

Tableau Server en entreprise

Guide de déploiement

Dernière mise à jour 02/05/2024

© 2024 Salesforce, Inc.



Sommaire

Guide de déploiement de Tableau Server en entreprise	1
À qui s'adresse ce guide	2
Version	2
Principales fonctionnalités	3
Licences	3
Partie 1 - Comprendre le déploiement en entreprise	4
Normes de l'industrie et exigences de déploiement	4
Mesures de sécurité	5
Niveau proxy Web	6
Équilibrateurs de charge	6
Niveau Application	7
Niveau Données	7
Partie 2 - Comprendre l'architecture de référence du déploiement de Tableau Server	8
Processus Tableau Server	9
Référentiel PostgreSQL	10
Nœud 1 : Nœud initial	11
Basculement du Nœud 1 et restauration automatisée	11
Nœuds 1 et 2 : Serveurs d'applications	12
Mise à l'échelle des serveurs d'applications	13
Nœuds 3 et 4 : Serveurs de données	14

Mise à l'échelle des serveurs de données	15
Partie 3 - Préparer le déploiement de Tableau Server en entreprise	16
Sous-réseaux	17
Règles de pare-feu/groupe de sécurité	17
Niveau Web	17
Niveau Application	18
Niveau Données	19
Bastion	19
Exemple : Configurer des sous-réseaux et des groupes de sécurité dans AWS	20
Architecture de référence AWS	21
Diapositive 1 : Topologie de sous-réseau VPC et instances EC2	21
Diapositive 2 : Flux de protocole et connectivité	22
Diapositive 3 : Zones de disponibilité	23
Diapositive 4 : Groupes de sécurité	24
Zones de disponibilité AWS et haute disponibilité	24
Configuration de VPC	24
Configurer VPC	25
Configurer les groupes de sécurité	26
Spécifier les règles de trafic entrant et sortant	27
Règles du groupe de sécurité Public	27
Règles du groupe de sécurité Privé	28
Règles du groupe de sécurité Données	29

Règles du groupe de sécurité de l'hôte Bastion	29
Activer l'attribution automatique de l'adresse IP publique	30
Équilibreur de charge	31
Configurer les ordinateurs hôtes	31
Matériel minimal recommandé	31
Structure du répertoire	32
Exemple : Installer et préparer les ordinateurs hôtes dans AWS	33
Détails de l'instance hôte	33
Tableau Server	33
Hôte Bastion	34
Tableau Server Independent Gateway	34
Hôte PostgreSQL EC2	34
Vérification : connectivité VPC	34
Exemple : connexion à l'hôte bastion dans AWS	35
Partie 4 - Installer et configurer Tableau Server	36
Avant de commencer	36
Installer, configurer PostgreSQL et créer une sauvegarde tar	37
Versions de PostgreSQL	37
Installer PostgreSQL	38
Configurer Postgres	39
Effectuer une sauvegarde tar PostgreSQL de l'Étape 1	40
Avant l'installation	42

Installer le nœud initial de Tableau Server	42
Exécuter le package d'installation et initialiser TSM	42
Activer et enregistrer Tableau Server	44
Configurer le magasin d'identités	45
Configurer Postgres externe	45
Terminer l'installation du Nœud 1	46
Vérification : Configuration du Nœud 1	47
Effectuer des sauvegardes tar de l'Étape 2	48
Installer Tableau Server sur les nœuds restants	52
Générer, copier et utiliser le fichier d'amorçage pour initialiser TSM	54
Configurer les processus	55
Configurer le Nœud 2	56
Configurer le Nœud 3	57
Déployer l'ensemble de service de coordination sur les Nœuds 1 à 3	58
Effectuer des sauvegardes tar de l'Étape 3	59
Configurer le Nœud 4	63
Configuration et vérification du processus final	63
Effectuer une sauvegarde	65
Partie 5 - Configuration du niveau Web	67
Tableau Server Independent Gateway	68
Authentification et autorisation	68
Pré-authentification avec un module AuthN	69

Présentation de la configuration	70
Exemple de configuration de niveau Web avec Tableau Server Independent Gateway	71
Préparer l'environnement	72
Installer la passerelle indépendante	72
Passerelle indépendante : connexion directe ou par relais	75
Configurer la connexion par relais	76
Configurer la connexion directe	77
Vérification : configuration de la topologie de base	78
Configurer l'équilibreur de charge d'application AWS	79
Étape 1 : Créer un groupe cible	79
Étape 2 : Lancer l'assistant d'équilibrage de charge	80
Configuration de l'assistant	80
Configuration d'une seule page	82
Étape 3 : Activer la persistance	83
Étape 4 : Définir le délai d'inactivité sur l'équilibreur de charge	83
Étape 5 : Vérifier la connectivité LBS	83
Mettre à jour le DNS avec l'URL publique de Tableau	84
Vérifier la connectivité	84
Exemple de configuration d'authentification : SAML avec IdP externe	84
Créer un compte d'administrateur Tableau	85
Configurer l'application de pré-authentification Okta	85
Créer et affecter un utilisateur Okta	87

Installer Mellon pour la pré-authentification	87
Configurer Mellon comme module de pré-authentification	88
Créer une application Tableau Server dans Okta	90
Définir la configuration du module d'authentification sur Tableau Server	91
Activer SAML sur Tableau Server pour IdP	92
Redémarrer le service tsig-httpd	94
Valider la fonctionnalité SAML	95
Configurer le module d'authentification sur la seconde instance de la passerelle indépendante	95
Partie 6 - Configuration après l'installation	98
Configurer SSL/TLS depuis l'équilibreur de charge vers Tableau Server	98
Avant de configurer TLS	99
Configurer les ordinateurs de la passerelle indépendante pour TLS	100
Étape 1 : Distribuer les certificats et les clés à l'ordinateur de la passerelle indépendante	100
Étape 2 : Mettre à jour les variables d'environnement pour TLS	101
Étape 3 : Mettre à jour le fichier de configuration stub pour le protocole HK	101
Étape 4 : Copier le fichier stub et redémarrer le service	102
Configurer le Nœud 1 de Tableau Server pour TLS	103
Étape 1 : Copier les certificats et les clés et arrêter TSM	103
Étape 2 : Définir les ressources de certificat et activer la configuration de la passerelle indépendante	103
Étape 3 : Activer « SSL externe » pour Tableau Server et appliquer les modifications	104

Étape 4 : Mettre à jour le fichier JSON de configuration de la passerelle et démarrer tsm	105
Mettre à jour les URL du module d'authentification de l'IdP sur HTTPS	106
Configurer l'équilibreur de charge AWS pour HTTPS	106
Valider TLS	108
Configurer la deuxième instance de passerelle indépendante pour SSL	108
Configurer SSL pour Postgres	110
Facultatif : activer la validation de la confiance des certificats sur Tableau Server pour SSL Postgres	113
Installer le client Postgres sur le Nœud 1	113
Copiez le certificat racine sur le Nœud 1	114
Connectez-vous à l'hôte Postgres sur SSL depuis le Nœud 1	114
Configurer SMTP et les notifications d'événement	115
Installer le pilote PostgreSQL	117
Configurer la politique de mot de passe fort	117
Partie 7 - Validation, outils et dépannage	119
Validation du système de basculement	119
Récupération automatisée du nœud initial	120
Résolution des problèmes de récupération du nœud initial	122
Reconstitution du nœud défaillant	122
switchto	123
Résolution des problèmes Tableau Server Independent Gateway	125
Redémarrer le service tableau-tsig	126

Trouver les chaînes incorrectes	126
Recherche dans les journaux pertinents	127
Fichiers journaux de la passerelle indépendante	127
Fichier journal de Tableau Server tabadminagent	127
Recharger le fichier stub httpd	128
Supprimer ou déplacer des fichiers journaux	129
Erreurs de navigateur	129
Vérifier la connexion TLS de Tableau Server à la passerelle indépendante	130
Annexe - Boîte à outils du déploiement AWS	132
Script d'installation automatisée TabDeploy4EDG	132
Exemple : Automatiser le déploiement de l'infrastructure AWS avec Terraform	135
Objectif	135
État final	136
Exigences	137
Avant de commencer	137
Étape 1 - Préparer l'environnement	137
A. Télécharger et installer Terraform	137
B. Générer une paire de clés privée-publique	138
C. Télécharger le projet et ajouter le répertoire d'état	138
Étape 2 : Personnaliser les modèles Terraform	139
versions.tf	139
key-pair.tf	139

locals.tf	140
providers.tf	140
elb.tf	140
variables.tf	141
modules/tableau_instance/ec2.tf	141
Étape 3 - Exécuter Terraform	143
A. Initialiser Terraform	143
B. Planifier Terraform	143
C. Appliquer Terraform	143
Facultatif : Supprimer Terraform	143
Étape 4 - Connexion à bastion	144
Étape 5 - Installer PostgreSQL	145
Étape 6 - (Facultatif) Exécuter DeployTab4EDG	145
Annexe - Niveau Web avec exemple de déploiement Apache	146
Installer Apache	147
Configurer le serveur proxy pour tester la connectivité à Tableau Server	148
Vérification : configuration de la topologie de base	149
Configurer l'équilibrage de charge sur le proxy	149
Copier la configuration sur le deuxième serveur proxy	150
Configurer l'équilibreur de charge d'application AWS	151
Étape 1 : Créer un groupe cible	151
Étape 2 : Lancer l'assistant d'équilibrage de charge	152

Configuration de l'assistant	152
Configuration d'une seule page	153
Étape 3 : Activer la persistance	154
Étape 4 : Définir le délai d'inactivité sur l'équilibreur de charge	155
Étape 5 : Vérifier la connectivité LBS	155
Mettre à jour le DNS avec l'URL publique de Tableau	156
Vérifier la connectivité	156
Exemple de configuration d'authentification : SAML avec IdP externe	156
Créer un compte d'administrateur Tableau	157
Configurer l'application de pré-authentification Okta	157
Créer et affecter un utilisateur Okta	159
Installer Mellon pour la pré-authentification	159
Configurer Mellon comme module de pré-authentification	160
Créer une application Tableau Server dans Okta	163
Activer SAML sur Tableau Server pour IdP	164
Valider la fonctionnalité SAML	166
Résolution des problèmes de validation	167
Configurer SSL/TLS depuis l'équilibreur de charge vers Tableau Server	168
Exemple : Configurer SSL/TLS dans l'architecture de référence AWS	168
Étape 1 : Rassembler les certificats et les clés associées	169
Étape 2 : Configurer le serveur proxy pour SSL	170
Étape 3 : Configurer Tableau Server pour SSL externe	173

Étape 4 : Configuration facultative de l'authentification173

Étape 5 : Configurer l'équilibreur de charge AWS pour HTTPS173

Étape 6 : Vérifier SSL175

Guide de déploiement de Tableau Server en entreprise

L'objectif du Guide de déploiement de Tableau Server en entreprise (EDG) est de fournir des recommandations pour le déploiement de Tableau Server (sur site ou dans le cloud). Le Guide inclut des conseils de déploiement pour des scénarios d'entreprise dans le contexte d'une architecture de référence. Nous avons testé l'architecture de référence pour vérifier la conformité avec les benchmarks de sécurité, d'évolutivité et de performance conformes aux meilleures pratiques de l'industrie.

Au niveau supérieur, les principales caractéristiques d'un déploiement standard en entreprise consistent en une topologie à plusieurs niveaux où chaque couche de fonctionnalité d'application serveur (niveau Passerelle Web, niveau Application et niveau Données) est liée et protégée par des sous-réseaux à accès contrôlé. Les utilisateurs accédant à l'application serveur depuis Internet sont authentifiés au niveau Web. Une fois authentifiée, la demande est envoyée par proxy à un sous-réseau protégé où le niveau Application gère la logique métier. Les données de grande valeur sont protégées par le sous-réseau : le niveau Données. Les services dans le niveau Application communiquent via le réseau protégé avec le niveau Données pour traiter les demandes de données adressées aux sources de données principales.

Dans ce déploiement, la sécurité est au premier plan de toutes les décisions de conception et de mise en œuvre. La fiabilité, les performances et l'évolutivité sont toutefois également des exigences prioritaires. Compte tenu de la conception distribuée et modulaire de l'architecture de référence, la fiabilité et les performances évoluent de manière linéairement prévisible avec co-localisation stratégique des services compatibles à chaque nœud et ajout de services aux goulets d'étranglement.

À qui s'adresse ce guide

L'EDG a été développé pour les administrateurs informatiques d'entreprise qui peuvent avoir besoin des éléments suivants :

- Déploiement Tableau géré par l'informatique
- Application de la conformité du secteur d'activité
- Meilleures pratiques de déploiement dans le secteur d'activité
- Déploiement sécurisé par défaut

L'EDG est un guide de mise en œuvre pour le déploiement de l'architecture de référence de l'entreprise. Bien que cette version de l'EDG comprenne un exemple d'implémentation AWS/Linux, le guide peut être utilisé comme ressource par des administrateurs informatiques d'entreprise expérimentés. Il les aidera à déployer l'architecture de référence prescrite dans n'importe quel environnement de centre de données aux normes du secteur.

Version

Cette version d'EDG a été développée spécifiquement pour la version 2021.2.3 (ou ultérieure) de Tableau Server. Bien que vous puissiez utiliser EDG comme référence générale pour le déploiement de versions plus anciennes de Tableau Server, nous vous recommandons de déployer l'architecture de référence avec Tableau Server 2021.2.3 ou ultérieur. Certaines fonctionnalités et options ne sont pas disponibles sur les anciennes versions de Tableau Server.

Pour les fonctionnalités et améliorations les plus récentes, nous vous recommandons de déployer EDG avec Tableau Server 2022.1.7 et versions ultérieures.

L'architecture de référence décrite dans ce guide prend en charge tous les clients Tableau : création Web avec des navigateurs compatibles, Tableau Mobile et Tableau Desktop version 2021.2.1 ou ultérieure. Les autres clients Tableau (Tableau Prep, Bridge, etc.) n'ont pas encore été validés avec l'architecture de référence.

Principales fonctionnalités

La première version de l'architecture de référence de Tableau Server présente les scénarios et fonctionnalités suivants :

- Pré-authentification du client : tous les clients Tableau pris en charge (Desktop, Mobile, création Web) s'authentifient auprès du fournisseur d'authentification d'entreprise au niveau Web avant d'accéder à l'instance Tableau Server interne. Ce processus est géré par la configuration d'un plug-in authN sur l'instance Tableau Server Independent Gateway faisant office de serveur proxy inverse. Consultez Partie 5 - Configuration du niveau Web.
- Déploiement zéro confiance : étant donné que tout le trafic vers les serveurs Tableau est pré-authentifié, l'ensemble du déploiement de Tableau fonctionne dans un sous-réseau privé qui n'exige pas une connexion de confiance.
- Référentiel externe : l'architecture de référence spécifie l'installation du référentiel Tableau sur une base de données PostgreSQL externe, permettant aux DBA de gérer, d'optimiser, de mettre à l'échelle et de sauvegarder le référentiel en tant que base de données générique.
- Récupération du nœud initial : l'EDG introduit un script qui automatise la restauration du nœud initial en cas de défaillance.
- Sauvegarde et restauration basées sur un fichier tar : utilisez des sauvegardes tar familières aux étapes stratégiques du déploiement de Tableau. En cas d'échec ou de mauvaise configuration du déploiement, vous pouvez rapidement revenir à l'étape de déploiement précédente en récupérant la sauvegarde tar associée.
- Amélioration des performances : la validation par le client et le laboratoire montre une amélioration des performances de 15 à 20 % en cas d'exécution d'EDG par rapport au déploiement standard.

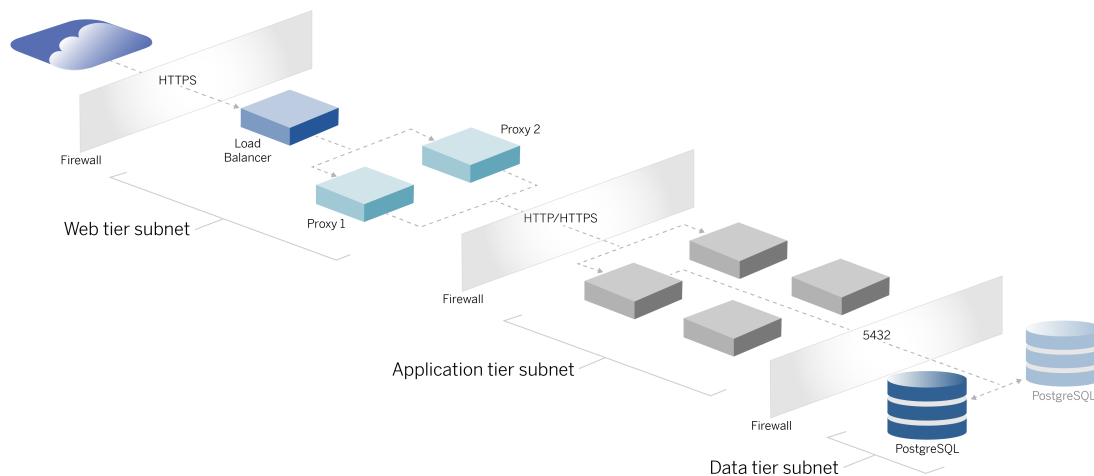
Licences

L'architecture de référence Tableau Server prescrite dans ce guide nécessite une licence Tableau Advanced Management pour activer le référentiel externe de Tableau Server. Vous pouvez également éventuellement déployer le répertoire de fichiers externe Tableau Server, qui nécessite également la licence Tableau Advanced Management. Consultez *À propos de Tableau Advanced Management sur Tableau Server* ([Linux](#)).

Partie 1 - Comprendre le déploiement en entreprise

La Partie 1 décrit plus en détail les fonctionnalités et les exigences d'un déploiement en entreprise aux normes de l'industrie pour lequel le Guide de déploiement de Tableau Server en entreprise a été conçu.

Le diagramme de réseau suivant montre un déploiement multi-niveaux de centre de données générique avec une architecture de référence Tableau Server.



Normes de l'industrie et exigences de déploiement

Cette section décrit les caractéristiques d'un déploiement aux normes de l'industrie. Voici les exigences pour lesquelles l'architecture de référence décrite a été conçue :

- **Modèle de réseau à plusieurs niveaux** : le réseau est lié par des sous-réseaux protégés pour limiter l'accès à chaque couche : couche Web, couche application et couche de données. Aucune communication unique ne peut traverser les sous-réseaux, car toutes les communications se terminent au sous-réseau suivant.

- Ports et protocoles bloqués par défaut : chaque sous-réseau ou groupe de sécurité bloquera par défaut tous les ports et protocoles entrants et sortants. La communication est activée, partiellement, en ouvrant des exceptions dans la configuration du port/-protocole.
- Authentification Web externe : les demandes des utilisateurs provenant d'Internet sont authentifiées par un module d'authentification sur le serveur proxy inverse au niveau Web. Par conséquent, toutes les demandes adressées à la couche Application sont authentifiées au niveau Web avant de passer à la couche Application protégée.
- Indépendant des plates-formes : la solution peut être déployée avec des applications serveur locales ou dans le cloud.
- Indépendant de la technologie : la solution peut être déployée dans un environnement de machine virtuelle ou dans des conteneurs. Elle peut également être déployée sur Windows ou Linux. Cette version initiale de l'architecture de référence et de la documentation d'accompagnement a toutefois été développée pour Linux s'exécutant dans AWS.
- Haute disponibilité : tous les composants du système sont déployés sous forme de cluster et sont conçus pour fonctionner dans un déploiement actif/actif ou actif/passif.
- Rôles cloisonnés : chaque serveur joue un rôle discret. Cette conception partitionne tous les serveurs de manière à ce que l'accès soit limité aux administrateurs spécifiques au service. Par exemple, les administrateurs de base de données gèrent PostgreSQL pour Tableau, les administrateurs d'identité gèrent le module d'authentification au niveau Web, les administrateurs réseau et cloud gèrent le trafic et la connectivité.
- Évolutivité linéaire : en tant que rôles discrets, vous pouvez faire évoluer chaque service de niveau indépendamment en fonction du profil de charge.
- Prise en charge des clients : l'architecture de référence prend en charge tous les clients Tableau : Tableau Desktop (versions 2021.2 ou ultérieures), Tableau Mobile et création Web Tableau.

Mesures de sécurité

Comme indiqué, une caractéristique principale de la conception de centre de données aux normes de l'industrie est la sécurité.

- Accès : chaque niveau est lié par un sous-réseau qui applique le contrôle d'accès au niveau de la couche réseau à l'aide du filtrage de port. L'accès à la communication entre les sous-réseaux peut également être appliqué par la couche application avec des

services authentifiés entre les processus.

- Intégration : l'architecture est conçue pour se connecter au fournisseur d'identité (IdP) sur le serveur proxy au niveau Web.
- Confidentialité : le trafic vers la passerelle Web est crypté à partir du client avec SSL. Le trafic vers les sous-réseaux internes peut également être crypté.

Niveau proxy Web

Le niveau Web est un sous-réseau dans la zone démilitarisée (DMZ) qui fait office de tampon de sécurité entre Internet et les sous-réseaux internes où les applications sont déployées. Le niveau Web héberge des serveurs proxy inverses qui ne stockent aucune information sensible. Les serveurs proxy inverses sont configurés avec un plugin AuthN pour pré-authentifier les sessions client avec un IdP de confiance, avant de rediriger la demande client vers Tableau Server. Pour plus d'informations, consultez Pré-authentication avec un module AuthN.

Équilibreurs de charge

La conception du déploiement recommande vivement de déployer une solution d'équilibrage de charge d'entreprise devant les serveurs proxy inverses.

Les équilibreurs de charge fournissent des améliorations importantes de la sécurité et des performances par les actions suivantes :

- Virtualisation de l'URL front-end pour les services de niveau Application
- Application du cryptage SSL
- Déchargement SSL
- Application de la compression entre le client et les services de niveau Web
- Protection contre les attaques DOS
- Haute disponibilité

Remarque : Tableau Server version 2022.1 inclut Tableau Server Independent Gateway. Independent Gateway est une instance autonome du processus Tableau Gateway

qui sert de proxy inverse compatible avec Tableau. Au moment de la publication, Independent Gateway a été validé, mais pas entièrement testé dans l'architecture de référence EDG. Une fois les tests complets terminés, l'EDG sera mis à jour avec les conseils prescriptifs de Tableau Server Independent Gateway.

Niveau Application

Le niveau Application se trouve dans un sous-réseau qui exécute la logique métier principale de l'application serveur. Le niveau Application se compose de services et de processus configurés sur des nœuds distribués dans un cluster. Le niveau Application n'est accessible qu'à partir du niveau Web et n'est pas directement accessible par les utilisateurs.

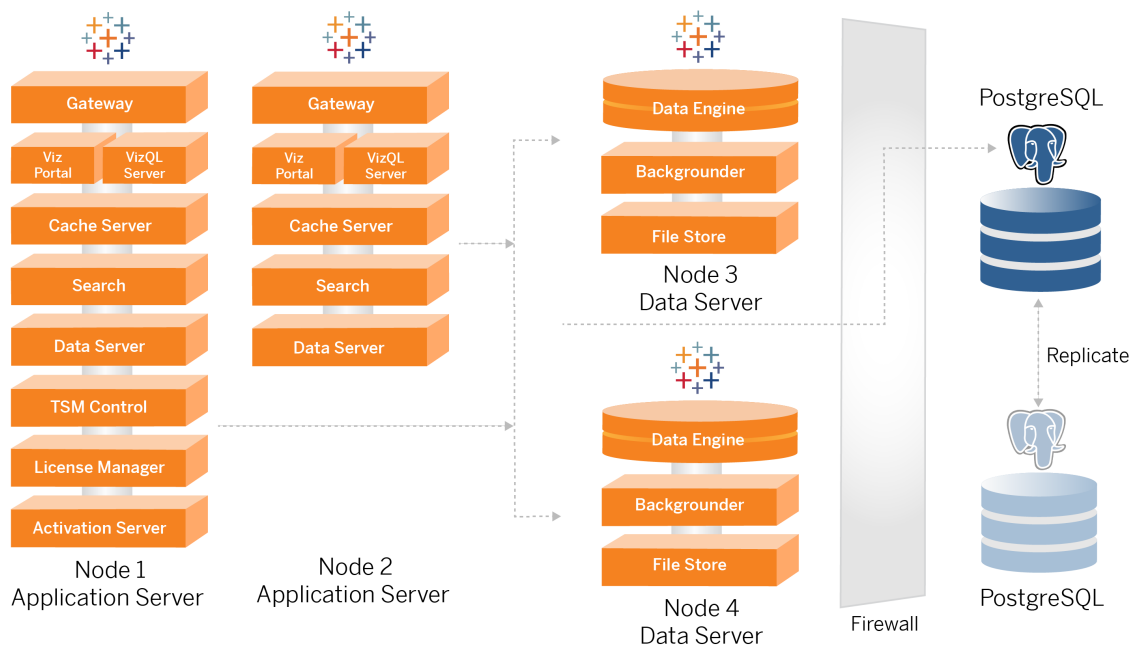
Les performances et la fiabilité sont améliorées grâce à la configuration des processus d'application visant à assurer la colocalisation des processus avec différents profils d'utilisation des ressources (par exemple, utilisation intensive des ressources du processeur ou des ressources de la mémoire).

Niveau Données

Le niveau Données est un sous-réseau qui contient des données précieuses. Tout le trafic vers ce niveau provient du niveau Application et est donc déjà authentifié. En plus des exigences d'accès au niveau de la couche réseau avec configuration de port, cette couche doit inclure un accès authentifié et, éventuellement, un trafic crypté au niveau de la couche Application.

Partie 2 - Comprendre l'architecture de référence du déploiement de Tableau Server

L'image suivante montre les processus Tableau Server pertinents et la manière dont ils sont déployés dans l'architecture de référence. Ce déploiement est considéré comme le déploiement Tableau Server minimal adapté à l'entreprise.



Les diagrammes de processus présentés dans cette rubrique sont destinés à montrer les principaux processus de définition de chaque nœud. Il existe de nombreux processus de prise en charge qui s'exécutent également sur des nœuds non affichés dans les diagrammes. Pour une liste de tous les processus, consultez la section de configuration de ce guide, Partie 4 - Installer et configurer Tableau Server.

Processus Tableau Server

L'architecture de référence de Tableau Server est un déploiement de cluster Tableau Server à quatre nœuds avec référentiel externe sur PostgreSQL :

- Nœud initial de Tableau Server (Nœud 1) : exécute les services d'administration et de licence TSM requis qui ne peuvent être exécutés que sur un seul nœud du cluster. Dans le contexte de l'entreprise, le nœud initial de Tableau Server est le nœud principal du cluster. Ce nœud exécute également des services d'application redondants avec le Nœud 2.
- Nœuds d'application Tableau Server (Nœud 1 et Nœud 2) : les deux nœuds traitent les demandes des clients, se connectent aux sources de données et les interrogent, et se connectent aux nœuds de données.
- Nœuds de données Tableau Server (Nœud 3 et Nœud 4) : deux nœuds dédiés à la gestion des données.
- PostgreSQL externe : cet hôte exécute le processus de référentiel Tableau Server. Pour un déploiement haute disponibilité, vous devez exécuter un hôte PostgreSQL supplémentaire pour la redondance active/passive.

Vous pouvez également exécuter PostgreSQL sur Amazon RDS. Pour plus d'informations sur les différences entre exécuter le référentiel sur une instance RDS plutôt que sur une instance EC2, consultez *Référentiel externe Tableau Server* ([Linux](#)).

Le déploiement de Tableau Server avec un référentiel externe nécessite une licence Tableau Advanced Management.

Si votre entreprise ne dispose pas d'une expertise interne en matière d'administration de bases de données, vous pouvez éventuellement exécuter le processus de référentiel Tableau Server dans la configuration PostgreSQL interne par défaut. Dans le scénario par défaut, le référentiel est exécuté sur un nœud Tableau avec PostgreSQL intégré. Dans ce cas, nous vous recommandons d'exécuter le référentiel sur un nœud Tableau dédié et un référentiel passif sur un nœud dédié supplémentaire pour prendre en charge le basculement du référentiel. Voir *Basculement du référentiel* ([Linux](#)).

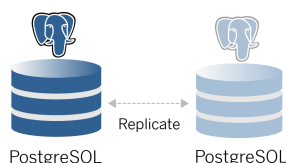
À titre d'exemple, l'implémentation AWS décrite dans ce guide explique comment déployer le référentiel externe sur PostgreSQL s'exécutant sur une instance EC2.

- Facultatif : si votre entreprise utilise un stockage externe, vous pouvez déployer le répertoire de fichiers externe Tableau en tant que service externe. Ce guide n'inclut pas le répertoire de fichiers externe dans le scénario de déploiement principal. Consultez *Installer Tableau Server avec un répertoire de fichiers externe* ([Linux](#)).

Le déploiement de Tableau Server avec un magasin de fichiers externe nécessite une licence Tableau Advanced Management.

Référentiel PostgreSQL

Le référentiel Tableau Server est une base de données PostgreSQL qui stocke les données de serveur. Ces données comprennent des informations sur les utilisateurs, les groupes, les affectations de groupes, les autorisations, les projets, les sources de données, les méta-données d'extraits et les informations d'actualisation Tableau Server.



Le déploiement PostgreSQL par défaut consomme près de 50 % des ressources de mémoire système. Selon son utilisation (pour la production et les grands déploiements de production), la consommation des ressources peut augmenter. Pour cette raison, nous vous recommandons d'exécuter le processus de référentiel sur un ordinateur qui n'exécute pas d'autre composant serveur exigeant beaucoup de ressources tel que VizQL, Backgrounder ou le moteur de données. L'exécution du processus de référentiel avec l'un de ces composants créera des conflits d'E/S, des contraintes de ressources et dégradera les performances globales du déploiement.

Nœud 1 : Nœud initial

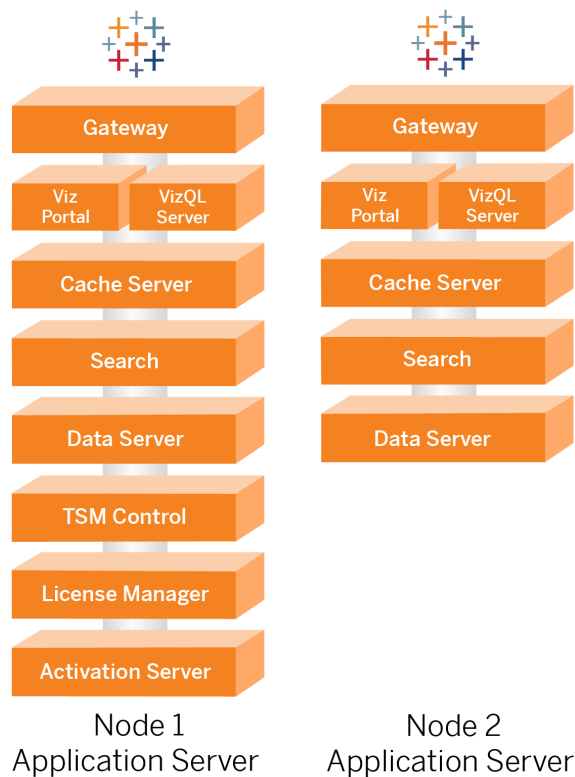
Le nœud initial exécute un petit nombre de processus importants et partage la charge des applications avec le Nœud 2.

Le premier ordinateur sur lequel vous installez Tableau, le « nœud initial », présente certaines caractéristiques uniques. Trois processus s'exécutent uniquement sur le nœud initial et ne peuvent pas être déplacés vers un autre nœud, sauf en situation d'échec : le service de licences (Gestionnaire de licences), le service d'activation et le contrôleur TSM (contrôleur d'administration).

Basculement du Nœud 1 et restauration automatisée

Les services Licence, Activation et Contrôleur TSM sont essentiels à l'intégrité d'un déploiement Tableau Server. En cas de défaillance du Nœud 1, les utilisateurs pourront toujours se connecter au déploiement de Tableau Server, car une architecture de référence correctement configurée acheminera les demandes vers le Nœud 2. Cependant, sans ces services de base, le déploiement sera dans un état critique d'échec imminent. Consultez Récupération automatisée du nœud initial.

Nœuds 1 et 2 : Serveurs d'applications



Les Nœuds 1 et 2 exécutent les processus Tableau Server qui traitent les demandes des clients, interrogent les sources de données, génèrent des visualisations, gèrent le contenu et l'administration, et exécutent la logique métier principale de Tableau. Les serveurs d'applications ne stockent pas de données utilisateur.

Remarque : Le terme « Serveur d'applications » fait également référence à un processus Tableau Server répertorié dans TSM. Le processus sous-jacent pour le « Serveur d'applications » est VizPortal.

Exécutés en parallèle, les Nœuds 1 et 2 s'adaptent aux demandes de service de la logique d'équilibrage de charge exécutée sur les serveurs proxy inverses. En tant que nœuds redon-

dants, si l'un de ces nœuds échoue, les demandes des clients et la maintenance sont gérées par le nœud restant.

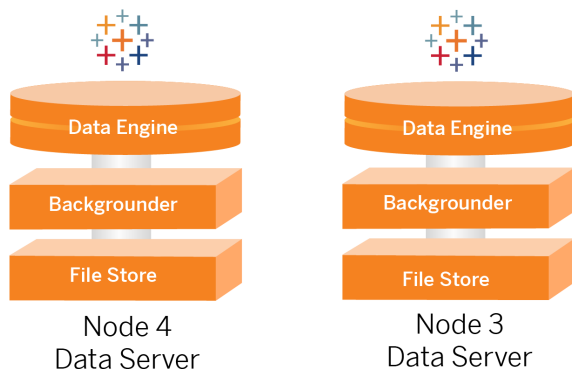
L'architecture de référence a été conçue pour que les processus d'applications supplémentaires s'exécutent sur le même ordinateur. Cela signifie que les processus ne sont pas en concurrence pour les ressources informatiques et ne créent pas de conflits.

Par exemple, VizQL, un service de traitement de cœurs sur les serveurs d'applications, est étroitement lié au processeur et à la mémoire et utilise près de 60-70 % du processeur et de la mémoire de l'ordinateur. Pour cette raison, l'architecture de référence est conçue de manière à ce qu'aucun autre processus lié à la mémoire ou au processeur ne se trouve sur le même nœud que VizQL. Les tests montrent que la quantité de charge ou le nombre d'utilisateurs n'affecte pas la mémoire ou l'utilisation du processeur sur les nœuds VizQL. Par exemple, la réduction du nombre d'utilisateurs simultanés dans notre test de charge affecte uniquement les performances du tableau de bord ou le processus de chargement de visualisation, mais ne réduit pas l'utilisation des ressources. Par conséquent, en fonction de la mémoire et du processeur disponibles lors des pics d'utilisation, vous pouvez envisager d'ajouter d'autres processus VizQL. Comme point de départ pour des classeurs standard, allouez 4 cœurs par processus VizQL.

Mise à l'échelle des serveurs d'applications

L'architecture de référence est conçue pour une échelle basée sur un modèle basé sur l'utilisation. Comme point de départ général, nous recommandons un minimum de deux serveurs d'applications, chacun prenant en charge jusqu'à 1000 utilisateurs. Au fur et à mesure que la base d'utilisateurs augmente, prévoyez d'ajouter un serveur d'applications pour chaque 1000 utilisateurs supplémentaires. Surveillez l'utilisation et les performances pour ajuster la base d'utilisateurs par hôte pour votre entreprise.

Nœuds 3 et 4 : Serveurs de données



Les processus du répertoire de fichiers, du moteur de données (Hyper) et du Backgrounder sont co-localisés sur les Nœuds 3 et 4 pour les raisons suivantes :

- Optimisation des extraits : l'exécution de Backgrounder, de Hyper et du répertoire de fichiers sur le même nœud optimise les performances et la fiabilité. Pendant le processus d'extraction, le Backgrounder interroge la base de données cible, crée le fichier Hyper sur le même nœud, puis le télécharge dans le répertoire de fichiers. En co-localisant ces processus sur le même nœud, le workflow de création d'extrait ne nécessite pas la copie de quantités de données sur le réseau ou les nœuds.
- Équilibrage des ressources complémentaire : le Backgrounder exige principalement des ressources processeur. Le moteur de données est un processus qui exige beaucoup de mémoire. Le couplage de ces processus permet une utilisation maximale des ressources sur chaque nœud.
- Consolidation des processus de données : chacun de ces processus étant des processus de données back-end, il est logique de les exécuter dans le niveau de données le plus sécurisé. Dans les futures versions de l'architecture de référence, les serveurs d'applications et les serveurs de données fonctionneront dans des niveaux distincts. Cependant, en raison des dépendances des applications dans l'architecture Tableau, les serveurs d'applications et de données doivent s'exécuter dans le même niveau à ce moment-là.

Mise à l'échelle des serveurs de données

Comme pour les serveurs d'applications, la planification des ressources requises pour les serveurs de données Tableau nécessite une modélisation basée sur l'utilisation. En général, partez de l'hypothèse que chaque serveur de données peut prendre en charge jusqu'à 2000 travaux d'actualisation d'extrait par jour. Au fur et à mesure que vos travaux d'extraction augmentent, ajoutez des serveurs de données supplémentaires sans le service Répertoire de fichiers. En règle générale, le déploiement du serveur de données à deux nœuds convient aux déploiements qui utilisent le système de fichiers local pour le service Répertoire de fichiers. Notez que l'ajout de serveurs d'applications supplémentaires n'a pas d'impact linéaire sur les performances ou l'évolutivité des serveurs de données. En fait, à l'exception d'une surcharge due aux requêtes utilisateur supplémentaires, l'impact de l'ajout d'hôtes d'application et d'utilisateurs supplémentaires est minime.

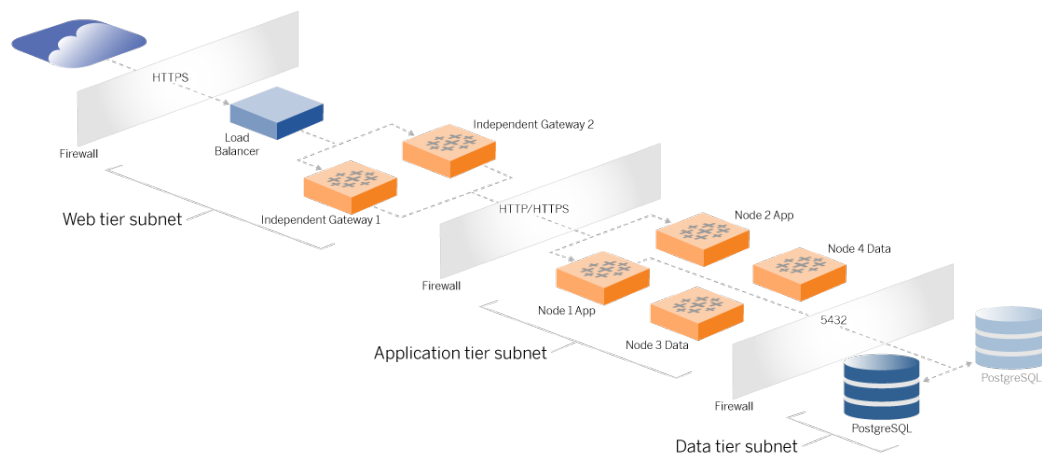
Partie 3 - Préparer le déploiement de Tableau Server en entreprise

La Partie 3 décrit les exigences de préparation de votre infrastructure pour déployer l'architecture de référence Tableau Server. Avant de commencer, nous vous recommandons de consulter Partie 2 - Comprendre l'architecture de référence du déploiement de Tableau Server.

Outre la description des exigences, cette rubrique fournit un exemple d'implémentation de l'architecture de référence dans un environnement AWS. Le reste de ce guide s'appuie sur l'exemple d'architecture de référence AWS démarré dans cette rubrique.

L'un des principes fondamentaux de l'architecture de référence est la standardisation avec les meilleures pratiques de sécurité des centres de données. Plus précisément, l'architecture est conçue pour séparer les services en sous-réseaux de réseau protégés. La communication entre les sous-réseaux est limitée à un protocole et à un trafic de port spécifiques.

Le diagramme suivant illustre la conception du sous-réseau dans l'architecture de référence pour un déploiement sur site ou un déploiement cloud géré par le client. Pour un exemple de déploiement cloud, consultez la section ci-dessous, Exemple : Configurer des sous-réseaux et des groupes de sécurité dans AWS.



Sous-réseaux

Créez trois sous-réseaux :

- Un niveau Web
- Un niveau Application
- Un sous-réseau de données.

Règles de pare-feu/groupe de sécurité

Les onglets ci-dessous décrivent les règles de pare-feu pour chaque niveau du centre de données. Pour les règles de groupe de sécurité spécifiques à AWS, consultez la section plus loin dans cette rubrique.

Niveau Web

Le niveau Web est un sous-réseau DMZ public qui gérera les demandes HTTPS entrantes et transmettra les demandes au niveau Application. Cette conception fournit une couche de défense contre les logiciels malveillants susceptibles de cibler votre entreprise. Le niveau Web bloque l'accès au niveau application/données.

Trafic	Type	Protocole	Plage de ports	Source
Entrant	SSH	TCP	22	Sous-réseau bastion (pour les déploiements cloud)
Entrant	HTTP	TCP	80	Internet (0.0.0.0/0)
Entrant	HTTPS	TCP	443	Internet (0.0.0.0/0)
Sortant	Tout le trafic	Tout	Tout	

Niveau Application

Le sous-réseau Application est l'endroit où réside le déploiement de Tableau Server. Le sous-réseau Application comprend les serveurs d'applications Tableau (Nœud 1 et Nœud 2). Les serveurs d'applications Tableau traitent les demandes des utilisateurs envoyées aux serveurs de données et exécutent la logique métier principale.

Le sous-réseau Application comprend également les serveurs de données Tableau (Nœud 3 et Nœud 4).

Tout le trafic client vers le niveau Application est authentifié au niveau Web. L'accès administratif au sous-réseau Application est authentifié et acheminé via l'hôte bastion.

Trafic	Type	Protocole	Plage de ports	Source
Entrant	SSH	TCP	22	Sous-réseau bastion (pour les déploiements cloud)
Entrant	HTTPS	TCP	443	Sous-réseau de niveau Web

Sortant	Tout le trafic	Tout	Tout	
---------	----------------	------	------	--

Niveau Données

Le serveur de base de données PostgreSQL externe réside dans le sous-réseau de données.

Trafic	Type	Protocole	Plage de ports	Source
Entrant	SSH	TCP	22	Sous-réseau bastion (pour les déploiements cloud)
Entrant	PostgreSQL	TCP	5432	Sous-réseau de niveau Application
Sortant	Tout le trafic	Tout	Tout	

Bastion

La plupart des équipes de sécurité d'entreprise n'autorisent pas la communication directe entre le système administratif sur site et les nœuds déployés dans le cloud. Au lieu de cela, tout le trafic SSH administratif vers les nœuds cloud est transféré via un hôte bastion (également appelé « jump server »). Pour les déploiements cloud, nous recommandons une connexion hôte proxy bastion à toutes les ressources de l'architecture de référence. Il s'agit d'une configuration facultative pour les environnements sur site.

L'hôte bastion authentifie l'accès administratif et autorise uniquement le trafic via le protocole SSH.

Trafic	Type	Protocole	Plage de ports	Source	Destination
--------	------	-----------	----------------	--------	-------------

Entrant	SSH	TCP	22	Adresse IP de l'ordinateur administrateur	
Sortant	SSH	TCP	22		Sous-réseau de niveau Web
Sortant	SSH	TCP	22		Sous-réseau de niveau Appli-cation

Exemple : Configurer des sous-réseaux et des groupes de sécurité dans AWS

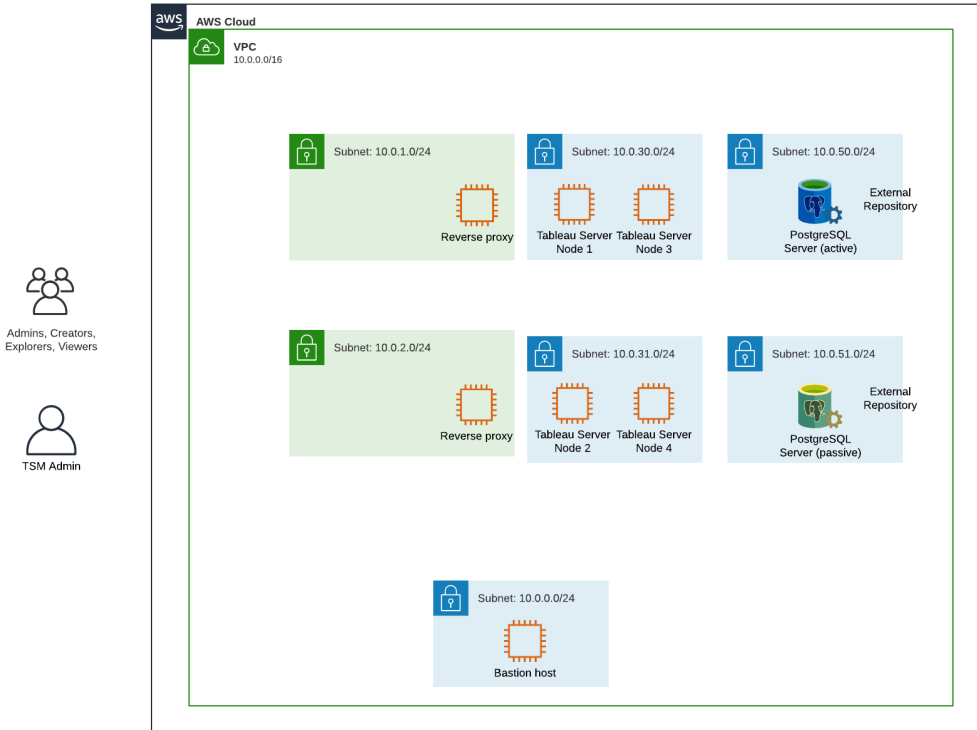
Cette section fournit des procédures détaillées de création et de configuration de l'environnement VPC et réseau pour le déploiement de l'architecture de référence Tableau Server dans AWS.

Les diapositives ci-dessous montrent l'architecture de référence en quatre couches. Au fur et à mesure que vous progressez dans les diapositives, les éléments des composants sont superposés sur la carte de topologie :

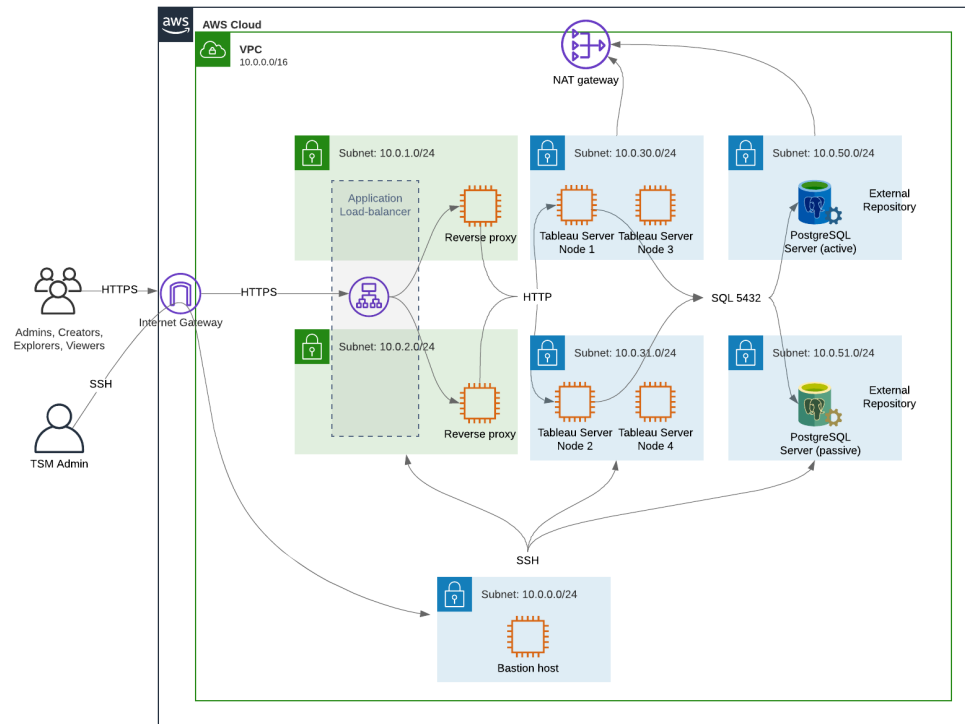
1. Topologie de sous-réseau VPC et instances EC2 : un hôte bastion, deux serveurs proxy inverses, quatre serveurs Tableau et au moins un serveur PostgreSQL.
2. Flux de protocole et connectivité Internet. Tout le trafic entrant est géré via la passerelle Internet AWS. Le trafic vers Internet est acheminé via le NAT.
3. Zones de disponibilité. Le proxy, Tableau Server ainsi que les hôtes PostgreSQL sont déployés uniformément sur les deux zones de disponibilité.
4. Groupes de sécurité. Quatre groupes de sécurité (Public, Privé, Données et Bastion) protègent chaque niveau au niveau du protocole.

Architecture de référence AWS

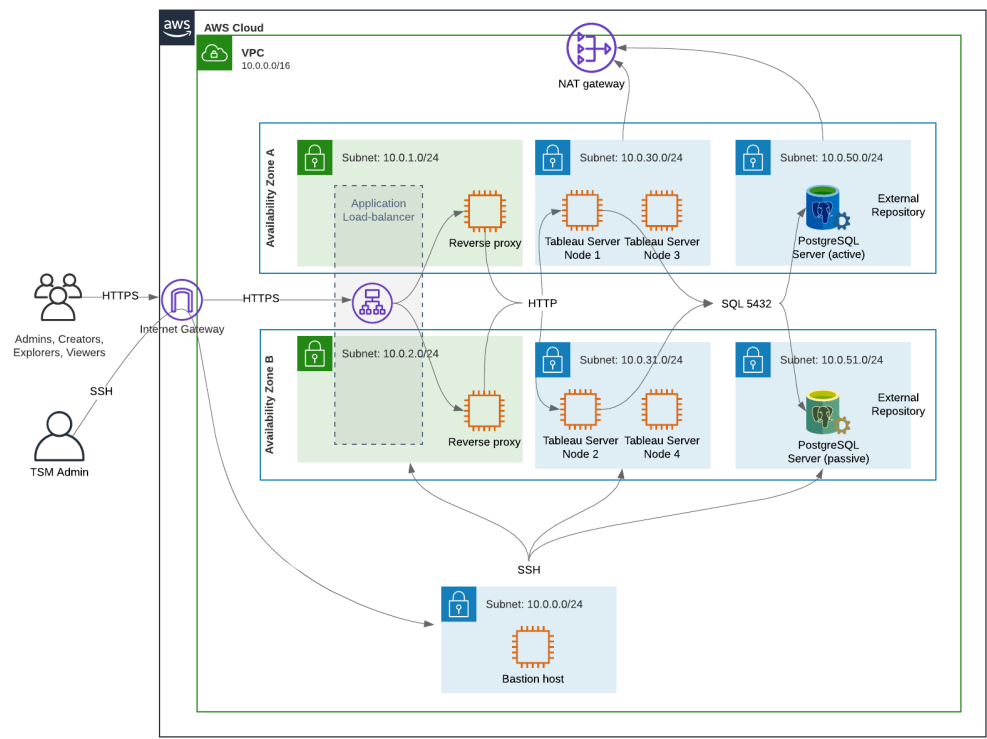
Diapositive 1 : Topologie de sous-réseau VPC et instances EC2



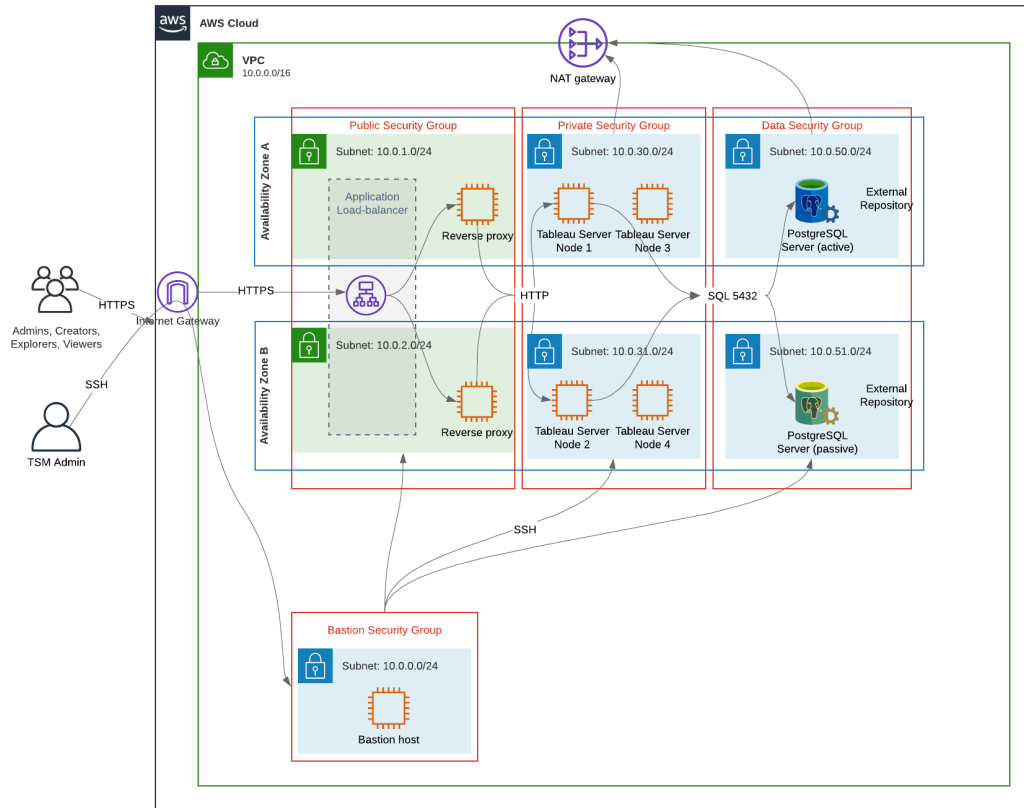
Diapositive 2 : Flux de protocole et connectivité



Diapositive 3 : Zones de disponibilité



Diapositive 4 : Groupes de sécurité



Zones de disponibilité AWS et haute disponibilité

L'architecture de référence telle que présentée dans ce Guide spécifie un déploiement qui assure la disponibilité via la redondance en cas de défaillance d'un seul hôte. Cependant, dans le cas AWS où l'architecture de référence est déployée sur deux zones de disponibilité, la disponibilité est compromise dans le cas très rare où une zone de disponibilité échoue.

Configuration de VPC

Cette section décrit comment :

- Installer et configurer le VPC
- Configurer la connectivité Internet

- Configurer les sous-réseaux
- Créer et configurer des groupes de sécurité

Configurer VPC

La procédure décrite dans cette section correspond à l'interface utilisateur de l'expérience VPC « classique ». Vous pouvez changer l'interface utilisateur sur la vue classique en désactivant la nouvelle expérience VPC dans le coin supérieur gauche du tableau de bord AWS VPC.

Exécutez l'assistant VPC pour créer des sous-réseaux privés et publics par défaut ainsi qu'un routage et une liste ACL réseau par défaut.

1. Avant de configurer un VPC, vous devez créer une adresse IP Elastic. Créez une allocation en utilisant toutes les valeurs par défaut.
2. Exécutez l'assistant VPC > « VPC avec des sous-réseaux publics et privés »
3. Acceptez toutes les valeurs par défaut. À l'exception des éléments suivants :
 - Entrez un nom de VPC.
 - Spécifiez l'ID d'allocation d'adresses IP Elastic.
 - Spécifiez les masques CIDR suivants :
 - CIDR IPv4 du sous-réseau public : 10.0.1.0/24. Renommez ce sous-réseau en `Public-a`.
 - CIDR IPv4 du sous-réseau privé : 10.0.30.0/24. Renommez ce sous-réseau en `Private-a`.
 - Zone de disponibilité : pour les deux sous-réseaux, sélectionnez l'option **a** pour la région dans laquelle vous vous trouvez.

Remarque : pour les besoins de cet exemple, nous utilisons **a** et **b** pour distinguer les zones de disponibilité dans un centre de données AWS donné. Dans AWS, les noms des zones de disponibilité peuvent ne pas correspondre aux exemples présentés ici. Par exemple, certaines zones de disponibilité incluent des zones **c** et **d** dans un centre de données.

4. Cliquez sur **Créer un VPC**.

5. Une fois le VPC créé, créez les sous-réseaux `Public-b`, `Private-b`, `Data` et `Bastion`. Pour créer un sous-réseau, cliquez sur **Sous-réseaux > Créer un sous-réseau**.

- `Public-b` : pour la zone de disponibilité, sélectionnez l'option **b** pour la région dans laquelle vous vous trouvez. Bloc CIDR : 10.0.2.0/24
- `Private-b` : pour la zone de disponibilité, sélectionnez l'option **b** pour la région dans laquelle vous vous trouvez. Bloc CIDR : 10.0.31.0/24
- `Data` : pour la zone de disponibilité, sélectionnez la zone **a** pour la région dans laquelle vous vous trouvez. Bloc CIDR : 10.0.50.0/24. Facultatif : si vous prévoyez de répliquer la base de données externe sur un cluster PostgreSQL, créez un sous-réseau `Data-b` dans la zone de disponibilité **b** avec un bloc CIDR de 10.0.51.0/24.
- `Bastion` : sélectionnez une des deux zones comme zone de disponibilité. Bloc CIDR : 10.0.0.0/24

6. Une fois les sous-réseaux créés, modifiez les tables de routage sur les sous-réseaux `Public` et `Bastion` de manière à utiliser la table de routage configurée pour la passerelle Internet associée (IGW). Modifiez également les sous-réseaux `Privé` et `Données` de manière à utiliser la table de routage configurée pour le traducteur d'adresses réseau (NAT).

- Pour déterminer quelle table de routage est configurée avec l'IGW ou le NAT, cliquez sur **Tables de routage** dans le tableau de bord AWS. Sélectionnez l'une des deux tables de routage pour ouvrir la page de propriétés. Examinez la valeur cible dans **Routes > Destination > 0.0.0.0/0**. La valeur cible différencie le type de route et commencera par la chaîne `igw-` ou `nat-`.
- Pour mettre à jour les tables de routage, **VPC > Sous-réseaux > [subnet_name] > Table de routage > Modifier l'association de table de routage**.

Configurer les groupes de sécurité

L'assistant VPC crée un seul groupe de sécurité que vous n'utiliserez pas. Créez les groupes de sécurité suivants (**Groupes de sécurité > Créer un groupe de sécurité**). Les hôtes EC2 seront installés dans ces groupes sur deux zones de disponibilité, comme montré dans le diagramme de diapositives ci-dessus.

Guide de déploiement de Tableau Server en entreprise

- Créez un nouveau groupe de sécurité : **Privé**. C'est là que les 4 nœuds de Tableau Server seront installés. À un stade ultérieur du processus d'installation, le groupe de sécurité Privé sera associé aux sous-réseaux 10.0.30.0/24 et 10.0.31.0/24.
- Créez un nouveau groupe de sécurité : **Public**. C'est là que les serveurs proxy seront installés. À un stade ultérieur du processus d'installation, le groupe de sécurité Public sera associé aux sous-réseaux 10.0.1.0/24 et 10.0.2.0/24.
- Créez un nouveau groupe de sécurité : **Données**. C'est là que le référentiel externe Tableau PostgreSQL sera installé. À un stade ultérieur du processus d'installation, le groupe de sécurité Données sera associé au sous-réseau 10.0.50.0/24 (et éventuellement 10.0.51.0/24).
- Créez un nouveau groupe de sécurité : **Bastion**. C'est là que vous installerez l'hôte bastion. À un stade ultérieur du processus d'installation, le groupe de sécurité Bastion sera associé aux sous-réseaux et 10.0.0.0/24.

Spécifier les règles de trafic entrant et sortant

Dans AWS, les groupes de sécurité sont analogues à des pare-feu dans un environnement local. Vous devez spécifier le type de trafic (par exemple, http, https, etc.), le protocole (TCP ou UDP) et les ports ou la plage de ports (par exemple 80, 443, etc.) qui sont autorisés à entrer dans le groupe de sécurité et/ou à en sortir. Pour chaque protocole, vous devez également spécifier la destination ou le trafic source.

Règles du groupe de sécurité Public

Règles de trafic entrant			
Type	Protocole	Plage de ports	Source
HTTP	TCP	80	0.0.0.0/0
HTTPS	TCP	443	0.0.0.0/0
SSH	TCP	22	Groupe de sécurité Bastion

Règles de trafic sortant

Type	Protocole	Plage de ports	Destination
Tout le trafic	Tout	Tout	0.0.0.0/0

Règles du groupe de sécurité Privé

Le groupe de sécurité Privé inclut une règle entrante pour autoriser le trafic HTTP à partir du groupe de sécurité Public. Autorisez le trafic HTTP uniquement pendant le processus de déploiement pour vérifier la connectivité. Nous vous recommandons de supprimer la règle HTTP entrante une fois que vous avez terminé de déployer le proxy inverse et de configurer SSL sur Tableau.

Règles de trafic entrant			
Type	Protocole	Plage de ports	Source
HTTP	TCP	80	Groupe de sécurité Public
HTTPS	TCP	443	Groupe de sécurité Public
PostgreSQL	TCP	5432	Groupe de sécurité Données
SSH	TCP	22	Groupe de sécurité Bastion
Tout le trafic	Tout	Tout	Groupe de sécurité Privé

Règles de trafic sortant			
Type	Protocole	Plage de ports	Destination
Tout le trafic	Tout	Tout	0.0.0.0/0
PostgreSQL	TCP	5432	Groupe de sécurité Données
SSH	TCP	22	Groupe de sécurité Bastion

Règles du groupe de sécurité Données

Règles de trafic entrant			
Type	Protocole	Plage de ports	Source
PostgreSQL	TCP	5432	Groupe de sécurité Privé
SSH	TCP	22	Groupe de sécurité Bastion

Règles de trafic sortant			
Type	Protocole	Plage de ports	Destination
Tout le trafic	Tout	Tout	0.0.0.0/0
PostgreSQL	TCP	5432	Groupe de sécurité Privé
SSH	TCP	22	Groupe de sécurité Bastion

Règles du groupe de sécurité de l'hôte Bastion

Règles de trafic entrant			
Type	Protocole	Plage de ports	Source
SSH	TCP	22	Adresse IP et masque réseau de l'ordinateur que vous utiliserez pour vous connecter à AWS (ordinateur administrateur).
SSH	TCP	22	Groupe de sécurité Privé
SSH	TCP	22	Groupe de sécurité Public

Règles de trafic sortant			
--------------------------	--	--	--

Type	Protocole	Plage de ports	Destination
SSH	TCP	22	Adresse IP et masque réseau de l'ordinateur que vous utiliserez pour vous connecter à AWS (ordinateur administrateur).
SSH	TCP	22	Groupe de sécurité Privé
SSH	TCP	22	Groupe de sécurité Public
SSH	TCP	22	Groupe de sécurité Données
HTTPS	TCP	443	0.0.0.0/0 (Facultatif : créez cette règle si vous devez accéder à Internet pour télécharger le logiciel de support sur l'hôte bastion)

Activer l'attribution automatique de l'adresse IP publique

Vous obtenez ainsi une adresse IP pour vous connecter aux serveurs proxy et à l'hôte bastion.

Pour les sous-réseaux Public et Bastion :

1. Sélectionnez le sous-réseau
2. Dans le menu **Actions**, sélectionnez « Modify auto-assign IP settings » (Modifier les paramètres IP d'attribution automatique).
3. Cliquez sur « Enable auto-assign public IPv4 addresses » (Activer l'attribution automatique d'adresses IPv4 publiques).
4. Cliquez sur **Enregistrer**.

Équilibreur de charge

Remarque : si vous effectuez l'installation dans AWS et suivez l'exemple de déploiement de ce guide, vous devez installer et configurer l'équilibreur de charge AWS plus tard dans le processus de déploiement, comme décrit dans la Partie 5 - Configuration du niveau Web.

Pour les déploiements sur site, travaillez avec vos administrateurs réseau pour déployer des équilibreurs de charge afin de prendre en charge le niveau Web de l'architecture de référence :

- Un équilibreur de charge d'application Web qui accepte les requêtes HTTPS des clients Tableau et communique avec les serveurs proxy inverses.
- Serveur proxy inverse :
 - Nous recommandons un minimum de deux serveurs proxy pour la redondance et pour gérer la charge client.
 - Reçoit le trafic HTTPS de l'équilibreur de charge.
 - Prend en charge la session persistante vers l'hôte Tableau.
 - Configurez le proxy pour l'équilibrage de charge à tour de rôle sur chaque Tableau Server exécutant le processus de passerelle.
 - Gère les demandes d'authentification de l'IdP externe.
- Proxy de transfert : Tableau Server a besoin d'un accès à Internet pour les licences et les fonctionnalités de carte. En fonction de votre environnement de proxy de transfert, vous devrez peut-être configurer des listes d'autorisations de proxy de transfert pour les URL de service Tableau. Voir *Communication avec Internet* ([Linux](#)).

Configurer les ordinateurs hôtes

Matériel minimal recommandé

Les recommandations suivantes sont basées sur nos tests de données réelles dans l'architecture de référence.

Serveurs d'applications :

- Processeur : 8 cœurs physiques (16 vCPU),
- RAM : 128 Go (16 Go par cœur physique)
- Espace disque : 100 Go

Serveurs de données :

- Processeur : 8 cœurs physiques (16 vCPU),
- RAM : 128 Go (16 Go par cœur physique)
- Espace disque : 1 To. Si votre déploiement utilise un stockage externe pour le répertoire de fichiers Tableau, vous devrez calculer l'espace disque approprié. Consultez *Installer Tableau Server avec un répertoire de fichiers externe* ([Linux](#)).

Serveurs proxy

- Processeur : 2 cœurs physiques (4 vCPU),
- RAM : 8 Go (4 Go par cœur physique)
- Espace disque : 100 Go

Base de données du référentiel externe

- Processeur : 8 cœurs physiques (16 vCPU),
- RAM : 128 Go (16 Go par cœur physique)
- L'espace disque requis dépend de la charge de vos données et de son impact sur la sauvegarde. Consultez la section *Processus de sauvegarde et de restauration* dans la rubrique *Espace disque requis* ([Linux](#)).

Structure du répertoire

L'architecture de référence recommande d'installer le package Tableau Server et les données dans des emplacements autres que ceux par défaut :

- Installez le package sur `/app/tableau_server`. Créez ce chemin de répertoire avant d'installer le package Tableau Server, puis spécifiez ce chemin lors de l'installation.
- Installez les données Tableau sur `/data/tableau_data`. Ne créez pas ce répertoire avant d'installer Tableau Server. Au lieu de cela, vous devez spécifier le chemin lors de

l'installation, puis le programme d'installation de Tableau créera et autorisera le chemin de manière appropriée.

Consultez Exécuter le package d'installation et initialiser TSM pour plus de détails sur la mise en œuvre.

Exemple : Installer et préparer les ordinateurs hôtes dans AWS

Cette section explique comment installer les hôtes EC2 pour chaque type de serveur dans l'architecture de référence Tableau Server.

L'architecture de référence nécessite huit hôtes :

- Quatre instances pour Tableau Server.
- Deux instances pour serveurs proxy (Apache).
- Une instance pour l'hôte bastion.
- Une ou deux instances de base de données PostgreSQL EC2

Détails de l'instance hôte

Installez les ordinateurs hôtes selon les détails ci-dessous.

Tableau Server

- Amazon Linux 2
- Type d'instance : m5a.8xlarge
- ID du groupe de sécurité : Privé
- Stockage : EBS, 150 Gio, type de volume gp2. Si votre déploiement utilise un stockage externe pour le répertoire de fichiers Tableau, vous devrez calculer l'espace disque approprié. Consultez *Installer Tableau Server avec un répertoire de fichiers externe (Linux)*.
- Réseau : installez deux hôtes EC2 dans chaque sous-réseau privé (10.0.30.0/24 et 10.0.31.0/24).

- Copiez la version de maintenance la plus récente du dernier package rpm de Tableau Server 2021.2 (ou ultérieur) depuis la [page Téléchargements de Tableau](#) sur chaque hôte Tableau.

Hôte Bastion

- Amazon Linux 2
- Type d'instance : t3.micro
- ID du groupe de sécurité : Bastion
- Stockage : EBS, 50 Gio, type de volume gp2
- Réseau : sous-réseau Bastion 10.0.0.0/24

Tableau Server Independent Gateway

- Amazon Linux 2
- Type d'instance : t3.xlarge
- ID du groupe de sécurité : Public
- Stockage : EBS, 100 Gio, type de volume gp2
- Réseau : installez une instance EC2 dans chaque sous-réseau public (10.0.1.0/24 et 10.0.2.0/24)

Hôte PostgreSQL EC2

- Amazon Linux 2
- Type d'instance : r5.xlarge
- ID du groupe de sécurité : Données
- Espace de stockage : l'espace disque requis dépend de la charge de vos données et de son impact sur la sauvegarde. Consultez la section *Processus de sauvegarde et de restauration* dans la rubrique *Espace disque requis* ([Linux](#)).
- Réseau : sous-réseau de données 10.0.50.0/24. (Si vous répliquez PostgreSQL dans un cluster HA, installez le deuxième hôte dans le sous-réseau 10.0.51.0/24)

Vérification : connectivité VPC

Après avoir installé les ordinateurs hôtes, vérifiez la configuration du réseau. Vérifiez la connectivité entre les hôtes en vous connectant avec SSH depuis l'hôte du groupe de

sécurité Bastion aux hôtes de chaque sous-réseau.

Exemple : connexion à l'hôte bastion dans AWS

1. Configurez votre ordinateur administrateur pour ssh-agent. Cela vous permet de vous connecter aux hôtes dans AWS sans placer votre fichier de clé privée sur une instance EC2.

Pour configurer ssh-agent sur un Mac, exécutez la commande suivante :

```
ssh-add -K myPrivateKey.pem ou pour le dernier Mac OS, ssh-add --apple-use-keychain myPrivateKey.pem
```

Pour Windows, consultez la rubrique [Se connecter en toute sécurité aux instances Linux s'exécutant dans un VPC Amazon privé](#).

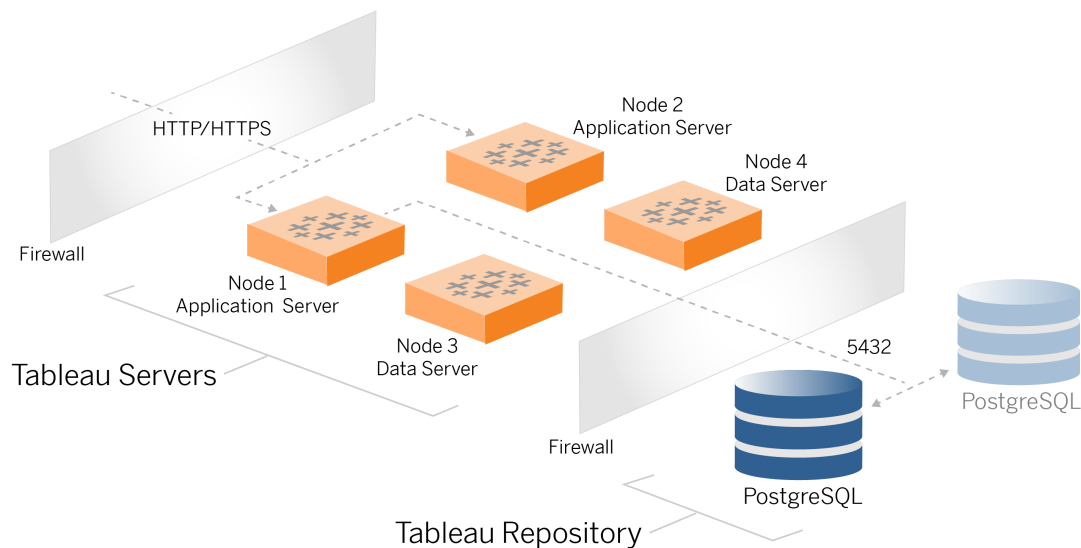
2. Connectez-vous à l'hôte bastion en exécutant la commande suivante :

```
ssh -A ec2-user@<public-IP>
```

3. Vous pouvez ensuite vous connecter à d'autres hôtes sur le VPC à partir de l'hôte bastion, en utilisant l'adresse IP privée, par exemple :

```
ssh -A ec2-user@10.0.1.93
```

Partie 4 - Installer et configurer Tableau Server



Cette rubrique décrit comment terminer l'installation et la configuration du déploiement de base de Tableau Server. La procédure ici se poursuit avec l'exemple d'architecture de référence AWS et Linux.

Les exemples Linux tout au long des procédures d'installation montrent des commandes pour les distributions de type RHEL. Plus précisément, les commandes présentées ici ont été développées avec la distribution Amazon Linux 2. Si vous exécutez la distribution Ubuntu, modifiez les commandes en conséquence.

Avant de commencer

Vous devez préparer et valider votre environnement comme décrit dans la Partie 3 - Préparer le déploiement de Tableau Server en entreprise.

Installer, configurer PostgreSQL et créer une sauvegarde tar

Cette instance PostgreSQL héberge le référentiel externe pour le déploiement de Tableau Server. Vous devez installer et configurer PostgreSQL avant d'installer Tableau.

Vous pouvez exécuter PostgreSQL sur Amazon RDS ou sur une instance EC2. Pour plus d'informations sur les différences entre exécuter le référentiel sur une instance RDS plutôt que sur une instance EC2, consultez *Référentiel externe Tableau Server* ([Linux](#)).

À titre d'exemple, la procédure ci-dessous montre comment installer et configurer Postgres sur une instance Amazon EC2. L'exemple montré ici est une installation et une configuration génériques pour PostgreSQL dans l'architecture de référence. Votre DBA doit optimiser votre déploiement PostgreSQL en fonction de la taille de vos données et de vos besoins de performances.

Exigences : notez que vous devez exécuter PostgreSQL 1.6 et que vous devez installer le module uuid-osp.

Versions de PostgreSQL

Vous devez installer des versions majeures compatibles de PostgreSQL pour le référentiel externe Tableau Server. De plus, les versions mineures doivent également répondre aux exigences minimales.

Versions de Tableau Server	Versions minimum compatibles avec PostgreSQL
2021.2.3 - 2021.2.8	12.6
2021.3.0 - 2021.3.7	
2021.4.0 - 2021.4.3	
2021.2.10 - 2021.2.14	12.8

2021.3.8 - 2021.3.13	
2021.4.4 - 2021.4.8	
2021.2.15 - 2021.2.16	12.10
2021.3.14 - 2021.3.15	
2021.4.9 - 2021.4.10	
2021.2.17 - 2021.2.18	12.11
2021.3.16 - 2021.3.17	
2021.4.11 - 2021.4.12	
2021.3.26	12.15
2021.4.23	
2022.1.0	13.3
2022.1.1 - 2022.1.3	13.4
2022.1.4 - 2022.1.6	13.6
2022.1.7 - 2022.1.16	13.7
2022.3.0 - 2022.3.7	
2023.1.0 - 2023.1.4	
2022.1.17 - 2022.1.19	13.11
2022.3.8 - 2022.3.11	
2023.1.5 - 2023.1.7	

Installer PostgreSQL

Cet exemple de procédure d'installation décrit comment installer PostgreSQL version 13.6.

Connectez-vous à l'hôte EC2 que vous avez créé dans la partie précédente.

1. Exécutez la mise à jour pour appliquer les derniers correctifs au système d'exploitation

Linux :

```
sudo yum update
```

2. Créez et modifiez le fichier `pgdg.repo` dans le chemin d'accès `/etc/yum.repos.d/`.

Remplissez le fichier avec les informations de configuration suivantes :

```
[pgdg13]
name=PostgreSQL 13 for RHEL/CentOS 7 - x86_64
baseurl=
l=https://download.postgresql.org/pub/repos/yum/13/redhat/rhel-
7-x86_64
enabled=1
gpgcheck=0
```

3. Installez Postgres 13.6 :

```
sudo yum install postgresql13-server-13.6-1PGDG.rhel7.x86_64
```

4. Installez le module `uuid-oss` :

```
sudo yum install postgresql13-contrib-13.6-1PGDG.rhel7.x86_64
```

5. Initialisez Postgres :

```
sudo /usr/pgsql-13/bin/postgresql-13-setup initdb
```

Configurer Postgres

Terminez l'installation de base en configurant Postgres :

1. Mettez à jour le fichier de configuration `pg_hba`, `/var/lib/pgsql/13/data/pg_hba.conf`, avec les deux entrées suivantes. Chaque entrée doit inclure le masque des

sous-réseaux sur lesquels vos serveurs Tableau Server s'exécuteront :

```
host all all 10.0.30.0/24 password
```

```
host all all 10.0.31.0/24 password
```

2. Mettez à jour le fichier PostgreSQL, /var/–

lib/pgsql/13/data/postgresql.conf, en ajoutant cette ligne :

```
listen_addresses = '*'
```

3. Configurez de manière à démarrer Postgres au redémarrage :

```
sudo systemctl enable --now postgresql-13
```

4. Définissez le mot de passe du super-utilisateur :

```
sudo su - postgres
```

```
psql -c "alter user postgres with password 'StrongPassword'"
```

Remarque : définissez un mot de passe fort. N'utilisez pas 'StrongPassword' comme indiqué dans l'exemple ici.

```
exit
```

5. Redémarrez Postgres :

```
sudo systemctl restart postgresql-13
```

Effectuer une sauvegarde tar PostgreSQL de l'Étape 1

Créez une sauvegarde tar de la configuration PostgreSQL. La création d'un instantané tar de la configuration actuelle vous fera gagner du temps si vous rencontrez des échecs lors de la

poursuite du déploiement.

Nous appellerons cela la sauvegarde « Étape 1 ».

Sur l'hôte PostgreSQL :

1. Arrêtez l'instance de base de données Postgres :

```
sudo systemctl stop postgresql-13
```

2. Exécutez les commandes suivantes pour créer la sauvegarde tar :

```
sudo su
```

```
cd /var/lib/pgsql
```

```
tar -cvf step1.13.bkp.tar 13
```

```
exit
```

3. Démarrez la base de données Postgres :

```
sudo systemctl start postgresql-13
```

Restaurez l'Étape 1

Restaurez à l'Étape 1 si le nœud initial de Tableau Server échoue lors de l'installation.

1. Sur l'ordinateur exécutant Tableau, exécutez le script obliterate pour supprimer complètement Tableau Server de l'hôte :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./-  
tableau-server-obliterate -a -y -y -y -l
```

2. Restaurez le fichier tar PostgreSQL Étape 1. Sur l'ordinateur exécutant Postgres, exécutez les commandes suivantes :

```
sudo su
```

```
systemctl stop postgresql-13
```

```
cd /var/lib/pgsql  
tar -xvf step1.13.bkp.tar  
systemctl start postgresql-13  
exit
```

Reprenez le processus d'installation au niveau de l'installation du nœud initial de Tableau Server.

Avant l'installation

Si vous déployez Tableau selon l'exemple d'implémentation AWS/Linux décrit dans ce guide, vous pourrez peut-être exécuter le script d'installation automatisée TabDeploy4EDG. Le script TabDeploy4EDG automatise l'exemple d'installation du déploiement Tableau à quatre nœuds qui est décrit dans les procédures qui suivent. Consultez Annexe - Boîte à outils du déploiement AWS.

Installer le nœud initial de Tableau Server

Cette procédure décrit comment installer le nœud initial de Tableau Server tel que défini par l'architecture de référence. A l'exception de l'installation du package et de l'initialisation de TSM, la procédure décrite ici utilise la ligne de commande TSM chaque fois que cela est possible. Outre son indépendance par rapport aux plates-formes, l'utilisation de l'interface en ligne de commande TSM permet une installation plus transparente dans des environnements virtualisés et administrés à distance.

Exécuter le package d'installation et initialiser TSM

Connectez-vous au serveur hôte du Nœud 1.

1. Exécutez la mise à jour pour appliquer les derniers correctifs au système d'exploitation Linux :

```
sudo yum update
```


Guide de déploiement de Tableau Server en entreprise

2. Copiez le package d'installation de la [page Téléchargements de Tableau](#) sur l'ordinateur hôte qui exécutera Tableau Server.

Par exemple, sur un ordinateur fonctionnant sur un système d'exploitation Linux de type RHEL, exécutez

```
wget https://-
downloads.tableau.com/esdalt/2022<version>/tableau-server-<ver-
sion>.rpm
```

où <version> correspond au numéro de version.

3. Téléchargez et installez les dépendances :

```
sudo yum deplist tableau-server-<version>.rpm | awk '/pro-
vider:/ {print $2}' | sort -u | xargs sudo yum -y install
```

4. Créez le chemin d'accès /app/tableau_server dans le répertoire racine :

```
sudo mkdir -p /app/tableau_server
```

5. Exécutez le programme d'installation et spécifiez le chemin d'installation /app/-tableau_server. Par exemple, sur un système d'exploitation Linux de type RHEL, exécutez :

```
sudo rpm -i --prefix /app/tableau_server tableau-server-<ver-
sion>.x86_64.rpm
```

6. Passez au répertoire /app/tableau_server/packages/scripts.<version_code>/ et exécutez le script initialize-tsm qui s'y trouve :

```
sudo ./initialize-tsm -d /data/tableau_data --accepteula
```

7. Une fois l'initialisation terminée, quittez l'interpréteur de commandes :

```
exit
```

Activer et enregistrer Tableau Server

1. Connectez-vous au serveur hôte du Nœud 1.
2. Fournissez la ou les clés produit Tableau Server à cette étape. Exécutez la commande suivante pour chaque clé produit que vous avez achetée :

```
tsm licenses activate -k <product key>
```

3. Créez un fichier d'enregistrement json au format indiqué ici :

```
{  
  "zip" : "97403",  
  "country" : "USA",  
  "city" : "Springfield",  
  "last_name" : "Simpson",  
  "industry" : "Energy",  
  "eula" : "yes",  
  "title" : "Safety Inspection Engineer",  
  "company_employees" : "100",  
  "phone" : "5558675309",  
  "company" : "Example",  
  "state" : "OR",  
  "opt_in" : "true",  
  "department" : "Engineering",  
  "first_name" : "Homer",  
  "email" : "homer@example.com"  
}
```

4. Après avoir enregistré les modifications apportées au fichier, transmettez-le avec l'option `--file` pour enregistrer Tableau Server :

```
tsm register --file path_to_registration_file.json
```

Configurer le magasin d'identités

Remarque : si votre déploiement utilise un stockage externe pour le magasin de fichiers Tableau, vous devrez activer le répertoire de fichiers externe avant de configurer le magasin d'identités. Consultez *Installer Tableau Server avec un répertoire de fichiers externe (Linux)*.

L'architecture de référence par défaut utilise un magasin d'identités local. Configurez l'hôte initial avec le magasin d'identités local en transmettant le fichier `config.json` avec la commande `tsm settings import`.

Importez le fichier `config.json` en fonction de votre système d'exploitation :

Le fichier `config.json` est inclus dans le chemin `scripts.<version>` du répertoire (par exemple `scripts.20204.21.0217.1203`) et est formaté pour configurer le magasin d'identités.

Exécutez la commande suivante pour importer le fichier `config.json` :

```
tsm settings import -f /app/tableau_server/packages/scripts.<version_code>/config.json
```

Configurer Postgres externe

1. Créez un fichier json de base de données externe avec les paramètres de configuration suivants :

```
{
  "flavor": "generic",
  "masterUsername": "postgres",
  "host": "<instance ip address>",
  "port": 5432
}
```

2. Après avoir enregistré les modifications apportées au fichier, transmettez le fichier avec la commande suivante :

```
tsm topology external-services repository enable -f <file-name>.json --no-ssl
```

Vous serez invité à saisir le mot de passe du nom d'utilisateur principal Postgres.

L'option `--no-ssl` configure Tableau de manière à utiliser SSL/TLS uniquement lorsque le serveur Postgres est configuré pour SSL/TLS. Si Postgres n'est pas configuré pour SSL/TLS, la connexion n'est pas chiffrée. La Partie 6 - Configuration après l'installation décrit comment activer SSL/TLS pour la connexion Postgres une fois que vous avez terminé la première phase de déploiement.

3. Appliquez les modifications.

Exécutez cette commande pour appliquer les modifications et redémarrez Tableau Server :

```
tsm pending-changes apply
```

4. Supprimez le fichier de configuration que vous avez utilisé à l'Étape 1.

Terminer l'installation du Nœud 1

1. Une fois Tableau Server installé, vous devez initialiser le serveur.

Exécutez la commande suivante :

```
tsm initialize --start-server --request-timeout 1800
```

2. Une fois l'initialisation terminée, vous devez créer un compte d'administrateur Tableau Server.

À la différence du compte d'ordinateur que vous utilisez pour installer et gérer les composants du système d'exploitation TSM, le compte administrateur Tableau Server est un compte d'application utilisé pour créer des utilisateurs, des projets et des sites

Tableau Server. L'administrateur Tableau Server applique également des autorisations aux ressources Tableau. Exécutez la commande suivante pour créer le compte administrateur initial. Dans l'exemple suivant, l'utilisateur est appelé `tableau-admin`:

```
tabcmd initialuser --server http://localhost --  
username "tableau-admin"
```

Tabcmd vous demandera de définir un mot de passe pour cet utilisateur.

Vérification : Configuration du Nœud 1

1. Exécutez la commande suivante pour vérifier que les services TSM sont en cours d'exécution :

```
tsm status -v
```

Tableau doit renvoyer les éléments suivants :

```
external:  
Status: RUNNING  
'Tableau Server Repository 0' is running (Active Repository).  
node1: localhost  
Status: RUNNING  
'Tableau Server Gateway 0' is running.  
'Tableau Server Application Server 0' is running.  
'Tableau Server Interactive Microservice Container 0' is running.  
'MessageBus Microservice 0' is running.  
'Relationship Query Microservice 0' is running.  
'Tableau Server VizQL Server 0' is running.  
...
```

Tous les services seront répertoriés.

2. Exécutez la commande suivante pour vérifier que le site administratif de Tableau est en cours d'exécution :

```
curl localhost
```

Les premières lignes devraient afficher Vizportal html, dans ce style :

```
<!DOCTYPE html>
<html xmlns:ng="" xmlns:tb="">
<head ng-csp>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="initial-scale=1, maximum-sca-
le=2, width=device-width, height=device-height, viewport-fit-
t=cover">
<meta name="format-detection" content="telephone=no">
<meta name="vizportal-config ...
```

Effectuer des sauvegardes tar de l'Étape 2

Après avoir vérifié l'installation initiale, effectuez deux sauvegardes tar :

- PostgreSQL
- Nœud initial de Tableau (Nœud 1)

Dans la plupart des cas, vous pouvez récupérer votre installation du nœud initial en restaurant ces fichiers tar. La restauration des fichiers tar est beaucoup plus rapide que la réinstallation et la réinitialisation du nœud initial.

Créer des fichiers tar de l'Étape 2

1. Sur le nœud initial de Tableau, arrêtez Tableau :

```
tsm stop
```

Attendez que Tableau s'arrête avant de passer à l'étape suivante.

2. Sur l'hôte PostgreSQL, arrêtez l'instance de base de données Postgres :

Guide de déploiement de Tableau Server en entreprise

```
sudo systemctl stop postgresql-13
```

3. Exécutez les commandes suivantes pour créer la sauvegarde tar :

```
sudo su  
  
cd /var/lib/pgsql  
  
tar -cvf step2.13.bkp.tar 13  
  
exit
```

4. Vérifiez que le fichier tar Postgres est créé avec les autorisations root :

```
sudo ls -al /var/lib/pgsql
```

5. Sur l'hôte Tableau, arrêtez les services administratifs de Tableau :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./s-  
top-administrative-services
```

6. Exécutez les commandes suivantes pour créer la sauvegarde tar :

```
cd /data  
  
sudo tar -cvf step2.tableau_data.bkp.tar tableau_data
```

7. Sur l'hôte Postgres, démarrez la base de données Postgres :

```
sudo systemctl start postgresql-13
```

8. Démarrez les services administratifs de Tableau :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./s-  
tart-administrative-services
```

9. Exécutez la commande `tsm status` pour surveiller l'état de TSM avant de redémarrer.

Dans la plupart des cas, la commande commencera par renvoyer un état **DEGRADED** ou **ERROR**. Attendez quelques minutes puis exécutez à nouveau la commande. Si

l'état ERROR ou DEGRADED est renvoyé, continuez d'attendre. N'essayez pas de démarrer TSM tant que l'état STOPPED n'est pas renvoyé. Ensuite, exécutez la commande suivante :

```
tsm start
```

Restaurer l'Étape 2

Ce processus restaure le Nœud 1 de Tableau et l'instance Postgres à l'Étape 2. Après avoir restauré cette étape, vous pouvez alors redéployer les nœuds Tableau restants.

1. Arrêtez le service tsm sur l'hôte Tableau initial (Nœud 1) :

```
tsm stop
```

2. Arrêtez les services administratifs de Tableau sur tous les nœuds du déploiement de Tableau Server. Exécutez la commande suivante sur chaque nœud, dans l'ordre (Nœud 1, Nœud 2, puis Nœud 3) :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./s-  
top-administrative-services
```

3. Une fois les services Tableau arrêtés, restaurez le fichier tar PostgreSQL Étape 2. Sur l'ordinateur exécutant Postgres, exécutez les commandes suivantes :

- ```
sudo su

systemctl stop postgresql-13

cd /var/lib/pgsql

tar -xvf step2.13.bkp.tar

systemctl start postgresql-13

exit
```



4. Restaurez le fichier tar de Tableau Étape 2. Sur l'hôte Tableau initial, exécutez les commandes suivantes :

```
cd /data

sudo rm -rf tableau_data

sudo tar -xvf step2.tableau_data.bkp.tar
```

5. Sur l'ordinateur Nœud 1 de Tableau, supprimez les fichiers suivants :

- `sudo rm /data/tableau_data/-data/tabsvc/appzookeeper/0/version-2/currentEpoch`
- `sudo rm /data/tableau_data/-data/tabsvc/appzookeeper/0/version-2/acceptedEpoch`
- `sudo rm /data/tableau_data/-data/tabsvc/tabadminagent/0/servicestate.json`

6. Démarrez les services administratifs de Tableau :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./s-
tart-administrative-services
```

7. Rechargez les fichiers sysmctl de Tableau, puis exécutez `start-administrative-services` de nouveau :

```
sudo su -l tableau -c "systemctl --user daemon-reload"

sudo /app/tableau_server/packages/scripts.<version_code>/./s-
tart-administrative-services
```

8. Sur le Nœud 1, exécutez la commande `tsm status` pour surveiller l'état de TSM avant de redémarrer.

Dans certains cas, vous obtiendrez une erreur, `Cannot connect to server....`

Cette erreur se produit car le service `tabadmincontroller` n'a pas redémarré. Continuer à

exécuter régulièrement `tsm status`. Si cette erreur ne disparaît pas après 10 minutes, réexécutez la commande `start-administrative-services`.

Après quelques instants, la commande `tsm status` renverra le statut `DEGRADED`, puis `ERROR`. Ne démarrez pas TSM tant que l'état `STOPPED` n'est pas renvoyé. Ensuite, exécutez la commande suivante :

```
tsm start
```

Reprenez le processus d'installation pour installer Tableau Server sur les nœuds restants.

## Installer Tableau Server sur les nœuds restants

Pour poursuivre le déploiement, copiez le programme d'installation de Tableau sur chaque nœud.

### Présentation de la configuration des nœuds

Cette section décrit le processus de configuration des Nœuds 2-4. Les sections suivantes fournissent des procédures détaillées de configuration et de validation pour chaque étape.

L'installation des Nœuds Tableau Server 2-4 nécessite de générer, copier et référencer un fichier bootstrap pendant l'installation des nœuds.

Pour générer le fichier bootstrap, vous devez exécuter une commande TSM sur le nœud initial. Vous copiez ensuite le fichier bootstrap sur le nœud cible, où vous l'exécutez dans le cadre de l'initialisation du nœud.

Le contenu json suivant montre un exemple de fichier bootstrap. (Le certificat et les valeurs liées à la cryptographie ont été tronqués pour rendre l'exemple de fichier plus facile à lire.)

```
{
 "initialBootstrapSettings" : {
 "certificate" : "-----BEGIN CERTIFICATE-----\r\...\r\n-----END
```

## Guide de déploiement de Tableau Server en entreprise

```
CERTIFICATE-----",
 "port" : 8850,
 "configurationName" : "tabsvc",
 "clusterId" : "tabsvc-clusterid",
 "cryptoKeyStore" : "zs7OzgAAAAIAAABAAAAA...w==",
 "toksCryptoKeystore" : "LS0tLS1CRUdJTtIBUT00tLS0tCjM5MDBh...L",
 "sessionCookieMaxAge" : 7200,
 "nodeId" : "node1",
 "machineAddress" : "ip-10-0-1-93.us-west-1.compute.internal",
 "cryptoEnabled" : true,
 "sessionCookieUser" : "tsm-bootstrap-user",
 "sessionCookieValue" : "eyJ-
jdHkiOiJKVlQiLCJlbmMiOiJBMTI4Q0JDLUhQ...",
 "sessionCookieName" : "AUTH_COOKIE"
 }
}
```

Le fichier bootstrap inclut une validation basée sur la connexion pour authentifier le Nœud 1 et crée un canal chiffré pour le processus d'amorçage. La session d'amorçage est limitée dans le temps, et la configuration et la validation des nœuds prennent du temps. Prévoyez de créer et de copier de nouveaux fichiers bootstrap lorsque vous configurez les nœuds.

Après avoir exécuté le fichier d'amorçage, vous vous connectez au nœud Tableau Server initial et configurez les processus pour le nouveau nœud. Une fois la configuration des nœuds terminée, vous devez appliquer les modifications et redémarrer le nœud initial. Le nouveau nœud est configuré et démarré. Au fur et à mesure que vous ajoutez des nœuds, la configuration et le redémarrage du déploiement prennent ensuite plus de temps.

Les exemples Linux tout au long des procédures d'installation montrent des commandes pour les distributions de type RHEL. Si vous exécutez la distribution Ubuntu, modifiez les commandes en conséquence.

1. Exécutez la mise à jour pour appliquer les derniers correctifs au système d'exploitation Linux :

```
sudo yum update
```

2. Téléchargez et installez les dépendances :

```
sudo yum deplist tableau-server-<version>.rpm | awk '/pro-
vider:/ {print $2}' | sort -u | xargs sudo yum -y install
```

3. Créez le chemin d'accès `/app/tableau_server` dans le répertoire racine :

```
sudo mkdir -p /app/tableau_server
```

4. Exécutez le programme d'installation et spécifiez le chemin d'installation `/app/tableau_server`. Par exemple, sur un système d'exploitation Linux de type RHEL, exécutez :

```
sudo rpm -i --prefix /app/tableau_server tableau-server-<ver-
sion>.x86_64.rpm
```

## Générer, copier et utiliser le fichier d'amorçage pour initialiser TSM

La procédure suivante décrit comment générer, copier et utiliser un fichier d'amorçage lors de l'initialisation de TSM sur un autre nœud. Dans cet exemple, le fichier d'amorçage est appelé `boot.json`.

Dans cet exemple, les ordinateurs hôtes s'exécutent dans AWS, où les hôtes EC2 exécutent Amazon Linux 2.

1. Connectez-vous au nœud initial (Nœud 1) et exécutez la commande suivante :

```
tsm topology nodes get-bootstrap-file --file boot.json
```

2. Copiez le fichier d'amorçage sur le Nœud 2.

```
scp boot.json ec2-user@10.0.31.83:/home/ec2-user/
```

## Guide de déploiement de Tableau Server en entreprise

3. Connectez-vous au Nœud 2 et basculez vers le répertoire des scripts de Tableau Server :

```
cd /app/tableau_server/packages/scripts.<version_number>
```

4. Exécutez la commande `initialize-tsm` et référez le fichier `bootstrap` :

```
sudo ./initialize-tsm -d /data/tableau_data -b /home/ec2-user/-
bootstrap.json --accepteula
```

5. Une fois que la commande `initialize-tsm` a été exécutée, supprimez `bootstrap.json`, puis quittez la session ou déconnectez-vous.

## Configurer les processus

Vous devez configurer le cluster Tableau Server sur le nœud où le contrôleur d'administration Tableau Server (contrôleur TSM) s'exécute. Le contrôleur TSM s'exécute sur le nœud initial.

### Process Status

The real-time status of processes running in Tableau Server.

| Process                | Node 1 | Node 2 | Node 3 | Node 4 | External Node |
|------------------------|--------|--------|--------|--------|---------------|
| Cluster Controller     | ✓      | ✓      | ✓      | ✓      |               |
| Gateway                | ✓      | ✓      |        |        |               |
| Application Server     | ✓      | ✓      |        |        |               |
| VizQL Server           | ✓✓     | ✓✓     |        |        |               |
| Cache Server           | ✓✓     | ✓✓     |        |        |               |
| Search & Browse        | ✓      | ✓      |        |        |               |
| Backgrounder           |        |        | ✓✓✓✓   | ✓✓✓✓   |               |
| Data Server            | ✓✓     | ✓✓     |        |        |               |
| Data Engine            | ✓      | ✓      | ✓      | ✓      |               |
| File Store             |        |        | ✓      | ✓      |               |
| Repository             |        |        |        |        | E             |
| Tableau Prep Conductor |        |        | ✓      | ✓      |               |
| Metrics                | ✓      |        |        |        |               |

✓ Active
⌛ Busy
✓ Passive
⚠ Unlicensed
✗ Down
E External
□ Status unavailable

## Configurer le Nœud 2

1. Après avoir initialisé TSM à l'aide du fichier bootstrap sur le Nœud 2, connectez-vous au nœud initial.
2. Sur le nœud initial (node1), exécutez les commandes suivantes pour configurer les processus sur le Nœud 2 :

```

tsm topology set-process -n node2 -pr clustercontroller -c 1
tsm topology set-process -n node2 -pr gateway -c 1
tsm topology set-process -n node2 -pr vizportal -c 1
tsm topology set-process -n node2 -pr vizqlserver -c 2
tsm topology set-process -n node2 -pr cacheserver -c 2
tsm topology set-process -n node2 -pr searchserver -c 1
tsm topology set-process -n node2 -pr dataserver -c 2

```

## Guide de déploiement de Tableau Server en entreprise

```
tsm topology set-process -n node2 -pr clientfileservice -c 1
tsm topology set-process -n node2 -pr tdsservice -c 1
tsm topology set-process -n node2 -pr collections -c 1
tsm topology set-process -n node2 -pr contentexploration -c 1
```

Si vous installez la version 2022.1 ou une version ultérieure, ajoutez également le serveur d'indexation et de recherche :

```
tsm topology set-process -n node2 -pr indexandsearchserver -c 1
```

Si vous installez la version 2023.3 ou une version ultérieure, incluez uniquement le serveur d'indexation et de recherche. N'ajoutez pas le service Rechercher et parcourir (searchserver)

3. Vérifiez la configuration avant de l'appliquer. Exécutez la commande suivante :

```
tsm pending-changes list
```

4. Après avoir vérifié que vos modifications sont dans la liste en attente (la liste en attente comprendra également d'autres services), appliquez les modifications :

```
tsm pending-changes apply
```

Vous devez redémarrer après une modification. La configuration et le redémarrage prendront un certain temps.

5. Vérifiez la configuration du Nœud 2. Exécutez la commande suivante :

```
tsm status -v
```

## Configurer le Nœud 3

Initialisez TSM à l'aide du processus d'amorçage sur le Nœud 3, puis exécutez les commandes `tsm topology set-process` ci-dessous.

Un avertissement du service de coordination s'affiche chaque fois que vous définissez un processus. Vous pouvez ignorer cet avertissement lorsque vous définissez les processus.

1. Après avoir initialisé TSM à l'aide du fichier bootstrap sur le Nœud 3, connectez-vous au nœud initial (`node1`) et exécutez les commandes suivantes pour configurer les processus :

```
tsm topology set-process -n node3 -pr clustercontroller -c 1
tsm topology set-process -n node3 -pr clientfileservice -c 1
tsm topology set-process -n node3 -pr backgrounder -c 4
tsm topology set-process -n node3 -pr filestore -c 1
```

Si vous installez la version 2022.1 ou une version ultérieure, ajoutez également le serveur d'indexation et de recherche :

```
tsm topology set-process -n node3 -pr indexandsearchserver -c 1
```

2. Vérifiez la configuration avant de l'appliquer. Exécutez la commande suivante :

```
tsm pending-changes list
```

3. Après avoir vérifié que vos modifications sont dans la liste en attente (la liste en attente comprendra d'autres services qui sont configurés automatiquement), appliquez les modifications :

```
tsm pending-changes apply --ignore-warnings
```

Vous devez redémarrer après une modification. La configuration et le redémarrage prendront un certain temps.

4. Vérifiez la configuration en exécutant la commande suivante :

```
tsm status -v
```

## Déployer l'ensemble de service de coordination sur les Nœuds 1 à 3

Pour le déploiement à quatre nœuds de l'architecture de référence standard, exécutez la procédure suivante :



1. Exécutez les commandes suivantes sur le Nœud 1 :

```
tsm stop
tsm topology deploy-coordination-service -n node1,node2,node3
```

Le processus comprend un redémarrage de TSM, ce qui demandera un certain temps.

2. Une fois le service de coordination déployé, démarrez TSM :

```
tsm start
```

## Effectuer des sauvegardes tar de l'Étape 3

Après avoir vérifié l'installation, effectuez quatre sauvegardes tar :

- PostgreSQL
- Nœud initial de Tableau (Nœud 1)
- Tableau Nœud 2
- Tableau Nœud 3

## Créer des fichiers tar de l'Étape 3

1. Sur le nœud initial de Tableau, arrêtez Tableau :

```
tsm stop
```

2. Une fois TSM arrêté, arrêtez les services administratifs Tableau sur chaque nœud. Exécutez la commande suivante sur chaque nœud, dans l'ordre (Nœud 1, Nœud 2, puis Nœud 3) :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./stop-administrative-services
```

3. Sur l'hôte PostgreSQL, arrêtez l'instance de base de données Postgres :

```
sudo systemctl stop postgresql-12
```

4. Exécutez les commandes suivantes pour créer la sauvegarde tar :

```
sudo su

cd /var/lib/pgsql

tar -cvf step3.12.bkp.tar 12

exit
```

5. Vérifiez que le fichier tar Postgres est créé avec les autorisations root :

```
sudo ls -al /var/lib/pgsql
```

6. Sur l'hôte Postgres, démarrez la base de données Postgres :

```
sudo systemctl start postgresql-12
```

7. Créez la sauvegarde tar sur le Nœud 1, Nœud 2 et Nœud 3. Exécutez les commandes suivantes sur chaque nœud :

- ```
cd /data
```
- ```
sudo tar -cvf step3.tableau_data.bkp.tar tableau_data
```
- Vérifiez que le fichier tar Tableau est créé avec des autorisations root :

```
ls -al
```

8. Démarrez les services administratifs Tableau sur chaque nœud dans l'ordre (Nœud 1, Nœud 2, puis Nœud 3) :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./s-
tart-administrative-services
```

9. Exécutez la commande `tsm status` pour surveiller l'état de TSM avant de redémarrer.

Dans la plupart des cas, la commande renverra un état DEGRADED, puis ERROR. Attendez quelques instants et exécutez à nouveau la commande. Si l'état ERROR ou DEGRADED est renvoyé, continuez d'attendre. N'essayez pas de démarrer TSM tant que l'état STOPPED n'est pas renvoyé. Ensuite, exécutez la commande suivante :

```
tsm start
```

## Restaurer l'Étape 3

Ce processus restaure le Nœud 1, le Nœud 2 et le Nœud 3 de Tableau. Il restaure également l'instance Postgres à l'Étape 3. Après avoir restauré cette étape, vous pouvez ensuite déployer le service de coordination, le Nœud 4, puis les configurations du nœud final.

1. Arrêtez le service tsm sur l'hôte Tableau initial (Nœud 1) :

```
tsm stop
```

2. Une fois TSM arrêté, arrêtez les services administratifs de Tableau sur le Nœud 1, le Nœud 2 et le Nœud 3. Exécutez la commande suivante sur chaque nœud :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./s-
top-administrative-services
```

3. Restaurer le fichier tar PostgreSQL Étape 3. Sur l'ordinateur exécutant Postgres, exécutez les commandes suivantes :

```
sudo su

systemctl stop postgresql-12

cd /var/lib/pgsql

tar -xvf step3.12.bkp.tar

systemctl start postgresql-12

exit
```

4. Restaurez le fichier tar de l'Étape 3 de Tableau sur le Nœud 1, le Nœud 2 et le Nœud 3. Exécutez les commandes suivantes sur chaque nœud Tableau :

```
cd /data

sudo rm -rf tableau_data

sudo tar -xvf step3.tableau_data.bkp.tar
```

5. Sur l'ordinateur Nœud 1 de Tableau, supprimez les fichiers suivants :

- `sudo rm /data/tableau_data/-data/tabsvc/appzookeeper/1/version-2/currentEpoch`
- `sudo rm /data/tableau_data/-data/tabsvc/appzookeeper/1/version-2/acceptedEpoch`
- `sudo rm /data/tableau_data/-data/tabsvc/tabadminagent/0/servicestate.json`

Si l'interpréteur de commandes renvoie une erreur « fichier introuvable », vous devrez peut-être modifier le nom du chemin pour incrémenter le nombre `<n>` dans cette section du chemin : `.../appzookeeper/<n>/version-2/...`

6. Redémarrez les services administratifs sur le Nœud 1, Nœud 2 et le Nœud 3. Exécutez les commandes suivantes sur chaque nœud :

```
sudo /app/tableau_server/packages/scripts.<version_code>/./start-administrative-services

sudo su -l tableau -c "systemctl --user daemon-reload"

sudo /app/tableau_server/packages/scripts.<version_code>/./start-administrative-services
```

7. Sur le Nœud 1, exécutez la commande `tsm status` pour surveiller l'état de TSM avant de redémarrer.

Dans certains cas, vous obtiendrez une erreur, `Cannot connect to server....`

Cette erreur se produit car le service `tabadmincontroller` n'a pas redémarré. Continuer à exécuter régulièrement `tsm status`. Si cette erreur ne disparaît pas après 10 minutes, réexécutez la commande `start-administrative-services`.

Après quelques instants, la commande `tsm status` renverra le statut `DEGRADED`, puis `ERROR`. Ne démarrez pas TSM tant que l'état `STOPPED` n'est pas renvoyé.

Ensuite, exécutez la commande suivante :

```
tsm start
```

Reprenez le processus d'installation pour déployer le service de coordination sur les Nœuds 1 à 3.

## Configurer le Nœud 4

Le processus de configuration du Nœud 4 est le même que celui du Nœud 3.

Définissez les mêmes processus que ceux définis pour le Nœud 3, en exécutant le même ensemble de commandes que celui indiqué ci-dessus, mais en spécifiant `node4` dans les commandes au lieu de `node3`.

Comme pour la vérification du Nœud 3, vérifiez la configuration du Nœud 4 en exécutant `tsm status -v`.

Avant de continuer, attendez que le processus Répertoire de fichiers sur le Nœud 4 ait terminé la synchronisation. L'état du service Répertoire de fichiers renvoie `is synchronizing` jusqu'à ce qu'il ait terminé. Lorsque l'état du service Répertoire de fichiers renvoie `is running`, vous pouvez continuer.

## Configuration et vérification du processus final

La dernière étape du processus de configuration consiste à supprimer les processus redondants du Nœud 1.

1. Connectez-vous au nœud initial (`node1`).
2. Désactivez le répertoire de fichiers sur le Nœud 1. Cela entraînera un avertissement concernant la suppression du répertoire de fichiers d'un contrôleur colocalisé. Vous pouvez ignorer l'avertissement. Exécutez la commande suivante :

```
tsm topology filestore decommission -n node1
```

3. Lorsque le répertoire de fichiers est mis hors service, exécutez la commande suivante pour supprimer le processus d'arrière-plan du Nœud 1 :

```
tsm topology set-process -n node1 -pr backgrounder -c 0
```

4. Vérifiez la configuration avant de l'appliquer. Exécutez la commande suivante :

```
tsm pending-changes list
```

5. Après avoir vérifié que vos modifications figurent dans la liste en attente, appliquez les modifications :

```
tsm pending-changes apply
```

Vous devez redémarrer après une modification. La configuration et le redémarrage prendront un certain temps.

6. Vérifiez la configuration :

```
tsm status -v.
```

Avant de continuer, attendez que le processus Répertoire de fichiers sur le Nœud 4 ait terminé la synchronisation. L'état du service Répertoire de fichiers renvoie `is synchronizing` jusqu'à ce qu'il ait terminé. Lorsque l'état du service Répertoire de fichiers renvoie `is running`, vous pouvez continuer.

# Effectuer une sauvegarde

Une restauration complète de Tableau Server nécessite un portefeuille de sauvegardes comprenant trois composants :

- Un fichier de sauvegarde du référentiel et des données de répertoire de fichiers. Ce fichier est généré par la commande `tsm maintenance backup`.
- Un fichier d'exportation de topologie et de configuration. Ce fichier est généré par la commande `tsm settings export`.
- Certificat d'authentification, clé et fichiers keytab.

Pour une description complète du processus de sauvegarde et de restauration, consultez cette rubrique de Tableau Server : *Effectuer une sauvegarde et une restauration complètes de Tableau Server (Linux)*.

À ce stade de votre déploiement, tous les fichiers et ressources pertinents requis pour une restauration complète sont inclus avec l'exécution des commandes `tsm maintenance backup` et `tsm settings export`.

1. Exécutez la commande suivante pour exporter les paramètres de configuration et de topologie vers un fichier appelé `ts_settings_backup.json`.

```
tsm settings export -f ts_settings_backup.json
```

2. Exécutez la commande suivante pour créer une sauvegarde du référentiel et des données du répertoire de fichiers dans un fichier nommé `ts_backup-<yyyy-mm-dd>.tsbak`. Ignorez l'avertissement indiquant que le répertoire de fichiers n'est pas sur le nœud de contrôleur.

```
tsm maintenance backup -f ts_backup -d --skip-compression
```

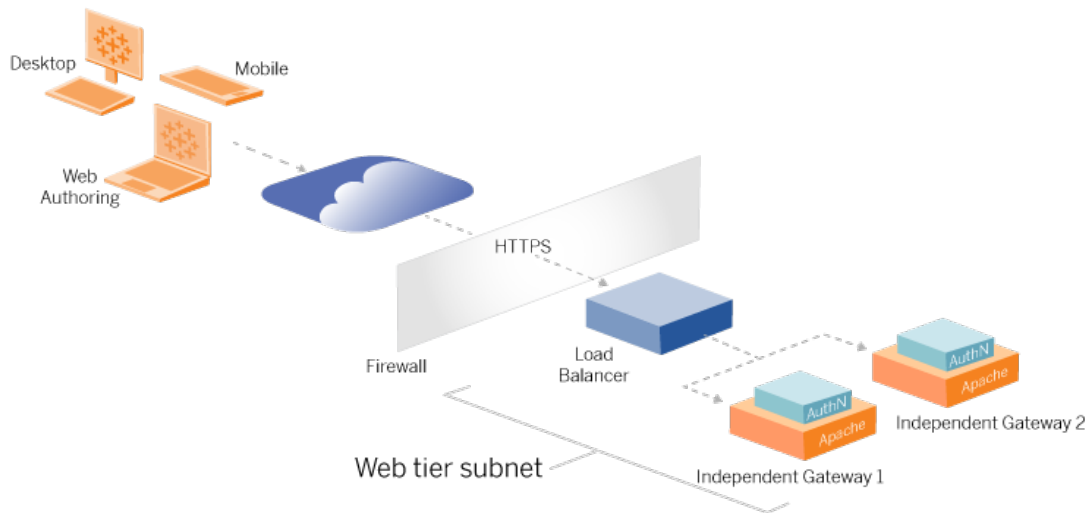
Emplacement du fichier de sauvegarde :

```
/data/tableau_data/data/tabsvc/files/backups/
```

3. Copiez les deux fichiers et enregistrez-les sur une ressource de stockage différente qui n'est pas partagée par votre déploiement Tableau Server.



# Partie 5 - Configuration du niveau Web



Le niveau Web de l'architecture de référence doit inclure les composants suivants :

- Un équilibreur de charge d'application Web qui accepte les requêtes HTTPS des clients Tableau et communique avec les serveurs proxy inverses.
- Serveur proxy inverse :
  - Nous vous recommandons de déployer Tableau Server Independent Gateway.
  - Nous recommandons un minimum de deux serveurs proxy pour la redondance et pour gérer la charge client.
  - Reçoit le trafic HTTPS de l'équilibreur de charge.
  - Prend en charge la session persistante vers l'hôte Tableau.
  - Configurez le proxy pour l'équilibrage de charge à tour de rôle sur chaque Tableau Server exécutant le processus de passerelle.
  - Gère les demandes d'authentification de l'IdP externe.
- Proxy de transfert : Tableau Server a besoin d'un accès à Internet pour les licences et les fonctionnalités de carte. Vous devez configurer des listes d'autorisations de proxy

de transfert pour les URL de service Tableau. Voir *Communication avec Internet (Linux)*.

- Tout le trafic lié au client peut être chiffré via HTTPS :
  - Équilibreur de charge de client vers application
  - Équilibreur de charge d'application vers serveurs proxy inverses
  - Serveur proxy vers Tableau Server
  - Gestionnaire d'authentification s'exécutant sur le proxy inverse vers l'IdP
  - Tableau Server vers l'IdP

## Tableau Server Independent Gateway

Tableau Server version 2022.1 a introduit Tableau Server Independent Gateway. Independent Gateway est une instance autonome du processus Tableau Gateway qui sert de proxy inverse compatible avec Tableau.

Independent Gateway prend en charge l'équilibrage de charge circulaire simple vers les serveurs principaux Tableau Server. Par contre, Independent Gateway n'est pas destiné à servir d'équilibreur de charge pour les applications d'entreprise. Nous vous recommandons d'exécuter Independent Gateway derrière un équilibreur de charge pour les applications de classe entreprise.

Independent Gateway nécessite une licence de Advanced Management.

## Authentification et autorisation

L'architecture de référence par défaut spécifie l'installation de Tableau Server avec configuration de l'authentification locale. Dans ce modèle, les clients doivent se connecter à Tableau Server pour être authentifiés par le processus d'authentification local natif de Tableau Server. Nous vous déconseillons d'utiliser cette méthode d'authentification dans l'architecture de référence. En effet, le scénario nécessite que les clients non authentifiés communiquent avec le niveau Application, ce qui constitue un risque de sécurité.

Au lieu de cela, nous vous recommandons de configurer un fournisseur d'identité externe de niveau entreprise associé à un module AuthN pour pré-authentifier tout le trafic vers le niveau

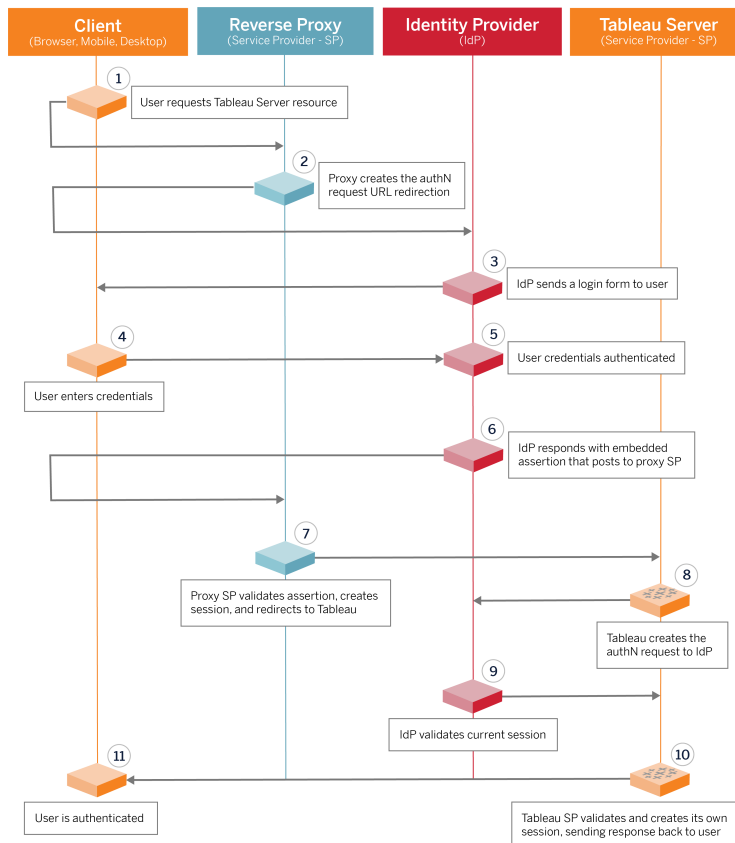
Application. Lorsqu'il est configuré avec un IdP externe, le processus d'authentification local natif de Tableau Server n'est pas utilisé. Tableau Server autorise l'accès aux ressources dans le déploiement une fois que l'IdP a authentifié les utilisateurs.

## Pré-authentification avec un module AuthN

Dans l'exemple documenté dans ce guide, l'authentification unique SAML est configurée, mais le processus de pré-authentification peut être configuré avec la plupart des fournisseurs d'identité externes et un module AuthN.

Dans l'architecture de référence, le proxy inverse est configuré de manière à créer une session d'authentification client avec l'IdP avant de transmettre ces demandes par proxy à Tableau Server. Nous appelons ce processus la phase de *pré-authentification*. Le proxy inverse redirige vers Tableau Server uniquement les sessions cliente authentifiées. Tableau Server crée ensuite une session, vérifie l'authentification de la session avec l'IdP, puis renvoie la demande du client.

Le schéma suivant montre le processus détaillé étape par étape de pré-authentification et d'authentification avec configuration d'un module AuthN. Le proxy inverse peut être une solution tierce générique ou Tableau Server Independent Gateway :



## Présentation de la configuration

Cette section présente le processus de configuration du niveau Web. Vérifiez la connectivité après chaque étape :

1. Configurez deux serveurs proxy inverses pour fournir un accès HTTP à Tableau Server.
2. Configurez la logique d'équilibrage de charge avec des sessions persistantes sur les serveurs proxy pour vous connecter à chaque instance de Tableau Server exécutant le processus de passerelle.
3. Configurez l'équilibrage de charge des applications avec des sessions persistantes au niveau de la passerelle Internet pour transférer les demandes aux serveurs proxy inverses.

4. Configurez l'authentification avec un IdP externe. Vous pouvez configurer l'authentification SSO ou SAML en installant un gestionnaire d'authentification sur les serveurs proxy inverses. Le module AuthN gère la négociation d'authentification entre l'IdP externe et votre déploiement Tableau. Tableau agira également en tant que fournisseur de services IdP et authentifiera les utilisateurs avec l'IdP.
5. Pour s'authentifier avec Tableau Desktop dans ce déploiement, vos clients doivent exécuter Tableau Desktop 2021.2.1 ou une version ultérieure.

## Exemple de configuration de niveau Web avec Tableau Server Independent Gateway

Le reste de cette rubrique fournit une procédure de bout en bout qui décrit comment implémenter un niveau Web dans l'architecture AWS de référence avec Tableau Server Independent Gateway. Pour un exemple de configuration utilisant Apache comme proxy inverse, voir Annexe - Niveau Web avec exemple de déploiement Apache.

L'exemple de configuration est composé des composants suivants :

- Équilibreur de charge d'application AWS
- Tableau Server Independent Gateway
- Module d'authentification Mellon
- IdP Okta
- Authentification SAML

**Remarque :** l'exemple de configuration de niveau Web présenté dans cette section comprend des procédures détaillées pour le déploiement de logiciels et de services tiers. Nous avons tout mis en œuvre pour vérifier et documenter les procédures permettant d'activer le scénario de niveau Web. Il peut toutefois arriver que le logiciel tiers change ou que votre scénario diffère de l'architecture de référence décrite ici. Veuillez vous référer à la documentation du fournisseur tiers pour les détails de configuration et l'assistance.

Les exemples Linux tout au long de cette section montrent des commandes pour les distributions de type RHEL. Plus précisément, les commandes présentées ici ont été développées avec la distribution Amazon Linux 2. Si vous exécutez la distribution Ubuntu, modifiez les commandes en conséquence.

Le déploiement du niveau Web dans cet exemple suit une procédure de configuration et de vérification par étapes. La configuration du niveau Web principal comprend les étapes suivantes pour activer HTTP entre Tableau et Internet. La passerelle indépendante est exécutée et configurée pour un proxy inverse/équilibre de charge derrière l'équilibreur de charge d'application AWS :

1. Préparer l'environnement
2. Installer la passerelle indépendante
3. Configurer le serveur de passerelle indépendante
4. Configurer l'équilibreur de charge d'application AWS

Une fois que vous avez configuré le niveau Web et vérifié la connectivité avec Tableau vérifiée, configurez l'authentification avec un fournisseur externe.

## Préparer l'environnement

Effectuez les tâches suivantes avant de déployer la passerelle indépendante.

1. Modifications apportées au groupe de sécurité AWS. Configurez le groupe de sécurité public pour autoriser le trafic de maintenance de la passerelle indépendante entrant (TCP 21319) à partir du groupe de sécurité privé.
2. Installez la version 22.1.1 (ou ultérieure) sur un cluster Tableau Server à quatre nœuds comme documenté dans Partie 4 - Installer et configurer Tableau Server.
3. Configurez les deux instances EC2 du proxy dans le groupe de sécurité publique comme expliqué dans Configurer les ordinateurs hôtes.

## Installer la passerelle indépendante

Tableau Server Independent Gateway nécessite une licence de Advanced Management.

## Guide de déploiement de Tableau Server en entreprise

Le déploiement de Tableau Server Independent Gateway consiste à installer et à exécuter le package .rpm, puis à configurer l'état initial. La procédure décrite dans ce guide fournit des conseils prescriptifs pour le déploiement dans l'architecture de référence.

Si votre déploiement diffère de l'architecture de référence, consultez la documentation principale de Tableau Server, *Installer Tableau Server avec Independent Gateway* ([Linux](#)).

**Important :** la configuration de la passerelle indépendante peut être un processus sujet aux erreurs. Il est très difficile de résoudre les problèmes de configuration sur deux instances de serveurs de passerelle indépendante. Pour cette raison, nous vous recommandons de configurer un serveur de passerelle indépendante à la fois. Après avoir configuré le premier serveur et vérifié qu'il fonctionne correctement, configurez le deuxième serveur de passerelle indépendante.

Même si vous configurez chaque serveur de passerelle indépendante séparément, exécutez cette procédure d'installation sur les deux instances EC2 que vous avez installées dans le groupe de sécurité public :

1. Exécutez la mise à jour pour appliquer les derniers correctifs au système d'exploitation Linux :

```
sudo yum update
```

2. Si Apache est installé, supprimez-le :

```
sudo yum remove httpd
```

3. Copiez le package d'installation d'Independent Gateway version 2022.1.1 de la [page Téléchargements de Tableau](#) sur l'ordinateur hôte qui exécutera Tableau Server.

Par exemple, sur un ordinateur fonctionnant sur un système d'exploitation Linux de type RHEL, exécutez

```
wget https://-
downloads.tableau.com/esdalt/2022<version>/tableau-server-tsig-
<version>.x86_64.rpm
```

4. Exécutez le programme d'installation. Par exemple, sur un système d'exploitation Linux de type RHEL, exécutez :

```
sudo yum install <tableau-tsig-version>.x86_64.rpm
```

5. Passez au répertoire `/opt/tableau/tableau_tsig/-`  
`packages/scripts.<version_code>/` et exécutez le script `initialize-`  
`tsig` qui s'y trouve. En plus de `--accepteula`, vous devez inclure la plage  
d'adresses IP des sous-réseaux sur lesquels le déploiement de Tableau Server est en  
cours d'exécution. Utilisez l'option `-c` pour spécifier la plage IP. L'exemple ci-dessous  
montre la commande avec les exemples de sous-réseaux AWS spécifiés :

```
sudo ./initialize-tsig --accepteula -c "ip 10.0.30.0/24
10.0.31.0/24"
```

6. Une fois l'initialisation terminée, ouvrez le fichier `tsighk-auth.conf` et copiez le  
secret d'authentification dans le fichier. Vous devrez soumettre ce code pour chaque  
instance de passerelle indépendante dans le cadre de la configuration principale de  
Tableau Server :

```
sudo less /var/opt/tableau/tableau_tsig/config/tsighk-auth.conf
```

7. Après avoir exécuté les étapes précédentes sur les deux instances de la passerelle  
indépendante, préparez le fichier de configuration `tsig.json`. Le fichier de confi-  
guration consiste en un tableau « `independentGateways` ». Le tableau contient des  
objets de configuration qui définissent chacun les détails de connexion pour une ins-  
tance de passerelle indépendante.



Copiez le fichier JSON suivant et personnalisez-le en fonction de votre environnement de déploiement. L'exemple ici montre un fichier pour un exemple d'architecture de référence AWS.

L'exemple de fichier JSON ci-dessous inclut uniquement les informations de connexion pour une passerelle indépendante. Plus tard dans le processus, vous inclurez les informations de connexion pour le deuxième serveur de passerelle indépendante.

Enregistrez le fichier sous `tsig.json` pour les procédures qui suivent.

```
{
 "independentGateways": [
 {
 "id": "ip-10-0-1-169.ec2.internal",
 "host": "ip-10-0-1-169.ec2.internal",
 "port": "21319",
 "protocol" : "http",
 "authsecret": "13660-27118-29070-25482-9518-22453"
 }
]
}
```

- "id" - Le nom DNS privé de l'instance AWS EC2 exécutant la passerelle indépendante.
- "host" - identique à "id".
- "port" - Le port de maintenance, "21319" par défaut.
- "protocol" - Le protocole pour le trafic client. Laissez-le sous la forme `http` pour la configuration initiale.
- "authsecret" - Le secret que vous avez copié à l'étape précédente.

## Passerelle indépendante : connexion directe ou par relais

Avant de continuer, vous devez décider du schéma de connexion à configurer dans votre déploiement : connexion directe ou relais. Chaque option est brièvement décrite ici, ainsi que les points de données de décision pertinents.

**Connexion par relais** : vous avez la possibilité de configurer la passerelle indépendante pour relayer la communication client sur un seul port vers le processus de passerelle sur Tableau Server. Nous appelons cette communication une connexion *par relais* :

- Le processus de relais entraîne un saut supplémentaire depuis la passerelle indépendante vers le processus principal de la passerelle Tableau Server. Le saut supplémentaire dégrade les performances par rapport à la configuration de connexion directe.
- TLS est pris en charge en mode relais. Toutes les communications en mode relais sont limitées à un seul protocole (HTTP ou HTTPS) et peuvent donc être cryptées et authentifiées avec TLS.

**Connexion directe** : la passerelle indépendante peut communiquer directement avec les processus principaux de Tableau Server sur plusieurs ports. Nous appelons cette communication une connexion *directe* :

- Étant donné que la connexion est directe avec le serveur principal Tableau Server, les performances du client sont nettement améliorées par rapport à l'option de connexion par relais.
- Nécessite l'ouverture de plus de 16 ports des sous-réseaux publics aux sous-réseaux privés pour une communication de processus directe depuis la passerelle indépendante vers les ordinateurs Tableau Server.
- TLS n'est pas encore pris en charge sur les processus depuis la passerelle indépendante vers Tableau Server.

## Configurer la connexion par relais

Pour exécuter TLS entre Tableau Server et la passerelle indépendante, vous devez utiliser une connexion relais dans la configuration. Les exemples de scénarios dans l'EDG sont configurés avec une connexion relais.

1. Copiez `tsig.json` sur le Nœud 1 de votre déploiement Tableau Server.
2. Sur le Nœud 1, exécutez les commandes suivantes pour activer la passerelle indépendante.

```
tsm stop
tsm configuration set -k gateway.tsig.proxy_tls_optional -v
none
tsm pending-changes apply
tsm topology external-services gateway enable -c tsig.json
tsm start
```

## Configurer la connexion directe

Étant donné que la connexion directe ne prend pas en charge TLS, nous vous recommandons de configurer la connexion directe uniquement si vous êtes en mesure de sécuriser tout le trafic réseau par d'autres moyens. Pour exécuter TLS entre Tableau Server et la passerelle indépendante, vous devez utiliser une connexion relais dans la configuration. Les exemples de scénarios dans l'EDG sont configurés avec une connexion relais.

Si vous configurez la passerelle indépendante pour une connexion directe à Tableau Server, vous devez activer la configuration pour déclencher la communication. Une fois que Tableau Server communique avec la passerelle indépendante, les cibles de protocole sont établies. Vous devez ensuite récupérer `proxy_targets.csv` à partir de l'ordinateur de la passerelle indépendante et ouvrir les ports correspondants des groupes de sécurité Public vers Privé dans AWS.

1. Copiez `tsig.json` sur le Nœud 1 de votre déploiement Tableau Server.
2. Sur le Nœud 1, exécutez les commandes suivantes pour activer la passerelle indépendante.

```
tsm stop
tsm topology external-services gateway enable -c tsig.json
tsm start
```

3. Sur l'ordinateur de la passerelle indépendante, exécutez la commande suivante pour afficher les ports utilisés par le cluster Tableau Server :

```
less /var/opt/tableau/tableau_tsig/config/httpd/proxy_target-
gets.csv
```

4. Configurez les groupes de sécurité AWS. Ajoutez les ports TCP répertoriés dans `proxy_targets.csv` pour autoriser la communication entre le groupe de sécurité Public et le groupe de sécurité Privé.

Nous vous recommandons d'automatiser la configuration d'entrée des ports car les ports peuvent changer si le déploiement de Tableau Server change. L'ajout de nœuds ou la reconfiguration des processus sur le déploiement de Tableau Server déclenchera des modifications dans l'accès au port requis par Independent Gateway.

## Vérification : configuration de la topologie de base

Vous devriez pouvoir accéder à la page d'administration de Tableau Server en accédant à `http://<gateway-public-IP-address>`.

Si la page de connexion de Tableau Server ne se charge pas ou si Tableau Server ne démarre pas, suivez ces étapes de dépannage :

Réseau :

- Vérifiez la connectivité entre le déploiement de Tableau et l'instance de passerelle indépendante en exécutant la commande suivante `wget` depuis le Nœud 1 de Tableau Server : `wget http://<Adresse IP interne de la passerelle indépendante>:21319`, par exemple :

```
wget http://ip-10-0-1-38:21319
```

Si la connexion est refusée ou échoue, vérifiez que le groupe de sécurité public est configuré pour autoriser le trafic de maintenance de la passerelle indépendante (TCP 21319) du groupe de sécurité privé.

Si le groupe de sécurité est configuré correctement, vérifiez que vous avez spécifié les bonnes adresses IP ou plages IP lors de l'initialisation de la passerelle indépendante. Vous pouvez afficher et modifier cette configuration dans le fichier

## Guide de déploiement de Tableau Server en entreprise

`environment.bash` à l'emplacement `/etc/opt/tableau/tableau_tsig/environment.bash`. Si vous apportez une modification à ce fichier, redémarrez le service `tsig-http` comme décrit ci-dessous.

Sur l'hôte Proxy 1 :

1. Remplacez le fichier `httpd.conf` par le fichier stub de la passerelle indépendante :

```
cp /var/opt/tableau/tableau_tsig/config/httpd.conf.stub
/var/opt/tableau/tableau_tsig/config/httpd.conf
```

2. Redémarrez `tsig-httpd` comme première étape de dépannage :

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
/
```

Sur le Nœud 1 de Tableau

- Revérifiez le fichier `tsig.json`. Si vous trouvez des erreurs, corrigez-les, puis exécutez `tsm topology external-services gateway update -c tsig.json`.
- Si vous exécutez une connexion directe, vérifiez que les ports TCP répertoriés dans `proxy_targets.csv` sont configurés en tant que ports d'entrée des groupes de sécurité Public vers Privé.

## Configurer l'équilibreur de charge d'application AWS

Configurez l'équilibreur de charge en tant qu'écouteur HTTP. La procédure ici décrit comment ajouter un équilibreur de charge dans AWS.

### Étape 1 : Créer un groupe cible

Un groupe cible est une configuration AWS qui définit les instances EC2 exécutant vos serveurs proxy. Ce sont les cibles pour le trafic provenant du LBS.

1. EC2>**Groupes cibles** > **Créer un groupe cible**
2. Sur la page Créer :

- Entrez un nom de groupe cible, par exemple `TG-internal-HTTP`
  - Type de cible : Instances
  - Protocole : HTTP
  - Port : 80
  - VPC : Sélectionnez votre VPC
  - Sous **Contrôles d'intégrité** > **Paramètres avancés des contrôles d'intégrité**  
> **Codes de réussite**, ajoutez la liste des codes pour lire : 200, 303.
  - Cliquez sur **Créer**
3. Sélectionnez le groupe cible que vous venez de créer, puis cliquez sur l'onglet **Cibles** :
- Cliquez sur **Modifier**.
  - Sélectionnez les instances EC2 (ou une seule instance si vous en configurez une à la fois) qui exécutent l'application proxy, puis cliquez sur **Ajouter à l'enregistrement**.
  - Cliquez sur **Enregistrer**.

## Étape 2 : Lancer l'assistant d'équilibrage de charge

1. EC2 > **Équilibreurs de charge** > **Créer un équilibreur de charge**
2. Sur la page « Sélectionner le type d'équilibreur de charge », créez un équilibreur de charge d'application.

**Remarque** : l'interface utilisateur qui s'affiche pour configurer l'équilibreur de charge peut présenter des différences selon les centres de données AWS. La procédure ci-dessous, « Configuration de l'assistant », correspond à l'assistant de configuration AWS qui commence par l'**Étape 1 Configurer l'équilibreur de charge**.

Si votre centre de données affiche toutes les configurations sur une seule page qui inclut un bouton **Créer un équilibreur de charge** en bas de la page, suivez la procédure « Configuration d'une seule page » ci-dessous.

## Configuration de l'assistant

1. Page **Configurer l'équilibreur de charge** :

- Précisez le nom
- Schéma : face à Internet (par défaut)
- Type d'adresse IP : ipv4 (par défaut)
- Écouteurs (écouteurs et routage) :
  - a. Laissez l'écouteur HTTP par défaut
  - b. Cliquez sur **Ajouter un écouteur** et ajoutez `HTTPS : 443`
- VPC : sélectionnez le VPC où vous avez tout installé
- Zones de disponibilité :
  - Sélectionnez **a** et **b** comme vos régions de centre de données
  - Dans chaque liste de sélection déroulante correspondante, sélectionnez le sous-réseau Public (où résident vos serveurs proxy).
- Cliquez sur : **Configurer les paramètres de sécurité**

2. Page **Configurer les paramètres de sécurité**

- Téléchargez votre certificat SSL public.
- Cliquez sur **Suivant : Configurer des groupes de sécurité**.

3. Page **Configurer les groupes de sécurité** :

- Sélectionnez le groupe de sécurité Public. Si le groupe de sécurité par défaut est sélectionné, effacez cette sélection.
- Cliquez sur **Suivant : Configurer le routage**.

4. Page **Configurer le routage**

- Groupe cible : Groupe cible existant.
- Nom : sélectionnez le groupe cible que vous avez créé précédemment
- Cliquez sur **Suivant : Enregistrer les cibles**.

5. Page **Enregistrer les cibles**

- Les deux instances de serveur proxy que vous avez configurées précédemment doivent s'afficher.
- Cliquez sur **Suivant : Vérifier**.

6. Page **Révision**

Cliquez sur **Créer**.

## Configuration d'une seule page

### Configuration de base

- Précisez le nom
- Schéma : face à Internet (par défaut)
- Type d'adresse IP : ipv4 (par défaut)

### Mappages réseau

- VPC : sélectionnez le VPC où vous avez tout installé
- Mappages :
  - Sélectionnez les zones de disponibilité **a** et **b** (ou comparables) comme vos régions de centres de données
  - Dans chaque liste de sélection déroulante correspondante, sélectionnez le sous-réseau Public (où résident vos serveurs proxy).

### Groupes de sécurité

Sélectionnez le groupe de sécurité Public. Si le groupe de sécurité par défaut est sélectionné, effacez cette sélection.

### Écouteurs et routage

- Laissez l'écouteur HTTP par défaut. Dans **Action par défaut**, spécifiez le groupe cible que vous avez précédemment configuré.
- Cliquez sur **Ajouter un écouteur** et ajoutez **HTTPS : 443**. Dans **Action par défaut**, spécifiez le groupe cible que vous avez précédemment configuré.

### Paramètres d'écoute sécurisés

- Téléchargez votre certificat SSL public.

Cliquez sur **Créer un équilibreur de charge**.



## Étape 3 : Activer la persistance

1. Une fois l'équilibreur de charge créé, vous devez activer la persistance sur le groupe cible.
  - Ouvrez la page Groupe cible AWS (**EC2 > Équilibreurs de charge > Groupes cibles**), sélectionnez l'instance d'équilibreur de charge cible que vous venez de configurer. Dans le menu **Actions**, sélectionnez **Modifier les attributs**.
  - Sur la page **Modifier les attributs**, sélectionnez **Persistance**, spécifiez une durée `1 day`, puis **Enregistrer les modifications**.
2. Dans l'équilibreur de charge, activez la persistance sur l'écouteur HTTP. Sélectionnez l'équilibreur de charge que vous venez de configurer, puis cliquez sur l'onglet **Écouteurs**.
  - Pour **HTTP:80**, cliquez sur **Afficher/modifier les règles**. Dans la page **Règles** résultante, cliquez sur l'icône de modification (une fois en haut de la page, puis à nouveau près de la règle) pour modifier la règle. Supprimez la règle THEN existante et remplacez-la en cliquant sur **Ajouter une action > Transférer vers....** Dans la configuration THEN résultante, spécifiez le même groupe cible que vous avez créé. Sous Persistance au niveau du groupe, activez la persistance et définissez la durée sur 1 jour. Enregistrez le paramètre puis cliquez sur **Mettre à jour**.

## Étape 4 : Définir le délai d'inactivité sur l'équilibreur de charge

Sur l'équilibreur de charge, mettez à jour le délai d'inactivité sur 400 secondes.

Sélectionnez l'équilibreur de charge que vous avez configuré pour ce déploiement, puis cliquez sur **Actions > Modifier les attributs**. Définissez le **Délai d'inactivité** sur 400 secondes, puis cliquez sur **Enregistrer**.

## Étape 5 : Vérifier la connectivité LBS

Ouvrez la page de l'équilibreur de charge AWS (**EC2 > Équilibreurs de charge**), puis sélectionnez l'instance d'équilibreur de charge que vous venez de configurer.

Sous **Description**, copiez le nom DNS et collez-le dans un navigateur pour accéder à la page de connexion Tableau Server.

Si vous obtenez une erreur de niveau 500, vous devrez probablement redémarrer vos serveurs proxy.

## Mettre à jour le DNS avec l'URL publique de Tableau

Utilisez le nom de zone DNS de votre domaine dans la description de l'équilibreur de charge AWS pour créer une valeur CNAME dans votre DNS. Le trafic vers votre URL (tableau.example.com) doit être envoyé au nom DNS public AWS.

## Vérifier la connectivité

Une fois vos mises à jour DNS terminées, vous devriez pouvoir accéder à la page de connexion Tableau Server en saisissant votre URL publique, par exemple `https://tableau.example.com`.

## Exemple de configuration d'authentification : SAML avec IdP externe

L'exemple suivant décrit comment installer et configurer SAML avec un IdP Okta et un module d'authentification Mellon pour un déploiement Tableau exécuté dans l'architecture de référence AWS.

Cet exemple reprend de la section précédente et suppose que vous configurez une passerelle indépendante à la fois.

L'exemple décrit comment configurer Tableau Server et la passerelle indépendante sur HTTP. Okta enverra la demande à l'équilibreur de charge AWS via HTTPS, mais tout le trafic interne transitera via HTTP. Lors de la configuration de ce scénario, tenez compte des protocoles HTTP et HTTPS lors de la définition des chaînes d'URL.

Cet exemple utilise Mellon comme module de fournisseur de services de pré-authentification sur les serveurs de la passerelle indépendante. Cette configuration garantit que seul le trafic authentifié se connecte à Tableau Server, qui fait également office de fournisseur de services avec l'IdP Okta. Par conséquent, vous devez configurer deux applications d'IdP : une pour le fournisseur de services Mellon et une pour le fournisseur de services Tableau.

## Créer un compte d'administrateur Tableau

Une erreur courante lors de la configuration de SAML est d'oublier de créer un compte administrateur sur Tableau Server avant d'activer l'authentification unique.

La première étape consiste à créer un compte sur Tableau Server avec un rôle d'administrateur de serveur. Dans l'exemple de scénario Okta, le nom d'utilisateur doit utiliser le format d'une adresse e-mail valide, par exemple `user@example.com`. Vous devez définir un mot de passe pour cet utilisateur, mais le mot de passe ne sera pas utilisé une fois SAML configuré.

## Configurer l'application de pré-authentification Okta

Le scénario de bout en bout décrit dans cette section nécessite deux applications Okta :

- Demande de pré-autorisation Okta
- Application Okta Tableau Server

Chacune de ces applications est associée à différentes métadonnées que vous devrez configurer sur le serveur proxy inverse et Tableau Server, respectivement.

Cette procédure décrit comment créer et configurer l'application de pré-authentification Okta. Plus loin dans cette rubrique, vous créerez l'application Okta Tableau Server. Pour un test gratuit de compte Okta avec un nombre limité d'utilisateurs, consultez la [page Web des développeurs Okta](#).

Créez une intégration d'application SAML pour le fournisseur de services de pré-authentification Mellon.

1. Ouvrez le tableau de bord d'administration d'Okta > **Applications** > **Créer une intégration d'application**.
2. Dans la page **Créer une nouvelle intégration d'application**, sélectionnez **SAML 2.0**, puis cliquez sur **Suivant**.
3. Dans l'onglet **Paramètres généraux**, saisissez un nom d'application, par exemple `Tableau Pre-Auth`, puis cliquez sur **Suivant**.
4. Dans l'onglet **Configurer SAML** :
  - URL d'authentification unique (SSO). Le dernier élément du chemin dans l'URL d'authentification unique est appelé `MellonEndpointPath` dans le fichier de configuration `mellon.conf` présenté plus loin dans cette procédure. Vous pouvez spécifier le point de terminaison de votre choix. Dans cet exemple, `sso` est le point de terminaison. Le dernier élément, `postResponse`, est requis :  
`https://tableau.example.com/sso/postResponse`.
  - Décochez la case : **Use this for Recipient URL and Destination URL** (À utiliser comme URL du destinataire et URL de destination).
  - URL du destinataire : identique à l'URL SSO, mais avec HTTP. Par exemple, `http://tableau.example.com/sso/postResponse`.
  - URL de destination : identique à l'URL du destinataire, mais avec HTTP. Par exemple, `http://tableau.example.com/sso/postResponse`.
  - URI d'audience (ID d'entité SP). Par exemple, `https://-tableau.example.com`.
  - Format d'identification du nom : `EmailAddress`
  - Nom d'utilisateur de l'application : `Email`
  - Déclarations d'attributs : Nom = `mail` ; Format du nom = `Unspecified` ; Valeur = `user.email`.

Cliquez sur **Suivant**.

5. Dans l'onglet **Commentaires**, sélectionnez :
  - **Je suis un client Okta ajoutant une application interne**
  - **Il s'agit d'une application interne que nous avons créée**
  - Cliquez sur **Terminer**.

6. Créez le fichier de métadonnées IdP de pré-authentification :

- Dans Okta : **Applications > Applications** > Votre nouvelle application (par exemple `Tableau Pre-Auth`) > **Connexion**
- À côté de **Certificats de signature SAML**, cliquez sur **Afficher les instructions de configuration SAML**.
- Dans la page **Comment configurer SAML 2.0 pour<pre-auth> Application**, faites défiler jusqu'à la section **Facultatif, Fournir les métadonnées d'IDP suivantes à votre fournisseur de SP**.
- Copiez le contenu du champ XML et enregistrez-le dans un fichier appelé `pre-auth_idp_metadata.xml`.

7. (Facultatif) Configurez l'authentification multifacteur :

- Dans Okta : **Applications > Applications** > Votre nouvelle application (par exemple `Tableau Pre-Auth`) > **Connexion**
- Sous **Stratégie de connexion**, cliquez sur **Ajouter une règle**.
- Dans **Règle de connexion aux applications**, spécifiez un nom et les différentes options MFA. Pour tester la fonctionnalité, vous pouvez laisser toutes les options par défaut. Par contre, sous **Actions**, vous devez sélectionner **Demander le facteur**, puis spécifier la fréquence à laquelle les utilisateurs doivent se connecter. Cliquez sur **Enregistrer**.

## Créer et affecter un utilisateur Okta

1. Dans Okta, créez un utilisateur avec le même nom d'utilisateur que vous avez créé dans Tableau (`user@example.com`) : **Répertoire > Personnes > Ajouter une personne**.
2. Une fois l'utilisateur créé, attribuez la nouvelle application Okta à cette personne : cliquez sur le nom d'utilisateur, puis attribuez l'application dans **Attribuer une application**.

## Installer Mellon pour la pré-authentification

Cet exemple utilise `mod_auth_mellon`, un module open source populaire. Certaines distributions Linux contiennent des versions obsolètes de `mod_auth_mellon` provenant d'un référentiel plus ancien. Ces versions obsolètes peuvent contenir des vulnérabilités de sécurité

inconnues ou des problèmes fonctionnels. Si vous choisissez d'utiliser `mod_auth_mellon`, vérifiez que vous utilisez une version actuelle.

Le module `mod_auth_mellon` est un logiciel tiers. Nous avons tout mis en œuvre pour vérifier et documenter les procédures permettant d'activer le scénario. Il peut toutefois arriver que le logiciel tiers change ou que votre scénario diffère de l'architecture de référence décrite ici. Veuillez vous référer à la documentation du fournisseur tiers pour les détails de configuration et l'assistance.

1. Sur l'instance EC2 active qui exécute la passerelle indépendante, installez une version actuelle du module d'authentification Mellon.
2. Créez le répertoire `/etc/mellon`:

```
sudo mkdir /etc/mellon
```

## Configurer Mellon comme module de pré-authentification

Exécutez cette procédure sur la première instance de passerelle indépendante.

Vous devez avoir une copie du fichier `pre-auth_idp_metadata.xml` que vous avez créé à partir de la configuration Okta.

1. Changez de répertoire :

```
cd /etc/mellon
```

2. Créez les métadonnées du fournisseur de services. Exécutez le script `mellon_create_metadata.sh`. Vous devez inclure l'ID d'entité et l'URL de retour de votre entreprise dans la commande.

L'URL de retour est appelée URL d'*authentification unique* dans Okta. Le dernier élément du chemin dans l'URL de retour est appelé `MellonEndpointPath` dans le fichier de configuration `mellon.conf` présenté plus loin dans cette procédure. Dans cet exemple, nous spécifions `sso` comme chemin de point de terminaison.

Par exemple :

```
sudo /usr/libexec/mod_auth_mellon/mellon_create_metadata.sh
https://tableau.example.com "https://tableau.example.com/sso"
```

Le script renvoie le certificat du fournisseur de services, la clé et les fichiers de métadonnées.

3. Renommez les fichiers du fournisseur de services dans le répertoire `mellon` pour une meilleure lisibilité. Nous désignerons ces fichiers par les noms suivants dans la documentation :

```
sudo mv *.key mellon.key
sudo mv *.cert mellon.cert
sudo mv *.xml sp_metadata.xml
```

4. Copiez le fichier `pre-auth_idp_metadata.xml` sur le même répertoire.
5. Modifiez la propriété et les autorisations sur tous les fichiers dans le répertoire `/etc/mellon`:

```
sudo chown tableau-tsig mellon.key
sudo chown tableau-tsig mellon.cert
sudo chown tableau-tsig sp_metadata.xml
sudo chown tableau-tsig pre-auth_idp_metadata.xml
sudo chmod +r * mellon.key
sudo chmod +r * mellon.cert
sudo chmod +r * sp_metadata.xml
sudo chmod +r * pre-auth_idp_metadata.xml
```

6. Créez le répertoire `/etc/mellon/conf.d`:

```
sudo mkdir /etc/mellon/conf.d
```

7. Créez le fichier `global.conf` dans le répertoire `/etc/mellon/conf.d`.

Copiez le contenu du fichier comme indiqué ci-dessous, mais mettez à jour `MellonCookieDomain` avec votre nom de domaine racine. Par exemple, si le nom de domaine de Tableau est `tableau.example.com`, entrez `example.com` pour le domaine racine.

```
<Location "/">
AuthType Mellon
MellonEnable auth
Require valid-user
MellonCookieDomain <root domain>
MellonSPPrivateKeyFile /etc/mellon/mellon.key
MellonSPCertFile /etc/mellon/mellon.cert
MellonSPMetadataFile /etc/mellon/sp_metadata.xml
MellonIdPMetadataFile /etc/mellon/pre-auth_idp_metadata.xml
MellonEndpointPath /sso
</Location>

<Location "/tsighk">
MellonEnable Off
</Location>
```

8. Créez le fichier `mellonmod.conf` dans le répertoire `/etc/mellon/conf.d`.

Ce fichier contient une seule directive qui spécifie l'emplacement du fichier `mod_auth_mellon.so`. L'emplacement indiqué dans l'exemple ici est l'emplacement par défaut du fichier. Vérifiez que le fichier se trouve à cet emplacement ou modifiez le chemin dans cette directive pour qu'il corresponde à l'emplacement réel de `mod_auth_mellon.so`:

```
LoadModule auth_mellon_module /usr/lib64/httpd/modules/mod_
auth_mellon.so
```

## Créer une application Tableau Server dans Okta

1. Dans le tableau de bord Okta : **Applications > Applications > Parcourir le catalogue d'applications**



2. Dans **Parcourir le catalogue d'intégration d'applications**, recherchez **Tableau**, sélectionnez la section **Tableau Server**, puis cliquez sur **Ajouter**.
3. Sur **Ajouter Tableau Server > Paramètres généraux**, saisissez une étiquette, puis cliquez sur **Suivant**.
4. Dans Options de connexion, sélectionnez **SAML 2.0** et faites défiler jusqu'à Paramètres de connexion avancés :
  - **ID d'entité SAML** : saisissez l'URL publique, par exemple `https://-tableau.example.com`.
  - **Format du nom d'utilisateur de l'application** : E-mail
5. Cliquez sur le lien **Métadonnées du fournisseur d'identité** pour lancer un navigateur. Copiez le lien du navigateur. C'est le lien que vous utiliserez lorsque vous configurerez Tableau dans la procédure qui suit.
6. Cliquez sur **Terminé**.
7. Attribuez la nouvelle application Tableau Server Okta à votre utilisateur (`user@example.com`) : cliquez sur le nom d'utilisateur, puis attribuez l'application dans **Attribuer une application**.

## Définir la configuration du module d'authentification sur Tableau Server

Exécutez les commandes suivantes sur le Nœud 1 de Tableau Server. Ces commandes spécifient les emplacements des fichiers pour les fichiers de configuration Mellon sur l'ordinateur distant de la passerelle indépendante. Vérifiez à nouveau que les chemins d'accès aux fichiers spécifiés dans ces commandes correspondent aux chemins d'accès et à l'emplacement des fichiers sur l'ordinateur distant de la passerelle indépendante.

```
tsm configuration set -k gateway.tsig.authn_module_block -v "/etc/-mellon/conf.d/mellonmod.conf" --force-keys
tsm configuration set -k gateway.tsig.authn_global_block -v "/etc/-mellon/conf.d/global.conf" --force-keys
```

Pour réduire les temps d'arrêt, n'appliquez pas les modifications tant que vous n'avez pas activé SAML comme décrit dans la section suivante.

## Activer SAML sur Tableau Server pour IdP

Exécutez cette procédure sur le Nœud 1 de Tableau Server.

1. Téléchargez les métadonnées de l'application Tableau depuis Okta. Utilisez le lien que vous avez enregistré de la procédure précédente :

```
wget https://dev-66144217.ok-ta.com/app/exklegxgt1fhjkSeS5d7/sso/saml/metadata -O idp_metadata.xml
```

2. Copiez un certificat TLS et le fichier de clé associé sur Tableau Server. Le fichier de clé doit être une clé RSA. Pour plus d'informations sur la certificat et les exigences de l'IdP, consultez *Exigences en matière d'authentification SAML* ([Linux](#)).

Pour simplifier la gestion et le déploiement des certificats, et comme meilleure pratique de sécurité, nous vous recommandons d'utiliser des certificats générés par une autorité de certification (CA) tierce majeure de confiance. Sinon, vous pouvez générer des certificats auto-signés ou utiliser les certificats d'un PKI pour TLS.

Si vous n'avez pas de certificat TLS, vous pouvez générer un certificat auto-signé en appliquant la procédure intégrée ci-dessous.

### Générer un certificat auto-signé

Exécutez cette procédure sur le Nœud 1 de Tableau Server.

- a. Générez la clé de l'autorité de certification racine (CA) de signature :

```
openssl genrsa -out rootCAKey-saml.pem 2048
```

- b. Créez le certificat CA racine :

## Guide de déploiement de Tableau Server en entreprise

```
openssl req -x509 -sha256 -new -nodes -key rootCAKey-saml.-
pem -days 3650 -out rootCACert-saml.pem
```

Vous serez invité à saisir des valeurs pour les champs du certificat. Par exemple :

```
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:Washington
Locality Name (eg, city) [Default City]:Seattle
Organization Name (eg, company) [Default Company Ltd]:Ta-
bleau
Organizational Unit Name (eg, section) []:Operations
Common Name (eg, your name or your server's hostname) []:ta-
bleau.example.com
Email Address []:example@tableau.com
```

- c. Créez le certificat et la clé associée (`server-saml.csr` et `server-saml.key` dans l'exemple ci-dessous). Le nom du sujet du certificat doit correspondre au nom de l'hôte public de l'hôte Tableau. Le nom du sujet est défini à l'aide de l'option `-subj` avec le format `"/CN=<host-name>"`, par exemple :

```
openssl req -new -nodes -text -out server-saml.csr -keyout
server-saml.key -subj "/CN=tableau.example.com"
```

- d. Signez le nouveau certificat avec le certificat CA que vous avez créé ci-dessus. La commande suivante génère également le certificat au format `crt` :

```
openssl x509 -req -in server-saml.csr -days 3650 -CA
rootCACert-saml.pem -CAkey rootCAKey-saml.pem -CAcrea-
teserial -out server-saml.crt
```

- e. Convertissez le fichier de clé en RSA. Tableau requiert un fichier de clé RSA pour SAML. Pour convertir la clé, exécutez la commande suivante :

```
openssl rsa -in server-saml.key -out server-saml-rsa.key
```

3. Configurez SAML. Exécutez la commande suivante, en spécifiant votre ID d'entité et votre URL de retour, ainsi que les chemins d'accès au fichier de métadonnées, au fichier de certificat et au fichier de clé :

```
tsm authentication saml configure --idp-entity-id "https://-
tableau.example.com" --idp-return-url "https://-
tableau.example.com" --idp-metadata idp_metadata.xml --cert-
file "server-saml.crt" --key-file "server-saml-rsa.key"
```

```
tsm authentication saml enable
```

4. Si votre entreprise exécute Tableau Desktop 2021.4 ou une version ultérieure, vous devez exécuter la commande suivante pour activer l'authentification via les serveurs proxy inverses.

Les versions de Tableau Desktop 2021.2.1 - 2021.3 fonctionneront sans que cette commande soit exécutée, à condition que votre module de pré-authentification (par exemple Mellon) soit configuré pour autoriser la conservation des cookies de domaine de niveau supérieur.

```
tsm configuration set -k features.ExternalBrowserOAuth -v false
```

5. Appliquez les changements de configuration :

```
tsm pending-changes apply
```

## Redémarrer le service tsig-httpd

Au fur et à mesure que votre déploiement Tableau Server applique les modifications, reconnectez-vous à l'ordinateur Tableau Server Independent Gateway et exécutez les commandes suivantes pour redémarrer le service tsig-httpd :

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

## Valider la fonctionnalité SAML

Pour valider la fonctionnalité SAML de bout en bout, connectez-vous à Tableau Server avec l'URL publique (par exemple, <https://tableau.example.com>) en utilisant le compte administrateur Tableau que vous avez créé au début de cette procédure.

Si TSM ne démarre pas (« erreur de passerelle ») ou si vous obtenez des erreurs de navigateur lorsque vous tentez de vous connecter, consultez [Résolution des problèmes Tableau Server Independent Gateway](#).

## Configurer le module d'authentification sur la seconde instance de la passerelle indépendante

Après avoir correctement configuré la première instance de passerelle indépendante, déployez la deuxième instance. L'exemple ici est le processus final d'installation du scénario AWS/Mellon/Okta décrit dans cette rubrique. La procédure suppose que vous avez déjà installé la passerelle indépendante sur la deuxième instance, comme décrit précédemment dans cette rubrique ([Installer la passerelle indépendante](#)).

Le processus de déploiement de la deuxième passerelle indépendante nécessite les étapes suivantes :

1. Sur la deuxième instance de la passerelle indépendante, installez le module d'authentification Mellon.

Ne configurez pas le module d'authentification Mellon comme décrit précédemment dans cette rubrique. Au lieu de cela, vous devez cloner la configuration comme décrit dans les étapes suivantes.

2. Sur la (première) instance configurée de passerelle indépendante :

Prenez une copie tar de la configuration Mellon existante. La sauvegarde tar conservera toute la hiérarchie des répertoires et les autorisations. Exécutez les commandes suivantes :

```
cd /etc
sudo tar -cvf mellon.tar mellon
```

Copiez `mellon.tar` sur la deuxième instance de passerelle indépendante.

3. Sur la deuxième instance de passerelle indépendante :

Extrayez (« décompresser ») le fichier tar sur la deuxième instance dans le répertoire `/etc`. Exécutez les commandes suivantes :

```
cd /etc
sudo tar -xvf mellon.tar
```

4. Sur le nœud 1 du déploiement de Tableau Server : mettez à jour le fichier de connexion (`tsig.json`) avec les informations de connexion de la deuxième passerelle indépendante. Vous devrez récupérer la clé d'authentification comme décrit précédemment dans cette rubrique ([Installer la passerelle indépendante](#)).

Voici un exemple de fichier de connexion (`tsig.json`) :

```
{
 "independentGateways": [
 {
 "id": "ip-10-0-1-169.ec2.internal",
 "host": "ip-10-0-1-169.ec2.internal",
 "port": "21319",
 "protocol" : "http",
 "authsecret": "13660-27118-29070-25482-9518-22453"
 },
 {
 "id": "ip-10-0-2-230.ec2.internal",
```

## Guide de déploiement de Tableau Server en entreprise

```
"host": "ip-10-0-2-230.ec2.internal",
"port": "21319",
"protocol" : "http",
"authsecret": "9055-27834-16487-27455-30409-7292"
}]
}
```

5. Sur le Nœud 1 du déploiement de Tableau Server : exécutez les commandes suivantes pour mettre à jour la configuration :

```
tsm stop

tsm topology external-services gateway update -c tsig.json

tsm start
```

6. Sur les deux instances de la passerelle indépendante : pendant le démarrage de Tableau Server, redémarrez le processus `tsig-httpd`:

```
sudo su - tableau-tsig

systemctl --user restart tsig-httpd

exit
```

7. Dans AWS **EC2>Groupes cibles** : mettez à jour le groupe cible de manière à inclure l'instance EC2 exécutant la deuxième instance de passerelle indépendante.

Sélectionnez le groupe cible que vous venez de créer, puis cliquez sur l'onglet Cibles :

- Cliquez sur **Modifier**.
- Sélectionnez l'instance EC2 du deuxième ordinateur de la passerelle indépendante, puis cliquez sur **Ajouter à enregistré**. Cliquez sur **Enregistrer**.

# Partie 6 - Configuration après l'installation

## Configurer SSL/TLS depuis l'équilibreur de charge vers Tableau Server

Certaines entreprises exigent un canal de cryptage de bout en bout du client au service principal. L'architecture de référence par défaut telle que décrite jusqu'à présent spécifie SSL du client à l'équilibreur de charge s'exécutant dans le niveau Web de votre entreprise.

Cette section décrit comment configurer SSL/TLS pour Tableau Server et la passerelle indépendante dans l'exemple d'architecture de référence AWS. Pour un exemple de configuration décrivant comment configurer SSL/TLS sur Apache dans l'architecture de référence AWS, consultez Exemple : Configurer SSL/TLS dans l'architecture de référence AWS.

À l'heure actuelle, TLS n'est pas pris en charge sur les processus principaux de Tableau Server qui s'exécutent dans la plage 8000-9000. Pour activer TLS, vous devez configurer la passerelle indépendante avec une connexion par relais à Tableau Server.

Cette procédure décrit comment activer et configurer TLS sur la passerelle indépendante vers Tableau Server et Tableau Server vers la passerelle indépendante. La procédure chiffre le trafic de relais sur HTTPS/443 et le trafic de maintenance sur HTTPS/21319.

Les procédures Linux décrites tout au long de cet exemple montrent des commandes pour les distributions de type RHEL. Plus précisément, les commandes présentées ici ont été développées avec la distribution Amazon Linux 2. Si vous exécutez la distribution Ubuntu, modifiez les commandes en conséquence.

Les conseils ici sont proposés à titre prescriptif pour l'exemple d'architecture de référence AWS spécifique tel que présenté dans ce guide. Par conséquent, les configurations



facultatives ne sont pas incluses. Pour obtenir une documentation de référence complète, consultez *Configurer TLS sur la passerelle indépendante* ([Linux](#)).

## Avant de configurer TLS

Effectuez les configurations TLS en dehors des heures ouvrables. La configuration nécessite au moins un redémarrage de Tableau Server. Si vous exécutez un déploiement complet d'architecture de référence à quatre nœuds, le redémarrage peut prendre un certain temps.

- Vérifiez que les clients peuvent se connecter à Tableau Server via HTTP. La configuration de TLS avec une passerelle indépendante est un processus en plusieurs étapes et peut nécessiter un dépannage. Par conséquent, nous vous recommandons de commencer par un déploiement Tableau Server entièrement opérationnel avant de configurer TLS.
- Collectez les certificats TLS/SSL, les clés et les ressources associés. Vous aurez besoin de certificats SSL pour les passerelles indépendantes et pour Tableau Server. Pour simplifier la gestion et le déploiement des certificats, et comme meilleure pratique de sécurité, nous vous recommandons d'utiliser des certificats générés par une autorité de certification (CA) tierce majeure de confiance. Vous pouvez sinon générer des certificats auto-signés ou utiliser des certificats d'une PKI pour TLS.

L'exemple de configuration de cette rubrique utilise les noms de ressources suivants à titre d'illustration :

- `tsig-ssl.crt` : le certificat TLS/SSL pour la passerelle indépendante.
- `tsig-ssl.key` : la clé privée pour `tsig-ssl.crt` sur la passerelle indépendante.
- `ts-ssl.crt` : le certificat TLS/SSL pour Tableau Server.
- `ts-ssl.key` : la clé privée pour `tsig-ssl.crt` sur Tableau Server.
- `tableau-server-CA.pem` spécifie le certificat racine pour l'autorité de certification qui génère le certificat présenté par les ordinateurs Tableau Server. Ce certificat n'est généralement pas requis si vous utilisez des certificats provenant de tiers de confiance majeurs.

- `rootTSIG-CACert.pem` : le certificat racine de l'autorité de certification qui génère les certificats pour les ordinateurs de la passerelle indépendante. Ce certificat n'est généralement pas requis si vous utilisez des certificats provenant de tiers de confiance majeurs.
- Il existe d'autres certificats et ressources de fichiers de clé requis pour SAML qui sont détaillés dans la Partie 5 de ce guide.
- Si votre implémentation nécessite l'utilisation d'un fichier de chaîne de certificats, consultez l'article de la Base de connaissances [Configurer TLS sur une passerelle indépendante en cas d'utilisation d'un certificat avec chaîne de certificats](#).
- Vérifiez que vous avez accès à l'IdP. Si vous utilisez un IdP pour l'authentification, vous devrez probablement apporter des modifications aux URL de destination et de destination au niveau de l'IdP après avoir configuré SSL/TLS.

## Configurer les ordinateurs de la passerelle indépendante pour TLS

La configuration de TLS peut être un processus sujet aux erreurs. Étant donné que le débogage sur deux instances de passerelle indépendante peut prendre du temps, nous vous recommandons d'activer et de configurer TLS sur le déploiement EDG avec une seule passerelle indépendante. Une fois que vous avez validé le fonctionnement de TLS sur l'ensemble du déploiement, configurez le deuxième ordinateur de passerelle indépendante.

### Étape 1 : Distribuer les certificats et les clés à l'ordinateur de la passerelle indépendante

Vous pouvez distribuer les ressources à n'importe quel répertoire arbitraire tant que l'utilisateur `tsig-httpd` dispose d'un accès en lecture aux fichiers. Les chemins d'accès à ces fichiers sont référencés dans d'autres procédures. Nous utiliserons les exemples de chemins sous `/etc/ssl`, comme illustré ci-dessous, tout au long de la rubrique.

## Guide de déploiement de Tableau Server en entreprise

1. Créer un répertoire pour la clé privée :

```
sudo mkdir -p /etc/ssl/private
```

2. Copiez les fichiers de certificat et de clé dans les chemins d'accès `/etc/ssl`. Par exemple :

```
sudo cp tsig-ssl.crt /etc/ssl/certs/
```

```
sudo cp tsig-ssl.key /etc/ssl/private/
```

3. (Facultatif) Si vous utilisez un certificat auto-signé ou PKI pour SSL/TLS sur Tableau Server, vous devez également copier le fichier de certificat racine CA sur l'ordinateur de la passerelle indépendante. Par exemple :

```
sudo cp tableau-server-CA.pem /etc/ssl/certs/
```

## Étape 2 : Mettre à jour les variables d'environnement pour TLS

Vous devez mettre à jour les variables d'environnement de port et de protocole pour la configuration de la passerelle indépendante.

Modifiez ces valeurs en mettant à jour le fichier `/etc/opt/tableau/tableau_tsig/environment.bash`, comme suit :

```
TSIG_HK_PROTOCOL="https"
```

```
TSIG_PORT="443"
```

```
TSIG_PROTOCOL="https"
```

## Étape 3 : Mettre à jour le fichier de configuration stub pour le protocole HK

Modifiez manuellement le fichier de configuration stub (`/var/opt/tableau/tableau_tsig/config/httpd.conf.stub`) pour définir les directives Apache httpd liées à TLS pour le protocole de maintenance (HK).

Le fichier de configuration de stub inclut un bloc de directives liées à TLS qui sont commentées avec un marqueur `#TLS#`. Supprimez les marqueurs des directives comme indiqué

dans l'exemple ci-dessous. Notez que l'exemple montre l'utilisation du certificat racine CA pour le certificat SSL utilisé sur Tableau Server avec l'option `SSLCACertificateFile`.

```
#TLS# SSLPassPhraseDialog exec:/path/to/file
<VirtualHost *:${TSIG_HK_PORT}>
SSLEngine on
#TLS# SSLHonorCipherOrder on
#TLS# SSLCompression off
SSLCertificateFile /etc/ssl/certs/tsig-ssl.crt
SSLCertificateKeyFile /etc/ssl/private/tsig-ssl.key
SSLCACertificateFile /etc/ssl/certs/tableau-server-CA.pem
#TLS# SSLCARevocationFile /path/to/file
</VirtualHost>
```

Ces modifications seront perdues si vous réinstallez la passerelle indépendante. Nous vous recommandons d'effectuer une copie de sauvegarde.

## Étape 4 : Copier le fichier stub et redémarrer le service

1. Copiez le fichier que vous avez mis à jour à la dernière étape pour mettre à jour `httpd.conf` avec les modifications :

```
cp /var/opt/tableau/tableau_tsig/config/httpd.conf.stub
/var/opt/tableau/tableau_tsig/config/httpd.conf
```

2. Redémarrez le service de passerelle indépendante :

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

Après le redémarrage, la passerelle indépendante sera opérationnelle une fois que vous aurez exécuté la prochaine série d'étapes sur Tableau Server. Une fois que vous avez terminé les étapes sur Tableau Server, la passerelle indépendante récupère les modifications et se met en ligne.

## Configurer le Nœud 1 de Tableau Server pour TLS

Exécutez ces étapes sur le Nœud 1 du déploiement Tableau Server.

### Étape 1 : Copier les certificats et les clés et arrêter TSM

1. Vérifiez que les certificats et les clés « SSL externe » de Tableau Server sont copiés sur le Nœud 1.
2. Pour réduire les temps d'arrêt, nous vous recommandons d'arrêter TSM, d'exécuter les étapes suivantes, puis de démarrer TSM une fois les modifications appliquées :

```
tsm stop
```

### Étape 2 : Définir les ressources de certificat et activer la configuration de la passerelle indépendante

1. Spécifiez l'emplacement des fichiers de certificat et de clé pour la passerelle indépendante. Ces chemins d'accès font référence à l'emplacement sur les ordinateurs de la passerelle indépendante. Remarque : cet exemple suppose que la même paire de certificats et de clés est utilisée pour protéger le trafic HTTPS et de maintenance :

```
tsm configuration set -k gateway.tsig.ssl.cert.file_name -v
/etc/ssl/certs/tsig-ssl.crt --force-keys
tsm configuration set -k gateway.tsig.ssl.key.file_name -v
/etc/ssl/private/tsig-ssl.key --force-keys
```

2. Activez TLS pour les protocoles HTTPS et HK pour la passerelle indépendante :

```
tsm configuration set -k gateway.tsig.ssl.enabled -v true --
force-keys
tsm configuration set -k gateway.tsig.hk.ssl.enabled -v true --
force-keys
```

3. (Facultatif) Si vous utilisez un certificat auto-signé ou PKI pour SSL/TLS sur la passerelle indépendante, vous devez télécharger le fichier de certificat racine CA. Le fichier

de certificat racine de l'autorité de certification est le certificat racine qui a été utilisé pour générer les certificats pour les ordinateurs de la passerelle indépendante. Par exemple,

```
tsm security custom-cert add -c rootTSIG-CACert.pem
```

4. (Facultatif) Si vous utilisez un certificat auto-signé ou PKI pour SSL/TLS sur Tableau Server, vous devez également copier le fichier de certificat racine CA sur le répertoire `/etc/ssl/certs` de la passerelle indépendante. Le fichier de certificat racine CA est le certificat racine qui a été utilisé pour générer les certificats pour les ordinateurs Tableau Server. Après avoir copié le certificat sur la passerelle indépendante, vous devez spécifier l'emplacement du certificat sur le Nœud 1 avec la commande tsm suivante. Par exemple,

```
tsm configuration set -k gateway.tsig.ssl.proxy.gateway_relay_
cluster.cacertificatefile -v /etc/ssl/certs/tableau-server-CA.-
pem --force-keys
```

5. (Facultatif : à des fins de test uniquement) Si vous utilisez le partage de certificats auto-signés ou PKI entre ordinateurs et que, par conséquent, les noms de sujet sur les certificats ne correspondent pas aux noms d'ordinateur, vous devez désactiver la vérification des certificats.

```
tsm configuration set -k gateway.tsig.ssl.proxy.verify -v optio-
nal_no_ca --force-keys
```

## Étape 3 : Activer « SSL externe » pour Tableau Server et appliquer les modifications

1. Activez et configurez « SSL externe » sur Tableau Server :

```
tsm security external-ssl enable --cert-file ts-ssl.crt --key-
file ts-ssl.key
```

2. Appliquez les modifications.

```
tsm pending-changes apply
```

## Étape 4 : Mettre à jour le fichier JSON de configuration de la passerelle et démarrer tsm

1. Mettez à jour le fichier de configuration de la passerelle indépendante (par exemple, `tsig.json`) côté Tableau Server pour spécifier le protocole `https` pour les objets de passerelle indépendante :

```
"protocol" : "https",
```

2. Supprimez (ou commentez) les informations de connexion pour la deuxième instance de passerelle indépendante. Assurez-vous de vérifier le fichier JSON dans un éditeur externe avant de l'enregistrer.

Après avoir configuré et validé TLS pour l'instance unique de passerelle indépendante, vous allez mettre à jour ce fichier JSON avec les informations de connexion pour la deuxième instance de passerelle indépendante.

3. Exécutez la commande suivante pour mettre à jour la configuration de la passerelle indépendante :

```
tsm topology external-services gateway update -c tsig.json
```

4. Démarrez TSM.

```
tsm start
```

5. Pendant le démarrage de TSM, connectez-vous à l'instance de passerelle indépendante et redémarrez le service `tsig-httpd`:

```
sudo su - tableau-tsig
```

```
systemctl --user restart tsig-httpd
```

```
exit
```

## Mettre à jour les URL du module d'authentification de l'IdP sur HTTPS

Si vous avez configuré un fournisseur d'identité externe pour Tableau, vous devrez probablement mettre à jour les URL de retour dans le tableau de bord administratif de l'IdP.

Par exemple, si vous utilisez une application de pré-authentification Okta, vous devrez mettre à jour l'application de l'IdP Mellon de manière à utiliser le protocole HTTPS pour l'URL du destinataire et l'URL de destination.

## Configurer l'équilibreur de charge AWS pour HTTPS

Si vous effectuez un déploiement avec l'équilibreur de charge AWS comme documenté dans ce guide, vous allez reconfigurer l'équilibreur de charge AWS de manière à envoyer le trafic HTTPS vers les ordinateurs exécutant la passerelle indépendante :

1. Supprimer le groupe cible HTTP existant :

Dans **Groupes cibles**, sélectionnez le groupe cible HTTP qui a été configuré pour l'équilibreur de charge, cliquez sur **Actions**, puis sur **Supprimer**.

2. Créez un groupe HTTPS cible :

### Groupes cibles > Créer un groupe cible

- Sélectionnez « Instances »
- Entrez un nom de groupe cible, par exemple `TG-internal-HTTPS`
- Sélectionnez votre VPC
- Protocole : HTTPS 443
- Sous **Contrôles d'intégrité** > **Paramètres avancés des contrôles d'intégrité** > **Codes de réussite**, ajoutez la liste des codes pour lire : 200, 303.
- Cliquez sur **Créer**.

3. Sélectionnez le groupe cible que vous venez de créer, puis cliquez sur l'onglet **Cibles**.



- Cliquez sur **Modifier**.
  - Sélectionnez les instances EC2 qui exécutent la passerelle indépendante Tableau Server, puis cliquez sur **Ajouter aux instances enregistrées**.
  - Cliquez sur **Enregistrer**.
4. Une fois le groupe cible créé, vous devez activer la permanence :
- Ouvrez la page Groupe cible AWS (**EC2 > Équilibreurs de charge > Groupes cibles**), sélectionnez l'instance d'équilibreur de charge cible que vous venez de configurer. Dans le menu **Actions**, sélectionnez **Modifier les attributs**.
  - Sur la page **Modifier les attributs**, sélectionnez **Persistance**, spécifiez une durée 1 day, puis **Enregistrer les modifications**.
5. Sur l'équilibreur de charge, mettez à jour les règles d'écoute. Sélectionnez l'équilibreur de charge que vous avez configuré pour ce déploiement, puis cliquez sur l'onglet **Écouteurs**.
- Pour **HTTP:80**, cliquez sur **Afficher/modifier les règles**. Dans la page **Règles** résultante, cliquez sur l'icône de modification (une fois en haut de la page, puis à nouveau près de la règle) pour modifier la règle. Supprimez la règle THEN existante et remplacez-la en cliquant sur **Ajouter une action > Rediriger vers....** Dans la configuration THEN résultante, spécifiez les ports **HTTPS** et **443** et conservez les paramètres par défaut des autres options. Enregistrez le paramètre puis cliquez sur **Mettre à jour**.
  - Pour **HTTPS:443**, cliquez sur **Afficher/modifier les règles**. Dans la page **Règles** résultante, cliquez sur l'icône de modification (une fois en haut de la page, puis à nouveau près de la règle) pour modifier la règle. Supprimez la règle THEN existante et remplacez-la en cliquant sur **Ajouter une action > Transférer vers....** Désignez sur le groupe HTTPS que vous venez de créer comme groupe cible. Sous **Persistance au niveau du groupe**, activez la persistance et définissez la durée sur 1 jour. Enregistrez le paramètre puis cliquez sur **Mettre à jour**.
6. Sur l'équilibreur de charge, mettez à jour le délai d'inactivité sur 400 secondes. Sélectionnez l'équilibreur de charge que vous avez configuré pour ce déploiement, puis cliquez sur l'onglet **Actions > Modifier les attributs**. Définissez le **Délai d'inactivité** sur 400 secondes, puis cliquez sur **Enregistrer**.

## Valider TLS

Pour valider la fonctionnalité TLS, connectez-vous à Tableau Server avec l'URL publique (par exemple, <https://tableau.example.com>) en utilisant le compte administrateur Tableau que vous avez créé au début de cette procédure.

Si TSM ne démarre pas ou si vous obtenez d'autres erreurs, consultez [Résolution des problèmes Tableau Server Independent Gateway](#).

## Configurer la deuxième instance de passerelle indépendante pour SSL

Une fois que vous avez correctement configuré la première instance de passerelle indépendante, déployez la deuxième instance.

Le processus de déploiement de la deuxième passerelle indépendante nécessite les étapes suivantes :

1. Sur la (première) instance configurée de la passerelle indépendante : copiez les fichiers suivants aux emplacements correspondants sur la deuxième instance de la passerelle indépendante :
  - `/etc/ssl/certs/tsig-ssl.crt`
  - `/etc/ssl/private/tsig-ssl.key` (Vous devrez créer le répertoire `private` sur la seconde instance).
  - `/var/opt/tableau/tableau_tsig/config/httpd.conf.stub`
  - `/etc/opt/tableau/tableau_tsig/environment.bash`
2. Sur le Nœud 1 du déploiement de Tableau Server : mettez à jour le fichier de connexion (`tsig.json`) avec les informations de connexion de la deuxième passerelle indépendante.

Voici un exemple de fichier de connexion (`tsig.json`) :

```
{
 "independentGateways": [
 {
 "id": "ip-10-0-1-169.ec2.internal",
 "host": "ip-10-0-1-169.ec2.internal",
 "port": "21319",
 "protocol" : "https",
 "authsecret": "13660-27118-29070-25482-9518-22453"
 },
 {
 "id": "ip-10-0-2-230.ec2.internal",
 "host": "ip-10-0-2-230.ec2.internal",
 "port": "21319",
 "protocol" : "https",
 "authsecret": "9055-27834-16487-27455-30409-7292"
 }
]
}
```

3. Sur le Nœud 1 du déploiement de Tableau Server : exécutez les commandes suivantes pour mettre à jour la configuration :

```
tsm stop

tsm topology external-services gateway update -c tsig.json

tsm start
```

4. Sur les deux instances de passerelle indépendante : pendant le démarrage de Tableau Server, redémarrez le processus `tsig-httpd` sur les deux instances de passerelle indépendante :

```
sudo su - tableau-tsig

systemctl --user restart tsig-httpd

exit
```

5. Dans AWS **EC2>Groupes cibles** : mettez à jour le groupe cible de manière à inclure l'instance EC2 exécutant la deuxième instance de passerelle indépendante.

Sélectionnez le groupe cible que vous venez de créer, puis cliquez sur l'onglet Cibles :

- Cliquez sur **Modifier**.
- Sélectionnez l'instance EC2 du deuxième ordinateur de la passerelle indépendante, puis cliquez sur **Ajouter à enregistré**. Cliquez sur **Enregistrer**.

## Configurer SSL pour Postgres

Vous pouvez éventuellement configurer SSL (TLS) pour la connexion Postgres au référentiel externe sur Tableau Server.

Pour simplifier la gestion et le déploiement des certificats, et comme meilleure pratique de sécurité, nous vous recommandons d'utiliser des certificats générés par une autorité de certification (CA) majeure tierce de confiance. Vous pouvez sinon générer des certificats auto-signés ou utiliser des certificats d'une PKI pour TLS.

Cette procédure décrit comment utiliser OpenSSL pour générer un certificat auto-signé sur l'hôte Postgres dans une distribution Linux de type RHEL pour l'exemple d'architecture AWS de référence.

Après avoir généré et signé le certificat SSL, vous devez copier le certificat CA sur l'hôte Tableau.

### Sur l'hôte exécutant Postgres :

1. Générez la clé de l'autorité de certification racine (CA) de signature :

```
openssl genrsa -out pgsql-rootCAKey.pem 2048
```

2. Créez le certificat CA racine :

```
openssl req -x509 -sha256 -new -nodes -key pgsql-rootCAKey.pem
-days 3650 -out pgsql-rootCACert.pem
```

## Guide de déploiement de Tableau Server en entreprise

Vous serez invité à saisir des valeurs pour les champs du certificat. Par exemple :

```
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:Washington
Locality Name (eg, city) [Default City]:Seattle
Organization Name (eg, company) [Default Company Ltd]:Tableau
Organizational Unit Name (eg, section) []:Operations
Common Name (eg, Postgres server's hostname) []:ip-10-0-1-189.us-west-1.compute.internal
Email Address []:example@tableau.com
```

3. Créez le certificat et la clé associée (`server.csr` et `server.key` dans l'exemple ci-dessous) pour l'ordinateur Postgres. Le nom du sujet du certificat doit correspondre au nom DNS privé EC2 de l'hôte Postgres. Le nom du sujet est défini à l'aide de l'option `-subj` avec le format `"/CN=<private DNS name>"`, par exemple :

```
openssl req -new -nodes -text -out server.csr -keyout server.key -subj "/CN=ip-10-0-1-189.us-west-1.compute.internal"
```

4. Signez le nouveau certificat avec le certificat CA que vous avez créé à l'étape 2. La commande suivante génère également le certificat au format `crt` :

```
openssl x509 -req -in server.csr -days 3650 -CA pgsql-rootCACert.pem -CAkey pgsql-rootCAKey.pem -CAcreateserial -out server.crt
```

5. Copiez les fichiers `crt` et `key` dans le chemin d'accès Postgres `/var/lib/pgsql/13/data/` :

```
sudo cp server.crt /var/lib/pgsql/13/data/
sudo cp server.key /var/lib/pgsql/13/data/
```

6. Basculez vers l'utilisateur `root` :

```
sudo su
```

7. Définissez les autorisations sur les fichiers `cer` et `key`. Exécutez les commandes suivantes :

```
cd /var/lib/pgsql/13/data
chown postgres.postgres server.crt
chown postgres.postgres server.key
chmod 0600 server.crt
chmod 0600 server.key
```

8. Mettez à jour le fichier de configuration `pg_hba`, `/var/lib/pgsql/13/data/pg_hba.conf`, pour spécifier la confiance `md5` :

Modifiez les instructions de connexion existantes de

```
host all all 10.0.30.0/24 password et
```

```
host all all 10.0.31.0/24 password
```

sur

```
host all all 10.0.30.0/24 md5 et
```

```
host all all 10.0.31.0/24 md5.
```

9. Mettez à jour le fichier `postgresql`, `/var/lib/pgsql/13/data/postgresql.conf`, en ajoutant cette ligne :

```
ssl = on
```

10. Quittez le mode utilisateur `root` :

```
exit
```

11. Redémarrez Postgres :

```
sudo systemctl restart postgresql-13
```

## Facultatif : activer la validation de la confiance des certificats sur Tableau Server pour SSL Postgres

Si vous avez suivi la procédure d'installation dans Partie 4 - Installer et configurer Tableau Server, Tableau Server est configuré avec SSL facultatif pour la connexion Postgres. Cela signifie que la configuration de SSL sur Postgres (comme décrit ci-dessus) entraînera une connexion cryptée.

Si vous souhaitez exiger une validation de certificat de confiance pour la connexion, vous devez exécuter la commande suivante sur Tableau Server pour reconfigurer la connexion hôte Postgres :

```
tsm topology external-services repository replace-host -f <filename>.json -c CACert.pem
```

Dans ce cas, `<filename>.json` est le fichier de connexion décrit dans Configurer Postgres externe, et `CACert.pem` est le fichier de certificat CA pour le certificat SSL/TLS utilisé par Postgres.

## Facultatif : Vérifier la connectivité SSL

Pour vérifier la connectivité SSL, vous devez :

- Installer le client Postgres sur le Nœud 1 de Tableau Server.
- Copier le certificat racine que vous avez créé dans la procédure précédente sur l'hôte Tableau.
- Vous connecter au serveur Postgres à partir du Nœud 1.

## Installer le client Postgres sur le Nœud 1

Cet exemple montre comment installer la version Postgres 13.4. Installez la même version que celle exécutée pour le référentiel externe.

1. Sur le Nœud 1, créez et modifiez le fichier `pgdg.repo` dans le chemin d'accès `/etc/yum.repos.d`. Remplissez le fichier avec les informations de configuration suivantes.

```
[pgdg13]
name=PostgreSQL 13 for RHEL/CentOS 7 - x86_64
baseurl=
1=https://download.postgresql.org/pub/repos/yum/13/redhat/rhel-
7-x86_64
enabled=1
gpgcheck=0
```

2. Installez le client Postgres :

```
sudo yum install postgresql13-13.4-1PGDG.rhel7.x86_64
```

## Copiez le certificat racine sur le Nœud 1

Copiez le certificat CA (`pgsql-rootCACert.pem`) sur l'hôte Tableau :

```
scp ec2-user@<private-DNS-name-of-Postgress-host>:/home/ec2-user/pg-
sql-rootCACert.pem /home/ec2-user
```

## Connectez-vous à l'hôte Postgres sur SSL depuis le Nœud 1

Exécutez la commande suivante à partir du Nœud 1 en spécifiant l'adresse IP de l'hôte du serveur Postgres et le certificat CA racine :

```
psql "postgresql://postgres@<IP-address>:5432/-
postgres?sslmode=verify-ca&sslrootcert=pgsql-rootCACert.pem"
```

Par exemple :

```
psql "post-
gresql://postgres@10.0.1.189:5432/postgres?sslmode=verify-ca&ssl-
rootcert=pgsql-rootCACert.pem"
```

Postgres vous demandera le mot de passe. Une fois la connexion réussie, l'interpréteur de commandes renvoie :



```
psql (13.4)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-
SHA384, bits: 256, compression: off)
Type "help" for help.
postgres=#
```

## Configurer SMTP et les notifications d'événement

Tableau Server envoie des notifications d'événement aux administrateurs et aux utilisateurs. Pour activer cette option, vous devez configurer Tableau Server de manière à ce qu'il envoie des e-mails à votre serveur de messagerie. Vous devez également spécifier les types d'événement, les seuils et les informations d'abonnement que vous souhaitez envoyer.

Pour la configuration initiale de SMTP et des notifications, nous vous recommandons d'utiliser le modèle de fichier de configuration ci-dessous pour créer un fichier json. Vous pouvez également définir toute clé de configuration unique répertoriée ci-dessous en utilisant la syntaxe décrite dans *tsm configuration set* ([Linux](#)).

Exécutez cette procédure sur le Nœud 1 dans votre déploiement Tableau Server :

1. Copiez le modèle json suivant dans un fichier. Personnalisez le fichier avec vos options de configuration SMTP et les notifications d'abonnement et d'alerte pour votre entreprise.
  - Pour voir une liste et une description de toutes les options SMTP, consultez la *Référence de configuration de l'interface en ligne de commande SMTP* ([Linux](#)).
  - Pour afficher une liste et une description de toutes les options d'événement de notification, consultez la section d'interface en ligne de commande *Configurer une notification d'événement serveur* ([Linux](#)).

```
{
 "configKeys": {
 "svcmonitor.notification.smtp.server": "SMTP server host
name",
```

## Guide de déploiement de Tableau Server en entreprise

```
"svcmonitor.notification.smtp.send_account": "SMTP user name",
"svcmonitor.notification.smtp.port": 443,
"svcmonitor.notification.smtp.password": "SMTP user account
password",
"svcmonitor.notification.smtp.ssl_enabled": true,
"svcmonitor.notification.smtp.from_address": "From email
address",
"svcmonitor.notification.smtp.target_addresses": "To email
address1,address2",
"svcmonitor.notification.smtp.canonical_url": "Tableau Server
URL",
"backgrounder.notifications_enabled": true,
"subscriptions.enabled": true,
"subscriptions.attachments_enabled": true,
"subscriptions.max_attachment_size_megabytes": 150,
"svcmonitor.notification.smtp.enabled": true,
"features.DesktopReporting": true,
"storage.monitoring.email_enabled": true,
"storage.monitoring.warning_percent": 20,
"storage.monitoring.critical_percent": 15,
"storage.monitoring.email_interval_min": 25,
"storage.monitoring.record_history_enabled": true
}
}
```

2. Exécutez la commande `tsm settings import -f file.json` pour transmettre le fichier json à Tableau Services Manager.
3. Exécutez la commande `tsm pending-changes apply` pour appliquer les modifications.
4. Exécutez la commande `tsm email test-smtp-connection` pour afficher et vérifier la configuration de la connexion.

# Installer le pilote PostgreSQL

Pour afficher les vues administratives sur Tableau Server, le pilote PostgreSQL doit être installé sur le Nœud 1 du déploiement Tableau Server.

1. Accédez à la page de [téléchargement des pilotes Tableau](#) et copiez l'URL du fichier jar PostgreSQL.
2. Exécutez la procédure suivante sur chaque nœud du déploiement Tableau :

- Créez le chemin de fichier suivant :

```
sudo mkdir -p /opt/tableau/tableau_driver/jdbc
```

- Depuis le nouveau chemin d'accès, téléchargez la version la plus récente du fichier jar PostgreSQL. Par exemple :

```
sudo wget https://-
downloads.tableau.com/drivers/linux/postgresql/postgresql-
42.2.22.jar
```

3. Sur le nœud initial, redémarrez Tableau Server :

```
tsm restart
```

## Configurer la politique de mot de passe fort

Si vous ne déployez pas Tableau Server avec une solution d'authentification d'IdP, nous vous recommandons de renforcer la sécurité de la politique de mot de passe Tableau par défaut.

Si vous déployez Tableau Server avec un IdP, vous devez gérer les politiques de mot de passe avec l'IdP.

La procédure suivante inclut la configuration json pour définir la politique de mot de passe sur Tableau Server. Pour plus d'informations sur les options ci-dessous, consultez *Authentification locale* ([Linux](#)).

1. Copiez le modèle json suivant dans un fichier. Renseignez les valeurs de clé avec votre configuration de politique de mot de passe.

```
{
 "configKeys": {
 "wgserver.localauth.policies.mustcontainletters.enabled":
true,
 "wgserver.localauth.policies.mustcontainuppercase.enabled":
true,
 "wgserver.localauth.policies.mustcontainnumbers.enabled":
true,
 "wgserver.localauth.policies.mustcontainsymbols.enabled":
true,
 "wgserver.localauth.policies.minimumpasswordlength.enabled":
true,
 "wgserver.localauth.policies.minimumpasswordlength.value": 12,
 "wgserver.localauth.policies.maximumpasswordlength.enabled":
false,
 "wgserver.localauth.policies.maximumpasswordlength.value":
255,
 "wgserver.localauth.passwordexpiration.enabled": true,
 "wgserver.localauth.passwordexpiration.days": 90,
 "wgserver.localauth.ratelimiting.maxbackoff.minutes": 60,
 "wgserver.localauth.ratelimiting.maxattempts.enabled": false,
 "wgserver.localauth.ratelimiting.maxattempts.value": 5,
 "features.PasswordReset": true
 }
}
```

2. Exécutez `tsm settings import -f file.json` pour transmettre le fichier json avec les valeurs appropriées à Tableau Services Manager et configurer Tableau Server.
3. Exécutez la commande `tsm pending-changes apply` pour appliquer les modifications.

# Partie 7 - Validation, outils et dépannage

Cette partie comprend les étapes de validation post-installation et des conseils de dépannage.

## Validation du système de basculement

Après avoir configuré votre déploiement, nous vous recommandons d'exécuter des tests de basculement simples pour valider la redondance du système.

Nous vous recommandons d'exécuter les étapes suivantes pour valider la fonctionnalité de basculement :

1. Arrêtez la première instance de passerelle indépendante (TSIG1). Tout le trafic entrant doit être acheminé via la deuxième instance de passerelle indépendante (TSIG2).
2. Redémarrez TSIG1 puis arrêtez TSIG2. Tout le trafic entrant doit passer par TSIG1.
3. Redémarrez TSIG2.
4. Arrêtez le Nœud 1 de Tableau Server. Tout le trafic du service Vizportal/Application basculera vers le Nœud 2.

**Remarque** Depuis septembre 2022, la haute disponibilité du Nœud 1 a été compromise sur certaines versions de Tableau Server 2021.4 et versions ultérieures. Les connexions client échoueront si le Nœud 1 est en panne. Ce problème a été résolu dans ces versions de maintenance :

- 2021.4.15 et versions ultérieures
- 2022.1.11 et versions ultérieures
- 2023.1.3 et versions ultérieures

Pour garantir que votre installation de Tableau Server à l'aide des activations ATR

bénéficiera d'une période de grâce de 72 heures après la panne initiale du nœud, installez ou mettez à niveau vers l'une de ces versions. Pour plus de détails, voir [Tableau Server HA utilisant ATR n'a pas de délai de grâce après la défaillance initiale du nœud](#) dans la Base de connaissances Tableau.

5. Redémarrez le Nœud 1 et arrêtez le Nœud 2. Tout le trafic du service Viz-portal/Application basculera vers le Nœud 1.
6. Redémarrez le Nœud 2.

Dans ce contexte, l'arrêt ou le redémarrage s'effectue en éteignant le système d'exploitation ou la machine virtuelle sans tenter au préalable un arrêt progressif de l'application. L'objectif est de simuler une panne de matériel ou de machine virtuelle.

L'étape de validation minimale pour chaque test de basculement consiste à s'authentifier auprès d'un utilisateur et à effectuer des opérations d'affichage de base.

Vous pouvez obtenir une erreur de navigateur « Bad Request » lorsque vous tentez de vous connecter après un échec simulé. Il est possible que vous voyiez cette erreur même si vous effacez le cache du navigateur. Ce problème se produit souvent lorsque la navigateur met en cache des données d'une session précédente de l'IdP. Si cette erreur persiste même après que vous avez effacé le cache du navigateur local, essayez de valider le scénario Tableau en vous connectant avec un autre navigateur.

## Récupération automatisée du nœud initial

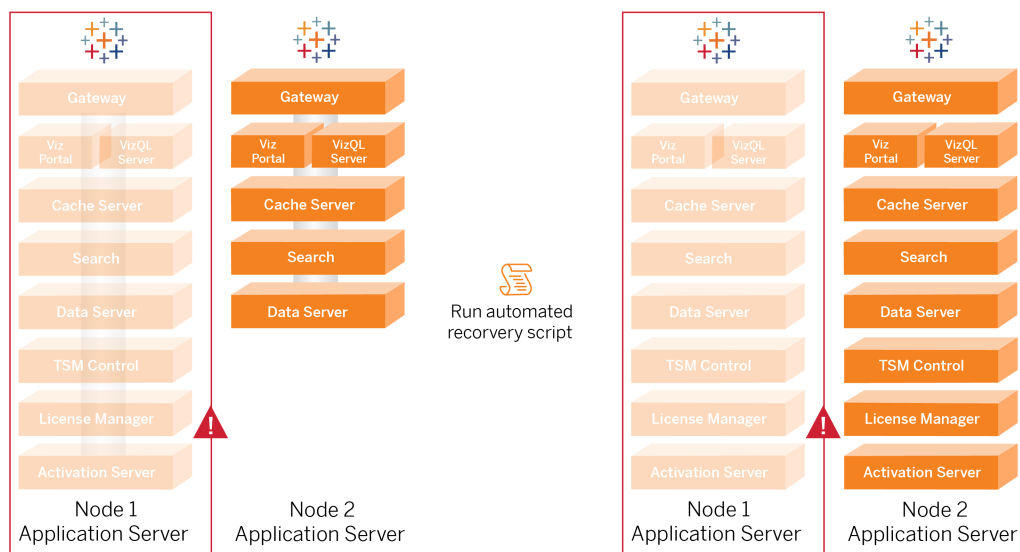
Tableau Server version 2021.2.4 et versions ultérieures inclut un script de récupération automatisée du nœud initial `auto-node-recovery` dans le répertoire des scripts (`/app/-tableau_server/packages/scripts.<version>`).

Si vous rencontrez un problème avec le nœud initial et que des processus redondants s'exécutent sur le Nœud 2, il n'y a aucune garantie que Tableau Server continuera de fonctionner. Tableau Server peut continuer à s'exécuter jusqu'à 72 heures suivant la défaillance d'un nœud initial, avant que l'absence du service de licence n'affecte d'autres processus. Si

## Guide de déploiement de Tableau Server en entreprise

tel est le cas, vos utilisateurs peuvent continuer à se connecter et également voir et utiliser leur contenu après la défaillance du nœud initial. Par contre, vous ne pourrez pas reconfigurer Tableau Server parce que vous n'aurez pas accès au contrôleur d'administration.

Même lorsqu'il est configuré avec des processus redondants, il est possible que Tableau Server cesse de fonctionner après l'échec du nœud initial.



Pour récupérer en cas d'échec du nœud initial (Nœud 1) :

1. Connectez-vous au Nœud 2 de Tableau Server.
2. Accédez au répertoire de scripts :

```
cd /app/tableau_server/packages/scripts.<version>
```

3. Exécutez la commande suivante pour lancer le script :

```
sudo ./auto-node-recovery -p node1 -n node2 -k <license keys>
```

Où `<license keys>` est une liste séparée par des virgules (sans espaces) des clés produit pour votre déploiement. Si vous n'avez pas accès à vos clés produit, accédez au [portail client Tableau](#) pour les récupérer. Par exemple :

```
sudo ./auto-node-recovery -p node1 -n node2 -k TSB4-8675-309F-TW50-9RUS,TSNM-559N-ULL6-22VE-SIEN
```

Le script `auto-node-recovery` exécutera environ 20 étapes pour récupérer les services sur le Nœud 2. Chaque étape est affichée dans le terminal au fur et à mesure de la progression du script. Un statut plus détaillé est enregistré dans `/data/tableau_data/logs/app-controller-move.log`. Dans la plupart des environnements, l'exécution du script prend entre 35 et 45 minutes.

## Résolution des problèmes de récupération du nœud initial

Si la récupération de nœud échoue, vous pourrez trouver utile d'exécuter le script de manière interactive pour autoriser ou interdire des étapes discrètes du processus. Par exemple, si le script échoue au cours du processus, vous pouvez consulter le fichier journal, apporter des modifications à la configuration, puis réexécuter le script. Si vous exécutez le mode interactif, vous pouvez alors ignorer toutes les étapes jusqu'à ce que vous arriviez à l'étape qui a échoué.

Pour une exécution en mode interactif, ajoutez le commutateur `-i` pour passer à l'argument de script.

## Reconstitution du nœud défaillant

Après avoir exécuté le script, le Nœud 2 exécutera tous les services qui étaient auparavant sur l'hôte du Nœud 1 défaillant. Pour ajouter le Nœud 4, vous devez déployer un nouvel hôte Tableau Server avec le fichier bootstrap et le configurer comme vous l'avez fait pour le Nœud 2 d'origine, comme spécifié dans la Partie 4. Consultez Configurer le Nœud 2.



# switchto

Le script switchto est un script conçu par Tim qui permet de passer facilement d'une fenêtre à l'autre.

1. Copiez le code suivant dans un fichier appelé `switchto` dans le répertoire de base de votre hôte Bastion.

```
#!/bin/bash
#-----
switchto
#
Helper function to simplify SSH into the various AWS hosts
when
following the Tableau Server Enterprise Deployment Guide
(EDG) .
#
Place this file on your bastion host and provide your AWS
hosts'
internal ip addresses or machine names here.
Example: readonly NODE1="10.0.3.187"
#
readonly NODE1=""
readonly NODE2=""
readonly NODE3=""
readonly NODE4=""
readonly PGSQL=""
readonly PROXY1=""
readonly PROXY2=""

usage() {
echo "Usage: switchto.sh [node1 | node2 | node3 | node4 |
pgsql | proxy1 | proxy2]"
}
```

```

ip=""

case $1 in
 node1)
 ip="$NODE1"
 ;;
 node2)
 ip="$NODE2"
 ;;
 node3)
 ip="$NODE3"
 ;;
 node4)
 ip="$NODE4"
 ;;
 pgsql)
 ip="$PGSQL"
 ;;
 proxy1)
 ip="$PROXY1"
 ;;
 proxy2)
 ip="$PROXY2"
 ;;
 ?)
 usage
 exit 0
 ;;
 *)
 echo "Unkown option $1."
 usage
 exit 1
 ;;

```

```
esac
```

```
if [[-z $ip]]; then
echo "You must first edit this file to provide the ip addresses
of your AWS hosts."
exit 1
fi
```

```
ssh -A ec2-user@$ip
```

2. Mettez à jour les adresses IP dans le script pour les mapper à vos instances EC2, puis enregistrez le fichier.
3. Appliquez les autorisations au fichier de script :

```
sudo chmod +x switchto
```

Utilisation :

Pour passer à un hôte, exécutez la commande suivante :

```
./switchto <target>
```

Par exemple, pour passer au Nœud 1, exécutez la commande suivante :

```
./switchto node1
```

## Résolution des problèmes Tableau Server Independent Gateway

La configuration de la passerelle indépendante, d'Okta, de Mellon et de SAML sur Tableau Server peut être un processus sujet aux erreurs. La cause première la plus courante des échecs est une erreur de chaîne. Par exemple, une barre oblique finale ( / ) sur les URL Okta spécifiées lors de la configuration peut provoquer une erreur de non-concordance liée à l'assertion SAML. Ceci n'est qu'un exemple. Lors de la configuration, il existe de nombreuses occasions de saisir une chaîne incorrecte dans l'une des applications.

## Redémarrer le service tableau-tsig

Commencez (et terminez) toujours le dépannage en redémarrant le service tableau-tsig sur les ordinateurs de la passerelle indépendante. Le redémarrage de ce service est rapide et déclenche souvent le chargement de la configuration mise à jour à partir de Tableau Server.

Exécutez les commandes suivantes pour mettre à jour l'ordinateur de la passerelle indépendante :

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

## Trouver les chaînes incorrectes

Si vous avez fait une erreur de chaîne (erreur de copier/coller, chaîne tronquée, etc.), prenez le temps de parcourir chacun des paramètres que vous avez configurés :

- Configuration de pré-authentification Okta. Examinez attentivement les URL que vous avez définies. Recherchez les barres obliques finales. Vérifiez HTTP versus HTTPS.
- Historique du shell pour la configuration SAML sur le Nœud 1. Vérifiez la commande `tsm authentication saml configure` que vous avez exécutée. Vérifiez que toutes les URL correspondent à celles que vous avez configurées dans Okta. Pendant que vous examinez l'historique du shell à partir du Nœud 1, vérifiez que les commandes `tsm configuration set` qui spécifient les chemins d'accès aux fichiers de configuration Mellon correspondent exactement aux chemins d'accès des fichiers où vous avez copié les fichiers sur la passerelle indépendante.
- Configuration de Mellon sur la passerelle indépendante. Dans l'historique du shell, vérifiez que vous avez créé les métadonnées avec la même chaîne d'URL que celle que vous avez configurée dans Okta et Tableau SAML. Vérifiez que tous les chemins spécifiés dans `/etc/mellon/conf.d/global.conf` sont corrects et que `MellonCookieDomain` est défini sur votre domaine racine, et non sur votre sous-domaine Tableau.

## Recherche dans les journaux pertinents

Si toutes les chaînes semblent correctement définies, vous devez inspecter les journaux pour détecter les erreurs.

Tableau Server consigne les erreurs et les événements dans des dizaines de fichiers journaux différents. La passerelle indépendante se connecte également à un ensemble de fichiers locaux. Nous vous recommandons d'inspecter ces journaux dans l'ordre suivant.

### Fichiers journaux de la passerelle indépendante

L'emplacement par défaut des journaux de la passerelle indépendante est `/var/opt/tableau/tableau_tsig/logs`

- `access.log` : ce journal est utile dans la mesure où il contient des entrées qui affichent les connexions à partir des nœuds Tableau Server. Si vous obtenez des erreurs de passerelle (ne démarre pas) lorsque vous essayez de démarrer TSM et qu'il n'y a aucune entrée dans le fichier `access.log`, il y a un problème de connectivité de base. Vérifiez toujours la configuration du groupe de sécurité AWS dans un premier temps. Un autre problème courant est une faute de frappe dans `tsig.json`. Si vous effectuez une mise à jour de `tsig.json`, exécutez `tsm stop` avant d'exécuter `tsm topology external-services gateway update -c tsig.json`. Une fois `tsig.json` mis à jour, exécutez `tsm start`.
- `error.log` : entre autres entrées, ce journal inclut les erreurs SAML et Mellon.

### Fichier journal de Tableau Server tabadminagent

L'ensemble de fichiers `tabadminagent` (et non `tabadmincontroller`) inclut les seuls fichiers journaux pertinents pour le dépannage des erreurs liées à la passerelle indépendante.

Vous devez trouver où les erreurs de passerelle indépendante ont été consignées dans `tabdminagent`. Ces erreurs peuvent se trouver sur n'importe quel nœud, mais elles ne se trouvent que sur un seul nœud. Effectuez les étapes suivantes sur chaque nœud du cluster Tableau Server jusqu'à ce que vous trouviez la chaîne « independent » :

1. Localisez l'emplacement du fichier journal tabadminagent sur les nœuds 1 à 4 de Tableau Server dans la configuration EDG :

```
cd /data/tableau_data/data/tabsvc/logs/tabadminagent
```

2. Ouvrez le dernier journal pour lire :

```
less tabadminagent_nodeN.log
```

(remplacez N par le numéro de nœud)

3. Recherchez toutes les instances de « Independent » et « independent » en utilisant la chaîne de recherche suivante :

```
/ndependent
```

S'il n'y a pas de correspondance, passez au nœud suivant et répétez les étapes 1 à 3.

4. Lorsque vous obtenez une correspondance : `Shift + G` pour déplacer vers le bas et obtenir les derniers messages d'erreur.

## Recharger le fichier stub httpd

La passerelle indépendante gère la configuration du protocole httpd pour Apache. Une opération générique qui résout souvent les problèmes transitoires consiste à recharger le fichier stub httpd qui amorce la configuration Apache sous-jacente. Exécutez les commandes suivantes sur les deux instances de la passerelle indépendante.

1. Copiez le fichier stub dans httpd.conf :

```
cp /var/opt/tableau/tableau_tsig/config/httpd.conf.stub
/var/opt/tableau/tableau_tsig/config/httpd.conf
```

2. Redémarrez le service de passerelle indépendante :

```
sudo su - tableau-tsig
systemctl --user restart tsig-httpd
exit
```

## Supprimer ou déplacer des fichiers journaux

La passerelle indépendante enregistre tous les événements d'accès. Vous devrez gérer le stockage des fichiers journaux pour éviter de remplir l'espace disque. Si votre disque est plein, la passerelle indépendante ne pourra pas écrire d'événements d'accès et le service échouera.

Le message suivant sera consigné dans `error.log` sur la passerelle indépendante :

```
(28)No space left on device: [client 10.0.2.209:54332] AH00646:
Error writing to /var/opt/tableau/tableau_tsig/-
logs/access.%Y_%m_%d_%H_%M_%S.log
```

Cet échec se traduira par un état `DEGRADED` pour le nœud `external` lorsque vous exécutez `tsm status -v` sur le Nœud 1 de Tableau. Le nœud `external` dans la sortie d'état fait référence à la passerelle indépendante.

Pour résoudre ce problème, supprimez ou déplacez les fichiers `access.log` hors du disque.

Les fichiers `access.log` sont enregistrés sur `/var/opt/tableau/tableau_tsig/logs`.

Après avoir effacé le disque, redémarrez le service `tableau-tsig`.

## Erreurs de navigateur

**Requête incorrecte** : une erreur courante pour ce scénario est une erreur « Bad Request » (Requête incorrecte) d'Okta. Ce problème se produit souvent lorsque le navigateur met en cache les données de la session Okta précédente. Par exemple, si vous gérez les applications Okta en tant qu'administrateur Okta, puis tenez d'accéder à Tableau à l'aide d'un autre compte compatible Okta, les données de session provenant des données de l'administrateur peuvent provoquer l'erreur « Bad Request ». Si cette erreur persiste même après la suppression du cache du navigateur local, essayez de valider le scénario Tableau en vous connectant avec un autre navigateur.

Une autre cause de l'erreur « Bad Request » est une faute de frappe dans l'une des nombreuses URL que vous entrez lors des processus de configuration Okta, Mellon et SAML. Vérifiez que vous avez entré tous ces éléments sans erreur.

Souvent le fichier `error.log` sur le serveur de la passerelle indépendante spécifie l'URL à l'origine de l'erreur.

**Introuvable - L'URL demandée était introuvable sur ce serveur** : cette erreur indique l'une des nombreuses erreurs de configuration possibles.

Si l'utilisateur est authentifié avec Okta, puis reçoit cette erreur, il est probable que vous ayez téléchargé l'application de pré-authentification Okta sur Tableau Server lorsque vous avez configuré SAML. Vérifiez que vous avez configuré les métadonnées de l'application Okta Tableau Server sur Tableau Server, et non pas les métadonnées de l'application de pré-authentification Okta

Autres étapes de résolution des problèmes :

- Passez en revue les paramètres de l'application de pré-authentification Okta. Assurez-vous que les protocoles HTTP vs HTTPS sont définis comme spécifié dans cette rubrique.
- Redémarrez `tsig-httpd` sur les deux serveurs de passerelle indépendants.
- Vérifiez que `sudo apachectl configtest` renvoie « Syntax OK » sur les deux passerelles indépendantes.
- Vérifiez que l'utilisateur test est affecté aux deux applications dans Okta.
- Vérifiez que l'adhérence est activée sur l'équilibreur de charge et les groupes cibles associés

## Vérifier la connexion TLS de Tableau Server à la passerelle indépendante

Utilisez la commande `wget` pour vérifier la connectivité et l'accès de Tableau Server à la passerelle indépendante. Des variantes de cette commande peuvent vous aider à comprendre si des problèmes de certificat causent des problèmes de connexion.

Par exemple, lancez la commande `wget` pour vérifier le protocole de maintenance (HK) à partir de Tableau Server ;

```
wget https://ip-10-0-1-38.us-west-1.compute.internal:21319
```



## Guide de déploiement de Tableau Server en entreprise

Créez l'URL avec la même adresse d'hôte que celle que vous avez incluse pour l'option d'hôte du fichier `tsig.json`. Spécifiez le protocole `https` et ajoutez l'URL avec le port HK 21319.

Pour vérifier la connectivité et ignorer la vérification du certificat :

```
wget https://ip-10-0-1-38.us-west-1.compute.internal:21319 --no-check-certificate
```

Pour vérifier que le certificat CA racine pour TSIG est valide :

```
wget https://ip-10-0-1-38.us-west-1.compute.internal:21319 --ca-certificate=tsigRootCA.pem
```

Si Tableau est capable de communiquer, il peut y avoir des erreurs liées au contenu, mais il n'y aura pas d'erreurs liées à la connexion. Si Tableau ne parvient pas du tout à se connecter, commencez par vérifier la configuration du protocole dans les groupes de pare-feu/sécurité. Par exemple, les règles entrantes du groupe de sécurité où réside la passerelle indépendante doivent autoriser TCP 21319.

# Annexe - Boîte à outils du déploiement AWS

Cette rubrique présente des outils et d'autres options de déploiement pour l'architecture de référence lorsqu'elle est déployée dans AWS. Plus précisément, cette rubrique décrit comment automatiser l'exemple de déploiement AWS décrit dans EDG.

## Script d'installation automatisée TabDeploy4EDG

Le script `TabDeploy4EDG` automatise la mise en œuvre du déploiement Tableau sur quatre nœuds décrit dans la Partie 4 - Installer et configurer Tableau Server. Si vous suivez l'exemple d'implémentation AWS décrit dans ce guide, vous pourrez peut-être exécuter le script `TabDeploy4EDG`.

**Exigences.** Pour exécuter le script, vous devez préparer et configurer l'environnement AWS conformément à l'exemple d'implémentation décrit dans Partie 3 - Préparer le déploiement de Tableau Server en entreprise :

- Le VPC, le sous-réseau et les groupes de sécurité ont été configurés comme décrit. Les adresses IP ne doivent pas nécessairement correspondre à celles qui sont affichées dans l'exemple d'implémentation.
- Quatre instances EC2 exécutant les dernières versions mises à jour d'AWS Linux 2
- PostgreSQL est installé et a été configuré comme décrit dans Installer, configurer PostgreSQL et créer une sauvegarde tar.
- Un fichier de sauvegarde tar de l'étape 1 se trouve sur l'instance EC2 où PostgreSQL est installé, comme décrit dans Effectuer une sauvegarde tar PostgreSQL de l'Étape 1.
- L'instance EC2 qui exécutera le Nœud 1 du déploiement de Tableau Server a été configurée pour communiquer avec PostgreSQL comme décrit dans Partie 4 - Installer et configurer Tableau Server.

## Guide de déploiement de Tableau Server en entreprise

- Vous vous êtes connecté à chaque instance EC2 avec une session SSH depuis l'hôte bastion.

Il faut compter environ 1,5 à 2 heures pour installer et configurer les quatre serveurs Tableau à l'aide du script. Le script configure Tableau en fonction des paramètres prescrits de l'architecture de référence. Le script effectue les actions suivantes :

- Restaure la sauvegarde de l'Étape 1 de l'hôte PostgreSQL si vous spécifiez un chemin vers le fichier tar de l'hôte PostgreSQL.
- Oblitère les installations Tableau existantes sur tous les nœuds.
- Exécute `sudo yum update` sur tous les nœuds.
- Télécharge et copie le package rpm Tableau sur chaque nœud.
- Télécharge et installe les dépendances sur chaque nœud.
- Crée `/app/tableau_server` et installe le package sur tous les nœuds.
- Installe le Nœud 1 avec un magasin d'identités local et configure le référentiel externe avec PostgreSQL.
- Effectue l'installation du fichier bootstrap et la configuration initiale du Nœud 2 - Nœud 4.
- Supprime le fichier bootstrap et le fichier de configuration pour TabDeploy4EDG.
- Configure les services dans le cluster Tableau selon les spécifications de l'architecture de référence.
- Valide l'installation et renvoie l'état de chaque nœud.

### Télécharger et copier le script sur l'hôte bastion

1. Copiez le script de la [page d'exemples TabDeploy4EDG](#) et collez le code dans un fichier appelé `TabDeploy4EDG`.
2. Enregistrez le fichier dans le répertoire de base sur l'hôte EC2 qui sert d'hôte bastion.
3. Exécutez la commande suivante pour modifier le mode du fichier afin de le rendre exécutable :

```
sudo chmod +x TabDeploy4EDG
```

### Exécuter TabDeploy4EDG

Vous devez exécuter `TabDeploy4EDG` depuis l'hôte bastion. Le script a été écrit selon l'hypothèse d'une exécution dans le contexte de l'agent de transfert ssh comme décrit dans

Exemple : connexion à l'hôte bastion dans AWS. Si l'exécution n'a pas lieu dans le contexte de l'agent de transfert ssh, vous serez invité à saisir des mots de passe tout au long du processus d'installation.

1. Créez, modifiez et enregistrez un fichier d'enregistrement (`registration.json`). Le fichier doit être un fichier json correctement formaté. Copiez et personnalisez le modèle suivant :

```
{
 "zip" : "97403",
 "country" : "USA",
 "city" : "Springfield",
 "last_name" : "Simpson",
 "industry" : "Energy",
 "eula" : "yes",
 "title" : "Safety Inspection Engineer",
 "phone" : "5558675309",
 "company" : "Example",
 "state" : "OR",
 "department" : "Engineering",
 "first_name" : "Homer",
 "email" : "homer@example.com"
}
```

2. Exécutez la commande suivante pour générer un fichier de configuration de modèle :

```
./TabDeploy4EDG -g edg.config
```

3. Ouvrez le fichier de configuration à modifier :

```
sudo nano edg.config
```

Au minimum, vous devez ajouter les adresses IP de chaque hôte EC2, un chemin d'accès du fichier d'enregistrement et une clé produit valide.

4. Lorsque vous avez terminé de modifier le fichier de configuration, enregistrez-le, puis fermez-le.

5. Pour exécuter TabDeploy4EDG, exécutez la commande suivante :

```
./TabDeploy4EDG -f edg.config
```

## Exemple : Automatiser le déploiement de l'infrastructure AWS avec Terraform

Cette section décrit comment configurer et exécuter Terraform pour déployer l'architecture de référence EDG dans AWS. L'exemple de configuration Terraform présenté ici déploie un VPC AWS avec les sous-réseaux, les groupes de sécurité et les instances EC2 décrits dans Partie 3 - Préparer le déploiement de Tableau Server en entreprise.

Des exemples de modèles Terraform sont disponibles sur le site Web Tableau Samples à l'adresse <https://help.tableau.com/samples/en-us/edg/edg-terraform.zip>. Ces modèles doivent être configurés et personnalisés pour votre organisation. Le contenu de configuration fourni dans cette section décrit les modifications de modèle minimales requises que vous devez personnaliser pour le déploiement.

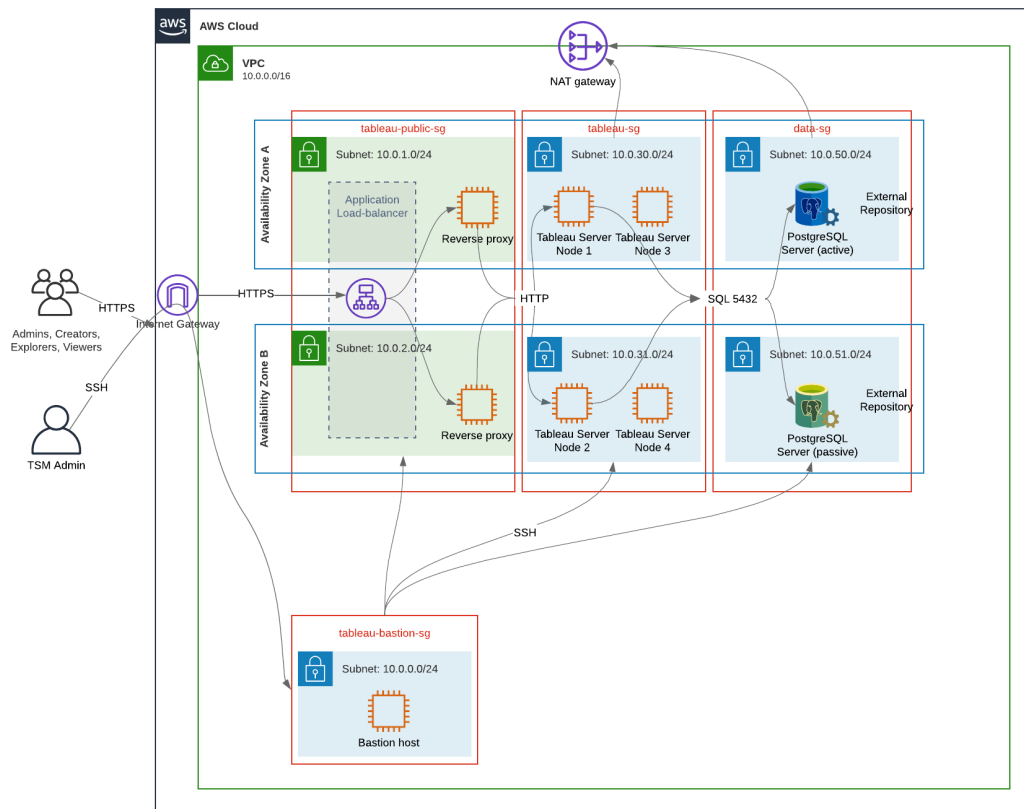
### Objectif

Les modèles et le contenu Terraform fournis ici sont conçus pour fournir un exemple de travail qui vous permettra de déployer EDG rapidement dans un environnement de test de développement.

Nous avons tout mis en œuvre pour tester et documenter l'exemple de déploiement de Terraform. Cependant, l'utilisation de Terraform pour déployer et maintenir EDG dans un environnement de production nécessitera une expertise Terraform qui dépasse le cadre de cet exemple. Tableau ne fournit pas de support pour l'exemple de solution Terraform documenté ici.

## État final

Suivez la procédure de cette section pour configurer un VPC dans AWS dont le fonctionnement est équivalent à celui du VPC spécifié dans Partie 3 - Préparer le déploiement de Tableau Server en entreprise.



Exemples de modèles Terraform et contenu associé de cette section :

- Création d'un VPC avec une adresse IP élastique, deux zones de disponibilité et une organisation de sous-réseaux comme indiqué ci-dessus (les adresses IP sont différentes)
- Création des groupes de sécurité Bastion, Public, Privé et Données.
- Définition de la plupart des règles d'entrée et de sortie sur les groupes de sécurité. Vous devrez modifier les groupes de sécurité après l'exécution de Terraform.

- Création des hôtes EC2 suivants (chacun exécutant AWS Linux2) : bastion, proxy 1, proxy 2, Tableau node 1, Tableau node 2, Tableau node 3, Tableau node 4.
- Les hôtes EC2 pour PostgreSQL ne sont pas créés. Vous devez créer EC2 manuellement dans le groupe de sécurité des données, puis installer et configurer PostgreSQL comme décrit dans Installer, configurer PostgreSQL et créer une sauvegarde tar.

## Exigences

- Compte AWS : vous devez avoir accès à un compte AWS qui vous permet de créer des VPC.
- Si vous exécutez Terraform à partir d'un ordinateur Windows, vous devrez installer l'interface en ligne de commande AWS.
- Une adresse IP élastique disponible dans votre compte AWS.
- Un domaine enregistré dans AWS Route 53. Terraform créera une zone DNS et les certificats SSL associés dans Route 53. Par conséquent, le profil sous lequel Terraform s'exécute doit également disposer des autorisations appropriées dans Route 53.

## Avant de commencer

- Les exemples de ligne de commande de cette procédure concernent Terminal avec Apple OS. Si vous exécutez Terraform sous Windows, vous devrez peut-être adapter les commandes avec les chemins d'accès aux fichiers, le cas échéant.
- Un projet Terraform est composé de nombreux fichiers de configuration texte (extension de fichier .tf). Vous configurez Terraform en personnalisant ces fichiers. Si vous ne disposez pas d'un éditeur de texte robuste, installez Atom ou Text++.
- Si vous partagez le projet Terraform avec d'autres personnes, nous vous recommandons de stocker le projet dans Git pour la gestion des modifications.

## Étape 1 - Préparer l'environnement

### A. Télécharger et installer Terraform

<https://www.terraform.io/downloads>

## B. Générer une paire de clés privée-publique

Il s'agit de la clé que vous utiliserez pour accéder à AWS et à l'environnement VPC résultant. Lorsque vous exécutez Terraform, vous incluez la clé publique.

Ouvrez le Terminal et entrez les commandes suivantes :

1. Create a private key. For example, `my-key.pem`:

```
openssl genrsa -out my-key.pem 1024
```

2. Créez une clé publique. Ce format de clé n'est pas utilisé pour Terraform. Vous la convertirez en clé ssh plus tard dans cette procédure :

```
openssl rsa -in my-key.pem -pubout > my-key.pub
```

3. Définissez les autorisations pour la clé privée :

```
sudo chmod 0600 my-key.pem
```

Pour définir des autorisations sous Windows :

- Localisez le fichier dans l'Explorateur Windows, faites un clic droit dessus puis sélectionnez **Propriétés**. Accédez à l'onglet **Sécurité**, puis cliquez sur **Avancé**.
- Modifiez le propriétaire sur votre nom, désactivez l'héritage et supprimez toutes les autorisations. Accordez-vous le **Contrôle total**, puis cliquez sur **Enregistrer**. Marquez le fichier en lecture seule.

4. Créez une clé publique ssh. Il s'agit de la clé que vous copierez dans Terraform plus tard dans le processus.

```
ssh-keygen -y -f my-key.pem >my-key-ssh.pub
```

## C. Télécharger le projet et ajouter le répertoire d'état

1. Téléchargez et décompressez le [projet EDG Terraform](#) et enregistrez-les sur votre ordinateur local. Après avoir décompressé le téléchargement, vous aurez un répertoire de niveau supérieur, `edg-terraform`, et une série de sous-répertoires.



2. Créez un répertoire appelé `state`, en tant qu'homologue du répertoire supérieur `edg-terraform`.

## Étape 2 : Personnaliser les modèles Terraform

Vous devez personnaliser les modèles Terraform en fonction de votre environnement AWS et EDG. Le présent exemple fournit les personnalisations de modèle minimales que la plupart des organisations devront effectuer. Il est probable que votre environnement particulier nécessitera d'autres personnalisations.

Cette section est organisée par nom de modèle.

Assurez-vous d'enregistrer toutes les modifications avant de passer à l'*Étape 3 - Exécuter Terraform*.

### versions.tf

There are three instances of `versions.tf` files where the `required_version` field must match the version of `terraform.exe` you're using. Check the version of `terraform` (`terraform.exe -version`) and update each of the following instances:

- `edg-terraform\versions.tf`
- `edg-terraform\modules\proxy\versions.tf`
- `edg-terraform\modules\tableau_instance\versions.tf`

### key-pair.tf

1. Ouvrez la clé publique que vous avez générée à l'étape 1B et copiez-la :

```
less my-key-ssh.pub
```

Windows : Copiez le contenu de votre clé publique.

2. Copiez la chaîne de clé publique dans l'argument `public_key`, par exemple :

```
resource "aws_key_pair" "tableau" {
 key_name = "my-key"
```

```
public_key = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQ (truncated
example) dZVHambOCw=="
```

Ensure that the `key_name` value is unique in the datacenter or terraform apply will fail.

### locals.tf

Update `user.owner` to your name or alias. The value you enter here will be used for the "Name" tag in AWS on the resources that Terraform creates.

### providers.tf

1. Ajoutez des balises selon les exigences de votre organisation. Par exemple :

```
default_tags {
 tags = {

 "Application" = "tableau",
 "Creator" = "alias@example.com",
 "DeptCode" = "8675309",
 "Description" = "EDG",
 "Environment" = "test",
 "Group" = "itcloud@example.com"
 }
}
```

2. If using provider, comment out the `assume_role` lines:

```
/* assume_role {
 role_arn = "arn:aws:iam::310946706895:role/terraform-
backend"
 session_name = "terraform"
}*/
```

### elb.tf

Under `'resource "aws_lb" "tableau" {'` choose a unique value to use for `name` and `tags.Name`.

## Guide de déploiement de Tableau Server en entreprise

If another AWS load balancer has the same name in the datacenter, then `terraform apply` will fail.

Add `idle_timeout`:

```
resource "aws_lb" "tableau" {
 name = "edg-again-alb"
 load_balancer_type = "application"
 subnets = [for subnet in aws_subnet.public : subnet.id]
 security_groups = [aws_security_group.public.id]
 drop_invalid_header_fields = true
 idle_timeout = 400
 tags = {
 Name = "edg-again-alb"
 }
}
```

### variables.tf

Mettez à jour le nom de domaine racine. Ce nom doit correspondre au domaine que vous avez enregistré dans Route 53.

```
variable "root_domain_name" {
 default = "example.com"
}
```

Par défaut, le sous-domaine `tableau` est indiqué comme nom de domaine DNS VPC. Pour le modifier, mettez à jour `subdomain`:

```
variable "subdomain" {
 default = "tableau"
}
```

### modules/tableau\_instance/ec2.tf

There are two `ec2.tf` files in the project. This customization is for the Tableau instance of the `ec2.tf` in the directory: `modules/tableau_instance/ec2.tf`.

- Si nécessaire, ajoutez des balises blob :

```
tags = {
 "Name" : var.ec2_name,
 "user.owner" = "ALIAS",
 "Application" = "tableau",
 "Creator" = "ALIAS@example.com",
 "DeptCode" = "8675309",
 "Description" = "EDG",
 "Environment" = "test",
 "Group" = "itcloud@example.com"
}
```

- Si nécessaire, mettez éventuellement à jour votre espace de stockage pour gérer vos besoins en données :

Volume racine :

```
root_block_device {
 volume_size = 100
 volume_type = "gp3"
}
```

Volume d'applications :

```
resource "aws_ebs_volume" "tableau" {
 availability_zone = data.aws_subnet.tableau.availability_zone
 size = 500
 type = "gp3"
}
```

## Étape 3 - Exécuter Terraform

### A. Initialiser Terraform

Dans le Terminal, passez au répertoire `edg-terraform` et exécutez la commande suivante :

```
terraform init
```

Si l'initialisation réussit, passez à l'étape suivante. Si l'initialisation a échoué, suivez les instructions de la sortie Terraform.

### B. Planifier Terraform

Depuis le même répertoire, exécutez la commande `plan` :

```
terraform plan
```

Cette commande peut être exécutée plusieurs fois. Exécutez-la autant de fois que nécessaire pour corriger les erreurs. Une fois que cette commande s'exécute sans erreur, passez à l'étape suivante.

### C. Appliquer Terraform

Depuis le même répertoire, exécutez la commande `apply` :

```
terraform apply
```

Terraform will prompt you to verify deployment, type `Yes`.

### Facultatif : Supprimer Terraform

Vous pouvez supprimer l'intégralité du VPC en exécutant la commande `destroy` :

```
terraform destroy
```

La commande `destroy` ne détruira que ce qu'elle a créé. Si vous avez apporté des modifications manuelles à certains objets dans AWS (par exemple, des groupes de sécurité, des sous-réseaux, etc.), la commande `destroy` échouera. Pour sortir d'une opération de

destruction en échec/suspendue, tapez `Control + C`. Vous devez ensuite nettoyer manuellement le VPC pour le remettre dans l'état où il se trouvait lorsque Terraform l'avait créé à l'origine. Vous pouvez alors exécuter la commande `destroy`.

## Étape 4 - Connexion à bastion

Toutes les connexions en ligne de commande se font via l'hôte bastion sur TCP 22 (protocole SSH).

1. Dans AWS, créez une règle entrante dans le groupe de sécurité bastion (**AWS > Groupes de sécurité > Bastion SG > Modifier les règles entrantes**) et créez une règle pour autoriser les connexions SSH (TCP 22) à partir de l'adresse IP ou du masque de sous-réseau où vous exécuterez les commandes Terminal .

Facultatif : vous trouverez peut-être utile d'autoriser la copie de fichiers entre les instances EC2 dans les groupes privé et public pendant le déploiement. Créez des règles SSH entrantes :

- Privé : créez une règle entrante pour autoriser SSH depuis Public
- Public : créer une règle entrante pour autoriser SSH depuis le privé et depuis le public

2. Utilisez la clé pem que vous avez créée à l'étape 1.B pour vous connecter à l'hôte bastion :

### Sur le Terminal Mac :

Exécutez les commandes suivantes à partir du répertoire sur lequel la clé pem est stockée :

```
ssh-add -apple-use-keychain <keyName>.pem
```

If you get a warning about private key being accessible by others, then run this command: `chmod 600 <keyName>.pem` and then run the `ssh-add` command again.

Connect to the bastion host with this command: `ssh -A ec2-user@IPaddress`

For example: `ssh -A ec2-user@3.15.12.112.`

### Sous Windows avec PuTTY et Pageant :

- a. Créer ppk à partir de la clé pem : utilisez le générateur de clé PuTTY. Chargez la clé pem que vous avez créée à l'Étape 1.B. Une fois la clé importée, cliquez sur **Enregistrer la clé privée**. Cela crée un fichier ppk.
- b. Dans PuTTY : ouvrez la configuration et apportez les modifications suivantes :
  - Sessions>Nom d'hôte : ajoutez l'adresse IP de l'hôte bastion.
  - Sessions>Port : 22
  - Connexion> Données> Nom d'utilisateur de connexion automatique : ec2-user
  - Connexion>SSH>Auth>Autoriser le transfert de l'agent
  - Connexion>SSH>Auth> Pour la clé privée, cliquez sur Parcourir et sélectionnez le fichier .ppk que vous venez de créer.
- c. Installez Pageant et chargez le ppk dans l'application.

## Étape 5 - Installer PostgreSQL

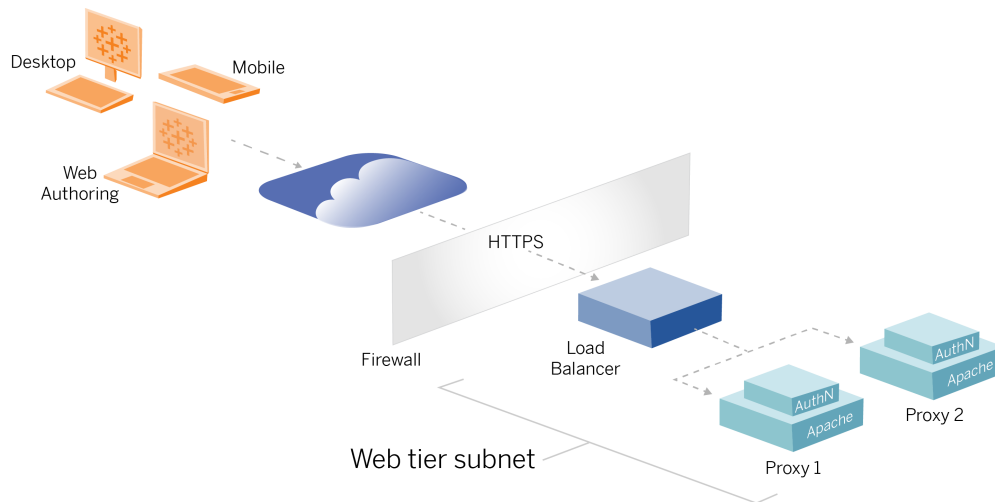
Le modèle Terraform n'installe pas PostgreSQL pour une utilisation en tant que référentiel externe. Par contre, le groupe de sécurité et le sous-réseau associés sont créés. Si vous installez le référentiel externe sur une instance EC2 exécutant PostgreSQL, vous devez déployer l'instance EC2 comme décrit dans Partie 3 - Préparer le déploiement de Tableau Server en entreprise.

Ensuite, installez, configurez PostgreSQL et sauvegardez-le au format tar comme décrit dans Partie 4 - Installer et configurer Tableau Server.

## Étape 6 - (Facultatif) Exécuter DeployTab4EDG

Le script TabDeploy4EDG automatise l'exemple d'installation du déploiement Tableau à quatre nœuds qui est décrit dans la Partie 4. Voir Script d'installation automatisée TabDeploy4EDG.

# Annexe - Niveau Web avec exemple de déploiement Apache



La présente rubrique fournit une procédure de bout en bout qui décrit comment implémenter un niveau Web dans l'architecture AWS de référence. L'exemple de configuration est composé des composants suivants :

- Équilibreur de charge d'application AWS
- Serveurs proxy Apache
- Module d'authentification Mellon
- IdP Okta
- Authentification SAML

**Remarque** : l'exemple de configuration de niveau Web présenté dans cette section comprend des procédures détaillées pour le déploiement de logiciels et de services tiers. Nous avons tout mis en œuvre pour vérifier et documenter les procédures permettant d'activer le scénario de niveau Web. Il peut toutefois arriver que le logiciel tiers change ou que votre scénario diffère de l'architecture de référence décrite ici. Veuillez vous



référer à la documentation du fournisseur tiers pour les détails de configuration et l'assistance.

Les exemples Linux tout au long de cette section montrent des commandes pour les distributions de type RHEL. Plus précisément, les commandes présentées ici ont été développées avec la distribution Amazon Linux 2. Si vous exécutez la distribution Ubuntu, modifiez les commandes en conséquence.

Le déploiement du niveau Web dans cet exemple suit une procédure de configuration et de vérification par étapes. La configuration du niveau Web principal comprend les étapes suivantes pour activer HTTP entre Tableau et Internet. Apache est exécuté et configuré pour un proxy inverse/équilibrage de charge derrière l'équilibreur de charge d'application AWS :

1. Installer Apache
2. Configurez le serveur proxy inverse pour tester la connectivité à Tableau Server
3. Configurer l'équilibrage de charge sur le proxy
4. Configurer l'équilibreur de charge d'application AWS

Une fois que vous avez configuré le niveau Web et vérifié la connectivité avec Tableau vérifiée, configurez l'authentification avec un fournisseur externe.

## Installer Apache

Exécutez la procédure suivante sur les deux hôtes EC2 (Proxy 1 et Proxy 2). Si vous effectuez un déploiement dans AWS conformément à l'exemple d'architecture de référence, vous devez avoir deux zones de disponibilité et exécuter un seul serveur proxy dans chaque zone.

1. Installez Apache :

```
sudo yum update -y
sudo yum install -y httpd
```

2. Configurez de manière à démarrer Apache au redémarrage :

```
sudo systemctl enable --now httpd
```

3. Vérifiez que la version de httpd que vous avez installée inclut `proxy_hcheck_module`:

```
sudo httpd -M
```

Vous devez utiliser `proxy_hcheck_module`. Si votre version de httpd n'inclut pas ce module, mettez à jour à une version de httpd qui l'inclut.

## Configurer le serveur proxy pour tester la connectivité à Tableau Server

Exécutez cette procédure sur l'un des hôtes proxy (Proxy 1). Le but de cette étape est de vérifier la connectivité entre Internet et votre serveur proxy et Tableau Server dans le groupe de sécurité privé.

1. Créez un fichier appelé `tableau.conf` et ajoutez-le au répertoire `/etc/httpd/conf.d`.

Copiez le code suivant et spécifiez les clés `ProxyPass` et `ProxyPassReverse` avec l'adresse IP privée du Nœud 1 de Tableau Server.

**Important** : la configuration présentée ci-dessous n'est pas sécurisée et ne doit pas être utilisée en production. Cette configuration ne doit être utilisée que pendant le processus d'installation pour vérifier la connectivité de bout en bout.

Par exemple, si l'adresse IP privée du nœud 1 est `10.0.30.32`, le contenu du fichier `tableau.conf` serait :

```
<VirtualHost *:80>
ProxyPreserveHost On
```

```
ProxyPass "/" "http://10.0.30.32:80/"
ProxyPassReverse "/" "http://10.0.30.32:80/"
</VirtualHost>
```

2. Redémarrez httpd :

```
sudo systemctl restart httpd
```

## Vérification : configuration de la topologie de base

Vous devriez pouvoir accéder à la page d'administration de Tableau Server en accédant à `http://<proxy-public-IP-address>`.

Si la page de connexion à Tableau Server ne se charge pas dans votre navigateur, suivez ces étapes de dépannage sur l'hôte Proxy 1 :

- Arrêtez puis démarrez httpd comme première étape de dépannage.
- Revérifiez le fichier `tableau.conf`. Vérifiez que l'adresse IP privée de Nœud 1 est correcte. Vérifiez les guillemets doubles et examinez attentivement la syntaxe.
- Exécutez la commande `curl` sur le serveur proxy inverse avec l'adresse IP privée de Nœud 1, par exemple `curl 10.0.1.90`. Si l'interpréteur de commandes ne renvoie pas html ou s'il renvoie html pour la page Web de test Apache, vérifiez la configuration du protocole/port entre les groupes de sécurité Public et Privé.
- Exécutez la commande `curl` avec l'adresse IP privée de Proxy 1, par exemple `curl 10.0.0.163`. Si l'interpréteur de commandes renvoie le code html de la page Web de test Apache, le fichier proxy n'est pas configuré correctement.
- Redémarrez toujours httpd (`sudo systemctl restart httpd`) après toute modification de configuration du fichier proxy ou des groupes de sécurité.
- Assurez-vous que TSM s'exécute sur le Nœud 1.

## Configurer l'équilibrage de charge sur le proxy

1. Sur le même hôte proxy (Proxy 1) où vous avez créé le fichier `tableau.conf`, supprimez la configuration d'hôte virtuel existante et modifiez le fichier pour inclure la logique d'équilibrage de charge.

Par exemple :

```
<VirtualHost *:80>
ServerAdmin admin@example.com
#Load balancing logic.
ProxyHCEExpr ok234 {%{REQUEST_STATUS} =~ /^[234]/}
Header add Set-Cookie "ROUTEID=.{%{BALANCER_WORKER_ROUTE}e;
path=/" env=BALANCER_ROUTE_CHANGED
<Proxy balancer://tableau>
#Replace IP addresses below with the IP addresses to the
Tableau Servers running the Gateway service.
BalancerMember http://10.0.3.40/ route=1 hcmethod=GET hcex-
pr=ok234 hcuri=/favicon.ico
BalancerMember http://10.0.4.151/ route=2 hcmethod=GET hcex-
pr=ok234 hcuri=/favicon.ico
ProxySet stickysession=ROUTEID
</Proxy>
ProxyPreserveHost On
ProxyPass / balancer://tableau/
ProxyPassReverse / balancer://tableau/
</VirtualHost>
```

2. Arrêtez puis démarrez httpd :

```
sudo systemctl stop httpd
sudo systemctl start httpd
```

3. Vérifiez la configuration en accédant à l'adresse IP publique du proxy 1.

## Copier la configuration sur le deuxième serveur proxy

1. Copiez le fichier `tableau.conf` depuis le proxy 1 et enregistrez-le dans le répertoire `/etc/httpd/conf.d` sur l'hôte proxy 2.
2. Arrêtez puis démarrez httpd :

```
sudo systemctl stop httpd
sudo systemctl start httpd
```

3. Vérifiez la configuration en accédant à l'adresse IP publique du proxy 2.

## Configurer l'équilibreur de charge d'application AWS

Configurez l'équilibreur de charge en tant qu'écouteur HTTP. La procédure ici décrit comment ajouter un équilibreur de charge dans AWS.

### Étape 1 : Créer un groupe cible

Un groupe cible est une configuration AWS qui définit les instances EC2 exécutant vos serveurs proxy. Ce sont les cibles pour le trafic provenant du LBS.

1. EC2>**Groupes cibles** > **Créer un groupe cible**
2. Sur la page Créer :
  - Entrez un nom de groupe cible, par exemple `TG-internal-HTTP`
  - Type de cible : Instances
  - Protocole : HTTP
  - Port : 80
  - VPC : Sélectionnez votre VPC
  - Sous **Contrôles d'intégrité** > **Paramètres avancés des contrôles d'intégrité** > **Codes de réussite**, ajoutez la liste des codes pour lire : 200, 303.
  - Cliquez sur **Créer**
3. Sélectionnez le groupe cible que vous venez de créer, puis cliquez sur l'onglet **Cibles** :
  - Cliquez sur **Modifier**.
  - Sélectionnez les instances EC2 (ou une seule instance si vous en configurez une à la fois) qui exécutent l'application proxy, puis cliquez sur **Ajouter à**

**l'enregistrement.**

- Cliquez sur **Enregistrer**.

## Étape 2 : Lancer l'assistant d'équilibrage de charge

1. EC2 > **Équilibreurs de charge** > **Créer un équilibreur de charge**
2. Sur la page « Sélectionner le type d'équilibreur de charge », créez un équilibreur de charge d'application.

**Remarque** : l'interface utilisateur qui s'affiche pour configurer l'équilibreur de charge peut présenter des différences selon les centres de données AWS. La procédure ci-dessous, « Configuration de l'assistant », correspond à l'assistant de configuration AWS qui commence par l'**Étape 1 Configurer l'équilibreur de charge**.

Si votre centre de données affiche toutes les configurations sur une seule page qui inclut un bouton **Créer un équilibreur de charge** en bas de la page, suivez la procédure « Configuration d'une seule page » ci-dessous.

## Configuration de l'assistant

1. Page **Configurer l'équilibreur de charge** :
  - Précisez le nom
  - Schéma : face à Internet (par défaut)
  - Type d'adresse IP : ipv4 (par défaut)
  - Écouteurs (écouteurs et routage) :
    - a. Laissez l'écouteur HTTP par défaut
    - b. Cliquez sur **Ajouter un écouteur** et ajoutez `HTTPS : 443`

- VPC : sélectionnez le VPC où vous avez tout installé
- Zones de disponibilité :
  - Sélectionnez **a** et **b** comme vos régions de centre de données
  - Dans chaque liste de sélection déroulante correspondante, sélectionnez le sous-réseau Public (où résident vos serveurs proxy).
- Cliquez sur : **Configurer les paramètres de sécurité**

2. Page **Configurer les paramètres de sécurité**

- Téléchargez votre certificat SSL public.
- Cliquez sur **Suivant : Configurer des groupes de sécurité**.

3. Page **Configurer les groupes de sécurité** :

- Sélectionnez le groupe de sécurité Public. Si le groupe de sécurité par défaut est sélectionné, effacez cette sélection.
- Cliquez sur **Suivant : Configurer le routage**.

4. Page **Configurer le routage**

- Groupe cible : Groupe cible existant.
- Nom : sélectionnez le groupe cible que vous avez créé précédemment
- Cliquez sur **Suivant : Enregistrer les cibles**.

5. Page **Enregistrer les cibles**

- Les deux instances de serveur proxy que vous avez configurées précédemment doivent s'afficher.
- Cliquez sur **Suivant : Vérifier**.

6. Page **Révision**

Cliquez sur **Créer**.

## Configuration d'une seule page

### Configuration de base

- Précisez le nom
- Schéma : face à Internet (par défaut)
- Type d'adresse IP : ipv4 (par défaut)

### Mappages réseau

- VPC : sélectionnez le VPC où vous avez tout installé
- Mappages :
  - Sélectionnez les zones de disponibilité **a** et **b** (ou comparables) comme vos régions de centres de données
  - Dans chaque liste de sélection déroulante correspondante, sélectionnez le sous-réseau Public (où résident vos serveurs proxy).

### Groupes de sécurité

Sélectionnez le groupe de sécurité Public. Si le groupe de sécurité par défaut est sélectionné, effacez cette sélection.

### Écouteurs et routage

- Laissez l'écouteur HTTP par défaut. Dans **Action par défaut**, spécifiez le groupe cible que vous avez précédemment configuré.
- Cliquez sur **Ajouter un écouteur** et ajoutez **HTTPS : 443**. Dans **Action par défaut**, spécifiez le groupe cible que vous avez précédemment configuré.

### Paramètres d'écoute sécurisés

- Téléchargez votre certificat SSL public.

Cliquez sur **Créer un équilibreur de charge**.

## Étape 3 : Activer la persistance

1. Une fois l'équilibreur de charge créé, vous devez activer la persistance sur le groupe cible.
  - Ouvrez la page Groupe cible AWS (**EC2 > Équilibreurs de charge > Groupes cibles**), sélectionnez l'instance d'équilibreur de charge cible que vous venez de configurer. Dans le menu **Actions**, sélectionnez **Modifier les attributs**.



- Sur la page **Modifier les attributs**, sélectionnez **Persistance**, spécifiez une durée 1 day, puis **Enregistrer les modifications**.
2. Dans l'équilibreur de charge, activez la persistance sur l'écouteur HTTP. Sélectionnez l'équilibreur de charge que vous venez de configurer, puis cliquez sur l'onglet **Écou-teurs**.
- Pour **HTTP:80**, cliquez sur **Afficher/modifier les règles**. Dans la page **Règles** résultante, cliquez sur l'icône de modification (une fois en haut de la page, puis à nouveau près de la règle) pour modifier la règle. Supprimez la règle THEN existante et remplacez-la en cliquant sur **Ajouter une action > Transférer vers....** Dans la configuration THEN résultante, spécifiez le même groupe cible que vous avez créé. Sous Persistance au niveau du groupe, activez la persistance et définissez la durée sur 1 jour. Enregistrez le paramètre puis cliquez sur **Mettre à jour**.

## Étape 4 : Définir le délai d'inactivité sur l'équilibreur de charge

Sur l'équilibreur de charge, mettez à jour le délai d'inactivité sur 400 secondes.

Sélectionnez l'équilibreur de charge que vous avez configuré pour ce déploiement, puis cliquez sur **Actions > Modifier les attributs**. Définissez le **Délai d'inactivité** sur 400 secondes, puis cliquez sur **Enregistrer**.

## Étape 5 : Vérifier la connectivité LBS

Ouvrez la page de l'équilibreur de charge AWS (**EC2 > Équilibreurs de charge**), puis sélectionnez l'instance d'équilibreur de charge que vous venez de configurer.

Sous **Description**, copiez le nom DNS et collez-le dans un navigateur pour accéder à la page de connexion Tableau Server.

Si vous obtenez une erreur de niveau 500, vous devrez probablement redémarrer vos serveurs proxy.

## Mettre à jour le DNS avec l'URL publique de Tableau

Utilisez le nom de zone DNS de votre domaine dans la description de l'équilibreur de charge AWS pour créer une valeur CNAME dans votre DNS. Le trafic vers votre URL (tableau.example.com) doit être envoyé au nom DNS public AWS.

## Vérifier la connectivité

Une fois vos mises à jour DNS terminées, vous devriez pouvoir accéder à la page de connexion Tableau Server en saisissant votre URL publique, par exemple `https://-tableau.example.com`.

## Exemple de configuration d'authentification : SAML avec IdP externe

L'exemple suivant décrit comment installer et configurer SAML avec un IdP Okta et un module d'authentification Mellon pour un déploiement Tableau exécuté dans l'architecture de référence AWS. L'exemple décrit comment configurer Tableau Server et les serveurs proxy Apache pour utiliser HTTP. Okta enverra la demande à l'équilibreur de charge AWS via HTTPS, mais tout le trafic interne transitera via HTTP. Lors de la configuration de ce scénario, tenez compte des protocoles HTTP et HTTPS lors de la définition des chaînes d'URL.

Cet exemple utilise Mellon comme module de fournisseur de services de pré-authentification sur les serveurs proxy inverses. Cette configuration garantit que seul le trafic authentifié se connecte à Tableau Server, qui fait également office de fournisseur de services avec l'IdP Okta. Par conséquent, vous devez configurer deux applications d'IdP : une pour le fournisseur de services Mellon et une pour le fournisseur de services Tableau.

## Créer un compte d'administrateur Tableau

Une erreur courante lors de la configuration de SAML est d'oublier de créer un compte administrateur sur Tableau Server avant d'activer l'authentification unique.

La première étape consiste à créer un compte sur Tableau Server avec un rôle d'administrateur de serveur. Dans l'exemple de scénario Okta, le nom d'utilisateur doit utiliser le format d'une adresse e-mail valide, par exemple user@example.com. Vous devez définir un mot de passe pour cet utilisateur, mais le mot de passe ne sera pas utilisé une fois SAML configuré.

## Configurer l'application de pré-authentification Okta

Le scénario de bout en bout décrit dans cette section nécessite deux applications Okta :

- Demande de pré-autorisation Okta
- Application Okta Tableau Server

Chacune de ces applications est associée à différentes métadonnées que vous devrez configurer sur le serveur proxy inverse et Tableau Server, respectivement.

Cette procédure décrit comment créer et configurer l'application de pré-authentification Okta. Plus loin dans cette rubrique, vous créerez l'application Okta Tableau Server. Pour un test gratuit de compte Okta avec un nombre limité d'utilisateurs, consultez la [page Web des développeurs Okta](#).

Créez une intégration d'application SAML pour le fournisseur de services de pré-authentification Mellon.

1. Ouvrez le tableau de bord d'administration d'Okta > **Applications** > **Créer une intégration d'application**.
2. Dans la page **Créer une nouvelle intégration d'application**, sélectionnez **SAML 2.0**, puis cliquez sur **Suivant**.

3. Dans l'onglet **Paramètres généraux**, saisissez un nom d'application, par exemple `Tableau Pre-Auth`, puis cliquez sur **Suivant**.
4. Dans l'onglet **Configurer SAML** :
  - URL d'authentification unique (SSO). Le dernier élément du chemin dans l'URL d'authentification unique est appelé `MellonEndpointPath` dans le fichier de configuration `mellon.conf` présenté plus loin dans cette procédure. Vous pouvez spécifier le point de terminaison de votre choix. Dans cet exemple, `sso` est le point de terminaison. Le dernier élément, `postResponse`, est requis :  
`https://tableau.example.com/sso/postResponse`.
  - Décochez la case : **Use this for Recipient URL and Destination URL** (À utiliser comme URL du destinataire et URL de destination).
  - URL du destinataire : identique à l'URL SSO, mais avec HTTP. Par exemple, `http://tableau.example.com/sso/postResponse`.
  - URL de destination : identique à l'URL du destinataire, mais avec HTTP. Par exemple, `http://tableau.example.com/sso/postResponse`.
  - URI d'audience (ID d'entité SP). Par exemple, `https://-tableau.example.com`.
  - Format d'identification du nom : `EmailAddress`
  - Nom d'utilisateur de l'application : `Email`
  - Déclarations d'attributs : Nom = `mail` ; Format du nom = `Unspecified` ; Valeur = `user.email`.

Cliquez sur **Suivant**.

5. Dans l'onglet **Commentaires**, sélectionnez :
  - **Je suis un client Okta ajoutant une application interne**
  - **Il s'agit d'une application interne que nous avons créée**
  - Cliquez sur **Terminer**.
6. Créez le fichier de métadonnées IdP de pré-authentification :
  - Dans Okta : **Applications > Applications > Votre nouvelle application** (par exemple `Tableau Pre-Auth`) > **Connexion**
  - À côté de **Certificats de signature SAML**, cliquez sur **Afficher les instructions de configuration SAML**.

- Dans la page **Comment configurer SAML 2.0 pour<pre-auth> Application**, faites défiler jusqu'à la section **Facultatif, Fournir les métadonnées d'IDP suivantes à votre fournisseur de SP**.
- Copiez le contenu du champ XML et enregistrez-le dans un fichier appelé `pre-auth_idp_metadata.xml`.

7. (Facultatif) Configurez l'authentification multifacteur :

- Dans Okta : **Applications > Applications** > Votre nouvelle application (par exemple `Tableau Pre-Auth`) > **Connexion**
- Sous **Stratégie de connexion**, cliquez sur **Ajouter une règle**.
- Dans **Règle de connexion aux applications**, spécifiez un nom et les différentes options MFA. Pour tester la fonctionnalité, vous pouvez laisser toutes les options par défaut. Par contre, sous **Actions**, vous devez sélectionner **Demander le facteur**, puis spécifier la fréquence à laquelle les utilisateurs doivent se connecter. Cliquez sur **Enregistrer**.

## Créer et affecter un utilisateur Okta

1. Dans Okta, créez un utilisateur avec le même nom d'utilisateur que vous avez créé dans Tableau (`user@example.com`) : **Répertoire > Personnes > Ajouter une personne**.
2. Une fois l'utilisateur créé, attribuez la nouvelle application Okta à cette personne : cliquez sur le nom d'utilisateur, puis attribuez l'application dans **Attribuer une application**.

## Installer Mellon pour la pré-authentification

1. Sur les instances EC2 qui exécutent le serveur proxy Apache, exécutez les commandes suivantes pour installer PHP et les modules Mellon :

```
sudo yum install httpd php mod_auth_mellon
```

2. Créez le répertoire `/etc/httpd/mellon`

# Configurer Mellon comme module de pré-authentification

Exécutez cette procédure sur les deux serveurs proxy.

Vous devez avoir une copie du fichier `pre-auth_idp_metadata.xml` que vous avez créé à partir de la configuration Okta.

1. Changez de répertoire :

```
cd /etc/httpd/mellon
```

2. Créez les métadonnées du fournisseur de services. Exécutez le script `mellon_create_metadata.sh`. Vous devez inclure l'ID d'entité et l'URL de retour de votre entreprise dans la commande.

L'URL de retour est appelée URL d'*authentification unique* dans Okta. Le dernier élément du chemin dans l'URL de retour est appelé `MellonEndpointPath` dans le fichier de configuration `mellon.conf` présenté plus loin dans cette procédure. Dans cet exemple, nous spécifions `sso` comme chemin de point de terminaison.

Par exemple :

```
sudo /usr/libexec/mod_auth_mellon/mellon_create_metadata.sh
https://tableau.example.com "https://tableau.example.com/sso"
```

Le script renvoie le certificat du fournisseur de services, la clé et les fichiers de métadonnées.

3. Renommez les fichiers du fournisseur de services dans le répertoire `mellon` pour une meilleure lisibilité. Nous désignerons ces fichiers par les noms suivants dans la documentation :

```
sudo mv *.key mellon.key
sudo mv *.cert mellon.cert
```

## Guide de déploiement de Tableau Server en entreprise

```
sudo mv *.xml sp_metadata.xml
```

4. Copiez le fichier `pre-auth_idp_metadata.xml` sur le même répertoire.
5. Créez le fichier `mellon.conf` dans le répertoire `/etc/httpd/conf.d`:

```
sudo nano /etc/httpd/conf.d/mellon.conf
```

6. Copiez le contenu suivant dans `mellon.conf`.

```
<Location />
MellonSPPrivateKeyFile /etc/httpd/mellon/mellon.key
MellonSPCertFile /etc/httpd/mellon/mellon.cert
MellonSPMetadataFile /etc/httpd/mellon/sp_metadata.xml
MellonIdPMetadataFile /etc/httpd/mellon/pre-auth_idp_metadata.xml
MellonEndpointPath /sso
MellonEnable "info"
</Location>
```

7. Ajoutez le contenu suivant dans le fichier `tableau.conf` existant :

À l'intérieur du bloc `<VirtualHost *:80>`, ajoutez le contenu suivant. Mettez à jour `ServerName` avec le nom d'hôte public dans votre ID d'entité :

```
DocumentRoot /var/www/html
ServerName tableau.example.com
ServerSignature Off
ErrorLog logs/error_sp.log
CustomLog logs/access_sp.log combined
LogLevel info
```

Ajoutez le bloc `Emplacement` en dehors du bloc `<VirtualHost *:80>`. Mettez à jour `MellonCookieDomain` avec le domaine de premier niveau pour conserver les informations sur les cookies, comme indiqué :

```
<Location />
AuthType Mellon
MellonEnable auth
Require valid-user
MellonCookieDomain example.com
</Location>
```

**Le fichier `tableau.conf` complet peut se présenter comme suit :**

```
<VirtualHost *:80>
ServerAdmin admin@example.com
ProxyHCEExpr ok234 {%{REQUEST_STATUS} =~ /^[234]/}
Header add Set-Cookie "ROUTEID=.%{BALANCER_WORKER_ROUTE}e;
path=/" env=BALANCER_ROUTE_CHANGED
<Proxy balancer://tableau>
BalancerMember http://10.0.3.36/ route=1 hcmethod=GET hcex-
pr=ok234 hcuri=/favicon.ico
BalancerMember http://10.0.4.15/ route=2 hcmethod=GET hcex-
pr=ok234 hcuri=/favicon.ico
ProxySet stickysession=ROUTEID
</Proxy>
ProxyPreserveHost On
ProxyPass / balancer://tableau/
ProxyPassReverse / balancer://tableau/
DocumentRoot /var/www/html
ServerName tableau.example.com
ServerSignature Off
ErrorLog logs/error_sp.log
CustomLog logs/access_sp.log combined
LogLevel info
</VirtualHost>
<Location />
AuthType Mellon
MellonEnable auth
Require valid-user
```



```
MellonCookieDomain example.com
</Location>
```

8. Vérifiez la configuration. Exécutez la commande suivante :

```
sudo apachectl configtest
```

Si le test de configuration renvoie une erreur, corrigez les erreurs et exécutez à nouveau configtest. Une configuration réussie retournera `Syntax OK`.

9. Redémarrez httpd :

```
sudo systemctl restart httpd
```

## Créer une application Tableau Server dans Okta

1. Dans le tableau de bord Okta : **Applications > Applications > Parcourir le catalogue d'applications**
2. Dans **Parcourir le catalogue d'intégration d'applications**, recherchez `Tableau`, sélectionnez la section `Tableau Server`, puis cliquez sur **Ajouter**.
3. Sur **Ajouter Tableau Server > Paramètres généraux**, saisissez une étiquette, puis cliquez sur **Suivant**.
4. Dans Options de connexion, sélectionnez **SAML 2.0** et faites défiler jusqu'à Paramètres de connexion avancés :
  - **ID d'entité SAML** : saisissez l'URL publique, par exemple `https://-tableau.example.com`.
  - **Format du nom d'utilisateur de l'application** : E-mail
5. Cliquez sur le lien **Métadonnées du fournisseur d'identité** pour lancer un navigateur. Copiez le lien du navigateur. C'est le lien que vous utiliserez lorsque vous configurerez Tableau dans la procédure qui suit.
6. Cliquez sur **Terminé**.
7. Attribuez la nouvelle application Tableau Server Okta à votre utilisateur (`user@example.com`) : cliquez sur le nom d'utilisateur, puis attribuez l'application dans **Attribuer une application**.

## Activer SAML sur Tableau Server pour IdP

Exécutez cette procédure sur le Nœud 1 de Tableau Server.

1. Téléchargez les métadonnées de l'application Tableau depuis Okta. Utilisez le lien que vous avez enregistré de la procédure précédente :

```
wget https://dev-66144217.ok-ta.com/app/exklegxgt1fhjkSeS5d7/sso/saml/metadata -O idp_metadata.xml
```

2. Copiez un certificat TLS et le fichier de clé associé sur Tableau Server. Le fichier de clé doit être une clé RSA. Pour plus d'informations sur la certificat et les exigences de l'IdP, consultez *Exigences en matière d'authentification SAML* ([Linux](#)).

Pour simplifier la gestion et le déploiement des certificats, et comme meilleure pratique de sécurité, nous vous recommandons d'utiliser des certificats générés par une autorité de certification (CA) tierce majeure de confiance. Sinon, vous pouvez générer des certificats auto-signés ou utiliser les certificats d'un PKI pour TLS.

Si vous n'avez pas de certificat TLS, vous pouvez générer un certificat auto-signé en appliquant la procédure intégrée ci-dessous.

### Générer un certificat auto-signé

Exécutez cette procédure sur le Nœud 1 de Tableau Server.

- a. Générez la clé de l'autorité de certification racine (CA) de signature :

```
openssl genrsa -out rootCAKey-saml.pem 2048
```

- b. Créez le certificat CA racine :

```
openssl req -x509 -sha256 -new -nodes -key rootCAKey-saml.-
pem -days 3650 -out rootCACert-saml.pem
```

Vous serez invité à saisir des valeurs pour les champs du certificat. Par exemple :

```
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:Washington
Locality Name (eg, city) [Default City]:Seattle
Organization Name (eg, company) [Default Company Ltd]:Ta-
bleau
Organizational Unit Name (eg, section) []:Operations
Common Name (eg, your name or your server's hostname) []:ta-
bleau.example.com
Email Address []:example@tableau.com
```

- c. Créez le certificat et la clé associée (`server-saml.csr` et `server-saml.key` dans l'exemple ci-dessous). Le nom du sujet du certificat doit correspondre au nom de l'hôte public de l'hôte Tableau. Le nom du sujet est défini à l'aide de l'option `-subj` avec le format `"/CN=<host-name>"`, par exemple :

```
openssl req -new -nodes -text -out server-saml.csr -keyout
server-saml.key -subj "/CN=tableau.example.com"
```

- d. Signez le nouveau certificat avec le certificat CA que vous avez créé ci-dessus. La commande suivante génère également le certificat au format `crt` :

```
openssl x509 -req -in server-saml.csr -days 3650 -CA
rootCACert-saml.pem -CAkey rootCAKey-saml.pem -CAcrea-
teserial -out server-saml.crt
```

- e. Convertissez le fichier de clé en RSA. Tableau requiert un fichier de clé RSA pour SAML. Pour convertir la clé, exécutez la commande suivante :

```
openssl rsa -in server-saml.key -out server-saml-rsa.key
```

3. Configurez SAML. Exécutez la commande suivante, en spécifiant votre ID d'entité et votre URL de retour, ainsi que les chemins d'accès au fichier de métadonnées, au fichier de certificat et au fichier de clé :

```
tsm authentication saml configure --idp-entity-id "https://-
tableau.example.com" --idp-return-url "https://-
tableau.example.com" --idp-metadata idp_metadata.xml --cert-
file "server-saml.crt" --key-file "server-saml-rsa.key"
```

```
tsm authentication saml enable
```

4. Si votre entreprise exécute Tableau Desktop 2021.4 ou une version ultérieure, vous devez exécuter la commande suivante pour activer l'authentification via les serveurs proxy inverses.

Les versions de Tableau Desktop 2021.2.1 - 2021.3 fonctionneront sans que cette commande soit exécutée, à condition que votre module de pré-authentification (par exemple Mellon) soit configuré pour autoriser la conservation des cookies de domaine de niveau supérieur.

```
tsm configuration set -k features.ExternalBrowserOAuth -v false
```

5. Appliquez les changements de configuration :

```
tsm pending-changes apply
```

## Valider la fonctionnalité SAML

Pour valider la fonctionnalité SAML de bout en bout, connectez-vous à Tableau Server avec l'URL publique (par exemple, <https://tableau.example.com>) en utilisant le compte administrateur Tableau que vous avez créé au début de cette procédure.

## Résolution des problèmes de validation

**Requête incorrecte** : une erreur courante pour ce scénario est une erreur « Bad Request » (Requête incorrecte) d'Okta. Ce problème se produit souvent lorsque le navigateur met en cache les données de la session Okta précédente. Par exemple, si vous gérez les applications Okta en tant qu'administrateur Okta, puis tenez d'accéder à Tableau à l'aide d'un autre compte compatible Okta, les données de session provenant des données de l'administrateur peuvent provoquer l'erreur « Bad Request ». Si cette erreur persiste même après la suppression du cache du navigateur local, essayez de valider le scénario Tableau en vous connectant avec un autre navigateur.

Une autre cause de l'erreur « Bad Request » est une faute de frappe dans l'une des nombreuses URL que vous entrez lors des processus de configuration Okta, Mellon et SAML. Vérifiez soigneusement tous ces points.

Souvent le fichier `httpd.error.log` sur le serveur Apache spécifiera l'URL qui est à l'origine de l'erreur.

**Introuvable - L'URL demandée était introuvable sur ce serveur** : cette erreur indique l'une des nombreuses erreurs de configuration possibles.

Si l'utilisateur est authentifié avec Okta, puis reçoit cette erreur, il est probable que vous ayez téléchargé l'application de pré-authentification Okta sur Tableau Server lorsque vous avez configuré SAML. Vérifiez que vous avez configuré les métadonnées de l'application Okta Tableau Server sur Tableau Server, et non pas les métadonnées de l'application de pré-authentification Okta

Autres étapes de résolution des problèmes :

- Vérifiez soigneusement que `tableau.conf` ne contient pas de fautes de frappe ou d'erreurs de configuration.
- Passez en revue les paramètres de l'application de pré-authentification Okta. Assurez-vous que les protocoles HTTP vs HTTPS sont définis comme spécifié dans cette rubrique.
- Redémarrez `httpd` sur les deux serveurs proxy.

- Vérifiez que `sudo apachectl configtest` renvoie « Syntax OK » sur les deux serveurs proxy.
- Vérifiez que l'utilisateur test est affecté aux deux applications dans Okta.
- Vérifiez que l'adhérence est activée sur l'équilibreur de charge et les groupes cibles associés

## Configurer SSL/TLS depuis l'équilibreur de charge vers Tableau Server

Certaines entreprises exigent un canal de cryptage de bout en bout du client au service principal. L'architecture de référence par défaut telle que décrite jusqu'à présent spécifie SSL du client à l'équilibreur de charge s'exécutant dans le niveau Web de votre entreprise.

Pour configurer SSL depuis l'équilibreur de charge vers Tableau Server, vous devez :

- Installer un certificat SSL valide sur les serveurs Tableau et proxy.
- Configurer SSL depuis l'équilibreur de charge vers les serveurs proxy inverses.
- Configurer SSL depuis les serveurs proxy vers Tableau Server.
- Vous pouvez également configurer SSL de Tableau Server vers l'instance PostgreSQL.

Le reste de cette rubrique décrit cette implémentation dans le contexte de l'exemple d'architecture de référence AWS.

## Exemple : Configurer SSL/TLS dans l'architecture de référence AWS

Cette section décrit comment configurer SSL sur Tableau et configurer SSL sur un serveur proxy Apache, le tout s'exécutant dans l'exemple d'architecture AWS de référence.

Les procédures Linux décrites tout au long de cet exemple montrent des commandes pour les distributions de type RHEL. Plus précisément, les commandes présentées ici ont été déve-

luppées avec la distribution Amazon Linux 2. Si vous exécutez la distribution Ubuntu, modifiez les commandes en conséquence.

## Étape 1 : Rassembler les certificats et les clés associées

Pour simplifier la gestion et le déploiement des certificats, et comme meilleure pratique de sécurité, nous vous recommandons d'utiliser des certificats générés par une autorité de certification (CA) tierce majeure de confiance.

Sinon, vous pouvez générer des certificats auto-signés ou utiliser les certificats d'un PKI pour TLS.

La procédure suivante explique comment générer des certificats auto-signés. Si vous utilisez des certificats tiers comme nous le recommandons, vous pouvez ignorer cette procédure.

Exécutez cette procédure sur l'un des hôtes proxy. Après avoir généré le certificat et la clé associée, vous les partagerez avec l'autre hôte proxy et avec le Nœud 1 de Tableau Server.

1. Générez la clé de l'autorité de certification racine (CA) de signature :

```
openssl genrsa -out rootCAKey.pem 2048
```

2. Créez le certificat CA racine :

```
openssl req -x509 -sha256 -new -nodes -key rootCAKey.pem -days
3650 -out rootCACert.pem
```

Vous serez invité à saisir des valeurs pour les champs du certificat. Par exemple :

```
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:Washington
Locality Name (eg, city) [Default City]:Seattle
Organization Name (eg, company) [Default Company Ltd]:Tableau
Organizational Unit Name (eg, section) []:Operations
Common Name (eg, your name or your server's hostname) []:ta-
```

```
bleau.example.com
```

```
Email Address []:example@tableau.com
```

3. Créez le certificat et la clé associée (`serverssl.csr` et `serverssl.key` dans l'exemple ci-dessous). Le nom du sujet du certificat doit correspondre au nom de l'hôte public de l'hôte Tableau. Le nom du sujet est défini à l'aide de l'option `-subj` avec le format `"/CN=<host-name>"`, par exemple :

```
openssl req -new -nodes -text -out serverssl.csr -keyout serverssl.key -subj "/CN=tableau.example.com"
```

4. Signez le nouveau certificat avec le certificat CA que vous avez créé à l'étape 2. La commande suivante génère également le certificat au format `crt` :

```
openssl x509 -req -in serverssl.csr -days 3650 -CA rootCACert.pem -CAkey rootCAKey.pem -CAcreateserial -out serverssl.crt
```

## Étape 2 : Configurer le serveur proxy pour SSL

Exécutez cette procédure sur les deux serveurs proxy.

1. Installez le module Apache ssl :

```
sudo yum install mod_ssl
```

2. Créez le répertoire `/etc/ssl/private` :

```
sudo mkdir -p /etc/ssl/private
```

3. Copiez les fichiers `crt` et `key` dans les chemins d'accès `/etc/ssl/` suivants :

```
sudo cp serverssl.crt /etc/ssl/certs/
```

```
sudo cp serverssl.key /etc/ssl/private/
```

4. Mettez à jour le fichier `tableau.conf` existant avec les mises à jour suivantes :



- Ajoutez le bloc de réécriture SSL :

```
RewriteEngine on
RewriteCond %{SERVER_NAME} =tableau.example.com
RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI}
[END,NE,R=permanent]
```

- Dans le bloc de réécriture SSL, mettez à jour le nom du serveur RewriteCond : ajoutez votre nom d'hôte public, par exemple tableau.example.com.
- Modifiez <VirtualHost \*:80> en <VirtualHost \*:443>.
- Enveloppez les blocs <VirtualHost \*:443> et <Location /> avec <IfModule mod\_ssl.c>...</IfModule>.
- BalancerMember : modifiez le protocole de http en https.
- Ajoutez les éléments SSL\* à l'intérieur du bloc <VirtualHost \*:443> :

```
SSLEngine on
SSLCertificateFile /etc/ssl/certs/serverssl.crt
SSLCertificateKeyFile /etc/ssl/private/serverssl.key
SSLProxyEngine on
SSLProxyVerify none
SSLProxyCheckPeerName off
SSLProxyCheckPeerExpire off
```

- Dans l'élément LogLevel : ajoutez ssl:warn.
- Facultatif : si vous avez installé et configuré un module d'authentification, vous pouvez avoir des éléments supplémentaires dans le fichier tableau.conf. Par exemple, le bloc <Location /> </Location> inclura des éléments.

Un exemple de fichier tableau.conf configuré pour SSL est présenté ici :

```
RewriteEngine on
RewriteCond %{SERVER_NAME} =tableau.example.com
RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R-
R=permanent]

<IfModule mod_ssl.c>
<VirtualHost *:443>
ServerAdmin admin@example.com
```

```

ProxyHCEExpr ok234 {%{REQUEST_STATUS} =~ /^[234]/}
Header add Set-Cookie "ROUTEID=.%{BALANCER_WORKER_ROUTE}e;
path=/" env=BALANCER_ROUTE_CHANGED
<Proxy balancer://tableau>
BalancerMember https://10.0.3.36/ route=1 hcmethod=GET hcex-
pr=ok234 hcuri=/favicon.ico
BalancerMember https://10.0.4.15/ route=2 hcmethod=GET hcex-
pr=ok234 hcuri=/favicon.ico
ProxySet stickysession=ROUTEID
</Proxy>
ProxyPreserveHost On
ProxyPass / balancer://tableau/
ProxyPassReverse / balancer://tableau/
DocumentRoot /var/www/html
ServerName tableau.example.com
ServerSignature Off
ErrorLog logs/error_sp.log
CustomLog logs/access_sp.log combined
LogLevel info ssl:warn
SSLEngine on
SSLCertificateFile /etc/ssl/certs/serverssl.crt
SSLCertificateKeyFile /etc/ssl/private/serverssl.key
SSLProxyEngine on
SSLProxyVerify none
SSLProxyCheckPeerName off
SSLProxyCheckPeerExpire off
</VirtualHost>
<Location />
#If you have configured a pre-auth module (e.g. Mellon) include
those elements here.
</Location>
</IfModule>

```

5. Ajoutez le fichier index.html pour supprimer les erreurs 403 :

```
sudo touch /var/www/html/index.html
```

6. Redémarrez httpd :

```
sudo systemctl restart httpd
```

## Étape 3 : Configurer Tableau Server pour SSL externe

Copiez les fichiers `serverssl.crt` et `serverssl.key` de l'hôte Proxy 1 vers l'instance Tableau Server initiale (Nœud 1).

Exécutez les commandes suivantes sur le Nœud 1 :

```
tsm security external-ssl enable --cert-file serverssl.crt --key-file serverssl.key
tsm pending-changes apply
```

## Étape 4 : Configuration facultative de l'authentification

Si vous avez configuré un fournisseur d'identité externe pour Tableau, vous devrez probablement mettre à jour les URL de retour dans le tableau de bord administratif de l'IdP.

Par exemple, si vous utilisez une application de pré-authentification Okta, vous devrez mettre à jour l'application de l'IdP Mellon de manière à utiliser le protocole HTTPS pour l'URL du destinataire et l'URL de destination.

## Étape 5 : Configurer l'équilibreur de charge AWS pour HTTPS

Si vous effectuez un déploiement avec l'équilibreur de charge AWS comme documenté dans ce guide, vous allez reconfigurer l'équilibreur de charge AWS de manière à envoyer le trafic HTTPS aux serveurs proxy :

1. Annulez l'enregistrement du groupe cible HTTP existant :

Dans **Groupes cibles**, sélectionnez le groupe cible HTTP qui a été configuré pour l'équilibreur de charge, cliquez sur **Actions**, puis sur **Enregistrer et annuler l'enregistrement de l'instance**.

Sur la page **Enregistrer et annuler l'enregistrement des cibles**, sélectionnez les instances actuellement configurées, cliquez sur **Annuler l'enregistrement**, puis sur **Enregistrer**.

2. Créez un groupe HTTPS cible :

**Groupes cibles > Créer un groupe cible**

- Sélectionnez « Instances »
- Entrez un nom de groupe cible, par exemple `TG-internal-HTTPS`
- Sélectionnez votre VPC
- Protocole : HTTPS 443
- Sous **Contrôles d'intégrité > Paramètres avancés des contrôles d'intégrité > Codes de réussite**, ajoutez la liste des codes pour lire : 200, 303.
- Cliquez sur **Créer**.

3. Sélectionnez le groupe cible que vous venez de créer, puis cliquez sur l'onglet **Cibles** :

- Cliquez sur **Modifier**.
- Sélectionnez les instances EC2 qui exécutent l'application proxy, puis cliquez sur **Ajouter aux instances enregistrées**.
- Cliquez sur **Enregistrer**.

4. Une fois le groupe cible créé, vous devez activer la permanence :

- Ouvrez la page Groupe cible AWS (**EC2 > Équilibreurs de charge > Groupes cibles**), sélectionnez l'instance d'équilibreur de charge cible que vous venez de configurer. Dans le menu **Actions**, sélectionnez **Modifier les attributs**.
- Sur la page **Modifier les attributs**, sélectionnez **Persistance**, spécifiez une durée 1 day, puis **Enregistrer les modifications**.

5. Sur l'équilibreur de charge, mettez à jour les règles d'écoute. Sélectionnez l'équilibreur de charge que vous avez configuré pour ce déploiement, puis cliquez sur l'onglet **Écou-teurs**.

- Pour **HTTP:80**, cliquez sur **Afficher/modifier les règles**. Dans la page **Règles** résultante, cliquez sur l'icône de modification (une fois en haut de la page, puis à nouveau près de la règle) pour modifier la règle. Supprimez la règle THEN

existante et remplacez-la en cliquant sur **Ajouter une action > Rediriger vers...** Dans la configuration THEN résultante, spécifiez les ports `HTTPS` et `443` et conservez les paramètres par défaut des autres options. Enregistrez le paramètre puis cliquez sur **Mettre à jour**.

- Pour **HTTP:443**, cliquez sur **Afficher/modifier les règles**. Dans la page **Règles** résultante, cliquez sur l'icône de modification (une fois en haut de la page, puis à nouveau près de la règle) pour modifier la règle. Dans la configuration **THEN**, sous **Transférer vers...**, modifiez le groupe cible par le groupe `HTTPS` que vous venez de créer. Sous **Persistance au niveau du groupe**, activez la persistance et définissez la durée sur 1 jour. Enregistrez le paramètre puis cliquez sur **Mettre à jour**.

## Étape 6 : Vérifier SSL

Vérifiez la configuration en accédant à `https://tableau.example.com`.