platform の取り込み API データを管理 (cdp_ingest_api)**
8. **[保存]** をクリックします。

アプリを保存すると、Tableau Server の構成に使用する次の ID が Salesforce の API セクションに表示されます。

- **コンシューマー キー**
- **コンシューマー シークレット**
- **コールバック URL**

ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する

接続済みアプリが Salesforce で作成され、カスタマー キー、カスタマー シークレット、およびコールバック URL を取得したら、Salesforce データの接続と出力、および Einstein Discovery 用に Tableau Server を構成できます。

1. Tableau Server コンピューターのコマンドプロンプトで、次のコマンドを実行します。
   ```
   tsm configuration set -k oauth.salesforce.client_id -v <your_customer_key>
   tsm configuration set -k oauth.salesforce.client_secret -v
   ```

```
<your_customer_secret>
tsm configuration set -k oauth.salesforce.redirect_uri -v
<your_redirect_URL>
```

2. (オプション) 既定のログイン サーバーを変更するには、次のコマンドを入力します。
   ```
   tsm configuration set -k oauth.salesforce.server_base_url -v
   <URL>
   ```
3. 次のコマンドを入力して変更を適用します。
   ```
   tsm pending-changes apply
   ```
   保留中の変更にサーバーの再起動が必要な場合は、**pending-changes apply** コマンドを実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、「tsm pending-changes apply」を参照してください。

サイトのカスタム OAuth の構成

セットアップ手順については、OAuth 接続のトピックを参照してください。

Tableau Server (バージョン 2023.1 以前)

警告: Customer Data Platform は 2023 年 10 月に非推奨になり、Tableau Server 2024.2 では利用できなくなりました。Tableau Srerver 2025.1 以降では完全に削除されます。CDP コネクタを使用するアセットを使用している場合は、代わりに Salesforce Data Cloud コネクタにアップグレードして使用することを強くお勧めします。

このセクションでは、OAuth 認証を使用して Tableau Server から Salesforce Customer Data Platform に接続する方法について説明します。

Tableau Server で Salesforce Customer Data Platform のデータを使用するには、このセクションで説明されているステップを実行する必要があります。

ステップ 1: コネクタを設定する

1. Tableau Exchange コネクタサイトから最新の Salesforce Customer Data Platform コネクタ (Salesforce_CDP.taco ファイル) をダウンロードします。
2. .taco ファイルを Tableau コネクタフォルダーに移動します。

- Windows: `C:\Users[Windows User]\Documents\My Tableau Repository\Connectors`
- Linux: `/opt/tableau/connectors` または `/var/opt/tableau_server/data/tabsvc/vizqlserver/Connectors/`

3. Tableau Server を再起動します。

ステップ 2: 顧客データプラットフォーム JDBC ドライバーをインストールする

注: Tableau Server バージョン 2023.1 は、JDBC ドライバーのバージョン 18 以上とのみ互換性があります。

1. Salesforce CDP GitHub サイトから最新の JDBC ドライバー (Salesforce-CDP-jdbc-[バージョン].jar ファイル) をダウンロードします。https://github.com/forcedotcom/Salesforce-CDP-jdbc/releases
2. ダウンロードした Salesforce-CDP-jdbc-[バージョン].jar ファイルを次の場所に移動します。

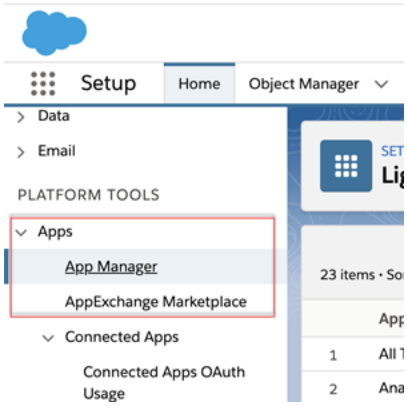
- Windows: `C:\Program Files\Tableau\Drivers`
- Linux: `/opt/tableau/tableau_driver/jdbc`

ステップ 3: Salesforce 接続済みアプリを作成する

API スコープを作成したら、以下の手順に従って、Tableau Server からの OAuth 委任を処理する Salesforce 接続済みアプリを作成します。

1. Salesforce Customer Data Platform アカウントに管理者としてサインインし、右上のユーザー名をクリックしてから **[設定]** を選択します。

2. 左側のペインの [アプリ] で、**[アプリマネージャ]** を選択します。



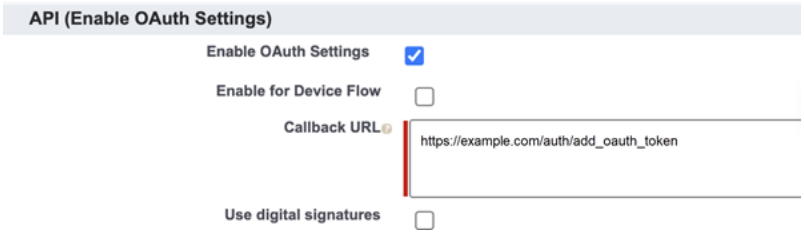
3. [接続済みアプリ] セクションで、**[新しい接続済みアプリ]** をクリックします。

4. **[基本情報]** で接続アプリに名前を付け (たとえば、Example.com)、API フィールドをタブで移動して正しい形式で自己入力されるようにし、アプリの連絡先メールアドレスを入力します。

5. **[API (Enable OAuth Settings) (API (OAuth 設定を有効にする))]** セクションで、**[OAuth 設定を有効にする]** をオンにします。

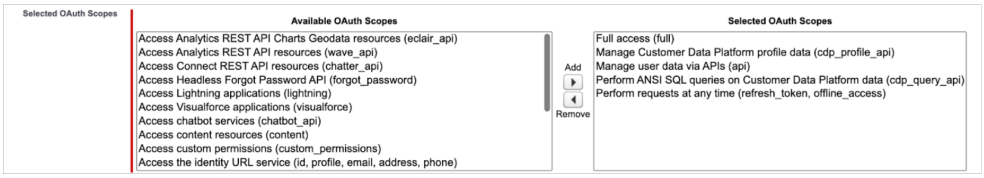
   a. 表示された新しい OAuth 設定で、`https` プロトコルを使用して **[コールバック URL]** に Tableau Server の完全修飾ドメイン名 (FQDN) を入力し、URL に `/auth/add_oauth_token` を追加します。

      例: `https://example.com/auth/add_oauth_token`



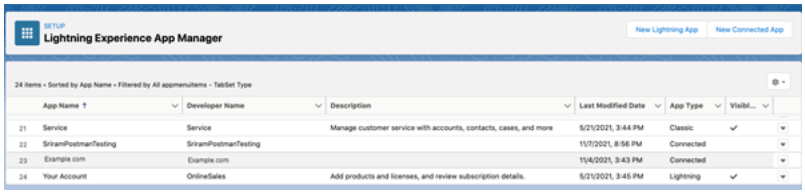
b. 次の項目を **[利用可能な OAuth スコープ]** から **[選択した OAuth スコープ]** に移動します。

- **API を介してユーザー データを管理する (api)**
- **ユーザーに代わっていつでも要求を実行する (refresh_token、offline_access)**
- **Customer Data Cloud のプロファイル データを管理する (cdp_profile_api)**
- **Customer Data Platform のデータに対して ANSI SQL クエリを実行する (cdp_query_api)**



6. 完了したら、**[保存]** をクリックします。

7. **[アプリマネージャ]** リストに移動してから接続済みアプリに移動し、ドロップダウン矢印をクリックして **[管理]** を選択します。



アプリを保存すると、Tableau Server の構成に使用する次の ID が **[API (OAuth 設定を有効にする)]** セクションに入力されます。

- **コンシューマー キー**
- **コンシューマー シークレット**

- **コールバック URL**

> **注:** コンシューマー キー、シークレット、コールバック URL は後で使用するため、保存しておいてください。

Customer Data Platform で OAuth を使用する

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

接続アプリが Salesforce で作成され、コンシューマー キー、コンシューマー シークレット、およびコールバック URL を取得したら、Customer Data Platform OAuth 接続用に Tableau Server を構成できます。開始するには、次の情報が必要です。

- **コンシューマー キー:** コンシューマー キー (Tableau ではクライアント ID とも呼びます) は、ステップ 4 の最後の手順で生成されます。この値を次の tsm コマンドの `[your_consumer_key]` で使用します。
- **コンシューマー シークレット:** コンシューマー シークレット (Tableau ではクライアント シークレットとも呼びます) は、ステップ 4 の最後の手順で生成されます。この値を次の tsm コマンドの `[your_consumer_secret]` で使用します。
- **コールバック URL:** Tableau ではリダイレクト URL とも呼ばれるコールバック URL は、Tableau Server の URL `https://example.com` およびその URL に追加される

"/auth/add_oauth_token" です。この値を次の tsm コマンドの [your_callback_url] で使用します。

- **構成 ID:** この値を次の tsm コマンドのパラメーター oauth.config.id で使用します。customer_360_audience

OAuth 設定での TSM コマンドの使用

次の tsm コマンドを実行して、OAuth を構成します。

```
tsm configuration set -k oauth.config.clients -v "
[{\"oauth.config.id\":\"customer_360_audience\",
\"oauth.config.client_id\":\"[your_consumer_key]\",
\"oauth.config.client_secret\":\"[your_consumer_secret]\",
\"oauth.config.redirect_uri\":\"[your_callback_url]\"}]" --force-
keys

tsm pending-changes apply
```

複数のコネクタの設定

複数のコネクタを設定する場合は、すべてを 1 つのコマンドにまとめる必要があります。例:

```
tsm configuration set -k oauth.config.clients -v "
[{\"oauth.config.id\":\"custom_360_audience\",
\"oauth.config.client_id\":\"[your_consumer_key]\",
\"oauth.config.client_secret\":\"[your_consumer_secret]\",
\"oauth.config.redirect_uri\":\"[your_callback_url]\"},
{\"oauth.config.id\":\"dremio\", \"oauth.config.client_id\":\"
[your_client_id]\", \"oauth.config.client_secret\":\"[your_client_
secret]\", \"oauth.config.redirect_uri\":\"[your_server_
url]/auth/add_oauth_token\"}, {\"oauth.config.id\":\"azure_sql_
dw\", \"oauth.config.client_id\":\"[your_client_id]\",
\"oauth.config.client_secret\":\"[your_client_secret]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}, {\"oauth.config.id\":\"azure_sqldb\",
\"oauth.config.client_id\":\"[your_client_id]\",
```

```
\"oauth.config.client_secret\":\"[your_client_secret]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}]" --force-keys

tsm pending-changes apply
```

ステップ 1: OAuth クライアント ID とクライアント シークレットの登録

以下の手順を実行して、カスタム OAuth クライアントをサイトに登録します。

1. サイト管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
2. **[OAuth Clients Registry (OAuth クライアント レジストリ)]** で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンを選択します。
3. **[接続タイプ]** で、[Customer Data Platform] を選択します。
4. [OAuth プロバイダー] で、**[カスタム IDP]** を選択します。
5. **[クライアント ID]** を入力します。
6. **[クライアント シークレット]** を入力します。
7. **[リダイレクト URL]** を入力します。
8. [Choose OAuth Config File (OAuth 構成ファイルの選択)] で、**[ファイルの選択]** ボタンを選択して、構成ファイルをアップロードします。
9. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンを選択して、登録プロセスを完了します。
10. [設定] ページの下部または上部にある [保存] ボタンを選択して、変更を保存します。

ステップ 2: 保存された認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイト ユーザー) は、以前に保存した認証資格情報を削除して、それをもう一度追加する必要があります。

1. [マイ アカウント設定] ページに移動します。
2. **[データ ソースの保存された認証資格情報]** で、既存の保存された認証資格情報の横にある **[削除]** を選択します。
3. 同じコネクタの横にある **[追加]** を選択します。
4. プロンプトに従って Customer Data Platform コネクタに接続します。
5. **[保存]** を選択します。

ステップ 3: 保存された認証資格情報を更新するようにユーザーに通知

Customer Data Platform コネクタ用に保存されている認証資格情報を更新するようにサイトユーザーに通知する必要があります。サイトのユーザーは、「データ接続用に保存された認証資格情報の管理」で説明されている手順を使用して、保存された認証資格情報を更新できます。

関連項目

- Salesforce ヘルプ: 顧客データプラットフォームで Tableau を設定する
- Salesforce ヘルプ: Tableau で顧客データプラットフォームを有効にする
- Salesforce ヘルプ: Tableau で顧客データプラットフォーム データを使用する

SAP HANA の外部 OAuth

Tableau Cloud 2024.3 以降では、OAuth 2.0/OIDC を使用して、外部の ID プロバイダーから HANA に ID を統合することができます。

ID プロバイダーによって、統合を設定するために必要な手順は異なります。これはおおまかな概要です。

> **注**: 現在 Tableau への OAuth 接続では、使い捨ての更新トークンはサポートされていません。ほとんどの場合、代わりにローリング更新トークンを使用するように ID プロバイダー (Okta など) を設定できます。詳細については、プロバイダーの OAuth ドキュメントを参照してください。

HANA で IdP を設定する

HANA で IdP を設定する方法については、SAP ヘルプ システム内の「JSON Web トークンを使用したシングル サインオン」を参照してください。

IdP を設定するさまざまな方法の例を次に示します。

- HANA Cockpit を使用した Okta IdP: 「Okta ID でユーザーが SAP HANA データにアクセスできるようにする」

- HANA Studio を使用した Azure IdP:「OAuth または SSO を使用した SAP HANA インスタンスの設定」

IdP を設定する

1. IDP に Tableau Desktop と Tableau Server または Tableau Cloud の OAuth クライアントを作成します。Desktop クライアントでは、PKCE を有効にし、http://localhost リダイレクトを使用する必要があります。

2. Tableau OAuth 構成ファイルを作成します。作成方法の詳細については、GitHub の「OAuth の設定と使用方法」(英語)、およびこちらの例を参照してください。他の IDP の例も歓迎します。

   a. Tableau OAuth 構成 ID には必ず「custom_」というプレフィックスを付けてください。

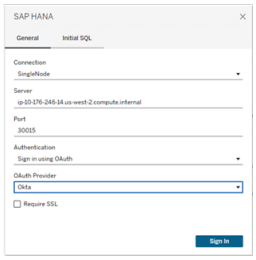
   > **注:** OAuth 設定 ID の最大文字数は 36 文字です。これより長い ID ではエラーが発生しない可能性がありますが、機能しません。

   b. IDP が動的 localhost ポートをサポートしている場合、OAUTH_CAP_FIXED_PORT_IN_CALLBACK_URL を無効にします。IDP がサポートしていない場合は、構成ファイルおよび IDP の許可リストにいくつかの localhost コールバック URL を追加してください。

3. Desktop ホスト (Tableau Desktop、Tableau Prep Builder、Tableau Bridge) 上、および OAuth を使用する各 Tableau Server と Tableau Cloud theサイト上の各アプリケーションに関連付けられた OAuthConfigs フォルダーに、新しい Tableau OAuth 構成ファイルをインストールします。詳細については、「Desktop 上のカスタム OAuth 構成」および「サイトレベルの OAuth クライアント」を参照してください。

HANA に接続する

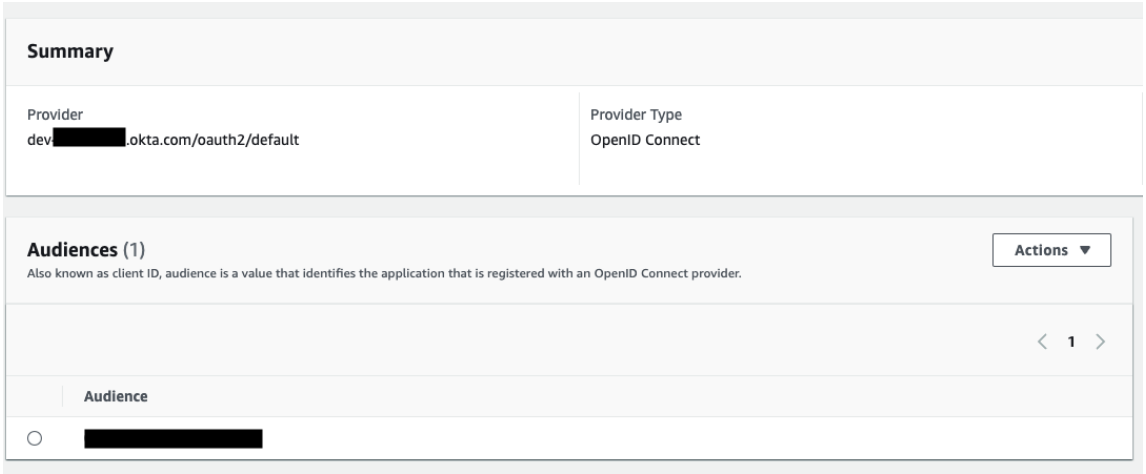
ユーザーは、**[OAuth を使用してサインイン]** を選択し、以前にインストールされた **OAuth プロバイダー**を選択する必要があります。



Okta

Okta を使用する場合は、「組織認可サーバー」ではなく「カスタム認可サーバー」を使用することをお勧めします。カスタム認可サーバーはより柔軟です。既定で作成されたカスタム認可サーバーがあり、これは「既定」と呼ばれます。認可 URL は次のようになります。

```
https://${yourOktaDomain}/oauth2/{authServerName}/v1/authorize
```



Snowflake の外部 OAuth

Tableau 2024.3 以降では、OAuth 2.0/OIDC を使用して、外部の ID プロバイダーから Snowflake に ID を統合することができます。

ID プロバイダーによって、統合を設定するために必要な手順は異なります。これは、設定の概要を上位レベルで紹介することを目的にしたものであり、設定に必要な詳細情報についてはご利用の ID プロバイダーのドキュメントを参照してください。ここでは、読者が OAuth の設定に精通していて、外部 ID プロバイダーを使用した認証の設定に必要な技術的詳細を理解していることを前提としています。

Snowflake で IDP を設定する

IDP の設定の詳細については、Snowflake のヘルプシステムで「外部 OAuth の概要」を参照してください。

Tableau で IDP を設定する

1. Tableau Desktop 用の IDP と、Tableau Cloud または Tableau Server に OAuth クライアントを作成します。Desktop クライアントは PKCE を有効にし、`http://localhost` リダイレクトを使用します。
2. Tableau OAuth 構成ファイルを作成します。作成方法の詳細については、「GitHub」の「OAuth の設定と使用方法」(英語)、およびこちらの例を参照してください。他の IDP のその他の例も歓迎します。
   A. Tableau OAuth 構成 ID には必ず「custom_」というプレフィックスを付けてください。

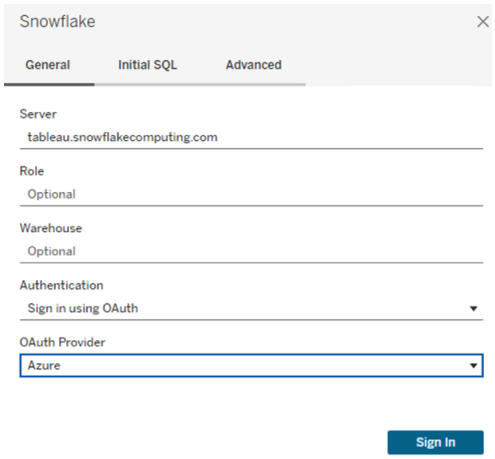
      > **注:** Oauth 設定 ID の最大文字数は 36 文字です。これより長い ID ではエラーが発生しない可能性がありますが、機能しません。

   B. IDP が動的 localhost ポートをサポートしている場合、`OAUTH_CAP_FIXED_PORT_IN_CALLBACK_URL` を無効にします。IDP がサポートしていない場合は、構成ファイルおよび IDP の許可リストにいくつかの localhost コールバック URL を追加してください。

3. Desktop ホスト(Tableau Desktop、Tableau Prep Builder、Tableau Bridge) 上、および OAuth を使用する各 Tableau Server とTableau Cloud サイト上の各 アプリケーションに関連付けられた OAuthConfigs フォルダーに、新しい Tableau OAuth 構成ファイルをインストールします。詳細については、「Desktop 上のカスタム OAuth 構成」および「サイトレベルの OAuth クライアント」を参照してください。

Snowflake に接続する

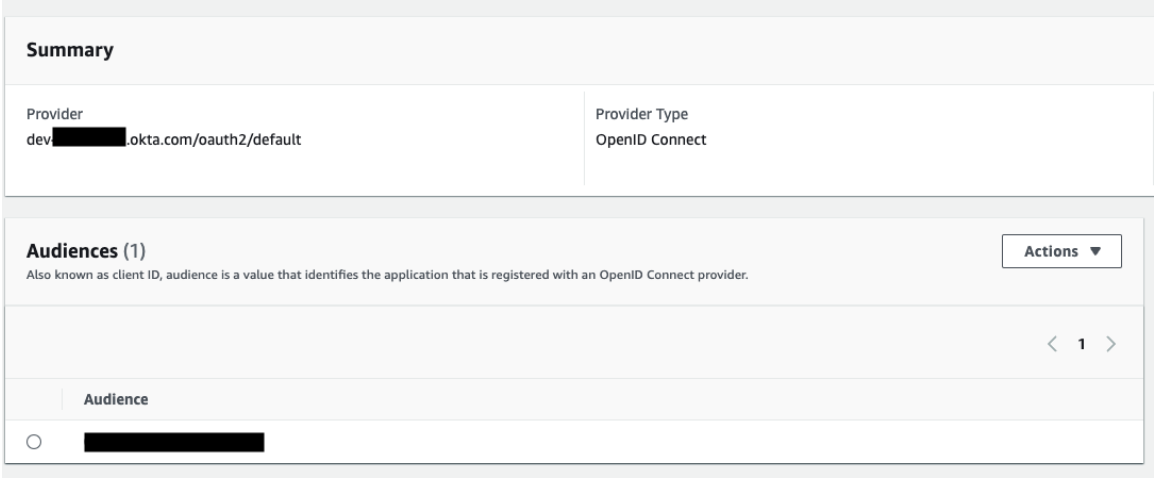
接続するときは、OAuth を選択し、事前にインストールした OAuth 構成を選択する必要があります。



Okta

Okta を使用する場合は、「組織認可サーバー」ではなく「カスタム認可サーバー」を使用することをお勧めします。カスタム認可サーバーはより柔軟です。デフォルトで作成されたカスタム認可サーバーがあり、これは「デフォルト」と呼ばれます。認可 URL は次のようになります。

```
https://${yourOktaDomain}/oauth2/{authServerName}/v1/authorize
```



Hyper Query Processing (ベータ版)

> **注:** Hyper Query Processing はベータ リリースであり、一般 リリース前 に名前 が変更 される可能性 があります。

現在、Customer Data Platform (CDP) コネクタは、Hyper Query Processing エンジンを使用しています。このエンジンは、高速なデータ クエリ処理によるインタラクティブな分析をサポートし、データへの接続エクスペリエンスを簡素化することによりデータ探索をスピードアップします。また、ライブ接続とTableau データ抽出を使用してTableau から顧客データプラットフォーム データをクエリする際のクエリ時間を短縮します。

ベータ機能: Hyper Query Processing のベータ機能は、追加機能を無料で提供します。このサービスは、ご自身の裁量で試すことができます。ベータ版サービスの使用には、「Agreements and Terms (契約条件)」に記載されている該当するベータ版サービス条件が適用されます。

Tableau Server で Hyper Query Processing を有効にする

Hyper Query Processing を有効にするには、次のステップを実行します。

1.Salesforce CDP に接続します。

2.画面の指示に従って hyper jdbc ドライバーをインストールします。

-Salesforce-cdp-hyper-beta-1.15.0.jar または、ここから.jar ファイルをダウンロードします。

3.ご利用のシステムに応じて、ダウンロードしたファイルを次のいずれかのディレクトリパスに移動します。

Windows - `C:\Program Files\Tableau\Drivers`

macOS - `/Users/[user]/Library/Tableau/Drivers`

4.ドライバーを正しいパスに移動したら、もう一度接続します。これで、Tableau Server で Customer Data Platform コネクタを使用できます。

注: Hyper Query Processing のサポートはベータリリースであり、Tableau Server で利用できます。Tableau Desktop とTableau Online は 2022 年 12 月からサポートされます。これらの日付は変更される場合があります。

関連項目

- Tableau ヘルプ: Salesforce CDP への Tableau Server の接続
- Salesforce ヘルプ: 顧客データプラットフォームで Tableau を設定する
- Salesforce ヘルプ: Tableau で顧客データプラットフォームを有効にする
- Salesforce ヘルプ: Tableau で顧客データプラットフォーム データを使用する

Amazon Redshift IAM OAuth の設定

Tableau 2023.3 以降より、OAuth 2.0/OIDC を使用して、外部の ID プロバイダーからAmazon Redshift に ID をフェデレートすることができます。

これらの手順は、古い AWS IAM サービス用です。IAM IDC 統合については、「Amazon Redshift IAM ID センター OAuth の設定」を参照してください。

IDプロバイダーによって、統合を設定するために必要な手順は異なります。これはおおまかな概要です。Tableauは、AWSまたはIDPの構成方法について詳細な手順を提供できませんが、一般的なアプローチは次のとおりです。

注: 現時点では、TableauへのOAuth接続では、1回のみ使用可能な更新トークン(ローリング更新トークンまたは更新トークンのローテーションと呼ばれることもあります)はサポートされていません。これらのトークンは今後のリリースでサポートされる予定です。

ステップ1: IDPを構成する

1. IDPにTableau DesktopとTableau ServerまたはTableau CloudのOAuthクライアントを作成します。Desktopクライアントでは、`PKCE` を有効にし、`http://localhost` リダイレクトを使用する必要があります。
2. ロールの承認に使用するカスタム クレームを追加します。特にオリジナルのIAMを使用している場合は、`DbUser` と`DbGroups` のクレームを追加することをお勧めします。これらは後でIAMポリシーで使用できます。
3. Tableau OAuth構成ファイルを作成します。GitHubのドキュメントと、ここでの例を参照してください。他のIDPの例も歓迎します。
    a. Tableau OAuth構成IDには必ず「`custom_`」というプレフィックスを付けてください。

    注: OAuth設定IDの最大文字数は36文字です。これより長いIDではエラーが発生しない可能性がありますが、機能しません。

    b. IDPが動的localhostポートをサポートしている場合、`OAUTH_CAP_FIXED_PORT_IN_CALLBACK_URL` を無効にします。IDPがサポートしていない場合は、構成ファイルおよびIDPの許可リストにいくつかのlocalhostコールバックURLを追加してください。

4. デスクトップ ホスト(Tableau Desktop、Tableau Prep Builder、Tableau Bridge) 上、および OAuth を使用する各 Tableau Server とTableau Cloud サイト上の各 アプリケーションに関連付けられた `OAuthConfigs` フォルダーに新しい Tableau OAuth 構成ファイルをインストールします。

AWS で IDP を構成する

1.AWS 上で IDP モデルを作成します。Amazon ドキュメントの「ウェブ ID フェデレーションについて」と「OpenID Connect (OIDC) ID プロバイダーの作成」を参照してください。

2.IDP 専用のロールとポリシーを作成します。AWS ドキュメントの「ウェブ ID または OpenID Connect フェデレーション用のロールの作成 (コンソール)」参照してください。

Redshift ユーザーのロールを構成する

Redshift に必要なポリシーをアタッチします。トークンのカスタム クレームを使用して、ロールに権限を付与できます。AWS ドキュメントには SAML を使用した例がいくつかあります。これらの例は OAuth に簡単に適応させることができます。OAuth の場合、クレームは単に「`DbUser`」や「`DbGroups`」などです。

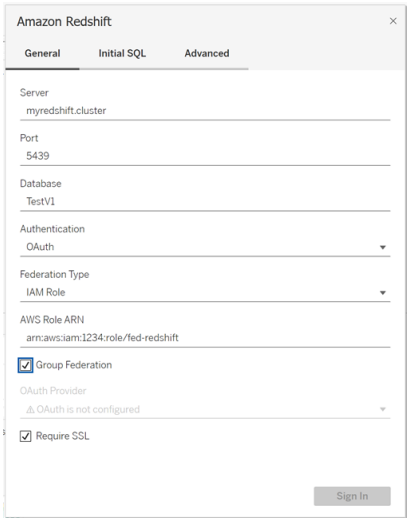
AWS ドキュメントのポリシー例は次のとおりです。

```
{
"Version": "2012-10-17",
"Statement": [
    {
        "Effect": "Allow",
        "Action": "redshift:GetClusterCredentials",
        "Resource": [
            "arn:aws:redshift:us-west-
1:123456789012:dbname:cluster-identifier/dev",
            "arn:aws:redshift:us-west-
1:123456789012:dbuser:cluster-identifier/${redshift:DbUser}",
            "arn:aws:redshift:us-west-
1:123456789012:cluster:cluster-identifier"
        ],
```

```
            "Condition": {
                "StringEquals": {
                    "aws:userid":
"AROAJ2UCCR6DPCEXAMPLE:${redshift:DbUser}@example.com"
                }
            }
        },
        {
            "Effect": "Allow"
            "Action": "redshift:CreateClusterUser",
            "Resource": "arn:aws:redshift:us-west-
1:12345:dbuser:cluster-identifier/${redshift:DbUser}"
        },
        {
            "Effect": "Allow",
            "Action": "redshift:JoinGroup",
            "Resource": "arn:aws:redshift:us-west-
1:12345:dbgroup:cluster-identifier/my_dbgroup"
        },
        {
            "Effect": "Allow",
            "Action": [
                "redshift:DescribeClusters",
                "iam:ListRoles"
            ],
            "Resource": "*"
        }
    ]
    }
```

Redshift に接続する

ユーザーは、引き受ける**ロール ARN** を指定し、前にインストールした OAuth 構成を選択する必要があります。



適切に構成されていると、ユーザーは IDP にリダイレクトされ、Tableau のトークンが認証および認可されます。Tableau は openid トークンとリフレッシュトークンを受け取ります。AWS は、IDP からのトークンと署名を検証し、トークンからクレームを抽出して、IAM ロールへのクレームのマッピングを検索し、Tableau がユーザーに代わってロールを引き受けることを許可または拒否します。(例: AssumeRoleWithWebIdentity)。

トークン

既定では、Redshift OAuth IAM は ID トークンをドライバーに渡します。Tableau Bridge を使用しているお客様を含むオンプレミスのお客様の場合は、代わりに TDC ファイルを使用してアクセストークンを渡すこともできます。

```
<connection-customization class='redshift' enabled='true'
version='10.0'>
       <vendor name='redshift' />
       <driver name='redshift' />
       <customizations>
              <customization name='CAP_OAUTH_FEDERATE_ACCESS_TOKEN'
value='yes'/>
       </customizations>
</connection-customization>
```

.tdc ファイルの設定とインストールの詳細については、「接続をカスタマイズして調整する」および「Tableau Server で .tdc ファイルを使用する」を参照してください。

フェデレーションのグループ化

IAM ロールで OAuth 認証を使用する場合、フェデレーションのグループ化を使用するかどうかを選択できます。これにより、コネクタが認証 API と対話して Redshift と相互作用する方法が変更されます。

- 接続時に [Group Federation (フェデレーションのグループ化)] のチェックボックス (2025.1 以降から利用可能) がオンになっている場合、Redshift ドライバーは `getClusterCredentialsWithIAM` API を使用して、プロビジョニングされたクラスター内の認証資格情報を取得します。
- [Group Federation (フェデレーションのグループ化)] のチェックボックスがオフになっている場合は、代わりに `getClusterCredentials` API が使用されます。

これら 2 つの API は、プロパティが若干異なる IAM トークンを返します。詳細については、上記のリンク先の AWS API ドキュメント、および「ODBC ドライバーオプション」を参照してください。

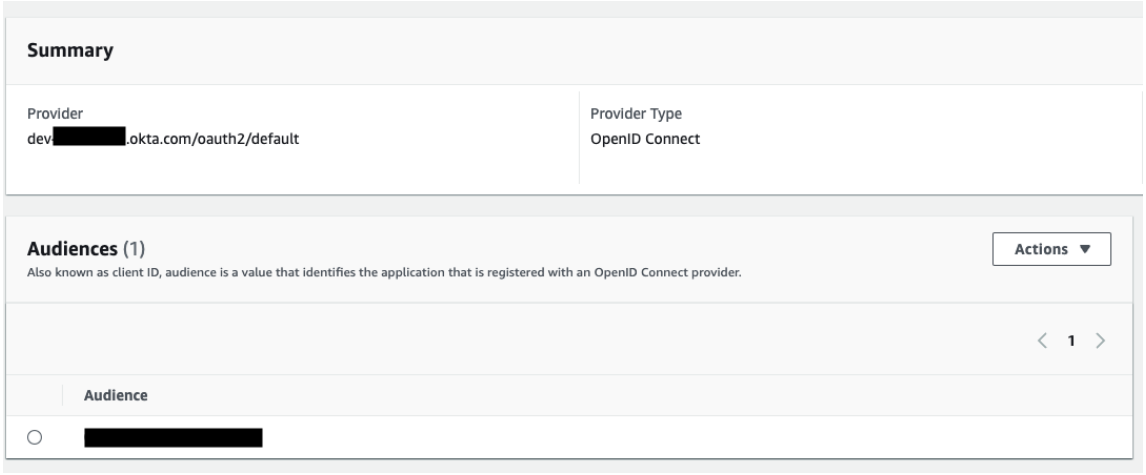
使用上の注意

- バージョン 2024.3 より前の Tableau Desktop および Tableau Prep Builder では、'group_federation=1' を使用して、高度なドライバー構成フィールドでこれを設定できます。また、TDC ファイルを使用してカスタム接続プロパティを定義することによって構成することもできます。TDC の使用に関する詳細については、「接続のカスタマイズと微調整」を参照してください。
- バージョン 2024.3 以降では、オンプレミス (Tableau Desktop、Tableau Prep Builder、Tableau Server) の接続ダイアログに group_federation フィールドが追加されました。このフィールドは、2025.1 以降の Tableau Cloud で利用可能です (2025 年 2 月から利用可能)。
- Tableau Server でフェデレーションのグループ化を使用するには、group_federation を ODBC extras の許可リストに追加する必要があります。詳細については、「ネイティブ コネクタの接続文字列をカスタマイズする」を参照してください。

Okta

Okta を使用する場合は、「組織認可サーバー」ではなく「カスタム認可サーバー」を使用することをお勧めします。カスタム認可サーバーはより柔軟です。既定で作成されたカスタム認可サーバーがあり、これは「既定」と呼ばれます。認可 URL は次のようになります。

```
https://${yourOktaDomain}/oauth2/{authServerName}/v1/authorize
```



ドライバーを更新する

オリジナルの IAM サービスを使用する Redshift OAuth の場合は、次のいずれかを使用できます。

- バージョン 1.59 以降の Redshift ODBC v1 ドライバー。このドライバーは、https://docs.aws.amazon.com/redshift/latest/mgmt/configure-odbc-connection.html からダウンロードできます。
- バージョン 2.0.1.0 以降の Redshift ODBC v2 ドライバー。このドライバーは、https://github.com/aws/amazon-redshift-odbc-driver/tags からダウンロードできます。OSX 用の v2 ドライバーはないことに注意してください。

トラブルシューティング

エラーを診断する最善の方法は、問題が発生している状況からTableau を削除することです。代わりに、ドライバー マネージャーまたは同様のツールを使用してテストすることができます。これはあくまでトラブルシューティングを目的とした作業です。この機能の通常の使用時には DSN や「その他の ODBC」コネクタを使用しないでください。有効なテストを行えるようにするためには、クラスター情報、データベース、トークン、および名前空間を除くパラメーターが以下に示すものと同じである必要があります。

> **注:** テストするときは DSN の接続文字列で DbUser または DbGroups を指定しないでください。Tableau はこれらにアクセスできず、接続文字列に含めることができません。トークン要求に含めることはできます。

最初の接続時にドライバーから無効/期限切れのトークンに関するエラー メッセージが表示された場合 (エラー メッセージには [28000] や [08001] などの SQLState エラー コードが含まれます)、Tableau は OAuth フローを正常に完了していますが、ドライバーに関して失敗しています。これは、AWS 側または IDP 側のいずれかに設定ミスがあることを意味します。ドライバーからパーミッションまたは認可のエラーが返される場合もありますが、これも Tableau の制御の対象外です。

テストを開始する前に、まず、ドライバーに送信するアクセス トークン (IAM IDC の既定値) または更新トークン (カスタマイズされている場合) を取得する必要があります。

Okta を使用した例を次に示します。ほぼすべての IDP は、同様の方法で Okta を使用します。このフローを使用するには、リソース所有者のパスワード付与タイプを有効にする必要があることに注意してください。IDP URL、クライアント シークレット、クライアント ID、ユーザー名、パスワードを置き換えます。

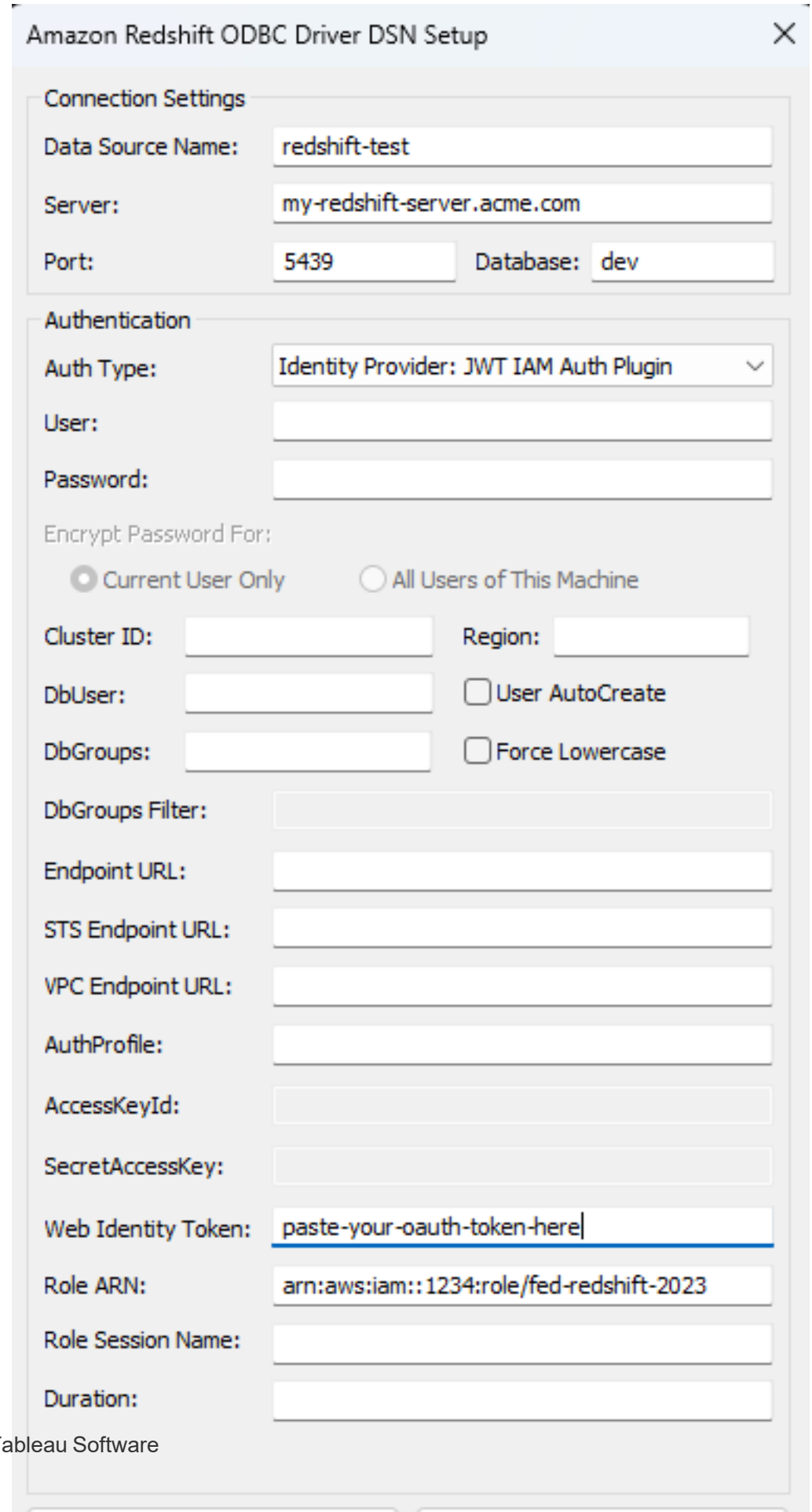
```
curl -X POST "https://OKTA_URL/v1/token" \
-H 'accept: application/json' \
-H "Authorization: Basic $(echo -n 'CLIENTID:CLIENTSECRET' |
base64)" \
-H "Content-Type: application/x-www-form-urlencoded" \
```

```
-d "grant_
type=password&username=USER&password=PASSWORD&scope=openid"
```

トークンを取得したら、DSN を使用してテストできます。Windows では、ODBC ドライバー マネージャーを使用できます。Mac では、iODBC ドライバー マネージャー UI を使用できます。Linux では、Tableau Server の customer-bin フォルダーに含まれている isql コマンド ライン ツールを使用できます。

固定の AWS プロファイルを使用するプラグインは、その接続のすべてのユーザーに同じプロファイルを使用したい場合を除き、機能しません。他の OAuth プラグインでは、システム ブラウザーに直接アクセスする必要がありますが、サーバー側では機能しません。

以下は、Windows での ODBC ドライバー マネージャーの使用例です。


Amazon Redshift ODBC Driver DSN Setup
Connection Settings
Data Source Name: redshift-test
Server: my-redshift-server.acme.com
Port: 5439
Database: dev
Authentication
Auth Type: Identity Provider: JWT IAM Auth Plugin
User:
Password:
Encrypt Password For:
Current User Only
All Users of This Machine
Cluster ID:
Region:
DbUser:
User AutoCreate
DbGroups:
Force Lowercase
DbGroups Filter:
Endpoint URL:
STS Endpoint URL:
VPC Endpoint URL:
AuthProfile:
AccessKeyId:
SecretAccessKey:
Web Identity Token: paste-your-oauth-token-here
Role ARN: arn:aws:iam::1234:role/fed-redshift-2023
Role Session Name:
Duration:

Amazon Redshift IAM ID センター OAuth の設定

Tableau 2023.3 以降より、OAuth 2.0/OIDC を使用して、外部の ID プロバイダーからAmazon Redshift に ID をフェデレートすることができます。

以下の手順は、新しい AWS IAM IDC サービス用です。オリジナルの IAM 統合については、「Amazon Redshift IAM OAuth の設定」を参照してください。

ID プロバイダーによって、統合を設定するために必要な手順は異なります。これはおおまかな概要です。Tableau は、AWS または IDP の構成方法について詳細な手順を提供できませんが、一般的なアプローチは次のとおりです。

Redshift で認証を実装する詳細な例については、「AWS IAM アイデンティティセンターを使用して Tableau とOkta を Amazon Redshift に統合する」(英語) および「AWS IAM アイデンティティセンターを使用して、Tableau とMicrosoft Entra ID を Amazon Redshift に統合する」(英語) を参照してください。

注: 現時点では、Tableau への OAuth 接続では、1 回のみ使用可能な更新トークン (ローリング更新トークンまたは更新トークンのローテーションと呼ばれることもあります) はサポートされていません。これらのトークンは今後のリリースでサポートされる予定です。

ステップ 1: IDP を構成する

1. IDP に Tableau Desktop とTableau Server または Tableau Cloud の OAuth クライアントを作成します。Desktop クライアントでは、`PKCE` を有効にし、`http://localhost` リダイレクトを使用する必要があります。
2. ロールを承認するために必要なカスタム クレームを追加します。
3. Tableau OAuth 構成ファイルを作成します。GitHub のドキュメントと例を参照してください。他の IDP の例も歓迎します。

a. Tableau OAuth 構成 ID には必ず「`custom_`」というプレフィックスを付けてください。

> **注:** OAuth 設定 ID の最大文字数は 36 文字です。これより長い ID ではエラーが発生しない可能性がありますが、機能しません。

b. IDP が動的 localhost ポートをサポートしている場合、`OAUTH_CAP_FIXED_PORT_IN_CALLBACK_URL` を無効にします。そうでない場合は、構成ファイルおよび IDP の許可リストにいくつかの localhost コールバック URL を追加してください。

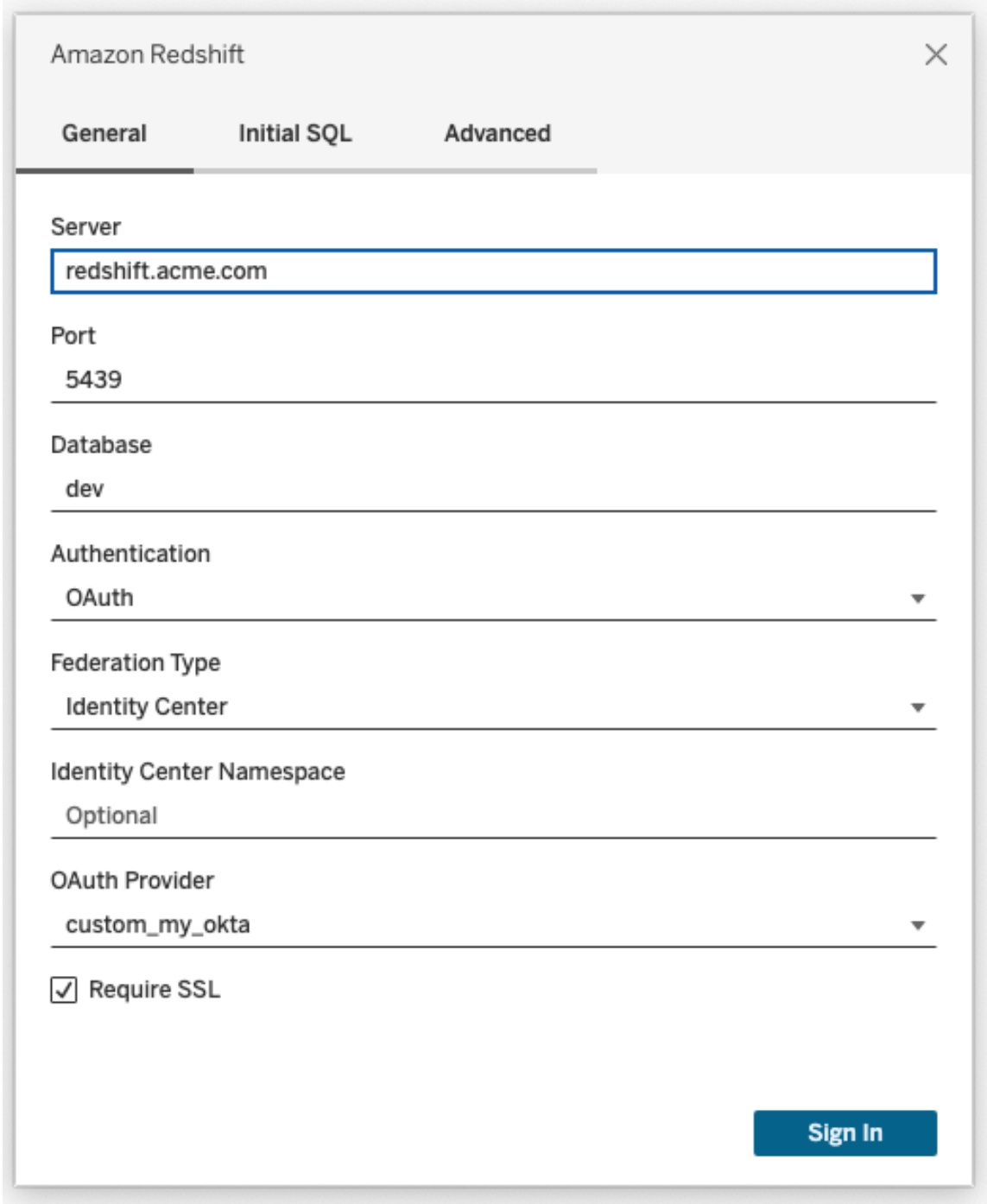
4. デスクトップホスト (Tableau Desktop、Tableau Prep Builder、Tableau Bridge) 上および OAuth を使用する各 Tableau Server と Tableau Cloud サイト上の各アプリケーションに関連付けられた `OAuthConfigs` フォルダーに新しい Tableau OAuth 構成ファイルをインストールします。

ステップ 2: AWS で IDP とロールを構成する

構成する方法については、AWS のドキュメントを参照してください。

ステップ 3: Redshift に接続する

1. Redshift に接続します。
2. **[認証]** には [OAuth] を選択します。
3. **[フェデレーション タイプ]** には [ID センター] を選択します。
4. (オプション) 必要に応じて、**[ID センターの名前空間]** を指定します。



正しく構成されていると、ユーザーは IDP にリダイレクトされ、Tableau のトークンが認証および認可されます。Tableau はアクセス トークンと更新トークンを受け取ります。認証のためにアクセス トークンをドライバーに送信します。

トークン

既定では、Redshift OAuth to IAM IDC はアクセス トークンをドライバーに渡します。Tableau Bridge を使用しているお客様を含むオンプレミスのお客様の場合は、代わりに TDC ファイルを使用して ID トークンを渡すこともできます。

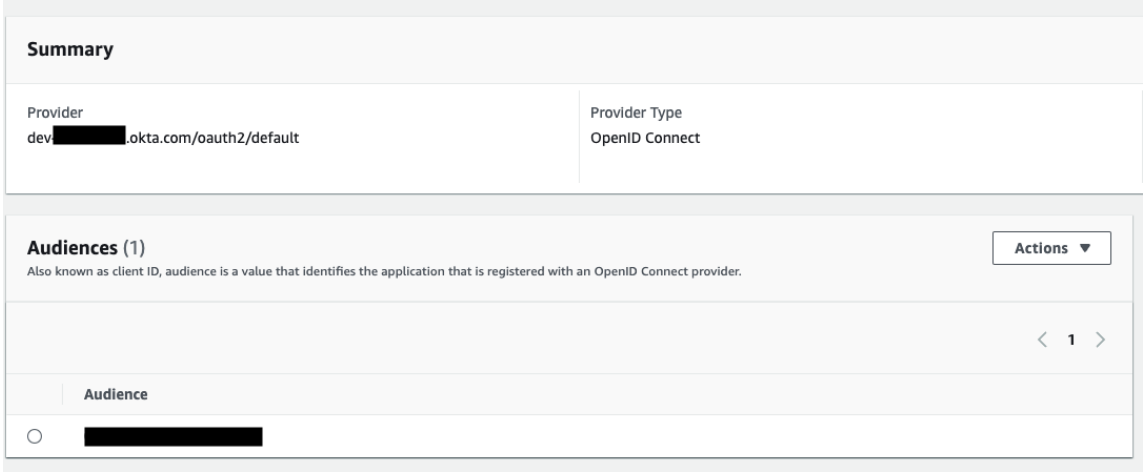
```
<connection-customization class='redshift' enabled='true'
version='10.0'>
       <vendor name='redshift' />
       <driver name='redshift' />
       <customizations>
              <customization name='CAP_OAUTH_FEDERATE_ID_TOKEN' value='yes'/>
       </customizations>
</connection-customization>
```

.tdc ファイルの設定とインストールの詳細については、「接続をカスタマイズして調整する」および「Tableau Server で .tdc ファイルを使用する」を参照してください。

Okta

Okta を使用する場合は、「組織認可サーバー」ではなく、「カスタム認可サーバー」を使用することをお勧めします。カスタム認可サーバーはより柔軟です。カスタム認可サーバーは既定で作成され、「既定」と呼ばれます。認可 URL は次のようになります。

```
https://${yourOktaDomain}/oauth2/{authServerName}/v1/authorize
```



ドライバーを更新する

IAM IDC サービスを使用する Redshift OAuth の場合、ODBC ドライバーのバージョン 2.x 以上を使用する必要があります。Redshift ODBC ドライバーの最新バージョンは、https://github.com/aws/amazon-redshift-odbc-driver/tags でダウンロードできます。OSX 用の v2 ドライバーはまだ提供されていないことに注意してください。

Redshift IAM IDC OAuth のトラブルシューティング

エラーを診断する最善の方法は、問題が発生している状況から Tableau を削除することです。代わりに、ドライバー マネージャーまたは同様のツールを使用してテストすることができます。これはあくまでトラブルシューティングを目的とした作業です。この機能の通常の使用時には DSN や「その他の ODBC」コネクタを使用しないでください。有効なテストを行えるようにするためには、クラスター情報、データベース、トークン、および名前空間を除くパラメーターが以下に示すものと同じである必要があります。

最初の接続時にドライバーから無効/期限切れのトークンに関するエラー メッセージが表示された場合 (エラー メッセージには [28000] や [08001] などの SQLState エラー コードが含まれます)、Tableau は OAuth フローを正常に完了していますが、ドライバーに関して失敗しています。これ

は、AWS 側または IDP 側のいずれかに設定ミスがあることを意味します。ドライバーからパーミッションまたは認可のエラーが返される場合もありますが、これも Tableau の制御の対象外です。

テストを開始する前に、まず、ドライバーに送信するアクセストークン (IAM IDC の既定値) または更新トークン (カスタマイズされている場合) を取得する必要があります。

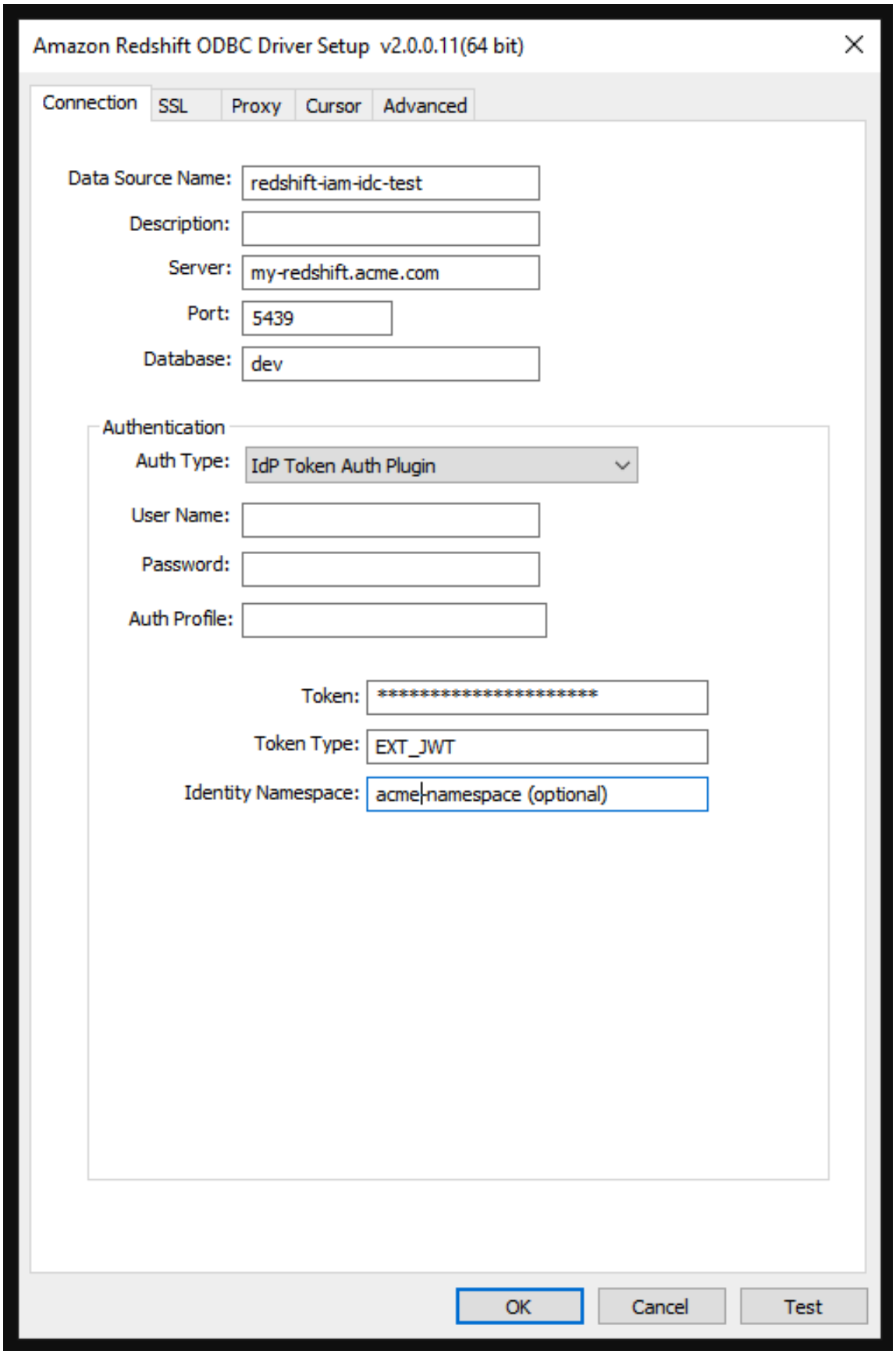
Okta を使用した例を次に示します。ほぼすべての IDP は、同様の方法で Okta を使用します。このフローを使用するには、リソース所有者のパスワード付与タイプを有効にする必要があることに注意してください。IDP URL、クライアントシークレット、クライアント ID、ユーザー名、パスワードを置き換えます。

```
curl -X POST "https://OKTA_URL/v1/token" \
-H 'accept: application/json' \
-H "Authorization: Basic $(echo -n 'CLIENTID:CLIENTSECRET' |
base64)" \
-H "Content-Type: application/x-www-form-urlencoded" \
-d "grant_
type=password&username=USER&password=PASSWORD&scope=openid"
```

トークンを取得したら、DSN を使用してテストできます。Windows では、ODBC ドライバー マネージャーを使用できます。Linux では、Tableau Server の customer-bin フォルダーに含まれている isql コマンド ライン ツールを使用できます。

サーバー環境では動作しない可能性があるため、他のプラグインを使用してテストしないことをお勧めします。固定の AWS プロファイルを使用するか、ブラウザーに直接アクセスする必要があります。

以下は、Windows での ODBC ドライバー マネージャーの使用例です。


Amazon Redshift ODBC Driver Setup v2.0.0.11(64 bit)
Connection
SSL
Proxy
Cursor
Advanced
Data Source Name:
redshift-iam-idc-test
Description:
Server:
my-redshift.acme.com
Port:
5439
Database:
dev
Authentication
Auth Type:
IdP Token Auth Plugin
User Name:
Password:
Auth Profile:
Token:
****
Token Type:
EXT_JWT
Identity Namespace:
acme-namespace (optional)
OK
Cancel
Test

Dremio 向け OAuth の設定

このトピックでは、OAuth 認証用に Dremio データソースを設定する方法を説明します。各 Tableau Server インスタンスでこれらのステップを実行します。

Dremio 向け OAuth の設定は、次のタスクで構成されています。

1. OAuth クライアントを Dremio に登録します。
2. ステップ 1 で取得した情報を使用して、Dremio OAuth 用に Tableau Server を構成します。
3. (オプション) サイト固有の OAuth を構成します。

ステップ 1: OAuth クライアントを Dremio に登録する

Dremio ドキュメントの「Identity Providers」トピックを使用して、Dremio でサポートされる IdP を構成し、Dremio OAuth 用に Tableau Server を構成するために必要な OAuth クライアント ID とシークレット構成パラメーターを取得します。

ステップ 2: Dremio OAuth 用に Tableau Server を構成する

Dremio OAuth 用に Tableau Server を構成するには、以下に記載されているパラメーターを後続の tsm コマンドで使用します。

- **Dremio クライアント ID:** このクライアント ID はステップ 1 の登録プロセスで生成されます。この値を tsm コマンドの `[your_client_id]` にコピーします。
- **Dremio クライアント シークレット:** このクライアント シークレットは、ステップ 1 の手順で生成されます。この値を tsm コマンドの `[your_client_secret]` にコピーします。
- **Tableau Server URL:** これは Tableau Server URL です。例: `https://myco.com` この値を tsm コマンドの `[your_server_url]` にコピーします。
- **構成 ID:** これは、tsm コマンド: `dremio` で使用する `oauth.config.id` パラメーターの値です。

次の tsm コマンドを実行して、Dremio 用の OAuth を構成します。

```
tsm configuration set -k oauth.config.clients -v "
[{\"oauth.config.id\":\"dremio\", \"oauth.config.client_id\":\"
[your_client_id]\", \"oauth.config.client_secret\":\"[your_client_
secret]\", \"oauth.config.redirect_uri\":\"[your_server_
url]/auth/add_oauth_token\"}]" --force-keys

tsm pending-changes apply
```

複数のコネクタの設定

複数のコネクタを設定する場合は、すべてを1つのコマンドにまとめる必要があります。例:

```
tsm configuration set -k oauth.config.clients -v "
[{\"oauth.config.id\":\"dremio\", \"oauth.config.client_id\":\"
[your_client_id]\", \"oauth.config.client_secret\":\"[your_client_
secret]\", \"oauth.config.redirect_uri\":\"[your_server_
url]/auth/add_oauth_token\"}, {\"oauth.config.id\":\"customer_360_
audience\", \"oauth.config.client_id\":\"[your_client_id]\",
\"oauth.config.client_secret\":\"[your_client_secret]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}, {\"oauth.config.id\":\"azure_sql_dw\",
\"oauth.config.client_id\":\"[your_client_id]\",
\"oauth.config.client_secret\":\"[your_client_secret]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}, {\"oauth.config.id\":\"azure_sqldb\",
\"oauth.config.client_id\":\"[your_client_id]\",
\"oauth.config.client_secret\":\"[your_client_secret]\",
\"oauth.config.redirect_uri\":\"[your_server_url]/auth/add_oauth_
token\"}]" --force-keys

tsm pending-changes apply
```

サイトのカスタム OAuth の構成

サイトに対してカスタム Dremio OAuth クライアントを構成できます。

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

カスタム OAuth クライアントが構成されている場合、サイトレベルの構成はサーバー側の構成よりも優先され、作成されたすべての新しい OAuth 認証資格情報では既定でサイトレベルの OAuth クライアントが使用されます。構成を有効にするために、Tableau Server を再起動する必要はありません。

重要: カスタム OAuth クライアントを構成する前に確立された既存の OAuth 認証資格情報は一時的に使用できますが、データアクセスが中断されないように、サーバー管理者とユーザーの双方が保存された認証資格情報を更新する必要があります。

1: OAuth クライアント ID、クライアントシークレット値、リダイレクト URL の準備

カスタム OAuth クライアントを構成する前に、以下の情報が必要です。この情報を準備したら、サイトのカスタム OAuth クライアントを登録できます。

- **OAuth クライアント ID とクライアントシークレット値:** 最初に OAuth クライアントをデータプロバイダー (コネクタ) に登録して、Tableau Server 用に生成されたクライアントの ID とシークレット値を取得します。
- **リダイレクト URL:** 正しいリダイレクト URL であることを確認します。これは、以下の**ステップ 2** の登録プロセスで必要になります。

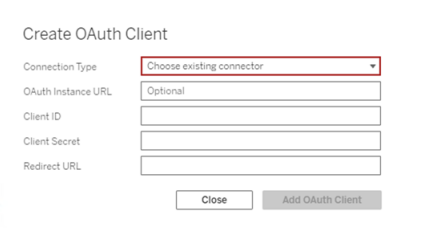
  https://<your_server_name> .com / auth / add_oauth_token

  例: https://example.com/auth/add_oauth_token

2: OAuth クライアント ID とクライアント シークレットの登録

以下の手順に従って、カスタム OAuth クライアントをサイトに登録します。

1. 管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
2. [OAuth Clients Registry (OAuth クライアント レジストリ)] で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックします。
3. 上記の**ステップ 1** の情報を含む、必要な情報を入力します。
   a. **[接続タイプ]** で、カスタム OAuth クライアントを設定するコネクタを選択します。
   b. 複数の OAuth クライアントを登録する場合は、**[OAuth インスタンス URL]** に情報を入力する必要があります。それ以外の場合は、任意で入力してください。
   c. **クライアント ID**、**クライアント シークレット**、**リダイレクト URL** には、上記の**ステップ 1** で準備した情報を入力します。
   d. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックして、登録プロセスを完了します。



4. (オプション) サポートされているすべてのコネクタに対して、ステップ 3 を繰り返します。
5. [設定] ページの下部または上部にある **[保存]** ボタンをクリックして、変更を保存します。

3: 保存した認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイト ユーザー) は、以前に保存した認証資格情報を削除してから、サイトのカスタム Oauth クライアントを使用できるよう

に再度追加する必要があります。

1. **[マイ アカウント設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。
   a. 上記の**ステップ 2** でカスタム OAuth クライアントを構成したコネクタの既存の、保存済み認証資格情報の横にある**[削除]** をクリックします。
   b. コネクタ名の横にある**[追加]** をクリックし、プロンプトに従って 1) 上記の**ステップ 2** で構成したカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

4: 保存した認証資格情報を更新するようにユーザーに通知する

上記の**ステップ 2** で構成したカスタム OAuth クライアントのコネクタの保存済み認証資格情報を更新するようにサイト ユーザーに通知します。サイトのユーザーは、保存済み認証資格情報の更新で説明されている手順を使用して、保存された認証資格情報を更新できます。

Dropbox 向け OAuth の設定

このトピックでは、OAuth 認証用に Dropbox データ ソースを設定する方法を説明します。各 Tableau Server インスタンスでこれらのステップを実行します。

Dropbox 向け OAuth の設定は、次のタスクで構成されています。

1. Dropbox 開発者ポータルのアプリコンソールで新しいアプリを作成します。
2. 新しいアプリの作成の一環として取得した情報を使用してサーバーを構成します。
3. (オプション) サイト固有の OAuth を構成します。

ステップ 1: 新しいアプリを作成する

1. Dropbox 開発者コンソールにサインインし、**[App console (アプリ コンソール)]** を選択します。
2. **[Create app (アプリの作成)]** ボタンをクリックします。
3. アプリを構成して、アプリに名前を付け、**[Create app (アプリの作成)]** ボタンを選択します。

4. アプリを作成したら、[Permissions (パーミッション)] タブに移動し、files.content.read パーミッションが選択されていることを確認します。
5. **[設定]** タブに移動し、Tableau Server のインターネットアドレスを使用してリダイレクト URI を追加します。
6. 文字列 auth/add_oauth_token を URI の末尾に追加します。例:
7. https://your_server_url.com/auth/add_oauth_token
8. **[設定]** タブからアプリキー、アプリシークレット、リダイレクト URI をコピーします。

ステップ 2: Dropbox 用に Tableau Server を構成する

Tableau Server コンピューターで bash シェルを開き、次の tsm コマンドを実行します。

```
tsm configuration set -k oauth.dropbox.redirect_uri -v <your_
authorized_redirect_uri>
```

```
tsm configuration set -k oauth.dropbox.client_id -v <your_app_key>
```

```
tsm configuration set -k oauth.dropbox.client_secret -v <your_app_
secret>
```

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージを表示しないようにすることができます。ただし、これを行っても、再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、「tsm pending-changes apply」を参照してください。

サイトのカスタム OAuth の構成

サイトに対してカスタム Dropbox OAuth クライアントを構成できます。

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

カスタム OAuth クライアントが構成されている場合、サイトレベルの構成はサーバー側の構成よりも優先され、作成されたすべての新しい OAuth 認証資格情報では既定でサイトレベルの OAuth クライアントが使用されます。構成を有効にするために、Tableau Server を再起動する必要はありません。

重要: カスタム OAuth クライアントを構成する前に確立された既存の OAuth 認証資格情報は一時的に使用できますが、データアクセスが中断されないように、サーバー管理者とユーザーの双方が保存された認証資格情報を更新する必要があります。

1: OAuth クライアント ID、クライアント シークレット値、リダイレクト URL の準備

カスタム OAuth クライアントを構成する前に、以下の情報が必要です。この情報を準備したら、サイトのカスタム OAuth クライアントを登録できます。

- **OAuth クライアント ID とクライアント シークレット値:** 最初に OAuth クライアントをデータプロバイダー (コネクタ) に登録して、Tableau Server 用に生成されたクライアントの ID とシークレット値を取得します。
- **リダイレクト URL:** 正しいリダイレクト URL であることを確認します。これは、以下の**ステップ 2** の登録プロセスで必要になります。

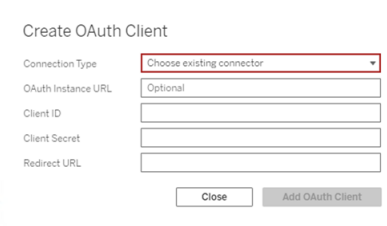
  https://<your_server_name> .com / auth / add_oauth_token

  例: https://example.com/auth/add_oauth_token

2: OAuth クライアント ID とクライアント シークレットの登録

以下の手順に従って、カスタム OAuth クライアントをサイトに登録します。

1. 管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
2. [OAuth Clients Registry (OAuth クライアントレジストリ)] で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックします。
3. 上記の**ステップ 1** の情報を含む、必要な情報を入力します。
   a. **[接続タイプ]** で、カスタム OAuth クライアントを設定するコネクタを選択します。
   b. 複数の OAuth クライアントを登録する場合は、**[OAuth インスタンス URL]** に情報を入力する必要があります。それ以外の場合は、任意で入力してください。
   c. **クライアント ID**、**クライアント シークレット**、**リダイレクト URL** には、上記の**ステップ 1** で準備した情報を入力します。
   d. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックして、登録プロセスを完了します。



4. (オプション) サポートされているすべてのコネクタに対して、ステップ 3 を繰り返します。
5. [設定] ページの下部または上部にある **[保存]** ボタンをクリックして、変更を保存します。

3: 保存した認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイトユーザー) は、以前に保存した認証資格情報を削除してから、サイトのカスタム Oauth クライアントを使用できるように再度追加する必要があります。

1. **[マイ アカウント設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。
    a. 上記の**ステップ 2** でカスタム OAuth クライアントを構成したコネクタの既存の、保存済み認証資格情報の横にある **[削除]** をクリックします。
    b. コネクタ名の横にある **[追加]** をクリックし、プロンプトに従って 1) 上記の**ステップ 2** で構成したカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

4: 保存した認証資格情報を更新するようにユーザーに通知する

上記の**ステップ 2** で構成したカスタム OAuth クライアントのコネクタの保存済み認証資格情報を更新するようにサイト ユーザーに通知します。サイトのユーザーは、保存済み認証資格情報の更新で説明されている手順を使用して、保存された認証資格情報を更新できます。

Google での OAuth の設定

既定では、Google アナリティクスとGoogle BigQuery のコネクタでは、プロバイダーによりTableau Server 用に生成され、同一サイト上のすべてのユーザーによって共有される OAuth トークンの管理キーチェーンを使用します。既定では、Google ドライブ接続では保存済み認証資格情報を使用します。Tableau 2022.3 以降では、Tableau Server を Google 用の OAuth クライアント ID およびシークレットを使用して設定する必要があります。

各コネクタの OAuth クライアント ID とシークレットを使用して Tableau Server を構成することで、管理キーチェーンを使用するコネクタを変換し、保存済み認証資格情報を使用することができます。

このトピックでは、保存済み認証資格情報を使用して、OAuth 用の Google 接続を設定する方法を説明します。各 Tableau Server インスタンスでこれらのステップを実行します。

管理キーチェーンと保存済み認証資格情報の詳細については、「OAuth 接続」を参照してください。

注:

- すべての Google ベースのコネクタには、管理キーチェーン (既定)、サーバー全体の OAuth、またはサイト固有の OAuth のいずれかが必要です。
- 保存済み認証資格情報をサイトに使用するには、最初にサーバー全体の OAuth を構成する必要があります。
- サーバー全体の OAuth は、サイト全体の OAuth が構成されているかどうかに関係なく使用できます。
- サイト固有の OAuth を使用する場合は、各サイトを個別に構成する必要があります。
- ライブ接続プロンプト、接続の編集、および Web 作成をサポートするには、管理キーチェーンを保存済み認証資格情報に変換して、エラーを回避します。

手順の概要

OAuth を設定するには、次の 3 つの一般的なステップを実行します。

1. API アクセスを有効にし、Google からアクセス トークンを作成します。
2. ステップ 1 で取得した情報を使用して Tableau Server を構成します。
3. (オプション) サイト固有の OAuth を構成します。
4. Google データ ソースを作成して編集します。

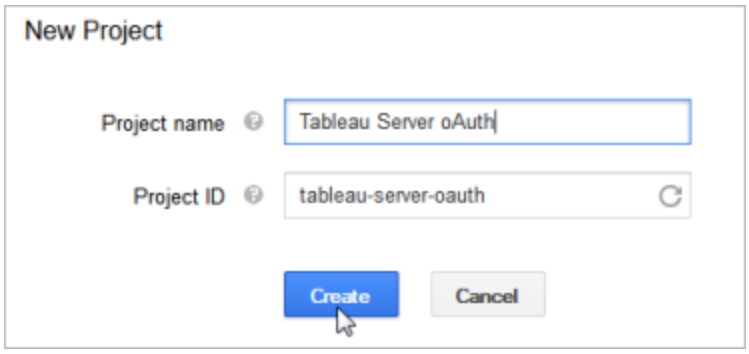
クライアント ID を取得して Google API を有効にする

注:これらの手順は、本書の執筆中に Google Cloud Platform コンソールの設定を反映します。詳細については、Google Developers Console Help の「Using OAuth 2.0 for Web Server Applications (Web サーバー アプリケーションでの Oauth 2.0 の使用)」を参照してください。

1. Google Cloud Platform にサインインし、**[Go to my console]** をクリックします。
2. ドロップダウン メニューで **[プロジェクトの選択]** をクリックして、**[プロジェクトの作成]** を選択します。
3. 表示される新しいプロジェクトフォームで次を実行します。

- プロジェクトに、このプロジェクトを使用するTableau Serverインスタンスを反映するような意味のある名前を付けてください。
- プロジェクトIDを変更するかを決めます。

> **注:** プロジェクトを作成した後は、プロジェクトID を変更することはできません。詳細を見るには、クエスチョン マークのアイコンをクリックしてください。



4. 新しいプロジェクトを開き、**[API とサービス] > [OAuth 同意画面]** に移動して、[ユーザー タイプ] を選択します。
5. **[OAuth 同意画面]** タブをクリックし、ユーザーに表示される **[製品名]** に意味のある名前を入力します。
6. **[認証資格情報] > [認証資格情報の作成]** タブの順にクリックしてから、**[OAuth クライアント ID] をクリック**します。
7. **[OAuth クライアント IDの作成]** 画面で、必須フィールドに入力します。OAuth トークンを承認する手順に従います。
   - **[Web Application]** を選択します。
   - クライアント名を入力します。

- **Authorized JavaScript Origins** の場合、**[ADD URI (URI の追加)]** をクリックして、HTTP または HTTPS を使用して Tableau Server のドメイン名を入力します。
- **Authorized Redirect URI** の場合、**[URI の追加]** をクリックして、サンプル テキストを Tableau Server のインターネット アドレスに置換し、末尾に **auth/add_oauth_token** を追加します。例は次のとおりです。

  ```
  https://your_server_url.com/auth/add_oauth_token
  ```

8. 認証済みのリダイレクト URI をコピーして、Tableau Server コンピューターからアクセス可能な場所に貼り付けます。
9. **[作成]** をクリックします。
10. Google が返す次の値をコピーし、使用している Tableau Server コンピューターからアクセスすることができる場所に貼り付けます。
    - クライアント ID
    - クライアント シークレット
11. **[APIs & Services (API とサービス)]** で、**[BigQuery API]**、**[Google ドライブ API]** (Google スプレッドシートを有効にするため)、または **[Analytics API (アナリティクス API)]** が有効になっていることを確認します。API を有効にするには、ページの一番上で **ENABLE API** をクリックします。

> **注:** Tableau Server と Google アナリティクス 4 の間の接続を確立するには、Google コンソールで Google アナリティクス管理 API と Google アナリティクス データ API の両方を有効にする必要があります。これらの API を追加すると、プロセス中に発生する可能性のある権限エラーを防ぐことができます。

Google OAuth のための Tableau Server の構成

クライアントID を取得してGoogle API を有効にするの手順を実行して取得した情報を使用することで、Tableau Serverを構成します。

- Tableau Server PC でシェルを開き、以下のコマンドを実行してアクセストークンとURI を指定します。

  ```
  tsm configuration set -k oauth.google.client_id -v <your_
  client_ID>

  tsm configuration set -k oauth.google.client_secret -v <your_
  client_secret>

  tsm configuration set -k oauth.google.redirect_uri -v <your_
  authorized_redirect_URI>

  tsm pending-changes apply
  ```

  保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

サイトのカスタム OAuth の構成

サイトに対してカスタム GoogleOAuth クライアントを構成できます。

カスタム OAuth クライアントを構成して、1) サーバー用に構成されている場合は OAuth クライアントを上書きするか、2) 一意の OAuth クライアントを必要とするデータに安全に接続するためのサポートを有効にすることを検討してください。

カスタム OAuth クライアントが構成されている場合、サイトレベルの構成はサーバー側の構成よりも優先され、作成されたすべての新しい OAuth 認証資格情報では既定でサイトレベルの OAuth クライアントが使用されます。構成を有効にするために、Tableau Server を再起動する必要はありません。

重要: カスタム OAuth クライアントを構成する前に確立された既存の OAuth 認証資格情報は一時的に使用できますが、データアクセスが中断されないように、サーバー管理者とユーザーの双方が保存された認証資格情報を更新する必要があります。

1: OAuth クライアント ID、クライアントシークレット値、リダイレクト URL の準備

カスタム OAuth クライアントを構成する前に、以下の情報が必要です。この情報を準備したら、サイトのカスタム OAuth クライアントを登録できます。

- **OAuth クライアント ID とクライアントシークレット値:** 最初に OAuth クライアントをデータプロバイダー (コネクタ) に登録して、Tableau Server 用に生成されたクライアントの ID とシークレット値を取得します。
- **リダイレクト URL:** 正しいリダイレクト URL であることを確認します。これは、以下の**ステップ 2** の登録プロセスで必要になります。

  https://<your_server_name> .com / auth / add_oauth_token

  例: https://example.com/auth/add_oauth_token

2: OAuth クライアント ID とクライアントシークレットの登録

以下の手順に従って、カスタム OAuth クライアントをサイトに登録します。

1. 管理者の認証資格情報を使用して Tableau Server にサインインし、**[設定]** ページに移動します。
2. [OAuth Clients Registry (OAuth クライアントレジストリ)] で、**[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックします。

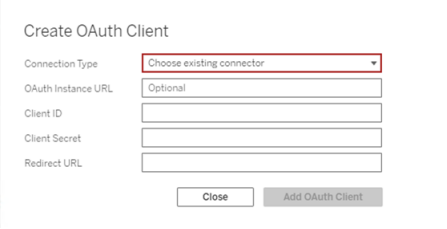
3. 上記の**ステップ 1** の情報を含む、必要な情報を入力します。

    a. **[接続タイプ]** で、カスタム OAuth クライアントを設定するコネクタを選択します。

    b. 複数の OAuth クライアントを登録する場合は、**[OAuth インスタンス URL]** に情報を入力する必要があります。それ以外の場合は、任意で入力してください。

    c. **クライアント ID**、**クライアント シークレット**、**リダイレクト URL** には、上記の**ステップ 1** で準備した情報を入力します。

    d. **[Add OAuth Client (Oauth クライアントの追加)]** ボタンをクリックして、登録プロセスを完了します。



4. (オプション) サポートされているすべてのコネクタに対して、ステップ 3 を繰り返します。
5. [設定] ページの下部または上部にある **[保存]** ボタンをクリックして、変更を保存します。

3: 保存した認証資格情報の検証と更新

データアクセスが中断されないようにするため、サーバー管理者 (およびサイトユーザー) は、以前に保存した認証資格情報を削除してから、サイトのカスタム Oauth クライアントを使用できるように再度追加する必要があります。

1. **[マイ アカウント設定]** ページに移動します。
2. **[データ ソースの保存済み認証資格情報]** で、次の手順を実行します。

    a. 上記の**ステップ 2** でカスタム OAuth クライアントを構成したコネクタの既存の、保存済み認証資格情報の横にある **[削除]** をクリックします。

b. コネクタ名の横にある **[追加]** をクリックし、プロンプトに従って 1) 上記の**ステップ 2** で構成したカスタム OAuth クライアントに接続し、2) 最新の認証資格情報を保存します。

4: 保存した認証資格情報を更新するようにユーザーに通知する

上記の**ステップ 2** で構成したカスタム OAuth クライアントのコネクタの保存済み認証資格情報を更新するようにサイトユーザーに通知します。サイトのユーザーは、保存済み認証資格情報の更新で説明されている手順を使用して、保存された認証資格情報を更新できます。

Google データソースの作成と編集

次に、Google データソースをサーバーにパブリッシュします。Tableau Desktop トピックの「Google BigQuery」を参照してください。

データソースをパブリッシュした後、最後のステップは、前に構成した埋め込みアクセストークンを使用するようにデータソース接続を編集することです。Tableau Server での接続の編集を参照してください。

アクセストークンの管理

OAuth 用のサーバーを構成したら、プロファイル設定でのアクセストークンの管理をユーザーに許可するか、トークンを一元的に管理できます。詳細については、保存済みアクセストークンの許可を参照してください。

Intuit QuickBooks Online 向け OAuth の設定

このトピックでは、OAuth 認証用に Intuit QuickBooks Online データソースを設定する方法を説明します。各 Tableau Server インスタンスでこれらのステップを実行します。

QuickBooks Online 向け OAuth の設定は、次のタスクで構成されています。

1. Intuit 開発者プラットフォームで [接続アプリ] を作成します。
2. [接続アプリ] の一部として取得した情報を使用してサーバーを構成します。

3. (オプション) サイト固有の OAuth を構成します。

ステップ 1: Intuit アプリを作成する

1. Intuit 開発者アカウントにサインインし、**[マイ アプリ]** をクリックします。
2. **[コーディングを開始したばかり]** セクションで **[API の選択]** をクリックします。
3. **[アカウント]** を選択し、**[アプリの作成]** をクリックします。
4. **[アプリの送信準備を整える]** セクションで、リンクをクリックして本番キーを取得します。

   重要:開発キーではなく、本番キーを使用する必要があります。

5. アプリトークン、OAuth コンシューマー キー、OAuth コンシューマー シークレットをコピーします。

ステップ 2: Intuit QuickBooks Online 用に Tableau Server を構成する

- Tableau Server コンピューターで bash シェルを開き、以下のコマンドを実行します。

  ```
  tsm configuration set -k oauth.quickbooks.oauth_callback_uri -v
  http://YOUR-SERVER/auth/add_oauth_token

  tsm configuration set -k oauth.quickbooks.consumer_key -v
  <your_consumer_key>

  tsm configuration set -k oauth.quickbooks.consumer_secret -v
  <your_consumer_secret>

  tsm pending-changes apply
  ```

  保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージ

はサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

アクセス トークンの管理

QuickBooks Online データソースに対して抽出の更新ジョブを実行すると、Tableau Server はアクセス トークンの更新を試みます。アクセス トークンの有効期限が切れていないことを確認するため、1 か月に 2 回以上抽出の更新ジョブを実行します。そうでない場合、QuickBooks Online のアクセス トークンの有効期限が切れ、抽出の更新ジョブが失敗します。アクセス トークンの有効期限が切れている場合、**[設定]** ページから保存済みの認証資格情報を編集できます。

保存されている認証資格情報は、一元的に、またはユーザーによって管理できます。詳細については、保存済みアクセス トークンの許可を参照してください。

OAuth 接続のトラブルシューティング

このトピックでは、OAuth データ接続を構成する際に発生する可能性のある問題の解決に関する情報を示します。

競合エラー

場合によっては、OAuth を使用して接続しようとするとエラーが表示されることがあります。エラー メッセージの最初の文は次のとおりです。

The server encountered an internal error or misconfiguration and was unable to complete your request. (内部エラーまたは構成ミスが原因で、サーバーが要求を完了できませんでした。)

このエラーは、Tableau Server の完全修飾ドメイン名 (FQDN) を Tableau Server の承認リスト リダイレクト キーに追加する必要があることを示しています。

ユーザーがローカル ホスト名 (https://tableau) によって Tableau Server にアクセスし、OAuth データ プロバイダーがパブリック DNS 名 (https://data.example.com) に応答しているとき、Tableau Server は外部 FQDN をローカル サーバー名に関連付ける必要があります。ローカル ホスト名は、ユーザーが内部ネットワークから Tableau Server にアクセスするときに入力する URL 内のサーバー名です。

このエラーを解決するには、`oauth.whitelisted.redirect_to_origin_host` キー オプションを指定して tsm configuration set を実行します。このキーは、値のペア `"internal_host,FQDN1,FQDN2"` を受け取ります。たとえば、次のコマンドではローカル ホスト名が *tableau* に設定され、FQDN が *tableau.example.com* に設定されます。

```
tsm configuration set -k oauth.whitelisted.redirect_to_origin_host -
v "tableau,tableau.example.com"

tsm pending-changes apply
```

内部 Tableau Server へのアクセスに複数のパブリック URL を使用する場合は、次の例のように追加 FQDN をコンマで区切ってコマンドに追加します。

```
tsm configuration set -k oauth.whitelisted.redirect_to_origin_host -
v "tableau,tableau.example.com,tableau2.example.com"
```

既存の承認リストのリダイレクト構成を編集する場合は、完全なマッピング セットを入力する必要があります。既存の構成キーを切り詰めたり追加したりすることはできません。

SAP HANA SSO の構成

SAML 委任を使用して SAP HANA にシングルサインオン (SSO) エクスペリエンスを提供するよう Tableau Server を構成できます。このシナリオは Tableau Server に対する SAML 認証に依存しません。HANA SAML 委任を使用するために Tableau Server で SAML にサインオンする必要はありません。好きな方法を選択して Tableau Server にサインインできます。

SAP HANA で SAML 委任を使うと、Tableau Server がアイデンティティプロバイダーとして機能します。

はじめる前に

SAP HANA で SAML 委任を構成するには、Tableau Server および SAP HANA 上の両方で構成が必要です。このトピックでは、Tableau Server の構成についての構成情報を提供します。Tableau Server を構成する前に、次を完了しなくてはなりません。

- Tableau Server のためにSAML 証明書およびキーファイルを取得します。
    - 証明書ファイルは、ファイル拡張子 .crt または .cert が付いた PEM 暗号化 x509 証明書でなくてはなりません。このファイルは Tableau Server によって使用され、HANA 上にもインストールする必要があります。
    - 秘密鍵は、PKCS#8 形式の DER エンコードされた秘密鍵ファイルで、パスワード保護されておらず、ファイル拡張子が .der である必要があります。このファイルは Tableau Server のみによって使用されます。
- HANA に証明書をインストールします。HANA での `libxmlsec` エラーを回避するために、SAP HANA でインメモリ証明書ストアを構成することをお勧めします。詳細については、「SAP サポートのトピック」を参照してください。
- 最新バージョンの SAP HANA ドライバー (最低でもバージョン 1.00.9) を Tableau Server にインストールします。
- Tableau Server から SAP HANA へのネットワーク暗号化を構成します (推奨)。

証明書 / キーペアの生成、SAML 接続の暗号化と SAP HANA の構成に関しての詳細は、Tableau コミュニティの「How to Configure SAP HANA for SAML SSO with Tableau Server (Tableau Server で SAP HANA を SAML SSO のために構成する方法)」を参照してください。

SAP HANA での Tableau Server SAML を構成する

次の手順は、Tableau Server 上で `tsm data-access` を使用して SAP HANA での SAML を構成する方法を説明しています。sapHanaSettings エンティティを使用しても、SAP HANA での SAML が構成できます。

分散展開で Tableau Server を実行している場合は、最初のノードで次の手順を実行します。

1. 証明書ファイルを saml フォルダーに配置します。例は次のとおりです。

   ```
   /var/opt/saml
   ```

2. 次のコマンドを実行し、証明書とキーファイルの場所を指定します。

   ```
   tsm data-access set-saml-delegation configure --cert-key <cert-key> --cert-file <cert-file>
   ```

   <cert-key> および <cert-file> にはそれぞれ秘密キーおよび証明書のファイルパスが入ります。

   例は次のとおりです。

   ```
   tsm data-access set-saml-delegation configure --cert-key /var/opt/saml/hana_pkey_pkcs8.der --cert-file /var/opt/saml/hana_cert.pem
   ```

   他のオプションも指定できます。例えば、ユーザー名形式と資格認証情報を正規化する方法が指定できます。tsm data-access を参照してください。

3. 次のコマンドを実行して委任を有効にします。

   ```
   tsm data-access set-saml-delegation enable

   tsm configuration set -k wgserver.sap_hana_sso.enabled -v true

   tsm configuration set -k wgserver.delegation.enabled -v true
   ```

4. 終了した後に、tsm pending-changes apply を実行します。

   保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合

は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

キーペア認証のための Snowflake の設定

キーペア認証で保護された Snowflake データソースを使用して Tableau Desktop または Tableau Prep から Tableau Cloud または Tableau Server サイトにワークブックまたはフローをパブリッシュするには、それらの認証資格情報に対応するようにそのサイトを設定する必要があります。これには、Snowflake キーペア認証の新規または更新した認証情報の詳細を、アカウント設定の [保存済み認証資格情報] セクションに追加する必要があります。

注

- Snowflake でキーペア認証を使用するには、バージョン 3.4.0 以降の ODBC ドライバーを Snowflake からインストールする必要があります。
- この認証用のキーを作成するには、OpenSSL バージョン 3.x 以降を使用する必要があります。
- キーペア認証で保護された Snowflake データを使用する Tableau Desktop の各インスタンスでは、データへの接続の一環としてユーザーがキーファイルを提供する必要があります。詳細については、「Snowflake」を参照してください。
- Snowflake キーペア認証を使用したデータソース、ワークブック、フローのパブリッシュは、Web 作成、tabcmd、REST API ではサポートされていません。Tableau Desktop を使用してデータソースまたはワークブックを作成するか、Tableau Prep Builder を使用してフローを作成してから、それらを Tableau Cloud または Tableau Server にパブリッシュする必要があります。

キーペア認証の詳細を追加する

1. Tableau Cloud または Tableau Server にサインインし、**[マイ アカウント設定]** のページに移動します。
2. **[セキュリティ]** セクションで **[データ ソースの保存済み認証資格情報]** を見つけます。
3. **[Snowflake]** の横にある **[追加]** をクリックします。
4. **[新しい保存済み認証資格情報の追加]** ウィンドウで **[キーペア認証]** を選択し、**[追加]** をクリックします。

5. **[キーペア認証資格情報の追加]** ウィンドウで、Snowflake サーバー、オプションのロール、およびこれらの認証資格情報に関連付けられたユーザー名を入力します。

   注: ロールとユーザー名は大文字と小文字が区別され、Tableau Cloud または Tableau Server の **[マイ アカウント設定] > [Snowflake キー ペア情報]** に保存されているものと一致する必要があります。

6. **[秘密鍵ファイル]** で [ファイルの選択] をクリックし、秘密鍵が含まれている .p8 ファイルを選択します。
7. (オプション) キーにパスワードまたはパスフレーズが関連付けられている場合は、**[パスワード]** フィールドに入力して **[追加]** をクリックします。

これで、キーペア認証で保護されたワークブックとフローを Tableau Desktop または Tableau Prep Builder から自分のサイトに共有する準備ができました。

関連項目

- Snowflake - Snowflake への接続を設定し、データを取得して分析する方法を学習します。

Kerberos サービス アカウントへのアクセスの有効化

データベースへのアクセスに Kerberos サービス アカウントを使用するように Tableau Server を構成することができます。このシナリオでは、Tableau Server がサービス アカウント ("実行アカウント" とも呼ばれます) を使用してデータベースに接続します。

Tableau Server で実行アカウント認証を使用するには、まず統合認証を使用するワークブックまたはデータソースを作成する必要があります。ユーザーが Tableau Server へのパブリッシュを行う際に、実行アカウント認証のオプションが表示されます。Tableau Server の Web 作成で統合認証を使用するデータソースを作成した場合、そのデータソースでは既定で実行アカウント認証が使用されます。

> **注**: *統合認証* は、一部のコネクタでは *Windows 認証* とも呼ばれます。どちらの場合も、Tableau Server では Kerberos 認証が使用されます。

実行サービス アカウントを使用したデータ アクセス

実行アカウント認証を使用するには、実行アカウントが外部データベースへの読み取りおよびクエリパーミッションを持っている必要があります。設計上、*Creator* ロールまたは *Explorer (パブリッシュ可能)* ロールを持つ Tableau Server ユーザーは、外部データベースへのクエリ用に実行アカウントへのフル アクセス権を持っています。

たとえば、Creator ロールを持つユーザーは、実行サービス アカウントに対してアクセスが許可されているすべてのデータベースを表示できます。また、表を一覧表示し、カスタム SQL を実行できます。

Creator ユーザーが Web 作成で新しいデータ ソースを作成する際にデータベース ホスト名を指定して統合認証を選択すると、実行アカウント アクセスが許可されているデータベースがユーザーに表示されます。

データベース資産への表示アクセスは、Web 作成を使用して Tableau Server に接続するユーザーに限定されません。上記と同じロールを持ち、データベース サーバー名を知っている高度なユーザーは、実行アカウント アクセスが許可されているデータベースを表示するワークブックを Tableau Desktop で作成できます。

推奨事項

これらのシナリオでデータベースへのユーザー アクセスが許可されるかどうかを、組織で評価する必要があります。一般に、実行サービス アカウントの使用量および使用範囲を減らすと、ユーザーがデータベースのコンテンツに不注意にアクセスする可能性も減少します。ただし、実行サービス アカウントの使用量および使用範囲を減らすことで、管理者やユーザーの認証資格情報管理の負担が大きくなる場合もあります。

貴社のビジネス ニーズとデータ アクセス ポリシーに照らして、次の推奨を評価してください。

- まず、Creator ロールまたは Explorer (パブリッシュ可能) ロールを持つすべてのユーザーを必ず信頼してください。Tableau で整合性のあるアクションを実行するには、これらのユーザーに依存する必要があります。
- 実行サービス アカウントでアクセスするデータ ソースに対する権限をパブリッシュしているユーザーの中に信頼できないユーザーがいる場合は、それらのデータ ソースに認証資格情報を埋め込むことを検討する必要があります。

- データソースで抽出更新の自動化が設定されていない場合、つまり、データソースが主にライブ接続でアクセスされている場合は、Kerberos 委任を使用できる可能性があります。要件については、Kerberos 委任の有効化を参照してください。

要件

- MIT Kerberos はサポートされません。
- 実行サービス アカウントには、ターゲット データベースへの読み取りアクセス権が付与されている必要があります。

構成プロセス

このセクションでは、Kerberos サービス アカウントへのアクセスを有効にするプロセスについての例を示します。

1. ドメイン ユーザー アカウントを作成し、実行サービス アカウントとして操作します。このアカウントには、ターゲット データベースへの読み取りアクセス権が付与されている必要があります。

   この例では、実行サービス アカウントは `tabsrv@example.com` というユーザー プリンシパル名です。

2. 実行サービス アカウントのキータブ ファイルを作成します。

   たとえば、ktutil ツールを使用して、次のコマンドでキータブ (`tabsrv-runas.keytab`) を作成します。

   ```
   ktutil
   ```

   ```
   ktutil:  addent -password -p tabsrv@EXAMPLE.COM -k 2 -e
   <encryption scheme>
   ```

   このコマンドの暗号化スキームには、`RC4-HMAC`、`aes128-cts-hmac-sha1-96`、`aes256-cts-hmac-sha1-96` などがあります。環境およびデータソースに適した暗号化スキームについては、IT チームにお問い合わせください。

   ```
   ktutil:  wkt tabsrv-runas.keytab
   ```

   Tableau Server では、実行サービス アカウントおよび関連付けられたキータブを使用して認証を行い、データベースへ直接接続します。

3. Tableau Server データディレクトリにキータブをコピーし、適切な所有権とパーミッションを設定します。キータブは権限なしユーザーが読み取り可能である必要があります。Tableau セットアップで作成される既定の権限なしユーザーは tableau です。

   マルチノード展開を実行している場合は、クラスタ内の各ノードで以下のコマンドを実行する必要があります。

   ```
   mkdir /var/opt/tableau/tableau_server/keytab
   sudo cp -p tabsrv-runas.keytab /var/opt/tableau/tableau_
   server/keytab
   sudo chown $USER /var/opt/tableau/tableau_server/keytab/tabsrv-
   runas.keytab
   chgrp tableau /var/opt/tableau/tableau_server/keytab/tabsrv-
   runas.keytab
   chmod g+r /var/opt/tableau/tableau_server/keytab/tabsrv-
   runas.keytab
   ```

4. 次の TSM コマンドを実行して、実行アカウントアクセスの有効化、実行サービスアカウントの設定、キータブファイルとサービスアカウントの関連付けを行います。

   ```
   tsm configuration set -k features.RunAsAuthLinux -v true --
   force-keys
   tsm configuration set -k native_api.datasource_runas_principal
   -v tabsrv@EXAMPLE.COM --force-keys
   tsm configuration set -k native_api.datasource_runas_keytab_
   path -v /var/opt/tableau/tableau_server/keytab/tabsrv-
   runas.keytab --force-keys
   ```

5. 次の TSM コマンドを実行して Tableau Server 展開に変更を適用します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのよ

うにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

JDBC コネクタの Kerberos 実行アカウント認証を有効にする

バージョン 2020.2 の時点で、Tableau Server は JDBC コネクタの Kerberos 認証をサポートしています。

データベースへのアクセスに Kerberos サービス アカウントを使用するように Tableau Server を構成することができます。このシナリオでは、Tableau Server がサービス アカウント(「実行サービス アカウント」とも呼ばれます) を使用してデータベースに接続します。このシナリオは、「実行アカウント認証」と呼ばれます。

Tableau Server で実行アカウント認証を使用するには、まず統合認証を使用する Tableau Desktop にワークブックまたはデータ ソースを作成する必要があります。Tableau Server へのパブリッシュを行う際に、実行アカウント認証のオプションが表示されます。Web 作成を使用してデータ ソースを作成する場合、統合認証を選択した場合の既定の操作は、実行アカウント認証です。

サポートされているデータ ソース

Tableau では次のデータ ソースを使用する JDBC Kerberos 委任がサポートされています。

- Oracle
- PostgreSQL

ネイティブと JDBC ベースの両方のコネクタは、Linux 上の Tableau Server で同じ構成を使用します。実行アカウント認証を構成するには、Kerberos サービス アカウントへのアクセスの有効化 を参照してください。

SQL Server の偽装

Tableau Server の文脈における偽装とは、あるユーザー アカウントが別のユーザー アカウントの代わりにアクションを実行することです。Tableau Server データベース ユーザーの偽装を行うように Tableau と Microsoft SQL Server を構成できます。これにより、TableauServer によって使用され

るSQL Server データベース アカウントは、Tableau ユーザーでもあるSQL Server データベース ユーザーの代わりにクエリを実行できます。

この機能の主な利点は、管理者がデータセキュリティポリシーを1 か所、つまりデータベースに実装して制御できることです。Tableau ユーザーがSQL Server データベースへのライブ接続でビューにアクセスすると、ビューにはユーザーのデータベース パーミッションで表示が許可されている内容のみが表示されます。もう1 つの利点は、ユーザーがビューを開くときにデータベース サインインプロンプトに応答する必要がないことです。また、ワークブックのパブリッシャーは、ビューに表示する内容を制限するためにユーザー固有のフィルターに依存する必要がありません。

偽装の要件

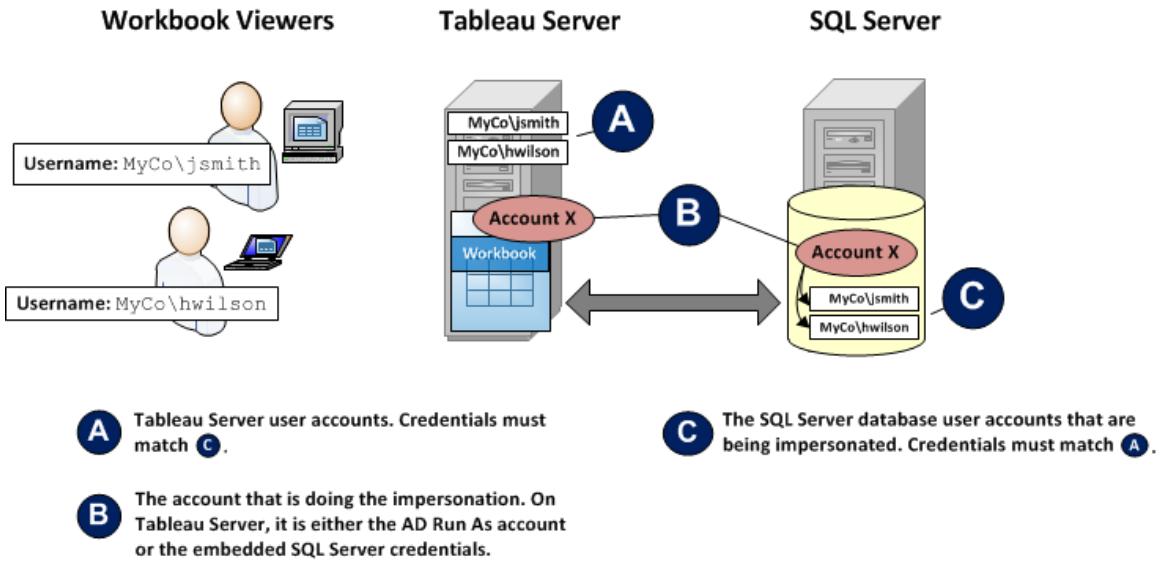
この機能を使用するための要件は次のとおりです。

- **SQL Server へのライブ接続のみ:** 偽装は、SQL Server データベースのバージョン2005 以降にライブ接続されているビューに対してのみ使用できます。
- **個別のデータベース アカウント:** ビューにアクセスする各ユーザーは、ビューの接続先であるSQL Server データベースに明示的な個別のアカウントを持っている必要があります。Active Directory (AD) グループのメンバーは偽装できません。たとえば、Jane Smith がAD グループ「Sales (販売)」のメンバーであり、そのデータベース管理者がSQL Server データベースにAD グループ「Sales (販売)」を追加した場合、Jane を偽装することはできません。
- **認証資格情報と認証タイプの一致:** 各 Tableau ユーザー アカウントの認証資格情報およびその Tableau ユーザー認証タイプが、SQL Server データベースの認証資格情報および認証タイプと一致する必要があります。たとえば、Jane Smith の Tableau Server ユーザーアカウントが `MyCo\jsmith` である場合、SQL Server データベース上のユーザー名も `MyCo\jsmith` である必要があります。SQL Server は、Windows 統合認証を使用している必要があります。
- **SQL Server の前提条件:** SQL Server では、データセキュリティを確保するビューであるデータセキュリティテーブルが必要であり、データベース ユーザーがそのビューを使用することを要求することをお勧めします。

- **SQL IMPERSONATE アカウント:** 上記のデータベース ユーザーに対する IMPERSONATE パーミッションを持つ SQL Server データベース アカウントが必要です。これは、sysadmin の役割を持つアカウント、または各ユーザー アカウントに対する IMPERSONATE パーミッションを与えられているアカウントのいずれかです (EXECUTE AS に関する MSDN の記事を参照)。SQL Server アカウントは、次のいずれかである必要があります。
    - Tableau Server 実行サービス アカウント。Kerberos サービス アカウントへのアクセスの有効化を参照してください。
    - ワークブック パブリッシャーのアカウント。埋め込み SQL 認証資格情報の擬装を参照してください。

擬装のしくみ

下の図は、データベース ユーザーの擬装のしくみを示しています。



上の図で、Jane Smith (MyCo\jsmith) は西海岸の営業担当者、Henry Wilson (MyCo\hwilson) は東部の営業担当者です。SQL Server データベースで、Jane のアカウント MyCo\jsmith に対するアカウント パーミッションでは、西海岸のデータにしかアクセスできません。Henry のアカウント MyCo\hwilson では、東海岸のデータにしかアクセスできません。

全国のデータを表示するビューが作成されています。このビューは、SQL Server データベースにライブ接続します。この 2 人のユーザーが Tableau Server にサインインし、ビューをクリックします。Tableau Server は、各ユーザーのデータベース アカウントに対する IMPERSONATE パーミッションを持つデータベース アカウントを使用して SQL Server に接続します。このアカウントが各ユーザーのデータベース アカウントの代わりにアクションを実行します。

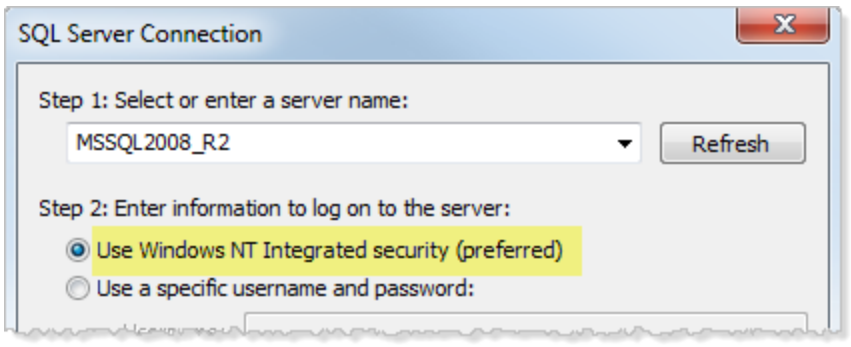
ビューを表示するときは、各ユーザーの個々のデータベース パーミッションによって制限されます。Jane には西海岸の販売データのみ、Henry には東海岸のデータのみが表示されます。

実行サービス アカウントでの擬装

擬装を行うときは、[実行するアカウント名] サービス アカウントを介した擬装をお勧めします。[実行するアカウント名] サービス アカウントとは、Tableau Server をホストしているマシンで Tableau Server サービスを実行できる Active Directory ユーザー アカウントです。この同じアカウントが、SQL Server のデータベース ユーザー アカウントに対する IMPERSONATE パーミッションを持っている必要があります。データ セキュリティの観点から、Tableau Server の実行サービス アカウントを擬装に使用することにより、管理者はほとんどの制御を行うことができます。

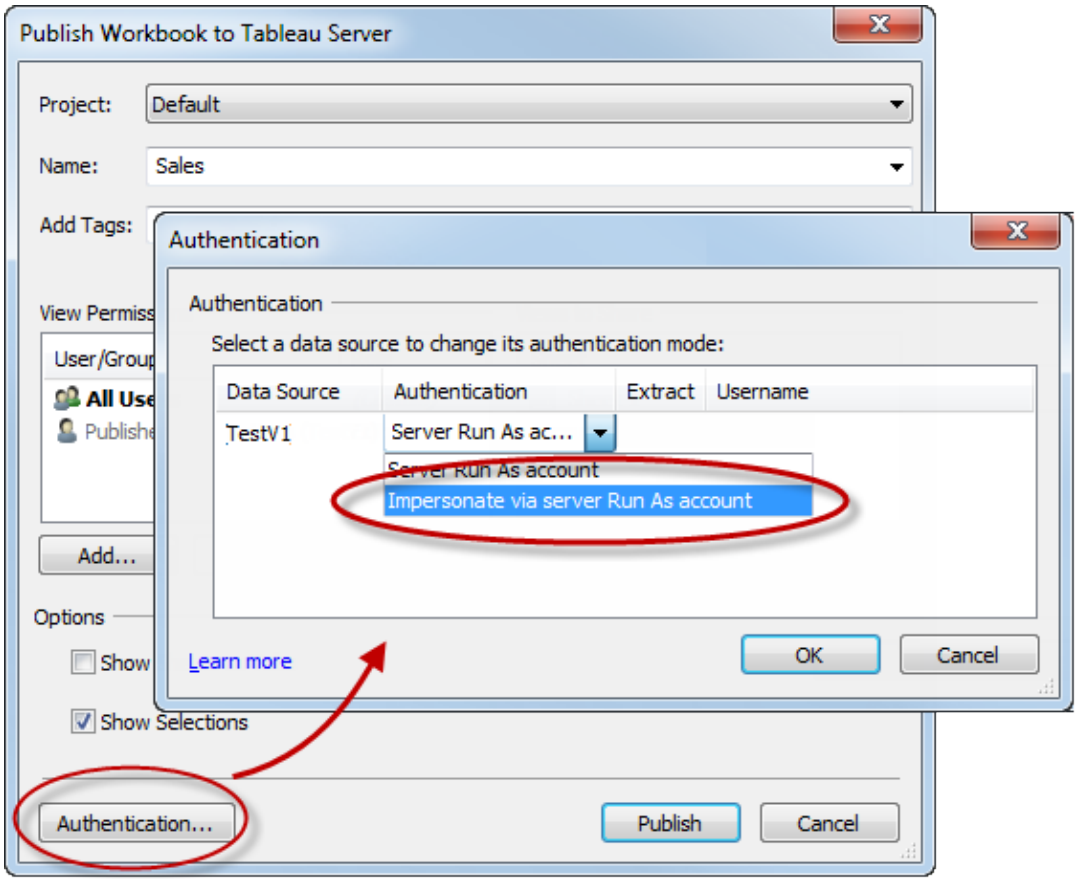
実行ユーザー アカウントの擬装を設定するには

1. Kerberos サービス アカウントへのアクセスの有効化。
2. Tableau Desktop でワークブックを作成します。データ接続を作成するときは、**[Windows NT 統合セキュリティを使用する]** をオンにして、ワークブックから SQL Server データベースへのライブ接続を行います。



3. Tableau Desktop でワークブックを Tableau Server にパブリッシュします (**[サーバー] > [ワークブックのパブリッシュ]**)。

4. [パブリッシュ] ダイアログ ボックスで [認証] をクリックし、[認証] ダイアログ ボックスでドロップダウン リストから **[サーバーの実行アカウント経由で擬装]** を選択します。



5. **[OK]** をクリックします。

6. Tableau Server にユーザーとしてサインインして、接続をテストします。ビューをクリックしたときに、データベースの認証資格情報が要求されず、そのユーザーへの表示が許可されているデータのみが表示される必要があります。

埋め込み SQL 認証資格情報の擬装

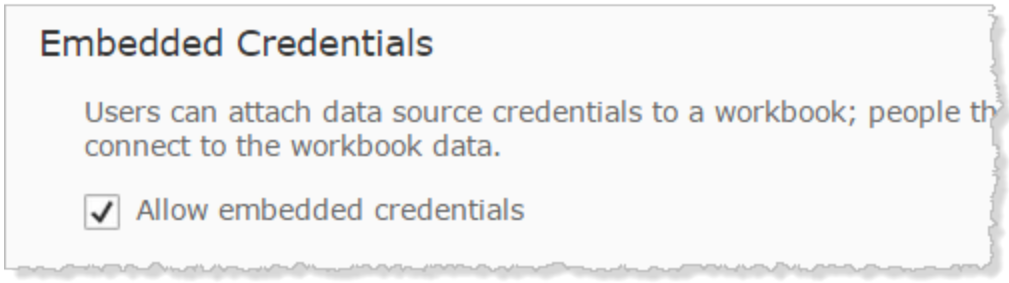
ビューをパブリッシュするユーザーに SQL Server アカウントの認証資格情報をビューに埋め込ませることによって、擬装を行うこともできます。Tableau Server はどのような種類のアカウントでも実行で

きますが、パブリッシャーが指定したこのような認証資格情報を使用してデータベースに接続します。

擬装を処理するアカウントをActive Directory (AD) アカウントにできない場合や、ワークブックのパブリッシャーにSQL Serverの潜在的に高いパーミッション レベルを持つアカウントを与えてもかまわない場合には、この方法が適しています。

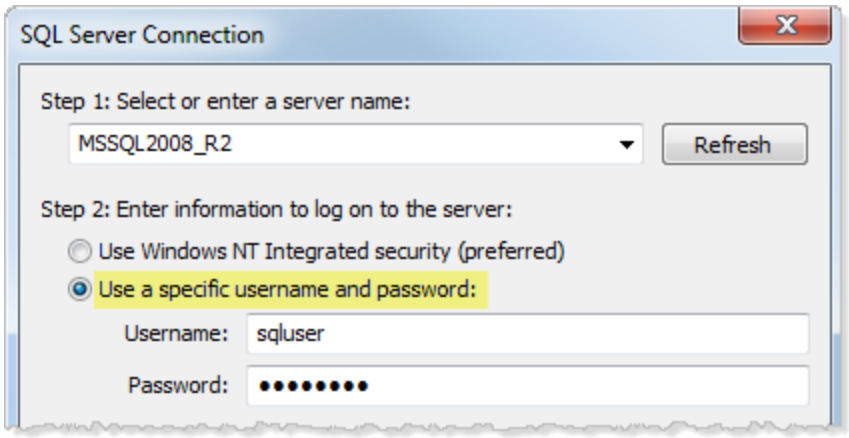
注:

この方法を使用するには、Tableau Serverのサーバーの[設定] ページで[埋め込み認証資格情報]が有効である必要があります。



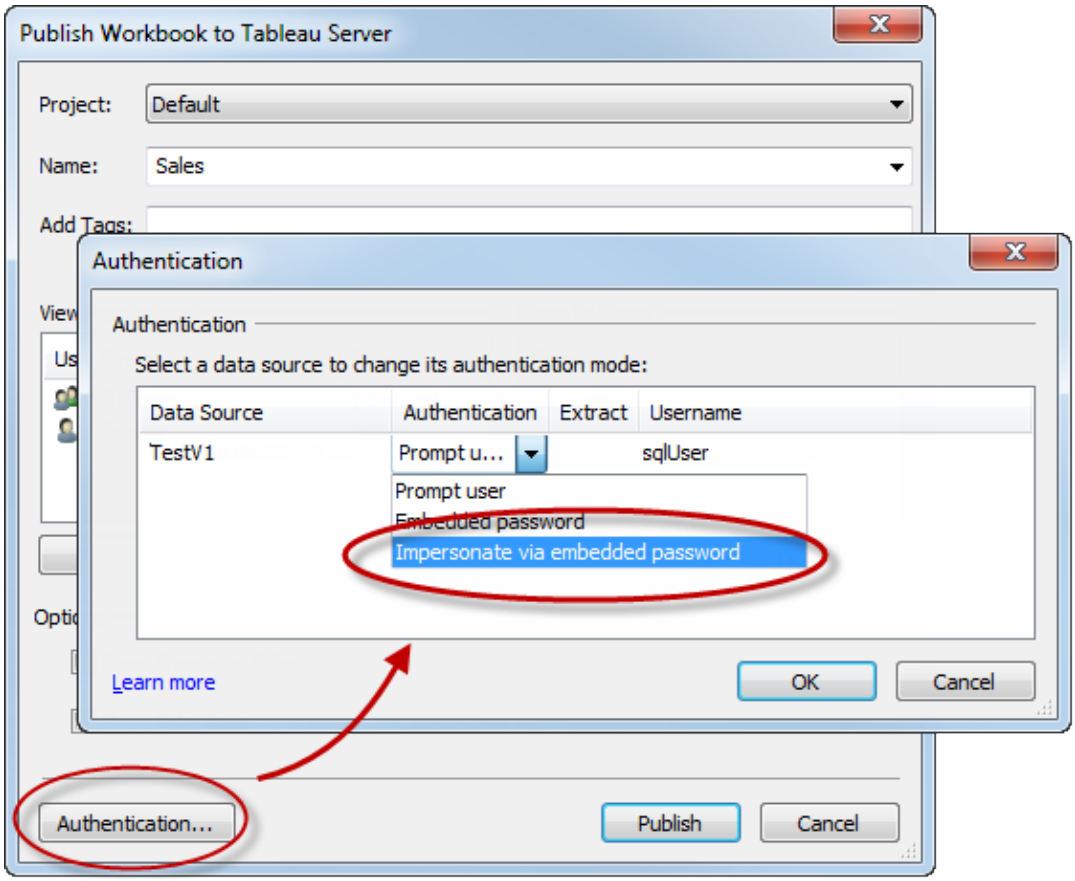
ワークブックのパブリッシャーのSQL アカウントで偽装を行うには

1. Tableau Desktop でワークブックを作成します。データ接続を作成するときは、[特定のユーザー名とパスワードを使用する] を選択して、ワークブックからSQL Server データベースへのライブ接続を行います。



2. ワークブックを Tableau Server にパブリッシュします (**[サーバー] > [ワークブックのパブリッシュ]**)。

3. [パブリッシュ] ダイアログ ボックスで [認証] をクリックし、[認証] ダイアログ ボックスでドロップダウン リストから **[埋め込みパスワード経由での偽装]** を選択します。



4. **[OK]** をクリックします。

5. Tableau Server にユーザーとしてサインインして、接続をテストします。ビューをクリックしたときに、データベースの認証資格情報が要求されず、そのユーザーへの表示が許可されているデータのみが表示される必要があります。

カスタム TSM 管理グループの構成

このトピックでは、カスタム TSM 管理グループを構成する方法について説明します。

既定では、Tableau Server のインストール プロセスによって tsmadmin というグループが作成されます。このグループのユーザーには TSM 管理者としての権限が与えられます。この既定のグループは、インストール中に変更できます。initialize-tsm スクリプトの出力ヒントを参照してください。

Tableau Server が既にインストールされていて、TSM 管理に使用するグループを変更する場合は、このトピックの手順に従ってください。

任意のグループをカスタム TSM 管理グループとして使用するように Tableau Server を構成することができます。構成が完了すると、カスタム グループのメンバーに属するユーザーは TSM を管理できるようになります。

ステップ 1: 新しいグループを作成する

自分のコンピューターで新しいグループを作成する。Linux で作成された既定のグループに対するパーミッションは変更しないでください。

ステップ 2: Tableau Server の構成

カスタム TSM 管理グループ名は、tsm.authorized.groups 構成 キーに保存されます。グループ名を指定する場合 (tsmadmin以外)、tsm.authorized.groups 構成キーを更新してからTableau Server を再起動する必要があります。

tsm configuration set コマンドを使用して、グループ名の値を設定します。たとえば、TSM 管理グループ名を myadmingroup に変更するには、次のコマンドを実行します。

```
tsm configuration set -k tsm.authorized.groups -v myadmingroup

tsm pending-changes apply
```

注

- tsm.authorized.groups 構成キーを設定すると、そのキーに保存されている既存の値が上書きされます。
- 既存の tsmadmin グループにユーザーが存在し、tsm.authorized.groups 構成キーを新しい値に上書きした場合、既存の tsmadmin グループのユーザーには TSM 管理権限が付与されなくなります。
- グループ名のコンマ区切りリストを値として入力すると、複数のグループを指定できます。

ステップ 3: 新しいグループにユーザーを追加する

`tsm.authorized.groups` を設定したら、指定した新しいグループに属するユーザーは、Tableau Server で完全な TSM 管理権限を持つことになります。

認可

認可 とは、認証が検証された後に Tableau Server にアクセスできるユーザーとその方法のことです。認可に含まれるもの:

- Tableau Server でホストされるコンテンツ (プロジェクト、サイト、ワークブック、ビューを含む) でユーザーが許可されている操作。
- Tableau Server で管理されるデータソースでユーザーが許可されている操作。
- Tableau Server の管理を実行するためにユーザーが許可されているタスク (サーバー設定の構成、コマンドラインツールの実行、サイトの作成、その他のタスクなど)。

これらのアクションの認可は Tableau Server によって管理され、特定のエンティティ (ワークブックやデータソースなど) に関連付られたユーザーのサイトロールとパーミッションの組み合わせによって決まります。

サイトロール

サイトロールはどのユーザーが管理者かを定義します。管理者は、サイトレベルまたはサーバーレベルで割り当てられます。管理者以外でのサイトロールとは、ユーザーが指定されたサイトに対し、コンテンツ資産に対して設定されたパーミッションのもとで持つことができる最大アクセスレベルを指します。たとえば、1 人のユーザーに Viewer (ビューアー) のサイトロールが割り当てられ、別のユーザーには Creator Creator が割り当てられます

サイトロールの詳細については、ユーザーのサイトロールの設定を参照してください。

パーミッション

パーミッションは、指定されたユーザーが、特定のコンテンツ資産上で特定のアクションを許可されているか拒否されているかを決定します。

管理者として Tableau Server をセットアップする際には、パーミッションの評価方法を把握しておくことが重要です。Tableau パーミッションプロセスを理解することで、コンテンツやデータの共有、パブ

リッシュ、表示、抽出、インポート方法を制御できるよう、サイトやプロジェクト、その他の資産でパーミッションを設定し、構成できるようになります。

Tableau のパーミッションを理解するための 4 つの重要な概念は次のとおりです。

- **パーミッションは資産ベースです**。パーミッション ルールは、個々のコンテンツ資産 (プロジェクト、データ ソース、ワークブック) に割り当てられ、ユーザーまたはグループに対して付与できます。
- **パーミッションは非明示的に拒否され、管理者以外のユーザーはコンテンツへのアクセスを明示的に許可される必要があります**。Tableau Server によってパーミッションを「許可」または「拒否」するプロセスの詳細については、パーミッションに記載されています。
- **パーミッション継承は、ロックされたプロジェクトおよびタブ付きビューを使用するワークブックにのみ存在します**。コンテンツ パーミッションが上位プロジェクトにロックされている場合、プロジェクト階層全体に含まれるワークブック、ビュー、およびデータ ソースは、上位プロジェクトで設定されている既定パーミッションを使用します。**[シートをタブとして表示]** オプションで保存されたワークブックでは、そこに含まれるビューはワークブック パーミッションを使用します。詳細については、パーミッションを参照してください。
- **ロックされていないプロジェクトでは、初期パーミッションは、コンテナー アイテムのパーミッションのワンタイム コピーです**。データ ソースまたはワークブックは既定のパーミッションから始まりますが、承認済みユーザーはその後これらの資産でパーミッションを編集できます。既定のパーミッションおよびプロジェクトの詳細については、パーミッションを参照してください。

Tableau Server は、多数のシナリオですべてのコンテンツへのアクセスを管理できる柔軟なパーミッション インフラストラクチャを提供します。詳細については、パーミッションを参照してください。

データ アクセスと外部認可

Tableau Server および Desktop がデータへのアクセスを有効にするために外部認可に依存する場合があります。例は次のとおりです。

- 外部データ ソースに接続するユーザーは、Tableau Server の権限の範囲外の認可が必要な場合があります。ユーザーが外部データ ソースをパブリッシュすると、Tableau Server はそのデータ ソースのアクセスと機能を管理します。ただし、ユーザーがワークブックに外部データ ソースを埋め込んだ場合は、ワークブックを開く他のユーザーによるワークブックの接続先の参照元データでの認証方法は、ワークブックをパブリッシュしたユーザーによって決定します。
- Active Directory を使用し、Tableau が実行ユーザー アカウントを使用するよう構成されている組織で Tableau Server を実行すると、Active Directory および NTFS に依存して認

可を行うことになります。たとえば、実行ユーザー アカウントをSQL に接続するユーザーの偽装に使用するようTableau Server を設定している場合、オブジェクトレベルの認可は NTFS および Active Directory に依存しています。

- 特定のデータベース ソリューションによるユーザー認証や認可のしくみは異なる場合があります。前述のように、Tableau Server はデータ ソースの構成時にアクセス認可を提供するよう設定できますが、一部のデータベースでは、独自の認証スキームにしたがってアクセスを認証します。

データ セキュリティ

Tableau では、どのユーザーがどのデータを表示できるかを制御するための方法がいくつか用意されています。稼働中のデータベースに接続するデータ ソースの場合、パブリッシュされたビューをユーザーがクリックしたときにデータベースの認証資格情報の指定をユーザーに要求するかどうかも制御できます。次の 3 つのオプションを組み合わせて使用すると、さまざまな結果が得られます。

- **データベース ログイン アカウント:** 稼働中のデータベースに接続するデータ ソースを作成するときに、データベースへの認証を Windows NT を通じて行うか、データベースのビルトイン セキュリティ機構を通じて行うかを選択します。

- **認証 モード:** 稼働中のデータベース接続を使用してデータ ソースまたはワークブックをパブリッシュするとき、**[認証 モード]** を選択できます。どのモードを使用できるかは、上記のオプションの選択によって異なります。

- **ユーザー フィルター:** ワークブックまたはデータ ソースにフィルターを設定できます。これにより、Tableau Server ログイン アカウントに基づいて、パブリッシュされたビューに表示するデータを制御できます。

次の表は、上記のオプションのいくつかの依存関係の概要を示しています。

データベース接続のオプション		*データ セキュリティの質問*		
データベース ログイン アカウントで使用する認証方法	**認証モード**	**Tableau Server ユーザーごとにデータ セキュリティは設定可能ですか?**	**ユーザー フィルターが各ユーザーに表示するデータを制限する唯一の方法ですか?**	**ユーザー間で Web キャッシュは共有されますか?**
Active Directory 認証資格情報 (Windows 認証)	Kerberos サービス アカウント	いいえ	はい	はい
	サーバーの Kerberos サービス アカウント経由での偽装	はい	いいえ*	いいえ
	ビューアーが認証資格情報を入力	はい	いいえ*	いいえ
ユーザー名とパスワード	*ユーザーにメッセージを表示*: Viewer (ビューアー)がビューをクリックすると、データベースの認証資格情報の入力を求められます。認証資格情報は保存できます。	はい	いいえ	いいえ
	埋め込み認証資格情報: ワークブックまたはデータ ソースのパブリッシャーは、データベースの	いいえ	はい	はい

データベース接続のオプション		*データセキュリティの質問*		
データベースログインアカウントで使用する認証方法	**認証モード**	**Tableau Server ユーザーごとにデータセキュリティは設定可能ですか?**	**ユーザーフィルターが各ユーザーに表示するデータを制限する唯一の方法ですか?**	**ユーザー間で Web キャッシュは共有されますか?**
	認証資格情報を埋め込むことができます。			
	埋め込みパスワード経由での偽装: 偽装パーミッションを持つデータベースの認証資格情報が埋め込まれます。	はい	いいえ*	いいえ

* 予期しない結果が発生する可能性があるため、ユーザーフィルターと共にこの認証モードを使用しないことをお勧めします。

ユーザーフィルター、埋め込み認証資格情報オプション、および偽装モードには、同じような効果があります。つまり、ユーザーがビューをクリックしたときにデータベースの認証資格情報の入力は求められず、そのユーザーに関係するデータのみが表示されます。ただし、ユーザーフィルターは作成者によってワークブックに適用され、偽装認証モードはデータベース自体の管理者によって定義されるセキュリティポリシーに依存します。

Tableau の行レベルのセキュリティオプションの概要

データのフィルターリングを、データを必要としているユーザーに基づいて行いたい場合があります。例えば、次のような場合です。

- 地域の営業担当者に、自分の地域の売上高のみを表示させたい。
- 営業マネージャーに、自分の部下の営業担当者の統計のみを表示させたい。
- 生徒に、自分の成績に基づいたビジュアライゼーションのみを表示させたい。

データをこのようにフィルターリングするアプローチは、行レベルのセキュリティ(RLS) と呼ばれます。行レベルのセキュリティを実現する方法は Tableau の内部と外部で複数あり、それぞれに長所と短所があります。

手動でユーザーフィルターを作成し、ユーザーを値にマッピングする

Tableau で行レベルのセキュリティを実現する最も簡単な方法は、ユーザーを手動で値にマッピングするユーザーフィルターを使用することです。たとえば、「Alice」という名前のユーザーを値「東部」に手動でマッピングして、「地域」列が「東部」であるデータソースの行のみが表示されるようにすることができます。

この方法は便利ですが、メンテナンスに手がかかり、セキュリティに注意を払う必要があります。ワークブックごとに行う必要があり、ユーザーベースが変更されるたびにフィルターを更新してデータソースをパブリッシュし直す必要があります。このタイプのユーザーフィルターを使用してアセットをパブリッシュする場合、ユーザーがアセットを保存またはダウンロードしてフィルターを削除し、その結果すべてのデータにアクセスできないようにパーミッションを設定する必要があります。

詳細については、Tableau Desktop および Web オーサリングヘルプの「手動でユーザーフィルターを作成し、ユーザーを値にマッピングする」を参照してください。

データのセキュリティフィールドを使用して動的ユーザーフィルターを作成する

この方法を使用して、ユーザーをデータ値へマッピングするプロセスを自動化する計算フィールドを作成します。この方法では、フィルターに使用するセキュリティ情報が参照元データに含まれている必要があります。たとえば、データソースの計算フィールド、USERNAME() 関数、および「マネージャー」列を使用して、ビューを必要としているユーザーがマネージャーであるかどうかを判断し、それに応じてビューのデータを調整します。

フィルターリングはデータレベルで定義され、計算フィールドで自動化されるため、この方法はユーザーを手動でデータ値にマッピングするよりも安全です。このタイプのユーザーフィルターを使用してアセットをパブリッシュする場合、ユーザーがアセットを保存またはダウンロードしてフィルターを削除し、その結果すべてのデータにアクセスできないようにパーミッションを設定する必要があります。

詳細については、Tableau Desktop および Web オーサリング ヘルプの「データのセキュリティフィールドを使用して動的フィルターを作成する」を参照してください。

データポリシーを使用する

Tableau 2021.4 以降、Tableau Server または Tableau Cloud でデータ管理が有効になっている場合、Creator ライセンスを持つユーザーは、仮想接続のデータポリシーを通じて行レベルのセキュリティを実装できるようになりました。仮想接続は一元化され再利用できるため、その接続を使用するすべてのコンテンツにわたって、各接続の行レベルのセキュリティを 1 か所で安全かつ確実に管理できます。

Tableau の行レベルのセキュリティに関する前述のソリューションとは異なり、この方法では、作成者がワークブックやデータソースのパーミッションを適切に設定しなかったために情報が漏えいするリスクはありません。すべてのクエリに対してサーバー上でポリシーが適用されるためです。

仮想接続のデータポリシーによる行レベルのセキュリティは、行レベルのセキュリティの他のソリューションの欠点に対処するために開発されました。このソリューションを選択できる場合は、ほとんどの状況でこのソリューションを使用することをお勧めします。

仮想接続でデータポリシーを使用する行レベルのセキュリティの詳細については、仮想接続とデータポリシーについてを参照してください。

データベース内の既存の RLS を使用する

多くのデータソースには、RLS のメカニズムが組み込まれています。組織がすでにデータソース内の行レベルのセキュリティの構築に力を入れている場合は、既存の RLS を利用できる場合があります。

組み込み RLS モデルの実装は、Tableau を念頭においた構築に比べて、必ずしも簡単であったり優れたりするとは限りません。これらの手法は、一般に、組織が既にこれらのテクノロジに投資しており、投資を活用する場合に利用されます。

組み込まれている RLS を使用する主な利点は、管理者がデータセキュリティポリシーを 1 か所、つまりデータベースに実装して制御できることです。

詳細については、データベースの行レベル セキュリティを参照してください。

行レベルのセキュリティオプションの比較

RLS オプション	役に立つ状況	長所	短所
手動ユーザーフィルター	• 概念実証を行っている、または、ユーザーフィルターリング機能をテストしている • 変わることのないユーザーグループで使用する静的なワークブックを作成している • パーミッションが正しく設定されていないことによる情報セキュリティのリスクを理解している	• 小規模でシンプル • わかりやすいマッピング • テストに適している	• メンテナンスに手がかかる • ユーザーの所属が変わるとフィルターを更新し、パブリッシュし直す必要がある • ユーザーがフィルターリングされていないデータを見ることができないように、パーミッションを設定する必要がある • すべてのワークブックに複製する必要がある
動的ユーザーフィルター	• データ管理 ライセンスがない • フィルターリングに使用できる情報がデータに含まれている • パーミッションが正しく設定されていないことによる情報セキュリティのリスクを理解している	• セットアップが比較的簡単	• ユーザーがフィルターリングされていないデータを見ることができないように、パーミッションを設定する必要がある • すべてのワークブックまたはデータソースに複製する必要がある
データポリシー	• データ管理 ライセンスがある	• 一元化 • 安全	• データ管理 ライセンスが必要

	• フィルターリングに使用できる情報がデータに含まれている • データセキュリティの容易さは重要な問題である	• メンテナンスが容易 • セキュリティ責任と分析責任を分離できる	
データベース内の RLS	• データベース内に組み込まれた RLS がすでにある • 抽出を使用していない	• 組織のデータベースにすでに組み込まれている可能性がある • Tableau 以外のデータベースクライアントにもポリシーを適用できる	• ライブクエリを使用する必要がある • 制限事項や必須要件がある場合があり、IT チームがそれらを識別できる

どの行レベルのセキュリティオプションを使用するか?

あなたの組織では、このプロジェクトにも利用できる、推奨の RLS ソリューションがデータベース内にありますか?	→ はい →	データベースの行レベル セキュリティを参照してください
↓ いいえ ↓		
データ管理 ライセンスがありますか?	→ はい →	仮想接続とデータポリシーについてを参照してください
↓ いいえ ↓		
概念実証、ユーザーフィルターの基本的なテスト、またはユーザーが変わらない静的なワークブックですか?	→ はい →	「手動でユーザー フィルターを使用する」を参照してください
↓ いいえ ↓		
「動的ユーザー フィルターを使用する」を参照してください		

データ ソースとワークブックの RLS のベストプラクティス

Tableau の行 レベル セキュリティ(RLS) は、特定のユーザーがワークブックに表示できるデータの行数を制限します。これは、コンテンツや機能へのアクセスを制御する Tableau パーミッションとは異なります。たとえば、パーミッションはユーザーがワークブックにコメントしたり、ワークブックを編集したりできるかどうかを制御しますが、行 レベルのセキュリティでは、同じダッシュボードを表示している 2 人のユーザーが、それぞれ表示可能なデータのみを表示することができるようにします。

Tableau ではさまざまな方法で RLS を実装することができます。たとえば、RLS をデータ ソース レベルまたはワークブック レベルで設定したり、データ ポリシーを使用した仮想接続を使用して接続レベルで RLS を設定 (データ管理 が必要です) したりできます。代替案の詳細については、「Tableau の行 レベルのセキュリティ オプションの概要」参照してください。

注: このトピックでは、データ ソースとワークブックの RLS のベストプラクティスに焦点を当てています。注: このトピックで概説する概念の詳細な例については、ブログ *Tableau and Behold* にあるホワイトペーパー「資格 テーブルによる行 レベル セキュリティのベストプラクティス」と「Tableau の行 レベル セキュリティ用にデータベースを設定する方法」を参照してください。

RLS のワークフロー

ライブ接続および複数テーブル抽出の場合、基本的な RLS ワークフローは次のとおりです。

1. ユーザーは、Tableau Server または Tableau Cloud にログインすることで識別されます。
    - これには、ユーザーごとの個別のユーザー名とセキュリティで保護されたシングル サインオン (SSO) が必要です。
    - Active Directory、LDAP、または Tableau REST API を使用してユーザー名を同期し、パーミッションを確立できます。
2. ユーザーのデータ資格のセットは、すべての可能なデータ資格から取得されます。
    - これには、資格を Tableau ユーザー名にリンクできるデータ構造が必要です。
3. データは、そのユーザーの資格によってフィルターリングされます。
    - 多くの場合、計算フィールドでユーザー関数を使用する必要があります。

4. パブリッシュ済で、フィルター処理されたデータは、コンテンツの構築に使用されます。
    - データ ソース フィルターでパブリッシュされた (埋め込みではない) データ ソースを使用すると、ワークブックをダウンロードまたは Web 編集して RLS を変更できなくなります。

結合、計算フィールド、およびフィルターの設定方法は、データの構造とユーザーの管理方法によって異なります。

資格テーブル

データをフィルター処理できる属性の一意の組み合わせは、資格です。通常、資格自体を指定し、それらの資格をユーザーまたはユーザー ロールにマップするための個別の表があります。結合は負荷の高い操作であるため、パフォーマンスの観点からは非正規化をお勧めします。

ユーザーまたはロールにマップされた資格で構成される資格ビューは、データと結合されます。次に、ユーザー ベースのデータ ソース フィルターが適用され、関連するユーザーの権利 (したがって適切なデータ行) のみを取り込む WHERE 句として機能します。(クエリの最適化では、データの重複を最小限に抑えるためにクエリが処理されるときに、結合する前にフィルター処理が行われるようにする必要があります。詳細については、操作のパフォーマンスと処理順序を参照してください。)

資格表モデル

一般に、資格を表すためのモデルは 2 つあります。

最も深い粒度への完全なマッピング

- 資格は、すべての列に対して完全に定義されます。
- マッピング テーブルには、ユーザーが持つ可能性のある資格ごとに 1 つの行があります。
- このモデルでは、必要な結合句が少なくなります。

スパース資格

- 資格は階層のすべてのレベルに対して定義され、NULL は "すべて" 状態を表すために使用されます。
- 資格階層内の特定のレベルのマッピング表には 1 つの行があり、階層内の上位レベルのユーザーの資格行の数が大幅に減少します。
- このモデルには、より複雑な結合とフィルターが必要です。

ユーザーとロール

権利の組み合わせは、一般にロールとして表され、多対多マッピング表のユーザーにリンクされます。これにより、ロールとその資格の記録を維持しながら、ロールのユーザーを簡単に変更または削除できます。

または、ロール表の結合ではなく、ユーザーを資格に直接割り当てる多対多マッピング表を作成することもできます。テーブル内の値をより直接的に管理する必要がありますが、結合は削除されます。

注: ロールまたは資格に関連付けられたユーザー値は、Tableau Desktop のユーザー機能を利用するために、Tableau サイトのユーザー名またはフル ネームと一致する必要があります。

結合

資格を表すために使用されるモデルに関係なく、すべての資格とマッピング表を 1 つの非正規化された資格ビューに結合することをお勧めします。最初は、資格の"ブローアップ"(非常に重複した)バージョンを引き起こしますが、ユーザーのデータソースフィルターはそれを減らします。抽出の使用を計画している場合も、このビューが必要になります。

最も粒度の深い方法では、すべてが階層的である場合にパフォーマンス上の利点を得ることができます。階層の最も深いレベルで単一結合をすればよいだけです。これは、最下位レベルのすべての属性が異なる場合にのみ機能します。重複の可能性がある場合 (たとえば、複数のリージョンの中央サブリージョン)、個別のキー値の効果を得るには、すべての列で結合する必要があります。

実際の詳細とそのパフォーマンス特性は、データシステムによって異なっており、テストが必要です。たとえば、1 つのキーを使用すると、結合が 1 つの列に対してのみ実行されるため、パフォーマンスが向上する可能性がありますが、すべての列に正しくインデックスを付けると、他の要因が考慮された場合に同等のパフォーマンスが得られる可能性があります。

行レベルのセキュリティを実装する

最も深い粒度

マッピングされた資格の非正規化ビューが作成されると、ビューとTableau データ接続ダイアログのデータの間に内部結合が設定されます。データは従来のスター スキーマに残すことができます。または、ディメンション表とファクト表を2 つのビューにマテリアライズすることもできます。複数表の抽出では、結合に合わせて抽出表が作成されるため、2 つのビューを作成すると、結果の抽出が簡略化されます。SQL は次の基本的なパターンに従います。

```
SELECT *
FROM data d INNER JOIN entitlements e ON
d.attribute_a = e.attribute_a AND
d.attribute_b = e.attribute_b AND ...
WHERE e.username = USERNAME()
```

スパース資格

資格がスパース資格モデルに近い場合、NULL 値が原因で、データを資格に結合するカスタム SQL はもう少し複雑になります。概念的には、次のようになります。

```
SELECT *
FROM data d
INNER JOIN entitlements e ON
(e.region_id = d.region_id OR ISNULL(e.region_id) AND
(e.sub_region_id = d.sub_region_id OR ISNULL(e.sub_region_id) AND
(e.country_id = d.country_id OR ISNULL(e.country_id)
```

カスタム SQL を使用しなくても、Tableau Desktop のクロス結合と追加のフィルターを使用してこれを行うことができます。整数 1 で構成される結合ダイアログの両側に結合計算を作成し、等しく設定します。これにより、データ表のすべての行が、資格表のすべての行に結合されます。

次に、階層内のレベルを考慮する計算 (または個別の計算) が必要です。たとえば、次の形式に従って複数の計算を行うことができる場合があります。`[region_id] = [region_id (Entitlements View)] OR ISNULL([region_id (Entitlements View)]`

または、すべてのレベルを 1 つにまとめて計算することもできます。

```
([region_id] = [region_id (Entitlements View)] OR ISNULL([region_id
(Entitlements View)])
AND
([sub_region_id] = [sub_region_id (Entitlements View)] OR ISNULL
([sub_region_id (Entitlements View)])
AND
([country_id] = [country_id (Entitlements View)] OR ISNULL
([country_id (Entitlements View)])
```

ISNULL 関数は、任意の資格列を他のフィールドのすべての項目と一致します。RLS と同様に、これらの計算はデータ ソース フィルターとして追加する必要があります。

データ ソース フィルター

どちらの方法でも、資格がデータと正しく結合されたら、特定のユーザーのデータを制限するようにフィルターを設定する必要があります。計算フィールドは、ユーザー関数を使用して作成する必要があります。たとえば、[ユーザー名] フィールドに表示されるユーザーが Tableau サイトにログインしているユーザーのユーザー名と同じかどうかの単純なブール比較を行います。`[Username] = USERNAME()`

この計算は、データ ソース フィルターとして使用する必要があります (TRUE が選択されている場合)。

データ ソースが埋め込まれており、ユーザーがブックを Web 編集またはダウンロードするパーミッションを持っている場合、RLS は存在しません。Tableau データ ソースは、ワークブックに埋め込まれたままになるのではなく、個別にパブリッシュする必要があります。

最も深い粒度を持つすべてのアクセス

また、組織内に 2 つのアクセス レベル ("すべてのアクセス") を表示できるユーザーと、合理的に定義できる権利のサブセットを持つユーザーという2 つのアクセス レベルがある一般的なシナリオもあります。これは、データをホストする組織がすべてを表示できるが、各クライアントは独自のデータしか表示できない、組み込みアプリケーションで最もよく見られます。この場合、他のすべてのユーザーの最も深い細分性結合を維持しながら、"すべてのアクセス" ユーザーの完全なデータを返す方法が必要です。

この手法では、Tableau グループを使用して、結合条件の計算を使用してオーバーライドを作成します。

1. すべてのデータを表示するユーザーのグループを作成する (ここでは [すべてのアクセス] と呼ばれます)
2. ファクトビューから、2 つの結合条件を持つ左結合を作成します。
    - 最初の結合条件は、最も深い粒度レベルを表す列に置く必要があります。
    - 2 番目の結合条件は、次の 2 つの計算にする必要があります。
        - 左側 (ファクトビュー) で、計算に対して次のように入力します。`True`
        - 右側 (資格ビュー) では、計算は次のようになります。`IF ISMEMBEROF ('All Access') THEN False ELSE True END`
3. シートで、次のように構造化された計算を作成します。`[Username] = USERNAME() OR ISMEMBEROF(['All Access'] ([Entitlements View)])`
4. ユーザー名の計算でデータソースフィルターを作成する

ユーザーが All Access グループのメンバーである場合、結合は `True = False` の左結合になります。つまり、資格ビューには一致するものがまったく存在しないため、権利ビューの列に対してファクトビュー全体が UL と共に返されます (重複ゼロ)。ユーザーが All Access グループに参加していない場合、`True = True` 結合条件は何も変更されず、結合は期待どおりに機能します。

データソースフィルターとして使用されるユーザー計算は、グループの上書きが機能している場合はすべての行に当てはまります。または、階層内にあるユーザーの最も深い粒度にまでフィルターされます。

操作のパフォーマンスと処理順序

ビジュアライゼーションが Tableau Desktop、Tableau Server、または Tableau Cloud で表示されると、Tableau は最適化されたクエリを RDBMS に送信し、クエリを処理して Tableau に結果を送信し、結果データを使用してビジュアライゼーションをレンダリングします。結合、計算、およびフィルターが実行される操作の順序は、クエリオプティマイザとクエリの実行方法によって異なります。

ライブ接続

Tableau でデータソースへのライブ接続を使用する場合、クエリ実行のパフォーマンスは、受信 SQL をデータ取得の効率的なプランに変換するクエリオプティマイザに依存します。

クエリを処理するには、次の 2 つの方法があります。

1. ユーザーの資格行をフィルター処理し、ファクト表に結合する
2. 資格をファクト表に結合し、ユーザーの行にフィルターを適用します。

理想的な状況では、クエリオプティマイザは、データベースが *フィルター処理してから結合* することによってクエリを処理することを保証します。ユーザーがすべてに対して資格を持っている場合、処理される最大行数はデータテーブル内の行数になります。

データベースが *結合してからフィルター処理して*クエリを処理する場合、データが重複している可能性があります。処理される行の最大数は、データ表の各行にその特定の行の回数を表示する権利を持つユーザーの数になります。

この 2 番目のシナリオが発生するかどうかは明らかです: クエリの完了に時間がかかる、エラーが発生する、またはデータベースのパフォーマンスの問題を示しています。データ量の合計が指数関数的に拡大し、バックエンドでシステムの負担が大きくなる可能性があります。

抽出

Tableau のデータソースがライブ接続の場合、Tableau は特定の viz またはダッシュボードを RDBMS にレンダリングするために必要なすべてのクエリを送信します。データソースが抽出の場合、基になるデータソースからデータをクエリするプロセスは、抽出の作成と更新時にのみ行われます。ビジュアライゼーションに対するすべての個々のクエリは、抽出ファイルから抽出エンジンによって応答されます。

単一の表抽出を作成する場合も、同じ順序の操作の問題が存在します。ただし、"ブローアップ" は、基になるデータソースと結果の抽出自体の両方で発生します。

抽出に関する考慮事項

Tableau 2018.3 以降、データエンジンは複数テーブルの抽出を作成でき、RLS は上記のように実装できます。複数表抽出を使用すると、結合を具体化しないことで、多対多リレーションシップを持つ抽出の生成にかかる時間が短縮されます。

抽出は、*データオブジェクト*と*資格オブジェクト*を使用して構築する必要があります。これは抽出の中で最も単純なストレージであり、最高のパフォーマンスを得ることができます。

- *データオブジェクト*は、ファクト表と必要なディメンション表の非正規化された組み合わせを表す表、ビュー、またはカスタム SQL クエリです。

- *資格オブジェクト*は、最も細かいレベルでデータをフィルターリングするために必要な資格の非正規化された表、ビュー、またはカスタム SQL クエリです。
  - Tableau Server または Tableau Cloud の正確なユーザー名と一致するユーザー名の列
  - データオブジェクトに対する最も詳細な権利の各行

この形式は、上記の最も深い粒度の方法でレイアウトされています。複数表抽出では同じ方法が使用されますが、2 つのデータオブジェクトのみが結合され、フィールド固有のフィルター処理が既にオブジェクト内に適用されていることに注意してください。

複数表抽出で抽出フィルターが無効になっているため、データソース内の接続先のビューまたは表でフィルター処理するか、[Tableau データ接続] ダイアログでカスタム SQL オブジェクトでフィルターを定義できます。

> **注**: ライブ接続の場合と同様に、データソースが埋め込まれており、ユーザーがブックを Web 編集またはダウンロードするパーミッションを持っている場合、RLS は存在しません。抽出は、ワークブックに埋め込まれたままになるのではなく、個別にパブリッシュする必要があります。

単一の表の抽出

次の方法は、2018.3 より前のバージョンの Tableau を使用している場合にのみ推奨されます。使用可能な場合、複数表抽出が好ましいです。

単一の表の抽出は、Tableau データソースの構築時に作成した結合を具体化し、1 つのクエリを通じてすべてを 1 つの表として格納します。この非正規化は、複数の資格またはユーザーに割り当てられたすべての行が多対多リレーションシップの結果として重複するため、大量のデータ重複を引き起こすリスクを伴います。

この重複を防ぐには、次の手順を実行します。

1. その資格のユーザー名を含むセキュリティユーザーフィールドを作成する
   - たとえば、値は "bhowell|Mosterheld|rdugger"
2. Tableau 内で CONTAINS() 関数を使用して個々のユーザーを正しく識別する
   - たとえば、`CONTAINS([Security Users Field], USERNAME())`

この方法には明らかにいくつかの注意事項があります。行内の資格から、SQL を使用して正しく区切られた単一の列に移動する必要があり、その列には非常に多くの文字しか含めることができない必要があります。部分一致が問題になる可能性があり、ID 自体では決して有効でない区切り記号を使用する必要があります。Tableau データエンジン内ではパフォーマンスが向上しますが、文字列の計算はほとんどのデータベースで非常に低速になります。これにより、ライブ接続に切り替える機能が制限されます。

または、「ロール」レベルまたは資格レベルごとに異なる抽出を使用して、その個人またはレベルに適したデータのみが抽出に含まれるようにすることもできますが、これには適切なパーミッションを付与してテンプレートを活用するプロセスが必要になります。通常は API を介して Tableau Server 内でパブリケーションを行います。

組み込みの行レベル セキュリティをデータベースで使用する

多くのデータベースには、RLS のメカニズムが組み込まれています。組織が既にデータベースの行レベル セキュリティの構築に力を入れている場合は、既存の RLS を利用できる場合があります。組み込みの RLS モデルを実装する方が Tableau を念頭に置いて構築するよりも容易であるとは限りません。なぜなら、これらの技術は一般的に、組織がすでにこうした技術に投資していて、その投資を活用しようとするときに利用できるものだからです。組み込まれている RLS を使用する主な利点は、管理者がデータ セキュリティ ポリシーを 1 か所、つまりデータベースに実装して制御できることです。詳細については、「データベースの行レベル セキュリティ」を参照してください。

データベースの行レベル セキュリティ

組織で既にデータベースの行レベル セキュリティ(RLS) の構築に力を入れている場合は、次のいずれかの方法を使用して既存の RLS を利用できる可能性があります。データベースのセキュリティ モデルを活用するには、ライブ接続が必要です。さらに、これらの方法は Tableau Cloud では使用できない可能性があります。Tableau Cloud の Tableau ユーザー名は一意のメール アドレスで、通常はデータベース側のユーザー ID ではありません。

組み込み RLS モデルの実装を Tableau での構築に比べて簡単にしたり改善したりする必要はありません。通常、これらの方法は組織が既にこれらのテクノロジに投資し、それを活用しようとしている場合に利用されます。

> **注:** Tableau で行レベル セキュリティを実装するために使用できる代替方法については、Tableau の行レベルのセキュリティオプションの概要を参照してください。

偽装 (Microsoft SQL Server)

Microsoft SQL Server (およびいくつかの関連システム) では、Security Junction Table または DBA で構築されたビューのいずれかを使用して、RLS フィルターが組む込まれたビューのみにデータベースのユーザーがアクセスできるように設定できます。Tableau では、"偽装" と呼ばれる概念を使用してこれを活用できます。

MS SQL Server 接続を含む Tableau データソースを Tableau Server にパブリッシュする場合に、偽装を使用できる認証オプションが 2 つあります。表示されるメニューは、ネットワーク認証を使用して SQL Server にログインしたか、ユーザー名とパスワードの認証資格情報を入力したかによって異なります。

Tableau Server のパブリッシュされたデータソースにアクセスできるすべてのユーザーに対して RLS フィルターリングを有効にするには、AD 実行アカウントまたは SQL サーバーの埋め込み認証資格情報に、ダッシュボードまたはデータソースにアクセスするデータベース内のすべての Tableau ユーザーに対する EXECUTE AS パーミッションが含まれている必要があります。すべての Tableau ユーザーがユーザーとしてデータベース サーバーに存在し、接続先 (および RLS の適用先) のビューに対する SELECT 権限を持っている必要があります。要件の包括的なリストについては、偽装の要件を参照してください。

Kerberos と制約付き委任

Kerberos を使用する Tableau Server 内の制約付き委任は、Tableau Server がワークブックのビュー、またはビューアーに代わってクエリを実行するビューの Kerberos 認証資格情報を使用できるという点で偽装と同様の動作を行うため、データベースに RLS が設定されている場合、ワークブックのビューアーには自分のデータのみが表示されます。

Kerberos 委任がサポートされているデータベースの包括的なリストを表示するには、Kerberos 委任の有効化を参照してください。Active Directory は必須です。Tableau Server がインストールさ

れているコンピューターを Active Directory のドメインに結合する必要があります。データソースをパブリッシュするときに指定する認証方法は、**ビューアー認証資格情報**にする必要があります。

Microsoft Analysis Services を使用する場合は、RLS に Kerberos を使用できます。

OLAP キューブ

Tableau の OLAP キューブ接続はデータソース フィルターと同等ではありません。この接続は Tableau のエンタイトルメント テーブルベースの RLS メソッド、または USERNAME() 関数へのアクセスで必要となります。こうした理由から、Kerberos と制約付き委任は OLAP データベースを使用する RLS に対して推奨される方法となり、これにより Tableau は OLAP サーバー側で既に実装されているユーザー フィルターを利用できるようになります。

ダッシュボードを表示するユーザーがドメインに含まれていない場合は、ユーザー フィルターを手動で作成する方法を実行できます。ただし、生成されたユーザー フィルター セットはデータソース フィルターとして追加できず、代わりにフィルター シェルフに追加されるため、この方法を使用してパブリッシュされたビューには Web 編集機能とワークブックのダウンロード機能を使用できないようにすることが重要です。

SAML 委任と SAP HANA

Tableau Server が SAP HANA SSO の構成を使用してシングル サインオンを使用できるように構成されている場合、ビューアーの認証資格情報を使用してクエリをユーザーとして実行し、ユーザー レベルで適用されるすべてのセキュリティ内で動作するようになります。データソースをパブリッシュするときに指定する認証方法は、**ビューアー認証資格情報**にする必要があります。

ユーザー固有のセッションを強制する初期 SQL (Oracle VPD)

初期 SQL では、セッション中に使用する一時テーブルまたはカスタム データ環境を設定するために一時テーブルを設定する目的で、接続がデータベースに作成されている場合に実行される SQL コマンドを指定できます。

Oracle VPD では、特定のストアドプロシージャまたは関数を実行してユーザーに固有のセッションを設定し、Tableau ユーザーのユーザー名と一致するようにデータベース接続のコンテキストを設定できます。

```
begin
DBMS_SESSION.SET_IDENTIFIER([TableauServerUser]);
end;
```

RLS でこれを偽装と同様に使用する場合は同じ高度な要件が当てはまります。DBA は、VPD および関連するすべてのユーザーがデータベースに存在するように設定する必要があります。

MS SQL Server で、EXECUTE AS コマンドを強制できます (ただし、これは Tableau が偽装で既に行っている内容と似ています)。

```
EXECUTE AS USER = [TableauServerUser] WITH NO REVERT;
```

注: データソースが埋め込まれていて、ユーザーが Web 編集またはワークブックのダウンロードに対するパーミッションを持っている場合、RLS を強制する初期 SQL は簡単に削除できるため、RLS は存在しません。データソースはワークブックに埋め込まれたままにするのではなく、個別にパブリッシュする必要があります。

行レベル セキュリティ メソッドの比較マトリックス

メソッド	役に立つ状況	長所	短所
エンタイトルメントテーブル (推奨)	• データベースにエンタイトルメントの既存の概念がある場合 • 組織が初めて行レベル セキュリティを設定しているとき	• テスト、更新、保守、拡張が簡単 • バージョン 2018.3 以降のライブ接続と抽出の両方で動作する	• エンタイトルメントテーブルの作成と維持が必要 • パフォーマンスを最適化するために適切なキーの選択と作成が必要になる場合がある
CONTAINS() と抽出	• バージョン 2018.3 以前の抽出で RLS を実装する場合	• 抽出の効率を活用できる	• すべてのユーザーを 1 つの列にマッピングする必要がある • 文字列計算のた

			め、ライブ接続に切り替えるのが困難
偽装	• データにアクセスするすべてのユーザーが SQL Server のユーザーとして存在する場合 (通常は、内部展開)	• セキュリティが 1 か所 (データベース) で処理および維持される	• ビューにアクセスするすべてのユーザーが SQL Server 内にユーザーとして存在している必要がある • Microsoft SQL Server でのみ動作する
Kerberos	• Kerberos 委任に必要なすべてのデータベースが設定され、RLS がデータベース上 (通常は内部展開) に設定されている場合	• データベースのアクセス ログにビューアーの名前が表示される • セキュリティがデータベースで処理および維持される	• Tableau が、LDAP - Active Directory を使用するように構成されている必要がある • Tableau Server を AD ドメインに結合する必要がある • すべてのユーザーが AD ドメイン内に存在している必要がある
初期 SQL	• データベースで初期 SQL をサポートし、RLS がデータベース側で設定されている場合	• 読み込み時に Tableau パラメーターを渡すことができる • 他のユーザーと共有できない専	• 初期 SQL をサポートしていないデータベースもある • キャッシュ共有が制限されている

	用の接続 • クエリをユーザーとして実行するため、ユーザーがデータベース内に存在している必要がある	ためにパフォーマンスに影響を及ぼす可能性がある

サーバー シークレットの管理

Tableau Server は一般的に、内部通信の保護、他のアプリケーションやオペレーティング システムとの通信、またはクライアントとの安全な通信の提供など、さまざまな機能の実行に使用する多数のシークレットを保存する必要があります。このコンテキストでは、*シークレット*という用語は、1 つのエンティティを別のエンティティに対して認証するために使用される、パスワード、トークン、またはその他の文字列を表します。

Tableau Server で実行する必要があるシークレットには 2 つのカテゴリーがあります。それらは、シークレットの生成方法によって異なります。

- **管理者によって生成されるシークレット**。これらには、認証資格情報や、Tableau Server で使用される実行ユーザー アカウントや SMTP 認証資格情報用の関連付けられたシークレットが含まれます。
- **システムのさまざまなプロセスによって自動生成されるシークレット**。たとえば、クラスター コントローラーとZooKeeper プロセスの間の通信を保護するためには、シークレットが必要です。Postgres と通信する各サービスやプログラミング ユーザーで、異なるパスワードが多数必要となります。

ほとんどのシークレットは保存中に暗号化されています。シークレットが必要な場合は、実行時に復号化されます。

このトピックでは、シークレット ストレージの仕組みと、Tableau Server でシークレットのストレージを正しく管理するため、何をする必要があるかについて説明します。

シークレット ストレージの仕組みについて理解する

インストール時、Tableau Server は Java キーストアでマスター キーを生成して保存します。マスター キーは、システム全体で使用される構成暗号化キーの暗号化に使用されます。

新しいシークレットが作成または更新されると、シークレットは構成暗号化キーを使用して暗号化されます。次に、暗号化された値はサーバー上の YAML ファイル内にある対応する構成パラメーターを使用して保存されます。暗号化された値を保持するパラメーターは `ENC(<encrypted string>)` 形式を使用します。ここで、`<encrypted string>` には、Base64 エンコードの暗号化された文字列を使用します。

実行時、指定されたシークレットへのアクセスが必要になると、暗号化された値はメモリに読み取られ、構成暗号化キーを使用して複合されます。

構成変更中にシークレットが入力されるようにする変更を保留中の場合は、トランザクション全体が暗号化されます。このシナリオでは、シークレットを入力してから保留中の変更を保存し、シークレットが暗号化された SSL を介して調整サービスに送信されます。調整サービスはシークレットを暗号化し、保留中の変更が適用されるまで保存します。変更が適用されると、暗号化されたままの状態のシークレットが現在の構成バージョンに昇格します。

Tableau Server は、GCM モードで 256 ビット AES を使用してシークレットを暗号化します。セキュア ストレージに使用されるキーは、リポジトリに保存される前に埋め込みデータベース認証資格情報の暗号化に使用されるアセット キーとは異なります。

マスター キーへのアクセス権を持っているのは誰か

既定のインストールでは、Tableau Server の Java キーストアは、`/var/opt/tableau/tableau_server/data/tabsvc/config` 内のノード上の各サービスの `/tabsvc/keystores` フォルダーに複製されます。

例は次のとおりです。

`/var/opt/tableau/tableau_server/data/tabsvc/config/tabadminagent_<version_number>/tabsvc/keystores/tableauserver.jks`。

カスタム インストール ディレクトリを使用する場合、キーストア ファイルは次の場所にあります。

```
<install directory>/tableau_server/data/tabsvc/config/<service name_
#.version_number>/tabsvc/keystores
```

既定では、次のユーザーおよびグループにこのディレクトリへのアクセス権があります。

- root
- tableau (ユーザー)
- 'tableau' グループのメンバー

構成情報のインポートおよびエクスポート

Tableau Services Manager には、tsm settings export を使用して構成情報をインポートおよびエクスポートする機能が導入されています。

> **注**:このバージョンの Tableau Server では、バックアップからの構成情報の復元はサポートされていません。代わりに、構成のエクスポートおよびインポート コマンドを使用して構成情報をバックアップおよび復元することをお勧めします。

構成シークレットはディスクに内部保存されるときに暗号化されますが、構成がファイルにエクスポートされるときには、シークレットがプレーン テキスト形式でファイルに書き込まれます。管理者はこのファイルを保護する責任があります。次のようなさまざまなオプションがあります。

- 暗号化されたファイル システムにファイルを書き込む。
- ファイル システム パーミッションによって特定のユーザーまたはグループのみに限定されるディレクトリにファイルを書き込む。
- 出力ファイルを暗号化する。

インポートおよびエクスポート操作におけるシークレットの保護

このセクションでは、バックアップ出力を PGP で暗号化する方法について説明します。この方法により、名前付きパイプを作成してファイル引数として指定し、それを gpg 暗号化の入力として使用します。ディスクにシークレットがプレーン テキストで書き込まれることがないという利点があります。gpg と秘密キーが利用できなければなりません。次のセクションに、この方法の例を示します。

このセクションでは、シークレットを別のコンピューターに保管するための処理方法の例について説明します。

詳細については、次の外部参照資料を参照してください。

- File encryption on the command line (コマンドラインでのファイルの暗号化)
- mkfifo(1) - Linux man page (mkfifo(1) - Linux man ページ)

例:暗号化とエクスポート

構成をエクスポートするときのファイルの保護方法について例を示します。

```
mkfifo -m 600 /tmp/secure1 && (gpg --symmetric --batch --yes --
passphrase-file ~/.secrets/pgppassphrase.txt --cipher-algo AES256 -
-output encrypted.enc < /tmp/secure1 &) && tsm settings export -f
/tmp/secure1 && rm /tmp/secure1
```

この操作の詳細は、次のとおりです。

- 現在のユーザーのファイル読み取り/書き込みパーミッションによってアクセス権が制限される名前付きパイプを作成します。

  ```
  mkfifo -m 600 /tmp/secure1
  ```

- 別のプロセスを実行しながら、gpg を呼び出して名前付きパイプに送信されたデータを暗号化します。gpg ではデータを待機しながらブロックします。これにより、暗号化されたデータを含むファイルが作成されます。

  ```
  gpg --symmetric --batch --yes --passphrase-file
  ~/.secrets/pgppassphrase.txt --cipher-algo AES256 --output
  encrypted.enc < /tmp/secure1 &
  ```

- 名前付きパイプをファイル引数に指定して tsm を呼び出し、構成をエクスポートします。

  ```
  tsm settings export -f /tmp/secure1
  ```

- 名前付きパイプを削除します。

  ```
  rm /tmp/secure1
  ```

暗号化されたデータは "encrypted.enc" ファイル内にあります。

例:復号化とインポート

構成の復号化およびインポート方法について例を示します。

```
mkfifo -m 600 /tmp/secret2 && (gpg --decrypt --batch --yes --
passphrase-file ~/.secrets/pgppassphrase.txt encrypted.enc >
/tmp/secret2 &) && tsm settings import -f /tmp/secret2 && rm
/tmp/secret2
```

この操作の詳細は、次のとおりです。

- 現在のユーザーのファイル読み取り/書き込みパーミッションによってアクセス権が制限される名前付きパイプを作成します。

  ```
  mkfifo -m 600 /tmp/secure2
  ```

- 構成を復号化して名前付きパイプに送信します。別のプロセスのバックグラウンドで実行されます。読み込みを待機しながらブロックします。

  ```
  gpg --decrypt --batch --yes --passphrase-file
  ~/.secrets/pgppassphrase.txt encrypted.enc > /tmp/secret2 &
  ```

- 必要に応じてログインし、tsm configuration import コマンドを実行します。

  ```
  tsm settings import -f /tmp/secret2
  ```

- 名前付きパイプを削除します。

  ```
  rm /tmp/secure1
  ```

保留中の構成に、インポートされた構成が含まれています。

tsm pending-changes apply を実行して変更をコミットします。保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変

更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

クラスター ノード

Tableau Server クラスタに新規ノードを追加するときは、まずノード構成ファイルを生成する必要があります (tsm topology)。ノード構成ファイルには、構成シークレットの暗号化に使用されるマスター キーストア ファイルのコピーが含まれています。

重要: シークレットが含まれた構成ファイルをエクスポートする場合は、ノード構成ファイルを保護するためにさらなる対策を講じること強くお勧めします。

新規ノードに Tableau Server をインストールして構成するときに、ノード構成ファイルを `initialize-tsm` コマンドに指定する必要があります。上記の説明と同様の方法を使用して以前に暗号化したファイルのコンテンツを復号化し、名前付きパイプ経由で `initialize-tsm` コマンドに送信します。

シークレット ストレージ イベントのログ作成

シークレット ストレージに関する次のイベントが記録されます。

- 新しい暗号化キーの生成
- 暗号化キーが登録または変更されている
- 構成ファイルの新しい値の暗号化

ログ ファイルおよびそれらの保存場所の詳細については、ログ ファイルの操作を参照してください。

シークレットの管理

Tableau Server 管理者にとって、シークレット ストレージに関する最も重要なタスクは、シークレットを定期的に更新することです。場合によっては (サーバーのトラブルシューティングや監査など) パスワードの取得が必要となることもあります。

バージョンのアップグレード、バックアップや復元、またはクラスターへの新規ノードの追加 (上記) などの他の操作の場合、Tableau Server はシークレット ストレージや関連プロセスを自動的に管理します。

シークレットの更新

会社のセキュリティポリシーに従って、シークレットを定期的に更新する必要があります。

マスターキーと自動生成されたシークレットを更新するには、tsm security regenerate-internal-tokens を実行します。

パスワードの取得

トラブルシューティングやその他の操作でパスワードの取得が必要となる場合があります。たとえば、Tableau Server によって生成および暗号化された Postgres readonly ユーザー認証資格情報が必要となる場合があります。この場合、パスワードを取得して復号化する tsm コマンドを実行します。

パスワードを取得するには、コマンドプロンプトを開き、以下の表に記載されているいずれかのパラメーターに `tsm configuration get` コマンドを発行します。

たとえば、readonly Postgres ユーザーのパスワードを取得するには、次のコマンドを入力します。

```
tsm configuration get -k pgsql.readonly_password
```

コマンドは、パスワードをクリアテキストで返します。

```
$ tsm configuration get -k pgsql.readonly_password

password
```

Configuration Parameter	Description
clustercontroller.zookeeper.password	Password for cluster controller to connect to zookeeper.
indexandsearchserver.client.password	Password for logging into Index and Search Server.
indexandsearchserver.ssl.admin.cert.bytes	Admin certificate that is used for administrative access to the Index and Search Server. The admin certificate is used to generate the node certificate.

indexandsearchserver.ssl.admin.key.file_bytes	Certificate key for administrative access to the Index and Search Server.
indexandsearchserver.ssl.node.cert.bytes	Certificate that is used for Index and Search Server node-to-node communication.
indexandsearchserver.ssl.node.key.file_bytes	Certificate key that is used for Index and Search Server node-to-node communication.
indexandsearchserver.ssl.root.cert.bytes	Certificate that is used to sign the admin and node certificates . This certificate is used by TSM for health check and by NLP to connect to Index and Search Server.
indexandsearchserver.ssl.root.key.file_bytes	Certificate key for root certificate.
filestore.zookeeper.password	Password for filestore to connect to zookeeper.
hyper.connection.init_password	Password used to initialize the Hyper database for user `tableau_internal_user` and is then used for connecting to Hyper.
jdbc.password	Password for the rails Postgres user.
kms.persistent_store	A collection of master encryption keys (MEKs) used by the Key Management System.
maestro.rserve.password	Password for connecting to an external Rserve instance used by Tableau Prep Conductor for running flows that have nodes with R scripts.

maestro.tabpy.password	Password for connecting to an external TabPy (Python server) instance used by Tableau Prep Conductor for running flows that have nodes with Python scripts.
oauth.google.client_secret	Client secret of the Google Cloud Platform account.
oauth.quickbooks.consumer_secret	Consumer secret of the Intuit developer account.
oauth.salesforce.client_secret	Client secret of the Salesforce developer account.
pgsql.adminpassword	tblwgadmin Postgres ユーザーのパスワード。 **注**: 構成パラメーターは Tableau の構成ファイル (tabsvc.yml、workgroup.yml) に暗号化されますが、このパスワードは、SAML で使用される他のファイルにプレーン テキストで保存されます。
pgsql.readonly_password	Password for the readonly Postgres user.
pgsql.remote_password	Password for the tableau Postgres user.
redis.password	Redis のパスワード **注**: 構成パラメーターは Tableau の構成ファイル (tabsvc.yml、workgroup.yml) に暗号化されますが、

	構成は Redis アプリケーションが使用する redis.conf ファイルにプレーン テキストで引き続き保存されます。Redis は暗号化/セキュア パスワードをサポートしていません。
servercrashupload.proxy_server_password	Password for custom proxy server used to upload crash reports.
service.runas.password	Password of the Run As users. Stored temporarily.
ssl.cert.file_bytes	The content of one of the three SSL certificate files uploaded by the administrator. The certificate files are required to enable secure external connections to Tableau Server.
ssl.chain.file_bytes	The chain file(s) for the certificates uploaded by the administrator for external SSL.
ssl.key.file_bytes	Key file(s) for the certificates uploaded by the administrator for external SSL.
ssl.key.passphrase	Optional passphrase used to protect the external SSL key.
svcmonitor.notification.smtp.password	SMTP Server password supplied by the administrator through TabConfig.exe.
tabadminservice.password	Password for the service that allows server admins to download log files through the web interface.
vizportal.openid.client_secret	This is the password ("provider client

	secret") used for OpenID Connect SSO.
vizqlserver.external_proxy_password	Password used to authenticate to an external proxy.
wgserver.domain.password	Password used to bind to Active Directory.
wgserver.saml.key.passphrase	Passphrase used to access the PKCS#8 SAML key file.
zookeeper.tsm.password	Password that TSM uses to connect to Zookeeper coordination service

拡張機能のセキュリティ- 展開に関するベストプラクティス

これらのベストプラクティスは、IT 部門の役員と管理者、Tableau のサーバー管理者とサイト管理者、Tableau でダッシュボードとViz 拡張機能を使用することに関心のあるすべての人を対象としています。拡張機能により、それぞれのお客様の分析ニーズに合わせてTableau の使用方法を拡張およびカスタマイズする大きな機会がもたらされます。この拡張性には、データとビジネスのセキュリティに関する適切な考慮事項が常に含まれている必要があります。これらのベストプラクティスは、Tableau Desktop および Tableau Server または Tableau Cloud を使用しているユーザーが混在している展開を対象としています。

Tableau での拡張機能のセキュリティ

拡張機能は、ダッシュボードの機能を拡張したり、Tableau に新しいビジュアライゼーションを追加したりするWeb アプリケーションです。これらのソリューションは、Tableau で直接扱ったり、完全にサードパーティーのカスタム サービスとして提供したりできます。これらのサービスは、独自のネットワーク内でホストすることも、ネットワーク外の外部プロバイダーから提供することもできます。拡張機能はワークブック内の他のコンポーネントとやり取りし、Tableau 分析コンテンツを使用して、エンドユーザーに統合されたエクスペリエンスを提供します。拡張機能は、この統合されたエクスペリエンスを提供するために、ワークブックで使用されている表示可能なサマリーデータにアクセスします。特定の構成では、必要に応じて、ワークブックの参照元データにアクセスすることもできます。このアクセス

とインタラクションのレベルは、Tableau プラットフォームの一部として設計およびサポートされている Tableau 拡張機能 API によって実現します。

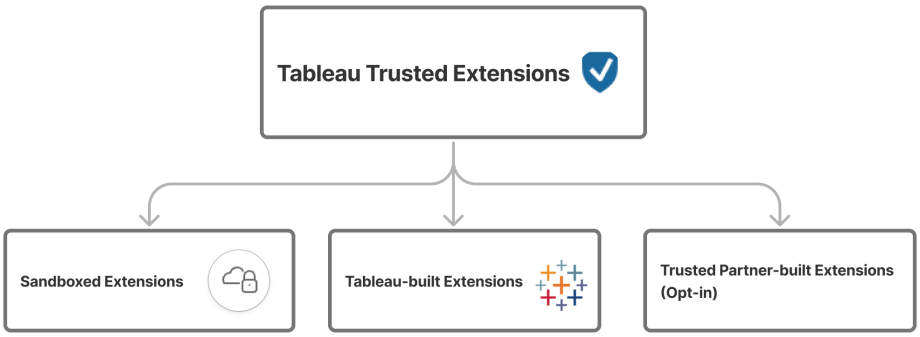
Tableau では、次の 2 つの拡張機能の標準がサポートされています。

- Tableau の信頼できる拡張機能
- ネットワーク対応拡張機能

Tableau の信頼できる拡張機能

Tableau では、コミュニティ全体で拡張機能の幅広い利用をサポートするため、拡張機能を大規模に使用できるようにするサービスを管理しています。これらの拡張機能は、Tableau がプラットフォーム機能として直接作成および保守する場合もあれば、Salesforce や Tableau パートナーの開発者が作成および保守する場合もあります。これらの開発者は、Tableau の製品チームと緊密に連携して、ソリューションを共同で展開し、Tableau Exchange を介してミュニティで幅広く利用できるようにしています。Tableau の信頼できる拡張機能は、Tableau インフラストラクチャ上で常にホストされ、Tableau 製品および開発チームによって管理されます。Tableau は、各拡張機能を展開または更新する前に、そのテストとレビューを行います。Tableau には、ユーザー クライアントとアクセス制御された Tableau 拡張機能ホスト間のみで通信を許可するという厳格な要件があります。この設計により、Tableau は、コミュニティに役立つ一般的な拡張機能のニーズに関する標準を提供することができます。この標準を満たすため、Tableau は、この目標の実現に向けたさまざまなソリューションを導入しています。

- サンドボックス化された拡張機能
- Tableau が作成した拡張機能
- 信頼できるパートナーが作成した拡張機能 (オプトイン)



サンドボックス化された拡張機能

Tableau はサンドボックス化された拡張機能をホストしています。サンドボックス化された拡張機能は、データ漏洩のリスクを排除するように特別に構成された保護環境で実行されます。Tableau のサンドボックス化された拡張機能ホストは、Web 上のその他のリソースやサービスからブロックされています。サンドボックス化された拡張機能を使用すると、拡張機能 API を使用してワークブックのデータをクエリできますが、そのデータをサンドボックスの外部に送信することはできません。クエリの結果はクライアントコンピューター(ブラウザー内)にあり、どこにも送信できません。アクセスを制限することにより、Tableau の従業員のみがサンドボックス化された拡張機能のすべての更新を管理したり、この拡張機能ホストにアクセスしたりできるようになります。サンドボックス化された拡張機能の環境とホスティング サービスに対しては、サードパーティーのコンサルタントによる侵入テストを実施済みです。サンドボックス化された拡張機能は既定で使用可能になっており、Tableau Public での使用も有効になっています。

Tableau が作成した拡張機能

Tableau が作成した拡張機能は、Tableau によってホストされます。具体的には、これらの拡張機能は、Tableau 開発チームがプラットフォーム機能として作成および保守しています。Tableau が作成した他の機能と同様に、Tableau が作成した拡張機能は Salesforce SDLC 標準に準拠しています。これらの拡張機能は、JavaScript をクライアントコンピューターに渡すように設計されていま

す。そのため、クライアントで拡張機能をレンダリングおよび処理できます。これらの機能は、Tableau チームによって直接サポートされており、Tableau の信頼できる同じホストインフラストラクチャを介して展開されます。Tableau が作成した拡張機能は既定で使用可能になっており、Tableau Public での使用も有効になっています。

信頼できるパートナーが作成した拡張機能 (オプトイン)

信頼できるパートナーが作成した拡張機能は、サードパーティーのパートナー開発者によって開発され、Tableau によってレビューおよびホストされます。パートナー開発者 (その多くは長年のコミュニティリーダーでもあります) のために展開を管理することで、Tableau とサードパーティーの Tableau Exchange パートナー間で、これらの新しいソリューションを共同で提供できるようになります。Tableau は、各サードパーティーソリューション (および各更新) を本番環境用に展開する前にテストします。これには、機能テストとネットワーク通信テストのほか、対処すべき脆弱性の自動コードスキャンが含まれます。Tableau の他の信頼できる拡張機能と同様に、これらの拡張機能は、ユーザー クライアントとTableau ホスト間の通信のみを許可します。ホストへの直接アクセスは制御され、Tableau の従業員のみに制限されています。Tableau がこれらのサードパーティー ソリューションに直接変更を加えたり、修正を適用したりすることはできませんが、サービスのアップタイムと可用性は Tableau によって管理されます。信頼できるパートナーが作成した拡張機能については、管理者が使用を有効にする必要がありますが、Tableau Public での使用が一般的に推奨されており、その使用が有効になっています。

ネットワーク対応拡張機能

ネットワーク対応拡張機能は、Tableau によって管理されない拡張機能です。また、ローカル ネットワークの内側か外側に配置された、Web へのフル アクセスを持つ Web サーバーでホストされます。そのため、ネットワーク対応拡張機能は、より具体的な統合ニーズや、Tableau の標準にとらわれないカスタム展開の目標に対応できるようになります。ネットワーク対応拡張機能は、他のアプリケーションやサービスに接続して、Tableau に新しい統合機能を提供することができます。これには、Tableau 以外の自然言語生成や書き戻しシナリオのサポートが含まれます。ネットワーク対応拡張機能には Web へのフル アクセス権があるため、外部リソースに接続して、豊富な機能とエクスペリエンスを提供できます。ただし、ネットワーク対応拡張機能は追加のサービスと通信する可能性があるほか、Tableau によって管理されないため、大規模な展開や導入を行う前に、これらの拡張機能を評価することをお勧めします。

ローカルに展開されたネットワーク対応拡張機能

拡張機能は、ローカル ネットワーク内で拡張機能をローカルに実行することなど、展開の柔軟性に対応できます。これは、エアギャップ ネットワーク要件のある展開や、独自の拡張機能を管理したい組織にとって特に便利です。たとえば、社内の開発者は、特定の展開のニーズを満たすカスタム拡張機能を作成できます。ネットワーク対応拡張機能は、自己展開の目標に対応しています。これらのローカル拡張機能は Tableau 以外のホスト上で実行され、独自のカスタム リソース定義の対象となるため、開発者と協力して、展開のユーザーのために、この内部拡張機能リソースと関連プロセスを管理する方法を適切に計画することをお勧めします。設計上、ネットワーク対応拡張機能のアーキテクチャは、Web へのフル アクセスを許可します。そのため、お客様は独自のレビュープロセスを開発して、拡張機能が許可された Web リソースのみと通信することや、リスクから保護するための適切なサイバーセキュリティ保護手段を備えていることを確認する義務があります。ローカル ネットワーク対応拡張機能は、使用を許可する前に、Tableau 管理者が展開拡張機能の許可リストに追加する必要があります。これらのローカル ソリューションは、Tableau Public では使用されません。

サードパーティープロバイダー管理のネットワーク対応拡張機能

ネットワーク対応拡張機能のアーキテクチャを使用すると、信頼できるサードパーティープロバイダーが拡張機能を提供することもできます。たとえば、正式なベンダー関係を確立している Tableau パートナーは拡張機能を直接提供できます。このような状況では、Tableau がネットワーク対応拡張機能を管理していなくても、プロバイダーは本番サポートやカスタム サービスを直接提供できます。そのため、信頼できる方法で処理すれば、信頼できるプロバイダーによってサポートされる Tableau のカスタマイズを合理化したり、拡張したりすることができます。お客様には、サードパーティープロバイダーと直接連携して、プロバイダーの開発プラクティスや拡張機能コードの設計のほか、セキュリティが内部ガイドラインをどのように満たしているかをより深く理解することをお勧めします。これらのネットワーク対応拡張機能は、使用を許可する前に、Tableau 管理者が展開拡張機能の許可リストに追加する必要があります。まれなケースでは、コミュニティの目標に対する唯一の例外として、これらのソリューションを Tableau Public で使用できるようにすることができます。

ネットワーク対応拡張機能のリスクに関する考慮事項

拡張機能は Web アプリケーションなので、ネットワーク対応拡張機能は特定のタイプの悪意ある攻撃に対して脆弱である可能性があり、潜在的にコンピューターやデータをリスクにさらす可能性が

あります。毎年、Open Web Application Security Project (OWASP) は最も危険な Web アプリケーションのセキュリティリスクを特定しています。このリスクには以下のものが含まれます。

- SQL インジェクション
- クロスサイトスクリプト (XSS)
- 機密データの流出

これらのリスクのため、拡張機能の開発者がユーザー入力を適切に検証および処理しない場合や、動的クエリを生成して機密データベースにアクセスする場合、拡張機能が危険にさらされる可能性があります。Tableau で許可する拡張機能を評価する際には、その拡張機能で認証、データアクセス、ユーザー入力を管理する方法のほか、セキュリティリスクの軽減方法を検討するようにしてください。

ネットワーク対応拡張機能に関するセキュリティ上の脅威を軽減する

拡張機能とは何かを理解することが企業のリスクを特定する最初の手順となります。多くの場合、ダッシュボードとViz の拡張機能はワークブックの参照元データにアクセスせず、すべての JavaScript コードは、ユーザーのコンピューターで実行中のブラウザーのコンテキストで実行されます。このような場合、拡張機能がドメイン外部のサードパーティー サイトでホストされている場合でも、データがコンピューターから出ることはありません。一部の拡張機能では、Tableau をドメインに既に展開している他のアプリケーションと接続することができます。

Tableau では拡張機能のセキュリティ対策とセキュリティ要件を提供しています。これらの要件は、Tableau Desktop、Tableau Server、および Tableau Cloud で有効です。

- すべての拡張機能は HTTP セキュア (HTTPS) プロトコルを使用する必要があります。
- 既定では、ネットワーク対応拡張機能を含むダッシュボードを使用しているすべてのユーザーにメッセージが表示され、拡張機能の実行パーミッションを求められます。拡張機能が参照元データにアクセスする場合、パーミッションを要求する必要があります。
- Tableau Server または Tableau Cloud 上で実行するには、ネットワーク対応拡張機能の URL を許可リストに追加する必要があります。このリストは、Tableau Server ではサーバー管理者が管理し、Tableau Cloud ではサイト管理者が管理します。
- Tableau Server とTableau Cloud では、サーバー管理者またはサイト管理者が各ネットワーク対応拡張機能でメッセージを表示するかどうかを制御できます。

詳細については、Tableau Server でダッシュボードとViz の拡張機能を管理するを参照してください。

Tableau を使用した拡張機能の管理

拡張機能を使用すると、ダッシュボードに独自の機能を追加したり、ワークシートに新たなビジュアライゼーションを追加したりすることができます。拡張機能を使用し、ダッシュボードを Tableau 外部のアプリケーションに直接統合することができます。拡張機能によってさまざまな可能性が開かれますが、会社や企業では拡張機能をどのように展開するかの制御を保つことが必要となる場合があります。この点で、拡張機能にはユーザーが使用する他のソフトウェアとの差はありません。会社でソフトウェア アプリケーションを展開する前に、ソフトウェアが想定どおりに動作し、安全であることを徹底的にテストして検証する必要があります。拡張機能にも同じことが当てはまります。

まず、ユーザーが持つべきアクセス権限のレベルを決定し、使用したい拡張機能 (または逆に、使用したくない拡張機能) を特定します。次に、Tableau 内のコントロールと機能を使用して、ユーザーがアクセスできるダッシュボードと Viz の拡張機能を制限し、管理します。

- Tableau Desktop で拡張機能を追加または使用可能なユーザーを制限する必要がありますか。Tableau Desktop での推奨事項を参照してください
- ユーザーがアクセス可能な拡張機能を制限または制御する必要がありますか。Tableau Server および Tableau Cloud での推奨事項を参照してください。

Tableau Desktop での推奨事項

会社に Tableau Desktop を展開するさまざまなオプションがあります。拡張機能への無制限のアクセスを許可することも、誰がどのような状況で拡張機能にアクセスするかに関して制限を設けることもできます。

既定では、Tableau Desktop ユーザーは、Tableau の信頼できる拡張機能とネットワーク対応拡張機能に無制限にアクセスできます。Tableau では、既定の設定を変更するために、インストール時に任意の順序で使用できる 5 つのオプションを提供しています。

- すべての拡張機能をオフにする (`DISABLEEXTENSIONS`)
- ネットワーク対応拡張機能をオフにする (`DISABLENETWORKEXTENSIONS`)
- サンドボックス化された拡張機能をオフにする (`DISABLESANDBOXEXTENSIONS`)
- Tableau が作成した拡張機能をオフにする (`DISABLETABTRUSTEDEXTENSIONS`)
- 信頼できるパートナーが作成した拡張機能をオフにする (`DISABLE3PTRUSTEDEXTENSIONS`)

注: Tableau Desktop のインストール後にこれらの設定を変更するには、各 Desktop でレジストリを編集するか (Windows)、スクリプトを実行し (Mac) ます。「ダッシュボードの拡張機能をオフにする」を参照してください。

展開シナリオの例

インストール設定を使用すると、Tableau Desktop をさまざまな方法で展開できます。

- **すべての拡張機能を許可する -** この展開シナリオでは、Tableau の作成者が使用したい Tableau の信頼できる拡張機能とネットワーク対応拡張機能を選択し、管理者はそれを信頼します。Tableau Desktop ユーザーに最大限の柔軟性を与えるには、既定のインストール設定を使用します。既定の設定では、Tableau Desktop ユーザーは、Tableau の信頼できる拡張機能とネットワーク対応拡張機能に無制限にアクセスできます。既定の設定は `DISABLEEXTENSIONS=0` と `DISABLENETWORKEXTENSIONS=0` です。「コマンドラインから Tableau Desktop をインストールする」を参照してください。

  Tableau の信頼できる拡張機能のみを許可する - このシナリオでは、Tableau 拡張機能サービスに、コミュニティで広範に使用する拡張機能の適切なレビュー、展開、およびキュレーションを任せますが、ネットワーク対応拡張機能については不明であるため、現時点ではその使用を禁止します。ネットワーク対応拡張機能のサポートをオフにするには、`DISABLENETWORKEXTENSIONS` プロパティを設定します (`DISABLENETWORKEXTENSIONS=1`)。拡張機能を有効にするには、既定の設定をそのまま使用します (`DISABLEEXTENSIONS=0`)。「コマンドラインから Tableau Desktop をインストールする」を参照してください。

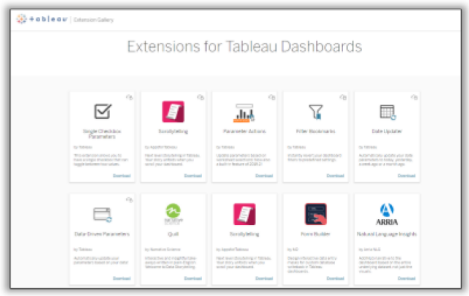
- **サンドボックス化された拡張機能のみを許可する -** このシナリオでは、管理者がサンドボックス化された拡張機能を以前に使用したことがあり、安全に使用できると考えています。ただし、Tableau が作成した拡張機能と信頼できるパートナーが作成した拡張機能をより広範に使用できるようにポリシーを更新する準備ができていないか、ネットワーク対応拡張機能を許可するようにポリシーを変更したいとは考えていません。サンドボックス化された拡張機能をこれまでどおり使用できるようにし、他のすべての形式の Tableau の信頼できる拡張機能とネットワーク対応拡張機能を使用できないようにしたいだけです。ネットワーク

対応拡張機能、Tableau が作成した拡張機能、信頼できるパートナーが作成した拡張機能のサポートをオフにするには、DISABLENETWORKEXTENSIONS プロパティ (DISABLENETWORKEXTENSIONS=1)、DISABLETABTRUSTEDEXTENSIONS プロパティ (DISABLETABTRUSTEDEXTENSIONS=1)、DISABLE3PTRUSTEDEXTENSIONS プロパティ(DISABLE3PTRUSTEDEXTENSIONS=1) を設定します。拡張機能を有効にするには、既定の設定をそのまま使用します (DISABLEEXTENSIONS=0)。「コマンド ラインから Tableau Desktop をインストールする」を参照してください。

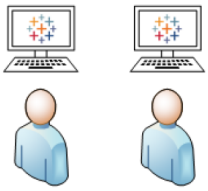
- **拡張機能を許可しない -** このシナリオでは、ユーザーがあらゆるタイプの拡張機能を使用できないようにします。この場合、DISABLEEXTENSIONS プロパティを使用してすべての拡張機能のサポートをオフにします (DISABLEEXTENSIONS=1)。「コマンド ラインから Tableau Desktop をインストールする」を参照してください。

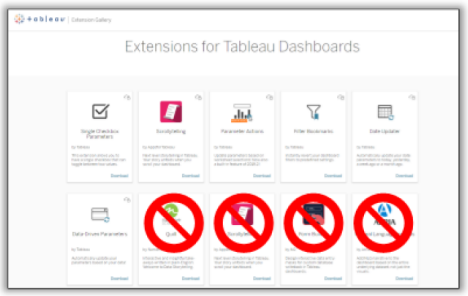
設定の組み合わせを使用する - すべての拡張機能に対する無制限のアクセスが必要なユーザー、サンドボックス化された拡張機能へのアクセスで十分なユーザー、拡張機能へのアクセスが不要なユーザーが混在する場合もあります。拡張機能のオプションはデスクトップごとに設定されるため、特定のユーザーや使用事例に合わせて展開を構成できます。

Web 作成 - ユーザーが Tableau Server または Tableau Cloud を使用できる場合、Web 作成を使用して拡張機能にアクセスできます。Web 作成では、拡張機能についてのサーバー設定やサイト設定が適用されます。このシナリオでは、サーバー管理者およびサイト管理者は、どの拡張機能へのアクセスをユーザーに許可するかを決定できます。これは、サーバーとサイトの拡張機能の設定に基づいて管理されます。この設定には、Tableau の信頼できる拡張機能のオプションと、明示的なネットワーク対応拡張機能の拡張機能許可リストが含まれます。



If extensions are enabled on Tableau Desktop, users have unrestricted access to extensions





You can turn off extensions on Tableau Desktop to restrict access per desktop. And restrict access to Trusted extensions only.

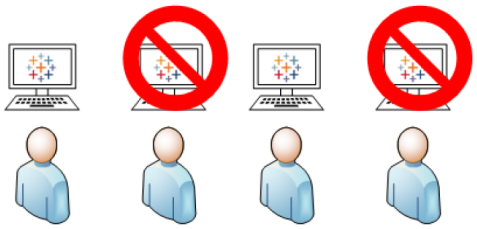


Tableau Server および Tableau Cloud での推奨事項

ユーザーが Tableau Server または Tableau Cloud にアクセス可能な場合、ビルトインのセキュリティ制御を使用し、使用可能な拡張機能やどのような環境かについて制限を設けることができます。Tableau Desktop で拡張機能をオフにした場合でも、Web 作成でユーザーが拡張機能を追加できるようにすることはできますが、使用できる拡張機能の数を管理者が実行を承認したものだけに制限できます。

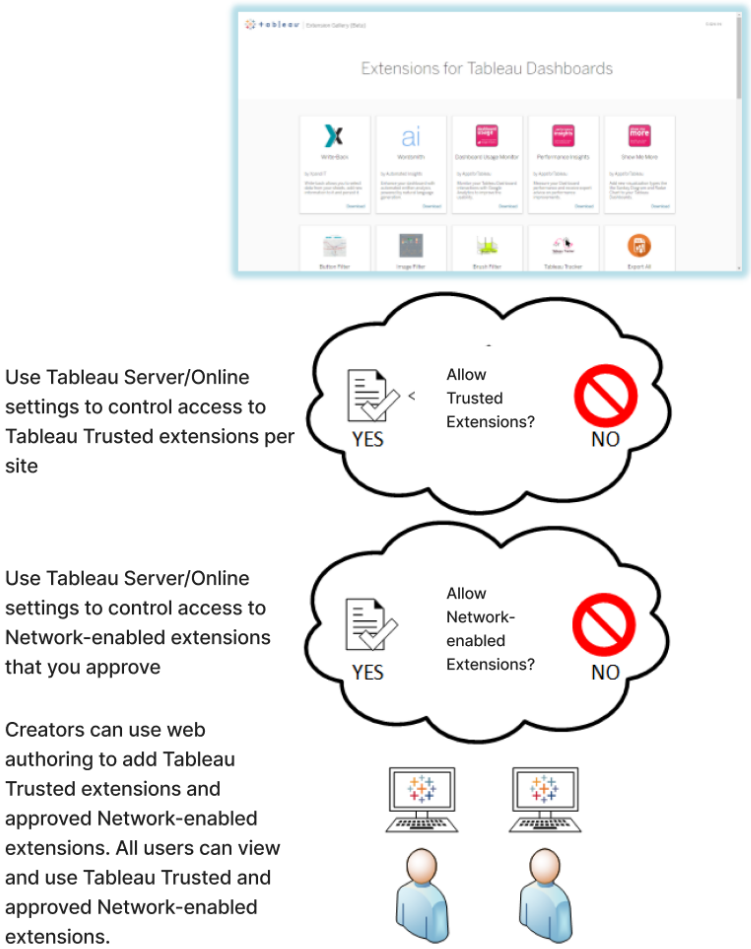


Tableau の信頼できる拡張機能と許可リスト上のネットワーク対応拡張機能を許可する

Tableau 2025.3 以降、Tableau では、Tableau の信頼できる拡張機能の一部を既定で実行することが許可されます。つまり、サンドボックス化された拡張機能と Tableau が作成した拡張機能は昇格して使用できますが、信頼できるパートナーが作成した拡張機能については、管理者が有効にできます。当社では、Tableau Public などの環境の場合と同じように、Tableau のすべての信

頼できる拡張機能の使用を推奨してますが、最終的には管理者が展開に関連するポリシーを定義することが極めて重要だと考えています。

ネットワーク対応拡張機能は許可リストに追加されていない限り許可されません。管理者は、設定を編集し、サイトの許可リストにネットワーク対応拡張機能を追加できます (**[設定] > [拡張機能] > [特定の拡張機能の有効化]**)。

注: Tableau 2018.2 とTableau 2018.3 で許可リストを拡張機能の既定の動作にするには、サイトの設定を変更する必要があります。[拡張] 設定ページの **[拡張の既定の動作]** で、**[不明の拡張の実行を有効化し...]** のオプションをオフにします。Tableau Server 2019.1、Tableau 2019.2、Tableau 2019.3 では、許可リストに追加されていない限り、既定では拡張機能の実行は許可されません。

許可リストのチェックリスト:

- 拡張機能はユーザーが知っている信頼できるソースより得られたものですか?
- 拡張機能の URL を確認してください。URL が疑わしく思えたり、怪しいドメイン名が含まれていますか?
- 拡張機能では完全なデータ(参照元データ) へのアクセスを要求していますか、またはサマリーデータへのアクセスを要求していますか? データアクセスの理解を参照してください。
- 広範な使用を許可する前に、拡張機能をテストしてください。「拡張機能でのセキュリティのテスト」を参照してください。「ネットワーク対応拡張機能のセキュリティテスト」を参照してください。

許可リストに拡張機能を追加します。

- 許可リストへの拡張機能の追加と拡張機能の構成を参照してください。

Tableau Server での特定の拡張機能の実行のブロック

Tableau Server では、ブロックリストに拡張機能の URL を追加すると、特定の拡張機能をブロックできます。これは、複数のサイトで拡張機能において異なる構成を行っている場合に便利です。たとえば、社内の拡張機能やサードパーティの拡張機能をテストできるテストサイトがある場合、拡張機能の既定の動作を有効にしており、リストに記載されていない拡張機能がワークブッ

クの参照元データにアクセスしなければその拡張機能の実行が許可されるかもしれません。ブロックリストに拡張機能を追加すると、テストサイトで誤って使用されるのを防げます。

- 許可しない拡張機能の URL をブロックリストに追加します。このオプションは、Tableau Server でのみ使用できます。特定の拡張機能のブロックを参照してください。

サイトでの拡張機能のオフ

Tableau Server および Tableau Cloud では拡張機能が既定で有効になっています。Tableau Server では、サーバー管理者がサイトの拡張機能をオフにできます。Tableau Cloud では、サイト管理者がサイトの拡張機能をオフにできます。Tableau Server では、サーバー管理者が拡張機能を完全にオフにすることができ、それによってサイト設定が上書きされます。許可するネットワーク対応拡張機能は許可リストで制御できるので、サーバーやサイトでこの設定を変更する必要はありません。また、Tableau の信頼できる拡張機能の設定を制御することもできます。サンドボックス化された拡張機能と Tableau が作成した拡張機能は既定で許可されます。

- サイト (Tableau Server、Tableau Cloud) で拡張機能を無効にするには、サイトでユーザーが拡張機能を実行できるようにするサイト設定を変更します。制御に関する拡張機能の使用を参照してください。

ネットワーク対応拡張機能を実行するためのユーザープロンプトの表示または非表示

許可リストにネットワーク対応拡張機能を追加すると、ユーザーが拡張機能をダッシュボードに追加している場合、または拡張機能を含むビューを操作している場合に既定でユーザーにプロンプトを表示するかどうかを構成することができます。このプロンプトには、ネットワーク対応拡張機能の詳細および拡張機能がフルデータにアクセスできるかどうかが記載されています。このプロンプトによりユーザーには、拡張機能の実行を許可または拒否する権限が与えられます。ユーザーにプロンプトが表示されないようにし、拡張機能がすぐに実行されるようにできます。サイトに対して有効にすると、サンドボックス化された拡張機能が既定で許可され、ユーザーにプロンプトが表示されなくなります。

Tableau の信頼できる拡張機能のすべてまたは一部をオフにする

Tableau 2025.3 以降、Tableau では、Tableau Server と Tableau Cloud で Tableau の信頼できる拡張機能の一部を既定で実行することが許可されます。サンドボックス化された拡張機能と Tableau が作成した拡張機能は昇格して使用できますが、信頼できるパートナーが作成した拡

張機能については、管理者が有効にできます。サンドボックス化された拡張機能は、保護された環境で実行され、Tableau によってホストされます。Tableau が作成した拡張機能は、Tableau によって開発、保守、ホストされます。信頼できるパートナーが作成した拡張機能は、サードパーティーのパートナー開発者によって開発および保守されますが、可用性と拡張性を確保するために、ソリューションと更新は、Tableau によってレビューおよびホストされます。

管理者は、ユーザーがサイトで Trusted の信頼できる拡張機能のすべてまたは一部を実行できるようにするかどうかを制御できます。信頼できる拡張機能を許可リストに追加する必要はありません。Tableau の信頼できる拡張機能が許可されている場合、ユーザーは、Tableau の信頼できる拡張機能をワークブックに自由に追加して、Tableau の信頼できる拡張機能が含まれたワークブックを開いたり、使用したりできます。Tableau の信頼できる拡張機能をブロックする必要がある場合は、サーバーの拒否リストに追加する特定の拡張機能の URL について、Tableau にお問い合わせください。
サーバー管理者は、任意の拡張機能を展開のグローバル拒否リストに追加できます。Tableau の信頼できる拡張機能を完全にオフにする必要がある場合は、サイトの既定の設定を変更できます。Tableau の信頼できる拡張機能の既定の設定を変更すると、許可リストにある拡張機能だけが実行を許可されます。

Tableau Server のキー管理システム

Tableau Server には、保存中の暗号化を有効にできるキー管理システム (KMS) のオプションが 3 つあります。1 つは、Tableau Server のすべてのインストールで使用できるローカル オプションです。その他の 2 つのオプションには Advanced Management の機能が必要ですが、別の KMS を使用できます。

重要: 2024 年 9 月 16 日以降、Advanced Management は独立したアドオン オプションとしては利用できなくなりました。Advanced Management 機能は、以前に Advanced Management を購入した場合、または特定のライセンス エディション (Tableau Enterprise (Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

バージョン2019.3以降、Tableau Serverでは次のKMSオプションが追加されました。

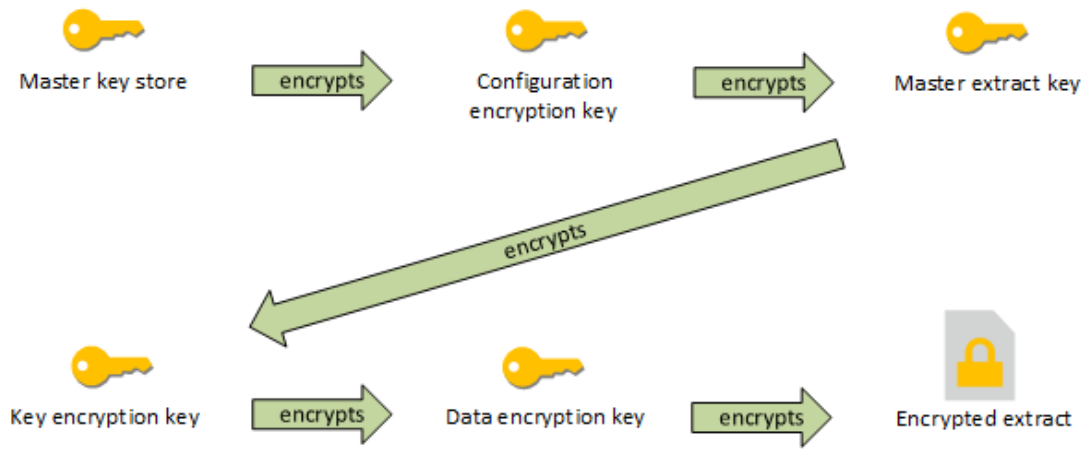
- すべてのインストールで使用できるローカルKMS。これについては以下で説明します。
- Advanced Managementの一部として提供されるAWSベースのKMS詳細については、「AWSキー管理システム」を参照してください。

バージョン2021.1以降、Tableau Serverでは次のKMSオプションが追加されました。

- Advanced Managementの一部として提供されるAzureベースのKMS詳細については、Azure Key Vaultを参照してください。

Tableau Serverのローカル KMS

Tableau Serverのローカル KMSは、サーバーシークレットの管理で説明されているシークレット保存機能を使用してマスター抽出キーの暗号化と保存を行います。このシナリオで、Javaキーストアはキー階層のルートとして機能します。Javaキーストアは Tableau Server と共にインストールされます。マスターキーへのアクセスは、オペレーティングシステムのネイティブファイルシステム認可メカニズムによって管理されます。既定の構成では、暗号化された抽出には Tableau Server のローカルKMSが使用されます。ローカルKMSと暗号化された抽出のキー階層は次のとおりです。



構成のトラブルシューティング

マルチノードの構成ミス

AWS KMS のマルチノードセットアップでは、クラスタ内の別のノードが正しく構成されていなくても、`tsm security kms status` コマンドから正常 (OK) ステータスが返される場合があります。KMS のステータスチェックでは、Tableau Server 管理コントローラープロセスが実行されているノードだけが報告され、クラスタ内の他のノードは報告されません。既定では、Tableau Server 管理コントローラープロセスは、クラスタ内の最初のノードで実行されます。

したがって、他のノードが誤って構成されているために Tableau Server が AWS CMK にアクセスできない場合、これらのノードではさまざまなサービスについてエラー状態が報告され、起動が失敗します。

KMS を AWS モードに設定した後で一部のサービスが起動に失敗する場合は、コマンド `tsm security kms set-mode local` を実行してローカルモードに戻します。

Tableau Server で RMK と MEK を再生成する

Tableau Server でルートマスターキーとマスター暗号化キーを再生成するには、`tsm security regenerate-internal-tokens` コマンドを実行します。

保存中の抽出の暗号化

保存中の抽出の暗号化は、.hyper 抽出を Tableau Server に保存しながら暗号化できるデータセキュリティ機能です。

Tableau Server 管理者は、サイト上のすべての抽出の暗号化を実施する、または、特定のパブリッシュ済みワークブックやデータソースに関連付けられたすべての抽出の暗号化をユーザーが指定するのを許可できます。

制限事項

暗号化を行うには、古い .tde ファイル抽出を .hyper ファイル抽出にアップグレードする必要があります。これは暗号化ジョブの一環として自動的に行われます。抽出アップグレードによる影響の詳細は、.hyper 形式への抽出のアップグレードを参照してください。

一時ファイルとキャッシュファイルは、この機能では保存中に暗号化されません。

ワークブック (.twb) およびデータ ソース ファイル (.tds) は、この機能では暗号化されません。これらのファイルには、データベース表の列名や書式設定手順などのメタデータが含まれます。場合によっては、フィルターに含まれている場合には、これらのファイルに一部の行レベルのデータが含まれていることがあります。

Excel ファイルや JSON ファイルなど他のデータ ファイルは、パブリッシュ前に抽出に変換されない限り、この機能では暗号化されません。

抽出はサーバーからダウンロードされるときに復号化されます。

パフォーマンスの概要

バックグラウンダー負荷の増加

保存中の暗号化をオンにすると、バックグラウンダー負荷が若干ないし中程度に増加することがあります。暗号化と復号化は計算集約型動作です。保存中の暗号化により既存のバックグラウンダー ジョブが変更され、バックグラウンダーでの新しいジョブの実行が導入されます。バックグラウンダー負荷の全体的な増加は、影響を受ける抽出の数とサイズ、および以下のシナリオが適用される頻度によって変わります。

- **初回パブリッシュ**: 暗号化の必要がある抽出を使用してワークブックやデータ ソースをパブリッシュする場合、暗号化はサーバーのバックグラウンダーで行われます。
- **Tableau Server による抽出更新**: Tableau Server での暗号化抽出の完全更新と増分更新は若干多くの CPU を使用します。
- **Tableau Bridge およびサードパーティー アプリケーション (Informatica、Alteryx など) による抽出更新**: これらのフローでは、更新された抽出のバックグラウンダーでスケジュールされた新しい暗号化ジョブが必要で、バックグラウンダー負荷が若干ないし中程度に増加します。
- **既にパブリッシュされたワークブックやデータ ソースでの抽出の暗号化と復号化**: 保存中の暗号化のサイト設定が **[Enable (有効)]** に設定されている場合、ユーザーは Tableau Server で、既にパブリッシュされているワークブックやデータ ソースに含まれる抽出の暗号化または復号化を選択することができます。抽出の数とサイズによっては、バックグラウンダーにかかる負荷が若干ないし中程度に増加します。
- **サイトの暗号化モードの変更**: 保存中の暗号化に関するサイト設定を **[Disable (無効)]** または **[Enforce (実行)]** に切り替えると、バックグラウンダーはサイト上のすべての既存の各抽

出を切り替えに応じて復号化または暗号化します。抽出の数とサイズによっては、すべての抽出が復号化または暗号化されるまで、バックグラウンダーに対する負荷が大幅に増加する可能性があります。

- **暗号化キーの回転**: 暗号化キーを回転すると、新しい暗号化キーを使用し、そのサイトでパブリッシュされたすべての既存の抽出がバックグラウンダーで再暗号化されます。抽出の数とサイズによっては、すべての抽出が再暗号化されるまで、バックグラウンダーに対する負荷が大幅に増加する可能性があります。

容量の上限かそれを超えて実行している場合は、次の点を考慮してください。

- 追加バックグラウンダープロセスとリソースを追加する。
- ユーザーがサイト全体を暗号化する代わりに個別のワークブックとデータソースを暗号化すること、またはそれが必要でないサイトで保存中の暗号化を無効にすることを許可する。スケジュールされた抽出更新およびアドホック抽出更新が暗号化ジョブや復号化ジョブより優先されることに注意してください。

Viz の読み込み時間とワーカー負荷の増加

たとえば、クエリパフォーマンスが Viz やダッシュボードを読み込むか操作するとき、ディスクからメモリに読み込まれる場合にデータを一度復号化する必要があります。このため、ユーザーが初めてワークブックを読み込む際に、Viz の読み込み時間とワーカー ノードでの CPU 消費が若干増加します。データは既にメモリ内で復号化されているため、これにより、同時にそれらのワークブックにアクセスする他のユーザーに影響が及ぶことはありません。

バックアップと復元への影響

バックアップに含まれる暗号化された抽出は暗号化されたままです。バックアップ ファイル (.tbks) のサイズは、暗号化された抽出の圧縮が効果的でないため最大 50 ~ 100% 増加する可能性があります。このサイズの増加は、他の要素のうち、暗号化されている抽出の数によって決まります。暗号化された抽出を含むバックアップを復元する時間は、暗号化キーを交換する時間が原因で若干長くなる可能性があります。

Tableau Server インストールに含まれる抽出がほとんどまたはすべて暗号化されている場合、バックアップ中に圧縮を無効にし、バックアップの所要時間を大幅に改善させることを検討してください。TSM バックアップの詳細については、tsm maintenance backup を参照してください。

サイトでの保存中の暗号化の実施

Tableau Server 管理者は、サイト上のすべての抽出の暗号化を実施することができます。

1. Web ブラウザーで Tableau Server にサーバー管理者としてサインインします。
2. 構成するサイトに移動します。
3. **[設定]** をクリックします。
4. [保存時の抽出の暗号化] セクションまで下にスクロールします。
   [Enforce (実行)] をクリックし、サイト上でパブリッシュおよび保存されているすべての抽出を暗号化します。
   サイトに保存されている既存の抽出をすべて暗号化するには、しばらく時間がかかる場合があります。
5. **[保存]** をクリックします。

サイトでの保存中の暗号化の有効化

Tableau Server 管理者は、特定のパブリッシュ済みワークブックやデータソースに関連付けられたすべての抽出の暗号化をユーザーが指定するのを許可できます。

1. Web ブラウザーで Tableau Server にサーバー管理者としてサインインします。
2. 構成するサイトに移動します。
3. **[設定]** をクリックします。
4. [保存時の抽出の暗号化] セクションまで下にスクロールします。
5. **[Enable (有効)]** をクリックすると、ユーザーにオプションでサイト上の抽出の暗号化を許可できます。
   [Enable (有効)] に変更すると、保留中の復号化ジョブと暗号化ジョブがキャンセルされます。暗号化ジョブは作成されません。
6. **[保存]** をクリックします。

サイトでの保存中の暗号化の無効化

1. Web ブラウザーで Tableau Server にサーバー管理者としてサインインします。
2. 構成するサイトに移動します。
3. **[設定]** をクリックします。
4. [保存時の抽出の暗号化] セクションまで下にスクロールします。

5. **[Disable (無効)]** をクリックすると、サイト上の抽出の暗号化を許可しません。
   [Disable (無効)] に変更すると、既存の暗号化された抽出がすべて復号化されます。サイトに保存されている抽出をすべて復号化するには、しばらく時間がかかる場合があります。
6. **[保存]** をクリックします。

すべてのサイトの抽出暗号化モードの表示

1. 複数サイトサーバーのサイトメニューで **[すべてのサイトを管理]** をクリックします。

   注: [すべてのサイトを管理] オプションは、サーバー管理者としてサインインした場合にのみ表示されます。

2. **[サイト]** をクリックします。
3. 各サイトの暗号化モードは、**[保存時の抽出の暗号化]** 列に表示されます。

パブリッシュ済みワークブックまたはデータソースでの抽出の暗号化または復号化

注: 特定のパブリッシュ済みワークブックまたはデータソースに関連付けられた抽出を暗号化または復号化するオプションは、保存中の暗号化に関するサイト設定が **[Enable (有効)]** に設定されている場合にのみ使用できます。サイトが [Disable (無効)] に設定されている場合、すべてのコンテンツは暗号化されません。サイトが [Enforce (実行)] に設定されている場合、すべてのコンテンツが暗号化されます。
注: 所有者または管理者である必要があります。

1. パブリッシュ済みワークブックまたはパブリッシュ済みデータソースページに移動します。
2. **[暗号化された抽出]** または **[復号化された抽出]** と記載されたドロップダウンメニューをクリックします。
3. **[復号化済み]** を選択します。
   「抽出を復号化しています」というメッセージが表示されます。
   - または -
   [暗号化済み] を選択します。
   暗号化ジョブが開始されます。

または、カードビューアクションメニュー、リストビューアクションメニュー、およびヘッダーセクションのアクションメニューの抽出を暗号化または復号化することもできます。

複数アイテムの暗号化または復号化

1. [データソース] ページに移動します。
2. 1 つ以上のデータソースの隣にあるチェックボックスを選択します。
3. [データソース] ページの左上で **[アクション]** をクリックします。
4. **[Encrypt (暗号化)]** または **[Decrypt (復号化)]** をクリックします。

単一項目の暗号化ステータスの表示

1. サイトにサインインします。
2. 単一のデータソース ページに移動します。
   - または -
   埋め込みデータソースを含むワークブックの単一のワークブック ページに移動します。
3. 暗号化ステータスがページに表示されます。

暗号化ステータスによるデータソースのフィルター

1. サイトで、**[検索]** をクリックします。
2. 右上隅で [検索: 最上位のプロジェクト] ドロップダウン メニューをクリックし、**[すべてのデータソース]** を選択します。
3. フィルター アイコンをクリックします。
4. [ライブまたは抽出] セクションまで下にスクロールし、フィルター オプション ([すべて]、[ライブ]、[抽出]、[復号化された抽出]、[暗号化された抽出]、[現在暗号化中]、または [現在復号化中]) を選択します。
5. フィルター結果に ".tde ファイルへのライブ" 接続と ".hyper ファイルへのライブ" 接続を含める場合は、「.tde および .hyper ファイルを含める」の隣のチェックボックスをオンにします。

暗号化ステータスによるワークブックのフィルター

1. サイトで、**[検索]** をクリックします。
2. 右上隅で [検索: 最上位のプロジェクト] ドロップダウン メニューをクリックし、**[Select All workbooks (すべてのワークブック)]** を選択します。
3. フィルター アイコンをクリックします。
4. [ライブまたは抽出] セクションまで下にスクロールし、フィルター オプション ([すべて]、[ライブ]、[抽出]、[パブリッシュ済み]、[復号化された抽出]、[暗号化された抽出]、[現在暗号化中]、または [現在復号化中]) を選択します。
5. フィルター結果に ".tde ファイルへのライブ" 接続と ".hyper ファイルへのライブ" 接続を含める場合は、「.tde および .hyper ファイルを含める」の隣のチェックボックスをオンにします。
   フィルター選択に一致する接続が少なくとも 1 つあるワークブックが表示されます。

抽出の暗号化または復号化のバックグラウンドタスクのステータスの表示

1. サイトで **[サイトのステータス]** をクリックします。
2. **[抽出以外のバックグラウンドタスク]** をクリックし、完了したバックグラウンドタスクと保留中のバックグラウンドタスクの詳細を表示します。
   注: **[抽出以外のバックグラウンドタスク]** には抽出更新に関連しないすべてのタスクが含まれているため、暗号化ジョブが含まれます。
3. [タスク] メニューで **[抽出の暗号化]** または **[抽出の復号化]** を選択し、**[適用]** をクリックします。
4. [時間範囲] メニューで範囲を選択します。
   抽出に基づくパブリッシュ済みのデータソースおよびワークブックすべてにおいて、"抽出の暗号化" と"抽出の復号化" のバックグラウンドタスクが表示されます。

tabcmd ユーティリティ

tabcmd コマンドライン ユーティリティには、抽出の暗号化を制御するためのコマンドとオプションがあります。詳細については、tabcmdのドキュメントを参照してください。

サイト作成時の抽出暗号化モードの指定

```
tabcmd createsite <site-name> --extract-encryption-mode [enforced |
enabled | disabled]
```

サイト編集時の抽出暗号化モードの指定

```
tabcmd editsite <site-name> --extract-encryption-mode [enforced |
enabled | disabled]
```

サイトのリスト作成時の抽出暗号化モードの取得

```
tabcmd listsites --get-extract-encryption-mode
```

サーバーにワークブック、データソース、または抽出をパブリッシュするときの抽出の暗号化

```
tabcmd publish "filename.hyper" --encrypt-extracts
```

サイト上のすべての抽出の復号化

> **注:** 抽出の数とサイズによっては、この操作は大量のサーバー リソースを消費する場合があります。このコマンドは、通常の営業時間外に実行することを検討してください。

```
tabcmd decryptextracts <site-name>
```

サイト上のすべての抽出の暗号化

> **注:** 抽出の数とサイズによっては、この操作は大量のサーバー リソースを消費する場合があります。このコマンドは、通常の営業時間外に実行することを検討してください。

```
tabcmd encryptextracts <site-name>
```

新しい暗号化キーを使用したサイト上のすべての抽出の再暗号化

サイトを指定する必要があります。

> **注:** 抽出の数とサイズによっては、この操作は大量のサーバー リソースを消費する場合があります。このコマンドは、通常の営業時間外に実行することを検討してください。

```
tabcmd reencryptextracts <site-name>
```

詳細については、reencryptextractsを参照してください。

Tableau Server Rest API

Tableau Server REST API では、プログラムによって Tableau Server リソースを管理できます。このアクセスを使って、独自のカスタム アプリケーションを作成したり、Tableau Server リソースとの相互作用をスクリプトしたりできます。

詳細については、「抽出の暗号化方法」を参照してください。

ネットワーク セキュリティ

Tableau Server には 3 つの主要なネットワーク インターフェイスがあります。

- **クライアントから Tableau Server へ**: クライアントとして、Web ブラウザー、Tableau Mobile、Tableau Desktop、または tabcmd ユーティリティを指定できます。
- **Tableau Server からデータベース**: データの抽出更新やデータベースへのライブ接続の処理を行うには、Tableau Server はご利用のデータベースと通信する必要があります。
- **サーバーのコンポーネント通信**: これは分散環境のみが対象です。

ほとんどの組織では、Tableau Server を構成してインターネットおよび SMTP サーバーとも通信します。

クライアントから Tableau Server へ

Tableau Server クライアントとして Web ブラウザー、Tableau Mobile、Tableau Desktop、または tabcmd コマンドを実行するデバイスを指定できます。Tableau Server とそのクライアント間の通信では、標準の HTTP 要求と応答が使用されます。すべての通信に HTTPS を使用するよう Tableau Server を構成することをお勧めします。Tableau Server を SSL 用に構成する場合、クライアント間のすべてのコンテンツと通信は SSL を使用して暗号化され、要求と応答には HTTPS プロトコルを使用します。

既定では、パスワードは、1024 ビットの公開/秘密キーの暗号化を使用してブラウザーおよび tabcmd から Tableau Server に通信されます。このレベルの暗号化は、セキュアな通信には不十分であると見なされます。さらに、この方法では、公開キーが明確かつネットワーク レイヤー認証なしで送信されるため、中間者攻撃の影響を受けやすくなります。

クライアントから Tableau Server へのネットワーク トラフィックを適切に保護するには、信頼できる認証機関からの証明書で SSL を設定する必要があります。

Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成するを参照してください。

インターネットからのクライアントアクセス

ゲートウェイプロキシ サーバーでインターネットからTableau Server へのセキュア クライアントアクセスを有効にすることをお勧めします。Tableau Server を DMZ や、その他の保護されている内部ネットワークの外で実行することはお勧めしません。

SSL を有効にしたリバースプロキシ サーバーが、インターネットからのすべてのインバウンドトラフィックを処理するよう構成します。このシナリオでは、リバースプロキシは Tableau Server が通信する唯一の外部 IP アドレス(または複数のリバースプロキシでインバウンドリクエストのロードバランシングを行う場合はアドレスの範囲)です。リバースプロキシは要求側クライアントに対して透過的なため、Tableau Server ネットワーク情報を難読化し、クライアント構成を簡素化します。

設定の詳細については、Tableau Server のプロキシとロードバランサーの設定を参照してください。

クリックジャック保護

既定では、Tableau Server は *クリックジャック保護*を有効にしています。これにより、攻撃者がユーザーを誘い込んでリンクをクリックし、情報を入力させるよう、無害に見えるページの上に透明バージョンのページを重ねる、特定の種類の攻撃を防止するのに役立ちます。クリックジャック保護が有効な場合、Tableau Server は埋め込みビューに特定の制限をかけます。詳細については、クリックジャック保護を参照してください。

Tableau Server からデータベースへ

Tableau Server はデータベースへの動的接続を行って、結果セットを処理し、抽出を更新します。また、可能な限りデータベースへの接続にネイティブ ドライバーを使用します。ネイティブ ドライバーが使用できないときは、汎用 ODBC アダプターを使用します。データベースへのすべての通信は、これらのドライバーを通じてルーティングされます。この場合、標準以外のポートで通信するように、またはトランスポートの暗号化を提供するようにドライバーを構成する作業は、ネイティブ ドライバーのインストール作業に含まれます。このタイプの構成は Tableau に対して透過的です。

ユーザーが Tableau Server で外部データソースの認証資格情報を保存している場合、それらの情報は Tableau Server の内部データベースに暗号化して保存されます。プロセスがそれらの認証資格情報を使用して外部データソースをクエリする場合、プロセスは内部データベースから暗号化された認証資格情報を取得して、進行中に暗号を解読します。

Tableau Server からインターネットへ

場合によっては Tableau マップ サーバーなどの外部データ ソースにユーザーが接続し、Tableau Server がインターネットに接続する必要があります。Tableau のすべてのコンポーネントを保護されているネットワーク内で実行することをお勧めします。したがって、インターネットに接続するには、フォワードプロキシを使用するようTableau Server の設定が必要な場合があります。

Tableau Server からSMTP サーバーへ

Tableau Server を構成して、イベント通知を管理者やユーザーに送信できます。バージョン 2019.4 の時点で、Tableau Server は SMTP 接続の TLS をサポートしています。SMTP セットアップの構成を参照してください。

リポジトリとの通信

Postgres リポジトリと他のサーバー コンポーネントとの間のすべてのトラフィックに対して SSL (セキュア ソケット レイヤー) 暗号化通信を使用するように Tableau Server を構成できます。デフォルトでは、SSL はサーバー コンポーネントとリポジトリとの間の通信について無効になっています。

詳細については、tsm security repository-ssl enableを参照してください

クラスタ内のサーバー コンポーネント通信

分散サーバーのインストールで Tableau Server コンポーネント間の通信には、信頼と送信という2つの要素があります。Tableau クラスタ内の各サーバーは、厳格な信頼モデルを使用して、クラスタ内のその他のサーバーから有効な要求を受け取っていることを確認します。ゲートウェイ プロセスを実行しているクラスタ内のコンピューターは、負荷分散装置が転送 (この場合、負荷分散装置が要求を受け取ります) しない限り、サード パーティ(クライアント) からの要求を受け入れます。ゲートウェイ プロセスを実行していないサーバーは、クラスタのその他の信頼できるメンバーからの要求のみを受け入れます。信頼は、IP アドレス、ポート、およびプロトコルの承認リストによって確立されます。これらのうちいずれかが無効である場合、要求は無視されます。クラスタのすべてのメンバーは互いに通信できます。

ユーザーが Tableau Server で外部データ ソースの認証資格情報を保存している場合、それらの情報は Tableau Server の内部データベースに暗号化して保存されます。プロセスがそれらの認

証資格情報を使用して外部データソースをクエリする場合、プロセスは内部データベースから暗号化された認証資格情報を取得して、進行中に暗号を解読します。

クリックジャック保護

Tableau Server にはクリックジャック攻撃に対する保護が含まれています。*クリックジャック*とは、無関係のページ上の透明なレイヤーに攻撃するページを表示することで、攻撃者がユーザーにクリックさせたり、コンテンツを入力させようとする、Web ページに対する攻撃の一種です。Tableau Server との関連では、攻撃者はユーザーの認証資格情報を取得したり、認証されたユーザーを取得してサーバー上の設定を変更したりするためにクラックジャック攻撃を使用する可能性があります。クラックジャック攻撃の詳細については、Open Web Application Security Project の Web サイト上の「クリックジャック」を参照してください。

> 注:クリックジャック保護は、Tableau Server の以前のバージョンでは利用できましたが、既定では無効になっていました。Tableau Server 9.1 以降の新しいインストールでは、クリックジャック保護は明示的に無効にしない限り常にオンになります。

クリックジャック保護の効果

Tableau Server でクリックジャック保護が有効になっている場合、Tableau Server から読み込まれたページの動作は次のように変化します。

- Tableau Server は、`X-Frame-Options: SAMEORIGIN` ヘッダーをサーバーから特定の応答に追加します。ほとんどのブラウザーの最新バージョンでは、このヘッダーによりコンテンツが **<iframe>** 要素に読み込まれないように防止されるので、クリックジャック攻撃を防ぐのに役立ちます。
- 最上位のページは Tableau Server から **<iframe>** 要素に読み込むことができません。これには、サインインページが含まれています。結果的に、作成したアプリケーションで Tableau Server のページをホストできなくなる場合があります。
- ビューのみ埋め込むことができます。

- 埋め込みビューがデータソースの認証資格情報を必要とする場合、ユーザーが安全に認証資格情報を入力できるような安全なウィンドウでビューを開くためのリンクと共にメッセージが **<iframe>** 要素に表示されます。ユーザーは認証資格情報を入力する前に、開いたウィンドウのアドレスを常に確認する必要があります。

- この例のように、クエリ文字列に `:embed=y` パラメーターを含んでいる場合のみビューを読み込むことができます。

  ```
  http://<server>/views/Sales/CommissionModel?:embed=y
  ```

  > **注**:クリックジャック保護が有効になっている場合、ブラウザーのアドレスバーからコピーされた URL を使用する埋め込みビューが読み込まれない場合があります。サーバー名の後にナンバー記号 (#) を含むビューの URL (たとえば、`http://myserver/#/views/Sales/CommissionModel?:embed=y`) は通常、Tableau Server でクリックジャック保護が有効化されるとブロックされます。

クリックジャック保護の無効化

ユーザーによる Tableau Server の使用に影響が及ばない限り、クリックジャック保護を有効のままにすることをお勧めします。クリックジャック保護を無効にするには、次の `tsm` コマンドを使用します。

1. ```
   tsm configuration set -k wgserver.clickjack_defense.enabled -v false
   ```

   ```
   tsm pending-changes apply
   ```
2.

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場
   ```

合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

HTTP 応答ヘッダー

Tableau Server では、OWASP Secure Code Project で仕様が作成された応答ヘッダーの一部がサポートされています。

このトピックでは、次の応答ヘッダーを Tableau Server 用に構成する方法について説明します。

- HTTP Strict Transport Security (HSTS)
- Referrer-Policy
- X-Content-Type-Options
- X-XSS-Protection

Tableau Server では、コンテンツ セキュリティ ポリシー (CSP) 標準もサポートされています。CSP の構成については、このトピックでは説明しません。コンテンツ セキュリティ ポリシーを参照してください。

応答ヘッダーの構成

応答ヘッダーはすべて、tsm configuration set コマンドを使用して構成します。

応答ヘッダーの構成が完了したら、tsm pending-changes apply を実行します。

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

HTTP Strict Transport Security (HSTS)

HSTS は、Tableau Server に接続するクライアントに HTTPS 接続を強制するものです。詳細については、OWASP の「HTTP Strict Transport Security (HSTS)」エントリを参照してください。

オプション

gateway.http.hsts

既定値: `false`

HTTP Strict Transport Security (HSTS) ヘッダーによって、有効になっているドメインの HTTPS がブラウザーで強制的に使用されます。

gateway.http.hsts_options

既定値: `"max-age=31536000"`

既定では、HSTS ポリシーが 1 年に設定されています (31536000 秒)。ブラウザーが HTTPS 経由でサーバーにアクセスする期間を指定します。

Referrer-Policy

2019.2 から、Tableau Server には Referrer-Policy HTTP ヘッダーの動作を構成する機能が含まれています。このポリシーは、すべての "secure as" 接続で起点の URL を含める既定の動作 (ポリシー `no-referrer-when-downgrade`) を指定して有効になっています。以前のバージョンでは、Referrer-Policy ヘッダーは Tableau Server から送信された応答に含まれていませんでした。Referrer-Policy がサポートする様々なポリシー オプションの詳細については、OWASP の「Referrer-Policy」エントリを参照してください。

オプション

gateway.http.referrer_policy_enabled

既定値: `true`

Tableau Server から送信される応答から Referrer-Policy ヘッダーを除外するには、この値を `false` に設定します。

gateway.http.referrer_policy

既定値: `no-referrer-when-downgrade`

このオプションでは、Tableau Server の参照元に関するポリシーを定義します。OWASP ページの Referrer-Policy の表に記載されている任意のポリシー値文字列を指定できます。

X-Content-Type-Options

X-Content-Type-Options の応答 HTTP ヘッダーによって、Content-Type ヘッダーの MIME タイプがブラウザーで変更されないように指定されます。MIME タイプが指定されていない場合、ブラウザーではペイロードの特性を評価して MIME タイプを判断しようとすることがあります。次に、ブラウザーにコンテンツが表示されます。このプロセスは「スニッフィング」と呼ばれます。MIME タイプを誤って解釈すると、セキュリティの脆弱性を招く恐れがあります。

詳細については、OWASP の「X-Content-Type-Options」エントリを参照してください。

オプション

gateway.http.x_content_type_nosniff

既定値: `true`

X-Content-Type-Options HTTP ヘッダーは、既定でこのオプションを使用して 'nosniff' に設定されます。

X-XSS-Protection

HTTP X-XSS-Protection 応答ヘッダーがブラウザーに送信され、クロスサイトスクリプティング (XSS) 保護が有効になります。X-XSS-Protection 応答ヘッダーは、ユーザーがブラウザーで XXS 保護を無効にした場合に構成を上書きします。

詳細については、OWASP の「X-XSS-Protection」エントリを参照してください。

オプション

gateway.http.x_xss_protection

既定値: `true`

既定では、X-XSS-Protection 応答ヘッダーがこのオプションを使用して有効になります。

コンテンツ セキュリティ ポリシー

Tableau Server では、コンテンツ セキュリティ ポリシー (CSP) 標準をサポートしています。CSP は、クロスサイト スクリプティングやその他の悪意のある Web ベースの攻撃に対する追加のセキュリティ レイヤーになることを目的としています。CSP は HTTP 応答ヘッダーとして実装され、スクリプトやイメージなどの外部リソースを安全に読み込むことができる場所を指定できます。

CSP の詳細については、Mozilla の Web サイトを参照してください。

CSP の構成と有効化

CSP は、tsm configuration set のオプション コマンドを使用して構成され、有効化されます。分散展開で Tableau Server を実行している場合は、クラスタ内の最初のノードでこれらのコマンドを実行します。tsm pending-changes apply を実行すると、クラスタに構成が適用されます。

ステップ 1: 既定のディレクティブの設定

Tableau Server には以下の表に記載された一連の既定のディレクティブが含まれています。

ディレクティブを設定するには、次の tsm 構文を使用します。

```
tsm configuration set -k content_security_
policy.directive.<directive_name> -v "<value>"
```

たとえば、connect_src ディレクティブを設定するには、次のコマンドを実行します。

```
tsm configuration set -k content_security_policy.directive.connect_
src -v "* unsafe-inline"
```

オプション	既定値	説明
content_security_ policy.directive.default_src	'none'	その他のフェッチ ディレクティブのフォールバックとして機能します。 default_src の有効な値

content_security_policy.directive.connect_src	*	スクリプトインターフェイスを使用して読み込むことのできるURL を制限します。 connect_src の有効な値
content_security_policy.directive.script_src	*	JavaScript の有効なソースを指定します。 script_src の有効な値
content_security_policy.directive.style_src	* 'unsafe-inline'	スタイルシートの有効なソースを指定します。 style_src の有効な値
content_security_policy.directive.img_src	* data:	イメージおよびファビコンの有効なソースを指定します。 img_src の有効な値
content_security_policy.directive.font_src	* data:	@font-face を使用して読み込まれるフォントの有効なソースを指定します。 font_src の有効な値
content_security_policy.directive.frame_src	* data:	<frame> や <iframe> などの要素を使用して読み込む、ネストされたブラウジング コンテキストの有効なソースを指定します。 frame_src の有効な値
content_security_	data:	<object>、<embed>、および

policy.directive.object_src		<applet> 要素の有効なソースを指定します。 object_src の有効な値
content_security_policy.directive.report_uri	/vizql/csp-report	ユーザー エージェントに、CSP に対する違反行為の試行を報告するよう指示します。これらの違反レポートは、指定された URI に対する HTTP POST 要求を介して送信された JSON ドキュメントで構成されます。 report_uri の有効な値

ステップ 2: ディレクティブの追加 (オプション)

Tableau Server に含まれる既定のディレクティブは、CSP でサポートされているディレクティブのサブセットです。

サポートされる CSP ディレクティブの完全なリストについては、https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy にアクセスしてください。

既存の既定セットにディレクティブを追加するには、content_security_policy.directive 名前空間に新しいディレクティブを追加します。新しいディレクティブを追加するときは、--force-keys パラメーターを含める必要があります。構文は、次のとおりです。

```
tsm configuration set -k content_security_policy.directive.<new_
directive_name> -v "<value>" --force-keys
```

たとえば、worker-src ディレクティブを追加するには、次のコマンドを実行します。

```
tsm configuration set -k content_security_policy.directive.worker-
src -v "none" --force-keys
```

ステップ 3: レポート専用ディレクティブの指定 (オプション)

CSP を構成していくつかのディレクティブを報告したり他のディレクティブを実行したりできます。content_security_policy.enforce_enabled を true に設定すると、すべてのディレクティブが実行されます (content_security_policy.report_only_enable が同様に true に設定されている場合でも適用されます)。

ディレクティブを「レポート専用」として指定し、実行しないように指定するには、ディレクティブを report_only_directive 名前空間に追加します。新しいディレクティブを追加するときは、`--force-keys` パラメーターを含める必要があります。構文は、次のとおりです。

```
tsm configuration set -k content_security_policy.report_only_
directive.<directive_name> -v "<value>" --force-keys
```

たとえば、`script_src` ディレクティブでレポート専用にするには、次のコマンドを実行します。

```
tsm configuration set -k content_security_policy.report_only_
directive.script_src -v " http://*.example.com" --force-keys
```

ステップ 4: Tableau Server での CSP の有効化

ディレクティブを構成したら、Tableau Server で CSP を有効にします。

次のオプションを使用して、設定したディレクティブの実施やレポート専用モードを有効にします。

オプション	既定値	説明
content_security_policy.enforce_enabled	false	CSP ヘッダーをすべての要求に追加して、すべての違反がブラウザーで実施されるようにします。
content_security_policy.report_only_enabled	true	CSP ヘッダーをすべての要求に追加して、すべての違反が vizql-client ログに記録されるようにしますが、ブラウザーで

		実施されることはありません。

指定したCSP ディレクティブの実施を有効にするには、次のコマンドを実行します。

```
tsm configuration set -k content_security_policy.enforce_enabled -v
true
```

ステップ 5: tsm pending-changes apply の実行

CSP の構成が完了したら、tsm pending-changes apply を実行します。

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

CSP レポートの表示

指定したviz にCSP 違反を表示するには、開発者ツールを含むブラウザーにviz を読み込みます。この例では Chrome ブラウザーを使用します。

1. CSP を構成したTableau Server 展開でホストされている、違反のあるテストviz を読み込みます。
2. `CTRL+Shift+I` を押してChrome で開発者ツールを開きます。
3. **[ネットワーク]** タブをクリックします。
4. **[フィルター]** フィールドで、`csp-report` と入力してから**[すべて検索]** をクリックします。

- 違反がない場合は、検索で CSP レポートが返されません。
- 違反がある場合は、結果ペインの [ヘッダー] タブをクリックして下部までスクロールし、**[リクエスト ペイロード]** を表示します。

SSL

SSL (Secure Sockets Layer) は Web サーバーとクライアントの間で暗号化されたリンクを確立する標準セキュリティ技術です。SSL を使用するには、Tableau Server に SSL 証明書をインストールする必要があります。

重要: バージョン 2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキーペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

Tableau Server を構成して、以下の方法で SSL を使用できます。

- 外部 HTTP トラフィックに対して SSL を使用する。
- クライアント (Tableau Desktop、Web ブラウザー、tabcmd.exe) と Tableau Server の間で相互 (2 方向) SSL を使用する。
- 内部サーバー コンポーネントとリポジトリの間ですべての HTTP トラフィックに対して SSL を使用する。

相互 SSL を使用している場合、各クライアントも証明書が必要となります。

注: Tableau Server では、ユーザー認証に SSL を使用します。SSL は、Tableau Server でホストされるコンテンツ (データ ソースおよびワークブック) のパーミッションや認証の処理には使用されません。

Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する

すべての外部 HTTP トラフィックに対して SSL 暗号化通信を使用するように Tableau Server を構成できます。SSL を設定すると、Tableau Server へのアクセスのセキュリティを確保することができ、サーバーと Tableau クライアント (Tableau Desktop、REST API 分析拡張機能など) の間で渡される機密情報を確実に保護できます。このトピックでは、サーバーが SSL を使用するように構成する方法の手順が説明されています。まず、認証機関から証明書を入手し、この証明書ファイルを Tableau Server にインポートする必要があります。

相互 SSL 認証は Tableau Mobile ではサポートされていません。

SSL 証明書要件

Apache SSL 証明書を認証機関 (Verisign、Thawte、Comodo、GoDaddy など) から入手します。自社が発行する社内証明書も使用できます。SSL を同じドメイン内の多数のホスト名で使用できる、ワイルドカード証明書もサポートされています。

Tableau Server との双方向の外部通信を行うために SSL 証明書を入手する場合は、次のガイドラインおよび要件に従います。

- 証明書ファイルはすべて、拡張子 `.crt` が付いた有効な PEM 暗号化 X509 証明書である必要があります。
- SHA-2 (256 または 512 ビット) SSL 証明書を使用します。ほとんどのブラウザーでは SHA-1 証明書を提示するサーバーに接続できなくなっています。

  > **重要:** バージョン 2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキーペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

- 証明書ファイルに加えて、対応する SSL 証明書キーファイルも取得する必要があります。証明書キーファイルは、有効な RSA または DSA 秘密キーファイル (規則により拡張子

.key を付加) である必要があります。

キー ファイルをパスフレーズで保護することもできます。構成時に入力するパスフレーズが保存中に暗号化されます。ただし、SSL とSAML に同じ証明書を使用する場合は、パスフレーズで保護 *されていない*キー ファイルを使用する必要があります。

キー ファイルがパスフレーズで保護されている場合は、関連する暗号アルゴリズムが実行中の Tableau Server のバージョンでサポートされていることを確認する必要があります。Tableau Server は OpenSSL を使用して、パスワードで保護されたキー ファイルを開きます。

Tableau Server (2022.1.17、2022.3.9、2023.1.5 以降) では OpenSSL を実行します。2025.1.x までのバージョンでは、OpenSSL 3.1 を実行します。バージョン 2025.3 以降、Tableau はレベル 2 のセキュリティキー (2048 ビット以上のキー) を使用する OpenSSL 3.4.0 を使用します。

- SSL 証明書チェーン ファイル: 証明書チェーン ファイルは、Mac の Tableau Desktop と、Mac および Windows の Tableau Prep Builderに必要です。また、Tableau Server の証明書チェーンがモバイル デバイスの iOS または Android オペレーティング システムによって信頼されていない場合は、Tableau Mobile アプリのチェーン ファイルも必要です。

  チェーン ファイルは、サーバー証明書の証明書チェーンを構成するすべての証明書を連結したものです。ファイル内のすべての証明書に PEM 暗号化 x509 証明書を指定し、ファイルには .pem ではなく.crt 拡張子を付ける必要があります。

- 複数のサブドメインがある場合、Tableau Server ではワイルドカード証明書もサポートされています。

- クライアントが Tableau Server への接続に使用するドメイン、ホスト名、または IP アドレスが [サブジェクトの別名 (SAN)] フィールドに含まれていることを確認します。多くのクライアント (Tableau Prep、Chrome、Firefox ブラウザーなど) では、安全な接続を確立するために SAN フィールドに有効な値を入力する必要があります。

注: Tableau Server で SAML を使用したシングル サインオンを構成することを計画している場合には、SAML 要件のSAML での SSL 証明書およびキー ファイルの使用を参照し、SSL とSAML の両方で同一の証明書ファイルを使用するか判断してください。

クラスタの SSL を構成

Tableau Server クラスタを構成して SSL を使用することができます。最初のノードがゲートウェイ プロセスを実行する唯一のノードである場合 (既定で実行)、そのノードでのみ SSL を設定する必要があります。手順については、このトピックを参照してください。

複数ゲートウェイのある SSL

可用性の高い Tableau Server クラスタには、ロード バランサーが転送した複数のゲートウェイを含めることができます (詳細について)。このタイプのクラスタで SSL を構成する場合、次の方法があります。

- **SSL 用のロード バランサーを構成する**: クライアント Web ブラウザーからロード バランサーへのトラフィックが暗号化されます。ロード バランサーから Tableau Server ゲートウェイ プロセスへのトラフィックは暗号化されません。ユーザーが Tableau Server で SSL 構成を行う必要はありません。この設定はすべてロード バランサーが行います。
- **Tableau Server で SSL を設定する**。クライアント Web ブラウザーからロード バランサーへのトラフィック、およびロード バランサーから Tableau Server ゲートウェイ プロセスへのトラフィックが暗号化されます。詳細については、次のセクションに進んでください。

Tableau Server クラスター環境の追加構成情報

ゲートウェイプロセスを実行するすべての Tableau Server ノードで SSL を使用したいときは、以下の手順を実行します。

1. SSL パススルー用の外部ロード バランサーを構成します。

   443 番以外のポートを使用する場合は、クライアントから標準以外のポートを切り離すように外部ロード バランサーを構成できます。このシナリオでは、443 番ポートで Tableau

Server に接続するようにロードバランサーを構成します。サポートについては、ロードバランサーに提供されている文書を参照してください。

2. SSL 証明書がロードバランサーのホスト名に発行されていることを確認します。
3. 初期 Tableau Server ノードの SSL を設定します。
4. 相互 SSL を使用している場合は、SSL CA 証明書ファイルをアップロードします。`tsm authentication mutual-ssl <commands>` を参照してください。

構成プロセスの一部として、SSL 証明書とキーファイルが各ノードに配布されます。

環境の準備

CA から証明書ファイルを取得するときに、それらを Tableau Server からアクセス可能な場所に保存し、証明書の .crt および .key ファイルの名前とそれらの保存場所を書き留めます。これらの情報は、SSL を有効化するときに Tableau Server に提供する必要があります。

Tableau Server での SSL の構成

最も精通している方法を使用します。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[セキュリティ] > [外部 SSL]** を選択します。

   > **注:** 既存の構成を更新または変更する場合は、**[リセット]** をクリックして既存の設定をクリアしてから続行してください。

3. **[外部 Web サーバーの SSL]** で **[SSL でサーバー通信を有効にする}** を選択します。

4. 証明書ファイルおよびキー ファイルをアップロードし、お使いの環境に必要な場合には、チェーン ファイルをアップロードして次のパスフレーズ キーを入力します。

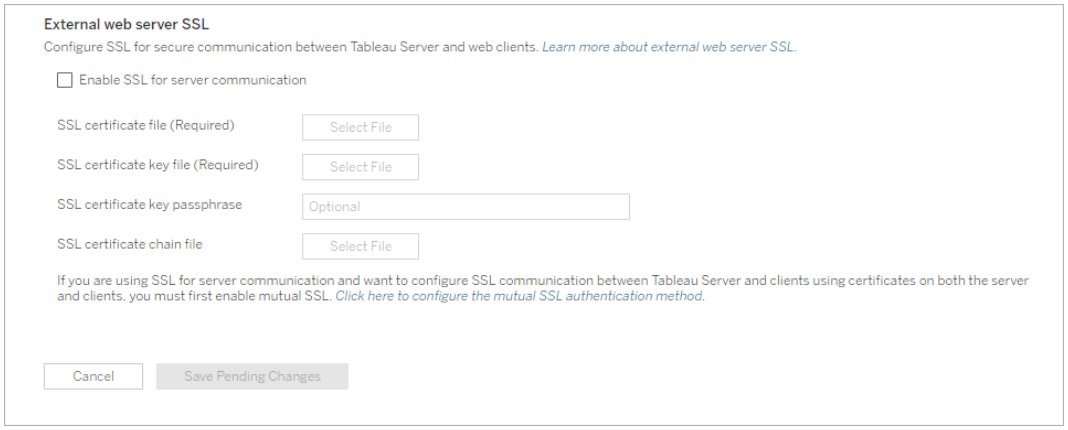


Tableau Server を分散展開で実行している場合は、これらのファイルがクラスタ内の適切なノードのそれぞれに自動的に配布されます。

5. **[保留中の変更を保存]** をクリックします。

6. ページ上部の **[変更を保留中]** をクリックします。

7. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

証明書ファイルをローカル コンピューターにコピーし、以下のコマンドを実行します。

```
tsm security external-ssl enable --cert-file <path-to-file.crt> --
key-file <path-to-file.key>

tsm pending-changes apply
```

tsm security external-ssl enable でコマンド リファレンスを参照して、`external-ssl enable` の追加オプションを含めるかどうかを判断してください。Tableau は `--protocols` オプションを含めることをお勧めします。

`external-ssl enable command` は、.crt および .key ファイルから情報をインポートします。このコマンドを Tableau Server クラスター内の 1 つのノードで実行すると、他のゲートウェイ ノードにも情報が分配されます。

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

ポート リダイレクトとロギング

SSL 用にサーバーを構成すると、非 SSL ポート(既定のポートは 80) に対する要求は受け入れられ、SSL ポート443 へ自動的にリダイレクトされます。

> **注:** Tableau Server では、ポート443 のみがセキュア ポートとしてサポートされています。別のアプリケーションがポート443 を使用しているコンピュータ上で Tableau Server を実行することはできません。

SSL エラーは次の場所に記録されます。検証や暗号化に関する問題をトラブルシューティングするには、このログを使用してください。

```
/var/opt/tableau/tableau_server/data/tabsvc/logs/httpd/error.log
```

SSL ポートのローカル ファイアウォールへの追加

ローカル ファイアウォールを実行している場合は、SSL ポートを Tableau Server 上のファイアウォールに追加する必要があります。以下の例は、RHEL/CentOS ディストリビューション上で稼働するファイアウォールの設定方法を説明しています。例では CentOS の既定ファイアウォールである Firewalld が使用されています。

1. Firewalld を起動します。

   ```
   sudo systemctl start firewalld
   ```

2. SSL 用のポート 443 を追加します。

   ```
   sudo firewall-cmd --permanent --add-port=443/tcp
   ```

3. ファイアウォールをリロードして、設定を検証します。

   ```
   sudo firewall-cmd --reload
   ```

   ```
   sudo firewall-cmd --list-all
   ```

SSL 証明書の変更または更新

SSL を構成した後、証明書を定期的に更新しなければならないこともあります。場合によっては、IT 環境の運用上の変更に合わせて証明書の変更が必要になることもあります。いずれの場合も、TSM を使用して、外部 SSL 用に既に構成されている SSL 証明書を置き換える必要があります。

オペレーティング システムのファイル ディレクトリに新しい証明書を直接コピーしないでください。代わりに、TSM Web UI または `tsm security external-ssl enable` コマンドを使用して証明書を追加すると、証明書ファイルが適切な証明書ストアにコピーされます。分散展開では、証明書はクラスター内のノード間でもコピーされます。

SSL 証明書を (必要に応じて、対応するキー ファイルと共に) 変更または更新するには、このトピックの前の方にあるセクション「Tableau Server での SSL の構成」の手順に従います。

証明書を変更した後、`tsm pending-changes apply` を実行して、Tableau Server サービスを再起動する必要があります。また、SSL 証明書を使用するコンピューター上の他のサービスを再起動することをお勧めします。オペレーティング システムでルート証明書を変更する場合は、コンピュータを再起動する必要があります。

例: SSL 証明書 - キーとCSR の生成

重要: この例は、SSL 要件や構成の経験がある IT プロフェッショナル向けに全般的なガイドを提供するためのものです。この記事で説明する手順は、必要なファイルの生成に使用できる多くのメソッドの 1 つにすぎません。ここで説明するプロセスは一例であり、推奨されるものではありません。

セキュア ソケット レイヤー (SSL) 暗号化を使用するようTableau Server を構成すると、サーバーへのアクセスがセキュアにし Tableau Server とTableau Desktop の間で送信されたデータが保護するのに役立ちます。

> Tableau Server on Windows については、「例: SSL 証明書 - キーとCSR の生成」を参照してください。

Tableau Server は OpenSSL が含まる Apache を使用します。OpenSSL ツールキットを使用してキー ファイルと証明書署名要求 (CSR) を生成できます。その後、それらを使用して署名付き SSL 証明書を取得できます。

Tableau Server バージョン 2022.1.17、2022.3.9、2023.1.5、および 2025.1.x では OpenSSL 3.1 が実行されます。

> **重要:** バージョン 2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキー ペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

キーおよび CSR の作成手順

SSL を使用するようTableau Server を構成するには、SSL 証明書が必要です。SSL 証明書を取得するには、次の手順を完了します:

1. キーファイルを生成する。
2. 証明書署名要求 (CSR) を作成する。
3. CSR を認証局 (CA) に送信して SSL 証明書を取得する。
4. キーおよび証明書を使用して、SSL を使用するように Tableau Server を構成する。

詳細については、Apache Software Foundation Web サイトの SSL FAQ ページで見つけることができます。

複数のドメイン名に対する証明書の設定

Tableau Server は、複数ドメイン用に SSL を許可します。この環境を設定するには、OpenSSL 構成ファイル、openssl.conf を変更し、Tableau Server でサブジェクトの別名 (SAN) 証明書を構成する必要があります。以下の「SAN 証明書の場合、OpenSSL 構成ファイルを変更する」を参照してください。

キーの生成

証明書署名要求の生成に使用するキーファイルを生成します。

1. 次のコマンドを実行して、キーファイルを作成します。

   ```
   openssl genrsa -out <yourcertname>.key 4096
   ```

   注:

   - このコマンドは、キーに 4096 ビット長を使用します。短いビット長で暗号化されている通信は安全性が低いため、2048 ビット以上のビット長を選択する必要があります。値が提供されない場合は、512 ビットが使用されます。
   - Tableau Server バージョン 2021.3.26、2021.4.21、2022.1.17、2022.3.9、2023.1.5 以降で PKCS#1 RSA キーを作成するには、OpenSSL 3.1 に基づいて `openssl genrsa"` コマンドを実行するときに、追加オプション `-traditional` を使用する必要があります。オプションの詳細については、https://www.openssl.org/docs/man3.1/man1/openssl-rsa.html を参照してください。

証明書署名要求を作成して認証局に送信する

上記の手順で作成したキーファイルを使用して、証明書署名要求 (CSR) を生成します。CSR を認証局 (CA) に送信して署名入り証明書を取得します。

重要: 複数ドメインに SSL を使用するようSAN 証明書を構成する場合は、最初に以下の「SAN 証明書の場合、OpenSSL 構成ファイルを変更する」の手順を完了してから、ここに戻って CSR を生成してください。

1. 次のコマンドを実行して、証明書署名要求 (CSR) ファイルを作成します。

   ```
   openssl req -new -key yourcertname.key -out yourcertname.csr  -
   config /opt/tableau/tableau_
   server/packages/apache.<version>/conf/openssl.cnf
   ```

2. プロンプトで要求されたら、必要な情報を入力します。

   注: [共通名] として、Tableau Server 名を入力します。Tableau Server 名は Tableau Server に到達するのに使用される URL です。たとえば、ブラウザーのアドレスバーに「`tableau.example.com` 」と入力することで Tableau Server に到達する場合、`tableau.example.com` が共通名になります。共通名によってサーバー名が解決されない場合、ブラウザーまたは Tableau Desktop が Tableau Server への接続を試みると、エラーが発生します。

CSR を認証局に送信して SSL 証明書を取得する

CSR を商業認証局 (CA) に送信してデジタル証明書を要求します。詳細については、Wikipedia の記事「認証局」や、使用する CA の決定に役立つ関連記事を参照してください。

キーおよび証明書を使用して Tableau Server を構成する

キーとCA からの証明書の両方がある場合は、SSL を使用するようTableau Server を構成できます。手順については、「外部 SSL の構成」を参照してください。

SAN 証明書の場合、OpenSSL 構成ファイルを変更する

OpenSSL の標準インストールでは、一部の機能が既定で有効化されていません。複数のドメイン名で SSL を使用するには、CSR を生成する前に、これらの手順を実行して **openssl.cnf** ファイルを変更します。

1. Tableau Server の Apache **conf** フォルダーに移動します。

   例: `/opt/tableau/tableau_server/packages/apache.<version_code>/conf`

2. テキストエディターで **openssl.cnf** を開き、次の行を見つけます: `req_extensions = v3_req`

   行の先頭にハッシュ記号 (#) を付けてこの行をコメントアウトさせることができます。

   ```
   UTF8Strings
   # so use this option with caution!
   string_mask = nombstr

   # req_extensions = v3_req # The extensions to add to a
   certificate request

   [ req_distinguished_name ]
   countryName                 = Country Name (2 letter code)
   ```

   行がコメントアウトされている場合は、行の先頭に **# とスペース**を削除します。

3. ファイルの **[v3_req]** セクションに移動します。最初の数行には、次のテキストが含まれます。

   ```
   # Extensions to add to a certificate request
   basicConstraints = CA:FALSE
   keyUsage = nonRepudiation, digitalSignature, keyEncipherment
   ```

keyUsage 行の後に、次の行を挿入します。

```
subjectAltName = @alt_names
```

自己署名 SAN 証明書を作成している場合は、以下を実行して、証明書に署名する証明書パーミッションを付与します。

a. `cRLSign` および `keyCertSign` を **keyUsage** 行に追加し、次のようにします:

```
keyUsage = nonRepudiation, digitalSignature,
keyEncipherment, cRLSign, keyCertSign
```

b. **keyUsage** 行の後に、次の行を挿入します。`subjectAltName = @alt_names`

4. **[alt_names]** セクションに、SSL で使用するドメイン名を入力します。

```
DNS.1 = [domain1]
DNS.2 = [domain2]
DNS.3 = [etc]
```

次のイメージは、ハイライトされた結果と、ドメイン名と置き換えられるプレースホルダーテキストを示しています。

```
[ v3_req ]

# Extensions to add to a certificate request

basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectAltName = @alt_names

[alt_names]
DNS.1 = domain1
DNS.2 = domain2
DNS.3 = etc
```

5. ファイルを保存して閉じます。

6. 上記の「証明書署名要求を作成して認証局に送信する」セクションの手順を完了します。

内部 Postgres 通信用に SSL を設定する

Postgres リポジトリと他のサーバー コンポーネントとの間の暗号化通信に SSL (TLS) を使用するようTableau Server を構成することができます。既定では、Tableau Server コンポーネントへの内部通信は暗号化されません。

内部 SSL のサポートを有効化する一方で、Tableau Desktop、Tableau Mobile、REST API、Web ブラウザーなどの Tableau クライアントからリポジトリへの直接接続のサポートを構成することもできます。

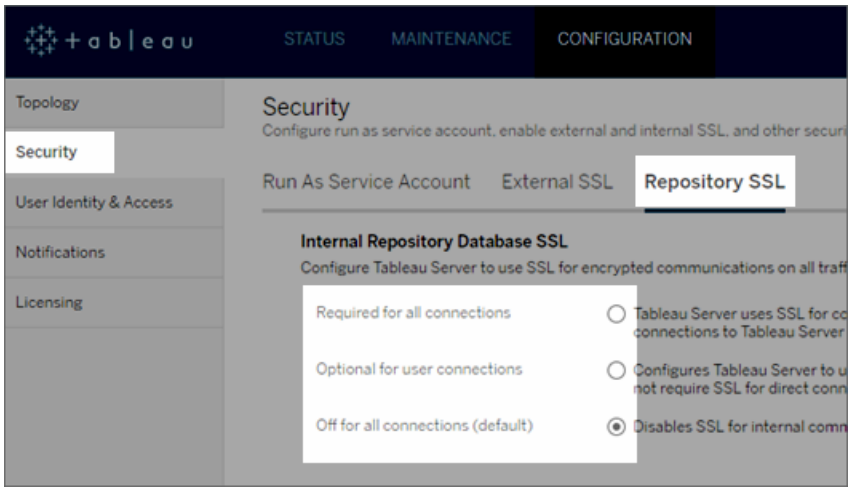
TSM Web インターフェイスの使用

1. サーバー管理者がブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[セキュリティ] > [リポジトリ SSL]** を選択します。



3. リポジトリSSL で使用するオプションを 1 つ選択します。

- **すべての接続で必要** — Tableau Server の内部通信に SSL を使用します。また、**tableau** や **readonly** ユーザーを使用しているクライアントなど、postgres リポジトリに直接接続する Tableau クライアントや外部クライアント (Tableau 以外) では SSL が必要です。

  > **重要:** クライアントからの直接接続を許可するよう Postgres SSL を構成するのステップを完了して、クライアントコンピューター上の正しい場所に証明書ファイルを配置しなければ、Tableau クライアントと外部 postgres クライアントは、クライアントコンピューター上の証明書とリポジトリコンピューターからの SSL 証明書を比較して Tableau リポジトリの ID を検証することができません。

- **ユーザー接続のオプション** — この機能を有効にすると、Tableau Server の内部通信に SSL が使用されます。Tableau クライアントや外部クライアントからサーバーへの直接接続では SSL がサポートされますが、必須ではありません。

- **すべての接続でオフ (既定)** — サーバーの内部通信は暗号化されておらず、SSL はクライアントからの直接接続に必要ではありません。

4. **[OK]** をクリックします。

   最初の 2 つのオプションでサーバーの証明書ファイルである **server.crt** と **server.key** が生成され、以下の場所に置かれます。

   ```
   /var/opt/tableau/tableau_server/data/tabsvc/config/pgsql_
   <version>/security
   ```

   直接接続にクライアントを構成する必要がある場合には、この .crt ファイルを使用します。

TSM CLI の使用

サーバーコンポーネント間の内部トラフィック用に SSL を有効にするには、次のコマンドを実行します。

```
tsm security repository-ssl enable
```

```
tsm pending-changes apply
```

コマンドの実行内容

repository-ssl enable によりサーバーの証明書ファイルが生成されます。ファイルは次の場所に配置されます。

```
/var/opt/tableau/tableau_server/data/tabsvc/config/pgsql_
<version>/security
```

既定では、このコマンドは、リポジトリと他のサーバー コンポーネントとの間のトラフィック、および Tableau クライアントとの直接接続 (**tableau** ユーザーまたは **readonly** ユーザーを介した接続を含む) の場合に SSL を必要とするように Tableau Server を設定します。

この構成を完了するには、クライアントからの直接接続を許可するようPostgres SSL を構成するで説明されている手順も完了し、証明書ファイルをクライアント コンピューターの正しい場所に置く必要があります。

保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

repository-ssl enable のオプション

Tableau Server 内部通信にのみ SSL が必要であり、クライアント アプリとの直接接続には必要ない場合、repository-ssl enable コマンドを以下のオプションと共に使用します。

```
--internal-only
```

クラスタ環境

クラスタ内のノードで `repository-ssl enable` を実行する場合、必要な証明書ファイルが他のノードの同じ場所にコピーされます。

直接接続用にパブリック証明書をダウンロードする詳細については、クライアントからの直接接続を許可するようPostgres SSL を構成するを参照してください。

TSM コントローラーのカスタム SSL 証明書の設定

Tableau Server 管理コントローラー (コントローラーとも呼びます) は、Tableau Server クラスターに対する変更を管理するための管理コンポーネントです。デフォルトでは、コントローラーは Tableau Server クラスターの初期ノードで実行されます。単一の Tableau クラスター展開で複数のコントローラーを実行することは技術的に可能ですが、推奨される方法ではありません。

コントローラーには、TSM CLI、TSM Web クライアント、REST クライアント (curl、postman) などのさまざまなクライアントで管理できる API が含まれています。これらのクライアントを使用して、Tableau Server 管理者はサーバー クラスターの構成を変更できます。コントローラーは、Zookeeper と共に、ノード全体の構成変更を管理および実行します。

重要: バージョン 2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキーペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

デフォルトの TSM SSL 機能

注: TLS で HTTPS トラフィックを保護することを指す場合でも、慣例に従って「SSL」という用語を使用しています。

デフォルトでは、クライアント接続は、自己署名証明書による SSL で暗号化されます。この証明書は、セットアップ時に Tableau Server が作成したものであり、コントローラーが更新を行います。暗号化に加えて、SSL ハンドシェイク時にこの証明書で提示されたサブジェクト名に対して、コントローラーのホストマシンの ID (ホスト名または IP) の検証が行われます。ただし、証明書は自己署名であるため、証明書の信頼性は絶対的ではありません。

コントローラーへの CLI 接続の場合、中間者攻撃では通常、プライベートネットワーク内で Tableau Server クラスターへの悪意のあるユーザー アクセスが必要になるため、証明書を完全に信頼できないことは重大なセキュリティリスクではありません。CLI シナリオで悪意のあるユーザーがコントローラーの証明書を偽装できるのであれば、悪意のあるユーザーはすでに「王国への鍵」を持っていることになります。

ただし、内部ネットワークの外から TSM Web UI 経由で管理者がコントローラーに接続するシナリオでは、信頼された認証局によるホストの検証が欠如していると、セキュリティリスクが大きくなります。

最近まで、Windows マシンで TSM Web UI を実行しているお客様は、Windows の信頼できるルートストアに Tableau Server CA 証明書を配置できました。ほとんどのブラウザーは、この設定を頼りにして証明書の信頼を検証していました。現在では、Chrome は、OS の信頼できるストアに配置されている自己署名証明書をもう検証 (信頼) しなくなりました。Chrome (およびほとんどの主要なブラウザー) は、信頼できるサードパーティのルート CA にチェーン バックする証明書のみを信頼します。

Tableau Server v2023.1 SSL カスタム証明書

カスタム SSL TSM 証明書の機能は、管理者が TSM コントローラーを構成する際、信頼できるサードパーティのルート CA にチェーン バックする ID 証明書を使用することで、信頼のギャップを埋めます。

理解しておくべき重要な詳細がいくつかあります。

- TSM カスタム SSL 証明書の信頼は、TSM Web UI との接続時に検証されます。
- TSM CLI シナリオでは、信頼の検証は試行されません。前述のように、CLI シナリオでの「中間者」攻撃には、起こりそうなリスクはありません。
- 証明書チェーンは設定に含めることができます。チェーンは、中間 CA によって署名されたすべての証明書を提示できます。チェーンは任意の位置で切れる可能性があり、チェーン

にない証明書は、オペレーティング システムのトラスト ストアにインストールされていると見なされます。

設定

TSM の SSL カスタム証明書を設定する(または更新する)には、TSM CLI を使用する必要があります。

tsm security custom-tsm-ssl enable を参照してください。

クライアントからの直接接続を許可するようPostgres SSL を構成する

Tableau Server が postgres リポジトリとの内部通信に SSL を使用するように構成されている場合、内部 postgres インスタンスによって提示された SSL 証明書を Tableau または外部 postgres クライアントに配布された証明書と比較することで、リポジトリに直接接続する Tableau クライアントと外部 postgres クライアントに Tableau postgres リポジトリの ID を検証するよう要求することもできます。

直接接続には、**tableau** ユーザーまたは **readonly** ユーザーを使用する接続が含まれます。Tableau クライアントの例には、Tableau Desktop、Tableau Mobile、および REST API 、Web ブラウザーが含まれます。

1. 以下のコマンドを実行して、リポジトリの内部 SSL を有効にします。

   ```
   tsm security repository-ssl enable

   tsm pending-changes apply
   ```

   これにより、内部 SSL のサポートが有効になり、新しいサーバー証明書とキー ファイルが生成されます。また、すべての Tableau クライアントで SSL を使用したリポジトリへの接続が必要となります。追加の repository-ssl コマンドおよびオプションについては、tsm security を参照してください。

   保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのよう

にしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

2. (オプション) Postgres SSL 接続を検証するようにクライアントコンピューターを構成した場合は、Tableau Server によって生成された証明書を Tableau Desktop を実行しているコンピューターにインポートする必要があります。リポジトリに直接接続する各クライアントコンピューターで、次の操作を行います。

    - **Server.crt** ファイルをクライアントコンピューターにコピーします。このファイルは以下のディレクトリにあります。

      ```
      /var/opt/tableau/tableau_server/data/tabsvc/config/pgsql_<version_code>/security
      ```

      > **注**: クライアントコンピューターに **server.key** をコピーしないでください。このファイルはサーバー上のみに存在します。

    - コンピューターの証明書ストアに証明書をインポートします。

      詳細については、オペレーティング システム制作会社のドキュメントを参照してください。

3. (オプション) Tableau Server postgres リポジトリの ID を検証するために、(Tableau 以外の) 外部 postgres クライアント (PgAdmin または Dbeaver など) を構成します。クライアントが接続で使用している postgresql JDBC ドライバーでこれを行うには、"sslmode」ディレクティブを "verify-ca" または "verify-full" に設定します。使用する postgres ドライバーのバージョンによって、使用できるオプションが異なる場合があります。詳しくは、SSL サポートに関するドライバーのドキュメントを参照してください。

相互 SSL 認証の構成

相互 SSL を使用して、Tableau Desktop、Tableau Mobile、その他の承認済み Tableau クライアントのユーザーに Tableau Server への安全な直接アクセスを提供することができます。相互

SSL を使用すれば、有効な SSL 証明書を持つクライアントが Tableau Server に接続するときに、Tableau Server でクライアント証明書の存在を確認し、クライアント証明書内のユーザー名に基づいてユーザーを認証します。クライアントに有効な SSL 証明書がない場合、Tableau Server は接続を拒否することができます。

相互 SSL に失敗する場合、ユーザー名/パスワード認証を使用するようTableau Server を構成することもできます。さらに、フォールバック認証が構成されているかどうかに関係なく、REST API でユーザー名とパスワード(存在する場合) を使用してログインできます。

ユーザー認証セッションの時間制限

ユーザーが相互 SSL でログインすると、認証セッションは、Tableau Server のグローバル認証セッションの構成を制御するのと同じ方法で管理されます。

Web ブラウザーを使用して Tableau Server に接続するクライアントの場合、グローバル認証セッションの構成は、*セキュリティ強化チェックリスト*に記載されています。「9.セッション存続時間設定の検証を参照してください。

接続されたクライアント(Tableau Desktop、Tableau Mobile、Tableau Prep Builder、Bridge) のセッションでは、OAuth トークンを使用して、セッションを再確立することによりユーザーをログインしたままにします。既定では、OAuth クライアント トークンは 1 年経過後にリセットされます。クライアント トークンが 14 日間使用されなかった場合には有効期限切れになります。これらの値は、`refresh_token.absolute_expiry_in_seconds` および `refresh_token.idle_expiry_in_seconds` オプションの設定により変更できます。tsm configuration set のオプションを参照してください。

証明書の使用

相互 SSL を有効にして構成する前に、外部 SSL を構成する必要があります。外部 SSL は、Tableau Server をクライアントが認証するものであり、外部 SSL を構成するための証明書とキーを使用してセッションを暗号化します。

相互 SSL では、証明書ファイルがもう1 つ必要です。ファイルは CA 証明書ファイルを連結したものです。ファイルの種類は `.crt`.である必要があります。"CA" は、Tableau Server に接続するクライアント コンピューターに証明書を発行する*認証局*です。CA 証明書ファイルをアップロードするアク

ションによって信頼が確立されます。これにより、Tableau Server はクライアントコンピューターによって提示される個々の証明書を認証できます。

災害復旧計画の一環として、証明書ファイルと失効ファイル (該当する場合) のバックアップを Tableau Server 外部の安全な場所に保管することをお勧めします。Tableau Server に追加する証明書ファイルと失効ファイルは、クライアントファイル サービスによって他のノードに格納および配布されます。ただし、ファイルは復元可能な形式では格納されません。Tableau Server クライアントファイル サービスを参照してください。

RSA キーのサイズと ECDSA 曲線のサイズ

相互 SSL に使用される CA 証明書では、RSA キーの強度が 2048 であるか、ECDSA 曲線サイズが 256 である必要があります。

それぞれの構成キーを設定することによって、安全性の低いサイズを受け入れるように Tableau Server を構成できます。

- ssl.client_certificate_login.min_allowed.rsa_key_size
- ssl.client_certificate_login.min_allowed.elliptic_curve_size

tsm configuration set のオプションを参照してください。

クライアント証明書要件

相互 SSL を使用して Tableau Server に認証するユーザーは、セキュリティの最小要件を満たすクライアント証明書を提示する必要があります。

署名アルゴリズム

証明書には SHA-256 またはそれ以上の署名アルゴリズムを使用する必要があります。

SSL 相互認証用に構成された Tableau Server では、ユーザーが SHA-1 署名アルゴリズムを使用したクライアント証明書を提示すると、認証が拒否されます。

ユーザーが SHA-1 クライアント証明書を使用してログインしようとすると「サインインできません」というエラーが表示され、VizPortal ログには次のエラーが記録されます。

```
Unsupported client certificate signature detected: [certificate
Signature Algorithm name]
```

ssl.client_certificate_login.blocklisted_signature_algorithms 構成オプションを設定することにより、安全性の低い SHA-1 署名アルゴリズムを受け入れるように Tableau Server を構成できます。

RSA キーのサイズと ECDSA 曲線のサイズ

相互 SSL に使用される クライアント証明書では、RSA キーの強度が 2048 であるか、ECDSA 曲線サイズが 256 である必要があります。

これらの要件を満たしていないクライアント証明書から相互認証が要求されると、Tableau Server の認証は失敗します。それぞれの構成キーを設定することにより、安全性の低いサイズを受け入れるように Tableau Server を構成できます。

- ssl.client_certificate_login.min_allowed.rsa_key_size
- ssl.client_certificate_login.min_allowed.elliptic_curve_size

tsm configuration set のオプションを参照してください。

TSM Web インターフェイスの使用

1. Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する。
2. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。
3. **[構成]** タブで **[ユーザー ID とアクセス] > [認証方法]** を選択します。
4. **[認証方法]** のドロップダウン メニューで **[相互 SSL]** を選択します。
5. [相互 SSL] で **[クライアント証明書による相互 SSL と自動サインインを使用]** を選択します。
6. **[ファイルの選択]** をクリックし、認証局 (CA) の証明書ファイルをサーバーにアップロードします。

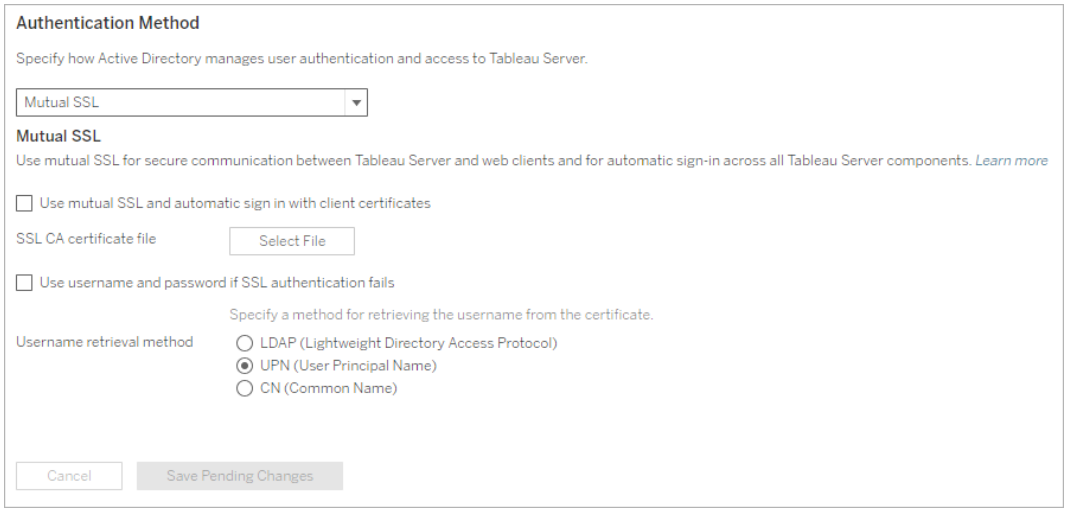
このファイル (.crt) は、クライアント認証に使用する複数の CA 証明書を含んでいるオールインワンのファイルです。アップロードするファイルは、PEM でエンコードされたさまざまな証明書ファイルを優先順に連結したものである必要があります。

7. 自分の組織の残りの SSL 構成情報を入力します。

   ユーザー名の形式: Tableau Server で相互 SSL を構成する場合、サーバーはクライアント証明書からユーザー名を取得し、クライアントユーザーの直接サインインを確立できるようにします。Tableau Server が使用する名前は、Tableau Server がユーザー認証用に設定される方法によって異なります。

   - ローカル認証 — Tableau Server は証明書から UPN (ユーザープリンシパル名) を使用します。
   - **Active Directory (AD)**—Tableau Server は LDAP (ライトウェイトディレクトリアクセスプロトコル) を使用してユーザー名を取得します。

   または、Tableau Server を設定してクライアント証明書から CN (共通名) を使用できます。



8. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

9. ページ上部の **[変更を保留中]** をクリックします。

10. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

ステップ 1: 外部サーバー通信に SSL が必要

Tableau Server とWeb クライアントの間の外部通信に SSL を使用するようTableau Server を構成するには、サーバー証明書の .crt および .key ファイルの名前を指定して、次のように `external-ssl enable` コマンドを実行します。

```
tsm security external-ssl enable --cert-file <file.crt> --key-
file <file.key>
```

- `--cert-file` および `--key-file` には、CA が発行したサーバーの SSL 証明書 (.crt) およびキー (.key) ファイルの保存場所とファイル名を指定します。
- 上記のコマンドは、ユーザーが Tableau Server 上で**サーバー管理者**サイトロールを持つユーザーとしてサインインしていることを前提としています。代わりに、`-u` および `-p` パラメーターを使用して、管理者ユーザーおよびパスワードを指定することもできます。
- 証明書キーファイルにパスフレーズが必要な場合は、`--passphrase` パラメーターと値を含めます。

ステップ 2: 相互 SSL を構成して有効にする

サーバーと各クライアント間の相互認証を追加し、Tableau クライアントユーザーが初回に認証資格情報を提供した後は直接認証されるようにします。

1. 次のコマンドを実行します。

   ```
   tsm authentication mutual-ssl configure --ca-cert <certificate-
   file.crt>
   ```

`--ca-cert` には、認証局 (CA) 証明書ファイルの場所とファイル名を指定します。

このファイル (.crt) は、クライアント認証に使用する複数の CA 証明書を含んでいるオールインワンのファイルです。アップロードするファイルは、PEM でエンコードされたさまざまな証明書ファイルを優先順に連結したものである必要があります。

2. 次のコマンドを実行して相互 SSL を有効化し、変更を適用します。

```
tsm authentication mutual-ssl enable

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

相互 SSL の追加オプション

`mutual-ssl configure` を使用して、次のオプションをサポートするようTableau Server を構成することができます。

詳細については、「tsm authentication mutual-ssl <commands>」を参照してください。

フォールバック認証

Tableau Server が相互 SSL 向けに構成されている場合、認証は自動となり、クライアントは有効な証明書を持つ必要があります。フォールバックオプションを許可し、ユーザー名およびパスワード認証を受け付けるようTableau Server を構成することができます。

```
tsm authentication mutual-ssl configure -fb true
```

上記のオプションが `false` に設定されている場合でも、Tableau Server は REST API クライアントからのユーザー名とパスワード認証を受け付けます。

ユーザー名マッピング

Tableau Server が相互 SSL 向けに構成されている場合、サーバーはクライアント証明書からユーザー名を取得することにより、ユーザーを直接認証します。Tableau Serverが使用する名前は、サーバーがユーザー認証用に設定される方法によって異なります。

- **ローカル認証 -** 証明書からUPN (ユーザープリンシパル名) を使用します。
- **Active Directory (AD) -** LDAP (ライトウェイトディレクトリアクセスプロトコル) を使用してユーザー名を取得します。

これらの既定のいずれかを上書きし、Tableau Server が共通名を使用するように設定できます。

```
tsm authentication mutual-ssl configure -m cn
```

詳細については、相互認証中にクライアント証明書をユーザーにマッピングするを参照してください。

証明書失効リスト(CRL)

秘密キーの侵害が疑われる場合、または認証機関 (CA) が証明書を適切に発行しなかった場合、CRL の指定が必要になる場合があります。

```
tsm authentication mutual-ssl configure -rf <revoke-file.pem>
```

相互認証中にクライアント証明書をユーザーにマッピングする

相互 (2 方向) SSL 認証を使用する場合、クライアントは認証プロセスの一環としてTableau Serverに証明書を示します。次に、クライアント証明書内のユーザー情報を既知のユーザー ID にマッピングします。Tableau ServerTableau Server がクライアントのマッピングに使用する方法は、組織のクライアント証明書の内容によって異なります。

このトピックでは、クライアント証明書の情報のユーザー ID へのマッピング方法と、Tableau Server でのマッピング実行方法を変更する方法について説明します。マッピングがどのように発生するか、

およびそれに対する変更が必要かどうかを理解するには、組織内でクライアント証明書がどのように構造化されているかを知る必要があります。

- ユーザー名マッピングオプション
- 証明書マッピングの変更
- 複数ドメイン組織でユーザー名のあいまいさに対処する

ユーザー名マッピングオプション

Tableau Serverは以下のアプローチの一つをクライアント証明書をユーザーIDにマッピングするため使用します。

- **Active Directory**。ユーザー認証でActive Directoryを使用するようにTableau Serverが構成された場合、Tableau Serverがクライアント証明書を受け取る際に証明書をActive Directoryに渡し、その証明書をActive Directory IDにマッピングします。証明書内の明示的なユーザー名情報は無視されます。

  > **注**:この方法では、Active Directory内のユーザーアカウントに対してパブリッシュされるクライアント証明書が必要です。

- **ユーザープリンシパル名 (UPN)** 。ユーザー名をユーザープリンシパル名 (UPN) フィールドに保管するようにクライアント証明書を構成できます。Tableau Serverは UPN 値を読み取って、Active Directory内のユーザーまたはローカル ユーザーへマッピングします。
- **Common name (CN)**。ユーザー名を証明書の共通名 (CN) フィールドに保管するようにクライアント証明書を構成できます。Tableau Serverは CN 値を読み取って、Active Directory内のユーザーまたはローカル ユーザーへマッピングします。

Active Directory認証およびUPNまたはCNユーザー名マッピングをサーバーに構成した場合、ユーザー名を以下の形式の一つで表現してください。

`username`、`domain/username`、または `username@domain`。

例:`jsmith`、`example.org/jsmith`、`jsmith@example.org`

サーバーがローカル認証を使用する場合、UPN または CN フィールドにおける名前の書式はあらかじめ設定されていませんが、フィールド名はサーバーのユーザー名と一致する必要があります。

証明書マッピングの変更

クライアント証明書を Tableau Server のユーザー ID にマッピングするために、tsm authentication mutual-ssl <commands> コマンドを使用します。

```
tsm authentication mutual-ssl configure -m <value>
```

考えられる値は、Active Directory マッピングの場合は `ldap`、UPN マッピングの場合は `upn`、CN マッピングの場合は `cn` です。

Tableau Server を最初にインストールして構成するとき、サーバーは既定のユーザー名マッピングをサーバーの認証タイプに合わせて設定します。

- サーバーが Active Directory を使用するよう構成されされている場合、ユーザーIDに認証をマッピングするためにも Active Directory を使用します。
- サーバーがローカル認証を使用するよう構成されている場合、サーバーは証明書内のUPNフィールドからユーザー名の値を取得します。

Tableau Server がユーザー名を ID にマッピングする方法に関する既定の動作がサーバー構成に対して適切でない場合、次のコマンドセットを実行して CN 値を使用するようマッピングを変更します。

```
tsm authentication mutual-ssl configure -m cn

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

複数ドメイン組織でユーザー名マッピングのあいまいさに対処する

状況によっては、証明書の UPN または CN フィールドのユーザー名があいまいになる可能性があります。このあいまいさにより、ユーザー名をサーバー上のユーザー ID にマッピングしたときに予期しない結果が生じる可能性があります。

例として、Tableau Server がドメインの含まれないユーザー名を取得した場合、サーバーは既定のドメインを使用して、そのユーザー名を ID にマッピングします。これにより、あるユーザーに異なるユーザーの ID とパーミッションを割り当てる可能性があり、不正なユーザー名マッピングを引き起こす可能性があります。

これは、特に次の条件がすべて適用する環境で発生する可能性があります。

- 組織が複数の Active Directory ドメインをサポートしている。
- サーバーが Active Directory 認証を使用するように構成されている。
- サーバーが UPN や CN マッピングを使用するように構成されている。
- 一部のユーザーが同じユーザー名を使っているが、ドメインは異なっている。例えば、`jsmith@example.org` および `jsmith@example.com`。
- 証明書の UPN または CN フィールドにあるユーザー名に、ユーザー名の一部としてドメインが含まれていない (例えば `jsmith` とだけ表示する)。

間違ったユーザー名マッピングを防ぐために、クライアントの証明書に完全修飾ユーザー名とドメインが、`jsmith@example.org` または `example.org/jsmith` 形式で含まれるようにする。

外部の LDAP アイデンティティストアへの暗号化チャネルの構成

外部の LDAP アイデンティティストアに接続するように構成されている Tableau Server は、LDAP ディレクトリに対してクエリを実行し、セッションを確立する必要があります。セッションを確立するプロセスは *バインディング* と呼ばれます。バインディングにはいくつかの方法があります。Tableau Server では、LDAP ディレクトリにバインドする 2 つの方法がサポートされています。

- シンプル バインドは、ユーザー名とパスワードを使用した認証によってセッションを確立する方法です。デフォルトでは、Tableau Server は Windows Active Directory に接続する際、セッションを暗号化するために StartTLS を試みます。有効な TLS 証明書が Tableau Server にある場合、セッションは暗号化されます。そうでない場合、シンプル バインドによる LDAP は暗号化されません。シンプル バインドによる LDAP を構成する場合、SSL/TLS 経由の LDAP を有効にすることを強くお勧めします。

- **GSSAPI バインド:** GSSAPI は Kerberos を使用して認証します。キータブ ファイルを使用して構成した場合、GSSAPI バインド時の認証はセキュリティで保護されます。ただし、LDAP サーバーへの後続のトラフィックは暗号化されません。SSL/TLS 経由で LDAP を構成することをお勧めします。**重要:** StartTLS は Active Directory への GSSAPI バインドではサポートされていません。

  Active Directory ドメインに結合されているコンピューターの Linux 上で Tableau Server を実行している場合は、GSSAPI を構成できます。GSSAPI (Kerberos) バインドを使用した LDAP を参照してください。

このトピックでは、Tableau Server と LDAP ディレクトリサーバー間の通信に使用するシンプル LDAP バインド用のチャネルを暗号化する方法について説明します。

証明書の要件

- 暗号化に使用できる有効な PEM エンコードされた x509 SSL/TLS 証明書が必要です。証明書ファイルの拡張子は .crt である必要があります。

- 自己署名証明書はサポートされていません。

- インストールする証明書には、SSL/TLS に使用するキー使用フィールドに `Key Encipherment` が含まれている必要があります。Tableau Server では、この証明書を LDAP サーバーへのチャネルの暗号化にのみ使用します。有効期限、信頼、CRL、およびその他の属性は検証されません。

- Tableau Server を分散展開で実行している場合は、SSL 証明書をクラスタの各ノードに手動でコピーする必要があります。Tableau Server のアプリケーション サーバープロセスが構

成されているノードだけに証明書をコピーします。クラスタ環境内の他の共有ファイルとは異なり、LDAP で使用される SSL 証明書は、クライアント ファイル サービスによって自動的に配布されることはありません。

- PKI またはサードパーティ以外の証明書を使用している場合は、CA ルート証明書を Java 信頼ストアにアップロードします。

Tableau キーストアへの証明書のインポート

LDAP サーバー向けに構成したコンピューターに証明書をまだ配置していない場合は、LDAP サーバーで使用する SSL 証明書を入手し、それを Tableau システムのキーストアにインポートする必要があります。

Java ツールの "keytool" を使用して証明書をインポートします。既定では、このツールは次の場所で Tableau Server と共にインストールされます。

`/opt/tableau/tableau_server/packages/repository.<installer version>/jre/bin/keytool`。

次のコマンドは、証明書をインポートします。

```
sudo "<installation_directory>/packages/repository*/jre/bin/keytool -importcert -file "<cert_directory/<cert_name.crt>" -alias "<cert_alias>" -keystore /etc/opt/tableau/tableau_server/tableauservicesmanagerca.jks -storepass changeit -noprompt
```

Java キーストアのパスワードは `changeit` です。(Java キーストアのパスワードは変更しないでください)。

暗号化方式

Tableau Server 2021.1 以降では、シンプル バインド用の LDAP チャネルを暗号化する 2 つの方法 (StartTLS および LDAPS) がサポートされています。

- StartTLS: Tableau Server 2021.2 で Active Directory と通信するためのデフォルト構成です。Tableau Server 2021.2 以降、Active Directory へのシンプル バインドLDAP 接続では TLS が適用されます。これは既定の TLS 構成であり、新規のインストールとアップグレードシナリオの両方に適用されます。

  > **注**: StartTLS は、Linux 上の Tableau Server が Active Directory とシンプル バインドで通信する場合にのみサポートされます。StartTLS は、他のタイプの LDAP サーバーや GSSAPI との通信ではサポートされていません。

  StartTLS 方式は、Active Directory サーバーとのセキュリティで保護されていない接続を確立することによって機能します。クライアント/サーバーのネゴシエーションが完了すると、接続は TLS 暗号化接続にアップグレードされます。デフォルトの構成では、このシナリオで必要になるのは Tableau Server 上の有効な TLS 証明書のみです。他の構成は必要ありません。

- LDAPS: セキュリティで保護された LDAP (LDAPS) は、より多くの構成を必要とする標準の暗号化チャネルです。具体的には、Tableau Server での TLS 証明書に加えて、ターゲット LDAP サーバーのホスト名とセキュリティで保護された LDAP ポートを設定する必要があります。

  LDAPS は、Active Directory サーバーを含む任意の LDAP サーバーでサポートされています。

シンプル バインド用の暗号化チャネルの構成

このセクションでは、LDAP シンプル バインド用の暗号化チャネルを使用するように Tableau Server を構成する方法について説明します。

構成するタイミング

Tableau Server は、初期ノード設定の構成の [TSM CLIの使用] タブで説明されているように、Tableau Server を初期化する前、または初期ノードの構成の一部として、LDAP シンプル バインド用の暗号化チャネルを使用するように構成する必要があります。

Tableau Server の新規インストールの場合

組織で Active Directory 以外の LDAP ディレクトリを使用している場合、Tableau Server インストールの一部として TSM GUI セットアップを使用してアイデンティティストアを構成することはできません。代わりに、JSON エンティティファイルを使用して LDAP アイデンティティストアを構成する必要があります。identityStore エンティティを参照してください。

identityStore エンティティを構成する前に、このトピックで前述されているように、有効な SSL/TLS 証明書を Tableau キーストアにインポートします。

LDAPS を構成するには、identityStore JSON ファイルでホスト名と sslPort オプションを設定する必要があります。

Active Directory 環境での新規インストールの場合

Active Directory を外部アイデンティティストアとして使用している場合は、GUI バージョンの Tableau Server セットアップを実行する必要があります。Tableau Server をインストールするための CLI プロセスとは異なり、GUI バージョンのセットアップには、Active Directory の構成を簡略化して検証するロジックが含まれています。

Active Directory を構成する Tableau Server セットアップ GUI は次のとおりです。

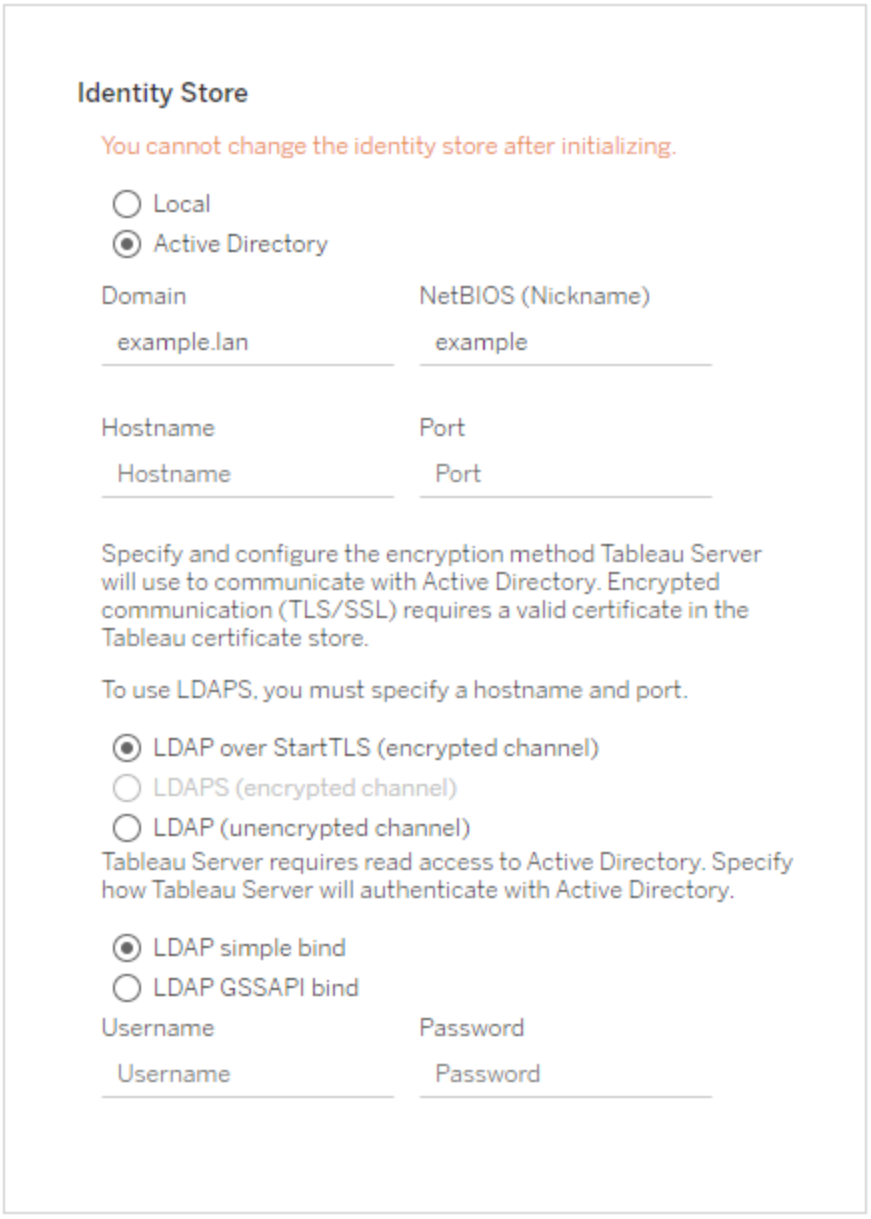


Tableau Server の新しいインスタンスを Linux にインストールする際、Tableau キーストアに有効な SSL/TLS 証明書がインストールされている場合は、デフォルトのオプションを StartTLS に設定したままにしておくことをお勧めします。

LDAPS を構成する場合は、LDAPS オプションを選択する前に、LDAP サーバーのホスト名とセキュア ポート (通常は 636) を入力します。

インストール後にこれらの構成を変更するには、TSM Web UI にサインインして、**[構成]** タブ、**[ユーザー ID とアクセス]**、**[アイデンティティストア]** の順にクリックします。

アップグレードシナリオ

Tableau Server を 2021.2 またはそれ以降のバージョンにアップグレードする際、外部 ID ストアとして Active Directory を使用している場合は、暗号化されたチャネルが LDAP シンプル バインド接続に適用されます。暗号化されたチャネルが構成されていない場合、アップグレードは失敗します。

2021.2 以降のバージョンに正常にアップグレードするには、次のいずれかである必要があります。

- 既存の Tableau Server インストールが LDAPS 用に構成されていて、Tableau キーストアに証明書がある。
- アップグレードする前に、有効な SSL/TLS 証明書が Tableau キーストアに存在する。このシナリオでは、デフォルトの StartTLS 構成によって暗号化されたチャネルが有効になります。
- 暗号化された LDAP チャネルが無効になっている。詳細については、次のセクションを参照してください。

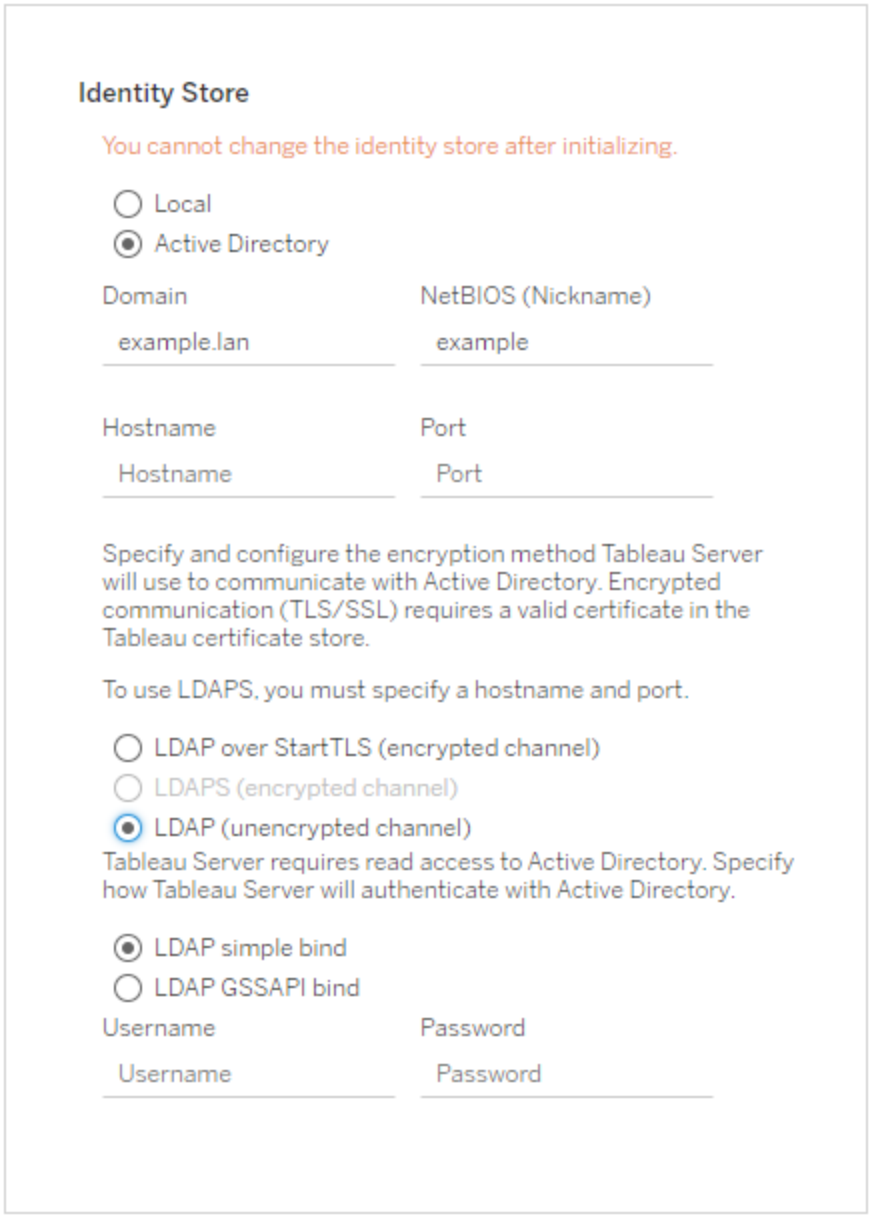
デフォルトの暗号化 LDAP チャネルを無効にする

Linux で Tableau Server を実行し、Active Directory に接続している場合、暗号化チャネルを必要とする条件を無効にできます。

無効にすると、Active Directory とのバインドセッションの確立に使用するユーザー認証資格情報は、Tableau Server と Active Directory サーバー間でプレーンテキストで通信されます。

新規インストールで無効にする

ID ストアとして Active Directory を使用する場合は、TSM GUI を使用して Active Directory との接続を構成する必要があります。初期ノード設定の構成を参照してください。



セットアップを実行時に **LDAP (暗号化されていないチャネル)** を選択します。

アップグレードする前に無効にする

Tableau Server を 2021.2 より前のバージョンから 2021.2 またはそれ以降のバージョンにアップグレードする場合は、アップグレードする前に、以前のバージョンの Tableau Server で次のコマンドを実行してください。

```
tsm configuration set -k wgserver.domain.ldap.starttls.enabled -v
false --force-keys

tsm pending-changes apply
```

キーが設定されていることを確認するには、次のコマンドを実行します。

```
tsm configuration get -k wgserver.domain.ldap.starttls.enabled
```

このコマンドは、`false`を返す必要があります。

エラー メッセージ

次のエラー メッセージが表示されるか、ログに記録される場合があります。これらのエラーが表示された場合は、次のステップを実行してください。

- このトピックで前述されているように、証明書が有効で Tableau キー ストアにインポートされていることを確認します。
- (LDAPS のみ) - ホストとポート名が正しいことを確認します。

GUI のセットアップの実行時

GUI のセットアップまたはアップグレードの実行時に LDAPS または StartTLS を正しく構成していない場合、次のエラーが表示されます。

```
TLS handshake failed. Tableau Server and the Active Directory
server could not negotiate a compatible level of security.
```

Vizportal ログ

CLI を使用して LDAPS または StartTLS を構成している場合は、次のエラー メッセージは表示されません。代わりに、エラーは次の vizportal ログに記録されます `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizportal`。

```
Authentication with LDAP server failed. The provided credentials or
configuration are either incorrect or do not have the necessary
permissions to bind.
```

システム ユーザー、sudo 権限、systemd

このトピックは Tableau Server のコンテキストでシステム ユーザー、systemd ユーザー サービスおよび sudo 権限について説明します。

権限の分離

標準的なセキュリティのベストプラクティスに従い、Tableau Server for Linux はできる限り権限が少ない状態でプロセスを実行します。インストールの際、権限なしのユーザーである ***tableau*** がサーバー認証グループ (***tableau***) に作成されます。

/etc/passwd ファイルでのユーザー入力の例を次に示します。

- `tableau:x:993:991:Tableau Server:/var/opt/tableau/tableau_server:/bin/bash`

すべてのプロセスは権限なしの ***tableau*** ユーザーとして実行されます。これは、Tableau Server プロセスの 1 つ (viz (ビジュアライゼーション) をユーザーに表示するプロセスなど) が何らかの形で損なわれた場合、影響があるのは Tableau Server だけになり、残りの Linux システムには影響がないことを意味します。このため、`tableau` 権限なしユーザーを `tsmadmin` グループに追加することはできません。`tsmadmin` グループには、OS 関連の Tableau 構成にアクセスするために許可を必要とするアカウントのみを含める必要があります。

`tableau` ユーザーおよび `tsmadmin` グループが Tableau Server 初期化プロセスで作成されます。インストール時に、権限のない別のユーザーまたは TSM 承認グループを指定できます。インストールとLDAP 構成のコンテキストでのシステム ユーザーとグループの詳細については、アイデンティティストアを参照してください。

sudo 権限

Tableau Server on Linux の最初のバージョン (10.5) は、sudoers ファイルを更新することにより sudo 権限に依存しています。sudoers ファイルの更新は、システム管理構成のベストプラクティスやセキュリティポリシーの一部と競合します。このため、2018.1 バージョン (以降) の Tableau Server では、権限のあるユーザー (`tsmagent`) の作成や使用が行われなくなりました。現在のバージョンの Tableau Server でも Tableau に固有の sudoers ファイルを更新したり、それが含まれることはなくなります。

現在、すべての権限操作はパッケージとソフトウェアのインストールの際に行われます。

systemd ユーザー サービス

10.5 バージョンの Tableau Server on Linux では、sudo 権限が TSM サービスの変更や再起動に求められ、systemctl コマンドを必要としました。すべての TSM サービスは通常のシステム全体の systemd プロセス (`process ID 1`、オペレーティング システムですべてのプロセスを実行する) から実行されました。このスキームでは systemd プロセスを root として実行します。このため、10.5 バージョンの Tableau Server は sudo 権限を必要としました。

現行の 2018.1 (以降) のリリースに伴い、systemd 権限を活用してユーザー サービスとして実行することで、sudo 権限の必要性をなくしました。systemd ユーザー サービスは通常ユーザーとして実行するため、一度有効にすると特別な権限は必要ありません。

通常の使用事例では、TSM により systemd へのコマンド発行が処理されるため、ユーザーがこれを行う必要はありません。ただし、トラブルシューティング シナリオの場合には TSM サービスを操作する必要がある場合があります。これまでのバージョンと同様、ユーザーはこれらのシナリオで同様の systemctl コマンドを発行します。とはいえ、コマンドは root としてではなく、`tableau` ユーザーとして実行する必要があります。Tableau Server のセットアップ中に権限のない別のシステム ユーザーを指定した場合、そのユーザーとしてコマンドを実行します。

systemctl コマンドの実行

systemctl コマンドを使用して systemd にリクエストを発行するには、以下の構文例を使用します。

権限なしのユーザーとしてセッションを開始します。環境変数を適切に設定するには `-l` フラグが重要です。

```
sudo su -l tableau
```

その後、コマンドを発行します。例は次のとおりです。

```
systemctl --user status tabadmincontroller_0
```

```
systemctl --user restart tabadmincontroller_0
```

セキュリティ強化チェックリスト

次のリストは、セキュリティの向上 (「強化」) を Tableau Server インストールで行うための推奨事項を提供します。

> Tableau Server on Windows については、「セキュリティ強化チェックリスト」参照してください。

セキュリティ更新のインストール

セキュリティ更新は、Tableau Server の最新バージョンおよびメンテナンス リリース (MR) に含まれています。セキュリティ更新をパッチとしてインストールすることはできません。代わりに、最新のセキュリティ修正で Tableau Server を更新するには、最新バージョンまたは MR にアップグレードする必要があります。

アップグレード後は、このトピックの最新バージョンを常に参照してください。最新バージョンには、トピックの URL に `/current/` が含まれています。

たとえば、米国バージョンの URL は https://help.tableau.com/current/server/ja-jp/security_harden.htm です。

1.現在のバージョンへの更新

常に最新バージョンの Tableau Server を実行することをお勧めします。さらに、Tableau では、既知のセキュリティ脆弱性に対する修正を含む Tableau Server のメンテナンス リリースを定期的にパブリッシュします。(セキュリティの脆弱性に関する既知の情報は、Tableau の「セキュリティ文書」ページ、および Salesforce の「セキュリティ アドバイザリ」ページにあります。) メンテナンス リリース通知を確認し、インストールするかどうかを決定することをお勧めします。

Tableau Server の最新バージョンまたはメンテナンス リリースを取得するには、カスタマー ポータルにアクセスしてください。

2.有効な信頼された証明書を使用してSSL/TLS を構成する

重要: バージョン2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキーペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

Tableau Server との通信のセキュリティを保護するには、セキュア ソケット レイヤー (SSL/TLS) が不可欠です。Tableau Desktop、モバイル デバイス、および Web クライアントが安全な接続を介してサーバーに接続できるように、有効で信頼できる証明書 (自己署名証明書以外) を使用して Tableau Server を構成します。詳細については、SSL を参照してください。

3.古いバージョンの TLS の無効化

Tableau Server は TLS を使用して、コンポーネントと外部クライアントの間の多くの接続を認証および暗号化します。ブラウザーなどの外部クライアント、Tableau Desktop、Tableau Mobile は TLS over HTTPS を使用して Tableau に接続します。トランスポート レイヤー セキュリティ (TLS) は、SSL の向上バージョンです。実際、古いバージョンの SSL (SSL v2 および SSL v3) は適切に保護された通信標準とは見なされなくなりました。その結果、Tableau Server 接続に SSL v2 または SSL v3 プロトコルを使用している外部クライアントの使用を許可しません。

外部クライアントが TLS v1.3 および TLS v1.2 を使用して Tableau Server に接続するのを許可することが推奨されています。

TLS v1.2 は依然として安全なプロトコルと見なされており、多くのクライアント (Tableau Desktop を含む) では TLSv1.3 はまだサポートされていません。

サーバーで TLS v1.2 がサポートされている場合でも、TLSv1.3 対応のクライアントは TLSv1.3 をネゴシエートします。

次の tsm コマンドは TLS v1.2 および v1.3 ("all" パラメーターを使用する) を有効にし、SSL v2、SSL v3、TLS v1、および TLS v1.1 を (指定されたプロトコルの先頭にマイナス [-] 文字を付けるこ

とにより) 無効にします。TLS v1.3 では、Tableau Server のコンポーネントでまだ対応していないものがあります。

```
tsm configuration set -k ssl.protocols -v 'all -SSLv2 -SSLv3 -TLSv1
-TLSv1.1'

tsm pending-changes apply
```

Tableau Server PostgreSQL リポジトリの SSL を管理するプロトコルを変更するには、pgsql.ssl.ciphersuite を参照してください。

また、Tableau Server が SSL/TLS セッションで使用する暗号スイートの既定リストを変更することもできます。詳細については、tsm configuration set のオプションの *ssl.ciphersuite* セクションを参照してください。

4.内部トラフィック用に SSL 暗号化を構成する

Postgres リポジトリと他のサーバーコンポーネントの間のすべてのトラフィックを暗号化するために SSL を使用するようTableau Server を設定します。デフォルトでは、SSL はサーバーコンポーネントとリポジトリとの間の通信について無効になっています。単一サーバーインストールであっても、Tableau Server のすべてのインスタンス用に内部 SSL を有効化することをお勧めします。マルチノード展開では特に、内部 SSL を有効にすることが重要です。内部 Postgres 通信用に SSL を設定するを参照してください。

5.ファイアウォールによる保護を有効にする

Tableau Server は保護された内部ネットワーク内で動作するように設計されました。

> **重要**: Tableau Server または Tableau Server のどのようなコンテンツもインターネット上または DMZ 内で実行しないでください。Tableau Server はインターネットファイアウォールで保護された会社ネットワーク内で実行する必要があります。Tableau Server に接続する必要があるインターネットクライアント用にリバースプロキシソリューションを構成することをお勧めします。Tableau Server のプロキシとロードバランサーの設定を参照してください。

シングル ノード展開およびマルチノード展開では、ローカルのファイアウォールをオペレーティング システムで有効化して Tableau Server を保護する必要があります。Tableau Server の分散 (マルチノード) インストールでは、ノード間の通信は安全な通信を使用しません。そのため、Tableau Server をホストするコンピューター上でファイアウォールを有効化することをお勧めします。ローカル ファイアウォールの構成を参照してください。

受動的な攻撃者によってノード間の通信が傍受されるのを防ぐため、分離された仮想 LAN または他のネットワーク レイヤー セキュリティ ソリューションを設定します。

Tableau Server で必要なポートとサービスを理解するには、Tableau サービス マネージャーのポートを参照してください。

6.サーバー コンピューターや重要なディレクトリへのアクセスを制限する

Tableau Server 構成ファイルおよびログ ファイルには、攻撃者に対して脆弱な情報が含まれる可能性があります。そのため、Tableau Server を実行しているマシンへの物理アクセスを制限します。さらに、認証され、信頼されたユーザーのみが `/var/opt/tableau/tableau_server/` ディレクトリの Tableau Server ファイルにアクセスできるようにします。

7.新しいシークレットとトークンの生成

リポジトリまたはキャッシュ サーバーと通信する Tableau Server サービスは、シークレット トークンを使用して最初に認証する必要があります。シークレット トークンは、Tableau Server セットアップ時に生成されます。セットアップ時には、Postgres リポジトリへのトラフィックを暗号化するために内部 SSL が使用する暗号化キーも生成されます。

Tableau Server をインストールした後、展開用に新しい暗号化キーを作成することをお勧めします。

これらのセキュリティ アセットは、`tsm security regenerate-internal-tokens` コマンドで再生成できます。

次のコマンドを実行します。

```
tsm security regenerate-internal-tokens
```

```
tsm pending-changes apply
```

8.使用していないサービスの無効化

Tableau Server の攻撃表面を最小限に抑えるため、不要な接続ポイントを無効化します。

JMX サービス

JMX は、既定で無効化されています。有効化されているにもかかわらず使用していない場合は、次を実行して無効化する必要があります。

```
tsm configuration set -k service.jmx_enabled -v false
```

```
tsm pending-changes apply
```

9.セッション存続時間設定の検証

既定では、Tableau Server に絶対的なセッション タイムアウトはありません。つまり、Tableau Server の非アクティブ タイムアウトを超えない場合、ブラウザーベースのクライアント セッション (Web 作成) を開いたままにすることができます。既定の非アクティブ タイムアウトは、240 分です。

セキュリティポリシーで要求される場合は、絶対的なセッション タイムアウトを設定できます。絶対的なセッション タイムアウトは、時間がかかる抽出のアップロードまたはワークブックのパブリッシュ操作が組織で実行できる範囲で設定してください。セッション タイムアウトの設定が短すぎると、操作に時間がかかりすぎて抽出やパブリッシュが失敗する場合があります。

セッション タイムアウトを設定するには、次のコマンドを実行します。

```
tsm configuration set -k wgserver.session.apply_lifetime_limit -v
true
```

`tsm configuration set -k wgserver.session.lifetime_limit -v` *`value`*、ここで *value* は分数です。既定では、1440 (24 時間) です。

`tsm configuration set -k wgserver.session.idle_limit -v` *`value`*、ここで *value* は分数です。既定では、240 です。

```
tsm pending-changes apply
```

接続されたクライアント(Tableau Desktop、Tableau Mobile、Tableau Prep Builder、Bridge、および個人用アクセス トークン) のセッションでは、OAuth トークンを使用して、セッションを再確立することによりユーザーをログインしたままにします。すべての Tableau クライアント セッションを、上記のコマンドによって制御されるブラウザーベースのセッション制限のみによって管理する場合は、この動作を無効にできます。詳細については、自動クライアント認証を無効にするを参照してください。

10. ファイルベースのデータ ソース用にサーバーの許可リストを設定する

2023 年 10 月の Tableau Server リリース以降、デフォルトのファイルベースのアクセス動作が変わりました。以前は、Tableau Server は許可された Tableau Server ユーザーに対し、サーバー上のファイルをファイル ベースのデータ ソース (スプレッドシートなど) として使用するワークブックの作成を許可していました。2023 年 10 月のリリースでは、Tableau またはリモート共有に保存されているファイルへのアクセスは、ここで説明する設定を使用して Tableau Server で特別に構成する必要があります。

この設定により、`tableau`システム アカウントによるアクセスを、指定したディレクトリのみに制限できます。

共有ファイルへのアクセスを構成するには、許可リスト機能を構成する必要があります。これにより、データ ファイルをホストしているディレクトリ パスのみに `tableau` アカウントのアクセスを制限できます。

1. Tableau Server を実行しているコンピューターで、データ ソース ファイルをホストするディレクトリを識別します。

   > **重要:** この設定で指定したファイル パスが存在し、システム アカウントでアクセス可能であることを確認してください。

2. 次のコマンドを実行します。

`tsm configuration set -k native_api.allowed_paths -v "path"`、ここで、*path* は許可リストに追加するディレクトリです。指定したパスのすべてのサブディレクトリは、許可リストに追加されます。指定したパスの末尾にバックスラッシュを追加する必要があります。複数のパスを指定する場合は、この例のようにセミコロンで区切ります。

```
tsm configuration set -k native_api.allowed_paths -v
"/datasources;/HR/data/"
```

```
tsm pending-changes apply
```

11.Web ブラウザー クライアントでの HTTP Strict Transport Security の有効化

HTTP Strict Transport Security (HSTS) は、Tableau Server などの Web アプリケーション サービスで構成されるポリシーです。適合するブラウザーに HSTS を実行している Web アプリケーションがある場合、サービスとのすべての通信が安全な接続 (HTTPS) を介して行われる必要があります。HSTS は主要なブラウザーでサポートされています。

HSTS のしくみとサポートされるブラウザーの詳細については、Open Web Application Security Project Web ページの「HTTP Strict Transport Security チートシート (HTTP Strict Transport Security Cheat Sheet)」を参照してください。

HSTS を有効にするには、Tableau Server で次のコマンドを実行してください。

```
tsm configuration set -k gateway.http.hsts -v true
```

既定では、HSTS ポリシーが 1 年に設定されています (31536000 秒)。ブラウザーが HTTPS 経由でサーバーにアクセスする期間を指定します。HSTS の初期ロールアウト時は最長期間を短く設定することを検討する必要があります。この期間を変更するには、`tsm configuration set -k gateway.http.hsts_options -v max-age=<seconds>` を実行します。たとえば、HSTS ポリシーの期間を 30 日に設定し、`tsm configuration set -k gateway.http.hsts_options -v max-age=2592000` と入力します。

```
tsm pending-changes apply
```

12.ゲストアクセスの無効化

Tableau Server のコアベースのライセンスには、ゲストユーザー オプションが含まれます。このオプションでは、組織のすべてのユーザーに対し、Web ページに埋め込まれた Tableau ビューの表示や操作を許可できます。

コアベースのライセンスを使用して展開された Tableau Server では、ゲストユーザー アクセスが既定で有効になっています。

ゲストアクセスによって、ユーザーは埋め込みビューを表示できます。ゲストユーザーは Tableau Server インターフェイスを参照できません。また、ビューを操作するサーバー インターフェイス エレメントは表示されません (ユーザー名、アカウント設定、コメントなど)。

組織でコア ライセンスを使用して Tableau Server を展開しており、ゲストアクセスが必要ない場合は、ゲストアクセスを無効にします。

ゲストアクセスをサーバー レベルまたはサイトレベルで無効にできます。

サーバー レベルまたはサイトレベルでゲストアカウントを無効にするには、サーバー管理者である必要があります。

ゲスト アクセスをサーバー レベルで無効にするには、次の手順を行います。

1. サイト メニューで、**[すべてのサイトを管理] > [設定] > [全般]** の順にクリックします。
2. **[ゲスト アクセス]** で、**[ゲスト アカウントを有効にする]** チェック ボックスをクリアします。
3. **[保存]** をクリックします。

サイトのゲスト アクセスを無効にするには、次の手順を行います。

1. サイト メニューで、サイトを選択します。
2. **[設定]** をクリックし、[設定] ページで、**[ゲスト アカウントを有効化]** チェック ボックスをクリアします。

詳細については、ゲストユーザーを参照してください。

13.Referrer-Policy HTTP ヘッダーを'same-origin' に設定する

2019.2 から、Tableau Server には Referrer-Policy HTTP ヘッダーの動作を構成する機能が含まれています。このポリシーは、すべての "secure as" 接続で起点の URL を含める既定の動作 (`no-referrer-when-downgrade`) を指定して有効になっています。この動作では、同種の接続 (HTTP からHTTP) またはよりセキュリティの高い接続 (HTTP からHTTPS) にのみ、起点の参照元に関する情報が送信されます。

この値を、参照元の情報が同じサイトの起点にのみ送信される `same-origin` に設定することをお勧めします。サイトの外部からの要求は参照元に関する情報を受信しません。

Referrer-Policy を `same-origin` に更新するには、次のコマンドを実行します。

```
tsm configuration set -k gateway.http.referrer_policy -v same-origin

tsm pending-changes apply
```

セキュリティ向上を目的とした追加のヘッダー構成の詳細については、HTTP 応答ヘッダーを参照してください。

14.TLS によるSMTP 接続の構成

2019.4 から、Tableau Server には TLS によるSMTP 接続を構成する機能が含まれています。Tableau Server では、STARTTLS (便宜的または明示的な TLS) のみがサポートされています。

オプションで、メール サーバーに接続するように Tableau Server を構成することができます。SMTP を構成したら、システム障害についてサーバー管理者にメールを送信し、サブスクライブしたビューおよびデータ主導アラートについてサーバー ユーザーにメールを送信するようTableau Server を構成することができます。

TLS によるSMTP を構成するには、次の手順を実行します。

1. 互換性のある証明書を Tableau Server にアップロードします。tsm security custom-cert add を参照してください。

2. TSM CLI を使用して TLS 接続を構成します。

   次の TSM コマンドを実行して、SMTP サーバーへの TLS 接続を有効化および強制し、証明書の検証を有効にします。

   ```
   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   enabled -v true

   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   required -v true

   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   check_server_identity -v true
   ```

   既定では、Tableau Server は TLS バージョン 1、1.1、および 1.2 をサポートしますが、SMTP サーバーがサポートする最も高い TLS バージョンを指定することをお勧めします。

   次のコマンドを実行してバージョンを設定します。有効な値は、`SSLv2Hello`、`SSLv3`、`TLSv1`、`TLSv1.1`、および `TLSv1.2` です。次の例では、TLS バージョンをバージョン 1.2 に設定します。

   ```
   tsm configuration set -k svcmonitor.notification.smtp.ssl_
   versions -v "TLSv1.2"
   ```

   TLS 構成オプションの詳細については、SMTP セットアップの構成 を参照してください。

3. 変更を反映するには、Tableau Server を再起動します。次のコマンドを実行します。

   ```
   tsm pending-changes apply
   ```

15.LDAP の SSL を設定

汎用 LDAP 外部 ID ストアを使用するように Tableau Server の展開が構成されている場合は、Tableau Server とLDAP サーバー間の認証を保護するように SSL を構成することをお勧めします。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

Active Directory を使用するように Tableau Server の展開が構成されている場合は、認証トラフィックを保護するために Kerberos を有効にすることをお勧めします。Kerberosを参照してください。

変更リスト

Date	Change
May 2018	Added clarification: Do not disable REST API in organizations that are running Tableau Prep.
May 2019	Added recommendation for referrer-policy HTTP header.
June 2019	Removed recommendation to disable Triple-DES. As of version 2019.3, Triple-DES is no longer a default supported cipher for SSL. See 変更箇所 - アップグレード前に知っておく事柄.
January 2020	Added recommendation to configure TLS for SMTP.
February 2020	Added recommendation to configure SSL for LDAP server.
May 2020	Added TLS v1.3 to the disabled list of TLS ciphers. Added clarification to introduction about topic versioning.
October 2020	Added TLS v1.3 as a default supported cipher.
January 2021	Added clarification: All products enabled by the Data Management license require REST API.
February 2021	Removed recommendation to disable REST API. The API is now used internally by Tableau Server and disabling it may limit functionality.

ライセンスの管理

Tableau Server ライセンスを管理し、ライセンス使用状況を表示できます。

ライセンス発行の概要

Tableau Server の展開で重要となる管理者ロールは、Tableau ポータル管理者です。ポータル管理者は、Tableau の展開に必要なライセンスの発行や関連するキーの管理を行います。ポータル管理者としての最初のステップは、Tableau カスタマー ポータルでライセンスを購入することです。ライセンスを購入すると、ポータルから対応するプロダクト キーが返されます。ライセンスを更新するには、Tableau の更新 Web ページにアクセスしてください。

Tableau には、多数の製品 (Desktop、Server、Prep Builder など) があります。Tableau 製品を使用するには、Tableau カスタマー ポータルで購入・保存したプロダクト キーを使用して Tableau ソフトウェアを更新し、ライセンスをアクティブにする必要があります。Tableau のライセンス認証を担当する管理者は、ライセンスとキーの関係を理解することが重要です。ライセンス モデルとプロダクト キーを理解するを参照してください。

アクティブ化

アクティブ化とは、Tableau のプロダクト キーを Tableau Server にアップロードして保存するプロセスです。この操作は、Tableau サービス マネージャー (TSM) で行います。TSM は、ローカル オペレーティング システムとファイル システムに変更を加えるツールであるため、ローカル コンピューターへの管理アクセスが必要です。TSM 管理者には、ユーザーやサイトの追加、プロジェクトやパーミッションの管理など、Tableau Server タスクの日常操作の管理ロールが必要です。これは、Tableau Server 管理者に付与されるパーミッションとアクセス権とは異なります。Tableau Server のさまざまな管理ロールについて詳しくは、管理者の役割を参照してください。

次のトピックでは、TSM への接続方法について説明します。

- Tableau サービス マネージャーの Web UI へのサインイン
- tsm コマンド ライン リファレンス

オンラインでのアクティブ化

インターネット通信ができる環境に Tableau Server をインストールした場合は、オンラインでアクティブにする既定の方法をお勧めします。

- インストール プロセスでアクティブにする方法については、Tableau Server のライセンス認証と登録を参照してください。

- サブスクリプションを更新した後にプロダクトキーをアクティブにする方法については、プロダクトキーの有効期限と属性の更新を参照してください。
- 購入した新機能やユーザーライセンスを追加した後でプロダクトキーをアクティブにする方法については、Tableau Server へのライセンスの追加を参照してください。

オフラインでのアクティブ化

Tableau Server をオフライン環境で実行しており、インターネット経由で Tableau ライセンスのサーバーにアクセスできない場合は、Tableau オフライン アクティブ化プロセスに従ってライセンスをアクティブにする必要があります。

- オフラインでアクティブ化する方法の詳細については、オフラインでの Tableau Server の有効化を参照してください。
- オフライン アクティベーションを使用してアクティブ化したプロダクトキーを非アクティブにする方法については、「オフラインでの Tableau Server のライセンス認証解除」を参照してください。

アクティブ認証の失敗

ライセンスをアクティブにした後でも、そのアクティブ認証が失敗する場合があります。このような障害は、ローカルプロセスの接続エラーや、VM またはハードウェア構成に変更があった場合に発生する可能性があります。たとえば、プロキシの変更、ポートのブロック、ネットワークの変更、またはマシンのハードウェアの変更によってライセンス認証が失敗する可能性があります。Tableau Server がライセンスを確認できない場合は、操作が中断し、サーバーはライセンスがない状態となります。

プロダクトキーとTableau Server ライセンスの状態を表示するには、`tsm licenses list` と `tsm status -v` を実行します。

未確認のプロダクトキーによっては、プロダクトキーが有効な状態になるまで Tableau Server の機能が低下した状態になることがあります。ライセンス発行のトラブルシューティングを参照してください。

ライセンス認証の解除

Tableau Server の同一のプロダクトキーは最大 3 つの環境でアクティブにすることができます。これにより、サンドボックス環境や QA 環境で Tableau Server をテストでき、実稼働環境で Tableau を使用することもできます。アクティブ化を最大限に活用するには、同じコンピューターに Tableau を再度インストールする場合以外は、Tableau Server をコンピューターから削除したり、VM を終了

する際に、プロダクトキーを非アクティブにする必要があります。これにより、別のコンピューターでライセンス認証を行うことができます。たとえば、あるコンピューターから別のコンピューターに Tableau Server を移動する場合は、プロダクトキーを非アクティブにしてから、元のコンピューターから Tableau を削除します。新しいコンピューターに Tableau をインストールするとき、競合を発生させずにキーをライセンス認証できます。同じコンピューターに再インストールするために Tableau Server を削除する場合は、キーのライセンス認証を解除する必要はありません。Tableau では、"-l" オプションを指定して削除コマンドを実行する場合を除き、再度インストールするときはそのキーを使用します。

プロダクトキーの認証解除を参照してください。

Tableau Server のライセンス発行と仮想マシン (VM)

ローカルまたはクラウド上の VM で Tableau Server を実行する場合は、ライセンスに関連する作業が複雑になる可能性があります。VM 上で Tableau Server をアップグレードするだけであれば、ライセンスに関連する追加のアクションは必要ありません。アップグレードする新しい実稼働環境またはテスト環境を作成するために VM を複製する場合は、複製する前に Tableau Server のライセンス認証をすべて解除する必要があります。これを行わなければ、新しい VM 環境は信頼済みライセンスがない状態になり、Tableau Server でアップグレードや起動を試みてもすべて失敗します。新しい VM でプロダクトキーをアクティブにしようとすると、ライセンスをアクティブにできる最大数に達する場合もあります。

VM でのライセンス認証に関する問題を回避するには、VM を複製したり、完全にシャットダウンする前に、Tableau のすべてのライセンス認証を解除します。

ログインベースのライセンス管理

ログインベースのライセンス管理を使用すると、Tableau Server と Tableau Cloud で Creator のロールを持つユーザーのライセンス発行を管理できます。Explorer または Viewer のロールを持つユーザーは、この機能を使用できません。Tableau Server や Tableau Cloud でロールベースのサブスクリプションを使用している場合は、ログインベースのライセンス管理を使用してライセンス管理を簡略化し、独立した Tableau Desktop や Tableau Prep Builder のプロダクトキーを取り除くことができます。管理する必要があるのは、オンプレミスの Tableau Server 用の 1 つまたは複数のプロ

ダクトキーだけです。Tableau Cloud の場合は、プロダクトキーを管理する必要はまったくありません。

ログインベースのライセンス管理を参照してください。

ユーザーの追加

Tableau Server でリソースにアクセスするユーザーはライセンスを取得する必要があります。

- ユーザーのロールとライセンス発行については、ライセンス モデルとプロダクトキーを理解するを参照してください。
- サーバーにユーザーを追加するには、Tableau Server へのユーザーの追加を参照してください。
- 購入した新しいユーザー ライセンスを追加した後にプロダクトキーをアクティブにする方法については、Tableau Server へのライセンスの追加を参照してください。

ライセンス モデルとプロダクトキーを理解する

このトピックでは、さまざまなライセンス モデルと、それらに関連する可能性のあるプロダクトキーまたはサブスクリプションについて説明します。Tableau Server でのプロダクトキーの表示について、視覚的にわかりやすい内容が tsm licenses list にあります。また、Tableau カスタマー ポータルには、タイプやシート数などのプロダクトキー情報も示されています。

プロダクトキーを tsm licenses list (Tableau Server) を使用して表示する場合、または Tableau カスタマー ポータル (Tableau Desktop および Tableau Prep Builder) で表示する場合は、製品固有のプレフィックスに注意してください。

プロダクトキープレフィックス	説明
TC	Tableau Creator のプロダクトキー。Tableau Desktop および Tableau Prep Builder のアクティブ化または非アクティブ化に使用できます。
TD	Tableau Desktop のプロダクトキー。Tableau Desktop のアクティブ化または非アクティブ化にのみ使用できます。これは従来のプロダクトキーであり、もう販売も提供もされていません。

TS	Tableau Server のプロダクト キー。Tableau Server のアクティブ化 または非 アクティブ化 に使用 できます。Tableau Server のプロダクト キーは、ロール ベース、コア ベース、または機能 ベースに設定 できます。

期間 ライセンス モデル

Tableau の期間 ライセンス モデルは、Tableau Server の使用 を許可 する指標で定義 されています。期間 ライセンスは、サブスクリプション ライセンスとも呼ばれます。Tableau は現在、Tableau Server へのアクセスをサブスクリプション ライセンスで販売 しています。サブスクリプション ライセンス モデルでは、お客様は 1 年間 のサブスクリプション料金 を支払います。サブスクリプションの有効期限 が切れると、ソフトウェアは動作 しなくなります。

サブスクリプション ライセンスには、ロール ベースのサブスクリプション とコア ベースのサブスクリプション があります。すべてのロール と機能 を備えた 1 つのライセンス キーを購入 することもできます。このライセンスは、更新可能 なサブスクリプション ライセンス (Updatable Subscription License、USL) と呼ばれます。Tableau Server でアクティブ化 する必要 があるキーは 1 つだけであり、購入 の全体 を表しています。

以前の (USL 以外の) サブスクリプション ライセンスでは、キーごとに 1 つのロール タイプが提供 されていたため、購入 したものを取得 して Tableau Server で構成 するには、「積み重ね」られたライセンスを一緒 にアクティブ化 する必要 がありました。

- *ロール ベースのライセンス*では、Tableau Server を 1 台 のコンピューターまたは クラスタ内 の複数 のコンピューターに展開 できます。Tableau Server にアクセスする各 ユーザーには ライセンスがあり、ロールが割 り当 てられている必要 があります。管理者 は、使用可能 な各 タイプのライセンス数 に基づいてユーザーを追加 できます。
- *コア ベースのライセンス*では、Tableau Server 内 のユーザー アカウント数 には制限 がありません。代わりに、Tableau Server を実行 できる PC コアの最大数 がライセンスによって指定 されます。このライセンスを使用 することで、PC 内 のコアの合計数 が許容 されているライセンスの合計数 を超えない限 り、1 台 の PC または マルチノード クラスタとして複数 の PC に Tableau Server をインストールできます。

Tableau Server にインストールされているすべてのプロセスが、使用するコアの合計数の計算に影響するわけではありません。プロセスのサブセットは "ライセンスプロセス" と見なされます。コア ライセンス発行は、ライセンスプロセスを実行する PC でのみ計算されます。PC に 1 つまたは複数のライセンスプロセスがインストールされている場合、その PC のコア数は使用されている合計コア数に対してカウントされます。ライセンスプロセスについての詳細情報は、ライセンスプロセスを参照してください。

- *実行可能なサブスクリプション ライセンス*により、1 つプロダクト キーでライセンスを統合し、Tableau Server を更新できます。既存の 1 つの Tableau Server ライセンスに対して、新機能の追加、容量の調整、ライセンス更新の適用が可能です。新しいライセンスを追加したり、既存のライセンスを置き換えたりする必要はなくなりました。USL:

    - 管理しなければならないプロダクト キーの数を減らし、保守を容易にすることで、キー管理を簡素化します。

    - ライセンスの更新、容量の調整、または新機能の追加後に Tableau Server を再起動する必要がないため、サービスの中断を最小限に抑えることができます。

更新可能なサブスクリプション ライセンス (USL) では、新機能やロール数の変更によりプロダクト キーが更新されても、Tableau カスタマー ポータルのプロダクト キーが変更されることはありません。USL 以外のサブスクリプション ライセンス モデルでは、ライセンスを更新するたびに、カスタマー ポータルで新しいキーを取得します。非 USL サブスクリプションのプロダクト キーを表示していると、前のプロダクト キーの有効期限が切れると、新しいプロダクト キーが TSM Web UI に表示されます。更新可能なサブスクリプション ライセンス (USL) の場合、ご利用の製品キーが UI や Tableau カスタマー ポータルで変更されることはありません。Tableau カスタマー ポータルでは、USL ライセンスにより、**[ライセンスの詳細]** ページで **[Is USL Key (USL キー)]** チェックボックスがオンになっていること、または **[ライセンス]** タブの **[Is USL Key (USL キー)]** 列が **true** になっていることが表示されます。

tsm licenses list が返す出力では、[タイプ] フィールドにユーザー ライセンスのメトリックが記述されています。Tableau Services Manager Web UI で、[タイプ] フィールド (または列) にカーソルを合わせると、そのキーが更新可能なサブスクリプション ライセンス (USL) のプロダクト キーであるかどうかを確認できます。

ロールベースのライセンス モデル

Tableau ではロールベース期間 ライセンスを提供しており、一連の機能をさまざまな価格で利用できます。ロールベースの期間 ライセンスには、Display、Viewer, Explorer、Creator の 4 種類があります。

- *Display ライセンス*を使用すると、広い範囲の内部ユーザーに Tableau コンテンツを共有して表示することができ、対話形式ではない共用の画面を使用したダッシュボードを提供できます。Display ライセンスには、個別のサイト ロールはありません。Display ライセンスを使用する場合、管理者は、ライセンスを取得したディスプレイの場所ごとに専用のログイン アカウントを作成します。これは、個々のユーザーのログイン アカウントとは異なり、最大のサイト ロールである Viewer を割り当てます。
- *Viewer ライセンス*では、Tableau Server でワークブックを表示し、操作することができます。Viewer ライセンスではユーザーに対し、Tableau Mobile へのアクセス、ワークブックへのコメント追加、さまざまな形式でのビジュアルのエクスポート、ワークブックのサマリー データのダウンロード、それ自体に関するサブスクリプションの作成、データ ドリブン アラートの受信も許可します。
- *Explorer ライセンス*では、以前の Tableau Server リリースで利用可能なユーザーベースの期間 ライセンスと似ており、Viewer ライセンスで提供される権限に加えて、追加の権限が含まれています。Explorer ライセンスでは、Web ブラウザーを使用したワークブック作成権限のほか、コラボレーション機能の完全なセットへのアクセスが許可されます。
- *Creator ライセンス*では、Tableau Server を使用する際に幅広い権限が付与され、Tableau Desktop や Tableau Prep Builder も使用できます。Creator ライセンスでは、Explorer ライセンスで使用できるすべての権限に加え、Tableau Server を使用する際に、次の権限を使用できます。
  - 新しいデータ ソースから新しいワークブックを作成したりパブリッシュしたりする。
  - [データ] ペインの埋め込みデータ ソースを編集する。
  - 新しいデータ接続を作成したりパブリッシュしたりする。
  - Tableau Desktop とTableau Prep Builder では、ログインベースのライセンス管理 アクティベーションを使用します。

注: Tableau Server 管理者は、使用可能な最高のロールを常に使用します。Creator ロールを含むプロダクト キーをアクティブ化すると、Tableau Server 管理者がこのロールを取得します。Tableau Server で使用可能な最高のロールが Explorer の場合、サーバー管理者は

> Explorer ロールを担います。現在アクティベートされているCreator がいないサーバーに Creator ロールを追加すると、Explorer ライセンスを使用している既存のサーバー管理者のアカウントは、Creator ライセンスを使用するように自動的に変換されます。
>
> TSM 管理者アカウントにはライセンスは必要ありません。

USL 以外のライセンスの場合、プロダクトキーを使用してライセンスをアクティブ化し、Tableau Server に追加します。USL 以外のプロダクトキーで容量または機能の更新を購入した場合、Tableau Server 管理者はこれらの追加のプロダクトキーをアクティブ化します。USL プロダクトキーの場合、すべてのロールと機能を含む 1 つのプロダクトキーがアクティブ化されます。USL ライセンスが更新され、機能、特長、ロール数が変更された場合、追加のプロダクトキーは発行されず、アクティブ化する必要もありません。

更新可能なサブスクリプション ライセンス (USL) プロダクトキーの場合、新しいサブスクリプション期間、機能、ロール数の変更によりキーが更新されても、Server ATR サービスが自動的にこれらの更新を取得するため、追加のキーをアクティブ化する必要はありません。USL ライセンスとServer ATR のアクティブ化は、これらの変更を取得するために更新する必要はありません。すべてServer ATR サービスによって処理されます。サーバーATR 以外のアクティブ化の方法をとるUSL プロダクトキーの場合、更新を行ってUSL プロダクトキーへの更新を取得する必要があります。USL 以外のプロダクトキーの場合、容量または機能の更新を購入すると、Tableau Server でアクティブ化する必要がある追加のプロダクトキーがTableau カスタマー ポータルで提供されます。

USL 以外のプロダクトキーを使用している場合は、Tableau カスタマー ポータルからTableau Server Creator のプロダクトキーを選択し、Tableau サーバー管理者を作成できるようにします。その後、Explorer とViewer のプロダクトキーを認証し、別のライセンスを追加できます。プロダクトキーが認証されると、管理者はユーザーを追加してサイトロールを割り当てることができ、これにより利用可能なライセンスが自動的に使用されます。

tsm licenses list によって返される出力の CREATOR、EXPLORER、VIEWER フィールドでは、各ロールのライセンス タイプのライセンス数が表示されます。USL 以外のプロダクトキーの場合、各ロールは自身のプロダクトキーによって管理されます。このため、組織が 3 種類すべてのロールのラ

イセンスを購入した場合は、3 種類のプロダクトキーを認証する必要があります。USL プロダクトキーの場合、TSM に反映されるすべてのロールと機能が 1 つのキーに含まれています。

コアベースのライセンス モデル

コア ライセンス モデルでは、ライセンスは、追加できるユーザー数ではなく、サーバーが実行できる PC の合計コア数を定義します。つまり、サーバーは事実上無制限のユーザー(Explorer または Viewer。これらのタイプのロールは以前「インタラクター」と呼ばれていました) をサポートできます。コアベースのライセンスではゲストユーザー アカウントも許可されますが、ロールベース ライセンスではできません。

ただし、コア ライセンスにはサーバー上の Creator シートは含まれません (これらのサイトロールはユーザーを追加するとグレー表示されます)。

新しいデータ ソースをサーバーに追加するには、サーバー上で Creator サイトロールを持つユーザーが、Tableau Desktop から、またはブラウザー経由でパブリッシュする必要があります。したがって、(a) サーバー上に Creator サイトロールを持つユーザーが存在するか、(b) ライセンスが設定されている Tableau Desktop と Explorer (パブリッシュ可能) サイトロールを持つユーザーが存在する必要があります。(B) の場合、コアベースのサーバー組織で Tableau Desktop のライセンスを設定するには 2 つの方法があります。

- ロールベースの Creator ライセンスが少なくとも 1 つ (Tableau Desktop と Tableau Prep Builder を含む) ある、または
- 従来の Tableau Desktop プロダクトキーを使用する、ライセンス設定された Tableau Desktop が少なくとも 1 つある。

Creator ライセンスが、コアベース ライセンスが設定されているサーバーにスタックされている場合、そのロールベースの Creator ライセンスは管理者が使用することになります。Creator サイトロールを管理者以外のユーザーに付与する唯一の方法として、まず、すべての管理者が Creator ライセンスを持っていることを確認します。次に、追加の Creator ライセンスを使用して、管理者以外のユーザーに Creator サイトロールを割り当てます。

tsm licenses list によって返される出力の TYPE フィールドでは、ライセンスされているコア数が表示されます。さらに、GUEST ACCESS フィールドに `true` が表示されます。

組み込み分析の使用量ベースのモデル

Tableau の埋め込み分析サービスは Tableau Server の使用量ベースのライセンスであり、Tableau Server の機能を外部向けソリューションに組み込んで、組織外のクライアントに Tableau コンテンツとインサイトを提供したいと考えているお客様が利用できます。

> **注:** 埋め込み分析ライセンスは、フル活用のライセンスと同じ環境では使用できません。埋め込み分析ライセンスに変更するには、まず既存のフル利用ライセンスを非アクティブ化してから、埋め込み分析ライセンスをアクティブ化します。

たとえば、ある組織が、消費者データを分析し、さまざまな消費者の人口統計に関する行動パターンのレポートを生成するサービスを実行しているところを考えてください。このシナリオでは、Tableau Server は「人口統計アナライザー」という名前の特定の専用アプリケーションをサポートし、エクスポートされた TXT ファイルや SQL データベースと接続します。組織はセキュア ポータルでクライアントがビジュアライゼーションを利用できるようにし、クライアントはそこにログインしてアカウントを管理し、結果を表示します。エンドユーザーは、ポータルへのアクセスに使用するアカウントによって一意に識別されます。これにより、ユーザーベースの埋め込み分析サーバーのライセンスの数が決まります。

埋め込み分析のライセンスは、tsm licenses list によって返される出力に表示されません。ライセンスを確認するには、カスタマー サクセスにお問い合わせください。

永続ライセンス モデル (従来)

以前 Tableau では、永続ライセンスを使用した Tableau Server へのアクセスを販売していました。これらのライセンスは利用できなくなりましたが、一部のお客様はこのライセンス制度を使用しています。

永続ライセンス モデルでは、お客様は毎年更新されるメンテナンス サブスクリプションを支払っていました。メンテナンスの有効期限が切れると、ソフトウェアは動作し続けますが、お客様はテクニカル サポートとソフトウェアのアップグレードにアクセスできなくなります。現在、永久ライセンスの年間メンテナンスを購入することはできません。

永続ライセンスは、インタラクターと呼ばれる特定の数のユーザーに対して販売されたか、特定の数のコアに対して販売されました。

- インタラクター ライセンスは、現在のロールベース ライセンスと同様に、お客様がシートごとにライセンスを購入した名前付きユーザー モデルです。ただし、ロールベース ライセンスで別々のアクセス ロールにはそれに応じ価格が設定されているのとは異なり、インタラクター ライセンスではライセンスはロールと結び付けられていませんでした。ライセンス ユーザーは、サーバー管理者、サイト管理者、パブリッシャー、インタラクター、または Viewer でした。ユーザー ロールは、管理者がコンテンツおよびサーバー構成へのアクセスを管理するための手段としてのみ設定されました。

  バージョン 2020.4 以降で有効な Server Interactor キーをアクティブ化すると、Interactor カウントが Explorer ロールにマッピングされます。

  Interactor 永久コア ライセンスは、ログインベースのライセンス管理を使用できません。Tableau Desktop または Tableau Prep Builder で ログインベースのライセンス管理 を使用するには、Creator 期間 ライセンス (Creator サブスクリプション ライセンスとも呼ばれます) を購入する必要があります。

- 永続コア ライセンスにはサブスクリプション コア ライセンスと同様のモデルがあり、ソフトウェアが実行可能な PC のコア数を指定し、無制限のユーザーとゲスト アカウントをサポートします。

tsm licenses list によって返される出力の TYPE フィールドには `Perpetual` が表示されます。また、MAINT EXP の日付も表示されることにも注意してください。

ライセンス エディション

ライセンス エディションには、ユーザーが利用できる一連の機能が含まれています。ライセンス エディションは展開内で混在させることはできません。つまり、1 つの展開上のすべてのユーザーは同じライセンス エディションを使用する必要があります。ライセンス エディションを選択するときは、展開全体のニーズを考慮してください。

Tableau ライセンス エディション

Tableau ライセンス エディションは、標準の Tableau Server エディションです。Tableau Server に関して、ライセンスされたユーザーごとのアクセス、ガバナンス、コラボレーション、データ準備、ビジュアル分析の機能を提供します。

Tableau Enterprise ライセンス エディション

Enterprise エディションは、高度なビジネス環境向けに設計された Tableau Server です。高度な管理、セキュリティ、データ管理機能を必要とする組織に最適であり、より複雑なデータ環境でより多くのユーザーに対応し、エンタープライズ標準を満たすことができます。

Tableau Enterprise に含まれる機能の一覧と詳細については、Tableau Enterprise についてを参照してください。

> **注:** Tableau+ ライセンス エディションは、Tableau Cloud でのみ使用できます。Tableau+ に含まれる機能の一覧と詳細については、Tableau Cloud ヘルプの「Tableau+ について」を参照してください。

プライベート接続ライセンス

Tableau Enterprise または Tableau+ ライセンス エディションの Tableau Cloud をご利用の場合は、アドオン ライセンスとしてプライベート接続を購入できます。プライベート接続ごとにアドオン ライセンスが必要です。

プライベート接続を介してデータを送受信するには、データ使用量ブロックも購入する必要があります。プライベート接続のデータ使用量はテラバイト単位で課金されます。

毎月の使用量のサマリは、「アカウントの概要 (月次) FAQ」の説明の通りに送信されます。より頻繁に詳細な統計を必要とする Cloud 管理者は、アクティビティログを使用して使用量を確認できます。Tableau Cloud Manager のプライベート接続のリストに表示されるプライベート接続の**エンドポイント ID** は、アクティビティログ イベントの `endpointID` と一致しています。

機能ライセンス

> **重要:** 2024 年 9 月 16 日以降、Advanced Management と Data Management は独立したアドオン オプションとしては利用できなくなりました。Advanced Management 機能と Data Management 機能は、以前にこれらを購入した場合、または特定のライセンス エディション

(Tableau Enterprise (Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

機能ライセンスは、他のライセンスとは異なる方法で販売されます。独立ライセンスを含む機能は、展開内のすべてのユーザー(またはすべてのコア)でライセンスを取得する必要があります。展開には、ライセンスされている本番環境の Tableau Server インストールのほか、本番環境インストールをサポートし、ライセンスされている非本番環境の Tableau Server インストールが含まれています。

これらの機能は毎年ライセンスを取得するものであり、ライセンスという意味では、これらの機能をユーザーベースで使用できるかどうかについては「すべてか無しか」となります。

- データ管理
- Advanced Management
- ログインベースのライセンス管理

注: 更新可能なサブスクリプション ライセンスには、機能とロールの両方が 1 つのライセンスに含まれています。別の機能やそれに関連するロールのために、複数のライセンスやプロダクトキーをアクティブ化する必要はなくなりました。

データ管理

データ管理 ライセンスには単一の Tableau Server 展開用の Tableau Catalog と Tableau Prep Conductor が含まれており、これはロールベースまたはコアベースの場合があります。詳細については、データ管理 についてを参照してください。

データ管理 にはリソース コアが必要な場合があり、これは Prep Conductor のフローを実行するのに使用されるコンピューティング能力を指定します。コアベースのライセンスがあるサーバーでは、4 つ以上のリソース コアの購入が必要です。データ管理 のライセンスを参照してください。

tsm licenses list によって返される出力では、DATA MANAGEMENT フィールドに データ管理 のプロダクト キーが 1 つ表示され、適切なライセンスの下では `true` と表示されます。

Advanced Management

Tableau Server の Advanced Management は展開ごとにライセンスされ、これはロールベースまたはコアベースの場合があります。Advanced Management と含まれている機能の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

tsm licenses list によって返される出力では、SERVER MANAGEMENT ADD-ON フィールドに Advanced Management のプロダクトキーが 1 つ表示され、適切なライセンスの下では `true` と表示されます。

ログインベースのライセンス管理

ログインベースのライセンス管理 (LBLM) により、Tableau Desktop とTableau Prep Builder のライセンス発行が簡素化されます。これらの製品がは、プロダクトキーを使用するのではなく、Creator ライセンスを持つユーザーが Tableau Server で認証されるとアクティブ化されます。お客様が Tableau Server なしで Tableau Desktop または Tableau Prep Builder を使用している場合、LBLM は使用できません。

> **注**: Tableau Server バージョン 2023.1 以降、LBLM はオフライン展開でサポートされています。LBLM を有効にして Tableau Server を展開するには、実行許可 (ATR) サービスを使用するように Tableau Server を構成する必要があります。詳細については、オフラインでの Tableau Server の有効化を参照してください。

詳細については、ログインベースのライセンス管理を参照してください。

tsm licenses list によって返される出力では、LBLM フィールドにログインベースのライセンス管理のプロダクトキーが 1 つ表示され、適切なライセンスの下では `true` と表示されます。

更新可能なサブスクリプション ライセンス (USL)

更新可能なサブスクリプション ライセンスにより、さまざまな製品タイプ、容量、またはサブスクリプション期間に応じて複数の個別のプロダクトキーを管理する必要がなくなるため、Tableau Server のライセンス管理の要件が簡素化されます。

USL の基本を理解する

USL を利用すると、Tableau 展開のライセンスが付与されたすべての機能と属性 (「ライセンス資格」) が、単一の統一されたプロダクトキーとしてプロビジョニングされます。

さらに、USL は、後続のライセンス更新、アドオン、資格の変更の配信を、単一の更新可能なプロダクトキーに対して経時的に行われる更新に統合するため、指定された展開に完全なライセンス資格が一貫して反映されます。

各更新は、既存のプロダクトキーに新しい属性のレイヤーを追加するようなものと考えることができます。これらの階層化された各更新には、対応する一意の有効化 ID があります。サーバーインストールを Tableau のライセンスサービスと同期すると(プロダクトキーを有効化または更新して)、最新の有効化 ID がローカル サーバー インストールに取得され、展開の更新されたライセンス資格が反映されるようになります。

プロダクトキーの最新の有効化 ID を取得するために必要なプロセスは、展開環境のインターネット接続によって異なります。

オンラインまたは接続された環境での USL の有効化

オンラインまたは接続された環境では、展開のライセンス資格の有効化とロック解除に使用される USL プロダクトキーは時間が経っても不変です。

- 元の USL プロダクトキー (Tableau カスタマー ポータルに「キー名」として反映される) を有効化または更新すると、接続環境では、ライセンス資格に利用可能な最新の有効化 ID が自動的に取得されます。これは、Tableau のホスティング型ライセンス サービスとのライブ通信を通じて行われます。
- Tableau の実行承認 (ATR) ライセンス サービスを使用するように構成されたサーバー インストールの場合、これらの更新は、ATR サービスによる定期的なサーバー チェックイン中にバックグラウンドでサイレントにダウンロードされます。

オフラインまたは切断された環境での USL の有効化

オフラインまたは切断された環境では、Server とホスティングされた Tableau ライセンス サービスとの間に直接の通信がないため、USL プロダクトキーを使用して新しい有効化 ID の更新を自動

的に取得することはできません。

- 代わりに、管理者は、Tableau のオフラインのライセンス認証プロセスを通じて最新の有効化 ID を直接有効化し、展開のライセンス資格への更新を反映させる必要があります。
- 新しい有効化 ID がライセンス資格に追加されると、以前のライセンス認証 ID を有効化に使用することはできなくなります。以前の有効化 ID は機能的に古くなっています。
- この目的のために、現在の最新の有効化 ID は、Tableau カスタマー ポータルに「オフラインの有効化 ID」として常に反映されます。オフラインの有効化 ID は、USL プロダクト キーの [ライセンスの詳細] ビューに表示されています。
- ライセンス資格の更新ごとに新しい有効化 ID が生成されるため、オフラインの有効化 ID は時間の経過とともに繰り返し変更されます。

重要: オフラインでの Tableau の有効化では、お客様は最新のオフラインの有効化 ID を使用してソフトウェアの新規インストールを有効化する必要があります。

オフライン環境でのライセンス資格の更新管理

切断された環境では USL プロダクト キーを有効化するときに最新のオフラインの有効化 ID を利用するという要件を考慮すると、Tableau Server の新規または既存のインストールを有効化する前に、Tableau カスタマー ポータルから現時点で最新のオフラインの有効化 ID を取得することが重要です。これは最後に使用したものと一致しない可能性があります。

Salesforce のプロビジョニング システムは定期的に同期して、一連のプロセスを完全かつ正確に実行します。これにより、特定の購入やサブスクリプションの更新とは無関係に、ライセンス資格の新しいオフラインの有効化 ID を生成することができます。サーバー ソフトウェアの新規インストールを導入する前など、導入内のアクティブなライセンスを変更する*前に*、必ずカスタマー ポータルをチェックして現在のオフラインの有効化 ID を確認し、有効化が成功していることを確認してください。

オフラインでの有効化 ID のライフサイクルに関連する重要なポイントを次に示します。

- 有効化を成功させるには、*常に*
  a. 以前のオフラインの有効化 ID の無効化、および
  b. 現在のオフラインの有効化 ID の有効化が必要です。

ATR (オフライン) を使用するように構成されたサーバー インストールの場合、以前のオフラインの有効化 ID の無効化は自動的に処理されます。

- 新しいオフラインの有効化 ID を発行すると、以前のオフラインの有効化 ID を使用した新しい有効化は行えなくなります。

  そのため、導入環境でライセンスを更新する前に、必ずカスタマー ポータルを参照してください。

- 新しいオフラインの有効化 ID を使用しても、以前に USL プロダクト キーを有効化したサーバー環境には影響しません。

  - これらのインストールには、新しいオフラインの有効化 ID が直接適用されるまで、更新されたライセンス資格は反映されません。

USL のオフラインでの有効化手順

有効化の方法によっては、プロダクト キーまたはオフラインの有効化 ID を使用する必要がある場合があります。

- ATR を使用する場合は、プロダクト キーを使用します。
- レガシー FNO を使用する場合は、オフラインの有効化 ID を使用します。

オフラインまたは切断された環境で USL を有効化するには以下を実施します。

1. Tableau Server 展開の現在のプロダクト キーまたはオフラインの有効化 ID を特定します。

   Tableau カスタマー ポータルでオフラインの有効化 ID を見つけるには、**[ライセンス]** ペインに移動して [USL プロダクト キー] をクリックし、[ライセンスの詳細] ビューを開きます。20 桁の文字列には [Offline Activation ID (オフラインのライセンス認証 ID)] というラベルが付けられています。詳細については、ナレッジベースの記事「Tableau Server でオフラインでライセンス認証する正しいキーを見つける」を参照してください。

2. 有効化の方法に応じて、Tableau Server でプロダクト キーまたはオフラインの有効化 ID を有効化します。

「オフラインでの Tableau Server の有効化」のステップに従って、Tableau Server をオフラインで有効化します (USL プロダクトキーとUSL 以前のプロダクトキーの両方に適用されます)。

USL オフライン ライセンス資格の更新

オフライン環境でライセンス資格を変更するために必要な手順は、ご利用の有効化の方法 (ATR または従来の FNO) とTableau Server のバージョンによって異なります。

- ATR およびバージョン 2023.1.0 以上:

  ATR で有効化し、バージョン 2023.1 以降を使用している場合は、カスタマー ポータルからプロダクトキー名を使用して、ライセンス資格の変更を更新します。その他の必要なアクションは ATR サービスが実施します。

- レガシー FNO での有効化 *または* バージョン 2023.1.0 より前:

  1. カスタマー ポータルから現在のオフラインの有効化 ID を取得します。
  2. Tableau Server を停止します。

     ```
     tsm stop
     ```

  3. サーバー上の既存の有効化されたキーのオフラインでの無効化を開始します。このオフラインの有効化 ID は、もうご利用の環境では使用できなくなりました。オフラインでの無効化により、返送要求ファイルが作成されます。このファイルを保存します。今後のステップで必要になります。
  4. 新しいオフラインの有効化 ID に対するオフラインでの有効化要求を開始します。これにより、有効化要求が作成されます。次のステップのためにこのファイルを保存します。
  5. 返送要求ファイルと有効化要求ファイルの両方を、インターネットにアクセスできるコンピューターに転送します。
  6. [オフラインでの有効化] ページに移動し、次の順序で 2 つの要求を開始します。

a. 返送要求ファイルをアップロードして応答ファイルをダウンロードし、保存して続行します。

b. 有効化要求ファイルをアップロードして応答ファイルをダウンロードし、保存して続行します。

7. 返送応答ファイルと有効化応答ファイルの両方を、切断された Tableau Server コンピューターに転送します。

8. 返送応答ファイルを適用します。

9. 有効化応答ファイルを適用します。

10. Tableau Server を起動します。

```
tsm start
```

サーバー ライセンスの表示

サーバー管理者は、Tableau Server のライセンス情報やプロダクト キー情報を表示できます。

Tableau Server Web UI からのライセンスの表示

Tableau Server の [ライセンス] ページに移動する方法は、単一サイトかマルチ サイトかで異なります。

- 単一サイトがあるサーバーで、**[設定]** および **[ライセンス]** をクリックします。
- 複数サイト サーバーのサイト メニューで **[すべてのサイトを管理]**、**[設定]**、および **[ライセンス]** をクリックします。

> **注: [すべてのサイトを管理]** オプションは、サーバー管理者としてサインインした場合にのみ表示されます。

このページには、ユーザーベース (期間) ライセンスまたはコアベース ライセンスを含め、サーバーでライセンス認証されているすべてのライセンスに関する情報が表示されます。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   http://<tsm-computer-name>:8850

2. **[適用]**、**[ライセンス発行]** の順にクリックします。

   表にはプロダクトキー、有効期限日、およびメンテナンスの有効期限が表示されます。

> **注**: TSM の Web UI では限られた量のライセンス情報しか得られません。ユーザーベースのライセンス (Creator、Explorer、および Viewer (ビューアー)) の各タイプの数を含め、その他のライセンス情報を表示するには、TSM CLI または Tableau Server の Web UI を使用します。

TSM CLI の使用

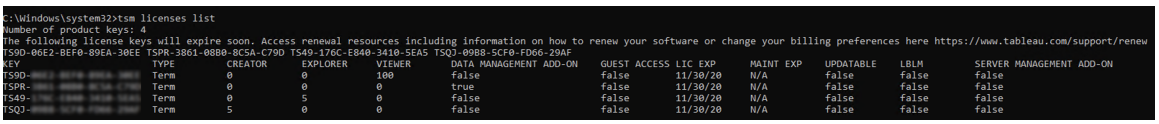
1. 最初のノード (TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを実行します。

   ```
   tsm licenses list
   ```

このコマンドは、Tableau Server の展開でライセンス認証されたライセンスを一覧表示します。

たとえば、5 つの Creator ライセンス、5 つの Explorer ライセンス、100 個の Viewer ライセンス、およびデータ管理 を持つサーバーは、次のようなコマンド出力を生成します。



次のフィールドが返されます。

- **KEY**: ライセンスを識別するグローバルに一意の 16 文字の文字列。
- **TYPE**: ライセンスの種類についての説明。
  - 期間: 期間ライセンスはサブスクリプション スケジュールにマップされ、更新する必要があります。有効期限は、[LIC EXP] フィールドに一覧表示されます。
  - 永続: 永続ライセンスは一度購入したら更新する必要はありませんが、MAINT EXP (メンテナンスの有効期限) は更新する必要があります。
  - コア: コア ライセンスは、特定の Tableau Server サービスを実行しているコンピューター上のコアの数にマップされるライセンスです。コア ライセンスを使用すると、ゲスト ユーザーがサーバー上のビューにアクセスしたり、他の Web サーバーに埋め込んだりできます。また、コア ライセンスでは、Explorer ユーザーと Viewer ユーザーを無制限に許可できます。
- **CREATOR**: Tableau Server の展開で発行された Creator ライセンスの数。
- **EXPLORER**: Tableau Server の展開で発行された Explorer ライセンスの数。
- **VIEWER**: Tableau Server の展開で発行された Viewer ライセンスの数。
- **DATA MANAGEMENT**: Tableau Server がデータ管理 (`True`/`False`) でライセンスされています。データ管理 についてを参照してください。
- **GUEST ACCESS**: Tableau Server がゲスト ユーザーに対してライセンスされています。ゲスト ユーザーを参照してください。ゲスト ユーザーを利用するには、コア ライセンスが必要です。[TYPE] フィールドを参照してください。
- **LIC EXP**: ライセンスの有効期限が切れ、Tableau Server が動作を停止する日付。期間ライセンスの有効期限が切れます。[TYPE] フィールドを参照してください。ライセンスを更新するには、Tableau のカスタマー ポータルにアクセスしてください。
- **MAINT EXP**: レガシー永続ライセンス (TYPE = 永続) にのみ適用されます。期間ライセンスの場合、このフィールドは `N/A` と出力されます。[MAINT EXP] には、Tableau Server 展開に対するメンテナンス契約の有効期限が切れる日付が表示されます。ライセンスのメンテナンス キーを更新するには、プロダクト キーの有効期限と属性の更新を参照してください。Tableau のカスタマー ポータルにアクセスしてメンテナンスの購入履歴を表示し、追加のメンテナンスを購入してください。
- **UPDATABLE**:更新可能なサブスクリプション ライセンスかどうかを指定します (`True`/`False`)。
- **LBLM**: Tableau Server の展開に対してログインベースのライセンス管理 (LBLM) が有効かどうかを指定します (`True`/`False`)。有効にすると、ユーザーはプロダクト キーを入力するのではなく、Tableau Server にログインすることで、Tableau Desktop または Prep のインスタンスのライセンスを取得できます。LBLM の詳細については、ログインベースのライセンス管理を参照してください。

- **SERVER MANAGEMENT**: Tableau Server が Advanced Management (旧称 Server Management Add-on) (`TrueFalse`) でライセンスされています。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

> **注**: Creator、Explorer、および Viewer (ビューアー) ユーザーのライセンスの期間は、存在する場合はユーザーベースのライセンス (期間 ライセンス) の期間に従って設定されます。そのため、コアベースのライセンスが 1 つしかないサーバーでは Explorer ユーザーと Viewer (ビューアー) ユーザー、ゲスト ユーザー アクセスは無制限ですが、Creator ユーザーはありません。詳細については、コアベース ライセンスが設定されているサーバーでロールベース ライセンスを使用するを参照してください。

プロダクト キーの有効期限と属性の更新

Server ATR と更新可能なサブスクリプション ライセンス (USL) を使用している場合、新しいサブスクリプション期間を購入するとき、またはロールや機能を展開に追加するときに、プロダクト キーを更新する必要はありません。これは、USL プロダクト キーを更新して、Tableau Server ライセンス容量、機能、サブスクリプション期間に変更を徐々に反映できるためであり、Server ATR はバックグラウンド プロセスとしてプロダクト キーを自動的に更新します。

> **注:** サブスクリプションの更新に合わせて USL を最近有効にした場合、ここで説明されているようにライセンスを更新することはできません。展開内でアクティブ化する必要がある新しい USL 互換のプロダクト キーについては、Tableau カスタマー ポータルを参照してください。「Tableau Server のライセンス認証と登録」をご覧ください。元のプロダクト キーは更新されていないため、展開で使用できなくなっています。

Server ATR を使用していない場合は、プロダクト キーを手動で更新できます。有効期限が切れる前にサブスクリプション (期間) プロダクト キーを更新した場合、プロダクト キーは変更されませんが、有効期限は変更されます。これを行うことで、Tableau カスタマー ポータルにリストされているプロダクト キーと、Tableau Server TSM プロダクト キーにリストされているプロダクト キーとの間に不一致が生じる可能性があります。プロダクト キーは、有効期限が切れると、またはその後すぐに変

更されます。サブスクリプション(期間)プロダクトキーが更新されずに有効期限が切れると、Tableau は動作を停止するため、アクティブ化の方法として Server ATR と手動のどちらを使用しているかにかかわらず、Tableau カスタマー ポータルから新しいプロダクトキーをアクティブ化して、Tableau Server をもう一度ライセンス認証する必要があります。

一方、プロダクトキーが永続的(レガシー)であり、メンテナンスの有効期限が切れている場合、Tableau は引き続き動作しますが、メンテナンスが更新されるまでアップグレードにアクセスすることはできません。メンテナンスを更新した後、Server ATR を使用していない場合は、既存のプロダクトキーを更新して、メンテナンスの有効期限を更新する必要があります。プロダクトキーは変更されません。さまざまなプロダクトキーの種類と関連するライセンスの詳細については、tsm licenses list を参照してください。

> **注**: このトピックでは、Tableau Server、Advanced Management、データ管理 の有効期限を更新する方法について説明します。Tableau Desktop のメンテナンス日の更新の詳細については、Tableau Desktop とTableau Prep 展開ガイドの「プロダクトキーの更新」を参照してください。

はじめる前に

ライセンスの有効期限を確認します。有効期限は、以下の TSM Web インターフェイスの手順に従うか、または CLI で tsm licenses list を実行することによって表示できます。

- Tableau カスタマーポータル に表示される日付とその日付を比較します。
- ポータルに予定されている日付が表示されない場合は、カスタマー サクセス に連絡してください。
- ライセンスを更新するには、Tableau の更新 Web ページにアクセスしてください。

TSM の日付が、Tableau カスタマー ポータルに表示されている日付と一致せず、次の更新操作が失敗した場合は、Tableau サポートにお問い合わせください。

> **注:** オフライン環境でプロダクトキーを更新するには、Tableau カスタマー ポータルにアクセスして、最新の更新のオフライン アクティブ化 ID を取得してから、プロダクトキーをアクティブ化し

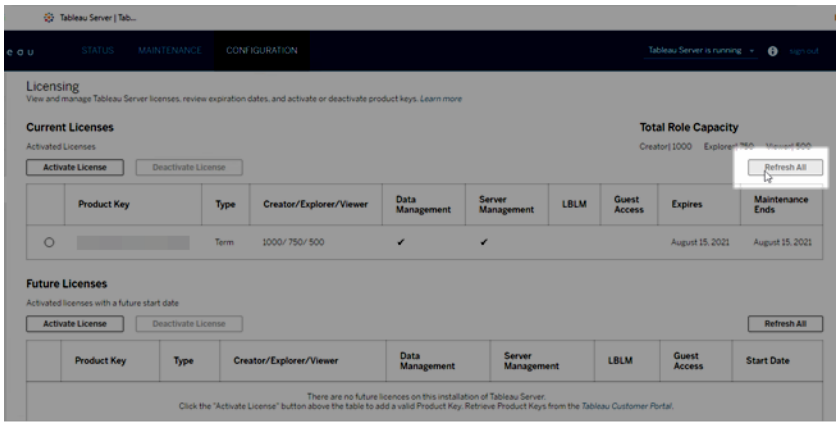
てください。オフラインでのアクティブ化の詳細については、オフラインでの Tableau Server の有効化を参照してください。Tableau Server を起動する前に、すべてのプロダクトキーをアクティブにしてください。そうしないと、一部のユーザーがライセンスなしになる可能性があります。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

2. **[構成]** および **[ライセンス発行]** をクリックし、**[すべて更新]** をクリックします。



TSM CLI の使用

1. 最初のノード(TSM がインストールされているノード)で管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを実行します。

   ```
   tsm licenses refresh
   ```

Tableau Server へのライセンスの追加

ユーザー数 (ユーザー ベースのライセンスの場合) またはコア数 (コア ベースのライセンスの場合) を増やすには、Tableau Server インストールに容量を追加しなければならない場合があります。

Tableau は、既存の Tableau Server インストールに容量を追加する新しいプロダクトキーを提供します。付与されているライセンスと組み合わせて使用するには、このキーをライセンス認証し、既存のプロダクトキーと一緒に使用する必要があります。各機能 (例:「データ管理」) とライセンスタイプ (例:「Explorer」) には、キーが必要です。キーとライセンスの関係の詳細については tsm licenses list を参照してください。

次の手順にしたがって、Tableau Server にプロダクトキーを追加します。

Tableau Server がインターネットに接続されていない場合は、オフラインアクティベーションを実行する必要があります。オフラインでの Tableau Server の有効化を参照してください。

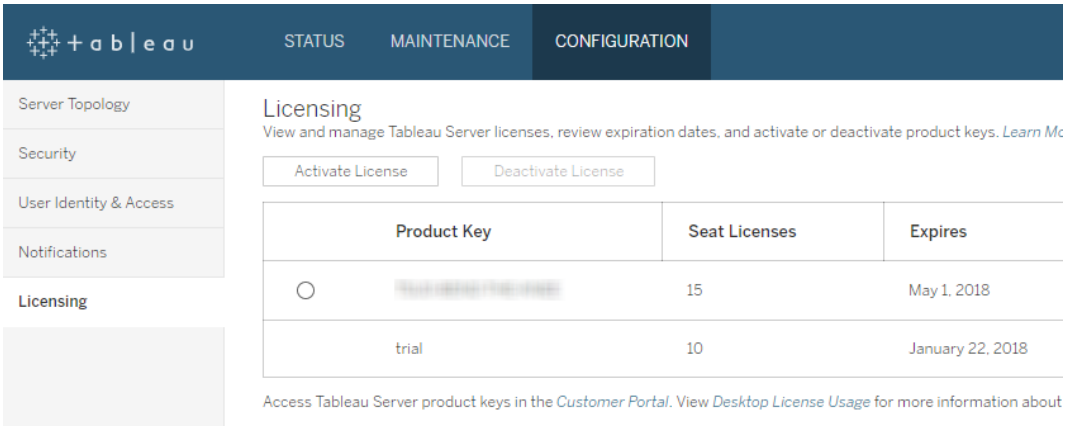
注: Tableau Server バージョン 2021.1 以降にアップグレードした場合、容量を追加するときに Tableau Server を再起動する必要はありません。詳細については、ゼロダウンタイムのライセンス発行を参照してください。

TSM Web インターフェイスの使用

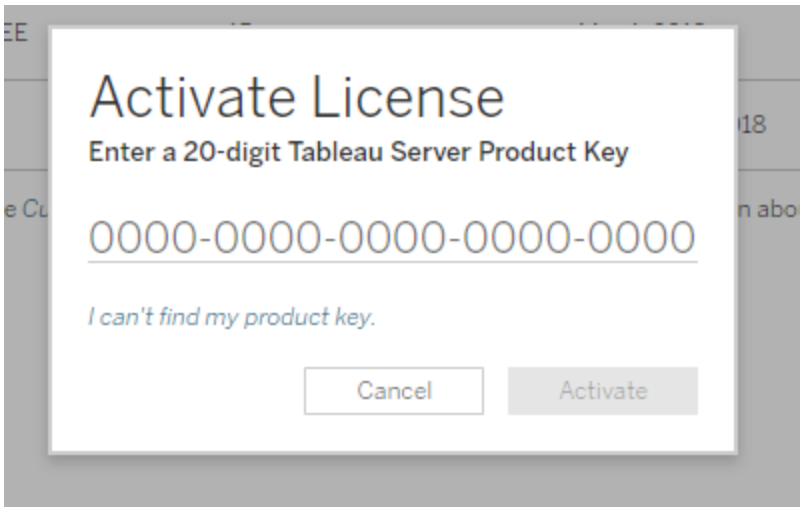
1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

2. **[構成]** タブで **[ライセンス発行]** をクリックし、**[ライセンス認証]** をクリックします。



3. 新しいプロダクトキーを入力するか貼り付け、**[ライセンス認証]** をクリックします。



4. ライセンスをアクティブ化した後、Tableau への登録を求められる場合があります。[登録] ページで情報をフィールドに入力し、**[登録]** をクリックします。

Register with Tableau. All fields are required.

Contact Information

First Name

Last Name

Phone Number

Email

Company Information

Organization

Industry

Company Size

Department

Job Role

Region Information

City

Postal Code

Country/Region

State/Province

TSM CLI の使用

1. プロダクトキーをコンピューターにコピーします。
2. 次のコマンドを実行します。

   ```
   tsm licenses activate --license-key <license key>
   ```

3. ライセンスをアクティブ化した後、Tableau Server の登録が必要になる場合があります。登録するには、登録ファイルを作成して tsm register コマンドを使用してオプションとしてこれを渡します。

   a. 次のコマンドを実行して編集可能なテンプレートを生成します。

      ```
      tsm register --template > /path/to/<registration_file>.json
      ```

   b. 完成した登録ファイルを作成するために、テンプレートファイルを編集します。

      次の例は、必要な形式で作成された登録ファイルです。

      ```
      {
       "zip" : "97403",
       "country" : "USA",
       "city" : "Springfield",
       "last_name" : "Simpson",
       "industry" : "Energy",
       "eula" : "yes",
       "title" : "Safety Inspection Engineer",
       "phone" : "5558675309",
       "company" : "Example",
       "state" : "OR",
       "department" : "Engineering",
       "first_name" : "Homer",
       "email" : "homer@example.com"
      }
      ```

   c. ファイルの変更を保存したら、--file オプションでこのファイルを渡し、Tableau Server を登録します。

```
tsm register --file /path/to/<registration_file>.json
```

例:

```
tsm register --file /usr/share/tableau-reg-file.json
```

オフラインでの Tableau Server の有効化

Tableau Server をインストールする場合は、少なくとも 1 つのプロダクトキーをアクティブ化する必要がありますが、Tableau カスタマーポータルにあるすべての Tableau Server ライセンスをアクティブ化することをお勧めします。これにより、サーバーがアクティブになり、ユーザーに割り当てることができるライセンスレベルの数が指定されます。ATR を使用した Tableau Server でオフラインアクティベーションを行う場合、Tableau カスタマーポータルの [Offline Activation Id (オフラインアクティベーション ID)] フィールドに記載されているプロダクトキーをアクティブ化する必要があります。適切なキーを見つける方法の詳細については、ナレッジ記事「Tableau Server で有効化するための正しいキーを見つける」を参照してください。

また、サーバーに容量を追加したり、新しいプロダクトキーを取得したりする場合は、Tableau Server のインストール後にライセンス認証が必要になる場合もあります。プロダクトキーをお持ちでない場合は、「Tableau Customer Account Center」からプロダクトキーを入手できます。

> **注:** Tableau Server を起動した後にプロダクトキーをライセンス認証するには、Tableau Server を再起動して変更を有効にする必要があります。

ほとんどの場合、Tableau サービスマネージャー (TSM) ライセンスページを使用して、インストール中やインストール後に Tableau Server から直接キーをライセンス認証できますが、これを実行できない状況もあります。たとえば、コンピューターがインターネットに接続されていない場合や、イントラネット外へのアクセスを制限するファイアウォールがある場合などです。このような場合は、オフラインでライセンス認証を実行する必要があります。
Tableau Server in a Container では、サーバー ATR を使用したライセンス認証のみをサポートします。Server ATR を使用したオフラインアクティベーションは、2023.1 以降でサポートされています。この機能はコンテナで利用できますが、追加の手順と承認が必要です。エアギャップ環境またはオフライン環境のコンテナで Tableau Server を実行する必要がある場合、詳細についてはアカ

ウント担当者にお問い合わせください。

オフラインでのアクティブ化とログインベースのライセンス管理 (LBLM)

Tableau Server バージョン 2023.1.0 以降、サーバーが実行承認 (ATR) サービスを使用するように構成されている場合、LBLM でオフラインでのライセンス認証を行うことができます。ATR サービスを使用するように Tableau Server を構成できるのは、新規インストール時のみです。既存のサーバー インストールをアップグレードするお客様は、バージョン 2023.1.0 以降の新しい Tableau Server のインスタンスをインストールし、既存のインストールのバックアップをその新しいインスタンスに復元する必要があります。

このプロセスの詳細については、「Tableau Server のアップグレードにブルー/グリーン アプローチを使用する」を参照してください。

ATR サービスの詳細については、「実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する」を参照してください。

オフラインでの有効化と更新可能なサブスクリプション ライセンス (USL)

更新可能なサブスクリプション ライセンスのオフラインでの有効化には特別な手順が必要です。詳細については、オフラインまたは切断された環境での USL の有効化を参照してください。

オフラインでのライセンス認証が必要となる場合、次の 2 つのシナリオがあります。

- インストール時にオフラインでライセンス認証する - Tableau Server のインストール時にオフラインでのライセンス認証を実施します。
- インストール後にオフラインでライセンス認証する - サーバーをインストールして実行した後にオフラインでのライセンス認証を実施します。

オフライン アクティブ化の概要

Tableau Server のオフラインでのアクティブ化は、アクティブ化の方法 (実行承認サービス (ATR) を使用する場合としない場合) によって異なります。

ATR *を使用しない*オフラインでのアクティブ化では、次のステップが含まれます。

1. オフライン アクティブ化要求ファイルを生成します。
2. オフラインでのアクティブ化要求ファイルを、インターネットにアクセスできるコンピューターにコピーします。
3. オフラインでのアクティブ化要求ファイルを、Tableau ライセンス認証 Web サイトにアップロードします。
4. 作成されたオフラインでのアクティブ化応答ファイルを Web サイトからダウンロードします。このファイルを使用して Tableau Server をアクティブ化します。

ATR *を使用した*オフラインでのアクティブ化では、オフラインでのアクティブ化 ID ではなく、プロダクトキー名を使用する必要があります。

オフライン アクティブ化ファイル名の変更

Tableau Server バージョン 2023.1 以降、Tableau ライセンス発行システムでは 2 つの基本的なライセンス発行テクノロジーがサポートされています。管理上の観点から見ると、2 つのシステムの構成の違いは、オフラインでのアクティブ化で生成および使用されるファイルの種類のみです。ライセンス発行テクノロジーは、Tableau Server の初期インストール時に決定され、インストール後に変更することはできません。

ライセンス発行テクノロジーのレガシー バージョンは、現在もサポートされており、FlexNet と呼ばれます。このテクノロジーの最新バージョンは、ATR と呼ばれます。詳細については、「実行承認 (ATR) サービスを使用して Tableau Server をライセンス認証する」を参照してください。次の表は、各テクノロジーのファイル名の命名規則について説明したものです。また、一般的なリファレンスも含まれています。

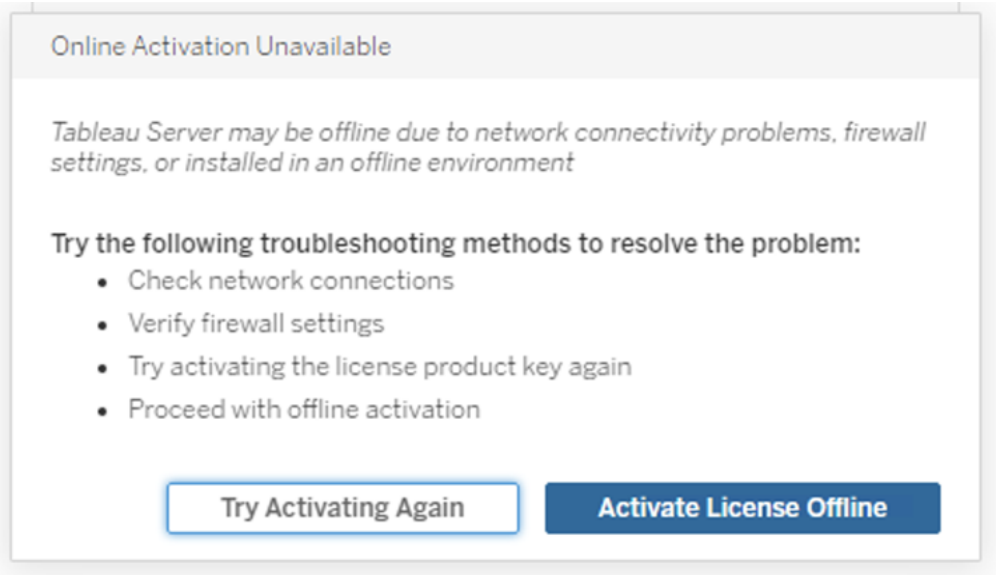
汎用ファイル名	サーバー ATR ファイル名	FlexNet ファイル名
OfflineActivationRequest	OfflineActivationRequestFile_yyyyMMdd.hhmmss.json	TableauOfflineActivationRequest.tlq
OfflineActivationResponse	OfflineActivationLicensingAtrs.zip	activation.tlf

> **注:** このドキュメントは Tableau Server の複数のバージョンに対応しているため、このトピックの残りの部分では一般的なファイル名のリファレンス (OfflineActivationRequest と OfflineActivationResponse) を使用します。Tableau Server のインストールで使用されるライセンス発行テクノロジーは次のステップで生成されるファイルの種類によって特定できます。

TSM Web インターフェイスの使用

TSM のライセンスページからプロダクトキーを認証しようとした際にオンラインでのライセンス認証を実行できないことを示すダイアログが表示された場合は、キーをオフラインで認証できます。オフラインでのライセンス認証プロセスは、プロダクトキーごとに 1 回ずつ実施する必要があります。

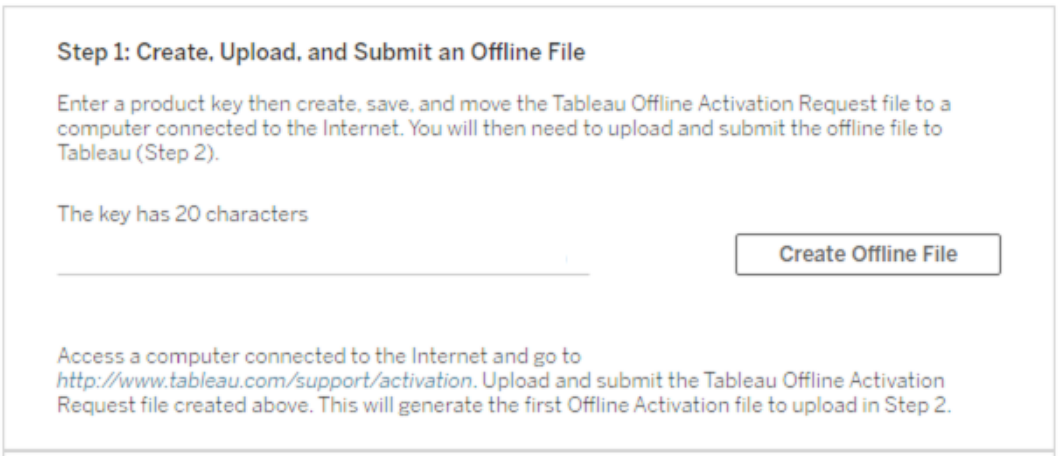
1. **[オフラインでライセンス認証]** をクリックします。



2. プロダクトキーのオフラインでのライセンス認証要求ファイル (OfflineActivationRequest) を作成します。

   Tableau ライセンス認証 Web サイトにアップロードする OfflineActivationRequest ファイルを作成します。プロダクトキーがフォームに事前に入力されていない場合は、キーを入力して

[オフライン ファイルの作成] をクリックし、ローカル コンピューター上に OfflineActivationRequest ファイルを生成します。



OfflineActivationRequest ファイルをインターネットにアクセスできるコンピューターにコピーします。ライセンス認証応答ファイルを生成するには、このファイルを Tableau ライセンス認証 Web サイトにアップロードする必要があります。

3. OfflineActivationRequest ファイルをアップロードして送信します。

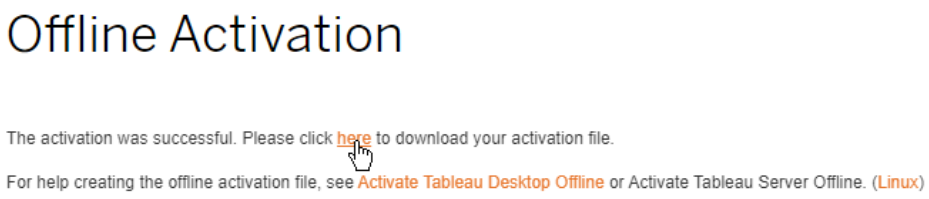
   OfflineActivationRequest ファイルをアップロードして Tableau ライセンス認証 Web サイトに送信します。これにより、ライセンス認証応答ファイル (OfflineActivationResponse) が自動的に生成されます。これを Tableau Server コンピューターにダウンロードしてコピーします。

   a. OfflineActivationRequest ファイルをコピーしたコンピューターで、ブラウザーを開き http://www.tableau.com/ja-jp/support/activation に移動して、Tableau サポートの [ライセンス認証] ページを開きます。

   b. [オフラインでのライセンス認証] ページで、**[ファイルの選択]** をクリックして [OfflineActivationRequest] ファイルを選択します。

   c. **[ライセンス認証ファイルのアップロード]** をクリックして、ファイルを Tableau ライセンス認証 Web サイトに送信します。

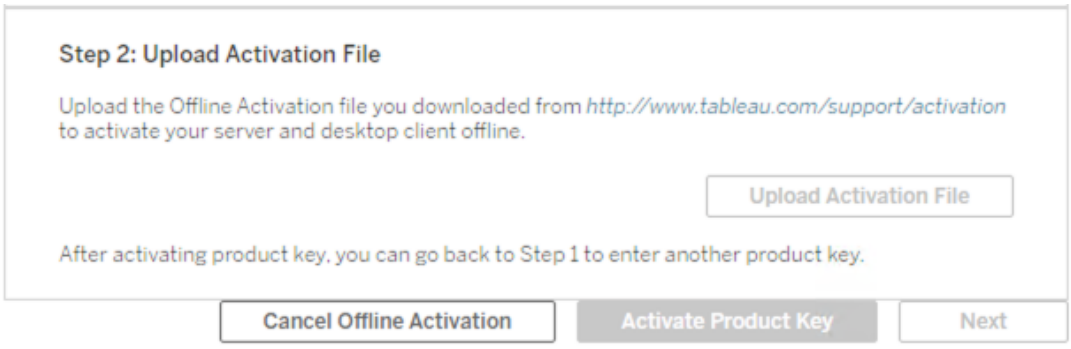
d. here のリンクをクリックして OfflineActivationResponse ファイルをご利用のコンピューターにダウンロードします。



e. OfflineActivationResponse ファイルを、Tableau Server がインストールされているコンピューターにコピーします。

4. OfflineActivationResponse ファイルをアップロードします。

   Tableau Server コンピューターで **[ライセンス認証ファイルのアップロード]** をクリックし、OfflineActivationResponse ファイルを Tableau Server にアップロードします。これを正常に実行すると、**[プロダクト キーのアクティブ化]** ボタンが有効になります。



5. **[プロダクト キーのアクティブ化]** をクリックしてオフラインでのアクティブ化を完了します。

6. (Tableau Server を初めてインストールする場合は、この手順をスキップします)。

   ライセンスの変更を有効にするには、Tableau Server を再起動します。

TSM CLI の使用

ステップ 1.Tableau サービス マネージャーへのログイン

手順を進めるには、Tableau サービス マネージャー (TSM) にログインする必要があります。TSM にログインするには、次のコマンドを実行します。

```
tsm login -u <username>
```

ログインできない場合

残りのインストールの構成に使用するアカウントは、初期化中に作成された tsmadmin グループのメンバーでなければなりません。tsmadmin グループのユーザー アカウントを表示するには、次のコマンドを実行します。

```
grep tsmadmin /etc/group
```

ユーザー アカウントがグループにない場合、次のコマンドを実行してユーザーを tsmadmin グループに追加します。

```
sudo usermod -G tsmadmin -a <username>
```

ユーザーを tsmadmin グループに追加した後に、tsm login コマンドを実行します。

ステップ 2.オフライン アクティブ化要求ファイルを生成する

1. 最初のノードで、ターミナル セッションを開きます。
2. オフラインのライセンス認証ファイルを取得するには、次のコマンドを入力します。

   ```
   tsm licenses get-offline-activation-file -k <product-key> -o
   <target-directory>
   ```

   プロダクト キーは「Tableau カスタマー ポータル」で取得できます。ターゲット ディレクトリが既に存在している必要があります。
3. オフライン アクティブ化ファイル (OfflineActivationRequest) をターゲット ディレクトリからコピーして、インターネットにアクセスできるコンピューターに配置します。

ステップ 3. オフライン アクティブ化要求を Tableau アクティベーション Web サイトにアップロードする

1. インターネットにアクセスできるコンピューターで、Tableau の「アクティブ化」ページに移動します。
2. 手順を完了し、OfflineActivationRequest ファイルをアップロードします。

   これにより、アクティブ化応答ファイル (OfflineActivationResponse) が作成されます。
3. Tableau アクティベーション Web サイトから OfflineActivationResponse ファイルをダウンロードします。

ステップ 4. ライセンスを初期化または認証します

1. OfflineActivationResponse ファイルを Tableau Server コンピューターからアクセスできる場所にコピーします。
2. 次のコマンドを実行します。

   ```
   tsm licenses activate -f <path-and-activation-file>
   ```

   注意: ATR を使用して Tableau Server を有効化する場合、`<path-and-activation-file>` は、パッケージ化された OfflineActivationResponse `.zip` ファイルを指す必要があります。このコマンドを実行する前に、OfflineActivationResponse ファイルを解凍しないでください。
3. (Tableau Server を初めてインストールする場合は、この手順をスキップします)。

   ライセンスの変更を有効にするには、Tableau Server を再起動します。

   ```
   tsm restart
   ```
4. (オプション) すべてのライセンスがアクティブ化されていることを検証するには、次のコマンドを実行します。

   ```
   tsm licenses list
   ```

   上記のステップを完了すると、成功メッセージが表示されます。

   ```
   Activation successful.
   ```

Tableau Server はライセンス認証されています。さらにサポートが必要な場合は、Tableau 技術サポートにお問い合わせください。

プロダクトキーの認証解除

プロダクトキーを非アクティブ化する必要があるシナリオがいくつかあります。

- ハードウェア構成を変更する
- プロダクトキーを変更する
- プロダクトキーを新規インストールに移動する

はじめる前に

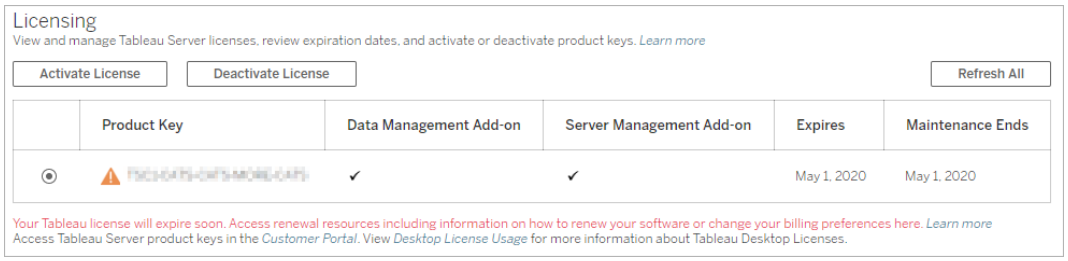
適切なプロダクトキーを削除しようとしていることを確認してください。CLI で tsm licenses list を実行することで、ライセンスの詳細を表示できます。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   ```
   https://<tsm-computer-name>:8850
   ```

2. **[構成]** の次に **[ライセンス発行]** をクリックします。



3. 認証解除するプロダクトキーを選択し、**[ライセンス認証を解除]** をクリックします。

4. キーが認証解除されたら、Tableau Server を再起動します。

> **注:** 再起動する前に他のプロダクトキーが有効化されていない場合、Tableau Server を使用可能な状態で再起動することはできません。この場合、TSM で新しいプロダクトキーを有効にするまで Tableau Server を使用することはできません。

TSM CLI の使用

1. 最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

```
tsm licenses deactivate --license-key <product-key>

tsm restart
```

オフラインでの Tableau Server のライセンス認証解除

Tableau Server がインターネットに接続されていない場合は、このトピックの指示を使用して Tableau Server のライセンス認証を解除できます。このライセンス認証プロセスを完了するには、インターネットに接続できる2 台目の PC を使用する必要があります。

> **注:** 現時点では、Tableau Server の展開で ATR サービスが有効になっている場合、Tableau Server を非アクティブ化することはできません。

1. Tableau Server の最初のノードで、次のステップで作成するオフラインでのライセンス認証解除ファイルを保存するディレクトリを作成します。
2. Tableau 管理者アカウントを使用して TSM にログインし、以下のコマンドを実行します。

```
tsm licenses get-offline-deactivation-file -k <productkey> -o
<deactivation-file-directory>
```

3. `TableauOfflineDeactivationRequest.tlq` ファイルを、前のコマンドで指定したライセンス認証解除ファイルのディレクトリから、インターネットにアクセス可能な信頼できる PC に移動します。

4. インターネットにアクセス可能な信頼できる PC で Web ブラウザーを開き、Tableau の [プロダクト アクティベーション] ページへアクセスします。そのページの指示に従って、`TableauOfflineDeactivationRequest.tlq` ファイルを送信します。

   > **重要:** このプロセスは、Microsoft Edgeブラウザでは機能しません。

5. プロンプトが表示される場合は、**[プロダクト アクティベーション]** ページからプロダクト キーのリターン ファイル (`return.tlr`) を保存します。

6. プロダクト キーのリターン ファイル (`return.tlr`) を、インターネットに接続可能な信頼できるコンピューターから、ライセンス サーバー サービスを実行する Tableau Server の初期ノードに移動します。

7. Tableau 管理者アカウントを使用して TSM にログインし、以下のコマンドを実行します。

   ```
   tsm licenses deactivate -f <path-to-license-key-return-file>

   tsm restart
   ```

ライセンス発行タスクの自動化

`tsm licenses` を使用して、ライセンス タスクを管理します (例:Tableau Server プロダクト キーのオンラインまたはオフラインでのライセンス認証/ライセンス認証解除、オフラインでのライセンス認証/ライセンス認証解除でのファイルの関連付け)。ただし、Tableau Server は既に展開して構成する必要があります。これらのライセンス タスクは、Tableau サービス マネージャー API を使用して自動化できます。

`C:\ProgramData\Tableau\Tableau Server\data\tabsvc\logs\`

ライセンス発行のトラブルシューティング

このトピックでは、Tableau Server のライセンス発行に関連する問題のトラブルシューティングの指示を記載しています。

ライセンスのないサーバーの処理

Tableau では 2 つのライセンス発行モデルを提供しています。1 つはロールベース モデルで、もう1 つはコアベース モデルですロールベースおよびコアベースのライセンス発行の詳細については、ライセンス発行の概要を参照してください。

ロールベース ライセンスでは、アクティブな各ユーザー アカウントがライセンスの対象となる必要があります。ロールベース ライセンスでは、容量が定義されているか、許可されるユーザー数が定義されています。各ユーザーにはサーバー上で一意のユーザー名が割り当てられ、サーバーに接続するときにそのユーザー名で識別する必要があります。

コアベース ライセンス発行では、システム内のユーザー アカウント数に制限はありませんが、Tableau Serverが使用できるプロセッサー コアの数が制限されます。Tableau Server を 1 つまたは複数のマシンにインストールしてクラスタを作成できますが、すべてのマシンの合計コア数がライセンスでカバーされているコア数を超えないこと、および特定のマシン上のコアがすべてライセンスでカバーされていることという制限があります。

ライセンスを解除されたユーザーベースのサーバー

サーバーがロールベース ライセンスを解除される最も一般的な理由は、プロダクト キーまたは メンテナンス契約の有効期限切れです。

ライセンスを解除されたコアベースのサーバー

コアベースのサーバーがライセンスなしになることには、さまざまな理由が考えられます。たとえば、プロダクト キーが有効期限切れになる場合や、ライセンス プロセスを実行中の Tableau Server ノードで、ライセンス マネージャー サービスを実行中の Tableau Server ノードに連絡を取ることができない場合などです。ライセンス プロセスの詳細については、Tableau Server プロセスを参照してください。

サーバーのライセンスが解除されると、サーバーを起動または管理できません。ただし、tsm licenses コマンドを使用してライセンスを管理できます。

ライセンスを解除されたサーバー管理者

すべての Tableau Server 管理者には、ユーザー ライセンスが必要です。Tableau Server 管理者は、使用可能な最高のロールを常に使用します。Creator プロダクト キーが認証されると、Tableau Server 管理者がこのロールを担います。Tableau Server で使用可能な最高のロールが Explorer の場合、サーバー管理者は Explorer ロールを担います。Creator ライセンスがサーバーに追加されると、Explorer ライセンスを使用している既存のサーバー管理者アカウントは自動的に Creator ライセンスを使用するように変換されます。

TSM 管理者アカウントにはライセンスは必要ありません。

サーバー管理者が使用しているライセンスの有効期限が切れると、アカウントはライセンスを解除され、サインインできなくなります。

サーバーの管理者のライセンスの有効期限を確認します。

- tsm licenses list を実行します。
- Tableau カスタマーポータル に表示される日付とその日付を比較します。
- ポータルに予定されている日付が表示されない場合は、カスタマー サクセス に連絡してください。
- ライセンスを更新するには、Tableau の更新 Web ページにアクセスしてください。
- tsm licenses activate コマンドを実行して、管理者アカウントの新しいライセンスをアクティブ化します。

TSM の日付がポータルの日付と一致し、次の更新操作が失敗した場合は、Tableau サポート にお問い合わせください。

管理者アカウントのライセンスの有効期限が切れている場合、または間もなく期限切れになる場合は、アカウントの新しいライセンスをアクティブ化する必要があります。または、管理者以外のユーザーのライセンスを解除して、サーバー管理者アカウント用にライセンスを解放することもできます。

Tableau Server 管理者が Creator、Explorer、または Viewer ライセンスを使用しており、それらのライセンスの有効期限が切れる場合、利用可能であれば同じ種類の別のライセンスを使用します。ライセンス シートが利用できない場合、ユーザーは「ライセンスなし」になります。

重要: 新しいライセンスをアクティブ化するか、サーバー管理者アカウントのサイト ロールを転送するまで、Tableau Server を再起動しないでください。

ロールベース ライセンス発行のトラブルシューティング

このセクションでは、ロールベースの Viewer、Explorer、および Creator ライセンスを Tableau Server または Tableau Cloud に追加する場合、またはこれらのライセンスの有効期限が切れる場合に起こる問題の解決に関する情報を提供します。利用可能な最高のライセンス タイプは Creator で、次に Explorer、最後に Viewer と続きます。ロールベース ライセンス発行の詳細については、ライセンス発行の概要を参照してください。

ライセンスの有効期限切れのためユーザーまたは管理者がライセンスなしである

ユーザーが予期せずライセンスなしになったり、別のサイト ロールに移行することがないよう、現在使用しているライセンスの有効期限が切れる前に、以下のうちいずれかを常に実行する必要があります。

- 置換ライセンスを更新してライセンス認証する。ユーザーが Creator、Explorer、または Viewer (ビューアー) ライセンスを使用しており、それらのライセンスの有効期限が切れる場合、利用可能であれば同じ種類の別のライセンスを使用します。
- それらのユーザーのサイト ロールを、有効期限が切れていないライセンスの使用を許可するように変更する。

別のライセンスを要求するようにサイト ロールを変更する方法については、ユーザーのサイト ロールの設定を参照してください。

新しいライセンスへのユーザーの再割り当ては、以下の論理で管理されます。

- サーバー管理者ユーザーが Creator ライセンスを使用し、そのライセンスの有効期限が切れる場合 (また、置換ライセンスを利用できない)、Explorer ライセンスが利用可能であれば Explorer ライセンスに再割り当てされます。このライセンス再割り当ては、最近ログインした

順番に行われます。サーバー管理者は、現在 Explorer ライセンスを使用している可能性がある他のユーザーを他に移動させます。利用可能な Creator または Explorer ライセンスがない場合、サーバー管理者はライセンスなしとなります。

- 非サーバー管理者ユーザーが Creator ライセンスを使用し、そのライセンスの有効期限が切れる場合 (また、置換ライセンスを利用できない)、ライセンスなしとなります。これらのユーザーがライセンスなしにならないようにするには、ライセンスの有効期限前にサイトロールを変更します。これはサイト管理者 Creator サイトロールのユーザーでは特に重要です。これらのユーザーは、サイト管理者権限を失うことがないよう、Creator ライセンスの有効期限が切れる前にサイト管理者 Explorer サイトロールに移行する必要があります。
- 非サーバー管理者ユーザーが Explorer または Viewer ライセンスを使用し、そのライセンスの有効期限が切れる場合 (また、置換ライセンスを利用できない)、より高いライセンス タイプが利用可能であれば、そのライセンス タイプにアップグレードされます。特に、ライセンスの有効期限が切れる場合には以下が起こります。
  - Explorer ライセンスを使用するユーザーは、Creator ライセンスが利用可能であればこれに移行します (サイトロールの変更なし)。
  - Viewer ライセンスを使用するユーザーは、利用可能であれば Explorer ライセンスに移行します。利用可能な Explorer ライセンスがない場合、Creator ライセンスが利用可能であればこれに移行します (サイトロールの変更なし)。.
  - 高いライセンス タイプで利用可能なライセンスがない場合、これらのユーザーはライセンスなしに移行します。

上記の通りに、ユーザーは最近ログインした順番で新しいライセンスに再割り当てされ、低いライセンス タイプが最初に再割り当てされます (最初に Viewer、次に Explorer、Creator の順)。。

たとえば、Viewer ライセンスを持つユーザー 2 人、Creator ライセンスを持つユーザー 1 人、Creator ライセンスを持つサーバー管理者 2 人でどのライセンスも有効期限が切れているとします。これらのユーザーで有効期限が切れていない 4 つの Explorer ライセンスが利用可能です。この状況で、次のことが以下に示す順序で起こります。

1. 最近ログインした Viewer ライセンスを持つユーザーが Explorer ライセンスに再割り当てされます。
2. 2 番目の Viewer ライセンスを持つユーザーが Explorer ライセンスに再割り当てされます。
3. 最近ログインした Creator ライセンスを持つサーバー管理者が Explorer ライセンスに再割り当てされ、2 番目の Creator ライセンスを持つサーバー管理者が Explorer ライセンスに再割り当てされます。
4. Creator ライセンスを持つユーザーはライセンスなしになります。

Creator ライセンスを使用する場合、サーバー管理者サイトロールは変更されない

サーバー管理者は、Tableau Server で Creator ライセンスを使用可能な場合には Creator 権限を得ますが、サイトロール名に変更はありません。Tableau Server および Tableau Cloud の他のすべてのユーザーは、サイトロール名に Creator を含むサイトロールに割り当てられる場合にのみ Creator ライセンスを得ます。

ライセンスはすぐには利用できない

Tableau Server にロールベース ライセンスを追加する場合、それらのライセンスは Tableau Server の再起動時にすべてのユーザーで利用可能になります。

Viewer ライセンスを持つユーザーは、Tableau Desktop から Tableau Server または Tableau Cloud のワークブックを開くことができない

Viewer ライセンスを持ち、それとは別に Tableau Desktop ライセンスも持つユーザーは、Tableau Desktop を使用して Tableau Server または Tableau Cloud のワークブックを開くことはできません。このように Tableau Desktop を使用してワークブックを開くには、そのユーザーに Tableau Server または Tableau Cloud で Explorer または Creator ライセンスが必要です。

コアベース ライセンスからロールベース ライセンスへの移行

Tableau Server をコアベースのライセンス メトリクス (Tableau Server をインストールしたプロセッサー コア数をカウント) からロールベースのライセンス メトリクス (名前の付いたユーザー数をカウント) に移行することができます。ライセンス メトリクスの詳細については、ライセンス発行の概要を参照してください。

ロールベース ライセンスへの移行準備

コアベースのライセンスを使用すると、表示のみのゲスト アカウントを含め、無制限のユーザー数を許可できます。どのユーザーも Tableau Server に追加されるとサイトロールが設定され、これらのユーザーとサイトロールはライセンス設定が変更されても保持されます。ロールベース ライセンスではユーザー数が制限されるため、ロールベースの新しいライセンスが、現在ゲスト アカウントを使用しているユーザーを含め、現在 Tableau Server に接続しているユーザー数に対応することを確認する必要があります。

重要: ロールベースの新しいライセンスが全ユーザー数に対応しない場合、一部のユーザーは**「ライセンスなし」**サイトロールに移行しますが、これが行われるとCPU にかなりの負担がかかります。多数のユーザーがいる場合、すべてのユーザーが処理されるまで、アプリケーション サーバープロセスが使用できなくなる可能性があります。この操作は完了するまでに数時間かかる可能性があるため、それを考慮して計画を立ててください。

移行完了後にすべてのユーザーが Tableau Server にアクセスできるようにするには、移行する前に、ユーザー ライセンスと対応するプロダクト キーがあることを確認します。

- Tableau カスタマー ポータルにサインインして、ライセンスを確認し、対応するプロダクト キーをコピーします。
- サイトロールの詳細については、ユーザーのサイトロールの設定を参照してください。
- お使いの Tableau Server インストールでのユーザー数をカウントするには、ユーザー リストをエクスポートし、Microsoft Excel などのツールでカウントしてください。ユーザー リストのエクスポート方法については、「ユーザー リストをエクスポートする」を参照してください。

ロールベース ライセンスへの移行

ロールベース ライセンスに移行するには、Tableau Server を停止し、コアベースのプロダクト キーのライセンス認証を解除し、ロールベースのプロダクト キーをライセンス認証してから、Tableau Server を起動します。このプロセスは Tableau Server を再起動させ、Tableau Server のユーザーにとってはダウンタイムが発生するため、使用率が低い期間にライセンス設定を移行するようにしてください。

1. Tableau Server を停止します。

   `tsm stop` コマンドを使用します。

2. コア ベースのプロダクト キーのライセンス認証を解除します。

   コア ベースのプロダクト キーで `tsm licenses deactivate` コマンドを使用します。

3. ロールベースのプロダクト キーをライセンス認証します。

   ロールベースのプロダクト キーで `tsm licenses activate` コマンドを使用します。

4. Tableau Server を起動します。

   `tsm start` コマンドを使用します。

コアベース ライセンスが設定されているサーバーでロールベース ライセンスを使用する

Tableau Server の 2018.1 リリースでは、既存のコアベース ライセンスが設定されている Tableau Server インストールに Creator ロールベース ライセンスを追加できます。

> **注:** ロールベースのライセンスを認証せずに Tableau Server インストールを 2018.1 にアップグレードする場合、Tableau Server はこれまでと同様に動作し、従来の **Viewer (ビューアー)** サイトロールの名前が**読み取り専用**に変更になったことを除き、Tableau Server の UI やパーミッションに変更はありません。バージョン 2018.2 では、**読み取り専用** サイトロールは非推奨となり、もう一度 **Viewer** サイトロールになりました。

Creator ライセンスが Tableau Server に導入されると、すべてのサーバー管理者ユーザーに **Creator** ライセンス ロールが必要になります。これは追加のライセンスが必要になる場合があります。管理者は Tableau Server で `tsm licenses activate` コマンドを使い、**Creator** の追加ライセンスを認証し、ライセンス ユーザーの容量を増やすことができます。これらのロールベース ライセンスを認証すると、ユーザーベースの Creator ライセンスにより付与される権限と、コアベース ライセンスにより付与される権限を組み合わせることができます。異なる種類のロールベース ライセンスの詳細については、「ロールベース ライセンス」を参照してください。

> **注:** コアベース ライセンスで Tableau Server へのアクセス権を持つ無制限な数のユーザーは、ロールベース ライセンスで **Explorer** ライセンスを持つユーザーと同等の権限を持っています。**Creator** 機能は、定義されたユーザー ライセンス モデルに制限されます。

たとえば、Tableau Server インストールに、ゲスト アクセスを含む 16 コア ライセンスがあり、10 の Creator ライセンスが含まれる更新可能なサブスクリプション ライセンス (USL) プロダクト キーを追加した場合、そのサーバーは以下の権限を持つことになります。

- Tableau Server を実行するハードウェアではプロセッサー コア数に 16 の制限
- ゲスト アクセス
- 無制限の Explorer/Viewer ライセンス (コア ライセンスに付随する無制限のユーザー ライセ

ンスによる)
- 10 の Creator ライセンス

コアベース ライセンスからの移行完了の例

上記の例を次のように拡張します: その後、コアベースのライセンス認証が解除された場合、以下の権限を利用できます。

- サーバー ハードウェアには制限なし
- ゲスト アクセスなし
- 10 の Creator ライセンス

その後、50 の Explorer ライセンスと200 の Viewer (ビューアー) ライセンスを追加した場合、コアベース ライセンス認証を解除した後、以下の権限を利用できます。

- サーバー ハードウェアには制限なし
- ゲスト アクセスなし
- 10 の Creator ライセンス
- 50 の Explorer ライセンス
- 200 の Viewer (ビューアー) ライセンス

クイック スタート: Tableau Server でログインベースのライセンス管理を使用する

以下の手順を使用すると、ログインベースのライセンス管理 ですぐに起動して実行できます。

ステップ 1: Tableau Server のインストール

ログインベースのライセンス管理 を使用して Tableau をライセンス認証するには、Tableau Server バージョン 2021.1 以降をインストールし、ログインベースのライセンス管理 が有効になっているプロダクト キーを使用してライセンス認証する必要があります。ログインベースのライセンス管理 を使用すると、Tableau の Creator ユーザーが Tableau Desktop または Tableau Prep Builder にサインインしてライセンス認証することができます。

> **注:** Tableau カスタマー ポータルで Tableau Server のプロダクト キーが ログインベースのライセンス管理 に対して有効になっていない場合は、Tableau アカウントの担当者に連絡してください。

次のトピックでは、Tableau Server のインストールと、ログインベースのライセンス管理 を使用した Tableau Desktop とTableau Prep Builder のライセンス認証に関する追加情報を確認できます。

- Tableau Server のインストールと構成

ログインベースのライセンス管理 の構成オプションを変更するには、または Tableau Server で ログインベースのライセンス管理 を無効にするには、「ログインベースのライセンス管理設定の変更」を参照してください。

ステップ 2: Tableau Server への許可されたユーザーの追加

Tableau Server をライセンス認証した後、許可されたユーザーを追加できます。ユーザーを追加するときは、そのユーザーのサイト ロール (Creator、Explorer、サイト管理者など) を選択する必要があります。Tableau Desktop または Tableau Prep Builder をライセンス認証するユーザーは、Creator のサイトロール (Creator またはサイト管理者 Creator) に割り当てられている必要があります。サイト ロールの詳細については、ユーザーのサイト ロールの設定を参照してください。

ステップ 3: Tableau Desktop または Tableau Prep Builder のライセンス認証

Creator ユーザーは、ログインベースのライセンス管理 を使用してライセンス認証する前に、Tableau Desktop または Tableau Prep Builder、あるいはその両方をコンピューターにダウンロードしてインストールする必要があります。詳細については、『Tableau Desktop および Tableau Prep の導入ガイド』の「ユーザー インターフェイスからの Tableau Desktop または Tableau Prep Builder のインストール」を参照してください。

1. Tableau Desktop または Tableau Prep Builder を起動します。

   Tableau Desktop または Tableau Prep Builder を初めて起動する場合は、14 日間の無料トライアルを開始するか、Tableau をライセンス認証するかを選択できます。

2. 無料トライアルをスキップするには、**[Tableau のライセンス認証]** をクリックします。

3. **[Tableau のライセンス認証]** 画面で、**[サーバーにサインインしてライセンス認証]** をクリックし、ログインベースのライセンス管理 を使用して Tableau にサインインしてライセンス認証します。

4. プロンプトが表示されたら、Tableau Server の URL を指定し、自分の認証資格情報を使用してサインインします。

5. Tableau Server によって認証されると、Tableau Desktop または Tableau Prep Builder、あるいはその両方がライセンス認証され、使用できるようになります。

ログインベースのライセンス管理

ログインベースのライセンス管理 を使用すると、Tableau Server とTableau Cloud で Creator ロールを持つユーザーのライセンス発行を管理しやすくなります。Explorer または Viewer のロールを持つユーザーは、この機能を使用できません。Tableau Server または Tableau Cloud でロール ベースのサブスクリプションを使用している場合は、ログインベースのライセンス管理 を使用してライセンス管理を簡略化し、個別の Tableau Desktop および Tableau Prep Builder プロダクト キーを削減できます。管理する必要があるのは、オンプレミスの Tableau Server 用の 1 つまたは複数のプロダクト キーだけです。Tableau Cloud の場合は、プロダクト キーを管理する必要は全くありません。

個々の Tableau Desktop または Tableau Prep Builder を承認するために必要な Tableau Server または Tableau Cloud サイトは 1 つだけです。ログインベースのライセンス管理 は Tableau Server 展開の本番インスタンスごとに有効化されます。Tableau Server、Tableau Desktop、Tableau Prep Builder を使用してライセンス認証と中央型ライセンス管理を行うユーザーには、Creator のロールを割り当てることができます。また、ログインベースのライセンス管理 によってライセンス使用率をより良く把握することができます。

Tableau Server とTableau Cloud の管理ビューを使用して、Creator ロールの割り当て数と使用率を 1 か所で表示できます。管理ビューでは、Creator ライセンスを使用しているユーザーに関する情報が提供され、Tableau Desktop とTableau Prep Builder の最新のライセンス リースとバージョンが表示されるため、Tableau の展開を監視するのに役立ちます。

ログインベースのライセンス管理 は以下の Tableau 製品でサポートされています。

- Tableau Cloud
- Tableau Server
- Tableau Desktop
- Tableau Prep Builder

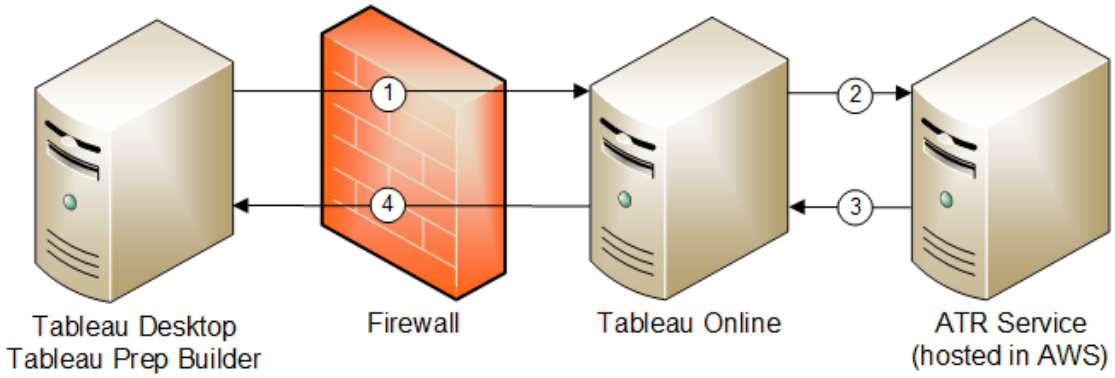
Tableau Server がログインベースのライセンス管理で有効になっていない場合は、Tableau 営業担当者に連絡して、ログインベースのライセンス管理対応の特別なプロダクトキーを入手してください。

ログインベースのライセンス管理の動作方法

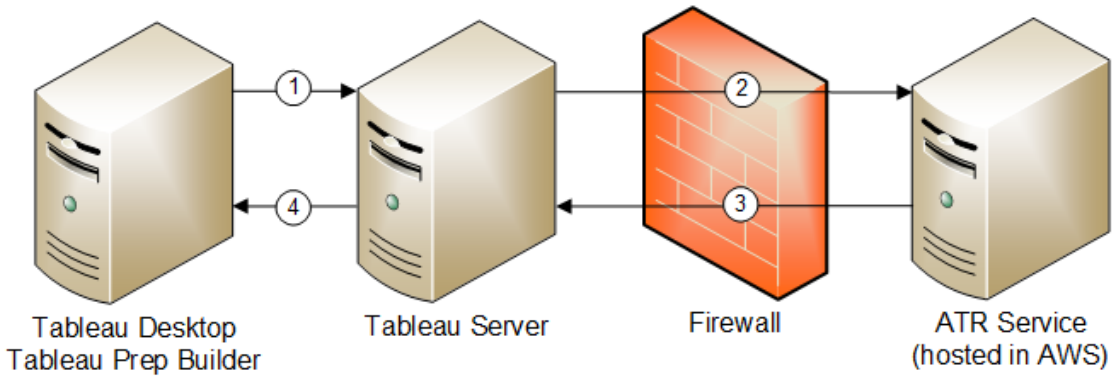
ログインベースのライセンス管理が使用されている場合、Tableau Desktop または Tableau Prep Builder は定期的に Tableau Server または、Tableau Cloud と連絡を取り、ユーザーが Creator であることを確認し、実行権限を取得します。Tableau Server または Tableau Cloud は Tableau でホストされる実行承認 (ATR) サービスに連絡して Tableau Desktop クライアントがアクティブ化を許可されていることを確認します。AWS でホストされている ATR サービスはライセンスと承認ウィンドウの長さを検証して管理します。下の図は、クライアント、Tableau Server または Tableau Cloud、およびライセンス発行サービスとの間の通信プロセスを示しています。

Tableau は共通ポート (80 および 443) を使用してインターネット要求を行います。ほとんどの場合、Tableau Desktop または Tableau Prep Builder がインストールされている PC、および、実行中のネットワークは、既に必要なアクセスを許可するよう構成されています。Tableau で使用されるポートの詳細については、Tableau サービス マネージャーのポートを参照してください。

ログインベースのライセンス管理 とTableau Cloud



ログインベースのライセンス管理 とTableau Server



1. Tableau Desktop または Tableau Prep Builder をインストールし、**[認証資格情報を使用してライセンス認証]** を選択します。Tableau Cloud または Tableau Server サイトにサインインします。

2. Tableau Cloud または Tableau Server は、ユーザーが Creator であることを確認します。そうでない場合は、エラーが発生します。そうである場合、Tableau Cloud または Tableau Server が ATR サービスと通信します。

3. ATR サービスは ATR リースを Tableau Cloud または Tableau Server に返します。

4. Tableau Cloud または Tableau Server は ATR リースを Tableau Desktop または Tableau Prep Builder に提供してアクティブ化を終了します。

ログインベースのライセンス管理を使用する

ログインベースのライセンス管理を使用するには、Tableau Cloud を使用するか、Tableau Server をインストールし、ログインベースのライセンス管理 に対応したプロダクトキーを使用して有効化する必要があります。エンドユーザーには、Tableau Cloud または Tableau Server のいずれかで Creator ロールを割り当てる必要があります。また、そのユーザーは Tableau Desktop をインストールする必要があります。ユーザーは最大 2 台のマシンで Tableau Desktop を有効にすることができますが、Creator のシートを割り当てる必要があるのは 1 台のみです。

> **注**: Tableau Server バージョン 2023.1 以降、LBLM はオフライン展開でサポートされています。LBLM を有効にして Tableau Server を展開するには、実行許可 (ATR) サービスを使用するように Tableau Server を構成する必要があります。詳細については、オフラインでの Tableau Server の有効化を参照してください。

ステップ 1: Tableau Server をインストールする

Tableau Server が既にインストールされている場合は、(オプション) ステップ 2: ログインベースのライセンス管理の設定を変更する。そうでない場合は、Tableau Server をインストールする必要があります。

1. 展開の章を参照して Tableau Server をインストールします。
2. ライセンス認証時は、ロールベースのサブスクリプションとログインベースのライセンス管理をサポートする Tableau Server プロダクトキーを入力する必要があります。
3. ユーザーを Tableau Server に追加し、3 種類の Creator ロール (サーバー管理者、サイト管理者 Creator、または Creator) のいずれかに設定します。これにより、Creator ライセンスが Tableau Server 上の当該のユーザーに割り当てられます。詳細については、「ユーザーのサイトロールの設定」を参照してください。

重要: 接続されているクライアントで ログインベースのライセンス管理 を使用している場合は、自動クライアント認証の無効化 を行わないでください。また、ログインベースのライセンス管理 を有効にした後は、接続されているデスクトップのユーザー インターフェイスで Tableau Server からサインアウト

しないでください。これを行うと、ログインベースのライセンス管理がライセンスのアクティブ化を自動的に更新したり、Tableau Server または Tableau Cloud の LBLM 使用状況レポートに現在のデータを提供したりすることができなくなります。デスクトップからの接続の詳細については、「Tableau Desktop が Tableau Server または Online に接続された状態を自動的に保持する」を参照してください。

(オプション) ステップ 2: ログインベースのライセンス管理の設定を変更する

ログインベースのライセンス管理は、Tableau Cloud、Tableau Server、および Tableau Desktop では既定で有効になっています。ただし、ログインベースのライセンス管理の設定の一部を変更することができます。

Tableau Server でログインベースのライセンス管理を無効にする

ログインベースのライセンス管理は、Tableau Desktop および Tableau Prep Builder をアクティブ化するために推奨されている方法ですが、組織内で、ログインベースのライセンス管理を使用する必要のないテスト用の Tableau Server がインストールされている場合があります。Tableau Server で、Tableau サービスマネージャー (TSM) コマンドラインユーティリティを使用して、ログインベースのライセンス管理をオンまたはオフにします。

ログインベースのライセンス管理をオフにするには、コマンドプロンプトで次のように入力します。

```
tsm configuration set -k licensing.login_based_license_
management.enabled -v false

tsm pending-changes apply
```

Tableau Desktop や Tableau Prep Builder のインストール時にログインベースのライセンス管理設定を変更します

コマンドラインでログインベースのライセンス管理設定を変更するには、管理者としてコンピュータのコマンドラインからインストーラー .exe ファイルを実行します。msi ファイルを抽出する必要がある

場合は、「Windows (MSI) インストーラーの抽出と実行」の手順を実行します。

既定の 14 日/1,209,600 秒以外の期間を使用するには、ATRREQUESTEDDURATIONSECONDS スイッチを含めます。例:

```
tableauDesktop-64bit-2024-2-0.exe /quiet /norestart ACCEPTEULA=1
ATRREQUESTEDDURATIONSECONDS=43200
```

.exe ファイルが格納されているディレクトリからコマンドを実行するか、コンピューターの .exe ファイルへの完全パスを指定する必要があります。ネットワーク上の共有ディレクトリからセットアッププログラムを実行しないでください。代わりに、インストール先の PC 上のディレクトリに .exe ファイルをダウンロードします。

次の例は、ログインベースのライセンス管理を無効にする Windows インストーラー コマンドを示しています。

```
tableauDesktop-64bit-2024-2-0.exe /quiet /norestart ACCEPTEULA=1
LBLM=disabled
```

または

```
tableauPrepBuilder-64bit-2024-2-0.exe /quiet /norestart ACCEPTEULA=1
LBLM=disabled
```

ログインベースのライセンス管理 を使用するときにユーザーがアクティブ化に使用する Tableau Server の既定の URL を設定するには、ACTIVATIONSERVER または WorkGroupServer オプションを追加します。

exe を更新するには:

```
tableauDesktop-64bit-2021-4-0.exe /quiet /norestart ACCEPTEULA=1
ACTIVATIONSERVER=http://<tableau_server_url>
```

レジストリを更新するには:

```
reg.exe add HKEY_LOCAL_MACHINE\SOFTWARE\Tableau\Tableau
2024.2\Settings /f /v WorkGroupServer /d https://<tableau_server_
url>
```

注: ACTIVATIONSERVER オプションは、初回のライセンス認証のみを目的としています。以前にこのバージョンの Tableau にサインインしたことがある場合は、WorkGroupServer (Windows) または WorkgroupServer (macOS) オプションを使用します。たとえば、Windows で Tableau Desktop を使用していて、以前に正常にサインインしたことがある場合は、WorkGroupServer オプションを使用してアクティベーション サーバーを指定します。macOS の Tableau Desktop では、WorkgroupServer オプションを使用します。macOS では、このオプションは大文字と小文字を区別し、小文字の "g " が使用されます。

レジストリの編集によって Tableau Desktop でログインベースのライセンス管理を変更します

Tableau Desktop が既にインストールされている場合は 、Windows レジストリを編集して ログインベースのライセンス管理 設定を変更できます。

ログインベースのライセンス管理 をオフにするには :

```
reg.exe add HKEY_LOCAL_MACHINE\SOFTWARE\Tableau\ATR /f /v LBLM /d
disabled
```

ログインベースのライセンス管理唯一のログインオプションを作成するには:

```
reg.exe add HKEY_LOCAL_MACHINE\SOFTWARE\Tableau\ATR /f /v LBLM /d
required
```

または、レジストリディレクトリを編集して、ログインベースのライセンス管理 を有効、無効、または必須オプションに設定できます。

1. Tableau Desktop を実行しているコンピューターの管理者として、変更を加える前にレジストリファイルのバックアップを作成してください。
2. レジストリを編集し、HKEY_LOCAL_MACHINE\SOFTWARE\Tableauで ATR という名前のハイブを見つけ、必要な設定を反映するように LBLM 値を変更します。
   a. 名前: LBLM。
   b. データ:enabled、disabled、または required。
3. Tableau を再起動して変更を有効にします。

ログインベースのライセンス管理 設定を Tableau Desktop macOS で変更する

macOS 上で ログインベースのライセンス管理 設定を変更するには、ターミナル ウィンドウで次のコマンドを実行して preferences ファイルを更新した後、Tableau Desktop をインストールまたは再起動します。

ログインベースのライセンス管理 をオフにするには :

```
sudo defaults write /Library/Preferences/com.tableau.ATR LBLM
"disabled"
```

ログインベースのライセンス管理唯一のログインオプションを作成するには:

```
sudo defaults write /Library/Preferences/com.tableau.ATR LBLM
"required"
```

macOS で ログインベースのライセンス管理 を使用するときに、ユーザーがアクティブ化に使用する Tableau Server の既定の URL を設定します。

初回 アクティブ化:

```
sudo defaults write /Library/Preferences/com.tableau.ATR LBLM
"required"
```

2 回目以降のアクティブ化:

```
sudo defaults write /Library/Preferences/com.tableau.Tableau-
<version> Settings.WorkgroupServer "https://<tableau_server_url>"
```

注: macOS の Tableau Desktop では、`WorkgroupServer` オプションを使用します。macOS では、このオプションは大文字と小文字を区別し、小文字の "g" が使用されます。

ログインベースのライセンス管理設定

次の設定を使用してログインベースのライセンス管理を変更し、ATR 期間を設定し、アクティブ化サーバーの URL を設定します。

設定	値	説明
`LBLM`	enabled、disabled、または required	`enabled` (既定値) に設定すると、ライセンス発行画面に 2 種類のライセンス認証オプション (プロダクトキー、認証資格情報) が表示されます。 `disabled` に設定すると、ログインベースのライセンス管理 がライセンス画面に表示されません。 `required` に設定すると、ログインベースのライセンス管理 が Tableau Desktop をライセンス認証する唯一の方法になります (ライセンス発行画面が表示されると

		き、認証資格情報によるライセンス認証のオプションだけが表示されます)。 **注:** LBLM が enabled または required に設定されている場合は、レポート作成も有効になります。
ATRREQUESTEDDURATIONSECONDS	1209600	実行の承認 (ATR) の期間 (秒) を設定します。これは、Tableau Desktop とTableau Prep Builder のインスタンスが実行を承認される期間です。既定値は 1209600 秒 (14 日) です。値で区切り文字としてコンマを使用しないでください。
ACTIVATIONSERVER	http://<tableau_server_url>	初回アクティブ化の場合、ユーザーがアクティブ化に使用する Tableau Server の既定の URL を設定します。
WorkGroupServer (Windows) WorkgroupServer (macOS)	http://<tableau_server_url>	Windows レジストリまたは macOS plist を更新する場合は、ユーザーが Tableau Server のアクティブ化に使用する既定の URL を設定します。 **注:** macOS の Tableau Desktop では、WorkgroupServer オプション

		を使用します。macOS では、このオプションは大文字と小文字を区別し、小文字の "g " が使用されます。
REPORTINGFREQUENCYSECONDS	3600	ログインベースのライセンス管理レポートを Tableau Server または Tableau Cloud に送信する頻度の既定値 (秒単位) を設定します。設定できる最小値は 60 秒で、デフォルトは 3,600 秒 (1 時間) です。この設定を変更して、Tableau Server の負荷を軽減したり、ネットワーク トラフィックを削減したりします。実行承認 (ATR) リース期間の設定値に関わらず、REPORTINGFREQUENCYSECONDS で、Tableau Desktop または Tableau Prep Builder クライアントがログインベースのライセンス管理ライセンス認証を Tableau Server または Tableau Cloud に報告する時間間隔を設定します。

仮想展開の追加構成

Tableau Desktop とTableau Prep Builder は、Tableau CloudまたはTableau Serverに定期的に連絡し、Tableau がライセンスベースで実行を許可されていることを検証します。Tableau CloudまたはTableau Serverは、ATR サービスに連絡し、ライセンスと承認ウィンドウの長さを検証します。

Tableau Desktop または Tableau Prep Builder の仮想(永続的ではない)展開を設定する際は、使用超過のアクティブ化エラーメッセージを回避するために、期間は 4 時間または 8 時間などの低い値のいずれかに設定する必要があります。仮想マシン(VM)が返された後、ATR サービスは、アクティブ化の監視を処理します。

Tableau Desktop をエンドユーザーにパブリッシュする前に、ソース画像で次のフラグを使用する必要があります。各エンドユーザーは、新しい VM が配信される都度 Tableau Server または Tableau Cloud にログインすることでソフトウェアをアクティブ化します。エンドユーザーが Tableau Server または Tableau Cloud で Tableau Creator である場合は、プロダクトキーを入力する必要はありません。

VM上の Tableau Desktop または Tableau Prep Builder でログインベースのライセンス管理を使用している場合、Tableau Desktop または Tableau Prep Builder で新しい VM を起動するたびにライセンス情報が変更されたというエラー メッセージが表示されることがあります。このエラーによって再起動が強制的に実行され、Tableau Desktop の再登録を求めるメッセージが表示されます。このエラーは、ATR サービスがライセンス キャッシュと一致しない新しいトークンを送信するために発生します。

Microsoft Windows

Microsoft Windows でこの問題が発生しないようにするには、Windows インストーラーで SYNCHRONOUSLICENSECHECK および SILENTLYREGISTERUSERオプションを使用します。例:

```
tableau-setup-std-tableau-2024 SYNCHRONOUSLICENSECHECK="true"
SILENTLYREGISTERUSER="true" ATRREQUESTEDDURATIONSECONDS=14400
```

または

```
tableauDesktop-64bit-2024-2-0.exe /quiet /norestart ACCEPTEULA=1
ATRREQUESTEDDURATIONSECONDS=14400 ACTIVATIONSERVER=http://<tableau_
server_url> SYNCHRONOUSLICENSECHECK="true"
SILENTLYREGISTERUSER="true"
```

Tableau Prep Builder とTableau Desktop を 1 つの仮想デスクトップに配信する場合、ATRREQUESTEDDURATIONSECONDS を設定する必要があるのは Tableau Desktop のインストー

ル中だけです。ただし、Tableau Prep Builder をスタンドアロンとしてインストールする予定の場合は、ATRREQUESTEDDURATIONSECONDS を Tableau Prep Builder のインストール中に設定する必要があります。

または、次のレジストリキーをマスター画像で編集できます。

```
Reg key path: HKLM\SOFTWARE\Tableau\<Tableau version>\Settings\
```

```
Reg key (String value, need to set to true to make that feature enabled)
```

```
SynchronousLicenseCheck
```

```
SilentlyRegisterUser
```

MacOS

macOS でエラーが発生しないようにするには、次のコマンドを実行して LicenseCache.Desktop フラグを 'false' に設定します。

```
sudo defaults write ~/Library/Preferences/com.tableau.Tableau-<version>.plist LicenseCache.Desktop false
```

(オプション) ステップ 3: 実行の承認 (ATR) の期間を変更する

実行承認 (ATR) 期間のログインベースのライセンス管理デフォルト設定はほとんどの環境に適していますが、必要に応じてこれらのデフォルト設定を変更できます。ログインベースのライセンス管理、実行の承認 (ATR) の期間として既定値の 1,209,600 秒 (14 日) が使用されます。これは、Tableau Desktop と Tableau Prep Builder のインスタンスが実行を承認される期間です。つまり、最初の認証からライセンス認証の有効期限が切れるまでに、Tableau をネットワーク接続なしで 14 日間使用できます。

ログインベースのライセンス管理の既定の期間の値は、エンドユーザーに非永続的な VM 配信ソリューションを配信するのには適切ではありません。ATR 期間は、VM の使用に応じて 4 時間または 12 時間に下げる必要があります。新しい VM がエンドユーザーに配信されると、トークンを実

行するための新しい権限が作成されます。VM が返されると、このトークンも返され、要求を実行する新しい VM 承認で使用できます。

注: ログインベースのライセンス管理では、ATR 期間を決定するときに以下の階層が使用されます。

1. **ATR サービス –** すべてのユーザー/インストールに適用される最小 (4 時間/14,400 秒) と最大 (90 日/7,776,000 秒) の ATR 期間を確立します。Tableau Server や Tableau Desktop で値が指定されていない場合は、既定の ATR 期間 (14 日/1,209,600 秒) が指定されます。

2. **Tableau Server -** オプションで、すべての Tableau Desktop インストールに対して、最大または既定の ATR 期間 (`licensing.login_based_license_management.max_requested_duration_seconds` または `licensing.login_based_license_management.default_requested_duration_seconds`) を指定できます。これらの ATR 期間設定を使用すると、すべての Tableau Desktop クライアントに対して、既定の ATR 期間と最大 ATR 期間をグローバルに設定できます。これにより、各 Tableau Desktop クライアントに個別にサインインして ATR 期間を設定する必要がなくなります。ATR サービスの最大設定は、最小 4 時間 (14 400 秒) から最大 180 日 (15552000 秒) の間で指定できます。

3. **Tableau Desktop –** オプションで、インストール先の PC 用の ATR 期間 (`ATRREQUESTEDDURATIONSECONDS`) を指定できます。必要に応じて、既定の ATR 期間 (14日/1,209,600 秒) を ATR サービスの最小 (4 時間/14,400 秒) および最大 (90 日/7,776,000秒) の範囲の設定に変更できます。このローカル ATR 期間は、ATR サービスや Tableau Server によって設定された期間より優先されます。ただし、この既定の ATR 期間は、Tableau Server で設定されている最大 ATR 期間より大きくすることができません。

Tableau Server を使用して、Tableau Desktop または Tableau Prep Builderの ATR 期間を変更する

Tableau Server では、Tableau サービス マネージャー (TSM) のコマンド ライン ユーティリティを使用して ATR 期間を設定します。

ATR 期間を設定するには、コマンドプロンプトで次のように入力します。

```
tsm configuration set -k licensing.login_based_license_
management.default_requested_duration_seconds -v <value in seconds>

tsm pending-changes apply
```

ATR 最大期間を設定するには、コマンドプロンプトで次のように入力します。

```
tsm configuration set -k licensing.login_based_license_
management.max_requested_duration_seconds -v <value in seconds>

tsm pending-changes apply
```

レジストリを編集して Tableau Desktop の ATR 期間を変更する

既定の 14 日/1,209,600 秒以外の期間を使用するには、`ATRRequestedDurationSeconds` レジストリ設定を更新します。例:

1. Tableau Desktop を実行しているコンピューターの管理者として、変更を加える前にレジストリファイルのバックアップを作成してください。
2. レジストリを編集し、`HKEY_LOCAL_MACHINE\SOFTWARE\Tableau\ATR` で ATRRequestedDurationSeconds を次のように更新します (0 は既定の設定を使用します)。
    a. 名前:「`ATRRequestedDurationSeconds`」という名前の文字列値を検索します。
    b. データ: 期間が持続する秒数を追加します。たとえば、43200 を追加して 12 時間の期間を設定します。
3. Tableau を再起動して変更を有効にします。

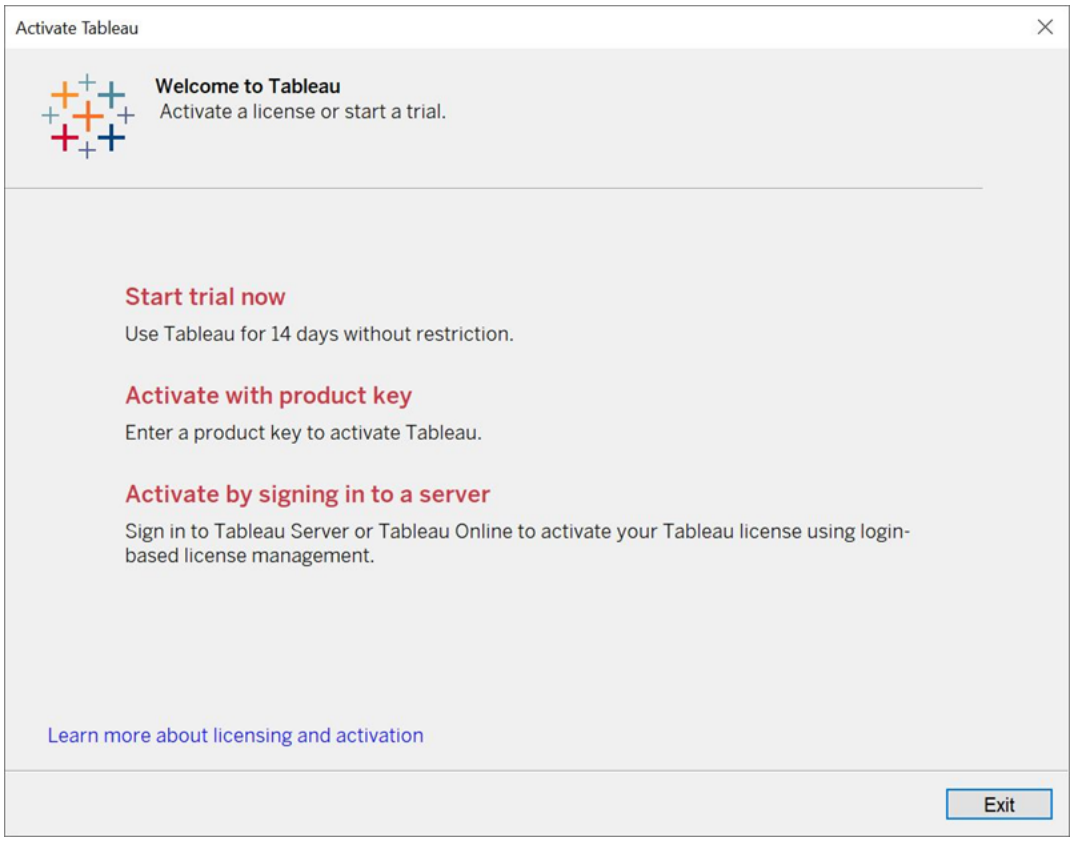
macOS で Tableau Desktop の ATR 期間を変更する

既定の 14 日/1,209,600 秒以外の期間を使用するには、ATRRequestedDurationSeconds 環境設定を含めます。例:

```
sudo defaults write /Library/Preferences/com.tableau.ATR
ATRRequestedDurationSeconds -string "43200"
```

ステップ 4: Tableau Desktop をライセンス認証する

1. Tableau Desktop セットアップを実行します。
2. [Tableau のライセンス認証] 画面には、**[サーバーへのサインインによってライセンス認証する]** オプションが含まれます。



3. **[サーバーへのサインインによってライセンス認証する]** をクリックし、次のいずれかを実行します。

    - Tableau Cloud を使用する場合は、Tableau Cloud リンクをクリックします。
    - Tableau Server を使用する場合は、Tableau Server URL を入力し、**[接続]** をクリックします。

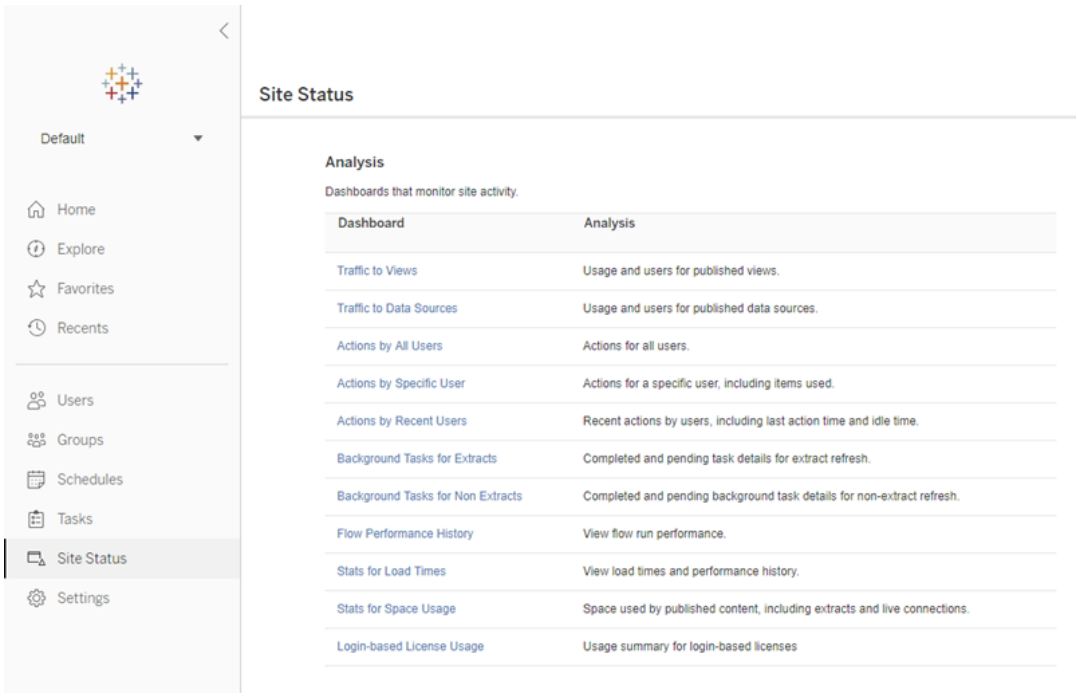
    プロンプトが表示されたら、Creator のロール サブスクリプションを持つユーザーの有効な認証資格情報を入力し、**[OK]** をクリックします。

ログインベースのライセンス使用状況の表示

Tableau Cloud または Tableau Server に関するログインベースのライセンス使用状況を表示できます。レポートには、ユーザー、ホスト、ユーザー ロール、製品、バージョン、ライセンス認証、使用中の Creator シート、使用中でない Creator シート、Creator シートが最後に使用された日時が表示されます。過去 30 日 ~ 183 日間のデータを表示できます。

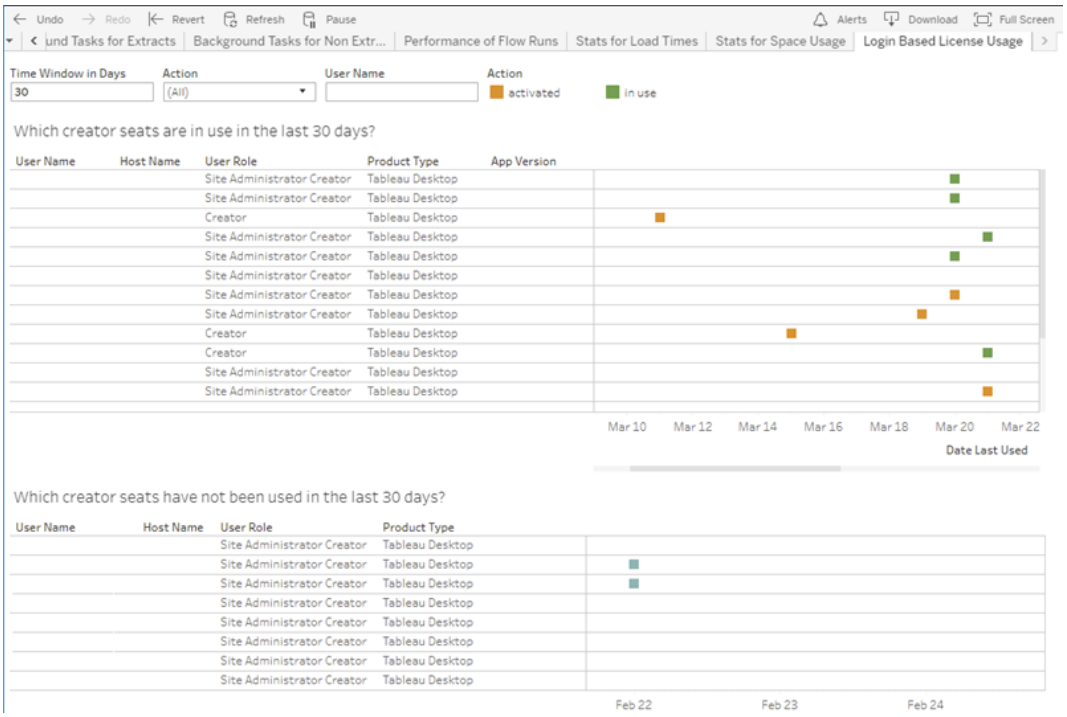
ログインベースのライセンス管理 ライセンス使用状況管理ビューを表示するには、次の手順を実行します。

1. Tableau Cloud または Tableau Server のナビゲーション ペインで、**[サイトのステータス]** をクリックします。



2. [サイトのステータス] ページで、**[ログインベースのライセンス使用率]** をクリックします。

3. オプション。レポート画面では、シートが最後に使用された日時を示す時間枠の変更、アクションのフィルター、ユーザー名のフィルター、列による並べ替えを行うことができます。



トラブルシューティング

ログインベースのライセンス管理 の使用中に、次のいずれかのエラー メッセージが表示される場合があります。以下の情報を使用して問題を解決してください。

ログインベースのライセンス管理 が Tableau Server で有効になっていない

ログインベースのライセンス管理 が Tableau Desktop ではアクティブ化オプションとして利用できるが Tableau Server では有効になっていない場合、または Tableau Server が ログインベースのライセンス管理 に対応したライセンスを持っていない場合は、次のエラー メッセージが表示されることがあります。

接続しているサーバー上で ログインベースのライセンス管理 が有効になっていません。別のサーバーを選択して接続するか、プロダクト キーを使用してライセンス認証してください。または、すぐに作業を始めるには試用版を起動してください。

また、管理者に連絡して、ログインベースのライセンス管理で有効になっている Tableau Server プロダクト キーを使用していることを確認します。適切なプロダクトキーをアクティブ化したことを確認

するには、TSM Web UI で **[構成]**、**[ライセンス発行]** の順にクリックします。ライセンス認証に関するガイダンスについては、tsm licenses activateを参照してください。以前に他のサーバー管理者よってオフに設定されていない限り、ログインベースのライセンス管理は既定で有効になっています。

ログインベースのライセンス管理 が Tableau Desktop で有効になっていない

ログインベースのライセンス管理 が Tableau Desktop で有効になっていない場合、認証資格情報を使用して Tableau をライセンス認証することができません。

インストール中または更新プログラムで ログインベースのライセンス管理 オプションがオフになっている場合は、「手順 2: ログインベースのライセンス管理を有効にする」を参照してください。

年間サブスクリプションを購入しても、プロダクトキーの有効期限は変更されません

Tableau Server の更新を購入すると、新しいサブスクリプション期間が Tableau Server に反映されます。新しい有効期限が Tableau Desktop の **[プロダクトキーの管理]** に表示されるまでに最大 24 時間かかる場合があります。

Creator ライセンスがない

Tableau Desktop からライセンス認証を試みると、次のエラー メッセージが表示される場合があります。

Creator ライセンスがありません。管理者に問い合わせてライセンスを取得してください。

このエラーは、Creator のロールが割り当てられていない場合に表示されます。Tableau Cloud または Tableau Server で複数のサイトに属している場合は、ログインベースのライセンス管理 を使用する際に Creator のロールを持つサイトにサインインする必要があります。そうしなければ、このエラーが発生します。

サーバーでアクティブ化したプロダクトキーに Creator ライセンスが含まれていることを確認するには、TSM Web UI を開き、**[構成]**、**[ライセンス発行]** の順にクリックします。

最大数の PC でライセンス認証した

Tableau Desktop からライセンス認証を試みると、次のエラー メッセージが表示される場合があります。

認証した Tableau のライセンス数がご契約アカウントで許可されている上限に達しました。他のコンピューターで Tableau ライセンス認証の有効期限が切れるまで、もう一度ライセンス認証することはできません。

このエラーは、複数の PC から同じ Creator ユーザーの認証資格情報を使用して Tableau をライセンス認証しようとした結果、ライセンス認証の最大数を超過したときに表示されます。新しい PC でライセンスを認証する前に、既存の PC のいずれかで実行の承認 (ATR) トークンの有効期限が切れるまで待つ必要があります。非永続的な仮想マシン (VM) を使用している場合は、このエラーが再度発生しないように ATR 期間を短くすることができます。

アクティブ化を最大にする ATR トークン期間を短縮するには

非永続的な仮想展開を使用する場合にこの最大使用エラーが発生した場合は、ATR の期間を 4 時間 (14400 秒) に短縮すると、今後のエラーを回避できます。または、個々のデスクトップで期間を変更する代わりに、すべてのユーザーに適用される既定の期間を Tableau Server で設定することもできます。

次のステップでは、ログインベースのライセンス管理 を使用して以前アクティブ化された、今後使用されなくなるコンピューターのリースを短縮して解放し、新しいコンピューターでアクティブ化できるようにします。

1. 使用されなくなる Tableau Desktop コンピューターで管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを使用し、Tableau バイナリ (\bin) ディレクトリに移動します。

   Windows: `cd Program Files\Tableau\Tableau <version>\bin`

   Mac: `cd /Applications/Tableau\ Desktop\ <version>.app/Contents/MacOS`

3. 次のコマンドを実行して、期間を 4 時間に設定します (秒単位で 14400)。

   Windows: `atrdiag.exe -setDuration 14400`

   Mac: `sudo ./atrdiag -setDuration 14400`

4. 次のコマンドを使用して、前の ATR トークンを削除します。

   Windows: `atrdiag.exe -deleteAllATRs`

   Mac: `./atrdiag -deleteAllATRs`

5. 次に、既存の ATR トークンを上書きします。Tableau Desktop を開きます。Tableau は、「ライセンスが変更されました」というメッセージを表示します。**[終了]** をクリックして、Tableau Desktop をもう一度開きます。

6. 登録ダイアログ ボックスで、**[アクティブ化]** をクリックしてから、既存のトークンを上書きするログインベースのライセンス管理 を使用して Tableau Server 経由で Tableau Desktop をもう一度アクティブ化します。

7. Tableau Desktop を閉じて ATR 期間が経過するまで待機し (4 時間など)、ATR トークンの有効期限が切れてユーザー シートが解放されるようにします。ATR 期間が経過するまで Tableau Desktop は開かないでください。ATR 期間が経過したことを確認してください。ATR トークンの TTL End には、将来の日付と時刻が表示されます (たとえば、今から 4 時間後)。

   Windows: `atrdiag.exe`

   Mac: `./atrdiag`

8. ATR トークンの有効期限が切れたら、新しいコンピューターで Tableau Server に正常にサインインできます。

コンピューターをライセンス認証されていない状態に戻すには

1. 管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを使用し、Tableau バイナリ (\bin) ディレクトリに移動します。

   ```
   cd Program Files\Tableau\Tableau <version>\bin
   ```

3. 次のコマンドを実行します。

Windows: `atrdiag.exe -deleteAllATRs`

Mac: `./atrdiag.exe -deleteAllATRs`

注: これにより、コンピューターからATR トークンのみが削除されます。ユーザー シートは開放されません。ユーザー シートは、削除されたATR トークンの有効期限が切れた後にのみ開放されます。

Tableau 認証資格情報が無効である

Tableau Desktop からライセンス認証を試みると、次のエラー メッセージが表示される場合があります。

Tableau 認証資格情報が無効です。管理者に問い合わせてアカウントをリセットしてください。

このエラーは、Tableau のライセンスが認識されない場合に表示されます。管理者にお問い合わせください。

PC の時計が現在の時刻と同期されていない

Tableau Desktop からライセンス認証を試みると、次のエラー メッセージが表示される場合があります。

PC の時計が現在時刻と同期していません。PC の時計を現在時刻と同期してからTableau をもう一度ライセンス認証してください。

このエラーは、PC の時計が現在の時刻と同期していない場合に表示されます。PC の時計をインターネット上のタイム サーバーと同期するか、自動時刻同期を有効にします。

認証資格情報でライセンス認証できない

Tableau をライセンス認証しようとすると、次のエラー メッセージが表示される場合があります。

Tableau Server はライセンス情報をインターネット経由で確認できません。管理者に問い合わせてインターネット接続を確認してください。

このエラーは、一部のアプリケーション サーバー (VizPortal) ノードでポート `atr.licensing.tableau.com:443` が開いていない場合や、使用しているプロキシがトラフィックを Tableau のライセンス サーバーに転送する適切な構成になっていない場合に表示されます。

Tableau のライセンス サーバーへの接続を診断するには、ブラウザーまたは curl コマンドプロンプトに次の URL (`https://atr.licensing.tableau.com/_status/healthz`) を貼り付けます。

ゼロダウンタイムのライセンス発行

Tableau Server バージョン 2021.1 で導入されたゼロダウンタイムのライセンス発行では、ライセンスの終了日、容量、またはインストール済みの機能の変更時に再起動する必要はなく、ライセンスのほぼすべての変更内容を Tableau Server に適用できます。

Tableau Server を再起動する必要がある場合

再起動を必要としないライセンス更新を適用すると、Tableau Server に次のメッセージが表示されます。**「サーバー全体でライセンスの更新内容が適用されています」**。ただし、ライセンス更新を適用する場合に、Tableau Server の再起動が必要な場合があります。たとえば、機能を減らすプロダクトキーや、許可されたデータソース接続を変更するプロダクトキーをアクティブまたは非アクティブにする場合は、変更を加えた後に Tableau Server を再起動する必要があります。再起動が必要な場合、Tableau Server には、次のメッセージが表示されます: **「ライセンスの更新内容を適用するには、サーバーを再起動してください」**。次の表は、ライセンスの変更時に Tableau Server の再起動が必要か不要かを示したものです。

ライセンスの変更	再起動が必要/不要
ライセンス期間の延長	なし
ユーザー ライセンスの容量の追加または削減	なし
コア ライセンスの容量の追加	なし
コア ライセンスの容量の削減	あり
データ管理 または Advanced Management ライセンスの追加	なし

データ管理 または Advanced Management ライセンスの削除	あり
許可されるデータソースの変更	あり
期限切れのプロダクトキー	あり

Tableau Enterprise について

2025 年 10 月 14 日以降、「Data Cloud」は「Data 360」というブランド名に変わりました。この変更の途上では、アプリケーションやドキュメントに「Data Cloud」と表示されることがあります。名前は新しいですが、機能と内容は変更されていません。

Tableau Enterprise は、Tableau Server を使用して、データをより速く探索および管理できるようにする高度なソフトウェア パッケージです。また、高度なデータ管理と展開管理のオプションを必要としている組織が、必要な機能を容易に購入できるようになります。これには、Tableau のロールベースのライセンス、データ管理、高度な管理、および Creator と Explorer 向けの e ラーニングが含まれます。

Tableau Enterprise ライセンス

Tableau Enterprise は、Creator、Explorer、Viewer などのユーザー ロールベース単位のライセンス モデルで販売されています。このモデルは、組織内の特定のロールを中心に構成されており、それぞれに異なるレベルの機能が必要です。Tableau Enterprise では、各ロールベースのライセンスに、データ管理、高度な管理、および Creator と Explorer 向けの e ラーニングが含まれています。

Tableau Enterprise 機能一覧

次の表に、Tableau Enterprise ライセンス エディションに含まれている機能を示します。以下の機能の利用可否は、Creator について記載されています。Creator、Explorer、Viewer では利用できる機能が異なります。

機能	説明
Tableau オーサリング	Tableau Desktop または Tableau Server 上の Tableau Web 作成機能を使用して、デー

	タに関するインサイトを作成、共同利用、共有します。Tableau を使用すると、問題を特定して解決したり、重要な発見を視覚的にわかりやすく強調表示したりすることができます。
Tableau Prep	Tableau Prep は、分析用にデータをクリーニング、整形、結合するためのデータ準備ツールです。
e ラーニング	ロールベースのトレーニングである Tableau ラーニング パスは、最新の Tableau トレーニング コンテンツを使用して、習熟度向上への明確な道筋を提供します。アセスメントは、学習パスのどこにいるかを評価し、新しいスキルに自信を持つのに役立ちます。新入社員がオンボーディングするプロセスを加速し、経験豊富なユーザーが Tableau の機能を最大限に活用できるようにします。
データ管理 について	Data Management は、お客様が Tableau Server 環境で Tableau コンテンツとデータ アセットを管理するのに役立つ機能のコレクションです。
Tableau Server 上の Tableau Advanced Management について	Advanced Management は、Tableau Server のセキュリティ、管理性、拡張性を強化するために設計された機能のコレクションです。

アイデンティティの移行について

バージョン 2022.1 以降、Tableau Server はアイデンティティサービスを使用してアイデンティティ情報の保存と管理を行います。アイデンティティサービスでは、Tableau Server は、より最新で安全な不変のアイデンティティ構造をユーザーのプロビジョニングと認証プロセスに使用します。アイデンティティプールを構成して使用するためには、アイデンティティの移行が前提条件となります。

注: アイデンティティプール機能を使用する予定がない場合は、アイデンティティの移行を実施しないことをお勧めします。アイデンティティプールを使用する計画がないままアイデンティティの移行を実施しても、Tableau Server の導入でメリットは得られません。

既存の展開の場合、Tableau Server をバージョン 2022.1 以降にアップグレードして、Tableau 2021.4 以前のバックアップをリストアすると、Tableau Server のアップグレードが完了した後にアイデンティティの移行が始まり、新しいアイデンティティサービスにデータが入力されます。アイデンティティの移行により、すべての Tableau Server ユーザー用に補助的なアイデンティティサービス テーブルが作成されます。このテーブルは、アイデンティティサービスでユーザーを認証するために使用されます。移行はバックグラウンドで実行されるため、ユーザーが Tableau Server の使用を中断したり影響を受けたりすることはありません。

管理者は、Tableau Server の [ユーザー] ページから利用できる専用の **[アイデンティティ移行]** ページを使用して、移行の監視と管理を行うことができます。この管理には、移行を実施するタイミングの変更や、移行の競合が発生した場合の解決なども含まれます。このページは、移行プロセスの期間中に使用できます。

既存の展開に対する手順の概要

新規および既存の展開の場合、アイデンティティ構成の改善を利用してアイデンティティプールを構成できるように、移行が完了したらアイデンティティサービスを使用するように Tableau Server を構成する必要があります。

ステップ 1: 始める前に

ステップ 2: 新しいスキーマとサインイン機能を有効にする

ステップ 3: アイデンティティの移行を開始する

ステップ 4: アイデンティティの移行を完了する

ステップ 5: アイデンティティサービスを使用するように Tableau Server を構成する

ステップ 6: アイデンティティプールを有効にする

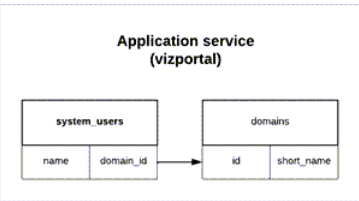
重要な用語

- **アイデンティティ サービス -** ユーザー アイデンティティの管理を担うTableau Server 2022.1 以降のサービス。認証やプロビジョニングのサービスも含まれます。このサービスが使用するアイデンティティスキーマは、アイデンティティサービス テーブルと従来の「system_users」テーブルでユーザー アイデンティティを表します。
- **アイデンティティプール -** ユーザーが Tableau Server へアクセスできるようにするための、プロビジョニングと認証情報を使用したアイデンティティ管理ツールです。ID プールにより、より集中的で柔軟な ID 管理ワークフローが ID サービス上に構築され、Tableau Server でユーザー ID を保存し、管理することが可能になります。
- **レガシー アイデンティティ ストア モード -** Tableau Server 2021.4 以前で使用されていた制限付きのアイデンティティスキーマ。ユーザー アイデンティティは従来の「system_users」テーブルでのみ表されます。
- **アイデンティティの移行 -** 既存の Tableau Server ユーザー アイデンティティを評価し、アップストリームの外部アイデンティティストアに対して (必要に応じて) 追加のアイデンティティ情報のクエリを実行し、その追加のアイデンティティ情報をアイデンティティサービスにインポートする監査プロセス。
- **外部 アイデンティティ ストア -** Tableau Server の外部にある上流のアイデンティティストア タイプ。外部のディレクトリサービス (Active Directory (AD) または LDAP) ですべてのアイデンティティ情報が格納され、管理されます。構成されると、Tableau Server はこの外部ディレクトリと同期して、アイデンティティ情報のコピーを Tableau Server に保持します。
- **ローカル アイデンティティ ストア -** Tableau Server が提供するアイデンティティストア タイプ。構成されると、Tableau Server は、アイデンティティ情報用に構成された外部ディレクトリがなくても、Tableau Server リポジトリにアイデンティティ情報を保管して管理します。
- **システム ユーザー -** Tableau Server のユーザー。ユーザーは、アイデンティティサービス (「system_users_identities」テーブル経由) と従来のアイデンティティストア モードの両方のサインイン レコード (「system_users」) に対応します。複数のサイトにサインインできるように、複数のユーザー アイデンティティが 1 つの「system_users」レコードに関連付けられている場合があります。1 つの「system_user」レコードと複数のサイトの間のつながりは、「users」テーブルで定義されます。

アイデンティティの移行の目的

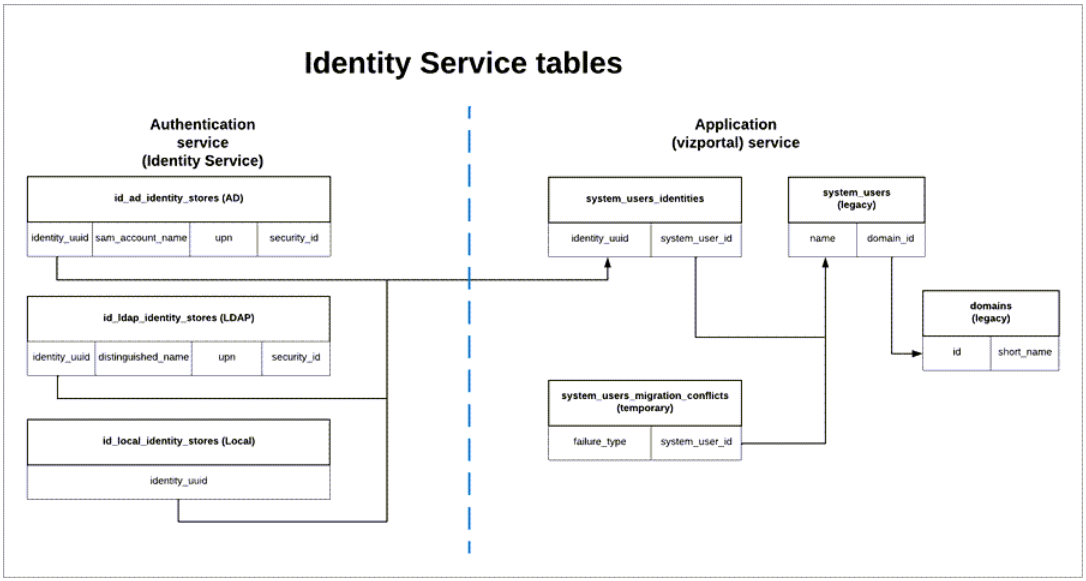
Tableau Server のバックアップを作成すると、アイデンティティ情報は、バックアップを作成した Tableau Server のバージョンで使用されるアイデンティティスキーマに保存されます。バックアップが使用するアイデンティティスキーマからアイデンティティサービスが使用するアイデンティティスキーマにアイデンティティ情報を取り込むために、移行が必要となります。

Tableau Server 2021.4 以前のアイデンティティスキーマ



従来のアイデンティティストア モードで使用されるアイデンティティスキーマは、「system_users」と「domains」の 2 つのテーブルで構成されています。

Tableau Server 2022.1 以降のアイデンティティスキーマ



アイデンティティサービスで使用されるアイデンティティスキーマは、従来の「system_users」テーブルと、より多くのアイデンティティ情報を取り込む補助的なアイデンティティサービス テーブル (*_identity_stores および *identities) とで構成されています。この追加のテーブルによって、上流の外部アイデンティティストアでの変更が原因で発生する問題を減らすことができます。

アイデンティティ移行時の注意点

ユーザー アイデンティティに関する情報を移行すると、従来の「system_users」テーブルに格納されていたアイデンティティ情報はアイデンティティサービス テーブルで補足されます。

アイデンティティ情報を補足するアイデンティティサービステーブルのタイプは、Tableau Server が構成されているアイデンティティストアのタイプ (ローカル、Active Directory (AD)、Lightweight Directory Access Protocol (LDAP) によって異なります。

- **AD** アイデンティティストア タイプの場合、アイデンティティサービス テーブルは、あいまいではない属性、または同じデータベース レコードに格納されていない属性のみを継承します。

  たとえば、sAMAccountNAme とuserPrincipalName (UPN) が従来の「systems_users」テーブルの同じ名前のレコードに格納されている場合があります。これは、複雑な一連のルールの結果として発生する可能性があります。ほとんどの場合、移行はユーザー アイデンティティを正しく解釈し、正常に移行できます。ただし、移行であいまいな結果が発生した場合は、あいまいさを手動で承認するか、専用の [アイデンティティ移行] ページを使用して競合を手動で解決する必要があります。詳細については、「アイデンティティ移行時の競合の解決」を参照してください。

- **LDAP** アイデンティティストア タイプの場合、AD アイデンティティストア タイプの場合と同じように、アイデンティティサービス テーブルはあいまいではない属性のみを継承します。ほとんどの場合、移行はユーザー アイデンティティを正しく解釈し、正常に移行できます。ただし、移行であいまいな結果が発生した場合は、あいまいさを手動で承認するか、専用の [アイデンティティ移行] ページを使用して競合を手動で解決する必要があります。詳細については、「アイデンティティ移行時の競合の解決」を参照してください。

- **ローカル** アイデンティティストア タイプの場合、アイデンティティサービス テーブルはユーザー フィールドとドメイン フィールドを直接継承します。つまり、追加する情報や手動での解決は必要ありません。Tableau Server がこのタイプのアイデンティティストア用に構成されている場合、ユーザー アイデンティティの移行は、Tableau Server のバックアップをリストアするプロセスの後に行われます。

アイデンティティの移行を実施する

ステップ 1: 始める前に

始める前に、以下で Tableau Server のバージョンとアップグレード方法 (該当する場合) を特定し、アイデンティティ移行の次のステップを決定します。

アップグレードする前のバージョン	アップグレード先のバージョン	アップグレード方法	次のステップ
N/A	Tableau Server バージョン: 2022.1.0 (以降)、2022.3.0 (以降)、2023.1 (以降) 以降	新規インストール	ステップ 2 に進みます。
Tableau Server バージョン: 2022.1-2022.1.7、2022.3-2022.3.9、または 2023.1-2023.1.5	Tableau Server バージョン: 2022.1.8 (以降)、2022.3.10 (以降)、2023.1.6 (以降) 以降	N/A	サインイン用の新しいスキーマを有効にしましたか (例: `wgserver.authentication.legacy_identity_mode.enabled -v false`)? • いいえの場合は、ステップ 2 に進みます。 • はいの場合は、ステップ 6 に進みます。
Tableau Server バージョン: 2021.4.x 以前	Tableau Server バージョン: 2022.1.8 (以降)、2022.3.10 (以降)、2023.1.6 (以降) 以降	N/A	ステップ 2 に進みます。
Tableau Server バージョン:	Tableau Server バージョン:	1) 新しいマシンに Tableau	「アイデンティティの移行に関する問題をトラブルシューティングする」に進みます。

アップグレードする前のバージョン	アップグレード先のバージョン	アップグレード方法	次のステップ
2021.4.x 以前	2022.1.0-2022.1.7、2022.3.0-2022.3.9、または 2023.1.0-2023.1.5	Server をインストールし、2) tsm メンテナンス (バックアップと復元) コマンドを使用して **Tableau Server をバックアップおよび復元**することによって、Tableau Server の**ブルー/グリーンアップグレード**または**手動によるアップグレード**を実行する場合、いくつかの追加ステップに従って移行を開始する必要があります。	
		ここで説明する方法で Tableau Server の**単一サーバーまたはマルチノードの「インプレース」アップグレード**を	ステップ 2 に進みます。

アップグレードする前のバージョン	アップグレード先のバージョン	アップグレード方法	次のステップ
		行う場合は、移行を開始するために追加の手順は必要ありません。移行は、Tableau Server をバージョン 2022.1 以降へアップグレードした後に開始されます。	
		1) 新しいマシンに Tableau Server をインストールして、2) tsm settings (エクスポートとインポート) **コマンドを使用して構成とトポロジの情報をエクスポートおよびインポートする**ことにより、Tableau Server を**手動でアップグレードする**場合も、移行を開始するために追加の手順は必要	ステップ 2 に進みます。

アップグレードする前のバージョン	アップグレード先のバージョン	アップグレード方法	次のステップ
		ありません。移行は、新しい Tableau Server マシンでインポートプロセスが完了した後に開始されます。	

ステップ 2: 新しいスキーマとサインイン機能を有効にする

アイデンティティの移行を開始する前に、tsm コマンド features.NewIdentityMode で新しいアイデンティティスキーマを有効にして、Tableau Server に引き続きサインインできるようにしながら、wgserver.authentication.legacy_identity_mode.enabled または tsm authentication legacy-identity-mode <commands> tsm コマンドでアイデンティティの移行プロセスが行われるようにします。

1. クラスタの最初のノード (TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

   ```
   tsm configuration set -k features.NewIdentityMode -v true
   ```

3. 実行している Tableau Server のバージョンに応じて、次のコマンドを実行します。

   - Tableau Server バージョン 2022.1.8 以降、2022.3.10 以降、2023.1.6 以降の場合:

     ```
     tsm configuration set -k wgserver.authentication.legacy_
     identity_mode.enabled -v true
     tsm pending-changes apply
     ```

- Tableau Server バージョン: 2022.1.0-2022.1.7、2022.3-2022.3.9、または 2023.1-2023.1.5 の場合:

```
tsm authentication legacy-identity-mode enable
tsm pending-changes apply
```

注: これらのコマンドを実行すると、Tableau Server が再起動されます。

ステップ 3: アイデンティティの移行を開始する

注: Tableau Server バージョン 2022.1-2022.1.7、2022.3-2022.3.9、または 2023.1-2023.1.5 にアップグレードした場合は、デフォルトでアイデンティティの移行が始まるので、以下の「ステップ 4: アイデンティティの移行を完了する」にスキップできます。

アイデンティティの移行を開始するには、tsm コマンド features.IdentityMigrationBackgroundJob を使用してアイデンティティの移行機能を有効にする必要があります。

1. 次のコマンドを実行します。

```
tsm configuration set -k
features.IdentityMigrationBackgroundJob -v true
```

アイデンティティの移行が始まると、Tableau Server に [アイデンティティ移行] ページにリンクする通知が表示されます。[アイデンティティ移行] ページでは、アイデンティティの移行のステータスと、解決する必要があるアイデンティティの競合を監視できます。

ステップ 4: アイデンティティの移行を完了する

注記: 新規インストールの場合はこの手順をスキップしてください。

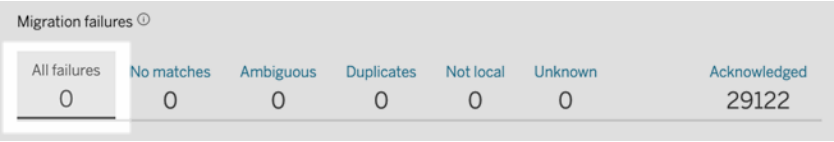
アイデンティティの移行を完了するには、Tableau Server のアイデンティティサービスを有効にする前に、すべてのアイデンティティの競合を解決または承認する必要があります。

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイト Tableau Server の場合は **[すべてのサイト] > [ユーザー]** を選択し、**[Identity Migration (アイデンティティの移行)]**

ページをクリックして、移行が開始されたことを確認します。

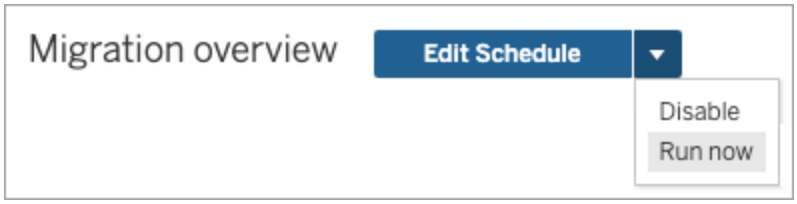
Tableau Serverの [ユーザー] ページから、専用の [アイデンティティの移行] ページを使用して、進行状況を監視および管理できます。詳細については、「アイデンティティの移行を管理する」を参照してください。

3. 「アイデンティティ移行時の競合の解決」の説明に従って、すべてのID競合を解決または確認して、下の画像のように **[All failures (すべての失敗)]** タブに「0」が表示されるようにします。



4. 次のいずれかを実行します。

   - アイデンティティの移行ジョブを今すぐ実行するには、[Migration Overview (移行の概要)] 見出しの横にある [Edit Schedule (スケジュールの編集)] ドロップダウン矢印をクリックし、**[今すぐ実行]** を選択します。



   - または、次にスケジュールされた時間に移行ジョブが実行されるのを待つこともできます。

5. 移行が完了したら、[アイデンティティの移行] ページで、[移行の概要] に **[100% 完了]** が表示されていることを確認します。

ステップ 5: アイデンティティサービスを使用するように Tableau Server を構成する

アイデンティティの移行が完了したら、アイデンティティサービスを使用するように Tableau Server を構成します。これにより、ユーザーのプロビジョニングと認証プロセスでより安全で不変のアイデンティティ構造が確保されます。

1. クラスターの初期ノード (TSM がインストールされているノード) で、管理者としてコマンドプロンプトを開きます。
2. 実行している Tableau Server のバージョンに応じて、次のコマンドを実行します。

    - Tableau Server バージョン 2022.1.8 以降、2022.3.10 以降、2023.1.6 以降の場合:

      ```
      tsm configuration set -k wgserver.authentication.legacy_
      identity_mode.enabled -v false
      ```

    - Tableau Server バージョン 2022.1 ~ 2022.1.7、2022.3 ~ 2022.3.9、2023.1 ~ 2023.1.5 の場合:

      ```
      tsm authentication legacy-identity-mode disable
      ```

    注: 上記のコマンドのいずれかを実行すると、専用の **[ID 移行]** ページが削除され、アクセスできなくなります。ページにアクセスできるのは、`tsm configuration set -k wgserver.authentication.legacy_identity_mode.enabled` が `true` であるか、`tsm authentication legacy-identity-mode` が有効になっている場合のみです。

Tableau Server をアイデンティティサービスを使用するように構成した後、ユーザーが Tableau Server にサインインすると、Tableau Server は構成されたアイデンティティストア内の識別子を使用してユーザー アイデンティティを検索します。識別子からユニバーサル一意識別子 (UUID) が返され、既存の Tableau Server ユーザー アイデンティティと照合するために使用されます。次に、このプロセスはユーザーのセッションを生成し、認証ワークフローを完了します。

ステップ 6: アイデンティティプールを有効にする

アイデンティティの移行を完了し、アイデンティティサービスを有効にしたら、アイデンティティプールの構成の準備としてアイデンティティプール機能を有効にします。

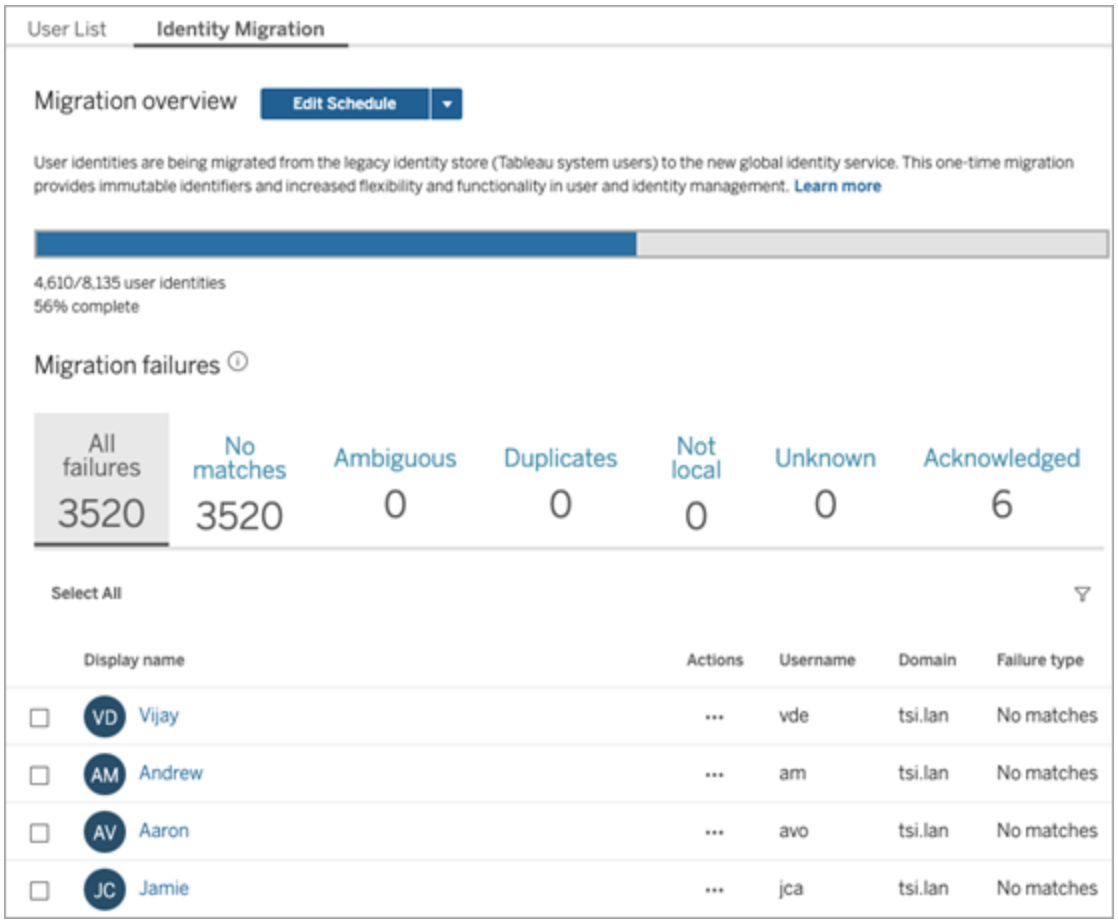
1. 次のコマンドを実行します。

```
tsm configuration set -k features.IdentityPools -v true
tsm pending-changes apply
```

注: これらのコマンドを実行すると、Tableau Server が再起動されます。

アイデンティティの移行を管理する

管理者は、Tableau Server の [ユーザー] ページから利用できる専用の [アイデンティティの移行] ページを使用して、移行ジョブの実行時の変更を含む、アイデンティティの移行を監視および管理できます。このページは、移行プロセスの期間中に使用できます。



移行ジョブは、Tableau Server の使用を中断または妨害することなくバックグラウンドで実行されるように設計されています。ただし、必要に応じて、移行ジョブの実行頻度、移行ジョブの実行日時、および移行ジョブの実行時間に影響を与える調整を行うことができます。

通常、移行には 3 分から 10 日かかる場合があります。所要時間は、Tableau Server の展開サイズと、移行中に行った既定設定への変更に応じて異なります。たとえば、10,000 人のユーザーがいる場合、移行には約 30 分かかることがあります。

注: 移行ジョブの実行中は、すべての認証およびユーザー関連の機能が正常に機能します。

アイデンティティの移行ジョブを管理する

アイデンティティの移行の次の側面を管理できます。

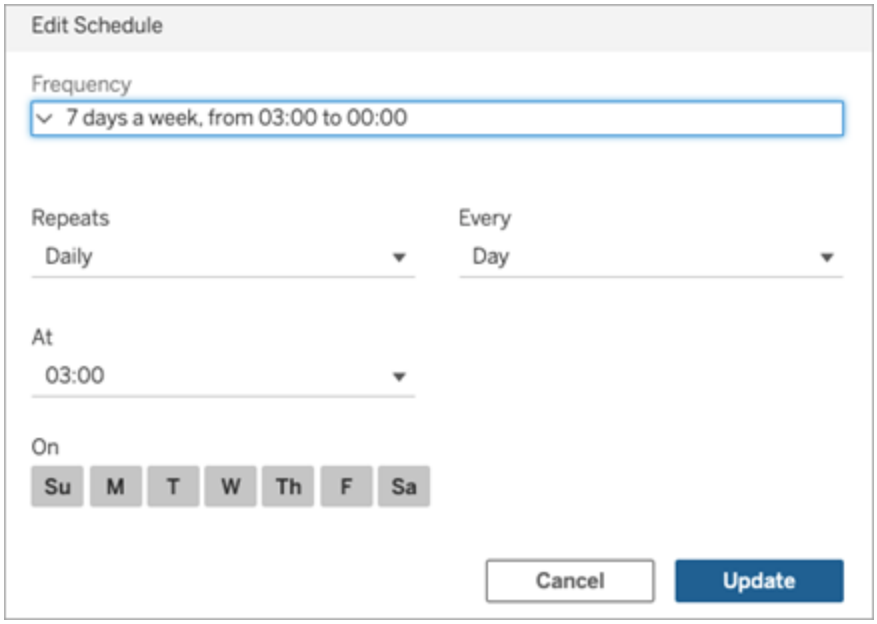
アイデンティティの競合を解決する

移行中に発生する可能性のあるアイデンティティの競合を確認するには、「アイデンティティ移行時の競合の解決」を参照してください。

毎日の移行ジョブスケジュールを変更する

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイトTableau Server の場合は **[すべてのサイト] > [ユーザー]**) を選択し、**[Identity Migration (アイデンティティの移行)]** ページをクリックします。
3. [移行の概要] 見出しの横にある **[スケジュールの編集]** ボタンをクリックします。
4. [スケジュールの編集] ダイアログボックスで、ジョブを実行できる日時と頻度を変更します。

   注: このダイアログ ボックスの **[Priority (優先度)]** オプションと **[実行]** オプションは無視することができます。

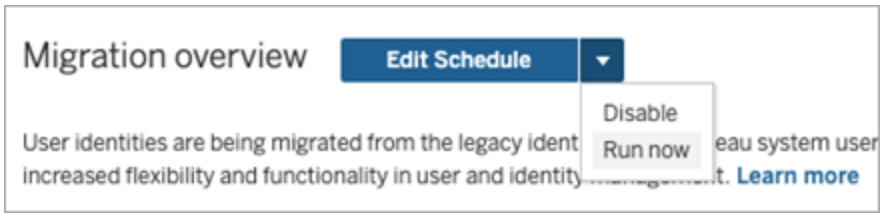


5. 終了したら、**[更新]** をクリックします。

移行ジョブを開始する

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイトTableau Server の場合は **[すべてのサイト] > [ユーザー]**) を選択し、**[Identity Migration (アイデンティティの移行)]** ページをクリックします。
3. 任意のタブから、1 人または複数のユーザーを選択します。
4. [アクション] メニューから、実行する必要がある内容に応じて、**[Retry Migration (移行の再試行)]** または **[Acknowledge (確認)]** を選択します。

5. [移行の概要] 見出しの横にある [スケジュールの編集] ドロップダウン矢印をクリックします。
6. **[今すぐ実行]** を選択します。



アイデンティティの移行を一時停止する

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイトTableau Server の場合は **[すべてのサイト] > [ユーザー]**) を選択し、**[Identity Migration (アイデンティティの移行)]** ページをクリックします。

3. [移行の概要] 見出しの横にある [スケジュールの編集] ドロップダウン矢印をクリックします。
4. **[Disable (無効にする)]** を選択します。

アイデンティティの移行を再開する

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイト Tableau Server の場合は **[すべてのサイト] > [ユーザー]**) を選択し、**[Identity Migration (アイデンティティの移行)]** ページをクリックします。
3. [移行の概要] 見出しの横にある [スケジュールの編集] ドロップダウン矢印をクリックします。
4. **[Enable (有効にする)]** を選択します。

アイデンティティの移行設定を変更する

アイデンティティの移行が Tableau Server に与える可能性のある制約を軽減するために、移行は以下にリストされている既定設定で実行されるように構成されています。

移行設定

タイプ	TSM コマンド	既定	手順
ジョブのスケジュール	N/A	毎日午前 3 時、完了するまで	[アイデンティティ移行] ページから設定できます。上記の毎日の移行ジョブスケジュールを変更するを参照してください。
1	`tsm authentication`	5 ま	必要に応じて、`tsm`

秒あたりのユーザーアイデンティティのリクエスト(レート)	`identity-migration configure --rate`	で	`authentication identity-migration` コマンドを使用して、以下のステップに従って、上記の移行設定を変更できます。 1. クラスタの最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。 2. `tsm authentication identity-migration` で説明されているコマンドの 1 つまたは両方を実行します。 たとえば、個々のジョブの実行時間とレートを既定値から変更するには、次のコマンドを実行できます。 `tsm authentication identity-migration configure --job-run-time 180 --rate 3`
個々のジョブの実行時間	`tsm authentication identity-migration configure --job-run-time`	120 分	
アイデンティティ	`tsm configuration set -k features.IdentityMigrationBackgroundJob`	false	アイデンティティの移行を有効にすると、Tableau Server はアイデンティティサービスを使用してユーザー アイデンティティ情報を保存および管理できるようになります。

の移行を有効にする			1. クラスタの最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。 2. 次のコマンドを実行します。 `tsm configuration set -k features.IdentityMigrationBackgroundJob -v true` **注:** アイデンティティの移行とアイデンティティサービスは、アイデンティティプールなどの特定の機能の前提条件です。tsm コマンドの詳細については、features.IdentityMigrationBackgroundJobを参照してください。

アイデンティティの移行を無効にする

Tableau Server バージョン 2021.4.21、2022.1.17、2022.3.9、および 2023.1.5 にアップグレードした場合は、アイデンティティの移行を無効にすることが必要になる場合があります。アイデンティティの移行を無効にすると、Tableau Server はアイデンティティサービスを使用してユーザー アイデンティティ情報を保存および管理できなくなります。

1. クラスタの最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

```
tsm configuration set -k
features.IdentityMigrationBackgroundJob -v false
```

注: アイデンティティの移行とアイデンティティサービスは、アイデンティティプールなどの特定の機能の前提条件です。

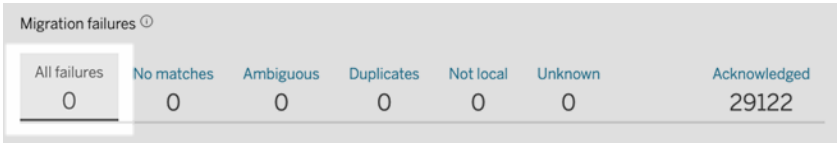
アイデンティティの移行を完了し、アイデンティティサービスを構成します

すべてのユーザーの競合を解決または対処し、移行ジョブを実行したら、アイデンティティサービスを使用してアイデンティティ移行プロセスを完了するように Tableau Server を構成する必要があります。

ステップ 1: アイデンティティの移行を検証して完了する

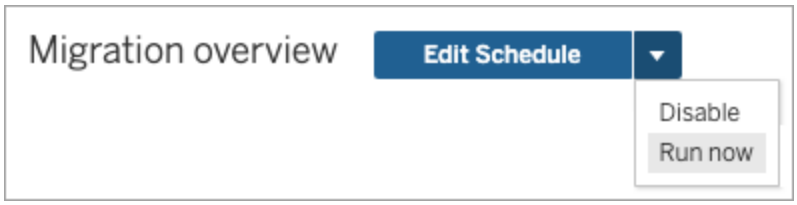
1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーションペインで、**[ユーザー]** (またはマルチサイト Tableau Server の場合は **[すべてのサイト] > [ユーザー]** を選択し、**[Identity Migration (アイデンティティの移行)]** ページをクリックして、移行が開始されたことを確認します。

   Tableau Serverの [ユーザー] ページから、専用の [アイデンティティの移行] ページを使用して、進行状況を監視および管理できます。詳細については、「アイデンティティの移行を管理する」を参照してください。
3. 「アイデンティティ移行時の競合の解決」の説明に従って、すべての ID 競合を解決または確認して、下の画像のように **[All failures (すべての失敗)]** タブに「0」が表示されるようにします。



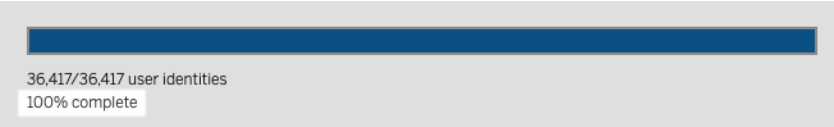
4. 次のいずれかを実行します。

- アイデンティティの移行ジョブを今すぐ実行するには、[Migration Overview (移行の概要)] 見出しの横にある [Edit Schedule (スケジュールの編集)] ドロップダウン矢印をクリックし、**[今すぐ実行]** を選択します。



- または、次にスケジュールされた時間に移行ジョブが実行されるのを待つこともできます。

5. 移行が完了したら、[アイデンティティの移行] ページで、[移行の概要] に **[100% 完了]** が表示されていることを確認します。



ステップ 2: アイデンティティサービスを使用するように Tableau Server を構成する

1. クラスターの初期ノード (TSM がインストールされているノード) で、管理者としてコマンドプロンプトを開きます。
2. 実行している Tableau Server のバージョンに応じて、次のコマンドを実行します。

   - Tableau Server バージョン 2022.1.8 以降、2022.3.10 以降、2023.1.6 以降の場合:

     ```
     tsm configuration set -k wgserver.authentication.legacy_
     identity_mode.enabled -v false
     ```

   - Tableau Server バージョン 2022.1 ~ 2022.1.7、2022.3 ~ 2022.3.9、2023.1 ~ 2023.1.5 の場合:

     ```
     tsm authentication legacy-identity-mode disable
     ```

注: 上記のコマンドのいずれかを実行すると、専用の **[ID 移行]** ページが削除され、アクセスできなくなります。ページにアクセスできるのは、`tsm configuration set -k wgserver.authentication.legacy_identity_mode.enabled` が `true` であるか、`tsm authentication legacy-identity-mode` が有効になっている場合のみです。

Tableau Server をアイデンティティサービスを使用するように構成した後、ユーザーが Tableau Server にサインインすると、Tableau Server は構成されたアイデンティティストア内の識別子を使用してユーザー アイデンティティを検索します。識別子からユニバーサル一意識別子 (UUID) が返され、既存の Tableau Server ユーザー アイデンティティと照合するために使用されます。次に、このプロセスはユーザーのセッションを生成し、認証ワークフローを完了します。

ステップ 3: アイデンティティプールを有効にする

1. 次のコマンドを実行します。

```
tsm configuration set -k features.IdentityPools -v true
tsm pending-changes apply
```

注: これらのコマンドを実行すると、Tableau Server が再起動されます。

アイデンティティ移行時の競合の解決

Tableau Server でアイデンティティサービスを使用するようにアイデンティティを移行するとき、特定のユーザー アイデンティティについては移行できない場合があります。移行できないユーザー アイデンティティは競合となり、管理者が手動で解決する必要があります。

ユーザー アイデンティティが正しく移行されるようにするには、アイデンティティの移行を完了する前に、専用の **[アイデンティティ移行]** ページですべてのアイデンティティの競合を解決または対処する必要があります。

ステップ 1: アイデンティティの競合を解決する

アイデンティティの競合を解決するには、競合の種類に応じていくつかの方法があります。競合の種類がどれであっても、以下の「ステップ 2」に進む前に、またアイデンティティの移行プロセスを完了する前に、ユーザー アイデンティティの競合はすべて解決または対処する必要があります。

アイデンティティの競合が発生すると、アイデンティティの移行プロセスが競合を種類別にグループ化します。これらの種類は、移行プロセスがユーザー アイデンティティを自動的に移行できない理由を絞り込むのに役立ちます。

アイデンティティの競合が発生する理由はいくつかあります。たとえば、1 つの Tableau Server ユーザーが、外部アイデンティティストアの複数のユーザー アイデンティティに対応すると移行プロセスが特定した場合、アイデンティティの競合が発生する可能性があります。

特定されたアイデンティティの競合に対処するには、次のいずれかのオプションを実施します。

- **移行の再試行 -** このオプションは、選択したユーザー アイデンティティをキューに戻してもう一度移行します。移行ジョブを再実行すると、アイデンティティの競合が自動的に解決されるか、元のアイデンティティの競合が再度発生するか、新しいアイデンティティの競合が発生する可能性があります。

- **確認 -** このオプションは、選択したユーザー アイデンティティを **[確認済み]** タブに移動します。ユーザー アイデンティティの確認を行うと、1) 一致するユーザー アイデンティティがアイデンティティストアにないため、このユーザーは移行されないこと、および 2) 以下のステップ 3 でアイデンティティサービスを有効すると、このユーザーは Tableau Server にサインインできなくなることを了承することになります。

- **再評価 -** 確認済みの競合について、**[確認済み]** タブでこのオプションを使用すると、選択したユーザー アイデンティティが競合状態に戻ります。このオプションを使用すると、元の競合を調べたり、競合を解決したり、もう一度アイデンティティの競合を確認したりできます。

クイック リファレンス: アイデンティティの競合

競合の種類	構成への適用	競合の理由	アクション
すべての失敗	すべて	このタブでは、アイデンティティの競合をすべて捕捉します。アイデンティティの競合は、[一致なし]、[あいまい]、[重複]、[ローカルではない]、および [不明] タブに分類されます。	移行の再試行または確認
一致	AD、	このユーザー アイデンティティに対応するユー	移行の再試行ま

なし	LDAP	ザーが、外部アイデンティティストアにありません。	たは確認
あいまい	AD、LDAP	このユーザー アイデンティティに対応する可能性のあるユーザーが、外部アイデンティティストアに複数あります。	移行の再試行、確認、または提案されたユーザー アイデンティティの 1 つを選択
重複	AD	1 つの AD アカウントを使用して 2 つのユーザー アイデンティティが作成されました。これは、アイデンティティサービスではサポートしていないレガシー機能の産物です。	移行の再試行または確認
ローカルではない	ローカル	ローカルではないユーザー アイデンティティがアイデンティティストアに関連付けられています。この競合は、サポートしていない手動の変更が行われたために発生します。	移行の再試行または確認
不明	すべて	この競合は、Tableau Server の内部エラー、またはこの表に記載されていない理由によって引き起こされたアイデンティティの競合を示している可能性があります。	移行の再試行または確認
確認	すべて	このタブでは、移行しないすべてのユーザー アイデンティティを捕捉します。アイデンティティサービスを使用するように Tableau Server を設定すると、これらのユーザーは Tableau Server にサインインできなくなります。	移行の再試行または確認

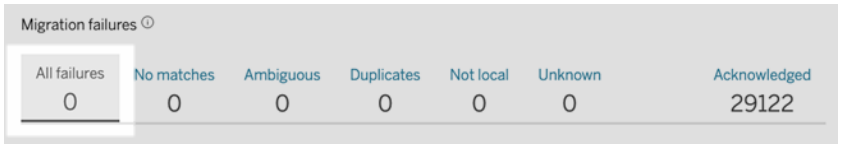
競合を解決するには、次の手順を行います。

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイト Tableau Server の場合は **[すべてのサイト] > [ユーザー]**) を選択し、**[Identity Migration (アイデンティティの移行)]** ページをクリックします。

3. **[すべての失敗]** タブまたは固有の競合のタブのいずれかで、1 つまたは複数のユーザー アイデンティティを選択します。

4. [アクション] ドロップダウン メニューから、**[移行の再試行]** または **[確認]** をクリックします。

   [移行の再試行] を選択すると、ユーザー アイデンティティによっては別の競合の種類が生成される可能性があります。この場合、下の画像のように **[All failures (すべての失敗)]** タブに「0」が表示されるまで、必要に応じて競合に対処します。



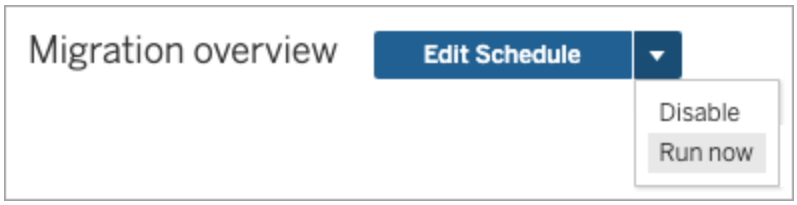
   注: 確認されたユーザー アイデンティティは、今後の Active Directory (AD) および LDAP のグループ同期では無視されます。これらのユーザー アイデンティティに関連付けられたユーザーが後で Tableau Server にアクセスする必要がある場合、詳細については「アイデンティティ移行に関する問題のトラブルシューティング」を参照してください。

ステップ 2: アイデンティティの移行を完了する

アイデンティティの移行を完了するには、アイデンティティの競合をすべて解決または対処することに加えて、すべての移行ジョブを実行する必要があります。その後、Tableau Server のアイデンティティサービスを有効にすることができます。

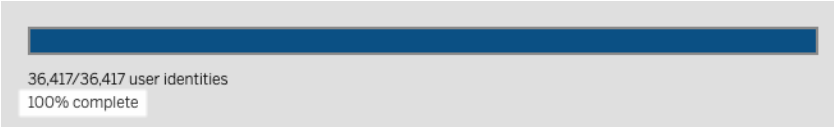
1. 次のいずれかを実行します。

   - アイデンティティの移行ジョブをその場で実行するには、[移行の概要] 見出しの横にある [スケジュールの編集] ドロップダウン矢印をクリックし、**[今すぐ実行]** を選択します。



- または、次回スケジュールした時刻に移行が実行されるのを待つこともできます。

2. [アイデンティティ移行] ページで、[移行の概要] に **[100% 完了]** と表示されていることを確認します。



ステップ 3: アイデンティティサービスを使用するように Tableau Server を構成する

アイデンティティの移行が完了したら、アイデンティティサービスを使用するように Tableau Server を構成します。これにより、ユーザーのプロビジョニングと認証プロセスでより安全で不変のアイデンティティ構造が確保されます。

1. クラスターの初期ノード (TSM がインストールされているノード) で、管理者としてコマンドプロンプトを開きます。
2. 実行している Tableau Server のバージョンに応じて、次のコマンドを実行します。

   - Tableau Server バージョン 2022.1.8 以降、2022.3.10 以降、2023.1.6 以降の場合:

     ```
     tsm configuration set -k wgserver.authentication.legacy_
     identity_mode.enabled -v false
     ```

   - Tableau Server バージョン 2022.1 ~ 2022.1.7、2022.3 ~ 2022.3.9、2023.1 ~ 2023.1.5 の場合:

     ```
     tsm authentication legacy-identity-mode disable
     ```

> **注:** 上記のコマンドのいずれかを実行すると、専用の **[ID 移行]** ページが削除され、アクセスできなくなります。ページにアクセスできるのは、`tsm configuration set -k wgserver.authentication.legacy_identity_mode.enabled` が `true` であるか、`tsm authentication legacy-identity-mode` が有効になっている場合のみです。

Tableau Server をアイデンティティサービスを使用するように構成した後、ユーザーが Tableau Server にサインインすると、Tableau Server は構成されたアイデンティティストア内の識別子を使用してユーザー アイデンティティを検索します。識別子からユニバーサル一意識別子 (UUID) が返され、既存の Tableau Server ユーザー アイデンティティと照合するために使用されます。次に、このプロセスはユーザーのセッションを生成し、認証ワークフローを完了します。

ステップ 4: アイデンティティプールが有効になっていることを確認する

1. 次のコマンドを実行します。

```
tsm configuration set -k features.IdentityPools -v true
tsm pending-changes apply
```

注: これらのコマンドを実行すると、Tableau Server が再起動されます。

アイデンティティの移行に関する問題をトラブルシューティングする

バックアップを復元できません

デフォルトのスキーマとしてアイデンティティサービスを使用する Tableau Server バージョン (バージョン 2022.1-2022.1.7、2022.3-2022.3.9、および 2023.1-2023.15) にアップグレードした後、Tableau Server バックアップを復元すると、次のエラーが発生する可能性があります。

"Tableau Server は既定で新しいアイデンティティサービスのテーブルを使用するため、バックアップを復元できません。"

この問題は、Tableau Server がアイデンティティサービスを設定するのに必要なプロセスであるアイデンティティの移行を実行する必要がある場合に発生します。アイデンティティサービスは、Tableau Server バージョン 2022.1 以降のいくつかのバージョンに導入されたアイデンティティスキーマで、ユーザーのプロビジョニングと認証に使用されます。潜在的な問題を回避するため、Tableau Server の

バックアップで復元したバージョンと異なるアイデンティティスキーマを使用していることが検出された場合は、アップグレードプロセスは続行できません。

この問題を解決するには、以下に説明するステップを実行します。

重要: 以下で説明する手順は、Tableau Server バージョン 2022.1.0-2022.1.7、2022.3.0-2022.3.9、および 2023.1.0-2023.15 を実行している場合にのみ適用されます。

ステップ 1: `legacy-identity-mode` を有効にして、バックアップを復元する

1. クラスターの初期ノード (TSM がインストールされているノード) で、管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行して、従来のアイデンティティストア モードを使用するように Tableau Server 2022.1 (またはそれ以降) を設定します。

   ```
   tsm authentication legacy-identity-mode enable
   tsm pending-changes apply
   ```

   Tableau Server は、従来のアイデンティティストア モードを使用して、アイデンティティサービスにデータを取り込む必要があります。tsm コマンドの詳細については、`tsm authentication legacy-identity-mode` を参照してください。
3. 次のコマンドを実行して、バックアップをもう一度復元し、移行を開始できるようにします。

   ```
   tsm maintenance restore --file <file_name>
   tsm start
   ```

   重要: バックアップの復元後、移行によって、アイデンティティサービスにアイデンティティ情報が入力されます。バックアップからの復元の概要については、バックアップからの復元を参照してください。

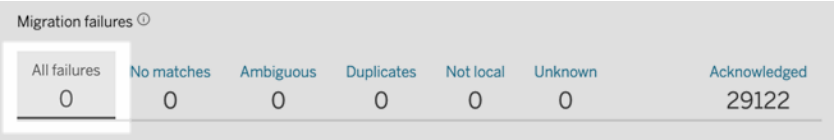
ステップ 2: アイデンティティの移行を検証して完了する

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインで、**[ユーザー]** (またはマルチサイト Tableau Server の場合は **[すべてのサイト] > [ユーザー]** を選択し、**[Identity Migration (アイデンティティの移行)]**

ページをクリックして、移行が開始されたことを確認します。

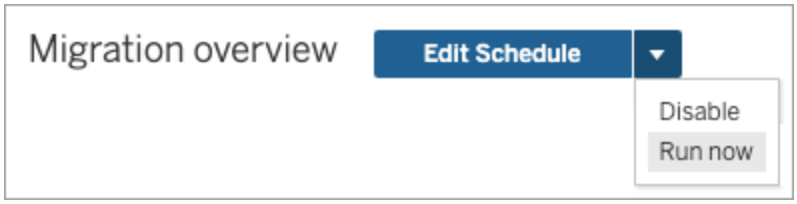
Tableau Serverの [ユーザー] ページから、専用の [アイデンティティの移行] ページを使用して、進行状況を監視および管理できます。詳細については、「アイデンティティの移行を管理する」を参照してください。

3. 「アイデンティティ移行時の競合の解決」の説明に従って、すべてのID競合を解決または確認して、下の画像のように **[All failures (すべての失敗)]** タブに「0」が表示されるようにします。



4. 次のいずれかを実行します。

   - アイデンティティの移行ジョブを今すぐ実行するには、[Migration Overview (移行の概要)] 見出しの横にある [Edit Schedule (スケジュールの編集)] ドロップダウン矢印をクリックし、**[今すぐ実行]** を選択します。



   - または、次にスケジュールされた時間に移行ジョブが実行されるのを待つこともできます。

5. 移行が完了したら、[アイデンティティの移行] ページで、[移行の概要] に **[100% 完了]** が表示されていることを確認します。

ステップ 3: アイデンティティサービスを使用するように Tableau Server を構成する

1. クラスターの初期ノード(TSM がインストールされているノード) で、管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを実行します。

   ```
   tsm authentication legacy-identity-mode disable
   tsm pending-changes apply
   ```

   注: 上記のコマンドを実行すると、専用の **[アイデンティティ移行]** ページが削除され、アクセスできなくなります。ページにアクセスできるのは、`tsm authentication legacy-identity-mode` が有効になっている場合のみです。

Tableau Server をアイデンティティサービスを使用するように構成した後、ユーザーが Tableau Server にサインインすると、Tableau Server は構成されたアイデンティティストア内の識別子を使用してユーザー アイデンティティを検索します。識別子からユニバーサル一意識別子 (UUID) が返され、既存の Tableau Server ユーザー アイデンティティと照合するために使用されます。次に、このプロセスはユーザーのセッションを生成し、認証ワークフローを完了します。

ステップ 4: アイデンティティプールが有効になっていることを確認する

1. 次のコマンドを実行します。

   ```
   tsm configuration set -k features.IdentityPools -v true
   tsm pending-changes apply
   ```

注: これらのコマンドを実行すると、Tableau Server が再起動されます。

[アイデンティティの移行] ページの “予期しないエラー”

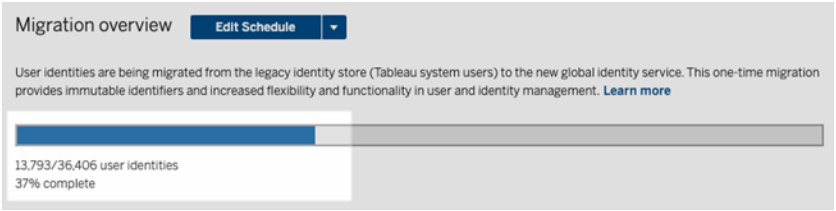
[アイデンティティの移行] ページからすべてのユーザー アイデンティティを解決または確認すると、“予期しないエラー” メッセージが表示されます。このメッセージは、一度に 1,000 を超えるユーザー アイデンティティを解決または確認しようとしたときに表示されることがあります。

この問題を解決するには、1,000 以下のユーザー アイデンティティを選択して解決または確認してから、もう一度やり直してください。

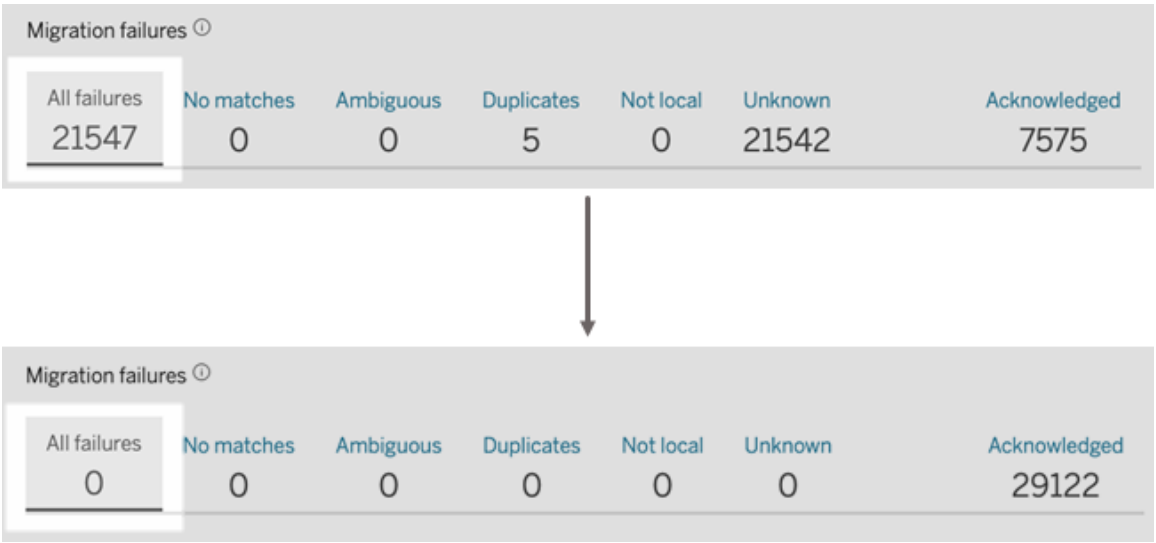
アイデンティティの競合の管理の詳細については、「アイデンティティ移行時の競合の解決」を参照してください。

移行の進行状況が応答しないか動いていない

移行ステータスまたは移行進行状況バーが応答しないか動いていない場合は、**[Migration failures (移行の失敗)]** ですべてのユーザーの競合を解決して承認していることを確認します。



この問題を解決するには、**[すべての失敗]** タブで 1 つ以上のユーザー アイデンティティを選択し、[アクション] ドロップダウン メニューから **[承認]** をクリックして、すべての競合が解決され、承認されている状態にします。**[すべての失敗]** タブに「0」が表示されるまで、このタスクを実行します。アイデンティティの競合の管理の詳細については、「アイデンティティ移行時の競合の解決」を参照してください。



注: すべてのアイデンティティの競合を解決または対処したら、Tableau Server のアイデンティティサービスを有効にする前に、すべての移行ジョブを実行する必要があります。移行ジョブを今すぐ実

行するには、[移行の概要] 見出しの横にある [スケジュールの編集] ドロップダウン矢印をクリックし、**[今すぐ実行]** を選択します。[移行の概要] に **[100% 完了]** と表示されたら、アイデンティティサービスを使用するように Tableau Server を構成できます。詳細については、「ステップ 3: アイデンティティの移行を開始する」を参照してください。

「アイデンティティの移行が進行中です」というポップアップが表示され続ける

アイデンティティサービスがまだ有効になっていない場合、アイデンティティの移行が完了したにもかかわらず、「アイデンティティの移行が進行中」という通知が表示され続けます。アイデンティティの移行を完了するには、アイデンティティサービスを有効にして、Tableau Server がアイデンティティプール機能を有効にするためのアイデンティティ構造を使用できるようにする必要があります。詳細については、ステップ 4: アイデンティティの移行を完了するを参照してください。

アイデンティティ移行ページが消える

アイデンティティの移行が完了し、Tableau Server がアイデンティティサービスを使用するように構成されている場合、専用の **[アイデンティティ移行]** ページは削除され、アクセスできなくなります。[アイデンティティ移行] ページは、アイデンティティを移行する場合、または `tsm authentication legacy-identity-mode` が有効になっている場合にのみ必要です。

ユーザーがサインインできない

アイデンティティの移行が完了し、アイデンティティサービスが有効になっていても、Tableau Server にサインインできないユーザーがいる場合があります。ほとんどの場合、この問題は、アイデンティティが競合していて、アイデンティティの移行中に後で**確認された**ユーザーに発生します。確認されたユーザー アイデンティティはアイデンティティサービスに移行されず、その後の Active Directory (AD) または LDAP のグループ同期中に無視されます。

これらの確認されたユーザー アイデンティティに関連付けられたユーザーが Tableau Server に再度アクセスする必要がある場合は、ユーザーを Tableau Server に手動で追加します。ユーザーを手動で追加すると、後続の AD または LDAP のグループ同期ではユーザー アイデンティティが認識され、期待どおりに同期されます。

アイデンティティの移行を元に戻す

特定のユーザーが Tableau Server にサインインできないなど、アイデンティティサービスが原因であると思われる問題がある場合は、`tsm authentication legacy-identity-mode` コマンドを使用して、従来のアイデンティティストア モードを使用するように戻すことができます。元に戻した後は、アイデンティティの移行後に追加された新しいユーザーと、移行前に Tableau Server にのみサインインできたユーザーの両方が、問題なく Tableau Server にサインインできます。

アイデンティティサービスから従来のアイデンティティストア モードに戻したら、[アイデンティティの移行] ページを使用して、問題のあるユーザー アイデンティティの移行を実行できます。アイデンティティの競合の管理の詳細については、「アイデンティティ移行時の競合の解決」を参照してください。

アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証

Tableau Server バージョン 2023.1 で導入された ID プールは、プロビジョニングと認証情報を使用して、ユーザーが Tableau Server へアクセスできるようにするための ID 管理ツールです。ID プールにより、より集中的で柔軟な ID 管理ワークフローが ID サービス上に構築され、Tableau Server でユーザー ID を保存し、管理することが可能になります。

ID プールは、Tableau Server のセットアップ時に Tableau サービス マネージャー (TSM) を使用して作成したユーザーのプロビジョニングや認証設定を置き換えるものではありません。代わりに、ID プールは、ユーザーのプロビジョニングや認証のオプションを追加することを補完およびサポートするものです。オプションの追加は、特に TSM が Active Directory (AD) または Lightweight Directory Access Protocol (LDAP) で構成されている組織で、必要になる可能性があります。ID プールを使用して別の方法を追加すると、Tableau Server のセットアップ後に、Tableau Server 管理者がユーザー (多くの場合、外部ユーザー、パートナー、または請負業者) を Tableau Server 展開に追加することができます。

ID プールは、次のユース ケース向けに最適化されています。

- **外部ユーザー:** 内部 AD に外部ユーザーを追加したくない大企業組織。

  たとえば、組織に正社員と契約社員の 2 種類の従業員がいるとします。正社員は、Active Directory (AD) でプロビジョニングされ、IdP の Okta で管理される SAML 認証を使用します。契約社員は、グループ メンバーシップが一時的に割り当てられるユーザーか、ユーザーを AD の外部でプロビジョニングして独立して認証する、別の組織のユーザーで構成されています。ID プールを使用すると、AD の外部にある Tableau Server ユーザーを追加できます。

- **複数の ID ストア:** 複数の ID ストアからユーザーを調達する SaaS アプリケーションをホストする組織。

  たとえば、組織では 1 つの Tableau サイトのコンテンツを、複数の外部組織と共有しているとします。これらのユーザーを、ローカル ID ストアで設定した別々の ID プールで分離すると、各組織のユーザーをより簡単に識別および管理できます。

- **内部組織間のセキュリティ境界:** 別個のセキュリティ境界を持つ、複数の買収された下位組織からなる組織。

  たとえば、新しく追加された組織のユーザーをローカル ID ストアで構成された ID プールに追加すると、ID ストアの結合に伴う複雑さを回避できます。

ID プールとは

ID プールを構成する 3 つの主要コンポーネントは、ユーザーをプロビジョニングするための ID ストア、OpenID Connect (OIDC) 認証、割り当てられたユーザーです。

- **ID ストア:** ユーザーを調達またはプロビジョニングする ID ストア には、ローカル ID ストアと外部 ID ストアがあります。

  - ローカル アイデンティティ ストアの場合、既存または新規のローカル アイデンティティ ストアを使用するようにアイデンティティ プールを構成できます。**注:** ローカル認証はサポートされていません。

- 外部アイデンティティストアの場合、アイデンティティプールで使用できるのは、Tableau Server のセットアップ中に TSM で構成したものと同じ外部アイデンティティストア (AD または LDAP) のみです。別の外部 ID ストアを使用するように ID プールを設定することはできません。

Tableau Server のセットアップ時に TSM で設定したプロビジョニングと認証の構成は、デフォルトまたは「初期プール (TSM 構成)」と呼ばれます。

- **認証:** ID プールでサポートされている唯一の認証方法は、OIDC です。
- **ユーザー:** ユーザーが Tableau Server にサインインするには、初期プール (TSM 構成) から供給されているか、少なくとも 1 つの ID プールのメンバーである必要があります。

ID プールを使用する場合

Tableau Server 管理者は ID プールを使用して、ユーザーのプロビジョニング元と Tableau Server へのユーザーの認証方法に基づいて、ユーザーを ID コホートにセグメント化できます。Tableau Server のセットアップ時に TSM で作成した ID ストアと認証の構成 (初期プール (TSM 構成) とも呼ばれます) は変更されませんが、ID プールは Tableau Server から設定可能です。

注: アイデンティティプールは現在、サーバーレベルの構成でのみ使用できます。アイデンティティプールの範囲をサイトに限定することはできません。

ID プールの詳細

初期プール (TSM 構成) と ID プール

上記のように、Tableau Server のセットアップ時に TSM で行うプロビジョニングと認証の構成の組み合わせは、「初期プール (TSM 構成)」と呼ばれます。初期プール (TSM 構成) は、Tableau Server セットアッププロセスの必須コンポーネントであり、変更できません。

ただし、ID プールはオプションであり、必要な数の ID プールを Tableau Server から直接作成できます。

ID プールがユーザーのサインイン エクスペリエンスに与える影響

デフォルトでは、Tableau Server 用に ID プールが作成されていない場合、ユーザーが Tableau Server ランディング ページに移動して Tableau Server にサインインする方法に変更はありません。

少なくとも 1 つの ID プールを作成すると、Tableau Server ランディング ページに複数のサインイン オプションが表示されます。プライマリサインイン オプションはページの上部に表示され、初期プール (TSM 構成) に属するユーザーがサインインできます。

プライマリサインイン オプションの下には、セカンダリサインイン オプションが表示されます。各オプションは各 ID プールを表し、作成した順序で表示されます。これらのプールに割り当てられたユーザーは、所属する ID プールのオプションを使用してサインインする必要があります。ユーザーを正しいサインイン オプションに導くには、作成時に ID プールに説明を追加することを検討してください。

注: どのプールに属しているかに関係なく、すべてのユーザーに Tableau Server に設定されたすべてのプールが表示されます。

Tableau におけるユーザー名と識別子

ユーザー名は、システム ユーザーを表す情報です。識別子は、ユーザー名の情報を補足するために使用され、外部のアイデンティティストアでユーザー名の代わりとして使用できます。

Tableau では、ユーザー名は Tableau へのサインインに使用される不変の値であり、識別子はユーザーをユーザー名に一致させる方法として Tableau のアイデンティティ構造で使用される可変の値です。識別子はユーザー名から逸脱することができるため、Tableau がより柔軟に対応できるようになります。外部アイデンティティストアのユーザー名が変更された場合、Tableau Server 管理者は識別子を更新して、ユーザーを正しいユーザー名に一致させることができます。

既存のユーザーをアイデンティティプールに追加するときは、識別子を設定する機能があることを想定してください。たとえば、既存のユーザーがローカル アイデンティティストアを使用して構成されたアイデンティティプールに属しており、AD アイデンティティストアを使用して構成されたアイデンティティプールにそのユーザーを追加する場合は、そのユーザーに関連付けられた識別子を検索するためにユーザー名を入力するよう求められます。一方、既存のユーザーが AD アイデンティティストアを使用して構成されたアイデンティティプールに属しており、ローカル アイデンティティストアを

使用して構成されたアイデンティティプールにそのユーザーを追加する場合は、オプションの識別子を入力するよう求められます。例外は、ローカル アイデンティティストアとローカル認証を使用して構成された初期プール (TSM で構成済み) にユーザーを追加する場合です。そのユーザーの識別子を設定することはできません。

ID プールの設定と管理

ID プールを作成および管理するには、Tableau REST OpenAPI を使用して、[ID プールのメソッド] に対してプログラム的に呼び出しを行う必要があります。ID プールのユーザーを追加または管理するには、Tableau Server ユーザー インターフェイス (UI) を直接、または Tableau REST API を介して使用できます。

以下の手順は、ID プールを設定するプロセスを要約したものです。

1. **Tableau Server を設定してセッションを確立します**。
2. 新しいローカル アイデンティティストア インスタンスを設定して、**ユーザーをプロビジョニングします**。**注:** この手順をスキップして、既存のローカル アイデンティティストアを使用するか、Tableau Server のセットアップ時に TSM で構成した外部アイデンティティストアを使用することができます。
3. **認証を設定する**。OpenID Connect (OIDC) を使用して、Tableau Server に対するユーザーを認証します。
4. **ID プールを作成する**。この ID プールは、設定した ID ストアと OIDC 認証を使用します。
5. **ユーザーを ID プールに追加する**。Tableau Server の UI または REST API を使用して、ユーザーが Tableau Server にサインインできるようにします。

セットアップ後、ID プールのテスト、管理、トラブルシューティングを行うことができます。

注: Salesforce 開発者の Postman ワークスペースのアイデンティティプール Postman コレクションを使用して、このトピックで説明されているメソッドについて学習、開発、テストできます。

前提条件

ID プールを使い始める前に、次の要件を満たす必要があります。

- Okta などの OIDC ID プロバイダー (IdP) との統合がすでに設定されている
- Tableau Server 2023.1 以降を実行している

- アイデンティティの移行を実行した - これは、新しい Tableau Server の展開と Tableau Server のアップグレードの両方で実行する必要があります。

はじめに

ステップ 1: Tableau Server を設定してセッションを確立する

ID プールのセットアップに関連する変更を有効にするには、1 回限りの TSM 設定と、セッション変数とホスト変数の宣言が必要です。

1. クラスターの初期ノード (TSM がインストールされているノード) で、管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。
   a. `tsm configuration set -k gateway.external_url -v http://<host>`
   b. `tsm pending-changes apply`

   たとえば、Tableau Server http://myco を設定するには、次のコマンドを実行します。

   ```
   tsm configuration set -k gateway.external_url -v http://myco
   tsm pending-changes apply
   ```

   詳細については、gateway.external_url を参照してください。

   注: バージョン 2025.3 以降、指定された URL (有効な場合) が、シングル ログアウト (SLO) に使用されるエンドポイントになります。
3. (オプション) 初期プール (TSM 構成) の説明を追加するには、次のコマンドを実行します。
   a. `tsm configuration set -k wgserver.authentication.identity_pools.default_pool_description -v "<description>"`
   b. `tsm pending-changes apply`

   たとえば、「MyCo 従業員のサインイン」の説明を追加するには、次のコマンドを実行します。

```
tsm configuration set -k wgserver.authentication.identity_
pools.default_pool_description -v "Sign-in for MyCo employees"
tsm pending-changes apply
```

詳細については、wgserver.authentication.identity_pools.default_pool_description を参照してください。

4. (オプション) シングル ログアウト (SLO) の有効化を完了し、初期プール (TSM 構成) を設定するには、次のコマンドを実行します。

```
tsm authentication openid enable-slo
tsm authentication openid get-slo-redirect-url
```

5. Tableau Server に管理者としてサインインし、以下を実行します。

   a. ブラウザーの開発者ツールに移動し、アプリケーションのクッキーに移動します。
   b. **workgroup_session_id** の値に注意が必要です。

   たとえば、Chrome で作業している場合、Tableau Server ホームページの任意の場所で右クリックし、[**検査**] を右クリックして選択します。上部のナビゲーション ペインから [**アプリケーション**] をクリックし、左側のナビゲーション ペインから [**クッキー**] をクリックします。[クッキー] で、http://myco.com などの Tableau Server 名をクリックし、中央のペインにある [**workgroup_session_id**] の値を確認します。

6. Tableau REST OpenAPI を使用して ID プール要求を作成するために使用しているスクリプトまたは API 開発者ツールで、次の操作を行います。

   a. workgroup_session_id の値をグローバル変数として追加します。
   b. さらに、ポート 80、ホスト (Tableau Server の URL)、プロトコル (HTTP または HTTPS) をグローバル変数に追加します。

   たとえば、次の表は、Tableau Server http://myco に必要なグローバル変数を示しています。

グローバル変数	値
ワークグループセッション ID	XXX_2abcDefDwGVzPu1hCQ|FJk5Z6OroPCLEDTKkwDxaeA0YzrIY04f|ca60-8d3c-fc01-4e40-ae5e-9b2131e4e7xx
ポート	80
ホスト	http://myco
プロトコル	HTTP

ステップ 2: ID ストアを設定する

Tableau Server では、Tableau Server ユーザーを調達またはプロビジョニングするために ID ストアを構成する必要があります。

アイデンティティプールを設定するときは、新規または既存のローカル アイデンティティストアを使用できます。または、Tableau Server のセットアップ時に外部アイデンティティストアが構成された場合は、Active Directory (AD) または Lightweight Directory Access Protocol (LDAP) のいずれかを外部 アイデンティティストアとして使用できます。

注: Tableau Server のセットアップ時に TSM で構成した AD インスタンスまたは LDAP インスタンス (初期プール (TSM 構成) とも呼ばれます) ではない新しい AD インスタンスや LDAP インスタンスは、アイデンティティプールで設定できません。

新しいローカル アイデンティティストアを設定するには、次の手順を実行します。既存のローカル アイデンティティストアを使用するか、Tableau Server のセットアップ時に構成したアイデンティティストアを使用する場合は、「ステップ 3: 認証を設定する」に進んでください。

1. Tableau REST API に対してサインイン要求を行い、認証資格情報 トークンを生成します。

   例

   URI

   ```
   POST https://myco/api/3.19/auth/signin
   ```

2. 認証資格情報 トークンが生成されたら、後続のすべての API 要求のヘッダーにこの認証資格情報 トークンを追加します。

3. Tableau REST API OpenAPI を使用して [ID ストアの構成] エンドポイントを呼び出し、ID ストアを設定します。

4. 要求では、以下を指定します。

   a. タイプ。ローカル アイデンティティストア タイプのタイプ値は常に 0 です。既存のローカル アイデンティティストアを使用するか、Tableau Server のセットアップ時に TSM で構成したアイデンティティストアを使用する場合は、新しいローカル アイデンティティストア インスタンスを設定する必要はありません。代わりに、以下の「ステップ 3: 認証を設定する」に進んでください。
   b. 名前。名前は一意である必要があります。
   c. 表示名。これはオプションです。

   例

   URI

   https://myco/api/-/authn-service/identity-stores/

   要求本文 (JSON)

   ```
   {
   "type": "0",
   "name": "Local identity store #1",
   ```

```
"display_name": "Local identity store #1"
}
```

レスポンス本文

なし

ステップ 3: 認証を設定する

OpenID Connect (OIDC) 認証方式を設定して、ユーザーを認証できるようにします。

注: アイデンティティプールで使用するアイデンティティストアタイプに関係なく、現在、アイデンティティプールで構成できる認証方式は OIDC のみです。

1. ID ストアを設定したら、Tableau REST API OpenAPI を使用して [認証設定の作成] エンドポイントを呼び出します。
2. 要求では、以下を指定します。
   a. 認証タイプ。認証タイプの値は「`OIDC`」です。
   b. iFrame。iFrame の既定値は「`false`」です。
   c. 要求された OIDC クライアント ID、クライアントシークレット、構成 URL、ID クレーム、クライアント認証、ユーザー名クレーム。
      - クライアント ID とクライアントシークレットは、OIDC IdP によって提供されます。
      - 構成 URL も IdP によって提供されます。通常、URL には次の形式を使用できます。`https://<idp_url>/.well-known/openid-configuration`
      - ID クレームの既定値は「`sub`」です。詳細については、「sub クレームを変更する」を参照してください。
      - クライアント認証の既定値は「`CLIENT_SECRET_BASIC`」です。
      - ユーザー名クレームの既定値は「`email`」です。詳細については、「既定: メールクレームを使用してユーザーをマッピングする」を参照してください。

      ユーザー名クレームについて

Tableau では、ID 照合の目的でユーザー名クレームを使用します。ユーザーを Tableau Server に追加するときに識別子を指定すると、その識別子はユーザー名クレームで提供された値と一致するように使用されます。識別子が指定されていない場合は、Tableau Server に設定されているユーザー名をデフォルトとして使用します。

注:

- AD を ID ストアとして使用する ID プールでこの認証設定を使用する場合は、割り当てられたユーザーのユーザー名クレームに AD の sAMAccountName 値があることを確認してください。
- LDAP を ID ストアとして使用する ID プールでこの認証設定を使用する場合は、割り当てられたユーザーのユーザー名クレームに LDAP のユーザー名の値があることを確認してください。

例

URI

https://myco/api/-/authn-service/auth-configurations/

要求本文 (JSON)

```
{
"auth_type": "OIDC",
"iframed_idp_enabled": true,
"oidc": {
       "client_id": "Xxx1hotzhjv4tyCd08",
       "client_secret": "XxXd2NCxY-BiLu_zcIwr2lJZLziT_7sw9Fi6XX3",
       "config_url": "https://admin.okta.com/.well-known/openid-
configuration",
       "custom_scope": "",
       "id_claim": "sub",
       "username_claim": "email",
       "client_authentication": "CLIENT_SECRET_BASIC",
       "essential_acr_values": "",
       "voluntary_acr_values": "",
       "prompt": "login,consent",
```

```
        "connection_timeout": 100,
        "read_timeout": 100,
        "ignore_domain": false,
        "ignore_jwk": false
        }
}
```

レスポンス本文

なし

ステップ 4: ID プールを作成する

Tableau Server のセットアップ時に構成したアイデンティティストアに応じて、作成するアイデンティティプールは、次のアイデンティティストアと認証方式の組み合わせのいずれかを持つことになります。

- AD アイデンティティストア + OIDC 認証
- LDAP アイデンティティストア + OIDC 認証
- ローカル アイデンティティストア + OIDC 認証

最初の 2 つの組み合わせでは、初期プール (TSM 構成) が AD または LDAP を使用するように構成されている必要があります。

以下で説明する手順では、最後の組み合わせ「ローカル アイデンティティストア + OIDC 認証」を使用してアイデンティティプールを作成します。

1. OIDC 認証を設定したら、Tableau REST API OpenAPI を使用して [ID プールの作成] エンドポイントを呼び出します。
2. 要求では、以下を指定します。
    a. アイデンティティプールの名前と説明。ID プールの名前と説明の両方が、Tableau Server のランディング ページですべてのユーザーに表示されます。
    b. アイデンティティストアのインスタンス ID と認証タイプのインスタンス ID。

注:

- アイデンティティストアのインスタンス ID と認証タイプのインスタンス ID を取得するには、「アイデンティティストアの一覧表示」と「認証構成の一覧表示」のエンドポイントを呼び出します。
- Tableau Server のセットアップ時に TSM で構成したアイデンティティストアを使用するアイデンティティプールを作成する場合、アイデンティティストアのインスタンスの値は常に `'1'` になります。

例

URI

https://myco/api/-/authn-service/identity-pools/

要求本文 (JSON)

```
{
"name": "MyCo contractors",
"identity_store_instance": "2",
"auth_type_instance": "0",
"is_enabled": true,
"description": "Sign-in for MyCo contractors"
}
```

レスポンス本文の例

なし

3. アイデンティティプールを作成したら、IdP 構成に移動し、サインイン リダイレクト URI を `http://<host>/authn-service/authenticate/oidc/<identity_pool_id>/login.` に設定します。

   たとえば、`http://myco/authn-service/authenticate/oidc/XXtgfe21-74d2-3h78-bdg6-g2g6h1234567/login`

   注: ID プールの ID を取得するには、[ID プールの一覧表示] エンドポイントを呼び出します。

注:

- 組織が必要とする数の ID プールを作成できます。
- その他のタイプの ID ストアと認証方式が初期プール (TSM 構成) でサポートされています。詳細については、「認証」を参照してください。

ステップ 5: 初期プールにユーザーを追加する

Tableau Server を直接使用して、ユーザーを ID プールに追加できます。Tableau Server にサインインするには、ユーザーが初期プール (TSM 構成) に属しているか、ID プールに追加されている必要があります。ユーザーを ID プールに追加すると、ID プールで設定された ID ストアに応じてワークフローが変わる可能性があります。

以下の手順では、Tableau Server UI を介してユーザーを ID に追加する方法について説明します。ただし、Tableau REST API を使用して [ユーザーを ID プールに追加する] エンドポイントを呼び出すことにより、ID プールにユーザーを追加することができます。

1. Tableau Server に管理者としてサインインします。
2. 左側のナビゲーション ペインから、**[ユーザー]** (複数サイトの Tableau Server の場合は **[すべてのサイト] > [ユーザー]**) を選択します。
3. **[ユーザーを追加]** ボタンをクリックし、**[新しいユーザーを作成]** または **[ファイルからユーザーをインポートする]** を選択します。

   新しいユーザーを作成する場合

   a. 新しいユーザーを追加する ID プールを選択し、[次へ] をクリックします。

      i. AD または LDAP の ID ストアで設定された ID プールを選択した場合は、ユーザー名を入力し、サイトのメンバーシップとサイト ロールを割り当てます。完了したら、[ユーザーをインポート] ボタンをクリックします。

      ii. ローカル ID ストアで設定された ID プールを選択した場合は、ユーザー名を入力します。ダイアログ ボックスが開き、表示名、識別子 (ほとんどの場合)、メール アドレスを追加し、サイトとサイト ロールを設定できます。完了したら、**[ユーザーを作成]** ボタンをクリックします。

ユーザー名、およびサイトのメンバーシップとサイトロールを割り当てる方法の詳細については、「ユーザーのサイトロールの設定」を参照してください。

Tableau でのユーザー名と識別子について

ユーザー名は、システム ユーザーを表す情報です。識別子は、ユーザー名の情報を補足するために使用され、外部のアイデンティティストアでユーザー名の代わりとして使用できます。

Tableau では、ユーザー名は Tableau へのサインインに使用される不変の値であり、識別子はユーザーをユーザー名に一致させる方法として Tableau のアイデンティティ構造で使用される可変の値です。識別子はユーザー名から逸脱することができるため、Tableau がより柔軟に対応できるようになります。外部アイデンティティストアのユーザー名が変更された場合、Tableau Server 管理者は識別子を更新して、ユーザーを正しいユーザー名に一致させることができます。

既存のユーザーをアイデンティティプールに追加するときは、識別子を設定する機能があることを想定してください。たとえば、既存のユーザーがローカル アイデンティティストアを使用して構成されたアイデンティティプールに属しており、AD アイデンティティストアを使用して構成されたアイデンティティプールにそのユーザーを追加する場合は、そのユーザーに関連付けられた識別子を検索するためにユーザー名を入力するよう求められます。一方、既存のユーザーが AD アイデンティティストアを使用して構成されたアイデンティティプールに属しており、ローカル アイデンティティストアを使用して構成されたアイデンティティプールにそのユーザーを追加する場合は、オプションの識別子を入力するよう求められます。例外は、ローカル アイデンティティストアとローカル認証を使用して構成された初期プール (TSM で構成済み) にユーザーを追加する場合です。そのユーザーの識別子を設定することはできません。

ファイルからユーザーをインポートする場合

a. 次の列をこのリスト順に含む .csv ファイルをアップロードします。

```
username, password, display name, license level, admin
level, publishing capability, email address, identity pool
name, identifier
```

注: 必須の列はユーザー名とパスワードのみです。ただし、ID プール名を指定しない場合、ユーザーは初期プール (TSM 構成) に追加されます。詳細については、「CSV インポートファイルのガイドライン」を参照してください。

たとえば、「Henry Wilson」と「Fred Suzuki」を「一般委託先」ID プールに追加するとします。.csv には次の値が含まれるでしょう。

```
henryw,henrypassword,Henry
Wilson,Viewer,None,yes,hwilson@myco.com,General
Contractors,hwilson
freds,fredpassword,Fred
Suzuki,Creator,None,no,fsuzuki@myco.com,General
Contractors,fsuzuki
```

注: ID プールを 1 つ以上作成すると、Tableau Server のランディングページが更新され、それらの ID プールに含まれるユーザー用のサインインオプションが追加されます。詳細については、「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。

ID プールのテスト

ID プールを設定したら、Tableau Server からサインアウトし、ID プールに属するユーザーとして再度サインインしてテストすることをお勧めします。サインインプロセスを実行して、OIDC 認証が正しく設定されていることを確認してください。

注: 「ステップ 1: Tableau Server を設定してセッションを確立する」で初期プール (TSM 構成) のオプションの説明を設定した場合、または Tableau Server のサーバー設定 (全般とカスタマイズ)に関する注記がある場合は、その説明は初期プール (TSM 構成) を使用してサインインするユーザーに固有のものとし、サインインのカスタマイズに関する注記は Tableau Server にサインインするすべてのユーザーに適用するものとすることをお勧めします。

ID プールの管理

ID プールのユーザーは、サーバー レベルとサイト レベルの両方の [ユーザー] ページから管理できます。[ユーザー] ページでは、ユーザーが属している ID プールと、ID プールに関する概要の内容を確認できます。

認証設定や ID プールの更新、ローカルの ID ストアや ID プールの削除など、他のすべての ID プール管理タスクについては、「ID プールのメソッド」で説明されている Tableau REST API OpenAPI を使用します。

ID プールのトラブルシューティング

ID プールの制限

ID プールは Tableau Server でのみ使用できます。

注: アイデンティティプールは現在、サーバー レベルの構成でのみ使用できます。アイデンティティプールの範囲をサイトに限定することはできません。

Tableau Server のランディング ページに IdP エラーが表示される

Tableau Server ランディング ページのプライマリサインイン オプションで、ID プールのサインイン オプションの横に IdP 関連のエラー メッセージが表示される場合があります。これは OIDC 認証関連の問題であり、次のいずれかまたは両方が当てはまる場合に発生する可能性があります。1) Tableau Server が外部 URL を IdP に送信するように構成されていない。2) グローバル変数が宣言されていない。

この問題を解決するには、上記の「ステップ 1: Tableau Server を設定してセッションを確立する」で説明されている手順を完了していることを確認してください。

Tableau Server のランディング ページにアイデンティティプールが表示されない

アイデンティティプールの機能が無効になっている場合は、次の TSM コマンドを使用して再度有効にすることができます。

```
tsm configuration set -k features.IdentityPools -v true
```

```
tsm configuration set -k features.NewIdentityMode -v true

tsm configuration set -k wgserver.authentication.legacy_identity_
mode.enabled -v false

tsm pending-changes apply
```

注: これらのコマンドを実行すると、Tableau Server が再起動されます。

Tableau Server へのユーザーの追加

Tableau Server へのユーザーの追加は、1 度または 1 バッチに 1 回のみ行なえます。それらのユーザーは、ライセンスのないユーザーとしてサーバーに追加し、その後サイトに追加し、Tableau Server にオンボードさせるときにサイトロールを割り当てることができます。または、サイトへのユーザー追加とサイトロール指定を同時に行うことができ、その時点でユーザーはサインインする準備が整います。

はじめる前に

- Tableau Server で Active Directory 認証が構成されている場合、ユーザーを追加する前に、外部 ID ストアを使用した展開におけるユーザー管理を見直し、複数のドメイン、ドメイン命名、NetBIOS、Active Directory ユーザー名の形式が Tableau ユーザー管理にどのように影響するかについて理解してください。

  注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

- お使いのユーザーベースに十分なユーザーライセンスとロールライセンスがあることを確認します。ライセンスがないサイトロールにユーザーを追加しようとすると、エラーが表示されます。
- Tableau Server の展開にユーザーライセンスを追加するには、Tableau カスタマーポータルにアクセスして、ライセンスと対応するプロダクトキーを購入します。ライセンスを購入した後は、Tableau Server へのライセンスの追加を参照して、新しいキーでサーバーを更新してください。

- このトピックの手順は、個々のユーザーを追加し、サイトロールを割り当てる方法について示しています。ユーザーをバッチで追加するには、ユーザーのインポートを参照してください。

ユーザーをサイトレベルではなくサーバーレベルで追加する場合

シングルサイト環境では、サーバー管理者は **[ユーザー]** ページでユーザーを追加できます。

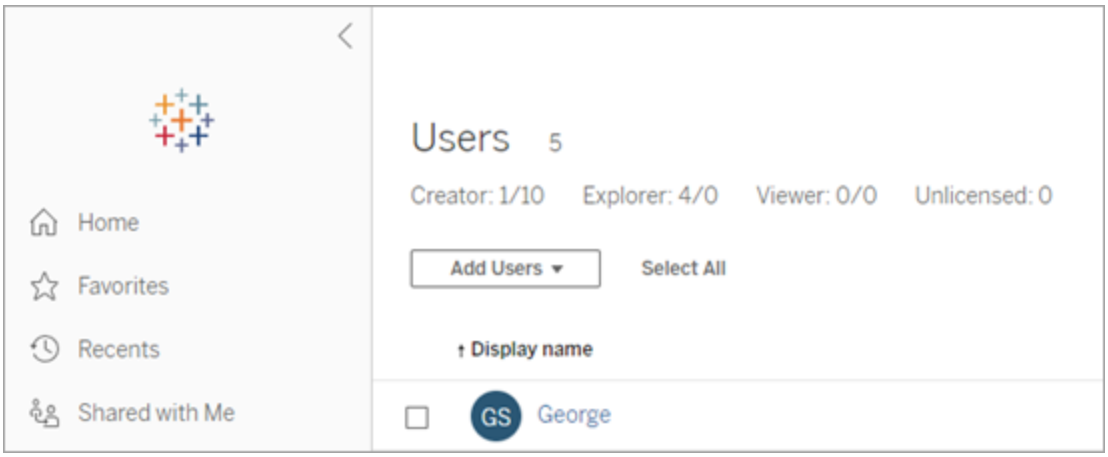
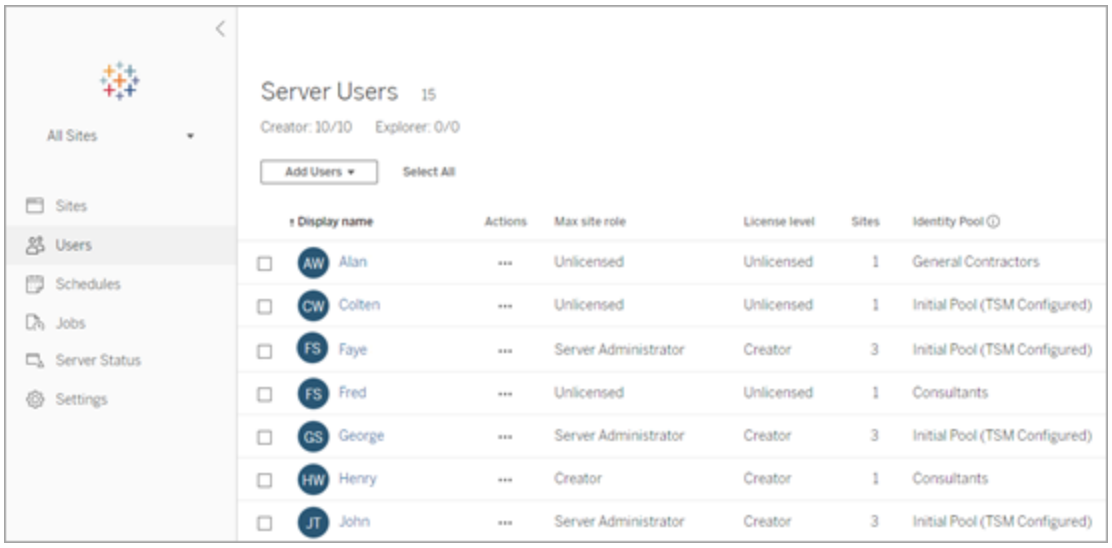


Tableau Server にサイトを追加すると、**[サーバー ユーザー]** ページ (各サイトのすべてのサーバーユーザーがここに表示される) および **[サイト ユーザー]** ページのあるマルチサイトサーバーとなります。サーバー管理者のみが **[サーバーユーザー]** ページにアクセスでき、サイト管理者とサーバー管理者の両方が **[サイト ユーザー]** ページにアクセスできます。

[サーバー ユーザー] ページは、ユーザーをマルチサイトに割り当て、サーバーからユーザーを削除し、サーバーがローカル認証を使用している場合はユーザーパスワードをリセットできる唯一の場所です。



サーバーへのユーザーの追加

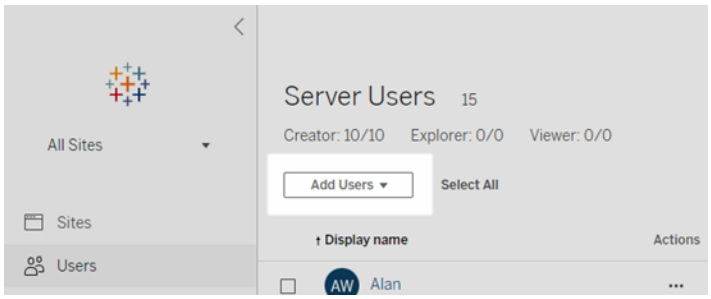
ユーザーを追加するとき、少なくとも 1 つの ID プールが Tableau Server 用に設定されているかどうかによって、ワークフローが変わります。

ID プールが設定されていない場合

ID プールが設定されていない場合、つまり、Tableau Server が Tableau Server のセットアップ時に設定されたユーザープロビジョニングと認証 (初期プール (TSM 構成) とも呼ばれます) のみを使用している場合、以下に説明する手順に従ってユーザーを追加します。

1. サイト メニューで、**[すべてのサイトを管理]**、**[ユーザー]**、**[ユーザーの追加]** の順にクリックします。

   サイトにユーザーを追加するには、サイトを選択して **[ユーザー]** ページに移動します。



2. 次のいずれかを実行します。

   - サーバーで**ローカル認証**が構成されている場合、**[新規ユーザー]** をクリックし、ユーザー名を入力します。ローカル認証では、ユーザー名の競合を回避する最善の方法は、ユーザー名にメール アドレスを提供することです。たとえば、*jsmith* でなく *jsmith@example.com* とします。

     ユーザー名は大文字と小文字を区別しません。ユーザー名で許可されていない文字はセミコロン(;) とコロン(,) です。

   - サーバーで **Active Directory 認証**が構成されている場合、**[Active Directory ユーザー]** をクリックします。Tableau Server を実行している同じ Active Directory ドメインのユーザーを追加する場合は、サーバー ドメインが使用され、ドメインなしで AD ユーザー名を入力できます。

     注: ユーザーの氏名を入力しないでください。インポートプロセス中にエラーが発生する可能性があります。

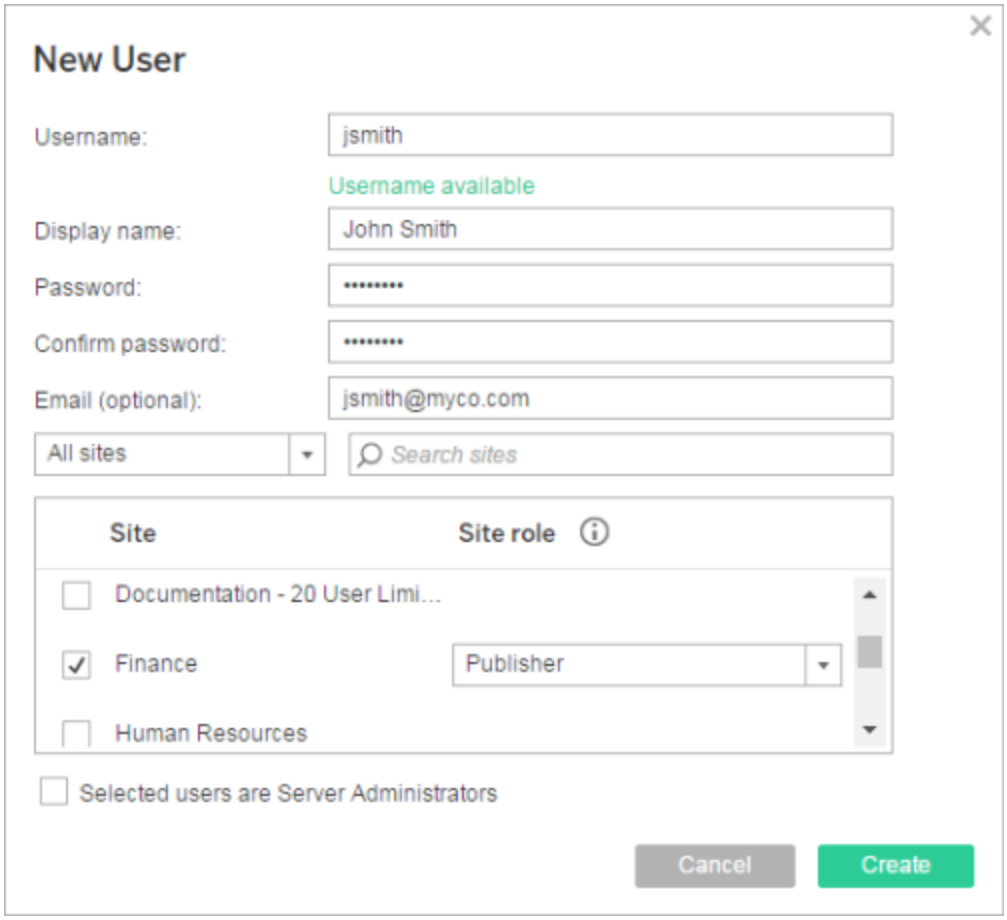
3. ローカル認証を使用しているサーバーでは、次の情報を入力します。

   - **表示名** - ユーザーの表示名を入力します (*John Smith* など)。
   - **パスワード**- ユーザーのパスワードを入力します。
   - **パスワードの確認** - パスワードを再入力します。
   - **メール** - これは任意です。ユーザープロフィール設定から後で追加できます。

- **選択されたユーザーはサーバー管理者**: ユーザーがサーバー管理者である必要があるかどうかを指定します。

- **名前 (サイト メンバーシップ) / サイト ロール**: ユーザーがサーバー管理者でない場合、ユーザーをゼロまたはそれ以上のサイトに割り当て、各サイトでサイト ロールを付与することができます。この時点では、サイト メンバーシップとサイト ロールを選択する必要はありません。新しいサーバー ユーザーのサイト メンバーシップとサイト ロールを指定しない場合、ユーザーは「ライセンスなし」のサイト ロールを持つサーバー ユーザーとしてのみ追加されます。詳細については、ユーザーのサイト ロールの設定を参照してください。

4. **[作成]** をクリックします。

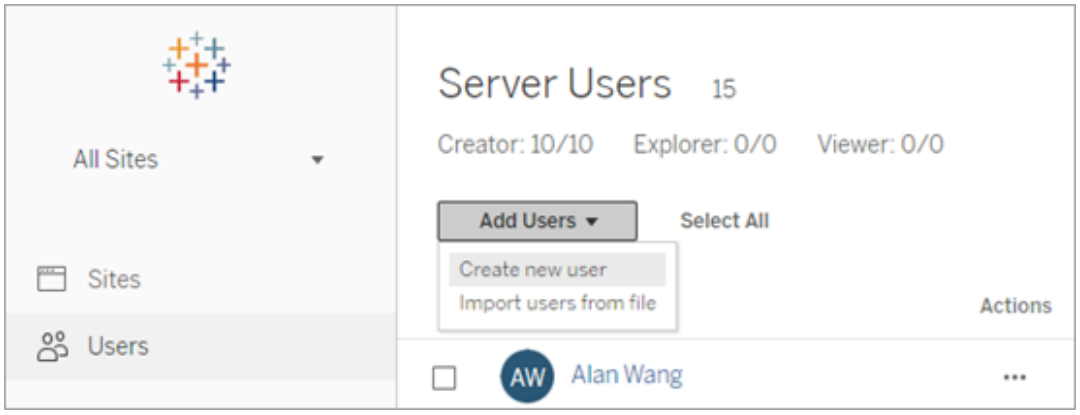


少なくとも 1 つの ID プールが設定されている場合

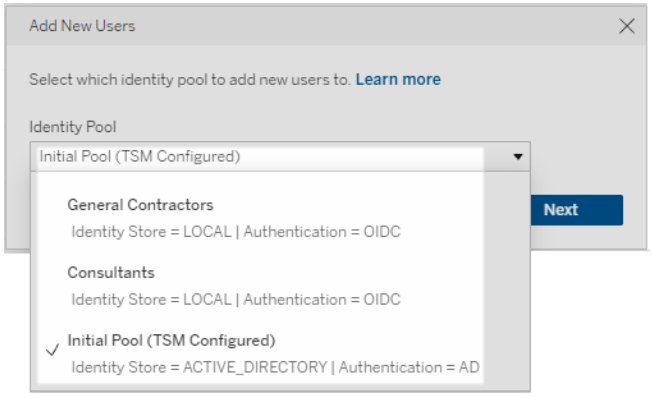
ユーザーを追加する際、少なくとも 1 つの ID プールが設定されている場合は、まず ID プールまたは **初期プール (TSM 構成)** を選択する必要があります。初期プール (TSM 構成) は、Tableau Server のセットアップ時に TSM で設定されたユーザープロビジョニングと認証です。

1. サイト メニューで、**[すべてのサイトの管理]**、**[ユーザー]** の順に選択し、**[ユーザーの追加]**、**[新しいユーザーの作成]** の順にクリックします。

   サイトにユーザーを追加するには、サイトを選択して **[ユーザー]** ページに移動します。



2. [新しいユーザーの追加] ダイアログ ボックスで、ユーザーを追加する ID プールを選択します。

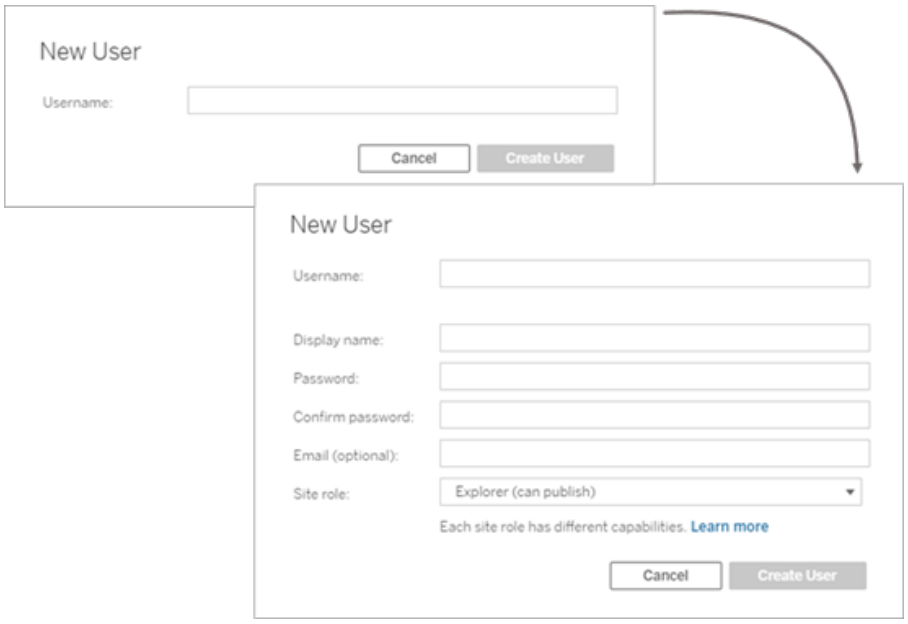


3. 次のいずれかを実行します。

- 手順 2 で選択した ID プールが **LOCAL** ID ストアと**LOCAL** 認証用に設定されている場合:

  a. [新しいユーザー] ダイアログ ボックスの [**ユーザー名**] テキスト ボックスに、ユーザー名を入力します。ローカル認証では、ユーザー名の競合を回避する最善の方法は、ユーザー名にメール アドレスを提供することです。たとえば、*jsmith* でなく*jsmith@example.com* とします。

  ユーザー名は大文字と小文字を区別しません。ユーザー名で許可されていない文字はセミコロン (;) とコロン (,) です。



  b. 以下の情報を入力します。

  - **表示名 -** ユーザーの表示名を入力します (*John Smith* など)。
  - **パスワード -** ユーザーのパスワードを入力します。
  - **パスワードの確認 -** パスワードを再入力します。

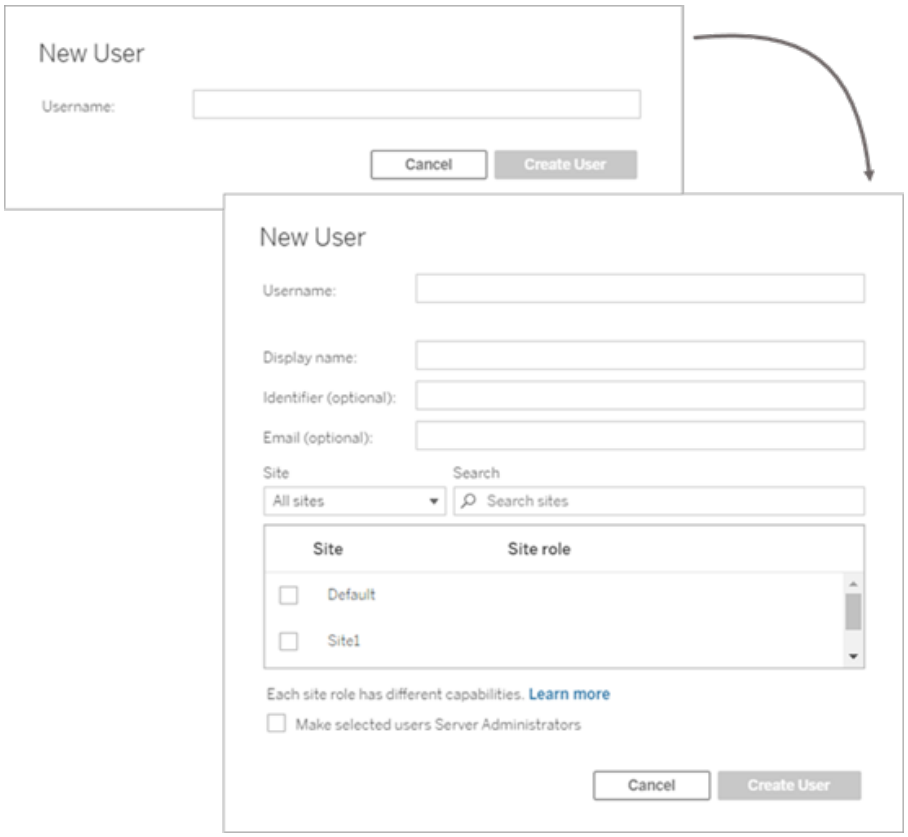
- **メール (オプション) -** これはオプションであり、後でユーザープロフィール設定で追加できます。
- **選択されたユーザーはサーバー管理者**: ユーザーがサーバー管理者である必要があるかどうかを指定します。
- **サイトロール**: ユーザーがサーバー管理者でない場合、サイトロールを割り当てることができます。この時点では、サイトロールを選択する必要はありません。新しいユーザーのサイトロールを指定しない場合、ユーザーはサーバーのユーザーとしてのみ追加され、サイトロールは「ライセンスなし」になります。詳細については、「ユーザーのサイトロールの設定」を参照してください。

c. 完了したら、**[ユーザーの作成]** をクリックします。

- 手順 2 で選択した ID プールが **LOCAL** ID ストアと **OIDC** 認証用に設定されている場合:

a. [新しいユーザー] ダイアログ ボックスの **[ユーザー名]** テキスト ボックスに、ユーザー名を入力します。ローカル認証では、ユーザー名の競合を回避する最善の方法は、ユーザー名にメール アドレスを提供することです。たとえば、*jsmith* でなく *jsmith@example.com* とします。

ユーザー名は大文字と小文字を区別しません。ユーザー名で許可されていない文字はセミコロン (;) とコロン (,) です。



b. 以下の情報を入力します。

- **表示名 -** ユーザーの表示名を入力します (*John Smith* など)。
- **パスワード -** ユーザーのパスワードを入力します。
- **識別子 (オプション) -** ユーザーに関連付ける識別子を入力します。識別子は、ID の照合を目的としています。詳細については、「Tableau におけるユーザー名と識別子」を参照してください。
- **メール (オプション) -** これはオプションであり、後でユーザープロフィール設定で追加できます。
- **サイトとサイト ロール -** ユーザーがサーバー管理者でない場合、ユーザーを 0 個以上のサイトに、各サイトのサイト ロールと共に割り当てる

ことができます。この時点では、サイトメンバーシップとサイトロールを選択する必要はありません。新しいサーバーユーザーのサイトメンバーシップとサイトロールを指定しない場合、ユーザーは「ライセンスなし」のサイトロールを持つサーバーユーザーとしてのみ追加されます。詳細については、ユーザーのサイトロールの設定を参照してください。

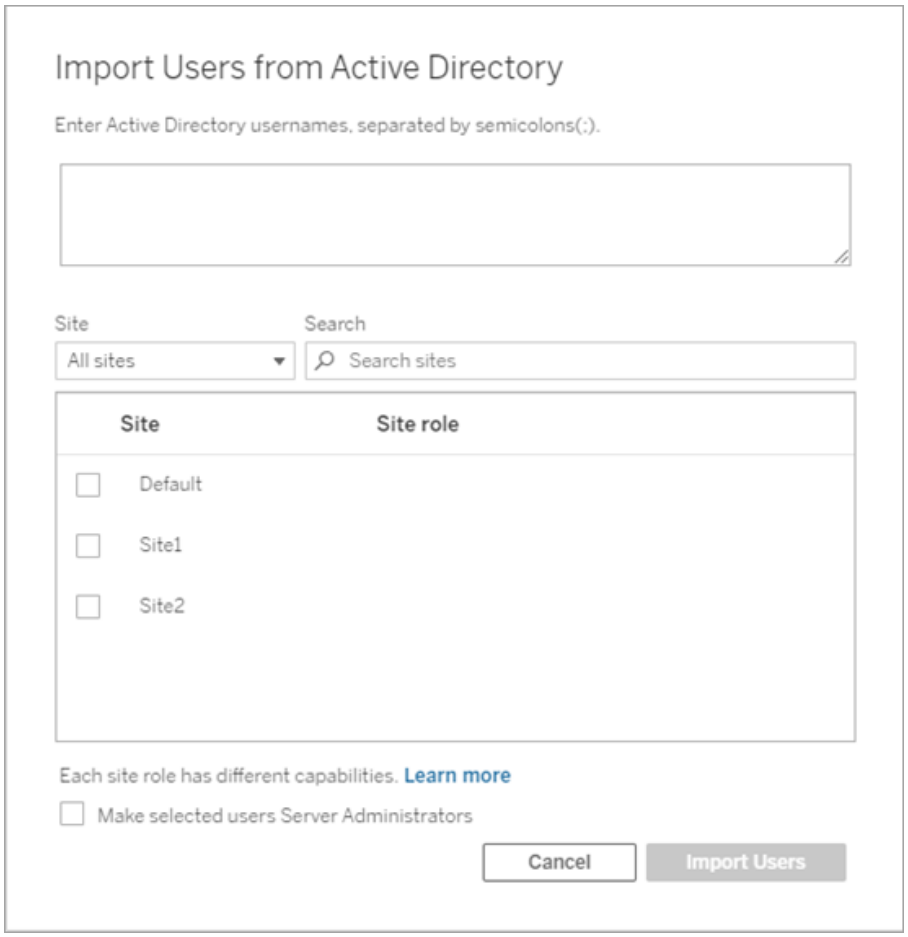
- **選択されたユーザーをサーバー管理者にする -** ユーザーをサーバー管理者にするかどうかを指定します。

c. 完了したら、**[ユーザーの作成]** をクリックします。

- 手順 2 で選択した ID プールが **ACTIVE_DIRECTORY** または **LDAP** の ID ストア用に設定されている場合:

a. AD または LDAP のユーザー名を、ドメインなしで入力します。このワークフローでは、Tableau Server のセットアップ時に TSM で Tableau Server を設定したのと同じ Active Directory ドメインからユーザーを追加します。そのため、サーバードメインは想定済みであり、ドメインなしで AD または LDAP のユーザー名を入力できます。

注: ユーザーの氏名を入力しないでください。インポートプロセス中にエラーが発生する可能性があります。



b. 完了したら、**[ユーザーのインポート]** をクリックします。

Tableau Server 管理者エリアへのサインイン

Tableau Server のサーバー管理者は、管理者設定にアクセスし、サイト、ユーザー、プロジェクトの構成や他のコンテンツ関連のタスクを行うことができます。

プロセッサー、キャッシュ、認証、分散展開、他の関連する構成など、サーバー設定を変更する場合は、Tableau サービス マネージャーの Web UI へのサインインを参照してください。

Tableau Desktop を実行している場合、Tableau Server にサインインしてコンテンツやデータ ソースのパブリッシュやアクセスを行う方法については、「Tableau Desktop での Tableau Server へのサインイン」を参照してください。

Tableau Server 管理者ページにサインインする方法は、次のとおりです。

1. ブラウザーを開き、サーバーURL を入力します。URL は次のようになります。

   `http://localhost/` (サーバー コンピューター上で直接作業している場合)

   `http://MarketingServer/` (サーバーの名前を知っている場合)

   `http://10.0.0.2/` (サーバーの IP アドレスを知っている場合)

   サーバーがポート80 を使用していない場合は、次の例のように URL にポート番号を含める必要があります。

   `http://localhost:8000/`

   `http://MarketingServer:8080/`

   `http://10.0.0.2:8888/`

   ここで、`8000`、`8080`、`8888` は自身が構成したポートです。

   Tableau Server は、アイデンティティプールが構成されているかどうかに応じて、以下のいずれかのページを表示します。

   - アイデンティティプールが構成されていない場合、ユーザー名とパスワードを入力するページが表示されます。

   - アイデンティティプールが構成されている場合、プライマリ(初期プール (TSM で構成済み)) およびセカンダリ(アイデンティティプール) サインイン オプションを含む Tableau

Server のランディング ページが表示されます。

2. Tableau Server のインストール時に作成したサーバー管理者の認証資格情報を入力します。

   "既定" サイトのメイン ページに移動します。ここからユーザーやサイトを作成したりコンテンツを管理したりできます。

サーバー管理者アカウントとパスワードのリセット

最初のサーバー管理者アカウントのパスワードを紛失した場合は、次のコマンドを実行します。

1. tsm reset
2. `tabcmd initialuser`「initialuser」を参照してください。

Tableau Web 環境の管理者エリアの移動

Tableau Server または Tableau Cloud の管理者は、他のユーザーが利用できない管理者設定にアクセスし、サイト、ユーザー、プロジェクトの構成や他のコンテンツ関連のタスクを行うことができます。

この記事の設定では、Tableau Web 環境について記載しています。適切な認証資格情報が設定されている Tableau Server 管理者は、プロセッサー、キャッシュ、認証、分散展開、関連する

構成など、TSM Web 環境を使用してサーバー設定を変更することもできます。詳細については、Tableau サービス マネージャーの Web UI へのサインインを参照してください。

サイト ロールとサイト数に基づくアクセス

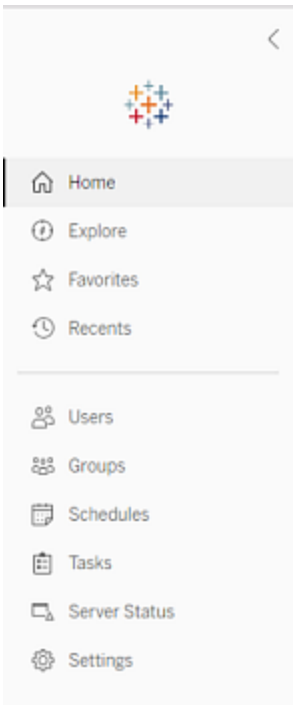
Tableau Server または Tableau Cloud にサインインすると利用できるメニューは、以下の条件によって変わります。

- サイト管理者やサーバー管理者かどうか。

  サイト管理者アクセスは Tableau Cloud および Tableau Server で利用できます。サーバー管理者アクセスは Tableau Server のみで利用できます。

- アクセスできるサイトは 1 つのみ、または複数なのか。

サーバー管理者

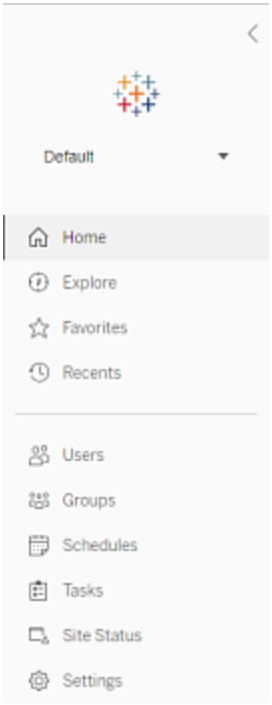
単一サイトサーバーではサイト セレクターが表示されず、他のすべてのメニューは同様です。	

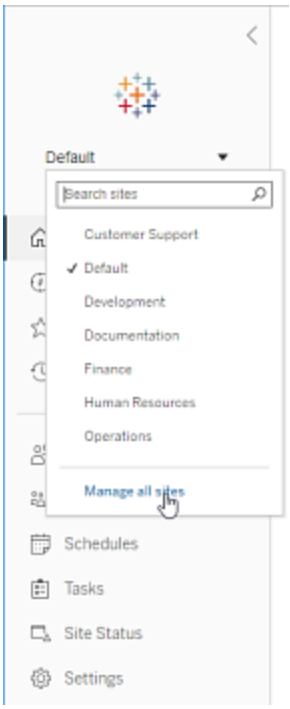
複数サイト環境では、左側のメニューにより、特定のサイトやすべてのサイトを修正したり、ユーザー、グループ、スケジュール、タスク、サーバー設定を構成したりすることができます。

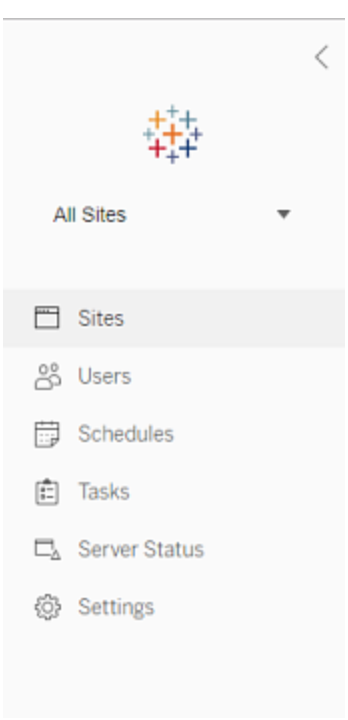
すべてのサイトに影響するサーバー管理者設定にアクセスするには、現在のサイト名の横にある矢印をクリックしてサイト メニューを開き、**[すべてのサイトを管理]** を選択します。

[コンテンツ] タブと**[グループ]** タブがなくなるほか、サイト メニューのテキストが **[すべてのサイト]** に変わり、サーバー全体の設定を管理していることを示します。また、**[サーバー ステータス]** などのオプションにサーバー全体の視点が反映されます。

サイト管理者 メニューに戻るには、**[すべてのサイト]** を選択して管理するサイトを選択します。





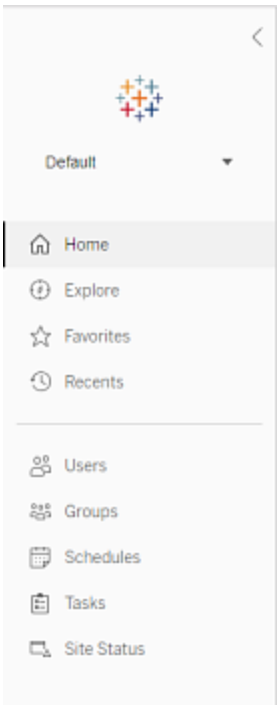


サイト管理者

Tableau Cloud または Tableau Server のサイト管理者であり、複数サイトへのアクセス権を持っている場合は、管理するサイトを選択するメニューや、そのサイトのコンテンツ、ユーザー、グループ、スケジュール、タスクを管理するメニュー、およびそのステータスをモニタリングするメニューが利用できます。

サイトセレクターには、現在のサイト名が表示されます。別のサイトに移動するには、サイトメニューを選択してからサイト名を選択します。

1 つのサイトのみへのアクセス権を持っている場合は、サイトセレクターは表示されませんが、他のすべてのメニューは同様です。



サーバー管理者のタスク

サーバー管理者 (Tableau Server Enterprise でのみ利用可能) は、次の操作を実行できます。

- サーバーのステータスとアクティビティを監視する。
- ログ ファイルを生成する。
- サイトを追加し、サイト設定を編集する。サーバー管理者のみが、サーバーにサイトを追加できます。
- サーバーにユーザーを追加し、ユーザーをサイトに割り当てる。
- サイト グループを追加および管理する。
- アイデンティティ プールにユーザーを追加/削除します。

特定サイトでのみ設定を管理するには、まずはそのサイトに移動する必要があります。各サイト内では、次の操作を実行できます。

- 管理者 コンテンツ: プロジェクトを作成する、あるプロジェクトから別のプロジェクトにコンテンツを移動する、パーミッションを割り当てる、コンテンツ リソースの所有権を変更するなど。
- 抽出更新とサブスクリプションのスケジュールを管理する。
- サイトの活動を監視し、ワークブック パフォーマンス メトリクスを記録する。
- ユーザーがパブリッシュするコンテンツ用に、保存スペースの制限を管理する。
- Web 作成を許可する。
- リビジョン履歴を有効にする。
- サイト管理者に、ユーザーの追加および削除を許可する。
- サイトが各ライセンス タイプ (Creator、Explorer、Viewer) に対して使用できるライセンスの最大数を設定します。

- ユーザーにワークブックおよびビューのサブスクライブを許可し、コンテンツの所有者に他のメンバーにコンテンツのサブスクライブを許可する。
- お気に入り用のオフライン スナップショットを有効にする (iOS のみ)。

サイト管理者のタスク

Tableau Cloud または Tableau Server のサイト管理者は、次の操作を実行できます。

- 管理者コンテンツ: プロジェクトを作成する、あるプロジェクトから別のプロジェクトにコンテンツを移動する、パーミッションを割り当てる、コンテンツ リソースの所有権を変更するなど。
- 抽出更新およびサブスクリプションのスケジュールを表示、管理、および手動で実行する。
- サイト ユーザーを追加および管理する (サーバー管理者によって許可されている場合。サイト設定 リファレンスを参照してください)。
- サイト グループを追加および管理する。
- サイト アクティビティを監視する。

Tableau サービス マネージャーの Web UI へのサインイン

このトピックでは、Tableau サービス マネージャー (TSM) の Web UI にサインインする方法について説明します。TSM Web ページは、ユーザー認証、サーバー プロセス、キャッシュ、他のサーバー関連設定など、Tableau Server の設定を構成するのに使用されます。TSM の構成は、コマンド ライン シェルから行うこともできます。tsm コマンド ライン リファレンスを参照してください。

重要: TSM への署名は、Tableau Server へのサインインと同じ*ではありません*。TSM を使用してサーバーを構成します。TSM を実行しているコンピューター上で管理者権限を持つアカウントが必要です。企業がどのように編成されているかに応じて、TSM 管理者は Tableau Server アカウントを持たないユーザーになる場合があります。Tableau Server 管理者は、サイト、ユーザー、製品、その他のコンテンツ関連のタスクを作成および編集するための管理ページにアクセスできます。

Tableau Server 管理者として Tableau Server へサインインするには、Tableau Server 管理者エリアへのサインインを参照してください。

Tableau Desktop を実行している場合、Tableau Server にサインインしてコンテンツやデータソースのパブリッシュやアクセスを行う方法については、「デスクトップでの Tableau Server または Online へのサインイン」を参照してください。

要件

- TSM へのサインインに使用するアカウントには、Tableau Server がインストールされているローカル コンピューターで変更を加えるための認可が必要です。

  特に TSM へのサインインに使用するアカウントは、初期化中に作成された TSM 認可グループのメンバーでなければなりません。TSM 認可グループの既定の名前は `tsmadmin` です。

  TSM認可グループのユーザー アカウントを表示するには、Bash シェルで次のコマンドを実行します。この例では、既定のグループ名 `tsmadmin` を使用します。

  ```
  grep tsmadmin /etc/group
  ```

  ユーザー アカウントがグループにない場合、次のコマンドを実行してユーザーを `tsmadmin` グループに追加します。

  ```
  sudo usermod -G tsmadmin -a <username>
  ```

  Tableau Server on Linux は、コア認証シナリオで PAM を使用しています。TSM 管理のための PAM 統合の詳細については、TSM 認証を参照してください。

- Tableau Server の分散展開を実行している場合は、最初のノードを実行しているコンピューターのホスト名または IP アドレスを入力してください。上記の説明の通り、入力する認証資格情報には、最初のノードを実行しているコンピューターへの管理アクセス権が必要です。

- サーバー URL で HTTPS プロトコルを指定します。Tableau Server はインストール プロセスの一環として自己署名証明書をインストールします。このため、プロトコルは `https` で指定する必要があります。自己署名証明書および TSM 接続の証明書の信頼の詳細については、TSM クライアントの接続を参照してください。
- URL で TSM の Web UI のポート (`8850`) を指定します。
- ローカル ファイアウォールを実行している場合は、ポート 8850 を開きます。ローカル ファイアウォールの構成を参照してください。
- TSM を実行しているコンピューターのホスト名または IP アドレスを指定する必要があります。Tableau Server の前にロード バランシングまたはプロキシ ソリューションを設定した場合には、ロード バランサーやプロキシ アドレスを指定しないでください。
- セキュリティのベストプラクティスとして、TSM ポート (既定では `8850`) をインターネットに公開しないでください。

TSM の Web UI へのサインイン

1. ブラウザーを開いて Tableau Server URL を入力し、専用の TSM の Web UI ポートを追加します。

   URL は次のようになります。

   `https://localhost:8850/` (サーバー コンピューター上で直接作業している場合)

   `https://MarketingServer:8850/` (サーバーの名前を知っている場合)

   `https://10.0.0.2:8850/` (サーバーの IP アドレスを知っている場合)

2. 表示されるサインイン ページで、管理者のユーザー名とパスワードを入力します。

   注: Tableau Server はインストール プロセスの一環として自己署名証明書を作成して構成します。この証明書は TSM Web UI へのトラフィックを暗号化するために使用されます。これは自己署名証明書であるため、デフォルトではブラウザーで信頼されませ

ん。そのため、接続できるようになる前に、お使いのブラウザーに証明書の信頼性に関する警告が表示されます。

サーバーのカスタマイズ

Tableau Server Web ページをカスタマイズして、自分の会社やグループ向けにパーソナライズできます。次のカスタマイズを実行できます。

- ブラウザー タブ、ツールヒント、およびメッセージに表示されるサーバー名を変更します。
- Web 環境に表示されるロゴを変更します。

  詳細については、tsm customizeを参照してください。

- Web 環境で使用する言語やビューに使用するロケールを設定します。Tableau Server の言語とロケールを参照してください。

- カスタム フォントは、Tableau Server および Tableau Server に接続されるクライアント コンピューター上にインストールする必要があります。Tableau Server でのカスタム フォントの使用を参照してください。

- サーバー サインイン ページにカスタム注記を追加します。[サインイン] 設定でテキストを追加できます。オプションで URL を追加し、テキストをリンクにすることができます。このメモは、ユーザーがサインイン エラーを受け取った場合にも表示されます。

  カスタム ノートは Tableau Mobile では表示されません。Tableau Server がアイデンティティ プールを使用して構成されている場合、サインイン カスタマイズ ノートは、すべてのサインイン オプションの下にある Tableau Server のランディング ページと、最初のプール (TSM で構成済み) ユーザーがユーザー名とパスワードを入力するページの両方に表示されます。

  カスタム注記を設定するには、Tableau Server のサイトにサインインします。左側のナビゲーション ペインで、ドロップダウンのサイト一覧から **[すべてのサイトの管理]** を選択します。**[設定]** を選択し、**[サインインのカスタマイズ]** にメッセージを追加します。

  詳細については、サーバー設定 (全般とカスタマイズ)を参照してください。

- すべてのサーバー ユーザーに表示するホーム ページのウェルカム バナーにカスタム メッセージを追加します。カスタム メッセージには、最大 120 文字のテキストとハイパーリンク、および段落区切りを 1 つ含めることができます。管理者は、サーバーの既定の Tableau ウェルカム バナーを無効にすることもできます。

  カスタムのウェルカム バナーを設定するには、Tableau Server 上のサイトにサインインします。左側のナビゲーション ペインで、ドロップダウンのサイト一覧から **[すべてのサイトの管理]** を選択します。**[設定]** を選択してから、**[カスタマイズ]** ページに移動します。

また、管理者やプロジェクト リーダーはサムネイル ビューにプロジェクトのイメージを追加することもできます。

Tableau Server の言語とロケール

Tableau Server は、複数の言語にローカライズされます。サーバー言語とロケールの設定は、これがユーザーにどのように影響するかに関するものです。**[言語]** 設定は、メニューやメッセージなどのユーザー インターフェイス (UI) 項目を制御します。**[ロケール]** 設定は、数値の表記形式や通貨などのビュー内の項目を制御します。

管理者はサーバー全体の言語とロケールを設定でき、個別ユーザーは自分の設定を構成できます(「アカウント設定」Tableau Server ヘルプで検索してください)。ユーザーが自身の言語とロケールを設定している場合、これらの設定はサーバー設定より優先されます。

サポート対象の言語

Tableau Server は、複数の言語にローカライズされます。詳細については、Tableau Server の技術仕様ページの「国際化」セクションを参照してください。

既定の設定

Tableau Serverの既定の言語はセットアップ中に決定します。ホスト コンピューターに設定されている言語が、Tableau Serverでサポートされている場合、Tableau Serverは、その言語を自身の既定としてインストールします。サポートされていない言語がコンピューターに設定されている場合、Tableau Serverは英語を既定の言語としてインストールします。

言語とロケールの決定のしくみ

ユーザーがビューをクリックしたときに表示される言語とロケールに影響を与えるもう1 つの要素はユーザーの Web ブラウザーです。サーバー ユーザーが [ユーザー アカウント] ページで **[言語]** 設定を指定しておらず、Web ブラウザーが Tableau Server でサポートされている言語に設定されている場合、その言語が使用されます。これは Tableau Server 自体が別の言語に設定されている場合も同様です。

たとえば、Tableau Server システム全体のすべてのユーザーに対して **[言語]** が英語に設定されているとします。サーバー ユーザーの Claude は Tableau Server の自分の [ユーザー アカウント] ページで言語を設定していません。Claude のブラウザーではドイツ語 (ドイツ) が言語/ロケールとして使用されています。

Claude がTableau Serverにサインインするとサーバーの UI はドイツ語で表示されます。Claude がビューをクリックすると、ビューの数字や通貨にはドイツがロケールとして使用されます。Claude が自分のユーザー アカウントの **[言語]** と**[ロケール]** をフランス語 (フランス) に設定している場合、UI とビューはフランス語で表示されます。ユーザー アカウント設定は Web ブラウザーの設定より優先され、両者は Tableau Server のシステム全体の設定より優先されます。

注意が必要なもう1 つの設定は Tableau Desktop の **[ロケール]** (**[ファイル] > [ワークブック ロケール]**) 設定です。この設定によって、ビューに表示されるデータのロケール (通貨や数値の表記形式) が決まります。既定では、Tableau Desktopの **[ロケール]** は **[自動]** に設定されています。ただし、作成者は特定のロケールを選択することによって、その設定をオーバーライドできます。上記の例では、ビュー A の作成者が **[ロケール]** を **[ギリシャ語 (ギリシャ)]** に設定した場合、ビュー A のデータはギリシャ語 (ギリシャ) ロケールを使用して表示されます。

Tableau Server ではこれらの設定を優先度の高い順に使用して、言語とロケールを決定します:

1. ワークブック ロケール (Tableau Desktop で設定)
2. Tableau Server の [ユーザー アカウント] の言語/ロケール設定
3. Web ブラウザーの言語/ロケール
4. Tableau Server の [メンテナンス] ページの言語/ロケール設定
5. ホスト コンピューターの言語/ロケール設定

Tableau Server でのカスタム フォントの使用

Tableau Server でカスタム フォントを使用できます。これを実行する際は、意図した体験をユーザーに対して確実に提供するために、次の内容を念頭に置いてください。

-

- Tableau Server を実行中のコンピューターにフォントがインストールされている必要があります。フォントをインストールしたら、Tableau Server を再起動して新しいフォントを使用します。

- フォントは、Tableau Server に接続されるクライアントコンピューター上にインストールする必要があります。フォントをブラウザーで正常に表示するには、そのフォントがローカルにインストールされている必要があります。

- ベストプラクティスとして、「web safe」を使用します。(すべての主要ブラウザーに既定でインストールされているフォント) を使用します。これにより、フォントがクライアントマシン上で正しく表示される可能性が高まります。

- ブラウザーが異なると、同じフォントでもレンダリング方法が異なります。そのため、クライアントブラウザーのカスタムのフォントがインストールされていても、異なるブラウザーで表示すると外観が異なる場合があります。この違いは、意図的な効果を出すためにコメントやタイトルで特定のスペースを使用している場合に顕著になる可能性があります。

注: Linux でのフォントのインストールに関する詳細は、Linux ディストリビューションのドキュメントとサポートを参照してください。

サーバー間でのサイトの管理

Tableau Server でサイトを計画および管理できます。サイトのユーザーとグループの管理、プロジェクトとコンテンツへのアクセスの管理、データの管理、Web 上でのビューの作成と操作を行うことができます。

サイトの概要

このセクションのトピックでは、Tableau Serverサイトの概念と複数サイトの操作の側面について説明します。トピックには、各サイトで使用する認証タイプ、ユーザーライセンスとサイトへの管理者レベルのアクセス権に関して知っておくべきことが含まれます。

サイトとは

これまで、*サイト*を「接続されているコンピューターの集まり」または「Web サイト」の略語として使用していた方もいるでしょう。Tableau では、サイトという言葉を、Tableau Server の同じインスタンス

上の他のグループおよびコンテンツから壁で守られている、ユーザー、グループ、およびコンテンツ (ワークブック、データソース) の集まりを意味するために使用します。つまり、Tableau Server はマルチテナント機能をサポートしており、サーバー管理者はサーバー上で複数セットのユーザーやコンテンツ向けの複数のサイトを作成できます。

すべてのサーバー コンテンツは、サイト単位でパブリッシュ、アクセス、および管理されます。各サイトには専用の URL と専用のユーザー セットがあります (各サーバー ユーザーを複数のサイトに追加できる場合でも)。各サイトのコンテンツ (プロジェクト、ワークブック、およびデータソース) は、他のサイトのコンテンツから完全に隔離されます。

サイト管理者のサイトへのユーザーの設定方法、パブリッシャーおよびその他のコンテンツユーザー用のサイトの構成方法、ユーザーへのコンテンツの共有および管理のパーミッションの与え方、その他についての推奨事項は、個別のサイトの管理セクションを参照してください。

ユーザーのコンテンツの入手方法についての詳細は、Tableau ユーザー ヘルプ のTableau Server 「データ ソースとワークブックのパブリッシュ」を参照してください。

認証とサインイン認証資格情報

既定では、サーバー上のすべてのサイトで、同じアイデンティティストアのタイプを使用します。Tableau Server のインストール時にこれらの設定を構成します。詳細は、初期ノード設定の構成を参照してください。

同じ Tableau Server インスタンスで一つ以上のサイトへのアクセスがあるユーザーは、各サイトで同じ認証資格情報を使用します。たとえば、Jane Smith のサイト A でのユーザー名が *jsmith* で、パスワードが *MyPassword* である場合、サイト B でもこれと同じ認証資格情報を使用します。Jane Smith が Tableau Server にサインインするときに、どのサイトにアクセスするかを選択できます。

既定サイト

Tableau Server は "既定" という名前のサイトがインストールされています。Tableau Server でシングルサイト環境を管理している場合、このサイトで作業します。また、そこであなたのユーザーが彼らの Tableau 分析を共有します。サイトを追加すると、"既定" は Tableau Server にサインインするときに選択することができるサイトの一つになります。"既定" サイトは、ユーザーがシステムに追加するサイトとは次の点で異なります。

- ユーザーが追加するサイトのように削除することはできません。ただし、名前は変更できます。
- Tableau Server に付属のサンプルおよびデータ接続が格納されています。
- 既定に使用される URL サイトが指定されていません。たとえば、"Sales (売上高)" というサイト上の "Profits (収益)" というビューの URL は `http://localhost/#/site/sales/views/profits` です。"既定" サイト上のこれと同じビューの URL は `http://localhost/#/views/profits` です。

サイトを追加する/しない理由

Tableau Server では、ユーザー、プロジェクト、グループ、データ ソース、ワークブックがサイトごとに管理されています。複数のサイトにユーザーを追加することができます。

各環境とそのニーズは固有のものです。ただし、基本的に、Tableau ビジョナリとプロダクト キーマネージャーは、本当の複数テナントのニーズのためにサイトを使用することを推奨する傾向があります。つまり、固有のユーザーのセットとそれらのコンテンツを、その他すべての Tableau ユーザーとコンテンツから完全に切り離して管理する必要がある場合のみ、新しいサイトを作成します。

サイト管理者のサイトへのユーザーの設定方法、パブリッシャーおよびその他のコンテンツユーザー用のサイトの構成方法、ユーザーへのコンテンツの共有および管理のパーミッションの与え方、その他についての推奨事項は、個別のサイトの管理セクションを参照してください。

サイトを使用するのが有意義な例

- あなたは、Tableau 分析を複数のクライアントのために使用しているコンサルタントです。そして、あるクライアントからのデータが他のクライアントに見えないように、各クライアントのためにサイトを作成しようとしています。
- ゲスト ユーザーに、サーバーの小さく限定されたエリアへのアクセスを許可しようとしています。

サイトよりも良く機能するプロジェクトの例

- データソースとレポートがサンドボックスから生産フェーズに進化するコンテンツ開発プロセス。

  ユーザーとコンテンツの一つのサイトから別のサイトへの移行は、骨の折れるプロセスです。これや、類似のプロセスにサイトを使用する充分な理由があるかもしれませんが、サイトを作成すると、サイト管理者としての管理負担が増大します。一つのサイトの各構成を更新(例えば、新規プロジェクトを作成しパーミッションを設定する)すると、通常、各追加サイトで同じ作業を繰り返す必要があります。

- 機能エリアごとにサーバーのエリアを分けようとするでしょう。

  Tableau ユーザーのグループでは、同じユーザーが複数エリアのコンテンツにアクセスする必要が出てくるのが普通です。サイトを使用すると、同じデータソースやレポートを複数のサイトにパブリッシュすることを助長します。それがデータソースの急増につながり、サーバーのパフォーマンスに悪影響を及ぼす恐れがあります。このシナリオでは、プロジェクトを使用する方が、よりシンプルな作業方法になります。

その他のアイデアについては、次のリソースを参照してください。

- トピック「プロジェクトを使用したコンテンツへのアクセスの管理」のプロジェクトを使用する理由。
- Tableau コミュニティフォーラムのサイトについて説明します。このリンクから始めてください。

サイトへの管理者レベルのアクセス権

Tableau Server には、管理者レベルのサイトロールが 3 つあります。サーバー管理者 Creator、サイト管理者 Explorer です。

サーバー管理者サイトロールは常に、利用可能な最高のライセンスであり、すべてのコンテンツへのアクセスを含め、Tableau Server へのフル アクセス権があります。このロールについての詳細は、サーバー管理者の概要を参照してください。サーバー管理者は、必要に応じてサイトも作成します。(サイト管理者は、これをするパーミッションを持っていません。)

サーバー管理者は、**サイト管理者**サイトロールのいずれかをユーザーに割り当て、特定のサイトのユーザーおよびコンテンツ フレームワークの作成および管理を委任することができます。コンテンツ フ

レームワークにより、Tableau ユーザーはデータソースおよびワークブックの共有、管理、およびそれらへの接続が可能になります。

- **サイト管理者 Creator** は、データへの接続のほかデータソースやワークブックの作成やパブリッシュも行う管理者に割り当てます。このサイトロールには **Creator** ライセンスがあります。
- **サイト管理者 Explorer** は、ユーザーがコンテンツフレームワークの管理は行うものの、コンテンツ自体の編集を行う必要がない場合に割り当てます。このサイトロールには **Explorer** ライセンスがあり、表示や操作のアクセス権が付与されます。

既定では、サイト管理者のサイトロールにより、サイトのユーザーやグループの作成および管理、サイト上のコンテンツを整理するためのプロジェクトを作成、必要なコンテンツへのアクセスを許可するようユーザー (グループ) にパーミッションを割り当てる、抽出の更新のスケジュール設定、その他いくつかのタスクなどが可能になります。

サーバー管理者は、サイト管理者のユーザー管理タスクを拒否できます。このようにするのは、データエキスパートに対してサイト管理者 Creator ロールを使用する場合などです。つまり、これらのユーザーに対して参照元データへの接続の管理、「真の単一ソース」のデータソースの作成やパブリッシュ、上位プロジェクトの作成、プロジェクト間のコンテンツ整理については制限なく許可するものの、不必要にサイトユーザーの追加や削除を行ってほしくない場合です。

各サイトで、サーバー管理者は、サイト管理者がグループとコンテンツを管理できるがユーザーの追加または削除、あるいはユーザーのサイトロールの設定はできないように、サイト管理者のアクセスを制限することも可能です。

組織によっては、同じ人物が 1 つ以上のサイトのサーバー管理者とサイト管理者の両方となる場合があります。たとえそうであっても、サイト管理者とサーバー管理者によって実行されるタスクは明確です。

ライセンスとユーザー数の上限

サーバーユーザーを複数のサイトに追加し、サイトごとにサイトロールとパーミッションを設定できます。複数サイトに所属するユーザーは、各サイトに対してライセンスを持つ必要はありません。各サーバーユーザーに必要なライセンスは 1 つだけです。ユーザーが使用するライセンスは、サーバー

上で最も上位のサイトロールに対応します。ライセンスとサイトロールの重なりについては、ユーザーのサイトロールの設定を参照してください。

サーバー管理者は、**[ユーザー数の制限]** 設定 (**[サイト] <*名前*> > [設定]**を選択) を使用してサイトのユーザー制限を指定したり、サイトレベルで割り当てられた Creators、Explorers、Viewers の数を制限するサイトロール制限を設定したりできます。詳細については、サイトロールの制限の管理を参照してください。

ライセンスを与えられているユーザーのみが数えられます。たとえば、ライセンスを持つユーザーの数が 90 (管理者を含む)、ライセンスを持たないユーザーの数が 20 のサイトでは、ユーザー数は 90 になります。サーバー全体のライセンス数とサイトロール数を表示する方法については、サーバー ライセンスの表示を参照してください。

サイトのエクスポートまたはインポート

注: Tableau Server から Tableau Cloud へのサイトの移行の詳細については、「Tableau Cloud マニュアル移行ガイド」を参照してください。

別のサイトから情報をインポート (移行) して、新しい Tableau Server サイトをプロビジョニングすることができます。これを行うには、既存サイト (ソースサイト) の情報をファイルにエクスポートします。その後、その情報を検証し、ターゲットサイトにインポートする手順を完了します。

サイト移行オプション

次のいずれかの方法を実行して、サイトを移行できます。

- 同じ Tableau Server インスタンス上の別のサイトへ
- 別の Tableau Server インスタンス上のサイトへ
- Tableau Server Windows 上から Tableau ServerLinux 上に、またはその逆に。

注: Tableau Server のインスタンス間でサイトを移行する場合、移行先サイトは、移行元サイトの Tableau Server のバージョンと同じかそれ以降の Tableau Server のバージョン上にある必要があります。移行元サイトと移行先サイトの両方が、サポートされているバージョンの Tableau Server からのものである必要があります。

サイト移行に関する制限事項

サイトエクスポートで保持される情報と保持情報

- 作成したエクスポート ファイルには、ワークブック、プロジェクト、データ ソース、およびユーザーが保持されます。これには、コンテンツに対して設定されたパーミッション、ユーザーのお気に入りリスト、およびサイト容量が含まれます。
- ユーザーのカスタム ビューは保持されますが、サイト移行のタイプによってはカスタム ビューの URL が変わり、ユーザーのビューに対するブックマークが機能しなくなる場合があります。
- サイトを Tableau Server にエクスポートして別の Tableau Server サイトにインポートする場合は、サブスクリプションおよび抽出の更新スケジュールが保持されます。

サイトエクスポートで保持されない情報

- サイトの管理ビューに表示される利用状況データは保持されません。たとえば、ビューとデータ ソースの数、ユーザー アクション、およびパフォーマンス データです。
- サイトをエクスポート中でも進行中のバックグラウンダー ジョブはエクスポートされず、インポートの完了後は新しいサイトに表示されなくなります。
- データ接続に埋め込まれた OAuth アクセス トークンはリセットされます。これらのデータ ソースでは、接続を編集して参照元データに再認証する必要があります。
- Prep フローのスケジュールとタスクは含まれません。これらは手動で移行する必要があります。
- ユーザーの個人スペースに保存されたコンテンツは含まれません。
- コレクションは、Tableau Server 2025.3 までサイトのエクスポートに含まれません。

ソースサイトとターゲットサイトの準備

サイトをエクスポートする前に、次のチェックリストを完成させて両方の環境を準備します。手順の一部は、両方のサイトが同じサーバー インスタンス上にあるか別のインスタンス上にあるかで異なります。

古くなったコンテンツの削除

ソース サイトには新しいサイトにインポートする内容だけが含まれるようにします。ベストプラクティスとして、新しいサイトに含めたくないものは、エクスポート/インポートに含まれるか、手動で移行する必要があるかに関係なく、ソース サイトから削除します。使用していないワークブック、データ ソース、プロジェクトは削除してください。使用しなくなった Prep フローまたはフロー スケジュールがある場合は、これらも削除します。

使用されなくなったユーザーの削除

すべてのサーバー ユーザーがライセンスを持っていることを確認し、使用されなくなったアカウントは削除します。インポートプロセス中にユーザーを削除することはできないため、2 つのサイトが同じサーバー インスタンス上にある場合、ソース サイトからエクスポートしたすべてのユーザーがターゲットサイトにインポートされます。

ターゲットサイトの作成または識別

サイト ファイルは、ターゲット Tableau Server インスタンス上に既に存在するサイトにインポートする必要があります。インポートプロセスでは、インポート ファイルに含まれていないすべてのデータがターゲット サイトから削除されるため、空のサイトにインポートすることをお勧めします。詳細については、「サイトの追加または編集」を参照してください。

サイト ID を見つける

サイトのエクスポートやインポートに使用する tsm コマンドには、サイト ID を取り出すパラメーターが必要です。サイト ID は、Web ブラウザーでサイトにサインインする際の URL から入手できます。

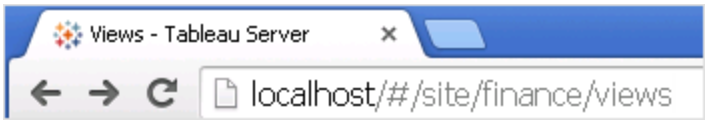
サーバー上に存在するサイトが 1 つだけである場合、サイトの名前は **"既定"** です。"既定" サイトにサインインすると、ブラウザーの URL は次のようになります。

```
https://server-name/#/projects
```

URL に /site パラメーターがない場合は、これが既定のサイトであることを示しています。既定サイトのサイトID は "Default" (二重引用符なし) です。

マルチサイトの Tableau Server 展開では、ブラウザー URL で #/site/ に続いてサイトID が含まれています。サイトID が finance であるサイトで [ビュー] ページに移動すると、次の URL が表示されます。

```
https://localhost/#/site/finance/views
```



アイデンティティストアを確認する

同じタイプのユーザー アイデンティティストアを使用していないサイトからのエクスポートや、そうしたサイトへのインポートを行うことができますが、インポートに使用するマッピング ファイルを変更する必要があります。この手順はインポートプロセスに組み込まれており、ステップ 3 :サイト設定が正しくマッピングされたことを確認するで説明されています。

必要に応じてターゲットサーバーでユーザーを作成する

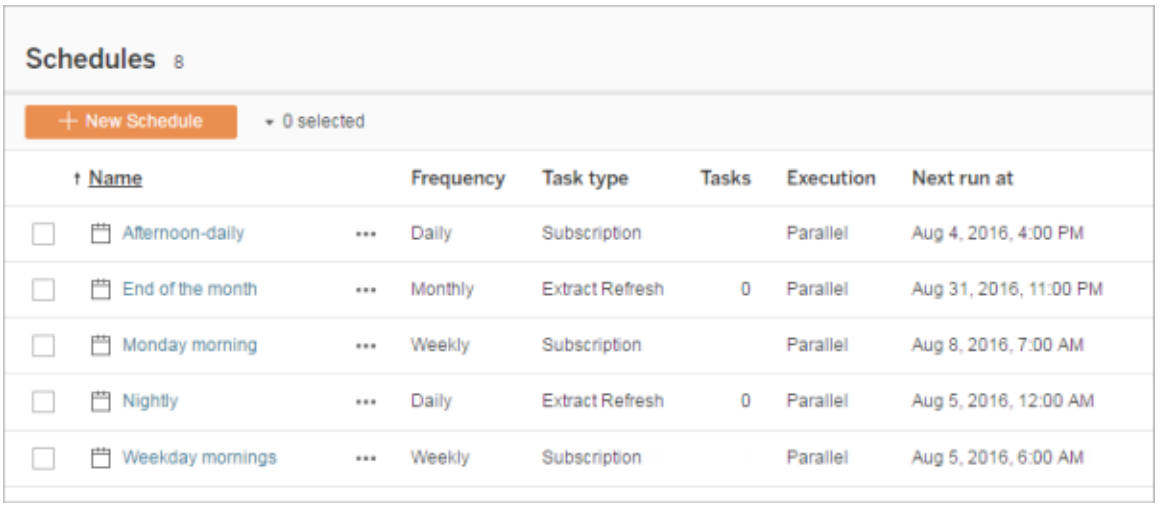
サイトのインポートプロセスではユーザーがターゲットサイトに割り当てられます。ソース サイトがターゲット サイトと異なる Tableau Server インスタンス上にある場合は、インポートを実行する前にターゲット サーバー上でユーザーを作成する必要があります。2 つのサイトが同じ Tableau Server インスタンス上にある場合は、ターゲット サイトから既存のユーザーにアクセスできるため、この手順をスキップできます。

ターゲットサーバーでのサブスクリプション配信の構成

サブスクリプションはインポートされますが、配信するようにサーバーを構成する必要があります。詳細については、サブスクリプションのサーバー設定を参照してください。

スケジュールの確認

[スケジュール] ページには、抽出更新およびサブスクリプションに使用する既存のスケジュールが一覧表示されます。



ある Tableau Server サイトから別のサイトへの移行の場合、ソースサイトの既定のスケジュールに割り当てられた更新およびサブスクリプションは、ターゲットサイトの同じスケジュールにマッピングされます。ターゲットサイトに存在しないスケジュールがソースサイトにあり、ターゲットサイトが別の Tableau Server インスタンス上にある場合は、ソースのスケジュールをマッピングするターゲットサイトでスケジュールを作成する必要があります。この処理が期待どおりに実行されるよう、マッピングファイルを編集できます。

ユーザーやスケジュールが少ないターゲットにインポートするためのヒント

ターゲットサイトのユーザーやスケジュールがソースサイトより少ない場合、多対 1 のインポートはサポートされません。これに対処する方法は、ソースサイトとターゲットサイトが同じ Tableau Server インスタンス上にあるかどうかで異なります。

次のアプローチのうち、お使いのサイトの移行使用事例に適したものを実行できます。

- エクスポートする前に余分なユーザーやスケジュールをソースサイトから削除します。これは、2 つのサイトが同じサーバーインスタンス上にある場合に推奨されるオプションです。
- インポートを開始する前に、欠落したユーザーやスケジュールをターゲットサイトに追加します。ターゲットサイトが別のサーバーインスタンス上にある場合は、これを実行する必要があります。

- インポートプロセスの途中で、欠落したユーザーやスケジュールをターゲットサイトに追加し、手動でマッピング ファイルを更新します。これは、2 つのサイトが同じサーバー インスタンス上にある場合にのみ実行できるオプションです。

- インポートプロセスの途中にユーザーやスケジュールをターゲットサイトの異なるユーザーやスケジュールに手動でマッピングします。サーバー間でユーザー名が異なる場合にはこれを実行する必要があります (たとえば、*adavis@company.com* という名前のエクスポートされたユーザーが、ターゲットサイトでは *davisa* と定義されている場合)。

サイトの移行

サイトの移行プロセスを完了するには、`tsm sites` コマンドを使用する必要があります。次の手順では、ソース サイトから情報をエクスポートし、サイト設定をマッピングし、マップされたファイルをターゲットサイトにインポートします。

ステップ 1: サイトをエクスポートする

ソース Tableau Server マシンで、次のコマンドを入力します。

```
tsm sites export --site-id <source-siteID> --file <filename>
```

`export` コマンドを使用するときには、Tableau Server を実行中である必要があります。エクスポートプロセス中、Tableau Server はエクスポートしているサイトをロックします。

たとえば、サイト ID **weather-data** のサイトをファイル **export-file.zip** にエクスポートするには、次のように入力します。

```
tsm sites export --site-id weather-data --file export-file
```

Tableau Server の既定では、`<export-file>.zip` は `/var/opt/tableau/tableau_server/data/tabsvc/files/siteexports` に保存されます。詳細については、tsm ファイル パスを参照してください。

ステップ 2 : インポート マッピング ファイルを生成する

ターゲットサイト用のインポート ファイルを生成するには、ステップ 1: サイトをエクスポートするの手順を完了した際に作成した .zip ファイルが必要です。

注: エクスポートされたファイルには、既定で "他のユーザー" の読み取りパーミッションが付与されていません。ファイルをインポートするユーザーによっては、所有者以外のユーザーがファイルを読み取れるように、ファイルのパーミッションを調整する必要がある場合があります。

1. ターゲット Tableau Server マシンで、エクスポートされた .zip ファイルを、Tableau Server がインポート用のファイルを検索するディレクトリにコピーします。例は次のとおりです。

   ```
   /var/opt/tableau/tableau_server/data/tabsvc/files/siteimports
   ```

2. インポートプロセスでは新しいサイトが作成されないので、ターゲットサイトが Tableau Server に既に存在することを確認します。詳細については、ソースサイトとターゲットサイトの準備を参照してください。

3. ターゲット Tableau Server マシンで次のコマンドを実行します (Tableau Server が実行されている必要があります)。

   ```
   tsm sites import --site-id <target-siteID> --file <export-file.zip>
   ```

   このコマンドにより、ソースサイト設定がターゲットサイトにマッピングされる方法を示す .csv ファイルのセットが生成されます。この記事の次のセクションで説明する手順では、これらのマッピングを確認し、必要に応じて調整します。

   既定では、、これらの .csv ファイルは `siteimports` の下に作成される `mappings` ディレクトリに生成されます。例は次のとおりです。

   ```
   /var/opt/tableau/tableau_
   server/data/tabsvc/files/siteimports/working/import_<id>_<date-time>/mappings
   ```

   詳細については、tsm ファイル パスを参照してください。

ステップ 3 :サイト設定が正しくマッピングされたことを確認する

前のセクションで生成した .csv ファイルには、インポートの完了時にソースサイトのリソースをターゲットサイトに割り当てる方法が記述されています。Tableau Server がマッピングできなかったために編集する必要があるファイル内のアイテムは、一連の疑問符 (`???`) で示されています。インポートプロ

セスを完了する前に、疑問符をターゲットサイト上の有効な割り当てに置き換える必要があります。

重要:ユーザー、スケジュール、およびパブリッシュされたコンテンツ リソースのマッピングには、特にソース サイトとターゲット サイトが別の Tableau Server インスタンス上にある場合に、いくつかの要件が適用されます。詳細については、この記事の前の部分にあるソースサイトとターゲットサイトの準備を参照してください。

マッピング ファイルを検証するには

1. `tsm sites import` コマンドによって生成された .csv マップ ファイルが含まれるディレクトリに移動します。既定のディレクトリは次のとおりです。

   ```
   /var/opt/tableau/tableau_
   server/data/tabsvc/files/siteimports/working/import_<id>_<date-
   time>/mappings
   ```

2. お気に入りのテキスト エディターを使用して `mappings` ディレクトリ内にある .csv ファイルのいずれかを開き、次のようにします。

   a. マッピングが正しいことを確認します。

   b. エントリに一連の疑問符 (`???`) が表示される場合は、有効な値で置き換えてください。

      これらの各ファイル内の設定については、この記事の後の部分のマッピングファイルの内容リファレンスにある表を参照してください。

   c. 変更を保存し、CSV ファイルの書式設定を保護します。

   残りの .csv ファイルについてこのプロセスを繰り返します。

ステップ 4: 正しくマッピングされたファイルをターゲット サイトにインポートする

.csv ファイル内のサイト マッピングを検証した後は、設定を新しいサイトにインポートして移行プロセスを完了できます。

1. ターゲットTableau Server マシンで次のコマンドを実行します:

   ```
   tsm sites import-verified --import-job-dir <import-id-
   directory> --site-id <target-siteID>
   ```

   例は次のとおりです。

   ```
   tsm sites import-verified --import-job-dir
   /var/opt/tableau/tableau_
   server/data/tabsvc/files/siteimports/working/import_ff00_
   20180102022014457
   --site-id new-site
   ```

2. 正常に処理されたことを通知するメッセージが表示されたら、新しいサイトにサインインし、予測どおりにすべてインポートされたことを確認します。

注: `tsm sites import` および `tsm sites export` コマンドは、エラーが発生した場合に、サイトをロック状態のままにすることができます。サイトをロック解除するには、`tsm sites unlock` コマンドを使用します。

マッピングファイルの内容リファレンス

次の表は、`tsm site import` コマンドの実行時に作成される各マッピング ファイル内の列を示しています。

CSV ファイル名:mappingsDomainMapperForGroups

列のタイトル	編集可能か?	説明
source_name	いいえ	ソース サイトのユーザー グループ名。
source_domain_name	いいえ	ソース サイトのアイデンティティ ストア タイプ。**local** (ローカル アイデンティティ ストアの場合) またはドメイン名 (Active Directory または

		LDAP 外部アイデンティティストアの場合)。
target_domain_name	はい*	ターゲットサイトのアイデンティティストア タイプ。ローカル アイデンティティストアの場合は **local**、Active Directory または LDAP 外部アイデンティティストアの場合はドメイン名 (example.com や example.lan など)。 ***[すべてのユーザー]** グループの場合は、ターゲットサーバーが Active Directory アイデンティティストア用に構成されているとしても、**target_domain_name** 値を **local** のままにしておきます。**[すべてのユーザー]** グループはすべての Tableau Server に存在する必要がある、特別な既定のユーザー グループです。

CSV ファイル名:mappingsScheduleMapper

列のタイトル	編集可能か?	説明
source_name	いいえ	ソースサイト上にあるカスタムおよび既定の抽出またはサブスクリプションのスケジュール名。
source_scheduled_action_type	いいえ	スケジュールのタイプ。ソースサイトの抽出更新の場合は**抽出**、またはサブスクリプション配信の場合は**サブスクリプション**。
target_name	はい	ターゲットサイト上のカスタム スケジュールの名前。この値は編集することができます。たとえば、スケジュールがソースサイト上で**金曜日のアップデート**と名付けられている場合、ターゲットサイト上で**金曜日の更新**に変更することができます。

target_scheduled_action_type	いいえ*	スケジュールのタイプ。ターゲットサイトの抽出更新の場合は**抽出**、またはサブスクリプション配信の場合は**サブスクリプション**。 *まれなケースとしてこの列に疑問符 (???) が表示される場合は、**source_scheduled_action_type** の下にあるエントリに合わせて**抽出の更新**または**サブスクリプション**に置き換えます。

CSV ファイル名:mappingsSiteMapper

列のタイトル	編集可能か?	説明
source_url_namespace	いいえ	ソースサイトのサイトID。
target_url_namespace	いいえ	ターゲットサイトのサイトID。

CSV ファイル名:mappingsSystemUserNameMapper

列のタイトル	編集可能か?	説明
source_name	いいえ	ソースサイトのユーザーのユーザー名属性。
source_domain_name	いいえ	ソースサイトのアイデンティティストアタイプ。**local** (ローカルアイデンティティストアの場合) またはドメイン名 (Active Directory または LDAP アイデンティティストアの場合)、または**外部**。
target_name	はい	インポート後にターゲットサイトに割り当てられるユーザーのユーザー名属性。 リスト内のすべてのユーザー名がターゲットサー

		バーに存在することを確認し、疑問符 (???) はターゲットサーバーに存在するユーザー名に置き換えます。 行を CSV ファイルに追加しても、ユーザー名を作成することはできません。同様に、行を削除しても、ユーザー名を削除することはできません。 **target_name** 列のユーザー名を編集してソースのユーザー名と別にすることができるのは、その名前を使用しているターゲットサーバー上にそのユーザーが既に存在している場合です。 たとえば、あるユーザーの **source_name** 値を **agarcia@company.com** にし、**target_name** 値を **ashleygarcia@company.com** にすることができます。 ソースサイト上のユーザーは、ターゲットサイト上の 1 つのユーザー名だけにマッピングできます。
target_domain_name	はい	ターゲットサイトのアイデンティティストアタイプ。**local** (ローカル アイデンティティストアの場合) またはドメイン名 (Active Directory または LDAP 外部アイデンティティストアの場合)。

CSV ファイル名: MappingsScheduleRecurrenceMapperWithAutoCreation

このファイルを更新する必要はありません。

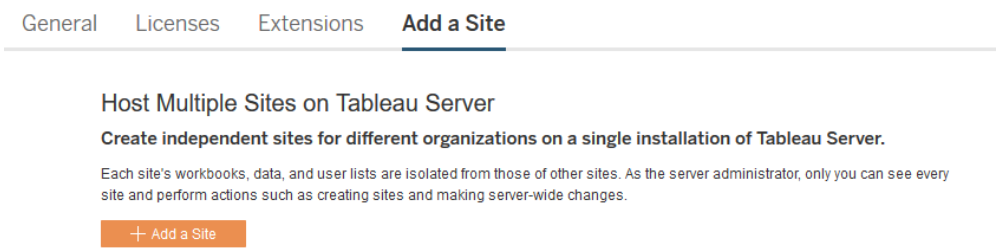
サイトの追加または削除

Tableau Server には、既定という名前のサイトがあります。サーバー管理者は、組織の変更のニーズに応じてサイトを追加または削除できます。

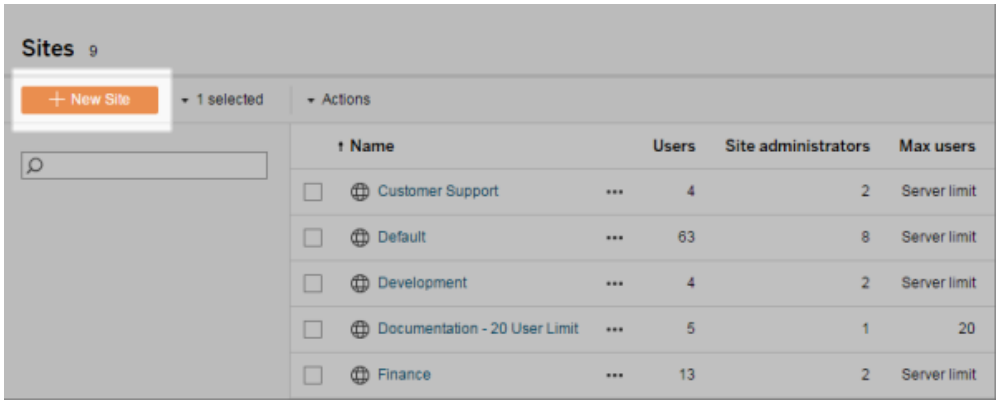
サイトを追加する

1. 次のいずれかを実行します。

   - サーバーに初めてサイトを追加する場合は、**[設定] > [サイトの追加]** を選択して、**[サイトの追加]** をクリックします。



   - 既にサイトを追加している場合は、サイト メニューで、**[すべてのサイトを管理]**、**[新しいサイト]** の順にクリックします。



2. サイトの設定を編集して、組織に合わせてカスタマイズします。

サイトを削除する

サーバー管理者は、Tableau Server に追加されたサイトを削除できます。サイトを削除すると、そのサイトにパブリッシュされたワークブックやデータ ソースもユーザーも削除されます。別のサイトにも所属しているユーザーは削除されません。ユーザーを完全に削除するには、[サーバー ユーザー] ページへ進みます。

注: 既定サイトは削除できません。

1. サイト メニューで、**[すべてのサイトを管理]**、**[サイト]** の順にクリックします。
2. 削除するサイトを選択し、**[アクション]** メニューで **[削除]** をクリックします。
3. 表示された確認ダイアログ ボックスで **[削除]** をクリックします。

サイトの可用性

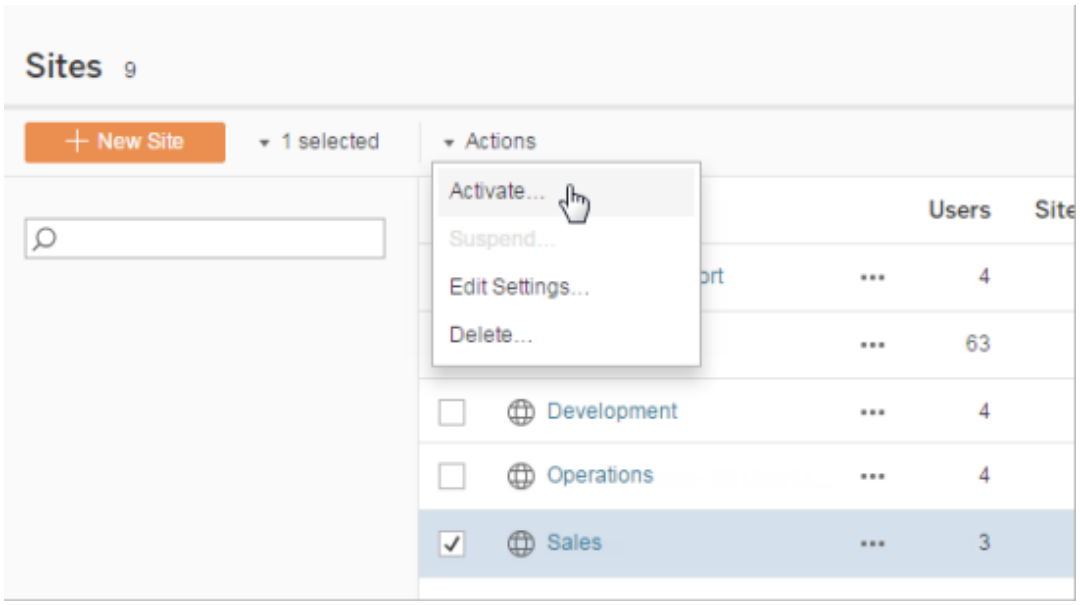
サイトのインポート失敗、またはサーバー管理者が一定期間サイトを一時停止するよう選択することにより、サイトが一時停止またはロックされる可能性があります。

サイトが一時停止された場合、サイトを有効化し、再度利用できるように設定できるのはサーバー管理者のみです。

注: サイトがロックされ、[サーバー] インターフェースから [サイト] ページにアクセスできない場合は、tsm sites unlock コマンドを使用して状態をアクティブに変更してください。

サイトを有効化または一時停止するには

1. サイト メニューで、**[すべてのサイトを管理]**、**[サイト]** の順にクリックします。
2. サイトを選択してから、**[アクション] > [有効化]** または **[一時停止]** を選択します。



サイトロールの制限の管理

サーバー管理者はサイトロールの制限を作成して、特定のサイトで使用できる各種類 (Creator、Explorer、または Viewer) のライセンスの最大数を設定できます。サーバー管理者がサイトロールの制限を設定した後、サイト管理者はそのサイトロールの制限数までユーザーを追加できます (このとき、ライセンスが自動的に使用されます)。ユーザーがサーバー全体で複数のサイトのメンバーである場合、そのユーザーは各サイトのロール制限に対してカウントされますが、ライセンスは 1 つのみ (サーバー上で最も高いサイトロールに対応) 使用します。サーバー管理者は、サイトロールの制限に対してカウントされません。

各サイトロールの機能の詳細については、Tableau Server ヘルプのライセンス発行の概要で「ユーザーベースのライセンス」を参照してください。サーバー管理者がサイトロールの制限を設定する方法については、サイト設定リファレンスを参照してください。

サイトでロールの制限を作成する

サイト管理者がユーザーを管理できるようにする前に、サーバーライセンスを表示して、サーバー間でライセンスを割り当てる方法を決定するか、新しいユーザーのための容量をサーバーに追加できます。

サーバー管理者は、Web UI の [設定] ページか REST API を使用してサイトロールの制限を構成できます。

サイトでサイトロールの制限を設定するには、次のようにします。

1. サイトの [設定] ページの [全般] タブに移動します。
   - サイトが 1 つである場合は、サイドナビゲーションで [設定] をクリックし、[全般] をクリックします。
   - 複数のサイトがある場合は、構成するサイトを選択してから、[設定] をクリックし、[全般] をクリックします。
2. [ユーザーの管理] で、サーバー管理者とサイト管理者がユーザーを追加および削除できることを指定します。
3. [ユーザー数を制限] で、[サイトロール制限] を選択します。
4. Creator、Explorer、Viewer についての制限を設定します。
5. [保存] をクリックします。

サーバー管理者がサイトロールの制限を設定する際は、次の制限が適用されます。

- サイトロールの制限は、サーバー上でライセンス認証された当該の種類のライセンス数を超えることはできません。
- サイトロールの制限を、そのサイトに既に存在するそのサイトロールのユーザー数より小さくすることができません。
- ある種類のライセンスに制限を設定する場合は、すべての種類のライセンスに制限を設定する必要があります。
- あるロールの制限を空白のままにした場合は、サーバーのライセンス制限が使用されます。

サイトロールの制限に達した場合

サイトが Explorer に対して設定されたロール制限に達していない場合、Explorer としてサイトに追加されたユーザーは Explorer ライセンスを使用します。ロールの制限に達している場合は、次のように処理されます。

- より高いレベルのライセンスが使用可能である場合、ユーザーは Explorer としてサイトに追加されますが、Creator ライセンスが使用されます。
- より高いライセンスが使用可能でない場合、ユーザーはライセンスのないユーザーとしてサイトに追加されます。

サイトロールの制限は、特定のサイトロールとは異なるライセンスを使用しているユーザー (Explorer ライセンスを使用している Viewer (ビューアー) など) の影響を受ける可能性があります。

この場合、管理者はそれらのユーザーのライセンスを解除し、サイトロール クォータを更新することができます。ライセンス発行に関する問題のトラブルシューティング方法の詳細については、ライセンス発行のトラブルシューティングを参照してください。

ユーザーに対してリビジョン履歴の保存を許可する

リビジョン履歴では、ワークブックやデータ ソース (コンテンツ リソース) が時間の経過と共にどのように変更したかを表示できます。ユーザーがコンテンツ リソースを保存 (パブリッシュ) するたびに、Tableau Server は新しいバージョンを作成し、これが最新のバージョンとなります。これにより、以前のバージョンがリビジョン履歴で最新のリビジョンとなります。リビジョン履歴があり、ユーザーは古いバージョンが利用可能であるとわかっているため、自信を持ってコンテンツを試すことができます。

注

- この情報は Tableau Server に適用され、パブリッシャーにリビジョンの使用を許可する必要があるサーバー管理者に対するものです。
- Tableau Cloud では、ワークブックとデータ ソースのリビジョン履歴は、すべてのサイトで有効化されています。ユーザーは最大 10 のリビジョンを保存できます。
- 潜在的な問題を含め、コンテンツのリビジョンの操作に関する詳細については、Tableau ヘルプの「ユーザー/アナリスト」セクションの「コンテンツのリビジョンの操作」を参照してください。

ユーザーがリビジョン履歴で操作する必要があるパーミッション

リビジョン履歴にアクセスするには、ユーザーには **Creator** または **Explorer (パブリッシュ可能)** サイトロールに加え、コンテンツ タイプに応じ、次のパーミッションを付与されている必要があります。

- プロジェクト: **表示**および**保存**
- プロジェクト内のワークブック: **表示**、**保存**、**ワークブックのダウンロード/名前を付けて保存**
- プロジェクト内のフロー: **[表示] > [保存] > [フローのダウンロード/Publish As (名前を付けてパブリッシュ)]**
- プロジェクト内のデータ ソース: **表示**、**保存**および**データ ソースのダウンロード**

プロジェクト内の仮想接続には、**Creator** サイトロールと**表示**および**上書き**のパーミッションが必要です。(仮想接続には、データ管理が必要です。詳細については、「データ管理 について」を参照してください。)

リビジョン履歴を有効化し、許可されるリビジョンの数を設定する

リビジョン履歴はサイトレベルで設定され、既定で有効になっており、各コンテンツ リソースの履歴に保存されるリビジョンは 25 に制限されます。

1. サーバー管理者としてサイトにサインインし、**[設定]** を選択します。
2. **[リビジョン履歴]** で、**[リビジョン履歴の保存]** を選択してから、各コンテンツ リソースで許可するリビジョンの最大数を入力します。
3. **[保存]** をクリックします。

リビジョンの数を減らすと、最新のリビジョンが保存されます。たとえば、限度を 15 に設定すると、ワークブックやデータ ソースの最新のバージョン 15 個が保存されます。

注:サーバー上でバックグラウンド クリーンアップ プロセスが実行するまで、コンテンツ リソースのリビジョン履歴 リストに制限への変更が表示されない場合があります。

すべてのリビジョンをクリア

サーバー管理者は、サイトからパブリッシュされたワークブックとデータ ソースの以前のリビジョンをすべて削除できます。パブリッシュされた各ワークブックおよびデータ ソースの最新のバージョンが常に保持されます。

1. サーバー管理者としてサイトにサインインし、**[設定]** を選択します。
2. **[リビジョン履歴]** で、**[リビジョン履歴のクリア]** をクリックします。
3. **[保存]** をクリックします。

ワークブックのプレビューおよび復元に対するセキュリティ

ユーザーがワークブックのリビジョンの **[復元]** または **[プレビュー]** を選択すると、ユーザーのブラウザーとサーバーのバックエンドの間でユーザーのパスワードが交換されます。Tableau Server公開/非公開キー暗号化を使用してこれらのパスワードを暗号化します。これらの公開キーが Tableau Server によって提供されたことを確認するため、SSL (HTTPS) を使用するようサーバーを構成する必要があります。詳細については、SSLを参照してください。

関連項目

Tableau ヘルプの「ユーザー/アナリスト」セクションの「潜在的なリビジョン履歴の問題」

Tableau Server サイトで AI を有効にする

Tableau Server バージョン 2025.3 以降でサポートされています。

サイト設定で Web 作成と Tableau Prep 用に Tableau の AI が構成されて有効になっている場合、Tableau Server (すべてのエディション) に接続する際に Tableau Desktop、Tableau Prep Builder、Web 上で Tableau Agent を使用します。

Tableau Cloud で Tableau Agent を設定する方法については、「Tableau Cloud サイトで AI を有効にする」を参照してください。

Tableau の AI 機能を使用すると、データをより効果的に分析できます。Tableau Prep を使用している場合、AI はデータをクリーニングする手順を提案したり、計算を作成したりすることもできます。Viz を作成する際には、インサイトに満ちた質問を提案して、データの探索や計算の作成を行ったり、データセットから魅力的な Viz を直接作成することを支援したりできます。

Tableau Agent を使用した Viz の作成やデータのクリーニングについての詳細は、「Tableau Agent を使用したビューの構築とデータの探索」と「Tableau Agent を使用して Tableau Prep フローを構築する」を参照してください。

Tableau Server で Tableau の AI を有効にするには、独自の大規模言語モデル (LLM) プロバイダーに接続する必要があります。このリリースでは、サポートされている LLM プロバイダーは OpenAI です。

PII、データセキュリティ、多言語サポート

OpenAI プロバイダーと連携した Tableau の AI 機能を使用する場合、リクエストでは、Einstein Trust Layer や、組み込みの PII マスキングなどの関連するデータガバナンス機能を使用しません。Tableau は、ユーザーの質問に加えて、値の小さなランダム サンプルをフィールドごとに OpenAI アカウントに送信します。

このプロセスでは Einstein Trust Layer のセキュリティ機能と PII マスキング機能を使用しないので、機密データの保護のために、LLM プロバイダーのセキュリティ対策に依存することになります。詳細については、「Tableau の AI と信頼」を参照してください。

課金に関する考慮事項

Tableau Server で Tableau Agent に接続する場合は、OpenAI の独自のインスタンスを使用します。データはこの専用接続を介して流れるため、リクエストは Einstein Trust Layer を経由せず、関連する請求も発生しません。使用量に対しては、特定のプロバイダー契約に従って、OpenAI から直接請求されます。

Tableau Server で AI を有効にするための前提条件

Tableau Server で Tableau Agent を構成して有効にするには、次の条件を満たす必要があります。

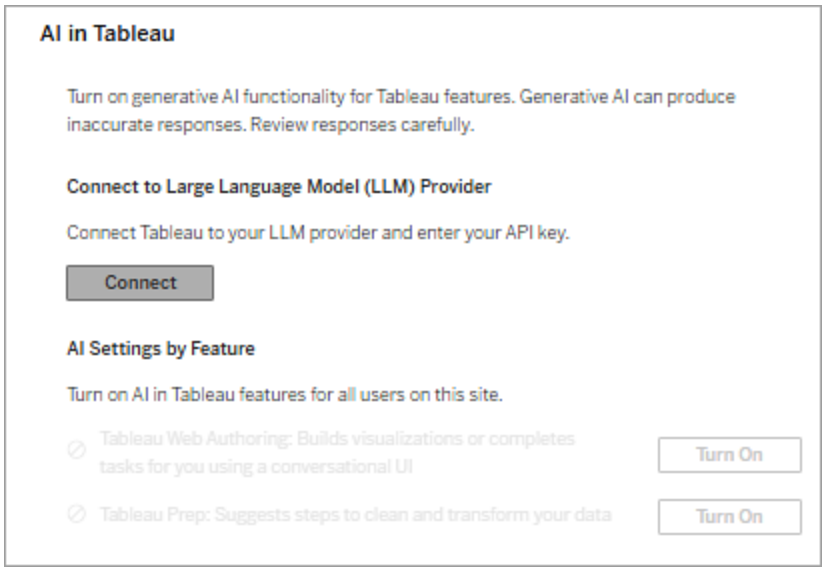
- Tableau Server の管理者ロール。Tableau の AI のサイト設定はサイト固有であり、オンにすると、そのサイトにアクセスできるすべてのユーザーが AI 機能を使用できるようになります。
- OpenAI との有効な契約。
- OpenAI API キー。これは、Tableau Server を OpenAI LLM プロバイダーに接続するために必要になります。セキュリティに関する標準的なベストプラクティスに従い、API キーを定期的にローテーションすることをお勧めします。
- ユーザーのリクエストを理解したり、関連する出力を返したりするといったタスクを完了するために Tableau Agent が使用する OpenAI モデル。Tableau では、最適な結果を得るため、

次のモデルが推奨されています。

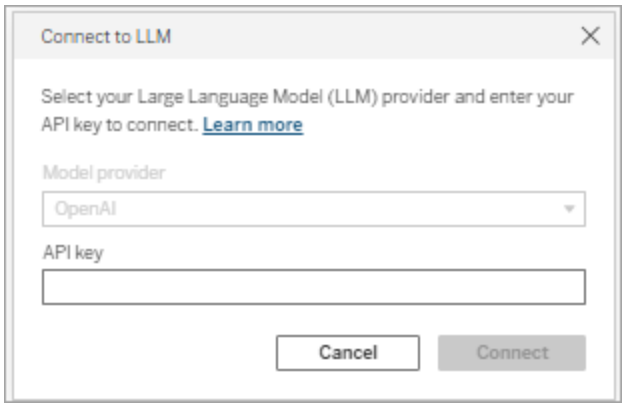
- [Understand user requests (ユーザーのリクエストの理解)]: gpt-4o-mini
- [Create visualizations, calculated fields, and build flows (Viz および計算フィールドの作成とフローの構築)]: gpt-4o-2024-11-20
- [Process data context (データコンテキストの処理)]: text-embedding-3-small

Tableau の AI を設定する

1. Tableau Server にサーバー管理者としてサインインします。
2. 生成 AI 機能を有効にするサイトで、**[設定]** を選択します。
3. [Tableau の AI] で、**[接続]** をクリックします。



4. [LLM に接続] ダイアログで、LLM プロバイダーの API キーを入力し、**[接続]** をクリックします。現在、サポートされている LLM プロバイダーは OpenAI のみです。



プロバイダーに正常に接続した後、必要に応じて、接続を編集、テスト、削除することもできます。

5. Tableau の有効にする AI 機能ごとに、**[オンにする]** を選択します。以下の機能から選択できます。

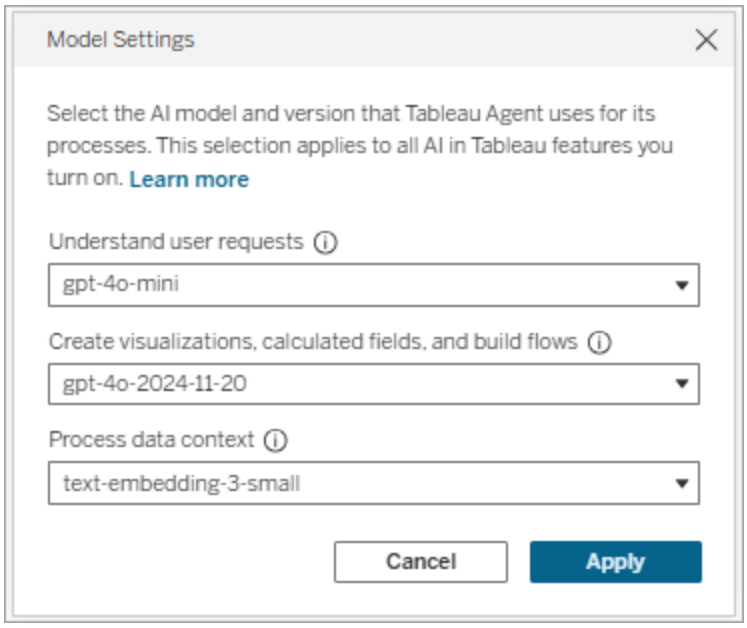
   - **[Tableau Web 作成: 対話形式の UI を使用して、Viz を作成したり、タスクを完了したりします]**: Tableau Server Web 作成および Tableau Desktop (バージョン 2025.3 以降) において、ユーザーが Tableau Agent の支援を受けて、データを探索したり、Viz や計算を作成したり、インサイトを発見したりできるようになります。Tableau Agent の詳細については、「Tableau Agent でデータを探索する」を参照してください。

   - **[Tableau Prep: データのクリーニングと変換の手順を提案します]**: ユーザーが Tableau Agent の支援を受けて、データのクリーニング、計算フィールドの作成、フローの構築を行えるようになります。この機能は、Tableau Server の Tableau Prep Web 作成および Tableau Prep Builder (バージョン 2025.3 以降) で使用できます。Tableau Prep の Tableau Agent の詳細については、「Tableau Agent で Tableau Prep フローを作成する」を参照してください。

6. Tableau Agent がプロセスを完了するために使用する AI モデルを選択します。

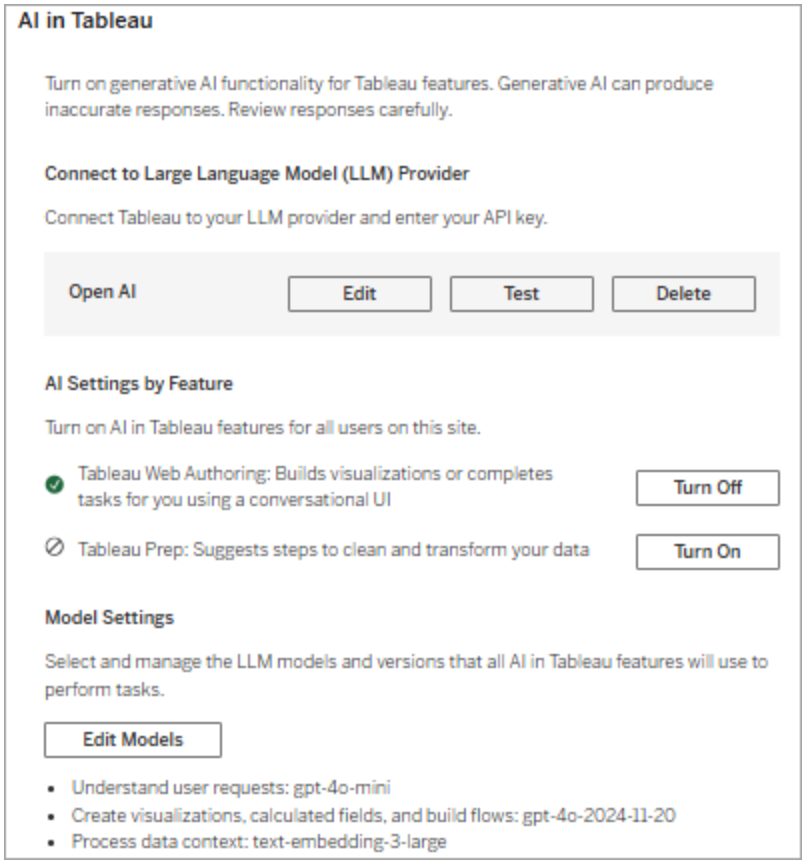
- この設定は普遍的であり、Tableau の有効にする AI 機能すべてに適用されます。ダイアログで、すべてのプロセスに対してモデルを選択する必要があります。
- **Web 作成用の Tableau Agent**: すべてのプロセスで指定されたモデルを使用します。
- **Prep 用の Tableau Agent**: **[Create visualizations, calculated fields, and build flows (Viz および計算フィールドの作成とフローの構築)]** プロセスで指定されたモデルのみを使用します。

モデルが OpenAI でサポートされなくなった場合、ユーザーは Tableau Agent の使用時にエラーを受け取ります。現在、この設定は手動で更新する必要があります。



次のモデルを選択して、**[保存]** をクリックします。Tableau では、最適なパフォーマンスを得るため、次のモデルが推奨されています。

- [Understand user requests (ユーザーのリクエストの理解)]: gpt-4o-mini
- [Create visualizations, calculated fields, and build flows (Viz および計算フィールドの作成とフローの構築)]: gpt-4o-2024-11-20
- [Process data context (データ コンテキストの処理)]: text-embedding-3-small



7. ページの上部にある **[保存]** をクリックします。

Tableau Mobile アプリのセキュリティ設定

Tableau Mobile アプリの Tableau Server セキュリティ ポリシーを調整します。これらのポリシーは、モバイル デバイスが侵害されているかどうかを確認し、Tableau Mobile アプリとの特定のやりとりを制限することで、データを安全に保ちます。

このポリシーが適用されるのは、MAM バージョンのアプリではなく、標準バージョンの Tableau Mobile のみです。MAM アプリをデプロイしている場合は、MAM システムの設定に加えて、Tableau Mobile に固有の を使用してアプリを保護します。

セキュリティ設定

Tableau Server のサイト設定ページ、または REST API を使用して、Tableau Mobile のセキュリティ設定を構成します。REST API の詳細については、Tableau REST API ヘルプの「モバイル設定のリファレンス」を参照してください。

サイトの設定ページに移動するには、以下の操作を行います。

1. 管理者として Tableau Server サイトにサインインします。
2. ナビゲーション ペインで **[設定]** を選択します。
3. **[モバイル]** タブを選択します。

より、モバイル デバイスのセキュリティに関連する Server 2023.1 の設定が利用できるようになりました。

これらの設定には、次の条件の検出が含まれます。

- **ジェイルブレイクの検出**

  この設定は、**重大** レベルではデフォルトで有効になっており、ジェイルブレイクされたり、ルート化されたりしたデバイスでアプリが実行されていないか検出します。

- **マルウェア検出** (Android デバイスのみ)

  この設定は、**重大** レベルではデフォルトで有効になっており、デバイスにマルウェアが存在しないか検出します。

  2025 年 9 月に、Tableau Mobile はマルウェア検出設定のサポートを終了します。マルウェア検出設定は、2025 年 9 月の Tableau Cloud 設定ページと、Tableau Server バージョン 2025.2.3 の Tableau Server 設定ページから削除されました。Tableau Server の以前のバージョンでマルウェア検出を設定しても、Tableau Mobile アプリに影響しません。

- **ポリシーの更新を行わないオフラインの最大日数**

この設定は、デフォルトで**重大**レベルで有効になっており、既定の最大値は 14 日です。この設定によって、設定された最大値よりも長い期間オフラインになっている(つまり、ポリシーが更新されていない) デバイスでアプリを使用できるかどうかが決定されます。

- **デバッグの防止**

  この設定は、既定でオンになっており、無効にすることはできません。デバイスにデバッガーが添付されているかどうかを検出します。

- **画面共有とスクリーンショット**(Android デバイスのみ)

  この設定はデフォルトで有効になっており、Tableau Mobile アプリユーザーがスクリーンショットを共有できるか、アプリで画面共有を使用できるかを確認します。

ジェイルブレイク検出の設定では、重大度のレベルを変更できます。

- **警告**: ポリシーを適用し、不合格になった場合は、解除可能なブロック メッセージを表示します。
- **エラー**: ポリシーを適用し、不合格になった場合は、問題が解決されるまでブロック メッセージを表示します。
- **重大**: ポリシーを適用し、不合格になった場合はブロック メッセージを表示し、アプリはプロバイダーを介してログアウトまたはセッション解除を処理する方法を決定します。これはデフォルトの設定です。

抽出更新スケジュール

Tableau Desktop の作成者およびデータ案内人は、*抽出*を作成およびパブリッシュできます。抽出は、元のデータのコピーまたはサブセットです。抽出はデータエンジンにインポートされるため、抽出に接続するワークブックは一般的に、ライブ データへ接続するワークブックより早く実行できます。また、抽出によって機能性も向上します。

抽出を更新する前に

抽出の更新が Tableau 10.4 以前で作成された抽出 (つまり、.tde 抽出) で実行されると、抽出は自動的に .hyper 抽出にアップグレードされます。.hyper 抽出にアップグレードする利点は多数ありますが、Tableau Desktop の以前のバージョンでは抽出を開けなくなります。Tableau 2024.2

は、.tde ベースのコンテンツを開くことができる最後のバージョンです。詳細については、.hyper 形式への抽出のアップグレードを参照してください。

更新スケジュールの設定

サーバー管理者は、抽出の更新タスクのスケジュールを有効化でき、その後、スケジュールの作成、変更、再割り当てができます。管理者がサーバー上で変更する一般的なスケジュールのオプションは、Tableau Desktop ユーザーが抽出をパブリッシュするときのパブリッシュプロセスの一部として利用できます。

作成されたスケジュールには次のオプションがあります。

優先度

優先度は、更新タスクが実行される順序を決定します。0 が最高優先度、100 が最低優先度を指します。優先度はデフォルトで 50 にセットされています。

実行モード

実行モードは、Tableau Server バックグラウンダープロセスに対し、更新を並列または連続で行うかどうかを示します。並列で実行するスケジュールでは使用可能なすべてのバックグラウンダープロセスを使用し、シリアル スケジュールは 1 つのバックグラウンダープロセスでのみ実行されます。ただし、スケジュールには 1 つ以上の更新タスクを含めることができ、各タスクでは並列モードかシリアルモードかにかかわらず、1 つのバックグラウンダープロセスのみを使用します。つまり、並列実行モードのスケジュールでは、**使用可能な**すべてのバックグラウンダープロセスを使用してその下にあるタスクを並列で実行しますが、各タスクでは 1 つのバックグラウンダープロセスのみを使用します。シリアルスケジュールでは、1 つのバックグラウンダープロセスのみを使用して一度に 1 タスクのみを実行します。

既定では、更新タスクをできるだけ速く終わらせるために、実行モードは "パラレル" にセットされています。大規模なスケジュールにより他のスケジュールの実行できない場合、実行モードを「連続」(および低い優先度) に設定することがあります。

頻度

頻度を毎時、毎日、毎週、または毎月にセットできます。

スケジュールの作成または変更を参照してください。

手動での抽出の更新

Tableau Server の Web 環境では、サーバーとサイトの両方の管理者は、**[スケジュール]** ページで、抽出の更新をオンデマンドで次のように実行できます。

- スケジュールを選択し、**[アクション]**、**[今すぐ実行]** の順にクリックします。

また、tabcmd `refreshextracts` コマンドで、コマンド ラインから抽出の更新を行うことができます。詳細については、tabcmd コマンドを参照してください。

Tableau Desktop からの抽出の更新

Tableau Desktop ユーザーはパブリッシュおよび所有する抽出を更新できます。これは、次の方法で実行できます。

- **パブリッシュ時:** 作成者は抽出を使用するワークブックまたはデータ ソースをパブリッシュするとき、サーバー更新スケジュールを追加することができます。更新には、完全更新と増分更新があります。

  増分更新は、日付、日付/時刻、またはタイムスタンプなど整数のデータ型がある抽出の列を参照します。Tableau はこの列を使用して、抽出に追加する必要がある新しい行を識別します。詳細については、Tableau ヘルプの「抽出の更新」および「ワークブックをパブリッシュしながら抽出の更新をスケジュールする」を参照してください。

- **ユーザー インターフェイス:** Tableau Desktop の **[ソースから更新]**、**[ファイルからデータを追加]**、および **[データ ソースからデータを追加]** コマンドを使用すると、Tableau Server への追加をアップロードしたり、抽出を更新したりできます。Tableau Server に参照元データへのアクセスに必要な十分な認証資格情報がない場合、ユーザーがこれを行うことができます。詳細については、Tableau ヘルプの「Tableau Server での抽出の更新」を参照してください。

- **データ抽出 コマンド ライン ユーティリティ:** データ抽出 コマンド ライン ユーティリティは、Tableau Desktop と共にインストールします。パブリッシュする抽出の追加や更新にこれを

使用できます。詳細については、Tableau ヘルプの「Tableau Data Extract コマンド ライン ユーティリティ」を参照してください。

抽出更新スケジュールと失敗通知を有効にする

パブリッシャーは、次の 2 つの条件が満たされたときに抽出更新をスケジュールできます。

- Tableau Server が、抽出更新が失敗したときにメール メッセージを送信するように構成されています。これは Tableau サービス マネージャー (TSM) 管理者によって構成され、既定でオンになっています。詳細については、サーバー イベント通知の構成を参照してください。
- パブリッシャーが抽出更新をスケジュールできるように設定するサイトが、更新の失敗時にメールを送信するように構成されています。これは Tableau Server のサーバー管理者によって構成され、既定でオンになっています。以下の手順では、これが有効になっていない場合の方法について説明します。

スケジュールを有効にすると同時に、抽出更新が正常に完了しなかった場合に更新されるデータソースやワークブックの所有者へのメール送信を有効にするかどうかを決定できます。これらのメールの詳細については、以下をお読みください。更新失敗通知を有効にした場合、スケジュールされた更新が行われるコンテンツの所有者はアカウント設定を変更して個別にオプトアウトできます。

1. Tableau Server にサーバー管理者としてサインインします。
2. サブスクリプション用に構成するサイトの [設定] ページの [全般] タブに移動します。
   - サイトが 1 つの場合は、ブラウザー ウィンドウの上部にある **[設定]** をクリックしてから、**[全般]** をクリックします。
   - 複数のサイトがある場合は、構成するサイトを選択してから、**[設定]** をクリックし、**[全般]** をクリックします。
3. **[全般]** ページで、次の手順を実行します。
   - **[通知の管理]** 設定までスクロールし、**[ジョブの抽出]** をオンにします。

     特定のデータソースのスケジュールされた更新が失敗した場合、メールはそのデータソースの所有者のみに送信され、そのデータソースに接続するワークブックの所有者には送信されません。

- **[埋め込み認証資格情報]** で両方のオプションを選択し、パブリッシャーが認証資格情報の埋め込みと抽出更新のスケジュールを行うようにします。(自動更新スケジュールで埋め込み認証資格情報が必要なため、Tableau Server がデータに直接アクセスできます)。

  > **注:** マルチサイトサーバーでは、失敗の通知はサイト設定、埋め込み認証資格情報はサーバー設定です。マルチサイトサーバーで埋め込み認証資格情報を構成するには、**[すべてのサイトを管理]** を選択し、**[設定]** を選択します。

サーバーからスケジュールを管理する

組織にとっては、埋め込み認証資格情報と更新スケジュールをサーバーから一元的に管理する方が、より適している場合があります。その場合、パブリッシュ中 Tableau Desktop パブリッシャーにスケジュール オプションが表示されないよう、上記の手順の **[埋め込み認証資格情報]** セクションでチェック ボックスをオフにすることができます。

スケジュールを一元的に管理することで、正確な更新の抽出とサブスクリプションのタスクを分散できるため、ほとんどのユーザーがオフラインのときにこれらのタスクを実行できます。さらに、どの認証資格情報を接続に埋め込むかをより細かく制御できます。

更新失敗メールのしくみ

失敗した抽出の更新のメール通知には、抽出名とサーバーの場所、前回の成功した更新の時刻、更新に連続して失敗した回数、障害の理由と考えられる解決策を提案します。

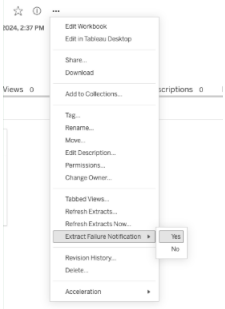
5 回連続して失敗した後、ユーザーが障害の原因を解決する措置 (データベース認証資格情報や、元のデータファイルへのパスの更新など) を講じるまで、更新は一時停止されます。

管理者以外のユーザーへの更新失敗メールを有効にする

> **注:** このオプションは、Bridge で作成または更新されているパブリッシュされたデータソースでは使用できません。

既定では、抽出更新に問題がある場合、管理者のみが通知を受け取ります。ただし、オプションで、抽出が失敗したときに通知を受け取ることを選択できます。

抽出が失敗したときのメール通知を有効にするには、抽出ダッシュボードのコンテキストのメニュー (...) で **[抽出失敗通知]** を選択し、通知を受け取るには **[はい]**、通知を停止するには **[いいえ]** を選択します。



> **注**: 現在のユーザーが抽出の所有者である場合、コンテキストメニューを使用して、失敗通知をオプトインまたはオプトアウトすることはできません。

前回成功した更新日の決定方法

前回の成功した更新の日時は、前回の更新がある日数以内に発生した場合に表示されます。既定では 14 日であり、この値は `wgserver.alerts.observed_days` に設定されます。前回更新に成功してからの日数がこの設定で指定された数値を超えている場合、メールのメッセージに "not in the last *N* days (過去 N 日間更新されていません)" と表示されます。

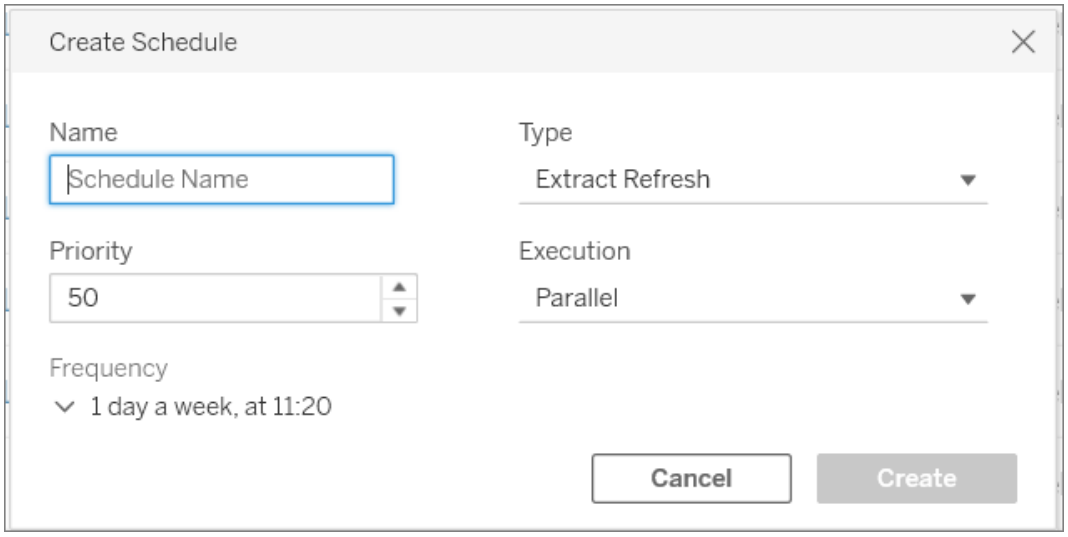
スケジュールの作成または変更

[スケジュール] ページには、Tableau Server 管理者のみがアクセスできます。スケジュールの名前、タイプ、目的 (範囲)、タスク数、動作 (並列処理や順次処理)、スケジュールされた実行時刻など、スケジュールのリストが表示されます。

注: サブスクリプションのカスタム スケジュールを 1 つ以上のサイトで有効化すると、ユーザーがそれらのサイトで自身のサブスクリプションのスケジュールを管理できるようになります。詳細については、サブスクリプションのカスタム スケジュールを可能にするを参照してください。

新しいスケジュールを作成するには

1. サイトで、**[スケジュール]** をクリックします。
2. **[新しいスケジュール]** をクリックします。

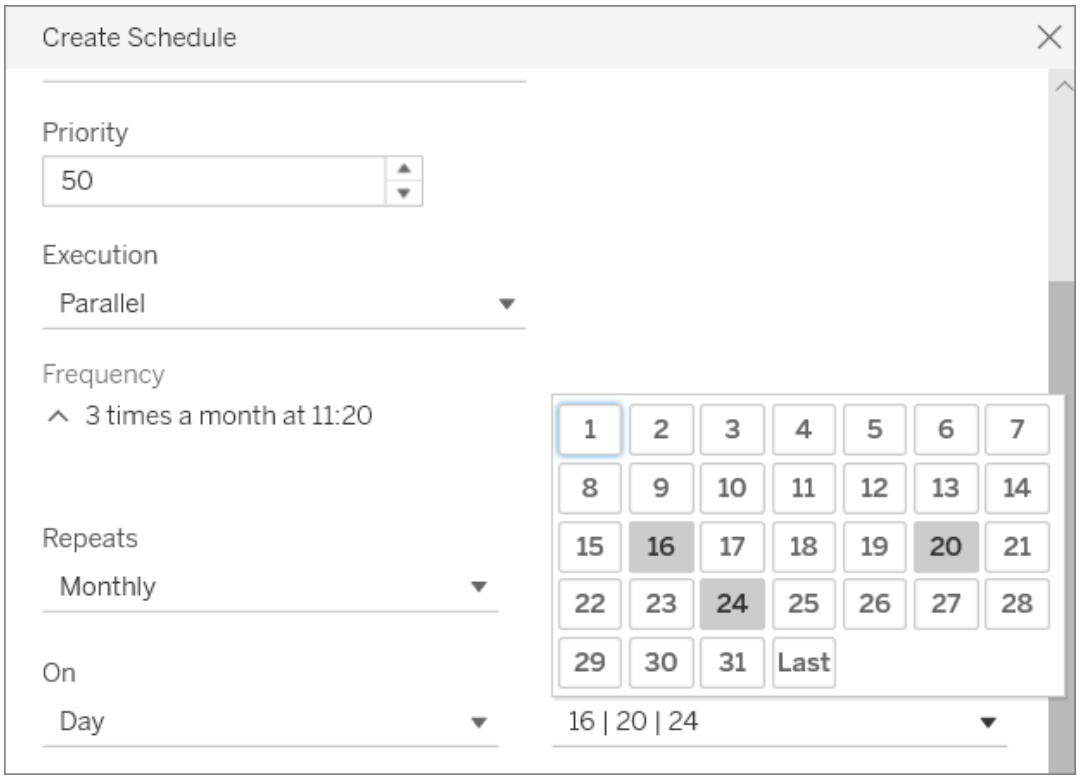


3. スケジュールに説明的な**名前**を指定します。
4. スケジュールが処理する **[タスクのタイプ]** で、抽出の更新、フローの実行、またはサブスクリプションの配信を選択します。
5. 優先度を 1 から 100 で定義できます。ここで、1 が最も高い優先度です。これは、既定でタスクに割り当てられる優先度です。キュー内で 2 つのタスクが保留中の場合、バックグラウンダーは抽出の更新とフローのタスクの優先度、サブスクリプションのスケジュール優先順位を評価して、どちらが最初に実行されるかを決定します。詳細については、「スケジュールされたサーバー ジョブの優先順位付け方法」を参照してください。

6. **[実行]**: スケジュールを並列的に実行するか、順に実行するかを選択します。並列的なスケジュールを選択すると、利用可能なすべてのバックグラウンダープロセスで実行されるため、完了がより迅速になります。

   > **注:** このオプションを並列に設定した場合でも、同じワークブックのスケジュールは常に順番に実行されます。

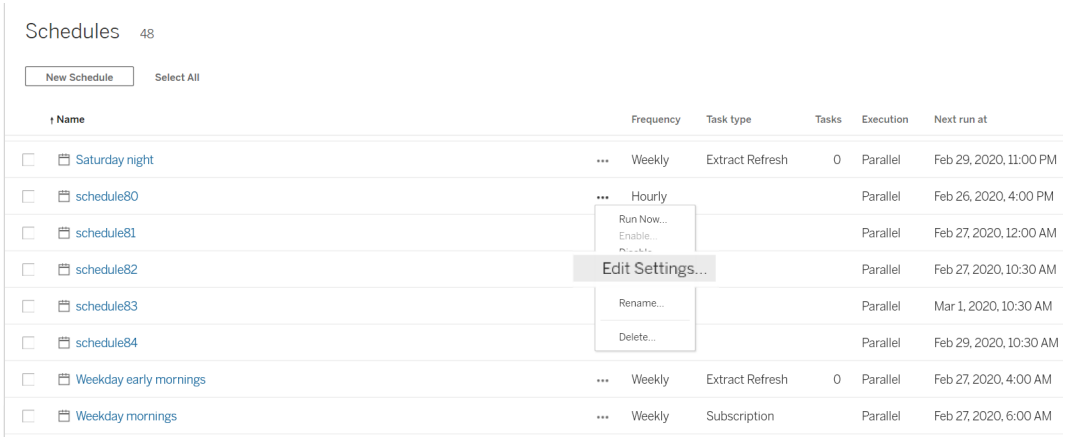
7. スケジュールの定義を終了します。毎時、毎日、毎週、または毎月行うようにスケジュールを定義できます。選択内容に基づいて、**[頻度]** が自動的に設定されます。



8. **[作成]** をクリックします。

既存のスケジュールを変更するには

1. [スケジュール] ページへ移動します。
2. 既存のスケジュールを選択し、[アクション] ドロップダウン矢印をクリックしてから、**[設定の編集]** を選択します。



3. スケジュールの編集を完了したら、**[保存]** をクリックします。

スケジュールの作成または変更のルール

新しいスケジュールを作成する場合や既存のスケジュールを変更する際に従う必要があるルールを次に示します。

注: これらのルールに従わない既存のスケジュールがある場合は、それに応じて変更する必要があります。これを行わない場合、予期しない動作が発生し、スケジュールされた時刻にタスクが実行されない可能性があります。

- 15 分または 30 分ごとに実行されるスケジュールには、時間の開始時刻と終了時刻が必要です。時間の例: 午前 5 時から午後 6 時まで。
- 定期的なアイテムの毎日のスケジュールは、開始と終了の分を同じにする必要があります。たとえば、午前 10 時 35 分から午後 4 時 35 分までです。時間は異なる場合がありま

す。ただし、1 日のスケジュールが 1 日 1 回のみ実行されるように設定されている場合は、開始時刻のみが必要で終了時刻は必要ありません。

関連項目

更新タスクの管理

抽出更新スケジュール

フロー タスクのスケジュール

サブスクリプションのカスタム スケジュールを可能にする

サブスクリプションのカスタム スケジュールにより、ユーザーはサーバー管理者が定義した固定スケジュールを使うのではなく、自身で定義したスケジュールに基づいてメール メッセージを受信できます。サブスクリプションのカスタム スケジュールは 2017 年 3 月依頼 Tableau Cloud のユーザーにご利用いただいています。これらのスケジュールは Tableau Server バージョン 2018.2 ではサイトごとベースで可能になりました。サブスクリプションのカスタム スケジュールを可能にすることは、この変更を行うサイトにおいて永久的な変更です。カスタム スケジュールを有効にしないサイトでは、サーバー管理者が定義した固定スケジュールが残ります。

カスタム スケジュールを有効にする

カスタム スケジュールを一つまたは複数のサイトで有効にする前に、まず Tableau Server でカスタム スケジュールを有効にして、次にそのサーバーの一つまたは複数のサイトでカスタム スケジュールを有効にします。Tableau Server でサブスクリプションを有効にする詳細については、サブスクリプションのサーバー設定を参照してください。

ステップ 1: Tableau Server でカスタム スケジュールを有効にする

Tableau 管理者パーミッションでコマンドプロンプトを開き、次のコマンドを実行します。

```
tsm configuration set -k features.SelfServiceSchedules -v true
tsm pending-changes apply
```

この操作は Tableau Server を再起動します。

ステップ 2: でカスタム スケジュールをサイトで有効にする

1. Web ブラウザーを使用して Tableau Server にサーバー管理者としてログインする。

   ```
   https://<hostname>/#/login
   ```

2. サイトの **[サイト設定]** ページを閲覧して、カスタム スケジュールを有効にします。

   1. **[全サイト]** をクリックし、ドロップダウンリストからサイトの一つを選択します。
   2. **[設定]** をクリックします。
   3. **[全般]** タブの、**[サブスクリプション]**の下で、以下のチェックボックスを選択します。**[管理者作成の固定 スケジュールからユーザー作成のカスタム スケジュールに永久に変換] (これは取り消しできません)**。
   4. **[保存]** をクリックします。

カスタム スケジュールがサイトで有効になると、サブスクリプションにある全固定スケジュールが、同等のカスタム スケジュールに変換されます。

スケジュールされたサーバー ジョブの優先順位付け方法

ジョブとタスク

Tableau Server では、定期的に実行される抽出更新、サブスクリプション、またはフローをユーザーがスケジュールできます。これらの**スケジュールされた項目**は**タスク**と呼ばれます。バックグラウンダープロセスは、これらのタスクの固有のインスタンスを開始して、スケジュールされた時刻に実行します。結果として開始された**タスクの固有インスタンス**は、**ジョブ**と呼ばれます。**[今すぐ実行]** オプションのクリックによって**手動で**実行が開始される**ジョブ**も作成されます。

たとえば、毎日午前 9 時に実行される抽出更新タスクを作成するとします。この抽出更新タスクについて、毎日午前 9 時にバックグラウンダーが実行するジョブが作成されます。

優先度番号をタスクとスケジュールに割り当てるには、1 から 100 の値を割り当てることができます。数値が小さく、優先度が高い、1 が最も高い優先順位となり、100 が最低となります。

ジョブの優先度ルール

スケジュールされた抽出更新、サブスクリプションおよびフロー実行を処理する場合、Tableau Server によってこの順序でバックグラウンド ジョブが優先順位付けされます。

1. 処理中のジョブは、最初に完了されます。

2. **[今すぐ実行]** を使用して手動で開始したタスクまたはスケジュールは、次のバックグラウンドプロセスが利用可能になると開始します。この例外は、フロー タスクとスケジュールです。フローの実行では、割り当てられたタスクの優先度を使用して、実行する順序を決定します。タスクの優先度が割り当てられていない場合、規定は 0 に設定され、これが最も高い優先順位になります。

   > 注: [今すぐ実行] 設定を制限する方法の詳細 - サーバー設定 (全般とカスタマイズ) を参照してください。

3. キュー内にいた時間の長さに関係なく、優先度が最も高い (数が最も小さい) ジョブが次に開始します。

   抽出更新とフローの場合、これがタスクの優先度です。タスクの優先度は、タスクが最初に作成されたときにスケジュールの優先順位から継承されます。タスクの優先度は後で変更できますが、データ ソースをもう一度パブリッシュすると、タスクの優先度は既定値に戻ります。

   サブスクリプションの場合、これはスケジュールの優先度です。サブスクリプションのカスタム スケジュールを有効にしている場合、それらのジョブの優先度は 50 に設定されます。

   たとえば、優先度が 20 のジョブは、優先度が 50 のジョブよりも先に実行されます (後者の方が待機時間が長い場合を含む)。タスクの優先順位を変更するには、スケジュールの作成または変更を参照してください。

4. 優先度が同じジョブは、キューに追加された順に実行されます。まずキューに最初に追加されたジョブを開始し、次に 2 番目のジョブを開始します。

5. 優先順位が同じ複数のジョブを同時に実行するようスケジュールされている場合、作成順または有効化された順序で開始します。同じ時間にスケジュールされたジョブは、最初に開始されるジョブの最も速いカテゴリーであるタスク タイプによって実行されます。フロー実行、データ駆動アラート、システム ジョブ、サブスクリプション、抽出作成、インクリメンタル抽出、

完全抽出の順で実行されます。

リンクされたタスクの一部として実行されるようにスケジュールされているフローは、すべて同じ優先順位に割り当てられ、リンクされたタスクで定義されている順序で実行されます。リンクされたタスクの詳細については、「リンクされたタスクのスケジュール」を参照してください。

6. 最終的な優先度を決める方法として、バックグラウンダーは実行時間履歴を使用します。以前の実行速度が速かったジョブは、実行に時間がかかっていたジョブよりも優先されます。

注: 特定のサイトでバックグラウンダーのリソース制限を設定すると、これらのジョブは他の優先度の高いジョブの後にピックアップされるため、キューが追加されることになります。詳細については、「Tableau Server Backgrounde のリソース制限」を参照してください。

ジョブの実行時には、次の制限も影響します。

- 同時ジョブの数は、Tableau Server に設定したバックグラウンダープロセスの数に制限されます。
- 同じ抽出またはデータソースに対する別の更新を同時に実行することはできません。
- 並列に実行するよう設定されたスケジュールと関連付けられたジョブは、1 度に 1 つのみ実行します。

スケジュールされた更新後のワークブックパフォーマンスの設定

ワークブックの読み込み時間を改善するため、Tableau Server はワークブックに含まれるクエリの結果をキャッシュします。大半のワークブックでは、クエリ結果は Tableau Server 上でユーザーによって最初に表示されたときに計算およびキャッシュされます。ただし、データ抽出に接続しているワークブックの場合、Tableau Server は、対応する抽出が更新タスクを実行したときに、クエリ結果を再計算できます。これによって、ワークブックを最初に表示したときのロード時間が短縮されます。最近表示したワークブックにはこのオプションが既定で選択されています。

重要! 外部クエリのキャッシュのウォームアップは、バージョン 2023.1 で非推奨になります。Tableau Server ワークブックのビューの読み込み時間を改善するには、代わりにサイト上でビューの高速化を許可する必要があります。詳細については、「ビューの高速化」を参照してください。

パフォーマンスへの影響に対する判断

このオプションによってワークブックの初期ロード時間は短縮されますが、クエリが再計算されるため、Tableau Server のロード時間は増大します。Tableau Server のインストール時に既にパフォーマンスが制限されている場合は、このオプションを選択解除するか、ワークブックのキャッシュに必要なしきい値を下げる必要があります。

状況によっては、このオプションを無効にしたり、しきい値を下げることを考慮することをお勧めします。次に、その理由をいくつか挙げます。

- 抽出以外のバックグラウンドタスク管理ビューの **[データ変更時に外部クエリのキャッシュを準備]** カテゴリーに、長期実行中のクエリが多数表示される。
- バックグラウンドタスクの遅延管理ビューに、長時間の遅延が表示される。
- バックグラウンダープロセスで使用される CPU およびメモリ消費量は一貫して高い。

ただし、これはバックグラウンドタスクのパフォーマンスに影響を与えるオプションの 1 つに過ぎません。パフォーマンスの詳細については、パフォーマンスを参照してください。

サーバーでのワークブックのキャッシュ停止

Tableau Server のロード時間を短縮するために、サーバーレベルでスケジュールされた更新の後に実行するワークブックのキャッシュを停止できます。このオプションを選択解除すると、最初にワークブックが表示されるときに Tableau Server でワークブックのクエリ結果がキャッシュされます。

次の tsm configuration set オプションを使用して、スケジュールされた更新後のワークブックのキャッシュを停止します。

```
backgrounder.externalquerycachewarmup.enabled
```

tsm set オプションの使用と適用の方法の詳細については、tsm configuration set のオプションを参照してください。

サイトでのワークブックのキャッシュ停止

ワークブックのキャッシュは、個別のサイトのスケジュールされた更新後に停止することもできます。たとえば、サーバーのロード時間が増大し、動きの遅くなったワークブックが多数含まれているサイトがある場合に実行します。

1. サイトのドロップダウンで、ワークブックのキャッシュを停止するサイトを選択します。
2. **[設定]** をクリックします。
3. **[スケジュールされた更新後のワークブック パフォーマンス]** セクションで、チェック ボックスをオフにします。

注:このオプションは個別のサイトの設定時に使用できますが、表示するにはサーバー管理者パーミッションが必要です。

ワークブックのキャッシュに必要なしきい値の設定

Tableau Server では、スケジュールされた更新のタスクが含まれるワークブック、および最近表示したワークブックの両方のクエリ結果のみが再計算されます。

次の tsm configuration set オプションを使用して、スケジュールされた更新後にキャッシュされるワークブックの数を増減できます。

```
backgrounder.externalquerycachewarmup.view_threshold
```

既定では、しきい値は 2.0 に設定されています。しきい値は、ワークブックが過去 7 日間で受信したビューの数を今後 7 日間にスケジュールされている更新の数で割った数と等しくなります。(過去 7 日間にワークブックが表示されていない場合はすぐに表示されない可能性があります。Tableau Server でワークブックのクエリを再計算するためにリソースを使用することはありません)。

サブスクリプションおよびデータ主導アラートへのアクセスの確保

ユーザーが Tableau Server ツールバーで [サブスクライブ] ボタンおよび [アラート] ボタンを表示し、関連するメールを受信できるようにするには、次を行います。

- **Tableau Server での SMTP 通知とイベント通知の構成**: サブスクリプションのサーバー設定」を参照してください。
- **ユーザーが Tableau Server の電子メール アドレスを持っていることを確認する**: ユーザーは、[アカウント設定] ページで電子メール アドレスを更新できます。
- **データベース認証資格情報を埋め込む、または要求しない**:ビューにデータをメールで送信するには、Tableau Server がユーザーの関与なしにデータにアクセスできる必要があります。これは、埋め込まれたデータベースのクレデンシャルを使用するワークブック、Tableau Server データ ソース、または認証資格情報を必要としないデータ(パブリッシュ時にワークブックに含まれるファイルなど) を使用することで達成できます。
- **ユーザーが必要なワークブックおよびビューにアクセスできるようにする**: サーバー上のワークブックやビューへのアクセスは、**[表示]** パーミッションによって制御されます。メール メッセージのコンテンツのイメージを受信するには、**[画像/PDF のダウンロード]** パーミッションも必要です。詳細については、パーミッションを参照してください。
- **埋め込みビューでは信頼できる認証の使用を避ける**:制限付きのチケット(既定) を使用して埋め込みビューのレンダリングを行っている場合、[サブスクライブ] ボタンおよび [アラート] ボタンは表示されません。

  (アラートのみ) ユーザーが表示および接続機能を使用して、パブリッシュされたデータソースにアクセスできることを確認します。

サブスクリプションのサーバー設定

ユーザーがワークブックやビューをサブスクライブすると、スケジュールに基づいてビューのスナップショットがそのユーザーにメールで送信されるため、ユーザーは Tableau Server にサインインしなくても最新の更新を確認できます。管理者、適切なサイト ロールを持つプロジェクト リーダー、およびコンテンツ所有者には、他のユーザーのワークブックやビューをサブスクライブできるオプションがあります。詳細については、「ビューのサブスクライブ」を参照してください。

注: サブスクリプションを作成および受信するには、関連するデータベースとビューへのアクセス権が必要です。詳細については、この要件のリストを参照してください。

Tableau Server on Windows については、「サブスクリプションのサーバー設定」を参照してください。

前提条件: サブスクリプション メールを送信するようサーバーを構成する

サブスクリプションをサイトに対して有効にする前に、サーバーでサブスクリプションを有効にする手順を完了する必要があります。サーバーでサブスクリプションを構成するには、以下のトピックの手順に従います。

1. SMTP セットアップの構成
2. サーバー イベント通知の構成

サブスクリプションを有効にする

SMTP とサーバー イベント通知を構成した後、サブスクリプションを有効にできます。

サブスクリプションを有効にするには、次のようにします。

1. Tableau Server にサーバー管理者としてサインインします。
2. サブスクリプション用に構成するサイトの [設定] ページの [全般] タブに移動します。
    - サイトが 1 つである場合は、サイド ナビゲーションで **[設定]** をクリックし、**[全般]** をクリックします。
    - 複数のサイトがある場合は、構成するサイトを選択してから、**[設定]** をクリックし、**[全般]** をクリックします。
3. **[サブスクリプション]** までスクロールし、ユーザーのサブスクリプション オプションを選択します。

> **注:** サブスクリプション オプションは、TSM 管理者がサーバー全体の構成オプション **[ユーザーがサブスクライブしたビューの電子メールを受信することを許可する]** を有効にした後にのみ表示されます。詳細については、サーバー イベント通知の構成を参照してください。

a. **[ユーザーにワークブックおよびビューのサブスクライブを許可する]** を選択します

b. (オプション) コンテンツを他のユーザーがサブスクライブするようにコンテンツ所有者が設定することを許可するには、**[コンテンツ所有者に対し、他のユーザーによるサブスクライブを許可する]** を選択します。

c. (オプション) ユーザーがサブスクリプションに添付ファイルを含めることを許可するには、**[サブスクライブしているワークブックおよびビューに添付ファイルを追加できるようにする]** を選択します。TSM 管理者が TSM で添付ファイルを有効にしていない場合、このオプションは使用できません。詳細については、サーバー イベント通知の構成を参照してください。

4. (オプション) **[メール設定]** までスクロールします。

a. **[メール送信元アドレス]** に、メール メッセージで "送信元" アドレスとして表示されるメール アドレスを入力します。

b. **[メール フッター]** に、メール メッセージのメール フッターを入力します。

サイトの "送信元" データ主導アラートのメールでは、アドレスまたはメッセージ フッターも使用されます。

5. (オプション) **[ビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベル]** までスクロールし、**[ビューとワークブックのサブスクリプション メールに、視認性の高い品質に関する警告と視認性の高い機密度ラベルを含める]** を選択します (以前のバージョンでは、**[サブスクリプションのデータ品質に関する警告]** までスクロールし、**[サブスクリプション メールにデータ品質に関する警告を含める]** を選択します)。

> **注:** サブスクリプション メールのデータ品質に関する警告は、Tableau Catalog が有効になっている場合にのみ表示されます。詳細については、Tableau Catalog の有効化を参照してください。

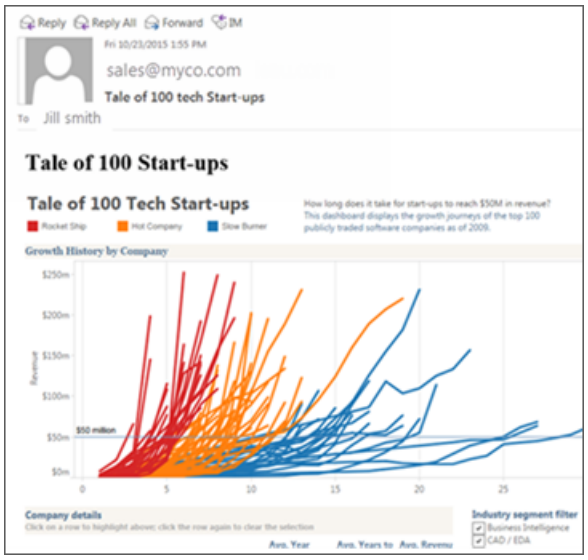
6. **[保存]** をクリックします。

ユーザーが使用できるサブスクリプション スケジュールを指定するには、「スケジュールの作成または変更」を参照してください。

サイトでのサブスクリプションのテスト

1. ビューをサブスクライブします。
2. テストするサブスクリプションが含まれるサイトのサイド ナビゲーションで、**[スケジュール]** をクリックします。
3. サブスクリプションに選択したスケジュールを選択してから、**[アクション] > [今すぐ実行]** をクリックします。

   ビューのスナップショットは 10 分以内にメールで送信されます。問題が発生した場合は、サブスクリプションのトラブルシューティングを参照してください。



すべてのユーザー サブスクリプションの管理

1. サイドナビゲーションで、**[タスク]** をクリックし、**[サブスクリプション]** をクリックします。

   現在のサイトのすべてのユーザー サブスクリプション (サブスクライバー名、ビュー名、配信スケジュールなどの情報を含む) が表示されます。

2. 更新するサブスクリプションを選択します。**[アクション]** メニューから、**[スケジュールの変更]**、**[件名の変更]**、**[空ビュー モードの変更]** または **[サブスクリプションの解除]** を選択します。

   (空ビュー オプションは、ビュー内にデータが存在する場合にのみサブスクリプション メールを送信します。高優先順位アラートに適した選択です。)

サブスクリプションの一時停止

既定では、サブスクリプションは連続 5 回失敗すると一時停止し、サブスクリプション メールが送信されなくなります。サブスクリプションが一時停止する前に発生するサブスクリプションの失敗回数のしきい値を変更するには、tsm configuration set のオプションである backgrounder.subscription_failure_threshold_for_run_prevention を使用します。これにより、サブスクリプションが一時停止するために必要となる連続したサブスクリプションの失敗回数のしきい値が設定されます。これはサーバー全体の設定です。

サーバー管理者のみが、サブスクリプションが一時停止するためのサブスクリプションの失敗回数のしきい値を構成することができます。

サーバー管理者は、サブスクリプションが一時停止となる場合にメール通知の受信をオプトインすることができます。これを設定するには、**[マイ アカウントの設定] -> [サブスクリプションの通知]** に移動します。この設定はサイトレベルで実行されるため、サイトごとに構成する必要があります。

一時停止したサブスクリプションの再開

サブスクリプションの失敗が 5 回を超えると、サブスクリプションが一時停止した旨の通知メールを受信します。サブスクリプション所有者または管理者は、いくつかの方法で、一時停止したサブスクリプションを再開することができます。

- Tableau Web ページの **[マイ コンテンツ]** エリアの [最終更新] 列に、サブスクリプションが一時停止していることを示すアイコンが表示されます。**[...] > [サブスクリプションの再開]** を選択して再開します。
- 影響が出ているワークブックの **[サブスクリプション]** タブで、[前回更新] 列に、サブスクリプションが一時停止になっていることを示すアイコンが表示されます。**[...]> [サブスクリプションの再開]** を選択して再開します。
- **[タスク]** の **[サブスクリプション]** タブで、[前回更新] 列に、サブスクリプションが一時停止になっていることを示すアイコンが表示されます。**[...]> [サブスクリプションの再開]** を選択して再開します (サーバー管理者のみ)。

サブスクリプションが再開されると、アラートの失敗回数が 0 に戻ります。サブスクリプションの次の評価は、スケジュールされている次回評価時に行われます。

関連項目

Tableau Desktop と Web 作成のヘルプの「ビューのサブスクライブ」。

プロジェクトレベルの管理では、どのサイトロールで完全なプロジェクトリーダー権限が許可されるかを確認できます。

データ主導アラートの設定

データがビジネスにとって重要なしきい値に達すると、データ主導アラートが自動的にユーザーが指定した主要な人々にメール通知を送信します。Tableau Server 管理者は、サブスクリプションの場合と同様にデータ主導アラートを設定します。ユーザーがこれらのアラートを作成および管理する方法の詳細については、Tableau ユーザーヘルプの「データ主導アラートの送信」を参照してください。

> 注:データ主導アラートを作成および受信するには、関連するデータベースとビューへのアクセス権が必要です。詳細については、この要件のリストを参照してください。サイトに対してアラートが有効である場合、そのサイト上のユーザー (Viewer ロールのユーザー以外) はアラートを作成できます。

データ主導アラートのメールの設定

1. 「SMTP セットアップの構成」のステップに従い、サーバーがメールを送信できるようにします。
2. サイトを表示した状態で、ブラウザー ウィンドウの最上部にある **[設定]** をクリックします。
3. [メール設定] の下に、サイト固有の "送信元" アドレスまたは メッセージ フッターを入力します。

   サイトの "送信元" アドレスとメッセージ フッターは、サブスクリプション用のメールでも使用されます。
4. **[保存]** をクリックします。

1 つのサイトですべてのデータ主導アラートを管理する

1. ブラウザー ウィンドウの最上部で、**[タスク]**、**[アラート]** の順にクリックします。
2. 更新するアラートを選択します。
3. **[アクション]** メニューから以下のいずれかを実行します。
   - 自身を受信者として追加または削除する。
   - アラートを編集して、データしきい値、配信スケジュール、および受信者のフル リストを変更する。
   - アラートの所有権を別のユーザーに変更する、またはアラートを削除する。

サイトのデータ主導アラートの無効化

既定ではデータ主導アラートはすべてのサイトでサポートされていますが、管理者は特定のサイトでそれらを無効にすることができます。

1. サイトを表示した状態で、左側のナビゲーション ペインにある **[設定]** をクリックします。
2. [データ主導アラート] の下で、**[ユーザーに警告の作成と警告メールの送信を許可する]** をオフにします。
3. **[保存]** をクリックします。

データ主導アラートの一時停止

既定では、連続して 350 回アラートが失敗するとアラートが一時停止します。サーバー管理者は、アラートが一時停止する前のアラートの失敗回数のしきい値を設定することができます。アラートが一時停止する前に発生するアラートの失敗回数のしきい値を変更するには、Tableau サービスマネージャー構成セットオプションである、`dataAlerts.SuspendFailureThreshold`tsm を使用します。

これにより、アラートを一時停止するために必要な連続した失敗アラート数のしきい値が設定されます。これはサーバー全体の設定です。しきい値は、サーバー上で設定されているすべてのデータドリブンアラートに適用されます。

一時停止したアラートの再開

アラートの失敗が何度も起こると、アラートが一時停止した旨の通知メールを受信します。管理者またはアラート所有者が一時停止アラートを再開する方法はいくつかあります。

- Tableau Web ページの [タスク] の [アラート] エリアで、[前回タップ] 列に、アラートが一時停止になっていることを示す アイコンが表示されています。**[...]> [アラートを再開]** を選択し、アラートを再開します。
- 通知メールで **[アラートを再開]** をクリックし、アラートを再開します。通知を使用し、アラートを再開することも、ビューが変更されたためアラートを削除する必要があることを示すこともできます。
- 影響を受けるビューまたはブックのアラートパネルから。ビューまたはブックからアラートを再開するには、**アラート** を選択してアラートパネルを開きます。一時停止したアラートの横にアイ

コンが表示されます。影響を受けるアラートの[**アクション** >]、[**アラート再開**]の順に選択します。

アラートが再び機能するとアラート所有者にアラート通知メールが送信されます。

サーバーのデータ主導アラートのチェック頻度を制御する

既定では、Tableau Server は 60 分ごとにアラートのデータ条件が true かどうかを確認します。パフォーマンスへの影響に気付いた場合は、この時間間隔を tsm configuration set オプションの `dataAlerts.checkIntervalInMinutes` でカスタマイズできます。

`dataAlerts.checkIntervalInMinute` の設定に関係なく、サーバーは関連するワークブック内の抽出が更新された場合も常にアラートをチェックします。設定で指定されているよりも頻繁にアラートをチェックするには、抽出更新スケジュールを変更します。

サーバーのアラートチェックプロセスの追跡

抽出以外のバックグラウンドタスクビューで、次のタスクを検索することで、サーバーのアラートチェックプロセスを追跡できます。

- チェックするデータアラートの検索
- データアラート条件が真かどうかのチェック

"検索" タスクは "チェック" タスクを、現在関連するメールを送信できるアラートに制限します。たとえば、ユーザーがメールの頻度として "最大毎日" を選択している場合は、アラート条件が真になった後に、サーバーは 24 時間待機してから再度アラートをチェックします。

各 "チェック" タスクが 1 つのサーバー バックグラウンドプロセスを使用し、関連するビューを読み込んでアラート条件を評価します。すべてのユーザーが同じバージョンのビューを表示している場合は、1 回だけ読み込まれます。ただし、ユーザーがビューにフィルターを適用している場合、または表示されるデータがユーザーレベル セキュリティによって制限されている場合、ビューは受信者ごとに 1 回ずつ読み込まれます。

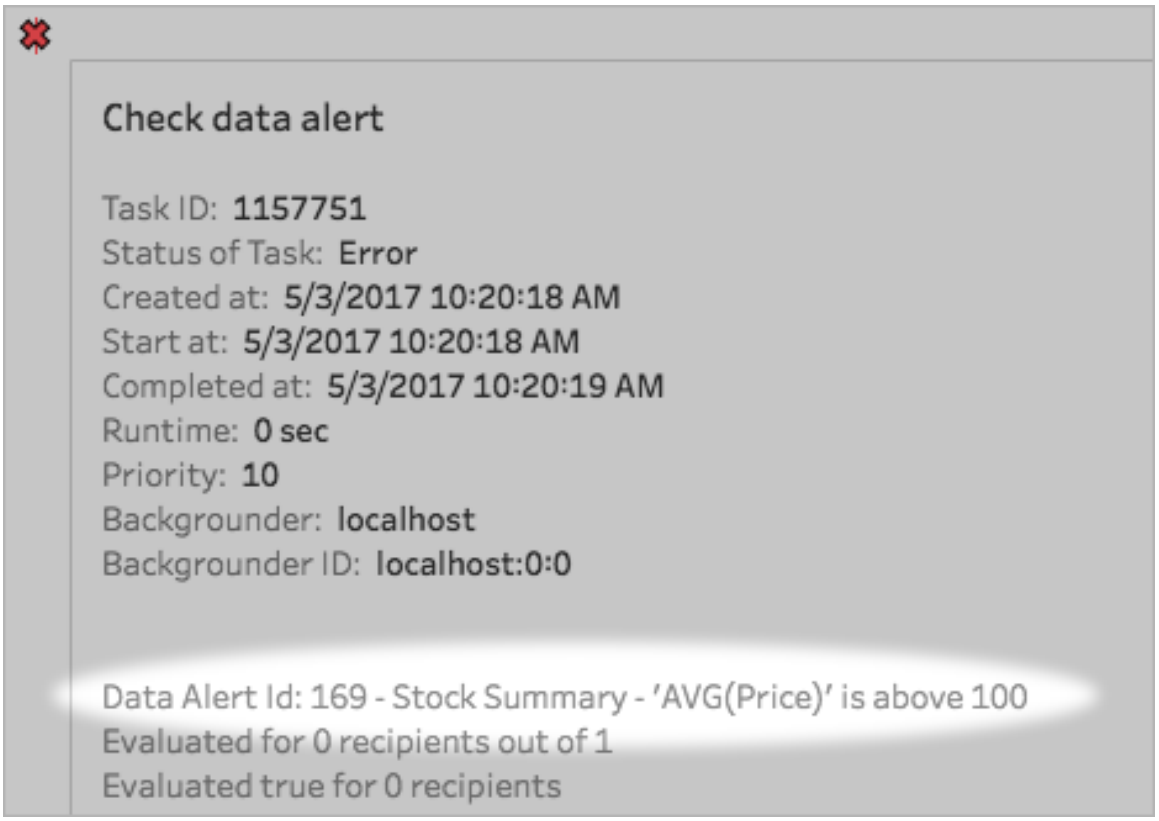
失敗したアラートの特定および修正

管理者は、ユーザーが気付いていない失敗したアラートをプロアクティブに識別できます。確認のため、以下を行ってください。

1. サイト メニューで [ステータス] を選択します。
2. [抽出以外のバックグラウンド タスク] を選択します。
3. [タスク] ドロップダウン メニューから、[Check if Data Alert is True (データ アラートが true かどうかをチェックする)] を選択します。
4. 右端の [エラー] をクリックし、失敗したアラートのリストを表示します。
5. 赤い障害アイコンをポイントすると、アラートの詳細を示すツールヒントが表示されます。

アラート所有者を判別するには、Tableau Server リポジトリの data_alerts テーブルのアラート ID 番号を探してください。でアラート ID 番号を検索します (サイトのアラート管理エリアで、番号に続くアラート名を検索できますが、複数の異なるアラートが同じ名前を使用している場合があることがわかります)。

> **注:** アラートが 10 回失敗すると、アラート所有者に自動的に通知されます。管理者はアラート所有者が通知を受け取るタイミングをカスタマイズすることができます。Tableau Server 2018.1 にアップグレードする前に失敗したアラートは、ユーザーに通知されません。



失敗したアラートは、多くの場合、Tableau Server でのコンテンツの変更によって引き起こされます。次のような変更が生じる場合は、アラートを作成し直すことをお勧めします。

- ワークブック、ビュー、またはデータフィールドの削除または名前の変更を行う。
- ワークブックに埋め込まれたデータベース クレデンシャルが期限切れである
- データに基づくアラートには、ライブ接続で使用する埋め込みの認証資格情報が必要です。現在、アラートで OAuth を使用することはできません。

- データソースにアクセスできなくなる。

ヒント: アラートが失敗したときに自動的にメールを取得するには、Tableau Server リポジトリでデータを収集するの手順に従い、"background_jobs" テーブルに接続します。そのテーブルから、"データアラート条件が真かどうかのチェック" ジョブ名とその終了コードを含むカスタム ビュー

を作成します。その後、終了コードが 1 に等しい (失敗) 場合は常にメールを受信するように
データ主導アラートをセットアップします。

メトリクスのセットアップ

従来のメトリクス機能の廃止

Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

メトリクスは、売上の合計などのメジャーの集計の値を追跡する Tableau コンテンツの一種です。メトリクスは頻繁に更新され、現在の値が一目でわかる形式で表示されるため、データを監視する場合に便利です。ユーザーがメトリクスを操作する方法の詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

Tableau Server 管理者は、メトリクス更新の頻度と更新の失敗の処理方法をコントロールできます。また、ユーザーが特定のサイトまたはサーバー全体のメトリクスを作成したり、無効にしたりできるようにすることもできます。

ユーザーがメトリクスを作成できるようにする

サイトでメトリクスが有効になっている場合、Creator または Explorer (パブリッシュ可能) のサイトロールを持つすべてのユーザーは、正しいパーミッションを持っていればメトリクスを作成できます。

メトリクスは、Tableau サイトの既存のビューから作成されます。ユーザーがビューにメトリクスを作成できるようにするには、次の点を確認します。

- ユーザーは、ビューが属するワークブックに対する [メトリクスの作成/更新] のパーミッション機能を持っている。詳細については、パーミッションを参照してください。

- 必要な場合は、データソースのパスワードは埋め込まれます。詳細については、Tableau Server での接続の編集を参照してください。

サイトのメトリクスの無効化

メトリクスは、既定ですべてのサイトで有効になっています。サイトごとにメトリクスを無効にすることができます。

1. メトリクスを無効にするサイトのナビゲーションパネルで、**[設定]** をクリックします。
2. **[メトリクスコンテンツタイプ]** で、**[メトリックを有効にする]** チェックボックスをオフにします。
3. **[保存]** をクリックします。

メトリクスコンテンツタイプを無効にすると、サイトにメトリクスが表示されなくなります。既存のメトリクスのデータは保持されますが、これらのメトリクスは更新されなくなります。メトリクスを再度有効にすると、これらのメトリクスが再表示され、更新が再開されます。

メトリクスの作成/更新のパーミッション機能を拒否することにより、特定のワークブックのメトリクスを無効にすることもできます。詳細については、パーミッションを参照してください。

サーバーのメトリクスの無効化

特定のサイトのメトリクスを無効にするだけでなく、サーバー全体のメトリクスを無効にすることもできます。サーバーレベルで無効にすると、メトリクスは更新されず、どのサイトにも表示されません。メトリクスプロセスは実行されず、メトリクスのサイト設定も使用できなくなります。既存のメトリクスデータは保持されるため、メトリクスを再度有効にすると、それらのメトリクスは復元されます。

メトリクス機能は既定で有効になっています。メトリクスを無効にするには、tsm configuration set オプション `metricsservices.enabled` を使用します。

メトリクスの更新頻度の構成

メトリクスが更新されると、作成元のビュー(接続ビューと呼ばれる)を介して新しいデータがチェックされます。サーバーへのパフォーマンスに影響が生じると、更新間隔を長くしたり、ユーザーが最新のデータを必要とする場合は更新間隔を短縮することができます。

ライブデータに依存するメトリクスは、既定で 60 分ごとに更新されます。ライブデータの更新間隔を調整するには tsm 構成セット オプション `metricsservices.checkIntervalInMinutes` を使用します。これはサーバー全体の設定です。

抽出ベースのデータ更新に依存するメトリクスは、抽出が更新されたときに更新されます。これらのメトリクスの更新頻度を制御するには、抽出の更新頻度を変更します。詳細については、抽出更新スケジュールを参照してください。

メトリクス更新の失敗通知を構成する

メトリクスが更新に必要なデータに接続できない場合、更新は失敗します。メトリクスの更新が 10 回連続して失敗すると、メトリクス所有者は電子メール通知を受け取ります。

警告メールが送信されるまでの連続した失敗の数を調整するには tsm 構成セット オプション `metricsservices.failureCountToWarnUser` を使用します。これはサーバー全体の設定です。

メトリクス更新が中断されるタイミングを構成する

メトリクス更新の失敗が 175 回連続して失敗すると、更新は一時停止されます。メトリクス更新が中断されると、更新が手動で再開されるまで、サーバーは新しいデータのチェックを試みなくなります。

更新が中断されるまでの連続した失敗の数を調整するには tsm 構成セット オプション `metricsservices.maxFailedRefreshAttempts` を使用します。これはサーバー全体の設定です。

メトリクスの管理

メトリクスはビューから作成されますが、アラートやサブスクリプションなどのビューに関連付けられてはいません。つまり、ワークブックの管理方法と同様にメトリクスを管理するには、メトリクスの名前の変更、移動、タグ付け、削除、または権限の設定を行います。

プロジェクト階層をナビゲートするか、次のパスを使用して管理するメトリクスを検索します。

- サイトのすべてのメトリクスを表示するには、[探索] セクションに移動し、**[すべてのメトリクス]** を選択します。

- ワークブック内のすべてのビューから作成されたメトリクスを表示するには、ワークブックに移動して、**[接続済みメトリクス]** タブを選択します。
- 1 つのビューから作成されたメトリクスを表示するには、ビューを開いて、ツールバーから **[Watch (視聴)] > [メトリクス]** を選択します。

アドレスの失敗および中断されたメトリクス更新

メトリクスの更新は、次のいずれかの理由で失敗する場合があります。

- 接続されているビューが削除または変更されました。
- 接続されているビューのパーミッションが変更されました。
- データソースのパスワードが埋め込まれなくなったか、有効でなくなりました。
- メトリクスの所有者がメトリクスの更新に必要なサイトロールを持っていません。Creator または Explorer (パブリッシュ可能) のサイトロールが必要です。
- 一時的な接続の問題が起きています。この場合は時間が過ぎると解決されます。

> **注:** 所有者が更新に必要なサイトロールを持っていないためメトリクスの更新が一時停止されている場合、所有者を変更しない限り、更新を再開したり、メトリクスを削除したりすることはできません。

メトリクス更新が失敗する理由と、ユーザーが更新を修正するために実行できる操作の詳細については、「失敗した更新の修正」を参照してください。

更新が失敗する原因となった方法で接続ビューが変更された場合、メトリクスを上書きするようユーザーに促しますが、そのビューは引き続き使用可能です。ユーザーは、既存のメトリクスと同じプロジェクトで同じ名前のメトリクスを作成することで、メトリクスを上書きできます。

一時停止された更新の再開

データソースの正しいパスワードを埋め込むなど、失敗の原因が修正された場合は、メトリクスの更新を再開できます。

1. 影響を受けたメトリクスを検索します。更新が中断されたメトリクスは、グリッドとリストビューで、前回の更新時刻ではなく、**[更新が中断されました]** というテキストを表示します。
2. 警告メッセージで、**[更新の再開]** をクリックします。

Tableau で更新が試行されます。この試行が成功すると、確認が送信され、更新はスケジュールに従って実行されるようになります。試行が成功しなかった場合、更新は一時停止されたままになります。自分またはメトリクスの所有者は、メトリクスを削除または上書きしたり、履歴データを参照するように維持したりできます。

管理ビューでメトリクスアクティビティを監視する

Tableau Server の管理ビューを使用して、メトリクスの更新を監視し、どのユーザーがメトリクスを作成および表示しているかを確認します。

1. 監視したいサイトに移動するか、サイトの選択で**[すべてのサイト]** を選択してサーバー全体のアクティビティを監視します。
2. ナビゲーション パネルで、**[サイトの状態]** または **[サーバーの状態]** をクリックします。
3. 検査するダッシュボードを選択します。

    - メトリクス更新のアクティビティを監視するには、抽出以外のバックグラウンド タスク ダッシュボードを開きます。

      タスク **更新するメトリクスを表示** または **ビューですべてのメトリクスを更新** にフィルターします。

    - メトリクスを作成しているユーザーを確認するには、すべてのユーザーによるアクションまたは特定のユーザーによるアクション ダッシュボードを開きます。

      [メトリクスの作成] アクションのフィルター。

    - メトリクスに関する最近のアクティビティを表示するには、最近のユーザーによるアクション ダッシュボードを開きます。

      [最近実行された操作] の一覧を確認します。

パブリッシュされたデータ ソースを編集する

あなたがデータ ソースをパブリッシュし、チームが多くのワークブックでそのデータ ソースを使用しているとします。これは順調なスタートですが、あなたは、データ ソースをすばらしいものにするために変更を加えることを検討しています。これらの変更を実装する前に、検討している変更が Tableau

でどのように見えるかを確認する必要があります。そして、最も重要なことは、変更をテストして、データソースを使用している既存のワークブックに悪影響を与えないようにすることです。

パブリッシュされたデータソースを編集することにより、データソースを単一のデータソースとして維持しながら、変更をテストしてデータソースを改善できます。

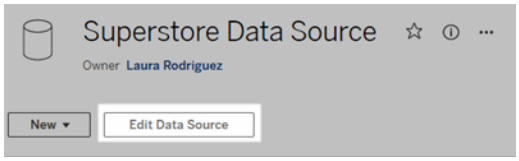
> **注**: パブリッシュされたデータソースをブラウザで編集できるのは、Creator のサイトロールを持つユーザーのみです。

変更を編集してテストする

新しいパブリッシュされたデータソースを作成したり、既存のパブリッシュされたデータソースを編集したりする場合でも、ブラウザーを離れることなく、[データソース] ページから結合を作成したり、スキーマを編集したりできます。その後、Scratchpad を使用して、データソースをパブリッシュする前に、変更のテスト、フォルダーの作成、階層の整理、フィールドとエイリアスの名前変更を行います。データソースを編集するときは、Tableau Cloud でオーサリングするときと同じ機能をすべて利用できます。詳細については、「Web 作成と Tableau Desktop 機能の比較」を参照してください。

パブリッシュされたデータソースを編集するには、次の手順を実行します。

1. スタートページまたは [探索] ページから、編集するデータソースに移動します。
2. **[データソースの編集]** をクリックします。



3. **[データソース]** ページをクリックして、結合を作成するか、スキーマを編集します。
4. **[Scratchpad]** シートをクリックします。
5. **[データ]** ペインから、フォルダーの作成、階層の整理、フィールドとそのエイリアスの名前変更、またはパブリッシュされたデータソースとともに保存されているメタデータの更新を行います。
6. フィールドを Scratchpad にドラッグアンドドロップして、変更が期待どおりに機能していることを確認します。
7. **[パブリッシュ]** をクリックします。

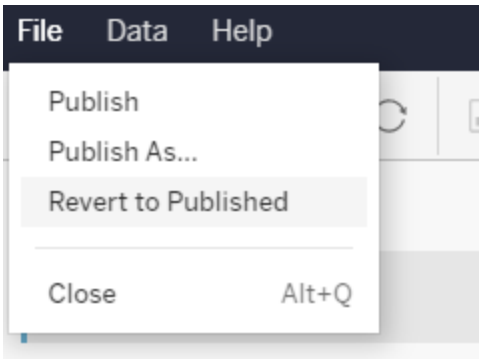
また、ワークブックの場合と同じように、データソースのコピーを作成する場合、**名前を付けてパブリッシュ**することもできます。

> **注:** パーソナル スペースは、パブリッシュされたデータ ソースをサポートしていません。

ロールバック変更

パブリッシュされたデータ ソースの最後のバージョンに戻すには、次の手順を実行します。

1. 元に戻すデータ ソースに移動します。
2. **[ファイル]** をクリックします。
3. **[前回パブリッシュしたときの状態に戻す]** を選択します。



これにより、そのデータ ソースの前回パブリッシュしたときのバージョンに戻ります。

サポートされている接続を理解する

パブリッシュされたデータ ソースの編集では、以下をサポートしていません:

- Tableau Cloud の Tableau Bridge コネクタ
- Tableau Cloud と Tableau Server に埋め込まれたパスワードを使用するデータ ソース

また、[データ ソース] ページは、サポートされていないパブリッシュされたデータ ソース接続タイプでは使用できません。これらのタイプには、.hyper ファイル タイプが含まれますが、これに限定されません。どの接続タイプがサポートされているかを確認するには、Creators: Web 上のデータへの接続を参照してください。

パーミッションについて学ぶ

パブリッシュされたデータソースを編集するには、Creator ライセンスが必要です。このライセンスには、それぞれのフォルダー内のデータソースに対する [保存] または [名前を付けて保存] パーミッションがあります。詳細については、「パーミッション」を参照してください。

フローによってパブリッシュされたデータソースを編集する

フローによってパブリッシュされたデータソースを編集すると、次にスケジュールされたフロー中に変更が上書きされます。その代わりに、フロー内のデータソースを編集してください。詳細については、「Tableau Server または Tableau Cloud へのフローのパブリッシュ」を参照してください。

Tableau Server でのバックグラウンドジョブの管理

Tableau Server では、定期的に実行される抽出更新、サブスクリプション、またはフローをユーザーがスケジュールできます。これらのスケジュールされた項目は**タスク**と呼ばれます。バックグラウンダープロセスは、これらのタスクの固有のインスタンスを開始して、スケジュールされた時刻に実行します。結果として開始されたタスクの固有インスタンスは**ジョブ**と呼ばれます。Web インターフェイスで **[今すぐ実行]** オプションをクリックしたり、プログラムで REST API を使用したり、tabcmd コマンドを使用したりして手動で実行を開始した場合も、ジョブが作成されます。

たとえば、毎日午前 9 時に実行される抽出更新タスクを作成するとします。この抽出更新タスクについて、毎日午前 9 時にバックグラウンダーが実行するジョブが作成されます。ユーザーが生成したジョブに加えて、バックグラウンダーは複数のシステムジョブをユーザーに代わって実行し、サムネイル生成などの一般的な Tableau 作業フローをサポートします。

これらのジョブをすべて実行するため、バックグラウンダーは日中のさまざまな時刻に多くのリソースを使用する可能性があります。ジョブ管理機能を使用すると、サーバーやサイトの管理者は自身のサーバーやサイトで実行されるジョブの詳細を取得し、それらのジョブに対してアクションを実行してサーバーリソースの使用量をより適切に管理できます。システムジョブはサーバー管理者のみが表示でき、既定では除外されています。

[全般] 設定ページの **[今すぐ実行]** 設定でユーザーによるジョブの手動実行を許可またはブロックすることによっても、リソースを管理できます。既定では、このオプションはユーザーが手動でジョブを実行できるように設定されています。ユーザーによるジョブの手動実行を防止するには、このチェックボックスをオフにします。バックグラウンダー リソースの管理の詳細については、Tableau Server バックグラウンダー プロセスを参照してください。

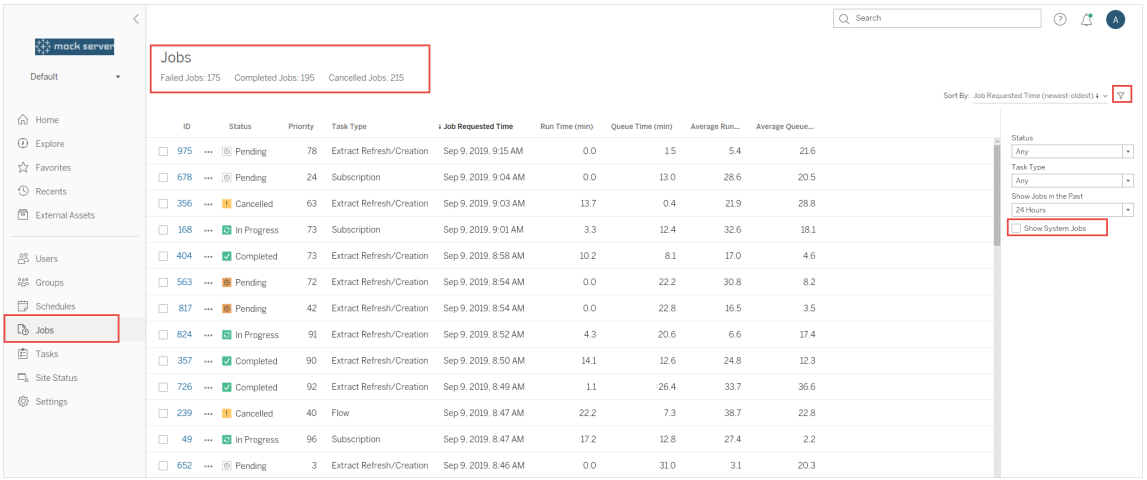
ジョブに関する情報を含まれる [ジョブ] ページには、左側のナビゲーション メニューの **[Existing Tasks (既存のタスク)]** メニューに移動してアクセスできます。

注: ジョブに関する情報は、サーバーやサイトの管理者だけが表示できます。

概要

このトピックでは、[ジョブ] ページに表示される情報を理解する方法について説明します。

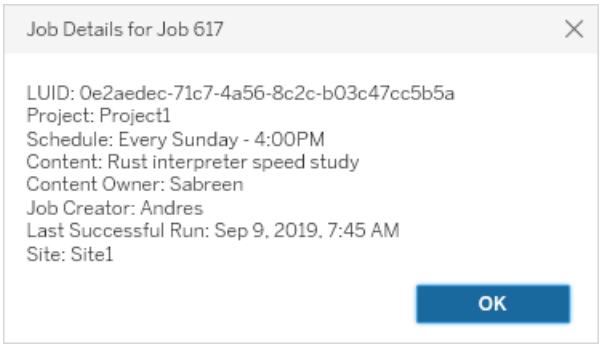
ページの最上部には、過去 24 時間内に**失敗したジョブ**、**完了したジョブ**、および**キャンセルされたジョブ**の数を示す高度な統計が表示されます。サーバー管理者の場合、これにはシステム ジョブも含まれます。フィルターを適用してもこれらの値は変化しません。



生成されたジョブごとに、ジョブ ID、ジョブのステータス、優先度、ジョブが生成されたタスクのタイプ、現在の実行時間、ジョブが進行中の場合は現在までのキュー時間、キューに入っている場合は平均実行時間と平均キュー時間が示されます。

Tableau は、過去の実行時間とキュー時間を記録し、平均実行時間と平均キュー時間を計算します。平均実行時間と平均キュー時間はどちらも、次の式を使用して加重平均として計算されます。((現在の実行時間またはキュー時間の平均 x 4) + 最新の実行時間またはキュー時間)/5

ジョブ ID は**管理ビュー**でジョブを表示する際に役立ち、ワークグループ データベースへのクエリを実行する場合にも使用できます。ジョブ ID をクリックすると、ジョブに関する詳細情報 (ジョブ LUID、プロジェクト名、スケジュール、コンテンツ名、コンテンツ所有者、ジョブ作成者、最後に成功したジョブ実行時間、サイト名など) が表示されます。サイト名は **[すべてのサイトを管理]** メニューを使用して [ジョブ] ページに移動した場合に表示されます。



注: [データ ソース] ページから **[今すぐ更新]** を実行すると、**[Job Details (ジョブの詳細)]** ダイアログ ボックスに LUID のみが表示されます。

重要: Tableau Server 2019.4 へのアップグレード前 24 時間以内に存在していたジョブでは、**[ジョブ]** ページに **平均キュー時間**、**平均実行時間**、**最後に成功した実行時間**、**ジョブ作成者** のデータは存在しません。

タスクのタイプ

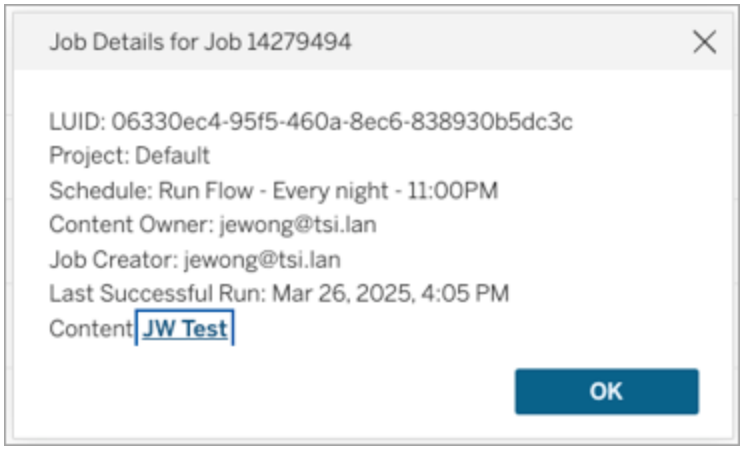
タスクには次のいくつかのタイプがあります。

- 抽出: これには抽出の作成、増分抽出の更新、完全抽出の更新が含まれます。抽出更新の詳細については、クイック スタート: スケジュールに基づく抽出の更新を参照してください。

- サブスクリプション: ワークブックとビューのサブスクリプションが含まれます。詳細については、サブスクリプションのサーバー設定を参照してください。

- フロー: スケジュールされたフローと手動のフロー実行が含まれます。フローの最大ランタイムについては、「ジョブ ランタイム容量」を参照してください。同時フロー ジョブを実行する場合の容量制限については、「「同時ジョブ容量」を参照してください。所有しているリソースブロックの数を超えるフローをスケジュールすると、エラーが発生する可能性があります。詳細については、ナレッジ ベースの記事「フロー ジョブの保留」を参照してください。

  バージョン 2025.2 以降では、**[ジョブの詳細]** ダイアログに Tableau Prep フローへのリンクが表示されます。フローの [配置] ページに簡単に移動して、フローの詳細を取得することができます。フローが削除されている場合、リンクは表示されません。



- 暗号化: 次のものが含まれます。

  - 抽出の暗号化と復号化
  - フローの暗号化と復号化
  - 抽出とフローのキー更新

- システム: これは、バックグラウンダーが Tableau Server をサポートするために背後で処理するすべてのシステム ジョブです。

フィルター

フィルターを実行すると、特定のジョブだけを表示できます。使用できるフィルターは、ジョブのステータスのタイプ、タスクのタイプ、および時間範囲です。時間範囲フィルターでは、過去 1 ~ 24 時間を4時間ごとに区切った時刻から選択できます。システム ジョブをフィルターして含めるオプションはサーバー管理者が使用できます。

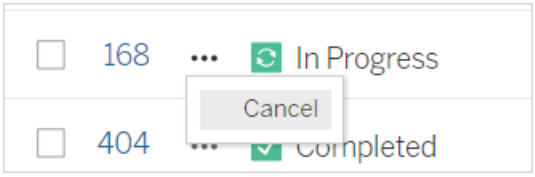
ジョブのキャンセル

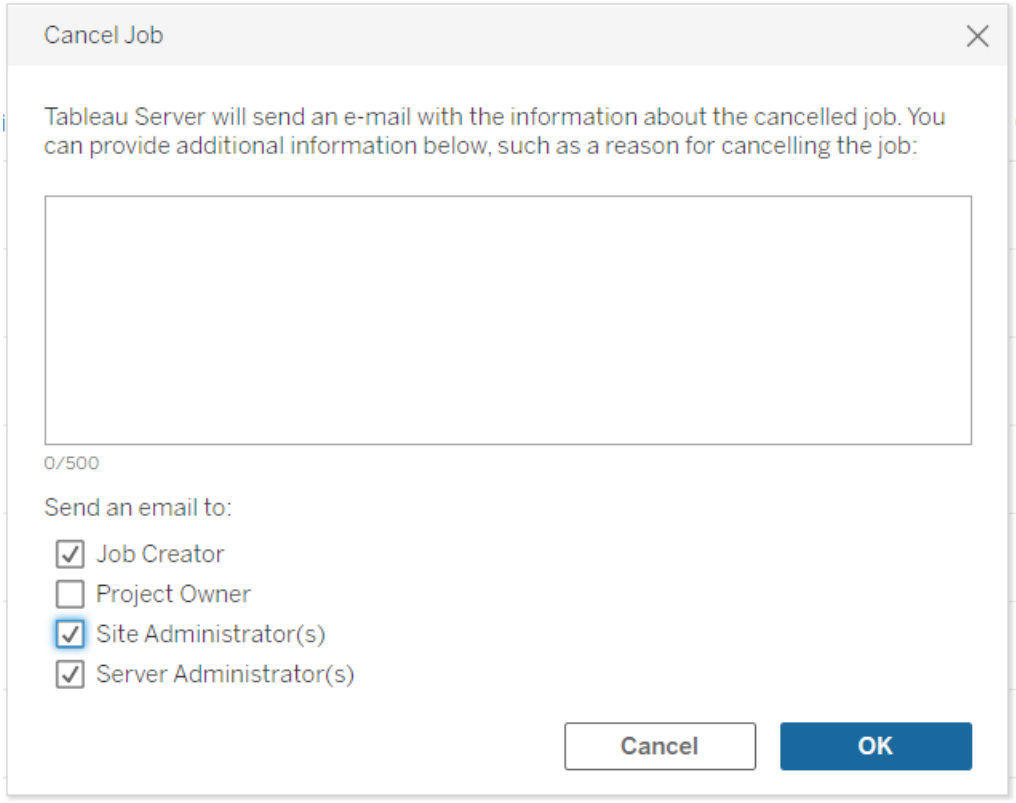
抽出更新、サブスクリプション、およびフロー実行ジョブはキャンセルできます。一度にキャンセルできるのは 1 つのジョブだけであり、一度に複数のジョブをキャンセル用に選択することはできません。

ジョブをキャンセルすると、ジョブがキャンセルされた時刻、影響を受けたコンテンツ、キャンセル前のジョブ実行時刻が記載されたメールが、**[ジョブのキャンセル]** ダイアログ ボックスで選択した受信者に送信されます。さらに、メールに含めるカスタマイズした注記を追加することができます。

受信者を選択しない場合、ジョブはキャンセルされますが、メールは送信されません。

ジョブをキャンセルするには、ジョブ ID の横にある省略符号 (...) をクリックし、ダイアログを使用してジョブをキャンセルします。

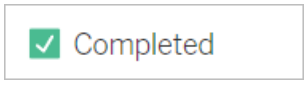


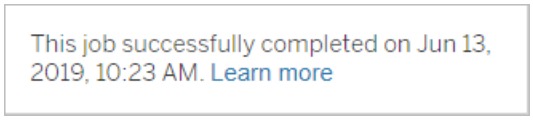


ステータス

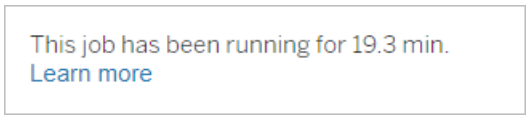
ジョブには 7 種類のステータスがあり、各ステータスにカーソルを合わせるとより多くの関連情報が表示されます。

- **完了しました:** このジョブは**正常に完了しました**。ステータスにカーソルを合わせたときに表示されるツールヒントでジョブの完了時刻を確認できます。

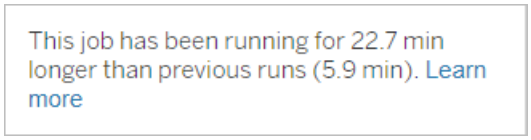




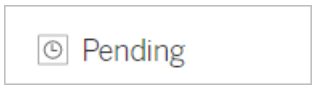
- **進行中**: このジョブは**進行中**です。ステータスにカーソルを合わせたときに表示されるツールヒントには、ジョブがこれまでの実行された時間が表示されます。

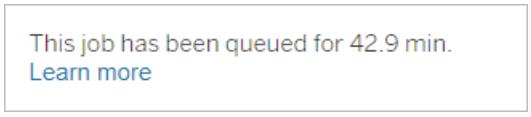


- **進行中:** このジョブは**進行中**ですが、**遅れて実行**されています。Tableau では同じジョブの平均実行時間が追跡されており、現在の実行時間が平均実行時間より長い場合は遅れて実行されているとみなされます。ステータスにカーソルを合わせたときに表示されるツールヒントには、ジョブの平均実行時間と、ジョブが現在平均よりどれほど長く実行されているかが表示されます。

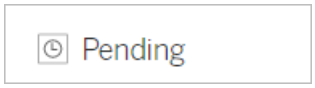


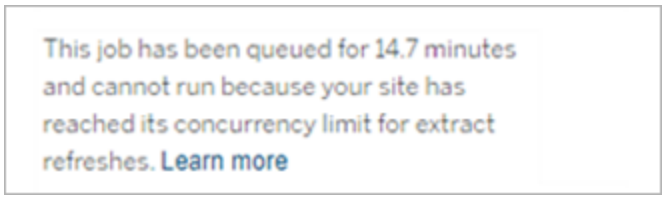
- **保留中:** このジョブは現在**保留中**であり、使用可能なバックグラウンダーの容量が確保されたときに実行されます。ステータスにカーソルを合わせたときに表示されるツールヒントには、ジョブがキュー内にある時間が表示されます。



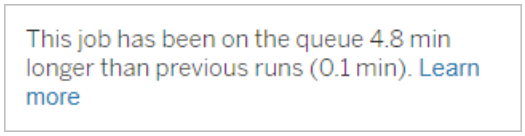


- **保留中:** リソースブロックの数がスケジュールしたフローの数より少なく、サイトが同時実行制限に達しているため、フローを実行できません。

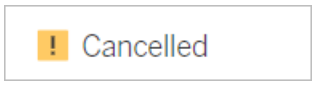


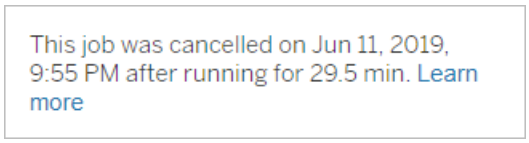


- **保留中:** このジョブは現在**保留中**ですが、**遅れて実行**されています。Tableau では同じジョブの平均キュー時間が追跡されており、現在のキュー時間が平均キュー時間より長い場合は遅れて実行されているとみなされます。ステータスにカーソルを合わせたときに表示されるツールヒントには、ジョブが現在平均よりどれほど長くキューに入っているかが表示されます。

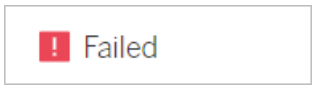


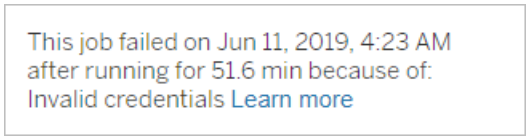
- **キャンセルされました:** このジョブはサーバーやサイトの管理者によって**キャンセルされました**。ステータスにカーソルを合わせたときに表示されるツールヒントには、ジョブがキャンセルされた時刻と、キャンセルされるまで実行された時間が表示されます。



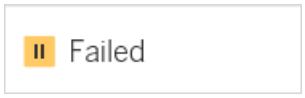


- **失敗しました:** このジョブは**失敗しました**。ステータスにカーソルを合わせたときに表示されるツールヒントには、ジョブが失敗した時刻、失敗するまで実行された時間、および失敗した理由が表示されます。





- **一時停止:** このジョブには一時停止アイコンと一緒に **[失敗しました]** との表示がされます。ジョブが 5 回連続して失敗すると、ジョブは一時停止します。一時停止したタスクは引き続き使用できますが、ユーザーがそれらのタスクを再開するまで、バックグラウンダーによりそのタスクのジョブが作成されることはありません。



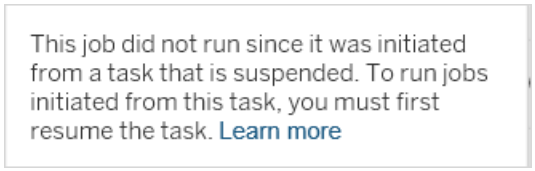


Tableau サービス マネージャーのジョブ

TSM ジョブは、Tableau Server の構成と保守に役立つ管理タスクです。これらのジョブは、Tableau サービス マネージャーによって実行されます。

主な TSM ジョブを次に示します。

- **クリーンアップ**: このジョブは、Tableau Server に cleanup コマンドが発行されたときに作成されます。cleanup コマンドによって古いログ ファイルと一時ファイルが削除されます。TSM CLI の cleanup コマンドに関する詳細については、tsm maintenanceを参照してください。

- **展開**: このジョブは、Tableau Server に対して行う構成の更新を適用するために作成されます。TSM Web インターフェイスや TSM CLI を介して開始できます。TSM CLI コマンドの詳細については、tsm pending-changesを参照してください。構成の更新例には、SSL の有効化、実行ユーザーの有効化、サーバー トポロジの変更などがあります。

  トポロジの変更が行われる回数と複雑さによっては、このジョブが前回の正常な実行よりも長くかかる可能性があります。たとえば、以前の変更がホット トポロジの変更であり、現在の変更が別の内容である場合、現在のジョブの方が前回の変更よりも完了までに時間がかかる可能性があります。

- **バックアップの生成**: このジョブは、Tableau Server に backup コマンドが発行されたときに作成されます。backup コマンドによって、Tableau データ(ファイル ストアとリポジトリ内のデータ) のバックアップ ファイルが作成されます。TSM CLI の backup コマンドに関する詳細については、tsm maintenance backupを参照してください。

  バックアップ ジョブの完了までに要する時間は、バックアップが必要なデータの量によって異なります。バックアップするデータの量が前回のジョブ実行時から増加した場合、ジョブの完了までの時間は前回と比べて長くなります。

- **Tableau Server の初期化**: このジョブは、インストール プロセス中に Tableau Server を初期化するために開始されます。TSM CLI コマンドの詳細については、tsm initializeを参照してください。

- **復元**: このジョブは、Tableau Server に対して restore コマンドが発行されたときに作成されます。restore コマンドによって、Tableau Server データのバックアップ ファイルが復元されます。TSM CLI の restore コマンドに関する詳細については、tsm maintenanceを参照してください。復元ジョブの完了までに要する時間は、バックアップ ファイルのサイズによって異なります。バックアップ ファイルが以前よりも大きい場合、復元ジョブの完了までにより長い時間がかかります。

- **サーバーの起動**: このジョブは、停止していたすべての Tableau Server プロセスを開始するために作成されます。TSM Web インターフェイスや TSM CLI を介して開始できます。TSM CLI コマンドの詳細については、tsm start を参照してください。
- **サーバーの停止**: このジョブは、実行中のすべての Tableau Server プロセスを停止するために作成されます。TSM Web インターフェイスや TSM CLI を介して開始できます。TSM CLI コマンドの詳細については、tsm stop を参照してください。

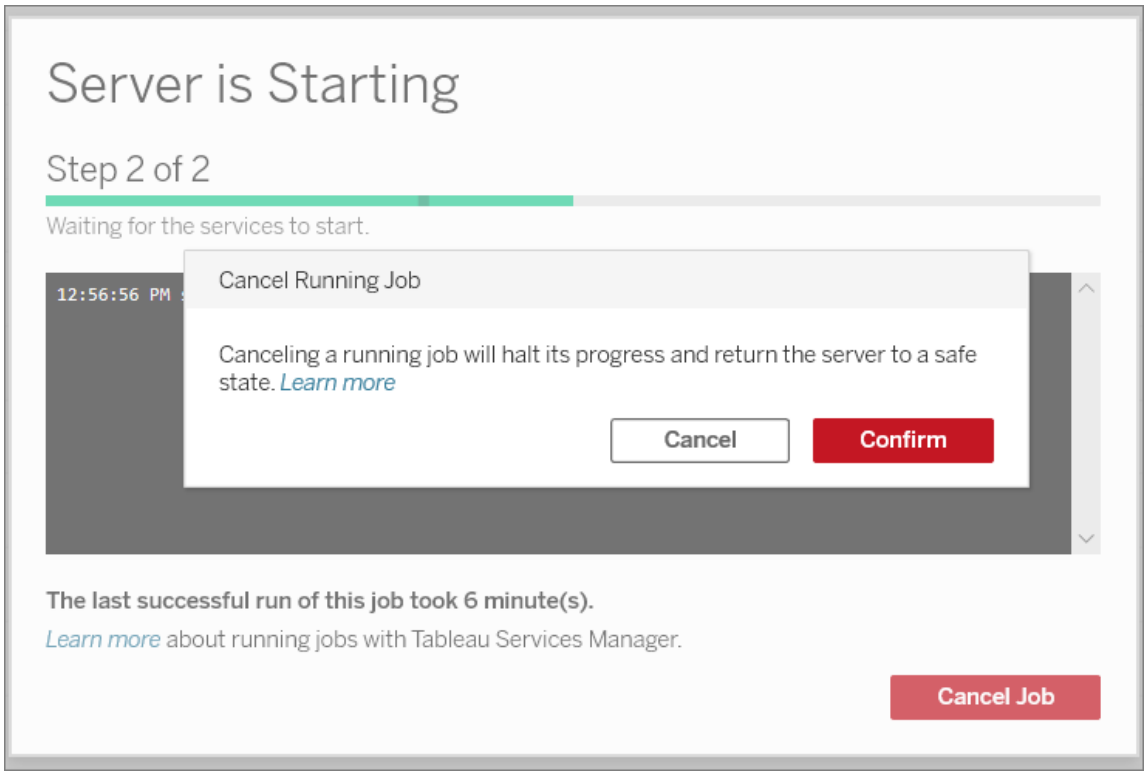
tsm ジョブのキャンセル

ジョブは、TSM Web インターフェイスや TSM CLI を使用してキャンセルできます。詳細については、TSM ジョブのキャンセルを参照してください。

TSM ジョブのキャンセル

TSM ジョブは、TSM Web インターフェイスや TSM CLI を介してキャンセルできます。TSM ジョブは多数ありますが、進行中の場合は特定のジョブのみキャンセルできます。まだ開始していないジョブは、TSM CLI を使用していずれもキャンセルすることができます。

TSM Web インターフェイス: ジョブの実行中にキャンセルできるジョブの場合、以下に表示されているように [ジョブ] ダイアログ ボックスで [キャンセル] オプションを使用できます。



TSM CLI: TSM CLI を使用してジョブをキャンセルするには、tsm jobs を参照してください。

TSM ジョブ全般の詳細については、Tableau サービス マネージャーのジョブを参照してください。

進行中のジョブのキャンセル

ジョブを既に実行中の場合は、特定のジョブ (クリーンアップ、ファイル ストアの使用停止、バックアップの生成、サーバーの再起動、サーバーの起動) のみをキャンセルできます。キャンセルの動作は、ジョブやキャンセル時のジョブの状態によって異なる場合があります。これについては、以下で詳しく説明します。

- **クリーンアップ:** クリーンアップ ジョブをキャンセルすると、クリーンアップを実行するために開始されたサービスがすべて停止します。キャンセルされた時期によっては、一部のファイルが削除されて一部が削除されないため、結果的に部分的なクリーンアップになる可能性があります。

- **ファイル ストアの使用停止:** このジョブをキャンセルすると、Tableau Server ファイル ストア トポロジが、使用停止プロセスを開始する前の状態に戻ります。
- **バックアップの生成:** このジョブをキャンセルすると、バックアップに使用されていたサービスがすべて停止し、Tableau Server でバックアップ プロセスの一部として作成されたファイルの削除が試行されます。
- **サーバーの再起動**:
  - Tableau Server プロセスが停止すると、ジョブがキャンセルされる: ジョブはキャンセルされ、サービスは停止状態に到達しようとします。
  - Tableau Server プロセスの再起動中にジョブがキャンセルされる: ジョブはキャンセルされますが、サービスは再起動を試みます。
- **サーバーの起動**: ジョブはキャンセルされますが、プロセスでは引き続き起動が試行されます。
- **サーバーの停止**: ジョブはキャンセルされ、サービスは停止しようとします。

ジョブをキャンセルする主な理由を次に示します。

1. tsm ジョブは一度に 1 つずつしか実行できないため、別のジョブを実行する必要がある場合は、現在のジョブをキャンセルする必要があります。
2. 実行中のジョブに、実行する予定のない Tableau Server への変更が含まれている場合。

管理ビュー

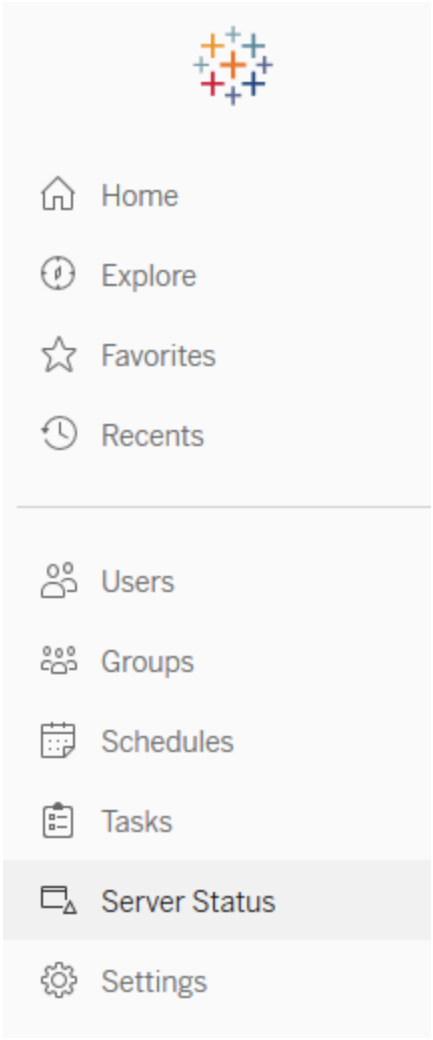
[ステータス] ページには、さまざまな管理ビューを備えた埋め込みの Tableau ワークブックがあります。これらのビューは、サーバーまたはサイトのさまざまな種類のアクティビティを監視するのに役立ちます。

> **注:** 管理ビューを表示するには、PostgreSQL ドライバーをインストールする必要があります。詳細については、データベース ドライバーを参照してください。

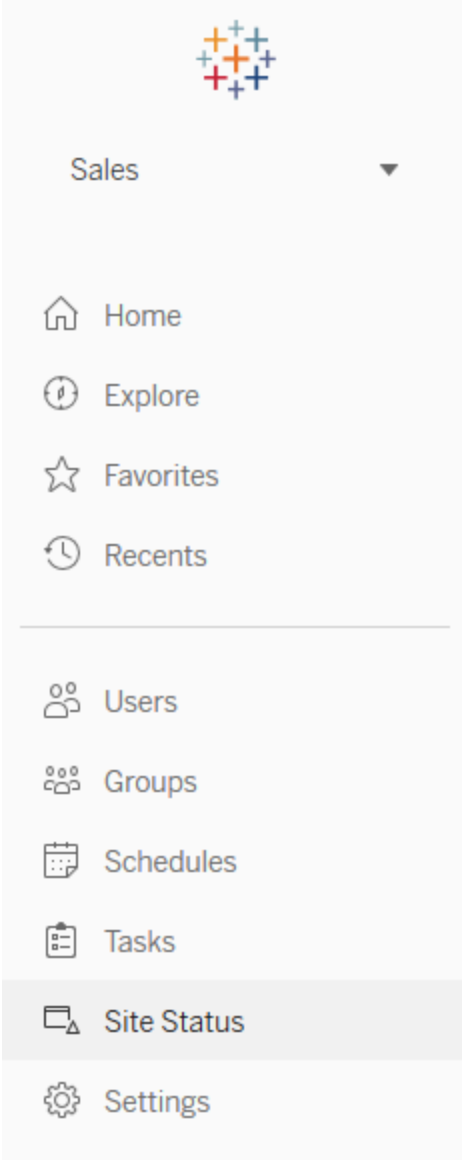
管理ビューへの移動

管理ビューを表示するには、**[ステータス]** をクリックします。サイト管理者はサイトの管理ビューを表示できます。複数のサイトの管理者は、現在のサイトのビューを表示できます。

複数サイトのサーバーで、サーバー管理者はサーバー全体のビューを表示できます。サイト メニューをクリックしてから、**[すべてのサイトの管理]** をクリックしてサーバー メニューにアクセスします。



複数サイトのサーバーで個別のサイトのビューを表示するには、サイト メニューをクリックしてサイト名を選択してから、**[サイトのステータス]** をクリックします。



プレビルト管理ビュー

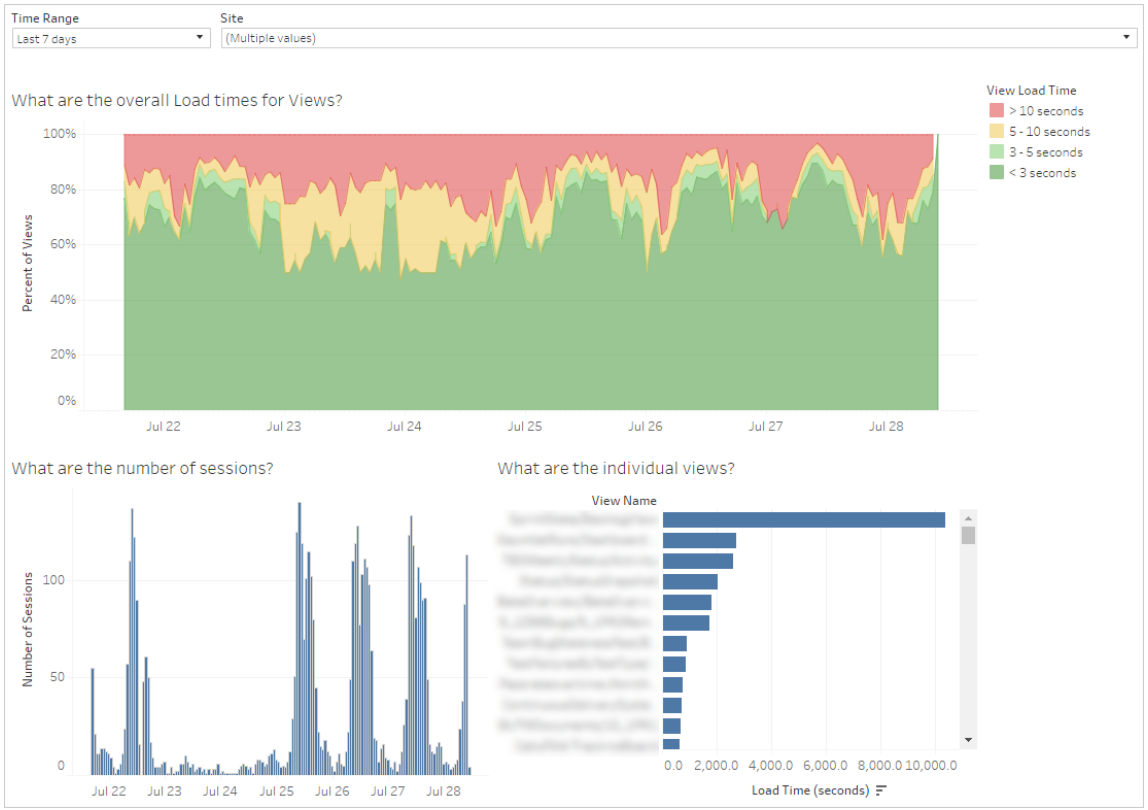
管理ビューは、Tableau Server を最適化し、ユーザーが Tableau コンテンツをどのように操作しているかをよりうまく把握するのに役立つ強力な監視ツールです。右にリストされる管理ビューは、Tableau Server に含まれます。ビューのリンクをクリックし、ビューが提供する情報の解釈や対応の詳細をご覧ください。

独自の管理ビューを作成するには、カスタム管理ビューの作成を参照してください。

ビューのパフォーマンス

> **注:** このビューは、サーバー管理者のみが使用できます。マルチサイト展開でサーバー ビューにアクセスするには、サイト メニューをクリックし、**[すべてのサイトを管理]** を選択します。管理ビューへの移動方法の詳細については、管理 ビュー を参照してください。

[ビューのパフォーマンス] 管理 ビューには、ビューの読み込みにかかる時間と、サーバーで一度に実行できるセッションの数が表示されます。



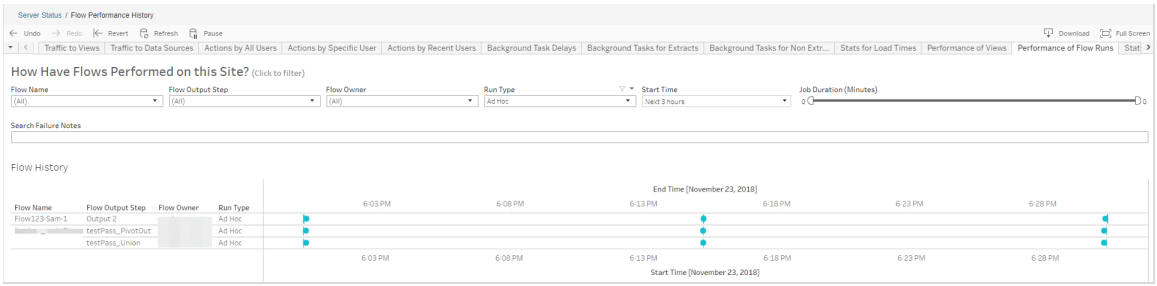
セッション数の急増と読み込み時間の低下の急増を比較し、高いユーザー トラフィックによってサーバーの速度が低下する時刻を識別することができます。また、読み込み時間別に個々のビューを見て、読み込みに最も時間がかかるビューを理解できます。

一部のビューでは、閲覧する時刻に関係なく読み込みに時間がかかる場合があります。**[読み込み時間の統計]** 管理ビューでは、最適化する必要があるワークブックを識別できます。ワークブックを最適化する簡単な方法の一部を次に示します。

- 各ビューに表示する情報を減らす。
- ビューを分割する。
- フィルターの数を減らす。
- データ抽出を使用する。

フロー実行のパフォーマンス

このビューは、サイト上のすべてのフローのパフォーマンス履歴を表示するために使用します。フローの名前、出力ステップ名、フローの所有者、実行タイプ (スケジュールまたはアドホック)、およびフロー実行の開始時間でフィルターできます。フローで使用できるその他の管理ビューについての詳細には、「フローの正常性とパフォーマンスの監視」を参照してください。



このビューを使用して回答可能な質問には次のものがあります。

- **現在スケジュールされているフロー タスクは何か?** – これを確認するには、[開始時刻] フィルターを使用して、調べたい時間枠を選択します。たとえば、今後 3 時間でスケジュールされているフロー タスクを表示するには、**[時間] -> [今後]** を選択し、**「3」**と入力します。
- **フロー タスクの期間はどの程度か? -** これに回答するには、ビュー内のマークをクリックし、タスク実行期間を含む詳細を表示します。

アドホックで実行されたフローの数、スケジュールされた実行の数はいくつか? - これに回答するには、**[実行タイプ]** フィルターを使用し、**[アドホック]** または **[スケジュールされています]** を選択します。

> **注**: 今回のリリースではこの機能に対応しておらず、実際にはデータはフィルターされません。

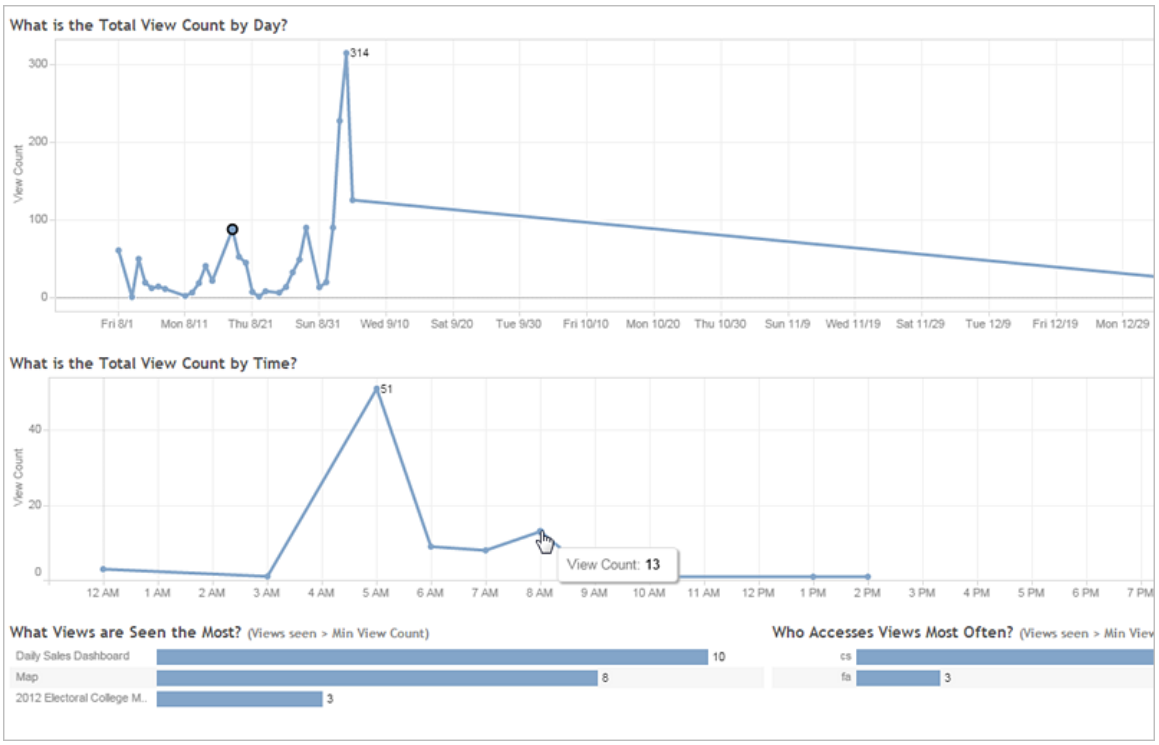
上記の質問以外にも、収集可能なインサイト例には次のようなものがあります。

- 最も頻繁に実行されるフローに、最も多くマークが付けられます。
- 現在いくつのフローが同時に実行されているかを表示するには、**[進行中]** または **[保留中]** と表示されているマークにカーソルを合わせて **[保持]** を選択し、現在実行中のすべてのフロー実行をフィルターします。
- 特定の期間に同時に実行されているフロー数を確認するには、**[開始時間]** フィルターで範囲を選択します。たとえば、**[今後 3 時間]** を選択すると、今後 3 時間にどのフローが実行されるかを表示することができます。

ビューへのアクセス量

[ビューへのアクセス量] ビューには、どれぐらいの量のユーザー トラフィックがビューに流れるかが表示されます。

ビュー、ワークブック、および時間範囲を選択することで、表示する情報やその情報を含む時間枠をフィルターリングできます。サーバー管理者はサイトを指定できます。



ビューの最上部にある2つのタイム ラインには、指定した時間範囲 (デフォルトは過去7日間) にわたって使用されているビューが表示されます。

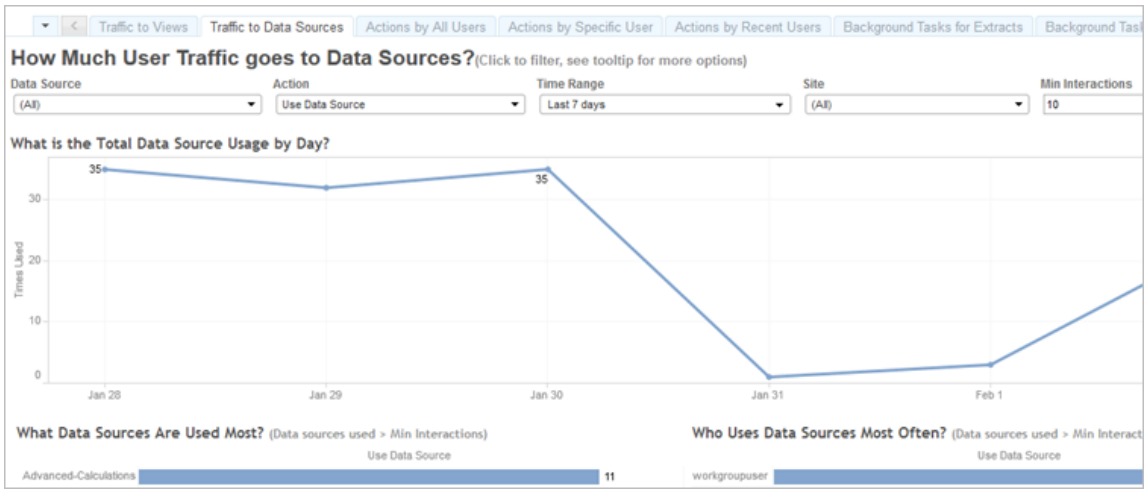
- **日ごとの合計表示回数 -** これは、設定されたフィルターに基づく日ごとの合計表示回数を示しています。表示回数を表示するには、線上の点にマウスのポインターを置きます。選択に基づいてビューの他のセクションを更新するための点を選択します。
- **時間ごとの合計表示回数 -** これは、時間ごとの合計表示回数を示しています。このグラフは、フィルターや選択によって影響を受けます。

ビューの下部にある2つの棒グラフには、ビューの上部にある**[表示回数の最小値]** フィルターによってフィルターリングされた結果が表示されます。これらは、最も頻繁にアクセスされたビューと、最も頻繁にビューにアクセスしたユーザーを示します。表示回数が最小値以上であるビューとユーザーのみが表示されます。

- **最も頻繁に表示されたビュー -** これは、最も頻繁にアクセスされたビューのリストです。ビューの他のセクションと同様、フィルターや任意の選択によって情報が制限されます。
- **最もビューを使用したユーザー -** これは、最もビューを使用したユーザーを示し、フィルターやユーザーによる選択によって限定されます。

データソースへのアクセス量

[データソースへのアクセス量] ビューは、Tableau Server インストールのデータソースの使用状況を表示する機能を提供します。これにより、最も頻繁に使用されているデータソースや比較的使用頻度の低いデータソースを特定できます。データソース、そのデータソースに実行されるアクション、および時間範囲を選択することで、表示される情報をフィルターリングできます。サーバー管理者はサイトを指定できます。



ビューの最上部にあるタイムラインには、指定した時間範囲 (デフォルトは過去7日間) にわたって使用されているデータソースが表示されます。

- **日ごとのデータソースの合計使用量 -** これは、設定されたフィルターに基づいて、日ごとのデータソースの合計使用量を示しています。カウントを表示するには、線上の点にマウスのポインターを置きます。選択に基づいてビューの他のセクションを更新するための点を選択します。

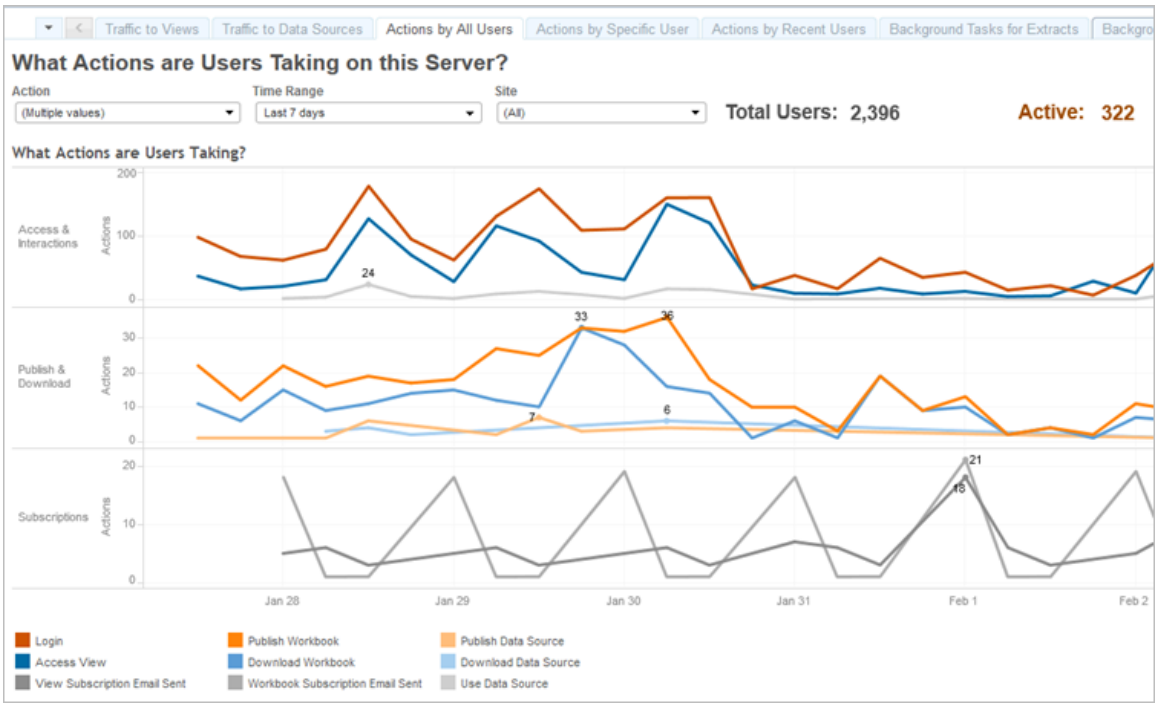
ビューの下部にある2つの棒グラフには、ビューの上部に **[最小限の操作]** フィルターによってフィルターリングされた結果が表示されます。これらは、最も使用されたデータソースや、どのユーザーが最も頻繁にデータソースを使用したかを示します。最小限の操作以上の操作カウントを持つデータソースとユーザーのみが表示されます。

- **最も使用されたデータソース -** これは、最も使用されたデータソースのリストです。ビューの他のセクションと同様、フィルターや任意の選択によって情報が制限されます。

- **最もデータソースを使用したユーザー -** これは、最もデータソースを使用したユーザーを示します。これは、フィルターやユーザーによる選択によって影響を受けます。

すべてのユーザーによるアクション

[すべてのユーザーによるアクション] ビューには、Tableau Server のインストールがどのように使用されているかに関する詳細情報が表示されます。ビューは、アクションまたは時間範囲ごとにフィルターリングできます。サーバー管理者はサイトごとにフィルターリングできます。[合計ユーザー数] は、アクションを実行したユーザー数を表します。この値は、フィルターリングの影響を受けません。[アクティブユーザー数] には、選択したアクションのいずれかを実行したアクティブユーザーの数が表示されます。



タイムラインの最大 3 つの別個のグループには、指定した時間範囲 (デフォルトは過去7日間) にわたって使用されている Tableau Server が表示されます。特定のグループに対してアクションが選択されていない場合、そのグループは表示されません。可能なグループは次のとおりです。

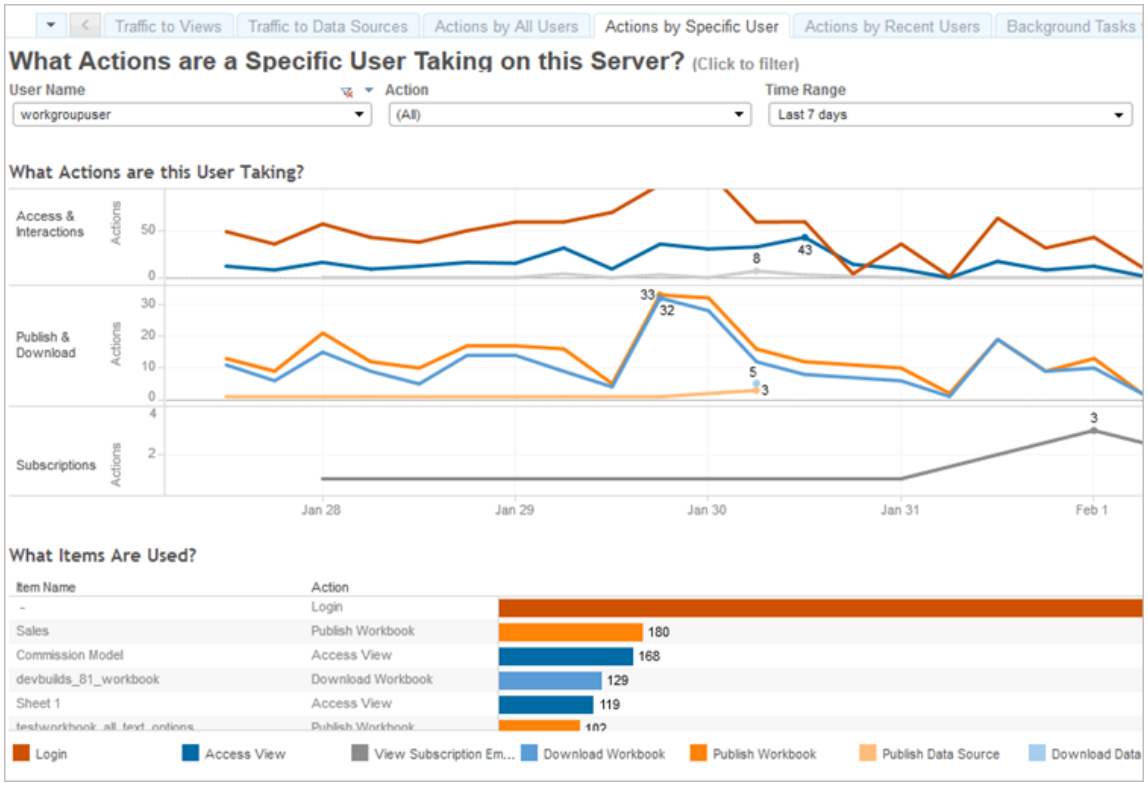
- **アクセスおよび操作 -** サインイン (ログオン) アクティビティ、ビュー アクセス、およびデータソースの使用について表示します。

- **パブリッシュおよびダウンロード -** フロー、ワークブックおよびデータ ソースのパブリッシュとダウンロードを表示します。
- **サブスクリプション -** ワークブックとビュー向けに送信されたサブスクリプション メールの数を表示します。また、フローの実行数も表示します。

表示されたアクションのサブセットを表示するには、下部にある凡例を使用します。1 つのアクションをクリックしてそのアクションの行をハイライトするか、複数のアクションを **Ctrl + クリック**して複数の行をハイライトします。選択を消去して、選択したすべてのアクションを表示するには、凡例内の任意のアクションをクリックします。

特定のユーザーによるアクション

[特定のユーザーによるアクション] ビューは、Tableau Serverのインストールで個々のユーザーがどのように作業をしているかに関する詳細について表示します。ビューはユーザー名、アクション、および時間範囲ごとにフィルターリングできます。マルチサイト インストール上のサーバー管理者は、サイトごとにフィルターリングできます。



タイム ラインの最大 3 つの別個のグループには、指定した時間範囲 (デフォルトは過去7日間) にわたって選択したユーザーが使用しているTableau Server が表示されます。特定のグループに対してアクションが選択されていない、または何のアクションも実行されていない場合、そのグループは表示されません。可能なグループは次のとおりです。

- **アクセスおよび操作 -** サインイン (ログオン) アクティビティ、ビュー アクセス、およびデータ ソースの使用について表示します。つまり選択したユーザーによるデータ ソースのすべての操作で、これにはそのユーザーまたはそのデータ ソースに関連付けられているワークブックにアクセスするユーザーがスケジュールした抽出更新が含まれます。
- **パブリッシュおよびダウンロード -** フロー、ワークブックおよびデータ ソースのパブリッシュとダウンロードを表示します。
- **サブスクリプション -** ワークブックとビュー向けに送信されたサブスクリプション メールの数を表示します。また、フローの実行数も表示します。

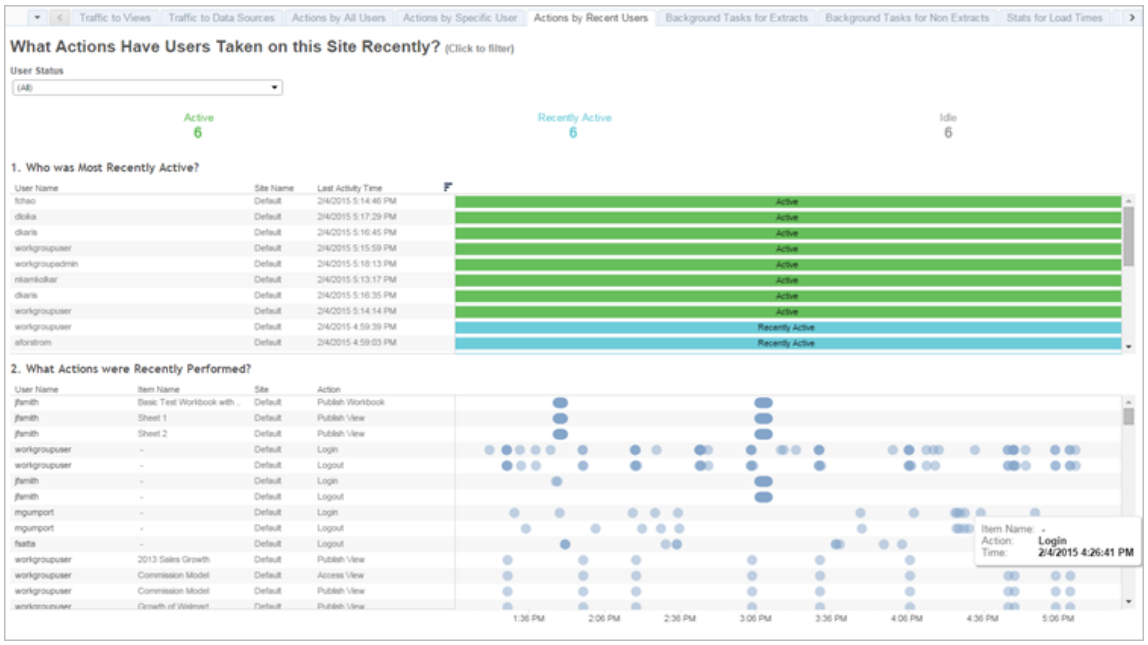
ビューの下部にある棒グラフは、選択したユーザーが使用しているアイテムを示します。

表示されたアクションのサブセットを表示するには、下部にある凡例を使用します。1 つのアクションをクリックしてそのアクションの行をハイライトするか、複数のアクションを **Ctrl + クリックして**複数の行をハイライトします。選択を消去して、選択したすべてのアクションを表示するには、凡例内の任意のアクションをクリックします。

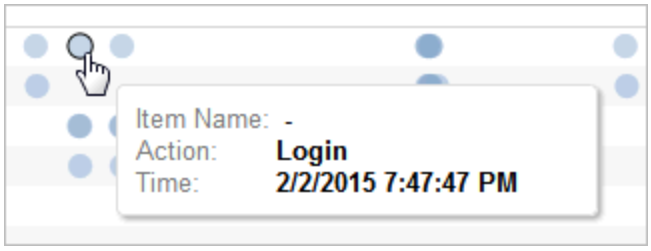
最近のユーザーによるアクション

[最近のユーザーごとのアクション] ビューには、過去 24 時間に Tableau Server でアクティブだったサインイン済みユーザーが表示されます。メンテナンス活動を実行する必要があり、影響が及ぶユーザー数、ユーザー名、さらにはそれらのユーザーが Tableau Server 上でどのような活動をしているのかについて知るのに役立ちます。

Tableau Server に現在サインインしている、**[アクティブ]**、**[最近アクティブ]**、および **[アイドル]** のユーザーを示すビュー。このビューでは、アクティブなユーザーとは直近の 5 分間にアクションを実行したユーザー、最近アクティブなユーザーとは直近の 30 分間にアクションを実行したユーザー、アイドルなユーザーとは、最後にアクションを実行してから30 分以上が経過したユーザーを指します。アクションは、ビューの下部に表示されます。



ユーザーを選択すると、そのユーザーが最近実行したアクションのみを表示できます。アクションの詳細を表示するには、アクションの上にマウスを置きます。



抽出でのバックグラウンドタスク

[抽出でのバックグラウンドタスク] ビューには、サーバー上で実行される抽出特有のタスクが表示されます。

このビューを理解する

この組み込みの管理ビューについて理解を深めるため、以下の点に留意してください。

- [このサーバーで実行した抽出] テーブルに、**[タイムライン]** に指定された期間で実行される抽出がリストされます。
- ステータスに基づいてテーブルをフィルターするには、**[成功]** または **[エラー]** をクリックできます。
- また、特定のタスクをクリックして選択したタスクの [抽出にかかった時間] グラフを更新することもできます。
- [成功または失敗した抽出の数] テーブルは、タスクのステータス (成功または失敗) を更新しますが、成功または失敗した抽出のカウントは変更されません。

ステータス

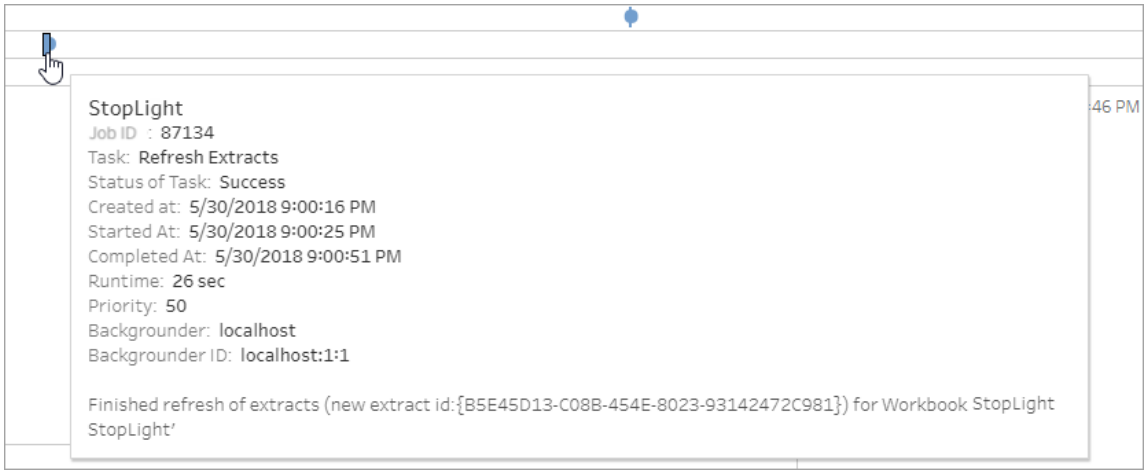
タスクには、成功またはエラーのステータスがあります。

アイコン	説明
*	**エラー -** サーバーでタスクを完了できませんでした。

アイコン	説明
	成功 - サーバーでタスクが完了しました。

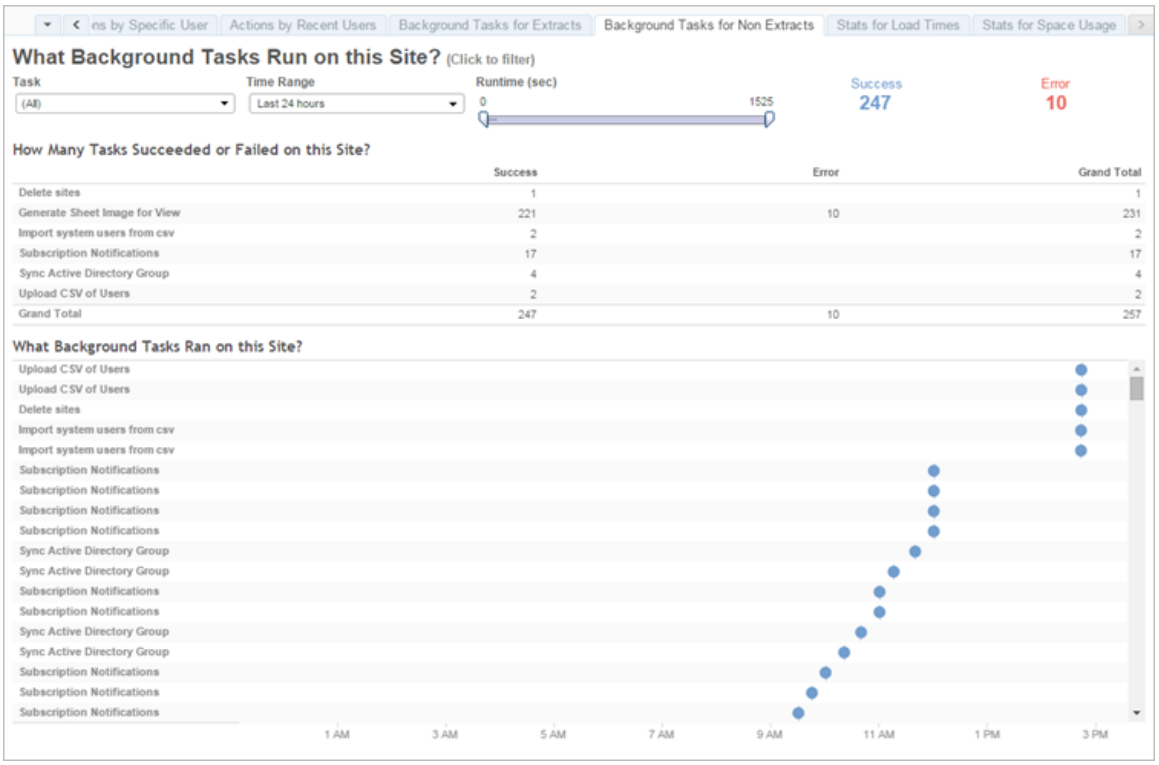
タスクの詳細を見る

タスクの詳細を確認するには、カーソルを成功アイコンまたはエラー アイコンに合わせます。



抽出以外のバックグラウンド タスク

[抽出以外のバックグラウンド タスク] ビューには、サーバーで実行されていて、かつ、抽出更新に関係のないタスクが表示されます。たとえば、、編集済みの OAuth 接続、サブスクリプション通知などです。



テーブルに、指定された期間で実行されるタスクがリストされます。ステータスに基づいてテーブルをフィルターするには、**[成功]** または **[エラー]** をクリックします。選択したタスクの **[このサイトで実行されたバックグラウンド タスク]** グラフを更新するには、**[このサイトで成功または失敗したタスクの数]** で特定のタスクをクリックします。

タスクには、成功またはエラーのステータスがあります。タスクの詳細を確認するには、カーソルを成功アイコンまたはエラー アイコンに合わせます。

アイコン	説明
	エラー - サーバーでタスクを完了できませんでした。
	成功 - サーバーでタスクが完了しました。

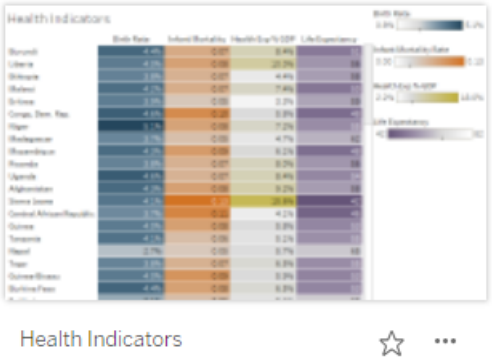
タスクに関して表示可能な内容は、タスクの ID、ステータス、優先度、タスクの作成時期、開始時期、完了時期です。また、ジョブのランタイムも表示でき、バックグラウンド ジョブの合計ランタイム (ジョブのランタイムに加え、初期化やクリーンアップなどのバックグラウンド ジョブのオーバーヘッドを含

む) も表示されます。さらに、どのバックグラウンダーでジョブを実行しているかを確認することもできます。

サムネイルのアップグレードジョブ

Tableau Server バージョン 2019.1 以降では、ワークブックおよびビューは解像度がさらに高いサムネイルを使用しています。Tableau Server バージョン 2019.1 以降にアップグレードする場合、バックグラウンダーはサムネイルのアップグレードのキューへの登録ジョブとサムネイルのアップグレードジョブを実行しますが、Tableau version 2018.2 以降で編集したりパブリッシュしていないビューのサムネイルを更新し、以下の図に示すように低解像度のサムネイル画像 (192 x192 ピクセル) から高解像度のサムネイル画像 (300 x 300 ピクセル) に変換します。

注: Tableau Server バージョン 2020.2 以降では、サムネイルのアップグレードジョブは既定でオフになっています。サーバー管理者は tabcmd を使用して、サムネイルのアップグレードジョブを実行できます。詳細については、upgradethumbnailsを参照してください。



サムネイルのアップグレードジョブによりサムネイル画像が更新されると、ビューの変更日がサムネイル更新日に一致するように変更されます。サムネイルのアップグレードジョブによって、含まれているビューのサムネイルが更新されてもワークブックの変更日が変更されることはありません。

サムネイルのアップグレードジョブは、所有するワークブック数によって異なりますが、数分から数時間実行可能です。バックグラウンダー ノードでの CPU 使用率は、サムネイルのアップグレードジョブの実行中は通常より高くなることがあります。また、Tableau Server 2019.1 以降にアップグレー

ドして最長 1 週間、サムネイルのアップグレードジョブはワークブックで使用されているすべてのサムネイル画像を処理しており、サムネイル画像が低解像度で表示される場合があります。Tableau Server をバックアップする前に、サムネイルのアップグレードジョブの完了を待機することをお勧めします。

サムネイルのアップグレードジョブの初回実行後は、事前に定義した毎週のスケジュールに従って実行します。サムネイルのアップグレードジョブの実行優先度は最も低く、background_jobs 表に含まれるワークブックごとに 1 つのタスクを作成し、低解像度のサムネイルすべてをアップグレードします。Tableau Desktop バージョン 2018.3 以前で Tableau Server にパブリッシュされた低解像度サムネイルは自動的に、毎週、サムネイルのアップグレードジョブの実行時にクリーンアップされます。

トラブルシューティング

抽出以外のバックグラウンドタスク管理ビューを使用し、サムネイルのアップグレードジョブのステータスを確認できます。サムネイルのアップグレードジョブには、成功ステータスまたはエラーステータスのどちらかが表示されます。

サムネイルのアップグレードジョブが失敗するか、完了したものの一部のサムネイルがまだ低解像度の場合。

認証資格情報が間違っている場合、サムネイルのアップグレードジョブにエラーステータスが表示されることがあります。このような場合には、ワークブックサムネイルがまだ不明瞭な低解像度 (192 x192 ピクセル) の状態で表示されます。認証資格情報を更新すると、サムネイルのアップグレードジョブを次回実行するときにワークブックサムネイルが更新されます。

バックグラウンドタスクの遅延

> **注:** このビューは、サーバー管理者のみが使用できます。マルチサイト展開でサーバービューにアクセスするには、サイトメニューをクリックし、**[すべてのサイトを管理]** を選択します。管理ビューへの移動方法の詳細については、管理ビューを参照してください。

[バックグラウンドタスクの遅延] ビューには、フロータスク、抽出の更新タスクおよびサブスクリプションタスクの遅延、つまり実行がスケジュールされている時刻と実際に実行される時刻の間の時間の

長さを示します。ビューを使用すると、タスクのスケジュールを分散させ、タスクを最適化してサーバーのパフォーマンスを改善できる場所の識別に役立ちます。

遅延の考えられる理由、および遅延を軽減できる可能性がある方法は次のとおりです。

- 多くのタスクが同時にスケジュールされている。ビューの例では、長い遅延を示すタスクは毎日同時にクラスター化されています。これにより、待機時間の急増が生じます。**[タイムライン]** フィルターを設定することで、時間別にタスクの遅延を表示し、同時に多数のタスクがスケジュールされている時刻を識別できます。この問題に対する解決策は、タスクをオフピーク時間に分散させ、サーバーの負荷を軽減することです。

- 実行に長い時間がかかる、および他のタスクの実行の妨げとなっている特定のタスク。たとえば、速度が低下しているデータソースに接続している、または大量のデータを処理中の抽出の更新ジョブがあるとします。**[抽出でのバックグラウンド タスク]** 管理ビューを使用して、実行速度が低下している抽出の更新タスクを識別します。データをフィルターリングおよ

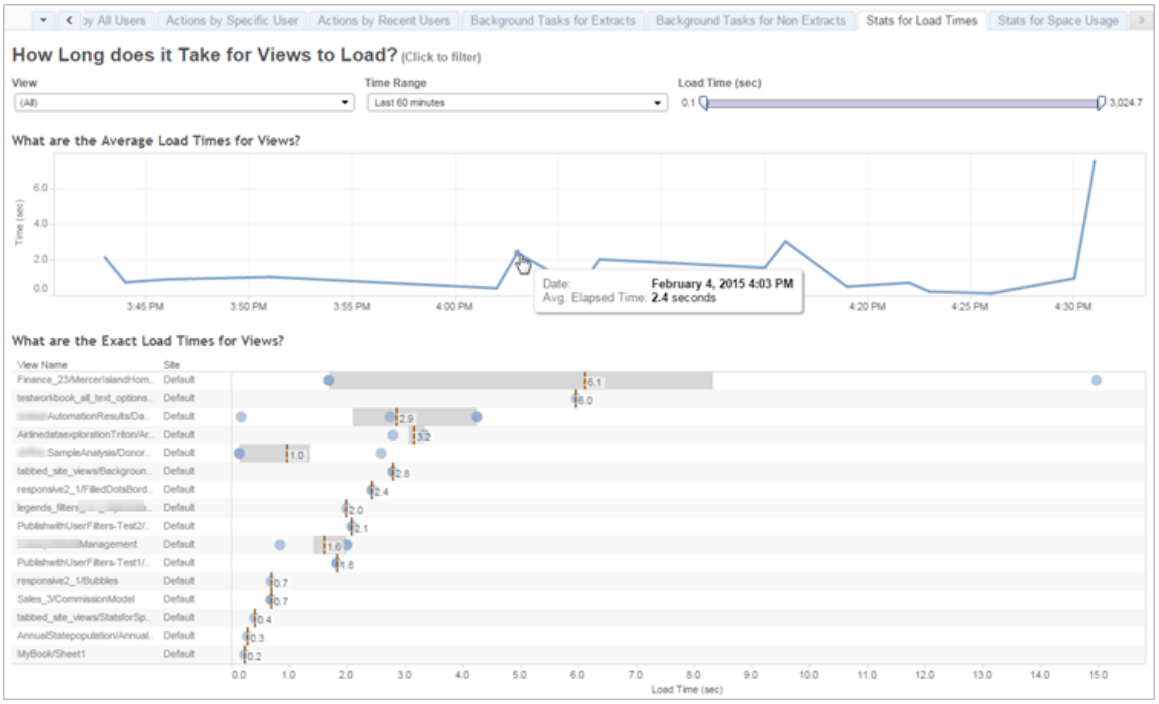
び集計し、データソース内の個別の表に対する複数のデータソースを作成して、抽出の更新タスクを最適化できます。

- 同時に実行され、サーバーリソースを消費し、パフォーマンスを低下させている他のサーバープロセス。サーバープロセスの CPU およびメモリ利用量を監視し、最もリソースを消費しているプロセスを確認してサーバー上のプロセスの構成を調節します。

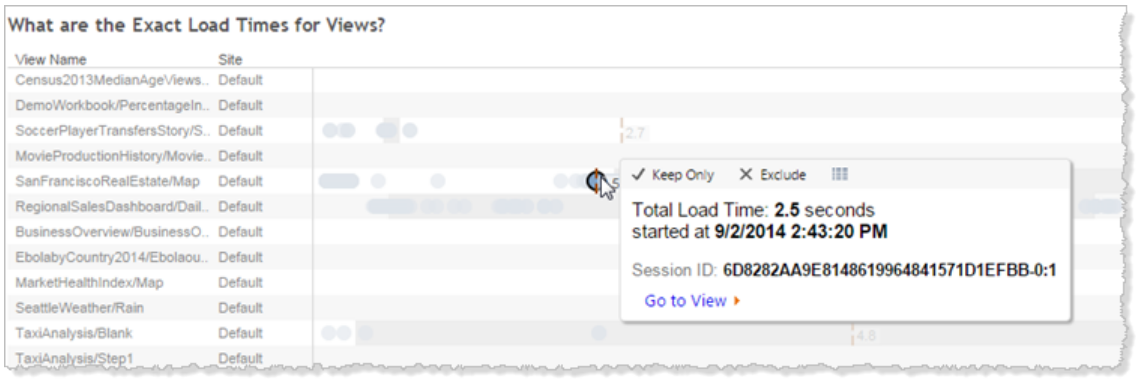
ロード時間の統計

[読み込み時間の統計] を使用すると、サーバーのパフォーマンスに最も大きな影響を与えているビューを確認できます。ビューと時間範囲でフィルターリングできます。サーバー管理者はサイトごとにフィルターリングできます。また、変動する [読み込み時間] フィルターを使用して、秒単位の読み込み時間に基づいてビューを制限することができます。読み込み時間はサーバー用です。クライアントブラウザーとネットワークに応じて、実際の読み込み時間が若干異なる場合があります。

[平均読み込み時間] グラフは、設定したフィルターに基づいてビューの平均読み込み時間を示します。詳細を参照するには点の上にマウスを移動します。線上の点を選択し、選択に関するビューの残りの部分を更新します。



[正確な読み込み時間] ビューには、リストされたビューを読み込むのにかかる正確な時間が表示されます。垂直の線は、各ビューの平均読み込み時間を示しています。ビューの読み込みの特定のインスタンスの詳細を表示するには、マークを選択します。

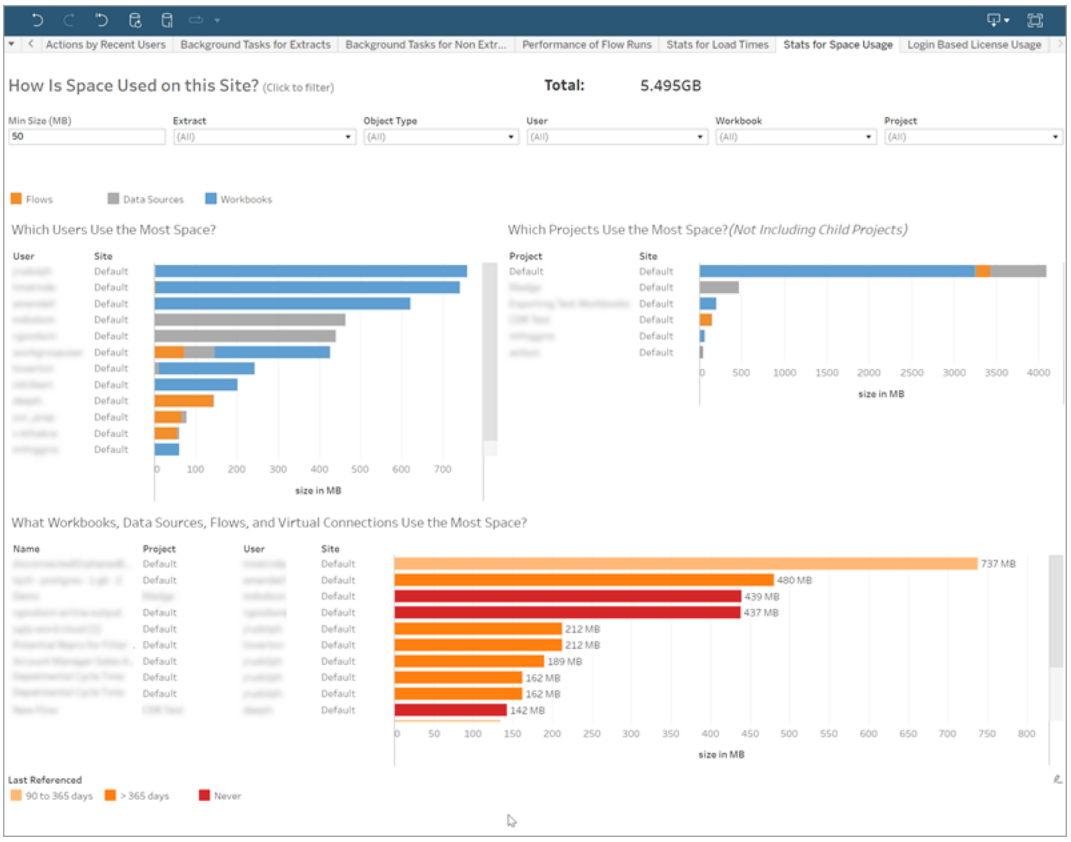


領域使用量の統計

[領域使用量の統計] ビューを使用すると、サーバーでディスク空き容量を最も多く使用している Tableau コンテンツを特定できます。ディスク空き容量の使用率は、ユーザー別、プロジェクト別、および Tableau コンテンツ (ワークブック、データソース、フロー出力、または仮想接続) のサイズ別に表示され、最も近い整数に切り捨てられます。

仮想接続には データ管理 が必要であることに注意してください。詳細については、「データ管理について について」を参照してください。

このビューを開くには、左側のナビゲーションで **[サイトの状態]** をクリックし、[ダッシュボード] で **[領域使用量の統計]** をクリックします。



[最小サイズ] フィルターを使用して、使用する領域に基づいてどの Tableau コンテンツを表示するか制御します。

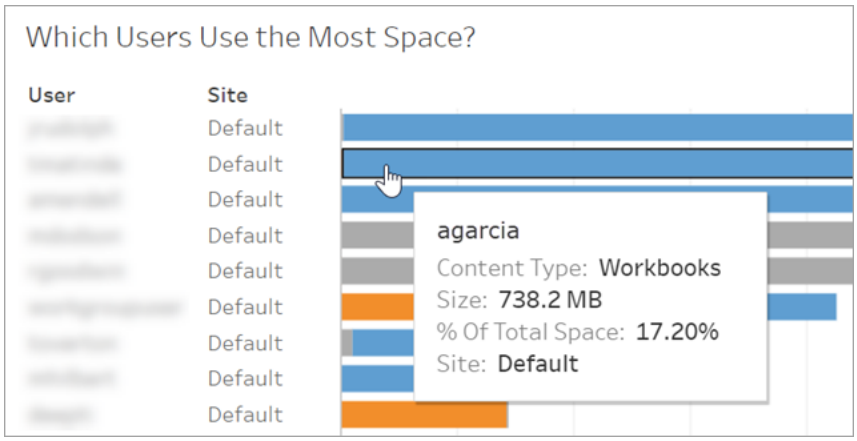
3 つの棒グラフは、Tableau Server 上の領域の使用状況についての情報を表示します。

- **領域を最も多く使用しているユーザー -** 領域を最も多く使用しているデータ ソースおよびワークブックを持つユーザーが表示されます。ユーザー名をクリックすると、次の 2 つのグラフをそのユーザーでフィルターできます。ユーザーのオブジェクトの種類に関する次の 2 つのグラフをフィルターするには、そのユーザーのデータ ソース バーまたはブックバーをクリックします。選択をクリアするには、ユーザーまたはバーをクリックします。

- **領域を最も多く使用しているプロジェクト -** 領域を最も多く使用しているデータ ソースおよびワークブックを持つプロジェクトが表示されます。[領域を最も多く使用しているユーザー] グラフで、ユーザーまたはオブジェクトのタイプが選択されている場合、その選択内容に固有の情報が表示されます。

- **領域を最も多く使用しているワークブック、データ ソース、フロー、および仮想接続** - 領域を最も多く使用しているTableau コンテンツが表示されます。棒グラフは、前回の更新からの時間の長さに基づいて色分けされています。

サイズ バー上にカーソルを移動すると、詳しい使用率が表示されます。



棒グラフをクリックして選択し、選択に基づいてビューの他の領域を更新します。

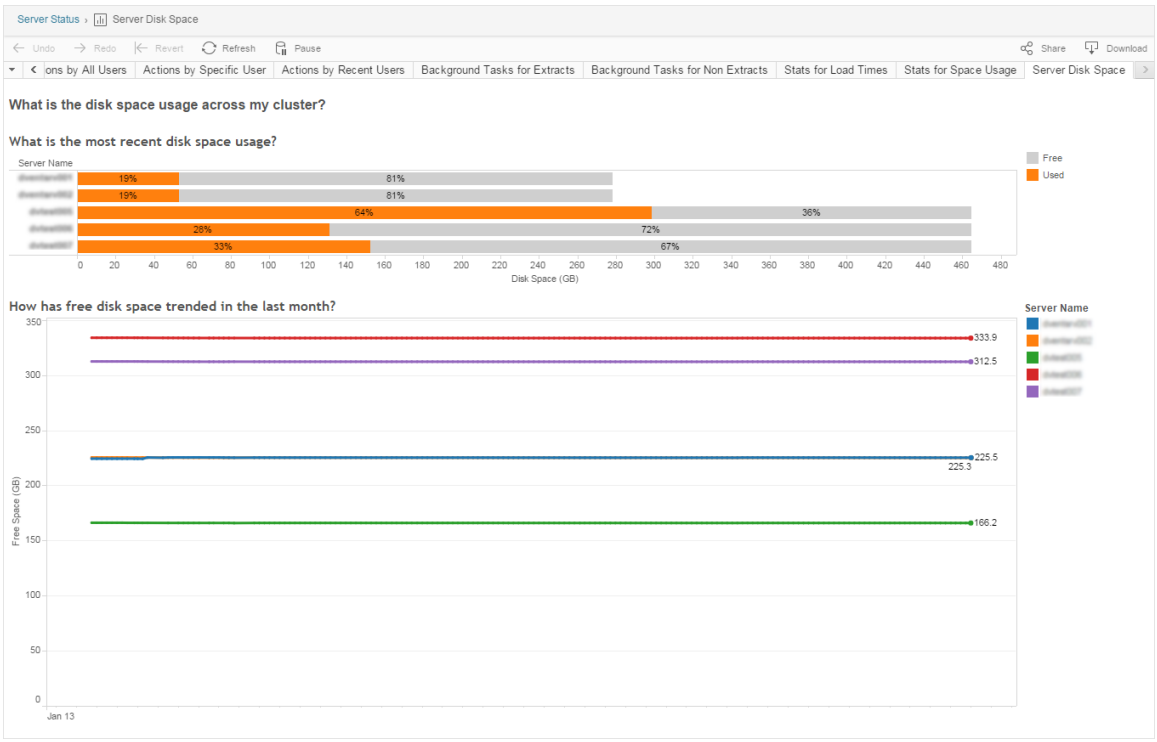
サーバーのディスク容量

> **注:** このビューは、サーバー管理者のみが使用できます。マルチサイト展開でサーバー ビューにアクセスするには、サイト メニューをクリックし、**[すべてのサイトを管理]** を選択します。管理ビューへの移動方法の詳細については、管理ビュー を参照してください。

Tableau Server を実行するコンピューターで使用中のディスク容量を見るには、[サーバーのディスク容量] ビューを使用します。ここで、ディスク領域は Tableau Server がインストールされているパーティションのみを参照します。また、このビューを使用して、ディスク使用量の突然の変更を特定することもできます。

このビューでは、ディスク領域の使用量が 10 進数の GB 値としてレポートされます。Tableau Server コンピューターのオペレーティング システムがバイナリGB を使用してレポートを報告する場合、使用量の値が異なる場合があります。

分散インストールの場合は、クラスタ内の各コンピューターに関する情報がビューに表示されます。



[サーバーディスク容量] ビューには 2 つのグラフが含まれます。

- **最新のディスク使用量は?—** このグラフは、過去 30 日間のディスク使用量をギガビットとパーセンテージの両方で示します。ディスク容量は、Tableau Server がインストールされているパーティションのみを参照します。
- **先月のディスク容量の傾向は?—** このグラフは、先月のディスク使用量の変化を示します。ある時点の正確なディスク空き容量を見るには、ラインにポインターを重ねます。

Tableau Server のディスク容量が低下すると、ファイルを削除して空き容量を増やすことができます。

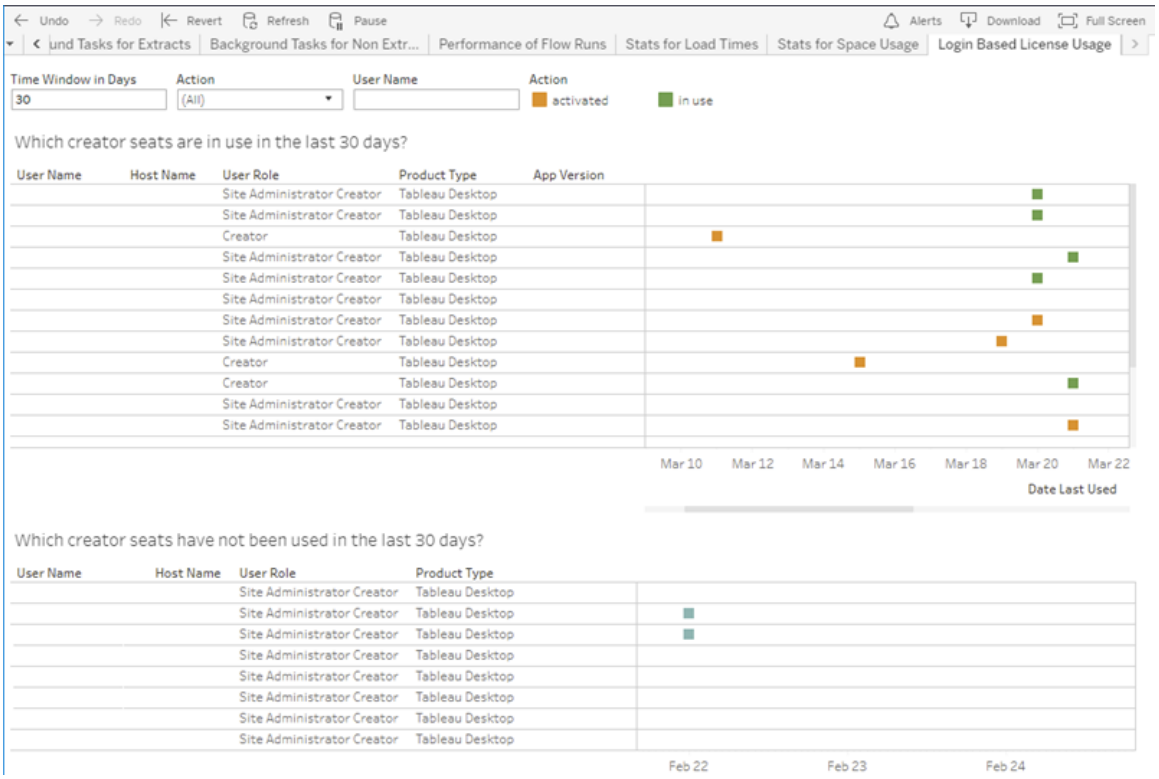
ヒント: ディスク空き容量が指定したしきい値を下回ったときに Tableau Server 通知を受信できます。詳細については、サーバーイベント通知の構成を参照してください。

ログインベースのライセンス使用状況

注: ログインベースのライセンス管理 が Tableau Server で有効になっている場合、このビューは、サイト管理者 とサーバー管理者のみが使用できます。管理ビューへの移動方法の詳細については、管理 ビュー を参照してください。

[ログインベースのライセンス使用状況] ビューでは、サーバー管理者は、Tableau Cloud または Tableau Server のログインベースのライセンス認証の使用状況を表示できます。[ログインベースのライセンス使用状況] ビューでは、ライセンスをより効率的に管理し、ライセンス数の追加や削減の必要性を判断できます。このビューを使用することで、次の質問の答えが得られます。

- 組織内で Tableau Desktop または Tableau Prep Builder ライセンスを使用しているユーザーは誰か?
- Creator ロールは共有または転送されていますか?
- 使用されるべきではないコンピューターでアクティブ化が行われたライセンスはありますか?
- どのホストでアクティブ化を行っていますか?
- ユーザーに割り当てられているのはどのロールか?
- どの Tableau 製品でライセンスが使用中か?
- どの Tableau バージョンでライセンスが使用中か?
- Creator ロールを、Tableau Desktop または Tableau Prep Builder でアクティブ化しましたか。
- Creator シートは有効になっていますか?
- 使用中の Creator シートはいくつあるか?
- 使用されていない Creator シートはいくつあるか?
- Creator シートが最後に使用されたのはいつか?



ログインベースのライセンス使用状況管理ビューを使用できるほかに、Tableau Server リポジトリの「ワークグループ」PostgreSQL データベース内のログインベースのライセンス使用状況データ (`identity_ based_activation_reporting identity_based_activation_user_role_change` および `identity_based_activation_admin_view`) にアクセスすることもできます。このデータにアクセスするには、Tableau Server リポジトリへのアクセスを有効にする必要があります。

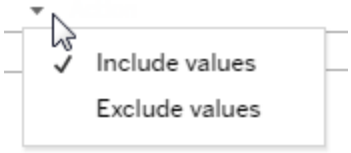
フィルター

レポート画面では、シートが最後に使用された日時を示す時間枠の変更、アクションのフィルター、ユーザー名のフィルター、列による並べ替えを行うことができます。

- **日数で表示されたタイム ウィンドウです。** ログインベースのライセンス管理によってアクティブ化されたクライアントの利用状況データを表示する日数を入力します。過去 30 日 ~ 183 日間のデータを表示できます。
- **(すべて) 。** すべてのフィルターをビューに適用します。

- **ライセンス認証済み。**ログインベースのライセンス管理 を使用してアクティブ化 された Creator ユーザーを表示します。
- **使用中**。ログインベースのライセンス管理 を使用してアクティブ化 され、シートが使用中になっている Creator ユーザーを表示します。
- **最終使用日**。ログインベースのライセンス管理 クライアントが最後に使用 された日を表示します。
- **未割り当て**。ログインベースのライセンス管理 を使用してアクティブ化 された Creator シートのうち、現在割り当てられていないシートを表示します。
- **ユーザー名。**指定したユーザーが使用中の ログインベースのライセンス管理 によるアクティブ化 を表示します。

フィルター カードをポイントすると、ドロップダウン アイコンが表示 されます。アイコンをクリックし、フィルターと一致するデータを含める (既定) か、フィルターと一致するデータを除外するかを指定します。



過去<nn>日間に使用 された Creator シートはどれか?

ダッシュボードのこの領域には、3 種類のライセンス (ライセンス認証、使用中、未割り当て) の一覧が表示 されます。ライセンス認証、使用中、未割り当てのマークをポイントすると、Tableau のコピーの登録ユーザーを含む情報が表示 されます。列見出しをクリックして、リストを並べ替えます。

過去 <nn> 日間使用 されていない Creator シートはどれか

ダッシュボードのこのエリアには、指定した期間に使用 されたライセンスのリストが表示 されます。タイムラインに、最後に使用 された日付が表示 されます。[最終使用日] マークをポイントすると、Tableau のコピーの登録ユーザーを含む情報が表示 されます。

デスクトップ ライセンスの使用率

注: このビューは、サーバー管理者のみが使用できます。マルチサイト展開でサーバー ビューにアクセスするには、サイト メニューをクリックし、**[すべてのサイトを管理]** を選択します。管理ビューへの移動方法の詳細については、管理ビュー を参照してください。

サーバー管理者は [デスクトップ ライセンスの使用量] ビューで、Tableau Desktop ライセンスの使用量データを確認できます。これにより、ライセンスをより効率的に管理し、ライセンス数の追加や削減の必要性を判断できます。このビューを使用することで、次の質問の答えが得られます。

- 組織内で Tableau Desktop ライセンスを使用しているユーザー
- ライセンスが共有または譲渡されているか?
- 使用されるべきでないコンピューターで使用されているライセンスはあるか?
- 特定のユーザーがこのライセンスを使用しているか?
- 組織内で使用されているライセンスの種類は?
- 試用版 ライセンスを変換する必要はあるか?

注: ライセンスに関するデータを取得するには、Tableau Server にデータを送信するよう Tableau Desktop バージョン 10.0 以降の各コピーを設定する必要があります。この設定はインストール時に、Tableau のインストールおよび設定を行うスクリプトまたはサードパーティ製ソフトウェアを使用して行うことも、インストール後にレジストリやプロパティ リスト ファイルを変更して行うこともできます。詳細については、「Desktop ライセンス レポートの構成」を参照してください。

ライセンス データを表示するには、Tableau Server でデスクトップ ライセンス レポートを有効化する必要があります。「ライセンス レポートの有効化と構成」を参照してください。

フィルター

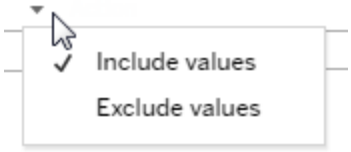
- **プロダクト キー**。文字列を入力してダッシュボードをフィルターし、プロダクト キーの任意の場所に文字列を含むライセンスのみを表示します。たとえば、TDTD で開始するライセンスのみを表示するには、「TDTD」と入力してから、Return を押してビューをフィルターします。フィルターをリセットするには、文字列の後の「**X**」をクリックします。

- **アクション**。このフィルターを使用し、ダッシュボードが表示する内容を、実行するアクションに基づいてコントロールします。アクションは、**[ライセンス認証]**、**[使用]**、または **[戻る]** (ライセンス認証の解除) です。**[使用]** アクションが選択されていない場合、一番上の棒グラフには何も表示されません。

- **部門**。このフィルターを使用し、ダッシュボードがライセンスを表示する部門をコントロールします。フィルターは、Tableau Desktop の登録時に指定された **[部門]** 値に基づいて生成されます。

- **期間 (日) の選択**。このスライダーを使用し、ダッシュボードが情報を表示する期間を日単位で指定します。既定値は 183 日です。

最初の 3 つのフィルターのフィルター カードをポイントすると、ドロップダウン アイコンが表示されます。アイコンをクリックし、フィルターと一致するデータを含める (既定) か、フィルターと一致するデータを除外するかを指定します。

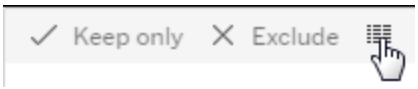


過去 <nn> 日間に Tableau を使用したユーザー

ダッシュボードのこのエリアには 3 種類の Tableau Desktop ライセンス (恒久版、試用版、期間指定版) の棒グラフと、指定した期間中に各ライセンスを使用したユーザー数が表示されます。ライセンス タイプ セグメントをポイントすると、ライセンス タイプの説明が表示されます。セグメントをクリックすると、残りのダッシュボードでそのライセンス タイプのみがフィルターされます。この操作は、使用されているライセンスと使用されていないライセンスを示す両方の表をフィルターします。たとえば、期間中に使用された期間 ライセンスのリストを表示するには、[期間] バーをクリックします。"使用された" および "使用されていない" リストは、期間 ライセンスのみを示すようフィルターされます。

詳細情報の表は棒グラフの下に表示されます。表の各行について、アクションアイコンは右側の、アクションが最後に実行された時期を示すタイムラインの上に表示されます。

メールやプロダクトキーなどの値を選択してコピーできる形式で、参照元データのリストを表示するには、ライセンスのリストで行をクリックしてから、[データの表示] アイコンをクリックします。



データがサマリー形式で表示されます。**[全データ]** をクリックしてすべてのデータを表示します。このビューから個々の値を選択してコピーしたり、データをテキストファイル形式でダウンロードできます。

過去 <nn> 日間使用されていないライセンス

ダッシュボードのこのエリアには、指定した期間に使用されたライセンスのリストが表示されます。タイムラインに、最後に使用された日付が表示されます。[最終使用日] マークをポイントすると、Tableau のコピーの登録ユーザーを含む情報が表示されます。

デスクトップ ライセンスの有効期限

> **注:** このビューは、サーバー管理者のみが使用できます。マルチサイト展開でサーバー ビューにアクセスするには、サイト メニューをクリックし、**[すべてのサイトを管理]** を選択します。管理ビューへの移動方法の詳細については、管理ビュー を参照してください。

サーバー管理者は [デスクトップ ライセンスの有効期限] ビューを使用して、組織内の Tableau Desktop ライセンスで、ライセンスが切れているものやメンテナンスの更新が必要なものに関する情報を確認できます。これにより、より効率的にライセンスを管理することができます。このビューを使用することで、次の質問の答えが得られます。

- 期限切れの試用版ライセンスまたは期間ライセンス
- メンテナンス期限が切れた恒久版ライセンスはどれですか。
- メンテナンス更新が近づいている恒久版ライセンスはどれですか。

ライセンスを更新するか更新に関する追加情報を入手するには、「Tableau ライセンスの更新方法」を参照してください。

注: ライセンスに関するデータを取得するには、Tableau Server にデータを送信するよう Tableau Desktop バージョン 10.0 以降の各コピーを設定する必要があります。この設定はインストール時に、Tableau のインストールや設定を行うため、スクリプトやサードパーティ製ソフトウェアを使用して実行できます。詳細については、「Desktop ライセンス レポートの構成」を参照してください。

ライセンス データを表示するには、Tableau Server でデスクトップ ライセンス レポートを有効化する必要があります。「ライセンス レポートの有効化と構成」を参照してください。

フィルター:

- **プロダクト キー-** 文字列を入力してダッシュボードをフィルターし、文字列を含むライセンスのみを表示します。たとえば、TDTD で開始するライセンスのみを表示するには、「TDTD」と入力してから、Return を押してビューをフィルターします。文字列の最後にある "x" をクリックしてフィルターをリセットします。
- **部門 -** このフィルターを使用し、ダッシュボードがライセンスを表示する部門をコントロールします。フィルターは、Tableau Desktop の登録時に使用された [部門] 値に基づいて生成されます。
- **期間 -** このフィルターを使用し、ダッシュボードが情報を表示する時間の長さをコントロールします。

このビューには、ビューの一番上で設定されたフィルターの影響を受ける、次の表が含まれます。

- **メンテナンス有効期限切れのキー—** この表は、メンテナンスの有効期限が切れたプロダクト キーを示し、縦線は 6 か月のメンテナンス更新期間が終了する時期を示します。キーのメンテナンス有効期限から 6 か月以上が経過している場合、サポートまたはアップグレード

を利用するには新しくキーを購入する必要があります。

- **期限切れの試用版ライセンスまたは期間ライセンス** — 期限切れの試用版または期間プロダクトキーを示します。
- **自分のキーのメンテナンス スケジュール** — キーとメンテナンスの状態を示します。

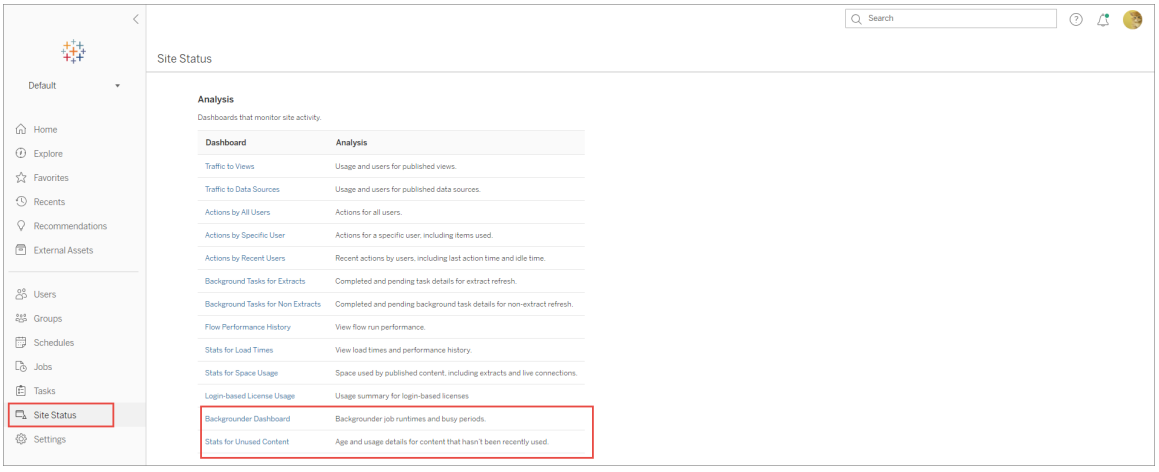
バックグラウンダーダッシュボード

バックグラウンダーダッシュボードビューは、バックグラウンドジョブの概要です。このビューを使用すると、次の情報を参照できます。

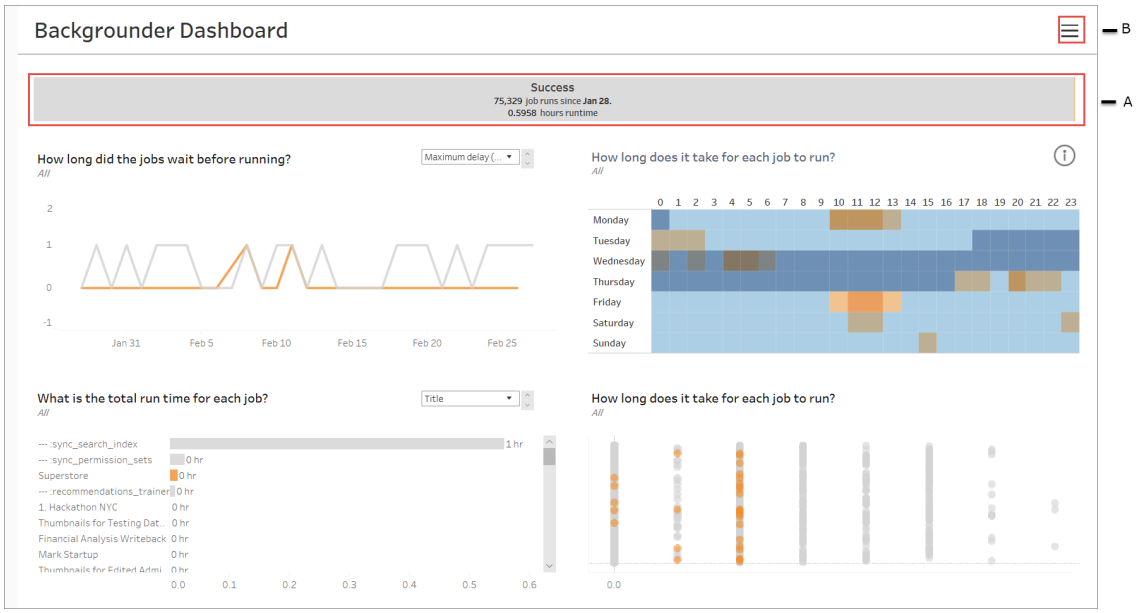
- ジョブの実行にかかる時間。
- バックグラウンダーがビジー状態または過負荷の場合。
- 正常に完了したジョブ、失敗したジョブ、または取り消されたジョブ。

> 注: このビューには、フロー実行ジョブは含まれません。

以前のバージョンの Tableau Server では、管理ビューは同じワークブック内の別のタブに表示されていました。しかし、新しい 2 つの管理ビューは、既存の管理ビュー ワークブックの一部ではなく別個のワークブックとして表示されます。新しい管理ビューには、サーバーやサイトのステータス ページから移動できます。

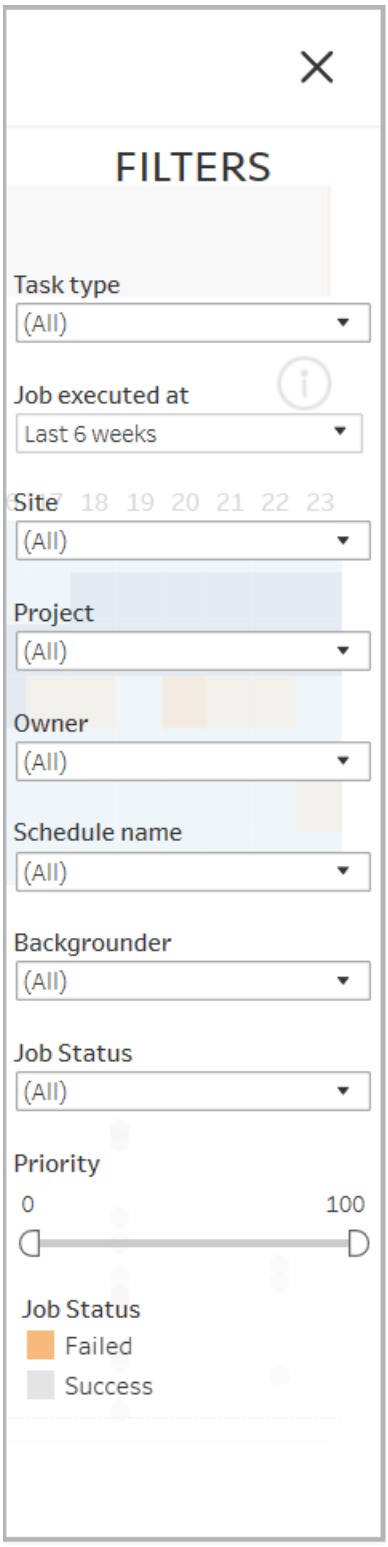


サマリーとフィルター



A- 上部には、成功、失敗、またはキャンセルされたジョブの数を示すサマリーグラフがあります。棒グラフのセクションをクリックすると、ジョブの状態で情報をフィルター処理できます。このフィルターはビュー全体に適用され、表示される情報には、選択したジョブステータスのジョブのみが含まれます。

B- フィルターアイコンをクリックすると、追加のフィルターオプションを見つけることができます。フィルターペインが表示され、**タスクの種類**、**実行されるジョブ**、**サイト**、**プロジェクト**、**コンテンツ所有者**、**スケジュール**、および**バックグラウンドID**でフィルターできます。バックグラウンドIDは、バックグラウンダープロセスに固有です。これを使用して、各バックグラウンダー処理で実行された作業に関する情報を確認できます。これらのフィルターを1つ以上選択すると、ビュー全体に適用されます。


FILTERS
Task type
(All)
Job executed at
Last 6 weeks
Site
(All)
Project
(All)
Owner
(All)
Schedule name
(All)
Backgrounder
(All)
Job Status
(All)
Priority
0
100
Job Status
Failed
Success

詳細

バックグラウンダーダッシュボードには、ジョブに関するさまざまな情報を示す 4 つのセクションがあります。これらのセクションには、適用できるドロップダウンメニューに追加のフィルターがあり、特定のセクションに適用できます。

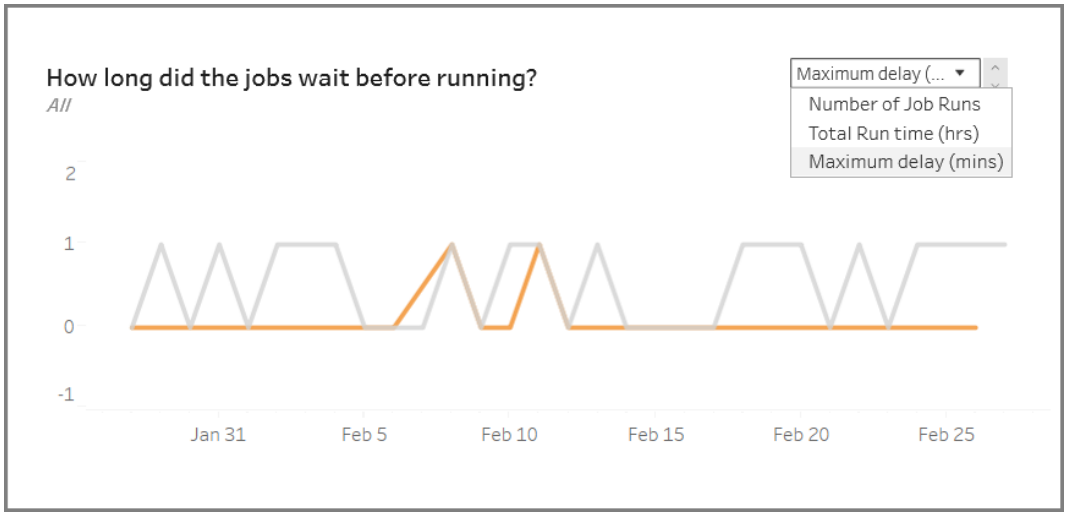
1. **左上のセクション**には、選択したジョブの状態を持つジョブ、またはフィルター ペインを使用して選択したフィルターに関する情報が表示されます。選択が行われなかった場合は、すべてのジョブが含まれます。

   表示される情報は、ドロップダウンを使用して選択した内容によっても異なります。

   1. **合計実行時間**: 選択したジョブの状態を伴うすべてのジョブの合計実行時間と、選択したその他のフィルターが表示されます。

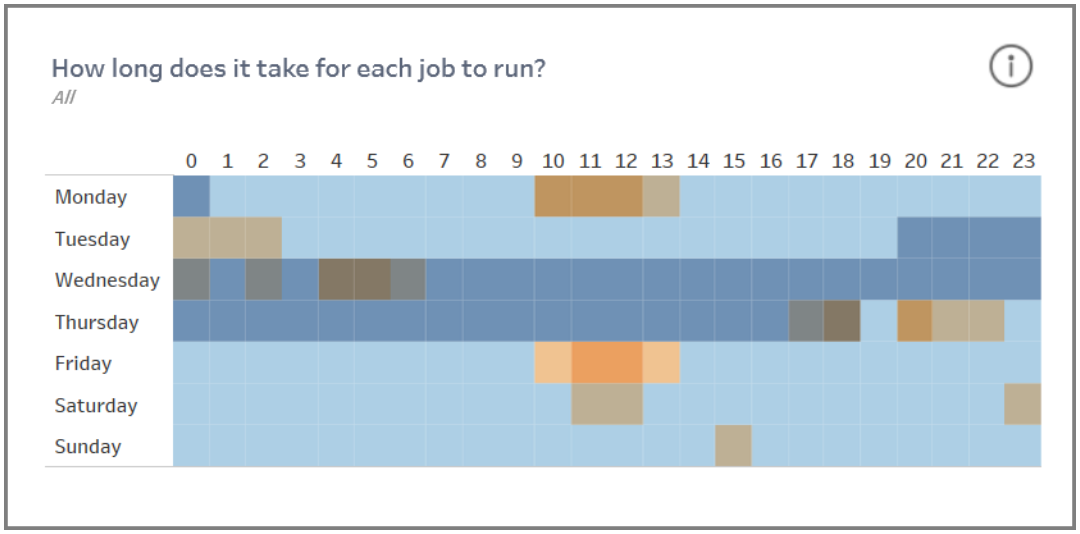
      たとえば、ジョブの状態: 失敗、およびタスクの種類 : サブスクリプション、を選択した場合、グラフは失敗したすべてのサブスクリプション ジョブを示します。

   2. **ジョブ数**: 選択したジョブの状態に対して実行されたジョブの数と、その他の選択したフィルター数を表示します。
   3. **最大遅延**: ジョブが実行されるまでの待ち時間を示します。



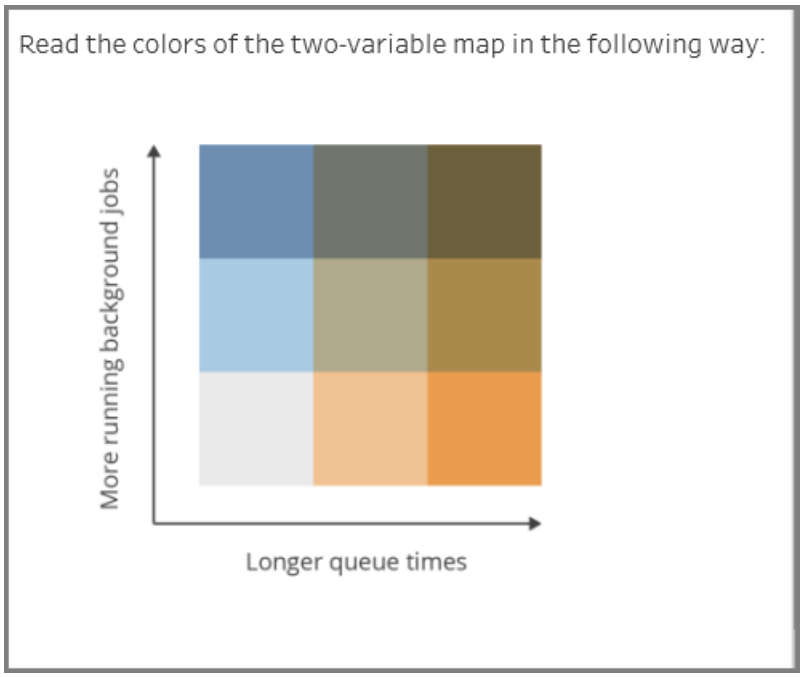
2. **右上のセクション**には、特定の日時にバックグラウンダーがどの程度ビジー状態であるか、過負荷になっているかの情報が表示されます。ただし、表示される情報は、ジョブの状態

に対して選択したフィルターと、[フィルター] ウィンドウの他のオプションによって異なります。

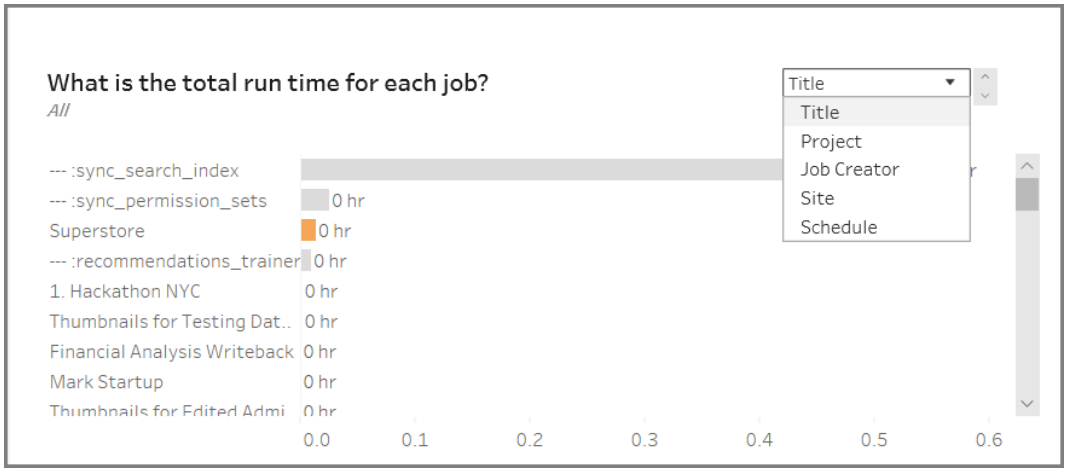


このセクションの右上にある情報アイコンをクリックすると、色の凡例が表示されます。

- 青い色の変化は、その期間に実行されているジョブの数に相関しています。青が濃いほど、実行中のジョブが多くなります。
- オレンジ色の変化は、キュー時間に相関します。オレンジ色が濃いほど、キュー時間が長くなります。
- これらのメトリクスの両方を使用して、バックグラウンダーがビジー状態 (濃い青色) または過負荷 (濃いオレンジ色) のタイミングを識別できます。



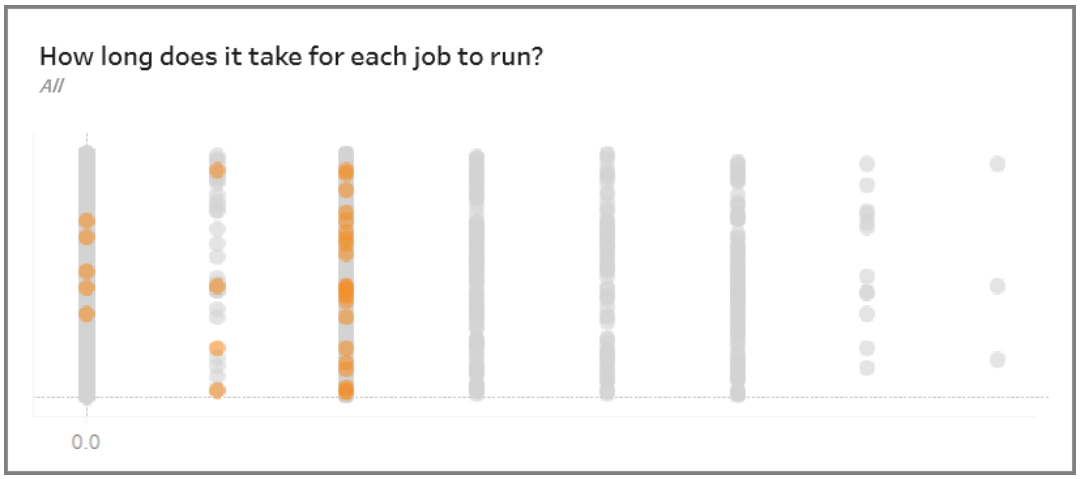
3. **左下のセクション**には、選択した状態のジョブ、またはフィルターペインを使用して選択したその他のフィルターに関する実行時情報が表示されます。選択が行われなかった場合は、すべてのジョブが含まれます。上部のセクションでマークを選択すると、選択したコンテンツの詳細がこのセクションに表示されます。



ドロップダウンの選択により、より多くのオプションが表示され、表示される情報がそれに応じて変化します。

- **[タイトル]**を選択すると、各ジョブの実行時間情報が表示されます。
- **[プロジェクト]** を選択すると、各プロジェクトの合計実行時間が表示されます。
- **[ジョブ作成者]**を選択すると、特定のユーザーによって作成されたジョブの合計実行時間が表示されます。
- **[サイト]**を選択すると、その特定のサイト上のジョブの合計実行時間が表示されます。
- **[スケジュール]**を選択すると、その特定のスケジュールを使用するジョブの合計実行時間が表示されます。

4. **右下のセクション**では、各ジョブの実行にかかった時間を示します。



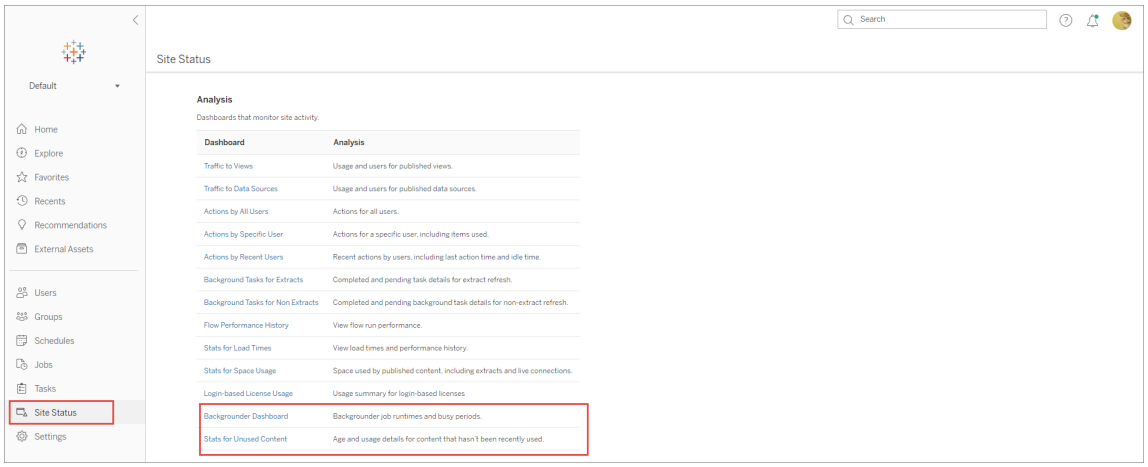
古いコンテンツ

[古いコンテンツ] ビューを使用すると、指定した期間に使用またはアクセスされていないコンテンツ (古いアクセスしきい値として表示) を識別できます。期間は日数で設定できます。期間の最小値は 1 日、最大値は 120 日です。

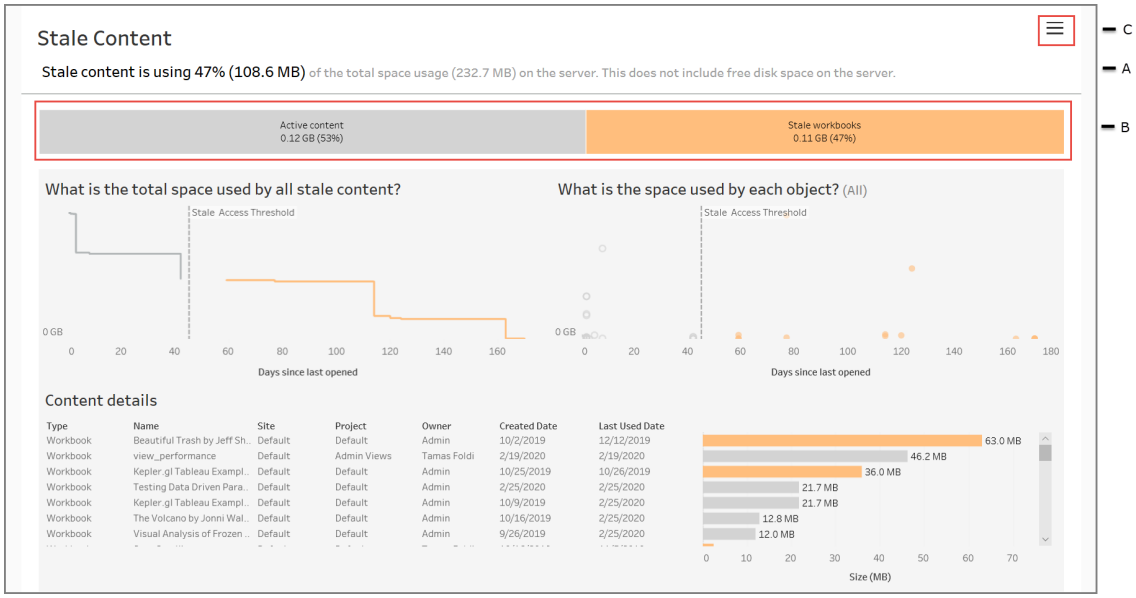
このビューには、古いコンテンツやアクティブ コンテンツが使用するディスク領域に関する情報も表示されます。

以前のバージョンの Tableau Server では、管理ビューは同じワークブック内の別のタブに表示されていました。しかし、新しい 2 つの管理ビューは、既存の管理ビュー ワークブックの一部ではなく別

個のワークブックとして表示されます。新しい管理ビューには、サーバーやサイトのステータス ページから移動できます。



サマリーとフィルター

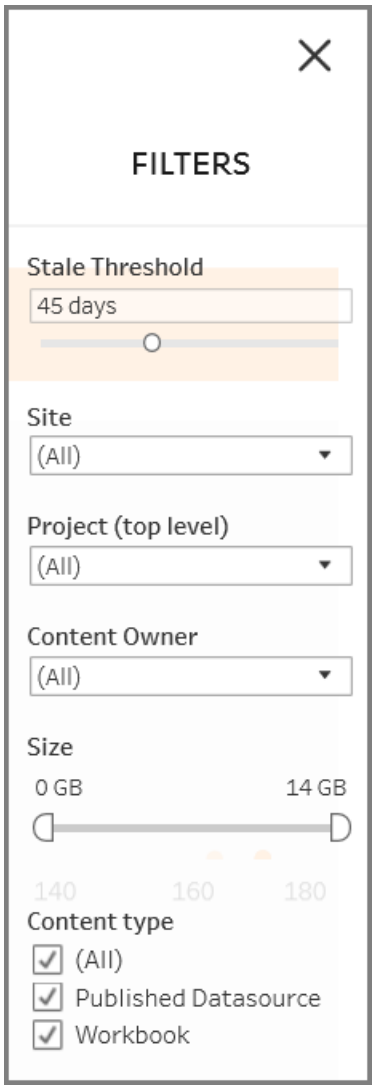


A- ビューの上部に、使用された合計スペースと比較して、古いコンテンツで使用されるスペースの量を要約したステートメントが表示されます。使用される合計領域は、アクティブなコンテンツと古いコンテンツによって使用されるディスク領域の合計として定義されます。

B- このサマリーの後に、失効したコンテンツの種類とアクティブとみなされるコンテンツの詳細を示すグラフが続きます。棒グラフをクリックして適用すると、ビューに表示されるデータをフィルター処理できます。

C- フィルター アイコンをクリックすると、追加のフィルター オプションを表示および適用できます。このフィルターペインには、次の項目が含まれます。

- 古いしきい値
- サイト
- プロジェクト
- コンテンツ所有者
- サイズ
- コンテンツタイプ



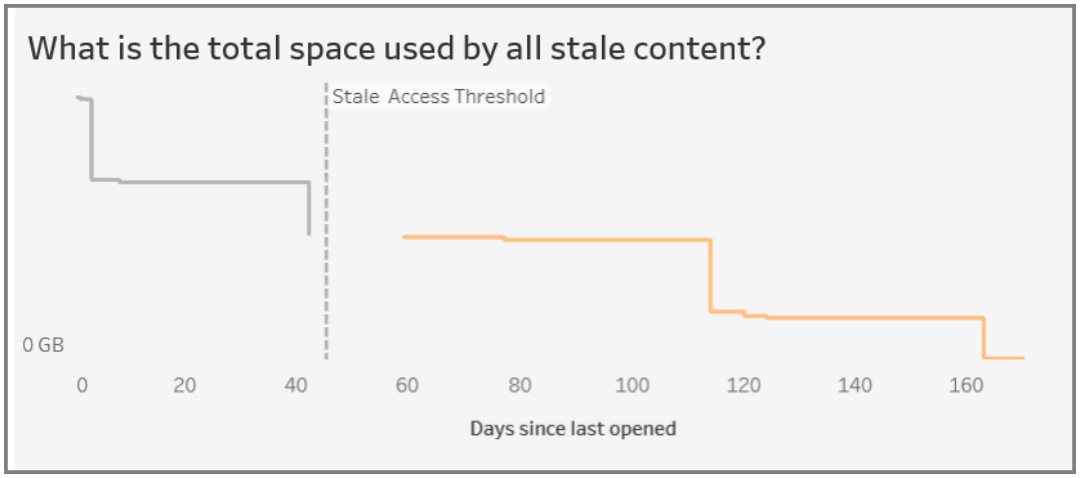
これらのフィルターはビュー全体に適用されます。

詳細

[古いコンテンツ] ビューには、以下に説明する3つのセクションがあり、詳細が表示されます。

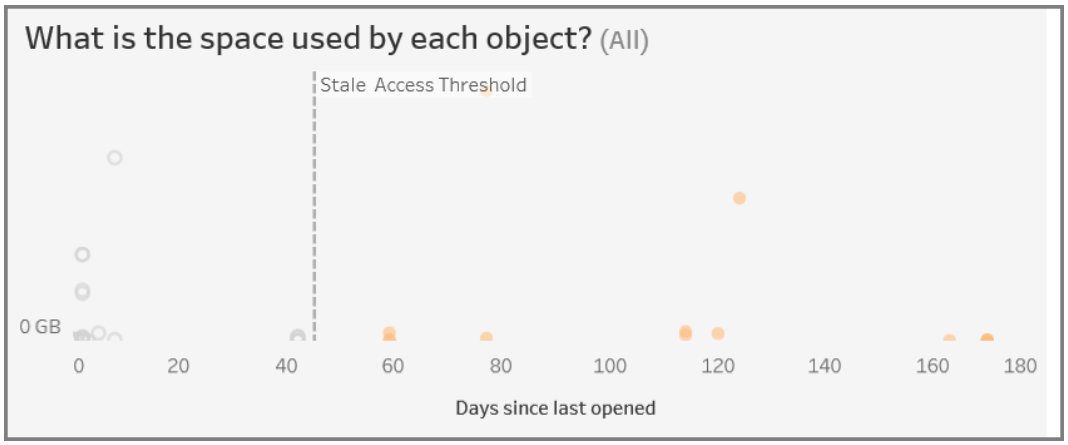
1. **左上のセクション**には、選択したコンテンツに使用された合計スペースが表示されます。x軸はコンテンツが最後に開かれてから経過した日数を示し、y軸はサイズを示します。グラフには、古いしきい値も表示されます。

目的のしきい値の古さを設定し、このビューを使用して最も古いコンテンツを特定します。上部のバーで **[古いワークブック]** や **[古いデータソース]** をクリックして、対象のコンテンツにフィルターを適用します。コンテンツの詳細を表示するには、**[古いアクセスしきい値]** の右側にあるマークを選択します。詳細は左下のセクションに表示されます。



2. **右上のセクション**には、選択した各コンテンツで使用されるスペースの量が表示されます。たとえば、[古いワークブック] を選択した場合、古い各ブックで使用される領域が表示されます。このセクションを使用して、最も古いコンテンツや、最もスペースを取っているコンテンツを確認できます。

   このセクションでは、長期間使用されていないコンテンツを特定できます。上部のバーで **[古いワークブック]** や **[古いデータソース]** をクリックします。未使用のコンテンツの中で最も古いセットを選択する (右端にマークする) と、詳細が表示されます。詳細は左下のセクションに表示されます。これは、サイズに関係なく、誰も使用していないコンテンツとして、アーカイブや削除を検討する次のコンテンツ セットになります。



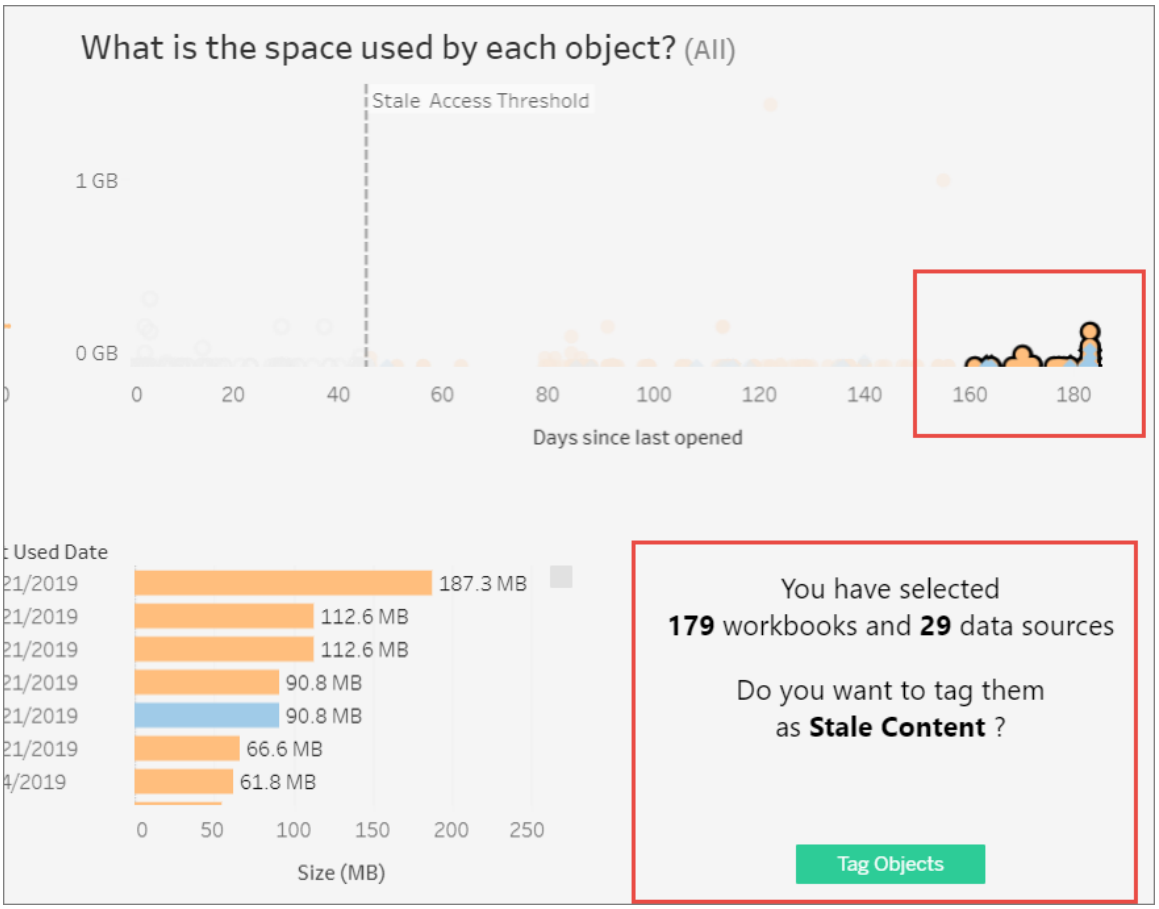
3. **下部のセクションには、**次に示すように、選択したコンテンツに関する詳細情報が表示されます。

古いコンテンツのアーカイブまたは削除

2020.3 以降では、[古いコンテンツ] 管理ビューに、コンテンツを選択して古いコンテンツとしてタグ付けできる機能が含まれています。コンテンツは、下部のセクションまたは右上のセクションから選択できます。選択すると、次のスクリーンショットに示すように、選択したオブジェクトの数とコンテンツのタイプが表示されます。**[タグ オブジェクト]** ボタンをクリックして、選択したコンテンツにタグを付けます。

> **注:** 一度にタグ付けできるのは 1,000 オブジェクトまでです。

次のスクリーンショットでは、過去 160 日以上開かれていないコンテンツが古いコンテンツとしてタグ付けされるように選択されています。



タグ付けされたコンテンツをすべて検索するには、**[探索]** に移動します。古いコンテンツとしてタグ付けされたすべてのコンテンツを表示するには、**[古いコンテンツ]** フィルターを選択します。コンテンツを選択して、アーカイブに使用するプロジェクトにコンテンツを移動するか、コンテンツを削除することができます。

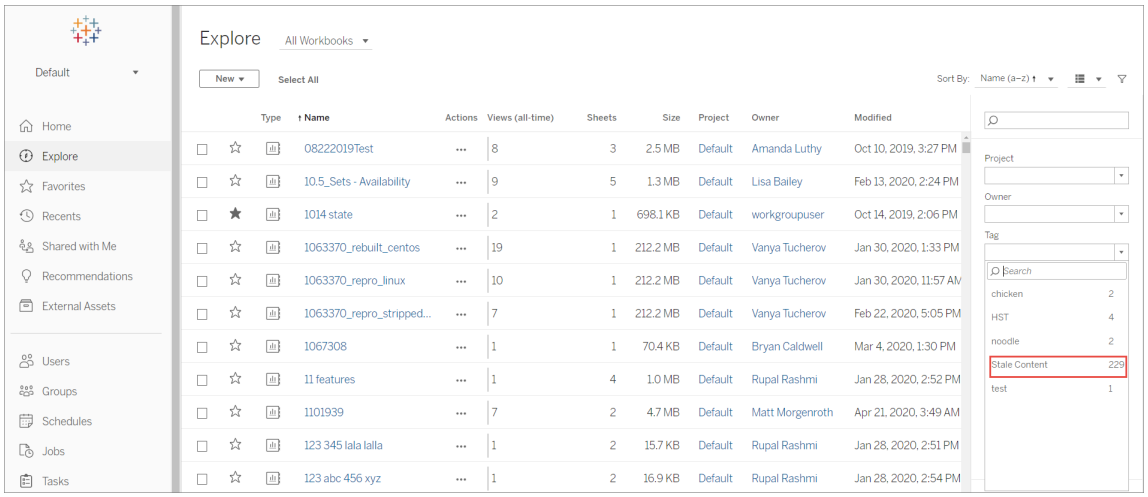


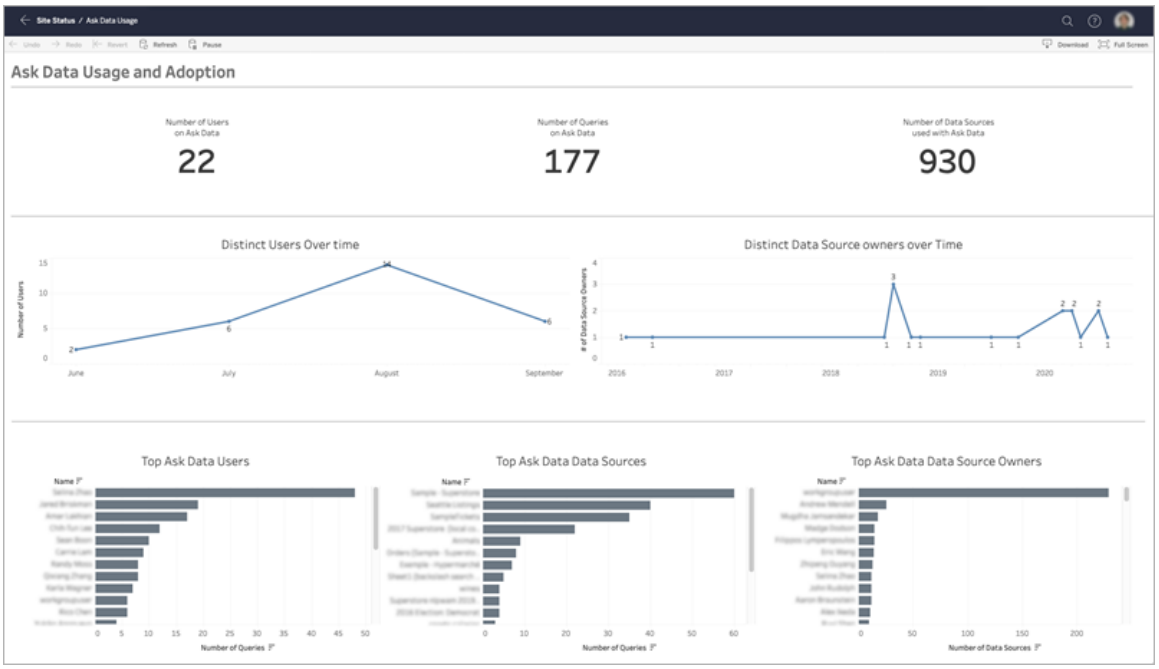
Tableau Server に Advanced Management がある場合は、Tableau Content Migration Tool を使用して、古いコンテンツのアーカイブを定期的なスケジュールで管理できます。たとえば、[古いコンテンツ] とタグ付けされたコンテンツを自動的に選択して [アーカイブ] プロジェクトに移動する、定期的なスケジュールで実行される計画を作成できます。一定の時間が経過すると、このプロジェクト内のコンテンツをシステムから削除できます。詳細については、「移行計画: ワークブック」を参照してください。

[「データに聞く」(Ask Data) の使用状況] ビュー

「データに聞く」とメトリクスへの重要な変更点

Tableau の「データに聞く」機能とメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。自然言語技術の進歩に伴い、Tableau は、データに関する質問や変更点の把握をより簡単に行えるように、インターフェイスの改善に取り組んでいます。詳細については、「*Tableau AI と Tableau Pulse によるデータエクスペリエンスの再構築*」を参照してください。

[「データに聞く」(Ask Data) の使用状況] ビューは事前に構築されたダッシュボードであり、サイト管理者またはサーバー管理者はこのビューで、サイトでの「データに聞く」(Ask Data) の使用パターンと価値を表示および把握できます。また、管理者は、「データに聞く」(Ask Data) に対するエンゲージメントの増加を確認したり、社内トレーニングやロールアウトの結果を監視したりできます。このダッシュボードでは、見出しの値のいくつかのメトリクスに加えて、「データに聞く」(Ask Data) を頻繁に使用しているユーザー、データソース、およびデータソース所有者がハイライトされます。



「データに聞く」(Ask Data) を有効にするには、サイトでの「データに聞く」(Ask Data) 機能の無効化または有効化を参照してください。

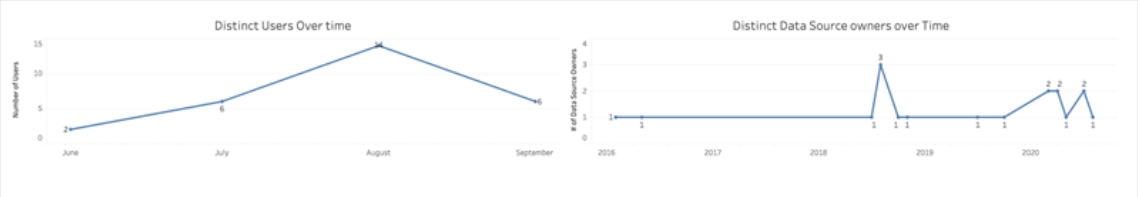
ダッシュボードの詳細

[「データに聞く」(Ask Data) の使用状況] ビューには、サイト全体での「データに聞く」(Ask Data) 機能についての情報が表示されます。次のメトリクスを使用して、ユーザー エンゲージメントを把握し、組織へのセルフサービス分析の導入を促進することができます。

ダッシュボードの上部にある 3 つの見出しメトリクスには、サイトでの「データに聞く」(Ask Data) の使用状況の概要が示されます。

- **「データに聞く」(Ask Data) のユーザー数 -** サイトで「データに聞く」(Ask Data) を使用しているユーザーの総数が表示されます。

- **「データに聞く」(Ask Data) のクエリ数 -** サイトで発行された「データに聞く」(Ask Data) のクエリの総数が表示されます。
- **「データに聞く」(Ask Data) で使用されたデータ ソースの数 -** 「データに聞く」(Ask Data) で使用されたデータ ソースの総数が表示されます。



ダッシュボードの中央では、「データに聞く」(Ask Data) の経時的な使用状況が 2 つの折れ線グラフで示されます。

- **一定期間にわたる個別ユーザー -** 「データに聞く」(Ask Data) を一定期間にわたって使用している個別ユーザーの数が表示されます。
- **一定期間にわたる個別データ ソース所有者 -** 一定期間にわたる個別データ ソース所有者の数が表示されます。

ダッシュボードの下部にある 3 つの棒グラフには、「データに聞く」(Ask Data) を頻繁に使用しているユーザー、データ ソース、およびデータ ソース所有者が示されます。

- **「データに聞く」(Ask Data) を頻繁に使用しているユーザー -** 「データに聞く」(Ask Data) を頻繁に使用しているユーザーと、各ユーザーが発行したクエリの総数が示されます。
- **「データに聞く」(Ask Data) で頻繁に使用されているデータ ソース -** 「データに聞く」(Ask Data) で頻繁に使用されているデータ ソースと、各データ ソースに対して発行されたクエリ

の総数が示されます。

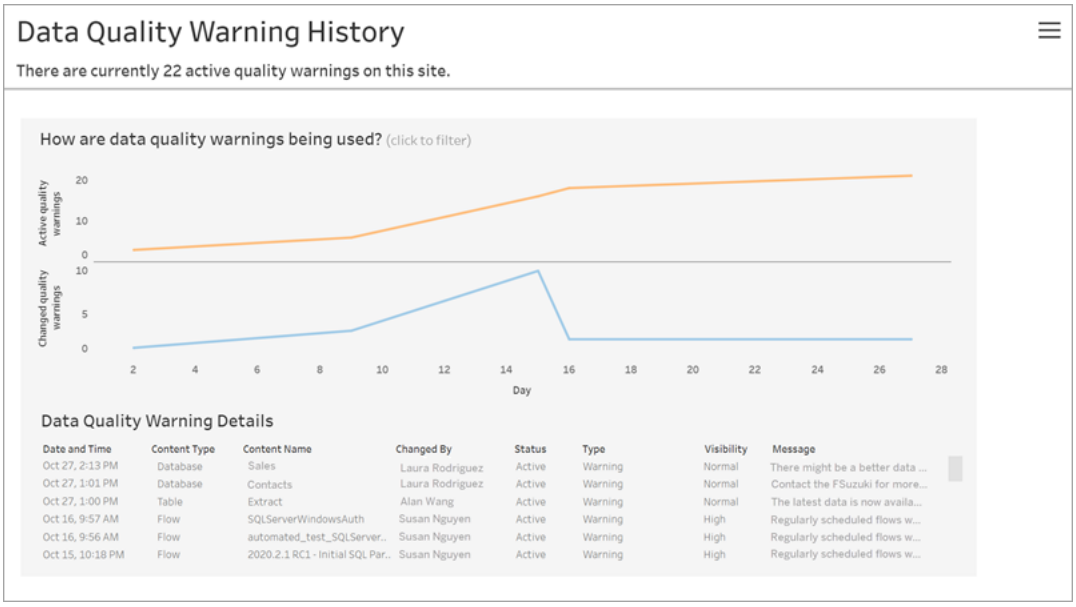
- **「データに聞く」(Ask Data) を頻繁に使用しているデータソース所有者 -** 「データに聞く」(Ask Data) を頻繁に使用しているデータソース所有者と、各ユーザーが所有しているデータソースの総数が示されます。

データ品質に関する警告履歴

環境内で Tableau Catalog が有効になっている場合、サイト管理者は、データ品質に関する警告履歴 (事前に作成された管理ビュー) を使用して、サイト上でデータ品質に関する警告がどのように使用されているかを確認できます。

Tableau Catalog (データ管理の一部) の詳細については、Tableau Server または Tableau Cloud ヘルプの「Tableau Catalog について」を参照してください。

[サイトステータス] ページで、[データ品質に関する警告履歴] ダッシュボードを選択します。



ダッシュボードには、一定期間アクティブになっていたデータ品質に関する警告の数が表示されます。また、同じ期間に変更された (作成、更新、削除された) 警告の数も表示されます。

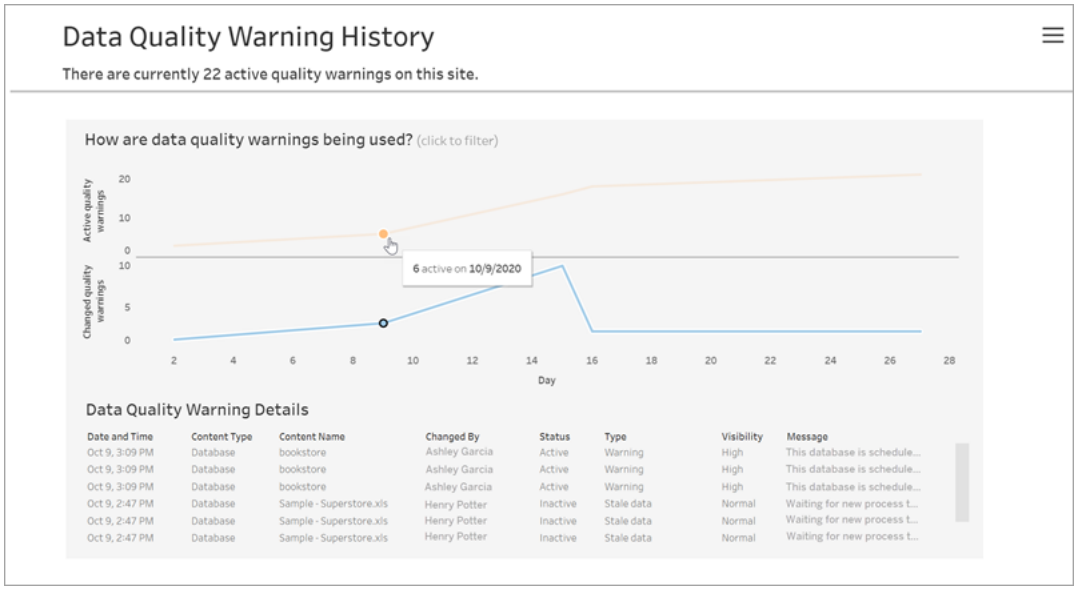
警告の詳細の表示

折れ線グラフの下には、次のようなデータ品質に関する警告の詳細が表示されます。

- **日時** - 警告が作成された日時、または最後に変更された日時。
- **コンテンツ タイプ**- データベース、テーブル、データ ソースなど、警告が設定されているアセットのタイプ。
- **コンテンツ名** - 警告が設定されているアセットの名前。
- **変更者** - 警告を作成または最後に変更したユーザーの名前。
- **ステータス** - 警告がアクティブか非アクティブかを示します。
- **タイプ** - 警告タイプは、古いデータ、警告、非推奨、機密データ、またはメンテナンス中です。
- **可視性** - 警告は、通常 (既定) と高い可視性のどちらかを示すように設定できます。
- **メッセージ** - ユーザーが警告の詳細を表示したときに、警告作成者がユーザーに表示するように書き込んだメッセージ。

警告履歴のフィルター処理

データ品質に関する警告履歴を確認するときに、ビュー上のマークをクリックして、ビューの下に表示される詳細をフィルター処理できます。



日付軸の数値は、時間範囲内の日付を表します。たとえば、今日が 11 月 18 日で、過去 7 日間のフィルター処理を行うと、日付軸には 12 ~ 18 と表示されます。

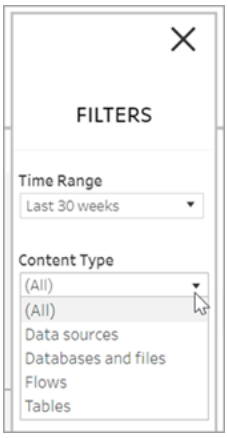
右上隅のフィルター アイコンをクリックすると、その他のフィルターを使用できます (時間範囲でフィルター処理とコンテンツ タイプでフィルター処理)。

時間範囲でフィルター処理

時間範囲は、年から分まで設定できます。

コンテンツタイプでフィルター処理

サイト上のすべてのデータ品質に関する警告を表示したり、フィルターを適用して、データソースやテーブルなどの特定の種類のアセットに対する警告を表示したりできます。



データ品質に関する警告履歴データへのアクセス

[データ品質に関する警告履歴] 管理ビューの他にも、Tableau Server リポジトリの "workgroup" PostgreSQL データベースでデータ品質に関する警告履歴データにアクセスすることができます。このデータにアクセスするには、Tableau Server リポジトリへのアクセスを有効にする必要があります。

データ品質に関する警告履歴データを含むこれらのテーブルを検索できる最新のデータディクショナリを開くリンクについては、「Tableau Server データディクショナリについて」を参照してください。

- historical_events
- historical_event_types
- hist_data_quality_indicators
- data_quality_indicators

実行可能なユーザー

データ品質に関する警告を設定するには、サーバー管理者またはサイト管理者である必要があります。

カスタム管理ビューの作成

Tableau Server の [メンテナンス] ページで使用できる組み込みの管理ビューに加えて、Tableau Desktop を使用すると、サーバーアクティビティに関する独自の分析をクエリおよびビルドできます。そのために、Tableau Server リポジトリに接続してリポジトリ内のビューに対してクエリを実行できます。その場合は、"tableau" または "readonly" ユーザーのどちらかを使用します。

Tableau Server リポジトリに接続するには、Tableau Server リポジトリでデータを収集するを参照してください。

- **tableau** ユーザー—tableau ユーザーは、リポジトリデータベース内の特殊なビューやテーブルのサブセットにアクセスできます。これらのビューやテーブルは、管理者がカスタム管理ビューを作成できるように用意されたものです。Tableau では、これらのテーブルやビューを使用して作成されたカスタムビューを破壊しないように、そうしたテーブルやビューに対する変更を制限するように努めています。

- **readonly** ユーザー: readonly ユーザーは、多数のリポジトリテーブルにアクセスでき、サーバー使用状況に関するデータをより多く提供できます。管理者はこれらのテーブルを使用してカスタム管理ビューを作成することもできますが、こうしたテーブルの多くは、基本的に Tableau Server の機能をサポートするためのものであり、警告なしに変更または削除できます。つまり、これらのテーブルから作成されたビューは、データベース構造が変化すると破壊される可能性があります。

カスタム管理ビューの例については、Tableau コミュニティを参照してください。または、組み込まれている管理ビューを表示するときに生成される一時的なワークブックを使用することもできます。

ビルトインユーザーのいずれかを使用して接続する前に、Tableau Server データベースへのアクセスを有効にする必要があります。その後、Tableau Desktop を使用して、データベースへの接続およびクエリ実行を tableau ユーザーまたは readonly ユーザーとして行うことができます。

`tsm configuration set` オプションの auditing.enabled は、Tableau Server がリポジトリ内のユーザーアクティビティ履歴やその他の情報を収集するかどうかを制御します。既定では有効になっています。履歴イベントの収集は Tableau Server のバックアップファイル (.tsbak) のサイズに影響を与えることに注意してください。

- `hist_` tables はすべて、`tsm configuration set` オプションの wgserver.audit_history_expiration_days により制御されます。このオプションは、何日分のイベント履歴をリポジトリに保管するかを制御し、既定値は 183 日です。
- `_http_requests` テーブルは、tsm maintenance cleanup の`--http-requests-table`オプションの実行時に、7 日よりも古いデータをすべて消去します。詳細については、不必要なファイルの削除を参照してください
- `_background_tasks` テーブルの内容が自動的に消去され、過去 30 日間のデータが保存されます。
- "_" 接頭辞で始まる名前を持つその他のテーブルは、現在のデータを含みます。

Tableau Server リポジトリの表の詳細については、ワークグループデータベースのデータディクショナリを参照してください。

パフォーマンス

Tableau Server のパフォーマンスをモニタリングおよび調整できます。

Tableau Server パフォーマンスの概要

時間をかけてTableau Serverのパフォーマンスを理解すると、Tableau Serverの効率を改善することで、ユーザーへの対応がより容易になります。各サーバー環境は一意であり、パフォーマンスに影響する多数の変数が存在しますが、Tableau Serverでのパフォーマンスデータを理解し、それに基づいて行動するための一般的な手順は同じです。

- **通知。**重要なサーバーイベントにメール通知を設定します。たとえば、サーバープロセスを利用できないときや、サーバーのディスク空き容量が少なくなっているときに通知を受信できます。
- **モニタリング**。サーバーのパフォーマンスがどの程度うまくいっているかを理解するため、Tableau Serverに関するデータを収集して分析します。
- **チューニング**。タスクやプロセス構成などを調節して、Tableau Serverのパフォーマンスを改善します。
- **トラブルシューティング**。リソース、ワークブックなどにおける弱点を特定し、Tableau Serverのパフォーマンスを改善します。

一般的なパフォーマンスのガイドライン

ハードウェアとソフトウェア

より多くのコアとメモリの追加:Tableau Serverを1台または複数のコンピューター上で実行しているかに関係なく、一般的には、CPUコアとメモリの量が多いほど、パフォーマンスも上がります。Tableau Serverが推奨するハードウェア要件とソフトウェア要件を満たすことを確認してください。

仮想環境でTableau Serverを実行している場合は、VMホスト上の物理CPUコア数と関連して、vCPU割り当てに対してVMホストのベストプラクティスを使用します。

外部リポジトリ

Tableau Serverのパフォーマンスを最適化するには、展開内の専用ノードでリポジトリを分離することをお勧めします。Advanced Managementライセンスをお持ちの場合は、リポジトリを外部データベースとして実行することを検討してください。

組織のピーク負荷が 1 時間あたり1000 VizQL セッションを超える場合は、Tableau Server on Linux を実行することもお勧めします。このシナリオでは、VizQL セッションは、Tableau Server からビジュアライゼーションを表示または生成するユーザー アクションのことを指します。

詳細については、「Tableau Server 外部リポジトリ」を参照してください。

構成

オフピーク時間の更新のスケジュール:バックアップ タスクは、バックアップが完了するまで、他のバックグラウンド タスクを停止させる傾向があります。抽出でのバックグラウンド タスク管理ビューを使用して、更新とバックアップのタスク スケジュールを確認します。更新タスクは、バックアップ ウィンドウとオーバーラップしないオフピーク時にスケジュールする必要があります。

キャッシュの確認:キャッシュは、Tableau Server がクライアントの要求にすばやく応答する場合に役立ちます。特に、ライブ データベースに接続するビューの場合に便利です。キャッシュ頻度が `low` (これが既定です) に設定されていることを確認するには、tsm data-access caching list コマンドを使用します。

Tableau Server では、クエリ キャッシュを使用してクエリ結果を保存します。クエリ キャッシュのサイズは、まだ手動で設定していない場合に限り、使用可能なシステム メモリの量に基づいて自動的に設定されるようになりました。クエリ キャッシュは、論理クエリのキャッシュ、メタデータのキャッシュ、およびネイティブ クエリのキャッシュで構成されます。既定の設定はほとんどの状況に適合しますが、TSM コマンド ライン インターフェイスを使用して手動で設定することもできます。TSM 設定は、`native_api.InitializeQueryCacheSizeBasedOnWeights`、`native_api.QueryCacheMaxAllowedMB`、`native_api.LogicalQueryCacheMaxAllowedWeight`、`native_api.MetadataQueryCachMaxAllowedWeight`、`native_api.NativeQueryCacheMaxAllowedWeight`、および `native_api.QueryCacheEntryMaxAllowedInPercent` です。詳細については、native_api.InitializeQueryCacheSizeBasedOnWeights を参照してください。

2 つのセッションのメモリ設定の変更について検討します。

- **VizQL セッションのタイムアウト制限**:既定の VizQL セッションのタイムアウト制限は 30 分です。VizQL セッションはアイドル状態でも、メモリとCPU サイクルを消費し続けます。下限値で済む場合は、tsm configuration set のオプションを使用して `vizqlserver.session.expiry.timeout` 設定を変更します。

- **VizQL クリア セッション**:既定では、ユーザーがビューから離れて移動する場合でも、VizQL セッションはメモリに保存されています。これにより、ビューを再構築する必要性は少なくなりますが、セッション メモリの消費は増加します。メモリを解放するには、`vizqlserver.clear_session_on_unload` 設定の値を `true` に変更して、ユーザーがビューから離れる際にセッションを終了できます。(この設定にかかわらず、Tableau Mobile アプリ用のセッションは、モバイルのパフォーマンスを向上させるために常にメモリに保存されます。)

プロセス構成の評価:Tableau Server は、サーバープロセスと呼ばれる6 つの異なるコンポーネントに分けられます。既定の構成は幅広いシナリオに対応できるように設計されていますが、異なるパフォーマンス目標を達成するように再構成することもできます。特に、プロセスが実行されるコンピューターと実行台数を制御できます。1、2、および 3 ノードの展開に関する一般的なガイドラインについては、「パフォーマンスの調整」を参照してください。

サーバー リソース マネージャー (SRM)

サーバー リソース マネージャー (SRM) は、各 Tableau プロセスが使用しているシステム リソースを監視するとともに、システム上の Tableau Server の合計使用率を追跡します。特定のプロセスまたは製品が全体としてシステム リソースを過剰に消費している場合、SRM はリソースを解放するようにプロセスに通知するか、またはそれらのプロセスを再起動することができます。

SRM がプロセスに通知する、またはプロセスを再起動するタイミングを決定するしきい値は、SRM 構成オプションで設定されます。Tableau 開発チームは内部テストに基づいて既定の値を設定しているため、これらの値を直接変更することはお勧めしません。

過剰なシステム リソースの使用率が検出された場合は、発生している問題を解決するためにこれらの構成オプションを変更する必要があるかどうか判断するために、Tableau サポートに問い合わせることをお勧めします。

パフォーマンス モニタリングの概要

サーバーを監視するときは、サーバーのパフォーマンスが低下しているかどうかや、問題が発生しているかどうかを示すデータを収集/分析します。たとえば、サーバーが処理容量の 100% を長期間使用していることが判明した場合、問題があることがわかります。

収集および分析する必要があるデータは、次の広範なカテゴリーに分類できます。

- リソースの利用状況データ- Tableau Server がディスク容量、メモリ、およびプロセッサーなどのハードウェア リソースをどのように使用活用しているか。
- セッションと読み込み時間データ- ユーザーによる Tableau Server の操作方法 (ビューの読み込みにかかる時間や、同時ユーザー数などを含む)
- バックグラウンド タスクデータ- ユーザー アクションと直接結びつけられていないタスクを Tableau Server が実行する方法。たとえば、バックグラウンド タスクには抽出の更新タスク、サブスクリプション タスクなどが含まれます。

このデータの一部は、読み込み時間データや抽出の更新データを含め、Tableau Server に内蔵されている管理ビューから既にアクセス可能です。ただし、リソース利用状況データを収集するには、外部パフォーマンス モニタリング ツールを使用する必要があります。追加の読み込み時間とバックグラウンド タスクデータを収集するには、Tableau Server リポジトリに接続できます。

ビルトイン管理ビューの詳細については、管理ビュー を参照してください。

> 注:同じワークブックを使用し、ビューを Tableau Server にパブリッシュするには、Tableau Desktop が必要です。

Tableau Server リポジトリでデータを収集する

Tableau Server リポジトリは、すべてのユーザー インタラクション、抽出の更新などに関するデータを格納する PostgreSQL データベースです。リポジトリへのアクセスを有効にして、Tableau Server のパフォーマンスの分析および理解にリポジトリ内のデータを役立てることができます。

重要: リポジトリを直接変更しないでください。Tableau Server が破損するおそれがあります。お客様がリポジトリを変更した場合、そのインストールは Tableau でサポートされなくなります。

Tableau Server on Windows については、「Tableau Server リポジトリでデータを収集する」を参照してください。

Tableau Server リポジトリへのアクセスを有効にした後、そのリポジトリのデータを使用してビューを作成できます。このデータを使用して作成したビューは、カスタム管理ビューと呼ばれる場合があります。カスタム管理ビューはパフォーマンス監視以外にも、ユーザー アクティビティ、ワークブック アクティビティなどの追跡に使用できます。これらのビューで使用できるデータ型についての詳細は、「カスタム管理ビューの作成」と「Tableau Server データ ディクショナリについて」を参照してください。または、パフォーマンス データのみに関心がある場合は、サンプル パフォーマンス ワークブックで事前に選択したデータベース テーブルを使用できます。

Tableau Server リポジトリへのアクセスを有効にする

Tableau Desktop を使用して Tableau Server リポジトリへの接続やクエリ実行を行うには、2 つのビルトイン ユーザーを使用します。`tableau` という名前のユーザーは、Tableau Server アクティビティに関する独自の分析の一環として使用できるいくつかのデータベース ビューにアクセスできます。`readonly` という名前のユーザーは、さらに詳細な分析用のビューの作成に使用可能な追加のデータベース表にアクセスできます。このユーザーを使用することをお勧めします。

リポジトリに接続するには、事前に `readonly` ユーザーのデータベースへのアクセスを有効にする必要があります。`tsm data-access repository-access enable` コマンドを使用して、リポジトリへのアクセスを有効にします。リポジトリへのアクセスを有効にする際に、`readonly` ユーザーのパスワードも作成します。このパスワードを使用してリポジトリにアクセスします。場合によっては、データベースに接続できるよう、リポジトリ ノードでポート 8060 を開いておく必要があります。

1. リポジトリがインストールされているコンピューターでポート8060 が開かされていることを確認します。これは、リモートで接続する場合の要件です。

2. 以下のコマンドでリポジトリへのアクセスを有効にし、readonly ユーザーのパスワードを作成します。

   ```
   tsm data-access repository-access enable --repository-username
   readonly --repository-password <PASSWORD>
   ```

   パスワードに特殊文字が含まれている場合、文字をエスケープするかパスワードを引用符で囲む必要が生じることがあります。bash シェルで特殊文字を渡す方法については、実行しているLinux distro のドキュメントを参照してください。

   このコマンドによりTableau Server が再起動されます。

   > **注:** 後で Tableau Server リポジトリへのリモートアクセスを無効にすることを決定した場合は、`tsm data-access repository-access disable` コマンドを使用します。このコマンドは リポジトリへの外部アクセスを無効にします。これによりlocalhost からのアクセスが無効になることはありません。詳細については、tsm data-access repository-access disableを参照してください

Tableau Server リポジトリへの接続

このセクションでは、Tableau Server リポジトリから表のカスタム セットに接続する方法について説明します。接続できるテーブルについての詳細は、「Tableau Server データディクショナリについて」を参照してください。

1. Tableau Desktop で **[データ] > [データに接続]** を選択し、接続先のデータベースとして **[PostgreSQL]** を選択します。

> **注**: PostgreSQL データベース ドライバーのインストールが必要になる場合があります。www.tableau.com/ja-jp/support/drivers からドライバーをダウンロードできます。

2. [PostgreSQL 接続] ダイアログボックスで、Tableau Server の名前または URL を **[サーバー]** ボックスに入力します。分散サーバー インストールの場合は、リポジトリをホストしているノードの名前または IP アドレスを入力します。

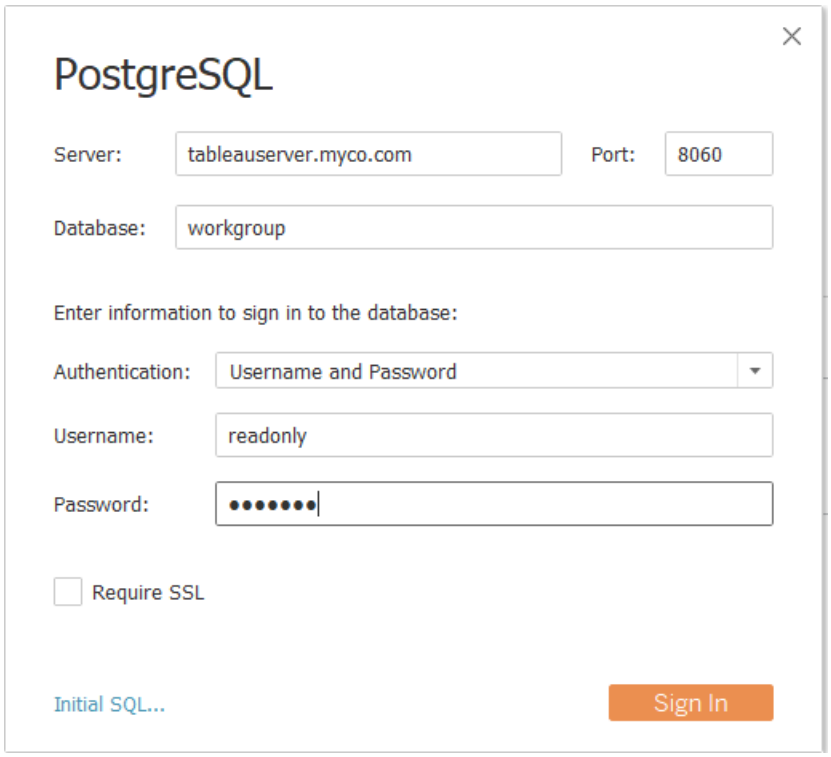
   pgsql.port で設定したポート(既定では 8060) を使用して接続します。

3. 接続先のデータベースとして「`workgroup`」を指定します。

4. 指定したユーザーとパスワードを使用して接続します。

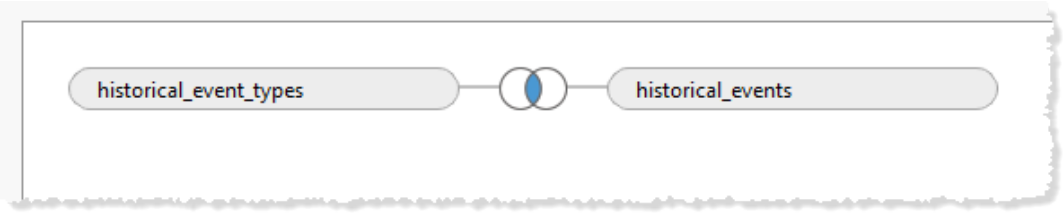
5. Tableau Server でリポジトリへの接続のために SSL を使用するように構成している場合は、**[SSL が必須]** オプションをクリックします。詳細については、クライアントからの直接接続を許可するようPostgres SSL を構成するを参照してください。

6. **[接続]** をクリックします。



7. 接続する1つまたは複数の表を選択します。

   tableau ユーザーは、"_" (アンダーライン) または hist_ で始まるすべての表にアクセスできます。たとえば、"_background_tasks" や "_datasources" などに接続できます。の表にはhist_、特定のユーザーによるアクション ビューに現在表示されていないサーバー ユーザーに関する情報が含まれます。readonly ユーザーは、サーバー使用状況に関する他の情報のクエリに使用できる追加の表にアクセスできます。



8. **[ワークシートに移動]** をクリックします。

PostgreSQL のバージョン

Tableau Server で使用されている PostgreSQL のバージョンを確認するには、次の手順を使用します。

1. sudo アクセス権を所有するユーザーとして Tableau Server にログインします。
2. マシンにインストールされている PostgreSQL のバージョンを表示するには、次のコマンドを使用します。

   $ postgres --version psql --version

上記のコマンドでエラーが発生する場合は、ディレクトリを確認する必要があります。ディレクトリを確認するには、以下の手順を使用します。

1. $ locate find /opt/tableau –name psql
2. このパスに移動した後、version コマンドを発行して PostgreSQL のバージョンを確認します。

   $ postgres psql --version

ワークグループ データベースに接続し、次のクエリを発行してバージョンを取得することもできます。

```
select version()
```

Tableau Server データディクショナリについて

Tableau Server データディクショナリには、Tableau Server リポジトリの "workgroup" PostgreSQL データベースに含まれる表およびビューに関する情報が含まれています。このデータベースは、Tableau Server に永続的なストレージを提供し、Tableau Server アプリケーションのサポートを主な目的としています。このデータディクショナリは、Tableau Server での使用方法に関する情報のクエリをこのデータベースに対して実行するユーザーを対象としたもので、データベースに含まれる表とフィールドをすべて網羅したものではありません。このデータベースとその内容は Tableau Server をサポートすることを目的としているため、構造や内容が警告なしに変更される場合もあります。つまり、データベースにクエリを直接実行して作成したカスタム ビューは使用できなくなる可能性があります。

重要: データディクショナリでは、**現状**レベルのサポートが提供されます。データベースへのクエリの記述や壊れたワークブックの更新など、ディクショナリに基づくレポートの作成については、Tableau コミュニティー フォーラムに参加してください。

データディクショナリを開きます (新しいウィンドウ)。

パフォーマンスの調整

このセクションは収集したパフォーマンス データを使用して、Tableau Server のパフォーマンス改善方法を識別する方法について説明します。同じサーバーの環境は二つとないため、サーバー パフォーマンスを調整する厳密なルールを提供することはできません。ただし、収集したデータのパターンからパフォーマンスに関する結論を導き出すことができます。

たとえば、繰り返し急増が発生しているか。モニタリング ツールの同様のパターンに対応したパターンが管理ビュー内にあるか。このようなパターンの観察は、テストや増分調節を導きます。

Tableau Server のほとんどのパフォーマンス調整は、煮詰めると2 つの一般的なアプローチになります。

- **ユーザー トラフィックの最適化:** ユーザーの要求に応答し、ビューを素早く表示するようサーバーを調節します。
- **抽出の最適化:** パブリッシュされたデータ ソースの抽出を更新するようサーバーを調節します。組織が大量のデータを保有し、データをできるだけ最新に保つ必要がある場合、抽出の更新の最適化が必要となる場合があります。
- **抽出クエリの負荷が大きい環境用に最適化:** これは、抽出をデータ ソースとして使用するワークブックのクエリ パフォーマンスを最適化することに特化したサーバー構成です。

ビューのレンダリングや抽出の更新では、サーバーに最も大きな負荷がかかります。そのため、組織が最も関心を持つタスクに合わせて最適化することをお勧めします。

ベストプラクティスとして、ワークブックのパフォーマンスを最適化します。ワークブックの最適化方法の詳細とリソースについては、「ワークブックのパフォーマンスを最適化」を参照してください。

ユーザー トラフィックの最適化

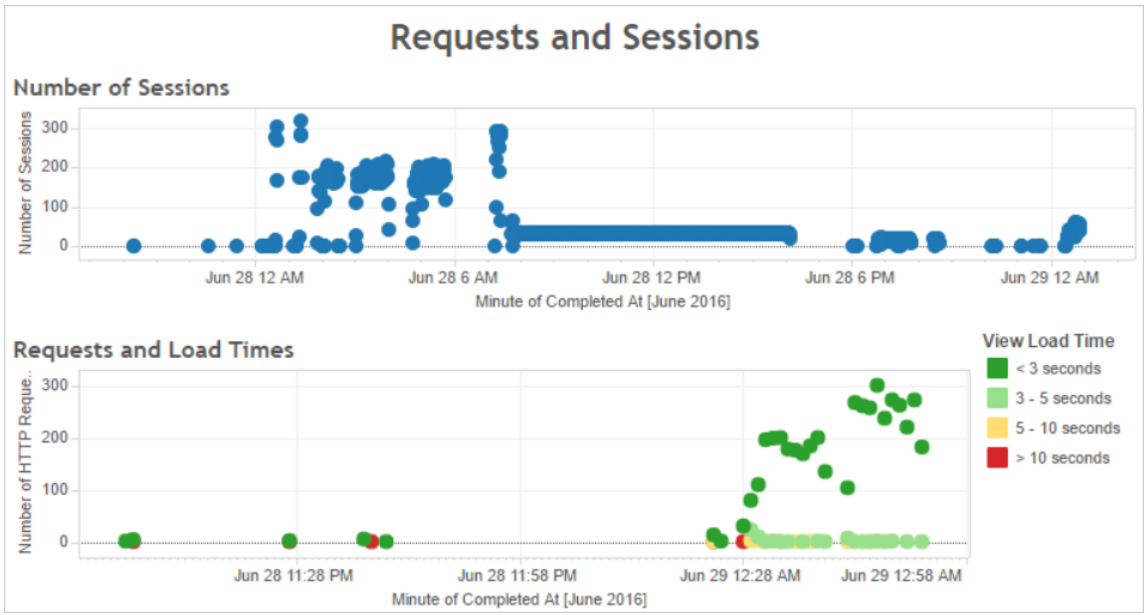
抽出の更新を必要とする多くのアクティブな Tableau Server ユーザーがあり、パブリッシュされたデータ ソースがあまりない場合には、トラフィックを最適化する必要があります。

- ユーザー トラフィックを最適化するタイミング
- ユーザー トラフィック用の最適化方法

ユーザー トラフィックを最適化するタイミング

ビューの読み込み時間が遅い

サンプル パフォーマンス ワークブックの **[リクエストとセッション]** ダッシュボードを使用して、ビューの読み込みにかかる時間を分析します。



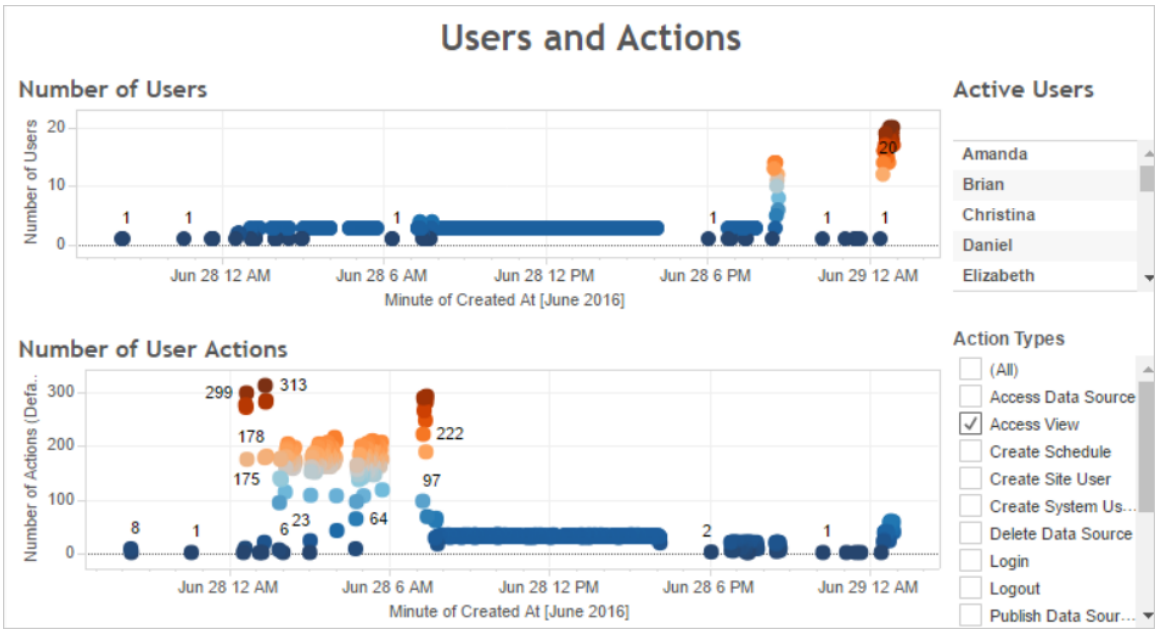
読み込みに 10 秒以上かかるビューが複数あり、読み込み時間の遅さが大量のセッションに対応している場合、ユーザー トラフィックによってサーバーの速度が低下していることを示している可能性があります。

ただし、特定のビューで、閲覧する時刻に関係なく読み込みに時間がかかる場合、ビューのワークブックを最適化する必要がある兆候です。ロード時間の統計 管理ビューでは、最適化する必要

があるワークブックを識別できます。ワークブックを最適化する簡単な方法には、各ビューに表示する情報を減らす、ビューを分割する、フィルターの数を減らす、データ抽出を使用するなどがあります。

高いリソース使用量がユーザー トラフィックに対応している

ピーク トラフィック時刻にサーバーが高い CPU およびメモリ使用量を示している場合は、ユーザー トラフィックを最適化することをお勧めします。ピーク トラフィック時間を判断し、サーバー上の同時ユーザーの数を分析するには、**[ユーザーとアクション]** ダッシュボードを使用します。さらに、ビューへのアクセス量 管理ビューを使用してユーザー トラフィックがどの程度ビューへのアクセスに関与しているかを確認できます (管理機能、パブリッシュやその他のタスクを実行するのではなく)。



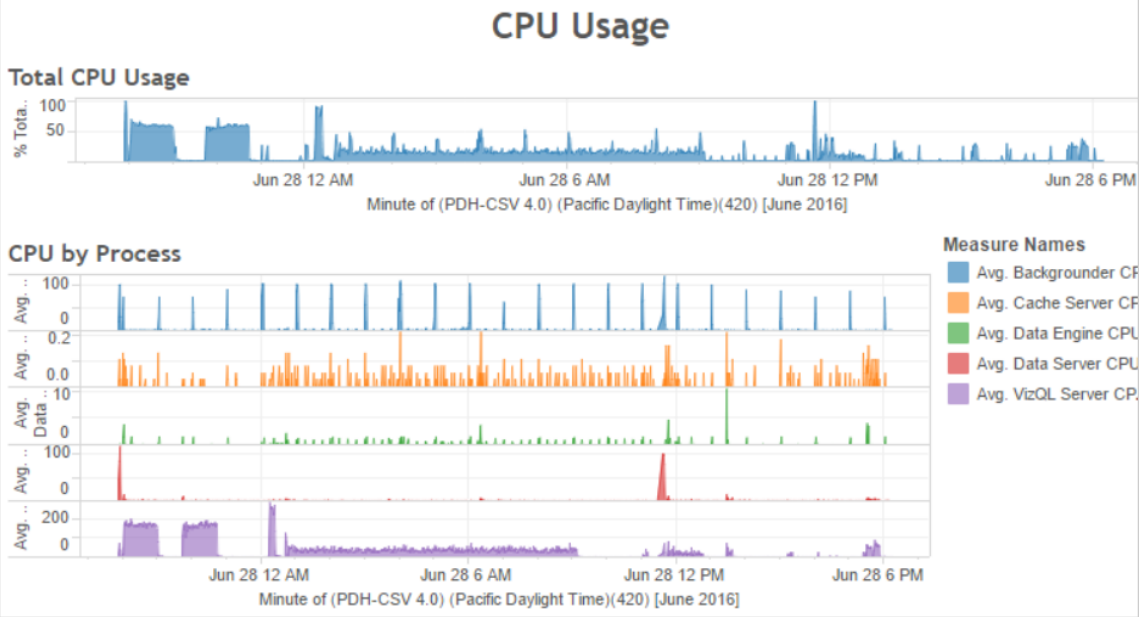
[ユーザー数] ビューのポイントをクリックすると、ダッシュボードには、その時点でアクティブとなっていたユーザー、およびそれらのユーザーが実行したユーザー アクションの数が表示されます。既定では、表示される唯一のユーザー アクションはユーザー ビューですが、**[アクション タイプ]** フィルターを使用して追加ユーザー アクションを表示できます。

同時ユーザーやビューが多数ある場合、時刻を書き留め、これをリソースの使用量と比較できるようにします。経験則として、ユーザー数は大量のユーザー アクションに比例する必要があります。ただし、この例のビューは、負荷生成テストの一貫として 1 人のユーザーに対する人工的に多数のア

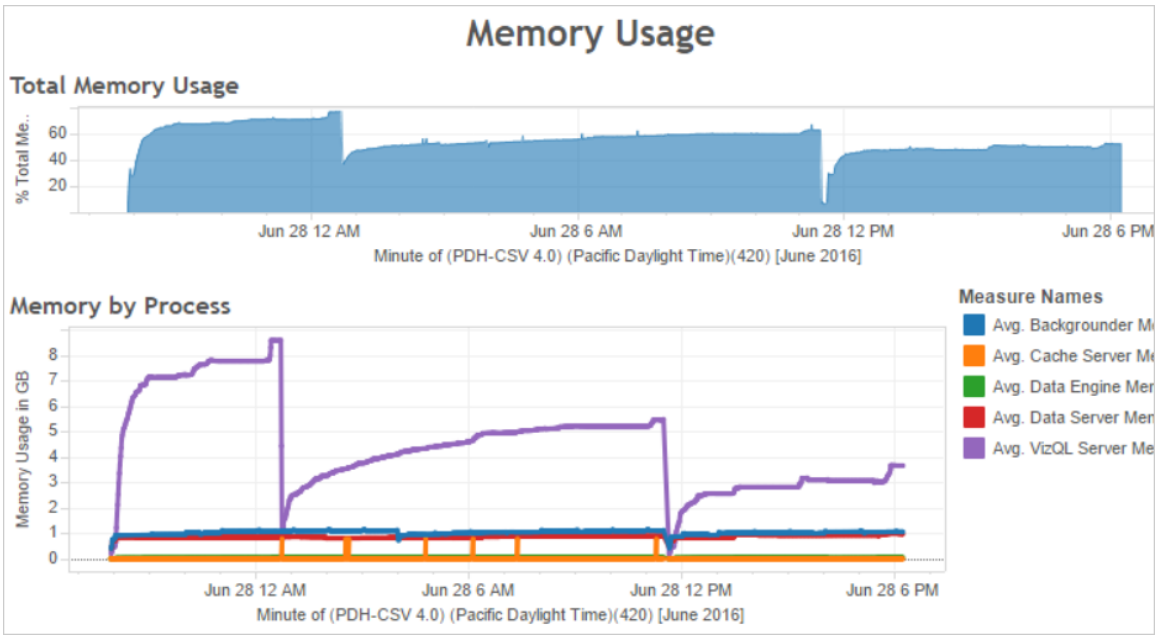
クションを示しています。たとえば、6 月 28 日 12 AM の大量のビューと、後ほど説明するダッシュボードのリソース使用量を比較できます。

[CPU 使用量] ダッシュボードを使用して、合計 CPU 使用量のパーセントや、各プロセスの CPU 使用量のパーセントを表示します。次の例で、6 月 28 日 12 AM における合計 CPU 使用量と VizQL サーバープロセスの急増に注意してください。VizQL サーバープロセスはビューを読み込んでレンダリングするため、大量のユーザー トラフィックによって最初に枯渇を示すのは VizQL サーバープロセスです。

> **注**: 個別プロセスの CPU 使用率の割合は、合計すると100 パーセントを超える場合があります。これは、個別プロセスのプロセッサー利用状況は与えられたプロセッサー コアに対して測定されるためです。対照的に、合計 CPU 使用量はすべてのプロセッサー コアに対して測定されます。



[メモリ使用量] ダッシュボードを使用して、合計 メモリ使用量のパーセントや、平均 メモリ使用量をギガバイト単位で表示します。一般的な規則として、メモリー使用量はユーザー トラフィックと安定して増加します。ここでも、大量のトラフィックにより最初に枯渇を示すのは VizQL サーバープロセスです。



ユーザー トラフィック用の最適化方法

以前の例で示したように、ユーザー トラフィックの高さがリソース使用率の高さに対応している場合、ユーザー トラフィック用に最適化することをお勧めします。

VizQL サーバープロセスの数を調節する

ユーザー トラフィックを最適化する最も効果的な方法は、VizQL サーバープロセスの数を調節することです。最初に VizQL サーバープロセスを 1 回に 1 つずつ追加し、さらにパフォーマンス モニタリングを行って影響を測定します。VizQL サーバープロセスは多くの CPU およびメモリを消費することがあるため、あまりにも多くのプロセスを追加すると、代わりにサーバーの速度が低下する可能性があります。一貫してメモリ使用量が高い場合、VizQL サーバープロセスの数を減らして予約されるメモリの量を減らしてみてください。

プロセスの構成の詳細については、ノードの構成を参照してください。

他のプロセスの数を調節する

ユーザー トラフィックのパフォーマンスを改善する最も効果的な方法は VizQL サーバープロセスの数を調節することですが、VizQL サーバープロセスをサポートする他のプロセスを調節したり、VizQL サーバープロセスがリソースにアクセスするのを防ぐこともできます。たとえば、VizQL サーバー

プロセスはキャッシュサーバープロセスに頻繁に要求を行うため、キャッシュサーバープロセスの数を増やす場合があります。一方、バックグラウンダープロセスは VizQL サーバープロセスとCPU リソースを争う可能性があります。その結果、抽出の更新を頻繁に実行する必要がない場合は、バックグラウンダーに使用するプロセスの数を減らすことができます。バックグラウンダーの追加インスタンスが必要であり、クラスターで Tableau Server を実行している場合、バックグラウンダープロセスを専用ノードに移行できます。

VizQL セッション タイムアウト制限を調節する

以前の例では、VizQL サーバープロセスによって使用されるメモリの量はユーザー トラフィックとともに増加し、トラフィックの終了後しばらくの間、Tableau Server によって引き続き予約されます。これは、VizQL サーバープロセスは指定された時間の各セッション用にメモリを確保するためです。VizQL サーバープロセスが利用可能なメモリの高いパーセンテージを使用している場合、メモリをより素早く利用できるよう、各セッションのタイムアウトを削減してみてください。

これを実行するには、`tsm configuration set` コマンドを使用して `vizqlserver.session.expiry.timeout` 設定を減らします。規定値は 30 分です。

キャッシュの更新頻度を減らす

ユーザーが常に最新のデータを必要としない場合は、Tableau Server でキャッシュを作成し、できるだけデータを再使用するよう構成することで、ユーザー トラフィックを最適化できます。

これを実行するには、`tsm data-access caching list` コマンドを使用して更新頻度を確認します。既定では `Low` です。更新頻度を変更するには、`tsm data-access caching set` コマンドを使用します。

ビューの応答性の評価

ユーザーがビューを開くと、ビューのコンポーネントが最初に取得および解釈され、その後にユーザーの Web ブラウザーに表示されます。ほとんどのビューでは、表示レンダリング フェーズはユーザーの Web ブラウザーで行われ、これによって最速の結果と最高レベルのインタラクティブな応答性が得られます。クライアント Web ブラウザーでほとんどの操作を処理すると、帯域幅が減少し、往復要求の待機時間がなくなります。ビューが非常に複雑である場合、一般的に最適なパフォーマンスをもたらすため、Tableau Server はクライアント Web ブラウザーではなく、サーバー上でレンダリング フェーズを処理します。ビューに必要な応答性がないことが判明した場合は、クライア

ントWeb ブラウザーではなくサーバーでビューのレンダリングを行うしきい値をテストして、変更することができます。詳細については、クライアント側レンダリングの構成を参照してください。

クライアント側レンダリングの構成

Tableau Server のビューに移動する際、ビューの表示に必要な処理 (*レンダリング*) を、クライアントデバイスまたは Tableau Server のいずれかで実行できます。選択内容は、マーク、行、列などの数によって決まるビューの複雑性に依存します。ビューがシンプルであるほど、クライアントデバイスでビューをより速くレンダリングできます。ビューが複雑であるほど、Tableau Server に要求を送信し、サーバーの優れた計算機能を活用した方が速くなります。

注: ビューがポリゴン マーク タイプまたはページ履歴機能を使用している場合、クライアント側レンダリングが有効になっていてもサーバー側レンダリングが常に実行されます。

サポートされるブラウザー

クライアント側レンダリングは、Internet Explorer バージョン 9.0 以降、Firefox、Chrome、Safari でサポートされています。これらの Web ブラウザーにはすべて、クライアント側レンダリングに必要な HTML 5 `<canvas>` 要素が含まれています。

クライアント側レンダリングは、Tableau Mobile アプリでもサポートされています。

コンピューターとモバイル デバイス用の複雑性のしきい値を構成する

PC はモバイル デバイスより処理能力が高いため、Tableau Server はモバイル デバイスよりPC 上でより多くのクライアント側レンダリングを実行します。

サーバー管理者は、複雑性のしきい値を調節することで、PC やモバイル デバイスでクライアント側レンダリングを発生させるタイミングを構成できます。たとえば、ビューの表示が遅いと感じたら、モバイル デバイスのしきい値を下げることができます。または、しきい値を増やして Tableau Server に対する要求の数を減らすことができます。

既定では、コンピュータWeb ブラウザーの複雑性しきい値は 100 です。PC の複雑性しきい値を調節するには、次のコマンドを使用します。

```
tsm configuration set -k vizqlserver.browser.render_threshold -v
[new value]
```

既定では、モバイル デバイスの複雑性しきい値が 60 に設定されています。モバイル デバイスの複雑性しきい値を調節するには、次のコマンドを使用します。

```
tsm configuration set -k vizqlserver.browser.render_threshold_
mobile -v [new value]
```

たとえば、モバイルしきい値を 40 に変更するには、次のコマンドを入力できます。

```
tsm configuration set -k vizqlserver.browser.render_threshold_
mobile -v 40
```

tsm オプション設定の使用方法の詳細については、tsm configuration set のオプションを参照してください。

クライアント側レンダリングを無効にする

クライアント側レンダリングは既定で有効になっており、ビューのパフォーマンスを改善することをお勧めします。ただし、テスト目的や、主に非常に処理能力の低いコンピューターやモバイル デバイスがサーバーにアクセスしている場合など、クライアント側のレンダリングを一時的に無効にする場合があります。

次のコマンドを使用して、クライアント側のレンダリングを無効にします。

```
tsm configuration set -k vizqlserver.browser.render -v false
```

tsm オプション設定の使用方法の詳細については、tsm configuration set のオプションを参照してください。

URL パラメーターを使用したテスト

セッション ベースでサーバー側のレンダリングをテストするには、ビューの URL の最後に「`?:render=false`」と入力します。例:

```
http://localhost/views/Supplies/MyView?:render=false
```

クライアント側レンダリングが Tableau Server で無効になっている場合は、`?:render=true` を入力してセッションに対して有効にします。

```
http://localhost/views/Supplies/MyView?:render=true
```

また、サーバーおよびネットワークの条件に対してサーバー全体のしきい値を調整するのに適切であるかどうかを確認するために、個別のビューで特定の複雑度のしきい値をテストすることもできます。たとえば、低い複雑度 (80 など) または高い複雑度 (120 など) の転換点により、ユーザー操作に対する応答性が向上する場合があります。しきい値をテストするには、サーバーの既定の構成 (クライアント側レンダリングが有効) を維持し、ビューの URL の最後にテストしきい値の数値を入力できます。例は次のとおりです。

```
http://localhost/views/Supplies/MyView?:render=80
```

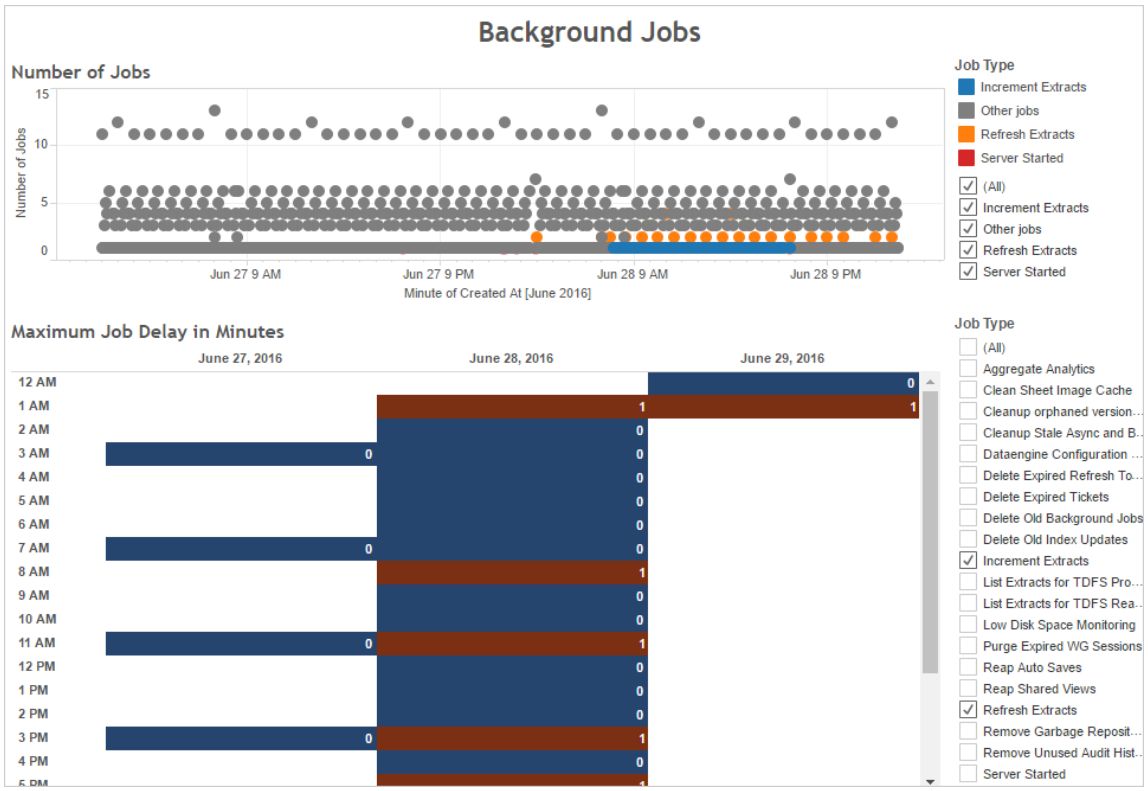
抽出の最適化

抽出スケジュールが高いリソース使用量に対応している場合、または抽出の完了に長い時間がかかる場合は、抽出の最適化を試みます。

抽出を最適化するタイミング

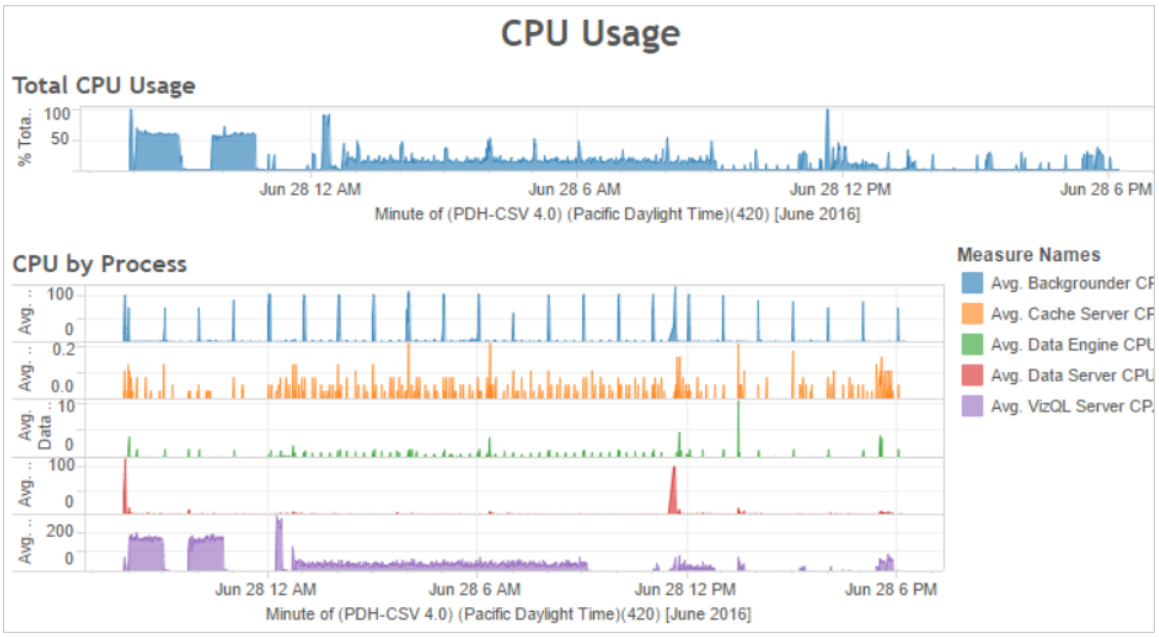
高い CPU 使用量が抽出の更新に対応している

サンプル パフォーマンス ワークブックの **[バックグラウンド ジョブ]** ダッシュボードを使用して、抽出の更新ジョブを含め、Tableau Server によって実行するバックグラウンド ジョブの数を表示します。ダッシュボードには、バックグラウンドの遅延の長さ、つまりバックグランド ジョブがスケジュールされている時刻と実際に実行される時刻の間の時間の長さも表示されます。特定の時刻に長い遅延が見られる場合、多くのジョブが同時に実行中の場合、サーバーの負荷を減らすため、別の時刻にジョブのスケジュールを分散させてみてください。



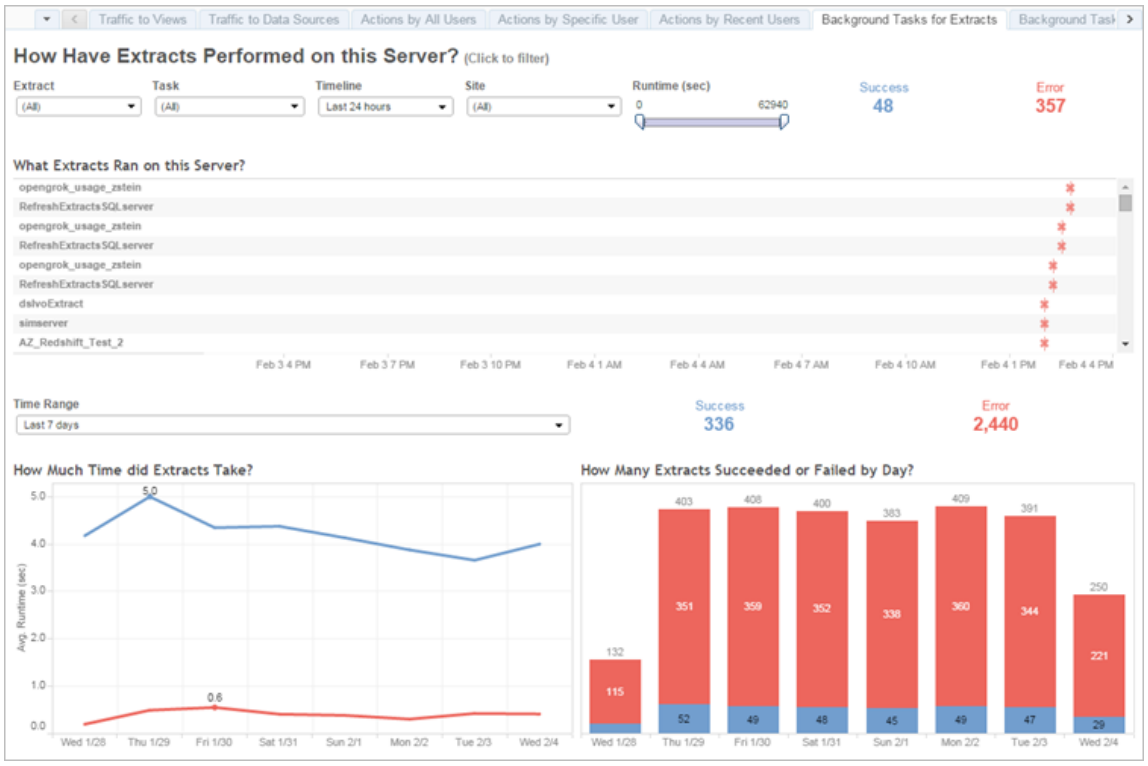
また、多数のバックグラウンドジョブがある時刻や、サーバーの CPU 使用量に長い遅延がある時刻を比較します。**[CPU 使用量]** ダッシュボードを使用して、合計 CPU 使用量のパーセントや、各プロセスの CPU 使用量のパーセントを表示します。バックグラウンダープロセスはバックグラウンドジョブを実行するため、多くの抽出更新ジョブがある場合や、速度が遅い抽出更新ジョブがある場合、最初のプロセスで枯渇が生じます。バックグラウンダープロセスの CPU 使用量が定期的に、短時間 100 % に到達していることに注意してください。これは、反復するスケジュールで集中的な更新ジョブが発生していることを示します。

注: 個別プロセスのプロセッサー利用状況は与えられたプロセッサー コアに対して測定されるため、個別プロセスの CPU 使用率の割合は、合計すると100 パーセントを超える場合があります。対照的に、合計 CPU 使用量はすべてのプロセッサー コアに対して測定されます。



抽出に失敗する、または実行速度が遅い

抽出でのバックグラウンドタスク管理ビューを使用して、抽出に失敗した回数と、抽出の完了にかかる時間を決定します。頻繁に失敗が発生する場合、特定のデータソースに問題があることを示している可能性があります。



抽出用の最適化方法

CPU 使用率の高さが抽出の更新スケジュールに対応している場合、抽出用に最適化することをお勧めします。

抽出更新スケジュールを調整する

サンプル パフォーマンス ワークブックの **[バックグラウンド ジョブ]** ダッシュボードを使用して、抽出を実行する最適な時刻を特定します。オフピーク時間に抽出を実行する以外に、抽出の更新を分散させ、同時に発生するサーバー負荷を最小限に抑えることができます。抽出の更新で引き続き問題が発生する場合は、次の方法で、抽出の更新頻度をできるだけ減らします。

- サーバーがビジーでない時間に抽出をスケジュールします。
- 更新の頻度を減らします。

特定の抽出の高速化

抽出でのバックグラウンドタスク管理ビューを使用して、失敗する抽出の更新および時間がかかる抽出を特定します。

- 抽出のサイズを小さくします。フィルターまたは集計によって抽出のデータセットを短くし、未使用のフィールドを非表示にして狭くすることで、サーバーのパフォーマンスの向上に役立ちます。この変更を行うには、Tableau Desktop オプション **[使用していないフィールドをすべて非表示]** および **[表示可能なディメンションのデータを集計する]** を使用します。詳細については、Tableau ヘルプの「抽出の作成」を参照してください。

  パフォーマンスの高いワークブックを構築するための一般的なヒントについては、Tableau ヘルプで「パフォーマンス」を検索してください。Tableau Server にパブリッシュした後、ワークブックの実行方法を確認するため、パフォーマンスの記録を作成できます。詳細については、パフォーマンスの記録の作成を参照してください。

- 増分更新ジョブを使用します。増分更新では、抽出を最初から作成する代わりに、既存の抽出に新しい行を追加します。このタイプの抽出の更新は、前回の抽出更新ジョブの実行後に追加されたデータのみを処理するため、迅速に実行できます。ただし、データソースに追加ではなく更新されたデータは考慮しません。その結果、増分更新ジョブを実行する場合でも、時折、完全更新ジョブを実行することをお勧めします。たとえば、毎日ではなく週に 1 回または 2 回データソースのジョブの完全更新を実行することができます。

抽出更新の実行モードを構成する

抽出の更新スケジュールを作成するときには、更新が並列実行モードで実行されていることを確認します。並行してスケジュールを実行する際は、スケジュールに 1 つの更新タスクしか含まれていない場合であっても、使用可能なすべてのバックグラウンダープロセスで実行します。"シリアル" でスケジュールを実行する際、1 つのバックグラウンダープロセスでのみ実行されます。既定では、更新タスクをできるだけ速く終わらせるために、実行モードは "パラレル" にセットされています。

ただし、状況によっては、実行モードをシリアルに設定した方が適切な場合があります。たとえば、大規模なスケジュールがすべての使用可能なバックグラウンダープロセスを使用しているために、他のスケジュールの実行できない場合、実行モードを "シリアル" に設定することがあります。

バックグラウンダープロセスの数を増やす

1 つのバックグラウンドプロセスで、特定のタスクに対して 1 つの CPU コアを 100 パーセント消費する場合があります。このため、実行する必要があるインスタンスの合計数は、コンピューターの使用可能なコアによって異なります。Tableau Server をクラスターにインストールしており、別のノードでバックグラウンダープロセスを実行する場合は、経験則として、バックグラウンダープロセスを実行するコンピューターのコアの半数と全数の間にバックグラウンダープロセス数を設定します。

プロセスの構成の詳細については、ノードの構成を参照してください。

プロセスの隔離

クラスターに Tableau Server をインストールしている場合、リソースの競合を防ぐため、バックグラウンダープロセスを別のノードに移動することで大きなメリットが見られます。バックグラウンダープロセスは非常に多くの CPU を消費し、他の CPU を消費するプロセスが実行されているものと同じノードで実行すると、サーバーの速度が低下する可能性があります。たとえば、VizQL サーバープロセスとデータエンジンの両方で CPU を大量に使用する場合があります。詳細については、推奨されるベースライン構成トピックを参照してください。

抽出クエリの負荷が大きい環境用に最適化

このトピックでは、抽出クエリの負荷が大きい環境でパフォーマンスを最適化および改善するために、Tableau Server のトポロジと構成を具体的に設定するためのガイドを提供します。

抽出クエリの負荷が大きい環境とは ワークブック、ビュー、ダッシュボードの読み込み中には、抽出とフェデレーション データ ソースに対してクエリが実行されるため、多くのクエリ ワークロードが作成されます。そのため、抽出とフェデレーション データ ソースが多くある場合は、「抽出クエリの負荷が大きい環境」であると言えます。

上記で定義したように、抽出クエリの負荷が大きい環境である場合、次の 2 つのセクションを参考にして、この構成が適切かどうかを判断することができます。

この構成を使用するタイミング

この構成の背後にある重要な理由: Hyperは、高速データ取り込みと分析処理に適した Tableau のメモリ最適化データエンジン テクノロジーであり、クエリの負荷が大きいワークロードを

最適化することの重要性を裏付けています。抽出の使用の拡大に伴い、**Tableau Server クラスタの専用 ノードでデータ エンジンを構成する**ことをお勧めします。この構成により、Tableau Server はインフラストラクチャをスケールアウトして、抽出をクエリするときのパフォーマンスを最適化できます。

抽出とフェデレーション データ ソースを使用してコンテンツを表示する場合、Tableau Server のパフォーマンスに影響を与えるいくつかの要因があります。ここでの目標は、サーバーでコンテンツを表示するときに、**一貫性のある信頼性の高いクエリ パフォーマンスを実現すること**です。この構成は、以下のいずれかの条件がお使いの環境に該当する場合に使用します。

- ワークブックの読み込み時間には大きなばらつきがあり、ワークブックでは抽出またはフェデレーション データ ソースが使用されている。
- Tableau Server の展開で、Creator、Explorer、Viewer の数、および抽出ベースのコンテンツの数が増加しているため、効率的にスケール アウトする必要がある。
- ファイル ストアがマシンに存在するときに、データ エンジンとVizQL Server 間でリソースの競合がある。
- 大量のデータを分析する。この構成は、データの取り込みと分析の両方で、ビッグ データ シナリオのパフォーマンスを最適化するのに役立ちます。Tableau とビッグ データの詳細については、「Tableau を使用したハイパーチャージ ビッグ データ分析」を参照してください。

> **注:** クエリの実行時間を確認するには、サーバー側のパフォーマンスの記録を使用します。Tableau のリソース利用状況を確認するには、Windows インストールでは **パフォーマンス モニター**、Linux インストールでは *sysstat* または *vmstat* ツールを使用します。

この構成を使用する利点

データ エンジンの専用 ノードを構成する主な利点は次のとおりです。

- 専用のデータ エンジン ノードにより、抽出クエリと、リソースを大量に消費する他のワークロード(VizQL Server によって処理されるワークロードなど) との間のリソースの競合が減少します。

- 抽出クエリは、専用ノードで動的に負荷分散されます。その際、過剰に活用されているノードや、十分に活用されていないノードがないように、システムの現在の状態がチェックされます。

- 抽出に依存するワークブックを読み込む際に、ユーザーエクスペリエンスのさらに一貫したパフォーマンスが得られます。ここで焦点を当てるのは、個々のクエリを改善することではなく、一貫性のある信頼できるパフォーマンスを確立することです。

- より多くのリソースを必要とするTableau Serverプロセスのスケールアウトをより詳細に制御できます。VizQL Server、データエンジン、およびバックグラウンダーがすべて同じノードで実行されており、抽出クエリが遅いことが問題である場合、3つのすべてのプロセスで2つ目のノードを追加しても、パフォーマンスを向上させることは困難です。この構成を使用すると、抽出クエリのワークロードを具体的に改善するノードを追加できます。

- 可用性とアップタイムを改善するのに役立ちます。障害が発生し、専用のデータエンジンノードの1つが使用できない場合、VizQL Serverは、問題のあるノードで保留中の要求を他の専用のデータエンジンノードにルーティングしようとします。

- データエンジンはマシンで利用可能な数のコアを活用します。これにより、専用のデータエンジンノードにリソースをさらに追加して、クエリ応答時間や高価な抽出クエリのばらつきを低減したり、専用のデータエンジンノードを追加して、サーバーでの抽出クエリのスループットを向上させたりするなど、柔軟に対応できます。

- データエンジンは、CPU利用率を1時間あたり平均75%に抑えるようにデフォルトで設定されています。これは、他のTableau Serverプロセスとの競合を回避することを目的としています。データエンジンを専用ノードで実行している場合は、この平均を95%まで増やすことができます。これを行うための詳細については、「hyper.srm_cpu_limit_percentage」を参照してください。

この構成を使用しないタイミング

- 抽出ベースのクエリの読み込みで問題が発生していない場合は、ハードウェアリソースをTableau Serverの他の部分に割り当てる方が適切な場合があります。

- ファイル ストア、データエンジン、および VizQL Server が共存するノードでは、データエンジンとVizQL Server との間でリソースの競合は発生しません。
- この構成を実装する前に、VizQL Server の CPU 使用率、およびファイル ストアと共にインストールされたデータエンジンが存在するノードの CPU 使用率を評価することが強く推奨されています。

設定

この構成の主な目標は、1 つ以上の専用ノードにデータエンジンを配置することです。

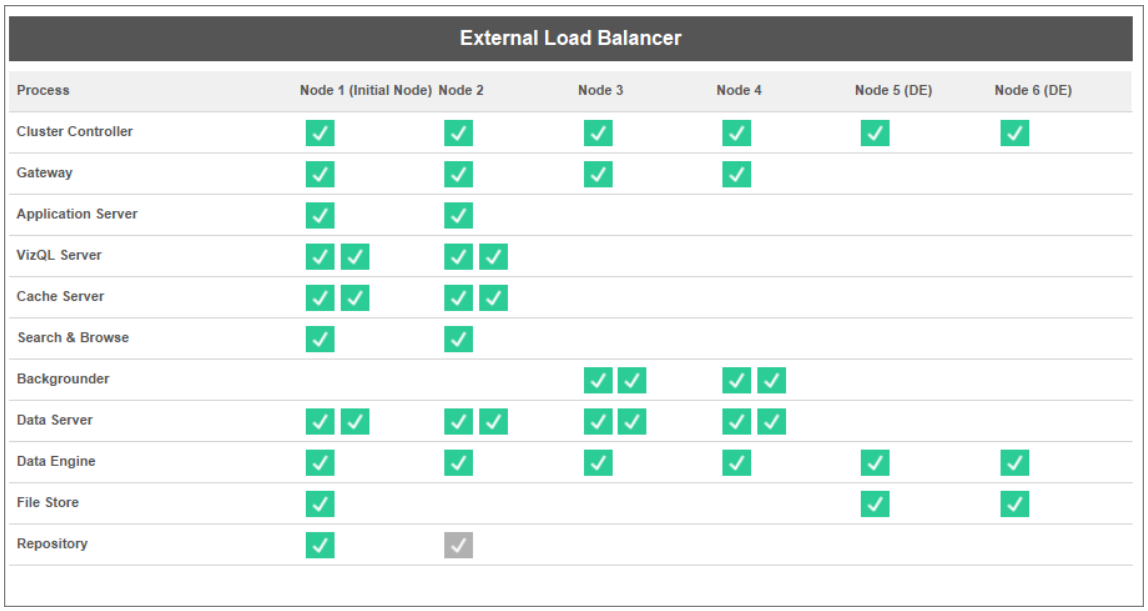
- つまり、ファイル ストアがローカルにインストールされている展開では、1 つ以上の専用ノードでファイル ストアを構成するということです。データエンジンは、ファイル ストアと同じノードに自動的にインストールされます。
- 外部ファイル ストアを構成している展開では、引き続き Tableau Server の専用ノードでデータエンジンを構成できます。

つまり、VizQL Server とファイル ストアのプロセスを分けることで、抽出クエリの負荷と、ビューの読み込みまたは操作にかかる負荷との間の均衡が取れ、管理を向上させることができます。この構成は、抽出をクエリする際に一貫したパフォーマンスを得ることを目標としています。

以下は、データエンジン/ファイル ストア プロセスにノード 5 と 6 の 2 つの専用ノードがある構成を視覚的に表したものです。これは、ファイル ストアがローカルに構成されている例であり、データエンジンとファイル ストアのプロセスが同じ場所に配置されています。

外部ファイル ストアを使用した展開で同じ構成を使用しても機能しますが、その場合、ノード 5 と 6 にはデータエンジンのみが構成されます。

さらに、ノード 1 にはリポジトリプロセスとファイル ストア プロセスがあり、バックアップの実行に必要なすべてのデータがノード 1 に存在するため、バックアップのパフォーマンスを向上させることができます。



ハードウェアのガイダンス

この構成を最大限に活用するには、さまざまなハードウェア サイズと構成を試して、ピーク負荷のパフォーマンス目標に最適なものを確認する必要があります。Hyper は高性能データベース テクノロジーであり、パフォーマンスに影響を与える主要なリソースは、メモリ、コア、およびストレージ I/O です。Hyper がリソースを使用してクエリを処理する方法を理解すると、さまざまな構成の中からハードウェアを選択でき、その理由を理解できるようになります。

- **メモリ:** 抽出ベースのクエリがユーザーまたはバックグラウンド プロセスに対して処理されると、Tableau Server は専用のデータ エンジン ノードを選択し、クエリを処理します。その専用のデータ エンジン ノードは、抽出をローカル ストレージ (多くの場合、サーバーのハードディスク) からメモリにコピーします。使用可能なシステム メモリが増えると、オペレーティング システムは Tableau のメモリ使用率をより適切に管理できるようになります。専用のデータ エンジン ノードは、システム メモリを使用して、実行されたクエリの結果セットを保存します。結果セットがまだ有効で、オペレーティング システムがそれをメモリからクリアしていない場合は、メモリ内の結果セットを再利用できます。

  Tableau Server の推奨される最小ハードウェア要件は 32 GB のメモリですが、抽出ベースのワークブックが大量に読み込まれることが予想される場合は、64 GB または 128 GB を検討する必要があります。コアなどのメモリに加えて、他のリソースの制限値に達した場合は、

128 Gbのメモリにスケール アップする代わりに、64 GB の専用データエンジン ノードにスケール アウトする方がよい場合があります。

抽出をローカル ストレージからメモリにコピーするプロセスには時間がかかるため、ディスク パフォーマンスの最適化が必要になる場合があります。ディスク パフォーマンスの最適化については、**[ストレージ I/O]** セクションで説明しています。

- **コア:** 抽出ベースのクエリを処理する場合、コアの数はパフォーマンスとスケーラビリティに影響を与える可能性のある重要なハードウェア リソースです。CPU コアはクエリの実行を担い、使用可能なコアが多いほど実行時間が短縮されます。一般的に、コアの数を2倍にすると、クエリの実行時間は半分になります。たとえば、現在 4 物理コアまたは 8 vCPU を使用しているクエリの実行時間は 10 秒ですが、8 物理コアまたは 16 vCPU にアップグレードすると、実行時間が5秒になります。

  現在の Tableau Server で推奨される最小ハードウェア要件は 8 コアですが、展開で抽出を使用する場合は、16 または 32 コアのマシンを検討してください。ここで注意すべき重要な点は、メモリとI/O がボトルネックである場合、使用可能なコアを増やしてもクエリのパフォーマンスは向上しません。

- **ストレージ I/O:** Hyper は、抽出ストレージ デバイスの利用可能なパフォーマンスを活用して、クエリ処理を高速化するように設計されています。読み取り/書き込み速度が速いソリッド ステート ドライブ (SSD) などの高速ディスク ストレージを選択することをお勧めします。現在利用可能な最速のディスク ストレージは、NVMe ストレージ プロトコルを使用する SSD です。

> **注:** 専用のデータエンジン ノードのリソースのサイズ設定は、抽出クエリのパフォーマンスにのみ影響します。ワークブックを読み込む際には、VizQL の読み込み要求の合計時間を構成する他の多くのプロセスが伴います。たとえば、VizQL Server プロセスは、データエンジンからデータを取得し、視覚化をレンダリングする役割を担います。

その他のパフォーマンスの調整と最適化:

上記の基本構成の他に、パフォーマンスを最適化するために使用できる追加機能があります。以下で説明する最適化は、ローカル ファイル ストアと外部 ファイル ストアの両方の展開に適用できま

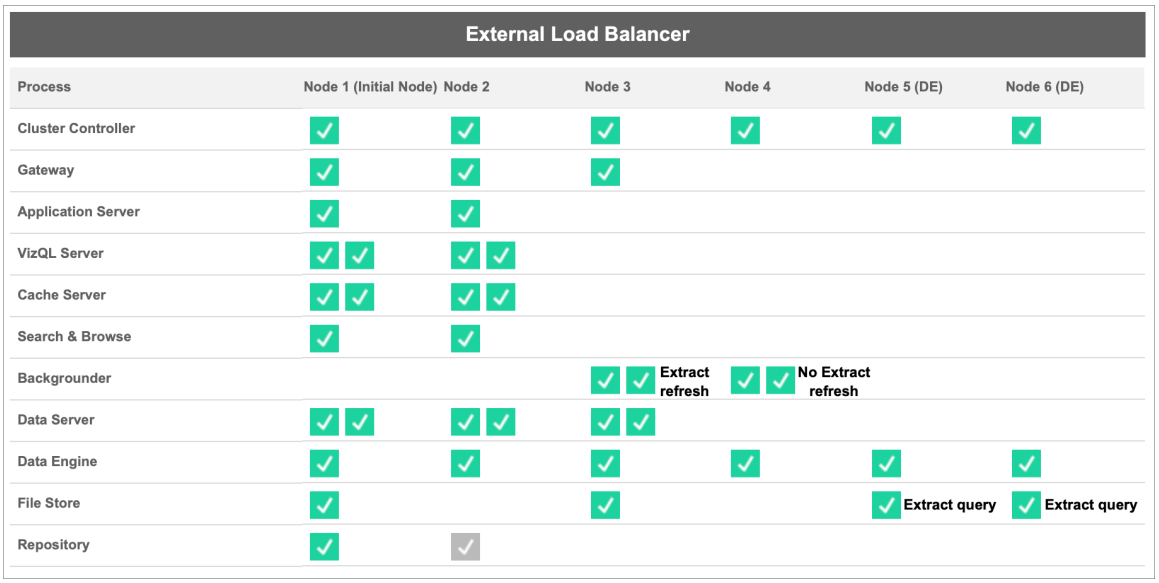
す。

- **抽出クエリのロードバランシング:** データエンジンは、抽出クエリをルーティングする場所を決定するために、サーバーヘルスメトリクスを使用します。これは、データエンジンが消費しているリソースの量と、同じノードで実行されている可能性のある他の Tableau プロセスからの負荷です。システムリソースの評価に加えて、抽出がノードのメモリに既に存在するかどうかも考慮され、クエリを処理するための可用性が最も高いリソースを持つノードに抽出クエリが送信されるようにします。これにより、メモリとディスクの使用効率が向上し、抽出がノード全体のメモリに複製されなくなります。詳細については、抽出クエリのロードバランシングのヘルプ記事を参照してください。

  抽出クエリのロードバランシング機能は、Tableau Server バージョン 2020.2 以降では既定で有効になっています。

- **ノードロールを使用したワークロードの最適化:** バックグラウンダーおよびファイルストアノードロールを使用すると、サーバー管理者は、抽出クエリと抽出更新を実行する専用のノードをより柔軟に制御できます。上記のトポロジ図で説明したように、特定のデータエンジンノードは抽出クエリを処理する専用のノードであるため、ファイルストアプロセスとデータエンジンプロセスのみを実行します。ノードロールは Advanced Management で使用できます。詳細については、ノードロールによるワークロード管理を参照してください。

次の図は、上記の基本構成と同じトポロジを使用していますが、ノードロールが設定されています。



- **抽出更新バックグラウンダーのノード ロール:** ノード3 を抽出更新バックグラウンダーのノード ロールに設定すると、増分更新、完全更新、暗号化/復号化ジョブのみがこのノードで実行されます。ノード4 を抽出更新なしのバックグラウンダーのノード ロールに設定すると、抽出更新以外のすべてのバックグラウンド ジョブがこのノードで実行されます。データ サーバーとゲートウェイは、フェデレーション抽出とシャドウ抽出を使用するときに、抽出更新ジョブを支援します。バックグラウンダーのノード ロールの詳細については、ファイル ストア ノード ロールを参照してください。

  さらに、ノード1 にはリポジトリプロセスとファイル ストア プロセスがあり、バックアップの実行に必要なすべてのデータがノード1 に存在するため、バックアップのパフォーマンスを向上させることができます。

  バックグラウンダーのノード ロールは、Tableau Server バージョン 2019.3 以降の Advanced Management で使用できます。

- **抽出クエリのファイル ストアのノード ロール: 専用** データ エンジン ノードであるノード5 とノード6 には、抽出クエリのファイル ストアのノード ロールがあり、viz のロード、サブスクリプション、データ駆動型アラートのクエリのみを処理します。

- **抽出クエリのインタラクティブなファイル ストアのノード ロール:** 抽出クエリのファイル ストアのノード ロールが割り当てられた専用データ エンジン ノードの場合、サーバー管理者は、インタ

ラクティブなワークロードとスケジュールされたワークロードを分離して、特定の**専用**データエンジン ノードで実行することができます。これは、大量のサブスクリプションが発生しているときに、多くのユーザーがワークブックを操作したり、読み込んだりしている場合に役立ちます。たとえば、月曜日の午前 8 時に 1000 のサブスクリプションがスケジュールされているとします。同時に、1 日の始めに多くのユーザーがダッシュボードを読み込んでいます。サブスクリプションにユーザー クエリの量が加わるため、ワークブックの読み込みが遅くなり、読み込み時間が変動しやすくなる可能性があります。抽出クエリのインタラクティブなファイル ストアのノード ロールを使用すると、専用データエンジン ノードを指定して、インタラクティブ ユーザー(画面を見て待機しているユーザー) のクエリのみを受け入れることができます。インタラクティブなワークロードに対して優先されるこれらの専用データエンジン ノードは、競合する大量のサブスクリプション ジョブから保護され、クエリ時間の一貫性が向上します。さらに、サーバー管理者はこのノード ロールを使用して、インタラクティブなワークロードとスケジュールされたワークロードにそれぞれ専用のデータエンジン ノードを追加できるため、増加に備えた優れたプランニングを行うことができます。詳細については、*ファイル ストア ノード ロール*を参照してください。

ファイル ストアのノード ロールは、Tableau Server バージョン 2020.4 以降の Advanced Management で使用できます。

- **外部ファイル ストアを使用した最適化:** この機能を使用すると、Tableau Server ノードでローカル ディスクを使用する代わりに、ファイル ストアのストレージとしてネットワーク共有を使用できます。一元化された場所にストレージを配置することで、ファイル ストア ノード間でデータを複製するために費やされるネットワーク トラフィックの量を大幅に削減できます。たとえば、ファイル ストアがローカル ディスクを使用している場合、ローカル ファイル ストアを使用して 1 GB の抽出が更新されると、1 GB のデータがネットワークを介してファイル ストア プロセスを実行しているすべてのノードに複製されます。Tableau Server が外部ファイル ストアで構成されている場合、1 GB の抽出をネットワーク共有にコピーする必要があるのは 1 回のみで、すべてのファイル ストア ノードがその単一のコピーにアクセスできます。ストレージを一元化することで、ファイル ストア ノードで必要なローカル ストレージの総量も削減されます。

さらに、Tableau Server バックアップでは、スナップショット テクノロジーを活用して、バックアップを完了する時間を大幅に短縮します。

外部ファイル ストアの利点を享受するために専用のデータエンジン ノードを構成する必要はありません。ファイル ストア ノード ロールと抽出クエリのインタラクティブ ノード ロールを備えた追加のワークロード管理機能を併用できます。詳細については、Tableau Server 外部ファイル ストアトピックを参照してください。

外部ファイル ストアは、Tableau Server バージョン 2020.1 以降の Advanced Management で使用できます。

ノードの追加と再構成の時期

Tableau Server は、ニーズと要件の拡大に応じて、スケールアップおよびスケールアウトを行うことができます。ここでは、ノードをシステムに追加し、サーバーを再構成するか、またはその両方の操作を行う時期にあるかどうかの判断に役立ついくつかのガイドラインを示します。

- **100 を超える同時ユーザー**: 展開でユーザーを多用する (同時ビューアー数が 100 を超える) 場合は、十分な VizQL プロセスを持っておくことは重要ですが、ハードウェアの処理能力を超えるほど多くは必要ありません。また、Tableau Server のゲスト ユーザー アカウントを有効にすると、潜在的な同時ビューアーの数が思っていたユーザー リストの数を超える可能性があります。管理ビューは、これを判断する際に役立ちます。詳細については、特定のユーザーによるアクションを参照してください。

- **抽出の多用**および**頻繁な抽出の更新**: 抽出を使用すると、大量のメモリと CPU リソースが消費される可能性があります。サイトを抽出集中型とみなす測定方法は存在しません。少数の極めて大きな抽出を持っているだけで、非常に多数の小さな抽出を持っているのと同じように、サイトはこのカテゴリに入る場合があります。抽出が頻繁に更新される (1 日に数回など) サイトは、更新タスクを処理するバックグラウンド プロセスをより強調することで助けられる場合がよくあります。抽出でのバックグラウンド タスク管理ビューを使用して、現在の更新率を確認します。抽出が集中的に行われているサイトは、バックグラウンダー プロセスを独自のマシンに分離することでメリットがあります。詳細については、推奨されるベースライン

構成トピックの「2 ノード構成」を参照してください。

- **クエリ負荷の高い環境**: 抽出を使用するワークブックでクエリのパフォーマンスが低下している場合は、抽出でクエリを処理するノードを VizQL プロセスから隔離するとパフォーマンスが改善し、安定する可能性があります。詳細については、抽出クエリの負荷が大きい環境用に最適化を参照してください。

- **停止する可能性**: サーバー システムがミッション クリティカルとみなされ、高可用性を必要とする場合は、抽出、リポジトリ、およびゲートウェイを処理するサーバープロセスの冗長性が確保されるように設定できます。

パフォーマンスの記録

このセクションでは、パフォーマンス記録の作成方法や、結果を使用してワークブックのパフォーマンスを改善する方法について説明します。パフォーマンス記録を使用することで、ワークブック イベントの所要時間を確認できます。たとえば、データ ソースへの接続、クエリの実行、データのレンダリングなどにかかる時間がわかります。

パフォーマンスの記録の作成

Tableau のパフォーマンス記録機能は、ワークブックの操作中に主要なイベントのパフォーマンス情報を記録します。次に、Tableau が作成するワークブックでパフォーマンス メトリクスを確認し、パフォーマンスに影響することが知られているさまざまなイベントの分析やトラブルシューティングを行うことができます。

- クエリの実行
- クエリのコンパイル
- ジオコーディング
- データ ソースへの接続
- レイアウト計算
- 抽出の生成
- データのブレンド
- サーバー レンダリング (Tableau Server のみ)

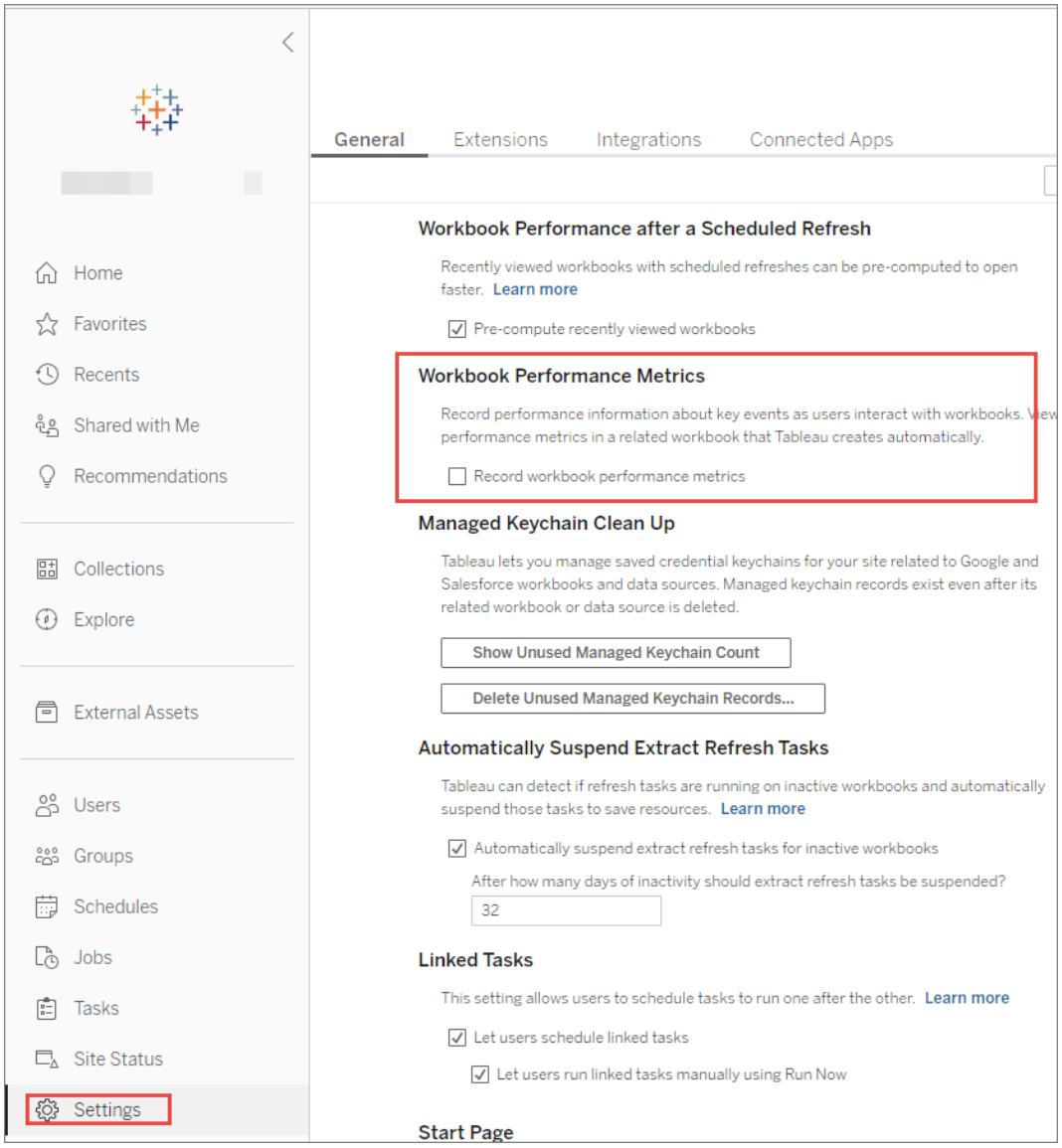
Tableau サポートにより、パフォーマンス問題の診断を支援するため、パフォーマンス ワークブックの作成が要請されることがあります。

> Tableau Server on Windows については、「パフォーマンスの記録の作成」を参照してください。

サイトのパフォーマンスの記録を開始する

既定では、サイトではパフォーマンスの記録は有効になっていません。サーバー管理者は、サイトごとの記録を有効にするかどうかを決定します。

1. パフォーマンスの記録を有効にするサイトへ移動します。
2. **[設定]** を選択します。



3. [ワークブック パフォーマンス メトリクス] で、**[ワークブック パフォーマンス メトリクスを記録]** を選択します。
4. **[保存]** を選択します。

ビューのパフォーマンスの記録を開始する

1. パフォーマンスを記録したいビューを開きます。

   ビューを開くと、Tableau Server によりURL の後に ":iid=<*n*>" が追加されます。これはセッションID です。例:

   ```
   http://10.32.139.22/#/views/Coffee_
   Sales2013/USSalesMarginsByAreaCode?:iid=1
   ```

2. セッションID の直前、表示 URL の末尾に `:record_performance=yes&` と入力します。例:

   ```
   http://10.32.139.22/#/views/Coffee_
   Sales2013/USSalesMarginsByAreaCode?:record_
   performance=yes&:iid=1
   ```

3. ツールバーで、**[更新]** ボタンを選択します。
4. ビューを読み込みます。

パフォーマンスの記録を表示

1. **[パフォーマンス]** を選択して、パフォーマンス ワークブックを開きます。これはパフォーマンスデータの最新スナップショットです。ビューを継続して操作しながら続けて追加のスナップショットを取ることができ、パフォーマンス データが累積されます。
2. 別のページへ移動するか URL から `:record_performance=yes` を削除すると、記録が停止します。

パフォーマンスの記録の解釈

ワークブックのパフォーマンスを評価するために記録を作成します。記録が完了したら、結果のワークブックをダウンロードし、Tableau Desktop で開いて分析することができます。

パフォーマンス記録ワークブックには、[Performance Summary (パフォーマンス サマリー)] と [Detailed Views (詳細ビュー)] という2 つの主要なダッシュボードがあります。[Performance Summary (パフォーマンス サマリー)] ダッシュボードには、最も時間のかかるイベントの高度な概要が表示されます。[Detailed Views (詳細ビュー)] ダッシュボードには詳細な情報が表示され、上級ユーザーがワークブックを作成するときに使用することを目的としています。**[Detailed Views (詳細**

ビュー)] ダッシュボードは、パフォーマンス記録ワークブックが Tableau Desktop を使用して開かれた場合にのみ表示されます。

Tableau Server でパフォーマンス記録を作成する方法については、パフォーマンスの記録の作成を参照してください。

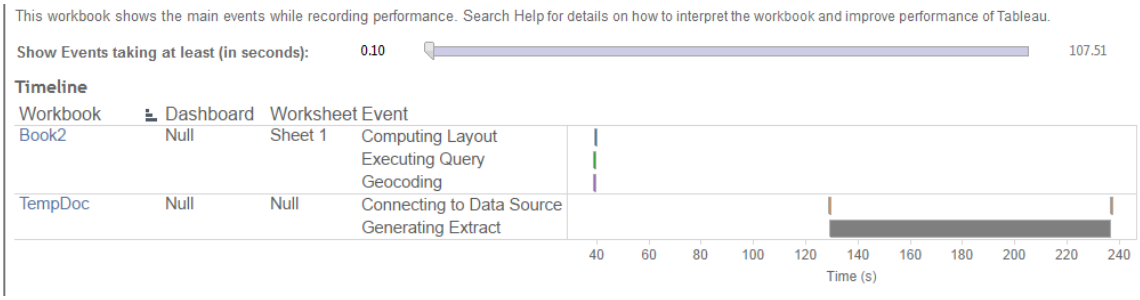
Performance Summary (パフォーマンス サマリー)

[Performance Summary (パフォーマンス サマリー)] ダッシュボードには、**[タイムライン]**、**[イベント]**、および **[クエリ]** の 3 つのビューがあります。

タイムライン

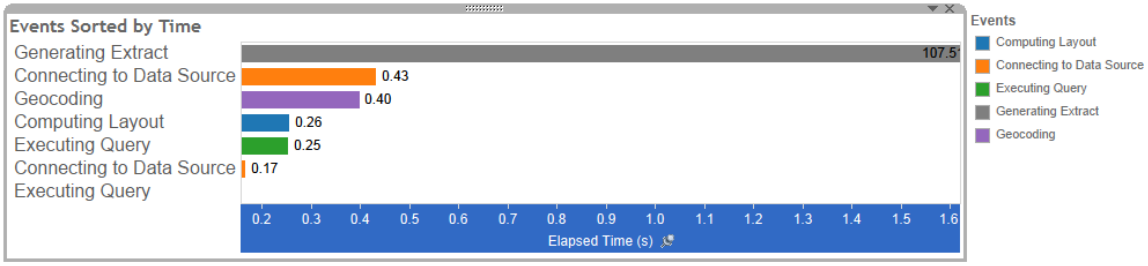
パフォーマンス サマリー記録ダッシュボードの一番上のビューには、記録中に発生したイベントが左から右へ時系列で配置され表示されます。一一番下の軸には、Tableauが起動してからの経過時間が秒数で表示されます。

[タイムライン] ビューでは、**[ワークブック]**、**[ダッシュボード]**、**[ワークシート]** の各列にイベントのコンテキストが識別されます。**[イベント]** の列には、イベントの性質が識別され、最終列には各イベントの期間と他の記録されたイベントとの時系列比較が表示されます。



イベント

パフォーマンス サマリー ダッシュボードの中央のビューには、期間 (長時間から短時間) の順にイベントが表示されます。期間がより長いイベントは、ワークブックを高速化する場合の最初の注目点を識別するのに役立ちます。



異なる色は異なるタイプのイベントを表します。記録できるイベントの範囲は、次のとおりです。

- レイアウトの計算

  レイアウトが長すぎる場合は、ワークブックの簡素化を検討します。

- データソースへの接続

  遅い接続は、ネットワーク問題またはデータベースサーバー問題が原因として考えられます。

- クエリのコンパイル

  このイベントは、Tableau がクエリの生成に費やした時間をキャプチャします。クエリのコンパイル時間が長い場合、生成されるクエリが複雑であるということです。そのような複雑さは、フィルターの数が多すぎたり計算が複雑であることに起因する場合もありますが、一般的にはワークブックが複雑であることに起因します。複雑な計算の例としては、長い計算、LOD 計算、ネストされた計算などがあります。ワークブックの簡略化、アクション フィルターの使用、または参照元データベースへの計算の移動を試みてください。

- クエリの実行

  - ライブ接続では、クエリに時間がかかりすぎる場合、参照元のデータ構造が Tableau 用に最適化されていない可能性があります。データベースサーバーのドキュメントを参照してください。代わりに、抽出を使用してパフォーマンスの速度を上げることを検討してください。

- 抽出でクエリに時間がかかりすぎる場合は、フィルターの使用を見直してください。フィルターの数が多すぎる場合、コンテキストフィルターの方が適していませんか。フィルターを使用するダッシュボードがある場合は、パフォーマンスの向上に役立つアクション フィルターの使用を検討してください。

- 抽出の生成

  抽出の生成を高速化するには、元のデータソースからデータの一部だけをインポートすることを検討します。たとえば、特定のデータフィールドでフィルターしたり、指定された行数やデータのパーセンテージに基づいてサンプルを作成したりすることができます。

- ジオコーディング

  ジオコーディングのパフォーマンスを高速化するには、少ないデータを使用するか、データをフィルターしてみます。

- データのブレンド

  データブレンドを高速化するには、使用するデータを少なくするか、データをフィルターしてみます。

- サーバー レンダリング

  別のマシンで追加の VizQL Server プロセスを実行することで、サーバー レンダリングを高速化できます。

クエリ

パフォーマンス サマリー ダッシュボードの **[タイム ライン]** または **[イベント]** セクションのいずれかで、**[クエリの実行]** イベントをクリックすると、そのクエリのテキストが [クエリ] セクションに表示されます。

パブリッシュされたデータソースに接続されている場合、クエリのテキストが XML で表示されます。データソースに直接接続されている場合、以下のようにクエリが SQL で表示されます。

Query

```
SELECT "State"."ID" AS "ID",
  "StateSynonyms"."Name" AS "State_Name",
  "State"."ParentID" AS "State_ParentID"
FROM "StateSynonyms"
  INNER JOIN "State" ON (("State"."ID" = "StateSynonyms"."ParentID") AND ("State"."MapCode" = "StateSynonyms"."MapCode
```

合理的であれば、データベース チームと協力するクエリテキストを使用し、データベース レベルでデータを最適化できます。クエリが切り捨てられている場合は、Tableau ログで完全なクエリを検索する必要があります。ほとんどのデータベース サーバーでは、インデックスの追加やその他の方法によりクエリを最適化する方法が提案されています。詳細については、ご使用のデータベース サーバーのドキュメンテーションを参照してください。

効率性を高めるため、Tableau はデータに対する複数のクエリを 1 つのクエリへと組み合わせることがあります。この場合、NULL ワークシートに **[クエリの実行]** イベントが表示され、名前を付けたワークシートには実行されているクエリが何も表示されない場合があります。

Detailed Timeline (詳細なタイムライン)

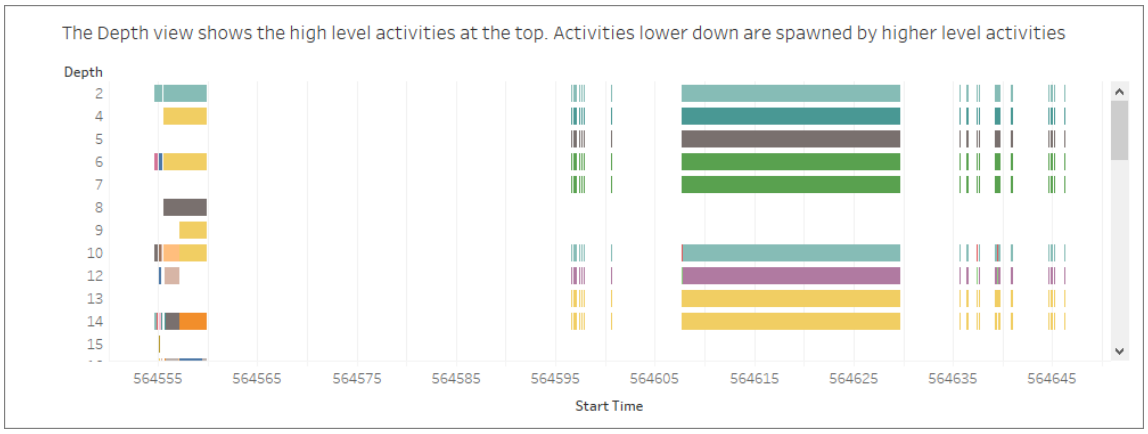
このビューは **[タイムライン]** ビューの詳細版です。すべてのイベントを表示し、**[タイムライン]** ビューでグループ化された個々のアイテムを分離して表示します。上級ユーザーがワークブックをデザインする際に使用することを目的としています。

Detailed Views (詳細ビュー)

[Detailed Views (詳細ビュー)] ダッシュボードには、**[Depth (深度)]**、**[Exclusive CPU (排他的 CPU)]**、**[Inclusive CPU (包括的 CPU)]**、**[Elapsed Time (経過時間)]** ビューがあります。

Depth (深度)

[Depth (深度)] ビューは、**[Detailed Views (詳細ビュー)]** ダッシュボードの一番上に表示されます。要求が行われたときに何が起きているかを把握できます。このビューは、単一のユーザー要求を表示するようにフィルターした場合に最も役立ちます。ユーザー要求の例としては、ビューの読み込み、マークの選択、フィルターの変更などがあります。



[Depth (深度)] ビューの各バーは、単一のアクティビティを表します。アクティビティとは、ユーザー要求の処理の一部として実行される作業単位です。単一のユーザー要求により複数のアクティビティが実行されます。[Depth (深度)] ビュー上の各バーの長さは、バーが表すアクティビティの経過時間に比例します。

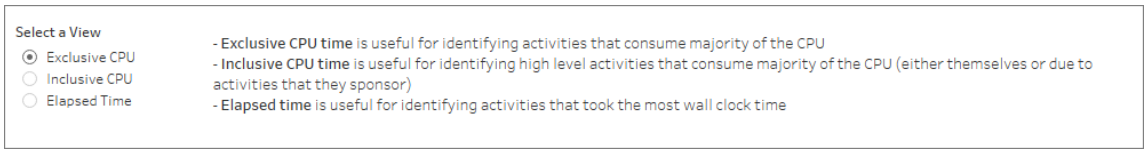
上位レベルのアクティビティがビューの上部に表示されます。下位のアクティビティは、上位レベルのアクティビティによって生成された子アクティビティです。

各バー上にカーソルを合わせると、アクティビティに関する詳細情報が表示され、次のセクションで説明する **[CPU]** および **[Elapsed Time (経過時間)]** ビューで対応する行がハイライトされます。

要求のどの部分が最も時間が掛かったかを絞り込むために、長時間実行される最上位レベルのアクティビティを調査します。

CPU および Elapsed Time (経過時間)

[CPU] および **[Elapsed Time (経過時間)]** ビューは、**[Detailed Views (詳細ビュー)]** ダッシュボードの一番下に表示されます。ラジオ ボタンをクリックすると、**[Exclusive CPU (排他的 CPU)]**、**[Inclusive CPU (包括的 CPU)]**、**[Elapsed Time (経過時間)]** の各ビューが切り替わります。



[Depth (深度)] ビューは、長時間実行されるアクティビティを視覚的にすばやく識別するのに役立ちます。ただし、各インスタンスの時間が短いアクティビティが複数回発生する場合、そのようなアクティビティがハイライトされるとは限りません。**[Exclusive CPU (排他的 CPU)]**、**[Inclusive CPU (包括的 CPU)]**、**[Elapsed Time (経過時間)]** ビューには、各アクティビティの総統計が表示されます。アクティビティが実行された回数が **[カウント]** 列に表示され、1 つのアクティビティに掛かった合計時間が棒グラフで表示されます。

効率性を高めるため、Tableau はデータに対する複数のクエリを 1 つのクエリへと組み合わせることがあります。この場合、NULL ワークシートに **[クエリの実行]** イベントが表示され、名前を付けたワークシートには実行されているクエリが何も表示されない場合があります。

パフォーマンス モニタリング ツール

このトピックは、パフォーマンスの監視や調節に使用できる外部リソースについて説明します。

Tableau Server には、サーバーのパフォーマンスと状態を監視するために使用できるツールが複数あります。これらのツールの詳細については、パフォーマンス モニタリングの概要を参照してください。

免責条項: このトピックには、サードパーティおよびコミュニティでサポートされる製品に関する情報が含まれています。サードパーティおよびコミュニティに関する内容が正確であるように確認する最善の努力を払っていますが、ここに参照される情報は、予告なしに変更される可能性があることに注意してください。最新の情報については、ここで参照されている製品のドキュメントを参照してください。コミュニティでサポートされるツールの詳細については、「Support levels for IT and developer tools」(英語) を参照してください。

- TabJolt。ロードおよびパフォーマンス テスト ツール。Tableau Server をお使いの環境のワークロードで拡張する方法について理解したり、必要となるスケーラビリティおよび容量を把握したりできます。以下に TabJolt を使用するときのための重要な使用事例を一部紹介します。
    - 本番環境にパブリッシュする前にサーバーのパフォーマンスとテスト展開の基礎を確立する。

    - 新しい Tableau Server で、使用中の環境での新しいサーバーの拡張方法 (特にハードウェアおよびワークロード) について理解する。
    - アップグレードする前に使用中の環境で新しいバージョンが拡張されることを理解する。
    - 最善のサーバー展開構成、特定のハードウェア、ワークブック、および環境について確認する。

- Replayer。他のサーバーまたは構成に対し、Tableau Server からログベースの実際のユーザー トラフィックを再生できるツール。Tableau Server のシングルユーザーまたはマルチユーザー セッションを再生します。[再生] を使用できる方法を次に示します。
    - 特定の Tableau Server セッションを再生し、開始時刻および RequestID に基づいてセッションをフィルターします。
    - これにより読み込み条件をシミュレーションし、Tableau Server のインストールを拡張したり調整したりする方法をテストできるようにします。
    - Tableau Server アップグレードに対するエンドツーエンドのユーザー シナリオを実行および比較して、回帰テストを実行します。
    - シングルユーザー セッションで発生する HTTP の例外をキャプチャし、レポートします。
    - エラー内容を再生してトラブルシューティングし、修正されたかどうかを確認します。
- Scout。Tableau Desktop と Tableau Server の両方のワークブックでパフォーマンス メトリクスをキャプチャする探索ツール。次のようにいくつかの使用方法があります。
    - サーバー上で遅延が発生しているワークブックを検出する。
    - サーバーの構成やトポロジを変更した後のパフォーマンス改善や回帰について検証する。

    - 新しい Tableau Server にアップグレードした後でワークブックおよびダッシュボードが適切に読み込まれていることを検証する。
    - データソースを変更した後もワークブックが適切に動作していることを検証する。
- Sitescope。エージェントレスアプリケーション監視ツール。
- Zabbix。オープンソースの、リアルタイム監視ツール。
- Splunk 。マシンデータ(ログを含む)を監視および分析するツール。
- Graylog。オープンソースのログ管理ツール。

クライアント側レンダリングの構成

Tableau Server のビューに移動する際、ビューの表示に必要な処理 (*レンダリング*) を、クライアントデバイスまたは Tableau Server のいずれかで実行できます。選択内容は、マーク、行、列などの数によって決まるビューの複雑性に依存します。ビューがシンプルであるほど、クライアントデバイスでビューをより速くレンダリングできます。ビューが複雑であるほど、Tableau Server に要求を送信し、サーバーの優れた計算機能を活用した方が速くなります。

> **注:** ビューがポリゴンマークタイプまたはページ履歴機能を使用している場合、クライアント側レンダリングが有効になっていてもサーバー側レンダリングが常に実行されます。

サポートされるブラウザー

クライアント側レンダリングは、Internet Explorer バージョン 9.0 以降、Firefox、Chrome、Safari でサポートされています。これらの Web ブラウザーにはすべて、クライアント側レンダリングに必要な HTML 5 `<canvas>` 要素が含まれています。

クライアント側レンダリングは、Tableau Mobile アプリでもサポートされています。

コンピューターとモバイル デバイス用の複雑性のしきい値を構成する

PC はモバイル デバイスより処理能力が高いため、Tableau Server はモバイル デバイスよりPC 上でより多くのクライアント側レンダリングを実行します。

サーバー管理者は、複雑性のしきい値を調節することで、PC やモバイル デバイスでクライアント側レンダリングを発生させるタイミングを構成できます。たとえば、ビューの表示が遅いと感じたら、モバイル デバイスのしきい値を下げることができます。または、しきい値を増やして Tableau Server に対する要求の数を減らすことができます。

既定では、コンピュータWeb ブラウザーの複雑性しきい値は 100 です。PC の複雑性しきい値を調節するには、次のコマンドを使用します。

```
tsm configuration set -k vizqlserver.browser.render_threshold -v
[new value]
```

既定では、モバイル デバイスの複雑性しきい値が 60 に設定されています。モバイル デバイスの複雑性しきい値を調節するには、次のコマンドを使用します。

```
tsm configuration set -k vizqlserver.browser.render_threshold_
mobile -v [new value]
```

たとえば、モバイルしきい値を 40 に変更するには、次のコマンドを入力できます。

```
tsm configuration set -k vizqlserver.browser.render_threshold_
mobile -v 40
```

tsm オプション設定の使用方法の詳細については、tsm configuration set のオプションを参照してください。

クライアント側レンダリングを無効にする

クライアント側レンダリングは既定で有効になっており、ビューのパフォーマンスを改善することをお勧めします。ただし、テスト目的や、主に非常に処理能力の低いコンピューターやモバイル デバイスがサーバーにアクセスしている場合など、クライアント側のレンダリングを一時的に無効にする場合があります。

次のコマンドを使用して、クライアント側のレンダリングを無効にします。

```
tsm configuration set -k vizqlserver.browser.render -v false
```

tsm オプション設定の使用方法の詳細については、tsm configuration set のオプションを参照してください。

URL パラメーターを使用したテスト

セッション ベースでサーバー側のレンダリングをテストするには、ビューの URL の最後に「`?:render=false`」と入力します。例:

```
http://localhost/views/Supplies/MyView?:render=false
```

クライアント側レンダリングが Tableau Server で無効になっている場合は、`?:render=true` を入力してセッションに対して有効にします。

```
http://localhost/views/Supplies/MyView?:render=true
```

また、サーバーおよびネットワークの条件に対してサーバー全体のしきい値を調整するのに適切であるかどうかを確認するために、個別のビューで特定の複雑度のしきい値をテストすることもできます。たとえば、低い複雑度 (80 など) または高い複雑度 (120 など) の転換点により、ユーザー操作に対する応答性が向上する場合があります。しきい値をテストするには、サーバーの既定の構成 (クライアント側レンダリングが有効) を維持し、ビューの URL の最後にテストしきい値の数値を入力できます。例は次のとおりです。

```
http://localhost/views/Supplies/MyView?:render=80
```

ビューの高速化

Creator または Explorer のライセンスを持つ管理者とワークブックの所有者は、Tableau Server でワークブックを高速化できます。管理者は、個々のビューを中断したり、サイトの高速化をオフにしたりすることができます。ビューの高速化では、バックグラウンドプロセスでワークブックのデータを事前に計算して取り出しておくことで、ビューを高速に読み込みます。ビューを読み込む際の潜在的なボトルネックは 2 つあります。

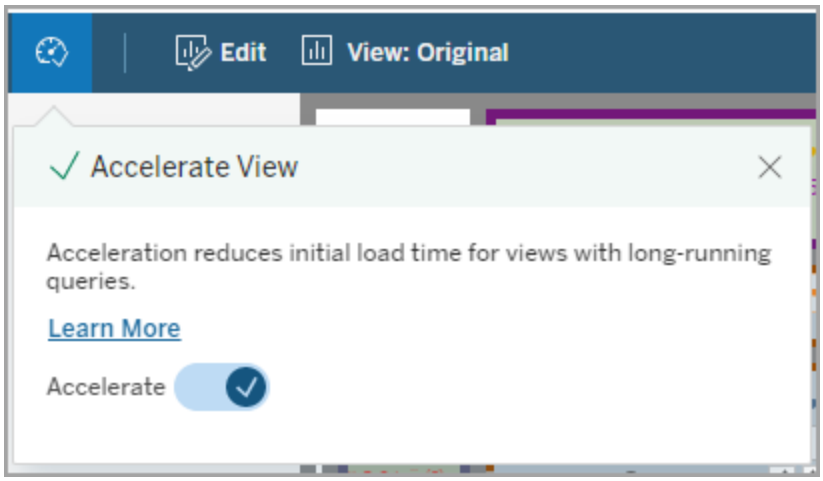
1. クエリ(データ ソースからのデータの取り出し)。
2. レンダリング (図形の描画やマップのレンダリングなどのビジュアルの作成)。

ワークブックの読み込み時間は、この 2 つのステップにかかる合計時間によって決まります。ただし、すべてのビューを高速化できるわけではありません。ビューの高速化によって、最初のステップ (クエリ) のパフォーマンスが向上します。ビューの読み込みがクエリ以外の理由で遅い場合、ビューの高速化ではワークブックのパフォーマンスは向上しません。

ユーザーが高速化されたビューの上にカスタム ビューを作成すると、最も使用される 10 個のカスタム ビューが自動的に事前計算されます。これらの高速化されたカスタム ビューは、ビューの制限数にはカウントされません。過去 14 日間アクセスされていないカスタム ビューは高速化されません。カスタム ビューを直接高速化すると、元のビューとカスタム ビューの両方が高速化されます。

ビューの高速化

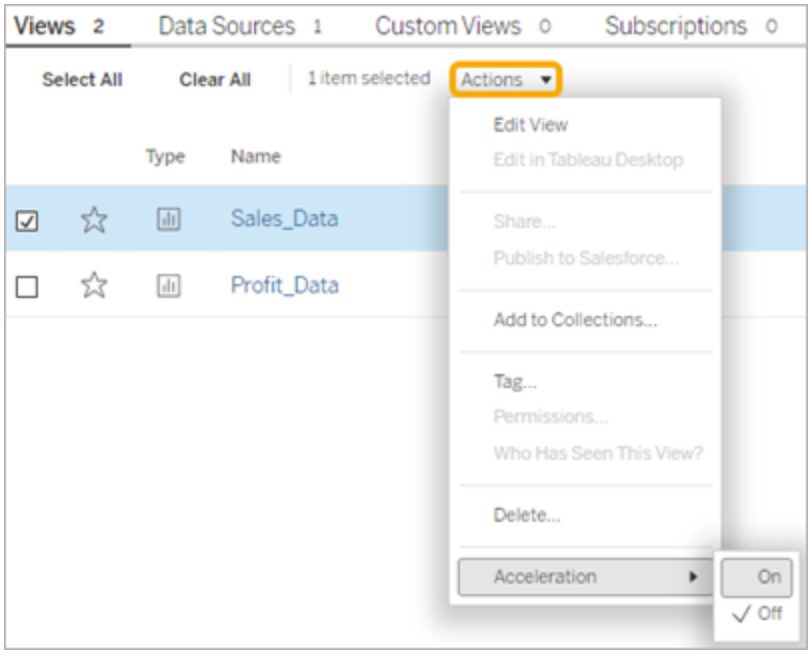
1. Tableau Server のサイトにサインインします。
2. [ホーム] ページまたは [探索] ページから、高速化したいビューに移動します。
3. **[アクセラレート]** アイコンを選択し、**[アクセラレート]**.に切り替えます。



注: 高速化 スイッチをオフにすると、ユーザーはビューを高速化できなくなりますが、ビューから高速化オプションは削除されません。

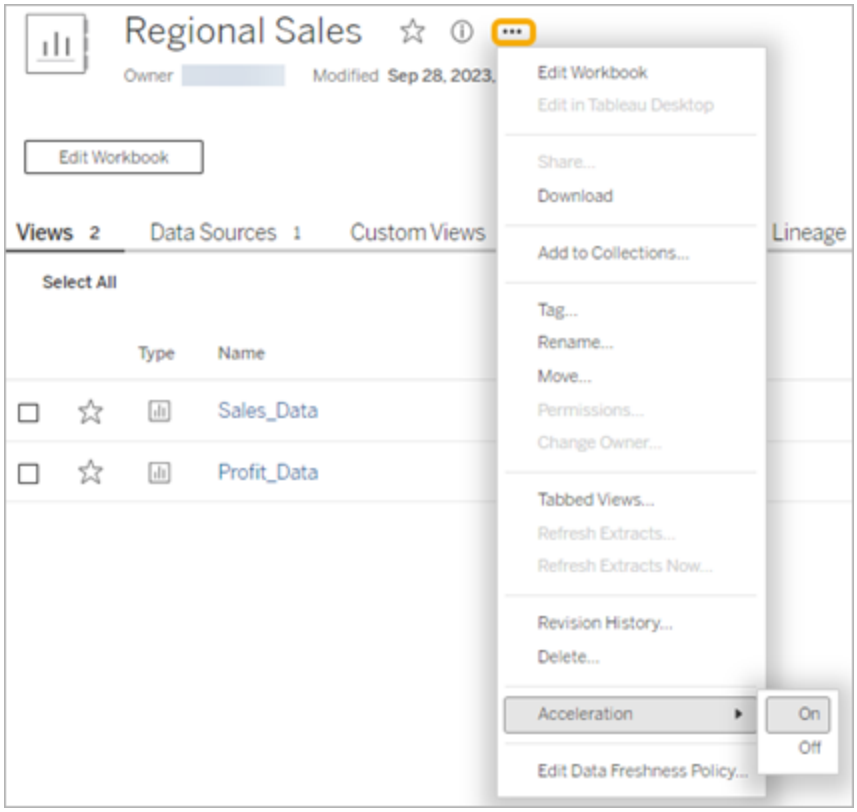
次の 3 つの方法のいずれかで、ワークブック ページからビューを高速化することもできます。

1. 目的のビューを選択し、**[アクション]** メニューから**[高速化]**、**[オン]** の順に選択します。



2. 目的のビューの **[その他のオプション]** (...) メニューを選択し、**[高速化]**、**[オン]** の順に選択します。

3. ワークブック内のすべてのビューを高速化するには、**[その他のオプション] (...)** メニューから **[高速化]**、**[オン]** の順に選択します。



注: ビューの高速化は、Tableau Desktop では使用できません。

ビューの高速化が利用できない、停止している、効果がない理由

ビューの高速化を利用できない

ビューの高速化を利用できないシナリオがいくつかあります。

問題	解決策
ビューには、クエリの実行に必要な時間の履歴データがありません。	ビューが読み込まれるたびに、viz データが使用可能になるまでに数分かかります。新しく作成したビューの場合は、ビューを読み込んでから数分間待って高速化を試みます。
ビューのクエリの実行にか	このようなビューでは、高速化によってビューのパフォーマンスが大

かる過去の時間は 2 秒未満です。	幅に向上するわけではないため、高速化はサポートされていません。
ビューには認証資格情報が埋め込まれていません。	データを事前に計算するには、Tableau はユーザーによる操作を必要とすることなく、バックグラウンドで自動的にデータソースに接続する必要があります。その結果、ビューの高速化を利用できるのは、接続の認証資格情報が埋め込まれているワークブックのみです。
ビューにユーザーベースの関数が含まれているか、ビューにユーザーベースの関数を含むデータソースがあります。	現在、Tableau はそのようなビューの高速化をサポートしていません。ユーザーベースの関数の例としては、USERDOMAIN() や USERNAME() があります。
ビューの所有者が非アクティブです。	データを事前に計算するには、ビューの所有者がアクティブなユーザーである必要があります。Tableau は、所有者が非アクティブなビューの高速化をサポートしていません。アクティブなユーザーに所有権を変更します。
ビューのデータ鮮度ポリシーが 2 時間未満です。	頻繁に更新されるビューの高速化はコストが高くなる可能性があり、Tableau はサイトのパフォーマンスに過負荷をかけることを望んでいません。詳細については、「データ鮮度ポリシーの設定」を参照してください。
サイトが高速化できるビュー数の制限に達しました。	サイト設定を更新して、高速化できる最大ビュー数を増やすか、[制限なし] を選択します。

ビューの高速化が停止している

高速化が停止するシナリオがいくつかあります。

問題	**解決策**
ビューの高速化が停止しました。	ビューの高速化をもう一度有効化します。
ビューのデータを事前に計算するためにバックグラウンドで	ジョブが定期的に失敗すると、ビューは自動的に一時停止します。次のような場合、ジョブは失敗する可能性があります。

高速化ジョブが実行されています。	• ビューの認証資格情報の有効期限が切れています。ビューの認証資格情報を更新します。 • ビューの所有者が非アクティブになります。アクティブなユーザーに所有権を変更します。 • ビューのデータソースが削除されました。ビューの所有者に連絡して、データソースを更新してください。 • データを事前に計算するジョブに時間がかかりすぎてタイムアウトになります。ビューの高速化の最長実行時間は 30 分です。ビューの所有者に連絡して、ワークブックを最適化してください。

ビューの高速化の効果がない

ビューの高速化により、ビューのクエリの実行にかかる時間が短縮されます。クエリの実行にかかる時間が viz の読み込み時間のボトルネックになっていない場合、viz の読み込みでパフォーマンスが大幅に向上することはありません。加えて、ビューには通常、多くのクエリが含まれます。now() などの一時的な関数や相対日付フィルターを使用するクエリは、高速化することができません。一時的な関数に長時間を要するクエリがビューにある場合、viz の読み込みでパフォーマンスの向上は得られません。

高速化されたビューの更新

高速化されたビューのイベントベースの更新

抽出を含むワークブックでは、抽出更新が完了すると、すべての高速化されたビューが更新されます。ワークブックが再パブリッシュまたは名前変更されると、ワークブック内のすべての高速化されたビューが更新されます。

高速化されたビューのスケジュールベースの更新

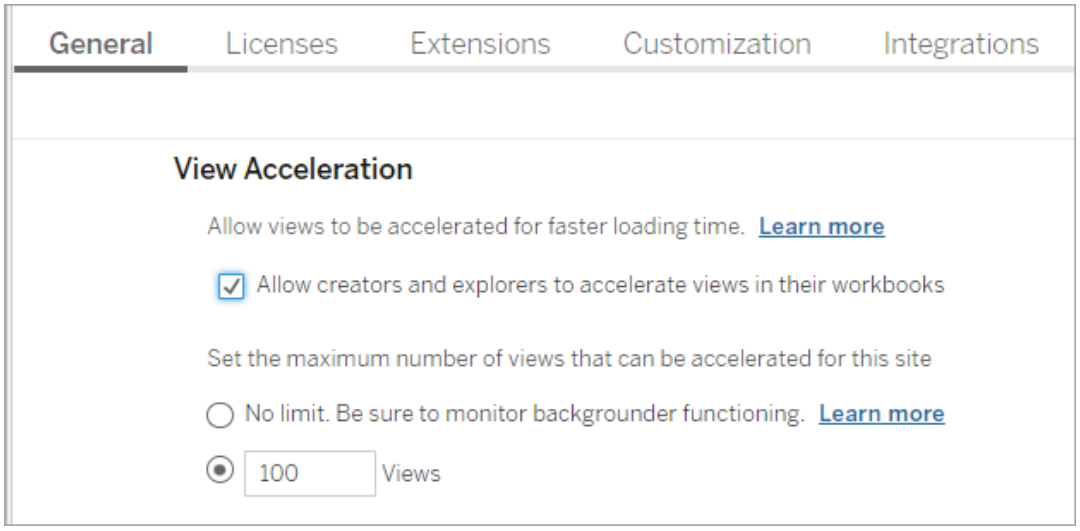
高速化されたビューを更新するスケジュールを設定できるのは、ワークブックに少なくとも 1 つのライブデータソースがある場合です。

ライブデータソースを利用するワークブックの高速化されたビューは、ワークブックのデータ鮮度ポリシーに基づいて更新されます。データを更新するスケジュールが設定されていない場合は、デフォルトのデータ鮮度ポリシーが使用されます。詳細については、「ワークブックのデータ鮮度ポリシーの編集」を参照してください。

サイトのビューの高速化を管理する

ビューの高速化は既定で許可されます。

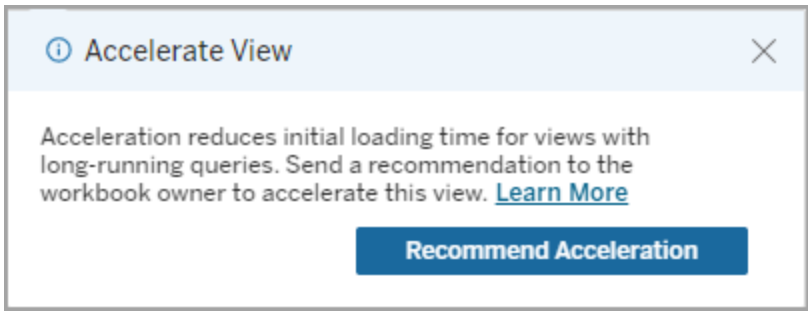
1. Tableau Server のサイトにサインインします。
2. 左側のペインで、**[設定]** を選択します。
3. **[全般]** タブで、**[ビューの高速化]** セクションまでスクロールします。
4. Creator とExplorer がワークブックでビューを高速化できるように、チェックボックスをオンにします。サイトのビューの高速化をオフにするには、チェックボックスをオフにします。
   注: チェックボックスをオフにすると、ユーザーはワークブック内のビューを高速化できなくなりますが、ビューから高速化オプションは削除されません。
5. サイトで高速化できるビューの最大数を入力するか、**[制限なし]** を選択します。



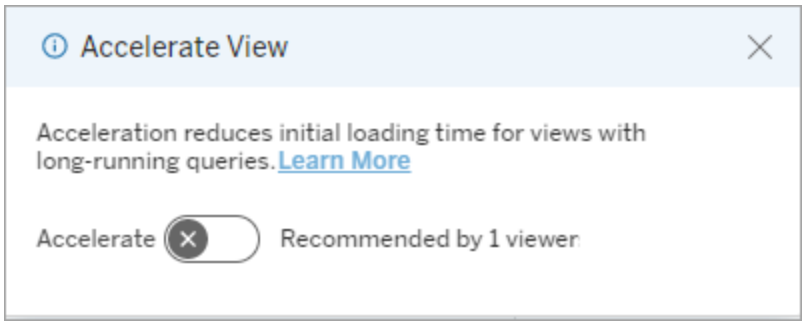
お勧めビューをアクセラレートする

ワークブックのクエリ時間と使用量に基づいて、Tableau は、速度が遅くなっている人気のビューやダッシュボードのパフォーマンスを向上させるためにアクセラレーションを推奨することがあります。ビューでアクセラレーションが利用できる場合、ユーザーは、30 日ごとにビューのアクセラレーションを推奨できます。

ユーザーがビューにアクセスすると、サイト管理者またはワークブック所有者にアクセラレーションを推奨するオプションが表示されます。



サイト管理者またはワークブック所有者が同じビューにアクセスすると、ビューを高速化するオプションに加えて、高速化を推奨しているユーザーの数が表示されます。



アクセラレーションが推奨されているビューを管理する

サイト管理者は、Tableau がビューのアクセラレーションをいつ推奨したかを確認できます。

1. Tableau サイトにサインインします。
2. 左側のペインで、**[タスク]** を選択します。
3. **[Acceleration status (アクセラレーションのステータス)]** 列から、**[Recommended (推奨)]** ステータスを持つビューを確認します。右側のペインの **[フィルター]** を使用して、**[推奨]** ステータスを持つビューをフィルターすることもできます。

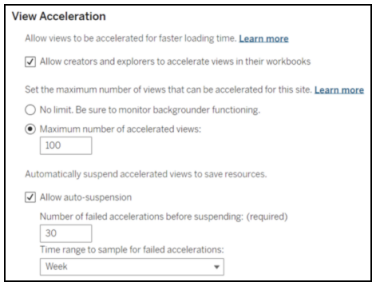
ワークブック所有者または管理者としてアクセラレーションの推奨をパーソナライズできます。

1. Tableau サイトにサインインします。
2. ページの右上から、アカウント メニューのアイコンを選択します。
3. **[マイ コンテンツ]** を選択します。
4. **[パフォーマンス]** タブを選択します。
5. **[アクション]** 列から **[アクセラレート]** を選択します。

リソースを節約するためにアクセラレーションを自動的に一時停止する

リソースを節約するために、管理者は常に失敗しているビューのアクセラレーションを自動的に一時停止できます。管理者は、アクセラレーションを自動的に一時停止する前に、1 日、1 週間、または 1 か月あたりのアクセラレーション タスクの失敗回数のしきい値を設定できます。

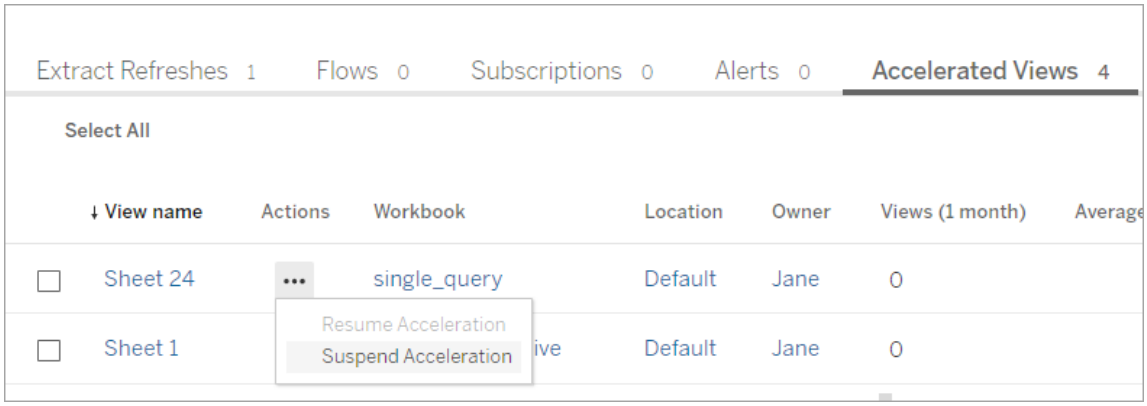
1. Tableau サイトにサインインします。
2. 左側のペインで、**[設定]** を選択します。
3. **[全般]** タブで、**[ビューの高速化]** セクションまでスクロールします。
4. **[Automatically suspend accelerated views to save resources (アクセラレートされたビューを自動的に一時停止してリソースを節約する)]** で、**[Allow auto-suspension (自動停止を許可する)]** を確認します。
5. ビューの高速化が一時停止されるまでの間にアクセラレーションが失敗した回数を入力します。
6. サンプリングする時間範囲 (日、週、月) を選択します。
   たとえば、数値として 25 を入力し、期間として週を入力した場合、ビューが 7 日間で 25 回以上失敗すると、ビューの高速化が一時停止されます。
7. **[保存]** を選択します。



高速化されたワークブックの表示と管理

1. Tableau サイトにサインインします。
2. 左側のペインで、**[タスク]** を選択します。
3. **[アクセラレートされたビュー]** タブを選択します。
4. **[アクション]** メニュー (...) を選択して、選択した 1 つまたは複数のビューの高速化を再開また

は一時停止します。



ビューの高速化の通知を管理する

管理者は、ビューが自動的に一時停止された場合の通知を受け取るかどうかを管理できます。

1. Tableau サイトにサインインします。
2. 左側のペインで、**[設定]** を選択します。
3. **[全般]** タブで、**[通知の管理]** セクションまでスクロールします。
4. 自動的に一時停止されたビューの通知を受け取るには、**[ビューの高速化]** のチェックボックスをオンにします。
5. **[保存]** を選択します。

ビューが自動的に一時停止された場合、サイト管理者とサーバー管理者に通知が送られます。通知には、ビューが一時停止された理由とその時間に関する情報が含まれます。通知を選択すると、**[タスク]** ページの **[アクセラレートされたビュー]** タブに移動します。管理者は、このページでアクセラレーションのステータスをフィルターして、自動的に一時停止されたビューを見つけることができます。

事前に計算を行うためのユーザー コンテキストを理解する

高速化を有効にしたワークブックの事前計算は、ユーザー1人によるユーザー コンテキストで実行されます。このユーザーは次のいずれかです。

- ワークブックの所有者 (ワークブックまたはデータ ソースにユーザー フィルターがない場合、またはデータ ソースにユーザー フィルターはあるが、データ ソースがパブリッシュされている場合)。
  - または -

- 前回ワークブックをパブリッシュしたときに、サムネイルの生成のために選択したユーザー(ワークブックにユーザー フィルターがあり、データ ソースがパブリッシュされていない場合)。

ビューの高速化のコスト

この機能を有効にすると、ビューの高速化のバックグラウンドプロセスによりデータ ソースから必要なデータがフェッチされるため、Tableau Server のバックグラウンドで処理される計算負荷とジョブ数が増加します。この機能を有効にしたワークブックのデータを事前に計算するバックグラウンド ジョブは、次のいずれかが発生した場合に実行されます。

- ワークブックとパブリッシュされたデータ ソースがもう一度パブリッシュされます (これには Web 編集の保存も含まれます)。
- ワークブックで使用する抽出を更新する。

管理者は、多数のワークブックでビューの高速化を有効にしたり、高速化ジョブのスケジュールを頻繁に設定する前に、これらのコストを検討する必要があります。

- ワークブックを頻繁に編集してパブリッシュを繰り返す場合は、パブリッシュするたびに事前計算が行われるため、高速化には不向きです。高速化は、情報を開示するためにパブリッシュするワークブックに使用することをお勧めします。
- ワークブックで複数の抽出を使用している場合は、その更新によってデータの事前計算がトリガーされます。設計上、ビューの高速化ジョブは抽出更新が正常に完了した後に実行されるため、有効なワークブックに対して頻繁に抽出更新を行うと、バックグラウンダー ジョブの負荷が急増する可能性があります。
- ワークブックの事前に計算されたデータは、Hyper にマテリアライズされたビューとして保存されます。

抽出クエリのロード バランシング

Tableau Server バージョン 2020.2 以降では、抽出ベースのクエリの負荷分散が改善され、抽出ベースのダッシュボードの読み込み時間が速くなっています。抽出が多く、ダッシュボードの負荷が大きい大規模な展開では、特にスタンドアロン ノードで Hyper が実行されている場合に最も改善が見られます。他のサーバープロセスと共有されるノードで Hyper が実行されている小規模な展開でも、パフォーマンスが改善されています。

Hyper では、消費しているリソースの量に関するサーバー ヘルス メトリックをログに記録し、同じサーバー ノードで実行されている可能性がある他の Tableau プロセスによるアカウントの負荷も考慮に

入れます。この情報に基づいて、抽出クエリは、クエリの処理に使用可能なリソースを持つノードに送信されます。システム リソースの評価に加えて、ロード バランサーでルーティングされるノードの抽出が既にキャッシュされている可能性が高まります。これにより、ノード間で重複する抽出の数が減り、メモリとI/O の使用状況が改善されます。

この機能を使用するには、キャッシュ サーバー プロセスがアクティブになっている必要があります。詳細については、Tableau Server のキャッシュ サーバーを参照してください。キャッシュ サーバー プロセスが有効になっていない場合、負荷分散は自動的に以前の機能に戻ります。

この機能は既定で有効になっています。無効にするには、次の tsm コマンドを使用します。

```
tsm configuration set -k hyper_standalone.consistent_
hashing.enabled -v false
```

```
tsm configuration set -k hyper_standalone.health.enabled -v false
```

次の tsm コマンドを使用して変更を適用します: `tsm pending-changes apply`。

詳細については、tsm configuration set のオプションを参照してください。

Tableau Server のモニタリング

Tableau Server のモニタリングに役立つ SMTP およびアラートとサブスクリプションを構成できます。

SMTP セットアップの構成

Tableau Server は、システム障害についてサーバー管理者にメールを送信し、サブスクライブしたビューおよびデータ主導アラートについてサーバー ユーザーにメールを送信することができます。ただし、最初に Tableau Server がメールの送信に使用する SMTP サーバーを設定する必要があります。SMTP を構成した後は手順を完了して通知を構成しますが (サーバー イベント通知の構成)、その後サーバーを起動または再起動するとメール通知が起動し、これで通知が正しく設定されたことが確認されます。

SMTP の構成には Tableau Server サービスの再起動が必要です。

セキュアSMTP

SMTP で TLS を有効にして構成するには、このトピックの説明に従って TSM CLI を使用する必要があります。Tableau Server では、STARTTLS (便宜的または明示的な TLS) のみがサポートされています。

組織が TLS 接続の検証にパブリック証明書を使用しない場合は、プライベート証明書を Tableau Server にアップロードして信頼できる接続を検証することができます。詳細については、tsm security custom-cert add コマンドを参照してください。

また、証明書の検証プロセスを無効にすることによってのみ、SMTP TLS で暗号化を構成することもできます。詳細については、下記の「*TSM CLI の使用*」タブの「*構成ファイル リファレンス*」セクションを参照してください。

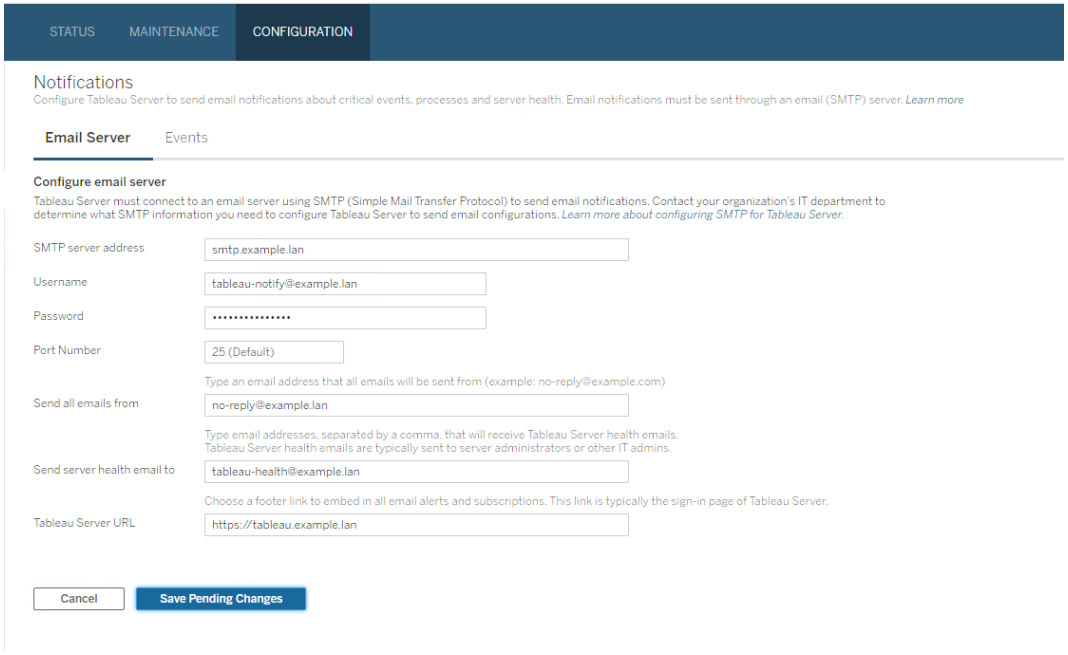
TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[通知]** をクリックし、**[メール サーバー]** をクリックします。

3. 自分の組織の SMTP 構成情報を次のように入力します。



4. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。

5. ページ上部の **[変更を保留中]** をクリックします。

6. **[変更を適用して再起動]** をクリックします。

7. `tsm email test-smtp-connection` を実行し、接続構成を表示して検証します。tsm email test-smtp-connection を参照してください。

TSM CLI の使用

SMTP の初期の構成では、次の構成ファイル テンプレートを使用して json ファイルを作成することをお勧めします。また、tsm configuration set で説明されている構文と共に以下の単一構成キーを設定することもできます。

1. 以下の json テンプレートをファイルにコピーします。

   > **重要**: 以下のテンプレートには、ほとんどの展開に共通するオプションが含まれています。テンプレートをテキストファイルにコピーした後、SMTP サーバー要件のオプション値を編集する必要があります。オプションの削除や追加が必要になる場合があります。サポートされているすべての SMTP キーオプションの詳細については、後続のリファレンスセクションを参照してください。

   ```
   {
   "configKeys": {
          "svcmonitor.notification.smtp.server": "SMTP server host
   name",
          "svcmonitor.notification.smtp.send_account": "SMTP user name",
          "svcmonitor.notification.smtp.port": 443,
          "svcmonitor.notification.smtp.password": "SMTP user account
   password",
          "svcmonitor.notification.smtp.ssl_enabled": true,
          "svcmonitor.notification.smtp.from_address": "From email
   address",
          "svcmonitor.notification.smtp.target_addresses": "To email
   address1,address2",
          "svcmonitor.notification.smtp.canonical_url": "Tableau Server
   URL"
          }
   }
   ```

2. `tsm settings import -f file.json` を実行し、適切な値を指定した json ファイルを Tableau サービスマネージャーに渡して Tableau Server の SMTP を構成します。Tableau サービスマネージャーによってエンティティの値が検証されます。

3. `tsm pending-changes apply` コマンドを実行して変更を適用します。tsm pending-changes apply を参照してください。

4. `tsm email test-smtp-connection` を実行し、接続構成を表示して検証します。tsm email test-smtp-connection を参照してください。

SMTP CLI 構成リファレンス

次の表は、TSM の CLI で SMTP を構成する際に使用できるすべてのオプションをリストしたものです。

オプション	説明
`svcmonitor.notification.smtp.server`	SMTP サーバーのアドレス。 例: `"svcmonitor.notification.smtp.server": "mail.example.com"`
`svcmonitor.notification.smtp.send_account`	SMTP アカウントのユーザー名。
`svcmonitor.notification.smtp.port`	SMTP サーバーのポート番号。既定は 25 です。
`svcmonitor.notification.smtp.password`	SMTP サーバー アカウントのパスワード。 例: `"svcmonitor.notification.smtp.password":"password"`
`svcmonitor.notification.smtp.ssl_enabled`	SMTP サーバーへの接続が暗号化されるかどうかを指定します。既定は false です。
`svcmonitor.notification.smtp.ssl_required`	有効にすると、Tableau Server は TLS を使用しない SMTP サーバーへの接続を拒否します。`svcmonitor.notification.smtp.ssl_enabled`オプションも true に設定する必要があります。

オプション	説明
	既定は false です。
`svcmonitor.notification.smtp.ssl_check_server_identity`	true に設定すると、Tableau Server は RFC 2595 で指定されている SMTP サーバー ID を確認します。サーバーの証明書の内容に基づきこの追加確認を行うのは、中間者攻撃を防ぐことが目的です。 既定は false です。
`svcmonitor.notification.smtp.ssl_trust_all_hosts`	TLS を使用する場合、すべてのメール サーバーからの証明書を信頼し、証明書の信頼チェーンの有効性を無視します。このキーを true に設定すると、TLS は SMTP ホストへのトラフィックの暗号化にのみ使用されます。 既定は false です。
`svcmonitor.notification.smtp.ssl_ciphers`	暗号スイートの既定とサポート対象のセットは、Tableau Server でインストールされている JDK のバージョンによって定義されます。サポート対象の暗号と既定の暗号のリストについては、以下の TLS 暗号セクションを参照してください。 Tableau Server が SMTP TLS 接続に使用する暗号スイートを更新するには、この値に暗号スイートを空白で区切ったリストを入力します。たとえば、`"TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256 TLS_DHE_DSS_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"` のように使用します。

オプション	説明
`svcmonitor.notification.smtp.ssl_versions`	このバージョンの Tableau Server で有効な既定の TLS バージョンは TLSv1、TLSv1.1、TLSv1.2、TLSv1.3 です。 TLS バージョンのサポートは、Tableau Server でインストールされている JDK のバージョンによって定義されます。 サポートされている TLS のバージョンは `SSLv2Hello, SSLv3, TLSv1, TLSv1.1, TLSv1.2, TLSv1.3` です。 Tableau Server が SMTP TLS 接続に使用するバージョンを更新するには、この値にバージョンを空白で区切ったリストを入力します。たとえば、`"TLSv1.2 TLSv1.3"` のように使用します。
`svcmonitor.notification.smtp.from_address`	システム障害が発生した場合に通知を送信する電子メール アドレス。電子メール アドレスは有効な構文 (ITalerts@bigco.com や noreply@mycompany など) でなければなりませんが、Tableau Server 上の実際のメール アカウントである必要はありません。(一部の SMTP サーバーでは、実際のメール アカウントが必要となる場合があります)。 **注**: サイトごとにシステム全体のメール アドレスを上書きできます。詳細については、サイトとはを参照してください。 例:

オプション	説明
	"svcmonitor.notification.smtp.from_address": "donot-reply@example.com"
svcmonitor.notification.smtp.target_addresses	通知を受信する電子メール アドレス。メール通知が有効化されている場合、1 つ以上のアドレスを含める必要があります。複数のアドレスはコンマで区切ります。 例: "svcmonitor.notification.smtp.target_addresses": "iluvdata@example.com"
svcmonitor.notification.smtp.canonical_url	Tableau Server の URL。http:// または https:// を入力してから Tableau Server の名前または IP アドレスを入力します。サブスクリプション メールのフッターで使用します。 例: "svcmonitor.notification.smtp.canonical_url": "http://myserver.example.com"

TLS 暗号

以下は、Tableau Server に含まれる JDK でサポートされている TLS 暗号のリストです。このバージョンの Tableau Server では、これらの暗号はすべて既定で有効になっています。上記の表の説明に従い、svcmonitor.notification.smtp.ssl_ciphers オプションを空白で区切ったリストを入力することで、SMTP 構成の暗号のカスタム リストを指定することができます。

TLS_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384

TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_128_GCM_SHA256	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256	TLS_DHE_DSS_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_256_GCM_SHA384	TLS_EMPTY_RENEGOTIATION_INFO_SCSV
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	TLS_DHE_DSS_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_SHA	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	TLS_DHE_RSA_WITH_AES_256_CBC_SHA

TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	TLS_DHE_DSS_WITH_AES_128_CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
TLS_RSA_WITH_AES_128_CBC_SHA	TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	TLS_AES_256_GCM_SHA384
TLS_AES_128_GCM_SHA256	

サーバー イベント通知の構成

Tableau サービス マネージャー (TSM) 管理者は、次のイベントの通知を許可するように Tableau Server を構成できます。

- コンテンツの更新
  - 抽出失敗 (既定では有効)
  - ユーザーのサブスクリプション ビュー (既定では無効)

- サーバーの健全性モニタリング
  - サーバー ステータスの変化 (既定では無効)
  - デスクトップ ライセンス レポート(既定では無効)
- ドライブの空き容量
  - ディスク空き容量が事前に設定したしきい値以下になった場合に送信するメール アラート(デフォルトでは無効)
  - 使用履歴の記録 (既定では有効)

注: サブスクリプションまたは通知を設定する前に、SMTP を設定する必要があります。詳細については、SMTP セットアップの構成を参照してください。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   `https://<tsm-computer-name>:8850`。

   詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[構成]** タブで **[通知]** をクリックし、**[イベント]** をクリックします。

3. 組織に合わせた通知設定を構成します。

   - コンテンツの更新

     - **抽出更新の失敗の際にメールを送信する**

       このオプションが有効 (既定) の場合、サーバー管理者は、抽出更新の失敗の際にメール通知を送信するように構成できます。これらのメッセージはサイト レベルで構成されるため、このオプションが有効になっていても、サイトで **[スケジュールされた更新に失敗した場合、データ ソースおよびワークブックの所有者にメールを送信する]** オプションが有効になっていない限り(既定では有効)、メッセージは送信されません。詳細については、抽出更新スケジュールと失敗通知を有効にするを参照してください。

- **ユーザーに登録しているビューに関するメール受信を許可**

  このオプションが有効になっている場合 (既定では無効)、サーバー管理者はサブスクリプション メールを送信するようにサイトを構成できます。これらのメール メッセージはサイト レベルで構成され、このオプションが有効な場合にのみ構成できます。詳細については、サブスクリプションのサーバー設定を参照してください。

  ユーザーがワークブックやビューをサブスクライブすると、ビューのスナップショットがそのユーザーにメールで送信されるため、ユーザーは Tableau Server にサインインせずに最新の更新を確認できます。

  ユーザーがサブスクリプション メールに PDF レンダリングを添付できるようにするには、**[サブスクライブ中のビューへの添付ファイルの追加をユーザーに許可する]** を選択します。

- サーバーの健全性モニタリング

  - **Tableau Server プロセス イベント (実行中、ダウン、フェールオーバー) に関するメールを送信する**

    データエンジン、ファイル ストア、ゲートウェイ、またはリポジトリサーバープロセスが停止または再開するとき、または最初の Tableau Server ノードが停止または再開するときに、Tableau Server がメール メッセージを送信します。

    単一サーバーへのインストールを実行している場合 (同じコンピューター上にすべてのプロセスが存在する場合)、健全性に関するアラートは、Tableau Server の稼働中にのみ送信されます。「DOWN (ダウン)」アラートは送信されません。フェールオーバーのために構成された分散インストールを実行している場合、DOWN (ダウン) アラートはアクティブなリポジトリまたはデータエンジンのインスタンスに障害が発生していることを意味し、後続の UP (アップ) アラートはプロセスのパッシブ インスタンス (リポジトリ) または 2 番目のインスタンス (データエンジン) が引き継いだことを意味します。

注: Tableau Server は、自己修正するように設計されています。サービスまたはプロセスが応答を停止するかダウンした場合、Tableau Server はそれを再起動しようとします。これを完了するまでに 15 ～ 30 分かかる場合があります。そのため、サービスまたはプロセスのアラートにすぐに対応しても、特に再起動中に要求を処理できる冗長サービスを備えたインストールでは、非生産的なアクションになる可能性があります。

- **Tableau Desktop ライセンス レポートの有効化**

  ライセンス レポートは Tableau Desktop で生成され、Tableau Server に送信されます。このオプションが有効になっている場合、Tableau Server はデスクトップ ライセンス レポートの管理レポートを生成して表示します。レポートの詳細については、デスクトップ ライセンスの使用率を参照してください。

- ドライブの空き容量

  Tableau Server のディスク空き容量についての通知 (アラート) を有効にします。

  - **未使用のドライブ空き容量がしきい値を下回っている場合にメールを送信する**

    いずれかのノードでディスク空き容量がしきい値を下回った場合や、継続して下回っている場合にメール通知を送信するように、Tableau Server を設定できます。しきい値の通知を送信する頻度も設定できます。

    [警告しきい値] と **[重要しきい値]** の 2 つを設定する必要があります。しきい値は、残っているディスク空き容量のパーセンテージで表されます。クリティカルしきい値は、警告しきい値よりも小さくなければなりません。

[Send threshold alert every (しきい値アラートを送信する頻度)] オプションも指定します。これにより、警告や緊急の通知を送信する間隔 (分) を決定します。既定値は 60 分です。

- **カスタム管理ビュー用のディスク空き容量の使用率に関する情報としきい値違反を記録する**

  ディスク空き容量の使用率を記録するようTableau Server を設定している場合、ディスク空き容量に関する情報はリポジトリに保存され、管理ビューを使用して使用履歴を表示できます。

4. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。
5. ページ上部の **[変更を保留中]** をクリックします。

6. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

tsm configuration set コマンドを使用し、前述のさまざまな通知の値を個別に設定できます。または、json ファイルを作成し、1 回の操作ですべての構成値を渡すこともできます。どちらの方法も本セクションで説明しています。

個別に通知の値を設定する

以下の表は、本トピックで前述した通知イベントにマッピングするキー/値のペアを示しています。単一のキー/値のペアを設定するには、以下の構文で tsm configuration set コマンドを使用します。

```
tsm configuration set -k <config.key> -v <config_value>
```

たとえば、ジョブ失敗の通知を有効にするには、以下のコマンドを実行します。

```
tsm configuration set -k backgrounder.notifications_enabled -v true
```

通知オプション	キー	値
抽出失敗またはフロー実行失敗	backgrounder.notifications_enabled	true|false
ユーザーのサブスクリプションビューの有効化	subscriptions.enabled	true|false
サブスクリプションの PDF 添付ファイルの有効化	subscriptions.attachments_enabled	true|false
サブスクリプション通知の最大添付ファイル サイズ (MB)	subscriptions.max_attachment_size_megabytes	整数値。既定は 150 です
サーバー ステータスの変化	svcmonitor.notification.smtp.enabled	true|false
ライセンスのレポート	features.DesktopReporting	true|false
残っている容量のしきい値: メール通知の有効化	storage.monitoring.email_enabled	true|false
残っている容量のしきい値: 警告を行うパーセンテージ	storage.monitoring.warning_percent	20 などの整数値
残っている容量のしきい値: クリティカルなパー	storage.monitoring.critical_percent	15 などの整数値

センテージ		
メール間隔の設定	`storage.monitoring.email_interval_min`	25 などの整数値 (分)
使用履歴の記録	`storage.monitoring.record_history_enabled`	`true`|`false`

値を設定したら、以下のコマンドを実行する必要があります。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

単一の json ファイルを使用してすべての通知の値を設定する

単一の構成ですべての通知設定を作成するには、json ファイルを渡します。

次のテンプレートをコピーして編集し、構成用のファイルを作成します。

```
{
  "configKeys": {
    "backgrounder.notifications_enabled": true,
    "subscriptions.enabled": true,
    "subscriptions.attachments_enabled": true,
    "subscriptions.max_attachment_size_megabytes": 150,
    "svcmonitor.notification.smtp.enabled": true,
    "features.DesktopReporting": true,
    "storage.monitoring.email_enabled": true,
    "storage.monitoring.warning_percent": 20,
    "storage.monitoring.critical_percent": 15,
    "storage.monitoring.email_interval_min": 25,
```

```
    "storage.monitoring.record_history_enabled": true
  }
}
```

ファイルを保存したら、それを以下のコマンドを使用して渡します。

```
tsm settings import -f <path-to-file.json>
```

変更を適用するには、次のコマンドを実行します。

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

メンテナンス

バックアップの作成、Active Directory グループの同期、不要なファイルの削除など、Tableau Server の定期的なメンテナンスを実行する必要があります。

バックアップと復元

Tableau Server 管理者は、定期的なデータベースのメンテナンス、サーバーのディスク使用率の監視、不必要なファイルの削除によるサーバーの空き領域増加、および Tableau Server とそれに含まれるデータのバックアップを行う必要があります。これらの手順を実行することで、Tableau Server を最大効率で実行するのに役立ちます。

Tableau データのバックアップおよび復元には、Tableau サービス マネージャー (TSM) コマンド ライン ツールを使用できます。Tableau データにはデータ抽出ファイルと Tableau Server 独自の PostgreSQL データベースが含まれています。このデータベースには、ワークブックおよびユーザーのメタデータ、サーバー構成データが保存されています。Tableau Server のログ ファイルはアクティビティを把握し、問題を診断するのに役立ちます。ログはサーバー上のフォルダーに書き込まれ、ユーザーはそのアーカイブの生成および削除を行ってディスク領域を節約できます。

> **注**: `tsm maintenance restore` コマンドを使用すると、`tabadmin backup` および `tsm maintenance backup` を使用して作成した Tableau Server のバックアップを復元できます。他の方法で作成されたデータベース バックアップや、仮想マシンのスナップショットは、Tableau Server の復元には使用できません。

稼働中のサーバーと同じタイプのアイデンティティ ストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。

プラットフォームの互換性

- Tableau Server on Windows を復元するには、Tableau Server on Linux で作成したバックアップを使用できます。
- Tableau Server on Linux を復元するには、Tableau Server on Windows で作成したバックアップを使用できます。Tableau Server on Linux で Windows のバックアップを復元する方法の詳細については、「Tableau Server の Windows から Linux への移行」を参照してください。
- 異なる Linux オペレーティング システムを実行している Tableau Server on Linux を復元するには、Tableau Server on Linux で作成したバックアップを使用できます。

バックアップと復元のディスク容量使用率

バックアップの作成に必要な空きディスク領域は、Tableau Server リポジトリやファイル ストア サービス内のデータの量、tabadmincontroller サービスとの併置によって異なります。バックアップ中は、古い抽出をクリーンアップするためのバックグラウンド タスクが一時的に一時停止されます。つまり、バックアップの期間中は抽出を更新しても抽出ファイルがそのまま残るため、ディスク領域の使用率が増えます。バックアップに長い時間がかかったり、組織で定期的に更新される抽出が多数使用されていたりする場合は、一時ディスク領域の使用率が大幅に増加します。これらの一時ファイルは、バックアップが完了すると削除されます。

次の表に、ノードがリポジトリ、ファイル ストア、コントローラー、またはそれらを組み合わせてホストするかどうかに基づいた、バックアップに必要なディスク領域を示します。マルチノードの Tableau Server 環境では、各ノードで必要なディスク空き容量を見積もる必要があります。

リポジトリ	ファイル ストア	コントローラー	必要なディスク領域
✓			リポジトリデータx 3 + 250 MB リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。 リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、workgroup.pg_dump ファイルのサイズを使用します。
	✓		ファイル ストア データx 1.5 ファイル ストア データ(抽出、フローなど)の推量を取得するには、`<data directory>/dataengine` ディレクトリのサイズを確認します。

		✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 2.5
✓	✓		リポジトリデータx 3 + 250 MB + ファイル ストア データx 1.5
	✓	✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 1.5
✓		✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 2.5
✓	✓	✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 1.5

復元のディスク空き容量の要件

データベースの復元プロセスを正常に実行するには、十分なディスク空き容量が必要です。

Tableau Server を復元するには:

- コントローラー ノードでは、バックアップ アーカイブのサイズ以上の空き領域が必要です。
- リポジトリ ノードでは、バックアップ アーカイブ内にリポジトリデータのサイズの少なくとも 3 倍 + 250 MB + pgsql データディレクトリに相当する空き領域が必要です。
- ファイル ストア ノードでは、バックアップ アーカイブ内にある dataengine フォルダーのサイズの少なくとも 2 倍の空き領域が必要です。

Tableau Server のバックアップに関するベストプラクティス

これらのセキュリティとパフォーマンスのベストプラクティスに従うことをお勧めします。

バックアップ ファイルの保護

構成シークレットはディスクに内部保存するときに暗号化されますが、構成をバックアップ ファイルにエクスポートするときは、一部のシークレットはプレーン テキスト形式でファイルに書き込まれます。

管理者はこのファイルを保護するための対策を講じる責任があります。次のようなさまざまなオプションがあります。

- 暗号化されたファイル システムにファイルを書き込む。
- 物理的に保護されていて、特定のユーザーに制限されているディスクにファイルを書き込む。
- バックアップ ファイルを暗号化する。

バックアップ効率を最大化する。

バックアップの効率を最大限に高めるには、いくつかの方法があります。それぞれの効果性は環境によって影響を受けるので、実際のデータを使用してテストを行い、どれが最適かを確かめてください。

トポロジの構成を使用した最適化:

- 管理コントローラーと同じノード上にファイル ストアを配置すると、バックアップ プロセス中にノード間でデータを転送する必要性が減少するか排除されるため、Tableau Server のバックアップにかかる時間を短縮できます。これは、組織で多数の抽出を使用している場合に特に当てはまります。
- リポジトリ(pgsql) を管理コントローラー ノードと同じ場所に配置すると、バックアップ時間は短縮できますが、ファイル ストアの場合ほど時間の節約は顕著ではありません。

最初のノードでエラーがあった場合や、コントローラーを別のノードに移動した場合を除き、管理コントローラーは通常最初のノード上にあります。

バックアップ戦略による最適化:

バックアップは、リソースを大量に消費するプロセスです。可能であれば、バックアップをオフピーク時に実行することは、一般的に好ましい戦略であると考えられています。ただし、良い戦略になるかどうかは、要件、Tableau Server データの更新頻度、および復元の要件によって変わります。バックアップと災害復旧の詳細については、「Tableau Server のディザスタリカバリ」を参照してください。ここでは、いくつかのバックアップ戦略を紹介し、要件に合わせて導入します。

- **ストレージのタイプ:** ソリッド ステート ディスクは一般的にバックアップに推奨されています。SSD では、従来のスピニング ディスクと比較して、バックアップがより迅速に行われるため、完了も早くなります。

- **バックアップの圧縮:** 圧縮を使用してバックアップを実行するオプションと、圧縮を使用せずにバックアップを実行するオプションがあります。圧縮を使用してバックアップを実行すると、バックアップサイズは比較的小さくなりますが、パフォーマンスが低下する場合があります。そのため、速度に重点を置くことが目的の場合は、`--skip-compression` オプションを選択します。

  Tableau Server をバックアップする場合は、`--skip-compression` オプションを使用してください。これにより、圧縮を使用せずにバックアップが作成されるため、バックアップファイルは大きくなりますが、バックアップが完了するまでの時間を短縮できます。詳細については、「tsm maintenance backup」を参照してください。

- **スナップショット バックアップ:** このオプションは、外部ファイル ストアを使用して Tableau Server を構成する場合にのみ使用できます。スナップショット バックアップのパフォーマンスはネットワーク接続ストレージのタイプによって異なりますが、一般的なスナップショット バックアップは従来の Tableau Server バックアップよりも高速です。詳細については、「Tableau Server 外部ファイル ストア」を参照してください。

Tableau Server の完全なバックアップと復元を実行します

以下のステップを使用して Tableau Server の展開をバックアップできます。特に、以下のステップは、バックアップデータおよびアセットの集合からのサーバーのクローンの復元方法について説明します。

> **注:** バックアッププロセスの実行には長時間かかります。バックアップ実行中は他のジョブは実行できないため、バックアップは業務時間以外に実行することを推奨します。

バックアップデータ型

Tableau Server が生成可能なバックアップデータは、2 種類あります。復旧シナリオでサーバーを復元しなければならない場合のために、定期的に各型のバックアップを取ることを推奨します。

- **Tableau Server で管理するデータ:** Tableau PostgreSQL データベースまたはリポジトリのほか、ワークブックとユーザーのメタデータ、データ抽出ファイル、サイト構成データが含まれるファイル ストアで構成されています。TSM を使用してバックアップを作成するときは、このデータすべてが .tsbak 拡張子を持つ 1 つのファイルに保存されます。このデータは `tsm maintenance backup` コマンドでバックアップされます。

  注: 外部ファイル ストアが構成されている場合、 `tsm maintenance backup` コマンドを使用して、Tableau Server データをバックアップすることはできません。このデータをバックアップする方法の詳細については、「外部ファイル ストアによるバックアップと復元」を参照してください。

  - 稼働中のサーバーと同じタイプのアイデンティティ ストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。
  - バックアップ ファイルは、バックアップが作成されたバージョンと同じか新しいバージョンの Tableau Server バージョンにのみ復元できます。Tableau の古いバージョンに復元することはできません。
  - **重要:** ブルー/グリーン アップグレードを実行する場合、または tsm メンテナンス (バックアップと復元) メソッドを使用して Tableau Server 2021.4 (またはそれ以前) を手動でアップグレードする場合は、Tableau Server 2022.1 (またはそれ以降) に復元する前に `legacy-identity-mode` を有効にする必要があります。詳細については、「アイデンティティの移行に関する問題をトラブルシューティングする」を参照してください。
  - バージョン 2022.3 以降、tabadmin を使用して作成したバックアップ (「以前の TSM バックアップ」) はサポートされていません。TSM 以前のバックアップを Tableau Server バージョン 2022.3 以降に復元することはできません。

- 構成およびトポロジ データ: サーバーを完全に復旧するために必要なサーバー構成情報の殆どが含まれます。SMTP、アラート、一部の認証アセットはすべて、バックアップのためにエ

クスポートできる構成データの例です。トポロジデータは単一サーバーおよび複数ノード両方の展開で Tableau Server プロセスを構成する方法を定義します。構成およびトポロジデータは `tsm settings export` コマンドでバックアップされます。

> **注:** `tsm maintenance backup` コマンドが使用しているファイルパスを、既定値から変更できます。詳細については、tsm ファイルパスを参照してください。

手動プロセスが必要なバックアップアセット

一部の構成データは、`tsm settings export` コマンドに含まれておらず、手動で文書化し復元する必要があります。次の構成データは、`tsm settings export` オペレーションから除外されます。バックアップメンテナンス手順には、次の Tableau Server 構成データの文書化を含む必要があります。

- システムユーザーアカウント。Tableau Server セットアップは、権限なしユーザーアカウント、`tableau` を作成します。このアカウントは Tableau Server リソースにアクセスするために使用されます。このアカウントはセットアップ中に変更可能です。このアカウントを変更していなければ、これを文書化する必要はありません。

- TSM グループメンバーシップ。Tableau Server で作成される 2 つのグループ: `tableau` および `tsmadmin` があります。Tableau Server をインストールするときに代替グループを構成する場合は、グループ名を文書化する必要があります。

  いずれの場合でも、これらのグループ内のユーザーアカウントを文書化する必要があります。グループのメンバーシップを表示するには、次のコマンド `grep <group_name> /etc/group` を実行します。

- 調整サービス展開構成。マルチモードクラスターを実行している場合は、どのノードが調整サービスプロセスを実行しているかを文書化します。ノードのプロセス構成を表示するには、`tsm topology list-nodes -v` を実行します。

- カスタマイズ設定。組織が、Tableau Server ウェブページ用のカスタムヘッダーまたはサインインロゴを使用している場合、バックアップポートフォリオとともにそれらのアセットのコピーを含

める必要があります。tsm customizeを参照してください。

- ほとんどの認証アセット。ファイルの場所はエクスポートされた `settings.json` ファイルに含まれる場合がありますが、ほとんどの証明書ファイル、キーファイル、キータブファイル、またはその他の認証関連のアセットは、TSM によってバックアップされません。移動しようとしているこれらのアセットは、もう一度作成する必要がないことを確認してください。

  例外が 3 つあります。

  - 内部 PostgreSQL データベースの公開証明書と秘密キー(有効な場合) がバックアップされます。
  - 外部 SSL の証明書とキーはバックアップされ、構成データに含まれます。
  - tsm security custom-cert addによってインストールされたカスタム証明書 (追加された場合) がバックアップされます。

  ただし、その他すべての認証関連アセットはバックアップされません。たとえば、PostgreSQL データベースへのアクセスを tsm data-access repository-access enable コマンドで有効にした場合、構成した各アカウントの名前/パスワードのペアを文書化するようにしてください。これらの認証資格情報はバックアップされません。相互 SSL の証明書とキーは、バックアップには含められません。

- LDAP アセット。キータブファイル、構成ファイル、またはその他の LDAP 関連アセットは、TSM でバックアップされません。

内部サーバーシークレットおよびリポジトリパスワードは、エクスポートされない暗号関連の構成です。しかし、構成値を文書化する必要はありません。新規インスタンスを初期化するときに、新しいシークレットが、復元プロセスの一部として作成されます。

復旧用 Tableau Server のバックアップ

Tableau Server には、Tableau Server 用のバックアップデータを生成するために実行するコマンドが含まれます。

> **注:** Tableau Server を Linux でバックアップしている場合、権限なしユーザーは、バックアップファイルが書き込まれるネットワーク共有への書き込みアクセスが必要です。アクセス権がない場合、バックアップは失敗します。

サーバートポロジおよび構成データをバックアップするには、tsm settings コマンドを使用します。

1. トポロジおよび構成データは、`tsm settings export` コマンドを実行するときに含まれます。結果は JSON ファイルとしてエクスポートされます。次のコマンドを実行して、JSON ファイルの名前と場所を指定します。

   ```
   tsm settings export -f <filename>.json
   ```

   > **注:** バックアップにはシークレットが含まれているので、バックアップを暗号化し安全な場所に保存することを推奨します。Tableau Server の機密の詳細については、「Server シークレットの管理」を参照してください。

2. リポジトリとファイル ストアのデータをバックアップします。リポジトリデータは `tsm maintenance backup` コマンドでバックアップされます。次のコマンドを実行して、バックアップ ファイルの名前と場所を指定します。

   ```
   tsm maintenance backup -f <filename>.tsbak -d
   ```

   バックアップ ファイルは、データディレクトリの一時的な場所に集められ、TSM `basefilepath.backuprestore` 変数に定義したディレクトリに書き込まれます

   ```
   /var/opt/tableau/tableau_
   server/data/tabsvc/files/backups/<filename>.tsbak
   ```

   バックアップ ファイルの書き込み先とその場所の変更方法の詳細については、tsm ファイル パスを参照してください。**注:** バックアップの場所を変更する場合でも、バックアップ プロセスではデータディレクトリの一時的な場所を使用してバックアップ ファイルを集めます。

注: ファイル ストアが Tableau Server の外部で構成されている場合、tsm maintenance backup コマンドを使用して Tableau Server データをバックアップすることはできません。このデータをバックアップする方法の詳細については、外部ファイル ストアによるバックアップと復元を参照してください。

コア Tableau Server 機能の復元

次の手順は、復旧シナリオで Tableau Server を再構築するために、前の 2 つのセクションのアセットを使用します。

注: 問題がなければ機能していた Tableau Server でリポジトリのみを復元したい場合は、バックアップからの復元を参照してください。分散型の展開を実行していて、最初のノードが故障した場合は、初期ノードの障害からの回復を参照してください。

Tableau Server on Linux からのトポロジおよび構成バックアップ データを使用する必要があります。Tableau Server on Windows で生成したバックアップ ファイルから構成データを復元することはできません。Tableau Server on Windows で作成したバックアップを Tableau Server on Linux で復元するには、Tableau Server の Windows から Linux への移行を参照してください。

次のアセットを準備する必要があります。

- トポロジおよび構成データ: これは、`tsm settings export` コマンドで生成される JSON ファイルです。
- リポジトリ バックアップ ファイル: これは、`tsm maintenance backup` コマンドで生成される .tsbak 拡張子を持つファイルです。

  稼働中のサーバーと同じタイプのアイデンティティ ストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。

`tsm maintenance restore` を使用して Tableau データを復元すると、データ抽出ファイルおよび PostgreSQL データベースの内容がバックアップ ファイル (`.tsbak`) の内容に上書きされます。Tableau Server の分散インストールを実行している場合は、TSM コントローラーを実行しているノード(通常は初期ノード) で復元を実行します。

- バックアップ アセット: 前のセクションで示したように、これらのアセットには、文書化された構成のリストが含まれます。

スタンドアロン Tableau Server の復元

1. Tableau Server を復元したいパーソナルコンピューターで、TSM のインストールと初期化を実行します。組織が既定以外のシステム ユーザー アカウントを使用している場合、このトピックの前のセクションで説明したとおり、このステップデーユーザーを指定する必要があります。
2. Tableau Server のライセンス認証と登録。
3. (オプション)。ローカル ファイアウォールの構成。
4. (オプション)。LDAP を確認します。
5. Tableau Server を初期化します。初期ノード設定の構成を参照してください。
6. トポロジと構成データのインポート。トポロジと構成 JSON バックアップ ファイルを、コンピュータにコピーします。次のコマンドを実行して、JSON ファイルをインポートします。

   ```
   tsm settings import -f <filename>.json
   ```

7. (オプション)。保留中の変更を適用します。コマンドプロンプトで、次のコマンドを実行します。

   ```
   tsm pending-changes apply
   ```

8. Tableau Server を再起動します。コマンドプロンプトで、次のコマンドを実行します。

```
tsm restart
```

9. リポジトリデータの復元バックアップからの復元を参照してください。

10. (オプション)。TSM グループ メンバーシップを再読み込みします。次のコマンドでユーザーとグループを追加します。

    ```
    sudo usermod -G <group_name> -a <username>
    ```

Tableau Server クラスタを復元するには

1. 初期ノードで、TSM のインストールと初期化。組織が既定以外のシステム ユーザー アカウントを使用している場合、このトピックの前のセクションで説明したとおり、このステップデーユーザーを指定する必要があります。

2. 初期ノードで、Tableau Server のライセンス認証と登録。

3. (オプション)。初期ノードで、ローカル ファイアウォールの構成。

4. 初期ノードで、LDAP の検証 (オプション) を行い、Tableau Server を初期化します。初期ノード設定の構成を参照してください。

5. 初期ノードで、`tsm topology nodes get-bootstrap-file --file <path\file>.json` を実行します。

6. クラスタにあるすべての追加ノードに bootstrap .json ファイルをコピーします。

7. クラスタ内の各ノードで以下を実行します。

    a. Tableau Server パッケージをインストールします。

    b. scripts ディレクトリに移動します。

    c. 最初のノードと追加のノード間の通信を開始します。

```
sudo ./initialize-tsm -b <path-to-bootstrap>.json -u
<admin-user-on-first-node> --accepteula
```

8. 初期ノードで、`tsm topology list-nodes -v` を実行し、ノード名がエクスポートしたトポロジ設定から変わっていないことを確認します。*ノード名が変わっている場合は、トポロジ設定を新しい名前で手動で更新するか、またはプロセスを手動で構成します。*

9. クラスタコントローラープロセスはすべてのノードで必要であり、明示的に追加する必要があります。最初のノードから、クラスタコントローラーのインスタンスを追加の各ノードに追加します。`<nodeID>` は、各追加ノードの ID です。プロセスを各ノードに個別に追加します。この例では、クラスタコントローラーをノード 2 と 3 に追加しています。

   ```
   tsm topology set-process -n node2 -pr clustercontroller -c 1
   tsm topology set-process -n node3 -pr clustercontroller -c 1
   tsm pending-changes apply
   ```

10. 最初のノードから、調整サービス アンサンブルの展開。アンサンブル構成は、以前の構成と一致する必要があります。

11. 最初のノードで、トポロジと構成データをインポートします。トポロジと構成 JSON バックアップファイルを、コンピュータにコピーします。次のコマンドを実行して、JSON ファイルをインポートします。

    ```
    tsm settings import -f <filename>.json
    ```

12. 初期ノードで、保留中の変更を適用します。コマンドプロンプトで、次のコマンドを実行します。

    ```
    tsm pending-changes apply
    ```

13. 初期ノードで、Tableau Server を再起動します。コマンドプロンプトで、次のコマンドを実行します。

    ```
    tsm restart
    ```

14. 初期ノードで、リポジトリデータの復元。バックアップからの復元を参照してください。

15. 初期ノードで、TSM グループ メンバーシップを再読み込みします。次のコマンドでユーザーとグループを追加します。

    ```
    sudo usermod -G <group_name> -a <username>
    ```

その他の機能の復元

以前サーバーが次の機能で構成されていた場合は、復元したサーバーでそれらを再び有効にし構成する必要があります。

- 認証ソリューション: OpenID、外部 SSL、信頼できる認証。認証を参照してください。
- サイトのカスタマイズ: tsm customize を参照してください。
- PostgreSQL リポジトリへのアクセスを有効にするには、tsm data-access repository-access enableを参照してください。

復元後の抽出の再暗号化

保存中の抽出の暗号化機能を使用している場合は、バックアップの復元後に異なる暗号化キーを使用して抽出を再暗号化することもできます。保存中の抽出の暗号化を参照してください。

`tabcmd reencryptextracts <site-name>` を実行して、特定のサイトで抽出を再暗号化します。詳細については、reencryptextractsを参照してください。暗号化された抽出を保存するすべてのサイトでこのコマンドを実行してください。サイト上の暗号化された抽出の数によっては、この操作によってサーバーの処理負荷が大幅に大きくなる可能性もあります。業務時間外にこの操作を実行することを検討してください。

Tableau Server データのバックアップ

サーバーの管理とメンテナンスを適切に行うには、Tableau Server を定期的にバックアップすることが重要です。`tsm maintenance restore` コマンドを使用すると、`tsm maintenance backup` コマンドによって作成された.Tableau Server バックアップを復元できます。他の方法で作成されたデータベース バックアップや、仮想マシンのスナップショットは、Tableau Server の復元には使用できません。そのため、最新のバックアップを持っていることが重要です。

稼働中のサーバーと同じタイプのアイデンティティストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。

> Tableau Server on Windows については、「Tableau Server データのバックアップ」を参照してください。

Tableau Server データにはデータ抽出ファイルと Tableau PostgreSQL データベースが含まれています。このデータベースには、ワークブックおよびユーザーのメタデータ、構成データが保存されています。TSM を使用してバックアップを作成するときは、このデータすべてが .tsbak 拡張子を持つ 1 つのファイルに保存されます。分散インストールを実行している場合、すべてのノードの Tableau Server データがバックアップされます。

サーバーのバックアップ頻度は、サーバーの使用量や、コンテンツおよびユーザーの変更頻度を含め、環境によって異なります。システムに障害が発生した場合、バックアップより後に行った変更や更新は失われ、Tableau Server を復元する必要があります。アクティビティが多ければ、サーバーのバックアップ頻度を増やす必要があります。

定期的なバックアップに加えて、Tableau Server の新しいバージョンにアップグレードする前に必ずバックアップを作成するようにしてください。アップグレードプロセスでは、PostGRES のバージョンが更新されている場合を除き、バックアップは作成されません。アップグレードプロセスでは、内部で使用される PostGRES のみのバックアップが作成されます。

データの損失を防ぐため、バックアップを作成した後、.tsbak ファイルを Tableau Server インストール環境に含まれない PC に保存してください。

バックアップのディスク容量使用率

バックアップの作成に必要な空きディスク領域は、Tableau Server リポジトリやファイル ストア サービス内のデータの量、tabadmincontroller サービスとの併置によって異なります。バックアップ中は、古

い抽出をクリーンアップするためのバックグラウンドタスクが一時的に一時停止されます。つまり、バックアップの期間中は抽出を更新しても抽出ファイルがそのまま残るため、ディスク領域の使用率が増えます。バックアップに長い時間がかったり、組織で定期的に更新される抽出が多数使用されていたりする場合は、一時ディスク領域の使用率が大幅に増加します。これらの一時ファイルは、バックアップが完了すると削除されます。

次の表に、ノードがリポジトリ、ファイル ストア、コントローラー、またはそれらを組み合わせてホストするかどうかに基づいた、バックアップに必要なディスク領域を示します。マルチノードの Tableau Server 環境では、各ノードで必要なディスク空き容量を見積もる必要があります。

リポジトリ	ファイル ストア	コントローラー	必要なディスク領域
✓			リポジトリデータx 3 + 250 MB リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。 リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、workgroup.pg_dump ファイルのサイズを使用します。
	✓		ファイル ストア データx 1.5 ファイル ストア データ(抽出、フローなど)の推量を取得するには、`<data directory>/dataengine` ディレクトリのサイズを確認します。
		✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 2.5
✓	✓		リポジトリデータx 3 + 250 MB + ファイル ストア データx 1.5

	✓	✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 1.5
✓		✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 2.5
✓	✓	✓	リポジトリデータx 3 + 250 MB + ファイル ストア データx 1.5

Tableau Server バックアップの最適化

バックアップの効率を最大限に高めるには、いくつかの方法があります。それぞれの効果性は環境によって影響を受けるので、実際のデータを使用してテストを行い、どれが最適かを確かめてください。

トポロジの構成を使用した最適化:

- 管理コントローラーと同じノード上にファイル ストアを配置すると、バックアップ プロセス中にノード間でデータを転送する必要性が減少するか排除されるため、Tableau Server のバックアップにかかる時間を短縮できます。これは、組織で多数の抽出を使用している場合に特に当てはまります。
- リポジトリ(pgsql) を管理コントローラー ノードと同じ場所に配置すると、バックアップ時間は短縮できますが、ファイル ストアの場合ほど時間の節約は顕著ではありません。

最初のノードでエラーがあった場合や、コントローラーを別のノードに移動した場合を除き、管理コントローラーは通常最初のノード上にあります。

バックアップ戦略による最適化:

バックアップは、リソースを大量に消費するプロセスです。可能であれば、バックアップをオフピーク時に実行することは、一般的に好ましい戦略であると考えられています。ただし、良い戦略になるかどうかは、要件、Tableau Server データの更新頻度、および復元の要件によって変わります。バックアップと災害復旧の詳細については、「Tableau Server のディザスタ リカバリ」を参照してください。ここでは、いくつかのバックアップ戦略を紹介し、要件に合わせて導入します。

- **ストレージのタイプ:** ソリッド ステート ディスクは一般的にバックアップに推奨されています。SSD では、従来のスピニング ディスクと比較して、バックアップがより迅速に行われるため、完

了も早くなります。

- **バックアップの圧縮:** 圧縮を使用してバックアップを実行するオプションと、圧縮を使用せずにバックアップを実行するオプションがあります。圧縮を使用してバックアップを実行すると、バックアップサイズは比較的小さくなりますが、パフォーマンスが低下する場合があります。そのため、速度に重点を置くことが目的の場合は、`--skip-compression` オプションを選択します。

  Tableau Server をバックアップする場合は、`--skip-compression` オプションを使用してください。これにより、圧縮を使用せずにバックアップが作成されるため、バックアップファイルは大きくなりますが、バックアップが完了するまでの時間を短縮できます。詳細については、「tsm maintenance backup」を参照してください。

- **スナップショット バックアップ:** このオプションは、外部ファイル ストアを使用して Tableau Server を構成する場合にのみ使用できます。スナップショット バックアップのパフォーマンスはネットワーク接続ストレージのタイプによって異なりますが、一般的なスナップショット バックアップは従来の Tableau Server バックアップよりも高速です。詳細については、「Tableau Server 外部ファイル ストア」を参照してください。

TSM コマンド ライン インターフェース (CLI) を使用したバックアップの作成

`tsm maintenance backup` コマンドを使用して Tableau Server に管理されているデータのバックアップを作成します。このデータにはデータ抽出ファイルと Tableau PostgreSQL データベースが含まれています。このデータベースには、ワークブックおよびユーザーのメタデータが保存されています。

重要: Tableau サポートからの指示がない限り、バックアップを生成するときに `pg-only` オプションを使用しないでください。このオプションはリポジトリのみをバックアップし、Server の復元には使用 *できません*。主な用途はトラブルシューティングであり、必要があれば Tableau サポートから `--pg-only` バックアップを作成するように依頼されます。

> **注:** Tableau Server を Linux でネットワーク場所にバックアップしている場合、権限なしユーザーはバックアップファイルが書き込まれるネットワーク共有への書き込みアクセスが必要です。そうでないと、バックアップは失敗します。

サーバー構成データをバックアップするには、tsm settings コマンドを使用します。`tsm maintenance backup` コマンドを使用すると、バックアップファイルに現在の日付が追加されます。

```
tsm maintenance backup -f <backup_file> -d
```

詳細については、tsm maintenance backupを参照してください。

アップグレード前のバックアップの作成

Tableau Server をアップグレードする前に、必ずバックアップを作成することをお勧めします。Tableau Server サーバーの実行中にバックアップを作成し、アップグレード中にサーバーを利用できない時間を最小限に抑えることができます。アップグレード前のバックアップの作成プロセスは、定期的なバックアップの作成と同じですが、分散型インストールに対する追加の考慮事項が 1 つあります。

> **注:** 新しいインストールに含めないノードからTableau Server をアンインストールして、以前のノードと新しいインストールとの間の競合を避けるようにしてください。

アップグレード中のバックアップ

Tableau Server のアップグレード中に、必要に応じて、アップグレードの一部として発生する移行を可能にするために、データベースの一時的なバックアップが作成される場合がありますバックアップの作成はアップグレード中に行われ、ほとんどの場合、アップグレードプロセスに顕著な影響はありません。特定の特殊なケースでは、付加的な影響が発生する可能性があります。

- バージョン 2021.4 (またはそれ以前) からTableau Server 2022.1 (またはそれ以降) へのアップグレード—ブルー/グリーン アップグレードを実行する場合、または tsm メンテナンス (バック

アップと復元) メソッドを使用して Tableau Server 2021.4 (またはそれ以前) を手動でアップグレードする場合は、Tableau Server 2022.1 (またはそれ以降) に復元する前に `legacy-identity-mode` を有効にする必要があります。詳細については、「アイデンティティの移行に関する問題をトラブルシューティングする」を参照してください。

- メジャーバージョンを pgsql で更新 - アップグレードに Tableau リポジトリで使用されるデータベースのメジャーバージョン更新が含まれている場合、時間を節約するために、内部アップグレードのバックアップは圧縮なしで実行されます。この場合、アップグレードプロセス中に追加の一時ディスク領域が必要になります。

  メジャーバージョンデータベースの更新を含む Tableau Server バージョン: 2020.4。

バックアップのスケジュール設定と管理

2020.4.0 以降では、tsm コマンドを使用してバックアップをスケジュールできます。これは、コマンドラインから実行する必要があります (バックアップをスケジュールする TSM UI はありません)。`tsm maintenance backup` コマンドを使用すると、バックアップスケジュールを作成および更新できます。`tsm schedules` コマンドを使用すると、スケジュールの表示、削除、一時停止、再開、および更新を行うことができます。

バックアップをスケジュールするには、次の手順に従います。

1. 最初のノード (TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

   ```
   tsm maintenance backup -f <backup-file> -sr <recurrence> -st
   <time-to-run> -sd <days-to-run> -sn <schedule-name>
   ```

   たとえば、毎月 15 日の午前 2 時に実行され、`<yyyy.mm.dd.hh.mm>-ts-mid_month_backup.tsbak` ファイルを生成する "monthly-backup" という名前のバックアップスケジュールを作成するには、次のコマンドを実行します。

   ```
   tsm maintenance backup -f ts-mid_month_backup -sr monthly -st
   02:00 -sd 15 -sn monthly-backup
   ```

スケジュールされたバックアップを表示するには、次の手順に従います。

1. 最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

   ```
   tsm schedules list
   ```

   スケジュールは、スケジュールされた実行時間で昇順に並べ替えるか、--next-run または --schedule-name オプションを使用して名前で並べ替えることができます。--schedule-id オプションを使用して、単一のスケジュールの詳細を表示することもできます。単一のスケジュールを表示すると、作成された日時、実行された回数、実行時に使用される特定のオプションなど、そのスケジュールに関する追加の詳細が表示されます。ジョブオプションは JSON 形式で "Job args" として表示されます。

スケジュールされたバックアップを更新するには、次の手順に従います。

1. 最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

   ```
   tsm schedules update --schedule-id <ID> --schedule-time <time-to-run> --schedule-recurrence <frequency> --schedule-days <day-to-run>
   ```

   注: 名前を追加または変更するには、tsm-maintenance-backup コマンドを使用します。

バックアップスケジュールを中断または再開するには、次の手順に従います。

1. 最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のいずれかのコマンドを実行します。

- スケジュールを中断するには、次のコマンドを実行します。

  ```
  tsm schedules suspend --schedule-id <scheduleID>
  ```

- 中断されたスケジュールを再開するには、次のコマンドを実行します。

  ```
  tsm schedules resume --schedule-id <scheduleID>
  ```

バックアップ プロセスのスクリプト作成

頻繁にバックアップを行う場合は、バックアップおよび関連タスクを実行するスクリプトを作成できます。これらのタスクには次が含まれます。

- バックアップの実行前にファイルやフォルダーをクリーンアップします。
- バックアップ自体を実行します。
- 安全に保管するため、バックアップ ファイルを別のコンピューターにコピーします。

このセクションでは、一緒に使用することでバックアップおよび関連タスクを実行できる tsm コマンドについて説明します。

ログ ファイルを削除して一時フォルダーをクリアする

古い Tableau Server ログ ファイルと一時ファイルを削除してバックアップの作成にかかる時間を短縮し、バックアップ ファイルをできるだけ小さくなるようにします。

数日以上経過したファイルを削除するには、次のコマンドを実行します。

```
tsm maintenance cleanup
```

バックアップの実行

> **注:** Tableau Server を Linux でネットワーク場所にバックアップしている場合、権限なしユーザーはバックアップファイルが書き込まれるネットワーク共有への書き込みアクセスが必要です。そうでないと、バックアップは失敗します。

バックアップを作成するには、tsm maintenance backup コマンドを使用します。

```
tsm maintenance backup --file <backup_file> --append-date
```

コマンドに関して、次に注意してください。

- コマンドに `--append-date` を追加すると、バックアップファイル名に日付が含まれます。
- バックアップファイルは、データディレクトリの一時的な場所に集められ、TSM `basefilepath.backuprestore` 変数に定義したディレクトリに書き込まれます

  ```
  /var/opt/tableau/tableau_
  server/data/tabsvc/files/backups/<filename>.tsbak
  ```

  バックアップファイルの書き込み先とその場所の変更方法の詳細については、tsm ファイルパスを参照してください。**注:** バックアップの場所を変更する場合でも、バックアッププロセスではデータディレクトリの一時的な場所を使用してバックアップファイルを集めます。

バックアップファイルを別のコンピューターにコピーする

ベストプラクティスとして、バックアップの作成後、Tableau Server とは離れた別の場所に、バックアップファイルをコピーします。

バックアップからの復元

`tsm maintenance restore` コマンドを使用して Tableau Server データを復元します。Tableau Server の以前のバージョンに戻す場合 (アップグレードで問題が発生した場合など) や、Tableau Server を新しいハードウェアに移行する場合など、システム障害が発生し、データの復元が必要となった場合にはこれを実行することもできます。`tsm maintenance restore` コマンドを使用し、`tabadmin backup` および `tsm maintenance backup` を使用して作成した Tableau Server のバックアップを復元できます。

Tableau Server を復元する際の制限事項

- ブルー/グリーンアップグレードを実行する場合、または tsm メンテナンス (バックアップと復元) メソッドを使用して Tableau Server 2021.4 (またはそれ以前) を手動でアップグレードする場合は、Tableau Server 2022.1 (またはそれ以降) に復元する前に `legacy-identity-mode` を有効にする必要があります。詳細については、「アイデンティティの移行に関する問

題をトラブルシューティングする」を参照してください。

- 他の方法で作成されたデータベースバックアップや、仮想マシンのスナップショットは、Tableau Server の復元には使用できません。
- `tsm maintenance restore` を使用して Tableau データを復元すると、データ抽出ファイルおよび PostgreSQL データベースの内容がバックアップファイル (`.tsbak`) の内容に上書きされます。Tableau Server の分散インストールを実行している場合は、TSM コントローラーを実行しているノード(通常は初期ノード) で復元を実行します。
- 稼働中のサーバーと同じタイプのアイデンティティストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。
- バックアップファイルは、バックアップが作成されたバージョンと同じか新しいバージョンの Tableau Server バージョンにのみ復元できます。Tableau の古いバージョンに復元することはできません。
- バージョン 2022.3 以降、tabadmin を使用して作成したバックアップ (「以前の TSM バックアップ」) はサポートされていません。TSM 以前のバックアップを Tableau Server バージョン 2022.3 以降に復元することはできません。
- 復元プロセスでは、Tableau Server が管理するコンテンツと外部アセットを完全に再インデックスする処理が開始されます。このプロセスは CPU リソースを消費し、バックアップと復元中にその消費が顕著になる可能性があります。

バックアップファイルから Tableau Server を復元する

注: この操作には、TSM コマンドラインを使用して実行する必要のある手順が含まれています。

1. (オプション) .tsbak ファイルを既定のファイルのファイルの場所にコピーします。

   restore コマンドは、TSM の basefilepath.backuprestore 変数で定義したディレクトリにバックアップ ファイルがあることを前提としています。既定のディレクトリは次のとおりです。

   ```
   /var/opt/tableau/tableau_server/data/tabsvc/files/backups/
   ```

   ファイル パスとその変更方法の詳細については、tsm ファイル パスを参照してください。

   > **注:** バックアップ フォルダーにコピーされたバックアップを復元している場合は、権限なしユーザー (既定: tableau) に少なくともバックアップ ファイルの読み取りアクセスがあることを確認してください。そうしないと、復元プロセスでバックアップ ファイルを解凍できない場合があり、復元に失敗します。

2. サーバーを停止します。コマンド プロンプトで、次のように入力します。

   ```
   tsm stop
   ```

3. バックアップ ファイルから復元します。コマンド プロンプトで、次のように入力します。

   ```
   tsm maintenance restore --file <file_name>
   ```

   上記の行では、<file_name> を、復元するバックアップ ファイルの名前に置き換えます。

   > **注:** バックアップからの復元試行中にエラーが発生する場合は、Linux での Tableau Server のトラブルシューティングを参照してください。

4. サーバーを再起動します。

   ```
   tsm start
   ```

サーバーのメンテナンス

管理者は、サーバー ステータスのチェック、サーバー上のアクティビティの分析と監視、スケジュールされているタスクの管理、特定のメンテナンス アクティビティ(保存されているデータ接続パスワードのクリアなど) を行います。また、サーバーでのユーザー体験をカスタマイズするための設定もいくつかあります。これらのタスクの一部は [ステータス] ページの [全般] ページから、それ以外は [設定] ページから実行できます。

サーバープロセス ステータスの表示

サーバープロセス ステータスを表示するには、TSM CLI コマンドを実行するか、Tableau Server の TSM Web UI または管理者ページにアクセスします。

TSM CLI を使用したプロセス ステータスの表示

次のコマンドを実行します。

```
tsm status -v
```

このコマンドにより、インスタンスで構成されているすべてのプロセスとそれに対応するステータスが出力されます。

Web UI でのプロセス ステータスの表示

Tableau Server または Tableau サービス マネージャー (TSM) には、管理者が Tableau プロセスの状態を確認可能な場所が 2 つあります。アカウントとサーバーの設定方法に応じ、これらの場所の 1 つにアクセスできる場合と両方にアクセスできる場合があります。表示されるプロセス ステータスの情報の多くは、両方のステータス ページで複製されます。このセクションでは各ページについて説明し、各ページで固有の内容を記載しています。

注 Web UI の両方の場所に表示されるステータス ページには、特定のノードで構成されている合計プロセスの一部が表示されます。すべてのプロセスを表示するには、TSM CLI コマンドである `tsm status -v` を実行する必要があります。

- Tableau サービス マネージャー (TSM) のステータス ページには TSM でアクセスでき、TSM 管理者が表示可能です。このページを表示するには、TSM にログインできる必要があります。TSM へのサインインの詳細については、Tableau サービス マネージャーの Web UI へのサインインを参照してください。

- Tableau Server ステータス ページは、Tableau Server の Web UI に表示され、Tableau Server 管理者がアクセス可能です。このページには Tableau Server プロセスのほか、プロセスが想定通りに実行されない場合のトラブルシューティング ドキュメントへのリンクが含まれます。プロセスのステータス インジケーターにカーソルを合わせると、ツールヒントでプロセスが実行中のノード名とポートが表示されます。Tableau Server ステータス ページには TSM プロセスが表示されません。管理者としての Tableau Server へのサインインの詳細については、Tableau Server 管理者エリアへのサインインを参照してください。

Tableau サービス マネージャー (TSM) のステータス ページ

TSM のステータス ページには、TSM コントローラーおよびライセンス サーバーを含め、サーバープロセスの状態が表示されます。これらの 2 つのプロセスは Tableau Server のステータス ページに表示されません。

可能性があるステータス インジケーターが、表の下部に一覧で表示されています。

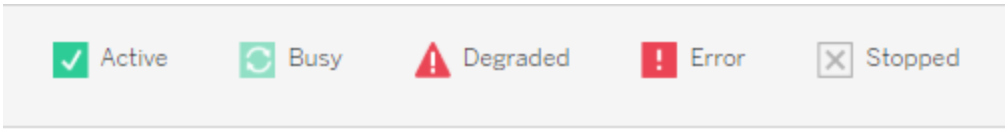
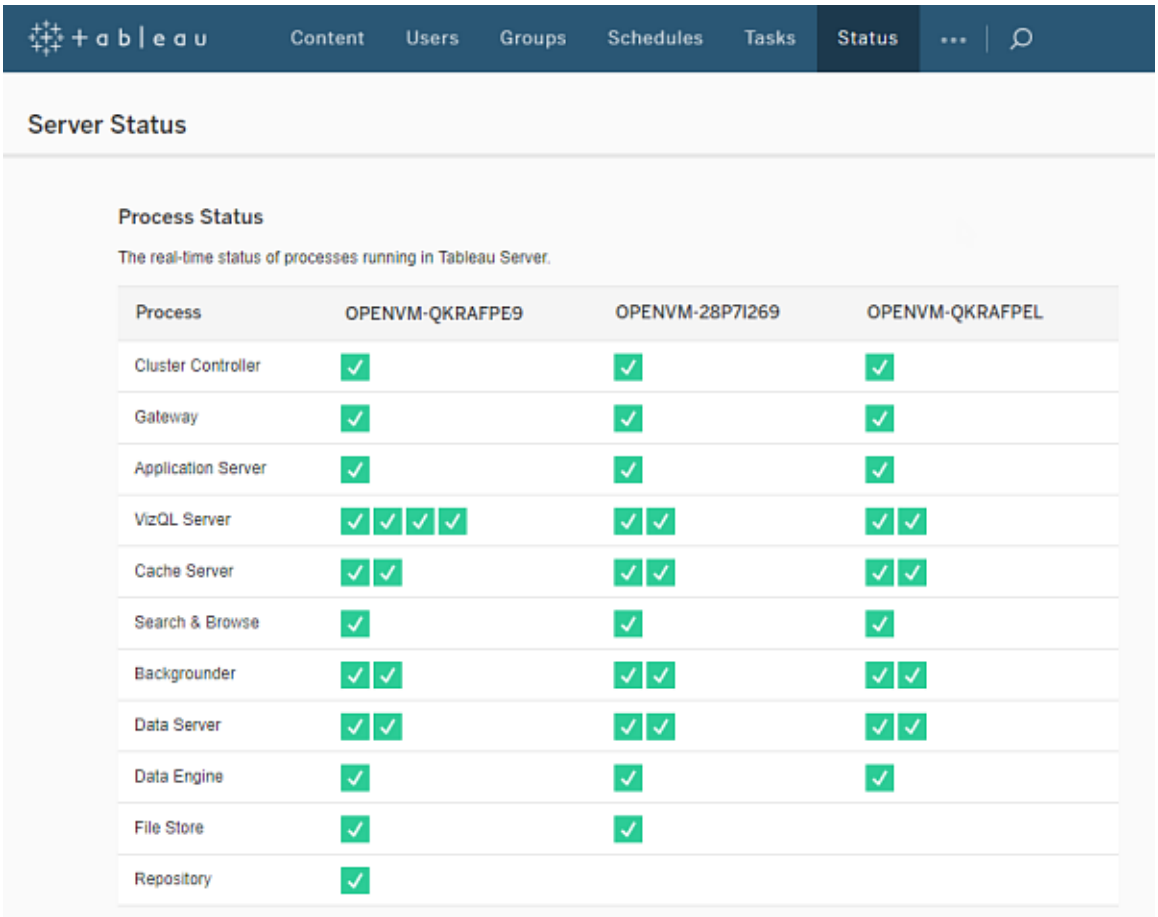


Tableau Server のステータス ページ



可能性があるステータス インジケーターが、表の下部に一覧で表示されています。

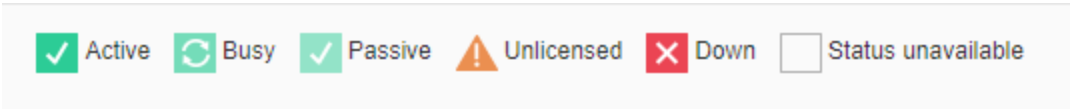
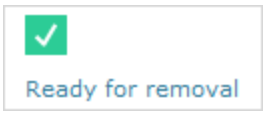


Tableau Server が正常に機能している場合、ほとんどのプロセスは [アクティブ]、[ビジー] または [パッシブ] (リポジトリ) として表示されます:

- **アクティブ**: プロセスは意図したとおりに機能しています。考えられるアクティブな状態の詳細については、サーバープロセスのトラブルシューティングのファイル ストアを参照してください。
- **ビジー**: プロセスは一部のタスクを処理中です。詳細については、サーバープロセスのトラブルシューティングの「ファイル ストアとリポジトリ」を参照してください。

- **パッシブ —** リポジトリがパッシブ モードになっているか、プロセスがトラフィックを受信できない状態になっています。詳細については、「サーバープロセスのトラブルシューティング」の「リポジトリ」および「Index and Search Server」を参照してください。
- **ライセンスなし**: プロセスはライセンスがありません。
- **停止中**: プロセスが停止しています。この意味合いはプロセスによって異なります。
- **ステータスを使用できません**: Tableau Server はプロセスのステータスを判定できません。

追加情報がある場合、ステータス アイコンの下にメッセージが表示され、適切なドキュメントへのリンクが示されています。



注: Tableau Server は、自己修正するように設計されています。サービスまたはプロセスが応答を停止するかダウンした場合、Tableau Server はそれを再起動しようとします。これを完了するまでに 15 ～ 30 分かかる場合があります。そのため、サービスまたはプロセスのアラートにすぐに対応しても、特に再起動中に要求を処理できる冗長サービスを備えたインストールでは、非生産的なアクションになる可能性があります。

プロセス ステータスのトラブルシューティングの詳細については、サーバープロセスのトラブルシューティングを参照してください。

外部ノード

一部のプロセスは Tableau Server の外部で構成できます。たとえば、ファイル ストアは SAN または NAS で構成でき、リポジトリは AWS RDS インスタンスに展開できます。このような場合、Tableau Server のステータス ページではこれらのプロセスが**外部ノード**にあることがステータス **E** によって示され、Tableau サービス マネージャー (TSM) のステータス ページではこれらのプロセスが**外部**で構成されていることがチェック マークによって示されます。

Tableau Server の外部で構成されたファイル ストアを示す Tableau サーバー マネージャー (TSM) のステータス ページ:

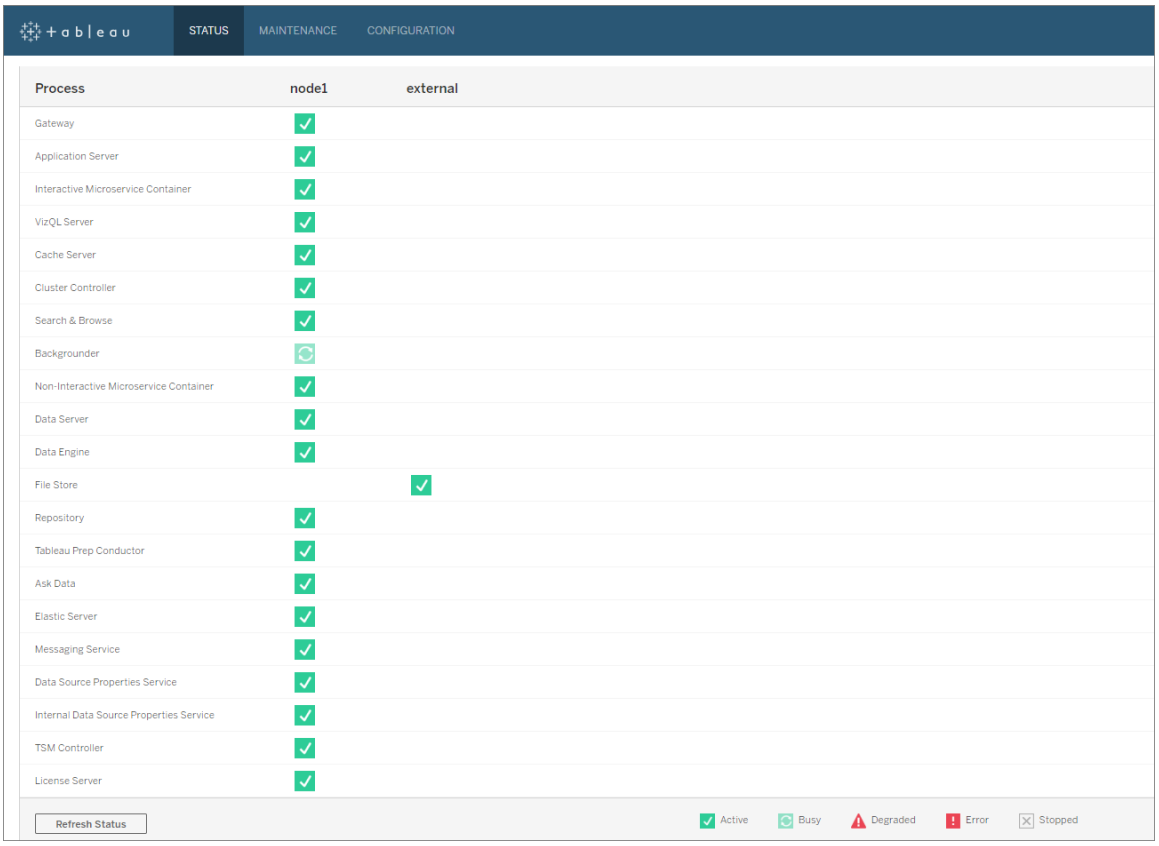
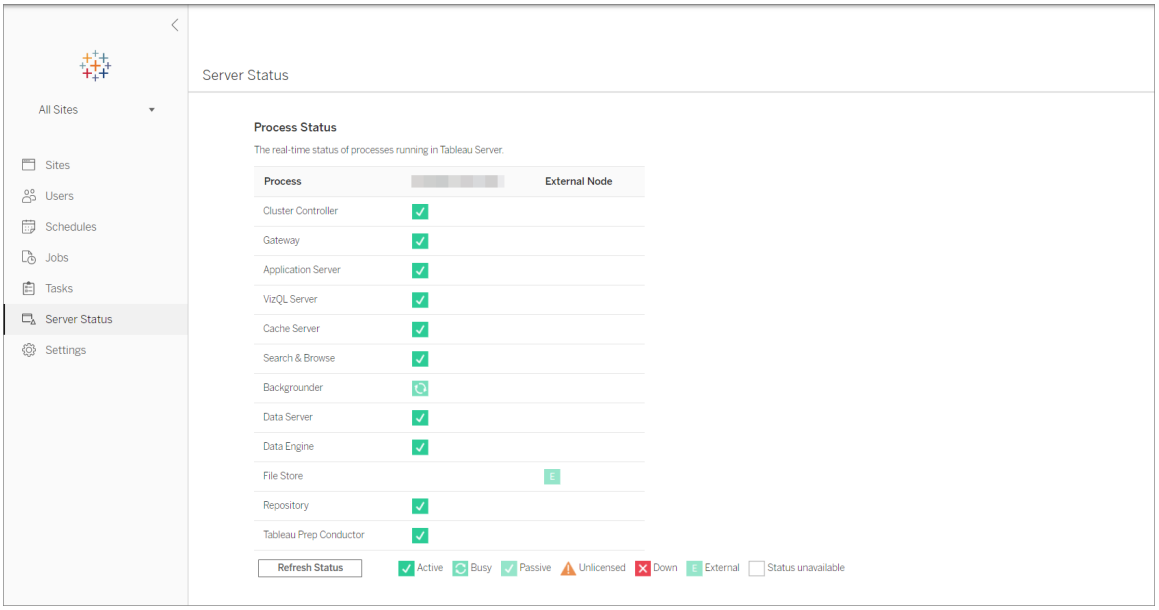


Tableau Server の外部で構成されたファイル ストアを示す Tableau Server のステータス ページ:



ステータスへのリモート アクセス

> **注**: この記事の情報は、Tableau Server ステータス ページについて言及しています。Tableau Server ステータス ページとTSM ステータス ページの詳細については、サーバー プロセス ステータスの表示を参照してください。

[サーバー ステータス] ページを表示するには Tableau Server 管理者である必要がありますが、他のコンピューターに対し、管理者以外のユーザーや最初の Tableau Server ノード以外のコンピューターによる、コンピューターが読み取り可能な [ステータス] 表の (XML) バージョンへのリモート アクセスを許可することができます。これを実行する1 つの理由は、リモート監視プロセスの一部であるためです。

Tableau Server ステータスへのリモート アクセスを許可するには、次の手順を行います。

1. 管理者としてコマンド プロンプトを開き、次のように入力します。

   ```
   tsm configuration set -k wgserver.systeminfo.allow_referrer_ips
   -v <ip address>
   ```

   上記のコマンドでは、<ip address> は Tableau Server ステータス XML へのリモート アクセスを有効にするコンピューターの IPv4 アドレスです。

   例は次のとおりです。

   ```
   tsm configuration set -k wgserver.systeminfo.allow_referrer_ips
   -v 10.32.139.31
   ```

   複数のコンピューターに対してリモート アクセスを有効にする場合は、それぞれの IP アドレスをコンマで区切ってください。

   ```
   tsm configuration set -k wgserver.systeminfo.allow_referrer_ips
   -v 10.32.139.31,10.32.139.35
   ```

   構成の変更を確定します。

   ```
   tsm pending-changes apply
   ```

2.

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

IP アドレスが追加されたコンピューターのユーザーは、ブラウザーに URL `http://<server>/admin/systeminfo.xml` を入力するか、コマンドライン (たとえば、`curl http://jsmith/admin/systeminfo.xml`) を使用して Tableau プロセスのステータスを表示できます。

ロードバランサーまたはプロキシサーバーで動作するように Tableau Server が構成されている場合は、最初の Tableau Server ノードのホスト名または IP アドレスを使用して、状態ページの XML バージョンにアクセスします。

返される XML の詳細については、プロセス ステータスを XML として取得を参照してください。

プロセス ステータスを XML として取得

機械で読み取り可能なバージョンのサーバープロセス ステータス (XML 形式のステータス) を取得するには、次の URL を使用します。

```
http://my_tableau_server/admin/systeminfo.xml
```

コンピューターが読み取り可能なプロセス ステータスを表示するには、Tableau Server にサインインするか、リモートアクセスを有効化する必要があります。

サーバーは次のようなステータス レポートを返します。

```
<systeminfo xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <machines>
    <machine name="my_tableau_server">
    <repository worker="my_tableau_server:8060" status="Active"
preferred="false"/>
    <dataengine worker="my_tableau_server:27042" status="Active"/>
```

```
    <applicationserver worker="my_tableau_server:8600"
status="Active"/>
    <apiserver worker="my_tableau_server:8000" status="Active"/>
    <vizqlserver worker="my_tableau_server:9100" status="Active"/>
    <dataserver worker="my_tableau_server:9700" status="Active"/>
    <backgrounder worker="my_tableau_server:8250" status="Active"/>
    <gateway worker="my_tableau_server:80" status="Active"/>
    <searchandbrowse worker="my_tableau_server:11000"
status="Active"/>
    <cacheserver worker="my_tableau_server:6379" status="Active"/>
    <filestore worker="my_tableau_server:9345" status="Active"
pendingTransfers="0" failedTransfers="0" syncTimestamp="2015-02-
27T20:30:48.564Z"/>
    <clustercontroller worker="my_tableau_server:12012"
status="Active"/>
    <coordination worker="my_tableau_server:12000"
status="Active"/>
    </machine>
  </machines>
  <service status="Active"/>
</systeminfo>
```

XML のステータス値

- **<プロセス> ワーカー -** プロセスを実行中のノードおよびプロセスが使用中のポートの名前。
- **status -** ノード上のプロセスのステータス。可能な値は次のとおりです: `Active`、`Passive`、`Unlicensed`、`Busy`、`Down`、`ReadOnly`、`ActiveSyncing`、`StatusNotAvailable`、`StatusNotAvailableSyncing`、`NotAvailable`、`DecommisionedReadOnly`、`DecomisioningReadOnly`、および `DecommissionFailedReadOnly`
- **pendingTransfers -** ノードが完全に同期する必要があるワークブックまたはデータソース抽出の数。これらはこのファイル ストア ノードにパブリッシュされた項目と、他のファイル ストア ノードにパブリッシュされ、このノードにコピーする必要がある項目を表します。

- **failedTransfers -** 前回の自動化ジョブの間にこのファイル ストア ノードに正常に転送されなかったワークブックまたはデータ ソースの数。自動化されたジョブは通常 15 分または 30 分毎に実行しますが、抽出の数や量が多い場合にはこれよりも時間がかかる場合があります。

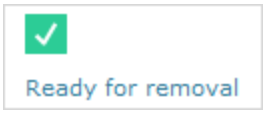
  転送に失敗したからといって必ずしも Tableau Server に問題があるとは限りません。定期的な自動ジョブでは通常、以前の同期中に失敗したファイルを転送します。ファイル転送の失敗理由はログに記載されています。

- **syncTimestamp -** ファイルを同期および実行した前回の自動化ジョブの時間 (UTC)。

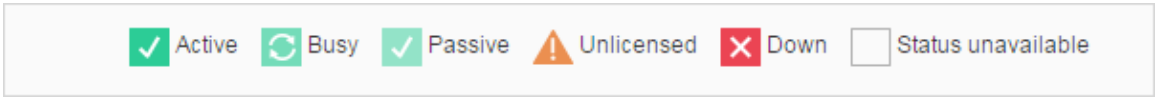
サーバープロセスのトラブルシューティング

> **注**: この記事の情報は、Tableau Server ステータス ページについて言及しています。Tableau Server ステータス ページとTSM ステータス ページの詳細については、サーバープロセス ステータスの表示を参照してください。

Tableau Server が正常に機能している場合、プロセスは [アクティブ]、[ビジー] または [パッシブ] (リポジトリ) として表示されます: 追加情報がある場合、ステータス アイコンの下にメッセージが表示されます。



可能性があるステータス インジケーター:



注: Tableau Server は、自己修正するように設計されています。サービスまたはプロセスが応答を停止するかダウンした場合、Tableau Server はそれを再起動しようとします。これを完了するまでに 15 ～ 30 分かかる場合があります。そのため、サービスまたはプロセスのアラートにすぐに対応しても、特に再起動中に要求を処理できる冗長サービスを備えたインストールでは、非生産的なアクションになる可能性があります。

次のセクションでは、表示される可能性のあるステータス メッセージに対するトラブルシューティングの推奨事項を提供します。

クラスタ コントローラー

このメッセージは、2 ノードより多い場合にのみ表示されます。

ステータス:ダウン、メッセージ: "ノードが低下しています"

次のうち 1 つ以上に当てはまります。

- ノード上のリポジトリが停止しました。
- ノードはクラスタ内の他の場所のフェールオーバーに応答できません。
- Tableau Server が高可用性に設定されており、これがアクティブなリポジトリの場合、2 番目のリポジトリへのフェールオーバーが発生します。
- このノード上のリポジトリまたはファイル ストアで利用可能なステータスがありません。

クラスタ コントローラーが定期的または長時間にわたって停止しない限り、操作は必要ありません。

これが発生した場合、問題が解決するまで、次の操作を順に実行します。

1. ディスク容量を確認する。ディスク空き容量が制限されている場合は、サポートが必要な場合に備えてログ ファイルを保存し (`tsm maintenance ziplogs` を使用)、不要なファイルを削除します。
2. Tableau Server を再起動します。
3. クラスタ コントローラーが引き続きダウンしている場合、ログ ファイル (`tsm maintenance ziplogs`) を保存してからサポートに連絡します。

ファイル ストア

ファイル ストアのステータスは、ページが読み込まれているファイル ストアの状態のみを反映します。

メッセージなしのアクティブ ステータス ( ) は、ページが読み込まれたときに抽出が同期されなかったことを示します。反復して "catch-all" ジョブが実行し、抽出を同期している可能性があります。

ステータス: ビジー、メッセージ: "同期中"

「同期中」は通常、ページが読み込まれたときにファイル ストア ノード全体で抽出が同期されていたことを示します。

ただし、"同期中" メッセージは、次のインストール (シングル ノードとマルチ ノードの両方) でも返されます。Tableau の初期化後、ステータスは 15 分または 20 分以内に消えます。

ステータス: ダウン、"データ抽出は使用できません"

シングル ノード インストールの場合: "データ抽出は使用できません" は、既存の抽出を利用可能な場合がありますが、パブリッシュ/更新に失敗することを示します。マルチ ノード インストールの場合、このメッセージはこのノードで抽出の同期が失敗することを示します。

ファイル ストアが定期的または長時間にわたって停止しない限り、操作は必要ありません。

これが発生した場合、問題が解決するまで、次の操作を順に実行します。

1. ディスク容量を確認する。ディスク空き容量が制限されている場合は、サポートが必要な場合に備えてログ ファイルを保存し (`tsm maintenance ziplogs` を使用)、不要なファイルを削除します。
2. Tableau Server を再起動します。
3. ファイル ストアが引き続きダウンしている場合、ログ ファイル (`tsm maintenance ziplogs`) を保存してからサポートに連絡します。

ステータス: ビジー、"コミッション解除"

このメッセージは、このファイル ストアが読み取り専用モードであり、このノードの 1 つ以上の一意のファイルが他のファイル ストア ノードへのレプリケートに失敗したことを示します。

このノードを削除するには、ステータス メッセージが「削除準備完了」に変更されるまで待機します。

ステータス: アクティブ、"削除準備完了"

このメッセージは、ファイル ストアが読み取り専用モードであることを示します。

クラスタを安全に停止 (`tsm stop`) してファイル ストア プロセスを削除するか、ノード全体を削除できます。

ステータス: アクティブ、"コミッション解除失敗"

このメッセージは、ファイル ストアが読み取り専用モードであり、1 つ以上の一意のファイルが他のファイル ストア ノードへのレプリケートに失敗したことを示します。

コミッション解除の失敗を解決するには:

1. `tsm topology filestore decommission` コマンドを再度実行します。
2. 他のファイル ストア ノード上のディスク容量を確認します。別のファイル ストア ノードに、すべての抽出を保存するのに十分な容量がない場合、コミッションの解除に失敗します。
3. 最初のノードと追加ノード上の `tsm.log` ファイルにエラーがないか確認します。
4. Tableau Server を停止 (`tsm stop`) してから、`tsm topology filestore decommission` コマンドを再度実行します。
5. ファイル ストア ノードをもう一度読み取り/書き込みモード (`tsm topology filestore recommission`) に戻し、ログを収集してからサポートに連絡します。
6. サポートを使用: このファイル ストアから別のファイル ストア ノード上の同じディレクトリへ `extracts` ディレクトリをコピーしてマージします。

インデックス作成/検索サーバー

ステータス: パッシブ、メッセージ: n/a

マルチノード環境では、パッシブ ステータスは、ノードが意図したとおりに動作しているが、クラスタに参加してトラフィックを受信できないことを示します。

インデックスおよび検索サーバープロセスをアクティブな状態にするには:

1. tsm topology set-process コマンドを使用して、ノードからパッシブ インデックスおよび検索サーバープロセスを削除します。

   ```
   tsm topology set-process -n <Node> -pr indexandsearchserver -c 0
   ```

2. 変更を適用します (tsm pending-changes apply)。

3. 再起動 しますTableau Server (tsm restart)。

4. tsm topology set-process コマンドを使用して、インデックスおよび検索サーバープロセスを一度に 1 つずつノードに追加します。

   ```
   tsm topology set-process -n <Node> -pr indexandsearchserver -c 1
   ```

5. 変更を適用します (tsm pending-changes apply --ignore-warnings)。

6. 再起動 しますTableau Server (tsm restart)。

7. tsm status コマンドを使用して、影響を受けるノードで `indexandsearchserver` のステータスをチェックします。

リポジトリ

ステータス: ビジー、メッセージ: "設定中"

"設定中" メッセージは次のステータスの 1 つ以上を示します。

- パッシブ リポジトリはアクティブなリポジトリと同期されます。
- リポジトリはフェールオーバーに対応する準備が整っていません。
- リポジトリはアクティブなリポジトリから2 分を超えて遅れており、再度セットアップされている可能性があります (同期を待機するより短時間で完了する)。
- フェールオーバーが発生し、アクティブであったリポジトリはクラスタに再結合されます。

リポジトリのステータス メッセージが「パッシブ" ではなくなるまで待機します。.

このメッセージが表示されない、または時間がかかりすぎる場合:

1. ディスク容量を確認し、可能であれば空き容量を増やします。
2. クラスタコントローラーのログにエラーがないことを確認します。
3. ノードを再起動します。

ステータス: ビジー、メッセージ: "同期中"

リポジトリは、フェールオーバー後などに同期されます。

ステータス: ダウン、メッセージ: n/a

リポジトリがダウンステータスを示し、メッセージがない場合、リポジトリは次のいずれかのステータスになります。

- インストールが高可用性用に設定されている場合、リポジトリのフェールオーバーが発生します。
- プロセスは、フェールオーバー後、更新されたデータベース接続構成で再起動します。
- 別のアクティブなリポジトリを利用できない場合、Tableau Server が停止しています。

手順により問題が解決するまで、これらの操作を順に実行します。

1. クラスタコントローラーが再起動を試みるのを数分間待機します。
2. Tableau Server を再起動します (`tsm restart`)。
3. ディスク容量を確認する。ディスク空き容量が制限されている場合は、サポートが必要な場合に備えてログファイルを保存し (`tsm maintenance ziplogs` を使用)、不要なファイルを削除します。
4. Tableau Server を再起動します。
5. リポジトリが引き続きダウンしている場合、ログファイル (`tsm maintenance ziplogs`) を保存してからサポートに連絡します。

ステータス: パッシブ、メッセージ: n/a

メッセージなしのパッシブステータスは、ノードが意図したとおりに動作しており、必要に応じてフェールオーバーの準備が整っていることを示します。

VizQL Server

ステータス: ライセンスなし、メッセージ: n/a

VizQL Server プロセスのライセンスなしステータスの詳細については、ライセンスのないサーバープロセスの処理 を参照してください。

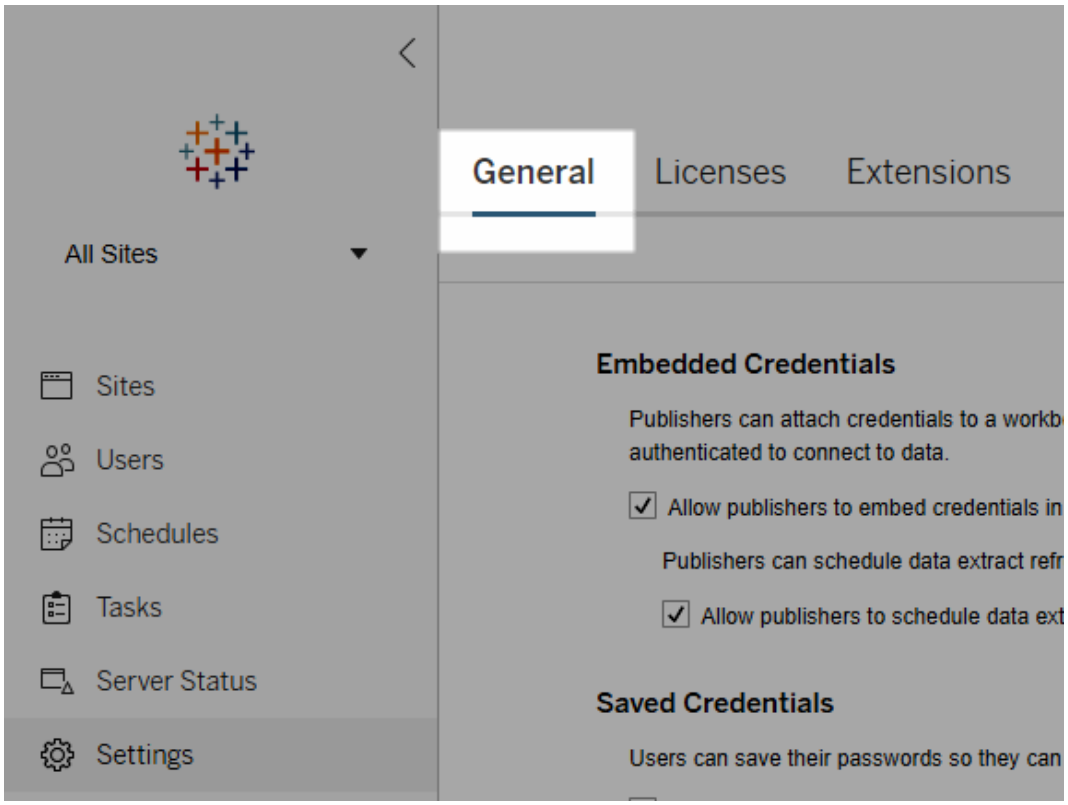
保存されたデータ接続パスワードのクリア

管理者として、ユーザーによるデータソース パスワードの保存を有効にすると、サーバー ユーザーは複数のアクセスとブラウザーにわたってデータ ソースのパスワードを保存できるようになり、データ ソースに接続するたびに認証資格情報を入力するように要求されなくなります。

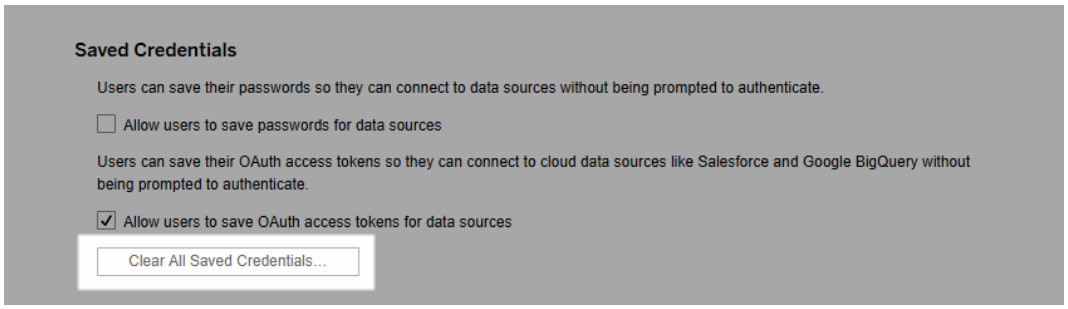
すべての Tableau Server ユーザーのデータ ソース パスワードをリセットできます。ユーザーが次回データベース認証を必要とするビューにアクセスしたときに、データ ソースへのサインインを強制できます。サーバー ユーザーは [ユーザー設定] ページを使用して、個別に保存されているデータ接続 パスワードをクリアすることもできます。

すべてのサーバー ユーザーの保存されているデータ接続 パスワードをクリアにするには:

1. 単一サイト サーバーでは、**[設定] > [全般]** をクリックします。複数サイト サーバーでは、**[すべてのサイトを管理]**、**[設定] > [全般]** の順にクリックします。

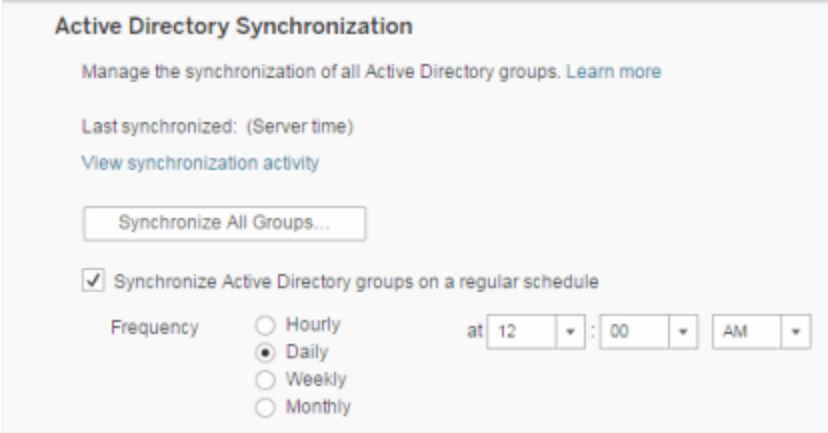


2. 保存された認証資格情報] から、**[保存されたすべての認証資格情報の削除]** をクリックします。



サーバー上のすべての外部ディレクトリグループの同期

サーバー管理者は、サーバーの **[設定]** ページの **[全般]** タブで、(Tableau Server で設定済みの) すべての外部ディレクトリグループ (Active Directory など) をスケジュールに基づいて、またはオンデマンドで同期できます。



注: ユーザーとグループの同期のコンテキストでは、LDAP アイデンティティストアを使用して構成された Tableau Server は Active Directory と同等です。Tableau Server の Active Directory 同期機能は、適切に構成された LDAP ディレクトリソリューションとシームレスに連動します。

はじめる前に

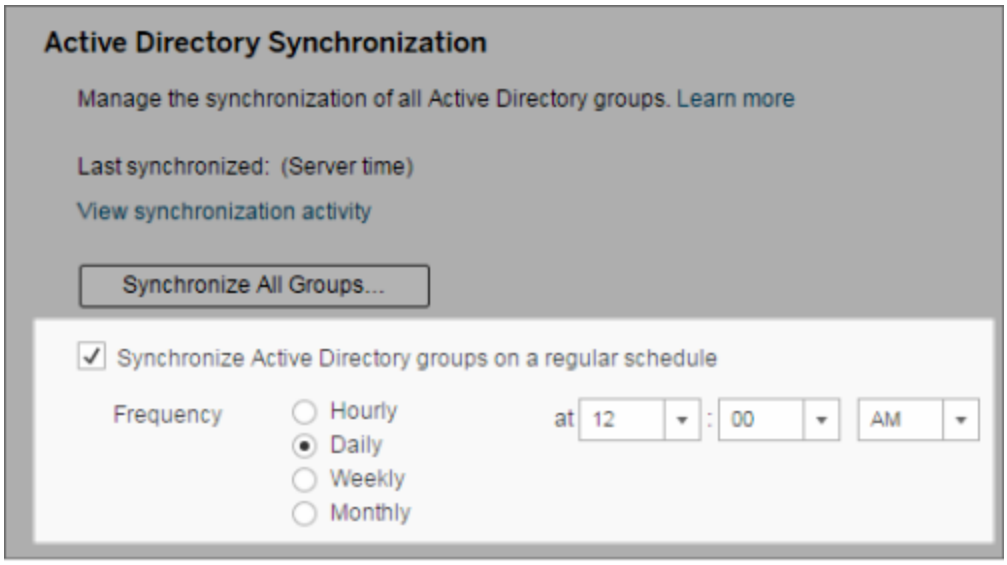
このトピックの説明に従ってグループを同期する前に、まず外部ディレクトリグループを Tableau Server にインポートする必要があります。Active Directory 経由でのグループの作成を参照してください。

外部ディレクトリグループのスケジュールに基づく同期

1. **単一のサイト**: **[設定] > [全般]** をクリックします。

   マルチサイト: サイト メニューで、**[すべてのサイトを管理]**、**[設定] > [全般]** の順にクリックします。

2. ページを下にスクロールして **[Active Directory の同期]** に移動し、**[スケジュール上の Active Directory グループの同期]** を選択します。



3. 同期の頻度と時間を選択します。
4. **[保存]** をクリックします。

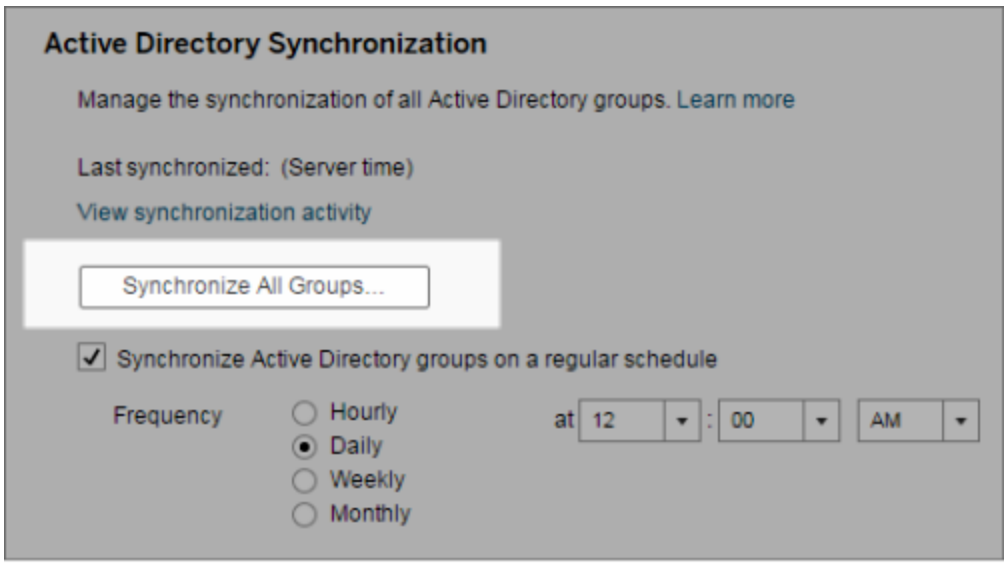
注: バージョン 2021.1.23、2021.2.21、2021.3.20、2021.4.15、2022.1.11、2022.3.3、2023.1 以降では、スケジュールされたグループ同期がキャンセルされるまでの待機時間が、既定の時間制限 (4 時間) になります。スケジュールされた同期が非常に大きなグループの同期である場合、または既定よりも長い時間がかかっている場合は、サーバー管理者はこの制限時間を変更することができます。詳細については、「すべての Active Directory グループをスケジュールに従って同期する」および「backgrounder.timeout.sync_ad_group」を参照してください。

外部ディレクトリグループのオンデマンドでの同期

外部ディレクトリグループ (Active Directory など) は、いつでもTableau Serverと同期でき、外部ディレクトリの新規ユーザーと変更はTableau Server上のすべての外部ディレクトリグループに反映されます。

1. **単一のサイト**: **[設定] > [全般]** をクリックします。

   マルチサイト: サイト メニューで、**[すべてのサイトを管理]**、**[設定] > [全般]** の順にクリックします。



2. **[Active Directory の同期]** で、**[すべてのグループの同期]** をクリックします。

同期アクティビティの表示

[抽出以外のバックグラウンド タスク] 管理ビューに同期ジョブの結果を表示できます。**[Active Directory グループ同期のキューイング]** とは、キューイングを行うタスクで、実行される **[Active Directory グループの同期]** タスクの数を示します。

1. **単一のサイト**: **[状態]** をクリックします。

   マルチサイト: サイト メニューで、**[すべてのサイトを管理]**、**[ステータス]** の順にクリックします。

2. **[抽出以外のバックグラウンド タスク]** リンクをクリックします。

3. **[タスク]** フィルターに **[Active Directory グループ同期のキューイング]** および **[Active Directory グループの同期]** を設定します。

サーバーの **[設定]** ページの **[同期アクティビティの表示]** リンクをクリックすると、この管理ビューに素早く移動できます。

外部ディレクトリグループのユーザーに対して最小限のサイトロールを設定する

[グループ - 詳細] ページでは、Active Directory の同期中に適用されるグループユーザー向けの最小限のサイトロールを設定できます。

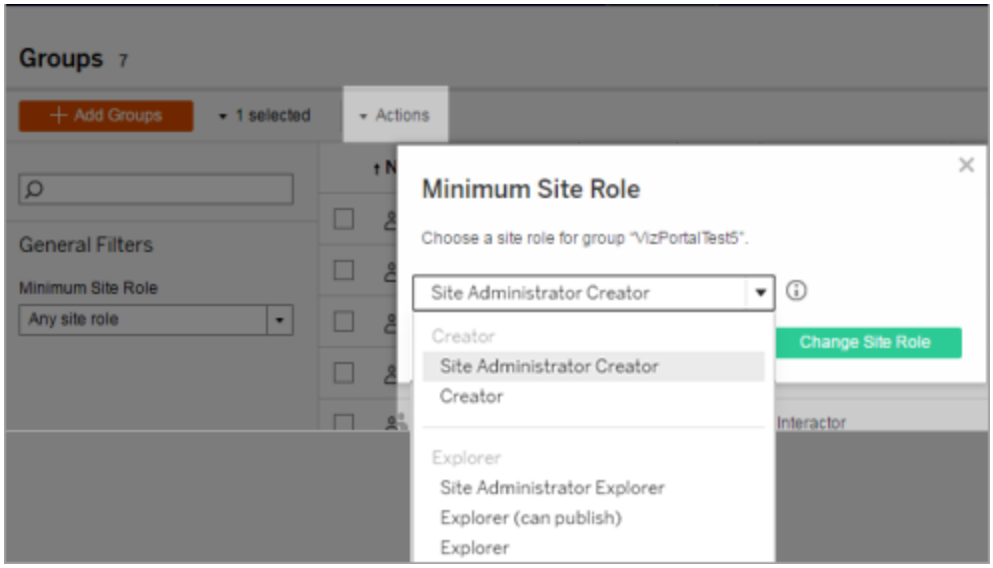
この設定では同期は実行されません。代わりに、同期が実行されるたびにグループに適用する最小限のサイトロールを設定します。その結果、外部ディレクトリグループを同期すると、最小限のサイトロールが割り当てられた新しいユーザーがサイトに追加されます。既にユーザーが存在する場合、ユーザーにとってサイト内でのアクセス範囲が増える場合にのみ最小限のサイトロールが適用されます。最小限のサイトロールを設定しない場合、新しいユーザーは既定で**ライセンスなし**として追加されます。

注: ユーザーのサイトロールのレベルが上がりますが、最小限のサイトロール設定に基づいてレベルが下がることはありません。ユーザーがパブリッシュする機能をすでに持っている場合、その機能は引き続き維持されます。最小限のサイトロールの詳細については、「サイトロールと Active Directory のインポートと同期」を参照してください。

1. サイト内で **[グループ]** をクリックします。
2. [グループ] ページでグループを選択します。

   [アクション] > [最小限のサイトロール] をクリックします。

3. 最小限のサイトロールを選択してから、**[サイトロールの変更]** をクリックします。



参照元の外部ディレクトリからユーザーが削除されるとどうなりますか。

外部ディレクトリの同期操作を通じて Tableau Server からユーザーを自動的に削除することはできません。外部ディレクトリグループからユーザーが無効または削除されても、Tableau Server では引き続き有効になっているため、管理者はユーザーのアカウントを完全に削除する前にユーザーのコンテンツを監視して再び割り当てることができます。詳細については、Active Directory からユーザーを削除する際の動作の同期を参照してください。

グループ同期のパフォーマンスの向上

外部ディレクトリの同期はバックグラウンダープロセスによって実行されます。このバックグラウンダープロセスは抽出の管理や作成に使用されるものと同じプロセスであり、サブスクリプション コンテンツの生成にも使用されます。動的なグループ メンバーシップと大量の抽出が使用される大規模な組織では、外部ディレクトリのグループ同期プロセスによって業務に支障が出る可能性があります。業務時間外にグループ同期を実行することをお勧めします。

既定では、バックグラウンダープロセスは逐次操作で同期を実行します。つまり、各グループが 1 つのバックグラウンダープロセスで 1 つずつ同期されます。単一の Tableau Server や分散展開でバックグラウンダープロセスの複数のインスタンスを実行している場合は、外部ディレクトリの同期で並列処理を有効にすることを検討してください。並列バックグラウンダー処理が有効になっていると、パ

フォーマンスを向上させるためにグループ同期が複数のバックグラウンダープロセスに分散されます。

グループ同期で並列バックグラウンダー処理を有効にするには、TSM CLI を開いて次のコマンドを入力します。

```
tsm configuration set -k backgrounder.enable_parallel_adsync -v
true

tsm pending-changes apply
```

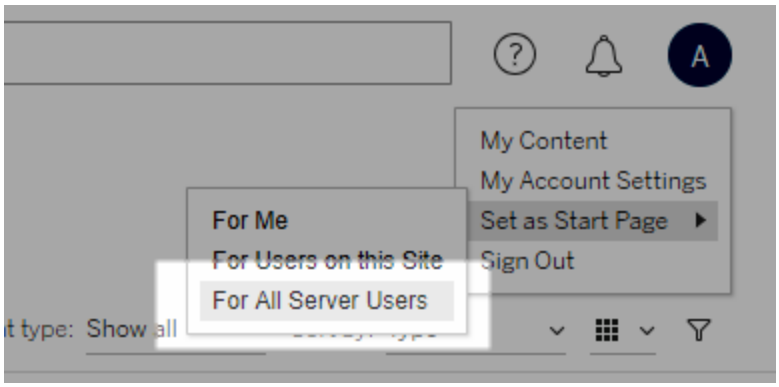
すべてのユーザーの既定のスタートページの設定

新しい展開でユーザーが Tableau Server Web 作成環境にサインインすると、役割ベースのウェルカム バナーが表示される [ホーム] 画面に移動します。[ホーム] には、最近のビュー、お気に入り、サイトで最も表示されたコンテンツも表示されます。サーバー管理者はユーザーの既定ランディング ページをサーバーおよびサイトレベルで変更できます。たとえば、すべてのワークブックを表示でき、ユーザーがサインインすると、アクセス権のあるワークブックが表示されます。

すべてのユーザーの既定のスタートページを設定するには

1. ユーザーがサイトにサインインする際に、既定ページとしたいページやフィルターされたビューを表示します。
2. ページの右上エリアにあるプロフィール アイコンを選択します。
3. 次のユーザーの既定のスタートページを設定するには:
    - サイト上のすべてのユーザー: **[スタート ページとして設定]**、**[このサイトのユーザー用]** を順にクリックします。
    - サーバー上のすべてのユーザー: **[スタート ページとして設定]**、**[すべてのサーバー**

ユーザー用] を順にクリックします。



ユーザーが設定したスタートページと階層

ユーザーはプロフィール アイコンから独自のスタートページを設定し、アカウント設定でスタートページをリセットすることができます (詳細については、お使いのオペレーティング システムの Tableau Server オンライン ヘルプで「プロファイルおよびアカウント設定へのアクセス」を検索してください)。

ユーザーが独自のスタートページを設定する場合、サーバー管理者によって設定されたスタートページは上書きされます。次回ユーザーがサインインするときに、設定したスタートページにランディングします。サーバー管理者がサーバーとサイトの両方にスタートページを設定する場合、ユーザーには既定で自分で設定したスタートページ、次にサイトのスタートページ、次にサーバーのスタートページが表示されます。ユーザーも管理者もスタートページを設定していない場合、ユーザーには既定でホームが表示されます。

接続されているクライアントからサイトにアクセス

既定では、Tableau Server でユーザーは Tableau *クライアント*からサイトに直接アクセスできます。ユーザーが初めてクライアントからサインインするときに認証資格情報を入力すると、アクセスが許可されます。この場合のクライアントは、Tableau Server と情報を交換できる Tableau アプリケーションまたはサービスです。Tableau クライアントの例として、Tableau Desktop、Tableau Prep Builder、および Tableau Mobile があります。

Tableau Server は、ユーザーがクライアントからサインインした際にユーザーを一意的に識別するセキュアな更新トークンを作成し、*接続済みのクライアント*を確立します。

自動クライアント認証を無効にする

Tableau に接続されたクライアント(Tableau Desktop、Tableau Mobile、Tableau Prep Builder など) および個人用アクセストークン(PAT) が Tableau Server に正常にサインインすると、それ以降は自動的に認証されます。接続されたクライアントセッションとPAT は両方とも、更新トークンで管理されます。

既定では、更新トークンは 1 年経過後にリセットされます。更新トークンが 14 日間使用されなかった場合、有効期限切れになります。サーバー管理者者は、`refresh_token.absolute_expiry_in_seconds` および `refresh_token.idle_expiry_in_seconds` オプションを設定して、これらの値を変更できます。tsm configuration set のオプションを参照してください。

Tableau Server 管理者は、接続されたクライアントの自動認証を無効にすることもできます。この場合、セッションの有効期限は、Web 作成セッションを管理する Tableau Server セッションの動作によってのみ管理されます。9.セッション存続時間設定の検証 を参照してください。9.セッション存続時間設定の検証を参照してください。Web 作成セッションは「接続されたクライアント」とは見なされず、更新トークンは使用されません。

Tableau Serverからすべてのクライアントを直ちに切断し、ユーザーが接続するたびに毎回サインインを要求するには:

1. Tableau Server にサーバー管理者としてサインインします。
2. サイトメニューで、**[すべてのサイトを管理]**、**[設定]** > **[全般]** の順にクリックします。
3. **[接続されたクライアント]** の下の **[クライアントに Tableau Server への自動接続を許可する]** チェックボックスをオフにします。
4. ページの上部または下部にある **[保存]** ボタンをクリックします。

注: 上記の設定は接続されたクライアントのみに適用され、PAT の作成と適用には影響しません。

不必要なファイルの削除

ベストプラクティスとして、サーバー上のディスク容量使用率を定期的に監視することをお勧めします。Tableau Server PC のディスク空き容量が不足し始めると、最終的に障害が発生することなど、影響が大きくなる可能性があります。空き容量が少なくなっていると判断した場合、保存する必要があるファイルをアーカイブして、不要なファイルを消去すると、Tableau の容量を解放できます。

ディスク容量使用率の監視

ディスク容量使用率を監視するために実行できる操作がいくつかあります。

- 通知: ディスク空き容量が事前に定義されたレベルに達したときに通知を送信するように Tableau Server を構成できます。詳細については、「サーバー イベント通知の構成」を参照してください。
- 管理ビュー: 事前に作成された管理ビューを使用して、ディスク容量使用率を監視できます。詳細については、「サーバーのディスク容量」を参照してください。

ディスク容量使用率の削減

使用可能なディスク容量を増やすには、次の手順を実行します。

- ログ ファイルのアーカイブ: Tableau Server は、実行時にログ ファイルを生成します。これらのログ ファイルは、問題のトラブルシューティングや Tableau サポートと連携する場合に役立ちますが、無期限にそのままにしておく必要はありません。ログをそのまま残しておいてもディスク容量を節約するには、`tsm maintenance ziplogs` コマンドを使用してログをアーカイブし、Tableau インストールの一部ではない PC に ziplog アーカイブをコピーして安全に保管します。詳細については、「ログ ファイルのスナップショット (ログのアーカイブ)」を参照してください。

- 不要なファイルのクリーンアップ: 保存するログをアーカイブした後、`tsm maintenance cleanup` コマンドを使用して、7 日間より古いログ ファイルや一時ファイルを削除したり、必要に応じて、`http_requests`Tableau Server リポジトリのテーブルから行を削除したりします。定期的にクリーンアップ コマンドを実行する必要があります。削除するファイルの詳細については、「tsm maintenance cleanup」を参照してください。

- 他のファイルの削除: Tableau Server では、そのままにする必要がないファイルが時間の経過とともに生成される可能性があります。上記のファイルに加えて、以前のバージョンからの古いバックアップなどのファイルに注意してください。Tableau Server バックアップ ファイルには .tsbak 拡張子が付きます。Tableau を定期的にバックアップし、安全に保管するために、Tableau Server インストールの一部ではない PC にバックアップ ファイルを保存することを強くお勧めします。ファイルを別の場所に保存すると、Tableau PC から削除できます。バックアップ ファイルの作成方法や保存場所など、バックアップ ファイルの詳細については「Tableau Server データのバックアップ」を参照してください。

サーバー設定 (全般とカスタマイズ)

次の設定は **[サーバー - 設定]** の **[全般]** および ***カスタマイズ*** ページで使用できます。

サーバー上に複数のサイトがある場合、これらの設定の多くは [サーバー設定] ページから[サイト設定] ページに移動します。これらの設定には、[Moves to Site Settings on multi-site servers (複数サイト サーバーのサイト設定に移動)] というマークが表示されます。

全般

設定	**説明**
サイト名と ID 複数サイト サーバーの [サイト設定] に移動します。	ユーザー インターフェイスに表示されるサイト名と、サイトの URL に表示される ID を指定します。([既定] サイトを編集している場合は、ID を変更できません)。 URL の "#/site" の部分を変更することはできません (例: http://localhost/#/site/sales)。マルチサイト サーバー環境では、これらの設定が既定サイト以外のサイトの URL に表示されます。
ストレージ 複数サイト サーバーの [サイト設定] に移動します。	**[サーバーの上限]** または **[GB]** のいずれかを選択します。[GB] を選択した場合は、パブリッシュされたワークブックや抽出などのデータ ソースに適用するストレージ容量の制限に必要な GB 数を入力します。 サーバーの上限を設定してサイトがそれを超えた場合、パブリッシャーはサイトが制限内に戻るまで、新しいコンテンツを

	アップロードできません。サーバー管理者は、[サイト] ページの [最大容量] および [使用容量] 列を使用して、容量に対するサイトの使用量を管理できます。
リビジョン履歴 複数サイトサーバーの [サイト設定] に移動します。	サーバーに保存されているワークブック、フロー、およびデータソースの以前のバージョンの数を指定します。
ユーザーの管理 複数サイトサーバーの [サイト設定] に移動します。	サーバー管理者のみがユーザーの追加と削除、およびサイトロールの変更を実行できるようにするか、サイト管理者にも権限を付与するかを指定します。 サイト管理者にユーザーの管理権限を与える場合、サイト管理者がサイトに追加できるユーザー数を、次のいずれかを選択して指定します。 • [サーバーの上限] では、利用可能なサーバー接続ライセンス数を追加します。コアベースのライセンス設定を使用するサーバーについては、制限はありません。 • [サイトの上限] では、サイト管理者が、指定された上限までユーザーを追加できます。 • [サイトロールの上限] では、サイト管理者が、サイトに指定されたライセンス上限まで各サイトロールのユーザーを追加できます。 詳細については、「サーバー ライセンスの表示」を参照してください。
Web 作成 複数サイトサーバーの [サイト設定] に移動します。	サイトのブラウザー ベースの作成を有効にするかどうかを制御します。ワークブックの Web 作成が無効になっている場合、ユーザーはパブリッシュされたワークブックをサーバー Web 環境から作成または編集することはできませんが、代わりに、Tableau Desktop を使用してワークブックをもう一度パブリッシュする必要があります。フローの Web 作成が無効になっている場合、ユーザーはパブリッシュされたフローをサーバー Web 環境から作成または編集することはできませんが、代わ

	りに、Tableau Prep Builder を使用してフローをもう一度パブリッシュする必要があります。 詳細については、Tableau Cloud ヘルプの「サイトの Web 作成アクセスと機能の設定」を参照してください。
Tableau Prep Conductor 複数サイトサーバーの [サイト設定] に移動します。	適切なパーミッションを持つユーザーがフローをスケジュール設定および監視できるかどうかを制御します。Tableau Prep Conductor は、Tableau データ管理 の一部です。詳細については、「Tableau Prep Conductor について」を参照してください。
Tableau Catalog 複数サイトサーバーの [サイト設定] に移動します。	Tableau Server または Tableau Cloud サイトが データ管理でライセンス設定されている場合は、Catalog 機能をオフにします。詳細については、「Catalog の無効化」を参照してください。
メール設定 複数サイトサーバーの [サイト設定] に移動します。	アラートとサブスクリプションの自動 メールに表示される送信元アドレスとメッセージ フッターを指定します。
スケジュールされた更新を実行した後のワークブック パフォーマンス 複数サイトサーバーの [サイト設定] に移動します。	最近表示したワークブックをスケジュールされた更新で事前に計算し、それらのワークブックをより速く開きます。詳細については、「スケジュールされた更新後のワークブック パフォーマンスの設定」を参照してください。
ワークブック パフォーマンスのメトリクス 複数サイトサーバーの [サイト設定] に移動します。	サイトユーザーがワークブックの実行方法 (読み込みの速度など) に関するメトリクスを収集できるようにします。記録を開始するには、ワークブックの URL にパラメーターを追加する必要があります。詳細については、「パフォーマンスの記録の作成」を参照してください。
[管理対象のキーチェーンのクリーン アップ] 複数サイトサーバーの [サイト	サイト管理者は、サイトで OAuth 接続用に保存された認証資格情報キーチェーンを管理できます。詳細については、「OAuth 接続」を参照してください。

設定] に移動します。	
[抽出更新タスクを自動的に一時停止する] 複数サイトサーバーの [サイト設定] に移動します。	Tableau ではリソースを節約するために、非アクティブなワークブックの抽出更新タスクを自動的に中断することができます。この機能は、毎週またはより頻繁に実行される更新スケジュールのみに適用されます。詳細については、Tableau Cloud ヘルプの「非アクティブなワークブックの抽出更新を自動的に中断する」を参照してください。
ユーザーの表示状態 複数サイトサーバーの [サイト設定] に移動します。	他のユーザーに表示されるユーザー名とグループ名を制御します。詳細については、Tableau Cloud ヘルプの「サイトユーザーの表示状態の管理」を参照してください。
「データに聞く」の可用性 複数サイトサーバーの [サイト設定] に移動します。	データソースに対して既定で「データに聞く」を有効にするか無効にするかを制御します。「データに聞く」では、ユーザーは会話言語を使用してデータを照会し、自動的に Viz を表示できます。詳細については、Tableau ユーザーヘルプの「「データに聞く」機能を使用した自動的なビューの構築」を参照してください。
「データの説明を見る」の可用性 複数サイトサーバーの [サイト設定] に移動します。	適切なパーミッションを持つサイトユーザーが「データの説明を見る」を実行できるかどうか、および作成者が「データの説明を見る」の設定にアクセスできるかどうかを制御します。詳細については、「データの説明を見る」へのアクセスの制御を参照してください。「データの説明を見る」の詳細については、「データの説明を見る」を使用して、インサイトをより早く発見するを参照してください。
[データベースおよびテーブルに関するメタデータへの自動アクセス] 複数サイトサーバーの [サイト設定] に移動します。	派生パーミッションを使用して、外部アセットに対する特定の機能をユーザーに自動的に付与します。詳細については、Tableau Cloud ヘルプの「派生パーミッションをオフにする」を参照してください。
[系列の機密データ]	関連するメタデータに対する適切なパーミッションをユーザーが持っていない場合に、系列の機密データを難読化する

複数サイトサーバーの [サイト設定] に移動します。	か、またはフィルターリングするかを指定します。詳細については、「系列の機密データ」を参照してください。
保存中の抽出の暗号化 複数サイトサーバーの [サイト設定] に移動します。	Tableau Server に保存中の .hyper 抽出を暗号化できます。Server 管理者は、サイト上のすべての抽出の暗号化を実施したり、特定のパブリッシュ済みワークブックやデータソースに関連付けられたすべての抽出をユーザーが暗号化するのを許可したりできます。詳細については、「保存中の抽出の暗号化」を参照してください。
Tableau Mobile 複数サイトサーバーの [サイト設定] に移動します。	• オフラインのプレビューを有効にする ユーザーが Tableau Mobile のサイトにアクセスしたときに、オフラインプレビューを生成して表示するかどうかを制御します。詳細については、「Tableau Mobile 展開ガイド」の「デバイスでの Tableau Mobile データの管理」を参照してください。 • アプリロックを有効にする Tableau Mobile でユーザーがサイトを開くには、生体認証方法またはデバイスパスコードが必要です。詳細については、「Tableau Mobile 展開ガイド」の「アプリロックを有効にしてセキュリティを強化する」を参照してください。
共有 複数サイトサーバーの [サイト設定] に移動します。	ユーザーが他のユーザーとアイテムを直接共有できるようにします。アイテムが共有されると、受信者は通知を受け取り、アイテムは受信者の [自分と共有] ページに追加されます。この設定が有効でない場合、ユーザーは共有するリンクのみをコピーできます。詳細については、Tableau ユーザーヘルプの「Web コンテンツの共有」を参照してください。
コメント 複数サイトサーバーの [サイト	ユーザーが各ビューの [コメント] サイドペインにコメントを追加し、@mentions で他の Tableau ユーザーにメールで通知できるかどうかを制御します。詳細については、Tableau ユー

設定] に移動します。	ザー ヘルプの「ビューのコメント」を参照してください。
データ主導アラート 複数サイト サーバーの [サイト設定] に移動します。	データが重要なしきい値に到達したときに、ユーザーが自動でメールを受信できるようにします。詳細については、Tableau ユーザー ヘルプのデータドリブン アラートの送信に関するページを参照してください。
タグ付け 複数サイト サーバーの [サイト設定] に移動します。	ユーザーがアイテムに追加できるタグの数を指定します。既定の制限は 50 タグで、最大は 200 タグです。詳細については、「タグの使用」を参照してください。
ビューの推奨事項 複数サイト サーバーの [サイト設定] に移動します。	推奨事項をサイトに表示するかどうか、および推奨アイテムを見たユーザーの名前を推奨ツールヒントに表示するかどうかを制御します。 **注:** Tableau Server を使用している場合、管理者が推奨を無効にすることができます。
アクセス権の要求 複数サイト サーバーの [サイト設定] に移動します。	ユーザーはコンテンツまたはプロジェクトの所有者にアクセス権の要求を送信できます。詳細については、Tableau Cloud ヘルプの「サイト ユーザーにコンテンツへのアクセスを要求できるようにする」を参照してください。
クロスデータベース結合 複数サイト サーバーの [サイト設定] に移動します。	複数のソースからのデータを結合するときに、結合プロセスが実行される場所を指定します。詳細については、Tableau ユーザーヘルプの「異なるデータベースのテーブルを組み合わせる」を参照してください。
メトリクス コンテンツのタイプ 複数サイト サーバーの [サイト設定] に移動します。	サイトでメトリクスを使用できるかどうかを制御します。有効にすると、ユーザーはビューからメトリクスを作成でき、メトリクスはコンテンツ タイプとして表示されます。無効にすると、メトリクスはサイトに表示されないか、同期を継続します。ただし、この機能をもう一度有効にすると、以前に作成したメトリクスを再び表示できます。詳細については、Tableau Cloud ヘル

	プまたは Tableau Server ヘルプの「メトリクスのセットアップ」を参照してください。 **従来のメトリクス機能の廃止** Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。
サイトにおける抽出のタイムゾーン 複数サイトサーバーの [サイト設定] に移動します。	サイトにおける抽出ベースのデータソースの既定タイムゾーンは、協定世界時 (UTC) です。サイト管理者は、別のタイムゾーンを設定できます。詳細については、Tableau Cloud ヘルプの「サイトのタイムゾーンを抽出用に設定する」を参照してください。
今すぐ実行 複数サイトサーバーでは、これはサーバー設定 (ページ下部) と [サイト設定] の両方に表示されます。	サーバー管理者はこの設定を使用して、ユーザーが抽出更新、フロー実行、サブスクリプションを手動で実行することを許可またはブロックできます。この設定は、サーバーレベルで適用して Tableau Server 上のすべてのサイトを含めたり、サイトレベルで適用して特定のサイトのみを含めたりすることができます。 • 既定では、このオプションはユーザーが手動でジョブを実行できるように設定されています。ユーザーによるジョブの手動実行を防止するには、このチェックボックスをオフにします。 • これは、ユーザーが Web インターフェース、REST API 呼び出し、tabcmd コマンドによって手動で開始した

	ジョブにのみ適用されます。スケジュールされたタスクによって開始されたジョブは、引き続きスケジュール時刻に実行され、影響を受けません。 • **[今すぐ実行]** を選択すると、ユーザーは Web 上のデータ ソースの接続タイプ (ライブ/抽出) を変更できます。
通知の管理 複数サイト サーバーの [サイト設定] に移動します。	抽出ジョブ、フローの実行、または他のユーザーによるコンテンツやコメント内のメンションの共有などのイベント通知を、サイト ユーザーが受信する方法を制御します。通知は、通知センターを介して Tableau サイトに表示するか、メールで送信できます。通知が有効になっている場合、ユーザーは [アカウント設定] ページで通知設定を構成できます。
フロー サブスクリプション 複数サイト サーバーの [サイト設定] に移動します。	フローの所有者がフローの出力データを含むメールを、自分自身や他のユーザー宛てにスケジュールに基づいて送信できるかどうかを制御します。フロー サブスクリプションを許可する場合、フロー出力データをサブスクリプション メールに含めるかどうか、およびフロー出力ファイルをメールに添付するかどうかを制御することができます。詳細については、「フロー実行の成功をユーザーに通知する」を参照してください。
Web ページ オブジェクト 複数サイト サーバーの [サイト設定] に移動します。	これらのダッシュボード オブジェクトでターゲット URL を表示できるかどうかを制御します。詳細については、Tableau ユーザー ヘルプの Web ページ オブジェクトのセキュリティに関するページを参照してください。
[パーソナル スペース] 複数サイト サーバーの [サイト設定] に移動します。	Creator および Explorer サイトのユーザーがコンテンツを作成してプライベート パーソナル スペースに保存できるようにします。パーソナル スペースがオンになっている場合、ユーザーのストレージ制限を設定できます。詳細については、パーソナル スペースでプライベート コンテンツを作成および編集を参照してください。
コレクション 複数サイト サーバーの [サイト設定] に移動します。	コレクションを利用できるかどうかを制御します。コレクションをオンにすると、ユーザーはコンテンツを整理するためのコレクションを作成したり、他のユーザーが利用できるコレクションを閲覧したりすることができます。詳細については、「コレクショ

	ン内のアイテムを整理する」を参照してください。
OAuth クライアント レジストリ	コネクタのサブセットの場合、サイトのカスタム OAuth クライアントを登録すると、サーバー用に構成された OAuth クライアントを上書きできます。カスタム OAuth クライアントを登録することにより、新規および既存の接続で、サーバー全体の OAuth クライアントの代わりにサイトレベルの OAuth クライアントを使用できるようになります。詳細については、サイトのカスタム OAuth の設定を参照してください。
ビューの高速化	Creator および Explorer のサイト ユーザーが、Tableau Server ワークブックのビューをアクセラレートして読み込み時間を短縮できるかどうかを制御します。ビューの高速化を許可する場合、高速化するビューの最大数を設定し、高速化タスクが繰り返し失敗するビューの高速化を自動的に停止するように選択できます。詳細については、「ビューの高速化」を参照してください。
スタート ページ	すべてのユーザーに対し、サーバーの現在の既定のスタート ページにリンクします。既定のスタート ページを変更する方法の詳細については、すべてのユーザーの既定のスタート ページの設定を参照してください。個々のユーザーがこの設定を上書きできます (詳細については、「プロファイルおよびアカウント設定へのアクセス」を Tableau Server ヘルプで検索してください)。
ゲスト アクセス	Tableau Server アカウントにサインインしなくても埋め込みビューの表示と操作を許可します。ゲスト ユーザー アカウントにパーミッションを割り当て、各ビューに対して許可する操作を制御できます。このオプションはコア ベースのサーバー ライセンスでのみ使用可能です。
コンテンツ内の埋め込み認証資格情報	• パブリッシャーに対し、データ ソース、フロー、またはワークブックへの認証資格情報の埋め込みを許可する パブリッシュされたワークブックまたはフローにパスワード

	を付けることで、データソースに接続するWebユーザーを自動的に認証することをパブリッシャーに許可します。パスワードはワークブックまたはフローの入力時に追加され、サーバーからのみアクセス可能です。たとえば、ワークブックをTableau Desktopで開くときは、引き続きユーザー名とパスワードを入力してデータソースに接続する必要があります。この設定をオフにすると、既存の埋め込みパスワードはすべて保存されますが、認証には使用されません。この設定を再びオンにしても、ユーザーはパスワードを再度埋め込む必要はありません。 • パブリッシャーに対し、フローの実行や抽出の更新スケジュールの設定を許可する パブリッシャーがワークブックやフローをスケジュールに割り当てることを許可します。このオプションは、**[パブリッシャーにデータソース、フロー、またはワークブックへの認証資格情報の埋め込みを許可する]** が有効になっている場合にのみ使用できます。この設定を有効にすると、パブリッシャーには [パブリッシュ] ダイアログボックスでスケジュール オプションが表示されます。
サインインのカスタマイズ	サーバー サインイン ページにカスタム注記を追加します。[サインイン] 設定でテキストを追加できます。オプションでURLを追加し、テキストをリンクにすることができます。このメモは、ユーザーがサインイン エラーを受け取った場合にも表示されます。 カスタム ノートは Tableau Mobile では表示されません。Tableau Server がアイデンティティプールを使用して構成されている場合、サインイン カスタマイズ ノートは、すべてのサインイン オプションの下にあるTableau Server のランディング

	ページと、最初のプール (TSM で構成済み) ユーザーがユーザー名とパスワードを入力するページの両方に表示されます。 カスタム注記を設定するには、Tableau Server のサイトにサインインします。左側のナビゲーション ペインで、ドロップダウンのサイト一覧から**[すべてのサイトの管理]** を選択します。**[設定]** を選択し、**[サインインのカスタマイズ]** にメッセージを追加します。 詳細については、サーバーのカスタマイズを参照してください。
推奨トレーニングのスケジュール	推奨事項には、データソースとテーブルの推奨事項 (Tableau Desktop の場合) とビューの推奨事項 (Tableau Server の場合) の 2 つの形式があります。推奨事項は、コンテンツの人気度、あるいは現在のユーザーと似た別のユーザーが使用しているコンテンツに基づいて提案されます。 トレーニング スケジュールは、推奨事項を最新に保つために、新しいコンテンツと新しい利用状況の情報をサーバーが確認する頻度を制御します。新しいコンテンツには、新規および更新済みのデータソースおよびワークブックが含まれます。新しい利用状況の情報には、"Laura Rodriquez が Food Catering データソースを使用" や "Henry Wilson が Monthly Sales ビューにアクセスした" といった情報が含まれます。 サーバーのパフォーマンスに影響がある場合は、サーバーの負荷が低いときにこのプロセスが実行されるようにスケジュールしてください。パフォーマンスへの影響を追跡するには、抽出以外のバックグラウンド タスク ビューの[推奨トレーナー]または[ビュー推奨トレーナー] タスクを探します。 推奨事項を無効化する場合は、tsm configuration set オプ

	ションの「recommendations.enabled」を false にします。ビューの推奨事項のみを無効にする場合は、オプション "recommendations.vizrecs.enabled" を false に変更します。
接続されたクライアント	モバイル ユーザーが Tableau Server に接続する度に毎回サインインして認証資格情報を提供する必要があるかどうか、またはデバイスが初回に正しく認証された後は認証資格情報を入力しなくてもデバイスを Tableau Server に接続できるかをコントロールします。詳細については、「自動クライアント認証を無効にする」を参照してください。
言語とロケール	サーバーのユーザー インターフェイスで使用する言語やビューに使用するロケールを制御します。ユーザーは自分の [アカウント設定] ページでこの設定をオーバーライドできます。なお、言語とロケールの決定に使用される最初の設定は Web ブラウザーの設定です。詳細については、Tableau Server の言語とロケールを参照してください。
データ ソースの保存済み認証資格情報	• ユーザーに対し、データ ソースのパスワードの保存を許可する ユーザーに [パスワードを保存する] を選択し、複数のアクセスとブラウザー、デバイスにわたってデータ ソース パスワードを保存できるように許可します(既定では、ユーザーは「ログアウトするまでパスワードを保存する」ように選択でき、1 回のブラウザー セッションの間のみ、パスワードを保存することができます)。 管理者はいつでも、すべての保存済みパスワードをクリアすることができます。また、ユーザーは自分の保存済みパスワードをクリアできます。 • ユーザーに対し、データ ソースの OAuth アクセス トークンの保存を許可する

	ユーザーによるアクセス トークンのユーザー設定への保存を許可します。アクセス トークンは OAuth 接続をサポートするクラウド データ ソースによって提供されます。また、アクセス トークンはデータへのアクセスを許可するためにユーザー名とパスワードの代わりに使用されます。
リンクされたタスク 複数サイト サーバーでは、これは [サーバー設定] と[サイト設定] の両方に表示されます。	サーバー管理者はこの設定を使用して、フロー タスクを順に実行するスケジュール設定をユーザーに許可します。また、ユーザーが **[今すぐ実行]** を使用して、スケジュールされたフロー タスクをトリガーして実行できるようにすることも可能です。 この設定は、サーバー レベルで適用して、Tableau Server 上のすべてのサイトを含めることができます。この設定は、サイト レベルで無効にして、特定のサイトのみを含めることができます。 リンクされたタスクをスケジュールした後にサーバー設定をオフにした場合、実行中のタスクはすべて完了しますが、スケジュール済みのリンクされたタスクは非表示になり、**[スケジュールされたタスク]** タブに表示されなくなります。 詳細については、「リンクされたタスクのスケジュール」を参照してください。
フロー パラメーター 複数サイト サーバーでは、これは [サーバー設定] と[サイト設定] の両方に表示されます。	ユーザーは、パラメーターを含むフローをスケジュールして実行できます。 管理者は、フロー パラメーターで任意の値が受け入れられるようにすることもできます。このオプションを有効にすると、フロー ユーザーはパラメーターに任意の値を入力できるため、ユーザーがアクセスすべきではないデータがパブリッシュされる可能性があります。

	パラメーターは、ファイル名、パス、テーブル名のインプット ステップ、または カスタム SQL クエリを使用する場合は、ファイル名、パス、テーブル名のアウトプット ステップ、およびフィルター または計算値の任意のステップ タイプに入力できます。 フロー パラメーターの設定は、サーバー レベルで適用して、Tableau Server 上のすべてのサイトを含めることができます。この設定は、サイト レベルで無効にして、特定のサイトのみを含めることができます。 パラメーターを使用する方法についての詳細は、Tableau Prep ヘルプの「フローでパラメーターを作成して使用する」を参照してください。
Active Directory の同期 サーバーが AD アイデンティティ ストア用に構成されている場合にのみ表示されます。	**Active Directory グループを定期的に同期する**オプションを選択した後で、指定したスケジュールに基づいて Tableau Server 内のすべての Active Directory の同期をコントロールします。詳細については、サーバー上のすべての外部ディレクトリグループの同期を参照してください。
既定の設定へのリセット	セットアップ後に変更された上記のサーバー設定がすべて元の状態に戻ります。
グループ メンバーシップのアサーション 複数サイトのサーバーでは、これは [サーバー設定] と[サイト設定] の両方に表示されます。	ユーザーが Tableau Server に認証されるときにグループ メンバーシップを動的にアサートすることにより、ローカル グループのメンバーシップを IdP または接続済みアプリを通じて制御および管理できるようになります。SAML アサーションまたは JSON Web トークン (JWT) で追加の設定が必要です。詳細については、アサーションを使用した動的グループ メンバーシップを参照してください。 **注:** サイト レベルの設定を有効にするには、サーバー全体の設定を有効にしておく必要があります。
グループ セット	**[グループ セット]** ページと、グループ セットを作成する機能を

複数サイトサーバーの [サイト設定] に移動します。	有効にします。グループセットは、特定のユーザー (サーバー管理者、サイト管理者、プロジェクト所有者、コンテンツ所有者) が使用できます。グループセットを使用するとパーミッションルールを適用することができ、パーミッションがグループセットに基づいているコンテンツにアクセスするには、ユーザーがグループセット内のすべてのグループのメンバーである必要があります。詳細については、「グループセットの操作」を参照してください。

カスタマイズ

設定	説明
ウェルカムバナー	すべてのサーバーユーザーに表示するホームページのウェルカムバナーにカスタムメッセージを追加します。カスタムメッセージには、最大 120 文字のテキストとハイパーリンク、および段落区切りを 1 つ含めることができます。管理者は、サーバーの既定の Tableau ウェルカムバナーを無効にすることもできます。 詳細については、サーバーのカスタマイズを参照してください。

モバイル

設定	説明
Tableau Mobile	
アプリロック	ユーザーがこのサイトを Tableau Mobile で開くには、生体認証方式またはデバイスのパスコードが必要です。詳細については、「Tableau Mobile 展開ガイド」の「アプリロックを有効にしてセキュリティを強化する」を参照してください。
オフラインプレビュー	ユーザーが Tableau Mobile のサイトにアクセスしたときに、オ

	フラインプレビューを生成して表示するかどうかを制御します。詳細については、「Tableau Mobile 展開ガイド」の「デバイスでの Tableau Mobile データの管理」を参照してください。
モバイル セキュリティ ポリシー バージョン 2023.1.0 の追加	一部のセキュリティポリシーは自動的に有効になり、無効にすることはできません。モバイル セキュリティポリシーは、Tableau Mobile の MAM バージョンでは利用できません。 **注:** サイトレベルで構成されたモバイル セキュリティポリシーは、サーバーレベルのモバイル セキュリティポリシーよりも優先順位が高くなります。
ジェイルブレイク検出	「ジェイルブレイク」または「ルート化」されたデバイスを使用している Tableau Mobile アプリユーザーが Tableau のコンテンツにアクセスできるかどうかに加えて、ジェイルブレイクまたはルート化されたデバイスが検出されたときに発生する応答のレベルを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。
マルウェア検出 (Android のみ)	モバイル デバイスでマルウェア検出を有効にするかどうかに加えて、マルウェアが検出されたときに発生する応答のレベルを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。
ポリシーの更新を行わないオフラインの最大日数	モバイル デバイスがオフラインの状態でアプリを使用できる最大日数を設定するかどうかを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。
デバッグの防止	モバイル デバイスでデバッガーを防止するかどうかを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。
画面共有とスクリーンショット (Android のみ)	Tableau Mobile ユーザーがアプリ内でスクリーンショットを撮ったり、画面共有を使用したりできるかどうかを制御します。詳細については、「Tableau Mobile アプリのセキュリティ設定」を参照してください。

Tableau Server PC の停止または再起動

ベストプラクティスとして、Tableau Server を実行中の PC を停止または再起動する際は、その前に Tableau Server を必ず停止してください。これは、仮想マシン (VM) や専用ハードウェアで Tableau を実行している場合にも当てはまります。Tableau Server を停止せずに PC を停止することが決してないようにしてください。Tableau を実行したまま PC をシャットダウンすると、Tableau Server の再起動時に問題が発生する可能性があり、予期しない結果が生じることもあります。

安全のため、システムをシャットダウンするスクリプトプロセスを使用する場合、または PC を手動でシャットダウンする場合のいずれも、次の手順に従ってください。

1. Tableau Server を停止します。

   これを実行するには、コマンドラインで `tsm stop` コマンドを使用するか、TSM Web UI で **[Tableau Server が実行中です]** をクリックして **[Tableau Server の停止]** を選択します。

   > **注:** 一部の TSM プロセスは Tableau Server の停止後も引き続き実行されます。これは正常であるため、引き続き PC を停止できます。これらの実行中のサービスは、PC のシャットダウン時に自動的に停止するように設計されています。

2. Tableau が停止したら、PC を停止します。

3. 準備ができたら、PC を再起動します。これは、計画されたメンテナンスの完了後、または一定期間のダウンタイムを確保するために PC をオフのままにした後で行われる場合があります。

4. Tableau Server を起動します。

   これを実行するには、コマンドラインで `tsm start` コマンドを使用するか、TSM Web UI で **[Tableau Server が停止しています]** をクリックして **[Tableau Server の起動]** を選択します。

   TSM は PC の起動時に自動的に起動するため、Tableau Server が停止していても tsm コマンドを実行できます。

tsm コマンド ライン リファレンス

このセクションのトピックには、Tableau Server をサポートする Tableau サービス マネージャー (TSM) コマンド ライン インターフェイス (CLI) の参照コンテンツが含まれています。

TSM は、Tableau Server のインストールおよび構成の管理に使用されます。TSM の詳細については、Tableau Services Manager の概要を参照してください。

TSM API を使用して、TSM CLI によってサポートされているインストールと構成を自動化できます。プレリリース (アルファ版) TSM API の詳細については、「Tableau サービス マネージャー API」を参照してください。

> Tableau Server on Windows 用の tsm コマンドについては、「tsm コマンド」を参照してください。

tsm CLI の使用

tsm コマンドは、第一ノード (TSM がインストールされているノード)、またはクラスター内の任意の追加ノードで実行できます。

tsm コマンドを実行するには、コマンドプロンプトを開く必要があります。

1. クラスター内のノード上で、`tsmadmin` グループのメンバー アカウントを使用してコマンドプロンプトを開きます。
2. 必要なコマンドを実行します。第一ノード以外のノードからコマンドを実行する場合は、`-s` オプションを含めて第一ノードの URL を名前 (IP アドレスではない) で指定し、TSM ポートの 8850 を含めます。

   第一ノードから TSM と Tableau Server のバージョンを確認するには、次のコマンドを実行します。

   ```
   tsm version
   ```

追加のノードからTSM とTableau Server のバージョンを確認するには、次のコマンドを実行します。

```
tsm version -s https://<inital_node_name>:8850
```

例:

```
tsm version -s https://myTableauHost:8850
```

tsm CLI での認証

Tableau Server 2019.2 リリースから、次の条件が該当する場合に、Tableau サービスマネージャーコマンドを実行する必要はありません。

- コマンドを実行しているアカウントがTSM承認リストグループのメンバーであり、既定では tsmadmin グループである。Tableau 権限なしユーザー(既定では、tableau ユーザー)および root アカウントも Tableau サービスマネージャーコマンドを実行している。
- Tableau Server 管理コントローラー サービスを実行しているTableau Server でローカルでコマンドを実行している。既定では、Tableau Server 管理コントローラー サービスが、分散展開の最初のノードにインストールされ構成されます。

Tableau サービスマネージャー CLI へのローカルでのログイン

Tableau サービスマネージャー承認グループのメンバーであるユーザーアカウントを使用して、ローカルコンピューターで Tableau サービスマネージャーコマンドを実行している場合、パスワードを指定する必要はありません。この場合は、たとえば、次のようにコマンドを実行します。

```
tsm version
```

tsm CLI へのリモートでのログイン

Tableau Server 管理コントローラー サービスが実行されていないクラスタ内のノードからTSM コマンドを実行している場合は、コマンドを実行する前に、リモートコンピューターの Tableau Server 管理コントローラー サービスとのセッションを認証する必要があります。たとえば、次のコマンドを実行します。

```
tsm login -s <server_name> -u <account_name>
```

ここで、<server_name> は Tableau Server 管理コントローラー サービスが実行されているノードの名前であり、<account_name> は TSM 承認グループのメンバーであるアカウントです。

このコマンドを実行すると、パスワードの入力を求められます。アカウントが認証されたら、Tableau サービスマネージャーコマンドを実行できます。

セキュリティのベストプラクティスとして、TSM ポート(既定では 8850) をインターネットに公開しないでください。

Tableau サービスマネージャー承認グループへのアカウントの表示と追加

サーバーのインストール中に、Tableau サービスマネージャー承認グループが作成されます。既定では、tsmadminという名前のTableau サービスマネージャー承認グループです。インストール中に代替のTableau サービスマネージャー認可グループを作成した場合は、次のコード例での tsmadmin のグループ名を置き換えます。

tsmadmin グループのユーザー アカウントを表示するには、次のコマンドを実行します。

```
grep tsmadmin /etc/group
```

ユーザーを tsmadmin グループに追加するには:

```
sudo usermod -G tsmadmin -a <username>
```

Tableau サービスマネージャー CLI でのスクリプト作成と自動化

スクリプトファイル内のパスワードを使用せずに Tableau Server でオートメーションを実行するには、最初のノードおよび適切なTableau サービスマネージャー承認グループのアカウントでスクリプトを実行します。詳細については、上記の「認証」セクションを参照してください。

シェルでヘルプ コンテンツを表示する

コマンド ラインから最小限のヘルプ コンテンツを表示するには、tsm help カテゴリを使用します。

シノプシス

```
tsm help [category] [command]
```

コマンド

`tsm help`

すべての tsm コマンドのヘルプ。

`tsm help <category>`

特定のコマンド *カテゴリ*のヘルプを表示します。たとえば、`tsm help authentication` のように使用します。

`tsm help <category> <command>`

特定のコマンドのヘルプを表示します。たとえば、`tsm help authentication open-id` のように使用します。

`tsm help commands`

上位のすべてのコマンドまたは *カテゴリ*をリストします。

カテゴリ

tsm authentication

`tsm authentication` コマンドを使用して Tableau Server のユーザー認証オプションを有効、無効、および設定します。

- identity-migration
- kerberos
  - configure
  - disable
  - enable

- legacy-identity-mode
  - enable
  - disable
- list
- mutual-ssl
  - configure
  - disable
  - enable
- openid
  - configure
  - disable
  - enable
  - get-redirect-url
  - map-claims
- pat-impersonation
  - disable
  - enable
- saml
  - configure
  - disable
  - enable
  - export-metadata
  - map-assertions
- sitesaml
  - disable
  - enable
- sspi
  - disable
  - enable
- trusted

tsm authentication identity-migration configure

アイデンティティの移行の既定のジョブ設定を変更します。詳細については、「アイデンティティの移行を管理する」を参照してください。

シノプシス

```
tsm authentication identity-migration configure –job-run-time

tsm authentication identity-migration configure –rate
```

オプション

`-j,--job-run-time <number>`

オプション。

スケジュールされた移行ジョブを実行できる許容最長時間を分単位で決定します。既定値は 120 分 です。

`-r,--rate <number>`

オプション。

移行ジョブ中に 1 秒あたりに実行できる要求の数を決定します。既定値は 1 秒あたり5個です。

tsm authentication kerberos <commands>

Tableau Server で Kerberos ユーザー認証を有効、無効、および設定します。Kerberos の構成を参照してください。

シノプシス

```
tsm authentication kerberos configure --keytab-file <keytab_
file.keytab> [global options]

tsm authentication kerberos enable [global options]

tsm authentication kerberos disable [global options]
```

kerberos configure のオプション

`-kt, --keytab-file <keytab_file.keytab>`

必須。

KDC へのリクエストに使用されるサービスの .keytab ファイルを指定します。

tsm authentication legacy-identity-mode <commands>

アイデンティティの移行中に必要になる可能性のある従来のアイデンティティストア モードを有効または無効にします。このコマンドをいつ使用するかを判断するには、「バックアップを復元できません」セクションを参照してください。

詳細については、「アイデンティティの移行について」を参照してください。

シノプシス

```
tsm authentication legacy-identity-mode enable
```

```
tsm authentication legacy-identity-mode disable
```

tsm authentication list

サーバーの認証に関連する既存の構成設定をリストします。

シノプシス

```
tsm authentication list [--verbose][global options]
```

オプション

`v, --verbose`

オプション。

すべての構成済みパラメーターを表示します。

tsm authentication mutual-ssl <commands>

Tableau Server でユーザー認証の相互 SSL を有効、無効、および設定します。相互 SSL の詳細については、相互 SSL 認証の構成を参照してください。

相互 SSL を有効にする前に、外部通信の SSL を有効および設定しておく必要があります。詳細については、Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成するを参照してください。

シノプシス

```
tsm authentication mutual-ssl configure [options] [global options]

tsm authentication mutual-ssl disable [global options]

tsm authentication mutual-ssl enable [global options]
```

オプション

`-cf, --ca-cert <certificate-file.crt>`

オプション。

証明書ファイルの場所およびファイル名を指定します。ファイルは認証局からの有効な信頼できる証明書である必要があります (Verisign など)。

`-fb, --fallback-to-basic <true | false>`

オプション。

SSL 認証に失敗する場合、Tableau Server で認証用にユーザー名およびパスワードを受け入れる必要があるかどうかを指定します。

既定値は false です。これは、相互 SSL を設定する際に SSL 認証に失敗すると Tableau Server で接続が許可されないことを示します。ただし、このオプションが `false` に設定されている場合でも、Tableau Server は REST API クライアントからのユーザー名とパスワード認証を受け付けます。

`-m, --user-name-mapping <upn | ldap | cn>`

オプション。

ユーザー名構文 (UPN、LDAP、CN) を指定してアイデンティティストアまたはディレクトリから取得します。構文は、ユーザー証明書のサブジェクトまたはサブジェクトの別名と形式が一致する必要があります。

`-rf, --revocation-file <revoke-file.pem>`

オプション。

証明書失効リストファイルの場所とファイル名を指定します。このファイルは .pem または .der ファイルに設定できます。

tsm authentication open-id <commands>

Tableau Server で OpenID Connect (OIDC) ユーザー認証を有効、無効、および設定します。

シノプシス

```
tsm authentication openid configure [options] [global options]

tsm authentication openid disable [global options]

tsm authentication openid enable [global options]

tsm authentication openid get-redirect-url [global options]

tsm authentication openid map-claims [options] [global options]

tsm authentication openid enable-slo [options][global options]

tsm authentication openid disable-slo [options][global options]

tsm authentication openid get-slo-redirect-url [options]
[global options]
```

注

- Tableau Server 2025.3 に `tsm authentication openid enable-slo`, `tsm authentication openid disable-slo` および `tsm authentication get-slo-redirect-url` コマンドが追加されました

openid configure のオプション

> **注:** オプションは、初期構成時または再構成時に設定する必要があります。個々のオプションを個別に設定することはできません。

`-a, --client-authentication <string>`

必須。

OpenID Connect のカスタム クライアント認証手法を指定します。

Salesforce IdP を使用するように Tableau Server を設定するには、この値を `client_secret_post` に設定します。

`-cs, --client-secret <string>`

必須。

プロバイダー クライアント シークレットを指定します。これは、Tableau が IdP からの応答の真偽を検証するために使用するトークンです。この値は秘密で、保護しておく必要があります。

`-cu, --config-url <url>`

必須。

OpenID プロバイダーのメタデータを含むプロバイダー構成のディスカバリドキュメントの場所を指定します。プロバイダーが公開 JSON ファイルをホストしている場合は、-config-url を

使用します。そうでない場合は、代わりに --metadata-file を使用して、ローカル コンピューター上のパスとファイル名を指定します。

`-mf, --metadata-file <file-path>, --config-file <config-file.json>`

オプション。

静的な OIDC ディスカバリJSON ドキュメントへのローカル パスを指定します。

`-i, --client-id <client-id>`

必須。

IdP がアプリケーションに割り当てているプロバイダー クライアントID を指定します。

`-id, --ignore-domain <true | false>`

オプション。既定:`false`

以下に当てはまる場合は、これを`true` に設定します。

- Tableau Server で電子メール アドレスをユーザー名として使用している
- IdP のユーザーを複数のドメイン名を使用してプロビジョニングしました
- IdP からの `email` 要求のドメイン名部分を無視する場合があります

先に進む前に、このオプションを`true` に設定した結果として使用されるユーザー名を見直します。ユーザー名の競合が発生する場合があります。ユーザー名が競合したとき、情報開示のリスクは高くなります。OpenID Connect の使用要件を参照してください。

`-if, --iframed-idp-enabled <true | false>`

オプション。既定:`false`

iFrame 内で IdP を許可するかどうかを指定します。iFrame での表示を許可する場合は、IdP でクリックジャック保護を無効にする必要があります。

`-ij, --ignore-jwk <true | false>`

オプション。既定:`false`

お使いの IdP が JWK 検証をサポートしていない場合は、これを `true` に設定します。この場合、相互 TLS または別のネットワークレイヤー セキュリティプロトコルを使用して IdP との通信を認証することをお勧めします。

`-r, --return-url <return-url>`

サーバーの URL です。一般的には、サーバーのパブリック名 (`"http://example.tableau.com"` など) になります。

`-sn, --custom-scope-name <string>`

オプション。

IdP のクエリに使用できるカスタム範囲のユーザー関連値を指定します。OpenID Connect の使用要件を参照してください。

openid map-claims のオプション

このオプションは、IdP との通信時に Tableau Server で使用する既定の OIDC クレームを変更するときに使用します。OpenID Connect の使用要件を参照してください。

`-i, --id <string>`

オプション。既定: `sub`

お使いの IdP が ID トークンのユーザーを一意に識別するときに `sub` クレームを使用しない場合は、この値を変更します。指定する IdP クレームには、単一かつ一意の文字列を含める必要があります。

`-un, --user-name <string>`

オプション。既定: `email`

この値を、組織が Tableau Server に保存されているユーザー名の照合に使用する IdP クレームに変更します。

注

- Tableau Server 2025.3 に `tsm authentication openid enable-slo`、 `tsm authentication openid disable-slo` および `tsm authentication get-slo-redirect-url` コマンドが追加されました。

tsm authentication pat-impersonation <commands>

Tableau Server の管理者の個人用アクセストークンの偽装を有効または無効にします。

個人用アクセストークンの偽装の詳細については、個人用アクセストークンを参照してください。

シノプシス

```
tsm authentication pat-impersonation disable [global options]

tsm authentication pat-impersonation enable [global options]
```

個人用アクセストークンの偽装が有効になっているかどうかを確認するには、次のコマンドを実行します。

```
tsm authentication list
```

tsm authentication saml <commands>

Tableau Server を構成して、SAML 2.0 標準を使用したシングルサインオンのサポート、サイトの SAML の有効化/無効化、および Tableau Server とアイデンティティプロバイダー (IdP) の間のアサーション属性名のマップを実行します。

利用可能なコマンド

```
tsm authentication saml configure [options] [global options]

tsm authentication saml disable [options] [global options]

tsm authentication saml enable [options] [global options]
```

```
tsm authentication saml export-metadata [options] [global options]

tsm authentication saml map-assertions [options]
```

tsm authentication saml configure

サーバーの SAML 設定を構成します。SAML 証明書およびメタデータファイルを指定して、追加の必要情報を提供し、追加のオプションを設定します。

SAML を初めて構成する場合、または無効にしたことがある場合は、このコマンドを `tsm authentication saml enable` と一緒に実行する必要があります。詳細については、サーバー全体の SAML の構成を参照してください。

シノプシス

```
tsm authentication saml configure [options] [global options]
```

オプション

`-e, --idp-entity-id <id>`

初回 SAML 構成では必須。その他の場合オプション。IdP エンティティ ID 値です。

通常これは Tableau Server のリターン URL (`--idp-return-url` パラメーターで指定される) と同じです。入力するエンティティ ID は、サイト固有のエンティティ ID を生成するためのベースに使用されます。たとえば、次のように入力した場合です。

http://tableau-server

SAML のために構成されたサイトは以下のエンティティ ID を表示する場合があります。

http://tableau-server/samlservice/public/sp/metadata?alias=48957410-9396-430a-967c-75bdb6e002a0

サイトのエンティティ ID を検索するには、サイトの **[設定]** ページに移動し、**[認証]** タブを選択します。SAML が有効な場合は、サイト固有の SAML を構成し、メタデータをエクスポートするための最初のステップで、エンティティ ID が表示されます。

`-r, --idp-return-url <idp-return-url>`

初回 SAML 構成では必須。その他の場合オプション。IdP で設定されている SAML リターン URL です。これは通常、https://tableau-server などの Tableau Server の外部 URL です。

注

- http://localhost は外部サーバーでは使用できません。
- URL 末尾へのスラッシュ追加 (https://tableau-server/) はサポートされていません。

`-i, --idp-metadata <idp-metadata.xml>`

初回 SAML 構成では必須。その他の場合オプション。IdP の設定からエクスポートした XML メタデータファイルの場所と名称を提供します。

たとえば、`/var/opt/tableau/tableau_server/data/saml/<metadata-file.xml>`

`-cf, --cert-file <certificate.crt>`

初回 SAML 構成では必須。その他の場合オプション。SAML 用の x509 証明書ファイルの場所およびファイル名。証明書ファイルの要件については、SAML 要件を参照してください。

たとえば、`/var/opt/tableau/tableau_server/data/saml/<file.crt>`

`-kf, --key-file <certificate.key>`

初回 SAML 構成では必須。その他の場合オプション。証明書とセットになっているキーファイルの場所と名称。

たとえば、`/var/opt/tableau/tableau_server/data/saml/<file.key>`

FIPS を使用する場合の初回 SAML 構成では必須。その他の場合オプション。暗号化証明書ファイルの場所およびファイル名。証明書ファイルの要件については、SAML 要件を参照してください。

たとえば、`/var/opt/tableau/tableau_server/data/saml/<file.crt>`

たとえば、C:\ProgramData\Tableau\Tableau Server\data\saml\<file.crt>

`-ekf, --encryption-key-file <FILE-PATH>`

FIPS を使用する場合の初回 SAML 構成では必須。その他の場合オプション。暗号化証明書と対になっている暗号化キーファイルの場所と名称。

たとえば、/var/opt/tableau/tableau_server/data/saml/<file.key>

たとえば、C:\ProgramData\Tableau\Tableau Server\data\saml\<file.key>

`-a, --max-auth-age <max-auth-age>`

オプション。既定値は -1 です (Tableau Server バージョン 2020.4.15、2021.1.12、2021.2.9、2021.3.8、2021.4.4、2022 以降)。以前の既定値は 7200 (2 時間) でした。

ユーザーの認証と AuthNResponse メッセージのプロセス間で許可される最大秒数。

この既定値を保持することをお勧めします。これにより、認証の最大有効期間のチェックが無効になります。この値が IdP と異なる場合、Tableau Server にサインインするユーザーにサインイン エラーが表示される可能性があります。このエラーの詳細については、Tableau ナレッジ ベースの記事「Tableau Server での SAML SSO の断続的なエラー "Unable to Sign In" (サインインできません)」を参照してください。

`-d, --desktop-access <enable | disable>`

オプション。既定値は enable です。

このオプションはサーバー全体の SAML にのみ適用されます。SAML を使用して Tableau Desktop からサーバーにサインインします。Tableau クライアント アプリケーションからのシングル サインオンが IdP と連携しない場合、これを `disable` に設定することができます。

`-m, --mobile-access <enable | disable>`

オプション。既定値は enable です。

Tableau Mobile アプリの古いバージョンからのサインインに SAML の使用を許可します。Tableau Mobile アプリのバージョン 19.225.1731 以上を実行しているデバイスでは、このオプションが無視されます。Tableau Mobile アプリのバージョン 19.225.1731 以上を実行しているデバイスを無効にするには、Tableau Server でクライアントログインオプションとしての SAML を無効にします。

`-so, --signout <enable | disable>`

オプション。デフォルトで有効に設定されています。

Tableau Server での SAML のサインアウトを有効または無効にします。

`-su, --signout-url <url>`

オプション。ユーザーがサーバーからサインアウトした後のリダイレクト先 URL を入力します。既定では Tableau Server のサインインページです。絶対 URL または相対 URL を指定できます。

例

```
tsm authentication saml configure --idp-entity-id https://tableau-
server --idp-metadata /var/opt/tableau/tableau_
server/data/saml/<metadata.xml> --idp-return-url https://tableau-
server --cert-file /var/opt/tableau/tableau_
server/data/saml/<file.crt> --key-file /var/opt/tableau/tableau_
server/data/saml/<file.key>
```

tsm authentication saml enable および saml disable

サーバー全体の SAML 認証を有効または無効にします。このコンテキストでは、SAML に対して有効にする全サイトおよびユーザーは単一のアイデンティティプロバイダーを通ることになります。

シノプシス

```
tsm authentication saml enable [global options]

tsm authentication saml disable [global options]
```

tsm authentication saml export-metadata

SAML IdP の構成に使用する Tableau Server の .xml メタデータファイルをエクスポートします。

シノプシス

```
tsm authentication saml export-metadata [options] [global options]
```

オプション

`-f, --file [/path/to/file.xml]`

オプション。

メタデータが書き込まれる場所とファイル名を指定します。このオプションが含まれていない場合は、`export-metadata` でファイルを現在のディレクトリに保存します。ファイル名は `samlmetadata.xml` となります。

`-o、--overwrite`

オプション。

`-f` で指定された同じ名前の既存のファイルが上書きされます。`-f` が含まれていない場合は、既定の名前の既存のファイルが上書きされます。`-f` で指定されたファイルがあるが `-o` が含まれていない場合、コマンドは既存のファイルを上書きしません。

tsm authentication saml map-assertions

IdP と Tableau Server 間の属性をマップします。IdP で各引数に指定された属性に使用する名前を指定します。

シノプシス

```
tsm authentication saml map-assertions --user-name <user-name>
[global options]
```

オプション

`-r, --user-name <user-name-attribute>`

オプション。IdP がユーザー名を保存する属性。Tableau Server ではこれは表示名と同じです。

`-e, --email <email-name-attribute>`

使用しないでください。このオプションは Tableau ではサポートされていません。

`-o, --domain <domain-name-attribute>`

オプション。IdP がドメイン名を保存する属性。このオプションは、Tableau Server コンピューターのドメインとは異なるドメインからユーザーを追加する場合に使用します。詳細については、複数のドメインを実行している場合を参照してください。

`-d --display-name <display-name-attribute>`

使用しないでください。このオプションは Tableau ではサポートされていません。

saml map-assertions の例

`tsm authentication saml map-assertions --user-name=<sAMAccountName> --domain=<FQDM>` または `tsm authentication saml map-assertions --user-name=jnguyen --domain=example.myco.com`

tsm authentication sitesaml enable および sitesaml disable

SAML 認証を許可または禁止するようサイトレベルでサーバーを設定します。サイト固有の SAML を有効にすることで、Tableau Server Web UI の **[設定] > [認証]** タブにアクセスできるようになります。**[認証]** タブには、サイト固有の SAML の構成設定が含まれます。

まだサイト固有 SAML を許可するようにサーバーを構成していない場合は、`saml configure` と共に `sitesaml enable` コマンドを使用します。詳細については、「サイト固有の SAML の構成」を参照してください。

シノプシス

```
tsm authentication sitesaml enable [global options]

tsm authentication sitesaml disable [global options]
```

tsm authentication sspi <commands>

このコマンドは Tableau Server on Windows でのみ機能します。Tableau Server on Linux で SSPI を有効にしようとすると、エラーが返されます。

Microsoft SSPI を使用して自動サインインを有効または無効にします。

認証に Active Directory を使用する場合、必要に応じて自動ログオンを有効にすることができます。自動ログオンを有効にすると、ユーザーは Microsoft SSPI を使用して Windows ユーザー名とパスワードに基づき自動的にサインインされます。これにより、シングル サインオン (SSO) のような操作が実現します。

Tableau Server で SAML、信頼できる認証、ロードバランサー、またはプロキシ サーバーを設定する予定がある場合は、SSPI を有効にしないでください。SSPI は、これらのシナリオではサポートされていません。

シノプシス

```
tsm authentication sspi disable [global options]

tsm authentication sspi enable [global options]
```

すべての認証コマンドと同様に、このコマンドを実行した後に `tsm pending-changes apply` を実行する必要があります。

tsm authentication trusted <commands>

Tableau Server でのユーザー認証用に信頼できる認証 (信頼できるチケット) を設定します。

シノプシス

```
tsm authentication trusted configure [options] [global options]
```

オプション

`-th, --hosts <string>`

オプション。

Tableau コンテンツでページをホストする Web サーバーの信頼できるホスト名 (または IPv4 アドレス) を指定します。

値が複数ある場合は、コンマ区切りリスト形式で名前を入力し、各値を二重引用符で囲みます。

例:

```
tsm authentication trusted configure -th "192.168.1.101",
"192.168.1.102", "192.168.1.103"
```

または

```
tsm authentication trusted configure -th "webserv1",
"webserv2", "webserv3"
```

`-t, --token-length <integer>`

オプション。

信頼できるチケットごとの文字数を決定します。既定設定の 24 文字は、ランダム性のある 144 ビットで提供されます。値は、9 から 255 (包含) の間の任意の整数に設定できます。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm configuration

tsm configuration コマンドを使用して、構成キー値を取得、設定および更新できます。

「不明なキー」の応答

特定のコンフィギュレーション キーは、現在の値を取得しようとしたり、新しい値を設定しようとすると「不明なキー」の応答を返します。この場合は、正しい大文字と大文字設定を含め、キーのスペルが正しいことを確認します。値を変更するには、`tsm configuration set`コマンドの`--force-keys`オプションを使用します。変更可能な具体的な構成キーについては、tsm configuration set のオプションを参照してください。

"NULL" 値の応答

構成キーによっては特定の既定値を持つ場合がありますが、現在の値を取得しようとすると "NULL" 応答が返されます。これらのキーは、Tableau Server コードから派生した既定値を使用します。キーが tsm configuration set のオプションに特定の既定値を持つキーとしてリストされていて、`tsm configuration get` コマンドが現在の値に対して "NULL" 値を返す場合、既定値は、Tableau Server を実行しているコードによって決定されます。キー値は `tsm configuration set` を使用して設定できます。ただし、値を変更する場合以外は、これを行う必要はありません。

- tsm configuration get
- tsm configuration list-dynamic-keys
- tsm configuration set

tsm configuration get

現在のサーバー構成とトポロジを表示します。

シノプシス

```
tsm configuration get --key <config.key> [global options]
```

オプション

`-k, --key`

必須。

指定した構成キーの現在の値を取得します。

tsm configuration list-dynamic-keys

(Tableau Server を再起動せずに) 動的に構成できるすべてのコンフィギュレーション キーを表示します。

シノプシス

```
tsm configuration list-dynamic-keys [global options]
```

tsm configuration set

サーバー構成またはトポロジを設定するか、インポートします。

スペースがない限り<config.key> と<config_value> を囲む引用符は任意で、この場合には引用符がキーまたは値を囲むように使用する必要があります。

注: 構成キーの値を設定後、`tsm pending-changes apply` を使用して保留中の構成の変更を適用する必要があります。適用するまでは、新しい値は Tableau で使用されない、または `tsm configuration get` コマンドの結果に表示されます。`tsm pending-changes list` を使用すると、保留中の変更を表示できます。詳細については、tsm pending-changesを参照してください。

シノプシス

```
tsm configuration set --key <config.key> --value <config_value>
[global options]
```

オプション

`-k,--key <config.key>`

必須。

構成キー。

`-v,--value <config_value>`

必須。3 月のメンテナンス リリース (バージョン 2021.2.10、2021.3.9、2021.4.5) 以降、このオプションを含めない場合は、値の入力を求めるプロンプトが表示されます。

構成値。

`-d`

オプション。

構成値を既定にリセットします。

`-frc, --force-keys`

オプション。

キーが以前に存在しなかった場合も、強制的に構成に追加します。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm configuration set のオプション

以下は、tsm configuration set コマンドで設定できる設定オプションまたはキーのリストです。多くの場合、この tsm configuration get コマンドを使用して、構成キーの現在の値を

確認できます。

このリストは、Tableau Server の構成設定の完全なリストではありません。サーバー管理者が設定できる構成キーの一部です。Tableau Server が内部で使用するキーの一部は、このリストに表示されません。

注: 構成キーでは、大文字と小文字が区別されます。

tsm CLI を使用する

tsm コマンドは、第一ノード (TSM がインストールされているノード)、またはクラスター内の任意の追加ノードで実行できます。

tsm コマンドを実行するには、コマンドプロンプトを開く必要があります。

1. クラスター内のノード上で、tsmadmin グループのメンバーアカウントを使用してコマンドプロンプトを開きます。
2. 必要なコマンドを実行します。第一ノード以外のノードからコマンドを実行する場合は、-s オプションを含めて第一ノードの URL を名前 (IP アドレスではない) で指定し、TSM ポートの 8850 を含めます。

   第一ノードから TSM と Tableau Server のバージョンを確認するには、次のコマンドを実行します。

   ```
   tsm version
   ```

   追加のノードから TSM と Tableau Server のバージョンを確認するには、次のコマンドを実行します。

   ```
   tsm version -s https://<inital_node_name>:8850
   ```

   例:

   ```
   tsm version -s https://myTableauHost:8850
   ```

Tsm 構成キーの基本的な使用方法

構成キーを設定する

```
tsm configuration set -k <config.key> -v <config_value>
```

場合によっては、以前に設定されていないキーの構成値を設定する `--force-keys` オプションを含める必要があります。詳細については、「不明なキー」の応答を参照してください。

構成キーの値を設定後、`tsm pending-changes apply` を使用して保留中の構成の変更を適用する必要があります。適用するまでは、新しい値は Tableau で使用されない、または `tsm configuration get` コマンドの結果に表示されます。`tsm pending-changes list` を使用すると、保留中の変更を表示できます。詳細については、tsm pending-changesを参照してください。

構成キーを既定値にリセットする

構成キーを既定値にリセットするには、以下の `-d` オプションを使用します。

```
tsm configuration set -k <config.key> -d
```

構成キーの現在の値を確認する

現在設定されている構成キーを確認するには、次のように `configuration get` コマンドを使用します。

```
tsm configuration get -k <config.key>
```

キーの有用な現在の値を返さない 2 つの特殊なケースがあります。

- 明示的に設定されていないキーの設定値を取得できない場合があります。代わりに、`tsm configuration get` コマンドは「不明なキー」応答を返します。詳細については、「不明なキー」の応答を参照してください。
- 既定値が事前定義されている特定のキーの場合、 `tsm configuration get` コマンドは "Null" 応答を返します。詳細については、"NULL" 値の応答を参照してください。

構成キー

adminviews.disabled

既定値: `false`

Tableau 管理ビューへのアクセスを無効にします。既定では、ビューへのアクセスが有効になっています (このオプションが [false] に設定されています)。

api.server.enabled

バージョン: バージョン 2023.1 で非推奨となりました。バージョン 2023.1 以降では、REST API を無効にすることはできません。

既定値: `true`

Tableau Server REST API へのアクセスを許可します。

既定では、この機能が有効になっています。この設定を維持することを強くお勧めします。REST API を無効にすると、Tableau 機能が広範囲にわたって中断されます。パフォーマンスの向上やセキュリティの強化が行われません。Tableau Server インストールで REST API を無効にすることを選択した場合は、必要な機能のテストを十分に行うようにしてください。

REST API を無効にすることにより影響を受ける機能:

- 検索
- お気に入り
- コレクション
- コンテンツ管理ツール (CMT)
- Resource Monitoring Tool (RMT)
- パーソナル スペース

auditing.enabled

既定値: `true`

PostgreSQL (Tableau Server の独自のデータベース) 履歴監査テーブルへのアクセスを許可します。

backgrounder.default_run_now_priority

既定値 (整数): 0

この設定は、「今すぐ実行」するジョブに割り当てる優先度を制御します。0 が最高の優先度です。値は 0 から 100 の範囲で指定します。

backgrounder.enable_parallel_adsync

バージョン: バージョン 2018.3.6 で追加。

既定値: `false`

複数のバックグラウンダーがある場合に、外部ディレクトリグループの同期ジョブの並列処理を許可するかどうかを制御します。既定では、外部ディレクトリグループのスケジュールされた同期は、単一のバックグラウンダーによって順番に処理されます。この値を `true` に設定すると、複数のバックグラウンダー インスタンスで並列処理が有効になります。

backgrounder.externalquerycachewarmup.enabled

バージョン: バージョン 2023.1 で非推奨となりました。Tableau Server のワークブックのビューの読み込み時間を改善するには、代わりにサイトでビューのアクセラレーションを許可してください。

既定値: `false`

スケジュールされた抽出の更新タスク後にワークブックのクエリ結果のキャッシュを制御します。

backgrounder.externalquerycachewarmup.view_threshold

バージョン: バージョン 2023.1 で非推奨となりました。Tableau Server のワークブックのビューの読み込み時間を改善するには、代わりにサイトでビューのアクセラレーションを許可してください。

既定値: `2.0`

スケジュールされた抽出の更新タスク後にワークブックのクエリ結果をキャッシュするしきい値。しきい値は、ワークブックが過去 7 日間で受信したビューの数を今後 7 日間にスケジュールされている更新の数で割った数と等しくなります。

次の 2 つの *backgrounder* コマンド オプションにより、フローのバックグラウンド タスクがキャンセルされるまでのフロー タスクの実行時間が決まります。これらの 2 つのコマンドの組み合わせで、フロー タスクの合計タイムアウト値が決まります。

backgrounder.extra_timeout_in_seconds

既定値: `1800`

`backgrounder.querylimit` の設定を超えて、バックグラウンド ジョブがキャンセルされるまでの秒数。この設定により、停止したジョブが後続のジョブを保持しないようにします。設定は、`backgrounder.timeout_tasks` にリストされているプロセスに適用されます。1,800 秒は 30 分です。

backgrounder.default_timeout.run_flow

既定値: `14400`

フロー実行タスクがキャンセルされるまでの秒数。14,400 秒は 4 時間です。

backgrounder.failure_threshold_for_run_prevention

既定値: `5`

サブスクリプションまたは抽出ジョブを連続して失敗するとジョブが中断される回数。連続して失敗するジョブを中断すると、他のジョブのバックグラウンダー リソースを保持するのに役立ちます。失敗するバックグラウンド タスクの中断を無効にするには、これを `-1` に設定します。

backgrounder.flow_failure_threshold_for_run_prevention

既定値: `5`

フロー実行ジョブが中断されるまでの連続失敗回数。連続して失敗するジョブを中断すると、他のジョブのバックグラウンダー リソースを保持するのに役立ちます。失敗したバックグラウンド タスクの中断を無効にするには、これを `-1` に設定します。

backgrounder.log.level

バージョン: バージョン 2020.3.0 で追加。

既定値: `info`

バックグラウンダープロセスのロギング レベル。このロギング レベルは動的に構成できるため、ロギング レベルだけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

backgrounder.querylimit

既定値: `7200`

1 つの抽出更新ジョブを完了するまでの秒単位の最大許容時間。7200 秒 = 2 時間。

> **注:** バックグラウンド ジョブがこの時間制限に到達しても、キャンセルされる前に数分間引き続き実行される場合があります。

backgrounder.restrict_serial_collections_to_site_level

既定値: `false`

Tableau Server では、定期的に実行する抽出更新、サブスクリプション、フローなどをユーザーがスケジューリングできます。これらのスケジュールされた項目はタスクと呼ばれます。バックグラウンダープロセスは、これらのタスクの固有のインスタンスを開始して、スケジュールされた時刻に実行します。結果として開始されたタスクの固有インスタンスは、ジョブと呼ばれます。

この設定は、順番に実行するように構成されているスケジュールに影響します。デフォルトでは、スケジュールを順番に実行するように設定している場合、そのスケジュールを使用するすべてのジョブは順番に実行されます。これを `true` に設定すると、異なるサイトで実行するジョブは並行して実行できるようになります。同じサイトでスケジュールされたタスクのジョブは、引き続き順番に実行されます。

以下の例は、次のシナリオを示したものです。

毎日午前 7 時にジョブを実行する「毎日」という名前のスケジュールが Tableau Server にあります。「毎日」スケジュールは、順番に実行するように設定されています。「人事」サイトと「給与」サイトにはそれぞれ、「毎日」スケジュールを使用する複数のタスクがスケジュールされています。この

設定が true に設定されている場合、「人事」サイトのこれらのスケジュールされたタスクのジョブは、「給与」サイトのジョブと並行して実行できますが、同じサイト上のジョブは引き続き順番に実行されます。

backgrounder.notifications_enabled

既定値: true

サーバー上のすべてのサイトで抽出更新アラートとフロー実行アラートを有効化するかどうかを制御します。既定では、アラートは有効になっています。サーバー上のすべてのサイトでアラートを無効にするには、これを false に設定します。

抽出アラートはサイト設定のサイト管理者別にサイトベースで、またはユーザー設定のユーザーレベルで有効化または無効化できます。

backgrounder.sort_jobs_by_type_schedule_boundary_heuristics_milliSeconds

既定値: 60000

同じ開始時間をスケジュールするよう判断するバックグラウンダージョブを特定する時間枠を制御します。

バックグラウンダープロセスは、最も速いカテゴリを最初に実行し、同時にジョブのタイプ別に実行するようスケジュールされた作業を命令します。サブスクリプション、増分抽出、完全抽出。

「同時」にスケジュールされているジョブを判断するため、ジョブが一括化されます。60,000 ミリ秒の値 (既定) は、1 分以内のウィンドウで開始するジョブを同じバッチに分類し、そのバッチ内でタイプ別に並べ替える必要があることを示します。

backgrounder.subscription_failure_threshold_for_run_prevention

既定値: 5

ある状態についてのアラートを一時停止する前に発生しなければいけない、連続するサブスクリプションの失敗数を決定します。既定値の 5 に設定されていると、サブスクリプションの失敗が連続 5 回発生した後に、アラートが一時停止になります。値を -1 にすると、通知メールを回数制限なし

で送信できます。このしきい値はサーバー全体に対するものであるため、サーバー上で定義されているすべてのサブスクリプションに適用されます。

backgrounder.subscription_image_caching

既定値: `true`

サブスクリプション用に生成されたイメージのキャッシュをバックグラウンダーで生成するかどうかを制御します。キャッシュによってサブスクリプションのパフォーマンスが改善されるため、キャッシュされたイメージを毎回再生成する必要はありません。既定では、イメージのキャッシュが有効になっています。サーバー上のすべてのサイトでイメージのキャッシュ作成を無効にするには、これを `false` に設定します。

backgrounder.timeout_tasks

既定値: デフォルト値は、ご利用の Tableau Server のバージョンによって異なる場合があります。ご利用の Tableau のバージョンに対するデフォルト値のリストを表示するには、`tsm configuration get` コマンドを実行します。

```
tsm configuration get -k backgrounder.timeout_tasks
```

タスクが `backgrounder.querylimit` や `backgrounder.extra_timeout_in_seconds` の組み合わされた値よりも長い時間実行する場合にキャンセルできるタスクのリスト。タスクのリストはコンマで区切られます。既定のリストは、この設定がとり得るすべての値を表します。

backgrounder.timeout.single_subscription_notify

バージョン: バージョン 2021.2 で追加。

既定値: `1800` 秒 (30 分)

単一のサブスクリプション ジョブを完了するまでの、秒単位で指定された最大許容時間です。

backgrounder.timeout.sync_ad_group

バージョン: バージョン 2021.1.23、2021.2.21、2021.3.20、2021.4.15、2022.1.11、2022.3.3、2023.1 で追加されました。

既定値: `14400` 秒 (4 時間)

これは、Active Directory のグループ同期を完了させるための最大許容時間 (秒単位) です。これはバックグラウンダー サービスによって実行されるスケジュールされたグループ同期に適用され、長時間実行される同期が無期限に実行されるのを防ぎます。これは、Tableau Server UI または REST API を使用して行われるグループ同期には影響しません。

backgrounder.vInstances_max_overflow_queue_size

バージョン: バージョン 2021.2 で追加。

既定値: `1000`

セカンダリキューに入れることができるジョブの最大数。実行中のジョブの数が設定された同時実行制限に達すると、セカンダリキューが作成されます。デフォルトの最大値は 1000 ジョブに設定されています。つまり、同時実行制限に達したときに 1000 を超えるジョブがある場合、1000 を超えるジョブはキューに入れられません。オーバーフローの最大キュー サイズを変更するには、tsm コマンドbackgrounder.vInstance_max_overflow_queue_size を使用します。

値は整数で指定する必要があります。

backup.zstd.thread_count

バージョン: バージョン 2021.1.0 で追加。このキーは動的に構成できます。詳細については、Tableau Server の動的トポロジの変更を参照してください。

既定値: 2

バックアップ作成時に使用するスレッドの数。

この数を増やすとバックアップのパフォーマンスが向上しますが、スレッド数は Tableau Server コンピューターの論理プロセッサー数を超えないようにし、多くても 4 つまでとすることをお勧めします。

basefilepath.backuprestore

既定値: `/var/opt/tableau/tableau_server/data/tabsvc/files/backups/`

`tsm maintenance backup` コマンドでバックアップを作成する場所。これは、`tsm maintenance restore` コマンドや `tsm maintenance send-logs` コマンドを使用して復

元する場合に、バックアップ ファイルが必要となる場所でもあります。これを設定した後、tsm maintenance validate-backup-basefilepath コマンド(バージョン 2022.1 以降で使用可能) を実行して、その場所の権限が正しく設定されていることを確認する必要があります。詳細については、tsm ファイル パスを参照してください。

basefilepath.log_archive

既定値: `/var/opt/tableau/tableau_server/data/tabsvc/files/log-archives/`

`tsm maintenance ziplogs` コマンドで圧縮されたアーカイブを作成する場所。詳細については、tsm ファイル パスを参照してください。

basefilepath.site_export.exports

既定値: `/var/opt/tableau/tableau_server/data/tabsvc/files/siteexports/`

`tsm sites export` コマンドでエクスポート ファイルを作成する場所。詳細については、tsm ファイル パスを参照してください。

basefilepath.site_import.exports

既定値: `/var/opt/tableau/tableau_server/data/tabsvc/files/siteimports/`

`tsm sites import` コマンドによりインポート ファイルの配置が想定される場所。詳細については、tsm ファイル パスを参照してください。

clustercontroller.log.level

バージョン: バージョン 2020.3.0 で追加。

既定値: `info`

クラスタ コントローラーのロギング レベル。このロギング レベルは動的に構成できるため、ロギング レベルだけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

clustercontroller.zk_session_timeout_ms

既定値: 300000

クラスタコントローラーがフェールオーバーの必要性を判断するまで調整サービス(ZooKeeper) を待機する時間 (ミリ秒)。

dataAlerts.checkIntervalInMinutes

既定値: 60

データアラート条件が True かどうかを判断するために Tableau Server がチェックする頻度 (分)。

(また、サーバーでは、データアラートに関連する抽出が更新される度にチェックを行います)。

dataAlerts.retryFailedAlertsAfterCheckInterval

既定値: true

失敗したデータアラートを Tableau Server が再チェックする頻度。true に設定すると、サーバーは、dataAlerts.checkIntervalInMinutes で定義した頻度で失敗したアラートを再チェックします。false に設定すると、サーバーは 5 分ごとに失敗したアラートを再チェックし、アラートの受信者に対してより頻繁にデータ条件の変更を通知しますが、これによってサーバーのパフォーマンスは低下します。

(また、サーバーでは、データアラートに関連する抽出が更新される度にチェックを行います)。

dataAlerts.SuspendFailureThreshold

既定値: 350

ある状態についてのアラートを一時停止する前に発生しなければいけない、連続するデータアラートの失敗数を決定します。既定値の 350 に設定されているとき、アラートが約 2 週間発生した後に、アラートは一時停止になります。このしきい値はサーバー全体に対するものであるため、サーバー上で定義されているすべてのデータアラートに適用されます。

databaseservice.max_database_deletes_per_run

バージョン: バージョン 2021.2 で追加。

既定値: NULL

このオプションを使用して、バックグラウンド処理のたびに削除できる埋め込み外部アセット(データベースおよびテーブル) の最大数を調整します。`features.DeleteOrphanedEmbeddedDatabaseAsset` で制御します。このオプションを空のままにすると、削除できる埋め込み外部アセットのデフォルトの最大数は 100 になります。

詳細については、features.DeleteOrphanedEmbeddedDatabaseAsset を参照してください。

dataserver.log.level

バージョン: バージョン 2020.3.0 で追加。

既定値: `info`

Data Server のロギング レベル。このロギング レベルは動的に構成できるため、ロギング レベルだけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

elasticserver.vmopts

バージョン: バージョン 2019.1 で追加。削除: 2022.1

この構成オプションは、Tableau Server バージョン 2022.1 以降では無効です。 Tableau Server バージョン 2022.1 以降の場合は、`indexandsearchserver.vmopts` 構成オプションを使用してください。

既定値: "`-Xmx<default_value> -Xms<default_value>`"

既定値は、システム メモリの量によって異なります。JVM の最大ヒープ サイズは、システム RAM 全体の 3.125% にスケールされます。

Elastic Server のヒープ サイズを制御します。既定値は自動的にスケールされます。どうしても必要な場合にのみ、このオプションを使用して既定値を上書きしてください。キロバイトを示すには 'k'、メガバイトを示すには 'm'、ギガバイトを示すには 'g' を値に追加します。一般的なルールとして、ガベージ コレクションを最小化するには、初期ヒープ サイズ (`-Xms`) を最大ヒープ サイズ (`-Xmx`) と等しくなるように設定します。

excel.shadow_copy_all_remote.enabled

バージョン: バージョン 2019.1.5、2019.2.1 で追加。

既定値: `false`

ライブ データ ソースとして使用されている共有 Excel スプレッドシート (`.xlxs` または `.xlxm`) の "シャドウ コピー" を作成するかどうかを制御します。このオプションを有効にすると、Excel で「共有違反エラー」や「ファイルが使用中です」といったメッセージが表示されなくなります。大きい Excel ファイルの場合、このオプションはパフォーマンスに影響を与える可能性があります。Excel ユーザーが共有ファイルを編集する必要がない場合、このオプションを有効にする必要はありません。

> **注:** `.xls` ファイルの場合、Tableau Server ではシャドウ コピーの作成が常に試行されます。このオプションを選択してもこの動作は変更されません。

extractservice.command.execution.timeout

バージョン: バージョン 2021.4 で追加。

既定値: `7200` 秒

VConn 抽出更新ランタイムのタイムアウト値を設定します。

例: `tsm configuration set -k extractservice.command.execution.timeout -v <timeout_in_seconds> --force-keys`

> **注:** この値を変更するには、`--force-keys` オプションを使用する必要があります。

features.ActiveMQ

バージョン: バージョン 2021.4 で追加。

既定値: `true`

Tableau Server で内部メッセージング メカニズムに Apache ActiveMQ サービス (Tableau Server Messaging Service) を使用するかどうかを制御します。

features.DeleteOrphanedEmbeddedDatabaseAsset

バージョン: バージョン 2021.2 で追加。

既定値: `true`

Tableau Catalog (または Tableau メタデータ API) のバックグラウンドプロセスを制御します。このプロセスは、ダウンストリームの Tableau コンテンツに関連付けられなくなった埋め込み外部アセット (データベースおよびテーブル) を削除します。このプロセスは毎日 22:00:00 UTC (協定世界時) に実行され、ダウンストリームの Tableau コンテンツへの接続がない外部アセットがなくなるまで、毎日最大 100 個の外部アセットを削除できます。このオプションを `false` に設定すると、このプロセスの実行を停止できます。または、databaseservice.max_database_deletes_per_run を使用して、削除できる外部埋め込みアセットの最大数を調整することもできます。

詳細については、「トラブルシューティング - コンテンツが見つからない」を参照してください。

features.DesktopReporting

既定値: `false`

サーバー上でデスクトップ ライセンス レポートを有効化するかどうかを制御します。`false` に設定すると(既定)、デスクトップ ライセンスに関連する管理ビューを使用できません。これを `true`に設定してライセンス レポートを有効にし、[サーバーのステータス] ページでライセンス使用状況と有効期限の管理ビューを表示できるようにします。**注:**Tableau Server に情報を報告するには、クライアント (Tableau Desktop) でデスクトップ ライセンス レポートを有効にする必要があります。

features.IdentityMigrationBackgroundJob

バージョン: バージョン 2022.1 で追加。

既定値: `false`バージョン 2021.4.22、2022.1.18、2022.3.10、2023.1.6、2023.3 以降、既定値は `false` に変更されました。

ID の移行を実行するプロセスを制御します。`true` に設定すると、ID の移行は、Tableau Server をバージョン 2022.1 (またはそれ以降) にアップグレードしてバージョン 2021.4 (またはそれ以前) のバックアップを復元した直後に、既存の展開で実行されます。`false` (デフォルト) に設定すると、ID の移行を無効にします。

たとえば、ID の移行を開始するには、次のコマンドを実行します。

```
tsm configuration set -k features.IdentityMigrationBackgroundJob -v
true
```

詳細については、「アイデンティティの移行について」を参照してください。

注: ID の移行が無効になっている場合、Tableau Server は ID サービスを使用してユーザーの ID 情報を保存したり管理したりすることができません。ID プールなどの特定の機能は、ID サービスを使用することが前提条件になっています。

features.IdentityPools

バージョン: バージョン 2023.1 で追加。

既定値: `false`

Tableau Server を新規にインストールする場合に有効にする必要があるアイデンティティプール機能のコンポーネント。`feature.NewIdentityMode` と `wgserver.authentication.legacy_identity_mode.enabled` が必要です。アイデンティティプールを有効にするには、`true` に設定します。アイデンティティプールを無効にするには、`false` (既定) に設定します。

たとえば、アイデンティティプールを有効するには、次のコマンドを実行します。

```
tsm configuration set -k features.IdentityPools -v true
tsm configuration set -k features.NewIdentityMode -v true
tsm configuration set -k wgserver.authentication.legacy_identity_
mode.enabled -v false
tsm pending-changes apply
```

詳細については、「ID プールのトラブルシューティング」を参照してください。

features.JWTSupportForSCIM

バージョン: バージョン 2025.3 で追加

既定値: `false`

SCIM 要求をサポートするために、外部で生成された JWT を (Tableau 接続済みアプリを通じて) 使用する機能を有効にします。外部トークンを使用すると、Tableau が生成する SCIM トークンがバイパスされます。

この機能を有効にするには、`true` に設定します。

```
tsm configuration set -k features.JWTSupportForSCIM -v true
tsm pending-changes apply
```

詳細については、「SCIM」を参照してください。

features.MessageBusEnabled

バージョン: バージョン 2019.4 で追加。

既定値: `true`

Tableau Server で新しい内部メッセージング メカニズムを使用するかどうかを制御します。

features.NewIdentityMode

バージョン: バージョン 2022.1 で追加。

既定値: `false`バージョン 2021.4.22、2022.1.18、2022.3.10、2023.1.6、2023.3 以降、既定値は `false` に変更されました。

アイデンティティプール機能の前提条件 アイデンティティプールを有効にするには、`wgserver.authentication.legacy_identity_mode.enabled` を `false` に設定する必要があります。アイデンティティプールを無効にするには、`true` に設定します。

```
tsm configuration set -k features.IdentityPools -v true
tsm configuration set -k features.NewIdentityMode -v true
tsm configuration set -k wgserver.authentication.legacy_identity_
```

```
mode.enabled -v false
tsm pending-changes apply
```

詳細については、「ID プールのトラブルシューティング」を参照してください。

features.PasswordlessBootstrapInit

既定値: `true`

Tableau Server の bootstrap ファイル内で埋め込みの認証資格情報を許可するかどうかを制御します。有効にした場合 (既定)、埋め込みの認証資格情報を含めないことを指定しない限り、この情報が bootstrap ファイルに追加されます。生成するすべての bootstrap ファイルに認証資格情報を含めない場合は、これを `false` に設定します。bootstrap ファイル生成の詳細については、tsm topology nodes get-bootstrap-file を参照してください。

このオプションは Tableau Server バージョン 2019.3 で追加されました。

features.PasswordReset

バージョン: バージョン 2024.2 で廃止。バージョン 2024.2 以降では、`vizportal.password_reset` を使用します。

既定値: `false`

ローカル認証を使用するサーバーにのみ適用されます。この値を `true` に設定し、ユーザーがサインイン ページの [パスワードを忘れましたか] オプションでパスワードをリセットできるようにします。

features.VizDataServiceClientEnable

バージョン: バージョン 2025.2 で追加。

既定値: `false`

Tableau 埋め込み API v3 ヘルプの「ユーザー属性を使用したデータアクセスのカスタマイズと制御」で説明されているユーザー属性関数の完全なサポートを有効にします。

`false` に設定すると、ユーザー属性関数は、パブリッシュされたデータソースをサポートしません。`true` に設定すると、ユーザー属性関数機能は、埋め込みワークフローのデータソースとパブリッ

シュされたデータソースをサポートされます。たとえば、次のコマンドを実行します。`tsm configuration set -k features.VizDataServiceClientEnable -v true`

features.VizqlDataServiceDeployWithTsm

バージョン: バージョン 2025.1 で追加。

既定値: `true`

Vizql データサービスが有効かどうかを制御します。有効 (既定) になっている場合、サービスは VizQL データサービス HTTP エンドポイントを公開し、ユーザーがプログラムで操作できるようにします。

プログラムによるアクセスのために VizQL データサービスを使用しない場合は、この機能を無効にすることができます。無効にすると、このサービスは TSM UI で非表示になります。そのためには、まずサービスのインスタンス数をゼロ (0) に設定する必要があります。

例:

```
tsm topology set-process --count 0 --process vizqldataservice --
node <nodeName>
tsm configuration set -k features.VizqlDataServiceDeployWithTsm -v
false
tsm pending-changes apply
```

VizQL データサービスは、API からアクセスできるように設計されています。この機能は、Tableau Server UI では利用できません。サービスの詳細については、「Tableau Server VizQL Data Service」を参照してください。

VizQL データサービス API の詳細については、「VizQL データサービス API リファレンス」を参照してください。

features.VizqlDataServicePermission

バージョン: バージョン 2025.1 で追加。

既定値: `true`

データソースのパーミッション ルール「API アクセス」の権限チェックを有効にします。API アクセスのデータソースのパーミッション ルールは、パーミッション ダイアログでグループまたはユーザーに対して設定します。パーミッション機能のフラグをオフにすると、権限チェックが無効になり、すべてのサイトロール (ライセンスなしのロールを除く) が VizQL データサービス API にアクセスできるようになります。「パーミッション」を参照してください。

VizQL データサービス応答へのフル アクセスを可能にするには、ユーザーにデータソースの表示権限と接続権限を付与して、参照元データソースにフル アクセスできるようにする必要があります。

VizQL データサービス API アクセスの推奨構成 (詳細については、パーミッションを参照してください。)

- パーミッション フラグを有効にしたまま、Tableau Server のパーミッション ダイアログで、ユーザーではなくグループに対して API アクセス機能のパーミッション ルールを設定します。
- 個々のデータソース コンテンツにパーミッションを設定するのではなく、プロジェクト レベルで API アクセス機能のパーミッションを管理します。
- すべてのユーザー グループの API アクセス機能パーミッションが、**[なし]** に設定されていることを確認します。

VizQL データサービスは、API からアクセスできるように設計されています。この機能は、Tableau Server UI では利用できません。サービスの詳細については、「Tableau Server VizQL Data Service」を参照してください。

VizQL データサービス API の詳細については、「VizQL データサービス API リファレンス」を参照してください。

filestore.empty_folders_reaper.enabled

バージョン: 2020.x (2020.1.14、2020.2.11、2020.3.6、2020.4.2) と2021.1.x で追加。デフォルト値は、2021.2 で `true` に変更されました。

既定値: `true`

空の Filestore フォルダーを“刈り取る” (削除する) ジョブを有効にします。

filestore_empty_folders_reap.frequency_s

バージョン: 2020.x (2020.1.14、2020.2.11、2020.3.6、2020.4.2) で追加。

既定値: `86400` (24 時間)

空の Filestore フォルダーを削除するジョブの実行頻度を分単位で指定します。

features.Hyper_DisallowTDEPublishing

バージョン: バージョン 2023.1.0 から `true` が既定になりました。2024.2 で非推奨。2025.1 で廃止。

既定値: `true`

ユーザーが .tde 形式のファイルをアップロードできるかどうかを指定します。この形式は、Tableau Server のバージョン 10.5 から .hyper 形式に置き換えられましたが、アップロードはブロックされていません。バージョン 2024.3 以降、.tde 形式のファイルは使用できなくなりました。いくつかあるアクションのいずれかが実行されると、ファイルは自動的に .hyper 形式に変換されました。詳細については、「.hyper 形式への抽出のアップグレード」を参照してください。

filestore.log.level

バージョン: バージョン 2020.3.0 で追加。

既定値: `info`

ファイル ストアのロギング レベル。このロギング レベルは動的に構成できるため、ロギング レベルだけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

filestore.reapemptyfoldersholdoffms

バージョン: 2020.x (2020.1.14、2020.2.11、2020.3.6、2020.4.2) で追加。2021.1 ではまだ利用できません。

既定値: `300000` (5 分)

空の Filestore フォルダーを削除するまでに待機する時間をミリ秒単位で指定します。

floweditor.max_datafile_upload_size_in_kb

バージョン: バージョン 2020.4 で追加。

既定値: `1048576`

Tableau Prep フローの Web オーサリングにおいて、Tableau Server にアップロードできる文字区切りテキストファイル (CSV や TXT など) の最大サイズ。

gateway.external_url

バージョン: バージョン 2023.1 で追加されました

既定値: Null

Tableau Server をセットアップするとき、または ID プールを使用するとき、OpenID Connect (OIDC) 認証を TSM で設定するために必要です。ID プロバイダー (IdP) が認証するユーザーを Tableau にリダイレクトするために使用する Tableau Server URL を指定します。ゲートウェイの外部 URL は、IdP でリダイレクト URL として指定したものと同じ URL です。この URL は、照合目的で使用されます。

たとえば、OIDC 認証設定で関連付けた IdP を Tableau Server (http://myco) にリダイレクトするには、次のコマンドを実行します。

```
tsm configuration set -k gateway.external_url -v http://myco
```

注: バージョン 2025.3 以降、指定された URL (有効な場合) が、シングル ログアウト (SLO) に使用されるエンドポイントになります。

gateway.http.cachecontrol.updated

既定値: `false`

Cache-Control HTTP ヘッダーは、クライアントブラウザーが Tableau Server から送信されたコンテンツをキャッシュする必要があるかどうかを指定します。クライアントでの Tableau Server データのキャッシュを無効にするには、このオプションを `true` に設定します。

gateway.http.hsts

既定値: `false`

HTTP Strict Transport Security (HSTS) ヘッダーによって、有効になっているドメインの HTTPS がブラウザーで強制的に使用されます。

gateway.http.hsts_options

既定値: `"max-age=31536000"`

既定では、HSTS ポリシーが 1 年に設定されています (31536000 秒)。ブラウザーが HTTPS 経由でサーバーにアクセスする期間を指定します。

gateway.httpd.loglevel

バージョン: 2021.3.0 で追加。

既定値: `notice`

ゲートウェイ (Apache HTTPD サーバー) のロギング レベルを指定します。既定では `notice` に設定されています。その他の選択肢には、`debug`、`info`、`warning`、および `error` があります。ロギング レベルを変更する場合は、ディスク容量使用率とパフォーマンスに影響が及ぶ可能性があることに注意してください。ベストプラクティスとして、必要な情報を収集した後、ロギング レベルを既定に戻します。Apache ロギングの詳細については、Apache HTTP に関するドキュメントを参照してください。

gateway.httpd.shmcb.size

バージョン: 2021.4 で追加

既定値: `2048000`

`shmcb` ストレージ タイプを使用する場合の循環バッファーのメモリ量をバイト単位で指定します。この構成キーは、`dbm` ストレージ タイプを使用する場合は適用されません。

gateway.httpd.socache

バージョン: 2021.4 で追加

既定値: `shmcb`

グローバルまたはプロセス間の SSL セッション キャッシュのストレージ タイプを指定します。デフォルトでは、`shmcb` に設定されています。構成可能なもう一つのオプションは `dbm` です。`shmcb` と`dbm` のストレージ タイプの詳細については、Apache Web サイトの「SSLSessionCache Directive」を参照してください。

gateway.http.request_size_limit

既定値: `16380`

HTTP 要求上で Apache ゲートウェイを通じて渡すことが許可されているヘッダー コンテンツの最大サイズ (バイト)。ヘッダーがこのオプションで設定された値を超えると、ブラウザーに HTTP Error 413 (要求するエンティティが長すぎます) や認証エラーなどのエラーが表示されます。

`gateway.http.request_size_limit` の値が小さいと、認証エラーが発生する可能性があります。Active Directory (SAML および Kerberos) と統合するシングル サインオン ソリューションでは、多くの場合、HTTP ヘッダーに大きな認証トークンが必要です。本番環境に展開する前に、必ず HTTP 認証シナリオをテストしてください。

`tomcat.http.maxrequestsize` オプションを、このオプションに設定した値と同じ値に設定することをお勧めします。

gateway.http.x_content_type_nosniff

既定値: `true`

X-Content-Type-Options の応答 HTTP ヘッダーによって、Content-Type ヘッダーの MIME タイプがブラウザーで変更されないように指定されます。MIME タイプが指定されていない場合、ブラウザーではペイロードの特性を評価して MIME タイプを判断しようとすることがあります。次に、ブラウザーにコンテンツが表示されます。このプロセスは「スニッフィング」と呼ばれます。MIME タイプを誤って解釈すると、セキュリティの脆弱性を招く恐れがあります。X-Content-Type-Options HTTP ヘッダーは、既定でこのオプションを使用して 'nosniff' に設定されます。

gateway.http.x_xss_protection

既定値: `true`

HTTP X-XSS-Protection 応答ヘッダーがブラウザーに送信され、クロスサイト スクリプティング (XSS) 保護が有効になります。X-XSS-Protection 応答ヘッダーは、ユーザーがブラウザーで XSS 保護を無効にした場合に構成を上書きします。既定では、X-XSS-Protection 応答ヘッダーがこのオプションを使用して有効になります。

gateway.log.level

バージョン: バージョン 2020.3.0 で追加。

既定値: `info`

ゲートウェイのロギング レベル。このロギング レベルは動的に構成できるため、ロギング レベルだけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

gateway.public.host

既定値: <hostname>

Tableau Server への外部からのアクセスで使用されるサーバーの名前 (URL)。Tableau Server がプロキシ サーバーまたは外部ロード バランサーを使用するように構成されている場合は、Tableau Server にアクセスするためにブラウザーのアドレス バーに入力した名前です。たとえば、`tableau.example.com` と入力して Tableau Server にアクセスする場合、gateway.public.host の名前は`tableau.example.com` です。

gateway.public.port

既定値: `80` (SSL の場合は `443`)

プロキシ サーバー環境のみに適用。プロキシ サーバーがリッスンする外部ポート。

gateway.slow_post_protection.enabled

既定値: `true`

有効にすると、データを非常に遅いレートで転送する POST 要求をタイムアウトすることで、スロー POST (Denial-of-Service) 攻撃から保護することにある程度役立ちます。

> **注:** これを有効にしても、これらの攻撃の脅威が排除されるわけではありません。また、接続速度の低下を終了させることによる影響が意図せずに出る場合があります。

gateway.slow_post_protection.request_read_timeout

既定値: `header=10-30,MinRate=500 body=30,MinRate=500`

前のオプション `gateway.slow_post_protection.enabled` を有効にした場合、このオプションで Apache httpd ReadRequestTimeout を設定します。この httpd ディレクティブについては、「Apache Module mod_reqtimeout」を参照してください。このオプションの主な用途は、Slowloris 攻撃からの防御です。Wikipedia のエントリ「Slowloris (computer security)」を参照してください。

注: 以前のバージョンで使用していたデフォルト値: `header=15-20,MinRate=500 body=10,MinRate=500`

gateway.timeout

既定値: `7200`

ゲートウェイが特定のイベントを待機する最大の秒数であり、この時間を過ぎると要求に失敗します (7200 秒 = 2 時間)。

gateway.trusted

既定値: プロキシ サーバー マシンの IP アドレス

プロキシ サーバー環境のみに適用。プロキシ サーバーの IP アドレスまたはホスト名。

gateway.trusted_hosts

既定値: プロキシ サーバーの代替名

プロキシ サーバー環境のみに適用。プロキシ サーバーの任意の代替ホスト名。

hyper.file_partition_size_limit

既定値: `0`

0 に設定するとサイズが無制限となり、使用可能なディスク空き容量がすべて使用されます。

このオプションを使用して、ディスクをスプールするクエリのディスク容量制限を設定します。spool.<id>.tmp ファイルによるディスク容量使用率がお使いの環境で必要な容量より多い場合、クエリがスプールされており、ディスク容量を消費していることを意味します。このオプションを使用して、クエリで使用できるディスク容量を制限します。spool.<id>.tmp ファイルは、Tableau Server を実行しているユーザー アカウントの一時フォルダー内にあります。この値は K(KB)、M(MB)、G(GB)、T(TB) 単位で指定できます。たとえば、ディスク容量使用率を 100 GB に制限する場合は、サイズ制限を 100G と指定できます。

スプーリングの詳細については、Tableau Server データ エンジンの「メモリと CPU 使用率」セクションを参照してください。

hyper.global_file_partition_size_limit

既定値: 0

0 に設定するとサイズが無制限となり、使用可能なディスク空き容量がすべて使用されます。

このオプションを使用して、ディスクをスプールするすべてのクエリのディスク容量制限を設定します。spool.<id>.tmp ファイルによるディスク容量使用率がお使いの環境で必要な容量より多い場合、クエリがスプールされており、ディスク容量を消費していることを意味します。spool.<id>.tmp ファイルは、Tableau Server を実行しているユーザー アカウントの一時フォルダー内にあります。このオプションを使用して、ディスクをスプーリングするときにすべてのクエリで使用するディスク容量の合計を制限します。この値は K(KB)、M(MB)、G(GB)、T(TB) 単位で指定できます。たとえば、ディスク容量使用率を 100 GB に制限する場合は、サイズ制限を 100G と指定できます。Tableau では、スプーリングの制限を微調整する際にこの構成を使用して開始することをお勧めします。

スプーリングの詳細については、Tableau Server データ エンジンの「メモリと CPU 使用率」セクションを参照してください。

hyper.enable_accesspaths_symbolic_canonicalization

既定値: false

Windows OS システムでは、シンボリックリンクを解決するために、Hyper は、抽出が保存されているディレクトリとそのすべての親ディレクトリにアクセスできる必要があります。そうでない場合は、Hyper ログに次のエラー メッセージが表示されることがあります: **次の正規パスを取得できません** `//dirA/subdir/myextract.hyper ... Access is denied`。

このような場合、これを `true` に設定することで、データエンジン (Hyper) が正規パスを使用するときにシンボリックリンクを解決しようとしないようにすることができます。

> **注:** 値を `true` に設定すると、Hyper は、潜在的な攻撃者がシンボリックリンクを配置して Hyper がアクセスできる許可されたディレクトリセットからエスケープしようとする場合に、保護を保証することができなくなります。

hyper.log_queries

既定値: `true`

true に設定すると、クエリ情報がログに記録されます。

既定では、クエリ情報がログに記録されます。ただし、ログ ファイルが使用可能なディスク容量に対して大きすぎると判明した場合は、これを `false` に設定してクエリ情報のログへの記録を無効にできます。Tableau では、この設定を `true` のままにしておくことをお勧めします。

hyper.log_query_cpu

既定値: `false`

この設定を使用して、クエリに要する時間と CPU 使用率をログに記録します。

hyper.log_timing

既定値: `false`

この設定は、編集時間や解析時間など、クエリに関する詳細情報を確認する際に便利です。既定では、この設定は無効化されています。この値を `true` に変更して、クエリに関する詳細をさらに収集することができます。ただし、これによりデータエンジンのログ ファイル (\logs\hyper) のサイズが

大きくなりますのでご注意ください。

hyper.log_troublesome_query_plans

既定値: `true`

`true` に設定すると、問題があると識別されたクエリのクエリプランがログに記録されます。キャンセルされたクエリや 10 秒以上遅れて実行しているクエリ、またはディスクにスプーリングしているクエリは、このカテゴリに分類されます。ログの情報は、問題のあるクエリをトラブルシューティングする際に便利です。ログのサイズが心配なときは、この設定を `false` に変更できます。

hyper.memory_limit

既定値: `80%`

Hyper が使用するメモリの最大量を制御します。バイト数を指定します。キロバイトを示すには 'k'、メガバイトを示すには 'm'、ギガバイトを示すには 'g'、テラバイトを示すには 't' を値に追加します。たとえば、`hyper.memory_limit="7g"` です。または、使用可能なシステム メモリ全体に対するパーセンテージとしてメモリ制限を指定します。たとえば、`hyper.memory_limit="90%"` です。

hyper.memtracker_hard_reclaim_threshold

既定値: `80%`

この設定は、Windows にのみ適用されます。Hyper では、抽出のうち解凍されて復号化された部分がメモリに保持されて、後続のアクセスが高速化されます。この設定は、メモリの負荷を軽減するためにワーカー スレッドがこのデータをディスク キャッシュに書き込み始めるタイミングを制御します。パーセンテージで指定した場合、値は全体的な設定 `hyper.memory_limit` に対するパーセンテージとして解釈されます。たとえば、`hyper.memtracker_hard_reclaim_threshold="60%"` です。絶対値は、'k' (キロバイト)、'm' (メガバイト)、'g' (ギガバイト)、't' (テラバイト) として指定できます。たとえば、`hyper.memtracker_hard_reclaim_threshold="10g"` です。値はしきい値 `hyper.memtracker_soft_reclaim` より大きくする必要があります。

hyper.memtracker_soft_reclaim_threshold

既定値: 50%

この設定は、Windows にのみ適用されます。Hyper ファイルを操作する際、Hyper はキャッシュまたは永続化するために一部のデータを書き出します。Windows には、新しく書き込まれたデータをメモリにロックするという特別な動作があります。スワップを避けるため、Hyper は再利用しきい値用に構成された制限に達したときにデータを強制的に書き出します。このソフト再利用しきい値に達すると、Hyper はバックグラウンドでキャッシュされたデータを再利用して、再利用しきい値より下の状態を維持しようとします。スワップが発生する可能性のある状況では、Hyper で再利用をトリガーすると、より良い結果が得られます。したがって、Tableau Server のインストールで大量のスワップが発生する場合は、この設定を使用してメモリの負荷を軽減できます。

バイト数を指定します。キロバイトを示すには 'k'、メガバイトを示すには 'm'、ギガバイトを示すには 'g'、テラバイトを示すには 't' を値に追加します。または、Hyper 用に構成されたメモリ全体に対するパーセンテージで値を指定します。たとえば、`hyper.memtracker_soft_reclaim_threshold="20%"` です。

hyper.network_threads

既定値: 150%

Hyper が使用するネットワークスレッドの数を制御します。ネットワークスレッド数 (`hyper.network_threads=4` など) を指定するか、論理コア数に対するスレッド数の割合 (`hyper.network_threads="300%"` など) を指定します。

ネットワークスレッドは、新しい接続を受け入れたりデータやクエリを送受信したりするために使用されます。Hyper では非同期ネットワークが使用されるため、1 つのスレッドで多くの接続を処理できます。通常、ネットワークスレッドで行われる作業の量は非常に少なくなっています。1 つの例外は、低速のファイルシステムでデータベースを開く場合です。これには長い時間がかかるため、ネットワークスレッドがブロックされる可能性があります。抽出を使用するダッシュボードをしばらく使用しなかった後で表示または編集する際の接続時間が遅く、Hyper ログに "asio-continuation-slow" メッセージが頻繁に表示され、Tableau ログで Hyper に対する "construct-protocol" の時間が長くなっている場合は、この値を増やします。

hyper.objectstore_validate_checksums

既定値: `false`

Hyper でのファイル整合性チェックを制御するブール値。`true` に設定すると、Hyper は抽出ファイルに初めてアクセスしたときにその中のデータをチェックします。これにより、Hyper のクラッシュを引き起こす無兆候の破損を検出できます。一般に、非常に低速なディスクを使用するインストール (パフォーマンスの低下が生じる可能性がある) を除いて、この設定をオンにすることをお勧めします。

hyper.query_total_time_limit

既定値: `0` (無制限 (unlimited) を意味します)

Hyper の個々のクエリで使用できる合計スレッド時間の上限を設定します。秒を示すには 's'、分を示すには 'min'、時間を示すには 'h' を値に追加します。

たとえば、すべてのクエリで使用可能な合計スレッド時間を 1500 秒に制限するには、次のコマンドを実行します。

```
tsm configuration set -k hyper.query_total_time_limit -v 1500s
```

クエリが指定された制限よりも長く実行されると、クエリは失敗し、エラーが返されます。この設定を使用すると、多数のリソースを使用する可能性のあるランナウェイ クエリを自動的に制御できます。

Hyper はクエリを並列に実行します。たとえば、100 秒間実行されるクエリが 30 個のスレッドで実行されると、合計スレッド時間は 3000 秒になります。各クエリのスレッド時間は、Hyper ログの "total-time" フィールドの "query-end" ログ エントリで報告されます。

hyper.session_memory_limit

既定値: `0` (無制限 (unlimited) を意味します)

個々のクエリで使用できる最大メモリ消費量を制御します。バイト数を指定します。キロバイトを示すには 'k'、メガバイトを示すには 'm'、ギガバイトを示すには 'g'、テラバイトを示すには 't' を値に追加します。

たとえば、メモリの制限値を900 MB に設定するには、次のコマンドを実行します。

```
tsm configuration set -k hyper.session_memory_limit -v 900m
```

または、セッション メモリの制限値を使用可能なシステム メモリ全体に対するパーセンテージで指定するには、次のコマンドを実行します。

```
tsm configuration set -k hyper.session_memory_limit -v 90%
```

あるクエリが過度の量のメモリを使用しているために他のクエリが長期間にわたって失敗する場合は、この値を減らすと役立ちます。この制限値を減らすと、単一の大きなクエリは失敗する(スプーリングがオフになっていない場合はスプーリングに移行する) ため、他のクエリに悪影響を及ぼすことがなくなります。

hyper.srm_cpu_limit_percentage

既定値 (パーセント): 75

Hyper が許可する1 時間あたりの平均 CPU 使用率の最大値を指定します。超過した場合、データエンジンは、コンピューター上の他のプロセスへの影響を最小限に抑えるために、自身を再起動します。

既定では、データエンジンは、1 時間の平均 CPU 使用率が75% を超えると、自動的に再起動します。この値は、Tableau サポートで作業している場合、または専用サーバー ノードでデータエンジンを実行している場合を除き、変更しないでください。データエンジンが専用 ノードで実行されている場合は、この値を安全に95% に増やして、利用可能なコンピューター ハードウェアを最大限に活用できます。専用 ノードでデータエンジンを実行する方法の詳細については、「抽出 クエリの負荷が大きい環境用に最適化」を参照してください。

これを95% に増やすには:

```
tsm configuration set -k hyper.srm_cpu_limit_percentage -v 95 --
force-keys

tsm pending-changes apply
```

これをデフォルトの75% にリセットするには:

```
tsm configuration set -k hyper.srm_cpu_limit_percentage -v 75 --
force-keys

tsm pending-changes apply
```

hyper_standalone.consistent_hashing.enabled

既定値: `true`

クエリの抽出が既にキャッシュされている可能性が高くなります。抽出がキャッシュされたノードで追加の読み込みに対応できない場合は、新しいノードにルーティングすると、抽出が新しいノードのキャッシュに読み込まれます。これにより、必要性が正当化された負荷がある場合にのみ抽出がメモリに読み込まれるため、システム使用率が向上します。

hyper_standalone.health.enabled

既定値: `true`

現在の Hyper アクティビティとシステム リソースの使用状況を組み合わせて構成された正常性スコアに基づいて、ロード バランシング メトリクスをランダムな選択からデータエンジン (Hyper) ノードの選択に切り替えます。これらの値に基づいて、ロード バランサーは抽出クエリの処理能力が最も高いノードを選択します。

hyper.temp_disk_space_limit

既定値: `100%`

Hyper が一時ファイル用の領域の割り当てを停止するディスク空き容量の上限を設定します。この設定は、ハードディスクが Hyper からの一時ファイルでいっぱいになってディスク空き容量が不足するのを防ぐのに役立ちます。ディスク空き容量がこのしきい値に達すると、管理者の介入なしに自動的に領域の解放が試行されます。

使用可能なディスク空き容量全体に対するパーセンテージで指定します。たとえば、`hyper.temp_disk_space_limit="96%"` のように使用します。100% に設定すると、使用可能なすべてのディスク空き容量を使用できます。

データエンジンを起動するには、構成されたディスク領域が使用可能でなければなりません。十分なディスク領域がない場合は、データエンジン ログ エントリが表示され、「一時 ファイルのディスク制限に達しました。デバイスのディスク領域を増やしてください。詳細については、Hyper ログを参照してください: デバイスに空き領域がありません」と表示されます。

hyper.hard_concurrent_query_thread_limit

既定値: 150%

このオプションは、Hyper がクエリの実行に使用する最大スレッド数を設定するのに使用します。CPU 使用率にハード制限を設定する場合はこれを使用してください。スレッドの数を指定するか、論理コア数に対するスレッドのパーセンテージを指定します。Hyper で使用されるリソースがこの設定の構成値を超える可能性はほとんどありませんが、Hyper のバックグラウンド スレッドやネットワーク スレッドはこの設定の影響を受けません (ただし、これらのスレッドは CPU を集中的に使用しない傾向があります)。

これを設定すると、同時に実行可能なクエリ数が制御されることを考慮に入れることが重要です。そのため、この設定を小さくすると、新しいクエリが現在実行中のクエリの完了を待機しなければならない可能性が高くなるため、ワークブックの読み込み時間に影響が出る可能性があります。

hyper.soft_concurrent_query_thread_limit

既定値: 100%

このオプションを使用すると、`hard_concurrent_query_thread_limit` の設定によって十分な数のスレッドが使用可能な場合に 1 つのクエリを並列で実行できるスレッドの数を指定できます。スレッドの数を指定するか、論理コア数に対するスレッドのパーセンテージを指定します。

これを説明するため、ここで簡単な例を示します。

この値を 10 スレッドに設定するとします。つまり、クエリは最大 10 スレッドまで並列で実行できます。2 つのクエリだけを実行している場合、残りの 8 スレッドが 2 つのクエリを並列で実行するのに使用されます。

hyper. hard_concurrent_query_thread_limit オプションと*hyper.soft_concurrent_query_thread_limit* オプションは連係し、クエリを迅速に完了するために利用可能な CPU リソースを最大化しながらCPU 使用率を管理するいくつかのオプションが得られます。データエンジンがマシン上で利用可能なすべての CPU を使用しないようにするには、100% 未満に変更し、お使いの環境に最適なパーセンテージにしてください。ソフト制限は CPU 使用率を制限する方法ですが、必要に応じ、ソフト制限を超過し、最大でハード制限までの使用を許可することができます。

注: Tableau Server バージョン 2018.3 以前で利用できたhyper.num_job_worker_threads オプションとhyper.num_task_worker_threads オプションは廃止され現在は利用できず、*hyper.hard_concurrent_query_thread_limit* オプションと*hyper.soft_concurrent_query_thread_limit* オプションに置き換えられました。hyper.num_job_worker_threads と hyper.num_task_worker_threads については、「tsm configuration set のオプション」を参照してください。

hyper.use_spooling_fallback

既定値: `true`

`true` に設定すると、抽出のクエリ作成時に使用率が設定されたRAM の使用率を超過した場合 (インストールされたRAM の 80%) にディスクへスプーリングできます。つまり、Hyper で RAM の使用率が設定値を超えた場合に、ディスクを使用してクエリを実行できます。

既定の設定を使用することをお勧めします。ディスク使用率に不安がある場合は、この値を `false` に変更できます。この設定をオフにすると、インストールされたRAM の 80% を超過したクエリはキャンセルされます。通常、クエリのスプーリングは完了するまでにかなり時間がかかります。

スプーリングの詳細については、Tableau Server データエンジンの「メモリとCPU 使用率」セクションを参照してください。

indexandsearchserver.vmopts

バージョン: バージョン 2022.1 で追加。

既定値: "-Xmx<default_value> -Xms<default_value>"

既定値はシステム メモリの量に基づいており、システム RAM 全体の 3.125% です。

Index and Search Server のヒープ サイズを制御します。既定値は自動的にスケールされます。どうしても必要な場合にのみ、このオプションを使用して既定値を上書きしてください。キロバイトを示すには 'k'、メガバイトを示すには 'm'、ギガバイトを示すには 'g' を値に追加します。一般的なルールとして、ガベージ コレクションを最小化するには、初期ヒープ サイズ (-Xms) を最大ヒープ サイズ (-Xmx) と等しくなるように設定します。

jmx.security.enabled

バージョン: バージョン 2022.1 で追加。

既定値: false

JMX はデフォルトで無効になっているため、セキュア JMX も無効になっています。JMX を有効にする場合は、セキュア JMX を有効にすることを強くお勧めします。

tsm maintenance jmx enable コマンドを実行し、JMX のセキュリティ機能を有効にするように求められたときに y と回答すると、この構成は true に設定され、SSL と基本的なユーザー名とパスワードの認証、読み取り専用アクセスでセキュア JMX が有効になります。

```
tsm maintenance jmx enable
We do not recommend you enable JMX unsecured on a production
environment. Would you like to enable security features for JMX?
(y/n): y
```

jmx.ssl.enabled

バージョン: バージョン 2022.1 で追加。

既定値: true

JMX に SSL を適用します。このオプションは、既定で true に設定されていますが、jmx.security.enabled も true に設定されていない限り、効果はありません。JMX セキュリティを有効にするには、tsm maintenance jmx enable コマンドを実行します。プロンプトが表示されたら、「y」と入力して SSL を有効のままにするか、「n」と入力して SSL を無効にします。

```
tsm maintenance jmx enable
...
Would you like to enable SSL?
(y/n): n
```

jmx.ssl.require_client_auth

バージョン: バージョン 2022.1 で追加。

既定値: false

tsm maintenance jmx enable コマンドを実行し、クライアント認証 (mTLS) を有効にするように求められたときに y と回答すると、この構成は true に設定されます。

```
tsm maintenance jmx enable
...
Would you like to require client authentication (mTLS)?
(y/n): y
```

構成を完了するには、クライアント証明書が必要です。クライアント証明書はクライアントコンピューターの適切な場所に配置する必要があります。

jmx.ssl.user.name

バージョン: バージョン 2022.1 で追加。

既定値: tsmjmxuser

Tableau Server をインストールまたはアップグレードするときに設定されます。

jmx.ssl.user.password

バージョン: バージョン 2022.1 で追加。

既定値: `<generated>`

Tableau Server をインストールまたはアップグレードするときに設定されます。

jmx.user.access

バージョン: バージョン 2022.1 で追加。

既定値: `readonly`

`tsm maintenance jmx enable` コマンドを実行し、`readwrite` アクセスを追加するように求められたときに `y` と回答すると、この構成は `readwrite` に変更できます。

```
tsm maintenance jmx enable
...
JMX access is readonly by default. Would you like to add readwrite
access?
(y/n): y
```

licensing.login_based_license_management.default_requested_duration_seconds

既定値: `0`

ユーザーのログインベースのライセンスが Tableau Server に接続せずにオフラインになり、ライセンス認証を再度行うように求めるメッセージが表示されるまでの時間 (秒単位) を設定します。この時間は、Tableau Desktop を使用していて Tableau Server に接続できるときに常に更新されます。

licensing.login_based_license_management.enabled

既定値: `true`

ログインベースのライセンス管理 を有効にするには、true に設定します。ログインベースのライセンス管理 を無効にするには、false に設定します。

> **注:** ログインベースのライセンス管理を使用するには、ログインベースのライセンス管理 に対して有効になっているプロダクトキーをライセンス認証する必要があります。`tsm licenses`

`list` を使用して、どのプロダクトキーが ログインベースのライセンス管理 で有効になっているかを確認できます。

licensing.login_based_license_management.max_requested_duration_seconds

既定値: `15552000`

ユーザーのログインベースのライセンスが Tableau Server に接続せずにオフラインになり、ライセンス認証を再度行うように求めるメッセージが表示されるまでの時間 (秒単位) の最大値を設定します。最大値は 15552000 秒 (180 日) です。この時間は、Tableau Desktop を使用していて Tableau Server に接続できるときに常に更新されます。

maestro.app_settings.sampling_max_row_limit

既定値: `1000000`

Web 上の Tableau Prep を使用して、大規模なデータセットからデータをサンプリングするための最大行数を設定します。

maestro.input.allowed_paths

既定値: ""

既定では、いずれのディレクトリへのアクセスも拒否され、tflx ファイルに含まれているコンテンツの Tableau Server へのパブリッシュのみが許可されます。

フロー入力接続で許可されるネットワークディレクトリの一覧。Tableau Server 上で Tableau Prep Conductor がフローのスケジュールを設定できるように有効化する必要があります。詳細については、Tableau Prep Conductorを参照してください。

この設定を構成するときは、次のルールを適用し、考慮に入れる必要があります。

- パスに Tableau Server からアクセスできる必要があります。これらのパスは、サーバー起動時およびフロー実行時に検証されます。

- ネットワークディレクトリパスは絶対パスである必要があり、ワイルドカードやパスを横断する他の記号を含めることはできません。たとえば、`\\myhost\myShare\*` または `\\myhost\myShare*` は無効なパスであるため、すべてのパスが許可されません。*myShare* の下にあるすべてのフォルダーを許可リストに登録する正しい方法は、`\\myhost\myShare or \\myhost\\myShare\` です。

  注: `\\myhost\myShare` 構成では `\\myhost\myShare1` が許可されません。これらのフォルダーを両方とも許可リストに登録するには、これらを \\myhost\myShare; \\myhost\myShare1 として許可リストに登録します。

- 値は、ローカル ("native_api.internal_disallowed_paths" を使用して構成された一部のシステム パスを除く) を含む任意のパスを意味する * にするか、パスのリストを ";" で区切ったものとします。

  注: パスが許可されたフローのリストとinternal_disasslowed リストの両方にある場合、internal_disallowed が優先されます。

重要:
このコマンドにより既存の情報が上書きされ、提供した新しい情報に取って代わります。既存のリストに新しい場所を追加する場合は、すべての場所、既存の場所、追加する新しい場所のリストを提供する必要があります。入力場所および出力場所の現在のリストを表示するには、次のコマンドを使用します。

```
tsm configuration get -k maestro.input.allowed_paths
tsm configuration get -k maestro.output.allowed_paths
```

フロー入力接続およびフロー出力接続で許可されるディレクトリの構成の詳細については、「手順 4 - 入力場所と出力場所を許可リストに登録する」を参照してください。

maestro.output.allowed_paths

既定値: ""

既定では、ディレクトリへのアクセスは拒否されます。

フロー出力接続で許可されるネットワークディレクトリの一覧。Tableau Server 上で Tableau Prep Conductor がフローのスケジュールを設定できるように有効化する必要があります。詳細については、Tableau Prep Conductorを参照してください。

この設定を構成するときは、次のルールを適用し、考慮に入れる必要があります。

- パスに Tableau Server からアクセスできる必要があります。これらのパスは、サーバー起動時およびフロー実行時に検証されます。
- ネットワークディレクトリパスは絶対パスである必要があり、ワイルドカードやパスを横断する他の記号を含めることはできません。たとえば、`\\myhost\myShare\*` または `\\myhost\myShare*` は無効なパスであるため、すべてのパスが許可されません。*myShare* の下にあるすべてのフォルダーを許可リストに登録する正しい方法は、`\\myhost\myShare or \\myhost\\myShare\` です。

  > **注:** `\\myhost\myShare` 構成では `\\myhost\myShare1` が許可されません。これらのフォルダーを両方とも許可リストに登録するには、これらを \\myhost\myShare; \\myhost\myShare1 として許可リストに登録します。

- 値は、ローカル ("native_api.internal_disallowed_paths" を使用して構成された一部のシステム パスを除く) を含む任意のパスを意味する * にするか、パスのリストを ";" で区切ったものとします。

  > **注:** パスが許可されたフローのリストとinternal_disasslowed リストの両方にある場合、internal_disallowed が優先されます。

フロー入力接続およびフロー出力接続で許可されるディレクトリの構成の詳細については、「手順 4 - 入力場所と出力場所を許可リストに登録する」を参照してください。

maestro.output.write_to_mssql_using_runas

バージョン: バージョン 2022.3.1 で追加。

既定値: `false`

有効にすると、Tableau Server にパブリッシュされたフロー出力は、実行認証資格情報を使用して、Microsoft SQL Server データベースに書き込みアクセスできるようになります。実行サービス アカウントで使用される認証資格情報には、データベースへの書き込みパーミッションが必要です。`maestro.output.write_to_mssql_using_runas` 設定を有効にする前に、セキュリティと展開の要件を評価してください。詳細については、「実行サービス アカウント」を参照してください。

> **注**: このコマンドには `--force-keys` オプションが必要です。たとえば、 `tsm configuration set -k maestro.output.write_to_mssql_using_runas -v true --force-keys`。

maestro.sessionmanagement.maxConcurrentSessionPerUser

既定値: `4`

ユーザーが一度に開くことができるフロー Web 編集セッションの最大数を設定します。

metadata.ingestor.blocklist

既定値: NULL

構成されると、Tableau Catalog で指定されたコンテンツの取り込みがブロックされます。ブロックするコンテンツを指定するには、拒否リストの値を指定する必要があります。これは、"インタラクティブでない" サーバー ログ ファイルからサイト ID、コンテンツ タイプおよびブロックするコンテンツのコンテンツ ID を組み合わせたものです。拒否リストの値は、コンマで区切る必要があります。

重要: このオプションは、Tableau サポートから指示された場合にのみ使用してください。

たとえば、`tsm configuration set --force-keys -k metadata.ingestor.blocklist` を使用して、データ ソース、ワークブック、およびフローを組み合わせた取り込みをブロックするには、次のコマンドを使用します。

```
tsm configuration set --force-keys -k metadata.ingestor.blocklist -v "sites/1/datasources/289, sites/2/datasources/111, sites/1/workbooks/32, sites/3/workbooks/15, sites/1/flows/13, sites/1/flows/18”
```

ブロックされたコンテンツを検証するには、"インタラクティブでない" サーバー ログ ファイルで次のイベントを確認します。

- `Skipping ingestion for`
- `Successfully updated blocklist to`

例:

```
Skipping ingestion for contentType [Workbook], contentId [sites/1/datasources/289], siteDisabled [false], swallowEvent [false], contentBlocked [true]
```

```
Skipping ingestion for contentType [Workbook], contentId [sites/3/workbooks/15], siteDisabled [false], swallowEvent [false], contentBlocked [true]
```

および

```
Successfully updated blocklist to: [sites/1/datasources/289, sites/1/workbooks/32, sites/2/datasources/111]
```

metadata.ingestor.pipeline.throttleEventsEnable

既定値: `false`

新しいコンテンツと更新されたコンテンツのインデックス作成 (Eventing とも呼ばれる) がサーバー上のすべてのサイトで制御されるかどうかを制御します。デフォルトでは、イベント スロットルはオフに

なっています。イベントスロットルをオンにするには、次のコマンドを使用して、この設定をtrue に変更します。

```
tsm configuration set -k
metadata.ingestor.pipeline.throttleEventsEnable -v true --force-keys
```

イベントスロットルの詳細については、Tableau Catalog の有効化を参照してください。

metadata.ingestor.pipeline.throttleLimit

既定値: 20

イベントスロットルが有効な場合、指定した期間中にインデックスを作成できる、新しいコンテンツアイテムと更新されたコンテンツアイテムの最大数です。指定したアイテムが制限に達すると、のインデックス作成は延期されます。

既定では、制限は 20 に設定され、2 より低い値に設定することはできません。次のコマンドを使用して、制限を変更できます。

```
tsm configuration set -k metadata.ingestor.pipeline.throttleLimit -v
25 --force-keys
```

スロットル済みのイベントは、サーバーの "noninteractive" ログファイルで ingestor event flagged for removal by throttle filter として識別できます。

metadata.ingestor.pipeline.throttlePeriodLength

既定値: 20

分単位の時間です。この間、イベントスロットルが有効な場合、指定された最大数の新しいコンテンツアイテムと更新されたコンテンツアイテムのインデックスを作成できます。指定した時間に達すると、追加の新しいコンテンツと更新されたコンテンツのインデックス作成は延期されます。

既定では、時間は 30 分に設定されています。次のコマンドを使用して、時間を変更できます。

```
tsm configuration set -k
metadata.ingestor.pipeline.throttlePeriodLength -v PT45M --force-
keys
```

metadata.query.limits.time

既定値: 20

これは、Catalog または メタデータAPI クエリを実行してから、タイムアウトが発生しクエリがキャンセルされるまでの最長許容時間 (秒単位) です。Tableau では、次のコマンドを使用して、タイムアウト制限を60 秒 *未満* で増やすことをお勧めします。

```
tsm configuration set -k metadata.query.limits.time –v PT30S --
force-keys
```

重要: このオプションは、「Timeout limit and node limit exceeded (タイムアウト制限 および ノード制限を超過しました)」メッセージに記載されたエラーが表示される場合にのみ変更する必要があります。タイムアウト制限を増やすと、より多くの CPU をより長く使用でき、Tableau Server 全体のタスクのパフォーマンスに影響を与える可能性があります。また、タイムアウト制限を増やすとメモリ使用率も増加する可能性があります。この場合、クエリを並列で実行するとインタラクティブなマイクロサービス コンテナで問題が発生することがあります。

metadata.query.limits.count

既定値: 20000

これは、ノードの制限を超え、クエリがキャンセルされる前に Catalog で返すことができるオブジェクトの数 (クエリ結果の数に大まかにマッピングできます) です。Tableau では、次のコマンドを使用してタイムアウト制限を 100,000 *未満* で段階的に増やしていくことをお勧めします。

```
tsm configuration set -k metadata.query.limits.count –v 3000 --
force-keys
```

重要: このオプションは、「Timeout limit and node limit exceeded (タイムアウト制限 および ノード制限を超過しました)」メッセージに記載されたエラーが表示される場合にのみ変更する必要があ

ります。ノードの制限を増やすと、メモリ使用率が増加する可能性があります。この場合、クエリを並列で実行するとインタラクティブなマイクロサービス コンテナで問題が発生することがあります。

metadata.query.throttling.enabled

バージョン: 2023.3 で追加。

既定値: `true`

メタデータAPI クエリのスロットルを有効にするかどうかを制御します。メタデータAPI クエリのスロットルは、サーバーの API 応答が全体的なパフォーマンスに悪影響を及ぼすのを防ぐために設計された機能です。`true` (デフォルト) に設定すると、メタデータAPI へのリクエストが定義済みのしきい値を超えると、`RATE_EXCEEDED` エラーが返されます。

メタデータAPI ユーザーに `RATE_EXCEEDED` エラーが頻繁に発生する場合、管理者は `metadata.query.throttl` 設定と`metadata.query.throttling.queryCostCapacity` 設定を使用してスロットルの調整を試みることができます。あるいは、管理者は `metadata.query.throttling.enabled` の値を `false` に設定することでスロットルを完全に無効にすることもできます。ただし、そうすると、この機能でパフォーマンスを保護するという利点は得られなくなります。

metadata.query.throttling.queryCostCapacity

バージョン: 2023.3 で追加。

既定値: `20000000`

メタデータAPI がクエリに応答する能力を表す数値。メタデータAPI への各リクエストには計算されたコストがあり、実行時にこの数値から差し引かれます (トークン バケット モデルを使用する場合、これはバケット内に入れることができるトークンの最大量になります)。

メタデータAPI ユーザーに `RATE_EXCEEDED` エラーが頻繁に発生する場合、管理者はスロットル設定を調整できます。`metadata.query.throttling.queryCostCapacity` の調整を試みる前に、`metadata.query.throttling.tokenRefilledPerSecond` を調整して結果を確認してください。あるいは、管理者は `metadata.query.throttling.enabled` を `false` に設定することでスロットルを完全に無効にすることもできます。ただし、そうすると、この機能でパフォーマンスを保護するという利点は得られなくなります。

metadata.query.throttling.tokenRefilledPerSecond

バージョン: 2023.3 で追加。

既定値: 5555

毎秒再生成されるメタデータAPI クエリ量を表す数値 (トークン バケット モデルを使用する場合、これは 1 秒ごとにバケットに入れられるトークンの数になります)。

メタデータAPI ユーザーに RATE_EXCEEDED エラーが頻繁に発生する場合、管理者はスロットル設定を調整できます。metadata.query.throttling.queryCostCapacity の調整を試みる前に、metadata.query.throttling.tokenRefilledPerSecond を調整して結果を確認してください。あるいは、管理者は metadata.query.throttling.enabled を false に設定することでスロットルを完全に無効にすることもできます。ただし、そうすると、この機能でパフォーマンスを保護するという利点は得られなくなります。

metricsservices.checkIntervalInMinutes

バージョン: バージョン 2024.2 で廃止。

既定値: 60

ライブ データ ソースに依存するメトリックの更新間隔を分単位で制御します。メトリクスは、サーバーがメトリックの接続ビューを介して新しいデータを確認するときに、更新されます。

metricsservices.enabled

バージョン: バージョン 2022.3 で追加。バージョン 2024.2 で廃止されました。

既定値: true

false に設定すると、サーバー上のすべてのサイトで、メトリクス コンテンツ タイプが無効になります詳細については、「サーバーのメトリクスの無効化」を参照してください。

従来のメトリクス機能の廃止

Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau

Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

metricsservices.failureCountToWarnUser

バージョン: バージョン 2024.2 で廃止。

既定値: `10`

メトリクス所有者に警告が表示されるまでに発生する必要がある、連続した更新失敗の数を制御します。デフォルトの 10 に設定すると、失敗に関する通知を所有者に送信する前にメトリクスの更新が、10 回連続して失敗する必要があります。

metricsservices.maxFailedRefreshAttempts

バージョン: バージョン 2024.2 で廃止。

既定値: `175`

メトリクス更新が保留されるまでに発生する必要がある、連続した更新失敗の数を制御します。

mobile.deep_linking.on_prem.enabled

既定値: `true`

Tableau Mobile アプリで Tableau Server へのリンクをディープリンクとして扱うかどうかを制御します。`true` に設定すると、サポートされているコンテンツタイプへのリンクがアプリで開きます。`false` に設定すると、リンクはモバイルブラウザーで開きます。詳細については、「Tableau Mobile のディープリンクを制御する」を参照してください。

monitoring.dataengine.connection_timeout

既定値: `30000`

クラスタコントローラーが接続のタイムアウトの発生を判断するまでデータエンジンを待機する時間 (ミリ秒)。既定値は 30,000 ミリ秒 (30 秒) です。

native_api.allowed_paths

既定値: ""

注: 2023 年 10 月以前の Tableau Server リリース (メンテナンス リリースを含む) では、この設定は既定ですべてのパスへのアクセスを許可するように構成されていました。

この設定を使用して、Tableau またはリモート共有に保存されているファイルへのアクセスの許可リストを指定します。このシナリオでは認可された Tableau Server ユーザーに対し、サーバー上のファイル (スプレッドシートなど) をファイル ベースのデータ ソースとして使用するワークブックの作成を許可します。

この設定により、指定したディレクトリへのアクセスのみに制限できます。そのため、tableau システム アカウントのアクセスは、データ ファイルをホストするディレクトリ パスに限定されます。

`tsm configuration set -k native_api.allowed_paths -v "path"`、ここで、*path* は許可リストに追加するディレクトリです。指定したパスのすべてのサブディレクトリは、許可リストに追加されます。複数のパスを指定する場合は、この例のようにセミコロンで区切ります。

```
tsm configuration set -k native_api.allowed_paths -v
"/datasources;/HR/data"
```

重要: この設定で指定したファイル パスが存在し、システム アカウントからアクセス可能であることを確認してください。

native_api.connection.limit.<connection class>

指定したデータ ソースに対する並列クエリ制限を設定します (接続クラス)。これにより、データ ソースのグローバル制限が上書きされます。

native_api.connection.globallimit

既定値: 16

並列クエリ用のグローバル制限。既定値は 16 です (既定値が 8 の Amazon Redshift を除く)。

native_api.ExplainDataEnabled

既定値: `true`

このオプションで、「データの説明を見る」をサーバーで有効にするか無効にするかを制御します。「データの説明を見る」の詳細については、Tableau ヘルプの「「データの説明を見る」を使い始める」を参照してください。

このオプションは Tableau Server バージョン 2019.3 で追加されました。

native_api.force_alternative_federation_engine

既定値: `false`

単一ファイル接続と SQL 単一データベース接続からデータを結合するときに、操作制限をオーバーライドします。このオプションを `True` に設定して、Tableau で、ライブデータベース接続を使用して強制的に結合を処理します。

native_api.ProtocolTransitionLegacyFormat

既定値: `false`

制約付き委任にレガシー名フォーマットを使用します。

名前形式がバージョン 10.1 で変更となり、クロスドメインプロトコル トランジッション (S4U) が使用できるようになりました。これにより既存の構成に問題が発生し、クロスドメインプロトコル トランジッションが必要ない場合は、Tableau Server を `true` に設定して古い動作を使用してください。

native_api.unc_mountpoints

既定値: なし

Tableau Server on Linux からアクセスされる共有 Windows ディレクトリの UNC および FQDN パスを指定します。各パスは対応する auto.cifs ファイルで参照されている必要もあります。各パスはセミコロンで区切ってください。次に例を示します。

```
'//filesrv01/development;/mnt/filesrv01/development;//filesrv01.example.lan/development;/mnt/filesrv01/development'
```

その後、native_api.unc_mountpoints 値に対する更新を行うと、既存の値が上書きされます。したがって、Windows 共有を追加するたびに、更新された値のすべての共有を含める必要があります。

詳細については、コミュニティウィキ トピックの「Windows 共有ディレクトリへの接続 (英語)」を参照してください。

native_api.InitializeQueryCacheSizeBasedOnWeights

既定値: True

クエリキャッシュのサイズを、使用できるシステム メモリの量に基づいて自動的に初期化するかどうかを設定します。クエリキャッシュには、論理クエリキャッシュ、メタデータキャッシュ、ネイティブ クエリキャッシュがあります。既定では、この機能が有効になっています。

native_api.LogicalQueryCacheMaxAllowedWeight

既定値: 70

クエリキャッシュの合計サイズにおける論理クエリキャッシュサイズの制限の重み。

native_api.MetadataQueryCachMaxAllowedWeight

既定値: 4

クエリキャッシュの合計サイズにおけるメタデータ クエリキャッシュサイズの制限の重み。

native_api.NativeQueryCacheMaxAllowedWeight

既定値: 26

クエリキャッシュの合計サイズにおけるネイティブ クエリキャッシュサイズの制限の重み。

native_api.QueryCacheEntryMaxAllowedInPercent

既定値: 60

クエリキャッシュに格納できるクエリ結果の最大サイズを指定します。クエリキャッシュの合計サイズに対する割合として設定されます。たとえば、論理クエリのキャッシュサイズが 100 MB で

native_api.QueryCacheEntryMaxAllowedInPercent が 60% に設定されている場合、60 MB 未満のクエリ結果のみを論理クエリキャッシュに格納できます。

native_api.QueryCacheMaxAllowedMB

クエリキャッシュの最大サイズです (メガバイト単位)。既定値は、システム メモリの量によって異なります。クエリキャッシュには、論理クエリキャッシュ、メタデータキャッシュ、ネイティブ クエリキャッシュがあります。次の表を使用して、既定値を決定します。

システム メモリ	**Tableau Server のデフォルト値**	**Tableau Desktop のデフォルト値**
64 GB 以上	3200 MB	1600 MB
32 GB から 64 GB まで	2400 MB	1200 MB
16 GB から 32 GB まで	1600 MB	800 MB
16 GB 以下	800 MB	400 MB

native_api.ResourceLuidsInSnowflakeSessionQueryTagsEnabled

既定値: true

true の場合は、クエリのタグ付け情報が Snowflake データベースに送信されます。クエリのタグ付けは、データベース クエリを生成している Tableau アセットを特定するのに役立ちます。false の場合は、Snowflake のクエリのタグ付けが無効になります。

native_api.ResourceLuidsInTeradataSessionQueryBandsEnabled

既定値: true

true の場合は、クエリのタグ付け情報が Teradata データベースに送信されます。クエリのタグ付けは、データベース クエリを生成している Tableau アセットを特定するのに役立ちます。false 場合は、Teradata のクエリのタグ付けが無効になります。

native_api.UserInfoInGeneratedSQLEnabled

既定値: `false`

Tableau Server 上のすべてのコンテンツに対して、[クエリのタグ付け] を有効にするかどうかを決定します。`true` の場合は、Tableau からデータベースに送信されるクエリに、クエリのソースに関するメタデータが含まれます。お客様のデータベースのログに記録された内容は、パフォーマンスやその他の問題のトラブルシューティングに使用することができます。

nlp.concepts_shards_count

既定値: 1

> **注:** ほとんどの Tableau Server インストールでは、既定のシャーディング数の値で十分です。

シャードに格納されている「データに聞く」、フィールド名、フィールドの同義語、および分析用語の Concepts インデックスのデータシャードの数を制御します。

- 2022.1 以降のバージョンのインデックスおよび検索サーバー。
- 2019.1 ~ 2021 の Elastic Server。4

シャーディング数は、検索インデックスをパーティション分割して、インデックスの合計サイズを小さくし、「データに聞く」のセマンティックパーサーのパフォーマンスを向上させます。実行している Tableau Server のバージョンに応じて、`elasticserver.vmopts` または `indexandsearchserver.vmopts` を使用してヒープサイズを増やすことに加えて、シャーディング数の調整は、パフォーマンスを向上させるために講じることができるもう1 つの方策です。

Tableau では、50 GB ごとにシャーディング数を 1 ずつ増やすことをお勧めします。シャーディング数を調整する必要がある回数を減らすには、現在のインデックスに 50% を加算して、合計インデックスサイズを計算します。たとえば、合計インデックスサイズが 50 GB 未満の場合は、1 シャーディングで十分です。実際のパフォーマンスは、サーバー、インデックスサイズが大きくなる速度、およびその他の要因によって異なります。

- 0 から 50 GB: 1
- 50 GB から 100 GB: 2

- 100 GB から150 GB: 3

次のコマンドを使用して、Concepts インデックスのシャーディングを既定値から2 に増やすことができます。

```
tsm configuration set -k nlp.concepts_shards_count -v 2
```

nlp.values_shards_count

既定値: 1

シャードに格納されている「データに聞く」、フィールド名、フィールドの同義語、および分析用語の Concepts インデックスのデータシャードの数を制御します。

- 2022.1 以降のバージョンのインデックスおよび検索サーバー。
- 2019.1 ~ 2021 の Elastic Server。4

シャーディング数は、検索インデックスをパーティション分割して、インデックスの合計サイズを小さくし、「データに聞く」のセマンティック パーサーのパフォーマンスを向上させます。実行している Tableau Server のバージョンに応じて、`elasticserver.vmopts` または `indexandsearchserver.vmopts` を使用してヒープ サイズを増やすことに加えて、シャーディング数の調整は、パフォーマンスを向上させるために講じることができるもう1 つの方策です。

Tableau では、50 GB ごとにシャーディング数を 1 ずつ増やすことをお勧めします。シャーディング数を調整する必要がある回数を減らすには、現在のインデックスに 50% を加算して、合計インデックス サイズを計算します。たとえば、合計インデックス サイズが 50 GB 未満の場合は、1 シャーディングで十分です。実際のパフォーマンスは、サーバー、インデックスサイズが大きくなる速度、およびその他の要因によって異なります。

- 0 から50 GB: 1
- 50 GB から100 GB: 2
- 100 GB から150 GB: 3

次のコマンドを使用して、Values インデックスのシャーディング数を既定値から2 に増やすことができます。

```
tsm configuration set -k nlp.values_shards_count -v 2
```

nlp.defaultNewSiteAskDataMode

既定値: `disabled_by_default`

このオプションを使用して、サイトの作成時に「データに聞く」モードの初期値を設定します。詳細については、サイトでの「データに聞く」(Ask Data) 機能の無効化または有効化を参照してください。

有効なオプションは `disabled_by_default` および `disabled_always` です。

このオプションは Tableau Server バージョン 2019.4.5 および 2020.1.3 で追加されました。

noninteractive.vmopts

既定値: `"-XX:+UseConcMarkSweepGC -Xmx<default_value>g -XX:+ExitOnOutOfMemoryError"`

既定値は、システム メモリの量によって異なります。JVM の最大ヒープ サイズは、システム RAM 全体の 6.25% にスケールされます。

このオプションにより、Tableau Catalog の取り込みにおける JVM の最大ヒープ サイズが制御されます。既定値は自動的にスケールされます。どうしても必要な場合にのみ、このオプションで引数 `-Xmx<default_value>g` を変更して既定値を上書きします。たとえば、次のコマンドを使用して、最大ヒープ サイズを 2 GB に増やすことができます。

```
tsm configuration set -k noninteractive.vmopts -v "-XX:+UseConcMarkSweepGC -Xmx2g -XX:+ExitOnOutOfMemoryError"
```

詳細については、非インタラクティブなマイクロサービス コンテナーのメモリを参照してください。

pgsql.port

既定値: `8060`

PostgreSQL がリッスンするポート。

pgsql.preferred_host

優先リポジトリがインストールされているノードのコンピューター名を指定します。`--preferred` または `-r` オプションが tsm topology failover-repository コマンドと共に指定されている場合にはこの値が使用されます。

例:

```
tsm configuration set -k pgsql.preferred_host -v "<host_name>"
```

注: `host_name` は大文字と小文字が区別され、`tsm status -v` の出力に表示されるノード名と一致する必要があります。

pgsql.ssl.ciphersuite

既定値: `HIGH:MEDIUM:!aNULL:!MD5:!RC4`

リポジトリに対する SSL で許容される暗号アルゴリズムを指定します。

許容される値と書式要件については、Postgres Web サイトの「ssl_ciphers」を参照してください。

pgsql.ssl.max_protocol_version

既定値: `TLSv1.3`

SSL でリポジトリに接続する際に使用する SSL/TLS プロトコル バージョンの最大値を設定します。

有効な値: `TLSv1`、`TLSv1.1`、`TLSv1.2`、`TLSv1.3`

pgsql.ssl.min_protocol_version

既定値: `TLSv1.2`

SSL でリポジトリに接続する際に使用する SSL/TLS プロトコル バージョンの最小値を設定します。

有効な値: `TLSv1`、`TLSv1.1`、`TLSv1.2`、`TLSv1.3`

pgsql.verify_restore.port

既定値: 8061

PostgreSQL データベースの整合性の検証に使用されるポート。詳細については、tsm maintenance backupを参照してください。

ports.blocklist

バージョン: バージョン 2021.1 で追加

既定値: 自動ポート割り当てに使用される範囲内でブロックされているポートはありません。

ポートを動的に割り当てるときに Tableau で使用しないポート割り当て範囲内のポートを指定するために使用します。この機能は、別のアプリケーションがその範囲内にあるポートを使用していることを知っている場合に役に立ちます。複数のポートをカンマで区切ります。例:

```
tsm configuration set -k ports.blocklist -v 8000,8089, 8090
```

ports.blocklist の使用の詳細については、範囲内の特定のポートをブロックするを参照してください。

recommendations.enabled

既定値: true

データソースとテーブルの推奨事項 (Tableau Desktop の場合) とビューの推奨事項 (Tableau Server の場合) を提供する推奨機能を制御します。推奨事項は、コンテンツの人気度、あるいは現在のユーザーと似た別のユーザーが使用しているコンテンツに基づいて提案されます。

recommendations.vizrecs.enabled

既定値: true

Tableau Server ユーザーのビューに関する推奨事項を制御します。このオプションは recommendations.enabled の子であり、親オプションが false に設定されている場合は無効です。親オプションが true に設定され、このオプションが false に設定されている場合でも、データソースとテーブルは Tableau Desktop ユーザーに推奨されますが、Tableau Server のビューに関する推奨事項は無効になります。

redis.max_memory_in_mb

既定値: 1024

キャッシュサーバーの外部クエリキャッシュのサイズをメガバイト単位で指定します。

refresh_token.absolute_expiry_in_seconds

既定値: 31536000

更新トークンと個人用アクセストークン (PAT) の絶対的な有効期限の秒数を指定します。

更新トークンは、接続されたクライアント (Tableau Desktop、Tableau Prep Builder、Tableau Mobile など) が初回サインイン後に Tableau Server への認証を行うために使用されます。

制限を削除するには、値を -1 に設定します。更新トークンとPAT を無効にするには、自動クライアント認証を無効にするを参照してください。

refresh_token.idle_expiry_in_seconds

既定値: 1209600

アイドル状態の更新トークンの有効期限が切れる秒数を指定します。更新トークンは、接続されたクライアント (Tableau Desktop、Tableau Prep Builder、Tableau Mobile など) が初回サインイン後に Tableau Server への認証を行うために使用されます。制限を削除するには、値を -1.に設定します。

refresh_token.max_count_per_user

既定値: 24

各ユーザーに発行できる更新トークンの最大数を指定します。ユーザーセッションの最大数が十分でない場合は、この値を増やすか、-1 に設定して更新トークンの制限を完全に削除してください。

rsync.timeout

既定値: 600

ファイルの同期を完了するまでの秒単位の最大許容時間 (600 秒 = 10 分)。ファイルの同期は、高可用性の構成、またはデータエンジンとリポジトリの移動プロセスの一部として発生します。

schedules.display_schedule_description_as_name

既定値: `false`

サブスクリプションまたは抽出の更新 (既定) を作成するときにスケジュール名を表示するか、スケジュールの時間と頻度を説明する"スケジュール頻度の説明" 名が表示するかを制御します。タイムゾーンを区別するスケジュールの名前を表示するようTableau Server を設定するには、この値を `true` に設定します。

true となっている場合、"スケジュール頻度の説明" はスケジュール リストページのスケジュール名の後にも表示されます。

schedules.display_schedules_in_client_timezone

既定値: `true`

true の場合、"スケジュール頻度の説明" をユーザーのタイムゾーンで表示します (クライアントブラウザーのタイムゾーンを使用して"スケジュール頻度の説明")。

schedules.ignore_extract_task_priority

既定値 (ブール値): `False`

この設定は、ジョブをキューからプルするタイミングを定めるジョブ ランクを決定する上で、タスクの優先度を考慮するかどうかを制御します。`true` に設定すると、タスクの優先度の編集が無効になり、ジョブ ランクの決定にはスケジュールの優先度のみが考慮されます。

searchserver.connection_timeout_milliseconds

バージョン: バージョン 2019.1 で追加。バージョン 2022.3 で非推奨になりました。バージョン 2023.3 で廃止されました。

既定値、ミリ秒単位: `100000`

検索および参照クライアントが検索および参照サーバーへの接続を確立するまでの待機時間をミリ秒単位で指定します。

特にビジーな Tableau Server コンピューターで、またはログ エラーが表示される場合「失敗した動物園のキーパーの正常性チェック。SOLR の起動を拒否する」がこの値を大きくします。

詳細については、クライアントセッションのタイムアウトを参照してください。

searchserver.index.bulk_query_user_groups

バージョン: バージョン 2022.3 で廃止。

既定値: `true`

CSV ファイルでユーザーをインポートまたは削除するときに、サイトユーザーの照会を一括で行うかどうかを指定します。`true` (既定) に設定すると、インデックス作成が一括で行われます。

searchserver.javamemopts

バージョン: バージョン 2019.1 で追加。2023.3 で廃止

既定値: `-Xmx512m -Xms512m -XX:+ExitOnOutOfMemoryError -XX:-UsePerfData`

SOLR の JVM オプションを決定します。

構成可能なすべてのオプションの中で、検索サーバーをチューニングする際に、`-Xmx`パラメーターによって構成される最大ヒープメモリが最も重要です。ほとんどの場合、Tableau Server コンピューター上の使用可能な物理メモリに基づいて、最大 24 GB までの可能な限り高い値を設定する必要があります。最大ヒープ メモリのみを変更するには、既定の文字列全体を指定しますが、`-Xmx` の値だけを変更します。

`-Xmx`の有効な値は Tableau Server コンピューター上の使用可能なメモリに依存しますが、24 GB を超えることはできません。詳細については、「最大ヒープメモリの検索と参照」を参照してください。

searchserver.startup.zookeeper_healthcheck_timeout_ms

バージョン: バージョン 2020.1 で追加。バージョン 2023.3 で廃止されました。

既定値、ミリ秒単位: `300000`

起動時に、Tableau Server が正常な Zookeeper ヘルスチェックを待機する時間をミリ秒単位で指定します。

特にビジーな Tableau Server コンピューターで、またはログ エラーが表示される場合「失敗した動物園のキーパーの正常性チェック。SOLR の起動を拒否する」がこの値を大きくします。

詳細については、起動時の ZooKeeper 接続のヘルスチェックのタイムアウトを参照してください。

searchserver.zookeeper_session_timeout_milliseconds

バージョン: バージョン 2022.3 で廃止。

既定値、ミリ秒単位: `100000`

検索および参照クライアントが調整サービス (Zookeeper) への接続を確立するまで待機する時間をミリ秒単位で指定します。

詳細については、クライアントセッションのタイムアウトを参照してください。

ServerExportCSVMaxRowsByCols

バージョン: バージョン 2020.3 で追加。

既定値: `0` (制限なし)

ビュー データから CSV ファイルにダウンロードできるデータのセルの最大数を指定します。既定では、制限はありません。セルの数を指定します。たとえば、次のコマンドを使用して、300 万の制限を設定します。

```
tsm configuration set -k ServerExportCSVMaxRowsByCols -v 3000000

tsm pending-changes apply
```

service.jmx_enabled

既定値: `false`

に設定すると`true`、必要に応じて監視とトラブルシューティング用に JMX ポートを使用できます。

service.max_procs

既定値: <number>

最大サーバープロセス数。

service.port_remapping.enabled

既定値: `true`

既定または構成済みのポートが利用できない場合に、Tableau Server がポートの動的な再配置を試みるかどうかを決定します。`false` に設定すると、動的なポートの再配置は無効になります。

sheet_image.enabled

既定値: `true`

ビューの画像を REST API で取得できるかどうかを制御します。詳細については、「REST API リファレンス」を参照してください。

ssl.ciphersuite

既定値: `HIGH:MEDIUM:!EXP:!aNULL:!MD5:!RC4:!3DES:!CAMELLIA:!IDEA:!SEED`

ゲートウェイに対する SSL で許容される暗号アルゴリズムを指定します。

許容される値と書式要件については、Apache Web サイトの「SSLCipherSuite」を参照してください。

ssl.client_certificate_login.blocklisted_signature_algorithms

既定値:

- バージョン 2020.4.0:

  ```
  sha1withrsaencryption,
  sha1withrsa
  ```

- バージョン 2020.4.1 以降:

```
sha1withrsaencryption,
sha1withrsa,
sha1withrsaandmgf1,
sha1withdsa,
sha1withecdsa
```

既定値は、SHA-1 署名アルゴリズムを使用した証明書をブロックします。SSL に対してブロックされるクライアント署名アルゴリズムを指定します。すべての署名アルゴリズムのブロックを無効にするには、このキーを空の引用符を付けて実行します。

このキーの詳細については、ナレッジ ベースの記事「証明書が SHA-1 で署名されている場合、アップグレード後に相互 SSL が失敗する」を参照してください。

ssl.client_certificate_login.min_allowed.elliptic_curve_size

既定値: `256`

ECDSA クライアント証明書の楕円曲線サイズに求められる最小値を指定します。この証明書で Tableau Server と相互 SSL を介して認証を行います。クライアントが提示した ECDSA クライアント証明書の曲線サイズがこの最小値を満たさない場合、認証要求は失敗します。

このオプションは Tableau Server バージョン 2021.1 で追加されました。

ssl.client_certificate_login.min_allowed.rsa_key_size

既定値: `2048`

RSA クライアント証明書のキーサイズに求められる最小値を指定します。この証明書で Tableau Server と相互 SSL を介して認証を行います。クライアントが提示した RSA クライアント証明書のキーサイズがこの最小値を満たさない場合、認証要求は失敗します。

このオプションは Tableau Server バージョン 2021.1 で追加されました。

ssl.protocols

既定値: `all +TLSv1.2 -SSLv2 -SSLv3 -TLSv1.3`

Tableau Server がゲートウェイへの TLS 接続に対してサポートする SSL プロトコルを指定します。許容値は、Apache SSL プロトコル ディレクティブから導き出します。セキュリティ強化チェックリストの説明に従って、SSL プロトコルを構成することをお勧めします。

storage.monitoring.email_enabled

既定値: `false`

メール通知でサーバーのディスク容量の監視を有効にするかどうかを制御します。既定では、メール通知が有効になっています。ディスク容量の監視に対する通知を有効にするには、これを `true` に設定します。

SMTP では、通知を送信するように設定する必要があります。詳細については、SMTP セットアップの構成を参照してください。

storage.monitoring.warning_percent

既定値: `20`

合計のディスク容量に対する残りのディスク容量の割合に関する警告しきい値。ディスク容量がこのしきい値を下回ると、警告通知が送信されます。

storage.monitoring.critical_percent

既定値: `10`

合計のディスク容量に対する残りのディスク容量の割合に関する重要のしきい値。ディスク容量がこのしきい値を下回ると、重要の通知が送信されます。

storage.monitoring.email_interval_min

既定値: `60`

ディスク容量の監視が有効になっていて、しきい値を超えた場合にメール通知を送信する頻度 (分単位)。

storage.monitoring.record_history_enabled

既定値: `true`

空きディスク領域の履歴を保存し、管理ビューで表示できるようにするかどうかを決定します。監視履歴の保存を無効にするには、`storage.monitoring.record_history_enabled` を `false` に設定します。

subscriptions.enabled

既定値: `false`

サブスクリプションがシステム全体で構成可能かどうかを制御します。詳細については、サブスクリプションのサーバー設定を参照してください。

subscriptions.timeout

既定値: `1800`

ビューで 1 つのワークブック サブスクリプション タスクのレンダリングがタイムアウトするまでの時間 (秒)。ビューのレンダリング中にこの制限時間に達すると、レンダリングは続行されますが、*ワークブック内の後続のビューはレンダリングされず*、ジョブがエラーで終了します。単一ビューのワークブックの場合、この値を指定しても、タイムアウトが原因でレンダリングが停止することはありません。

svcmonitor.notification.smtp.enabled

既定値: `false`

サーバープロセス イベントに対してメール通知を有効にするかどうかを制御します。既定では通知は、プロセスがダウン、フェールオーバー、または再起動したときに送信されます。サーバープロセスの通知を有効にするには、これを `true` に設定します。

SMTP では、通知を送信するように設定する必要があります。詳細については、SMTP セットアップの構成を参照してください。

svcmonitor.notification.smtp.mime_use_multipart_mixed

バージョン: バージョン 2020.1.8、2020.2.5、2020.3.1 で追加

既定値: `false`

サブスクリプションの HTML MIME 添付ファイルを *マルチパート/関連* (デフォルト) として送信するか、*マルチパート/混合 として送信するかを制御*します。

まれに、Tableau Server から送信されたメールをメール クライアントが適切に解析できない場合があります。多くの場合、この問題は、プロパティを `true` に設定すると修正できます。既知のクライアントには、iOS Mail とMicrosoft Outlook (Exchange S/MIME 暗号化と組み合わせた場合) が含まれます。

tabadmincontroller.auth.expiration.minutes

既定値: `120`

セッション Cookie が有効な期間を制御します。既定では、120 分に設定されています。この値によって、ノードbootstrap ファイル内の埋め込みの認証資格情報がどのくらい有効であるかも決まります。詳細については、tsm topology nodes get-bootstrap-fileを参照してください。

tdsservice.log.level

バージョン: バージョン 2020.3.0 で追加。

既定値: `info`

データソースプロパティサービスのロギング レベル。このロギング レベルは動的に構成できるため、ロギング レベルだけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

tomcat.http.maxrequestsize

既定値: `16380`

HTTP 要求上で Apache ゲートウェイを通じて渡すことが許可されているヘッダー コンテンツの最大サイズ (バイト)。ヘッダーがこのオプションで設定された値を超えると、ブラウザーに HTTP Error 413 (要求するエンティティが長すぎます) や認証エラーなどのエラーが表示されます。

`tomcat.http.maxrequestsize` の値が小さいと、認証エラーが発生する場合があります。Active Directory (SAML および Kerberos) と統合するシングル サインオン ソリューションでは、多くの

場合、HTTP ヘッダーに大きな認証トークンが必要です。本番環境に展開する前に、必ず HTTP 認証シナリオをテストしてください。

`gateway.http.request_size_limit` オプションを、このオプションに設定した値と同じ値に設定することをお勧めします。

tomcat.http.proxyHost

IdP へ OpenID リクエストを行うフォワードプロキシのホスト名を指定します。OpenID Connect に対応するように Tableau Server を構成するを参照してください。

tomcat.http.ProxyPort

IdP へ OpenID リクエストを行うフォワードプロキシのポートを指定します。OpenID Connect に対応するように Tableau Server を構成するを参照してください。

tomcat.https.proxyHost

IdP へ OpenID リクエストを行うフォワードプロキシのホスト名を指定します。OpenID Connect に対応するように Tableau Server を構成するを参照してください。

tomcat.https.ProxyPort

IdP へ OpenID リクエストを行うフォワードプロキシのポートを指定します。OpenID Connect に対応するように Tableau Server を構成するを参照してください。

tomcat.https.port

既定値: `8443`

Tomcat の SSL ポート(未使用)。

tomcat.server.port

既定値: `8085`

Tomcat がシャットダウン メッセージをリッスンするポート。

tomcat.useSystemProxies

既定値: `false`

Tomcat コンポーネント(OpenID) が、ローカルの Windows オペレーティング システムのフォワードプロキシ構成にアクセスする必要があるかどうかを指定します。OpenID Connect に対応するように Tableau Server を構成するを参照してください。

tomcatcontainer.log.level

既定値: `info`

インタラクティブなマイクロサービス コンテナとインタラクティブでないマイクロサービス コンテナのマイクロサービスのロギング レベル。バージョン 2020.4 以降では、このロギング レベルを動的に構成できるため、この設定だけを変更する場合は Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

tsm.log.level

既定値: `info`

TSM サービスのロギング レベル。これらのログには、管理エージェント、管理コントローラー、クライアント ファイル サービス、クラスタ コントローラー、サービス マネージャー、ライセンス サービスなどの TSM サービスに問題が発生した場合に役立つ情報が含まれています。この構成キーによって、調整サービスやメンテナンス プロセスのロギング レベルが変わることはありません。詳細については、ロギング レベルの変更とTableau Server プロセスを参照してください。

tsm.controlapp.log.level

既定値: `info`

`control_<app>` サービスのロギング レベル。これらのログには、TSM や Tableau Server プロセスの開始や再構成に関する問題が発生した場合に役立つ情報が含まれています。詳細については、ロギング レベルの変更を参照してください。

usernotifications.reap_after_days

既定値: `30`

ユーザー通知がサーバーから削除されるまでの日数。

vizportal.adsync.update_system_user

既定値: `false`

Active Directory グループが Tableau Server 内で同期されている場合、ユーザーの電子メール アドレスと表示名を変更するかどうか (Active Directory で変更される場合を含む) を指定します。同期中にユーザーの電子メール アドレスと表示名が更新されていることを確認するには、`vizportal.adsync.update_system_user` を `true` に設定してからサーバーを再起動します。

vizportal.alwaysUseEmbeddedShareLinks

バージョン: バージョン 2021.3.0 で追加。

既定値: `false`

[リンクのコピー] オプションに "embed=y" パラメーターを含めるかどうかを指定します。バージョン 2019.4 以降では、このパラメーターは既定で含まれていません。この構成キーを true に設定すると、動作が変更され、"embed=y" パラメーターが含まれるようになります。**[リンクのコピー]** のオプションを使用して Web ページに埋め込むためのリンクを共有する方法の詳細については、Tableau Desktop および Web オーサリング ヘルプの「Web ページへのビューの埋め込み」を参照してください。

vizportal.art_skip_list

バージョン: バージョン 2024.2 で追加。

既定値: `null`

アクティビティとリソースのトレース (ART) が有効になっているとき、ART を使用せずに大量の不要なデータを生成する Tableau Server 機能の側面を指定するには、この構成キーを使用します。

このキーは、アプリケーション サーバー (VizPortal) の問題のトラブルシューティングのために、`vizportal.log_art_java` および `vizportal.enable_art` といっしょに使用されます。[何に設定されるかについての情報をここに入力する必要があります。] に設定すると、この構成設

定を使用する方法の詳細については、アプリケーション サーバーの問題のトラブルシューティングを参照してください。

vizportal.commenting.delete_enabled

既定値: `true`

`true` に設定すると、ユーザーはビューのコメントを削除できます。コメントの作成者、コンテンツ所有者、適切なサイト ロールを持つプロジェクト リーダー、または管理者である場合、コメントを削除できます。完全なプロジェクト リーダー アクセス権を得るために必要なサイト ロールの詳細については、プロジェクト レベルの管理を参照してください。

vizportal.csv_user_mgmt.index_site_users

バージョン: バージョン 2022.3 で非推奨。バージョン 2023.3 で廃止 (完全に削除) されました。

既定値: `true`

CSV ファイルでユーザーをインポートまたは削除する際、サイト ユーザーのインデックス作成をユーザーごとに行うかどうかを指定します。`true` (既定) に設定すると、各ユーザーが追加または削除されるときにインデックス作成が実行されます。CSV ファイル全体の処理が完了するまでサイト ユーザーのインデックス作成を遅らせるには、これを `false` に設定します。

vizportal.csv_user_mgmt.bulk_index_users

バージョン: バージョン 2022.3 で非推奨。バージョン 2023.3 で廃止 (完全に削除) されました。

既定値: `false`

CSV ファイルでユーザーをインポートまたは削除するときに、サイト ユーザーのインデックス作成を一括で行うかどうかを指定します。`false` (既定) に設定すると、インデックス作成がユーザーごとに行われます。CSV ファイルを処理した後、インデックス作成を一括で行うには、これを `true` に設定します。

vizportal.enable_art

バージョン: バージョン 2024.2 で追加。

既定値: `false`

この設定キーは、アプリケーション サーバー (VizPortal) の問題のトラブルシューティングのために、`vizportal.log_art_java` および `vizportal.art_skip_list` といっしょに使用されます。true に設定すると、アプリケーション サーバーでアクティビティとリソースのトレースが有効になります。この構成設定を使用する方法の詳細については、アプリケーション サーバーの問題のトラブルシューティングを参照してください。

vizportal.log_art_java

バージョン: バージョン 2024.2 で追加。

既定値: `false`

この設定キーは、アプリケーション サーバー (VizPortal) の問題のトラブルシューティングのために、`vizportal.enable_art` および `vizportal.art_skip_list` といっしょに使用されます。true に設定すると、アプリケーション サーバーでアクティビティとリソースのトレースが有効になります。この構成設定を使用する方法の詳細については、アプリケーション サーバーの問題のトラブルシューティングを参照してください。

vizportal.log.level

既定値: `info`

vizportal Java コンポーネントのログ レベル。ログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizportal/*.log` に書き込まれます。

詳細については、`debug` に設定してください。デバッグ設定を使用するとパフォーマンスに大きな影響を及ぼす可能性があるため、Tableau サポートにより指示される場合にしか使用しないでください。

バージョン 2020.4.0 以降では、この設定は動的に構成できるため、この設定だけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

vizportal.oauth.connected_apps.max_expiration_period_in_minutes

バージョン: バージョン 2021.4 で追加。

既定値: `10`

JSON Web トークン (JWT) が有効な最大期間 (分単位)。JWT の検証時に、Tableau Server は、JWT で指定された期間がこの既定値を超えていないことを確認します。この設定は、Tableau REST API を使用して Tableau Server で Tableau 接続アプリが構成されている場合に使用されます。

たとえば、最大期間を 5 分に変更するには、次のコマンドを実行します。

```
tsm configuration set -k vizportal.oauth.external_authorization_
server.max_expiration_period_in_minutes -v 5
```

vizportal.oauth.external_authorization.enabled

バージョン: バージョン 2021.4 で追加。

既定値: `false`

Tableau Server 2024.2 以降では、Tableau Server に対して **[接続済みアプリを有効にする]** オプションがオンになっています。Tableau Server 2023.2 以前で **[Enable OAuth Access for Embedding Content (コンテンツを埋め込むための OAuth アクセスを有効にする)]** オプションを有効にするかどうかを指定します。

このオプションを使用して、Tableau Server で外部認証サーバー (EAS) を登録し、アプリケーション統合を有効にすることができるようにします。詳細については、OAuth 2.0 信頼を使用して接続済みアプリを設定するを参照してください。

このオプションを有効にするには、次のコマンドを実行します。

```
tsm configuration set -k vizportal.oauth.external_
authorization.enabled -v true
```

vizportal.oauth.external_authorization_server.blocklisted_jws_algorithms

バージョン: バージョン 2021.4 で追加。

既定値: `ES256K`

外部認証サーバー (EAS) が登録されている場合、または接続アプリが構成されている場合、このコマンドを使用して、JSON Web トークン (JWT) ヘッダーで使用される署名アルゴリズムを指定できます。詳細については、OAuth 2.0 信頼を使用して接続済みアプリを設定するまたはTableau 接続済みアプリを使用してアプリケーションを統合するを参照してください。

たとえば、必要に応じて、次のコマンドを実行してアルゴリズムを削除できます。

```
tsm configuration set - k vizportal.oauth.external_authorization_
server.blocklisted_jws_algorithms -v
```

重要: 上記のコマンド例では、安全でない署名アルゴリズムが許可されており、エラーのトラブルシューティングのみに使用する必要があります。

vizportal.oauth.external_authorization_server.issuer

バージョン: バージョン 2021.4 で追加。

既定値: `null`

必須。このコマンドを使用して、発行元の URL を指定します。Tableau Server で外部認証サーバー (EAS) を登録するには、発行元の URL が必要です。詳細については、OAuth 2.0 信頼を使用して接続済みアプリを設定するを参照してください。

たとえば、EAS が Okta の場合、次のようなコマンドを実行できます。

```
tsm configuration set -k vizportal.oauth.external_authorization_
server.issuer -v "https://dev-
12345678.okta.com/oauth2/abcdefg9abc8eFghi76j5"
```

vizportal.oauth.external_authorization_server.jwks

バージョン: バージョン 2021.4 で追加。

既定値: `null`

外部認証サーバー (EAS) が登録されている場合、このコマンドを使用して、JSON Web キー セット (JWKS) の URL を指定できます。ID プロバイダー (IdP) が外部認証サーバーのメタデータ エンドポイントを公開しない場合は、JWKS URL が必要です。

たとえば、IdP が Amazon Cognito の場合、次のようなコマンドを実行できます。

```
tsm configuration set -k vizportal.oauth.external_authorization_
server.jwks -v "https://cognito-idp.us-west-2.amazonaws.com/us-west-
2_Ab129faBb/.well-known/jwks.json"
```

vizportal.oauth.external_authorization_server.max_expiration_period_in_minutes

バージョン: バージョン 2021.4 で追加。

既定値: `10`

JSON Web トークン (JWT) が有効な最大期間 (分単位)。JWT の検証時に、Tableau Server は、JWT で指定された期間がこの既定値を超えていないことを確認します。この設定は、Tableau Server で EAS が登録されている場合に使用されます。詳細については、OAuth 2.0 信頼を使用して接続済みアプリを設定するを参照してください。

たとえば、最大期間を 5 分に変更するには、次のコマンドを実行します。

```
tsm configuration set -k vizportal.oauth.external_authorization_
server.max_expiration_period_in_minutes -v 5
```

vizportal.openid.client_authentication

OpenID Connect のカスタム クライアント認証手法を指定します。

`client_secret_post` を必要とする IdP を使用するように Tableau Server を構成するには、この値を `client_secret_post` に設定します。

Salesforce IDP に接続する場合がその例で、これが必要となります。

vizportal.openid.essential_acr_values

バージョン: バージョン 2020.4 で追加。

OpenID Connect IdP を必須のクレーム要求として提示するために、認証コンテキストクラス参照 (ACR) 値のリストを指定します。IdP は、期待される基準を満たした認証を行う責任があります。構成キー `vizportal.openid.essential_acr_values` が設定されている場合、Tableau Server は証明書利用者として機能し、トークン応答で ACR クレームを検査します。Tableau Server は、期待する構成キーの値と ACR クレームが一致しない場合にのみ警告します。

このオプションを設定するには、ACR 値を優先順に入力し、二重引用符で囲みます。この例のように、複数の値はコンマとスペースで区切る必要があります。

```
tsm configuration set -k vizportal.openid.essential_acr_values -v
"value1, value2"
```

vizportal.openid.full_server_request_logging_enabled

既定値: `false`

OpenID 動作の完全なログを記録するかどうかを指定します。

OpenID Connect の問題をトラブルシューティングするときは、これを `true` に設定します。これにより詳細なログを収集でき、トラブルシューティングを役立てることができます。

ログの収集とトラブルシューティングが終了したら、ログ関連のすべての設定と同じように、このキーをデフォルト値 (`false`) にリセットすることをお勧めします。これにより、ログに記録される情報量が制限され、ログ ファイルのサイズが最小限に抑えられます。

vizportal.openid.voluntary_acr_values

バージョン: バージョン 2020.4 で追加。

OpenID Connect IdP を任意のクレーム要求として提示するために、認証コンテキストクラス参照 (ACR) 値のリストを指定します。IdP は、期待される基準を満たした認証を行う責任があります。構成キー `vizportal.openid.voluntary_acr_values` が設定されている場合、Tableau Server は証明書利用者として機能し、トークン応答で ACR クレームを検査します。ACR クレー

ムが見つからない場合、または指定したクレーム値が期待する構成キーの値と一致しない場合、認証要求は失敗します。

このオプションを設定するには、ACR 値を優先順に入力し、二重引用符で囲みます。この例のように、複数の値はコンマとスペースで区切る必要があります。

```
tsm configuration set -k vizportal.openid.voluntary_acr_values -v
"value1, value2"
```

vizportal.password_reset

バージョン: バージョン 2024.2 で `features.PasswordReset` から置き換えられました。

既定値: `false`

ローカル認証を使用するサーバーにのみ適用されます。この値を `true` に設定し、ユーザーがサインインページの [パスワードを忘れましたか] オプションでパスワードをリセットできるようにします。

vizportal.rest_api.cors.allow_origin

`vizportal.rest_api.cors.enabled` が `true` に設定されている場合に、Tableau Server 上の REST API エンドポイントへのアクセスを許可するオリジン (サイト) を指定します。各エントリをコンマ (,) で区切ることで、複数のオリジンを指定できます。

```
tsm configuration set -k vizportal.rest_api.cors.allow_origin -v
https://mysite, https://yoursite
```

`vizportal.rest_api.cors.enabled` が `false` の場合、このオプションでリストされる起点は無視されます。詳細については、「Tableau Server の CORS を有効にする」を参照してください。

> **注:** すべてのサイトに一致するワイルドカードとしてアスタリスク文字 (*) を使用できます。サーバーに接続するすべての起点からのアクセスを許可し、セキュリティ上のリスクがあるため、アスタリスク文字の使用は推奨されません。影響とサイトへのリスクを十分に理解している場合を除き、アスタリスク文字 (*) は使用しないでください。

vizportal.rest_api.cors.enabled

既定値: `false`

Tableau Server がクロス オリジン リソース シェアリング (CORS) を許可するかどうかを制御します。に設定されている場合`true`、サーバーは Web ブラウザーに Tableau REST API エンドポイントへのアクセスを許可します。このオプションを使用して REST API でカスタム ポータルを作成することができます。既定では、この機能は有効になっていません。どの起点 (サイト) にアクセス権を付与するかを指定するには、`vizportal.rest_api.cors.allow_origin` オプションを使用します。このオプションで指定された起点のみ、Tableau Server REST API に対する要求を作成できます。詳細については、「Tableau Server の CORS を有効にする」を参照してください。

vizportal.site_user_group_count_enabled

バージョン: バージョン 2022.3.5 以降および 2023.1.0 以降で追加されました。

既定値: `false`

[サイト ユーザー] ページに各ユーザーのグループ数を示す列を含めるかどうかを制御します。

vizqlserver.allow_insecure_scripts

既定値: `false`

Tableau Desktop からサーバーへワークブックをパブリッシュし、ワークブックに安全でない可能性がある SQL または R 式 (たとえば、SQL インジェクションを許可する可能性がある SQL 式) が含まれている場合でも、サーバーからワークブックを開けるようにします。この設定が `false` になっている場合 (既定)、ワークブックをパブリッシュまたはサーバーから開くとエラー メッセージが表示され、ワークブックがブロックされます。この値を `true` に設定する前に、ナレッジ ベース記事の「Tableau Server での安全でないスクリプトのブロック/許可」を確認してください。

vizqldataservice.vmopts

バージョン: バージョン 2025.1 で追加。

既定値: `"-Xmx<default_value>m -XX:+CrashOnOutOfMemoryError -XX:-CreateCoredumpOnCrash"`

既定値は、システム メモリの量によって異なります。JVM の最大ヒープ サイズは、システム RAM 全体の 3.125% にスケールされます。

このオプションにより、Tableau Catalog の取り込みにおける JVM の最大ヒープ サイズが制御されます。既定値は自動的にスケールされます。どうしても必要な場合にのみ、このオプションを使用して、引数 -Xmx<default_value>g を変更して既定値を上書きしてください。たとえば、次のコマンドを使用して、最大ヒープ サイズを 6 GB に増やすことができます。

```
tsm configuration set -k vizqldataservice.vmopts -v "-Xmx6g -
XX:+CrashOnOutOfMemoryError -XX:-CreateCoredumpOnCrash"
```

既定値はシステム メモリの量に基づいており、システム RAM 全体の 3.125% です。

Index and Search Server のヒープ サイズを制御します。既定値は自動的にスケールされます。どうしても必要な場合にのみ、このオプションを使用して既定値を上書きしてください。キロバイトを示すには 'k'、メガバイトを示すには 'm'、ギガバイトを示すには 'g' を値に追加します。一般的なルールとして、ガベージ コレクションを最小化するには、初期ヒープ サイズ (`-Xms`) を最大ヒープ サイズ (`-Xmx`) と等しくなるように設定します。

vizqlserver.browser.render

既定値: `true`

`vizqlserver.browser.render_threshold` または `vizqlserver.browser.render_threshold_mobile` によって設定されたしきい値を下回るビューは、サーバーの代わりにクライアント Web ブラウザーによってレンダリングされます。詳細については、クライアント側レンダリングの構成を参照してください。

vizqlserver.browser.render_threshold

既定値: `100`

既定値は、PC に表示されるビューの高レベルの複雑度を表します。複雑度の要因には、マーク、ヘッダー、リファレンス ライン、および注釈の数が含まれます。このレベルの複雑度を超えるビューは、PC の Web ブラウザーの代わりにサーバーによってレンダリングされます。

vizqlserver.browser.render_threshold_mobile

既定値: 60

既定値は、タブレットに表示されるビューの高レベルの複雑度を表します。複雑度の要因には、マーク、ヘッダー、リファレンス ライン、および注釈の数が含まれます。このレベルの複雑度を超えるビューは、タブレットの Web ブラウザーの代わりにサーバーによってレンダリングされます。

vizqlserver.clear_session_on_unload

既定値: false

ユーザーが別のビューに移動したときやブラウザーを閉じたときに VizQL セッションをメモリ内に保存するかどうかを決定します。既定値 (false) ではセッションをメモリ内に保存します。ブラウザーの閲覧をやめるかブラウザーを閉じて VizQL セッションを終了するには、この設定を true にします。

vizqlserver.force_maps_to_offline

バージョン: バージョン 2020.4.0 で追加。

既定値: false

Tableau Server がマップ用にオフライン モードで実行されるかどうかを判定します。このキーは、インターネットとマップ サーバーへのアクセスが制限されている切断された環境で役立ちます。マップ用のオフライン モードを有効にするには、この値を true に設定します。インターネットへのアクセスがない環境で Tableau Server をインストールして構成する方法の詳細については、Tableau Server を未接続の (エア ギャップ) 環境にインストールするを参照してください。

vizqlserver.geosearch_cache_size

既定値: 5

サーバー メモリに同時に読み込むことができる、異なる地理的検索ロケール/言語データセットの最大数を設定します。メモリにないロケール/言語データセットの地理的検索要求をサーバーが受け取ると、メモリにセットが読み込まれます。データセットの読み込みが指定されている制限を超過する場合は、要求されたデータセットが読み込まれるように最近使用されたロケール/言語データセットがメモリからクリアされます。最小値は 1 です。各キャッシュは約 60 MB のメモリを消費します。したがって、これを 10 に設定すると、メモリ消費量は 600 MB (60 * 10) になります。

vizqlserver.initialsql.disabled

既定値: `false`

すべてのデータソースで初期 SQL ステートメントを無視するかどうかを指定します。これを以下のように true に設定し、初期 SQL を無視します。

```
tsm configuration set -k vizqlserver.initialsql.disabled -v true
```

vizqlserver.log.level

既定値: `info`

VizQL Server Java コンポーネントのロギング レベル。ログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizqlserver/*.log` に書き込まれます。

詳細については、`debug` に設定してください。デバッグ設定を使用するとパフォーマンスに大きな影響を与えるため、Tableau サポートで指示された場合にのみ使用します。

バージョン 2020.3.0 以降では、この設定は動的に構成できるため、この設定だけを変更する場合は、Tableau Server を再起動する必要はありません。詳細については、ロギング レベルの変更を参照してください。

vizqlserver.NumberOfWorkbookChangesBetweenAutoSaves

既定値: `5`

Web 作成の自動復元の構成。自動保存を起動するためにユーザーが行う必要のある変更の数を指定します。この値の変更は慎重に行ってください。自動復元機能は、Tableau Server での Web 作成や他の Viz に関連した操作のパフォーマンスに影響する場合があります。この値の調整は、時間をかけて増分調整することをお勧めします。

vizqlserver_<n>.port

"<n>" で指定した VizQL サーバーのインスタンスが実行されているポート。

vizqlserver.protect_sessions

バージョン: 2024.2.0 で廃止されました。2024.2.0 以降、Tableau Server は常に、元のユーザーがサインアウトした後に VizQL セッションが再度使用されるのを防止します。

既定値: `true`

`true` に設定すると、元のユーザーがサインアウトした後に VizQL セッションが再度使用されるのを防ぎます。

vizqlserver.querylimit

既定値: `1800`

ビューを更新するまでの最大許容時間 (秒)。1800 秒 = 30 分。この構成オプションは、VizQL Server とデータサーバーに影響を与えます。

vizqlserver.RecoveryAttemptLimitPerSession

既定値: `3`

Web 作成の自動復元の構成。同じセッションの復元の最大試行回数。この値の変更は慎重に行ってください。自動復元機能は、Tableau Server での Web 作成や他の Viz に関連した操作のパフォーマンスに影響する場合があります。この値の調整は、時間をかけて増分調整することをお勧めします。

vizqlserver.session.expiry.minimum

既定値: `5`

アイドル時間を分で表した値で、この時間を過ぎると、VizQLプロセスがメモリ不足になった場合に VizQL セッションを破棄できます。

vizqlserver.session.expiry.timeout

既定値: `30`

セッションが破棄されるまでのアイドル時間 (分)。

vizqlserver.sheet_image_api.max_age_floor

既定値: 1

REST API の Query View Image メソッドで生成される画像をキャッシュする時間 (分)。詳細については、REST API ヘルプの「REST API リファレンス」を参照してください。

vizqlserver.showdownload

既定値: true

ビューでの [ダウンロード] メニューの **[Tableau ワークブック]** オプションの表示を制御します。false に設定すると、[Tableau ワークブック] オプションは使用できなくなります。

> **注:** この設定では、Web 編集モードのユーザーのオプションは削除されません。

vizqlserver.showshare

既定値: true

ビューにある [共有] オプションの表示を制御します。このオプションを非表示にするには、false に設定します。

> **注:** ユーザーは、「showShareOptions」JavaScript または URL パラメーターを設定することにより、サーバーのデフォルト値を上書きできます。

vizqlserver.url_scheme_whitelist

ビューおよびダッシュボードの URL アクションを使用する場合に、許可リスト (承認リスト) に 1 つまたは複数の URL スキームを指定します。既定では、http、https、gopher、mailto、news、sms、tel、tsc、および tsl の各スキームが許可リスト (承認リスト) に登録されています。このコマンドは、次の例のように、複数のコンマとスペース区切り値を含めることができます。

```
tsm configuration set -k vizqlserver.url_scheme_whitelist -v
scheme1, scheme2
```

指定した値は以前の設定を上書きします。そのため、set コマンドをスキームの完全なリストを含める必要があります。(set コマンドを繰り返し実行して、スキームのリストを修正することはできません。)

vizqlserver.web_page_objects_enabled

既定値: true

ダッシュボードの Web ページ オブジェクトでターゲット URL を表示できるかどうかを制御します。Web ページが表示されないようにするには、false に設定します。

vizqlserver.WorkbookTooLargeToCheckpointSizeKiB

既定値: 5120

Web 作成の自動復元の構成。自動保存するワークブックのサイズ制限 (KB)。この値より大きなワークブックは自動保存されません。この値の変更は慎重に行ってください。自動復元機能は、Tableau Server での Web 作成や他の Viz に関連した操作のパフォーマンスに影響する場合があります。この値の調整は、時間をかけて増分調整することをお勧めします。

注: 以前のバージョンのサーバーで使用していたデフォルト値: 1024

vizqlserver.workflow_objects_enabled

既定値: true

Tableau 外部アクション ワークフローのオブジェクトをダッシュボードに追加できるかどうかを決定します。

webdataconnector.refresh.enabled

非推奨です。代わりに tsm data-access web-data-connectors allow を使用してください。

Tableau Server で Web データコネクタ (WDC) の抽出の更新を有効にするかどうかを判断します。すべての WDC で更新を無効にするには、このキーの値を以下のように false に設定します。

```
tsm configuration set --key webdataconnector.refresh.enabled --value
false
```

詳細については、「Tableau Server での Web データコネクタ」を参照してください。

webdataconnector.whitelist.fixed

非推奨です。代わりに `tsm data-access web-data-connectors add` を使用してください。

HTTP または HTTPS 経由でアクセス可能なデータ接続へのアクセスに使用できる 1 つまたは複数の Web データコネクタ(WDC) を指定します。このコマンドは、1 つの行に JSON データとして書式設定され、バックスラッシュ(\) を使用して二重引用符 (") がすべてエスケープされます。

例として、San Francisco Film Locations の WDC を許可リストに追加します。

```
tsm configuration set --key webdataconnector.whitelist.fixed --value
"'{\"https://tableau.data.world:443\": {\"properties\": {
\"secondary_whitelist\": [\"(https://data.world/)(.*)\"] } } }'"
```

詳細については、「Tableau Server での Web データコネクタ」を参照してください。

webdataconnector.enabled

非推奨です。代わりに `tsm data-access web-data-connectors allow` を使用してください。

既定値: `true`

`true` に設定すると、`tsm` コマンドを使用してサーバー上で Web データコネクタを管理できます。

webdataconnector.whitelist.mode

既定値: `mixed`

Tableau Server での Web データコネクタの実行方法を決定します。サポートされるモードは次の通りです。

- `mixed`。ユーザーは URL の許可リスト(承認リスト) にあるコネクタを実行できます。このモードでは、当初、インポートされた WDC を実行することもできました。WDC のインポートはサポートされなくなりました。
- `fixed`。ユーザーは URL の許可リスト(承認リスト) にあるコネクタを実行できます。
- `insecure`。ユーザーは任意のコネクターを実行できます。

重要: `insecure` オプションは、開発やテストでのみ使用してください。コネクターはカスタムコードを実行するため、吟味されていないコネクターを実行すると、セキュリティの脅威に晒される可能性があります。

wgserver.audit_history_expiration_days

既定値: `183`

履歴イベントのレコードが PostgreSQL データベース (Tableau Server のデータベース) から削除されるまでの日数を指定します。

wgserver.authentication.legacy_identity_mode.enabled

バージョン: 2022.1 で追加されました

既定値: Tableau Server 2022.1 以降では `false`2022.1 以降にアップグレードされた 2022.1 より前の Tableau Server の導入では、既定値は `true`.です。

アイデンティティプールを有効にするには、`false` に設定します。

詳細については、「ID プールのトラブルシューティング」を参照してください。

wgserver.authentication.identity_pools.default_pool_description

バージョン: バージョン 2023.1 で追加されました

既定値: Null

オプションとして、初期プール (TSM 設定) の説明を Tableau Server のランディング ページに追加し、すべてのユーザーに表示することができます。少なくとも 1 つの ID プールを作成すると、この説

明はプライマリサインイン オプションに追加され、初期プール (TSM 設定) に属するユーザーを正しいサインイン オプションに誘導するために使用できます。

たとえば、「正社員のサインインはこちら」 (Regular employees sign in here) という説明を追加するには、次のコマンドを使用します。

```
tsm configuration set -k wgserver.authentication.identity_
pools.default_pool_description -v "Regular employees sign in here"
```

注: 初期プール (TSM 設定) の説明は、サインインのカスタマイズの注記とは異なります。サインインのカスタマイズに関する注記は、Tableau Server ランディング ページですべてのサインイン オプションの下、および初期プール (TSM 設定) ユーザーがユーザー名とパスワードを入力するページに表示されます。

wgserver.authentication.prep_builder_externalbrowser

バージョン: バージョン 2025.3 で追加

既定値: `False`

Tableau Prep Builder から Tableau Server にサインインするときに優先する認証方法を設定します。`True`に設定すると、Tableau Server 認証ウィンドウが、組み込みブラウザーではなく、ユーザーの外部ブラウザーで開きます。

設定した場合、Tableau Prep Builder の [サーバー] メニューで **[サーバー設定を使用する]** が選択されたときにこのオプションが適用されます。この設定に関係なく、ユーザーは Tableau Prep Builder のサーバー メニューでサーバー認証オプションを選択することもできます。詳細については、「Tableau Server または Tableau Cloud へのフローのパブリッシュ」を参照してください。

Tableau Prep Builder ユーザー向けに、外部ブラウザーへの優先的なサーバー認証方法に設定するには、次のコマンドを使用します。

```
tsm configuration set -k wgserver.authentication.prep_builder_
externalbrowser -v true
```

注: このオプションは、ログインベースのライセンス管理 (LBLM) ではサポートされていません。

wgserver.change_owner.enabled

既定値: `true`

ワークブック、データソース、またはプロジェクトの所有権が変更可能かどうかを制御します。その他の選択肢には `false` および `adminonly` があります。

wgserver.clickjack_defense.enabled

既定値: `true`

に設定すると`true`、悪意のある人物によるTableau Server ユーザーへの "クリックジャッキング" Tableau Server ユーザー。クリックジャック攻撃では、ターゲットページが 2 ページ目の上に透明に表示され、攻撃者はユーザーに 2 ページ目で操作をしていると思わせておいて、ターゲットページをクリックさせたり、情報を入力させたりします。

詳細については、クリックジャック保護を参照してください。

wgserver.domain.accept_list

バージョン: この設定は、バージョン 2020.4.0 で追加されました。`wgserver.domain.whitelist` を置き換えるものです。

既定値: NULL

Tableau Server からセカンダリアクティブディレクトリドメインへの接続を許可します。セカンダリドメインはユーザーが同期するときに Tableau Server が接続するドメインですが、Tableau Server はインストールされていません。Tableau Server は、ユーザーとグループの同期のためにセカンダリドメインへの接続を試みます。場合によっては、Tableau Server がセカンダリドメインに接続できないことがあり、その結果、「Domain not in accept list (ドメインが許可リストにありません) (errorCode=101015)」というエラーが発生します。

`wgserver.domain.accept_list` オプションの設定は、セキュリティの脆弱性に対する修正プログラムで必要です。「[重要] ADV-2020-003: Tableau Server の強制認証」を参照してください。2020 年 2 月現在、この脆弱性に対する修正プログラムは、Tableau Server のすべての最新バージョンおよびメンテナンスリリースに含まれています。

このオプションを設定するには、二重引用符で囲まれたセカンダリドメインを入力します。複数のドメインは、コンマとスペースで区切る必要があります。たとえば、`tsm configuration set -k wgserver.domain.accept_list -v "example.org, domain.com"` と入力します。

ワイルドカード機能はサポートされていません。たとえば、Tableau が`sub1.example.org` と `sub2.example.org` とに接続する場合は、両方のドメインを追加する必要があります。

`wgserver.domain.accept_list` オプションを更新すると、既存の値が上書きされます。したがって、値に格納されている既存のドメインセットに新しいドメインを追加する場合は、オプションを設定する際に、新しいドメインを持つ既存のドメインをすべて含めます。`tsm configuration get -k wgserver.domain.accept_list` を実行すると、既存のドメインの一覧を取得できます。

wgserver.domain.ldap.domain_custom_ports

既定値: NULL

子ドメインとその LDAP ポートをマッピングできます。ドメインとポートはコロン (:) で区切られ、ドメインとポートのペア間は `FQDN1:port,FQDN2:port` の形式のようにコンマ (,) で区切られます。

例: `tsm configuration set -k wgserver.domain.ldap.domain_custom_ports -v childdomain1.lan:3269,childdomain2.lan:3269,childdomain3.lan:389`

wgserver.domain.password

既定値: Null

LDAP 接続に使用するユーザー アカウントのパスワードを指定します。外部のアイデンティティストアの構成 リファレンスを参照してください。

wgserver.domain.username

既定値: Null

LDAP 接続に使用するユーザー アカウントの名前を指定します。外部のアイデンティティストアの構成 リファレンスを参照してください。

wgserver.domain.whitelist

> **重要:** このキーはバージョン 2020.4.0 で非推奨になりました代わりに `wgserver.domain.accept_list` を使用してください。

既定値: NULL

Tableau Server からセカンダリアクティブディレクトリドメインへの接続を許可します。セカンダリドメインはユーザーが同期するときに Tableau Server が接続するドメインですが、Tableau Server はインストールされていません。Tableau Server は、ユーザーとグループの同期のためにセカンダリドメインへの接続を試みます。場合によっては、Tableau Server がセカンダリドメインに接続できない場合があり、その結果、「ドメインが許可リストにない (エラーコード=101015)」というエラーが発生することがあります。

wgserver.extended_trusted_ip_checking

既定値: `false`

信頼できるチケットの要求に対して IP クライアント照合を実施します。

wgserver.ignore_domain_in_username_for_matching

バージョン: バージョン 2021.4.21、2022.1.17、2022.3.9、2023.1.5 で追加

既定値: `false`

SAML を有効にすると、ID プロバイダー (IdP) のユーザー名と Tableau Server のユーザーアカウントを照合するときに、SAML ユーザー名属性のドメイン部分を無視するように Tableau Server を構成できます。ユーザー名属性のプレフィックス部分に一致するが、ユーザー名属性のドメイン部分には一致しないユーザーを Tableau Server ですでに定義していれば、ユーザー名属性のドメイン部分を無視できる場合があります。詳細については、SAML 要件のトピックのセクション「SAML ユーザー名属性を照合するときにドメインを無視する」を参照してください。

たとえば、SAML ユーザー名属性のドメイン名を無視するには、次のコマンドを実行します。

```
tsm configuration set -k wgserver.ignore_domain_in_username_for_
matching -v true
```

重要:

- ドメイン名を無視する際は、細心の注意を払ってください。特に、IdP で作成した構成済みドメイン全体で、ユーザー名が一意であることを確認してください。
- このコマンドは、`legacy-identity-mode` になっている Tableau Server 展開か、アイデンティティの移行を通じて更新されていない (アイデンティティサービスを使用しない) Tableau Server 展開のみで機能します。

wgserver.restrict_options_method

既定値: `true`

Tableau Server が HTTP OPTIONS 要求を受け入れるかどうかを制御します。このオプションが `true` に設定されている場合、サーバーは HTTP OPTIONS 要求に対して HTTP 405 (許可されていないメソッド) を返します。

wgserver.saml.blocklisted_digest_algorithms

バージョン: バージョン 2021.1 で追加。

既定値: `SHA1`

SAML 証明書の署名や、SAML アサーションのダイジェスト メソッドまたは署名 メソッドで、使用が認められないハッシュアルゴリズムを指定します。拒否リストに登録されたアルゴリズムで署名およびハッシュされた証明書やアサーションは、この設定を行うと拒否され失敗します。

Tableau 側と IdP 側の両方で SHA-1 が使用される可能性のある場面は複数あります。例:

- Tableau Server が IdP に送信される要求に署名するために使用する、TSM でアップロードされた証明書。
- 証明書の公開鍵を使用して IdP から受信した AuthnResponse (署名) の検証に使用する IdP メタデータの証明書。
- SHA-1 で署名およびハッシュされた着信アサーション (DigestMethod を SHA-1 に設定し、SignatureMethod を SHA-1 に設定)。

デフォルト値は Tableau Server 2021.2 で `SHA1` に変更されました。SAML を構成している 2021.2 へのアップグレードの詳細については、ナレッジベースの記事「Tableau Server 2021.2 へのアップグレード後、SAML 認証を使用した Tableau Server の起動に失敗するかログインが拒否される」を参照してください。

wgserver.saml.forceauthn

バージョン: バージョン 2019.3 で追加。

既定値: `false`

`true` に設定した場合、Tableau ユーザー セッションの有効期限が切れると、Tableau Server は IdP を使用してユーザーを再認証します。また、このオプションを使用して、ユーザーがアクティブな IdP セッションを持っている場合でも、ユーザーに再認証を求めるよう IdP に要求することができます。

wgserver.saml.idpattribute.username

SAML IdP がユーザー名を保存する属性の名前を指定します。既定では `username` に設定されています。IdP が使用する属性名にスペースが含まれている場合は、引用符で囲ってください。詳細については、サーバー全体の SAML の構成あるいはサイト固有の SAML の構成を参照してください。

wgserver.saml.iframed_idp.enabled

既定値: `false`

false の既定は、埋め込みビュー上でユーザーがサインイン ボタンを選択したときに、IdP のサインイン フォームがポップアップ ウィンドウで開くことを意味します。

true に設定し、既にサインインしているサーバー SAML ユーザーが埋め込みビューのある Web ページに移動すると、ユーザーはビューを見るためにサインインする必要がありません。

IdP が iframe 内のサインインをサポートする場合にのみ、これを true に設定できます。この iframe オプションはポップアップを使用するよりも安全性が低くなるため、すべての IdP でサポートされているわけではありません。IdP のサインイン ページでクリックジャック攻撃防止機能を使用している場

合 (ほとんどの場合で使用しています)、サインインページが iframe に表示されず、ユーザーはサインインできません。

お使いの IdP が iframe 経由でのサインインをサポートしていない場合、明示的に有効化する必要が生じることがあります。ただし、このオプションを使用できる場合でも、Tableau Server の SAML に対するクリックジャック攻撃防止機能が無効になるため、セキュリティリスクが生じるおそれがあります。

wgserver.saml.maxassertiontime

既定値: `3000`

作成から SAML アサーションが利用可能な最大秒数を指定します。

wgserver.saml.min_allowed.elliptic_curve_size

既定値: `256`

バージョン: バージョン 2021.1 で追加されましたが、デフォルト値は含まれていませんでした。デフォルト値は 2021.2 で `256` に設定されました。

このオプションは、SAML 認証に使用する証明書の ECDSA 曲線の最小許容サイズを指定します。ECDSA 曲線サイズが 256 未満の証明書をアップロードした場合、変更を適用すると TSM のログにエラーが記録されます。

SAML 証明書が 256 未満のサイズの ECDSA 曲線を使用している場合、Tableau Server 2021.2 以降にアップグレードしても、Tableau Server はアップグレード後に起動しません。アップグレードする前に、ECDSA 曲線のサイズが 256 またはそれ以上の新しい証明書をアップロードすることをお勧めします。代替策として、このコマンドを実行して、アップグレードする前に、Tableau Server の古いバージョン (2021.1 より前) でサイズがより短い ECDSA 曲線を設定することもできます。2021.1 より前のバージョンでこのコマンドを実行する場合は、`--force-keys` オプションをコマンドに含める必要があります。SAML を構成している 2021.2 へのアップグレードの詳細については、ナレッジベースの記事「Tableau Server 2021.2 へのアップグレード後、SAML 認証を使用した Tableau Server の起動に失敗するかログインが拒否される」を参照してください。

wgserver.saml.min_allowed.rsa_key_size

既定値: 2048

バージョン: バージョン 2021.1 で追加されましたが、デフォルト値は含まれていませんでした。デフォルト値は 2021.2 で 2048 に設定されました。

このオプションは、SAML 認証に使用する証明書の RSA キーの最小許容長を指定します。RSA キーの長さが 2048 未満の証明書をアップロードした場合、変更を適用すると TSM のログにエラーが記録されます。

RSA キーの長さが 1024 の SAML 認証を実行するには (非推奨)、この値を 1024 に設定します。

SAML 証明書が 2048 未満の長さのキーを使用している場合、Tableau Server 2021.2以降にアップグレードしても、Tableau Server はアップグレード後に起動しません。アップグレードする前に、キーの長さが 2048 またはそれ以上の新しい証明書をアップロードすることをお勧めします。代替策として、このコマンドを実行して、アップグレードする前に、Tableau Server の古いバージョン (2021.1より前) で強度がより弱いキーを設定することもできます。2021.1 より前のバージョンでこのコマンドを実行する場合は、--force-keys オプションをコマンドに含める必要があります。SAML を構成している 2021.2 へのアップグレードの詳細については、ナレッジベースの記事「Tableau Server 2021.2 へのアップグレード後、SAML 認証を使用した Tableau Server の起動に失敗するかログインが拒否される」を参照してください。

wgserver.saml.responseskew

既定値: 180

Tableau Server の時刻と、依然としてメッセージを処理できるアサーション作成 (IdP サーバーの時刻に基づく) 時間の最大秒数の差を設定します。

wgserver.saml.sha256

既定値: true

true に設定すると、Tableau Server は SAML アサーションで SHA-256 を使用し、メッセージの署名と要約を IdP にハッシュします。IdP が SHA-256 ハッシュコンテンツを含むアサーションを拒否した場合にのみ、このオプションを false に設定します。

wgserver.session.apply_lifetime_limit

既定値: false

サーバー セッションのセッション存続期間があるかどうかを制御します。サーバー セッションの存続期間を構成するには、これを true に設定します。

wgserver.session.idle_limit

既定値: 240

Web アプリケーションへのサインインがタイムアウトになるまでのアイドル時間 (分)。

wgserver.session.lifetime_limit

既定値: 1440

セッションの存続期間が設定されている場合は、サーバー セッションを持続する分数。既定値は 1440 分 (24 時間) です。wgserver.session.apply_lifetime_limit が false の場合 (デフォルト)、この設定は無視されます。

wgserver.unrestricted_ticket

既定値: false

信頼できるチケットによるユーザー認証に対し、サーバー リソースへアクセスを拡張するかどうかを指定します。既定の動作では、ユーザーにビューへのアクセスのみを許可します。これを true に設定すると、有効な信頼できるチケットを持つユーザーに対し、認証資格情報を使用してサインインしている場合のように、サーバー リソース (プロジェクト、ワークブックなど) へのアクセスを許可します。

workerX.gateway.port

既定値: 80 (SSL の場合は 443)

Apache が workerX ("worker" は、クラスタの後続のサーバー ノードで使用する用語) でリッスンする外部ポート。worker0.gateway.port は Tableau Server の外部ポートです。分散環境では、worker0 が Tableau Server の初期ノードです。

workerX.vizqlserver.procs

既定値: <number>

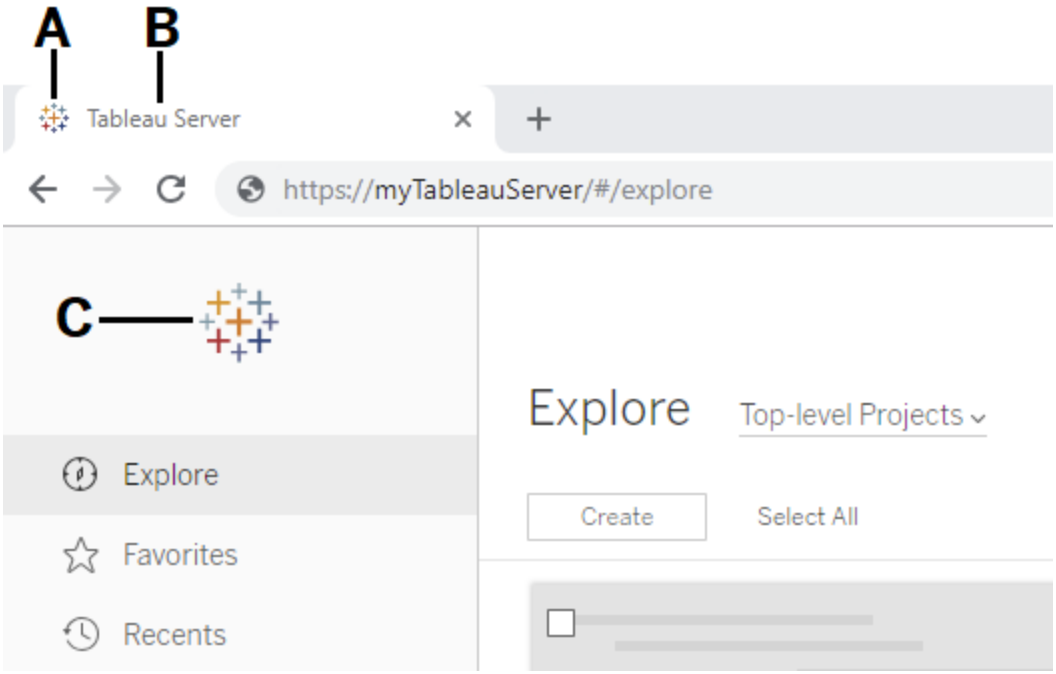
VizQL サーバー数。

zookeeper.config.snapCount

調整サービスでログのスナップショットが作成されるために必要なトランザクションの数を指定します。既定では、この値は 100,000 トランザクションです。調整サービスでスナップショットの作成に十分な数のトランザクションが書き込まれていない場合、5 日を超えたスナップショットの自動クリーンアップは実行されず、トランザクション ログによってディスク空き容量が失われる可能性があります。既定では、トランザクション ログとスナップショットは Tableau のデータディレクトリで作成されます。

tsm customize

`tsm customize` コマンドを使用すると Tableau Server クライアントを使用するブラウザー体験の外観をカスタマイズできます。



ロゴ	オプション フラグ	最小サイズ/最大サイズ (ピクセル単位)	推奨サイズ (ピクセル単位)
A - ウィンドウ タブのロゴ	変更できません	変更できません	変更できません
B - サーバー名	`--server-name`	適用されません	適用されません
C - ヘッダー ロゴ	`--header-logo`	32 x 32 (最小値)、160 x 48 (最大値)	48 x 48
表示なし - サインイン ロゴ	`--signin-logo`	3000 x 3000 (最大値)	
ヘッダー ロゴ/サインイン ロゴ	`--logo`	32 x 32 (最小値)、160 x 48 (最大値)	48 x 48
表示なし - ナビゲーション ペインが最小化されている場合に表示されるロゴ	`--compact-logo`	32 x 32 (最大値)	32 x 32

使用するイメージ ファイルは、GIF、JPEG、または PNG 形式でなければなりません。

ヘッダーとサインイン ページの背景色は同じではありません。両方の場所に同じイメージを使用する場合 (`--logo` オプションを使用する場合など)、サーバー インターフェイスでの表示位置によっては、ロゴが異なって見える場合があります。

災害復旧計画の一環として、カスタマイズした画像ファイルのバックアップをTableau Server 外部の安全な場所に保存することをお勧めします。Tableau Server に追加する画像ファイルは、クライアント ファイル サービスによって他のノードに格納および配布されます。ただし、ファイルは復元可能な形式では格納されません。Tableau Server クライアント ファイル サービスを参照してください。

シノプシス

```
tsm customize [options] [global options]
```

`customize` コマンドを実行後、次のコマンドを実行して変更を適用する必要があります。

```
tsm pending-changes apply
```

オプション

> **注:** パスとイメージ ファイル名にはスペースを含めることはできません。

`--compact-logo "<path-to-logo>"`

オプション。

ナビゲーション ペインのサイズが最小化されている場合は、表示されるイメージ ファイルへのパスを指定します。最大かつ最適なサイズは、32 × 32 ピクセルです。

`--header-logo "<path-to-logo>"`

オプション。

ヘッダーにのみ表示されるイメージ ファイルへのパスを指定します。

`--logo "<path-to-logo>"`

オプション。

ヘッダーとサインイン ウィンドウの両方に表示される単一のイメージ ファイルへのパス。

`--restore-defaults`

オプション。

すべてのカスタマイズ オプションを既定のインストールの状態にリセットします。

`--server-name <server_name>`

オプション。

ブラウザーのタブ、ツールチップ、メッセージに表示されるサーバー名。

`--signin-logo "<path-to-logo>"`

オプション。

サインイン ウィンドウにのみ表示されるイメージ ファイルへのパスを指定します。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm data-access

`tsm data-access` コマンドを使用すると、データ キャッシュの構成、データ リポジトリ アクセスの有効化または無効化、シングル サインオンで使用する SAML の有効化、および Web データ コネクタ (WDC) 設定の構成を行うことができます。

- caching
  - data-access caching list
  - data-access caching set
- repository
  - repository-access disable
  - repository-access enable
  - repository-access list
- set-saml-delegation
  - set-saml-delegation configure
  - set-saml-delegation disable
  - set-saml-delegation enable
- web-data-connectors
  - web-data-connectors add
  - web-data-connectors allow
  - web-data-connectors delete
  - web-data-connectors list

tsm data-access caching list

データ接続のキャッシュ設定を表示します。データ接続のキャッシュ設定の詳細については、データキャッシュの構成を参照してください。

シノプシス

```
tsm data-access caching list [global options]
```

tsm data-access caching set

データ接続のキャッシュ設定を行います。データ接続のキャッシュ設定の詳細については、データキャッシュの構成を参照してください。

シノプシス

```
tsm data-access caching set [options] [global options]
```

オプション

`-r, --refresh-frequency`

オプション。

参照元データソースへの新しいクエリを使用してキャッシュに保存されたデータを更新する頻度を設定します。データをキャッシュする最大分数を定義する数値を指定できます。また、データをできるだけ長い間キャッシュし再使用するには **low**、またはページを読み込むたびにデータを更新するには **always** (**0** に相当) を指定することもできます。このオプションを指定しない場合は、既定で **low** となります。

tsm data-access repository-access disable

既定のリモートユーザーによるTableau PostgreSQL データベースへの外部アクセスを無効にします。これによりlocalhost からのアクセスが無効になることはありません。

シノプシス

```
tsm data-access repository-access disable [options]
[global options]
```

オプション

`--repository-username <username>`

必須。

データリポジトリへのアクセス権を持つユーザー名で **tableau** または **readonly** のいずれか。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1500 (25 分) です。

`--ignore-prompt`

オプション。

再起動のメッセージを表示せずに Tableau Server を再起動します。

tsm data-access repository-access enable

Tableau PostgreSQL データベースへのアクセスを有効にします。

既定では、PostgreSQL トラフィックではポート 8060 (TCP) を使用します。ローカル ファイアウォールを実行している場合は、必ずこのポートでトラフィックを許可してください。PostgreSQL のポートを変更するには、動的マッピングされないポートを参照してください。

シノプシス

```
tsm data-access repository-access enable [options] [global options]
```

オプション

```
--repository-password <password>
```

必須。

指定したユーザー名のデータリポジトリにアクセスするパスワードを設定 (または変更) します。

```
--repository-username <username>
```

必須。

データリポジトリへのアクセス権を持つユーザー名で **tableau** または **readonly** のいずれか。

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1500 (25 分) です。

```
--ignore-prompt
```

オプション。

再起動のメッセージを表示せずに Tableau Server を再起動します。

tsm data-access repository-access list

Tableau PostgreSQL データベースへのアクセス権を持つユーザーを一覧表示します。

シノプシス

```
tsm data-access repository-access list [global options]
```

tsm data-access set-saml-delegation configure

SAML SAP HANA のシングル サインオンを設定します。これにより Tableau Server は、SAP HANA にデータ接続するシングル サインオンをユーザーに提供するアイデンティティプロバイダー (IdP) として機能します。

シノプシス

```
tsm data-access set-saml-delegation configure [options]
[global options]
```

オプション

```
-kf, --cert-key <cert-key>
```

オプション。

SAML 証明書キー ファイル。

```
-cf, --cert-file <file-path>
```

オプション。

SAML 証明書ファイルの場所。

```
-uf, --username-format <username-format>
```

オプション。

ユーザー名の形式。有効な形式キーは、'username'、'domain_and_username'、'email' です。

`-uc,--username-case <username-case>`

オプション。

ユーザー名の大文字/小文字の指定。有効な大文字/小文字の指定キーは、'lower'、'upper'、'preserve' です。

.

tsm data-access set-saml-delegation disable

SAML SAP HANA でのシングル サインオンを無効にします。

シノプシス

```
tsm data-access set-saml-delegation disable [global options]
```

tsm data-access set-saml-delegation enable

SAML SAP HANA でのシングル サインオンを有効にします。

シノプシス

```
tsm data-access set-saml-delegation enable [global options]
```

tsm data-access web-data-connectors add

Web データコネクタ(WDC) を WDC セーフリストに追加します。

シノプシス

```
tsm data-access web-data-connectors add [options] [global options]
```

オプション

`-n, --name <name>`

必須。

Tableau Server のデータソース リストに表示される WDC の名前。この名前は単一引用符 (') または二重引用符 (") で囲う必要があります。名前にスペースが含まれる場合は、二重引用符 (") を使用します。

`-sec, --secondary <secondary-URL-1>, <secondary-URL-2>`

WDC がセカンダリドメインを使用する場合に必要となります。

コネクタからリクエスト送信またはデータ受信が可能なドメインを示す URL のコンマ区切りリスト。例: 外部 JavaScript ライブラリ、REST API、またはローカルファイルなどURL を引用符で囲まないでください。この 2 番目の許可リストにドメイン全体を追加するには、次の例に示すように、URL の末尾にワイルドカードの式 `.*` を使用します。例: `https://www.example.com/.*`

Windows では、ワイルドカードの前後にかっこ (`.*`) を付けることができますが、必須ではありません。Linux では、かっこを付けるとエラーが発生します。`.*` をワイルドカードとして使用します。

WDC がセカンダリドメインを使用しているかどうか、またはセカンダリドメインが何であるかがわからない場合は、WDC の開発者に問い合わせる必要がある場合があります。また、`http://.*` や `https://.*` のようなワイルドカードURL を使用してすべてのドメインを許可することもできます。ただし、セキュリティを強化するために、より具体的な URL を使用することを強くお勧めします。

`--url <URL>`

必須。

WDC の URL (`<scheme>://<host>:<port>/<path>` として書式設定。例: `https://www.tableau.com:443/example/`)多くの WDC では、HTTP 接続に使用される既定のポート<port> 値 は 443 ですが、Tableau Server または Tableau Cloud

のデータソースの詳細からお使いのコネクタの値を確認することができます。WDC の URL の一部としてワイルドカード(.*) を使用することはできません。

tsm data-access web-data-connectors allow

WDC 更新を有効または無効にします。また、Tableau Server での WDC の使用を有効または無効にします。

シノプシス

```
tsm data-access web-data-connectors allow [options] [global options]
```

オプション

一方または両方のオプションを使用します。少なくとも --refreshes または --type のいずれかが必要です。

```
-r, --refreshes <refreshes-allowed>
```

--type が指定されている場合は省略可能です。

false に設定するとWDC の更新が却下され、true に設定するとWDC の更新が許可されます。

```
-t, --type <WDC-allowed>
```

--refreshes が指定されている場合は省略可能です。

none に設定すると、Tableau Server での WDC の使用が却下 (およびバックアップで WCD が省略) され、all に設定するとWDC の使用が許可されます。

tsm data-access web-data-connectors delete

特定の WDC、またはすべての WDC を Tableau Server のセーフリストから削除します。

シノプシス

```
tsm data-access web-data-connectors delete [options]
[global options]
```

オプション

`--all`

オプション。

このオプションを使用するとすべての WDC が削除されます。

`--url <URL>`

オプション。

削除する WDC の URL。

tsm data-access web-data-connectors list

セーフリストにあるすべての WDC をリストします。

シノプシス

```
tsm data-access web-data-connectors list [options] [global options]
```

オプション

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm email

`tsm email` コマンドを使用して、SMTP の構成を表示およびテストします。

SMTP の構成の詳細については、SMTP セットアップの構成を参照してください。

tsm email test-smtp-connection

STMP 接続をテストするには、このコマンドを実行します。実行すると、TSM は Tableau Server 用に構成した SMTP サーバーとの接続の確立を試行します。TSM は、接続状態と SMTP 構成の

詳細も返します。

場合によっては、コマンドは誤検知ステータスを返します。たとえば、Postfix SMTP サーバーが TLS を要求するように設定されていても、Tableau Server が TLS 用に構成されていない場合、接続が確立され、TSM は正常な接続を報告します。ただし、このシナリオでは、TSM が接続された後、Postfix は実際に電子メール メッセージを拒否します。

シノプシス

```
tsm email test-smtp-connection [global options]
```

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm initialize

`tsm initialize` コマンドを使用して Tableau Server を初期化できます。

注: `tsm initialize` を実行する前に保留中の変更は適用または破棄する必要があります。そうでないと、初期化は失敗します。保留中の変更を適用するには、`tsm pending-changes apply` コマンドを使用します。適用しない保留中の変更を破棄するには、`tsm pending-changes discard` を使用します。

シノプシス

```
tsm initialize [options] [global options]
```

オプション

`-r,--start-server`

オプション。初期化の完了後にサーバーを実行したままにします。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm jobs

tsm jobs コマンドを使用して、ジョブのリスト、再接続、およびキャンセルを実行できます。

- cancel
- list
- reconnect

tsm jobs cancel

サーバー上でジョブをキャンセルします。すべてのジョブは、実行を開始する前にキャンセルできます (キューに入れられている場合)。ジョブを既に実行中の場合は、特定のジョブ (クリーンアップ、ファイル ストアの使用停止、バックアップの生成、サーバーの再起動、サーバーの起動) のみをキャンセルできます。ジョブのキャンセルの詳細については、「TSM ジョブのキャンセル」を参照してください。

シノプシス

```
tsm jobs cancel --id <jobID> [global options]
```

オプション

```
-i,--id <jobID>
```

必須。

キャンセルするジョブの ID。

tsm jobs list

サーバー上の非同期ジョブをリストします。

シノプシス

```
tsm jobs list [--status <status>] [global options]
```

オプション

```
-t,--status <status>
```

オプション。

指定されたステータスと一致するジョブをフィルターします。

tsm jobs reconnect

非同期ジョブに再接続して進行状況を表示します。ジョブ ID が指定されていない場合は最新のジョブに再接続されます。

シノプシス

```
tsm jobs reconnect[--id <jobID>] [global options]
```

オプション

```
-i,--id <jobID>
```

オプション。

再接続する必要があるジョブの ID を指定します。

グローバル オプション

```
-h, --help
```

オプション。

コマンドのヘルプを表示します。

```
-p, --password <password>
```

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm licenses

tsm licenses コマンドを使用して、サーバー ライセンス タスクを管理します (例:Tableau Server プロダクト キーのオンラインまたはオフラインでのライセンス認証/ライセンス認証解除、オフラインでのライセンス認証/ライセンス認証解除でのファイルの関連付け)。

- tsm licenses activate
- tsm licenses atr-configuration get
- tsm licenses atr-configuration set
- tsm licenses deactivate
- tsm licenses get-offline-activation-file
- tsm licenses get-offline-deactivation-file
- tsm licenses list
- tsm licenses refresh

tsm licenses activate

Tableau Server プロダクト キーをライセンス認証します。

シノプシス

```
tsm licenses activate --license-key <product-key> [global options]
```

オプション

`-f, --license-file <file.tlf>`

オフラインでのライセンス認証を実行する場合は必須。

オフラインでのライセンス認証に使用するライセンス ファイル (<file>.tlf) を指定します。

`-k, --license-key <product-key>`

有効なプロダクト キーをライセンス認証する場合は必須。

オンラインでのライセンス認証に使用するプロダクト キーを指定します。

`-t, --trial`

注： このオプションは、2023.3.0 よりも前のバージョンでのみ使用できます。新しいバージョンでの試用ライセンスについては、アカウント担当者にお問い合わせください。

試用版をライセンス認証する場合は必須。

試用版をライセンス認証します。

tsm licenses atr-configuration get

サーバーの実行承認 (ATR) の期間を表示します。ATR 期間は、ライセンスの更新前に Tableau Server での実行が承認されている期間です。

シノプシス

```
tsm licenses atr-configuration get --duration [global options]
```

オプション

--期間

現在の実行承認 (ATR) を秒単位で表示します。たとえば、432,000 秒 (5 日間) です。

tsm licenses atr-configuration set

サーバー実行承認 (ATR) の期間を設定します。ATR 期間は、ライセンスの更新前に Tableau Server での実行が承認されている期間です。

シノプシス

```
tsm licenses atr-configuration set --duration <duration_in_seconds>
[global options]
```

オプション

--期間 <duration_in_seconds>

実行承認 (ATR) の期間 (秒) を設定します。たとえば、432,000 秒 (5 日間) です。

tsm licenses deactivate

Tableau Server プロダクトキーをオンラインまたはオフラインでライセンス認証します。

シノプシス

```
tsm licenses deactivate --license-key <product-key>
[global options]
```

オプション

`-f, --license-file <return_file.tlr>`

オフラインでのライセンス認証の解除を実行する場合は必須。

オフラインでのライセンス認証の解除に使用するライセンス ファイル (<file>.tlf) を指定します。

`-k, --license-key <product-key>`

プロダクト キーのライセンス認証を解除する場合は必須。

オンラインでのライセンス認証の解除に使用するプロダクト キーを指定します。

tsm licenses get-offline-activation-file

Tableau Server のオフラインでのライセンス認証に使用するオフラインのライセンス認証ファイルを生成します。詳細については、オフラインでの Tableau Server の有効化を参照してください。

注: サーバー ATR を使用していない場合は、一度にアクティブ化できるプロダクト キーは 1 つだけです。サーバー ATR を使用すると、オフライン アクティブ化ファイルにプロダクト キーのコンマ区切りリストを指定して、複数のプロダクト キーを同時にアクティブ化できます。

シノプシス

```
tsm licenses get-offline-activation-file --license-key <product-
key> --output-dir <path> [global options]
```

オプション

`-k, --license-key <product-key>`

必須。

オフラインでのライセンス認証に使用するプロダクトキーを指定します。

`-o, --output-dir <path>`

必須。

オフラインのライセンス認証ファイルを保存する場所。この場所を設定する必要があります。

tsm licenses get-offline-deactivation-file

Tableau Server のオフラインでのライセンス認証解除に使用するオフラインのライセンス認証解除ファイルを生成します。詳細については、オフラインでの Tableau Server のライセンス認証解除を参照してください。

シノプシス

```
tsm licenses get-offline-deactivation-file --license-key <product-key> --output-dir <path> [global options]
```

オプション

`-k, --license-key <product-key>`

必須。

オフラインでのライセンス認証の解除に使用するプロダクトキーを指定します。

`-o, --output-dir <path>`

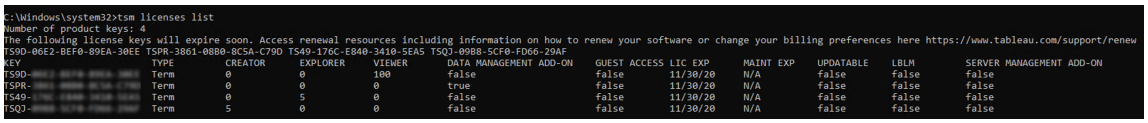
必須。

オフラインのライセンス認証解除ファイルを保存する既存の場所。

tsm licenses list

Tableau Server の展開でライセンス認証されたライセンスを一覧表示します。

たとえば、5 つの Creator ライセンス、5 つの Explorer ライセンス、100 個の Viewer ライセンス、および データ管理 を持つサーバーは、次のようなコマンド出力を生成します。



次のフィールドが返されます。

- **KEY**: ライセンスを識別するグローバルに一意の 16 文字の文字列。
- **TYPE**: ライセンスの種類についての説明。
  - 期間: 期間 ライセンスはサブスクリプション スケジュールにマップされ、更新する必要があります。有効期限は、[LIC EXP] フィールドに一覧表示されます。
  - 永続: 永続 ライセンスは一度購入したら更新する必要はありませんが、MAINT EXP (メンテナンスの有効期限) は更新する必要があります。
  - コア: コア ライセンスは、特定の Tableau Server サービスを実行しているコンピューター上のコアの数にマップされるライセンスです。コア ライセンスを使用すると、ゲスト ユーザーがサーバー上のビューにアクセスしたり、他の Web サーバーに埋め込んだりできます。また、コア ライセンスでは、Explorer ユーザーと Viewer ユーザーを無制限に許可できます。
- **CREATOR**: Tableau Server の展開で発行された Creator ライセンスの数。
- **EXPLORER**: Tableau Server の展開で発行された Explorer ライセンスの数。
- **VIEWER**: Tableau Server の展開で発行された Viewer ライセンスの数。
- **DATA MANAGEMENT**: Tableau Server が データ管理 (`True`/`False`) でライセンスされています。データ管理 についてを参照してください。
- **GUEST ACCESS**: Tableau Server がゲスト ユーザーに対してライセンスされています。ゲスト ユーザーを参照してください。ゲスト ユーザーを利用するには、コア ライセンスが必要です。[TYPE] フィールドを参照してください。
- **LIC EXP**: ライセンスの有効期限が切れ、Tableau Server が動作を停止する日付。期間 ライセンスの有効期限が切れます。[TYPE] フィールドを参照してください。ライセンスを更新するには、Tableau のカスタマー ポータルにアクセスしてください。
- **MAINT EXP**: レガシー永続 ライセンス (TYPE = 永続) にのみ適用されます。期間 ライセンスの場合、このフィールドは `N/A` と出力されます。[MAINT EXP] には、Tableau Server 展開に対するメンテナンス契約の有効期限が切れる日付が表示されます。ライセンスのメン

テナンス キーを更新するには、プロダクト キーの有効期限と属性の更新を参照してください。Tableau のカスタマー ポータルにアクセスしてメンテナンスの購入履歴を表示し、追加のメンテナンスを購入してください。

- **UPDATABLE**:更新可能なサブスクリプション ライセンスかどうかを指定します (`True`/`False`)。
- **LBLM**: Tableau Server の展開に対してログインベースのライセンス管理 (LBLM) が有効かどうかを指定します (`True`/`False`)。有効にすると、ユーザーはプロダクト キーを入力するのではなく、Tableau Server にログインすることで、Tableau Desktop または Prep のインスタンスのライセンスを取得できます。LBLM の詳細については、ログインベースのライセンス管理を参照してください。
- **SERVER MANAGEMENT**: Tableau Server が Advanced Management (旧称 Server Management Add-on) (`TrueFalse`) でライセンスされています。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

シノプシス

```
tsm licenses list [global options]
```

tsm licenses refresh

Tableau Server ですべてのプロダクト キーのメンテナンス有効期限を更新します。

シノプシス

```
tsm licenses refresh [global options]
```

グローバル オプション

```
-h, --help
```

オプション。

コマンドのヘルプを表示します。

```
-p, --password <password>
```

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm login

リモート ノードから Tableau サービス マネージャーにログインするには、tsm login コマンドを使用します。

TSM 承認グループのメンバーとしてログインしているアカウントの場合、tsm CLI をローカルで実行する際にコマンドを実行するために認証資格情報を指定する必要はありません。詳細については、tsm CLI での認証を参照してください。

シノプシス

```
tsm login [global options]
```

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm logout

`tsm logout` コマンドを使用して、Tableau サービス マネージャー (TSM) をログアウトできます。

シノプシス

```
tsm logout [global options]
```

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm maintenance

`tsm maintenance` コマンドを使用して、バックアップの作成や以前に作成したバックアップからの Tableau Server の復元など、サーバー メンテナンス タスクを管理できます。

- tsm maintenance backup
- tsm maintenance cleanup

- tsm maintenance jmx
  - tsm maintenance jmx disable
  - tsm maintenance jmx enable
- tsm maintenance metadata-services
  - tsm maintenance metadata-services disable
  - tsm maintenance metadata-services enable
  - tsm maintenance metadata-services get-status
- tsm maintenance reindex-search
- tsm maintenance reset-searchserver
- tsm maintenance restore
- tsm maintenance send-logs
- tsm メンテナンス スナップショット バックアップ (外部 ファイル ストア)
  - tsm maintenance snapshot-backup complete
  - tsm maintenance snapshot-backup prepare
  - tsm maintenance snapshot-backup restore
- tsm maintenance validate-backup-basefilepath
- tsm maintenance validate-resources
- tsm maintenance ziplogs

tsm maintenance backup

Tableau Server によって管理されるデータのバックアップを作成します。このデータには、Tableau の PostgreSQL データベース (リポジトリ) が含まれます。これには、ワークブック、ユーザーのメタデータ、および抽出ファイル (バージョン 2024.2 以前の .hyper ファイルおよび .tde ファイル) が含まれます。このデータには、構成データは含まれません。「Tableau Server の完全なバックアップと復元を実行します」を参照してください。

注: 外部 ファイル ストアを使用している Tableau Server インストールでは、このコマンドを使用しないでください。「外部 ファイル ストアによるバックアップと復元」を参照してください。

トポロジの構成を使用した最適化:

- 管理コントローラーと同じノード上にファイル ストアを配置すると、バックアップ プロセス中にノード間でデータを転送する必要性が減少するか排除されるため、Tableau Server のバックアップにかかる時間を短縮できます。これは、組織で多数の抽出を使用している場合に特に当てはまります。
- リポジトリ(pgsql) を管理コントローラー ノードと同じ場所に配置すると、バックアップ時間は短縮できますが、ファイル ストアの場合ほど時間の節約は顕著ではありません。

最初のノードでエラーがあった場合や、コントローラーを別のノードに移動した場合を除き、管理コントローラーは通常最初のノード上にあります。

バックアップ ファイルは、データディレクトリの一時的な場所に集められ、TSM `basefilepath.backuprestore` 変数に定義したディレクトリに書き込まれます

`/var/opt/tableau/tableau_server/data/tabsvc/files/backups/<filename>.tsbak`

バックアップ ファイルの書き込み先とその場所の変更方法の詳細については、tsm ファイル パスを参照してください。**注:** バックアップの場所を変更する場合でも、バックアップ プロセスではデータディレクトリの一時的な場所を使用してバックアップ ファイルを集めます。

シノプシス

```
tsm maintenance backup --file <backup_file> [options]
[global options]
```

オプション

`-f, --file <backup_file>`

必須。

リポジトリデータのバックアップの詳細については、「Tableau Server データのバックアップ」を参照してください。

`-d, --append-date`

オプション。

バックアップ ファイル名の末尾に現在の日付を追加します。

注: スケジュールを使用して作成されたバックアップには、バックアップファイル名の先頭に日付/時刻が自動的に含まれます。-d オプションも使用する場合、ファイル名には日付が 2 回含まれます。バックアップのスケジュール設定についての詳細は、「バックアップのスケジュール設定と管理」を参照してください。

`-i, --description <string>`

オプション。

指定したバックアップ ファイルの説明が含まれます。

`--ignore-prompt`

オプション。バージョン 2020.2 での追加

ファイル ストアが管理コントローラーと同じノード(通常は初期ノード) に存在しない場合でも、プロンプトを表示せずにバックアップします。このプロンプトは、バックアップを自動化する(たとえば、スクリプトを使用して) 場合に使用します。

`-k, --skip-verification`

オプション。

データベースのバックアップの整合性を検証しないでください。

`--multithreaded`

任意の項目です。バージョン 2021.1 での追加

Tableau Server のバックアップを作成する場合、既定では単一のスレッドが使用されます。このオプションを指定すると、複数のスレッドを使用したバックアップが作成されます。このオプションを指定すると、既定で 2 つのスレッドが使用されます。使用するスレッドの数は `backup.zstd.thread_count` 構成キーを設定して変更できます。

`--override-disk-space-check`

オプション。

ディスク容量の低下が警告されている場合でもバックアップの作成を試行します。

`-po, --pg-only`

オプション。

リポジトリバックアップのみを生成します。

> **重要:** Tableau サポートからの指示がない限り、バックアップを生成するときに `pg-only` オプションを使用しないでください。このオプションはリポジトリのみをバックアップし、Server の復元には使用できません。主な用途はトラブルシューティングであり、必要があれば Tableau サポートから `--pg-only` バックアップを作成するように依頼されます。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまでの待機秒数既定値は 86400 (1440 分) です。

`-sd, --schedule-days <day[,day]>`

オプション。バージョン 2020.4 での追加

スケジュールを実行する日数週単位のスケジュールには 1 ~ 7 (月曜日は 1、日曜日は 7)、月単位のスケジュールには 1 ~ 31 を使用します (月に指定した日が含まれていない場合は、その月の最終日を使用します)。複数の値はコンマで区切ります。

`-si, --schedule-id <ID>`

オプション。バージョン 2020.4 での追加

更新する既存のスケジュールの ID を指定します。

`--skip-compression`

オプション。

圧縮を使用せずにバックアップを作成します。バックアップ ファイルは大きくなりますが、バックアップの完了にかかる合計時間が短くできます。これをマルチノード インストールで使用している場合は、ファイル ストア インスタンスを最初のノードで構成することを強くお勧めします。

`-sn, --schedule-name <name>`

オプション。バージョン 2020.4 での追加

作成または更新するスケジュールの名前を指定します。

`-sr, --schedule-recurrence <frequency>`

オプション。バージョン 2020.4 での追加

スケジュールを繰り返す頻度有効なオプションは、"毎日"、"毎週"、"毎月".です。

`-st, --schedule-time <HH:MM>`

オプション。バージョン 2020.4 での追加

スケジュールを実行する必要がある時刻 (24 時間形式: HH:MM)。

例

この例では、次のように `/var/opt/tableau/tableau_server/data/tabsvc/files/backups/` ディレクトリに `ts_backup-<yyyy-mm-dd>.tsbak` というバックアップを作成します。

```
tsm maintenance backup -f ts_backup -d
```

この例では、毎週土曜日の正午に実行する "weekly-saturday-backup" という名前の毎週定期的に行うバックアップ スケジュールを作成し、`<yyyy.mm.dd.hh.mm>-ts_saturday_backup.tsbak:` という名前のバックアップを作成します。

```
tsm maintenance backup -f ts_saturday_backup -sr weekly -st 12:00 -sd 6 -sn weekly-saturday-backup
```

スケジュールされたバックアップの管理の詳細については、バックアップのスケジュール設定と管理を参照してください。

tsm maintenance cleanup

既定では、`tsm maintenance cleanup` コマンドは、1 日以上経過した一時ファイルとログファイルを削除します。コマンドオプションを使用して、保持期間や削除するファイルを変更できます。
注: このコマンドでは crashdump ファイルは削除されません。

このコマンドの影響は、Tableau Server が実行中かどうかよって異なります。

- サーバーが実行されている場合、ほとんどの古いファイルと`http_requests` テーブルのエントリは削除できますが、Tableau Server プロセスの一時ディレクトリの下にあるファイルは、プロセスが実行されている場合には削除されません。すべての一時ファイルを完全に削除するには、このコマンドの実行前にサーバーを停止する必要があります。`http_requests` テーブル エントリを削除するには、`-q` オプションを使用します。
- サーバーが停止している場合、`http_requests` エントリは削除できません。

Tableau Server を分散型の展開で実行している場合は、このコマンドを管理コントローラ(*TSM コントローラ*とも呼ばれる)プロセスを実行しているノード上で実行してください。既定では、コントローラーはほとんどの場合、クラスターの初期ノード上にあります。

> **注:** このコマンドは、Tableau Server バージョン 10.5.1 で追加されました。また、一部のオプションはバージョン 2018.1 で追加されました。

シノプシス

```
tsm maintenance cleanup [options] [global options]
```

オプション

`-a, --all`

オプション。

既定の保持値を使用してすべてのクリーンアップ操作を実行します。 `-l -t -r -q -ic` オプションを指定して `cleanup` コマンドを実行する操作と同じです。

`--http-requests-table-retention <# of days>`

オプション。

既定: 7 日

保持するテーブル エントリの `http_requests` 日数を指定します。このオプションを `-q` と一緒に使用すると、保持するテーブル エントリの日数を指定して、既定の 7 日間を上書きすることができます。このオプションはテーブル エントリの保持期間を指定しますが、テーブル エントリの実際の削除をトリガーしません。これを `-q` オプションと一緒に使用すると、エントリの削除がトリガーされます。

`-ic, --sheet-image-cache`

オプション。バージョン 2019.4 での追加

イメージのキャッシュをクリアします。このキャッシュには、オフライン プレビューのイメージ、サブスクリプション メール メッセージのスナップショット、サブスクリプション PDF、およびパブリッシュされた REST API エンドポイントで要求されたすべてのイメージが含まれている可能性があります。(詳細については、「rest_api_ref.htm」を参照してください。)

`-l, --log-files`

オプション。

1 日以上前のログ ファイルを削除します。`data/tabsvc/logs` の下にあるサブディレクトリのファイルは削除されます。

`--log-files-retention <# of days>`

オプション。

既定: 1 (24 時間)

この日数より古いログを削除します。このオプションを使用して、既定の保持期間 (1 日) を上書きすることができます。このコマンドは、一時ファイルには適用されません。

`-q, --http-requests-table`

オプション。

古い `http_requests` テーブル エントリーを削除します。テーブル エントリを削除するには、Tableau Server が実行されている必要があります。Tableau Server が停止している場合、このオプションは無視されます。このオプションのみを使用して、既定の保持期間 (7 日) より古いエントリの削除を指定したり、`--http-requests-table-retention` と一緒に使用して既定以外の保持期間を指定したりできます。

注: `http_requests` テーブル エントリを削除すると、カスタム管理ビューで使用できるデータが完全に削除されます。このデータを削除しても、必要なカスタム ビューに影響しないことを確認してください。

`-r, --redis-cache`

オプション。

Redis キャッシュをクリアします。

`--request-timeout <timeout in seconds>`

オプション。

既定: 3600

コマンドが終了するまで指定した時間待機します。

`-t, --temp-files`

オプション。

以下のディレクトリにあるすべてのファイルとサブディレクトリを削除します。

- `/var/opt/tableau/tableau_server/data/tabsvc/temp`: 各 Tableau Server プロセスディレクトリの下のファイルを削除するには、コマンドを実行する前に Tableau Server を停止する必要があります。
- `/var/opt/tableau/tableau_server/data/tabsvc/httpd/temp`
- `/var/opt/tableau/tableau_server/temp`

例

この例では、2 日を超えて経過したすべてのログ ファイルがクリーン アップされます。

```
tsm maintenance cleanup -l --log-files-retention 2
```

tsm maintenance jmx disable

バージョン: バージョン 2022.1 で追加されたコマンド

Tableau Server で JMX を無効にするには、`tsm maintenance jmx disable` コマンドを使用します。

このコマンドを実行するには、TSM サービスを含む Tableau Server を再起動する必要があります。

シノプシス

```
tsm maintenance jmx disable [options] [global options]
```

オプション

`--ignore-prompt`

オプション。

プロンプトを表示せずに JMX を無効にします。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance jmx enable

バージョン: バージョン 2022.1 で追加されたコマンド

Tableau Server で JMX を有効にするには、`tsm maintenance jmx enable` コマンドを使用します。

このコマンドはインタラクティブであり、コマンド ライン パラメーターとしてオプションを指定しない場合は、適用可能なオプションの入力を求めるプロンプトが表示されます。

このコマンドを実行するには、TSM サービスを含む Tableau Server を再起動する必要があります。

シノプシス

```
tsm maintenance jmx enable
```

オプション

`--access <readonly | readwrite>`

オプション。

読み取り専用または読み取り/書き込みアクセスのいずれかで JMX を有効にします。既定は `readonly` です。

`--ignore-prompt`

オプション。

追加のセキュリティオプションのプロンプトを表示せずに JMX を有効にします。これにより、他のコマンド オプションを含めない場合は、SSL を使用した JMX が有効になり、Tableau Server を再起動します。例:

```
tsm maintenance jmx enable --ignore-prompt
```

`--no-ssl`

オプション。

SSL なしで JMX を有効にします。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

`--ssl-client-auth`

オプション。

SSL のクライアント認証を要求する JMX を有効にします。

`--unsecured`

オプション。

セキュリティ機能を有効にせずに JMX を有効にします (これは推奨されていません)。これは、2022.1 より前のバージョンで `service.jmx_enabled` 構成キーを `true` に設定することと同じです。

tsm maintenance metadata-services disable

バージョン: バージョン 2019.3 で追加されたコマンド

`tsm maintenance metadata-services disable` コマンドを使用して、Tableau メタデータ API を無効にします。

メタデータ API を無効にすると、Tableau Server でコンテンツに関する情報の継続的な取り込みとインデックス化が停止し、Tableau Server およびそのコンテンツに関連付けられたアセットにパブリッシュされたコンテンツに関する情報のインデックスが削除され、メタデータ API に対するクエリと Tableau Catalog へのアクセスを行う機能の両方が無効になります。

このコマンドの実行を停止し、Tableau Server で使用するいくつかのサービスを起動します。これによって、推奨事項など一部の機能が一時的に使用できなくなります。

シノプシス

```
tsm maintenance metadata-services disable
```

オプション

```
--ignore-prompt
```

オプション。

メタデータAPI を無効にするときに行う確認の要求を却下します。

tsm maintenance metadata-services enable

バージョン: バージョン 2019.3 で追加されたコマンド

`tsm maintenance metadata-services enable` コマンドを使用して、Tableau Server の Tableau メタデータAPI を有効にします。

Tableau Server がデータ管理でライセンスされている場合は、メタデータAPI を有効にすると Tableau Catalog が有効になります。

メタデータAPI を有効にすると、Tableau Server のコンテンツに関する情報が取り込まれ、メタデータAPI ストアにインデックス化されます。メタデータAPI を使用して、Tableau Server にパブリッシュされたコンテンツに関するスキーマ、系列、ユーザーが管理するメタデータをクエリできます。メタデータAPI を有効にした後は、メタデータAPI が無効になるまでメタデータは継続的に取り込まれ、インデックス化されます。

このコマンドを実行するときは、次の点に留意してください。

- このコマンドでは、Tableau Server で使用されているいくつかのサービスが停止してから起動します。これによって、ユーザーは推奨事項などの一部の機能を一時的に使用できなくなります。
- このコマンドを使用するたびに、メタデータの新しいインデックスが作成されて前のインデックスが置き換えられます。

Tableau Catalog の詳細については、「Tableau Catalog について」を参照してください。

シノプシス

```
tsm maintenance metadata-services enable
```

オプション

--ignore-prompt

オプション。

メタデータAPI を有効にするときに行う確認の要求を却下します。

tsm maintenance metadata-services get-status

バージョン: バージョン 2019.3 で追加されたコマンド

`tsm maintenance metadata-services get-status` コマンドを使用して、メタデータサービスのステータス情報を取得します。

メタデータサービスのステータスは、メタデータAPI ストアが初期化されたかどうか、Tableau メタデータAPI が実行されているかどうかを示します。

シノプシス

```
tsm maintenance metadata-services get-status
```

tsm maintenance reindex-search

`tsm maintenance reindex-search` コマンドを使用して検索インデックスを再構築します。

シノプシス

```
tsm maintenance reindex-search [options] [global options]
```

オプション

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance reset-searchserver

バージョン: このコマンドは 2023.3.0 で廃止 (削除) され、検索と参照 (検索サーバーとも呼ばれます) は廃止されました。検索と参照は、インデックスと検索サーバーに置き換えられました。

注: バージョン 2023.3.0 以降でこのコマンドを実行しても何も起こりません。

検索サーバーをクリーンな状態にリセットし、検索情報を削除して検索インデックスを再構築します。

シノプシス

```
tsm maintenance reset-searchserver [options] [global options]
```

オプション

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance restore

指定したバックアップファイルを使用して、Tableau Server を復元します。バックアップファイルを復元しても、構成データは復元されません。「Tableau Server の完全なバックアップと復元を実行します」を参照してください。

稼働中のサーバーと同じタイプのアイデンティティストアを持つバックアップからのみ復元可能です。たとえば、ローカル認証を使用しているサーバーからのバックアップは、ローカル認証で初期化された

Tableau Server に復元可能ですが、Active Directory 認証を使用しているサーバーからのバックアップは、ローカル認証で初期化されたサーバーには復元できません。

バージョン 2022.3 以降、tabadmin を使用して作成したバックアップ (「以前の TSM バックアップ」) はサポートされていません。TSM 以前のバックアップを Tableau Server バージョン 2022.3 以降に復元することはできません。

シノプシス

```
tsm maintenance restore --file <file_name> [--restart-server]
[global options]
```

オプション

`-f, --file <file_name>`

必須。

復元するバックアップ ファイルを指定します。

`restore` コマンドは、TSM の `basefilepath.backuprestore` 変数で定義したディレクトリにバックアップ ファイルがあることを前提としています。既定のディレクトリは次のとおりです。

`/var/opt/tableau/tableau_server/data/tabsvc/files/backups/`

ファイル パスとその変更方法の詳細については、tsm ファイル パスを参照してください。

`-ak, --asset-key-file <file_name>`

オプション。バージョン 2021.4.0 で非推奨になりました。

このオプションは、Tableau Server (バージョン 2018.1 以前) で tabadmin により作成されたアセットから復元する場合にのみ指定します。

復元元のアセット キー ファイルの名前。アセット キー ファイルは、`tabadmin assetkeys` コマンドによって作成されます。ファイルがサーバー上の事前に定義されたバックアップ/復元場所に置かれている必要があります。

`-k, --skip-identity-store-verification`

オプション。このオプションは、Tableau Server (バージョン 2018.1 以前) で tabadmin により作成されたバックアップ ファイルから復元する場合にのみ指定します。

元のバックアップ ファイルを作成した Tableau Server からアイデンティティ ストアのタイプを変更する場合は、このキーを使用しないでください。ID ストアを変更するには、アイデンティティ ストアの変更 を参照してください。

`-po, --pg-only`

オプション。

リポジトリのみを復元します。

`-r, --restart-server`

オプション。

復元後にサーバーを再起動します。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance send-logs

指定したファイルを Tableau にアップロードし、サポートするケースと関連付けます。Tableau にファイルを正常にアップロードするには、Tableau Server が送信ログ サーバーと通信できる必要があります。詳細については、「インターネットとの通信」を参照してください。

シノプシス

```
tsm maintenance send-logs --case <case_number> --email <contact_
email> --file <path/to/file> [global options]
```

オプション

`-c,--case <case_number>`

必須。

サポートするケースの番号。

`-e,--email <contact_email>`

必須。

連絡先メール。

`-f, --file <path/to/file>`

必須。

送信するログファイルアーカイブの場所と名前を指定します。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance snapshot-backup complete

バージョン: コマンドはバージョン 2020.1 で追加され、Tableau Server が外部ファイルストア用に構成されている場合にのみ使用できます。

Tableau Server でスナップショットバックアッププロセスを完了します。外部ストレージのスナップショットバックアップを作成した後、この手順を実行します。

tsm maintenance snapshot-backup prepare コマンドと *tsm maintenance snapshot-backup complete* コマンドは、外部ファイルストアを使用して構成された Tableau Server インストール用

の Tableau Server データのバックアップを作成するために使用します。詳細については、「外部ファイル ストアによるバックアップと復元」を参照してください。

シノプシス

```
tsm maintenance snapshot-backup complete [options] [global options]
```

オプション

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance snapshot-backup prepare

バージョン: コマンドはバージョン 2020.1 で追加され、Tableau Server が外部ファイル ストア用に構成されている場合にのみ使用できます。

スナップショット バックアップの準備を行います。準備手順が完了したら、ネットワーク ストレージのスナップショット バックアップを作成できます。

`tsm maintenance snapshot-backup prepare` コマンドと`tsm maintenace snapshot-backup complete` コマンドは、外部ファイル ストアを使用して構成された Tableau Server インストールを行うための Tableau Server データのバックアップを作成するために使用します。詳細については、「外部ファイル ストアによるバックアップと復元」を参照してください。

シノプシス

```
tsm maintenance snapshot-backup prepare [options] [global options]
```

オプション

`--include-pg-backup`

任意の項目です。バージョン 2021.1 での追加以前のバージョンには、常に外部リポジトリのバックアップが含まれていました。

任意の項目です。

リポジトリのバックアップが作成され、ネットワーク共有にコピーされます。これは、外部リポジトリと外部ファイル ストアの両方が構成されている展開にのみ適用されます。詳細については、外部ファイル ストアによるバックアップと復元を参照してください。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance snapshot-backup restore

バージョン: コマンドはバージョン 2020.1 で追加され、Tableau Server が外部ファイル ストア用に構成されている場合にのみ使用できます。

ストレージ スナップショットから Tableau Server にリポジトリ バックアップを復元します。

詳細については、「外部ファイル ストアによるバックアップと復元」を参照してください。

シノプシス

```
tsm maintenance snapshot-backup restore [options] [global options]
```

オプション

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance validate-backup-basefilepath

バージョン: バージョン 2022.1 で追加されたコマンド

バックアップ/復元のベースファイルパスの場所に、バックアップおよび復元機能が正常に動作するための正しいパーミッションがあることを確認します。バックアップと復元のベースファイルパスを設定した後、これを実行します。詳細については、現在のファイルの場所を変更するを参照してください。

シノプシス

```
tsm maintenance validate-backup-basefilepath [options]
[global options]
```

オプション

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance validate-resources

サイトのワークブックおよびデータソースを検証します。サイトを移行する前にこのコマンドを使用して、サイトのインポートが失敗する原因となるワークブックやデータソースなどのサイトリソースに関する問題を検出します。リソースの問題の中には、ローカルソースから再発行することで解決できるものもあります。その他の問題では、Tableau サポートのサポートが必要になる場合があります。

シノプシス

```
tsm maintenance validate-resources --site-id <site ID>
[global options]
```

オプション

`-id,--site-id <site ID>`

必須。

リソースの検証を行っているサイトの ID。

`-r,--repair`

オプション。

無効なリソースの修復を試みます。修復できないものは出力に記載されています。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。

tsm maintenance ziplogs

`ziplogs` コマンドを使用して、Tableau Server ログ ファイルのアーカイブを作成します。

注: ziplogs コマンドを正常に実行できない場合は、Tableau Server ログを手動で圧縮できます。詳細については、「Linux での Tableau Server のトラブルシューティング」を参照してください。

シノプシス

```
tsm maintenance ziplogs [options] [global options]
```

オプション

`-a, --all`

オプション。

PostgreSQL データ以外のすべてのファイルを含めます。

`-d, --with-postgresql-data`

オプション。

Tableau Server が停止している場合は PostgreSQL データフォルダーを含め、Tableau Server が実行中の場合は PostgreSQL ダンプファイルを含めます。このフラグは、Tableau Server 外部リポジトリで実行されている展開では無視されます。

`--enddate "<mm/dd/yyyy H:mm>"`

オプション。時刻オプション (`H:mm`) がバージョン 2021.4.0 で追加されました。

含むべきログファイルの最終日付。このオプションは `--startdate` と一緒に使用する必要があり、`--minimumdate` オプションとは一緒に使用できません。このオプションを指定しない場合は、00:00 GMT (グリニッジ標準時) を開始時刻として、最大 2 日間のログが含まれます。

時刻オプションを含める場合は、日付と時刻を引用符で囲む必要があります。時間オプションは GMT を使用しますが、結果のログファイルは Tableau Server マシンのローカルタイムゾーンを使用して書き込まれます。

例: Tableau Server マシンのローカルタイムゾーンが PDT で、ログファイルを 2022 年 7 月 28 日の午前 7 時 (太平洋夏時間) に開始し、午後 7 時 (太平洋夏時間) に終了する場合は、次を使用します。

```
tsm maintenance ziplogs -f logs.zip --startdate "07/28/2022
14:00" --enddate "07/29/2022 02:00"
```

`-f, --file <name>`

オプション。

zip 形式で圧縮されたアーカイブファイルの名前を指定します。名前が指定されていない場合は logs.zip となります。ファイルは `basefilepath.log_archive` 変数で TSM に定義したディレクトリに書き込まれます。既定のディレクトリは次のとおりです。

/var/opt/tableau/tableau_server/data/tabsvc/files/log-archives/

ファイル パスとその変更方法の詳細については、tsm ファイル パスを参照してください。

-i, --description <string>

オプション。

指定したアーカイブ ファイルの説明が含まれます。

-l, --with-latest-dump

オプション。

いずれかのサービスがクラッシュすると、Tableau Server がダンプ ファイルを生成します。このオプションを設定すると、最新のサービス クラッシュ ダンプ ファイルが追加されます。このオプションを設定しない場合、結果として生成される ziplog にはダンプ ファイルが追加されません。

-m, --minimumdate <mm/dd/yyyy>

オプション。

ログ ファイルを最初に含めた日付。指定しない場合は、ログ ファイルの中から最大で 2 日間が含まれます。日付形式は mm/dd/yyyy で指定する必要があります。.このオプションは、--startdate および --enddate または --all と一緒に使用することはできません。

--nodes

任意の項目です。バージョン 2020.3 での追加

ziplog ファイルを作成するノードを指定します。指定しない場合、すべてのノードで ziplog ファイルが作成され、初期ノードに保存されます。ノードはコンマで区切ります。たとえば、ノード 2 と 4 のログを収集するには、次の手順を実行します。

```
tsm maintenance ziplogs --nodes node2,node4
```

`-o、--overwrite`

オプション。

既存の ziplog ファイルの上書き。同じ名前のファイルが既に存在していてこのオプションが使用されていない場合、ziplog コマンドは失敗します。

既定では、ファイルは次のように書き込まれます。

`/var/opt/tableau/tableau_server/data/tabsvc/files/log-archives/`

ファイル パスとその変更方法の詳細については、tsm ファイル パスを参照してください。

`--request-timeout <seconds>`

任意の項目です。

コマンドが終了するまでの待機秒数既定値は 7200 (120 分) です。

`--startdate "<mm/dd/yyyy H:mm>"`

オプション。時刻オプション (`H:mm`) がバージョン 2021.4.0 で追加されました。

含むべきログ ファイルの最終日付。このオプションは `--enddate` と一緒に使用する必要があり、`--minimumdate` オプションとは一緒に使用できません。このオプションを指定しない場合は、00:00 GMT (グリニッジ標準時) を開始時刻として、最大 2 日間のログが含まれます。

時刻オプションを含める場合は、日付と時刻を引用符で囲む必要があります。時間オプションは GMT を使用しますが、結果のログ ファイルは Tableau Server マシンのローカル タイム ゾーンを使用して書き込まれます。

例: Tableau Server マシンのローカル タイム ゾーンが PDT で、ログ ファイルを 2022 年 7 月 28 日の午前 7 時 (太平洋夏時間) に開始し、午後 7 時 (太平洋夏時間) に終了する場合は、次を使用します。

```
tsm maintenance ziplogs -f logs.zip --startdate "07/28/2022
14:00" --enddate "07/29/2022 02:00"
```

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm pending-changes

tsm pending-changes コマンドを使用すると、Tableau Server に対する保留中の構成やトポロジの変更を適用、破棄、表示できます。

TSM 構成時に入力したパスワードとシークレットは、保存後に暗号化されます。シークレットは、保留中の変更の適用中も適用前後も暗号化されたままの状態を維持します。シークレット ストレージの詳細については、サーバー シークレットの管理を参照してください。

- tsm pending-changes apply
- tsm pending-changes discard
- tsm pending-changes list

tsm pending-changes apply

tsm pending-changes apply コマンドを使用すると、保留中の構成やトポロジの変更を Tableau Server に適用できます。

保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。

シノプシス

```
tsm pending-changes apply [global options]
```

オプション

`-iw, --ignore-warnings`

オプション。

警告レベルの制約を無視します。

`--ignore-prompt`

オプション。

再起動のメッセージを表示させないようにします。メッセージの表示のみが抑制され、再起動の動作に変化はありません。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

tsm pending-changes discard

`tsm pending-changes discard` コマンドを使用して、保留中の構成や Tableau Server へのトポロジの変更を破棄します。

シノプシス

```
tsm pending-changes discard [options] [global options]
```

オプション

`--config-only`

オプション。

保留中の構成の変更のみを破棄します。

`--topology-only`

オプション。

保留中のトポロジの変更のみを破棄します。

tsm pending-changes list

Tableau Server に対する保留中の構成とトポロジの変更をリストします。サーバーの再起動が不要な変更は、再起動を必要としないものとしてリストされます。再起動が必要な変更が存在しない場合は、変更にサーバーの再起動が必要ないことを示すメッセージが表示されます。リスト内に再起動が必要な変更がある場合は、リスト全体の適用時に再起動が行われます。動的な構成やトポロジの変更の詳細については、「Tableau Server の動的トポロジの変更」を参照してください。

シノプシス

```
tsm pending-changes list [options] [global options]
```

オプション

```
--config-only
```

オプション。

保留中の構成の変更のみをリストします。

```
--topology-only
```

オプション。

保留中のトポロジの変更のみをリストします。

グローバル オプション

```
-h, --help
```

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザーアカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm register

tsm register コマンドを使用して、Tableau Server を登録します。このコマンドでは、登録ファイル テンプレートを表示したり(--template オプションを使用)、完了した登録ファイルへのパスを入力したり(--file オプションを使用) することができます。tsm register コマンドを呼び出す場合は、これらの 2 つのオプションのいずれかを使用する必要があります。

シノプシス

```
tsm register --template | --file <registration-filename>
[global options]
```

オプション

--file <registration-filename>

必須。

登録データが含まれるファイルへのパス。

--template

必須。

登録ファイルテンプレートを表示します。

グローバル オプション

-h, --help

オプション。

コマンドのヘルプを表示します。

-p, --password <password>

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm reset

tsm reset コマンドを使用して、初期管理ユーザーをクリアします。それにより新しい初期管理ユーザーを入力できるようになります。tsm reset の実行後、tabcmd initialuser コマンド

を再実行して新しい初期管理ユーザーを作成する必要があります。新しい名前を以前の管理ユーザーと同じユーザー名にすることはできません。

既存の初期管理ユーザーを使用するには、次の手順を実行します。

1. `tsm reset` 実行して、既存の初期管理ユーザーをリセットします。
2. `tabcmd initialuser` を実行して、新しい初期管理ユーザーを作成します。
3. `tsm reset` 実行して、その新しい管理ユーザーをリセットします。
4. `tabcmd initialuser` 実行して、元の初期管理ユーザーを作成し直します。

組織で Tableau アイデンティティストアの Active Directory または LDAP を使用している場合は、指定したアカウントとパスワードがディレクトリ内のアカウントと一致する必要があります。

シノプシス

```
tsm reset[option] [global options]
```

オプション

`-d,--delete-all-sessions`

オプション。

サーバーがリセットされるときに、すべてのアクティブなユーザー セッションを削除します。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm restart

tsm restart コマンドを使用して Tableau Server を再起動できます。このコマンドは、必要に応じてサーバーを停止してから、起動します。

シノプシス

```
tsm restart [global options]
```

オプション

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

グローバル オプション

```
-h, --help
```

オプション。

コマンドのヘルプを表示します。

```
-p, --password <password>
```

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

```
-s, --server https://<hostname>:8850
```

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm schedules

注: これらのコマンドは、tsm maintenance backup コマンドを使用してスケジュールされたバックアップ ジョブにのみ適用されます。バックグラウンダーによって実行されるジョブ (抽出更新、フロー、サブスクリプションなど) には適用されません。

`tsm schedules` コマンドを使用して、スケジュールされたバックアップ ジョブを管理できます。スケジュールされたバックアップ ジョブを作成するには、`tsm maintenance backup` コマンドを使用します。詳細については、tsm maintenance backup を参照してください。スケジュールされたバックアップの管理の詳細については、バックアップのスケジュール設定と管理を参照してください。

- tsm schedules delete
- tsm schedules list
- tsm schedules resume
- tsm schedules suspend
- tsm schedules update

tsm schedules delete

指定したスケジュールを削除します。

シノプシス

```
tsm schedules delete [-si <scheduleID> | -sn <scheduleName>]
[global options]
```

オプション

`-si,--schedule-id <scheduleID>`

--schedule-name が使用されていない場合は必須。

削除するスケジュールの ID。

`-sn,--schedule-name <scheduleName>`

--schedule-id が使用されていない場合は必須。

削除するスケジュールの名前。

tsm schedules list

サーバー上のスケジュールをリストします。

シノプシス

```
tsm schedules list [--next-run | --schedule-id <scheduleID> | --
schedule-name <scheduleName>] [global options]
```

オプション

`-nr,--next-run`

オプション。

スケジュールを "next run" の時間で昇順に並べ替えます。

`-si,--schedule-id <scheduleID>`

オプション。

詳細をリストするスケジュールの ID。

`-sn,--schedule-name <scheduleName>`

オプション。

スケジュールを名前でアルファベット順に並べ替えます。

tsm schedules resume

指定した中断スケジュールを再開します。

シノプシス

```
tsm schedules resume [-si <scheduleID> | -sn <scheduleName>]
[global options]
```

オプション

`-si,--schedule-id <scheduleID>`

--schedule-name が使用されていない場合は必須。

再開するスケジュールの ID。

`-sn,--schedule-name <scheduleName>`

--schedule-id が使用されていない場合は必須。

再開するスケジュールの名前。

tsm schedules suspend

指定したスケジュールを中断します。

シノプシス

```
tsm schedules suspend [-si <scheduleID> | -sn <scheduleName>]
[global options]
```

オプション

`-si,--schedule-id <scheduleID>`

--schedule-name が使用されていない場合は必須。

中断するスケジュールの ID。

`-sn,--schedule-name <scheduleName>`

--schedule-id が使用されていない場合は必須。

中断するスケジュールの名前。

tsm schedules update

指定したスケジュールを更新します。

シノプシス

```
tsm schedules update[-si <scheduleID> -st <time_to_run> -sr
<recurrence_frequency> -sd <day-or-days>] [global options]
```

オプション

`-si,--schedule-id <scheduleID>`

必須。

更新するスケジュールの ID。

`-sr,--schedule-recurrence <frequency>`

必須。

スケジュールの繰り返し頻度。有効なオプションは、"daily"、"weekly"、または "monthly" です。

`-st,--schedule-time <HH:MM>`

必須。

スケジュールを実行する必要がある時刻 (24 時間形式: HH:MM)。

`-sd,--schedule-days <day[,day]>`

オプション。

スケジュールを実行する日数週次スケジュールの場合は、1 ~ 7 (1 が月曜日、7 が日曜日) を使用します。月次スケジュールの場合は、1 ~ 31 を使用します。特定の月に日 (たとえば、2 月の 30 日) が存在しない場合は、その月の最後の有効な日が使用されます。複数の値はコンマで区切ります。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm security

`tsm security` コマンドを使って Tableau Server の外部 (ゲートウェイ) SSL または リポジトリ (Postgres) SSL のサポートを構成します。リポジトリの SSL 設定には、Tableau クライアント

(Tableau Desktop、Tableau Mobile、Web ブラウザーなど) からリポジトリへの直接接続を介して SSL を有効にするオプションが含まれます。

- tsm security authorize-credential-migration
- tsm security cancel-credential-migrations
- tsm security custom-cert
  - tsm security custom-cert add
  - tsm security custom-cert delete
  - tsm security custom-cert list
- tsm security custom-indexandsearch-ssl
  - tsm security custom-indexandsearch-ssl add
  - tsm security custom-indexandsearch-ssl list
- tsm security custom-tsm-ssl
  - tsm security custom-tsm-ssl disable
  - tsm security custom-tsm-ssl enable
  - tsm security custom-tsm-ssl list
- tsm security external-ssl
  - tsm security external-ssl disable
  - tsm security external-ssl enable
  - tsm security external-ssl list
- tsm security kms
  - tsm security kms set-mode aws
  - tsm security kms set-mode azure
  - tsm security kms set-mode local
  - tsm security kms status
- tsm security maestro-rserve-ssl
  - tsm security maestro-rserve-ssl disable
  - tsm security maestro-rserve-ssl enable
- tsm security maestro-tabpy-ssl
  - tsm security maestro-tabpy-ssl disable
  - tsm security maestro-tabpy-ssl enable
- tsm security regenerate-internal-tokens
- tsm security repository-ssl
  - tsm security repository-ssl disable
  - tsm security repository-ssl enable

    - tsm security repository-ssl get-certificate-file
    - tsm security repository-ssl list
- tsm security rotate-coordination-service-secrets
- 2020.2 リリースでは、Rserve および TabPy 分析拡張機能を構成するには、Tableau Server 管理ページを使用します。分析拡張機能を使用した接続の構成を参照してください。

前提条件

SSL を設定する前に証明書を取得して、Tableau Server ゲートウェイプロセスを実行するコンピュータにコピーしておく必要があります。クライアントからの直接接続を有効にする場合は、さらに準備が必要です。詳細については、次の記事を参照してください。

Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する

内部 Postgres 通信用に SSL を設定する

相互 (2 方向) SSL の詳細については、相互 SSL 認証の構成および「tsm authentication mutual-ssl コマンド」を参照してください。

tsm security authorize-credential-migration

Content Migration Tool を使用して、埋め込まれた認証資格情報を Tableau Server インストールから Tableau Cloud サイトに移行することを Tableau ユーザーに許可します。Tableau Server、Tableau Cloud ともに、コンテンツの移行には Advanced Management のライセンスが必要です。詳細については、「認証資格情報が埋め込まれたワークブックとデータソースの移行」を参照してください。

認証をキャンセルするには、`tsm security cancel-credential-migrations` コマンドを使用します。

シノプシス

```
tsm security authorize-credential-migration --source-site-url-
namespace <Tableau Server site ID> --destination-site-url-namespace
<Tableau Cloud site ID> --destination-server-url <Tableau Cloud
site url> --authorized-migration-runner <username> --destination-
public-encryption-key <public key>
```

オプション

`--source-site-url-namespace`

必須。Tableau Server サイトのサイト ID。サイト ID は、サイトを一意に識別するために URL で使用されます。

たとえば、West Coast Sales という名前のサイトには、west-coast-sales というサイト ID が設定されている可能性があります。

`--destination-site-url-namespace`

必須。Tableau Cloud サイトのサイト ID。サイト ID は、サイトを一意に識別するために URL で使用されます。

`--destination-server-url`

必須。Tableau Cloud サイトがデプロイされているポッドの URL。指定する URL には、末尾のスラッシュ(/) を含める必要があります。

Tableau Cloud にサインインすると、サイト URL の最初の部分にポッドが表示されます。たとえば、`https://10az.online.tableau.com/` は米国西部 (10AZ) ポッドです。ポッドの詳細については、Salesforce Trust のページを参照してください。

`--authorized-migration-runner`

必須。埋め込み認証資格情報の移行を許可された Tableau Server ユーザーのユーザー名。

`--destination-public-encryption-key`

必須。Tableau Cloud サイトで生成された公開鍵を指定します。

`--expiration-time-in-days`

オプション。承認が期限切れになるまでの日数。デフォルト値は 7 日です。

バージョン: バージョン 2023.1 で廃止されました。2023.1.0 以降、このオプションは無効になったため、使用するとエラーが発生します。有効期限は 7 日間としてハードコードされています。

例

次の例では、ユーザー「admin」に、Tableau Server サイト「ExampleA」から Tableau Cloud サイト「ExampleB」への、認証資格情報が埋め込まれたワークブックとパブリッシュされたデータソースの移行を承認しています。認証の有効期限は 9 日間です。

```
tsm security authorize-credential-migration --source-site-url-
namespace ExampleA --destination-site-url-namespace ExampleB --
destinationServerUrl https://10ay.online.tableau.com/ --authorized-
migration-runner admin --destination-public-encryption-key <public
key> --expiration-time-in-days 9
```

tsm security cancel-credential-migrations

Content Migration Tool を使用して埋め込まれた認証資格情報を移行するために付与された承認を取り消します。詳細については、「認証資格情報が埋め込まれたワークブックとデータソースの移行」を参照してください。

シノプシス

```
tsm security cancel-credential-migrations --source-site-url-
namespace <Tableau Server site ID>
```

オプション

`--source-site-url-namespace`

必須。Tableau Server サイトのサイトID。サイトID は、サイトを一意に識別するために URL で使用されます。

たとえば、West Coast Sales という名前のサイトには、west-coast-salesというサイトID が設定されている可能性があります。

tsm security custom-cert add

カスタム CA 証明書を Tableau Server に追加します。この証明書は、必要に応じて、SMTP サーバーとTableau Server 間の TLS 通信の信頼を確立するために使用されます。

カスタム証明書が既に存在する場合、このコマンドは失敗します。`tsm security custom-cert delete` コマンドを使用して、既存のカスタム証明書を削除できます。

注: このコマンドで追加する証明書は、TLS 接続の他の Tableau Server サービスで使用される場合があります。

災害復旧計画の一環として、証明書ファイルのバックアップを Tableau Server 外部の安全な場所に保管することをお勧めします。Tableau Server に追加する証明書ファイルは、クライアントファイル サービスによって他のノードに格納および配布されます。ただし、ファイルは復元可能な形式では格納されません。Tableau Server クライアントファイル サービスを参照してください。

シノプシス

```
tsm security custom-cert add --cert-file <file.crt>
[global options]
```

オプション

`-c, --cert-file <file.crt>`

必須。有効な PEM または DER 形式で証明書ファイルの名前を指定します。

tsm security custom-cert delete

サーバーの既存のカスタム証明書を削除します。これにより、新しいカスタム証明書を追加できるようになります。

シノプシス

```
tsm security custom-cert delete[global options]
```

tsm security custom-cert list

カスタム証明書の詳細を一覧表示します。

シノプシス

```
tsm security custom-cert list[global options]
```

tsm security custom-indexandsearch-ssl add

Tableau Server 2023.1 以降のインデックス サーバーと検索サーバーのカスタム証明書を追加します。SSL 実装は、Opensearch.org の TLS 実装に基づいています。詳細については、「TLS 証明書の設定」を参照してください。

`--admin <file.crt>`
: 必須。
管理者証明書ファイル。PEM エンコードされた有効な x509 証明書へのパスを指定します。拡張子は .crt です。

`--admin-key <file.key>`
: 必須。
RSA または DSA の有効な秘密鍵ファイル (PKCS#8) へのパスを指定します。拡張子は慣例により.keyです。

`-- ca <file.crt>`
: 必須。

信頼された CA ファイル。PEM エンコードされた有効な x509 証明書へのパスを指定します。拡張子は .crt です。

`--node <file.crt>`

必須。

ノード証明書ファイル。PEM エンコードされた有効な x509 証明書へのパスを指定します。拡張子は .crt です。このコマンドは、この証明書をクラスター内の各ノードに配布します。ワイルドカード証明書を使用して、1 つの証明書でノード識別名 (DN) の完全な配列を許可します。

`-- node-key <file.key>`

必須。

RSA または DSA の有効な秘密鍵ファイル (PKCS#8) へのパスを指定します。拡張子は慣例により.keyです。

シノプシス

```
tsm security custom-indexandsearch-ssl add --node <file.crt> --
admin <file.crt> --node-key <file.key> --admin-key <file.key> --ca
<file.crt> [parameters] [global options]
```

tsm security custom-indexandsearch-ssl list

インデックス サーバーと検索サーバーの SSL カスタム証明書の設定内容を一覧表示します。

シノプシス

```
tsm security custom-indexandsearch-ssl list[global options]
```

tsm security custom-tsm-ssl disable

TSM コントローラーへの接続に使用するカスタム SSL 証明書を無効にします。自動管理された自己署名証明書に戻します。

シノプシス

```
tsm security custom-tsm-ssl disable [global options]
```

tsm security custom-tsm-ssl enable

Tableau Server 2023.1 以降の TSM コントローラーへの接続用に、カスタム SSL 証明書を有効にします。すでに SSL を有効にしていて、期限切れの証明書を更新する必要がある場合は、このコマンドを使用します。

`-cf,--cert-file <file.crt>`

必須。

PEM エンコードされた有効な x509 証明書へのパスを指定します。拡張子は .crt です。証明書のサブジェクト名は、管理コントローラーが動作している Tableau コンピューターのホスト名または IP アドレスと一致する必要があります。デフォルトでは、管理コントローラーは Tableau Server 展開の初期ノードで実行されます。

`-kf,--key-file <file.key>`

必須。

RSA または DSA の有効な秘密鍵ファイル (PKCS#8) へのパスを指定します。拡張子は慣例により.keyです。このキーはパスフレーズで保護できません。

`--chain-file <file.crt>`

オプション。

証明書チェーンファイル (.crt) へのパスを指定します。

チェーンファイルは、サーバー証明書の証明書チェーンを形成するすべての証明書を連結したものです。

ファイル内のすべての証明書は PEM でエンコードされた x509 である必要があり、ファイルの拡張子は .crt でなければなりません (.pem ではありません)。

`--skip-validation`

オプション。

認証局のルート検証をスキップするには、このオプションを渡します。

シノプシス

```
tsm security custom-tsm-ssl enable --key-file <file.key> --cert-
file <file.crt> [global options]
```

tsm security custom-tsm-ssl list

TSM カスタム証明書の設定内容を一覧表示します。

シノプシス

```
tsm security custom-tsm-ssl list[global options]
```

tsm security external-ssl disable

サーバーの既存 SSL 構成設定を削除し、外部クライアントとサーバー間のトラフィックの暗号化を停止します。

シノプシス

```
tsm security external-ssl disable [global options]
```

tsm security external-ssl enable

外部 HTTP 通信を介した SSL の証明書およびキーファイルを有効にし、指定します。

シノプシス

```
tsm security external-ssl enable --cert-file <file.crt> --key-
file <file.key> [options] [global options]
```

オプション

```
--cert-file <file.crt>
```

必須。拡張子 .crt が付いた有効な PEM 暗号化 x509 証明書の名前を指定します。

```
--key-file <file.key>
```

必須。規則による拡張子 .key が付いた有効な RSA または DSA 秘密キー ファイルを指定します。

`--chain-file <chainfile.crt>`

証明書チェーン ファイル (.crt) を指定します。

Mac の Tableau Desktop には証明書チェーン ファイルが必要です。場合によっては、Tableau Mobile で証明書チェーン ファイルが必要になることがあります。

証明書プロバイダーによっては、Apache 用に 2 つの証明書を発行する場合があります。2 つ目の証明書はチェーン ファイルです。これは、サーバー証明書の証明書チェーンを構成するすべての証明書を連結したものです。

ファイル内のすべての証明書は PEM でエンコードされた x509 である必要があり、ファイルの拡張子は .crt でなければなりません (.pem ではありません)。

`--passphrase`

オプション。証明書ファイルのパスフレーズ。入力するパスフレーズは保存中に暗号化されます。

注: パスフレーズに証明書のキー ファイルを作成する場合、SAML に SSL 証明書キーを再利用することはできません。

`--protocols <list protocols>`

オプション。許可したい、または禁止したいトランスポート レイヤー セキュリティ (TLS) プロトコル バージョンをリストします。

TLS は SSL の向上バージョンです。Tableau Server は接続を認証し、暗号化するために TLS を使用します。許容される値には Apache でサポートされるプロトコル バージョンが含まれます。プロトコルを無効化するには、- の記号をプロトコル バージョンの先頭に追加します。

既定の設定: `"all, -SSLv2, -SSLv3"`

この規定は明示的に外部クライアントが Tableau Server に接続するために SSL v2 または SSL v3 プロトコルを使用することを許可しません。ただし、TLS v1 および TLS v1.1 を禁止することを推奨します。

特定の TLS を拒否する前に、ユーザーたちが Tableau Server に接続するブラウザーが TLS v1.2 をサポートすることを確認してください。ブラウザーが更新されるまで、TLS v1.1 のサポートを維持する必要がある可能性があります。

TLS v1 または v1.1 をサポートする必要がなければ、次のコマンドを使用して TLS v1.2 (値 `all` を使用して) を許可し、明示的に SSL v2、SSL v3、TLS v1、および TLS v1.1 を拒否します。

```
tsm security external-ssl enable --cert-file file.crt --key-
file file.key --protocols "all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1"
```

tsm security external-ssl list

ゲートウェイ外部 SSL の構成に関する設定のリストを表示します。リストには使用中の証明書ファイルの名称が含まれますが、場所は含まれません。

シノプシス

```
tsm security external-ssl list [global options]
```

tsm security kms set-mode aws

KMS モードを AWS に設定します。

AWS KMS から完全な ARN 文字列を取得する必要があります。この文字列は、AWS KMS 管理ページの [General configuration (一般設定)] セクションにあります。ARN は arn:aws:kms:<region>:<account>:key/<CMK_ID> の形式で表示されます (たとえば `arn:aws:kms:us-west-2:867530990073:key/1abc23de-fg45-6hij-7k89-1l0mn1234567`)。

詳細については、「AWS キー管理システム」を参照してください。

シノプシス

```
tsm security kms set-mode aws --key-arn "<arn>" --aws-region
"<region>" [global options]
```

オプション

`--key-arn`

必須。`--key-arn` オプションは、AWS KMS 管理ページの [General configuration (一般設定)] セクション内の ARN から文字列を直接コピーします。

`--aws-region`

必須。「Amazon API Gateway」表の [Region (地域)] 列で示されるとおりに地域を指定します。

例

たとえば、AWS KMS インスタンスが `us-west-2` リージョンで実行されていて、アカウント番号が `867530990073`、CMK キーが `1abc23de-fg45-6hij-7k89-1l0mn1234567` である場合、コマンドは次のようになります。

```
tsm security kms set-mode aws --aws-region "us-west-2" --key-arn
"arn:aws:kms:us-west-2:867530990073:key/1abc23de-fg45-6hij-7k89-
1l0mn1234567"
```

tsm security kms set-mode azure

KMS モードを [Azure Key Vault] に設定します。

> **注:** KMS モードは、`tsm security kms status` を実行すると"Azure Key Vault" と表示されますが、"azure" として設定します。

Azure Key Vault の名前と、Azure 内のキーの名前が必要になります。

詳細については、「Azure Key Vault」を参照してください。

シノプシス

```
tsm security kms set-mode azure --key-name "<key_name>" --vault-
name "<vault_name>" [global options]
```

オプション

`--key-name`

必須。Azure Key Vault に保存されている非対称キーの名前。

`--vault-name`

必須。Azure Key Vault の名前。

例

たとえば、Azure Key Vault の名前が `tabsrv-keyvault` で、キーの名前が `tabsrv-sandbox-key01` の場合、コマンドは次のようになります。

```
tsm security kms set-mode azure --key-name "tabsrv-sandbox-key01" -
-vault-name "tabsrv-keyvault"
```

tsm security kms set-mode local

KMS モードを [ローカル] に設定します。ローカルは、既定の KMS モードです。詳しくは、Tableau Server のキー管理システムを参照してください。

シノプシス

```
tsm security kms set-mode local [global options]
```

tsm security kms status

KMS 構成の状態を表示します。返されるステータスには、以下が含まれます。

- ステータス: [OK] は、複数ノードのインストール時に、Tableau またはコントローラーノードから KMS にアクセスできることを示します。
- モード: ローカル、AWS、または Azure Key Vault。使用されている KMS モードを示します。

- マスター暗号化キーの暗号化と復号:

  KMS にはマスター抽出キー (MEK) のコレクションが保管されます。各 MEK は次の情報を持ちます。

  - ID (8ddd70df-be67-4dbf-9c35-1f0aa2421521 など)
  - "暗号化または復号化キー" または "復号化専用キー" のステータス。キーが "暗号化または復号化キー" である場合、Tableau Server はそれを使用して新しいデータを暗号化します。そうでない場合、キーは復号化にのみ使用されます。
  - 作成タイムスタンプ ("Created at:2019-05-29T23:46:54Z" など)。
  - 暗号化または復号化キーへの最初の移行日時: キーが暗号化または復号化キーになった日時を示すタイムスタンプ。
  - 復号化専用キーへの移行日時: キーが復号化専用に移行した日時を示すタイムスタンプ。

返されるその他の値は KMS モードによって変わります。

KMS モードが AWS の場合は、次の情報が返されます。

- カスタマーマスターキー (CMK) の ARN (ID)
- CMK がある地域。
- 使用中のルートマスターキー (RMK) の ID。RMK は CMK によって暗号化されたキーです。Tableau Server は AWS KMS を呼び出すことによって CMK を復号化します。その後、RMK を使用してマスター抽出キー (MEK) を暗号化/復号化します。RMK は変更できますが、同時に 1 つしか存在できません。

KMS モードが Azure Key Vault の場合は、次の情報が返されます。

- Vault 名: Azure Key Vault の名前。
- Azure Key Vault キー名: コンテナー内のキーの名前。

シノプシス

```
tsm security kms status [global options]
```

tsm security maestro-rserve-ssl disable

Rserve 接続を無効にします。

詳細については、フローで R(Rserve)スクリプトを使用する」を参照してください。

tsm security maestro-rserve-ssl enable

Rserve サーバーとTableau Server バージョン 2019.3 以降の間の接続を構成します。

詳細については、フローで R(Rserve)スクリプトを使用する」を参照してください。

シノプシス

```
tsm security maestro-rserve-ssl enable --connection-type <maestro-
rserve-secure | maestro-rserve> --rserve-host <Rserve IP address or
host name> --rserve-port <Rserve port> --rserve-username <Rserve
username> --rserve-password <Rserve password> --rserve-connect-
timeout-ms <RServe connect timeout>
```

オプション

`--connection-type`

maestro-rserve-secure を選択してセキュリティで保護された接続を有効にするか、maestro-rserve を選択してセキュリティで保護されていない接続を有効にします。maestro-rserve-secureを選択した場合は、コマンドラインに証明書ファイルのパスを指定します。

`--rserve-host`

ホスト

`--rserve-port`

ポート

`--rserve-username`

ユーザー名

`--rserve-password`

パスワード

`--rserve-connect-timeout-ms`

接続タイムアウト(ミリ秒単位)。例: `--rserve-connect-timeout-ms 900000`。

tsm security maestro-tabpy-ssl disable

TabPy 接続を無効にします。

詳細については、フローで Python スクリプトを使用する」を参照してください。

tsm security maestro-tabpy-ssl enable

TabPy サーバーとTableau Server バージョン 2019.3 以降の間の接続を構成します。

詳細については、フローで Python スクリプトを使用する」を参照してください。

シノプシス

```
tsm security maestro-tabpy-ssl enable --connection-type <maestro-
tabpy-secure | maestro-tabpy> --tabpy-host <TabPy IP address or host
name> --tabpy-port <TabPy port> --tabpy-username <TabPy username> --
tabpy-password <TabPy password> --tabpy-connect-timeout-ms <TabPy
connect timeout>
```

オプション

`--connection-type`

`maestro-tabpy-secure` を選択してセキュリティで保護された接続を有効にするか、`maestro-tabpy` を選択してセキュリティで保護されていない接続を有効にします。`maestro-tabpy-secure`を選択した場合は、コマンドラインで証明書ファイル **-cf<certificate file path>** を指定します。

`--tabpy-host`

ホスト

`--tabpy-port`

ポート

`--tabpy-username`

ユーザー名

`--tabpy-password`

パスワード

`--tabpy-connect-timeout-ms`

接続タイムアウト(ミリ秒単位)。例: `--tabpy-connect-timeout-ms 900000`。

tsm security regenerate-internal-tokens

このコマンドを使用して、次の操作を実行します。

1. Tableau Server が実行中の場合は停止します。
2. Postgres リポジトリの検索サーバーに新しい内部 SSL 証明書を生成する。
3. 社内で管理されるすべてのパスワードに新規パスワードを生成する。

4. Postgres リポジトリのすべてのパスワードを更新する。
5. 資産キー管理で新しい暗号化キーを生成し、資産キーデータを新しいキーで暗号化する。
6. 構成シークレットで新しい暗号化キー(マスターキー)を生成し、これを使用して構成を暗号化する。
7. Tableau Server をこれらのすべてのシークレットで再構成し、更新する。分散展開では、このコマンドによって再構成が分散され、クラスタ内のすべてのノードで更新されます。
8. 新しいマスターキーを再生成し、マスターキーストアファイルに追加してから、内部で使用するための新しいセキュリティトークンを作成します。
9. Tableau Server を起動します。

このコマンドを実行後にクラスタにノードを追加する場合は、新しいノード構成ファイルを生成して、このコマンドで生成されるトーク、キー、およびシークレットを更新する必要があります。追加のノードのインストールと構成を参照してください。

内部パスワードの詳細については、サーバーシークレットの管理を参照してください。

シノプシス

```
tsm security regenerate-internal-tokens [options] [global options]
```

オプション

```
--ignore-prompt
```

オプション。

プロンプトを表示せずに(必要に応じて)再起動を実行します。このオプションはプロンプトのみを抑制します。再起動の動作に変化はありません。

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

tsm security repository-ssl disable

リポジトリと他のサーバー コンポーネントとの間のトラフィックの暗号化を停止して、Tableau クライアントからの直接接続のサポートを停止します。

シノプシス

```
tsm security repository-ssl disable [global-options]
```

tsm security repository-ssl enable

リポジトリがローカルの場合は、SSL を有効化し、Postgres リポジトリとその他のサーバー コンポーネント間の暗号化トラフィックに使用される、サーバーの .crt および .key ファイルを生成します。

バージョン 2021.4 以降、外部リポジトリを使用する場合は、外部 PostgreSQL リポジトリと Tableau Server コンポーネント間のトラフィックの暗号化に使用されるサーバーの .crt および .key ファイルをインポートします。

これを有効化することで、Tableau クライアントからサーバーへの直接接続を介した SSL も有効になります。

シノプシス

```
tsm security repository-ssl enable [options] [global options]
```

オプション

```
-i, --internal-only
```

オプション。このオプションは、リポジトリが Tableau Server に対してローカルであり、Tableau Server の外部で構成されていない場合にのみ適用されます。このオプションは、外部リポジトリを使用して構成された Tableau Server には使用しないでください。

`--internal-only` に設定されている場合、Tableau Server はリポジトリと他のサーバーコンポーネントとの間で SSL を使用し、SSL をサポートしても **tableau** または **readonly** (読み取り専用) ユーザーを通じて直接的に接続する場合は SSL を必要としません。

このオプションが設定されていない場合は、Tableau Server がリポジトリと他のサーバー コンポーネントとの間のトラフィック、および、Tableau クライアントとの直接的な接続 (**tableau** または **readonly** (読み取り専用) ユーザーを通じた接続) で SSL を必要とするように設定します。

このオプションを指定する場合は、クライアントからの直接接続を許可するようPostgres SSL を構成するで説明する手順も完了する必要があります。

`-c, --certificate`

オプション。バージョン 2021.4 での追加 このオプションは、外部リポジトリを使用して構成された Tableau Server にのみ適用可能であり、インストール後に SSL 接続を有効または無効にするために使用できます。

このオプションを使用すると、Tableau Server と外部リポジトリ間の SSL/TSL 接続の使用を有効にできます。このオプションを使用する場合は、外部リポジトリのファイル名を含む SSL 証明書ファイルへのフル パスを指定します。このファイルは、外部リポジトリを有効にするときに使用されるファイルと同じです。

tsm security repository-ssl get-certificate-file

Tableau リポジトリとの SSL 通信に使用されるパブリック証明書ファイルを取得します。証明書の取得前に、リポジトリの通信で SSL を有効にする必要があります。証明書ファイルは、Tableau Server クラスタでリポジトリの内部クライアントに自動分散されます。リモート クライアントが SSL 経由でリポジトリに接続できるようにするには、パブリック証明書ファイルを各クライアントにコピーする必要があります。

このコマンドは、ローカル リポジトリを使用する Tableau Server に対してのみ機能し、Tableau Server が外部リポジトリを使用して構成されている場合はエラーが表示されます。

シノプシス

```
tsm security repository-ssl get-certificate-file [global-options]
```

オプション

`-f, --file`

必須。

証明書ファイルを保存するフル パスとファイル名 (.cert 拡張子付き)。重複ファイルがある場合は上書きされます。

tsm security repository-ssl list

既存のリポジトリ(Postgres) の SSL 設定を返します。

シノプシス

```
tsm security repository-ssl list [global-options]
```

tsm security rotate-coordination-service-secrets

バージョン: バージョン 2022.1 での追加

安全な接続のために、コーディネーション サービスが使用する新しい証明書、キー、およびトラストストアを生成します。

シノプシス

```
tsm security rotate-coordination-service-secrets [options]
[global options]
```

オプション

`--coord-svc-restart-timeout <seconds>`

オプション。

コーディネーション サービスが再起動するまで、指定した秒数だけ待機します。既定: 1200 (20 分)

`--ignore-prompt`

オプション。

プロンプトを表示せずに (必要に応じて) 再起動を実行します。

`--request-timeout <seconds>`

オプション。

コマンドが終了するまで、指定した秒数だけ待機します。既定: 1800 (30 分)

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用し

ます。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm settings

`tsm settings` コマンドを使用して、構成値をエクスポート(取得) およびインポート(設定) できます。

- tsm settings clone
- tsm settings export
- tsm settings import

重要: このトピックで参照しているサーバー構成ファイルには、構成シークレットの暗号化に使用されるマスター キーストア ファイルのコピーが含まれます。ノード構成ファイルを保護するため、インポートおよびエクスポート操作におけるシークレットの保護で説明されているメカニズムを使用して、追加の対策を講じることを強く推奨します。

tsm settings clone

Tableau Server インストールの構成とトポロジ (外部サービスとポートを含む) で構成される "クローン ペイロード" を作成します。このペイロードは、Server インストールの正確なコピーを再作成するために使用できます。このコマンドは、外部リポジトリと外部ファイル ストアを含む Server インストールで最適に動作するように設計されています。クローン ペイロードを使用してインストールのコピーを作成する方法の詳細については、「Tableau Server のクローンを作成」を参照してください。

シノプシス

```
tsm settings clone --output-directory <output-directory>
[global options]
```

オプション

```
-d, --output-directory <output-directory>
```

必須。

クローン ペイロードを書き込む場所を指定します。

tsm settings export

現在のサーバー構成とトポロジをファイルにエクスポートします。

次のファイルは、`tsm settings import` や `tsm settings export` コマンドを使用しても、エクスポートまたはインポートすることができません。これらのファイルは手動で管理する必要があります。

- SAML 証明書ファイル
- SAML キー ファイル
- SAML IdP メタデータ ファイル
- OpenID.static.file
- Kerberos.keytab ファイル
- LDAP Kerberos keytab ファイル
- LDAP Kerberos conf ファイル
- 相互 SSL 証明書ファイル

- 相互 SSL 失効ファイル
- カスタマイズ ヘッダー ロゴ ファイル
- カスタマイズ サインイン ロゴ ファイル
- カスタマイズ コンパクト ロゴ ファイル

シノプシス

```
tsm settings export --output-config-file <path/to/output_file.json>
[global options]
```

オプション

`-f, --output-config-file <file>`

必須。

この操作で作成したファイルの場所と名前を指定します。

tsm settings import

サーバー構成またはトポロジをインポートします。

次のファイルは、`tsm settings export` や `tsm settings import` コマンドを使用しても、エクスポートまたはインポートすることができません。これらのファイルは手動で管理する必要があります。

- SAML 証明書ファイル
- SAML キー ファイル
- SAML IdP メタデータ ファイル
- OpenID.static.file
- Kerberos.keytab ファイル
- LDAP Kerberos keytab ファイル
- LDAP Kerberos conf ファイル
- 相互 SSL 証明書ファイル
- 相互 SSL 失効ファイル
- カスタマイズ ヘッダー ロゴ ファイル
- カスタマイズ サインイン ロゴ ファイル
- カスタマイズ コンパクト ロゴ ファイル

シノプシス

```
tsm settings import --import-config-file <path/to/import_file.json>
[global options]
```

オプション

`-f,--import-config-file <FILE>`

必須。

入力ファイルへのパス。

`--config-only`

オプション。

`--topology-only`

オプション。

`-frc, --force-keys`

オプション。

キーが以前に存在しなかった場合も、強制的に構成に追加します。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm sites

tsm sites コマンドを使用して、新しいサイトにインポートするために既存のサイトをエクスポートしたり(サイト移行とも呼ばれます)、新しいサイトをインポートすることもできます。unlock コマン

ドは、エラーでサイトがロック状態になる場合に利用できます。

tsm sites コマンドは、ローカル ファイル ストアを使用して、エクスポートデータとインポートデータを保持します。マルチノードTableau クラスタを実行している場合は、データエンジンプロセスを稼働しているTableau Server で tsm sites コマンドを実行する必要があります。データエンジンプロセスとそれを必要とするプロセスの詳細については、Tableau Server プロセスを参照してください。

注: Tableau Server のインスタンス間でサイトを移行する場合、移行先サイトは、移行元サイトの Tableau Server のバージョンと同じかそれ以降の Tableau Server のバージョン上にある必要があります。移行元サイトと移行先サイトの両方が、サポートされているバージョンの Tableau Server からのものである必要があります。

サイトを移行するための包括的手順については、サイトのエクスポートまたはインポートを参照してください。

- tsm sites export
- tsm sites import
- tsm sites import-verified
- tsm sites unlock

tsm sites export

指定された Tableau Server サイトを .zip ファイルにエクスポートします。ある特定のタイミングでサイトの設定をアーカイブするために、またはサイト移行プロセスの最初のステップを完了するためにサイトをエクスポートできます。

注: tsm sites import および tsm sites export コマンドは、エラーが発生した場合に、サイトをロック状態のままにすることができます。サイトをロック解除するには、tsm sites unlock コマンドを使用します。

シノプシス

```
tsm sites export --site-id <source-siteID> --file <export-file>
[options] [global options]
```

オプション

`-f,--file <export-file>`

必須。

Tableau Server がサイトの情報すべてを保存するファイルの名前を指定します。

このファイルは TSM の `basefilepath.site_export.exports` 変数に定義されているディレクトリに生成されます。既定のディレクトリは次のとおりです。

`/var/opt/tableau/tableau_server/data/tabsvc/files/siteexports`

ファイル パスとその変更方法の詳細については、tsm ファイル パスを参照してください。

`-id,--site-id <source-siteID>`

必須。

エクスポートしているサイトのサイトID。サイトID は、Web ブラウザーでサイトにサインインする際の URL から入手できます。サイトID をどこで調べるかについての情報は、ソースサイトとターゲットサイトの準備を参照してください。

`-ow,--overwrite`

オプション。

既に存在する同じ名前のエクスポートを上書きします。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 43200 (720 分) です。

tsm sites import

このコマンドは tsm sites export で作成した .zip ファイルを使用してエクスポートされた参照元サイト設定がどのように新しいターゲットサイトにマッピングされるかを示す一連の .csv ファイルを生成します。

既定では .zip ファイルが生成され、次にある siteexports ディレクトリに保存されます。

```
/var/opt/tableau/tableau_server/data/tabsvc/files/siteexports
```

このコマンドを使用する前に、.zip ファイルを Tableau がそこにあると期待するディレクトリにコピーしなくてはなりません。この場所は、TSM basefilepath.site_import.exports 変数に定義されています。既定のインポートディレクトリは以下です。

```
/var/opt/tableau/tableau_server/data/tabsvc/files/siteimports
```

ファイル パスとその変更方法の詳細については、tsm ファイル パスを参照してください。

> **注**: tsm sites import および tsm sites export コマンドは、エラーが発生した場合に、サイトをロック状態のままにすることができます。サイトをロック解除するには、tsm sites unlock コマンドを使用します。

シノプシス

```
tsm sites import --file <export-file.zip> --site-id <target-siteID>
[options] [global options]
```

オプション

`-f,--file <export-file.zip>`

必須。

tsm sites export プロセスで作成された .zip ファイルの名前であり、インポートディレクトリにコピーする必要があります。既定のディレクトリは次のとおりです。

/var/opt/tableau/tableau_server/data/tabsvc/files/siteimports

`-id,--site-id <target-siteID>`

必須。

インポート先の新しいサイト(ターゲットサイト)のサイトID。サイトIDをどこで調べるかについての情報は、ソースサイトとターゲットサイトの準備を参照してください。

`-c,--continue-on-ignorable-errors`

オプション。

無視可能なエラーが発生する場合にはサイトのインポートを続行します。これらのエラーは、特定のワークブックまたはデータソースのインポートでの問題を示している可能性があります。

`-k,--no-verify`

オプション。

マッピング ファイルの検証をスキップします。

`-m, --override-schedule-mapper <mapping-file.csv>`

オプション。

通常のマッピングを名前で上書きするためのスケジュール マッピング ファイル。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 7200 (120 分) です。

tsm sites import-verified

新しいサイトにインポートするため、エクスポートされたサイトの .csv マッピングファイルが含まれるディレクトリを指定します。これがサイト移行プロセスの最終ステップです。

シノプシス

```
tsm sites import-verified --import-job-dir <importjob-directory> --
site-id <target-siteID> [options] [global options]
```

オプション

`-id,--site-id <target-siteID>`

必須。

インポート先の新しいサイト(ターゲットサイト) のサイトID。サイトID をどこで調べるかについての情報は、ソースサイトとターゲットサイトの準備を参照してください。

`-w, --import-job-dir <importjob-directory>`

必須。

エクスポートされた(参照元) サイトからの .csv ファイルを含む mappings ディレクトリの親。この親ディレクトリの名前には import id および日時が含まれます。例:

```
/var/opt/tableau/tableau_
server/data/tabsvc/files/siteimports/working/import_ff00_
20180102022014457
```

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 7200 (120 分) です。

tsm sites unlock

このコマンドはサイトのロック解除に使用します。

オプション

`-id,--site-id <target-siteID>`

必須。

ロック解除しているサイトのサイトID。サイトID をどこで調べるかについての情報は、ソースサイトとターゲットサイトの準備を参照してください。

`-d, --desired-state <state to leave unlocked site in>`

オプション。

ロック解除後に移るサイトの状態。オプションは "active" と"suspended" です。指定しない場合、既定は "active" です。

例:

```
tsm sites unlock -id mysite -d suspended
```

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 300 (5 分) です。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm start

tsm start コマンドを使用して Tableau Server を起動できます。サーバーが既に実行されている場合、このコマンドは何も実行しません。

シノプシス

```
tsm start [option][global options]
```

オプション

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm status

`tsm status` コマンドを使用して、Tableau Server と、Tableau Server の一部として実行される個々のサービス (プロセス) のステータスを表示できます。

シノプシス

```
tsm status [global options]
```

オプション

`-v, --verbose`

オプション。

Tableau Server クラスターのすべてのノードのステータスを表示します。

`tsm status` は、Tableau Server ノードに関して可能性のある次のいずれかのステータスを返します。

- `RUNNING`: ノードは、どのサービスまたはプロセスについてもエラー ステータスなく稼働しています。

- DEGRADED: ノードが稼働していますが、1 つまたは複数のプライマリサービス (リポジトリなど) がエラー状態です。メッセージング サービスのインスタンスが 1 つあり、そのインスタンスに障害が発生した場合、Tableau Server は引き続き機能しますが、ステータスには「低下」と表示され、イベント メッセージが失われる可能性があります。詳細については、「Tableau Server メッセージング サービス」を参照してください。

- ERROR: ノードですべてのプライマリサービスまたはプロセスがエラー状態です。

- STOPPED: ノードは停止しており、エラー ステータスはありません。

tsm status に --verbose オプションを指定して実行すると、TSM は個々のサービス (プロセス) のステータスを返します。可能性のあるステータス メッセージは次のとおりです。

- is running: サービスは実行中です。

- status is unavailable: ステータスを判定できません (サービスの起動時など)。

- is in a degraded state: サービスは実行中ですが、エラーが返されました。このステータスは、サービスが適切にインストールされなかった、構成されていない、または何らかの形で失敗したことを示しています。

- is in an error state: サービスは実行中ですが、エラーが返されました。このステータスは、サービスが適切にインストールされなかったか、構成されていないことを示しています。

- is synchronizing: ファイル ストア プロセスがファイル ストアの別のインスタンスと同期しています。

- is decommissioning: ファイル ストア プロセスの使用を停止しています。

- is running (Active Repository): アクティブなリポジトリが実行されています。これは期待どおりのステータスです。

- is running (Passive Repository): パッシブ リポジトリが実行されています。2 つのリポジトリが構成されている場合、これは予期されたステータスです。

- `is stopped`: サービスが停止しています。このステータスは、サービスにエラーまたは問題が発生していることを意味するものではありません。一部のサービスは、必要なときにのみ実行されます (たとえば、データベース メンテナンス サービス)。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm stop

`tsm stop` コマンドを使用して Tableau Server を停止できます。Tableau Server が既に停止している場合、このコマンドは何も実行しません。

シノプシス

```
tsm stop [option] [global options]
```

オプション

`--ignore-node-status <nodeID>`

オプション。

サーバーが停止したかどうかを判別する場合は、指定されたノードのステータスを無視します。不良ノードを削除する場合に便利です。複数のノードはコンマで区切ります。

たとえば、ノード2、3、5 が適切に応答していない場合: `tsm stop --ignore-node-status node2,node3,node5`

注: バージョン 2020.1 で追加されたオプション

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm topology

`tsm topology` コマンドを使用してファイル ストア ノードを安全に取り外す準備をするか、読み取り/書き込みモードに戻します。また、リポジトリのフェールオーバー開始、ノードまたはポートのリスト取得、クラスタにノードを追加するために必要な bootstrap 構成ファイルの取得、ノードの削除、外部リポジトリと外部ファイル ストアの構成も実行できます。

重要: トポロジに変更を加える場合、変更を有効にするには、保留中の変更を適用する必要があります。詳細については、tsm pending-changes を参照してください。

- cleanup-coordination-service
- deploy-coordination-service
- external-services
  - gateway
    - gateway disable
    - gateway enable
    - gateway update
  - list
  - repository
    - disable
    - enable
    - replace-host
  - storage (filestore)
    - storage disable
    - storage enable
    - storage switch-share
- failover-repository

- filestore
  - decommission
  - recommission
- list-nodes
- list-ports
- node-nickname
  - list
  - remove
  - set
- nodes
  - get-bootstrap-file
- remove-nodes
- set-node-role
- set-ports
- set-process
- toggle-coordination-service

tsm topology cleanup-coordination-service

注: バージョン 2020.1.0 以降、すべての調整サービス アンサンブルのコマンドでは、サーバーの再起動が行われることを確認する「y/n」プロンプトの入力が必要になります。入力せずにこれらのコマンドを実行するには、`--ignore-prompt` オプションを含めます。

`tsm topology cleanup-coordination-service` コマンドを使用して、新しいアンサンブルを展開した後に稼働中でない Tableau Server 調整サービス アンサンブルを削除します。このコマンドによって、稼働中でない調整サービス アンサンブルのすべてのノードで古い調整サービス インスタンスを削除します。また、このコマンドは新しい調整サービス アンサンブルを展開した後で必要となります。調整サービス アンサンブルについての詳細は、調整サービス アンサンブルの展開を参照してください。

バージョン 2020.1.0 以降では、`tsm topology deploy-coordination-service` コマンドにより古いアンサンブルも削除されます。展開が失敗しない限り、このコマンドを個別に実行する必要はありません。

シノプシス

```
tsm topology cleanup-coordination-service [option] [global options]
```

オプション

```
--request-timeout <timeout in seconds>
```

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 2700 (45 分) です。

tsm topology deploy-coordination-service

注: バージョン 2020.1.0 以降、すべての調整サービス アンサンブルのコマンドでは、サーバーの再起動が行われることを確認する「y/n」プロンプトの入力が必要になります。入力せずにこれらのコマンドを実行するには、`--ignore-prompt` オプションを含めます。

`tsm topology deploy-coordination-service` コマンドを使用して、Tableau Server 調整サービス アンサンブルを展開できます。このコマンドは、サーバー クラスタの指定ノード上で実行される調整サービス インスタンス セットである調整サービス アンサンブルを展開します。クラスタ内で何台のノードに調整サービス インスタンスを含めるかなど、調整サービス アンサンブルの詳細については、調整サービス アンサンブルの展開を参照してください。

バージョン 2020.1.0 以降では、`tsm topology deploy-coordination-service` コマンドにより古いアンサンブルも削除されます。`cleanup-coordination-service` コマンドを個別に実行する必要はありません。

シノプシス

```
tsm topology deploy-coordination-service --nodes
<nodeID,nodeID,...> [option] [global-options]
```

オプション

```
-n, --nodes <nodeID,nodeID,...>
```

必須。

新しい調整サービス アンサンブル内に含まれるノードのコンマで区切られたノードID。クラスタ内の合計ノード数により、1、3 または 5 台の調整サービス ノードを指定できます。詳細については、調整サービス クォーラムを参照してください。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 2700 (45 分) です。

tsm topology external-services gateway disable

Tableau Server 上の独立したゲートウェイのすべてのインスタンスを無効にします。

シノプシス

```
tsm topology external-services gateway disable [options] [global
options]
```

オプション

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 2700 (45 分) です。

tsm topology external-services gateway enable

Tableau Server 上の独立したゲートウェイのインスタンスを有効にします。

シノプシス

```
tsm topology external-services gateway enable [options] [global
options]
```

オプション

`-c, --config <configuration-file>`

必須

独立したゲートウェイのすべてのインスタンスの構成詳細を含む JSON ファイルの名前を指定します。

tsm topology external-services gateway update

このコマンドを使用して、Tableau Server の独立したゲートウェイの構成を更新します。独立したゲートウェイのインスタンスを追加または削除する場合、または独立したゲートウェイをアップグレードする場合は、これを行う必要があります。このコマンドを実行する前に、独立したゲートウェイのコンピューター上の変更を収集し、構成ファイルを更新してください。

シノプシス

```
tsm topology external-services gateway update [option] [global
options]
```

オプション

`-c, --config <configuration-file>`

必須

独立したゲートウェイのすべてのインスタンスの構成詳細を含む JSON ファイルの名前を指定します。

tsm topology external-services list

tsm topology external-service-list コマンドは、Tableau Server 外部リポジトリ用に使用されるサービスを取得するのに使用します。たとえば、Amazon RDS を使用するように Tableau Server を構成している場合、次のメッセージが表示されます。

これらの外部で構成されたサービスは Tableau Server によって現在使用されています。

- pgsql

シノプシス

```
tsm topology external-service list [global options]
```

オプション

このコマンドにはオプションはありません。

tsm topology external-services repository disable -n nodeN

tsm topology external-services repository disable コマンドは、外部リポジトリの使用を停止し、ローカル リポジトリを使用するようにインストールを再構成するのに使用します。これにより、データがローカル リポジトリに移行され、ローカル リポジトリを使用するように Tableau Server が構成されます。

シノプシス

```
tsm topology external-services repository disable -n nodeN
```

オプション

```
-n, --node-name <nodeID>
```

必須。

リポジトリを移動する必要があるノードのノードID を指定します。

重要: これによりRDS インスタンスが停止したり削除されたりすることはありません。RDS インスタンスを削除する方法の詳細については、AWS の Web サイトで「DB インスタンスを削除する」を参照してください。

tsm topology external-services repository enable

tsm topology external-services repository enable コマンドは、外部リポジトリを使用するように Tableau Server を構成するのに使用します。このコマンドは、新しい Tableau Server のインストール時に外部リポジトリを構成するために使用できます。このコマンドを既に存在する実行中の Tableau Server で実行する場合、データがローカル ノードから外部リポジトリに移行され、移行完了後は外部リポジトリを使用するように Tableau Server を構成します。

シノプシス

```
tsm topology external-services repository enable -f <filename>.json
-c <ssl certificate file>.pem
```

オプション

--f <file name>

必須。

構成ファイルを保存するフル パスとファイル名。詳細については、Tableau Server リポジトリの再構成を参照してください。

--c <ssl certificate file>

バージョン 2021.2 および 2021.2.1 に必要です。バージョン 2021.2.2 以降ではオプションです。

SSL 構成の場合は、このオプションで使用できるように、証明書ファイルをダウンロードして、このファイルを指定します。

1. **Amazon RDS:**「SSL を使用して DB インスタンスへの接続を暗号化する」を参照してください。
2. **Azure Database:**「PostgreSQL 用 Azure Database の TLS 接続の構成」を参照してください。
3. **スタンドアロン PostgreSQL インスタンス:**「SSL の構成」を参照してください。

--no-ssl

オプション。このオプションは、2021.2.2 以降のバージョンで使用することができます。

つまり、外部リポジトリに接続するときに SSL は必要ありません。暗号化された接続を使用する必要がない場合は、暗号化されていない接続を許可するように外部リポジトリも構成する必要があります。このオプションを使用すると、外部リポジトリが TLS / SSL 接続をサポートするように構成されている場合、接続は暗号化されます。それ以外の場合、Tableau Server は暗号化されていない接続を使用します。

Tableau Server で使用するために外部リポジトリがすでに構成されているかどうかを確認するためのチェックをスキップします。このオプションでは、複数の Tableau Server インストールで同じリポジトリが使用され、エラーが発生する可能性があるため、通常はお勧めしません。このオプションは、テストまたは災害復旧の目的で役立つ場合があります。

--skip-state-check

オプション。このオプションは、2022.3.0 以降のバージョンで使用することができます。

Tableau Server で使用するために外部リポジトリがすでに構成されているかどうかを確認するためのチェックをスキップします。このオプションでは、テストまたは災害復旧の目的で役立つ場合がありますが、複数の Tableau Server インストールで同じリポジトリが使用される可能性があるため、通常の使用にはお勧めしません。

tsm topology external-services repository replace-host

このコマンドを使用して、指定された外部リポジトリを使用するように Tableau Server の構成設定を更新します。tsm topology external-services repository replace-host コマンドを使用して、現在の外部リポジトリからデータを移動することなく、新しい外部リポジトリを直ちに使用するように Tableau Server を再構成します。データを手動で移行する必要が生じる場合があります。この操作は、データ損失の可能性を十分に評価し、その影響を理解したうえで行う必要があります。

このコマンドは、次のシナリオで使用できます。

- RDS インスタンスで使用する SSL 証明書の予定される有効期限: RDS インスタンスを新しい証明書で更新する必要があります。また、新しい証明書ファイルを使用して RDS インスタンスに接続するように Tableau Server を構成する必要があります。
- 災害復旧: 災害復旧シナリオで新しい RDS インスタンスに接続します。詳細については、AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成を参照してください。

シノプシス

```
tsm topology external-services repository replace-host -f
<filename>.json -c <ssl certificate file>.pem
```

オプション

-f <file name>

必須。

構成ファイルを保存するフル パスとファイル名。詳細については、Tableau Server リポジトリの再構成を参照してください。

-c <ssl certificate file>

オプション。

証明書ファイルは、インスタンスへの接続を許可するためにインポートされる証明書です。RDS の場合、これは、インスタンスの証明書に署名するために使用される CA 証明書です。これは通常、最新のルート証明書である **rds-ca-XXXX-root.pem** ファイルとなります。RDS インスタンスで証明書が変更された場合は、このパラメーターを使用して Tableu Server を更新します。

詳細については、「SSL/TLS を使用した DB インスタンスへの接続の暗号化」を参照してください。

.pem ファイルの取得方法の詳細については、「SSL を使用して DB への接続を暗号化する」を参照してください。

--ignore-prompt

オプション。

プロンプトを表示せずにこのコマンドを実行します。

tsm topology external-services storage disable

ファイル ストアをローカルで実行するように Tableau Server を構成します。このコマンドを使用すると外部ファイル ストアが無効になり、ファイル ストア データが Tableau Server に移動します。

シノプシス

```
tsm topology external-services storage disable [options] [global options]
```

オプション

`-fsn <nodeID, nodeID,...>`

必須

ファイル ストアを構成するノードを指定します。複数のノードを指定できます。データはリストの最初のノードに移行されてから他のノードにレプリケートされます。

詳細については、ファイル ストアを再構成するを参照してください。

tsm topology external-services storage enable

外部ファイル ストアを使用して Tableau Server を構成します。外部ファイル ストアでは、ファイル ストア データが SAN または NAS に保存されます。

シノプシス

```
tsm topology external-services storage enable [options] [global
options]
```

オプション

`--network-share <network share mount point>`

必須

外部ファイル ストアに使用するネットワーク共有のマウント ポイントを指定します。例: `/mnt/<network share name>/tableau`

詳細については、ファイル ストアを再構成するを参照してください。

tsm topology external-services storage switch-share

このコマンドを使用して、外部サービスを別のネットワーク共有に移動します。たとえば、現在のネットワーク接続ストレージの寿命が来たため、新しいハードウェアが搭載されている新しいネットワーク接続ストレージを使用する必要がある場合などです。詳細については、ファイル ストアを再構成するを参照してください。

シノプシス

```
tsm topology external-services storage switch-share [option]
[global options]
```

オプション

```
--network-share <network share mount point>
```

必須

切り替え先のネットワーク共有のマウントポイントを指定します。例: `/mnt/<network share name>/tableau`

tsm topology failover-repository

`tsm topology failover-repository` コマンドを使用して、現在のアクティブなリポジトリから第二の、パッシブなリポジトリへのリポジトリのフェールオーバーを手動で開始できます。

`tsm topology failover-repository` コマンドは永続的です。フェールオーバーリポジトリは、このコマンドを再度発行するか、Tableau Server がそのように構成されている場合は自動フェールオーバーが発生するまでアクティブなリポジトリであり続けます。優先されるアクティブなリポジトリを構成済みの場合、`--preferred` オプションを使用してそのリポジトリへ切り替えます。優先するアクティブなリポジトリを構成する方法の詳細については、Tableau Server リポジトリを参照してください。Tableau Server が高可用性用に設定されている場合、必要時のリポジトリのフェールオーバーは自動です。リポジトリを手動でフェールオーバーするには、failover-repository コマンドを使用してください。

シノプシス

```
tsm topology failover-repository --preferred | --target <node_id>
[global options]
```

オプション

```
-r, --preferred
```

-t または --target が使用されていない場合は必須。

リポジトリフェールオーバーのターゲットとして、構成され優先ノードを使用します。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

`-t, --target <node_id>`

-r または --preferred が使用されていない場合は必須。

フェールオーバーが発生するターゲットノードのノード ID。`tsm topology list-nodes` コマンドを使用してノード ID を検索します。

tsm topology filestore decommission

`tsm topology filestore decommission` コマンドを使用して、ファイルストアノードを安全に取り外す準備をする必要があります。このコマンドは指定ノードを読み取り専用モードにして、指定ノード上に固有のコンテンツがないようにします。

コミッション解除の結果としてファイルストアノードが 1 台だけになってしまう場合、`--override` オプションを使用しないとコミッション解除が失敗します。

シノプシス

```
tsm topology filestore decommission --nodes <nodeID,nodeID,...>
[options] [global options]
```

オプション

`-n, --nodes <nodeID,nodeID,...>`

必須。

コンマで区切られたノード ID で指定されたコミッション解除する 1 台以上のノードのリスト。

`--delete-filestore`

オプション。

コミッション解除されていない場合でも、強制的にファイル ストアを削除します。このオプションを使用するのは、ファイル ストアがあるノードがエラー状態で、コミッション解除を行えない場合のみにしてください。ノード上の一意のファイルが完全に削除されます。

`-o, --override`

オプション。

対象のファイル ストア ノードを削除すると、ファイル ストア ノードが 1 台になってしまう場合に通常発生する警告または失敗を上書きします。このオプションは、`--delete-filestore` と一緒に使用することはできません。

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

tsm topology filestore recommission

`tsm topology filestore recommission` コマンドを使用してコミッション解除されたノードをすべて読み取り・書き込みモードに戻します。

シノプシス

```
tsm topology filestore recommission --nodes <nodeID,nodeID,...>
[global options]
```

オプション

`-n, --nodes <nodeID,nodeID,...>`

必須。

コンマで区切られたノード ID で指定された再コミッションする 1 台以上のノードのリスト。

tsm topology list-nodes

クラスター内のノードを表示し、(オプションで) 各ノードのサービスを表示します。

シノプシス

```
tsm topology list-nodes [options] [global options]
```

オプション

-v, --verbose

オプション。

各ノードの ID、ノードの役割 (詳細については、以下の「`set-node-role`」を参照)、ノードのアドレス、各ノードでのプロセスを表示します。

tsm topology list-ports

クラスター内のポートを表示します。

シノプシス

```
tsm topology list-ports [options] [global options]
```

オプション

`--node-name <nodeID>`

オプション。

ポートをリストするノードを指定します。

`--service-name`

オプション。

ポートをリストするサービスを指定します。

tsm topology node-nickname list

クラスタ内のノードのニックネームを表示します。

シノプシス

```
tsm topology node-nickname list [options] [global options]
```

オプション

```
--nodes <nodeID,nodeID,...>
```

任意の項目です。

ニックネーム付きで一覧表示するノードのノードIDを指定します。

tsm topology node-nickname remove

指定されたノードからニックネームを削除します。

シノプシス

```
tsm topology node-nickname remove [options] [global options]
```

オプション

```
--all
```

--nodesが指定されていない場合は必須です。

クラスタ内のすべてのノードからニックネームを削除します。

```
--nodes <nodeID,nodeID,...>
```

--allが指定されていない場合は必須です。

ニックネームを削除するノードのノードIDを指定します。

tsm topology node-nickname set

指定したノードのニックネームを設定します。

シノプシス

```
tsm topology node-nickname set [options] [global options]
```

オプション

`-id, --node <nodeID>`

必須。

ニックネームを設定するノードを指定します。

`-nn, --nickname <name>`

必須。

指定したノードのニックネームです。

tsm topology nodes get-bootstrap-file

`tsm topology nodes get-bootstrap-file` コマンドを使用して、クラスタへの新しいノードの追加に必要な bootstrap ファイルを取得できます。

重要: bootstrap ファイルには、構成シークレットの暗号化に使用されるマスター キーストア ファイルのコピーが含まれています。また、このファイルには事前定義された時間で有効となる埋め込み認証資格情報も含まれており(tabadmincontroller.auth.expiration.minutes を参照)、セッション Cookie として機能します。booststrap ファイルを保護するために追加措置を講じること強くお勧めします。

次のコマンド セットは、booststrap ファイルの出力を暗号化する方法の例です。この方法は、インポートおよびエクスポート操作におけるシークレットの保護で詳しく説明した暗号化プロセスに似ています。

ただし、この方法では、次のような末尾演算子 `&& \` を使用して別々の引数として渡す必要があります。

```
mkfifo -m 600 /tmp/secure1 && \

tsm topology nodes get-bootstrap-file --file /tmp/secure1 && \

gpg --symmetric --batch --yes --passphrase-file
~/.secrets/pgppassphrase.txt --cipher-algo AES256 --output
encrypted.enc < /tmp/secure1 && \

rm /tmp/secure1
```

シノプシス

```
tsm topology nodes get-bootstrap-file --file <path\file>.json
[global options]
```

オプション

`-f,--file <file>`

必須。

構成ファイルを保存するフル パスとファイル名。重複ファイルがある場合は上書きされます。

`-nec,--no-embedded-credential`

オプション。

バージョン 2019.3 での追加

既定では埋め込みの認証資格情報は bootstrap ファイルに含まれています。このオプションは、bootstrap ファイルに認証資格情報を含めない場合に使用します。埋め込みの認証資格情報は一時的なもので、`tabadmincontroller.auth.expiration.minutes` 構成キーの値に基づいて有効期限が切れますが、既定では 120 分間です。

> **注:** 埋め込みの認証資格情報をサーバーレベルで含める機能は、構成オプションを使用して無効にすることができます。詳細については、features.PasswordlessBootstrapInitを参照してください。

tsm topology remove-nodes

クラスターからノードを削除します。

ノードの削除を完了するには、`tsm pending-changes apply` コマンドも実行する必要があります。一部のシナリオでは、ノードを削除する前にプロセスを移動または再展開する必要があります。ノードの削除を参照してください。

ノードを削除してクラスタに再追加する場合は、まず obliterate スクリプトを実行して Tableau を完全に削除し、その後、新しいノードを追加する通常のプロセスでノードを再インストールする必要があります。詳細については、コンピュータからTableau Server を削除するおよび追加のノードのインストールと構成を参照してください。

> **注**: クラスタからノードを削除する際は、過去のどこかの時点でそのノード上にプロセスが構成されていたはずです。プロセスを構成していないノードを削除しようとしている場合、まずプロセスを追加して、`tsm pending-changes apply` を実行してから、ノードを削除しなくてはなりません。

シノプシス

```
tsm topology remove-nodes --nodes <nodeID,nodeID,...>
[global options]
```

オプション

`-n, --nodes <nodeID,nodeID,...>`

必須。

削除するノードを1つ以上指定します。複数のノードを指定する場合は、ノードIDの間をコンマで区切ります。

tsm topology set-node-role

バックグラウンダーおよび抽出クエリノードの役割を設定します。これにより、ノード上で実行されるタスクのタイプが決まります。次のノードの役割は、マルチノードクラスタがある場合に役立ちます。異なるノードの役割には、Advanced Managementとデータ管理のライセンスのいずれか、またはその両方が必要な場合があります。ライセンス要件の詳細については、ノードロールによるワークロード管理を参照してください。

注: ノードの役割を構成するには、サーバーを再起動する必要があり、いくらかのダウンタイムが必要になります。詳細については、tsm pending-changesを参照してください。

シノプシス

```
tsm topology set-node-role [options] [global options]
```

オプション

```
-n, --nodes <nodeID,nodeID,...>
```

必須。

ノードIDで指定され、コンマで区切られたノードの役割を設定する1台以上のノードのリスト。ノード間にスペースはありません。

```
-r --role <all-jobs,flows,no-flows,extract-
refreshes,subscriptions,extract-refreshes-and-subscriptions,no-
extract-refreshes,no-subscriptions,no-extract-refreshes-and-
subscriptions,extract-queries,extract-queries-interactive>
```

必須

指定されたノードの役割を設定します。このオプションの有効な値は次のとおりです。

- all-jobs: バックグラウンダーはすべてのタイプのジョブを実行します。
- flows: バックグラウンダーはフロー実行ジョブのみを実行します。
- no-flows: バックグラウンダーはフロー実行ジョブを実行しません。
- extract-refreshes: バックグラウンダーは抽出更新ジョブのみを実行します。これには、フロー出力が生成する抽出を含めすべての抽出における増分更新、完全更新、暗号化、復号化が含まれます。
- subscriptions: バックグラウンダーはサブスクリプション ジョブのみを実行します。
- extract-refreshes-and-subscriptions: バックグラウンダーは、フロー出力が作成する抽出を含めすべての抽出における extract-refreshes、暗号化、復号化のほか、サブスクリプション ジョブを実行します。
- no-extract-refreshes: バックグラウンダーは、フロー出力から作成された抽出を含め extract-refreshes、抽出の暗号化と復号化を除くすべてのジョブを実行します。
- no-subscriptions: バックグラウンダーは、サブスクリプションを除くすべてのジョブを実行します。
- no-extract-refreshes-and-subscriptions: バックグラウンダーは、フロー出力が作成した抽出を含めすべての抽出における extract-refreshes、暗号化、および復号化を除くすべてのジョブのほか、サブスクリプションを実行します。
- extract-queries: 選択されたノードは all-jobs として実行され、抽出クエリの処理を優先して実行します。
- Extract-queries-interactive: 選択したノードは、すべてのジョブとして実行され、インタラクティブな抽出クエリの処理に優先順位を付けます。このクエリは、ユーザーが画面を見て抽出ベースのダッシュボードの読み込みを待機しているときに実行されるものです。これは高度な設定であるため、クラスターのサブスクリプションおよびアラート ジョブのワークロードの負荷が大きく、スケジュールされたロードと同じ時間に実行される Viz の読み込み時間のパフォーマンスが低下する場合にのみ使用してください。
- system: バックグラウンダーは、クラッシュしたジョブのクリーニング、データベース イベントの取得、Active Directory の同期など、他の Tableau Server プロセスと相互に作

用するシステム メンテナンス ジョブのみを実行します。

- no-system: バックグラウンダーは、システム メンテナンス ジョブを除くすべてのジョブを実行します。

tsm topology set-ports

サービス インスタンスのポートを設定します。

シノプシス

```
tsm topology set-ports --node-name <nodeID> --port-name <port_name>
--port-value <port_value> [options] [global options]
```

オプション

`-i, --instance <instance_id>`

オプション。

サービスのインスタンス ID を指定します。指定しない場合は既定で 0 (ゼロ) に設定されます。

`-n, --node-name <nodeID>`

必須。

ノードのノード ID を指定します。

`-pn, --port-name <port_name>`

必須。

設定するポートの名前。`service_name:port_type`の形式にする必要があります。ポートのタイプを指定しない場合、プライマリ ポートが使用されます。ポート名構文に関しては、動的マッピングされたポートを参照してください。

`-pv, --port-value <port_value>`

必須。

設定するポート。

`-r, --restart`

オプション。

必要に応じて、Tableau Server の再起動を要求するプロンプトが表示されないようにします。

tsm topology set-process

ノードのプロセスのインスタンス数を設定します。ノードに指定されたプロセスがすでにある場合は、指定されたカウントに一致するようにこの数が更新されます。

- 一度に 1 つのプロセスしか設定できません。複数のプロセスを指定した場合、最初のプロセス以外のプロセスは、通知なく無視されます。
- プロセスは、各ノードで一度に 1 つずつ設定する必要があります。複数のノードを指定すると、コマンドは "invalid node name (無効なノード名)" エラーを表示します。

ノード上のプロセス数を更新したら、保留中の変更も適用する必要があります。ほとんどの場合、ここでサーバーの再起動が必要となります (メッセージが表示されます) が、サーバーを再起動せずに動的なトポロジの変更を実行できる特殊なケースもあります。詳細については、Tableau Server の動的トポロジの変更を参照してください。

注: プロセス名の完全なリストについては、Tableau Server プロセスを参照してください。

シノプシス

```
tsm topology set-process --count <process_count> --node <nodeID> --
process <process_name> [global options]
```

オプション

`-c, --count <process_count>`

必須。

設定するプロセスのカウント(インスタンス数)。

`-n, --node <nodeID>`

必須。

プロセスを設定するノードのノードID を指定します。

`-pr, --process <process_name>`

必須。

設定するプロセスの名前。

tsm topology toggle-coordination-service

注: バージョン2020.1.0 以降、すべての調整サービス アンサンブルのコマンドでは、サーバーの再起動が行われることを確認する「y/n」プロンプトの入力が必要になります。入力せずにこれらのコマンドを実行するには、`--ignore-prompt` オプションを含めます。

`tsm topology toggle-coordination-service` コマンドを使用して、調整サービス アンサンブル間を切り替えられます。調整サービス アンサンブルについての詳細は、調整サービス アンサンブルの展開を参照してください。

バージョン2020.1.0 以降では、`tsm topology deploy-coordination-service` コマンドにより新しいアンサンブルにも切り替わります。このコマンドを個別に実行する必要はありません。

シノプシス

```
tsm topology toggle-coordination-service [option] [global options]
```

オプション

`--request-timeout <timeout in seconds>`

オプション。

コマンドが終了するまで指定した時間待機します。既定値は 1800 (30 分) です。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm user-identity-store

`tsm user-identity-store` コマンドを使用すると、Tableau Server 用のアイデンティティストアの設定を初期の構成の後で変更できます。

アイデンティティストアの初期の構成は、インストール プロセスの一部です。初期ノード設定の構成を参照してください。

アイデンティティストアの概念については、アイデンティティストアを参照してください。

LDAP/Active Directory の構成リファレンス表については、外部のアイデンティティストアの構成リファレンスを参照してください。

- get-group-mappings
- get-user-mappings
- list
- set-connection
- set-group-mappings
- set-user-mappings
- verify-group-mappings
- verify-user-mappings

tsm user-identity-store get-group-mappings [options]

アイデンティティストア グループ マッピングを表示します。

シノプシス

```
tsm user-identity-store get-group-mappings [global options]
```

tsm user-identity-store get-user-mappings [options]

アイデンティティストア ユーザー マッピングを表示します。

シノプシス

```
tsm user-identity-store get-user-mappings [global options]
```

tsm user-identity-store list [options]

user-identity 構成をリストします。

シノプシス

```
tsm user-identity-store list [options] [global options]
```

オプション

```
-v, --verbose
```

オプション。

すべての構成パラメーターをリストします。

tsm user-identity-store set-connection [options]

アイデンティティストア接続パラメーターを設定します。

シノプシス

```
tsm user-identity-store set-connection --kerbkeytab <kerbkeytab>
[options] [global options]
```

オプション

```
-b,--bind <username and password | Kerberos>
```

オプション。

LDAP バインド タイプを設定します。

`-d,--domain <domain>`

オプション。

ドメイン名です。

`-hn,--hostname <hostname>`

オプション。

LDAP サーバーのホスト名です。この値のホスト名または IP アドレスを入力できます。ここで指定するホストは、プライマリドメインのユーザー/グループ クエリに使用されます。ユーザー/グループ クエリが他のドメインにある場合、Tableau Server は DNS にクエリを実行して適切なドメイン コントローラーを識別します。

`-kc,--kerbconfig <kerbconfig>`

オプション。

Kerberos 構成ファイル パスです。

`-kp,--kerbprincipal <kerbprincipal>`

オプション。

Kerberos Principal です。

`-kt,--kerbkeytab <kerbkeytab>`

必須。

Kerberos keytab ファイルのパスです。

`-l,--port <port>`

オプション。

LDAP ポート値を設定します。

`-lp,--ldappassword <ldappassword>`

オプション。

LDAP パスワードです。

`-lu,--ldapusername <ldapusername>`

オプション。

LDAP ユーザー名の値を設定します。

`-n,--nickname <nickname>`

オプション。

NetBIOS 名 (ニックネーム) です。

tsm user-identity-store set-group-mappings [options]

アイデンティティストアのグループ マッピングを設定し、任意またはカスタムのスキーマを実装する LDAP ディレクトリを構成します。

シノプシス

```
tsm user-identity-store set-group-mappings [options]
[global options]
```

オプション

`-b,--basefilter <groupbasefilter>`

オプション。

グループの BaseFilter (基本フィルター) 値を設定します。

`-cn,--classnames <group_classnames>`

オプション。

既定のユーザー クラス名値 ("group" 文字列を含む) をここで設定した値で上書きします。コンマで区切って複数のクラス名を入力できます。

`-d,--description <description>`

オプション。

グループの説明です。

`-e,--groupemail <groupemail>`

オプション。

Group Email (グループのメール) 値です。

`-m,--member <member>`

オプション。

グループのメンバーを設定します。

`-n,--groupname <groupname>`

オプション。

グループの名前です。

tsm user-identity-store set-user-mappings [options]

アイデンティティストアのユーザー マッピングを設定し、任意またはカスタムのスキーマを実装する LDAP ディレクトリを構成します。

シノプシス

```
tsm user-identity-store set-user-mappings --certificate
<certificate> [options] [global options]
```

オプション

`-c,--certificate <certificate>`

オプション。

ユーザーの証明書ファイルの場所です。

`-cn,--classnames <user_classnames>`

オプション。

既定のユーザー クラス名値 ("user" および "inetOrgPerson") をここで設定した値で上書きします。コンマで区切って複数のクラス名を入力できます。

`-dn,--displayname <displayname>`

オプション。

ユーザーの表示名です。

`-e,--email <email>`

オプション。

ユーザーの電子メール アドレスです。

`-jp,--jpegphoto <jpegfile>`

オプション。

ユーザーの jpeg 画像の場所です。

`-m,--memberof <groupname>`

オプション。

ユーザーがメンバーになっているグループです。

`-t,--thumbnail <thumbnail>`

オプション。

ユーザーのサムネイルの場所です。

`-b,--basefilter <groupbasefilter>`

オプション。

ユーザーの BaseFilter (基本フィルター) です。

`-uu,--ldapusername <ldapusername>`

オプション。

ユーザー名。

tsm user-identity-store verify-group-mappings [options]

LDAP グループ マッピングの構成を検証します。

シノプシス

```
tsm user-identity-store verify-group-mappings --verify <group_name>
[global options]
```

オプション

`-v,--verify <group_name>`

オプション。

検索するグループの名前。

tsm user-identity-store verify-user-mappings [options]

LDAP ユーザー マッピングの構成を検証します。

シノプシス

```
tsm user-identity-store verify-user-mappings --verify <user_name>
[global options]
```

オプション

`-v,--verify <user_name>`

オプション。

検索するユーザーの名前。

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて `-u` または `--username` と一緒に使用します。

`-u` または `--username` で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

```
--password 'my password'
```

`-s, --server https://<hostname>:8850`

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は `https` で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用し

ます。例: `https://<tsm_hostname>:8850`。サーバーが指定されていない場合は、仮に `https://<localhost | dnsname>:8850` とします。

`--trust-admin-controller-cert`

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

`-u, --username <user>`

アクティブなセッションがない場合は、必要に応じて `-p` または `--password` と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm version

`tsm version` コマンドを使用して、TSM および Tableau Server のバージョンを取得できます。

シノプシス

```
tsm version [global options]
```

グローバル オプション

`-h, --help`

オプション。

コマンドのヘルプを表示します。

`-p, --password <password>`

アクティブなセッションがない場合は、必要に応じて -u または --username と一緒に使用します。

-u または --username で指定されたユーザーのパスワードを指定します。

パスワードにスペースや特殊文字が含まれる場合、パスワードを引用符で囲みます。

--password 'my password'

-s, --server https://<hostname>:8850

オプション。

Tableau Services Manager で指定したアドレスを使用します。URL は https で始まる必要があります。また、ポート 8850 を含めるとともに、IP アドレスではなくサーバー名を使用します。例: https://<tsm_hostname>:8850。サーバーが指定されていない場合は、仮に https://<localhost | dnsname>:8850 とします。

--trust-admin-controller-cert

オプション。

このフラグは、TSM コントローラーで自己署名証明書を信頼するために使用します。証明書の信頼および CLI 接続の詳細については、TSM クライアントの接続を参照してください。

-u, --username <user>

アクティブなセッションがない場合は、必要に応じて -p または --password と一緒に使用します。

ユーザー アカウントを指定します。このオプションが含まれていない場合は、コマンドがサインインに使用した認証資格情報を使用して実行されています。

tsm ファイル パス

特定の tsm コマンドで、既定の場所に対してファイルの読み書きを行います。これらの既定の場所は、コマンドごとに定義される basefilepath 構成キーによって決まります。tsm を使用して、この

キーの現在の値を表示し、場所を変更することができます。

ファイルの既定の場所

`tsm maintenance backup`、`restore`、`send-logs`、`ziplogs` プロセスおよび `tsm sites export`、`sites import` プロセスの実行中、Tableau Server はこれらのコマンドが作成および使用するファイル用に既定の場所を使用します。

Tableau Server のバックアップに必要なディスク空き容量の詳細については、バックアップのディスク容量使用率を参照してください。

既定のディレクトリは次のとおりです。

- tsm maintenance コマンド:
    - backup— バックアップ用の `.tsbak` ファイルが初期ノードのデータディレクトリにある一時的な場所で作成された後、次の場所に保存されます。

      `/var/opt/tableau/tableau_server/data/tabsvc/files/backups/`
    - restore— 復元プロセスで復元するバックアップ ファイルの場所:

      `/var/opt/tableau/tableau_server/data/tabsvc/files/backups/`
    - send-logs— ログ送信で送信するログ ファイルの場所:

      `/var/opt/tableau/tableau_server/data/tabsvc/files/backups/`
    - ziplogs—ziplog ファイルの生成場所:

      `/var/opt/tableau/tableau_server/data/tabsvc/files/log-archives`
- tsm sites

- export—エクスポートする .zip ファイルの生成場所:

  ```
  /var/opt/tableau/tableau_
  server/data/tabsvc/files/siteexports
  ```

- import—インポートプロセス中に Tableau Server がファイルを検索する場所:

  ```
  /var/opt/tableau/tableau_
  server/data/tabsvc/files/siteimports
  ```

現在のファイルの場所を取得する

特定のコマンドの現在のファイルの場所を表示するには、`tsm configuration get` を使用します。

- tsm maintenance コマンド:
  - backup、restore、send-logs:

    ```
    tsm configuration get -k basefilepath.backuprestore
    ```

  - ziplogs:

    ```
    tsm configuration get -k basefilepath.log_archive
    ```

- tsm sites コマンド:
  - export

    ```
    tsm configuration get -k basefilepath.site_export.exports
    ```

  - import

    ```
    tsm configuration get -k basefilepath.site_import.exports
    ```

現在のファイルの場所を変更する

tsm configuration set コマンドを使用して想定するファイルの場所を変更し、basefilepath 変数を更新できます。具体的なベース ファイル パスの詳細については、tsm configuration set のオプションを参照してください。

basefilepath 変数を変更しても、既存のファイルは元のディレクトリから新しいディレクトリに移動しません。既存の backup、restore、log ファイルまたは site export/import ファイルを指定した新しいディレクトリに配置するには、それらのファイルを手動で移動する必要があります。新しい場所の作成と適切なパーミッションの設定を行って、配置されるファイルおよびこれらのファイルが含まれるディレクトリ構造に対する tsm のアクセスを許可する必要があります。パーミッションと tsm の詳細については、TSM におけるファイルとパーミッションを参照してください。バックアップ / 復元のベース ファイル パスを変更する場合は、tsm maintenance validate-backup-basefilepath コマンド (バージョン 2022.1 以降で使用可能) を実行して、パーミッションが正しく設定されていることを確認する必要があります。

tsm maintenance backup コマンドを実行すると、データディレクトリ内の一時的な場所でバックアップ ファイルが構成された後、 basefilepath.backuprestore 変数で指定された場所に保存されます。ベース ファイル パスを変更しても、tsm maintenance backup コマンドがバックアップ ファイルを構成する場所には影響しません。

- tsm maintenance コマンド:
    - backup、restore、または send-logs ディレクトリを変更するには、次のコマンドを実行します。

      ```
      tsm configuration set -k basefilepath.backuprestore -v
      "/new/directory/path"
      ```

    - ziplogs ディレクトリを変更するには、次のコマンドを実行します。

      ```
      tsm configuration set -k basefilepath.log_archive -v
      "/new/directory/path"
      ```

- tsm sites コマンド:

- sites export ディレクトリを変更するには、次のコマンドを実行します。

  ```
  tsm configuration set -k basefilepath.site_export.exports -
  v "/new/directory/path"
  ```

- sites import ディレクトリを変更するには、次のコマンドを実行します。

  ```
  tsm configuration set -k basefilepath.site_import.exports -
  v "/new/directory/path"
  ```

ファイルの既定の場所を変更した後に、次の操作を実行する必要があります。

1. 保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

2. Tableau Server を停止します。

   ```
   tsm stop
   ```

3. TSM コントローラーを再起動します (*tableau* システム アカウントとして)。

   ```
   sudo su -l tableau -c "systemctl --user restart
   tabadmincontroller_0.service"
   ```

4. コントローラーが再起動するまで数分間待機します。次のコマンドを指定してコントローラーが再起動したことを確認できます。

```
tsm status -v
```

このコマンドを実行すると、Tableau Server 管理コントローラーにコントローラーが再起動したことを示す 'running' と表示されます。

5. Tableau Server を起動します。

```
tsm start
```

エンティティ定義とテンプレート

tsm settings import コマンドでエンティティ定義とテンプレートを使用して、さまざまな Tableau Server 設定を構成できます。

構成ファイルの例

この記事では、エンティティ gatewaySettings および identityStore を指定した完全な JSON 構成ファイルの例を提供します。さらに、構成キーによりゲートウェイ タイムアウトを 900 秒に設定します。

設定が必要なオプションによって表示される構成ファイルは異なります。

インストール時に、複数の .json 構成ファイルを設定できます。Tableau Server で各ファイルの値を設定するには、各構成ファイルごとに次のコマンドを 1 回ずつ実行します。

```
tsm settings import -f path-to-file.json
```

構成ファイルの設定後、tsm pending-changes apply を実行し、設定したすべての .json ファイルから変更を適用します。

```
{
   "configEntities": {
      "gatewaySettings": {
         "_type": "gatewaySettingsType",
```

```
            "port": 80,
            "publicHost": "localhost",
            "publicPort": 80
         },
         "identityStore": {
            "_type": "identityStoreType",
            "type": "local",
            "domain": "example.lan",
            "nickname": "EXAMPLE"
         }
      },
       "configKeys": {
          "gateway.timeout": "900"
       }
}
```

エンティティとキー

上記の例のとおり、configEntities とconfigKeys の 2 クラスの構成パラメーターがあります。

configEntities

アイデンティティストアやゲートウェイ構成など、特定のシナリオにマッピングするエンティティセットを通じて、特定のタイプの構成が行われます。`tsm settings import -f path-to-file.json` コマンドで configEntities のセットを渡すと、TSM が構成を検証します。渡された値が無効である場合、TSM がエラーを表示します。これにより、初期化または実行時に構成が失敗することなく、構成プロセス中に変更を加えることができます。

configEntities ブロックを .json ファイルに含めることでのみ、エンティティを設定できます。

> **重要**: configEntities で参照されているすべてのファイルがローカル コンピューター上に配置されている必要があります。UNC パスは指定しないでください。

configKeys

エンティティは、設定可能な構成値のほんの一部しかカバーしていません。.yml ファイルに格納されているパラメーターには数百個のキーが対応します。Tableau Server は、これらのパラメーターを使用して、すべてのサービスのすべての構成情報を格納します。

tsm configuration コマンドを使用して、個々のキーを設定することができます。しかし、展開時に上記の方法で、JSON ファイル内の他の構成シナリオと共にキーを設定する方が便利です。

configEntities とは異なり、configKeys は検証されません。

> **注:**tsm configuration set のオプションに記載されていないパラメーターを設定することはお勧めしません。

gatewaySettings エンティティ

Tableau Server コンピューターのゲートウェイの設定を構成する必要があります。

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

```
tsm settings import -f /path/to/file.json

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

ゲートウェイの設定

下記のテンプレートのゲートウェイ設定は Tableau Server の HTTP 設定を指定します。SSL/TLS の使用をお勧めします。Tableau Server は SSL/TLS にポート443 を使用するようにハードコード

されています。そのため、SSL を有効にしている場合は、gatewaySettings エンティティを更新する必要はありません。

構成テンプレート

このテンプレートを使用してゲートウェイの設定を構成します。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

```
{
  "configEntities": {
        "gatewaySettings": {
         "_type": "gatewaySettingsType",
          "port": 80,
          "sslRedirectEnabled": true,
          "publicHost": "localhost"
        }
       }
 }
```

構成ファイル リファレンス

次の表には、"gatewaySettings" エンティティセットに含めることができるすべてのオプションが示されています。

`_type`

必須。

値: `"gatewaySettingsType"`

変更しないでください。

`port`

HTTP ポートを指定します。既定は、ポート 80 です。

`sslRedirectEnabled`

オプション: `true` または `false`。

`publicHost`

http/s サービスにホスト名を指定します。

`trustedIPs`

Tableau Server と通信する信頼できる IP アドレスを指定します。信頼できる IP アドレスにはアップストリームのプロキシ サーバーと Tableau Server との信頼できる認証に使用されるサーバーが含まれます。Tableau Server のプロキシとロード バランサーの設定および Tableau Server への信頼できる IP アドレスまたはホスト名の追加を参照してください。

Tableau Server をクラスタ内で実行している場合は、クラスタ内のその他すべてのノードが、このエンティティが更新する対応する構成ファイルに自動的に含まれます。つまり、`trustedIPs` に新しい値を指定した場合、その値に他のノードの IP アドレスも含めなくてはなりません。

このオプションは、各 IP またはホストを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。たとえば: `["192.168.1.101","192.168.1.102","192.168.1.103"]` や `["webserv1","webserv2","webserv3"]` です。

`trustedHosts`

Tableau Server と通信する信頼できる IP アドレスを指定します。通常、この値にはアップストリーム プロキシ サーバーのリストが含まれます。trustedHosts の値はクライアント要求対象を特定するために使用されます。

Tableau Server をクラスタ内で実行している場合は、クラスタ内のその他すべてのノードが、このエンティティが更新する対応する構成ファイルに自動的に含まれます。つまり、`trustedIPs` に新しい値を指定した場合、その値に他のノードの IP アドレスも含めなくてはなりません。

このオプションは、各 IP またはホストを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。たとえば:

`["192.168.1.101","192.168.1.102","192.168.1.103"]` や `["webserv1","webserv2","webserv3"]` です。

identityStore エンティティ

Tableau Server にはユーザーやグループの情報を格納するためのアイデンティティストアが必要です。アイデンティティストアを始めて構成する前に認証およびアイデンティティストアのトピックを確認しておいてください。Tableau Server でアイデンティティストアをインストールした後は、サーバーを再インストールしない限り、これを変更することはできません。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

はじめる前に

次の情報を確認します。

- ローカルのアイデンティティストアを使用しない場合は、LDAP のいずれかのバージョンを使用することになります。この場合は、ディレクトリ/LDAP の管理者に問い合わせて、お使いの LDAP スキーマとバインド要件に応じて Tableau Server を構成してください。
- Tableau Server 構成は Active Directory 用に最適化されています。Active Directory にインストールしている場合、初期ノード設定の構成を使用してアイデンティティストアを構成することをお勧めします。
- LDAP バインドはユーザー認証とは無関係です。たとえば、LDAP ディレクトリに対してシンプルなバインドを使用して認証するように Tableau Server を構成し、その後、インストール後に Kerberos によりユーザーを認証するように Tableau Server を構成できます。
- 非セキュア接続でシンプル バインドの LDAP を接続しないでください。既定では、シンプル バインドの LDAP はデータをクリアテキストで送信します。LDAPS を使用して、シンプル バインドのトラフィックを暗号化します。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。
- Tableau Server サービスで LDAP バインドに対して Kerberos 認証を使用するには、以下のセクションで説明するとおり、GSSAPI バインドの keytab ファイルが必要です。「Keytab 要

件の理解」も参照してください。Kerberos では、Tableau Server のベース インストール時には GSSAPI バインドのみが必要です。サーバーのインストール後、Kerberos でユーザー認証を構成し、データ ソースに対する Kerberos 委任を構成することができます。

- このトピックでは、*LDAP* (ディレクトリサービスに接続するためのプロトコル) と *LDAP* サーバー(ディレクトリサービスの実装) との違いを明確化します。たとえば、`slapd` は LDAP サーバーであり、OpenLDAP プロジェクトの一部です。

- サーバーを初期化する前に LDAP 構成を検証します。「初期ノード設定の構成」を参照してください。

- JSON 構成ファイルを初期構成の一部としてのみインポートします。JSON 構成ファイルをインポートして Tableau Server を初期化した後に LDAP に変更を加える必要がある場合は、JSON ファイルをインポートし直さないでください。その代わり、ネイティブの tsm コマンドまたは `tsm configuration set` を使用して、個々のキーを変更します。「外部のアイデンティティストアの構成リファレンス」を参照してください。

構成テンプレート

このセクションの JSON テンプレートは、さまざまなアイデンティティストア シナリオで Tableau Server を構成するために使用されます。ローカルアイデンティティストアを構成する場合を除き、LDAP 環境に固有の構成ファイル テンプレートを選択して編集する必要があります。

Tableau のアイデンティティストア構成 ツールを使用して LDAP JSON 構成ファイルを生成することを検討してください。ツール自体は Tableau ではサポートされていません。ただし、ファイルを手動で作成する代わりにこのツールで作成された JSON ファイルを使用しても、サポートされるサーバーのステータスは変わりません。

編集するアイデンティティストアの構成テンプレートを次から選択します。

- ローカル
- LDAP - Active Directory
- OpenLDAP - GSSAPI バインド
- OpenLDAP - シンプル バインド

構成ファイル、エンティティ、およびキーの詳細については、「構成ファイルの例」を参照してください。

ローカル

ユーザー認証に Active Directory または LDAP サーバーを導入していない組織の場合は、アイデンティティストアのタイプとしてローカルを構成してください。アイデンティティストアのタイプにローカルを選択する場合は、ユーザーの作成および管理には Tableau Server を使用します。

別の方法で Tableau Server をローカル アイデンティティストア用に構成するには、セットアップ GUI を実行し、インストール プロセス中に [ローカル] を選択します。「初期ノード設定の構成」を参照してください。

```
{
  "configEntities": {
    "identityStore": {
       "_type": "identityStoreType",
       "type": "local"
     }
   }
}
```

重要

以下の LDAP 構成テンプレートは例です。表示されたテンプレートでは組織に LDAP 接続が構成されません。ディレクトリの管理者と連携し、正常に実行された展開の LDAP テンプレート値を編集する必要があります。

また、configEntities で参照されているすべてのファイルがローカル コンピューター上に配置されている必要があります。UNC パスは指定しないでください。

LDAP - Active Directory

Tableau Server 構成は Active Directory 用に最適化されています。Active Directory にインストールしている場合は、初期ノード設定の構成を使用してアイデンティティストアを構成してください。

Active Directory への暗号化された接続が必要です。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

何らかの理由で、Tableau サービス マネージャーの Web インターフェイスを使用して Active Directory と通信するようにアイデンティティストアを構成できない場合は、この JSON テンプレートを使用して、Tableau Server が Active Directory に接続するように構成します。このテンプレートは GSSAPI (Kerberos) バインドを使用して、Active Directory に対して Tableau Server サービスを認証します。Tableau Server には Active Directory スキーマのサポートが含まれています。このため、"directoryServiceType" オプションを "activedirectory" に設定すると、"identityStoreSchemaType" オプションにスキーマ情報を指定する必要がありません。

Active Directory に Tableau Server for Linux をインストールしており、Tableau Server のインストール先の PC が既にドメインに結合している場合、PC には既に Kerberos 構成ファイルおよび keytab ファイルがあります。厳密にはこれらのファイルを GSSAPI バインドに使用することはできますが、その使用を推奨していません。代わりに、Active Directory の管理者に連絡して Tableau Server サービス専用のキータブを要求してください。

```
{
  "configEntities":{
    "identityStore": {
                "_type": "identityStoreType",
                "type": "activedirectory",
                "domain": "your-domain.lan",
                "nickname": "YOUR-DOMAIN-NICKNAME",
                "directoryServiceType": "activedirectory",
                "bind": "gssapi",
                "kerberosKeytab": "<path to local key tab file>",
                "kerberosConfig": "/etc/krb5.conf",
                "kerberosPrincipal": "your-principal@YOUR.DOMAIN"
                }
        }
}
```

GSSAPI を使用して Active Directory にバインドする方法が推奨されますが、シンプル バインドや LDAPS による接続もできます。シンプル バインドで接続するには、bind を simple に変更し、

3 つの Kerberos エンティティを削除して、port/sslPort、username とpassword オプションを追加します。次の例はシンプル バインドjson の Active Directory を示します。

```
{
  "configEntities":{
       "identityStore": {
               "_type": "identityStoreType",
               "type": "activedirectory",
               "domain": "your-domain.lan",
               "nickname": "YOUR-DOMAIN-NICKNAME",
               "directoryServiceType": "activedirectory",
               "hostname": "optional-ldap-server",
               "sslPort": "636",
               "bind": "simple",
               "username": "username",
               "password": "password"
               }
       }
}
```

OpenLDAP - GSSAPI バインド

以下のテンプレートを使用し、GSSAPI バインドで OpenLDAP を構成します。組織が Active Directory を実行している場合はこのテンプレートを使用しないでください。Active Directory にインストールしている場合は、上記のテンプレートである LDAP - Active Directory を使用してください。

GSSAPI (Kerberos) で OpenLDAP を使用するほとんどの組織は、認証資格情報を格納するために keytab ファイルを使用します。次の例では、認証資格情報に keytab ファイルを使用します。

ただし、username とpassword エンティティを通じて認証資格情報を提供できます。

また、keytab、ユーザー名、およびパスワードのペアをすべて指定することもできます。この場合は、Tableau Server は keytab の使用を試みますが、何らかの理由で認証に失敗した場合はフォールバックし、ユーザー名とパスワードの認証資格情報を使用します。

```
{
  "configEntities":{
              "identityStore": {
                      "_type": "identityStoreType",
                      "type": "activedirectory",
                      "domain": "your-domain.lan",
                      "nickname": "YOUR-DOMAIN-NICKNAME",
                      "directoryServiceType": "openldap",
                      "bind": "gssapi",
                      "kerberosKeytab": "<path to local key tab file>",
                      "kerberosConfig": "/etc/krb5.conf",
                      "kerberosPrincipal": "your-principal@YOUR.DOMAIN",
                      "identityStoreSchemaType": {
                         "userBaseFilter": "(objectClass=inetOrgPerson)",
                         "userUsername": "user",
                         "userDisplayName": "displayname",
                         "userEmail": "email",
                         "userCertificate": "certificate",
                         "userThumbnail": "thumbnail",
                         "userJpegPhoto": "photo",
                         "groupBaseFilter": "(objectClass=groupofNames)",
                         "groupName": "groupname",
                         "groupEmail": "groupemail",
                         "groupDescription": "groupdescription",
                         "member": "member",
                         "distinguishedNameAttribute": "",
                         "serverSideSorting": "",
                         "rangeRetrieval": "",
                         "userClassNames": ["inetOrgPerson","someClass2"],
                         "groupClassNames":
["groupOfUniqueNames1","groupOfUniqueNames2"]
                         }
                  }
         }
}
```

OpenLDAP - シンプル バインド

```
{
  "configEntities":{
               "identityStore": {
                       "_type": "identityStoreType",
                       "type": "activedirectory",
                       "domain": "my.root",
                       "nickname": "",
                       "hostname": "optional-ldap-server",
                       "port": "389",
                       "directoryServiceType": "openldap",
                       "bind": "simple",
                       "username": "cn=username,dc=your,dc=domain",
                       "password": "password",
                       "identityStoreSchemaType": {
                          "userBaseFilter": "(objectClass=inetOrgPerson)",
                          "userUsername": "user",
                          "userDisplayName": "displayname",
                          "userEmail": "email",
                          "userCertificate": "certificate",
                          "userThumbnail": "thumbnail",
                          "userJpegPhoto": "photo",
                          "groupBaseFilter": "(objectClass=groupofNames)",
                          "groupName": "groupname",
                          "groupEmail": "groupemail",
                          "groupDescription": "groupdescription",
                          "member": "member",
                          "distinguishedNameAttribute": "",
                          "serverSideSorting": "",
                          "rangeRetrieval": "",
                          "userClassNames": ["inetOrgPerson","someClass2"],
                          "groupClassNames":
["groupOfUniqueNames1","groupOfUniqueNames2"]
                          }
                }
  }
```

```
}
```

構成テンプレート リファレンス

共有されるアイデンティティ ストア オプション

`type`
: ユーザーの ID 情報を格納する場所。`local` または `activedirectory`。(LDAP サーバーに接続したい場合は、`activedirectory` を選択します。)

`domain`
: Tableau Server をインストールしたコンピューターのドメイン。

`nickname`
: ドメインのニックネーム。これは Windows 環境の NetBIOS 名としても参照されます。すべての LDAP エンティティに `nickname` オプションが必要です。組織でニックネームや NetBIOS 名が不要な場合は、ブランク キーを渡します (例: `"nickname": ""`)。

LDAP GSSAPI バインド オプション

`directoryservicetype`
: 接続するディレクトリ サービスのタイプ。`activedirectory` または `openldap`。

`kerberosConfig`
: ローカル コンピューター上の Kerberos 構成ファイルのパス。Active Directory にインストールしている場合、ドメインで結合したコンピューターにある可能性がある既存の Kerberos 構成ファイルや keytab ファイルを使用することは推奨されていません。「アイデンティティ ストア」を参照してください。

`kerberosKeytab`
: ローカル コンピューター上の Kerberos keytab ファイルのパス。Tableau Server サービス専用のキーで keytab ファイルを作成し、この keytab ファイルをコンピューター上の他のアプリケーションと共有しないことが推奨されます。たとえば、Linux では `/var/opt/tableau/keytab` ディレクトリに keytab ファイルを配置することができます。

`kerberosPrincipal`
: ホストマシン上の Tableau Server のサービスプリンシパル名。keytab には、このプリンシパルに対するパーミッションがなくてはなりません。`/etc/krb5.keytab` にある既存のシステム keytab を使用しないでください。代わりに、新しいサービスプリンシパル名の登録を推奨しています。指定された keytab のプリンシパルを表示するには、`klist -k` コマンドを実行します。「Keytab 要件の理解」を参照してください。

LDAP シンプル バインドのオプション

`directoryservicetype`
: 接続するディレクトリサービスのタイプ。`activedirectory` または `openldap`。

`hostname`
: LDAP サーバーのホスト名です。この値のホスト名または IP アドレスを入力できます。ここで指定するホストは、プライマリドメインのユーザー/グループ クエリのみに使用されます。ユーザー/グループ クエリが他のドメイン (プライマリドメインではない) にある場合、Tableau Server はこの値を使用しませんが、代わりに DNS に対してクエリを実行して適切なドメイン コントローラーを識別します。

`port`
: LDAP サーバーの非セキュア ポートを指定するためにこのオプションを使用します。プレーン テキストは通常 389 です。

`sslPort`
: このオプションを使用して LDAPS を有効にします。LDAP サーバーのセキュア ポートを指定します。通常 LDAPS はポート 636 です。LDAPS を使用するには、ホスト名 オプションも指定する必要があります。外部の LDAP アイデンティティ ストアへの暗号化チャネルの構成を参照してください。

`username`
: ディレクトリサービスとの接続に使用するユーザー名です。指定するアカウントには、ディレクトリサービスをクエリするパーミッションが必要です。Active Directory にはユーザー名 (`jsmith` など) を入力します。LDAP サーバーには、接続に使用するユーザーの識別名 (DN) を入力します。たとえば、「`cn=username,dc=your-local-domain,dc=lan`」と入力できます。

`password`

LDAP サーバーとの接続に使用するユーザー名のパスワードです。

LDAPS とサブドメイン

Active Directory で LDAPS を有効にしてサブドメインに接続している場合は、次の TSM コマンドを実行して、サブドメインの LDAPS ポート(TCP 636) を構成する必要があります。コマンドは、`subdomainFQDN:port` を指定する引数を受け取ります。

例: `tsm configuration set -k wgserver.domain.ldap.domain_custom_ports -v subdomain1.lan:636,subdomain2.lan:636,subdomain3.lan:636`

詳細については、「tsm configuration set のオプション」を参照してください。

共有 LDAP オプション。

以下のオプションは、汎用 LDAP、OpenLDAP、または Active Directory の実装に設定することができます。

`bind`

Tableau Server サービスから LDAP ディレクトリサービスへの通信を認証する方法。
GSSAPI (Kerberos) には「`gssapi`」と入力します。

`domain`

Active Directory 環境で、Tableau Server がインストールされているドメインを指定します (例: "example.lan")。
Active Directory 環境ではない LDAP の場合: この値に入力した文字列は、ユーザー管理ツールの [ドメイン] 列に表示されます。任意の文字列を入力できますが、キーを空白にすることはできません。

root

LDAP のみ。Active Directory 用に指定しないでください。
LDAP ルートでドメイン コンポーネントを使用しない場合、またはより複雑なルートを指定する必要がある場合には、LDAP ルートの設定が必要です。"o=my,u=root" の形式を使用します。たとえば、`example.lan` ドメインのルートは `"o=example,u=lan"` となります。

`membersRetrievalPageSize`

このオプションによって、LDAP クエリで返される結果の最大数が決まります。
たとえば、Tableau Server で 50,000 ユーザーを含む LDAP グループをインポートするというシナリオについて検討します。単一の操作でこのような多数のユーザーが含まれるインポートを試行することは、ベストプラクティスではありません。このオプションを 1,500 に設定すると、Tableau Server は最初の応答でまず 1,500 ユーザーをインポートします。それらのユーザーが処理されたら、Tableau Server で次の1,500 ユーザーを LDAP サーバーからリクエストする、という手順です。
このオプションは、お使いの LDAP サーバーの要件に対応する場合にのみ変更することをお勧めします。

`identityStoreSchemaType` オプション

LDAP サーバーとの LDAP 接続を構成する場合、LDAP サーバーに固有のスキーマ情報を `identityStoreSchemaType` オブジェクトに入力できます。

> **重要**: Active Directory (`"directoryServiceType": "activedirectory"`) に接続している場合、次のオプションは構成しないでください。

`userBaseFilter`

Tableau Server のユーザーが使用するように設定するフィルターです。たとえば、オブジェクトクラス属性および組織単位属性を指定する場合があります。

`userUsername`

LDAP サーバーのユーザー名に該当する属性です。

`userDisplayName`

LDAP サーバーのユーザーの表示名に該当する属性です。

`userEmail`

LDAP サーバーのユーザーのメール アドレスに該当する属性です。

`userCertificate`

LDAP サーバーのユーザーの証明書に該当する属性です。

`userThumbnail`
LDAP サーバーのユーザーのサムネイル画像に該当する属性です。

`userJpegPhoto`
LDAP サーバーのユーザープロフィール画像に該当する属性です。

`groupBaseFilter`
Tableau Server のユーザーグループが使用するように設定するフィルターです。たとえば、オブジェクトクラス属性および組織単位属性を指定する場合があります。

`groupName`
LDAP サーバーのグループ名に対応する属性です。

`groupEmail`
LDAP サーバーのグループの電子メールアドレスに対応する属性です。

`groupDescription`
LDAP サーバーのグループの説明に対応する属性です。

`member`
グループに一覧されているユーザーについて説明する属性です。

`distinguishedNameAttribute`
ユーザーの識別名を格納する属性です。この属性はオプションですが、LDAP クエリのパフォーマンスが大きく向上します。

`serverSideSorting`
クエリ結果をサーバー側で並べ替えるように LDAP サーバーを構成しているかどうかを示します。お使いの LDAP サーバーがサーバー側の並べ替えをサポートしている場合は、このオプションを `true` に設定します。LDAP サーバーがこれをサポートしているかどうかが不明な場合は、構成でエラーが発生する可能性があるため `false` と入力します。

`rangeRetrieval`

リクエストに対するクエリ結果の範囲を返すように LDAP サーバーを構成しているかどうかを示します。これは、多数のユーザーが含まれるグループが一度にではなく小分けにしてリクエストされることを意味します。範囲取得をサポートする LDAP サーバーは、クエリが大きい場合により良いパフォーマンスを発揮します。お使いの LDAP サーバーが範囲取得をサポートしている場合は、このオプションを `true` に設定します。LDAP サーバーがこれをサポートしているかどうかが不明な場合は、構成でエラーが発生する可能性があるため `false` と入力します。

`groupClassNames`

既定では、Tableau Server が「group」文字列を含む LDAP グループ オブジェクト クラスを探します。お使いの LDAP グループ オブジェクトが既定のクラス名に適合しない場合、この値を設定することでデフォルトを上書きします。コンマで区切って複数のクラス名を入力できます。このオプションは、各 クラスを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。例: `["basegroup","othergroup"]`。

`userClassNames`

既定では、Tableau Server が「user」および「inetOrgPerson」の文字列を含む LDAP ユーザー オブジェクト クラスを探します。お使いの LDAP ユーザー オブジェクトが既定のクラス名に適合しない場合、この値を設定することでデフォルトを上書きします。コンマで区切って複数のクラス名を入力できます。このオプションは、各 クラスを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。例: `["userclass1",userclass2”]`。

JSON ファイルのインポート

JSON ファイルの編集が完了したら、次のコマンドを使用して、ファイルを渡し、設定を適用します。

```
tsm settings import -f path-to-file.json

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプ

ションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

kerberosSettings エンティティ

Kerberos 認証を設定する前に、Kerberos の要件を確認してください。

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

```
tsm settings import -f /path/to/file.json
```

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

構成テンプレート

このテンプレートを使用して、Kerberos 設定を構成します。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

Kerberos 認証の初期構成を終了した後、tsm authentication kerberos <commands> sub-category を使用して追加の値を設定します。

```
{
  "configEntities": {
       "kerberosSettings": {
              "_type": "kerberosSettingsType",
```

```
                "enabled": "true",
                "keytabFile": "/path/to/keytab_file"
            }
        }
    }
```

構成ファイル リファレンス

"kerberosSettings" エンティティセットに含めることができるすべてのオプションは次のとおりです。

オプション
: 値

enabled
: オプション: true または false。

  Kerberos 認証を有効にします。

keytabFile
: 必須。

  有効な Kerberos keytab ファイルへのパス。

dBClasses
: グローバルな認証資格情報のデータベース クラスのコンマ区切りのリスト。Cloudera データソースに接続する場合に必要となることがあります。

mutualSSLSettings エンティティ

相互 SSL を構成する前に、Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成するを確認してください。

mutualSSLSettings エンティティは SSL および相互 SSL 構成の両方を組み合わせます。相互 SSL は外部 SSL が有効化され、適切に構成されていることを必要とします。

TSM エンティティは、JSON とキー値のペアを使用します。下記の構成ファイル テンプレートを使用して .json ファイルを作成します。使用する環境に適したキーの値を指定してから、次のコマンドを使用して .json ファイルを Tableau Server に渡します。

```
tsm settings import -f <path-to-file.json>

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

構成テンプレート

このテンプレートを使用して相互 SSL を構成します。

重要: エンティティ オプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

```
{
  "configEntities": {
    "mutualSSLSettings": {
        "_type": "mutualSSLSettingsType",
        "sslEnabled": true,
        "proxyLogin": false,
        "clientCertRequired": true,
        "caCertFile": "required",
        "keyFileName": "required",
        "keyPassphrase": "",
```

```
            "chainFile": "",
            "revocationFile": "",
            "redirect": false,
            "fallbackToPassword": true,
            "protocols": "",
            "cipherSuite": "",
            "forceHttpsForPublicEmbed": false
        }
    }
}
```

構成ファイル リファレンス

sslEnabled

SSL を有効化します。これは相互 SSL を有効化するための前提条件です。

clientCertRequired (MutualSSL)

true を設定して相互 SSL 認証を有効化します。false を設定して無効化します。

caCertFile (MutualSSL)

必須。

双方向 SSL のための CA 発行の証明書ファイルを指定します。ファイル パスは Tableau Server により読み取り可能でなくてはなりません。

certFileName

サーバー証明書の証明書チェーンを構成しているPEM 暗号化された CA 証明書の連結を含むファイルを指定します。

代わりに、便宜上CA 証明書がサーバー証明書に直接追加された場合は、参照されたファイルが caCertFile と同じでも問題ありません。

`keyFileName`

キーが証明書と組み合わされていない場合、この構成キーを使用してキーファイルをポイントします。RSA および DSA秘密キーの両方を所有している場合、両方を平行に構成できます (たとえば、DSA 暗号かぎの使用も許可する)。

`keyPassphrase`

オプション。証明書ファイルのパスフレーズ。入力するパスフレーズは保存中に暗号化されます。

注: パスフレーズに証明書のキーファイルを作成する場合、SAML に SSL 証明書キーを再利用することはできません。

`revocationFile`

SSL CA 証明書失効リスト (.crl) ファイルのファイル パスを指定します。

`Redirect`

既定は true です。Tableau Server が http リクエストを https リクエストとして適切なエンドポイントへリダイレクトすべきかどうかを指定します。

`clientCertMapping (MutualSSL)`

証明書からユーザー名を取得する方法を指定します。

許容される値は `ldap`、`upn`、`cn`

- ローカル認証を使用しているサーバーでは、規定の設定は `upn` (ユーザープリンシパル名) です。
- Tableau Server 認証が Active Directory (AD) 向けに構成されている場合、既定値は `ldap`(ライトウェイトディレクトリアクセスプロトコル) です。これは、サーバーに AD に行ってユーザーを検証するよう伝え、証明書内の名前は無視します。

どちらの認証タイプにも `cn` を設定して、証明書のサブジェクト DN の CN を使用することができます。

詳細については、相互認証中にクライアント証明書をユーザーにマッピングするを参照してください。

`fallbackToPassword (MutualSSL)`

true を設定すると、相互 SSL 認証が失敗した場合に、ユーザー名とパスワードを通して Tableau Server へサインインできるオプションをユーザーに付与します。false を設定すると、このフォールバック オプションが無効化されます。

`protocols`

許可したい、または禁止したいトランスポート レイヤー セキュリティ(TLS) プロトコル バージョンをリストします。

既定値: `"all -SSLv2 -SSLv3"`

ただし、次の設定の使用を推奨します。

`"all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1"`

詳細については、tsm security external-ssl enableを参照してください。一般的な情報については、Apach のオンライン ドキュメントを参照してください。

`cipherSuite`

SSL に許可または禁止する暗号鍵をリストします。

既定値:
`"HIGH:MEDIUM:!aNULL:!MD5:!RC4:!3DES:!CAMELLIA:!IDEA:!SEED"`

暗号リストの書式設定については、「OpenSSL」の「ciphers」ページを参照してください。このオプションを変更する場合は注意が必要です。既定値では、適切にセキュリティ保護されていないと見なされた暗号は許可されません。

`proxyLogin`

既定値: false。Tableau Server がサインインだけで SSL にプロキシを使用することを示しま

す。サーバーがサインイン API のために Tableau Desktop にレポートするプロトコルを制御します。

`forceHTTPForPublicEmbed`

既定値は false です。埋め込みビューのコードに SSL の使用を強制します。

openIDSettings エンティティ

OpenID 認証を構成する前にOpenID Connect の使用要件を見直してください。

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

```
tsm settings import -f path-to-file.json
```

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

構成テンプレート

このテンプレートを使用して OpenID 設定を構成します。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

OIDC の初期構成が終了したら、tsm authentication open-id <commands> サブカテゴリを使用して追加の値を設定します。

```
{
        "configEntities": {
```

```
            "openIDSettings": {
                "_type": "openIDSettingsType",
                "enabled": true,
                "clientId": "required",
                "clientSecret": "required",
                "configURL": "required if staticFile value is not set",
                "staticFile": "required if configURL value is not set",
                "externalURL": "required"
                }
        }
}
```

構成ファイル リファレンス

"openIDSettings" エンティティセットに含めることができるすべてのオプションは次のとおりです。

_type

必須。

変更しないでください。

enabled

必須。

true に設定します。

clientId

必須。

IdP がアプリケーションに割り当てているプロバイダー クライアント ID を指定します。たとえば、"laakjwdlnaoiloadjkwha"。

clientSecret

必須。

プロバイダー クライアント シークレットを指定します。これは、Tableau が IdP からの応答の真偽を検証するために使用するトークンです。この値は秘密で、保護しておく必要があります。

たとえば、`"fwahfkjaw72123="`。

`configURL`

必須。

プロバイダーの構成 URL を指定します。構成 URL を指定しない場合は、このオプションを削除して `staticFile` のパスとファイル名を代わりに指定してください。

`staticFile`

必須。

静的な OIDC ディスカバリ JSON ドキュメントへのローカル パスを指定します。静的ファイルを指定しない場合は、このオプションを削除して `configURL` の URL を代わりに指定してください。

`externalURL`

必須。

サーバーの URL です。一般的には、サーバーのパブリック名 (`http://example.tableau.com` など) になります。

`connectionTimeout`

オプション。

接続のタイムアウト間隔を秒数で指定します。既定は `10` です。

`readTimeout`

オプション。

読み取りのタイムアウト間隔を秒数で指定します。既定は 30 です。

`ignoreDomain`

以下に当てはまる場合は、これを true に設定します。

- Tableau Server で電子メール アドレスをユーザー名として使用している
- IdP のユーザーを複数のドメイン名を使用してプロビジョニングしました
- IdP からの email 要求のドメイン名部分を無視する場合があります

先に進む前に、このオプションを true に設定した結果として使用されるユーザー名を見直します。ユーザー名の競合が発生する場合があります。ユーザー名が競合したとき、情報開示のリスクは高くなります。OpenID Connect の使用要件を参照してください。

`ignoreJWK`

お使いの IdP が JWK 検証をサポートしていない場合は、これを true に設定します。この場合、相互 TLS または別のネットワーク レイヤー セキュリティ プロトコルを使用して IdP との通信を認証することをお勧めします。既定は false です。

`customScope`

IdP のクエリに使用できるカスタム範囲のユーザー関連値を指定します。OpenID Connect の使用要件を参照してください。

`idClaim`

お使いの IdP が ID トークンのユーザーを一意に識別するときに sub クレームを使用しない場合は、この値を変更します。指定する IdP クレームには、単一かつ一意の文字列を含める必要があります。

`usernameClaim`

この値を、組織が Tableau Server に保存されているユーザー名の照合に使用する IdP クレームに変更します。

`clientAuthentication`

OpenID Connect のカスタム クライアント認証手法を指定します。

Salesforce IdP を使用するように Tableau Server を設定するには、この値を `client_secret_post` に設定します。

`iFramedIDPEnabled`

IdP の iFrame への表示を許可するには `true` に設定します。iFrame での表示を許可する場合は、IdP でクリックジャック保護を無効にする必要があります。

samlSettings エンティティ

この記事では、Tableau Server で`samlSettings` エンティティのキーと値を含む構成ファイルを使用してサーバー全体の SAML を構成するためのテンプレートとリファレンスを提供します。この情報はサーバー全体の SAML の構成 の SAML 構成の手順を補完します。

SAML 構成テンプレートを作成し、Tableau Server に適用するには、次の手順を完了します。

1. テンプレートと、それがどのように構築されているかについて説明する次の 2 つのセクションを確認します (テンプレートのカテゴリーと定義および samlSettings 構成テンプレート)。
2. テンプレートに示される JSON コードを新しいテキストファイルに貼り付けて、.json の拡張子を使って保存します。
3. SAML 構成エンティティリファレンス を必要がある個所に値を提供するために役立てます。
4. 環境に応じてオプションのキー/値のペアを追加します。たとえば、SAML 証明書キー ファイルにパスフレーズが必要な場合、tsm configuration set コマンドを使用して `wgserver.saml.key.passphrase` パラメーターでパスワードを指定する必要があります。
5. Tableau Server に構成ファイルを渡す。

テンプレートのカテゴリーと定義

テンプレートは各キー値にプレースホルダーを使用します。プレースホルダーは次のようにカテゴリー分けされます。

- **Required (必須)**: "required" 値の属性は、構成コマンドを実行する前に有効なデータで置き換える必要があります。有効な値については、構成ファイル リファレンスを参照します。
- **Hard-coded (ハードコード)**: 下線 (_) が前置されている属性名。たとえば、"_type"はハードコード値を保持します。このような値は変更しないでください。
- **既定値**: "required"以外の値に設定されている属性は既定値です。これらは必須の属性で、環境に応じて変更することができます。
- **Empty sets (空のセット)**: 空の ("") 値は、そのまま渡すことも、インストール時に値を指定することもできます。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

samlSettings 構成テンプレート

このコードをテキストファイルに貼り付けて、下記のリファレンスを使用してお使いの環境に合わせてカスタマイズします。

```
{
  "configEntities": {
    "samlSettings": {
      "_type": "samlSettingsType",
      "enabled": true,
      "returnUrl": "required",
      "entityId": "required",
      "certFile": "required",
      "keyFile": "required",
      "idpMetadataFile": "required",
```

```
      "idpDomainAttribute": "",
      "idpUsernameAttribute": "required"
    }
  }
}
```

SAML 構成エンティティリファレンス

次のリストは、"samlSettings" エンティティセットに含むことができる、すべてのオプションをまとめたものです。

`idpMetadataFile`

必須。IdP で生成された XML ファイルのパスとファイル名。XML メタデータにはユーザー名属性 (アサーション) が含まれている必要があります。

サーバー全体の SAML の構成に説明されている手順を完了した場合は、ここに入力する値は次のようになります。

`/var/opt/tableau/tableau_server/data/saml/<metadata-file.xml>`

`enabled`

true | false

必須。SAML 認証が有効かどうかを示します。他の必要な SAML 構成オプションを設定するまで、このオプションを `true` に設定しないでください。

`returnURL`

通常は外部 URL であり、ユーザーが Tableau Server にアクセスするためにブラウザーに入力する、`https://tableau_server.example.com` など URL です。IdP を構成するときに、この値を使用して ACS URL 属性を作成します。

`entityId`

必須。サービスプロバイダーのエンティティID の値。ここでは Tableau Server がサービスプロバイダーです。

IdP に対する Tableau サーバの構成を識別します。`returnURL` オプションと同じ値を入力することを推奨します。

`idpUsernameAttribute`

必須。IdP メタデータで、ユーザー名値の指定に使用する属性を検索し、属性の名前を入力します。既定は `username` です。

`certFile`

必須。SAML 用の x509 証明書 (.crt) ファイルの場所およびファイル名を入力します。例:

```
/var/opt/tableau/tableau_server/data/saml/<file.crt>
```

詳細については、SAML 要件およびサーバー全体の SAML の構成を参照してください。

`keyFile`

必須。証明書ファイルに付属している、秘密鍵のファイル (.key ファイル) の場所を指定します。例:

```
/var/opt/tableau/tableau_server/data/saml/<file.key>
```

> **注**: パスフレーズを必要とする RSA PKCS#8 キーを使用している場合は、configKey エンティティ(構成ファイルの例を参照) を使用してパスフレーズを設定するか、tsm configuration set を使用してパスフレーズを設定する必要があります。これらのメソッドを使用するパスフレーズのキーは `wgserver.saml.key.passphrase` です。値は null 以外の文字列である必要があります。

`idpDomainAttribute`

LDAP または Active Directory を使用する組織の場合、Tableau Server がドメイン名を決定するために参照する SAML 属性をこの値で指定します。たとえば、IdP が `domain` 属性でドメイン名を指定する場合、この値に `domain` を指定します。**注**: 複数のドメインからユーザーがサインインしている組織の場合、この値は必須です。

このキーの値を指定しない場合、使用される値は Tableau Server の ID ストアの設定によって変わります。

- ローカル ID ストアの場合、`idpDomainAttribute` の値は無視されます。
- Active Directory または LDAP の ID ストアの場合、Tableau では構成設定 `wgserver.domain.default` から FQDN を使用します。

  `wgserver.domain.default` の値を取得するには、次のコマンドを実行できます。

  ```
  tsm configuration get --key wgserver.domain.default
  ```

`desktopNoSAML`

true | false

任意の項目です。Tableau Desktop からユーザーがサインインした場合に SAML 認証の使用を許可します。

既定ではこれが設定されていないため、false に設定すると効果的な動作になります。Tableau クライアントアプリケーションからのシングル サインオンが IdP と連携しない場合、これを true に設定して Tableau Desktop を通じた SAML 認証を無効にすることができます。

`appNoSAML`

true | false

オプション。Tableau Mobile アプリの古いバージョンからのサインインに SAML の使用を許可します。Tableau Mobile アプリのバージョン 19.225.1731 以上を実行しているデバイスでは、このオプションが無視されます。Tableau Mobile アプリのバージョン 19.225.1731 以上

を実行しているデバイスを無効にするには、Tableau Server でクライアント ログイン オプションとしての SAML を無効にします。

`logoutEnabled`

true | false

任意の項目です。SAML でログオンしたユーザーに対して、シングル ログアウトを有効にします。既定では `true` です。

IdP 構成メタデータに、POST バインディングを持つシングル ログアウトのエンドポイントが含まれている必要があります。

この設定は、サーバー全体の SAML に適用されます。

`false` に設定すると、Tableau Server はシングル ログアウトを試行しません。

`logoutUrl`

オプション。ユーザーがサーバーからサインアウトした後のリダイレクト先 URL を入力します。このオプションを設定するには、`logoutEnabled` を `true` に設定する必要があります。

既定では Tableau Server のサインイン ページです。絶対 URL または相対 URL を指定できます。

`maxAuthenticationAge`

オプション。IdP でのユーザーの認証と AuthNResponse メッセージのプロセスとの間で許容される最大秒数を指定します。既定値は -1 です。これは、maxAuthenticationAge が設定されていないか、既定で無視されることを意味します。2022 年 2 月以前の既定値は 7200 (2 時間) でした。

セッションの長さを最適化するには、IdP で設定されているのと同じタイムアウト値を使用します。

`maxAssertionTime`

オプション。作成からSAML アサーションが利用可能な最大秒数を指定します。既定値は 3000 (50 分) です。

`sha256Enabled`

true | false

任意の項目です。メッセージを IdP に送信するときに Tableau Server が使用する署名の種類。`true` に設定すると、Tableau Server は SHA 256 署名アルゴリズムでメッセージに署名します。`false` に設定すると、Tableau Server は SHA 1 でメッセージに署名します。既定は `true` です。

このオプションは、次のメッセージに Tableau Server が署名する署名アルゴリズムを設定します。

- AuthnRequest メッセージ (`signRequests`が有効になっている場合)
- LogoutRequest メッセージ (`logoutEnabled`が有効になっている場合)

`signRequests`

true | false

任意の項目です。IdP に送信する AuthnRequests に Tableau サーバーが署名するかどうかを指定します。署名付き要求は、すべての IdP に必ずしも必要ではありません。SAML を構成する際に最も安全なオプションを確保するため、要求に署名することをお勧めします。IdP が署名付き要求を受け入れるかどうかを確認するには、IdP メタデータを調べます。`wantAuthnRequestsSigned` が `true` に設定されている場合、IdP は署名付き要求を受け入れます。

既定値は `true` です。署名付き要求を無効化するには、このオプションを `false` に設定します。

`acceptableAuthnContexts`

オプション。SAML 属性 `AuthNContextClassRef` を設定します。このオプション属性は、IdP が開始するフローで特定の認証の "コンテキスト" の検証を強制します。この属性

の値をコンマ区切りのセットで設定します。この属性が設定されると、Tableau Server は、リストされている値の少なくとも 1 つが SAML 応答に含まれていることを検証します。SAML 応答に構成済みの値のいずれかが含まれていない場合、ユーザーが IdP によって正常に認証された場合でも、認証は拒否されます。

このオプションを空白のままにすると、デフォルトの動作になります。正常に認証された SAML 応答を受け取ると、Tableau Server のセッションへの権限がユーザーに付与されます。

`iFramedIdpEnabled`

true | false

任意の項目です。規定値は `false` であり、埋め込みビュー上でユーザーがサインインボタンを選択したときに、IdP のサインイン フォームがポップアップ ウィンドウで開きます。

true に設定し、既にサインインしているサーバー SAML ユーザーが埋め込みビューのあるウェブ ページに移動すると、ユーザーはビューを見るためにサインインする必要がありません。

IdP が iframe 内のサインインをサポートする場合にのみ、これを true に設定できます。この iframe オプションはポップアップを使用するよりも安全性が低くなるため、すべての IdP でサポートされているわけではありません。IdP のサインイン ページでクリックジャック攻撃防止機能を使用している場合 (ほとんどの場合で使用しています)、サインイン ページが iframe に表示されず、ユーザーはサインインできません。

お使いの IdP が iframe 経由でのサインインをサポートしていない場合、明示的に有効化する必要が生じることがあります。ただし、このオプションを使用できる場合でも、Tableau Server の SAML に対するクリックジャック攻撃防止機能が無効になるため、セキュリティ リスクが生じるおそれがあります。

Tableau Server に構成ファイルを渡す

構成テンプレートに含める各エンティティに適切な値を提供した後、次のコマンドを使って Tableau Server に .json ファイルを渡し、設定を適用します。

```
tsm settings import -f path-to-file.json

tsm pending-changes apply
```

関連項目

初回 SAML 構成を完了した後に、tsm authentication mutual-ssl <commands> を使用して追加の値を設定します。

SAML の構成のためのコマンド ライン リファレンスについては、tsm authentication saml <commands> を参照してください。

sapHanaSettings エンティティ

sapHanaSettings エンティティを使用して SAP HANA の SAML 委任を構成します。

SAP HANA SSO の構成を見直してから先に進んでください。

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

```
tsm settings import -f path-to-file.json

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes apply を参照してください。

SAP HANA SAML の設定

以下のテンプレートの設定で、SAP HANA を使用した SAML 委任シナリオにおける Tableau Server の設定が指定されます。

構成テンプレート

このテンプレートを使用して、信頼できる認証の設定を構成します。

configEntities で参照されているすべてのファイルがローカル コンピューター上に配置されている必要があります。UNC パスは指定しないでください。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

```
{
  "configEntities": {
        "sapHanaSettings": {
        "_type": "sapHanaSettingsType",
        "enabled": "true",
        "usernameFormat": "username",
        "usernameCase": "preserve",
        "certFile": "path-to-cert_file",
        "keyFile": "path-to-key_file"
         }
        }
}
```

構成ファイル リファレンス

次の表には、"gatewaySettings" エンティティセットに含めることができるすべてのオプションが示されています。

`enabled`

必須。

値: `true` または `false`

`usernameFormat`

値: `username`、`domain_and_username`、または `email`

ユーザー名の認証資格情報の形式を指定します。

`usernameCase`

値: `lower`、`upper`、または `preserve`

ユーザーの入力名に使用する大文字と小文字の区別を指定します。

`certFile`

ローカル コンピューター上の証明書ファイルのファイル パスと名前を指定します。

たとえば、`"/var/opt/tableau/tableau_server/data/saml/saml_cert.crt"`。

`keyFile`

ローカル コンピューター上の証明書キーのファイル パスと名前を指定します。

たとえば、`"/var/opt/tableau/tableau_server/data/saml/saml_key.der"`。

shareProductUsageDataSettings エンティティ

このエンティティを構成する前に、製品利用状況データを参照して、Tableau が収集する行動データおよび利用状況データについての理解を深めてください(機密データベース値は含まれていません)。

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

```
tsm settings import -f /path/to/file.json

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する

動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

構成テンプレート

このテンプレートを使用して、製品利用状況データの設定を構成します。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

```
{
  "configEntities": {
       "shareProductUsageDataSettings": {
               "_type": "shareProductUsageDataSettingsType",
               "enabled": "true"
        }
     }
 }
```

構成ファイル リファレンス

"shareProductUsageDataSettings" エンティティセットに含めることができるすべてのオプションは次のとおりです。

`_type`

必須の値: `shareProductUsageDataSettingsType`

`enabled`

オプション: `true` または `false`

既定では、`true` を使用してサーバーからの製品利用状況データを Tableau と共有します。

trustedAuthenticationSettings エンティティ

信頼できる認証を構成する前に、信頼できる認証を確認してください。

下記の構成ファイル テンプレートを使用して json ファイルを作成してください。オプションに適切な値を指定したら、次のコマンドを実行して json ファイルを渡し、設定を適用します。

```
tsm settings import -f /path/to/file.json
```

```
tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

構成テンプレート

このテンプレートを使用して、信頼できる認証の設定を構成します。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

信頼できる認証の初期構成が完了したら、tsm authentication trusted <commands> サブカテゴリーを使用して追加の値を設定します。

```
{
"configEntities": {
       "trustedAuthenticationSettings": {
              "_type": "trustedAuthenticationSettingsType",
              "trustedHosts": ["webserv1","webserv2","webserv3"]
              }
       }
}
```

構成ファイル リファレンス

"trustedAuthenticationSettings" エンティティセットに含めることができるすべてのオプションは次のとおりです。

`trustedHosts`

必須。

Tableau Server に信頼できるチケットを要求する Web サーバーの IP アドレスまたはホスト名。

このオプションは、各 IP またはホストを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。たとえば: `["192.168.1.101","192.168.1.102","192.168.1.103"]` や `["webserv1","webserv2","webserv3"]` です。

指定した値は以前の設定を上書きします。そのため、この値を構成する場合は、ホストの完全なリストを含める必要があります。

`tokenLength`

オプション。

値は、9 から 255 (包含) の間の任意の整数に設定できます。

信頼できるチケットごとの文字数を決定します。既定設定の 24 文字は、ランダム性のある 144 ビットで提供されます。このオプションは `useV2Tickets` が `true` に設定されている場合を除いて無視されます。これは推奨されるベストプラクティスではありません。

`logLevel`

オプション。

`all |debug |info |warn|error|fatal|off`

既定: `info`

信頼できるチケットの作成および引き換えに関連するプロセスのロギング レベルを指定します。ロギング レベルの変更を参照してください。

`timeoutInSeconds`

オプション。

既定: `180`

信頼できるチケットが作成されてから無効化されるまでの時間 (秒) を指定します。

`tryCount`

オプション。

整数。

既定: `10`

信頼できるチケット エントリの作成を試みる回数を指定します。

`use9DigitToken`

オプション。

`true | false`

既定: `false`

`true` に設定する場合、チケットは 9 桁の長さになり(バージョン 8.0 以前のように)、オプションの `tokenLength` は無視されます。このオプションは、レガシー コードを一時的にサポートすることを目的としています。

> **警告**: このオプションを `true` に設定すると、信頼できるチケット認証のセキュリティ強度に重大なマイナスの影響を与えます。

`useV2Tickets`

オプション。

`true | false`

既定: `false`

信頼できるチケット要求に対し、Tableau Server がレガシー URL 形式を返すかどうかを指定します。レガシー URL 形式には 24 文字の, Base64 エンコード文字列が含まれます。Tableau Server 10.3 以降、返される URL が更新され、Base64 エンコード UUID および 24 文字の安全なランダム文字列が含まれるようになりました。レガシー URL 形式を要求するカスタム コードで信頼できるチケットを展開した場合にのみ、このオプションを `true` に設定してください。代わりに、新しい URL 形式を受け入れるよう、カスタム コードを更新することをお勧めします。

web-data-connector-settings エンティティ

このエンティティは、Web データ コネクタ (WDC) の設定を管理するのに使用します。Tableau Server での WDC の使用の詳細については、Tableau Server での Web データ コネクタおよびtsm data-accessを参照してください。

下記の構成ファイル テンプレートを使用して .json ファイルを作成します。適切な値でオプションに入力した後、次のコマンドで .json ファイルを渡して設定を適用します。

```
tsm settings import -f /path/to/file.json

tsm pending-changes apply
```

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

Web データコネクタの設定

以下のテンプレートの Web データコネクタ(WDC) の設定では、WDC を有効にするか、WDC の更新を有効にするか、および 1 番目と2 番目のセーフリストを指定します。セーフリストは Tableau Server インストールでの使用が承認されている WDC の URL や、コネクタがリクエストを送受信できるドメインや URL を示します。

構成テンプレート

このテンプレートを使用して WDC の設定を構成します。

重要: エンティティオプションは、すべて大文字と小文字を区別します。

構成ファイル、エンティティ、およびキーの詳細については、構成ファイルの例を参照してください。

単一の WDC

```
{
  "configEntities": {
    "web-data-connector-settings": {
      "_type": "webDataConnectorSettingsType",
      "refreshEnabled": true,
      "whitelist": {
        "https://www.example.com:443/wdc/": {
          "secondaryWhitelist": [
          "https://www.example.com/.*",
          "https://www.coolapi.com/.*"
          ]
        }
      },
      "enabled": true
    }
  }
}
```

複数の WDC

```
{
  "configEntities": {
    "web-data-connector-settings": {
      "_type": "webDataConnectorSettingsType",
      "refreshEnabled": true,
      "whitelist": {
        "https://www.example.com:443/wdc/": {
          "secondaryWhitelist": [
          "https://www.example.com/.*"
          ]
        },
        "https://www.mysite.com:443/coolwdc/": {
          "secondaryWhitelist": [
          "https://www.mysite.com/.*",
          "https://www.coolapi.com/.*"
          ]
        }
      },
      "enabled": true
    }
  }
}
```

構成ファイル リファレンス

次の表には、web-data-connector-settings エンティティセットに含めることができるすべてのオプションが示されています。

`_type`

必須。

値: `webDataConnectorSettingsType`

変更しないでください。

`refreshEnabled`

WDC の更新を無効にするには、`false` に設定します。既定は `true` です。

`whitelist`

必須。

セーフリストおよび 2 番目のセーフリストの一致するセットを 1 つ以上含めることができます (WDC あたり 1 つのセット)。提供されている最初の URL はセーフリストであり、このリストでは WDC の URL およびポートを指定しますが、以下のように書式設定されています。

`<scheme>://<host>:<port>/<path>`

多くの WDC では、HTTP 接続に使用される既定のポート `<port>` 値は 443 ですが、Tableau Server または Tableau Cloud のデータソースの詳細からコネクタの値を確認することができます。

`secondaryWhitelist`

必須。

外部 JavaScript ライブラリ、REST API、ローカル ファイルなど、コネクタが要求を送受信できるドメインまたは URL を指定します。この 2 番目の許可リストにドメイン全体を追加するには、次の例に示すように、URL の末尾にワイルドカードの式 `.*` を使用します。

```
https://www.example.com/.*
```

`enabled`

WDC の使用を無効にするには、`false` に設定します。既定は `true` です。

tabcmd

注: tabcmd コマンドライン ユーティリティ バージョン 2.0 は、Tableau tabcmd で入手できます。この新バージョンでは、MacOS や Linux で tabcmd コマンドを実行し、個人アクセス トークン (PAT) を使って認証することができます。バージョン 2.0 は、Python ベースの Tableau Server クライアント (TSC) で使用可能なパブリック エンドポイントに基づいて構築されています。この最新バージョンでは、Tableau Server のサポートに制限があります。

Tableau では、Tableau Server サイトでのサイト管理タスクの自動化に使用できる tabcmd コマンドライン ユーティリティを提供しています。サイトの管理タスクを自動で実行できます (ユーザー、プロジェクト、グループの作成や削除など)

> **注:** Tableau バージョン 2024.1 より前では、tabcmd バージョン 1 は Viz のエクスポートには機能しません。

このユーティリティは Tableau Server に含まれ、サーバー ノードに自動でインストールされています。このユーティリティは他のコンピューターから実行することもできます (Tableau Server のインストールに含まれていないコンピューターも含む)。ただし、これには Tableau の Web サイトから tabcmd インストーラーをダウンロードする必要があります。詳細については、tabcmd のインストールを参照してください。

tabcmd のインストール

> **注:** これらの手順は、tabcmd 1.0 コマンドライン ユーティリティをインストールするためのものです。tabcmd 2.0 コマンドライン ユーティリティをインストールするには、Tableau tabcmd (新しいウィンドウ) に移動します。

Tableau Server または Tableau Cloud を新しいバージョンにアップグレードする際に、更新後のバージョンの tabcmd が必要な場合は、Tableau Web サイトの Tableau *Server* リリース ページからダウンロードできます。

Tableau Server の場合は、サーバー バージョンと一致するバージョンをダウンロードすることをお勧めします。Tableau Cloud の場合は、バージョンの不適合によって生じる問題を避けるために、常に

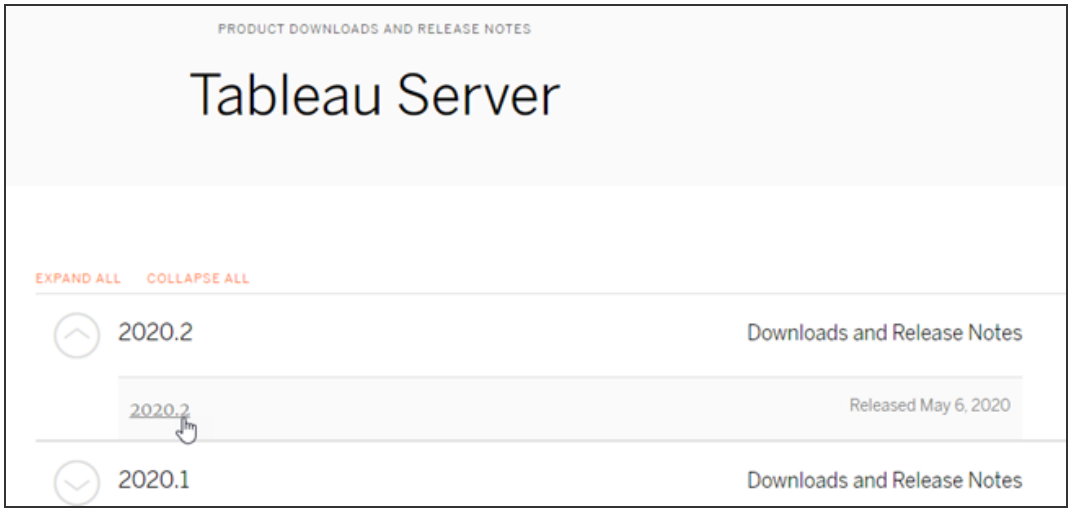
最新バージョンをダウンロードすることをお勧めします。どちらの場合でも、古いバージョンの tabcmd を使用すると、エラーや予期しない結果を引き起こすことがあります。

1. Web ブラウザーを開き、Tableau Server リリース ページに移動します。Tableau Online を使用している場合でも、このページに移動します。

2. 次の製品を使用している場合:

   - **Tableau Cloud** では、Tableau tabcmd 2.0 (新しいウィンドウ) を使用します。
   - **Tableau Server (Windows または Linux):** お使いのサーバー バージョンと一致するリリースを選択してください。

   どちらの場合でも、展開された情報にメンテナンス リリースが表示されていれば、最新のメンテナンス リリースまたはサーバー バージョンと一致するものを選択してください。



   これによりリリース ノート ページ (解決済みの問題) が表示され、セキュリティ改善と解決した問題について読むことができます。

3. 解決した問題の下にある **[ファイルのダウンロード]** セクションまでスクロールして、tabcmd コマンドを実行する予定のコンピューターと互換性のある tabcmd ダウンロード リンクを選択します。

残りの手順では、このコンピューターを "tabcmd コンピューター" と呼びます。

4. インストーラーを tabcmd コンピューターまたはそのコンピューターからアクセス可能な場所 (たとえば、マウントされたドライブ) に保存します。

5. tabcmd コンピューターのオペレーティング システムで適切なインストール手順を完了します。

- ## Windows

  既定では、tabcmd は `C:\Program Files\Tableau\Tableau Server\<version>\extras\Command Line Utility` にインストールされています。これはインストール中に変更でき、C:\ドライブ (`C:\tabcmd`) の `tabcmd` という名前のフォルダーに tabcmd をインストールすることを推奨します。これにより、検索と実行が容易になり、Windows PATH に tabcmd ディレクトリを追加すると、Windows オペレーティング システムに関するいくつかの制限に対応できます。

  > **注**:tabcmd 設定プログラムは、tabcmdディレクトリを Windows PATH 変数に追加しません。手動で追加するか、呼び出すたびに tabcmd への完全パスを含めることができます。

  tabcmd は、Windows に次の 2 つの方法でインストールできます。

- インストーラーをダブルクリックして、UI の手順に従います。

  a. 使用許諾契約書に同意します。

  b. 既定以外の場所にインストールする場合は、**[カスタマイズ]** をクリックし、tabcmd をインストールする場所を入力または参照します。

  c. **[インストール]** をクリックします。

  Windows Defender ファイアウォールまたはユーザーアカウントによるメッセージが表示される場合は、**[アクセスを許可]** をクリックします。

- インストーラーをコマンドプロンプトから実行します。

  a. tabcmd コンピューターで管理者としてコマンドプロンプトを開きます。

  b. Tabcmd インストーラーをコピーしたディレクトリに移動します。

  c. tabcmd のインストール:

  ```
  tableau-setup-tabcmd-tableau-<version_code>-
  x64.exe /quiet ACCEPTEULA=1
  ```

  既定以外の場所へのインストール:

  ```
  tableau-setup-tabcmd-tableau-<version_code>-
  x64.exe /quiet ACCEPTEULA=1
  INSTALLDIR="<path\to\install\directory>"
  ```

  例:

  ```
  tableau-setup-tabcmd-tableau-<version_code>-
  x64.exe /quiet ACCEPTEULA=1
  INSTALLDIR="C:\tabcmd"
  ```

  tabcmd インストーラーで使用できるコマンドラインオプションの完全な一覧については、/?を指定してインストーラーを実行します。tabcmd イ

ンストーラーのコマンド ライン オプションの詳細については、tabcmd 用のスイッチとプロパティのインストール (Windows)を参照してください。

tabcmd のセットアップ プログラムでは、tabcmd のインストールで問題が発生した場合に使用できるログが `C:\Users\<user>\AppData\Local\Temp` 内に作成されます。ログでは、`Tableau_Server_Command_Line_Utility_(<version_code>)_###############.log` という命名規則が使用されます。

- ## Linux

  注: Linux コンピューターで tabcmd を実行するには、Java 11 がインストールされている必要があります。RHEL ライクなシステムでは、これは tabcmd のインストール時に依存関係としてインストールされます。Ubuntu のようなシステムでは、Java 11 がまだインストールされていない場合は個別にインストールする必要があります。

  2022 年 7 月の時点で、Debian ディストリビューションはサポートされなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください。

  a. sudo アクセスのあるユーザーで tabcmd コンピューターにログインします。

  b. ダウンロードした `.rpm` または `.deb` パッケージをコピーしたディレクトリに移動します。

     - CentOS を含む RHEL ライクなディストリビューションで、次のコマンドを実行します。

       ```
       sudo yum install tableau-tabcmd-<version>.noarch.rpm
       ```

     - Ubuntu で次のコマンドを実行します。

```
sudo apt-get install ./tableau-tabcmd-<version>_
all.deb
```

Linux PC からtabcmd をアンインストールするには、実行中の各種 Linux に関するドキュメントを参照してください。

6. (オプション) tabcmd がインストールされている完全修飾場所をお使いのシステム パスに追加すると、その場所へ変更したり、各コマンドで場所を指定したりせずに tabcmd コマンドを実行できます。この手順は、オペレーティング システムの種類とバージョンによって異なります。詳細については、「PATH (変数)」を参照してください。

tabcmd の使用方法

tabcmd を使用する基本的な手順は次のとおりです。

1. 管理者としてコマンドプロンプトを開きます。

   > **注:** Windows では、PowerShell を使用して tabcmd コマンドを実行しないでください。PowerShell を使用すると、予期しない動作が発生する可能性があります。

2. Windows コンピューターで、最初のノード以外のコンピューターに tabcmd をインストールした場合は、tabcmd をインストールしたディレクトリに変更します。

   Linux コンピューターでは、インストール ディレクトリに変更する必要はありません。

3. tabcmd コマンドを実行します。

tabcmd を使用する場合、認証されたサーバー セッションを確立する必要があります。セッションは、Tableau Server や Tableau Cloud サイト、およびセッションを実行しているユーザーを識別します。最初にセッションを開始してコマンドを指定するか、セッションの開始とコマンドの実行を一度に行うことができます。

> **重要:** tabcmd を使用して複数のタスクを実行している場合、各タスクを同時に (並行して) ではなく、順番に (連続して) 実行する必要があります。

コマンド (login など) およびオプション (-s、-u など) は大文字と小文字を区別しませんが、ユーザーが指定する値 (User@Example.com など) は大文字と小文字を区別します。

例

次のコマンドは、Tableau Serverでの *tabserver.mycompany.com* という名前のセッションの開始を示します。

```
tabcmd login -s http://tabserver.mycompany.com -u admin -p
mypassword
```

次の例は、*Sales_Workbook* というワークブックを削除するコマンドを示します。

```
tabcmd delete "Sales_Workbook"
```

次に、1 つのコマンドで上記の手順すべてを達成する方法を示します。ここで login は必要ありません。

```
tabcmd delete "Sales_Workbook" -s http://tabserver.mycompany.com -u
admin -p mypassword
```

Tableau Server では複数のサイトを実行できます。ワークブックがマルチサイトサーバーの既定サイトにある場合、既定を指定する必要はありません。上記のコマンドで十分です。ただし、コマンドが既定以外のサイトのものに適用されている場合、そのサイト用にサイトID を指定する必要があります (loginを参照)。West Coast Sales サイト上にあるワークブックの場合の同じコマンドは、次のとおりです (サイトID は wsales)。

```
tabcmd delete "Sales_Workbook" -s http://tabserver.mycompany.com -t
wsales -u admin -p mypassword
```

オプションの -s、-t、-u、-p などは、tabcmd間のグローバル変数であるため、任意のコマンドと共に使用できます。

詳細については、tabcmd コマンドを参照してください。

ステータス メッセージおよびログ

コマンドが成功すると、tabcmdはステータス コード0 を返します。0 以外のステータス コードの詳しいエラー メッセージは **stderr** に出力されます。また、情報提供や進捗状況のメッセージも **stdout** に出力されます。

デバッグ、進捗状況、エラー メッセージを含む **tabcmd.log** という名前の詳細ログが `<home dir>/.tableau/tabcmd/` に書き込まれます。

> **重要:** Tableau サポートは、カスタム スクリプトでしか再現できない問題のトラブルシューティングを行うことはできません。カスタム スクリプトのサポートについては、Tableau コミュニティー フォーラムにアクセスしてください。

tabcmd コマンド

注: tabcmd コマンドライン ユーティリティ バージョン 2.0 は、Tableau tabcmd で入手できます。この新バージョンでは、MacOS や Lunux で tabcmd コマンドを実行し、個人アクセス トークン (PAT) を使って認証することができます。バージョン 2.0 は、Python ベースの Tableau Server クライアント (TSC) で使用可能なパブリック エンドポイントに基づいて構築されています。この最新バージョンでは、Tableau Server のサポートに制限があります。

> Tableau Server on Windows については、「tabcmd コマンド」を参照してください。

tabadmin コマンド ライン ツールで次のコマンドを使用することができます。

addusers (グループに対して)
createextracts
creategroup
createproject
createsite
createsiteusers
createusers
decryptextracts

delete *workbook-name* または delete *datasource-name*
deleteextracts
deletegroup
deleteproject
deletesite
deletesiteusers
deleteusers
editdomain
editsite
encryptextracts
export
get *url*
initialuser
listdomains
listsites
login
logout
publish
publishsamples
reencryptextracts
refreshextracts
removeusers
reset_openid_sub
runschedule
set
syncgroup
upgradethumbnails
validateidpmetadata
version

addusers *group-name*

指定したグループにユーザーを追加します。

例

```
tabcmd addusers "Development" --users "users.csv"
```

オプション

`--users`

特定の .csv ファイル内のユーザーを指定されたグループに追加します。ファイルは、1 行にユーザー名が 1 つの簡単なリストでなければなりません。ユーザー名は大文字と小文字を区別しません。ユーザーは、既に Tableau Server で作成されている必要があります。

詳細については、CSV インポートファイルのガイドラインを参照してください。

`--[no-]complete`

このオプションを complete に設定した場合、変更を正常に行うには、すべての行が有効でなければなりません。指定しない場合、--complete が使用されます。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証 トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は

export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

createextracts

パブリッシュ済みワークブックまたはデータソースの抽出の作成。

オプション

`-d, --datasource`

抽出作成対象のデータソースの名前。

`--embedded-datasources`

ターゲットワークブック内の埋め込みデータソース名をスペースで区切ったリスト。データソース名にスペースが含まれている場合は、二重引用符で囲みます。ワークブックの抽出を作成する場合にのみ使用できます。

`--encrypt`

暗号化された抽出の作成。

`--include-all`

ターゲットブック内のすべての埋め込みデータソースを含めます。ブックの抽出を作成する場合にのみ使用できます。

`--parent-project-path`

ターゲットリソースを含むプロジェクトの親であるプロジェクトのパス。プロジェクト名を --project で指定する必要があります。

`--project`

ターゲット リソースを含むプロジェクトの名前。--workbook または --datasource が指定されている場合にのみ必要です。指定されていない場合は、既定のプロジェクト 'Default' が使用されます。

`-u, -url`

URL に表示されるリソースの正規名です。

`-w, -workbook`

抽出作成先のブックの名前。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

creategroup *group-name*

グループを作成します。グループ作成後にユーザーを追加するには、`addusers` (ローカル グループの場合) を使用します。Tableau Server グループを作成して Active Directory グループと同期するには、`syncgroup` (Active Directory グループの場合) を使用します。

例

```
tabcmd creategroup "Development"
```

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

createproject *project-name*

プロジェクトを作成します。

例

```
tabcmd createproject -n "Quarterly_Reports" -d "Workbooks showing
quarterly sales reports."
```

オプション

`-n, --name`

作成するプロジェクトの名前を指定します。

`--parent-project-path`

`-n` オプションで指定されたネストされたプロジェクトで、親プロジェクトの名前を付けます。たとえば、"Main (主)" プロジェクトにある "Nested (ネスト)" という名前のプロジェクトを指定するには、構文 `--parent-project-path "Main" -n "Nested"` を使用します。

`-d, --description`

プロジェクトの説明を指定します。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID

では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

createsite *site-name*

サイトを作成します。

例

West Coast Sales という名前のサイトを作成します。のサイトIDWestCoastSales は自動的に作成され、サイトには保存容量の制限がなく、サイト管理者はユーザーの追加や削除を行うことができます。

```
tabcmd createsite "West Coast Sales"
```

wsalesのサイトID を持つWest Coast Sales という名前のサイトを作成します。

```
tabcmd createsite "West Coast Sales" -r "wsales"
```

サイト管理者がユーザーをサイトに追加できないようにします。

```
tabcmd createsite "West Coast Sales" --no-site-mode
```

保存容量をMB 単位で設定します。

```
tabcmd createsite "West Coast Sales" --storage-quota 100
```

オプション

`-r, --url`

サイトを指定するために URL で使用されます。サイト名とは異なります。

`--user-quota`

サイトに追加できる最大ユーザー数。

`--[no-]site-mode`

サイトに対してユーザーの追加または削除を行う権限をサイト管理者に許可/拒否します。

`--storage-quota`

MBで、サイト上に格納できるワークブック、抽出、およびデータソースの量。

`--extract-encryption-mode`

サイトに対する抽出の暗号化モードを**実行**、**有効**、または**無効**にできます。詳細については、保存中の抽出の暗号化を参照してください。

`--run-now-enabled`

ユーザーが抽出の更新、フロー、またはスケジュールを手動で実行することを許可または拒否します。ユーザーが手動でタスクを実行できるようにする場合は**真**、ユーザーが手動でタスクを実行できないようにする場合は**偽**を設定します。詳細については、サーバー設定 (全般とカスタマイズ)を参照してください。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

--

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

createsiteusers *filename.csv*

コンマ区切りの値 (CSV) で提供される情報に基づいて、サイトにユーザーを追加します。ユーザーがサーバー上にまだ作成されていない場合、このコマンドはそのユーザーを作成してからサイトに追加します。

CSV ファイルには 1 つまたは複数のユーザー名が含まれる必要があります。また、(各ユーザー用の) パスワード、フル ネーム、ライセンス タイプ、管理者レベル、パブリッシャー (はい/いいえ)、および電子メール アドレスを含めることができます。CSV ファイルの書式設定の詳細については、CSV インポート ファイルのガイドラインを参照してください。

管理者レベルおよびパブリッシャーのパーミッションを CSV ファイルに含める他の方法として、--role オプションを含め、CSV ファイルにリストされているユーザーを割り当てたいサイト ロールを指定することにより、アクセス レベル情報を渡すことができます。

デフォルトでは、ユーザーはログインしているサイトに追加されます。異なるサイトにユーザーを追加するには、広範囲の --site オプションを使用してそのサイトを指定します。(指定するサイト上でユーザーを作成するには、パーミッションが付与されている必要があります。)

サーバーに複数のサイトがある場合、createsiteusers コマンドを使用してサーバー (システム) 管理者を追加することはできません。代わりに createusers を使用してください。--role オプションの ServerAdministrator サイト ロールを指定する場合、コマンドによりエラーが返されま

す。CSV ファイルに管理者の値として System が含まれる場合、値は無視され、ユーザーに Unlicensed ライセンス タイプが割り当てられます。

サーバーに含まれるサイトが 1 つのみ (既定のサイト) の場合、ユーザーに対し管理者の値として system を指定できます。または、CSV ファイルに含まれるすべてのユーザーをサーバー管理者にする必要がある場合には --role オプションを使用して ServerAdministrator サイト ロールを割り当てることもできます。

既定では、このコマンドは同期操作 (続行する前にすべての操作が完了するのを待つ) を使用してユーザーを作成します。--no-wait オプションを使用して、非同期操作を指定できます。

ローカル認証

サーバーがローカル認証を使用するよう構成されている場合、CSV ファイル内の情報はユーザーを作成するために使用されます。

Active Directory 認証

サーバーが Active Directory 認証を使用するよう構成されている場合、ユーザー情報は Active Directory からインポートされ、CSV ファイル内のパスワードとわかりやすい名前の情報は無視されます。さらに、ユーザーが CSV ファイル内で指定されていても、対応するユーザーが Active Directory 内に存在しない場合、ユーザーは Tableau Server に追加されません。Active Directory ユーザーの場合、ユーザー名がドメイン全体で一意であることが保証されないため、ユーザー名の一部としてドメインを含める必要があります。これを domain\username または username@domain.com として指定できますが、domain\username 形式の使用をお勧めします。詳細については、外部 ID ストアを使用した展開におけるユーザー管理を参照してください。

例

```
tabcmd createsiteusers "users.csv" --role "Explorer"
```

オプション

--admin-type

非推奨です。代わりに、--role オプションを使用します。

`--auth-type`

すべてのユーザーの認証タイプ (`Local` または `SAML`) を、`.csv` ファイルで設定します。指定がない場合、既定では `Local` です。

注: SAML 認証を使用するには、Tableau Server の設定で、サイトをサイト固有の SAML 用に構成する必要があります。詳細については、「サイト固有の SAML の構成」を参照してください。

`--[no-]complete`

非推奨です。既定のエラー動作: 10 行の範囲内に 3 つ以上のエラーがある場合、コマンドは失敗します。

`--no-publisher`

非推奨です。代わりに、`--role` オプションを使用します。

`--nowait`

非同期ジョブが完了するのを待ちません。

`--publisher`

非推奨です。代わりに、`--role` オプションを使用します。

`--role`

`.csv` ファイルに含まれるすべてのユーザーに対して 1 つのサイトロールを指定します。**--role option** で複数のサイトロールを割り当てる必要がある場合、各サイトロールに対応する別個の CSV ファイルを作成します。

有効な値は、`ServerAdministrator`、`SiteAdministratorCreator`、`SiteAdministratorExplorer`、`SiteAdministrator`、`Creator`、

ExplorerCanPublish、Publisher、Explorer、Interactor, Viewer、および Unlicensed です。

既定値は、新しいユーザーの場合は Unlicensed で、既存のユーザーの場合は変更されません。また、ユーザーベースのサーバー インストールを行っている場合、および createsiteusers コマンドによって新しいユーザーが作成されたものの、ユーザーのライセンス数が既に上限に達している場合、ユーザーはライセンスなしとして追加されます。

> **注:** 複数サイトの Tableau Server で ServerAdministrator サイト ロールを --role オプションを使用して割り当てる必要がある場合、createsiteusers ではなくcreateusers コマンドを使用します。

--silent-progress

コマンドの進捗状況メッセージを表示しません。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

-h, --help

コマンドのヘルプを表示します。

-c, --use-certificate

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は

`export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

createusers *filename.csv*

コンマ区切りの値 (CSV) で提供される情報に基づいて、Tableau Server にユーザーを作成します。

CSV ファイルには 1 つまたは複数のユーザー名が含まれる必要があります。また、(各ユーザー用の) パスワード、フル ネーム、ライセンス タイプ、管理者レベル、パブリッシャー (はい/いいえ)、および電子メール アドレスを含めることができます。CSV ファイルの書式設定の詳細については、CSV インポート ファイルのガイドラインを参照してください。

管理者レベルおよびパブリッシャーのパーミッションを CSV ファイルに含める他の方法として、`--role` オプションを含め、CSV ファイルにリストされているユーザーを割り当てたいサイト ロールを指定することにより、アクセス レベル情報を渡すことができます。

サーバーに 1 つのサイト (既定のサイト) しかない場合、ユーザーが作成され、サイトに追加されます。サーバーに複数のサイトがある場合、ユーザーは作成されますが、サイトには追加されません。サイトにユーザーを追加するには、`createsiteusers` を使用します。

ユーザーベースのサーバー インストールを行っている場合、およびコマンドによって新しいユーザーが作成されたものの、ユーザーのライセンス数が既に上限に達している場合、そのユーザーはライセンスなしのユーザーとして追加されます。

ローカル認証

サーバーがローカル認証を使用するよう構成されている場合、CSV ファイル内の情報はユーザーを作成するために使用されます。

Active Directory 認証

サーバーが Active Directory 認証を使用するよう構成されている場合、ユーザー情報は Active Directory からインポートされ、CSV ファイル内のパスワードとわかりやすい名前の情報は無視されます。さらに、ユーザーが CSV ファイル内で指定されていても、対応するユーザーが Active Directory

内に存在しない場合、ユーザーは Tableau Server に追加されません。Active Directory ユーザーの場合、ユーザー名がドメイン全体で一意であることが保証されないため、ユーザー名の一部としてドメインを含める必要があります。これを domain\username または username@domain.com として指定できますが、domain\username 形式の使用をお勧めします。詳細については、外部 ID ストアを使用した展開におけるユーザー管理を参照してください。

例

```
tabcmd createusers "users.csv" --role "ServerAdministrator"
```

```
tabcmd createusers "users.csv"
```

オプション

`--admin-type`

非推奨です。代わりに、`--role` オプションを使用します。

`--[no-]complete`

非推奨です。既定のエラー動作: 10 行の範囲内に 3 つ以上のエラーがある場合、コマンドは失敗します。

`--no-publisher`

非推奨です。代わりに、`--role` オプションを使用します。

`--nowait`

非同期ジョブが完了するのを待ちません。

`--publisher`

非推奨です。代わりに、`--role` オプションを使用します。

`-r, --role`

.csv ファイルに含まれるすべてのユーザーに対して 1 つのサイトロールを指定します。--role option で複数のサイトロールを割り当てる必要がある場合、各サイトロールに対応する別個の CSV ファイルを作成します。

有効な値は、ServerAdministrator、SiteAdministratorCreator、SiteAdministratorExplorer、SiteAdministrator、Creator、ExplorerCanPublish、Publisher、Explorer、Interactor, Viewer、および Unlicensed です。

複数のサイトがあるサーバーでは、ユーザーはこのコマンドでサイトに割り当てられません。そのため、このコマンドによって割り当てできるサイトロールは ServerAdministrator と Unlicensed のみです。他のサイトロールを指定すると、コマンドによりUnlicensed ロールが割り当てられます。

単一サイトのサーバー上で、指定するロールを使用してユーザーが作成され、既定のサイトに追加されます。

ユーザーベースのサーバー インストールを行っている場合、およびコマンドによって新しいユーザーが作成されたものの、ユーザーのライセンス数が既に上限に達している場合、そのユーザーはライセンスなしのユーザーとして追加されます。

--silent-progress

コマンドの進捗状況メッセージを表示しません。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

-h, --help

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンドラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符

または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

decryptextracts

サイト上のすべての抽出を復号化します。サイトが指定されていない場合は、既定のサイトの抽出が復号化されます。詳細については、保存中の抽出の暗号化を参照してください。

抽出の数とサイズによっては、この操作によって大量のサーバーリソースが消費される場合があります。このコマンドは、通常の営業時間外に実行することを検討してください。

例

```
tabcmd decryptextracts "West Coast Sales"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

```
-h, --help
```

コマンドのヘルプを表示します。

```
-c, --use-certificate
```

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID

では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

delete *workbook-name* または delete *datasource-name*

指定のワークブックまたはデータソースをサーバーから削除します。

このコマンドは、パブリッシュ時のファイル名ではなく、サーバー上に表示されるワークブック名またはデータソース名を取得します。

例

```
tabcmd delete "Sales_Analysis"
```

オプション

`-r, --project`

削除するワークブックまたはデータソースを含むプロジェクトの名前。未指定の場合は "既定" プロジェクトが使用されます。

`--parent-project-path`

-r オプションで指定されたネストされたプロジェクトで、親プロジェクトの名前を付けます。たとえば、"Main (主)" プロジェクトにある "Nested (ネスト)" という名前のプロジェクトを指定するには、構文 `--parent-project-path "Main" -r "Nested"` を使用します。

`--workbook`

削除するワークブックの名前。

`--datasource`

削除するデータソースの名前。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

deleteextracts

パブリッシュ済みワークブックまたはデータ ソースの抽出の削除。

オプション

`-d, --datasource`

抽出削除対象のデータ ソースの名前。

`--embedded-datasources`

ターゲットワークブック内の埋め込みデータソース名をスペースで区切ったリスト。データソース名にスペースが含まれている場合は、二重引用符で囲みます。ブックの抽出を削除する場合にのみ使用できます。

`--encrypt`

暗号化された抽出の作成。

`--include-all`

ターゲットブック内のすべての埋め込みデータソースを含めます。

`--parent-project-path`

ターゲットリソースを含むプロジェクトの親であるプロジェクトのパス。プロジェクト名を --project で指定する必要があります。

`--project`

ターゲットリソースを含むプロジェクトの名前。--workbook または --datasource が指定されている場合にのみ必要です。指定されていない場合は、既定のプロジェクト 'Default' が使用されます。

`-u, -url`

URL に表示されるリソースの正規名です。

`-w, -workbook`

抽出削除先のブックの名前。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認

証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された .txt ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス no-

を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

deletegroup *group-name*

指定のグループをサーバーから削除します。

例

```
tabcmd deletegroup "Development"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

deleteproject *project-name*

指定のプロジェクトをサーバーから削除します。

tabcmd を使用して、プロジェクト階層で上位のプロジェクトのみを指定できます。親プロジェクト内のプロジェクトに対して実行するタスクを自動化するには、同等の Tableau REST API 呼び出しを使用します。

例

```
tabcmd deleteproject "Designs"
```

オプション

`--parent-project-path`

コマンドで指定されたネストされたプロジェクトで、親プロジェクトの名前を付けます。たとえば、"Main (主)" プロジェクトにある "Designs (デザイン)" という名前のプロジェクトを指定するには、構文 `--parent-project-path "Main" "Designs"` を使用します。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

deletesite *site-name*

指定のサイトをサーバーから削除します。

例

```
tabcmd deletesite "Development"
```

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

--

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

deletesiteusers *filename.csv*

ログインしているサイトからユーザーを削除します。削除するユーザーは、1 行に 1 つのユーザー名を指定した簡単なリストを含むファイルで指定されます。(ユーザー名以外の追加情報は必要ありません。)

既定では、サーバーに 1 つのサイトしかない場合、またはユーザーが 1 つのサイトにのみ属している場合、ユーザーはサーバーからも削除されます。Tableau Server Enterprise のインストールでは、サーバーに複数のサイトがある場合、**サーバー管理者**のサイトロールを割り当てられたユーザーは、サイトから削除されますが、サーバーからは削除されません。

ユーザーがコンテンツを所有している場合、ユーザーロールは**「ライセンスなし」**に変更されますが、ユーザーはサーバーやサイトから削除されません。コンテンツの所有者はそのユーザーのままです。ユーザーを完全に削除するには、コンテンツの所有者を変更してからユーザーの削除を再試行する必要があります。

ユーザーが Active Directory からインポートされた場合、ユーザーはサイトから削除されます。また、サーバーからも削除される可能性があります。ただし、Active Directory からは削除されません。

例

```
tabcmd deletesiteusers "users.csv"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

deleteusers *filename.csv*

指定したコンマ区切り値 (`.csv`) ファイルにリストされているユーザーを削除します。

`.csv` ファイルには 1 行に 1 つのユーザー名を指定した簡単なリストが含まれている必要があります。

例

```
tabcmd deleteusers "users.csv"
```

オプション

`--[no-]complete`

このオプションを `--complete` に設定した場合、変更を正常に行うには、すべての行が有効でなければなりません。指定しない場合、`--complete` が使用されます。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

editdomain

注: ベストプラクティスとして、ドメインを編集する前に Tableau Server をバックアップする必要があります。ドメインは Tableau リポジトリに保存されます。ドメインが誤って変更されると、管理者がサインインできなくなる可能性があります。

サーバー上の Active Directory ドメインのニックネームや完全修飾ドメイン名を変更します。ドメインの「ニックネーム」は、Windows NetBIOS ドメイン名です。

サーバーが使用しているドメインのニックネームは変更可能です。通常、サインインに使用したドメイン以外のドメインでは、フルドメイン名を変更できます。ただし、現在サインインしているユーザー名が現在のドメインと新しいドメインの両方に存在する場合、現在のドメインのフルネームを変更することができます。

Tableau Server が他のアクティブディレクトリドメインに接続できるようにするには TSM で `wgserver.domain.whitelist` オプションを設定して Tableau Server が接続するセカンダリドメインも指定する必要があります。セカンダリドメインおよび接続の設定の詳細については、wgserver.domain.whitelistを参照してください。

外部 ID ストアを使用した展開におけるユーザー管理を見直し、複数のドメイン、ドメイン名マッピング、ユーザー名が Tableau Server をどのように操作するかを理解してください。

ドメインのリストを参照するには、listdomains を使用します。

例

```
tabcmd editdomain --id 2 --nickname "new-nickname"
```

```
tabcmd editdomain --id 3 --name "new-name"
```

オプション

`--id`

変更するドメインの ID です。ドメイン ID のリストを取得するには、listdomains を使用します。

`--name`

ドメインの新しい名前です。

`--nickname`

ドメインの新しいニックネームです。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

editsite *site-name*

サイトまたはその Web フォルダー名を変更します。このコマンドを使用して、サイト管理者がユーザーを追加/削除したりユーザーによる特定のタスクの手動実行を防止したりすることを許可または拒否することもできます。サイト管理者にユーザー管理権限がある場合は、サイトに追加できるユーザー数を指定できます。

例

```
tabcmd editsite wc_sales --site-name "West Coast Sales"

tabcmd editsite wc_sales --site-id "wsales"
```

```
tabcmd editsite wsales --status ACTIVE

tabcmd editsite wsales --user-quota 50
```

オプション

`--site-name`

表示されたサイトの名前。

`--site-id`

サイトを一意に識別するために URL で使用されます。

`--user-quota`

サイトのメンバーになれる最大ユーザー数。

`--[no-]site-mode`

サイトへのユーザーの追加をサイト管理者に許可/拒否します。

`--status`

`ACTIVE` に設定してサイトを有効化するか、あるいは `SUSPENDED` に設定してサイトを一時停止します。

`--storage-quota`

MBで、サイト上に格納できるワークブック、抽出、およびデータソースの量。

`--extract-encryption-mode`

サイトに対する抽出の暗号化モードを**実行**、**有効**、または**無効**にできます。詳細については、保存中の抽出の暗号化を参照してください。抽出の数とサイズによっては、この操作に

よって大量のサーバー リソースが消費される場合があります。

`--run-now-enabled`

ユーザーが抽出の更新、フロー、またはスケジュールを手動で実行することを許可または拒否します。ユーザーが手動でタスクを実行できるようにする場合は**真**、ユーザーが手動でタスクを実行できないようにする場合は**偽**を設定します。詳細については、サーバー設定 (全般とカスタマイズ)を参照してください。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

encryptextracts

サイト上のすべての抽出を暗号化します。サイトが指定されていない場合は、既定のサイトの抽出が暗号化されます。詳細については、保存中の抽出の暗号化を参照してください。

抽出の数とサイズによっては、この操作によって大量のサーバー リソースが消費される場合があります。このコマンドは、通常の営業時間外に実行することを検討してください。

例

```
tabcmd encryptextracts "West Coast Sales"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

export

Tableau Server からビューまたはワークブックをエクスポートして、ファイルに保存します。このコマンドはビューに使用されるデータのみをエクスポートすることもできます。ビュー データは、集計レベル

でエクスポートされます。詳細レベルのデータをエクスポートするには、Tableau Server UI を使用する必要があります。詳細については、「「ビューとワークブックのダウンロード」を参照してください。

このコマンドを使用するときには、以下に注意してください。

- **パーミッション**: エクスポートするには、**[イメージのエクスポート]** パーミッションが必要です。既定では、このパーミッションはすべての役割で許可されているか継承されますが、ワークブックまたはビューごとにパーミッションを設定することもできます。
- **データのエクスポート**: ビューに使用されるデータのみをエクスポートするには、`--csv` オプションを使用します。ビューで使用されるサマリーデータを .csv ファイルにエクスポートします。
- **ビュー、ワークブック、またはエクスポートするデータの指定**:
  - URL の一部をエクスポート対象を特定するために使用します。具体的にはワークブックまたはビューの URL に現れる `"workbook/view"` 文字列を使用します。「わかりやすい名前」は使用せず、URL の末尾にあるセッション ID「`:iid=<n>`」を除外します。

    例えば、Tableau の *Superstore* ワークブックのサンプル ビュー *[オーダーの詳細]* には次のような URL があります。`<server_name>/#/views/Superstore/OrderDetails?:iid=2`

    [オーダーの詳細] ビューをエクスポートするには、文字列「`Superstore/OrderDetails`」を使用します。

    `Superstore/Order Details` や `Superstore/OrderDetails?:iid=2` は *使用しないでください*。
  - サーバーが複数のサイトを実行し、ビューまたはワークブックが "既定" 以外のサイトにある場合、`-t <site_id>` を使用します。
  - ワークブックをエクスポートするには、ワークブック内でビューを開いて、URL 文字列を取得し、使用するストリングにビューを含めます。

上の例では、*Superstore* ワークブックをエクスポートするために、文字列 `Superstore/OrderDetails` を使用します。

- ワークブックをエクスポートするには、[Tableau Desktop パブリッシュ] ダイアログ ボックスで **[シートをタブとして表示]** を選択して、それをパブリッシュしておく必要があります。

  注: 管理ビューを含む Tableau ワークブックはエクスポートできません。

- ダウンロードするデータをフィルターするには、次の形式を使用してパラメーター フィルターを追加します。

  ```
  ?<filter_name>=value
  ```

  または、パラメーターのフィルター処理で、そのパラメーターにメジャーまたはディメンションの名前と一致する表示名が含まれている場合は、次の形式を使用します。

  ```
  ?Parameters.<filter_name>=value
  ```

- **ファイルの保存形式:** 形式オプションは、エクスポートするファイルによって異なります。ワークブックは `--fullpdf` 引数を使用して PDF としてエクスポートします。ビューは PDF (`--pdf`) または PNG (`--png`) としてエクスポートできます。

- **保存するファイルの名前と場所** (省略可): 名前を指定しなければ、ビューまたはワークブックの名前から生成されます。場所を指定しなければ、ファイルは現在の作業ディレクトリに保存されます。それ以外の場合は、完全パスまたは現在の作業ディレクトリに対して相対的なパスを指定できます。

  > **注:** `.csv` や `.pdf` などのファイル名拡張子を含める必要があります。このコマンドは、指定したファイル名に拡張子を自動的に追加しません。

- **PDF エクスポートに含まれていないダッシュボード Web ページ オブジェクト:** ダッシュボードはオプションで Web ページ オブジェクトを含めることができます。Web ページ オブジェクトを

含むダッシュボードの PDF へエクスポートする場合、Web ページ オブジェクトは PDF に含まれません。

- **非 ASCII 文字と非標準の ASCII 文字、PDF のエクスポート:** 名前に ASCII 文字セット以外の文字、または非標準の ASCII 文字セットが含まれるビューまたはワークブックをエクスポートする場合は、その文字を URL エンコード (パーセント エンコード) する必要があります。

  たとえば、コマンドに都市「Zürich」が含まれる場合、URL を「Z%C3%BCrich」とエンコードする必要があります。

  ```
  tabcmd export "/Cities/Sheet1?locationCity=Z%C3%BCrich" -
  fullpdf
  ```

キャッシュをクリアしてリアルタイム データを使用

キャッシュからデータを取得する代わりに新しいデータ クエリを強制的に実行するために、オプションで URL パラメーターの `?:refresh=yes` を追加できます。独自のスクリプトで tabcmd を使用していて、`refresh` URL パラメーターを頻繁に使用している場合、パフォーマンスに悪影響を与える可能性があります。リアルタイム データが必要な場合は `refresh` のみを使用することをお勧めします (ワークブック全体の代わりに 1 つのダッシュボードなど)。

例

ビュー

```
tabcmd export "Q1Sales/Sales_Report" --csv -f "Weekly-Report.csv"

tabcmd export -t Sales "Sales/Sales_Analysis" --pdf -f "C:\Tableau_
Workbooks\Weekly-Reports.pdf"

tabcmd export "Finance/InvestmentGrowth" --png

tabcmd export "Finance/InvestmentGrowth?:refresh=yes" --png
```

ワークブック

```
tabcmd export "Q1Sales/Sales_Report" --fullpdf
```

```
tabcmd export "Sales/Sales_Analysis" --fullpdf --pagesize tabloid -
f "C:\Tableau_Workbooks\Weekly-Reports.pdf"
```

オプション

`-f, --filename`

指定したファイル名と拡張子でファイルが保存されます。

`--csv`

ビューのみ。ビューのデータ(サマリー データ) を `.csv` 形式でエクスポートします。

`--pdf`

ビューのみ。PDF としてエクスポートします。

`--png`

ビューのみ。イメージとして `.png` 形式でエクスポートします。

`--fullpdf`

ワークブックのみ。PDF としてエクスポートします。ワークブックは **[シートをタブとして表示]** を有効にしてパブリッシュする必要があります。

`--pagelayout`

エクスポートする PDF のページの向き (`landscape` または `portrait`) を設定します。指定しなければ、Tableau Desktopの設定が使用されます。

`--pagesize`

エクスポートする PDF のページサイズを次のいずれかに設定します。`unspecified`、`letter`、`legal`、`note folio`、`tabloid`、`ledger`、`statement`、`executive`、`a3`、`a4`、`a5`、`b4`、`b5`、または `quarto`。既定は `letter` です。

`--width`

幅をピクセル単位で設定します。既定値は 800 px です。

`--height`

高さをピクセル単位で設定します。既定値は 600 px です。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

get *url*

指定した (部分的) URL によって表される Tableau Server からリソースを取得します。結果はファイルとして返されます。

このコマンドを使用するときには、以下に注意してください。

- **パーミッション**: ファイルを取得するには、**[ダウンロード/名前を付けて Web に保存]** パーミッションが必要です。既定では、このパーミッションはすべての役割で許可されているか継承されますが、ワークブックまたはビューごとにパーミッションを設定することもできます。

- **取得ビューまたはワークブックの指定:** 取得するビューを指定するには "/views/<workbookname>/<viewname>.<extension>" 文字列を使用し、取得するワークブックを指定するには、"/workbooks/<workbookname>.<extension>" 文字列を使用します。<workbookname> と<viewname> を、ブラウザーでビューを開いたときに URL に表示されるワークブック名とビューに置き換え、**<extension>** を保存するファイル形式に置き換えます。URL の末尾にセッション ID (?:iid=<n>) を使用したり、「わかりやすい」名前をワークブックやビューに使用しないでください。

  たとえば、*"メトリクス サマリー"* というワークブックでビュー *"地域合計"* を開くと、URL は次のようになります。

  /views/MetricsSummary_1/RegionalTotals?:iid=1

  文字列 /views/MetricsSummary_1/RegionalTotals.<extension> を使用してビューを取得します。

  文字列 /workbooks/MetricsSummary_1.<extension> を使用してビューを取得sいます。

  ワークブックやビューを Tableau Server からダウンロードする場合、.twb または .twbx ファイルのコンテンツは、プレーン テキストで保存されます。ユーザーがファイルを開くと、データの意味的な手がかりとなるフィルター値を含むすべてのデータを読むことができます。

- **ファイル拡張子:** URL にはファイル拡張子を含める必要があります。拡張子によって返される結果が異なります。ビューは、PDF、PNG、または CSV (サマリー データのみ) 形式で返すことができます。Tableau ワークブックは、パブリッシュされたデータ ソースに接続するかライブ接続を使用する場合、TWB として返されます。データ抽出に接続する場合は TWBX として返されます。

> **注**: ビューを PDF または PNG ファイルにダウンロードし、.pdf または .png 拡張子を含む `--filename` パラメーターを含める場合、その URL に .pdf または .png 拡張子を含める必要はありません。

- **保存するファイルの名前と場所** (省略可): `--filename` に使用する名前に、ファイル拡張子を含める必要があります。名前とファイル拡張子を指定しなければ、両方とも URL の文字列から取得されます。場所を指定しなければ、ファイルは現在の作業ディレクトリに保存されます。それ以外の場合は、完全パスまたは現在の作業ディレクトリに対して相対的なパスを指定できます。
- **PNG サイズ** (オプション): 保存されているファイルが PNG の場合、ピクセル単位で URL のサイズを指定できます。

キャッシュをクリアしてリアルタイム データを使用

キャッシュからデータを取得する代わりに新しいデータクエリを強制的に実行するために、オプションで URL パラメーターの `?:refresh=yes` を追加できます。独自のスクリプトで tabcmd を使用していて、`refresh` パラメーターを頻繁に使用している場合、パフォーマンスに悪影響を与える可能性があります。リアルタイム データが必要な場合は `refresh` のみを使用することをお勧めします (ワークブック全体の代わりに 1 つのダッシュボードなど)。

例

ビュー

```
tabcmd get "/views/Sales_Analysis/Sales_Report.png" --filename
"Weekly-Report.png"

tabcmd get "/views/Finance/InvestmentGrowth.pdf" -f "Q1Growth.pdf"

tabcmd get "/views/Finance/InvestmentGrowth" -f "Q1Growth.pdf"

tabcmd get "/views/Finance/InvestmentGrowth.csv"
```

```
tabcmd get "/views/Finance/InvestmentGrowth.png?:size=640,480" -f
growth.png

tabcmd get "/views/Finance/InvestmentGrowth.png?:refresh=yes" -f
growth.png
```

ワークブック

```
tabcmd get "/workbooks/Sales_Analysis.twb" -f "C:\Tableau_
Workbooks\Weekly-Reports.twb"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

initialuser

初期の管理ユーザーが定義されていないサーバーに、初期サーバーの管理者ユーザーを作成します。これにより、サーバー管理者が作成されます。この操作では、TSM 管理者は作成されません。

単一引用符で値を囲みます。

> **注: tabcmd initialuser** コマンドは、Tableau Server への認証を要求しませんが、初期サーバー ノード上でコマンドを実行する必要があります。

注:

- **tabcmd initialuser** コマンドは、Tableau Server への認証を必要としませんが、初期サーバー ノード上でコマンドを実行する必要があります。
- ユーザー名のサフィックスが Tableau Server のプライマリドメインと一致しない限り、`username` 値に @ マークを含めることはできません。たとえば、Tableau Server がドメイン "myco.com" に接続する場合、"user@example.com@myco.com" というユーザー名は使用できません。

例

```
tabcmd initialuser --username 'admin' --password 'password' --server
http://localhost
```

```
tabcmd initialuser --username 'admin' --password 'password' --
friendly 'Tableau Admin' --server http://localhost
```

シェルでパスワードを要求するには、コマンドに `--password` パラメーターを含めません。例は次のとおりです。

```
tabcmd initialuser --username 'admin' --server http://localhost
```

オプション

`-f, --friendly`

表示名を持つ初期サーバー管理者ユーザーを作成します。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

listdomains

サーバーで使用中の Active Directory ドメインのリストをニックネームや ID と一緒に表示します。サーバーがローカルの認証を使用するように構成されている場合、コマンドによりドメイン名の `local` のみが返されます。

例

```
tabcmd listdomains
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

listsites

ログインしたユーザーが属しているサイトのリストを返します。

例

```
tabcmd listsites --username adam --password mypassword
```

オプション

`--get-extract-encryption-mode`

サイトに対する抽出の暗号化モードを**実行**、**有効**、または**無効**にできます。詳細については、保存中の抽出の暗号化を参照してください。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンドラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

login

Tableau Server ユーザーをログインします。

セッションを作成するために、`--server`、`--site`、`--username`、`--password`グローバル オプションが使用されます。

注: tabcmd login コマンドを使用する場合は、サーバーが SAML を使用するよう構成されている場合でも SAML シングル サインオン (SSO) を使用できません。ログインするには、サーバー上で作成されているユーザー名とパスワードを渡す必要があります。サインインしている Tableau Server ユーザーのパーミッションを取得します。詳細については、ユーザーのサイト ロールの設定およびパーミッションを参照してください。

セッションの作成で既に使用した情報を使用してログインする場合は、--password オプションだけを指定してください。cookie に保管されているサーバーとユーザー名が使用されます。

サーバーが 80 (既定) 以外のポートを使用している場合は、ポートを指定する必要があります。

サーバーで複数のサイトを実行し、デフォルトサイトでないサイトにログインしている場合にのみ、--site(-t) オプションが必要になります。パスワードを指定しない場合、パスワードを要求するプロンプトが表示されます。--no-prompt オプションを指定し、パスワードを指定しない場合、このコマンドは失敗します。

ログインした後、セッションはサーバーで有効期限が切れるか、logout コマンドが実行されるまで継続されます。

例

ユーザー jsmith をローカル マシンで稼働している Tableau Server にログインさせます。

```
tabcmd login -s http://localhost -u jsmith -p password
```

管理者を sales-server の Sales (セールス) サイトにログインさせます。

```
tabcmd login -s http://sales-server -t Sales -u administrator -p
password
```

```
tabcmd login -s http://sales-server:8000 -t Sales -u administrator -
p password
```

管理者は SSL を使用して sales-server の Sales サイトにログインしますが、サーバーの SSL 証明書は検証しません。

```
tabcmd login --no-certcheck -s https://sales-server -t Sales -u
administrator -p password
```

localhost の順行プロキシとポートを確立します。

```
tabcmd login --proxy myfwdproxyserver:8888 -s http://localhost -u
jsmith -p password
```

ユーザーjsmith に SSL を使用して、逆行プロキシにログインさせます。

```
tabcmd login -s https://myreverseproxy -u jsmith -p password
```

オプション

`-s, --server`

ネットワーク上の Tableau Server コンピューターからコマンドを実行する場合は、`http://localhost` を使用できます。そうでない場合は、`http://bigbox.myco.com` または `http://bigbox` のようにコンピューターの URL を指定します。

サーバーが SSL を使用している場合は、コンピューターの URL で `https://` を指定する必要がありますにあります。

Tableau Cloud の場合は、サイトがデプロイされているポッドを含む完全な URL を指定します。例: `https://prod-useast-b.online.tableau.com`。

`-t, --site`

サーバーに複数のサイトがあり、ドフォルトサイトでないサイトにログインしている場合、このオプションを含めます。

サイトID は、サイトを一意に識別するために URL で使用されます。たとえば、West Coast Sales という名前のサイトには、west-coast-salesというサイトID が設定されている可能性があります。

`-u, --username`

ユーザーがログインするユーザー名。Tableau Cloud の場合、ユーザー名はユーザーの電子メール アドレスです。

`-p, --password`

`--username` 用に指定されたユーザーのパスワード。パスワードを指定しない場合、パスワードを要求するプロンプトが表示されます。

`--password-file`

パスワードを、コマンド ラインではなく指定された `filename.txt` ファイルに保管し、セキュリティを向上させることができます。

`-x, --proxy`

tabcmd 要求に HTTP プロキシ サーバーとポート (Host:Port) を指定するために使用します。

`--no-prompt`

パスワードを要求しません。パスワードを指定しないと、`login` コマンドが失敗します。

`--no-proxy`

HTTP プロキシ サーバーを使用しません。

`--cookie`

ログオン時のセッション ID を保存します。後続コマンドは、ログインを必要としません。この値は、このコマンドで既定です。

`--no-cookie`

ログインが正常に完了した後はセッション ID 情報を保存しません。後続コマンドにはログインが必要です。

`--timeout SECONDS`

`login` コマンドの処理まで、サーバーが待機する秒数です。既定: 30 秒です。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

logout

サーバーからログアウトします。

例

```
tabcmd logout
```

filename.twb(x)、*filename.tds(x)*、または *filename.hyper* のパブリッシュ

指定したワークブック (.twb(x))、データ ソース (.tds(x))、または抽出 (.hyper) を Tableau Server にパブリッシュします。

ワークブックをパブリッシュする場合、デフォルトでは、ワークブック内のすべてのシートは、データベースのユーザー名またはパスワードなしでパブリッシュされます。

ワークブックやデータソースに最初に割り当てられるパーミッションは、ファイルがパブリッシュされるプロジェクトからコピーされます。パブリッシュされるリソースのパーミッションは、ファイルがパブリッシュされた後に変更できます。

ワークブックにユーザーフィルターが含まれている場合、いずれかのサムネイルオプションを指定する必要があります。

例

```
tabcmd publish "analysis.twbx" -n "Sales_Analysis" --db-username
"jsmith" --db-password "secret-password"

tabcmd publish "analysis_sfdc.hyper" -n "Sales Analysis"
--oauth-username "user-name" --save-oauth
```

ファイルが tabcmd と同じディレクトリにない場合、ファイルへの完全なパスを含めます。

例

```
tabcmd publish "\\computer\volume\Tableau Workbooks\analysis.twbx" -
n "Sales_Analysis" --db-username "jsmith" --db-password "secret-
password"

tabcmd publish "\\computer\volume\Tableau Workbooks\analysis_
sfdc.hyper" -n "Sales Analysis" --oauth-username "username" --save-
oauth
```

オプション

`-n, --name`

サーバー上のワークブックまたはデータソースの名前。省略した場合は、ワークブック、データソース、またはデータ抽出がファイル名の後で指定されます。

`-o, --overwrite`

既にサーバー上に存在するワークブック、データソース、またはデータ抽出を上書きします。

`-r, --project`

指定されたプロジェクトにワークブック、データソース、データ抽出をパブリッシュします。未指定の場合は "既定" プロジェクトにパブリッシュされます。

`--parent-project-path`

`-r` オプションで指定されたネストされたプロジェクトで、親プロジェクトの名前を付けます。たとえば、"Main (主)" プロジェクトにある "Nested (ネスト)" という名前のプロジェクトを指定するには、構文 `--parent-project-path "Main" -r "Nested"` を使用します。

`--db-username`

ワークブック、データソース、またはデータ抽出と共にデータベースユーザー名をパブリッシュするにはこのオプションを使用します。

`--db-password`

ワークブック、データソース、または抽出と共にデータベースパスワードをパブリッシュするにはこのオプションを使用します。

`--save-db-password`

指定されたデータベースパスワードをサーバーに保存します。

`--oauth-username`

ユーザーアカウントの電子メールアドレス。`--name` に指定されたクラウドデータソースの保存されたアクセストークンをユーザーが既に持っている場合、事前に構成された OAuth 接続からユーザーに接続します。アクセストークンはユーザー設定で管理できます。

データソースへの既存の OAuth 接続では `--db-username` および `--db-password` ではなく、このオプションを使用します。

`--save-oauth`

--oauth-username により指定された認証資格情報を、パブリッシュされたワークブックまたはデータソースと一緒に埋め込み認証資格情報として保存します。

その後、パブリッシャーまたはサーバー管理者がサーバーにサインインして、そのワークブックまたはデータソースの接続を編集すると、接続設定ではこの OAuth 認証資格情報がコンテンツに埋め込まれているように表示されます。

パブリッシュ後に抽出更新をスケジュールする場合、--oauth-usernameを使用してこのオプションを含める必要があります。これは、従来のデータベース接続で --save-db-password を使用するのと似ています。

--thumbnail-username

ワークブックにユーザーフィルターが含まれている場合、指定されたユーザーが表示できる内容に基づいてサムネイルが生成されます。--thumbnail-group オプションが設定されている場合は、指定できません。

--thumbnail-group

ワークブックにユーザーフィルターが含まれている場合、指定されたグループが表示できる内容に基づいてサムネイルが生成されます。--thumbnail-username オプションが設定されている場合は、指定できません。

--tabbed

タブ付きのビューと共にワークブックがパブリッシュされた場合、各シートはタブになります。ビューアーはこのタブを使用して、ワークブック内のシートを切り替えることができます。シートレベルのセキュリティは、この設定によってオーバーライドされます。

--append

既存のデータソースに抽出ファイルを追加します。

--replace

既存のデータソースを置換するために抽出ファイルを使用します。

`--disable-uploader`

増分ファイル アップローダーを無効にします。

`--restart`

ファイルのアップロードを再開します。

`--encrypt-extracts`

ワークブック、データ ソース、または抽出をサーバーにパブリッシュするときに、抽出を暗号化します。詳細については、保存中の抽出の暗号化を参照してください。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

publishsamples

説明

Tableau のサンプル ワークブックを指定したプロジェクトにパブリッシュします。既存のサンプルはすべて上書きされます。

構文

```
tabcmd publishsamples -n [project name] [Global options]
```

例

[既定] サイトで、内部営業プロジェクトにサンプルをパブリッシュします (ユーザー = jsmith)。

```
tabcmd publishsamples -n "Inside Sales" -t "" -s localhost --
username "jsmith" --password "secret-password"
```

オプション

`-n, --name`

必須。Tableau のサンプルを指定したプロジェクトにパブリッシュします。プロジェクト名にスペースが含まれている場合は、名前全体を引用符で囲んでください。

`--parent-project-path`

`-n` オプションで指定されたネストされたプロジェクトで、親プロジェクトの名前を付けます。たとえば、"Main (主)" プロジェクトにある "Nested (ネスト)" という名前のプロジェクトを指定するには、構文 `--parent-project-path "Main" -n "Nested"` を使用します。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認

証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された .txt ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス no-

を使用します。既定ではセッションは保存されます。

--timeout

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

--

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

reencryptextracts

新しい暗号化キーを使用して、サイト上のすべての抽出を再暗号化します。このコマンドはキー暗号化キーとデータ暗号化キーを再生成します。サイトを指定する必要があります。詳細については、保存中の抽出の暗号化を参照してください。

抽出の数とサイズによっては、この操作によって大量のサーバーリソースが消費される場合があります。このコマンドは、通常の営業時間外に実行することを検討してください。

例

```
tabcmd reencryptextracts "Default"
```

```
tabcmd reencryptextracts "West Coast Sales"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンドラインではなく指定された .txt ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス no-

を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

refreshextracts *workbook-name* または refreshextracts *datasource-name*

指定されたワークブックまたはデータソースに属する抽出の完全更新または増分更新を実行します。

このコマンドは、パブリッシュ時のファイル名ではなく、サーバー上に表示されるワークブック名またはデータソース名を取得します。ワークブックまたはデータソースの管理者や所有者のみがこの操作を実行できます。

> **注:** サーバー管理者がサイトで **[今すぐ実行]** 設定を無効にしている場合、このメソッドは失敗してエラーになります。詳細については、「サーバー設定」を参照してください。

例

```
tabcmd refreshextracts --datasource sales_ds
```

```
tabcmd refreshextracts --project "Sales External" --datasource
sales_ds
```

```
tabcmd refreshextracts --project "Sales External" --parent-project-
path "Main" --project "Sales External" --datasource sales_ds

tabcmd refreshextracts --workbook "My Workbook"

tabcmd refreshextracts --url SalesAnalysis

tabcmd refreshextracts --workbook "My Workbook" --addcalculations

tabcmd refreshextracts --datasource sales_ds --removecalculations
```

オプション

`--incremental`

増分更新操作を実行します。

`--synchronous`

バックグラウンダープロセスが使用できるとすぐに実行されるように、完全なリフレッシュ操作をバックグラウンダープロセスで使用されるキューに追加します。バックグラウンダープロセスが使用できる場合、操作はすぐに実行されます。リフレッシュ操作がバックグラウンドタスクレポートに表示されます。

同期更新の間、tabcmd はリフレッシュ操作の実行中にサーバーとのライブ接続を保ち、バックグラウンドジョブが実行されるまで毎秒ポーリングします。

> **注:** `--synchronous` オプションは、Tableau Bridge で更新されたデータソースでは使用できません。

`--workbook`

更新する抽出を含むワークブックの名前。ワークブック名にスペースが含まれる場合、名前を引用符で囲みます。

`--datasource`

更新する抽出を含むデータソースの名前。

`--project`

"*既定*" 以外のプロジェクトのワークブックまたはデータソースを指定するには、`--workbook` または `--datasource` と共に使用します。未指定の場合は "既定" プロジェクトが使用されます。

`--parent-project-path`

`--project` オプションで指定されたネストされたプロジェクトで、親プロジェクトの名前を付けます。

例:

- たとえば、"Main (主)" プロジェクトにある "Nested (ネスト)" という名前のプロジェクトを指定するには、次の構文を使用します
- `--parent-project-path "Main" --project "Nested"`
- "ネスト" プロジェクト内でネストされている "Nested2" という名前のプロジェクトを指定するには、次の手順を実行します。
- `--parent-project-path "Main/Nested" --project "Nested2"`

`--url`

URL に表示されるワークブック名です。"Sales Analysis" としてパブリッシュされたワークブックの URL 名は "SalesAnalysis" になります。

`--addcalculations`

ワークブックの埋め込まれた抽出の計算をマテリアライズするには `--workbook` と一緒に使用し、抽出データソースの計算をマテリアライズするには `--datasource` と一緒に使用します。バックグラウンダープロセスが使用するキューに操作を追加します。バックグラウンダープロセスが使用できる場合、操作はすぐに実行されます。この操作は [抽出でのバックグラウンドタスク] 管理ビューに表示されます。

`--removecalculations`

以前マテリアライズされた計算を削除するには、`--workbook` または `--datasource` と一緒に使用します。バックグラウンダープロセスが使用するキューに操作を追加します。バックグラウンダープロセスが使用できる場合、操作はすぐに実行されます。この操作は [抽出でのバックグラウンドタスク] 管理ビューに表示されます。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

reset_openid_sub

既に Tableau Server ID に関連付けられている OpenID Connect 識別子 (sub 値) をクリアします。Tableau Server で OpenID Connect 用に IdP を変更するを参照してください。

例

```
tabcmd reset_openid_sub --target-username jsmith
```

オプション

`--target-username`

指定した個々のユーザーの sub 値をクリアします。

`--all`

すべてのユーザーの sub 値をクリアします。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

removeusers *group-name*

指定したグループからユーザーを削除します。

例

```
tabcmd removeusers "Development" --users "users.csv"
```

オプション

`--users`

指定されたグループから指定された .csv ファイル内のユーザーを削除します。ファイルは、1 行にユーザー名が 1 つの簡単なリストでなければなりません。

`--[no-]complete`

変更を正常に行うには、すべての行が有効であることが必要です。指定しない場合、`--complete`が使用されます。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンド ラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

runschedule *schedule-name*

現在ログインしているサイトのタスクを指定されたスケジュールで実行します。すべてのサイトに対してTabcmd を使用して実行することはできません。すべてのサイトのタスクをスケジュールで実行するには、Web インターフェイスにログインし、**[スケジュール]** ページから**[すべてのサイト]** を選択して、スケジュールで **[今すぐ実行]** を実行します。

このコマンドは、サーバーで指定したスケジュールの名前を使用します。

このコマンドは Tableau Cloud では使用できません。

> **注:** サーバー管理者がサイトで **[今すぐ実行]** 設定を無効にしている場合、このメソッドは失敗してエラーになります。詳細については、「サーバー設定」を参照してください。

例

```
tabcmd runschedule "5AM Sales Refresh"
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンドラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト

ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

set *setting*

サーバー上の指定された設定を有効にします。各設定の詳細は、サーバーの [メンテナンス] ページで参照できます。

設定を無効にするには、設定名の前に感嘆符を付けてください。有有効にしたり無効にしたりできる設定は次のとおりです。

- allow_scheduling
- embedded_credentials
- remember_passwords_forever

例

```
tabcmd set embedded_credentials
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

--

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

syncgroup *group-name*

Tableau Server グループを Active Directory グループと同期化します。Tableau Server グループがまだ存在しない場合、指定された Active Directory グループと同期して作成されます。

グループの名前に "@" が含まれている場合 (ドメイン セパレータ以外で)、16 進形式 "\0x40" を使用して記号を参照する必要があります.

例

```
tabcmd syncgroup "Development"
```

```
tabcmd syncgroup "Dev\0x40West"
```

注: 自分がメンバーであるグループを同期する場合、このコマンドを使用して行った変更は、自分自身のユーザーに適用されません。たとえば、自分がメンバーであるグループのユーザーから、このコマンドを使用して管理者権限を削除した場合、そのコマンドが実行されても自分自身は管理者のままです。

オプション

--grant-license-mode <grant-license-mode>

サインイン時にロールを付与するかどうかを指定します。既定は on-syncです。有効な値は、on-login、on-sync です。値を指定しない場合、on-syncが使用され、グループの同期時に既定のロールが付与されます。詳細については、"サインイン時にロールを付与"が有効になっているユーザー ロールを変更するを参照してください。

--no-publisher

非推奨です。代わりに、--role オプションを使用します。

--overwritesiterole

--role を使用すると、ユーザーのサイトロールを特権の少ない役割で上書きできます。デフォルトでは、ユーザーのサイトロールは --role を使用して昇格できますが、降格はできません。--overwritesiterole オプションはユーザーのサイトロールを降格するため、注意して使用する必要があります。

--publisher

非推奨です。代わりに、--role オプションを使用します。

-r, --role

グループ内のユーザーのサイトロールを指定します。既定では Unlicensed です。

有効な値は、SiteAdministratorCreator、SiteAdministratorExplorer、SiteAdministrator、Creator、ExplorerCanPublish、Publisher、Explorer、Interactor、Viewer、Unlicensed.です。

--silent-progress

コマンドの進捗状況メッセージを表示しません。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

upgradethumbnails

サムネイルのアップグレードジョブを開始および停止します。詳細については、サムネイルのアップグレードジョブを参照してください。

例

サムネイルのアップグレードジョブを開始するには:

```
tabcmd upgradethumbnails --server <serverURL>
```

進行中のアップグレードサムネイルジョブを停止するには:

```
tabcmd upgradethumbnails --server <serverURL> --stop
```

オプション

`--stop`

このオプションを指定すると、進行中のアップグレードサムネイル ジョブが停止します。このオプションを指定しない場合、サムネイルのアップグレードジョブが開始します。

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。`--` を使用し、`--` の後に続く値はオプション設定ではなく、コマンドの値として解釈することを `tabcmd` に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、`tabcmd` コマンドで `--` の使用が必要になる場合を示しています。ここで、`-430105/Sheet1` は `export` コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

validateidpmetadata

安全でないダイジェストアルゴリズム SHA-1 を使用する IdP を用いて構成されている Tableau Server サイトを識別します。このコマンドは、RSA キーのサイズまたは楕円曲線のサイズが不十

分である証明書を使用しているIdP も識別します。

> **注:** このコマンドは、サイト固有の SAML にのみ使用できます。詳細については、「サイト固有の SAML の構成」を参照してください。

オプション

`--digest-algorithms <ALGORITHMS>`

ダイジェストアルゴリズムのスペース区切りのリスト有効な値は、`sha1` と`sha256` です。指定しない場合、サーバーはサーバー構成設定の wgserver.saml.blocklisted_digest_algorithms から値を使用します。

`--min-allowed-elliptic-curve-size <SIZE>`

指定しない場合、サーバーはサーバー構成設定の wgserver.saml.min_allowed.elliptic_curve_size から値を使用します。

`--min-allowed-rsa-key-size <SIZE>`

指定しない場合、サーバーはサーバー構成設定の wgserver.saml.min_allowed.rsa_key_size から値を使用します。

`--site-names <SITENAMES>`

証明書の検証を実行するサイト名のスペース区切りのリスト指定されていない場合は、すべてのサイトが検査されます。

グローバル オプション

次のオプションは、すべての `tabcmd` コマンドによって使用されます。セッションを開始するには、`--server`、`--user`、および `--password` オプションを少なくとも 1 回使用する必要があります。認

証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された .txt ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス no-

を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

`--`

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

version

tabcmd ユーティリティの現在のインストールのバージョン情報を出力します。

例

```
tabcmd version
```

グローバル オプション

次のオプションは、すべての tabcmd コマンドによって使用されます。セッションを開始するには、--server、--user、および --password オプションを少なくとも 1 回使用する必要があります。認証トークンは保管されているため、後続のコマンドを実行するときに、これらのオプションを含める必要はありません。このトークンは、それを使用した最後のコマンドの実行後、5 分間だけ有効です。

`-h, --help`

コマンドのヘルプを表示します。

`-c, --use-certificate`

クライアント証明書を使用してサインインします。相互 SSL が有効な場合に指定する必要があります。

証明書の構成方法の詳細については、最初に、Tableau Server OS の次の該当するトピックを参照してください。

- **Windows:** 相互 SSL の構成
- **Linux:** 相互 SSL の構成

`-s, --server`

Tableau Server URL。セッションを開始するために少なくとも 1 回指定する必要があります。

`-u, --user`

Tableau Server ユーザー名。セッションを開始するために少なくとも 1 回指定する必要があります。

`-p, --password`

Tableau Server パスワード。セッションを開始するために少なくとも 1 回指定する必要があります。

`--password-file`

パスワードを、コマンド ラインではなく指定された `.txt` ファイルに保管し、セキュリティを向上させることができます。

`-t, --site`

コマンドを、単一引用符または二重引用符で囲んだ Tableau Server サイト ID によって指定されたサイトに適用することを示しています。既定のサイトを指定するには、単一引用符または二重引用符 (" または "") を付けた空の文字列を使用するか、二重引用符で囲んだ Default を使用します ("Default")。キャッシュされた認証トークンを使用する場合、サイト ID では大文字と小文字が区別されます。大文字と小文字が一致しない場合、トークンがまだ有効であっても、パスワードの入力を求めるメッセージが表示されることがあります。

`-x, --proxy`

Host:Port

指定された HTTP プロキシを使用します。

`--no-prompt`

指定した場合、このコマンドではパスワードが要求されません。有効なパスワードを指定しないと、コマンドが失敗します。

`--no-proxy`

指定した場合、HTTP プロキシは使用されません。

`--no-certcheck`

指定した場合、tabcmd (クライアント) はサーバーの SSL 証明書を検証しません。

`--[no-]cookie`

指定した場合、セッション ID はログイン時に保存されるため、その後にコマンドを実行するときにログインする必要はなくなります。セッション ID を保存しない場合は、プレフィックス `no-` を使用します。既定ではセッションは保存されます。

`--timeout`

サーバーがコマンドの処理を完了するまで、指定された時間 (秒単位) 待機します。既定では、プロセスはサーバーが応答するまで待機します。

--

コマンドラインでオプションの最後を指定します。-- を使用し、-- の後に続く値はオプション設定ではなく、コマンドの値として解釈することを tabcmd に示すことができます。ハイフンを含むコマンドで値を指定する必要がある場合に便利です。次の例では、tabcmd コマンドで -- の使用が必要になる場合を示しています。ここで、-430105/Sheet1 は export コマンドで必要な値です。

```
tabcmd export --csv -f "D:\export10.csv" -- -430105/Sheet1
```

ヒント: Tableau Cloud で使用できる Tabcmd 1.0 コマンドについては、「tabcmd コマンド」を参照してください。

tabcmd 用のスイッチとプロパティのインストール (Windows)

注: tabcmd コマンドライン ユーティリティ バージョン 2.0 は、Tableau tabcmd で入手できます。この新バージョンでは、MacOS や Lunux で tabcmd コマンドを実行し、個人アクセス トークン (PAT) を使って認証することができます。バージョン 2.0 は、Python ベースの Tableau Server クライアント (TSC) で使用可能なパブリック エンドポイントに基づいて構築されています。この最新バージョンでは、Tableau Server のサポートに制限があります。

Tableau Server コマンド ライン ユーティリティ (tabcmd) バージョン 2019.4.0 以降を Windows のコマンド ラインからインストールする場合は、次のスイッチを使用できます。

注: Linux バージョンの tabcmd インストーラーでは相当するスイッチはありません。

スイッチ	説明	コメント
/install | /repair | /uninstall |	セットアップを実行してインス	既定では、UI とすべてのプロンプトをインストールして表示します。新規インストールでディレクトリが指定されていない場合は、C:\Program

/layout "<directory>"	トール、修復、またはアンインストールするか、/layout を使用して、指定したディレクトリにインストール バンドルの完全なローカル コピーを作成します。	Files\Tableau\Tableau Server\<version>\extras\Command Line Utility が使用されます。
/passive	最小限の UI を使用してセットアップを実行します。プロンプトは表示されません。	
/quiet | /silent	無人の完全なサイレント モードでセットアップを実行します。UI もプロンプトも表示されません。	**注:** /silent または /quiet のいずれか (両方ではない) を使用します。
/norestart	再起動が必要な場合でも、Windows を再起動せずにセットアップを	**注:** まれに、このオプションを使用しても再起動を抑制できない場合があります。これは、他のソフトウェアのインストール中など、以前のシステムの再起動

	実行します。	がスキップされた場合に生じる可能性が最も高くなります。
/log "<log-file>"	指定したファイルおよびパスに情報を記録します。 既定では、ログ ファイルはユーザーの %TEMP% フォルダーに作成され、Tableau_Server_Command_Line_utility_<version_code>.log という命名規則が使用されます。	ファイルの場所を指定しない場合、ログ ファイルはユーザーの TEMP フォルダーに書き込まれます (C:\Users\<username>\AppData\Local\Temp)。インストール後、このログ ファイルでエラーを確認します。 例: <Setup file> /silent /log "C:\Tableau\Logs\tabcmd-Install" ACCEPTEULA=1

プロパティ	説明	コメント
ACCEPTEULA=1|0	エンド	/passive、/silent、または /quiet の使用時に

	ユーザーライセンス契約 (EULA) に同意します。クワイエット、サイレント、およびパッシブインストールに必要です。1 = EULA を受け入れる、0 = EULA を受け入れない。	含まれていない場合は、セットアップはダイアログを表示せずに失敗します。 含まれているが 0 に設定されている場合、セットアップは失敗します。
`INSTALLDIR="<path\to\installation\directory>"`	指定した既定以外のインストール場所に tabcmd をインストール	Tabcmd をインストールする場所を指定します。使用されていない場合、tabcmd は `C:\Program Files\Tableau\Tableau Server\<version_code>\extras\Command Line Utility` にイン

	します。	ストールされます 例: `<Setup file> /silent INSTALLDIR="C:\tabc-md"`

トラブルシューティング

次のトピックを使用して、Tableau Server に関する問題のトラブルシューティングと解決を行うことができます。

Linux での Tableau Server のトラブルシューティング

Tableau Server の一般的な問題を解決するには、このトピックの提案に従ってください。ステータスページに表示されるプロセス ステータスに基づいた追加のトラブルシューティングの手順については、サーバープロセスのトラブルシューティングを参照してください。

次の表に、インストール、データ、ログ、およびスクリプトのディレクトリが格納される既定の場所を示します。

ディレクトリ	既定の場所
インストール:	`/opt/tableau/tableau_server`
データ:	`/var/opt/tableau/tableau_server/data`
ログ:	`/var/opt/tableau/tableau_server/data/tabsvc/logs/`
スクリプト:	`/opt/tableau/tableau_server/packages/scripts.<version_code>/`

一般的なトラブルシューティングの手順

Tableau Server の問題の多くは、次のような基本的な手順で対処または試験できます。

クリーン インストール

Tableau Server on Linux を、今まで一度も Tableau をインストールしたことのないコンピューターにインストールします。以前のバージョンの Tableau Server をインストールしたことのあるコンピューターまたは VM を再利用する場合、新しいバージョンをインストールする前にコンピュータから Tableau Server を削除するの手順に従ってコンピューター上から Tableau を完全に削除してください。

Tableau Server のインストール中に問題が発生する場合は、コンピューターから Tableau を完全に削除してからクリーン インストールを実行する必要があります。詳細については、コンピュータから Tableau Server を削除するを参照してください。

ディスク容量

Tableau Server を実行している各コンピューターで十分なディスク容量があることを確認します。ディスク容量が限られている場合、インストールの失敗、アップグレードの失敗、または Tableau Server の実行で問題が生じる可能性があります。

古いログ ファイルの削除

ディスク空き容量が足りない場合は、古い Tableau Server のログ ファイルをクリーンアップできます。これらはディスクのスペースを占有することがあるため、ベストプラクティスとして定期的に削除してください。

バージョン 10.5.x

バージョン 10.5.1 以降の場合、ターミナルのプロンプトで次のコマンドを実行し、必要のないログ ファイルをクリーン アップします。

```
tsm maintenance cleanup
```

バージョン 10.5.0

バージョン 10.5.0 の Tableau Server on Linux を実行している場合、クリーンアップ コマンドは使用できないため、ターミナルのプロンプトで次のコマンドを実行する必要があります。

```
sudo find /var/opt/tableau/tableau_server/data/tabsvc/temp/* -mtime
+2 -type f -delete
```

```
sudo find /var/opt/tableau/tableau_server/data/tabsvc/logs/* -mtime
+2 -type f -delete
```

重要: Linux ファイル システムによって、開いているファイルを削除することができます。この場合、Tableau プロセスでファイルを再作成することはできません。これにより、ログ ファイルが空になります。これを修正するには、Tableau Server を停止し、TSM コントローラーを再起動して Tableau を再起動します。

1. Tableau Server を停止します。

   ```
   tsm stop
   ```

2. TSM コントローラーを再起動します。

   ```
   sudo systemctl restart tabadmincontroller_0.service
   ```

3. コントローラーが再起動するまで数分間待機します。次のコマンドを指定してコントローラーが再起動したことを確認できます。

   ```
   tsm status -v
   ```

   このコマンドを実行すると、Tableau Server 管理 コントローラーにコントローラーが再起動したことを示す 'running' と表示されます。

4. Tableau Server を起動します。

   ```
   tsm start
   ```

手動のログ収集

何らかの理由で `tsm maintenance ziplogs` を実行できない場合 (`tsm initialize` を実行する前に重大な障害が発生するなど)、次のコマンドを Tableau Server 展開の各 ノードのターミ

ナル ウィンドウで実行すると、ログを手動で収集して圧縮することができます。

```
cd /var/opt/tableau/tableau_server/data/tabsvc/

cp /var/opt/tableau/tableau_server/logs/app-install.log logs

cp ~/.tableau/tsm/tsm.log logs

tar -czvf ~/logs.tar.gz logs
```

これにより、ホーム ディレクトリに logs.tar.gz というファイルが作成されます。このファイルを Tableau にアップロードまたは送信できます。

サーバーの再起動

Tableau Server を再起動します。インデックスやプロセスが完全に動作しない問題については、Tableau Server を制御された方法で再起動することによって解決できます。Tableau Server を再起動するには、`tsm restart` コマンドを使用します。このコマンドでは、Tableau Server に関連付けられたすべてのプロセスを停止して、再起動します。

Linux を使用したインストール ファイルおよび構成ファイルの編集

Linux オペレーティング システムを使用して Tableau Server on Linux のインストールまた構成に使用されたファイルを編集または作成する必要があります。Microsoft Windows を使用して作成されたファイルによって、Tableau Server on Linux のインストールおよび構成時にエラーが発生します。これは、Linux のオペレーティング システム ファイルがライン フィード (LF) 文字で終了しているのに対し、Windows のファイルではキャリッジ リターン文字とライン フィード文字 (CR LF) で終了しているためです。また、登録中、アイデンティティ ストア設定の構成時、またはゲートウェイの設定時にも、Linux 以外 (CR LF) のファイル末尾によってエラーが発生する可能性があります。

systemd ログの確認

Tableau Server が起動せず、Tableau ログで有用な情報が見つからない場合 (詳細については ログ ファイルの操作を参照)、systemd ログで TSM サービスの起動と停止に関するメッセージのログを確認できます。このログは `/var/log/messages` (RHEL ライクなディストリビューション) または `/var/log/syslog` (Ubuntu) に保存されます。`journalctl` コマンドを使用して、システムのログを検索して解析することをお勧めします。

Tableau Server のインストール

ハードウェア要件のためインストールが失敗する

Tableau Server は、インストール先のコンピュータがハードウェアの最低要件を満たしていない場合はインストールできません。要件の詳細については、インストールの前に...を参照してください。

タイムアウトのためインストールが失敗する

Tableau Server をリソースに制限のあるコンピューター、たとえば、最低ハードウェア要件をちょうど満たすコンピューターにインストールする場合、応答が遅いため tsm コマンドがタイムアウトになる問題が発生する可能性があります。すべての tsm コマンドで `--request-timeout` グローバル オプションを使用し、タイムアウトが長くなるように指定できます。`--request-timeout` オプションの詳細については、tsm initializeなどを参照してください。

"Failed to initialize the instance of the temporary database" と表示されインストールに失敗する

Tableau Server on Linux は UTF-8 文字エンコードのみに対応しています。お使いの Linux のロケールで UTF-8 エンコードが不足している場合、インストールは以下のようなエラーが表示されて失敗する可能性があります。

```
Failed to initialize the instance of the temporary database
```

お使いのロケールで UTF-8 エンコードを使用しているかどうかを確認するには、コマンドプロンプトで `localectl` コマンドを使用します。結果として生じる出力は以下のようになります (ロケールが異なる可能性があります)。

```
[tableauserver-centos1a ~]$ localectl
System Locale: LANG=en_US.UTF-8
[tableauserver-centos1a ~]$
```

`LANG` 値に `.UTF-8`が含まれない場合、`localectl` を実行してそれを追加する必要があります。

```
sudo localectl set-locale LANG=<your_locale>.UTF-8
```

注: 場合によっては、`systemd` のバージョンが古く`localectl` が完了しない (タイムアウトする) ことがあります。この問題は systemd を更新すると修正され、UTF-8 エンコードを設定できる場合があります。RHEL ライクなシステムでこのコマンドを使用して `systemd`: `sudo yum update systemd` を更新します

Parallels の仮想マシンでインストールが失敗する

Parallels は現在サポートされていません。Parallels の Linux 向け仮想マシンに Tableau Server をインストールする場合、インストールが失敗する可能性があります。

Tableau Server が起動しない

Tableau Server が開始しないか機能が低下した状態で実行されている場合、`tsm restart` コマンドを実行します。これによって実行中のすべてのプロセスをシャットダウンして、Tableau Server を再起動します。

インストール後に Tableau Server を起動できない

インストール後にコンピュータのホスト名が変わると、Tableau Server が起動しない可能性があります。ホスト名が変わる可能性がある主な理由の 1 つは、CentOS で cloud-init パッケージを使用する場合です。cloud-init パッケージを使用する場合は、インストール処理を開始する前に、Tableau Server をインストールするコンピュータを*再起動*してください。あるいは、次のコマンドを実行することにより、再起動せずにホスト名を修正できます。

```
sudo hostnamectl set-hostname `hostnamectl --static`
```

一般に、cloud-init パッケージは、新しい仮想マシンの初期化や SSH 公開キー認証の構成などに使用されます。たとえば、一部の CentOS イメージでは cloud-init が使用されます。また一般に、OpenStack の展開でも cloud-init が使用されます。ただし、CentOS 7.x リポジトリに既定で含まれているバージョンの cloud-init (cloud-init 0.7.5-10.el7.centos.1) には、コンピュータを再起動しないとコンピュータのホスト名と共に完全修飾ドメイン名 (FQDN) が表示されないという既知の問題があります。

Tableau Server のインストール処理では、コンピュータのホスト名を使用してサーバープロセスの構成および TLS 証明書の生成が行われるため、Tableau Server が FQDN の付いていないホスト名を使用するよう構成されている場合は起動しない可能性があります。

コンピュータで正しいホスト名が表示されているかどうかを確認するには、hostnamectl コマンドを実行します。次の例では、FQDN を返さないために再起動の必要があることを示す transient ホスト名がコマンドによって表示されています。

```
$ hostnamectl
   Static hostname: server01.example.com
Transient hostname: server01
[...]
```

それに対して次の例では、コマンドによって正しいホスト名とFQDN が表示されています。

```
$ hostnamectl
   Static hostname: server01.example.com
[...]
```

複数の Active Directory (AD) ドメインを使用すると初期管理者アカウントを作成できない

Tableau Server で初期管理者アカウントを作成するときに、認証タイプとして AD を選択した場合、次のようなエラーが表示される可能性があります。

```
Failed to authenticate username and password
```

これは、Tableau Server が複数の AD ドメインに接続しようとしたときにすると発生します。たとえば、あるドメインに属するコンピュータに Tableau Server をインストールし、別のドメインに属する AD ユーザーを認証しようとすると、このエラーが表示される可能性があります。

フォント

Tableau Server はワークブック作成時のフォントに基づき、システムにインストールされているフォントを使用してワークブックをレンダリングします。あるフォントを利用できない場合、Tableau Server はフォントファミリーに基づき最も近い同等のフォントを使用します。Windows でも Linux サーバーでも処理は同じです。Linux では Windows や OS/X システムと比較して出荷時に含まれるフォント数が少ないため、Linux サーバーでは不足フォントがより明白になることがあります。これが問題とな

るのは、多くのワークブックは Tableau Desktop on Windows または on Mac で作成されているためです。

Tableau Server on Linux には出荷時に以下のフォントが含まれています。

- Arial
- Courier
- Georgia
- Times New Roman
- Verdana
- Trebuchet MS
- Tableau のフォント

これら以外のフォントを使用するワークブックは、Tableau Server on Linux で表示すると、フォントが見つからないため期待した外観と異なる場合があります。この問題を解決するには Tableau Server インストールのすべてのノードに適切なフォントをインストールします。

アジア言語の文字セットへのサポート

Tableau Server に表示されるワークブックでアジア言語の文字が表示されるはずのボックスが空になっている場合は、Linux 環境に適切な言語のフォントパッケージをインストールする必要があります。

Tableau Server の初期化

TSM 初期化は、`tableau` ユーザーアカウントは存在しますが、`tableau` グループのメンバーではないため失敗します。

Tableau サービスマネージャー (TSM) および Tableau Server をインストールして初期化する際、初期化スクリプト (`initialize-tsm`) は実行するために必要なユーザーとグループを作成するか、既存のものが必要な特徴で構成されているか確認します。既定ではスクリプトが`tableau` と呼ばれるユーザーを作成し、そのユーザーを `tableau` というグループに追加します。`tableau` ユーザーが既に存在するが `tableau` グループの一員ではない場合、スクリプトは警告を出して失敗します。

このような状況が起きた場合、`--unprivileged-user` フラグを使って異なるユーザーを指定して、このユーザーが作成され`tableau` グループに追加されることでこの競合を解消できます。

たとえば、tableauserver というユーザーを指定するには、次のコマンドを使用して /opt/tableau/tableau_server/packages/scripts.<version_code> ディレクトリからスクリプトを実行するでしょう。

```
sudo ./initialize-tsm --unprivileged-user="tableauserver" --
accepteula
```

initialize-tsm スクリプトで使用できるオプションの完全なリストを参照するには、-h オプションを使います。

```
sudo ./initialize-tsm -h
```

未サポートのシステム ロケール上での Tableau Server 初期化エラー

サポート対象ロケールに含まれないロケールの PC 上に Tableau Server のインストールを試行すると、インストール中にエラーが発生します。

Tableau Server は次のロケールの 1 つを使用するシステム上で実行します。

de_DE, en_GB, en_US, es_ES, fr_FR, it_IT, ja_JP, ko_KR, nl_NL pt_BR, zh_CN, zh_TW

fr_CA (バージョン 2022.3 以降)

th_TH、sv_SE (バージョン 2023.1)

nl_NL (バージョン 2025.1 以降)

その他のロケールでは、エラーが発生します。

en_US.utf8 がロケール リストに含まれず発生する Tableau Server 初期化エラー

en_US.utf8 がロケール リストに含まれないコンピューター上に Tableau Server のインストールを試行すると、インストール中にエラーが発生します。en_US.utf8 が含まれているかどうか確認するには、shell プロンプトで locale -a と入力します。

en_US.utf8 がリストにない場合、Ubuntu 上の shell プロンプトで sudo locale-gen en_US.UTF-8 と入力するか、RHEL ライクなディストリビューションの shell プロンプトで sudo localedef -i en_US -f UTF-8 と入力して en_us をロケール リストに追加できます。

エラー: ステータス 10 - データディレクトリのパスにピリオドが含まれる場合の Tableau Server 初期化

Tableau Server のインストールでピリオド(".") を含むパスのデータディレクトリを指定しようとすると、以下を含むエラーにより初期化に失敗します。

```
Connection timed out
```

および

```
ERROR: TSM services returned status 10
```

この問題を回避するには、パスにピリオドを含まないデータディレクトリを選択してください。

再インストール後の Tableau Server の初期化エラー

Tableau Server をアンインストールして再インストールする場合、Tableau Server の初期化エラーが発生する可能性があります。たとえば、次のエラーが表示される場合があります。

```
ERROR com.tableau.tabadmin.webapp.asyncjobs.JobStepRunner - Running
step WaitForConfigure failed
com.tableau.tabadmin.webapp.exceptions.ServiceFailedStateException
```

前のインストールからサービス開始の失敗原因となるアーティファクトが残っている場合にこのエラーが発生します。このエラーを防ぐため、`tableau-server-obliterate` スクリプトを `/opt/tableau/tableau_server/packages/scripts.<version_code>` フォルダーで使用します。Tableau Server の完全削除に関する詳細は、コンピュータから Tableau Server を削除するを参照してください。

Tableau Server のライセンス認証

Tableau Server のライセンス認証の失敗

場合によって、`tsm licenses activate -k <product_key>` コマンドを使用した Tableau プロダクトキーのライセンス認証が次のエラーにより失敗することがあります。

```
License Server not available
```

使用しているコンピュータが TCP ポート 443 で `licensing.tableau.com` の Tableau ライセンス発行サーバーに接続できない場合にこのエラーが発生します。

この問題を解決するには、このアドレスおよびポートへのアクセスを許可するようにネットワークおよび/またはホストベースのファイアウォールを構成するか、または Tableau をオフラインでライセンス認証する必要があります。詳細については、オフラインでの Tableau Server の有効化を参照してください。

Tableau Server の検索とブラウズの再インデックス

検索とブラウズの再インデックスによって解決する他の問題

再構築が必要なインデックスの症状には次が含まれます。

- ユーザーがログインを試みるとサイトのリストが空になっている
- ユーザーがプロジェクトの選択を試みるとプロジェクトのリストが空になっている
- コンテンツがない (ワークブック、ビュー、ダッシュボード)
- 予期しない、または不正確なアラート (たとえば、"更新の失敗" 抽出を含まないワークブックのアラート)

これらの症状が見られる場合、`tsm maintenance reindex-search` コマンドを使用して検索および閲覧インデックスを再構築します。

Tableau Server の再起動

Tableau Server の再起動または変更の適用が失敗する

Tableau Server サービスのいずれかが失敗すると、サーバーの再起動または構成変更の適用を行おうとしたときに、エラーが表示される可能性があります。

失敗したサービスがエラーの原因となっているのかを確認するには、次のコマンドを入力します。

```
tsm status -v
```

サービスの失敗原因を調べるには、データディレクトリ内の tabadminagent および tabadmincontroller ログ ファイルを確認します。たとえば、同時性の問題やポート構成の問題が原因でサービスが失敗する場合があります。フィードバックには、発生した問題について記入してください。

回避策として、TSM でサービスを削除してから追加しなおすことにより障害の解決を試みることができます。サービスが開始されたら、`tsm restart` コマンドで以前行った構成変更またはサーバーの再起動を再試行します。

ノードの追加または構成後の Tableau Server の再起動エラー

ゲートウェイプロセスを使用せずにノードを追加または構成する場合、Tableau Server の再起動が失敗し、次のようなエラーが表示される可能性があります。

```
ERROR :
com.tableau.tabadmin.configuration.PortConfigurationExtractor -
Unable to find port config key worker1.gateway.port
```

および

```
Message: Missing port configuration value for key
'worker1.gateway.port'
```

これらのエラーは、gateway.log ファイルに表示され、Tableau Server ノードがゲートウェイを使用せずにアプリケーション サーバーまたは VizQL Server を使用して構成されている場合に発生します。アプリケーション サーバーまたは VizQL Server のいずれかがノードで稼働している場合は、ゲートウェイプロセスが必要です。

バックアップ/復元

Tableau Server により作成されるバックアップの復元に関する問題は、許可の問題の結果である可能性があります。TSM が復元しているファイルと、ファイルの場所の両方に適切な許可が必要です。TSM がバックアップを取り扱うとき、そのファイルを既定の場所に配置し、適切な許可を設定します。許可の問題は、お使いの Linux サーバーにコピーされたバックアップ、またはお使いのサーバー上の既定の場所でないところに配置されたバックアップの復元を試みた場合に発生する可能性があります。既定でない場所の使用と、場所を変更する方法を含む詳細については、tsm ファイル パスを参照してください。

次のエラーが含まれる場合があります。

```
Server Was Denied Access to File
```

または

```
Restoring the backup '<backup>.tsbak' was unsuccessful
```

または

```
Comparing authentication methods failed
```

Tableau Server のバックアップおよび復元プロセスは次を必要とします。

- [読み取り] パーミッション - プロセスで .tsbak バックアップファイルに直接アクセスする必要があります。
- [実行] パーミッション - プロセスでは、.tsbak ファイルが配置されているディレクトリ構造への実行パーミッションも必要となります。

TSM からバックアップを既定の場所に作成すると、必要な許可が設定されます。ファイルを Linux サーバーにコピーする、または既定でない場所に移動すると、TSM プロセスに適切なアクセスが許可されない場合があります。ファイルおよびそれを含むディレクトリツリーの両方が TSM ユーザー *tableau* のアクセスを許可することを確認する必要があります。ファイルのパーミッションでは、ユーザー *tableau* に .tsbak ファイルに対する読み取りアクセスが付与されている必要があります。これは、ファイル上のグループを*tableau* グループに設定し、そのグループに読み取りアクセスを付与することで実行できます。ディレクトリ許可から、ユーザー *tableau* に読み取りアクセスが付与されなくてはなりません。これは、ディレクトリ上のグループを*tableau* グループに設定し、そのディレクトリに読み取りおよび実行アクセスを付与することで実行できます。

TSM とファイル許可の詳細については、TSM におけるファイルとパーミッションを参照してください。

ファイルの場所

basefilepath を変更しても既存のファイルの場所が変わらない

いくつかの tsm コマンドは既定の場所にファイルを書き込みます。tsm set コマンドを使って各コマンドの既定の場所を変更できますが、それでも既存のファイルを元の場所から新しい場所に移動したり、新しい場所を作成したりすることはありません。新しい場所はユーザーが作成し、その場所にあるファイルおよびファイルを含むディレクトリ全体の構造への tsm アクセスを許可する適切なパーミッションを指定します。

バックアップ、復元、サイトのインポートおよびエクスポート、および圧縮されたログ ファイルの既定の場所の変更について詳細は、tsm ファイル パスを参照してください。

tsm パーミッションの詳細については、TSM におけるファイルとパーミッションを参照してください。

TSM コマンド

TSM コマンド ラインが長期実行されるタスクの進捗を表示しない

完了までに 2 時間を超える restore や ziplogs などの tsm コマンドを実行する場合、コマンドはサーバー上で作業が完了するまで実行され続けます。そのようなジョブの進捗を表示するには、`tsm jobs reconnect` コマンドを使用します。

ファイアウォール ポートを開く

Ubuntu でファイアウォール ポートを手動で開く

現在の Tableau Server バージョンでは、Ubuntu で使用される `ufw` ファイアウォールがサポートされません。Ubuntu で `firewalld` をインストールしない場合は、これらのポートを手動で開くこともできます。次の手順は、`ufw` の稼動を確認し、任意のソース アドレスからの接続に対して TCP ポート 8850 および 80 を開きます。

1. 次のコマンドを実行して `ufw` の稼動を確認します。

   ```
   sudo ufw status
   ```

   `Status: inactive` という結果が返された場合は、`ufw` を有効にして `ssh` 経由で接続を継続できることを確認する必要があります (本リリース ノートの対象範囲外)。

2. 次のコマンドを実行してポート 8850 へのアクセスを許可します。

   ```
   sudo ufw allow 8850
   ```

3. 次のコマンドを実行してポート 80 へのアクセスを許可します。

   ```
   sudo ufw allow 80
   ```

OpenID が初回のサインイン試行で失敗する

Tableau Server に Open ID Connect 認証を構成した場合、初回のサインイン試行は失敗します。ログインするには、最初の失敗の後に認証を再試行しなくてはなりません。

管理者ビューが表示されない

Tableau Server の [ステータス] タブには、サーバーのメトリクスを表示する可視化へのリンクが含まれています。これらの可視化においては、Tableau Server リポジトリから適切なデータにアクセスするために PostgreSQL ドライバーが必要です。PostgreSQL ドライバーは自動的にインストールされないため、このドライバーをインストールしていないと、ビューが表示されません。詳細については、データベース ドライバーを参照してください。

> **注:** 管理者ビューを使用するには、PostgreSQL ドライバーが、VizQL Server プロセスを実行するノードにインストールされる必要があります。

ビューでのロケールの変更

ビューを開いた後でユーザー ロケールを変更すると、その後にビューを開こうとしても「予期しないエラー」で開くことができなくなります。これまで開いたことのないビューは開けます。

この問題を回避するには、ロケールを変更した後で Tableau Server からサインアウトし、再びサインインします。これで、すべてのビューが適切に表示されるようになります。

ログ ファイルの操作

Tableau Server は正常なアクティビティの一環としてログ ファイルを作成します。Tableau Server の問題をトラブルシューティングする場合、または Tableau サポートが問題を解決するためにログを要求する場合に、サーバー ログ ファイルの使用が必要になることがあります。

zip 形式で圧縮したログ ファイルのアーカイブは `tsm maintenance ziplogs` コマンドを使用して作成できます。zip 形式で圧縮されたアーカイブは、圧縮を解除して確認したり、Tableau サポートへ送ることができるログのコピーを含みます。アーカイブをコピーすると、サーバーからアーカイブ

を削除することができます。ログ ファイルのアーカイブの詳細については、ログ ファイルのスナップショット(ログのアーカイブ)」を参照してください。

このようにトピックを収集することで、ログ ファイルのアーカイブの作成方法、特定のログ ファイルのコンテンツに関する情報、ログを参照する可能性がある場合やその方法についての詳細などの情報が得られます。

Tableau Server ログのコンテンツ

すべての Tableau Server プロセスでは、その処理内容に関する情報が独自のログ ファイルに書き込まれます。これにより、各プロセスのアクションに関する詳細な情報が提供されます。これらのログ ファイルには、ユーザーの要求を処理したり、自動化タスクを実行している際の、Tableau Server コンポーネント間の内部通信に関する詳細情報が含まれています。Tableau Server ログには、トラブルシューティングに役立つ技術情報のみが含まれています。たとえば、さまざまなコンポーネントのステータス、異なるプロセスによるアクション、通信の試行、データベースへのクエリ(結果を含まない)、および要求のタイミングなどです。

ログ ファイルには、データベース サーバーの名前、IP アドレス、ポート、Tableau Server コンピューターの名前または IP アドレス、ユーザーがアクセスするワークブックやビューの URL と名前など、特定のデータが含まれる場合があります。

ログ ファイルには、パスワード、クエリの結果、ビューに表示されるデータなどの重要な顧客データは含まれません。

> **注:** DEBUG レベルでログを記録する場合、Tableau の起動時に完全な環境情報が収集されます。これは、環境変数に機密情報がある場合、その機密情報がログに含まれる可能性があることを意味します。デフォルトの INFO レベルでログを記録する場合は、安全な環境情報のみが収集されます。

`tsm maintenance ziplogs` コマンドを使用すると、ログ ファイルの圧縮アーカイブを生成するだけでなく、`-d` オプションが指定されている場合に Tableau Server リポジトリデータも含めることができます。リポジトリには、Tableau Server からのメタデータ(ユーザー名、グループ、プロジェクト、

Tableau Server でのパーミッション、抽出更新スケジュールなど) が含まれています。リポジトリには、ワークブックのレイアウト情報や接続情報も含まれますが、パスワード、データベースからの実際のデータ、ビューに表示されるデータなどのデータは含まれません。

ビューに表示されるデータは、抽出ファイルまたはデータベースから取得され、メモリにキャッシュされます。これらのデータは、ログに保存されないか、ライブ接続の場合は、Tableau Server コンピューター上の別個のファイルに保存されません。抽出ファイルは、Tableau Server コンピューターで `dataengine` フォルダー内の `.hyper` ファイルとして格納されますが、zip 形式のログ アーカイブには含まれません。

Tableau Server の問題の調査

Tableau Server で発生する可能性がある問題の範囲や複雑さを考慮すると、すべての問題を調査するのに使用できる簡単なプロセスはありません。ただし、一般的な方法として次の手順が含まれます。

1. サイズを小さくするために、既存のログ ファイルの**クリーン アップ**を行います。詳細については、「不必要なファイルの削除」を参照してください

   重要: Tableau サポートから支援を受けて、問題のトラブルシューティングを行う場合は、ログをクリーンアップする前に、ログのアーカイブを zip 形式で必ず作成してください。クリーンアップすると、サポートに必要な重要な情報が削除される可能性があります。ログ アーカイブの作成の詳細については、「ログ ファイルのスナップショット (ログのアーカイブ)」を参照してください。

2. **適切なロギング レベルの設定**。これは、Tableauサポートが指示を行います。さまざまなログ レベルの影響などの詳細については、「ロギング レベルの変更」を参照してください。
3. トラブルシューティングを行って**問題を再現**することで、問題に関連するイベントをログに記録します。

4. ログの**アーカイブを作成**します。詳細については、「ログ ファイルのスナップショット (ログのアーカイブ)」を参照してください。

重要: ログ ファイルを見るときにはこのアーカイブを使用します。サーバー上にあるファイルを直接編集、移動、または削除しないことをお勧めします。

5. TSM 管理コントローラー ログ (`/tabadmincontroller/tabadmincontroller_node<n>-<n>.log`) を確認して、TSM から開始されたジョブを含む、コマンド ライン、Web UI、または API から行われた構成や展開を把握します。コントローラー ログから始めてください。有用な情報はほとんどここから得ることができます。

   注: `tsm.log` の説明は `tabadmincontroller_*.log` よりも短いですが、役に立つ補足のトラブルシューティング情報を提供できます。

6. **Apache ログ** (`/httpd/access.####_##_##_##_##_##.log` および `/httpd/error.log`) を確認し、調査中の問題に関連すると思われる要求を見つけてください。

   Apache ログには相当な量の「ノイズ (不要な情報)」が含まれており、これは発生中の問題に該当しません。

   - 問題に関連すると思われる要求を確認する場合、`vizqlserver` ディレクトリで Apache ログの一意の要求 ID を含むエントリを検索します。
   - 要求 ID に関連する応答コードおよびメッセージを確認します。
   - 問題に関連するワークブック、ビュー、ダッシュボード、またはデータ ソースの名前を検索します。該当するタイムスタンプを探します。
   - 問題に関連している可能性のある要求が見つかった場合、要求に関連付けられた応答コードを確認します (200 番台は問題なく、500 番台は問題があることを示しています)。
   - 特定した要求に関連付けられた固有の要求 ID を見つけます (固有の要求 ID は要求の最後にある 24 個の英数字の文字列です)。

7. **ログ アーカイブをさらに確認**して、他のメッセージや潜在的なエラーを検索します。

   - Apache ログの要求 ID を使用して、関連するログ エントリを含むファイルがあるかどうか、ログ アーカイブの `vizqlserver` フォルダーを検索します。問題を示す事例を探します (たとえば、エラー メッセージや長期実行中のクエリなど)。

- 無料のオープンソース ツールである Logshark は、ログ アーカイブをレビューするための便利なオプションです。詳細については、Tableau Blueprint の「トラブルシューティング」を参照してください。

8. **スクリプトのログを確認します。**

   Tableau Server には、`/opt/tableau/tableau_server/packages/scripts.<version_code>/` にある scripts ディレクトリに含まれているほとんどの bash スクリプト用のログが含まれています。これらのログはスクリプトが実行される度に `/var/tmp` ディレクトリに保存されます。

   既定: `/var/opt/tableau/tableau_server/data/tabsvc/logs/`

9. **サポートに連絡**

   ご自身で問題を解決できない場合や Tableau サポートから要求があった場合は、アーカイブを zip 形式で圧縮したファイルを Tableau まで送信してください。

Tableau Server ログとログ ファイルの場所

Tableau Server は通常の機能の一部としてログ ファイルを生成します。Tableau Server の一部として稼働している各サービスは、それぞれ独自のログを生成します。それらのログ ファイルには、サーバーで発生している事象、サービスやプロセスが行っている処理、エラーや警告が生成された場合の情報などが含まれます。ログに含まれる情報の範囲は、ログを書き込んでいるサービス、ログ レベルの設定、サーバーで発生している事象などによって異なります。

> Tableau Server on Windows については、「サーバー ログ ファイルの場所」を参照してください。

ログ ファイルは、Tableau Server が抱える問題を特定して修正するのに役立ちます。システム管理者がログを調べると、何が起こっているかについての手がかりが得られる場合があるかもしれません。いっぽう、ほとんどの場合、Tableau Server のログは Tableau サポートにとって大変役立ちます。サポートに問い合わせを行ってケースを開くと、サーバーからログ ファイルを送信するように求められる場合があります。

注: Tableau Server が生成する具体的なディレクトリとログは、実行しているサーバーのバージョンと、構成したプロセスによって異なります。新しいサービスとプロセスは、新しい機能をサポートするために定期的に追加されます。ログを確認する可能性のあるプロセスまたはサービスの詳細については、Tableau Server プロセスを参照してください。

アクティブなクラスター上の Tableau Server ログ ファイル

ベストプラクティスとして、アクティブな Tableau Server インストールにおいては、ログ ファイルを編集または削除しないでください。これを行うと、予期しない動作やサーバーのダウンタイムが発生する可能性があります。ほとんどの Tableau Server ログは、データディレクトリ内の場所に書き込まれます。他の場所に書き込まれるログもあります。

サーバー ログ ファイルを収集して閲覧する最も簡単で安全な方法は、ログ アーカイブを作成することです。ログ アーカイブは、クラスター内のすべてのノードからのログを zip 形式で集めたものです。何らかの理由で古いログが必要だと思われる場合があります。たとえば、アップグレード後に新しいログと比較したり、サーバーの問題のトラブルシューティングを行う際に Tableau サポートに送信したりするためなどです。その場合は、zip アーカイブを作成し、アーカイブを Tableau Server インフラストラクチャの一部ではない安全な場所に移動します。ログ アーカイブ内のログ ファイルの詳細については、Zip アーカイブ内のサーバー ログ ファイルを参照してください。

ログは、特に使用頻度の高いサーバー上で、かなりの領域を占める可能性があります。tsm maintenance cleanup コマンドを使用して不要になったログを削除できますが、既存のログが必要と思われる場合は、クリーンアップの前にアーカイブすることを検討してください。

動作中の Tableau Server インストールのプライマリログの場所

Tableau Server ログのほとんどは、デフォルトでデータディレクトリ `/var/opt/tableau/tableau_server/data/tabsvc/logs/` に書き込まれます。サブディレクトリは、サービスの各インスタンスに対して作成され、サービス名とバージョン コードを含む名前が付けられます。例:

```
/var/opt/tableau/tableau_server/data/tabsvc/logs/backgrounder
```

動作中の Tableau Server インストールの構成ファイルの場所

各サービスまたはプロセスのログに加えて、config サブディレクトリにはサービスに関する構成情報が含まれます。

/var/opt/tableau/tableau_server/data/tabsvc/config/backgrounder

Tableau サポートでは、サーバーの問題に対応している場合、これらの一部を収集するよう求める場合があります。内容を Tableau サポートで分析する場合があります。

プライマリの場所に書き込まれないログ

主要なログ群に含まれず、通常のログディレクトリ以外の場所に書き込まれるログがあります。

- TSM ログ。tsm.log ファイルは <home dir>/.tableau/tsm にあります。
- インストール ログ。app-install.log ファイルは /var/opt/tableau/tableau_server/logs にあります。
- アップグレード ログ。app-upgrade.log ファイルは /var/opt/tableau/tableau_server/logs にあります。
- Bash スクリプトのログ。ほとんどの Tableau Server bash スクリプトは /scripts ディレクトリ (/opt/tableau/tableau_server/packages/scripts.<version>) にあり、独自のログを生成します。これらのログは、スクリプトが実行される度に /var/tmp ディレクトリに保存されます。

Zip アーカイブ内のサーバー ログ ファイル

Tableau Server のログ ファイルを確認したり、サーバーに問題がありTableau サポートに送信したりすることが必要になる場合があります。tsm maintenance ziplogs コマンドを使用して、インストール内のすべてのノードからログ ファイルの圧縮アーカイブを作成します。デフォルトでは、Tableau Server のログ ファイルのアーカイブは logs.zip という名前の zip ファイルに集められますが、アーカイブを作成するときに異なる名前のファイルを指定することもできます。アーカイブをサーバーからローカル コンピューターにコピーし、そこで開くか、Tableau サポートに送信することができます。

アーカイブを解凍すると、クラスタ内の各ノードに対するディレクトリが作成され、そのディレクトリ内には各サービスまたは各プロセスのサブディレクトリが次の命名規則で作成されます。

<service_name>_<instance>.<version>.<build>

ノード上にサービスのインスタンスが複数ある場合、そのサービスにはインスタンスごとに 1 つずつ、複数のディレクトリが存在します。たとえば、ノードに 2 つの Backgrounders がある場合、次のようなディレクトリが表示されます。

```
backgrounder_0.<version>.<build>
backgrounder_1.<version>.<build>
```

zip ファイル内の具体的なディレクトリとログは、お持ちの Tableau Server のバージョンと、構成したプロセスによって異なります。ログを確認する可能性のあるプロセスまたはサービスの詳細については、Tableau Server プロセスを参照してください。

ログ ファイルのスナップショット (ログのアーカイブ)

Tableau Server には、アーカイブ目的でログ ファイルのスナップショットを生成する機能があります。定期的なサーバー メンテナンスの一環として古いログ ファイルのクリーンアップや削除を計画している場合、それを削除する前に、サーバー外の保存場所にログ ファイルをアーカイブすることを検討するかもしれません。

または、あるケースに関して Tableau サポートと協力している場合には、サポート エンジニアがサーバー ログ ファイルのスナップショットを要求する場合があります。

このトピックで説明する内容:

- ログ ファイルのスナップショットを作成する方法
- Tableau Server 管理ツールから Tableau サポートに直接スナップショットを送信する方法
- スナップショットをダウンロードする方法
- アーカイブ済みのログを削除する方法

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

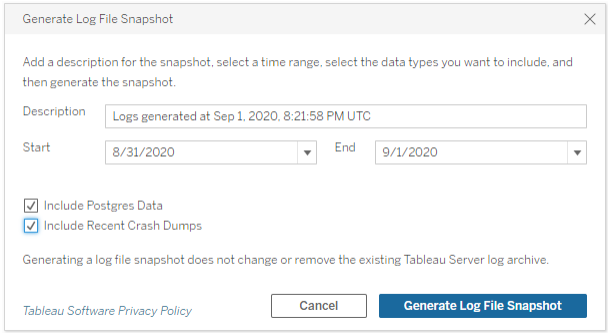
   https://<tsm-computer-name>:8850詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

2. **[メンテナンス]** をクリックします。

3. ログ ファイルのスナップショットを作成します。

a. サーバーの [メンテナンス] ページの [ログ ファイル] で、**[ログ ファイルのスナップショットを作成]** をクリックします。

   オプションのダイアログが表示されます。



b. [オプション] ページで、**[説明]**、含めるログ ファイルの期間の **[範囲]**、含めるログの種類 (オプション) など (**[Postgres データを含める]**、**[最近のクラッシュダンプを含める]**)、必要なオプションを入力するか選択し、**[ログ ファイルのスナップショットを作成]** をクリックします。

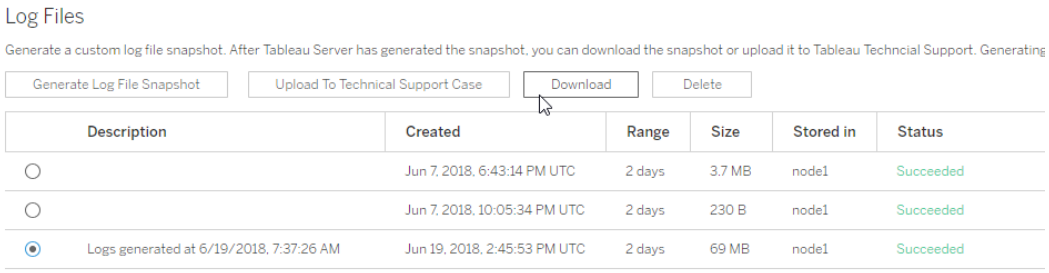
ログ ファイルのスナップショットが、TSM および Tableau Server がインストールされているコンピューターの固定された場所に保存されます。マルチノード インストールを行う場合、スナップショットはクラスタの最初のノードに保存されます。この場所は、`basefilepath_log_archives` 変数で指定します。

既定ではスナップショットは次に保存されます。

```
/var/opt/tableau/tableau_server/data/tabsvc/files/log-archives
```

現在の場所は `basefilepath.log_archives` 設定をクエリして確認でき、`basefilepath.log_archive` に新しい値を指定することで場所を変更できます。詳細については、tsm ファイル パスを参照してください。

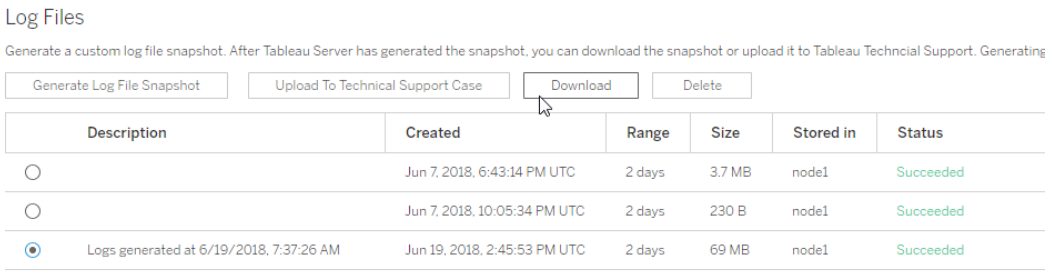
4. スナップショットの生成後は、それを選択してテクニカル サポートへのアップロード、ローカル マシンへのダウンロード、または削除を行えます。



5. スナップショットのローカル コピーをダウンロードするには、またはそれを削除するには、[ログファイル] でスナップショットを選択し、該当する **[アクション]** を選択します。

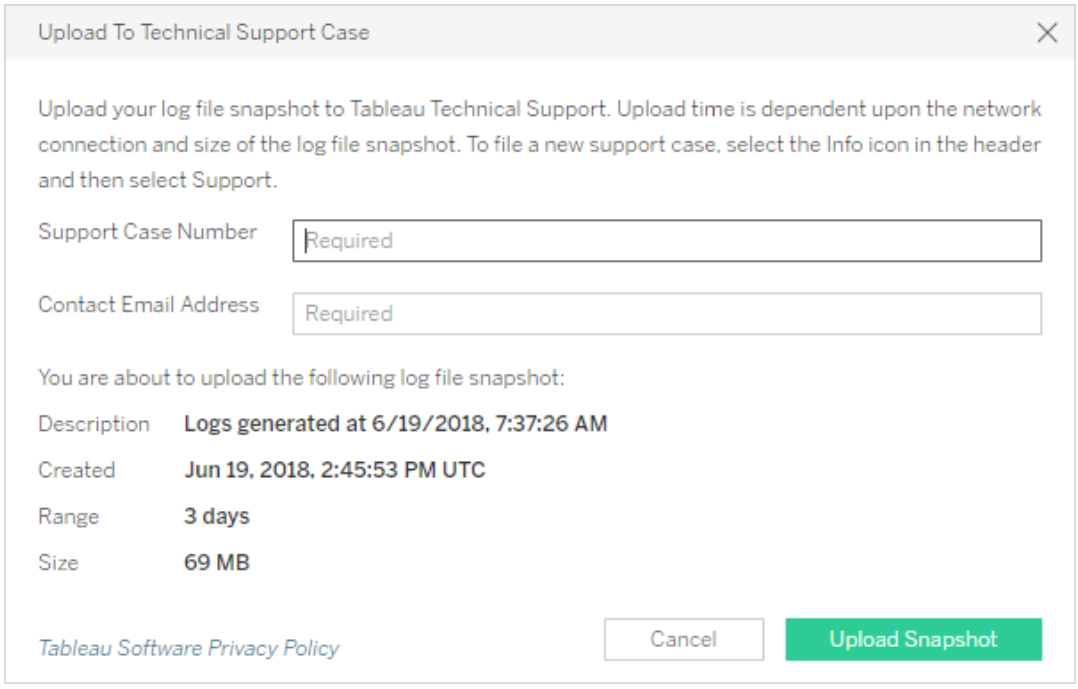
Tableau サポート用のログのスナップショットのアップロード

1. **[メンテナンス]** タブをクリックします。
2. 送信するスナップショットを選択します。



3. **[テクニカル サポート ケースへのアップロード]** をクリックします。
4. 表示されるダイアログで **[サポート ケース番号]** および **[連絡先 メール アドレス]** に入力し、

[スナップショットのアップロード] をクリックします。



ログ アーカイブを Tableau サポートに送信するその他の方法については、Salesforce ヘルプの「Tableau への大きなファイルの送信」を参照してください。

TSM CLI の使用

`tsm maintenance ziplogs` コマンドを使用して、Tableau Server ログ ファイルのスナップショットのアーカイブを作成します。

既定では、このコマンドによりすべてのログ ファイルを含む zip ファイルが作成されます。Tableau Server の分散インストールを実行している場合、初期ノードからこの手順を実行します。すべてのノードからのログが zip ファイルに含まれます。

> **注:** ziplogs コマンドを正常に実行できない場合は、Tableau Server ログを手動で圧縮できます。詳細については、Linux での Tableau Server のトラブルシューティングを参照してください。

ログ ファイルのスナップショットを作成するには、次の手順を実行します。

1. 最初のノードで、ターミナル セッションを開きます。
2. 次のコマンドを入力します。

   ```
   tsm maintenance ziplogs -l -f <filename>
   ```

   ここで、<filename> は作成する zip アーカイブ ファイルの名前です。スペースを含まない一意の名前を指定します。同じファイル名の ziplog が既存する場合は、コマンドに -o オプションを含めて強制的な上書き、既存ファイルの削除、または別の名前の指定を行わない限り、ファイルの作成に失敗します。

   スナップショットの時間範囲と、含めるログのタイプを指定することができます。たとえば、エラーが発生したタイミングがわかっている場合は、--startdate オプションと--enddate オプションを使用して、エラー発生前後の数時間からログをキャプチャします。

   ```
   tsm maintenance ziplogs -f <filename> --startdate "<mm/dd/yyyy H:mm>" --enddate "<mm/dd/yyyy H:mm>"
   ```

   詳細については、tsm maintenance ziplogs を参照してください。

   ログ ファイルのスナップショットが、TSM および Tableau Server がインストールされているコンピューターの固定された場所に保存されます。マルチノード インストールを行う場合、スナップショットは クラスタの最初のノードに保存されます。この場所は、basefilepath_log_archives 変数で指定します。

   既定では、ログ ファイルのスナップショットは次の場所に保存されます。

   ```
   /var/opt/tableau/tableau_server/data/tabsvc/files/log-archives
   ```

   basefilepath.log_archives 設定を照会すると、現在の場所を見つけることができます。

   ```
   tsm configuration get -k basefilepath.log_archive
   ```

   basefilepath.log_archiveをクリックし、新しい値を指定して場所を変更します。

```
tsm configuration set -k basefilepath.log_archive -v
"<drive>:/new/directory/path"
```

詳細については、tsm ファイル パスを参照してください。

Tableau Support のログ アーカイブの送信

カスタマー サポート ケースの一部としてログ ファイルを Tableau サポートへ送信できます (カスタマー サポート ケース番号が必要)。ログ ファイルを送信する前に、`tsm maintenance ziplogs` コマンドを使用して、ログ ファイルを単一の zip ファイル アーカイブにまとめます。

- ターミナル セッションで、次のコマンドを入力します。

  ```
  tsm maintenance send-logs -f <zip file name> -c <case number> -
  e <email address>
  ```

  ここで、`<case number>` はサポート ケース番号、`<email address>` はこのサポート ケースの連絡先メール、`<zip file name>` は .zip ファイル拡張子が付いたアーカイブのファイル名です。

ログ アーカイブを Tableau サポートに送信するその他の方法については、Salesforce ヘルプの「Tableau への大きなファイルの送信」を参照してください。

ロギング レベルの変更

既定では、Tableau サービス マネージャー (TSM) と Tableau Server は**情報**レベルでイベントのログを記録しています。さらに多くの情報を収集する必要がある場合 (たとえば、Tableau サポートと連絡している場合)、これを変更することができます。

ベストプラクティスとして、問題のトラブルシューティングを行っている場合以外は、ロギング レベルを引き上げないことをお勧めします。ロギング レベルをデバッグに設定する必要があるのは、特定の問題を調査する場合のみです。ログレベルを変更すると、次のような影響を受ける可能性があります。

- ロギング レベルを `debug` または `trace` に引き上げると、ログに記録される情報量が増加するため、パフォーマンスに大きく影響する可能性があります。問題を再現したら、ロギング レベルをリセットし、情報に戻してください。

- ロギング レベルを warn または error に設定すると、情報量が大幅に減ってしまうため、Tableau サポートに役立てることができません。

注: DEBUG レベルでログを記録する場合、Tableau の起動時に完全な環境情報が収集されます。これは、環境変数に機密情報がある場合、その機密情報がログに含まれる可能性があることを意味します。デフォルトの INFO レベルでログを記録する場合は、安全な環境情報のみが収集されます。

ロギング レベル

以下のロギング レベルは、ログに記録される情報量が増加する順に列挙されています。

- オフ
- fatal (致命的)
- error (エラー)
- warn (警告)
- info (the default) (情報 (既定))
- debug (デバッグ)
- trace (追跡)

ロギング レベルの変更

TSM とTableau Server プロセスのロギング レベルを、**tsm configuration set** コンフィギュレーション キーを使用して設定します。使用するキーは、TSM や Tableau Server のどのコンポーネントのロギング レベルを変更するかによって異なります。

動的ログ レベルの構成

動的構成は、バージョン 2020.2 で導入されました。この機能は、それ以降のリリースで拡張されています。1 つまたは複数のコンポーネントに対してログ レベルのみを変更する場合は、Tableau Server を再起動せずに変更できます。

各ログ レベルを動的に設定できるのは以下のバージョンからです。

- 2020.2 - tsm サービス (`tsm.log.level`) および制御アプリケーション サービス (`tsm.controllerapp.log.level`)。

- 2020.3 - バックグラウンダー (`backgrounder.log.level`)、クラスタ コントローラー (`clustercontroller.log.level`)、データ サーバー (`dataserver.log.level`)、ファイル ストア (`filestore.log.level`)、データ ソース プロパティ (`tdsservice.log.level`) および VizQL サーバー (`vizqlserver.log.level`)。
- 2020.4 - インタラクティブなマイクロサービス コンテナ (`tomcatcontainer.log.level`) およびアプリケーション サーバー (`vizportal.log.level`)。

ロギング レベルを変更するための設定キー

この表には、動的に設定できるキーとできないキーの両方が含まれています。

構成キー	影響するログの場所 (`/var/opt/tableau/tableau_server/data/tabsvc/logs/` で始まるパス)
`tsm.log.level` TSM のロギング レベルの変更: activationservice、appzookeeper、backuprestore、clientfileservice、databasemaintenance、licenseservice、siteimportexport、tabadminagent、tabadmincontroller、tabsvc	`/<service>/<service>_node<n>-<instance>.log` 例: `/clientfileservice/clientservice_node1-0.log`

`tsm.controlapp.log.level`
 	
TSM ロギング レベルの変更: 制御アプリケーション
	`/<service>/control_<service>_node<n>-<instance>.log`
	
	例: `/clientfileservice/control_clientservice_node1-0.log`
	
	`/filestore/control_filestore_node1-0.log`

`<process>.native_`	`/vizqlserver/*.txt`

`api.log.level` 有効なプロセス名: backgrounder、vizportal、vizqlserver、dataserver **注:** これらは動的に構成することはできません。	
`backgrounder.log.level` ロギング レベルの変更: バックグラウンダー	`/backgrounder/*.log`
`clustercontroller.log.level` ロギング レベルの変更: クラスタ コントローラー	`/clustercontroller/*.log`
`dataserver.log.level` ロギング レベルの変更: データサーバー	`/dataserver/*.log`
`filestore.log.level` ロギング レベルの変更: ファイル ストア	`/filestore/*.log`
`gateway.log.level` ロギング レベルの変更: ゲートウェイ コントロール プロセス	`/gateway/*.log`
`gateway.httpd.loglevel` **注:** バージョン 2021.3.0 での追加 ロギング レベルの変更: ゲートウェイ	`/gateway/*.log`
`hyper.log.level`	`/hyper/*.log`

ロギング レベルの変更: Hyper	
`tdsservice.log.level` ロギング レベルの変更: データ ソース プロパティ サービス	`/tdsservice/*.log`
`tomcatcontainer.log.level` マイクロサービスのロギング レベルの変更: インタラクティブなマイクロサービス コンテナーとインタラクティブでないマイクロサービス コンテナー	`/tomcatcontainer/*.log`
`vizportal.log.level` ロギング レベルの変更: アプリケーション サーバー	`/vizportal/*.log`
`vizqlserver.log.level` ロギング レベルの変更: VizQL サーバー	`/vizqlserver/*.log`

詳細については、tsm configuration set のオプションを参照してください。

動的に構成可能なロギング レベルのみを変更する場合は、サーバーを停止または起動する必要はありません (詳細については、上記の動的ログ レベルの構成を参照してください)。その他のロギング レベルを変更する場合は、Tableau Server を停止してからロギング レベルを変更し、その後サーバーを再起動する必要がある場合があります。この場合、プロンプトが表示されます。

Tableau Server の分散インストールを実行している場合、最初のノードからロギング レベルの設定を行います。

ロギング レベルを変更するには、次の手順を実行します。

1. (2020.2.0 以降で動的に構成可能なロギングのオプション) コマンドプロンプトを開き、次のように入力して Tableau Server を停止します。

   ```
   tsm stop
   ```

2. 「`tsm configuration set -k <config.key> -v <config_value>`」と入力し、ロギング レベルを設定します。

   ここで、`<config.key>` は上記の表に含まれるキーの 1 つで、`<config_value>` は有効なロギング レベルです。

   例:

   - `tsm configuration set -k backgrounder.native_api.log.level -v debug`
   - `tsm configuration set -k tsm.log.level -v debug`
   - `tsm configuration set -k tsm.controlapp.log.level -v debug`

3. 保留中の変更を適用するには、`tsm pending-changes apply` コマンドを実行します。

4. (サーバーが停止している場合のみオプション) 次のコマンドを実行して Tableau Server を起動します。

   ```
   tsm start
   ```

ロギング レベルのリセット

問題を再現し、問題に関連する情報を収集したら、パフォーマンスに影響が残らず、追加のディスク領域も使い切らないようにロギング レベルをリセットします。

適切なコマンドに `-d` オプションを使用して、ロギングレベルを既定 (情報) に戻します。レベルをリセットした後に保留中の変更を適用する必要があります。Tableau Server プロセスのロギング レベルをリセットする場合は、変更を行う前にサーバーを停止し、保留中の変更を適用してからサーバーを起動する必要がある場合があります。

例:

- `tsm configuration set -k backgrounder.native_api.log.level -d`
- `tsm configuration set -k tsm.log.level -d`

Tableau Server のインストールとアップグレードのトラブルシューティング

Tableau Server の一般的な問題を解決するには、このトピックの提案に従ってください。ステータスページに表示されるプロセス ステータスに基づいた追加のトラブルシューティングの手順については、サーバープロセスのトラブルシューティングを参照してください。

一般的なトラブルシューティングの手順

多くの Tableau Server の問題は、幾つかの基本的な手順で対応することができます。

1. Tableau Server を実行している各コンピューターで十分なディスク容量があることを確認します。ディスク容量が限られている場合、インストールの失敗、アップグレードの失敗、または Tableau Server の実行で問題が生じる可能性があります。
2. Tableau Server を再起動します。完全に開始していないプロセスに関連する問題は、Tableau Server を制御された方法で再起動することによって解決できます。Tableau Server を再起動するには、`tsm restart` コマンドを使用します。このコマンドでは、Tableau Server に関連付けられたすべてのプロセスを停止して、再起動します。
3. Tableau Server インデックスの再作成を実行します。インデックス作成に関連する問題は、Tableau Server インデックスの再作成を行うことによって解決できます。Tableau Server インデックスを再作成するには、`tsm maintenance reindex-search` コマンドを使用します。詳細については、下の Tableau index および Tableau Server (2023.3 以降) の再構築を参照してください。
4. Tableau Server が実行されている PC を再起動します。データソース接続に関連するような問題は、サーバー PC を再起動して解決できます。

Tableau Server のインストールに関する全般的な問題

インストール ログの場所

インストール ログである app-install.log は /var/opt/tableau/tableau_server/logs にあります。

アップグレード ログである app-upgrade.log は /var/opt/tableau/tableau_server/logs にあります。

インストールの試行が何度も失敗する

Tableau Server のインストールを試みて、インストールに失敗する場合、tableau-server-obliterate スクリプトを実行して Tableau をコンピューターからクリーンアップしない限り、その後インストールを試みても失敗する可能性が高くなります。

インストールの試行に失敗すると、コンピューターはその後の試行も失敗する状態となり、前回のインストール試行に直接は関係がないように見えるエラーが残る可能性があります。考えられるエラーの 1 つは以下です。

```
Enabling and starting all services
+ services=(appzookeeper* tabadmincontroller* tabsvc*
licenseservice* fnplicenseservice* tabadminagent*
clientfileservice*)
+ systemctl_user enable appzookeeper_0.service
'tabadmincontroller*' 'tabsvc*' 'licenseservice*'
fnplicenseservice_0.service 'tabadminagent*' 'clientfileservice*'
++ id -ru a_tabadminpoc
+ local unprivileged_uid=222954
+ su -l a_tabadminpoc -c 'XDG_RUNTIME_DIR=/run/user/222954
systemctl --user enable appzookeeper_0.service tabadmincontroller*
tabsvc* licenseservice* fnplicenseservice_0.service tabadminagent*
clientfileservice*'
Failed to execute operation: No such file or directory
```

この問題を修正するには、tableau-server-obliterate スクリプトを実行し、前回インストールを試みたときに残ったものをすべてクリーンアップしてから、コンピューターを再起動します。詳細については、tableau-server-obliterate スクリプトの実行を参照してください。

重要: 保持する Tableau (<file>.tsbak) のバックアップを作成 (新しいインストール場所に復元するなど) したら、そのファイルを別のコンピューターの安全な場所にコピーして、Tableau コンピューターをクリーンアップしても削除されないようにします。

ハードウェア要件のためインストールが失敗する

Tableau Server は、インストール先のコンピュータがハードウェアの最低要件を満たしていない場合はインストールできません。この要件は、Tableau Server をインストールしているすべてのコンピューターに適用されます。ハードウェアの最低要件の詳細については、Tableau Server の最小ハードウェア要件と推奨事項を参照してください。

CPU 要件が原因でインストールまたはアップグレード失敗する

バージョン 2020.4 0 以降、Tableau Server では、SSE4.2 および POPCNT 命令セットをサポートする CPU が必要です。これらの命令セットをサポートしていない CPU が搭載されているコンピューターでは、Tableau Server 2020.4.0 以降をインストールまたはアップグレードすることはできません。

新規インストール時、または既存のインストールのアップグレードの準備中に、次のエラー メッセージが表示されることがあります。

```
Your computer's processor doesn't meet the minimum requirements that
Tableau requires to install the software. If you are using a VM,
make sure Processor compatibility mode is off.
```

SSE4.2 および POPCNT 命令セットは 10 年以上にわたって使用されている一般的なセットであり、ほとんどの新しい CPU がサポートしていますが、仮想マシン (VM) に Tableau Server をインストールまたはアップグレードしようとしたときにプロセッサーの最小要件に関連するエラーが発生した場合は、プロセッサー互換モードが VM で有効になっている可能性があります。Tableau を VM に正常にインストールまたはアップグレードするには、プロセッサー互換モードがオフになっていることを確認してください。

Tableau Server のアップグレードに関する全般的な問題

アップグレードログの場所
アップグレードログである app-upgrade.log は /var/opt/tableau/tableau_server/logs にあります。

アップグレード後にマップが表示されないか、完全に表示されません

Tableau バージョン 2019.2 以降、マップのインターネットアクセス要件が変更されました。バージョン 2019.1.x 以前からバージョン 2019.2.x 以降にアップグレードしているときに、マップが期待どおりに表示されない場合は、お使いの環境がポート 443 で mapsconfig.tableau.com および api.mapbox.com にアクセスできるように構成されていることを確認してください。

バージョン 2019.1.x 以前では、maps.tableausoftware.com へのアクセスが必要でした。

インターネットアクセスの要件の詳細については、インターネットとの通信を参照してください。

アップグレードスクリプトエラー:"Tableau Server Version change validation failed. (Tableau Server のバージョン変更の検証に失敗しました。)"

アップグレード時に、以前のバージョンの scripts.<version_code> ディレクトリから upgrade-tsm スクリプトを実行すると、次のエラーが発生し、アップグレードが失敗します。

```
Tableau Server Version change validation failed.
Tableau Server <version> is already installed.
```

このエラーが発生した場合は、インストールしたバージョンの scripts.<version_code> ディレクトリに変更し、そこからスクリプトを実行します。

マルチノードをアップグレードし、追加ノードを初期化すると"Enter your credentials again (認証資格情報を再度入力してください)" というエラーで失敗する

Tableau Server のアップグレード時に追加ノードを初期化しようとすると、次のエラーが表示されます。

```
Enter your credentials again. The credentials you enter must
provide administrative access to the computer where you generated
the configuration file.
```

これは、ノードが接続できないか、最初のノードと通信していることを示します。これには以下の複数の理由が考えられます。

- 入力した認証資格情報が有効でないか、入力ミスがある。認証資格情報は、Tableau Server を最初にインストールしたコンピューターの管理パーミッションを持つユーザーの情報でなければなりません。bootstrap ファイルを作成したユーザーの認証資格情報を使用する必要はありませんが、これが確実に有効な認証資格情報となります。
- 追加しようとしているコンピューターのローカル ファイアウォールで最初のノードへの通信が許可されていない。詳細については、ローカル ファイアウォール構成を参照してください。

ディスク容量の不足によるアップグレードの失敗

Tableau Server のセットアップ プログラムを実行およびアップグレードするためのディスク容量が不足している場合、インストールは失敗します。必要なディスク容量は、リポジトリデータベースのサイズおよび抽出の数とサイズによって異なります。

ディスク容量を空けるには:

1. `tsm maintenance ziplogs` コマンドを使用してログのアーカイブのスナップショットを作成します。

   圧縮されたログ ファイルを作成した後、Tableau Server インストールとは別の安全な場所に保存します。

2. `tsm maintenance cleanup` コマンドを使って不必要なファイルをクリーンアップします。詳細については、不必要なファイルの削除を参照してください

RebuildSearchIndex ジョブでアップグレードが失敗する

バージョン 2020.1.x 以降、アップグレードの最後のステップで検索インデックスを再構築します。この時点ですべてのサービスはアップグレードされているため、このジョブが失敗した場合は、サーバーを手動でリセットできます。最初からやり直す必要はありません。

エラーは次のようになります。

```
An error occurred while rebuilding search index.
```

検索サーバーをリセットするには、以下を実行します。

1. 最初のノードで、ターミナル セッションを開きます。

   アップグレード スクリプトで新しいバージョンのシステム環境に更新するため、新しいのターミナル セッションである必要があります。

2. tsm maintenance reindex-search コマンド (2023.3 以降) または tsm maintenance reset-searchserver コマンド (2023.1 以前) を使用して、検索インデックスを再構築します。

2022.1 以降でアップグレードに失敗する

Tableau Server 2022.1 (またはそれ以降) をアップグレードした後、アップグレードプロセスの一部として Tableau Server のバックアップを復元すると、次のエラーが発生する可能性があります。

"Tableau Server は既定で新しいアイデンティティサービスのテーブルを使用するため、バックアップを復元できません。"

この問題は、Tableau Server 2022.1 (およびそれ以降) がバックアップで使用されるアイデンティティスキーマとは異なるアイデンティティスキーマを使用するために発生します。この問題を解決するには、「アイデンティティの移行に関する問題をトラブルシューティングする」を参照してください。

2020.4.0 以降でアップグレードが失敗する

バージョン 2020.4.0 以降では、Checkpoint Upgrade 機能を使用して、失敗したアップグレードを再試行できます。一般に、これは、Tableau Server ログ ファイルに慣れていて、ログを検索することをいとわない経験豊富なサーバー管理者や IT プロフェッショナルにとって役立ちます。この機能を使用すると、upgrade-tsm スクリプトを再実行できるので、失敗したすべてのアップグレードに役立ちます。スクリプトは最後に成功したステップから実行するため、時間を節約できます。経験のある人にとっては、ディスク領域の問題やアクセス権限の問題などを特定して修正し、アップグレードを再実行することができます。

バージョン 2020.4.0 以降へのアップグレードに失敗した場合は、次の手順でアップグレードを完了できるかもしれません。

- upgrade-tsm スクリプトをもう一度実行します。アップグレードの失敗は、アップグレードプロセス中にタイムアウトが発生した結果である場合があり、スクリプトを再実行すると、断続的または偶発的なタイミングの問題から抜け出せる可能性があります。これは、安全で簡単に実行できる手順です。スクリプトを再実行しても害はありません。最悪の場合、アップグレードは同じ時点で失敗しますが、その前の手順を実行する必要はありません。

  スクリプトは \scripts ディレクトリにあります:

  ```
  opt/tableau/tableau_server/packages/scripts.<version_
  code>/upgrade-tsm
  ```

  upgrade-tsm スクリプトを再実行しても Tableau Server のアップグレードに成功せず、Tableau Server のログにも問題がない場合は、さらに次のトラブルシューティングの手順を実行します。

- コマンド画面でスクリプトの出力を確認します。エラー メッセージは、アップグレードの失敗の原因を特定し、問題を解決する方法のアイデアを得るのに役立ちます。

- app-upgrade.log ファイルを確認します。コマンド ラインに表示されるエラーは app-upgrade.log ファイルにも表示され、多くの場合、より詳細に表示されます。

- tabadmincontroller.log ファイルを確認します。アップグレードの問題を上記の 2 つのインスタンスで簡単に識別できない場合は、ジョブに問題がある可能性があります。tabadmincontroller.log ファイルには、問題の診断に役立つ詳細情報が含まれている場合があります。

  注: ログ ファイルの場所については、Tableau Server ログとログ ファイルの場所を参照してください。

バックアップ/復元ファイルの場所にパーミッションの問題があるため、アップグレードに失敗する

2022.1.0 より前のバージョンの Tableau Server では、バックアップ/復元ファイルのファイルの場所に正しいパーミッションがない場合、アップグレード スクリプトは失敗し、バックアップ ファイルを読み取れない、またはリポジトリを復元できないというエラーが発生します。

バージョン 2022.1 以降、アップグレード スクリプトは、アップグレードを開始する前に、バックアップ/復元ファイルのファイルの場所のパーミッションを確認します。これにより、新しいバージョンの Tableau Server へのアップグレード中に、その場所からのファイルの書き込みおよび読み取りを実行できます。

エラーは次のようになります。

```
The tableau user does not have permission to read the backup file:
<backup/restore basefilepath>.

Repository restore failed.
An error occurred during installation.
An error occurred while restoring repository.
```

TSM がバックアップと復元に使用する場所は、basefilepath.backuprestore 構成キーによって定義され、既定によりインストール プログラムが正しいパーミッションを使用して設定されますが、これらは組織の IT ルールによって影響を受けたり、自分で作成した場所に変更した場合に影響を受けたりする可能性があります。2022.1 以降で使用可能な新しいコマンドを使用すると、ファイルの作成直後にバックアップ/復元ファイルの場所に関するパーミッションを確認して、パーミッションに関連する問題を回避することができます。このコマンドの詳細については、「tsm maintenance validate-backup-basefilepath」を参照してください。

バックアップ/復元のファイル パスの詳細については、「tsm ファイル パス」を参照してください。

アップグレードは成功したが、パブリッシュ済みデータ ソースにアクセスできない

限定的な特定のシナリオでは、Tableau Server をバージョン 2021.3 から 2023.1 または 2023.3 の初期バージョンにアップグレードした後、既存のパブリッシュ済みデータ ソースへ接続または更新しようとすると、次のエラーで失敗します。

```
java.io.FileNotFoundException: Unable to fetch data from any other
host. This may indicate a lost or invalid folder.
```

これは次の場合に発生する可能性があります。

1. かつてバージョン 2021.3.x であった Tableau Server のインストールをアップグレードしようとしている (現在 2021.3 を実行しているか、以前に 2021.3 から 2022.x バージョンにアップグレードした場合が考えられます)

   および

2. そのインストールを 2023.1 または 2023.3 の初期バージョンにアップグレードしようとしている

影響なし

以下の状況では問題はありません。

- 2021.3 *以外からのアップグレードパス*
- 2023.1 または 2023.3 以外へのアップグレードパス
- 2023.1 および 2023.3 のすべての新規インストール

詳細

2024 年 9 月 16 日現在、問題のあるバージョンはすべてダウンロードサイトから削除されています。バージョン 2023.1.x または 2023.3.x にアップグレードする必要がある場合は、メンテナンス リリース バージョン 2023.1.16 以上、または 2023.3.9 以上にアップグレードしてください。

この問題の詳細については、「既知の問題」を参照してください。

共通設定のインポートに関する問題

設定ファイルをインポートすると、サービスが欠落しているために "not present on any node (ノードに存在しません)" 検証エラーが発生する

Tableau Server の新しいバージョンをインストールして、以前のバージョンから設定ファイルをインポートしてアップグレードする場合、`tsm settings import` コマンドの実行時にトポロジ検証エラーが発生する可能性があります。

これは、古いバージョンの Tableau Server から設定ファイルをエクスポートして新しいバージョンにインポートする際に、Tableau の 2 つのバージョン間で新しいサービスが追加されている場合に発生する可能性があります。

次のようなエラーが発生します (特定のサービスが異なる場合があります)。

```
>tsm settings import -f 20183-export.json

Pending topology set.
There are 1 topology validation errors/warnings.

Service 'elasticserver' is not present on any node in the cluster.
Service: Elastic Server
```

この問題を解決するには、欠落しているサービスを Tableau Server に追加します。

1. 検証エラーを生成したサービスの場合、インスタンス数が 1 のサービスを追加します。

   たとえば、Elastic Server がクラスタに存在しない場合は、検証エラー メッセージの最初の行に表示されるサービス名を使用して、プロセス インスタンス数を 1 に設定します。

   ```
   tsm topology set-process -n node1 -pr elasticserver -c 1
   ```

   エラーが発生した各サービスに対してこの手順を繰り返します。

2. 警告やエラーが表示されなくなると、保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

これで、設定が正常にインポートされるはずです。

設定ファイルをインポートすると"configuration value you specified does not match" (指定した構成値が一致しません) エラーが発生する

Tableau Server の新しいバージョンをインストールし、以前のバージョンから設定ファイルをインポートすると、`tsm settings import` コマンドの実行時に構成検証エラーが発生する場合があります。Tableau から削除された構成値が設定ファイルに含まれている場合、これらのエラーが発生する可能性があります。

次のようなエラーが発生します (構成キーが異なる場合があります)。

```
>tsm settings import -f 20183-export.json
Configuration error: At least one configuration value you specified
does not match a known configuration key. This applies to the
following keys: '[features.TsmConfigFileService]'
Use this parameter to override unknown key error: --force-keys
```

この問題を解決するには、インポートする設定ファイルを編集して、エラー内にある構成キーへの参照を削除します。

1. JSON 設定ファイルをコピーし、バックアップ用にコピーを保存します。
2. プレーンテキストエディターで JSON 設定ファイルを開きます。
3. キーを含む行全体を見つけて削除します。この例では、キーを含む行は `features.TsmConfigFileService` です。

   ```
   "configKeys" : {
     "config.version" : 19,
     "tabadmincontroller.port" : "8850",
     "endpoints.enabled" : false,
     "endpoints.health.enabled" : true,
     "features.TsmConfigFileService" : true,
     "tableau_projects.language" : "en",
   ```

   上記のコードは、エクスポートされた設定ファイルの小さなセクションの例であり、ファイルの内容全体を表すものではありません。
4. 設定ファイルを保存し、インポートし直します。

トポロジ検証に関連する追加のエラーが発生する場合があります。これらのエラーを解決する方法の詳細については、上記の設定ファイルをインポートすると、サービスが欠落しているために "not present on any node (ノードに存在しません)" 検証エラーが発生するを参照してください。

エラー:「調整サービスのインスタンスを直接変更することはできません」

このエラーは、次の 2 つの状況で発生する可能性があります。

- Tableau Server 設定ファイルを、設定ファイルのとは異なる調整サービス トポロジ持つインストールにインポートする場合
- `tsm topology set-process` コマンドを使用して調整サービスを構成しようとする場合

設定ファイルをインポートした後にこのエラーが表示される場合:

Tableau Server 設定ファイルには、ターゲット サーバーのとは異なる調整サービス トポロジがあります。これは、新しいバージョンをインストールし、以前のバージョンの設定ファイルをインポートすることによって Tableau Server をアップグレードすると、発生する可能性があります。ターゲット サーバーに調整サービス アンサンブルを明示的に展開していない場合、そのサーバーの最初のノードには調整サービスの単一のインスタンスが含まれます。

このエラーを修正するには、コマンド ラインから不一致を修正するか、設定のインポート ファイルを編集します。保留中の変更をすべて破棄して、インポート ファイルの設定に一致するようにターゲット コンピューターに調整サービスを展開し、設定ファイルをもう一度インポートすることもできます。

コマンド ラインからの不一致を修正するには、エラーが発生する各ノードに対して、tsm topology set-process コマンドを使用して、調整サービスのインスタンス数を元に戻します。

1. `tsm pending-changes list` コマンドを実行します。出力には、変更が加えられているノードが表示されます。
2. 調整サービスのカウントが変更された単一のノードまたは複数のノードを見つけます。

   たとえば、設定ファイルでは node2 上に調整サービス インスタンスが存在しているものの、ターゲット システムではそのノード上に調整サービス インスタンスがない場合、node2 のカウントでは、設定ファイルのインポートによって 0 から 1 に変更されたことが表示されます。

   ```
   C:\Windows\system32>tsm pending-changes list
   Configuration
   There are no pending configuration changes.
   Topology
   node2:
              Coordination Service
   ```

```
New Instance Count:1
Old Instance Count:0
```

3. `tsm topology set-process` コマンドを使用して、カウントを"古いインスタンス" 値に戻します。

   上記の例では、次のようになります。

   ```
   tsm topology set-process -n node2 -c 0 -pr "Coordination
   Service"
   ```

4. 変更された調整サービス インスタンスのカウントをリセットしたら、保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

調整サービスのプロセス カウントを手動で設定する際にエラーが表示される場合:

このエラーは、調整サービスを管理するための `tsm topology` コマンドではなく、`tsm topology set-process` コマンドを使用して、コーディネーション サービスを直接更新しようとした場合にも発生します。以下を試した場合:

1. 保留中の変更を破棄するには、`tsm pending-changes discard` コマンドを使用します。
2. 調整サービスを構成するための正しいコマンドを使用します。詳細については、調整サービス アンサンブルの展開を参照してください。

Tableau Server の起動

Tableau Server は完全に起動したことを確認できません

Tableau Server はスタートアップ時にすべてのコンポーネントが正常に起動したことを確認できないと報告する場合があります。次のメッセージが表示されます。"サービスのすべてのコンポーネントが正常に起動したことを確認できませんでした。"

起動後にこのメッセージが表示された場合、`tsm status -v` コマンドを使用して Tableau Server が正常に実行されていることを確認します。

ステータスが実行中 ("Status: RUNNING") であることを示している場合、サーバーは正常に起動されており、メッセージを無視することができます。ステータスが DEGRADED または STOPPED の場合は、次のセクションの「Tableau Server が起動しない」を参照してください。

Tableau Server が起動しない

Tableau Server が開始しないか機能が低下した状態で実行されている場合、コマンドプロンプトから`tsm restart` コマンドを実行します。これによって実行中のすべてのプロセスをシャットダウンして、Tableau Server を再起動します。

Tableau index および Tableau Server (2023.3 以降) の再構築

インデックス作成/検索サーバーのインデックスを再構築することによって解決できる問題

再構築が必要になるのは、以下のようなインデックスです。

- ユーザーがログインを試みるとサイトのリストが空になっている
- ユーザーがプロジェクトの選択を試みるとプロジェクトのリストが空になっている
- コンテンツがない (ワークブック、ビュー、ダッシュボード)
- 予期しない、または不正確なアラート (たとえば、"更新の失敗" 抽出を含まないワークブックのアラート)

このような状態になっている場合は、`tsm maintenance reindex-search` コマンドを使用して、インテックス作成/検索サーバーのインデックスをリセットして再構築します。

Tableau Server の検索 & 参照 (2023.1 以前の場合) のインデックス再作成

検索とブラウズの再インデックスによって解決する他の問題

再構築が必要なインデックスの症状には次が含まれます。

- ユーザーがログインを試みるとサイトのリストが空になっている
- ユーザーがプロジェクトの選択を試みるとプロジェクトのリストが空になっている
- コンテンツがない (ワークブック、ビュー、ダッシュボード)
- 予期しない、または不正確なアラート (たとえば、"更新の失敗" 抽出を含まないワークブックのアラート)

これらの症状が見られる場合、`tsm maintenance reset-searchserver` コマンドを使用して検索とブラウズインデックスを再構築します。

Tableau Server のライセンス認証

Tableau Server のライセンス認証の失敗

一部のインスタンスの Tableau Server のライセンス認証に失敗する場合があります。さまざまなエラー メッセージがありますが、中でも一般的なメッセージは次のとおりです。

- `An error has occurred`

また、より具体的なメッセージの例を以下に挙げます。

- `Function flxActCommonLicSpcPopulateFromTS returned error 50030, 71521,`
- `No license found for 'Tableau Server'`

この問題を解決するには、以下の解決方法を記載された順に試してください。

ライセンシング サーバーへアクセスできることを確認する

Tableau のライセンス発行サービスは、2018 年 10 月 6 日に新しいデータセンターに移行しました。つまり、licensing.tableau.com にアクセスするために特別な構成を必要とする (静的 IP の許可リストなど) 環境では、Tableau プロダクト キーをライセンス認証、更新、または認証解除する前にアップデートが必要となります。

アクセスをテストするには、ブラウザーにライセンスサーバーの URL とポートを入力します。

`https://licensing.tableau.com:443`

および:

`https://atr.licensing.tableau.com/_status/healthz`

サーバーにアクセスできる場合は、最初のサーバーに対して "テスト成功" メッセージが表示され、2 番目のサーバーに対して "OK" というメッセージが表示されます。

Tableau Server は、ライセンス発行の目的で、次のインターネット上の場所に接続する必要があります。

- atr.licensing.tableau.com:443
- licensing.tableau.com:443
- register.tableau.com:443
- o.ss2.us
- s.ss2.us
- crt.rootca1.amazontrust.com
- crt.sca1b.amazontrust.com
- crt.sca0a.amazontrust.com
- crt.sca1a.amazontrust.com
- crt.sca2a.amazontrust.com
- crt.sca3a.amazontrust.com
- crt.sca4a.amazontrust.com
- *.digicert.com
- ocsp.*.amazontrust.com
- crl.*.amazontrust.com
- crt.rootg2.amazontrust.com

上記のドメインへのリクエストは、ポート80 またはポート443 で行われることがあります。ポート80 は、証明書の検証 (失効、証明書チェーンなど) に使用されます。ポート443 は SSL 接続に使用されます。

`ocsp.*.amazontrust.com` および `crl.*.amazontrust.com` ドメインへのリクエストは、証明書失効に関する情報であるため、Amazon によって管理されます。詳細については、「ACM certificate characteristics」を参照してください。Tableau を実行しているコンピューターの証明書信頼ストアに Amazon ルート証明書をインストールすることをお勧めします。Amazon ルート証明書をダウンロードしてインストールする方法については、Amazon Trust Services Web サイトの「認証局」を参照してください。

日付と時刻を確認します

最初の Tableau Server コンピューター上の日付と時刻が正しいことを確認してください。時計が現在の日付より早い日時に設定されていると、Tableau Server は起動できません。

プロダクトキーを強制的に再度読み取る

1. 最初の Tableau Server コンピューターで、sudo アクセス権を持つユーザーとしてログオンします。

   Tableau Server の bin ディレクトリに移動します。既定では、次になります。
2.

   ```
   /opt/tableau/tableau_server/packages/bin.<version_code>/
   ```

3. 次のコマンドを入力します。

   ```
   tsm stop

   ./lmreread
   tsm start
   ```

信頼されているストレージのコンテンツを Tableau サポートに送信する

FlexNet ライセンス発行サービスがインストールされ、実行中で、エラーが表示されていない場合、Tableau プロダクトキー情報に問題がある可能性があります。この問題を解決するため、次の手順を完了して、信頼されているストレージに格納されているキー情報のファイルを作成します。

1. 最初の Tableau Server コンピューターで、sudo アクセス権を持つユーザーとしてログオンします。

2. 次のコマンドを入力します。

   ```
   serveractutil -view > <machine_name>-LicResults.txt
   ```

   これにより、現在のディレクトリに <machine_name>-LicResults.txt ファイルが作成されます。ログ ファイルの場所に対する書き込み権限がなく、エラーが表示される場合は、ファイルを作成する権限がある場所に変更してから、コマンドをもう一度実行します。

3. Tableau サポート (http://www.tableau.com/ja-jp/support/request) に連絡し、作成した <machine_name>-LicResults.txt ファイルを含めます。

tabcmd のインストールの問題

tabcmd を個別にインストールする

tabcmd は、Tableau Server をインストールするときに最初の Tableau Server ノードに自動的にインストールされますが、別のコンピューターで実行する場合は、tabcmd を個別にダウンロードしてインストールする必要があります。詳細については、tabcmd のインストールを参照してください。

Linux で tabcmd をインストールときの問題

Tabcmd を正しく実行するには、Java11 が必要です。RHEL のようなシステムでは、tabcmd のインストール時に依存関係としてインストールされます。Debian のようなシステムでは、Java 11 がまだインストールされていない場合は個別にインストールする必要があります。

2022 年 7 月の時点で、Debian ディストリビューションはサポートされなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください。

Java がインストールされていない

tabcmd をインストールするときに次のようなエラーが表示される場合は、Linux コンピューターに Java 11 がインストールされていることを確認します。

```
Cannot find 'java' in your PATH. Install 'java' and make sure it is
in your PATH to continue.
```

Java のバージョンが正しくない

次のようなエラーが表示された場合は、Java 11 がインストールされていることを確認します。

```
Exception in thread "main" java.lang.UnsupportedClassVersionError:
com/tableausoftware/tabcmd/Tabcmd : Unsupported major.minor version
52.0
```

または

```
*** Uncaught exception NoClassDefFoundError:
javax/xml/bind/JAXBException
*** See the logs for the stacktrace.
```

systemd ユーザー サービスのエラー

新規インストール中に `initialize-tsm` をアップグレードしたり実行したりすると、次のいずれかのエラーが表示されることがあります。

- 「Failed to get D-Bus connection: No such file or directory (D-Bus 接続に失敗しました: そのようなファイルやディレクトリはありません)」
- 「$XDG_RUNTIME_DIR not found ($XDG_RUNTIME_DIR が見つかりません)」
- 「systemd unit user@<userID> is not running. (Systemd ユニットuser@<userID> が実行されていません。)」「Check /var/log/messages or /var/log/syslog. (/var/log/messages or /var/log/syslog を確認してください。)」

背景

2018.1 時点で、Tableau Server は `systemd` ユーザー サービスを使用してプロセスを管理しています。これは、特権のないユーザーとして実行される `systemd` プロセスがあることを意味します。既定では、Tableau Server のセットアップで `tableau` という名前の特権のないアカウントが作成されます。Tableau Server プロセスは `systemd` プロセスからスポーンされたもので、ルートとして実行されるシステム全体の `systemd` プロセスではありません。

重要: このトラブルシューティングに関する注記は、主に RHEL 7 ベースのディストリビューションに適用されます。ただし、これらのエラーのいずれかが表示される場合、Ubuntu ディストリビューションにも同じ問題が存在する可能性があります。

systemd ユーザー サービスは、通常の systemd プロセス マネージャーほど一般的には使用されていません。Red Hat では、RHEL 7 (CentOS、Oracle Linux 7、Amazon Linux 2 など、RHEL から取得されるすべてのディストリビューション) の systemd ユーザー サービスが無効になっています。ただし、RedHat では、systemd ユーザー サービスが再度有効になっている限り、このユーザー サービスを実行している Tableau に対応することを保証しています。

Tableau Server on Linux 10.5 からのアップグレード

Tableau Server 10.5 からアップグレードする場合は、特権のないユーザーに有効なシェルとホーム ディレクトリがあることを確認します。Tableau Server 10.5 では、シェルが /sbin/nologin に設定され、ホーム ディレクトリが「/」の特権のないユーザーが意図的に作成されました。特権のないユーザーが initialize-tsm によって作成された場合は、2018.1 へのアップグレード時に Tableau でシェルとホーム ディレクトリが更新されます。

ただし、10.5 の初期インストール時に特権のないユーザーを作成した場合は、アップグレードしようとするとエラーが発生します。

これを修正するには、シェルを /sbin/nologin に、ホーム ディレクトリを「/」に設定してから、再度アップグレードを実行する必要があります。

新規インストール時に発生したエラーのトラブルシューティング

systemd ユーザー サービスが実行中であることを確認します。

ps -fww $(pgrep -f "systemd --user") コマンドを実行して確認します。

systemd ユーザー サービスが実行されていない場合は、何らかが起動を妨害しています。

トラブルシューティングを行うには、次のリストに従います。

- /var/log/messages でログを確認します。
- journalctl を実行します。
- PAM 構成が含まれるように実行したカスタマイズで pam_systemd.so が削除されていないことを確認します。

  RHEL 7 PAM ファイルの場合は、/etc/pam.d/system-auth で次の行が欠落しています。

  ```
  -session optional pam_systemd.so
  ```

  Tableau Server が機能するには、この行を追加し直す必要があります。

- -session optional pam_systemd.so が PAM 構成に存在しているがユーザー サービスを開始できず、エラー メッセージ $XDG_RUNTIME_DIR not found が /var/log/messages に表示される場合は、環境変数の設定を試行しないでください。このシナリオのエラーは正確ではありません。

  実際のエラーは、PAM モジュール pam_systemd.so がユーザー セッションを割り当てることができないというものです。既定の構成では、pam_systemd.so からのエラー メッセージが表示されないようになっています。エラー メッセージとデバッグ メッセージを表示するには、/etc/pam.d/system-auth の行で-session optional pam_systemd.so を session optional pam_systemd.so debug に変更します(先頭のハイフンを削除するとエラー メッセージが表示され、debug を追加すると詳細なログ記録が表示されます)。

  これで、/var/log/messages、/var/log/secure、/var/log/audit/audit.log ファイルを確認するとエラー メッセージが表示されるようになります。

例

次のエラー メッセージが表示される場合があります。

```
systemd-logind: Failed to mount per-user tmpfs directory
/run/user/0: Permission denied
```

この場合、オンラインでエラーを検索するとRedhat の KB 記事 (https://access.redhat.com/solutions/2460611) が見つかります。

この記事では、`sudo yum update selinux-policy` を実行して `selinux-policy` パッケージを更新するよう推奨されています。

場合によっては、バージョン 3.12.X から 3.13.X にアップグレードすると `$XDG_RUNTIME_DIR not found` に関する問題が解決することがあります。パッケージの更新後は、`sudo reboot` を必ず実行してください。

サービス障害によるジョブの失敗に関するトラブルシューティング

Tableau Server バージョン 2021.1 以降、サービス障害が原因でジョブが失敗した場合に、一連の新しいエラー メッセージが表示されるようになりました。このトピックでは、メッセージとその意味について説明します。

エラー メッセージは次の形式で表示されます。

```
<nodeId>
<service>_<instanceId>.<version>: <error>
```

エラーには次の 4 つのカテゴリがあります。

- **ステータスが見つかりません -** サービスがステータスを報告できない場合 (たとえば、ノードがダウンしている場合、tabadminagent がステータスを報告できない場合、またはサービスに障害が発生して依存サービスをインストールできなくなった場合) に、"ステータスが見つかりません" というエラーが表示されます。
- **構成の更新に失敗しました -** サービスが構成ファイルを更新できない場合、"構成の更新に失敗しました" というエラー表示されます。このエラーは、アップグレード中に、保留中の変更を新しいサービスに適用しようとしたときに発生します。エラーの詳細については、サービスのコントロール アプリログ `<dataDir>\tabsvc\logs\<service>\control-<service>_<nodeId>-<instanceId>` を参照してください。
- **要求された状態に到達できませんでした。現在の状態:<currentState> -** サービスをインストール/削除/開始/停止できない場合、"要求された状態に到達できませんでした" というエラーが発生します。このエラーは、新しいサービスがインストールされ、古いサービスが削除されているアップグレード中に発生します。<currentState> には次の状態オプションがあります。DEPLOY_FAILED, INSTALL_FAILED, DISABLE_FAILED, ENABLE_FAILED,

CONFIGURE_FAILED, UNINSTALL_FAILED, REMOVE_FAILED最も一般的なものは、INSTALL_FAILED、UNINSTALL_FAILED、および REMOVE_FAILED です。エラーの詳細については、サービスのコントロール アプリログ `<dataDir>\tabsvc\logs\<service>\control-<service>_<nodeId>-<instanceId>` を参照してください。

- **開始/停止に失敗しました。現在のステータス: <currentStatus>** - このエラーは、実行する必要のあるサービスが停止している場合、または停止する必要のあるサービスが引き続き実行されている場合に発生します。<currentStatus> の値には、ACTIVE、BUSY、PASSIVE、UNLICENSED、DOWN、STATUS_UNAVAILABLE、および DEGRADED などがあります。最初の 3 つ (ACTIVE、BUSY、PASSIVE) は、"実行中" のステータスと見なされます。最後の 4 つは "停止" ステータスです。エラーの詳細については、メインのサービスログ`<dataDir>\tabsvc\logs\<service>\<service>_<nodeId>-<instanceId>.log` を参照してください。

エラー メッセージの例は次のとおりです。

```
This job failed due to unexpected error:
'ServiceOperationTimeoutException'
One or more services failed to reach their expected state.
node1:
     vizportal_0.2021.4.0.0: Failed to reach requested state.
Current state: INSTALL_FAILED
```

バージョン 2021.3 以降、コントロールアプリログで検出されたエラーと一致する 2 番目と 3 番目のエラー タイプに追加のエラー メッセージが追加されました。

サーバーへのサインインに関する問題のトラブルシューティング

Tableau サービス マネージャー (TSM) と Tableau Server にはいくつかの異なるサインイン オプションがあります。

- **TSM** — TSM にサインインできない場合は、TSM がインストールされている PC に対して管理権限を持つユーザーの認証資格情報を使用していることを確認してください。Tableau Server 管理者の場合もありますが、そうでない場合もあります。この点は、サインイン先が Web UI か CLI かに関わらず適用されます。詳細については、「Tableau サービス マネージャーの Web UI へのサインイン」を参照してください。

- **Tableau Server —**

  - 管理者: 管理者として Tableau Server にサインインしている場合、Tableau Server で管理者の役割を持つユーザーの認証資格情報を使用する必要があります。初めて Tableau をインストールする場合は最初の管理者を作成しますが、Tableau のインストールが完了すると、実行中には他のユーザーを管理者として追加できます。詳細については、Tableau Server 管理者エリアへのサインインを参照してください。

  - 管理者以外のユーザー: ユーザーとして Tableau Server にサインインしている場合、Tableau Server に追加されているユーザーの認証資格情報を使用する必要があります。詳細については、「Tableau Server または Tableau Cloud へのサインイン」を参照してください。

    > **注:** 有効な認証資格情報を持つユーザーが Tableau Server にサインインできない場合、保留中の変更を適用していないノードを追加していないことを確認してください。保留中の新しいノードがある場合、Tableau Server にサインインできないことがあります。

シナリオのトラブルシューティング

ライセンス発行のトラブルシューティング

このトピックでは、Tableau Server のライセンス発行に関連する問題のトラブルシューティングの指示を記載しています。

ライセンスのないサーバーの処理

Tableau では 2 つのライセンス発行モデルを提供しています。1 つはロールベース モデルで、もう1つはコアベース モデルですロールベースおよびコアベースのライセンス発行の詳細については、ライセンス発行の概要を参照してください。

ロールベース ライセンスでは、アクティブな各ユーザー アカウントがライセンスの対象となる必要があります。ロールベース ライセンスでは、容量が定義されているか、許可されるユーザー数が定義さ

れています。各ユーザーにはサーバー上で一意のユーザー名が割り当てられ、サーバーに接続するときにそのユーザー名で識別する必要があります。

コアベースライセンス発行では、システム内のユーザーアカウント数に制限はありませんが、Tableau Serverが使用できるプロセッサーコアの数が制限されます。Tableau Server を 1 つまたは複数のマシンにインストールしてクラスタを作成できますが、すべてのマシンの合計コア数がライセンスでカバーされているコア数を超えないこと、および特定のマシン上のコアがすべてライセンスでカバーされていることという制限があります。

ライセンスを解除されたユーザーベースのサーバー

サーバーがロールベースライセンスを解除される最も一般的な理由は、プロダクトキーまたはメンテナンス契約の有効期限切れです。

ライセンスを解除されたコアベースのサーバー

コアベースのサーバーがライセンスなしになることには、さまざまな理由が考えられます。たとえば、プロダクトキーが有効期限切れになる場合や、ライセンスプロセスを実行中の Tableau Server ノードで、ライセンス マネージャー サービスを実行中の Tableau Server ノードに連絡を取ることができない場合などです。ライセンスプロセスの詳細については、Tableau Server プロセスを参照してください。

サーバーのライセンスが解除されると、サーバーを起動または管理できません。ただし、tsm licenses コマンドを使用してライセンスを管理できます。

ライセンスを解除されたサーバー管理者

すべての Tableau Server 管理者には、ユーザーライセンスが必要です。Tableau Server 管理者は、使用可能な最高のロールを常に使用します。Creator プロダクトキーが認証されると、Tableau Server 管理者がこのロールを担います。Tableau Server で使用可能な最高のロールが Explorer の場合、サーバー管理者は Explorer ロールを担います。Creator ライセンスがサーバーに追加されると、Explorer ライセンスを使用している既存のサーバー管理者アカウントは自動的に Creator ライセンスを使用するように変換されます。

TSM 管理者アカウントにはライセンスは必要ありません。

サーバー管理者が使用しているライセンスの有効期限が切れると、アカウントはライセンスを解除され、サインインできなくなります。

サーバーの管理者のライセンスの有効期限を確認します。

- tsm licenses list を実行します。
- Tableau カスタマーポータル に表示される日付とその日付を比較します。
- ポータルに予定されている日付が表示されない場合は、カスタマー サクセス に連絡してください。
- ライセンスを更新するには、Tableau の更新 Web ページにアクセスしてください。
- tsm licenses activate コマンドを実行して、管理者アカウントの新しいライセンスをアクティブ化します。

TSM の日付がポータルの日付と一致し、次の更新操作が失敗した場合は、Tableau サポート にお問い合わせください。

管理者アカウントのライセンスの有効期限が切れている場合、または間もなく期限切れになる場合は、アカウントの新しいライセンスをアクティブ化する必要があります。または、管理者以外のユーザーのライセンスを解除して、サーバー管理者アカウント用にライセンスを解放することもできます。

Tableau Server 管理者が Creator、Explorer、または Viewer ライセンスを使用しており、それらのライセンスの有効期限が切れる場合、利用可能であれば同じ種類の別のライセンスを使用します。ライセンス シートが利用できない場合、ユーザーは「ライセンスなし」になります。

> **重要**: 新しいライセンスをアクティブ化するか、サーバー管理者アカウントのサイト ロールを転送するまで、Tableau Server を再起動しないでください。

ロールベース ライセンス発行のトラブルシューティング

このセクションでは、ロールベースの Viewer、Explorer、および Creator ライセンスを Tableau Server または Tableau Cloud に追加する場合、またはこれらのライセンスの有効期限が切れる場合に起こる問題の解決に関する情報を提供します。利用可能な最高のライセンス タイプは

Creator で、次に Explorer、最後に Viewer と続きます。ロールベース ライセンス発行の詳細については、ライセンス発行の概要を参照してください。

ライセンスの有効期限切れのためユーザーまたは管理者がライセンスなしである

ユーザーが予期せずライセンスなしになったり、別のサイト ロールに移行することがないよう、現在使用しているライセンスの有効期限が切れる前に、以下のうちいずれかを常に実行する必要があります。

- 置換 ライセンスを更新してライセンス認証する。ユーザーが Creator、Explorer、または Viewer (ビューアー) ライセンスを使用しており、それらのライセンスの有効期限が切れる場合、利用可能であれば同じ種類の別のライセンスを使用します。
- それらのユーザーのサイト ロールを、有効期限が切れていないライセンスの使用を許可するように変更する。

別のライセンスを要求するようにサイト ロールを変更する方法については、ユーザーのサイト ロールの設定を参照してください。

新しいライセンスへのユーザーの再割り当ては、以下の論理で管理されます。

- サーバー管理者ユーザーが Creator ライセンスを使用し、そのライセンスの有効期限が切れる場合 (また、置換 ライセンスを利用できない)、Explorer ライセンスが利用可能であれば Explorer ライセンスに再割り当てされます。このライセンス再割り当ては、最近ログインした順番に行われます。サーバー管理者は、現在 Explorer ライセンスを使用している可能性がある他のユーザーを他に移動させます。利用可能な Creator または Explorer ライセンスがない場合、サーバー管理者はライセンスなしとなります。
- 非サーバー管理者ユーザーが Creator ライセンスを使用し、そのライセンスの有効期限が切れる場合 (また、置換 ライセンスを利用できない)、ライセンスなしとなります。これらのユーザーがライセンスなしにならないようにするには、ライセンスの有効期限前にサイト ロールを変更します。これはサイト管理者 Creator サイト ロールのユーザーでは特に重要です。これらのユーザーは、サイト管理者権限を失うことがないよう、Creator ライセンスの有効期限が切れる前にサイト管理者 Explorer サイト ロールに移行する必要があります。
- 非サーバー管理者ユーザーが Explorer または Viewer ライセンスを使用し、そのライセンスの有効期限が切れる場合 (また、置換 ライセンスを利用できない)、より高いライセンス タイプが利用可能であれば、そのライセンス タイプにアップグレードされます。特に、ライセンスの有効期限が切れる場合には以下が起こります。

- Explorer ライセンスを使用するユーザーは、Creator ライセンスが利用可能であればこれに移行します (サイトロールの変更なし)。
- Viewer ライセンスを使用するユーザーは、利用可能であれば Explorer ライセンスに移行します。利用可能な Explorer ライセンスがない場合、Creator ライセンスが利用可能であればこれに移行します (サイトロールの変更なし)。.
- 高いライセンス タイプで利用可能なライセンスがない場合、これらのユーザーはライセンスなしに移行します。

上記の通りに、ユーザーは最近ログインした順番で新しいライセンスに再割り当てされ、低いライセンス タイプが最初に再割り当てされます (最初に Viewer、次に Explorer、Creator の順).。

たとえば、Viewer ライセンスを持つユーザー 2 人、Creator ライセンスを持つユーザー 1 人、Creator ライセンスを持つサーバー管理者 2 人でどのライセンスも有効期限が切れているとします。これらのユーザーで有効期限が切れていない 4 つの Explorer ライセンスが利用可能です。この状況で、次のことが以下に示す順序で起こります。

1. 最近ログインした Viewer ライセンスを持つユーザーが Explorer ライセンスに再割り当てされます。
2. 2 番目の Viewer ライセンスを持つユーザーが Explorer ライセンスに再割り当てされます。
3. 最近ログインした Creator ライセンスを持つサーバー管理者が Explorer ライセンスに再割り当てされ、2 番目の Creator ライセンスを持つサーバー管理者が Explorer ライセンスに再割り当てされます。
4. Creator ライセンスを持つユーザーはライセンスなしになります。

Creator ライセンスを使用する場合、サーバー管理者サイトロールは変更されない

サーバー管理者は、Tableau Server で Creator ライセンスを使用可能な場合には Creator 権限を得ますが、サイトロール名に変更はありません。Tableau Server および Tableau Cloud の他のすべてのユーザーは、サイトロール名に Creator を含むサイトロールに割り当てられる場合にのみ Creator ライセンスを得ます。

ライセンスはすぐには利用できない

Tableau Server にロールベース ライセンスを追加する場合、それらのライセンスは Tableau Server の再起動時にすべてのユーザーで利用可能になります。

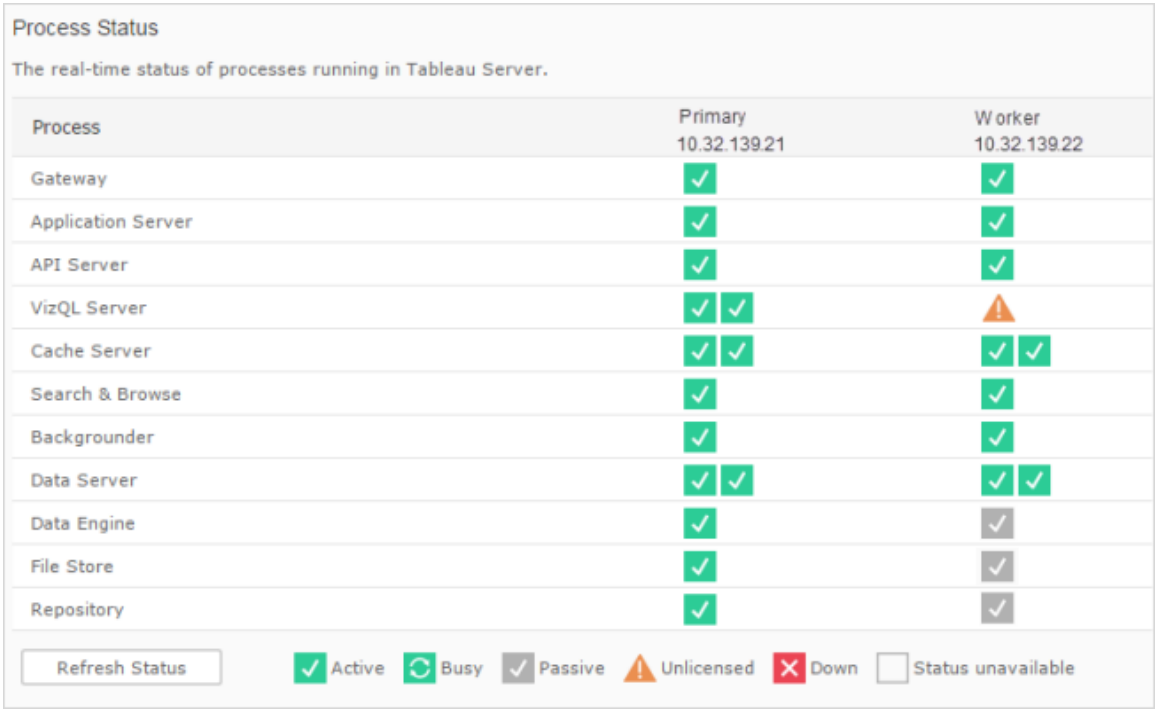
Viewer ライセンスを持つユーザーは、Tableau Desktop から Tableau Server または Tableau Cloud のワークブックを開くことができない

Viewer ライセンスを持ち、それとは別に Tableau Desktop ライセンスも持つユーザーは、Tableau Desktop を使用して Tableau Server または Tableau Cloud のワークブックを開くことはできません。このように Tableau Desktop を使用してワークブックを開くには、そのユーザーに Tableau Server または Tableau Cloud で Explorer または Creator ライセンスが必要です。

ライセンスのないサーバープロセスの処理

Tableau Server の [ステータス] ページには、Tableau Serverプロセスの状態の把握に役立ついくつかのステータス インジケーターがあります。オレンジ色のステータス ボックスの "ライセンスなし" は、いずれかのサーバープロセスで Tableau Server のライセンス情報を取得できないことを示します。

次の図は、VizQL プロセスの 1 つにライセンスがないことを示します。



プロセスでライセンス情報を取得できない理由として、いくつかの原因が考えられます。たとえば、ネットワーク上の問題により、追加ノードで実行されているプロセスでライセンス発行サービスとの通信ができなくなっている場合があります。または、ある特定の時点で、ライセンスなしのプロセスで受

け入れできる数よりも多くの要求が送られているために、ライセンス発行の要求を処理できなくなっている可能性があります。ユーザーへの影響は、ライセンスを確認できないプロセスや、いずれかのサーバー ノードにプロセスの他のインスタンスがあるかどうかによって異なります。上記のライセンスなし VizQL プロセスの場合、一部のユーザーはビューにアクセスできても、他のユーザーはアクセスできない可能性があります。

この問題を解決するには、Tableau Serverを停止した後に起動します。

Tableau サービス マネージャー (TSM) のコマンド タイムアウト

Tableau Server で 2 つのインスタンスのリポジトリを構成しており、バックアップ リポジトリに対するフェールオーバーが起こると、TSM は元のリポジトリを再起動し、それをバックアップとして利用できるようにします。これが何らかの理由で行えない場合、その後の TSM コマンドは元のリポジトリが回復するのを待機している間はタイムアウトのためにエラーになる可能性があります。

影響を受ける可能性があるコマンドとしては以下のものがあります。

- tsm maintenance restore
- tsm maintenance reindex-search
- tsm reset
- tsm security regenerate-internal-tokens
- tsm sites export
- tsm sites import

これらのコマンドのいずれかがエラーになっており、回復していないリポジトリがある場合は、そのリポジトリをサーバー トポロジから削除し、保留中の変更を適用してから、それを再追加してください。

Tableau サービス マネージャー (TSM) のバックアップのトラブルシューティング

サービスが起動しないため、バックアップを開始できない

Tableau Server をバックアップする場合に行う最初の手順の 1 つは、主要なサービスが実行中であることを確認し、実行中でなければ起動することです。これらのサービスを起動できない場合:

- アクティブ リポジトリ
- ファイル ストア
- クラスタ コントローラー

Tableau Server をバックアップしようとすると、次のいずれかのエラーのために失敗します。

```
An error occurred starting one or more of the following services:
Active Repository, File Store, Cluster Controller.
```

```
One or more of the following services did not start in a timely
fashion: Active Repository, File Store, Cluster Controller.
```

Tableau Server を正常にバックアップするには、これらのプロセスが起動可能であることを確認してください。

Cookie 制限エラー

ユーザーが Tableau Server にサインインしたとき、セッション Cookie はローカルのブラウザーに保存されています。保存された Cookie は、ユーザーの Tableau Server へのサインイン認証とサーバーへのアクセス許可を管理する方法です。Cookie はブラウザーのアドレス バーと同じドメインまたはサブドメインで設定されているので、ファーストパーティの Cookie とみなされます。ユーザーのブラウザーがファースト パーティの Cookie をブロックするように設定されている場合、ユーザーは Tableau Server にサインインできません。

埋め込みビュー経由で、または信頼できる認証が設定された環境で、ユーザーが Tableau Server にサインインする場合も、同じことが起こります。cookie は保存されます。しかし、この場合、ブラウザーは Cookie をサード パーティ Cookie として扱います。これは、Cookie がブラウザーのアドレス バーに表示されているものと異なるドメインで設定されていることが理由です。ユーザーの Web ブラウザーがサード パーティの Cookie をブロックするように設定されている場合、Tableau Serverの認証は失敗します。失敗しないようにするには、Web ブラウザーがサード パーティの Cookie を許可するように構成する必要があります。

サブスクリプションのトラブルシューティング

"このメール内のビュー スナップショットを正しくレンダリングできませんでした。"

このエラー メッセージ付きサブスクリプションを受け取った場合は、いくつかの理由がある可能性があります。

- 認証資格情報が見つかりません。いくつかのビューは、埋め込み認証資格情報を使用してパブリッシュされます。埋め込み認証資格情報が期限切れの場合、または埋め込み認証資格情報を使用せずにビューが再度パブリッシュされた場合、上記のエラーが表示される可能性があります。
- データベースが一時的にダウンしています。ビューが稼働中のデータベースと接続しており、サブスクリプションの生成時にデータベースが一時的にダウンした場合、上記のエラーが表示される可能性があります。
- **バックグラウンド プロセスのタイムアウト**: 既定では、サブスクリプションを処理するバックグラウンドプロセスで、ビューのレンダリングのタイムアウト値はビューあたり30 分間です。ビューのレンダリングがこの時間制限を超える場合、ワークブックに含まれる次のビューで、タイムアウトのためジョブのエラーが発生します。ほとんどの場合、既定で十分な時間があります。しかし、バックグラウンドプロセスが非常に大規模で複雑なダッシュボードを処理している場合は、十分な時間がない可能性があります。時間が十分あるかどうかを判断するために、抽出以外のバックグラウンド タスク管理者ビューを確認できます。タイムアウトのしきい値を増やすには、`tsm configuration set subscriptions.timeout` コマンドを使用します。

メール内の画像を表示できない

登録メール内でコンテンツの画像を表示するには、ビューにサブスクライブしているユーザーに **[ビュー]** パーミッションに加え、**[画像/PDF のダウンロード]** パーミッションも必要です。詳細については、パーミッションを参照してください。

サブスクライブできない

ユーザーが Tableau Server 上でビューを表示でき、右上隅にサブスクリプション アイコン (✉+) がある場合、ビューをサブスクライブできます。

ビューをサブスクライブするには、Tableau Server を正しく設定する必要があります (「サブスクリプションの管理」を参照)。さらに、サブスクライブしているビューのデータソースに埋め込み認証資格情報を指定するか、認証資格情報に依存しないようにする必要があります。後者の例には、更新していない抽出に接続するワークブック、またはパブリッシュ時にワークブックに含まれていたファイル内に、データが置かれているワークブックが含まれます。認証資格情報の埋め込みは、Tableau Desktop で発生する 1 つの手順です (詳細については Tableau ヘルプを参照)。

サブスクリプション アイコンがない

ビューを表示できても、サブスクライブできない可能性があります。この原因として、以下の複数の理由が考えられます。

- **サブスクリプションがスケジュールされていない**: サブスクリプションがスケジュールされていない場合や、サブスクリプションのスケジュールが無効になっている場合は、サブスクリプション アイコンが表示されません。サブスクリプションのスケジュールを設定するには、「スケジュールの作成または変更」を参照してください。
- **ビューが稼働中のデータベースの接続を使用している**: 稼働中のデータベースと接続しているビュー (最初にビューをクリックしたときにデータベース認証資格情報の入力が要求される) は、サブスクリプション用に使用できません。サブスクリプションには、ビュー (またはワークブック)、データ、およびスケジュールが含まれます。ビューに必要なデータを配信するために、Tableau Server には埋め込みデータベース認証資格情報、または認証資格情報を必要としないデータのいずれかが必要です。稼働中のデータベースへの接続が関係している場合、Tableau Server は認証資格情報を持たず、各ユーザーのみが認証資格情報を持ちます。この理由により、サブスクライブできるのは、認証資格情報が必要でないビューか、認証資格情報が埋め込まれているビューのみです。
- **Tableau Server が信頼できる認証用に設定されている**: Tableau Server が信頼できる認証用に設定されている場合、ビューを表示できてもサブスクライブできない (サブスクリプション アイコンがない) 場合があります。詳細については、「サブスクリプションのためのアクセス確保」を参照してください。
- **ライブ データベース接続で Oauth 認証を使用している**: OAuth 認証資格情報がライブ データソースに埋め込まれていないか、ユーザーキーチェーンに保存されていない場合、また

は OAuth 更新トークンの有効期限が切れている場合、ユーザーはビューをサブスクライブできない可能性があります。

無効または「壊れた」subscriptions

稼動中のインスタンスに加えて、Tableau Serverのテストインスタンスまたは開発インスタンスでサブスクリプションを構成した場合は、稼動中でないインスタンスでのサブスクリプションを無効にします。すべてのインスタンスでサブスクリプションを有効に保つことにより、ユーザーは有効だと表示されるが動作しないサブスクリプションを受信したり、ビューまたはワークブックからサブスクリプションを解除したにもかかわらず、サブスクリプションを受信する可能性があります。

添付ファイルがない

管理者がサブスクリプションを有効にしている場合は、PDF 添付ファイルを追加できます。PDF 添付ファイルがサブスクリプションにない場合は、PDF のサイズがメール サーバーのサイズ制限またはサーバー管理者が設定した最大サイズの制限を超えている可能性があります。Tableau Server では、サブスクリプションへの PDF 添付ファイルの最大サイズ制限を tsm configuration オプション `subscriptions.max_attachment_size_megabytes` を使用して調整できます。詳細については、サーバー イベント通知の構成およびサブスクリプションのサーバー設定を参照してください。

Tableau 2024.1 以降では、独自の送信サーバーを使用してメールを送信できるようになりました。最大メール サイズが 10 MB の添付ファイルを送信できるようになります。

この機能を有効にするには、サイト設定に移動し、**[メール通知のカスタマイズ]** セクションで **[独自の SMTP サーバーを使用する]** の横のボックスをオンにします。

> 注: 独自の送信サーバーを使用している場合、Tableau は 10 MB の制限を下回るように、添付ファイルの一部としてサブスクリプション メールの送信を試みます。ただし、これは保証されません。Tableau が添付ファイルを送信できない場合は、添付ファイルが大きすぎて送信できないことを通知するメッセージが表示されます。

サブスクリプションの一時停止

既定では、連続して5回失敗するとサブスクリプションが一時停止します。サブスクリプションが一時停止する前に発生するサブスクリプションの失敗回数のしきい値を変更するには、tsm configuration set のオプションである`backgrounder.subscription_failure_threshold_for_run_prevention`を使用します。これにより、サブスクリプションが一時停止するために必要となる連続したサブスクリプションの失敗回数のしきい値が設定されます。これはサーバー全体の設定です。

サーバー管理者のみが、サブスクリプションが一時停止するためのサブスクリプションの失敗回数のしきい値を構成することができます。しきい値設定の詳細については、「サブスクリプションのサーバー設定」を参照してください。

既定では、サブスクリプションが一時停止しても管理者へのメールでの通知は行われませんが、**[マイ アカウントの設定]** でサイトごとに一時停止に関するメール通知を設定することができます。

一時停止したサブスクリプションの再開

管理者およびサブスクリプション所有者は、次の複数の方法でサブスクリプションを再開できます。

- コンテンツ設定の [マイ サブスクリプション] から
- 各ワークブックの [サブスクリプション] タブから
- タスクの [サブスクリプション] タブから (サーバー管理者のみ)

サブスクリプションが再開されると、アラートの失敗回数が0に戻ります。サブスクリプションの次の評価は、スケジュールされている次回評価時に行われます。

サブスクリプションの頻度を "データ更新時" に設定できない

ワークブックでパブリッシュされた抽出への接続を使用している場合、抽出が更新されたときにサブスクリプションを実行するように設定できます。サブスクリプションを作成または変更するとき、ワークブックで次の値を使用している場合は、**[頻度]** オプションが表示されないことがあります。

- 複数の抽出更新
- ライブデータ接続

サブスクリプションが届かない ("メール送信エラー。コマンドを SMTP ホストに送信できない。")

サブスクリプションが届いておらず、SMTP サーバーが暗号化 (TLS) セッションを使用している場合、Windows Event Viewer に上記のエラーが表示される可能性があります。TLS を使用して構成された SMTP サーバーにサブスクリプションを送信するには、Tableau Server でセキュリティで保護された SMTP を構成する必要があります。SMTP セットアップの構成を参照してください。(このエラーが発生している場合、Tableau Server は [抽出以外のバックグラウンドタスク] 管理ビューで、サブスクリプションが送信中であることを示します。)

データ品質に関する警告または機密度ラベルが欠落している

データ品質に関する警告と機密度ラベルがサブスクリプション メールに含まれるのは、次の場合です。

- Tableau Server または Tableau Cloud がデータ管理 でライセンス設定されている。詳細については、データ管理 についてを参照してください。
- Tableau Catalog が有効になっている。詳細については、Tableau Catalog の有効化を参照してください。
- サイト設定で、**[ビューとワークブックのサブスクリプションにおける視認性の高いデータ ラベル]** のチェック ボックスが選択されています (以前のバージョンでは、チェックボックスは**[サブスクリプションのデータ品質に関する警告]** にあります)。

サーバー管理者参考資料

Tableau Serverプロセス、ポート、アカウントとアクセス許可の詳細については、こちらを参照してください。

Tableau Server プロセス

このトピックでは、プロセス構成の設定オプションについて説明します。Tableau Server プロセスを構成するには、各 ノードでどのプロセスのインスタンスをいくつ実行するべきかを指定する必要があります。これには、tsm topology set-process コマンドを使用します。詳細については、ノードのプロセス数を変更するを参照してください。

> Tableau Server on Windows については、「Tableau Server プロセス」を参照してください。

下記の表で明示的に示す場合を除き、プロセスに変更を適用すると、変更の適用時に Tableau Server が実行中の場合は、それを停止します。変更の適用後、Tableau Server はプロセス構成前の状態に戻るため、適用時にサーバーが実行されていた場合は、再起動されます。

> **重要**:使用するプロセス トポロジは組織のニーズによって異なります。

ライセンス プロセス

Tableau Server の一部としてインストールされた一部のプロセスは "ライセンス" プロセスです。ライセンス プロセスを実行するには、有効な Tableau Server ライセンスが必要です。Tableau Server の一部としてインストールされている他のプロセスは、有効なライセンスと結び付けられません。これにより、次のような影響があります。

- ライセンス プロセスはすべて、定期的に最初の Tableau Server コンピューター上で実行する Tableau Server ライセンス マネージャー サービスに定期的に接続する必要があります。有効なライセンスがあることを確認できない場合や、最初のノードが利用できない場合、プロセスは実行されず、Tableau Server は適切に機能しない、または安定して機能しない場合があります。
- コア ベースの Tableau Server ライセンスを持っている場合、ライセンス プロセスを使用するノード上のコアは、ライセンスを持つコアの合計数に対してカウントされます。

  > **注:** データ管理 とコアベースのライセンスを持っている場合は、各 ライセンスに含まれるライセンスを持つコアの合計数に対してライセンス プロセスがカウントされる方法を理解する必要があります。詳細については、データ管理 のライセンスを参照してください。

以下の表の "ライセンス" 列は、有効なライセンス プロセスと、コアベースのライセンスのコア数に影響を与えるプロセスを特定します。

Tableau Server プロセス これらのプロセスは、Tableau Server の実行時にステータスが `running` になり、Tableau Server の停止時に `stopped` になります。				
`tsm status -v` に表示される名前	`tsm topology set-process` で使用される名前	目的	注	ライセンス
分析拡張機能のマイクロサービス	`analyticsextensions`	分析拡張機能のマイクロサービスでは、分析拡張に式を渡すための一連の関数がサポートされ、R、Python、および Einstein Discovery と統合できます。	アプリケーション サーバー (VizPortal) がインストールされているノードに自動的にインストールされます。	なし
アプリケーション サーバー	`vizportal`	アプリケーション サーバー (VizPortal) は、Web アプリケーションおよび REST API 呼び出しを処理し、参	アプリケーション サーバーがインストールされている場合は、データエンジンもインストールされます (ノードにデータエンジンのインスタンスが含まれていない場合)。 アプリケーション サーバーの最初のインスタンスがノードにインストールされると、インタラクティブ	はい

		照と検索をサポートします。	なコンテナー サービスもインストールされます。	
データに聞く	手動で構成することはできません。	「データに聞く」サービスは、「データに聞く」機能により使用されます。	Data Server が実行中のすべてのノードで自動的に実行されます。	いいえ
認証 追加: 2022.1	手動で構成することはできません。	認証サービスは、アイデンティティの移行プロセスおよびアイデンティティプールを処理します。 アイデンティティの移行プロセスの完了後、認証サービスはアイデンティティプールの管理の一環として、以下を行います。	このプロセスのステータスは、tsm CLI を介してのみ使用できます。 ユーザー アイデンティティプールを設定するには、アイデンティティの移行が完了し、アイデンティティサービスがオンになっている必要があります。詳細については、「アイデンティティの移行について」および「アイデンティティプール (ID プール) を使用したユーザーのプロビジョニングと認証」を参照してください。	いいえ

		• 適切なアイデンティティサービステーブルでユーザーアイデンティティ検索し、返されたユニバーサル一意識別子 (UUID) を使		

		用して、従来の system_user テーブルに対してクエリを実行します。適切なシステムユーザーが見つかったら、ユーザーセッションを付		

		• 構成されたアイデンティティストアが外部 (AD または LDA-P) の場合は、変更不能なユーザーアイデンティティと追加のユーザー属		

		性をインポートします。		
バックグラウンダー	backgrounder	バックグラウンダーでは、抽出の更新、サブスクリプション、「今すぐ実行」タスク、tabcmd から開始したタスクなどのサーバータスクを実行します。	バックグラウンダーがインストールされている場合は、データエンジンもインストールされます (ノードにデータエンジンのインスタンスが含まれていない場合)。 バックグラウンダーは単一スレッドのプロセスです。ノードにバックグラウンダーのインスタンスをさらに追加し、ジョブを並行で実行するためにノードの容量を拡張することができます。 多くの場合、サーバーを停止して再起動しなくても、実行中のサーバーの既存のノードでバックグラウンダー インスタンスの数を増減できます。ただし、データ管理のライセンスが付与されている場合やバックグラウンダーの最初のインスタンスをノードに追加する場合、またはノードからバックグラウンダーの最後のインスタンスを削除する場合を除きます。詳細については、Tableau Server の動的トポロ	はい

			ジの変更を参照してください。	
キャッシュサーバー	cacheserver	キャッシュサーバーは、サーバー クラスタ全体で分散および共有されるクエリキャッシュです。このインメモリキャッシュは、さまざまなシナリオにわたってユーザー エクスペリエンスを高速化します。VizQL Server、バックグラウンダー、データサーバー(アプリケーション サーバーも限定的に含む)は、ユーザーまたはジョブの代わりにキャッシュサーバーへ	キャッシュはシングル スレッドのため、より優れたパフォーマンスが必要な場合は、キャッシュサーバーの追加インスタンスを実行する必要があります。 各ノードが 2 個以下になるように制限して、最大 6 個のキャッシュサーバー インスタンスをインストールすることをお勧めします。当社のテストでは、Tableau Server インストールに合計 6 個を超えるキャッシュサーバー インスタンスをインストールしてもパフォーマンスの向上が見られず、パフォーマンスに悪影響を与える可能性があることが示されています。	いいえ

		キャッシュ要求を行います。		
クラスタコントローラー	clustercontroller	クラスタコントローラーは、さまざまなコンポーネントの監視、障害検出、および必要に応じてフェールオーバーの実行を担当します。	各ノードに自動的にすべてのインストールされます。	いいえ
コレクション 追加: 2021.2.-0	collections	コレクションサービスは、コレクションとお気に入り機能のメタデータを提供します。	コレクション サービスは、アプリケーション (vizportal) がインストールされている最初のノードにインストールされます。 高可用性を実現するため、アプリケーション サーバーのインスタンスがインストールされているすべてのノードでコレクション サービスのインスタンスをインストールすることをお勧めします。詳細については、Tableau Server コレクション サービスを参照してください。	いいえ
接続プール		接続プール サービス	接続プール サービスは、すべてのノードに無効な状態でインス	いいえ

追加: 2023.1.- 0		は、リポジトリへのデータベース接続をプールするオプションを提供します。	トールされています。TSM 以外では有効にすることはできません。 Tableau サポートからの指示がない限り、有効にしないでください。	
コンテンツの探索 追加: 2021.1.- 0	`contentexplora-tion`	コンテンツ探索サービスを使用すると、Tableau Server の検索機能と参照機能が拡張します。その操作は、Index and Search Server に依存します。	コンテンツ探索サービスは最初のノードにインストールされています。 高可用性を実現するため、アプリケーション サーバーがインストールされているすべてのノードでコンテンツ探索 サービスのインスタンスをインストールすることをお勧めします。詳細については、Tableau Server コンテンツ探索サービスを参照してください。	いいえ
データエンジン	手動で構成することはできません。	データエンジンで、データ抽出を作成し、クエリを処理します。	ファイル ストア、VizQL Server、アプリケーション サーバー (VizPortal)、データ サーバー、Prep フロー作成、またはバックグラウンダーをインストールする場合は、自動的にインストールされます。 **注:** ファイル ストアが外部で構成されている場合、データエン	はい

			ジンがファイル ストアと共にインストールされることはなくなりました。詳細については、Tableau Server 外部ファイル ストアを参照してください。	
データプロファイリング 追加: 2021.4.-0	dataprofiling	データプロファイリング サービスは、仮想接続エディターの列プロファイリング要求を処理します。	データ管理 ライセンスを持っている場合にのみインストールされます。その場合、バックグラウンダーがインストールされているノードに自動的にインストールされます。	はい、データ管理 ライセンスが必要です
データ サーバー	dataserver	データ サーバーは、Tableau Server のデータ ソースへの接続を管理します。	データ サーバーがインストールされている場合は、データ エンジンもインストールされます (ノードにデータ エンジンのインスタンスが含まれていない場合)。	はい
データ ソース プロパティ 追加: 2020.1.-0	tdsservice	データ ソース プロパティ サービスは、「データに聞く」などのクライアント サービスに、パブリッシュされたデータ	データ ソース プロパティは、既定では初期ノードに追加されます。1 つの Tableau Server インストールに少なくとも 1 つのインスタンスが含まれている必要があります。パフォーマンス上の理由から、アプリケーション サーバー (VizPortal) がインストールされているノードにデータ ソース プロパティ サービスをインストー	いいえ

		ソースのメタデータを提供します。	ルすることをお勧めします。	
データストーリー 追加: 2023.1.-0		データストーリーサービスは、データストーリーのダッシュボード拡張機能を動作させる分析エンジンを処理します。	データストーリーは、Tableau Server インストールのすべてのノードに自動的に追加されます。手動で構成することはできません。	いいえ
Elastic Server 追加: 2019.1 削除: 2022.1 バージョン 2022.1 以降では、代わりに Index and Search	`elasticserver`	Elastic Server は、「データに聞く」およびコンテンツ探索サービスで、検索可能なコンテンツのインデックスを作成する際に使用するものです。	Elastic Server プロセスは、クラスタ内の複数のノードで実行できます。任意のノードに移動することもできます。実行される Elastic Server プロセスの数は奇数にすることをお勧めします。 Elastic Server のヒープサイズは、`elasticserver.vmopts` TSM 構成オプションを使用して構成できます。詳細については、tsm configuration set のオプションを参照してください。	いいえ

Server が使用されます。				
抽出サービス 追加: 2021.4.-0 廃止済み: 2023.3.-0 このバージョンおよびそれ以降のバージョンでは、仮想接続はバックグラウンダーによって処理されます。	`extractservice`	抽出サービスは、仮想接続の抽出を管理します。	データ管理 ライセンスを持っている場合にのみインストールされます。その場合、バックグラウンダーがインストールされているノードに自動的にインストールされます。	はい、データ管理 ライセンスが必要です
ファイルストア	`filestore`	ファイル ストアは、Tableau	ファイル ストアのインストール時にデータ エンジンもインストールされます (ノードにデータ エンジ	ローカル ファイル ストア: いい

		Server 上でローカルに実行するか、SAN または NAS ストレージを使用して外部で実行するように構成できます。 ローカルで構成した場合、ファイル ストアは、データ エンジン ノード全体で抽出を自動的にレプリケートします。	ンのインスタンスがすでに存在する場合や、ファイル ストアが外部で構成されている場合を除く。	え 外部ファイル ストア: Advanced Manage-ment ライセンスが必要です。
ゲートウェイ	`gateway`	ゲートウェイは、ブラウザー、Tableau Desktop、およびその他のクライアントから Tableau Server への	VizQL Server、Vizportal、Tableau Prep フロー作成のインスタンスがあるノードで必要です。	いいえ

		すべての要求を処理するWebサーバーです。		
インデックス作成/検索サーバー 追加: 2022.1	`indexandsearch-server`	Index and Search Server は、AWS OpenSear-ch に基づいています。Tableau は、Open Search の検索機能を使用して「データに聞く」およびコンテンツ探索サービスのデータにインデックスを作成します。 このサーバープロセスは、バージョン 2022.1 以降で使用されなくなった	Index and Search Server は、クラスタ内の複数のノードで構成できます。 実行する Index and Search Server プロセスの合計数は奇数にすることをお勧めします。3 つ以上のノードを持つ Tableau Server クラスターでは、少なくとも 3 つの異なるノードで Index and Search Server を構成することをお勧めします。 Index and Search Server のヒープサイズは、`indexandsearchserver.v-mopts` オプションを指定した `tsm set configuration` コマンドを使用して構成できます。詳細については、「tsm configuration set のオプション」を参照してください。	いいえ

		Elastic Server に代わるものです。 バージョン 2023.3.0 では、このプロセスは検索と参照から置き換わります。		
内部データソースプロパティ 追加: 2020.1.-0	tdsnativeservice 手動で構成することはできません。	内部データソースプロパティサービスは、データソースプロパティサービスとのみ通信する内部サービスです。	データソースプロパティのインスタンスがある各ノードでは、内部データソースプロパティの 1 つのインスタンスが自動的に構成されます。	いいえ
メッセージングサービス 追加: 2019.4.-0	activemqserver	メッセージング サービスは、Tableau Server のマイクロサービス間の通信をサポー	Tableau Server をインストールする際に、各ノードに自動的にインストールされています。サービスのインスタンスが 1 つ必要です。 Tableau Server の分散インストールでは、メッセージング サー	いいえ

		トするために使用されます。	ビスを別のノードに移動できます。バージョン 2020.1 では、一定の冗長性を提供するために、追加のノードにメッセージング サービスの 2 番目のインスタンスを追加できます (2019.4 では、クラスター内に複数のインスタンスを構成することはできません)。詳細については、Tableau Server メッセージング サービスを参照してください。	
メトリクス サービス 追加: 2020.2.-0 廃止: 2024.2	metrics	メトリクス サービスは、Tableau Server でメトリクス データの読み取りと書き込みを行います。	Tableau Server をインストールする際に、1 つのインスタンスが初期ノードに自動的にインストールされます。サービスのインスタンスが 1 つ必要です。 必要に応じてインスタンスを追加できます。 Tableau Server のマルチノード インストールでは、各ノードにメトリクス サービスのインスタンスを最低 1 つインストールすることをお勧めします。詳細については、「Tableau Server メトリクス サービス」を参照してください。 (従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳	いいえ

			細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。)	
Minerva Service 追加: 2021.4.-0	minerva	Minerva Service は、仮想接続に対してクエリを実行します。	データ管理 ライセンスを持っている場合にのみインストールされます。その場合、バックグラウンダーがインストールされているノードに自動的にインストールされます。	はい、データ管理 ライセンスが必要です
非リレーショナル ストレージ サービス 追加: 2023.1.-0 廃止: 2024.1.-0 **注:** このサービスはバージョン 2023.x の CLI に表示されますが機能	nrs	非リレーショナル ストレージ サービスは、他の Tableau サービスによって内部的に管理および使用されるマイクロサービスです。	Tableau Server の第一ノードに自動的に言インストールされます。このサービスは、システム管理者が管理することはできません。	いいえ

しません。				
仮想接続サービス 追加: 2021.4.-0	publishedconne-ctions	仮想接続サービスは、仮想接続に対するクエリを処理します。	データ管理 ライセンスを持っている場合にのみインストールされます。その場合、バックグラウンダーがインストールされているノードに自動的にインストールされます。	はい、データ管理 ライセンスが必要です
クエリゲートウェイ マイクロサービス 追加: 2021.4.-0	querygateway	クエリゲートウェイ マイクロサービスは、クエリのタイプとソースに応じて、クエリを適切なマイクロ サービスにルーティングします。	データ管理 ライセンスを持っている場合にのみインストールされます。その場合、バックグラウンダーがインストールされているノードに自動的にインストールされます。	はい、データ管理 ライセンスが必要です
クエリポリシー サービス 追加: 2021.4.-0 廃止済み: 2023.3.-0	querypolicy	クエリポリシー サービスは、クエリを処理するときにデータポリシーに関する情報を提供します。	データ管理 ライセンスを持っている場合にのみインストールされます。その場合、バックグラウンダーがインストールされているノードに自動的にインストールされます。	はい、データ管理 ライセンスが必要です

リポジトリ	pgsql	PostgreSQL リポジトリは、Tableau Server 用のメインデータベースです。ワークブックとユーザーのメタデータを保存します。Tableau Catalog (または Tableau メタデータ API) が有効な場合、リポジトリには Tableau のコンテンツと外部アセットのメタデータが保存されます。	1 つのクラスタに追加できるリポジトリインスタンスは最大 2 つに制限されます。また、2 つ目のリポジトリインスタンスを追加するには、少なくとも 3 つのノードがクラスタになければなりません。	いいえ
リソース制限マネージャー 追加:	"Resource Limits Manager"	Tableau Server リソース制限マネージャーは、設定された	リソース制限マネージャーは、既定で Tableau Server の初期ノードに自動的にインストールされます。Tableau Server の追加ノードで、プロセスを追加したり、これを構成したりするこ	いいえ Advanced Management が必要です

2022.1		リソース制限に対するバックグラウンダーリソースの使用状況を追跡し、リソース制限が正しく適用されていることを確認します。	とはお勧めしません。	
SAML サービス	手動で構成することはできません。	SAML サービスは Tableau Server と SAML アイデンティティプロバイダー (IdP) 間のプロキシとして機能します。	Tableau Server をインストールした各ノードに自動的にインストールされています。 サイトの SAML が有効でない限り、`stopped` 状態で `tsm status -v` 出力にリストされます。 SAML サービスは手動で構成できません。	いいえ
検索と参照 廃止済み: 2023.3.-0	`searchserver`	検索と参照サービスは、サーバー上のコンテンツメタデータの高速検索、フィルター、取得、および表示を処	バージョン 2022.3.0 以降、検索と参照は使用できなくなりました。インデックス作成/検索サーバーがこのサービスに完全に置き換わります。バージョン 2023.3.0 以降、インストールされません。 **注:** Tableau Server バージョン 2022.3 または 2023.1 を実行し	いいえ

		理します。	ている場合は、どのインストールでも*検索と参照のインスタンスを複数構成しないでください。*インスタンスを複数構成すると、まれに安定性の問題が発生することがあります。	
Tableau Prep Conductor	`flowprocessor`	Tableau Prep Conductor はフローを実行し、Data Catalog で取り込めるようにフローを処理します。このプロセスでは Tableau Server のスケジュール調整機能および追跡機能を活用しているため、フロー実行を自動化してフロー出力を更新することができます。	既定では、バックグラウンダーが有効になっているノードで自動的に有効になります。ノードロールがフローを除外するように設定されている場合、そのノードには Tableau Prep Conductor はインストールされません。詳細については、ノードロールによるワークロード管理を参照してください。2020.4 以降、Tableau Server でこのプロセスを有効にするためにデータ管理 を使用する必要はありません。	はい

Tableau Prep フロー作成 バージョン 2020.4 での追加	floweditor	ブラウザーで Prep のインタラクティブなフロー エクスペリエンスを提供します。	Tableau Prep フロー作成 (floweditor) がインストールされている場合、データ エンジン、Tableau Prep フロー サービス (flowqueryservice)、ゲートウェイもインストールされます。ただし、ノードにそれぞれのインスタンスが既に存在していない場合に限られます。	はい
Tableau Prep Minerva Service バージョン 2020.4 で Tableau Prep フロー サービスとして追加され、バージョン 2021.2 で名前が変更されました。	flowminerva 注: 旧称 flowqueryservice	Tableau Prep フロー作成 (floweditor) で、データソースのクエリを実行するために使用します。	既定では、Tableau Prep フロー作成 (floweditor) が有効になっているノードで自動的に有効になります。	はい

Tableau 統計サービス 追加: 2022.1 廃止済み: 2023.3	`statsservice`	Tableau 統計サービスは、「データの説明を見る」および予測モデリング関数の統計エンジンを管理します。	VizQL がインストールされているノードに自動的にインストールされます。 詳細については、「Tableau 統計サービス」を参照してください。	いいえ
Tableau Server VizData サービス 追加: 2024.2	`vizdataservice`	VizData サービスは、Tableau Server 上のパブリッシュされたデータソースへの接続を管理します。	新規インストールでは、VizData Service の 1 つまたは複数のインスタンスが最初のノードにインストールされます。インスタンスの数は利用可能なリソースによって決まります。 2024.2.3 へアップグレードする際、VizData サービスのインスタンスは、Data Server のすべてのインスタンスにインストールされます。 **注:** 2024.2.3 以降、Tableau Server のインストールには、VizData Service のインスタンスが少なくとも 1 つ必要です。	はい
Tableau Server VizData ネイティブサービス	`vizdatanatives-ervice`	VizData ネイティブサービスは、Tableau Server 上	VizData ネイティブサービスのインスタンスは、VizData サービスの各インスタンスに自動的に追加されます。	はい

追加: 2024.2		のVizDataサービスと通信します。		
Tableau Server VizQL Data Service 追加: 2025.1	`vizqldataservice`	VizQL データサービスは標準のHTTP サービスであり、Tableau でパブリッシュされたデータソースにプログラムによるクエリを通じてアクセスするためのAPI を提供します。	VizData サービスがインストールされているノードに自動的にインストールされます。 **注:** VizQL データサービスは、API を使用したプログラムによるアクセスを通じてのみ利用できます。VizQL データサービス API の詳細については、「VizQL データサービス API リファレンス」を参照してください。 サービスは、オフになっていない限り、TSM UI に表示されます。サービスをオフにするには、インスタンスを0 に設定してから、機能フラグを「false」に設定する必要があります。詳細については、「features.VizqlDataServiceDeployWithTsm」を参照してください。	いいえ
VizQL Server	`vizqlserver`	VizQL Server は、ビューを読み込んでレンダリングし、クエリを	VizQL Server がインストールされている場合は、ゲートウェイとデータエンジンもインストールされます。ただし、ノードにゲートウェイとデータエンジンのインスタンスが含まれていない場合に限	はい

		計算して実行します。	られます。 たいていの場合、サーバーを停止、再起動せずに実行中のサーバーの既存のノードで VizQL インスタンスの数を変更できます。ゲートウェイとデータエンジンをインストールする VizQL やその他のプロセスがなかった既存のノードに VizQL を追加する場合は例外です。詳細については、Tableau Server の動的トポロジの変更を参照してください。	
Tableau マイクロサービス コンテナ プロセス このプロセスは、バックグラウンダーまたはアプリケーション サーバーの最初のインスタンスがノードに追加されるときに自動的に追加されます。バックグラウンダーまたはアプリケーション サーバーのすべてのインスタンスがノードから削除されると、マイクロサービス コンテナ プロセスも削除されます。 コンテナーのステータスは、コンテナー内のマイクロサービスのステータスによって決まります。すべてのマイクロサービスを実行中の場合、コンテナープロセスのステータスは `running` です。すべてのマイクロサービスが停止している場合、コンテナープロセスのステータスは `error` になります。1 つまたは複数のマイクロサービスが実行され、それ以外は実行されていない場合のコンテナ サービスのステータスは `degraded` です。詳細については、Tableau Server マイクロサービス コンテナーを参照してください。				
インタラクティブなマイクロサービス コンテナ		Tableau Server 内部マイクロサービス向けのコンテナープロセ	これらのコンテナーとそれに含まれるマイクロサービスは手動で構成できません。マイクロサービスは時間の経過とともに変化する場合があります。	いいえ

		ス。展開を容易にするため、またスケーラビリティ目的で一緒にバンドルされています。		
インタラクティブでないマイクロサービスコンテナ	noninteractive	Tableau Server 内部マイクロサービス向けのコンテナープロセス。展開を容易にするため、またスケーラビリティ目的で一緒にバンドルされています。	これらのコンテナーとそれに含まれるマイクロサービスは手動で構成できません。マイクロサービスは時間の経過とともに変化する場合があります。	いいえ
Tableau サービス マネージャー (TSM) プロセス これらのプロセスは、TSM の初期化が完了するとステータスが `running` になり、Tableau Server が停止されてもその実行状態を維持します。				
アクティブ化サービス バージョン 2021.1 での追	手動で構成することはできません。	アクティブ化サービス (実行承認サービス (ATR) とも呼ばれます) を使用	バージョン 2023.1.3、2022.3.7、2022.1.15 以降では、初期ノードに自動的にインストールされます。 それより前のバージョンでは、ATR が有効になっているときに、初期ノードに自動的にイン	いいえ

加		すると、ライセンスを使い切る前に、Tableau Server をアクティブ化することができます。プロダクトキーの有効期限が切れるまで、構成可能な期間中に短期リースが提供されます。	ストールされます。	
管理エージェント	手動で構成することはできません。	TSM エージェントは、構成またはトポロジーへの変更がないか調整サービスを監視して、新しい構成を各サービス(構成)へ提供するか、新しいサービスを展開して古いものを削	Tableau Server をインストールした各ノードに自動的にインストールされています。 管理エージェントは手動で構成できません。 詳細については、Tableau Server 管理エージェントを参照してください。	いいえ

		除します(トポロジー)。		
管理コントローラー	別のノードに移動する以外は、管理コントローラーを手動で構成することはできません。詳細については、初期ノードの障害からの回復を参照してください。	TSM コントローラーは、TSM への要求を処理し、構成とトポロジの変更、およびサービスプロセス全体のワークフローを調整します。コントローラーは REST API のエンドポイント (HTTPS) としても機能します。	TSM を最初のノードにインストールする際に、自動的にインストールされます。 別のノードに移動する以外は、管理コントローラーを手動で構成することはできません。詳細については、初期ノードの障害からの回復を参照してください。 詳細については、Tableau Server 管理コントローラーを参照してください。Tableau Server 管理コントローラー	いいえ
クライアントファイルサービス	`clientfileserv-ice`	複数ノードのクラスタ内でほとんどの共有ファイルを管理するのが Client File Service (CFS) で	自動的に最初のノードにインストールされます。他のインスタンスは、ユーザーが明示的に構成しない限りインストールされません。クライアントファイル サービスの構成を参照してください。 マルチノード展開では、調整サービスを展開する各ノードで CFS のインスタンスを構成する	いいえ

		す。例えば、認証関連の証明書、キー、ファイル (OpenID、相互 SSL、SAML、Kerbero-s)、カスタマイズ ファイルなどが CFS によって管理されます。	ことを推奨します。調整サービスを再展開しても CFS に影響はありません。 CFS は [ステータス] ページや [構成] ページには表示されませんが、`tsm status -v` コマンドの出力では表示されます。 CFS のインスタンスを表示または設定するには、tsm topology コマンドを使用します。	
調整サービス	`tsm topology set-process` では設定できません。	調整サービスは唯一の真実の参照元の役割を担います。	自動的に最初のノードにインストールされます。新しく調整サービス アンサンブルを明示的に展開しない限り、他のインスタンスはインストールされません。詳細は、調整サービス アンサンブルの展開を参照してください。	いいえ
ライセンス マネージャー	手動で構成することはできません。	ライセンス マネージャーはライセンシングを取り扱います。	最初のノードに TSM をインストールすると、自動的にインストールされます。 このインスタンスは、Tableau Server に 1 つだけインストールされます。ライセンス マネージャー プロセスは、最初のノードに障害が発生したときのみ手動で構成されるべきです。詳細	いいえ

			については、初期ノードの障害からの回復を参照してください。	
Tableau Server メンテナンス プロセス これらのプロセスは、ジョブを完了するためにアクティブに実行されている場合を除き、ステータスが `stopped` になっています。				
データベースメンテナンス	手動で構成することはできません。	データベース メンテナンス サービスは Tableau Server リポジトリの保守操作を実行します。	Tableau Server をインストールした各ノードに自動的にインストールされています。 データベース メンテナンスを積極的に実行中でない限り、`stopped` 状態で `tsm status -v` 出力にリストされます。メンテナンスにはリポジトリへのリモートアクセスを可能にするための更新およびリポジトリにアクセスするために使用されるパスワードの変更が含まれます。 データベース メンテナンス サービスは手動で構成できません。	いいえ
バックアップ/復元	手動で構成することはできません。	バックアップと復元サービスは Tableau Server リポジトリおよびファイル ストアに保管されているデータのバックアップ	Tableau Server をインストールした各ノードに自動的にインストールされています。 バックアップまたは復元操作を実行中でない限り、`stopped` 状態で `tsm status -v` の出力にリストされます。 バックアップと復元サービスは手動で構成できません。	いいえ

		および復元操作を実行します。		
サイトのインポート/エクスポート	手動で構成することはできません。	サイトのインポートとエクスポートサービスはサーバークラスタ間で Tableau Server を移行します。	Tableau Server をインストールした各ノードに自動的にインストールされています。 インポートまたはエクスポートを実行中でない限り、`stopped` 状態で `tsm status -v` の出力にリストされます。 サイトのインポートとエクスポートサービスは手動で構成できません。	いいえ

プロセス ワークフロー

Tableau Server プロセスとその連携方法は、どのようなアクションやアクティビティが行われているかによって異なります。たとえば、使用されているプロセスや、それらのプロセスの連携方法は、ワークブックをパブリッシュする場合とSAML を使用してサインインする場合とで異なります。プロセス ワークフローの一部のインタラクティブ ビューについては、以下のワークブックを参照してください。特定のワークフローを選択し、初めから終わりまで順を追って従うことができます。

免責条項: このワークブックは Tableau Public でパブリッシュされており、Tableau ドキュメントにより保持されていません。最新バージョンの Tableau Server に対応する最新版であることは保証いたしかねます。

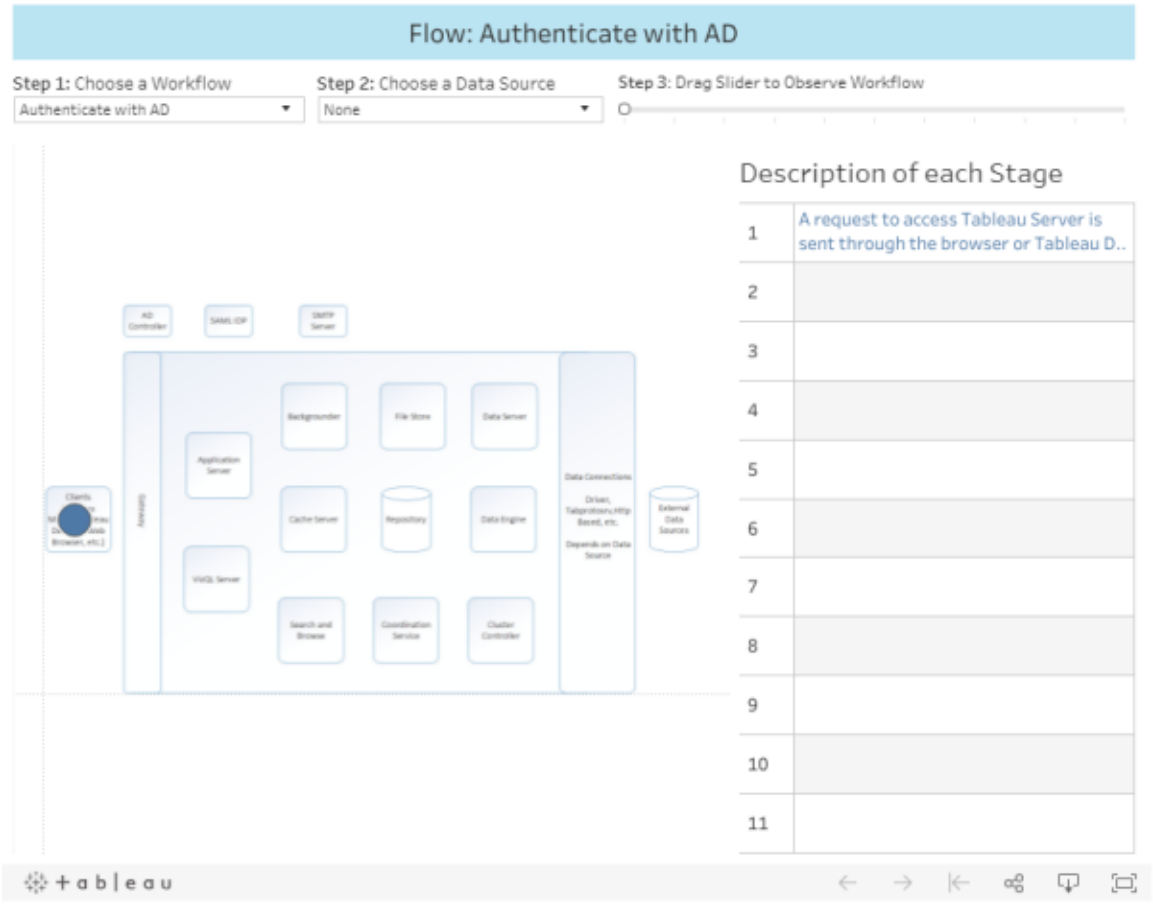


Tableau Server 管理エージェント

管理エージェントは、構成またはトポロジーへの変更がないか調整サービスを監視して、新しい構成を各サービス(構成)へ提供するか、新しいサービスを展開して古いものを削除します(トポロジー)。。管理エージェントは、各サービスのステータスをチェックし、それを Coordination サービスに報告します。このプロセスは、インストール中に、クラスタのノードごとに自動的に構成されます。明示的に構成する必要はなく、それを行うこともできません。

管理エージェントは、*Tableau* サービスマネージャーエージェントとも呼ばれます。

プロセス	Administration Agent

ステータス	Administration Agent プロセスのステータスは [ステータス] ページに表示されません。ステータスを表示するには、TSM CLI を使用します。詳細については、サーバープロセスステータスの表示を参照してください。
ロギング	Administration Agent プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/tabadminagent` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

管理エージェントプロセスが失敗した場合の処理同じノードで実行されている他のすべての Tableau Server プロセスは、[Tableau サービスマネージャー] ステータスページに"使用不可"と表示されます。Tableau Server は正常に動作しますが、クラスタに設定/トポロジー変更を加えることはできません。管理エージェントプロセスが失敗しても、パーソナルコンピューター自体が正常である場合は自動的に再起動します。ノードで管理エージェントが起動していない場合は、次のコマンドを実行してサービスを手動で開始できます。

```
sudo su -l tableau

systemctl --user start tabadminagent_0
```

Tableau Server 管理コントローラー

管理コントローラーのプロセスは TSM REST API をホストし、Tableau Server の展開を構成および管理します。クラスタ全体には管理コントローラーのインスタンスが 1 つしかないことがあります。

このプロセスは、インストール中にクラスタの最初のノードに自動的に構成されます。明示的に構成する必要はありません。

管理コントローラーは、*TSM コントローラー*や*TSM 管理コントローラー*とも呼ばれます。

プロセス	Administration Controller
ステータス	Administration Controller プロセスのステータスは [ステータス] ページに **TSM コントローラー**として表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Administration Controller プロセスが生成したログは `/var/opt/tableau/tableau_`

	server/data/tabsvc/logs/tabadmincontroller にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

管理コントローラープロセスが失敗した場合の処理

管理コントローラーに障害が発生した場合でも、Tableau Server クラスタは引き続き機能します。ただし、tsm コマンドとTSM Web UIは使用できなくなります。管理コントローラーが再起動し、実行中になるまで、構成やトポロジに対する変更や更新を行うことはできません。他の TSM サービスと同様、管理コントローラーが停止したり障害が発生した場合は、自動的に再起動されます。

ノードで管理コントローラーが起動していない場合は、次のコマンドを実行してサービスを手動で開始できます。

```
sudo su -l tableau

systemctl --user start tabadmincontroller_0
```

管理コントローラーを移動する

最初のノードに障害が発生した場合は、管理コントローラーとライセンス発行サービスを別のノードに移動して、Tableau Server が機能し続けることができるようにする必要があります。この方法の詳細については、初期ノードの障害からの回復を参照してください。

管理コントローラーを再起動する

TSM 管理コントローラーを (*tableau* システム アカウントとして) を再起動します。

```
sudo su -l tableau -c "systemctl --user restart tabadmincontroller_
0.service"
```

注: tabadmincontroller の再起動には数分かかる場合があります。コントローラーを完全に再起動させる前に次の手順で保留中の変更を適用しようとする場合、TSM はコントローラーに接続できなくなります。tsm status -v コマンドを使用して、コントローラーが実行されていることを確認できます。Tableau Server 管理コントローラーが「実行中」として表示されます。

Tableau Server のアプリケーション サーバー

アプリケーション サーバー (VizPortal) は、Web アプリケーションと REST API 呼び出しを処理します。アプリケーション サーバーは参照と検索もサポートしています。アプリケーション サーバーの高可用性を確保するため、Tableau Server クラスタ内の各ノードでインスタンスを構成します。

プロセス	Application Server
ステータス	Application Server のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Application Server プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizportal` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

アプリケーション サーバーのプロセスが失敗するとどうなりますか? そのインスタンスによって処理される要求は失敗しますが、それ以降の要求は、他の実行中のアプリケーション サーバープロセスにルーティングされます。失敗したアプリケーション サーバーを含むノードがまだ実行中の場合、失敗したプロセスは数秒以内に自動的に再起動します。

アプリケーション サーバーの問題のトラブルシューティング

バージョン 2024.2.0 以降、管理者はアクティビティとリソースのトレース (ART) データを有効にして、アプリケーション サーバーの問題を解決できます。キャプチャされた ART データは、メモリと CPU の使用状況に関する詳細な情報を提供します。これは、アプリケーション サーバー (VizPortal) のネイティブ メモリの飽和により、サーバーで サーバー リソース マネージャー (SRM) の再起動が発生している場合に役立ちます。

> **重要:** ART を有効にすると、vizportal java および cpp ログに追加のエントリが書き込まれます。調査が終了したら、ART を有効のままにしないでください。余分なディスク領域の使用を避けるため、必ず ART を無効にしてください。

Tableau Server で ART データを有効にする

ART データを有効にするには、以下を実施します。

1. 初期ノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次の tsm コマンドを実行します。
   a. `tsm configuration set -k vizportal.enable_art -v true`
   b. `tsm configuration set -k vizportal.log_art_java -v true`
   c. `tsm configuration set -k vizportal.art_skip_list -v "/v1/reportEventUnauthenticated,/v1/getSessionInfo,/v1/hasUnseenNotifications,/v1/refreshSession,/v1/getViews,/v1/getUsers,/v1/getWorkbooks,/v1/getView,/v1/getServerSettingsUnauthenticated,/v1/getProjectAncestors,/v1/getWorkbook"`
   d. `tsm restart`

アプリケーション サーバー ログは、既定では次の場所にあります。

```
C:\ProgramData\Tableau\Tableau Server\data\tabsvc\logs\vizportal
```

ART を無効にする

調査が完了したら、ART データを無効にします。ART を無効にするには、コマンドプロンプトで次のコマンドを実行します。

1. `tsm configuration set -k vizportal.enable_art -v false`
2. `tsm configuration set -k vizportal.log_art_java -v false`
3. `tsm configuration set -k vizportal.art_skip_list -v ""`
4. `tsm restart`

Tableau Server バックグラウンダー プロセス

バックグラウンダー プロセスでは、抽出更新、サブスクリプション、フロー実行、データ ドリブン アラートなどのサーバー ジョブが実行されます。ジョブは、スケジュールされたタスクから開始されるか、[今すぐ実行] オプション、REST API、または tabcmd コマンドを使用して手動で開始されます。

プロセス	Backgrounder
ステータス	Backgrounder のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Backgrounder プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/backgrounder` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

バックグラウンダープロセスが停止するとどうなりますか? エラーが発生したバックグラウンダープロセスにあるジョブは、バックグラウンダープロセスのエラーが復旧すると再試行されます。ほとんどのバックグラウンドジョブは定期的に実行するようにスケジュールされており、機能しているバックグラウンダープロセスによって次回のスケジュール時刻に同じバックグラウンドタスクが選択され正常に実行されます。

エラーが発生したバックグラウンダープロセスは、コンピューター自体に他の点で問題がない限り自動的に再起動し、失敗したジョブは再試行されます。

バックグラウンダープロセスの可用性を高くするには、1 つまたは複数のインスタンスをクラスタ内の複数のノードで実行するように構成する必要があります。

バックグラウンダー リソースの管理

このトピックで前述したように、バックグラウンダーではサーバー タスクが実行されるため、リソースが大量に消費される可能性があります。バックグラウンダーでサーバー タスクの実行用のリソースを管理するには、いくつかの方法があります。

- 特定のノードのインスタンスの数を増やす: バックグラウンダーはシングルスレッドのプロセスです。一度に起動できるジョブは 1 つだけです。ノードにさらにバックグラウンダー インスタンスを追加すると、そのノードで並列実行できるジョブの数を増やすことができますが、起動される各ジョブ自体で複数のスレッドを使用できることにも留意してください。バックグラウンダー インスタンスは、コア数の半分まで追加できます。バックグラウンダーの実行場所と実行数を決定する際は、ジョブに対して起動された各バックグラウンダーのプロセスでは複数のスレッドを使用できることを考慮に入れてください。バックグラウンダー インスタンスを追加すること

によって、各プロセスの有効性が制限される可能性があります。また、他のサーバープロセスが各マシンの空き容量に与える影響についても留意してください。

- バックグラウンダープロセスを分離する: マルチノードクラスタで Tableau Server を実行している場合は、1 つまたは複数のノードをバックグラウンダー専用にすることができます。詳細については、推奨されるベースライン構成を参照してください。

- ノードロール: ノード上のバックグラウンダーで実行される特定のタイプのジョブやワークロードを分離することもできます。たとえば、あるノードを抽出更新の実行専用にすることができます。詳細については、ノードロールによるワークロード管理を参照してください。

- ユーザーによるジョブの手動実行を制限する: 現在、ユーザーは Web インターフェイス、REST API、tabcmd コマンドを使用して抽出更新、フロー実行、サブスクリプションをいつでも実行できます。そのため、サーバーが他のアクティビティでビジー状態の時間帯にサーバーリソースが使用される可能性があります。Tableau Server 2020.1 以降では、サーバー管理者は新しい [今すぐ実行] 設定により、ユーザーにジョブの手動実行を許可するかどうかを選択できます。[今すぐ実行] オプションを無効にすると、バックグラウンダーの使用方法をより適切に制御できるほか、負荷をより適切に予測できます。これは、スケジュールされたタスク用に生成されたジョブに適用されたり、影響を与えたりはしません。この設定の構成の詳細については、サーバー設定 (全般とカスタマイズ)を参照してください。

関連コンテンツ

- グループ同期のパフォーマンスの向上

Tableau Server のキャッシュサーバー

キャッシュサーバーにより、共有されている外部クエリのキャッシュが提供されます。今後の要求を高速で実行するために以前のクエリの情報を保持するキーと値のペアのキャッシュです。キャッシュサーバーの可用性を高くするには、1 つまたは複数のキャッシュサーバープロセスをクラスタの複数のノードで構成します。

プロセス	Cache Server
ステー	Cache Server のステータスは [ステータス] ページに表示されます。詳細については、

タス	サーバープロセス ステータスの表示を参照してください。
ロギング	Cache Server プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/cacheserver` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

キャッシュサーバープロセスが停止するとどうなりますか? 比較的軽い影響があります。Tableau Server は引き続き動作しますが、キャッシュされた結果を利用できないため、操作に時間がかかる場合があります。クエリが再実行される際に、再起動されたキャッシュサーバーで再読み込みが行われ、最終的にエンドユーザーでの速度が向上します。実際には、キャッシュサーバーには可用性の影響はありませんが、エンドユーザーでのさまざまなシナリオのパフォーマンスに影響があります。

ユーザーのパフォーマンスに対する影響を軽減するには、クラスタ全体でこのタイプのプロセスを複数実行します。各ノードが 2 個以下になるように制限して、最大 6 個のキャッシュサーバー インスタンスをインストールすることをお勧めします。当社のテストでは、Tableau Server インストールに合計 6 個を超えるキャッシュサーバー インスタンスをインストールしてもパフォーマンスの向上が見られず、パフォーマンスに悪影響を与える可能性があることが示されています。

エラーが発生したキャッシュサーバープロセスは、コンピューター自体に他の点で問題がない限り自動的に再起動し、キャッシュサーバープロセスは再起動します。

Tableau Server クライアント ファイル サービス

クライアント ファイル サービス (CFS) は、TSM が必要とするファイル (証明書、カスタマイズ ファイルなど) を保存および配布します。クライアント ファイル サービスによって管理されるファイルは、展開全体に配布される前に名前が変更され、難読化されます。このプロセスでは、Tableau サービスに必要なファイル属性もパラメーター化されます。その結果、ファイルはファイル システムの 1 つのファイルの場所にマッピングされません。CFS で管理されるすべてのファイルの外部バックアップがあることを確認します。

以下のファイルは CFS によって管理されます。

- SAML 証明書ファイル
- SAML キー ファイル
- SAML IdP メタデータ ファイル

- OpenID.static.file
- Kerberos.keytab ファイル
- LDAP Kerberos keytab ファイル
- LDAP Kerberos conf ファイル
- 相互 SSL 証明書ファイル
- 相互 SSL 失効ファイル
- カスタマイズ ヘッダー ロゴ ファイル
- カスタマイズ サインイン ロゴ ファイル
- カスタマイズ コンパクト ロゴ ファイル

以下のファイルは CFS によって管理や配布が行われません。

- 外部 SSL ファイル。外部 SSL 用の証明書とキー ファイルは、調整サービスによって保存および管理されます。これらのファイルを手動で配布する必要はありません。
- LDAP 外部アイデンティティストアの SSL ファイルこれらの SSL 証明書ファイルは、クラスタ内の各 ノードに手動で配布する必要があります。外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。

クライアント ファイル サービスの機能は、ファイル ストアがビジネス サービスで必要なファイルに対して実行する機能とよく似ています。既定では、CFS は Tableau Server インストールの最初の ノードにのみインストールされます。CFS で高可用性を構成するには、調整サービスを展開する各 ノードで CFS のインスタンスを構成することを推奨します。

クラスターでは、CFS の唯一のインスタンスを実行しているノードに障害が発生すると、CFS によって管理されているファイルが失われるため、証明書とカスタム イメージをもう一度インポートし、関連する構成を変更して、CFS のこれらのファイルをもう一度設定する必要があります。

プロセス	Client File Service
ステータス	Client File Service プロセスのステータスは [ステータス] ページに表示されません。ステータスを表示するには、TSM CLI を使用します。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Client File Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/clientfileservice` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

CFS のプロセスが失敗するとどうなりますか? クラスタに機能している CFS プロセスが 1 つ以上ある限り、影響はありません。コントローラーは、ファイル転送要求を他の機能している CFS プロセスにリダイレクトします。

エラーが発生したクライアント ファイル サービスのプロセスは、コンピューター自体に他の点で問題がない限り自動的に再起動します。

Tableau Server コレクション サービス

Tableau Server コレクション サービスは、Tableau Server バージョン 2021.2 で追加されました。コレクション サービスにより、コレクション機能が強化されます。コレクションに関する情報を提供し、コンテンツ探索サービスに接続して、コレクションおよびコレクション内のアイテムに関するメタデータを取得します。コレクション サービスでは、お気に入り機能もサポートされます。

サーバー構成

コレクション サービスは、アプリケーション サーバー (vizportal) がインストールされている最初のノードに自動的にインストールされます。

マルチノード構成

高可用性とパフォーマンスの向上のために、アプリケーション サーバー (vizportal) を実行している各ノードに、少なくとも 1 つのコンテンツ探索サービスのインスタンスをインストールすることをお勧めします。

プロセス	Collections Service
ステータス	Collections Service のステータスは [ステータス] ページに表示されます。 `tsm status -v` コマンドを使用してコマンド ラインから表示することもできます。詳細については、サーバー プロセス ステータスの表示を参照してください。
ロギング	Collections Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/collections` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

コレクション サービスが正常に実行されていない場合の影響

コレクション サービスが停止しても、コンピューター自体に問題がなければ自動的に再起動します。

コレクション サービスの 1 つのインスタンス

コレクション サービスの単一のインスタンスが構成されていて、そのインスタンスに障害が発生した場合、コレクションとお気に入りは使用できなくなります。

コレクション サービスの複数のインスタンス

Tableau Server をマルチノードでインストールしていて、複数のノードでコレクション サービスのインスタンスを構成している場合、1 つのインスタンスに障害が発生すると、一部のユーザーがコレクションとお気に入りを使用できなくなる場合があります。

ログ ファイル

コレクション サービスにより以下の 2 つのセットのログ ファイルが作成されます。

- `control_collections*.log`: これらのログには、開始したり有効にしたりするサービスに関する情報が含まれます。
- `collections_*.log`: エラーや問題は、ここに記録されます。

詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

Tableau Server コンテンツ探索サービス

Tableau Server コンテンツ探索サービスは、検索および参照の処理機能を拡張したものであり、Tableau Server のすべてのコンテンツのインデックス作成を行います。このサービスは Tableau Server 2021.1 以降で利用でき、データベースやテーブルなどの外部アセットを検索して参照するために必要です。データベースとテーブルは、Tableau Catalog を有効にしている場合にのみ使用できます。

このコンテンツ探索サービスは、クイック サーチとフィルター検索の両方に適用できます。このサービスは SOLR に基づいていませんが、Elastic Search に依存して機能を実行します。

サーバー構成

コンテンツ探索サービスは最初のノードに自動的にインストールされています。

マルチノード構成

高可用性と優れたパフォーマンスを実現するには、アプリケーション サーバーを実行している各ノードに少なくとも 1 つのコンテンツ探索 サービスのインスタンスをインストールすることをお勧めします

プロセス	Content Exploration Service
ステータス	Content Exploration Serviceのステータスは ステータス ページに表示されますが、TSM CLI を使用してアクセスすることもできます。詳細については、サーバー プロセス ステータスの表示を参照してください。
ロギング	Content Exploration Service が生成したログは、`/var/opt/tableau/tableau_server/data/tabsvc/logs/contentexploration` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

コンテンツ探索 サービスが失敗するとどうなりますか?

コンテンツ探索 サービスが動作を停止すると、そのコンテンツ タイプに対する検索要求は検索結果に表示されません。サービスの複数のインスタンスがインストールされている場合、後続の要求は正常に稼働しているインスタンスにルーティングされます。

失敗したコンテンツ探索 サービスは、コンピューター自体が正常に稼働している限り、自動的に再起動されます。

パフォーマンスの調整

コンテンツ探索 サービスには、ノードで使用可能なメモリに基づいて、必要に応じてスケールアップする機能があります。ただし、サービスで使用できるメモリの最大量を決定するメモリ割り当てがデフォルトで設定されています。メモリ割り当ての不足によるパフォーマンスの問題が発生する場合は、次のコマンドを実行して最大 メモリ割り当ての設定を変更できます。ただし、通常、この設定を変更することはお勧めしません。

まず、次のコマンドを実行して、現在の最大 メモリ割り当てを取得します。

```
tsm configuration get -k contentexploration.vmopts
```

次の set コマンドを実行して、-Xmx 値を更新し、保留中の変更を適用することで最大メモリ割り当てを変更します。その他のオプションは変更しないでください。

```
tsm configuration set -k contentexploration.vmopts -v "-Xmx<new value>m -XX:+ExitOnOutOfMemoryError -Dspring.profiles.active=monolith"
```

例:

```
tsm configuration set -k contentexploration.vmopts -v "-Xmx1024m -XX:+ExitOnOutOfMemoryError -Dspring.profiles.active=monolith"
```

```
tsm pending-changes apply
```

ログ ファイル

コンテンツ探索サービスにより以下の 2 つのセットのログ ファイルが作成されます。

- `control_contentexploration*.log`: これらのログには、開始したり有効にしたりするサービスに関する情報が含まれます。
- `contentexploration_*.log`: エラーや問題は、ここに記録されます。

詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

検索精度

コンテンツ探索サービスは、いくつかの方法で検索可能なコンテンツを最新の状態に保ちます。

- Tableau Server または Tableau Cloud が管理するコンテンツが変更されるたびに (パーミッションの変更も含む)、コンテンツ探索サービスは、関連するドキュメントのインデックスを再作成して、検索インデックスを最新の状態に保ちます。
- システムが生成した Backgrounder タスクで、すべてのコンテンツが適切に同期されていることを定期的に検証し、見つかった不一致を再インデックス化します。

インデックスの再作成

復元プロセスでは、Tableau Server が管理するコンテンツと外部アセットを完全に再インデックスする処理が開始されます。このインデックスの再作成プロセスは CPU リソースを消費するため、バックアップや復元中にその消費が顕著になる可能性があります。

Tableau Server 調整サービス

調整サービス アンサンブルはオープン ソース プロジェクトである Apache ZooKeeper 上に構築され、サーバー上の活動を調整し、障害発生時にはクォーラムを保証し、サーバー トポロジー、校正、状態に関する「真実」の参照元としての役割を担います。このサービスは最初の Tableau Server ノードに自動的にインストールされますが、他のノードを追加しても追加のインスタンスはインストールされません。Tableau Server が順調に機能を発揮するには、調整サービスが適切に機能している必要があるため、3 台以上のサーバー インストール環境では、新しい調整サービス アンサンブルを展開することで調整サービスの追加のインスタンスを加えることを推奨します。これにより、調整サービスのインスタンスの 1 つに問題が発生した場合に冗長性および向上された可用性が提供されます。

クラスタ用ハードウェアは、調整サービスの正常な実行にいくらか影響します。特に:

- メモリ。調整サービスはメモリ内に状態情報を保持します。設計上メモリのフットプリントは小さく、一般的に、サーバーの全体的なパフォーマンスの要因ではありません。
- ディスク速度。サービス ストアはディスク上の情報を示すため、個々のノード コンピューター上のディスク速度が高速であるというメリットがあります。
- ノード間の接続速度。サービスはクラスタ ノード間で継続的に通信します。ノード間の接続速度が速いと、効率的に同期を実行できます。

プロセス	Coordination Service
ステータス	Coordination Service プロセスのステータスは [ステータス] ページに表示されません。ステータスを表示するには、TSM CLI を使用します。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Coordination Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/appzookeeper` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

調整サービスの構成

調整サービスは Tableau Server の最初のノードに自動的にインストールされます。シングルノードインストールの実行中は、調整サービスの展開または構成のために何かを行う必要はありません。インストールに 3 個以上のノードが含まれる場合は、3 番目のノードを追加する際に調整サービスアンサンブルを構成するようにメッセージが表示されます。これは必須ではありません。ただし、調整サービスは高可用性を得るための主要な機能であり、サーバートポロジ、構成、状態に関する「真」の情報源として機能するため、強く推奨されます。

調整サービス アンサンブルを構成するには、TSM CLI を使用し、調整サービスを実行する必要があるノードにこれを追加します。調整サービス アンサンブルを展開する方法についての詳細は、調整サービス アンサンブルの展開を参照してください。

調整サービス クォーラム

調整サービスが適切に動作することを確認するには、サービスにクォーラム、すなわち最低限の数のサービスのインスタンスが必要です。つまり、インストールに含まれるノードの数は、アンサンブルで構成する必要がある調整サービスのインスタンスの数に影響を与えます。

使用する調整サービスインスタンス数

Tableau Server 上のアンサンブルで一度に使用できる調整サービルの最大数は、実装中の Tableau Server ノードの数に依存します。調整サービスアンサンブルをこれらのガイドラインに基づいて構成します。

サーバーノードの合計数	**アンサンブル内の調整サービスノード推奨数 (1、3または5でなければなりません)**	**注**
1-2 ノード	1 ノード	これが既定であり、調整サービスを最初のノードから追加のノードに移動したい場合以外は設定を変更する必要はありません。
3-4	3 ノード	3 ノード調整サービス アンサンブルを使用すると、いずれかの

サーバーノードの合計数	アンサンブル内の調整サービスノード推奨数 (1、3または5でなければなりません)	注
ノード		アンサンブル ノードが失敗しても、Tableau Server で障害を発生することはありません。
5 以上のノード	3 ノード または 5 ノード	インストールできる調整サービスインスタンスの最大数は5台です。3 ノード調整サービス アンサンブルを使用すると、いずれかのアンサンブル ノードが失敗しても、Tableau Server で障害を発生することはありません。5 ノードのアンサンブルを使用すると、2 つのアンサンブル ノードが失敗しても Tableau Server で障害が発生することはありません。 ほとんどのインストールでは、3 つの調整サービス ノードが適切です。調整サービスでは I/O に負荷がかかるため、この構成にすることで、パフォーマンスが最も高くなります。 高可用性が絶対的な優先事項である場合は、5 ノードの調整サービス アンサンブルを展開することを検討してください。これにより、1 つ以上のノードで障害が発生したにもかかわらず、より多くのシステム リソースが必要になる場合であっても、最も高い冗長性を確保できます。(ノード上の他のサービスが、機能しているノードにも存在している限り)、最大 2 つのアンサンブル ノードが Tableau Server に影響を与えることなく失敗する可能性があります。 パフォーマンスへの影響を軽減するには、他のサービスをより少ない数で実行しているノードに調整サービスを配置するか、調整サービス専用 ノードを使用することを検討してください。詳細については、調整サービス専用 ノードで高可用性用を Tableau Server に設定するを参照してください。

ノードの数を減らす場合

クラスタ内のノード数を3つ(またはそれ以上)から2つに減らした場合、Tableau Serverが高可用性をサポートできなくなるという警告メッセージが表示されます。

```
A minimum of three Tableau Server nodes are required for high
availability. You can add a third node now,
or continue with only two nodes. Continuing with only two nodes
means Tableau Server will not be highly available.
You can always add a third node later. Click OK to continue with 2
nodes, or Cancel to go back and add a node.
```

続行すると、Tableau Serverは実行しますが、リポジトリの自動フェールオーバーはありません。

調整サービスのステータスの表示

サーバープロセスステータスの表示したとき、調整サービスはリストに含まれません。サービスの状態を表示するには、tsm statusコマンドを使用します:

```
tsm status -v
```

コマンドの出力は、サービスが実行中であるかどうかを示します。

```
node1: TABLEAUSVR01
Status: RUNNING
'Tableau Server Gateway 0' is running.
'Tableau Server Application Server 0' is running.
'Tableau Server VizQL Server 0' is running.
'Tableau Server VizQL Server 1' is running.
'Tableau Server VizQL Server 2' is running.
'Tableau Server VizQL Server 3' is running.
'Tableau Server Cache Server 0' is running.
'Tableau Server Cache Server 1' is running.
'Tableau Server Coordination Service 0' is running.
'Tableau Server Cluster Controller 0' is running.
'Tableau Server Search And Browse 0' is running.
'Tableau Server Backgrounder 0' is running.
'Tableau Server Backgrounder 1' is running.
```

```
'Tableau Server Data Server 0' is running.
'Tableau Server Data Server 1' is running.
'Tableau Server Data Engine 0' is running.
'Tableau Server File Store 0' is running.
'Tableau Server Repository 0' is running (Active Repository).
'Tableau Server Administration Agent 0' is running.
'Tableau Server Administration Controller 0' is running.
'Tableau Server Service Manager 0' is running.
'Tableau Server License Manager 0' is running.
'Tableau Server Client File Service 0' is running.
'Tableau Server Database Maintenance 0' is stopped.
'Tableau Server Backup/Restore 0' is stopped.
'Tableau Server Site Import/Export 0' is stopped.
'Tableau Server SAML Service 0' is stopped.
```

Tableau Server データ エンジン

Hyper は、サイズの大きいまたは複雑なデータセットでの高速データインジェクトと分析クエリ処理に最適化された、Tableau のインメモリデータエンジン テクノロジーです。Hyper を使用すると、Tableau Server 、Tableau Desktop 、Tableau Cloud、Tableau Public でデータエンジンが強化されます。データエンジンは、抽出を作成、更新、またはクエリするときに使用されます。また、複数の接続を備えたフェデレートされたデータ ソースをサポートするためのクロスデータベース結合でも使用されます。

プロセス	Data Engine
ステータス	Data Engine のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Data Engine プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/hyper` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

メモリとCPU 使用率

データエンジンは、マシン上のすべての使用可能な CPU およびメモリを活用して、最速の応答時間を提供するように設計されています。

CPU 使用率

Hyper テクノロジーは、CPU 内で新しい命令セットを使用し、すべての使用可能なコアの並列化とスケーリングに対応しています。Hyper テクノロジーは、効率的に多数のコアへ拡大するとともに、各単一のコアを可能な限り最大限使用するように設計されています。これは、クエリ処理中に 1 時間あたり平均 75% の合計 CPU 使用率が見込めることを意味します。CPU をさらに追加すると、パフォーマンスの改善につながります。

> **注:** 1 時間平均の使用率は 75% がデフォルトであり、専用サーバー ノードでデータエンジンを実行している場合を除き、変更しないでください。専用 ノードでデータエンジンを実行する方法の詳細については、「抽出 クエリの負荷が大きい環境用に最適化」を参照してください。

Microsoft Windows、Apple macOS、Linux などの最新のオペレーティング システムは、CPU がフルに使用されている場合でも、着信およびその他のアクティブなプロセスを確実に同時実行できるメカニズムを備えています。さらに、全体的なリソース消費量を管理し、マシン上で稼働している他のプロセスの過負荷や完全なリソースの枯渇を防ぐために、データエンジンは Tableau Server リソース マネージャー (SRM) に設定された制限内に留まるように自身を監視します。Tableau Server リソース マネージャーは、リソース消費量を監視し、事前定義された制限を超えた場合は、使用量を削減するようにデータエンジンに通知します。

データエンジンは使用可能な CPU を利用するように設計されているため、時折、CPU 使用率が急増するのは通常動作です。ただし、長時間 (1 時間以上) 高い CPU 使用率 (例: 95%) が続く場合は、次のことを意味しています。

- 高負荷のクエリが存在する。これは、複数のクライアント要求による過負荷のためにサーバーにストレスがかかり、複数のクエリがキューで待機状態になっている場合に発生することがあります。これが頻繁に発生する場合は、クライアントに対応するためにより多くのハード

ウェアが必要であることを示しています。この場合は、より多くの CPU を追加すると、パフォーマンスの改善に役立ちます。

- 1 つのクエリが長時間実行されている。この場合は、Tableau Server リソース マネージャーは、タイムアウト設定に基づいて、長時間実行されているクエリを停止します。このことは、10.5 より前の Tableau Server バージョンにも当てはまります。

Tableau Server リソース マネージャーの詳細については、一般的なパフォーマンスのガイドラインを参照してください。

メモリ使用率

データエンジンのメモリ使用率は、クエリに応答するために必要なデータの量によって異なります。データエンジンは、このインメモリの実行を最初に試みます。ワーキング セット メモリは、クエリ処理中の中間データ構造を保存するために割り当てられます。通常、システムにはこれらのタイプの処理を実行するために十分なメモリがありますが、使用可能なメモリが十分にない場合や RAM の 80% 超が利用されている場合は、データエンジンは一時的にディスクに書き込むことでスプーリングに移行します。一時ファイルはクエリの回答後に削除されます。つまり、スプーリングは、より多くのメモリが必要である可能性を示しています。スプーリングによって引き起こされるパフォーマンス問題を回避するために、メモリ使用率を監視し、適切にアップグレードする必要があります。

マシン上のメモリ リソースを管理するために、データエンジンの最大メモリ制限が Tableau Server リソース マネージャー (SRM) によって設定されています。

サーバー構成、スケーラビリティ、パフォーマンス

- Tableau Server にファイル ストア、アプリケーション サーバー (VizPortal)、VizQLServer、データ サーバー、またはバックグラウンダーのインスタンスがインストールされているノードごとに、データエンジンの単一のインスタンスが自動的にインストールされます。データエンジンはそれ自身で拡大し、必要なだけ CPU とメモリを使用できるため、データエンジンの複数のインスタンスを保持する必要はなくなりました。サーバープロセスの詳細については、Tableau Server プロセスを参照してください。

- ファイル ストアがインストールされているノードにインストールされたデータエンジンのインスタンスは、ビューの要求のデータをクエリするために使用されます。バックグラウンダーがインストールされているノードにインストールされたデータエンジンのインスタンスは抽出の作成と更新のために使用されます。これは、パフォーマンス調整を行う際の重要な考慮事項です。詳細については、「パフォーマンスの調整」を参照してください。

- データサーバー、VizQL Server、およびアプリケーション サーバー (VizPortal) はすべて、データエンジンのローカル インスタンスを使用して、クロスデータベース結合を実行し、シャドー抽出を作成します。シャドー抽出ファイルは、非レガシーの Excel またはテキスト、もしくは統計ファイルに基づいたワークブックを操作している場合にのみ作成されます。Tableau は、データをよりすばやく読み込むために、シャドー抽出ファイルを作成します。

- Tableau Server 10.5 では、バックグラウンダーをインストールすると、データエンジンの 1 つのインスタンスが自動的にインストールされます。バックグラウンダープロセスは、同じノードにインストールされているデータエンジンの単一のインスタンス (hyperd.exe) を使用します。

重要! データエンジンがファイル ストアと同じノードにインストールされる場合には例外があります。ファイル ストアが Tableau Server の外部で構成されている場合、データエンジンがファイル ストアと共にインストールされることはなくなりました。ただし、Tableau Server が外部ファイル ストアを使用して構成されている場合でも、その他のプロセスをインストールした場合は、データエンジンが引き続き上記のようにインストールされます。また、ファイル ストアが外部で構成されている場合は、他のプロセスのないノードでデータエンジンを構成することもできます。外部ファイル ストアの詳細については、Tableau Server 外部ファイル ストアを参照してください。

スケーラビリティ:

新しいデータエンジンをスケール アップすることができます。コアはフルに活用されるため、より多くのコアを追加することで個々のクエリの実行が迅速化され、より多くのクエリをより短い時間で実行できます。

スプーリングによって引き起こされるパフォーマンス問題を回避するために、メモリ使用率を監視し、適切にアップグレードする必要があります。

スケーラビリティの詳細については、「Tableau Server のスケーラビリティ (英語)」を参照してください。

パフォーマンス:

パフォーマンスにおけるメリット

10.5 以降、Hyper テクノロジーが Tableau データエンジンに統合され、次の主なメリットを提供しています。

- **より高速な抽出作成:** Hyper テクノロジーでは、ソース システムがデータを配信できる速度とほぼ同じ速度で抽出が生成されます。データを並べ替える必要はありません。
- **より大きなサイズの抽出をサポート:** このリリースより前は、すべてのデータを単一の抽出に取り込むことができない場合がありました。Hyper テクノロジーを使用すると、単一の抽出としてはるかに大量のデータを含めることができます。
- **より高速な抽出分析:** 多くの場合、サイズの大きい抽出または複雑な計算のワークブックでも、データのクエリが迅速化されます。

Hyper により強化されたデータエンジンがサイズの大きいまたは複雑な抽出でより高性能を発揮し、より高速なクエリに最適化されている理由は次のとおりです。

- **Hyper テクノロジーは、より高速にデータを使用するように設計されています**。以前のバージョンとは異なり、データエンジンは、並べ替えなどの後処理を実行しません。Hyper では、データエンジンがサイズの大きい抽出でもより効率的に実行できるため、並べ替えなどの後処理は必要ありません。
- **Hyper テクノロジーは メモリ最適化されています**。つまり、必要な場合は、すべてのデータがメモリに内包されます。この結果、データアクセス時間が高速化します。
- **Hyper テクノロジーは CPU 最適化されています**。つまり、データエンジンでは クエリ実行が完全に並列化されており、クエリ実行時間はマシンのコア数に応じてほぼ直線的に加速されるような方法で使用可能な CPU が使用されます。
- **Hyper は コンパイル クエリ エンジンです**。パフォーマンスを最大化するためにクエリはマシンコードに解釈またはコンパイルされるため、データエンジンは最新のハードウェア (CPU、大容量 メイン メモリ) の最高のパフォーマンスを引き出すことができます。

- **Hyper テクノロジーは、高度なクエリ最適化を使用して、クエリを高速化します。**各列の最小値と最大値のマテリアライズ、検索範囲を最適化するミニインデックス、よりきめ細かいデータブロックレベルのディクショナリ、結合とサブクエリのパフォーマンスを最適化するための高度なロジックなどの多数の追加された高度な技術とともに、新しいデータエンジンではパフォーマンスとスケーラビリティの点で以前の Tableau データエンジンに多数の改善が加えられています。

パフォーマンスの詳細については、最初に一般的なパフォーマンスのガイドラインおよびパフォーマンスの調整を参照してください。

Tableau Server の Data Server

データサーバーは、Tableau Server 上のパブリッシュされたデータソースへの接続を管理します。Data Server の可用性を高くするには、1 つ以上の Data Server プロセスをクラスタの複数のノードで実行するように構成します。

プロセス	Data Server
ステータス	Data Server のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Data Server プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/dataserver` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

Data Server のプロセスが失敗するとどうなりますか? Data Server プロセスを介して実行されているクエリではエラーが発生し、ビューのレンダリング、抽出更新、またはアラートの失敗に繋がります。失敗した操作の再試行を含むその後の要求は、機能している Data Server が再ルーティングされた要求を受け入れることができる限り成功します。

Tableau Server の機能は Data Server に依存していませんが、実行中の Data Server がないと、サーバー上のワークブックではパブリッシュされたデータソースに対してクエリを実行したり、接続したりする機能が失われます。それらのデータソースで Data Server を使用しないビューは、引き続き正常に機能します。

Tableau Server データ ソース プロパティ サービス

バージョン 2020.1.0 で導入された Tableau Server データ ソース プロパティ サービスは、アプリケーション サーバー (VizPortal) から「データに聞く」などのクライアント サービスにパブリッシュされたデータ ソースのメタデータを提供します。

注: 「データに聞く」機能とメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。

データ ソース プロパティ サービスの 1 つのインスタンスは、Tableau Server の初期ノードで構成されます。初期ノードや分散インストールの他のノードにインスタンスを追加できます。アプリケーション サーバーのインスタンスが含まれるノードでは、データ ソース プロパティ サービスを少なくとも 1 つ構成する必要があります。

プロセス	データ ソース プロパティ サービス
ステータス	データ ソース プロパティ サービスのステータスは、 `tsm status -v` コマンドを使用して [ステータス] ページとコマンド ラインから表示されます。詳細については、サーバー プロセス ステータスの表示を参照してください。
ロギング	データ ソース プロパティ サービスによって生成されたログは、`/var/opt/tableau/tableau_server/data/tabsvc/logs/tdservice` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

データ ソース プロパティ サービスのログ ファイル

データ ソース プロパティ サービスでは、次の 2 つのログ ファイル セットを作成します。

- `control-tdservice*.log`。これらのログには、開始したり有効にしたりするサービスに関する情報が含まれます。
- `tdservice_*.log`。エラーや問題は、ここに記録されます。

詳細については、ログ ファイルのスナップショット (ログのアーカイブ) を参照してください。

マルチノード クラスタのデータ ソース プロパティ サービス

マルチノード クラスタでデータ ソース プロパティ サービスを設定する方法は、ノードの構成方法によって異なります。Tableau Server のインストール時には、サービスのインスタンスが最低 1 つ必要です。また、アプリケーション サーバー (VizPortal) で構成されているノードでは、データ ソース プロパティのインスタンスを少なくとも 1 つ構成する必要があります。

Tableau Server ファイル ストア

このトピックでは、Tableau Server でローカルに実行されるようにファイル ストア プロセスを構成した場合について説明します。ただし、ファイル ストアはローカルだけでなくTableau Server の外部でも実行できます。Tableau Server 外部ファイル ストアの詳細については、Tableau Server 外部ファイル ストアを参照してください。

Tableau Server File Store プロセスは、抽出のストレージを管理します。ファイル ストアのインストール時に、ノードにデータ エンジンのインスタンスがなければ、Data Engine のインスタンスもインストールされます。高可用性 (HA) 環境では、File Storeは抽出が他のファイル ストア ノードと同期されていることを確認し、1 つのファイル ストア ノードが停止した場合に利用できるようにします。

プロセス	File Store
ステータス	File Storeのステータスは [ステータス] ページに表示されます。詳細については、サーバー プロセス ステータスの表示を参照してください。
ロギング	File Store プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/filestore` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

decommission コマンド

ファイル ストア インスタンスを削除する場合は、`tsm topology filestore decommission` コマンドを使用して最初にファイル ストアのコミッションを解除する必要があります。ファイル ストアのコミッションを解除せずに削除しようとした場合、解除するようにメッセージが表示されます。コミッションを解除すると、ファイル ストア インスタンスは読み取り専用モードになり、インスタンス内に含まれる一意のデータはクラスタ内の別のファイル ストアへコピーされます。ファイル ストア インスタンスのコ

ミッションが解除される間、これらのデータは [ステータス] ページに表示されますが、一意のコンテンツがすべて別のファイル ストア ノードにコピーされると、コミッション解除されたノードは削除準備完了として表示されます。

ファイル ストアが管理コントローラーと同じ場所にある場合のコミッション解除

Tableau のバックアップが最も速いのは、ファイル ストアのインスタンスが TSM 管理コントローラーと同じノードにある場合です。コントローラーと同じ場所にあるファイル ストアのインスタンスを削除する場合は、バックアップのパフォーマンスへの影響に関する警告が表示されます。

Tableau Server ゲートウェイ プロセス

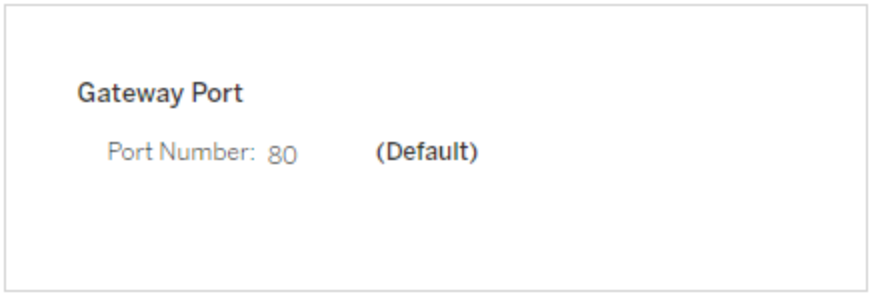
Tableau Server ゲートウェイ プロセスは Apache Web サーバー コンポーネント (`httpd.exe`) です。ロールは、すべてのクライアント (Tableau Desktop、モバイル デバイス、プロキシ、ロード バランサーなど) からサーバーへの要求を処理することです。

サーバーはゲートウェイ プロセスの 1 つのインスタンスを実行します。マシン 1 台に対して複数のインスタンスを実行することはできません。ゲートウェイ プロセスは VizQL Server または Vizportal のインスタンスがあるノードで必要です。

プロセス	ゲートウェイ
ステータス	Gatewayのステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	リポジトリが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/httpd` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

ポートの割り当て

既定ではゲートウェイ プロセスはポート 80 (HTTP 要求の場合) および 443 (SSL 要求の場合) 上の要求をリスンします。コンピューターに Tableau Server をインストールすると、サーバー構成の一部は、コンピューターのファイアウォールでこのポートが開いていることを確認します。コンピューターが HTTP に対してポート 80 を要求する別のプロセスを実行中の場合、インストール中にゲートウェイ プロセスに対するポートの割り当てを変更できます (SSL/HTTPS ポートを変更することはできません)。



インストール後に HTTP ポートの設定が必要な場合には、以下の TSM コマンドを実行します。ここで、*node* はノード番号 (`node1` など)、*nn* は新しいポート番号です。

```
tsm topology set-ports --node-name node --port-name gateway:primary
--port-value nn
```

リバースプロキシ サーバーを使用して Tableau Server を実行している場合は、Tableau Server で他のポートおよびホストに関連した設定を構成する必要があります。Tableau Server のプロキシとロード バランサーの設定を参照してください。

ゲートウェイ プロセスのログ ファイル

ゲートウェイ プロセスにより以下の 2 つのセットのログ ファイルが作成されます。

- アクティビティ ログ。これらのログ ファイルの名前の書式は `access.yy_mm_dd_hh_mm_ss.log.` のようになります。
- エラー ログ。すべてのエラーは `error.log` という名前の単一のファイルに記録されます。

詳細については、ログ ファイルのスナップショット (ログのアーカイブ) を参照してください。

クラスターのゲートウェイ プロセス

サーバー環境が複数のマシンに分散されている場合、クラスターの各ノード上で単一のゲートウェイ プロセスを実行できます。クラスター内で複数のコンピューター上のゲートウェイ プロセスを実行する最も一般的なシナリオとして、クラスターの前にロード バランサーがあるものです。このシナリオでは、ロード バランサーはクラスター内の任意のゲートウェイに要求を配信します。ノードをオフラインにする必要がある場合 (例: そのノードのメンテナンスを実行するため、そのマシンに対するロード バラン

サーのルーティングを無効にできます。メンテナンスが完了したら、ロードバランサーでノードを再度有効化できます。

クラスター内の 1 台以上のコンピューター上でゲートウェイプロセスを実行する必要があります。プライマリサーバーからゲートウェイプロセスを削除する場合、クラスター内の他のコンピューターがゲートウェイプロセスを実行していることを確認する必要があります。また、クライアントがコンピューターにアクセス可能なことを確認する必要があります。

VizQL サーバー、Vizportal、Tableau Prep Flow Authoring のいずれかのプロセスやサービス用に構成されているノードでは、ゲートウェイプロセスのインスタンスが必要です。

Tableau Server が SSL を使用するよう構成されている場合、SSL サポートに対する証明書が、ゲートウェイプロセスを実行しているすべてのコンピューター上の同じ場所にあることを確認する必要があります。SSL の使用の詳細について、「Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する」を参照してください。

同様に、サーバー インストールがカスタム ロゴを使用している場合、ロゴはゲートウェイプロセスを実行するすべてのコンピューター内で同一の場所に置かれる必要があります。

ゲートウェイプロセスがリスンするポートの番号を変更する場合は、構成ダイアログボックスを使用するか、ゲートウェイプロセスを実行している各 ワーカー コンピューターに対して次のコマンドを実行できます。

```
tsm topology set-ports --node-name node --port-name gateway:primary
--port-value nn
```

追加情報

Tableau Server のプロキシとロードバランサーの設定

ロードバランサーの追加

インデックス作成/検索サーバー

OpenSearch に基づいた Tableau Server インデックス作成/検索サーバープロセスは、Tableau Server サイト上のコンテンツ メタデータの高速検索、フィルター、取得、および表示を処理しま

す。Tableau Server バージョン 2023.3 以降より、Index and Search Server は、検索および参照プロセスを完全に置き換えます。

サーバー構成

インデックス作成/検索サーバーは最初のノードに自動的にインストールされます。

マルチノード構成

高可用性のためにインデックス作成/検索サーバーを構成するには、複数のノードでプロセスを構成します。奇数のインデックス作成/検索サーバー インスタンスを構成することをお勧めします。3 つ以上のノードを持つ Tableau Server クラスターでは、少なくとも 3 つの異なるノードでインデックス作成/検索サーバーを構成する必要があります。

プロセス	Index and Search Server
ステータス	Index and Search Serverのステータスはステータスページに表示されますが、TSM CLI を使用してアクセスすることもできます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Index and Search Server が生成したログは、`/var/opt/tableau/tableau_server/data/tabsvc/logs/indexandsearchserver` にあります。詳細については、「Tableau Server ログとログ ファイルの場所」を参照してください。

インデックス作成/検索サーバーに障害が発生するとどうなりますか?

インデックス作成/検索サーバーに障害が発生してもユーザーは Tableau Server にサインできますが、異なるタイプのコンテンツ (ワークブック、プロジェクト、データ ソース、ビューなど) は、ホームページ、検索バー、探索ページに表示されなくなります。既存のワークブックとビューは、ブックマークされた URL からであればアクセスでき、完全に機能するでしょう。コンテンツは実際に存在しないわけではなく、検索結果に返されないため、インデックス作成/検索サーバーが再起動した後にもう一度表示されるようになります。障害が発生したインデックス作成/検索サーバーは、コンピューター自体が正常に稼働している限り、自動的に再起動されます。

インデックス作成/検索サーバーが 2 つ以上構成されており、障害が発生したときに複数のノードで実行中の場合、障害が発生したインデックス作成/検索サーバーに対して行われた要求も失敗し

ますが、その後の要求は機能しているインデックス作成/検索サーバーにルーティングされます。各インデックス作成/検索サーバー インスタンスは、クラスター内のすべてのノードにわたってインデックスを作成するため、1 つのインデックス作成/検索サーバー インスタンスが実行されている限り、すべてのノードにわたって結果が返されます。

パフォーマンスの調整

Index and Search Server のヒープ サイズは、`indexandsearchserver.vmopts` オプションを指定した `tsm set configuration` コマンドを使用して構成できます。詳細については、indexandsearchserver.vmoptsを参照してください

インデックスの再作成

復元プロセスでは、Tableau Server が管理するコンテンツと外部アセットを完全に再インデックスする処理が開始されます。このインデックスの再作成プロセスは CPU リソースを消費するため、バックアップや復元中にその消費が顕著になる可能性があります。

Tableau Server 内部データ ソース プロパティ サービス

バージョン 2020.1.0 で導入された Tableau Server 内部データ ソース プロパティ サービスは、データ ソース プロパティ サービスと通信します。これは Tableau Server によって内部的に管理され、管理者が構成することはできません。

内部データ ソース プロパティ サービスのインスタンスは、データ ソース プロパティ サービスのインスタンスがあるノードで自動的に構成されます。

プロセス	内部データ ソース プロパティ サービス
ステータス	内部データ ソース プロパティ サービスの状態は、[ステータス] ページに表示されるほか、コマンド ラインで `tsm status -v` コマンドを使用して確認できます。詳細については、サーバー プロセス ステータスの表示を参照してください。
ロギング	内部データ ソース プロパティ サービスによって生成されたログは、`/var/opt/tableau/tableau_server/data/tabsvc/logs/tdsnativeservice` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

内部データソースプロパティサービスのログファイル

内部データソースプロパティサービスでは、次の 2 つのログファイルセットが作成されます。

- `control-tdsnativeservice_*.log`。これらのログには、開始したり有効にしたりするサービスに関する情報が含まれます。
- `nativeapi_tdsnativeservice_*.log`。エラーや問題は、ここに記録されます。

詳細については、ログファイルのスナップショット(ログのアーカイブ)を参照してください。

マルチノードクラスタでの内部データソースプロパティサービス

データソースプロパティのインスタンスが構成されている各ノードに、内部データソースプロパティサービスの 1 つのインスタンスが追加されます。内部データソースプロパティサービスは Tableau によって内部的に管理されるため、管理者がこのサービスに関連して構成できることはありません。

Tableau Server メッセージング サービス

Tableau Server メッセージング サービスは、バージョン 2019.4 以降の Apache ActiveMQ を使用します。これは、マイクロサービスのセキュリティで保護された、スケーラブルで、パフォーマンスが高く、可用性の高いメッセージ指向の通信を可能にするパブリッシュ/サブスクライブプラットフォームです。メッセージング サービスは、Tableau Server のマイクロサービス間の通信をサポートするために使用されます。

サーバーが実行する既定のメッセージング サービスは、単一のインスタンスです。バージョン 2020.1 以降、マルチノードの Tableau Server インスタンスがある場合、メッセージング サービスで 2 番目のインスタンスを設定できます。

プロセス	メッセージング サービス
ステータス	メッセージング サービスの状態は、 `tsm status -v` コマンドを使用して [状態] ページとコマンドラインから表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	リポジトリが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/activemqserver` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

メッセージング サービスが正常に実行されていない場合の影響

メッセージング サービスが停止したときや、起動に失敗したときの影響は、メッセージング サービスのインスタンスを 1 つ構成しているか、または 2 つ構成しているかによって異なります。

メッセージング サービスの複数のインスタンス (マルチノード、バージョン 2020.1 以降)

バージョン 2020.1 以降があり、Tableau Server のマルチノード インストールがある場合は、メッセージング サービスのインスタンスを 2 つ設定できます。この場合、いずれかのインスタンスに障害が発生した場合、もう一方のインスタンスがすべてのトラフィックを処理するため、障害を起こしたインスタンスを削除してインストールし直すことができます。

メッセージング サービスの 1 つのインスタンス

Tableau Server をシングル ノードでインストールしている場合、メッセージング サービスのインスタンスは 1 つに制限されます。この場合、構成されたインスタンスに障害が発生すると、Tableau Server は引き続き機能しますが、ステータスは「低下」と表示され、サービスが再起動しても、ダウン中にメッセージング サービスに送信されたイベント メッセージは失われる可能性があります。イベント メッセージには、プロジェクトの権限の変更が含まれます。

メッセージング サービスが停止しても、コンピューター自体に問題がなければ自動的に再起動します。

マルチノード クラスター内のメッセージング サービス

マルチノード クラスターでメッセージング サービスを構成する方法は、実行している Tableau Server のバージョンによって異なります。いずれの場合も、メッセージング サービスのインスタンスを 1 つ実行する必要があります。

バージョン 2020.1 以降では、マルチノード環境でサービスのインスタンスを 2 つ実行できるようになりました。これにより冗長性が提供されるため、2 つのインスタンスを実行することをお勧めします。メッセージング サービスがまだ構成されていない任意のノードに、2 番目のインスタンスを追加できます。マルチノード クラスターでは、合計で 2 個を超えるインスタンスを追加することはできません。

Tableau Server メトリクス サービス

従来のメトリクス機能の廃止

Tableau の従来のメトリクス機能は、2024 年 2 月の Tableau Cloud および Tableau Server バージョン 2024.2 で廃止されました。2023 年 10 月に、Tableau は Tableau Cloud および Tableau Server バージョン 2023.3 に従来のメトリクスを埋め込む機能を廃止しました。Tableau Pulse では、メトリクスを追跡し、データについて質問するためのエクスペリエンスが向上しました。新しいエクスペリエンスの詳細については、「Tableau Pulse を使用したメトリクスの作成」を参照し、廃止される機能のについては、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。

メトリクス サービスは、Tableau Server でメトリクス データの読み取りと書き込みを行います。メトリクスが正しく機能するためには、このサービスが必要です。メトリクス サービスの可用性を高くするには、1 つまたは複数のサービスのインスタンスをクラスタの複数のノードに構成します。アプリケーション サーバー (VizPortal) を実行している各ノードに、最低 1 つのインスタンスを設定することをお勧めします。

プロセス	Metrics Service
ステータス	Metrics Service のステータスは、`tsm status -v` コマンドを使用して [TSM ステータス] ページとコマンド ラインから表示されます。詳細については、サーバー プロセス ステータスの表示を参照してください。
ロギング	Metrics Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/metrics` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

メトリクス サービスに障害が発生した場合の影響

メトリクス サービスは、Tableau Server でメトリクスが正しく機能するために必要となります。問題を最小限に抑えるには、サーバーのインストール時にサービスの複数のインスタンスをインストールします。Tableau Server のシングル ノード インスタンスがある場合は、ノード上で最低 2 つのサービス インスタンスを設定します。メトリクス サービスが動作を停止すると、処理中のメトリクスの要求がすべて失敗します。後続の要求は、正常なサービスのインスタンスにルーティングされます。

マルチノード クラスタ内のメトリクス サービス

メトリクス サービスは、Tableau Server でメトリクスが正しく機能するために必要となります。問題を最小限に抑えるには、サーバーのインストール時にサービスの複数のインスタンスをインストールしま

す。マルチノード インストールでは、アプリケーション サーバー (VizPortal) を実行している各ノードにメトリクス サービスのインスタンスを最低 1 つ構成することをお勧めします。これにより、冗長性と最高のパフォーマンスが得られます。

メトリクス サービスのログ ファイル

メトリクス サービスにより以下の 2 つのセットのログ ファイルが作成されます。

- `control-metrics*.log`。これらのログには、開始したり有効にしたりするサービスに関する情報が含まれます。
- `metrics_*.log`。エラーや問題は、ここに記録されます。

詳細については、ログ ファイルのスナップショット (ログのアーカイブ) を参照してください。

Tableau Server マイクロサービス コンテナー

デフォルトにより、インタラクティブなマイクロサービスコンテナーの 1 つのインスタンスがアプリケーションサーバーが(Vizportal)インストールされているすべてのノードに追加され、インタラクティブでないマイクロサービスコンテナーの 1 つのインスタンスがバックグラウンダーがインストールされているすべてのノードに追加されます。マイクロサービス コンテナーを直接追加することはできませんが、必要に応じて、TSM CLI を使用して両方のマイクロサービス コンテナーのインスタンス数を変更することができます。バックグラウンダーまたはアプリケーション サーバーのすべてのインスタンスがノードから削除されると、コンテナー プロセスも削除されます。

マイクロサービス コンテナーとマイクロサービス:

- インタラクティブなマイクロサービス コンテナー:
    - MessageBus マイクロサービス
    - 関係クエリ マイクロサービス
    - 認証資格情報サービス
    - リコメンデーション マイクロサービス (バージョン 2023.3.3 で追加されました)
- インタラクティブでないマイクロサービス コンテナー:
    - 関係インジェスター マイクロサービス
    - 外部コンテンツ プロバイダー マイクロサービス

- フロープロバイダー マイクロサービス
- コンテンツプロバイダー マイクロサービス

マイクロサービス コンテナー ステータスの表示

マイクロサービス コンテナープロセスのステータスは、[TSM ステータス] ページから、または `tsm status -v` コマンドを使用してコマンドラインから表示できます。[TSM ステータス] ページを使用してサーバープロセス ステータスの表示を行う場合、各コンテナープロセスのステータスは表示されますが、コンテナーに含まれるマイクロサービスのステータスはいずれも表示できません。コマンドラインを使用すると、個々のマイクロサービスのステータスなど、他の詳細も表示されます。

マイクロサービス コンテナー ステータス

コンテナープロセスのステータスは、コンテナーに含まれるマイクロサービスのステータスによって決まります。コンテナープロセスに含まれるすべてのマイクロサービスが想定どおりに実行している場合、コンテナー ステータスは [アクティブ] ([TSM ステータス] ページで) または `running` (TSM コマンドラインから表示する場合) となります。コンテナープロセスに含まれるすべてのマイクロサービスが停止する場合、コンテナーのステータスは [エラー] ([TSM ステータス] ページで) または `stopped` (TSM コマンドラインで) となります。マイクロサービスが停止したものの、他のマイクロサービスが 1 つ以上実行中の場合、コンテナー ステータスは [劣化] ([TSM ステータス] ページで) または `degraded` (TSM コマンドラインで) となります。

コンテナープロセスに含まれるすべてのマイクロサービスのステータスが実行中の場合、コンテナー ステータスは `Active` となります。コンテナーに含まれるマイクロサービスのいずれかがエラー状態の場合 (ステータスが `stopped`)、コンテナープロセスのステータスは `degraded` となります。コンテナーに含まれるすべてのマイクロサービスがエラー状態の場合、コンテナーのステータスは `error` となります。

TSM Web インターフェイスの使用

[TSM ステータス] ページからマイクロサービス コンテナー ステータスを表示するには、次を実行します。

1. ブラウザーで TSM を開きます。

   http://<tsm-computer-name>:8850

2. **[ステータス]** をクリックします。

   ページにインタラクティブなマイクロサービス コンテナーとインタラクティブでないマイクロサービス コンテナーのプロセスのステータスのほか、TSM または Tableau Server の一部として実行している他のプロセスのステータスが表示されます。

   コンテナープロセスに含まれる個々のマイクロサービスのステータスを表示することはできませんが、コンテナープロセスのステータスが [アクティブ] (緑色のチェック) の場合、含まれるマイクロサービスはいずれも想定どおりに実行しています。個々のマイクロサービスのステータスを表示するには、TSM コマンド ラインを使用してください。

   > **注**: コンテナープロセスのステータスは、以前の Tableau Server のステータス ページには表示されません。2 つのステータス ページとその違いの詳細については、サーバープロセス ステータスの表示を参照してください。

TSM CLI の使用

TSM コマンド ラインからマイクロサービス コンテナー ステータスを表示するには、次を実行します。

1. 最初のノード (TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。

2. 次のコマンドを実行します。

   ```
   tsm status -v
   ```

   コマンドからの出力に、コンテナー サービスとそれに含まれるマイクロサービスのステータスが次のように表示されます。

   ```
   node1: TABLEAUSVR01
   Status: RUNNING
   'Tableau Server Gateway 0' is running.
   'Tableau Server Application Server 0' is running.
   'Tableau Server Interactive Microservice Container 0' is
   ```

```
running.
    'MessageBus Microservice 0' is running.
    'Relationship Query Microservice 0' is running.
'Tableau Server VizQL Server 0' is running.
'Tableau Server VizQL Server 1' is running.
'Tableau Server VizQL Server 2' is running.
'Tableau Server VizQL Server 3' is running.
'Tableau Server Cache Server 0' is running.
'Tableau Server Cache Server 1' is running.
'Tableau Server Coordination Service 0' is running.
'Tableau Server Cluster Controller 0' is running.
'Tableau Server Search And Browse 0' is running.
'Tableau Server Backgrounder 0' is running.
'Tableau Server Backgrounder 1' is running.
'Tableau Server Non-Interactive Microservice Container 0' is
running.
    'Relationship Ingestor Microservice 0' is running.
    'External Content Provider Microservice 0' is running.
    'Flow Provider Microservice 0' is running.
    'Content Provider Microservice 0' is running.
'Tableau Server Data Server 0' is running.
'Tableau Server Data Server 1' is running.
'Tableau Server Data Engine 0' is running.
'Tableau Server File Store 0' is running.
'Tableau Server Repository 0' is running (Active Repository).
'Tableau Server Tableau Prep Conductor 0' is running.
'Tableau Server Elastic Server 0' is running.
'Tableau Server Ask Data 0' is running.
'Tableau Server Administration Agent 0' is running.
'Tableau Server Administration Controller 0' is running.
'Tableau Server Service Manager 0' is running.
'Tableau Server License Manager 0' is running.
'Tableau Server Client File Service 0' is running.
'Tableau Server Database Maintenance 0' is stopped.
'Tableau Server Backup/Restore 0' is stopped.
```

```
'Tableau Server Site Import/Export 0' is stopped.
'Tableau Server SAML Service 0' is stopped.
c:\Program Files\Tableau\Tableau
Server\packages\scripts.near.18.1216.1859>
```

Tableau Server リポジトリ

Tableau Server リポジトリは、サーバーデータを格納するデータベースです。このデータには、Tableau Server のユーザー、グループとグループ割り当て、パーミッション、プロジェクト、データソース、ワークブック、抽出のメタデータと更新に関する情報が含まれています。

リポジトリは *PostgreSQL* リポジトリまたはデータベースとも呼ばれます。

重要: リポジトリを直接変更しないでください。Tableau Server が破損するおそれがあります。お客様がリポジトリを変更した場合、そのインストールは Tableau でサポートされなくなります。

プロセス	リポジトリ
ステータス	リポジトリのステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	リポジトリが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/pgsql` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

優先するアクティブなリポジトリ

Tableau Server を構成するとき、優先アクティブ リポジトリとしてノードを指定することができます。Tableau Server がリポジトリフェールオーバー用に構成されている場合、この優先アクティブ リポジトリ ノードがアクティブ リポジトリ用に使用されます。このステップはオプションです。優先アクティブ リポジトリ ノードを指定しない場合、Tableau Server は起動時にアクティブ リポジトリ ノードを選択します。

優先アクティブ リポジトリを構成するには、tsm configuration set コマンドを使用して、pgsql.preferred_host オプションを次のように構成します。

```
tsm configuration set -k pgsql.preferred_host -v "<host_name>"
```

> **注:** host_name は大文字と小文字が区別され、tsm status -v の出力に表示されるノード名と一致する必要があります。

Tableau Server が起動時に特定のノードを選択するようにする場合は、優先するアクティブなリポジトリを設定します。アクティブなリポジトリ(たとえば、ディスク容量やメモリの大きいコンピューター)に特定のサーバーを使用する場合や、カスタムの管理ビューを使用している場合、この操作が必要となる場合があります。カスタム管理ビューには、ビューを作成したリポジトリを参照する接続情報が埋め込まれています。Tableau Server リポジトリへの接続の詳細については、Tableau Server リポジトリでデータを収集するを参照してください。

failoverrepository コマンド

フェールオーバーが発生し、パッシブ リポジトリがアクティブなリポジトリになった場合、そのリポジトリは Tableau Server を再起動するか、tsm topology failover-repository コマンドを使用して元に戻すまでアクティブなままとなります。アクティブにしたいリポジトリを選択するか、優先するアクティブなリポジトリ(設定している場合) を指定するともう一度アクティブになります。詳細については、tsm topology failover-repositoryを参照してください。

Tableau Server リソース制限 マネージャー

このプロセスは、Tableau Server 2022.1 で導入されました。

Tableau Server リソース制限 マネージャーは、設定されたリソース制限に対するバックグラウンダー リソースの使用状況を追跡し、リソース制限が正しく適用されていることを確認します。

プロセ	Resource Limits Manager

ス	
ステータス	Resource Limits Manager のステータスは、TSM CLI にのみ表示されます。
ロギング	Content Exploration Service が生成したログは、`/var/opt/tableau/tableau_server/data/tabsvc/logs/allegro` にあります。 • 構成ログは、バックグラウンダーログに含まれています • リソース消費ログは vizportal ログに含まれています。 詳細については、「Tableau Server ログとログファイルの場所」を参照してください。

リソース制限マネージャーは、既定で Tableau Server の初期ノードに自動的にインストールされます。Tableau Server の追加ノードで、プロセスを追加したり、これを構成したりすることはお勧めしません。

最大メモリ使用量は、512 MB に設定されています。

リソース制限マネージャーに障害が発生するとどうなりますか?

リソース制限は適用されなくなりますが、ジョブは利用可能なバックグラウンダーリソースを使用して引き続き実行されます。この動作は、リソース制限が設定されていないシナリオに似ています。

ステータスは、tsm コマンド --tsmstatus-v を使用してを確認できます

Tableau Server SAML サービス

サイト固有の SAML が有効になっている Tableau Server インストールでは、アプリケーションサーバーが構成されている各ノードで実行中の SAML サービスインスタンスもあります。このサービスは、サイト固有の SAML がサーバーで有効になっている場合に自動的に構成されます。

- バージョン 2023.1.x 以降では、サイト SAML が有効になるまで、Tableau Server の SAML サービスは表示されません。
- 2022.3.X 以降では、サイトの SAML が有効になっていない場合、Tableau Server の SAML サービスは停止と表示されます。

プロセス	SAML Service
ステータス	SAML Service プロセスのステータスは [ステータス] ページに表示されません。ステータスを表示するには、TSM CLI を使用します。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	SAML Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/samlservice` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

このプロセスが失敗した状態になると、ユーザーの要求がこのノードでアプリケーション サーバーにルーティングされる場合、ユーザーは Tableau Server にサインインできなくなります。他のプロセスと同様、SAML サービスを含むノードでエラーが発生すると、失敗したプロセスは数秒以内に自動的に再起動します。

Tableau Server の検索 & 参照

注: バージョン 2023.3 以降、検索と参照は廃止されました (インストールされなくなりました)。検索と参照は、インデックス作成/検索 サーバーに置き換わりました。

バージョン 2022.3 以降、検索と参照は非推奨になりました (インストールされますが、Tableau Server では使用できません)。Tableau Server バージョン 2022.3 または 2023.1 を実行している場合は、どのインストールでも*検索と参照の複数のインスタンスを構成しないでください*。インスタンスを複数構成すると、まれに安定性の問題が発生することがあります。

Apache SOLR ベース (Tableau Server バージョン 2020.4 以前) の検索および参照プロセスは検索 サーバーとも呼ばれ、Tableau Server サイト上のコンテンツ メタデータの高速検索、フィルター、取得、および表示を処理します。検索 & 参照のプロセスで高可用性を構成するには、複数のノードでプロセスを構成します。Tableau Server バージョン 2021.1 以降、検索および参照プロセスの機能は、コンテンツ探索 サービスによって強化されました。詳細については、Tableau Server コンテンツ探索 サービスを参照してください。

プロセ	Search & Browse

ス	
ステータス	Search & Browse のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Search & Browse プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/searchserver` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

検索 & 参照のプロセスが失敗するとどうなりますか? ユーザーは Tableau Server には引き続きサインインできますが、ワークブックのコンテンツは表示されなくなります。コンテンツは実際に存在しないわけではありません。そうではなく、コンテンツは検索結果に返されないため、検索 & 参照のプロセスが再起動した後に再度表示されるようになります。

検索 & 参照のプロセスが 2 つ以上構成されており、エラーが発生したときに複数のノードで実行中の場合、エラーが発生した検索 & 参照プロセスに対して行われた要求も失敗しますが、その後の要求は機能している検索 & 参照プロセスにルーティングされます。それぞれの検索 & 参照プロセスはクラスタ内のすべてのノードでインデックスを作成するため、すべての検索 & 参照プロセスのうち 1 つでもエラーが発生していないものがあれば、すべてのノードで引き続き結果が返されます。

検索と参照のプロセスのチューニング

Tableau Server での検索と参照のプロセスの動作を変更する方法はいくつかあります。ほとんどの場合、何も変更する必要はありませんが、以下のいくつかの構成変更を行うことができます。

最大ヒープメモリの検索と参照

`searchserver.javamemopts` コンフィギュレーションキーを使用して、検索と参照のオプションを構成できます。構成できる最も重要な設定は、最大ヒープ メモリの大きさを制御する設定であり、`-Xmx` オプションで指定します。既定では、これは、検索と参照サービスを実行しているサーバーノードで使用可能な物理メモリに基づいて、Tableau によって設定されます。使用可能な物理メモリとメモリ使用量に基づいて、できるだけ高い値を設定する必要がありますが、24 GB 以下に設定する必要があります。

最大ヒープ メモリの量を変更する場合は、システム上の他のサービスを考慮する必要があります。Tableau Server を実行しているコンピューターのメモリが不足している場合は、メモリを追加するか、コンピューター上のトポロジをスケールダウンすることを検討してください。他のプロセスが少ないノードや、大量の物理メモリを持つノードで検索と参照を実行している場合は、Search サービスで使用される量を増やすことを検討してください。既定のプロセス数を超えるノードで検索と参照を実行している場合は、サービスが使用可能なメモリを超えるメモリを使用しないように、ヒープ メモリの最大割り当てを減らすことを検討する必要があります。

既定の最大ヒープメモリ割り当て

次の表は、使用可能なメモリに基づいて、サーバー ノードの検索と参照に Tableau が割り当てるヒープ メモリの*既定*の量を示しています。

システム メモリ	**SOLR ヒープ メモリ**
<= 16 GB	1 GB
<= 32 GB	2 GB
<= 64 GB	4 GB
<= 128 GB	8 GB
> 128 GB	16 GB

最大ヒープメモリの値を変更するには、`tsm configuration set` コマンドを使用して `searchserver.javamemopts` の `-Xmx` 値を設定します。

まず、現在の値を取得します。

```
C:\WINDOWS\system32>tsm configuration get -k
searchserver.javamemopts
-Xmx512m -Xms512m -XX:+ExitOnOutOfMemoryError -XX:-UsePerfData
```

次に、`-Xmx` オプションの値を変更します。次のオプションを含めますが、その他のオプションは変更しないでください。

```
tsm configuration set -k searchserver.javamemopts -v "-Xmx8g -
Xms512m -XX:+ExitOnOutOfMemoryError -XX:-UsePerfData"
```

searchserver.javamemopts コンフィギュレーション キーの詳細については、searchserver.javamemoptsを参照してください。

クライアントセッションのタイムアウト

検索およびブラウズクライアントが検索と参照サーバーと調整サービス(Zookeeper)への接続を確立するまでの待機時間を設定できます。両方のタイムアウト値は比較的高い値に設定されていますが、サーバーコンテンツの参照に問題が発生し、vizportalとバックグラウンドログにSOLRに接続するタイムアウトが表示される場合は、これらの設定を増やしてみてください。この問題が発生しても影響がない場合は、Tableau Server コンピューター上で制限されたリソースを使用している可能性があります。

```
searchserver.connection_timeout_milliseconds
```

```
searchserver.zookeeper_session_timeout_milliseconds
```

値を変更するには、tsm configuration set コマンドを使用します。

```
tsm configuration set -k searchserver.connection_timeout_
milliseconds -v
```

searchserver.connection_timeout_milliseconds 構成キーの詳細については、searchserver.connection_timeout_millisecondsを参照してください。

起動時の ZooKeeper 接続のヘルスチェックのタイムアウト

Tableau Server の起動時には、特に CPU 使用率に関連するリソースの使用率が高くなります。検索と参照がコーディネーション サービス(動物園の飼育員)に接続しない場合、起動に失敗します。これを考慮して、Tableau Server は、検索と参照を開始する前に調整サービスのヘルスチェックを実行します。

Tableau Server コンピューターが特にビジー状態の場合、または検索と参照が開始できない場合は、このタイムアウト値を増やします。

この正常性チェックが失敗すると、control-searchserver.log ファイルにエラーが書き込まれます。

```
Failed zookeeper health check. Refusing to start SOLR.
```

Tableau Server がヘルスチェックが成功するまで待機する時間を増やすには、`tsm configuration set` コマンドを使用して `searchserver.startup.zookeeper_healthcheck_timeout_ms` の値を設定します。

```
tsm configuration set -k searchserver.startup.zookeeper_healthcheck_timeout_ms -v <nnnnnn>
```

`searchserver.startup.zookeeper_healthcheck_timeout_ms` コンフィギュレーションキーの詳細については 、searchserver.startup.zookeeper_healthcheck_timeout_msを参照してください。

Tableau 統計サービス

Tableau 統計サービスは、Tableau Server 上の「データの説明を見る」と予測モデリング関数の背後にある統計エンジンを管理します。このサービスは、Tableau Server 2022.1 から2023.1.x で使用できます。

> **注:** 統計サービスはバージョン 2023.3.0 で廃止され、[ステータス] ページや tsm status コマンドの出力には表示されなくなりました。

サーバー構成

Tableau 統計サービスは、VizQL がインストールされているすべてのノードに自動的にインストールされます。

プロセス	Tableau Statistical Service
ステータス	Tableau Statistical Service プロセスのステータスは、[ステータス] ページに表示されます。TSM CLI を実行すると表示できます。詳細については、「サーバープロセス ステータスの表示」を参照してください。
ロギン	Tableau Statistical Service プロセスが生成したログは

グ	/var/opt/tableau/tableau_server/data/tabsvc/logs/statsservice にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

Tableau 統計サービスに障害が発生した場合の影響

Tableau 統計サービスが機能しなくなると、「データの説明を見る」は使用できなくなり、予測関数 MODEL_PERCENTILE および MODEL_QUANTILE を使用した計算を含むビューはレンダリングされません。

パフォーマンス

Tableau Server で「データの説明を見る」または予測モデリング関数の使用が増える場合は、VizQL を実行しているノードに Tableau 統計サービスの追加のインスタンスをインストールすることで、利点が得られる場合があります。Tableau Server ノードのトポロジの構成の詳細については、「ノードの構成」を参照してください。

ログ ファイル

統計サービスにより以下の 2 つのセットのログ ファイルが作成されます。

- control_statsservice*.log: これらのログには、開始したり有効にしたりするサービスに関する情報が含まれます。
- stdout_statsservice_*.log: これらのログには、ステータスとエラーに関する情報が含まれます。

詳細については、「Tableau Server ログとログ ファイルの場所」を参照してください。

Tableau Server TSM メンテナンス サービス

クラスタのすべてのノードにインストールされている、3 つの TSM メンテナンス サービスがあります。データベースのメンテナンス、バックアップ/復元、およびサイトのインポート/エクスポートです。

プロセス	Database Maintenance、Backup/Restore、およびSite Import/Export。

ステータス	TSM メンテナンス サービスのステータスは [ステータス] ページに表示されません。ステータスを表示するには、TSM CLI を使用します。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Service Manager プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/` の `databasemaintenance`、`backuprestore`、`siteimportexport` ディレクトリにあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

これらのサービスは、それらを必要とする特定のメンテナンス タスクが管理者によって開始されない限り停止したままとなります。これらのサービスでは、高可用性を持たせるための他の構成は必要ありません。これらのサービスは、バックアップや復元などのメンテナンス タスクにのみ使用されるため、エンド ユーザーでの Tableau Server の機能には影響ありません。

Tableau Server VizData サービス

VizData サービス (vizdataservice) は、Tableau Server にパブリッシュされたデータ ソースへの接続を処理する新しいサービスです。Tableau Server にパブリッシュされたデータ ソースに対するクエリはすべて、VizData サービスによって処理されます。Prep のフローと Data Catalog のクエリは、Data Server によって処理されます。

VizData サービスの高可用性を確保するには、Tableau Server クラスター内の複数のノードで 1 つ以上の VizData サービス プロセスを構成します。

注: 2024.2.3 以降、Tableau Server のインストールには、VizData Service のインスタンスが少なくとも 1 つ必要です。

プロセス	VizData Service
ステータス	VizData Service のステータスは [ステータス] ページに表示されます。詳細については、「サーバープロセス ステータスの表示」を参照してください。
ロギング	VizData Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizdataservice` にあります。詳細については、

	「Tableau Server ログとログ ファイルの場所」を参照してください。

VizData サービスが失敗するとどうなりますか? インスタンスによって処理されているパブリッシュされたデータ ソースへの要求は失敗します (ワークブック ビューをレンダリングできません) が、後続の要求は実行中の他の VizData サービス インスタンスにルーティングされます。失敗した VizData サービスを含むノードがまだ実行中の場合、失敗したプロセスは数秒以内に Tableau Server によって自動的に再起動されます。

`control_vizdataservice_node*.log` には、サービスの開始と有効化に関する情報が含まれています。

`vizdataservice_node*.log` には、発生したエラーまたは問題が含まれています。

VizData サービスは、Tableau Server でパブリッシュされたデータ ソースに対してクライアントがクエリを実行したり、接続したりするときに、データ サーバーから VizData サービスへのクライアント トラフィックの一部を置き換えます。VizData サービス プロセスが失敗すると、VizData サービスを通じて実行されるクエリも失敗します。ワークブックでは、パブリッシュされたデータ ソースに対してクエリを実行したり、接続したりすることができなくなります。

推奨事項

データ サーバーと同じインスタンスを持つノードで VizData Service のインスタンスを構成し、1 つのノードで両方のサービスを有効にして、さまざまなクライアントの Tableau Server でデータ ソースをパブリッシュするための接続を提供および管理することをお勧めします。

データ サービスを展開する各ノードで VizData サービスのインスタンスを構成することをお勧めします。また、このサービスは、Tableau Server にパブリッシュされたデータ ソースへの接続も提供および管理します。VizData サービスのインスタンスをさらに他のノードに追加すると、プロセスが複製され、VizData サービスでワークブックのダウンタイムが発生する可能性が低くなります。

最終的には、VizData サービスがデータ サーバーに代わって、Tableau Server にパブリッシュされたデータ ソースへの接続を行います。当面の間、VizData サービスは、後方互換性のためにデータ サーバー サービスと共存します。

Tableau Server VizData ネイティブ サービス

VizData ネイティブ サービスは、Tableau Server 上で VizData サービスと通信し、パブリッシュされたデータ ソース接続とセッションを管理するコア サービスです。このサービスは Tableau Server によって内部的に管理され、管理者が構成することはできません。VizData ネイティブ サービスのインスタンスは、VizData サービスのインスタンスを持つすべてのノード上で 1 対 1 のマッピングで自動的に構成されます。

プロセス	VizData Native Service
ステータス	VizData Native Service のステータスは [ステータス] ページに表示されます。詳細については、「サーバー プロセス ステータスの表示」を参照してください。
ロギング	VizData Native Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizdatanativeservice` にあります。詳細については、「Tableau Server ログとログ ファイルの場所」を参照してください。

VizData ネイティブ サービスに障害が発生した場合、プロセスは、Tableau Server によって数秒以内に自動的に再起動されます。エラーや問題は、`nativeapi_vizdatanativeservice_*.txt` に記録されます。

`control_vizdatanativeservice_node*.log` には、サービスの開始と有効化に関する情報が含まれています。

VizData ネイティブ サービスのインスタンスは、VizData サービスのインスタンスの数に合わせて各ノードで自動的にスケールアップまたはスケールダウンされ、Tableau によって内部的に管理されます。管理者がサービスに関連して設定できることは何もありません。

パブリッシュされたデータ ソース接続を正常に保つには、VizData サービスと VizData ネイティブ サービスの両方が正常な状態で実行されて、要求に応答する必要があります。既定では、更新またはクリーン インストール後にデータ サーバーがインストールされているノードに両方のサービスが追加されます。

Tableau Server VizQL Data Service

VizQL Data Service (VDS) は、ビジュアライゼーションではなくプログラムによるクエリを通じて Tableau でパブリッシュされたデータソースにアクセスできるようにする API です。

VizQL Data Service の可用性を高くするには、1 つまたは複数の VizQL Data Service プロセスをクラスターの複数のノードで実行するように設定します。

プロセス	VizQL Data Service
ステータス	VizQL Data Service プロセスのステータスは、サービスが無効になっていても、[ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	VizQL Data Service プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizqldataservice` にあります。詳細については、「Tableau Server ログとログ ファイルの場所」を参照してください。

VizQL Data Service のプロセスが失敗するとどうなりますか? VizQL Data Service のプロセスを介して実行されるクエリは失敗し、その結果、すべてのプログラム クエリは、顧客アプリケーション、自動化ワークフローなどでパブリッシュされたデータソースへ直接アクセスすることができなくなります。Tableau Server の UI とviz 作成の機能は、VizQL データサービスに依存しません。

1 つのノード上で実行される既定の VizQL Data Service のプロセスは 1 つです。高可用性を実現するには、VizQL Data Service とVizData サービスのインスタンスを冗長化して設定する必要があります。VizQL Data Service のプロセスを複数実行する場合、1 つのプロセスでエラーが発生すると、すべてのクエリでエラーが発生し、エラー発生時点でのクエリ結果が失われます。その後の要求は、Tableau Server クラスター上で機能している他の VizQL Data Service のプロセスにルーティングされます。

Tableau Server の VizQL Server

VizQL Server は、ビューを読み込んでレンダリングし、クエリを計算して実行します。VizQL Server プロセスで高可用性を実現するには、1 つまたは複数のインスタンスを複数のノードで実行するように構成します。

プロセス	VizQL Server
ステータス	VizQL Server プロセスのステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	VizQL Server プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/vizqlserver` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

VizQL Server のプロセスが失敗するとどうなりますか? 1 つしかない VizQL Server プロセスでエラーが発生した場合、Tableau Server はビューをレンダリングできなくなります。高可用性を実現するには、冗長性のある VizQL プロセスを構成する必要があります。非常に一般的な構成では、各ノードに 2 ~ 4 つの VizQL Server プロセスがあります。

これにより、高可用性とスケーラビリティの必要を同時に満たすことができます。複数の VizQL Server プロセスが実行されている場合、単一プロセスでエラーが発生すると、いずれかの要求でエラーが発生し、エラー発生時のセッション データが失われます。その後の要求は、Tableau Server クラスタ上で機能している他の VizQL Server プロセスにルーティングされます。

Tableau Prep Conductor

Tableau Prep Conductor プロセスは、フローを実行し、Data Catalog による取り込みができるようにフローを処理します。このプロセスでは Tableau Server のスケジュール調整機能および追跡機能を活用しているため、フロー実行を自動化してフロー出力を更新することができます。2020.4 以降、データ管理 を使用する必要があるのは、Tableau Server で実行するフローのスケジュールを設定する場合のみです。詳細については、Tableau Prep Conductorを参照してください。

プロセス	Tableau Prep Conductor
ステータス	Tableau Prep Conductor のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Tableau Prep Conductor プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/flowprocessor` にあります。詳細については、

	Tableau Server ログとログ ファイルの場所を参照してください。

Tableau Prep Conductor は以下の構成要素を使用してフローを実行します。

- **バックグラウンダー:** Tableau Prep Conductor はフローの実行にバックグラウンダープロセスを使用します。バックグラウンダーはシングル スレッドのため、ノード上のバックグラウンダープロセスの各インスタンスは一度に 1 つのフローを実行可能です。ノードにさらにバックグラウンダーを追加すると、そのノードで並列で実行可能なフローの数を増やすことができます。バックグラウンダープロセスの最大数は、そのノードの物理コア数の半数です。

- **コネクタ:** Prep Conductor は、データへの接続に、サポートされている Tableau データ コネクタを使用します。サポートされているコネクタのリストについては、「サポートされるコネクタ」を参照してください。

- **データ エンジン:** 参照元データ ソースにプッシュできないフロー内のデータや変換ステップに対する変更は、データエンジンプロセスを使用して処理されます。たとえば、SQL Server は正規表現をネイティブにサポートしていません。SQL Server に接続する際、Tableau Prep を使用すると正規表現計算を記述できます。Tableau Prep Conductor はデータエンジンを使用してデータを一時的に読み込み、その後、正規表現を実行します。

パフォーマンスと規模に関する推奨事項

- **別のノードへのフローの分離:** 別のノードで Tableau Prep Conductor を実行すると、他の Tableau ワークロードからフローのワークフローが分離されます。Prep のフローは CPU と RAM を大量に使用するため、これが強く推奨されます。

- **フローのスケジュールの管理:** フローのスケジュールを作成することにより、フローの実行を制御できます。これらのスケジュールにより、フローの実行タイミング、実行頻度、そのスケジュールの優先度、そのスケジュールでアイテムを順番に実行するか、並列で実行するかが決まります。

- **リソースの追加:** Tableau Prep Conductor の環境を拡張する場合、ノードあたり最大 8 つの物理コア ボックスにまで拡張し、各ノードで 4 つのバックグラウンダーを実行することをお勧めします。リソースがさらに必要な場合は、サーバー環境にさらにノードを追加することをお勧めします。

管理ビューを使用し、ユーザー アクティビティとフローのパフォーマンスを監視できます。詳細については、フローの正常性とパフォーマンスの監視を参照してください。

トポロジと構成

既定では、Tableau Prep Conductor はバックグラウンダーが有効になっているノードで自動的に有効になります。ノード ロールがフローを除外するように設定されている場合、そのノードには Tableau Prep Conductor はインストールされません。詳細については、ノード ロールによるワークロード管理を参照してください。

Tableau Prep Conductor はフロー実行の専用ノードで有効にすることをお勧めします。詳細については、次のトピックを参照してください。

- Tableau Server の新規インストールの場合は、「手順 1 (新規インストール): Tableau Server を Tableau Prep Conductor と一緒にインストール」を参照してください。
- Tableau Prep Conductor を既存の Tableau Server インストールで有効にする場合は、「手順 1 (既存インストール): Tableau Prep Conductor の有効化」を参照してください。

Tableau Prep フロー作成

Tableau Prep フロー作成プロセスは、ブラウザーでのインタラクティブな Prep フロー エクスペリエンスを提供します。ユーザーはこのプロセスを使用して、Tableau Server でフローを作成および操作して、データのクリーニングと準備を行うことができます。詳細については、Web 上でのフローの作成と操作を参照してください。この機能は、バージョン 2020.4 で追加されました。

プロセス	Tableau Prep Flow Authoring
ステータス	Tableau Prep Flow Authoring のステータスは [ステータス] ページに表示されます。詳細については、サーバープロセス ステータスの表示を参照してください。
ロギング	Tableau Prep Flow Authoring プロセスが生成したログは `/var/opt/tableau/tableau_server/data/tabsvc/logs/floweditor` にあります。詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

Tableau Prep フロー作成は、既定で有効になっています。次のコンポーネントを使用します。

- **Tableau Prep Minerva Service**: データソースのクエリを実行するために、Tableau Prep フロー作成で使用されます。既定では、Tableau Prep フロー作成が有効になっているノードで自動的に有効になります。
- **データ エンジン**: データエンジンで、データ抽出を作成し、クエリを処理します。既定では、Tableau Prep フロー作成が有効になっているノードで自動的に有効になります。
- **ゲートウェイ**: ゲートウェイ プロセスは、ブラウザー、Tableau Desktop、その他のクライアントからTableau Server へのすべての要求を処理します。既定では、Tableau Prep フロー作成が有効になっているノードで自動的に有効になります。

パフォーマンスと規模に関する推奨事項

Tableau Prep フロー作成は CPU とRAM を大量に使用する可能性があります。サーバーを監視し、フロー作成を別個のノードに分離するか、サーバー リソースを追加して、必要に応じて展開を調整することをお勧めします。

別個のノードへのフロー作成の分離

Tableau Prep フロー作成を別個のノードで実行すると、フロー作成ワークフローが他の Tableau ワークロードから分離されます。多数の抽出更新とViz の編集および表示が行われるビジー状態のサーバーがあり、このワークロードに中断を発生させないようにする場合は、Prep フロー作成を専用のサーバー ノードに分離することをお勧めします。つまり、すべてのフロー編集は専用 ノードに転送されます。詳細については、「ノードの構成」を参照してください。

注: Prep の Web 作成では、アプリケーション サーバー (VizPortal)、PostgreSQL リポジトリ、キャッシュ サーバー、Hyper などの共有サービスを使用します。これらのサービスがすでに容量に達している場合は、Prep の Web 作成でさらに負荷が発生するため、追加リソースも必要になる場合があります。Prep Web Authoring が機能するには、設定されている最も番号の小さいノードでキャッシュ サーバーが実行されている必要があります。たとえば、キャッシュ サーバーがノード2、3、4 に設定されていても、ノード2 上ではダウンしている場合、Prep Web Authoring は機能しません。詳細については、ナレッジベースの「キャッシュ サーバーが最小番号のノードで実行されていない」を参照してください。

Tableau Server をマルチノード クラスタで実行している場合は、1 つまたは複数のノードをバックグラウンダー専用にすることができます。バックグラウンダープロセスを使用すると、Tableau Prep Conductor などのバックグラウンド ワークロードを、Prep のフロー作成や VizQL Server などのすべてのインタラクティブ ワークロードから分離できます。詳細については、「Tableau Server バックグラウンダープロセス」と「ノード ロールによるワークロード管理」を参照してください。

コアベースおよびユーザベースの展開では、最適なパフォーマンスを得るために、少なくとも 1 つのノードをフロー専用にすることを強くお勧めしますフローはライセンスが付与された任意の Server コアで実行できますが、追加のリソース コアを購入する場合は、Tableau Prep Conductor のみを実行し、更新または VizQL プロセスの抽出は行わないでください。

リソースの追加

リソースがさらに必要な場合は、サーバー環境にさらにノードを追加することができます。ノードを計画する際に、いくつかの要因が、割り当てる必要があるハードウェアの追加に関する決定に影響を与える場合があります。

ノードを計画する際に考慮すべき主な事項は次のとおりです。

- ピーク時に予想される同時ユーザー数または同時セッション数。ユーザーごとの同時セッション数は、TSM オプション `maestro.sessionmanagement.maxConcurrentSessionPerUser` を使用して設定できます。詳細については、tsm configuration set のオプションを参照してください。
- フローに含まれるデータ入力ノードの平均数とそのデータ量。最大行サンプリング制限は、TSM オプション `maestro.app_settings.sampling_max_row_limit` を使用して設定できます。詳細については、tsm configuration set のオプションを参照してください。
- 作成されるフローの複雑さとノードの数。結合、ユニオン、集計、ピボットなどのノード タイプでは、通常、より多くのリソースが必要になります。

追加ライセンス

ノードの構成には、データ管理 とAdvanced Management の追加ライセンスが必要です。

オファリング	次の操作を実行できます。
データ管理	フローのみを実行するようにノードを構成するか、フロー以外

オファリング	次の操作を実行できます。
	のすべてのジョブを実行するようにノードを構成します。Tableau Prep Conductor がノードで実行されている必要があります。 データ管理 ライセンスには、フローのスケジュール設定と追跡が可能な Tableau Prep Conductor が含まれています。このライセンスは、ロールベースまたはコアベースの単一の Tableau Server 展開に使用できます。 Creator は、サーバー上で直接フローを作成および編集するために データ管理 を使用する必要はありません。
Advanced Management	ノードルールを使用して、さまざまなタイプのワークロードを処理する場所を構成します。たとえば、あるノードでフローを実行し、別のノードでサブスクリプションとアラートを実行できます。

トポロジと構成

Tableau Prep フロー作成の可用性を高めるには、Tableau Prep フロー作成を有効にするときに、その 2 つのインスタンスをノードで構成します。

2 ノード構成の例を次に示します。

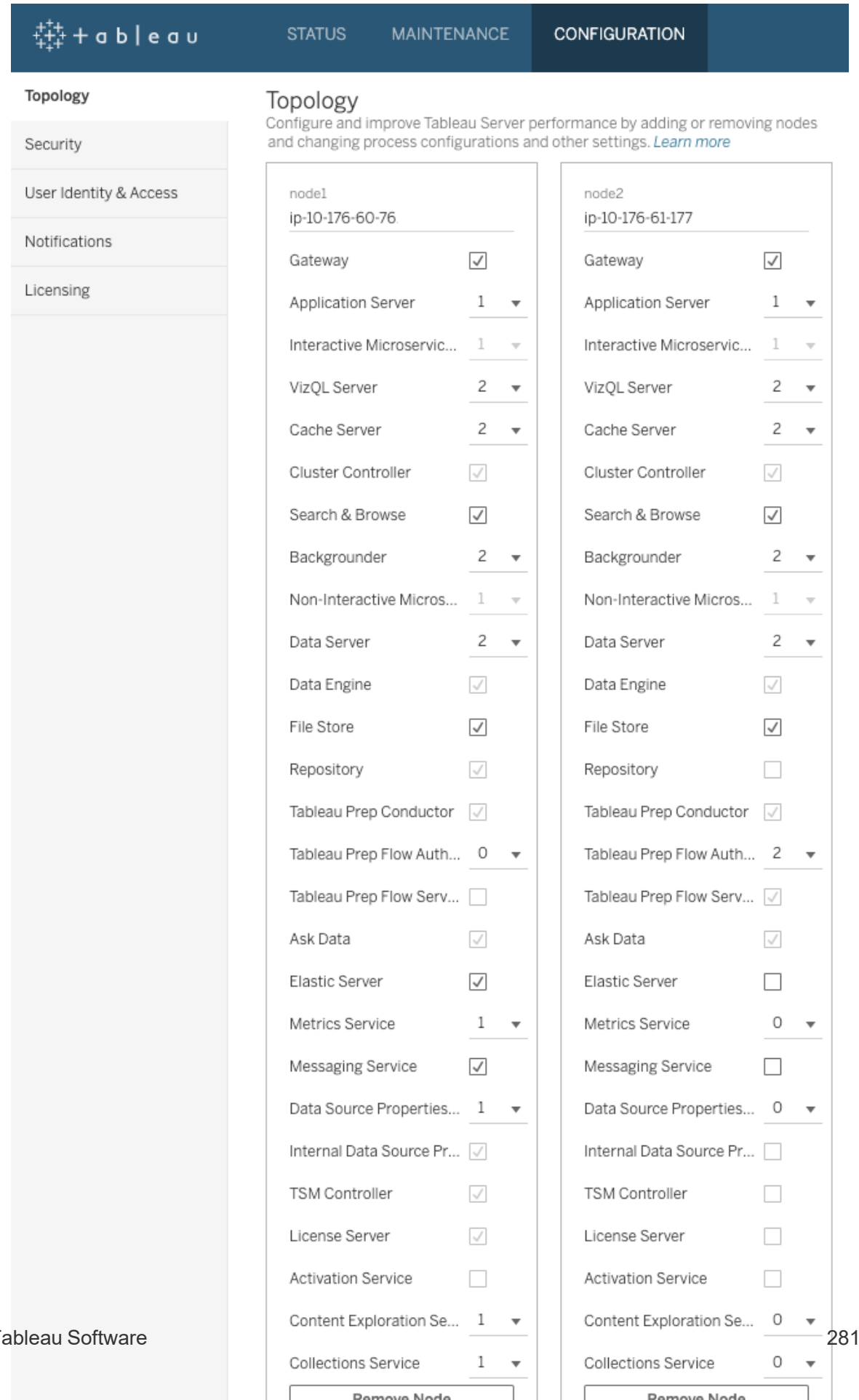
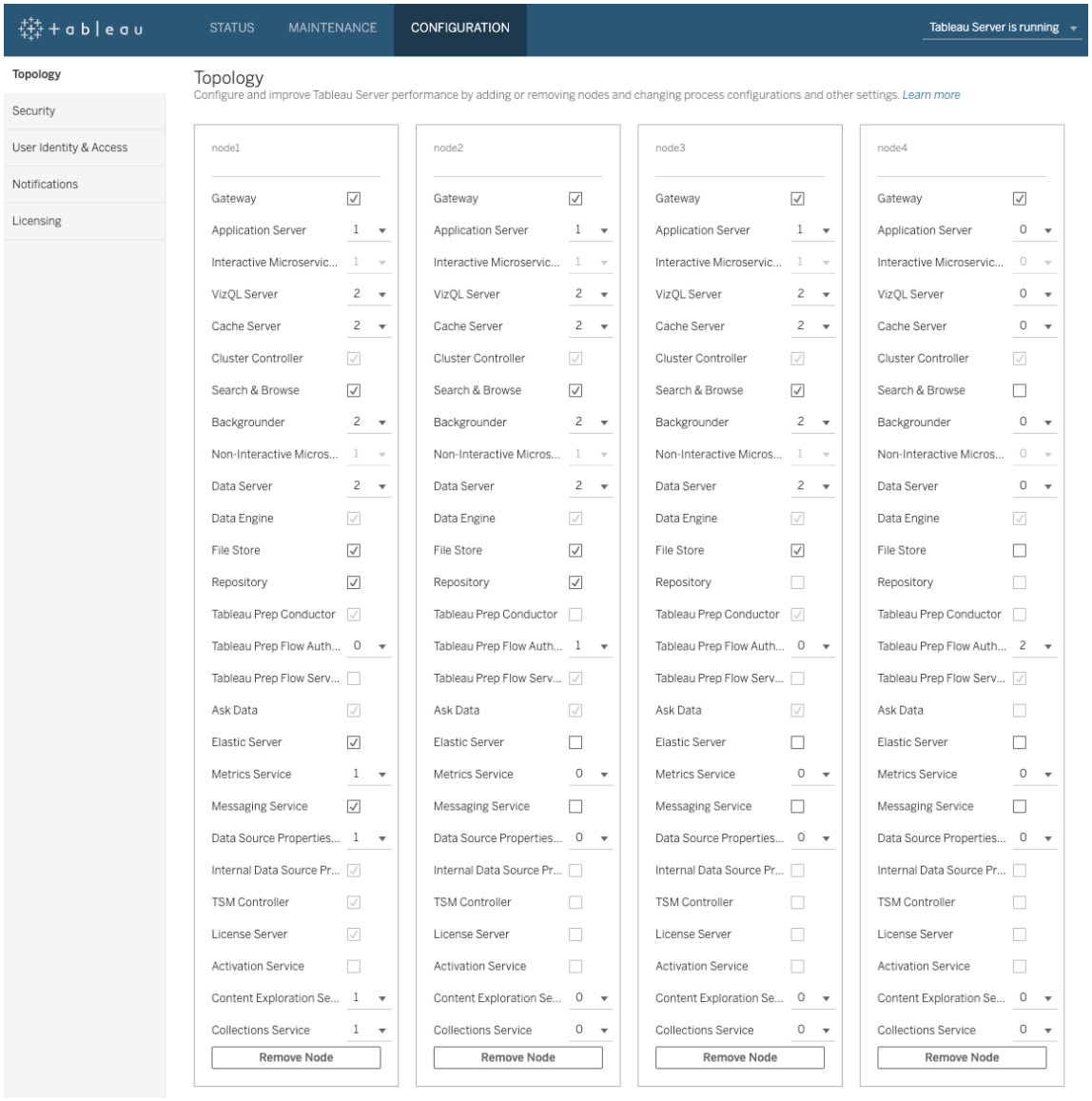


tableau
STATUS
MAINTENANCE
CONFIGURATION
Topology
Security
User Identity & Access
Notifications
Licensing
Topology
Configure and improve Tableau Server performance by adding or removing nodes and changing process configurations and other settings. Learn more
node1
ip-10-176-60-76
Gateway
Application Server 1
Interactive Microservic... 1
VizQL Server 2
Cache Server 2
Cluster Controller
Search & Browse
Backgrounder 2
Non-Interactive Micros... 1
Data Server 2
Data Engine
File Store
Repository
Tableau Prep Conductor
Tableau Prep Flow Auth... 0
Tableau Prep Flow Serv...
Ask Data
Elastic Server
Metrics Service 1
Messaging Service
Data Source Properties... 1
Internal Data Source Pr...
TSM Controller
License Server
Activation Service
Content Exploration Se... 1
Collections Service 1
node2
ip-10-176-61-177
Gateway
Application Server 1
Interactive Microservic... 1
VizQL Server 2
Cache Server 2
Cluster Controller
Search & Browse
Backgrounder 2
Non-Interactive Micros... 1
Data Server 2
Data Engine
File Store
Repository
Tableau Prep Conductor
Tableau Prep Flow Auth... 2
Tableau Prep Flow Serv...
Ask Data
Elastic Server
Metrics Service 0
Messaging Service
Data Source Properties... 0
Internal Data Source Pr...
TSM Controller
License Server
Activation Service
Content Exploration Se... 0
Collections Service 0

2 ノード構成のステータス ページの例を次に示します。

Process	node1 ip-10-176-60-76	node2 ip-10-176-61-177
Gateway	✓	✓
Application Server	✓	✓
Interactive Microservice Container	✓	✓
VizQL Server	✓ ✓	✓ ✓
Cache Server	✓ ✓	✓ ✓
Cluster Controller	✓	✓
Search & Browse	✓	✓
Backgrounder	✓ ✓	✓ ✓
Non-Interactive Microservice Container	✓	✓
Data Server	✓ ✓	✓ ✓
Data Engine	✓	✓
File Store	✓	✓
Repository	✓	
Tableau Prep Conductor	✓	✓
Tableau Prep Flow Authoring		✓ ✓
Tableau Prep Flow Service		✓
Ask Data	✓	✓
Elastic Server	✓	
Metrics Service	✓	
Messaging Service	✓	
Data Source Properties Service	✓	
Internal Data Source Properties Service	✓	
TSM Controller	✓	

4 ノード構成の例を次に示します。



4 ノード構成のステータスページの例を次に示します。

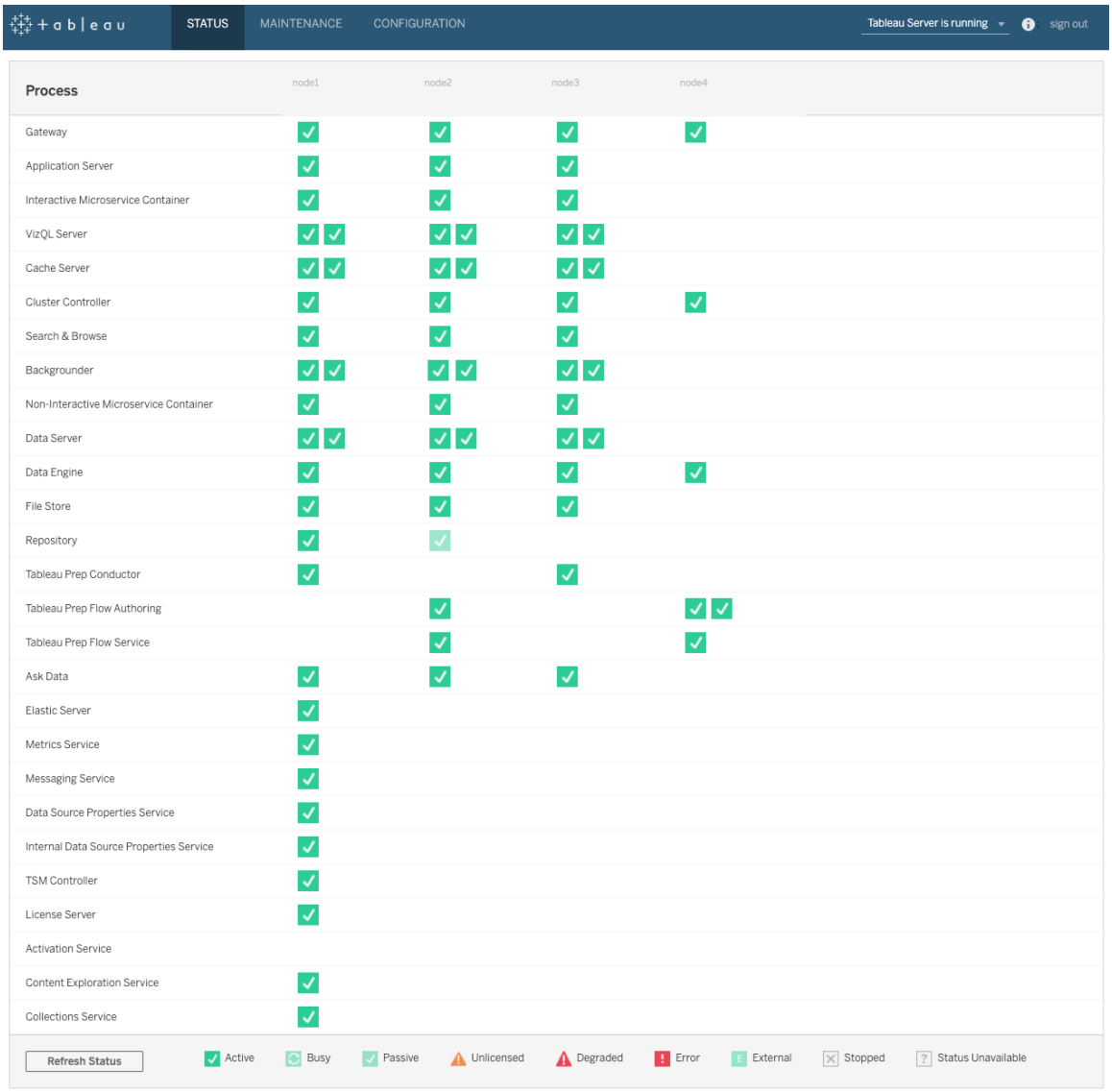


Tableau Server の動的トポロジの変更

TSM の導入に伴い、Tableau Server でサーバーを再起動せずに特定のトポロジの変更や更新を実行する機能も導入されました。これらは動的トポロジの変更として知られ、バックグラウンダーや VizQL Server のプロセスで可能です。

ノードで既にプロセスのインスタンスが少なくとも 1 つ以上実行されている場合は、Tableau Server を再起動しなくても、ノード上のバックグラウンダーまたは VizQL Server インスタンスの数を増減することができます。変更する必要があるのはバックグラウンダーまたは VizQL Server のインスタンス

数のみです。別のプロセスを追加/削除する場合、またはバックグラウンダーまたは VizQL Server の最初のインスタンスをノードに追加する場合、あるいはそのいずれかの最後のインスタンスをノードから削除する場合は、Tableau Server を再起動する必要があります

動的な構成変更

Tableau Server のバージョン 2020.2.0 以降では、特定の構成変更を構成キーを使用して動的に行うこともできます。動的構成の変更を加えるのと同時に、動的トポロジの変更を行うことができ、Tableau Server を再起動する必要はありません。動的構成の変更の詳細については、Tableau Server の 2020.2 の新機能と変更点でTableau Server のリリース ノートを参照してください。

シナリオの例

これが役立つ理由について理解を深めるため、次の例を考えてください。

- バックグラウンダー - 営業四半期末に、営業チームが Tableau Server を使用して数値を追跡しています。抽出に左右されるダッシュボードには、営業担当者の実績が表示されています。抽出の更新に遅延があると、最新の数値が表示されません。バックグラウンダーまたは VizQL Server が既に 1 つ以上ある任意のノードに追加バックグラウンダーを追加し、抽出の更新のスループットを向上させることにより、四半期の終了時点で確実に最新の数値を得ることができます。その後、四半期ごとのプッシュの後に、バックグラウンダーのインスタンスを再度減らし、Tableau Server を元の構成に戻すことができます。
- VizQL Server - 同様に Tableau Server がビューの更新に遅れないようにすることができない場合、VizQL Server またはバックグラウンダーが構成されているインスタンスが既に 1 つ以上ある任意のノードに、追加 VizQL Server インスタンスをすぐに追加することができます。上記のバックグラウンダーの例で、ユーザーが朝になって戻ってくる前に、VizQL Server を一時的に削除して追加バックグラウンダーに対応してから、それを再追加したいと考えるかもしれません。

動的トポロジの変更を行う

動的トポロジの変更は、TSM Web UI またはコマンドラインを使用して行うことができます。Web UI を使用するにはブラウザーで TSM にサインインし、[構成] タブで、更新するノードのバックグラウンダーまたは VizQL Server のインスタンス数を更新します。詳細については、ノードの構成を参照してください。TSM CLI を使用して変更を行うには、コマンドプロンプトで tsm topology set-

process コマンドを実行します。詳細については、ノードのプロセス数を変更するを参照してください。

動的トポロジの変更による影響

VizQL Server またはバックグラウンダーの既存インスタンスを削除する動的トポロジの変更を行うと、インスタンスはすぐに削除されます。ユーザーおよび現在実行中のジョブに対し、以下の影響が及ぶ可能性があります。

- バックグラウンダー — 現在実行中のすべてのジョブが終了します。Tableau Server の通常の再試行論理により、別のバックグラウンダー インスタンスを使用し、これらのジョブが再開始します。
- VizQL Server - 現在アクティブなすべてのセッションが終了します。ユーザーにエラー メッセージが表示される場合があります。ブラウザーを更新すると、エラーがクリアされます。

ベストプラクティス

本番環境での実装前に、使用する計画がある動的トポロジの変更をテストすることをお勧めします。これにより、ユーザー、スケジュールされた更新、サブスクリプションに及ぶ可能性のある影響を十分に理解し、動的トポロジによって得られる柔軟性を最も効率的に活用することができます。

動的トポロジの変更の自動化

動的トポロジの変更は自動化することができます。たとえば、抽出の更新の大半が夜間にスケジュールされており、ユーザーがサインインしていないためにサーバーに余分の容量があることがわかっている場合、使用効率が最も高くなるときはスクリプトまたは他の自動展開ツールを使用してバックグラウンダー インスタンスを追加し、その後、それを営業日の始業前に削除することができます。

プロセスのステータスを自動的に取得するには、`tsm status -v` コマンドを使用してスクリプト内の出力を解析します。または、TSM REST API を使用してサーバー ステータスを取得することもできます。API は現在アルファ版です。詳細については、「サーバー ステータスの取得」を参照してください。

node2 でバックグラウンダーの 4 つのインスタンスを設定するサンプル スクリプトは、次のようになります。

```
echo Adding/Removing Processes
tsm topology set-process -pr backgrounder -n node2 -c 4
tsm pending-changes apply
echo Done!
```

追加情報

Tableau Server プロセス

サーバープロセスの制限

Tableau Server のプロセスを再構成する場合は、プロセス インスタンスの数を増やすことができる量に制限があります。既定では、制限は 8 に設定されています。お使いのマシンに十分な RAM および CPU コアがあり、この制限を超えて使用する場合、service.max_procs 構成オプションを使用して制限を変更できます。Tableau では、プロセス インスタンスごとに、プロセスを実行するマシンに 1 GB の RAM および 1 論理 CPU コア以上を確保することをお勧めします。

使用可能な最大プロセス数を変更するには、次の手順を行います。

1. 次のコマンドを入力します。ここで、number は許可するプロセス インスタンスの最大数です。

   ```
   tsm configuration set -k service.max_procs -v <number>
   ```

   例は次のとおりです。

   ```
   tsm configuration set -k service.max_procs -v 10
   ```

2. 次のように入力します。

   ```
   tsm pending-changes apply
   ```

   保留中の変更にサーバーの再起動が必要な場合は、pending-changes apply コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場

合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

Tableau サービス マネージャーのポート

Tableau サービス マネージャー (TSM) および Tableau Server on Linux のコンポーネントを構成するプロセスとサービスは通信にさまざまなポートを使用します。既定では、これらのポートのほとんどは事前定義されたポート範囲から動的に割り当てられます (マッピング)。ポート割り当ては、各サービスまたはプロセスがインストールされた時点で行われます。どのポートが使用されるかはさまざまな方法で制御できます。

プロセスの小規模なサブセットは動的ポート マッピングを使用せず、独自に動作します。詳細については、後述の動的マッピングされないポートを参照してください。

注: この操作には、TSM コマンド ラインを使用して実行する必要のある手順が含まれています。

エフェメラル ポートの使用

オペレーティング システムがエフェメラル ポートを使用するように設定されている場合、以下に説明するように設定されている場合でも、Tableau トラフィックがこれらのポートを経由してルーティングされる可能性があります。これは OS レベルで発生するものであり、Tableau で設定できるものではありません。ポートの使用を制御する場合は、エフェメラル ポートの使用を制限する方法についてオペレーティング システムのドキュメントを参照してください。

ファイアウォールの要件

Tableau Server をインストールするコンピューターでファイアウォールを実行している場合は、Tableau Server トラフィック用に次の既定のポートを開く必要があります。443 を除くすべてのポート番号を変更できます。

ポート	TCP/UDP	使用...	インストールのタイプ	
			すべて	分散/高可用性
80	TCP	ゲートウェイ	**X**	
443	TCP	SSL。Tableau Server で SSL を構成している場合、アプリケーション サーバーは このポートに要求をリダイレクトします。このポートは変更しないでください。	**X**	
8850	TCP	Tableau Services Manager。	**X**	
8060	TCP	PostgreSQL データベース。	**X**	
8061	TCP	PostgreSQL バックアップ検証 ポート	**X**	
8000-9000	TCP	Tableau プロセスの動的 マッピング用 に既定で予約されているポートの範囲		**X**
27000-27009	TCP	Tableau Server がライセンス サービスで使用するポートの範囲。この範囲は、ライセンス サービスを実行しているノードで開いている必要があり、また他のノードからアクセス可能である必要があります。既定では、最初のノードは ライセンス サービスを実行します。	**X**	

ローカル ファイアウォールの構成を参照してください。

ポートの割り当て

TSM でのポート割り当てまたは マッピングに使用できる 2 通りのアプローチがあります。

- **動的 ポート割り当て**。これは既定で、管理者で必要な介入は最小限です。
- **手動 ポート割り当て**。このオプションでは、管理者が個別に各 ポートを割り当てる必要があります。

ポート割り当てを変更し、ローカル ファイアウォールを実行している場合、ローカル ファイアウォール構成を参照してください。

動的ポート割り当て

動的割り当てプロセスに使用可能なポート範囲を変更することで、どのポートが使用されるかを制御できます。これを実行すると動的ポート割り当てはそのまま保持されますが、選択できるポートは制限されます。動的ポート割り当てを保持したままでも、引き続き、特定のプロセスに手動でポートを割り当てることができます。このアプローチによって、特定のポートを特定のプロセスに割り当て、その他は動的にマッピングされるようにしておくことができます。一部の個別に割り当てられるポートの有無に関係なく、動的マッピングの使用が最も簡単なアプローチで、大半のお客様の要件が満たされます。動的に割り当てられたポートは、Tableau Server 構成をエクスポートした場合も、保持されます。

既定では、8000 ~ 9000 の使用可能なポートの中から各サービスまたはプロセスに割り当てられます。この割り当ては、ノード上でのサービスの初回インストール時に実行されます。Tableau Server が初期化された後、次のコマンドを実行することで、どのサービスまたはプロセスがどのポートを使用しているか確認できます。

```
tsm topology list-ports
```

マルチノード クラスタがある場合は、すべてのノードのポートがリストされます。

ポート範囲の変更

使用されるポートについて特定の要件がある組織の場合、これを制御する最も簡単な方法は、ポートが動的に選択されるポート範囲を変更することです。構成ファイルで範囲として最小ポートと最大ポートを指定し、インストール時に実行する必要があります。

注: ポート範囲の最小許容サイズは、サーバーのインストールと実行されているサービスまたはプロセスの数によって異なります。一般的なベストプラクティスとして、ポート割り当ては範囲内でランダムにポートを選択して実行されるため、過度に狭い範囲に制限しないようにしてくださ

い。十分な大きさの範囲を許可しないと、選択時に使用可能なポートを見つけられない場合があります。

使用可能なポートを選択する範囲を8300から8600までの間に制限するには、構成ファイルに次のようなエントリを含めます。

```
"configKeys": {
    "ports.range.min": "8300",
    "ports.range.max": "8600"
}
```

範囲内の特定のポートをブロックする

バージョン2021.1.0以降では、動的に割り当てられた範囲内にある特定のポートを指定することができますが、Tableauでは使用しないでください。これは、Tableauが使用している範囲内のポートに依存するTableau Serverコンピューター上に他のソフトウェアがある場合に役に立ちます。

TSMおよびTableau Serverで使用しないポートを指定するには、次の構成キー`ports.blocklist`を使用します。

```
tsm configuration set -k ports.blocklist -v <port>[,<port>,<port>]
```

例:

```
tsm configuration set -k ports.blocklist -v 8000,8088, 8090
```

動的ポート割り当ての無効化

ポート範囲の制限と個別のポート割り当てを組み合わせて得られる制御レベルより、さらに厳密にポート割り当てを制御する必要がある場合は、初期サーバー構成時に、動的ポートマッピングを無効にします。動的ポートマッピングを無効にすると各プロセスの各ポートを手動で割り当てる必要があるため、単一のポート割り当てをそれぞれ制御する必要がある場合を除き、これはお勧めしません。

動的ポートマッピングを無効にした場合は、インストールの各ノードで、各プロセスのポートを構成する必要があります。

動的マッピングを無効にするには、構成ファイルに次のようなエントリを含めます。

```
"configKeys": {
    "service.port_remapping.enabled": false
}
```

重要: 動的ポート割り当てを無効にした場合、ライセンスサービスのポート範囲は含まれません。この範囲 (27000 ~ 27009) は、ライセンスサービスを実行しているノードに開放し、他のノードからアクセスできる必要があります。既定では、最初のノードはライセンスサービスを実行します。

手動ポート割り当て

自動ポート割り当てを完全に無効化し、各プロセスのポートを個別に割り当てることができます。これを実行すると、すべてのノードで各プロセスのポートを割り当てる必要があります。プロセスの初回インストール時は構成ファイルで、またはインストール後は TSM コマンドを使用して、ポートを割り当てることができます。Tableau Server 構成をエクスポートした場合は、プロセスのインストール時に割り当てられたポートのみが保持されます。

動的マッピングが有効かどうかに関係なく、特定のプロセスに個別のポートを指定することができます。プロセスで特定のポートを使用する必要がある場合、または動的マッピングを無効にした場合に、これを実行する必要が生じることがあります。プロセスのポートを指定する方法は 2 つあります。インストール時とインストール後です。

インストール時のポートの構成

ここで説明するように、ポート割り当ての構成はインストールプロセスの際に行うことをお勧めします。インストール後にポート変更を行うのは、はるかに多くの労力が必要となります。

インストール時にポートを構成するには、お使いのポート構成を指定する json ファイルを作成します。このプロセスは、既定以外のポート範囲を定義することに似ていますが、代わりに、特定のサービスまたはプロセスに特定のポートを指定します。特定のポートを割り当てようとしている場合は、こ

のアプローチがそれを実行するための最も確実な方法です。`tsm settings export` コマンドを使用してサーバー構成やトポロジ設定をエクスポートしても、ポートマッピングが保持されるからです。

インストール時にポートを定義するには、構成ファイルに情報を追加してノード(`workerN`)、プロセス(`servicename`)、インスタンスID (`instanceid`)、ポートのタイプ(`porttype`)、使用するポートを指定します。形式は次のようになります。

```
workerN.{servicename}_{instanceid}.{porttype}.port:X
```

説明:

- `workerN` はオプションのパラメーターで、再マッピングが適用されるノードを特定します。ノード番号は 0 から始まります。
  異なるノード上の同じサービスに異なるポートをマッピングする必要がないかぎり、このパラメーターを含めないことをお勧めします。このパラメーターをオフにしておくと、初期ノードにサービス ポートをマッピングしたり、複数ノードに同じサービス ポートをマッピングしたりすることができます。
- `servicename` は、ポートを使用するプロセスまたはサービスの名前です。
- `instanceid` は、プロセスのインスタンスです。1 つのノード上で 1 つのプロセスの複数のインスタンスを設定しようとしている場合は、インスタンスごとにこの値を増分する必要があります。instanceid はゼロ (0) で始まり、プロセスのインスタンスごとに 1 ずつ増分します。対象のノード上に単一のインスタンスのみがインストールされているサービスの場合は、これをオフにしておく必要があります。
- `porttype` プライマリポートを設定している場合は、このオプションを含めないでください。
- `port` は、プロセスまたはサービスが使用するポートです。

たとえば、最初のノードでファイル ストア プロセスの 1 番目のインスタンスのポートを 8500 に設定するには、次のようなエントリを構成ファイル エントリに含めます。

```
"configKeys": {
    "filestore_0.port: "8500"
}
```

上記の例には、オプションの workerN パラメーターが含まれていないため、クラスタ内のすべてのノードにポートが設定されます。また、プライマリファイルストア ポートを設定しているため、porttype オプションはオフのままです。

> **重要**: 構成ファイルでポートの変更を指定する場合は、`tsm settings import` コマンドに `--force-keys` パラメーターを含める必要があります。

インストール後のポートの構成

Tableau Server のインストール後にポートの変更が必要な場合、`tsm topology set-ports` コマンドを使用します。このアプローチによって、プロセスがインストールされた後に、特定のプロセスのポートを指定できます。次の制限事項による制約があります。

- ノードごとにポートを個別に設定する必要がある。
- 個々のポートを設定後に `tsm restart` を実行する必要がある。
- `tsm settings import` を使用して Tableau Server 構成をインポートする必要がある場合、ポート割り当ては保持されない。
- ポート名が configKeys で必要な構文とは異なる構文を tsm コマンドで使用する。このトピックの最後にある表には、構文の参照が記載されています。

たとえば、最初のノードでファイル ストアの 2 番目のインスタンスがポート 8500 を使用するように設定するには、次のようにします。

```
tsm topology set-ports --node-name node1 --port-name filestore --
port-value 8500 --instance 2
```

以下の例は、JMX ポートを設定するために簡略化されたコマンドを使用する方法を示しています。

```
tsm topology set-ports -n node1 --port-name vizqlserver:jmx.rmi -pv
9403 -i 1
```

```
tsm topology set-ports -n node1 --port-name vizqlserver:jmx -pv 9404
-i 2
```

> **注**: ポートの入力時には、これの検証は行われません。このため、既に割り当て済みのポートを使用する場合や、コマンドの構文に入力ミスがある場合、再起動を行うまでは Tableau にエラーは表示されません。再起動後、*The reconfigure async job failed. (非同期ジョブの再構成に失敗しました。)* という一般的なエラーが表示される場合があります。

有効なポートで誤ったportname:typeを追加すると、誤った入力を削除できません。ポートを更新するには、その値に未使用のポートを再割り当てし、再度ポートを解放する必要があります。

動的マッピングされないポート

Tableau Server リポジトリは、動的マッピングされない2つのポートを使用します。これらにはそれぞれ、tsm configuration setコマンドとprocess-specific パラメーターを使用して上書きできる既定のポートがあります。

ポート名	ポート(既定)	説明
pgsql.port	8060	Tableau リポジトリ(PostgreSQL データベース) のポート。 このポートを上書きするには: `tsm configuration set -k pgsql.port -v <port>`
pgsql.verify_restore.port	8061	リポジトリバックアップの整合性を検証するためのポート。 このポートを上書きするには: `tsm configuration set -k pgsql.verify_restore.port -v <port>`

これらのポートは動的ポートマッピング システムを使用しないため、tsm topology list-portsコマンドの出力には表示されません。これらの値を確認するには、tsm configuration get -k <config.value>コマンドを使用する必要があります。例は次のとおりです。

```
tsm configuration get -k pgsql.port
```

initialize-tsm によるポートリマッピングの制御

ポート割り当ては、サービスのインストール時に行われます。つまり、TSM 固有のプロセスに手動でポートをマッピングするには、initialize-tsm スクリプトの実行時にポートを割り当てる必要があります。スクリプトには、個々の TSM サービスのポートを指定するオプション、動的マッピングで使用されるポート範囲の下限と上限を定義するオプション、および動的マッピングを無効にできるオプションが含まれています。

以下の表は、initialize-tsm スクリプトを実行する際のポートのオプションをリストしています。

表: initialize-tsm スクリプトのポートオプション

スクリプトのオプション	パラメーター	説明
-i	<port>	調整サービス クライアント ポートを設定します。
-e	<port>	調整サービス ピア ポートを設定します。
-m	<port>	調整サービス リーダー ポートを設定します。
-n	<port>	TSM エージェント ファイル転送 ポートを設定します。
-o	<port>	TSM コントローラー ポートを設定します。
-l	<min-port>	ポートを動的マッピングするために使用されるポート範囲の下限を設定します。
-r	<max-port>	ポートを動的マッピングするために使用されるポート範囲の上限を設定します。
--disable-port-		動的ポート マッピングを無効にします。

スクリプトのオプション	パラメーター	説明
remapping		これを実行する場合は、TSM および Tableau Server によって使用されるすべてのサービスまたはプロセスにポートを割り当てる必要があります。詳細については、前述の手動ポート割り当てを参照してください。

動的マッピングされたポート

この表は、動的マッピングされたポートを使用するプロセスまたはサービスをリストしています。

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
activemqserver.port	activemqserver:primary	ActiveMQ Service サービスポート。
activemqserver.openwire.port	activemqserver:openwire	ActiveMQ サービスオープンワイヤー ポート。
appzookeeper_0.client.port	appzookeeper:client	調整サービス クライアント ポート。
appzookeeper_0.peer.port	appzookeeper:peer	調整サービス ピア ポート。
appzookeeper_0.leader.port	appzookeeper:leader	調整サービ

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
		ス リーダー ポート。
backgrounder_0.port	backgrounder	バックグラウンダー プライマリ ポート。
backgrounder_0.debug.port	backgrounder:debug	バックグラウンダー デバッグ ポート。
backgrounder_0.jmx.port	backgrounder:jmx	バックグラウンダー JMX ポート。
backgrounder_0.jmx.rmi.port	backgrounder:jmx.rmi	バックグラウンダー JMX RMI ポート。
backgrounder_ 0.recommendations.trainer.port	backgrounder:recommendations.train-er	バックグラウンダー推奨 ポート。
backuprestore.port	backuprestore	バックアップ/復元 サービス ポート。
cacheserver_0.port	cacheserver	キャッシュ サーバー ポート

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
clustercontroller.status.port	clustercontroller:status	クラスタ コントローラー ステータス ポート。
clustercontroller.storage.port	clustercontroller:storage	クラスタ コントローラー ストレージ ポート。
databasemaintenance.port	databasemaintenance	データベース メンテナンス ポート。
dataserver_0.port	dataserver	データ サーバー プライマリ ポート。
dataserver_0.debug.port	dataserver:debug	データ サーバー デバッグ ポート。
dataserver_0.jmx.port	dataserver:jmx	データ サーバー JMX ポート。
dataserver_0.jmx.rmi.port	dataserver:jmx.rmi	データ サーバー JMX RMI ポート。

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
filestore.port	filestore	ファイル ストア プライマリ ポート。
filestore.status.port	filestore:status	ファイル ストア ステータス ポート。
gateway.port	gateway	ゲートウェイ ポート。この既定は 80 ですが、80 が使用できない場合は 8080 になります。8080 が使用できない場合は、8000 が試行されます。動的ポート割り当てが有効かどうかに関係なく、そのシーケンスに従

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
		います。それらのポートがどれも使用できず、動的マッピングが有効な場合は、定義された範囲内で使用可能なポートが選択されます。ゲートウェイ ポートはマルチノード クラスタ内のすべてのノードで同一である必要があるため、最初のノードでポート80が選択された場合は、これがすべてのノードで使用されるポートとな

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
hyper.port	hyper	データエンジンプライマリポート。
hyper.connection.port	hyper:connection	データエンジン接続ポート。
indexandsearchserver.port	indexandsearchserver	インデックスおよび検索サーバーのプライマリポート。
indexandsearchserver.transport.port	indexandsearchserver:transport	インデックスおよび検索サーバーのトランスポートポート。
licenseservice.vendor_daemon.port	licenseservice:vendor_daemon	ライセンスサービスベンダーデーモンポート。マルチノードインストールでノード間のライセンス関連の通

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
		信に使用されます。
samlservice.port	samlservice	SAML サービス ポート。
siteimportexport.port	siteimportexport	サイト インポート/エクスポート ポート。
tabadmincontroller.port	tabadmincontroller	TSM コントローラー ポート。
tabadminagent.columbo.port	tabadminagent:columbo	管理エージェント サービス検出 ポート。
tabadminagent.filetransfer.port	tabadminagent:filetransfer	TSM エージェント ファイル転送 ポート。
vizportal_0.authentication.port	vizportal:authentication	アプリケーション サーバー認証 ポート。
vizportal_0.authorization.port	vizportal:authorization	アプリケーション サーバー認可 ポート。

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
vizportal_0.maintenance.port	vizportal:.maintenance	アプリケーション サーバー メンテナンス ポート。
vizportal_ 0.microservice.extensions.port	vizportal:.microservice:extensions	アプリケーション サーバー拡張ポート。
vizportal_0.monolith_grpc.port	vizporta:monolith_grpc	アプリケーション サーバー GRPC ポート。
vizportal_0.publishing.port	vizportal:publishing	アプリケーション サーバー パブリッシュ ポート。
vizportal_0.recommendations.port	vizportal:recommendations	アプリケーション サーバー推奨ポート。
vizportal_0.port	vizportal	アプリケーション サーバー プライマリ ポート。
vizportal_0.debug.port	vizportal:debug	アプリケー

ポート名: json ファイルの構文 (configKeys)	ポート名: tsm CLI の構文	説明
		ション サーバー デバッグ ポート。
vizportal_0.jmx.port	vizportal:jmx	アプリケーション サーバー JMX ポート。
vizportal_0.jmx.rmi.port	vizportal:jmx.rmi	アプリケーション サーバー JMX RMI ポート。
vizqlserver_0.port	vizqlserver	VizQL Server プライマリ ポート。
vizqlserver_0.debug.port	vizqlserver:debug	VizQL Server デバッグ ポート。
vizqlserver_0.jmx.port	vizqlserver:jmx	VizQL Server JMX ポート。
vizqlserver_0.jmx.rmi.port	vizqlserver:jmx.rmi	VizQL Server JMX RMI ポート。

JMX ポートの有効化

Tableau Server の操作での問題の解決に役立てるために、Tableau サポートによってサーバーの JMX ポートの有効化を要求されることがあります。これらのポートは、通常 JConsole のようなツールで監視とトラブルシューティングを行う場合に便利です。バージョン 2022.1 以降では、JMX ポートを安全に有効にすることができます。これは推奨されている方法です。2022.1 より前のバージョンでは、セキュリティで保護されていないポートのみを有効にできます。

安全な JMX ポートの有効化

Tableau Server バージョン 2022.1 以降、JMX ポートを安全に有効にできます。この手順では、安全な JMX を有効にする方法について説明します。以前のバージョンで JMX を有効にするには、以下の「セキュリティで保護されていない JMX ポートの有効化」を参照してください。

Tableau Server で安全な JMX ポートを有効にするには、以下を行います。

1. sudo アクセス権を持つユーザーとして TSM をインストールするコンピューターにログオンします。
2. 以下のコマンドを実行します。

   ```
   tsm maintenance jmx enable
   ```

   コマンドには、実行時に指定できるいくつかのオプションがあります。コマンドの実行時にオプションを指定しない場合は、回答内容に基づいてオプションの入力を求められます。

   例:

   ```
   C:\Windows\system32>tsm maintenance jmx enable
   We do not recommend you enable JMX unsecured on a production
   environment. Would you like to enable security features for
   JMX?
   (y/n): y
   JMX access is readonly by default. Would you like to add
   readwrite access?
   (y/n): n
   Would you like to enable SSL?
   ```

```
(y/n): y
Would you like to require client authentication (mTLS)?
(y/n): n
Enabling JMX with the specified settings. This will perform a
server restart. Do you want to continue?
(y/n): y
Starting enable JMX asynchronous job.
```

コマンドとオプションの詳細については、「tsm maintenance jmx enable」を参照してください。

セキュリティで保護されていない JMX ポートの有効化

2022.1.0 より前の Tableau Server バージョンでは、JMX ポートはセキュリティで保護されていない状態でのみ有効にできます。

重要: セキュリティで保護されていない JMX ポートを有効にすると、いくつかのセキュリティリスクが発生する可能性があります。安全な JMX を使用することを強くお勧めします。これがサポートされている Tableau Server バージョンを持っていない場合は、リスクを認識し、JMX ポートへのアクセスをこのシナリオで実用的なクライアントの最小数に制限することによって、リスクを軽減してください。一般的に、ホストのファイアウォール ルール、外部セキュリティデバイス、ルーティング ルールなどを使用してアクセスを制限します。

セキュリティで保護されていない JMX ポートを Tableau Server で有効にするには

1. サーバーを停止します。
2. 次のコマンドを入力します。

   ```
   tsm configuration set -k service.jmx_enabled -v true
   ```

   保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```
3.

保留中の変更にサーバーの再起動が必要な場合は、`pending-changes apply` コマンドの実行時に、再起動が行われることを知らせるメッセージが表示されます。このメッセージはサーバーが停止していても表示されますが、その場合には再起動は行われません。`--ignore-prompt` オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。変更に再起動が必要ない場合は、メッセージなしで変更が適用されます。詳細については、tsm pending-changes applyを参照してください。

4. 調整サービスとTSM 管理コントローラーを再起動します (*tableau* システム アカウントとして)。

   ```
   sudo su -l tableau -c "systemctl --user restart appzookeeper_
   0.service"
   ```

   ```
   sudo su -l tableau -c "systemctl --user restart
   tabadmincontroller_0.service"
   ```

   tabadmincontroller の再起動には数分かかる場合があります。コントローラーを完全に再起動させる前に次の手順で保留中の変更を適用しようとする場合、TSM はコントローラーに接続できなくなります。`tsm status -v` コマンドを使用して、コントローラーが実行されていることを確認できます。Tableau Server 管理コントローラーが「実行中」として表示されます。

5. サーバーを起動します。

JMX ポートの決定のしくみ

既定では、使用可能なポートの範囲内からJMX ポートが動的に割り当てられます。ポート割り当てのしくみおよび動的マッピングの上書き方法の詳細については、Tableau サービス マネージャーのポートを参照してください。

ATRDiag.exe コマンド ライン リファレンス

ATRDiag コマンド ライン ユーティリティを使用すると、ログインベースのライセンス管理と実行承認 (ATR) 機能に関連するライセンス発行の問題を管理できます。

シノプシス

```
ATRDiag -dumpATR -showAll -showVerbose -product ["Tableau Desktop"]
["Tableau Prep"]["Tableau Server"]

-subProduct {subProduct} -version {version} -setVersion {version}

-container -serverDataPath {path}

-log [Trace][Debug][Info][Warn][Error][Fatal][Off] -status -
deleteAllATRs

-deleteInvalidATRs -deleteATR {atrIdToDelete}

-enableATRFeature -disableATRFeature -enableLBLMFeature -
disableLBLMFeature

-requireLBLMFeature -setDuration {seconds}
```

注: すべての ATRDiag オプションでは大文字と小文字が区別されません。

オプション

`-dumpATR`

コマンドプロンプト(存在する場合) に有効な実行承認 (ATR) を表示します。複数の ATR が有効である場合は、`-version/-product/-subProduct/-version` を使用して、どの ATR をダンプするかを制御します。

`-showAll`

レジストリの内容を列挙して、使用可能な ATR を確認します。(一部の ATR は無効な場合があります)。

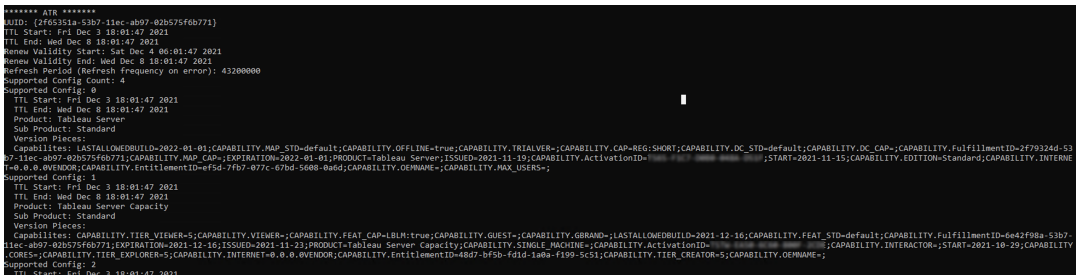
`-showVerbose`

レジストリの内容を列挙し、-version/-product/-subProduct/-version を使用して製品ごとに表示します。

-product ["Tableau Desktop"]["Tableau Prep"]["Tableau Server"]

既定は "Tableau Desktop" です。サーバー ATR を表示するには、"Tableau Server" を指定する必要があります。

例: atrdiag -product "Tableau Server"



-subProduct {subProduct}

既定は "プロフェッショナル".です。

-container

コンテナー モードは、Tableau Server でのみ使用できます。-product "Tableau Server" を指定する必要があります。

-serverDataPath

コンテナー モードでのサーバー データの場所。既定は "/var/opt/tableau/tableau_server/" です。

-version {version}

既定なし。このフィールドの有効な値は "Tableau 2021.1" です。

`-setVersion {version}`

-version の既定値を保持します。

`-log [Trace][Debug][Info][Warn][Error][Fatal][Off]`

ATR のログ情報を表示します。

`-status`

ATR 機能のステータス (有効または無効)、ライセンス サーバーを提供し、ATR をダンプします。

`-deleteAllATRs`

マシン上のすべての ATR を削除します。

`-deleteInvalidATRs`

無効な ATR をすべて削除します。

`-deleteATR {atrIdToDelete}`

ID ごとに ATR を削除します。

`-enableATRFeature`

ATR 機能をオンにします。管理者として実行する必要があります。Tableau Desktop でのみ使用できます。

`-disableATRFeature`

ATR 機能をオフにします。管理者として実行する必要があります。Tableau Desktop でのみ使用できます。

`-enableLBLMFeature`

ログインベースのライセンス管理 をオンにします (LBLM)。管理者として実行する必要があります。

`-disableLBLMFeature`

ログインベースのライセンス管理 をオフにします (LBLM)。管理者として実行する必要があります。

`-requireLBLMFeature`

ログインベースのライセンス管理 (LBLM) 機能を [必須] に設定します。管理者として実行する必要があります。

`-setDuration {seconds}`

`ATRRequestedDurationSeconds` を [秒] に設定します。管理者として実行する必要があります。

グローバル オプション

`-h, --help`

任意の項目です。

コマンドのヘルプを表示します。

initialize-tsm スクリプトの出力ヒント

以下のヘルプ コンテンツは、次のコマンドを実行したときに出力されます。

```
sudo ./initialize-tsm -h
```

initialize-tsm スクリプトは、`/opt/tableau/tableau_server/packages/scripts.<version_code>/` にインストールされています。

出力

```
REQUIRED
--accepteula           Indicate that you have accepted the End User
License Agreement (EULA).
                       You can find the EULA in /opt/tableau/tableau_
server/packages/docs.<version_code>

OPTIONAL
-c config-name         Set the service configuration name.
                       If not set, the default is "tabsvc".

-d data-directory      Set a custom location for the data directory
                       if it's not already set. If not set, the default is
                       "/var/opt/tableau/tableau_server".

-b bootstrap-file      Optional. Location of the bootstrap file
downloaded from the Tableau Services Manager
                       on existing node. Must be provided to join existing Tableau
Server cluster.

-u username            Name of the user with admin privileges on
existing Tableau Services Manager.
                       Required if -b option specified.

-p password            Password for the Tableau Services Manager
admin user.
Note: This option was removed beginning in version 2021.3.0 to
improve script security.

-f                     Bypass warning messages.

-g                     Do NOT add the current user to the
"tsmadmin" administrative
                       group, used for default access to Tableau Services Manager,
                       to the "tableau" group, used for easier access to log files.
```

```
-a username            The provided username will be used as the
user to be added
                       to the appropriate groups, instead of the user running this
                       script. Providing both -a and -g is not allowed.

-q                             Quiet, suppress output except for errors and
warnings.

-i coordinationservice-client-port      Client port for the
coordination service

-e coordinationservice-peer-port        Peer port for the
coordination service

-m coordinationservice-leader-port      Leader port for the
coordination service

-t licenseservice-vendordaemon-port     Vendor daemon port for the
licensing service

-n agent-filetransfer-port              Filetransfer port for the
agent service

-o controller-port                      Https port for the
controller service

-l port-range-min                       Lower port bound for
automatic selection

-r port-range-max                       Upper port bound for
automatic selection

--disable-port-remapping                Disable automatic port
selection
```

```
--unprivileged-user=<value>            Name of the unprivileged account
to run Tableau Server.
Default: "tableau".

--tsm-authorized-group=<value>         Name of the group(s) that allows
authorization to access Tableau Services Manager.
Default: "tsmadmin".

--disable-account-creation                    Do not create groups or
user accounts for Server and TSM authorization.
                                       However, the values in: unprivileged-user an
group will still be used
                                       in TSM configuration.

--http_proxy=<value>                    Http forward proxy for
Tableau Server. Its value should be http://<proxy_address>:<proxy_
port>/
                                        For example, --http_proxy=http://1.2.3.4:31
proxy=http://example.com:3128/

--https_proxy=<value>                   Https forward proxy for
Tableau Server. Its value should be http://<proxy_address>:<proxy_
port>/
                                                                  For example, --https
proxy=http://example.com:3128/
                                                                  Take care to use htt
https_proxy environmental variable.
                                                                  Do not specify the h
https_proxy environmental variable.

--no_proxy=<value>                     Environment variable that directs certain UR
to bypass the forward proxy. For example,
                                       --no_proxy=localhost,127.0.0.1,localaddress,

--[no-]activation-service              Specify whether the Tableau
authorization-to-run (ATR) service should be used to activate
```

```
Tableau Server.
                                        This option is ideal for cloud-based or virt
but is available to anyone who
                                        can activate their copy of Tableau Server on
for product activation is a permanent
                                        choice that cannot be undone later. If no op
the Tableau authorization-to-run (ATR) service will be used to
activate Tableau Server.
```

関連トピック

- initialize-tsm によるポートリマッピングの制御
- TSM のインストールと初期化
- システム ユーザーとグループ
- データディレクトリ

upgrade-tsm Script の出力ヒント

以下のヘルプ コンテンツは、次のコマンドを実行したときに出力されます。

```
sudo ./upgrade-tsm -h
```

upgrade-tsm script は、`/opt/tableau/tableau_server/packages/scripts.<version_code>/` にインストールされています。

出力

```
Usage: upgrade-tsm --accepteula [optional arguments]

Upgrade Tableau Server cluster to version <version number>.
This script should be run from any Tableau Server cluster node after
Tableau Server <version number> package
has been installed on all nodes.
```

```
REQUIRED
--accepteula                              Indicate that you have
accepted the End User License Agreement (EULA).
                                          You can find the EULA in docs directory

-u <value>, --username=<value>            TSM administrator user
name. Required if it is run using a non-TSM administrator
                                          account on the initial node, or if upgrading
prior to 2019.2.
-p <value>, --password=<value>            TSM administrator password.
Required if the --username option is specified.
                                          If a password is required but not provided,
prompted for the password.
OPTIONAL
--debug                                   Print each command as it is
run for debugging purposes. Produces extensive
output.
--trust-admin-controller-cert             Do not validate the server
certificate.
--no-prompt                               Suppress script prompts.
You will only be prompted for missing required
                                          parameters, for example, if you specify a us
password. If the
                                          script needs to stop or restart Tableau Serv
without warning or
                                          prompting. Use this if you automate the upgr
with a script.

--external-repository-config-file=filename
                                          Required if upgrading from a Tableau Server
Tableau Server
                                          that uses a later major version of PostgreSQ
configured to use an
                                          external repository. The filename is a confi
describing a new
```

```
                                        instance of the external repository. The new
   use the same
                                        type of external service as the current exte
   but with the
                                        supported version of PostgreSQL.

   --external-repository-cert-file=filename
                                        Required if upgrading from a Tableau Server
   Tableau Server
                                        that uses a later major version of PostgreSQ
   to use an
                                        external repository. The filename is an SSL
   file for the new
                                        external repository. The new repository shou
   type
                                        of external service as the current external
   with the
                                        supported version of PostgreSQL.
```

サーバーのバージョン表示

実行中の Tableau Server のバージョンは大切です。これによって、アクセス可能な機能や権限が決まります。また場合によっては、アップグレード方法はアップグレード元のバージョンとアップグレード先のバージョンによって決まるため、アップグレードの際にも重要となります。バージョンの確認は、一度見つけ方がわかれば簡単です。

- Tableau Server でのバージョンの表示 — すべてのサーバー ユーザーは、サーバー Web UI の [ヘルプ] メニューから、Tableau Server のバージョンを表示できます。
- Tableau サービス マネージャー (TSM) でのバージョンの表示 — TSM 管理者は、TSM コマンド ライン (CLI) から TSM と Tableau Server のバージョンを表示できます。

Tableau Server Web UI からのサーバー バージョンの表示

- Tableau Server へのログイン中に、情報アイコン( )、**[Tableau Server について]** をクリックします。

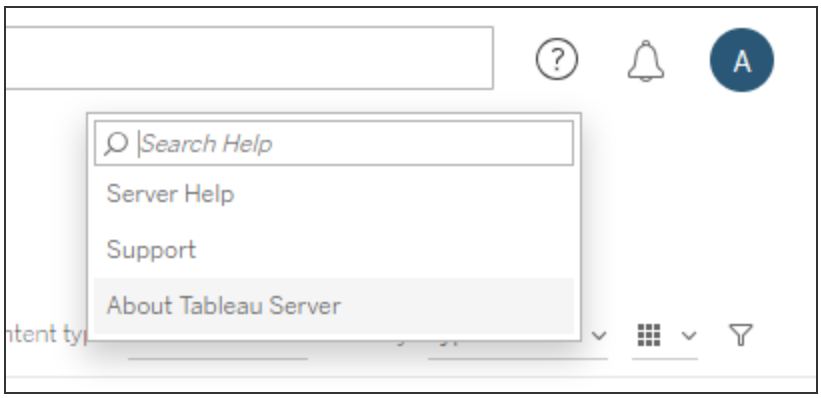


Tableau Server のバージョンが [Tableau Server について] ダイアログ ボックスに記載されています。

TSM コマンド ラインからのサーバー バージョンとTSM バージョンの表示

1. 最初のノード(TSM がインストールされているノード) で管理者としてコマンドプロンプトを開きます。
2. 次のコマンドを実行します。

```
tsm version
```

出力に Tableau サービス マネージャー (TSM) とTableau Server のバージョンが表示されます。

例:

```
C:\>tsm version
Tableau Services Manager command line version 20182.18.1214.0751.
Tableau Server version 20182.18.1214.0751.
```

短いバージョン、長いバージョン、version_code

バージョン番号を知る必要があるほとんどの場合、"短い" バージョンを知る必要があります。このバージョン番号は [Tableau Server について] ダイアログ ボックスに表示され、メジャー バージョン、マイナー バージョン、およびメンテナンス バージョンの 3 つの部分で構成されています。短いバージョン番号は nnnn.n.n の形式です。例: 2018.2.5.

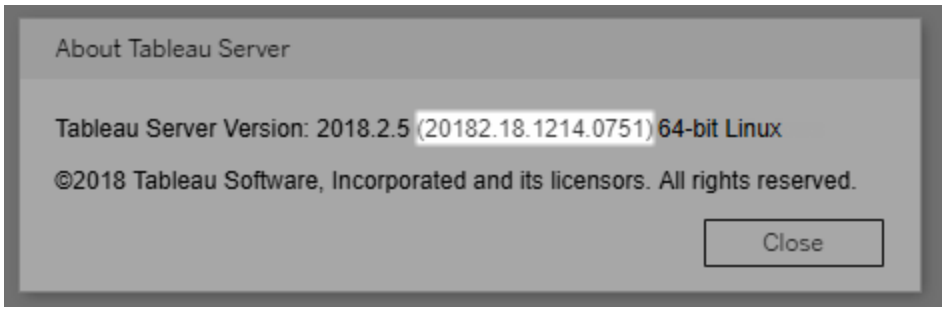
長いバージョンにはメジャー バージョン番号とマイナー バージョン番号のほか、他のビルド固有の情報が含まれ、nnnnn.nn.nnnn.nnnn の形式で、20182.18.1214.0751 などとなります。このドキュメントで <version_code> への参照があるときは、長いバージョンを意味します。このドキュメントでは、この version_code や長いバージョンへの参照は、インストール場所やスクリプトまたはビン ディレクトリへのパスについて取り上げている場合に最も多く行われます。

Tableau Server は /packages ディレクトリの、パス名に長いバージョンを含むサブディレクトリにインストールされています。つまり、/packages ディレクトリ内の場所に移動しようとしている場合は、長いバージョンを含める必要があります。たとえば、/scripts ディレクトリでは次のとおりです: /opt/tableau/tableau_server/packages/scripts.<version_code>/。

例: /opt/tableau/tableau server/packages/scripts.20182.18.1214.0751

長いバージョン番号の確認

また、この長いバージョンは、[Tableau Server について] ダイアログ ボックスでかっこに囲まれて表示されています。



Einstein Discovery 統合の構成

バージョン 2021.1.0 以降の Tableau Server では Einstein Discovery と統合することができます。ワークブックやダッシュボードの作成者と閲覧者が Einstein Discovery の予測を利用できるようになりました。バージョン 2021.2.0 以降では、Web 上でフローを作成するときに Einstein Discovery 予測も利用できるようになりました。

Tableau の Einstein Discovery は salesforce.com が提供しています。適用条件については、salesforce.com との契約を確認してください。

Tableau Server と Einstein Discovery を統合するには、いくつかの構成手順を実施する必要があります。手順のいくつかは Tableau Server 側で必要であり、手順のいくつかは Einstein Discovery を実行している Salesforce 組織側で必要です。ここでは、ダッシュボードの拡張機能、分析の拡張機能、および Tableau Prep 拡張機能の手順の概要を説明し、サーバーの構成手順が記載されている特定のトピックへのリンクを示します。

Tableau における Einstein Discovery 予測の使用方法 (ライセンス発行とパーミッションの要件を含む) については、Tableau Desktop および Web 作成ヘルプの「Tableau の Einstein Discovery 予測の統合」を参照してください。予測をフローに追加する方法については、「Einstein Discovery 予測をフローに追加する」を参照してください。

Einstein Discovery ダッシュボードの拡張機能

Einstein Discovery のダッシュボード拡張機能を使用すると、ワークブックの作成者は Tableau で予測をリアルタイムに表示できます。ダッシュボード拡張機能は、Tableau ワークブックのソース

データと、Salesforce にデプロイされた Einstein Discovery を利用したモデルを使用して、インタラクティブにオンデマンドで予測を提供します。

Einstein Discovery ダッシュボードの拡張機能を使用するように Tableau Server を構成するには、以下を行う必要があります。

1. Tableau Server で次の手順を実行します。
    a. Tableau Server のデータ接続と拡張機能に対して OAuth の保存済みトークンを有効にします。保存済みアクセス トークンの許可
    b. サーバーのダッシュボード拡張機能を有効にします。「Tableau Server でダッシュボードと Viz の拡張機能を管理する」を参照してください。
2. Salesforce で、Einstein Discovery を実行している組織において次の手順を実行します。
    a. Salesforce.com で CORS を構成して、Einstein Discovery を Tableau Server に統合する。
    b. Salesforce で、Tableau CRM を実行している組織において接続アプリを作成します。「ステップ 1: Salesforce に接続されたアプリを作成する」を参照してください。
3. Tableau Server で、接続アプリからの情報を使用して、Salesforce の保存済み OAuth 認証資格情報用にサーバーを構成します。「ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する」

Einstein Discovery の分析拡張機能

Einstein Discovery 分析拡張機能を使用すると、Tableau の計算フィールドに予測を直接埋め込むことができます。Salesforce にデプロイされたモデルからの予測をテーブル計算スクリプトで要求するには、関連する予測 ID とモデルに必要な入力データを渡します。Salesforce の Model Manager を使用して Tableau のテーブル計算スクリプトを自動生成し、そのスクリプトを Tableau ワークブックで使用できるように計算フィールドに貼り付けます。

Einstein Discovery の分析拡張機能を使用するように Tableau Server を構成するには、以下を行う必要があります。

1. Tableau Server で次の手順を実行します。
    a. Tableau Server のデータ接続と拡張機能に対して OAuth の保存済みトークンを有効にします。保存済みアクセス トークンの許可
    b. サーバーの分析拡張機能を有効にし、接続タイプを構成します。分析拡張機能を使用した接続の構成を参照してください

2. Salesforce で、Einstein Discovery を実行している組織の接続アプリを作成します。ステップ1: Salesforce に接続されたアプリを作成するを参照してください。
3. Tableau Server で、接続アプリからの情報を使用して、Salesforce の保存済み OAuth 認証資格情報用にサーバーを構成します。「ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する」

Einstein Discovery Tableau Prep の拡張機能

バージョン 2021.2.0 以降の Tableau Server および Tableau Cloud でサポートされています

Einstein Discovery Tableau Prep の拡張機能を使用すると、ユーザーは Web 上でフローを作成するときに、Einstein 予測をフローに直接埋め込むことができます。

Einstein Discovery Tableau Prep の拡張機能を使用するように Tableau Server または Tableau Cloud を構成するには、次を行う必要があります。

1. Tableau Server で次の手順を実行します。
   a. Tableau Server のデータ接続と拡張機能に対して OAuth の保存済みトークンを有効にします。保存済みアクセストークンの許可を参照してください。
   b. サーバーに対して Tableau Prep 拡張機能を有効にします。Tableau Prep 拡張機能を有効にする参照してください。
2. Salesforce で、Einstein Discovery を実行している組織の接続アプリを作成します。ステップ1: Salesforce に接続されたアプリを作成するを参照してください。
3. Tableau Server で、接続アプリからの情報を使用して、Salesforce の保存済み OAuth 認証資格情報用にサーバーを構成します。「ステップ 2: Salesforce.com の Oauth 用に Tableau Server を構成する」

Salesforce.com で CORS を構成して、Einstein Discovery を Tableau Server に統合する

バージョン 2021.1.0 では、Einstein Discovery 予測を Tableau ダッシュボードに統合する機能が追加されました。これを行うには、Einstein Discovery ダッシュボードの拡張機能を使用します。前提条件として、Tableau CRM をホストしていて、かつ、使用するモデルと予測が含まれる Salesforce org で、クロスオリジンリソース共有 (CORS) を構成する必要があります。

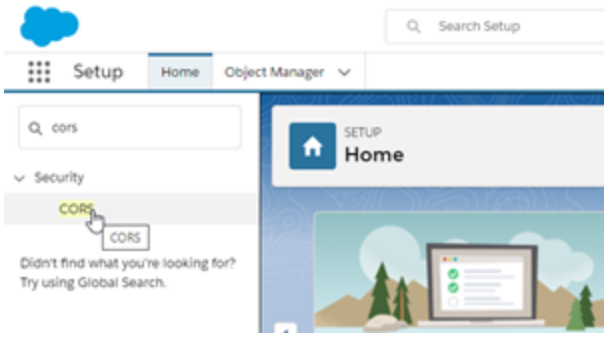
この手順では、Salesforce.com 組織の管理者がこの構成を行う方法について説明します。CORS の詳細については、Salesforce ドキュメントの「Salesforce CORS 許可リストの設定」を参照してください。

Einstein Discovery 用の CORS を構成します。

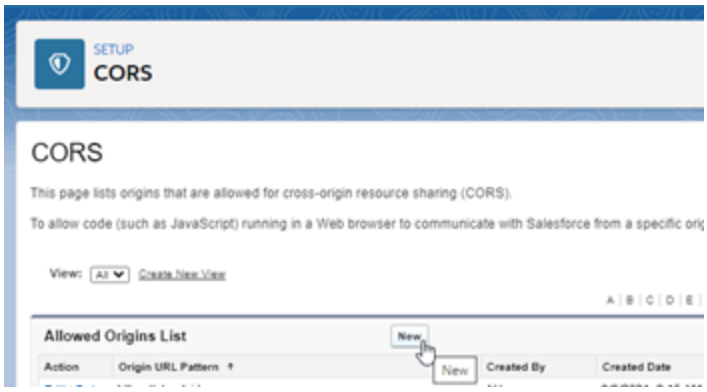
> **注:** この手順では、Salesforce Lightning でのプロセスについて説明します。従来のインターフェイスを使用している場合、ナビゲーションは異なる場合がありますが、設定は同じです。

1. Salesforce.com 開発者アカウントにサインインし、右上のユーザー名をクリックしてから **[設定]** を選択します。

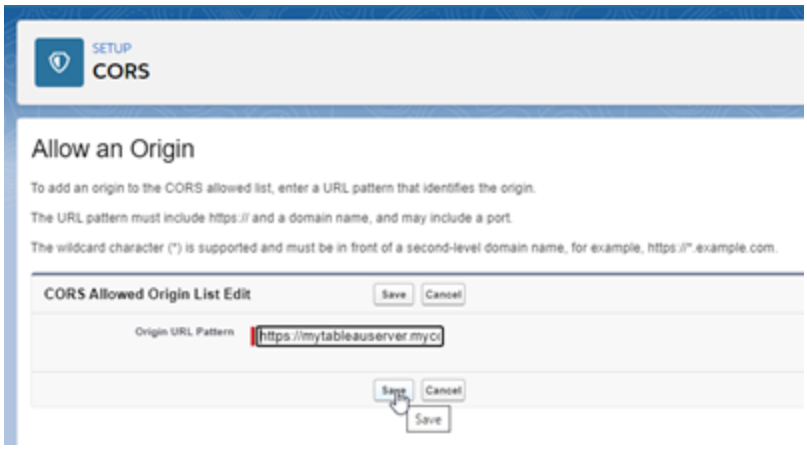
2. 左側のナビゲーション列で「CORS」を検索し、**[CORS]** を選択します。



3. **CORS** の **[Allowed Origins List (オリジン許可リスト)]** セクションで、**[New (新規)]** をクリックします。



4. **[CORS Allowed Origin List Edit (CORS オリジン許可リストの編集)]** ボックスに、Tableau Server の URL を「https://」から入力します。



URL パターンの詳細については、Salesforce 開発者向けドキュメント https://developer.salesforce.com/docs/atlas.ja-jp.chatterapi.meta/chatterapi/extend_code_cors.htm を参照してください。

5. **[保存]** をクリックします。

分析拡張機能を使用した接続の構成

Tableau では、分析拡張機能に式を渡して、R や Python、Einstein Discovery と統合できるようにする関数セットがサポートされています。

> **注**: R および Python スクリプトを使用すると、Tableau Prep フローで複雑なクリーニング操作を実行することができますが、サポートされる構成と機能が異なる場合があります。詳細については、Tableau Prep ヘルプの「フローでの R および Python スクリプトの使用」を参照してください。

このトピックでは、拡張機能を使用してサイトを Tableau Server で構成する方法について説明します。

Tableau Server は認証機構を提供しているため、Tableau Desktop より Tableau Server の方がより安全に分析拡張機能をユーザーに表示できます。

ユーザーシナリオの詳細と Tableau Desktop または Tableau Cloud の構成については、*Tableau Desktop と Web 作成*のヘルプの「分析拡張機能を使用した式の受け渡し」を参照してください。

この記事の構成ステップは、ワークブックに固有のものです。R および Python スクリプトを使用して予測モデリングデータをフローに組み込む方法については、*Tableau Prep* ヘルプの「フローでの R および Python スクリプトの使用」を参照してください。

Server SSL

分析拡張用に SSL を構成するには、Tableau Server を実行しているコンピューターに有効な証明書をインストールする必要があります。証明書は Tableau Server を実行しているコンピューターによって信頼されている必要があります。証明書の [件名] フィールドや SAN エントリの 1 つが、分析拡張機能サービス構成の URI と完全に一致する必要があります。

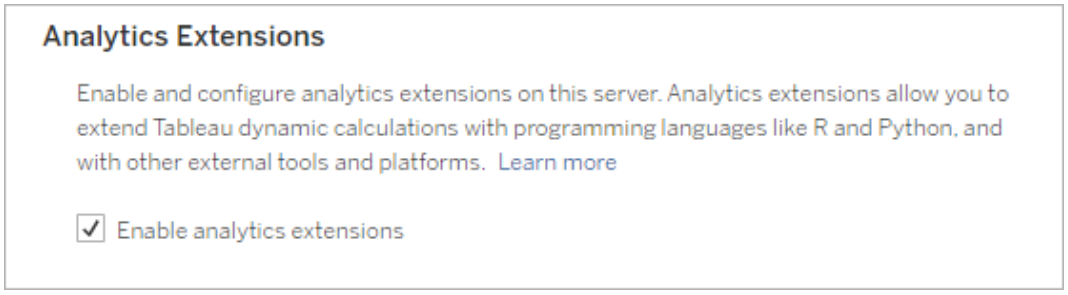
分析拡張機能を有効にする

拡張機能を構成する前に、サーバー全体で分析拡張機能を有効にする必要があります。

1. Tableau Server 管理者エリアへのサインインします。
    - サーバー上にサイトが 1 つしかない場合は (既定)、**[設定]** をクリックし、ステップ 2 に進みます。

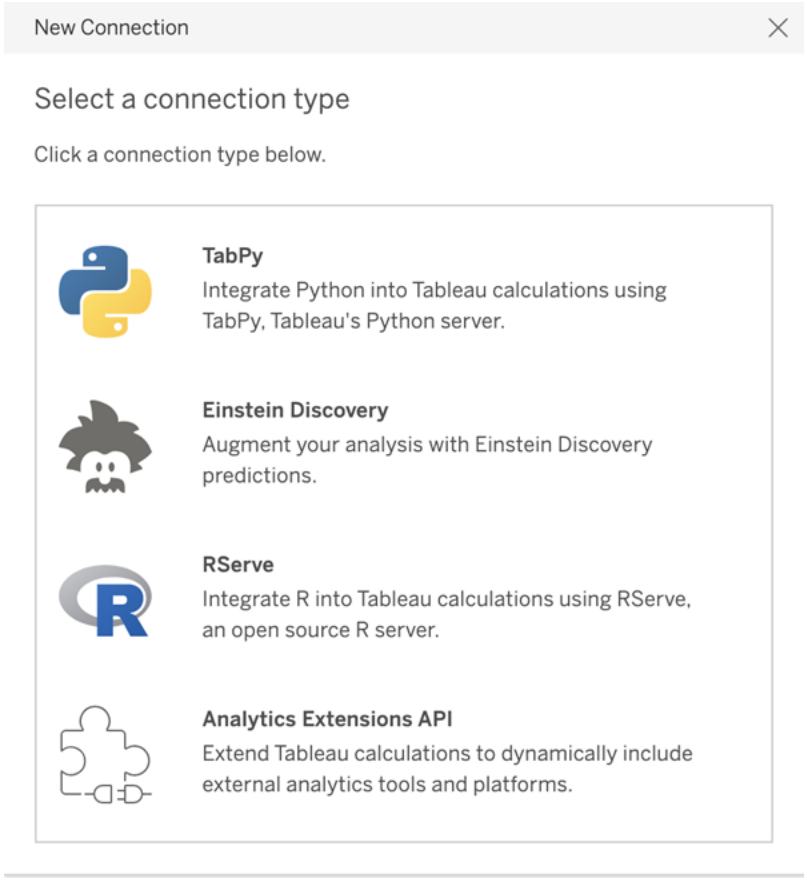
- サーバーに複数のサイトがある場合は、
  a. **[すべてのサイト]** で、**[すべてのサイトを管理]** をクリックします。
  b. **[拡張]** タブをクリックします。

2. **[分析拡張機能]** までスクロールし、**[Enable analytics extension (分析拡張機能を有効にする)]** を選択して **[保存]** をクリックします。



分析拡張機能設定の構成

1. Tableau Server 管理者エリアへのサインインします。
2. [設定] ページで **[拡張機能]** タブをクリックし、**[分析拡張機能]** までスクロールします。(Tableau Server のマルチサイト展開で、分析拡張機能を構成するサイトに移動し、**[設定] > [拡張機能] をクリック**します。)
3. **マルチサイト展開のみ:** 各サイトで分析拡張機能を有効にする必要があります。[分析拡張機能] で、**[Enable analytics extensions for site (サイトの分析拡張機能を有効にする)]** を選択します。
4. [分析拡張機能] で、**[新しい接続の作成]** をクリックします。
5. **[新しい接続]** ダイアログで、追加する接続タイプをクリックし、分析サービスの構成設定を入力します。



6. 構成する必要があるオプションは、選択した接続タイプによって異なります。

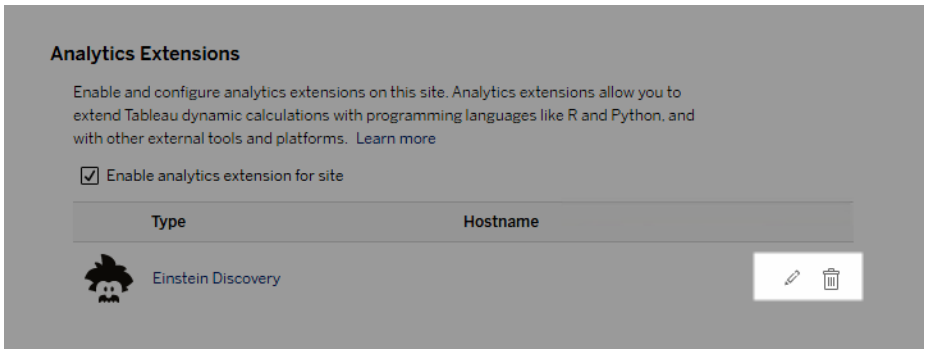
   - Einstein Discovery 接続の場合は、**[有効]** をクリックします。
   - TabPy、RServer、および分析拡張機能 API 接続の場合は、次の情報を入力します。
     - **接続名**: (必須) 接続するサーバーのタイプを指定します。RSERVE は、RServe パッケージを使用した R への接続に対応します。TABPY は、TabPy を使用した Python への接続やその他の分析拡張機能への接続に対応します。
     - **SSL を必須にする** (推奨): 分析サービスへの接続を暗号化するには、このオプションを選択します。**[ホスト名]** フィールドで `HTTPS` URL を指定する場合は、このオプションを選択する必要があります。
     - **ホスト名**: (必須) 分析サービスを実行しているコンピューター名や URL を指定します。このフィールドでは大文字と小文字を区別します。
     - **ポート**: (必須) サービスのポートを指定します。

- **ユーザー名およびパスワードを指定してサインイン**(推奨): 分析サービスの認証に使用するユーザー名とパスワードを指定するには、このオプションを選択します。

7. **[作成]** をクリックします。

分析拡張機能接続の編集または削除

構成を編集または削除するには、サイトの **[拡張機能]** タブにある **[分析拡張機能]** に移動します。



[編集] アイコンまたは **[削除]** アイコンをクリックし、プロンプトに従って構成を変更します。

クライアント要件: Rserve 外部サービスの中間証明書チェーン

Tableau Server 経由で Rserve 外部接続に接続している Tableau Desktop コンピューター (Windows と Mac) に完全な証明書チェーンをインストールする必要があります。この要件は、Rserve がセキュア接続でハンドシェイクを管理する方法が原因です。

Tableau Desktop にルート証明書をインポートするだけでは不十分であり、証明書チェーン全体をクライアントコンピューターにインポートする必要があります。

スクリプトエラー

Tableau では、分析拡張機能を使用するワークブックが Tableau Server に適切にレンダリングされることを確認できません。必須の統計ライブラリがユーザーのコンピューター上では使用できても Tableau Server で使用している分析拡張機能インスタンス上では使用できない場合があります。

分析拡張機能を使用するビューがワークブックに含まれている場合、ワークブックをパブリッシュするときに警告が表示されます。

> このワークシートには外部サービス スクリプトが含まれており、管理者が外部サービス接続を構成するまでターゲットプラットフォーム上では表示できません。

分析拡張機能の使用状況の判定

バージョン 2021.2 以降、分析拡張機能の設定はワークブック レベルで反映されます。これにより、管理者はカスタム ビューを使用して Tableau リポジトリにクエリを実行することができ、どのワークブックでどの拡張機能が使用されているか、それらはどのくらいの頻度で使用されているかを判断できます。

これを行うには、ワークブック接続テーブルをワークブックの使用状況を示すテーブルに結合する必要があります。カスタム管理ビューの作成と使用の詳細については、「Tableau Server リポジトリでデータを収集する」および「カスタム管理ビューの作成」を参照してください。

アイデンティティストアの変更

インフラストラクチャやビジネスの変更により、Tableau Server でのアイデンティティストアの変更が必要となる場合があります。アイデンティティストアには、ローカルと外部の 2 種類があります。Tableau Server をインストールした場合は、ローカルまたは外部のアイデンティティストアを構成しました。

ローカルのアイデンティティストアで Tableau Server を構成する場合は、すべてのユーザー情報およびグループ情報は Tableau Server リポジトリに格納されて管理されます。ローカル アイデンティティストアのシナリオでは、ユーザーおよびグループ用の外部のソースはありません。

外部のストアで Tableau Server を構成する場合は、すべてのユーザー情報およびグループ情報は外部ディレクトリサービスによって格納され管理されます。Tableau Server は、ローカルのユーザーとグループのコピーが Tableau Server リポジトリに存在できるように外部のアイデンティティストアと同期する必要があります。しかし、ユーザーとグループのすべてのデータの信頼できるソースは外部のアイデンティティストアです。外部のアイデンティティストアには、OpenLDAP や Active Directory などがあります。

Tableau アイデンティティストアの詳細については、アイデンティティストアを参照してください。

ローカル ストアから外部 ストアに変更したり、外部 ストアからローカル ストアに変更したりできます。いずれの場合も、アイデンティティストアのタイプを変更するには、次の手順を完了します。

1. Tableau Server をアンインストールし、再インストールします。完全なアンインストールとクリーン インストールの手順についてはこのトピックの最後に説明します。
2. コンテンツとパーミッションを復元します。

   これらの手順における"復元" という用語は、作成したバックアップを `TSM maintenance restore` コマンドによって復元することを指してはいません。ターゲットの Tableau Server とは異なるアイデンティティストアを使用する Tableau Server インスタンスで作成したバックアップ (`.tsbak`) を復元することはできません。バックアップは、元の Tableau Server 構成に戻す必要が生じた場合に使用できる最善の保護策です。

警告

Tableau Server でのインストール タイプの変更は複雑で、時間のかかるプロセスになる場合があります。ユーザーのデータのコンテンツの紛失や孤立を防ぐため、このプロセスを慎重に計画する必要があります。いずれの場合でも、ワークブックおよびデータ ソースに適用されるユーザー フィルターは、変更後に手動で更新する必要があります。

最も重要な点として、Tableau Server を再インストールした後にコンテンツとパーミッションを新しいアイデンティティストアに移行する方法を決定します。

コンテンツとパーミッションの復元方法

次のリストは、Tableau Server を再インストールした後にコンテンツとパーミッションを復元するための 2 つの方法について説明しています。環境要件に最も適した方法を選択してください。

- **方法 1: サイトのエクスポートとインポートを使用する —** この方法では、最初に、既存の展開で各サイトをエクスポートします。次に、新しいサーバーをインストールして、新しいアイデ

ンティティストアのタイプ用に構成します。次に、新しいサーバーの既定サイトで新しいユーザーを作成します。最後に、元のサイトをすべてインポートします。インポート段階で、既定サイトで作成した新しいユーザーに元の ID をマッピングできます。

> **注:** Tableau Server のインスタンス間でサイトを移行する場合、移行先サイトは、移行元サイトの Tableau Server のバージョンと同じかそれ以降の Tableau Server のバージョン上にある必要があります。移行元サイトと移行先サイトの両方が、サポートされているバージョンの Tableau Server からのものである必要があります。

この方法はすべてのコンテンツとパーミッションを各サイトでエクスポートするため、組織にとっての優れた方法は、アイデンティティストアの変更が完了した後でコンテンツとパーミッションの厳密なレプリカを要求することです。組織によっては、認証を変更した結果としてアイデンティティストアの変更が必要となる場合があります。こうした場合、別のユーザー名構文の設定が新しいモデルでの要件となることが多くなります。この方法には元のユーザー名を新しいユーザー名にマッピングするプロセスが含まれ、それぞれのシナリオに柔軟性を提供します。

- **方法 2: フレッシュインストール、ユーザーがコンテンツを再パブリッシュする** — この方法では、Tableau Server の新しいバージョンをインストールし、設定中に新しいアイデンティティストアを選択します。また、新しいサイトを作成します。次に、ユーザーを作成して彼らにアクセス権を供与し、ユーザーはワークブックやデータソースを再パブリッシュします。他の方法とは異なり、この方法では、既存の Tableau Server インフラストラクチャを再使用しません。

  この方法は、非常に自動化された小さな展開とデータに精通したユーザーに最適です。管理上の観点から、この方法はコンテンツを積極的に移植しないため、最もシンプルです。ただし、コンテンツのパブリッシュは完全にユーザーに依存しているため、コンテンツの一元化された監督が必要となる大きな組織では、この方法は成功しない場合があります。

ユーザー フィルター

ユーザー フィルターはドメイン固有のものです。そのため、Tableau Server のドメインが変わったり、認証タイプが変わったりすると、フィルターが期待どおりに機能しなくなります。ユーザー フィルターは Tableau Server で生成されますが、ユーザーが設定を行った後は、ワークブックおよびデータソース

に保存されます。これらの方法のどれでアイデンティティストアを変更しても、ワークブックまたはデータソースのコンテンツは変更されません。

アイデンティティストアの変更を計画する際は、Tableau Desktop ですべてのワークブックおよびデータソースのユーザーフィルターを修正する最終作業を必ず計画に含めてください。

ユーザー名とTableau アイデンティティストア

方法 1 を使用している場合は、Tableau Server が Tableau アイデンティティストアでユーザー名をどのように保存するかを知っておくと役立ちます。Tableau は、Tableau Server のさまざまなサービスを使用してコンテンツ パーミッションやサイト メンバーシップを調節するリポジトリで、すべてのユーザー ID を保存します。一般的に、Active Directory 向けに構成されているアイデンティティストアではユーザー名が `domain\username` の形式で保存されています。一部の組織は UPN (`jsmith@domain.lan`) を使用しています。

一方、ローカルのアイデンティティストアを使用して Tableau Server を構成する組織は通常、標準の切り詰められたユーザー名 (`jsmith` など) を作成します。

すべての場合において、これらのユーザー名は Tableau アイデンティティストア内で一意のリテラル文字列にする必要があります。1 つのアイデンティティストア タイプから別のタイプに変更する場合は、ターゲットの認証である SSO またはユーザープロビジョニング ソリューションで特定のユーザー名形式が必要となる場合があります。

そのため、すべてのパーミッション、コンテンツ、およびユーザーの実行可能性を保つため、アイデンティティストア タイプの変更後に次のいずれかが当てはまる必要があります。

- 新しいユーザー名は元のユーザー名と一致している必要があります。または
- 新しい書式設定に一致する用、元のユーザー名を更新する必要があります。

認証の変更によりアイデンティティストアの変更が引き起こされる場合、ターゲット認証スキームにより元のユーザー名と異なるユーザー名構文が使用される可能性があります。メソッド 1 には、元のユーザー名を新しいユーザー名にマッピングできるプロセスが含まれます。

元のユーザー名の書式設定が新しい認証タイプで機能する可能性があります。たとえば、ローカル アイデンティティストア展開で UPN 名を使用している場合、Active Directory 展開で同じユーザー名を使用できる場合があります。また、ユーザーが Tableau Server へのサインインに引き続き

domain\username 形式を使用している場合は、その形式をローカル アイデンティティストアに使用することもできます。

ローカル アイデンティティストアを外部 Active Directory ストアに変更する場合は、計画プロセスの一部として、外部 ID ストアを使用した展開におけるユーザー管理トピックを見直してください。

方法 1: サイトのエクスポートとインポートを使用する

エクスポート操作とインポート操作には、同じバージョンの Tableau Server を使用する必要があります。

1. すべてのサイトをサーバーにエクスポートします。サイトのエクスポートまたはインポートを参照してください。
2. バックアップを作成し、削除してから再インストール
3. Tableau Server で新規ユーザーを作成します。元のサーバーの各ユーザーに対応する新しいユーザーが必要です。
4. ステップ 1 でエクスポートしたサイトをインポートします。サイトのエクスポートまたはインポートを参照してください。インポート時、新しいユーザーを元のユーザーにマッピングするよう要求されます。

方法 2: フレッシュインストール – ユーザーがコンテンツを再パブリッシュする

アイデンティティストアの変更の一部としてコンテンツの移植を計画しない場合であっても、サーバーをバックアップすることをお勧めします。

1. バックアップを作成し、削除してから再インストール。
2. ユーザー、サイトおよびグループを作成します。
3. 新しい Tableau Server についてユーザーに通知し、新しい認証資格情報を提供してからコンテンツのパブリッシュを許可します。

バックアップを作成し、削除してから再インストール

どちらの方法にも、次のステップが含まれます。

1. Tableau Server のバックアップ
2. Tableau Server の削除。
3. 新しいアイデンティティストア タイプを使用した Tableau Server の再インストール。

ステップ 1: Tableau Server のバックアップ

ベストプラクティスとして、続行前にサーバーをバックアップする必要があります。

TSM コマンドライン インターフェース (CLI) を使用したバックアップの作成の手順に従ってください。 `-d` オプションを使用して `backup` コマンドを実行します。`-d` オプションによりデータスタンプが追加されます。

完了したら、バックアップ ファイル (`.tsbak`) を Tableau Server インストールの一部ではない安全な場所にコピーします。

ステップ 2: Tableau Server の削除

コンピューターから Tableau Server を完全に削除する必要があります。コンピュータから Tableau Server を削除するを参照してください。

ステップ 3: 新しい認証タイプを使用して Tableau Server を再インストールする

1. Tableau のカスタマー ポータルにサインインし、Tableau のユーザー名とパスワードを使ってサイン インしたら、Tableau Server をダウンロードします。
2. Tableau Server をインストールします。詳細については、Tableau Server のインストールと構成を参照してください。インストール時、新しいアイデンティティストア タイプを選択します。初期ノード設定の構成を参照してください。

外部のアイデンティティストアの構成リファレンス

Tableau Server は、LDAP を使用した外部ディレクトリへの接続に対応しています。このシナリオでは、Tableau Server は外部 LDAP ディレクトリから Tableau Server リポジトリにユーザーをシステム ユーザーとしてインポートします。

このトピックでは、Tableau Server が対応しているすべての LDAP 関連の構成オプションについて説明します。Active Directory に接続する場合は、手動で接続を構成するのではなく、セットアップの一部として Tableau Server との LDAP 接続を自動的に構成することを強くお勧めします。「初期ノード設定の構成」を参照してください。

このリファレンスに一覧表示されているオプションは、LDAP に準拠した任意のディレクトリに使用できます。LDAP を構成したことがない場合は、ディレクトリ管理者または LDAP の専門家と協力して行ってください。

このトピックはリファレンスに関するものです。Tableau Server でユーザー情報を格納して管理する方法の詳細については、「アイデンティティストア」を参照してください。

構成方法

Tableau Server を LDAP ディレクトリに接続できるようにする構成パラメーターは、yml ファイルに格納されます。これらのファイルは、Tableau Server のさまざまなサービスによって管理および同期されます。.yml ファイルの更新は、Tableau サービス マネージャー (TSM) インターフェイスを使用して行う必要があります。

テキスト エディタを使用して yml ファイルを直接更新しないでください。適切に動作させるためには、TSM ですべての更新を管理する必要があります。

yml 構成ファイルは、キー値のペアで構成されます。たとえば、`wgserver.domain.username` というキーには、値としてユーザー名が入っています。このキーは、バインド操作中に LDAP ディレクトリの認証に使用するユーザー名を定義します。

TSM では yml キー値を設定できる 4 つの方法があります。例として `wgserver.domain.username` というキーを使用して、4 つの異なる設定方法を説明します。

- configKey キー値ペア - .yml 構成ファイルのキーを更新するには、tsm configuration set のオプションを実行している `wgserver.domain.username` キーを更新するか、configKey エンティティの下の JSON 構成ファイルにキーを含めます。「構成ファイルの例」を参照してください。

  JSON 構成ファイルの configKey キー値ペアは、`tsm configuration set` で使用するものと同じですが、設定が異なります。このトピックでは、どちらの方法も *configKey* と呼びます。

  以下で説明する configEntities およびネイティブの tsm コマンドを使用する場合とは異なり、configKey の入力は検証されません。configKey を使用してオプションを設定する場

合、入力する値は参照元となるyml構成ファイルにリテラル文字列としてコピーされます。たとえば、`true` または `false` が有効な入力であるキーの場合、configKey キー値ペアを使用してキーを構成すると、任意の文字列値を入力でき、キー用に保存されます。このような場合、無効な値は確実にLDAP構成のエラーにつながります。

configKeys を使用するのは、次に示す他の3つのオプション(configEntities、ネイティブの tsm コマンド、TSM Web UI)で構成を設定するオプションがない場合に限ることをお勧めします。configKeys を使用するときは、必ず値をダブルチェックして、大文字と小文字を区別するようにしてください。

- configEntities JSON - *configEntities* JSON で `username` オプションを渡すことにより、.yml 構成ファイルを更新できます。

  JSON ファイルの configEntities オプションを使用して値を設定すると、値は保存される前に検証されます。値の大文字と小文字は区別されます。configEntities を使用して値を設定する方法の詳細については、「identityStore エンティティ」の例を参照してください。JSON ファイルは、tsm settings import コマンドによりインポートされます。configEntities で使用できるオプションは、すべての .yml キー値ペアの一部です。

  検証とは、JSON ファイル内のすべての値が有効なデータ型である場合にのみ、インポートコマンドが成功することを意味します。たとえば、`true` または `false` のみを受け入れる値に対して `no` を入力した場合は、エラーが返され、構成はインポートされません。

  JSON 構成ファイルは、初期構成の一部としてのみインポートできます。JSON 構成ファイルをインポートして Tableau Server を初期化した後に LDAP に変更を加える必要がある場合は、JSON ファイルをインポートし直さないでください。代わりに、可能であればネイティブの tsm コマンドを使用するか、configKeys と `tsm configuration set` を使用して、個々のキーを変更します。

- ネイティブ tsm コマンド - *ネイティブ tsm コマンド* `tsm user-identity-store` で `ldapuser` オプションを渡すことにより、.yml 構成ファイルを更新できます。configEntities と同様に、ネイティブ tsm コマンドで入力した値は、保存する前に検証されます。

.yml ファイル内のすべてのキー値ペアが、ネイティブ tsm コマンドを使用して設定できるわけではありません。

- TSM GUI — TSM GUI を使用して、セットアップ中に構成値を設定できます。Active Directory に接続していて、セットアップ中に GUI を使用して Tableau ID ストアを構成する場合は、AD の読み取りアクセス権を持つアカウントの入力を求められます。`wgserver.domain.username` キーは、認証資格情報を入力するときに設定されます。

  このシナリオは、Active Directory に接続している場合にのみ機能します。Tableau Server では、GUI セットアッププロセスの一部としての任意の LDAP 構成はサポートされていません。

Tableau のアイデンティティストア構成ツールを使用して LDAP json 構成ファイルを生成することを検討してください。Tableau のアイデンティティストア構成ツールでは、tsm configuration set のオプションを実行して設定できるキー/値のペアのリストも生成されます。ツール自体は Tableau ではサポートされていません。ただし、ファイルを手動で作成する代わりにこのツールで作成された JSON ファイルを使用しても、サポートされるサーバーのステータスは変わりません。

Active Directory の構成

Tableau Server を構成して Active Directory を使用する場合は、インストールに TSM Web UI を使用することをお勧めします。TSM Web UI は、必要な入力を最小限に抑えながら Active Directory 向けに Tableau Server を構成するよう最適化されています。初期ノード設定の構成を参照してください。

構成リファレンステーブル

configEntities オプション **(オプションでは、大文字と小文字が区別されます)**	**ネイティブの tsm コマンド**	**configKey** **(tsm configuration set コマンドまたは JSON ファイルの configKeys セクションで使用)**	**シナリオ**	**注**

type	N/A	wgserver.authenticat-e	A-D-、LD-A-P-、ロ-ーカル	ユーザーの ID 情報を格納する場所。値: `local` または `activedirectory` LDAP サーバーに接続する場合は、`activedirectory` と入力します。
sslPort	N/A	wgserver.domain.ssl_port	A-D-、LD-A-P	LDAP サーバーのセキュア ポートを指定します。シンプル バインドにはセキュア LDAP が推奨されます。通常 LDAPS はポート636 です。
N/A	N/A	Wgserver.domain.lda-p.starttls が有効になりました。	A-D-、LD-A-P	値: `true` または `false` バージョン 2021.2 以降、Tableau Server が Active Directory に接続するように構成されている場合、このキーは既定で `true` に設定されています。その結果、有効な SSL/TLS 証明書が Tableau キーストアに存在する場合、LDAP ディレクトリへのシンプル バインドが暗号化されます。詳細については、外部の LDAP アイデンティティストアへの暗号化チャネルの構成を参照してください。 This key is set to `false` by default

				when Tableau Server is configured to connect to a an arbitrary (but not Active Directory) LDAP server. このキーはバージョン 2021.1 で導入されました (設定はされていません)。
port	N/A	wgserver.domain.port	AD、LDAP	LDAP サーバーの非セキュア ポートを指定するためにこのオプションを使用します。プレーン テキストは通常 389 です。
domain	domain	wgserver.domain.default	AD	Active Directory 環境で、Tableau Server がインストールされているドメインを指定します (例: "example.lan")。 Active Directory 環境ではない LDAP の場合: この値に入力した文字列は、ユーザー管理ツールの [ドメイン] 列に表示されます。任意の文字列を入力できますが、キーを空白にすることはできません。 このキーを wgserver.domain.fqdn にすると冗長です。両方のキーの値は同じにする必要があります。 ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
username	ldapusername	wgserver.domain.username	AD	ディレクトリサービスとの接続に使用するユーザー名です。

	me		、LD-A-P	指定するアカウントには、ディレクトリサービスをクエリするパーミッションが必要です。 Active Directory にはユーザー名 (`jsmith` など) を入力します。 LDAP サーバーには、接続に使用するユーザーの識別名 (DN) を入力します。たとえば、`"cn=jsmith,dc=example,dc=la-n"` とします。 ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
password	ldapp-assw-ord	wgserver.domain.pas-sword	A-D-、LD-A-P	LDAP サーバーとの接続に使用するユーザー アカウントのパスワード。 ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
directorySe-rviceType	N/A	wgserver.domain.dire-ctoryservice.type	A-D-、LD-A-P	接続する LDAP ディレクトリサービスのタイプ。値: `activedirectory` または `openldap`
kerberosPri-ncipal	kerbp-rincip-al	wgserver.domain.lda-p.principal	A-D-、LD-	ホストマシン上の Tableau Server のサービスプリンシパル名。keytab には、このプリンシパルに対するパーミッションがなくてはなりません。

			A-P	/etc/krb5.keytab にある既存のシステム keytab を使用しないでください。代わりに、新しいサービスプリンシパル名の登録を推奨しています。指定された keytab のプリンシパルを表示するには、klist -k コマンドを実行します。Keytab 要件の理解を参照してください。 ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
hostname	hostn-ame	wgserver.domain.lda-p.hostname	A-D-、LD-A-P	LDAP サーバーのホスト名です。この値のホスト名または IP アドレスを入力できます。ここで指定するホストは、プライマリドメインのユーザー/グループ クエリに使用されます。ユーザー/グループ クエリが他のドメインにある場合、Tableau Server は DNS にクエリを実行して適切なドメイン コントローラーを識別します。 ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
membersR-etrievalPag-eSize	N/A	wgserver.domain.lda-p.members.retrieval.-page.size	A-D-、LD-A-P	このオプションによって、LDAP クエリで返される結果の最大数が決まります。 たとえば、Tableau Server で 50,000 ユーザーを含む LDAP グループをインポートするというシナリオについて検討します。単一の操作でこのような多数のユーザーが含まれるインポートを試行することは、ベストプラクティスではありませ

				ん。このオプションを1,500 に設定すると、Tableau Server は最初の応答でまず 1,500 ユーザーをインポートします。それらのユーザーが処理されたら、Tableau Server で次の1,500 ユーザーをLDAP サーバーからリクエストする、という手順です。 このオプションは、お使いのLDAP サーバーの要件に対応する場合にのみ変更することをお勧めします。
N/A	N/A	wgserver.domain.ldap.connectionpool.enabled	AD、LDAP	このオプションを `true` に設定すると、Tableau Server は LDAP サーバーにクエリを送信するときに同じ接続を再利用しようとします。この動作によって、新規リクエストごとにLDAP サーバーで再認証を行うオーバーヘッドを軽減することができます。接続プールは、シンプルバインドおよびTSL/SSL バインドによる接続でのみ機能します。接続プールは、GSSAPI バインド接続ではサポートされていません。
N/A	N/A	wgserver.domain.accept_list	AD	Tableau Server からセカンダリアクティブディレクトリドメインへの接続を許可します。セカンダリドメインはユーザーが同期するときに Tableau Server が接続するドメインですが、Tableau Server はインストールされていません。Tableau Server が他の Active Directory ドメインに確実に接続できるようにするには TSM で `wgserver.domain.accept_list` オプションを設定して信頼されるドメイン

				を指定する必要があります。詳細については、wgserver.domain.accept_list を参照してください。
N/A	N/A	wgserver.domain.whitelist	AD	**重要:** バージョン 2020.4.0 で非推奨になりました。代わりに wgserver.domain.accept_list を使用してください。 Tableau Server からセカンダリアクティブディレクトリドメインへの接続を許可します。セカンダリドメインはユーザーが同期するときに Tableau Server が接続するドメインですが、Tableau Server はインストールされていません。Tableau Server が他の Active Directory ドメインに確実に接続できるようにするには TSM で `wgserver.domain.whitelist` オプションを設定して信頼されるドメインを指定する必要があります。詳細については、wgserver.domain.whitelistを参照してください。
kerberosConfig	kerbconfig	ダイレクトマッピングなし	AD、LDAP	ローカル コンピューター上の Kerberos 構成ファイルのパス。Active Directory にインストールしている場合、ドメインで結合したコンピューターにある可能性がある既存の Kerberos 構成ファイルや keytab ファイルを使用することは推奨されていません。アイデンティティストアを参照してください。

				ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
kerberosKeytab	kerbkeytab	ダイレクト マッピングなし	AD、LDAP	ローカル コンピューター上の Kerberos keytab ファイルのパス。Tableau Server サービス専用のキーで keytab ファイルを作成し、この keytab ファイルをコンピューター上の他のアプリケーションと共有しないことが推奨されます。たとえば、Linux では `/var/opt/tableau/keytab` ディレクトリに keytab ファイルを配置することができます。 ネイティブの tsm コマンド: tsm user-identity-store set-connection [options] コマンドを使用します。
nickname	N/A	wgserver.domain.nickname	AD、LDAP	ドメインのニックネーム。これは Windows/Active Directory 環境の NetBIOS 名としても参照されます。すべての LDAP エンティティに `nickname` オプションが必要です。値を NULL にすることはできません。組織でニックネームや NetBIOS 名が不要な場合は、ブランク キーを渡します (例: "")。
root	N/A	wgserver.domain.ldap.root	LDAP	LDAP ルートでドメイン コンポーネントを使用しない場合、またはより複雑なルートを指定する必要がある場合には、LDAP ルートの設定が必要です。"o=my,u=root" の形式を使用します。

				たとえば、example.lan ドメインのルートは "o=example,u=lan" となります。
serverSideSorting	N/A	wgserver.domain.ldap.server_side_sorting	LDAP	クエリ結果をサーバー側で並べ替えるように LDAP サーバーを構成しているかどうかを示します。お使いの LDAP サーバーがサーバー側の並べ替えをサポートしている場合は、このオプションを true に設定します。LDAP サーバーがこれをサポートしているかどうかが不明な場合は、構成でエラーが発生する可能性があるため false と入力します。
rangeRetrieval	N/A	wgserver.domain.ldap.range_retrieval	LDAP	リクエストに対するクエリ結果の範囲を返すように LDAP サーバーを構成しているかどうかを示します。これは、多数のユーザーが含まれるグループが一度にではなく小分けにしてリクエストされることを意味します。範囲取得をサポートする LDAP サーバーは、クエリが大きい場合により良いパフォーマンスを発揮します。お使いの LDAP サーバーが範囲取得をサポートしている場合は、このオプションを true に設定します。LDAP サーバーがこれをサポートしているかどうかが不明な場合は、構成でエラーが発生する可能性があるため false と入力します。
bind	N/A	wgserver.domain.ldap.bind	LDAP	ディレクトリサービスへの通信を保護する方法。Kerberos で LDAP サーバーに接続しているのでなければ、LDAP には「simple」と入力してください。Kerberos には、「gssapi」と入力します。

N/A	N/A	wgserver.domain.ldap.domain_custom_ports	LDAP	**注**: このキーは、Tableau Server on Linux でのみサポートされます。 子ドメインとその LDAP ポートをマッピングできます。ドメインとポートはコロン (:) で区切られ、ドメインとポートのペア間は `FQDN1:port,FQDN2:port` の形式のようにコンマ (,) で区切られます。 例: `tsm configuration set -k wgserver.domain.ldap.domain_custom_ports -v childdomain1.lan:3269,childdomain2.lan:3269,childdomain3.lan:389`
distinguishedNameAttribute	N/A	wgserver.domain.ldap.dnAttribute	LDAP	ユーザーの識別名を格納する属性です。この属性はオプションですが、LDAP クエリのパフォーマンスが大きく向上します。 **重要**: 初期構成の一部としてこのオプションを設定しないでください。このオプションは、LDAP の全体的な機能を検証した後にのみ設定します。このキーを設定する前に、組織内で dnAttribute を設定する必要があります。
groupBase-	N/A	wgserver.domain.lda-	LD-	グループに代替ルートを指定します。た

Dn		p.group.baseDn	A-P	とえば、グループの全員が「groups」と呼ばれるベース組織に保存されている場合は、"o=groups" と入力します。
N/A	class-name-s	wgserver.domain.lda-p.group.classnames	LD-A-P	既定では、Tableau Server は「group」、「groupOfNames」、または「groupOfURLs」を検索します。お使いの LDAP グループ オブジェクトが既定のクラス名に適合しない場合、この値を設定することでデフォルトを上書きします。コンマで区切って複数のクラス名を入力できます。 グループ名にカンマが含まれる場合は、バックスラッシュ(\) でエスケープする必要があります。たとえば、グループ名が groupOfNames, top の場合は、"groupOfNames\, top" と入力します。 Tableau の LDAP 実装では、LDAP オブジェクトはユーザーまたはグループとして解釈されます。したがって、最も具体的なクラス名を入力するようにしてください。ユーザーとグループの間でクラス名が重複すると、競合が発生する可能性があります。 ネイティブの tsm コマンド: tsm user-identity-store set-group-mappings [options] コマンドを使用します。
groupBase-	basef-	wgserver.domain.lda-	LD-	Tableau Server のユーザー グループが

Filter	ilter	p.group.baseFilter	A-P	使用するように設定するフィルターです。オブジェクトクラス属性および組織単位属性を指定する場合があります。例: "(&(objectClass=groupofNames)(ou=Group))" "(&(objectClass=inetOrgPerson)(ou=People))" が LDAP 実装で機能しない場合は、Tableau ユーザーベースに対して機能する基本フィルターを指定します。 これは必須のキーです。空白にすることはできません。 ネイティブの tsm コマンド: tsm user-identity-store set-group-mappings [options] コマンドを使用します。
groupName	group-name	wgserver.domain.lda-p.group.name	LD-A-P	LDAP サーバーのグループ名に対応する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-group-mappings [options] コマンドを使用します。
groupEmail	group-Email	wgserver.domain.lda-p.group.email	LD-A-P	LDAP サーバーのグループの電子メールアドレスに対応する属性です。 ネイティブの tsm コマンド: tsm user-

				identity-store set-group-mappings [options] コマンドを使用します。
groupDescr-iption	descr-iption	wgserver.domain.lda-p.group.description	LD-A-P	LDAP サーバーのグループの説明に対応する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-group-mappings [options] コマンドを使用します。
member	mem-ber	wgserver.domain.lda-p.group.member	LD-A-P	グループに含まれるユーザーの識別名のリストを含む LDAP 属性を指定します。 ネイティブの tsm コマンド: tsm user-identity-store set-group-mappings [options] コマンドを使用します。
N/A	N/A	wgserver.domain.lda-p.group.memberURL	LD-A-P	動的なグループに LDAP クエリを格納する LDAP 属性の名前を指定します。
userBaseD-n	N/A	wgserver.domain.lda-p.user.baseDn	LD-A-P	ユーザーに代替ルートを指定します。たとえば、すべてのユーザーが「users」と呼ばれるベース組織に保存されている場合は、`"o=users"` と入力します。
N/A	class-name-s	wgserver.domain.lda-p.user.classnames	LD-A-P	既定では、Tableau Server が「user」および「inetOrgPerson」の文字列を含む LDAP ユーザー オブジェクト クラスを探します。お使いの LDAP ユーザー オブジェクトが既定のクラス名に適合しない場合、この値を設定することでデフォルトを上書きします。コンマで区切って複数のクラス名を入力できます。例:

				"userclass1, userclass2"。 ユーザー名にカンマが含まれる場合は、バックスラッシュ(\) でエスケープする必要があります。たとえば、名前が Names, top の場合は、"Names\, top" と入力します。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
userBaseFilter	basefilter	wgserver.domain.ldap.user.baseFilter	LDAP	Tableau Server のユーザーが使用するように設定するフィルターです。オブジェクトクラス属性および組織単位属性を指定する場合があります。 例: "(&(objectClass=inetOrgPerson)(ou=People))" ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
userUsername	ldapusername	wgserver.domain.ldap.user.username	LDAP	LDAP サーバーのユーザー名に該当する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
userDispla-	displ-	wgserver.domain.lda-	LD-	LDAP サーバーのユーザーの表示名に

yName	ayna-me	p.user.displayname	A-P	該当する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
userEmail	email	wgserver.domain.lda-p.user.email	LD-A-P	LDAP サーバーのユーザーの メール アドレスに該当する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
userCertific-ate	certifi-cate	wgserver.domain.lda-p.user.usercertificate	LD-A-P	LDAP サーバーのユーザーの証明書に該当する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
N/A	thum-bnail	wgserver.domain.lda-p.user.thumbnail	LD-A-P	LDAP サーバーのユーザーのサムネイル画像に該当する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
userJpegP-hoto	jpegp-hoto	wgserver.domain.lda-p.user.jpegphoto	LD-A-P	LDAP サーバーのユーザープロフィール画像に該当する属性です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
memberOf	mem-	wgserver.domain.lda-	LD-	ユーザーが メンバーになっているグループ

	berof	p.user.memberof	A-P	です。 ネイティブの tsm コマンド: tsm user-identity-store set-user-mappings [options] コマンドを使用します。
groupClass-Names	N/A	wgserver.domain.lda-p.group.classnames	LD-A-P	既定では、Tableau Server は「group」、「groupOfNames」、または「groupOfURLs」を検索します。お使いの LDAP グループ オブジェクトが既定のクラス名に適合しない場合、この値を設定することでデフォルトを上書きします。 configEntities の場合: このオプションは、各 クラスを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。例: `["basegroup","othergroup"]`。 configKey の場合: 各 クラスをコンマ (スペースなし) で区切り、二重引用符で囲んで入力します。例: `"basegroup,othergroup”`。
userClassN-ames	N/A	wgserver.domain.lda-p.user.classnames	LD-A-P	既定では、Tableau Server が「user」および「inetOrgPerson」の文字列を含む LDAP ユーザー オブジェクト クラスを探します。お使いの LDAP ユーザー オブジェクトが既定のクラス名に適合しない場合、この値を設定することでデフォルトを上書きします。

				configEntities の場合:このオプションは、各 クラスを引用符で囲み、コンマ (スペースなし) で区切り、括弧内で囲む必要がある文字列のリストを受け取ります。例: `["userclass1",userclass2”]`。 configKey の場合: 各 クラスをコンマ (スペースなし) で区切り、二重引用符で囲んで入力します。例: `"userclass1,userclass2”`。

計算済みの configKeys

以下の Kerberos 関連の configKeys が計算され、複数の環境入力に従って設定されます。このため、configKeys は ネイティブの tsm コマンドや configEntities で設定する必要があります。これらの configKeys を手動で設定しようとしないでください。

計算済みの configKeys	**ネイティブの TSM コマンドを使用するには:**	**configEntity json を使用する場合:**
wgserver.domain.ldap.kerberos.conf, cfs.ldap.kerberos.conf	Kerberos 構成ファイルの場所を、tsm user-identity-store set-connection [options] コマンドの `kerbconfig` オプションで設定します。	Kerberos 構成ファイルの場所を、`kerberosConfig` configEntity オプションで設定します。
wgserver.domain.ldap.kerberos.keytab, cfs.ldap.kerberos.keytab	Kerberos キータブファイルの場所を、tsm user-identity-store set-connection [options] コマンドの	Kerberos キータブファイルの場所を、`kerberosKeytab` configEntity オプションで設定します。

	`kerbkeytab` オプションで設定します。	

サポート対象外の configKeys

参照元となる .yml 構成ファイルには、サポート対象外の configKeys がいくつか存在します。以下のキーは標準の展開向けではありません。これらのキーを構成しないようにしてください。

- wgserver.domain.ldap.kerberos.login
- wgserver.domain.ldap.guid
- Wgserver.domain.fqdn: このキーを wgserver.domain.default にすると冗長です。両方のキーの値は同じにする必要があります。値が wgserver.domain.default と一致しない場合にのみ、wgserver.domain.fqdn を更新します。

製品基礎データ

既定では、Tableau 製品から製品利用状況データが Tableau に送信されるため、当社はお客様がソフトウェアをどのように使用しているかを理解でき、さらに、お客様がどこで成功していて、どこで問題に遭遇する可能性があるかのインサイトを得ることで、問題の対処に役立てることができます。たとえば、このデータを使用すると、アップグレードが失敗しやすい箇所を把握できるため、それらの問題に対処するよう製品を変更したり、特定のバージョンの Tableau Server に適用されるセキュリティ問題に関する情報がどのユーザーベースで必要とされているかを特定したりすることができます。このデータの送信は、インストール時またはそれ以降に無効にすることができます。これを行う方法の詳細については、Tableau Desktop または Tableau Server の手順を参照してください。

製品利用状況データの送信を無効にしても、特定の製品基礎データは Tableau に送信されます。この製品基礎データには、製品とそのプロセスに関する情報 (実行している製品やプロセス、起動時刻、実行しているオペレーティング システム、ライセンス情報、データを送信したマシンまたはクラスタ(一意の仮名識別子を使用)、製品が製品利用状況データを送信するように構成されているかどうかなど) が含まれます。

製品基礎データの送信をマシン レベルまたはエンタープライズ レベルで無効にするには、**prod.telemetry.tableausoftware.com** に送信されるトラフィックをブロックします。

個々のコンピューターで製品基礎データの共有を無効にする

重要: この手順では、ローカル hosts ファイルを変更します。これが何であるかわからない場合は、変更しないでください。この変更は、ファイルに変更を加えることによる影響とファイルの変更方法を理解していて、かつ、安全対策としてファイルのバックアップが作成してある場合にのみ行ってください。

hosts ファイルを変更すると、コンピューターのネットワーク動作が変わります。hosts ファイルの変更に関する詳細な手順は、Microsoft、Apple、Linux ディストリビューションなどのオペレーティング システム プロバイダーによって提供されます。

1. 既存のホスト ファイルのコピーを作成し、それを Tableau コンピューターではないコンピューターに保存します。これは、変更を元に戻す必要がある場合に備えるためのバックアップです。バックアップ コピーを作成するまで、ファイルの変更を開始しないでください。
2. 次の行を含むようにコンピューターの hosts ファイルを変更します。

```
# Stops sending Product Usage to Tableau
(prod.telemetry.tableausoftware.com).
# Learn more here: http:\\tableau.com\derived-data
127.0.0.1    prod.telemetry.tableausoftware.com
```

最初の行と2 番目の行には 3 行目を説明するコメントを入力します。

3 行目の入力は、prod.telemetry.tableausoftware.com (http://prod.telemetry.tableausoftware.com/) へのすべてのトラフィックを内部ホスト ループバック アドレスに送信することで、ローカル マシンから発信されなくなるようにするものです。データはコンピューターの外部に送信されません。

エンタープライズ レベルで製品基礎データの共有を無効にする

エンタープライズ レベルで製品基礎データの送信を無効にするには、ネットワーク ファイアウォールを変更して、prod.telemetry.tableausoftware.com にアウトバウンド トラフィックが送信されないようにします。

このドメインは、Tableau がプロセスの起動とシャットダウンに関する製品基礎データを受信するために使用されます。また、一般的な製品利用状況データにも使用されます。このドメインへのトラフィックをブロックすると、どちらのタイプのデータも送信されなくなります。

このドメインへのトラフィックは、ポート 80 (製品データクライアントの初期登録用) とポート 443 (後続のすべてのトラフィック) で発生します。製品データの送信を完全に停止するには、このドメインへのすべてのトラフィックをブロックします。

ネットワークファイアウォールの構成方法の詳細については、ベンダーまたは社内の IT 部門に問い合わせてください。Tableau ではこれらの手順を提供することはできません。

アーカイブ済みコンテンツ

これはアーカイブされたコンテンツです

パブリッククラウドへの展開は引き続きサポートされますが、サードパーティーのパブリッククラウドへの展開に関するコンテンツは更新されなくなります。

Tableau Server の展開の最新コンテンツについては、「Tableau Server 導入ガイド」および Tableau Server ヘルプの「展開」「」セクションを参照してください。

アクセスが可能な場合は、Tableau Cloud を使用することをお勧めします。詳細については、以下を参照してください。

- Tableau Cloud 手動移行ガイド
- 管理者向けの Tableau Cloud トライアル版
- Tableau Cloud: 管理者向けのスタート

パブリッククラウドサービスで Tableau Server をセルフホストする

これはアーカイブされたコンテンツです

パブリッククラウドへの展開は引き続きサポートされますが、サードパーティーのパブリッククラウドへの展開に関するコンテンツは更新されなくなります。

Tableau Server の展開の最新コンテンツについては、「Tableau Server 導入ガイド」および Tableau Server ヘルプの「展開」「」セクションを参照してください。

アクセスが可能な場合は、Tableau Cloud を使用することをお勧めします。詳細については、以下を参照してください。

- Tableau Cloud 手動移行ガイド
- 管理者向けの Tableau Cloud トライアル版
- Tableau Cloud: 管理者向けのスタート

概要

独自のインフラおよびサーバー ハードウェアを所有していなくても、エンタープライズ レベルの Tableau Server をクラウド上に展開することができます。クラウド ベースのソリューションを構築することは、オンプレミスの設置に比較して多数の利点があります。たとえば、高価なハードウェアを購入して取りそろえる必要がなくなるため、クラウド上に Tableau Server ソリューションを構築するのにかかる全体のコストは、同様の環境をオンプレミスで用意するよりも普通はずっと安上がりになります。さらに、クラウドはアップタイム、信頼性、障害耐性の面でより良いパフォーマンスを提供できます。この傾向は異なる地域および可用性ゾーンにわたってソリューションを展開する際に顕著です。

> Tableau Server on Windows については、「パブリック クラウド サービスで Tableau Server をセルフホストする」参照してください。

お使いの Tableau 環境を以下の cloud 環境で構築し拡張可能です。

- **アマゾン ウェブ サービス** - Tableau Server をアマゾン ウェブ サービス (AWS) にインストールして管理します。詳細については、「Tableau Server を AWS Cloud にインストールする」を参照してください。
- **Google Cloud Platform** - Tableau Server を Google Cloud Platform にインストールして管理します。詳細については、「Tableau Server を Google Cloud Platform にインストールする」を参照してください。

- **Microsoft Azure** - Tableau Server を Microsoft Azure にインストールして管理します。詳細については、「Tableau Server を Microsoft Azure にインストールする」を参照してください。
- **Alibaba Cloud** - Tableau Server を Alibaba Cloud にインストールして管理します。詳細については、「Alibaba Cloud 上での Tableau Server のインストール」を参照してください。

Tableau Server 上の Tableau Advanced Management について

重要: 2024 年 9 月 16 日以降、Advanced Management は独立したアドオン オプションとしては利用できなくなりました。Advanced Management 機能は、以前に Advanced Management を購入した場合、または特定のライセンス エディション (Tableau Enterprise (Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

Tableau Advanced Management は、セキュリティ、管理性、拡張性の機能が強化された、Tableau Server の機能のコレクションです。Tableau Enterprise をお持ちの場合は、Advanced Management を利用できます。

注: Tableau Advanced Management には、いくつかの Tableau Server 機能と、個別にインストールされている 2 つのツール、Tableau Content Migration Tool と Tableau Resource Monitoring Tool が含まれています。詳細については、このトピック後半の機能に関する表を参照してください。

Advanced Management のライセンス要件

Advanced Management は展開ごとにライセンスされ、これにはユーザーベースまたはコアベースの場合があります。展開には、ライセンスされている本番環境の Tableau Server インストールのほ

か、実稼働環境のインストールをサポートし、ライセンスされている非本番環境の Tableau Server インストールが 2 つ含まれています。展開の詳細については、エンドユーザーライセンス契約に関する文書を参照してください。

- Advanced Management は、ライセンスが付与されている Tableau Server 展開でのみ有効にすることができます。つまり、Advanced Management プロダクトキーを適用する前に、ユーザーベースまたはコアベースの有効なキーを使用して Tableau Server をまずライセンス認証する必要があります。Advanced Management 機能を購入してプロダクトキーを取得する方法については、アカウントマネージャーに問い合わせてください。
- プロダクトキーがアクティブで有効な場合は、Advanced Management に含まれているすべての機能を使用できます。
- Advanced Management のプロダクトキーを削除または非アクティブ化すると、有効な Advanced Management ライセンスを必要とする機能を使用できなくなります。関連付けられたデータは削除されません。ライセンスの有効期限が切れる場合の状況については、各機能にわずかな違いがある場合があります。個々の機能の詳細については、以下の表のリンクを参照してください。

次の表では、有効な Advanced Management ライセンスを必要とする含まれている機能を示しています。

機能	説明	機能を使用する要件	バージョン
Tableau Resource Monitoring Tool について	Tableau Server の正常性を包括的に確認できます。Resource Monitoring Tool を使用すると、読み込み時間の遅延、抽出の失敗、その他の重要な問題を引き起こす原因を特定でき、エンドユーザー エクスペリエンスに影響を与える問題に前もって対応するのに役立ちます。	監視している Tableau Server インストールには Advanced Management ライセンスが必要です。	2019.3 以降で利用可能です。 Linux のサポートは、バージョン 2020.4 で導入されました。

機能	説明	機能を使用する要件	バージョン
Tableau Content Migration Tool について	Content Migration Tool を使用すると、Tableau Server のプロジェクト間、サイト間、展開間でコンテンツを簡単にコピーまたは移行できます。独立した Tableau Server インストールにあるプロジェクト間 (Tableau Server の開発インスタンスと本番インスタンスの間など)、または単一の Tableau Server インストールにあるプロジェクト間で実行可能です。	• 移動元の Tableau Server (コンテンツの移動元である Server) と移動先の Tableau Server (コンテンツの移動先である Server) の両方には、有効な Advanced Management ライセンスが必要です。 • Content Migration Tool は .NET 4.6.1 をサポートするバージョンの Microsoft Windows (Windows 7 以降、Windows Server 2008R2 以降) にインストールされる必要があります。	バージョン 2019.3 以降で利用可能です。
アクティビティログ	アクティビティログによって、イベントがローカル ハード ドライブの vizportal ログ フォルダーに書き込まれ、さらなる分析と監査が行われます。	Tableau Server には有効な Advanced Management ライセンスが必要です。	バージョン 2022.3 以降で利用可能です。
Tableau	Tableau Server リポジトリを Tableau	外部リポジトリを使用	バージョン

機能	説明	機能を使用する要件	バージョン
Server 外部リポジトリ	Server 外部に展開できます。Tableau Server リポジトリは、すべてのユーザーインタラクション、抽出の更新などに関するデータを格納する PostgreSQL データベースです。	している Tableau Server には Advanced Management ライセンスが必要です。	2019.3 以降で利用可能です。 - AWS は、Tableau Server バージョン 2019.3 以降でサポートされています。 - Azure は Tableau Server バージョン 2020.4 以降でサポートされています。
ノードロールによるワークロード管理	ノードロールを使用すると、Tableau Server インストールで特定のタイプのワークロードが処理される場所を構成できます。ノードロール機能を使用すると、リソースを特定のワークロード(抽出更新、サブスクリプションなど) 専用にしたり、ワークロードに合わせて拡張したりできます。	Tableau Server には有効な Advanced Management ライセンスが必要です。	バージョン 2019.3 以降で利用可能です。

機能	説明	機能を使用する要件	バージョン
Tableau Server のキー管理システム	Tableau Server で抽出暗号化のために AWS とKMS を使用するように構成する追加機能が得られます。	Tableau Server には有効な Advanced Management ライセンスが必要です。	バージョン 2019.3 以降で利用可能です。 - AWS は、Tableau Server バージョン 2019.3 以降でサポートされています。 - Azure は Tableau Server バージョン 2021.1 以降でサポートされています。
Tableau Server 外部ファイル ストア	ネットワークに接続されているストレージ (NAS) をファイル ストアとして使用できます。これにより、ファイル ストアを Tableau Server でローカルに実行する必要がなくなりました。	Tableau Server には有効な Advanced Management ライセンスが必要です。	バージョン 2020.1 以降で利用可能です。
Tableau バックグラウンダー	Tableau Server リソース制限マネージャーは、設定されたリソース制限に対	Tableau Server には有効な Advanced	バージョン 2022.1 以

機能	説明	機能を使用する要件	バージョン
のリソース制限	するバックグラウンダー リソースの使用状況を追跡し、リソース制限が正しく適用されていることを確認します。	Management ライセンスが必要です。	降で利用可能です。
Tableau Server の独立したゲートウェイ	Tableau Server の独立したゲートウェイを使用すると、Tableau Server が管理するリバースプロキシを別のコンピューターにインストールできます。DMZ に独立したゲートウェイをインストールして、サーバーのインストールとすべてのデータのセキュリティを確保します。独立したゲートウェイはさまざまな構成で使用でき、クライアント接続の要求に合わせて拡張できます。	Tableau Server には有効な Advanced Management ライセンスが必要です。	バージョン 2022.1 以降で利用可能です。

Advanced Management プロダクトキーを有効化する

Advanced Management ライセンスは Tableau Server インストールに適用され、ユーザーベースとコアベース両方のインストールに使用できます。

ここでは、Tableau Server インストールで Advanced Management を有効化する方法の概要を説明します。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850

2. **[構成]** タブで **[ライセンス発行]** をクリックします。**[ライセンス認証]** をクリックします。

3. Advanced Management プロダクトキーを入力するか貼り付けて、**[アクティブ化]** をクリックします。

4. **[登録]** ページで登録情報を入力し、**[登録]** をクリックします。
5. 登録が完了したら、プロンプトに従って Tableau Server を再起動します。

> **注:** これが新しい Tableau Server のインストールであり、Tableau Server プロダクトキーを適用する前に Advanced Management キーを適用する場合、エラーが表示されます。とはいえ、上記の手順でインストールを続行し、Tableau Server プロダクトキーを適用してかまいません。

TSM CLI の使用

1. Tableau Server クラスタ内のノードでコマンドプロンプトを管理者として開きます。
2. ライセンスを認証するには、次のコマンドを Advanced Management プロダクトキーと一緒に実行します。

```
tsm licenses activate -k <server-management-add-on-product key>
```

> **注:** これが新しい Tableau Server インストールの場合、コマンドを 2 回実行してください。1 回目は Tableau Server プロダクトキーと一緒に、次に Advanced Management のプロダクトキーと一緒に実行します。

3. 変更を適用し、サーバーを再起動します。

```
tsm pending-changes apply
```

実行可能なユーザー

Advanced Management ライセンスキーを認証できるのは、サーバー管理者だけです。

Tableau Resource Monitoring Tool について

このドキュメントでは、Tableau Resource Monitoring Tool の概要を説明します。

Resource Monitoring Tool とは

Resource Monitoring Tool は、Tableau Server の正常性とパフォーマンスを監視するために使用できるツールです。Tableau Server からデータが収集されるので、その正常性を包括的に確認できます。このツールを使用すると、読み込み時間の遅延、抽出の失敗、その他の重要な問題の原因を特定することができます。Resource Monitoring Tool を使用するには、Advanced Management をサーバーで有効にする必要があります。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください、

Resource Monitoring Tool には、Resource Monitoring Tool サーバー (RMT サーバー) とエージェントという2 つの主要なコンポーネントがあります。

- RMT サーバーは、Tableau Server からのデータが Web インターフェイスを介して照合され、提供される場所です。ここでは、Tableau Server の正常性とパフォーマンスを構成、監視、および分析することもできます。
- エージェントは Tableau クラスタ内の各ノードで実行され、パフォーマンスやアクティビティを監視します。次の図は、Tableau Server ノードとRMT サーバー間の相互作用を示しています。

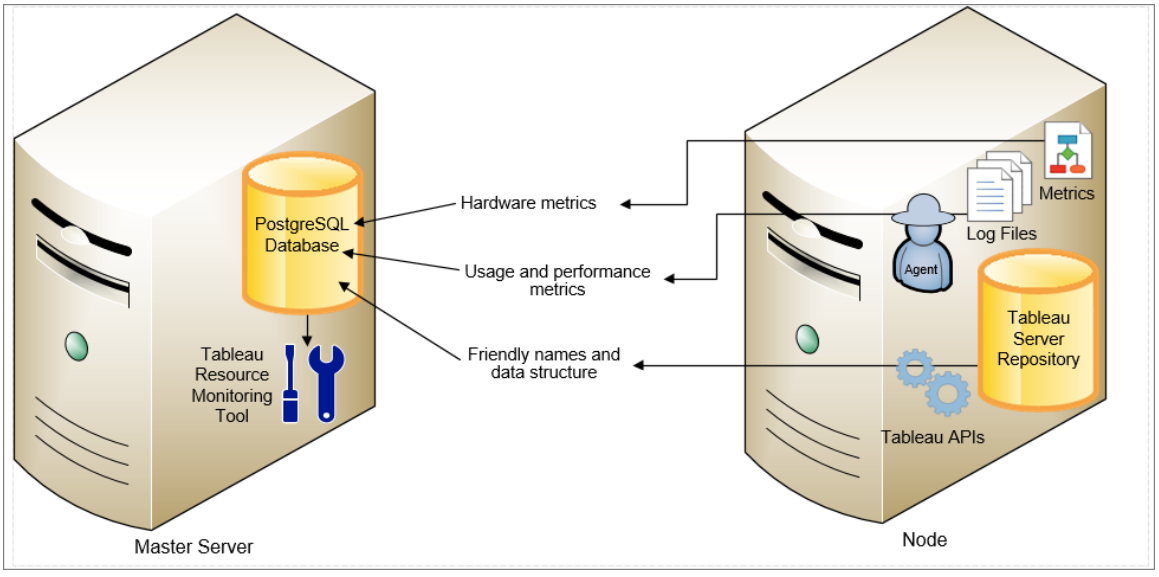


Tableau Resource Monitoring Tool の使用開始

この記事は、Tableau Resource Monitoring Tool の使用を開始するときに役立ちます。Resource Monitoring Tool をインストールする前に必要な準備についての他の記事へのリンク、アップグレードの手順や他の有用なリソースへのリンク記載されています。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

インストール前

Tableau Server との製品互換性

Resource Monitoring Tool をインストールすると、サポートされている Tableau Server のバージョンを監視することができます。

次の表は、Resource Monitoring Tool (RMT) に対応する Tableau Server のバージョンを示しています。

RMT バージョン	**OS サポート**	**Tableau Server バージョン**
2025.1.x	Windows、Linux	2022.3.x、2023.1.x、2023.3.x、2024.2.x、2025.1.x
2024.2.x	Windows、Linux	2022.3.x、2023.1.x、2023.3.x、2024.2.x
2023.3.x	Windows、Linux	2022.3.x、2023.1.x、2023.3.x
2023.1.x	Windows、Linux	2022.3.x、2023.1.x

RMT バージョン	OS サポート	Tableau Server バージョン
2022.3.x	Windows、Linux	2022.3.x

Resource Monitoring Tool サーバー(RMT サーバー) とエージェントの互換性

一般的に、互換性を確保するため、同じバージョンの RMT サーバーとエージェントをインストールすることが推奨されています。RMT サーバーのバージョンと互換性のないバージョンのエージェントを使用している場合、重大なインシデントがログに記録されます。詳細については、「エージェントのインシデント」を参照してください。

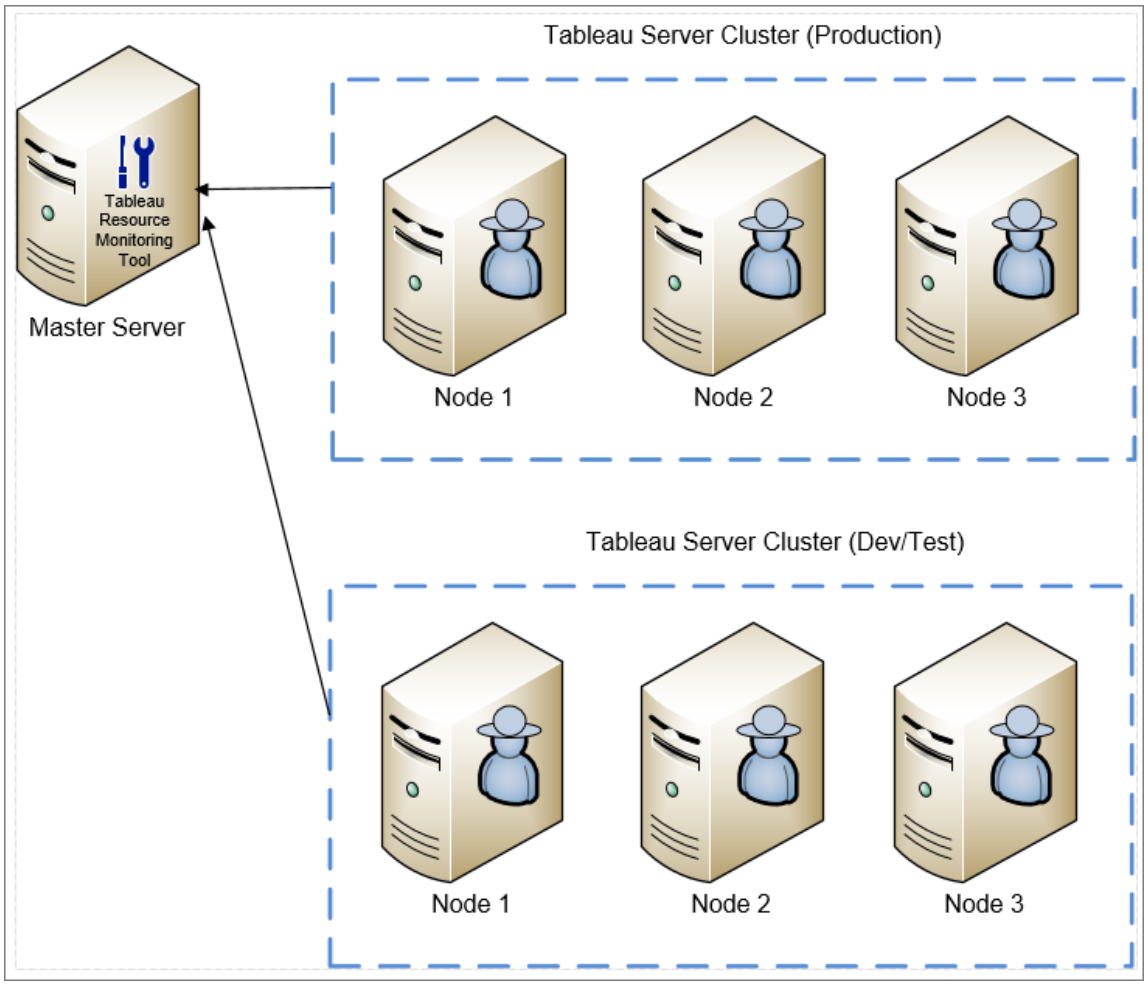
概念と用語

Resource Monitoring Tool に関するトピックで使用される概念と用語を理解します。詳細については、概念を参照してください。

セットアップ アーキテクチャ

Resource Monitoring Tool には、Resource Monitoring Tool サーバー (RMT サーバー) とエージェントという2 つの主要なコンポーネントがあります。RMT サーバーは独自のマシンにインストールする必要があり、エージェントは Tableau Server クラスタの各 ノードにインストールします。セットアップ アーキテクチャの詳細については、Tableau Resource Monitoring Tool のインストールを参照してください。

Resource Monitoring Toolセットアップの概要。次の図に、3 ノードTableau サーバー クラスタを示します。



Resource Monitoring Tool の最小ハードウェア要件と推奨事項

Resource Monitoring Tool は、Tableau Server のインストール時に収集されたモニタリング データの処理、集計、保存、および Web インターフェイスにすべて対応します。稼働環境において、Resource Monitoring Tool をインストールするマシンは最小ハードウェア推奨事項を満たすか上回る必要があります。詳細については、Tableau Resource Monitoring Tool の最小ハードウェア要件と推奨事項を参照してください。

Resource Monitoring Tool のインストール前チェックリスト

Resource Monitoring Tool をインストールする前に、これらのアクション項目を確認して必要な手順 (インストール前チェックリスト- Tableau Resource Monitoring Tool) を完了しておきます。

トラブルシューティング

サポートに問い合わせる前に、よくある質問の回答をTableau Resource Monitoring Tool の問題のトラブルシューティングで確認してください。

概念

このドキュメントでは、一部の Tableau Resource Monitoring Tool で中核となる概念について簡単に説明し、よく使用されるいくつかの用語を定義します。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。ただし、CLI のコマンドとオプションおよびその他のインスタンスで引き続きこの用語が表示される場合があります。詳細については、「Tableau ヘルプについて」を参照してください。

エージェント

Resource Monitoring Tool エージェントは Tableau の各サーバーでバックグラウンドサービスとして実行します。エージェントは、監視する Tableau Server インストールのすべてのノードにインストールする必要があります。

エージェントは、Tableau のアクティビティとインストールされているサーバーの全般的なパフォーマンスを監視する簡単なプロセスです。エージェントはこれらの詳細を Resource Monitoring Tool サーバーに報告します。

Resource Monitoring Tool サーバー (RMT サーバー)

Resource Monitoring Tool サーバーは、ユーザーが操作する Web アプリケーションをホストします。また、エージェントからのデータを照合および監視するバックグラウンド処理の多くを実行します。このコンポーネントを指す際には、**RMT サーバー**という用語を使用します。

環境

環境は、単一ノードの Tableau Server のインストール、または Tableau Server クラスタを指して使用される用語です。RMT サーバー Web インターフェイスでは、Tableau Server の各ノードは**サーバー**と呼ばれます。

Resource Monitoring Tool では、複数の Tableau Server のインストールを監視できます。たとえば、本番環境とステージング環境の Tableau Server を使用しているとします。Resource Monitoring Tool で 2 つの環境 (*本番*と*ステージング*) を作成し、これら両方の Tableau Server クラスタを監視できます。

環境ステータス

*環境ステータス*は、現在の Tableau 環境の状態を示します。ステータスは、インシデントの数とタイプ、および Tableau のさまざまなプロセスのステータスによって決まります。以下のセクションでは、これらの用語の各ステータスが意味する内容について説明します。

OK

警告インシデントや重要インシデントが現時点で発生しておらず、失敗したプロセスもありません。

警告

次のいずれかが該当する場合、環境は**警告**状態になります。

- 現時点で少なくとも 1 つの警告インシデントが発生している。
- プロセスが**ライセンスなし**である。
- プロセスが**オフライン**である。

重要

次のいずれかが該当する場合、環境は**重要**状態になります。

- 現時点で少なくとも 1 つの重要インシデントが発生している。
- プライマリゲートウェイが**オフライン**である。
- プロセスのすべてのインスタンスが**ライセンスなし**である。
- プロセスのすべてのインスタンスが**オフライン**である。

サーバー

サーバーは、Tableau Server インストールの各ノードに使用される用語です。Resource Monitoring Tool エージェントをインストールしているすべてのマシンがサーバーと見なされます。

インストール前チェックリスト - Tableau Resource Monitoring Tool

次のトピックでは、Tableau Resource Monitoring Tool のインストールの前提条件について説明します。また、使用されるサービス アカウントに必要なオペレーティング システムの権限についても説明します。

マシン、ネットワーク、アカウントの要件

> **注:** Resource Monitoring Tool (RMT) のサーバーとエージェントは連邦情報処理標準 (FIPS) 140-2 に準拠していないため、FIPS モードが有効になっているオペレーティング システムでは正しく機能しません。

Tableau Resource Monitoring Tool をインストールする前に、次の前提条件を完了しておきます。

1. Resource Monitoring Tool を実行するすべてのコンピューターにおいて、管理者権限があることを確認します。これには、RMT サーバーを実行するコンピューターと、Resource Monitoring Tool を使用して監視を行うインストール内のすべての Tableau Server ノードが含まれます。
2. RMT サーバーを実行しているコンピューターでポート 5672 をインバウンド TCP トラフィックに開きます。

Resource Monitoring Tool エージェントアプリケーションがポート5672経由で収集したデータを送信するため、RMT サーバーがインストールされているマシンでは、そのポートでインバウンドTCP トラフィックを許可する必要があります。

3. 次のポートを開きます。

   - RMT サーバーを実行しているコンピューターでポート80 をインバウンドTCP トラフィックに開きます。

     これは、RMT サーバーがWeb インターフェイスをホストするポートです。このポートを使用するサードパーティ製のサービスが既に存在する場合は、インストール時にポート番号を変更できます。RMT サーバー以外からWeb インターフェイスにアクセスできるようにするには、このポートを開放する必要があります。

   - Resource Monitoring Tool RMT サーバーを実行しているコンピューターでポート443 をインバウンドTCP トラフィックに開きます。

     このポートは、RMT サーバーへの接続をテストするときにエージェントから接続されるポートです。

   - すべてのエージェントノードで、ポート5672 と5671 をアウトバウンドTCP トラフィックに開放します。

     エージェントアプリケーションは、このポートを経由してRMT サーバーに収集するデータを送信します。

   - すべてのエージェントPC で、アウトバウンドTCP トラフィックにポート80 を開きます。

     エージェントは、RMT サーバーのWeb インターフェイスへ定期的に要求を行います。

   - すべてのエージェントPC で、アウトバウンドTCP トラフィックにポート443 を開きます。

     このポートは、エージェントがRMT サーバーへの接続をテストするポートです。

ポートと通信の詳細については、Tableau Resource Monitoring Tool の通信ポートを参照してください。

4. すべてのエージェントコンピューターが RMT サーバーと通信できることを確認します。

   RMT サーバーがホストする Web インターフェイスは、セットアッププロセス中に特定のホスト名または IP アドレスにバインドされます。特定のホスト名に送信される要求にのみ応答します。たとえば、RMT サーバーが http://RMTServer/ でホストされるように構成されていても、エージェントは http://myrmtserver.myserver.com/ を介してのみ RMT サーバーに接続できる場合、要求は完了しません。

5. Tableau Server 管理者アカウントの認証資格情報を収集します。

   Resource Monitoring Tool は、Tableau Server の Web API を使用してコンテンツに関する情報を収集します。そのため、Tableau Server の管理者であるユーザーがすべてのサイトやプロジェクトなどにアクセスできる認証資格情報が必要です。Resource Monitoring Tool 専用の認証資格情報を作成することをお勧めしますが、Tableau Server 管理者になることも可能です。

   注: RMT は SAML をサポートしていませんが、SAML SSO を使用する Server インストールと連携することができます。これを行うには、Tableau Server 用に構成されたローカル アイデンティティストアが必要です。また、RMT で使用される管理者アカウントのパスワードをローカルで設定する必要があります。

6. Tableau Server リポジトリデータベースへのアクセスを有効にします。**これは、バージョン 2022.3 以降必須となります。**

   Resource Monitoring Tool は、パフォーマンス上の理由からリポジトリデータベースに直接アクセスします。これが動作するには "読み取り専用" データベース ユーザーのパスワード セットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。

7. Resource Monitoring Tool とTableau Server リポジトリ間の SSL 接続を使用する場合は、Tableau Server が内部 Postgres 接続で SSL を使用するように構成されていることを

確認します。詳細については、内部 Postgres 通信用に SSL を設定するを参照してください。

Resource Monitoring Tool では、SSL 接続で証明書ファイル (`server.crt`) または指紋を使用できます。Postgres データベースの証明書ファイルは次の場所にインストールされます。

```
/var/opt/tableau/tableau_server/data/tabsvc/config/pgsql_
<version>/security
```

証明書ファイル (`server.crt`) を使用する場合は、Tableau Server で生成された内部 Postgres SSL 接続用の証明書ファイルを、RMT サーバーをインストールするマシンにコピーします。オペレーティング システムが証明書を信頼していることを確認します。

8. Tableau Server で REST API が有効になっていることを確認します (これが既定です)。有効かどうかを確認するには、`tsm configuration get -k api.server.enabled` コマンドを使用します。戻り値 `true` は、REST API が有効であることを意味します。REST API を有効にするには、`tsm configuration set` コマンドを使用します。詳細については、api.server.enabledを参照してください。

9. Tableau Server のログのサイズを確認します。エージェント サーバーをインストールして構成すると、Resource Monitoring Tool は Tableau Server ログの関連する履歴データを処理し、データを表示します。履歴のログ データが大量にある場合は、処理に時間がかかる場合があります。これにより、サーバーでの新しいイベントの処理が遅れる可能性があります。

   遅延が懸念となり、履歴情報がなくても問題ない場合には、以下の手順で既存ファイルをクリーン アップできます。

   - 不必要なファイルの削除 を選択し、ログ ファイルを削除する前に ログ ファイルのスナップショット (ログのアーカイブ) を検討します。

> **注:** CPU 使用率やメモリ使用量などのパフォーマンス データは履歴ログ データを使用して収集されず、エージェントのインストールと構成が完了した後に収集されるため、履歴データをクリーン アップしてもパフォーマンス データへの影響はありません。

10. (オプション) RMT サーバーがアクセスできる SMTP サーバーの接続情報を収集します。

    メール通知が必要な場合は、メールの送信に使用できるサーバー名、ユーザー名とパスワード(存在する場合)、SMTP サーバーのポート番号を指定した RMT サーバーと TLS バージョンを提供する必要があります。現在、TLS バージョン 1.2 が必要ですが、TLS バージョン 1.3 もサポートされています。TLS バージョン 1.3 を使用する場合は、RMT サーバーがインストールされるマシンに Open SSL 1.1.1f 以上がインストールされていることを確認してください。詳細については、メール通知を参照してください。メール通知

11. (オプション) 既定以外のアカウントに対するオペレーティング システムのサービス アカウント権限を確認します。

    組織に特定のセキュリティ要件があり、既に定義されているシステム アカウントまたはユーザーを使用する必要がある場合は、確認が必要です。

    既定の権限の詳細については、「既定のインストール パーミッション - Tableau Resource Monitoring Tool」を参照してください。

実行可能なユーザー

上記のすべてのステップを実行するには、Resource Monitoring Tool をインストールするマシンの管理者であり、かつ、Tableau Server 管理者である必要があります。

Tableau Resource Monitoring Tool の最小ハードウェア要件と推奨事項

Tableau Resource Monitoring Tool は、Tableau Server のインストール時に収集されたモニタリング データの処理、集計、保存、および Web インターフェイスにすべて対応します。

RMT サーバーの最小ハードウェア推奨事項

稼働環境において、Resource Monitoring Tool をインストールするマシンは以下のハードウェア推奨事項を満たすか上回っている必要があります。

- 8 個の物理 CPU コア (16 個の vCPU)
- CPU は SSE4.2 および POPCNT 命令セットをサポートする必要がある
- 64GB RAM
- RMT サーバー用の 500 GB のディスク空き容量
- パフォーマンス用の SSD ドライブまたは同等のもの

上記で指定された最小ハードウェアに一致する RMT サーバーのインストールの場合、Resource Monitoring Tool は、1 時間あたり最大 10,000 ビューを処理する Tableau Server 展開のパフォーマンス レポートを提供できます。

1 時間あたりのビューの負荷が大きい展開、またはバックグラウンド ジョブの数が非常に多い展開では、データの遅延が発生する可能性があります。このような場合は、ハードウェアのアップグレードが必要になることがあります。

Resource Monitoring Toolエージェント- リソース使用率

Resource Monitoring Tool エージェントは、オペレーティング システムとTableau Server のプロセスとログ ファイルを監視し、パフォーマンス メトリクスとTableau のログ データを Resource Monitoring Tool に送信して、ほぼリアルタイムの処理を行います。エージェントは、Tableau Server を実行しているすべてのマシンにインストールする必要があります。

Tableau Server の最小ハードウェア要件と推奨事項に一致する Tableau Server のインストールでは、次のようになります。

- 通常、エージェントは平均して 0 ～ 5% の CPU を使用しており、それを超えるスパイクはまれにしか発生しません。パフォーマンスの目標は CPU 使用率が平均 10% 以下であることです。
- エージェントは通常 RAM 200 MB 以下を使用しますが、Tableau Server のアクティビティ負荷が高い期間にまれにスパイクがあります。
- エージェントがインストールされているマシンには、少なくとも 10 GB の空きディスク領域があることが推奨されています。

クラウド環境でのインストール

Resource Monitoring Tool は、Tableau Server で使用する可能性がある任意のクラウド環境の仮想マシンにインストールできます。RMT サーバーがインストールされている仮想マシンは、上記と同じ最小ハードウェア要件を満たしている必要があります。

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のパーミッションがすべて必要です。

- Resource Monitoring Tool をインストールするマシンの管理者
- Tableau Server 管理者。
- Resource Monitoring Tool 管理者。

既定のインストール パーミッション - Tableau Resource Monitoring Tool

次のトピックでは、インストール中にさまざまなシステム アカウントまたはグループに設定される既定のパーミッションについて説明します。お客様の環境または組織では既定以外のアカウントを使用する必要がある場合、Resource Monitoring Tool (RMT) が適切に機能するために必要なパーミッションを判断するのに役立ちます。

> **注:** Resource Monitoring Tool (RMT) のサーバーとエージェントは連邦情報処理標準 (FIPS) 140-2 に準拠していないため、FIPS モードが有効になっているオペレーティング システムでは正しく機能しません。

Windows インストール

既定では、Tableau Resource Monitoring Tool はインストール時に次のアカウントを作成します。

- NT SERVICE\TableauResourceMonitoringTool
- NT SERVICE\TableauResourceMonitoringToolPostgreSQL
- NT SERVICE\TableauResourceMonitoringToolRabbitMQ

既定のアカウントの代わりに他のアカウントを使用する要望や必要がある場合は、インストール後に `rmt-admin` コマンド ライン ユーティリティを使用してそれらを指定できます。詳細については、rmtadmin コマンド ライン ユーティリティを参照してください。

これらの既定のサービス アカウントで使用されるパーミッションと、最小限必要なパーミッションを以下に示します。サービス アカウントは、最小限のパーミッションまたは既定のパーミッションと一致する必要があります。

サービス アカウント	リソース	既定のパーミッション	最小限必要なパーミッション
Tableau Resource Monitoring Tool マスター			
	アプリケーション ファイル `Tableau Resource Monitoring Tool\master`	フル コントロール: 読み取り、書き込み、実行、変更	フル コントロール
	ログ ディレクトリとファイル `Tableau Resource Monitoring Tool\master\logs`	フル コントロール: 読み取り、書き込み、実行、変更	読み取り、書き込み
	設定ディレクトリ `Tableau Resource Monitoring Tool\master\config`	フル コントロール: 読み取り、書き込み、実	フル コントロール

サービス アカウント	リソース	既定のパーミッション	最小限必要なパーミッション
		行、変更	
Tableau Resource Monitoring Tool PostgreSQL			
	アプリケーション ディレクトリ `Tableau Resource Monitoring Tool\prerequisites\postgresql<nn>`	フル コントロール: 読み取り、書き込み、実行、変更	フル コントロール
	データ ディレクトリ `Tableau Resource Monitoring Tool\data\postgresql<nn>`	フル コントロール: 読み取り、書き込み、実行、変更	フル コントロール
	ログ ディレクトリ `Tableau\Tableau Resource Monitoring Tool\master\logs\pgsql`	フル コントロール: 読み取り、書き込み、実行、変更	読み取り、書き込み
Tableau Resource Monitoring Tool RabbitMQ			
	データ ディレクトリ	フル コント	フル コ

サービス アカウント	リソース	既定のパーミッション	最小限必要なパーミッション
	Tableau Resource Monitoring Tool\data\rabbitmq	ロール: 読み取り、書き込み、実行、変更	ントロール
	アプリケーション ファイル Tableau Resource Monitoring Tool\prerequisites\rabbitmq Tableau Resource Monitoring Tool\prerequisites\erlang	フル コントロール: 読み取り、書き込み、実行、変更	フル コントロール
	ログ ファイル Tableau Resource Monitoring Tool\master\logs\rabbitmq	フル コントロール: 読み取り、書き込み、実行、変更	読み取り、書き込み
Tableau Resource Monitoring Tool エージェント			
	アプリケーション ファイル Tableau Resource Monitoring Tool\agent	フル コントロール: 読み取り、書き込み、実行、変更	フル コントロール
	ログ ディレクトリとファイル	フル コント	読み

サービス アカウント	リソース	既定のパーミッション	最小限必要なパーミッション
	`Tableau Resource Monitoring Tool\agent\logs`	ロール: 読み取り、書き込み、実行、変更	取り
	設定ディレクトリ `Tableau Resource Monitoring Tool\agent\config`	フル コントロール: 読み取り、書き込み、実行、変更	フル コントロール

Linux インストール

既定では、Tableau Resource Monitoring Tool はインストール時に特定のアカウントを作成します。既定の代わりにグループやユーザーを使用する要望または必要がある場合は、インストール後に `rmt-admin` コマンド ライン ユーティリティを使用してそれらを指定できます。詳細については、rmtadmin コマンド ライン ユーティリティを参照してください。

これらのサービス アカウントで使用される既定のパーミッションを以下に示します。パーミッションは次のパーミッションと一致する必要があります。

サービス アカウント	リソース	パーミッション
Tableau Resource Monitoring Tool マスター		

サービス アカウント	リソース	パーミッション
	アプリケーション ファイル /var/opt/tableau/tabrmt/master	所有者 - フル パーミッション (読み取り、書き込み、実行) グループ - 読み取り、書き込み、実行 その他 - 読み取り、

サービス アカウント	リソース	パーミッション
		実行
	ログ ディレクトリとファイル /var/opt/tableau/tabrmt/master/logs	所有者 - フル パーミッション (読み取り、書き込み、実行) グループ - 読み取り、書き込み、実行 その他 - 読み

サービス アカウント	リソース	パーミッション
		取り、実行
	設定ディレクトリ /var/opt/tableau/tabrmt/master/config	所有者 - フルパーミッション (読み取り、書き込み、実行) グループ - 読み取り、書き込み、実行 その

サービス アカウント	リソース	パーミッション
		他 - 読み取り、実行
Tableau Resource Monitoring Tool PostgreSQL		
	アプリケーション ディレクトリ `/opt/tableau/tabrmt/prerequisites/postgresql13`	所有者 - フルパーミッション (読み取り、書き込み、実行) グループ - 読み

サービス アカウント	リソース	パーミッション
		取り、実行 その他 - 読み取り、実行
	サービス ディレクトリ `/var/opt/tableau/tabrmt/data/postgresql13/pg_*`	所有者 - フルパーミッション (読み取り、書き込み、実行) グループ - なし その

サービス アカウント	リソース	パーミッション
		他 - なし
	データディレクトリ /var/opt/tableau/tabrmt/data/postgresql13/	所有者 - フルパーミッション (読み取り、書き込み、実行) グループ - 読み取り、書き込み、実行 その他 -

サービス アカウント	リソース	パーミッション
		読み取り、実行
	ログ ディレクトリ `/var/opt/tableau/tabrmt/master/logs/pgsql`	所有者 - フル パーミッション (読み取り、書き込み、実行) グループ - 読み取り、書き込み、実行

サービス アカウント	リソース	パーミッション
		その他 - 実行
	上記以外の PostgreSQL ディレクトリ下のディレクトリ `/var/opt/tableau/tabrmt/data/postgresql13/<not mentioned above>`	所有者 - 読み取り、書き込み グループ - なし その他 - なし
	`/var/opt/tableau/tabrmt/data/postgresql13/base` `/var/opt/tableau/tabrmt/data/postgresql13/global`	所有者 - フルパーミッション (読み取り、書き込み、

サービス アカウント	リソース	パーミッション
		実行) グループ- なし その他 - なし
	/var/opt/tableau/tabrmt/data/postgresql13/certific-ates	所有者 - フルパーミッション (読み取り、書き込み、実行) グループ- フルパー

サービス アカウント	リソース	パーミッション
		ミッション (読み取り、書き込み、実行) その他 - 実行
Tableau Resource Monitoring Tool RabbitMQ		
	データディレクトリ `/var/opt/tableau/tabrmt/data/rabbitmq`	所有者 - フルパーミッション (読み取り、

サービス アカウント	リソース	パーミッション
		書き込み、実行) グループ - 読み取り、書き込み、実行 その他 - 読み取り、実行
	アプリケーション ファイル /var/opt/tableau/tabrmt/rabbitmq/prerequisites/rabbitmq /var/opt/tableau/tabrmt/rabbitmq/prerequisites/erlang	所有者 - フルパーミッション(読

サービス アカウント	リソース	パーミッション
		み取り、書き込み、実行) グループ - 読み取り、実行 その他 - 読み取り、実行
	ログ ファイル /var/opt/tableau/tabrmt/master/logs/rabbitmq	所有者 - フルパーミッション(読み取り、

サービス アカウント	リソース	パーミッション
		書き込み、実行) グループ - 読み取り、書き込み、実行 その他 - 実行
	その他のディレクトリ(/rabbitmq ディレクトリ) /var/opt/tableau/tabrmt/data/rabbitmq/certificates /var/opt/tableau/tabrmt/data/rabbitmq/mnesia	所有者 - フルパーミッション(読み取り、書き込

サービス アカウント	リソース	パーミッション
		み、実行) グループ - 読み取り、書き込み、実行 その他 - 実行
	/var/opt/tableau/tabrmt/data/rabbitmq/<other than mentioned>	所有者 - 読み取り、実行 グループ - 読み取

サービス アカウント	リソース	パーミッション
		り、実行 その他 - なし
Tableau Resource Monitoring Tool エージェント		
	アプリケーション ファイル /var/opt/tableau/tabrmt/agent	所有者 - フル パーミッション (読み取り、書き込み、実行) グループ -

サービス アカウント	リソース	パーミッション
		読み取り、書き込み、実行 その他 - 読み取り、実行
	ログ ディレクトリとファイル /var/opt/tableau/tabrmt/agent/logs	所有者 - フル パーミッション (読み取り、書き込み、実行) グ

サービス アカウント	リソース	パーミッション
		ループ - 読み取り、書き込み、実行 その他 - 読み取り、実行
	設定ディレクトリ `/var/opt/tableau/tabrmt/agent/config`	所有者 - フルパーミッション (読み取り、書き込み、実行)

サービス アカウント	リソース	パーミッション
		グループ - 読み取り、書き込み、実行 その他 - 読み取り、実行

実行可能なユーザー

上記のすべてのステップを実行するには、Resource Monitoring Tool をインストールするマシンの管理者であり、かつ、Tableau Server 管理者である必要があります。

Resource Monitoring Tool (RMT) サービス

Resource Monitoring Tool (リソース監視ツール) には、Resource Monitoring Tool マスター (RMT サーバー) とRMT エージェントという2 つの主要コンポーネントがあります。

- RMT マスターは、Tableau Server からのデータを照合し、Web インターフェイスを介して提供する場所です。ここでは、RMT を設定し、Tableau Server の正常性とパフォーマンスを監

視および分析することもできます。

- エージェントは Tableau サーバー クラスター内の各ノードで実行され、パフォーマンスとアクティビティを監視します。

このトピックでは、RMT とともにインストールされるサービスについて説明します。以下の表には、RMT サーバー マシンにインストールされるサービスと、RMT エージェント マシンにインストールされているサービスの一覧です。

RMT サーバーのサービス			
名前	**目的**	**ログ ファイルの場所**	**注**
		Windows: `C:\Program Files\Tableau\Tableau Resource Monitoring Tool\master\logs\` Linux: `/var/opt/tableau/tabrmt/master/logs` **注:** ログ ファイルの詳細については、Tableau Resource Monitoring Tool ログ ファイルを参照してください。	
ホスト	インストール中に使用され、他のプロセスが実行されていることを確認するためにも使用され	Windows: `host\YYYYMMDD.log` Linux: `host/YYYMMDD.log`	役割: • プロセス ウォッチャー: 他のプロセスが実行されていることを確認

	ます。		します。 • 設定ファイル ウォッチャー: 設定ファイルの変更を検出し、それに応答します。
バックグラウンダー	エージェントから送信された情報を処理し、RMT が収集した postgres データを維持します。管理者コマンドライン ユーティリティも提供します。処理された情報に基づいてインシデントと通知を生	Windows: `background\YYYYMMDD-pts.log` Linux: `background/YYYYMMDD-pts.log`	役割: • rmtadmin: 管理者向けにコマンドライン機能を提供するコマンドライン ユーティリティ。 • エージェント データ処理: さまざまなエージェントからデータを収集して処理します

	成します。		• TS ステータス ポーリング: Tableau Server (TS) のステータスを定期的にポーリングし、システムがアクティブで正常であることを確認します。 • インシデント分析: エラー、失敗、さまざまなメジャーのしきい値への到達などのインシデントを分析します。 • データク

			リーンアップ: 古いデータのクリーンアップを担当します。 • アドホックグラフ情報収集: 監視のためにオンデマンドでデータを収集します。 • 通知: アラートやインシデントの通知の送信を管理します。 • データアーカイブ (無効): リソースを解放するため、またはコンプラ

			イアンスのために古いデータをアーカイブします。現在無効になっています。
ディレクター	一種の調整プロセスであり、その主な役割の 1 つは、Tableau Server にライセンスのステータスを照会し、システム全体のステータスを設定することです。	Windows: `director\YYYYMMDD-pts.log` Linux: `director/YYYYMMDD-pts.log`	役割: • クラスターステータス: サーバー クラスターの健全性とステータスを監視し、システム全体のステータスを設定します。 • ライセンス: Tableau Server にライセンスのステータスを照会

			します。 • グラフスクレイピング: 監視およびレポート用のデータを抽出します。サーバーデータベースにアクセスできる可能性があります。 • データベースのアップグレード: データベースのスキーマまたはバージョンの更新を処理します • 日次集計処理: 報告や分析のため

			のデータの集計など、日次処理タスクを実行します。
Web	Web サーバーを実行します。ユーザー インターフェイスを提供します。情報収集のためにエージェントを登録します	Windows: `web\YYYYMMDD-pts.log` Linux: `web/YYYYMMDD-pts.log`	役割: • エージェント登録: Tableau サーバー上でデータを監視または収集するエージェントの登録を管理します。 • Config UI: システムのさまざまな設定とパラメーターを構成するためのユーザー インターフェイ

			スを提供します。 • パフォーマンス/アクティビティ/コンテンツ UI: ユーザーがシステム内のパフォーマンス指標、アクティビティログ、コンテンツ関連データを表示するための UI を提供します。
PostgreSQL	ローカルデータリポジトリ	Windows: `pgsql\*.log and *.csv` Linux: `pgsql/*.log and *.csv`	
RabbitMQ	メッセージブローカー	Windows: `\master\logs\rabbitmq\*.log` Linux: `master/logs/*.log`	

RMT エージェントのサービス			
名前	目的	ログ ファイルの場所 Windows: `C:\Program Files\Tableau\Tableau Resource Monitoring Tool\master\logs\` Linux: `/var/opt/tableau/tabrmt/master/logs` **注:** ログ ファイルの詳細については、Tableau Resource Monitoring Tool ログ ファイルを参照してください。	注
ホスト	インストール中に使用され、他のプロセスが実行されていることを確認するためにも使用されます。	Windows: `host\YYYYMMDD.log` Linux: `host/YYYYMMDD.log`	役割: • プロセス ウォッチャー: 他のプロセスが実行されていることを確認します。 • 設定ファイル ウォッチャー: 設定ファイルの変更を検出し、それに応答します。

バックグラウンダー	エージェントから送信された情報を処理し、RMT が収集した postgres データを維持します。管理者コマンドライン ユーティリティも提供します。処理された情報に基づいてインシデントと通知を生成します。	Windows: `background\YYYYMMDD-pts.log` Linux: `background/YYYMMDD-pts.log`	役割: • rmtadmin: 管理者向けにコマンドライン機能を提供するコマンド ライン ユーティリティ。 • ログ ウォッチャー: Tableau サーバーから送信された情報をログから処理します。
Web	Web サーバーを実行します。 データのパブリッシュのためにマスターへのエージェントの登録を実行します。	Windows: `web\YYYYMMDD-pts.log` Linux: `web/YYYMMDD-pts.log`	

Tableau Resource Monitoring Tool のインストール

Tableau Resource Monitoring Tool には、RMT サーバーとエージェントという2 つのコンポーネントがあります。

Resource Monitoring Tool の必須コンポーネントとRMT サーバーは、必要なリソースを確保し、他のプログラムやソフトウェアとの間でマシン リソースの競合がないことを保証するため、専用のハードウェアにインストールする必要があります。これらは、例外的な場合 (たとえば、本番以外の環境での限られたデモ目的の場合) を除いて Tableau Server PC にはインストールしないでください。

エージェントは、監視する必要がある Tableau Server のすべてのノードにインストールされます。

Resource Monitoring Tool をインストールするためのインストーラーは Advanced Management のダウンロード サイトにあります。**Tabrmt** で始まるファイルをすべてダウンロードします。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

インストールのバージョン履歴

このセクションでは、インストール プロセスに関する重要な変更点を取り上げます。このセクションをよく読み、現在のバージョンとアップグレード先のバージョンの変更内容を理解してください。

- バージョン **2022.3** 以降、Resource Monitoring Tool で使用する RabbitMQ メッセージング サービスとPostgreSQL データベースをクラウドプラットフォームで構成するオプションを利用

できます。現在、RabbitMQ には AWS AMQ、PostgreSQL データベースには Amazon RDS がサポートされています。今回の追加により、リポジトリとメッセージ キュー サービスを RMT サーバーとともに同じマシンにインストールするか、AWS プラットフォームに個別にインストールするかを選択できるようになりました。

- バージョン **2020.4** 以降、Resource Monitoring Tool を Linux マシンにインストールできるようになりました。以前は、Resource Monitoring Tool インストールのみが Windows マシンでのみサポートされていました。
- **2020.2** 以降、1つのインストーラーで前提条件と RMT Server がインストールされます。必要なインストーラーは、1 つは前提条件を含む RMT サーバー用とエージェントのインストール用の 2 つのインストーラーだけです。

Resource Monitoring Tool をインストールする前に

- 次の要件と手順を確認してください。
  - Tableau Resource Monitoring Tool の最小ハードウェア要件と推奨事項
  - インストール前チェックリスト - Tableau Resource Monitoring Tool
  - 次の OS に対応しています。
    - AlmaLinux:

      バージョン 2023.3.1 以降: AlmaLinux 8.x、AlmaLinux 9.x
    - Amazon Linux:

      バージョン 2023.3.1 以降: Amazon Linux 2、Amazon Linux 2023

      バージョン 2023.3.0 以前: Amazon Linux 2
    - CentOS:

      バージョン 2023.3.1 以降: Stream 8.x,、Stream 9.x
    - Oracle:

      バージョン 2023.3.1 以降: Oracle 8.x、Oracle 9.x
    - Red Hat Enterprise Linux (RHEL):

バージョン 2023.3.1 以降: 9.x、8.3+

バージョン 2023.3.0: 9.x、8.3+

バージョン 2023.1.x 以前: 8.3+

- Rocky Linux:

  バージョン 2023.3.1 以降: Rocky Linux 8.x、Rocky Linux 9.x

- Ubuntu:

  バージョン 2023.1.1: 最新バージョンの Ubuntu 16.04 LTS、18.04 LTS (17.04 を除く)、20.04 LTS、22.04

  バージョン 2023.1.0: 最新バージョンの Ubuntu 16.04 LTS、18.04 LTS (17.04 を除く)、20.04 LTS

  バージョン 2022.3.3 以前: 最新バージョンの Ubuntu 16.04 LTS、18.04 LTS (17.04 を除く)

Linux ディストリビューションに関するその他の注意事項:

- Red Hat Enterprise Linux (RHEL)、CentOS、、Amazon Linux ディストリビューションは、このドキュメントでは総称して RHEL ライクと呼ばれます。
- Tableau Server ではプロセス管理に systemd が必要なことから CentOS および Ubuntu の以前のバージョンはサポートされていません。
- ファイルの接尾辞に「.deb」が付いたインストーラーのバージョンは、Ubuntu ディストリビューションでインストールされます。

  2022 年 7 月の時点で、Debian ディストリビューションはサポートされなくなりました。詳細については、Tableau コミュニティのこの投稿を参照してください。

- カスタム カーネルはサポートされていません。

- 監視予定の Tableau Server インストールが Advanced Management ライセンスにより認証されていることを確認します。Resource Monitoring Tool を使用するには、有効な Advanced Management ライセンスが必要です。

- Tableau Server のログのサイズを確認します。エージェントをインストールして構成すると、Resource Monitoring Tool は Tableau Server ログの関連する履歴データを処理し、データを表示します。履歴ログ データが大量の場合は、情報処理にしばらく時間がかかる場合があるため、Server で新しいイベントを処理するのに遅延が発生する場合があります。

  遅延が懸念となり、履歴情報がなくても問題ない場合には、以下の手順で既存ファイルをクリーン アップできます。

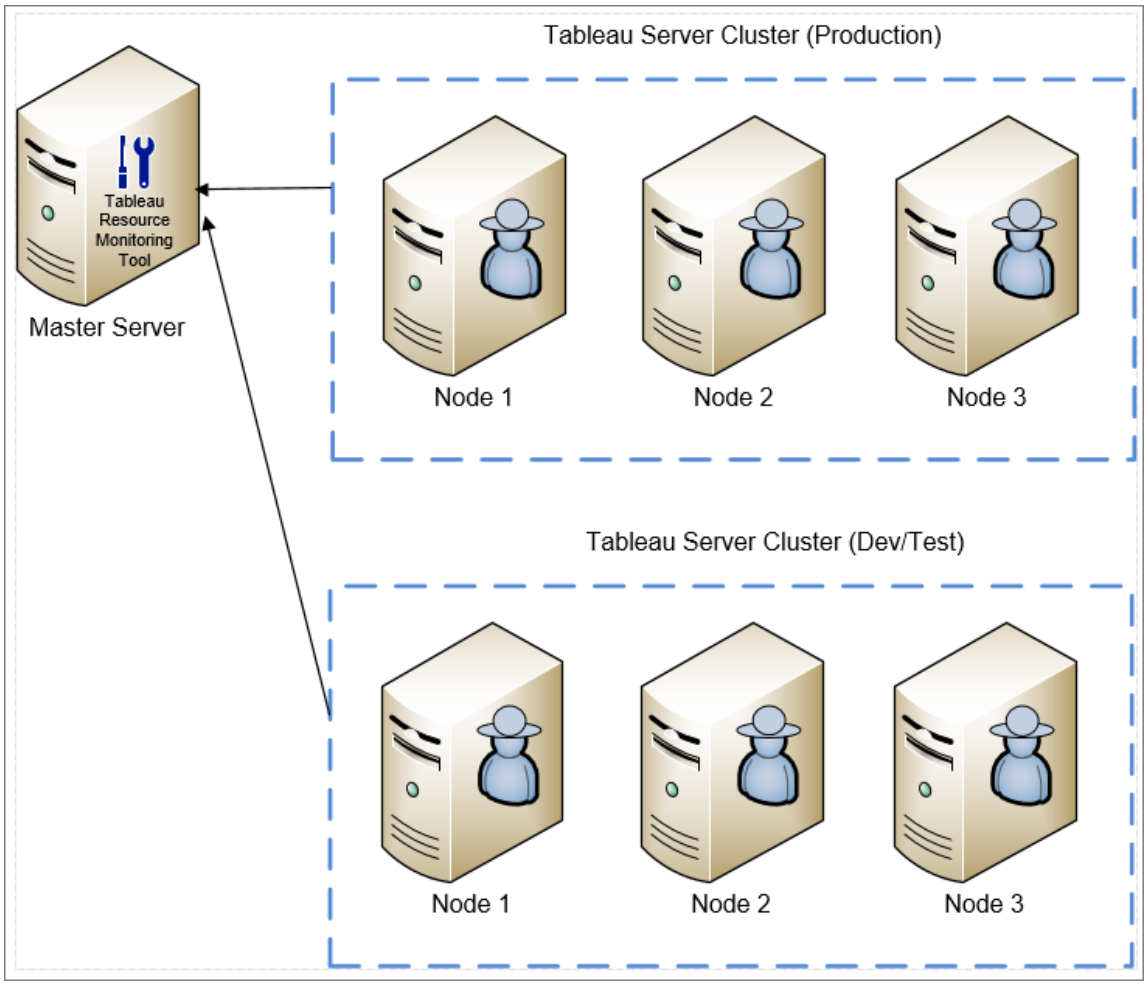
  - 不必要なファイルの削除 を選択し、ログ ファイルを削除する前に ログ ファイルのスナップショット (ログのアーカイブ) を検討します。

  > **注:** CPU 使用率やメモリ使用量などのパフォーマンス データは履歴ログ データを使用して収集されず、エージェントのインストールと構成が完了した後に収集されるため、履歴データをクリーン アップしてもパフォーマンス データへの影響はありません。

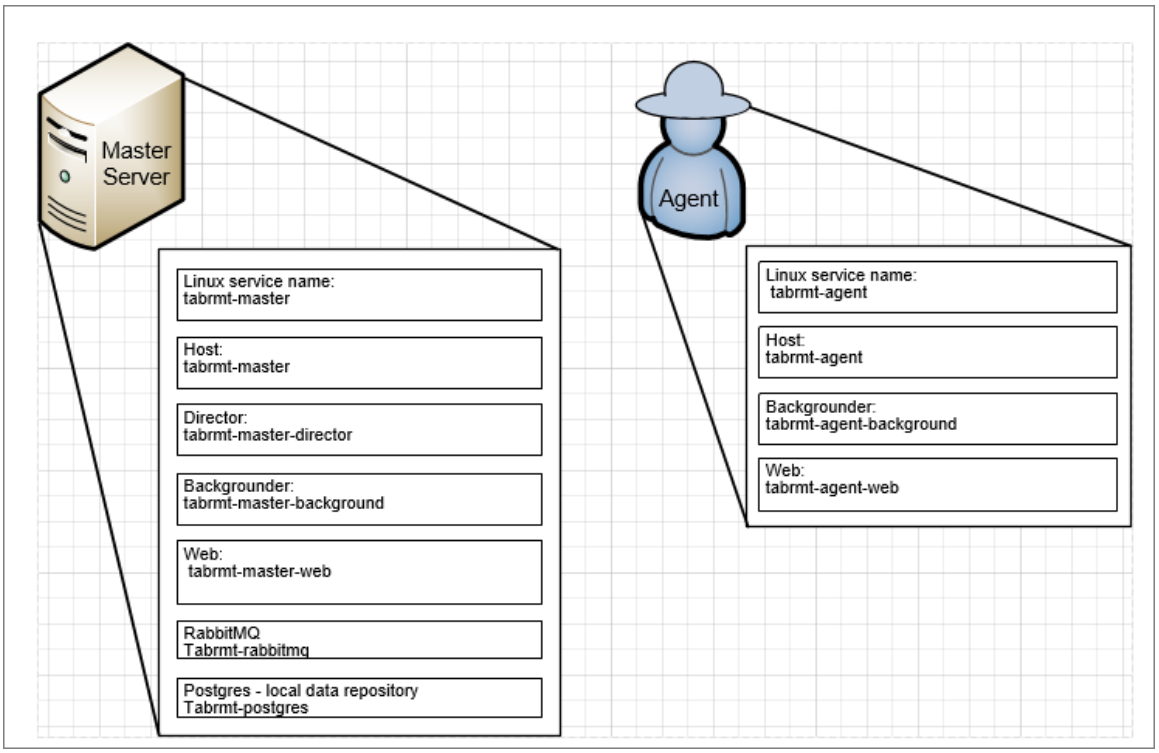
- 読み取り専用の組み込みユーザーを Tableau Server リポジトリに対して有効にします。このユーザーは Resource Monitoring Tool エージェントのインストール時に使用します。**これは、バージョン 2022.3 以降必須となります。**詳細については、Tableau Server リポジトリでデータを収集するを参照してください。
- リポジトリとメッセージ キュー サービスを Resource Monitoring Tool. の外部に構成するかどうかを決定します。

セットアップ アーキテクチャ

Resource Monitoring Tool セットアップの概要。この図は、RMT サーバーによって監視されている 2 つの Tableau Server クラスターを示しています。



RMT サーバーおよびエージェントにインストールされた Resource Monitoring Tool コンポーネントの詳細ビュー。



インストールの概要

RMT サーバーは、どちらのオペレーティング システムにもインストールでき、エージェントがインストールされているオペレーティング システムに依存しません。つまり、RMT サーバーとエージェントを異なるオペレーティング システムにインストールできます。

ただし、エージェントは Tableau Server クラスター ノードにインストールされるため、エージェントのオペレーティング システムの選択は、Tableau Server インストールに依存します。

同じ Resource Monitoring Tool を使用して、複数の Tableau Server クラスターを監視できます。各 Tableau クラスターは個別の環境として設定する必要があります。

外部構成

リポジトリとメッセージ キュー サービスは、いずれも AWS プラットフォーム上で RMT サーバーの外部でホストすることができます。リポジトリのホスティングには AWS RDS インスタンスを、メッセージ キューには AWS AMQ を使用します。構成、手順、その他詳細については、次を参照してください。

- Tableau Resource Monitoring Tool の外部リポジトリ
- Tableau Resource Monitoring Tool の外部メッセージ キュー サービス (RabbitMQ)

Linux でのインストール

Linux オペレーティング システムでは、コマンド ラインを使用して RMT サーバーとエージェントをインストールできます。

Resource Monitoring Tool をインストールする手順は次のとおりです。

1. コマンド ラインを使用して RMT サーバーをインストールします。

   Resource Monitoring Tool のインストーラーを起動すると、Resource Monitoring Tool を実行するために必要な特定のプログラムが最初にインストールされます。インストールされるプログラムには、RabbitMQ、Erlang、専用の PostgreSQL データベースが含まれます。その後、RMT サーバーのインストールに進みます。

2. コマンド ラインを使用してエージェントをインストールします。

   エージェントは、すべての Tableau Server ノードにインストールします。エージェントは、Tableau Server の使用状況とパフォーマンスに関する情報をレポート作成用に RMT サーバーに送信します。

HTTPS

ベストプラクティスとして、機密情報やユーザー認証資格情報を保護するために HTTPS を使用することをお勧めします。

サーバー構成で**必要な HTTPS** 設定は、ユーザーと RMT サーバーとの間の通信に使用されます。エージェントを登録するときにも使用します。エージェントと RMT サーバーの間でデータを収集する定期的なエージェント通信は、Rabbit MQ を介して行われます。

最初は自己署名証明書を使用して RMT サーバーをインストールし、エージェント登録時の通信を含む HTTPS 通信にその証明書を使用します。独自の証明書を使用して、自己署名証明書と置き換えることもできます。その置き換えは、RMT サーバーのインストール時に [サーバの設定] ページで行うか、インストールが完了した後に行うことができます。

SSL 証明書のモードと要件

Resource Monitoring Tool では、SSL 証明書を使用する3 つのモードがサポートされています。

- **既定:** このモードでは、インストーラーによって提供される既定の自己署名証明書が使用されます。
- **ローカル:** このモードでは、`/var/opt/tableau/tabrmt/master/config` フォルダーにファイルベースの証明書を指定できます。

証明書の次のガイドラインと要件に従ってください。

- 適切なドメインの HTTPS 証明書 (X.509 など) が必要です。これは、ローカルのセキュリティポリシーと証明書の要件によって異なります。たとえば、Resource Monitoring Tool で CName または SSL パススループロキシを使用している場合は、SAN 証明書を使用する必要がある場合があります。複数のサブドメインの場合は、ワイルドカード証明書がサポートされています。
- Resource Monitoring Tool では、PKCS # 12 および PEM 形式のみがサポートされています。

- Resource Monitoring Tool Web サーバーには、証明書、プライベートキー、およびオプションで信頼チェーンが必要です。

  プライベートキーは RSA または DSA のいずれかです。

  これらは、単一のファイルまたはグループ化されたファイルで提供できます。

  - 単一のファイルの例:
    - PKCS #12:.pfx または.p12 ファイル拡張子を持つ単一のファイル。
    - PEM: .pem 拡張子を持つ単一のファイルに、PEM でエンコードされた証明書 + プライベートキー (およびオプションでルート CA にチェーンする中間 Ca)。ファイル内のアイテムは、特定の順序にする必要はありません。
  - グループ化されたファイルの例:
    - .crt または .cer file ファイルに PEM でエンコードされた証明書 +
    - .key ファイルに PEM でエンコードされたプライベートキー、+ (オプションで)
    - 1 つ以上の .ca ファイルに PEM でエンコードされた認証局

ファイルとディレクトリの既定の場所:

RMT サーバー:

- インストール ディレクトリ: `/opt/tableau/tabrmt/master`
- 構成ファイル: `/var/opt/tableau/tabrmt/master/config.json`
- ログ: `/var/opt/tableau/tabrmt/master/logs`

エージェント:

- インストール ディレクトリ: `/opt/tableau/tabrmt/agent`
- Bootstrap: `/var/opt/tableau/tabrmt/agent/bootstrap`
- ログ: `/var/opt/tableau/tabrmt/agent/logs`

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

- sudo へのフル アクセス権を持つユーザー アカウント
- Tableau Server の管理者サイト ロール
- Resource Monitoring Tool 管理者アカウント。

次のステップ

RMT サーバーのインストール:

- Web インターフェイスを使用
- コマンド ラインを使用

Web インターフェイスを使用した RMT サーバーのインストール

Resource Monitoring Tool サーバー (RMT サーバー) は、ユーザーが操作する Web アプリケーションをホストします。また、エージェントからのデータを照合したり監視したりするバックグラウンド処理の多くを実行します。RMT サーバーは専用のハードウェアにインストールする必要があります。

> 現在、Web インターフェイスを使用したインストールは、Windows インストールのみでサポートされています。Linux でインストールする場合は、コマンド ラインを使用して RMT サーバーをインストールを参照してください。

RMT サーバーをインストールするには、次の手順を実行します。

RMT サーバーをインストールする主な手順は 3 つあります。

- サーバー ソフトウェアのインストール
- RMT サーバーの構成
- 新しい環境の作成

詳細なプロセスを順を追って以下に説明します。

1. **サーバー ソフトウェアのインストール**

    1. RMT サーバーのセットアップ プログラムを実行します。
    2. EULA を読んだ後、**[I agree to the license terms and conditions]** を選択し、**[Install]** をクリックします。
    3. [ユーザー アカウント制御] ダイアログが開く場合は、**[Yes]** をクリックし、インストーラーによる変更を許可します。
    4. 最初のステップとして、RMT サーバー インストーラーが最初に特定の前提条件をインストールします。必須コンポーネントには、RabbitMQ、Erlang、PostgreSQL データベースが含まれます。PostgreSQL データベースは、Tableau Server から収集した利用状況データを保存するために使用されます。その後、RMT サーバーのインストールに進みます。
    5. インストールの最後に、**Server Configuration** Web ページが表示されます。

2. **RMT サーバーの構成**

    1. ページには既に値が入力されています。必要に応じて、値を更新します。

        注: ホスト名 をメモしておきます。RMT サーバーへのアクセスに使用される Web インターフェイスでは、*https://<hostname>* 形式が使用されます。

2. 既定では、Resource Monitoring Tool は自己署名証明書を使用して、RMT サーバーとの HTTPS 通信を行います。独自の証明書を利用するには、証明書の指紋の値を、使用する証明書の指紋の値に置き換えます。セキュアな HTTPS 通信を義務付ける場合は、**Require HTTPS** オプションをオンにします。

   Require HTTPS オプションをオンにした場合は、証明書モードを選択し、必要に応じて追加の詳細を指定する必要があります。主に、次の 3 つのオプションから選択できます。

   1. **既定:** このモードでは、インストーラーによって提供される既定の自己署名証明書が使用されます。

   2. **ローカル:** このモードでは、Resource Monitoring Tool**/config** フォルダーにファイルベースの証明書を指定できます。**[ローカル]** モードを選択すると、**[Certificate Name (証明書名)]** フィールドが使用可能になります。一覧に表示されるオプションは、**/config** フォルダーにある証明書ファイル グループに対応します。

   3. **ストア:** このモードでは、Windows 証明書ストアにある証明書のサムプリントを入力することができます。

      [Password (パスワード)] フィールドは、選択した証明書にパスワードが必要な場合にのみ使用されます。

3. [**Test Server Configuration**] をクリックして、サーバー構成をテストします。

4. **[パスワード]** セクションで、ユーザー アカウントならびに後で作成する管理ユーザー アカウントに適用するパスワード要件を設定できます。

5. **[認証]** セクションで、セッションのタイムアウト期間を設定できます。設定したタイムアウト期間中にユーザーからセッションへのアクセスがない場合、セッションは期限切れになり、再度ログインする必要があります。デフォルトでは、240 分に設定されています。**[スライド有効期限]** オプションを有効にして、タイムアウト期間内にセッションにアクセスしたときにタイムアウト期間をリセットすることもできます。

6. [**Save and Restart Server**] をクリックします。

7. サーバーが再起動すると、管理ユーザーとパスワードの作成を求めるプロンプトが表示されます。

8. 前のステップで作成した管理者資格情報を使用してサインインすると、Resource Monitoring Tool に環境が作成されていないというメッセージが表示されます。新しい環境を設定する場合は、[**Add an environment**] をクリックします。

3. **新しい環境の作成**

   1. 新しい環境を作成します。識別子の名前など、環境の詳細を入力します。Tableau Server の REST API とTableau Server リポジトリの構成は、Tableau Server との通信に使用されます。Tableau Server リポジトリの構成はオプションですが、Tableau Server にアクセスするために推奨されている方法です。

      RMT が Tableau Server リポジトリに接続するときにセキュリティで保護された暗号化接続を構成するオプションがあります。RMT とTableau Server リポジトリデータベース間の SSL 接続を使用するには、SSL を使用するように Tableau Server を構成する必要があります。詳細については、内部 Postgres 通信用に SSL を設定するを参照してください。

   2. Tableau リポジトリを構成します。

      Tableau Repository Configuration セクション:

      1. **SSL Mode** ドロップダウン ボックスで **Prefer SSL** または **Require SSL** を選択して、Tableau リポジトリへの SSL 接続を構成します。**Disable** を選択すると、Tableau Server リポジトリ接続の作成で SSL が使用されることはありません。

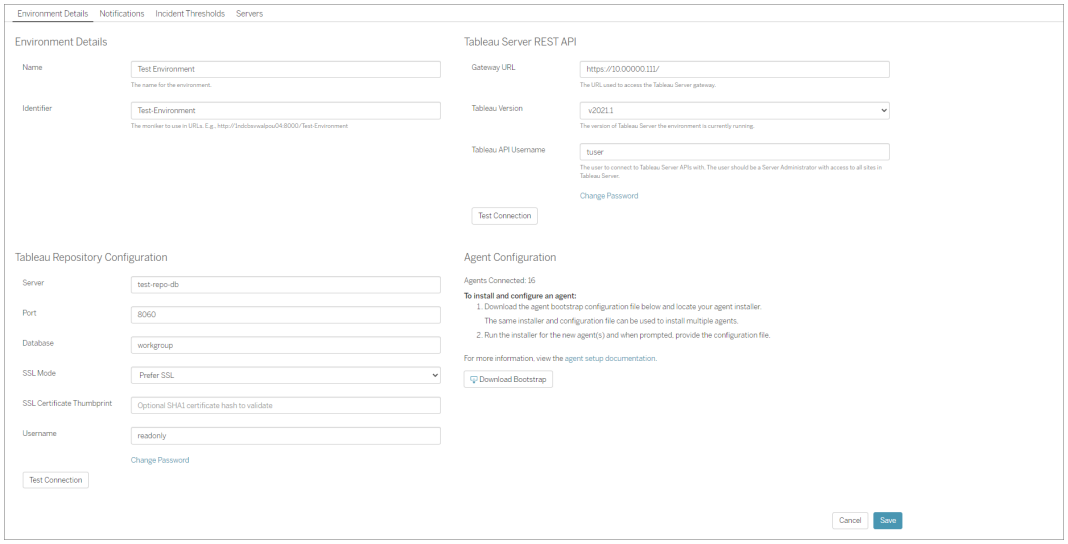
         Prefer SSL モードでは、Resource Monitoring Tool が最初に SSL の使用を試みて、失敗した場合は暗号化されていない接続を試みます。

Require SSL モードでは、SSL 接続が失敗した場合、Tableau Server リポジトリへの接続は完全に失敗します。この場合、Tableau Server の REST API 接続が Tableau Server との通信に使用されます。

2. Tableau Server によって生成された指紋を指定するか、**server.crt** ファイルを Resource Monitoring Tool マスター サーバー マシンにコピーするかを選択できます。証明書ファイルをコピーする場合は、指紋を指定する必要はありません。詳細については、クライアントからの直接接続を許可するようPostgres SSL を構成するを参照してください。

4. [**Save**] をクリックします。[**Agent Configuration**] ページに追加された新しいセクションが表示されます。このセクションでは、エージェントのインストールと構成に必要な bootstrap ファイルをダウンロードできます。

注: 環境設定を更新する場合は、bootstrap ファイルをダウンロードする前に **[保存]** をクリックする必要があります。



実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

- Resource Monitoring Tool をインストールするマシンでの管理者パーミッション。
- Tableau Server の管理者サイトロール
- Resource Monitoring Tool 管理者アカウント。

次のステップ

Web インターフェイスを使用したエージェントのインストール

Web インターフェイスを使用したエージェントのインストール

エージェントは、最小のサーバー リソースを消費して Resource Monitoring Tool サーバー (RMT サーバー) にデータを送信する軽量なプロセスです。Tableau Server の各ノードに Resource Monitoring Tool エージェントをインストールします。エージェントをインストールして登録するには、エージェントの bootstrap 構成ファイルをダウンロードして、Resource Monitoring Tool エージェントノードがアクセスできる場所に保存します。

> 現在、Web インターフェイスを使用したインストールは、Windows インストールのみでサポートされています。Linux でインストールする場合は、コマンド ラインを使用してエージェントをインストールするを参照してください。

インストールの前に

- bootstrap ファイルをダウンロードします。**bootstrap ファイルは、ダウンロードしてから 24 時間のみ有効です。bootstrap ファイルを 24 時間を超えて使用している場合は、bootstrap ファイルを再生成する必要があります。**
- エージェント登録を完了するには、https エンドポイントと RabbitMQ の両方を介して通信する必要があります。これらの通信のために、ポート 443 とポート 5672 の両方が開いていることを確認してください。

エージェントの bootstrap ファイルをダウンロードするステップ

RMT サーバーから bootstrap ファイルをまだダウンロードしていない場合は、次の手順を実行します。

1. RMT サーバーの Web インターフェイス (*https://<hostname>*) を使用し、[**Admin**] メニューから [**Environments**] を選択します。
2. bootstrap ファイルは、ホーム ページの [環境の概要] タブから直接ダウンロードできます。

> **注:** RMT サーバーのセットアップの一部として環境を作成していない場合は、Web インターフェイスを使用した RMT サーバーのインストールトピックで説明されている手順 12 ~ 15 に従ってください。

3. bootstrap 構成ファイルを、Resource Monitoring Tool エージェントをインストールしている Tableau Server ノードからアクセス可能な場所に保存します。

エージェントをインストールするステップ

> 現在、Web インターフェイスを使用したインストールは、Windows インストールのみでサポートされています。Linux でインストールする場合は、コマンドラインを使用してエージェントをインストールするを参照してください。

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

- Resource Monitoring Tool をインストールするマシンでの管理者パーミッション。
- Tableau Server の管理者サイトロール

次のステップ

Resource Monitoring Tool サーバー構成

コマンドラインを使用して RMT サーバーをインストール

Resource Monitoring Tool Server (RMT サーバー) は、ユーザーが操作する Web アプリケーションをホストします。また、エージェントからのデータを照合したり監視したりするバックグラウンド処理の多くを実行します。RMT サーバーは専用のハードウェアにインストールする必要があります。

このトピックでは、コマンドラインを使用して RMT サーバーをインストールする手順について説明します。コマンドライン インストールは、Windows オペレーティング システムと Linux オペレーティング システムの両方でサポートされています。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

Linux でのインストール

RMT サーバーをインストールするには、次の手順を実行します。

1. **RMT サーバーのインストール:**

    1. rpm または .deb RMT サーバー インストーラーをダウンロードし、RMT サーバーのインストール先のマシンからアクセスできる場所に保存します。

    2. sudo アクセス権を持つユーザーとして RMT サーバーのインストール先のマシンにログインします。

    3. 次のコマンドを実行して、RMT サーバーをインストールします。そこでは、<version> が major-minor-maintenance として書式設定されます。

        CentOS を含む RHEL ライクなディストリビューションの場合:

        ```
        sudo yum install <pathtormtserverinstaller>/Tabrmt-Master-x86_64-<version>.rpm
        ```

        Ubuntu ディストリビューションの場合:

        ```
        sudo apt install ./<pathtormtserverinstaller>/Tabrmt-Master-amd_64-<version>.deb
        ```

    これにより、パッケージと、RabbitMQ、Erlang、PostgreSQL データベースを含む前提条件がインストールされます。PostgreSQL データベースは、Tableau Server から収集した利用状況データを保存するために使用されます。その後、RMT サーバーのインストールに進みます。

2. **RMT サーバーの初期化**

RMT サーバーを初期化するときは、エンドユーザー ライセンス契約 (EULA) に明示的に同意する必要があります。また、既定以外の構成を指定するオプションもあります。RMT サーバーを既定の構成で初期化するには、次のコマンドを実行します。

```
sudo /opt/tableau/tabrmt/master/install-scripts/initialize-rmt-master --accepteula
```

EULA は `/opt/tableau/tabrmt/master/docs` フォルダーにあります。

バージョン 2023.1 より、RMT で使用するカスタム実行アカウントやその他の構成オプションを指定することができます。既定では、RMT は `rmt-master` というアカウントを作成し、そこで実行します。RMT サーバーが使用するカスタム実行アカウントを指定するには、初期化スクリプトの実行時に `--unprivileged-user` オプションを含めます。`initialize-rmt-master` スクリプトで使用可能なすべてのスイッチについては、「RMT サーバー初期化スクリプトのオプション」を参照してください。

3. **RMT サーバーの構成**

   1. 次のコマンドを `tabrmt-master` ユーザーとして実行します。

      ```
      sudo su --login tabrmt-master
      ```

      ```
      rmtadmin master-setup [options]
      ```

      構成オプションは、コマンドプロンプトまたは構成ファイルから指定できます。オプションを指定しない場合、管理者パスワードを除く既定値が適用されます。管理者ユーザー名が [admin] に設定され、パスワードの入力を求められます。

      必須のパスワード パラメーターを含むコマンド例:

      ```
      rmtadmin master-setup --admin-username=<name of the administrator user> --admin-password=<administrator user password>
      ```

      次の表に、RMT サーバーの構成に使用される必須オプションと一般的に使用されるオプションを示します。構成オプションの完全なリストについては、「rmtadmin コマンド ライン ユーティリティ」を参照してください。

注: HTTPS オプションを要求すると、RMT サーバーとユーザー間の通信を安全に行うことができます。通信に HTTPS が必要な場合は、これらの通信に使用する証明書のモードも選択する必要があります。以下の表に、さまざまなオプションを示します。これらのモードと証明書の詳細については、「SSL 証明書のモードと要件」を参照してください。

オプション	必須か?	既定	説明
admin-password	はい パスワードは、コマンドラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。	N/A	管理者ユーザーのパスワード
admin-password-file	いいえ パスワードは、コマンドラインで指	N/A	管理者ユーザーのパスワードが保存されているファイル **注:** `tabrmt-master` ユーザー

オプション	必須か?	既定	説明
	定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。		はこのファイルへのアクセス権を持っている必要があります。
admin-username	いいえ	admin	管理者ユーザーのユーザー名
http-port	いいえ	80	
require-https	いいえ	True	HTTP トラフィックを HTTPS にリダイレクトします。
https-certificate-mode	いいえ	**"Default"** 使用可能なオプション: 既定 ローカル	HTTPS 証明書に対して実行する証明書検索のタイプ **既定:** このモードでは、インストーラーによって提供される既定の自己署名証明書が使用されます。 **Local:** このモードでは、*/var/opt/tableau/tabrmt/master/config* フォルダーにファイルベースの証明書を指定できます。

オプション	必須か?	既定	説明
https-certificate-local-name	いいえ 注: 指定しない場合、Resource Monitoring Tool は自己署名証明書と共にインストールされ、HTTPS 通信にその証明書を使用します。	Null	ファイル拡張子のない HTTPS 証明書ファイルの名前。
https-certificate-local-password	いいえ	Null	HTTPS 証明書に使用するパスワード
https-certificate-local-password-file	いいえ	Null	HTTPS 証明書に使用するパスワードを含むファイルへのパス

4. **環境を作成する:**

    1. 次のコマンドを実行して、環境を作成します。

    ```
    rmtadmin create-env --name=<myenvironment> --api-
    username=<TableauServer API user name> --api-
    ```

```
password=<password for the Tableau Server API user
account>
```

このコマンドで使用可能なオプションを使用して環境を構成します。考慮すべきいくつかの重要な構成オプションを次に示します。

- Tableau Server の REST API とTableau Server リポジトリの構成は、Tableau Server との通信に使用されます。Tableau Server リポジトリの構成はオプションですが、Tableau Server にアクセスするために推奨されている方法です。
- RMT が Tableau Server リポジトリに接続するときにセキュリティで保護された暗号化接続を構成するオプションがあります。RMT とTableau Server リポジトリデータベース間の SSL 接続を使用するには、SSL を使用するように Tableau Server を構成する必要があります。詳細については、内部 Postgres 通信用に SSL を設定するを参照してください。

次の表に、いくつかの一般的なオプションを示します。オプションの完全なリストについては、「rmtadmin コマンド ライン ユーティリティ」を参照してください。

オプション	必須か?	既定	説明
--name	あり	N/A	環境 の名前。
--gateway-url	あり	N/A	Tableau Server ゲートウェイへのアクセスに使用される URL。
--version	あり	N/A	この環境が監視する Tableau Server のバージョン。
--api-username	なし	Null	Tableau Server API への接続に使用されるアカウントのユーザー名。ユーザー アカウントは、すべての Tableau Server サイトにアクセスできる Tableau Server 管理者である必要があります。

オプション	必須か?	既定	説明
--api-password	なし (Tableau API ユーザー名を指定する場合は、パスワードを指定するか、ファイル パスとパスワードを含むファイルを指定します)	Null	Tableau Server API への接続に使用される Tableau Server API ユーザー アカウントのパスワード。
--api-password-file	なし	Null	ファイルへのパスと、Tableau Server API ユーザー アカウントのパスワードを含むファイルの名前。
--repository-server	はい	Null	Tableau Server と共にインストールされている PostgreSQL データベースのサーバー名。
--repository-port	はい	Null	Tableau Server リポジトリデータベースのポート番号。
--repository-username	はい	Null	Tableau Server リポジトリと共にインストールされている PostgreSQL データベースへの接続に使用するユーザー名。

オプション	必須か?	既定	説明
			Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベースユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password	はい	Null	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザーアカウントのパスワード。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベースユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password-file	なし	Null	Tableau Server と共にインストールされているPostgreSQL データベー

オプション	必須か?	既定	説明
			スへの接続に使用するユーザーアカウントのパスワードが保存されているファイル名を含むパス。
--repository-ssl-mode	なし	Prefer	Tableau Server リポジトリSSLモード。 **Prefer** は SSL を優先して、**Require** は SSL を必須として、Tableau リポジトリへの SSL接続を構成します。 **Disable** は SSL を使用せずに、Tableau Server リポジトリへの接続を行います。
--repository-ssl-thumbprint	なし	Null	Tableau Server によって生成された指紋を指定するか、**server.crt** ファイルを Resource Monitoring Tool のサーバー (RMT サーバー) マシンにコピーするかを選択できます。証明書ファイルをコピーする場合は、指紋を指定する必要はありません。詳細については、クライアントからの直接接続を許可するよう Postgres SSL を構成するを参照してください。

2. bootstrap ファイルをダウンロードし、Tableau Server ノードからアクセスできる場所に保存します。

```
rmtadmin bootstrap-file --env=<myenvironment> --
filename=<The absolute or relative path including the file
name>
```

5. **オプションのステップ - SSD を使用しない場合のみ:** Resource Monitoring Tool は、既定で SSD 用に最適化されています。SSD ハードウェアを使用していない場合は、次のコマンドを実行します。

   ```
   sudo /opt/tableau/tabrmt/master/tabrmt-master optimize --no-ssd
   ```

Windows でのインストール

RMT サーバーをインストールするには、次の手順を実行します。

1. **RMT サーバーのセットアップ プログラムを実行する:**

   1. RMT サーバーのインストーラーをダウンロードし、RMT サーバーをインストールするマシンからアクセスできる場所に保存します。
   2. 管理者としてコマンドプロンプトを開きます。
   3. RMT サーバー インストーラーを保存した場所に移動し、次のコマンドを使用して exe ファイルを実行します。

      ```
      Tabrmt-Master-64bit-<version>.exe /silent ACCEPTEULA=1
      ```

      注: /silent または /quiet スイッチを使用して、UI やプロンプトを表示せずに無人でセットアップを実行します。

      既定のインストール ディレクトリは、通常次のようになります。*C:\Program Files\Tableau\Tableau Resource Monitoring Tool\master*別のインストール場所を指定するには、次のコマンドを実行します。

      ```
      Tabrmt-Master-64bit-<version>.exe /silent ACCEPTEULA=1
      InstallFolder="D:\Tableau Resource Monitoring Tool\master"
      ```

すべてのインストール コマンドのプロパティとスイッチの完全なリストは、以下のセクションで参照できます。

これにより、パッケージと、RabbitMQ、Erlang、PostgreSQL データベースを含む前提条件がインストールされます。PostgreSQL データベースは、Tableau Server から収集した利用状況データを保存するために使用されます。その後、RMT サーバーのインストールに進みます。

2. **RMT サーバーの構成**

    1. 次のコマンドを実行し、オプションを指定します。

        ```
        rmtadmin master-setup [options]
        ```

        構成オプションは、コマンドプロンプトまたは構成ファイルから指定できます。オプションを指定しない場合、管理者パスワードを除く既定値が適用されます。管理者ユーザー名が [admin] に設定され、パスワードの入力を求められます。

        必須のパスワード パラメーターを含むコマンド例:

        ```
        rmtadmin master-setup --admin-username=<name of the
        administrator user> --admin-password=<administrator user
        password>
        ```

        次の表に、RMT サーバーの構成に使用される必須オプションと一般的に使用されるオプションを示します。構成オプションの完全なリストについては、「rmtadmin コマンド ライン ユーティリティ」を参照してください。

        > **注:** HTTPS オプションを要求すると、RMT サーバーとユーザー間の通信を安全に行うことができます。通信に HTTPS が必要な場合は、これらの通信に使用する証明書のモードも選択する必要があります。以下の表に、さまざまなオプションを示します。これらのモードと証明書の詳細については、「Tableau Resource Monitoring Tool のインストール」を参照してください。

オプション	必須か?	既定	説明
admin-password	はい パスワードは、コマンドラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。	N/A	管理者ユーザーのパスワード
admin-password-file	いいえ パスワードは、コマンドラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指	N/A	管理者ユーザーのパスワードが保存されているファイル

オプション	必須か?	既定	説明
	定しなければ、パスワードの入力が求められます。		
admin-username	いいえ	admin	管理者ユーザーのユーザー名
http-port	いいえ	80	
require-https	いいえ	True	HTTP トラフィックを HTTPS にリダイレクトします。
https-certificate-mode	いいえ	**既定** 使用可能なオプション: • 既定 • ストア • ローカル	HTTPS 証明書に対して実行する証明書検索のタイプ **既定:** このモードでは、インストーラーによって提供される既定の自己署名証明書が使用されます。 **ストア:** このモードでは、Windows 証明書ストアに証明書の指紋を入力できます。 **Local:** このモードでは、*<installation directory>\config* フォルダーでファイルベースの証明書を指定できます。既定の

オプション	必須か?	既定	説明
			場所は *C:\Program Files\Tableau\Tableau Resource Monitoring Tool\master\config* です。
https-certificate-store-thumbprint	いいえ	Null	HTTPS 証明書のハッシュ/指紋を使用して'ストア' 証明書モードで検索する
https-certificate-local-name	いいえ	Null 注: 指定しない場合、Resource Monitoring Tool は自己署名証明書と共にインストールされ、HTTPS 通信にその証明書を使用します。	ファイル拡張子のない HTTPS 証明書ファイルの名前。
https-certificate-local-password	いいえ	Null	HTTPS 証明書に使用するパスワード
https-certificate-local-password-file	いいえ	Null	HTTPS 証明書に使用するパスワードを含むファイルへのパス

3. **環境を作成する:**

1. 次のコマンドを実行して、環境を作成します。

   ```
   rmtadmin create-env --name=<myenvironment> --api-
   username=<TableauServer API user name> --api-
   password=<password for the Tableau Server API user account>
   ```

   このコマンドで使用可能なオプションを使用して環境を構成します。考慮すべきいくつかの重要な構成オプションを次に示します。

   - Tableau Server の REST API とTableau Server リポジトリの構成は、Tableau Server との通信に使用されます。Tableau Server リポジトリの構成はオプションですが、Tableau Server にアクセスするために推奨されている方法です。
   - RMT が Tableau Server リポジトリに接続するときにセキュリティで保護された暗号化接続を構成するオプションがあります。RMT とTableau Server リポジトリデータベース間の SSL 接続を使用するには、SSL を使用するように Tableau Server を構成する必要があります。詳細については、内部 Postgres 通信用に SSL を設定するを参照してください。

   次の表に、いくつかの一般的なオプションを示します。オプションの完全なリストについては、「rmtadmin コマンド ライン ユーティリティ」を参照してください。

オプション	必須か?	既定	説明
--name	あり	N/A	環境の名前。
--gateway-url	あり	N/A	Tableau Server ゲートウェイへのアクセスに使用される URL。
--version	あり	N/A	この環境が監視する Tableau Server のバージョン。
--api-username	なし	Null	Tableau Server API への接続に使用されるアカウントのユーザー名。ユーザー アカウントは、すべての

オプション	必須か?	既定	説明
			Tableau Server サイトにアクセスできる Tableau Server 管理者である必要があります。
--api-password	なし (Tableau API ユーザー名を指定する場合は、パスワードを指定するか、ファイル パスとパスワードを含むファイルを指定します)	Null	Tableau Server API への接続に使用される Tableau Server API ユーザー アカウントのパスワード。
--api-password-file	なし	Null	ファイルへのパスと、Tableau Server API ユーザー アカウントのパスワードを含むファイルの名前。
--repository-server	はい	Null	Tableau Server と共にインストールされている PostgreSQL データベースのサーバー名。
--repository-port	はい	Null	Tableau Server リポジトリデータベースのポート番号。
--repository-username	はい	Null	Tableau Server リポジトリと共にインストールされている PostgreSQL

オプション	必須か?	既定	説明
			データベースへの接続に使用するユーザー名。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベースユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password	はい	Null	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザーアカウントのパスワード。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベースユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。

オプション	必須か?	既定	説明
--repository-password-file	なし	Null	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザー アカウントのパスワードが保存されているファイル名を含むパス。
--repository-ssl-mode	なし	Prefer	Tableau Server リポジトリSSL モード。 **Prefer** は SSL を優先して、**Require** は SSL を必須として、Tableau リポジトリへの SSL接続を構成します。 **Disable** は SSL を使用せずに、Tableau Server リポジトリへの接続を行います。
--repository-ssl-thumbprint	なし	Null	Tableau Server によって生成された指紋を指定するか、**server.crt** ファイルを Resource Monitoring Tool のサーバー (RMT サーバー) マシンにコピーするかを選択できます。証明書ファイルをコピーする場合は、指紋を指定する必要はありません。詳細については、クライアントからの直接接続を許可するようPostgres SSL を構成するを参照してください。

2. bootstrap ファイルをダウンロードし、Tableau Server ノードからアクセスできる場所に保存します。

```
rmtadmin bootstrap-file --env=<myenvironment> --
filename=<The absolute or relative path including the file
name>
```

Windows インストールのプロパティとスイッチ

スイッチ:

スイッチ	説明	コメント
`/install | /uninstall`	セットアップを実行して、Resource Monitoring Tool をインストールするかアンインストールします。	既定では、UI とすべてのプロンプトがインストールされ、表示されます。新規インストールで InstallFolder プロパティを使用してディレクトリを指定していない場合は、*C:\Program Files\Tableau\Tableau Resource Monitoring Tool\master* が使用されます。Resource Monitoring Tool が既にインストールされている場合、セットアップは現在のインストールと同じ場所を使用します。 データディレクトリを含む Resource Monitoring Tool を完全に削除するには、`/uninstall DELTEDATADIR=1` を使用します。
`/passive`	最小限の UI を使用してセットアップを実行します。プロンプトは表示されません。	
`/quiet | /silent`	無人の完全なサイレントモードでセットアップを実行します。Web インターフェイスもプロンプトも表示されません。	/quiet と/silent のどちらかを使用します。
`/norestart`	再起動が必要な場合	注: まれに、このオプションを使用しても再起動を

スイッチ	説明	コメント
	でも、Windows を再起動せずにセットアップを実行します。	抑制できない場合があります。以前にシステムの再起動をスキップしたことがあると、これが発生する可能性が高くなります。たとえば、他のソフトウェアのインストール中に再起動をスキップした場合などです。
`/log <log-file>`	指定したファイルおよびパスに情報を記録します。既定では、ログ ファイルは %TEMP% に作成され、Tableau_Resource_Monitoring_Tool_<version_code> という命名規則が使用されます。	ファイルの場所を指定しない場合、ログ ファイルは TEMP フォルダー *C:\Users\<username>\AppData\Local\Temp* に書き込まれます。インストール後、このログ ファイルでエラーを確認します。 例: `<Setup file> /quiet /log="C:\Tableau\Logs\RmtInstall`

プロパティ:

プロパティ	説明	コメント
`InstallFolder=<path\to\installation\directory>`	指定した既定以外のインストール場所にインストールします。	RMT をインストールする場所を指定します。指定しない場合、RMT は *installedC:\Program Files\Tableau\Tableau Resource Monitoring Tool\master* にインストールされます。 例: `<Setup file> /silent InstallFolder="D:\Tableau\Tableau Resource Monitoring Tool\master"`

プロパティ	説明	コメント
ACCEPTEULA=1|0	エンドユーザーライセンス契約 (EULA) に同意します。最初のノードと追加ノードのどちらにおいても、静的インストール、サイレントインストール、パッシブインストールに必要です。1 = true、	/passive、/silent、/quiet の使用時に含まれていない場合、セットアップはエラーなどが発生することなく失敗します。含まれているが 0 に設定されている場合、セットアップは失敗します。

プロパティ	説明	コメント
	EULA を受け入れる、0 = false、EULA を受け入れない。	

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

Windows

- Resource Monitoring Tool をインストールするマシンでの管理者パーミッション。
- Tableau Server の管理者サイトロール
- Resource Monitoring Tool 管理者アカウント。

Linux

- エージェントのインストールに使用するユーザーアカウントの sudo へのフルアクセス権
- Resource Monitoring Tool の管理者アカウント

次のステップ

コマンドラインを使用してエージェントをインストールする

RMT サーバー初期化スクリプトのオプション

RMT サーバーをインストールしたら、サーバーを初期化する必要があります。既定では、初期化スクリプトを実行する際に含める必要があるフラグは `--accepteula` だけです。その他のオプションを使用すると、環境や企業のセキュリティ要件に基づいてインストールを柔軟にカスタマイズできます。

--accepteula

必須。

エンドユーザー ライセンス契約 (EULA) の条件を読み、同意したことを示します。

`-a <username>`

オプション。

初期化スクリプトを実行するユーザーではなく、指定されたユーザー名を適切なグループに追加します。これにより、ユーザーはグループが所有するリソースにアクセスできるようになります。これは、実行ユーザー アカウントとは異なります。

既定: 初期化スクリプトを実行しているユーザー

`-f`

オプション。

警告メッセージやディストリビューションのバージョン チェックを回避します。

`-h | -?`

オプション。

スクリプトのヘルプ テキストを表示します。

`-q`

オプション。

エラーと警告以外の出力を制止します。

`--debug`

オプション。

デバッグのため、実行される各コマンドを出力します。大量の出力が生成されます。

`--default-group=<value>`

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

Tableau RMT エージェントへのアクセスを許可するグループの名前。指定する場合、`--unprivileged-user` も指定する必要があります。

`---disable-account-creation`

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

存在しないアカウントやグループは作成されません。指定する場合、`--unprivileged-user` パラメーターに加え、`--default-group` パラメーターと`--rmt-<...>-group` パラメーターのいずれかまたは両方も指定する必要があります。これらのパラメーターが参照するユーザー ID とグループは、既に存在している必要があります。

`--rmt-authorized-group=<value>`

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

インストール ベースのインストール ディレクトリとinstall_dir/prerequisites フォルダーを所有するグループの名前。指定する場合、--unprivileged-user も指定する必要があります。

既定:"rmtmasterapp" または --default-group の値

--rmt-config-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/master/config ディレクトリを所有するグループの名前。指定する場合、--unprivileged-user も指定する必要があります。

既定:"rmtmasterconfig" または --default-group の値

--rmt-logs-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/master/logs ディレクトリを所有するグループの名前。指定する場合、--unprivileged-user も指定する必要があります。

既定:"rmtmasterlogs" または --default-group の値

--rmt-openssl-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/prerequisites/openssl ディレクトリを所有するグループの名前。

既定:"rmtopenssl" または --default-group の値

--rmt-postgres-app-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/prerequisites/postgresql13 ディレクトリを所有するグループの名前。

既定: "rmtpostgresapp" または --default-group の値

--rmt-postgres-data-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/master/logs ディレクトリを所有するグループの名前。指定する場合、--unprivileged-user も指定する必要があります。

既定: "rmtmasterlogs" または --default-group の値

--rmt-rabbitmq-app-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/prerequisites/rabbitmq ディレクトリを所有するグループの名前。

既定: "rmtrabbitmqapp" または --default-group の値

--rmt-rabbitmq-data-group=<value>

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

install_dir/data/rabbitmq ディレクトリを所有するグループの名前。

既定: "rmtrabbitmqdata" または --default-group の値

`--unprivileged-user=<value>`

バージョン: バージョン 2023.1.0 で追加されました。

オプション。

Tableau RMT Master を実行する権限のないユーザー アカウントの名前。RMT を初期化した後にアカウントを変更することはできません。

既定: "tabrmt-master"

コマンド ラインを使用してエージェントをインストールする

エージェントは、最小のサーバー リソースを消費して Resource Monitoring Tool サーバー (RMT サーバー) にデータを送信する軽量なプロセスです。Tableau Server の各ノードに Resource Monitoring Tool エージェントをインストールします。エージェントをインストールして登録するには、エージェントの bootstrap 構成ファイルをダウンロードして、Resource Monitoring Tool エージェント ノードがアクセスできる場所に保存します。

このトピックでは、コマンド ラインを使用して Resource Monitoring Tool エージェントをインストールする手順について説明します。コマンド ライン インストールは、Windows オペレーティング システムと Linux オペレーティング システムの両方でサポートされています。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

インストールの前に

- bootstrap ファイルをダウンロードし、RMT エージェントをインストールするノードがアクセスできる場所に保存します。**bootstrap ファイルは、ダウンロードしてから 24 時間のみ有効です。bootstrap ファイルを 24 時間を超えて使用している場合は、bootstrap ファイルを再生成する必要があります。**

- エージェント登録を完了するには、https エンドポイントとRabbitMQ の両方を介して通信する必要があります。これらの通信のために、ポート443 とポート5672 の両方が開いていることを確認してください。

Linux でのインストール

エージェントのインストールと構成

1. **インストールする**

    1. .rpm または .deb のエージェントインストーラーをダウンロードし、エージェントをインストールするマシンからアクセスできる場所に保存します。

    2. 次のコマンドを実行してエージェントをインストールします。<version> は "メジャー番号 - マイナー番号 - メンテナンス番号" の形式です。

        CentOS を含む、RHEL のようなディストリビューションの場合:

        ```
        sudo yum install Tabrmt-Agent-x86_64-<version>.rpm
        ```

        Ubuntu の場合:

        ```
        sudo apt install Tabrmt-Agent-amd64-<version>.deb
        ```

2. RMT エージェントの初期化

    RMT エージェントを初期化するときは、エンドユーザー ライセンス契約 (EULA) に明示的に同意する必要があります。また、既定以外の構成を指定するオプションもあります。RMT エージェントを既定の構成で初期化するには、次のコマンドを実行します。

    ```
    sudo /opt/tableau/tabrmt/agent/install-scripts/initialize-rmt-agent --accepteula
    ```

    EULA は `/opt/tableau/tabrmt/agent/docs` フォルダーにあります。

    バージョン 2023.1 より、RMT で使用するカスタム実行アカウントやその他の構成オプションを指定することができます。既定では、RMT は `rmt-agent` というアカウントを作成し、そこで実行します。RMT エージェントが使用するカスタム実行アカウントを指定するには、初期

化スクリプトの実行時に --unprivileged-user オプションを含めます。initialize-rmt-agent スクリプトで使用可能なすべてのスイッチについては、「RMT エージェントの初期化スクリプトのオプション」を参照してください。

3. **登録する**

   1. ログオフして ***tabrmt-agent*** ユーザーとしてログオンすると、***abrmt-agent*** ユーザーの権限で実行する必要がある rtmadmin コマンドを実行できます。再びログオンした時点で新しいセッションが作成され、このときにグループのメンバーシップの変更が反映されます。

      ```
      sudo su --login tabrmt-agent
      ```

   2. 次のコマンドを実行し、bootstrap ファイルが配置されているパスを指定します。エージェントがインストールされているノードの説明を入力します。

      ```
      rmtadmin register <bootstrap file path\file> --server-name=<Friendly name of machine> --server-description=<server description> --username=<name of the RMT admin user>
      ```

      RMT 管理ユーザーのパスワードを求めるプロンプトが表示されます。

      > **注:** tabrmt-agent ユーザーはデフォルトで基本の作業ディレクトリ: /var/opt/tableau/tabrmt/agent からコマンドを実行するため、それに応じたファイルパスを指定する必要があります。たとえば、bootstrap ファイルを /var/opt/tableau/tabrmt/agent/bootstrap/ フォルダーに推奨どおりに配置した場合、ファイル パスは /var/opt/tableau/tabrmt/agent/bootstrap/<bootstrap_file_name>. となります。

      次の表に、エージェントの登録に使用する構成オプションを示します。

オプション	必須か?	既定	説明
--bootstrap file	はい	<なし>	bootstrap ファイルの場所
--username	はい	<なし>	通常、RMT サーバーのインストール中に作成した管理者ユーザーです。
--password	はい	<なし>	ユーザー アカウントのパスワードです。
--password-file	いいえ パスワードは、コマンド ラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。	<なし>	パスワードが保存されているファイル名を含むパス。
-- server-name	いいえ	マシンのホスト名	エージェントをインストールしているコンピュータの名前。オプションを指定しない場合、このフィールドはデフォルトでマシンのホスト名になります。
--server-description	いいえ	<なし>	エージェントをインストールしているコンピュータの説明。オプションを指定しない場合、このフィールドは空白のままになります。

エージェントのインストールと構成

1. bootstrap ファイルをダウンロードし、Tableau Server ノードからアクセスできる場所に保存します。

   ```
   rmtadmin bootstrap-file --env=<myenvironment> --filename<The absolute or relative path including the file name>
   ```

2. エージェントのセットアップ プログラムを実行する

   1. エージェント インストーラーをダウンロードし、Tableau Server マシンからアクセスできる場所に保存します。

   2. 管理者としてコマンドプロンプトを開きます。

   3. エージェントインストーラーの場所に移動し、次のコマンドを使用して **exe** ファイルを実行します。

      ```
      Tabrmt-Agent-64bit-<version>.exe /silent ACCEPTEULA=1
      ```

      注: /silent または /quiet スイッチを使用して、UI やプロンプトを表示せずに無人でセットアップを実行します。

      デフォルトのインストール フォルダーは、通常、*:C:\Program Files\Tableau\Tableau Resource Monitoring Tool\agent* です。別のインストール場所を指定するには、次のコマンドを実行します。

      ```
      Tabrmt-Agent-64bit-<version>.exee /silent ACCEPTEULA=1 InstallFolder="D:\Tableau Resource Monitoring Tool\agent"
      ```

3. エージェントを登録する

   1. 次のコマンドを実行してエージェントを登録します。

      ```
      rmtadmin register <bootstrap file path\file> --server-name=<Friendly name of machine> --server-description=<server description>
      ```

次の表に、エージェントの登録に使用する構成オプションを示します。

オプション	必須か?	既定	説明
--bootstrap file	はい	<なし>	bootstrap ファイルの場所
--username	はい	<なし>	通常、RMT サーバーのインストール中に作成した管理者ユーザーです。
--password	はい	<なし>	ユーザー アカウントのパスワードです。
--password-file	いいえ パスワードは、コマンド ラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。	<なし>	パスワードが保存されているファイル名を含むパス。
-- server-name	いいえ	マシンのホスト名	エージェントをインストールしているコンピュータの名前。オプションを指定しない場合、このフィールドはデフォルトでマシンのホスト名になります。
-- server-description	いいえ	<なし>	エージェントをインストールしているコンピュータの説明。オプションを指定しない場合、このフィールドは空白のままになります。

Windows インストールのプロパティとスイッチ

スイッチ:

スイッチ	説明	コメント
`/passive`	最小限の UI を使用してセットアップを実行します。プロンプトは表示されません。	
`/quiet | /silent`	無人の完全なサイレントモードでセットアップを実行します。Web インターフェイスもプロンプトも表示されません。	/quiet と/silent のどちらかを使用します。

プロパティ:

プロパティ	説明	コメント
`InstallFolder=<path\to\installation\directory>`	指定した既定以外のインストール場所にインストールします。	RMT をインストールする場所を指定します。指定しない場合、RMTは *installedC:\Program Files\Tableau\Tableau Resource Monitoring Tool\master* にインストールされます。 例: `<Setup file> /silent InstallFolder="D:\Tableau\Tableau Resource Monitoring Tool\agent"`

プロパティ	説明	コメント
ACCEPTEULA=1|0	エンドユーザーライセンス契約 (EU-LA) に同意します。最初のノードと追加ノードのどちらにおいても、静的インストール、サイ	/passive、/silent、/quiet の使用時に含まれていない場合、セットアップはエラーなどが発生することなく失敗します。含まれているが 0 に設定されている場合、セットアップは失敗します。

プロパティ	説明	コメント
	レントインストール、パッシブインストールに必要です。1 = true、EULA を受け入れる、0 = false、EULA を受け入れない。	

マルチノードTableau Server でのエージェントのインストール

Tableau Server の各 ノードで、上記の手順を実行します。RMT サーバーの Web インターフェイスで、エージェントがインストールされているすべてのノードを表示できるはずです。

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

Windows:

- Resource Monitoring Tool をインストールするマシンでの管理者パーミッション。
- Tableau Server の管理者サイトロール

Linux:

- エージェントのインストールに使用するユーザー アカウントの sudo へのフル アクセス権

次のステップ

Tableau Resource Monitoring Tool の構成

RMT エージェントの初期化スクリプトのオプション

RMT エージェントをインストールしたら、エージェントを初期化する必要があります。既定では、初期化スクリプトを実行する際に含める必要があるフラグは `--accepteula` だけです。その他のオプションを使用すると、環境や企業のセキュリティ要件に基づいてインストールを柔軟にカスタマイズできます。

RMT エージェントの初期化オプション

--accepteula

必須。

エンドユーザー ライセンス契約 (EULA) の条件を読み、同意したことを示します。

`-a <username>`

任意の項目です。

初期化スクリプトを実行するユーザーではなく、指定されたユーザー名を適切なグループに追加します。これにより、ユーザーはグループが所有するリソースにアクセスできるようになります。これは、実行ユーザー アカウントと同じではありません。

既定: 初期化スクリプトを実行しているユーザー

`-f`

任意の項目です。

警告メッセージやディストリビューションのバージョン チェックを回避します。

`-h | -?`

任意の項目です。

スクリプトのヘルプ テキストを表示します。

`-q`

任意の項目です。

エラーと警告以外の出力を制止します。

`--debug`

任意の項目です。

デバッグを目的として実行される各コマンドを印刷します。広範な出力を生成します。

`--default-group=<value>`

バージョン: 2023.1.0 で追加されました。

任意の項目です。

Tableau RMT エージェントへのアクセスを許可するグループの名前。指定する場合は、--unprivileged-user も指定する必要があります。

---disable-account-creation

バージョン: 2023.1.0 で追加されました。

任意の項目です。

存在しないアカウントやグループは作成されません。指定する場合、--unprivileged-user パラメーターに加え、--default-group パラメーターと--rmt-<...>-group パラメーターのいずれかまたは両方も指定する必要があります。これらのパラメーターが参照するユーザー ID とグループは、既に存在している必要があります。

--rmt-authorized-group=<value>

バージョン: 2023.1.0 で追加されました。

任意の項目です。

インストール ベースのインストール ディレクトリとinstall_dir/prerequisites フォルダーを所有するグループの名前。指定する場合は、--unprivileged-user も指定する必要があります。

既定: "rmtagentapp" または --default-group の値

--rmt-config-group=<value>

バージョン: 2023.1.0 で追加されました。

任意の項目です。

install_dir/agent/config ディレクトリを所有するグループの名前。指定する場合は、--unprivileged-user も指定する必要があります。

既定: "rmtagentconfig" または --default-group の値

--rmt-logs-group=<value>

バージョン: 2023.1.0 で追加されました。

任意の項目です。

`install_dir/agent/logs` ディレクトリを所有するグループの名前。指定する場合は、`--unprivileged-user` も指定する必要があります。

既定: "`rmtagentlogs`" または `--default-group` の値

`--unprivileged-user=<value>`

バージョン: 2023.1.0 で追加されました。

任意の項目です。

Tableau RMT エージェントの実行に使用される権限のないユーザー アカウントの名前。RMT を初期化した後にアカウントを変更することはできません。

既定: "`tabrmt-agent`"

Tableau Resource Monitoring Tool の外部リポジトリ

Tableau Resource Monitoring Tool は PostgreSQL データベースをリポジトリとして使用し、パフォーマンスや使用状況に関する情報を提供するために使用される Tableau Server の監視データを収集して保存します。このデータベースは Tableau Server データベースから分離されています。

2022.3 リリース以前は、この PostgreSQL リポジトリは、RMT サーバーを使用するマシンに RMT サーバーとともに自動的にインストールされ、これ以外のオプションはありませんでした。バージョン 2022.3 から、ローカルにインストールされたデータベースの代わりに、外部でホストされている PostgreSQL データベースを使用するように RMT サーバーを再構築できるようになりました。RMT サーバーが外部でホストされている PostgreSQL データベースを使用するように構成されている場合、これを外部リポジトリと呼びます。

外部リポジトリをホストするプラットフォームとして、AWS RDS インスタンスがサポートされています。リポジトリデータベースを外部でホストすることで、RMT サーバーのリポジトリコンポーネントの可用性、バックアップ、および災害復旧を管理することができます。

この外部リポジトリを構成するオプションが追加されたことにより、RMT サーバーを次のように構成することができます。

- **ローカルにインストールされたリポジトリの使用を継続する**: RMT サーバーとともに同じマシンに自動的にインストールされている PostgreSQL データベースを RMT サーバーのリポジトリとして使用します。この構成にこれ以上の変更を加えることはありません。
- **RMT サーバーが外部リポジトリを使用するように構成する**: RMT が使用する PostgreSQL データベースを RMT サーバーの外部でホストします。この場合、インストール後にいくつかの構成を変更する必要があります。現在、PostgreSQL データベースのホスティングプラットフォームとして、AWS RDS インスタンスのみがサポートされています。

外部リポジトリを設定および管理するためには、AWS RDS プラットフォームと PostgreSQL データベースをよく理解する必要があります。プラットフォームに関する説明は、AWS サイトのドキュメントを参照することをお勧めします。

また、Resource Monitoring Tool の外部リポジトリの構成を設定し、管理する方法も知っておく必要があります。このトピックでは、以下のセクションでその方法を取り上げます。

Resource Monitoring Tool の新規インストール

このセクションで説明する内容は、新規にインストールされた Resource Monitoring Tool に適用されます。RMT の既存の展開があり、外部リポジトリ構成に移行したい場合は、このトピックの「既存の Resource Monitoring Tool インストール」のセクションを参照してください。

RMT サーバーをインストールすると、RMT サーバーを使用するマシンにリポジトリデータベースが自動的にインストールされます。外部リポジトリの構成プロセスは、インストール後に行う構成です。ここで、外部でホストされている PostgreSQL データベースを使用するように RMT サーバーを再度構成します。

以下のステップに従い、Resource Monitoring Tool をインストールし、外部リポジトリを使用するように RMT サーバーを構成し直します。

1. **外部リポジトリを作成する:** 以下の推奨事項に従って、外部リポジトリをホストする RDS インスタンスを作成します。

- **PostgreSQL インスタンスの最小推奨仕様:**
  - db.m6g.2xlarge (8 vcpu、32 GiB RAM)
  - SSD ストレージ、ディスク空き容量 500GB
- **インスタンスの構成値**
  - 最大ワーカープロセス数: 合計論理プロセッサー数
  - Gather 当たりの最大ワーカー数: Min(合計論理プロセッサー数/2、4)
  - 最大パラレルワーカー数: 合計論理プロセッサー数
  - 最大パラレル メンテナンス ワーカー数: Min(合計論理プロセッサー数/2、4)
  - 共有バッファ: 合計 メモリ量/4
  - 有効キャッシュサイズ: 合計 メモリ量の 75%
  - メンテナンス作業用 メモリ: 合計 メモリ量/16 Linux、最大 2GB、Windows は 1MB
  - WAL バッファ: 共有バッファから算出、16 MB 程度
  - ワーク メモリ: 共有バッファとGather 当たりの最大ワーカー数から算出

AWS 上の PostgreSQL DB インスタンスを作成して接続する方法については、AWS ドキュメントサイトのこちらのページをご参照ください。

2. **RMT のインストール:**「コマンドラインを使用して RMT サーバーをインストール」の説明に従って RMT サーバーをインストールします。ただし、**環境構築のステップは省略します。このステップは、外部 リポジトリを使用するように RMT サーバーを構成した後に行います。ローカルの PostgreSQL データベースが最初にインストールされることに注意してください。**
3. **外部 リポジトリへの接続をテストする:** RMT サーバーとともにインストールされている PostgreSQL の管理ツール psql を使用して、新しいデータベースへの接続をテストします。

- ローカルにインストールされた PostgreSQL のインストール ディレクトリの 'bin' フォルダーに移動します。

  ```
  /var/opt/tableau/tabrmt/prerequisites/postgresql<version number>/bin
  ```

- 次のコマンドを実行して接続をテストします。**Aws_rds_servername** に下記の値を入力し、プロンプトが表示されたら、AWS で作成した PostgreSQL のパスワードを入力します。

  ```
  psql -h <aws_rds_servername> -p 5432 -d postgres -U postgres
  ```

  この時点でエラーが発生し、正常に接続できない場合は、コマンドのパラメータ値を AWS RDS のコンソールの値と照らし合わせて確認してください。

- 接続に成功したら、次のコマンドを使用して psql セッションを終了します。

  ```
  \q
  ```

4. `rmtadmin master-setup` を実行し、**外部リポジトリを使用するように RMT サーバーを構成します**。以下のコマンド例を参考に、コマンドに編集して、Resource Monitoring Tool のインストール パス、AWS RDS のインスタンス名、ポート番号、RDS PostgreSQL 管理者ユーザー パスワードを反映させてください。

   - Resource Monitoring Tool RMT サーバーのインストール ディレクトリに移動します:

     ```
     sudo /var/opt/tableau/tabrmt/master
     ```

   - 外部リポジトリを構成するには、次のコマンドを実行します:

     ```
     rmtadmin master-setup --db-config=external --db-server=<aws_rds_servername> --db-database=<aws_rds_
     ```

```
database_name> --db-port=5432 --db-admin-username=postgres
--db-admin-password=<postgres_user_password>
```

5. **環境を作成**して、bootstrap ファイルをダウンロードします。

   - 次のコマンドを実行して、環境を作成します。

     ```
     rmtadmin create-env --name=<myenvironment> --api-
     username=<TableauServer API user name> --api-
     password=<password for the Tableau Server API user account>
     ```

   - bootstrap ファイルをダウンロードして、エージェントを登録します:

     ```
     rmtadmin bootstrap-file --env=<myenvironment> --
     filename=<The absolute or relative path including the file
     name>
     ```

     このステップの詳細は、「コマンド ラインを使用して RMT サーバーをインストール」を参照してください。

6. 「コマンド ラインを使用してエージェントをインストールする」の説明に従って、Tableau Server のノードにエージェントをインストールして登録します。

既存の Resource Monitoring Tool インストール

このリリースでは、外部リポジトリを設定するには、上記のセクションで説明されているステップに従って、Tableau Resource Monitoring Tool の新規インストールを開始する必要があります。

アップグレードのベストプラクティス

Tableau Resource Monitoring Tool のアップグレードと外部リポジトリへの移行を同時に行いたい場合の一般的なステップは以下の通りです。

環境の再構築による移行:

1. RMT サーバーとすべてのエージェントを 2022.3 以降にアップグレードします。
2. 2 - 4 までのステップは、新規インストールの手順と非常によく似ています。上記の「Resource Monitoring Tool の新規インストール」を参照してください。この移行には、主に以下の内容

が含まれます。

- AWS PostgreSQL DB インスタンスを作成します。
- 外部リポジトリを使用するように RMT サーバーを構成します。
- 環境を再構築し、すべてのエージェントを再登録します。

3. カスタム構成がある場合は再設定します。

注: 過去のデータは失われます。また、カスタム構成は設定し直す必要があります。

RMT の新しいバージョンが PostgreSQL のメジャー バージョンのアップグレードを必要とする場合のアップグレードについて

Resource Monitoring Tool の PostgreSQL メジャー バージョン要件に変更があった場合、ベストプラクティスとして、Resource Monitoring Tool をアップグレードする前に、まず外部リポジトリ PostgreSQL のバージョンをアップグレードします。詳細は以下の通りです。外部リポジトリ PostgreSQL のバージョンをアップグレードする必要があるかどうかを確認するには、「製品の互換性表」を参照してください。

PostgreSQL のメジャー バージョンのアップグレードが必要な場合、以下のステップに従い RMT と RDS インスタンスをアップグレードしてください。

1. RDS インスタンスのバックアップを作成します。これは、アップグレードをロールバックする必要がある場合に使用します。詳細については、AWS サイトの「Amazon RDS DB インスタンスのバックアップと復元」トピックを参照してください。
2. Resource Monitoring Tool ディレクトリに構成ファイルのコピーを作成します。構成ファイルは、以下の場所にあります。

   ```
   /var/opt/tableau/tabrmt/master/config.json
   ```

3. RDS インスタンスを新しいバージョンの PostgreSQL にアップグレードする方法については、AWS サイトの「Amazon RDS の PostgreSQL DB エンジンのアップグレード」トピックをご覧ください。

4. RMT サーバーをアップグレードします。データベースの移行を含むアップグレードが正常に完了した場合は、次のステップに進みます。アップグレードに失敗した場合は、このセクションの手順を参照して、アップグレードを復元してロールバックする方法を確認してください。

5. Tableau Server ノードのすべてのエージェントを新しい RMT バージョンにアップグレードします。詳細については、「Resource Monitoring Tool のアップグレード」を参照してください。

アップグレードの失敗からの復元

1. アップグレードした RMT サーバーをアンインストールします。

2. AWS RDS インスタンスをアップグレード前のバージョンに復元します。詳細については、AWS サイトの「Amazon RDS DB インスタンスのバックアップと復元」トピックを参照してください。

3. アップグレード前にバックアップしておいた構成ファイルを以下の場所に置き換えます。このフォルダーは、アンインストール時に削除されている可能性があるため、作成する必要があります。

   ```
   /var/opt/tableau/tabrmt/master/config.json
   ```

4. ローカル リポジトリをインストールする RMT サーバーをインストールします。

5. 外部 リポジトリを使用するように RMT サーバーを構成します:

   ```
   rmtadmin master-setup --db-config=external --db-server=<aws_
   rds_servername> --db-database=<aws_rds_database_name> --db-
   port=5432 --db-admin-username=postgres --db-admin-
   password=<postgres_user_password>
   ```

RMT とPostgreSQL のバージョンの互換性

外部リポジトリはバージョン2022.3 以降でのみ利用可能であるため、この表には RMT のバージョン2022.3 以降のみが記載されています。

RMT バージョン	RMT に付属の PostgreSQL バージョン	外部リポジトリでサポートされている PostgreSQL バージョン
2022.3 ~ 2024.2	13.7	13.7
2025.x	15.6	15.6

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

- sudo へのフル アクセス権を持つユーザー アカウント
- Tableau Server の管理者サイト ロール
- Resource Monitoring Tool 管理者アカウント。

Tableau Resource Monitoring Tool の外部メッセージ キュー サービス (RabbitMQ)

Tableau Resource Monitoring Tool は、RabbitMQ をメッセージ キュー サービスとして使用して、エージェントからデータを収集し、RMT サーバーに転送します。このキュー内の情報は処理され、最終的に RMT リポジトリ(PostgreSQL データベース) に格納されます。

2022.3 リリースより前は、RabbitMQ メッセージ キュー サービスは RMT サーバーと共に自動的にインストールされ、これが使用可能な唯一の構成でした。RMT バージョン 2022.3 以降では、外部でホストされた RabbitMQ サービスを使用するように RMT サーバーを再構成できるようになりました。外部でホストされたメッセージ キュー サービスを使用するように RMT サーバーが構成されている場合、その構成を外部メッセージ キュー サービスと呼びます。

バージョン 2022.3 で追加されたこの新しいオプションにより、RMT サーバーは次の方法で構成できます。

- **ローカルにインストールされたメッセージ キュー サービスを引き続き使用する。**これは、RMT サーバーと同じマシンに自動的にインストールされた RabbitMQ が、RMT サーバーのメッ

セージ サービスとして使用されることを意味し、この設定にはそれ以上の変更は加えられません。

- **外部メッセージ キュー サービスを使用するように RMT サーバーを構成する。**これは、RMT が使用する RAbbitMQ が RMT サーバーの外部でホストされることを意味します。現在サポートされている RabbitMQ のホスティング プラットフォームは、AWS AMQ のみです。エージェントからのすべてのメッセージは RabbitMQ を通過するため、これを外部でホストすることにより、RMT サーバーがインストールされているマシンからリソースを解放できます。

外部メッセージ キュー サービスをセットアップして管理するには、AWS AMQ プラットフォームを十分に理解している必要があります。AWS サイト上のドキュメントを確認することを推奨します。また、RMT の外部メッセージ キュー サービスを設定し、管理する方法も知っておく必要があります。このトピックでは、以下のセクションでその方法を取り上げます。

Resource Monitoring Tool の新規インストール

このセクションで説明する手順は、Resource Monitoring Tool の新規インストールに適用されます。既存のインストールがあり、ローカルの RabbitMQ を外部ホスト構成に移行したい場合は、このトピックの「Tableau Resource Monitoring Tool の既存のインストール」セクションを参照してください。

次の手順を使用して、Tableau リソース モニタリング ツールをインストールし、外部メッセージ キュー サービスを使用するように RMT サーバーを再構成します。

1. 次の推奨事項に従って、**Rabbit MQ 用の Amazon AMQ を作成し、**外部メッセージ キュー サービスをホストします。

    - エンジンの種類には、RabbitMQ エンジンを使用します。Rabbit MQ のバージョンについては、製品の互換性セクションを参照してください。
    - Rabbit MQ をローカルにインストールしたときに使用したものと同じデフォルト バージョンを使用します。詳細については、製品の互換性テーブルを参照してください。
    - 単一インスタンスのブローカーを使用します。
    - インスタンスの仕様: mq.m5.large、2 vCPU/8 GiB RAM
    - RabbitMQ のユーザー名とパスワードを作成します。

詳細については、AWS ドキュメントサイトの「Amazon MQ for Rabbit MQ の使用」を参照してください。

2. **RMT サーバーからの接続テスト**をメッセージ ブローカーに対して行います。RabbitMQ Web コンソールの URL を AWS MQ ページからコピーし、RMT サーバーの Web ブラウザーに貼り付けます。ブローカーをセットアップしたときに作成したユーザー名とパスワードでログインします。

3. **RMT サーバーのインストール**をこのトピックの手順に従って行いますが、**環境を作成する手順はスキップ**します。**このステップは、外部リポジトリを使用するように RMT サーバーを構成した後に行います。**

4. 次のように rmtadmin setup を実行して、外部の Rabbit MQ メッセージ キュー サービスを設定します。

```
rmtadmin master-setup --mq-config=external --mq-server=aws_amq_
servername --mq-vhost='/' --mq-port=5671 --mq-username=aws_amq_
username --mq-password='aws_amq_password' --mq-tls-certificate-
host=aws_amq_servername
```

5. 環境を作成し、bootstrap ファイルをダウンロードします。

   - 次のコマンドを実行して、環境を作成します。`rmtadmin create-env --name=<myenvironment> --api-username=<TableauServer API user name> --api-password=<password for the Tableau Server API user account>`

   - bootstrap ファイルをダウンロードして、エージェントを登録します。`rmtadmin bootstrap-file --env=<myenvironment> --filename<The absolute or relative path including the file name>`

     この手順については、「コマンド ラインを使用して RMT サーバーをインストール」トピックで詳しく説明しています。

6. 「コマンドラインを使用してエージェントをインストールする」トピックの手順に従って、Tableau Server ノードでエージェントを再登録します。

Tableau Resource Monitoring Tool の既存のインストール

このリリースでは、外部メッセージング サービスを構成するには、基本的に上記のセクションで説明した手順に従って、Tableau Resource Monitoring Tool を新しくインストールする必要があります。

アップグレードのベストプラクティス

バージョン 2022.3 以降にアップグレードし、同時に外部メッセージ キュー サービスを使用するように移行する場合の一般的な手順は次のとおりです。

環境の再構築による移行:

1. RMT サーバーとすべてのエージェントを 2022.3 以降にアップグレードします
2. Amazon AMQ ブローカー サービスを作成します。
3. 外部メッセージ キュー サービスを使用するように RMT サーバーを設定します
4. 環境を再構築し、すべてのエージェントを再登録します
5. カスタム構成を再作成します。

注: 一部のイベントやハードウェア処理データが失われ、インシデントのしきい値を再設定する必要があります。

RabbitMQ の TLS を有効にするアップグレード ステップ

2022.3 より前のバージョンのエージェントは、RabbitMQ メッセージ キュー サービスへの暗号化されていない接続を使用して通信していたため、バージョン 2022.3 以降にアップグレードする場合は、それらのエージェントを更新して、新しい安全な接続の詳細を使用する必要があります。これを行うには、次のステップに従います。

1. 上記のセクションで説明したアップグレード ステップを完了したら、次のコマンドを実行してすべてのエージェントを停止させます。

   ```
   rmtadmin stop --agent
   ```

2. 次のコマンドを実行して、使用する環境のブートストラップ ファイルをダウンロードします。

   ```
   rmtadmin bootstrap-file --env<myenvironment> --filename <The absolute or relative path including the file name>
   ```

3. エージェントがインストールされている各マシンで、次のコマンドを実行します。

   ```
   rmtadmin rotate-mq-certificate <BOOTSTRAP_FILE> --username=<RMT Server Username> --password-file=<RMT Server Password file name>
   ```

4. `rmtadmin rotate-mq-certificate` コマンドを正常に実行したら、各エージェント マシンを再起動します。

製品の互換性

外部メッセージ キューはバージョン 2022.3 以降からしか利用できないため、この表では RMT バージョン 2022.3 以降のみを掲載しています。

RMT バージョン	RMT に同梱されている RabbitMQ のバージョン	外部メッセージ キュー サービスに対応している RabbitMQ のバージョン
22.3	3.10.5	3.10.5

実行可能なユーザー

Resource Monitoring Tool をインストールするには、次のものがすべて必要です。

- sudo へのフル アクセス権を持つユーザー アカウント
- Tableau Server の管理者サイト ロール
- Resource Monitoring Tool 管理者アカウント。

Tableau Resource Monitoring Tool 前提条件 - ライセンス

Tableau Resource Monitoring Tool には、以下のオープン ソース アプリケーションが含まれています。

RMT バージョン	OPENSSL バージョン	ERLANG バージョン	RABBITMQ バージョン	PostgreSQL バージョン
2022.1	1.1.1l	24.1.2	3.9.7	13.3
2022.1.1	1.1.1n	24.1.2	3.9.7	13.3
2022.1.5	1.1.1q	24.3.4.2	3.10.5	13.6
2022.2	1.1.1l	24.3.0	3.9.13	13.3
2022.3 - 2023.1	1.1.1q	24.3.4.2	3.10.5	13.7
2023.2 - 2024.3	1.1.1t	24.3.4.10	3.10.20	13.7
2025.1	3.1.5.20240229235146	24.0	3.8.16	12.6

- **Erlang:** Copyright 2016 Industrial Erlang User Group、Apache 2.0詳細については、「Erlang」および「Apache 2.0」のサイトを参照してください。
- **RabbitMQ:** Copyright、MPL 2.0、Copyright (c) 2007-2021 VMware, Inc. or its affiliates 詳細については、「Mozilla Public License」のサイトを参照してください。

  Resource Monitoring Tool には、リクエストに応じて利用できる、単一の変更された RabbitMQ ソースファイルが含まれています。
- **PostgreSQL:** Copyright Portions Copyright © 1996-2021、The PostgreSQL Global Development Group、Portions Copyright © 1994、The Regents of the University of California、PostgreSQL ライセンス詳細については、「PostgreSQL」のサイトを参照してください。
- **OpenSSL:** Copyright (c) 1998-2019 The OpenSSL Project、Copyright (c) 1995-1998 Eric A. Young、Tim J. Hudson、OpenSSL ライセンス、OpenSSL ライセンスと元の SSLeay ライセンスの基づくダブル ライセンス詳細については、「Open SSL ライセンス」を参照してください。

Resource Monitoring Tool のアップグレード

次の手順を実行して、Tableau Resource Monitoring Tool の既存のインストールをアップグレードします。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

> **注:** Resource Monitoring Tool は、インプレース アップグレードを実行し、現在のインストールを新しいバージョンにアップグレードします。アップグレードの**前に**既存のインストールをアンインストールしないでください。

アップグレードに関する注意

アップグレード プロセスを開始する前に、次の点を考慮してください。

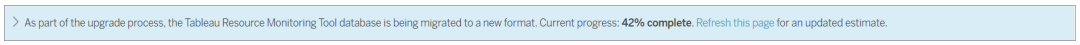
1. 外部リポジトリ(PostgreSQL) または外部メッセージ キュー サービス (RabbitMQ) の使用に移行する場合は、以下のトピックにあるアップグレードに関するベストプラクティスのセクションを必ず確認してください。
   - 外部リポジトリ用 アップグレードのベストプラクティス
   - 外部メッセージ キューのアップグレードのベストプラクティス

2. バージョン 2022.3 より、Tableau Resource Monitoring Tool には、RMT サーバーとエージェント間の暗号化通信が組み込まれています。ただし、TLS を有効にするには、すべてのエージェントで `rmtadmin rotate-mq-certificates` を実行する必要があります。RMT サーバーと同じマシンでローカルに構成されている RabbitMQ が対象となります。この方法の詳細については、「**RabbitMQ の TLS を有効にするアップグレードのステップ**」を参照してください。

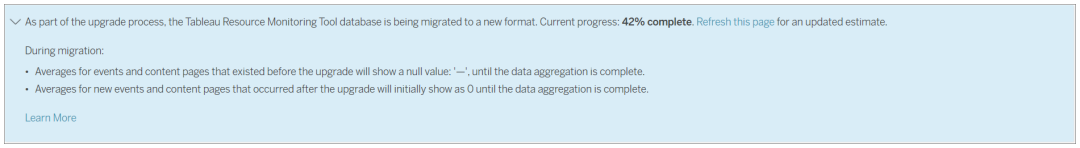
3. アップグレード中にマシンの再起動が必要になる場合があります。これは通常、ファイルがロックされていて、インストーラーによって更新できない場合に発生します。必要に応じて、

インストーラーからプロンプトが表示されます。そのため、アップグレードは勤務時間外に行うことを検討してください。

4. 場合によっては、Resource Monitoring Tool でデータベースが変更されます。そのような場合は、アップグレードプロセスにデータベースの移行が含まれる可能性があります。データベースの移行を行う場合、RMT サーバーの Web ページの上部に、プロセスを完了するための時間の見積もりと共にメッセージ バナーが表示されます。スクリーンショットの例を次に示します。



バナーを展開すると、アップグレードの影響の詳細を確認できます。変更は各リリースに固有のものであり、表示される詳細はリリースに固有のものです。



5. Resource Monitoring Tool は、Tableau Server バージョンと下位互換性がありますが、上位互換性はありません。つまりResource Monitoring Tool のバージョンは、監視している Tableau Server のバージョン以降である必要があります。

   互換性の理由から、最初に Resource Monitoring Tool をアップグレードしてから、Tableau Server のアップグレードを行うことをお勧めします。詳細については、Tableau Server との製品互換性を参照してください。

Resource Monitoring Tool のアップグレード方法

Resource Monitoring Tool のインプレース アップグレードを行うことをお勧めします。Resource Monitoring Tool をアンインストールして新しいバージョンのデータベースをインストールすると、データが破損する可能性があるため、前のバージョンを再インストールしてアップグレードプロセスをもう一度行うよう促されます。

1. アップグレードを開始する前に、ブラウザーで Resource Monitoring Tool にログインし、動作していることを確認します。

2. 新しいバージョンの RMT サーバー パッケージを RMT サーバー マシンにコピーします。

3. RMT エージェントがインストールされているマシンに新しいバージョンの RMT エージェント パッケージをコピーします。RMT エージェントは、監視している Tableau Server ノードにインストールされます。

4. RMT エージェント サービスを実行している各 Tableau Server ノードで、tabrmt-agent ユーザーに切り替えて、RMT エージェント サービスを停止します。

   ```
   sudo su --login tabrmt-agent
   rmtadmin stop
   exit
   ```

5. RMT サーバーで tabrmt-master ユーザーに切り替えて、次のコマンドを実行して RMT サーバー サービスを停止します。

   ```
   sudo su --login tabrmt-master
   rmtadmin stop
   exit
   ```

6. 名前に tabrmt が含まれる実行中のプロセスがなくなるまで待機してから、次のステップに進みます。

   注: サービスが停止されたら、ベストプラクティスとして、サービスの停止後に tabrmt-agent か tabrmt-master のいずれかを使用した Resource Monitoring Tool プロセスが実行されていないかどうかを確認してください。これには、PostgreSQL や RabbitMQ は含まれません。次のコマンドを使用するとステータスを確認できます。

   ```
   rmtadmin status
   ```

7. RMT サーバーでコマンドのアップグレードを実行します。これにより、既存のバージョンが新しいバージョンにアップグレードされます。

   CentOS を含む、RHEL のようなディストリビューションの場合:

```
sudo yum install <pathtomasterserverinstaller>/<tabrmt-master-setup-<version>-x86_64.rpm>

sudo /opt/tableau/tabrmt/master/install-scripts/upgrade-rmt-master --accepteula
```

Ubuntu ディストリビューションの場合:

```
sudo apt install <pathtomasterserverinstaller>/<tabrmt-master-setup-<version>-amd_64.deb>

sudo /opt/tableau/tabrmt/master/install-scripts/upgrade-rmt-master --accepteula
```

8. RMT サーバーがアップグレードされたら、次のコマンドを実行して、すべての RMT エージェントをアップグレードします。

   CentOS を含む、RHEL のようなディストリビューションの場合:

   ```
   sudo yum install <pathtoagentinstaller>/<tabrmt-agent-setup-<version>-x86_64.rpm>

   sudo /opt/tableau/tabrmt/agent/install-scripts/upgrade-rmt-agent --accepteula
   ```

   Ubuntu ディストリビューションの場合:

   ```
   sudo apt install <pathtoagentinstaller>/<tabrmt-agent-setup-<version>-amd_64.deb>

   sudo /opt/tableau/tabrmt/agent/install-scripts/upgrade-rmt-agent --accepteula
   ```

9. RMT サーバーとエージェントが実行中であることを確認します。アップグレードの完了後に RMT サーバーとエージェントが自動的に再起動しない場合は、再起動してください。

10. RMT エージェントがアップグレードされたことを確認するには、ブラウザーで Resource Monitoring Tool にログインしてから**Admin** メニューに移動し、**Environments**を選択します。**Edit Environment** アイコンをクリックして、環境の詳細を表示します。**Servers** タブに、RMT エージェントのバージョンが表示されます。これは、複数ノードの Tableau Server クラスターを使用している場合に、どの RMT エージェントがアップグレードされているかを特定する場合に役立ちます。

RabbitMQ の TLS を有効にするアップグレードステップ

2022.3 より前のバージョンのエージェントは、RabbitMQ メッセージ キュー サービスへの暗号化されていない接続を使用して通信していたため、バージョン 2022.3 以降にアップグレードする場合は、それらのエージェントを更新して、新しい安全な接続の詳細を使用する必要があります。これを行うには、次のステップに従います。

1. 上記のセクションで説明したアップグレードステップを完了したら、次のコマンドを実行してすべてのエージェントを停止させます。

   ```
   rmtadmin stop --agent
   ```

2. 次のコマンドを実行して、使用する環境のブートストラップ ファイルをダウンロードします。

   ```
   rmtadmin bootstrap-file --env<myenvironment> --filename <The
   absolute or relative path including the file name>
   ```

3. エージェントがインストールされている各マシンで、次のコマンドを実行します。

   ```
   rmtadmin rotate-mq-certificate <BOOTSTRAP_FILE> --username=<RMT
   Server Username> --password-file=<RMT Server Password file
   name>
   ```

4. `rmtadmin rotate-mq-certificate` コマンドを正常に実行したら、各エージェントマシンを再起動します。

実行可能なユーザー

Resource Monitoring Tool をアップグレードするには、次のパーミッションが必要です。

- sudo へのフル アクセス権を持つユーザー アカウント
- Tableau Server 管理者。
- Resource Monitoring Tool 管理者。

Resource Monitoring Tool のアンインストール

Resource Monitoring Tool がサポートする「アンインストール」のシナリオは、主に 2 つあります。

- **Resource Monitoring Tool をアンインストール:** Resource Monitoring Tool をアンインストールには、`remove` コマンドを使用します。このコマンドを実行しているコンピュータから各 Tableau Resource Monitoring Tool サービスが削除されます。データや Resource Monitoring Tool ユーザーのアカウントとグループは削除されますが、設定ファイル、ログ、バックアップ ファイルは、`/opt/tableau/tabrmt/data` フォルダーの下の一時ディレクトに移動し、保持されます。

- **Resource Monitoring Tool を削除:** Resource Monitoring Tool をコンピューターから完全に削除したい場合は、Tableau 提供のスクリプトを使用すると、Resource Monitoring Tool とすべての関連ファイルを削除できます。*これにより、すべてのデータと Resource Monitoring Tool コンポーネントが削除されるので、コンピューターを Tableau 導入前の状態にリセットしたい場合にだけ行ってください。*テクニカルサポートがインストールに関する問題のトラブルシューティングを行う際に、このステップが推奨される場合は実行する必要があるかもしれません。事前に他のバージョンをアンインストールすることなく、Resource Monitoring Tool を完全に削除します。スクリプトはコンピューターに存在するすべての既存のバージョンをアンインストールします。既存のバージョンをすでにアンインストール済で、Tableau を完全に削除したい場合、そうするためのスクリプトを一時的な場所で検索可能です。

`remove` を使用して Resource Monitoring Tool をアンインストール:

- CentOS を含む、RHEL のようなディストリビューションの場合:

```
sudo yum remove tabrmt-master

sudo yum remove tabrmt-agent
```

- Ubuntu ディストリビューションの場合:

```
sudo apt remove tabrmt-master
```

```
sudo apt remove tabrmt-agent
```

tableau-rmt-obliterate スクリプトを使用して Resource Monitoring Tool を削除:

remove コマンドを使用してアンインストールするときに問題が発生した場合は、Tableau 提供の削除スクリプトを使用すると、コンピューターからすべてのインストール ファイルを削除できます。デフォルトでは、削除スクリプトは、/opt/tableau/tabrmt/master/install-scripts/tableau-rmt-obliterate (RMT サーバー マシン上) および /opt/tableau/tabrmt/agent/install-scripts/tableau-rmt-obliterate (エージェント マシン上) にあります。

remove コマンドで Resource Monitoring Tool をアンインストールしようとすると、削除スクリプトは自動的に /var/tmp/tableau-rmt-obliterate にコピーされます。

このスクリプトを実行するための情報を以下に示します。

1. root ユーザーとして、RMT サーバー上で次のコマンドを実行し、RMT サーバーを完全にアンインストールします。

   ```
   /var/tmp/tableau-rmt-obliterate -m -y -y -y
   ```

2. root ユーザーとして、各エージェント マシン上で次のコマンドを実行し、エージェントを完全にアンインストールします。

   ```
   /var/tmp/tableau-rmt-obliterate -a -y -y -y
   ```

次のパラメーターを使用すると、ログとバックアップを含むすべてのファイルを強制的に削除できます。

-y

必須。

このコンピューターから Resource Monitoring Tool を削除します。確定するには、3 回指定する (-y-y-y) 必要があります。

`-m`

RMT サーバーをアンインストールするために必要です。

RMT サーバーを削除します。

`-a`

エージェントをアンインストールするために必要です。

Resource Monitoring Tool エージェントがインストールされている場合は、このエージェントを削除します。

`-k`

任意の項目です。

バックアップを `logs-temp` ディレクトリにコピーしません。

`-g`

任意の項目です。

ログを `logs-temp` ディレクトリにコピーしません。

実行可能なユーザー

Resource Monitoring Tool をアンインストールするには、次のものがすべて必要です。

- Resource Monitoring Tool をインストールするマシンでの管理者パーミッション。
- Tableau Server の管理者サイトロール
- Resource Monitoring Tool 管理者アカウント。

Tableau Resource Monitoring Tool の構成

このセクションには、お使いの環境に合うように Tableau Resource Monitoring Tool を構成する方法に関する情報が記載されています。

Resource Monitoring Tool サーバー構成

このトピックでは、Web インターフェイスを使用して設定可能な Resource Monitoring Tool サーバー (RMT サーバー) の構成オプションについて説明します。

インストール後のセットアップ構成

セットアップ中に指定した構成を更新できます。

構成を変更する 2 つの推奨方法を次に示します。

- **Web インターフェイスの使用:** RMT サーバーがインストールされているマシンで、*http://<hostname>/setup/server* に移動します。
- **コマンド ライン を使用してこれを行うには、**`rmtadmin master-setup` コマンドを `--skip-admin-creation` オプションとともに使用して、インストール後に管理ユーザーを作成するよう要求するプロンプトが表示されないようにします。詳細については、rmtadmin master-setupを参照してください。

注: 構成値は、構成ファイルに保存されます。このファイルには直接変更を加えることができますが、構成オプションは、UI で `rmtadmin` コマンド ライン ユーティリティを介して利用することをお勧めします。構成ファイルへの変更を適用するには、再起動する必要があります。

ここでは、初期セットアップ後に実行できる更新の種類の例をいくつか示します。

- SSL 構成の変更: 既定の構成は false に設定されています。この構成を変更して HTTPS を要求することができます。この設定を更新するには、rmtadmin master-setup コマンドを使用します。
- SSL 証明書の更新: セキュリティで保護された通信の既定の証明書モードを使用して最初の RMT サーバーのセットアップを完了していて、独自の証明書を使用して証明書を更新する場合があります。

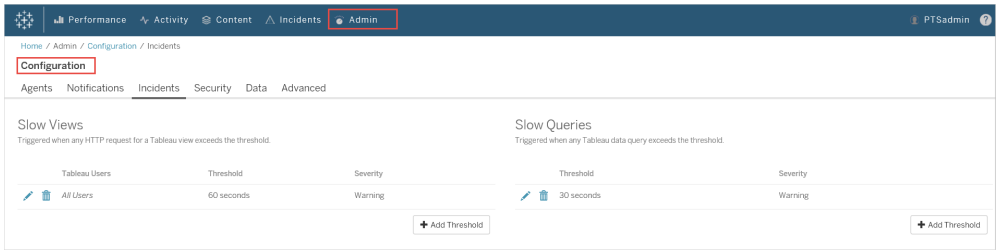
- ポートの変更: ネットワーク要件に基づいて、既定とは異なるポート番号を使用する必要がある場合があります。Resource Monitoring Tool の各種コンポーネント間のポートと通信の詳細については、Tableau Resource Monitoring Tool の通信ポートを参照してください。

- セッションの有効期限の更新: 設定されたタイムアウト期間中にユーザーがセッションにアクセスしない場合は、セッションは期限切れになるため、もう一度ログインする必要があります。デフォルトでは、240 分に設定されています。**[スライド有効期限]** オプションを有効にして、タイムアウト期間内にセッションにアクセスしたときにタイムアウト期間をリセットすることもできます。

> **注:** Tableau Server ノードの追加または削除: Resource Monitoring Tool を使用して監視している Tableau Server クラスタに新しいノードを追加する場合は、このノードをインストールして登録する必要があります。詳細については、Tableau Server のトポロジの変更を参照してください。

コマンドラインでこれを行うには、`rmtadmin-master-setup` コマンドを使用します。詳細については、rmtadmin コマンドライン ユーティリティを参照してください。

Resource Monitoring Tool は、[管理] メニューにある [構成] をクリックして構成できます。[構成] セクションはタブで分割されます。構成ファイルを使用してこれを実行するには、RMT サーバー構成ファイルを参照してください。



通知

通知は、グローバル レベルおよび環境レベルで設定できます。

通知の受信方法を選択できます。メールまたは Slack、あるいはその両方を選択できます。

以下は、メール通知とSlack 通知の両方に適用されます。

メールとSlack の通知は、いずれもグローバル レベルと環境レベルで設定できます。通知を送信するには、環境レベルだけでなく、グローバル レベルでも設定を有効にする必要があります。通知を送信する最小の重大度レベルを設定することもできます。このレベルは、メールとSlack で異なる設定にすることも、グローバル レベルと環境レベルで異なる設定にすることもできます。

グローバル構成は、その環境がカスタムしきい値を使用していない限り、既存の環境または作成された任意の新しい環境に適用されます。

Slack 通知の設定

Slack 通知は Resource Monitoring Tool バージョン 2022.1 で導入されました。

Resource Monitoring Tool で Slack 通知を設定する**前に**、Slack で行う必要のある初期設定がいくつかあります。詳細については この Slack の記事 で紹介されていますが、実行する必要がある主なことは次のとおりです。

1. Slack ワークスペースまたはアプリを新しく作成するか、既存のものを使用して、着信 Webhook を受信できるようになっていることを確認します。
2. 通知を投稿するチャネルを承認します。
3. ワークスペース用に生成された Webhook URL をコピーします。

Resource Monitoring Tool で Slack 通知を設定するには次を実行します。

1. **[管理]** メニューから、**[グローバル構成]** を選択します。
2. **[通知]** タブの **[Slack]** セクションで、セクションの上部にある切り替えボタンを使用して、Slack 通知を有効にします。
3. 次の設定を構成します。

1. **最小重大度**: Slack 通知を受信する最小重大度レベル。
2. **Webhook URL**: Slack ワークスペースの着信 Webhook の URL。この URL は、Slack ワークスペースの着信 Webhook を有効にすると自動的に生成されます。

4. **[テスト メッセージの送信]** ボタンを使用してテストし、設定が有効で機能していることを確認します。

各環境の通知を指定するには次を実行します。

1. **[管理]** メニューから変更する環境を選択し、**[環境の編集]** を選択します。
2. **[通知]** タブで、グローバル設定を上書きする構成タイプとして **[カスタム]** を選択します。
3. 最小重大度とWebhook URL を設定します。
4. **[テスト メッセージの送信]** ボタンを使用してテストし、設定が有効で機能していることを確認します。

メール通知

メール通知を設定するには、まずグローバル レベルで SMTP サーバーの設定を行います。グローバル レベルで構成された SMTP サーバーの設定は、すべての環境に自動的に適用されます。各環境の最小重大度レベルや送信者と受信者のメールなど、特定の設定をカスタマイズできます。

TLS の構成のしくみについて、技術的な詳細を説明します。

- RMT サーバーは STARTTLS SMTP オプションを使用します。これは、SMTP 接続が開始された後、かつ、メール コンテンツが送信される前に SMTP 接続を TLS にアップグレードするものです。RMT では、REQUIRETLS SMTP オプションはサポートされていません。
- RMT サーバーは、SMTP サーバーへの送信時に STARTTLS を使用してメールを暗号化します。SMTP サーバーが静止時にメールを暗号化して保存するかどうかは、SMTP サーバーの構成に依存します

Resource Monitoring Tool でメール通知を設定する前に、SMTP サーバーをセットアップし、次の情報を用意する必要があります。

- SMTP サーバーの**名前**。
- 暗号化された通信を使用する場合は、SMTP サーバーが受け入れる**TLS バージョン**をメモしておいてください。現在、Resource Monitoring Tool では **TLS 1.2** が要求されており、**TLS 1.3** もサポートしています。
- **証明書のサムプリント**(オプション)。
- SMTP サーバーの**ポート番号**
- **ユーザー名とパスワード**(オプション)。SMTP サーバーがユーザー名とパスワードを使用して認証するように構成されている場合にのみ必要です。
- 通知の送受信に使用される**送信者と受信者のメール アドレス**。

Resource Monitoring Toolで SMTP サーバーの設定を指定するには、次を実施します。

1. [管理] メニューから、**[グローバル構成]** を選択します。
2. **[通知]** タブの **[メール]** セクションで、セクションの上部にある切り替えボタンを選択し、メール通知を有効にします。
    1. 次の SMTP サーバー設定を構成します。
        1. **サーバー:** SMTP サーバーの完全修飾 DNS 名を指定します。
        2. **暗号化:** RMT サーバーとSMTP サーバー間の通信を暗号化するかどうかを指定します。選択するオプションは、暗号化用の SMTP サーバー構成と、RTM サーバーとSMTP サーバー間で暗号化された通信を使用するための設定によって異なります。
            1. **必須:** SMTP サーバーで暗号化された通信の使用が有効になっていて、通信が常に暗号化されていることを確認する場合に使用します。必須に設定すると、接続は常に暗号化されて試行されます。SMTP サーバーで暗号化の使用が有効になっていない場合、RMT サーバーは SMTP サーバーと通信できません。

2. **推奨:** SMTP サーバーの暗号化設定がわからない場合はこれを使用しますが、可能な場合は暗号化された通信を使用することをお勧めします。SMTP サーバーで暗号化の使用が有効になっていない場合、暗号化されていない通信が行われます。

3. **無効:** SMTP で暗号化通信の使用が有効になっていない場合に使用します。RMT サーバーとSMTP サーバー間の通信は暗号化されません。

   > **注:** SMTP サーバーで暗号化を必要としている場合、接続は失敗します。SMTP サーバーで暗号化が有効になっているが、暗号化が必要としていない場合、接続は成功します。

4. **オプション:** SMTP サーバーを検証する方法を決定します。次のオプションがあります。

   1. サーバーID を確認する: RMT は、使用されている証明書の名前が SMTP サーバー名と一致するかどうかを確認します。一致するものがない場合、接続は失敗します。

   2. すべてのホストを信頼する: すべての証明書エラーは無視され、チェックするサーバーの ID を上書きします。これを使用するのは、RMT サーバーが使用している証明書が、ご利用のサーバーから提供されたものであると確信できる場合のみにしてください。

5. **TLS バージョン:** SMTP サーバーでサポートされている TLS のバージョン。デフォルトバージョンがすでに選択されていますが、他のバージョンを選択することもできます。複数の TLS バージョンが選択されている場合、RMT サーバーは RMT サーバーおよび SMTP サーバーと互換性のある最も安全なバージョンを使用します。TLS バージョン 1.3 には、Open SSL 1.1.1f が必要です。TLS 1.3 の使用を計画している場合は、RMT サーバーがインストールされているマシンに Open SSL 1.1.1f

があることを確認してください。

6. **証明書のサムプリント**: これはオプションです。SHA1 証明書が使用されるのは、提供されたときに有効であり、SMTP サーバーが使用するものである場合です。有効なサムプリントは、有効期限切れやサーバー名の不一致などの他の証明書エラーを上書きします。

7. **ポート**: メール サーバーのポート設定。

8. **ユーザー名:** オプション。SMTP サーバーが認証用のユーザー名とパスワードで構成されている場合に、メール サーバーへの認証に使用されるアカウントの名前。

9. **パスワード:** オプション。SMTP サーバーが認証用のユーザー名とパスワードで構成されている場合に、メール サーバーへの認証に使用されるアカウントのパスワード。

10. **最小重大度**: メール通知を受信する最小重大度レベル。これは、環境レベルで変更されない限り、すべての環境に適用されます。

11. **送信者のメール**: 通知の送信元として使用するメール アドレス。これは、環境レベルで変更されない限り、すべての環境に適用されます。

12. **受信者のメール**: 通知を受信する人たちのメール アドレス。これは、環境レベルで変更されない限り、すべての環境に適用されます。

13. [テスト メッセージの送信] ボタンを使用して**テスト**し、設定が有効であり、メール通知を正常に送信できることを確認します。

環境設定をカスタマイズするには、次を実施します。

1. **[管理]** メニューから変更する環境を選択し、**[環境の編集]** を選択します。
2. **[通知]** タブで、グローバル設定を上書きする構成タイプとして **[カスタム]** を選択します。
3. 重大度レベル、送信者と受信者のメール情報を設定します。

4. [テストメッセージの送信] ボタンを使用してテストし、設定が有効であり、メール通知を正常に送信できることを確認します。

接続障害のトラブルシューティング

接続エラーはさまざまな理由で発生する可能性がありますが、設定の問題が原因で発生する可能性のあるものを次に示します。

- **暗号化設定エラー:** RMT サーバーとSMTP サーバーの間で暗号化設定の不一致がある場合に発生します。たとえば、RMT サーバーが暗号化を要求するように設定されているが、SMTP サーバーが暗号化を使用するように構成されていない場合です。逆もあり得ます。SMTP サーバーが暗号化された通信を必要とし、RMT サーバーが暗号化を無効にするように設定されている場合、接続は失敗します。
- **証明書エラー:** 暗号化された通信を使用する場合、[**オプション**] で [**すべてのホストを明示的に信頼**] することを選択しない限り、認証局や証明書名とSMTP サーバーの一致などが考慮されます。
- **TLS エラー:** TLS バージョンのサポートは、RMT サーバーがインストールされているオペレーティング システム (OS) によって異なります。選択した TLS のバージョンが OS でサポートされているかどうかを確認してください。TLS 1.2 が必要ですが、TSL 1.3 もサポートされています。TLS 1.3 には、Open SSL 1.1.1f 以降が必要です。RMT サーバーがインストールされているマシンに Open SSL 1.1.1f があることを確認してください。

インシデントのしきい値

グローバルなインシデント通知の設定を構成します。これらのグローバルしきい値は、個別の環境で上書きされない限り、既定ですべての環境に適用されます。

使用可能なインシデントオプションとその構成方法については、「インシデント」を参照してください。

セキュリティ

Resource Monitoring Tool ユーザー アカウントのセキュリティ設定を構成します。

データ

既定では、Resource Monitoring Tool に 2 週間分の詳細なアクティビティデータおよびダッシュボードデータと、10 年分の集計済みレポートデータが格納されます。

Data Retention (データの保持): Tableau Server の詳細なアクティビティデータによって、最近のパフォーマンス インシデントを診断するために必要なダッシュボードとインシデントが提供されます。関連付けられたストレージとプロセスの要件により、このデータは既定で 2 週間分のみ格納されます。

Reporting Data (レポート データ): Tableau Server の集計済みアクティビティデータが、レポート用に保存されます (チャージバック レポートなど)。このデータはコンパクトなため、複数年分の履歴レポートを簡単に格納できます。

詳細設定

Resource Monitoring Tool の診断ロギング レベルの構成を許可します。

ロギングの詳細については、「ログ ファイル」を参照してください。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つ Resource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

RMT サーバー構成ファイル

このトピックでは、構成ファイルを使用して使用可能な構成オプションについて説明します。Web インターフェイスを使用してこれを行うには、Resource Monitoring Tool サーバー構成を参照してください。

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

構成ファイルは `/var/opt/tableau/tabrmt/master/config.json` にあります。

> **注:** 構成値は、構成ファイルに保存されます。このファイルには直接変更を加えることができますが、構成オプションは、UI で `rmtadmin` コマンドライン ユーティリティを介して利用することをお勧めします。構成ファイルへの変更を適用するには、再起動する必要があります。

構成ファイルに変更を加えた後は、RMT サーバーを再起動する必要があります。

データ保持

デフォルトでは、Resource Monitoring Tool に 2 週間分の詳細データと、10 年分の集計済みレポートデータが格納されます。

これは構成可能です。データ保持設定を定義するスニペット例を次に示します。

```
{
  "db":  {
    "cleanup":  {
      "afterDays":  14,
      "aggregate":  {
        "afterDays":  3650
      }
    }
  }
}
```

SMTP 構成

SMTP 情報を定義する `config.json` のスニペット例を次に示します。

```
{
  "smtp":  {
    "server":  "localhost",
    "port":  25,
    "username":  "my-username",
    "password":  "my-password",
    "requireSSL":  true
  }
```

```
}
```

smtp 構成セクションで使用できる構成値の完全なリストを次に示します。

キー	データ型	必須か?	説明
server	文字列	オプション	使用する SMTP サーバー。既定値: localhost
port	数値	オプション	SMTP サーバーのポート番号。既定値: 25
username	文字列	オプション	サーバーで認証が必要な場合に使用するユーザー名。
password	文字列	オプション	サーバーで認証が必要な場合に使用するパスワード。
requireSSL	ブール値	オプション	SMTP サーバーで SSL が要求されている場合は true、それ以外の場合は false。既定値: false

通知構成

通知情報を定義する config.json のスニペット例を次に示します。

```
{
  "notifications":  {
    "email":  {
      "from":  "sender@domain.com",
      "to":  "recipient1@domain.com,recipient2@domain.com"
    }
  }
}
```

グローバル通知情報 (上記のような情報) は、environments セクション内の環境ごとに上書きできます。

```
{
  "environments":  {
    "Environment1":  {
      "notifications":  {
        "email":  {
          "from":  "sender@domain.com",
          "to":  "recipient1@domain.com,recipient2@domain.com"
        }
      }
    }
  }
}
```

notifications.email 構成セクションで使用できる構成値の完全なリストを次に示します。

キー	データ型	必須か?	説明
from	文字列	必須	送信者のメール アドレス。
to	文字列	必須	受信者のメール アドレス。複数のアドレスはコンマまたはセミコロンで区切る必要があります。
minimumIncidentSeverity	文字列	オプション	メールが送信される最小のインシデント重大度。既定値: critical。Tableau Resource Monitoring Tool - インシデントも参

キー	データ型	必須か?	説明
			照してください。

ヒストグラム構成

Web インターフェイスでヒストグラムを生成するために使用される境界は、各境界を表す値の配列を使用して構成できます。

monitoring.histograms セクション内の構成可能なヒストグラムの完全なリストを次に示します。

キー	データ型	必須か?	説明
viewLoadDuration	数値の配列	オプション	ビュー読み込み用のヒストグラム境界。値はミリ秒単位で指定します。既定値: [1000, 3000, 6000, 10000]
externalDataRequestDuration	数値の配列	オプション	外部データ要求用のヒストグラム境界。値はミリ秒単位で指定します。既定値: [1000, 3000, 6000, 10000]

キー	データ型	必須か?	説明
backgroundTaskDuration	数値の配列	オプション	バックグラウンドタスク用のヒストグラム境界。値はミリ秒単位で指定します。既定値: [60000, 300000, 600000, 1800000]

たとえば、すべてのデータ用にグローバル レベルで次のヒストグラム バケットを使用するとします。

- 1 秒以下
- 1 秒より大きく10 秒以下
- 10 秒より大きく30 秒以下
- 30 秒より大きい

構成は次のようになります。

```
{
  "monitoring":  {
    "histograms":  {
      "viewLoadDuration":  [1000,  10000,  30000],
      "externalDataRequestDuration":  [1000,  10000,  30000],
      "backgroundTaskDuration":    [1000,  10000,  30000]
    }
  }
}
```

ヒストグラムの境界は、環境ごとに設定することもできます。たとえば、識別子が "staging-environment" である環境で次のビュー ヒストグラム バケットを使用するとします。

- 2.5 秒以下
- 2.5 秒より大きく5 秒以下

- 5 秒より大きく30 秒以下
- 30 秒より大きく1 分以下
- 1 分より大きく10 分以下
- 10 分より大きい

構成は次のようになります。

```
{
  "environments":  {
    "staging-environment":  {
      "monitoring":  {
        "histograms":  {
          "viewLoadDuration":  [2500,  5000,  30000,  60000,
600000]
        }
      }
    }
  }
}
```

"staging-environment" では、バックグラウンド タスク用のグローバル ヒストグラム構成が使用されます。

最小 TLS バージョン

既定で、RMT はトラフィックを暗号化するために TLS の安全なバージョンを使用します。既定の最小バージョンは 1.2 ですが、古いバージョンの TLS を無効にする必要がある特定のセキュリティ要件がある場合は、構成ファイルの `server.minimumTlsVersion` セクションを変更して、最小 TLS バージョンを強制します。以下の例のスニペットでは、最小バージョンが 1.3 に設定されています。`minimumTlsVersion` の有効な値のリストは「.Net Core の SslProtocols」で定義されています。

```
"server":  {
  "url":  "https://rmtserver:443",
  "https":  {
    "enforce":  true,
    "certificate":  {
      "mode":  Default,
```

```
      "local":  {
        "name":  null,
        "password":  null
       },
      "store":  {
        "certificateThumbprint":  ""}
     },
     "minimumTlsVersion":  "Tls13"
   },
```

インシデント構成

使用可能なインシデントオプションとその構成方法については、「インシデント」を参照してください。

RMT ServerLogging

Tableau Resource Monitoring Tool ログ ファイルを参照してください。

エージェント

エージェントサービスの構成ファイルは /var/opt/tableau/tabrmt/master/config.json にあります

構成ファイルに変更を加えた後は、エージェントサービスを再起動する必要があります。

Tableau Server の検出

ほぼすべての状況で エージェントは Tableau Server インストールを自動的に検出するため、標準のエージェント セットアップ プロセス以外の構成は必要ありません。ただし、必要に応じて config.json ファイルを使用して Tableau Server 情報を手動で設定することもできます。

エージェントの実行に必要な Tableau Server 情報を定義する config.json のスニペット例を次に示します。

```
{
  "agent":  {
    "tableauServer":  {
      "override":  true,
```

```
      "productVersion":  2021.4,
      "applicationDirectory":  "/var/opt/tableau/tableau_
server/2021.4"
    }
  }
}
```

agent.tableauServer 構成セクションで使用できる構成値の完全なリストを次に示します。

キー	データ型	必須か?	説明
applicationDirectory	文字列	必須	Tableau Server アプリケーション ルート ディレクトリ。
dataDirectory	文字列	オプション	Tableau Server データ ファイルを保持するディレクトリ。
override	ブール値	オプション	true の場合、自動的に検出された値より構成値が優先されます。 false の場合、構成値が自動的に検出された値のフォールバックとして機能します。
productVersion	数値	必須	Tableau Server のバージョン番号。

エージェント ログ

Tableau Resource Monitoring Tool ログ ファイルを参照してください。

共通

共通の構成値は、RMT サーバー アプリケーションとエージェント アプリケーションの両方で使用できます。構成ファイルを見つける方法については、アプリケーション固有のセクションを参照してください。

構成ファイルに変更を加えた後は、アプリケーション サービスを再起動する必要があります。

暗号化されたメッセージング

暗号化されたメッセージングを有効にするには、まず、TLS を許可するように RabbitMQ サーバーを構成する必要があります。詳細については、「暗号化されたデータの収集」管理ガイドを参照してください。

RMT サーバーやエージェントを構成して、暗号化されたメッセージングを有効にする場合は、次のことに注意してください。

- 暗号化を有効にするには、`enabled` フラグと`certificateHostName` の両方が構成されている必要があります。
- `certificateHostName` 変数は、サーバー証明書上の正規名 (CN=) と一致する必要があります。そうでないと接続に失敗します。
- `mq` セクションの `port` 設定は、RabbitMQ を構成した TLS ポートに基づいて変更しなければならない可能性があります。

```
{
  "mq":  {
    "port":  5671,
    "tls":  {
      "enabled":  true,
      "certificateHostName":  "foo"
    }
  }
}
```

`mq.tls` 構成セクションで使用できる構成値の完全なリストを次に示します。

キー	データ型	必須か?	説明
enabled	ブール値	オプション	true の場合、メッセージング接続用に TLS 暗号化が有効になります。false の場合、メッセージング用に暗号化されていない接続が使用されます。既定値: false
certificateHostName	文字列	オプション	certificateHostName は、サーバー証明書の正規名 (CN=) と一致する必要があります。そうでないと接続に失敗します。既定値: ""

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つ Resource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

rmtadmin コマンド ライン ユーティリティ

Resource Monitoring Tool には、コマンド ライン インターフェイス、rmtadmin.cmd がインストール フォルダーに含まれています。デフォルトのインストール フォルダーは次の通りです。

- RMT サーバー: /var/opt/tableau/tabrmt/master/.
- エージェント: /var/opt/tableau/tabrmt/agent/

  rmtadmin は **RMT サーバーとエージェント**の両方のインストールに含まれています。コマンドの中には、RMT サーバーまたはエージェントのどちらの rmtadmin コマンド ライン ユーティリティを使用しているかによって変わるものもあります。たとえば、users コマンドは *RMT サーバー*のみで使用できます。ziplogs コマンドはどこでも使用できますが、コマンドを実行しているアプリケーションのログ ファイルのみを含めます。

注: 以下のコマンドは、tabrmt-master ユーザーとして実行する必要があります。

```
sudo su --login tabrmt-master
```

rmtadmin コマンドラインで使用できるコマンドは、次のとおりです。

注: rmtadmin コマンドは、位置パラメーターとオプションの両方を使用します。
位置パラメーターは、値のみを使用して指定する必要があります。実際のキーワードを指定する必要はありません。オプションのキーワードと値は、等号を使用して指定する必要があります。
例:

```
rmtadmin <command> <positional parameter value> --<option keyword>=<value>
```

- rmtadmin add-user
- rmtadmin agents
- rmtadmin bootstrap-file
- rmtadmin cleanup
- rmtadmin create-admin-user
- rmtadmin create-env
- rmtadmin delete-env
- rmtadmin data-access
- rmtadmin delete-env-data
- rmtadmin delete-server
- rmtadmin delete-server-data
- rmtadmin delete-user
- rmtadmin deregister-agent
- rmtadmin environments
- rmtadmin get
- rmtadmin get-user
- rmtadmin help
- rmtadmin master-setup
- rmtadmin passwd
- rmtadmin query
- rmtadmin register
- rmtadmin restart
- rmtadmin rotate-mq-certificate
- rmtadmin rotate-mq-certificates
- rmtadmin servers

- rmtadmin set
- rmtadmin start
- rmtadmin stop
- rmtadmin status
- rmtadmin test-env
- rmtadmin update-baseline
- rmtadmin update-env
- rmtadmin users
- rmtadmin version
- rmtadmin ziplogs

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます、詳細については、「Tableau ヘルプについて」を参照してください。

rmtadmin add-user

注: バージョン 2024.2 で追加。

指定されたユーザー名を持つユーザー アカウントを作成します (存在しない場合)。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin add-user [options][global option]
```

オプション

`--username`

追加する新しいユーザー アカウントのユーザー名。

`--auth-mode`

新しいユーザー アカウントの認証モード。値は [Local] または [Delegated] にすることができます。

--password

ユーザー アカウントのパスワード。認証モードが [Local] の場合にのみ必要です。

--password-file

ユーザー アカウントのパスワードを含むファイルへのパス。これは、パスワードを直接入力せずにアクセス可能なファイルにパスワードを保存する場合に使用できます。認証モードがローカルの場合にのみ必要です。

ロール

--download-log-bundles

任意の項目です。ログ バンドルをダウンロードする権限をユーザーに付与します。

--download-tds-files

任意の項目です。TDS ファイルをダウンロードする権限をユーザーに付与します。

--generate-chargeback-report

任意の項目です。チャージバック レポートを生成する権限をユーザーに付与します。

--environment-management

任意の項目です。環境を管理する権限をユーザーに付与します。

--user-management

任意の項目です。他のユーザーを管理する権限をユーザーに付与します。

--webhook-management

任意の項目です。Webhook を管理する権限をユーザーに付与します。

例:

```
rmtadmin add-user --username=<username> --password=<password> --
auth-mode=<Local|Delegated> --download-log-bundles=[T|F] --
download-tds-files=[T|F] --generate-chargeback-report=[T|F] --
environment-management=[T|F] --user-management=[T|F] --webhook-
management=[T|F]
```

注: `DisableLocalAuthentication` 構成キーが `true` に設定されている場合、ローカル認証モードは無効になります。

rmtadmin agents

注: バージョン 2021.2 で追加

すべての環境の Tableau Server ノードに登録されているすべてのエージェントを一覧表示します。

Resource Monitoring Tool エージェントがインストールされている Tableau Server の場所を確認するのに役立ちます。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin agents [options][global option]
```

オプション

`--env`

任意の項目です。このオプションを使用すると、特定の環境のエージェントを一覧表示できます。これは、システムで生成された識別子です。これを確認するには、**Admin >**

Environments > Edit Environment の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

`--keys`

任意の項目です。出力にエージェントキーを含めます。

例:`rmtadmin agents --env=<myenvironmentidentifier> --keys`

rmtadmin bootstrap-file

注: バージョン 2021.2 で追加

Tableau Server ノードにエージェントを登録するために使用する、bootstrap ファイルを作成して保存します。

bootstrap ファイルが作成され、指定された絶対パスまたは相対パスに保存されます。相対パスは、現在の作業ディレクトリに対して解決されます。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin bootstrap-file [options][global option]
```

オプション

`--env`

必須。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。
Tableau Server ノード上のエージェントは、一度に 1 つの環境にのみ登録できます。

`--filename`

任意の項目です。ファイル名を含む絶対パスまたは相対パス。相対パスは、現在の作業ディレクトリに対して解決されます。指定しない場合、デフォルトの出力ファイル名が使用されます。

`--force`

任意の項目です。同じ名前の既存のファイルを上書きします。

例: `rmtadmin bootstrap-file --env=<myenvironmentidentifier>`

rmtadmin cleanup

注: バージョン 2021.2 で追加

特定の日数より古いログファイルを削除します。デフォルトでは、31 日より古いログファイルが削除されます。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

注: Resource Monitoring Tool プロセスが実行されている場合、アクティブなログ ファイルはこのコマンドによってクリーンアップされません。すべてのファイルが確実に削除されるようにするには、rmtadmin_stop コマンドを使用して RMT を停止してから、次のコマンド `rmtadmin cleanup --log-files-retention=0` を使用してクリーンアップを実行する必要があります。次に rmtadmin_start コマンドを使用してプロセスを再起動します。

シノプシス

```
rmtadmin cleanup [option][global option]
```

オプション

`--log-files-retention`

任意の項目です。指定した日数より古いログ ファイルを削除します。

例:`rmtadmin cleanup --log-files-retention=<number of days>`

rmtadmin create-admin-user

バージョン: 2022.1 で追加されました

他のユーザーが存在しない場合は、初期管理ユーザーを作成します。

注: バージョン 2023.1 以降では、複数の管理ユーザーを作成することができます。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin create-admin-user [options][global option]
```

オプション

`--username`

任意の項目です。管理者ユーザー アカウントのユーザー名。指定しない場合は既定で "admin" に設定されます。

`--password`

必須。管理者ユーザー アカウントのパスワード。

`--password-file`

任意の項目です。管理者ユーザー アカウントのパスワードを含むファイル パス。これは、パスワードを直接入力せずにアクセス可能なファイルにパスワードを保存する場合に使用できます。

例:`rmtadmin create-admin-user --username<admin user name> --password <password for the admin user account>`

rmtadmin create-env

注: バージョン 2021.2 で追加

新しい環境を作成します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin create-env [options][global option]
```

オプション

オプション	必須か?	既定	説明
--name	はい	N/A	環境の名前。
--gateway-url	はい	N/A	Tableau Server ゲートウェイへのアクセスに使用される URL。
--version	はい	N/A	この環境が監視する Tableau Server のバージョン。
--non-interactive	いいえ	インタラクティブなプロンプトはデフォルトで有効になっています。	すべてのインタラクティブなプロンプトを無効にします。
--no-test	いいえ	API とリポジトリの接続テストはデフォルトで有効になっています。	API とリポジトリの接続テストを無効にします。

オプション	必須か?	既定	説明
--api-username	いいえ	Null	Tableau Server API への接続に使用されるアカウントのユーザー名。ユーザー アカウントは、すべての Tableau Server サイトにアクセスできる Tableau Server 管理者である必要があります。
--api-password	いいえ (Tableau API ユーザー名を指定する場合は、パスワードを指定するか、ファイル パスとパスワードを含むファイルを指定します)	Null	Tableau Server API への接続に使用される Tableau Server API ユーザー アカウントのパスワード。
--api-password-file	いいえ	Null	ファイルへのパスと、Tableau Server API ユーザー アカウントのパスワードを含むファイルの名前。
--repository-server	はい	Null	Tableau Server と共にインストールされている PostgreSQL データベースのサーバー名。
--repository-database	可	Null	Tableau Server と共にインストールされている PostgreSQL データベースの名前。
--repository-port	可	Null	Tableau Server リポジトリデータベースのポート番号。
--repository-username	可	Null	Tableau Server リポジトリと共にインストールされている PostgreSQL データベースへの接続に使用するユー

オプション	必須か?	既定	説明
			ザー名。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベース ユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password	可	Null	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザー アカウントのパスワード。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベース ユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password-file	いいえ。ただし、コマンドプロンプトまたはスクリプトで直	Null	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザー アカウン

オプション	必須か?	既定	説明
	接パスワードを入力しない場合は必要です。		トのパスワードが保存されているファイル名を含むパス。
--repository-ssl-mode	いいえ	Prefer	Tableau Server リポジトリSSL モード。 **Prefer** は SSL を優先して、**Require** は SSL を必須として、Tableau リポジトリへの SSL接続を構成します。 **Disable** は SSL を使用せずに、Tableau Server リポジトリへの接続を行います。
--repository-ssl-thumbprint	いいえ	Null	直接接続を許可するように Tableau Server PostgreSQL を構成する場合、Tableau Server は証明書とキーを作成します。Tableau Server によって生成された証明書の指紋を指定するか、**server.crt** ファイルを Resource Monitoring Tool のサーバー マシンにコピーするかを選択できます。証明書ファイルをコピーする場合は、指紋を指定する必要はありません。詳細については、クライアントからの直接接続を許可するよう Postgres SSL を構成するを参照してください。

例:`rmtadmin create-env --name=<myenvironment> --api-username=<TableauServer API user name> --api-password=<password for the Tableau Server API user account> --gateway-url <Tableau Server Gateway URL> --version <Tableau Server version>`

rmtadmin data-access

注: バージョン 2022.3 で追加

PostgreSQL データベースへのアクセスを有効化または無効化します。注: Resource Monitoring Tool が Tableau Server からすべての監視データを収集できるようにするために、バージョン 2022.3 以降では、PostgreSQL データベースにアクセスする必要があります。

このコマンドは RMT サーバーのみで機能します。

> **注:** この構成を有効にするには、PostgreSQL データベースを再起動する必要があります。

シノプシス

```
rmtadmin data-access [positional parameter][options] [global
option]
```

位置パラメーター

モード

必須。PostgreSQL データベースへのリモートデータアクセスに使用する必要があるモード。値は、*None*、*ReadOnly*、*Admin* のいずれかにする必要があります。*None* に設定すると、データアクセスが無効になります。ReadOnly とAdmin の場合、データベースへのアクセスに使用するユーザー アカウントを指定します。

オプション

--

例:

```
rmtadmin data-access ReadOnly

rmtadmin restart --db
```

rmtadmin delete-env

注: バージョン 2021.2 で追加

特定の環境と、その環境用に収集されたすべてのデータを削除します。その環境が監視している Tableau Server に関する接続およびトポロジ情報も削除し、すべての Resource Monitoring Tool エージェントの登録を解除します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin delete-env [positional parameter][options] [global option]
```

位置パラメーター

`env`

必須。これは、システムで生成された識別子です。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

オプション

`--confirm`

任意の項目です。環境を削除することを確認します。このオプションを指定しない場合、確認を要求するプロンプトが表示されます。

`--skip-agent-disconnect`

任意の項目です。エージェントの切断をスキップし、エージェントの登録解除を続行します。エージェントがアクセスできないと考えられる場合に、このオプションを使用します。たとえば、Tableau Server ノードが削除されている場合、またはエージェントがそのノードでアンインストールされている場合などです。

例:`rmtadmin delete-env <myenvironmentidentifier>`

rmtadmin delete-env-data

指定した環境で収集されたすべての Tableau Server 関連のデータが完全に削除されます。環境構成、Tableau Server 情報、エージェント登録は削除されません。

これは、環境自体を削除せずに環境内のすべての既存データをクリアする場合に便利です。既存のデータを削除した後も、エージェントによって送信された新規データは引き続き処理されます。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin delete-env-data [positional parameter][option][global
option]
```

位置パラメーター

`env`

データを削除する必要がある環境の識別子。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

オプション

`--confirm`

任意の項目です。環境データを削除することを確認します。このオプションを指定しない場合、確認を要求するプロンプトが表示されます。

例: `rmtadmin delete-env-data <myenvironmentidentifier>`

rmtadmin delete-server

注: バージョン 2021.2 で追加

Tableau Server ノードの構成情報を環境から削除し、そのノードのエージェントの登録を解除し、そのノードに固有に収集されたすべての監視データを削除します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin delete-server [options][global option]
```

オプション

`--env`

必須。Tableau Server ノードが接続されている環境の識別子。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

`--hostname`

環境に接続されている 1 つ以上の情報を削除するために必要。**-all** が指定されていない場合も必要。

Tableau Server ノードのマシン名です。複数のホスト名を指定する場合は、コンマを使用して複数の値を区切ります。

`--all`

オプションですが、**--hostname** が指定されていない場合は必須です。

環境に接続されているすべての Tableau Server ノードの情報を削除します。

`--skip-agent-disconnect`

任意の項目です。エージェントの切断をスキップし、エージェントの登録解除を続行します。エージェントにアクセスできない場合、このオプションを使用します。たとえば、ノードが Tableau Server から削除されている場合、またはエージェントがそのノードでアンインストールされている場合です。

`--confirm`

Tableau Server ノードに関するすべての情報を削除することを確認します。このオプションを指定しない場合、確認を要求するプロンプトが表示されます。

例:`rmtadmin delete-server --hostname=<machine name of the Tableau Server node>`

rmtadmin delete-server-data

注: バージョン 2021.2 で追加

1 つ以上の Tableau Server ノードから収集されたデータを削除します。指定したノードのみから収集されたデータを削除します。***all*** を指定すると、すべてのノードから収集されたすべてのデータが削除されます。環境構成、Tableau Server 情報、エージェント登録は削除されません。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin delete-server-data [options][global option]
```

オプション

`--env`

必須。Tableau Server ノードが接続されている環境の識別子。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

`--hostname`

環境に接続されている 1 つ以上の情報を削除するために必要。**-all** が指定されていない場合も必要。

Tableau Server ノードのマシン名です。複数のホスト名を指定する場合は、コンマを使用して複数の値を区切ります。

`--all`

オプションですが、**--hostname** が指定されていない場合は必須です。

環境に接続されているすべての Tableau Server ノードの監視データを削除します。

`--confirm`

Tableau Server ノードからすべての監視データを削除することを確認します。このオプションを指定しない場合、確認を要求するプロンプトが表示されます。

例: `rmtadmin delete-server-data --all`

rmtadmin delete-user

注: バージョン 2024.2 で追加。

指定されたユーザー名を持つユーザー アカウントを削除します (存在する場合)。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin delete-user [options][global option]
```

オプション

`--username`

削除するユーザーアカウントのユーザー名

例:

```
rmtadmin delete-user --username=<username>
```

rmtadmin の登録解除

注: バージョン 2021.2 で追加

環境からエージェントの登録を解除します。このノードからの監視データは収集されなくなります。既存のデータは残り、削除されません。

このコマンドは RMT エージェントでのみ機能します。登録を解除したいエージェント上でこれを実行します。

シノプシス

```
rmtadmin deregister [options]
```

オプション

`--confirm`

任意の項目です。エージェントの登録を解除することを確認します。これにより、確認プロンプトが省略されます。

`--ignore-master-errors`

任意の項目です。このプロセス中に RTM サーバーとの通信で発生するエラーは無視されます。

例: `rmtadmin deregister --confirm`

rmtadmin deregister-agent

注: バージョン 2021.2 で追加

指定されたエージェントの登録を環境から解除します。このノードからの監視データは収集されなくなります。既存のデータは残り、削除されません。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin deregister-agent [options][global option]
```

オプション

`--env`

--key が指定されていない場合は必須です。

エージェントが現在登録されている環境の識別子です。これは、システムで生成された識別子です。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

`--key`

--env が指定されている場合は省略可能です。

エージェントキーです。複数の値を区切るには、コンマを使用します。rmtadmin agents コマンドを実行すると、キー値を取得できます。

`--hostname`

任意の項目です。エージェントがインストールされている Tableau Server ノードのマシン名。複数の値を区切るには、コンマを使用します。

`--all`

任意の項目です。このオプションを使用して、すべてのノードのエージェントの登録を解除します。

`--ignore-agent-errors`

任意の項目です。このプロセス中にエージェントに接続できない場合に発生するエラーを無視します。

`--skip-agent-disconnect`

任意の項目です。エージェントの切断をスキップし、エージェントの登録解除を続行します。エージェントにアクセスできないと思われる場合は、このオプションを使用します。たとえば、ノードが Tableau Server から削除されている場合、またはエージェントがそのノードでアンインストールされている場合などです。

`--confirm`

任意の項目です。エージェントの登録を解除することを確認します。このオプションを指定しない場合、確認を要求するプロンプトが表示されます。

例:`rmtadmin deregister-agent --env=<myenvironmentidentifier> --all`

rmtadmin environments

注: バージョン 2021.2 で追加

Resource Monitoring Tool サーバー上のすべての環境を一覧表示します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin environments [global option]
```

rmtadmin get

指定した構成キーの値を取得します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin get <config.key> [global option]
```

位置パラメーター

`key`

構成キーの名前。

例： `rmtadmin get db:database`

サポートされる構成キー

キー	データ型	適用先	デフォルト	説明
auth:disableLocalAuthentication	ブール値	RMT サーバー	FALSE	TRUE の場合、認証モードをローカルとしてユーザーを作成することは許可されません。バージョン

キー	データ型	適用先	デフォルト	説明
				2024.2 で追加
auth:disableUserManagementUI	ブール値	RMT サーバー	FALSE	TRUE に設定すると、UI からのユーザー管理は許可されません。バージョン 2024.2 で追加
db:database	文字列	RMT サーバー	tabrmtd-b	RMT が使用する PostgreSQL データベースの名前。
db:readOnlyUsername	文字列	RMT サーバー	リードオンリー	RMT データへの読み取り専用アクセスのみを持つ PostgreSQL ユーザーアカウントのユーザー名。
db:readOnlyPassword	文字列	RMT サーバー	インストーラープログラムによって生成さ	PostgreSQL 読み取り専用ユーザーのパスワード。

キー	データ型	適用先	デフォルト	説明
			れます。	
monitoring:antiDomainLockoutEnabled	ブール値	RMT サーバー	TRUE	TRUE (既定) に設定すると、ドメイン ロックアウト防止機能が有効になります。バージョン 2025.1.1 で追加。この機能を使用すると、ドメインがロックされる前にサインインの試行を停止するように RMT を設定できます。
monitoring:antiDomainLockoutTimeSeconds	整数	RMT サーバー	600	Tableau Server への接続が再試行される前に、すべての試行がスロットルされる時間。バージョン 2025.1.1 で追加。

キー	データ型	適用先	デフォルト	説明
monitoring:antiDomainMaxFailedLoginAttempts	整数	RMT サーバー	3	ドメイン ロックアウト防止機能を有効にするための、Tableau Serverへの連続ログイン失敗回数のしきい値。バージョン2025.1.1で追加。
mq:tls:certificateHostName	文字列	RMT サーバーとエージェント	N/A	TLS 経由で RabbitMQ に接続するときに使用する証明書のホスト名。
mq:tls:enabled	ブール値	RMT サーバーとエージェント	FALSE	RabbitMQ への TLS 接続を有効または無効にします。
mq:port	整数	RMT サーバーとエージェン	5672	RabbitMQ ホストのポート番号。

キー	データ型	適用先	デフォルト	説明
mq:virtualHost	文字列	RMT サーバーとエージェント	tabrmt	RMT サーバーまたはエージェントへの接続。エージェントの場合、ブートストラップ ファイルにエージェントを登録することで、この設定を変更できます。RMT サーバーの場合、値はインストーラーによって管理されます。
server.web.run	ブール値	エージェント	TRUE	エージェントの Web インターフェイスを有効または無効にします。

set コマンドを使用して、現在の構成値を変更します。

rmtadmin get-user

注: バージョン 2024.2 で追加。

ユーザー アカウントが存在する場合、指定されたユーザー名のユーザー アカウント詳細をリストします。詳細には、ユーザー名、認証モード、ロール権限が含まれます。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin get-user [options][global option]
```

オプション

`--username`

ユーザー アカウントのユーザー名。

例:

```
rmtadmin get-user --username=<username>
```

rmtadmin help

コマンド ライン インターフェイスと使用可能なコマンドに関する一般的なヘルプを示します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin help
```

特定のコマンドのヘルプおよび使用状況情報を表示する場合:

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

```
rmtadmin help [command]
```

rmtadmin master-setup

指定したオプションを使用して RMT サーバーを構成します。インストール時とインストール後の両方の構成で使用します。

シノプシス

```
rmtadmin master-setup [options][global option]
```

構成オプションは以下のとおりです。

オプション	必須か?	既定	説明
--admin-password	はい (初回インストールでのみ必要) パスワードは、コマンドラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。	N/A	管理者ユーザーのパスワード

オプション	必須か?	既定	説明
--admin-password-file	はい (初回インストールでのみ必要) パスワードは、コマンドラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。	N/A	管理者ユーザーのパスワード
--admin-username	いいえ	admin	管理者ユーザーのユーザー名
--skip-admin-creation **追加:** バージョン 2020.4.0 **廃止:** バージョン 2022.3.0	なし このオプションは、インストール後に構成を更新する場合にのみ使用す	False	管理者ユーザーとパスワードの作成をスキップします。

オプション	必須か?	既定	説明
	る必要があります。 このオプションが有効なバージョン (2020.4.0 - 2022.2.x) では、インストール後の構成の更新中にこのオプションを使用しない場合、そのコマンドは失敗し、更新は行われなくなります。 バージョン 2022.3.0 以降では、このコマンドは必要に応じて管理者を作成します。		
--http-port	いいえ	80	
--require-https	いいえ	False	HTTP トラフィックを HTTPS にリダイ

オプション	必須か?	既定	説明
			レクトします。
--https-certificate-mode	いいえ	‘既定' 使用可能なオプション: • 既定 • ストア (Windows のみ) • ローカル	HTTPS 証明書に対して実行する証明書検索のタイプ **既定:** このモードでは、インストーラーによって提供される既定の自己署名証明書が使用されます。 **ストア:** このモードでは、Windows 証明書ストアに証明書の指紋を入力できます。 **ローカル:** このモードでは、**config** フォルダーにあるファイルベースの証明書を指定できます。
--https-certificate-store-thumbprint	いいえ	Null	HTTPS 証明書のハッシュ/指紋を使用して'ストア' 証明書モードで検索する
--https-certificate-local-name	いいえ	Null 注: 指定しない場合、Resource Monitoring Tool は自己署名証明書と共にインストールされ、HTTPS 通信にその証明書を使用します。	HTTPS 証明書ファイルの名前
--https-certificate-local-password	いいえ	Null	HTTPS 証明書に使用するパスワード

オプション	必須か?	既定	説明
--https-certificate-local-password-file	いいえ	Null	HTTPS 証明書に使用するパスワードを含むファイルへのパス
--confirm	いいえ	確認のプロンプトを表示します。	RMT サーバーの再起動を確認します。
--host	いいえ **注**: この値を変更することは推奨しません。	現在の構成値またはマシン名	Resource Monitoring Tool サーバーにアクセスするための優先 URL。サーバは、ホスト名に関係なくすべての IP をリッスンします。
--db-config=external	いいえ	なし。 このオプションは、リポジトリが RMT サーバーの外部で構成されている場合にのみ使用されます。	このオプションを使って、外部リポジトリを使用するように RMT サーバーを構成します。PostgreSQL データベースをローカルでインストールする場合は、このオプションを指定する必要はありません。
--db-server	いいえ	現在の構成値またはインストーラーのデフォルト値	Resource Monitoring Tool と共にインストールされている PostgreSQL データベースのサーバー名。 AWS RDS でホストされている外部リポジトリを使用している場合、この名前は RDS のインスタンス名である必要があります。

オプション	必須か?	既定	説明
--db-ssl-mode バージョン: バージョン 2023.1.0 で追加されました。	いいえ	`Prefer`	Resource Monitoring Tool と共にインストールされてた PostgreSQL データベースへの接続の SSL/TLS 暗号化が決定されます。 オプションは、`Prefer` (既定)、`VerifyCA`、および `VerifyFull` です。詳細については、Npgsql のドキュメントを参照してください。
--db-port	いいえ	現在の構成値またはインストーラーのデフォルト値	データベース サーバーのポート番号
--db-username	いいえ	現在の構成値またはインストーラーのデフォルト値	Resource Monitoring Tool と共にインストールされている PostgreSQL データベースへの接続に使用するユーザー名。 AWS RDS でホストされている外部リポジトリを使用している場合、この名前は"postgres" である必要があります。
--db-password	いいえ	現在の構成値またはインストーラーのデフォルト値	Resource Monitoring Tool と共にインストールされている PostgreSQL データベースへの接続に使用するユーザー アカウントのパスワード。 AWS RDS でホストされている外部リポジトリを使用している場合、このパスワードは、RDS インスタンスを作成するときに設定したユーザー パス

オプション	必須か?	既定	説明
			ワードです。
--mq-config=external	いいえ	なし このオプションは、メッセージング サービスが RMT サーバーの外部で構成されている場合にのみ指定されます。	このオプションを使って、外部 RabbitMQ メッセージング サービスを使用するように RMT サーバーを構成します。
--mq-server	いいえ	現在の構成値またはインストーラーのデフォルト値	メッセージ キュー サーバーの名前。
--mq-port	いいえ	現在の構成値またはインストーラーのデフォルト値	メッセージ キューのポート。
--mq-vhost	いいえ	現在の構成値またはインストーラーのデフォルト値	メッセージ キューの仮想ホスト。
--mq-username	いいえ	現在の構成値またはインストーラーのデフォルト値	メッセージ キューへの接続に使用するユーザー名。
--mq-password	いいえ	現在の構成値またはインストーラーのデフォルト値	メッセージ キューへの接続に使用するユーザー アカウントのパスワード。
--mq-enable-tls	いいえ	現在の構成値または **false**。	メッセージ キューに接続するには TLS 接続が必要です。
--mq-tls-certificate-host	いいえ	現在の構成値または **null**。	メッセージ キュー サーバーの正規名。証明書の名前と一致している必要があります。

オプション	必須か?	既定	説明
--password-salt	いいえ	ランダム値	パスワードのハッシュ化に使用するグローバル ソルト。 これは Resource Monitoring Tool で作成したローカル ユーザー アカウントに適用されます。
--password-min-length	いいえ	現在の構成値または **10**。	パスワードの最小長。 これは Resource Monitoring Tool で作成したローカル ユーザー アカウントに適用されます。
--password-min-numeric	いいえ	現在の構成値または **1**。	パスワードに必要な数字の最小個数。 これは Resource Monitoring Tool で作成したローカル ユーザー アカウントに適用されます。
--password-min-special	いいえ	現在の構成値または **1**。	パスワードに必要な最小特殊文字数 これは Resource Monitoring Tool で作成したローカル ユーザー アカウントに適用されます。
--password-min-latin	いいえ	現在の構成値または **5**。	パスワードに必要な英字の最小個数。 これは Resource Monitoring Tool で作成したローカル ユーザー アカウントに適用されます。

オプション	必須か?	既定	説明
--password-require-mixed-case	いいえ	現在の構成値または **true**。	パスワードに大文字と小文字を混在させることが必要。 これは Resource Monitoring Tool で作成したローカル ユーザー アカウントに適用されます。
--auth-timeout-minutes	なし	現在の構成値または **240**。	セッションのユーザー認証が期限切れになるまでの時間 (分)。
--auth-sliding-expiration	なし	現在の構成値または **true**。	ユーザー アクティビティで認証タイムアウト期間をリセットする、またはリセットしない。

例:

管理者パスワードを指定するには、次を実行します。`rmtadmin master-setup --admin-password=<password> --skip-admin-creation`

インストール後にポートを更新するには、次を実行します。 `rmtadmin master-setup --http-port=8000 --skip-admin-creation`

rmtadmin passwd

特定の Resource Monitoring Tool ユーザー アカウントのパスワードをリセットします。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin passwd [positional parameter][global option]
```

位置パラメーター

`username`

パスワードを変更するユーザーの名前。

例: `rmtadmin passwd <username>`

rmtadmin query

Resource Monitoring Tool データベースに対して未加工の SQL クエリを実行し、結果を出力ファイルに保存します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin query [positional parameter][options] [global option]
```

位置パラメーター

`sql`

実行する SQL コマンドのテキスト。複数の SQL コマンドを指定できます。SQL コマンドでファイルを使用するには、ファイル名の頭に **@** を付けます。

オプション

`--outfile=VALUE`

クエリ結果を出力する zip ファイルの名前。デフォルトは *queryresults.zip* です。

`--force`

既存のファイルを上書きします。

`--timeout=VALUE`

クエリのタイムアウト値です。秒単位で指定します。

`--commit`

SQL コマンドによって行われた変更をデータベースにコミットします。デフォルトでは、SQL コマンドはトランザクションとして実行されますが、完了時にロールバックされます。

例: `rmtadmin query <SQLCommand> --outfile=<path and the output file name>`

rmtadmin register

注: バージョン 2020.2 で追加

ブートストラップ ファイルを使用してエージェントを登録します。bootstrap ファイルは、rmtadmin bootstrap-file コマンドを使用してダウンロードできます。

このコマンドはエージェントでのみ使用でき、エージェントをインストールするマシンで実行する必要があります。

シノプシス

```
rmtadmin register [options][positional parameter] [global option]
```

位置パラメーター

`bootstrap`

ブートストラップ ファイルの名前を含むファイル パス。

オプション

`--username`

必須。RMT サーバーのインストール中に作成された管理者ユーザーの名前。

`--password`

必須。ユーザー アカウントのパスワード。

`--password-file`

パスワード情報を含むファイル名が含まれたパス。パスワードは、コマンド ラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。

`--server-name`

エージェントがインストールされているノードの名前。名前が指定されていない場合は、デフォルトでマシン名が使用されます。

`--server-description`

サーバーに関する自由な説明。

例： `rmtadmin register <bootstrap file name and path> --server-name=<server name>`

rmtadmin restart

Resource Monitoring Tool アプリケーションを再起動します。RMT サーバーがインストールされているマシンから実行すると、RMT サーバー アプリケーションが再起動します。エージェントがインストールされているノードで実行すると、その特定のノードでエージェント アプリケーションのみが再起動します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin restart [options] [global option]
```

オプション

オプションを指定しない場合、コマンドは `-all` が使用されているかのように動作します。

RMT サーバー:

`--all`

すべてのサービスを再起動します。

`--master`

RMT サーバーを再起動します。

`--mq`

メッセージ キュー サービスを再起動します。

`--db`

データベース サービスを再起動します。

エージェント:

`--agent`

マシンで実行されているエージェント サービスを再起動します。

例： `rmtadmin restart --db`

rmtadmin rotate-mq-certificate

エージェント マシンで、エージェント メッセージ キューのクライアント証明書をローテーションします。Server マシンで `rmtadmin rotate-mq-certificate` を実行した後に、このコマンドを実行します。

このコマンドは、RMT エージェントから実行する必要があります。

シノプシス

```
rmtadmin rotate-mq-certificate [options] [positional parameter]
```

位置パラメーター

`bootstrap`

ブートストラップ ファイルの名前を含むファイル パス。

オプション

`--username`

必須。マスター サーバーのユーザー名

`--password`

必須。マスター サーバーのユーザー アカウントのパスワード。

`--password-file`

マスター サーバーのパスワード情報を含むファイル名が含まれたパス。パスワードは、コマンドラインで指定するか、使用するパスワードをファイルに含めて指定できます。いずれかの方法で指定しなければ、パスワードの入力が求められます。

`--confirm-restart`

サービスの再起動を確認し、確認プロンプトをバイパスします。

rmtadmin rotate-mq-certificates

Server マシンで、メッセージ キューのサーバー証明書とクライアント証明書をローテーションします。

このコマンドは、RMT サーバーから実行する必要があります。

シノプシス

```
rmtadmin rotate-mq-certificates [options]
```

オプション

`--confirm-restart`

サービスの再起動を確認し、確認プロンプトをバイパスします。

rmtadmin servers

注: バージョン 2021.2 で追加

すべての環境または特定の環境にあるすべての Tableau Server ノードを一覧表示します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin servers [positional parameter][global option]
```

位置パラメーター

`env`

任意の項目です。環境の識別子を指定して、その環境に登録されている Tableau Server ノードのリストを取得します。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

rmtadmin set

指定した構成キーの値を設定します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin set [positional parameter] [global option]
```

位置パラメーター

`key`

値を変更する構成キー。

値

使用する新しい値。

例： `rmtadmin set mq:port <port number>` `mq:port` がキーで、`<port number>` が値です。

サポートされる構成キー

キー	データ型	適用先	デフォルト	説明
auth:disableLocalAuthentication	ブール値	RMT サーバー	FALSE	true に設定すると、認証モードをローカルとしてユーザーを作成することは許可されません。バージョン 2024.2 で追加
auth:disableUserManagementUI	ブール値	RMT サーバー	FALSE	true に設定すると、UI からのユーザー管理は許可されません。バージョン 2024.2 で追加
monitoring:antiDomainLockoutEnabled	ブール値	RMT サーバー	TRUE	TRUE (既定) に設定すると、ドメインロックアウト防止機

キー	データ型	適用先	デフォルト	説明
				能が有効になります。バージョン 2025.1.1 で追加。
monitoring:antiDomainLockoutTimeSeconds	整数	RMT サーバー	600	Tableau Server への接続が再試行される前に、すべての試行がスロットルされる時間。バージョン 2025.1.1 で追加。
monitoring:antiDomainMaxFailedLoginAttempts	整数	RMT サーバー	3	ドメインロックアウト防止機能を有効にするための、Tableau Serverへの連続ログイン失敗回数のしきい値。バージョン

キー	データ型	適用先	デフォルト	説明
				2025.1.1 で追加。
mq:tls:certificateHostName	文字列	RMT サーバーとエージェント	N/A	TLS 経由で RabbitMQ に接続するときに使用する証明書のホスト名
mq:tls:enabled	ブール値	RMT サーバーとエージェント	FALSE	RabbitMQ への TLS 接続を有効または無効にします。
mq:port	整数	RMT サーバーとエージェント	5672	RabbitMQ ホストのポート番号。
server.web.run	ブール値	エージェント	TRUE	エージェントの Web インターフェイスを有効または無効にします。

オプション

`--backup`

変更を行う前に、構成ファイルのバックアップを作成します。

get コマンドを使用して、現在の構成値を表示します。

rmtadmin start

Resource Monitoring Tool サービスを開始します。RMT サーバー上で実行すると、RMT サーバーのみが起動します。エージェント上で実行すると、コマンドを実行しているマシン上でエージェントが起動します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin start [options] [global option]
```

オプション

少なくとも 1 つのオプションを指定する必要があります。

RMT サーバー:

`--all`

すべてのサービスを開始します。

`--master`

RMT サーバーを起動します。

`--mq`

メッセージ キュー サービスを開始します。

`--db`

データベース サービスを開始します。

エージェント:

`--agent`

マシン上で実行されているエージェント サービスを開始します。

例： `rmtadmin start --all`

rmtadmin status

このコマンドを実行しているマシン上で、アプリケーションおよび実行中のサービスの状態を確認します。RMT サーバーの場合、ステータス レポートで、RMT サーバーがデータベースとメッセージ キューに接続されていることを確認します。エージェントの場合、ステータス レポートで、エージェントが RMT サーバーに接続されていることを確認します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

このコマンドは次を返します。

- アプリケーションの状態 (実行中かどうか)
- ライセンスのステータス
- MQ 接続
- キューの詳細
- 使用されているメモリ
- 使用されているディスク スペース
- マシン上の Tableau Server のバージョン
- エージェント構成ファイルに格納されている Tableau Server のプロセス トポロジ。

バージョン 2024.2.0 以降では、RMT サーバーでコマンドを実行すると、次が返されます。

- ローカル データベース内のメッセージング テーブルが警告しきい値 (4 GB) を超える領域を使用している場合、ディスク領域使用量の警告が表示されます。詳細については、メッセージング テーブルのディスク使用量の警告のトラブルシューティングを参照してください。

シノプシス

```
rmtadmin status [global option]
```

rmtadmin stop

Resource Monitoring Tool サービスを停止します。このコマンドを RMT サーバー上で実行すると、RMT サーバーのみを停止します。エージェントがインストールされているマシン上で実行すると、そのマシン上のエージェントのみを停止します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin stop [options] [global option]
```

オプション

少なくとも 1 つのオプションを指定する必要があります。

RMT サーバー:

`--all`

すべてのサービスを停止します。

`--master`

RMT サーバーを停止します。

`--mq`

メッセージ キュー サービスを停止します。

`--db`

データベース サービスを停止します。

エージェント:

`--agent`

マシン上で実行されているエージェント サービスを停止します。

例： `rmtadmin stop --db`

rmtadmin test-env

注: バージョン 2021.2 で追加

Resource Monitoring Tool および Tableau Server 間のさまざまな接続ポイントをテストします。Tableau Server リポジトリ接続、Tableau Server API 接続、Advanced Management 機能などのテストが含まれます。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin test-env [positional parameter][global option]
```

位置パラメーター

`--env`

任意の項目です。Resource Monitoring Tool が監視している環境の識別子。これはシステムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

`--force`

任意の項目です。このオプションを使用すると、十分なデータを使用できない場合でも、ベースラインをすぐに更新できます。既定では、このコマンドは 50 回のロードが正常に終了してベースラインが計算されるまで待機します

rmtadmin update-baseline

注 バージョン 2021.4 で追加

すべてのビューの読み込みのベースラインを更新します。ベースラインは、ビューの 50 回の正常な初期ロード時間の 95 パーセンタイルです。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin update-baseline [options][global option]
```

オプション

`env`

必須。これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。

rmtadmin update-env

注: バージョン 2021.2 で追加

環境設定を更新します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin update-env [options][global option]
```

オプション

オプション	必須か?	説明
--non-interactive	いいえ	すべてのインタラクティブなプロンプトを無効にします。
--no-test	いいえ	API とリポジトリの接続テストを無効にします。
--name	はい	環境 の名前。
--id	いいえ	Web インターフェイスの URL で使用される環境の識別子。 これは、システムで生成された識別子です。これを確認するには、**Admin > Environments > Edit Environment** の順にクリックします。このページの **Environment Details** セクションには、環境の識別子が表示されます。
--gateway-url	いいえ	Tableau Server ゲートウェイへのアクセスに使用される URL。

オプション	必須か?	説明
--version	いいえ	この環境が監視するTableau Serverのバージョン。
--api-username	いいえ	Tableau Server API への接続に使用されるアカウントのユーザー名。ユーザー アカウントは、すべてのTableau Server サイトにアクセスできるTableau Server 管理者である必要があります。
--api-password	いいえ	Tableau Server API への接続に使用されるTableau Server API ユーザー アカウントのパスワード。
--api-password-file	いいえ	Tableau Server API ユーザー アカウントのパスワードを含むファイルの名前。
--repository-server	いいえ	Resource Monitoring Tool と共にインストールされているPostgreSQL データベースのサーバー名。
--repository-port	いいえ	Tableau Server リポジトリデータベースのポート番号。
--repository-username	いいえ	Tableau Server リポジトリと共にインストールされているPostgreSQL データベースへの接続に使用するユーザー名。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接

オプション	必須か?	説明
		アクセスします。これが動作するには、**読み取り専用**データベースユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password	いいえ	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザーアカウントのパスワード。 Resource Monitoring Tool は、パフォーマンス上の理由からTableau Server リポジトリデータベースに直接アクセスします。これが動作するには、**読み取り専用**データベースユーザーのパスワードセットを使用してリポジトリへのアクセスを有効にする必要があります。詳細については、Tableau Server リポジトリへのアクセスを有効にするを参照してください。
--repository-password	いいえ	Tableau Server と共にインストールされているPostgreSQL データベースへの接続に使用するユーザーアカウントのパスワードが保存されているファイル名を含むパス。
--repository-ssl-mode	いいえ	Tableau Server リポジトリSSL モード。

オプション	必須か?	説明
		Prefer は SSL を優先して、**Require** は SSL を必須として、Tableau リポジトリへの SSL接続を構成します。 **Disable** は SSL を使用せずに、Tableau Server リポジトリへの接続を行います。
--repository-ssl-thumbprint	いいえ	直接接続を許可するように Tableau Server PostgreSQL を構成する場合、Tableau Server は証明書とキーを作成します。Tableau Server によって生成された証明書の指紋を指定するか、**server.crt** ファイルを Resource Monitoring Tool のサーバー マシンにコピーするかを選択できます。証明書ファイルをコピーする場合は、指紋を指定する必要はありません。詳細については、クライアントからの直接接続を許可するよう Postgres SSL を構成するを参照してください。

例:`rmtadmin update-env --name=<new name>`

rmtadmin users

Resource Monitoring Tool ユーザー アカウントのリストを表示します。

このコマンドは RMT サーバーでのみ機能します。

シノプシス

```
rmtadmin users
```

rmtadmin version

RMT サーバー マシンで実行すると、RMT サーバーの現在のバージョン情報を表示します。エージェントがインストールされているマシン上で実行すると、エージェントの現在のバージョン情報を表示します。

このコマンドは、RMT サーバーとエージェントの両方で使用できます。

シノプシス

```
rmtadmin version
```

rmtadmin ziplogs

Resource Monitoring Tool ログ ファイルを含む ZIP 形式のアーカイブ ファイルを作成します。

このコマンドは、RMT サーバーとエージェントの両方で使用できますが、コマンドが実行されるアプリケーションのログ ファイルのみが含まれます。

シノプシス

```
rmtadmin ziplogs [positional parameter] [option] [global option]
```

位置パラメーター

`filename`

出力 zip ファイルの名前。名前が指定されていない場合は、デフォルトで log.zip が使用されます。

オプション

`--force`

既存のファイルを上書きします。

例： `rmtadmin ziplogs <zip file name> --force`

グローバル オプション

--help

コマンドのヘルプを表示します。

Tableau Resource Monitoring Tool の通信ポート

この記事では、Tableau Resource Monitoring Tool が使用する通信ポート、デフォルトのポート設定、およびネットワーク要件でこれらのポートをデフォルトの設定とは異なるものにする必要がある場合の変更方法について説明します。

RMT サーバー

ポート情報は、次の方法で変更または更新できます。

- rmtadmin master-setup コマンド
- Web インターフェイス: RMT サーバーがインストールされているマシンから `http://localhost/setup/server` に移動します。

RMT サーバーの通信

使用ポート: 9001、443、80

- セットアップでは、最初はポート9001 に既定値を設定しますが、初期構成の後は既定値を80 に変更します。
- 初期インストール時の SSL 自己署名証明書は、ポート443 用です。RMT サーバーは既定でポート80 でリッスンし、トラフィックを SSL 用のポート443 に転送しようと試みます。ユーザー指定の証明書もポート443 用になります。
    - 証明書が使用されるのは、ユーザー トラフィックが RMT サーバーに転送されるときと、エージェントが Web インターフェイスで登録されるときです。

    - 一般的にポート443の変更はお勧めしません。ただし、環境の制限により変更する必要がある場合は、RMT サーバーに接続するときのポート番号を明示的にリストする必要があります。Web インターフェイスを使用してエージェントを登録する場合は、RMT サーバーに接続するためのポート番号を明示的に指定する必要があります。

- ポート80は非 SSL トラフィック用に使用され、初期設定時またはインストール後に変更できます。

- RMT サーバーは、Tableau Server ゲートウェイに直接アクセスして、REST API を呼び出します。

- PostgreSQL 情報が提供されている場合 (オプション)、RMT サーバーは Tableau Server リポジトリに直接アクセスして情報をクエリします。RMT サーバーとTableau Server リポジトリ間の SSL の構成については、「インストール前チェックリスト - Tableau Resource Monitoring Tool」を参照してください。

RabbitMQ

RabbitMQ は、エージェントとRMT サーバー間で情報を仲介するためのコンポーネント (メッセージキュー) です。

使用されるポート: 5671、5672

- RabbitMQ がインストールされ、ポート5671 (TLS)、5672(TLS 以外) をリッスンします。

- RabbitMQ の TLS 通信はデフォルトで有効になっています。

PostgreSQL データベース

使用されるポート: 5555

- PostgreSQL データベースがインストールされ、ポート5555 をリッスンします。これは変更できません。

- PostgreSQL には、Resource Monitoring Tool が使用する `admin` ユーザーがあります。`readonly` ユーザーを使用して、Resource Monitoring Tool の Web インターフェイスからダウンロードした .tds ファイルの Resource Monitoring Tool PostgreSQL データベースに接続します。.tds ファイルのダウンロードの詳細については、「Tableau データ ソース ファイルを使用したモニタリング データの探索」を参照してください。
- PostgreSQL では、SCRAM-SHA-256 認証が必要です。詳細については、Tableau データ ソース ファイルを使用したモニタリング データの探索 を参照してください。

エージェント

使用されるポート: 9002、443、5672

- インストール中に、エージェント インストーラーは、ポート 9002 でリッスンする登録のみに使用される Web ブラウザーを開きます。
- エージェントの登録が Web インターフェイスを使用して行われる場合、登録プロセスではポート 443 が使用されます。コマンド ラインを使用して登録を行った場合、登録プロセスは RabbitMQ を経由してポート 5672 を使用します。

Tableau Resource Monitoring Tool レスポンス ヘッダー

この記事では、Tableau Resource Monitoring Tool でカスタムのレスポンス ヘッダーを設定する方法について説明します。この機能は、2024 年 8 月の RMT メンテナンス リリースで追加されました。

重要: レスポンス ヘッダーを変更すると、RMT が壊れる可能性があります。ヘッダーを更新した場合は、RMT が正常に動作することを後の試験で確認してください。

レスポンス ヘッダーの表示と更新

レスポンス ヘッダーを表示および変更するには、`rmtadmin get` コマンドと `rmtadmin set` コマンドを使用します。

現在設定されているレスポンス ヘッダーの一覧を表示するには、次のコマンドを実行します。

```
rmtadmin get server.web.responseheaders
```

レスポンス ヘッダーを設定または変更するには、次のコマンドを実行します。

```
rmtadmin set server.web.responseheader.<some header>
```

これにより、指定したヘッダーの値が設定されます。

ヘッダーは、完全なヘッダーと任意のキーと値のペアからなる文字列です。たとえば、次の場合、キープ アライブのタイムアウトと最大値を設定します。

```
rmtadmin set server.web.responseheader.keep_alive "Keep-Alive:
timeout=5, max=997"
```

無効なヘッダー

RMT では無効なレスポンス ヘッダーを設定できます。指定したレスポンス ヘッダーが有効でない場合、RMT は警告を表示しますが、その値を設定することはできます。レスポンス ヘッダーを更新した後、RMT が適切に動作していることを確認するのはお客様の責任です。

レスポンス ヘッダーに*何らかの変更*を加えた後は、RMT をテストしてください。

ユーザーの管理

Resource Monitoring Tool をインストールする際は、設定時に管理者ユーザーを作成します。このユーザーは、RMT サーバーですべてのタスクを実行する権限を持っています。後で他のユーザーを追加して、RMT に関連するタスクを実行できるように指定することができます。

ローカル ユーザーの追加

ユーザーを RMT に追加するには、**[管理]** メニューをクリックして **[ユーザー]** を選択します (環境を設定していない場合、このオプションは使用できない場合があります)。

Resource Monitoring Tool に新しいユーザーを追加する方法

1. **[Admin]** メニューから、**[Users]** を選択します。
2. **[New User]** をクリックします。
3. ユーザー名とパスワードを入力し、サーバー ロールを適用します。
4. **[Save]** をクリックします。

委任されたユーザーの追加

バージョン 2023.1 以降、RMT にローカルで保存されている個別のパスワードを使用するのではなく、ドメイン アカウントの認証資格情報を使用してユーザーを認証できるようになりました。代理認証用に設定されたユーザーが RMT にサインインすると、ユーザーが入力したユーザーとパスワードを、RMT は検証のためにコンピューターのオペレーティング システム (OS) に渡します。

Resource Monitoring Tool に委任された新しいユーザーを追加する方法

1. [**Admin**] メニューから、[**Users**] を選択します。
2. [**New User**] をクリックします。デフォルトは ローカル ユーザーです。
3. [**オペレーティング システム (委任)**] を選択します。
4. **ユーザー名** を入力します。

   ユーザー名 *だけ*を入力してください。ドメインは含めないでください。

   RMT にサインインするときに、ユーザーが標準のネットワーク パスワードを入力すると、OS がこれを検証します。
5. ユーザーが持つべきサーバー ロールを選択します。サーバー ロールの詳細については、以下を参照してください。
6. [**Save**] をクリックします。

ユーザー認証の変更

バージョン 2023.1.0 以降では、既存のユーザーをある認証タイプから別の認証タイプに変更できます。既存のユーザーの認証を変更する方法

1. 編集アイコンをクリックします。
2. 必要な**認証モード**を選択します。
   - **オペレーティング システム (委任)** モードに変更する場合は、ドメインなしで、ユーザー名のみを入力する必要があります。ユーザーは、サインイン時に自分のパスワードを提供します。RMT に保存されている既存のパスワードは削除され、復元できません。
   - **RMT (ローカル)** モードに変更する場合は、ユーザー名とパスワードを入力する必要があります。これらは RMT によってローカルに保存されます。

3. ユーザーに対して選択された**サーバー ロール**を確認します。
4. **[保存]** をクリックして変更を保存します。

Tableau Resource Monitoring Tool のサーバー ロール

Tableau Resource Monitoring Tool には、特定のタスクを実行する権限をユーザーに付与するために、ユーザーに割り当てることができるさまざまなサーバー ロールがあります。

次の表に、使用可能なロールと、各ロールを持つユーザーが実行できる操作を示します。

サーバー ロール	ロールの説明
Download Log Bundles (ログ バンドルのダウンロード)	ログ ファイルをダウンロードできます。
Download TDS Files (TDS ファイルのダウンロード)	**.tds** ファイルをダウンロードできます。
Generate Chargeback Reports (チャージバック レポートの生成)	チャージバック レポートを作成できます。
Server/Environment Management (サーバー/環境管理)	すべての構成を更新するパーミッションを持ちます。
User Management (ユーザー管理)	ユーザーを作成して編集できます。
ウェブフック管理	Slack 通知設定を管理します。

認証の問題のトラブルシューティング

ユーザーのサインインと認証の問題をトラブルシューティングする方法の詳細については、「ユーザー認証のトラブルシューティング」を参照してください。

Tableau Resource Monitoring Tool - インシデント

インシデントは、通常とは異なるイベントに対して報告されるもので、人的な注意が必要になる場合があります。インシデントは、グローバル レベルでさまざまなイベントに対して構成したり、環境ごとにカスタマイズしたりすることができます。

環境について報告された現在のインシデントを表示するには、環境を選択した後、**[インシデント]** メニューから **[All Incidents (すべてのインシデント)]** を選択して、ログに記録および報告されたすべてのインシデントのリストを表示します。

通知の設定によっては、メールまたは Slack で通知を受け取る場合もあります。通知の仕組みの詳細については、「通知」を参照してください。

通常、インシデントは次の 2 つのタイプに分類できます。

1. 既定で設定され、変更できないインシデントこれらは **"システム定義のインシデント"**. と呼ばれます。
2. ユーザーが構成できるインシデントこれらは **"構成可能なインシデント"**. と呼ばれます。

すべてのインシデントには、インシデントの重要度レベルがあります。重要度レベルは、具体的に定義されているものです。システム定義のインシデントの場合、Resource Monitoring Tool が重要度レベルを設定します。構成可能なインシデントの場合は、ユーザーが、環境に適したものに応じて重要度レベルを設定します。

重要度レベルのリストは次のとおりです。

重要度レベル	**キー**	**説明**
情報	`information`	サービスの中断を引き起こさないインシデントを通知する場合
警告	`warning`	サービスの中断を引き起こす可能性のあるインシデント
重要	`critical`	サービスの重大な中断を引き起こすインシデント、またはサービスが完全にダウンしている場合

システム定義のインシデント

以下は、既定で設定されていて、変更できないシステム インシデントのリストです。

インシデント	インシデントのレベル	説明
エージェントダウン	warning	1 つ以上のエージェントがダウンし、RMT サーバーがエージェントと通信できない場合、インシデントがログに記録され、報告されます。
エージェントのライセンスなし	critical	これは、Tableau Server が Tableau Resource Monitoring Tool を使用するライセンスが適切でない場合、または接続の問題が原因で発生する可能性があります。考えられる原因の詳細については、エージェント インシデントを参照してください。
環境のダウン	critical	Tableau Server がオフラインのときにインシデントがログに記録され、レポートされます。

構成可能なインシデント

これらのインシデントは、環境の特性と組織の優先順位に基づいて構成できます。以下に示すのは、重要度レベル、またはしきい値を設定できるイベントです。リンクには、各イベントに対して構成できる内容の詳細が記載されています。

- 抽出失敗インシデント
- ハードウェア インシデント: CPU 使用率、メモリ使用量、メモリの可用性、ディスク空き領域、およびディスク キューの長さが含まれています。
- Hyper スプーリング インシデント
- 低速クエリ インシデント
- 低速ビュー インシデント

グローバル レベルでインシデントを構成するには:

1. **[管理]** メニューから、[グローバル構成] を選択します。
2. **[インシデント]** タブで、インシデントのしきい値と重要度レベルを構成します。

グローバル構成は、その環境がカスタム しきい値を使用していない限り、既存の環境または作成された任意の新しい環境に適用されます。

環境のダウンインシデント

*環境のダウンインシデント*は、お使いの Tableau Server がオフラインであることを検出すると、重要としてログに記録されます。以下に、Tableau Server が Tableau Resource Monitoring Tool によってオフラインと見なされるいくつかのユースケースを示します。

- Resource Monitoring Tool が Tableau Server から応答を得ることができません。Resource Monitoring Tool がインシデントがログに記録される前に 3 回試行します。
- Resource Monitoring Tool がすべてのノードでプロセスがダウンしていることを検出し、ログに環境ダウンインシデントが記録されます。

Resource Monitoring Tool は Tableau Server の `http://{TableauServerUrl}/admin/systeminfo.xml` ページを 30 秒 (既定) ごとにポーリングし、ステータスを確認します。30 秒のポーリング間隔が十分でない場合は、60 秒に増やします。ステータスがオフラインであるか、ポーリング間隔において 3 回連続して設定値以内に応答しない場合は、重要インシデントが作成されます。

注: rmtadmin set コマンドを使用してポーリング間隔を設定することはできません。この設定は、config.json ファイルの編集によってのみ変更できます。

`config.json` スニペットの例を次に示します。

```
{
  "background":  {
   "run":true,
   "loglevel":"Information",
    "pollRates":  {
      "processStatus":  30000
      }
    }
}
```

キー	データ型	必須か?	説明
processStatus	数値	オプション	Tableau Server のステータスのポーリング間隔で一時停止する時間 (ミリ秒単位)。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つ Resource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

エージェント インシデント

エージェントのライセンスなし (重要)

互換性のないエージェント バージョン (重要)

エージェント メッセージ キューの認証資格情報ローテーションの失敗 (重大)

エージェントダウン (警告)

当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

エージェントのライセンスなし

エージェントのライセンスなしのインシデントは、Tableau Server が Tableau Resource Monitoring Tool を使用するよう適切にライセンスされていないことを検出する場合に重要としてログに記録されます。

Resource Monitoring Tool を使用するTableau Server を監視するには、Tableau Server で Tableau Advanced Management がライセンスされている必要があります。

エージェントはライセンスなしの状態ではデータを収集してRMT サーバーに送信することはなくなります。ただし、環境が適切にライセンスされている場合には、以前収集したデータを引き続き表示できます。

この問題を解決するには、Advanced Management を使用してTableau Server にライセンスを付与します。詳細については、「Tableau Server 上の Tableau Advanced Management について」を参照してください。

注: インシデントをクリアするには、Tableau Server に有効なライセンスを追加してから最大 1 時間かかる場合があります。

Agents Unlicensed インシデントの原因としては、他にも考えられる原因があります。

- Tableau Server API の認証資格情報が正しくない可能性があります。環境構成ページで、ユーザー名とパスワードが正しいこと、および使用するアカウントに Tableau Server のサーバー管理者ロールがあることを確認します。
- Tableau Server REST API に接続できません。Tableau Server で REST API が有効にされていること、および Tableau Server が実行されていることを確認します。

  REST API 接続の問題を解決するには、**Admin** メニューで **Environments** を選択し、変更する環境で **Edit Environment** をクリックし、**Environment Details** タブで、ページの右側の **Tableau Server REST API** セクションで**Test Connection** ボタンをクリックします。
- RMT サーバーは、ネットワーク接続または類似の問題が原因で、Tableau Server に接続できませんでした。

互換性のないエージェントバージョン

環境内の 1 つ以上のエージェントが RMT サーバーのバージョンと互換性のないバージョンを使用している場合、互換性のないエージェントは「重要」なインシデントとしてログに記録されます。

この問題を解決するには、エージェントが現在の RMT サーバーのバージョンと互換性のあるバージョンであることを確認する必要があります。

この問題を解決するには、次のいずれかの手順を実行します。

RMT サーバーとすべてのエージェントを同じバージョンにすることをお勧めします。

- エージェントが互換性のある最小バージョンよりも前のバージョンを使用している場合は、インシデントレポートに記載されている互換性のある最小バージョンにエージェントをアップグレードする必要があります。
- エージェントが RMT サーバーのバージョンよりも新しいバージョンを使用している場合は、RMT サーバーも同じ新しいバージョンにアップグレードすることをお勧めします。現在使用しているバージョンによっては、他のエージェントをアップグレードする必要がある場合があります。

エージェントのアップグレード:

エージェントをアップグレードするには、以下のステップを行います。

1. エージェントをインストールしたマシンにエージェントパッケージをコピーします。エージェントは、監視している Tableau Server ノードにインストールされます。エージェントは、RMT サーバーと同じバージョンであるか、RMT サーバーのバージョンと互換性のあるバージョンを使用する必要があります。インシデントの詳細には、互換性のある最小バージョンが記載されています。

2. 次のコマンドを使用して、すべての Tableau Server ノードで Resource Monitoring Tool エージェントサービスを停止します。

   ```
   rmtadmin stop --agent
   ```

3. 次のコマンドを実行して、すべてのエージェントをアップグレードします。

   CentOS を含む、RHEL のようなディストリビューションの場合:

   ```
   sudo yum install <pathtoagentinstaller>/<tabrmt-agent-setup-
   <version>-x86_64.rpm>
   ```

```
sudo /opt/tableau/tabrmt/agent/install-scripts/upgrade-rmt-
agent --accepteula
```

Ubuntu ディストリビューションや Debian ディストリビューションの場合:

バージョン 2020.4 から 2020.4.1 以降にアップグレードする場合は、次の手順を実行します。

```
touch /tmp/tabrmt-agent-upgrading.txt && sudo apt install
<tabrmt-agent-setup-<version>-amd_64.deb>
```

```
sudo /opt/tableau/tabrmt/agent/install-scripts/upgrade-rmt-
agent --accepteula
```

注: touch コマンドは、2020.4 からアップグレードする場合にのみ必要です。パッケージをインストールする前に touch コマンドを実行しない場合は、アップグレードの前に既存のバージョンの Resource Monitoring Tool をアンインストールします。

バージョン 2020.4.1 から 2020.4.2 以降にアップグレードする場合は、次の手順を実行します。

```
sudo apt install <pathtoagentinstaller>/<tabrmt-agent-setup-
<version>-amd_64.deb>
```

```
sudo /opt/tableau/tabrmt/agent/install-scripts/upgrade-rmt-
agent --accepteula
```

4. エージェントが実行中であり、アップグレードされていることを確認します。**[管理] > [環境] > [環境の編集] > [サーバー]** タブに移動すると、エージェントの登録ステータスを表示して、Tableau Server ノードの完全なリストとエージェントのステータスを確認できます。

RMT サーバーのアップグレード

1 つ以上のエージェントが新しいバージョンである場合は、RMT サーバーをアップグレードします。

RMT サーバーをアップグレードするには、以下のステップを実行します。

1. RMT サーバーがインストールされているマシンに RMT サーバー パッケージの新しいバージョンをコピーします。

2. 次のコマンドを使用して、RMT サーバーで Resource Monitoring Tool を停止します。

   ```
   rmtadmin stop --master
   ```

3. 次のコマンドを使用して、Tableau Server のすべてのノードで Resource Monitoring Tool を停止します。

   ```
   rmtadmin stop --agent
   ```

4. 注: サービスが停止されたら、ベストプラクティスとして、サービスの停止後に tabrmt-agent か tabrmt-master のいずれかを使用した Resource Monitoring Tool プロセスが実行されていないかどうかを確認してください。これには、PostgreSQL や RabbitMQ は含まれません。次のコマンドを使用するとステータスを確認できます。

   ```
   rmtadmin status
   ```

5. RMT サーバーでコマンドのアップグレードを実行します。これにより、既存のバージョンが新しいバージョンにアップグレードされます。

   CentOS を含む、RHEL のようなディストリビューションの場合:

   ```
   sudo yum install <pathtomasterserverinstaller>/<tabrmt-master-setup-<version>-x86_64.rpm>
   ```

   ```
   sudo /opt/tableau/tabrmt/master/install-scripts/upgrade-rmt-master --accepteula
   ```

   Ubuntu ディストリビューションや Debian ディストリビューションの場合:

   バージョン 2020.4 から 2020.4.1 以降にアップグレードする場合は、次の手順を実行します。

```
touch /tmp/tabrmt-master-upgrading.txt && sudo apt install
<tabrmt-master-setup-<version>-amd_64.deb>

sudo /opt/tableau/tabrmt/master/install-scripts/upgrade-rmt-
master --accepteula
```

注: touch コマンドは、2020.4 からアップグレードする場合にのみ必要です。パッケージをインストールする前に touch コマンドを実行しない場合は、アップグレードの前に既存のバージョンの Resource Monitoring Tool をアンインストールします。

バージョン 2020.4.1 から 2020.4.2 以降にアップグレードする場合は、次の手順を実行します。

```
sudo apt install <pathtomasterserverinstaller>/<tabrmt-master-
setup-<version>-amd_64.deb>

sudo /opt/tableau/tabrmt/master/install-scripts/upgrade-rmt-
master --accepteula
```

6. エージェントとRMT サーバーが実行中であることを確認します。アップグレードの完了後にエージェントとRMT サーバーが自動的に再起動しない場合は、エージェントとRMT サーバーを起動します。

エージェント メッセージ キューの認証資格情報ローテーションの失敗

バージョン 2021.3 以降、セキュリティのベストプラクティスとして、エージェントとRabbit MQ 間の接続の認証資格情報はそのエージェントに固有のものにする必要があります。一意の認証資格情報は、新規インストール時に作成されるか、2021.3 へのアップグレード時に更新されます。

アップグレード中にネットワークに問題が発生すると、認証資格情報の更新プロセスが失敗し、重大なインシデントレポートが作成される可能性があります。ただし、これによってアップグレードプロセスが中断されることはなく、プロセスは続行されます。アップグレードが完了すると、Resource Monitoring Tool は、一意の認証資格情報の作成を毎日再試行します。重大なインシデントレ

ポートは、最初の失敗と後続の再試行の失敗に対して作成されます。その間、エージェントは、新しい認証資格情報が作成されるまで、以前の認証資格情報を使用して作業を続けます。

この問題は自動的に解決する可能性がありますが、問題が解決しない場合は、次の手順を実行してください。

- エージェントがRMT サーバーに接続できることを確認します。
- エージェントとRMT サーバー間のファイアウォールの問題を確認します。

エージェントダウン

エージェントダウン インシデントは、Tableau Resource Monitoring Toolが Tableau Server 上のエージェントと通信できない場合に警告としてログに記録されます。

この問題を解決するには、次のステップを行います。

1. エージェントをホストするハードウェアが実行されており、RMT サーバーと通信できることを確認します。
2. エージェント サービスが正常に登録され、実行されていることを確認します。**[管理] > [環境] > [環境の編集] > [サーバー]** タブに移動すると、エージェントの登録ステータスを表示して、Tableau Server ノードの完全なリストとエージェントのステータスを確認できます。

エージェントのポーリングとインシデントの作成時間

- エージェントは *ハートビート メッセージ*を5 分ごとにResource Monitoring Toolサーバーに送信します。
- Resource Monitoring Tool サーバーは、エージェントから最新の *ハートビート メッセージ*を受信してから15 分以上経過していないかどうかを確認します。
    - エージェントから最新の *ハートビート メッセージ*を受信してから15分以上経過していない場合、そのエージェントはオンラインであると見なされ、インシデントは作成されず、既存の**エージェントのダウン** インシデントはすべてクリアされます。
    - エージェントから最新の *ハートビート メッセージ*を受信してから15 分以上経過している場合、それは最初の試行が失敗したとみなされます。その後、1 分毎にさらに2 回チェックが行われます。3 回の試行 (最初の試行と2 回の再試行) を行った後も最新のハートビート メッセージを受信していない場合は、**エージェントのダウン** インシデントが作成されます。

注: 上記のポーリング間隔に基づき、エージェントがオフラインになった後、**エージェントのダウン** インシデントが作成されるまでに約 17 (15 + 1 + 1) 分かかります。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つResource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

抽出失敗インシデント

*抽出失敗*インシデントは、Tableau で抽出に失敗した場合に警告としてログに記録されます。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つResource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

ハードウェア インシデント

Hardware インシデントでは、サーバー自体を監視します。これらは、Tableau Server のパフォーマンスに影響を与える可能性のあるサーバーの問題を特定するのに役立ちます。

次の項目に関するしきい値を設定できます。

- CPU 使用率
- 使用可能なメモリ
- メモリ使用率
- 空きディスク領域

新しい Tableau Resource Monitoring Tool をインストールすると、既定で次のインシデントが構成されます。

- 空きディスク領域が 10 分以上 10 GB を下回ると警告インシデントがログに記録され、10 分以上 5 GB を下回ると重大インシデントがログに記録されます。
- 使用可能なメモリが 10 分以上 8 GB を下回ると警告インシデントがログに記録されます。

- サーバー全体の CPU 使用率が 5 分間 80% 以上になると警告インシデントがログに記録されます。

> **注:** メモリー関連のインシデントはバイト数で構成されます。

しきい値は、RMT サーバー Web インターフェースを使用して構成するか、構成ファイル `config.json` を更新することによって構成できます。

RMT サーバー Web インターフェイスの使用

ハードウェア インシデントのしきい値を設定するには、**[管理]** メニューで **[構成]** を選択し、**[インシデント]** タブに移動します。

[CPU 使用率] の場合は、次のように設定します。

ハードウェア インシデントのしきい値を設定するには、**[管理]** メニューで **[構成]** を選択し、**[インシデント]** タブに移動します。

キー	必須か?	説明
重大度	必須	「インシデントの重要度レベル」を参照してください。
プロセス	必須	このしきい値は、Tableau Server 全体または指定された単一プロセスに適用されます。
開始しきい値	必須	CPU 使用率が指定された値を超えると、インシデントが作成されて監視が開始します。このしきい値のパーセントと期間を設定してください。
終了しきい値	オプション	CPU 使用率が指定された値を下回ると、インシデントが解決済みと見なされます。

[使用可能なメモリ] の場合は、次のように設定します。

キー	必須か?	説明
重大度	必須	「インシデントの重要度レベル」を参照してください。
開始しきい値	必須	使用可能なメモリが指定された値を下回ると、インシデントが作成されて監視が開始します。このしきい値のパーセントと期間を設定してください。
終了しきい値	オプション	使用可能なメモリが指定された値を超えると、インシデントが解決済みと見なされます。

[メモリ使用率] の場合は、次のように設定します。

キー	必須か?	説明
重大度	必須	「インシデントの重要度レベル」を参照してください。
プロセス	必須	このしきい値は、Tableau Server 全体または指定された単一プロセスに適用されます。
開始しきい値	必須	メモリ使用量が指定された値と等しくなると、インシデントが作成されて監視が開始します。このしきい値のパーセントと期間を設定してください。
終了しきい値	オプション	メモリ使用量が指定された値を下回ると、インシデントが解決済みと見なされます。

[空きディスク領域] の場合は、次のように設定します。

キー	必須か?	説明
重大	必須	「インシデントの重要度レベル」を参照してください。

キー	必須か?	説明
度		
開始しきい値	必須	空きディスク領域が指定された値を下回ると、インシデントが作成されて監視が開始します。このしきい値のパーセントと期間を設定してください。
終了しきい値	オプション	空きディスク領域が指定された値を超えると、インシデントが解決済みと見なされます。

[Disk Queue Length (ディスク キューの長さ)] の場合は、次のように設定します。

キー	必須か?	説明
重大度	必須	「インシデントの重要度レベル」を参照してください。
開始しきい値	必須	ディスクキューの長さが指定された値と等しくなると、インシデントが作成されて監視が開始します。このしきい値のパーセントと期間を設定してください。
終了しきい値	オプション	ディスクキューの長さが指定された値を下回ると、インシデントが解決済みと見なされます。

構成ファイル (config.json) を使用する

次に、2 つのハードウェア インシデントを定義する `config.json` スニペットの例を示します。

```
{
 "monitoring":  {
    "incidents":  {
      "triggers":  [
        {
```

```
            "counter":  "DiskSpaceAvailableKB",
            "severity":  "warning",
            "threshold":  1048576
          },
          {
            "counter":  "ProcessorTimePercent",
            "severity":  "warning",
            "threshold":  0.95,
            "thresholdDuration":  300000,
            "endThreshold":  0.90,
            "endThresholdDuration":  5000
          }
        ]
      }
    }
}
```

- **DiskSpaceAvailableKB** インシデントでは、使用可能なディスク領域が 10 GB を下回ると警告がトリガーされます。
- **ProcessorTimePercent** インシデントでは、CPU の使用率が 5 分以上、95% 以上になると警告がトリガーされます。このインシデントは、CPU の使用率が 5 秒間、90% を下回ると解決されたと見なされます。

既定の設定が要件を満たしている場合と満たしていない場合があり、これはお使いの環境によって変わります。たとえば、使用可能なディスク領域が 2 GB を下回ると識別子が "staging-environment" となり警告がトリガーされる環境では、次のような構成になります。

```
{
  "environments":  {
    "staging-environment":  {
     "monitoring":  {
        "incidents":  {
          "triggers":  [
            {
              "counter":  "DiskSpaceAvailableKB",
              "severity":  "warning",
              "threshold":  2097152
            }
```

```
            ]
          }
        }
      }
    }
  }
```

キー	データ型	必須か?	説明
counter	文字列	必須	監視するハードウェア インシデントの識別子。使用可能なオプションは次のとおりです。 • ProcessorTimePercent • DiskSpaceAvailableKB • DiskQueueTotalLength • MemoryAvailableKB • MemoryCommittedKB
severity	文字列	オプション	「インシデントの重要度レベル」を参照してください。既定値: Warning
threshold	数値	必須	インシデントが監視対象となる前に上回る必要のあるしきい値。
thresholdDuration	数値	オプション	インシデントがトリガーされる前に状況を監視する時間 (ミリ秒単位)。指定しない場合は、インシデントが threshold に到達するとすぐにトリガーされます。
endThreshold	数値	オプション	インシデントが解決と見なされる前に上回る必要のあるしきい値。
endThresholdDuration	数値	オプション	インシデントを完了する前に状況を監視する時間 (ミリ秒単位)。指定しないと、インシデントが endThreshold に到達するとすぐに解決と見なされます。endThreshold が定義されていない場合は、threshold が使用されます。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つResource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

Hyper スプーリング インシデント

Hyper スプーリング インシデントは、Hyper クエリがディスクにスプールされるときに警告としてログに記録されます。これは通常、十分なメモリがないときに、Tableau Server データエンジンが、一時的にディスクに書き込むことによってスプーリングに移行する場合に発生します。詳細は、「Tableau Server データエンジン」トピックの「モリ使用率」のセクションに記載されています。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つResource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

低速クエリインシデント

*低速クエリインシデント*は、データのクエリに時間がかかりすぎて完了できない場合にトリガーされます。既定では、データクエリの実行に 30 秒以上かかる場合に低速クエリインシデントで警告がトリガーされます。

しきい値は、RMT サーバー Web インターフェースを使用して構成するか、構成ファイル `config.json` を更新することによって構成できます。

RMT サーバー Web インターフェイスの使用

低速クエリインシデントのしきい値を設定するには、**[管理]** メニューの **[構成]** を選択し、**[インシデント]** タブに移動します。

キー	必須か?	説明
重大	必須	「インシデントの重要度レベル」を参照してください。

キー	必須か?	説明
度		
期間	必須	クエリが低速であるとみなされるまでの最短時間。値は秒単位で指定します。

構成ファイル (config.json) を使用する

低速クエリインシデントを定義する config.json スニペットの例:

```
{
 "monitoring":  {
    "incidents":  {
      "triggers":  [
        {
          "counter":  "DataQueryDuration",
          "severity":  "critical",
          "threshold":  45000
        }
      ]
    }
  }
}
```

この低速クエリインシデントは、データクエリの実行に 45 秒以上かかる場合に起動します。

インシデントは環境ごとに構成できます。たとえば、データクエリの実行に 30 秒以上かかる場合に識別子が “staging-environment” となり警告をトリガーする環境では、構成が次のようになります。

```
{
  "environments":  {
    "staging-environment":  {
     "monitoring":  {
       "incidents":  {
          "triggers":  [
```

```
            {
              "counter":  "DataQueryDuration",
              "severity":  "warning",
              "threshold":  30000
            }
          ]
        }
      }
    }
  }
}
```

キー	データ型	必須か?	説明
counter	文字列	必須	低速クエリインシデントでは、DataQueryDuration 識別子を使用します。
severity	文字列	オプション	「インシデントの重要度レベル」を参照してください。既定値: Warning
threshold	数値	必須	クエリが低速であるとみなされるまでの最短時間。値はミリ秒単位で指定します。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つ Resource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

低速ビュー インシデント

*低速ビュー*インシデントは、ビューの要求を読み込むまでの時間が長すぎる場合にトリガーされます。

既定では、ビューの要求を読み込むまでに 1 分以上かかると、低速ビュー インシデントによって警告がトリガーされます。

低速ビュー インシデントのしきい値の構成

低速ビュー インシデントのしきい値を設定するには、**[管理]** メニューの **[構成]** を選択し、**[インシデント]** タブに移動します。

キー	必須か?	説明
重大度	必須	「インシデントの重要度レベル」を参照してください。
期間	必須	クエリが低速であるとみなされるまでの最短時間。値は秒単位で指定します。
Tableau ユーザー	必須	このしきい値は、特定のユーザーがビュー要求を行うときに適用することも、すべてのユーザーに適用することもできます。デフォルトは **[すべてのユーザー]** です。
コンテンツ	必須	コンテンツには、ワークブックとビューが含まれます。このしきい値は、単一のビューまたはすべてのビューに適用できます。ワークブックが指定されている場合、しきい値はそのワークブック内のすべてのビューに適用されます。ビューを指定するには、共有 URL を使用します。詳細については、「ビュー URL の構築方法」を参照してください。 例外を指定して、特定のビューまたはワークブックを除外することもできます。 既定は **[すべてのビュー]** です。

ビューの要求ごとに作成されるインシデントは 1 つのみです。Resource Monitoring Tool は、すべてのインシデント トリガーを評価し、複数のトリガーが特定のビューの要求に一致する場合、トリガーは優先度と特異性の順にランク付けされます。最高ランクのトリガーは、インシデントの作成に使用されます。

たとえば、重大度が「重要」のトリガーは、「警告」のトリガーよりも上位にランク付けされます。

暗号化されたデータの収集

Tableau Server から Tableau Resource Monitoring Tool へのデータ収集が確実に暗号化されるようにするには、RMT サーバーとエージェント間の通信、および Tableau Server リポジトリへの接続で、暗号化メッセージングを使用できるようにする必要があります。

バージョン 2022.3 以降

- RMT サーバーと同じマシンに RabbitMQ をインストールした場合 (ローカル構成)、Tableau Resource Monitoring Tool には RMT サーバーとエージェント間の暗号化通信が組み込まれます。RabbitMQ と RMT サーバー間の暗号化には、設定は必要ありません。
- RabbitMQ が RMT サーバの外部でホストされている場合 (外部構成)、このトピックのバージョン 2022.2 以前のセクションで説明されているのと同じステップを実行する必要があります。

上記のいずれの場合も、Tableau Server のリポジトリデータベースとの暗号化通信を有効化し、構成する必要があります。

Tableau リポジトリ SSL の構成

1. Tableau Server が内部 Postgres 接続で SSL 接続を使用するように構成されていることを確認します。詳細については、内部 Postgres 通信用に SSL を設定するを参照してください。Resource Monitoring Tool を使用すると、SSL 接続で証明書ファイルまたは指紋を使用できます。証明書ファイルを使用する場合は、Tableau Server で生成された内部 Postgres SSL 接続用の証明書ファイルを、RMT サーバーをインストールするマシンにコピーします。詳細については、クライアントからの直接接続を許可するようPostgres SSL を構成するを参照してください。

2. RMT サーバー Web インターフェイスで、[管理] メニューから [Environments (環境)] を選択します。環境の編集アイコンをクリックします。

   Tableau Repository Configuration セクション:

   1. **SSL Mode** ドロップダウンボックスで **Prefer SSL** または **Require SSL** を選択して、Tableau リポジトリへの SSL 接続を構成します。**Disable** を選択すると、Tableau Server リポジトリ接続の作成で SSL が使用されることはありません。

Prefer SSL モードでは、Resource Monitoring Tool が最初に SSL の使用を試みて、失敗した場合は暗号化されていない接続を試みます。

Require SSL モードでは、SSL 接続が失敗した場合、Tableau Server リポジトリへの接続は完全に失敗します。この場合、Tableau Server の REST API 接続が Tableau Server との通信に使用されます。

2. Tableau Server によって生成された指紋を指定するか、**server.crt** ファイルを Resource Monitoring Tool マスター サーバー マシンにコピーするかを選択できます。証明書ファイルをコピーする場合は、指紋を指定する必要はありません。詳細については、クライアントからの直接接続を許可するようPostgres SSL を構成するを参照してください。

バージョン 2022.2 以前

次の操作を実行すると、エージェントとRMT サーバー間の通信を暗号化できます。

- SSL/TLS 証明書を使用した RabbitMQ の構成。
- 暗号化されたメッセージングを有効にするための、RMT サーバーとエージェントの構成。
- Tableau リポジトリへの暗号化接続の構成。

RabbitMQ の設定

RabbitMQ サーバーの設定に関する詳細については、RabbitMQ のドキュメント「TLS Setup」を参照してください。

Tableau Resource Monitoring Tool の設定

RabbitMQ を TLS 用に構成したら、すべてのクライアント アプリケーション (Tableau Resource Monitoring ToolRMT サーバーおよびすべてのエージェント) を構成して暗号化されたメッセージングを有効にする必要があります。RMT サーバーの Web インターフェイスで次の操作を行います。

1. RMT サーバーがインストールされているマシンで、*http://<hostname>/setup/server* に移動します。
2. [メッセージ キュー] セクションで、[Enable TLS (TLS の有効化)] チェック ボックスをオンにして、証明書のホスト名を指定します。
3. 必要に応じてポート情報を更新します。

RMT サーバーやエージェントを構成して、暗号化されたメッセージングを有効にする場合は、次のことに注意してください。

- 暗号化を有効にするには、`enabled` フラグと`certificateHostName` の両方が構成されている必要があります。
- `certificateHostName` 変数は、サーバー証明書上の正規名 (CN=) と一致する必要があります。一致していなければ接続に失敗します。
- `port` 番号は、onRabbitMQ を構成した TLS ポートに基づいて変更しなければならない可能性があります。

注: SSL が構成される前にエージェントがすでに登録されている場合は、エージェントを再登録する必要があります。これを行うには、新しいブートストラップ ファイルをダウンロードし、新しいブートストラップ ファイルを使用してエージェントを再登録します。エージェントの再登録の詳細については、エージェントの再登録 を参照してください。

実行可能なユーザー

暗号化されたデータ収集を構成するには、Tableau Server 管理者かつ Resource Monitoring Tool 管理者である必要があります。

RMT サーバーのハードウェアを変更した場合 - PostgreSQL データベースを調整する

RMT サーバーがインストールされているマシンのハードウェアには、アップグレードや変更が必要です。ハードウェア (特にメモリとCPU) を変更した場合は、RMT サーバーと共にインストールされている参照元の PostgreSQL データベースを調整することをお勧めします。

セットアップ プログラムを使用してRMT サーバーをインストールするとき、PostgreSQL データベースはマシン上で利用可能なハードウェアの使用を最適化するように構成されます。この最適化には、バッファーとキャッシュ サイズの管理が含まれます。マシンのハードウェアを変更すると、パフォーマンスが影響を受ける可能性があります。

ハードウェアの変更を反映するように構成を更新するには、次の手順を使用します。

1. sudo へのフル アクセス権を持つユーザーとしてRMT サーバーに接続します。
2. "マスター" ディレクトリに移動し、次のコマンドを実行します。

```
sudo /opt/tableau/tabrmt/master/tabrmt-master optimize
```

構成ファイルが更新されたことを示すメッセージが表示されます。また、変更を有効にするには PostgreSQL サービスの再起動が必要であることが示されます。構成ファイルの更新には、サーバーのダウンタイムは必要ありません。

3. PostgreSQL データベースを再起動します。この操作は、Resource Monitoring Tool が使用されていない勤務時間外に行うことができます。

Tableau Server のトポロジの変更

この記事は、Tableau Server に特定のトポロジ変更を加えたときに必要な操作を理解するのに役立ちます。変更には、ノードの追加または削除、ノードからのプロセスの追加または削除、ノード上のプロセス数の変更などがあります。

ノードの追加

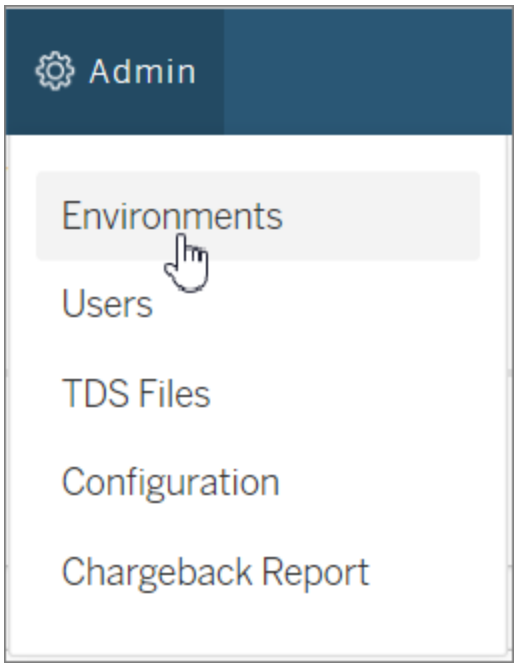
Tableau Server に新しいノードを追加する場合、このノードにエージェントをインストールする必要があります。Tableau Server クラスターにエージェントをインストールする詳細については、Tableau Resource Monitoring Tool のインストール のインストールを参照してください。エージェントをインストールするまで、このノードに関する情報はレポートに含まれません。

ノードの削除

Tableau Server クラスタから既存のノードを削除する場合、RMT サーバー上の環境を更新する必要があります。

環境を更新するには、以下のステップを行います。

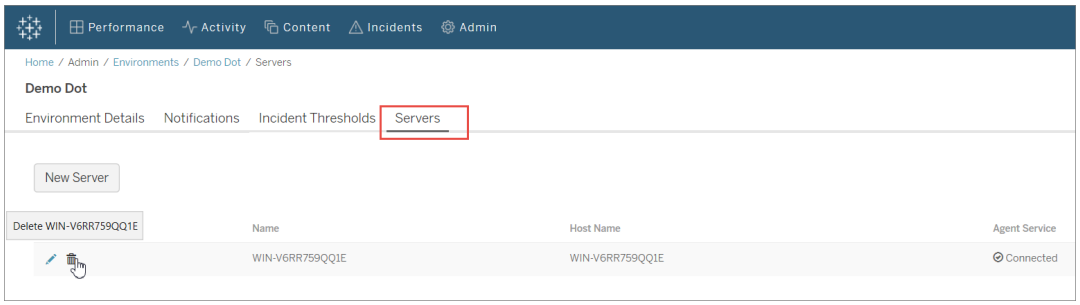
1. RMT サーバーWeb インターフェイスにログインします。
2. **[管理]** メニューから**[環境]** を選択します。この変更を反映する環境を選択します。



3. **[環境の編集]** を選択します。これにより、環境に関する詳細情報が表示されます。

4. **[サーバー]** タブで、Tableau Server から削除されたノードを選択し、**[削除]** を選択します。

> 注: ホスト名はノードの名前または IP アドレスと一致する必要があります。



エージェントの再登録

エージェントを再登録する必要がある場合があります。そのようなユースケースの 1 つは、1 つのノードまたは Tableau クラスタ全体に別の環境を指定する場合です。別の例は、Tableau Server ノードのイメージを再適用する必要があり、エージェントをインストールして再登録する必要がある場合です。エージェントを再登録するには、次の手順を実行します。

1. RMT サーバーWeb インターフェイスで、**[管理] -> [環境]** に移動します。環境を選択します。**[サーバー]** タブで、[削除]をクリックして、この環境のこのノードのエージェント登録を完全に削除します。これにより、このノードで収集された監視データの履歴も削除されます。この方法は、履歴を失っても問題がないシナリオで使用します。たとえば、このノードに新しい環境を指定することを計画している場合、履歴データを削除して、新しい環境から始めても構いません。

> **注:** RMT サーバーWeb インターフェイスでは、Tableau Server の各ノードは**サーバー**と呼ばれます。Tableau サーバー全体が環境を構成します。詳細については、概念を参照してください。

実行可能なユーザー

トポロジの変更を実行するには、Tableau Server 管理者とResource Monitoring Tool 管理者の両方の役割が必要です。

Tableau Resource Monitoring Tool ログ ファイル

ログ ファイル

Tableau Resource Monitoring Tool には多くのコンポーネントが含まれます。各コンポーネントは、独自のログ ファイルのセットを保持します。

既定では、各セットで最大ファイル サイズが 1 GB のログ ファイルを 31 個保持します。これを超過すると、余分なログ ファイルが自動的に削除されます。

注: 当社の企業価値である「平等」に合致させるため、包括的ではない用語を可能な限り変更しました。場面に応じて用語を変更すると重大な影響につながる可能性があるため、既存の用語を維持します。そのため、CLI コマンドとオプション、インストール フォルダー、構成ファイル、その他のインスタンスには、これらの用語が引き続き使用されます。詳細については、「Tableau ヘルプについて」を参照してください。

コンポーネント

コンポーネント **RMT でインストールされるサービスの詳細については、Resource Monitoring Tool (RMT) サービスを参照してください。**	ログ ファイルの場所 `/var/opt/tableau/tabrmt/master/logs`	説明
バックグラウンダー	`background\YYYYMMDD-pts.log`	一般的なバックグラウンドプロセッサーによって生成されたログ。
ディレクター	`director\YYYYMMDD-pts.log`	ディレクターのバックグラウンドプロセッサーによって生

コンポーネント **RMT でインストールされるサービスの詳細については、Resource Monitoring Tool (RMT) サービスを参照してください。**	ログ ファイルの場所 /var/opt/tableau/tabrmt/master/logs	説明
		成されたログ。
ホスト	host\YYYYMMDD.log	ホストによって生成されたログ。
Web サーバー	web\YYYYMMDD-pts.log	Web サーバーによって生成されたログ。
RabbitMQ (メッセージ ブローカー)	rabbitmq/*.log	RabbbitMQ によって生成されたログ
PostgreSQL (ローカル データ リポジトリ)	pgsql/*.log and *.csv	ローカルの PostgreSQL データベースによって生成されたログ。

高度なトラブルシューティングでは、詳細なロギングをより多くの内部アクティビティに対して有効にできます。これらのアクティビティログは、関連付けられたコンポーネントのログ ディレクトリに格納されます。

アクティビティ	ログ ファイルの場所	説明
データベース	logs*\YYYYMMDD-ef.log	内部データベース クエリのログ。既定では、このログにはエラーのみが書き込まれます。

アクティビティ	ログ ファイルの場所	説明
		db.logLevel を Debug 以上に変更して有効にします。
メッセージ キュー	logs*\YYYYMMDD-mq.log	内部メッセージ キュー通信のログ。既定では、このログは生成されません。mq.logLevel を Debug 以上に変更して有効にします。

ログ レベルの構成

Resource Monitoring Tool のログ レベルは、次の方法で調整できます。

- Resource Monitoring Tool Web インターフェイス: [管理] メニューから[構成] に移動し、[Advance (詳細)] タブを選択します。ログ レベルを設定し、データベース クエリとメッセージ キューの通信を含めるかどうかを指定することもできます。

  Web インターフェイスを使用して設定できるログ レベル:

レベル
既定: 情報、警告、エラー、および重要なレベルが含まれます
詳細: 情報、警告、エラー、重要なレベル (既定のすべて)、およびデバッグ レベルが含まれます

- config.json ファイルの設定。ログ レベルは、コンポーネントごとに個別に構成されます。既定の構成ファイル パスは以下のとおりです。これはインストール フォルダーによって変わる場合があります。

インストール	既定の場所
• Tableau Resource Monitoring Tool	/var/opt/tableau/tabrmt/master/config.json
Tableau Resource Monitoring Tool エージェント	/var/opt/tableau/tabrmt/agent/config.json

構成ファイルを使用して構成できるログ レベル:

レベル
追跡
デバッグ
情報
警告
エラー
重要
なし

以下は、一部の既定のログ レベル設定を示すスニペットです。

```
{
  "db":  {
    "logLevel":  "Error"
  },
  "mq":  {
    "logLevel":  "Warning"
  },
  "server":  {
    "background":  {
      "logLevel":  "Information"
```

```
    },
    "director":  {
      "logLevel":  "Information"
    },
    "web":  {
      "logLevel":  "Information"
    }
  }
}
```

Tableau カスタマー サポートへのログ ファイルの送信

Tableau サポートと協力しており、サポートよりログ ファイルの送信を求められる場合は、ファイルの送信前に zip ファイルに圧縮してください。

1. RMT サーバー、および Resource Monitoring Tool エージェントがインストールされている各サーバーに接続します。
2. コマンドプロンプトを開き、`rmtadmin ziplogs <output file path>` を実行してログ ファイルの ZIP 形式のアーカイブを作成します。

   Tableau へのログ ファイルの送信の詳細については、「Tableau ナレッジ ベース」を参照してください。

注: インストール ログ ファイルは、`/var/tmp/` にあります。。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つ Resource Monitoring Tool 管理者や Resource Monitoring Tool ユーザー。

Tableau ログ ファイル

Tableau Resource Monitoring Tool エージェントは、Tableau ログ ファイルをほぼリアルタイムで監視し、ログ メッセージを RMT サーバーに送信して処理およびレポートを行います。

以下は、監視される特定の Tableau ログ ファイルのリストです。

すべてのディレクトリパスが、Tableau Server のデータディレクトリに関連します。既定では、次の場所にあります。/var/opt/tableau/tabrmt/data/tabsvc

ディレクトリ	ファイル名
logs\backgrounder	backgrounder-*.log
logs\httpd	*.log
logs\vizportal	vizportal-*.log
logs\dataserver	dataserver_*.txt
logs\vizqlserver	vizqlserver_*.txt

Tableau のアップグレード

Tableau Resource Monitoring Tool で監視されている Tableau Server のインストールをアップグレードする場合、追加の手順をいくつか実行する必要があります。

Resource Monitoring Tool で Tableau の新しいバージョンがサポートされていることを確認する

Tableau をアップグレードする前に、インストールしているバージョンの Resource Monitoring Tool が、アップグレードする Tableau のバージョンをサポートしていることを確認します。

最も簡単に確認する方法は、Resource Monitoring Tool Web インターフェイスにログインして、**[Admin]** | **[Environments]** 画面に移動することです。アップグレードする環境を編集し、アップグレードするバージョンの **[Tableau Version]** ドロップダウンをチェックします。

エージェントの停止

Tableau Server のアップグレード中は、Resource Monitoring Tool エージェントを停止する必要があります。Tableau Server クラスタの各マシンで、次の手順に従います。

次のコマンドを tabrmt-master ユーザーとして実行します。

```
sudo su --login tabrmt-master

rmtadmin stop --agent
```

Tableau のアップグレード

Tableau Server をアップグレードするには、通常のプロセスに従います。このプロセスは、Tableau Server のドキュメントで概説されています。

Resource Monitoring Tool での Tableau バージョンの更新

1. Resource Monitoring Tool Web インターフェイスにログインします。
2. **Admin** | へ移動します。**Environments**
3. アップグレードした環境を編集します。
4. **Tableau Version** が一致するように修正します。

エージェントの再起動

アップグレードが完了したら、Resource Monitoring Tool エージェントを再起動できます。Tableau Server クラスタの各マシンで、次の手順に従います。

次のコマンドを tabrmt-master ユーザーとして実行します。

```
sudo su --login tabrmt-master
```

```
rmtadmin restart --agent
```

実行可能なユーザー

Resource Monitoring Tool をアップグレードするには、次のものがすべて必要です。

- Resource Monitoring Tool をインストールするマシンのユーザーに対する sudo へのフル アクセス権
- Tableau Server の管理者サイトロール
- Resource Monitoring Tool 管理者アカウント。

Tableau Server のパフォーマンスを監視する

Tableau Resource Monitoring Tool は、Tableau Server の正常性とパフォーマンスを監視したり分析したりするために使用されます。Tableau Server ノードにインストールされたエージェントを通じて、パフォーマンス、使用状況、ハードウェアのメトリクスが収集され、RMT サーバーに送信されます。その後、集計および分析されたデータが、RMT サーバーの Web インターフェイス上のフォームやチャート、およびビューに表示されます。

RMT サーバーの Web インターフェイスには、読み込み時間の遅延、抽出の失敗、およびその他の重大な問題の原因を特定するために使用できるチャートとビューが組み込まれています。詳細については、Tableau Resource Monitoring Tool を使用して Tableau Server のパフォーマンスを監視するを参照してください。

また、事前に設定されたグラフの作成に使用できるデータをダウンロードして、さらに詳しく調べることもできます。詳細については、Tableau データ ソース ファイルを使用したモニタリング データの探索を参照してください。

Tableau Resource Monitoring Tool を使用して Tableau Server のパフォーマンスを監視する

Tableau Resource Monitoring Tool を使用すると、Tableau Server 環境内の正常性とパフォーマンスの問題を簡単に検出して解決できます。Resource Monitoring Tool の 1 つのインスタンスは、管理者が複数の Tableau Server を監視するための単一の統合インターフェイスを提供します。

Resource Monitoring Tool は、次の 2 つのコンポーネントで構成されます。

- Tableau Server での操作のリソース使用量およびパフォーマンス記録を収集するエージェント。
- このパフォーマンス データを集計して、Web インターフェイスにチャート形式で表示する RMT サーバー

Resource Monitoring Tool では、Tableau Server の外部プロセスに関する情報を提供できない場合があります。

- 外部 リポジトリ: プロセス ステータスは監視されず、チャートに含まれていません
- 外部 ファイル ストア: プロセス ステータスは監視されず、抽出 ファイル サイズの情報は追跡されません。抽出の失敗と完了までにかかった時間に関する情報は、引き続き提供されます。
- 外部 ゲートウェイ: プロセス ステータスは監視されず、チャートに含まれません。

事前に設定されたチャート

RMT サーバーの Web インターフェイスには、パフォーマンスのボトルネックと問題を特定するために使用できるダッシュボードとチャートが組み込まれています。これらの組み込みのチャートとメトリクスは、**[Performance (パフォーマンス)]**、**[Activity (アクティビティ)]**、および **[Content (コンテンツ)]**

ページにあります。構成したしきい値に基づいて外れ値または異常な動作をレポートするアラート(**インシデント**と呼ばれる) を設定することもできます。

Resource Monitoring ToolTableau Server の監視に役立つ機能の完全なリストは次のとおりです。

機能	コンポーネント	RMT サーバー内の場所
インシデント/アラートの構成	• Tableau Server ノードおよびプロセス レベルのハードウェア リソース • ビューの読み込み時間 • クエリ時間 • Tableau Server ノードのダウン イベント	**[管理]** -> **[環境]** -> **[環境の編集]** -> **[インシデント]** または **[通知]** タブ • **[インシデント]** タブを使用して、インシデントのしきい値を構成します。 • **[通知]** タブを使用して、通知をいつどのように受信するかを構成します。
ハードウェア リソース	• CPU • メモリ • ディスク • ネットワーク	概要を見るには、**[Performance (パフォーマンス)]** -> **[Select environment (環境の選択)]** -> **[Environment (環境)]** に移動します。 詳細を見るには、**[Performance (パフォーマンス)]** -> **[Select environment (環境の選択)]** -> **[Servers (サーバー)]** に移動します。
Tableau Server の使用状況	• 同時ユーザー • サーバー全体で集計されたビューの読み込み時間 • バックグラウンド タスク • 失敗したタスクの期間、状態、詳細なエラー	これらの情報のほとんどは、パフォーマンス ダッシュボードの **[環境]** タブにあります。 **[Performance (パフォーマンス)]** -> **[Select environment (環境の選択)]** -> **[Environment (環境)]** クエリの詳細を見るには、**[Activity (アクティビティ)]** -> **[Data Queries (データ クエリ)]** に移動し、クエリ

機能	コンポーネント	RMT サーバー内の場所

	• 失敗したタスクのログ スニペット
	• クエリの詳細
	  • 実行時間の長いクエリまたは高価なクエリの期間、状態、クエリテキスト
	• 次のログ情報:
	  • 個々のビューの読み込み時間
	  • VizQL セッション情報
	  • ビュー読み込み時のイベントの期間
	  • クエリテキスト、読み込み時間、返された行数
	  • バックグラウンド タスクと実行時間
		を選択して詳細を確認します。
		
		ログ情報を見るには、**[Activity (アクティビティ)]** に移動し、リストから選択して詳細を確認します。

カスタム チャート

事前に設定されたチャートの作成に使用できるデータをダウンロードして、Tableau でさらに詳しく調べることもできます。詳細については、Tableau データ ソース ファイルを使用したモニタリング データの探索を参照してください。

実行可能なユーザー

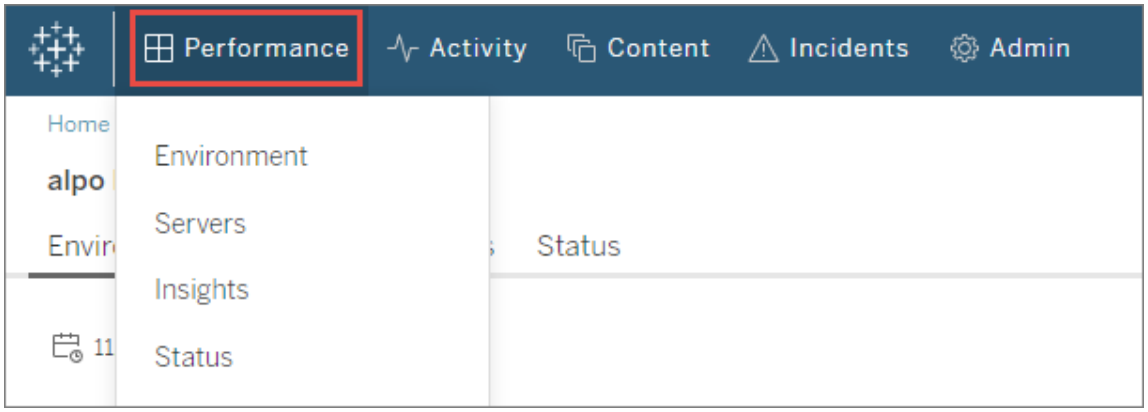
すべての Resource Monitoring Tool ユーザーがチャートを表示できます。

Tableau Resource Monitoring Tool のパフォーマンス チャート

Tableau Resource Monitoring Tool には、さまざまなパフォーマンス メトリクスを監視および分析するのに役立つダッシュボードが含まれています。メトリクスには、ハードウェア リソースの使用状況、ユーザー アクティビティ、Tableau Server プロセスのステータスなどがあります。これらのダッシュボードは、パフォーマンスのボトルネックを特定したり、Tableau Server の全体的な正常性を確認したりする場合に便利です。

このトピックでは、**[パフォーマンス]** ページのチャートについて説明します。パフォーマンス ページには、Tableau Server の全体的な状態のダッシュボード ビューがあり、次のタブが含まれています。

1. **環境:** 選択した環境に固有のパフォーマンス メトリクスのダッシュボードです。
2. **サーバー:** ハードウェア リソースのメトリクスに焦点を当てます。
3. **インサイト:** 低速のビューと時間のかかる抽出更新を強調表示します。
4. **ステータス:** Tableau Server プロセスのステータスです。ノード上の各プロセスが、アクティブ、ビジー、またはダウンしているかに関する情報を表示します。これは、Tableau Server の設定ページに表示されるものと似ています。



環境タブ

環境タブのチャートは、Tableau Server の全体的な状態を表示します。

- ページの左側には、リソース関連のメトリクス (パフォーマンス、Tableau プロセス、バックグラウンド タスク) があります。
- 右側のチャートでは、ユーザー アクティビティとその影響に焦点を当てています (同時ユーザー数、低速のビュー読み込み要求数、ビュー読み込み要求合計数)。

ナビゲーションのヒント:
ページの左上隅で、このページのすべてのチャートに適用されるタイムラインを選択できます。特定のチャートの一部を選択して時間範囲を選択することもできます (範囲選択)。
- ハードウェア パフォーマンス チャートとバックグラウンド タスク チャートで範囲を選択すると、タイムラインはこのタブのすべてのチャートのフィルターとして機能します。
- 低速のビュー読み込みとユーザー アクティビティ チャートで範囲を選択すると、[アクティビティ] ページに自動的に移動し、選択した時間範囲の詳細が表示されます。

パフォーマンス チャート

このチャートは、各Tableau Serverノードのハードウェア リソースの全体的な状態と使用状況を表示します。このチャートに含まれるハードウェア リソースは、CPU、メモリ、ディスク キュー、およびネットワークです。

> **注:** ネットワーク情報は利用できず、現在、Linux で実行されているTableau Server ではサポートされていません。

Tableau プロセス チャート

このチャートは、特定のノードとそのノードで実行されているプロセスに関する詳細を表示します。**2021.4 以降**、ほぼすべてのTableau Server プロセスが追跡されます。ただし、外部リポジトリ、外部ファイル ストア、独立したゲートウェイなど、Tableau Server クラスターの外部で構成されたクラスター コントローラーやプロセスは追跡されません。このチャートを使用すると、特定のノードでCPU やメモリの使用率を高めているプロセスを特定できます。チャートは動的に更新され、選択した時間範囲に最も多くのリソースをノード上で使用している上位 10 のプロセスが強調表示されます。

注: tabprotosrv、postgres、gateway などの一部の Tableau Server サブプロセスは、RMT サーバーによってキャプチャされないため、パフォーマンス チャートとTableau プロセス チャートからのすべての値の合計の間に明らかな不一致が生じる可能性があります。

バックグラウンド タスク チャート

このチャートは、タスクの種類で分類された、バックグラウンド タスク量の概要を表示します。切り替えボタンを使用すると、タスクの総数、タスク時間の中央値、選択した時間範囲の実行時間を示すヒストグラムを表示できます。どのタスクの実行に長い時間がかかっているかをドリルダウンしてすぐに確認できます。

同時ユーザー数チャート

このチャートは、選択した時間範囲にリクエストを送信したユーザーの数を表示します。

注: このチャートの一部を選択すると、関連するアクティビティページに自動的に移動し、その特定の時間範囲の情報が表示されます。

低速のビュー読み込み要求数チャート

このチャートでは、ビューごとに設定された**ベースライン**を使用して、そのビューのレンダリングにかかった時間を比較し、想定よりも長く表示に時間がかかっているかどうかを判断できます。

バージョン 2021.4 では、特定のワークブックが正常にレンダリングされた最初の 10 回の中央値を計算して、ベースラインを定めます。

バージョン 2021.4.1 以降では、特定のワークブックが正常にレンダリングされた 50 回の 95 パーセンタイルを計算して、ベースラインを定めます。

ベースラインが確立されると、その同じワークブックが将来レンダリングされるたびに、ワークブックの読み込みにかかる時間がそのベースラインと比較されます。ビューのレンダリングにかかった時間が予想範囲内にあるか、どの程度予想範囲外にあるかに応じて、次のように分類されます。

- **通常:** ベースラインの 2 倍以下
- **長い:** ベースラインの 2 倍以上
- **非常に長い:** ベースラインの 4 倍以上
- **失敗:** 読み込みに失敗したか、エラーが発生した

このベースラインと比較は、ビューの初期レンダリングにのみ適用されます。フィルターの選択などの後続のアクションにも適用されます。また、新しいバージョンのワークブックをパブリッシュすると、ベースラインの再計算がトリガーされます。

> **注:** このチャートの一部を選択すると、関連するアクティビティページに自動的に移動し、その特定の時間範囲の情報が表示されます。

このチャートの使用方法とビューレンダリングのパフォーマンスの調査方法の詳細については、低速ビュー読み込み要求の調査を参照してください。

ビュー読み込み要求合計数チャート

このチャートは、一定の時間範囲にレンダリングされたビューの数の概観を表示します。このチャートは、特定のインシデントの影響の範囲を評価するのに役立ちます。

> **注:** このチャートの一部を選択すると、関連するアクティビティページに自動的に移動し、その特定の時間範囲の情報が表示されます。

サーバー タブ

このタブのチャートは、**環境**タブの**パフォーマンス**チャートのより詳細なビューを表示します。

インサイト タブ

最も低速のビュー

このチャートは、最も低速の 5 つのビューを平均のビュー読み込み時間順に表示します。オレンジ色のドット インジケーターは、そのビューの最長の読み込み時間を表します。ここに示されている時間範囲は、利用可能なデータの全範囲であり、デフォルトは 2 週間です。

最長の抽出更新

このチャートは、最も時間のかかった 5 つの抽出更新タスクを時間順に表示します。データ ソース抽出とワークブック抽出の両方が考慮されます。これらは、名前の横にあるアイコンによって区別できます。ここに示されている時間範囲は、利用可能なデータの全範囲であり、デフォルトは 2 週間です。

ステータス タブ

このタブには、Tableau Server クラスターの各ノードにおける Tableau Server プロセスとそのステータスが一覧表示されます。これには、外部リポジトリ、外部ファイル ストア、独立したゲートウェイなど、Tableau Server クラスターの外部で構成された Tableau Server プロセスは含まれません。

実行可能なユーザー

すべての Resource Monitoring Tool ユーザーがチャートを表示できます。

関連トピック

- Tableau Resource Monitoring Tool を使用して Tableau Server のパフォーマンスを監視する
- Tableau Resource Monitoring Tool のアクティビティページ
- Tableau Resource Monitoring Tool のコンテンツ ページ

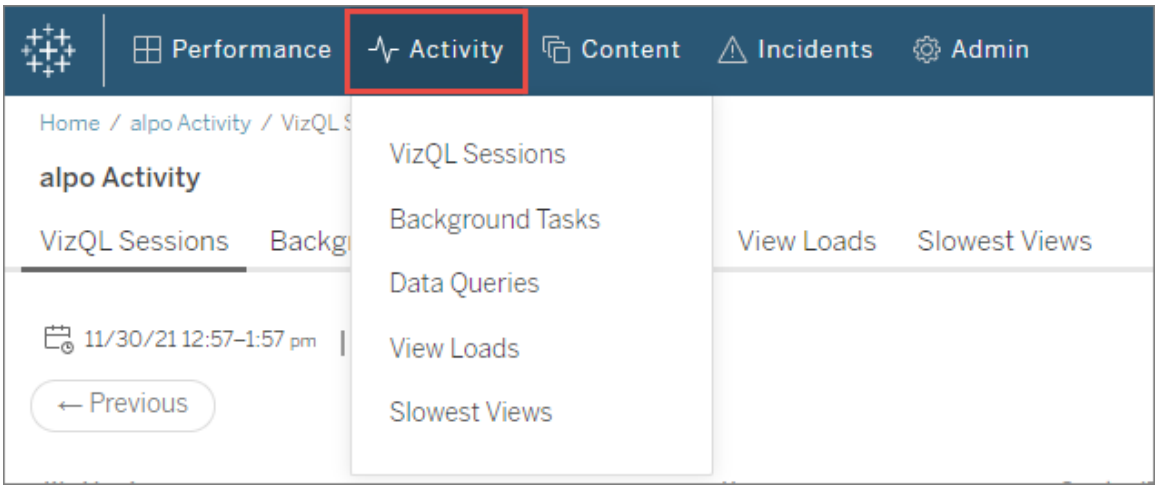
Tableau Resource Monitoring Tool のアクティビティページ

Tableau Resource Monitoring Tool には、さまざまなパフォーマンス メトリクスを監視および分析するのに役立つダッシュボードが含まれています。メトリクスには、ハードウェア リソースの使用状況、ユーザー アクティビティ、Tableau Server プロセスのステータスなどがあります。これらのダッシュボードは、パフォーマンスのボトルネックを特定したり、Tableau Server の全体的な正常性を確認したりする場合に便利です。

このトピックでは、**[Activity (アクティビティ)]** ページの情報について説明します。[Activity (アクティビティ)] ページは、**[Performance (パフォーマンス)]** ページのチャートに表示される内容に次のレベルの詳細を提供します。これらのダッシュボードのいずれかで選択したフィルターは他のダッシュボードに引き継がれるため、パフォーマンスの問題を特定しようとしているときに、対応する情報を確認できます。

[Activity (アクティビティ)] ページには、以下に関する詳細情報が含まれています。

- VizQL セッション
- バックグラウンド タスク
- データ クエリ
- ビューの読み込み
- 最も低速のビュー



VizQL セッション

選択した時間範囲内のすべてのVizQL セッションのリストを表示します。VizQL セッションは、ユーザーがTableau Server 上のワークブックとやり取りする一連の操作です。

セッションID をクリックすると、サマリーチャートに加えて、セッション中に行われた要求、要求時間、関連するワークブックセッションなどの詳細が表示されます。また、関連するデータクエリ、レポートされたインシデント、およびVizQL セッションの時間枠と同じ期間にフィルターリングされた環境アクティビティを確認できます。

バックグラウンドタスク

選択した時間範囲内のすべてのバックグラウンドタスクのリストを表示します。

タスクの開始時間をクリックすると、タスクの完了にかかった時間と同様のタスクの完了にかかった平均時間を比較した結果のサマリーなど、バックグラウンドタスクの詳細が表示されます。また、このタスクの実行中にレポートされた関連インシデントと全体的な環境アクティビティを確認できます。

サイト名をクリックすると、そのサイトの各プロジェクトのワークブック、ビュー、VizQL セッションの数など、サイトに関する詳細情報が表示されます。

データクエリ

選択した時間範囲内のすべてのデータクエリのリストを表示します。

クエリをクリックすると、クエリのパフォーマンスの詳細、完全なクエリテキスト、接続の詳細が表示されます。

ビューの読み込み

選択した時間範囲内にレンダリングされたビューのリストを表示します。

[Load Time Severity Category (読み込み時間の重大度カテゴリー)] フィルターを使用すると、読み込みに通常予想されるよりもはるかに長い時間がかかるビューをフィルターリングすることができます。これは、**[Slow View Load Request (低速ビュー読み込み要求)]** チャートで使用されるものと同じベースライン概念と比較を使用します。ベースラインは、特定のワークブックがレンダリングされた最初の 10 回の中央値を計算することによって確立されます。ベースラインが確立されると、その同じワークブックが将来レンダリングされるたびに、ワークブックの読み込みにかかる時間がそれ自体のベースラインと比較されます。

読み込み時間の重大度のカテゴリは次のとおりです。

- **通常:** ベースラインの 2 倍以下
- **長い:** ベースラインの 2 倍以上
- **非常に長い:** ベースラインの 4 倍以上
- **失敗:** 読み込みに失敗したか、エラーが発生した
- **処理中:** ベースラインは計算中であり、まだ確立されていない

低速ビュー

最も時間がかかったすべてのビューのリストを降順で表示します。このリストは、ベースラインとの比較ではなく、平均読み込み時間に基づいています。時間、パブリッシャー、サイト、または特定のビューでフィルターリングすることにより、このリストを絞り込むことができます。

- ビューをクリックすると、読み込み時間に関する詳細が表示されます。また、このタスクの実行中にレポートされた関連インシデントと全体的な環境アクティビティを確認できます。
- ワークブックをクリックすると、特定のワークブックの詳細が表示されます。

実行可能なユーザー

すべての Resource Monitoring Tool ユーザーがチャートを表示できます。

関連トピック

- Tableau Resource Monitoring Tool を使用して Tableau Server のパフォーマンスを監視する

- Tableau Resource Monitoring Tool のパフォーマンス チャート
- Tableau Resource Monitoring Tool のコンテンツ ページ

Tableau Resource Monitoring Tool のコンテンツ ページ

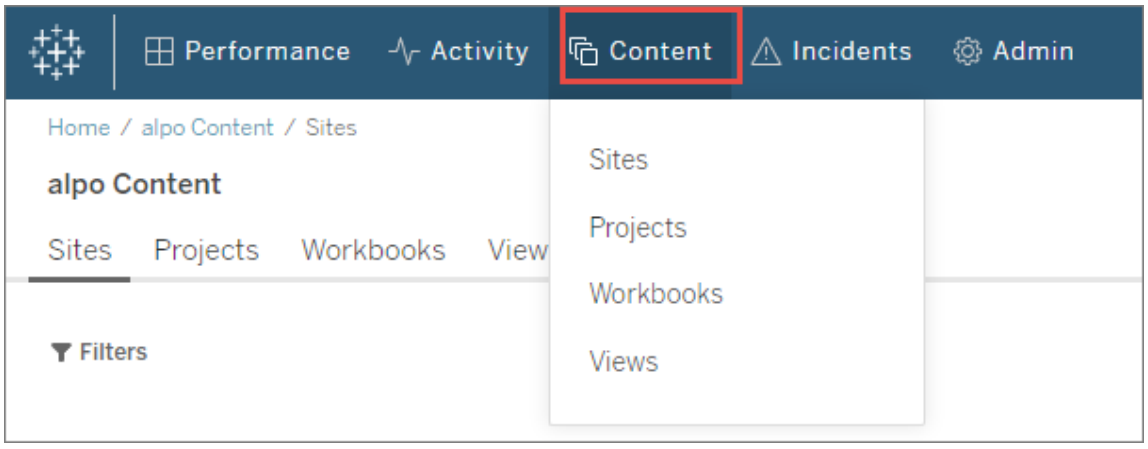
Tableau Resource Monitoring Tool には、さまざまなパフォーマンス メトリクスを監視および分析するのに役立つダッシュボードが含まれています。メトリクスには、ハードウェア リソースの使用状況、ユーザー アクティビティ、Tableau Server プロセスのステータスなどがあります。これらのダッシュボードは、パフォーマンスのボトルネックを特定したり、Tableau Server の全体的な正常性を確認したりする場合に便利です。

このトピックでは、**コンテンツ** ページのチャートに関する情報について説明します。このページでは、特定のコンテンツ アイテムの詳細を確認できます。特定のワークブックまたはビューのパフォーマンスを調査する必要がある場合は、このページから開始します。

コンテンツ ページには、以下に関する詳細情報が含まれています。

- Tableau サイト
- 各 Tableau サイトのプロジェクト
- ワークブック
- ビュー

プロジェクト、ワークブック、またはビューに関連する VizQL セッションやデータ クエリのパフォーマンス メトリクスを表示できます。また、ワークブックまたはビューに固有の関連するインシデントを表示することもできます。



サイト

プロジェクト、ワークブック、および VizQL セッションの合計数など、Tableau Server 環境上のサイトのリストを表示します。

プロジェクト

サイト上のすべてのプロジェクトのリストを表示します。プロジェクトをクリックすると、すべてのワークブック、ビュー、関連する VizQL セッション、およびデータクエリのリストを表示します。

ワークブック

サイト上のすべてのワークブックのリストを表示します。ワークブックをクリックすると、読み込み時間、VizQL セッションの情報、関連するデータクエリ、およびこのワークブックに固有の報告済みインシデントを表示します。

ビュー

サイト上のすべてのビューのリストを表示します。ビューをクリックすると、ビューの読み込み時間、VizQL セッションの情報、関連するデータクエリ、およびビューに固有の報告済みインシデントを表示します。

実行可能なユーザー

すべての Resource Monitoring Tool ユーザーがチャートを表示できます。

関連トピック

- Tableau Resource Monitoring Tool を使用して Tableau Server のパフォーマンスを監視する
- Tableau Resource Monitoring Tool のパフォーマンス チャート
- Tableau Resource Monitoring Tool のアクティビティページ

低速ビュー読み込み要求の調査

[パフォーマンス] ページの **[低速ビュー読み込み要求]** グラフは、Tableau Server でのビューのパフォーマンスと、ユーザー インタラクションに及ぶ影響を理解するための便利な指標です。

[Slow View Load Request (低速ビュー読み込み要求)] チャートは、Tableau Server でいつビューのレンダリングが通常よりも遅くなっているかを示します。これを行うために、チャートは各ビューに対して確立された**ベースライン**を使用して、そのビューのレンダリングにかかる時間を比較し、ビューのレンダリングが予想よりも長くかかっているかどうかを判定します。

バージョン 2021.4 では、特定のワークブックが正常にレンダリングされた最初の 10 回の中央値を計算して、ベースラインを定めます。

バージョン 2021.4.1 以降では、特定のワークブックが正常にレンダリングされた 50 回の 95 パーセンタイルを計算して、ベースラインを定めます。

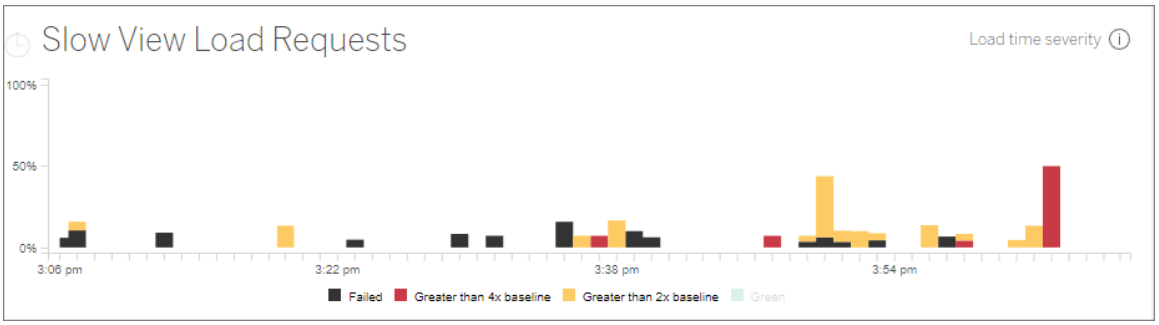
ベースラインが確立されると、その同じワークブックが将来レンダリングされるたびに、ワークブックの読み込みにかかる時間がそのベースラインと比較されます。ビューのレンダリングにかかった時間が予想範囲内にあるか、どの程度予想範囲外にあるかに応じて、次のように分類されます。

- **通常:** ベースラインの 2 倍以下
- **長い:** ベースラインの 2 倍以上
- **非常に長い:** ベースラインの 4 倍以上
- **失敗:** 読み込みに失敗したか、エラーが発生した

> **注**: このベースラインと比較は、ビューの初期レンダリングのみに適用されます。フィルターの選択などの後続のアクションにも適用されます。また、新しいバージョンのワークブックをパブリッシュすると、ベースラインの再計算がトリガーされます。

このチャートには、選択した時間範囲で通常の範囲外にあるビュー読み込みの割合が示されます。したがって、このチャートに黄色 (長い) または赤色 (非常に長い) のスパイクが表示されている場合、問題が発生している可能性があることを示す最初の兆候になります。

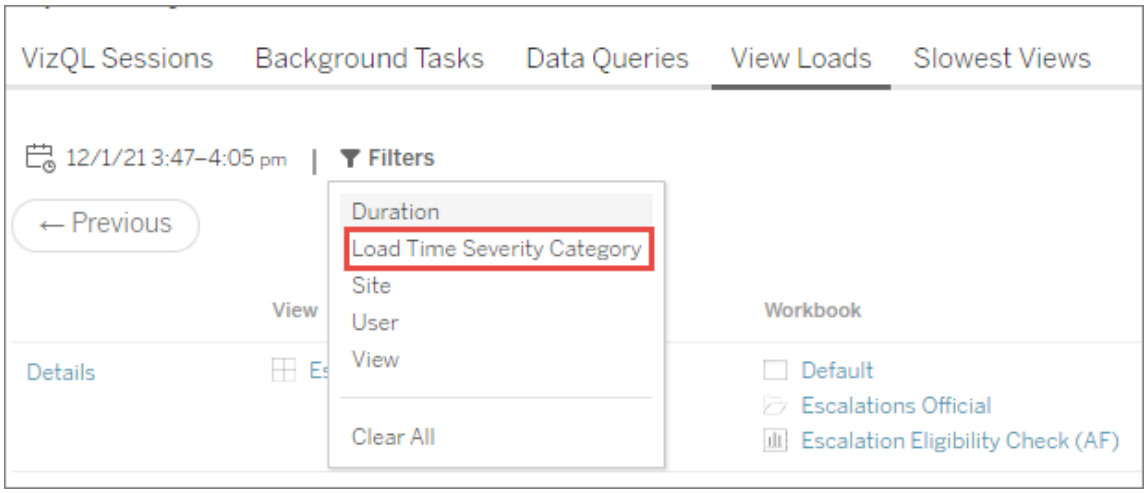
ヒント: 「過去 48 時間」の時間範囲を選択すると、過去と現在のアクティビティに関するコンテキストが得られるため、適切な開始点になる場合があります。



同じページにある **[同時ユーザー]** チャートと **[Total View Load Requests (合計ビュー読み込み要求)]** チャートを使用して、同じ時間範囲でのビュー読み込みが遅いことによる影響の範囲を確認できます。また、**[Tableau プロセス]** チャートを使用して、低速ビュー読み込みパフォーマンスとリソース使用率の相関関係を特定することもできます。たとえば、**[Slow View Load Request (低速ビュー読み込み要求)]** チャートにスパイクが示されているときと同じ時間範囲において、特定のノードで VizQL Server のリソース使用率が高くなっている場合があります。

[Slow View Load Request (低速ビュー読み込み要求)] チャートにスパイクが見られる場合は、さらにドリルダウンして、問題の原因が単一のビューであっても、はるかに広範な問題であっても、その原因を特定できる可能性があります。これを行うには、**[Slow View Load Request (低速ビュー読み込み要求)]** チャート内で、低速ビューの大部分を含める範囲を選択します。そうすると、**[View Loads (ビュー読み込み)]** アクティビティページが開き、同じ時間枠のビュー読み込み要求が表示されます。

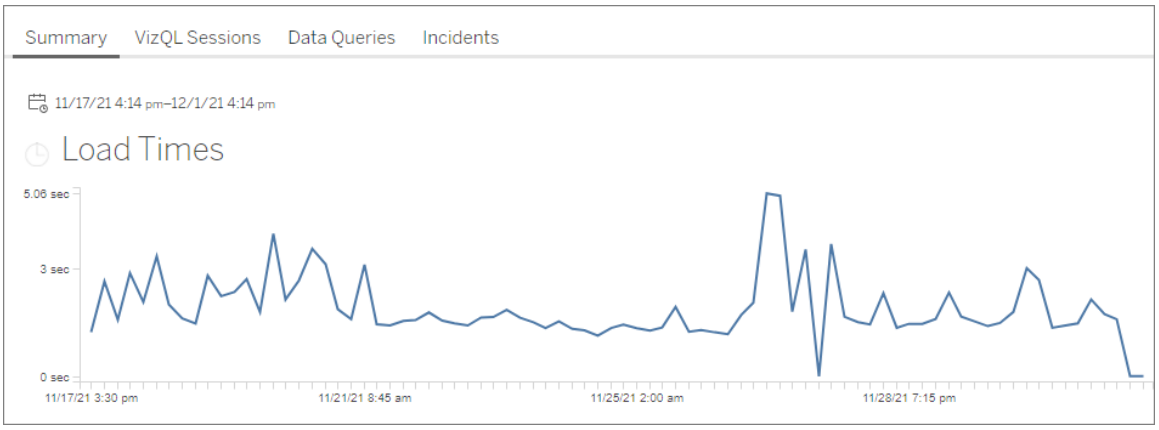
[Load Time Severity Category (読み込み時間の重大度カテゴリー)] でフィルターリングし、調査するビュー読み込みに適切なカテゴリーを選択します。



リストには、チャートの結果を引き起こした可能性のあるビューが示されているはずであり、次のように理解できます。

複数のビュー: リストに複数のビューが表示されている場合、Tableau Server の広範な問題である可能性があります。[Performance (パフォーマンス)] ページに戻り、**[Performance (パフォーマンス)]** と**[Tableau Server Processes (Tableau Server プロセス)]** チャートを見て、リソースの使用状況を調査します。[Performance (パフォーマンス)] ページの **[ステータス]** タブには、プロセスがアクティブ、ビジー、またはダウンしているかどうかのステータスが示されます。このリストで VizQL Server、Data Server、データエンジンプロセスを探します。

Same view (同じビュー): リストにあるビューのほとんが 1 つの単一ビューである場合、そのビューまたはワークブックに問題があることを意味している可能性があります。問題の原因を特定するには、さらに調査が必要になる場合があります。リスト内の **[view name (ビュー名)]** をクリックすると、読み込み時間、関連するデータクエリ、VizQL セッションに関する詳細が表示されます。



重要! [Load Times (読み込み時間)] チャートで、スパイクが始まる直前と直後の日付範囲を調整しても平均読み込み時間が一貫している場合は、ビュー読み込みが大量にキャッシュされたときにベースライン計算が作成された可能性があり、その後のビューの読み込みが「遅い」と見なされます。このシナリオは、Tableau Server またはビューの問題を反映していません。この場合、新しいバージョンのワークブックをパブリッシュすることで、ベースラインの再計算をトリガーできます。

特定のビューのパフォーマンスのトラブルシューティングに役立つリソースを次に示します。

- ワークブックのパフォーマンスの最適化
- ワークブック パフォーマンスの記録と分析

実行可能なユーザー

すべての Resource Monitoring Tool ユーザーがチャートを表示できます。

データの収集で使用されるツール

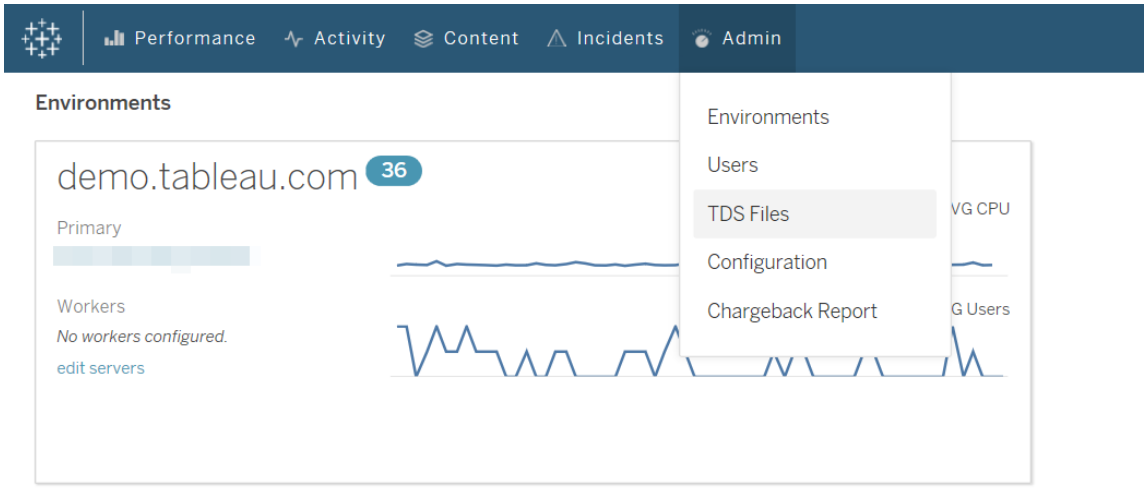
以下は、監視データの収集に使用される特定のクラスのリストです。

クラス	コマンド	監視カテゴリ
`LinuxSystemCpuCollector`	top (上限値)	CPU
`LinuxProcessPerformanceCollector`	top (上	プロセ

クラス	コマンド	監視カテゴリ
	限値)	ス
LinuxSystemMemoryCollector	無料	メモリ
LinuxDiskQueueLengthCollector	iostat	ディスク
LinuxDiskUsageCollector	df	ディスク

Tableau データ ソース ファイルを使用したモニタリング データの探索

Tableau Resource Monitoring Tool には、Tableau Server の正常性とパフォーマンスの監視と分析に使用できる組み込みのグラフが含まれています。Tableau Resource Monitoring Tool が利用するデータは、Tableau データ ソース (.tds) ファイルとしてダウンロードし、Tableau Desktop で探索することができます。.tds ファイルは、Resource Monitoring Tool Web インターフェースを使用して、**Admin** メニューからダウンロードできます。この方法は、ローカルと外部の両方のリポジトリ構成で機能します。



以下は、ダウンロードできる Tableau データ ソース (.tds) ファイルのリストです。

- **バックグラウンド タスク**: 抽出の更新、サブスクリプション、フローなどの、スケジュール済みのバックグラウンダー タスクに関する情報。
- **データ クエリ**: Tableau Server が実行するすべてのクエリに関する情報。

- **ゲートウェイ要求**: VizQL Server セッションの詳細を含む、Tableau Server が処理する HTTP 要求。
- **インシデント**: Resource Monitoring Tool が記録したインシデント。
- **サーバーのパフォーマンス**: Resource Monitoring Tool が収集した、Tableau Server のハードウェアとプロセスの情報。
- **Tableau エンティティ**: Resource Monitoring Tool が収集した、Tableau Server のサイト、プロジェクト、ワークブック、ビューの情報。

要件

- 暗号化には、Tableau Desktop 2020.4 以降でサポートされている SCRAM-SHA-256 を使用しています。

Resource Monitoring Tool PostgreSQL データベースへのアクセスの有効化

Tableau データソース (.tds) ファイルには、Resource Monitoring Tool PostgreSQL データベースへの接続が含まれています。ダウンロードした .tds ファイルに接続する前に、`readonly` ユーザーによる Resource Monitoring Tool PostgreSQL データベースへのアクセス権を有効化する必要があります。`readonly` ユーザーにアクセス権が与えられると、`readonly` のユーザー名とパスワードを使用して、Tableau Desktop の .tds ファイルから Resource Monitoring Tool PostgresSQL データベースに接続できるようになります。

Resource Monitoring Tool バージョン 2022.3 以降:

ローカル リポジトリを持つ Resource Monitoring Tool

1. RMT サーバー マシンで、`readonly`ユーザーによる Resource Monitoring Tool PosgreSQL データベースへのアクセス権を有効化します。

   ```
   rmtadmin data-access ReadOnly
   ```

2. 設定変更を有効にするため、Resource Monitoring Tool PostgreSQL データベースを再起動します。

   ```
   rmtadmin restart --db
   ```

3. `readonly` ユーザーのパスワードを取得します。

   ```
   rmtadmin get db.readOnlyPassword
   ```

外部リポジトリを持つ Resource Monitoring Tool

RDS インスタンスを設定して、Tableau Desktop からのアクセスを許可する必要があります。Resource Monitoring Tool PostgreSQL データベースのユーザー名とパスワードを取得し、それらを使用して .tds ファイルをダウンロードします。詳細については、AWS サイトのドキュメントを参照してください。

Resource Monitoring Tool バージョン 2022.2 以前:

1. `postgresql.conf` ファイルを開きます。デフォルトでは、ファイルは `/var/opt/tableau/tabrmt/data/postgresql<version>` にあります。

2. `Listen_addresses = 'localhost'` を `Listen_addresses = '*'` に更新します。

   > **注:**「`#`」をこの行から削除する必要があります。

3. `pg_hba.conf` ファイルを開きます。このファイルも `postresql.conf` ファイルと同じディレクトリにあります。デフォルトでは、ファイルは `/var/opt/tableau/tabrmt/data/postgresql<version>` にあります。

4. `pg_hba.conf` ファイルの最後に以下を追加し、ファイルを保存します。

   ```
   host all all 0.0.0.0/0 scram-sha-256

   host all all ::/0 scram-sha-256
   ```

5. 設定変更を有効にするため、Resource Monitoring Tool PostgreSQL データベースを再起動します。

   ```
   rmtadmin restart --db
   ```

6. `readonly` ユーザーのパスワードを取得します。

   ```
   rmtadmin get db.readOnlyPassword
   ```

Tableau Desktop から RMT .tds ファイルへの接続

`readonly` ユーザーに対して Resource Monitoring Tool PostgreSQL データベースへのアクセスが有効になったら、Tableau Desktop からダウンロードした .tds ファイルに接続できます。

1. Tableau Desktop で、**[ファイル]**、**[開く]** の順に移動し、Resource Monitoring Tool Web インターフェースからダウンロードした .tds ファイルを選択します。

   > **注**: PostgreSQL データベース ドライバーのインストールが必要になる場合があります。www.tableau.com/ja-jp/support/drivers からドライバーをダウンロードできます。

   .tds ファイルを開いた後、Tableau Desktop が Resource Monitoring Tool Postgres データベースに自動的に接続されない場合、`readonly` のユーザー名とパスワードを Tableau Desktop の接続編集ウィンドウに手動で入力しなければならない場合があります。

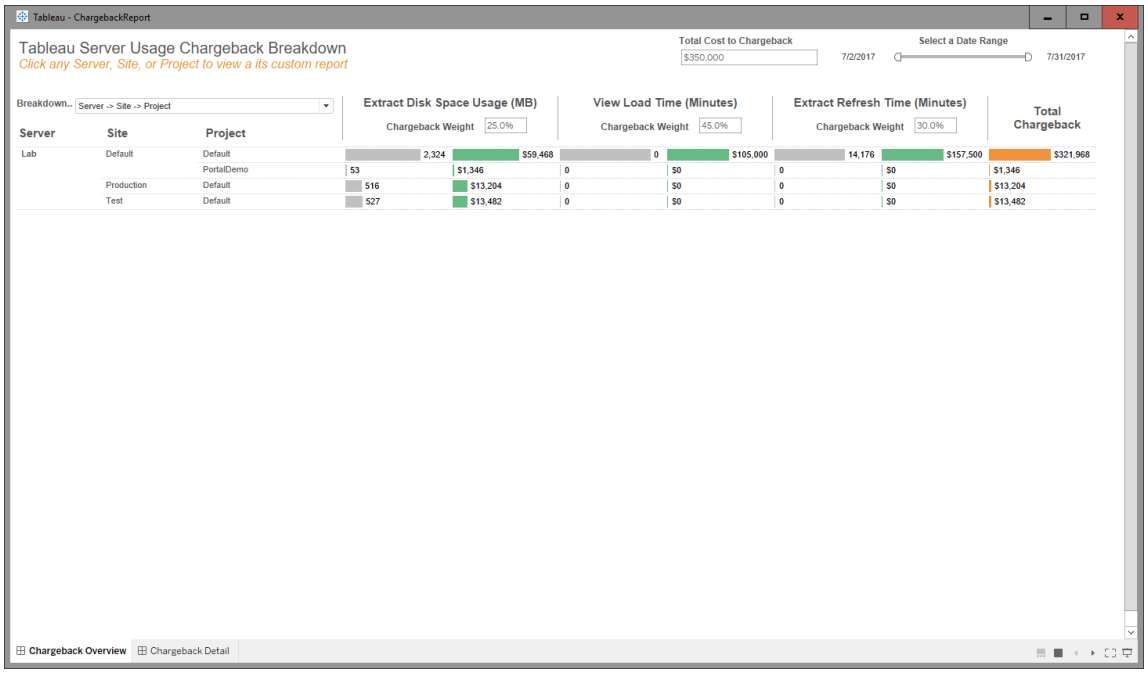
実行可能なユーザー

Resource Monitoring Tool 管理者または「**TDS ファイルのダウンロード**」のサーバー ロールを持つ Resource Monitoring Tool ユーザー。

チャージバック レポート

チャージバック レポートは、プロジェクトごとまたはサイトごとの利用状況を示し、**[管理]** メニューから許可されたユーザーが使用できます。このレポートは生成された抽出を使用して Tableau ワークブックとして生成され、ユーザーは必要に応じ、レポートの変更や抽出の再利用を行うことができます。

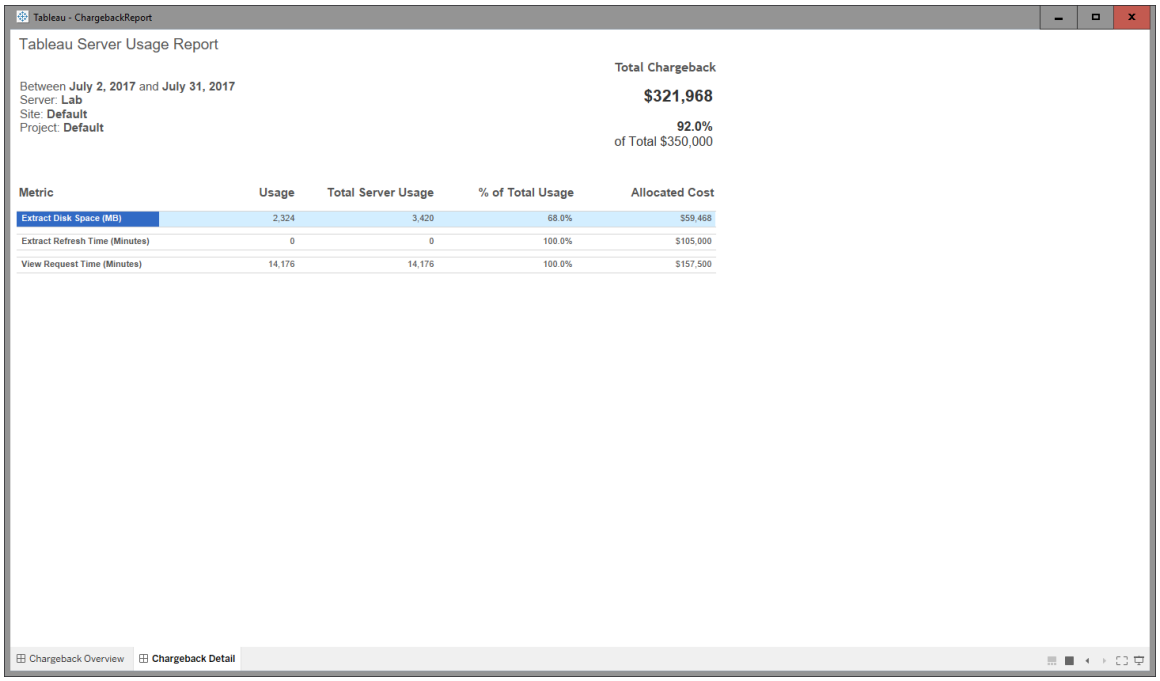
[Chargeback Overview (チャージバックの概要)] ワークシートには、サイトやプロジェクトに基づいてさまざまなメトリクスの内訳が表示されます。



概要

ドル建ての金額は、[合計費用] フィールドを使用し、各メトリクスのコスト加重を調整してカスタマイズできます。

プロジェクトまたはサイトをクリックすると、そのプロジェクト/サイト用のカスタマイズされた請求書形式の詳細レポートに移動します。



詳細

セキュリティ

ユーザー アクセスは、**[Generate Chargeback Reports (チャージバック レポートの作成)]** サーバー ロールを使用して制御されます。

データの生成

チャージバック レポートのデータは、次のデータセットに含まれるメトリクスを使用して、日次レベルで集計されます。

メトリクス	説明
抽出ファイルの利用状況	抽出ファイルのサイズは、1 日に 1 回だけ収集されます (キロバイト単位)。既定のレポートでは、プロジェクト/サイト間での比率/比較の目的で、単純に指定した期間で合計処理されます。
抽出のクエリ期間	抽出ファイルに対するクエリが実行され、日次で合計処理されるのに要した時間 (ミリ秒単

メトリクス	説明
	位)。
抽出更新のタスク期間	抽出更新のバックグラウンドタスクが実行され、日次で合計処理されるのに要した時間(ミリ秒単位)。
クエリ期間	クエリが実行され、日次で合計処理されるのに要した時間(ミリ秒単位)。これには、抽出クエリと非抽出クエリの両方が含まれます。
ビューの要求期間	ビューのレンダリングに対する要求が実行され、日次で合計処理されるのに要した時間(ミリ秒単位)。これには、ビューのレンダリングをブロックしたデータクエリの期間が含まれます。

実行可能なユーザー

"Server/Environment Management (サーバー/環境管理)" の役割を持つResource Monitoring Tool管理者やResource Monitoring Toolユーザー。

Tableau Resource Monitoring Toolの問題のトラブルシューティング

このセクションには、トラブルシューティングの手順とヒントについて説明する記事が含まれています。ここに記載されていない質問やその他の問題が発生した場合は、Tableau カスタマー サポートにお問い合わせください。

ハードウェア パフォーマンス データが見つからない場合のトラブルシューティング

プロセッサー(CPU)の使用率、メモリ使用率、ディスクキュー、ネットワークパフォーマンスの統計情報は、Tableau Resource Monitoring Toolではハードウェア パフォーマンス データと見なされます。このデータを表示する最も一般的な場所は、次の場所です。

- [Performance (パフォーマンス)] グラフと[Tableau Processes (Tableau プロセス)] グラフの[**Environment Overview**] ダッシュボード
- [**Servers**] ダッシュボード

このデータは、それぞれの Tableau Server マシンで実行中の Resource Monitoring Tool エージェントプロセスによってほぼリアルタイムで報告されます。

これらのグラフに長期間データが表示されない場合は、以下の原因が考えられます。

1. Tableau Server の正しいライセンスが設定されていない。これは、Advanced Management 機能がないことが原因である可能性があります。Tableau Server に必要なライセンスがあることを確認してください。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。
2. エージェントと Tableau Server 間の接続に問題がある。以下のステップに従って、これらの問題のトラブルシューティングを行い、問題を特定します。

ステップ 1: エージェントの接続ステータスを確認

まず、現在エージェントが Resource Monitoring Tool に接続していることを確認します。エージェントは、接続ステータスを示す定期的なハートビートメッセージを RMT サーバーに送信します。

1. 管理者として Resource Monitoring Tool にログインします。
2. [**Admin**] > [**Environments**] ページへ移動します。
3. パフォーマンスデータが見つからない環境の [編集] リンクをクリックします。
4. [**Servers**] のリストを探し、各サーバーでエージェントサービスが [**Connected**] と表示されていることを確認します。[**Connected**] のステータスにカーソルを合わせると、最後のハートビートメッセージを受信したタイムスタンプを確認できます。

ステップ 2: エージェントが実行中であることを確認

Resource Monitoring Tool でエージェントが [**Disconnected**] と表示されている場合、エージェントの Windows サービスが実行されていないことがあります。

1. エージェントを実行中のマシンに接続します。
2. Tableau Resource Monitoring Tool エージェントの Windows サービスが実行中であることを確認します。

ステップ 3: エージェントが正しく構成されていることを確認

Resource Monitoring Tool でエージェントが [**Disconnected**] と表示されているもののサービスが実行中の場合、エージェントは Resource Monitoring Tool のメッセージキューに到達できない場合があります。

1. エージェントを実行中のマシンに接続します。
2. エージェントのインストール フォルダーに移動します。例:
   `/opt/tableau/tabrmt/agent`
3. *rmtadmin status* コマンドを実行します。

rmtadmin status コマンドは、エージェントのメッセージ キューへの接続や、Resource Monitoring Tool の RMT サーバーへの接続をテストします。

ステップ 4: エージェントを再起動

場合によっては、エージェントが実行中で、すべてのステータス インジケーターが成功と表示されているにも関わらず、エージェントでは引き続きハードウェア パフォーマンス データが送信されないことがあります。これは既知の問題で、エージェントとメッセージ キューとの間の一時的な接続エラーが原因で発生する可能性があります。たとえば、メッセージ キュー サーバーを再起動している間や、ネットワークが短時間中断しているときなどです。

こうした状況を確認するには、エージェントを再起動し、数分間待ってから、パフォーマンス データが送信されるようになったかどうかを確認してください。

1. エージェントを実行中のマシンに接続します。
2. Tableau Resource Monitoring Tool エージェントの Windows サービスを再起動します。
3. サービスが正常に開始されていることを確認します。
4. 少なくとも 10 分待機してから Resource Monitoring Tool にログインし、[サーバー] ダッシュボードをチェックしてハードウェア パフォーマンス データを受信したかどうかを確認します。

ステップ 5: 実行アカウントの構成を確認

エージェントの実行アカウントが、Tableau Server に接続してパフォーマンスと CPU データを取得するために、Tableau Server が使用するものと同じ実行アカウントを使用するように構成されていることを確認します。アカウントは、Tableau Server が構成されているアカウントと同じである必要があります。このアカウントに、Tableau Server ログにアクセスする権限があることを確認してください。

ステップ 6: サポートへの問い合わせ

上記の手順に従っても問題が解決しない場合は、サポートにお問い合わせください。

RMT サーバーおよび接続の問題が発生している各エージェントから収集した Resource Monitoring Tool のログ ファイルのコピーを、サポート チームに送信する必要があります。ログ ファイルの収集方法と Tableau カスタマー サポートへの送信方法の詳細については、「Tableau カスタマー サポートへのログ ファイルの送信」を参照してください。

実行可能なユーザー

Resource Monitoring Tool 管理者。

RMT サーバーのサービス中断のトラブルシューティング

RMT サーバーの定期的な停止によって、サービスが中断する場合があります。これは、ほとんどの場合、Resource Monitoring Tool が内部の IT ポリシーに準拠していないサービス アカウントで構成されていることが原因です。多くの IT 部門は自動化ツールを採用しており、このツールを使って、標準に準拠していないと見なされるアカウントやサービス中断の原因となっているアカウントの権限を取り消します。

一時的な解決策: `rmtadmin start -master`.を実行して RMT サーバーを再起動します。

長期的な解決策: 内部ガバナンス ポリシーに準拠したアカウントで実行するように Resource Monitoring Tool を構成するか (推奨)、IT 部門と協力してポリシーから例外を取得します。

実行可能なユーザー

構成とデータベースに変更を加えるには、そのマシンの管理者であり、かつ Resource Monitoring Tool 管理者である必要があります。

Tableau Server プロセスのステータスが不明である場合のトラブルシューティング

特定の条件下では、Tableau Server プロセスのステータスが **[Unknown]** と報告される場合があります。多くの場合、Tableau Server のプロセス構成の変更がまだ Tableau Resource Monitoring Tool で更新されていないことが原因です。

> **注:** 構成を更新した後、Resource Monitoring Tool によって報告されたステータスが更新されるのに数分間かかる場合があります。既定では、Resource Monitoring Tool は Tableau

Server のステータスを 15 秒ごとに確認しますが、確認頻度を減らすように構成していると長くなる場合があります。

この問題を解決するには、次のトラブルシューティング手順を使用します。

ステップ 1: Tableau Server 環境の設定の確認

Resource Monitoring Tool は Tableau Server に接続して監視とデータ収集を行います。Tableau Server がアップグレードされる場合や Tableau Server の認証資格情報が有効期限切れの場合、Resource Monitoring Tool は期待通りに監視を行うことができなくなります。

Resource Monitoring Tool が Tableau Server に接続できることを確認するには、次の操作を行います。

1. Go to **Admin > Environments list** page.
2. 問題が発生している環境を編集します。
3. 選択した Tableau Server バージョンが正しいことを確認します。Tableau Server をアップグレードする場合、バージョンを Resource Monitoring Tool で更新する必要があります。
4. [テスト接続] ボタンを使用して Tableau Server REST API 接続をテストします。

   必要に応じ、[テスト接続] ボタンを使用して Tableau リポジトリ接続をテストします。

ステップ 2: Tableau Server マシンとプロセスの更新

Tableau Server の監視を正常に行うには、Tableau Server マシン名とプロセスポートの完全なレコードを使用して Resource Monitoring Tool を構成する必要があります。この情報は新しい環境の作成時に自動収集されますが、Tableau Server の構成変更時には手動更新が必要になる場合があります。

1. **[Admin (管理者)] > [Environments (環境)]** ページに移動します。
2. 問題が発生している環境を編集します。
3. **[サーバー]** のリストに、最初のノード/ゲートウェイである Tableau Server マシンのほか、追加のノードマシンすべてが含まれていることを確認します。
4. 各サーバーが正しく構成されていることを確認するには、次の手順に従います。

ステップ 3: マシン名の更新

1. 次の Tableau Server URL を開きます: *http://<your TableauServer URL>/admin/systeminfo.xml*。
2. このページで Resource Monitoring Tool の構成を比較して更新し、最新の状態になるようにします。
    - サーバーのホスト名を更新し、<machine name="{HOST NAME}" /> 属性値と正確に一致するようにします。

ステップ 4: サポートへの問い合わせ

上記の手順に従っても問題が解決しない場合は、Tableau カスタマー サポートにお問い合わせください。

VizQL セッションの詳細ページに VizQL プロセスが不明と記載されています

Tableau ビューの HTTP 要求は、要求の VizQL セッション ID を最後にロックした VizQL プロセス (実際の VizQL PID) にリンクされています。

場合によっては一致する VizQL プロセス (PID) を見つけることができない場合があり、この場合にはビューのセッション詳細ページに VizQL プロセスが不明であるというメッセージが表示されます。

次のいくつかのまれな状況で発生することがあります。

- 新しい VizQL ワーカー インスタンスの追加により Tableau クラスタが変更されている。
- 別のプロセスと競合しないように、VizQL プロセスのポート番号を Tableau が調整している。

この場合、Resource Monitoring Tool にサインインし、環境管理画面に移動します。サーバーのプロセス定義を確認する必要があります。VizQL プロセスが予期したサーバーで定義されており、ポート番号が正しいことを確認してください。

実行可能なユーザー

Tableau Server プロセスの問題をトラブルシューティングするには、Tableau Server 管理者と Resource Monitoring Tool 管理者の両方の役割が必要です。

ユーザー認証のトラブルシューティング

リソース管理ツールのユーザーが RMT にサインインできない場合は、いくつかの理由が考えられます。トラブルシューティングのステップは、使用している RMT のバージョンと、ユーザーに対して構成されている認証タイプによって異なります。

RMT ユーザー認証に関する問題のトラブルシューティング

ユーザーが Resource Management Tool にサインインできない場合は、以下の点を確認して問題を解決してください。

- 入力しているユーザー名は、RMT でユーザーとして追加されていますか?
- ユーザーが RMT で持っている認証タイプは何ですか?
    - **ローカル:** ユーザーがローカル認証を持っている場合は、パスワードをリセットして新しいパスワードを提供します。
    - **委任** (バージョン 2023.1.0 以降)**:**
        - ユーザーが委任認証を持っている場合は、ユーザー名が RMT に正しく入力されていることを確認してください。ユーザー名の前後にドメインを含めないでください。
            - 正: `<username>`
            - 誤: `<subnet.network>\<username>` または `<username@<subnet>.<network>`
        - ユーザーに、ドメイン認証資格情報を使用する別のシステムにサインインして、認証資格情報を確認してもらいます。
        - サインインの失敗が多すぎてユーザーがロックアウトされていないことを確認します (この問題は RMT では対応できないため、IT ヘルプデスクに問い合わせる必要があります)。

注: バージョン 2025.1.1 以降、ドメイン ロックアウト対策の構成を追加しました。これにより、RMT がサインイン試行を停止する前に、連続してサインイン試行に失敗する回数のしきい値を設定できます。これは、サインイン試行を停止するように RMT を柔軟に設定するとともに、試行の失敗回数が多すぎることによるドメイン ロックアウトを回避するように設計されています。詳細については、「サポートされる構成キー」を参照してください。

ログを使用して認証に関する問題をトラブルシューティングする

認証に関する問題のログは次の場所にあります: `\[Install Directory]\master\logs\web\tabrmt_YYYMMDD.log`

検索するエラー文字列:

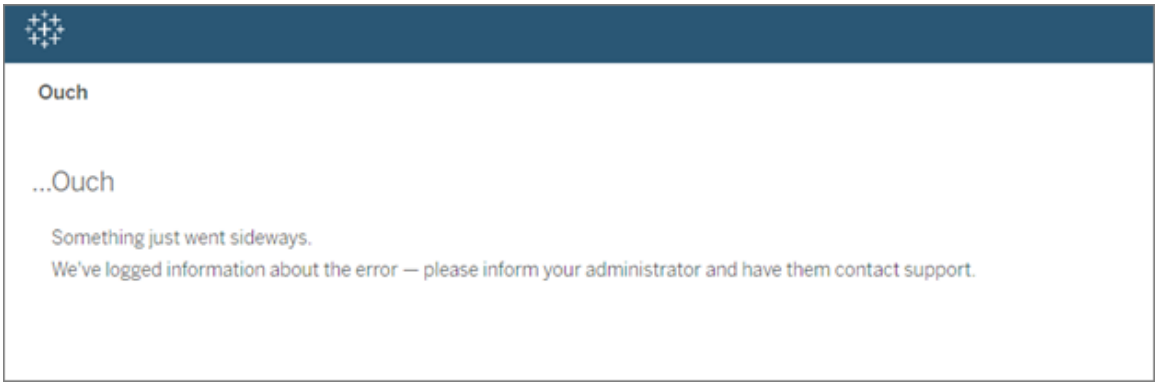
- "ユーザーのパスワードが無効です" - ローカル認証を割り当てたRMTユーザーにマッピングされている有効なユーザー名を入力しましたが、無効なパスワードが入力されました。
- "ユーザーが正しいパスワードを入力できませんでした" - 委任認証を割り当てたRMTユーザーにマッピングされている正しいユーザー名を入力しましたが、無効なパスワードが入力されました。
- "アクション メソッドを実行しています \"Tableau.PowerTools.Server.Master.Web.Controllers.AuthenticationController.Login" - サインインを試みています。

実行可能なユーザー

ユーザー認証に関する問題を解決するには、マシンの管理者であり、かつResource Monitoring Toolの管理である必要があります。

Webインターフェイスのタイムアウトのトラブルシューティング

Tableau Resource Monitoring Tool Webインターフェイスに移動しようとすると、次のエラーが表示されます。



このエラーは、読み込み時間が既定の30秒のしきい値よりも長くかかる場合に発生します。読み込み時間は、次の影響を受ける可能性があります。

- 参照元のPostgreSQLデータベースのサイズ。
- RMTが、サポート対象の現在のスケールをプッシュするTableau Server環境に接続されている場合も、読み込み時間に影響を与える可能性があります。

- ポートの競合。ポートResource Monitoring Tool が、Tableau Server が使用しているポートと競合していないことを確認します。

この問題を解決するには、データ保持時間を変更するか、しきい値を30秒より長い値に変更します。

データ保持期間を変更するには、次の手順を実行します。

1. このトピックの説明に従って、構成 UI に移動します。
2. データ タブに移動します。

   次の2種類のデータ構成があります。

   - 既定で2週間に設定されているデータ保持。これには、詳細なアクティビティデータとチャートに表示されるデータが含まれます。
   - 既定で10年の期間に設定されているレポートデータ。これには、集計された履歴データとチャージバック レポートに使用されるデータが含まれます。
3. Tableau では、データ保持期間を1週間に変更することを推奨しています。

タイムアウトしきい値を変更するには、次の手順を実行します。

1. RMT サーバーがインストールされているマシンにログインします。
2. "マスター" 構成ファイルを見つけます。既定の場所にインストールしたファイルは、次の場所にあります。

   ```
   /var/opt/tableau/tabrmt/master/config.json
   ```

3. キー **db:timeoutSeconds** を追加または変更します。

PostgreSQL データベースのサイズを直ちに縮小する必要がある場合は、rmtadmin コマンド ライン ユーティリティを使用してデータを削除します。詳細については、「rmtadmin コマンド ライン ユーティリティ」を参照してください。

実行可能なユーザー

構成とデータベースに変更を加えるには、そのマシンの管理者であり、かつResource Monitoring Tool 管理者である必要があります。

メッセージング テーブルのディスク使用量の警告のトラブルシューティング

まれに、rmtadmin status コマンドを実行した後に、コマンドラインにディスク使用量の状況に関する警告が表示される場合があります。

```
Internal RMT messaging tables total disk usage is now: <nn.nn> KB.
```

ディスク使用量が 4 GB の警告しきい値に達するかそれを超えると、メッセージが表示されます。

通常、データベースは不要なテーブルを自動的にクリーンアップしますが、この警告が表示された場合は、次の手順に従ってテーブルを手動でクリーンアップしてください。

1. RMT サーバーを停止します:

   ```
   rmtadmin stop
   ```

2. データベース サービスを開始します:

   ```
   rmtadmin start --db
   ```

3. 不要なテーブルをクリーンアップします。

   ```
   rmtadmin query "DROP SCHEMA hangfire CASCADE;" --outfile=drop_
   hangfire_schema_output.zip --commit
   ```

4. RMT サーバーを起動します:

   ```
   rmtadmin start
   ```

実行可能なユーザー

必要なコマンドを実行するには、マシンの管理者であり、Resource Monitoring Tool の管理者である必要があります。

Tableau Content Migration Tool について

この一連の記事では、Tableau Content Migration Tool のセットアップ、使用、およびメンテナンスについて順を追って説明します。

注: Tableau Content Migration Tool は、Tableau Server から Tableau Cloud への移行では推奨されません。Tableau Server から Tableau Cloud に移行するには、Tableau 移行 SDK を使用してください。Tableau Content Migration Tool の推奨される使用例については、Tableau Content Migration Tool の使用事例を参照してください

Content Migration Tool とは

Content Migration Tool を使用すると、Tableau Server サイト間でコンテンツを簡単にコピーまたは移行できます。これは、Tableau Server を 1 回インストールしてサイト間で実行するか、ユーザーベース ライセンスが発行されている場合は、複数の別個のインストールを行ってサイト間 (Tableau Server の開発インスタンスと製品インストール間など) で実行できます。Content Migration Tool ユーザー インターフェイスでは、1 回使用するか、複数の移行のテンプレートとして使用できる「移行計画」の作成に必要な手順を順を追って説明します。

コンテンツを移行する前に、Tableau Blueprint ヘルプの「コンテンツのガバナンス」セクションを確認することをお勧めします。

ヘルプとサポート

このドキュメントで解決できない問題がある場合は、Tableau テクニカル サポートにお問い合わせください。

Tableau Content Migration Tool の使用開始

この記事は、Tableau Content Migration Tool の使用を開始するときに役立ちます。Content Migration Tool をインストールする前に準備する必要がある情報に関するその他の記事へのリンクと、移行計画を設計し、既存のインストールをアップグレードする手順が含まれています。

インストール前

インストール要件

Content Migration Tool は、Windows オペレーティング システムにのみインストールできます。インストール前に、Content Migration Tool がインストールされるコンピューターから Tableau ソース サイ

ト(移行元のサイト) と展開先サイト(移行先のサイト) に接続できる必要があります。展開元サイトと展開先サイトの両方に、有効な Advanced Management ライセンスが必要です。Content Migration Tool のインストールとアップグレードの詳細については、「Tableau Content Migration Tool のインストール」を参照してください。

Tableau Server との互換性

Content Migration Tool では、Tableau Server バージョン 2019.3 以降のコンテンツの移行がサポートされています。

この表は、インストールされている Content Migration Tool のバージョンに基づいて使用可能な Tableau Server のバージョンを示しています。

CMT バージョン	**Tableau Server バージョン**
2025.2x	2023.3x ~ 2025.1.x
2025.1x	2023.3x ~ 2025.1.x
2024.3x	2022.3x ~ 2024.2.x
2024.2x	2022.3x ~ 2024.2.x
2024.1x	2022.1x ~ 2024.1.x
2023.3.x	2021.4.x ~ 2023.3.x
2023.2.x	2021.3.x ~ 2023.2.x
2023.1.x	2021.2.x ~ 2023.1.x
2022.4.x	2021.1.x ~ 2022.4.x
2022.3.x	2020.4.x ~ 2022.3.x
2022.2.x	2020.3.x ~ 2022.2.x
2022.1.x	2020.2.x ~ 2022.1.x
2021.4.x	2020.1.x ~ 2021.4.x

CMT バージョン	Tableau Server バージョン
2021.3.x	2019.4.x ~ 2021.3.x
2021.2.x	2019.3.x ~ 2021.2.x
2021.1.x	2019.3.x ~ 2021.1.x
2020.4.x	2019.3.x ~ 2020.4.x
2020.3.x	2019.3.x ~ 2020.3.x

Tableau Cloud との互換性

Content Migration Tool バージョン 2022.2.1 以降では、すべての Tableau Cloud の展開でコンテンツの移行がサポートされています。最新の機能と修正を活用できるように、Tableau Advanced Management のダウンロードページから最新バージョンをインストールすることをお勧めします。

Tableau コンテンツとの互換性

Content Migration Toolでは、過去 8 つのバージョンの Tableau で保存されたワークブックとパブリッシュされたデータソースを移行できます。既存のデータソースは移行できますが、移行中に変更および修正できるのは、以下の表に示す接続のタイプを使用するデータソースのみです。詳細については、移行計画: ワークブックと移行計画: パブリッシュ済みデータソースに記載されている「データソースの変換」を参照してください。

Actian Matrix	Google ドライブ	Pivotal Greenplum Database
Actian Vectorwise	HortonWorks Hadoop Hive	PostgreSQL
Amazon Athena	HP Vertica	Progress OpenEdge
Amazon Aurora	IBM DB2	Salesforce
Amazon EMR	IBM Netezza	SAP HANA
Amazon Redshift	Map R Hadoop Hive	SAP Sybase ASE
Apache Drill		SAP Sybase IQ

Aster Database	Microsoft Access	Snowflake
Box	Microsoft Analysis Services	Spark SQL
Cloudera Hadoop	Microsoft Excel	統計ファイル
タブ区切りテキストファイル	Microsoft Excel Direct	Tableau 抽出
EXASOL	Microsoft OneDrive	Tableau のパブリッシュされたデータソース
Firebird	Microsoft SQL Server	Teradata
Google アナリティクス	MySQL	テキストファイル
Google BigQuery	OData	Web データコネクタ
Google Cloud SQL	Oracle	その他のデータベース (ODBC)
	Oracle Essbase	

インストール後

コンテンツ移行に伴う制限事項

開始する前に、Content Migration Tool を使用するコンテンツ移行に伴う制限事項について理解しておいてください。詳細については、移行に関する制限事項を参照してください。

移行計画を作成する

Content Migration Toolでは、単一サイト上のプロジェクト間、同じ Tableau Server インスタンス上の新しいサイト、異なる Tableau Server インスタンスに存在するサイトにコンテンツを移行する方法について説明します。作成したプランは保存し、今後の移行に再利用できます。詳細については、「移行計画の概要」を参照してください。

Tableau Content Migration Tool のインストール

Tableau Content Migration Tool のインストールは単純で簡単です。

インストール要件

Content Migration Tool ツールは Windows コンピューターから実行され、有効な Advanced Management ライセンスがある Tableau Cloud サイトと Tableau Server 19.3 以降に接続できます。互換性のあるバージョンの詳細については、Tableau Content Migration Tool の使用開始を参照してください。

Content Migration Tool のインストール先のコンピューターは、以下の要件を満たしている必要があります。

- Microsoft Windows 10 以降 (x64)
- Intel Core i3 または AMD Ryzen 3 (デュアル コア)
- 4 GB 以上のメモリ
- ソース サイトと展開先サイトに接続できる。コンテンツを移行するには、両方のサイトに有効な Advanced Management ライセンスが必要です。
- 2 GB 以上の HDD。`\temp` フォルダーを置くドライブには、1 回の移行で移行されるすべてのコンテンツのコピーを保持するのに十分なディスク容量が必要です。すべてのコンテンツはディスクにローカルに保存され、移行が完了すると削除されます。
- アプリケーションとログを保持する十分な空きディスク領域がある。

さらに、Tableau Server で REST API が有効になっていることを確認します (これが既定です)。有効かどうかを確認するには、`tsm configuration get -k api.server.enabled` コマンドを使用します。戻り値 `true` は、REST API が有効であることを意味します。REST API を有効にするには、`tsm configuration set` コマンドを使用します。詳細については、api.server.enabled を参照してください。

Content Migration Tool のインストール

Content Migration Tool をインストールするには

1. Tableau Advanced Management ダウンロード ページから、使用している Tableau Server バージョンに対応する Content Migration Tool インストーラー (`Tabcmt-64bit-<version>.exe`) をダウンロードします。

2. Content Migration Tool セットアッププログラムを実行します。

   > **注:** Content Migration Tool セットアッププログラムを実行すると、以前のバージョンが上書きされます。

3. EULA を読んだ後、**[I agree to the license terms and conditions]** を選択し、**[Install]** をクリックします。
4. [ユーザーアカウント制御] ダイアログが開く場合は、**[Yes]** をクリックし、インストーラーによる変更を許可します。

Content Migration Tool のアップグレード

最新バージョンの Content Migration Tool にアップグレードすると、新しいバージョンに含まれる最新機能と修正を利用できます。

重要:

- Content Migration Tool セットアッププログラムを実行すると、以前のバージョンが上書きされます。
- Content Migration Tool では、以前のバージョンのサイドバイサイドインストールはサポートされていません。

Content Migration Tool をアップグレードするには:

1. Content Migration Tool がインストールされているマシンにログオンします。開いている Content Migration Tool のインスタンスがある場合は、移行計画を保存して、アプリケーションを終了します。
2. 「インストールContent Migration Tool」に記載されているステップに従って、最新のインストーラーをダウンロードし、アップグレードを完了します。

コマンドラインからの Content Migration Tool のインストール

マシンのローカル管理者は、コマンドラインから Content Migration Tool をインストールできます。

スイッチのインストール

インストーラーのコマンドラインに 1 つ以上のスイッチを指定します。例:

```
Tabcmt-64bit-2022-3-0.exe /quiet /norestart
```

スイッチ	説明	コメント
`/install` | `/repair` | `/uninstall` | `/layout "<directory>"`	セットアップを実行して Content Migration Tool をインストール、修復、またはアンインストールするか、`/layout` を使用して、指定したディレクトリにインストール バンドルの完全なローカル コピーを作成します。	既定では、UI とすべてのプロンプトをインストールして表示します。新規インストールでディレクトリが指定されていない場合は、`C:\Program Files\Tableau\Tableau Content Migration Tool` が使用されます。Content Migration Tool が既にインストールされている場合、セットアップは現在のインストールと同じ場所を使用します。
`/passive`	最小限の UI を使用してセットアップを実行します。プロンプトは表示されません。	Content Migration Tool は、`/passive` モードでインストールされている場合、自動的に起動しません。Content Migration Tool を起動するには、アプリケーションを手動で開きます。
`/quiet` | `/silent`	無人の完全なサイレント モードでセットアップを実行します。UI もプロンプトも表示されません。	Content Migration Tool は、`/silent` モードまたは `/quiet` モードでインストールされている場合、自動的に起動しません。Content Migration Tool を起動するには、アプリケーションを手動で開きます。 **注:** `/silent` または `/quiet` のいずれか (両方ではない) を使用します。
`/norestart`	再起動が必要な場合でも、Windows を再起動せずにセットアップを実行します。	**注:** まれに、このオプションを使用しても再起動を抑制できない場合があります。これは、他のソフトウェアのインストール中など、以前のシステムの再起動がスキップされた場合に生じる可能

		性が最も高くなります。

実行可能なユーザー

マシンで管理者アクセス権を持つユーザー。

Tableau Content Migration Tool の使用

以下の手順は、Tableau Content Migration Tool の使い方について説明しています。

- 移行計画の概要
  - 移行計画: サイト
  - 移行計画: ソースプロジェクト
  - 移行計画: ワークブック
  - 移行計画: パブリッシュ済みデータソース
  - 移行計画: パーミッションと所有権
  - 移行計画: 移行スクリプト
  - 移行計画: 計画のオプション
- Tableau Content Migration Tool コンソール ランナーの使用

Tableau Content Migration Tool の使用事例

Tableau Content Migration Tool は名前が示すように、主にサイト間で Tableau Server コンテンツを移動するために使用されます。ただし、このツールには、コンテンツの移行とメンテナンスに関連するいくつかのタスクを実行するのに最適な多くの機能が含まれています。

注: 多くの使用事例では、環境、サイト、またはプロジェクト間のコンテンツの移動を説明するときに移行という用語を使用します。ただし実際には、Content Migration Tool ではコンテンツがコピーされるだけで、元のコンテンツやソース コンテンツの自動的な削除やアーカイブは行われません。

以下の情報は、Content Migration Tool を活用できる場合の一般的ないくつかの使用事例について説明しています。

コンテンツの昇格

Content Migration Tool を使用して、開発サイト用のコンテンツを作成してから、定期移行を実行し、コンテンツをステージング環境または本番環境に昇格することができます。

コンテンツを本番環境に昇格させるには、次の手順に従ってください。

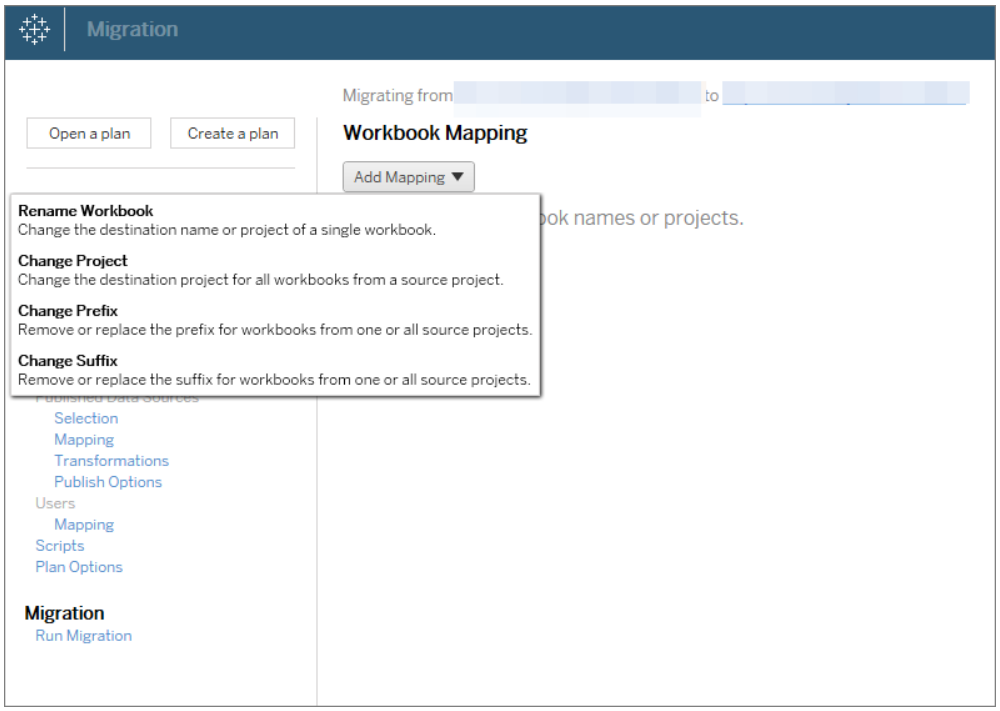
1. **計画を作成**し、開発に使用するサイトをソースとして、ステージング サイトまたは本番サイトを移行先として選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

   同じサイト上の 2 つのプロジェクト間でワークブックを移行する場合、移行元と移行先のサインイン認証資格情報が似ているか同じである場合があります。このシナリオでは、接続を再利用しやすくするために、個人用アクセス トークンを使用することをお勧めします。詳細については、個人用アクセス トークンを参照してください。

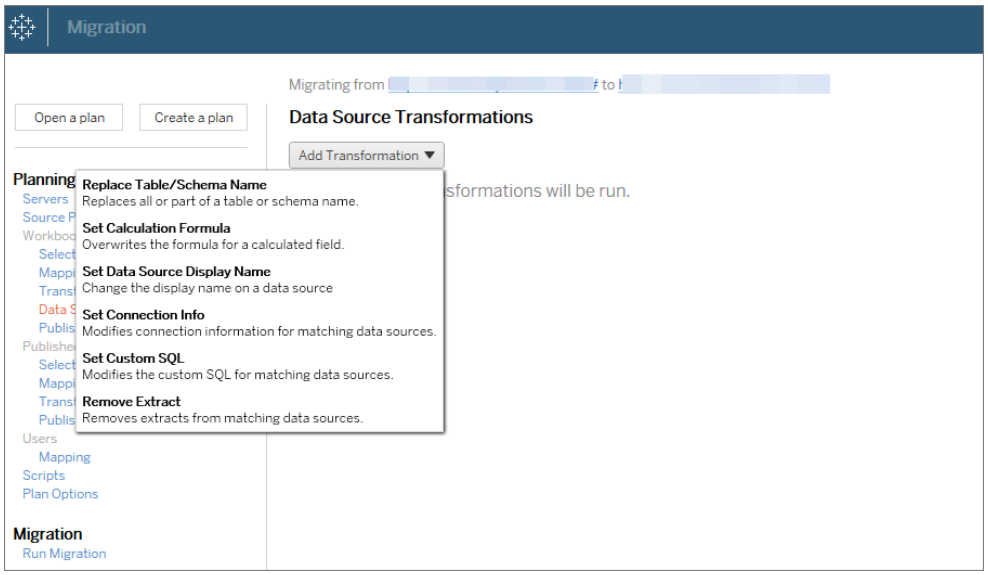
2. ソース サイトから移行する**コンテンツを選択**します。プロジェクト全体、特定のワークブックとデータ ソース、ユーザー パーミッションを選択できます。詳細については、「移行計画の概要」トピックの「計画」を参照してください。

3. この移行中にコンテンツに**変更や変換**を加える必要がある場合は、計画のステップで変更や変換を行う必要があります。このプロセスは**マッピング**と呼ばれます。作成できるマッピングのタイプは次のとおりです。

   - **ワークブックに対する変更:** ワークブック名の変更や移行先プロジェクトの変更が含まれます。ワークブックの変換の完全なリストについては、「移行計画: ワークブック」を参照してください。



- **データ ソースに対する変更:** テーブル名またはスキーマ名の置き換え、計算式の設定、および接続情報の設定が含まれます。データ ソースの変換の完全なリストについては、「移行計画: ワークブック」(埋め込みデータ ソース) および「移行計画: パブリッシュ済みデータ ソース」(パブリッシュされたデータ ソース) を参照してください。



- **ユーザーに対する変更:** 移行先のドメイン、ユーザー、およびグループ名の変更が含まれます。

4. 準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

5. この操作を定期的に実行するように**スケジュールを設定**するには、Content Migration Tool Runner を使用して操作をジョブとしてスクリプト化し、操作のスケジュールを設定します。Content Migration Tool Runner の使用については、「Tableau Content Migration Tool コンソール ランナーの使用」を参照してください。

顧客向けコンテンツの調整

コンサルティング シナリオで作業する場合は、Content Migration Toolを使用して、各顧客のコンテンツをカスタマイズできます。各ワークブックは移行計画のテンプレートとして機能するため、スタイル (テキストや画像など) を適用したり、特定の顧客のデータ ソースを置き換えたりすることができます。

顧客のコンテンツをカスタマイズするには、次の手順に従ってください。

1. **計画を作成**し、本番サイトをソースとして、顧客サイトを移行先として選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

2. 移行計画の**ワークブック**ステップで、ワークブックのマッピングと変換を使用してコンテンツをカスタマイズします。頻繁に使用される変換の 2 つの例を以下に示します。ワークブックの変換の完全なリストについては、「移行計画: ワークブック」を参照してください。

    - コンテンツをパーソナライズする場合は、**[画像の置換]** 変換と**[テキストの置換]** 変換を使用して、ワークブックを顧客の会社名とロゴで更新できます。
    - データソースに関しては、**[テーブル/スキーマ名の置換]** 変換または **[カスタム SQL の設定]** 変換を使用して、顧客のコンテンツを変更できます。

3. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

環境間の移行

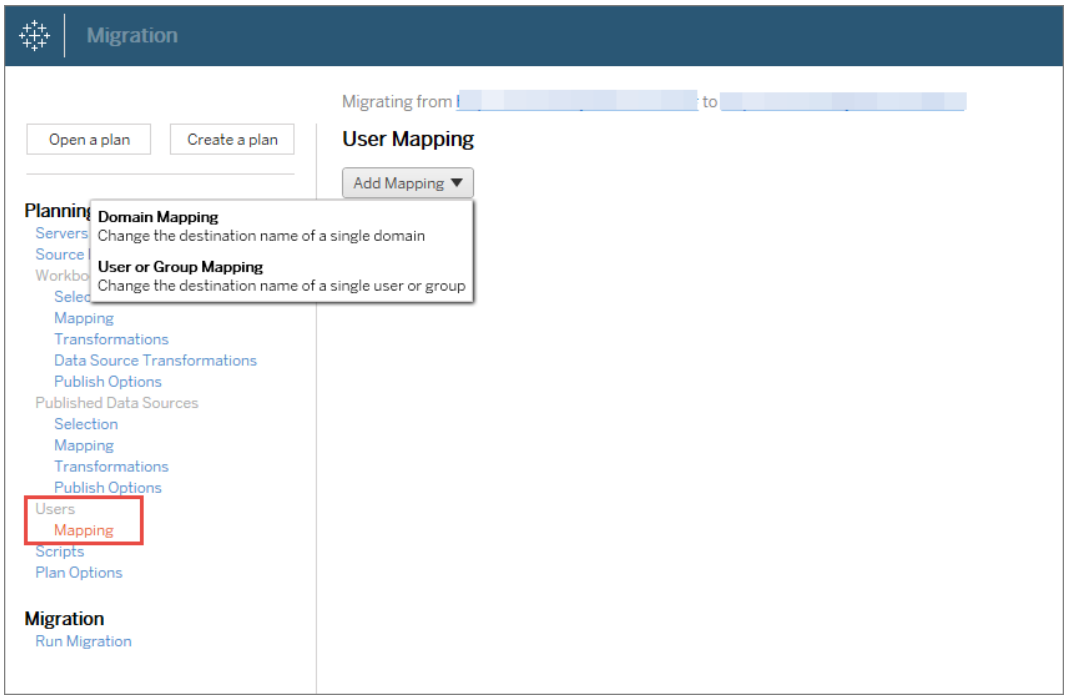
Content Migration Tool を使用して、Tableau Server 環境間でコンテンツを移行できます。

Tableau 展開間でコンテンツを移行するには、次の手順に従ってください。

1. **計画を作成**し、移行元のサイトをソースとして**選択**します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

2. ソースサイトから移行する**コンテンツを選択**します。プロジェクト全体、特定のワークブックとデータソース、ユーザーパーミッションを選択できます。

3. **ユーザーパーミッションのマッピングを作成**して、コンテンツをカスタマイズしてセキュリティで保護します。詳細については、「移行計画: パーミッションと所有権」を参照してください。



4. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

ヒント

- 環境間の移行を実行する前に、Content Migration Tool を使用する際の「移行に関する制限事項」を理解しておいてください。
- 最終的な移行が完了する前に、コンテンツを段階的に移行し、コンテンツをテストして検証できます。この移行方法を使用する場合、サーバーのダウンタイムはありません。この方法は、サイトのインポート/エクスポートの代わりに使用できます。
- Content Migration Tool を使用した移行では、埋め込みの認証資格情報、サブスクリプション、およびカスタム ビューは処理されません。これらは手動で移行する必要があります。

外部コンテンツの共有

Content Migration Tool を使用すると、サイトへのアクセスを許可することなく、外部の共同作業者と内部コンテンツを共有できます。これにより、データのセキュリティが保たれ、選択したワークブックとデータ ソースのみをパブリッシュできます。コンテンツを共有すると、共同作業者は Tableau

Server サイトにサインインし、内部サーバーに保存されているコンテンツに影響を与えることなく、コンテンツを表示したり、変更したりできます。

続行する前に、共有しているコンテンツが内部 (ソース) サイトと外部 (移行先) サイトの間で互換性があることを確認してください。外部サイトは、内部サイトと同じバージョンの Tableau またはそれ以降のバージョンを実行している必要があります。互換性の詳細については 、Tableau Desktop のヘルプの「バージョン間でワークブックの互換性があるようにする」を参照してください。

コンテンツを外部で共有するには、次の手順に従ってください。

1. 外部のサイト管理者と**協力**して、サイトでパブリッシュ権限を持つユーザー アカウントを決定します。このユーザー アカウントを使用して移行計画を作成します。詳細については、「ユーザーのサイト ロールの設定」および「パーミッション」を参照してください。

2. **内部コンテンツを準備**します。ベストプラクティスとして、ロックされたパーミッションと厳格なガバナンス ルールを使用して、内部サーバー上のコンテンツをプロジェクトに分離することをお勧めします。ワークブックとデータ ソースには、コンテンツが外部で使用されることを示す明確なラベルを付ける必要があります。詳細は、「プロジェクトを使用したコンテンツへのアクセスの管理」を参照してください。

   > **注:** 外部サイトと共有されるコンテンツは、データ ソースが公的にアクセス可能でない限り、データ抽出を使用する必要があります。抽出の作成とデータ ソースの置換の詳細については、Tableau Desktop ヘルプの「データの抽出」と「データ ソースの置換」を参照してください。

   行レベルのセキュリティを実装している場合は、これらのデータ ソースを更新して、外部サイトのユーザー フィルターやその他の詳細が反映されるようにする必要があります。行レベルのセキュリティに関する詳細については、Tableau Desktop ヘルプの「データ行レベルでのアクセスの制限」を参照してください。

3. **計画を作成**し、内部サイトをソースとして、外部サイトを移行先として選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

4. 外部サイトと共有する**コンテンツを選択**します。プロジェクト全体、特定のワークブックとデータソース、ユーザー パーミッションを選択できます。

5. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

データベースの移行の検証

この使用事例では、参照元データベースの移行後にコンテンツを検証します。データベースの移行の一例として、SQL Server からSnowflake への移行があります。CMT は、移行を完了する前に、両方のデータソースから構築されたコンテンツが同一であることを検証するのに役立ちますが、データベースの移行を実際に実行することはできません。

データベースの移行を検証するには、次の手順に従ってください。

1. **計画を作成**し、ソースとして使用するTableau サイトを選択します。この例では、同じTableau サイト上のプロジェクト間の移行について説明しているため、移行先と同じサイトを選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

2. **移行を構成**し、コンテンツを新しいプロジェクトにコピーします。**ソース**プロジェクトを**プロジェクトA** と呼び、新しいプロジェクトまたは**移行先**プロジェクトを**プロジェクトB** と呼ぶことにします。

    - ワークブックに対する変更: ワークブックのマッピングを作成して、プロジェクトA をプロジェクトB に変更します。ワークブックの変換の完全なリストについては、「移行計画: ワークブック」を参照してください。

    - データソースに対する変更: データソースのマッピングを作成して、プロジェクトA をプロジェクトB に変更します。データソースの変換の完全なリストについては、「移行計画: パブリッシュ済みデータソース」を参照してください。

3. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

4. **プロジェクトB** のコンテンツを新しいデータベース接続で**更新**するか、データソースを置き換えます。これは、オーサリングによって手動で行う必要があります。

5. **プロジェクトA** の各ワークブックを**プロジェクトB** 内のコピーで**テスト**し、データソースの変更によるデータの不整合がないか確認します。

6. すべてが期待どおりに動作していることを確認したら、プロジェクトAのコンテンツをプロジェクトBの更新されたコンテンツで**上書き**します。

> **注:** コンテンツが移行先プロジェクトに既に存在していて、**[新しい方のワークブックを上書きする]** オプションと**[新しい方のデータソースを上書きする]** オプションを選択しない場合、コンテンツは移行先プロジェクトにコピーされません。

コンテンツの地理的な移行

地理的に分散したマルチサイト環境を維持している場合は、このコンテンツの一部を共有し、すべてのサーバーでアクセスできるようにする必要があります。この使用事例では、異なる地域のサーバー間でコンテンツを移行する方法について説明します。サーバーは、同じ国に配置するか、大陸をまたいで配置できます。

ベストプラクティス:
- 最も必要なコンテンツを優先することをお勧めします。Content Migration Tool を使用してサーバー環境全体を複数の地域にコピーしないでください。
- コンテンツの移行は一方向 (プライマリからセカンダリ) のみにすることをお勧めします。ここで使用するプライマリという用語はソースサイトを、セカンダリは移行先を示しています。複数の移行計画を作成することにより、1 つ以上の移行先を設定できます。

異なる地域に分散されている Tableau Server 間で移行するには、次の手順に従ってください。

1. **計画を作成**し、プライマリサイトをソースとして、セカンダリサイトを移行先として選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

2. プライマリとセカンダリの間で共有する**コンテンツを選択**します。

3. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

4. この操作を定期的に実行するように**スケジュールを設定**するには、Content Migration Tool Runner を使用して操作をジョブとしてスクリプト化し、操作のスケジュールを設定します。Content Migration Tool Runner の使用については、「Tableau Content Migration Tool コンソール ランナーの使用」を参照してください。

5. ソースの**コンテンツを定期的に確認**して、移行計画に新しいアイテムを追加する必要があるかどうかを判断します。

サイトの統合

複数のサイトのコンテンツを 1 つのサイトに組み合わせる必要がある場合 (たとえば、組織の再編成によってサイトの配置方法を変更する必要がある場合)、Content Migration Tool を使用して実行できます。

注: サイトを統合する前に、Content Migration Tool を使用してコンテンツを移行するときの制限事項について理解しておいてください。詳細については、移行に関する制限事項を参照してください。

次の手順を実行して、すべてのワークブックとデータ ソースを 1 つのサイトから別のサイトにコピーします。

1. 計画を作成し、ソースとして統合するサイトを選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

2. [ソース プロジェクト] ページで、**[すべてのプロジェクト]** を選択し、**[次へ]** をクリックします。

   移行先サイトに同じ名前のプロジェクトが存在する場合、コンテンツは同じフォルダーに移行されます。

3. [Project Options (プロジェクト オプション)] ページで、移行先サイトの設定を選択し、**[次へ]** をクリックします。

   コンテンツが移行先プロジェクトに既に存在していて、上書きオプションを選択しない場合、コンテンツは移行先プロジェクトにコピーされません。

4. [Workbook Selection (ワークブックの選択)] 画面で、**[All Workbooks (すべてのワークブック)]** を選択します。

5. (オプション) パブリッシュされたデータ ソースをコピーする場合:

   - 左側のナビゲーション メニューの [パブリッシュされたデータ ソース] で、**[Selection (選択)]** をクリックします。
   - **[すべてのデータ ソース]** を選択します。

6. **[Run Migration (移行の実行)]** をクリックして、移行計画を確認します。準備ができたら、画面の下部にある **[Run (実行)]** をクリックして移行を実行します。

すべてのサイトが統合されるまでこの手順を繰り返します。詳細については、「移行計画の概要」トピックの「計画」を参照してください。

メンテナンス タスク

Content Migration Toolを使用して、さまざまなメンテナンス タスクを実行できます。

古くなったコンテンツのタグ付け

Content Migration Tool を使用すると、古いコンテンツのアーカイブを管理できます。たとえば、[古いコンテンツ] とタグ付けされたコンテンツを自動的に選択して [アーカイブ] プロジェクトに移動する、定期的なスケジュールで実行される計画を作成できます。一定の時間が経過すると、このプロジェクト内のコンテンツをシステムから削除できます。詳細については、「移行計画: ワークブック」を参照してください。

コンテンツの復元

Content Migration Tool を使用して、バックアップ サーバーからのコンテンツが含まれる本番環境の Tableau Server から削除された (誤ってまたは意図的に) コンテンツを復元できます。復元プロセスは単純で、バックアップ ファイルを使用した復元とは異なり、ダウンタイムは不要です。

バックアップ サーバーからコンテンツを復元するには、次の手順に従ってください。

1. **計画を作成**し、バックアップ用の Tableau Server をソースとして、本番環境のサーバーを移行先として選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。
2. バックアップ用の Tableau Server から復元する**コンテンツを選択**します。
3. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。
4. 本番環境のサーバーの**コンテンツを確認**します。

部分的なバックアップ

Tableau Server 環境をバックアップしたら、Content Migration Tool を使用して、新しいコンテンツを本番環境から Tableau Server のバックアップ環境に転送します。Tableau Server のバックアップ環境を構成していない場合、詳細については、ホワイトペーパーの「Tableau Server のディザスタリカバリ」を参照してください。

注:
- 部分的なバックアップを実行する前に、Content Migration Tool を使用する際の「移行に関する制限事項」を理解していることを確認してください。完全なバックアップと復元を定期的に実行して、Tableau Server のすべてのコンテンツをバックアップする必要がある場合があります。詳細については、Tableau Server の完全なバックアップと復元を実行しますを参照してください。
- Content Migration Tool を使用して最初のバックアップを実行しないでください。

コンテンツを部分的バックアップするには、次の手順に従ってください。

1. **計画を作成**し、バックアップ用の Tableau Server をソースとして、本番環境のサーバーを移行先として選択します。詳細については、「移行計画の概要」トピックの計画の作成に関するセクションを参照してください。

2. バックアップする**コンテンツを選択**します。プロジェクト全体、特定のワークブックとデータ ソース、ユーザー パーミッションを選択できます。新しいコンテンツのみを移行するには、パブリッシュ オプションの **[新しい方のワークブックを上書きする]** と **[新しい方のデータ ソースを上書きする]** を選択していないことを確認してください。詳細については、「移行計画: ワークブック」を参照してください。

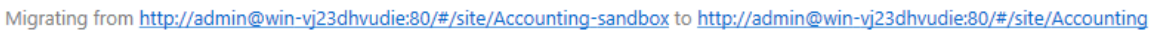
3. 計画を**確認して実行**します。準備ができたら、**[Run Migration (移行の実行)]** をクリックして計画段階を終了し、計画の実行を準備します。

4. この操作を定期的に実行するように**スケジュールを設定**するには、Content Migration Tool Runner を使用して操作をジョブとしてスクリプト化し、操作のスケジュールを設定します。Content Migration Tool Runner の使用については、「Tableau Content Migration Tool コンソール ランナーの使用」を参照してください。

移行計画の概要

Tableau Content Migration Tool は、サイトとプロジェクト間で Tableau コンテンツを移行するための合理的なプロセスを作成します。この移行計画はわかりやすく、監査や反復使用が可能で、バッチ処理により作業を行うため、ワークブックやデータ ソースの数を問わず、シンプルかつ効率的なプロセスで移行することができます。

Content Migration Tool では、移行計画の作成または編集を順を追って説明するヒントが表示されます。移行元サイトと移行先サイトを選択すると、移行の概要が次のように画面の上部に表示されます。

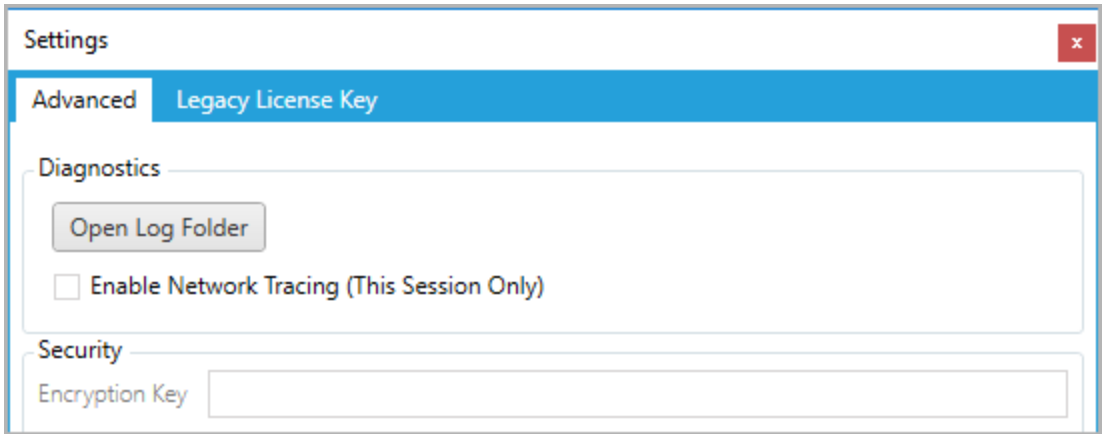


コンテンツ移行に伴う制限事項

開始する前に、Content Migration Tool を使用するコンテンツ移行に伴う制限事項について理解しておいてください。詳細については、移行に関する制限事項を参照してください。

暗号化キー

各移行計画ファイルは、その計画を作成したアプリケーションに固有の暗号化キーと共に生成されます。ファイルの生成に使用したものと異なるアプリケーションを使用して移行計画を実行する必要がある場合は、暗号化キーを共有できます。暗号化キーを共有する際は、移行計画の実行に使用するアプリケーション内の既存のキーを上書きする必要があります。暗号化キーを表示するには、**[Help] > [Settings]** を選択します。



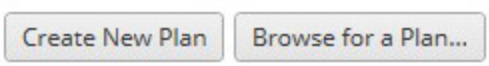
移行計画に Content Migration Tool コンソール ランナーを使用する場合は、計画を実行する前に `tabcmt-runner encryption` コマンドを使用して暗号化キーを指定する必要があります。詳細については、Tableau Content Migration Tool コンソール ランナーの使用を参照してください。

移行プロセス

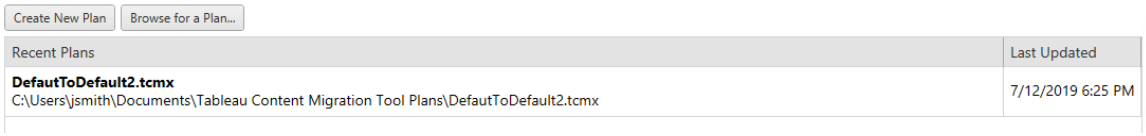
ステップ 1: 開始

移行プロセスの中核部分は計画を作成することです。この計画は保存しておいて将来の移行で再利用することもできますし、必要に応じて変更や更新を行うこともできます。最初の手順は新しい計画を作成するか、以前保存した計画を選択するかを選択することです。

新しい計画を作成するには、**[Create New Plan]** をクリックします。移行計画を既に作成済みでそれを使用する場合は、**[Browse for a Plan]** をクリックします。



既定では、保存された移行計画はすべて、[マイドキュメント] フォルダー内の [Tableau Content Migration Tool Plans] フォルダーに保存されます。すべての移行プランは拡張子 .tcmx により保存され、最近アクセスしたプランは個別にリストされ、簡単に選択できます。



最近アクセスした計画を選択し、それを複製して変更し、新しい計画として保存することができます。コピーする計画を選択し、**[Duplicate]** をクリックします。

ステップ 2: 計画の作成

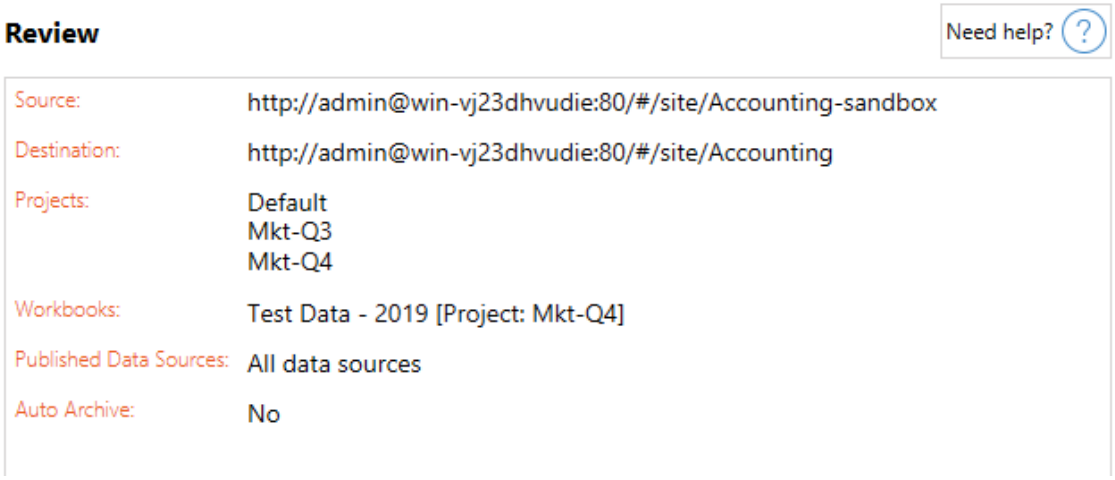
Content Migration Tool では 6 つの手順で移行計画を作成または編集する方法を説明します。

詳細な方法については、各手順をクリックしてください。

- 移行計画: サイト
- 移行計画: ソースプロジェクト
- 移行計画: ワークブック
- 移行計画: パブリッシュ済みデータソース
- 移行計画: パーミッションと所有権
- 移行計画: 移行スクリプト
- 移行計画: 計画のオプション

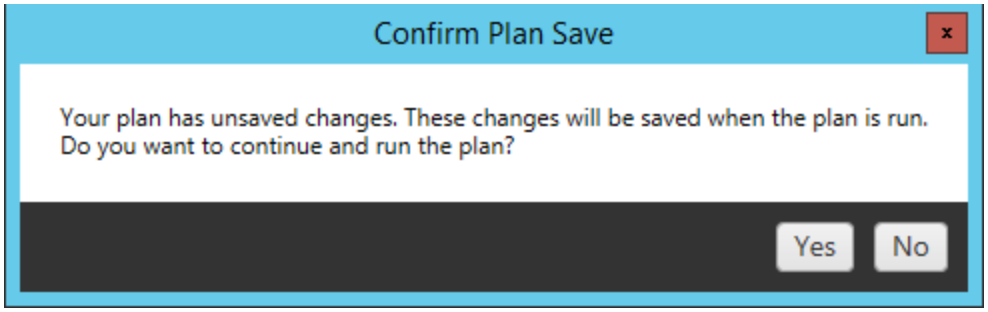
ステップ 3: 移行

計画が完了したら、移行のバッチ処理を実行する準備が整っています。移行の最終手順まで来ると、確認のために計画概要が表示されます。



計画のいずれかの点を変更する場合は、左側のサイドバーのセクションをクリックし、その段階に直接移動することができます。準備ができたら、**[Run]** をクリックして移行を実行します。

[Run] をクリックすると、移行ツールにより、計画の未保存の要素に関するメッセージが表示されます。既定では、**[Yes]** をクリックすると未保存の要素が保存されます。移行プロセスの開始段階で以前の計画を複製することで、変更を加えない状態でいつでも保持できることに留意してください。

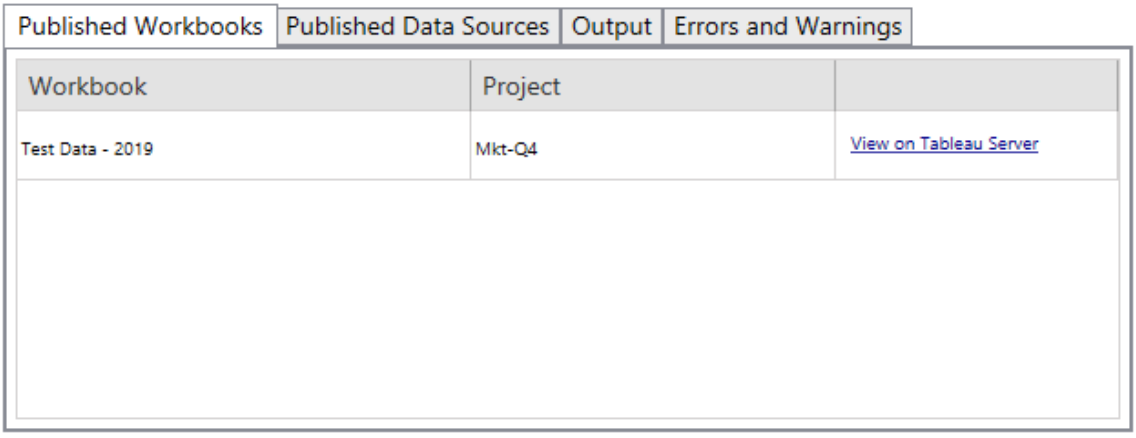


移行計画が実行され、計画全体の進捗状況と展開先サーバーに送信される各ワークブックにはステータスバーが表示されます。

計画の実行が終了したら、画面下部にあるタブをクリックすると移行の詳細を確認できます。

パブリッシュされたワークブック

Published Workbooks 新しくパブリッシュされたワークブックと、それらが移行されたプロジェクトについて記載されています。

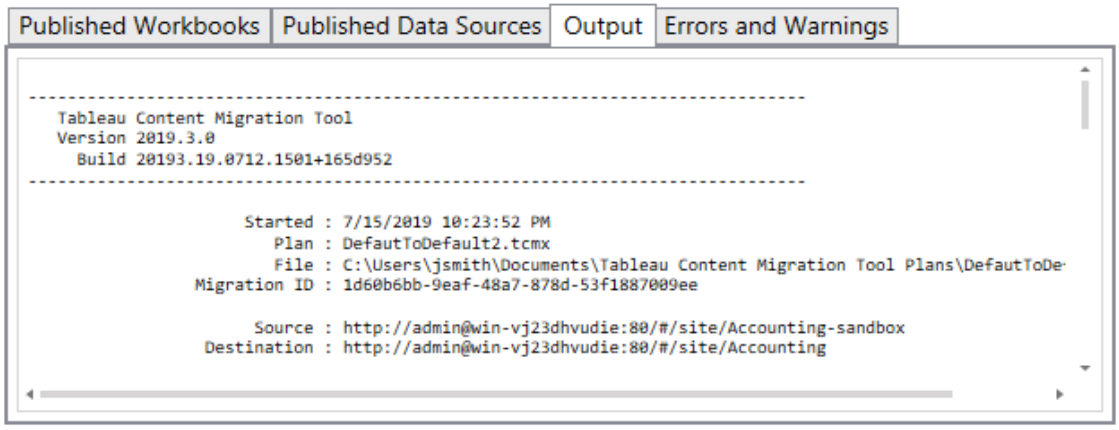


パブリッシュされたデータソース

Published Data Sources 新しくパブリッシュされたデータソースと、それらが移行されたプロジェクトについて記載されています。

出力

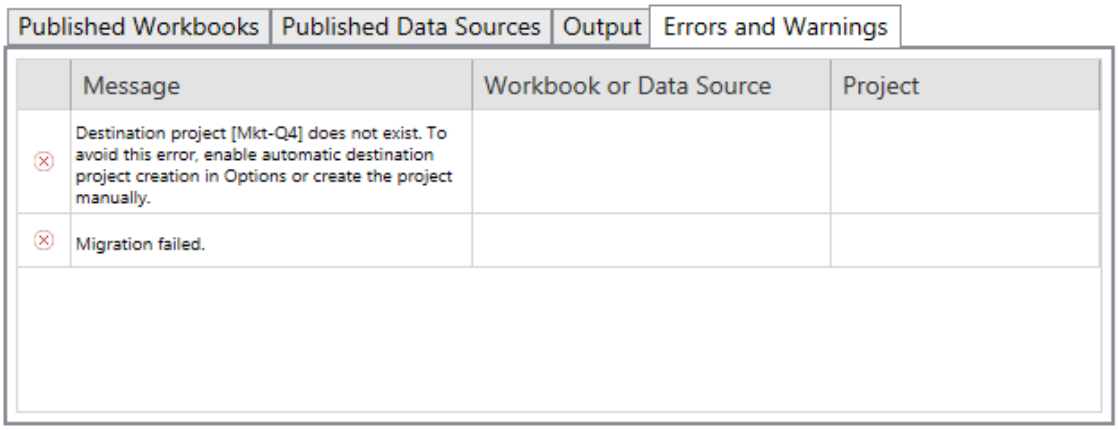
[Output] タブには、計画の移行ログについて記載されています。



このログを保存するには、**[Save Log]** をクリックします。

エラーと警告

[Errors and Warnings] タブは移行中に発生した問題を強調しています。



これらは修正して計画を再実行することができます。移行が完了して計画を保存したら、**[Done]** をクリックして終了します。

実行可能なユーザー

Tableau サイトのユーザー (Explorer 以上の役割を持つ)。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

移行に関する制限事項

Tableau Content Migration Tool を使用した移行には特定の制限事項があります。移行計画を作成する前に、以下のセクションを確認して、バージョンの互換性と移行されないコンテンツについて学習してください。

Tableau コンテンツとの互換性

Content Migration Toolでは、過去 8 つのバージョンの Tableau で保存されたワークブックとパブリッシュされたデータソースを移行できます。CMT では、バージョン 2018.1.x より前に保存されたワークブックとパブリッシュされたデータソースは、サポートされていません。詳細については、Tableau Content Migration Tool の使用開始を参照してください。

Tableau の以前のバージョンでは認識されない機能や形式がコンテンツに含まれている場合、コンテンツを以前のバージョンへ移行しようとすると失敗する可能性があります。非互換性が見つかった場合は Content Migration Tool によって通知され、移行は失敗します。

構成

次の構成は、Content Migration Tool を使用したときに移行先サイトに移行されません。

- ユーザー
- グループ
- サイト設定 (カスタム ロゴ、ビューの推奨事項など)

データ接続

既存のデータソースは移行できますが、移行中に変更および修正できるのは、以下の表に示す接続のタイプを使用するデータソースのみです。詳細については、「移行計画: ワークブック」と「移行計画: パブリッシュ済みデータソース」に記載されている「データソースの変換」を参照してください。

Actian Matrix	Google ドライブ	Pivotal Greenplum Database
Actian Vectorwise	HortonWorks Hadoop Hive	PostgreSQL
Amazon Athena	HP Vertica	Progress OpenEdge
Amazon Aurora	IBM DB2	Salesforce
Amazon EMR	IBM Netezza	SAP HANA
Amazon Redshift	Map R Hadoop Hive	SAP Sybase ASE
Apache Drill	Microsoft Access	SAP Sybase IQ
Aster Database	Microsoft Analysis Services	Snowflake
Box	Microsoft Excel	Spark SQL
Cloudera Hadoop	Microsoft Excel Direct	統計ファイル
タブ区切りテキストファイル	Microsoft OneDrive	Tableau 抽出
EXASOL	Microsoft SQL Server	Tableau Server データソース
Firebird	MySQL	Teradata
Google アナリティクス	OData	テキストファイル
Google BigQuery	Oracle	Web データコネクタ
Google Cloud SQL	Oracle Essbase	その他のデータベース (ODBC)

サポートされないコンテンツ

次のコンテンツは、Content Migration Tool を使用したときに移行先サイトに移行されないため、追加の構成が必要です。

コンテンツ	必要なアクション
「データに聞く」レンズ	ユーザーは、移行先サイトで「データに聞く」レンズをもう一度作成する必要があります。詳細については、「特定の対象者に焦点を当てた「データに聞く」のレンズを作成する」を参照してください。
コレクション	ユーザーは、移行先サイトでコレクションをもう一度作成する必要があります。詳細については、Tableau Desktop ヘルプの「コレクション」を参照してください。
コメント	ユーザーは、宛先先サイトでビューにコメントをもう一度追加する必要があります。詳細については、Tableau Desktop ヘルプの「ビューのコメント」を参照してください。
カスタムビュー	ユーザーは、移行先サイトでカスタム ビューをもう一度作成する必要があります。詳細については、Tableau Desktop ヘルプの「カスタムビューの使用」を参照してください。
データの役割	ユーザーは、移行先サイトでデータの役割をもう一度作成する必要があります。詳細については、Tableau Prep Builder ヘルプの「データの役割を使用したデータの検証」を参照してください。
データソース認証	次のサイトロールと機能がある場合は、移行先サイトでデータソースを認定できます。 • サイト管理者 Creator • データソースが含まれたプロジェクトでプロジェクト リーダー機能を持つ Creator または Explorer (パブリッシュ可能) 詳細については、「ユーザーが信頼できるデータを見つけられるように認証を使用する」を参照してください。
データ	ユーザーは、移行先サイトでダッシュボードとビューのデータ主導アラートをもう一度

主導アラート	作成する必要があります。データ主導アラートが作成されると、ビューにアクセスできるすべてのユーザーが既存のアラートに自分自身を追加できます。 詳細については、Tableau Desktop ヘルプの「Tableau Cloud または Tableau Server からのデータ主導アラートの送信」を参照してください。
ワークブックおよびデータソースの説明	コンテンツアイテムを所有しているか、適切なパーミッションがあれば、移行先サイトでアイテムの説明を編集できます。詳細については、Tableau Desktop ヘルプの「説明の追加または編集」を参照してください。
埋め込み認証資格情報	セキュリティ上の理由から、Tableau Server ではダウンロードプロセス中にデータソースから埋め込みの認証資格情報が削除されます。 • Tableau Server から Tableau Cloud への移行時に埋め込みの認証資格情報を含めるには、[Migrate Embedded Credentials for Workbooks (ワークブックの埋め込み認証資格情報を移行)] および [Migrate Embedded Credentials for Data Source (データソースの埋め込みの認証資格情報を移行)] パブリッシュオプションを使用します。詳細については、「認証資格情報が埋め込まれたワークブックとデータソースの移行」を参照してください。 • Tableau Server サイトへのパブリッシュ時に埋め込みの認証資格情報を含めるには、Set Connection Info (接続情報の設定) データソース変換を使用します。詳細については、移行計画: パブリッシュ済みデータソースを参照してください。 **注:** CMT は、OAuth 接続での埋め込みの認証資格情報の移行をサポートしていません。OAuth 認証資格情報を移行するには、Set Connection Info (接続情報の設定) データソース変換を使用します。
外部資産	外部資産のカスタマイズされた属性は、移行先サイトに移行されません。たとえば、タグ、証明書、データ品質に関する警告、説明、パーミッション、ユーザーの連絡先、テーブル、列をもう一度作成する必要があります。詳細については、「外部資産

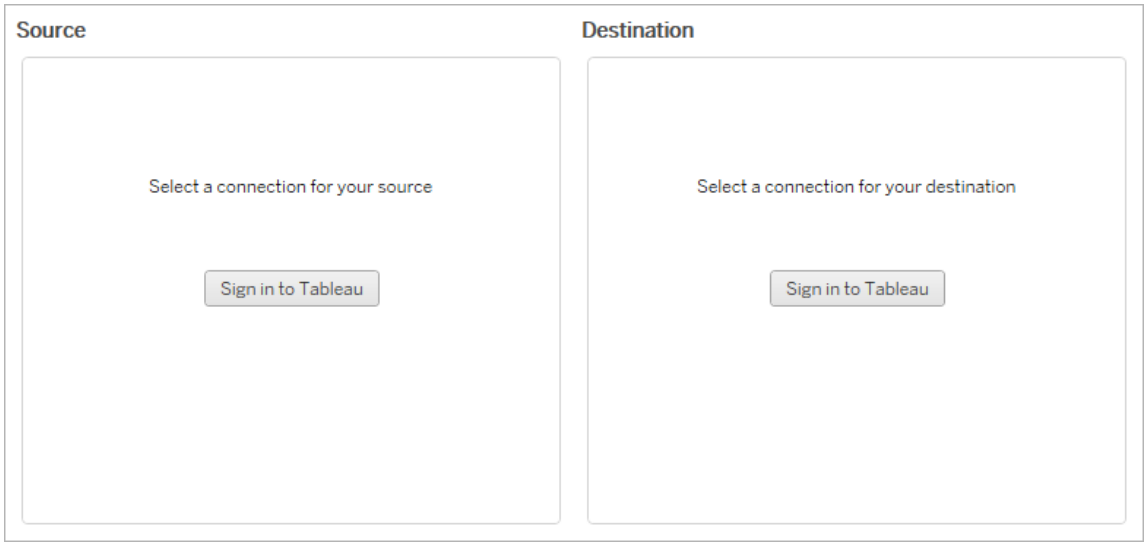
	でのパーミッションの管理」を参照してください。
抽出更新スケジュール	抽出更新スケジュールは、Tableau Cloud 移行先サイトに移行できません。Tableau Cloud でデータを更新するには、抽出更新を手動で実行するか、新しい抽出更新スケジュールを作成します。詳細については、「Tableau Cloud での更新のスケジュール」を参照してください。
お気に入り	ユーザーは、移行先サイトでお気に入りのコンテンツをもう一度選択する必要があります。詳細については、Tableau Desktop ヘルプの「お気に入りとしてマークする」を参照してください。
フロー	フローをスケジュールに従って実行するには、ユーザーは Tableau Prep を使用してフローを移行先サイトにもう一度パブリッシュする必要があります。詳細については、Tableau Prep ヘルプの「Tableau Server または Tableau Cloud へのフローのパブリッシュ」を参照してください。
増分抽出の更新	移行先サイトで、増分抽出の更新が完全抽出の更新に変更されました。ユーザーは、Tableau Desktop で増分更新を再設定し、移行後に移行先サイトに抽出したデータをパブリッシュする必要があります詳細については、Tableau Desktop ヘルプの「抽出の更新」を参照してください。
メトリクス	メトリクスの履歴値はビューから削除されるため、ユーザーは移行先サイトでメトリクスをもう一度作成する必要があります。詳細については、メトリクスの作成とトラブルシューティング (廃止)を参照してください。従来のメトリクス機能は、2024 年 2 月の Tableau Cloud、Tableau Server バージョン 2024.2 で廃止されました。詳細については、「メトリクスの作成とトラブルシューティング (廃止)」を参照してください。
リビジョン履歴	以前のバージョンのワークブックを移行先サイトに移行するには、ユーザーは保持したいバージョンをダウンロードして、ワークブックを移行先サイトにもう一度パブリッシュする必要があります。詳細については、Tableau Desktop ヘルプの「コンテンツのリビジョンの操作」を参照してください。
サブスクリプション	ユーザーは、移行先サイトでビューとワークブックをもう一度サブスクライブする必要があります。詳細については、「ビューまたはワークブックへのサブスクリプションの作成」を参照してください。
ワーク	Content Migration Tool を使用して移行されたワークブックとビューは、移行計画にビューが異なる方法でレンダリングされる変換が含まれている場合でも、元のサムネ

ブックとビューのサムネイル	イルを保持します (データ接続が変更される場合など)。 サムネイルを更新するには、展開先サイトでワークブックまたはビューを編集し、もう一度保存します。詳細については、Tableau Desktop ヘルプの「Web での Tableau ビューの編集」を参照してください。
仮想接続	ユーザーは、移行先サイトで仮想接続をもう一度作成する必要があります。詳細については、「仮想接続の作成」を参照してください。

移行計画: サイト

Tableau Content Migration Tool で移行計画を作成するときの最初のステップは、展開元サイトと展開先サイトにサインインすることです。

計画フェーズの [Sites] セクションで、展開元サイトと展開先サイトにサインインします。使用するユーザー認証資格情報の権限によって、移行計画を作成するときに表示されるサイトとプロジェクトを管理します。ユーザーがアクセスできるコンテンツのみを移行できます。



必要なパーミッションとライセンス

展開元サイトと展開先サイトにサインインするために使用するユーザー アカウントには、Explorer 以上のロールに加え、移行するコンテンツに対する次の権限が必要です。

- ビュー
- ワークブックのダウンロード/コピーを保存
- オプション: 管理者 (ワークブックを選択する、ユーザー リストにアクセスする)

移行元サイトと移行先サイトの両方に、Advanced Management 機能が必要です。詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

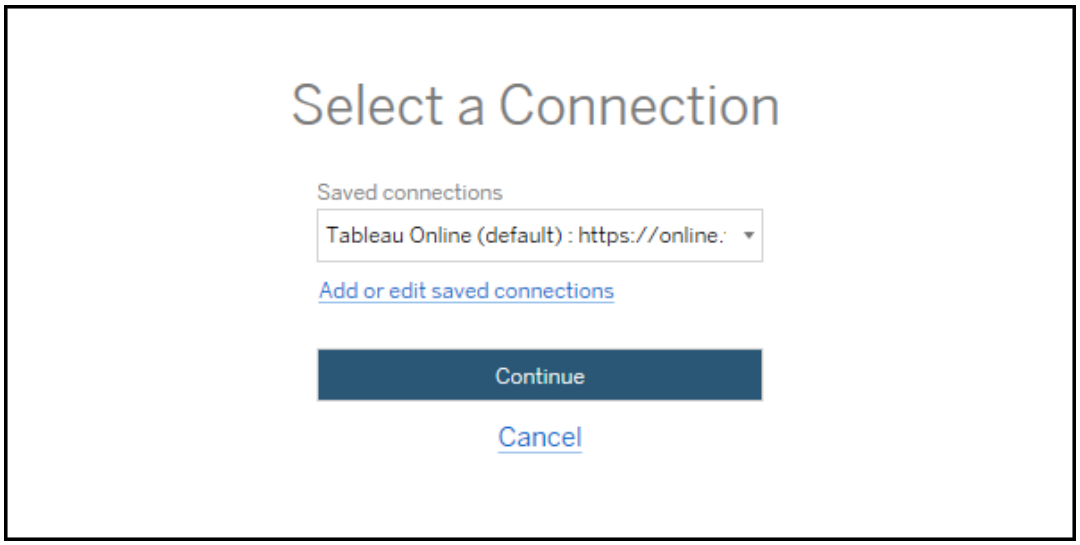
ステップ 1: ソース

移行の開始点です。Tableau Server にサインインし、展開元として使用するサイトを選択します。サイトは、選択したユーザーに関連するコンテンツをグループ化するために Tableau 内で作成したものであり、ワークブック、データ、ユーザー リストなどを集めた専用の場所です。コンテンツを移行できるのは、一度に 1 つのサイトのみです。

展開元サイトへのサインイン

次の手順を実行して、Tableau Cloud または Tableau Server にサインインします。シングル サインオンや Tableau 多要素認証 (MFA) を使用して Tableau Cloud にサインインする方法については、「Tableau Cloud へのサインイン」を参照してください。

1. **[Tableau にサインイン]** をクリックします。
2. **[接続の選択]** ダイアログ ウィンドウで、保存された接続を選択して **[続ける]** をクリックします。



利用できる接続がない場合は、**[保存された接続の追加または編集]** をクリックして、新しい接続を作成します。詳細については、「保存された接続」を参照してください。

3. ユーザー名とパスワードを入力し、**[サインイン]** をクリックします。

   サーバーを SAML またはシングル サインオンで構成している場合は、ID プロバイダーのサインイン ページにリダイレクトされ、認証プロセスを完了します。

4. 使用するサイトを選択します。

ソース サーバーを変更したり、サイトの選択を更新したりするには、**[別のソースを選択]** をクリックします。

ステップ 2: 展開先

展開先サイト (コンテンツの移行先であるサイト) でもサインイン プロセスを繰り返します。

同じ Tableau サイト上の 2 つのプロジェクト間でワークブックを移行する場合、展開元サイトと展開先サイトのサインイン認証資格情報は、サーバーの URL とサイト名も含め、同一です。

保存された接続

保存された接続を使用すると、再利用可能なサーバー接続を作成することができ、展開元サイトと展開先サイトにすぐにサインインできるようになります。保存された接続を追加するには、ご利用のサイトで利用したいサインイン方式を指定する必要があります。

2021.2 以降、Content Migration Tool は次のサインイン方式をサポートしています。

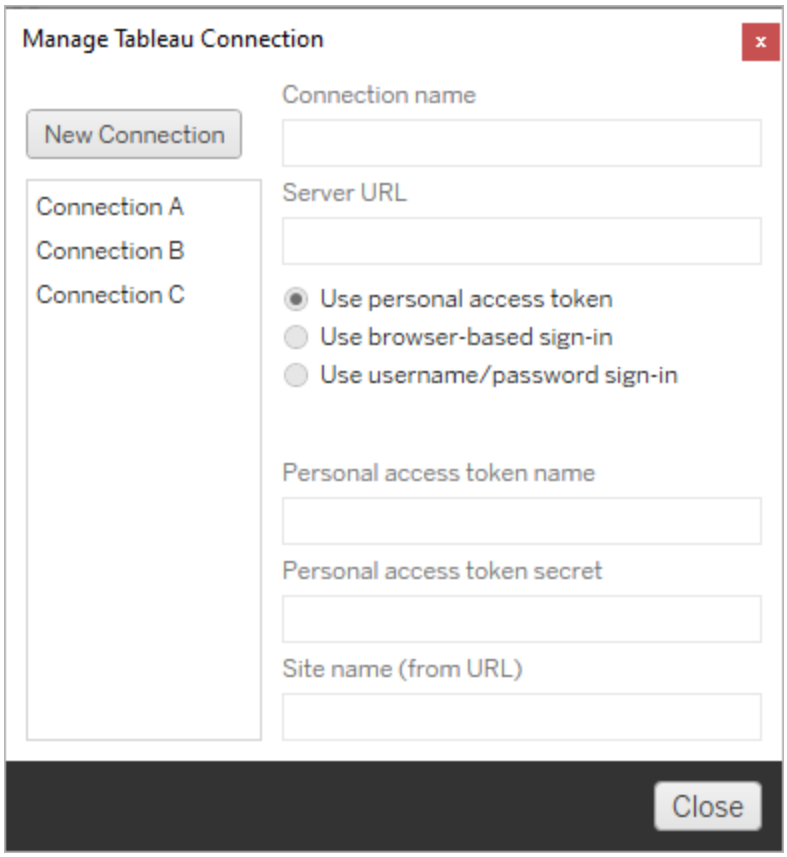
- **個人用アクセス トークン:** 長期間有効な認証トークンを作成することにより、セキュリティの向上、監査、移行計画の自動化などが可能になります。個人用アクセス トークンを使用すると、ユーザーは Content Migration Tool で対話式のログインを繰り返さなくてもサインインできるようになります。詳細については、個人用アクセス トークンを参照してください。

- **ブラウザー ベースのサインイン:** ユーザーは、組み込みの Web ブラウザーを介して認証資格情報を入力し、認証を完了します。このオプションは、通常 Tableau に対して認証する方法と似ています。

- **ユーザー名とパスワードによるサインイン:** ユーザーは、組み込みのブラウザー画面ではなく、Content Migration Tool を使用して認証します。このオプションは、Tableau REST API を使用してサーバーに認証資格情報を渡します。ユーザー名とパスワードによるサインインを使用すると、バージョン 2020.3 より前に作成された移行計画にサインインしたり、ブラウザー ベースのサインインを使用できないような問題をトラブルシューティングするときにサインインしたりできます。

保存されている接続の追加または編集

「**Add or edit saved connections**」へのリンクは、Content Migration Tool の下部に、展開元サイトや展開先サイトにサインインするときに表示されます。このリンクをクリックすると、**[Manage Tableau Connections]** ウィンドウが開きます。



保存された接続を追加するには、以下の手順を使用します。

1. **[Tableau 接続の管理]** ウィンドウで **[新しい接続]** をクリックするか、既存の接続を選択して変更を加えます。
2. **[接続名]** (サーバーを説明する名前) と **[サーバー URL]** を入力します。

   サーバー URL のプレフィックスを含めなければ、Content Migration Tool は `http://` を使用します。
3. 接続のサインイン方式を選択します。

   個人用アクセストークンを使用している場合は、「個人用アクセストークンを使用して保存された接続を追加する」を参照してください。
4. **[保存]** をクリックします。

保存された接続を作成すると、展開元サイトや展開先サイトに次回サインインするときに [接続の選択] ウィンドウに表示されます。

個人用アクセストークンを使用して保存された接続を追加する

個人用アクセストークンを使用して保存された接続を追加するには、他のサインイン方式よりも多くの情報が必要です。開始するには、展開元サイトと展開先サイトで新しい個人用アクセストークンを作成する必要があります。個人用アクセストークンは、アプリケーション間で共有しないでください。詳細については、個人用アクセストークンを参照してください。

個人用アクセストークンの作成

1. Web ブラウザーで Tableau サイトにサインインします。
2. ページの一番上でプロファイル画像またはイニシャルをクリックし、**[アカウント設定]** を選択します。
3. **[個人用アクセス トークン]** の下の **[トークン名]** フィールドにトークンのわかりやすい名前を入力し、**[新しいトークンの作成]** をクリックします。
4. 結果のウィンドウで **[クリップボードにコピー]** をクリックし、ウィンドウを閉じます。
5. トークン シークレットをファイルに貼り付けます。ファイルを安全な場所に保存します。

個人用アクセストークンの追加

1. Content Migration Tool で、**[保存された接続の追加または編集]** をクリックします。
2. **[Tableau 接続の管理]** ウィンドウで、**接続名** と **サーバー URL** を入力します。

   Tableau Cloud に接続している場合は、サイトの完全なポッド URL を入力する必要があります。たとえば、`https://10ay.online.tableau.com` と入力します。Tableau Cloud にサインインすると、サイト URL の最初の部分にポッドが表示されます。

3. 前のセクションで個人用アクセス トークンを作成したときに取得した、**個人用アクセス トークン名**と**個人用アクセス トークン シークレット**を入力します。

4. **[サイト名]** フィールドに、URL に表示されているとおりのサイト名をスペースなしで入力します。これは、フレンドリーなサイト名とは異なります。たとえば、「サイト A」は、ブラウザーの URL では「sitea」になります。

5. **[保存]** をクリックします。

ステップ 3: 次のステップに進む

展開元サイトと展開先サイトの両方に正常にサインインした後、**[Next]** をクリックして、計画フェーズの [移行計画: ソース プロジェクト] セクションに進みます。

実行可能なユーザー

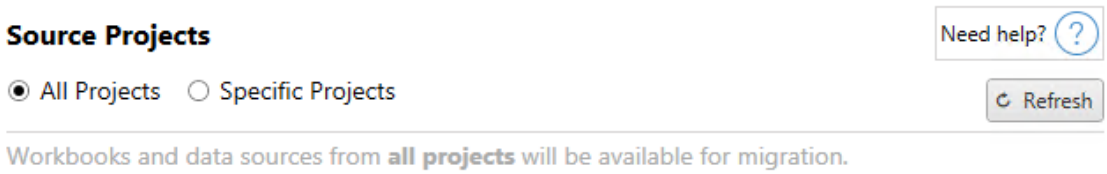
Explorer 以上の役割を持つ Tableau サイト ユーザー。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲット プロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

移行計画: ソース プロジェクト

Tableau Content Migration Tool で移行計画を作成する次の手順は、ソース プロジェクトを選択することです。ソース プロジェクトとは、ワークブックとパブリッシュ済みデータ ソースの移行元となるプロジェクトです。選択するプロジェクトによって、移行計画の次のステップでどのワークブックを移行に使用できるかが決まります。

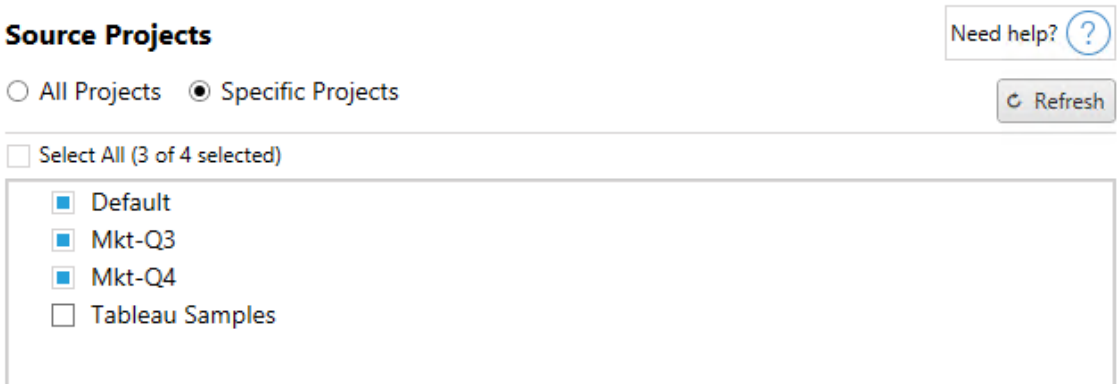
ステップ 1: ソース プロジェクトを選択する

ソース プロジェクトを選択する際には、[**All Projects**] と [**Specific Projects**] の 2 つのオプションがあります。

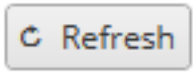


[**All Projects**] オプションでは、サーバー ステップで指定したソース サイトからすべてのプロジェクトを選択します。[**Specific Projects**] オプションを使用すると、ソース サイトから特定のプロジェクトを選択することができます。

> **注:** ソース プロジェクトには、ワークブックまたはデータ ソースが含まれている必要があります。Content Migration Tool では、空のプロジェクトは移行されません。

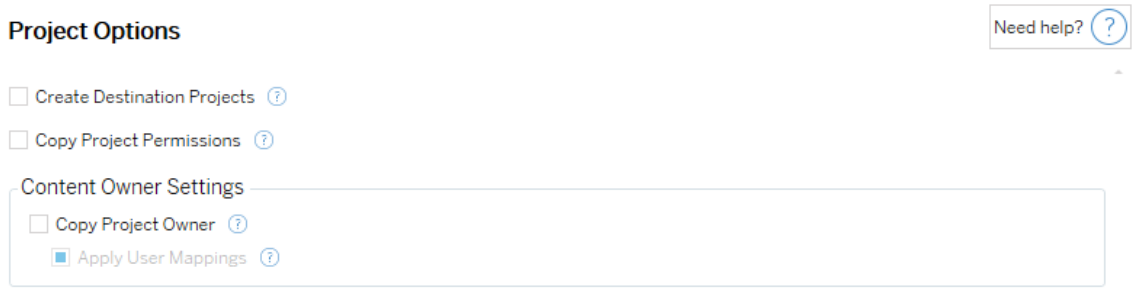


各プロジェクトを個別に選択することも、[**Select All**] ボタンを使用してから追加しないプロジェクトの選択をオフにすることもできます。この手順でソース サイトに変更を加える場合は、[**Refresh**] ボタンを使用してプロジェクト リストを更新できます。



ステップ 2: プロジェクト オプションを選択する

ソース プロジェクトを選択したら、移行先の場所に適用するプロジェクト オプションを選択します。ソースの場所からプロジェクト パーミッションと所有権をコピーすることに加えて、存在しないプロジェクトを作成するオプションがあります。ユーザー マッピングに基づいて新しいコンテンツの所有権を割り当てるには、**[ユーザー マッピングを適用]** を選択します。



- **[移行先プロジェクトを作成]**: 移行先の場所に存在しないプロジェクトを自動的に作成します。Content Migration Tool では、ソースプロジェクトが空の場合、またはワークブックやデータソースが選択されていない場合、移行先プロジェクトは作成されません。デフォルトでは、存在しないプロジェクトに移行しようとすると、移行が失敗します。
- **[プロジェクトのパーミッションをコピー]**: ソースプロジェクトのパーミッションを可能な限り厳密にコピーします。
- **[プロジェクトの所有者をコピー]**: プロジェクトの所有権設定をソースの場所からコピーして、プロジェクトの所有者を割り当てます。
- **[ユーザー マッピングを適用]**: ユーザー マッピングを適用して、移行先の場所にプロジェクトのコンテンツ所有権を割り当てます。移行先のプロジェクトがすでに存在する場合、コンテンツの所有権は適用されません。詳細については、移行計画: パーミッションと所有権を参照してください。

ステップ 3: 次のステップに進む

ソースプロジェクトを選択したら、**[Next]** をクリックし、計画段階の移行計画: ワークブックセクションに進みます。同じ Tableau サイト上の 2 つのプロジェクト間でワークブックを移行する場合は、次のセクションで移行先プロジェクトを選択します。

実行可能なユーザー

Tableau サイトのユーザー (Explorer 以上の役割を持つ)。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

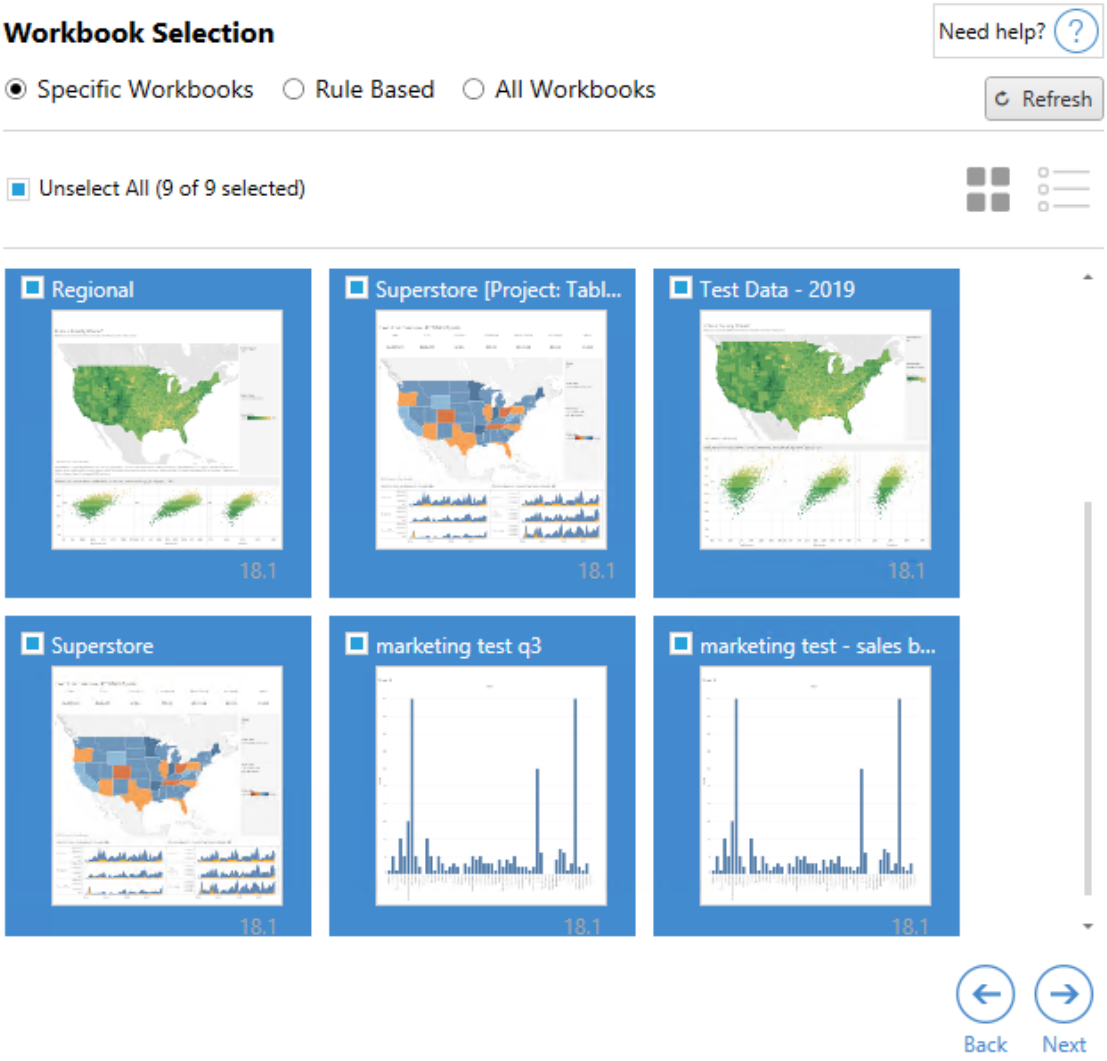
移行計画: ワークブック

展開元サイトと展開先サイト、および選択したプロジェクトに正常にサインインしました。次の手順では、移行用のワークブックを準備します。

> **注:** ワークブックまたはデータソースに抽出が含まれている場合は、抽出を使用するワークブックとデータソースの移行に記載された情報を読んで理解しておいてください。

ステップ 1: ワークブックの選択

展開元サイトと選択したプロジェクト内にあるすべてのワークブックが **[Workbook Selection]** 画面に表示されるようになります。



この手順でソースサイトに含まれるワークブックに変更を加える場合は、**[Refresh]** をクリックして、ワークブックのリストから更新できます。これらのワークブックを選択する方法はいくつかあります。

特定のワークブックの選択

[Specific] セクションには 3 つのボタンがあります。[基本] セクションから選択すると、移行計画用に選択したワークブックが即時に追加されます。または、各ワークブックをクリックして個別に選択することもできます。

すべて選択

このボタンによって、サイト内のすべてのワークブックが選択または選択解除されます。計画を保存した後でサイトにワークブックを追加しても、次に計画に使用したときに自動的に追加されることはありません。

表示:

サムネイル

既定のビューではサムネイルプレビューでワークブックが表示されるため、各ワークブックを区別するのに便利です。このビューでサムネイルにマウスカーソルを合わせると、ワークブック内のその他のワークシートとダッシュボードのプレビューが表示されます。

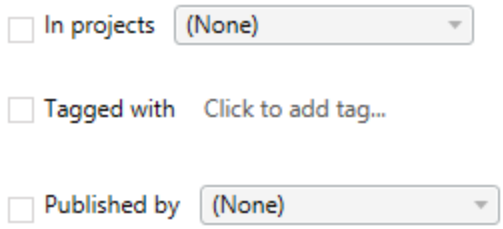
リスト

リストビューは、ワークブック名、プロジェクト、Tableau のバージョン、最終変更日などの追加情報を提供する、より簡潔なリストです。

列ヘッダーのいずれかをクリックすると、ワークブックが適切に並べ替えされます。また、ワークブックにマウスカーソルを合わせると、そのワークブック内のワークシートとダッシュボードの浮動プレビューも表示されます。リストビューは、1 つのサイト内に大量のワークブックがある場合に特に便利です。

ルールベースの選択

特定の条件に基づいてワークブックを選択するには、**Rule Based** の選択を使用できます。ルールベースのオプションを使用すると、移行計画の実行時に使用されるワークブックの選択条件が作成されます。**[Rule Based]** オプションのいずれかで "すべて" を選択することは、**[Specific Workbooks]** を選択することとは異なります。ルールベースの "すべて" を選択すると常にすべてのワークブックが含まれるため、新しく追加されたワークブックが今後の移行に含められます。



[Rule Based] ラジオ ボタンでは、次のオプションを使用してワークブックを選択できます。

Workbooks in projects

このメニューでは、特定のプロジェクトからワークブックを選択できます。

Workbooks tagged with

このメニューでは、タグ別にワークブックを選択できます。

Workbooks published by

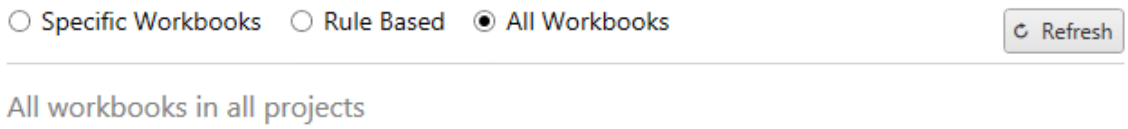
このメニューでは、作成者別にワークブックを選択できます。

各オプションで、各エントリの横にあるオプションをオンにすることで個別または複数のワークブックを選択できます。選択したワークブックは、すべて **[Selection Description]** ボックスに表示されます。

すべてのワークブックの選択

最後のオプションは、**[All Workbooks]** ラジオ ボタンを選択することです。これによりサイト内のすべてのプロジェクトにあるすべてのワークブックを選択します。

[All Workbooks] ラジオ ボタンを使用することは、**[Specific Workbook]** 方式を使用してすべてのワークブックを選択することとは異なります。これは、今後移行計画を使用するたびに展開元サイト内のすべてのワークブックを使用するようになるためです。

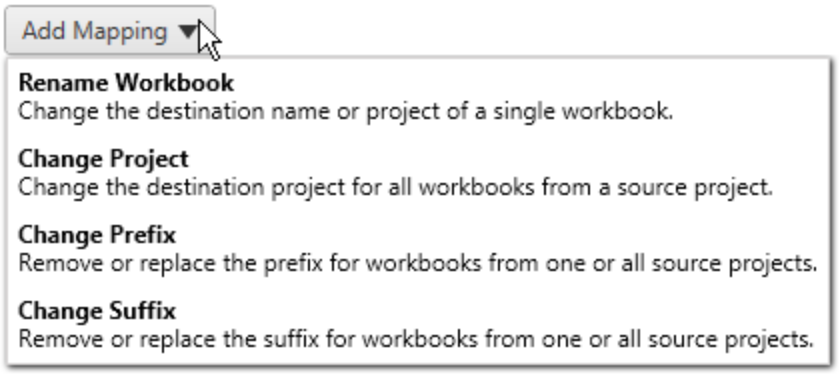


ワークブックの選択に満足している場合には、**[Next]** をクリックします。

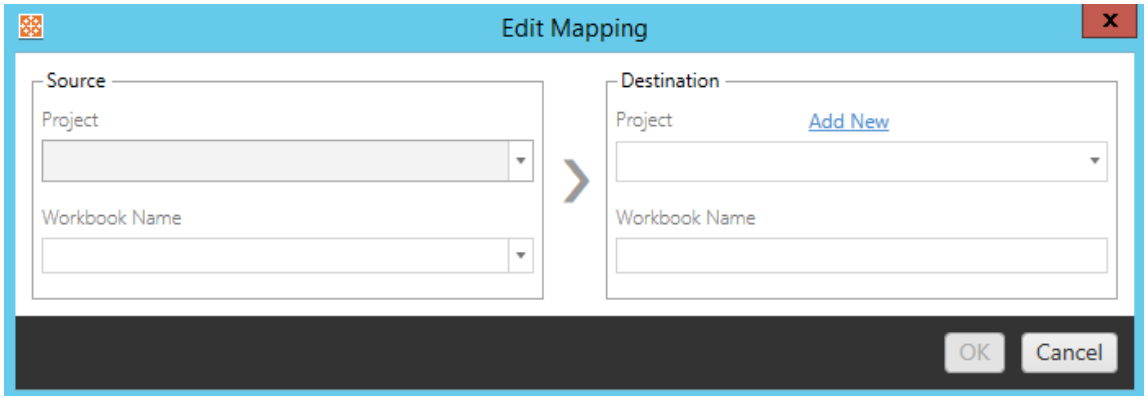
ステップ 2: ワークブックのマッピング

これで、選択したワークブックをソース ファイルから展開先ファイルにマッピングできるようになりました。マッピングによって、ソース ワークブックを移行するときに名前を変更したり別の展開先を選択したりできます。また、マッピングを追加してワークブックのプロジェクト、プレフィックス、またはサフィックスを変更することもできます。プロジェクトをこのセクションの展開先にも追加することができます。

ここで変更しない場合、選択したワークブックは、同じ名前を使用してソースと同じプロジェクトに移行されます。プロジェクトを展開先サイトで定義していない場合は、デフォルトのプロジェクトに移行されます。ワークブックのマッピングを追加するには、**[Add Mapping (マッピングの追加)]** ボタンをクリックします。マッピング領域に次のオプションが表示されます。

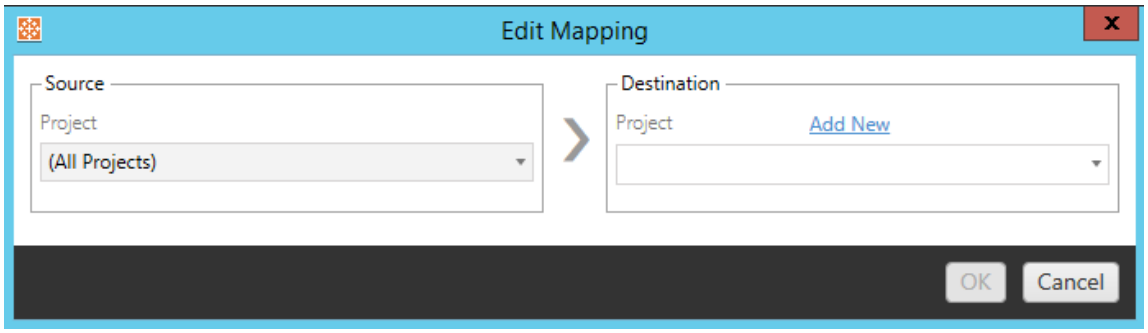


ワークブック名の変更



この変換により、**[Source]** プロジェクトでフィルタリングし、名前を変更する対象のワークブックを選択できるようになります。**[Destination]** フィールドでワークブックの送信先となるプロジェクトを選択し、所望の名前を入力します。

プロジェクトの変更

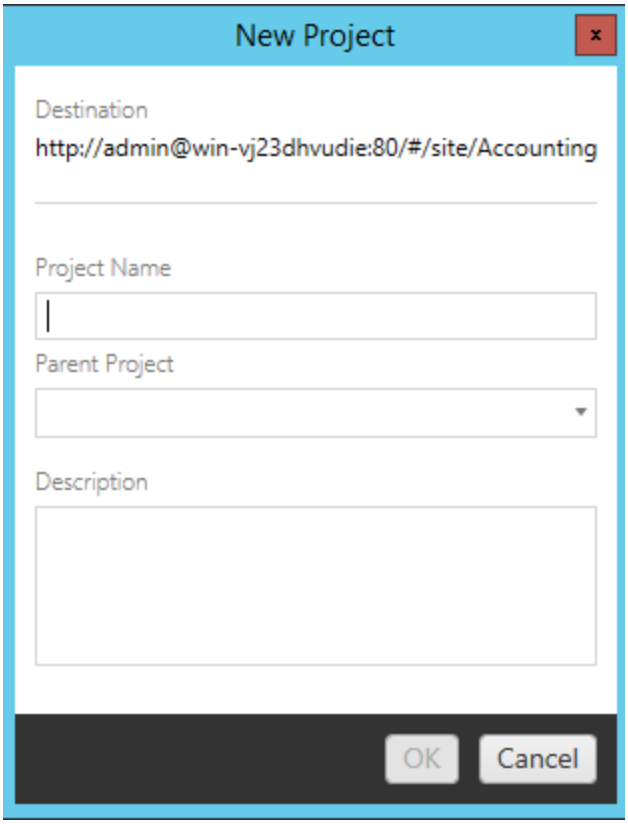


既定では、ワークブックが展開先の同じプロジェクトに移行されます。このマッピングにより、ソースプロジェクトのすべてのワークブックで展開先プロジェクトを変更できます。

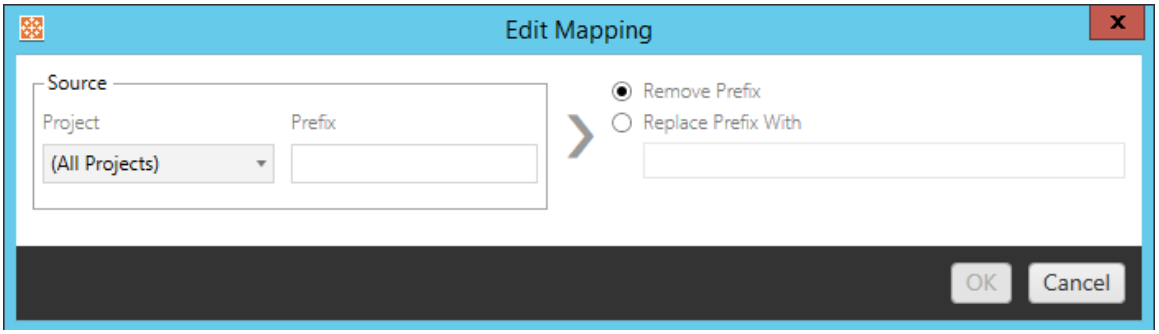
Add Project (プロジェクトの追加)

ワークブック名を変更する際、またはプロジェクトを変更する際に **[Add New]** オプションを使用すると、展開先のサイトにサインインしてプロジェクトを手動で作成することなく、展開先プロジェクトを

作成できます。**[Add New]** ダイアログ ボックスを使用すると、プロジェクトおよびネストされたプロジェクトを作成できます。



プレフィックスの変更



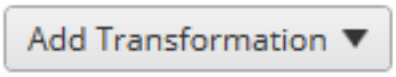
1 つまたはすべてのソース プロジェクトからワークブックのプレフィックスを削除または置換できます。

サフィックスの変更

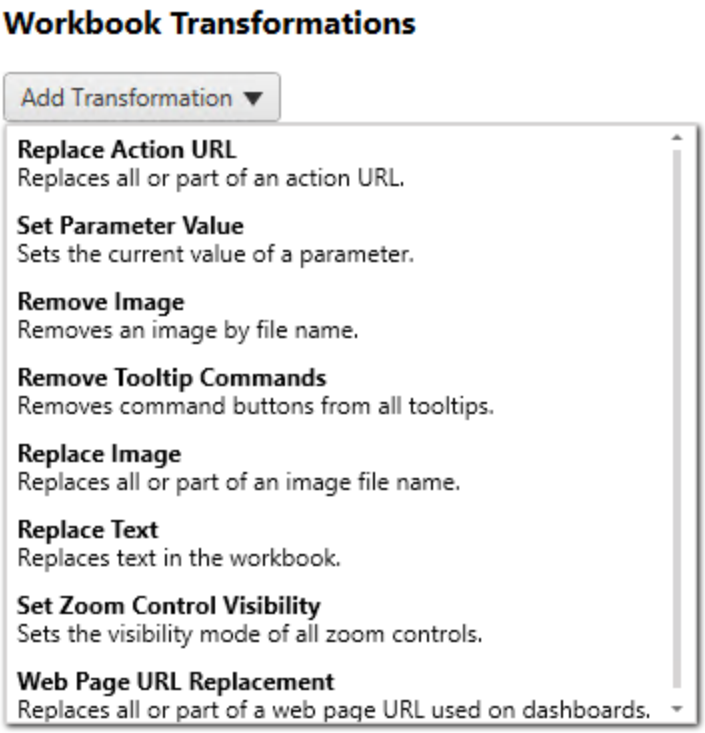
プレフィックスのマッピングと同様に、1 つまたはすべてのソースプロジェクトからワークブックのサフィックスを削除または置換できます。

ステップ 3: ワークブックの変換

変換ステップを使用してワークブックを変更できます。

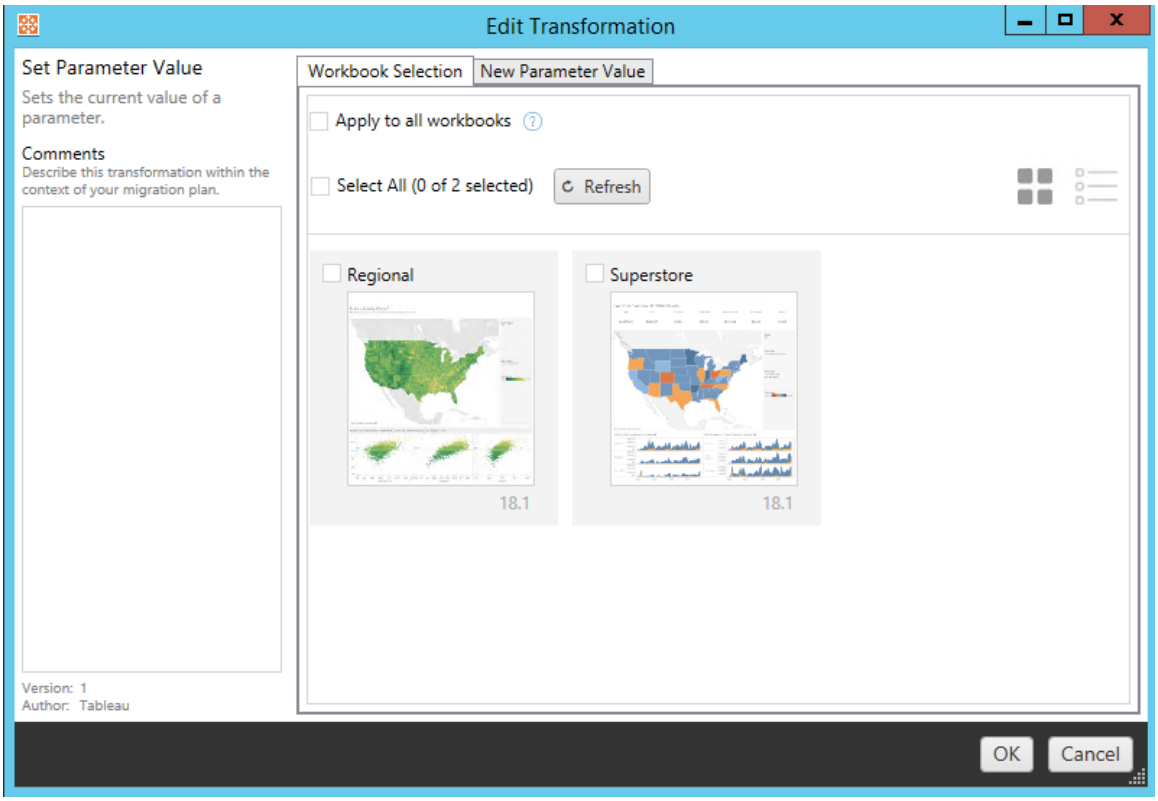


変換により指定された方法でワークブックが変更されます。追加の変換をプラグインを使用して組み込むことができます。また、今後のアプリケーション バージョンにも追加される予定です。**[Add Transformation (変換の追加)]** ドロップダウン メニューをクリックして、現在選択可能な変換を確認します。

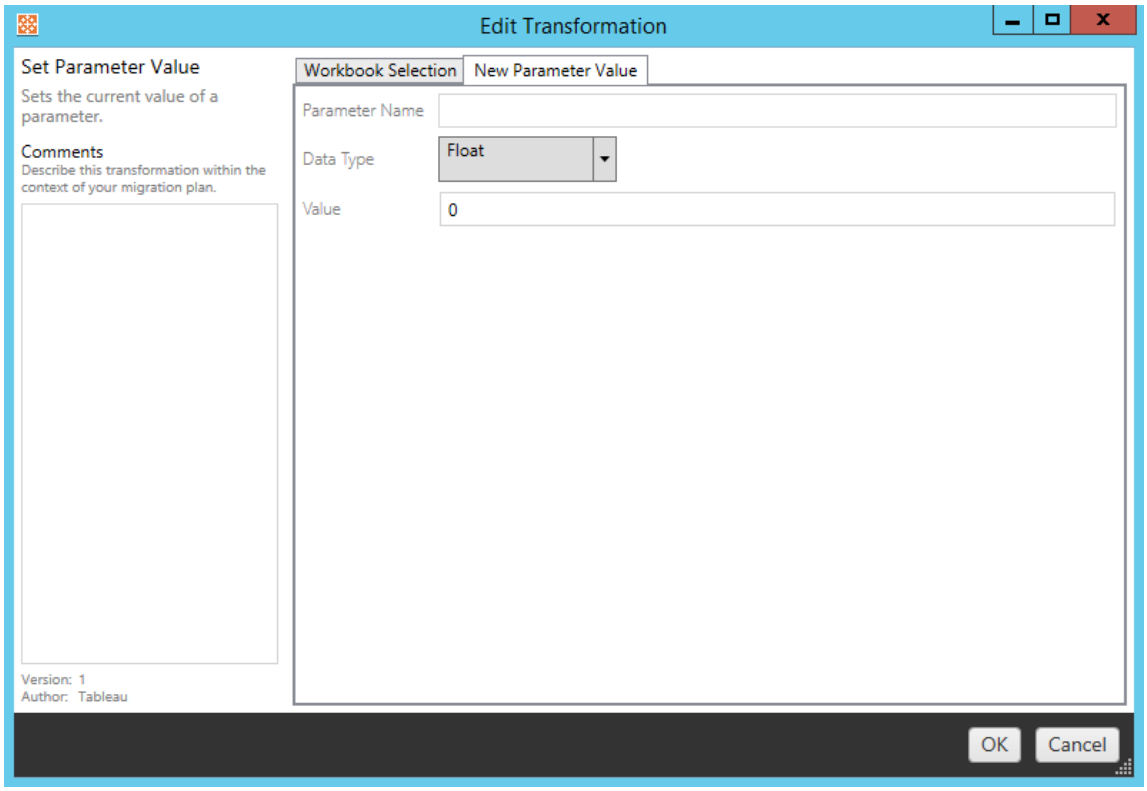


いずれかの変換を選択すると[Edit Transformation (変換の編集)] ウィンドウが表示され、選択したワークブックにカスタマイズできるようになります。すべての変換は、上から下に記載された順序で完了します。

さまざまなタイプの変換がありますが、基本的なステップは 2 つです。まず、変換の選択を行います。ここでは、変換するワークブックを選択します。選択領域は、計画段階の「ワークブックの選択」セクションと似ていて、[Basic selection (基本的な選択)] ラジオ ボタンのすべての機能を使用します (**[Select/Unselect All (すべて選択/選択解除)]**、**[更新]**、**[サムネイル]** 表示、**[リスト]** 表示)。リストの上部で、**[Select All]** を選択できます。これは、今後の変換ですべてのワークブックを自動的に選択するオプションです。また、ワークブックの表示ウィンドウを**更新**して、ソース サイトへの変更や更新を反映させることもできます。



2 番目のステップとして、[オプション] タブを使用して、選択したすべての変換に対し特定の選択内容を入力します。



[オプション] タブで各ワークブックの変換に異なる値を入力すると、編集しようとしている変換に応じ、タブに異なる名前が付けられます。

アクション URL の置換

この変換を使用して、ワークブック内の一部またはすべての URL アクションを置換します。[オプション] タブで、一致させる必要のあるテキストとその置換値を入力します。置換機能は URL アクションを対象としており、ワークブック全体の XML はチェックしません。

例:

URL: `www.exampledev.com`

一致: `dev`

置換: `Prod`

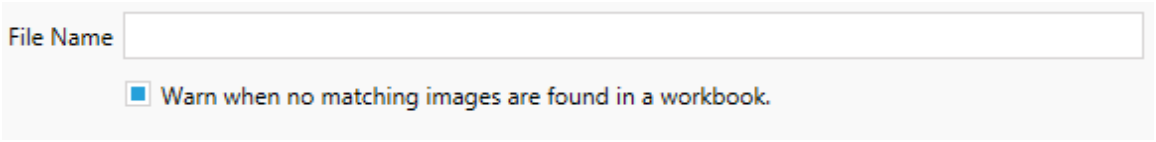
結果: `www.exampleProd.com`

パラメーター値の設定

新しいパラメーターを定義します。[オプション] タブで、パラメーターの名前、データ型 (ドロップダウンメニューから選択)、および値を入力します。

イメージの削除

[オプション] タブでファイル名を入力して、選択したワークブックのイメージ (透かしなど) を削除します。イメージが見つからない場合に移行時に警告を受信する追加のチェックボックスもあります。



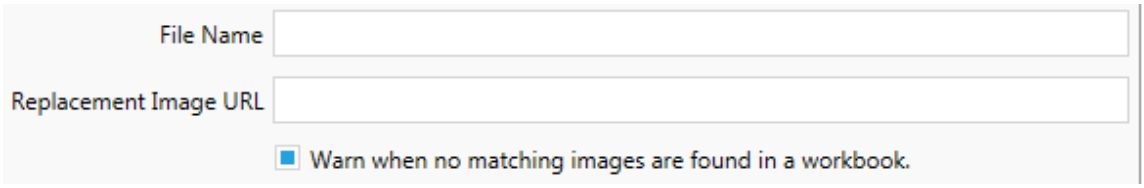
ツールヒントコマンドの削除

選択したワークブックからすべてのツールヒントコマンドを削除します。この変換で定義する追加のオプションはありません。

イメージの置換

選択したワークブックに埋め込まれているイメージを置換します。[オプション] タブで、現在のイメージと置換するイメージのファイル名を入力します。ローカル ファイル パスまたは URL を使用して、イメー

ジを置き換えることができます。



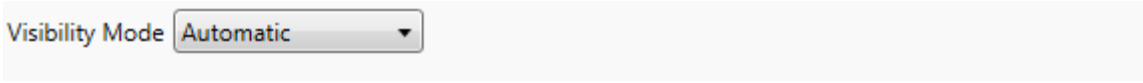
例:

ファイル名: `image.png`

置換するイメージの URL: `https://www.exampledev.com/replacementImage.png`

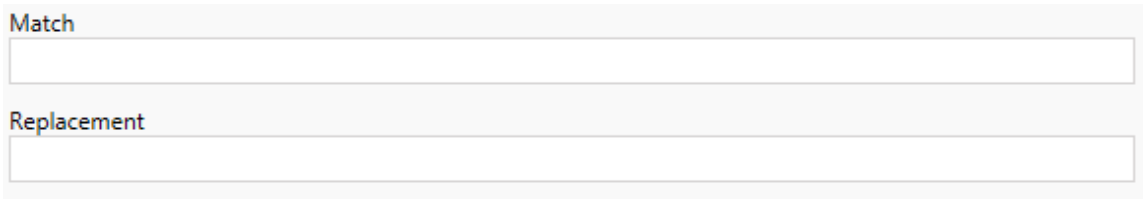
ズーム コントロールの表示

ドロップダウン メニューから表示モードを設定します ([オプション] タブの **[自動]**、**[ポイント時に表示]**、[非表示])。



Web ページ URL の置換

この変換を使用して、ダッシュボードで使用される Web ページ URL の一部または全部を置き換えます。[オプション] タブで、一致させる必要のあるテキストとその置換値を入力します。



例:

URL: `www.exampledev.com`

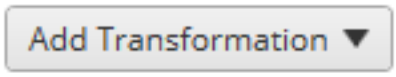
一致: `dev`

置換: `Prod`

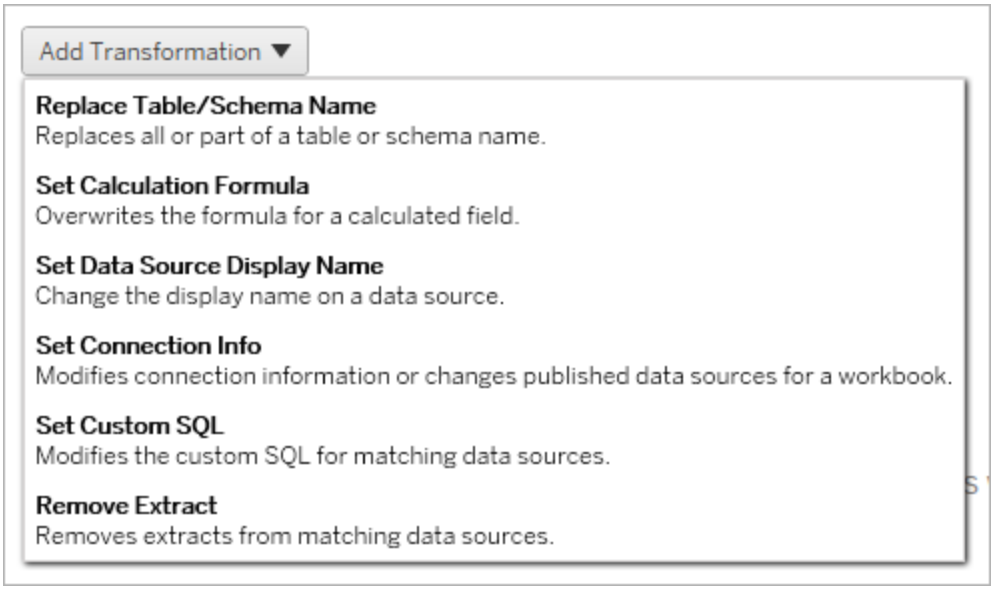
結果: `www.exampleProd.com`

ステップ 4: データ ソースの変換

企業環境への移行でワークブックを計画する次のステップは、データ ソースの変換です。この機能は、ワークブックの変換ステップに似ています。これらはワークブック内にパッケージ化されたデータ ソース向けです。パブリッシュされたデータ ソースは、プロセス内の別のステップで処理されます。

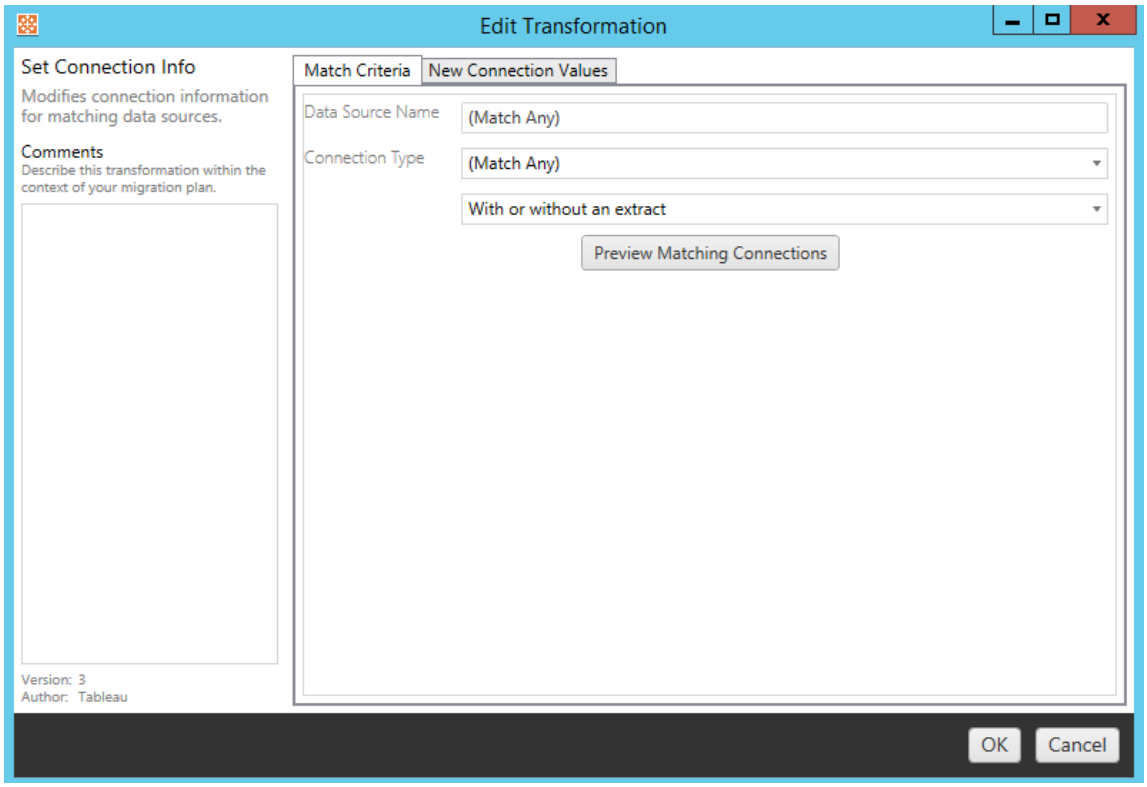


[Add Transformation (変換の追加)] ドロップダウン メニューをクリックすると、次のオプションが表示されます。

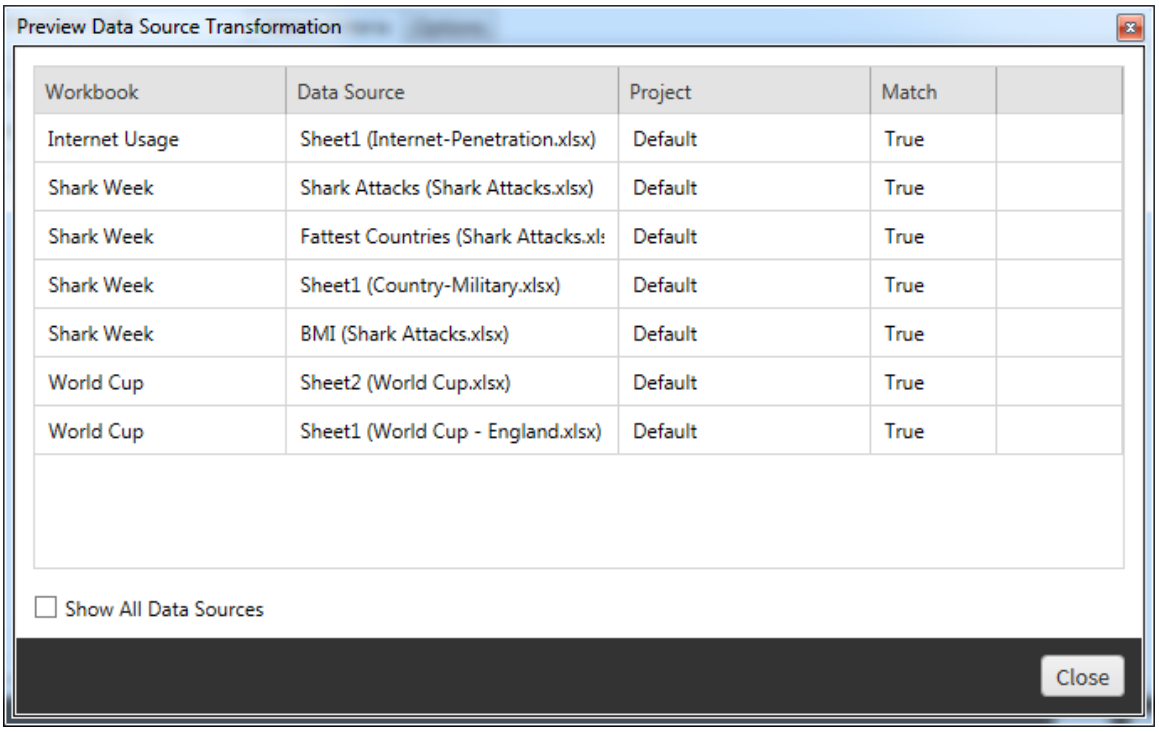


いずれかのデータ ソースの変換を選択すると[Edit Transformation (変換の編集)] ウィンドウが表示され、選択したデータ ソースにカスタマイズできるようになります。すべての変換は、上から下に記載された順序で完了します。

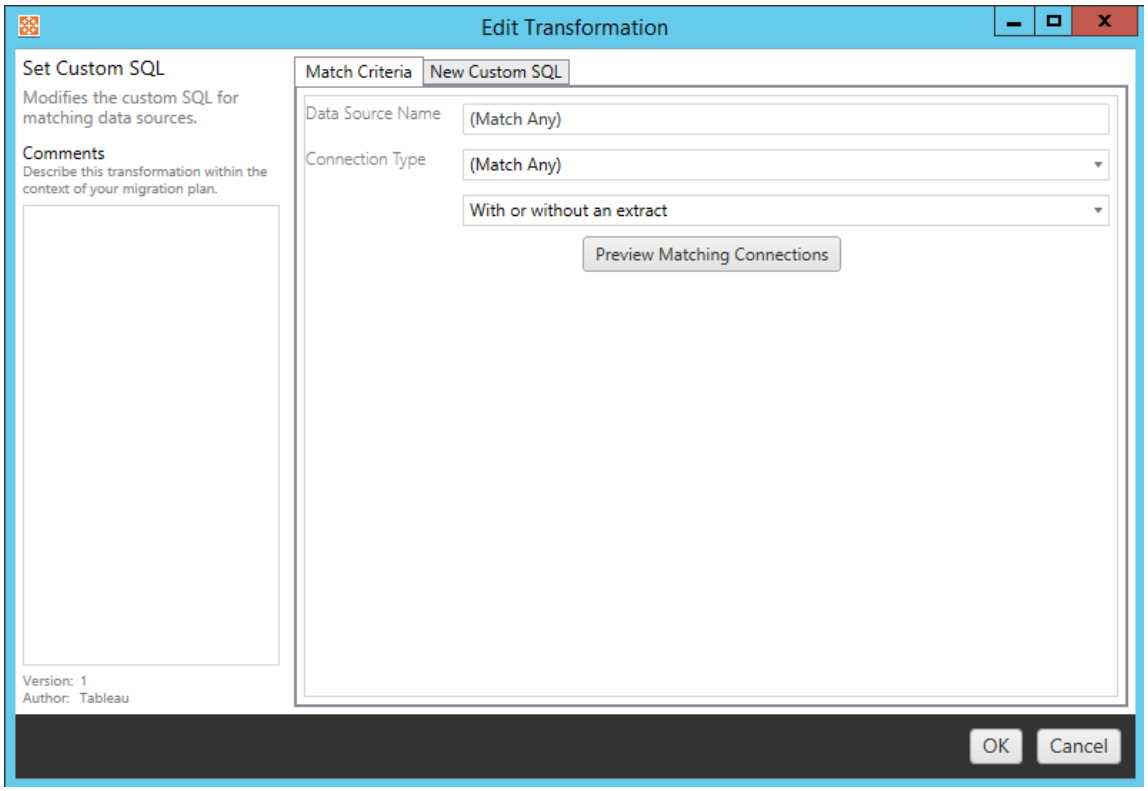
さまざまなタイプのデータソース変換がありますが、基本的なステップは 2 つです。まず、目的のデータソースで [Match Criteria (一致条件)] に条件を入力します。**[Match Criteria (一致条件)]** タブには、選択する接続タイプに応じて、表示されるフィールドが追加されます。



[Preview Source Connections (ソース接続のプレビュー)] をクリックして、入力した条件に一致する接続を見つけます。



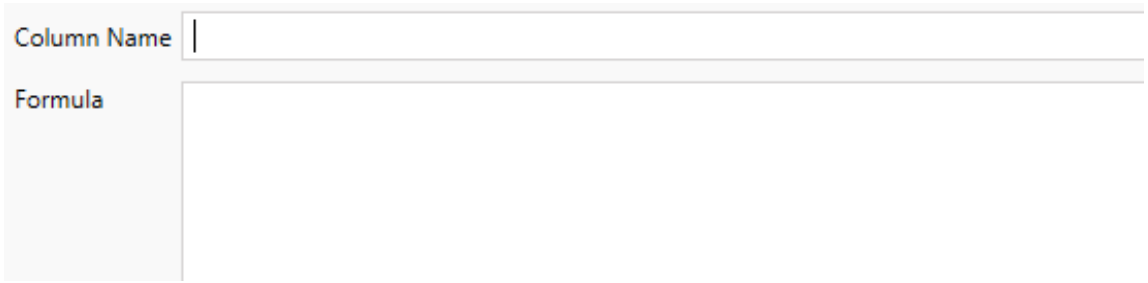
2 番目のステップとして、[オプション] タブを使用して、選択したすべての変換に対し特定の選択内容を入力します。



[オプション] タブで、各データソースの変換に異なる値を入力します。

Set Calculation Formula (計算式の設定)

[オプション] タブで、列の計算を置き換えることができます。



Set Connection Info (接続情報の設定)

[New Connection Values (新しい接続値)] タブで、新しいデータソースの認証方法と接続の詳細を入力します。選択した接続タイプに応じて、追加のフィールドが表示されます。

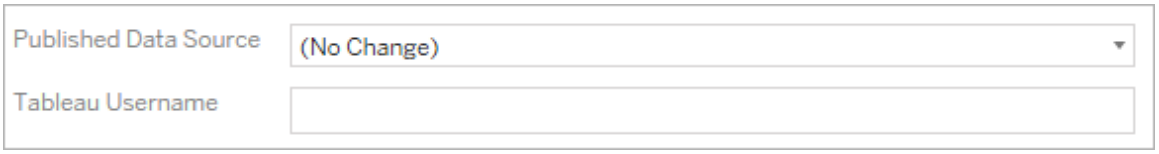
CMT を使用してパブリッシュされたデータ ソースを変更する

バージョン 2022.1 以降、ワークブックのパブリッシュされたデータ ソースを変更するには、**Tableau Server (パブリッシュされたデータ ソース)** の接続タイプを選択します。これにより、ワークブックを Tableau 環境間で移行するときに必要となる手動のステップ (コンテンツを開発環境から実稼働環境に昇格させるなど) を減らすことができます。

パブリッシュされたデータ ソースを変更するには、ドロップダウン メニューからデータ ソースを選択し、認証に使用する **Tableau ユーザー名** を入力します。ユーザーは、展開先サイトに存在し、パブリッシュされたデータ ソースの接続機能を持っている必要があります。

- ファイルベースのデータ ソースの場合、ユーザーはワークブックにアクセスし、指定された Tableau ユーザーのパーミッションに基づいてデータを表示します。
- 他のすべてのデータ ソースの場合、ビューまたはワークブックが読み込まれると、ユーザーは独自のデータベースの認証資格情報を入力するよう求められます。

Tableau ユーザー名が指定されていない場合、接続機能を持つユーザーのみがワークブックのデータを表示できます。



Set Custom SQL (カスタム SQL の設定)

[新しいカスタム SQL] タブで、変更するカスタム SQL クエリの名前を **[クエリ名の一致]** に入力します。クエリ名は、データ ソースの物理レイヤーのカスタム SQL クエリ名と一致する必要があります。これらの名前が一致しない場合、変換は失敗します。データ モデリングと物理レイヤーの詳細については、Tableau データ モデルを参照してください。

クエリ名を入力したら、目的の **カスタム SQL** をテキスト フィールドに入力します。カスタム SQL を不適切に使用した場合、ワークブックのパフォーマンスに悪影響を及ぼす可能性があることに注意してください。

Remove Extract (抽出の削除)

この変換に [オプション] タブはありません。**[Match Criteria]** に情報を入力するだけで、移行時に抽出が削除されます。

さらに、各変換について、**[Edit Transformation]** ウィンドウの左側にある **[Comments]** セクションにメモを入力できます。

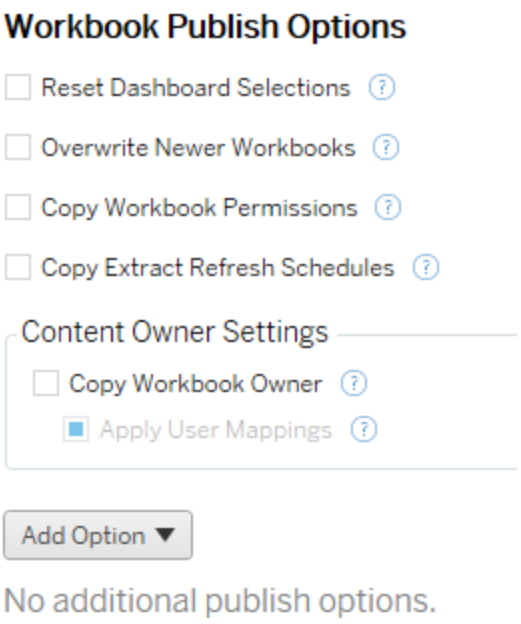
保存済み認証資格情報の適用

バージョン 2022.3 で非推奨になりました。代わりに、Set Connection Info (接続情報の設定) データソース変換を使用してください。

[オプション] タブで、データ接続に使用する **Tableau ユーザー名** と、それに対応する **保存済み認証資格情報ユーザー名** を入力します。保存した認証資格情報は、Tableau サイトの [アカウント設定] ページで既存のデータ接続にのみ適用できます。詳細については、データ接続のために保存された認証資格情報の管理を参照してください。

ステップ 5: パブリッシュオプション

ワークブック段階の最終ステップでは、パブリッシュオプションを選択して、タグの変換、抽出の更新スケジュール、およびパーミッションを作成します。



Reset Dashboard Selections (ダッシュボードの選択をリセットする)

このオプションを選択すると、ダッシュボード上のすべてのオブジェクトが選択解除されます。

Overwrite Newer Workbooks (新しい方のワークブックを上書きする)

このチェックボックスをオンにすると、ワークブックの移動と同時に作成されたか、それよりあとに作成されたワークブックを上書きすることになってもワークブックを移行します。

Copy Workbook Permissions (ワークブックのパーミッションをコピーする)

このオプションを選択すると、移行ツールは、展開元ワークブックのパーミッションにできる限り一致させようと試みます。

抽出更新スケジュールのコピー

このオプションを選択すると、移行ツールは、展開先ワークブックの抽出更新スケジュールを展開元ワークブックの名前と一致するスケジュールに設定しようと試みます。

注: 抽出更新スケジュールは、Tableau Cloud で作成することはできません。展開先が Tableau Cloud サイトの場合、このオプションは使用できません。詳細については、移行に関する制限事項を参照してください。

ワークブックの認証資格情報をコピーする

ワークブックに埋め込まれたデータソースの埋め込みの認証資格情報をコピーします。Tableau Server から Tableau Cloud サイトへの移行時にのみ使用できます。詳細については、「認証資格情報が埋め込まれたワークブックとデータソースの移行」を参照してください。

注: CMT は、OAuth 接続での埋め込みの認証資格情報の移行をサポートしていません。OAuth 認証資格情報を移行先サイトに移行するには、Set Connection Info (接続情報の設定) データソース変換を使用します。

Copy Workbook Owners (ワークブックの所有者をコピーする)

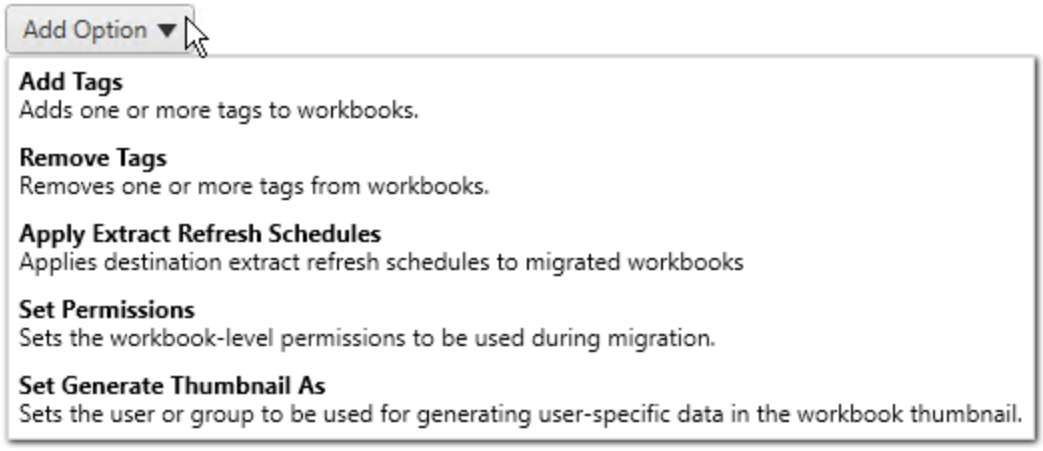
ワークブックの所有者設定をソースの場所からコピーして、ワークブックの所有者を割り当てます。選択されていない場合、Content Migration Tool のユーザーには、展開先の場所にあるワークブックの所有権が与えられます。

ユーザー マッピングの適用

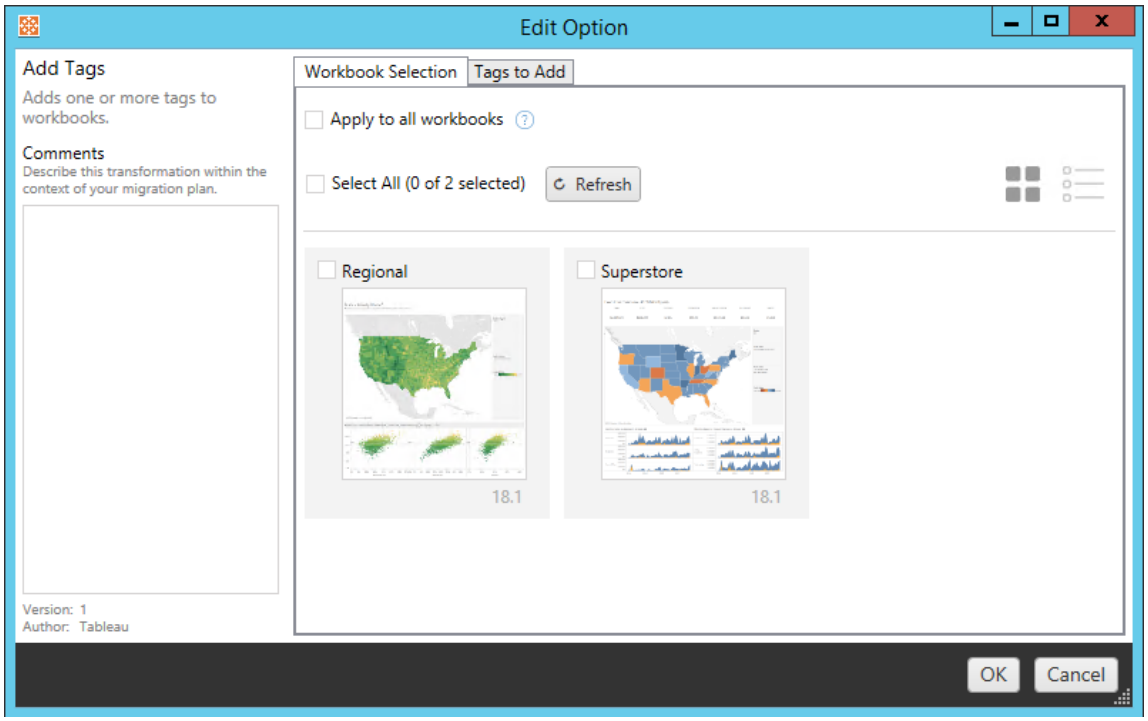
ユーザー マッピングを適用して、コンテンツの所有権を割り当てます。展開先の場所でユーザー名の構文に違いがある場合は、このオプションを選択します。詳細については、移行計画: パーミッションと所有権を参照してください。

オプションの追加

[Add Option] ドロップダウン メニューをクリックして、追加できる変換のタイプを選択します。



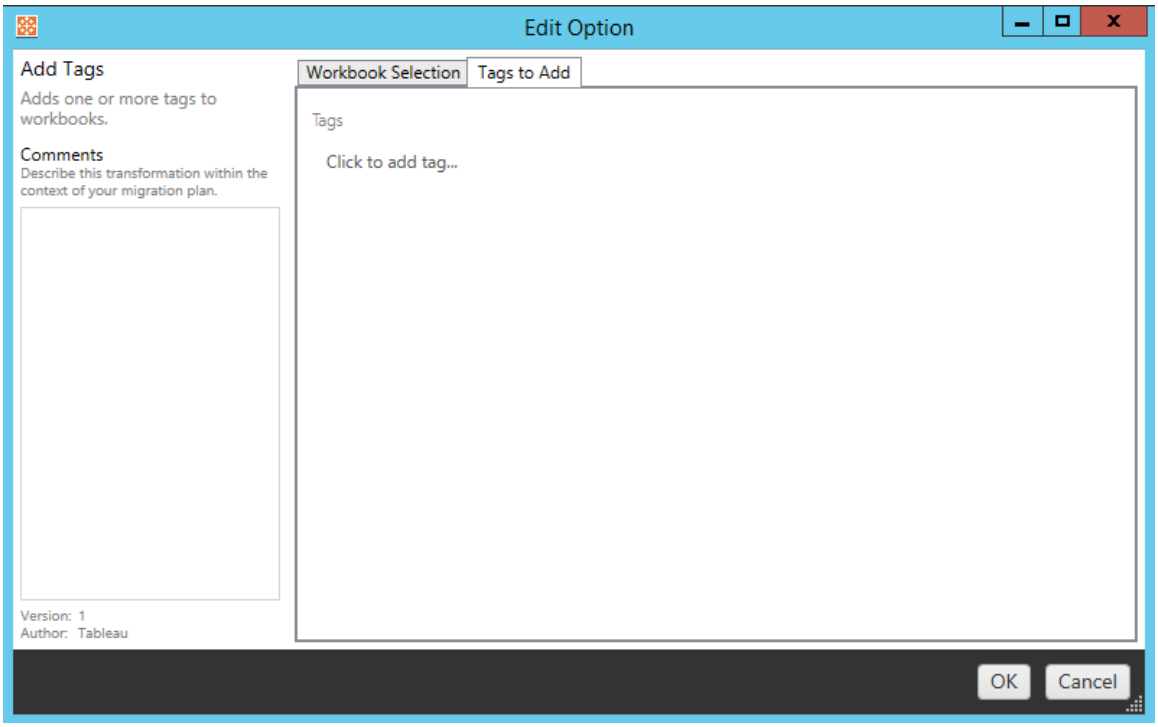
さまざまなタイプの変換がありますが、基本的なステップは 2 つです。まず、変換の選択を行います。ここでは、変換するワークブックを選択します。選択領域は、計画段階の「ワークブックの選択」セクションと似ていて、[基本] セクションのラジオ ボタンの機能をすべて使用します ([**Select/Unselect All**]、[**Refresh**]、[**Thumbnail Display**]、[**List Display**])。リストの上部で、[**Select All**] を選択できます。これは、今後の変換ですべてのワークブックを自動的に選択するオプションです。また、ワークブックの表示ウィンドウを**更新**して、ソース サイトへの変更や更新を反映させることもできます。



2 番目のステップとして、[オプション] タブを使用して、選択したすべての変換に対し特定の選択内容を入力します。**注:** [オプション] タブには、編集しようとしている変換に応じ、異なる名前が付けられます。

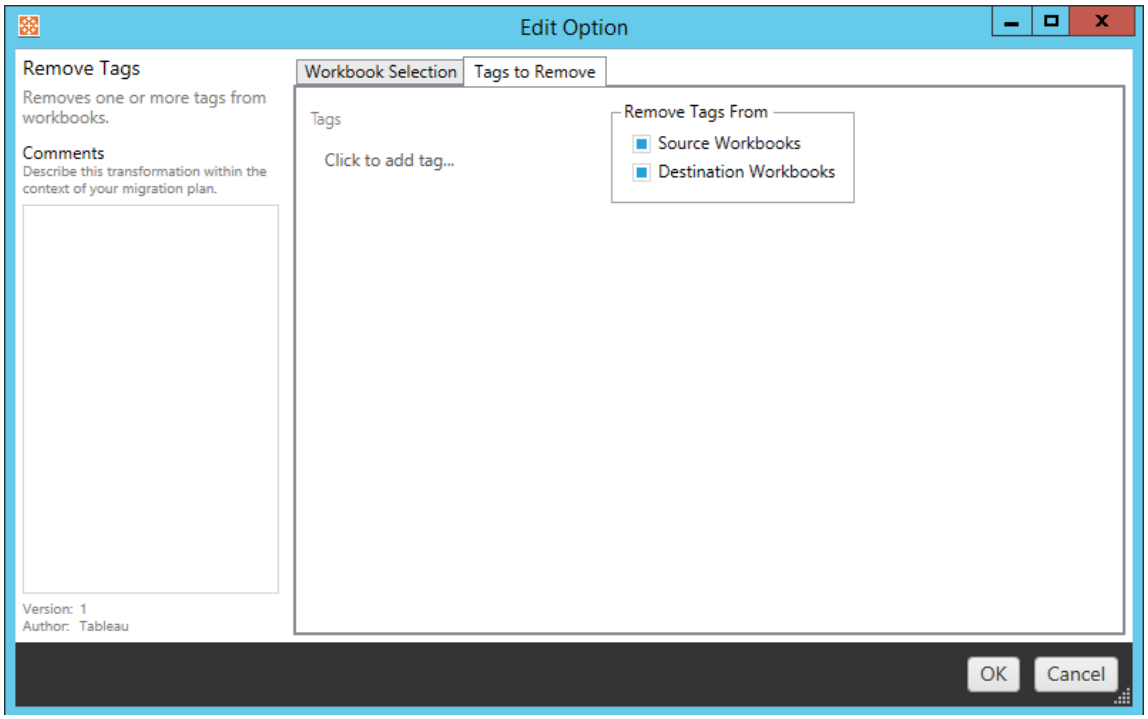
Add Tags (タグの追加)

これにより、1 つ以上のタグをワークブックに追加できます。以前に入力したタグにカーソルを合わせると、青字で「X」と表示され、削除できるようになります。



Remove Tags (タグの削除)

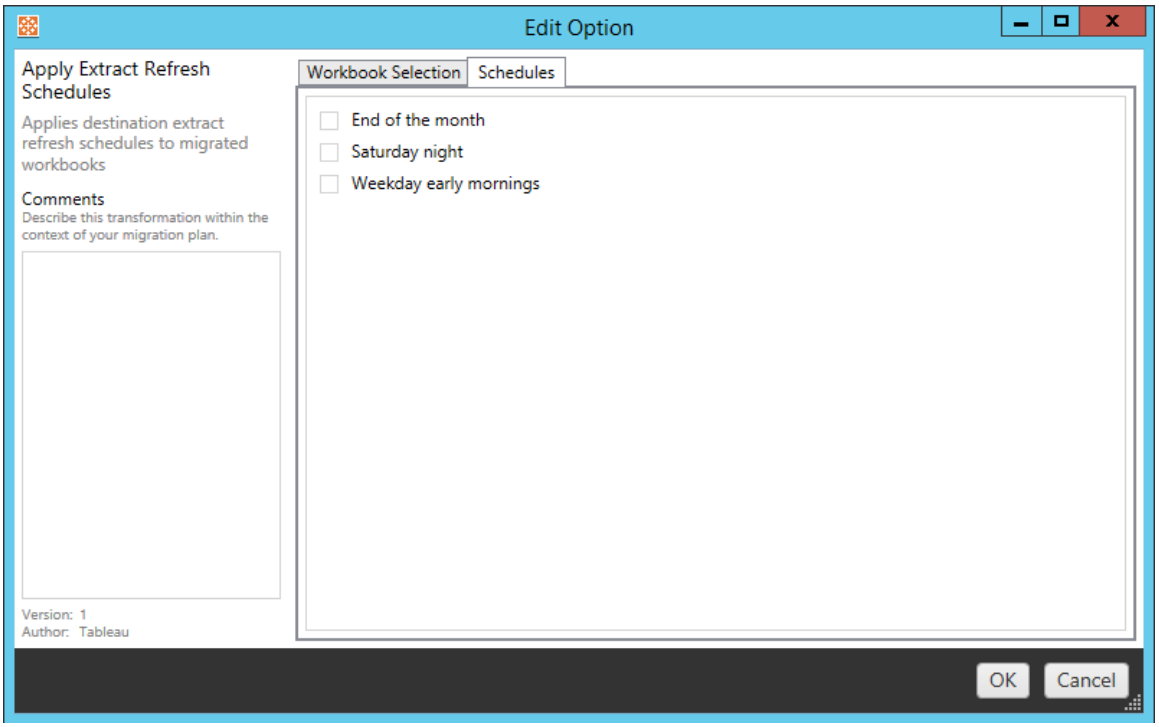
これにより、1 つ以上のタグをワークブックに追加できます。以前に入力したタグにカーソルを合わせると、青字で「X」と表示され、削除できるようになります。展開元または展開先のワークブックからタグを削除することもできます。



Apply Extract Refresh Schedules (抽出更新スケジュールの適用)

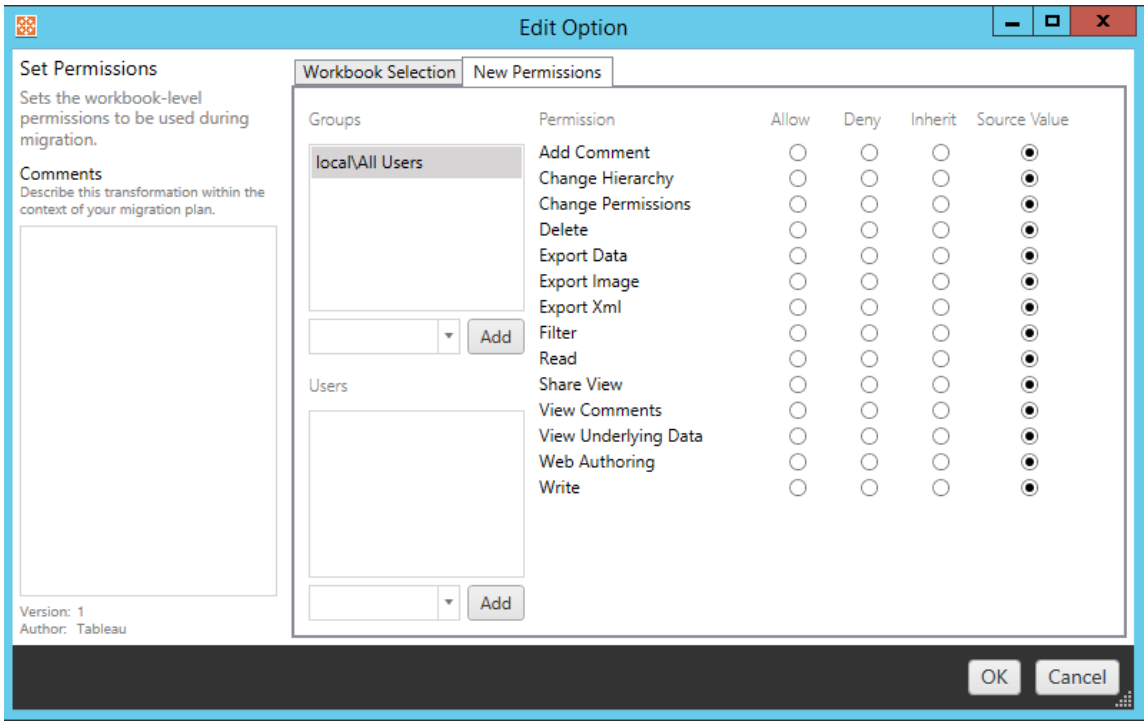
ここでは、展開先の抽出更新スケジュールを、移行したワークブックに適用できます。生成済みスケジュールのリストは、展開先から取得されます。

注: 抽出更新スケジュールは、Tableau Cloud で作成することはできません。展開先が Tableau Cloud サイトの場合、このオプションは使用できません。詳細については、移行に関する制限事項を参照してください。



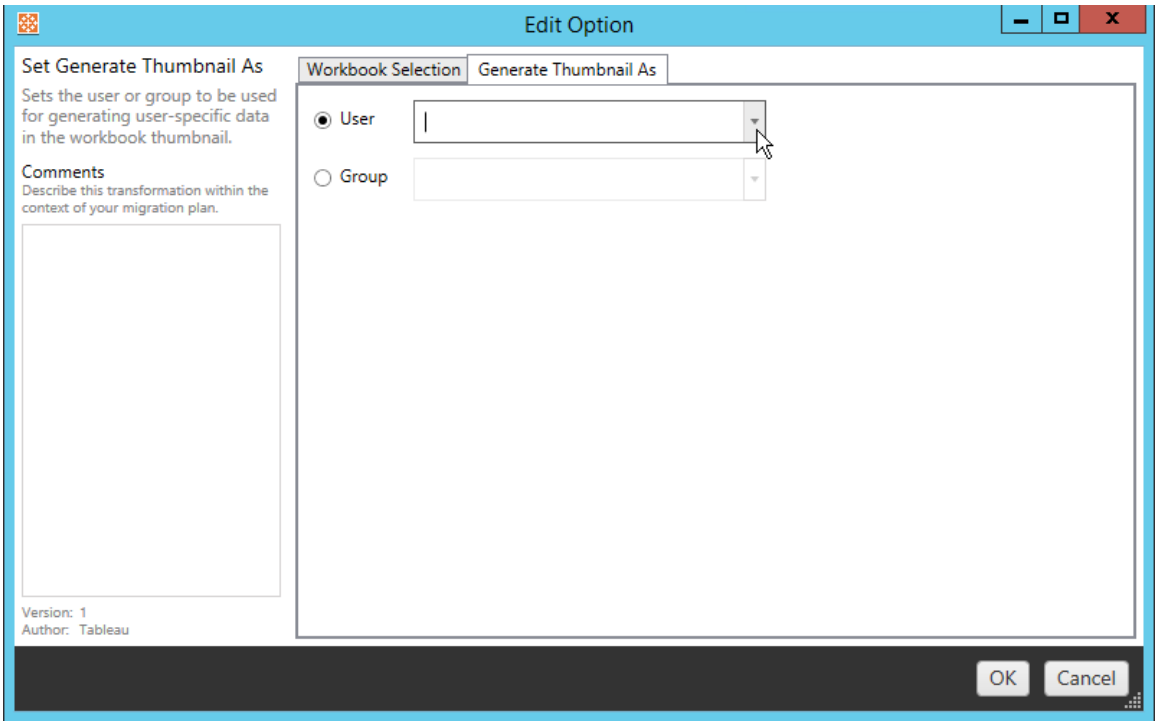
パーミッションの設定

この変換によって、選択したワークブックのパーミッションを編集します。[グループ] または [ユーザー] 内に値を入力し、**[Add]** をクリックします。必要に応じてパーミッションを調整してください。パーミッションの許可 (**[Allow]**)、パーミッションの拒否 (**[Deny]**)、継承 (**[Inherit]**)、ソース値の保持 (**[Source Value]**) の 4 種類のオプションがあります。



Set Generate Thumbnail As (名前を付けてサムネイルの生成を設定)

これにより、移行後にワークブックのサムネイルでユーザー固有のデータを生成するために使用される **[ユーザー]** または **[グループ]** を設定できるようになります。各オプションに、目的のユーザーまたはグループを選択するドロップダウンがあります。



ステップ 6: 次のステップに進む

ワークブックと環境設定を選択した後、**[Next]** をクリックして、計画フェーズの [移行計画: パブリッシュ済みデータソース] セクションに進みます。

実行可能なユーザー

Explorer 以上の役割を持つ Tableau サイトユーザー。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

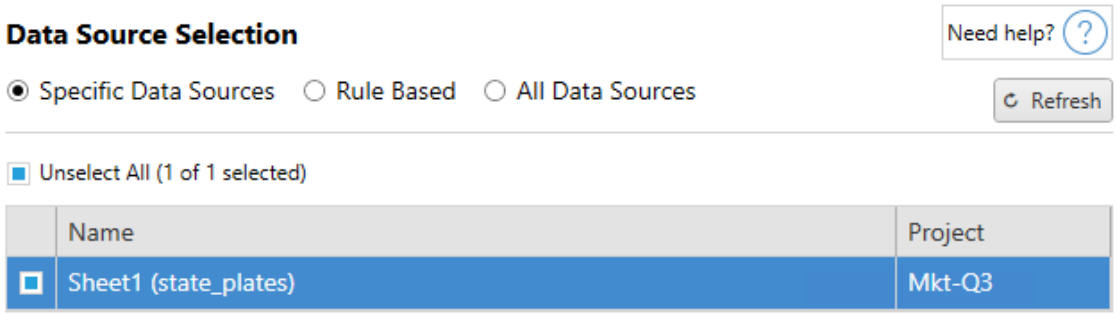
移行計画: パブリッシュ済みデータソース

Tableau Content Migration Tool での移行計画作成の次のステップでは、パブリッシュされたデータソースを選択し、マッピングを実行した後、変換を追加します。このプロセスはワークブックの計画段階のステップによく似ています (特にデータソースのマッピング ステップ)。

注: ワークブックまたはデータソースに抽出が含まれている場合は、抽出を使用するワークブックとデータソースの移行に記載された情報を読んで理解しておいてください。

ステップ 1: 選択

移行計画のパブリッシュ済みデータソースの段階を開始する際に、移行計画に含めるデータソースを選択します。

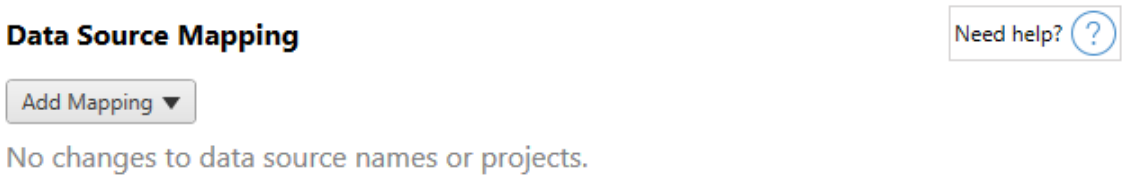


データソースの選択を行うのは、移行の時点でのみです。2 種類の選択方法があります。**Specific Data Sources** を使用して、1 つまたは複数のパブリッシュされたデータソースを選択します。**[Refresh]** をクリックして、パブリッシュされたデータソースのリストを再度読み込みます。

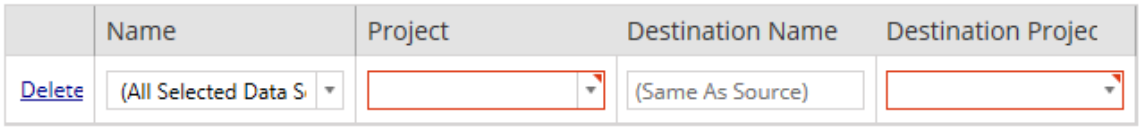
2 つ目のオプションは **All Data Sources** を使用することであり、展開元サイト内のすべてのデータソースを選択します。

ステップ 2: マッピング

次に、展開元データソースを新しい展開先にマッピングします。これはワークブックのマッピングと似ています。



ここで変更を行わないと、選択したデータソースが単にソースと同じ名前およびプロジェクトで展開されます。データソースのマッピングを追加するには、**[Add Mapping]** をクリックします。マッピング領域に次のオプションが表示されます。



エントリには次のオプションがあります。

削除

[Delete] リンクをクリックすると、このマッピング エントリが削除されます。

名前

[Name] メニューでは、マッピングするデータソースを選択します。**[(All Selected Data Sources)]** を選択すると、すべてのデータソースを選択できます。

プロジェクト

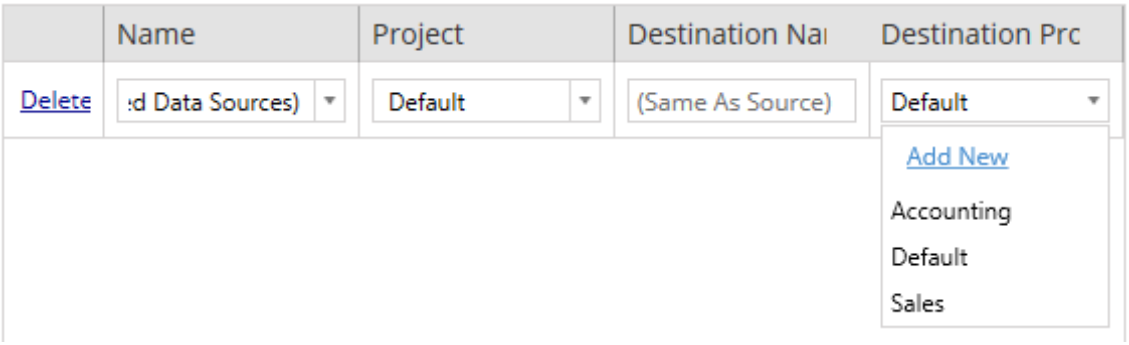
[Project] は、関連付けられたデータソース名のプロジェクトです。

展開先名

既定では、Content Migration Tool は同じ **[Destination Name]** (**[(Same As Source)]**) を使用し、ソース ファイルにある元の名前を保持しますが、展開先フォルダーの新しい名前を入力することもできます。

展開先プロジェクト

展開先プロジェクトがサイト上にすでに作成されている場合は、移行したワークブックを配置するプロジェクトを選択するか、**[Add New]** をクリックして新しいプロジェクトを作成することができます。個別のデータソースに対して、異なる展開先プロジェクトを作成できます。

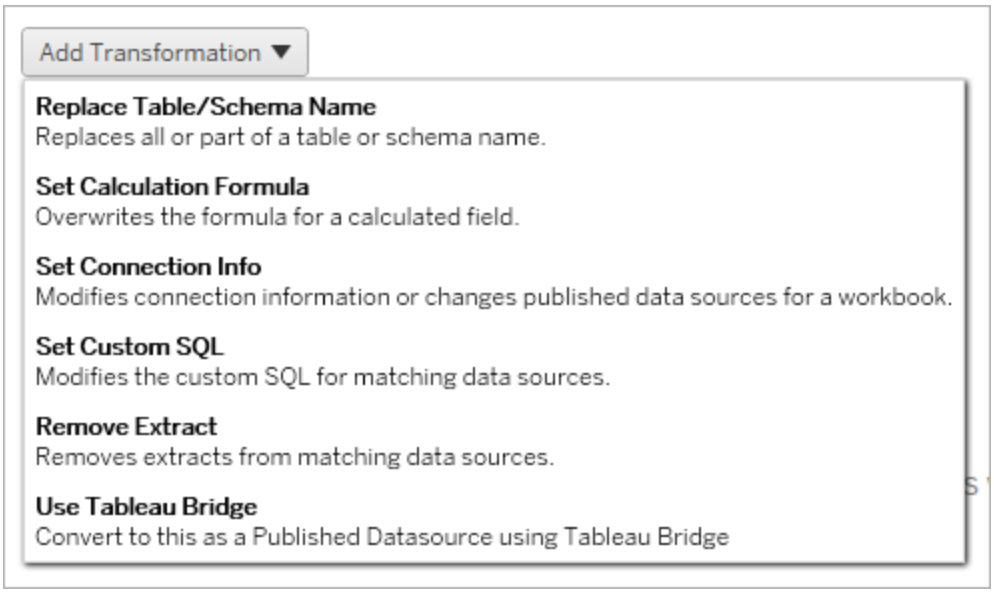


ほとんどの場合、1 つのデータソースに対して複数のマッピング エントリが存在する場合は、検証エラーが表示され、続行するには修正の必要があります。ただし、重要な例外として、1 つのデータソースが特定の選択範囲とプロジェクト全体のマッピング エントリの両方に一致することは可能です。この場合、より具体的なエントリが使用されます。

必要なデータソース マッピングがすべて完成したら、**[Next]** をクリックして続行します。

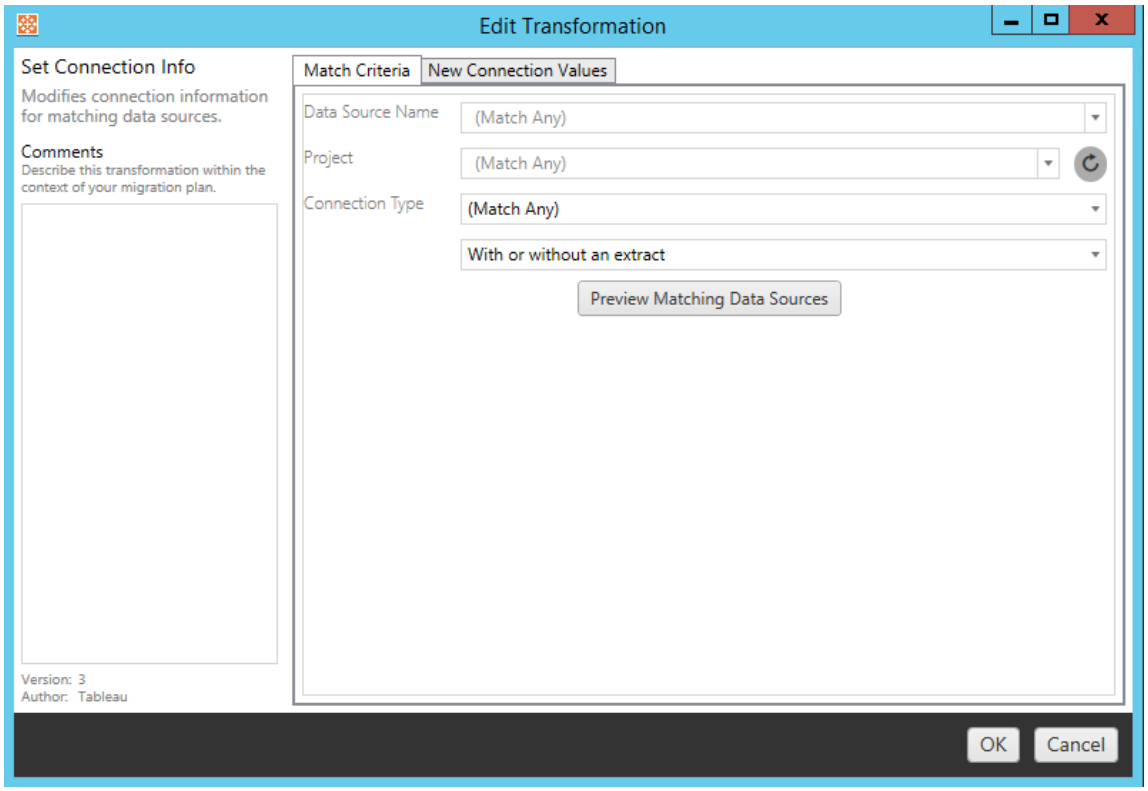
ステップ 3: データソースの変換

変換オプションを使用して、データソースを変更できます。**Add Transformation** をクリックすると、使用可能な変換の一覧が表示されます。



いずれかのデータソース変換を選択すると、**[Edit Transformation]** ウィンドウが開きます。これは選択したデータソースをカスタマイズするときに使用します。変換は、上から下に記載された順序で実行されます。

ほとんどのデータソース変換には 2 つの基本的な手順があります。まず、目的のデータソースで **[Match Criteria]** に条件を入力します。選択した接続タイプによっては、**[Match Criteria]** タブに追加のフィールドが表示されます。



2 つ目の手順は、追加する変換タイプによって異なります。2 つ目のタブで、各データソースの変換に異なる値を入力します。

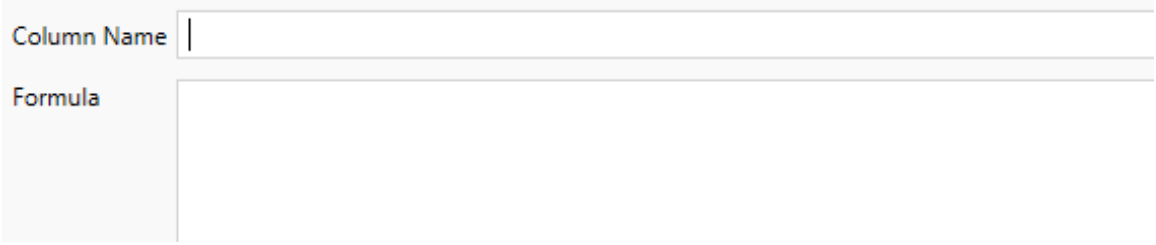
追加する各変換の **[Edit Transformation]** ウィンドウの左側にある **[Comments]** セクションに注記を追加できます。

表/スキーマ名の置換

[Options] タブでは、表またはスキーマ名の全部または一部を置き換えることができます。

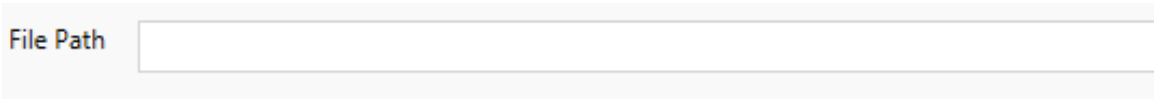
Set Calculation Formula (計算式の設定)

[New Calculation Formula] タブで、列の計算式を置き換えることができます。



Set Connection Info (接続情報の設定)

[New Connection Values (新しい接続値)] タブで、新しいデータ ソースの認証方法と接続の詳細を入力します。選択した接続タイプに応じて、追加のフィールドが表示されます。



Set Custom SQL (カスタム SQL の設定)

[新しいカスタム SQL] タブで、変更するカスタム SQL クエリの名前を **[クエリ名の一致]** に入力します。クエリ名は、データ ソースの物理レイヤーのカスタム SQL クエリ名と一致する必要があります。これらの名前が一致しない場合、変換は失敗します。データ モデリングと物理レイヤーの詳細については、「Tableau データ モデル」を参照してください。

クエリ名を入力したら、目的の **カスタム SQL** をテキスト フィールドに入力します。カスタム SQL を不適切に使用した場合、ワークブックのパフォーマンスに悪影響を及ぼす可能性があることに注意してください。

Remove Extract (抽出の削除)

この変換に **[Options]** タブはありません。**[Match Criteria]** の情報を入力すると、移行時に抽出が削除されます。

Tableau Bridge の使用

この変換に **[Options]** タブはありません。**[Match Criteria]** の情報を入力すると、プライベートネットワーク内のデータソース (パブリックインターネットにはアクセスできません) を Tableau Bridge を使用して更新できます。

データソースを移行する前に、移行先サイトTableau Cloudで Tableau Bridge を構成する必要があります。Tableau Bridge の詳細については、Tableau Cloud ヘルプの「Tableau Bridge の使用」を参照してください。移行後、Tableau Cloud を通じてデータソースに更新スケジュールを割り当てる必要があります。

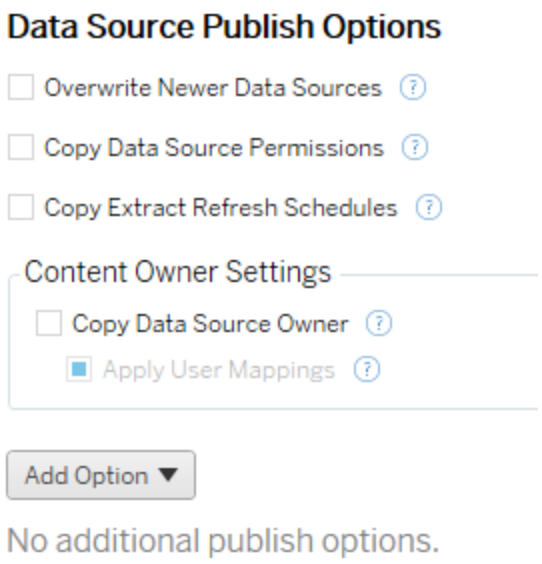
保存済み認証資格情報の適用

バージョン 2022.3 で非推奨になりました。代わりに、Set Connection Info (接続情報の設定) データソース変換を使用してください。

[オプション] タブで、データ接続に使用する **Tableau ユーザー名** と、それに対応する **保存済み認証資格情報ユーザー名** を入力します。保存した認証資格情報は、Tableau サイトの [アカウント設定] ページで既存のデータ接続にのみ適用できます。詳細については、データ接続のために保存された認証資格情報の管理を参照してください。

ステップ 4: パブリッシュオプション

パブリッシュされたデータソースの段階の最後のステップは、パーミッションとタグ用の変換を作成し、データソースに固有のパブリッシュオプションをファイナライズすることです。



より新しいデータソースの上書き

これを選択すると、より最近更新されたデータソースが上書きされた場合でもデータソースがパブリッシュされます。

データソース権限のコピー

これを選択すると、移行ツールは、展開元のパブリッシュされたデータソースの権限にできる限り一致させようと試みます。

抽出更新スケジュールのコピー

これを選択すると、移行ツールは、展開先データソースの抽出更新スケジュールを展開元の名前と一致するスケジュールに設定しようと試みます。

> **注:** 抽出更新スケジュールは、Tableau Cloud で作成することはできません。展開先が Tableau Cloud サイトの場合、このオプションは使用できません。詳細については、移行に関する制限事項を参照してください。

データソースの埋め込みの認証資格情報をコピーする

パブリッシュされたデータソースの埋め込みの認証資格情報をコピーします。Tableau Server から Tableau Cloud サイトへの移行時にのみ使用できます。詳細については、「認証資格情報が埋め込まれたワークブックとデータソースの移行」を参照してください。

注: CMT は、OAuth 接続での埋め込みの認証資格情報の移行をサポートしていません。OAuth 認証資格情報を移行先サイトに移行するには、Set Connection Info (接続情報の設定) データソース変換を使用します。

データソース所有者のコピー

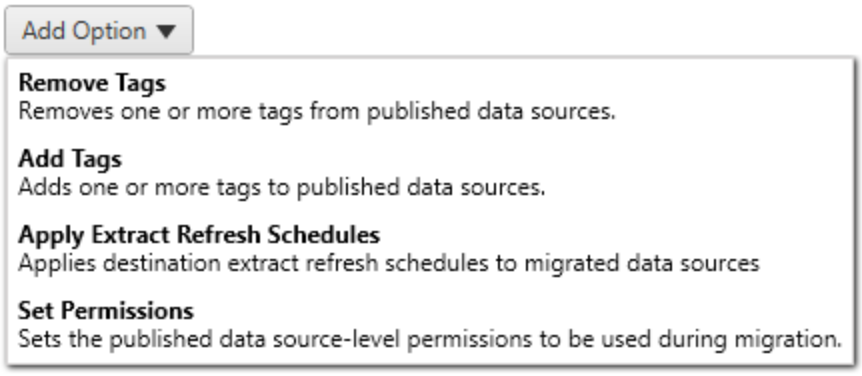
データソース所有者の設定を展開元の場所からコピーして、データソース所有者を割り当てます。選択されていない場合、Content Migration Tool ユーザーには、展開先の場所にあるデータソースの所有権が与えられます。

ユーザー マッピングの適用

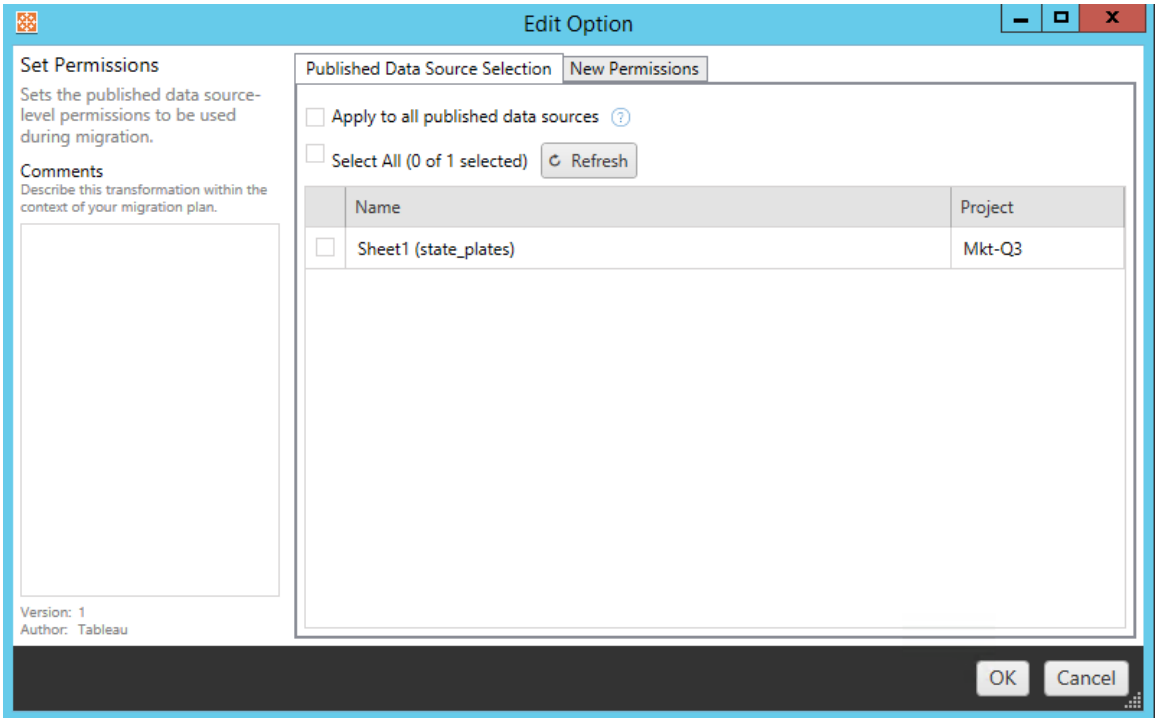
ユーザー マッピングを適用して、コンテンツの所有権を割り当てます。展開先の場所でユーザー名の構文に違いがある場合は、このオプションを選択します。詳細については、移行計画: パーミッションと所有権を参照してください。

オプションの追加

[Add Option] ドロップダウン メニューをクリックして、追加できる変換のタイプを選択します。



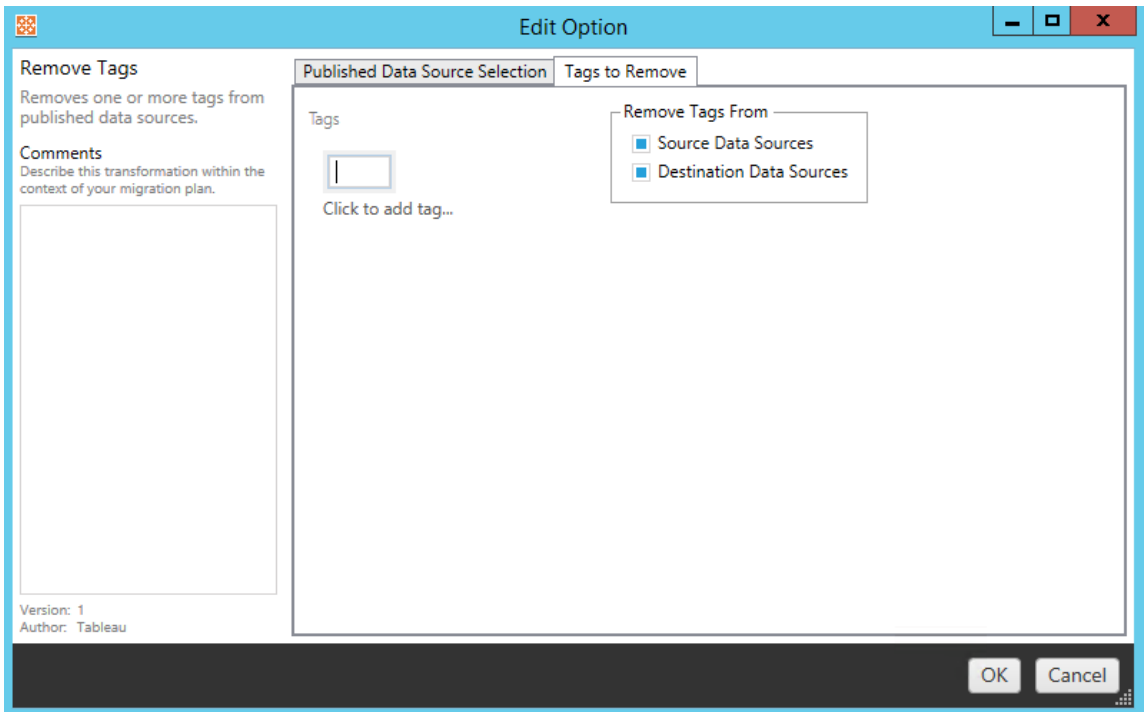
さまざまなタイプの変換がありますが、基本的なステップは 2 つです。まず、変換の選択を行います。この場合は、変換するデータソースを選択します。リストの上部で **[Apply to all published data sources]** オプションを選択すると、将来の変換ですべてのデータソースが自動的に選択されます。また、**[Refresh]** を選択して、展開元サイトの変更や更新を反映するようにデータソース表示ウィンドウを更新することもできます。



2 番目のステップは、選択した変換に固有の選択内容を入力することです。

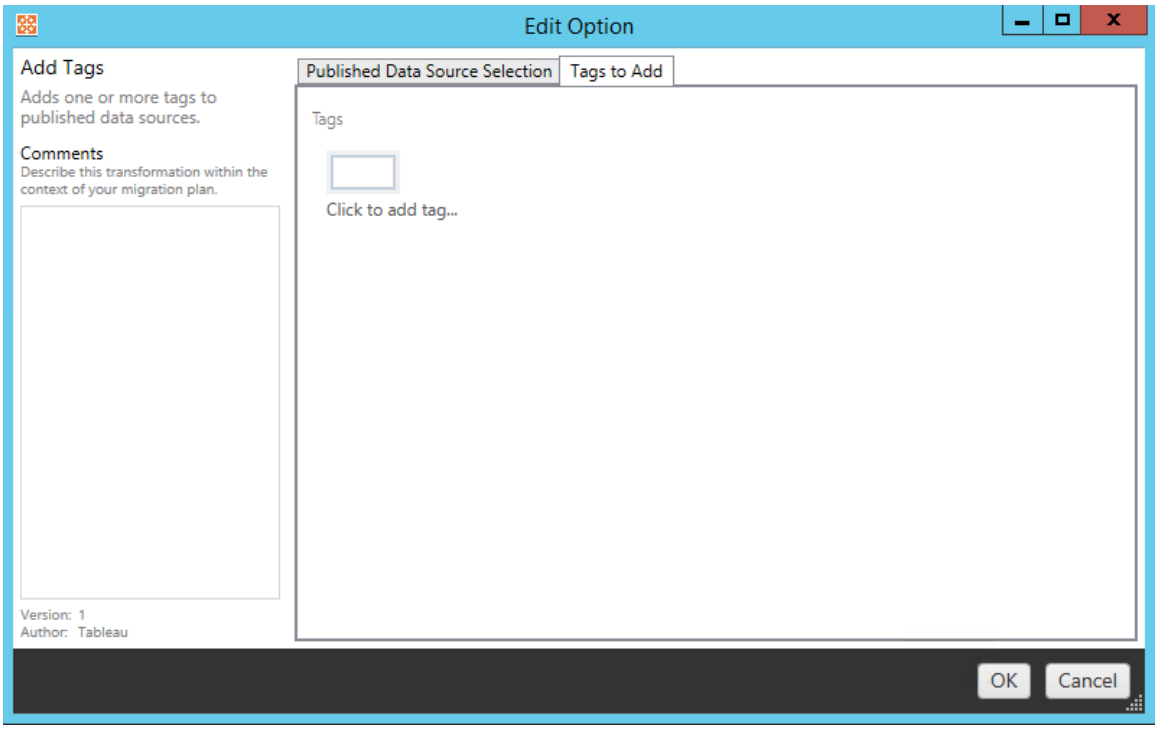
Remove Tags (タグの削除)

タグを削除するデータソースを選択したら、削除するすべてのタグを下部のフィールドに入力し、**[Add]** をクリックします。この画面では、展開元データソースと展開先データソースのどちらから削除するかを選択することもできます。以前に入力したタグを削除する場合は、そのタグをクリックして Delete キーを押します。



Add Tags (タグの追加)

目的のデータソースを選択したら、割り当てるタグを下部のフィールドに入力し、**[Add]** をクリックして追加します。タグを削除する場合は、そのタグをクリックして Delete キーを押します。

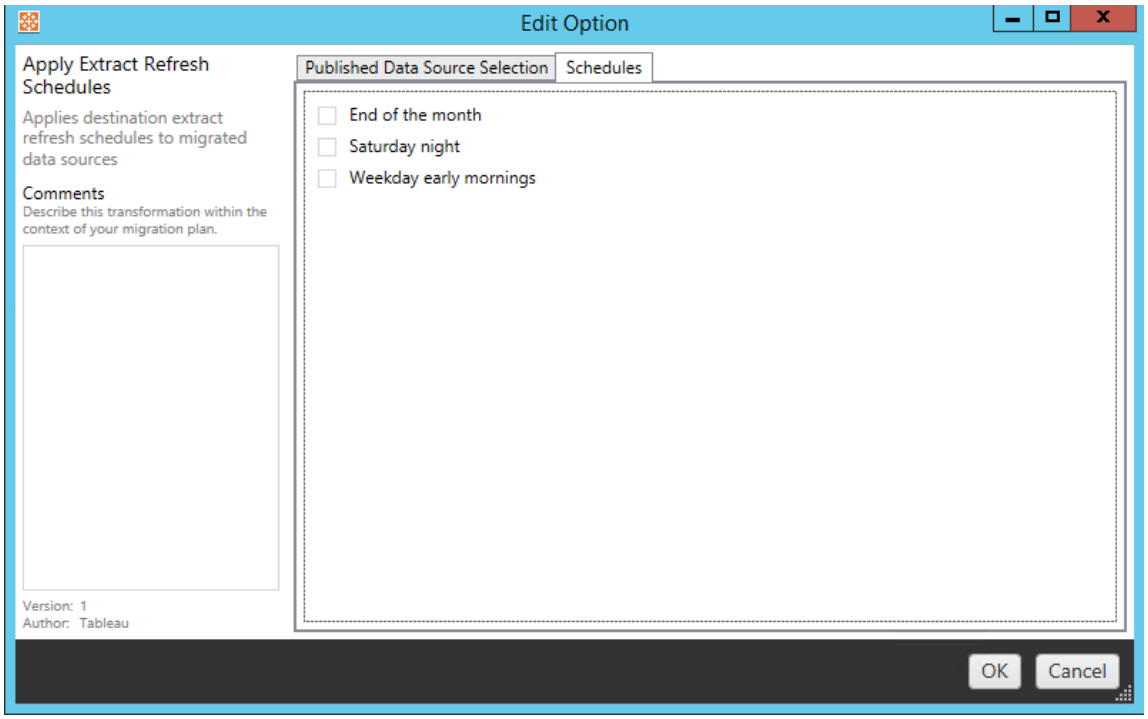


さらに、各変換について、[Edit Transformation (変換の編集)] ウィンドウの左側にある [**Comments**] セクションにメモを入力できます。

Apply Extract Refresh Schedules (抽出更新スケジュールの適用)

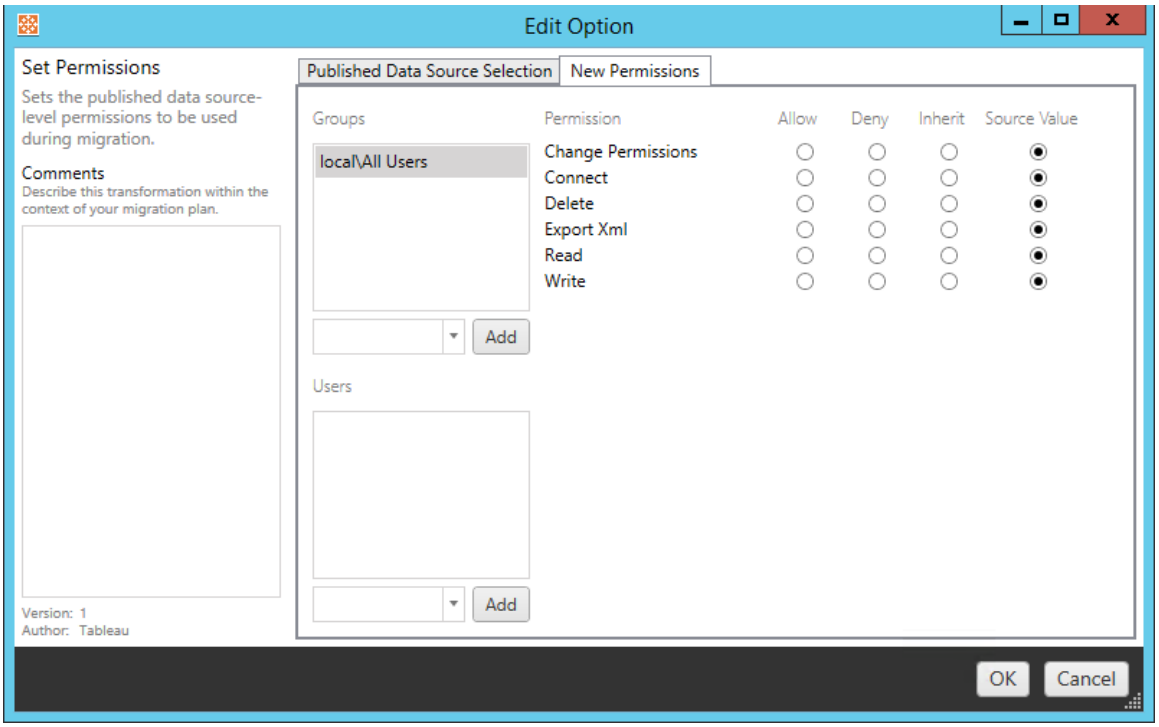
この変換により、移行されたデータソースに展開先の抽出更新スケジュールが適用されます。生成済みスケジュールのリストは、展開先から取得されます。

> **注:** 抽出更新スケジュールは、Tableau Cloud で作成することはできません。展開先が Tableau Cloud サイトの場合、このオプションは使用できません。詳細については、移行に関する制限事項を参照してください。



パーミッションの設定

最後のタイプの変換は、選択したデータソースに関するパーミッションの編集です。[グループ] または [ユーザー] 内に値を入力し、**[Add]** をクリックします。必要に応じてパーミッションを調整してください。パーミッションの許可 (**[Allow]**)、パーミッションの拒否 (**[Deny]**)、継承 (**[Inherit]**)、ソース値の保持 (**[Source Value]**) の 4 種類のオプションがあります。



ステップ 5: 次のステップに進む

準備ができたら、**[Next]** をクリックし、計画段階の移行計画: パーミッションと所有権 セクションに進みます。

実行可能なユーザー

Explorer 以上の役割を持つ Tableau サイトユーザー。コンテンツを移行するには、移行元サイトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

移行計画: パーミッションと所有権

Content Migration Tool を使用すると、さまざまなネットワークドメインに存在する Tableau 環境や、ユーザー名やグループの構文に違いがある Tableau 環境にワークブックおよびデータソースのパーミッションをレプリケートできます。コンテンツを移行先の場所にパブリッシュした後、ユーザーパーミッションのマッピングを作成して、コンテンツをカスタマイズしてセキュリティで保護することができます。**[Apply User Mappings (ユーザー マッピングを適用)]** に加えて、**[Copy Project**

Permissions]、**[Copy Workbook Permissions]**、または **[Copy Data Source Permissions]** が選択されている場合、マッピングが適用されます。

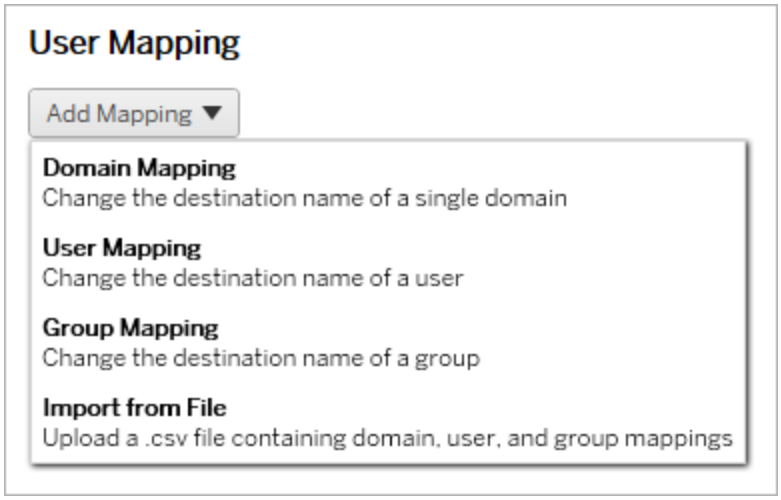
詳細については、移行計画: ソースプロジェクト、移行計画: ワークブック、移行計画: パブリッシュ済みデータソースを参照してください。

マッピングの制限事項

- 移行先の場所でマッピングされたユーザーまたはグループが見つからない場合、Content Migration Tool はその移行プロセスを停止します。最初のエラーが発生した後に後続のユーザー パーミッションまたはグループ パーミッションのマッピングはチェックされないため、この計画を再実行する必要があります。
- ソース コンテンツに同じ名前の複数のユーザーおよびグループに対するパーミッションがある場合、Content Migration Tool はパーミッションをレプリケートできません。これは、別のドメインをソースとする重複したユーザー名またはグループ名がある場合にのみ発生します。

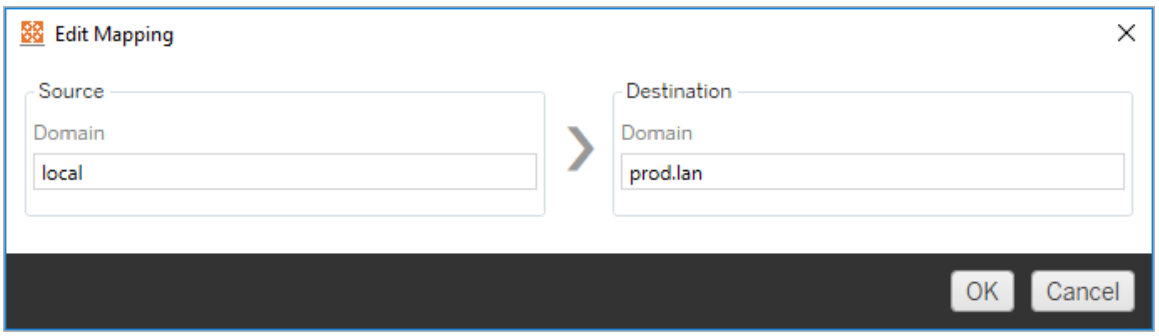
ステップ 1: マッピングを追加する

ユーザー パーミッションのマッピングを追加するには、**[Add Mapping]** をクリックし、ドメイン、ユーザー、グループの名前を変更するか、コンマ区切りの値 (CSV) ファイルからマッピングをインポートするかのどちらかを選択します。Content Migration Toolが移行先の場所にある許可に一致できない場合、ソース コンテンツは移行されません。



ドメイン マッピング

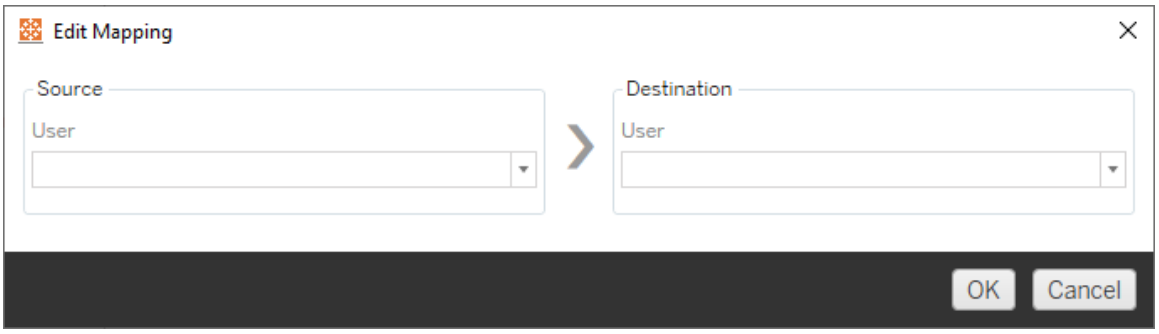
ドメイン パーミッションのマッピングは、移行先の場所にあるすべてのユーザーとグループに適用されます。ソース ドメインまたは移行先ドメインが不明な場合は、Tableau サイトでユーザー ページとグループ ページを確認できます。ローカル ユーザー プロビジョニングが選択されている場合は、そのドメインを `local` として指定する必要があります。



ユーザー マッピング

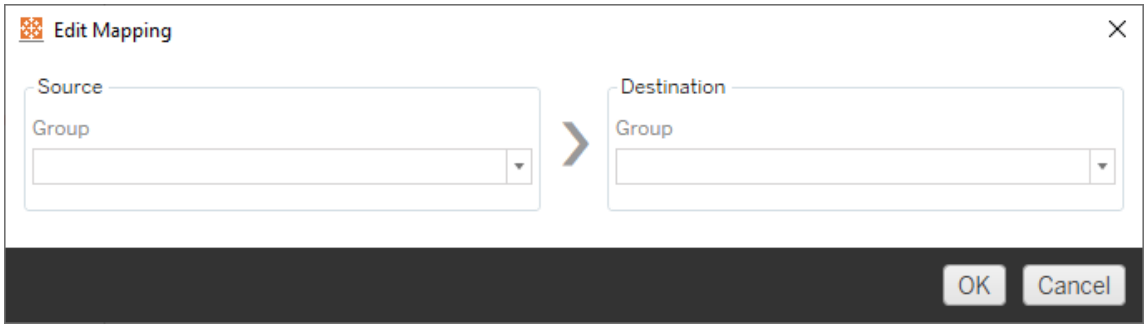
ユーザー パーミッションのマッピングでは、構文 `domain\user` を使用して、移行元と移行先の場所からユーザーのリストが自動的に生成されます。存在しないユーザーの名前を入力して保存することはできません。

注: 同じサイトのプロジェクト間で移行する場合、移行先の場所とユーザーのリストは同じになります。マッピングを使用して、サイトでコンテンツの所有権を `User_A` から `User_B` に更新できます。



グループ マッピング

グループ パーミッションのマッピングでは、構文 `domain\group` を使用して、移行元と移行先の場所からユーザーのリストが自動的に生成されます。存在しないグループの名前を入力して保存することはできません。



CSV ファイルからマッピングをインポート

バージョン 2021.4 以降、ドメイン、ユーザー、およびグループのマッピングを含む CSV ファイルをインポートして、データの移行を迅速に準備できるようになりました。マッピングをインポートすると、Content Migration Tool の外部でマッピングを一括で作成および編集できるため、移行計画の実行に必要な手動によるステップを減らすことができます。マッピングをインポートするには、[Add Mapping (マッピングの追加)] メニューから **[Import from File (ファイルからインポート)]** を選択します。

CSV ファイル形式の要件

マッピングをインポートするための CSV ファイルを作成する場合は、ファイルが次の要件を満たしていることを確認してください。

- ファイルに列ヘッダーが含まれていない。Tableau は、すべての行がマッピングを表すものとみなします。
- ファイルには、行ごとに 3 つのコンマ区切り値 (マッピングタイプ、移行元ドメイン/ユーザー/グループ、および移行先ドメイン/ユーザー/グループ) が含まれている。

- サーバーが Active Directory 認証を使用する場合は、ユーザー名とグループのドメインを含め、ローカル アイデンティティストアが使用される場合は "local" を含める。

次の表に示すように、マッピング タイプには "ドメイン"、"ユーザー"、または "グループ" を指定する必要があります。移行元列と移行先列には、Active Directory とローカル アイデンティティストアの構文例が示されています。CSV ファイルの実際の値は、組織によって異なります。

マッピング タイプ	移行元	移行先
ドメイン	`<domain>`	`<domain>`
ユーザー	`<domain>\<user name>` local\`<user name>`	`<domain>\<user name>` local\`<user name>`
グループ	`<domain>\<group name>` local\`<group name>`	`<domain>\<group name>` local\`<group name>`

ユーザー パーミッションのマッピングをインポート

Content Migration Tool でユーザー パーミッションのマッピングをインポートするには:

1. **[Add Mapping (マッピングの追加)]** をクリックしてから、**[ファイルからインポート]** を選択します。
2. ダイアログ ウィンドウで、**[Export CSV (CSV のエクスポート)]** をクリックして、すべてのユーザーとグループを含む .csv ファイルをソース サイトからエクスポートします。結果のファイルをテキスト エディタで編集して、移行先サイトのマッピングを追加します。

   マッピング ファイルがすでにある場合は、ステップ 3 に進みます。

   > **注:** エクスポートされた CSV ファイルには、ソース サイトのドメインは含まれていません。ドメインのマッピングを作成するには、ドメインを CSV に手動で追加する必要があります。

3. **[Import Mappings (マッピングのインポート)]** をクリックして、インポートするマッピング ファイルを選択します。

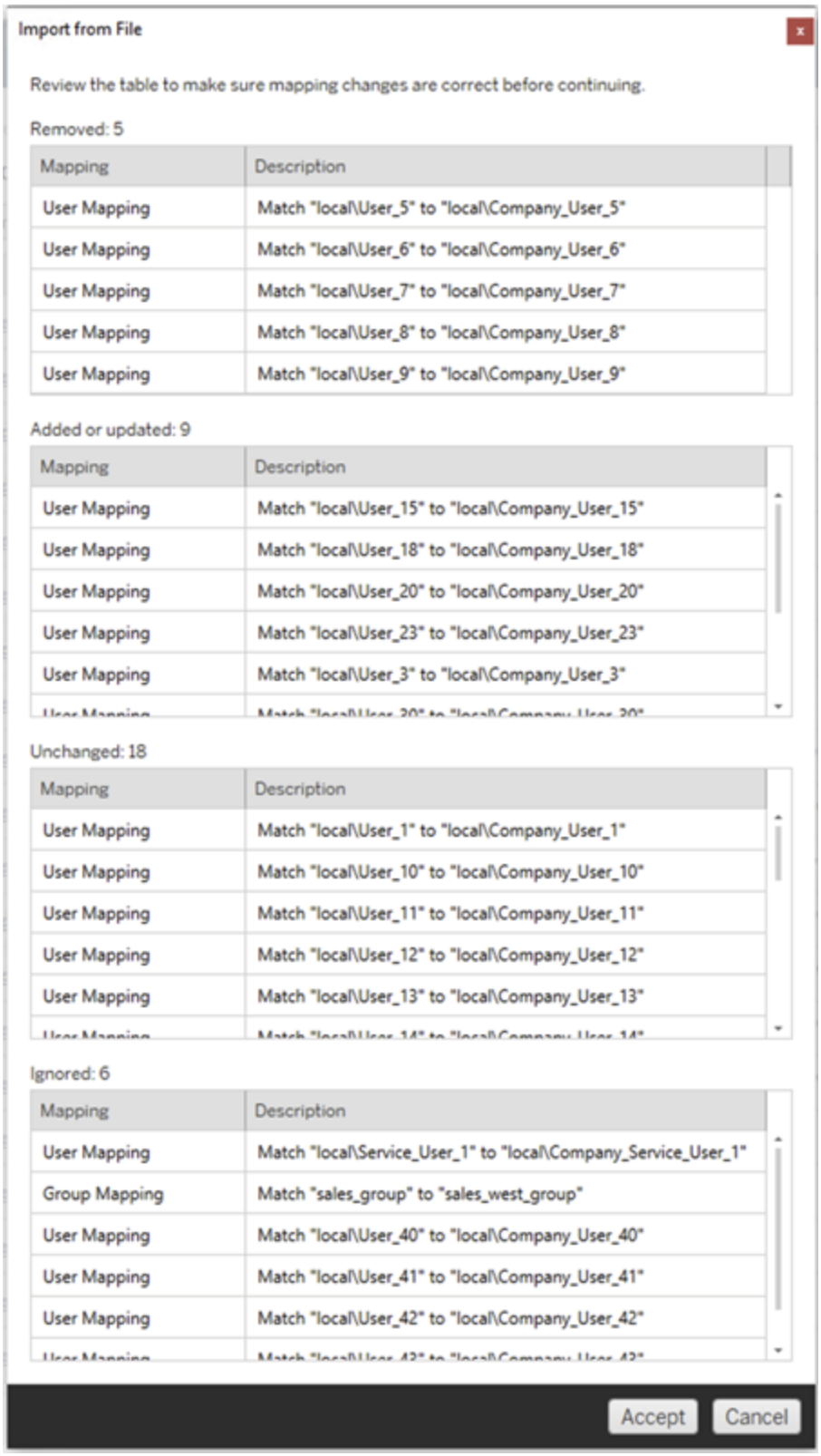
Content Migration Tool は、ファイルをインポートする際のマッピングのエラーを検証します。エラーが検出された場合は、CSV ファイルの各エラーを修正してから、もう一度インポートする必要があります。

CSV インポートの例

次の例は、複数のマッピング タイプを含む CSV ファイルを示しています。

```
user,local\hwilson,companyx.lan\henry.wilson
user,local\jjohnson,companyx.lan\janna.johnson
user,local\mkim,companyx.lan\michele.kim
user,local\fsuzuki,companyx.lan\fred.suzuki
user,local\awang,companyx.lan\alan.wang
user,local\snguyen,companyx.lan\susan.nguyen
user,local\lrodriguez,companyx.lan\laura.rodriguez
user,local\agarcia,companyx.lan\ashley.garcia
group,local\All Users,companyx.lan\All Users
group,local\Finance Team,companyx.lan\Finance Group
domain,dev.mycompany,prod.mycompany
```

CSV のインポート中にプレビュー ウィンドウが表示され、マッピングが削除、追加、更新、変更されていない、および無視されたことが示されます。マッピングの変更が正しいことを確認し、**[Accept (承認)]** をクリックします。


Import from File
Review the table to make sure mapping changes are correct before continuing.
Removed: 5
Mapping
Description
User Mapping
Match "local\User_5" to "local\Company_User_5"
User Mapping
Match "local\User_6" to "local\Company_User_6"
User Mapping
Match "local\User_7" to "local\Company_User_7"
User Mapping
Match "local\User_8" to "local\Company_User_8"
User Mapping
Match "local\User_9" to "local\Company_User_9"
Added or updated: 9
Mapping
Description
User Mapping
Match "local\User_15" to "local\Company_User_15"
User Mapping
Match "local\User_18" to "local\Company_User_18"
User Mapping
Match "local\User_20" to "local\Company_User_20"
User Mapping
Match "local\User_23" to "local\Company_User_23"
User Mapping
Match "local\User_3" to "local\Company_User_3"
Unchanged: 18
Mapping
Description
User Mapping
Match "local\User_1" to "local\Company_User_1"
User Mapping
Match "local\User_10" to "local\Company_User_10"
User Mapping
Match "local\User_11" to "local\Company_User_11"
User Mapping
Match "local\User_12" to "local\Company_User_12"
User Mapping
Match "local\User_13" to "local\Company_User_13"
Ignored: 6
Mapping
Description
User Mapping
Match "local\Service_User_1" to "local\Company_Service_User_1"
Group Mapping
Match "sales_group" to "sales_west_group"
User Mapping
Match "local\User_40" to "local\Company_User_40"
User Mapping
Match "local\User_41" to "local\Company_User_41"
User Mapping
Match "local\User_42" to "local\Company_User_42"
Accept
Cancel

マッピングが正常にインポートされると、ステップ 2 の説明に従って、マッピングの順序を編集、削除、または変更できます。

ステップ 2: マッピング順序の変更

パーミッションのマッピングを作成した後、**Up** または **Down** オプションを使用して、移行時にマッピングを処理するタイミングを決定する順序を変更できます。ドメイン、ユーザー、またはグループがパーミッションのマッピングで処理される場合、そのソースドメイン、ユーザー、またはグループに対するそれ以降のマッピングは無視されます。

次の例では、`User_A` のパーミッションが `User_B` にマッピングされています。Content Migration Toolは、`User_A` が既に処理されているため、2 番目のマッピングを無視します。

	Mapping	Description
Edit Delete Up Down	User Mapping	Match "local\User_A" to "local\User_B"
Edit Delete Up Down	User Mapping	Match "local\User_A" to "local\User_C"

次の例では、パーミッションの最初のマッピングにより、すべてのユーザーのドメインが `prod` に関連付けられます。Content Migration Toolは、`User_A` のドメインが既に処理されているため、パーミッションの2 番目のマッピングを無視します。

	Mapping	Description
Edit Delete Up Down	Domain Mapping	Match "local" to "prod"
Edit Delete Up Down	User Mapping	Match "User_A" to "dev\User_B"

ステップ 3: 次のステップに進む

準備ができたら、**[Next]** をクリックし、計画段階の移行計画: 移行スクリプトセクションに進みます。

実行可能なユーザー

Explorer 以上の役割を持つTableau サイトユーザー。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先

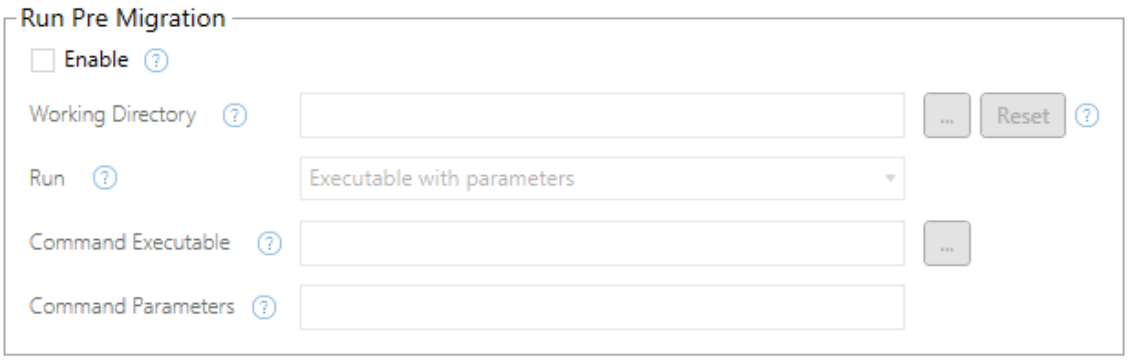
サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

移行計画: 移行スクリプト

Tableau Content Migration Tool での移行計画作成の次のステップは、移行前後に計画で実行するスクリプトを作成することです。

ステップ 1: 移行前

画面の **[Run Pre Migration]** セクションは、移行前に実行するスクリプト専用です。



各フィールドには、カーソルを合わせると情報が得られるヘルプアイコンがあります。移行前のスクリプトを開始するには、**[Enable]** を選択して以下のフィールドをアクティブ化します。

Working Directory

これは、スクリプトの作業ディレクトリです。既定のディレクトリは、移行計画と同じフォルダーです。[参照] ボタンをクリックして別のフォルダーを選択します。**Reset** ボタンを押すと、現在の移行計画フォルダーが作業ディレクトリとして復元されます。

Run

このドロップダウンを使用して、カスタム スクリプトを実行するか、パラメーターを使用した実行可能ファイルを実行するかを選択できます。

Command Executable

[Run] メニューから**[Executable with Parameters]** を選択した場合に、このフィールドが表示されます。これは、移行前に実行する実行可能なコマンドへのファイル パスです。パスを直接入力するか、[参照] ボタンを使用して実行可能ファイルを見つけます。これは必須フィールドです。

Command Parameters

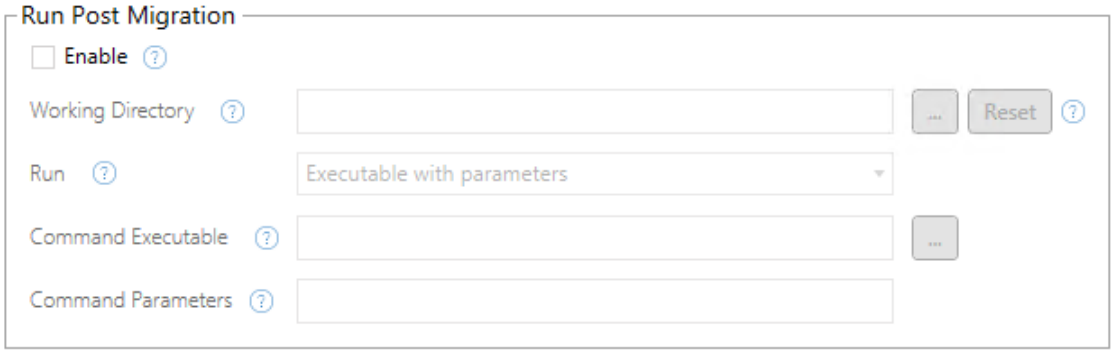
[実行] ドロップダウン メニューで **[Executable with Parameters]** を選択した場合に、このフィールドが表示されます。実行可能なコマンドで使用するには、コマンド ライン パラメーターをここに入力します。

スクリプト

[実行] メニューから**[Custom script]** を選択した場合は、ここに移行前スクリプトを入力します。*.cmd ファイルとして実行されます。これは必須フィールドです。

ステップ 2: 移行後

[Run Post Migration] 画面の半分は、移行後に実行されるスクリプト専用です。



各フィールドには、カーソルを合わせると情報が得られるヘルプ アイコンがあります。移行後のスクリプトを開始するには、**[Enable]** を選択して以下のフィールドをアクティブ化します。

Working Directory

これは、スクリプトの作業ディレクトリです。既定のディレクトリは、移行計画と同じフォルダーです。[参照] ボタンをクリックして別のフォルダーを選択します。**Reset** ボタンを押すと、現在の移行計画フォルダーが作業ディレクトリとして復元されます。

Run

このドロップダウンを使用して、カスタム スクリプトを実行するか、パラメーターを使用した実行可能ファイルを実行するかを選択できます。

Command Executable

[**Run**] メニューから [**Executable with Parameters**] を選択した場合に、このフィールドが表示されます。これは、移行前に実行する実行可能なコマンドへのファイル パスです。パスを直接入力するか、[参照] ボタンを使用して実行可能ファイルを見つけます。これは必須フィールドです。

コマンド パラメーター

[**Run**] メニューから [**Executable with Parameters**] を選択した場合に、このフィールドが表示されます。実行可能なコマンドで使用するには、コマンド ライン パラメーターをここに入力します。

スクリプト

[**Run**] メニューから [**Custom script**] を選択した場合は、移行後のスクリプトをここに入力します。*.cmd ファイルとして実行されます。これは必須フィールドです。

ステップ 3: 次のステップに進む

準備ができたら、[**Next**] をクリックします。

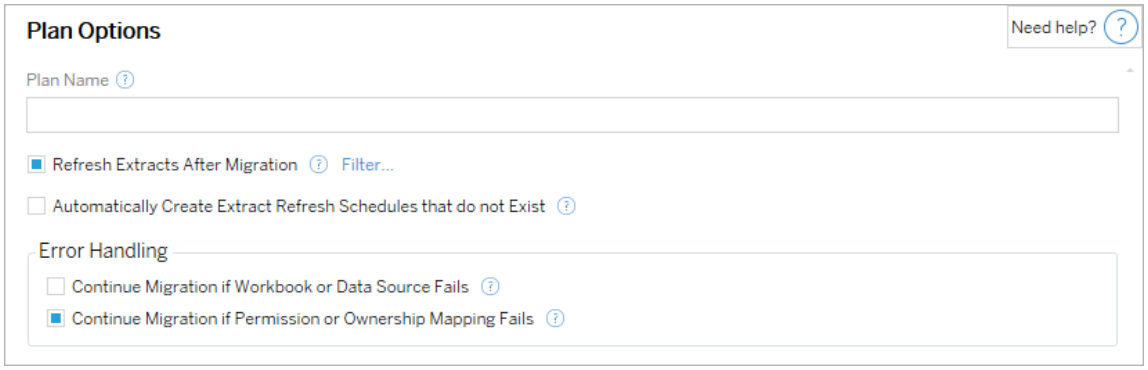
実行可能なユーザー

Tableau サイトのユーザー (Explorer 以上の役割を持つ)。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

移行計画: 計画のオプション

Tableau Content Migration Tool で移行計画を作成する最後のステップは、計画のオプションを構成することです。

ステップ 1: オプションの構成



[Plan Name] は、Content Migration Tool に表示される計画の名前です。計画名にはわかりやすい名前を付けることをお勧めします。

以下のオプションを利用できます。

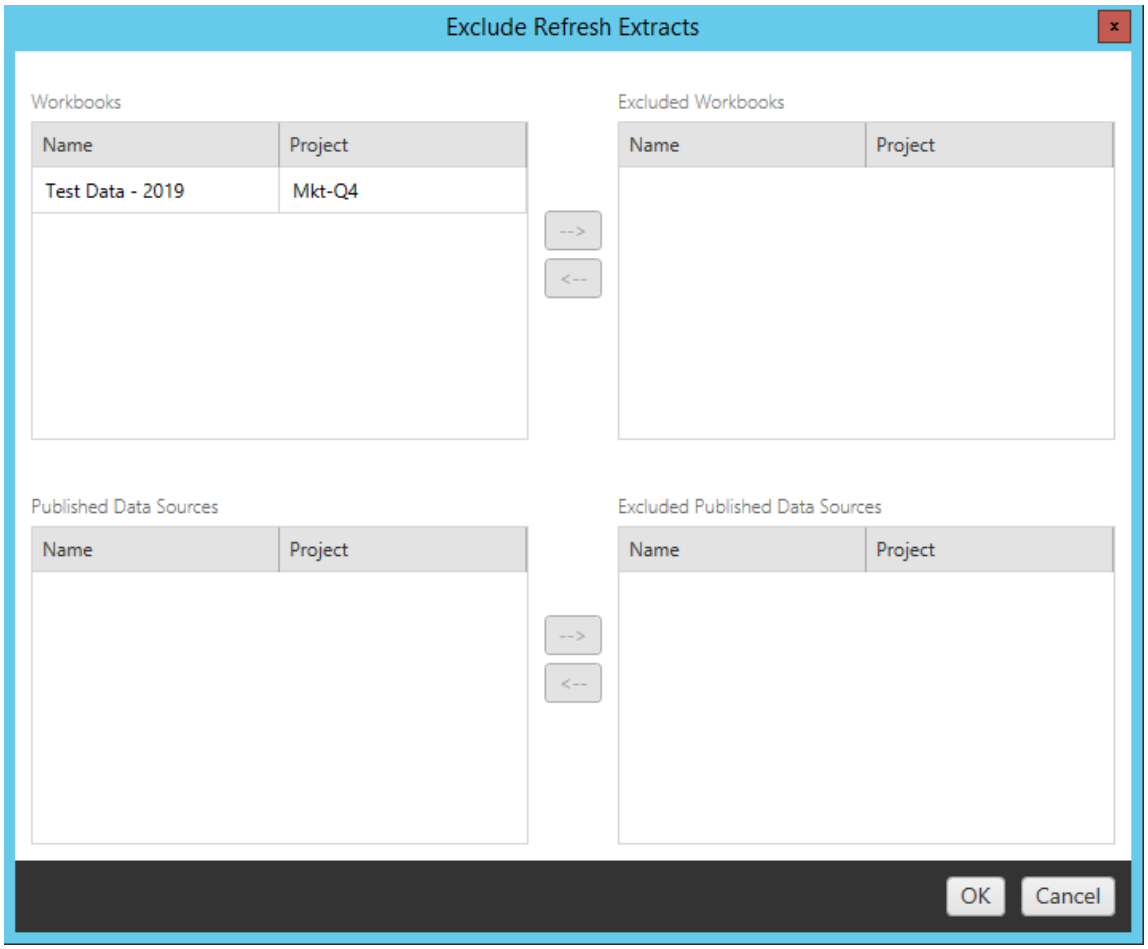
- **Refresh Extracts After Migration**: このオプションを選択すると、Content Migration Tool で移行時にデータ抽出が変更された可能性があることが検出された場合、移行後すぐにデータ抽出が更新されます。**Filter** リンクをクリックして、特定の抽出物を除外します。詳細については、「抽出更新の除外」を参照してください。
- **Automatically create Extract Refresh Schedules that do not Exist**: 存在しない展開先の抽出スケジュールを自動的に作成します。オフにすると、展開先サーバーに存在しない展開元のスケジュールはコピーされません。

- **Continue Migration if Workbook or Data Source Fails**: チェックをオンにすると、ワークブックまたはデータソースを移行するときにエラーが発生しても、移行は停止しません。エラーがログに記録され、移行が続行されます。バージョン管理中のエラーの場合は、必ず移行が停止します。

- **Continue Migration if Permission or Ownership Mapping Fails (パーミッションまたは所有権のマッピングが失敗した場合に移行を続行する)**: チェックをオンにすると、パーミッションまたは所有権のコピー中にエラーが発生しても、移行は停止しません。エラーがログに記録され、移行が続行されます。

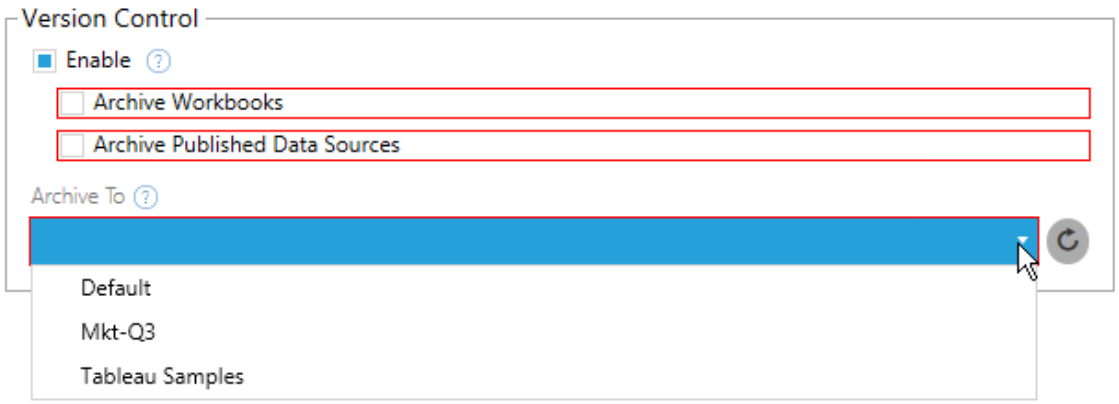
抽出更新の除外

[移行後に抽出を更新する] の横にある **[フィルター]** をクリックすると、自動的に更新されないワークブックまたはパブリッシュされたデータソースを選択できます。矢印ボタンを使用して除外する項目を選択し、**OK** をクリックします。



ステップ 2: バージョン管理

これらのオプションを使用すると、展開先のサイトで移行済みのワークブックに置き換えられる可能性のある既存のワークブックが消失されるのを回避できます。



Enable を選択して、コンテンツの以前のバージョンを保存します。ワークブックおよび/またはパブリッシュされたデータソースをアーカイブするかどうかを選択できます。バージョン管理を有効にしたら、**Archive To** メニューからプロジェクトを選択する必要があります。このメニューには、展開先サイトにあるすべてのプロジェクトが一覧表示されています。バージョン管理されたコンテンツを保存するためのアーカイブプロジェクトを別途作成することをお勧めします。サイトで追加または変更されたプロジェクトを表示するには、更新ボタンをクリックします。

ステップ 3: 計画の保存

計画オプションを選択したら、**Save Plan** をクリックして、後で使用できるように計画を保存します。計画は、ローカルマシンの `Documents\Tableau Content Migration Tool Plans` フォルダーに保存されます。

ステップ 4: 次のステップに進む

準備ができたら、**Verify & Run** をクリックして計画段階を終了し、計画の実行を準備します。

実行可能なユーザー

Tableau サイトのユーザー (Explorer 以上の役割を持つ)。コンテンツを移行するには、移行元イトのワークブックに対して **[表示]** および **[ダウンロード/コピーの保存]** 機能を持っていて、かつ、移行先サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

抽出を使用するワークブックとデータソースの移行

Tableau Server ユーザーは、元のデータのコピーまたは一部である抽出をパブリッシュすることができます。これらの抽出は、ワークブックやデータソースに埋め込むことができます。既定では、Tableau Content Migration Tool を使用して抽出を含むワークブックやデータソースを移行すると、その抽出もワークブックやデータソースと共に移行されます。Content Migration Tool では、以下の動作を制御するいくつかのオプションを利用できます。

- **ライブ接続への切り替え**

  [Remove Extract (抽出の削除)] 変換を移行計画に追加し、移行中にワークブックまたはデータソースから抽出を削除することができます。通常どおり、展開元のワークブックやデータソースは変更されません。展開先のサイトに移行されたワークブックやデータソースのコピーからは、抽出が削除されます。これにより、データ接続がライブ接続へと効率的に切り替えられます。

- **移行後の抽出の更新**

  移行計画で **[Refresh Extracts After Migration (移行後に抽出を更新)]** オプションを有効にすると、ワークブックまたはデータソースの移行後に抽出更新タスクを直ちにスケジュールすることができます。

  移行計画で **[接続情報の設定]** 変換を使用し、データ接続が異なるデータのセット (例えば、異なるデータベース サーバーやデータベースなど) を指すようにデータ接続を変更する場合、**[移行後に抽出を更新]** オプションを使用することはお勧めしません。異なるデータを指すように接続情報を変更し、**[Refresh Extracts After Migration (移行後に抽出を更新)]** オプションを使用すると、セキュリティ上の問題となる可能性がある方法でデータが不用意に公開される場合があります。

  詳細については、「オプション: 移行後に抽出を更新」を参照してください。

抽出を使用するデータ接続の変更

Tableau データ接続は、データソースを直接クエリするライブ接続であるか、データソースの抽出です。抽出は、元のデータのコピーまたは一部で、ワークブックまたはデータソースに埋め込むことが

できます。存在する場合には、ビューは参照元データソースではなく抽出からデータのクエリを実行します。

通常、移行中にデータソース接続を変更して、(展開先サイトで展開元サイトとは異なる) データベースを指すようにします。

たとえば、ワークブックをステージング環境のサイトから実稼働環境のサイトに移行するには、ワークブック内のデータ接続を更新して実稼働環境のデータベースに接続します。これを実装するには、移行計画で **[Set Connection Info (接続情報の設定)]** 変換を使用します。これにより、ワークブックをステージング環境から本番環境にコピーし、本番環境のデータベースを指すようにデータ接続を更新する移行計画が作成されます。

ワークブックで抽出を使用する場合は、追加の作業が必要です。このシナリオでは、ワークブックが移行され、ライブデータ接続が更新されます。ただし、ビューには、ステージング データベースのデータが引き続き表示されます。これは、展開元 (ステージング) サイトからコピーされたステージング データベースの抽出が含まれているためです。これにはいくつかの対処方法があります。

オプション 1: パブリッシュ済みデータソースの使用

代わりにパブリッシュ済みデータソースを使用するようにワークブックを変更することができます。この方法を使用すると、抽出はパブリッシュ済みデータソースの一部として管理され、ライブデータベースやデータ抽出への接続に関して心配する必要がなくなるので、そのデータソースを使用するワークブックへの更新の移行を簡略化することができます。

オプション 2: 移行中に抽出を削除

移行計画に **[Remove Extract (抽出の削除)]** 変換を追加することができます。これにより、ワークブックから抽出が削除され、データソースからライブ接続への切り替えが効率的に行われます。

オプション 3: 移行後に抽出を更新

移行計画で **[Refresh Extracts After Migration (移行後に抽出を更新)]** オプションを使用できます。これを使用すると、抽出がワークブックと共に移行されますが、移行の完了後にはそのワークブックの抽出更新タスクが直ちにスケジュールされます。

このオプションを**[Set Connection Info (接続情報の設定)]**変換と組み合わせて使用すると、セキュリティ上の問題が生じる可能性があるため、通常はお勧めしません。

問題とは、移行の完了から抽出更新タスクの完了までの間に、展開先サイトのに移行されたワークブックに、古い(移行元の)抽出データが表示されたままになることです。抽出の更新タスクが失敗する場合、古い/展開元の抽出データは、抽出が更新されるまで残ります。

ステージング環境から本番環境への移行という上記のようなシナリオではこれが許容される場合がありますが、留意すべき点として、ワークブックが最近移行されたばかりで抽出がまだ更新されていないために、ワークブックに古い/ステージング環境のデータが表示されていることにワークブックのユーザーが気付かない場合があることです。

[Set Connection Info (接続情報の設定)]を使用してデータ接続を変更し、顧客またはクライアントのデータの異なるセットを指す他のシナリオでは、移行後に抽出を更新するまでワークブックの抽出に別のクライアントや顧客のデータが含まれるという深刻なセキュリティ上の問題が発生する可能性があります。

この問題を軽減する1つの方法は、2段階の移行を実装する方法です。この方法では、2つの移行計画(下記の各手順に対して1つずつ)を作成する必要があります。これにより、ワークブックとデータソースにアクセスできるようになる前に、それらの抽出が最新状態になります。

- **段階 1:** 管理者のみがアクセス権を持つ展開先サイト上のプロジェクトにコンテンツを移行します。この移行では、抽出更新に失敗しても許可を受けていないユーザーが古いデータを表示する機会はないため、**[Refresh Extracts After Migration (移行後に抽出を更新)]**オプションと**[Set Connection Info (接続情報の設定)]**変換を使用してデータ接続を更新することができます。
- **段階 2:** 段階 1 が完了して抽出更新が成功したことを確認した後に、2番目の移行計画を実行し、段階 1 の展開先からエンドユーザーに表示される最終的な展開先にコンテンツを移行します。

実行可能なユーザー

Explorer 以上の役割を持つTableau サイトユーザー。コンテンツを移行するには、移行元イトのワークブックに対して**[表示]**および**[ダウンロード/コピーの保存]**機能を持っていて、かつ、移行先

サイトのターゲットプロジェクトに対して **[表示]** および **[パブリッシュ]** 機能を持っている必要があります。詳細については、パーミッションを参照してください。

認証資格情報が埋め込まれたワークブックとデータソースの移行

バージョン 2023.1 以降、承認されたユーザーは、認証資格情報が埋め込まれたワークブックとパブリッシュされたデータソースを Tableau Server から Tableau Cloud に移行することができます。Content Migration Tool で移行する前に、追加の設定が必要です。

> **注:** Content Migration Tool は、OAuth 接続の埋め込み認証資格情報の移行をサポートしていません。詳細については、「移行に関する制限事項」を参照してください。

概要

Content Migration Tool (CMT) を使用した埋め込み認証資格情報の移行は、Tableau Server をソースサイトとし、Tableau Cloud を宛先サイトとした接続時に使用できます。どちらのサイトも、Advanced Management のライセンスが必要です。

要件について確認したので、移行のしくみについて説明します。Tableau Cloud のサイト管理者と TSM 管理者 (場合によっては同じ人) と緊密に連携し、機能の許可と、サイトユーザーの承認を行う必要があります。機能がアクティブ化された後、許可されたサイトユーザーは移行計画を作成し、パブリッシュのオプション (「*ワークブックの埋め込み認証資格情報の移行*」と「*データソースの埋め込み認証資格情報の移行*」) を選択します。

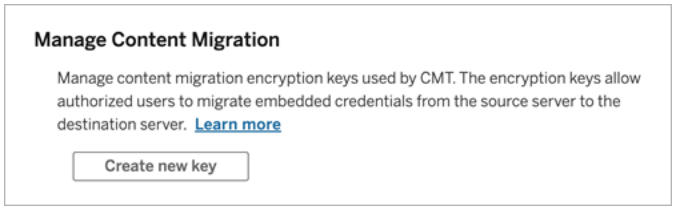
移行計画を実行すると、必要なすべてのコンテンツの認証資格情報が暗号化されたコンテンツマニフェストで Tableau Server から Tableau Cloud に送信されます。CMT がコンテンツをパブリッシュするとき、宛先の Tableau Cloud サイトは、一致した認証資格情報をマニフェストからコンテンツ (ワークブックまたはパブリッシュされたデータソース) に安全に埋め込みます。移行中に発生した問題は、CMT の [エラーと警告] タブに表示されます。詳細については、「移行計画の概要」を参照してください。

埋め込み認証資格情報の移行を許可する

Tableau Server から Tableau Cloud への埋め込み認証資格情報の移行を許可するには、次の手順を使用します。

Tableau Cloud

1. ブラウザー ウィンドウを開き、サイト管理者として Tableau Cloud にサインインします。
2. **[設定]**、**[一般]** の順に選択し、**[コンテンツ移行の管理]** まで下にスクロールします。



3. **[新しいキーを作成]** をクリックし、暗号鍵ペアを生成します。

> **注:** 公開鍵は一度だけ表示されます。設定を完了する前にキーを紛失した場合は、新しいキーを生成する必要があります。

4. 結果のウィンドウで **[クリップボードにコピー]** をクリックし、ウィンドウを閉じます。
5. 公開鍵をファイルに貼り付けて、安全な場所に保管します。TSM 管理者は、公開鍵を使用して移行を許可します。公開鍵の有効期限は [設定] ページで確認できます。

TSM コマンド ライン インターフェイス

1. クラスター内のノード上で、`tsmadmin` グループのメンバー アカウントを使用してコマンドプロンプトを開きます。
2. `tsm security authorize-credential-migration` を使用して、Tableau Cloud サイトへの埋め込み認証資格情報の移行を許可します。詳細については、「tsm

security」を参照してください。

```
tsm security authorize-credential-migration --source-site-url-
namespace <Tableau Server site ID> --destination-site-url-
namespace <Tableau Cloud site ID> --destination-server-url
<Tableau Cloud site url> --authorized-migration-runner
<username> --destination-public-encryption-key <public key>
```

> **注:** リモート ノードからTSM コマンドを実行する場合は、`tsm security authorize-credential-migration` を実行する前に `tsm login` を使用して、Tableau Server Administration Controller サービスでセッションを認証します。

3. (オプション) 付与した権限を取り消すには、`tsm security cancel-credential-migrations` を使用します。デフォルトでは、移行の承認は 7 日または `--expiration-time-in-days` オプションで指定された日数で有効期限が切れます。

Content Migration Tool

1. Content Migration Tool を開き、**[新しいプランを作成]** または **[プランを参照]** を選択します。
2. [サイト] ページで **[Tableau にサインイン]** をクリックし、[Tableau Server] をソース、[Tableau Cloud] を宛先として接続します。埋め込み認証資格情報の移行は、Tableau Server から Tableau Cloud へ移行する場合にのみ実施できます。
3. 移行計画を作成し、次のパブリッシュ オプションを選択します。
   - [ワークブックのパブリッシュ オプション] ページで、**[ワークブックの埋め込み認証資格情報の移行]** を選択します。詳細については、「移行計画: ワークブック」を参照してください。
   - [データ ソースのパブリッシュ オプション] ページで、**[データ ソースの埋め込み認証資格情報の移行]** を選択します。詳細については、「移行計画: パブリッシュ済みデータ ソース」を参照してください。
4. 準備ができたら、**[検証と実行]** をクリックして、移行を開始します。

選択したワークブックとパブリッシュされたデータ ソースは Tableau Cloud サイトに移行され、認証を求められることはありません。埋め込まれた認証資格情報の移行中に問題が発生した場合は、「トラブルシューティング」を参照してください。

トラブルシューティング

このセクションでは、発生する可能性のある一般的な移行の問題と、それらを解決するための提案を記載しています。

埋め込み認証資格情報を移行するオプションがない

埋め込み認証資格情報のみを、Tableau Server から Tableau Cloud サイトへ移行できます。Tableau Server と Content Migration Tool は、バージョン 2023.1 以降を実行している必要があります。詳細については、「Tableau Content Migration Tool のインストール」を参照してください。

埋め込み認証資格情報の移行に失敗した

CMT の [エラーと警告] タブに、埋め込み認証資格情報の移行に失敗したことを示すエラーが表示される場合があります。これは、移行の承認に使用された公開鍵の有効期限が切れた場合に発生する可能性があります。

Tableau Cloud のサイト管理者は、[設定] ページに移動して、公開キーが有効であることを確認してください。公開鍵の有効期限が切れている場合は、新しい暗号化ペアを作成して移行を承認する必要があります。詳細については、「埋め込み認証資格情報の許可」を参照してください。

実行可能なユーザー

- Tableau Cloud のサイト管理者と TSM 管理者は、埋め込まれた認証資格情報の移行を許可する必要があります。
- 許可されたサイト ユーザーには、Explorer 以上の役割が必要です。また、ソース サイトのワークブックに対する表示権限およびコピーのダウンロード/保存権限と、宛先サイトのターゲット プロジェクトに対する表示権限およびパブリッシュ権限が必要です。

詳細については、パーミッションを参照してください。

Tableau Content Migration Tool コンソール ランナーの使用

Tableau Content Migration Tool には、移行を実行するためのコマンド ライン ユーティリティ `tabcmt-runner.exe` が含まれています。このユーティリティはインストール フォルダーにあります。既定のインストール フォルダーは `%PROGRAMFILES%\Tableau\Tableau Content Migration Tool` です。

注: `tabcmt-runner.exe` ユーティリティは、Content Migration Tool グラフィカル アプリケーションの構成に使用される `tabcmt.cmd` コマンド ライン ユーティリティとは異なります。`tabcmt.cmd` の詳細については、Tableau Content Migration Tool コマンド ライン インターフェイスの使用を参照してください。

使用方法:

- tabcmt-runner [options] <plan_file.tcmx>
- tabcmt-runner license --remove
- tabcmt-runner license <new license key>
- tabcmt-runner license <license file path> [--passphrase=<license file passphrase>]
- tabcmt-runner encryption --reset
- tabcmt-runner encryption <new_key>
- tabcmt-runner improvement [on|off]
- tabcmt-runner --help
- tabcmt-runner --version
- tabcmt-runner script-warning [on|off]

オプション:

- --version
- --help
- --quiet
- --info
- --logfile=VALUE
- --src-user=VALUE
- --src-password=VALUE
- --dest-user=VALUE
- --dest-password=VALUE

- --https=VALUE
- --allow-scripts

計画の実行

移行計画を直ちに実行します。

```
tabcmt-runner [options] <plan file>
```

使用可能なオプション:

- `--logfile=<file name>`: ファイル名を設定し、出力をログに記録します
- `--https=<secure|legacy>`: HTTPS モードを設定します
- `--quiet` : stdout へのログの記録を無効にします
- `--src-user=<username>`: ソース接続のユーザー名を設定します
- `--src-password=<password>`: ソース接続のパスワードを設定します
- `--dest-user=<username>`: 移行先接続のユーザー名を設定します
- `--dest-password=<password>`: 移行先接続のパスワードを設定します

終了コード:

- 0: 移行が正常に完了したことを示します。
- 1: 移行が成功したが警告メッセージがログに記録されたことを示します。
- 2: 移行が失敗したことを示します。具体的なエラーはログ出力に含まれます。

計画の概要を表示

移行計画の概要を表示してから、終了します。

```
tabcmt-runner --info <plan file>
```

help

コマンドラインユーティリティの使用情報を表示します。

```
tabcmt-runner --help
```

version

現在のアプリケーションのバージョン情報を表示します。

```
tabcmt-runner --version
```

encryption

暗号化キーをリセットするか、新しいキーを指定します。Content Migration Tool UI で既に実行している場合でも、tabcmt-runner ユーティリティを使用する前に暗号化キーを指定する必要があります。

```
tabcmt-runner encryption <new_key> | --reset
```

改善

既定値: on

アプリケーションによる匿名での使用状況情報の収集を有効または無効にします。この情報は完全に匿名であり、定期的に Tableau へ送信されるため、Content Migration Tool の改善に役立ちます。

例

改善プログラムが有効か無効かを表示する場合:

```
tabcmt-runner improvement
```

改善プログラムを有効または無効にする場合:

```
tabcmt-runner improvement <on|off>
```

license

2022 年 7 月に非推奨になりました。

このコマンドは、レガシー ライセンスにのみ適用されます。現在のユーザーのレガシー アプリケーション ライセンスを管理します。レガシ キーを使用している場合に tabcmt-runner ユーティリティを使用するには、ライセンスが既に Content Migration Tool UI で認証済みであっても、このコマンドを使用してライセンスを認証する必要があります。

例

現在のライセンス情報を表示する場合:

```
tabcmt-runner license
```

シリアル キーまたはオフライン ライセンス キーを設定/ライセンス認証する場合:

```
tabcmt-runner license <key>
```

現在のライセンスを削除/認証解除する場合:

```
tabcmt-runner license --remove
```

ライセンス ファイルを使用して設定/ライセンス認証する場合:

```
tabcmt-runner license <file path> [--passphrase=<password>]
```

script-warning

既定値: on

移行スクリプトを含む移行計画を実行すると、警告メッセージが表示されます。

> **注:** このコマンドは、[設定] ページでの選択を更新します。詳細については、「Tableau Content Migration Tool の設定」を参照してください。

例

スクリプト警告がオンかオフかを表示します。

```
tabcmd-runner script-warning
```

スクリプト警告をオンまたはオフにします

```
tabcmd-runner script-warning <on|off>
```

オンになっている場合は、オプション --allow-scripts を含めて移行計画を実行する必要があります。

```
tabcmd-runner --allow-scripts <plan file>
```

実行可能なユーザー

コンソール ランナーを使用するには、次のものがすべて必要です。

- Content Migration Tool マシンでの管理者権限。
- Tableau サイトのユーザー アカウント(Explorer 以上の役割を持つ)。
- ソース サイトでのワークブックのダウンロード/コピーを保存パーミッション。
- 展開先サイトのパブリッシュ権限。

例: 移行計画のスクリプト作成

> **注:** このトピックには、ニーズと環境を満たす複数の移行計画に対するスクリプトを作成するための基礎として使用できるサンプル スクリプトが含まれています。このスクリプトは、サンプルとしてのみ使用され、実際にそのまま使用することはできません。コンソール ランナーの使用手順に関する詳細については、Tableau Content Migration Tool コンソール ランナーの使用を参照してください。

移行の実行に Tableau Content Migration Tool コマンド ライン ユーティリティを使用して、外部スケジューラ(Windows のタスク スケジューラなど) またはカスタム スクリプトから移行計画の実行を自動化できます。コンソール ランナーでは、.edt ファイルに保存された移行計画を一度に 1 つのみ実行します。グループとして実行する移行計画のグループがある場合は、Content Migration Tool コンソール ランナーと組み合わせたカスタム スクリプトを使用することができます。

次の例は PowerShell で記述されており、コンソール ランナーを使用して移行計画のリストをグループとして実行します。

次の例では、コードが以下の内容を示します。

- コンソール ランナーを使用して、グループとして複数の移行計画を実行する。
- グループ内の移行が 1 つでも失敗すると、計画グループの展開を直ちに停止する(オプション)。
- コンソール ランナーの終了コードを使用して、移行が失敗したか、警告がログに記録されたかを判断する。

```
# List of migration plans to execute as a group.
$planFiles = @(
       'customer 1.tcmx',
       'customer 2.tcmx'
```

```
)

# True of false whether to continue with the next plan if a
migration fails.
$continueOnFailure = $false

# Path to the CMT console runner executable
$runnerExe = 'C:\Program Files (x86)\Tableau\Tableau Content
Migration Tool\tabcmt-runner.exe'

# Store the exit code from the previously run migration plan.
$lastResult = -1

# Loop through and run each migration plan one at a time.
$planFiles | % {
       $file = $_

       if ($lastResult -ge 2 -and -not($continueOnFailure)) {
               Write-Warning "Skipping plan because previous migration failed.
`nSkipped plan: $file"
               return
       }

       Write-Verbose "Running migration plan: $file"
       & $runnerExe $file
       $lastResult = $LASTEXITCODE

       if ($lastResult -ge 2) {
               Write-Error "Migration failed. See output or log file for error
details.`nPlan: $file" -ErrorAction 'Continue'
       }
       elseif ($lastResult -eq 1) {
               Write-Warning "Migration completed with warnings. See output or
log file for warning details.`nPlan: $file"
       }
}
```

実行可能なユーザー

移行計画のスクリプトを作成するには、次のものがすべて必要です。

- Content Migration Tool マシンでの管理者権限。
- Tableau サイトのユーザー アカウント(Explorer 以上の役割を持つ)。
- ソース サイトでのワークブックのダウンロード/コピーを保存パーミッション。
- 展開先サイトのパブリッシュ権限。

Tableau Content Migration Tool コマンド ライン インターフェイスの使用

Tableau Content Migration Tool には、コマンド ライン インターフェイス、`tabcmt.cmd` がインストール フォルダーに含まれています。既定のインストール フォルダーは `%PROGRAMFILES%\Tableau\Tableau Content Migration Tool` (32 ビットの Windows) または `%PROGRAMFILES(x86)%\Tableau\Tableau Content Migration Tool`(64 ビットの Windows) です。

> **注:** `tabcmt.cmd` ユーティリティは、Content Migration Tool コンソール ランナーである `tabcmt-runner.exe` と同じではありません。コンソール ランナーは、コマンド ラインから移行を実行するために使用する、独立したコマンド ライン ユーティリティです。Content Migration Tool コンソール ランナーの使用については、Tableau Content Migration Tool コンソール ランナーの使用を参照してください。

`tabcmt` コマンド ラインで使用できるコマンドは、次のとおりです。

- migrate
- help
- update
- version

migrate

移行計画ファイルを GUI の移行ステップに開きます。

```
tabcmt migrate <plan file>
```

help

コマンド ライン インターフェイスと使用可能なコマンドに関する一般的なヘルプを示します。

例

使用可能なコマンドをすべて表示する場合:

```
tabcmt help
```

特定のコマンドのヘルプおよび使用状況情報を表示する場合:

```
tabcmt help <command>
```

license

2022 年 7 月に非推奨になりました

このコマンドは、レガシー ライセンスにのみ適用されます。現在のユーザーのアプリケーション ライセンスを管理します。

例

現在のライセンス情報を表示する場合:

```
tabcmt license
```

現在のライセンスを削除/認証解除する場合:

```
edt license remove
```

シリアル キーまたはオフライン ライセンス キーを設定/ライセンス認証する場合:

```
tabcmt license <key>
```

ライセンス ファイルを使用して設定/ライセンス認証する場合:

```
tabcmt license <file path> [--passphrase=<password>]
```

update

アプリケーション更新のオプションを管理します。

例

現在の更新設定を表示する場合:

```
tabcmt update
```

自動更新通知を有効または無効にする場合:

```
tabcmt update --disabled=<true|false>
```

URL を設定して更新情報を検出またはダウンロードする場合:

```
tabcmt update --url=<url>
```

ベータ版の更新プログラムの表示を有効または無効にします。安定したリリースの更新プログラムのみを表示するには、false に設定します。

```
tabcmt update --beta=<true|false>
```

version

現在のアプリケーションのバージョン情報を表示します。

```
tabcmt version
```

実行可能なユーザー

コマンドライン インターフェイスを使用するには、次のものがすべて必要です。

- Content Migration Tool マシンでの管理者権限。
- Tableau サイトのユーザー アカウント (Explorer 以上の役割を持つ)。
- ソース サイトでのワークブックのダウンロード/コピーを保存パーミッション。
- 展開先サイトのパブリッシュ権限。

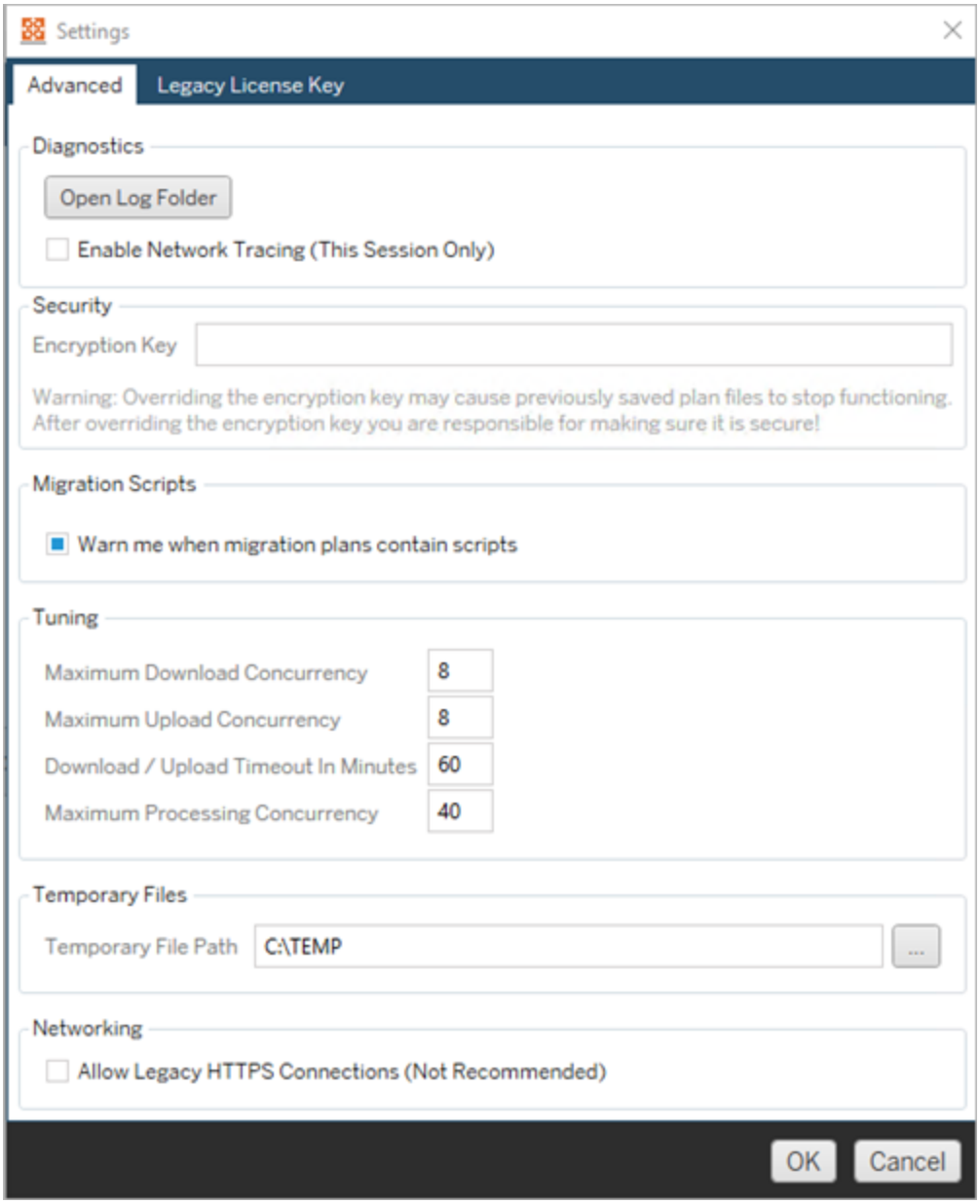
Tableau Content Migration Tool の設定

Tableau Content Migration Tool は既定の設定でほとんどの場合に動作しますが、必要がある場合や、Tableau サポートと協力しておりサポートから変更を行うよう依頼がある場合には、設定を変更することができます。

Content Migration Tool の設定を表示または更新するには、以下を実行します。

1. Content Migration Tool を開きます。

2. **Help > Settings**.をクリックします。[設定] ダイアログが開きます:



Diagnostics— [**Open Log Folder**] をクリックしてログがある場所を開きます。ここでは、ログを表示したり、ログを Tableau に送信する必要がある場合に zip ファイルに圧縮したりす

ることができます。詳細については、Tableau Content Migration Tool ログ ファイルを参照してください。

サポートと協力しており、ログにネットワーク トレースを含めるように依頼がある場合には **[Enable Network Tracing]** を選択します。このオプションをオフにするか、Content Migration Tool を再起動するまで適用されます。

Security— 暗号化キーがインストール時に自動的に生成されます。暗号化キーを変更すると、以前のキーで作成された埋め込みパスワードを含む移行計画は開くことができません。Tableau Content Migration Tool の複数のインストールがあり、移行計画を共有する場合には、ツールの各インスタンスが使用する暗号化キーが同じであることを確認する必要があります。

Migration Scripts— 既定では、移行スクリプトまたは実行可能ファイルを含む移行計画を実行すると、警告が表示されます。他のユーザーはこれらのファイルを編集できるため、移行を実行する前にファイルが安全であることを確認してください。この設定のオンとオフを切り替えると、コンソール ランナーの警告設定も更新されます。詳細については、「Tableau Content Migration Tool コンソール ランナーの使用」を参照してください。

Tuning— ほぼすべての場合、これらは既定に設定したままでかまいません。サポートと協力している場合、サポートよりこの設定を変更するように依頼される場合があります。

Temporary Files— 既定を変更する場合は、一時ファイルの場所を選択します。この場所は、移行時にコンテンツがコピーされる場所です。既定の場所に移行されたコンテンツを一時的に保持する十分な領域がない場合は、この場所を変更する必要があるかもしれません。

Networking— **[Allow Legacy HTTPS Connections]** を選択すると、以前の HTTPS 構成 (SSL v3 など) を使用して実行している Tableau Server インストールに接続できます。これは推奨されません。

実行可能なユーザー

通常、上記のタスクは、Content Migration Tool がインストールされているマシンで管理者アクセス権を持つユーザーのみが実行できます。

Tableau Content Migration Tool ログ ファイル

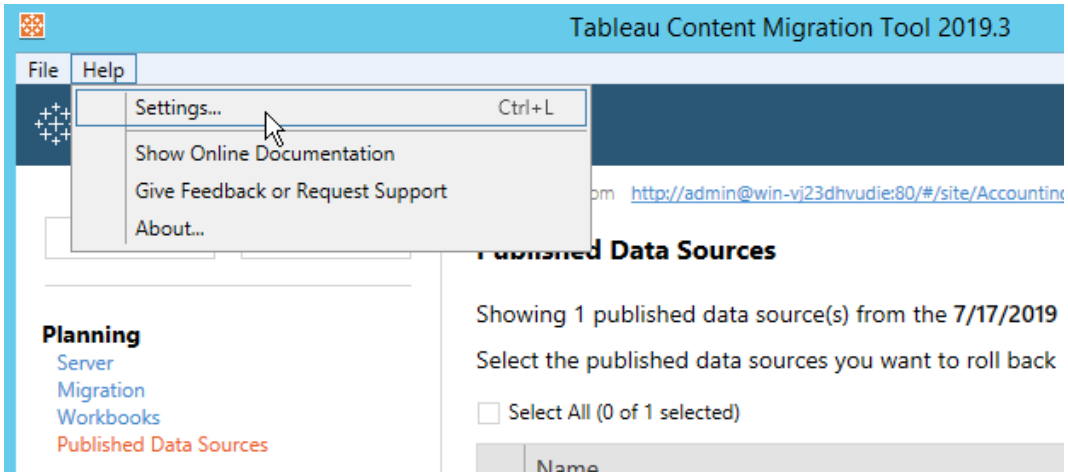
Tableau Content Migration Tool は移行の実行時にログ ファイルを生成します。これらのログ ファイルは問題のトラブルシューティングに役立つ場合があります。

> **注:** Content Migration Tool のすべての設定については、Tableau Content Migration Tool の設定を参照してください。

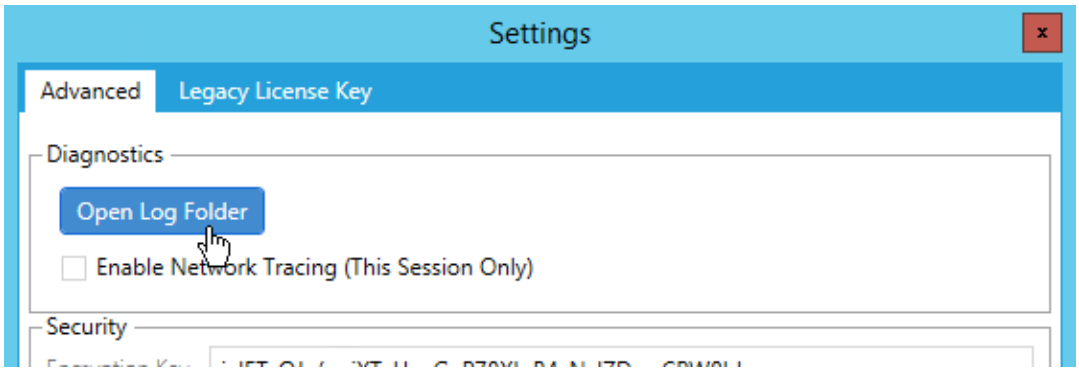
Content Migration Tool ログ ファイルの場所

Content Migration Tool から Content Migration Tool ログ ファイルを検索するには、以下を実行します。

1. Content Migration Tool を起動します。
2. [**Help**] および [**Settings**] をクリックします。



3. [**Settings**] ダイアログで [**Open Log Folder**] をクリックします。



ログ ファイルを含むウィンドウが開きます。

Tableau サポートと協力しており、サポートよりログ ファイルの送信を求められる場合は、ファイルの送信前に zip ファイルに圧縮してください。Tableau へのログ ファイルの送信の詳細については、「Tableau ナレッジ ベース」を参照してください。

実行可能なユーザー

通常、上記のタスクは、マシンで管理者アクセス権を持つユーザーのみが実行できます。

アクティビティ ログ

Tableau Server を Advanced Management でご利用の場合、自動的にファイルに記録されるアクティビティ ログを使用すると、詳しい分析や監査を行うことができます。

これらのログを使用すると、以下を行うことができます。

- **イベント データの表示と分析:** Tableau Server の詳細なイベント データにアクセスすることができ、コンプライアンス情報を収集したり、サイト上のアクティビティを追跡したりすることができます。すべてのイベントには、タイムスタンプと行為者の ID、および、関連がある場合、影響を与えたコンテンツの ID が含まれます。

- **パーミッション変更の監査:** コンテンツのパーミッションの変更を追跡および監査します。これには、グループへのユーザーの追加やグループからの削除、プロジェクト間でのコンテンツの移動、コンテンツのパーミッションの明示的な変更などが含まれます。これは、コンプライアンス目的で堅牢な制御を実装するために不可欠です。

- **管理ビューの補足:** 管理ビューが提供する情報を強化して、サイトのアクティビティと使用状況のメトリクスを追跡します。アクティビティログのデータを Splunk や Amazon EventBridge などの監視ツールと統合すると、ログのフィールドをクエリすることにより、ユーザーが実行した最後の 10 件のアクション、コンテンツに対して最後にイベントを実行したユーザー、コンテンツに対する最後のアクションは何であったかなどの質問に答えることができます。

アクティビティログのイベントへのアクセス

アクティビティログのイベントは、ローカル ハード ドライブの `vizportal` および `vizqlserver` フォルダーに保存されます。デフォルトでは、それらのフォルダーは `/var/opt/tableau/tableau_server/data/tabsvc/logs/` にあります。各ディレクトリには、ご利用の導入環境のイベントと操作を記録する `<process name>-cepp-canonical-events_*.log` という名前のファイルがあります。

アクティビティログのイベントの完全なリストについては、アクティビティログのサイト イベント タイプのリファレンスを参照してください。

アクティビティログを使用した権限の監査

権限の監査により、システム管理者は、どのユーザーが Tableau コンテンツへのアクセス制御を変更したかを監視できます。アクセス制御を変更するには 2 つの方法があります。*明示的な*変更 (プロジェクトやコンテンツ アイテムのパーミッション機能の変更) と、*効果的な*変更 (ユーザーのサイト ロールやグループ メンバーシップの変更、コンテンツの移動など) です。これらの変更はすべて記録されるため、管理者はセキュリティとアクセス制御が維持されていることを証明できます。

パーミッション ルールの評価方法の詳細については、「有効なパーミッション」を参照してください。

ログの形式

ユーザーまたはグループがコンテンツへアクセスできる権限を変更すると、すべてのアクションはログ エントリに記録されます。各ログ エントリは、さまざまな情報を表す特定のキーを使用して、JSON 形式で構造化されています。ログ エントリには、次の 2 つの部分が含まれます。

- メタデータ: アクションが発生した日時と場所、およびアクションを実行したユーザーに関する情報が含まれます。
- アクション: コンテンツのどの部分の権限が変更されたか、どの機能が変更されたか、その機能の値がどのように変更されたかに関する情報が含まれます。

> **注:** アクティビティログには、パーミッションのダイアログ UI および REST API を介して行われた変更が記録されます。API メソッドの詳細については、「パーミッション メソッド」を参照してください。

アクティビティログのエントリはフォーマットされておらず、キーはログ内で特定の順序でソートされていません。パーミッションを監査する場合、アクティビティログ データをの他のデータ ソースと組み合わせ、ID を名前にリンクして、イベントを解釈しやすくすることができます。

例

以下は、あるグループがデータ ソースへの接続を許可されたことを示すログ エントリの例です。


```
{
        "actorUserId": 39872,
        "actorUserLuid": "4e6b42bf-9040-4e60-b326-1c56a4fb96f8",
        "authorizableType": "DATASOURCE",
        "capabilityId": 32,
        "capabilityValue": "connect",
        "contentId": 2099835,
        "contentName": "Superstore ExtractNeal3",
        "eventTime": "2023-01-31T22:44:23.650058Z",
        "granteeId": 22,
        "granteeLuid": "dae0717a-d524-436d-b469-fadeaa22a5dd",
        "granteeType": "Group",
        "granteeValue": "GROUP_ALLOW",
        "initiatingUserId": 39872,
        "initiatingUserLuid": "4e6b42bf-9040-4e60-b326-1c56a4fb96f8",
        "isError": false,
        "eventCategory": "security",
        "eventType": "set_permissions",
```

```
        "eventVersion": "1.0",
        "permissionType": "explicit",
        "siteLuid": "b45e272d-10c7-49d5-9037-e53ce47dbf4e"
}
```

ログ エントリには、次のような、イベントに関する重要な情報が記録されます。

- eventType は、パーミッションの設定イベントが発生したことを示します
- permissionType は、パーミッションに対する明示的な変更を示します
- contentId は、変更されたコンテンツの ID を示します
- authorizableType は、コンテンツ タイプ (この場合はデータ ソース) を示します
- capabilityValue は、変更された機能を示します
- granteeId は、影響を受けた、権限の被付与者を示します
- actorUserId は、変更を行ったユーザーの ID を示します
- eventTime は、変更の日時を示します

イベント

ログ エントリには、コンテンツ所有者が変更された場合の content_owner_change、またはコンテンツに対する明示的なパーミッション ルールが削除されたときの delete_permissions など、権限変更に関するさまざまなイベント タイプが含まれます。イベント タイプ、属性、記録されるタイミングについての詳細は、「アクティビティ ログのサイト イベント タイプのリファレンス」を参照してください。

アクティビティ ログのサイト イベント タイプのリファレンス

このトピックでは、アクティビティ ログのサイト イベント タイプと属性について説明します。

イベント タイプの詳細

次のコンテンツでは、アクティビティ ログのサイト イベント タイプについて説明します。右側にあるアルファベット順に並べられたイベント タイプのリストを使用するか、**Ctrl/cmd-F** を使用して、目的の

キーワードに直接移動します。

> **注:** イベントのタイムスタンプは ISO 8601 UTC 形式で記録されます。

共通の属性

次の表には、アクティビティログのすべてのサイトイベントに共通の属性が含まれています。イベント固有の属性については、このトピックの個々のイベント表を確認してください。

属性名	タイプ	説明
actorUserId	整数	イベントを開始するアクションを実行したユーザーの ID
actorUserLuid	文字列	イベントを開始するアクションを実行したユーザーの LUID
eventOutcome	文字列	操作の最終結果を表します。表示される値は次のとおりです。 • **success:** 試行された操作が正常に完了しました。 • **unauthorized:** 操作を開始するユーザーに、実行権限が付与されていません。 • **client_error:** クライアント/呼び出し元に起因する理由によって、操作が失敗しました。例: 無効な API リクエストパラメーター、クライアント接続のタイムアウト、スロットリング • **internal_error:** 予期しない内部エラーによって、操作が失敗しました。
eventOutcomeReason	文字列	イベントの結果に関する追加の詳細を提供します
eventTime	文字列	イベントが発生したときのタイムスタンプ
initiatingUserId	整数	開始ユーザーの ID。擬装の場合、これは擬装を開始した管理ユーザーの ID です。標準ログインの場合、この値

		は userId と同じです。
initiatingUserLuid	文字列	開始ユーザーの LUID。擬装の場合、これは擬装を開始した管理ユーザーの LUID です。標準ログインの場合、この値は userLuid と同じです。
licensingRoleName	文字列	イベント発生時のユーザーのライセンス発行ロールの名前
serviceName	文字列	イベントを開始したサービスの名前 (vizportal、vizqlserver、sitesaml など)。
siteLuid	文字列	イベントが発生した Tableau サイトの LUID
siteRoleId	整数	ユーザーのサイトロールID。値 0 = SiteAdministratorExplorer、1 = SupportUser、2 = ExplorerCanPublish、3 = Explorer、7 = Guest、8 = Unlicensed、9 = Viewer、10 = Creator、11 = SiteAdministratorCreator。
systemAdminLevel	整数	ユーザーがシステム管理者であるかどうかを示します。値が 10 = サイト管理者、0 = サイト管理者でない

add_delete_user_to_group

add_delete_user_to_group イベントは、ユーザーがグループに追加、またはグループから削除されたときにログに記録されます。

属性名	タイプ	説明
groupId	整数	グループの ID
groupLuid	文字列	グループの LUID
groupOperation	文字列	グループ操作: ユーザーをグループに追加またはグループから削除する
isError	ブール	監査シナリオが正常に完了したか、またはエラーにより失敗し

	値	たかどうかを示します
userId	整数	ユーザーの ID
userLuid	文字列	ユーザーの LUID

content_owner_change

content_owner_change イベントは、コンテンツの所有者が変更されたときにログに記録されます。

属性名	タイプ	説明
contentId	整数	所有者が変更されたコンテンツの ID
contentLuid	文字列	所有者が変更されたコンテンツの LUID
contentName	文字列	所有者が変更されたコンテンツの名前
contentType	文字列	データソース、ワークブック、ビューなどのコンテンツ タイプ
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
newOwnerId	整数	新しいコンテンツ所有者の ID
newOwnerLuid	文字列	新しいコンテンツ所有者の LUID
oldOwnerId	整数	古いコンテンツ所有者の ID
oldOwnerLuid	文字列	古いコンテンツ所有者の LUID

create_delete_group

create_delete_group イベントは、グループが作成または削除されたときにログに記録されます。

属性名	タイプ	説明
groupDomain	文字列	ローカルなどのグループのドメイン

groupId	整数	グループの ID
groupLuid	文字列	グループの LUID
groupName	文字列	パーミッションが変更されたグループの名前
groupOperation	文字列	グループ操作: 作成または削除
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します

create_permissions

`create_permissions` イベントは、新しい明示的なパーミッション ルールが作成されたときにログに記録されます

注記: 2024 年 10 月に非推奨になりました。代わりに set_permissions を使用してください。

属性名	タイプ	説明
authorizableType	文字列	プロジェクトやワークブックなど、パーミッションが変更されたコンテンツのタイプ
capabilityId	整数	機能の ID機能とは、コンテンツに対して、表示、フィルター、ダウンロード、削除などのアクションを実行する機能のことです
capabilityValue	文字列	機能の説明
contentId	整数	パーミッションが更新されたコンテンツの ID
contentLuid	文字列	コンテンツ アイテムの LUID
contentName	文字列	パーミッションが更新されたコンテンツの名前

granteeId	整数	被付与者の ID
granteeLuid	文字列	被付与者の LUID
granteeType	文字列	被付与者のタイプ: ユーザーまたはグループ
granteeValue	文字列	'ユーザー許可' または 'グループ許可' など、更新されたパーミッションの値
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します

delete_all_permissions

`delete_all_permissions` イベントは、コンテンツに対する明示的なパーミッション ルールがすべて削除されたときにログに記録されます (通常はコンテンツが削除されたときに記録されます)。

属性名	タイプ	説明
authorizableType	文字列	プロジェクトやワークブックなど、パーミッションが変更されたコンテンツのタイプ
contentId	整数	パーミッションが更新されたコンテンツの ID
contentLuid	文字列	コンテンツの LUID
contentName	文字列	パーミッションが更新されたコンテンツの名前
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します

delete_permissions

`delete_permissions` イベントは、コンテンツの明示的なパーミッション ルールが削除されたときにログに記録されます。

属性名	タイプ	説明

authorizableType	文字列	プロジェクトやワークブックなど、パーミッションが変更されたコンテンツのタイプ
capabilityId	整数	機能の ID機能とは、コンテンツに対して、表示、フィルター、ダウンロード、削除などのアクションを実行する機能のことです
capabilityValue	文字列	機能の説明
contentId	整数	パーミッションが更新されたコンテンツの ID
contentLuid	文字列	コンテンツの LUID
contentName	文字列	パーミッションが更新されたコンテンツの名前
granteeId	整数	被付与者の ID
granteeLuid	文字列	被付与者の LUID
granteeType	文字列	被付与者のタイプ: ユーザーまたはグループ
granteeValue	文字列	'ユーザー許可' または 'グループ許可' など、更新されたパーミッションの値
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します

delete_permissions_grantee

`delete_permissions_grantee` イベントは、ユーザーに対する明示的なパーミッション ルールがすべて削除されたときにログに記録されます (通常はユーザーが削除されたときに記録されます)。

属性名	タイプ	説明

granteeId	整数	被付与者の ID
granteeLuid	文字列	被付与者の LUID
granteeType	文字列	被付与者のタイプ: ユーザーまたはグループ
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します

display_sheet_tabs

`display_sheet_tabs` イベントは、ワークブックの "タブ付きビュー"の値が更新されたときにログに記録されます。

属性名	タイプ	説明
displayTabs	ブール値	ワークブックのシートがタブとして表示されるかどうかを示します
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
workbookId	整数	ワークブックの ID

move_content

`move_content` イベントは、プロジェクト間でワークブックを移動させるなど、コンテンツを移動したときにログに記録されます。

属性名	タイプ	説明
contentId	整数	所有者が変更されたコンテンツの ID
contentLuid	文字列	所有者が変更されたコンテンツの LUID
contentName	文字列	所有者が変更されたコンテンツの名前

contentType	文字列	データソース、ワークブック、ビューなどのコンテンツ タイプ
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
newContainerLuid	文字列	新しいコンテナーの LUID
newContainerType	文字列	新しいコンテナー タイプ (プロジェクトなど)
oldContainerLuid	文字列	以前のコンテナーの LUID
oldContainerType	文字列	プロジェクトなどの以前のコンテナー タイプ

project_lock_unlock

`project_lock_unlock` イベントは、プロジェクトのパーミッションがロックされたとき、またはロックが解除されたときにログに記録されます。

属性名	タイプ	説明
controllingProjectLuid	文字列	ネストされたプロジェクトのパーミッションを制御するプロジェクトの LUID
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
projectLuid	文字列	プロジェクトの LUID
projectOperation	文字列	プロジェクト操作: ロックまたはロック解除

set_permissions

`set_permissions` イベントは、コンテンツ アイテムに対して明示的なパーミッション ルールが作成または更新されたときにログに記録されます。

属性名	タイプ	説明
authorizableType	文字列	パーミッションが変更されたコンテンツのタイプ (プロジェクトやワークブックなど)

capabilityId	整数	機能の ID機能とは、特定のコンテンツに対するあるアクション (表示、フィルター、ダウンロード、削除など) を実行する能力のことです。
capabilityValue	文字列	機能の説明
contentId	整数	パーミッションが設定されたコンテンツの ID
contentLuid	文字列	コンテンツ アイテムの LUID
contentName	文字列	パーミッションが設定されたコンテンツの名前
granteeId	整数	被付与者の ID
granteeLuid	文字列	被付与者の LUID
granteeType	文字列	被付与者のタイプ: ユーザーまたはグループ
granteeValue	文字列	'ユーザー許可' または 'グループ許可' など、設定されたパーミッションの値
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
permissionType	文字列	パーミッションのタイプ: 明示的または未指定

update_permissions

`update_permissions` イベントは、コンテンツ アイテムに対して明示的なパーミッション ルールが更新されたときにログに記録されます。

注記: 2024 年 10 月に非推奨になりました。代わりに set_permissions を使用してください。

属性名	タイプ	説明
authorizableType	文字列	プロジェクトやワークブックなど、パーミッションが変更されたコンテンツのタイプ
capabilityId	整数	機能のID機能とは、コンテンツに対して、表示、フィルター、ダウンロード、削除などのアクションを実行する機能のことです
capabilityValue	文字列	機能の説明
contentId	整数	パーミッションが更新されたコンテンツのID
contentLuid	文字列	コンテンツのLUID
contentName	文字列	パーミッションが更新されたコンテンツの名前
granteeId	整数	被付与者のID
granteeLuid	文字列	被付与者のLUID
granteeType	文字列	被付与者のタイプ: ユーザーまたはグループ
granteeValue	文字列	'ユーザー許可' または 'グループ許可' など、更新されたパーミッションの値
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
permissionType	文字列	パーミッションのタイプ: 明示的または未指定

update_permissions_template

`update_permissions_template` イベントは、プロジェクトのパーミッション テンプレートが更新されたときにログに記録されます。

属性名	タイプ	説明
authorizableType	文字列	プロジェクトやワークブックなど、パーミッションが変更されたコンテンツのタイプ
capabilityId	整数	機能のID機能とは、コンテンツに対して、表示、フィルター、ダウンロード、削除などのアクションを実行する機能のことです
capabilityValue	文字列	機能の説明
contentId	整数	パーミッションが更新されたコンテンツのID
contentLuid	文字列	コンテンツのLUID
contentName	文字列	パーミッションが更新されたコンテンツの名前
granteeId	整数	被付与者のID
granteeLuid	文字列	被付与者のLUID
granteeType	文字列	被付与者のタイプ: ユーザーまたはグループ
granteeValue	文字列	'ユーザー許可' または 'グループ許可' など、更新されたパーミッションの値
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
permissionType	文字列	パーミッションのタイプ: 明示的または未指定

templateType	文字列	ワークブックやデータソースなど、パーミッションを変更するために使用するパーミッション テンプレートのタイプ

user_create_delete

`user_create_delete` イベントは、ユーザーが作成または削除されたときにログに記録されます。

属性名	タイプ	説明
forUserName	文字列	アカウントの作成、更新、削除を行ったユーザーの名前
isError	ブール値	監査シナリオが正常に完了したか、またはエラーにより失敗したかどうかを示します
siteRole	文字列	ユーザーのサイト ロール。ユーザーがサイト上で持つことができる最大アクセス レベルを決定します
targetUserId	整数	アカウントの作成、更新、削除を行ったユーザー ID
targetUserLuid	文字列	アカウントの作成、更新、削除を行ったユーザー LUID
userOperation	文字列	作成、削除、サイト ロールの変更など、ユーザーに対して実行されるアクション

Tableau Server のキー管理システム

Tableau Server には、保存中の暗号化を有効にできるキー管理システム (KMS) のオプションが 3 つあります。1 つは、Tableau Server のすべてのインストールで使用できるローカル オプションです。その他の 2 つのオプションには Advanced Management の機能が必要ですが、別の KMS を使用できます。

重要: 2024 年 9 月 16 日以降、Advanced Management は独立したアドオン オプションとしては利用できなくなりました。Advanced Management 機能は、以前に Advanced Management を購入した場合、または特定のライセンス エディション (Tableau Enterprise

(Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

バージョン 2019.3 以降、Tableau Server では次の KMS オプションが追加されました。

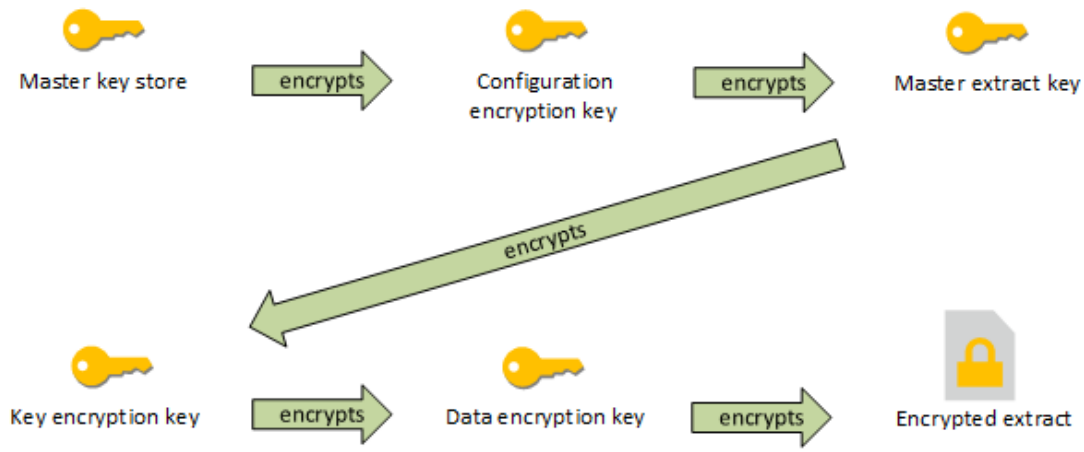
- すべてのインストールで使用できるローカル KMS。これについては以下で説明します。
- Advanced Management の一部として提供される AWS ベースの KMS詳細については、「AWS キー管理システム」を参照してください。

バージョン 2021.1 以降、Tableau Server では次の KMS オプションが追加されました。

- Advanced Management の一部として提供される Azure ベースの KMS詳細については、Azure Key Vaultを参照してください。

Tableau Server のローカル KMS

Tableau Server のローカル KMS は、サーバー シークレットの管理で説明されているシークレット保存機能を使用してマスター抽出キーの暗号化と保存を行います。このシナリオで、Java キーストアはキー階層のルートとして機能します。Java キーストアは Tableau Server と共にインストールされます。マスター キーへのアクセスは、オペレーティング システムのネイティブ ファイル システム認可メカニズムによって管理されます。既定の構成では、暗号化された抽出には Tableau Server のローカル KMS が使用されます。ローカル KMS と暗号化された抽出のキー階層は次のとおりです。



構成のトラブルシューティング

マルチノードの構成ミス

AWS KMS のマルチノードセットアップでは、クラスタ内の別のノードが正しく構成されていなくても、`tsm security kms status` コマンドから正常 (OK) ステータスが返される場合があります。KMS のステータスチェックでは、Tableau Server 管理コントローラープロセスが実行されているノードだけが報告され、クラスタ内の他のノードは報告されません。既定では、Tableau Server 管理コントローラープロセスは、クラスタ内の最初のノードで実行されます。

したがって、他のノードが誤って構成されているために Tableau Server が AWS CMK にアクセスできない場合、これらのノードではさまざまなサービスについてエラー状態が報告され、起動が失敗します。

KMS を AWS モードに設定した後で一部のサービスが起動に失敗する場合は、コマンド `tsm security kms set-mode local` を実行してローカルモードに戻します。

Tableau Server で RMK と MEK を再生成する

Tableau Server でルートマスターキーとマスター暗号化キーを再生成するには、`tsm security regenerate-internal-tokens` コマンドを実行します。

AWS キー管理システム

Tableau Server には、保存中の暗号化を有効にできるキー管理システム (KMS) のオプションが 3 つあります。このうちの 2 つは Advanced Management の機能を必要とし、残りの 1 つはローカルであり Tableau Server のすべてのインストールで使用できます。

> **重要:** 2024 年 9 月 16 日以降、Advanced Management は独立したアドオンオプションとしては利用できなくなりました。Advanced Management 機能は、以前に Advanced Management を購入した場合、または特定のライセンスエディション (Tableau Enterprise (Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

バージョン 2019.3 以降、Tableau Server では次の KMS オプションが追加されました。

- すべてのインストールで使用できるローカル KMS。詳細については、Tableau Server のキー管理システム」を参照してください。
- Advanced Management の一部として提供される AWS ベースの KMS。これについては以下で説明します。

バージョン 2021.1 以降、Tableau Server では次の KMS オプションが追加されました。

- Advanced Management の一部として提供される Azure ベースの KMS。詳細については、Azure Key Vaultを参照してください。

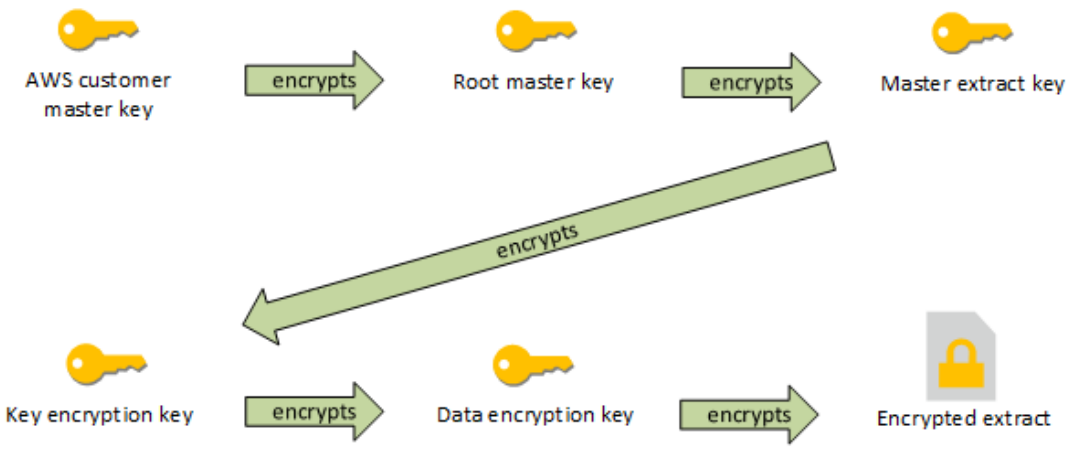
Tableau Server のバージョン 2019.3 のリリースでは、Advanced Management 一部として AWS キー管理システム (KMS) がサポートされます。

保管時の暗号化用に使用できるAWS KMS

> AWS KMS は、Tableau Server で Advanced Management の一部として利用できます。詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

組織で保存データ抽出の暗号化を展開している場合は、AWS を抽出暗号化用の KMS として使用するように Tableau Server を構成することもできます。AWS KMS を有効にするには、Tableau Server を AWS EC2 で展開する必要があります。AWS のシナリオでは、Tableau Server は AWS KMS カスタマー マスター キー (CMK) を使用して AWS データキーを生成します。Tableau Server は、AWS データキーを、暗号化されたすべての抽出のルート マスター キーとして使用します。ただし、AWS KMS 用に構成されている場合でも、Tableau Server 上にシークレットを安全に保存するためにネイティブの Java キーストアとローカル KMS が引き続き使用されます。AWS KMS は、暗号化された抽出のルート マスター キーを暗号化するためにのみ使用されます。

AWS を使用してマスター ルート キーを暗号化すると、マスター キーが抽出と同じパーミッションで保存されないため、セキュリティ特性が向上します。



AWS KMS を使用して Tableau Server が構成されている場合のキー階層

Tableau Server の暗号化された抽出用に AWS KMS を構成する

AWS カスタマー マスター キー (CMK) を使用して Tableau Server KMS 階層内のルート キーを暗号化するには、このセクションの説明のように Tableau Server を構成する必要があります。

開始する前に、次の要件を満たしていることを確認してください。

- Tableau Server は AWS EC2 で展開されている必要があります。
- Tableau Server は Advanced Management を使用して構成されている必要があります。Tableau Server 上の Tableau Advanced Management についてを参照してください。
- AWS Key Management Service で作成された顧客マスター キー (CMK) の管理権限を持っている必要があります。

ステップ 1: AWS で CMK を作成して Tableau Server 用のキー ポリシーを設定する

以下の手順は、AWS KMS サービスで実行します。参照情報は AWS のドキュメントに含まれています。

1. Tableau Server に使用する CMK を作成します。AWS のトピック「データ キーの作成」を参照してください。

2. サーバー インスタンスの IAM ロールを更新します。

   Tableau Server は、インスタンスの IAM ロールを使用して AWS KMS で認証を実行できる必要があります。このロールにはポリシーが付いている必要があります。そのポリシーにより、CMK に対して "GenerateDataKey" および "Decrypt" アクションを呼び出すためのインスタンス パーミッションを付与する必要があります。「Amazon EC2 の IAM ロール」を参照してください。

   Tableau Server のマルチノード展開では、サーバーの全ノードがこのポリシー (または同等のポリシー) の付いたロールの下で実行されている必要があります。同じロールをクラスタ内のすべてのノードに割り当てることができます。

3. 少なくとも CMK には、`Effect` が `Prinicpal` (サーバー インスタンスに関連付けられている IAM ロール) に `GenerateDataKey` および `Decrypt` の `Action` を `Allow` するように設定されているキー ポリシーが必要です。「AWS KMS でのキー ポリシーの使用」を参照してください。

ステップ 2: AWS 構成パラメーターを収集する

AWS KMS から完全な ARN 文字列を取得する必要があります。この文字列は、AWS KMS 管理ページの [General configuration (一般設定)] セクションにあります。ARN は arn:aws:kms:<region>:<account>:key/<CMK_ID> の形式で表示されます (たとえば `arn:aws:kms:us-west-2:867530990073:key/1abc23de-fg45-6hij-7k89-1l0mn1234567`)。

また、AWS の地域も指定する必要があります。これも ARN 文字列に含まれています。上記の例では、地域は `us-west-2` です。地域とは KMS インスタンスが存在する場所のことです。次のステップでは、「Amazon API Gateway」表の [Region (地域)] 列で示されるとおりに地域を指定する必要があります。

ステップ 3: AWS KMS 用に Tableau Server を構成する

Tableau Server で次のコマンドを実行します。このコマンドによりサーバーが再起動します。

- `tsm security kms set-mode aws --aws-region "<region>" --key-arn "arn:aws:kms:<region>:<account_number>:key/<CMK_ID>"`

`--key-arn` オプションは、AWS KMS 管理ページの [General configuration (一般設定)] セクション内の ARN から文字列を直接コピーします。

たとえば、AWS KMS インスタンスが us-west-2 地域で実行されており、アカウント番号が 867530990073、CMK キーが 1abc23de-fg45-6hij-7k89-1l0mn1234567 である場合、コマンドは次のようになります。

```
tsm security kms set-mode aws --aws-region "us-west-2" --key-
arn "arn:aws:kms:us-west-2:867530990073:key/1abc23de-fg45-6hij-
7k89-1l0mn1234567"
```

ステップ 4: 保存中の暗号化を有効にする

保存中の抽出の暗号化を参照してください。

ステップ 5: インストールを検証する

1. 次のコマンドを実行します。

   ```
   tsm security kms status
   ```

   返される可能性のある情報は次のとおりです。

   - カスタマー マスター キー (CMK) の ARN (ID)
   - CMK がある地域
   - 使用中のルートマスターキー (RMK) の ID。RMK は CMK によって暗号化されたキーです。Tableau Server は AWS KMS を呼び出すことによって CMK を復号化します。その後、RMK を使用してマスター抽出キー (MEK) を暗号化/復号化します。RMK は変更できますが、同時に 1 つしか存在できません。
   - KMS にはマスター抽出キー (MEK) のコレクションが保管されます。各 MEK は次の情報を持ちます。
     - ID (8ddd70df-be67-4dbf-9c35-1f0aa2421521 など)
     - "暗号化または復号化キー" または "復号化専用キー" のステータス。キーが "暗号化または復号化キー" である場合、Tableau Server はそれを使用して新しいデータを暗号化します。そうでない場合、キーは復号化にのみ使用されます。
     - 作成タイムスタンプ ("Created at:2019-05-29T23:46:54Z" など)。

- 暗号化または復号化キーへの最初の移行日時: キーが暗号化または復号化キーになった日時を示すタイムスタンプ。
- 復号化専用キーへの移行日時: キーが復号化専用に移行した日時を示すタイムスタンプ。

2. 抽出を暗号化して復号化した後でログを表示します。

   - 抽出をサイトにパブリッシュしてから暗号化します。保存中の抽出の暗号化を参照してください。

   - Tableau Desktop またはブラウザーで Web 作成を使用して抽出にアクセスします (このとき、抽出は使用のために復号化されます)。

   - vizqlserver_node ログ ファイルで `AwsKmsEncryptionEnvelopeAccessor` と `AwsKmsEncryptionEnvelope` を検索します。このログの既定の場所は `/var/opt/tableau/tableau_server/data/tabsvc/logs/` です。

     構成に成功したことを示すログ エントリの例を次に示します。

     - Decrypted the RMK with ID 1abc23de-fg45-6hij-7k89-1l0mn1234567 using the CMK with ARN arn:aws:kms:us-west-2:867530990073:key/1234567d-a6ba-451b-adf6-3179911b760f
     - Using RMK with ID 1abc23de-fg45-6hij-7k89-1l0mn1234567 to decrypt KMS store

     KMS に関連するパブリッシュと抽出更新については、バックグラウンダーのログを検索します。ログの詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

構成のトラブルシューティング

マルチノードの構成ミス

AWS KMS のマルチノード セットアップでは、クラスタ内の別のノードが正しく構成されていなくても、`tsm security kms status` コマンドから正常 (OK) ステータスが返される場合があります。KMS のステータス チェックでは、Tableau Server 管理コントローラー プロセスが実行されているノードだけが報告され、クラスタ内の他のノードは報告されません。既定では、Tableau Server 管理コントローラー プロセスは、クラスタ内の最初のノードで実行されます。

したがって、他のノードが誤って構成されているために Tableau Server が AWS CMK にアクセスできない場合、これらのノードではさまざまなサービスについてエラー状態が報告され、起動が失敗します。

KMS を AWS モードに設定した後で一部のサービスが起動に失敗する場合は、コマンド `tsm security kms set-mode local` を実行してローカル モードに戻します。

AWS CMK を更新する

AWS CMK の更新は AWS で実行するタスクです。既定では、AWS CMK は 1 年に 1 回更新されます。AWS のトピック「自動キー ローテーションの仕組み」を参照してください。ARN と地域が変更されないため、通常の CMK 更新シナリオでは Tableau Server で KMS 構成を更新する必要はありません。

AWS CMK が更新された後、Tableau Server で内部 RMK と MEK を再生成する必要があります。また、すべての抽出を新しい CMK で再暗号化する必要があります。

1. `tsm security regenerate-internal-tokens` コマンドを実行して、Tableau Server ですべての内部キーを再生成します。これには抽出の暗号化に使用される RMK と MEK が含まれます。
2. `tabcmd reencryptextracts <site-name>` を実行して、特定のサイトで抽出を再暗号化します。暗号化された抽出を保存するすべてのサイトでこのコマンドを実行してください。サイト上の暗号化された抽出の数によっては、この操作によってサーバーの処理負荷が大幅に大きくなる可能性もあります。業務時間外にこの操作を実行することを検討してください。保存中の抽出の暗号化を参照してください。

Tableau Server で RMK と MEK を再生成する

Tableau Server でルート マスター キーとマスター暗号化キーを再生成するには、`tsm security regenerate-internal-tokens` コマンドを実行します。

AWS KMS を使用してバックアップと復元を実行する

サーバーのバックアップは、AWS モードで追加の構成や手順を使用せずに作成できます。バックアップには、RMK と MEK の暗号化されたコピーが含まれます。キーを復号化するには、AWS CMK にアクセスして操作する必要があります。

復元シナリオの場合、復元先のサーバーは、ローカルを含むいずれかの KMS モードにすることができます。唯一の要件は、バックアップの復元先のサーバーが、バックアップで使用された CMK に対する復号化アクセス権を持っていることです。

復元の際は、バックアップから取り出された MEK が復号化専用キーとしてインポートされます。RMK は移行されません。インストール/復元プロセスの一部として新しい RMK が生成されます。

Azure Key Vault

Tableau Server には、保存中の暗号化を有効にできるキー管理システム (KMS) のオプションが 3 つあります。このうちの 2 つは Advanced Management の機能を必要とし、残りの 1 つはローカルでありTableau Server のすべてのインストールで使用できます。

> **重要:** 2024 年 9 月 16 日以降、Advanced Management は独立したアドオン オプションとしては利用できなくなりました。Advanced Management 機能は、以前に Advanced Management を購入した場合、または特定のライセンス エディション (Tableau Enterprise (Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

バージョン 2019.3 以降、Tableau Server では次の KMS オプションが追加されました。

- すべてのインストールで使用できるローカル KMS。詳細については、Tableau Server のキー管理システム」を参照してください。
- Advanced Management の一部として提供される AWS ベースの KMS。詳細については、「AWS キー管理システム」を参照してください。

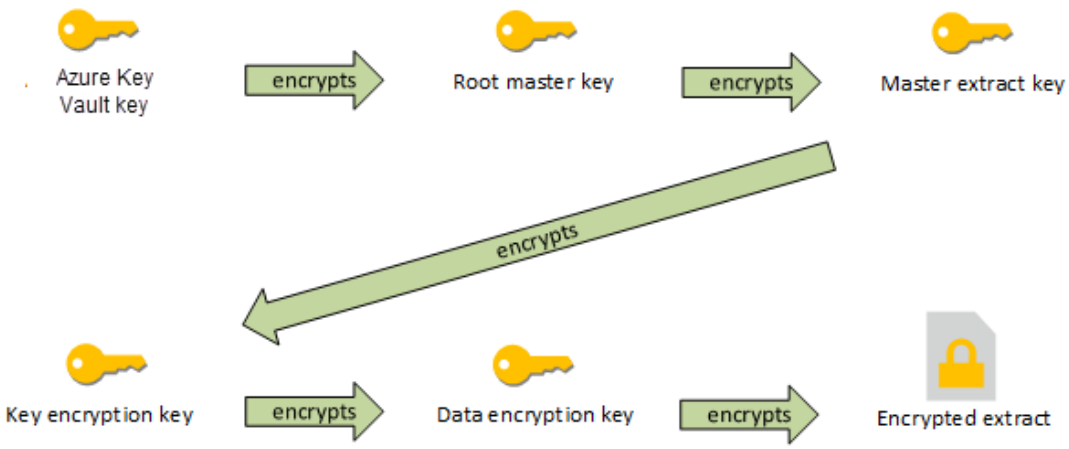
バージョン 2021.1 以降、Tableau Server では次の KMS オプションが追加されました。

- Advanced Management の一部として提供される Azure ベースの KMS。これについては以下で説明します。

保存中の暗号化用の Azure キー コンテナ

> Azure キー コンテナ は、バージョン 2021.1.0 以降の Tableau Server で Advanced Management の一部として利用できます。詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

組織で保存データ抽出の暗号化を展開している場合は、Azure キー コンテナ を抽出暗号化用の KMS として使用するように Tableau Server を構成することもできます。Azure キー コンテナを有効にするには、Azure に Tableau サーバーをデプロイする必要があります。Azure のシナリオでは、Tableau Server は Azure Key Vault を使用してルート マスター キー (RMK) を暗号化します。RMK は暗号化されたすべての抽出に使用します。ただし、Azure キー コンテナ 用に構成されている場合でも、Tableau Server ネイティブの Java キーストアとローカル KMS を引き続き使用して、Tableau Server 上にシークレットを安全に保存します。Azure キー コンテナ は、暗号化された抽出のルート マスター キーを暗号化するためにのみ使用します。



Azure キー コンテナ を使用して Tableau Server を構成する場合のキー階層

Tableau Server の暗号化された抽出用に Azure キー コンテナを構成する

Azure キー コンテナを使用して Tableau Server KMS 階層内のルート キーを暗号化するには、このセクションの説明のように Tableau Server を構成する必要があります。

開始する前に、次の要件を満たしていることを確認してください。

- Tableau Server は Azure にデプロイされている必要があります。
- Tableau Server は Advanced Management ライセンスを使用して構成されている必要があります。Tableau Server 上の Tableau Advanced Management についてを参照してください。
- キーが存在する Azure キー コンテナに対する管理権限が必要です。

ステップ 1: Azure で Tableau サーバーのキー コンテナとキーを作成する

以下の手順は、Azure キー コンテナ サービスで実行します。参照情報は Azure のドキュメントに含まれています。

1. Tableau Server に使用するキー コンテナを作成します。Azure のトピック「Key Vault を作成する」を参照してください。
2. コンテナにキーを作成します。Azure のトピック「キーとシークレットの管理」を参照してください。

   キーは非対称の RSA タイプである必要がありますが、任意のサイズにすることができます (Tableau Server はキーのサイズを考慮しません)。最大限のセキュリティを確保するには、最小権限の原則を適用することをお勧めします。

   Tableau では GET、UNWRAP KEY、および WRAP KEY コマンドを実行するためのパーミッションが必要であるため、最小権限の原則に従い、これらのコマンドを実行する場合にのみアクセスを許可することをお勧めします。Tableau Server を実行している VM にアクセス ポリシーを割り当てます。

   Tableau Server をマルチノードでデプロイしている場合は、アクセス ポリシーをサーバー クラスターのすべてのノードに割り当てる必要があります。

ステップ 2: Azure 構成パラメーターを収集する

キー コンテナ名とキー名を Azure から取得します。

ステップ 3: Azure キー コンテナ用に Tableau Server を構成する

Tableau Server で次のコマンドを実行します。このコマンドによりサーバーが再起動します。

- `tsm security kms set-mode azure --vault-name "<vault name>" --key-name "<key name>"`

  オプション `--vault-name` と `--key-name` は、Azure キー コンテナから文字列を直接コピーします。

  たとえば、Azure のキー コンテナの名前が `tabsrv-keyvault` で、キーの名前が `tabsrv-sandbox-key01` の場合、コマンドは次のようになります。

  ```
  tsm security kms set-mode azure --vault-name "tabsrv-keyvault" --key-name "tabsrv-sandbox-key01"
  ```

ステップ 4: 保存中の暗号化を有効にする

保存中の抽出の暗号化を参照してください。

ステップ 5: インストールを検証する

1. 次のコマンドを実行します。

   ```
   tsm security kms status
   ```

   返される可能性のある情報は次のとおりです。

   - ステータス: OK (キー コンテナが制御ノードからアクセス可能であることを示します)
   - モード: Azure キー コンテナ
   - コンテナ名: <key_vault_name>
   - Azure キー コンテナのキー名: <key_name>
   - アクティブなキーを示す、MEK で使用可能な UUID のリスト
   - KMS データにアクセスできない場合のエラー情報

2. 抽出を暗号化して復号化した後でログを表示します。

   - 抽出をサイトにパブリッシュしてから暗号化します。保存中の抽出の暗号化を参照してください。

   - Tableau Desktop またはブラウザーで Web 作成を使用して抽出にアクセスします (このとき、抽出は使用のために復号化されます)。

- vizqlserver_node ログ ファイルで `AzureKeyVaultEnvelopeAccessor` と `AzureKeyVaultEnvelope` を検索します。このログの既定の場所は `/var/opt/tableau/tableau_server/data/tabsvc/logs/` です。

  Azure キー コンテナに関連するパブリッシュと抽出更新については、バックグラウンダーのログを検索します。ログの詳細については、Tableau Server ログとログ ファイルの場所を参照してください。

構成のトラブルシューティング

マルチノードの構成ミス

Azure キー コンテナのマルチノード セットアップでは、クラスタ内の別のノードが正しく構成されていなくても、`tsm security kms status` コマンドから正常 (OK) ステータスが返される場合があります。KMS のステータス チェックでは、Tableau Server 管理コントローラー プロセスが実行されているノードだけが報告されます。クラスタ内の他のノードについては報告されません。既定では、Tableau Server 管理コントローラー プロセスは、クラスタ内の最初のノードで実行されます。

したがって、他のノードが誤って構成されているために Tableau Server が Azure キーにアクセスできない場合、これらのノードではさまざまなサービスについてエラー状態が報告され、起動が失敗します。

KMS を「azure」モードに設定した後で一部のサービスが起動に失敗する場合は、コマンド `tsm security kms set-mode local` を実行してローカル モードに戻します。

Azure キーの更新

Azure で Azure キーを更新します。必須またはスケジュールされたキー更新期間はありません。キーを更新するには、Azure で新しいキー バージョンを作成します。キー コンテナ名とキー名が変更されないため、通常の Azure キーの更新シナリオでは Tableau Server で KMS 構成を更新する必要はありません。

Azure キー コンテナを使用したバックアップと復元

サーバーのバックアップは、Azure キー コンテナ モードで追加の構成や手順を使用せずに作成できます。バックアップには、RMK と MEK の暗号化されたコピーが含まれます。キーを復号化するには、Azure キー コンテナ にアクセスして操作する必要があります。

復元シナリオの場合、復元先のサーバーは Azure キー コンテナ モードまたはローカル KMS モードのいずれかになります。唯一の要件は、バックアップの復元先のサーバーが、バックアップで使用された Azure キー コンテナ に対する復号化アクセス権を持っていることです。

Tableau Server 外部ファイル ストア

このトピックでは、Tableau Server 外部ファイル ストアの概要について説明します。

Tableau Server ファイル ストアには、抽出とワークブックのリビジョンが保存されます。通常、Tableau Server ファイル ストアは、Tableau Server にローカルにインストールされる組み込みの Tableau Server プロセスです。Tableau Server 2020.1 以降では、ファイル ストア データを外部ストレージに保存するように Tableau Server を構成できます。外部ストレージはネットワーク共有で、一元化されたディスク容量から複数のユーザーやさまざまなクライアント デバイスがデータを取得できる専用のファイル ストレージである必要があります。Windows の場合は サーバー メッセージ ブロック (SMB)、Linux インストールの場合はネットワーク ファイル システム (NFS) がこれに該当します。LAN (Local Area Network) 上のユーザーは標準のイーサネット接続を介して共有ストレージにアクセスします。

この新しい機能を使って、Tableau Server を次の 2 つの方法で構成できるようになりました。

- ファイル ストアをローカルにインストールします。つまり、ファイル ストアは Tableau Server ノードにインストールされます。
- 外部ファイル ストアを使用します (2020.1 以降)。

外部ファイル ストアを使用する理由

外部ストレージを使用すると、ファイル ストアをローカルにインストールする場合に次のような利点があります。

- **集中管理された場所:** ファイル ストアをローカルにインストールする場合、データを複数のファイル ストア ノードにレプリケートする必要があり、ネットワーク帯域幅が使われます。データの場所を一元化すると、Tableau クラスタ内の複数のノードでファイル ストアを実行したり、ノード間のレプリケーションを行う必要がなくなります。また、これにより、個々のノードで必要となるディスク空き容量が少なくなるほか、データが複数のノードにレプリケートされないため、ネットワーク帯域幅の使用率が少なくなる可能性があります。

- **バックアップ時間の改善:** スナップショットバックアップ技術が効果的です。Tableau Data のスナップショットバックアップを使用すると、Tableau のバックアップにかかる時間を大幅に短縮できます。

外部ファイルストアを管理する

ライセンス管理

外部ファイルストアを設定するには、Tableau Server の Advanced Management 機能を利用できる必要があります。詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。Advanced Management 機能を利用できない場合、次の動作が見られます。

- インストール中に外部ファイルストアで Tableau Server を構成しようとするとエラーメッセージが表示されますが、インストールは続行でき、Tableau Server ファイルストアはローカルにインストールされます。
- 既に外部ファイルストアを使用していて、ご利用のインストールで Advanced Management を利用できなくなっている場合は、次のような動作になります。
  - そのサーバーは再起動に失敗します。
  - バックアップは失敗します。
  - Advanced Management 機能を利用できなくなっているが、有効な Tableau Server ライセンスを持っている場合は、外部ファイルストアをローカルファイルストアに移行して、サーバーを再び稼働させることができます。外部リポジトリからローカルリポジトリへの移行方法の詳細については、ファイルストアを再構成するを参照してください。

サポートされる移行シナリオ

- Tableau Server にローカルにインストールされているファイルストアを外部管理ストレージ (ネットワークに接続されているストレージ) に移動します。
- ファイルストアを外部管理ストレージから Tableau Server に移動します。

バックアップと復元

外部ファイルストアを使用した Tableau Server 上のバックアップは、ファイルストアがローカルにインストールされている場合のバックアップの作成方法とは異なります。外部ファイルストアによる Tableau Server へのバックアップと復元を行う方法の詳細については、外部ファイルストアによるバックアップと復元を参照してください。

アップグレードに関する考慮事項

外部ファイル ストアを使用して構成された Tableau Server をアップグレードする場合、特別なステップは必要ありません。通常のアップグレードプロセスに従ってください。

高可用性に関する考慮事項

Tableau Server では、外部ファイル ストアの高可用性の管理やセットアップは行われません。お使いの管理ストレージには、冗長性と高可用性をサポートするソリューションが用意されている場合があります。

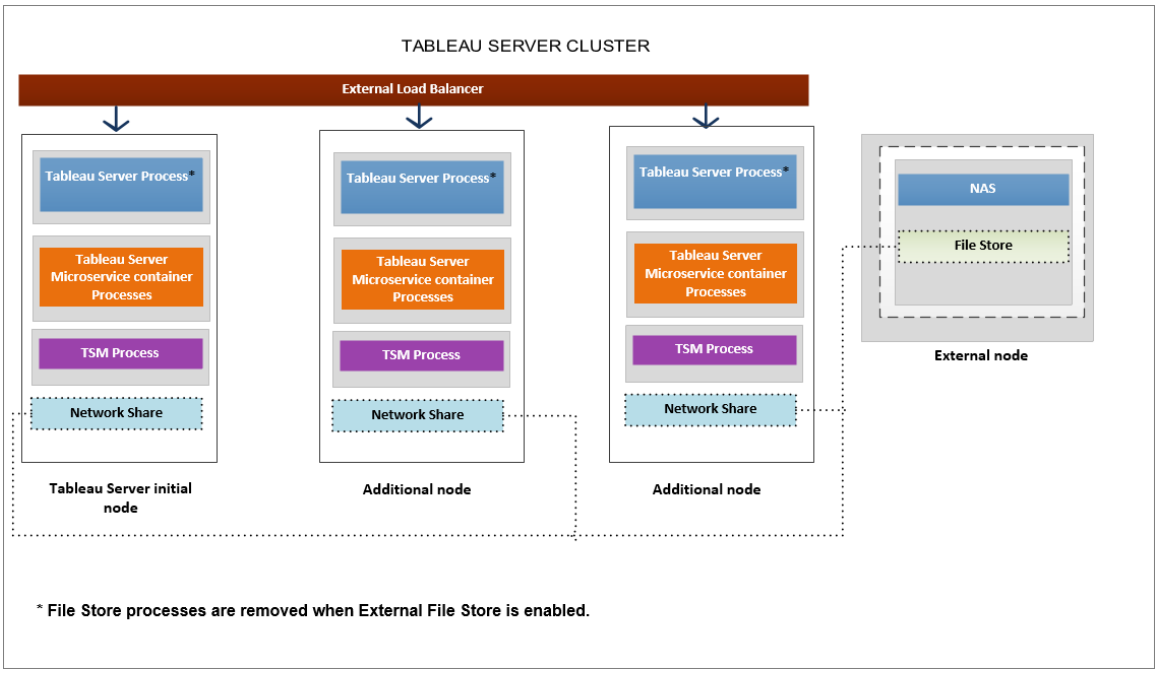
トポロジ

外部ファイル ストアを使用して Tableau Server を構成すると、ファイル ストアはローカルで実行されなくなります。[サーバー ステータス] ページでは、ファイル ストア プロセスが外部ノード上にあることが示されます。

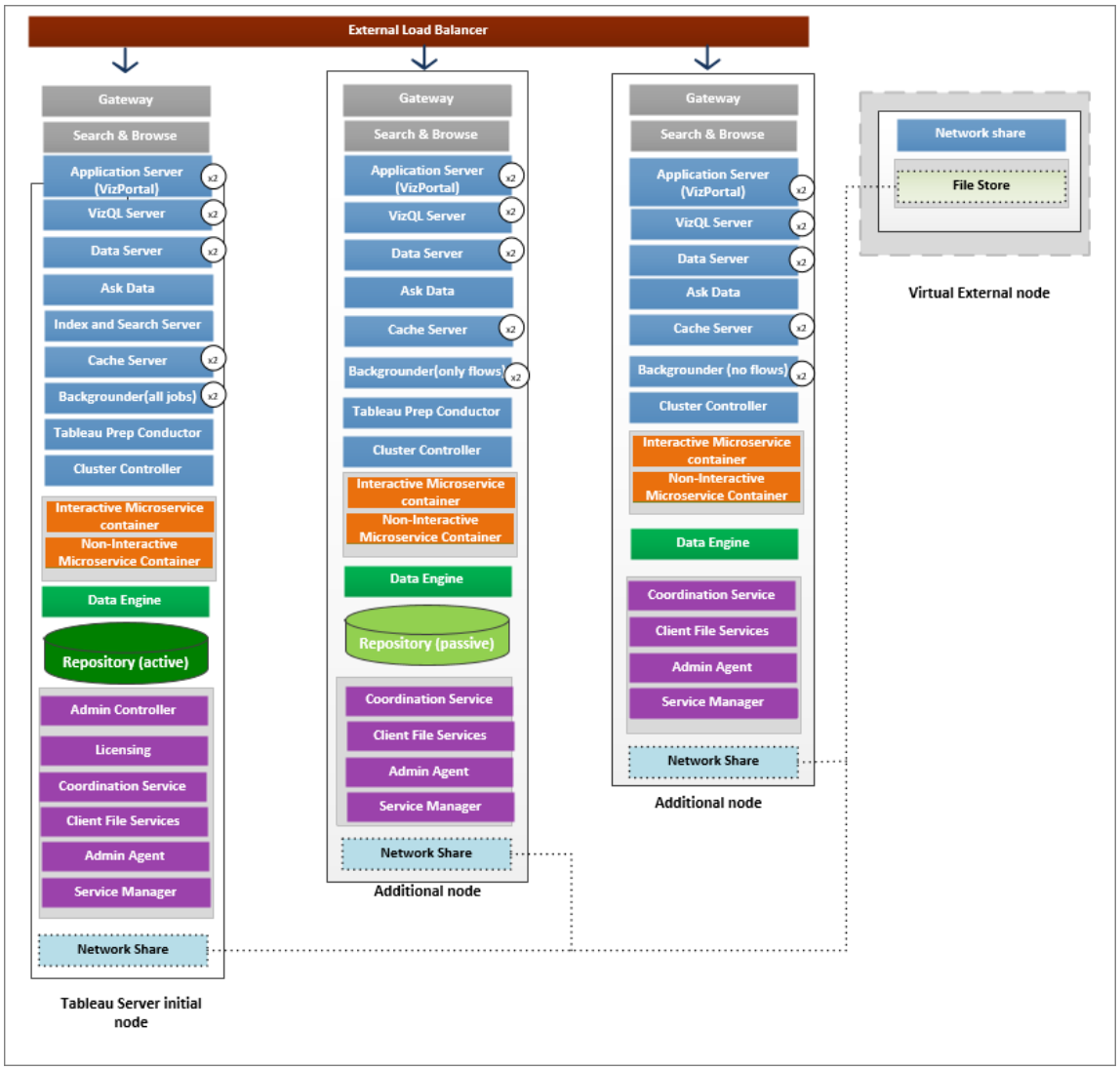
ファイル ストアが Tableau Server の外部で構成されている場合、データエンジンとファイル ストアが同じ場所に配置されることはありません。Tableau Server データエンジンで説明されているように、セットアップの際、データエンジンは引き続きファイル ストアを除く他のプロセスと共に自動的にインストールされます。ただし、Tableau Server を外部ファイル ストアで構成した場合は、他のプロセスが存在しない別のノードにデータエンジンをインストールできます。

ファイル ストアが外部で構成されている場合、データエンジンはネットワーク経由でストレージ システム上のファイル ストア データ(抽出) にアクセスします。システム全体が要件に合わせて動作するよう、ネットワークとストレージ システムに関するいくつかの事項を考慮する必要があります。詳細については、外部ファイル ストアのパフォーマンスに関する考慮事項を参照してください。

次の図は、外部ファイル ストアを使用した Tableau Server トポロジの概要バージョンです。



次の図は、外部ファイル ストアを使用した **Tableau Server** トポロジの詳細なバージョンであり、各ノードにインストールされているすべてのプロセスを示しています。



次へ

外部ファイル ストアを使用して Tableau Server をインストールする

外部ファイル ストアを使用して Tableau Server をインストールする

このトピックでは、ネットワーク共有を新しいインストール用の Tableau Server ファイル ストアとして構成するプロセスについて説明します。ファイル ストアがローカルで実行されている既存の Tableau Server インストールでこれを行う場合は、ファイル ストアを再構成するを参照してください。

前提条件

- Tableau Server 2020.1 以降を使用する必要があります。
- ストレージ オプションとして使用できるネットワーク共有が必要です。ストレージ ソリューションに関する推奨事項については、外部 ファイル ストアのパフォーマンスに関する考慮事項を参照してください。

  Linux インストールでは NFS を使用します。

  ストレージ サイズの見積もり: パブリッシュと抽出更新に必要なストレージの容量を考慮する必要があります。さらに、トピックオプション 2: リポジトリを個別にバックアップするで説明されているように、リポジトリのバックアップを個別に実行するオプションを特に選択しない限り、リポジトリのバックアップ サイズも考慮する必要があります。

  - 抽出:
    - Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。
    - 抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。
  - リポジトリのバックアップ:
    - リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。
    - リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、`workgroup.pg_dump` ファイルのサイズを確認します。
- ご利用のサーバー上で Advanced Management 機能が有効になっている必要があります。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

外部ファイル ストアを使用して Tableau Server をインストールする

ネットワーク共有を Tableau Server データを保存するためのファイル ストアとして構成して、Tableau Server をインストールできます。このソリューションを使用すると、ファイル ストア プロセスをローカルで実行する必要性がなくなります。このソリューションとそのメリットの詳細については、Tableau Server 外部ファイル ストアを参照してください。

インストール時に外部ファイル ストアを構成して Tableau Server をインストールするには、次の手順に従います。

ステップ 1: ネットワーク共有を構成する

ファイル サーバーで次の手順を実行します。

1. Tableau Server 外部ファイル ストアとして使用するディレクトリを作成して共有します。
2. すべての Tableau Server ノードから同じ場所のディレクトリとしてネットワーク共有にアクセスできることを確認します。

3. ネットワーク共有内で tableau ディレクトリを作成し、**tableau ユーザー**と**tableau グループ**にフル アクセス権を付与します。tableau ユーザーには、ネットワーク共有上のディレクトリに対する読み取りおよび書き込みパーミッションが必要です。ディレクトリ**'tableau'** を呼び出すことをお勧めします。

   ```
   /mnt/<network share>/tableau/
   ```

4. **ネットワーク共有が正しく構成されていることを検証する**: Tableau Server から、ネットワーク共有に書き込むコマンドを実行し、書き込み可能であることを確認します。

ステップ 2: TSM をダウンロードしてインストールする

1. 使用している Linux のディストリビューションに基づいて、適切なインストーラーをダウンロードします。
2. Tableau Server をインストールするコンピューターに sudo アクセス権を持つユーザーとしてログオンします。
3. .rpm または .deb のインストーラー パッケージをダウンロードします。
4. .rpm または .deb パッケージをコピーしたディレクトリに移動します。
5. パッケージ マネージャーを使用して、Tableau Server をインストールします。

- CentOS など**RHEL ライクな**ディストリビューションでは、既定以外の場所に Tableau Server をインストールするオプションがあります。
- 既定の場所 - 既定の場所 (/opt/tableau/tableau_server) にインストールするには、以下のコマンドを実行します。

  ```
  sudo yum update

  sudo yum install tableau-server-<version>.x86_64.rpm
  ```

- 既定以外の場所 - 既定以外の場所にインストールするには、rpm -i を使用する必要があります。また、すべての依存パッケージをインストールする必要があります。下記の注を参照してください。

  次のコマンドを実行します。

  ```
  sudo rpm -i--prefix/preferred/install/path tableau-server.rpm
  ```

  注: Tableau Server のインストールに yum を使用した場合、すべての依存パッケージが自動的にダウンロードされてインストールされます。これが、推奨される Tableau Server のインストール方法です。既定以外の場所にインストールする場合、または所属の組織で yum の使用が禁止されているため、rpm -i を使用しなくてはならない場合は、すべての依存パッケージを個別にインストールする必要があります。依存パッケージのインストールの詳細については、「Tableau Server を Linux を実行するエアギャップ環境にインストールする」を参照してください。

- **Ubuntu とDebian** では、次のコマンドを実行します。

  ```
  sudo apt-get update

  sudo apt-get -y install gdebi-core

  sudo gdebi -n tableau-server-<version>_amd64.deb
  ```

ステップ 3: TSM を初期化する

1. 次のスクリプトを実行して TSM を起動します。

   ```
   sudo ./initialize-tsm --accepteula --<optional_parameters>
   ```

   initialize-tsm スクリプトで必要なパラメーターは --accepteula だけです。このパラメーターは、Tableau Server のエンドユーザー ライセンス契約 (EULA) に同意するために含める必要があります。EULA は次の場所にあります。

   ```
   /opt/tableau/tableau_server/packages/docs.<version_
   code>/Commercial_EULA.txt
   ```

2. Tableau Server を構成する前に、ターミナルからログオフしてもう一度ログオンしてください。

   再びログオンした時点で新しいセッションが作成され、このときにグループのメンバーシップ変更が反映されます。この新しいセッションは、initialize-tsm スクリプトによって追加された環境変数へのアクセス権もあります。

   あるいは、次のコマンドを実行すると、現在のセッションのパスを更新することができます (グループのメンバーシップは更新されません)。

   ```
   source /etc/profile.d/tableau_server.sh
   ```

ステップ 4: Tableau Server をライセンス認証して登録する

ライセンス認証手順で Tableau Server のキーと Advanced Management のキーを使用します。次のコマンドを 2 回実行する必要があります。まずは Tableau Server のプロダクト キーと、次に Advanced Management プロダクト キーと共に実行します。

```
tsm licenses activate -k <product key>
```

ステップ 5: 外部ファイル ストアを有効にする

外部リポジトリを使用する Tableau Server の構成は、TSM CLI によってのみ実行できます。

1. 次の tsm コマンドを使用してネットワークストレージ機能を有効にします。

   ```
   tsm topology external-services storage enable --network-share
   /mnt/<network share name>/tableau
   ```

   セットアッププログラムにより、共有内で次のディレクトリ構造が自動的に作成されます。

   PostgreSQL データのバックアップ:

   tableau_data/tabsvc/repository_backup

   > **注:** このディレクトリはバックアップを初めて作成するときに作成されます。

   抽出とワークブックのリビジョン:

   tableau_data tabsvc/dataengine/extracts

   tableau_data/tabsvc/dataengine/revisions

ステップ 6: 最初のノードの設定を構成する

初期ノード設定の構成トピックの指示に従います。

ステップ 7: インストールを完了する

Tableau Server 用の初期管理アカウントを作成する必要があります。

- セットアップ中にローカル アイデンティティストアを構成した場合、使用する名前とパスワードを指定します。
- セットアップ中に LDAP または Active Directory のアイデンティティストアを構成した場合は、ディレクトリのメンバーであるユーザー アカウントを指定する必要があります。

最初のユーザーを作成するには、次の tabcmd コマンド実行します。

```
tabcmd initialuser --server localhost:80 --username '<new-admin-
username>'
```

このコマンドを実行した後、管理者パスワードを入力するようシェルから求められます。

ステップ 8: インストール後のタスク

Tableau Server 管理者アカウントの作成後は、インストール後のタスクで説明される構成手順に従って作業し、展開を続行します。

実行可能なユーザー

Tableau Server 管理者は、Tableau Server と外部ファイル ストアをインストールして構成できます。さらに、Tableau Server で使用するネットワーク共有を構成するためのパーミッションとアクセス権が必要です。

次へ

外部ファイル ストアによるバックアップと復元

ファイル ストアを再構成する

Tableau Server は、ローカルで実行されるファイル ストアまたは外部ファイル ストアを使用して構成できます。このトピックでは、既存の Tableau Server を再構成するのに必要な手順を説明します。

- 外部ファイル ストアを使用するように Tableau Server を再構成します。これにより、ファイル ストアがネットワーク共有に移動します。
- ファイル ストアをローカルで実行するように Tableau Server を再構成します。この場合、ファイル ストアが外部ストレージから Tableau Server に移動されます。
- 別のストレージを使用するように Tableau Server を構成します。たとえば、現在のネットワーク共有の寿命が来たため、新しいハードウェアが搭載されている新しいネットワーク共有を使用する必要がある場合などです。

外部ファイル ストアを使用して Tableau Server を再構成する

前提条件

- Tableau Server はバージョン 2020.1 以降である必要があります。
- ネットワーク共有を外部ストレージとして使用する必要があります。

  Linux インストールでは NFS を使用します。

ストレージとネットワークに関する考慮事項については、外部ファイル ストアのパフォーマンスに関する考慮事項を参照してください。

ストレージ サイズの見積もり: パブリッシュと抽出更新に必要なストレージの容量を考慮する必要があります。さらに、トピックオプション 2: リポジトリを個別にバックアップするで説明されているように、リポジトリのバックアップを個別に実行するオプションを特に選択しない限り、リポジトリのバックアップ サイズも考慮する必要があります。

- 抽出:
  - Tableau Server にパブリッシュする抽出の数、および各抽出のサイズを考慮に入れます。いくつかの抽出を Tableau Server にパブリッシュし、使用されたディスク容量を確認することで、必要なサイズを調べることができます。このディスク容量を確認すると、今後 Tableau Server にパブリッシュする抽出の数や、既存の各抽出のサイズが増える程度を把握できます。
  - 抽出更新の際に一時ディレクトリで必要となる容量を考慮に入れます。一時ディレクトリは、更新時に抽出を保存する場所です。最終的な抽出ファイルのサイズに対して、最大 3 倍の容量が必要となる場合があります。
- リポジトリのバックアップ:
  - リポジトリデータのサイズは、`<data directory>/pgsql/data/base` ディレクトリのサイズを確認することで見積もることができます。
  - リポジトリデータの正確なサイズを取得するには、バックアップ ファイルを開き、`workgroup.pg_dump` ファイルのサイズを確認します。
- Tableau Server で、有効な Advanced Management ライセンスが認証済みである必要があります。

ステップ 1: Tableau Server をアップグレードする

Tableau Server を 2020.1 以降にアップグレードします。「2018.1 以降からのアップグレード (Linux)」を参照してください。Tableau Server が既にバージョン 2020.1 以降である場合は、このステップをスキップできます。

ステップ 2: Advanced Management ライセンスを認証する

1. サーバー ライセンスの表示、Advanced Management ライセンスが Tableau Server で認証済みであることを確認します。

2. Advanced Management が Tableau Server にインストールされていない場合は、次の tsm コマンドを使用してライセンスを認証します。Advanced Management のキーをプロダクトキーとして指定してください。

   ```
   tsm licenses activate -k <product key>
   ```

ステップ 3: 外部ストレージを使用するファイル ストアを構成する

アップグレードを完了し、ライセンスを確認したら、外部ファイル ストアを使用して Tableau Server を構成します。これにより、ローカル ファイル ストアにある既存のデータが、選択した外部ストレージに移動されます。

1. ネットワーク共有を構成します。ファイル サーバーで次の手順を実行します。
   - ファイルをホストするディレクトリを作成して共有します。
   - すべての Tableau Server ノードで、ネットワーク共有をマッピングするか、UNC パスを使用します。

2. ネットワーク共有内で tableau ディレクトリを作成し、**tableau ユーザー**と**tableau グループ**にフル アクセス権を付与します。tableau ユーザーには、ネットワーク共有上のディレクトリに対する読み取りおよび書き込みパーミッションが必要です。ディレクトリ**tableau** を呼び出すことをお勧めします。

3. 次の tsm コマンドを使用してネットワーク共有機能を有効にします。

   ```
   tsm topology external-services storage enable --network-share
   /mnt/<network share name>/tableau
   ```

   次のディレクトリ構造が自動的に作成され、ローカル ファイル ストアにあるデータが外部ストレージに移動されます。このプロセスの間、ローカル ファイル ストアは自動的にコミッション解除されます。

   抽出とワークブックのリビジョン:

tableau_data tabsvc/dataengine/extracts

tableau_data/tabsvc/dataengine/revisions

PostgreSQL データのバックアップ:

tableau_data/tabsvc/repository_backup

> **注:** このディレクトリはバックアップを初めて作成するときに作成されます。

ローカル ファイル ストアを使用するように Tableau Server を再構成する

1. 次のコマンドを実行して、Tableau Server を停止します。

   ```
   tsm stop
   ```

2. 次のコマンドを実行すると、ファイル ストア データが外部ストレージから Tableau Server に移動されます。

   ```
   tsm topology external-services storage disable -fsn <node1,
   node2>
   ```

3. 次のコマンドを実行して、Tableau Server を起動します。

   ```
   tsm start
   ```

Tableau Server クラスタの場合は、ファイル ストアをインストールするノードを指定してください。データはコマンドで指定した最初のノードにコピーされてから他のノードにレプリケートされます。

> **注:** ファイル ストアを外部からローカルに移動する場合は、データエンジン プロセスが単独で別のノードにインストールされ、ファイル ストア、アプリケーション サーバー (VizPortal)、VizQL サーバー、データ サーバー、およびバックグラウンダーなどのコア サービスのいずれかと共にインストールされていることを確認します。
>
> 通常、Tableau Server データエンジンのインストールは自動的に実行され、1 つまたは複数の

コア サービスを持つノードにインストールされます。ただし、外部ストレージを使用するよう Tableau Server を構成する場合は、データエンジンをコアプロセスと同じ場所に配置しなくても、手動でノードにインストールできます。

現在、別のノードにデータエンジンプロセスがインストールされている場合は、disable コマンドを実行する前に、そのノードにファイル ストアをインストールするか、そのノードからデータエンジンを削除するかを選択できます。現在データエンジンがインストールされていないノードにファイル ストアをインストールすると、データエンジンが自動的に追加されます。

disable コマンドを実行するときにデータエンジンのみのノードがある場合は、エラーが発生します。

別の外部ストレージを使用するように Tableau Server を構成する

1. 新しいネットワーク共有を構成します。ファイル サーバーで次の手順を実行します。
    1. ファイルをホストするディレクトリを作成して共有します。
    2. すべての Tableau Server ノードで、ネットワーク共有をマッピングするか、UNC パスを使用します。
2. ネットワーク共有内で tableau ディレクトリを作成し、**tableau ユーザー**と**tableau グループ**にフル アクセス権を付与します。tableau ユーザーには、ネットワーク共有上のディレクトリに対する読み取りおよび書き込みパーミッションが必要です。ディレクトリ**tableau** を呼び出すことをお勧めします。
3. 次のコマンドを実行して、Tableau Server を停止します。

    ```
    tsm stop
    ```

4. 次のコマンドを実行して、新しいネットワーク共有を使用するように Tableau Server を構成します。

```
tsm topology external-services storage switch-share --network-
share /mnt/<newshare>/tableau
```

5. 次のコマンドを実行して、Tableau Server を起動します。

```
tsm start
```

実行可能なユーザー

Tableau Server 管理者は、ファイル ストアの場所を移動できます。さらに、外部ファイル ストアに使用する外部ストレージへのアクセス権が必要です。

外部ファイル ストアによるバックアップと復元

Tableau Server で外部ファイル ストアを有効にしている場合、`tsm maintenance backup` コマンドを使用して Tableau Server リポジトリとファイル ストア データをバックアップすることはできません。代わりに、"スナップショット" のバックアップ プロセスを使用して、ある時点でのネットワーク共有のスナップショットを作成します。

- 外部ファイル ストアを使用して構成された Tableau Server
- 外部ファイル ストアと外部リポジトリを使用して構成された Tableau Server

バックアップ戦略:

使用するバックアップ戦略は、復旧計画によって異なります。スナップショット バックアップ プロセスは、ファイル ストア (および要求された場合はリポジトリデータ) のバックアップを作成するだけであるため、それだけでは不十分な場合があります。また、完全復旧を行うためには、他の構成や設定が必要になることもあります。

重要: アップグレードを実行する前に、スケジュールされたタスクを無効にすることをお勧めします。これにはデータ コンテンツのすべての更新も含まれ、アップグレード前のバックアップを作成する前に実行する必要があります。これには、REST API ベースの抽出更新や tabcmd の使

用によって開始されるジョブなど、Tableau Server の外部でトリガーされるジョブの無効化が含まれる場合があります。

スナップショット バックアップ プロセスが十分な場合と不十分な場合を示すシナリオをいくつか紹介します。

- **スタンバイ Tableau Server —** 本番環境のサーバーがダウンした場合に使用するスタンバイ Tableau Server を維持する場合は、スナップショット バックアップを作成し、それを定期的にスタンバイ サーバーに復元するだけで十分な場合があります。バックアップ スケジュールは、目標復旧時点に基づいて設定する必要があります。
- **新しい Tableau Server、既存の構成は不要 —** 障害発生時に新しい Tableau Server インストールを使用する予定であるが、既存の Tableau Server インストールの構成と設定を必ずしも使用する必要がない場合は、Tableau Server の新しいインスタンスをインストールし、スナップショットを使用してデータを復元することができます。
- **新しい Tableau Server、既存の構成が必要 —** 既存の構成と設定、およびバックアップ日を含む新しい Tableau Server インストールを使用する予定の場合は、スナップショット バックアップと共に追加のファイルが必要です。すべての構成と設定を含む完全バックアップを実行するには、次の手順に従います。

    1. トポロジと構成データをエクスポートします。これにより、Tableau Server のほとんどの構成とトポロジがエクスポートされます。詳細については、「Tableau Server の完全なバックアップと復元を実行します」を参照してください。
    2. このトピックの「外部ファイル ストアによるバックアップと復元」セクションの説明に従って、ファイルストア (および必要に応じてリポジトリデータ) のネットワーク共有のスナップショットを作成します。
    3. エクスポートに含まれていない設定をメモします。これらには、システム ユーザー アカウント、調整サービスの展開構成、およびカスタマイズされた設定の値が含まれます。詳細については、Tableau Server の完全なバックアップと復元を実行しますを参照してください。

外部ファイル ストアを使用して構成された Tableau Server

外部ファイル ストアを使用して Tableau Server を構成している場合、バックアップ プロセスには、外部ファイル ストアを使用したネットワーク共有のポイントインタイム スナップショット バックアップの

作成を含める必要があります。次の手順に従ってバックアップを作成します。

> **注:** 外部ファイル ストアと外部リポジトリの両方がある場合は、「外部ファイル ストアと外部リポジトリを使用して構成された Tableau Server」を参照してください。

スナップショット バックアップの作成

次のステップに従って、スナップショット バックアップを作成します。

1. スナップショット バックアップを準備する。

   以下のコマンドを実行して、リポジトリのバックアップ ファイルを作成し、それをネットワーク共有に一時的にコピーします。Tableau Server は、バックアップ処理中も正常に動作し続けます。一貫性のあるスナップショットを作成するために、使用されていない抽出を削除する内部プロセスが一時停止されます。このプロセスは、後のステップで説明されているバックアップ プロセスが完了すると、再開されます。

   ```
   tsm maintenance snapshot-backup prepare
   ```

   準備のステップが完了すると、**スナップショット バックアップの準備に成功しました**というメッセージが表示されます。

   ネットワーク共有にリポジトリのバックアップ ファイルが作成されたことを確認します。

   > **注:** 準備のステップには、リポジトリと KMS キーおよびアセット キーのバックアップ ファイルの作成が含まれます。暗号化キー ソリューションにクラウド ソリューションを使用している場合は、復元に必要なキーの復号に使用する CMK にアクセスできる必要があります。AWS での暗号化キー ソリューションの詳細については、「AWS キー管理システム」を参照してください。Azure での暗号化キー ソリューションの詳細については、「Azure Key Vault」を参照してください。

2. ネットワーク共有のスナップショットを作成する。

適切なプロセスを使用して、ネットワーク共有のスナップショットを作成します。スナップショットは、特定の時点で取得された共有の読み取り専用バージョンです。これには、ファイル ストア データと、前のステップで共有に一時的にコピーされたリポジトリ バックアップ ファイルが含まれます。このネットワーク共有のスナップショットを作成するステップは、ネットワークによって異なります。詳細については、ご使用のネットワークのドキュメントを参照してください。

3. スナップショット バックアップ プロセスを完了します。

   以下のコマンドを実行して、バックアップ プロセスを完了し、一時停止していた内部処理を再開します。これにより、ネットワーク共有にコピーされたリポジトリ バックアップ ファイルが削除されます。

   ```
   tsm maintenance snapshot-backup complete
   ```

   ネットワーク共有からリポジトリの一時的なバックアップ ファイルが削除されたことを確認します。

スナップショット バックアップの復元

これらのステップは、シングル ノードとマルチ ノードの両方の Tableau Server インストールに適用されます。

1. Tableau Server を停止します。

   コマンド プロンプトで、次のコマンドを実行します。

   ```
   tsm stop
   ```

2. ネットワーク共有のスナップショットをネットワークに復元して、ファイル ストア データを復元します。具体的なステップは、ネットワークによって異なります。

3. リポジトリ データを復元します。

   次の復元コマンドを使用して、リポジトリ データを復元します。

   ```
   tsm maintenance snapshot-backup restore
   ```

> **注:** 暗号化キー ソリューションにクラウド ソリューションを使用している場合は、バックアップを復元するサーバーに、CMK が展開されているクラウド インスタンスを復号する権限が付与されていることを確認してください。

4. Tableau Server を再起動します。

   コマンド プロンプトで、次のコマンドを実行します。

   ```
   tsm start
   ```

外部ファイル ストアと外部リポジトリを使用して構成された Tableau Server

外部ファイル ストアと外部リポジトリの両方を使用して Tableau Server を構成している場合は、バックアップを作成するために必要な特別なステップがあります。これには、外部ファイル ストアを使用したネットワーク共有のポイントインタイム スナップショット バックアップの作成が含まれます。また、外部リポジトリの個別のバックアップが含まれる場合もあります。次の手順に従ってバックアップを作成します。

> **注:** 外部ファイル ストアを持っているが、既定のリポジトリを使用している場合は、「外部ファイル ストアを使用して構成された Tableau Server」を参照してください。

リポジトリのバックアップ

外部ファイル ストアと外部リポジトリの両方がある場合、リポジトリデータのバックアップには 2 つのオプションがあります。各オプションを選択する理由は次のとおりです。

- ネットワーク共有スナップショットにリポジトリのバックアップを含める。

  管理がしやすい。リポジトリを個別にバックアップする必要がなく、バックアップはファイル ストア データと同期されます。

- リポジトリを個別にバックアップする。

インスタンスのスナップショットのバックアップを実行できるクラウドソリューションを使用している場合は、外部リポジトリのバックアップを個別に実行する方が高速になります。バックアップファイルのサイズは、ネットワーク共有のスナップショットの準備にかかる時間に大きな影響を与える可能性があります。

オプション 1: ネットワーク共有のスナップショットにリポジトリのバックアップを含める

スナップショットバックアップの作成

1. スナップショットバックアップを準備する。

   以下のコマンドを実行して、リポジトリのバックアップファイルを作成し、それをネットワーク共有に一時的にコピーします。Tableau Server は、バックアップ処理中も正常に動作し続けます。一貫性のあるスナップショットを作成するために、使用されていない抽出を削除する内部プロセスが一時停止されます。このプロセスは、後のステップで説明されているバックアッププロセスが完了すると、再開されます。

   ```
   tsm maintenance snapshot-backup prepare --include-pg-backup
   ```

   > **注:** 2021.1 以降、外部ファイルストアと外部リポジトリの両方が有効になっている場合、リポジトリのバックアップを作成するには `--include-pg-backup` オプションを使用する*必要があります*。2021.1 より前のバージョンからアップグレードする場合で、バックアップを実行またはスケジュールするスクリプトがあり、引き続きリポジトリのバックアップを含める場合は、上に示すように準備コマンドに **`--include-pg-backup`** オプションを追加します。2021.1 より前のバージョンでは、オプションは必要ありません。リポジトリのバックアップは自動的に含まれます。

   準備のステップが完了すると、**スナップショット バックアップの準備に成功しました**というメッセージが表示されます。

   ネットワーク共有にリポジトリのバックアップファイルが作成されたことを確認します。

> **注:** 準備のステップでは、リポジトリとKMS キーおよびアセットキーのバックアップが作成されます。暗号化キーソリューションにクラウドソリューションを使用している場合は、復元に必要なキーの復号に使用するCMK にアクセスできる必要があります。AWS での暗号化キーソリューションの詳細については、「AWS キー管理システム」を参照してください。Azure での暗号化キーソリューションの詳細については、「Azure Key Vault」を参照してください。

2. ネットワーク共有のスナップショットを作成する。

   適切なプロセスを使用して、ネットワーク共有のスナップショットを作成します。スナップショットは、特定の時点で取得された共有の読み取り専用バージョンです。これには、ファイル ストア データと、前のステップで共有に一時的にコピーされたリポジトリ バックアップ ファイルが含まれます。このネットワーク共有のスナップショットを作成するステップは、ネットワークによって異なります。詳細については、ご使用のネットワークのドキュメントを参照してください。

   > **重要:** ネットワーク共有とリポジトリバックアップのスナップショットは、準備のステップを完了してから3 時間以内に完了する必要があります (ステップ 1)。これは、ファイル ストアとリポジトリのバックアップが同期され、復元が正し動作するようにするためです。

3. スナップショット バックアップ プロセスを完了します。

   以下のコマンドを実行して、バックアップ プロセスを完了し、一時停止していた内部処理を再開します。また、ネットワーク共有にコピーされたリポジトリ バックアップ ファイルが削除されます。

   ```
   tsm maintenance snapshot-backup complete
   ```

スナップショット バックアップの復元

これらのステップは、シングル ノードとマルチノードの両方の Tableau Server インストールに適用されます。

重要: ブルー/グリーン アップグレードを実行する場合、または tsm メンテナンス (バックアップと復元) メソッドを使用して Tableau Server 2021.4 (またはそれ以前) を手動でアップグレードする場合は、Tableau Server 2022.1 (またはそれ以降) に復元する前に `legacy-identity-mode` を有効にする必要があります。詳細については、「アイデンティティの移行に関する問題をトラブルシューティングする」を参照してください。

1. Tableau Server を停止します。

   コマンドプロンプトで、次のコマンドを実行します。

   ```
   tsm stop
   ```

2. ネットワーク共有のスナップショットをネットワークに復元して、ファイル ストア データを復元します。具体的なステップは、ネットワークによって異なります。

3. リポジトリデータを復元します。

   次の復元コマンドを使用して、リポジトリデータを復元します。

   ```
   tsm maintenance snapshot-backup restore
   ```

   > **注:** 暗号化キー ソリューションにクラウド ソリューションを使用している場合は、バックアップを復元するサーバーに、CMK が展開されているクラウド インスタンスを復号する権限が付与されていることを確認してください。

4. Tableau Server を再起動します。

   コマンドプロンプトで、次のコマンドを実行します。

   ```
   tsm start
   ```

オプション 2: リポジトリを個別にバックアップする

このオプションは、外部リポジトリのホストプラットフォームでスナップショット バックアップを実行できる場合にのみ推奨されます。ホストプラットフォームとして Azure を使用している場合は、オプション 1 を使用することをお勧めします。

スナップショットバックアップの作成

1. スナップショットバックアップを準備する。

   以下のコマンドを実行して、リポジトリのバックアップファイルを作成し、それをネットワーク共有に一時的にコピーします。Tableau Server は、バックアップ処理中も正常に動作し続けます。一貫性のあるスナップショットを作成するために、使用されていない抽出を削除する内部プロセスが一時停止されます。このプロセスは、後のステップで説明されているバックアッププロセスが完了すると、再開されます。

   ```
   tsm maintenance snapshot-backup prepare
   ```

   準備のステップが完了すると、**スナップショットバックアップの準備に成功しました**というメッセージが表示されます。

   > **注:** 準備のステップでは、KMS キーおよびアセットキーのバックアップが作成されます。暗号化キーソリューションにクラウドソリューションを使用している場合は、復元に必要なキーの復号に使用する CMK にアクセスできる必要があります。AWS での暗号化キーソリューションの詳細については、「AWS キー管理システム」を参照してください。Azure での暗号化キーソリューションの詳細については、「Azure Key Vault」を参照してください。

2. ネットワーク共有のスナップショットを作成する。

   適切なプロセスを使用して、ネットワーク共有のスナップショットを作成します。スナップショットは、特定の時点で取得された共有の読み取り専用バージョンです。これには、ファイルストアデータが含まれます。このネットワーク共有のスナップショットを作成するステップは、ネットワークによって異なります。詳細については、ご使用のネットワークのドキュメントを参照してください。

3. リポジトリのバックアップを作成する: 外部リポジトリをホストしているプラットフォームのバックアップテクノロジーを使用してバックアップを作成します。

> **重要:** ネットワーク共有とリポジトリバックアップのスナップショットは、準備のステップを完了してから3時間以内に完了する必要があります(ステップ1)。これは、ファイルストアとリポジトリのバックアップが同期され、復元が正しく動作するようにするためです。

AWS DB インスタンスのスナップショットの作成の詳細については、「DB スナップショットの作成」を参照してください。

Azure DB インスタンスのバックアップの作成の詳細については、「Flexible Server でのバックアップと復元」(PostgreSQL 12 以降)、または「単一サーバーでのバックアップと復元」(PostgreSQL 11 以前) を参照してください。

> **注:** 暗号化キーソリューションにクラウドソリューションを使用している場合は、バックアップを復元するサーバーが、CMK がデプロイされているクラウドインスタンスに対する復号化の権限を持っていることを確認してください。

4. スナップショットバックアッププロセスを完了します。

   以下のコマンドを実行して、バックアッププロセスを完了し、一時停止していた内部処理を再開します。

   ```
   tsm maintenance snapshot-backup complete
   ```

スナップショットバックアップの復元

これらのステップは、シングルノードとマルチノードの両方の Tableau Server インストールに適用されます。

1. 外部リポジトリの データベース バックアップを使用します。クラウドプラットフォームを使用してリポジトリをホストしている場合、通常は、バックアップを復元するための新しい データベース インスタンスを作成する必要があります。

   新しいインスタンスを作成する詳細な手順については、「外部リポジトリのインストール」のホスティング ソリューションのオプションを参照してください。

2. 「PostgreSQL 外部リポジトリでの Tableau Server のインストール」トピックのステップ 1 の手順を使用して、新しいインスタンスの構成ファイルを作成します。

3. Tableau Server を停止します。

   コマンドプロンプトで、次のコマンドを実行します。

   ```
   tsm stop
   ```

4. 外部リポジトリの復元に新しいデータベース インスタンスが必要な場合は、次のコマンドを使用して Tableau Server に新しいデータベース インスタンスを指定し、その変更を Tableau Server に適用します。

   ```
   tsm topology external-services repository replace-host -f
   <filename>.json -c <ssl certificate file>.pem

   tsm pending-changes apply
   ```

   この .json fファイルは、ステップ 2 で作成した構成ファイルです。この証明書ファイルは、新しいデータベース インスタンスからダウンロードした SSL 証明書です。

5. ネットワーク共有のスナップショットをネットワークに復元して、ファイル ストア データを復元します。具体的なステップは、ネットワークによって異なります。

   > **注:** 一部のテクノロジでは、復元を実行するときに新しいネットワーク共有を作成する必要があります。これがネットワーク接続ストレージに当てはまる場合は、Tableau Server を停止する**前に**復元を実行できます。ファイル ストアのデータを新しいネットワーク共有に復元する場合は、新しいネットワーク共有を使用するように Tableau Server を構成する必要があります。詳細については、「別の外部ストレージを使用するように Tableau Server を構成する」を参照してください。

6. 次のコマンドを実行して、KMS および Asset のキーを復元します。

   ```
   tsm maintenance snapshot-backup restore
   ```

> **注:** 暗号化キー ソリューションにクラウド ソリューションを使用している場合は、バックアップを復元するサーバーが、CMK がデプロイされているクラウド インスタンスに対する復号化の権限を持っていることを確認してください。

7. 次のコマンドを実行して Tableau Server を再起動します。

```
tsm start
```

実行可能なユーザー

Tableau Server 管理者は、Tableau Server のバックアップと復元を行うことができます。さらに、外部ストレージにアクセスし、それに対するスナップショット バックアップを実行するパーミッションが必要です。

外部ファイル ストアのパフォーマンスに関する考慮事項

このトピックでは、外部ファイル ストアを使用している場合に最適なパフォーマンスを確保するために考慮する必要がある要因を示します。

ファイル ストアが Tableau Server の外部で構成されているこのシナリオでは、抽出をネットワーク共有に保存します。これは、Tableau Server からこのデータへのアクセスがネットワーク経由で行われることを意味します。最適なパフォーマンスを確保するため、次のことをお勧めします。

- エンタープライズ グレードのストレージ システムを使用して、信頼性と高いデータ アクセス パフォーマンスを確保します。
- 十分な読み取り IOPS をサポートするストレージ システムを使用します。
  - ソリッド ステート ドライブを使用します。ハード ディスクが唯一のオプションである場合は、できるだけ高速なディスクを使用します。
- 次のようなネットワーク インフラストラクチャを使用します。
  - Tableau Server とストレージ システム間の高速データ転送をサポートするため、10 GB 以上の速度のイーサネット。
  - Tableau Server とストレージ システムの間のストレージ遅延が 10 ミリ秒以下。

上記の推奨事項は Tableau チームが行ったテストに基づいており、実際の要件とパフォーマンスは異なる場合があります。パフォーマンスを評価してリソース要件を決定するために、独自のベンチマークを作成することを強くお勧めします。

ベンチマークを作成する際は、ワークブックの読み込み時間を主要なメトリクスの 1 つとして Tableau Server の全体的なパフォーマンスを考慮してください。これが特に重要なのは、外部ファイル ストアが主に抽出ベースのワークブックに影響するためです。

Tabjolt を使用してベンチマークを行うことができます。

実行可能なユーザー

Tableau Server 管理者は、Tableau Server のパフォーマンスを監視します。ただし、ネットワーク、ハードウェア、ストレージについては考慮事項があり、これらのリソースの構成を変更するには、このいずれかへのアクセスが必要になる可能性があります。これらのリソースにアクセスできない場合は、ネットワーク管理者と協力して変更を加えることもできます。

Tableau Server 外部リポジトリ

Tableau Server リポジトリは、すべてのユーザー インタラクション、抽出の更新などに関するデータを格納する PostgreSQL データベースです。

リポジトリは、Tableau Server と同じノードにローカルにインストールするか、外部にインストールすることができます。

- **ローカル リポジトリ:** PostgreSQL データベースがローカルにインストールおよび展開されます。つまりTableau Server と共に展開されます。
- **外部 リポジトリ:** PostgreSQL データベースは外部に展開されます。外部 リポジトリは、Amazon RDS、Azure Database、Google Cloud にインストールするか、またはスタンドアロンとしてインストールできます。

Tableau Server リポジトリの全般的情報については、以下を参照してください。

- Tableau Server データディクショナリについて
- Tableau Server リポジトリでデータを収集する

外部リポジトリとしてサポートしているホストは次のとおりです。

- Amazon RDS (バージョン 2019.3 以降)
- Azure Database (バージョン 2020.4 以降)
- スタンドアロン PostgreSQL インスタンス (バージョン 2021.2 以降)
- Google Cloud SQL for PostgreSQL インスタンス (バージョン 2021.4 以降)

このトピックでは、Tableau Server の外部リポジトリの概要を説明します。

外部リポジトリに関する考慮事項

Amazon RDS と Azure Database では、PostgreSQL 向けにスケーラビリティ、信頼性、高可用性、およびセキュリティがに組み込まれています。クラウド製品とより密接に統合することで、このような追加のメリットを活用できるようになります。

スタンドアロン PostgreSQL インスタンスを設定する場合は、必要に応じて高可用性とスケールを設定および管理する必要があります。

> **重要:** リポジトリを直接変更しないでください。Tableau Server が破損するおそれがあります。お客様がリポジトリを変更した場合、そのインストールは Tableau でサポートされなくなります。

クラウドプラットフォーム

クラウドプラットフォームを使用して外部リポジトリをホストする場合、ホストされたインフラストラクチャで PostgreSQL インスタンスを実行して自分で管理するか、フルマネージドサービスオプションを選択できます。

- **自己管理型:** ホストされたインフラストラクチャで PostgreSQL インスタンスを自分で設定して管理します。たとえば、AWS をクラウドプラットフォームとして使用している場合は、EC2 インスタンスを使用して、PostgreSQL インスタンスを実行、管理、維持できます。
- **フルマネージド:** フルマネージドサービスを選択します。たとえば、AWS をクラウドプラットフォームとして使用している場合は、RDS オプションを使用して、外部リポジトリをホストできます。

自己管理型とフルマネージドのどちらかを選択する際の重要な考慮事項の 1 つは、自己管理型オプションではほとんどの制御が可能ですが、VM を維持して、多くのデータベース管理タスクを行う必要があります。フルマネージドオプションでは、セットアップ、構成管理、メンテナンスが容易になります。

2 つのオプションのいずれかを選択する際に考慮する必要があることの包括的なリストを次に示します。

- セットアップとメンテナンスの要件
- 高可用性とディザスター リカバリオプション
- パフォーマンス、スケーラビリティ、監視機能
- セキュリティ メンテナンス
- 運用コスト、サービス コスト、人件費

Azure でこれらの 2 つのオプションを比較する方法の例は、Microsoft のサイト「Azure で適切な PostgreSQL サーバー オプションを選択する」に示されています。

要件

- Tableau Server が以下のバージョンであることが必要です。
    - AWS で使用する場合、2019.3 以降
    - Azure で使用する場合、2020.4 以降

      使用する PostgreSQL の正しいバージョンについては、「Azure Database for PostgreSQL - フレキシブル サーバー」を参照してください。
    - スタンドアロン PostgreSQL インスタンスで使用する場合、2021.2 以降 (オンプレミス、Azure Vm、または AWS EC 2 のインストールで使用できます)。
    - Google Cloud で使用する場合、2021.4 以降 (Google Cloud VM で PostgreSQL インスタンスまたはスタンドアロン PostgreSQL 用に使用)
- Tableau Server で Advanced Management のキーがアクティブ化されている必要があります。
- 外部リポジトリをホストする予定の場所に応じて、次のいずれかに精通している必要があります。

- Amazon RDS データベースの設定と管理。
- Azure データベースの設定と管理。
- スタンドアロン インストールとしての PostgreSQL データベースの設定と管理。
- Google Cloud PostgreSQL インスタンスの設定と管理

バージョニング

スタンドアロン Tableau Server リポジトリを使用できるように、正しいバージョンの PostgreSQL を実行する必要があります。Tableau Server のバージョンの互換性を次の表に示します。

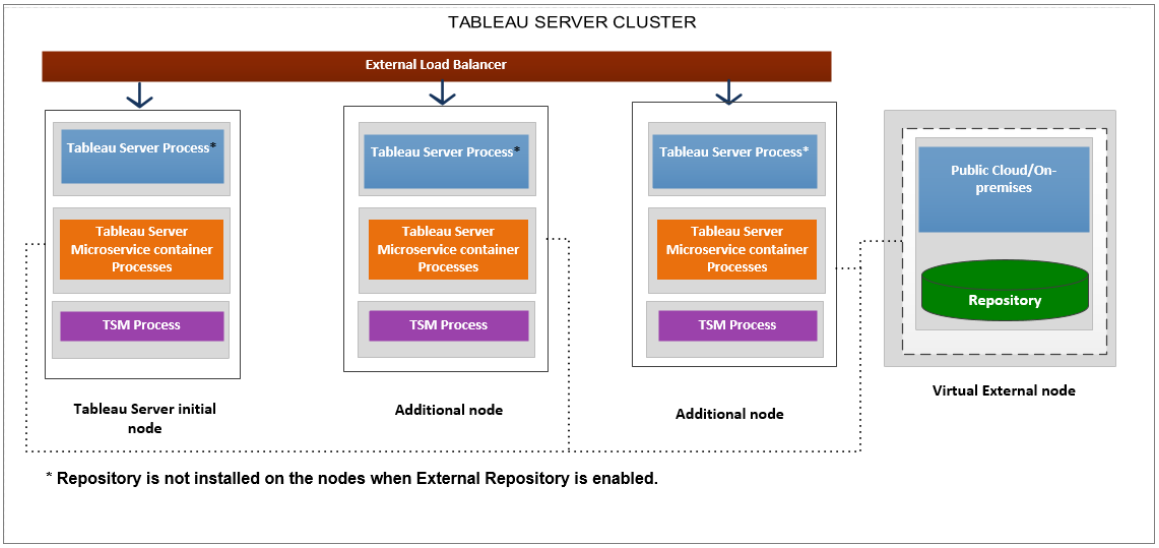
注: Tableau Server バージョンの最大互換バージョンは、最小メジャー バージョンまたはそれに対するマイナー アップデートです。たとえば、PostgreSQL の最小互換バージョンが 13.4 の場合、最大互換バージョンは 13.x で、<x> は 4 と同じかそれ以上です。

Tableau Server のバージョン	**PostgreSQL の互換性のある最低バージョン**
2022.3.0 ~ 2022.3.7 2023.1.0 ~ 2023.1.4	13.7
2022.1.17 - 2022.1.19 2022.3.8 ~ 2022.3.19 2023.1.5 ~ 2023.1.15 2023.3.0 ~ 2023.3.8	13.11
2022.3.20 ~ 2022.3.x 2023.1.16 ~ 2023.1.x 2023.3.9 ~ 2023.3.x	13.14
2019.3 ~ 2019.3.8	15.6
2024.2.9 ~ 2024.2.13	15.10

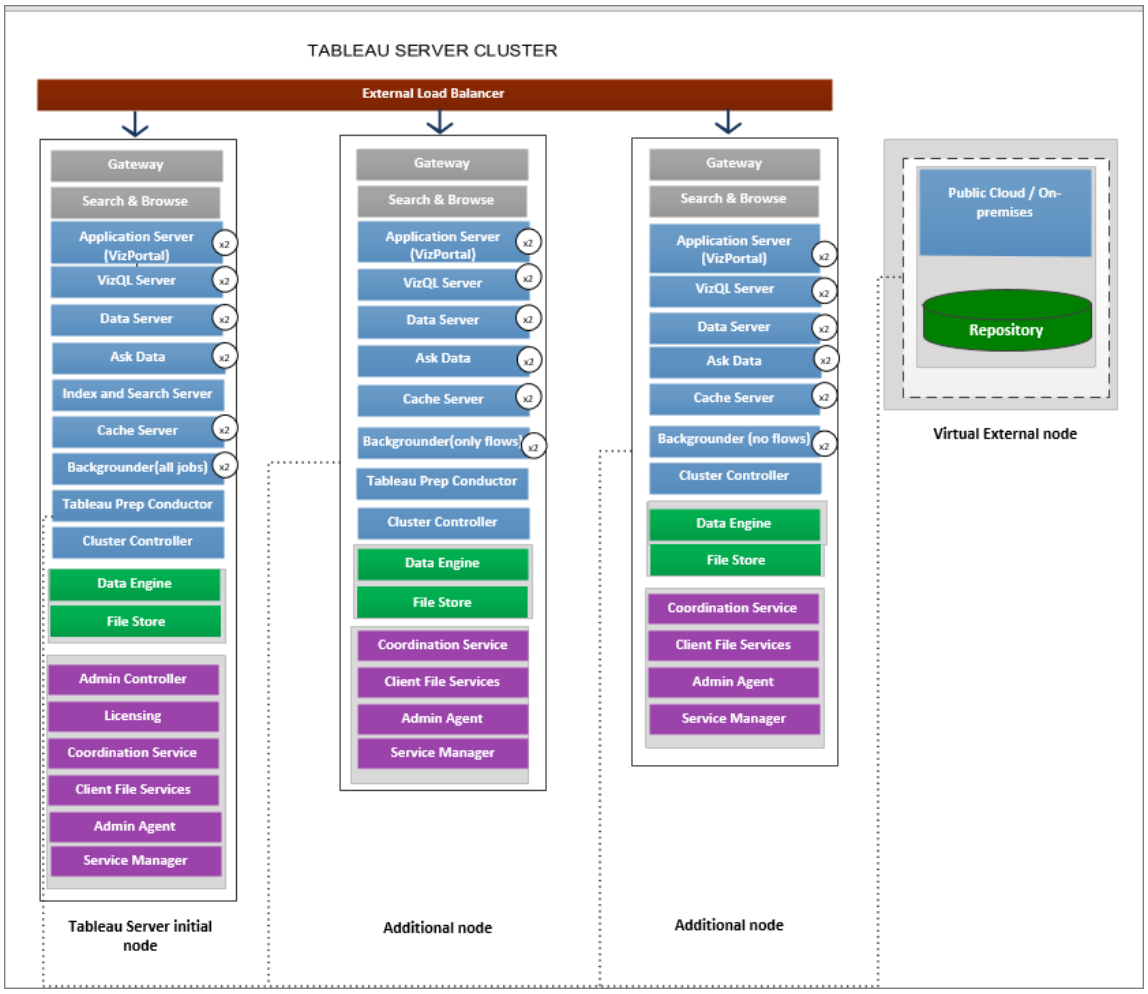
2025.1.0 ~ 2025.1.6	
2024.2.14 以降 2025.1.7 以降	15.13

トポロジ

次の図は、外部リポジトリを使用した **Tableau Server** トポロジの概要バージョンです。



次の図は、外部リポジトリを使用した **Tableau Server** トポロジの詳細バージョンであり、各ノードにインストールされているすべてのプロセスを示しています。



外部リポジトリの管理

ライセンス管理

この機能を有効にするには、まず Tableau Server で Advanced Management を有効化する必要があります。詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。Advanced Management がない、またはライセンスの有効期限が切れている場合は、次のような動作になります。

- インストール中に Tableau Server で外部リポジトリを使用するように構成しようとするとエラー メッセージが表示されますが、インストールは続行でき、Tableau Server リポジトリは

ローカルにインストールされます。単一サーバーインストールの場合、リポジトリは Tableau Server と同じマシンにインストールされます。分散インストールの場合、リポジトリは Tableau Server クラスタのノードのいずれかにインストールされます。

- Tableau Server インストール上で外部の Tableau Server リポジトリを既に使用しており、Advanced Management ライセンスの有効期限が切れている場合、サーバーの再起動時にエラーが発生します。Advanced Management 機能を利用できなくなっているが、有効な Tableau Server ライセンスがある場合は、引き続きバックアップを作成できます。また、外部リポジトリから Advanced Management を必要としないローカル リポジトリに移行することにより、サーバーをもう一度稼動させることもできます。外部リポジトリからローカル リポジトリへの移行方法の詳細については、Tableau Server リポジトリの再構成を参照してください。

サポートされる移行シナリオ

- ローカルから外部へのリポジトリの移動
- 外部からローカルへのリポジトリの移動

バックアップと復元

外部リポジトリのみが構成されている場合:

バックアップと復元のプロセスは、ローカルと外部リポジトリの両方で引き続き変わりなく、Tableau Server データのバックアップトピックで説明されています。

- バックアップと復元のコマンドは、ローカルと外部リポジトリの両方で同じように動作します。Tableau Server が外部リポジトリを使用する場合のバックアップでは、バックアップ用に必要なディスク空き容量が大きくなるため、ディスク領域を構成するときにはこの点を留意する必要があります。
- バックアップ ファイルを格納する既定の場所および他のサポートされている場所は、リポジトリがローカルか外部かに関わらず同じです。

次の例外は外部リポジトリを使用している Tableau Server に適用されます。

- 外部の Tableau Server リポジトリで作成されたカスタム ユーザー アカウントは復元に含まれますが、カスタム ユーザー アカウントのパスワードは含まれません。復元が完了したら、パスワードの再構成が必要です。カスタム ユーザー アカウントは PostgreSQL データベースのユーザー アカウントであり、PostgreSQL データベースに接続するために SQL または他のデータベース クライアント ソフトウェアにより使用されます。

  注: これらのカスタム アカウントはセキュリティ対策として無効になりますが、再構成することもできます。

- 構成とトポロジはバックアップに含めないでください。構成とトポロジ設定のエクスポート方法の詳細については、Tableau Server の完全なバックアップと復元を実行しますを参照してください。

外部リポジトリと外部ファイル ストアの両方が構成されている場合:

Tableau Server 用に外部リポジトリと外部ファイル ストアの両方が構成されている場合は、2 つのオプションから選択できます。これらのオプションを使用すると、外部リポジトリとファイル ストアをホストするために使用するクラウドプラットフォームのスナップショット バックアップ機能を活用することができます。ステップごとの詳細な説明については、外部ファイル ストアによるバックアップと復元を参照してください。

SSL 接続

Tableau Server から外部リポジトリへの TLS / SSL 接続を要求するかどうかを選択できます。

暗号化された接続を使用する必要がない場合は、暗号化されていない接続を許可するように外部リポジトリを構成し、Tableau Server の外部リポジトリを構成するときに `-no-ssl` オプションを使用する必要があります。詳細については、tsm topology external-services repository enableを参照してください。

インストール後に SSL を有効または無効にする場合は、tsm security repository-ssl enable または tsm security repository-ssl disable を使用します。このオプションは、2021.4以降で使用できます。

SSL 証明書の更新

RDS、Azure データベース、Google Cloud PostgreSQL インスタンス、またはスタンドアロン PostgreSQL インスタンスの SSL 証明書の計画された有効期限の一環として、インスタンスを新しい証明書ファイルを使用して更新する必要がある場合は、新しい証明書ファイルを使用するように Tableau Server 設定も更新する必要があります。これを行うには、最新のファイルをダウンロードした後、*tsm topology external-services repository replace-host* コマンドを実行して新しい証明書ファイルを指定します。

高可用性に関する考慮事項

Tableau Server では、外部リポジトリの高可用性の管理やセットアップは行われません。

- **AWS:** Amazon RDS には、高可用性の実現やフェールオーバーの管理に使用できる機能が用意されています。詳細については、「Amazon RDS の高可用性」を参照してください。
- **Azure:** Azure には、高可用性の実現やフェールオーバーの管理に使用できる機能が用意されています。詳細については、「Azure Database の高可用性」を参照してください。
- **Google Cloud:** Google Cloud には、高可用性の実現やフェールオーバーの管理に使用できる機能が用意されています。詳細については、Google Cloud の高可用性に関するドキュメントを参照してください。
- **スタンドアロン PostgreSQL インスタンス:** PostgreSQL では、高可用性の実現やフェールオーバーの管理に使用できる可用性に優れた機能が用意されています。詳細については、「PostgreSQL の高可用性」を参照してください。

アップグレードに関する考慮事項

これは、Tableau Server で外部リポジトリ構成を使用している場合にのみ適用されます。

外部リポジトリを使用している場合は、アップグレード時に追加の手順が必要になる場合があります。

- **バージョンの変更なし** - PostgreSQL にバージョンの変更がない場合、特別なアクションは必要ありません。
- **マイナー バージョンの変更** - PostgreSQL にマイナー バージョンの変更がある場合、Tableau Server をアップグレードする前に外部リポジトリをアップグレードする必要があります。ほとんど

の場合、アップグレードを行うためのインプレース メソッドがあります。使用するメソッドはリポジトリの場所によって異なり、このドキュメントの対象外です。

- **メジャー バージョンの変更** - PostgreSQL にメジャー バージョンの変更がある場合、「新しいメジャー バージョンの PostgreSQL を外部リポジトリとして使用する Tableau Server のアップグレード」で説明されている手順に従う必要があります。

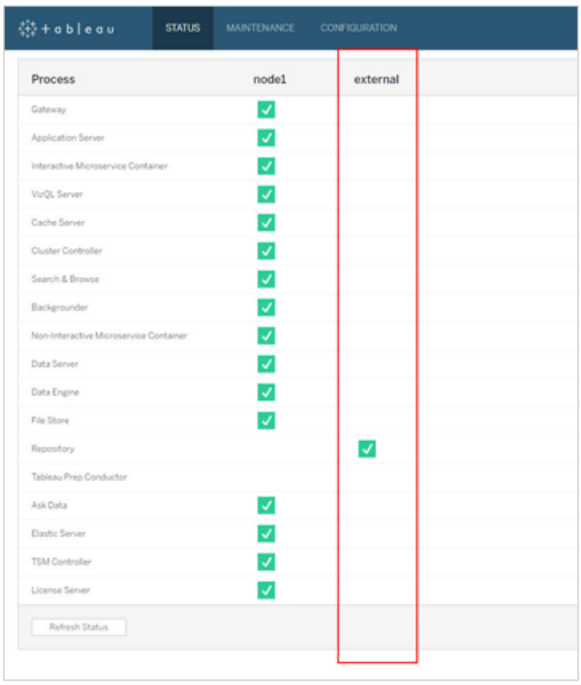
  次のステップが含まれます。

  1. PostgreSQL DB の新しいインスタンスを作成します。詳細については、以下を参照してください。
     - AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成
     - 詳細については、Azure で Azure Database PostgreSQL インスタンスを作成するを参照してください。
     - Google Cloud 上で PostgreSQL インスタンスの Cloud SQL を作成する
     - PostgreSQL データベースをスタンドアロン インストールで作成
  2. 構成ファイルを作成し、ステップ 1 で作成した新しいインスタンスの SSI 証明書ファイルをダウンロードします。

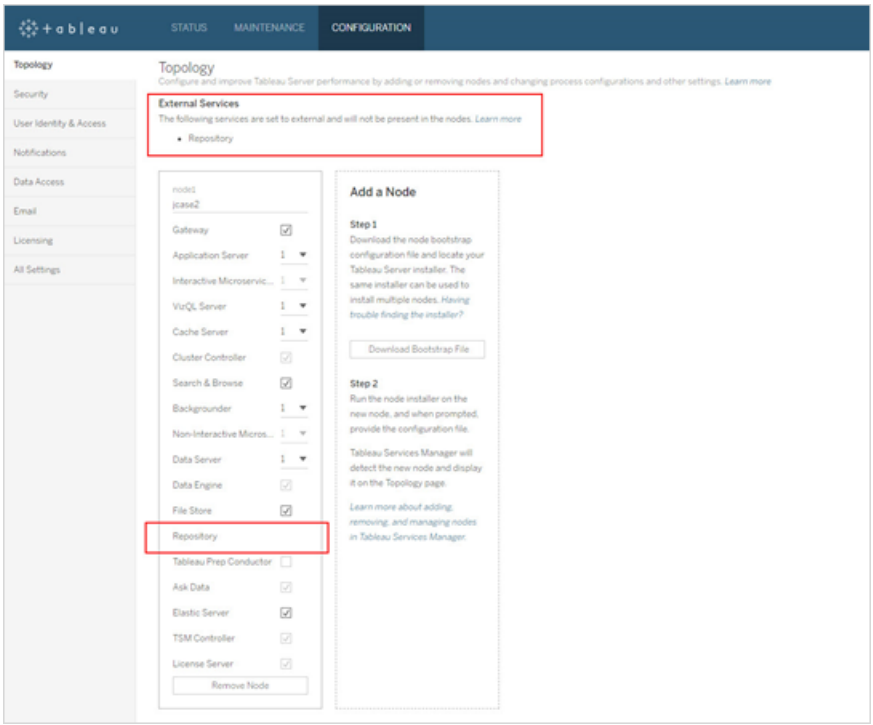
  アップグレード中に、構成ファイルを使用して新しいインスタンスに Tableau Server を指定する必要があります。アップグレードプロセスによって、コンテンツが現在の外部リポジトリから新しいインスタンスに移行します。詳細については、新しいメジャー バージョンの PostgreSQL を外部リポジトリとして使用する Tableau Server のアップグレードを参照してください。

リポジトリのステータスの監視

TSM のステータス ページには、Tableau Server インストールの追加ノードとして Tableau Server 外部サービスが表示されます。



トポロジ タブには、Tableau Server の外部サービスが構成されているかどうかが示されます。



ログの取得

Tableau Server のログには、外部リポジトリからのログは含まれません。以下の方法を使用して、インスタンスに固有のログを取得します。

- **AWS:** Amazon RDS PostgreSQL インスタンスでのログの設定の詳細については、「PostgreSQL データベースのログ ファイル」を参照してください。
- **Azure:** Azure Database PostgreSQL インスタンスでのログの設定の詳細については、「PostgreSQL の Azure Database のログ」を参照してください。
- **Google Cloud:** PostgreSQL インスタンスのログを表示およびクエリする方法については、「Cloud SQL のロギング」を参照してください。
- **スタンドアロン PostgreSQL インスタンス:** スタンドアロン PostgreSQL インスタンスのログの設定については、「エラー レポートとログ」参照してください。

次のステップ

- PostgreSQL データベースインスタンスを作成するには、次のいずれかを使用します。
    - AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成
    - 詳細については、Azure で Azure Database PostgreSQL インスタンスを作成するを参照してください。
    - Google Cloud での PostgreSQL インスタンスの作成
    - PostgreSQL データベースをスタンドアロン インストールで作成
- PostgreSQL 外部リポジトリでの Tableau Server のインストール
- Tableau Server リポジトリの再構成
    - ローカルから外部への移行
    - 外部からローカルへの移行

AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成

バージョン 2019.3 以降では、AWS クラウドプラットフォームで外部リポジトリをホストできます。このトピックでは、Tableau Server 外部リポジトリとして使用する AWSRDSPostgreSQL DB インスタ

ンスを作成する方法について説明します。

外部リポジトリで使用できるホストの完全なリストについては、Tableau Server 外部リポジトリを参照してください。

要件と推奨事項

- 少なくとも 8 個の vCPU と32GB の RAM を備えたインスタンスを使用してください。これは、Tableau Server 外部リポジトリに使用する AWS RDS インスタンスの推奨最小サイズですが、正確な要件は実際の要件や使用方法によって異なります。パフォーマンスを向上させるには、16 個の vCPU と128GB の RAM を備えた、Amazon RDS メモリに最適化されたインスタンス タイプを使用することをお勧めします。

  小さなインスタンス タイプから始めて、後で大きなインスタンス タイプが必要になった場合に既存の RDS インスタンスをアップグレードできます。詳細については、RDS インスタンスのアップグレードを参照してください。

  Tableau Server と外部 PostgreSQL DB インスタンス間の SSL 接続を使用した安全な通信は、**必須ではありません**が、推奨されています。

- PostgreSQL DB インスタンスを Tableau Server クラスタ内のすべてのノードから到達可能にする必要があります。これを行う方法のひとつは、PostgreSQL DB インスタンスを Tableau Server クラスタ内のすべてのノードからアクセスするために必要なパーミッションを持つセキュリティグループのメンバーにすることです。

- PostgreSQL のバージョンは、Tableau Server をローカルにインストールするときに使用するバージョンと一致する必要があります。Tableau Server 2020.4 は、PostgreSQL バージョン 12 を使用します。

- Tableau は、内部のデータベース ユーザーが内部で使用するためのパスワードを自動生成します。このパスワードは長さが 32 文字で、小文字と数字で構成されています。管理者がアクセスしたり、設定したりすることはできません。外部の PostgreSQL インスタンスでは、パスワード ポリシーを設定できる場合があります (使用しているプラットフォームによって異なります)。数字と小文字以外の文字タイプを含むポリシーを指定すると、Tableau Server が外部リポジトリを使用するように設定したときにエラーが発生する可能性があります。

Amazon RDS で PostgreSQL DB インスタンスを作成する

ステップ 1: パラメーター グループを作成する

postgresql.conf ファイル内でローカル PostgreSQL インスタンス用に設定した PostgreSQL パラメーターは、DB インスタンス用の DB パラメーター グループ内で保持されます。DB インスタンスを作成する際は、関連付けられた DB パラメーター グループ内のパラメーターが読み込まれます。

Tableau Server の観点では、ほとんどのパラメーターはデフォルトに設定できます。特別なパフォーマンスやログ取得の要件がある場合はパラメーター値を変更できますが、次のパラメーターはデフォルト値のままにして、変更しないことを強くお勧めします。

- `standard_conforming_strings`
- `escape_string_warning`

また、パフォーマンスの問題を回避するために、`work_mem` の値は少なくとも 16384 に設定することをお勧めします。

PostgreSQL パラメーターの詳細と完全なリストについては、AWS サイトで「PostgreSQL パラメーターの使用」を参照してください。

ステップ 2: Amazon RDS で PostgreSQL インスタンスを作成する

新しい PostgreSQL DB インスタンスを作成するには、この Amazon ドキュメント サイトに記載されている手順に従ってください。

以下に、新しい PostgreSQL DB インスタンス用の構成オプションと推奨値を示します。

- **インスタンスの仕様**
  - **ステップ 1** で作成したパラメーター グループを使用します。
  - 使用している Tableau Server バージョンと互換性のある PostgreSQL バージョンを使用します。Tableau Server バージョンと互換性のある PostgreSQL バージョンの完全なリストについては、製品の互換性を参照してください。
  - db.m4.2xlarge かこれより大きい DB インスタンス クラスを使用します。
  - 少なくとも 100 GB のストレージを割り当てます。
  - ストレージのタイプとプロビジョニングされた IOPS: 既定値のままにします (負荷テストに応じて推奨値が変化する可能性があります)。

- **設定**
    - マスター ユーザー名として **rails** を使用する**必要**があります。

      これは外部 リポジトリが Tableau Server とともに正しく動作するための要件です。

    - AWS の要件を満たすパスワードを選択します。
- **ネットワークとセキュリティ**
    - すべての Tableau Server ノードから RDS インスタンスに到達できることを確認します。通常、ノードからのアクセスを許可するセキュリティグループを作成する必要があります。
- **データベースに関するオプション**
    - 初期データベースは作成しません。RDS インスタンスで必要なデータベースは Tableau Server によって作成されるため、データベース名は空白のままにしておく必要があります。
    - ポートは任意のものを使用できますが、既定の 5432 のままにすることをお勧めします。
    - DB パラメーターグループを**ステップ 1** で作成したものに設定します。
    - IAM DB 認証は無効のままにしておきます。
- **暗号化**
    - 暗号化を行うかどうかを選択できます。
- **バックアップ**
    - これは Tableau Server のバックアップではなく AWS の自動バックアップに関する設定です。要件を満たす設定を指定できます。
- **監視**

    - 要件に基づく設定を指定できます。

- **ログのエクスポート**

    - 要件に基づく設定を指定できます。

- **メンテナンス**

    - マイナーバージョン自動アップグレードを無効にします。Tableau Server は特定のバージョンの PostgreSQL を使用するように構築されており、Tableau Server アップグレードの際は必要に応じて PostgreSQL バージョンをアップグレードするように求められます。

- **保護の削除**

    - 要件に基づく設定を指定できます。

ステップ 3: PostgresSQL DB インスタンスのエンドポイントを取得する

作成した PostgreSQL データベースインスタンスは、AWS による初期化が完了するまで使用できません (数分間かかる可能性があります)。インスタンスの準備が完了したら、このインスタンスを Tableau Server リポジトリ用に使用するように Tableau Server を構成するためのエンドポイント情報を取得します。

ステップ 4: SSL 証明書ファイル (.pem ファイル) をダウンロードする

Tableau Server と外部リポジトリ間の SSL を使用した安全な接続は、**必須ではありません**が、推奨されています。

Tableau Server と外部リポジトリ間で安全な接続を設定する場合は、外部 DB インスタンスを Tableau Server リポジトリ用に使用するように Tableau Server を構成する際に、.pem ファイルが必要です。詳細については、「PostgreSQL DB インスタンスで SSL を使用する」を参照してください。

Tableau Server と外部リポジトリ間で安全な接続を使用する必要がない場合は、暗号化されていない接続を許可するように RDS インスタンスを構成する必要があります。

重要: RDS インスタンスの SSL 証明書で計画された有効期限の一環として、RDS インスタンスを新しい証明書ファイルで更新する必要があります。また、新しい証明書ファイルを使用するように Tableau Server の設定も更新する必要があります。これを行うには、最新のファイルをダウンロードした後、*tsm topology external-services repository replace-host* コマンドを実行して新しい証明書ファイルを指定します。

PostgreSQL DB の高可用性の構成

Tableau Server では、外部リポジトリの高可用性の管理やセットアップは行われません。Amazon RDS では、高可用性を提供したりフェールオーバーを管理したりするために使用できる高可用性機能が提供されています。詳細については、「Amazon RDS での高可用性」を参照してください。

PostgreSQL DB の災害復旧

災害が発生すると、新しい RDS インスタンスの設定が必要になる場合があります。RDS インスタンスの問題を復旧する必要がある場合には、別のシナリオがあります。たとえば、Tableau Server をアップグレードする場合、お使いの RDS インスタンスで PostgreSQL バージョンもアップグレードする必要があります。PostgreSQL アップグレードが正常に実行されなかった場合は、新しい RDS インスタンスを使用する必要があります。このようなシナリオで新しい RDS インスタンスを使用するように Tableau Server を構成するには、次の手順を使用します。

1. **新しい RDS インスタンスにスナップショットを復元する。** AWS では、既存の RDS インスタンスへのスナップショットの復元に対応していません。RDS スナップショットのバックアップと復元に関する詳細については、「Amazon RDS のバックアップと復元」を参照してください。

2. **JSON 設定ファイルを新規作成する。** 新しい RDS インスタンスの接続情報が含まれています。JSON 設定ファイルの作成に関する詳細については、「PostgreSQL 外部リポジトリでの Tableau Server のインストール」の**ステップ 1** を参照してください。

3. **`tsm topology external-services repository replace-host` コマンドを使用して、**新しい RDS インスタンスに Tableau Server を指定します。

   `tsm topology external-services repository replace-host` コマンドの詳細については、tsm topology を参照してください。

実行可能なユーザー

Tableau Server を、外部リポジトリを使用するように構成できるのは、Tableau Server 管理者のみです。RDS インスタンスを作成するには、AWS アカウントも必要です。

次のステップ

新規インストールの場合: PostgreSQL 外部リポジトリでの Tableau Server のインストール

外部リポジトリを使用するように既存の Tableau Server を構成する場合は、「Tableau Server リポジトリの再構成」を参照してください。

Azure で Azure Database PostgreSQL インスタンスを作成する

バージョン 2020.4 以降では、Azure クラウドプラットフォームで外部リポジトリをホストできます。このトピックでは、PostgreSQL 用の Azure Database インスタンスを作成して Tableau Server の外部リポジトリとして使用する方法について説明します。

要件と推奨事項

- Tableau Server 外部リポジトリ用に 50 GB のストレージを備えた 8 台の仮想コア メモリ最適化サーバーを使用することを推奨します。ただし、要求事項と使用状況によって正確な要件は異なります。Tableau Server がすでにある場合は、既存のリポジトリの使用状況を確認して、ストレージのニーズを判断します。

  必要な場合は、リソースを拡張することもできます。詳細については、「PostgreSQL Azure Database リソースのスケーリング」を参照してください。

- Tableau Server と外部 PostgreSQL DB インスタンス間の SSL を使用した安全な通信は、推奨されていますが、必須ではありません。

  Tableau Server と外部リポジトリ間の安全な接続を使用しない場合は、暗号化されていない接続を許可するように Azure Database を構成する必要があります。

- PostgreSQL DB インスタンスを Tableau Server クラスタ内のすべてのノードから到達可能にする必要があります。データベース インスタンスは、すべての Tableau Server ノードからの

接続を許可するように設定する必要があります。この設定には 2 つの方法があります。

- 最も安全な方法: PostgreSQL 用の Azure Database インスタンスを構成して、仮想ネットワークのサービス エンドポイントを介したプライベート アクセスのみを許可します。詳細については、「Azure Database for PostgreSQL の仮想ネットワーク サービス エンドポイントと規則を使用する」および「VNet サービス エンドポイントを作成および管理する」を参照してください。

  Azure 仮想ネットワークに関する概要トピックを確認することもできます。

- もう1 つの方法として、PostgreSQL 用 Azure Database を構成して、パブリック IP アドレスの範囲からの接続を許可します。この方式では、Azure Database のエンドポイントをインターネット上からのパブリック アクセスにさらします。

- Azure Database インスタンスを設定する場合は、管理者ユーザー名として **postgres** を使用することをお勧めします。別のユーザー名を使用する場合は、ユーザー名が **pg** または **azure** で始まらないようにしてください。また、ユーザー名は、**rails**、**tblwgadmin**、**tableau**、**readonly**、または **tbladminviews** にすることはできません。

- PostgreSQL のバージョンは、Tableau Server をローカルにインストールするときに使用するバージョンと一致する必要があります。Tableau Server バージョンと互換性のある PostgreSQL バージョンの完全なリストについては、製品の互換性を参照してください。

- Tableau は、内部のデータベース ユーザーが内部で使用するためのパスワードを自動生成します。このパスワードは長さが 32 文字で、小文字と数字で構成されています。管理者がアクセスしたり、設定したりすることはできません。外部の PostgreSQL インスタンスでは、パスワード ポリシーを設定できる場合があります (使用しているプラットフォームによって異なります)。数字と小文字以外の文字タイプを含むポリシーを指定すると、Tableau Server が外部リポジトリを使用するように設定したときにエラーが発生する可能性があります。

Azure Database で PostgreSQL インスタンスを作成する

ステップ 1: PostgreSQL 用 Azure Database インスタンスの委任サブネットを作成する

ネットワークのオプションとしてプライベート アクセスを設定する場合、このステップはインスタンスを作成するときの必須条件になります。通信をセキュリティで保護するためには、データベースへのアクセスをプライベートに設定する必要があります。これにより、仮想ネットワーク内に作成した仮想マシン

であればデータベース インスタンスに接続できますが、仮想ネットワークの外部にあるマシンからは接続できません。

現在 Tableau Server をホストしている同じ仮想ネットワークで、Azure データベース インスタンス用の新しい委任サブネットを作成します。プライベート アクセスの設定の詳細については、Azure Web サイトの「PostgreSQL 用 Azure Database のネットワーク オプション - フレキシブル サーバー」を参照してください。

ステップ 2: PostgreSQL 用 Azure Database インスタンスを作成する

新しい PostgreSQL 用 Azure Database を作成するには、「Amazon ドキュメント サイト」に記載されている手順に従ってください。

以下に、新しい PostgreSQL DB インスタンス用の設定オプションと必要な値、および推奨事項を示します。

- **サーバーの詳細**
    - 新しいサーバーを作成するには、データ ソースとして [なし] を指定します。
    - 管理者のユーザー名は **postgres** を使用することをお勧めします。別のユーザー名を使用する場合は、ユーザー名が **pg** または **azure** で始まらないようにしてください。また、ユーザー名は、**rails**、**tblwgadmin**、**tableau**、**readonly**、または **tbladminviews** にすることはできません。
    - AWS の要件を満たすパスワードを選択します。
    - 使用している Tableau Server バージョンと互換性のある PostgreSQL バージョンを使用します。Tableau Server バージョンと互換性のある PostgreSQL バージョンの完全なリストについては、製品の互換性を参照してください。
    - 少なくとも 512 GB のストレージを割り当てます。
    - Azure Database for PostgreSQL の "azure_pg_admin" ユーザーには、uuid-ossp および unaccent 拡張機能の使用を許可する必要があります。
- **コンピューティングとストレージ**

- 少なくとも、汎用コンピューター層とStandard_D8s_v3 (8 仮想コア、32 GB RAM) の計算サイズを備えたFlexible Server を使用します。

- **ネットワーク オプション**

  - プライベート アクセス (仮想ネットワーク) を選択します。これにより、データベースとのプライベートで安全な通信を確保します。

- **高可用性**

  - 要件に応じて高可用性オプションを有効にします。

- **バックアップ**

  - 要件に応じて保持期間を設定します。これは Tableau Server のバックアップではなく、Asure の自動バックアップに関する設定です。要件を満たす設定を指定できます。

ステップ 3: サーバー レベルのファイアウォール ルールを構成する

データベースを作成したら、サーバー レベルのファイアウォール規則を構成して、Tableau Server ノードへのアクセスを許可します。

すべての Tableau Server ノードからDatabase インスタンスに接続できることを確認します。

ステップ 4: PostgreSQL インスタンス用の Azure Database を構成する

Tableau Server の観点では、インスタンスのほとんどのパラメーター値はデフォルトに設定できます。特別なパフォーマンスやログ取得の要件がある場合はパラメーター値を変更できますが、次のパラメーターはデフォルト値のままにして、変更しないことを強くお勧めします。

- `standard_conforming_strings`

- `escape_string_warning`

また、パフォーマンスの問題を回避するために、`work_mem` の値は少なくとも 16384 に設定することをお勧めします。

サーバー パラメーターの構成方法については、「Azure のドキュメント」を参照してください。

ステップ 5: PostgresSQL DB インスタンスのエンドポイントを取得する

インスタンスの準備が完了したら、Tableau Server リポジトリとしてこのインスタンスを使用するように Tableau Server を構成するためのエンドポイント情報を取得します。

ステップ 6: SSL 証明書ファイルをダウンロードする

Tableau Server と外部リポジトリ間の SSL を使用した安全な通信は、**必須ではありません**が、推奨されています。

Tableau Server と外部リポジトリ間の通信にセキュリティで保護された接続を設定する場合は、証明書ファイルをダウンロードします。この証明書ファイルは、この外部リポジトリを使用するように Tableau Server を構成する際に必要になります。詳細については、「PostgreSQL 用 Azure Database の TLS 接続の構成」を参照してください。

Tableau Server と外部リポジトリ間でセキュリティで保護された接続を使用する必要がない場合は、暗号化されていない接続を許可するように Azure Database インスタンスを構成する必要があります。

PostgreSQL DB の高可用性の構成

Tableau Server では、外部リポジトリの高可用性の管理やセットアップは行われません。Azure には、高可用性を提供するために使用できる機能が用意されています。詳細については、「Azure Database の高可用性」を参照してください。

PostgreSQL DB の災害復旧

災害が発生すると、PostgreSQL 用 Azure Database の新しいインスタンスの設定が必要になる場合があります。他のシナリオにおいても、データベース インスタンスの問題から復旧することが必要にある場合があります。このようなシナリオで新しい Azure Database インスタンスを使用するように Tableau Server を構成するには、次の手順に従います。

1. **新しい Azure Database インスタンスにバックアップを復元する。** PostgreSQL 用 Azure Database で復元を実行すると、元のサーバーのバックアップから新しいサーバーが作成されます。PostgreSQL 用 Azure Database のバックアップと復元の詳細については、「PostgreSQL 用 Azure Database のバックアップと復元」を参照してください。

2. **JSON 設定ファイルを新規作成する。**設定ファイルには、新しい PostgreSQL 用 Azure Database インスタンスへの接続情報が含まれています。JSON 設定ファイルの作成に関する詳細については、「PostgreSQL 外部リポジトリでの Tableau Server のインストール」の**ステップ 1** を参照してください。

3. **`tsm topology external-services repository replace-host` コマンドを使用して、**新しい Azure Database for PostgreSQL インスタンスに Tableau Server を指定します。

   `tsm topology external-services repository replace-host` コマンドの詳細については、tsm topology を参照してください。

実行可能なユーザー

Tableau Server を、外部リポジトリを使用するように構成できるのは、Tableau Server 管理者のみです。Azure Database を作成するには、AWS アカウントも必要です。

次のステップ

新規インストールの場合: PostgreSQL 外部リポジトリでの Tableau Server のインストール

外部リポジトリを使用するように既存の Tableau Server を構成する場合は、「Tableau Server リポジトリの再構成」を参照してください。

Google Cloud での PostgreSQL インスタンスの作成

バージョン 2021.4 以降では、Google Cloud Platform で Tableau Server 外部リポジトリをホストできます。このトピックでは、Tableau Server 外部リポジトリとして使用する PostgreSQL インスタンスを Google Cloud で作成する方法について説明します。

外部リポジトリで使用できるホストの完全なリストについては、Tableau Server 外部リポジトリを参照してください。

要件と推奨事項

- 少なくとも 8 個の vCPU と32GB の RAM を備えたハイメモリマシン タイプを使用してください。これは、Tableau Server 外部 リポジトリに使用する PostgreSQL インスタンスの推奨最小サイズですが、正確な要件は実際の要件や使用方法によって異なります。ほとんどのシナリオで良好なパフォーマンスを得るには、16 個の vCPU と128GB の RAM を備えたハイメモリインスタンス タイプをお勧めします。

- Tableau Server と外部 PostgreSQL DB インスタンス間の SSL 接続を使用した安全な通信は、必須ではありませんが、推奨されています。

- PostgreSQL インスタンスを Tableau Server クラスタ内のすべてのノードから到達可能にする必要があります。これを行う方法の 1 つは、PostgreSQL インスタンスを、Tableau Server クラスタ内のすべてのノードからアクセスするために必要なパーミッションを持つセキュリティグループのメンバーにすることです。

- PostgreSQL のバージョンは、サポートされているバージョンである必要があります。詳細については、製品の互換性でサポートされているバージョンの情報を参照してください。

- Tableau は、内部のデータベース ユーザーが内部で使用するためのパスワードを自動生成します。このパスワードは長さが 32 文字で、小文字と数字で構成されています。管理者がアクセスしたり、設定したりすることはできません。外部の PostgreSQL インスタンスでは、パスワード ポリシーを設定できる場合があります (使用しているプラットフォームによって異なります)。数字と小文字以外の文字タイプを含むポリシーを指定すると、Tableau Server が外部 リポジトリを使用するように設定したときにエラーが発生する可能性があります。

Google Cloud での PostgreSQL データベース インスタンスの作成

ステップ 1: 新しい PostgreSQL インスタンスを作成する

こちらの Google Web サイトに記載されている手順を使用して、PostgreSQL インスタンスを作成します。

16 個の vCPU と128GB の RAM を備えたハイメモリマシン タイプを使用することをお勧めします。

ステップ 2: PostgreSQL インスタンスのデータベース フラグを構成する

Tableau Server の観点では、インスタンスのほとんどのパラメーター値はデフォルトに設定できます。特別なパフォーマンスやログ取得の要件がある場合はパラメーター値を変更できますが、次のパラメーターはデフォルト値のままにして、変更しないことを強くお勧めします。

- `standard_conforming_strings`
- `escape_string_warning`

また、パフォーマンスの問題を回避するために、`work_mem` の値は少なくとも 16384 に設定することをお勧めします。

データベース フラグの詳細については、Google Web サイトのこのトピックを参照してください。

ステップ 3: PostgreSQL DB インスタンスのエンドポイントを取得する

インスタンスの準備が完了したら、このインスタンスを Tableau Server リポジトリ用に使用するように Tableau Server を構成するためのエンドポイント情報を取得します。

ステップ 4: SSL 証明書ファイルをダウンロードする

Tableau Server と外部リポジトリ間の SSL を使用した安全な通信は、必須ではありませんが、推奨されています。

Tableau Server と外部リポジトリ間の安全な接続を設定するには、この外部リポジトリを使用するように Tableau Server を構成するときに、証明書ファイルを使用する必要があります。詳細については、Google Web サイトの「SSL / TLS 証明書を構成する」を参照してください。

PostgreSQL DB の高可用性の構成

Tableau Server では、外部リポジトリの高可用性の管理やセットアップは行われません。Google Cloud には、高可用性を提供するために使用できる機能が用意されています。詳細については、Google Web サイトの「インスタンスでの高可用性の有効化と無効化」を参照してください。

PostgreSQL DB の災害復旧

災害が発生すると、PostgreSQL インスタンス用の新しい PostgreSQL データベースの設定が必要になる場合があります。他のシナリオにおいても、データベース インスタンスの問題から復旧すること

が必要にある場合があります。このようなシナリオで新しい PostgreSQL インスタンスを使用するように Tableau Server を構成するには、次のステップを実行します。

1. **新しい PostgreSQL インスタンスにバックアップを復元します。** Google Cloud プラットフォームでは、同じインスタンスに復元するか、新しいインスタンスを作成するかを選択できます。詳細については、Google Web サイトの「インスタンスの復元」を参照してください。

   新しいインスタンスを作成し、次の手順を実行して復元することをお勧めします。

2. 新しいインスタンスの場合は、新しい Azure Database for PostgreSQL インスタンスの接続情報が含まれた**新しい JSON 設定ファイルを新規作成します。** JSON 設定ファイルの作成に関する詳細については、「PostgreSQL 外部リポジトリでの Tableau Server のインストール」の**ステップ 1** を参照してください。

3. **`tsm topology external-services repository replace-host` コマンドを使用して、**新しい Azure Database for PostgreSQL インスタンスに Tableau Server を指定します。

   `tsm topology external-services repository replace-host` コマンドの詳細については、tsm topology を参照してください。

実行可能なユーザー

Tableau Server を、外部リポジトリを使用するように構成できるのは、Tableau Server 管理者のみです。PostgreSQL データベースインスタンスを作成するには、Google Cloud アカウントも必要です。

次のステップ

新規インストールの場合: PostgreSQL 外部リポジトリでの Tableau Server のインストール

外部リポジトリを使用するように既存の Tableau Server を構成する場合は、Tableau Server リポジトリの再構成を参照してください。

PostgreSQL データベースをスタンドアロン インストールで作成

バージョン 2021.2 以降、スタンドアロン インストールとして Tableau Server リポジトリを個別にホストできるようになりました。これは、AWS RDS や Azure Database などのマネージド クラウド サービスを使用する場合とは異なります。この設定は、オンプレミス、AWS EC2、Azure VM などで行うことができます。このような Tableau Server リポジトリのインストールは、スタンドアロンの外部リポジトリと呼ばれます。

外部リポジトリで使用できるホストの完全なリストについては、Tableau Server 外部リポジトリを参照してください。

このトピックでは、Tableau Server を PostgreSQL のインストールに接続し、PostgreSQL を Tableau Server 外部リポジトリとして使用するために必要な要件と構成について説明します。PostgreSQL のインストール方法については、詳しく手順を説明していません。その詳細については、PostgreSQL サイトのドキュメントを参照してください。

要件と推奨事項

- **ハードウェアの推奨事項:** CPU とストレージは要件によって異なります。小規模なインストールの場合、少なくとも 50 GB のディスク容量と、32 GB の RAM を備えたクアッドプロセッサー (または 4 つの仮想コア) のシステムが必要です。このトピックのガイダンスを確認して、バックアップとリストアに必要なディスク容量を計算してください。一般的には、大きめのハードウェア リソースで始めて、状況を見てからスケール バックすることをお勧めします。
- **ネットワーク:** PostgreSQL データベースのインスタンスを Tableau Server クラスタのすべてのノードから到達可能にします。これを行う方法のひとつは、PostgreSQL データベースのインスタンスを Tableau Server クラスタ内のすべてのノードからアクセスするために必要なパーミッションを持つセキュリティ グループのメンバーにすることです。
- **バージョン互換性:** PostgreSQL のバージョンは、ローカルにインストールする場合の Tableau Server リポジトリのバージョンと一致させる必要があります。互換性の詳細については、「製品の互換性」を参照してください。
- **セキュリティ:** Tableau Server と外部リポジトリ間の SSL を使用した安全な接続は、**必須ではありません**が、推奨されています。

Tableau Server と外部リポジトリ間に安全な接続を設定したくない場合は、暗号化されていない接続を許可するようにスタンドアロンの PostgreSQL データベースを構成する必要があります。

スタンドアロンの PostgreSQL データベース インスタンスの作成

ステップ 1: PostgreSQL をインストールして初期化する

1. PostgreSQL のドキュメントを使用して、Tableau Server の外部リポジトリとして機能する PostgreSQL データベースのインスタンスをインストールします。高可用性の要件を満たすように PostgreSQL データベースのクラスターを設定することをお勧めします。
2. uuid-ossp 拡張機能を含む contrib パッケージをインストールします。このモジュールは、Tableau Server がデータベース内のキーに使用する UUID を生成するために使用されます。
3. PostgreSQL インスタンスを初期化します。

ステップ 2: PostgreSQL インスタンスを設定する

次の 2 つの構成ファイルを使用して PostgreSQL インスタンスを構成します。

- `pg_hba`: ホストベース認証の構成ファイルです。
- `postgresql.conf`: 一般的なサーバー構成ファイルです。

既定では、これらのファイルは次の場所にあります。

`/var/lib/pgsql/12/data` (これはディストリビューションによって異なる場合があります)

スーパー ユーザーの設定

要件に合うユーザー名を選択します。管理者のユーザー名は **postgres** を使用することをお勧めします。別のユーザー名を使用する場合は、ユーザー名が **pg** で始まらないようにしてください。また、ユーザー名は、**rails**、**tblwgadmin**、**tableau**、**readonly**、または **tbladminviews** にすることはできません。

これは外部リポジトリが Tableau Server とともに正し動作するための要件です。

ネットワークとセキュリティ

すべての Tableau サーバー ノードからデータベースのインスタンスに到達できることを確認します。通常、ノードからのアクセスを許可するセキュリティグループを作成する必要があります。

データベースに関するオプション

ポートは任意のものを使用できますが、既定の 5432 のままにすることをお勧めします。

パラメーターの更新

Tableau Server の観点では、インスタンスのほとんどのパラメーター値はデフォルトに設定できます。特別なパフォーマンスやログ記録の要件がある場合はパラメーター値を変更できますが、次のパラメーターはデフォルト値のままにして、変更しないことを強くお勧めします。

- `standard_conforming_strings`
- `escape_string_warning`

また、パフォーマンスの問題を回避するために、`work_mem` の値は少なくとも `16384` に設定することをお勧めします。

リモート接続の構成

構成ファイルを更新するには、以下のステップを行います。

1. `postgresql.conf` は、既定でローカル接続のみをリッスンするように構成されています。`postgresql.conf` ファイルの接続と認証のセクションを次のように変更して、リモート接続を有効にします。

   次の行を追加して、リモート接続を許可します。

   ```
   listen_addresses = '*'
   ```

2. PostgreSQL インスタンスを再起動します。

SSL の構成

Tableau Server と外部リポジトリ間の安全な接続は、**必須ではありません**が、推奨されています。

Tableau Server と外部リポジトリ間の暗号化された接続を構成するには、以下に説明するガイダンスとステップの詳細に従ってください。

Tableau Server でスタンドアロンの PostgreSQL データベースインスタンスを使用する場合、Tableau Server への接続を確認するために、信頼されたルート認証局 (CA) が必要です。理想的には、スタンドアロン PostgreSQL インスタンスのサーバー証明書で解決可能なホスト名を指定して、Tableau Server が sslmode、**verify-full** を使用できるようにする必要があります。このモードは、PostgreSQL サーバーの証明書が信頼できる CA によって署名されていること、PostgresSQL サーバーの証明書のホスト名が PostgreSQL インスタンスへの接続に使用されるホスト名と一致することを確認します。ただし、それが不可能な場合は、sslmode、**verify-ca** は、Postgres サーバーの証明書が信頼できる CA によって署名されていることを確認するだけになります。

次の手順は、PostgreSQL サーバーでルート CA 証明書を生成するための一般的なステップを示しています。詳細については、PostgreSQL Web サイトの「SSL ドキュメント」を参照してください (リンクはバージョン 12 を指しています)。

1. 署名ルート認証局 (CA) キーを生成します。
2. ルート CA 証明書を作成します。
3. PostgreSQL サーバーの証明書と関連キー (たとえば、`server.csr` と `server.key`) を作成します。証明書のサブジェクト名は、PostgreSQL サーバーの DNS 名と一致する必要があります。サブジェクト名は、"`/CN=<private DNS name>`" 形式の **-subj** オプションで設定します。
4. ステップ 2 で作成した CA 証明書を使用して新しい証明書に署名します。
5. crt ファイルと key ファイルをデータディレクトリ (`/pgsql/<version>/data`) にコピーします。
6. `pg_hba.conf` ファイルは、データベースへの接続を制御します。次の行を追加して、リモート接続を可能にします。例:

   ```
   host all all 10.0.0.0/8 md5
   ```

7. SSL を有効にするには、`postgresql.conf` ファイルを次のように更新します。

```
ssl = on
```

接続を SSL に限定するには、*host* の代わりに *hostssl* を使用します。

高可用性とディザスター リカバリ

Tableau Server では、外部リポジトリの高可用性の管理やセットアップは行われません。これらの目的のために、PostgreSQL データベースでは、レプリケーションやログ配布などのソリューションをサポートしています。詳細については、PostgreSQL Web サイトの「高可用性のドキュメント」を参照してください。

災害が発生した場合に、新しい PostgreSQL インスタンスをセットアップする必要がある場合は、次の手順に従って、新しいインスタンスを使用する Tableau Server を構成してください。

1. **JSON 設定ファイルを新規作成する。**新しい RDS インスタンスの接続情報が含まれています。JSON 設定ファイルの作成に関する詳細については、「PostgreSQL 外部リポジトリでの Tableau Server のインストール」の**ステップ 1** を参照してください。

2. **`tsm topology external-services repository replace-host` コマンドを使用して、**Tableau Server を新しい PostgreSQL インスタンスに指定します。

   `tsm topology external-services repository replace-host` コマンドの詳細については、tsm topology を参照してください。

実行可能なユーザー

Tableau Server を、外部リポジトリを使用するように構成できるのは、Tableau Server 管理者のみです。AWS EC2 または Azure VM を使用してスタンドアロンの外部リポジトリをセットアップしている場合は、これらのプラットフォームにアクセスするためのアカウントが必要です。

PostgreSQL 外部リポジトリでの Tableau Server のインストール

このトピックでは、Tableau Server をインストールし、Tableau Server リポジトリ向けの外部サービスを使用するように構成する方法を説明します。

インストールの前に

- Tableau Server で Advanced Management を有効化している必要があります。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

- Tableau Server 環境が次のいずれかのサービスで展開されている必要があります。

  - パブリック クラウド サービス:
    - AWS クラウド サービス。
    - Azure クラウド サービス。Azure での Tableau Server インストールの詳細については、「Microsoft Azure に Tableau Server をインストールする」を参照してください。
    - Google クラウド サービスGoogle クラウド上での Tableau Server インストールの詳細については、「Google Cloud Platform 上での Tableau Server のインストール」を参照してください。
  - オンプレミス: これは、パブリック クラウドではなく、組織または会社のハードウェアで実行されている Tableau Server です。

- PostgreSQL データベースのインスタンスをインストールして準備ができている必要があります。PostgreSQL DB インスタンスのエンドポイントも必要です。

  - **AWS** 上の Tableau Server
    - Amazon RDS を使用したフルマネージド サーバー オプションの場合は、「AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成」で説明されているガイダンスに従ってください。
    - フルマネージド サーバー オプションの場合: AWS EC2 を使用して、PostgreSQL データベースをスタンドアロン インストールで作成
  - **Azure** 上の Tableau Server:
    - Azure DB を使用するフルマネージド サーバー オプションの場合は、「Azure で Azure Database PostgreSQL インスタンスを作成する」を参照してください。
    - セルフマネージド サーバー オプションの場合は、Azure VM を使用して、PostgreSQL データベースをスタンドアロン インストールで作成
  - **Google クラウド**上の Tableau Server
    - Google Cloud PostgreSQL インスタンスを使用するフルマネージド サーバー オプションについては、「Google Cloud での PostgreSQL インスタンスの作成」のガイダンスに従ってください。

        - セルフマネージドサーバーオプションの場合は、Google Cloud VM を使用して、PostgreSQL データベースをスタンドアロンインストールで作成
    - これをオンプレミスでインストールする場合は、PostgreSQL データベースをスタンドアロンインストールで作成を参照してください。PostgreSQL データベースをスタンドアロンインストールで作成

- SSL 証明書のダウンロード:

  Tableau Server と外部リポジトリ間の安全な接続は、**必須ではありません**が、推奨されています。

  Tableau Server と外部リポジトリ間の通信に SSL 接続を設定する場合は、次を実行します。

    - Amazon RDS:「SSL を使用して DB インスタンスへの接続を暗号化する」を参照してください。
    - Azure Database for PostgreSQL:「Azure Database for PostgreSQL (単一サーバー) で TLS 接続を構成する」を参照してください。
    - Google Cloud データベース:「SSL/TLS 証明書の構成」を参照してください。
    - スタンドアロン PostgreSQL データベース: データベースの SSL の構成に使用した CA 証明書は Tableau Server の最初のノードにコピーする必要があります。PostgreSQL データベースの SSL の構成については、「SSL の構成」を参照してください。

Tableau Server のインストールと構成

ステップ 1: 構成ファイルの作成

以下の構成設定を使用し、json ファイルを作成します。

```
{
 "flavor":"<flavor name>",
 "masterUsername":"<admin user name>",
 "masterPassword":"<password>",
 "host":"<instance host name>",
 "port":5432
}
```

- **flavor:** Tableau Server リポジトリで使用する外部サービスのタイプです。
  - Amazon RDS: "rds" を使用します。
  - Azure データベース: "azure" を使用します。
  - Google Cloud Database: "gcp "を使用します。
  - スタンドアロン PostgreSQL データベース: "generic" を使用します。
- **masterUsername:**
  - Amazon RDS: "rails" をユーザー名として使用します。これは RDS インスタンスを作成するときに指定したユーザーです。

    masterUsername には "rails" を使用する必要があります。外部リポジトリが Tableau Server で正しく動作するには、この設定が必要です。
  - Azure Database、Google Cloud PostgresSQL インスタンス、またはスタンドアロン PostgreSQL データベース: 要件を満たすユーザー名を選択します。管理者のユーザー名は **postgres** を使用することをお勧めします。別のユーザー名を使用する場合は、ユーザー名が **pg** または **azure** で始まらないようにしてください。また、ユーザー名は、**rails**、**tblwgadmin**、**tableau**、**readonly**、または **tbladminviews** にすることはできません。
- **masterPassword**: PostgreSQL DB インスタンスを作成するときに指定したものと同じパスワードです。
- **host:** PostgreSQL データベース インスタンスのエンドポイントです。
- **port:** PostgreSQL DB インスタンスを作成するときに指定したデータベース ポートです。

ステップ 2: Tableau Server のインストールと外部リポジトリの構成

TSM CLI を使用:

1. TSM のインストールと初期化: このトピックの TSM CLI の使用 の手順に従いステップ 1 ~ 5 を完了し、セットアップ プログラムを実行して TSM をインストールしてください。
2. Tableau Server のライセンス認証と登録: ライセンス認証手順で Tableau Server キーと Advanced Management キーを指定します。次のコマンドを 2 回実行する必要がありま

す。まずは Tableau Server のプロダクトキーと、次に Advanced Management プロダクトキーと共に実行します。

```
tsm licenses activate -k <product key>
```

3. 初期ノード設定の構成: このトピックの TSM CLI の使用 の手順に従い、初期ノード設定を構成します。

   重要! 初期ノードを構成するときに、**Tableau Server の初期化と開始**の手順を実行しないでください。初期ノード設定の構成トピックの他のステップを完了したら、このページに戻り、残りの指示に従います。

4. 次のコマンドを使用し、Tableau Server で外部リポジトリを使用するように構成します。

   - 前の手順で作成した json ファイルを使用し、外部リポジトリの設定を指定します。

     ```
     tsm topology external-services repository enable -f
     <filename>.json -c <ssl certificate file>
     ```

     json ファイルは、構成設定の最初の手順で作成したファイルです。

     > **注:** SSL 証明書は、Tableau Server と外部リポジトリ間で暗号化された接続を使用している場合にのみ必要です。これが必須ではない場合は、-no-ssl オプションを指定する必要があります。その場合、tsm コマンドは次のようになります。
     > ```
     > tsm topology external-services repository enable -f
     > <filename>.json --no-ssl
     > ```

   - 変更を適用します。

     ```
     tsm pending-changes apply
     ```

ステップ 3: tsm による初期化の完了

Tableau Server を初期化して起動するには、以下を実行します。

```
tsm initialize --start-server --request-timeout 1800
```

ステップ 4: インストールの完了

管理者アカウントの追加、インストールを完了します。

実行可能なユーザー

Tableau Server 管理者は、Tableau Server のインストールと構成を行うことができます。

Tableau Server リポジトリの再構成

Tableau Server は、ローカル リポジトリまたは外部 リポジトリのどちらかを使用するように構成できます。このトピックでは、次のいずれかのオプションを使用して既存の Tableau Server を再構成するのに必要な手順を説明します。

- ローカルの Tableau Server リポジトリを外部 リポジトリに移動し、Tableau Server を外部 リポジトリを使用するように構成します。
- 外部の Tableau Server リポジトリをローカルの Tableau Server インストールに移動し、Tableau Server をローカル リポジトリを使用するように構成します。つまり、Tableau Server リポジトリが Tableau Server と同じマシンにインストールされます。

これらのオプションと外部 リポジトリの詳細については、Tableau Server 外部 リポジトリを参照してください。

リポジトリをローカルから外部に移動

ローカル リポジトリから外部 リポジトリに移行するには、Tableau Server を停止する必要があります。

Tableau Server リポジトリをローカルから外部に移動するには、次の手順を実行します。

1. Advanced Management プロダクト キーがライセンス認証されていなければ、これを Tableau Server 上でライセンス認証します。外部 リポジトリを使用して Tableau Server を構成するには、Advanced Management ライセンスが必要です。
2. Amazon PostgreSQL DB インスタンスを外部 リポジトリとして使用するように構成します。
    1. Amazon: AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成

2. Azure データベース: Azure で Azure Database PostgreSQL インスタンスを作成する
3. Google Cloud データベース: Google Cloud での PostgreSQL インスタンスの作成
4. スタンドアロン PostgreSQL インスタンス: PostgreSQL データベースをスタンドアロン インストールで作成。

3. 以下の構成設定を使用し、json ファイルを作成します。

```
{
 "flavor":"<flavor name>",
 "masterUsername":"<admin user name>",
 "masterPassword":"<password>",
 "host":"<instance host name>",
 "port":5432
}
```

- **flavor:** Tableau Server リポジトリで使用する外部サービスのタイプです。
  - Amazon RDS: "rds" を使用します。
  - Azure データベース: "azure" を使用します。
  - Google Cloud Database: "gcp "を使用します。
  - スタンドアロン PostgreSQL データベース: "generic" を使用します。

- **masterUsername:**

  - **Amazon RDS:** "rails" をユーザー名として使用します。これは RDS インスタンスを作成するときに指定したユーザーです。

    masterUsername には "rails" を使用する必要があります。外部リポジトリが Tableau Server で正しく動作するには、この設定が必要です。

  - **Azure Database、Google Cloud インスタンス、スタンドアロン PostgreSQL インスタンス:** 要件を満たすユーザー名を選択します。管理者のユーザー名は **postgres** を使用することをお勧めします。別のユーザー名を使用する場合は、ユーザー名が **pg** または **azure** で始まらないようにしてください。また、ユーザー名は、**rails**、**tblwgadmin**、**tableau**、**readonly**、または **tbladminviews** にすることはできません。「postgres」以外のユーザー名を選択すると、バックアップを復元するたびに、リポジトリの最初の「postgres」システム ユーザーにサフィックス「_bak」が追加されます (「postgres_bak」など)。

- **masterPassword**: PostgreSQL DB インスタンスを作成するときに指定したものと同じパスワードです。
- **host:** PostgreSQL データベース インスタンスのエンドポイントです。
- **port:** PostgreSQL DB インスタンスを作成するときに指定したデータベース ポートです。

4. 以下の TSM CLI コマンドを実行し、Tableau Server で外部リポジトリを使用するように構成します。

   ```
   tsm topology external-services repository enable -f file.json -
   c <ssl certificate file>.pem
   ```

   > **注:** SSL 証明書は、Tableau Server と外部リポジトリ間で暗号化された接続を使用している場合にのみ必要です。これが必要ない場合は、`--no-ssl` オプションを指定する必要があります。その場合、tsm コマンドは次のようになります。
   > `tsm topology external-services repository enable -f <filename>.json --no-ssl`

   json ファイルは、構成設定の最初の手順で作成したファイルです。SSL 証明書ファイルは、このトピックの説明に従ってダウンロードできます。

   上記のコマンドを実行すると、ローカル リポジトリが新しい外部 PostgreSQL DB インスタンスに移行されます。

リポジトリを外部からローカルに移動

Tableau Server リポジトリを外部からローカル インストールに移動するには、次のステップを実行します。

1. 次の TSM CLI コマンドを実行し、リポジトリを特定のノードに移動します。

   ```
   tsm topology external-services repository disable -n nodeN
   ```

2. リポジトリで高可用性を設定する場合、リポジトリを2番目のノードにインストールします。詳細については、例: 3 ノードHA クラスタのインストールおよび構成を参照してください。

> **注:** リポジトリを2番目のノードにインストールするには、まず前の手順で説明したコマンドを実行する必要があります。最初の手順により、外部リポジトリがローカル リポジトリに移行されます。その後、Tableau Server 上の2番目のノードにリポジトリをインストールできます。

実行可能なユーザー

Tableau Server 管理者は、外部リポジトリを再構成することができます。Amazon または Azure で PostgreSQL データベース インスタンスを作成するためのアクセス権も必要です。

新しいメジャー バージョンの PostgreSQL を外部リポジトリとして使用する Tableau Server のアップグレード

PostgreSQL メジャー バージョンに関する Tableau Server の要件に変更があった場合、Tableau Server を正常にアップグレードするために従う必要のある手順がいくつかあります。たとえば、Tableau Server 2020.4 では、PostgreSQL バージョン 12 をリポジトリとして使用する必要があります。バージョン 2020.4 より前の Tableau Server では PostgreSQL バージョン 9.x を使用していましたが、PostgreSQL のメジャー バージョンを変更する必要があります。Tableau Server を以前のバージョンから 2020.4 以降にアップグレードする場合は、次のセクションで説明する手順に従ってアップグレードします。

このトピックでは、PostgreSQL と Tableau サーバーの製品互換性についても説明します。

アップグレードする前に

既存の Amazon RDS 上の PostgreSQL DB インスタンス、または既存の PostgreSQL DB 用の Azure Database において、PostgreSQL のバージョンを更新するためにインプレース アップグレードを実行することはできません。代わりに、新しいインスタンスを作成して、アップグレード中に Tableau Server に新しいインスタンスを指定する必要があります。新しいインスタンスを作成して、アップグレードを準備するには、次の情報を使用します。

1. PostgreSQL データベースの新しいインスタンスを作成します。
    1. AWS Relational Database Service (RDS) での PostgreSQL DB インスタンスの作成
    2. 詳細については、Azure で Azure Database PostgreSQL インスタンスを作成するを参照してください。
    3. Google Cloud での PostgreSQL インスタンスの作成
    4. PostgreSQL データベースをスタンドアロン インストールで作成
2. SSL 接続は、**必須ではありません**が、推奨されています。Tableau Server と外部リポジトリ間の通信に SSL 接続を設定する場合は、次を実行します。
    - **Amazon RDS:**「SSL を使用して DB インスタンスへの接続を暗号化する」を参照してください。
    - **Azure Database:**「PostgreSQL 用 Azure Database の TLS 接続の構成」を参照してください。
    - **Google Cloud インスタンス:**「SSL/TLS 証明書の構成」を参照してください。
    - **スタンドアロン PostgreSQL インスタンス:**「SSL の構成」を参照してください。

3. ステップ 1 で作成した新しいインスタンスの構成ファイルを作成します。

    以下の構成設定を使用し、json ファイルを作成します。

    ```
    {
     "flavor":"<flavor name>",
     "masterUsername":"<admin user name>",
     "masterPassword":"<password>",
     "host":"<instance host name>",
     "port":5432
    }
    ```

    - **flavor:** Tableau Server リポジトリで使用する外部サービスのタイプです。
        - Amazon RDS: "rds" を使用します。
        - Azure データベース: "azure" を使用します。
        - Google Cloud Database: "gcp "を使用します。
        - スタンドアロン PostgreSQL データベース: "generic" を使用します。
    - **masterUsername:**

- **Amazon RDS**: "rails" をユーザー名として使用します。これは RDS インスタンスを作成するときに指定したユーザーです。

  masterUsername には "rails" を使用する必要があります。外部リポジトリが Tableau Server で正し く動作するには、この設定が必要です。

- **Azure Database、Google Cloud インスタンス、または PostgreSQL インスタンス:** 要件を満たすユーザー名を選択します。管理者のユーザー名は **postgres** を使用することをお勧めします。別のユーザー名を使用する場合は、ユーザー名が **pg** または **azure** で始まらないようにしてください。また、ユーザー名は、**rails**、**tblwgadmin**、**tableau**、**readonly**、または **tbladminviews** にすることはできません。

- **masterPassword:** PostgreSQL DB インスタンスを作成するときに指定したものと同じパスワードです。

- **host:** PostgreSQL データベース インスタンスのエンドポイントです。

- **port:** PostgreSQL DB インスタンスを作成するときに指定したデータベース ポートです。PostgreSQL の既定のポートは 5432 です。

Tableau Server のアップグレード

> **注:** Tableau Server で外部リポジトリを使用している場合は、コマンドライン オプションを使用して Tableau Server をアップグレードする必要があります。

以下は、アップグレード スクリプトの実行時に外部リポジトリ パラメーターを指定する方法の概要です。

Tableau Server のアップグレードプロセスの詳細な手順については、「以前のバージョンからのアップグレード」を参照してください。

1. Tableau Server で、管理者としてコマンドプロンプトを開きます。

   > **注:** セットアッププログラムは新しいインストールのパスを更新するため、新しいコマンドウィンドウを開く必要があります。

2. 新しいインストール場所の scripts フォルダーに移動します。

   既定のディレクトリは次のとおりです。

   ```
   /opt/tableau/tableau_server/packages/scripts.<version_code>/
   ```

3. アップグレードスクリプトを実行し、構成ファイルとSSL 証明書を指定します。

   ```
   upgrade-tsm --external-repository-config-file=<json config
   file> --external-repository-cert-file=<SSL certificate file>
   ```

製品の互換性

以下の表は、Tableau Server がサポートする PostgreSQL のバージョンを示しています。この表を使用して、外部リポジトリー用にインストールする PostgreSQL のバージョンを判別します。

Amazon RDS のサポート: Amazon RDS を使用する外部リポジトリは、Tableau Server バージョン 2019.3 以降でサポートされています。
Azure Database のサポート: Azure Database インスタンスを使用する外部リポジトリは、Tableau Server バージョン 2020.4 以降でサポートされています。

Google Cloud のサポート: Google Cloud SQL インスタンスを使用する外部リポジトリは、Tableau Server バージョン 2021.4 以降でサポートされています。
スタンドアロン PostgreSQL インスタンスのサポート: PostgreSQL データベースのスタンドアロンインストールを使用する外部リポジトリは、Tableau Server バージョン 2021.2 以降でサポートされています。

Google Cloud Platform のサポート: Google Cloud Platform で PostgreSQL インスタンスを使用する外部リポジトリは、Tableau Server バージョン 2021.4 以降でサポートされています。

Tableau Server のバージョン	**PostgreSQL** の互換性のある最低バージョン
2022.3.0 ~ 2022.3.7 2023.1.0 ~ 2023.1.4	13.7
2022.1.17 - 2022.1.19 2022.3.8 ~ 2022.3.19 2023.1.5 ~ 2023.1.15 2023.3.0 ~ 2023.3.8	13.11
2022.3.20 ~ 2022.3.x 2023.1.16 ~ 2023.1.x 2023.3.9 ~ 2023.3.x	13.14
2019.3 ~ 2019.3.8	15.6
2024.2.9 ~ 2024.2.13 2025.1.0 ~ 2025.1.6	15.10
2024.2.14 以降 2025.1.7 以降	15.13

レガシーとの互換性

次の表には、2 つの列が含まれています。

1. PostgreSQL バージョン (Tableau サーバーに同梱) は、ローカル リポジトリ用に Tableau Server とともにインストールされるバージョンです。
2. 外部リポジトリでサポートされている PostgreSQL のすべてのバージョン。

注:
-バージョン 10 より前の PostgreSQL バージョンの場合、最初の 2 つの数字がメジャー バージョンを表し、最後の数字がマイナー バージョンを表しています。たとえば、バージョン 9.4.1 では、9.4 はメジャー バージョンを示し、.1 はマイナー バージョンを示します。
-PostgreSQL バージョン 10 以降では、最初の数字がメジャー バージョンを表し、最後の数字がマイナー バージョンを表しています。たとえば、バージョン 11.1 では、11 はメジャー バージョンを示し、.1 はマイナー バージョンを示します。
-Tableau Server に同梱の Postgres バージョンとメジャー バージョンが同じで、マイナー バージョンがより新しい PostgreSQL バージョン、使用が明示的に許可されている古い Postgres SQL バージョンは、外部リポジトリで使用できます。

Tableau Server バージョン	PostgreSQL バージョン (Tableau サーバーに同梱)	外部リポジトリでサポートされている PostgreSQL の代替バージョン
2019.3 - 2019.3.3	9.6.11	9.6.x (x が 12 以上)
2019.3.4 - 2019.3.10	9.6.15	9.6.x (x が 16 以上)
2019.3.11 - 2019.3 14	9.6.17	9.6.15 または 9.6.x (x が 18 以上)
2019.4 - 2019.4.1	9.6.14	9.6.x (x が 15 以上)
2019.4.2 - 2019.4.6	9.6.15	9.6.x (x が 16 以上)
2019.4.7 - 2019.4.13	9.6.17	9.6.15 または 9.6.x (x が 18 以上)
2020.1 - 2020.1.6	9.6.15	9.6.x (x が 16 以上)
2020.1.7 -	9.6.17	9.6.15 または 9.6.x (x が 18 以上)

2020.3.2		
2020.4 - 2021.4	12.8	12.8 または 12.x (x が 9 以上)
2022.1 以降	13.3	13.3 または 13.x (x が 4 以上)

実行可能なユーザー

Tableau Server 管理者が、Tableau Server のアップグレードと構成を行うことができます。

RDS インスタンスのアップグレード

Tableau リポジトリのホストに使用している現在の RDS インスタンスがパフォーマンスのボトルネックとなっている場合は、RDS インスタンスをより大きなサイズにアップグレードできます。このトピックでは、RDS インスタンスのアップグレードに使用できる手順について説明します。

1. Tableau Server データのバックアップ。
2. Tableau Server を停止します。

   ```
   tsm stop
   ```

3. サーバーがシャットダウンしたことを確認したら、AWS 管理コンソールにサインインして Amazon RDS コンソールを開きます。

   https://console.aws.amazon.com/rds/.

4. ナビゲーション ペインで **[データベース]** を選択し、変更する DB インスタンスを選択します。
5. **[変更]** を選択します。[DB インスタンスの変更] ページが表示されます。
6. DB インスタンス クラスを必要に応じて設定し、RDS インスタンスを変更します。
7. **[直ちに適用]** を選択して、変更がすぐに適用されていることを確認します。詳細については、AWS ドキュメント サイトの「PostgreSQL データベース エンジンを実行している DB インスタンスの変更」を参照してください。
8. AWS コンソールで RDS インスタンスのステータスを監視します。しばらく時間がかかる場合がありますが、ステータスが [利用可能] と表示されたら、Tableau Server を起動して通常の

操作を再開できます。

```
tsm start
```

実行可能なユーザー

Amazon RDS アカウントへのアクセスを持つ Tableau Server 管理者は、Amazon RDS インスタンスのアップグレードに必要なすべての手順を実行することができます。

ノードロールによるワークロード管理

ノードロールを使用すると、Tableau Server インストールで特定のタイプのワークロードが処理される場所を構成できます。ノードロール機能を使用すると、リソースを特定のワークロード専用にしたり、ワークロードに合わせて拡張したりできます。バックグラウンダーとファイル ストアのノードロールを構成できます。

バックグラウンダー ノードロールは、ノードで実行する必要があるバックグラウンド タスクのタイプを指定しますが、ファイル ストア ノードロールは、ノードで実行する必要がある抽出のワークロードのタイプを指定します。どちらのノードロールも、ノードレベルで指定されます。これらのノードロールは独立して機能して選択したワークロードを最適化できますが、2 つのノードロールを組み合わせて使用すると、サーバー ノードを、選択したワークロードを優先的に実行し、抽出負荷の大きいワークロードでパフォーマンスを最適化するための専用のノードにすることができます。この組み合わせについては、後ほど「ファイル ストア ノードロール」セクションで詳しく説明します。

バックグラウンダー ノードロール

バックグラウンダープロセスでは、抽出の更新、サブスクリプション、フロー タスク、「今すぐ実行」タスク、*tabcmd* から開始したタスクなどの Tableau Server タスクを実行します。これらのタスクをすべて実行すると、多くのマシン リソースが使用される可能性があります。クラスタ内に複数のバックグラウンダー ノードがある場合は、バックグラウンダー ノードロール機能を使用してバックグラウンダーがノードで実行できるタスクのタイプを指定することにより、バックグラウンダー ワークロードを管理できます。

現在、この構成オプションは TSM CLI コマンド経由でのみ使用でき、マルチノードクラスタでのみ有用です。ノードが 1 つしかない場合、バックグラウンダーは既定ですべてのタスクを実行するように設定され、これは変更できません。

バックグラウンダー ノード ロールの使用

バックグラウンダー ノード ロール機能は、Tableau Server インストールで特定のタイプのバックグラウンダー ワークロードが処理される場所をより細かく制御して、リソースを特定のワークロード専用にしたりワークロードに合わせて拡張できるようにすることを意図しています。

たとえば、展開で抽出が大量に実行され、ユーザーが多くの抽出更新や暗号化ジョブを実行している場合は、あるノードを抽出更新専用にすると便利です。サブスクリプションの場合も同様で、Tableau Server インストールで大量のサブスクリプションが処理される場合、他のジョブがサブスクリプション用のリソースを取得しないようにするには、あるノードをサブスクリプション専用にすることができます。これらの場合は、他のバックグラウンダー ノードを抽出更新やサブスクリプション以外のワークロード専用にすることもできます。

高可用性をサポートするため、複数のノードをそれぞれ特定のワークロード専用にすることをお勧めします。たとえば、あるノードを抽出更新専用にする場合は、抽出更新ワークロードを処理する 2 番目のノードも構成するとよいでしょう。このようにすると、抽出更新専用のノードが使用できなくなった場合でも、抽出更新を他のノードで引き続き処理できます。

構成オプション

構成	ジョブ
all-jobs (既定値)	すべての Tableau Server ジョブ。
flows	フロー実行ジョブ。
no-flows	フロー以外のすべてのジョブ。
extract-refreshes	次の処理用に作成されたジョブ。 増分更新、完全更新、フロー出力が作成する抽出を含むすべての抽出の暗号化と復号化。

subscriptions	サブスクリプション ジョブ。
system	他の Tableau Server プロセスと連携するシステム メンテナンス ジョブ。たとえば、クラッシュしたジョブのクリーニング、データベース イベントの取得、Active Directory の同期などがあります。
extract-refreshes-and-subscriptions	抽出更新、フロー出力が生成する抽出を含むすべての抽出の暗号化と復号化、およびサブスクリプション ジョブ。
no-extract-refreshes	抽出更新、フロー出力から作成された抽出を含むすべての抽出の暗号化と復号化以外のすべてのジョブ。
no-subscriptions	サブスクリプション以外のすべてのジョブ。
no-extract-refreshes-and-subscriptions	抽出更新、フロー出力から作成された抽出を含むすべての抽出の暗号化と復号化、サブスクリプション以外のすべてのジョブ。
no system	システム メンテナンス ジョブを除くすべてのジョブ。

tsm コマンドを使用してノード ロールを設定する方法の詳細については、tsm topologyを参照してください。

注: ノードの役割を構成するには、サーバーを再起動する必要があり、いくらかのダウンタイムが必要になります。詳細については、tsm pending-changesを参照してください。

ライセンスの要件

フロー、抽出更新、サブスクリプションなどの特定のタイプのタスクのみを実行するようにノードを構成するには、Tableau Server で次のいずれかのライセンスがアクティブ化されている必要があります。

- フローを実行するようにノードを構成するには、有効なデータ管理 ライセンスがサーバーで有効になっており、Tableau Prep Conductor がそのノード上で実行されている必要があります。Tableau Prep Conductor の詳細については、Tableau Prep Conductorを参照してください。

- 抽出更新、サブスクリプション、および抽出やサブスクリプションに関連するあらゆる組み合わせを実行するようにノードを設定するには、Tableau Server 上で Advanced Management 機能を有効化している必要があります。ライセンスの有効期限が切れた場合、または非アクティブになっている場合は、サーバー構成に変更を加えるたびにエラーが表示されます。Advanced Management の詳細については、Tableau Server 上の Tableau Advanced Management についてを参照してください。

重要!
フロー、抽出更新、サブスクリプションはコストがかかり、リソースを大量に消費する可能性がありますが、専用のリソースが必要なジョブはこれだけではありません。**all-jobs** グループには、ワークブックのサムネイル生成など、バックグラウンダーが実行するさまざまなシステム ジョブがあります。抽出更新、サブスクリプション、フロー以外のジョブを実行するノードに十分なマシン リソースがあることを確認してください。

TSM コマンドを使用したノード ロールの構成の詳細については、tsm topology set-node-roleを参照してください。

考慮事項

バックグラウンダー ノード ロールを構成する際は、以下のようないくつかの規則を考慮する必要があります。

- ノードに対して一度に設定できるノード ロール構成は 1 つだけです。1 つのノードで複数のノード ロールを構成することはできません。

- ノード ロールを構成するには、そのノードに少なくとも 1 つのバックグラウンダー プロセスが存在する必要があります。

- バックグラウンダー ノードが 1 つしかない場合は、このノードによってすべてのジョブを実行するように構成する必要があります。これは既定の構成であり、追加のライセンスは必要ありま

せん。

- 複数のバックグラウンダー ノードがある場合は、その組み合わせによってすべてのジョブを処理するように構成する必要があります。これは、次の方法で実現できます。

    - all-jobs オプションを使用して、すべてのジョブを実行するようにいずれかのノードを構成します。これは最も簡単な方法です。

    - いずれかのノードで以下のいずれかの例外構成を使用します。

        - no-flows
        - no-subscriptions
        - no-extract-refreshes
        - no-extract-refreshes-and-subscriptions

たとえば、3 つのバックグラウンダーがあるクラスタでは、1 台のノードをフローの実行用に構成し、1 台のノードをサブスクリプションと抽出更新の実行用に構成し、1 台のノードをフロー、サブスクリプション、抽出更新を除くすべてのジョブの実行用に構成できます。

注: フロー、フロー以外のすべてのジョブ、またはすべてのジョブの実行用のノード ロールを指定する機能は、2019.1 で導入されました。

ファイル ストア ノード ロール

Tableau Server ファイル ストアは、抽出のストレージを管理します。抽出に依存するワークロードには、大きく分けて 3 つのカテゴリーがあります。

抽出のワークロード	実行サービス
更新	バックグラウンダー
クエリ	データ エンジン
バックアップ/復元	バックアップ/復元

ファイル ストア ノード ロールの管理とバックグラウンダー ノード ロールの管理を組み合わせると、サーバー管理者はサーバー ノードを、選択したワークロードを優先的に実行し、抽出負荷の大きいすべてのカテゴリーのワークロードでパフォーマンスを最適化する専用のノードとすることができます。

スタンドアロンのデータエンジン ノードのみを持つトポロジを使用し、抽出クエリワークロードを実行する専用のノードとすることができます。詳細については、抽出クエリの負荷が大きい環境用に最適化を参照してください。とはいえ、これはバックグラウンダー ノードによって実行される抽出更新ワークロードを犠牲にして成り立つものです。トポロジベースの分離アプローチを使用すると、バックグラウンダー ノードのいずれもファイル ストアがなく、すべての抽出更新トラフィックがネットワーク経由で流れるため、抽出更新の負荷が大きいバックグラウンダー ワークロードは遅くなる可能性があります。

ファイル ストア ノード ロールの構成オプションを使用すると、抽出クエリを処理する特定のサーバー ノードを、それが可能なサーバー ノードのリストから優先的に選択するように指定できます。これは、サーバー管理者がバックグラウンダー サーバー ノード上でファイル ストアを有効にするのを許可し、これらのノードで抽出クエリを実行できないようにすることで、バックアップや抽出更新などのワークロードを高速化するのに役立ちます。この機能は、抽出負荷の大きいクエリワークロードと更新ワークロードがあり、抽出クエリと更新パフォーマンスを最適化する場合に便利です。

抽出更新ワークロードとバックアップまたは復元ワークロードを最適化するためのガイドライン

専用のデータエンジン ノードを含むトポロジから開始します 抽出クエリの負荷が大きい環境用に最適化を参照)。

注: 以下の図と手順では、ノード1は初期ノード、ノード2は追加ノード1、ノード3は追加ノード2、ノード4は追加ノード3です。

Process	Initial Node	Additional Node 1	Additional Node 2	Additional Node 3
Cluster Controller	✓	✓	✓	✓
Gateway	✓	✓		
Application Server	✓	✓		
VizQL Server	✓ ✓	✓ ✓		
Cache Server	✓ ✓	✓ ✓		
Search & Browse	✓	✓		
Backgrounder	✓ ✓ ✓ ✓	✓ ✓ ✓ ✓		
Data Server	✓ ✓	✓ ✓		
Data Engine	✓	✓	✓	✓
File Store			✓	✓
Repository	✓	✓		

トポロジ 1 - 専用データエンジン ノード

1. ファイル ストアをノード1 に追加します。

   ```
   tsm topology set-process -n node1 -pr filestore -c 1
   ```

2. 抽出クエリのワークロードを優先的に実行するようにノード3 とノード4 を指定します。

   ```
   tsm topology set-node-role -n node3, node4 -r extract-queries
   ```

3. 非抽出更新のワークロードを優先的に実行するようにノード1 を指定します。

   ```
   tsm topology set-node-role -n node1 -r extract-refreshes
   ```

4. 非抽出更新のワークロードを優先的に実行するようにノード2 を指定します。

   ```
   tsm topology set-node-role -n node2 -r no-extract-refreshes
   ```

5. 保留中の変更を適用します。

   ```
   tsm pending-changes apply
   ```

Process	Initial Node	Additional Node 1	Additional Node 2	Additional Node 3
Cluster Controller	✓	✓	✓	✓
Gateway	✓	✓		
Application Server	✓	✓		
VizQL Server	✓ ✓	✓ ✓		
Cache Server	✓ ✓	✓ ✓		
Search & Browse	✓	✓		
Backgrounder	✓ ✓ ✓ ✓	✓ ✓ ✓ ✓		
Data Server	✓ ✓	✓ ✓		
Data Engine	✓	✓	✓	✓
File Store	✓		✓	✓
Repository	✓	✓		

トポロジ 2 - 余分のファイル ストア ノード

注: Tableau Server 展開では、既存ノードにファイル ストア ロールを追加すると、新しいファイル ストアを同期しながら、すべてのファイル ストア ノード間のネットワーク I/O が一時的に増加します。この動作期間は、ファイル ストア上のデータ量とネットワーク帯域幅の容量によって変わります。同期のステータスは、TSM Web GUI を使用して監視できます。展開に複数のファイル ストアを追加する場合、それを連続で追加し、各ファイル ストアの追加の間に最初の同期が完了するのを待機することをお勧めします。

抽出クエリワークロード管理の微調整

ユーザーが抽出ベースの Viz をインタラクティブに表示しているときに、メール登録とメトリクス アラートの抽出クエリがそれと同時に実行されると、Viz の読み込み時間が通常よりも遅くなることがあります。次のノード ロールを使用して、これらのワークロードの優先順位を微調整します。

使用するノード ロール	抽出クエリ ワークロードのタイプ	例
`extract-queries`	スケジュール済み	メール登録とメトリクス アラート
`extract-queries-interactive`	インタラクティブ	抽出ベースの Viz を表示するユーザー

サーバーの展開でメール登録とメトリクスアラートの増加が確認される場合は、ノードを追加して extract-queries ノードロールを割り当てることができます。そうすると、メール登録とメトリクスアラートを処理するためのノードの可用性が増大します。

サーバーの展開で抽出ベースの Viz を表示するユーザーの増加が確認される場合は、ノードを追加して extract-queries-interactive ノードロールを割り当てることができます。そうすると、ノードでインタラクティブな抽出クエリに優先順位を付けて、抽出ベースの Viz の読み込み時間を短縮できます。extract-queries-interactive ノードロールは、厳密に分離されるものではなく、優先されるロールです。つまり、クエリは extract-queries-interactive ノードロールが割り当てられているノードにルーティングされます。extract-queries-interactive ロールを持つ複数のノードがある場合、クエリはノードの正常性に基づいてルーティングされます。

たとえば、ノードを追加し、extract-queries-interactive ワークロードを優先的に実行するように指定します。

- `tsm topology set-node-role -n node4 -r extract-queries-interactive`

構成オプション

構成	ジョブ
all-jobs (既定値)	すべての Tableau Server ジョブ。
extract-queries	抽出クエリ用に作成されるジョブ。選択されたノードは all-jobs として実行され、抽出クエリの処理を優先して実行します。
extract-queries-interactive	抽出クエリ用に作成されるジョブ。選択されたノードは all-jobs として実行され、インタラクティブな抽出クエリの処理を優先して実行します。これらのクエリは、ユーザーが画面を見ていて、抽出ベースのダッシュボードの読み込みを待機しているときに実行されるクエリのようなものです。これは高度な設定であるため、クラスターのサブスクリプションおよびアラートジョブのワークロードの負荷が大きく、スケジュールされたロードと同じ時間に実行される Viz の読み込み時間のパフォーマンスが低下する場合にのみ使用してください。

TSM コマンドを使用したノードロールの構成の詳細については、tsm topology set-node-roleを参照してください。

ライセンスの要件

ノードを抽出クエリの実行用に構成するには、有効な Advanced Management ライセンスが Tableau Server でアクティブ化されている必要があります。

ノードロールの表示方法

次のコマンドを使用して、Tableau Server で現在構成されているノードロールを表示します。

```
tsm topology list-nodes -v
```

実行可能なユーザー

Tableau Server 管理者は、ノードロールを構成し、必要なプロダクトキーを認証することができます。

Tableau Server の独立したゲートウェイ

このトピックでは、Tableau Server の独立したゲートウェイの概要について説明します。

独立したゲートウェイは、Apache httpd に基づくリバースプロキシ サーバーおよびロードバランサーです。これには、Tableau Server クラスタの内部にある Tableau Server ゲートウェイプロセスと同じ Apache httpd バージョンを使用しますが、ネットワーク DMZ での展開に適しています。独立したゲートウェイは Tableau Server の一部であるため、TSM によって管理されます。そのため、個別に構成する必要はありません。

その構成には、外部からアクセス可能な Tableau Server コンポーネントのトポロジーに関する完全なナレッジが含まれており、クラスタのトポロジーが変更されると更新されます。独立したゲートウェイの単純なインストール プロセスの後、Tableau サービス マネージャー (TSM) の構成アイテムを使用して構成の選択が一元的に行われます。

この新しい機能を使用して、Tableau Server を次の 2 つの方法で構成できるようになりました。

- 自己完結型のインストールで Tableau Server をインストールします。リバースプロキシは、個別にインストールおよび管理する必要があります。

- Tableau Server をインストールして、独立したゲートウェイを Tableau Server (バージョン 2022.1 以降) によって管理されるリバースプロキシとしてインストールします。

独立したゲートウェイを使用する理由

独立したゲートウェイを使用すると、個別のリバースプロキシをインストールする場合と比べて、次の利点があります。

- **完全なサポート:** 独立したゲートウェイは Tableau Server インストールの一部であるため、Tableau によって完全にサポートされています。
- **Tableau Server 対応:** 別のリバースプロキシを使用する場合は、Tableau Server のトポロジが変更されたときに更新する必要があります。独立したゲートウェイは、外部から呼び出し可能な Tableau Server コンポーネントをすべて認識しているため、これらが変更されると更新されます。

独立したゲートウェイの管理

ライセンス管理

独立したゲートウェイを使用するには、Tableau Server で Advanced Management 機能が有効になっている必要があります。独立したゲートウェイのノードで発行されるライセンスはありません。詳細については、「Tableau Server 上の Tableau Advanced Management について」を参照してください。Advanced Management キーがアクティブになっていない、またはライセンスの有効期限が切れている場合は、次のような動作になります。

- 独立したゲートウェイを使用して Tableau Server を構成しようとすると、失敗します。
- 既に独立したゲートウェイを使用していて、Advanced Management ライセンスの有効期限が切れている場合は、サーバーは再起動に失敗します。

バックアップと復元

独立したゲートウェイを使用することによるバックアップまたは復元への影響はありません。Tableau Server のバックアップまたは復元には、独立したゲートウェイの情報や構成は含まれません。バックアップを使用して Tableau の新しいインストールを作成する場合は、新しい Tableau Server のインストールに対して、独立したゲートウェイを個別にインストール、構成、および有効にする必要があります。

高可用性に関する考慮事項

独立したゲートウェイの複数のインスタンスをインストールすると、リバースプロキシで堅牢な高可用性を実現できます。Tableau にアクセスするクライアントセッションが多数ある場合は、独立したゲートウェイのノード数を増やすこともできます。

すべてのノードが、同じオペレーティング システムを実行していなければなりません。詳細については、「マルチノード サーバーのソフトウェア要件」を参照してください。

トポロジ

Tableau Server を独立したゲートウェイを使用して構成する場合、個別のリバースプロキシを設定して構成する必要はありません。独立したゲートウェイは、外部サービスとして TSM ステータスページに表示されます。

> **注:** 複数の独立したゲートウェイのノードをインストールしている場合でも、CLI のステータスページとステータス出力には、独立したゲートウェイの単一のインスタンスのみが表示されます。

次へ

Tableau Server と独立したゲートウェイのインストール

Tableau Server と独立したゲートウェイのインストール

このトピックでは、Tableau Server Independent Gateway をインストールする手順を説明します。

このプロセスに従うと、バックエンドに展開されている Tableau Server に直接接続する独立したゲートウェイ構成になります。*直接*接続モードと*リレー*接続モードの詳細については、トピック「Independent Gateway を使用した Tableau Server の構成」を参照してください。

前提条件

- 少なくとも 2 コア (4 vCPU)、8 GB の RAM、および 100 GB の空きディスク領域を備えた専用サーバーが必要です。
- Tableau Server 2022.1 以降を使用する必要があります。

- Tableau Server の独立したゲートウェイのインストーラーは、Tableau Server のバージョンと一致するメジャー バージョン (たとえば、**2022.1**) を使用する必要があります。メンテナンス バージョン (たとえば、2022.1.**1** または 2022.1.**5**) も一致させることをお勧めしますが、これは必須ではありません。「静的アセット」がバージョン間で変わりバージョンが一致しない場合、画像が予期しない影響を受ける可能性があります。たとえば、独立したゲートウェイが Tableau Server より前のバージョンである場合、マップが最新ではない可能性があります。
- Tableau Server で Advanced Management を有効化している必要があります。Advanced Management の詳細については、「Tableau Server 上の Tableau Advanced Management について」を参照してください。
- 既定では、独立したゲートウェイは、インストール中にポート 80 および 21319 でバックエンドの Tableau Server 展開と通信できる必要があります。このトピックで後述されているように、これらの既定のポートは初期化中に変更できます。
- 独立したゲートウェイをインストールして構成する前に、Tableau Server の展開が完全で正常であることを確認してください。
- 独立したゲートウェイをインストールしているコンピューターで他の Web 対応アプリケーションを実行していないことを確認します。たとえば、Apache httpd がコンピューターにインストールされている場合は、それをアンインストールするか、ポート 80 でアクティブにリッスンしないように `httpd` を構成します。

Tableau Server とIndependent Gateway のインストール

Independent Gateway は、完全な Tableau Server インストーラーと区別するために、ファイル名の一部に「tsig」を含むスタンドアロンのインストール パッケージを使用してインストールします。Tableau Server をインストールして、期待どおりに機能していることを確認した後に、Independent Gateway をインストールすることを強くお勧めします。インストールするには、root ユーザーである (または root に sudo できる) が必要です。Independent Gateway のインスタンスは 1 つまたは複数インストールできますが、Independent Gateway の各インスタンスは個別にインストールする必要があります。複数のインスタンスをインストールして高可用性を確保したり、クライアントの高い負荷を分散したりする場合は、インスタンスごとにインストール手順を繰り返します。

インストール後、手順を完了するために `initialize-tsig` というスクリプトを実行するように求められます。このスクリプトによって、パラメーターで指定された情報を受け取り、Independen tGateway を構成します。Independent Gateway のインストールを完了したら、Tableau Server の初期ノードで TSM コマンドを実行して、Independent Gateway のインスタンスに関する詳細を使用してサーバーを構成する必要があります。

IG のインストールは、次の手順で行います。

- プラットフォーム固有のインストーラーを実行します。
- インストール後のスクリプトを実行します。
- TSM を使用して Independent Gateway インスタンスを有効にします。

ステップ 1: Tableau Server のダウンロードとインストール

1. TSM のインストールと初期化: このトピックの説明に従い、セットアップ プログラムを実行して TSM をインストールするステップ 1 ～ 5 を完了してください。
2. Tableau Server のライセンス認証と登録: ライセンス認証手順で Tableau Server キーと Advanced Management キーを指定します。次のコマンドを 2 回実行する必要があります。まずは Tableau Server のプロダクト キーと、次に Advanced Management プロダクト キーと共に実行します。

   ```
   tsm licenses activate -k <product key>
   ```

3. 初期ノード設定の構成: トピックの説明に従って、最初のノード設定を構成します。

ステップ 2: Independent Gateway のダウンロードとインストール

ディストリビューションのパッケージ マネージャーを使用して Tableau Server Independent Gateway をインストールしてから、スクリプトを実行して Independent Gateway を初期化します。スクリプトはインストールされたパッケージに含まれています。

Independent Gateway は `/opt` ディレクトリにインストールされます。

1. Independent Gateway をインストールするコンピューターに `sudo` アクセス権を持つユーザーとしてログオンします。

   > **注:** 複雑な問題を避けるため、特殊文字 (たとえば、非ASCII 文字、"+"、"-" など) を含まないユーザー アカウントを使用することをお勧めします。特殊文字を使用すると、

> 環境の構成によっては、Independent Gateway の完全インストールの失敗などの問題が発生する可能性があります。

2. `.rpm` または `.deb` インストーラー パッケージを「Tableau Server のダウンロードおよびリリース ノート」ページからダウンロードします。
3. `.rpm` または `.deb` パッケージをコピーしたディレクトリに移動します。
4. パッケージ マネージャーを使用して、Independent Gateway パッケージをインストールします。

   シンボリック リンクを使用している場所や、ネットワーク ファイル システム (NFS) ボリューム上のディレクトリにインストールしないでください。次のコマンドを実行して Independent Gateway をインストールします。`<version>` は「メジャー番号 - マイナー番号 - メンテナンス番号」の形式です (例: `2022-1-0`)。

   - RHEL 系ディストリビューション (CentOS を含む) の場合:

     ```
     sudo yum update

     sudo yum install tableau-tsig-<version>.x86_64.rpm
     ```

   - Ubuntu の場合:

     ```
     sudo apt-get update
     sudo apt-get upgrade
     sudo apt-get -y install gdebi-core
     sudo gdebi -n tableau-tsig-<version>_amd64.deb
     ```

Tableau Server Independent Gateway の初期化

次のステップでは、スクリプト `initialize-tsig` を実行します。

1. `scripts` ディレクトリへ移動します。

   ```
   cd /opt/tableau/tableau_tsig/packages/scripts.<version_code>/
   ```

2. 次のスクリプトを実行し、Independent Gateway を初期化して起動します。

   ```
   sudo ./initialize-tsig --accepteula -c <ts_cluster_location> --
   <optional_parameters>
   ```

   `initialize-tsig` スクリプトで必要なパラメーターは `--accepteula` と `-c` のみです。

   - **--accepteula -** このパラメーターは、Tableau エンドユーザー ライセンス契約 (EULA) に同意するために含める必要があります。EULA へのリンクは次の場所にあります。

     ```
     /opt/tableau/tableau_tsig/packages/docs.<version_code>/
     ```

   - **-c -** このパラメーターは、Tableau Server クラスター内のすべてのノードのネットワーク上の場所を指定するために含める必要があります。これらのノードは、「ハウスキーピング」要求を Independent Gateway に送信している可能性があります。ワイルドカードとサブネットマスクを使用すると、複数のノードを指定できます。複数のアドレスを指定するには、アドレスをスペースで区切り、全体を引用符で囲みます。値は、Apache `httpd mod_authz_host` の「必須」ディレクティブで受け入れ可能な形式で指定する必要があります。詳しくは、「https://httpd.apache.org/docs/2.4/mod/mod_authz_host.html」を参照してください。

   その他のパラメーター (HTTP やハウスキーピング プロセスで使用されるデフォルトのポートなど) はオプションであり、オフのままにするとデフォルト値が付与されます。スクリプトを実行する前に、パラメーターとそのデフォルト値を確認してください。「initialize-tsig スクリプトの出力ヒント」

ステップ 3: Tableau Server で独立ゲートウェイを有効化

Independent Gateway をインストールして構成する最後の手順は、Tableau Server で Independent Gateway を有効にすることです。これを行うには、1 つまたは複数の Independent Gateway インスタンスを識別し、Tableau Server と Independent Gateway 間の通信に必要な詳細を Tableau Server に提供する JSON ファイルとともに、TSM コマンド `tsm topology external-services gateway enable -c <file>` を使用します。

Independent Gateway を有効にするには、Tableau Server を停止状態にする必要があります。

Independent Gateway JSON ファイルの内容

Tableau Server で Independent Gateway インスタンスの有効化に使用する JSON ファイルには、以下が含まれている必要があります。

- **id**: ID の値は、特定のインスタンスの `tsig_instance_id` と一致する必要があります。これを指定しなかった場合のデフォルト値は、Independent Gateway コンピューターの完全修飾ドメイン名 (小文字) です。JSON ファイル内の値は、`hostname` コマンドの出力と一致する必要があります。
- **host**: ホストの値は、Tableau Server ノードが DNS を使用して解決できる、Independent Gateway コンピューターの DNS で解決可能な名前である必要があります。
- **port**: ポートは、Independent Gateway インスタンスで指定したハウスキーピング ポート (`tsig_housekeeping_port`) と一致する必要があります。初期化中にこれを指定しなかった場合のデフォルト値は「21319」です。
- **protocol**: プロトコルは、Independent Gateway インスタンスで指定したハウスキーピング プロトコル (`tsig_housekeeping_port_protocol`) と同じである必要があります。初期化中にこれを指定しなかった場合のデフォルト値は「http」です。
- **authsecret**: authsecret は、Independent Gateway インスタンスの初期化スクリプトで作成されたシークレットと一致する必要があります。

Independent Gateway の認証シークレット

初期化スクリプトは、各 Independent Gateway コンピューターに一意の共有シークレットを作成します。Tableau Server で Independent Gateway を有効にするには、このシークレットが必要です。シークレットをコピーして、JSON ファイルに「authsecret」として含めます。

共有シークレットは、次の場所の `tsighk-auth.conf` ファイルにあります。

```
/var/opt/tableau/tableau_tsig/config/tsighk-auth.conf
```

Independent Gateway の JSON ファイルの例

JSON ファイルは以下の形式である必要があります。この JSON ファイルの例では、デフォルト値がある場合はデフォルト値を示しています。ご利用のファイルでは、Independent Gateway のインストール状況とお客様の組織に合わせて、実際の値を使用する必要があります。

```
{
 "independentGateways": [
```

```
  {
    "id": "<mycomputer.example.com>",
    "host": "<DNS name of Independent Gateway computer>",
    "port": "21319",
    "protocol": "http",
    "authsecret": "<shared-secret01>"
  },
  {
    "id": "<mycomputer2.example.com>",
    "host": "<DNS name of second Independent Gateway computer>",
    "port": "21319",
    "protocol": "http",
    "authsecret": "<shared-secret02>"
  }
 ]
}
```

Tableau Server で Independent Gateway を有効化

Independent Gateway のインストールを完了するには、TSM を使用して Independent Gateway を有効にする必要があります。

1. JSON 構成ファイルを Tableau Server の初期ノードにコピーします。
2. 初期ノードで、tsmadmin グループのメンバーであるアカウントで、コマンドプロンプトを開きます。
3. 次のコマンドを実行して Tableau Server を停止し、JSON 構成ファイルを使用して Independent Gateway を有効にして、サーバーを再起動します。

   ```
   tsm stop
   tsm topology external-services gateway enable -c tsig.json
   tsm start
   ```

ステップ 4: Tableau Server で Independent Gateway を確認

ブラウザーに Independent Gateway のアドレスを入力すると、Tableau Server のサインインページに移動できるはずです。

Independent Gateway とバックエンドの Tableau Server デプロイメントとの間にファイアウォールがある場合は、直接接続できるように Tableau Server プロセスのポートを開く必要があります。詳細については、「直接接続」を参照してください。

または、Independent Gateway をリレー接続用に構成することにより、ポートの要件を最小限に抑えることもできます。詳細については、「リレー接続」を参照してください。

Independent Gateway を使用した Tableau Server の構成

このトピックでは、Independent Gateway を使用して Tableau Server を構成する方法を、異なる接続シナリオとカスタム認証モジュールの場合について説明します。

インストール手順については、「Tableau Server と独立したゲートウェイのインストール」を参照してください。

AWS の Tableau Server for Linux で実行されているエンドツーエンドの展開の例については、企業環境への導入ガイドの「Web 層の構成」を参照してください。

直接接続とリレー接続

Independent Gateway は、複数のポートを介してバックエンドの Tableau Server プロセスと直接通信できます。この通信を*直接*接続と呼びます。

または、クライアント通信を単一のポートで Tableau Server のゲートウェイプロセスを介して中継するように Independent Gateway を構成できます。この通信を*リレー*接続と呼びます。

この接続タイプを設定する TSM 構成キーは `gateway.tsig.proxy_tls_optional` です。

以下のセクションでは、これらの接続の違いと設定方法について説明します。

直接接続

この構成では、Independent Gateway は Tableau Server のバックエンドプロセスと複数のポートを介して直接通信します。このために、Independent Gateway を Tableau Server のバックエンド展開から分離しているファイアウォールでポートを開く必要があります。

Independent Gateway の現在の実装では、これらのプロセスでの TLS 接続をサポートしていません。

直接接続を行うと、Independent Gateway はゲートウェイプロセスでプロキシされることなくバックエンドの Tableau Server プロセスと通信できます。直接接続は、代わりとなるリレー接続よりも優れたパフォーマンスを提供します。

構成

デフォルトの構成は直接接続です。そのため、設定するためにコマンドを実行する必要はありません。ただし、デフォルトの直接接続にリセットする必要がある場合は、次のコマンドを実行します。

```
tsm configuration set -k gateway.tsig.proxy_tls_optional -v all --
force-keys
tsm pending-changes apply
```

ポート入力の管理

インストール後、Independent Gateway は Tableau Server と複数のポートを介して通信できる必要があります。これらのポートはセットアップ中に動的に割り当てられ、TCP 8000 ~ 9000 の範囲にあります。Tableau Server との通信に使用される特定のポートと対応するプロセスは、Independent Gateway を実行しているコンピューター上の CSV ファイル `TSIG_DATA/config/httpd/proxy_targets.csv` に書き込まれます。

デフォルトのインストールの場合: `/var/opt/tableau/tableau_tsig/config/httpd/proxy_targets.csv`

ネットワークを介した Tableau Server へのポート入力構成を設定または自動化するには `proxy_targets.csv` を使用します。Tableau Server 展開のトポロジが変更されるとポートが変更される可能性があるため、ポート入力構成を自動化することをお勧めします。Tableau Server 展開にノードを追加したり、プロセスを再構成したりすると、Independent Gateway が必要とするポートアクセスが変更されます。

リレー接続

リレー接続構成では、Independent Gateway はバックエンドプロセスに直接接続しません。代わりに、Independent Gateway は、バックエンド Tableau Server 展開上のゲートウェイプロセスへ HTTP 上で通信を中継します。この中継プロセスにより、余分なホップが発生するため、直接接続構成と比較するとパフォーマンスが低下します。

独立したゲートウェイをリレー接続として構成する利点の 1 つは、TLS を使用してトラフィックを保護することです。「独立したゲートウェイで TLS を構成する」を参照してください。

構成

Tableau Server へのリレー接続用に Independent Gateway を構成するには、次のコマンドを実行します。

```
tsm configuration set -k gateway.tsig.proxy_tls_optional -v none --
force-keys
tsm pending-changes apply
```

ハウスキーピング プロトコル

直接接続とリレー接続の両方で、Tableau Server ハウスキーピング (HK) プロトコルでの通信が必要です。HK プロセスによって、バックエンドの Tableau Server 展開と Independent Gateway の間の構成状態を維持します。インストール中、Tableau Server はポート 21319 を介して Independent Gateway と通信できる必要があります。

ハウスキーピング プロトコルの通信の詳細:

- HK 要求は、独立したゲートウェイのステータスを確認し、必要に応じて構成を更新します。これらの要求に顧客データが含まれることはありません。構成にパスワードやその他のシークレットが含まれることもありません。
- 独立したゲートウェイがリバースプロキシ機能を実行できるようにするために、構成ファイルには、Tableau Server クラスタトポロジに関する詳細が含まれています。クラスタトポロジの構成は、攻撃者にターゲット情報を提供する可能性があるため、機密性が高いと考えることができます。このような構成データは、Tableau Server クラスタにアクセスする可能性がある攻撃者のみに有用であることに注意してください。
- 構成更新ファイルには、ハッシュコンテンツのチェックが含まれています。これにより、独立したゲートウェイの更新に使用される構成ファイルの整合性を検証するセキュリティレイヤーが追加されます。

HK プロセスは、デフォルトでは TCP 21319 を使用します。

Tableau Server 2022.1.2 以降、HK 接続で TLS がサポートされます。「独立したゲートウェイで TLS を構成する」を参照してください。

HK ポートの変更

Independent Gateway を初期設定するとき、HK プロトコルで使用するポートを変更できます。initialize-tsig スクリプトの出力ヒントを参照してください。

Independent Gateway をすでにインストールしている場合は、`environment.bash` の `TSIG_HK_PORT` の値を更新するとポートを変更できます。

デフォルトでは、`environment.bash` は `/etc/opt/tableau/tableau_tsig` にあります。

ファイルを更新したら、tsig-httpd を再起動する必要があります。

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

ログ ファイルの場所

Tableau Server でよく役立つログ エントリは、`tabadminagent` ログ ファイル ディレクトリにあります。ただし、Tableau Server をクラスターで実行している場合は、各インスタンスを調べて最新の tabadminagent ログの場所を確認する必要があります。

Independent Gateway では、次のログ ファイルが `TSIG_DATA/logs/` ディレクトリに書き込まれます。

デフォルトでは、パス `/var/opt/tableau/tableau_tsig/logs` にあります。

- `access.log`: Independent Gateway では、httpd.conf.stub 構成が生成するログは `access.log` に書き込まれます。タイムスタンプ付きのログファイル (例: `access_date.log`) は httpd.conf 構成によって生成されます。
- `error.log`
- `startup.log`

これらのログは、そのまま Tableau Server の展開にも中継され、Cluster Controller ログ ディレクトリのサブディレクトリに保存されます。そのため、独立したゲートウェイのログは、`tsm maintenance ziplogs` コマンドによって生成された ziplog ファイルに含まれています。

トラブルシューティング

トラブルシューティングのヒントについては、企業環境への導入ガイド(EDG)の「Tableau Server の独立したゲートウェイのトラブルシューティング」を参照してください。EDG には、Tableau Server on Linux の展開例が記載されています。トラブルシューティングのステップは、Tableau Server の Windows または Linux バージョンに対応しています。

Independent Gateway を使用した認証モジュールの構成

セキュリティの一般的なプラクティスでは、DMZ サーバーの内部ファイアウォールは認証された要求のみを通過させます。Independent Gateway は、Tableau Server の従来の認証方法をサポートしていますが、カスタムの認証用に Apache httpd のロード可能モジュールを統合できる構成プロパティもサポートしています。

たとえば、Tableau Server で SAML を設定してカスタム認証モジュールを構成すると、IdP を使用して認証するように Independent Gateway ですべてのユーザーに要求できます。認証されたユーザーのみが Tableau Server にアクセスでき、その後で Tableau Server はユーザー アクセスの認証と認可を行うことができます。

この認証スキームの詳細については、企業環境への導入ガイドの「AuthN モジュールによる事前認証」を参照してください。

認証モジュールを構成するには、次の手順を実行する必要があります。

1. 認証モジュール構成ファイルを生成します。セットアップが完了すると、各モジュールとその構成ディレクティブは Include オプションとして扱われ、インクルードされたファイルは論理的に全体的な httpd 構成の一部になります。
2. Independent Gateway を実行している各コンピューターに構成ファイルをコピーします。ファイルはすべて、各 Independent Gateway コンピューターの同じ場所にコピーする必要があります。各ファイルは、Tableau Server が管理する構成プロパティにマッピングされます。
3. Tableau Server で `tsm configuration set` コマンドを使用して構成プロパティを設定します。

Independent Gateway には、TSM コマンドを使用して Tableau Server 上で行われた変更に基づいて httpd 構成を更新するロジックが含まれているため、Independent Gateway の httpd 構成ファイル (httpd.conf) は編集しないでください。

認証モジュールの構成例

エンドツーエンド認証モジュールの構成例については、企業環境への導入ガイド「認証構成の例: 外部 IdP を使用した SAML」を参照してください。次の例では、AWS で実行している Tableau Server on Linux の展開に対して、Okta IdP と Mellon 認証モジュールを使用して SAML を設定および構成する方法について説明します。この例では Linux のプロセスについて説明していますが、構成例は Tableau Server on Windows にも役立ちます。

構成プロパティ

次の表では、参照できるさまざまな構成ファイルについて説明しています。各ファイルは、Tableau Server で設定される構成プロパティに対応します。カスタムの認証構成を形成するために必要なプロパティのみを定義する必要があります。不要な構成プロパティはスキップしてください。

構成プロパティ	説明
gateway.tsig.authn_module_block	通常ロードされる Apache httpd モジュールのセットの最後に表示されます。その意図は、ファイルには 1 つまたは複数の `LoadModule` ディレクティブが含まれることです。モジュール自体はフルパスで識別される必要があります。
gateway.tsig.authn_global_block	すべての `LoadModule` の参照の後、他のほとんどの Apache httpd ディレクティブの前に表示されます。その意図は、カスタム モジュールに必要な構成ディレクティブの場所を提供することです。
gateway.tsig.authn_globalbottom_block	構成ファイルの一番下に、グローバル レベルで表示されます。その意図は、特に他のさまざまなディレクティブの後に、カスタム モジュールに必要な構成ディレクティブの場所を提供することです。(これが必要になる可能性はほとんどありません)
gateway.tsig.authn_location_block	`<Location "/">` ブロックの内側に表示され、すべての URL パスをカバーします。
gateway.tsig.authn_directory_block	`<Directory "/">` ブロックの内側に表示され、Independent Gateway が提供するすべてのファイルへの

	パスをカバーします。(これが必要になる可能性はほとんどありません。Independent Gateway が提供するファイルのほとんどは、画像や JavaScript ファイルなどの機密性の低い静的なアセットです)
gateway.tsig.authn_virtualhost_block	1 つまたは 2 つの<VirtualHost> ブロック内に表示されます: 1 つは TLS 用 (有効な場合)、もう1 つは非 TLS 用です。構成されている場合、両方の場所に同じファイルが含まれます。この 2 つのケースを区別する必要がある場合は、標準の Apache httpd HTTPS 環境変数を使用することができます。

<Location "/tsighk"> ブロック

通常の要求トラフィック用の <Location "/"> ブロックに加えて、Independent Gateway のハウスキーピング (HK) 要求を内部で処理するために使用される <Location "/tsighk"> ブロックもあります。これらの HK 要求には独自の認証のしくみがあり、通常のカスタム SSO ソリューションでは機能しません。

カスタム モジュールは、HK URL パスの認証を試行しないように明示的に除外する必要がある場合があります。

ご利用のモジュールを除外する必要があるかどうかを判断するには、まずモジュールを構成します。次に、Independent Gateway のアクセス ログで HK 要求を探します。少なくとも 1 分に 1、2 回はステータス チェックが表示されます。その要求が応答コード 200 を受信している場合は、おそらく問題ありません。一方、要求が応答コード 3xx (ご利用のカスタム認証プロバイダーにリダイレクト) を受信している場合は、何らかの対応が必要です。

考えられる解決策は以下のとおりです。

- <Location "/tsighk"> ブロックには、AuthType None というディレクティブがあり、それで十分かもしれません。
- Independent Gateway の httpd.conf には標準の Apache httpd ディレクティブ ProxyPreserveHost On があります。Off や他の値が必要になるような異常な場合

は、TSM の設定項目 gateway.tsig.proxypreservehost でその値を設定できます。

- <Location "/tsighk"> の認証モジュールを無効にするには、モジュール固有のディレクティブが必要になる場合があります。httpd.conf ファイルでそのブロックを直接変更することはできません。代わりに、gateway.tsig.authn_global_block ファイルに別の <Location "/tsighk"> ブロックを作成し、Apache httpd にそれらを論理的にマージさせることができます。たとえば、オープンソースの認証モジュールとして有名な mod_auth_mellon の一部のバージョンでは、MellonEnable が適用されないセクションで AuthType None が設定されていても、そのセクションで MellonEnable Off を必要としています。
- 前の箇条書きで説明したように、追加の <Location "/tsighk"> セクションを作成する場合、httpd.conf ファイル内で各種のセクションが現れる順序によって、それらが互いに与える影響が異なる場合があります。標準の <Location "/tsighk"> セクションは、標準の <Location "/"> セクションの前に現れます。試験を行った結果、異なる順序が必要であることがわかった場合は、別の <Location "/tsighk"> セセクションに加えて、gateway.tsig.authn_global_block ブロックに別の <Location "/"> セクションを定義する必要があるかもしれません。この場合、gateway.tsig.authn_location_block ブロックには何も必要ない可能性があります。

カスタム認証モジュール構成のトラブルシューティング

Independent Gateway が httpd.conf ファイルを構成する方法を理解するには、Independent Gateway コンピューター上の空のファイルを指す値を使用して TSM 構成項目を設定すると便利です (そのファイルは存在している必要がありますが、空でもかまいません)。次に、Independent Gateway の httpd.conf ファイルを見ると、さまざまな構成ファイルの Include ディレクティブが実際にどこにあるかを具体的に理解することができます。

Independent Gateway の httpd.conf の構成に問題があると、tsig-httpd サービスを開始できないことがあります。他にも構成上の問題があると、Tableau Server クラスター上の Independent Gateway コンパニオンサービスから構成の更新を正常に受信できない場合があります。問題の原因を修正した後、回復する 1 つの方法は、TSIG_DATA/config/httpd.conf.stub を TSIG_DATA/config/httpd.conf にコピーし、tsig-httpd サービスを再起動することです。

その他のトラブルシューティングのヒントについては、企業環境への導入ガイド (EDG) の「Tableau Server の独立したゲートウェイのトラブルシューティング」を参照してください。EDG には、Tableau Server on Linux の展開例が記載されています。トラブルシューティングのステップは、Tableau Server の Windows または Linux バージョンに対応しています。

独立したゲートウェイで TLS を構成する

独立したゲートウェイの TLS サポートは、Tableau Server 2022.1.2 以降で提供されています。

Tableau Server とTableau Server の独立したゲートウェイはどちらも、OpenSSL で構築された SSL モジュール (mod_ssl) を使用して、トランスポートレイヤー セキュリティ(TLS) 機能を実装します。

TLS の構成は複雑でセキュリティに影響されやすいため、Apache httpd の TLS に精通しているIT 専門家が計画および実装することをお勧めします。

多くの場合、既存の TSM または Apache httpd 構成のプロパティまたは概念との互換性を保つために、物事の名前に "SSL" を使用します。"SSL" とは、現在は安全でなく、廃止されていると見なされているプロトコル バージョンを指します。しかし、従来の名前は存続したままで、慣例として TLS と同じ意味でよく使用されています。Tableau Server と独立したゲートウェイでは、SSL 時代のプロトコルはサポートされていません。

重要: バージョン 2025.3以降、Tableau は OpenSSL 3.4.0 とレベル 2 のセキュリティを使用します。レベル 2 のセキュリティには、2048 ビット以上のキーが必要です。既存のレベル 1 のセキュリティキーは使用できなくなるため、新しいキーペアを生成して展開する必要があります。推奨どおりに少なくとも 2048 ビット長のキーを生成している場合、アクションは必要ありません。詳細については、「キーの生成」を参照してください。

TLS 構成の例

エンドツーエンドの TLS 構成の例については、企業環境への導入ガイドの「ロードバランサーから Tableau Server への SSL/TLS の構成」を参照してください。このトピックでは、AWS 展開の Tableau Server on Linux で TLS を構成する例を段階的に示します。この例では Linux のプロセスについて説明していますが、構成例は Tableau Server on Windows にも役立ちます。

TLS 構成の概要

インターネットからTableau Server へのパスの次のセクションのいずれかで、HTTPS の TLS を構成できます。

- 外部ネットワーク(インターネットまたはフロントエンドロードバランサー)から独立したゲートウェイへ
- 独立したゲートウェイからTableau Server へ
- Tableau Server から独立したゲートウェイへのハウスキーピング (HK) プロセス

このトピックでは、これらの各ホップを構成する手順について説明します。

独立したゲートウェイのコンピューターとTableau Server クラスターの構成を変更する必要があります。

証明書の要件と考慮事項

独立したゲートウェイの証明書要件は、Tableau Server の "外部 SSL" に指定されているものと同じです。下記の「SSL 証明書要件」を参照してください。

その他の考慮事項:

- 証明書の管理と展開を簡素化するため、およびセキュリティのベストプラクティスとして、信頼性の高い主要なサードパーティーの認証局 (CA) によって生成された証明書を使用することをお勧めします。または、自己署名証明書を生成するか、TLS 用の PKI からの証明書を使用することもできます。この場合、CA 証明書を信頼して証明書を検証するための構成オプションに注意してください。
- 実装で証明書チェーンファイルを使用する必要がある場合は、ナレッジベースの記事「証明書チェーンを持つ証明書を使用するときに、独立したゲートウェイで TLS を構成する」を参照してください。
- 独立したゲートウェイの複数のインスタンスを実行している場合は、証明書を各コンピューターの同じ場所 (ファイルパス) に配布する必要があります。
- 複数のノードで Tableau Server 展開を実行している場合、TSM コマンドを使用してアップロードした証明書は、ノード間で自動的に分散されます。初期ノードですべての TSM コマンドを実行します。

グローバルな TLS の構成

次の構成はグローバルです。以下の構成オプションは、`tsm configuration set` コマンドを使用して設定する必要のある構成キーを示します。コマンドには、`--force-keys` オプションを含める必要があります。

これらの値を変更する必要はほとんどありません。

キーの各ペアは同じ命名形式を共有しており、文字列 `tsig` が独立したゲートウェイの値を設定します。文字列 `tsig` を含まないキーが、Tableau Server クラスターのゲートウェイプロセスの値を設定します。

`tsig` キーの値を設定しない場合、既定の Tableau Server のゲートウェイ値が使用されます。

`gateway.tsig.httpd.socache` または `gateway.httpd.socache`

既定: `shmcb`

代替値: `dbm`

プロセス間の SSL セッション キャッシュのストレージ タイプshmcb および dbm のストレージ タイプの詳細については、Apache Web サイトの「SSLSessionCache Directive」を参照してください。

`gateway.tsig.httpd.shmcb.size` または `gateway.httpd.shmcb.size`

既定: `2048000`

`shmcb` ストレージ タイプを使用する場合の循環バッファに使用するメモリの量 (バイト単位)

注: もう1 つのグローバル キーは `gateway.tsig.ssl.key.passphrase.dialog` です。該当する場合、`gateway.tsig.ssl.key.passphrase.dialog` の構成は 1 つだけです。設計により、構成内のすべての暗号化された秘密キー ファイルのパス フレーズを収集します。このキーの使用方法については、このトピックの後半の該当セクションで説明します。

独立したゲートウェイへの外部 TLS

独立したゲートウェイ サーバーで TLS を終了するように外部接続を構成するプロセスは、概念的には、Tableau Server クラスターに対して "外部SSL" を構成する方法と似ています。メカニズムは異なります。TSM は、証明書とキーの情報を独立したゲートウェイのノードに自動的に配布しません。また、独立したゲートウェイは、起動時にオプションの TLS キー パスフレーズを提供する方法を自動的に提供しません。

次のステップでは、外部ソースから独立したゲートウェイのコンピューターに TLS を構成する方法について説明します。

ステップ 1: 独立したゲートウェイのコンピューターにファイルを配布する

1. 証明書と関連ファイルを、独立したゲートウェイのサービス (tsig-httpd) が読み取ることができるパーミッションがある場所に配置します。キー ファイルへのアクセスを制限して、独立したゲートウェイのサービスのみがキー ファイルを読み取ることができるようにすることをお勧めします。
2. すべてのファイル、証明書、およびキーを、すべての独立したゲートウェイのコンピューターのまったく同じ場所に配置します。独立したゲートウェイを再インストールまたはアップグレードするときにファイルが削除されないようにするため、ファイルを TSIG_INSTALL パスと TSIG_DATA パスの外側に配置します。

ステップ 2: 独立したゲートウェイのコンピューターの環境変数を更新する

独立したゲートウェイの各コンピューターで、`TSIG_PORT` と `TSIG_PROTOCOL` 環境変数を `443` (慣例により、未使用の TCP ポート番号はすべてサポートされます) および `https` にそれぞれ設定します。

これらの値を変更するには、`environment.bash` の `TSIG_PORT` および `TSIG_PROTOCOL` の環境変数を更新します。

デフォルトでは、`environment.bash` は `/etc/opt/tableau/tableau_tsig` にあります。

ファイルを更新したら、tsig-httpd を再起動する必要があります。

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

ステップ 3: Tableau Server で TLS 構成プロパティを設定する

以下の表にある TSM 構成キーのほとんどは、Apache httpd ディレクティブから派生したものです。そのため、これらの TSM 構成キーの構成値は、対応する Apache ディレクティブの有効な値に直接マッピングされます。対応するディレクティブへのリンクは、次の表に含まれています。

特定のキーが設定されていない場合、フォールバック構成が使用されることもあります。これらは、以下の表で呼び出されます。

次の表の構成オプションは、`tsm configuration set` コマンドを使用して設定する必要のある構成キーを示します。すべてのコマンドに `--force-keys` オプションを含める必要があります。例:

```
tsm configuration set -k gateway.tsig.ssl.enabled -v true --force-keys
```

構成キーを設定したら `tsm pending-changes apply` を実行する必要があります。

構成プロパティ	説明	対応する Apache ディレクティブ
Gateway.tsig.ssl.enabled	必須。 TLS を有効にします。`true` に設定する必要があります。	N/A
Gateway.tsig.ssl.cert.file_name	必須。 独立したゲートウェイの証明書ファイルのパス + ファイル名。例: `/etc/ssl/certs/tsig-ssl.crt`	SSLCertificateFile
Gateway.tsig.ssl.key.file_name	必須。 独立したゲートウェイの証明書キー ファイルのパス + ファイル名	SSLCertificateKey-File

	例: `/etc/ssl/keys/tsig-ssl.key`	
Gateway.tsig.ssl.key.passphrase.dialog	キーにパスフレーズが必要な場合は、Apache httpd SSLPassPhraseDialog ディレクティブで期待される正しい文字列を使用してこのキーを構成する必要があります。このキーの文字通りのパスフレーズを入力しないでください。このキーを構成する方法については、Apache のドキュメントを参照してください。 この構成は、独立したゲートウェイに対してグローバルです。	SSLPassPhraseDialog
Gateway.tsig.ssl.protocols フォールバック: ssl.protocols	サポートされている SSL/TLS のバージョンを指定します。既定の構成については、「セキュリティ強化チェックリスト」を参照してください。	SSL プロトコル
Gateway.tsig.ssl.ciphersuite Fallback: ssl.ciphersuite	クライアントが SSL 接続をネゴシエートするのを許可する暗号を指定します。	SSLCipherSuite
Gateway.tsig.ssl.client_certificate_login.required	この値を `true` に設定し、この接続で相互 TLS を有効にします。 また、`gateway.tsig.ssl.cacert-.file` プロパティを以下のように設定する必要があります。	N/A
Gateway.tsig.ssl.cacert.file	クライアント認証プロセス用に連結された CA 証明書を含むファイ	SSLCACertificateFile

	ルを指定します。	
Gateway.tsig.ssl.revocation.file	独立したゲートウェイに接続するクライアントの連結 CA 失効リストを含むファイルを指定します。	SSLCARevocation-File
Gateway.tsig.ssl.redirect	独立したゲートウェイが TLS 用に構成されている場合、このオプションにより、ポート 80 (既定) からのクライアント要求が TLS にリダイレクトされます。 既定: `true`	N/A
gateway.tsig.ssl.redirect_from_port	`gateway.tsig.ssl.redirect` を `true` に設定すると, このオプションでトラフィックのリダイレクト元のポートを指定できます。 既定: `80`	N/A

Tableau Server への独立したゲートウェイ

このセクションでは、独立したゲートウェイと Tableau Server 間の接続を暗号化する方法について説明します。

ステップ 1: Tableau Server で TLS を構成し、有効にする

「Tableau Server との双方向の外部 HTTP トラフィック用に SSL を構成する」を参照してください。

"SSL" は実際には TLS 実装であり、"外部" は Tableau Server への外部接続を指すことに注意してください。このシナリオでは、独立したゲートウェイは "外部" 接続です。

独立したゲートウェイを構成する前に、TLS を有効にして、クライアントが TLS を使用して Tableau Server に直接接続できることを確認することをお勧めします。

ステップ 2: 独立したゲートウェイのコンピューターで証明書ファイルを配布する

次のいずれかに該当する場合は、独立したゲートウェイのコンピューターで証明書ファイルを配布する必要があります。

- Tableau Server 展開の TLS 証明書に自己署名証明書または PKI 証明書を使用している。
- 独立したゲートウェイから Tableau Server への接続で相互 TLS を有効にしている。

独立したゲートウェイのコンピューター上のすべての TLS 関連ファイルと同様に、各コンピューターの同じパスにファイルを配置する必要があります。TLS 共有ファイルのすべてのファイル名も同じにする必要があります。

ステップ 3: Tableau Server で TLS 構成プロパティを設定する

以下の表にある TSM 構成キーのほとんどは、Apache httpd ディレクティブから派生したものです。そのため、これらの TSM 構成キーの構成値は、対応する Apache ディレクティブの有効な値に直接マッピングされます。対応するディレクティブへのリンクは、次の表に含まれています。

特定のキーが設定されていない場合、フォールバック構成が使用されることもあります。これらは、以下の表で呼び出されます。

次の表の構成オプションは、`tsm configuration set` コマンドを使用して設定する必要のある構成キーを示します。すべてのコマンドに `--force-keys` オプションを含める必要があります。例:

```
tsm configuration set -k gateway.tsig.ssl.enabled -v true --force-keys
```

構成キーを設定したら `tsm pending-changes apply` を実行する必要があります。

構成プロパティ	説明	対応する Apache ディレクティブ
Gateway.tsig.ssl.proxy.cacertificatefile	組織が Tableau	SSLProxyCACertificateFile

	Server で自己署名証明書または PKI で生成された TLS 証明書を使用している場合は、ルート CA 証明書ファイルへのパスを指定する必要があります。このルート CA 証明書ファイルは、独立したゲートウェイのコンピューターに保存する必要があります。	
Gateway.tsig.ssl.proxy.protocols フォールバック: ssl.protocols	サポートされている SSL/TLS のバージョンを指定します。既定の構成については、	SSL プロトコル

	「セキュリティ強化チェックリスト」を参照してください。	
Gateway.tsig.ssl.proxy.ciphersuite Fallback: ssl.ciphersuite	クライアントが SSL 接続をネゴシエートするのを許可する暗号を指定します。	SSLCipherSuite
Gateway.tsig.ssl.proxy.machinecertificatefile	相互 TLS の場合。Tableau Server への独立したゲートウェイの認証用に連結された証明書とキーのペアを含むファイルを指定します。	SSLProxyMachineCertificateFile
Gateway.tsig.ssl.proxy.verify	独立したゲートウェイが Tableau	SSLProxyVerify

	Server によって提示された証明書を検証するかどうかを指定します。 既定は `require` です。	
Gateway.tsig.ssl.proxy.checkpeername	独立したゲートウェイが Tableau Server 証明書を検査してサブジェクト名がサーバー名と一致することを確認するかどうかを指定します。 既定は `off` です。	SSLProxyCheckPeerName
Gateway.tsig.ssl.proxy.checkpeerexpire	独立した	SSLProxyCheckPeerExpire

	ゲートウェイが Tableau Server 証明書を検査して有効期限を確認するかどうかを指定します。 既定は `off` です。	

ステップ 4: ルート CA 証明書を Tableau Server にアップロードする

独立したゲートウェイのコンピューターで使用している TLS 証明書が自己署名証明書または PKI で生成された証明書である場合は、この追加のステップを実行する必要があります。独立したゲートウェイのコンピューターで使用している TLS 証明書が、信頼できるサードパーティーの認証局からの証明書である場合は、このステップをスキップできます。

独立したゲートウェイのコンピューターに使用されるルート CA 証明書を Tableau サーバーの初期ノードにコピーしてから、次のコマンドを実行します。

```
tsm security custom-cert add -c <root-certificate-file-name>.pem
tsm pending-changes apply
```

Tableau Server と独立したゲートウェイの間のハウスキーピング接続

ハウスキーピング (HK) プロセスによって、バックエンドの Tableau Server 展開と独立したゲートウェイの間の構成状態が維持されます。

独立したゲートウェイがインストールされている場合、既定の構成では暗号化されていない HTTP 接続が提供されます。独立したゲートウェイは、(インストール時に定義したとおり)、Tableau Server クラスターから発信されるハウスキーピング要求をリッスンします。

独立したゲートウェイの複数のインスタンスを実行している場合は、すべてのサーバーで TLS を使用してハウスキーピング要求を受け入れるか、TLS を使用せずにハウスキーピング要求を受け入れる必要があります。このセクションでは、HK 接続を TLS に構成する方法について説明します。このプロセスでは、Tableau Server を再起動する必要があり、ダウンタイムが発生します。

前述の TLS シナリオと同様に、HK 接続の構成変更の多くは、Tableau Server クラスターによって管理される構成プロパティで設定されます。ただし、HK TLS 構成では、独立したゲートウェイで追加のステップが必要です。

ステップ 1: 独立したゲートウェイのコンピューターにファイルを配布する

外部ネットワークと独立したゲートウェイで TLS を有効にしている場合は、HK 接続に同じ証明書とキー ファイルを使用できます。

同じアセットを使用している場合、配布する必要がある他の証明書ファイルは、Tableau Server で使用される証明書のルート CA 証明書のみです。Tableau Server によって提示された TLS 証明書が信頼できるサードパーティーの CA によって生成された場合、ルート CA 証明書を独立したゲートウェイのコンピューターにコピーする必要はありません。

1. 証明書と関連ファイルを、独立したゲートウェイのサービス (tsig-httpd) が読み取ることができるパーミッションがある場所に配置します。キー ファイルへのアクセスを制限して、独立したゲートウェイのサービスのみがキー ファイルを読み取ることができるようにすることをお勧めします。
2. すべてのファイル、証明書、およびキーを、すべての独立したゲートウェイのコンピューターのまったく同じ場所に配置します。

ステップ 2: 独立したゲートウェイのルート CA 証明書を Tableau Server の信頼ストアにインポートする

独立したゲートウェイのコンピューターで使用している TLS 証明書が自己署名証明書または PKI で生成された証明書である場合は、この追加のステップを実行する必要があります。独立したゲートウェイのコンピューターで使用している TLS 証明書が、信頼できるサードパーティーの認証局からの証明書である場合は、このステップをスキップできます。

Tableau Server にアップロードできるルート CA 証明書は 1 つだけです。したがって、ルート CA 証明書を既にアップロードしている場合は、HK 接続に使用する証明書に同じルート CA 証明書が署名されている必要があります。

独立したゲートウェイのコンピューターに使用されるルート CA 証明書を Tableau サーバーの初期ノードにコピーしてから、次のコマンドを実行します。

```
tsm security custom-cert add -c <root-certificate-file-name>.pem
tsm pending-changes apply
```

ステップ 3: 独立したゲートウェイのコンピューターの環境変数を更新する

独立したゲートウェイの各コンピューターで、TSIG_HK_PROTOCOL 環境変数を https に設定します。また、TSIG_HK_PORT 環境変数を設定して、HK (既定は 21319) の代替ポートを指定することもできます。

これらの値を変更するには、environment.bash の TSIG_HK_PORT および TSIG_HK_PROTOCOL の環境変数を更新します。

デフォルトでは、environment.bash は /etc/opt/tableau/tableau_tsig にあります。

ファイルを更新したら、tsig-httpd を再起動する必要があります。

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

ステップ 4: 独立したゲートウェイで httpd.conf.stub を更新する

独立したゲートウェイの各サーバーで `httpd.conf.stub` ファイルを更新する必要があります。`httpd.conf.stub` ファイルは、グローバル httpd 構成のシードに使用されます。

ファイルは `TSIG_DATA/config/httpd.conf.stub` にあります。

デフォルトのインストールの場合: `/var/opt/tableau/tableau_tsig/config/httpd.conf.stub`

1. テキストエディターで `httpd.conf.stub` ファイルを開きます。HK 構成の詳細を使用して `<VirtualHost *:${TSIG_HK_PORT}>` ブロックを更新する必要があります。次の例は、必要な変更を示しています。

   ```
   <VirtualHost *:${TSIG_HK_PORT}>
    SSLEngine on
    #TLS# SSLHonorCipherOrder on
    #TLS# SSLCompression off
    SSLCertificateFile /etc/ssl/certs/tsig-ssl.crt
    SSLCertificateKeyFile /etc/ssl/private/tsig-ssl.key
    SSLCACertificateFile /etc/ssl/certs/rootTS-CACert.pem
   #TLS# SSLCARevocationFile /path/to/file
   </VirtualHost>
   ```

   注:

   - 既定では、`<VirtualHost *:${TSIG_HK_PORT}>` の各行が `#TLS#` という文字列でコメントアウトされます。ブロック内の行を"有効"にするには、行の先頭にある `#TLS#` 文字列を削除します。
   - すべての httpd 構成と同様に、各参照ファイルにはファイルへの絶対パスが必要です。
   - `SSLCACertificateFile` は、Tableau Server によって提示される証明書を生成する CA のルート CA 証明書を指定します。これを設定する必要があるのは、Tableau Server で使用される TLS 証明書が自己署名されているか、PKI によって生成されている場合のみです。

2. tsig-httpd サービスを停止します。

```
sudo su - tableau-tsig

systemctl --user stop tsig-httpd

exit
```

この時点で、失敗したステータス チェックのを受信が開始され、独立したゲートウェイのコンポーネントが劣化していることが TSM に示されます。

3. httpd.conf.stub を httpd.conf にコピーします。

   httpd.conf ファイルは同じディレクトリにあります。 httpd.conf を httpd.conf.stub ファイルで上書きします。

   ```
   cp httpd.conf.stub httpd.conf
   ```

4. tsig-httpd サービスを開始します。

   ```
   sudo su - tableau-tsig
   systemctl --user start tsig-httpd
   exit
   ```

   この時点で、失敗したステータス チェックを引き続き受信し、独立したゲートウェイのコンポーネントが劣化していることが TSM に示されます。これらのステータスチェックは、次のステップに従って構成を完了するまで失敗します。

ステップ 5: TLS 構成プロパティを Tableau Server に設定する

構成の変更を適用するには、サーバーを再起動する必要があります。長いタイムアウト時間を回避するため、ここで設定した変更を適用する前にサーバーを停止することをお勧めします。ステップ 6 では、更新 コマンドを実行してから、TSM を再始動します。この段階で TSM を停止させることで、ダウンタイムが短くなります。

1. TSM を停止します。次のコマンドを実行します。

   ```
   tsm stop
   ```

2. 以下の表にあるTSM構成キーのほとんどは、Apache httpdディレクティブから派生したものです。そのため、これらのTSM構成キーの構成値は、対応するApacheディレクティブの有効な値に直接マッピングされます。対応するディレクティブへのリンクは、次の表に含まれています。

   プレフィックスに`hk`ノードを含むTSMの構成プロパティ名があります: `gateway.tsig.hk.xyz.abc`。設定されている場合、これらの値はHK TLS構成に使用されます。設定されていない場合、多くの構成プロパティは`gateway.tsig.xyz.abc`へのフォールバックを使用しますが、`gateway.xyz.abc`にフォールバックする場合としない場合があります。フォールバック構成プロパティは、関連する場合に一覧表示されます。

   次の表の構成オプションは、`tsm configuration set`コマンドを使用して設定する必要のある構成キーを示します。すべてのコマンドに`--force-keys`オプションを含める必要があります。例:

   ```
   tsm configuration set -k gateway.tsig.hk.ssl.enabled -v true --force-keys
   ```

構成プロパティ	説明	対応するApacheディレクティブ
gateway.tsig.hk.ssl.enabled (フォールバックなし)	必須。 TLSを有効にします。`true`に設定する必要があります。	N/A
gateway.tsig.hk.ssl.cert.file_name フォールバック: Gateway.tsig.ssl.cert.file_name	独立したゲートウェイの証明書ファイルのパス+ファイル名。例: `/etc/ssl/certs/tsig-ssl.crt`	SSLCertificateFile
Gateway.tsig.hk.ssl.key.file_name	独立したゲートウェイの証明書キーファイルのパス+ファイル名	SSLCertificateKeyFile

フォールバック: Gateway.tsig.ssl.key.file_name	例: `/etc/ssl/keys/tsig-ssl.key`	
Gateway.tsig.ssl.key.passphrase.dialog (グローバル プロパティ)	キーにパスフレーズが必要な場合は、Apache httpd SSLPassPhraseDialog ディレクティブで期待される正しい文字列を使用してこのキーを構成する必要があります。 この構成は、独立したゲートウェイに対してグローバルです。	SSLPassPhrase-Dialog
Gateway.tsig.hk.ssl.protocols フォールバック: Gateway.tsig.ssl.protocols ssl.protocols	サポートされている SSL/TLS のバージョンを指定します。既定の構成については、「セキュリティ強化チェックリスト」を参照してください。	SSL プロトコル
Gateway.tsig.hk.ssl.ciphersuite フォールバック: Gateway.tsig.ssl.ciphersuite ssl.ciphersuite	クライアントが SSL 接続をネゴシエートするのを許可する暗号を指定します。	SSLCipherSuite
Gateway.tsig.hk.ssl.client_certificate_login.required (フォールバックなし)	この値を `true` に設定し、この接続で相互 TLS を有効にします。 また、`gateway.tsig.hk.ssl.ca-`	N/A

	cert.file プロパティを以下のように設定する必要があります。	
Gateway.tsig.hk.ssl.cacert.file フォールバック: Gateway.tsig.ssl.cacert.file	クライアント認証プロセス用に連結された CA 証明書を含むファイルを指定します。	SSLCACertificateFile
Gateway.tsig.hk.ssl.revocation.file フォールバック: Gateway.tsig.hk.ssl.revocation.file	独立したゲートウェイに接続するクライアントの連結 CA 失効リストを含むファイルを指定します。	SSLCARevocationFile

3. 変更を適用します。次のコマンドを実行します。

   ```
   tsm pending-changes apply。
   ```

ステップ 6: 独立したゲートウェイの JSON 構成ファイルを更新する

最後のステップでは、https への切り替えを反映した JSON ファイルで独立したゲートウェイの構成を更新します。また、該当する場合は、他のポート番号も更新します。

このファイルの編集の詳細については、インストールのトピックを参照してください。「ステップ 3: Tableau Server で独立ゲートウェイを有効化」を参照してください。

JSON ファイルを更新したら、次のコマンドを実行します。

```
tsm topology external-services gateway update -c tsig.json
tsm start
```

トラブルシューティング

トラブルシューティングのヒントについては、企業環境への導入ガイド(EDG)の「Tableau Server の独立したゲートウェイのトラブルシューティング」を参照してください。EDG には、Tableau Server on Linux の展開例が記載されています。トラブルシューティングのステップは、Tableau Server の Windows または Linux バージョンに対応しています。

Tableau Server の独立したゲートウェイのアップグレード

このトピックでは、Tableau Server の独立したゲートウェイをアップグレードする手順を説明します。このトピックで説明するプロセスは、すべてのバージョンのアップグレードで同じです。つまり、このプロセスを使用して、メジャー バージョン (たとえば、2022.1 から 2023.1)、メンテナンス バージョン (2022.1.1 から 2023.1.1) のアップグレードを実施できます。

Tableau Server の独立したゲートウェイのインストーラーは、Tableau Server のバージョンと一致するメジャー バージョンを使用する必要があります。メンテナンス バージョン (たとえば、2022.1.1 または 2022.1.2) も一致させることをお勧めしますが、これは必須ではありません。「静的アセット」がバージョン間で変わりバージョンが一致しない場合、画像が予期しない影響を受ける可能性があります。たとえば、独立したゲートウェイが Tableau Server より前のバージョンである場合、地図が最新ではない可能性があります。

概要

独立したゲートウェイをアップグレードするプロセスは、実際には、ソフトウェアをアンインストールして再インストールするプロセスです。ただし、このトピックで説明するプロセスに従うと、設定作業と全体的なダウンタイムを最小限に抑えることができます。

このトピックで説明するプロセスでは、Tableau Server と Tableau Server の独立したゲートウェイの機能がすでに展開され、稼動していることを前提としています。アップグレードの一環としてバックエンドの Tableau Server 展開をアップグレードする場合は、独立したゲートウェイのサーバーを最初にアップグレードし、Tableau クライアントとの基本的な接続を検証してから、Tableau Server のアップグレードに進むことをお勧めします。

このトピックで説明する手順は、次のとおりです。

1. アップグレード中に使用する既存のファイルをいくつかコピーします。
2. obliterate スクリプトを実行して、独立したゲートウェイを削除します。
3. 新しいバージョンの独立したゲートウェイをインストールします。
4. オプション: tsighk-auth ファイルを元のコピーで上書きします。
5. TLS 設定を更新します。
6. オプション: バックエンドの Tableau Server を更新します。
7. 独立したゲートウェイのサービスを再起動します。

ステップ 1: 参照用にファイルをコピーする

次のファイルに保存されている構成設定を参照する必要がある場合があります。これらのファイルを安全でアクセス可能な場所にコピーします (ファイル パスはデフォルトの場所です)。

- /var/opt/tableau/tableau_tsig/config/tsighk-auth.conf
- /var/opt/tableau/tableau_tsig/config/httpd.conf.stub
- /etc/opt/tableau/tableau_tsig/environment.bash

ステップ 2: 独立したゲートウェイを削除する

サーバーから独立したゲートウェイを削除するには、スクリプト `tableau-tsig-obliterate` を実行します。

1. 最初のノードで、ターミナル セッションを開きます。

2. スクリプト `tableau-tsig-obliterate` を実行します。

   ```
   sudo /opt/tableau/tableau_tsig/packages/scripts.<version_
   code>/tableau-tsig-obliterate -y -y -y
   ```

3. コンピューターを再起動します。

ステップ 3: 独立したゲートウェイをインストールする

アップグレードしたいバージョンの独立したゲートウェイをダウンロードし、トピック「Tableau Server と独立したゲートウェイのインストール」で説明されているのと同じインストール手順に従います。

元のインストールと同様の初期設定を、セットアップの一環として指定する必要があります。同じ値を使用するには、ステップ 1 でコピーしておいた `enviornment.bash` ファイルを参照します。新しい bash ファイルを元のファイルで上書きしないでください。

ステップ 4: (オプション) tsighk-auth ファイルを元のコピーで上書きする

tsighk-auth ファイルは、認証シークレットと呼ばれる一意の文字列を格納する構成ファイルです。認証シークレットは、バックエンドの Tableau Server が信頼できる独立したゲートウェイのインスタンスと通信していることを確認するために使用されます。独立したゲートウェイの元のインスタンスをセットアップしたとき、バックエンドの Tableau Server の構成ファイルをこの認証シークレットで更新する必要がありました。

セキュリティポリシーで許可されている場合は、独立したゲートウェイで元の認証シークレットを引き続き使用できます。そうすることで、バックエンドの Tableau Server を新しい認証シークレットで更新して再起動するプロセスを回避できます。

元の認証シークレットを維持するには、tsighk-auth.conf ファイル (/var/opt/tableau/tableau_tsig/config/tsighk-auth.conf にあります) を、ステップ 1 で保存しておいたコピーと共に使用します。

セキュリティポリシー上、認証シークレットを更新する必要がある場合は、インストールプロセスで生成された新しい認証シークレットを書き留めておいてください。プロセスの後半で、バックエンドの Tableau Server を新しい認証シークレットで更新します。

ステップ 5: ハウスキーピング TLS 設定を更新する

独立したゲートウェイのインスタンスとバックエンドの Tableau Server 展開の間のハウスキーピング (HK) 通信用に TLS を構成していない場合は、この手順をスキップできます。

HK TLS を構成している場合は、元の httpd.conf.stub ファイルを /var/opt/tableau/tableau_tsig/config/httpd.conf.stub の場所にあるファイルに手動でコピーする必要があります。新しいファイルには別の設定が含まれている可能性があるため、新しいファイルを元のファイルで上書きしないでください。

1. 次のブロックで、関連する TLS 構成設定を更新します。

```
<VirtualHost *:${TSIG_HK_PORT}>
SSLEngine on
#TLS# SSLHonorCipherOrder on
#TLS# SSLCompression off
```

```
SSLCertificateFile /etc/ssl/certs/tsig-ssl.crt
SSLCertificateKeyFile /etc/ssl/private/tsig-ssl.key
SSLCACertificateFile /etc/ssl/certs/rootTS-CACert.pem
#TLS# SSLCARevocationFile /path/to/file
</VirtualHost>
```

これらの設定の詳細については、「独立したゲートウェイで TLS を構成する」を参照してください。

2. `httpd.conf.stub` の更新を終えたら、保存します。
3. `httpd.conf.stub` をコピーして保存し、同じディレクトリにある `httpd.conf` を上書きします。

ステップ 6: (オプション) バックエンドの Tableau Server 展開を更新する

ステップ 4 の説明に従って元の認証シークレットファイル (`tsighk-auth.conf`) を独立したゲートウェイの新しいインスタンスにコピーした場合は、このステップを省略できます。

バックエンドの Tableau Server 展開で認証シークレットを更新する場合は、最初のノードの `tsig.json` ファイルを新しい認証シークレットで更新する必要があります。「Tableau Server と独立したゲートウェイのインストール」を参照してください。手順を完了したら、次の TSM コマンドを実行します。

```
tsm stop

tsm topology external-services gateway update -c tsig.json

tsm start
```

ステップ 7: tsig-httpd サービスを再起動する

構成が完了したら、`tsig-httpd` サービスを再起動します。

```
sudo su - tableau-tsig

systemctl --user restart tsig-httpd

exit
```

Tableau Server の独立したゲートウェイのアンインストール

このトピックでは、Tableau Server の独立したゲートウェイをアンインストールするプロセスについて説明します。

独立したゲートウェイのアンインストール

独立したゲートウェイのアンインストールは、2 段階のプロセスです。

1. TSM を使用して Tableau Server の独立したゲートウェイのインスタンスを無効にします。

   Tableau Server で独立したゲートウェイを無効にするには、最初の Tableau Server ノードのコマンドプロンプトで次のコマンドを実行します。

   ```
   tsm topology external-services gateway disable
   ```

2. アップグレードする独立したゲートウェイのインスタンスを、それを実行しているコンピューターからアンインストールします。

   独立したゲートウェイを Linux コンピューターからアンインストールするには、`/opt/tableau/tableau_tsig/packaged/scripts.<version_code>` ディレクトリにあるスクリプト `tableau-tsig-obliterate` を実行します。これにより、コンピューターから独立したゲートウェイが完全に削除されます。

   ```
   sudo /opt/tableau/tableau_tsig/packages/scripts.<version_
   code>/tableau-tsig-obliterate -y -y -y
   ```

独立したゲートウェイのインスタンスをアンインストールするには、TSM を使用してサーバー クラスタからインスタンスを削除し、Tableau Server が完全に再構成されてから、インストールされているコンピューターから独立したゲートウェイをアンインストールします。TSM を使用して Tableau Server からインスタンスを削除すると、TSM から独立したゲートウェイのインスタンスに通信が送信されなくなるため、独立したゲートウェイは構成の変更を認識できなくなります。しかし、独立したゲートウェイは最後に認識された構成に基づいて応答を続け、Tableau Server もそれらの応答を受け入れ続けます。また、独立したゲートウェイのインスタンスを実行しているコンピューターからそれを削除する必要があります。

initialize-tsig スクリプトの出力ヒント

以下のヘルプコンテンツは、次のコマンドを実行したときに出力されます。

```
sudo ./initialize-tsig -h
```

Initialize-tsig スクリプトは、/opt/tableau/tableau_tsig/packages/scripts.<version_code>/ にインストールされています。

出力

```
REQUIRED
--accepteula            Indicate that you have accepted the End User
License Agreement (EULA).
                        You can find a link to the EULA in /opt/tableau/tableau_
tsig/packages/docs.<version_code>

-c <ts_cluster_location>
                        The network location of all nodes in the Tableau
Server cluster. These may send
                        "housekeeping" requests to the TSIG node.
The locations must be one of the forms
                        that are acceptable to Apache httpd mod_
authz_host "Require" directive as described
                        at
https://httpd.apache.org/docs/2.4/mod/mod_authz_host.html. Use
quotes if there are
                        embedded spaces.

OPTIONAL

-i <tsig_instance_id>   A unique identifier for the TSIG instance.
                        Default: The computer name.

-p <tsig_external_port>
                        Port listening for external requests.
                        Default: 80.
```

```
-t <tsig_external_port_protocol>
                        Protocol used for external requests. Options are "http" or
"https".
                        Default: "http".

-k <tsig_housekeeping_port>
                        Port listening for housekeeping requests from Tableau Server
                        Default: 21319.

-s <tsig_housekeeping_port_protocol>
                        Protocol used for housekeeping requests. Options are "http"
"https".
                        Default: "http".

-d data-dir             Set a custom location for the data directory
                        if it's not already set. If not set, the default is
                        "/var/opt/tableau/tableau_tsig".

-f                      Bypass warning messages and distribution version checks.

-g                      Do not add the current user to the "tableau-tsig" group.
                        Use this for easier access to log files and runtime files.

-a <username>           Name of the user to be added to the appropriate
groups instead of
                        the current user running the script. You cannot use both -g
a.

-q                      Quiet, suppress output except for errors and
warnings.

--unprivileged-user=<name>
                        Name of the unprivileged account to run Tableau Server
Independent Gateway.
                        Default: "tableau-tsig".
```

```
--disable-account-creation
                        Do not create groups or user accounts for Tableau Server
Independent Gateway.
                        However, the values in: unprivileged-user will still be used
TSIG configuration.
```

関連トピック

- Tableau Server と独立したゲートウェイのインストール

Tableau Server Backgrounde のリソース制限

Tableau Server Backgrounder のリソース制限機能は Tableau Server 2022.1 で導入されました。

概要と概念

概要

Backgrounder のリソース制限機能を使用すると、Backgrounder のリソースを管理して、リソースの使用方法を制御できます。2022.1 以降、サイトで同時に実行できるバックグラウンドジョブの数に制限を設定できます。サイトがジョブの実行に使用できる 1 日あたりの Backgrounder 時間数を指定することもできます。

デフォルトの制限をすべてのサイトに適用することも、サイトごとにカスタム制限を設定することもできます。これにより、サイトの特定の要件に基づいて Backgrounder リソースを管理できるようになります。

使用するタイミング

この機能は、最も必要とされる場所に確実にリソースを使用したい場合に使用します。さらに重要なことは、1 つのサイトが Backgrounder リソースを大量に消費して、他のサイトのジョブの完了やジョブの待ち時間に影響を与えるのを防ぐことができることです。

現在、Backgrounder リソースの使用量が不均衡であったり、バックグラウンドジョブの完了が遅れたりする場合は、この機能を使用すると、組織のニーズやコンテンツの優先順位に応じてリソース使用量を最適化することができます。

要件と推奨事項

1. この機能を使用するには、Tableau Server の Advanced Management ライセンスが有効になっている必要があります。
2. 設定したリソース制限を適用するには、**Resource Limits Manager** と呼ばれる新しい Tableau Server プロセスが必要です。Tableau Server 2022.1 以降をインストールまたはアップグレードすると、このプロセスは初期ノードで自動的に構成されます。
    - Backgrounder のリソース制限機能を最適に実行するには、Tableau Server で少なくとも計 5 つの Backgrounder プロセスを実行することをお勧めします。デフォルトの構成とトポロジーの推奨事項の詳細については、「Tableau Server リソース制限マネージャー」を参照してください。
    - ご利用の Tableau Server に Resource Limits Manager プロセスを追加することはお勧めしません。初期ノードに自動的にインストールされるプロセスで十分です。

用語と概念

- **サイトのジョブ制限:** サイトのバックグラウンドジョブの同時実行と実行時間の制限。
- **サイトのデフォルト制限:** サイトのデフォルトのジョブの同時実行と実行時間の制限。
- **サイトのカスタム制限:** そのサイトに固有のサイト制限。
- **ジョブ タイプ**: タスク タイプと同じ。抽出更新、サブスクリプション、およびフローが含まれます。
- **同時ジョブ制限:** 特定のタイプのジョブを同時に実行できる最大数。最大数は、サーバーにデプロイされる Backgrounder プロセスの総数と同じです。
- **1 日の制限:** 1 日あたりの制限には、実行時間の制限とリセット時間が含まれます。
- **実行時間制限:** サイトに割り当てられた 1 日あたりの Backgrounder ジョブ時間の合計数です。最大数は、サーバーにデプロイされる Backgrounder プロセスの総数に 24 を掛けたものと同じです (24 は 1 日の時間数です)。
- **リセット時刻:** 1 日あたりの制限がリセットされる時刻。これは自動的に UTC 午前 0 時に設定され、変更できません。
- **制限なし:** リソース制限が設定されていない場合と同じです。この場合、並行して実行できるサイトのジョブは、Tableau Server で使用可能なすべてのバックグラウンドプロセスを使用できます。

実施できること

Tableau Server 管理者として、

1. タスクタイプ (抽出更新、サブスクリプション、フロー) ごとに同時実行ジョブの制限を指定できます。この制限は、特定のタスクタイプについて、サイト上で同時に実行できるバックグラウンドジョブの数を制御します。この制限は、タスクタイプごとに固有です。つまり、抽出更新、サブスクリプション、フローに異なる制限を設定できます。
2. サイトがジョブの実行に使用できる24時間あたりのBackgrounder時間数、つまり1日のジョブ実行時間制限を指定できます。同時実行ジョブの制限とは異なり、1日のジョブ実行時間の制限は、すべてのタスクタイプにわたって累積されます。
3. 過去24時間にサイトで使用されたBackgrounder時間数を確認できます。
4. 同時実行制限が設定されているためにジョブが遅延するタイミングを確認できます。
5. 利用可能なBackgrounderリソースが不足しているために、ジョブがいつキャンセルされたか特定できます。

ジョブのステータスと詳細については、「Tableau Serverでのバックグラウンドジョブの管理」を参照してください。

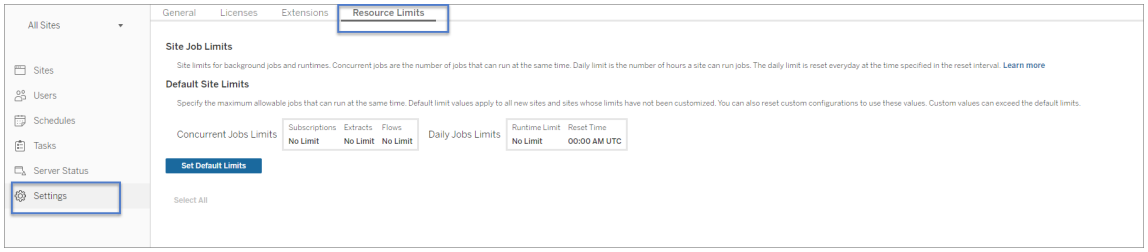
タスクの所有者として、サイトが1日の実行時間制限に達したためにバックグラウンドジョブがキャンセルされると、メールを受信します。

Backgrounder リソース制限の設定方法

Tableau Serverは、リソース制限を自動的に設定しません。リソース制限を設定するまで、Backgrounderリソースの使用に制限はありません。

リソース制限を初めて設定したり、その後変更を加えたりするには、**[設定]** ページに移動して **[リソース制限]** タブを選択します。

カスタム制限は、デフォルト制限を設定した後でのみ設定できます。

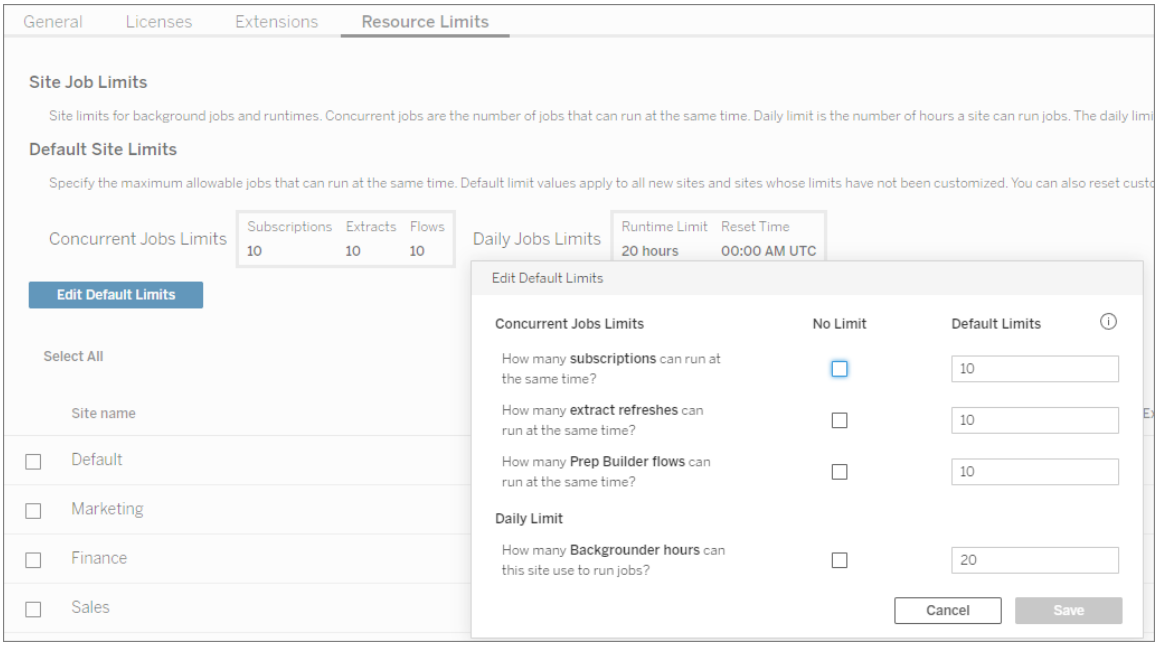


サイトのデフォルト制限

サイトのデフォルト制限はサーバー レベルで設定されているため、すべてのサイトに適用できます。サイトのデフォルト制限への変更は、すべての新しいサイトと、デフォルト制限を使用するように設定されているすべての既存のサイトに適用されます。カスタム制限が設定されているサイトは、この変更の影響を受けません。

- **同時ジョブ制限**と**1 日の実行時間制限**は、整数で指定する必要があります。
- サイトのデフォルト制限への変更はすぐに有効になり、サーバーを再起動する必要はありません。

デフォルト制限を初めて設定するには、**[リソース制限]** タブで **[デフォルト制限の設定]** を選択します。既存のデフォルト制限を更新するには、**[デフォルト制限の編集]** を選択します。

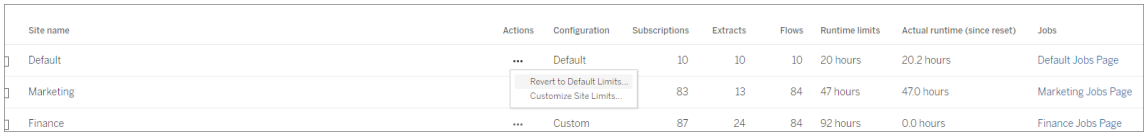


サイトのカスタム制限

サイトによっては、必要なリソースがデフォルトの制限よりも多いまたは少ない場合があります。これは、サイトにあるコンテンツの量と、ビジネス運営におけるコンテンツの重要性によって異なります。サイトのデフォルト値が適切な容量でない場合は、そのサイトにカスタムのリソース制限を設定できます。サイトのカスタム制限は、デフォルト制限を超える可能性があります。

- **同時ジョブ制限**と**1 日の実行時間制限**は、整数で指定する必要があります。
- サイトのカスタム制限の変更はすぐに有効になり、サーバーを再起動する必要はありません。

サイトのカスタム制限を設定するには、**[リソース制限]** タブの **[アクション]** で楕円をクリックし、**[サイト制限のカスタマイズ]** を選択します。



リソース制限を構成後の動作

サイトのリソース制限を設定すると、Tableau Server は Backgrounder リソースの使用状況を監視して追跡し、適切な制限が適用されていることを確認します。

サイトが最大同時実行制限に達すると、ジョブはキューに入れられ、そのサイトで現在実行されているジョブが完了して Backgrounder の処理能力が使用可能になるまで実行されません。

12 時間経っても使用可能な同時実行リソースがない場合、まだ保留中のジョブはキューから削除されます。1 日の実行時間制限に達すると、**その日の保留中のジョブはキャンセルされ**、タスクの所有者に通知が送信されます。

この機能を使用するには、Tableau Server が有効なサーバー管理ライセンスで認証されている必要があります。このライセンスがないと、制限は適用されません。ライセンスが無効であるか、何らかの理由でライセンス認証が解除された場合、以前に設定されていた制限は保存され、ライセンスの問題が解決されると適用されます。

リソース制限を調整するタイミング

構成を変更する前に、Tableau Server が実行中であり、良好な状態にあることを確認してください。

[ジョブ] ページを使用すると、リソース制限が原因で保留されたりキャンセルされたりしたジョブを特定できます。

リソース制限の監視と調整を行うためのいくつかのパターンを次に示します。

- サイトのジョブが常に 1 つまたは複数キャンセルされている場合は、1 日の実行時間制限を増やす必要がある場合があります。
- サイトのジョブが常に 1 日の早い時間にキャンセルされている場合は、そのサイトの 1 日のジョブ実行時間が十分でない可能性があります。
- 特定のタイプの複数のジョブが一定期間保留されている場合は、そのタイプのタスクの同時ジョブ制限を増やすことを検討することをお勧めします。または、すべてのタスクが同時に実行されないように、異なる時間にタスクをスケジュールすることを検討してください。また、より長い期間にまたがるスケジュールを作成することを検討することもできます。そうすれば、すべてのタスクが互いに近づきすぎて実行されないようにスケジュールすることができます。
- 上記のようにリソース制限を変更したくない場合は、ジョブの実行がスケジュールされる頻度を調整することもできます。たとえば、ジョブが 1 時間ごとに実行されるようにスケジュールされている場合は、低い頻度のスケジュールで実行されるように調整します。

実行可能なユーザー

リソース制限は、Tableau Server 管理者が構成できます。

ジョブがキャンセルされたときのメール通知は、抽出更新、サブスクリプション、またはフロー実行タスクを所有する Tableau Server ユーザーが受け取ります。

コンテナでの動的スケーリング - Tableau Server バックグラウンダー

概要

コンテナでのバックグラウンダーの動的スケーリングにより、Tableau Server のバックグラウンダー ジョブとスケジュールされたジョブにさまざまなスケーリング戦略を適用できます。ここで言う自動スケーリングとは、人手を介さず、また他のサーバー システムの稼働時間に影響を与えることなく、タスクの負荷が変動しても処理できるように、サービスを独立してスケーリングできることを意味します。Tableau Server プロセスのノードが全部そろった Tableau Server コンテナは、引き続き一体的なシステムとして実行されます。その代わりに、切り離されて独立した、一連の小さなコンテナ サービスが「バックグラウンダー」サービス ロールを構成します。これは、動的に拡張することができ、通常は Tableau Server コンテナが処理する計算負荷を処理します。バックグラウンダー サービスは、抽出の更新や作成、サブスクリプションの送信、データ アラートの確認、および多くのメンテナンス ジョブを含むシステム タスクの処理を担当します。たとえば、多数のデータセットを更新したり、計算コスト

の高い一連のデータアラートを計算したりすることが有利な場合は、Kubernetes を活用して計算能力をスケールアップし、これらのタスクを効率的に完了することができます。このガイドでは、Kubernetes での自動スケーリング バックグラウンダーの構成とデプロイの要件について説明します。このドキュメントは、「Tableau Server in a Container」ドキュメントの補足です。

前提条件

自動スケーリング バックグラウンダーは Kubernetes でのみ使用でき、Tableau Server in a Container に基づいています。自動スケーリング バックグラウンダー機能を使用するには、特定の前提条件を満たす必要があります。

- Tableau Server で Advanced Management 機能が有効になっている必要があります。詳細については、「Tableau Server 上の Tableau Advanced Management について」を参照してください。
- ロールベース ライセンスを使用している必要があります。コア ライセンスはサポートされていません。ライセンスの詳細については、「ライセンス モデルとプロダクト キーを理解する」を参照してください。
- 次の Advanced Management 機能を有効にする必要があります。
    - Tableau Server 外部ファイル ストア
    - Tableau Server 外部リポジトリ
- バージョン 1.20 以降の Kubernetes クラスタがあり、その使用方法と管理方法を理解している必要があります。

制限事項

- この機能は、Linux ベースの展開の一部としてのみ機能します。
- フロー ジョブは、自動スケーリング バックグラウンダーではサポートされていません。フロー ジョブは、Tableau Server コンテナで引き続き実行されるバックグラウンダー サービスによって処理されます。

Tableau Server およびバックグラウンダー ポッドのイメージの作成

コンテナで自動スケーリング バックグラウンダーを使用するための最初のステップは、Tableau Server のインストールを構成するサービス イメージを作成することです。これらのイメージには、Tableau Server のイメージと、個別のバックグラウンダーおよびサポート サービスのイメージが含まれます。使用するビルド ツールは、包括的なオールインワンの Tableau Server コンテナ イメージを作成するために使用するものと同じですが、ツールはバージョン 2022.3.0 以降であること、

Advanced Management ライセンスがあること、イメージをビルドするときに特別なフラグを使用することなどが必要です。

1. サービスイメージを作成するには、次のコマンドを実行します。

   ```
   build-image --accepteula -i <installer> --backgrounder-images
   ```

   これにより、Tableau Server と4 つの新しいイメージが作成されます。これらの追加のイメージには、自動スケーリング バックグラウンダーの新しいポッドを構成する、個々のサービスが含まれています。

   `docker images` コマンドを使用すると、ローカルの docker リポジトリに作成されたイメージを一覧表示できます。

   ```
   hyper                          20214.21.1117.1006
   52fd9843c178   10 seconds ago
   gateway                        20214.21.1117.1006
   2308e602a2a3   11 seconds ago
   backgrounder                   20214.21.1117.1006
   4540e459cf23   12 seconds ago
   dataserver                     20214.21.1117.1006
   c5345ed47f51   12 seconds ago
   tableau_server_image           20214.21.1117.1006
   b27817047dd1   7 minutes ago
   ```

   新しいバックグラウンダー ポッドを構成するイメージは、hyper、gateway、backgrounder、および dataserver です。カスタム ドライバー、インストール スクリプト、およびプロパティは、これら5 つのイメージすべてで共有されます。詳細については、「イメージをカスタマイズする」を参照してください。

2. これらのすべてのイメージを、デプロイ用の内部イメージ リポジトリにパブリッシュします。

展開ガイド

以下の情報は、Tableau Server を自動スケーリング バックグラウンダーとともにコンテナにデプロイする方法を説明するものです。この情報では、self-container コンテナに Tableau Server を展開する方法を既に理解していることを前提としています。詳細については、Tableau Server in a

Containerを参照してください。「Kubernetes 構成」セクションの 3 つの Kubernetes 構成ファイルは、デプロイメントのセットアップに使用できるテンプレートです。展開の要件と詳細については、このガイドの他のセクションで説明しています。

Tableau Server と自動スケーリング バックグラウンダーのデプロイは、ガイドの下部にある入力済みの Kubernetes 構成ファイルをデプロイするのと同様に簡単です。

```
kubectl apply -f <tableau-kubeconfig-dir>
```

バックグラウンダー ジョブ

バックグラウンダー ポッドは、スケジュールされた追加のワークロードを Tableau Server in a Conainer が並行して処理するのを援助します。バックグラウンダーは、抽出更新、サブスクリプション、アラート、フロー、およびシステム ワークロードを処理します。バックグラウンダー ポッド間でジョブを分散するということは、Tableau Server が他のタスクの処理に使用できるコンピューティング リソースが増えることを意味します。Tableau Server が処理するタスクには、ワークブックやダッシュボードのレンダリングなど、インタラクティブなユーザー アクティビティなどがあります。フロー ジョブは、このバックグラウンダーで実行されない唯一のタイプのバックグラウンダー ジョブです。バックグラウンダー ジョブの詳細については、「Tableau Server でのバックグラウンド ジョブの管理」を参照してください。

バックグラウンダー ポッドは、フロー ジョブ以外のあらゆる種類の負荷を処理できます。フロー ジョブは、バックグラウンダー サービスを実行し続ける、メインの Tableau Server コンテナで実行する必要があります。

ノード ロール機能を使用すると、特定の種類のジョブ用にバックグラウンダー ポッドを柔軟に割り当てることができます。この機能は、Tableau サーバーのノード ロール機能の拡張です。さまざまなノード ロールに関する詳細については、こちらを参照してください。バックグラウンダー ポッドはフロー ジョブを実行できないため、デフォルトでは、バックグラウンダー ポッドでフロー ジョブが無効になっていることに注意してください (つまり、ロールは「フローなし」に設定されています)。

バックグラウンダーのノード ロールを設定するには、バックグラウンダー サービスを実行するコンテナの kubeconfig の一部として、環境変数 NODE_ROLES を設定する必要があります。例えば、backgrounder を設定して抽出更新のジョブのみを実行するには、以下に示すように NODE_ROLES 環境変数を抽出更新に設定します。

NODE_ROLE_CONFIG

```
containers:
      - name: backgrounder
        image: <backgrounder_image> # Backgrounder Image
        ports:
        - containerPort: 8600
        volumeMounts:
        - name: logmount
          mountPath: /var/log/tableau
        - name: clone-volume
          mountPath: /docker/clone
        - name: dataengine-volume
          mountPath: /docker/dataengine
        - name: temp
          mountPath: /var/opt/tableau/tableau_
server/data/tabsvc/temp
        env:
          - name: ROOT_LOG_DIR
            value: /var/log/tableau
          - name: CLONE_ARTIFACT_DIR
            value: /docker/clone/clone-artifacts
          - name: FILESTORE_MOUNT_PATH
            value: /docker/dataengine
          - name: NODE_ROLES
            value: extract-refreshes
```

Tableau Server ポッドには、バックグラウンダー ジョブを実行する場所が常に存在することを保証するために、トポロジーで構成されたバックグラウンダーが少なくとも 1 つあります。デフォルトでは、バックグラウンド ジョブのすべてのロールを処理できるバックグラウンダーが TSM で必要とされています。シナリオによっては、バックグラウンダー ポッドに特定のタイプのジョブをすべて処理させたい場合があります。これを行うには、サーバー構成キー `topology.roles_handle_all_jobs_constraint_disabled` を `true` に設定する必要があります。これにより、TSM トポロジーがすべてのジョブ タイプを処理するという要求を無効にできます。このパラメーターを設定すると、Tableau Server バックグラウンダー インスタンスのバックグラウンダー ロールが `no-extract-refreshes` に設定され、バックグラウンダー ポッドのロールが `extract-refreshes` に設定され

ます。これにより、すべての抽出更新ジョブがバックグラウンダー ポッドでのみ実行されるように設定できます。

注: この制約を無効にすると、ある種のジョブが決してスケジュールされないようにロールを設定することもできます。TSM バックグラウンダーとバックグラウンダー ジョブのロール構成は慎重に設定する必要があります。これは、すべての種類のバックグラウンダー ジョブをスケジュールできることを、もはや TSM は検証しなくなるためです。

Tableau Server in a Container のポッド

自動スケーリング バックグラウンダー ポッドの一部として Tableau Server を含むコンテナは、既存の Tableau Server in a Container とほぼ同じ方法でデプロイされます。重要な要件の変更がいくつかあります。

- Tableau Server コンテナとバックグラウンダー ポッドの間で構成を転送するには、ネットワーク ファイル共有が必要です。
- 外部ファイルストア機能を有効にして使用する必要があります。これには、専用のネットワーク ファイル共有も必要です。

バックグラウンダー ポッド

バックグラウンダー ポッドは、連携して動作する 4 つの独立したサービス コンテナ (**gateway**、**hyper**、**dataserver**、**backgrounder**) で構成されます。これらのポッドは、典型的な独立した Kubernetes コンテナ ポッドのようにデプロイできます。ポッドには次の要件があります。

- バックグラウンダー ポッドは、ホスト名の DNS 解決を使用して Tableau Server ノードに到達できる必要があります。
- 外部ファイルストアとクローン ネットワーク ファイル共有が提供される必要があります。

注: バックグラウンダー ポッドは、init-container で構成され、Tableau Server コンテナがクローン構成の出力を正常に生成するまで待機してから実行されます。

ログ

バックグラウンダー ポッド サービス (Tableau Server など) は、依然として主にディスクにログを書き込みます。バックグラウンダー ポッドは、スケール インとスケール アウトが可能であり、一時的な存在です。そのため、ポッドからログを取り出して保存することが重要です。既存の K8s 環境を使用している多くのお客様は、デプロイしたポッドからログを収集するために、すでに何らかの種類のログ集約サービスを使用しているでしょう。ログ集約サービスの例としては、Splunk や fluentd などがあります。何らかのログ集約サービスを使用して、バックグラウンダー ポッドからログを収集することを強くお勧めします。ログ管理を容易にするために、Tableau が提供する kubeconfig は、共有ログ ボリュームに書き込むようにポッド内の各サービスを構成します。各サービス コンテナ内のディレクトリのパスは、ROOT_LOG_DIR 環境変数によって指定されます。

サポート ケースを開いてログを提供する必要がある場合は、メイン サーバー コンテナから収集された ziplog と、バックグラウンダー ポッドからのログ (ログ集約サービスから取得するか、以下の手動プロセスを使用) の 2 セットのログを提供してください。

ログ集約サービスを使用できないお客様は、ポッドから手動でログを取得できます。

注: ログを含むボリュームに対して「永続ボリューム要求」を使用していない場合、ポッドからのログは、ポッドがスケールダウンされると失われてしまいます。

関連するすべてのログは、バックグラウンダー ポッド内の /var/log/tableau ディレクトリ(ROOT_LOG_DIR 環境変数で構成可能) にあります。この場所に PersistantVolumeClaim をマウントして、ポッドが停止してもログを利用できるようにすることを強くお勧めします。

バックグラウンダー ポッドの実行中にログを収集する

コンテナ内のログの tar ファイルを作成します。

```
kubectl exec -it <backgrounder-pod-name> -- bash -c "tar -czf
/docker/user/backgrounder-pod-logs.tar.gz /var/log/tableau"
```

tar ファイルをコンテナの外部にコピーします。

```
kubectl cp <backgrounder-pod-name>:docker/user/backgrounder-pod-
logs.tar.gz ./backgrounder-pod-logs.tar.gz
```

バックグラウンダー ポッドが終了した (または開始に失敗した) ときにログを収集する

バックグラウンダー ポッドのデプロイ ログに使用される PersistantVolumeClaim マウントを使用して、実行時間の長いポッドをアタッチします。以下は構成例です。

```
apiVersion: v1
kind: Pod
metadata:
  name: <name>
  namespace: <namespace>
spec:
  containers:
  - name: get-logs-pod
    image: busybox:1.28
    securityContext:
      runAsUser: 0
      allowPrivilegeEscalation: true
    command: ['sh', '-c', "while :; do sleep 5; done"]
    volumeMounts:
    - name: logmount
      mountPath: /var/log/tableau
restartPolicy: Never
volumes:
  - name: logmount
    persistentVolumeClaim:
      claimName: logvolume
```

コンテナ内のログの tar ファイルを作成します。

```
kubectl exec -it <backgrounder-pod-name> -- bash -c "tar -czf
/backgrounder-pod-logs.tar.gz /var/log/tableau"
```

tar ファイルをコンテナの外部にコピーします。

```
kubectl cp <backgrounder-pod-name>:/backgrounder-pod-logs.tar.gz
./backgrounder-pod-logs.tar.gz
```

ライブ構成変更

Tableau Server in a Container の構成を変更し(例えば、tsm コマンドラインを使用)、それらの構成変更をバックグラウンダー ポッドで反映させたい場合は、`tsm settings clone` コマンドを実行して、一連の新しいクローン構成ファイル (クローン ペイロード) を生成する必要があります。

1. TSM を使用して、Tableau Server in a Container ポッドの構成を変更し、構成の変更をサーバーに適用します。
2. Tableau Server in a Container ポッドで次のコマンドを実行します。

   ```
   ## Run this command in the Tableau Server in a Container pod.
   tsm settings clone -d $CLONE_ARTIFACT_DIR
   ```

   このコマンドは、一連の新しい構成ファイルを生成し、クローン NFS ドライブの場所に書き込みます。
3. バックグラウンダー ポッドを再デプロイします。ポッドは、クローン NFS ドライブを使用するように構成する必要があり、新しい構成を取得します。

スケーリング戦略

バックグラウンダー ポッドは、さまざまな手法と戦略を使用して Kubernetes でスケーリングできます。タイム スケジュールに基づいてバックグラウンダー ポッドのプール サイズを変更するスケーリング戦略の例を示します。

CPU とメモリの使用率は、バックグラウンダー ポッドをスケーリングするための適切な指標**ではない**いことに注意してください。CPU とメモリの使用率は、クラスタの全体的な負荷需要を正確に表していません。たとえば、抽出を更新するためにバックグラウンダー ポッドの使用率が最大になったとしても、バックグラウンダー ジョブのキューで待機している他のジョブはありません。この場合、自動スケーリングによってジョブのスループットが向上することはありません。

スケジュールされたスケーリング

cron ジョブを使用した標準の Kubernetes メカニズムにより、スケーリング機能をスケジュールできます。

この Kubernetes 構成の例は、以下の「Kubernetes の構成」セクションに記載されています。

Kubernetes の構成

新しい環境変数

標準の Tableau Server コンテナ環境変数 (初期構成のオプションを参照) に加えて、Kubernetes 構成で設定する必要がある新しい必須環境変数がいくつかあります。

環境変数	推奨値	説明
FILESTORE_MOUNT_PATH	/docker/dataengine	外部ファイルストアディレクトリのマウント場所。このディレクトリは、デプロイされた各 Tableau コンテナ内にマウントされた、データエンジン NFS ディレクトリを指す必要があります。外部ファイルストアの詳細については、「Tableau Server 外部ファイル ストア」を参照してください。この値は、Tableau Server in a Container のポッドとバックグラウンダーのポッドで同じにする必要があります。
CLONE_ARTIFACT_DIR	/docker/clone/clone-artifacts	クローン構成ディレクトリのマウント場所。このディレクトリは、各 Tableau コンテナ内にマウントされた、NFS ディレクトリを指す必要があります。Tableau Server は、バックグラウンダー ポッドがクラスタのメンバーになるために使用する構成データを出力します。
ROOT_LOG_DIR	/var/log/tableau	(バックグラウンダー ポッドのみ) バックグラウンダー ポッドで実行されているすべてのサービスの共通ログ ディレクトリの場所。

バックグラウンダー ポッドのポート

バックグラウンダー ポッドは 4 つのサービスで構成され、デフォルトではそれぞれが指定されたポートで実行されるように設定されています。コンテナ内でサービスが接続するポートを変更する場合は、サービスのポート割り当てに対応するキーを指定する必要があります。このような構成は、サイドカー コンテナやその他のコンポーネントがポッドに追加され、サービスのポートが競合するようなことがない限り、ほとんどの場合必要ありません。

ポート環境変数	既定
BACKGROUNDER_PORT	8600
DATASERVER_PORT	8400
HYPER_PORT	8200
GATEWAY_PORT	8080

Dataserver もポート 8300 を使用しますが、これは再構成できません。

共有ネットワークディレクトリ

この Tableau Server の展開では、適切に機能するために 2 つのネットワーク共有が必要です。すべての Tableau Server と、バックグラウンダー や Kubernetes の構成テンプレートでは、これらのネットワークディレクトリが存在することに注意してください。

- **Dataengine ディレクトリ**(FILESTORE_MOUNT_PATH): バックグラウンダー ポッドには、外部ファイルストア機能が必要です。このネットワーク共有には、Tableau Server とバックグラウンダー ポッドの間で共有される、抽出とその他のファイルベースのアーティファクトが含まれています。
- **Clone ディレクトリ**(CLONE_ARTIFACT_DIR): Tableau Server は静的な接続と構成の情報をネットワーク共有に書き込みます。バックグラウンダー ポッドは、この情報を使用して Tableau Server クラスタのメンバーになります。将来のプレリリースでは、この構成は Kubernetes 構成の標準ライフサイクルに組み込まれます。

重要: クラスタ全体 (新しい Tableau Server コンテナを含む) を再デプロイする場合は、クローン NFS マウントの内容を消去する必要があります (そうしなければ、バックグラウンダー ポッドが古いサーバーに接続しようとします)。

Kubernetes の構成例

注: 構成例には、レディネスプローブの使用が含まれています。Tableau Server コンテナーの展開が単一ノードの TSM 展開である場合 (展開には複数のバックグラウンダー ポッドを含めることができます)、レディネスプローブを使用できます。マルチノードの Tableau Server in a Container 展開では、レディネスプローブを使用することはできません。

Tableau Server コンテナの構成

```
---
apiVersion: v1
kind: Service
metadata:
  name: <service_name>
  namespace: <namespace>
spec:
  selector:
    app: <service_name>
  ports:
    - protocol: TCP
      port: 8080
      nodePort: <nodeport-number>
      name: http
    - protocol: TCP
      port: 8443
      nodePort: <nodeport-number>
      name: https
  type: NodePort
---
apiVersion: v1
```

```
kind: ConfigMap
metadata:
  name: configfile
  namespace: <namespace>
data:
  config.json: |-
    {
      "configEntities": {
      "identityStore": {
        "_type": "identityStoreType",
        "type": "local"
      }
    },
    "configKeys" : {
      "tabadmincontroller.init.smart_defaults.enable" : "false",
      "wgserver.domain.ldap.starttls.enabled" : "false"
    },
    "daysLeftForMaintenanceExpiring" : 0
  }
---
apiVersion: v1
kind: ConfigMap
metadata:
  name: extrepojsonfile
  namespace: <namespace>
data:
  config.json: |-
    {
      "flavor":"generic",
      "masterUsername":"<admin-name>",
      "masterPassword":"<password>",
      "host":"<hostname>",
      "port":5432,
      "prerequisiteCheckEnabled":false,
      "requireSsl":false
    }
---
```

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: datadir1
  namespace: <namespace>
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
    storage: 50Gi
---
# This is required for multi-node tableau server in container
apiVersion: v1
kind: PersistentVolume
metadata:
  name: bootstrapnfs
  namespace: <namespace>
spec:
  accessModes:
    - ReadWriteMany
  capacity:
    storage: 1Gi
  nfs:
    server: <nfs-ip>
    path: <mount-path>
---
# This is required for multi-node tableau server in container
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: bootstrapnfs
  namespace: <namespace>
spec:
  accessModes:
    - ReadWriteMany
  storageClassName: ""
```

```
    resources:
      requests:
        storage: 1Mi
---
apiVersion: v1
kind: PersistentVolumn
metadata:
  name: clonenfs
  namespace: <namespace>
spec:
  accessModes:
    - ReadWriteMany
capacity:
  storage: 1Gi
nfs:
  server: <nfs-ip>
  path: <mount-path>
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: clonenfs
  namespace: <namespace>
spec:
  accessModes:
    - ReadWriteMany
  storageClassName: ""
  resources:
    requests:
      storage: 1Mi
---
apiVersion: v1
kind: PersistentVolume
metadata:
    name: dataenginenfs
    namespace: <namespace>
spec:
```

```
  accessModes:
    - ReadWriteMany
  capacity:
    storage: 1Gi
  nfs:
    server: <nfs-ip>
    path: <namespace>
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: dataenginenfs
  namespace: <namespace>
spec:
  accessModes:
    - ReadWriteMany
  storageClassName: ""
  resources:
    requests:
      storage: 1Mi
---
apiVersion: v1
kind: Secret
type: Opaque
metadata:
  name: tableau-server-in-a-container-secrets
  namespace: <namespace>
stringData:
  license_key: <license_key> # Tableau License Key String
  tableau_username: <tableau_username> # Initial admin username in
Tableau Server
  tableau_password: <tableau_password> # Initial admin password
---
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: tableau-server
```

```
    namespace: <namespace>
    labels:
      app: <service_name>
  spec:
    selector:
      matchLabels:
        app: <service_name>
    replicas: 1
    serviceName: <service_name>
    template:
      metadata:
        labels:
          app: <service_name>
      spec:
        securityContext:
          runAsUser: 999
          fsGroup: 998
          fsGroupChangePolicy: "OnRootMismatch"
        terminationGracePeriodSeconds: 120
        dnsPolicy: "None"
        dnsConfig:
          nameservers:
            - <dns_ip> # DNS IP for resolving container hostnames
          searches:
            - <service_name>.<namespace>.svc.<cluster_
domain>.<example> # SRV Record
            - <namespace>.svc.<cluster_domain>.<example> # SRV Record
            - svc.<cluster_domain>.<example> # SRV Record
            - <cluster_domain>.<example> # SRV Record
          options:
            - name: ndots
              value: "5"
        initContainers: # init containers are optional, to clear
directory content if already exists
          - name: clean-bootstrap-dir
            image: busybox:1.28
            securityContext:
```

```
            runAsUser: 0
            allowPrivilegeEscalation: true
          volumeMounts:
          - name: bootstrap
            mountPath: /docker/config/bootstrap
          command: ['sh', '-c', 'rm -rf /docker/config/bootstrap/*
|| true']
        - name: clean-clone-artifacts-dir
          image: busybox:1.28
          securityContext:
            runAsUser: 0
            allowPrivilegeEscalation: true
          volumeMounts:
          - name: clone
            mountPath: /docker/clone
          command: ['sh', '-c', 'rm -rf /docker/clone/clone-
artifacts || true']
      containers:
      - name: <container_name> # Name of container
        image: <tableau_server_image> # Tableau Server in Container
Image
        env:
        - name: LICENSE_KEY
          valueFrom:
            secretKeyRef:
              name: tableau-server-in-a-container-secrets
              key: license_key
        - name: FILESTORE_MOUNT_PATH
          value: /docker/dataengine
        - name: CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS
          value: /docker/clone/clone-artifacts
        - name: SERVER_FOR_INDEPENDENT_SERVICE_CONTAINERS
          value: "1"
        - name: EXT_REP_JSON_FILE
          value: /docker/external-repository/config.json
        - name: TABLEAU_USERNAME
          valueFrom:
```

```
      secretKeyRef:
        name: tableau-server-in-a-container-secrets
        key: tableau_username
  - name: TABLEAU_PASSWORD
    valueFrom:
      secretKeyRef:
        name: tableau-server-in-a-container-secrets
        key: tableau_password
  resources:
    requests:
      memory: 40Gi
      cpu: 15
    limits:
      memory: 40Gi
      cpu: 15
  ports:
  - containerPort: 8080
  volumeMounts:
  - name: configmount
    mountPath: /docker/config/config.json
    subPath: config.json
  - name: externalrepomount
    mountPath: /docker/external-repository
  - name: datadir1
    mountPath: /var/opt/tableau
  - name: bootstrap
    mountPath: /docker/config/bootstrap
  - name: clone
    mountPath: /docker/clone
  - name: dataengine
    mountPath: /docker/dataengine
  imagePullPolicy: IfNotPresent
  startupProbe:
    exec:
      command:
      - /bin/sh
      - -c
```

```
      - /docker/server-ready-check
    initialDelaySeconds: 300
    periodSeconds: 15
    timeoutSeconds: 30
    failureThreshold: 200
  readinessProbe:
    exec:
      command:
      - /bin/sh
      - -c
      - /docker/server-ready-check
    periodSeconds: 30
    timeoutSeconds: 60
  livenessProbe:
    exec:
      command:
      - /bin/sh
      - -c
      - /docker/alive-check
    initialDelaySeconds: 600
    periodSeconds: 60
    timeoutSeconds: 60
volumes:
- name: configmount
  configMap:
    name: configfile
- name: externalrepomount
  configMap:
    name: extrepojsonfile
- name: datadir1
  persistentVolumeClaim:
    claimName: datadir1
- name: bootstrap
  persistentVolumeClaim:
    claimName: bootstrapnfs
- name: clone
  persistentVolumeClaim:
```

```
        claimName: clonenfs
    - name: dataengine
      persistentVolumeClaim:
        claimName: dataenginenfs
```

バックグラウンダー ポッドの構成

```
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: logvolume
  namespace: <namespace>
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 20Gi
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backgrounder
  labels:
    app: backgrounder
  namespace: <namespace>
spec:
  replicas: 2
  selector:
    matchLabels:
      app: backgrounder
  template:
    metadata:
      labels:
        app: backgrounder
    spec:
      securityContext:
        runAsUser: 999
        runAsGroup: 998
        fsGroup: 998
        fsGroupChangePolicy: "OnRootMismatch"
      hostname: backgrounder
```

```
        dnsPolicy: "None"
        dnsConfig:
          nameservers:
            - <dns_ip> # DNS IP for resolving container hostnames
          searches:
            - <service_name>.<namespace>.svc.<cluster_
domain>.<example> # SRV Record
            - <namespace>.svc.<cluster_domain>.<example> # SRV Record
            - svc.<cluster_domain>.<example> # SRV Record
            - <cluster_domain>.<example> # SRV Record
          options:
            - name: ndots
              value: "5"
        initContainers:
          - name: init-myservice
            image: busybox # This init-container is optional (as long
as there is a mechanism to set the log volume directory permissions
and the pod waits for clone artifacts)
            securityContext:
              runAsUser: 0
              allowPrivilegeEscalation: true
            env:
              - name: CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS
              value: /docker/clone/clone-artifacts
            volumeMounts:
            - name: logmount
              mountPath: /var/log/tableau
            - name: clone-volume
              mountPath: /docker/clone
            command: ['sh', '-c', "chmod 777 /var/log/tableau && while
[ ! -d ${CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS} ]; do sleep
5; done"]
        containers:
        - name: backgrounder
          image: <backgrounder_image> # Backgrounder Image
          ports:
          - containerPort: 8600
```

```
        imagePullPolicy: IfNotPresent
        readinessProbe:
          exec:
            command:
              - /bin/sh
              - -c
              - /tsm_docker_utils/status_check.sh | grep -E
'ACTIVE|BUSY'
          periodSeconds: 30
          timeoutSeconds: 60
        livenessProbe:
          exec:
            command:
              - /bin/sh
              - -c
              - /tsm_docker_utils/status_check.sh | grep -E
'ACTIVE|BUSY'
          initialDelaySeconds: 600
          periodSeconds: 60
          timeoutSeconds: 60
        volumeMounts:
        - name: logmount
          mountPath: /var/log/tableau
        - name: clone-volume
          mountPath: /docker/clone
        - name: dataengine-volume
          mountPath: /docker/dataengine
        - name: temp
          mountPath: /var/opt/tableau/tableau_
server/data/tabsvc/temp
        env:
          - name: ROOT_LOG_DIR
            value: /var/log/tableau
          - name: CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS
            value: /docker/clone/clone-artifacts
          - name: FILESTORE_MOUNT_PATH
            value: /docker/dataengine
```

```
        - name: dataserver
          image: <dataserver_image> # Dataserver Image
          ports:
          - containerPort: 8400
          imagePullPolicy: IfNotPresent
          readinessProbe:
            exec:
              command:
                - /bin/sh
                - -c
                - /tsm_docker_utils/status_check.sh | grep -E
  'ACTIVE|BUSY'
            periodSeconds: 30
            timeoutSeconds: 60
          livenessProbe:
            exec:
              command:
                - /bin/sh
                - -c
                - /tsm_docker_utils/status_check.sh | grep -E
  'ACTIVE|BUSY'
            initialDelaySeconds: 600
            periodSeconds: 60
            timeoutSeconds: 60
          volumeMounts:
          - name: logmount
            mountPath: /var/log/tableau
          - name: clone-volume
            mountPath: /docker/clone
          - name: dataengine-volume
            mountPath: /docker/dataengine
          - name: temp
            mountPath: /var/opt/tableau/tableau_
  server/data/tabsvc/temp
          env:
            - name: ROOT_LOG_DIR
              value: /var/log/tableau
```

```
          - name: CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS
            value: /docker/clone/clone-artifacts
          - name: FILESTORE_MOUNT_PATH
            value: /docker/dataengine
      - name: gateway
        image: <gateway_image> # Gateway Image
        ports:
        - containerPort: 8080
        imagePullPolicy: IfNotPresent
        readinessProbe:
          exec:
            command:
              - /bin/sh
              - -c
              - /tsm_docker_utils/status_check.sh | grep -E
'ACTIVE|BUSY'
          periodSeconds: 30
          timeoutSeconds: 60
        livenessProbe:
          exec:
            command:
              - /bin/sh
              - -c
              - /tsm_docker_utils/status_check.sh | grep -E
'ACTIVE|BUSY'
          initialDelaySeconds: 600
          periodSeconds: 60
          timeoutSeconds: 60
        volumeMounts:
        - name: logmount
          mountPath: /var/log/tableau
        - name: clone-volume
          mountPath: /docker/clone
        - name: dataengine-volume
          mountPath: /docker/dataengine
        - name: temp
          mountPath: /var/opt/tableau/tableau_
```

```
server/data/tabsvc/temp
        env:
          - name: ROOT_LOG_DIR
            value: /var/log/tableau
          - name: CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS
            value: /docker/clone/clone-artifacts
          - name: FILESTORE_MOUNT_PATH
            value: /docker/dataengine
    - name: hyper
      image: <hyper_image> # Hyper Image
      ports:
      - containerPort: 8200
      imagePullPolicy: IfNotPresent
      readinessProbe:
        exec:
          command:
            - /bin/sh
            - -c
            - /tsm_docker_utils/status_check.sh | grep -E
'ACTIVE|BUSY'
        periodSeconds: 30
        timeoutSeconds: 60
      livenessProbe:
        exec:
          command:
          - /bin/sh
          - -c
          - /tsm_docker_utils/status_check.sh | grep -E
'ACTIVE|BUSY'
        initialDelaySeconds: 600
        periodSeconds: 60
        timeoutSeconds: 60
      volumeMounts:
        - name: logmount
          mountPath: /var/log/tableau
        - name: clone-volume
          mountPath: /docker/clone
```

```
        - name: dataengine-volume
          mountPath: /docker/dataengine
        - name: temp
          mountPath: /var/opt/tableau/tableau_
server/data/tabsvc/temp
      env:
        - name: ROOT_LOG_DIR
          value: /var/log/tableau
        - name: CLONE_ARTIFACT_DIR_FOR_INDEPENDENT_CONTAINERS
          value: /docker/clone/clone-artifacts
        - name: FILESTORE_MOUNT_PATH
          value: /docker/dataengine
    volumes:
      - name: clone-volume
        nfs:
          server: <nfs_ip>
          path: <mount_path>
      - name: dataengine-volume
        nfs:
          server: <nfs_ip>
          path: /dataengine
      - name: logmount
        persistentVolumeClaim:
          claimName: logvolume
      - name: temp
        emptyDir: {}
```

スケジュールされたスケーリングの構成

```
apiVersion: v1
kind: ServiceAccount
metadata:
name: backgrounder-scaler-service-account
namespace: <namespace> # Namespace
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
name: scale-backgrounder-pods
namespace: <namespace> # Namespace
subjects:
- kind: ServiceAccount
name: backgrounder-scaler-service-account
roleRef:
kind: ClusterRole
name: cluster-admin
apiGroup: rbac.authorization.k8s.io
---
apiVersion: batch/v1beta1
kind: CronJob
metadata:
name: scale-up-job
namespace: <namespace> # Namespace
spec:
schedule: "0 7 * * *" # Cron Job timing to scale up deployment
replicas
jobTemplate:
spec:
template:
spec:
serviceAccountName: backgrounder-scaler-service-account
restartPolicy: OnFailure
containers:
- name: scale
image: bitnami/kubectl:1.21
```

```
imagePullPolicy: IfNotPresent
args:
- scale
- --replicas=4
- deployment/backgrounder
---
apiVersion: batch/v1beta1
kind: CronJob
metadata:
name: scale-down-job
namespace: <namespace>
spec:
schedule: "0 9 * * *" # Cron Job timing to scale down deployment
replicas
jobTemplate:
spec:
template:
spec:
serviceAccountName: backgrounder-scaler-service-account
restartPolicy: OnFailure
containers:
- name: scale
image: bitnami/kubectl:1.21
imagePullPolicy: IfNotPresent
args:
- scale
- --replicas=2
- deployment/backgrounder
```

クローン構成を消去する Kubernetes ジョブ (オプション)

これは、テスト時に使用できる便利な Kubernetes ジョブです。個別のデプロイを実行時に Tableau Server in a Container が生成したクローン構成をクリアしたい場合は、次のようなジョブを実行して NFS をクリーンアップできます。

```
apiVersion: batch/v1
kind: Job
metadata:
```

```
name: delete-clone-artifacts-job
namespace: manatee-cluster
spec:
template:
spec:
containers:
- name: delete-clone-artifacts
image: busybox:1.28
command: ['sh', '-c', "rm -rf ${CLONE_ARTIFACT_DIR}"]
env:
- name: CLONE_ARTIFACT_DIR
value: /docker/clone/clone-artifacts
securityContext:
runAsUser: 0
allowPrivilegeEscalation: true
volumeMounts:
- name: clone-volume
mountPath: /docker/clone
restartPolicy: Never
volumes:
- name: clone-volume
nfs:
server: <nfs_ip> # IP for shared NFS directory for clone output
path: /clone
```

データ管理 について

重要: 2024 年 9 月 16 日以降、Data Management は独立したアドオン オプションとしては利用できなくなりました。Data Management 機能は、以前に Data Management を購入した場合、または特定のライセンス エディション (Tableau Enterprise (Tableau Server または

> Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

データ管理 は、お客様が Tableau Server または Tableau Cloud 環境で Tableau コンテンツやデータアセットを管理する際に役立つ機能のコレクションです。

Tableau Server 2019.1 より、Tableau Prep Conductor はオンプレミスの Tableau Server 展開で使用できるようになりました。また、バージョン 2019.3 では、Tableau Prep Conductor を Tableau Cloud 展開で使用できるようになりました。Tableau Prep Conductor を使用すると、フローのスケジュールや監視ができます。

Tableau 2019.3 より、データ管理 に Tableau Catalog が追加され、データ管理領域でさまざまな追加機能を使用できるようになりました。Tableau Catalog を使用すると、データの検出、データアセットのキュレーション、データ品質の伝達、インパクト分析の実施、および Tableau コンテンツで使用されるデータ系列の追跡を実行できます。

Tableau 2021.4 より、データ管理 にガバナンス機能とセキュリティ機能 (仮想接続とデータポリシー) が追加されました。仮想接続エディターを使用して、以下を作成できます。

- 共有可能なデータへの中央アクセスポイントを提供する仮想接続
- 接続レベルで行レベルのセキュリティを適用するデータポリシー

データ管理 の機能

次の表は、以下を含む データ管理 の機能を示したものです。

- Tableau Catalog
- Tableau Prep Conductor
- 仮想接続
- データポリシー

Tableau Catalog

機能	説明
メタデータに対するパーミッション	Tableau Catalog では、パーミッションを設定することで、外部資産を表示および管理できるユーザーや、系列を通じて表示されるメタデータを制御できます。
接続エクスペリエンスの拡張 - データ検出	Web または Tableau Desktop のいずれかで作成するかにかかわらず、Tableau Server または Tableau Cloud サイト上のパブリッシュ済みのデータソースやワークブックで使用される特定のデータベースと表を検索し、それらに接続できるようになりました。
拡張検索	Tableau Catalog では、検索を拡張して、列、データベース、およびテーブルに基づいて結果が表示されるようになりました。
外部アセットのタグ付け	Tableau Server および Tableau Cloud のアイテムをタグで分類すると、ユーザーは外部アセット(データベース、ファイル、テーブル、および列)をフィルターできます。
データベースと表の認証	データベースと表の認証を行うと、設定した基準を満たす信頼できるデータをユーザーが見つけやすくなります。
データ品質に関する警告の設定	警告を設定し、古いデータや廃止データなどデータ品質の問題に関してユーザーにアラートを出すことができます。

機能	説明	
系列と影響分析	系列ツールは、データのソースをトレースします。お使いのデータに対する変更の影響を分析したり、影響を受ける可能性のあるユーザーを特定したり、データ関連の更新について、ワークブック、データソース、フローの所有者にメールを送ったり、あるいはデータベースや表の連絡先に電子メールを送信したりできます。	
データの詳細	使用するデータに関する情報を表示することで、ユーザーがパブリッシュされたビジュアライゼーションを理解しやすくします。	
資産に説明を追加する	データベース、表、および列に説明を追加することで、ユーザーが探しているデータを見つけやすくします。	
開発者向けリソース	Tableau REST API - メタデータメソッド	外部アセットをプログラムによって追加、更新、削除し、Tableau コンテンツや説明などの外部アセットにメタデータを追加します。
	Tableau メタデータAPI	Tableau Server または Tableau Cloud にパブリッシュされたコンテンツからメタデータに対するクエリをプログラムによって実行できます。Tableau Server REST API で「メタデータ メソッド」を使用して、特定のメタデータをプログラムで更新します。 **注:** データ管理はメタデータAPI では必要ありません。
	GraphiQL	GraphiQL と呼ばれるインタラクティブなブラウザー内ツールを使用して、メタデータAPI スキーマに対するクエリを調査およびテストします。 **注:** データ管理は GraphiQL では必要ありません。

Tableau Prep Conductor

機能	説明
Tableau Cloud または Tableau Server のヘルプで *フロー タスクをスケジュールします*。	スケジュールされたフロー タスクを作成して、特定の時間または定期的にフローを実行できます。
フローの正常性とパフォーマンスの監視	フローが失敗したらサイトまたはサーバー レベルでメール通知を設定し、中断されたフロー タスクを表示して再開し、エラーとアラートを表示します。
フローの管理ビュー	管理ビューを使用して、フロー、パフォーマンス履歴、およびサーバー レベルまたはサイト レベルで使用されるディスク空き容量に関連するアクティビティを監視します。
Tableau REST API - フロー メソッド	プログラムによってフローをスケジュールします。

仮想接続とデータ ポリシー

機能	説明
仮想接続の作成	キュレートされたデータへの共有可能かつ再利用可能な接続を作成できる Tableau コンテンツ タイプ
行レベルのセキュリティに使用するデータ ポリシーの作成	仮想接続エディターを使用して、接続レベルのデータに行レベルのセキュリティを適用するポリシー条件を含むデータ ポリシーを作成します。
[ユーザーとしてプレビュー] を使用して、行レベルのセキュリティをテストする	[ユーザーとしてプレビュー] を使用して、データ ポリシーをテストし、ユーザーが自分のデータのみを表示できることを確認します。
仮想接続の抽出更新のスケジュールを設定する	接続内のテーブルの抽出更新スケジュールを作成し、その仮想接続を使用するコンテンツのデータが最新であることを確認します。

データ管理 のライセンス

重要: 2024 年 9 月 16 日以降、Data Management は独立したアドオン オプションとしては利用できなくなりました。Data Management 機能は、以前に Data Management を購入した場合、または特定のライセンス エディション (Tableau Enterprise (Tableau Server または Tableau Cloud 用) または Tableau+ (Tableau Cloud 用)) を購入した場合にのみ利用できます。

データ管理 には Tableau Catalog、Tableau Prep Conductor、仮想接続、データ ポリシーが含まれており、Tableau Enterprise を購入すると利用できます。詳細については、アカウント マネージャーに問い合わせるか、Tableau 価格 ページにアクセスしてください。

データ管理 は、ライセンスを認証された Tableau Server 展開でのみ有効にできます。展開には、ライセンスされている本番環境の Tableau Server インストールのほか、本番環境インストールをサポートし、ライセンスされている非本番環境の Tableau Server インストールが含まれています。展開の詳細については、エンドユーザーライセンス契約に関する文書を参照してください。

Tableau Prep Conductor

Tableau Enterprise を購入したら、Tableau Server で Prep Conductor を有効にする必要があります。詳細については、Tableau Server での Tableau Prep Conductor の有効化および構成を参照してください。

- データ管理 がライセンス認証され有効になっていると、Tableau Server または Tableau Cloud でフローのスケジュールや監視ができます。
- データ管理 が削除されているか、非アクティブ化されている場合、または データ管理 のライセンスの期限が切れている場合は、フローをスケジュールする機能が無効になります。
- Tableau Server または Tableau Cloud のライセンスがまだアクティブで有効な場合は、Tableau Server REST API を使用してフローをダウンロードできます。詳細については、「フロー メソッド」を参照してください。

Tableau Catalog

Tableau Enterprise を購入したら、Tableau Server で Catalog を有効にする必要があります。詳細については、「Tableau Catalog の有効化」を参照してください。

- データ管理 がライセンス認証され有効になっていると、Tableau Catalog を使用して、データの検出、データ資産の整理、インパクト分析の実施、および Tableau コンテンツで使用されるデータの系列追跡を実行できます。
- データ管理 が削除されたり、非アクティブ化されたり、ライセンスが期限切れになっても、情報はサーバーに残ります。このとき、Tableau Catalog 固有の情報は Tableau のメタデータ API を使用してのみアクセスでき、製品には表示されなくなります。詳細については、「メタデータ API」を参照してください。
- データ管理 が削除されるか、非アクティブ化されるか、ライセンスが期限切れになると、すべての新規 Tableau Catalog 情報 (表の説明、データ品質に関する警告、列の説明など) 用の書き込み API が無効になります。情報はメタデータ API を使用して引き続き読み取ることができますが、表やデータベースに対するパーミッションを製品で明示的に管理することはできません。

仮想接続とデータポリシー

Tableau Enterprise を購入して Tableau の データ管理 を有効にすると、仮想接続とデータポリシーが Tableau Server で自動的に有効になります。

Tableau の データ管理 を購入してライセンスを取得すると、仮想接続とデータポリシーが自動的に有効になります。

- データ管理 がライセンス認証され有効になっている場合は、仮想接続を使用して、データへの中心的なアクセスポイントを提供する共有可能なリソースを作成できます。また、一元的な行レベルのセキュリティを使用して、ユーザーのデータをフィルターできるデータポリシーを作成することもできます。
- データ管理 が削除されるか、非アクティブ化されるか、ライセンスが有効期限切れになると、情報はサーバーに残りますが、アクセスすることはできません。
- データ管理 が再度アクティブ化されると、情報はサーバーに復元され、アクセス可能になります。

データ管理 ライセンスのしくみ

Tableau Server の展開には、購入したライセンスに応じて、ユーザーベースの展開とコアベースの展開があります。

ユーザーベース

ユーザーベースのライセンス メトリクスでは、Tableau Server を単一の PC またはクラスタ内の複数の PC に展開できます。Tableau Server にアクセスする各ユーザーはライセンスが必要です。管理者はユーザーを追加し、ユーザーにライセンスを発行します。Tableau Server に追加した最初の Creator または Explorer プロダクト キーにより Tableau Server がアクティブ化され、このプロダクト キーがサーバー管理者により使用されます。

データ管理 プロダクト キーを使用すると展開レベルで含まれている機能が有効になり、Tableau Server のライセンスを既に付与されているすべてのユーザーに対して、これらの機能がライセンス認証されます。

コアベース

コアベースのライセンス メトリクスでは、Tableau Server 内のユーザー アカウントの数は制限されません。代わりに、Tableau Server を実行できる PC コアの最大数がライセンスによって指定されます。

> **注:** コア ベースのライセンスを購入して使用する場合は、データ管理 プロダクト キーと Resource Core プロダクト キーの両方を Tableau の展開に適用する必要があります。最初のキーを使用すると、Tableau Server でフローを実行することができますが、Tableau Prep Conductor と 2 番目のキーを使用する場合は、Tableau Prep Conductor のノードにコアが追加されます。すべてのプロダクト キーは、カスタマー ポータルより入手可能です。

通常は、すべての PC のコアの合計数が Tableau Server ライセンスによって許可されている合計数を超えないようにする必要があります。Tableau Server の展開で データ管理 プロダクト キーを有効にする場合、キーには特定の数の Tableau Prep Conductor コアが含まれます。このシナリオでは、すべてのコンピューターのコアの合計数が Tableau Server ライセンスとデータ管理 ライセンスによって許可されている合計数を超えないようにする必要があります。

このトピックでは、Tableau Server ライセンスによってライセンスが付与されたコアを Tableau Server コアと呼び、データ管理 によってライセンスが付与されたコアを Tableau Prep Conductor コアと呼びます。

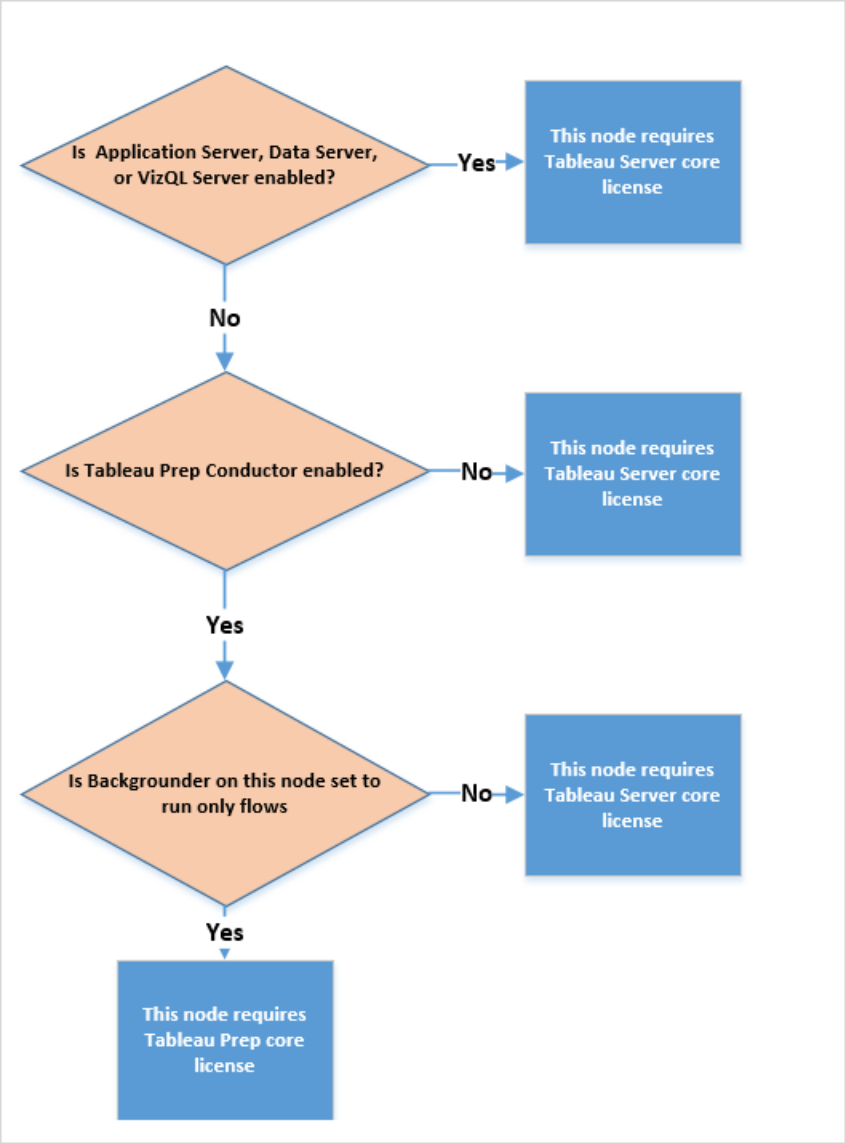
以下は、コアベースのメトリクスでライセンス付与が適用される方法に関連する概念です。

- ノードは、Tableau Server コアと Tableau Prep Conductor コアのうち 1 つのタイプでのみライセンスを受けることができます。
- Tableau Prep Conductor コアは、Tableau Prep Conductor の実行専用のノードで、そのノード上のバックグラウンダーがフロー バックグラウンド ジョブのみを実行するよう設定されている場合に適用されます。この場合、このノードの合計コア数は、データ管理 ライセンスによって許可されるコアの数を超えることができません。このノードで Tableau Prep Conductor、バックグラウンダー、データ エンジン以外のライセンス プロセスが有効になっている場合、このノードには Tableau Server コア ライセンスが必要です。
- 前述のとおり、バックグラウンダー ノード ロールは ノードで使用されるライセンスにも影響します。たとえば、バックグラウンダー ノード ロールがすべてのタイプのジョブを実行するように設定されている場合 (これが既定値です)、このノードは Tableau Server コアによってライセンスを付与されます。ノードの役割の詳細については、「Tableau Server でのノードの役割」を参照してください。

  ノードがライセンスを付与される方法を理解するには、次の表と決定フローを参照してください。

ノードの状態	ノード上のコアのカウント対象	ノードのライセンス付与の方式
次のプロセスのいずれかが有効: • アプリケーション サーバー • バックグラウンダー (ノード ロールはすべてのジョブを実行するように設定) • ファイル スト	Tableau Server コアの合計数。	Tableau Server コア。

ア • データサーバー • VizQL Server		
次のプロセスだけが有効: • Tableau Prep Conductor • バックグラウンダー(ノードロールはフローのみを実行するように設定) • データエンジン	データ管理 を通じて購入した Tableau Prep Core の総数。 **注:** Tableau Prep Conductor コアは使用できないが Tableau Server コアは使用できる場合は、Tableau Server コアが使用されます。	データ管理 に含まれる Tableau Prep Cores。 **注:** Tableau Prep Conductor コアは使用できないが Tableau Server コアは使用できる場合は、Tableau Server コアが使用されます。



ライセンスプロセスの詳細については、「Tableau Server プロセス」を参照してください。

Tableau Server のライセンス発行の詳細については、「ライセンス発行の概要」を参照してください。

Tableau Prep Conductor

Tableau Prep Conductor を使用すると、Tableau Server で利用可能なスケジュール調整機能および追跡機能を活用して、フローを自動的に実行してフロー出力を更新できます。Tableau Prep

Conductor は、Tableau Server バージョン 2019.1 で導入された Tableau データ管理 の一部であり、フローの実行をスケジュールするために有効にする必要があります。

Tableau Server での Tableau Prep Conductor プロセスの詳細については Tableau Prep Conductorを参照してください。

> **注**: バージョン 2020.4 以降、スケジュールに従ってフローを実行する場合や REST API を使用してフローを実行する場合は、データ管理 ライセンスが必要です。
> フローをパブリッシュして Web 上で手動で実行したり、Creator としてサーバー上で直接フローを作成および編集したりする場合は、データ管理 は必要ありません。

Tableau Prep Builder で作成されたフローは、その実行をスケジュールする前に Tableau Server にパブリッシュする必要があります。

フローのパブリッシュは、Tableau Desktop でデータソースやワークブックをパブリッシュすることに類似しています。フローでファイルをパッケージ化するか、データソースへの直接接続を指定して、データの変更に応じてフロー入力を更新できます。フローがデータベースに接続する場合は、認証タイプを指定し、データにアクセスするための認証資格情報を設定します。

フローをパブリッシュして他のユーザーと共有したり、Web 上で編集を続けたりすることもできます。たとえば、不完全なフローを Tableau Server にパブリッシュしてから、編集モードで Web 上でフローを開き、作業を続行します。また、入力ステップ (適切に構成されている) のみでフローを作成して同僚と共有すると、その同僚は、フローをコンピューターにダウンロードし、独自のフローを作成してパブリッシュできます。

フローを実行するには、フローに出力ステップが含まれていて、エラーや互換性のない機能が含まれていないようにする必要があります。フローのパブリッシュの詳細については、「Tableau Server または Tableau Cloud へのフローのパブリッシュ」を参照してください。非互換性の詳細については、「Tableau Prep とのバージョン互換性」を参照してください。

フローの正常性を追跡するのは簡単です。無効な計算や失敗した接続などのエラーが原因でフローが実行されない場合は、Tableau Server でエラーをすぐに修正できます。接続を編集する

か、フローを編集してエラーを修正してから、もう一度パブリッシュして中断したところから再開できます。

次の表は、Tableau Prep Conductor が有効になっているときにデータ管理 がある場合とない場合のフロー管理機能を示しています。

データ管理 と Tableau Prep Conductor が有効になっている場合	データ管理 がない場合
• **[Content (コンテンツ)]** ページでの最近のアクティビティなど、フローに関する詳細を表示および監視します。 • フローを編集します (バージョン 2020.4 以降)。 • **[実行履歴]** タブにフロー実行の結果とエラーを表示します。 • **[管理ビュー]** を使用して、フローのパフォーマンス履歴を追跡する新しいビューなど、サーバーとサイトのアクティビティを監視します。 • REST API を使用してフローを実行します。 • 失敗したフロー実行の詳細なアラートを表示します。 • フローの実行に失敗したことやその理由を通知するメールをフロー所有者に送信するように、メール通知アラートを設定します。 アラートの設定の詳細については、「フローの正常性とパフォーマンスの監視」を参照してください。	• **[Content (コンテンツ)]** ページでの最近のアクティビティなど、フローに関する詳細を表示します。 • フローを編集します (バージョン 2020.4 以降)。 • **[接続]** タブで接続を表示および編集します。

Tableau Server での Tableau Prep Conductor の有効化

Tableau Server へのフローのパブリッシュを開始できるようになる前に、サーバーレベルおよびサイトレベルの設定を構成または確認し、Tableau Server でフローのパブリッシュ、スケジュール、およびモニタリングを行えるよう準備する必要があります。

以下のトピックを確認して、Tableau Prep Conductor のライセンス発行を理解し、Tableau Prep Conductor を有効にする方法をご覧ください。

- Tableau Server での Tableau Prep Conductor の有効化および構成: このトピックでは Tableau Prep Conductor を有効および構成する方法について順を追って説明し、Tableau Server でフローのパブリッシュ、スケジュール、管理、および維持を許可する準備をします。
- データ管理 のライセンス

フローのワークスペースについて

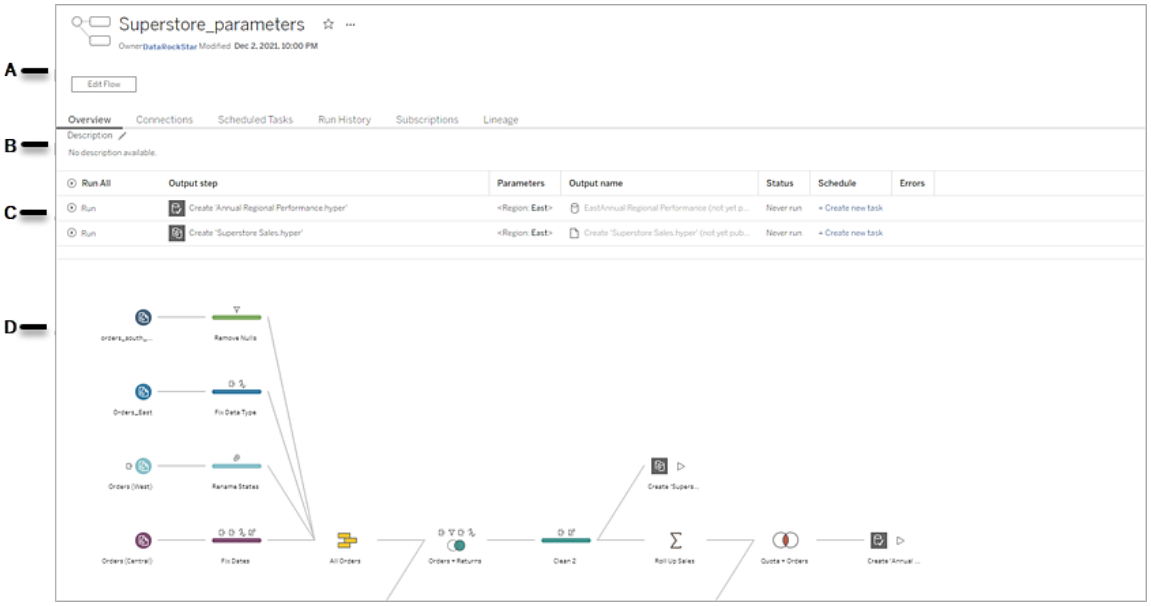
フローをパブリッシュした後、Tableau Server でタスクまたはリンクされたタスク (バージョン 2021.3 以降) をスケジュールすると、フローを定期的に自動で実行して、出力データを最新の状態に保つことができます。フローはいつでも手動で実行できます (データ管理 は必要ありません)。

Tableau Prep Conductor は、Tableau Server で Tableau Desktop からのワークブックやデータソースを管理するための機能と同じような、フロー管理の機能を多く活用します。たとえば、抽出更新と同じように、スケジュールされたフロー タスクやオンデマンド フローの実行はバックグラウンド タスクとしてキューに入れられます。ただし、フローの操作に関しては、いくつかの違いがあります。

フローの [概要] ページ

フローの **[概要]** ページは、メインのランディング ページです。このページでは、フローに関するデータを表示し、フローのスケジュール、監視、保守を行うことができます。データ管理 がない場合は、別のオプションがあります。

リスト内のフローをクリックして、フローの **[概要]** ページを開きます。**[コンテンツ] > [探索] > [すべてのフロー]** から、またはフローを含むプロジェクトを開くことで、そこに移動できます。



A. ヘッダーには、フロー名、フロー所有者、フローの最後更新日がリストされます。バージョン 2020.4 以降では、**[編集]** をクリックすると既存のフローを編集できます。

フローをお気に入りに追加したり、**[その他のアクション]** ••• メニューから、フローの編集、実行、ダウンロード、権限設定、フロー所有者の変更、以前のバージョンのフローへの復元などを行うこともできます。

B. フローの説明を表示および編集し、タグを設定して、他の人がそのフローを検索して見つけられるようにします。

C. フローの出力ステップを、フローに適用されたパラメーター (バージョン 2021.4 以降)、最終更新のステータス、出力が割り当てられたスケジュール、および最後のフロー実行におけるエラーとともに表示します。**[実行]** ボタンをクリックして、すべての出力ステップまたは個々の出力ステップをオンデマンドで実行します。

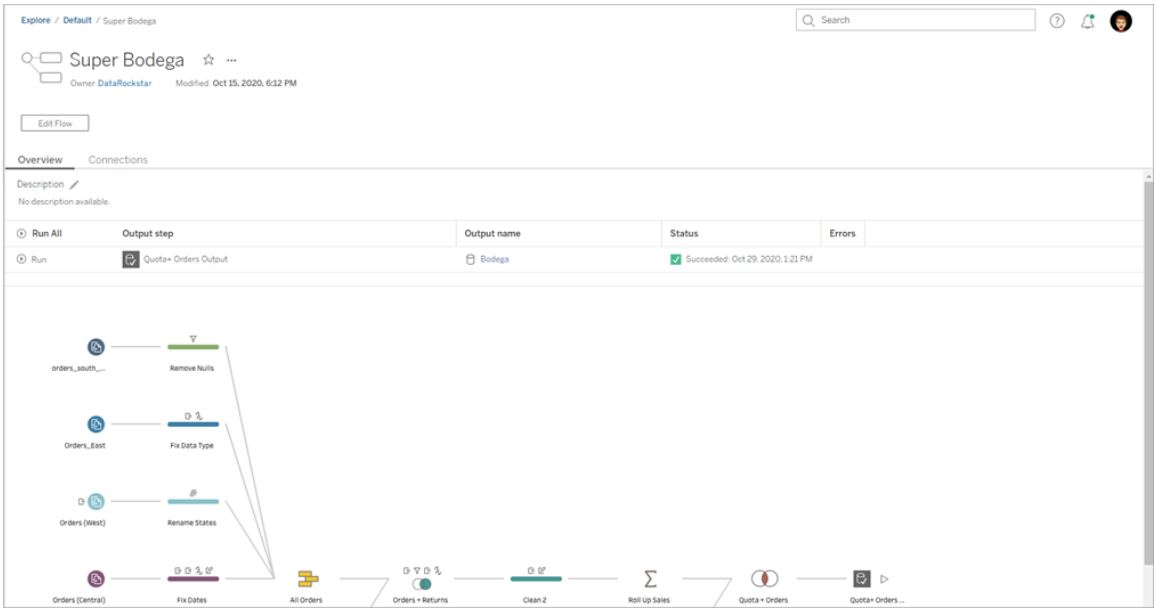
パラメーター	フローにパラメーターが含まれている場合、フローで最後に実行されたパラメー

	ター値が表示され、生成されたすべての出力が**[出力]**列に表示されます。フローが実行されると、パラメーター値を入力するように求められます。 システム パラメーター(バージョン2023.2以降)はフロー実行時に自動的に生成され、システム パラメーターのタイプが**[パラメーター]**列に表示されます。フローに最後に適用されたシステム パラメーター値を確認するには、フローを編集します。 フローでパラメーターを使用する方法の詳細については、Tableau Prep ヘルプの「フローでパラメーターを作成して使用する」を参照してください。
ステータス	フローが正常に実行されると、データ ソースである出力がリンクになり、クリックして**[データ ソース]**ページを開き、データソースに関する詳細情報を表示したり、フロー入力接続を編集したりできます。
スケジュール	**[スケジュール]**フィールドで、出力ステップを割り当てているスケジュールされたタスクを表示します。フロー出力は、1つまたは複数のタスクに割り当てることができます。 スケジュールがまだ割り当てられていない場合は、**[新しいタスクの作成]**をクリックして、出力ステップをスケジュールに追加します。フローをすぐに実行して特定の出力ステップを更新するには、行の左側にある**[実行]** ボタンをクリックします。
エラー	フローにエラーがある場合、フローの実行は失敗します。接続エラーは、フローの**[接続]**タブに移動し、入力接続を編集することで直接解決できます。 その他のフロー エラーを解決するには、フローを編集してからパブリッシュし直して、フローの実行をもう一度試行します。以前のバージョンの Tableau Prep Builder を使用している場合は、**[その他のアクション]** メニューから、Tableau Prep Builder でフローをダウンロードして開き、パブリッシュし直してフローの実行をもう一度試すこともできます。

D. フローのイメージを表示します。

データ管理 のないフロー概要ページ

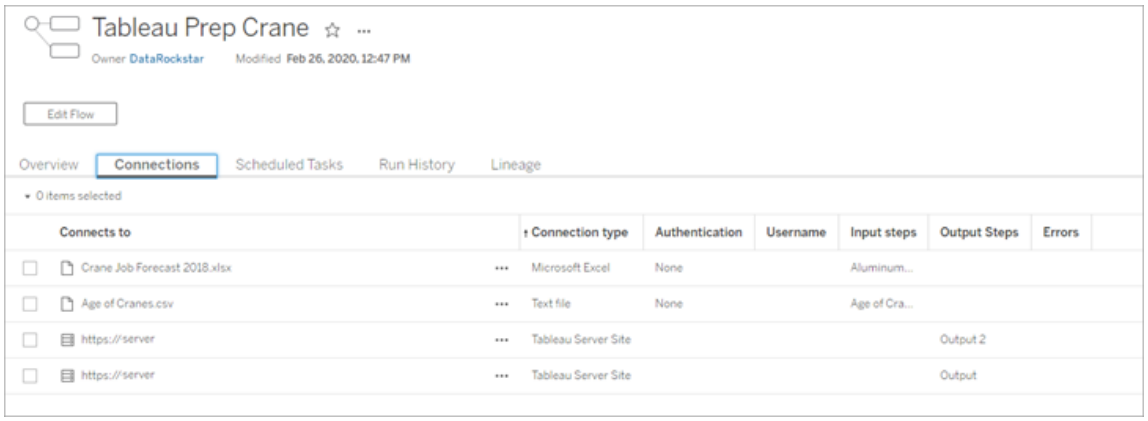
サーバーに データ管理 がインストールされていなくても、フローを Tableau Server にパブリッシュできますが、フローを管理するためのオプションは少ししか表示されません。



フロー接続ページ

フローの入力場所と出力場所、接続タイプ、認証設定、入力ステップと出力ステップ、および接続エラーを表示します。フローをパブリッシュするときは認証を設定できます。詳細については、「フローのパブリッシュ」を参照してください。

データベース入力タイプの場合、入力接続の **[その他のアクション]** ••• メニューをクリックすると、接続を編集してサーバー名、ポート、ユーザー名、パスワードを変更することができます。



フローのスケジュールされたタスクのページ (データ管理必須)

フローが割り当てられているスケジュール、それらのスケジュールに含まれている出力、およびフローに適用されているパラメーターを表示します (バージョン 2021.4 以降)。管理者は、スケジュールのリンクをクリックして **[スケジュール]** ページを開くと、そのスケジュールに割り当てられているフローのリストを表示できます。フローをスケジュールに割り当てる方法の詳細については、Tableau Cloud または Tableau Server ヘルプの*「フロー タスクのスケジュールを設定する」*を参照してください。

スケジュール上の出力や、リンクされたタスク (バージョン 2021.3 以降) に割り当てられたタスクを表示するには、**[スケジュールの種類]** 列のリンクをクリックします。

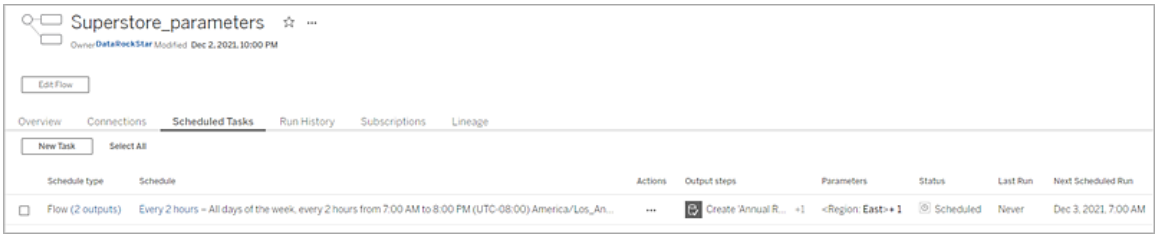
このページから、新しいタスクを追加したり、既存のタスクを管理したりすることもできます。既存のタスクに対してアクションを実行するには、タスク カードのチェック ボックスを選択し、**[アクション]** ドロップダウン メニューをクリックしてタスクを実行、編集、または削除します。

設定した回数連続してフローを実行できなかった場合、フローは自動的に中断されます。

そのステータスは、**[概要]** タブとこのタブで確認できます。中断されたタスクは、このメニューから再開できます。

中断されたフロー タスクのしきい値を設定する方法については、「ステップ 5: オプションのサーバー構成」を参照してください。中断されたフロー タスクの詳細については、エラーを表示して解決するを参照してください。

注: フローの [スケジュールされたタスク] ページは、バージョン 2021.3 で再設計されました。サーバーのバージョンによって、ビューが異なる場合があります。

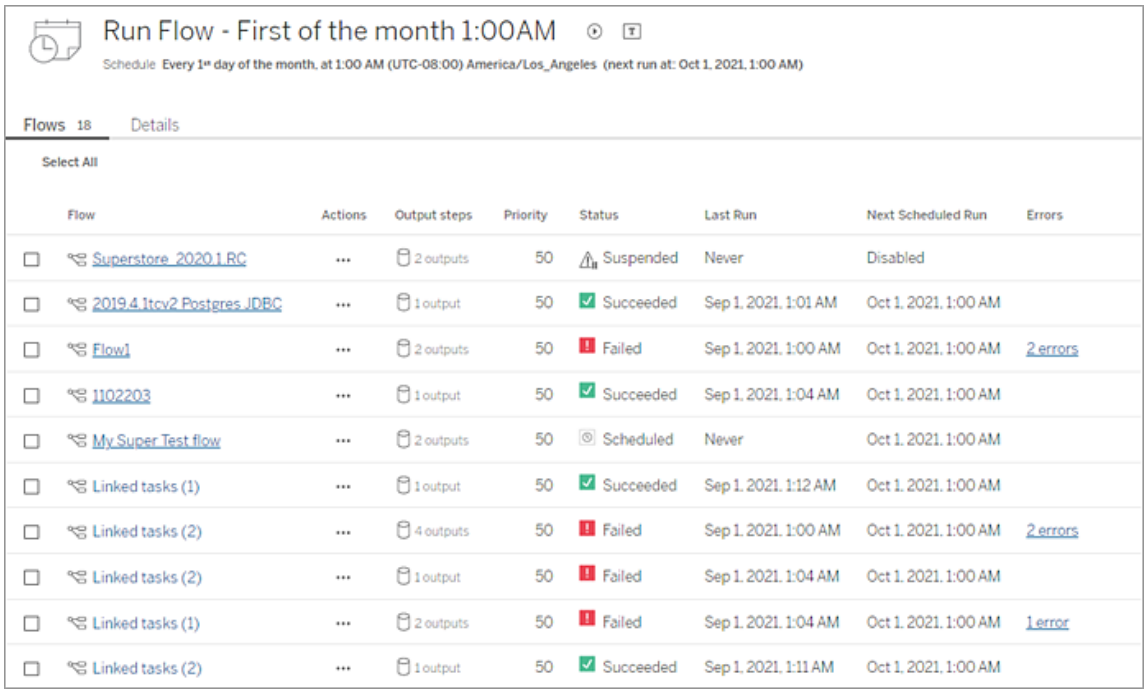


スケジュール ページ

[スケジュール] ページで、スケジュールに割り当てられたフローと、フロー実行の詳細を表示できます。スケジュールにリンクされたタスクが含まれている場合 (バージョン 2021.3 以降)、リンクされたタスクに含まれるフローの数が表示されます。

スケジュールをオンデマンドで実行し、スケジュールに割り当てられたすべてのフローを実行できます。1 つまたは複数のフローを選択してから、**[アクション]** メニューを使用して、フローのスケジュールまたは優先度を変更したり、スケジュールから選択したフローを削除したり、中断されたフローを再開したりすることもできます。

スケジュールの設定方法については、「ステップ 3: フロー タスクのスケジュールを作成する」を参照してください。

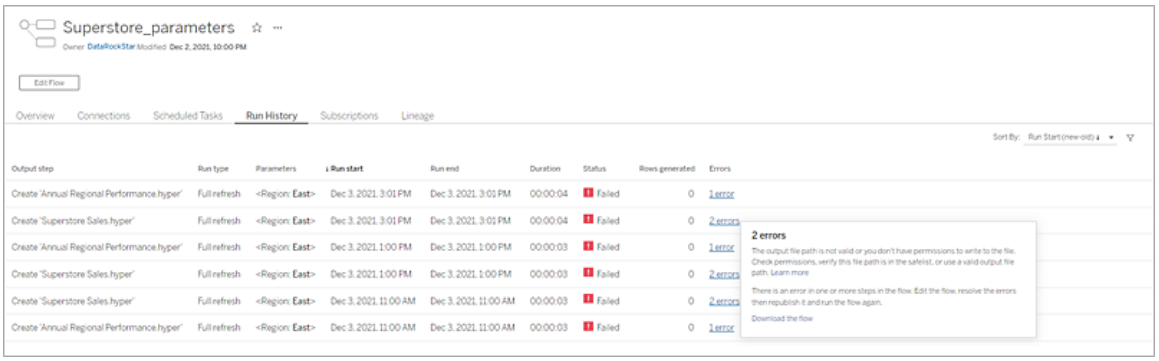


フロー実行履歴 (データ管理必須)

フローの実行履歴のリストを表示、検索、および並べ替えます。このページには、実行タイプ、各フロー実行に含まれるフローに適用されるパラメーター値 (バージョン 2021.4 以降)、生成された行の期間と数など、フロー実行に関する詳細も含まれています。

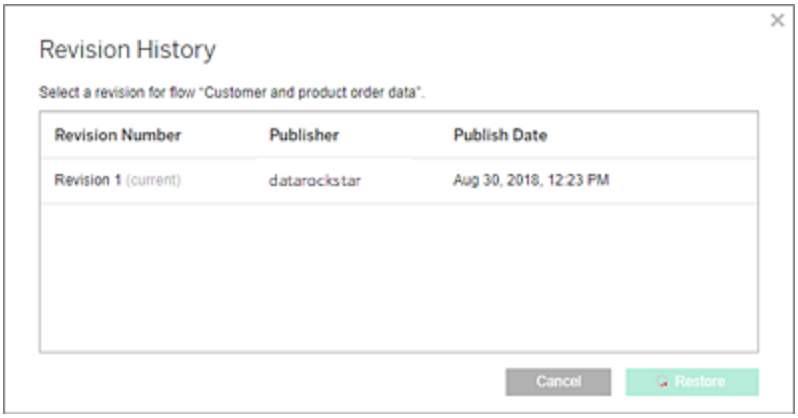
フロー出力にエラーがある場合、エラーにカーソルを合わせるとメッセージを表示できます。該当する場合は、エラー メッセージ内の **[接続に移動]** リンクをクリックし、**[接続]** ページに移動して接続エラーを修正します。フローを直接編集してエラーを修正することも、**[フローのダウンロード]** をクリックして Tableau Prep Builder でフロー エラーをダウンロードして修正し、フローをもう一度パブリッシュして Tableau Prep Conductor で管理を継続することもできます。

注: フローの実行履歴は、フローが削除されない限り保持されます。



フロー リビジョン履歴

フローを以前のバージョンに戻す必要がある場合は、そのフローの **[その他のアクション]** ••• メニューで、**[リビジョン履歴]** を選択します。**[リビジョン履歴]** ダイアログで、元に戻したいフロー バージョンをリストから選択します。



実行可能なユーザー

サーバー管理者は、データ管理 ライセンス キーを認証できます。

サーバー管理者は、Tableau Prep Conductor を有効にできます。

Creator は、フローを手動で作成、編集、および実行できます。データ管理 がインストールされている場合、Creator はスケジュールに従ってフローを実行できます。

Tableau Server での Tableau Prep Conductor の有効化および構成

Tableau Server バージョン 2019.1 以降でサポートされています。

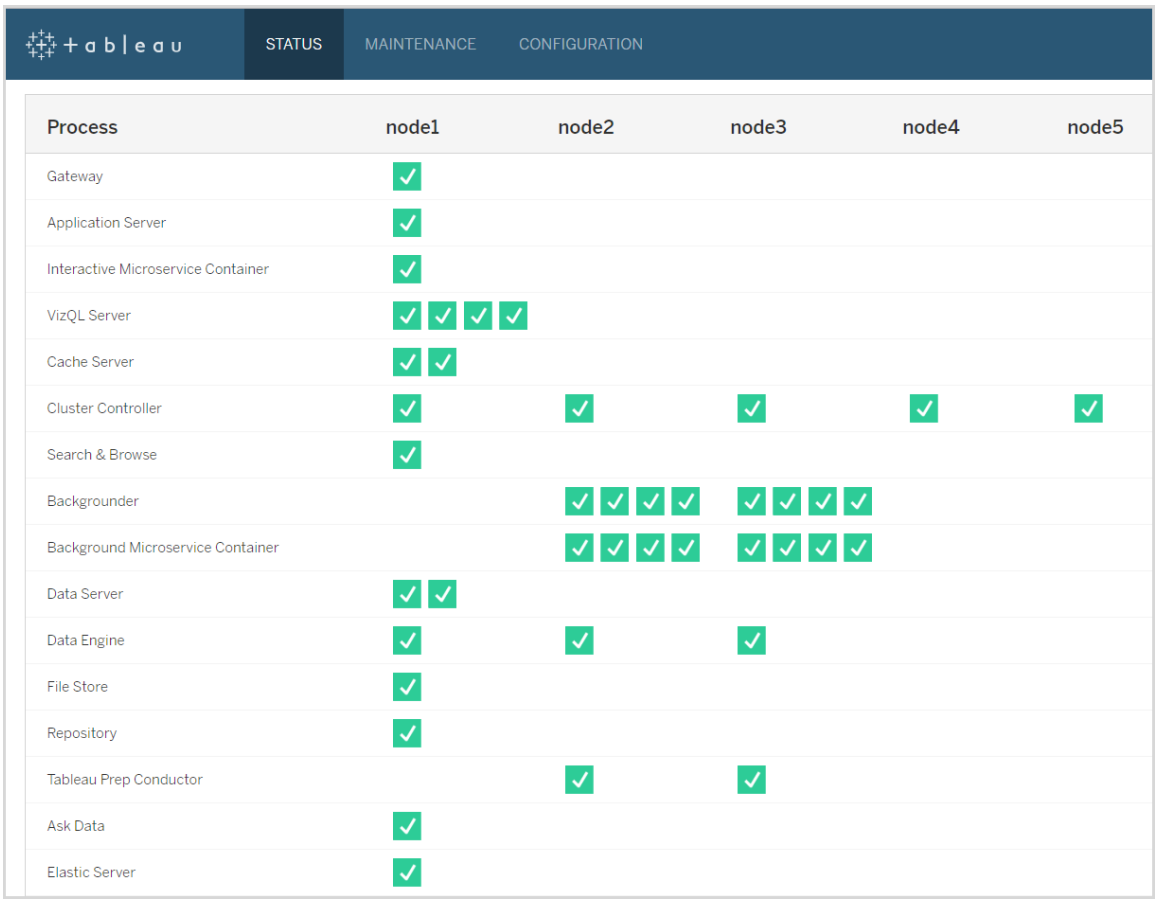
Tableau Prep Conductor は データ管理 を介して、ユーザーベースまたはコアベースである展開ごとにライセンスされています。展開には、ライセンスされている本番環境の Tableau Server インストールのほか、本番環境インストールをサポートし、ライセンスされている非本番環境の Tableau Server インストールが含まれています。展開の詳細については、「Tableau 導入ガイド」を参照してください。

Tableau Prep Conductor のライセンス発行の仕組みについては、データ管理 のライセンスを参照してください。

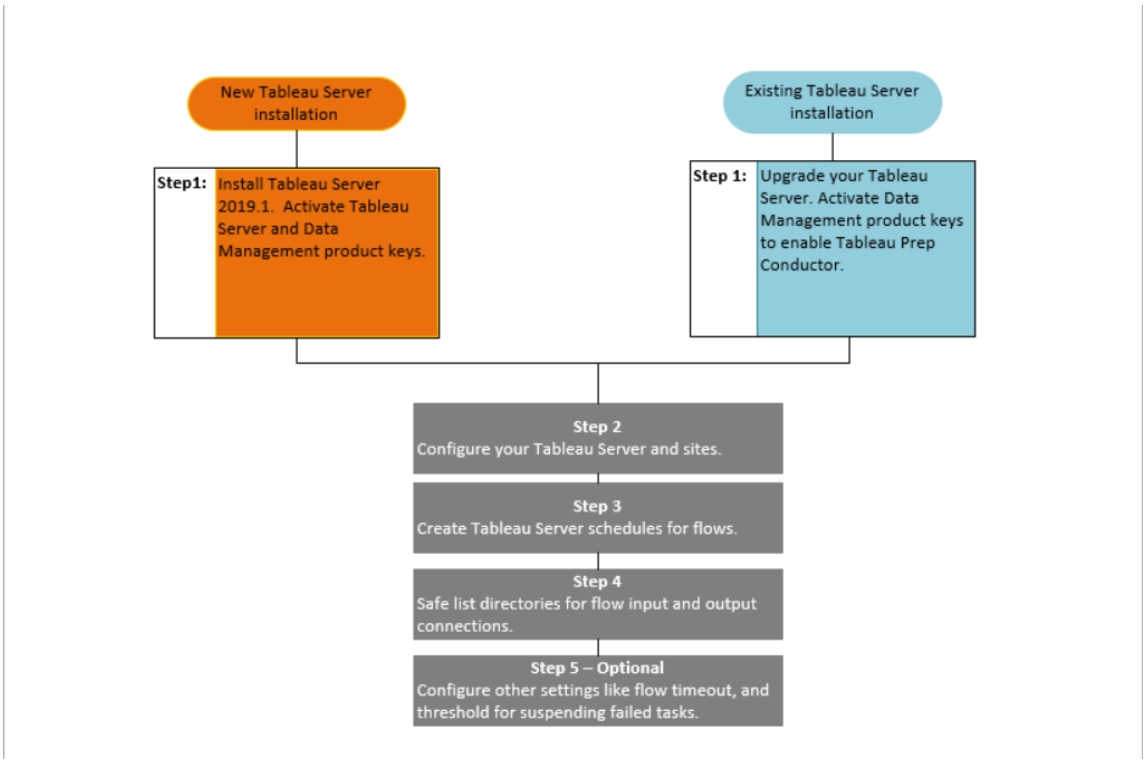
Server のトポロジ

Tableau Server をインストールし、データ管理 プロダクトキーを使用して Tableau Prep Conductor を有効にすると、セットアップ プログラムによって Tableau Prep Conductor が既定で自動的に有効になります。

分散インストールでは、バックグラウンダーをインストールしている任意のノードで Tableau Prep Conductor の 1 つのインスタンスが既定で有効になっています。下の例では、バックグラウンダーも有効になっているノード2 と3 で Tableau Prep Conductor が有効になっていますが、ノード1、4、5 では有効になっていません。



以下は、そのワークフローを視覚的に表現したものです。



次の手順:

Tableau Server の新規インストール: 手順 1 (新規インストール): Tableau Server を Tableau Prep Conductor と一緒にインストール

既存の Tableau Server のインストール: 手順 1 (既存インストール): Tableau Prep Conductor の有効化

実行可能なユーザー

サーバー管理者は Tableau Server をインストールし、Tableau Prep Conductor を有効にすることができます。

サーバーレベルの設定は Tableau Server 管理者が構成でき、サイトレベルの設定は、Tableau Server およびサイトの管理者が構成できます。

手順 1 (新規インストール): Tableau Server を Tableau Prep Conductor と一緒にインストール

このトピックでは、Tableau Server の新規インストールで Tableau Prep Conductor を有効にする方法について説明します。

Tableau Prep Conductor は、Tableau Server バージョン 2019.1 以降でのみサポートされています。

Tableau Prep Conductor は データ管理 を介して、展開ごとにライセンスされています。展開には、ライセンスされている本番環境の Tableau Server インストールのほか、本番環境インストールをサポートし、ライセンスされている非本番環境の Tableau Server インストールが含まれています。展開の詳細については、「Tableau 導入ガイド」を参照してください。

インストールの前に

本番環境の Tableau Server インストールのトポロジとして、フロー実行のための専用ノードを使用することが推奨されます。現在、単一ノードの Tableau Server インストールを検討している場合は、2 番目のノードを追加し、それをフローの実行専用にすることをお勧めします。

- Tableau Server および Tableau Prep Conductor のハードウェア推奨事項を確認してください。
    - Windows での Tableau Server インストールの最小ハードウェア要件と推奨事項。
    - Linux での Tableau Server インストールの最小ハードウェア要件と推奨事項。

Tableau Server のインストールおよび Tableau Prep Conductor の有効化

次のトピックの指示に従って Tableau Server をインストールします。

Windows: Tableau Server のインストール

Linux: Tableau Server のインストール トピック

[ライセンス認証] の手順まで来たら、Tableau Server プロダクトキーを使用して Tableau Server をライセンス認証します。

すべてのプロダクトキーは、カスタマー ポータルより入手可能です。

パブリックゲートウェイの設定を構成する

Tableau Server が次のいずれかで設定されている場合です。

- ロードバランサを使用して、要求をゲートウェイ間で分散している。
- リバースプロキシを使用して、外部 (インターネット) クライアントの要求を認証したり、SSL ベースの暗号化やオフロードを実行している。

パブリックゲートウェイの次の設定を構成する必要があります。

`tsm configuration set -k gateway.public.host -v <name>` (Tableau Server へのアクセスにユーザーが使用している URL である必要があります)

```
tsm configuration set -k gateway.public.port -v 443
```

ゲートウェイ設定の構成の詳細については、「Tableau Server プロキシの設定」を参照してください。

Tableau Prep Conductor を有効にする

データ管理プロダクトキーを Tableau Server に追加するには、次の手順を使用します。このプロセスでは、Tableau Server を再起動する必要があります。

> **注:** コアベースのライセンス発行を使用している場合は、データ管理プロダクトキーとリソースコアプロダクトキーの両方をTableau の展開に適用する必要があります。最初のキーを使用すると、Tableau Server でフローを実行することができますが、Tableau Prep Conductor と2番目のキーを使用する場合は、Tableau Prep Conductor のノードにコアが追加されます。すべてのプロダクトキーは、カスタマー ポータルより入手可能です。

1. Tableau Server を実行しているコンピューターがフォワードプロキシ経由でインターネットに接続するように構成されている場合は、続行する前に「フォワードプロキシを使用したプロダクトキー操作の構成」のトピックの手順に従ってください。
2. ブラウザーで TSM を開きます。

```
https://<tsm-computer-name>:8850
```

3. **[構成]** タブで **[ライセンス発行]** をクリックし、**[アクティブ化]** をクリックします。
4. **データ管理 プロダクト キー**を入力するか貼り付けて、**[アクティブ化]** をクリックします。
5. [登録] ページで情報をフィールドに入力し、**[登録]** をクリックします。
6. 登録が完了したら、プロンプトに従って Tableau Server を再起動します。

Tableau Prep Conductor が有効で実行中であることの確認

データ管理 プロダクト キーをライセンス認証すると、バックグラウンダーが有効な任意のノードで Tableau Prep Conductor の単一インスタンスが自動的に有効になります。

次の手順により、有効で実行中であることを確認します:

1. ブラウザーを開き、Tableau Server の URL を入力して、専用の TSM Web UI ポートを追加します。いくつかの例として、URL は以下のようになります。

   https://localhost:8850/ (サーバー コンピューター上で直接作業している場合)

   https://MarketingServer:8850/ (サーバー名がわかっている場合)

   https://10.0.0.2:8850/ (サーバーの IP アドレスがわかっている場合)

   表示されるサインイン ページで、管理者のユーザー名とパスワードを入力します。

   > **注:** Tableau Server はインストール プロセスの一環として自己署名証明書を作成して構成します。この証明書は、TSM Web UI へのトラフィックを暗号化するために使用されます。これは自己署名証明書であるため、デフォルトではブラウザーで信頼されません。そのため、接続できるようになる前に、お使いのブラウザーに証明書の信頼性に関する警告が表示されます。

2. Tableau サービス マネージャーの Web インターフェイスで、**[ステータス]** タブをクリックし、ステータスを表示します。

   - Tableau Prep Conductor が有効になり、実行中であると、プロセスのリストに Tableau Prep Conductor が 1 つ以上のノードで **[アクティブ]** として表示されます。Tableau Prep Conductor が有効になっていないと、プロセスのリストに Tableau Prep Conductor は表示されるものの、いずれのノードでもステータス情報はありません。

     Tableau Prep Conductor が有効になっていない:

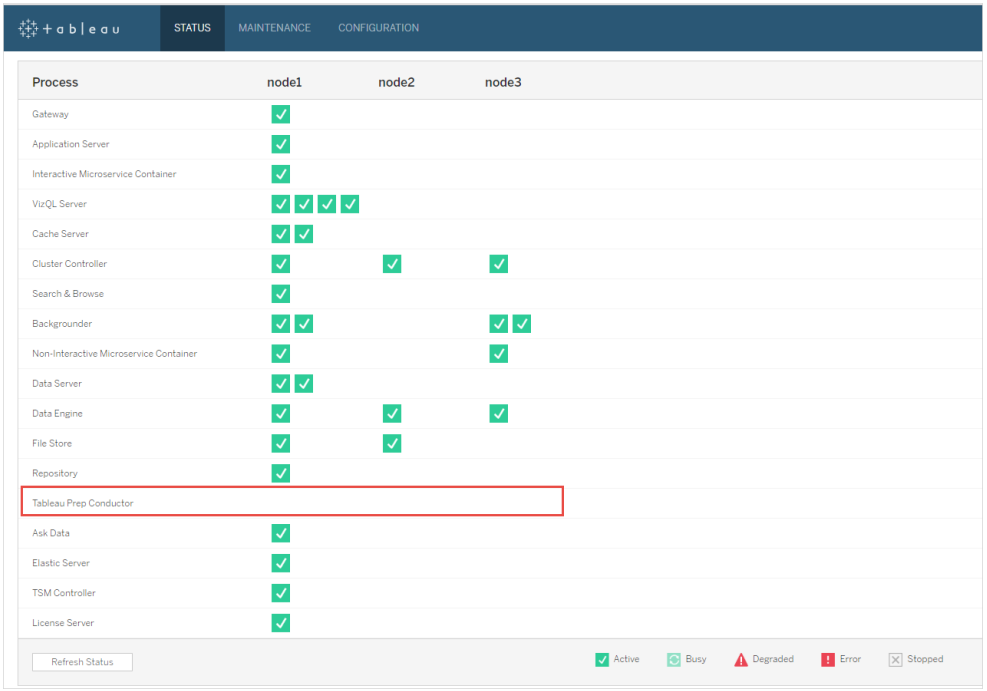
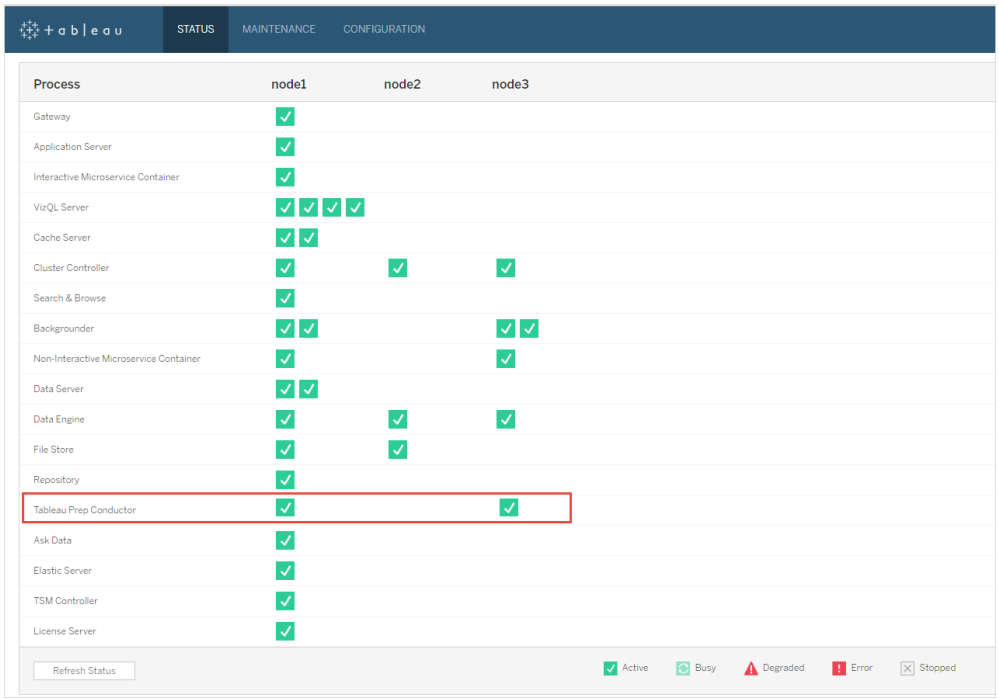


     Tableau Prep Conductor が有効で実行中である。以下の画像では Tableau

Prep Conductor が node1 と node3 で有効になっている:



ノードを Tableau Prep Conductor 専用にする

フローの実行専用にしようと考えているノードで、バックグラウンダープロセスがまだ有効でなければ有効にします。このノードでは VizQL サーバーなど、他のプロセスは実行しないことをお勧めします。

このノードはフローの実行専用にしているため、フロー タスクのみを実行するようにバックグラウンダーを構成する必要があります。既定では、バックグラウンダープロセスは、フロー、抽出更新、サブスクリプションなど、すべてのタイプのタスクを実行します。詳細については、「Tableau Server でのノードの役割」を参照してください。

その専用ノードで以下の tsm コマンドを実行し、フロー タスクのみを実行します。

1. 以下のコマンドを実行し、このノードのバックグラウンダーでフロー タスクの実行のみを許可します。

   ```
   tsm topology set-node-role -n node1 -r flows
   ```

2. 最初のノードのノードの役割を、フローなしに設定します。このノードのバックグラウンダーは、次のフローを除くすべてのジョブを実行します。

   ```
   tsm topology set-node-role -n node1 -r no-flows
   ```

3. 変更を適用し、Tableau Server を再起動します。

   ```
   tsm pending-changes apply
   ```

分散インストール

Tableau Server インストールに 2 つを超えるノードがある場合は、フロー以外のすべてのタスクを実行するように他のノードを構成することもできます。

1. フローを許可しないようにノードを制限します。このコマンドを使用すると、このノードから Tableau Prep Conductor が削除され、このノードのバックグラウンダーでフロータスクが実行されません。

   ```
   tsm topology set-node-role -n node1 -r no-flows
   ```

2. 変更を適用し、Tableau Server を再起動します。

   ```
   tsm pending-changes apply
   ```

次の手順

手順 2 - Tableau Server のフロー設定の構成

実行可能なユーザー

サーバー管理者は Tableau Server をインストールし、Tableau Prep Conductor を有効にすることができます。

手順 1 (既存インストール): Tableau Prep Conductor の有効化

このトピックでは、Tableau Server の既存インストールで Tableau Prep Conductor を有効にする方法について説明します。

Tableau Prep Conductor は、Tableau Server バージョン 2019.1 以降でのみサポートされています。Tableau Server 2018.3 以前を使用している場合は、まず Tableau Server を 2019.1 にアップグレードしてから、Tableau Server インストールで Tableau Prep Conductor を有効にする必要があります。

Tableau Prep Conductor は データ管理 を介して、ユーザーベースまたはコアベースである展開ごとにライセンスされています。展開には、ライセンスされている本番環境の Tableau Server インストールのほか、本番環境インストールをサポートし、ライセンスされている非本番環境の Tableau Server インストールが含まれています。展開の詳細については、「Tableau 導入ガイド」を参照してください。

このトピックでは、Tableau Server の既存インストールで Tableau Prep Conductor を有効にする方法について説明します。

アップグレードする前に

アップグレードの準備:

- アップグレード前に知っておく事柄
- Tableau Prep Conductor のライセンス発行
- Tableau Server のハードウェア要件と推奨事項

パブリックゲートウェイの設定を構成する

Tableau Server が次のいずれかで設定されている場合です。

- ロードバランサを使用して、要求をゲートウェイ間で分散している。
- リバースプロキシを使用して、外部 (インターネット) クライアントの要求を認証したり、SSL ベースの暗号化やオフロードを実行している。

パブリックゲートウェイの次の設定を構成する必要があります。

`tsm configuration set -k gateway.public.host -v <name>` (Tableau Server へのアクセスにユーザーが使用している URL である必要があります)

```
tsm configuration set -k gateway.public.port -v 443
```

ゲートウェイ設定の構成の詳細については、「Tableau Server プロキシの設定」を参照してください。

ユーザーベース ライセンスを使用した Tableau Server インストール

本番環境の Tableau Server インストールのトポロジとして、フロー実行のための専用ノードを使用することが推奨されます。詳細については、「Tableau Server の最小ハードウェア要件と推奨事項」を参照してください。

Tableau Server の単一ノード インストール

現在、単一ノードの Tableau Server インストールを使用している場合、2 番目のノードを追加し、それをフローの実行専用にすることをお勧めします。

1. 以下のトピックの情報を使用し、現在の Tableau Server インストールでアップグレードを実行します。

    - Windows
    - Linux

    [ライセンス認証] の手順まで来たら、Tableau Server プロダクト キーを使用して Tableau Server をライセンス認証します。

    すべてのプロダクト キーは、カスタマー ポータルより入手可能です。

2. インストールが完了したら、データ管理 プロダクト キーを追加してお使いのノードで Tableau Prep Conductor を有効にします。データ管理 プロダクト キーは、他のサーバー キーと同様、カスタマー ポータルより入手可能です。

    - Tableau サービス マネージャーの Web インターフェイスの **[構成]** タブで **[ライセンス発行]** をクリックし、**[ライセンス認証]** をクリックします。

- 新しいプロダクトキーを入力するか貼り付けし、**[ライセンス認証]** をクリックします。
- **[登録]** ページで情報をフィールドに入力し、**[登録]** をクリックします。

3. サーバーの再起動するようにメッセージが表示されます。サーバーを再起動し、Tableau Prep Conductor が有効になり、実行中であることを確認します。

   - Tableau サービス マネージャーの Web インターフェイスで、**[ステータス]** タブをクリックし、ステータスを表示します。Tableau Prep Conductor が有効になり、実行中であると、プロセスのリストに Tableau Prep Conductor が **[アクティブ]** として表示されます。Tableau Prep Conductor が有効になっていないと、プロセスのリストに Tableau Prep Conductor は表示されるものの、ステータス情報はありません。

     Tableau Prep Conductor が有効になっていない:

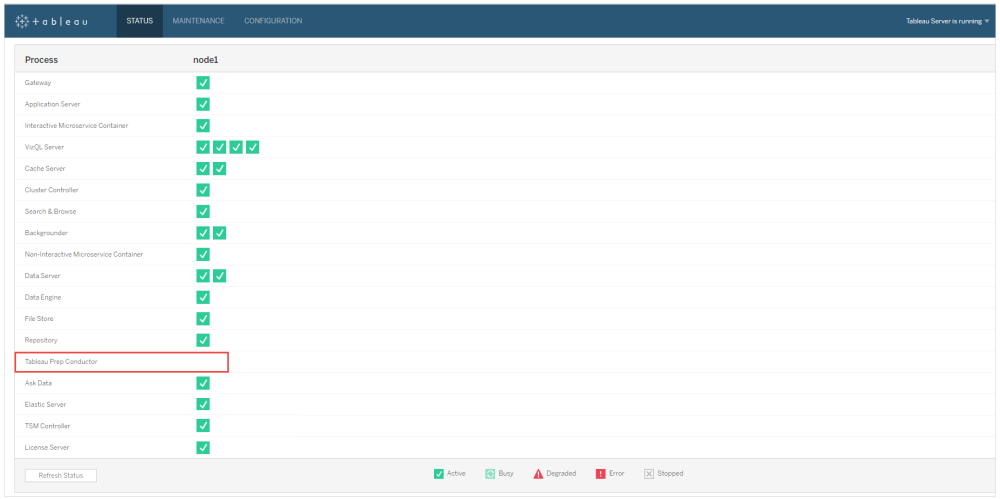
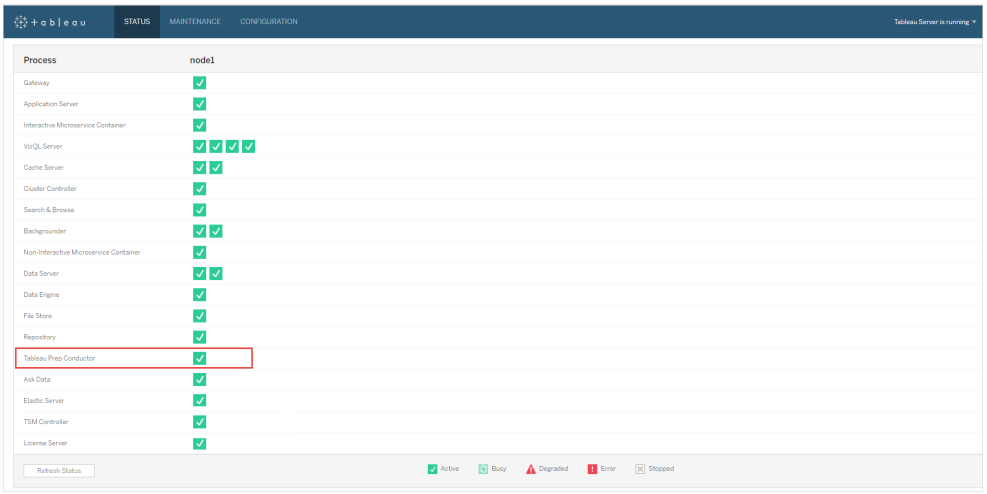


Tableau Prep Conductor が有効で実行中である:



4. 2 番目のノードを Tableau Server インストールに追加します。インストーラーにより、クラスタコントローラーなど、特定の必須プロセスが有効にされます。バックグラウンダープロセスはスケジュールされたフロータスクの実行に必要であるため、これを有効にします。バックグラウンダープロセスを有効にすると、インストーラーは自動的に、そのノード上のデータエンジンおよび Tableau Prep Conductor の単一インスタンスを有効にします。このノード上に他のプロセスを追加しないでください。

5. このノードをフロータスクの実行専用にするには、次のコマンドを実行します。ノードの役割の詳細については、「Tableau Server でのノードの役割」を参照してください。

   - 各ノードのサービスのリストを表示するには、専用ノードの nodeID を取得します。

     ```
     tsm topology list-nodes -v。
     ```

   - 上記のコマンドを実行して取得した nodeID を使用し、専用ノードのノードの役割を設定します。

     ```
     tsm topology set-node-role -n <nodeID> -r flows。
     ```

   - 変更を適用し、サーバーを再起動します。

     ```
     tsm pending-changes apply。
     ```

- ステータスを確認し、すべてのプロセスが起動して実行中であり、正しく構成されていることを確認します。

  ```
  tsm status -v。
  ```

Tableau Prep Conductor の Tableau Server インストールへの追加に成功しました。

Tableau Server の分散インストール

1. 以下のトピックの情報を使用し、現在の Tableau Server インストールでアップグレードを実行します。

   - Windows
   - Linux

   [ライセンス認証] の手順まで来たら、Tableau Server プロダクトキーを使用して Tableau Server をライセンス認証します。

   すべてのプロダクトキーは、カスタマー ポータルより入手可能です。

2. インストールが完了したら、データ管理 プロダクトキーを追加して Tableau Prep Conductor を有効にします。バックグラウンダープロセスが既に有効になっているノードで Tableau Prep Conductor が自動的に有効になります。データ管理 プロダクトキーは、他のサーバー キーと同様、カスタマー ポータルより入手可能です。

   - Tableau サービス マネージャーの Web インターフェイスの **[構成]** タブで **[ライセンス発行]** をクリックし、**[ライセンス認証]** をクリックします。
   - 新しいプロダクトキーを入力するか貼り付けし、**[ライセンス認証]** をクリックします。
   - **[登録]** ページで情報をフィールドに入力し、**[登録]** をクリックします。

3. サーバーの再起動するようにメッセージが表示されます。サーバーを再起動し、Tableau Prep Conductor が有効になり、実行中であることを確認します。

- Tableau サービス マネージャーの Web インターフェイスで、**[ステータス]** タブをクリックし、すべてのプロセスのステータスを表示します。Tableau Prep Conductor が有効になり、実行中であると、プロセスのリストに Tableau Prep Conductor が **[アクティブ]** として表示されます。Tableau Prep Conductor が有効になっていないと、プロセスのリストに Tableau Prep Conductor は表示されるものの、ステータス情報はありません。

Tableau Prep Conductor が有効になっていない:

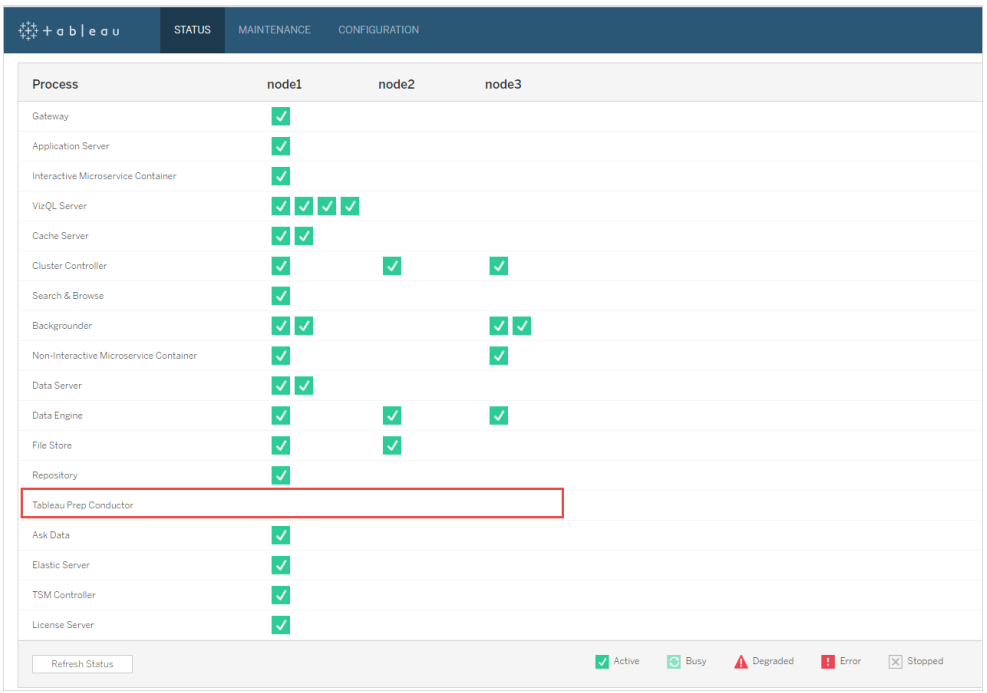
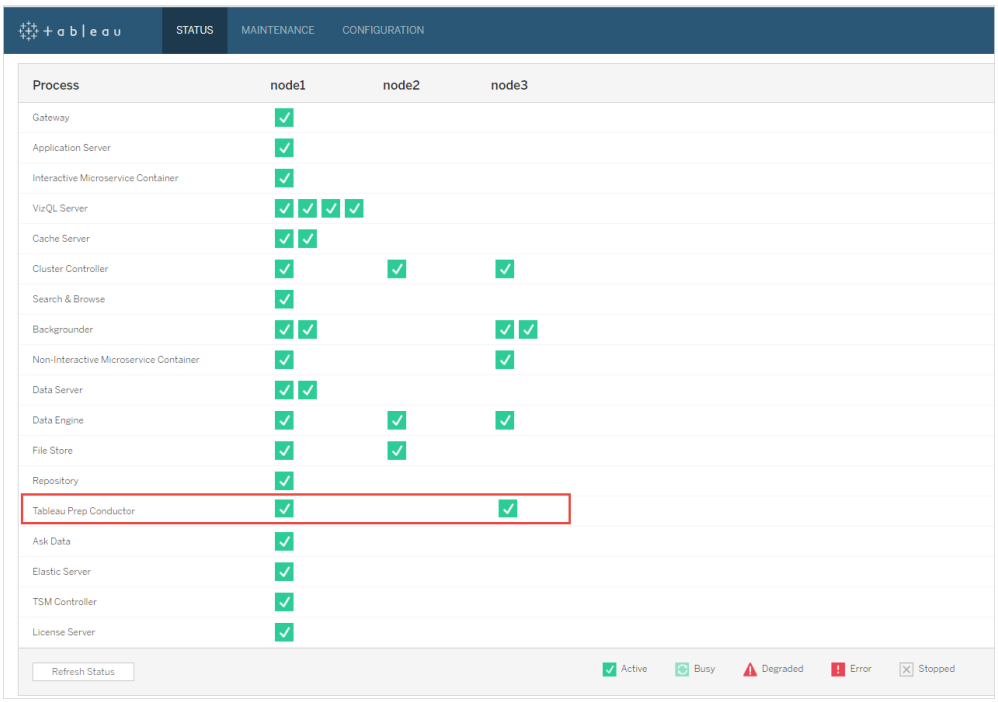


Tableau Prep Conductor が有効で実行中である:



4. 新しいノードを Tableau Server インストールに追加します。インストーラーにより、クラスタコントローラーなど、特定の必須プロセスが有効にされます。バックグラウンダープロセスはスケジュールされたフロータスクの実行に必要であるため、これを有効にします。バックグラウンダープロセスを有効にすると、インストーラーは自動的に、そのノード上のデータエンジンおよび Tableau Prep Conductor の単一インスタンスを有効にします。このノード上に他のプロセスを追加しないでください。

   注: 専用ノードの数は、調整サービスアンサンブルの合計数に反映されます。新しい専用ノードを含むクラスタ内のノードの合計数に応じ、新しいノードで調整サービスの展開が必要になる場合があります。詳細については、「調整サービス アンサンブルの展開」を参照してください。

5. このノードをフローに関係する操作の実行専用にするには、次のコマンドを実行します。ノードの役割の詳細については、「Tableau Server でのノードの役割」を参照してください。

- 各ノードのサービスのリストを表示するには、専用ノードの nodeID を取得します。
- `tsm topology list-nodes -v`。
- 上記のコマンドを実行して取得した nodeID を使用し、専用ノードのノードの役割を設定します。
- `tsm topology set-node-role -n <nodeID> -r flows`。
- 変更を適用し、サーバーを再起動します。
- `tsm pending-changes apply`。
- ステータスを確認し、すべてのプロセスが起動して実行中であり、正しく構成されていることを確認します。
- `tsm status -v`。

6. この段階では、他のノードで Tableau Prep Conductor を有効にしている場合があります。既定では、ノード上のバックグラウンダープロセスは、フロータスクを含め、すべてのタイプのタスクすべてを実行します。Tableau Prep Conductor およびフロータスクを特定のノードのみに分離するには、バックグラウンダーで以下のいずれかを実行するように構成します。

   - フロータスクのみを実行する場合: `tsm topology set-node-role -n <nodeID> -r flows`。
   - フロー以外の他のすべてのタスクを実行する場合: `tsm topology set-node-role -n <nodeID> -r no-flows`。

Tableau Prep Conductor の Tableau Server インストールへの追加に成功しました。

コアベース ライセンスを使用した Tableau Server インストール

本番環境の Tableau Server インストールのトポロジとして、フロー実行のための専用ノードを使用することが推奨されます。詳細については、「Tableau Server の最小ハードウェア要件と推奨事項」を参照してください。

コアベース ライセンスのデータ管理には、Tableau Server の Tableau Prep Conductor を有効にするプロダクトキーと、4 つのユニットに入る Tableau Prep Conductor コアが含まれます。この Tableau Prep Conductor コアは、フローの実行専用のノードに適用する必要があります。これらのプロダクトキーは、他のサーバーキーと同様、カスタマー ポータルより入手可能です。

Tableau Prep Conductor のライセンス発行の詳細については、「Tableau Server で使用する Tableau Prep Conductor のライセンス発行」を参照してください。

Tableau Server の単一ノードインストール

現在、単一ノードの Tableau Server インストールを使用している場合、2 番目のノードを追加し、それをフローの実行専用にすることをお勧めします。

1. 以下のトピックの情報を使用し、現在の Tableau Server インストールでアップグレードを実行します。

    - Windows
    - Linux

2. プロダクトキーをライセンス認証します。これにより、バックグラウンダープロセスが既に有効になっているノードで Tableau Prep Conductor が有効になります。コアベースのライセンス発行を使用している場合は、データ管理プロダクトキーとリソースコアプロダクトキーの両方を Tableau の展開に適用する必要があります。最初のキーを使用すると、Tableau Server でフローを実行することができますが、Tableau Prep Conductor と2 番目のキーを使用する場合は、Tableau Prep Conductor のノードにコアが追加されます。すべてのプロダクトキーは、カスタマー ポータルより入手可能です。

    - Tableau サービス マネージャーの Web インターフェイスの **[構成]** タブで **[ライセンス発行]** をクリックし、**[ライセンス認証]** をクリックします。
    - 新しいプロダクトキーを入力するか貼り付けし、**[ライセンス認証]** をクリックします。
    - **[登録]** ページで情報をフィールドに入力し、**[登録]** をクリックします。

3. サーバーの再起動するようにメッセージが表示されます。サーバーを再起動し、Tableau Prep Conductor が有効になり、実行中であることを確認します。

    - Tableau サービス マネージャーの Web インターフェイスで、**[ステータス]** タブをクリックし、ステータスを表示します。Tableau Prep Conductor が有効になり、実行中であると、プロセスのリストに Tableau Prep Conductor が **[アクティブ]** として表示されます。Tableau Prep Conductor が有効になっていないと、プロセスのリストに Tableau Prep Conductor は表示されるものの、ステータス情報はありません。

Tableau Prep Conductor が有効になっていない:

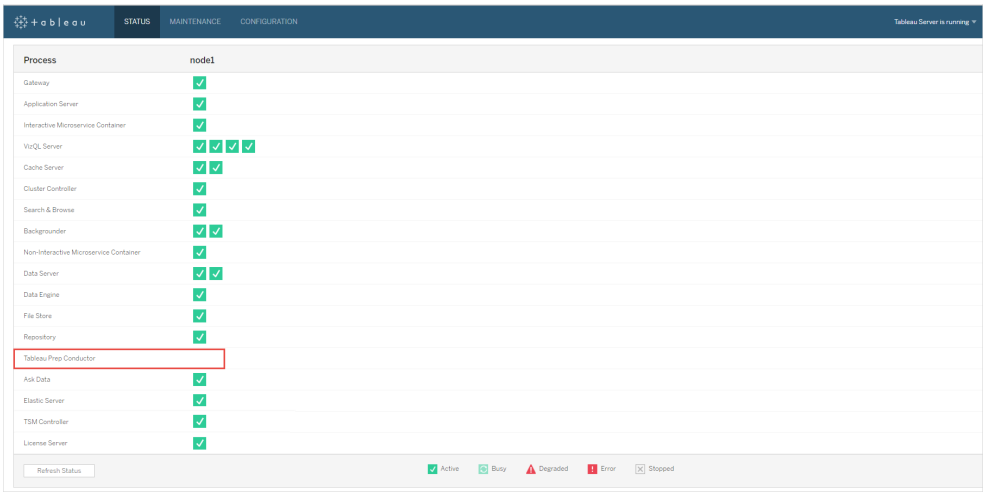
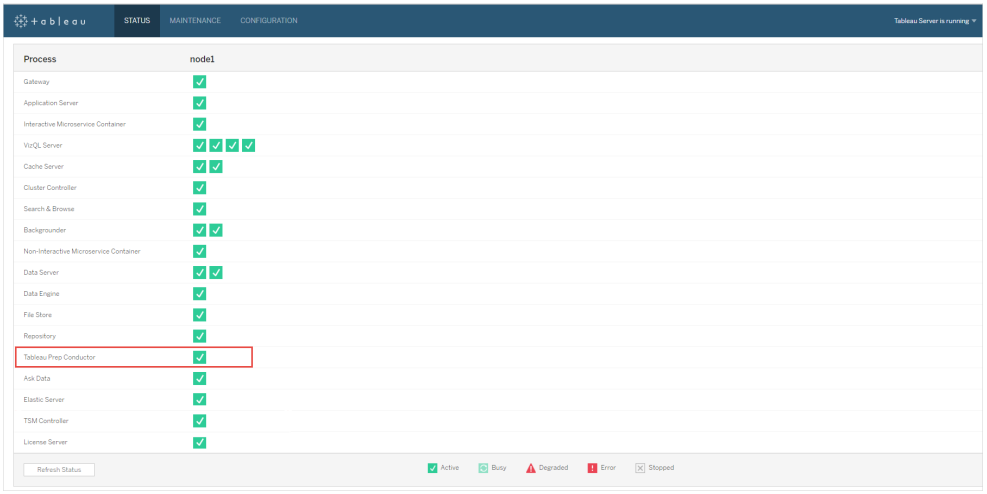


Tableau Prep Conductor が有効で実行中である:



4. 2 番目のノードを Tableau Server インストールに追加します。インストーラーにより、クラスタコントローラーなど、特定の必須プロセスが有効にされます。バックグラウンダープロセスはスケジュールされたフロータスクの実行に必要であるため、これを有効にします。バックグラウンダープロセスを有効にすると、インストーラーは自動的に、そのノード上のデータエンジンおよび Tableau Prep Conductor の単一インスタンスを有効にします。このノード上に他のプロセスを追加しないでください。

重要: このマシンの物理コアの数は、購入した Tableau Prep Conductor コアの単位数以下にする必要があります。たとえば、Tableau Prep Conductor コアを 4 つ購入した場合、ノードには最大 4 つまで物理コアを持たせることができます。Tableau Prep Conductor のライセンス発行のしくみについて理解するには、「Tableau Server で使用する Tableau Prep Conductor のライセンス発行」を参照してください。

5. このノードをフロータスクの実行専用にするには、次のコマンドを実行します。ノードの役割の詳細については、「Tableau Server でのノードの役割」を参照してください。

    - 各ノードのサービスのリストを表示するには、専用ノードの nodeID を取得します。

      `tsm topology list-nodes -v`。

    - 上記のコマンドを実行して取得した nodeID を使用し、専用ノードのノードの役割を設定します。

      `tsm topology set-node-role -n <nodeID> -r flows`。

    - 変更を適用し、サーバーを再起動します: `tsm pending-changes apply`。

    - ステータスを確認し、すべてのプロセスが起動して実行中であり、正しく構成されていることを確認します。

      `tsm status -v`。

Tableau Prep Conductor の Tableau Server インストールへの追加に成功しました。

Tableau Server の分散インストール

1. 以下のトピックの情報を使用し、現在の Tableau Server インストールでアップグレードを実行します。

    - Windows
    - Linux

2. プロダクトキーをライセンス認証します。これにより、バックグラウンダープロセスが既に有効になっているノードで Tableau Prep Conductor が有効になります。コアベースのライセンス発行を使用している場合は、データ管理 プロダクトキーとリソース コアプロダクトキーの両方を Tableau の展開に適用する必要があります。最初のキーを使用すると、Tableau Server でフローを実行することができますが、Tableau Prep Conductor と2 番目のキーを使用する場合は、Tableau Prep Conductor のノードにコアが追加されます。すべてのプロダクトキーは、カスタマー ポータルより入手可能です。

    - Tableau サービス マネージャーの Web インターフェイスの **[構成]** タブで **[ライセンス発行]** をクリックし、**[ライセンス認証]** をクリックします。
    - 新しいプロダクトキーを入力するか貼り付けし、**[ライセンス認証]** をクリックします。
    - **[登録]** ページで情報をフィールドに入力し、**[登録]** をクリックします。

3. サーバーの再起動するようにメッセージが表示されます。サーバーを再起動し、Tableau Prep Conductor が有効になり、実行中であることを確認します。

    - Tableau サービス マネージャーの Web インターフェイスで、**[ステータス]** タブをクリックし、ステータスを表示します。Tableau Prep Conductor が有効になり、実行中であると、プロセスのリストに Tableau Prep Conductor が **[アクティブ]** として表示されます。Tableau Prep Conductor が有効になっていないと、プロセスのリストに Tableau Prep Conductor は表示されるものの、ステータス情報はありません。

      Tableau Prep Conductor が有効になっていない:

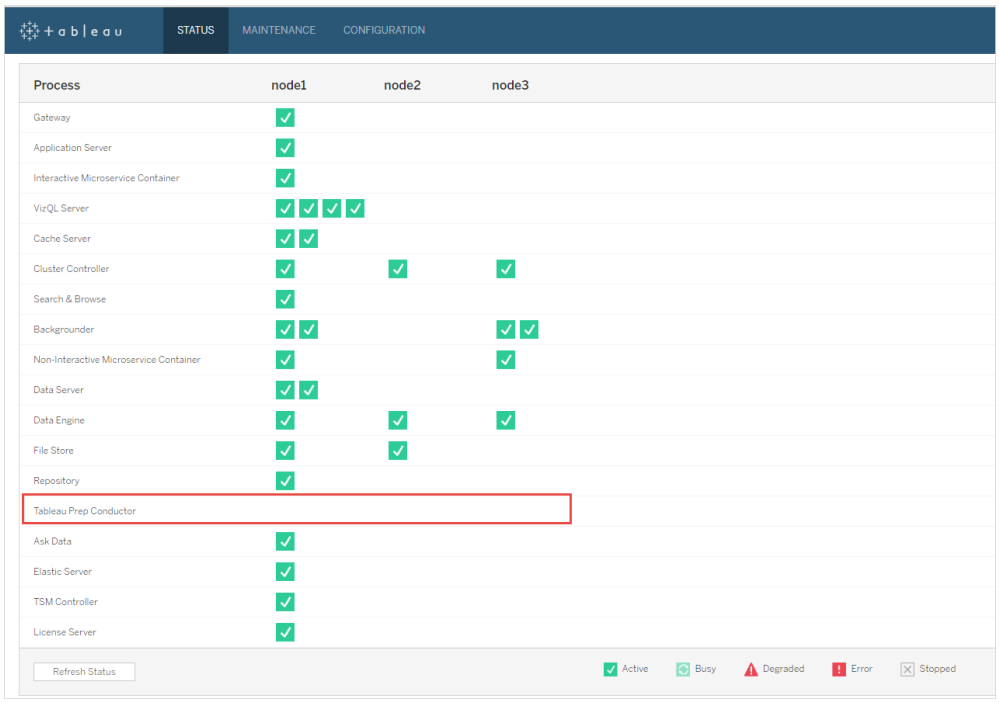
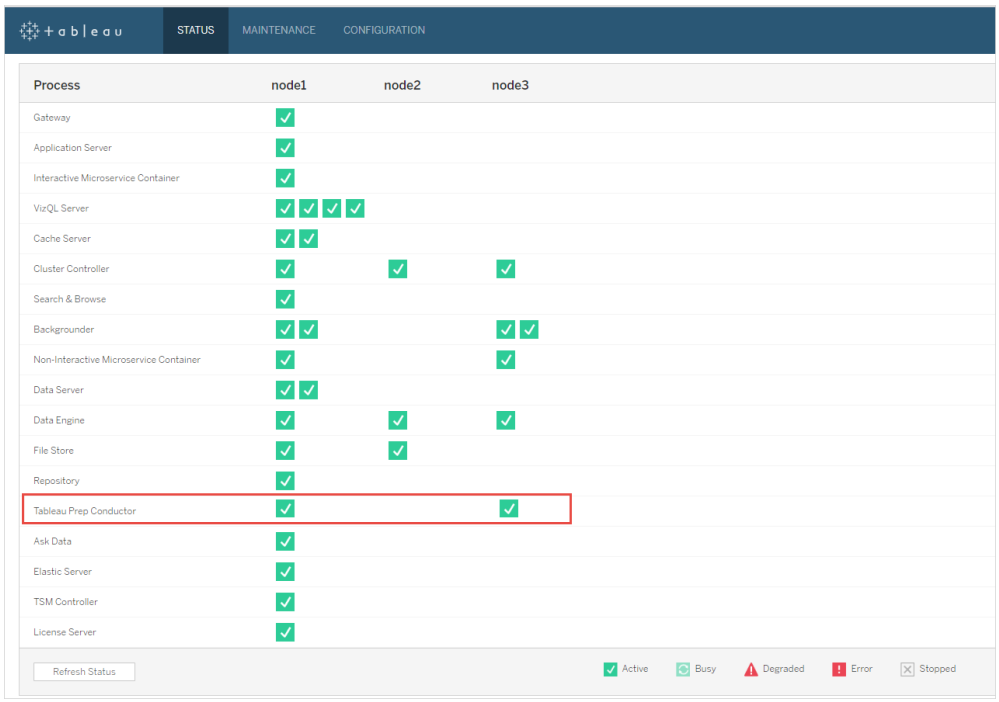


Tableau Prep Conductor が有効で実行中である:



4. 新しいノードを Tableau Server インストールに追加します。本番環境の Tableau Server インストールでは、フローに関係した操作を実行する専用ノードを使用することをお勧めします。インストーラーにより、クラスタコントローラーなど、特定の必須プロセスが有効にされます。バックグラウンダープロセスはスケジュールされたフロータスクの実行に必要であるため、これを有効にします。バックグラウンダープロセスを有効にすると、インストーラーは自動的に、そのノード上のデータエンジンの単一インスタンスを有効にします。このノード上に他のプロセスを追加しないでください。

   注: 専用ノードの数は、調整サービスアンサンブルの合計数に反映されます。新しい専用ノードを含むクラスタ内のノードの合計数に応じ、新しいノードで調整サービスの展開が必要になる場合があります。詳細については、「調整サービスアンサンブルの展開」を参照してください。

   重要:
   このマシンの物理コアの数は、購入した Tableau Prep Conductor コアの単位数以下にする必要があります。たとえば、Tableau Prep Conductor コアを 4 つ購入した場合、ノードには最大 4 つまで物理コアを持たせることができます。Tableau Prep Conductor のライセンス発行のしくみについて理解するには、「Tableau Server で使用する Tableau Prep Conductor のライセンス発行」を参照してください。

5. このノードをフロータスクの実行専用にするには、次のコマンドを実行します。これにより、新しいノードで Tableau Prep Conductor が有効になります。詳細については、「Tableau Server でのノードの役割」を参照してください。

   - 各ノードのサービスのリストを表示するには、専用ノードの nodeID を取得します。

     `tsm topology list-nodes -v`。

   - 上記のコマンドを実行して取得した nodeID を使用し、専用ノードのノードの役割を設定します。

```
tsm topology set-node-role -n nodeID -r flows。
```

- 変更を適用し、サーバーを再起動します。

  ```
  tsm pending-changes apply。
  ```

- ステータスを確認し、すべてのプロセスが起動して実行中であり、正しく構成されていることを確認します。

  ```
  tsm status -v。
  ```

6. この段階では、バックグラウンダープロセスがある他のノードで Tableau Prep Conductor を有効にしている場合があります。既定では、ノード上のバックグラウンダープロセスは、フロータスクを含め、すべてのタイプのタスクすべてを実行します。Tableau Prep Conductor およびフロー操作を特定のノードのみに分離するには、バックグラウンダーで以下のいずれかを実行するように構成します。

   - フロー タスクのみを実行する場合:

     ```
     tsm topology set-node-role -n <nodeID> -r flows。
     ```

   - フロー以外の他のすべてのタスクを実行する場合:

     ```
     tsm topology set-node-role -n <nodeID> -r no-flows。
     ```

次の手順

手順 2 - Tableau Server のフロー設定の構成。

実行可能なユーザー

Tableau Server 管理者は、Tableau Server のインストールやアップグレードのほか、Tableau Server での Tableau Prep Conductor の有効化を行うことができます。

手順 2 - Tableau Server のフロー設定の構成

このトピックでは、Tableau Server で構成可能な様々なフロー設定について説明します。フローの Web 作成を有効にするために必要なさまざまな設定についての詳細は、「Web 上でのフローの作

成と操作」を参照してください。

パブリッシュ、スケジュール、および認証資格情報の設定

Data Management プロダクトキーを使用して Tableau Prep Conductor をライセンス認証すると、Tableau Server インストール全体で Tableau Prep Conductor が有効になります。サイトの設定は、さらに変更してカスタマイズすることができます。

すべてのサイトのまたは個々のサイトのフローに関連する設定を構成するには、以下の指示に従います。

Tableau Server 管理者ページにサインインするには、以下の指示に従います。

- Windows: Tableau Server 管理者ページ。
- Linux: Tableau Server 管理者ページ。

サイトでフローのパブリッシュとスケジュールを許可するかどうかを構成する:

1. **ユーザーによるフローのパブリッシュとスケジュールの有効化:** この設定は、Tableau Prep Conductor を有効にすると、既定で有効になります。複数のサイトがある場合、サーバーの Tableau Prep を個々のサイトに関して選択的にオフにすることができます。一度フローを許可したサイトでこの設定を無効にする場合は、Tableau Prep Conductor の無効化が意味することを参照してください。

   [設定] の **[全般]** ページで、**[Tableau Prep Conductor]** セクションまでスクロールし、**[Allow users to schedule and monitor flows (ユーザーにフローのスケジュールと監視を許可)]** チェックボックスをオフにします。

2. **リンクされたタスクを使用して、ユーザーがフロー実行をリンクできるようにする (バージョン 2021.3 以降):** フロータスクを次々に実行するようにスケジュールできます。バージョン 2022.1 以降では、このオプションは既定で有効になっています。以前のバージョンでは、管理者が最初にこの機能を有効にする必要がありました。

   複数のサイトがある場合、個々のサイトで**リンクされたタスク**を選択的にオフにすることができますが、このオプションは最初に**サーバー設定**レベルで有効にする必要があります。

リンクされたタスクをスケジュールした後にサーバー設定をオフにした場合、実行中のタスクはすべて完了しますが、スケジュール済みのリンクされたタスクは非表示になり、**[スケジュールされたタスク]** タブに表示されなくなります。

3. **認証資格情報の埋め込み**

- パブリッシャーにデータ ソース、フロー、またはワークブックへの認証資格情報の埋め込みを許可: この設定により、パブリッシャーはパブリッシュ済みのフローにパスワードを添付し、Web ユーザーを自動的に認証することができます。

- パブリッシャーにフローの実行とデータ抽出の更新のスケジュールを許可: このオプションは、上記の設定が有効になっている場合にのみ利用可能です。この設定を有効にすると、パブリッシャーには [パブリッシュ] ダイアログ ボックスでスケジュール オプションが表示されます。

Tableau Prep Conductor の無効化が意味すること

Tableau Prep Conductor をしばらく使用した後に無効にすると、フロー、スケジュール、タスク、およびフローに関連する他の内容を表示できなくなります。次の表では、サーバーで Tableau Prep を完全に無効にするか、特定サイトのみ無効にする場合に表示できる情報と表示できない情報を記載しています。

	サーバー レベルで Prep が有効になっていない	サーバー レベルで Prep が有効になっているが、サイトでは無効になっている	サーバーとサイトの両方で Prep が有効になっている
フローの表示	あり	あり	あり
サーバー ビューでのタスク/スケジュールの表示	なし	あり	あり
サイト ビューでのタスク/スケジュールの表示	なし	なし	あり
サイト設定の表示 (サーバー管	あり(無効)	あり	あり

理者のみ)			
TSM ステータスの表示	あり(Tableau Prep Conductor が表示されない)	あり	あり
TSM 設定の表示	あり(無効)	あり	あり

重要: スケジュールされたタスクは、そのサイトで Tableau Prep Conductor が無効になっている場合でも引き続き実行されますが、エラーが発生します。

フローの失敗に関する通知の構成

Tableau Server で、フローの実行失敗に関するメール通知の送信を構成することができます。フローの実行時に発生したエラーに関する通知は、スケジュールされたタスクやリンクされたタスクにより、または **[今すぐ実行]** メニュー オプションを使用した手動実行により送信されます。最初にサーバー全体の設定を有効にしてから、サイトレベルで構成する必要があります。

サーバー全体のメール通知を有効にするには

以下に説明するように、Tableau サービス マネージャー (TSM) の Web インターフェイスまたは TSM CLI を使用できます。

TSM Web インターフェイスの使用

1. ブラウザーで TSM を開きます。

   https://<tsm-computer-name>:8850

2. **[構成]** タブで **[通知]** をクリックし、**[メール サーバー]** をクリックします。

3. メール サーバー情報を入力します。

4. **[イベント]** タブをクリックします。

5. デフォルトでオンになっていない場合は、**[コンテンツの更新]** で **[フロー実行、暗号化ジョブ、スケジュールされた更新の失敗時にメールを送信]** を選択します。
6. 構成情報を入力したら、**[保留中の変更を保存]** をクリックします。
7. **[変更を適用して再起動]** をクリックします。

TSM CLI の使用

この通知値は、tsm configuration set コマンドを使用して個別に設定可能です。

Windows: tsm configuration。

Linux: tsm configuration。

通知値の設定

次の構文で tsm configuration set コマンドを使用し、

フロー失敗の通知を有効にし、以下のコマンドを実行します。

```
tsm configuration set -k backgrounder.notifications_enabled -v true
```

> **注:** これにより、抽出更新の失敗とフローの失敗の両方でメール通知が有効になります。

値を設定したら、以下のコマンドを実行する必要があります。

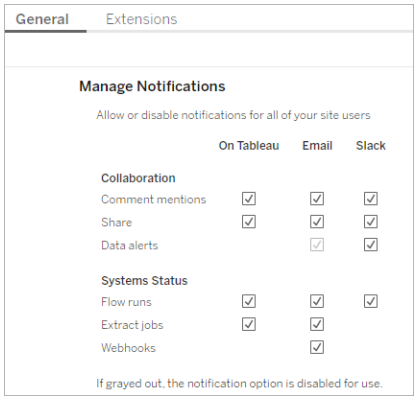
```
tsm pending-changes apply
```

pending-changes apply コマンドは、Tableau Server が実行中の場合にはそれが再起動することを知らせるプロンプトを表示します。このプロンプトはサーバーが停止しても表示されますが、その場合には再起動は行われません。--ignore-prompt オプションを使用してメッセージが表示されないようにできますが、そのようにしても再起動に関する動作が変わることはありません。

サイトの通知を構成するには、次を実行します。

[設定] の **[全般]** ページで、**[通知の管理]** 設定までスクロールし、サイトユーザーが受信する通知のタイプを選択します。

通知は、メールまたは Tableau サイトで受信するか、または管理者がサイトを Slack に接続している場合は Slack ワークスペースで受信することができます。詳細については、「サイト設定 リファレンス」を参照してください。



注: 2020.4 以前から2021.1 以降にアップグレードする場合は、通知設定をもう一度作成してください。古い通知設定は、[通知の管理] 設定に自動的に移動されません。

次の手順

手順 3 - フロータスクのスケジュールの作成

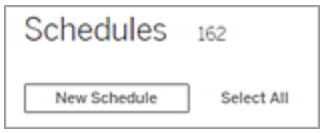
実行可能なユーザー

Tableau Server 管理者は、サーバーレベルの設定とサイトレベルの設定を構成できます。
Tableau サイト管理者は、サイトレベルの設定を構成できます。

手順 3 - フロー タスクのスケジュールの作成

新しいスケジュールの作成:

1. **[スケジュール]** タブで、**[新しいスケジュール]** をクリックします。

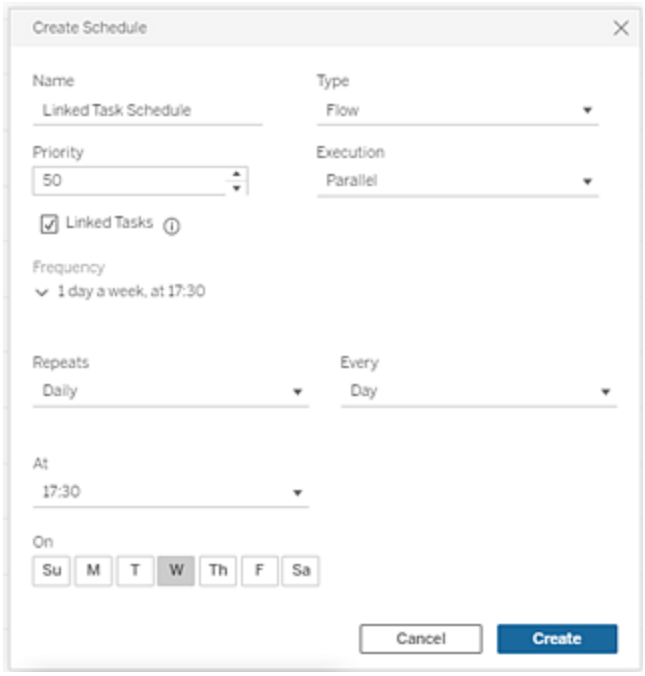


2. **[新しいスケジュール]** ダイアログ ボックスに次の情報を入力し、**[作成]** をクリックします。

   - **[名前]:** スケジュールのわかりやすい名前を入力します。スケジュール頻度について説明するのが一般的です。
   - **タイプ:** タスクの種類としてフローを選択します。
   - **優先度:** デフォルトの優先度を 1 ~ 100 で定義できます。1 が最も高い優先度を表します。既定でこの値がタスクに割り当てられます。キュー内で 2 つのタスクが保留中の場合、優先度が高いタスクが最初に実行されます。
   - **[実行]:** スケジュールを並列的に実行するか、または順に実行するかを選択します。並列的なスケジュールを選択すると、利用可能なすべてのバックグラウンダープロセスで実行されるため、完了がより迅速になります。
   - **リンクされたタスク (バージョン 2021.3 以降):** スケジュールを使用してフローを次々に実行するように設定する場合は、チェックボックスをオンにします。リンクされたタスクは、**並列**処理方式で実行できる必要があります。

     バージョン 2022.1 以降では、このオプションは既定で有効になっています。以前のバージョンでは、リンクされたタスクに対応するようにスケジュールを設定するには、サーバー管理者が最初に、リンクされたタスクをサーバーで有効にする必要があります。詳細については、手順 2 - Tableau Server のフロー設定の構成を参照してください。
   - **[頻度]:** スケジュールを時間単位、日単位、週単位、または月単位のいずれかに定

義できます。



次の手順

手順 4 - 入力場所と出力場所を許可リストに登録する

実行可能なユーザー

Tableau Server 管理者は、スケジュールを作成して変更できます。スケジュールはサーバー レベルで作成され、サーバー上のすべてのサイトに適用されます。

手順 4 - 入力場所と出力場所を許可リストに登録する

このトピックでは、この機能に適用されるルールと、ネットワーク上のディレクトリを許可リストに登録する方法について説明します。

フローの入力接続と出力接続では、ネットワーク上のディレクトリにあるデータベースやファイルへの接続が必要な場合があります。アクセスを許可するディレクトリを許可リストに登録する必要があります。入力接続と出力接続は、許可リストに登録されている場所にあるデータへの接続のみ許可されます。既定では、接続は許可されません。

> **注**: フローとフローファイル (tflx) に埋め込まれているデータを Tableau Server に引き続きパブリッシュすることはできますが、ディレクトリが組織の許可リストに含まれていない場合、フローの実行でエラーが発生します。

入力場所と出力場所を許可リストに登録する方法

この設定を構成するときは、次のルールを適用し、考慮に入れる必要があります。

- ディレクトリパスに Tableau Server からアクセスできる必要があります。これらのパスはサーバー起動時およびフロー実行時に検証されますが、フローを Tableau Server にパブリッシュする時点では検証**されません**。
- ネットワークディレクトリパスは絶対パスである必要があり、ワイルドカードやパスを横断する他の記号を含めることはできません。たとえば、`\\myhost\myShare\*` **または** `\\myhost\myShare*` は無効なパスであるため、すべてのパスが許可されません。*myShare* の下にあるすべてのフォルダーを許可リストに登録する正しい方法は、`\\myhost\myShare` **または** `\\myhost\\myShare\` です。

  > **注:** `\\myhost\myShare` 構成では `\\myhost\myShare1` が許可されません。これらのフォルダーを両方とも許可リストに登録するには、これらを \\myhost\myShare; \\myhost\myShare1 としてセーフリストに登録します。

- **Windows:**
  - 値は `*` (例: `tsm configuration set -k maestro.input.allowed_paths -v "*"`) にして任意のネットワークディレクトリを許可するか、または指定するネットワークディレクトリパスのリストをセミコロン **(;)** で区切ったものとします。ディレクトリパスのリストを指定する場合は、ファイル共有のルートではなく、特定のディレクトリを指定してください。

- パスにスペースまたは特殊文字が含まれる場合、単一引用符または二重引用符を使用する必要があります。単一引用符または二重引用符を使用するかどうかは、使用しているシェルによって異なります。

- 値が * に設定されている場合でも、ローカル ディレクトリパスは許可されません。

- フロー出力をネットワーク共有に保存するには、まず Tableau Server で実行ユーザーのサービス アカウントを構成する必要があります。既定のシステム アカウントを使用してネットワーク共有にフローを保存することはできません。次に、ネットワーク共有上のターゲット ディレクトリを構成して、作成した実行ユーザー アカウントがフル コントロール パーミッションでアクセスできるようにします。

  ネストされたフォルダーのパーミッションの管理方法は組織によって異なるため、場合によっては、実行ユーザー アカウントに対して、フォルダー階層での読み取り、書き込み、実行、削除、およびフォルダーの一覧表示のみを許可する追加のパーミッションを付与して、実行ユーザーがターゲット フォルダーにアクセスできるようにする必要があります。

- **Linux:**

  - 値は、ローカル ("native_api.internal_disallowed_paths" を使用して構成された一部のシステム パスを除く) を含む任意のパスを意味する * にする (例: `tsm configuration set -k maestro.input.allowed_paths -v "*"`) か、パスのリストをセミコロン (;) で区切ったものとします。

  - 使用するカーネル バージョンは 4.7 以降である必要があります。4.7 より前のカーネル バージョンでは、ネットワーク共有から許可リストへの登録はサポートされていません。以前のバージョンでは、出力がネットワーク共有に書き込まれると、Hyper 出力ファイルの生成が失敗するため、実行時にフローが失敗します。以前のバージョンのネットワーク共有から入力ファイルを読み取ると、フローの実行は失敗します。カーネル バージョンを確認するには、Linux のターミナルでコマンド `uname -r` を入力します。これにより、Linux マシンで実行しているカーネルのバージョンが完全に表示されます。Red Hat Enterprise Linux の場合、カーネル バージョン 4.7 以降は、Red Hat

Enterprise Linux バージョン 8 でのみ使用可能であることに注意してください。

- フロー出力をネットワーク共有に保存するには、Tableau Server リソースにアクセスできるローカルの Linux アカウントに、ネットワーク共有上のターゲットディレクトリへのフル コントロール パーミッションが付与されている必要があります。許可されたフローのリストとinternal_disallowed リストの両方にパスがある場合、internal_disallowed が優先されます。
  フローで使用される入力パスと出力パスの両方のマウントポイントは、 `native_api.unc_mountpoints` 構成キーを使用して構成する必要があります。例:
  `tsm configuration set -k native_api.unc_mountpoints -v 'mountpoints'`
  この構成の詳細については、Tableau ナレッジ ベースの記事「Linux 上の Tableau Server - Windows 共有ディレクトリへの接続方法」を参照してください。

許可されているネットワークディレクトリパスのリストを作成するには、次のコマンドを使用します。

入力接続の場合:

```
tsm configuration set -k maestro.input.allowed_paths -v your_
networkdirectory_path_1;your_networkdirectory_path_2

tsm pending-changes apply
```

出力接続の場合:

```
tsm configuration set -k maestro.output.allowed_paths -v your_
networkdirectory_path_1;your_networkdirectory_path_2

tsm pending-changes apply
```

重要:
これらのコマンドにより既存の情報が上書きされ、提供した新しい情報に取って代わります。既存のリストに新しい場所を追加する場合は、すべての場所、既存の場所、追加する新しい場所のリストを提供する必要があります。入力場所および出力場所の現在のリストを表示するには、次のコマンドを使用します。

```
tsm configuration get -k maestro.input.allowed_paths
tsm configuration get -k maestro.output.allowed_paths
```

次の手順

手順 5 - オプションのサーバー構成

実行可能なユーザー

Windows では、ローカル コンピューターの管理グループのメンバーが *tsm* コマンドを実行できます。

Linux では、**tsmadmin** グループのメンバーが *tsm* コマンドを実行できます。**Tsmadmin** グループは、*tsm.authorized.groups* 設定を使用して構成できます。

手順 5 - オプションのサーバー構成

このトピックで説明するオプションは、Tableau Server でのフローのパブリッシュやフローのスケジュールを有効化するために必須というわけではありません。これらを使用すると、要件に応じた環境のカスタマイズができます。

フローのタイムアウト期間を設定する

フローを実行できる時間の上限を設定すると、後続タスクの実行が、停止しているタスクのために延期されないようにできます。次の 2 つの *tsm* コマンドのオプションにより、フローのバックグラウンド タスクがキャンセルされるまでの、フロー タスクの実行可能時間が決まります。これらの 2 つのコマンドの組み合わせで、フロー タスクの合計タイムアウト値が決まります。

`backgrounder.default_timeout.run_flow` は、フロー実行タスクがキャンセルされるまでの秒数を設定します。

例:

```
tsm configuration set -k backgrounder.default_timeout.run_flow -v
<new value>

tsm pending-changes apply
```

(既定値: 14400 秒または 4 時間)

`backgrounder.extra_timeout_in_seconds` コマンドは、`backgrounder.querylimit` の設定を超えてバックグラウンド ジョブがキャンセルされるまでの

秒数を設定します。この設定により、停止したジョブが後続のジョブを保持しないようにします。設定は、backgrounder.timeout_tasks にリストされているプロセスに適用されます。

例:

```
tsm configuration set -k backgrounder.extra_timeout_in_seconds -v
<value>
```

(既定値: 1800 秒または 30 分)

フローを実行しているサーバーで使用可能なリソースを確認してください。Tableau Prep Conductor の専用ノードを設けることをお勧めします。

フロー タスクを中断するしきい値を設定する

既定では、フロー タスクが 5 回連続して失敗すると、中断されます。フロー タスクが中断される前に、失敗が可能な回数のしきい値を変更するには、次の tsm configuration set コマンドを使用します。

```
tsm configuration set -k backgrounder.flow_failure_threshold_for_
run_prevention -v <number>
```

これにより、タスクが中断される前に必要となる、フロー タスクの連続失敗回数のしきい値が設定されます。これはサーバー全体の設定です。

実行可能なユーザー

Tableau Server 管理者は、サーバー構成を変更することができます。

フロー タスクのスケジュール

注: フローを Tableau Cloud で実行するか、Tableau Prep Conductor を使用して Tableau Server で実行するようにスケジュールを設定することができます。Prep Conductor は、導入ごとに データ管理 を介してライセンスが設定されます。データ管理を購入してライセンスを設定したら、Prep Conductor を有効にする必要があります。
バージョン 2020.4.1 以降、フローを Web にパブリッシュするための データ管理 ライセンスは不要になりました。作成者は、フローの作成と編集をサーバー上で直接行うこともできます。
Web 上でのフローの作成の詳細については、Web 上の Tableau Prepを参照してください。

注: 2024.1 リリースでは、Tableau Cloud のフロー スケジュールに変更が加えられました。左側のナビゲーション ペインの [スケジュール] タブが削除され、フロー タスクをスケジュールできるパーミッションのあるユーザーであれば、フローの [スケジュールされたタスク] タブから直接 カスタム スケジュールを作成できるようになりました。

フローを特定の時間に実行したり定期的に実行したりするために、スケジュール タスクを作成できます。スケジュール タスクは、事前に設定されたスケジュールに従います。スケジュールはシステム管理者によって作成されます。Tableau Server でスケジュールを作成する方法については、Tableau Server ヘルプの「ステップ 3: フロー タスクのスケジュールの作成」を参照してください。

バージョン 2021.3 以降では、フローを次々に実行するようにタスクをスケジュールするときに、フローの実行をリンクできます。詳細については、このトピックのリンクされたタスクのスケジュールを参照してください。

バージョン 2022.1 以降では、新しいフロー メソッドである**「リンクされたタスクのクエリ」**、**「リンクされた単一タスクのクエリ」**、および**「リンクされたタスクを今すぐ実行」**を使用して、リンクされたタスクを REST API を介して実行することもできます。詳細については、Tableau REST API ヘルプの「フロー メソッド」を参照してください。

パラメーターを含む実行中のフロー

バージョン 2021.4 以降では、フローにユーザーパラメーターを含めて、フローをより動的にすることができます。フローが実行されると、パラメーター値を入力するように求められます。スケジュールに沿ってフローを設定する場合は、その時点でパラメーター値を指定します。

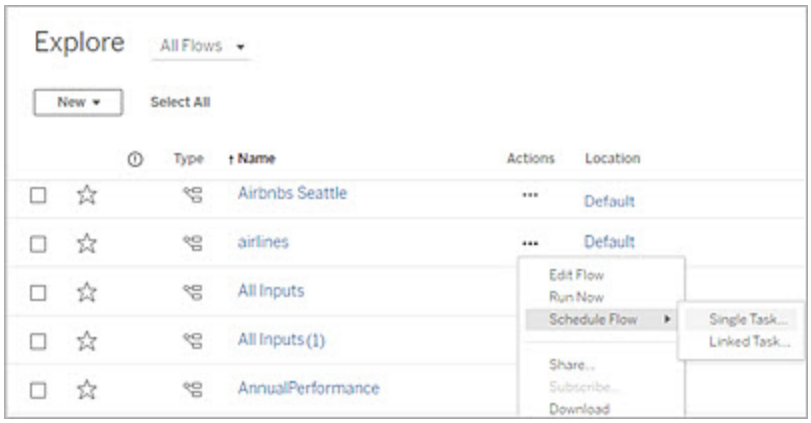
必須のパラメーターの場合は、パラメーター値を指定する必要があります。オプションのパラメーターの場合は、その値を入力することも、パラメーターの現在の (デフォルト) 値を受け入れることもできます。詳細については、Tableau Prep ヘルプの「スケジュールに従ってフローを実行する」を参照してください。

Tableau Prep Builder および Tableau Cloud バージョン 2023.2 以降では、ファイルおよびパブリッシュされたデータソース出力タイプのフロー出力名に日付または時刻のシステム パラメーターを適用できます。フローの出力名に開始時間が自動的に追加されます。

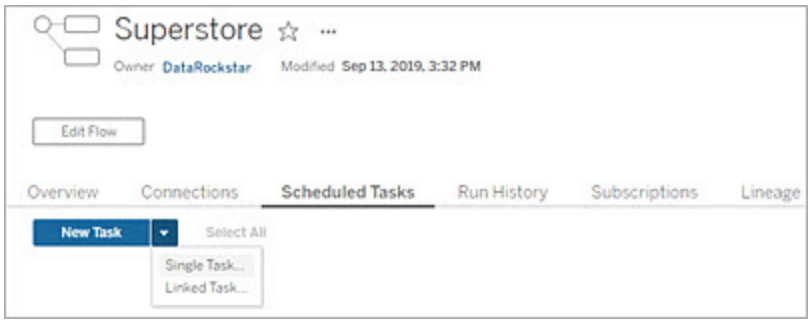
> **注:** 管理者は、Tableau Server または Tableau Cloud でパラメーターを含むフローを実行する前に、サーバーとサイトの**フロー パラメーター**設定をサーバーで有効にする必要があります。詳細については、Tableau Server または Tableau Cloud ヘルプの「**Web 上でのフローの作成と操作**」を参照してください。

フロー タスクのスケジュールを設定する

1. 次のいずれかを実行します。
    - (バージョン 2022.1 以降) **[探索]** ページの **[リスト]** ビューの **[アクション]** メニューで、**[フローのスケジュール設定] > [単一タスク]** を選択します。リストでフローを選択すると、上部の **[アクション]** メニューを使用することもできます。

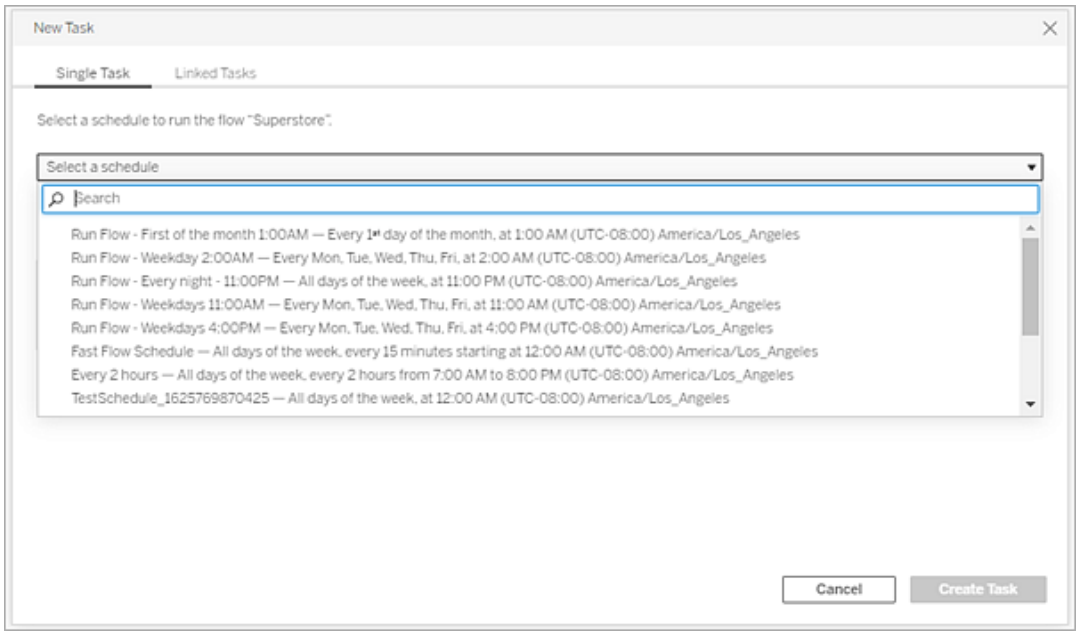


- フローの **[概要]** ページの **[スケジュールされたタスク]** タブで、**[新しいタスク]** をクリックするか、ドロップダウンをクリックして **[単一のタスク]** を選択します。



出力ステップがタスクに割り当てられていない場合は、**[概要]** ページから新しいタスクを作成することもできます。このページの **[スケジュール]** フィールドで、**[タスクを新規作成]** をクリックします。

2. **[新しいタスク]** ダイアログの **[単一タスク]** (以前のリリースでは **[新しいタスク]**) タブで、ドロップダウン リストからスケジュールを選択します。



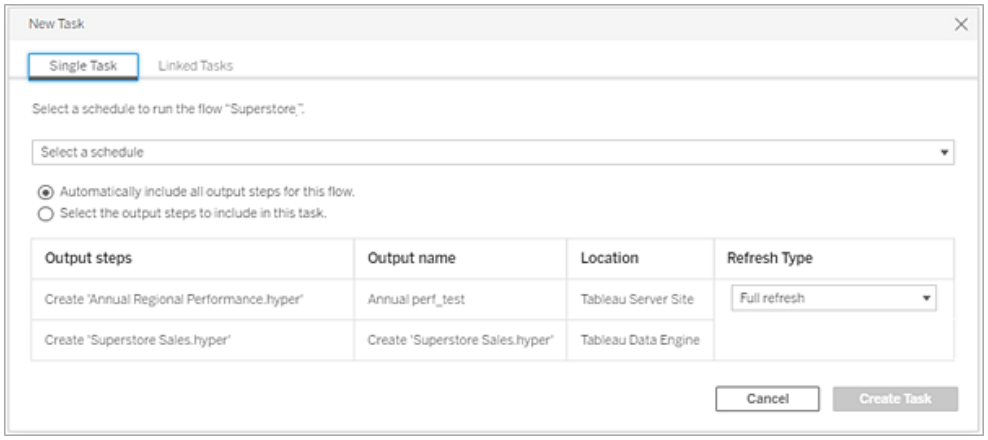
3. 以下のいずれかのオプションを選択します。

   - **このフローのすべての出力ステップを自動的に含める:** (既定) このオプションを選択すると、このフローの現在および今後のすべての出力ステップをスケジュール タスクに含めます。時間が経過してフローに新しい出力ステップが追加されると、これらは実行時にスケジュールに自動的に含まれます。

   - **このタスクに含める出力ステップを選択**: このオプションを選択して、スケジュールされたタスクに含める出力ステップを手動で選択します。

     すべての出力ステップをフロー タスクに含めるには、**[出力ステップ]** の横にあるチェックボックスをオンにします。**[このフローのすべての出力ステップを自動的に含める]** ラジオボタンを選択した場合は、この領域は編集できません。このセクションを有効にするに

は、他のラジオ ボタンを選択してください。



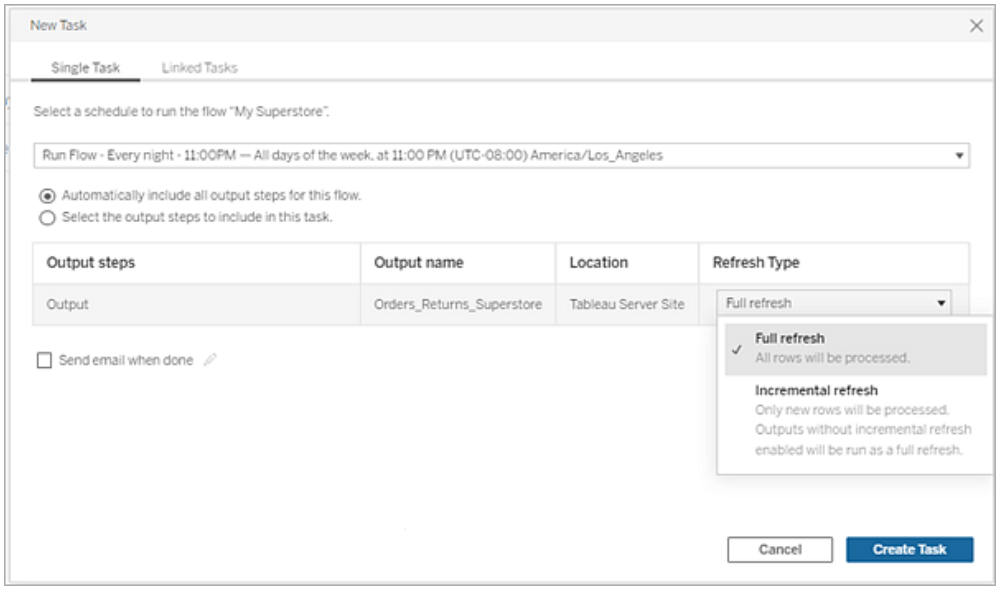
4. (バージョン 2020.2.1 以降) **[更新タイプ]** を選択します。これらの設定についての詳細は、「増分更新を使用したフローデータの更新」を参照してください。

> **注**: バージョン 2020.2.1 以降で、増分更新を使用するように構成した 1 つの入力を複数の出力に関連付けた場合は、同じ更新タイプを使用してそれらの出力を同時に実行する必要があります。それ以外の場合は、フローを実行できません。

- **完全更新** (既定): すべてのデータを更新し、フローの出力設定に基づいてテーブルにデータを作成または追加します。
- **増分更新**: 新しい行のみを更新し、フローの出力設定に基づいてテーブルにデータを作成または追加します。増分更新は、増分更新を使用するようにフローを設定している場合にのみ選択できます。

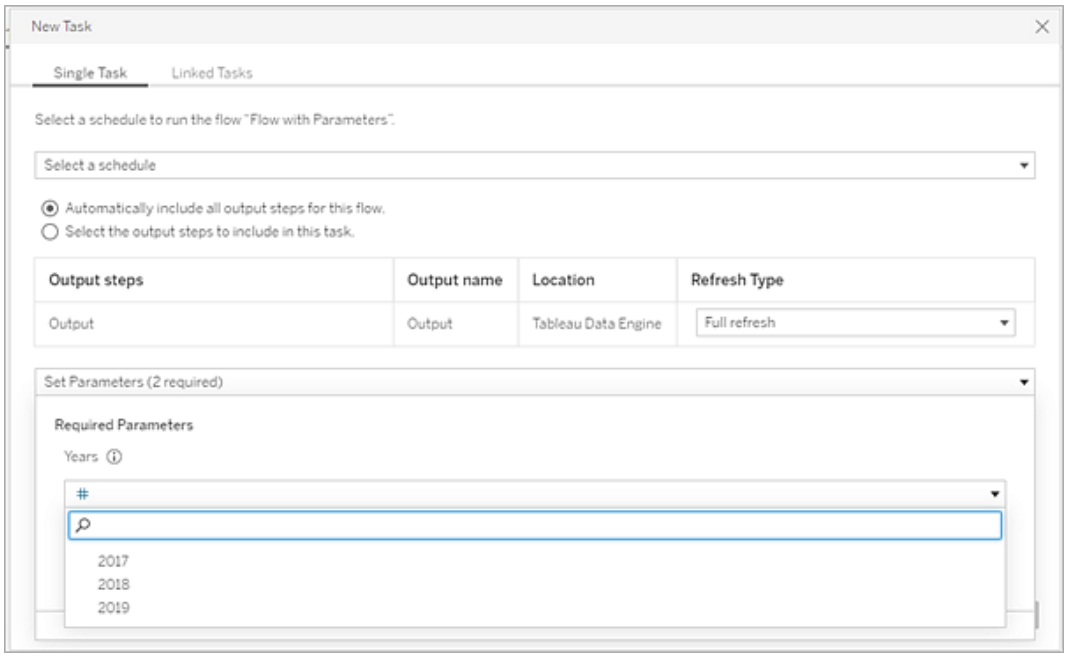
> **注**: Tableau Prep Conductor では、既存の出力が見つからない場合、選択した実行オプションに関係なく、すべての出力に対して完全更新が実行されます。後続のフローを実行するときは、増分更新の構成データが欠落していない

か、既存の出力を削除していない限り、増分更新プロセスを使用して、新しい行のみを取得して処理します。



5. (オプション) フローの所有者である場合は、**[完了時にメールを送信]** を選択して、フローが成功したときにユーザーに通知します。フロー実行でメール通知を送信する方法の詳細については、フロー実行の成功をユーザーに通知するを参照してください。

6. (バージョン 2021.4 以降) フローにパラメーターが含まれている場合は、必須またはオプションのパラメーター値を入力します。フローを実行するために必要な値を入力する必要があります。



7. **[タスクの作成]** をクリックして、スケジュール タスクを作成します。

リンクされたタスクのスケジュール

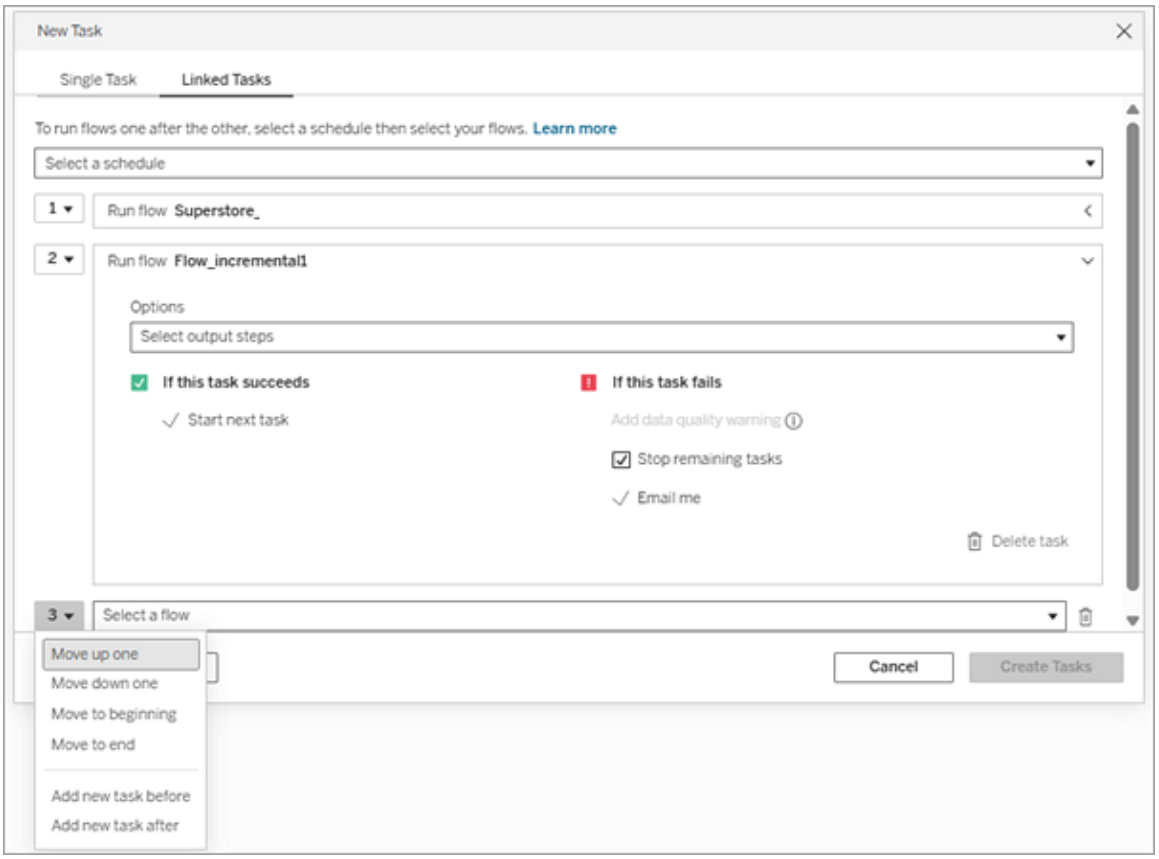
Tableau Server および Tableau Cloud バージョン 2021.3.0 以降でサポートしています。

注: バージョン 2022.1 以降、リンクされたタスクの機能はデフォルトで有効になっています。サーバーとサイトの管理者は、**[設定]** ページおよび **[スケジュール]** ダイアログのフロー スケジュールでこの機能をオフにできます。以前のバージョンでは、この機能を使用して管理するには、サーバー管理者は最初にこの機能を有効にする必要がありました。詳細については、「手順 2 - Tableau Server のフロー設定の構成」および「手順 3 - フロー タスクのスケジュールの作成」を参照してください。

[リンクされたタスク] オプションを使用して、最大 20 件のフローを次々に実行するようにスケジュールできます。フローの一覧はスケジュールを選択することにより簡単に設定でき、次に後続のフローを選択することにより、選択した順序でフローを実行することができます。

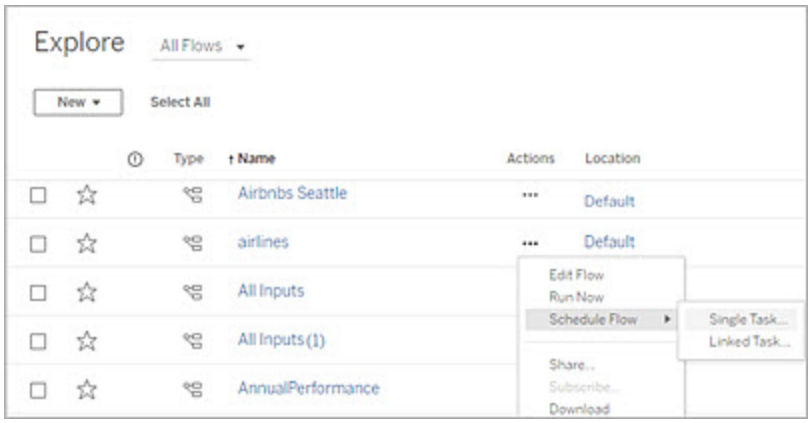
フローは、リストで指定された順序で実行されます。メニューを使用すると、リスト内のフローを移動したり、先行のフローや後続のフローをいつでもリストに新しく追加したりできます。

フロー実行に含める出力を選択し、前のフロー実行が失敗したときにスケジュール内の残りのフローをどう処理するかをTableauに指示する設定を行います。

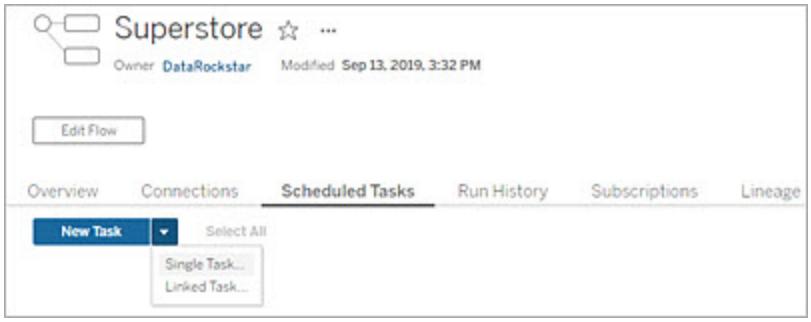


1. 次のいずれかを実行します。

   - (バージョン2022.1以降) **[探索]** ページの **[リスト]** ビューの **[アクション]** メニューで、**[フローのスケジュール設定] > [リンクされたタスク]** を選択します。リストで複数のフローを選択した場合は、上部の **[アクション]** メニューを使用することもできます。

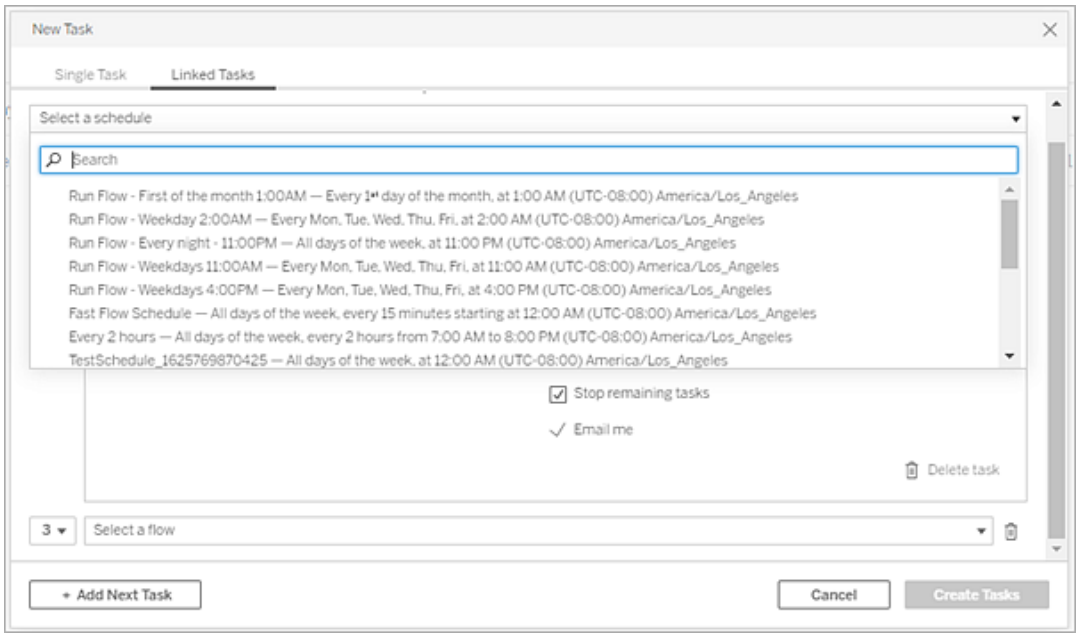


- フローの **[概要]** ページの **[スケジュールされたタスク]** タブで、**[新しいタスク]** をクリックして **[リンクされたタスク]** タブを選択するか、ドロップダウンをクリックして **[リンクされたタスク]** を選択します。



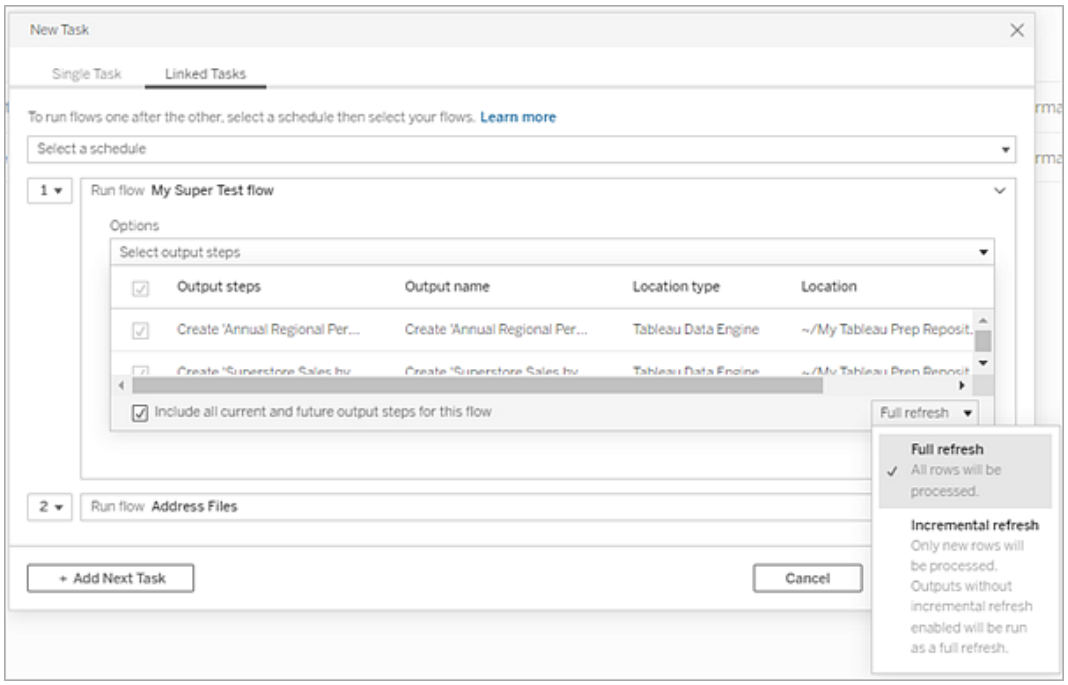
出力ステップがタスクに割り当てられていない場合は、**[概要]** ページから新しいタスクを作成することもできます。このページの **[スケジュール]** フィールドで、**[タスクを新規作成]** をクリックします。

2. **[新しいタスク]** ダイアログの **[リンクされたタスク]** タブで、ドロップダウン リストからスケジュールを選択します。リンクされたタスクに対して有効になっているスケジュールのみが表示されます。



3. **[出力ステップの選択]** ドロップダウンをクリックして、実行するフロー出力を選択します。デフォルトでは、すべてのフロー出力が含まれています。特定の出力を選択するには、**[このフローの現在および将来のすべての出力ステップを含める]** チェックボックスをオフにします。

   タスクを開始するフローは、実行する最初のフローとして自動的に設定されますが、フローの実行順序は、リストに他のフローを追加した後でもメニューを使用して変更できます。



4. 次のオプションから更新タイプを選択します。

> **注**: 増分更新を使用するようにある入力を設定し、その入力を複数の出力に関連付けた場合は、同じ更新タイプを使用してそれらの出力を同時に実行する必要があります。それ以外の場合は、フローを実行できません。

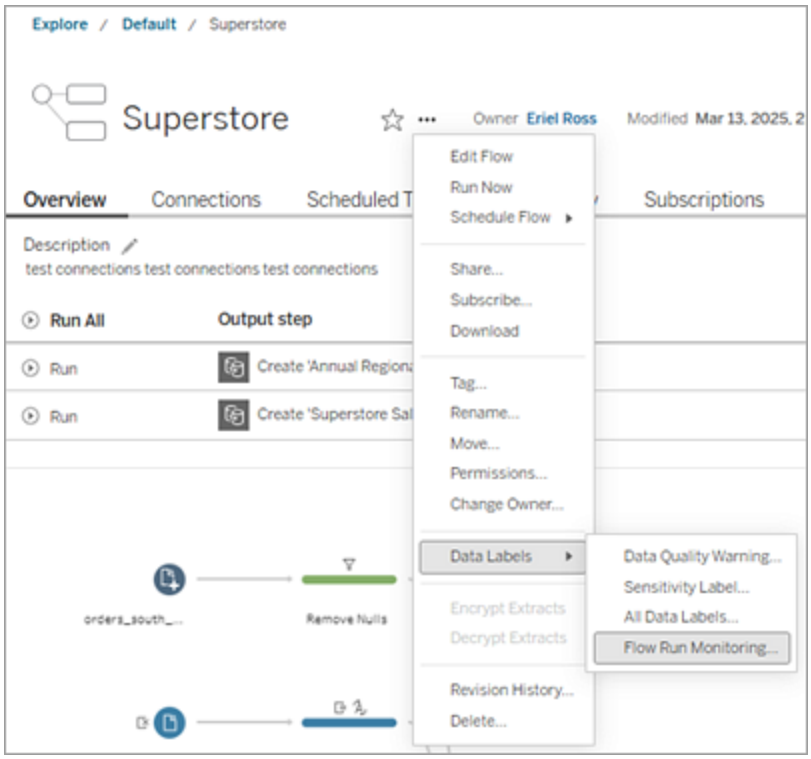
- **完全更新** (既定): すべてのデータを更新し、フローの出力設定に基づいてテーブルにデータを作成または追加します。
- **増分更新**: 新しい行のみを更新し、フローの出力設定に基づいてテーブルにデータを作成または追加します。増分更新は、増分更新を使用するようにフローを設定している場合にのみ選択できます。詳細については、「増分更新を使用したフローデータの更新」を参照してください。

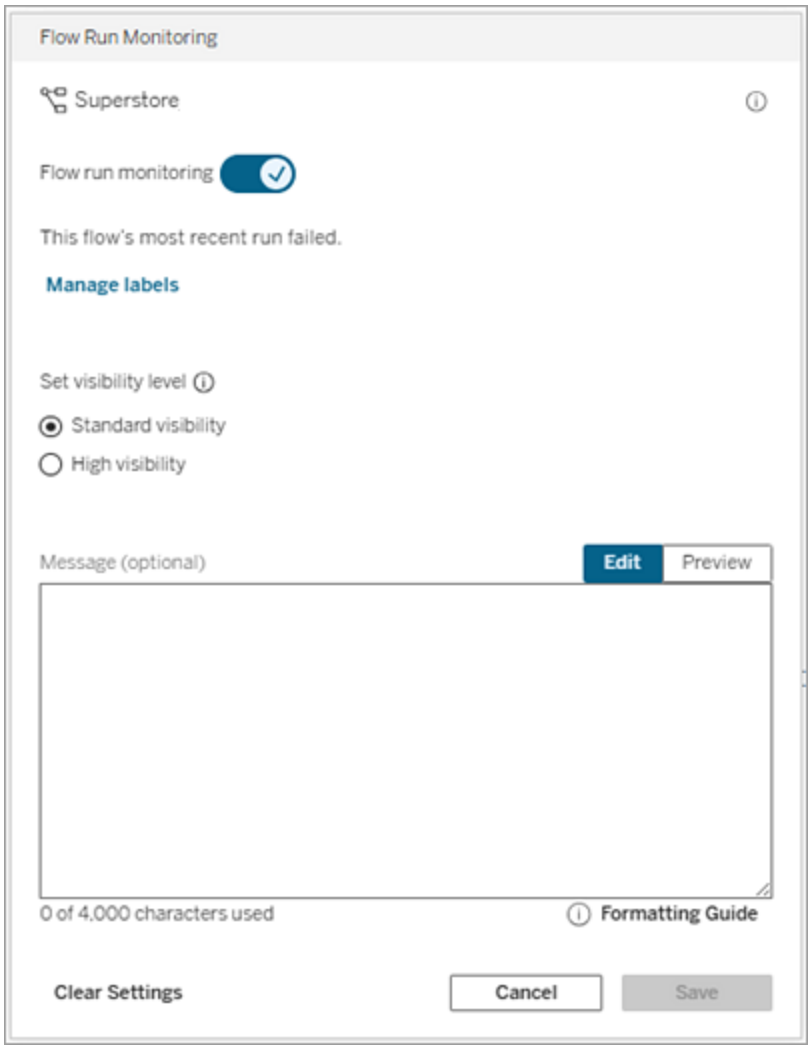
> **注**: Tableau Prep Conductor では、既存の出力が見つからない場合、選択した実行オプションに関係なく、すべての出力に対して完全更新が実行されます。後続のフローを実行するときは、増分更新の構成データが欠落していないか、既存の出力を削除していない限り、増分更新プロセスを使用して、新しい行のみを取得して処理します。

5. (オプション) フローの所有者である場合は、**[完了時にメールを送信]** を選択して、フローが成功したときにユーザーに通知します。フロー実行でメール通知を送信する方法の詳細については、フロー実行の成功をユーザーに通知するを参照してください。

6. データ品質に関する警告を追加します。フロー実行のモニタリングはフロー レベルでオンまたはオフに切り替え可能で、フロー実行が失敗した場合にフローに表示されるデータ品質に関する警告を生成できます。データ品質に関する警告の詳細については、「データ品質に関する警告の設定」を参照してください。

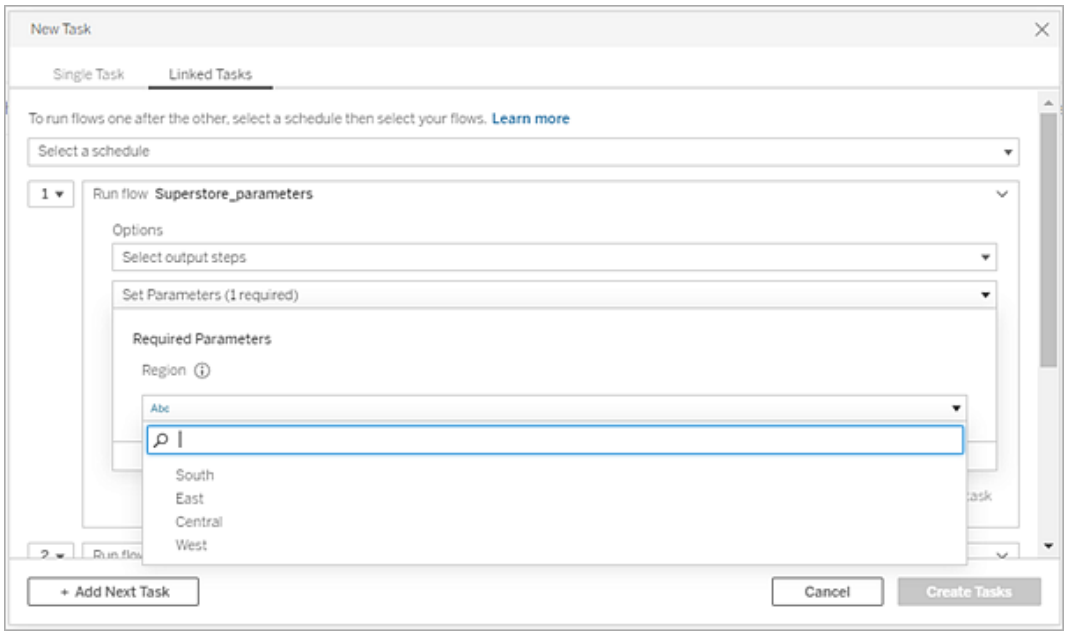
   - **[その他のアクション]** ··· メニューから、フロー名の横にある **[データ ラベル] > [フロー実行のモニタリング]** を選択します。
   - [フロー実行のモニタリング] ダイアログで、フロー実行のモニタリングのオンとオフを切り替え、必要に応じて警告メッセージを入力して、データのユーザーが問題を認識できるようにします。メッセージは、フローが正常に実行されるまで残ります。


Explore / Default / Superstore
Superstore
Owner Eriel Ross
Modified Mar 13, 2025, 2
Overview
Connections
Scheduled T
Subscriptions
Description
test connections test connections test connections
Run All
Output step
Run
Create 'Annual Region
Run
Create 'Superstore Sal
orders_south_...
Remove Nulls
Edit Flow
Run Now
Schedule Flow
Share...
Subscribe...
Download
Tag...
Rename...
Move...
Permissions...
Change Owner...
Data Labels
Encrypt Extracts
Decrypt Extracts
Revision History...
Delete...
Data Quality Warning...
Sensitivity Label...
All Data Labels...
Flow Run Monitoring...



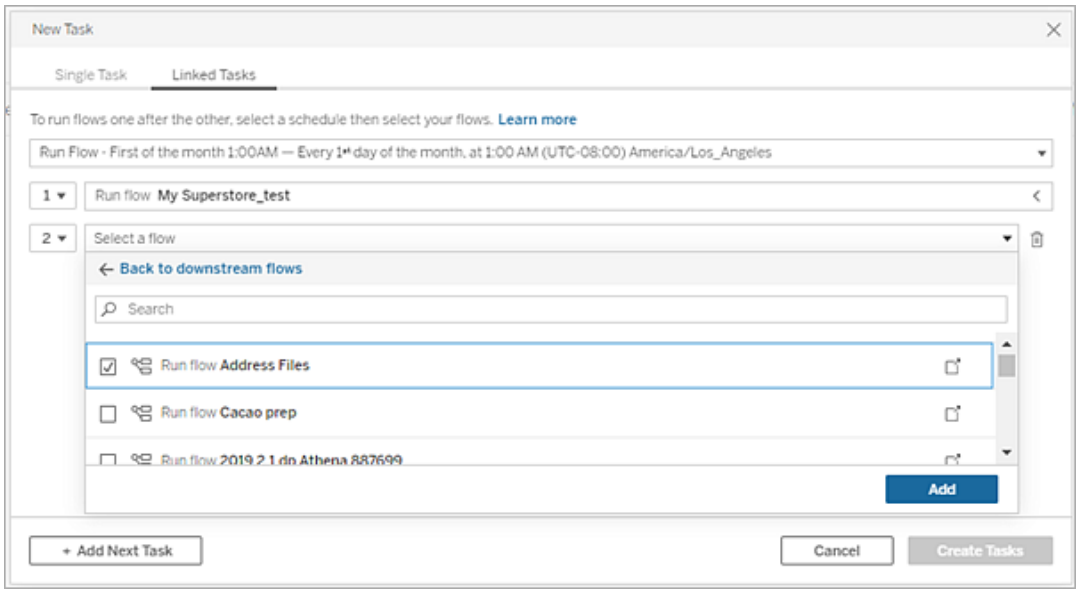
7. フロー障害のオプションを設定します。
   - **残りのタスクの停止**: このオプションを選択すると、リスト内の後続タスクが実行キューに入らないようにできます。
   - **メール通知**: フローが失敗したり、一時停止したり、キャンセルされた場合に、フローの所有者とリンクされたタスクの作成者に自動的にメール通知が送られます。

8. (バージョン 2021.4 以降) フローにパラメーターが含まれている場合は、必須またはオプションのパラメーター値を入力します。フローを実行するために必要な値を入力する必要があります。

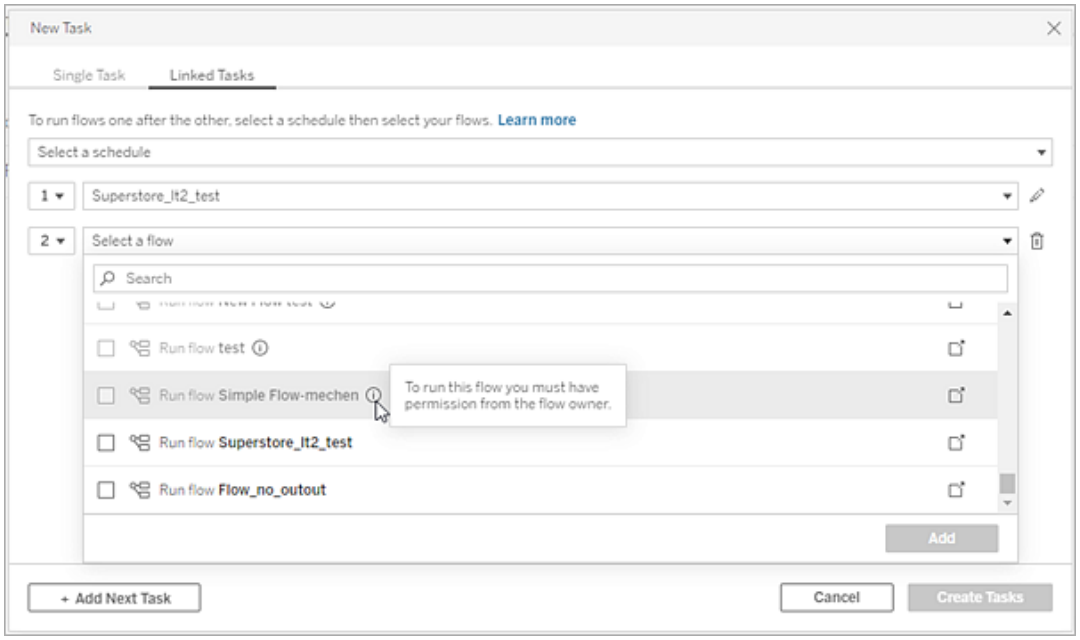


9. 2 番目のタスクのドロップダウンをクリックして、次のフローを追加します。前のフローの出力を使用するフローが自動的に表示されていない場合は、**[すべてのフローを表示]** をクリックして使用可能なすべてのフローを表示します。

   1 つまたは複数のフローを選択して **[追加]** をクリックするか、**[次のタスクの追加]** をクリックしてフロー実行タスクをリストに追加します。



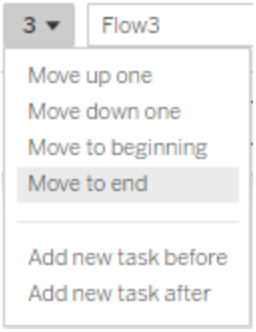
フローを実行する権限がない場合は、フローをリストに追加する前に、フローの所有者に連絡して権限を付与される必要があります。



10. 手順 3 ~ 5 を繰り返して、フロー実行オプションを構成します。

> **注**: バージョン 2021.4 以前では、新しいフローの横にある **[編集]** アイコン をクリックして [オプション] ペインを展開します。

11. (オプション) 番号の付いたタスクの横にあるドロップダウンをクリックしてメニューを開くと、フロー タスクの順序を変更したり、既存のタスクの間に新しいタスクを挿入したりすることができます。



12. **[タスクの作成]** をクリックして、リンクされたタスクのスケジュールを作成します。

実行可能なユーザー

- サーバー管理者は、サーバー内のすべてのサイトでこれを実行できます。サイト管理者は、自身がアクセス権を持つサイトの設定で、タスクのパブリッシュおよびスケジュールをユーザーに許可している場合、これを実行できます。
- リンクされたタスクの場合、サーバー管理者は、リンクされたタスクのスケジュールをユーザーに許可しているすべてのサイトで、これを実行できます。サイト管理者は、自身がアクセス権を持つサイトの設定で、リンクされたタスクのスケジュールをユーザーに許可している場合、これを実行できます。
- Creator サイト ロールを持つフロー所有者およびプロジェクト リーダーは、自身が所有するフローまたはプロジェクトのフロー タスクをそれぞれ作成できます。

サイト管理者、フロー所有者、プロジェクト リーダーは、自身が所有するフローやプロジェクトのフロー タスクをそれぞれ作成できます。フロー タスクを作成または編集できるのは、Creator サイト ロール以上を持つユーザーのみです。

詳細については、「ユーザーのサイトロールの設定」および「コンテンツのパーミッションと所有権」を参照してください

フロー実行の成功をユーザーに通知する

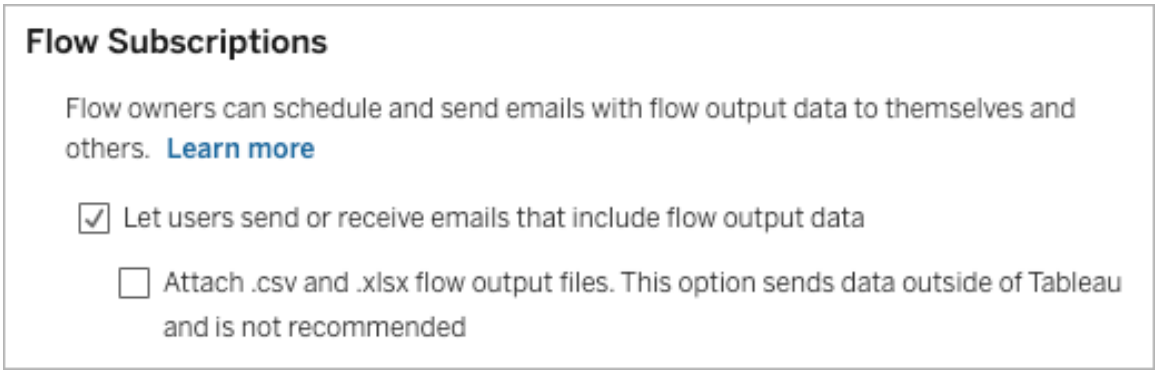
Tableau Prep Builder バージョン 2021.4.1 以降、Tableau Server および Tableau Cloud バージョン 2021.4 以降でサポートされています。この機能を使用するには データ管理 が必要です。

フローの所有者は、自分自身、個々のユーザー、およびグループをサブスクライブして、スケジュールされたタスクについてフローが正常に実行されたことをメールで通知できます。メールには、Tableau 環境内のデータへのリンクが含まれているほか、Tableau Server の場合は、オプションでフロー実行の詳細を Excel ファイルや CSV ファイルで添付することができます。

フロー サブスクリプションは、フローのスケジュールされたタスクに追加されます。フロー サブスクリプションは、新しいフロー タスクを作成するときに、または既存のフロー タスクに対して追加できます。スケジュールされたタスクが正常に完了すると、メール通知が送信されます。

フロー サブスクリプションのサイト設定を構成する

デフォルトでは、**フロー サブスクリプション**のサイト設定は、メール通知を送受信するように有効になっています。



- **[フロー出力データを含むメールの送受信をユーザーに許可する]** オプションを使用すると、フローの所有者は、ユーザーとグループをサブスクライブして、フローが正常に実行されたことを知らせる通知を受け取れるように設定できます。ユーザーは、通知メールをもとにして、完全なデータ ソースにアクセスしたり、Tableau 内からフローの詳細を表示したりできます。

- Tableau Server のみ (非推奨): **[.csv および.xlsx フロー出力ファイルの添付]** オプションを使用すると、フロー所有者は通知メールにファイルを添付できます。メールの受信者を Tableau サーバーに追加する必要がありますが、ファイルにはデータソースが含まれており、Tableau システムの外部に漏えいする可能性があります。このオプションは、オンプレミス環境でのみ使用できます。

フローをパブリッシュする

フロー出力をファイル、データベース表、またはデータソースとしてパブリッシュします。フローを保存するときは、次の点を考慮してください。

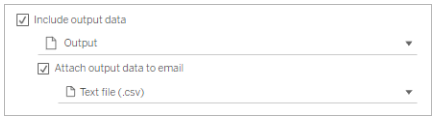
- (オンプレミスのみ) パブリッシュする場合、出力をファイルまたはデータベース表として保存し、.csv または .xlsx のファイルタイプでメールに添付することを選択できます。
- パブリッシュされたデータソースとして出力を保存すると、メール通知で Tableau のフローへのリンクを提供できます。メールにファイルを添付することはできません。
- ファイル出力として保存することを選択する場合は、ネットワーク共有を使用する必要があり、出力と入力の場所がセーフリストに含まれている必要があります。詳細については、「手順 4 - 入力場所と出力場所をセーフリストに登録する」を参照してください。
- フローサブスクリプションは Windows と Linux でサポートされています。Linux のフローサブスクリプションには、次の制限が適用されます。
  - ファイル出力は、Windows サーバーに出力する必要があります。
  - ファイルに出力するフローの場合、パスには次の UNC 形式を使用します: *\\server\path\filename* ローカルドライブ文字は使用しないでください。
  - マウントされたパスは許可リストに登録されている必要があります。
- メールにファイルを添付する場合、Tableau Cloud でのファイルサイズの制限は約 25 MB です。オンプレミスの Tableau Server を使用する場合は、添付ファイルサイズの制限を設定します。

フローサブスクリプションを追加する

1. フローの所有者は、Tableau Server または Tableau Cloud でパブリッシュされているフローから、メール通知のサブスクリプションを追加したいフローを選択します。

2. サブスクリプションは、新しいタスクまたは既存のタスクに追加できます。
   - 新しいタスクにサブスクリプションを追加する場合:
     1. **[スケジュールされたタスク] > [新しいタスク]** をクリックします。
     2. [新しいタスク] ダイアログの **[スケジュールの選択]** ドロップダウン リストから、フローを実行するスケジュールを選択します。
     3. **[完了したらメールを送信]** を有効にします。
   - 既存のタスクにサブスクリプションを追加する場合:
     1. **[サブスクリプション] > [サブスクライブ]** をクリックします。
     2. **[フロー サブスクリプションの追加]** ダイアログで、**[頻度]** ドロップダウン リストからスケジュールを選択します。
3. **[送信先]** フィールドに、本文に挿し込むユーザーまたはグループの名前を入力します。通知を送りたいユーザーとグループを選択します。

   ユーザーとグループは、管理者が Tableau 環境に追加する必要があります。
4. (オプション) 自分を通知に含めるには、**[自分に送信]** チェック ボックスをオンにします。
5. (オプション) **[件名]** フィールドで、フロー実行通知のデフォルトのメール件名をカスタマイズします。
6. (オプション) フロー実行に関する情報を **[メール メッセージ]** テキスト ボックスに追加します。
7. **[出力データを含める]** をクリックして、メールに含める出力のタイプを選択します。
   - Tableau Server の場合、フローをファイルまたはデータベース表の出力としてパブリッシュした場合は、データ ソースを含む .csv または .xlsx のファイルをメールに添付することを選択できます。ただし、データが Tableau システムの外部に漏えいする可能性があるため、推奨しません。

   - フローをデータ ソースとしてパブリッシュした場合は、データ ソースへのリンクを含めることを選択できます。メールへのファイルの添付はサポートされていません。

8. 既存のタスクにサブスクリプションを追加する場合は、[サブスクライブ] をクリックします。

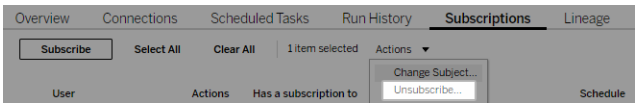
フローのサブスクリプションを解除する

メール通知の配信を停止するには、次の手順を行います。

1. 登録メールの下にある **[サブスクリプションの解除]** をクリックします。
2. フローの所有者として、Tableau Server または Tableau Cloud にサインインします。ページの上部で、**[通知]** アイコンをクリックします。
3. **[...]** メニューを展開し、**[通知の解除]** を選択します。

フロー所有者としてサブスクリプションを解除して削除するには、次の手順を行います。

1. **[サブスクリプション]** をクリックします。
2. Tableau Server または Tableau Cloud でパブリッシュされたフローを開きます。
3. フローサブスクリプションの一覧から、サブスクリプションを解除するフローの選択ボックスをクリックします。
4. **[アクション] > [サブスクリプションの解除]** を選択します。



サブスクリプションを表示する

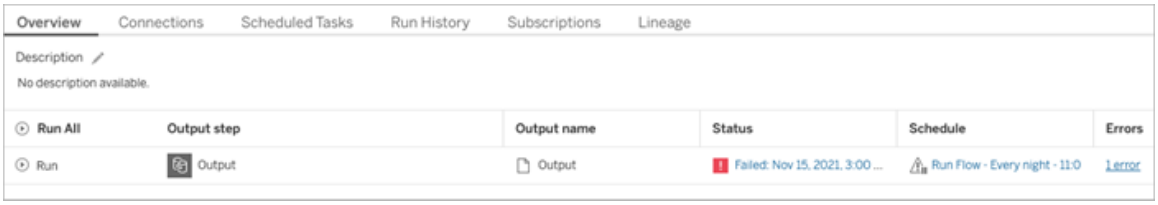
現在のフローサブスクリプションは、Tableau Server または Tableau Cloud で表示できます。

- フローの **[概要]** ページの **[サブスクリプション]** タブから、現在のサブスクリプションの一覧を確認できます。
- **[タスク]** ページの **[サブスクリプション]** タブから、ワークブックのサブスクリプションといっしょにサブスクリプションの一覧を確認できます。

一時停止したフロー サブスクリプションの再開

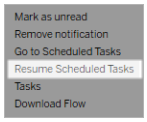
フローの問題が原因でサブスクリプションが失敗する場合があります。サブスクリプションの失敗が 5 回を超えると、スケジュールされたフロー タスクが一時停止したことを通知するメールを受け取ります。

フローの [概要] ページから、スケジュールされたフロー タスクがいつ失敗したか確認できます。

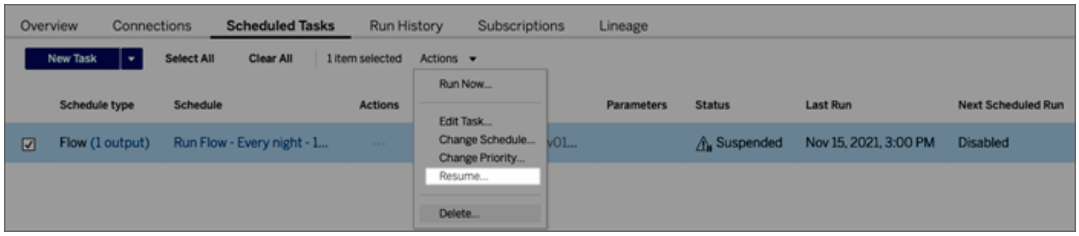


一時停止したフロー タスクを再開する方法はいくつかあります。フロー所有者の場合:

- Tableau Web ページの [マイ コンテンツ] エリアの [最終更新] 列に、サブスクリプションが一時停止していることを示すアイコンが表示されます。**[...] > [スケジュールされたタスクの再開]** を選択して再開します。



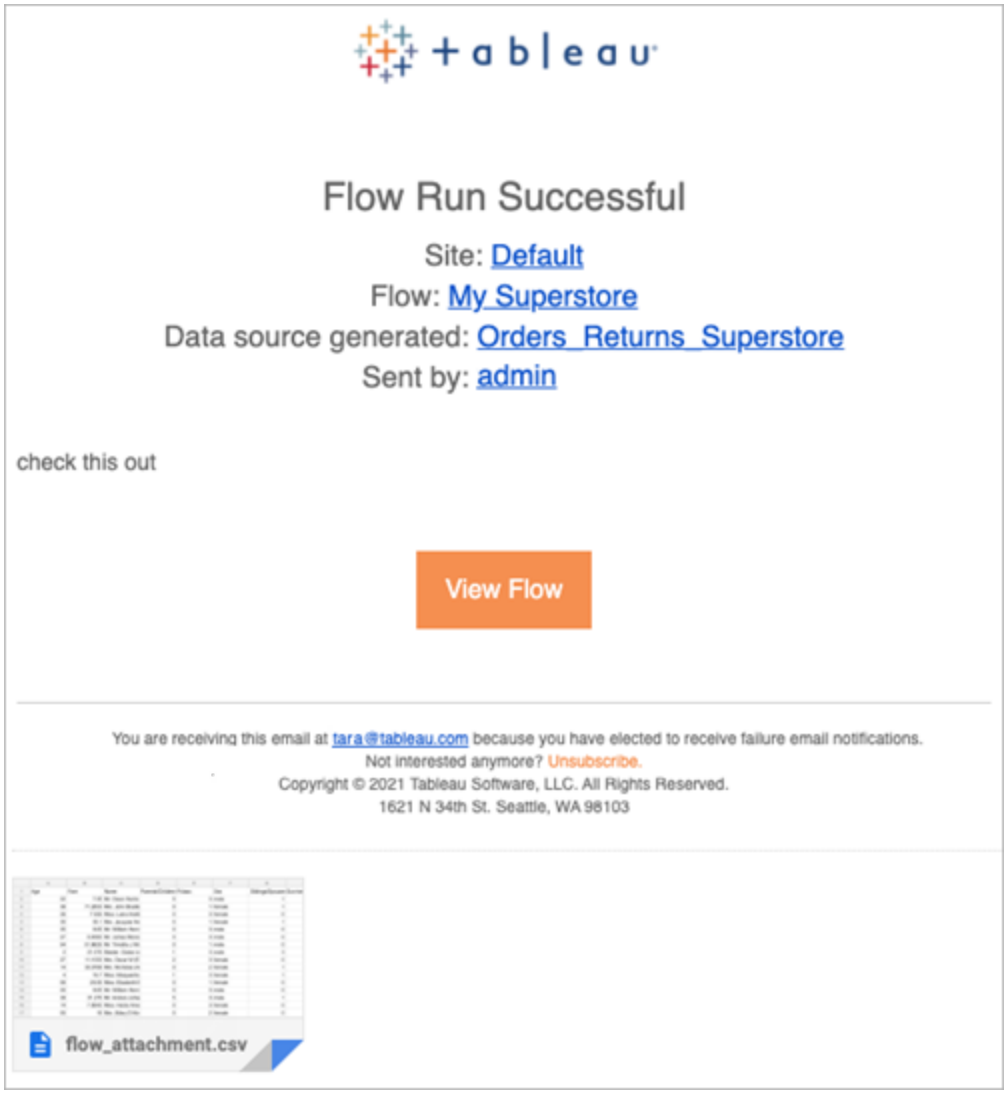
- [スケジュールされたタスク] ページの [最終更新] 列に、サブスクリプションが一時停止していることを示すアイコンが表示されます。フローを選択し、**[アクション] > [再開]** をクリックします。



通知メールからフロー データにアクセスする

フロー通知の設定によっては、通知メールからデータ ソースや添付ファイルにアクセスできます。

- [フローを表示] をクリックして、Tableau Server または Tableau Cloud でフローを開きます。
- 添付ファイルをクリックすると、フロー データが表示されます。



実行可能なユーザー

- フローの所有者は、自分が所有するフローに対してフロー通知サブスクリプションを作成できます。
- 通知を受け取るには、管理者がユーザーとグループをTableau環境に追加する必要があります。

詳細については、次の記事を参照してください。

- Windows:「ユーザーのサイトロールの設定」および「コンテンツのパーミッションと所有権」
- Linux:「ユーザーのサイトロールの設定」および「コンテンツのパーミッションと所有権」

フローの管理

フローを Tableau Server または Tableau Cloud にパブリッシュしたら、フローを管理し必要に応じて変更することができます。このトピックでは、フローを管理するために実行できるさまざまなアクションについて説明します。

> **注**: このトピックの内容は Tableau Server と Tableau Cloud の両方に適用され、例外があれば具体的に明示されます。

フローの管理

以下は、フローを管理するために実行できるアクションの一覧です。

- **フローの作成:** バージョン 2020.4 以降、作成者は Web 上で直接フローを作成できます。**ホーム** ページで **[作成] > [フロー]** をクリックするか、**[探索]** ページで **[新規作成] > [フロー]** をクリックします。詳細については、「Web 上の Tableau Prep」を参照してください。

- **フローの編集:** バージョン 2020.4 以降、作成者は Web 上で直接フローを編集できます。フローのリストに移動し、**[アクション]** を選択して **[フローの編集]** をクリックするか、フローを開いて **[編集]** ボタンをクリックします。

  フローを編集すると、変更した内容は下書きの状態に移ります。編集を完了したら、フローをパブリッシュして変更をコミットし、新しいバージョンのフローを作成します。詳細については、自動保存と下書き作業」を参照してください。

  データ接続を編集するには、**[接続]** ビューで、そのデータソースの **[アクション]** (…) メニューから、**[接続の編集]** メニュー オプションを使用します。新しい接続は保存され、Tableau Prep Conductor でフローが次回実行されるときに使用されます。ただし、フローを開いて表示すると、元の接続は保持され、フローを編集してコネクタを更新し、再パブリッシュするまで更新されません。

元の接続を変更せずにフローを開いて編集し、再パブリッシュすると、**[接続の編集]** メニューを使用して行われた接続の変更は、上書きされて元の接続に置き換えられます。

- **フローの実行:** Creator は、特定の時刻に実行されるスケジュール済みのフロータスクを作成できるだけでなく、手動でフローを実行することもできます。
-

> **注**: データ管理は、フローを手動で実行する場合は必要ありませんが、フローをスケジュール設定して実行する場合は必要です。

- フローのリストに移動し、実行するフローを 1 つまたは複数選択します。**[アクション]** を選択し、**[今すぐ実行]** をクリックします。
- **タグ:** タグは、コンテンツの検索、フィルター、および分類に役立つフローに対して作成できるキーワードです。作成者は、フローを公開するときにタグを追加できます。ただし、タグをアクセス権限のある .twb、表示、またはデータソースに追加したり、さらに追加したタグを削除したりすることもできます。タグをフローのリストに追加できます。

  フローのリストに移動し、タグ付けする項目を 1 つまたは複数選択します。**[アクション]** を選択し、**[タグ]** をクリックします。タグを特定のフローに追加するには、前述のフローのリストからそのタグを実行します。[フロー]を開き、[概要] タブから **[アクション]** を選択して、**[タグ]** をクリックします。

- **所有者の変更:** 管理者とフロー所有者は、所有者を自分のサイト上の任意の Creator に変更できます。バージョン 2025.1 以前では、所有者を自分自身にしか変更できませんでした。詳細については、「コンテンツ所有権の管理」を参照してください。
- **権限:** ユーザーの権限を設定し、保存操作、ダウンロード操作、別のプロジェクトへの移動や削除のような、編集操作を実行できるかどうかを指定できます。さらに、フローを表示および実行できるユーザーを指定することもできます。
- **ダウンロード:** Tableau Prep を使用すると、フローをダウンロードして表示または変更できます。フローをダウンロードするには、ダウンロード権限が必要です。所有者であればデフォルトにより上記を備えていますが、他のユーザーに対しては追加する必要があります。

- **リビジョン履歴:** フローに変更を加え、同じ名前の同じプロジェクトに対してパブリッシュすると、フローの新しいバージョンが作成されます。リビジョン履歴は、[アクション]メニューから[リビジョン履歴]を選択して表示できます。フロー所有者には、以前のバージョンのフローを復元する権限があります。

- **移動:** プロジェクト間でフローを移動できます。フローを移動するには、[移動]権限が必要です。所有者であればデフォルトにより上記を備えていますが、他のユーザーに対しては追加する必要があります。

- **名前の変更:** フローの名前を変更できます。フローの名前を変更するには、ユーザーには保存権限が必要です。既定では、自分が所有するフローに対しての権限はありますが、他のユーザーに対して追加する必要がある場合があります。

- **削除:** フローを削除できます。フローを削除するには、ユーザーには削除権限が必要です。所有者であればデフォルトにより上記を備えていますが、他のユーザーに対しては追加する必要があります。バージョン 2025.3 以降では、削除されたフローをごみ箱から復元できるようになりました。詳細については、「削除されたフローの復元」を参照してください。

実行可能なユーザー

Tableau Server の管理者

すべてのサイトで次のタスクを実行できます。

- フローの作成
- パブリッシュ済みのフローの編集
- フローの下書きの一覧表示
- フローの実行
- フローの削除
- ごみ箱をオンにして復元する
- 削除されたフローの復元

- フローのダウンロード
- 所有者の変更
- 権限の変更
- プロジェクトの変更
- タグの追加と削除
- 説明の変更
- 名前の変更
- フロータスクの更新
- フロータスクの削除
- フロータスクの作成*
- バージョン管理*

*以下のアクションには、追加の条件が適用されます。

- フロータスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロータスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成できます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
  - このフローには少なくとも1つの出力ステップが必要です。
  - このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。
- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

Tableau サイト管理者

サイト管理者を務めるサイトに公開されているフローに次のタスクを実行できます。

- フローの作成
- パブリッシュ済みのフローの編集
- フローの下書きの一覧表示
- フローの実行
- フローの削除
- ごみ箱をオンにして復元する
- 削除されたフローの復元
- フローのダウンロード
- 所有者の変更
- 権限の変更
- プロジェクトの変更
- タグの追加と削除
- 説明の変更
- 名前の変更
- フロータスクの更新
- フロータスクの削除
- フロータスクの作成*
- バージョン管理*

以下のアクションには、追加の条件が適用されます。

- フロー タスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロー タスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成できます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
  - このフローには少なくとも1つの出力ステップが必要です。
  - このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。
- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

プロジェクトリーダー

プロジェクトリーダー権限を持つプロジェクトにパブリッシュされたフローに次のタスクを実行できます。

- フローの作成
- パブリッシュ済みのフローの編集
- フローの実行
- 削除
- ダウンロード
- 権限の変更
- プロジェクトの変更
- タグの追加と削除
- 説明の変更

- 名前の変更
- フロータスクの更新
- フロータスクの削除
- フロータスクの作成*
- バージョン管理*

*以下のアクションには、追加の条件が適用されます。

- フロータスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロータスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成できます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
  - このフローには少なくとも1つの出力ステップが必要です。
  - このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。
- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

プロジェクト所有者

所有するプロジェクト宛に発行されたフローに次のタスクを実行できます。

- フローの作成
- パブリッシュ済みのフローの編集
- フローの実行
- 削除

- ダウンロード
- 権限の変更
- プロジェクトの変更
- タグの追加と削除
- 説明の変更
- 名前の変更
- フロータスクの更新
- フロータスクの削除
- フロータスクの作成*
- バージョン管理*

*以下のアクションには、追加の条件が適用されます。

- フロータスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロータスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成できます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
  - このフローには少なくとも1つの出力ステップが必要です。
  - このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。
- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

フロー所有者

所有するフローに次のタスクを実行できます。

- フローの作成
- 下書き(所有するフロー)とパブリッシュ済みフローの編集
- フローの実行
- フローの削除
- 削除したフローの復元(ごみ箱が管理者によってオンになっている場合)
- フローのダウンロード
- 所有者の変更
- 権限の変更
- プロジェクトの変更
- タグの追加と削除
- 説明の変更
- 名前の変更
- フロータスクの更新
- フロータスクの削除
- フロータスクの作成*
- バージョン管理*

*以下のアクションには、追加の条件が適用されます。

- フロータスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロータスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成でき

ます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
- このフローには少なくとも1つの出力ステップが必要です。
- このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。

- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

Creatorライセンスを持つユーザー

次のタスクを実行できます。

- フローの作成
- 下書き(所有するフロー)とパブリッシュ済みフローの編集
- フローの実行 (フローの実行権限により)
- 削除 (削除権限により)
- 削除したフローの復元 (ごみ箱が管理者によってオンになっている場合)
- ダウンロード(ダウンロードまたは名前をつけて保存、および読み取り権限)
- 権限の変更(変更権限により)
- プロジェクトの変更(移動権限、および移行先プロジェクトに対する書き込み権限による)
- タグの追加と削除(読み取り権限による)
- 説明の変更(保存権限による)
- 名前の変更(保存権限による)
- フロータスクの更新(実行権限による)
- フロータスクの削除(実行権限による)

- フロータスクの作成*(実行権限による)
- バージョン管理*(表示、読み取り、名前を付けて保存、ダウンロード権限による)

*以下のアクションには、追加の条件が適用されます。

- フロータスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロータスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成できます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
  - このフローには少なくとも1つの出力ステップが必要です。
  - このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。
- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

Explorerライセンスを持つユーザー

次のタスクを実行できます。

注: バージョン 2020.4 以降、Explorer ライセンスのユーザーは Tableau Server 上でフローを実行できなくなりました。

- Explorer ライセンスのユーザー(フロー実行のパーミッションを持つ) は、Tableau Cloud でフローを実行できます。
- 削除 (削除権限により)

- 削除したフローの復元 (Explorer (パブリッシュ可能) の権限があり、ごみ箱が管理者によってオンになっている場合)
- ダウンロード(ダウンロードまたは名前をつけて保存、および読み取り権限)
- 権限の変更(変更権限により)
- プロジェクトの変更(移動権限および移行先プロジェクトに対する書き込み権限による)
- タグの追加と削除(読み取り権限による)
- 説明の変更(保存権限による)
- 名前の変更(保存権限による)
- フロータスクの更新(実行権限による)
- フロータスクの削除(実行権限による)
- フロータスクの作成*(実行権限による)
- バージョン管理*(表示、読み取り、名前を付けて保存、ダウンロード権限による)

*以下のアクションには、追加の条件が適用されます。

- フロータスクを作成するには:
  - フロースケジュールが利用可能である必要があります。Tableau Server では、スケジュールを作成できるのはサーバー管理者だけです。Tableau Cloud では、フロータスクをスケジュールするパーミッションを持つユーザーがカスタム スケジュールを作成できます。詳細については、「フロー タスクのスケジュール設定」を参照してください。
  - このフローには少なくとも1つの出力ステップが必要です。
  - このフローバージョンは、Tableau Server バージョンと互換性がある必要があります。
- バージョン管理:
  - サイトでリビジョン履歴を有効にする必要があります。
  - ユーザーロールがプロジェクトへのパブリッシュが可能である必要があります。

ビューワーライセンスを持つユーザー

ビューワーはフローを管理することはできません。ただし、フローおよび別のバージョンのフローを表示することはできます。

フローに設定できるフル機能の詳細については、パーミッションの機能を参照してください。

フローの正常性とパフォーマンスの監視

Tableau Prep フローをパブリッシュし、定期的な実行スケジュールを設定したら、フローが予想どおりに実行されたことや、問題の発生時にどのように解決したかを確認できます。また、フローを監視し、トラブルシューティングを行い、フローのパフォーマンスを確認することもできます。

このトピックでは、Tableau Prep フローの監視とトラブルシューティングに役立つ、Tableau Server とが提供するさまざまな方法について説明します。

発生した問題を検出して解決する

Tableau Server を設定して、フローが失敗したときにメール通知を送信したり、Tableau Server で [アラート] メニューを使用してエラーを見つけて確認したりできます。また、関心のあるフローを [フロー] ページで確認することもできます。この種類の監視では、問題が発生した時点で問題を検出できます。

フローが失敗したときに通知を受け取る

フロー障害に関する通知をメール、Tableau サイト、または Slack を介して送信するように Tableau Server を構成できます。Tableau Server およびサイトの通知を設定するには、「手順 2 - Tableau Server のフロー設定の構成」の手順に従ってください。

エラーを表示して解決する

> **注**: バージョン 2020.4.1 以降では、Tableau Server および Tableau Cloud でフローの作成と編集を行えるようになりました。このセクションの内容は、特に記載がない限り、すべてのプラットフォームに適用されます。Web 上でのフローの作成の詳細については、Web 上の Tableau Prepを参照してください。

フローの実行中は、次のようなエラーが発生する場合があります。

- **接続エラー:** 接続エラーは通常、Tableau Server が 1 つ以上のデータ入力に接続できない場合、または 1 つ以上の出力ステップで接続できない場合に発生します。
    - 入力接続エラーの場合は、**[接続]** タブの **[接続の編集]** オプションを使用して接続の詳細を変更してから、フローを再度実行します。
    - 出力接続エラーの場合は、フローの出力ステップが出力する場所を確認します。フローの出力がネットワーク共有で実行される場合は、出力ステップで安全な場所を指定していることを確認してください。変更を行うと、フローが再パブリッシュされ、実行が再試行されます。

      > **注**: ファイルまたはネットワーク共有に出力されるフローの出力接続エラーを修正するには、フローを Tableau Prep Builder にダウンロードしてから、フローをサーバーにもう一度パブリッシュします。パブリッシュされたデータソースまたはデータベースに出力されるフローは、Web 上で直接編集できます。

- **フロー内のエラー:** フローの 1 つ以上のステップでエラーがある場合、エラー メッセージが表示されます。フローは、Web 上で直接編集してもう一度パブリッシュできます。フローを Tableau Pre Builder にダウンロードしてエラーを解決し、フローをサーバーにパブリッシュし直して、もう一度実行します。

- **フロー タスクの中断:** スケジュールされたフロー タスクの実行が構成した回数失敗すると、このフロー タスクは中断されます。既定では、フロー タスクが 5 回連続して失敗すると、中断されます。

  1 つのフローに複数のスケジュール タスクを割り当てることができますが、失敗したタスクのみが中断されます。他のすべてのフロー タスクは、エラーが発生しない限り引き続き実行されます。中断したタスクを再開するには、エラーを確認して問題を解決してから、フローをオンデマンドで実行するか、割り当てられたスケジュールに基づいて自動的にフローを実行するようにします。

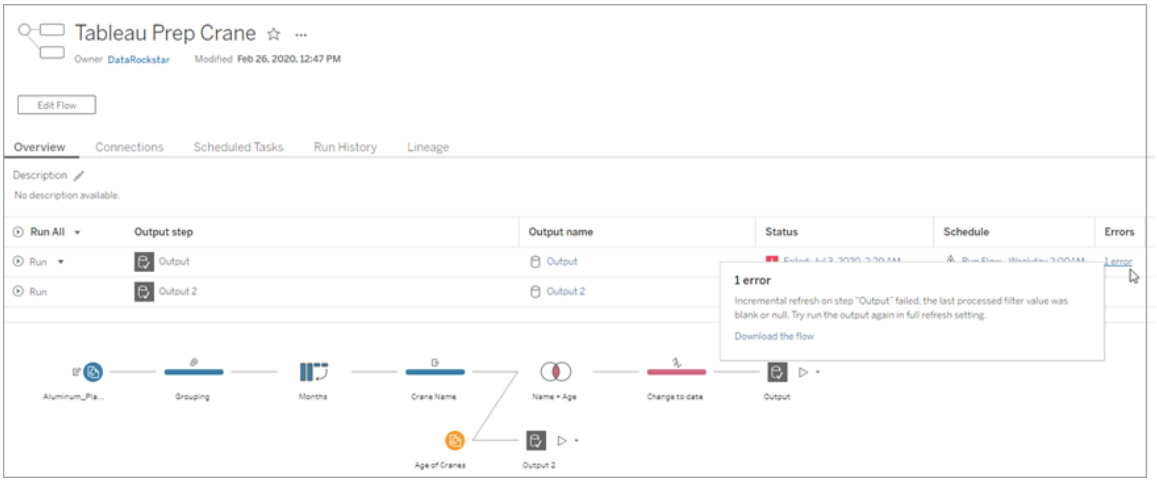
> **注:** サーバー管理者は、tsm configuration set のオプションを使用して、フロー実行が中断されるまでの試行回数を構成できます。詳細については、手順 5 - オプションのサーバー構成を参照してください。

エラーは、次のページで確認することができます。

フローの [概要] ページ

このページでは、最新のフロー実行とエラーのステータスを確認できます。エラー テキストにカーソルを合わせて、エラーの詳細を確認します。スケジュール タスクが中断された場合は、警告アイコンがスケジュールの横に表示されます。アイコンにカーソルを合わせるとステータスが表示されます。

フローの失敗やタスクの中断の原因となったエラーを解決したら、フローを手動で実行するか、割り当てられたスケジュールに基づいてフローを実行できます。フロー タスクの中断の場合は、中断したタスクのツールヒントで **[スケジュールされたタスクに移動]** リンクをクリックして **[スケジュールされたタスク]** ページに移動し、**[スケジュールされたタスクの再開]** ボタンをクリックして中断したタスクを再開します。

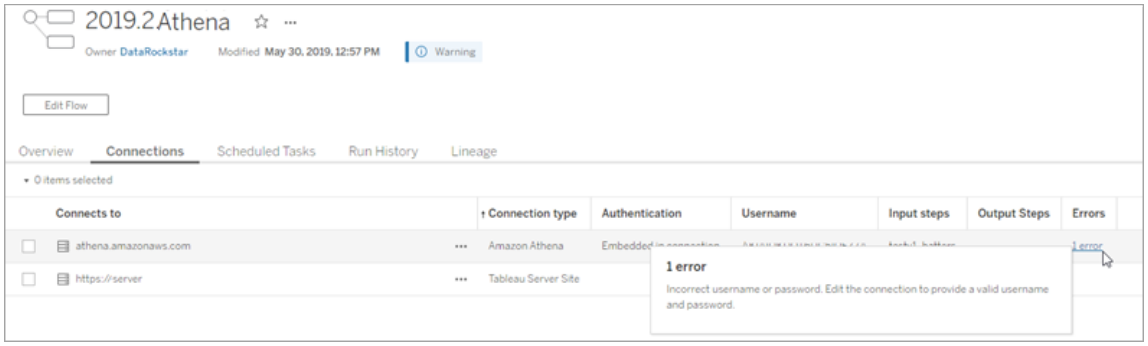


[接続] ページ

[接続] ページには、最新のステータスと関連するすべての接続エラーが表示されます。入力エラーを修正するには、入力接続の **[その他のアクション]** ••• メニューをクリックして接続を編集し、サー

バー名、ポート、ユーザー名、パスワードを変更します。

出力接続のエラーを修正するには、Tableau Prep Builder でフローを直接変種するか、フローをダウンロードしてファイル パスを修正し、フローをもう一度パブリッシュして実行します。



[スケジュール タスク] ページ

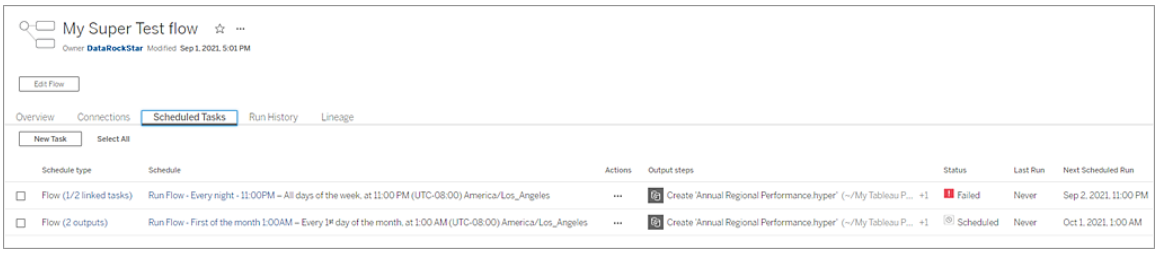
注: このタブを表示するには、データ管理 が必要です。

フローに割り当てられている、スケジュール タスクが表示されます。スケジュール タスクが中断している場合は、そのタスクのステータスが表示され、このページから手動でフロー タスクを再開できます。中断しているタスクを再開する前に、フローにあるすべてのエラーを解決してください。

このページにはエラーの詳細は表示されませんが、**[概要]** ページまたは **[実行の履歴]** ページでエラーの詳細を確認できます。**[スケジュールの種類]** 列のリンクをクリックして、スケジュールされた内容の詳細を表示したり、タスクを編集したりすることもできます。

フローの接続を編集するか、フロー タスクを手動で実行すると、中断したスケジュール タスクは、フロー タスクが再パブリッシュされた時点で自動的に再開されます。中断したタスクを手動で再開するには、**[スケジュールされたタスク]** ペインで **[スケジュールされたタスクの再開]** をクリックします。この操作により、中断していたすべてのフローのタスクが再開されます。

タスクを個別に再開するには、スケジュールされたタスクの **[その他のアクション]** ••• メニューをクリックして、**[再開]** を選択します。**[今すぐ実行]** をクリックして、フローのすべてのタスクをすぐに実行することもできます。

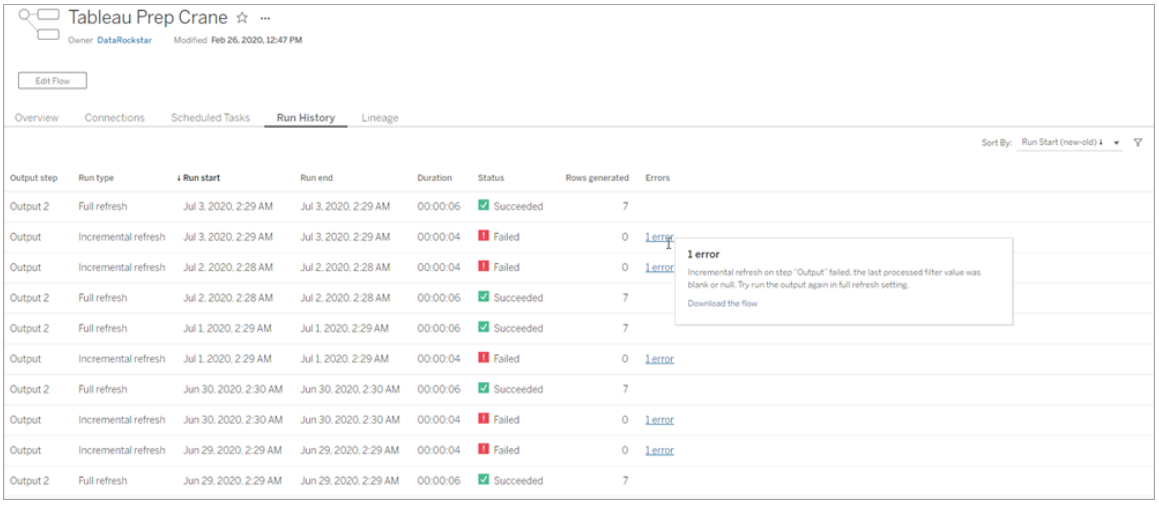


[実行の履歴] ページ

注: このタブを表示するには、データ管理 が必要です。

[実行の履歴] ページには、完了済みおよび処理中のすべてのフロー実行の詳細が、出力ごとに表示されます。**[エラー]** 列のエラーにカーソルを合わせると、エラーの詳細が表示されます。[時間] 列には、フローの実行時間が表示されます。

注: バージョン 2020.2.1 以降では、**[実行タイプ]** フィールドに出力の更新タイプが表示されます。以前のリリースでは、このフィールドに、出力をスケジュールに従って実行したか、オンデマンドで実行したかが示されていました。出力の更新タイプの設定についての詳細は、「増分更新を使用したフローデータの更新」を参照してください。



アラート

フローが失敗すると、[アラート] メニューに、フローの再実行や、フローをダウンロードしてトラブルシューティングを行うオプションと共にエラーの詳細が入力されます。

> **注**: フローの所有者、サーバーまたはサイトの管理者は、このメニューを表示できます。

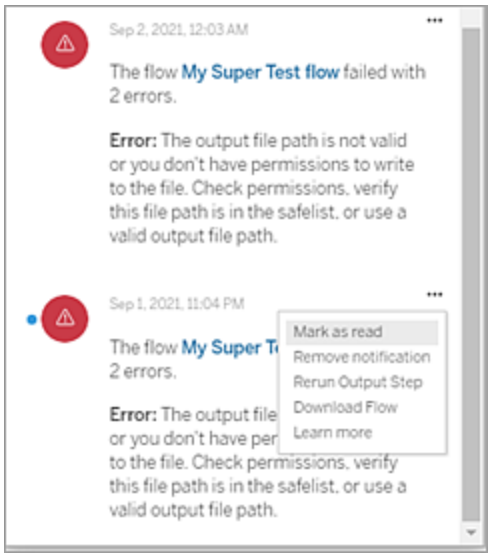


Tableau Prep Conductor プロセスのステータス

- Tableau サービス マネージャー (TSM) のステータス ページには TSM でアクセスでき、TSM 管理者が表示可能です。このページを表示するには、TSM にログインできる必要があります。
- Tableau Server ステータス ページは、Tableau Server の Web UI に表示され、Tableau Server 管理者がアクセス可能です。プロセスのステータス インジケーターにカーソルを合わせると、ツールヒントでプロセスが実行中のノード名とポートが表示されます。Tableau Server ステータス ページには TSM プロセスが表示されません。

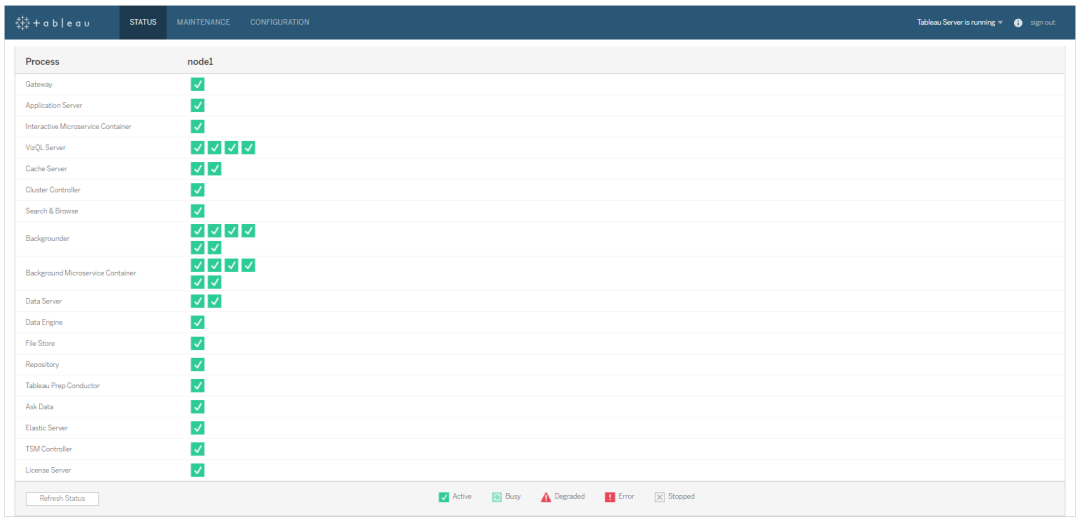


Tableau Server が正常に機能している場合、Tableau Prep Conductor は [アクティブ] または [ビジー] として表示されます。

- **アクティブ:** プロセスは意図したとおりに機能しています。
- **ビジー**: プロセスは一部のタスクを処理中です。
- **停止中**: プロセスが停止しています。この意味合いはプロセスによって異なります。
- **ステータスを使用できません**: Tableau Server はプロセスのステータスを判定できません。

実行可能なユーザー

- **Tableau Server 管理者**は、以下を実行できます。
  - サーバーレベルでメール通知を設定する
  - サイトのメール通知を設定する
  - エラーを表示する
  - 中断したタスクを再開する
  - アラートを表示する
  - プロセスのステータスを表示する
- **Tableau サイト管理者**は、以下を実行できます。
  - サイトレベルでメール通知を設定する
  - エラーを表示する

    - 中断したタスクを再開する
    - アラートを表示する
- **フローの所有者、プロジェクト リーダー、およびフローを表示する権限が付与されているすべてのユーザー**は、以下を実行できます。
    - エラーを表示する
    - 中断したタスクを再開する
    - アラートを表示する(フローの所有者)

フローの管理ビュー

管理ビューを使用すると、フロー、パフォーマンス履歴、ディスク使用量に関連するアクティビティを監視できます。**[ステータス]** ページにはさまざまな管理ビューを備えたTableau ワークブックが埋め込まれており、これらのビューを利用するとさまざまな種類のサーバーやサイトのアクティビティを監視することができます。

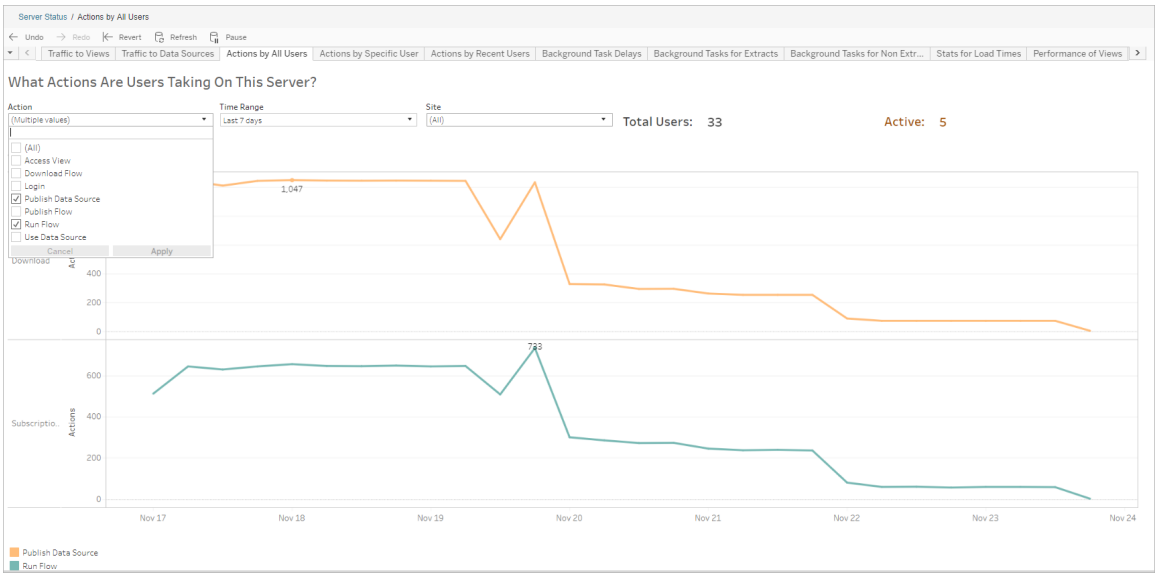
Tableau Server インストールの場合、管理ビューを表示するには、PostgreSQL ドライバーをインストールしておく必要があります。詳細については、「データベース ドライバー」(Linux)、「データベース ドライバー」(Windows) を参照してください。サーバー管理者は、これらのビューを使用すると、サーバー レベル(全サイトの集計)または特定のサイトでアクティビティを確認できます。サイトでフィルターをかけることができるのは、サーバー管理者だけです。

実行可能なユーザー

Tableau Server 管理者とTableau サイト管理者は、管理ビューの表示と操作を行うことができます。サイトごとにフィルターをかけることができるのは、サーバー管理者だけです。

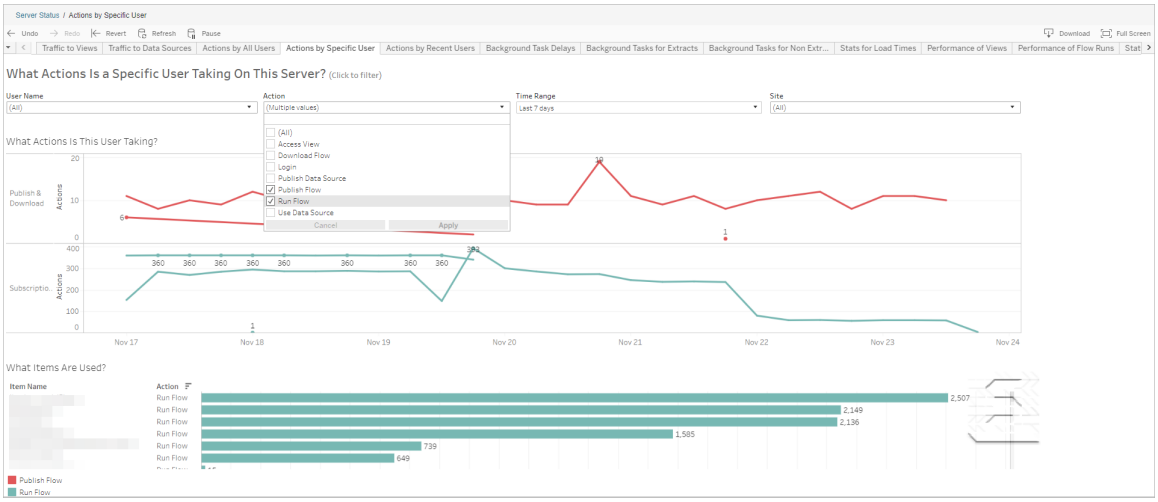
すべてのユーザーによるアクション

このビューを使用すると、フローがどのように使用されているかについての洞察を収集できます。これには、パブリッシュ、ダウンロード、フロー実行などのアクションが含まれます。ビューは、アクション、サイト、時間範囲などでフィルターリングできます。[合計ユーザー数] は、アクションを実行したユーザー数を表します。この値は、フィルターリングの影響を受けません。アクティブ ユーザー数は、選択した期間中にアクティブであり、選択したアクションのいずれかを実行したユーザーの数を示します。



特定のユーザーによるアクション

このビューを使用すると、個々のユーザーがフローをどのように操作しているかについての洞察を収集できます。ビューは、ユーザー名、アクションの種類、時間範囲、およびサイトでフィルターリングできます。



最近のユーザーによるアクション

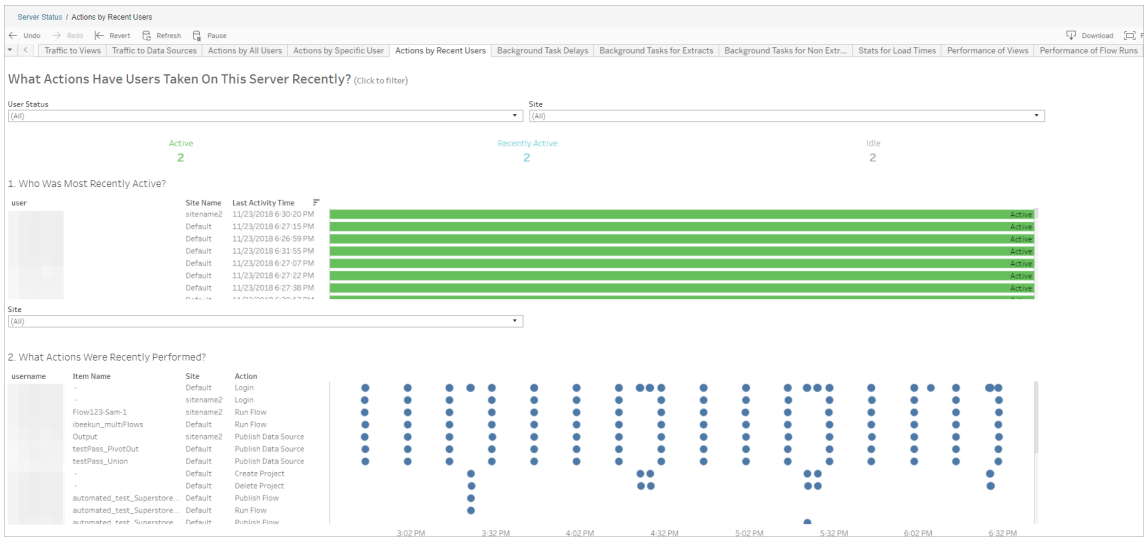
このビューを使用すると、過去 24 時間に Tableau Server または でアクティブであったユーザーが表示されます。

サーバー上でメンテナンス作業を行う必要がある場合、どのユーザーにどれくらいの影響があるか、さらにはそれらのユーザーが何をしているのかについて知るのに役立ちます。

このビューには、最近 Tableau Server または にサインインした、**[アクティブ]**、**[最近アクティブ]**、および **[アイドル]** のユーザーが表示されます。

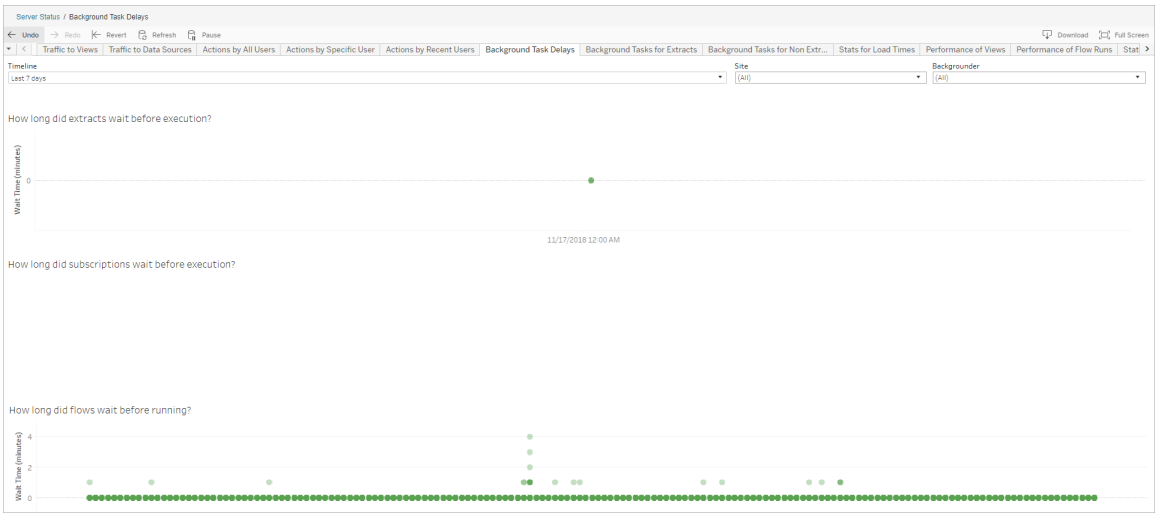
このビューでは、アクティブなユーザーとは直近の 5 分間にアクションを実行したユーザー、最近アクティブなユーザーとは直近の 30 分間にアクションを実行したユーザー、アイドルなユーザーとは、最後にアクションを実行してから30 分以上が経過したユーザーを指します。

ユーザーを選択すると、そのユーザーが最近実行したアクションのみを表示できます。アクションの詳細を表示するには、アクションの上にマウスを置きます。



バックグラウンドタスクの遅延

このビューには、抽出更新タスク、サブスクリプション タスク、フロー タスクの遅延が表示されます。この遅延は、タスクの実行がスケジュールされていた時刻から、実際に実行された時刻までの時間の長さです。このビューを使用すると、タスク スケジュールの分散やタスクの最適化によってサーバーのパフォーマンスを改善できる余地がどこにあるかを特定できます。



遅延の考えられる原因と遅延を小さくする方法には、次のものがあります。

- 多くのタスクが同時にスケジュールされている。

  ビューの例では、長い遅延を示すタスクは毎日同時にクラスター化されています。これにより、待機時間の急増が生じます。[タイムライン] フィルターを 1 日に設定すると、タスクの遅延が時間ごとに表示されるので、多くのタスクが同時にスケジュールされている時間帯を特定できます。1 つの解決策は、タスクをオフピークの時間帯に分散して、サーバーの負荷を軽減することです。

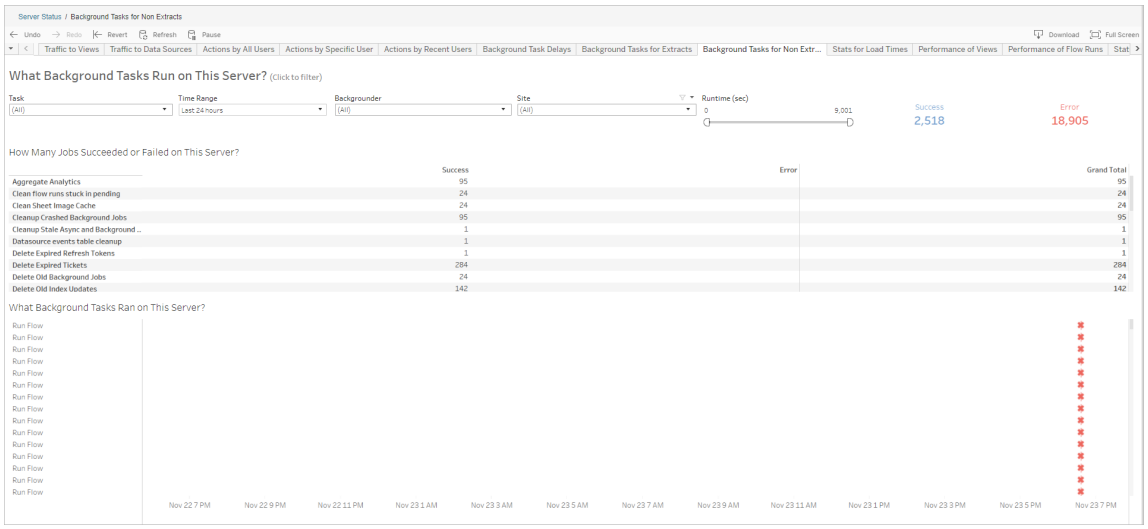
- 同時に実行されている他のサーバープロセスが、サーバー リソースを消費し、パフォーマンスを低下させています。

  サーバープロセスの CPU 使用量とメモリ使用量を監視して、最もリソースを消費しているプロセスを特定し、サーバー上のプロセスの設定を調節します。

  モニタリング プロセスの詳細については、「Windows Performance Monitor でデータを収集する」を参照してください。

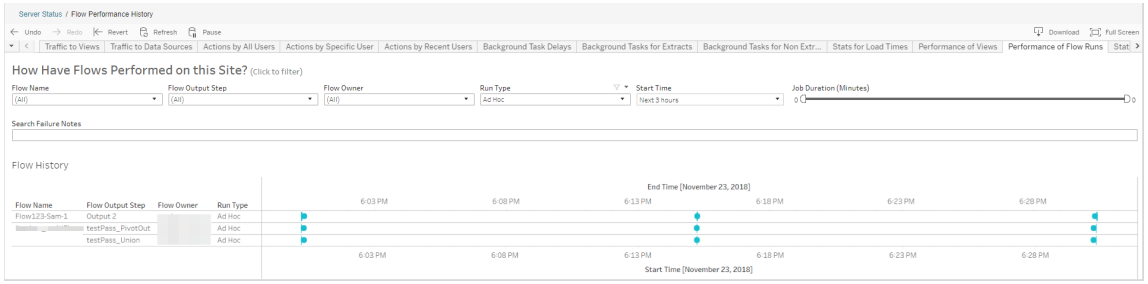
抽出以外のバックグラウンド タスク

バックグラウンド タスクは、スケジュールされたフローやアドホックのフローを実行するためにも作成されます。このビューを使用すると、サイト上で成功または失敗したフロー タスクの数を確認できます。タスクの詳細を確認するには、タスクのアイコンにカーソルを合わせてください。



フロー実行のパフォーマンス

このビューは、サイト上のすべてのフローのパフォーマンス履歴を表示するために使用します。フロー名、出力ステップ名、フロー所有者、実行タイプ(スケジュールまたはアドホック)、およびフロー実行の開始時刻でフィルターリングすることができます。



このビューを使用すると、次のような質問に答えることができます。

- **現在スケジュールされているフロー タスクは何か?** – これを確認するには、[開始時刻] フィルターを使用して、調べたい時間枠を選択します。たとえば、今後 3 時間でスケジュールされているフロー タスクを表示するには、**[時間] -> [今後]** を選択し、**「3」**と入力します。
- **フロー タスクの期間はどの程度か?** - これに回答するには、ビュー内のマークをクリックし、タスク実行期間を含む詳細を表示します。

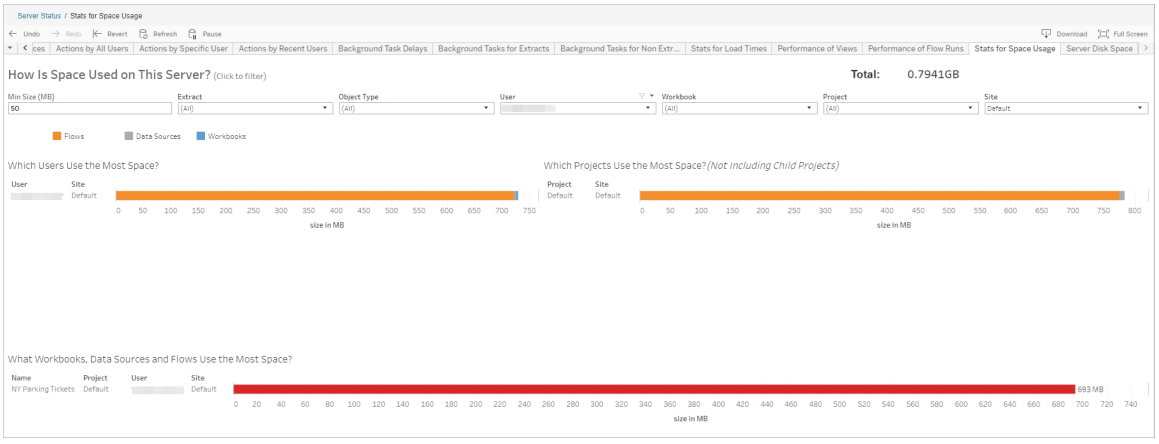
- **アドホックで実行されたフローの数、スケジュールされた実行の数はいくつか? -** これに回答するには、**[実行タイプ]** フィルターを使用し、**[アドホック]** または **[スケジュールされています]** を選択します。

このビューでは、次の情報も表示されます。

- 実行頻度が最も高いフローには、最も多くのマークが表示されます。
- 現在同時に実行されているフローを表示するには、**[進行中]** または **[保留中]** と表示されているマークにカーソルを合わせて **[保持のみ]** を選択すると、現在実行中のすべてのフローをフィルターにかけることができます。
- 特定の期間帯に同時に実行されているフローを確認するには、**[開始時間]** フィルターで範囲を選択します。たとえば、**[今後 3 時間]** を選択すると、今後 3 時間にどのフローが実行されるかを確認することができます。

領域使用量の統計

このビューを使用すると、サーバー上で最も多くのディスク容量を使用しているフロー出力を特定できます。ディスク容量の使用率は、ユーザー別、プロジェクト別、フロー出力のサイズ別に表示され、最も近い整数に切り捨てられます。



[最小サイズ] フィルターを使用して、占有するスペースの量に基づいて、どのフロー出力を表示するか制御します。フローにはオブジェクトタイプフィルターを使用します。

- **最もスペースを使用しているユーザーは誰か -** このセクションでは、(フローをフィルターにかけた時点で) 最も多くのスペースを消費しているフローの所有者を表示します。ユーザー名をクリックすると、次の 2 つのグラフをそのユーザーでフィルターできます。

- **最もスペースを使用しているプロジェクトは何か -** このセクションでは、(フローをフィルターにかけた時点で) 最も多くのスペースを消費しているフローを持つプロジェクトを表示します。

- **最もスペースを使用しているワークブック、データソース、フローは何か –** このセクションでは、(フローをフィルターにかけた時点で) 最も多くのスペースを消費しているフローを表示します。

実行可能なユーザー

- **Tableau Server 管理者**は、以下を実行できます。
  - サーバーレベルでメール通知を設定する
  - サイトのメール通知を設定する
  - エラーを表示する
  - 中断したタスクを再開する
  - アラートを表示する
  - プロセスのステータスを表示する
- **Tableau サイト管理者**は、以下を実行できます。
  - サイトレベルでメール通知を設定する
  - エラーを表示する
  - 中断したタスクを再開する
  - アラートを表示する
- **フローの所有者、プロジェクトリーダー、およびフローを表示する権限が付与されているすべてのユーザー**は、以下を実行できます。
  - エラーを表示する
  - 中断したタスクを再開する
  - アラートを表示する (フローの所有者)

開発者向けリソース - REST API

Tableau Server REST API を使用すると、Tableau を既存のワークフローに自動的およびシームレスに統合できます。Tableau Server REST API を使用すると、コンテンツ、ユーザー、サイト、さらに今ではフローにも、プログラムでアクセスして利用できます。HTTP を経由すると、Tableau Server または Tableau Cloud でプロビジョニング、パーミッション、パブリッシュを管理できます。REST API を使用すると、データソース、プロジェクト、ワークブック、サイトユーザー、サイト、およ

びフローの各機能にアクセスできます。このアクセスを使ってカスタム アプリケーションを作成したり、サーバー リソースとの相互作用をスクリプトしたりできます。

Tableau REST API は、次のフローの機能をサポートしています。

新しいフローのエンドポイントが追加されたことで、フローのパブリッシュ、フローのスケジュール設定、オンデマンドによるフローの実行、パーミッションの管理、フローのダウンロードなどがサポートされるようになりました。フローの新しい REST API エンドポイントの全リストについては、「フローのメソッド」を参照してください。

また、既存のエンドポイントが更新され、フローの新しいスケジュールの作成、新しいサイトの作成、既存のサイトの更新、および既定のパーミッションの管理などがサポートされるようになりました。

注意: REST API を使用してフローを実行する場合は、データ管理 ライセンスが必要です。

Tableau Catalog について

2025 年 10 月 14 日以降、「Data Cloud」は「Data 360」というブランド名に変わりました。この変更の途上では、アプリケーションやドキュメントに「Data Cloud」と表示されることがあります。名前は新しいですが、機能と内容は変更されていません。

データのボリュームや形式、重要性が増しているため、環境はさらに複雑になりつつあります。データが変化するペースが急速であることから、そのデータとそのような複雑な環境での使用方法を追跡するのが困難になることがあります。それと同時に、さまざまな場所にあるデータをより多く求めるユーザーが増えているため、ユーザーが適切なデータを見つけるのは難しくなっています。こうしたことから、ユーザーは適切なソースを使用しているか、それが最新かどうかを疑問に感じており、最終的にデータの信頼性が損なわれています。

Tableau Catalog は、系列、インパクト分析、データディクショナリ、データ品質に関する警告、検索などの機能を Tableau アプリケーションに統合し、これらの問題をスタンドアロンのカタログと異なった方法で解決するのに役立ちます。IT とエンドユーザーの両方に重点が置かれているため、Tableau Server または Tableau Cloud を使用しているすべてのユーザーはデータの可視性と信頼

性だけでなく、検出可能性も高めることができます。Tableau Catalog は組織で使用されている Tableau コンテンツからカタログを構築し、次のような包括的な機能を使用できるようにします。

- **インパクト分析と系列**。
  - 管理している表またはデータソースの特定の列やフィールドを利用するワークブックなどの Tableau コンテンツを表示できます。データを変更する必要がある場合は、影響を受ける Tableau の作成者にメールで通知できます。
  - ワークブック作成者は、系列を使用して、利用するフィールドを追跡できます。
  - ユーザーは Tableau のビジュアライゼーションを使用する場合、ビューの作成に使用されたデータの由来を表示することができます。
- **キュレーションと信頼**。データ案内人は、説明や証明書など有用なメタデータを追加し、ユーザーが適切なデータを見つけるようにすることができます。データ品質に関する警告の設定、[データの詳細] ペインでのデータ詳細の表示、資産の認証、カタログからの資産の削除を行うことができます。
- **データディスカバリ**。Tableau Desktop または Tableau Web 作成では、Tableau Catalog を使用して、Tableau で分析を行うデータベース、テーブル、データソース、および仮想接続を検索し、検索結果からそれらに接続することができます。

2019.3 以降、データ管理 は Tableau Server と Tableau Cloud の Tableau Catalog の一部として使用できます。プロダクトキーがアクティブで有効になっている場合、上記のカタログ機能は使用中の製品に統合されているため、見つけた場所でデータを操作することができます。

Tableau Catalog のしくみ

Tableau Catalog はサイト上のすべてのコンテンツ (ワークブック、データソース、シート、仮想接続、およびフロー) を検出してインデックスを作成し、コンテンツに関するメタデータを収集します。メタデータから外部アセット (データベース、テーブル、その他のオブジェクト) が特定されます。Tableau がコンテンツと外部アセットとの関係を理解すると、コンテンツと外部アセットの系列を表示できるようになります。また、Tableau Catalog を使用すると、ユーザーは Tableau Server または Tableau Cloud を使用して外部資産に接続できます。

サイトのユーザーは、コンテンツのパブリッシュや削除、データ品質に関する警告や認証の添付、サイト上のコンテンツやそのメタデータの変更などを行うことができ、Tableau Catalog はそれに応じて情報を更新します。

Tableau Catalog を使用して組織のデータガバナンスをサポートする方法については、Tableau Blueprint ヘルプの「Tableau のガバナンス」を参照してください。

Tableau Catalog の主要な用語

- メタデータ - データに関する情報。
- Tableau コンテンツ - Tableau で作成されたコンテンツ (ワークブック、データソース、仮想接続、およびフローなど)
- 外部アセット - Tableau Server または Tableau Cloud にパブリッシュされた、Tableau コンテンツで使用されるデータベース (データファイルを含む) およびテーブル (テーブルのようなオブジェクトを含む)。

Tableau Catalog のライセンス

Tableau Catalog は データ管理 を通じてライセンスされています。データ管理 のライセンスのしくみについては、「データ管理 のライセンス」を参照してください。

Tableau Catalog の有効化

Tableau Server または Tableau Cloud に データ管理 機能のライセンスが付与された後に、次のいずれかの作業を行うとTableau Catalog を有効にできます。

- **Tableau Cloud の場合**、アクションは必要ありません。Tableau Catalog は既定でオンで、派生 パーミッション を使用するように構成されており使用準備ができています。派生 パーミッション の詳細については、メタデータに対するパーミッション を参照してください。

- **Tableau Server の場合**、サーバー管理者はまず `tsm maintenance metadata-services` コマンドを使用し、Tableau メタデータ API を有効にする必要があります。詳細については、Tableau Catalog の有効化 を参照してください。

  メタデータ API を有効にした後、Tableau Catalog は既定でオンで、派生 パーミッション を使用するように構成されており使用準備ができています。派生 パーミッション の詳細については、メタデータに対するパーミッション を参照してください。

特長と機能

Tableau Catalog で使用できる機能の詳細については、次のヘルプ記事を参照してください。

データディスカバリ

- Tableau Desktop の **[接続]** ペインの **[データの検索]** で **[Tableau Server]** を選択し、Tableau Server または Tableau Cloud を使用してデータに接続します。Tableau Catalog が有効になっている場合は、接続先のパブリッシュ済みデータソースを検索するだけでなく、Tableau Server または Tableau Cloud サイト上のパブリッシュ済みのデータソースやワークブックで使用される特定のデータベース、テーブル、オブジェクトを検索し、それらに接続できるようになりました。
- Tableau Catalog を有効にすると、検索が拡張され、列、データベース、テーブル、その他のオブジェクトに基づいて検索結果が表示されるようになります。
- Web で作成する場合、パブリッシュされたデータソースに加えて、データベースやテーブルに接続することもできます。
- Tableau Prep を Web で使用する場合、データベースやテーブルなどの外部アセットに基づいて新しいフローを作成できます。
- Salesforce Data Cloud に接続すると、Tableau Catalog に組み込まれたネイティブ Data Cloud オブジェクトのサポートが表示されます。データレイクオブジェクト (DLO)、データモデルオブジェクト (DMO)、および計算されたインサイトは、検索、接続、系統のページに個別に表示されるため、それらの検出、接続、再利用が簡単になります。

キュレーションと信頼。

- データ資産を認証し、信頼でき推奨されるデータをユーザーが見つけるのを助けます。
- データ品質に関する警告を設定し、古いデータや廃止データなどデータ品質の問題に関してユーザーにアラートを出します。
- 機密度ラベルを追加して、取り扱いに注意が必要なデータについてユーザーに警告します。
- カスタム ラベルを追加して、組織のニーズに合った方法でデータを分類します。
- データ ラベルの管理 ユーザーが使用できるラベル名とカテゴリを拡張します。
- タグを使用して Tableau Server および Tableau Cloud でアイテムを分類すると、ユーザーは外部アセットをフィルターできます。
- [データの詳細] タブを使用してパブリッシュ済みのビジュアライゼーションについての理解を深め、使用されているデータの詳細を確認します。
- データベース、表、列に説明を追加して、ユーザーがデータを簡単に見つけられるようにします。

系列とインパクト分析

- 系列を使用してデータの出所を追跡し、データに対する変更の影響に関する分析や影響を受ける可能性があるユーザーの特定を行います。

- データ関連の更新について、ワークブック、データソース、またはフローの所有者にメールを送信します。データベース、テーブル、またはオブジェクトの連絡先にも同様のメールを送信します。

開発者向けリソース

Tableau REST API のメタデータ メソッドを使用すると、プログラムによって特定のメタデータを更新することができます。メタデータ メソッドの詳細については、Tableau Server REST API の「メタデータ メソッド」を参照してください。

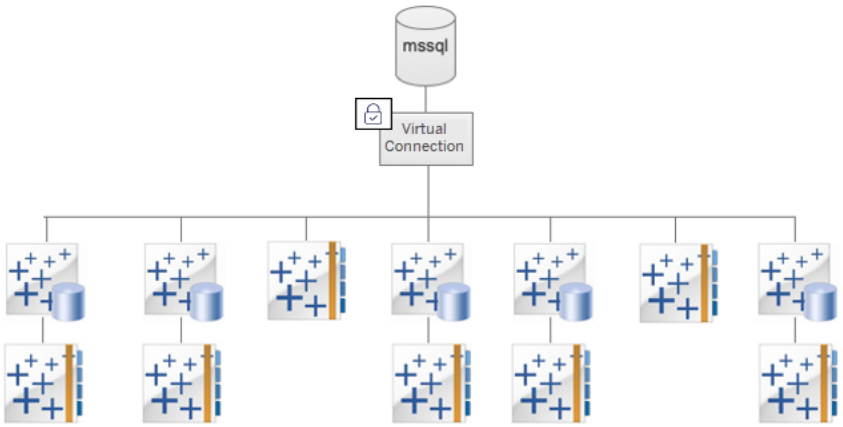
REST API に加えて、Tableau メタデータAPI を使用すると、Tableau Server または Tableau Cloud にパブリッシュされたコンテンツからプログラムによってメタデータに対するクエリを実行することができます。メタデータAPI は高速で柔軟性が高く、メタデータとその構造との関係について特定の情報を見つけようとしている場合に最適です。GraphiQL と呼ばれるインタラクティブなブラウザー内ツールを使用して、メタデータAPI に対するクエリを調査およびテストします。

注: メタデータAPI または GraphiQL の使用には データ管理 は必要はありません。

仮想接続とデータポリシーについて

仮想接続は、データソース、ワークブック、フローと並ぶ Tableau コンテンツ タイプの 1 つであり、データを表示し理解するのに役立ちます。仮想接続は、データへのアクセス ポイントを一元化するものです。仮想接続で導入されたもう1 つの重要な機能は、データ ポリシーです。データ ポリシーは、ワークブックやデータ ソースのレベルではなく、行 レベルのセキュリティを接続 レベルで提供します。行 レベルのセキュリティのためのデータ ポリシーは、仮想接続を使用するすべてのワークブック、データ ソース、フローに適用されます。

仮想接続は、複数のデータベースにまたがる複数のテーブルにアクセスすることができます。仮想接続では、データの抽出とセキュリティを接続 レベルで一元管理できます。



行レベルのセキュリティオプションについては、Tableau の「Tableau の行レベルのセキュリティオプションの概要」を参照してください。

すべての仮想接続に、関連するデータポリシーがあるわけではありません。また、仮想接続は、単に接続の認証資格情報を管理するための一元的な場所として使用することもできます。

重要な用語

- 仮想接続。データへのアクセスポイントを一元化する共有可能なリソースです。
- 接続データへのアクセスに使用するサーバー名、データベース、認証資格情報仮想接続には、1 つまたは複数の接続が含まれます。各接続は、1 つのデータベースまたはファイルにアクセスします。
- 仮想接続テーブル。仮想接続内のテーブルです。
- データポリシー。ユーザーのデータをフィルターリングするために仮想接続の 1 つまたは複数のテーブルに適用されるポリシーです。たとえば、データポリシーを使用して、仮想接続のテーブルに行レベルのセキュリティを適用します。
- ポリシー テーブル。フィルターリングされたデータポリシーのファクトまたはデータテーブルです。
- ポリシー列。ポリシー テーブルのデータをフィルターリングするために使用する列。ポリシー列は、ポリシー テーブルまたは資格テーブルに含めることができます。
- 資格テーブル。ポリシー テーブルのフィルターリングに使用できるポリシー列と、ポリシー テーブルの列に関連付ける(マッピングする)ことができる別の列の両方を含むテーブル。
- ポリシー条件。クエリ時に各行に対して評価される式または計算。ポリシー条件が TRUE のとき、その行はクエリに表示されます。

仮想接続とデータポリシーのライセンス

仮想接続とデータポリシーは、データ管理 を介してライセンスされます。データ管理 のライセンスのしくみについては、「データ管理 のライセンス」を参照してください。

仮想接続とデータポリシーの有効化

仮想接続とデータポリシーは、データ管理 を使用して Tableau Server と Tableau Cloud で自動的に有効になります。

パーミッション

仮想接続のパーミッションは、他の Tableau コンテンツのパーミッションとほぼ同様に機能します。仮想接続をパブリッシュすると、誰でもその接続を表示できるようになります。ただし、接続の作成者がそれ以上のパーミッションを明示的に付与するまで、その接続を使用してデータにアクセスできるのは接続の作成者と管理者に限られます。

仮想接続を作成するときは、他のユーザーが仮想接続を使用してデータに接続できるように、接続機能のパーミッションを設定する必要があります。Connect 機能により、仮想接続の共有が可能になるため、ユーザーはその接続に関するクエリを実行することができます。接続権限を持つユーザーは、仮想接続でテーブルを表示し、そのテーブルを使用したコンテンツを作成することができます。詳細については、「仮想接続でのパーミッションの設定」を参照してください。

パーミッションとデータポリシー

パーミッションは、Tableau のコンテンツでユーザーが実行できる操作と実行できない操作を定義します。パーミッションは、コンテンツの表示、Web 編集、データソースのダウンロード、コンテンツの削除などを実行する機能で構成されています。パーミッション ルールは、コンテンツでユーザーまたはグループに対して許可または拒否される機能を定義します。ライセンス レベル、サイト ロール、および複数のパーミッション ルール間での相互作用には、最終的に決定されたユーザーが実行できる操作とできない操作、つまり、有効なパーミッションが反映されています。詳細については、「パーミッション」を参照してください。

データポリシーは、仮想接続でデータをフィルターし、表示されるべきデータのみがユーザーに表示されるようにします。データポリシーを適用すると、データは Tableau コンテンツ (ワークブックやフロー

など) で表示するときにフィルターされます。データポリシーのポリシー条件は、データへのアクセスを定義する計算または式です。ユーザー関数は、多くの場合、ユーザーまたはグループへのアクセスを制限するために使用されます。アクセスは、ユーザー名、ユーザーが属するグループ、または地域の値に基づいて行うことができます。詳細については、「行レベルのセキュリティに使用するデータポリシーの作成」を参照してください。

アクセスには、パーミッションとデータポリシーの両方が適用されます。簡単に言えば、パーミッションが、表示、アクセス、使用、または作成できるコンテンツを決定し、データポリシーが、表示できるデータを決定します。

パーミッションとデータポリシーが連携するしくみ

Tableau のパーミッションは、最初に Tableau コンテンツに適用されます。ユーザーは Tableau コンテンツで実行できる機能しか実行できません。データポリシーは Tableau のパーミッションを上書きしません。パーミッションが評価された後に、データポリシーが適用され、ポリシー条件に基づいて、仮想接続内で表示できるデータが決定されます。

次の例では、給与データを含む仮想接続に対するパーミッションとデータポリシーの影響について説明します。

- 仮想接続は HR プロジェクト内にあり、HR グループの Tableau ユーザーに制限されています。HR グループ外のユーザーは、HR プロジェクトのコンテンツを表示できません。つまり、仮想接続を参照、接続、または表示することはできません。
- 仮想接続には、HR ビジネス パートナー グループのメンバーにのみ付与される接続パーミッションがあります。HR グループ内の他のすべてのユーザーは、仮想接続が存在することを確認できますが、仮想接続に含まれるデータを表示することはできません。その仮想接続を使用するワークブックを表示しても、データは表示されません。
- 仮想接続には、個々のユーザーに基づいて給与データをフィルターするデータポリシーも含まれているため、HR ビジネス パートナーは、ビジネス ユニット内の従業員に関連する行のみを表示できます。その仮想接続を使用するワークブックを表示すると、ビジネス ユニットのデータのみが表示されます。

特長と機能

データの管理者にとっての仮想接続の利点は以下のとおりです。

- **安全に管理されたサービス アカウント。**「サービス アカウント」モデルを使用する場合、そのデータにアクセスしたいユーザーとサービス アカウントの情報を共有することなく、仮想接続の作成権限を与えられた少数のアナリストに限ってサービス アカウントの認証資格情報を付与できるようになります。
- **アジャイルな物理データベース管理。**データベースの変更 (たとえば、フィールドの追加やテーブル名の変更) は、データを使用するコンテンツごとに行うのではなく、仮想接続で 1 回だけ行う必要があります。
- **データの増殖を抑制。**抽出更新のスケジュールを一元管理することにより、更新をスケジュールするのは 1 回だけとなり、その仮想接続からデータにアクセスするすべてのユーザーに新しいデータが表示されるようになります。
- **一元管理された行レベルのセキュリティ。**行レベルのセキュリティを接続レベルで適用するデータ ポリシーを、Tableau 抽出とライブ クエリの両方に作成できます。データ ポリシーは、仮想接続を使用するすべてのワークブック、データ ソース、またはフローに適用されます。

> **注:**データ ポリシーはフロー入力データには有効ですが、フロー出力データには有効ではありません。フロー出力データにアクセスできるユーザーには、自分に関係のあるデータのサブセットだけではなく、すべてのデータが表示されます。

データのユーザーは、仮想接続を使用すると次の恩恵を受けることができます。

- 行レベルのセキュリティがすでにデータに適用されているため、見るべきデータのみが表示される**適切なアクセス。**
- 整理され保護されたデータを使用できる**柔軟性。**仮想接続は、接続情報を保存し、共有します。必要なのは、ニーズに特化したデータ モデルを持つデータ ソースを作成することです。
- 抽出更新のスケジュールがすでに設定されているため、データが新鮮である**信頼性。**
- データ ポリシーが常に適用されるため、セキュリティを危険にさらすことなく、コンテンツを自由に**共有**できる機能。

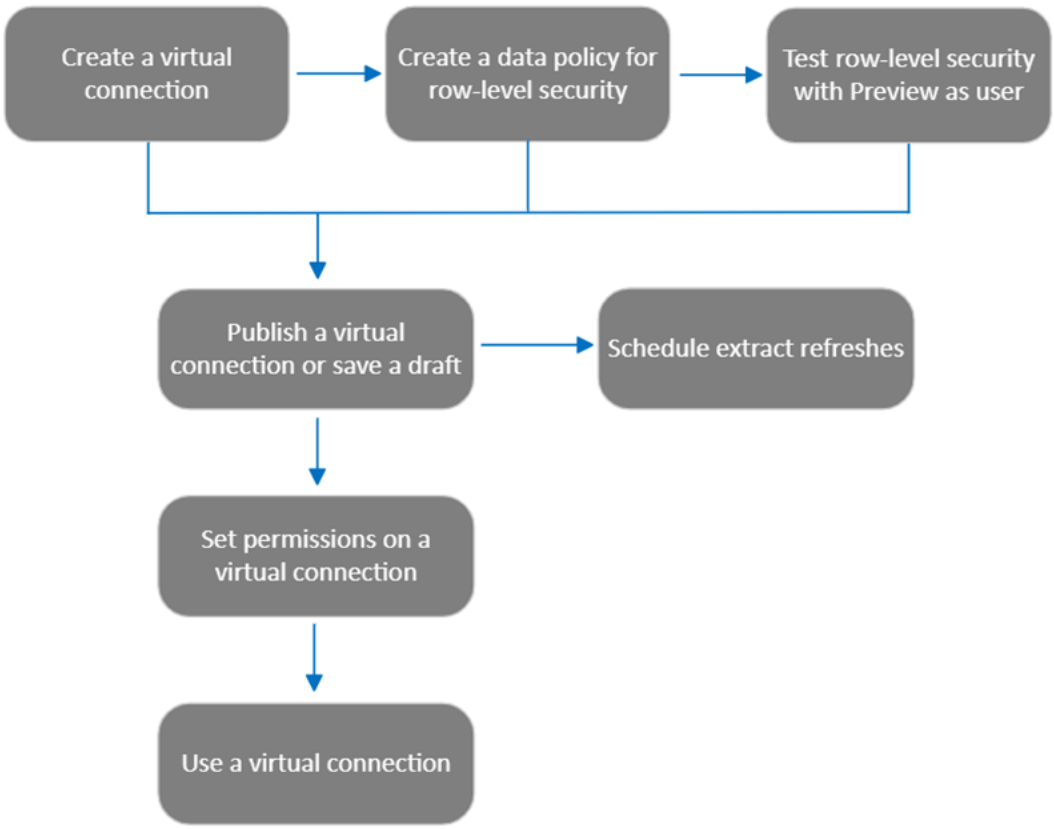
仮想接続エディターのワークフロー

仮想接続エディターを使用すると、次を作成できます。

- 仮想接続。データへの一元化されたアクセス ポイントを共有できる、Tableau コンテンツ タイプの 1 つです。
- データ ポリシー。行レベルのセキュリティを接続レベルでサポートします。

仮想接続とそれに関連するデータポリシーを作成したら、それをパブリッシュして、他のユーザーと共有するためのパーミッションを設定できます。抽出更新をスケジュールすると、仮想接続を使用するすべてのコンテンツが新しいデータにアクセスできるようになります。

次の図は、仮想接続を作成するためのワークフローを示しています。プロセス中はいつでも接続のドラフトをパブリッシュまたは保存できますが、抽出更新をスケジュールしたり、仮想接続を使用(または編集)したりするには、事前に接続をパブリッシュする必要があります。また、他の人が接続を使用できるようにするには、事前にパーミッションを設定する必要があります。



プロセスのステップをクリックして、そのヘルプトピックに移動します。

次のステップ

最初のステップは、「仮想接続の作成」です。

仮想接続の作成

仮想接続は、データへの共有可能な中央アクセス ポイントを提供し、接続レベルで行レベルのセキュリティをサポートする Tableau コンテンツ タイプです。仮想接続は複数ステップのプロセスで作成します。このトピックでは、共有するデータへの接続と、仮想接続エディターの [テーブル] タブでの作業について説明します。

データへの接続

Tableau Cloud または Tableau Server で仮想接続を作成するには、次の手順を実行します。

1. [ホーム] または [探索] ページで、**[新規] > [仮想接続]** をクリックします。
2. [データに接続] ダイアログ ボックスで、データのコネクタを選択します。仮想接続でサポートされているコネクタのリストについては、Tableau Desktop および Web 作成ヘルプの「Creators: データへの接続」を参照してください。
3. 要求された情報を入力します。入力した認証資格情報は仮想接続に保存されるため、接続ユーザーは、データに接続するために認証資格情報を入力する必要はありません。
4. プロンプトが表示されたら、**[サインイン]** をクリックします。別の接続を追加するには、 をクリックしてコネクタを選択し、認証資格情報を入力してサインインします。

1 つの仮想接続は複数の接続を持つことができます。各接続は、1 つのデータベースまたはファイルにアクセスします。

注: Tableau Cloud の場合、プライベート ネットワーク データに接続する仮想接続では、Tableau Bridge を使用して、データを最新に保ちます。Tableau Bridge の構成については、「Bridge クライアント プールの構成と管理」を参照してください。サポートされる接続については、「Bridge を使用した接続」を参照してください。

別の接続の追加

必要に応じて、仮想接続に別の接続を追加し、**[接続]** の横にある をクリックして複数のデータベースに接続します。別のサーバーやデータベース、または同じサーバーやデータベースへの接続を追加できます。

複数の接続では、次のことができます。

- 他の接続やデータベースからテーブルを保護するデータポリシーで、任意の接続またはデータベースのテーブルを資格テーブルとして使用できます。
- 仮想接続のテーブルを別のデータベースのテーブルに追加または置換できます。たとえば、あるデータベースから別のデータベースにデータを移行するとします。仮想接続エディターで、2 番目のデータベースへの接続を追加し、最初のデータベースの既存のテーブルを 2 番目のデータベースのテーブルに置き換えることができます。
- 同じサーバーまたはデータベースに複数の接続を追加できます。これは、たとえば、同じデータベースのデータに異なる認証資格情報でアクセスする必要がある場合に役立ちます。
- 関連性のある、または一緒に使用する予定のあるテーブルの集まりを、それらが物理的にどこにあるかに関わらず共有できます。たとえば、複数のデータベースから、従業員の情報に関連するテーブルをグループ化することができます。

仮想接続を開いて編集する場合、プロンプトが表示されたら、接続を順番に認証する必要があります。いずれかの接続が認証に失敗すると、仮想接続を編集できません。

接続に含めるテーブルの選択

必要に応じて、データベースを選択してその中のテーブルを表示します。

1. 左側の **[テーブル]** でテーブルを選択してクリックするか、右側の [テーブル] タブにドラッグします。さまざまな接続からのテーブルを含めることができます。資格テーブルを使用している場合は、それを含めます。
2. (オプション) **[新しいカスタム SQL]** をクリックして、カスタム テーブル スキーマを作成します。

注: 仮想接続は、空間データ型のテーブルをサポートしていません。

テーブルのライブ モードと抽出モードの選択

複数の接続からのテーブルかどうかにかかわらず、同じ仮想接続の中で個々のテーブルをライブモードまたは抽出モードに設定することができます。

- **ライブ -** テーブルはデータベースから直接クエリされます。(ライブがデフォルトです)。
- **抽出 -** テーブルは抽出され、Tableau に保存されます。

たとえば、一部のテーブルを抽出モードに設定することで、レポート生成や大量の顧客トラフィックに影響されないようにすることができます。

[テーブル] で、モードを変更したいテーブルを複数選択し、**[アクション]**、**[Change to Live (ライブに変更)]**、または **[Change to Extract (抽出に変更)]** を選択します。または、テーブルの行にある [アクション メニュー] (...) を選択し、**[ライブ]** または **[抽出]** を選択します。

増分抽出

2024 年 6 月の Tableau Cloud および Tableau Server 2024.2 以降、増分更新用にテーブル抽出を設定できます。

増分抽出を設定するときは、新しい行の識別に使用するキー列を指定します。増分抽出が更新されると、キー列が増加した行のみが抽出に追加されます。処理される行数が少ないほど、抽出更新ジョブが高速化し、データベースへの負荷が軽減されます。

たとえば、**「打者」**テーブルを抽出し、抽出内のデータが次のとおりであるとします。

年	チーム
1978	ライオンズ
1979	タイガース

「打者」テーブルは増分抽出更新用に設定されており、キー列は**「年」**です。

ライブ テーブルは**「1980」**年の新しい行で更新されます。

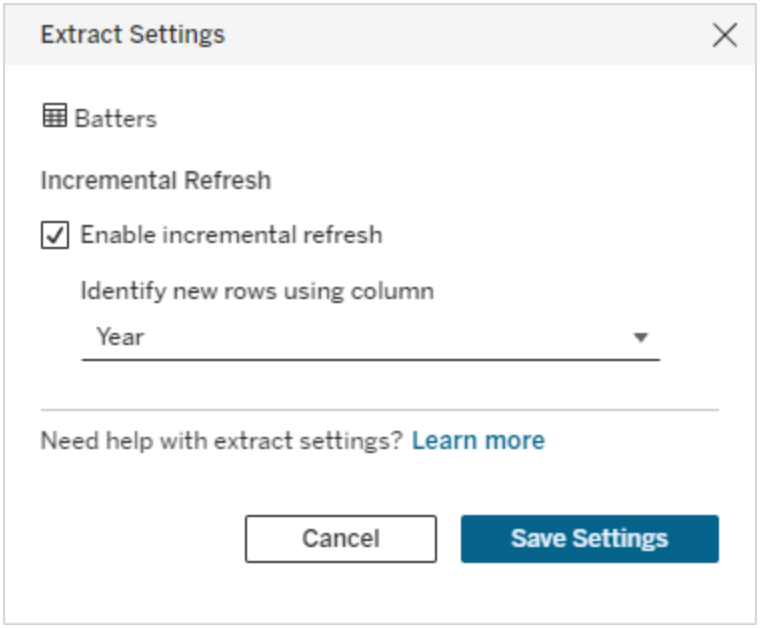
年	チーム
1978	ライオンズ
1979	タイガース
1980	ベアーズ

「打者」テーブルの抽出が増分的に更新されると、抽出のキー列の最大値を超える行のみが追加されます。この場合、**「1980」**年の行が抽出に追加されます。抽出ファイル全体を更新するのではなく、1 行のみが処理され、追加されます。

抽出全体を更新したい場合は、増分更新用に構成された抽出に対して完全更新を実行することもできます。

テーブル抽出の増分更新を設定するには、次の操作を行います。

1. テーブルを**[ライブ]**から**[抽出]**に変更します。
2. テーブルの行にある[アクション メニュー] (...) を選択し、**[抽出の設定...]**を選択します。
3. **[増分更新を有効にする]**をチェックします。
4. 追加する行を決定するときに使用する増分列を選択します。
5. **[設定を保存]**を選択します。



カスタム SQL への変換

2024 年 6 月の Tableau Cloud および Tableau Server 2024.2 以降、テーブルへの接続に使用される SQL を近似し、それを独自のカスタム SQL のひな型として使用できるようになります。カスタム SQL を使用すると、フィルタリングやその他のクエリ変更を行い、異なる結果を得ることができます。データペインで**[新しいカスタム SQL]**を使用せずに、この方法でカスタム SQL を作成すると、既存の仮想接続への影響が少なくなります。下流のアセットは、このテーブルを新しいテーブルではなく同じテーブルとして認識します。

テーブルをカスタム SQL に変換するには、次の操作を行います。

1. テーブルの行で [アクション メニュー] (...) を選択します。
2. **[カスタム SQL に変換]** を選択します。
3. **[カスタム SQL の編集]** ダイアログで、必要に応じて SQL を編集します。
4. **[テーブルを生成]** を選択します。

> 注: ダイアログに最初に表示される SQL はひな型として考えるべきであり、編集しなければ機能しない可能性があります。仮想接続エディターには、接続で使用される特定の SQL 構文に関する詳細な情報が欠けています。[テーブルの生成] ボタンを選択したときにエラーが発生した場合は、一重引用符、二重引用符、逆引用符、角括弧を削除または編集して、使用しているデータベースに SQL が準拠するようにしてください。

カスタム SQL を編集するには、次の操作を行います。

1. テーブルの行で [アクション メニュー] (...) を選択します。
2. **[カスタム SQL の編集]** を選択します。
3. **「カスタム SQL の編集」** ダイアログで、SQL を編集します。
4. **[テーブルを生成]** を選択します。

カスタム SQL を使用せずにテーブルをデフォルトの状態に戻すには、次の操作を行います。

1. テーブルの行で [アクション メニュー] (...) を選択します。
2. **[置換]** を選択します。
3. テーブルを置換するダイアログで、元のテーブル名を選択します。
4. **[OK]** を選択します。

テーブル データの抽出

1 つまたは複数のテーブルがライブから抽出に変更されたが、まだ抽出されていない場合は、**[保留中の抽出を作成]** (Tableau Server 2023.3 以前では **[保留中の抽出を実行]**) をクリックして保留中の抽出を実行します。保留中の抽出がすべて実行されたら、**[すべての抽出を更新]** (Tableau Server 2023.3 以前では **[すべての抽出を実行]**) をクリックして、その時点のすべてのテーブル データを抽出します。

または、2024 年 6 月の Tableau Cloud または Tableau Server 2024.2 以降では、テーブルの行で [アクション メニュー] (...) を選択し、**[抽出の更新...]** を選択します。テーブルに増分更新が設定されていない場合は、**[更新 (完全)]** のみを選択できます。増分更新が設定されている場合は、抽

出を完全に更新する**[更新 (完全)]**、または抽出を増分更新する**[更新 (増分)]**のいずれかを選択できます。

保留中の抽出がある場合は、仮想接続をパブリッシュする前に実行する必要があります。抽出が生成されている間は、接続を編集することはできません。

接続をパブリッシュしたら、仮想接続ページで仮想接続内のテーブルの抽出更新のスケジュールを設定します。「仮想接続の抽出更新のスケジュールを設定する」を参照してください。

テーブルの表示状態の設定

[テーブル] タブの表示切替トグルを使用して、テーブルとそのデータをユーザーに表示または非表示にします。

ユーザーはテーブル データを表示できます。どのデータをユーザーに表示するかを管理するデータ ポリシーを作成できます。(表示がデフォルトです)。

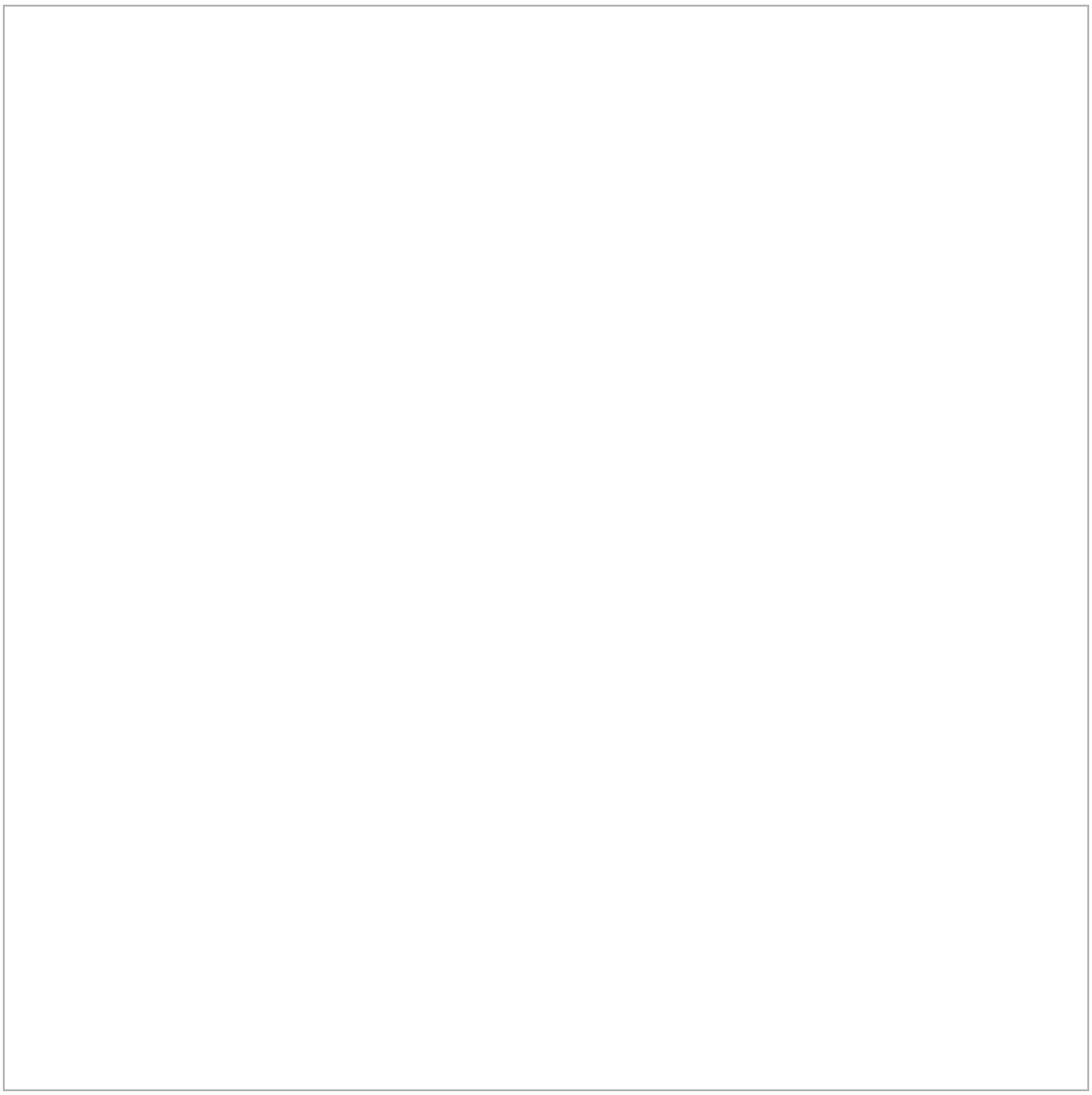
ユーザーはテーブル データを表示できません。非表示のテーブルはデータ ポリシーで使用でき、資格テーブルとして使用できます。

テーブルの詳細の確認

[テーブル] タブの上部にあるテーブルをクリックすると、その詳細が表示されます。[テーブルの詳細] セクションで、テーブル名の変更、列の非表示または名前変更、データ型の変更など、簡単な編集を行うことができます。

次のアイコンを使用して、表示されるテーブル情報を切り替えます。

- テーブル内の列のリストと各列のデータ型。
- 各列のサンプル データと、可能な場合はリンクされたキー。リンクされたキーは、どの列が他のテーブルにリンクしているかを示します。リンクされたキーは、データベースに主キーと外部キーの情報がある場合にのみ表示されます。
- 選択した各列のヒストグラムの値の範囲。



データベースからデータを更新する

ツールバーの更新アイコン⟳をクリックして、仮想接続内のすべての接続についてデータベースから最新のデータを取得します。これには以下が含まれます。

- データベース、テーブル、および列のリスト。仮想接続に含まれるテーブルと含まれないテーブルの両方が更新されます。
- テーブルとヒストグラムのデータ。

ライブ モードのテーブルの場合、更新すると、データベース、テーブル、および列の最新リストと、最近のテーブルおよびヒストグラム データが取得されます。抽出モードのテーブルの場合、更新すると、テーブルと列の更新されたリストが取得されます。ただし、最近のテーブルとヒストグラム データを表示するには、新しい抽出を開始する必要があります。たとえば、データベース テーブルに新しい列があり、更新アイコンをクリックすると、新しい列がエディターに表示されますが、そのデータは表示されません。最新のデータを表示するには、新しい抽出を開始する必要があります。

データを更新すると、現在キャッシュされているデータが無効になります。エディターを閉じて再度開いたり、抽出モードからライブ モードにテーブルを切り替えたり、ユーザー名やパスワードなどの接続の認証資格情報を変更しても、データが更新されます。

実行可能なユーザー

仮想接続を作成するには、次の認証資格情報とロールを持っている必要があります。

- 仮想接続が接続するデータベースにサインインするための認証資格情報
- サーバー管理者、サイト管理者、または Creator

次のステップ

[テーブル] タブでテーブルを追加して構成した後、行レベルのセキュリティに使用するデータ ポリシーの作成するか、仮想接続のパブリッシュとパーミッションの設定を行うかを選択できます。

関連項目

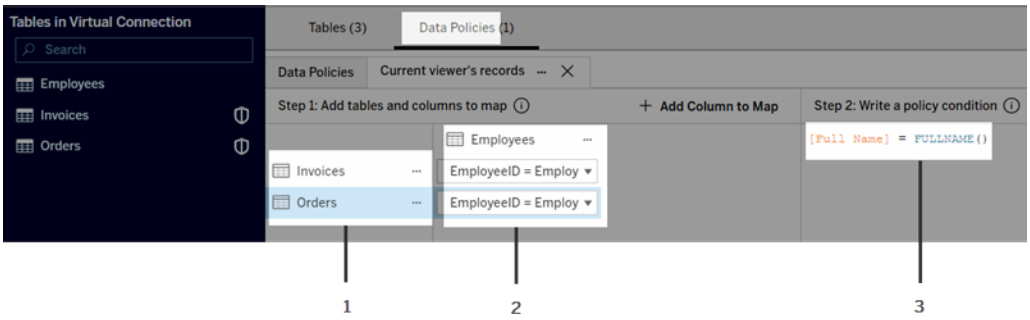
プロパティファイルを使用して JDBC 接続をカスタマイズする — JDBC ベースの接続をカスタマイズする場合は、プロパティファイルでカスタマイズすることもできます。

行レベルのセキュリティに使用するデータ ポリシーの作成

データ ポリシーを使用すると、仮想接続内の 1 つまたは複数のテーブルに行レベルのセキュリティを適用できます。データ ポリシーは、データをフィルターリングし、ユーザーに表示されるべきデータのみが表示されるようにします。データ ポリシーは、ライブ接続と抽出接続の両方に適用されます。

データ ポリシーについて

データ ポリシーには、3 つの主要な要素があります。



1. ポリシー テーブルと呼ばれる、適用対象のテーブル。これらはフィルターされるテーブルです。
2. マッピングされた列。テーブル間の関係 (たとえば、資格テーブルとファクト テーブル間の関係)、およびテーブル列とポリシー列間の関係を定義するものです。ポリシー列は、データのフィルターに使用される列です。
3. ポリシー条件。クエリの実行時にすべての行に対して評価される式または計算です。ポリシー条件が TRUE のとき、その行はクエリに表示されます。

データ ポリシーを作成するときは、データのフィルターリングに使用できる列が必要です。この列はポリシー列と呼ばれます。データは、通常 USERNAME() や FULLNAME() などのユーザー関数を使用して、ポリシー条件でフィルターリングします。

ポリシー テーブルにフィルターリング可能な列が含まれている場合は、その列をポリシー列として使用します。

ポリシー テーブルにそのような列が含まれていない場合は、データのフィルターリングに使用できる列を持つ資格テーブルを使用します。資格テーブルには、ポリシー テーブルのフィルターリングに使用できるポリシー列と、ポリシー テーブルの列に関連付ける (マップする) ことができる別の列の両方が含まれます (上のデータ ポリシーの例の画像を参照)。

ポリシー テーブルのポリシー列でフィルターリングする

データをフィルターリングする最も一般的な方法は、フィルターリングするデータを含むテーブルの列を使用することです。その列をポリシー列として使用してから、適切なテーブル列をポリシー列にマップします。

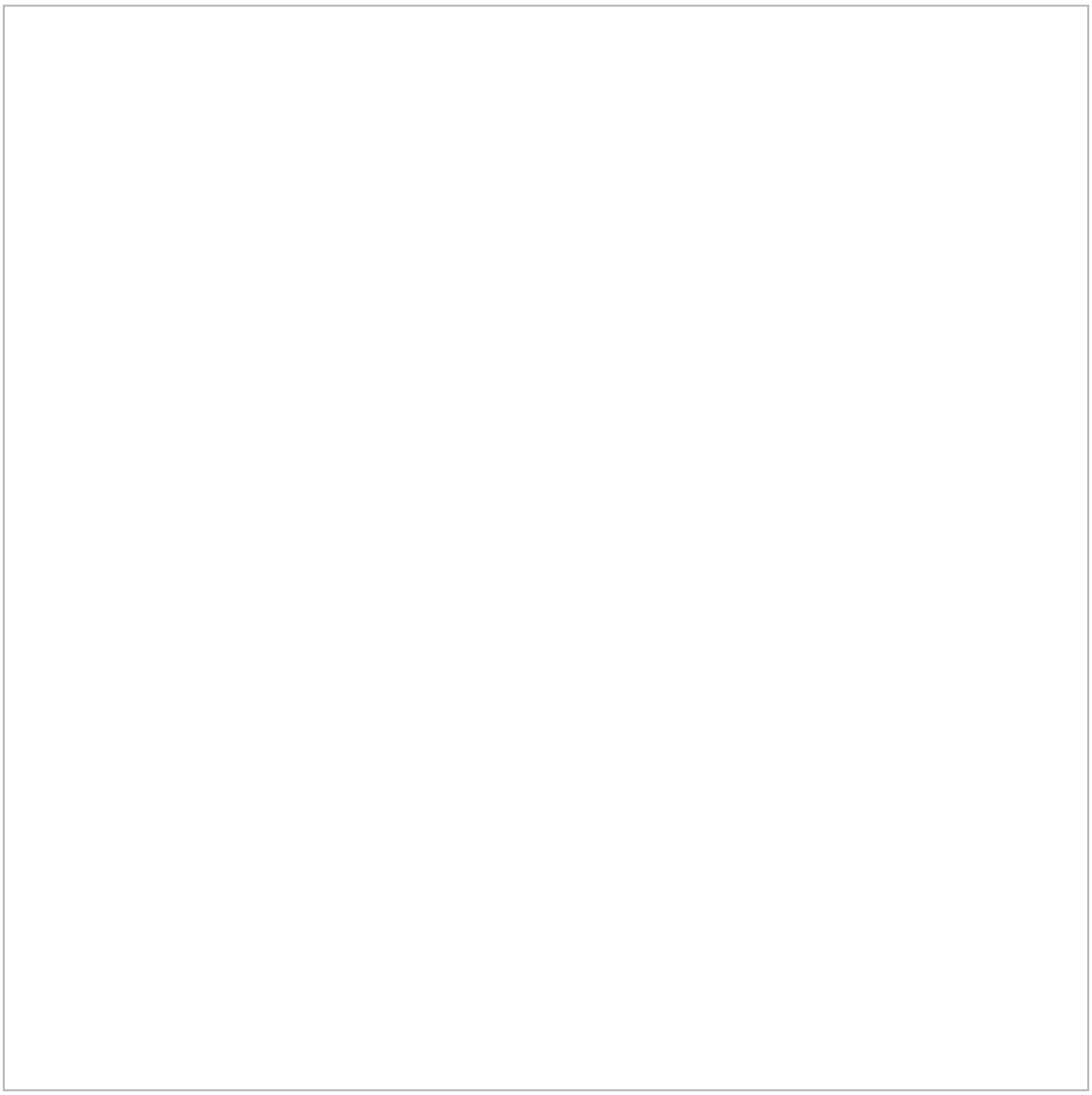
ポリシー列を使用してデータをフィルターリングするには、まず、左側のペインからポリシーにテーブルを追加します。テーブルを追加するには、次のいずれかの操作を行います。

- テーブル名をダブルクリックします。
- テーブル名の近くにあるドロップダウン矢印をクリックして、**[Manage table with policy (ポリシーを使用してテーブルを管理)]** を選択します。
- または、テーブルを右にドラッグして、**[ポリシー テーブルとして追加]** にドロップします。

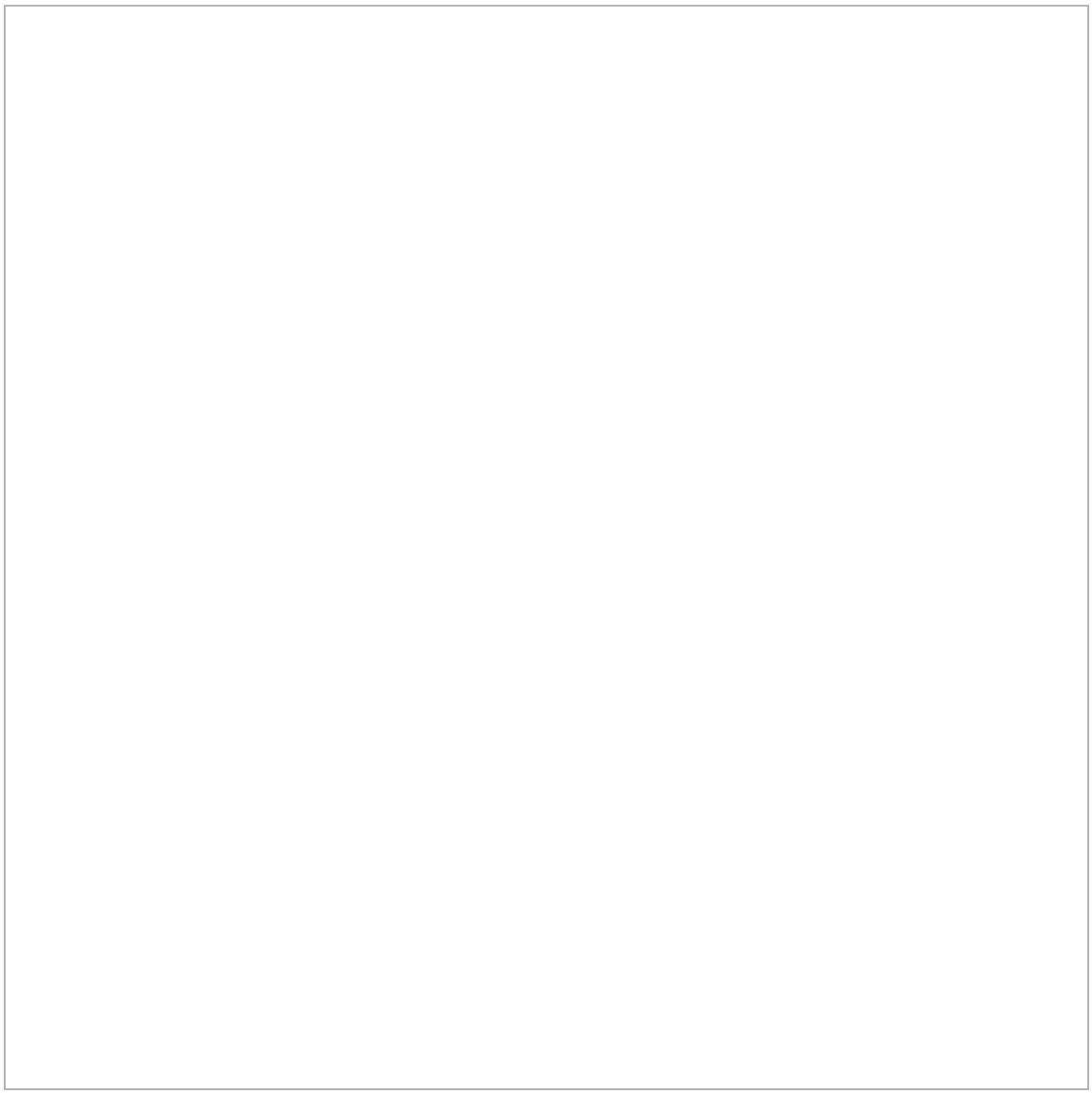
ポリシーにテーブルが追加されると、シールドアイコン が左側のペインのテーブル名の右側に表示され、ポリシー テーブルであることが示されます。

次に、列をマッピングして、テーブルの列名とポリシー列名の間にリレーションシップを作成します。データ ポリシー条件のポリシー列名を使用して、ユーザーがデータにアクセスできる権限を行レベルで制御します。

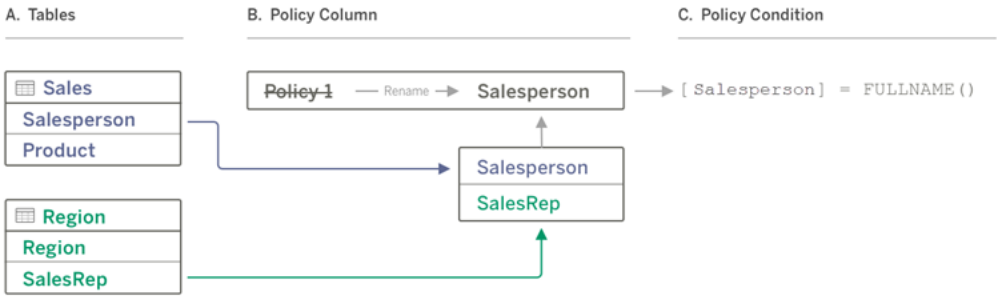
1. **[+マッピングする列を追加]** をクリックして、データのフィルター処理に使用する 1 つまたは複数の列を追加します。
2. ポリシー列に名前を付けます。この名前はポリシー条件で使用します。
3. ポリシーを適用するテーブルごとに、ドロップダウン メニューを使用して、ポリシー列にマッピングするテーブル列を選択します。
4. ポリシー条件で使用するポリシー列の数だけこのプロセスを繰り返します。



ヒント: [+マッピングする列を追加] ボタンを使用する代わりに、ポリシー条件領域に計算の入力を開始し、オートコンプリートを使用して列名を選択すると、手順 1 でポリシー列の情報が入力されます。



ポリシー テーブルのポリシー列を使用した例



A. "営業" テーブルには [セールスパーソン] 列があり、"地域" テーブルには [営業担当] 列があります。セールスパーソンと営業担当のデータは、サイト上の Tableau ユーザーのフルネームと一致します。
B. "セールスパーソン" で "営業" と "地域" のデータをフィルターリングしたいので、ポリシー列に "セールスパーソン" という名前を付けてから、"営業" の "セールスパーソン" 列と "地域" の "営業担当" 列を "セールスパーソン" ポリシー列にマッピングします。
C. 次に、両方のテーブルをフィルターリングするためのポリシー条件を記述します。[セールスパーソン] ポリシー列と FULLNAME() ユーザー関数を使用して、各ユーザーが自分のデータのみを表示できるようにします。

資格テーブルのポリシー列でフィルターリングする

資格テーブルは、ポリシー テーブルにフィルターリングできる列が含まれていない場合に使用します。資格テーブルを使用すると、データ テーブルの列を資格テーブルの列にマッピングできます。次の点に注意してください。

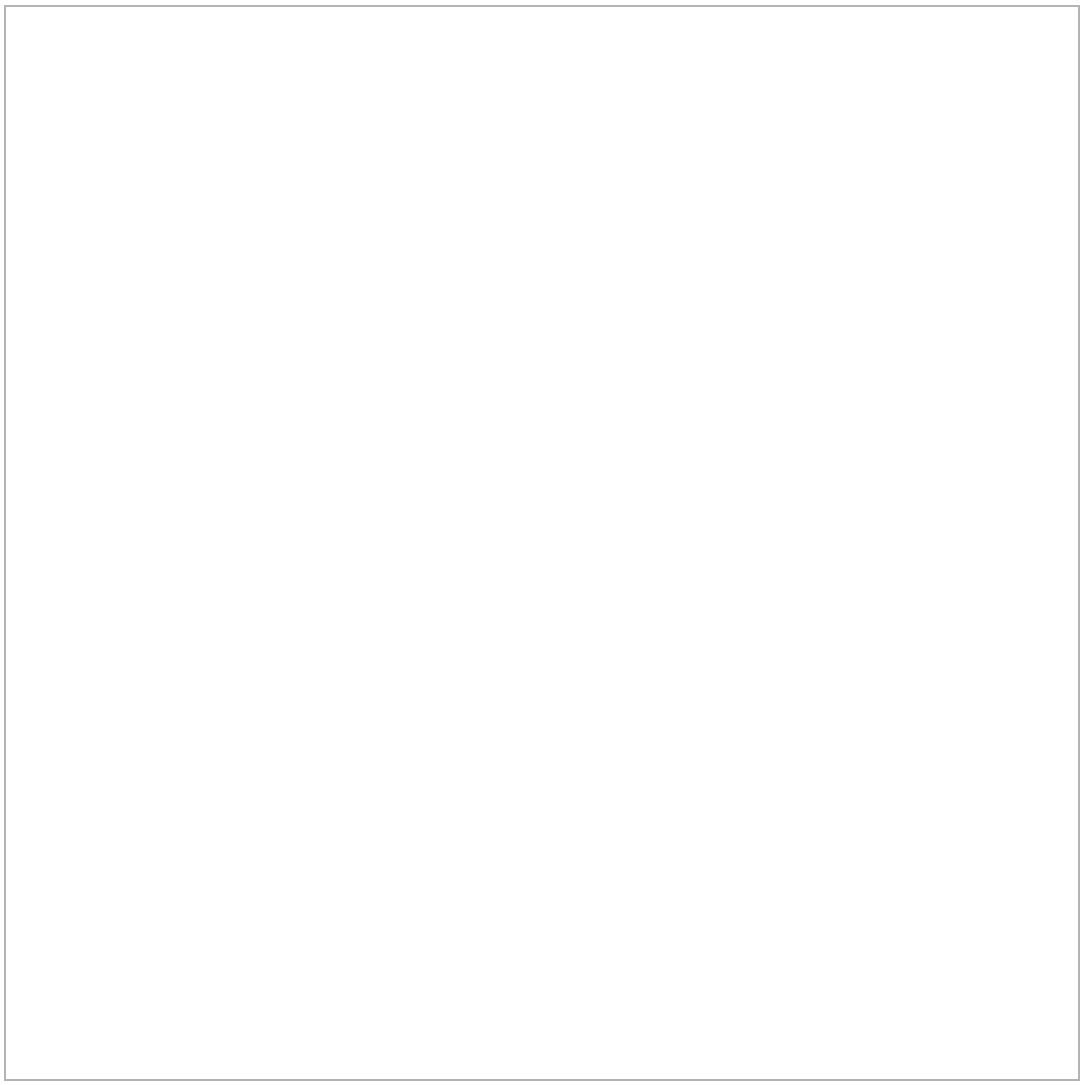
- 資格テーブルは、仮想接続のテーブルとして必ず含めてください。接続またはデータベースのテーブルを、他の多くのデータベースのテーブルを保護する中央の資格テーブルとして使用できます。場合によっては、保護しているテーブルと同じデータベースにある資格テーブルにより従業員データが公開される可能性があるため、その資格テーブルが潜在的なセキュリティリスクになる可能性があります。また、資格テーブルを別のデータベースに配置すると、たとえば、データベースへのアクセス権の付与など、パーミッションの制御が容易になります。
- 仮想接続ユーザーに資格テーブルを表示させたくない場合は、[テーブル] タブの [表示] 列の設定を切り替えて非表示にすることができます。非表示にした資格テーブルは、引き続きポリシー フィルタリングに使用できますが、Viz やワークブックのデータ ソースでは使用できません。

注: 資格テーブルでは、フロー出力 (.hyper ファイル) への直接接続はサポートされていません。フロー出力は、データベースに直接書き込む必要があります。

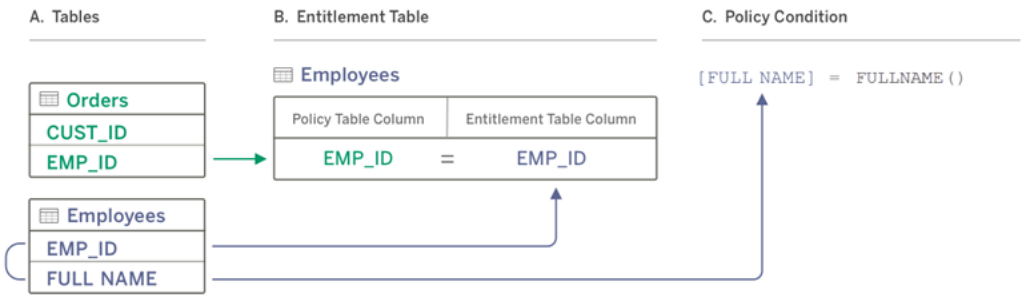
資格テーブルを使用してデータをフィルターリングするには、次の操作を行います。

1. まず、データポリシーを適用したいデータテーブルを追加します。次のいずれかを実行します。
    - テーブル名をダブルクリックします。
    - テーブル名の近くにあるドロップダウン矢印をクリックして、**[Manage table with policy (ポリシーを使用してテーブルを管理)]** を選択します。
    - または、テーブルを右にドラッグして、**[ポリシー テーブルとして追加]** にドロップします。
2. ポリシーにテーブルが追加されると、シールドアイコン が左側のペインのテーブル名の右側に表示され、ポリシー テーブルであることが示されます。
3. 資格テーブルを選択して、次のいずれかを選択します。
    - ドロップダウン矢印をクリックして、**[Use as entitlement table (資格テーブルとして使用)]** を選択します。
    - または、テーブルを右にドラッグして、**[資格テーブルとして追加]** にドロップします。
4. ポリシーを適用するテーブルごとにドロップダウン メニューをクリックして列を選択し、ポリシー

テーブルを資格テーブルにマッピングします。



資格テーブルのポリシー列を使用した例



A. フィルターリングするデータには EMP_ID 列がありますが、従業員名の列はありません。ただし、EMP_ID と従業員の FULL NAME の両方の列を含む 2 番目のテーブルがあります。また、従業員の FULL NAME 列の値は、サイトの Tableau ユーザーのフルネームと一致します。
B. 従業員テーブルを資格テーブルとしてポリシーに追加してから、ポリシー テーブルの列名 EMP_ID を各テーブルの資格の列名 EMP_ID にマッピングできます。
C. 次に、ポリシー条件で FULLNAME() 関数を使用して、Tableau Server ユーザーのフルネームを資格テーブルの [FULL NAME] 列 (ポリシー列) と照合し、各ユーザーが自分のデータのみを表示できるようにします。

ポリシー条件を記述する

データポリシーを作成する最後のステップは、ポリシー条件を作成することです。これは計算または式であり、行レベルのアクセス権限を定義するために使用します。ポリシー条件は、ユーザー機能を通じてユーザーまたはグループへのアクセスを制限するためによく使用されます。

ポリシー条件は、

- データポリシーで必要です。
- true または false に評価できる必要があります。
- ポリシー条件が true の場合に行を表示します。

ポリシー タブを閉じても、作業内容は破棄されません。

ポリシー条件の例

[地域] 列の値が "北部" である行のみを表示します。

```
[Region] = "North"
```

サインインしたユーザーが、ユーザーの名前が EmployeeName の値と一致する行を表示できるようにします。

```
FULLNAME() = [EmployeeName]
```

[マネージャー] グループのメンバーがすべての行を表示できるようにしますが、ユーザーはユーザー名が [employee_name] 列の値と一致する行のみを表示できます。

```
ISMEMBEROF('Managers') OR USERNAME() = [employee_name]
```

ポリシー条件でサポートしている Tableau 関数

ポリシー条件は、一部分の Tableau 関数をサポートします。

- 論理 (Null 関連を除く)
- 文字列
- ユーザー
- 日付
- 数字: MIN、MID、MAX

サポートされている具体的な機能を確認するには、仮想接続エディターの [データポリシー] タブの画面右側にある **[参照]** パネルを参照してください。

> **注**: 仮想接続にユーザー関数 (例: `USERNAME()`) を含むデータポリシーがあり、ワークブックまたはデータソースから接続して抽出を作成すると、抽出には抽出の作成時点で仮想接続データポリシーに一致する行のみが含まれます。データポリシーに含まれるユーザー関数を使用した仮想接続を活用するには、抽出ではなく、ワークブックまたはデータソースから仮想接続へのライブ接続を使用します。

実行可能なユーザー

データポリシーを作成するには、次のことが必要です。

- 仮想接続が接続する先のデータベースへの認証資格情報を持っている。
- サーバー管理者、サイト管理者、または Creator である。

次のステップ

データポリシーを作成したら、次のステップは、期待どおりに機能することを確認することです。「[ユーザーとしてプレビュー] を使用して、行レベルのセキュリティをテストする」を参照してください。または、仮想接続とデータポリシーを他のユーザーと共有する準備ができている場合は、「仮想接続のパブリッシュとパーミッションの設定」を参照してください。

リソース

計算の詳細については、Tableau Desktop および Web オーサリング ヘルプの「Tableau での計算を理解する」を参照してください。

ユーザー関数の詳細については、Tableau Desktop および Web 作成ヘルプの「ユーザー関数」を参照してください。

Tableau のその他の行レベルのセキュリティオプションについては、Tableau Server ヘルプの「Tableau の行レベルのセキュリティオプションの概要」を参照してください。

[ユーザーとしてプレビュー] を使用して、行レベルのセキュリティをテストする

[ユーザーとしてプレビュー] を使用して、データポリシーをテストします。ユーザーに表示されるデータを表示して、行レベルのセキュリティが期待どおりに機能していることを確認します。このテストは、データポリシーによってテーブル内の行が表示されなくなっている場合 (たとえば、営業担当者だけが行を表示でき、あなたは営業担当者ではない場合など) に役立ちます。

データポリシーが適用されたときにデータをプレビューするには:

1. テーブルを選択します。
2. [テーブルの詳細] セクションで、**[ポリシーを適用]** チェックボックスをオンにします。
3. **[ユーザーとしてプレビュー]** をクリックして、**[グループ]** (オプション) と **[ユーザー]** を選択します。
4. ポリシーに、テーブルの詳細にあるユーザーの正しいデータが表示されていることを確認します。
5. 必要に応じて、他のユーザーに対してこの手順を繰り返します。

ヒント: [テーブルの詳細] で、 をクリックして、表示する値やデータポリシーによって除外する値など、列の値の範囲を表示します。ポリシーがデータを正しくフィルターしていることを示す適切な指標となる 1 つまたは 2 つの列を選択します

実行可能なユーザー

仮想接続をテストするには、以下が必要です。

- 仮想接続が接続する先のデータベースへの認証資格情報を持っている。
- サーバー管理者、サイト管理者、または Creator である。

次のステップ

データポリシーをテストした後、仮想接続を他のユーザーと共有する準備ができたら、「仮想接続のパブリッシュとパーミッションの設定」を参照してください。

仮想接続のパブリッシュとパーミッションの設定

仮想接続エディターで作業する場合、作業中の変更は下書きとして自動的に保存されます。新しい仮想接続を他のユーザーと共有するには、パブリッシュする必要があります。

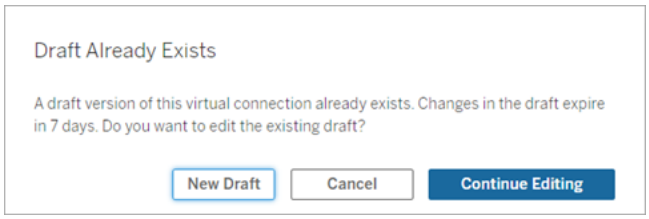
下書きの保存

接続の下書きを手動で保存するには、ツールバーで保存アイコン をクリックするか、メニューから **[ファイル] > [下書きを保存]** を選択します。

パブリッシュされた仮想接続を編集する場合でも、他のユーザーは、現在パブリッシュ状態にある接続を引き続き使用できます。エディターで接続を処理している間、更新を下書きとして保存できます。仮想接続への更新を他のユーザーと共有するには、更新をパブリッシュする必要があります。

進行中の下書き

パブリッシュされた仮想接続の更新中にエディターを閉じた場合、次の 7 日以内にエディターで接続を開いたときに、既存の下書きを引き続き編集するか、新しい下書きを開始するか、**[キャンセル]** をクリックして、現在パブリッシュ状態にある接続を開くかを選択できます。



パブリッシュされていない仮想接続の下書きバージョンに戻るには、エディタを閉じる**前**に、下書きの URL を手動で保存する必要があります。次の 7 日以内に接続を処理するときに、その URL を使用して、エディターで下書きを開くことができます。例:

```
https://yourserver.test.com/published-connection-
editor/?draft=d1789edc-5d9f-40ae-988d-9fc879f37a98
```

接続のパブリッシュ

新しい接続をパブリッシュするには、次の手順を実行します。

1. エディターの右上隅にある **[パブリッシュ]** ボタンをクリックするか、メニューから **[ファイル] > [パブリッシュ]** を選択します。
2. [パブリッシュ] ダイアログ ボックスで次の手順を実行します。
   a. **[名前]** フィールドに接続の名前を入力します。
   b. 接続を保存するプロジェクトを選択します。
3. **[パブリッシュ]** をクリックします。

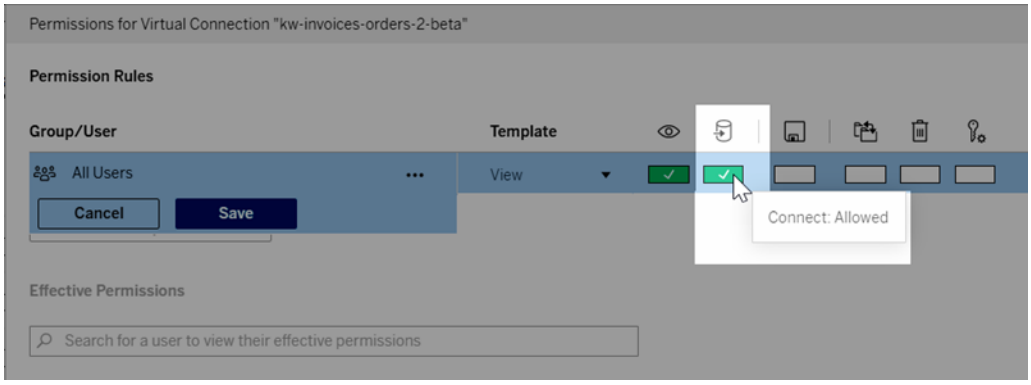
更新された接続をパブリッシュするには、エディターの右上隅にある **[パブリッシュ]** ボタンをクリックするか、メニューから **[ファイル] > [パブリッシュ]** を選択します。

仮想接続でのパーミッションの設定

仮想接続をパブリッシュした後、他のユーザーが使用できるようにパーミッションを設定する必要があります。デフォルトでは、すべてのユーザーが接続を**表示**できます。つまり、Tableau で [仮想接続] の下に接続が表示されますが、**接続**機能を許可に設定しない限り、仮想接続を使用できるのは自分と管理者だけになります。Connect 機能の詳細については、「パーミッション」を参照してください。

パーミッションを設定するには、次の手順を実行します。

1. 仮想接続に移動します。
2. [アクション] メニュー (...) を開いて **[パーミッション]** をクリックします。
3. [接続] アイコンの下のボックスをオンにして、すべてのユーザーに接続を許可します。



ヒント: 特定のユーザーまたはグループのみにパーミッションを付与する場合は、ルールを追加できます。

4. **[保存]** をクリックします。

Tableau コンテンツのパーミッションに関する詳細は、パーミッションを参照してください。仮想接続を使用するデータソースやワークブックなどの Tableau コンテンツをパブリッシュするときにパスワードを埋め込む方法については、Tableau Server ヘルプの「仮想接続」を参照してください。

実行可能なユーザー

仮想接続をパブリッシュしたり、パーミッションを設定したりするには、以下が必要です。

- 仮想接続が接続する先のデータベースへの認証資格情報を持っている。
- サーバー管理者、サイト管理者、または Creator である。

次のステップ

仮想接続をパブリッシュして、そのパーミッションを設定した後、仮想接続を使用するできます。

仮想接続の抽出更新のスケジュールを設定する

仮想接続のメリットの 1 つは、同じ抽出を何回も再利用することにより、データの急増を減らし、冗長な抽出更新ジョブを削除できることです。仮想接続を使用するすべてのコンテンツで抽出データ

を新鮮な状態に保つために、接続をパブリッシュした後に、接続のテーブルの抽出更新スケジュールを作成することができます

また、仮想接続を使用するデータソースとワークブックの抽出更新のスケジュールを設定することもできます。「Tableau Cloud での更新のスケジュール」と「定期的なデータ更新」(Tableau Server) を参照してください。

テーブルを抽出する

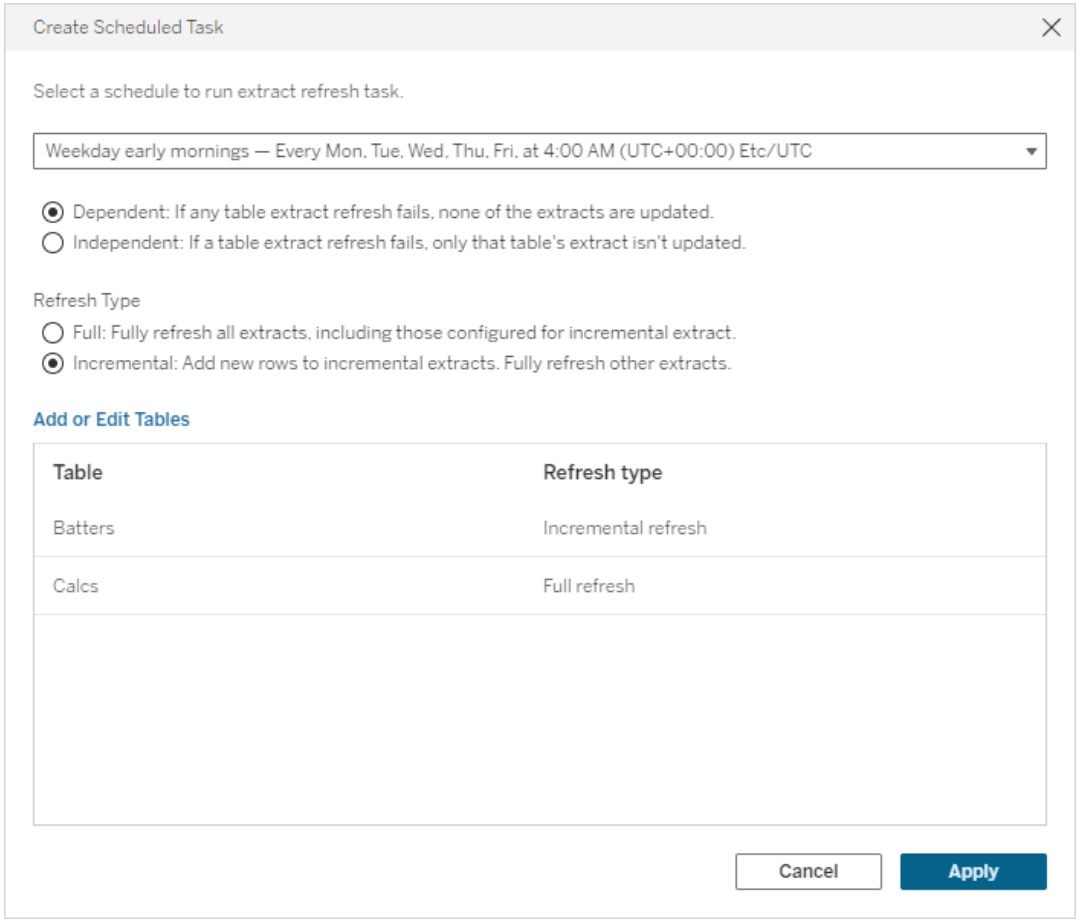
「テーブル データの抽出」を参照してください。

Tableau Server での抽出更新のスケジュール

1. 仮想接続のページに移動します(**[ホーム]** または **[探索]** ページで、ドロップダウン メニューから**[仮想接続]** をクリックし、仮想接続を選択します。)
2. ページの上部に**「データが抽出されます」**という見出しが表示されます。**「データはライブです」**と表示される場合は、ブラウザーをリフレッシュしてください。
3. **[スケジュールされたタスク]** タブを選択し、**[+新しいタスク]** をクリックします。
4. **[スケジュールされたタスクの作成]** ダイアログ ボックスが開きます。**[スケジュールされたタスクの作成]** ダイアログ ボックスは、Tableau Server とTableau Cloud では外観が異なります。Tableau Cloud を使用している場合は、Tableau Cloud 製品ヘルプの「仮想接続の抽出更新のスケジュール」を参照してください。
5. ドロップダウン メニューからスケジュールを選択します。
6. 複数のテーブルで抽出を使用する場合は、**[依存]** または **[独立]** を選択します (Tableau Server 2023.3 以前では、これらのオプションは [テーブルの同期を維持する] と[テーブルを個別に更新する] でしたが、基本的な意味は同じです)。
    - **[依存]** は、1 つまたは複数のテーブルの抽出更新ジョブが失敗した場合、どの抽出も更新されないことを意味します。
    - **[独立]** は、あるテーブルの抽出ジョブの成功または失敗が、他のテーブルの抽出が更新されるかどうかに影響しないことを意味します。
7. **[更新タイプ]** を選択します。仮想接続エディターで抽出の種類を設定できます。詳細については、「仮想接続の作成」ページの「増分抽出」セクションを参照してください (Tableau Server 2023.3 以前では、仮想接続は増分抽出をサポートしていないため、これらのオプションは表示されません)。
    - **[完全]** は、仮想接続内のすべての抽出に対して、完全抽出更新または増分抽出更新のどちらに設定されているかに関係なく、完全抽出更新ジョブが実行されることを意味します。

- **[増分]** は、仮想接続内のすべての増分抽出に対して、増分抽出更新ジョブが実行されることを意味します。仮想接続内の他のすべての抽出については、完全抽出更新ジョブが実行されます。

8. **[テーブルの追加または編集]** を選択して、更新するテーブルを選択します。
9. **[OK]** を選択します。
10. **[適用]** を選択します。



更新の抽出の時間制限

実行時間の長い更新タスクがすべてのシステム リソースを消費して、サイト上の他の抽出の更新を妨げてしまうことがないようにするため、仮想接続の抽出更新には 2 時間の時間制限が設けられています。更新タスクのタイムアウト制限に関する詳細とこれらのエラーを解決するための推奨事項については、「抽出更新の時間制限」を参照してください。注: 2024 年 6 月の Tableau Cloud および Tableau Server 2024.2 以降、仮想接続はテーブル抽出の増分更新をサポートします。このリリースより前は、仮想接続は完全更新のみをサポートしていました。

実行可能なユーザー

仮想接続をパブリッシュしたり、パーミッションを設定したりするには、次の認証資格情報とロールを持っている必要があります。

- 仮想接続が接続する先のデータベースへの認証資格情報を持っている。
- サーバー管理者、サイト管理者、または Creator である。

次のステップ

仮想接続用の抽出更新をスケジュールした後、仮想接続を使用する。仮想接続を使用する

仮想接続を使用する

仮想接続をパブリッシュして、パーミッションを設定すると、ユーザーは仮想接続を使用して、Tableau のすべてのデータにアクセスするのと同じ方法でデータに接続できるようになります。仮想接続または接続のデータポリシーを編集する必要がある場合 (参照元スキーマを変更する場合など) は、仮想接続エディターで接続を開き、変更を加えて、更新を保存またはパブリッシュします。ワークブック内の既存のデータソースを仮想接続に置き換えることもできます。

仮想接続に接続する

Tableau Cloud または Tableau Server で Web を作成するには、次の手順を実施します。

1. [ホーム] または [探索] ページで、**[新規]** をクリックします。
2. 作成するコンテンツのタイプ (ワークブック、フロー、またはパブリッシュされたデータソース) を選択します。
3. [データへの接続] > [このサイト上] > [コンテンツ タイプ] ドロップダウン メニューで、**[仮想接続]** を選択します。
4. 接続の名前を選択し、**[接続]** をクリックします。

Tableau Desktop とTableau Prep

1. [接続] ペインの [データの検索] で、**[Tableau Server]** をクリックします。
2. サーバー名を入力して **[接続]** をクリックするか、**[Tableau Cloud]** をクリックします。
3. 求められた情報を入力します。
4. [データの検索] ダイアログ ボックスの [コンテンツ タイプ] ドロップダウン メニューから、**[仮想接続]** を選択します。
5. 接続の名前を選択し、**[接続]** をクリックします。

> **注:** 仮想接続を使用して接続する場合は、認証資格情報を入力する必要はありません。データにアクセスするための認証資格情報は接続に埋め込まれています。

仮想接続またはデータポリシーを編集する

パブリッシュされた仮想接続を編集する場合でも、他のユーザーは、現在パブリッシュ状態にある接続を引き続き使用できます。詳細については、仮想接続のパブリッシュとパーミッションの設定を参照してください。

接続を編集するには、[探索] ページから接続に移動します。データベースの認証資格情報が接続に埋め込まれている場合でも、データベースの認証資格情報を持つユーザーのみが仮想接続に変更を加えることができることに注意してください。

1. ドロップダウン メニューから**[すべての仮想接続]** を選択し、編集する接続を選択します。
2. **[仮想接続を編集]** をクリックします。
3. 接続のために求められた情報を入力します。接続を編集するには、データにアクセスするために必要な認証資格情報を入力する必要があります。
4. **[サインイン]** をクリックします。
5. 仮想接続エディターで変更を加えてから、下書きを保存するか、接続をパブリッシュします。

参照元スキーマの変更に対応する

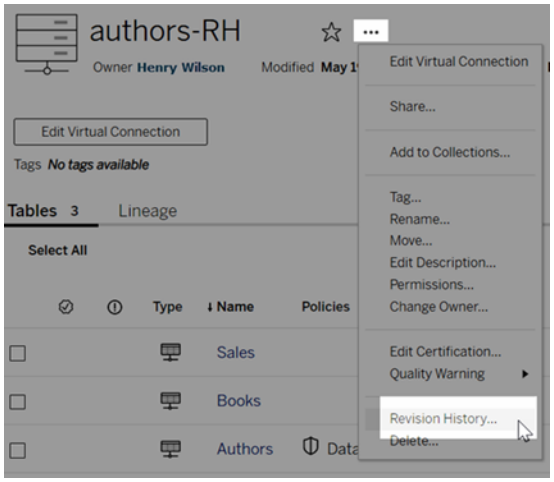
仮想接続の参照元スキーマが変更された場合 (たとえば、テーブルが追加または削除された場合、列が追加または列の名前が変更された場合など)、スキーマの変更を反映するように仮想接続を編集してから、接続をもう一度パブリッシュする必要があります。(接続に抽出がある場合は、必ず抽出を更新してください。)このように、新しいデータがすべてのユーザーに公開される前に、接続のテーブル、列、およびポリシーを追加または編集することができます。

仮想接続のリビジョン履歴を操作する

仮想接続をパブリッシュすると、Tableau Cloud または Tableau Server のリビジョン履歴にバージョンが保存されます。いつでも前のバージョンに戻すことができます。

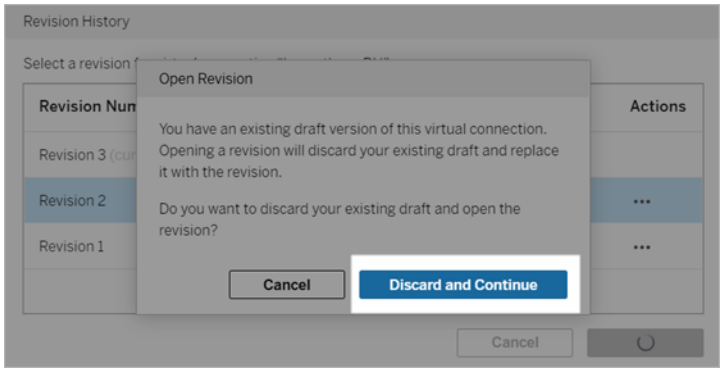
リビジョン履歴にアクセスするには、**Creator** サイトロールと**表示**および**上書き**のパーミッションが必要です。

仮想接続のリビジョン履歴を表示するには、仮想接続のアクション メニューをクリックします (。。) **[リビジョン履歴]** をクリックします。

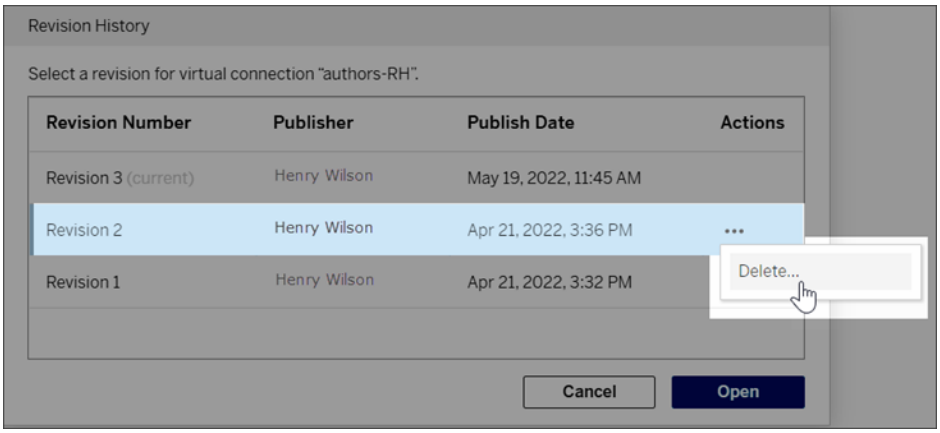


仮想接続リビジョンを復元または削除する

仮想接続リビジョンを復元するには、リビジョンを選択し、**[開く]** をクリックします。次に、既存のバージョンの接続を破棄するように求められます。**[Discard and Continue (破棄して続行)]** をクリックすると、選択したリビジョンが現在の接続のバージョンになります。



リビジョンを削除するには、リビジョンのアクション メニューから (。。.)、**[削除]** をクリックします。



ワークブック内の既存のデータソースを仮想接続に置き換える

Tableau Cloud または Tableau Server で Web を作成するには、次の手順を実施します。

1. ワークブックをダウンロードします。詳細については、Tableau Desktop ヘルプの「ビューとワークブックのダウンロード」を参照してください。
2. Tableau Desktop でワークブックを開き、既存のデータソースを仮想接続に置き換えます。詳細については、Tableau Desktop ヘルプの「データソースの置換」を参照してください。
3. Tableau Desktop で、ワークブックを Tableau Cloud または Tableau Server サイトにアップロードします。詳細については、Tableau Desktop ヘルプの「Tableau サイトへのワークブックのアップロード」を参照してください。
4. Tableau Cloud または Tableau Server で **[パブリッシュ]** をクリックして、変更をサーバーに保存します。

Tableau Desktop の場合:

1. ワークブックを開き、既存のデータソースを仮想接続に置き換えます。詳細については、Tableau Desktop ヘルプの「データソースの置換」を参照してください。
2. ワークブックをもう一度パブリッシュします。詳細については、Tableau Desktop ヘルプの「ワークブックをパブリッシュする簡単な手順」を参照してください。

実行可能なユーザー

仮想接続を使用するには、サーバー管理者かサイト管理者 Creator、または Creator である必要があります。

仮想接続またはデータポリシーを編集するには、次の認証資格情報とロールを持っている必要があります。

- 仮想接続が接続するデータベースにサインインするための認証資格情報
- サーバー管理者、サイト管理者、または Creator

既存のコンテンツを移行して仮想接続を使用するには、次のロールを持っている必要があります。

- サーバー管理者またはサイト管理者
- データソースの所有者でもある Creator