Tableau Server Enterprise
Guía de implementación

Última actualización:14/11/2024

Contenido

Guía de implementación de Tableau Server Enterprise

Hemos desarrollado la Guía de implementación de Tableau Server Enterprise (EDG) para proporcionar una guía prescriptiva para implementar Tableau Server (en las instalaciones o en la nube). La Guía ayuda a implementar escenarios empresariales en el contexto de una arquitectura de referencia. Hemos probado la arquitectura de referencia para verificar el cumplimiento de las pruebas comparativas de seguridad, escala y rendimiento, que se ajustan a los procedimientos recomendados estándar del sector.

A grandes rasgos, las características principales de una implementación empresarial estándar del sector consisten en una topología por niveles donde cada capa de funcionalidad de la aplicación del servidor (capa de puerta de enlace web, capa de aplicación y capa de datos) está vinculada y protegida por subredes con control de acceso. Los usuarios que acceden a la aplicación del servidor desde Internet se autentican a nivel web. Una vez autenticada, la solicitud se envía a una subred protegida donde la aplicación maneja la lógica empresarial. Los datos de gran valor están protegidos por la tercera subred: el nivel de datos. Los servicios del nivel de aplicación se comunican mediante la red protegida con el nivel de datos para ofrecer solicitudes de datos a las fuentes de datos del backend.

En esta implementación, la seguridad es la prioridad de todas las decisiones de diseño e implementación. No obstante, la fiabilidad, el rendimiento y la capacidad de adaptación también son requisitos prioritarios. Dado el diseño distribuido y modular de la arquitectura de referencia, la fiabilidad y el rendimiento escalan de una manera linear y predecible, colocando estratégicamente servicios compatibles en cada nodo y agregando servicios en cuellos de botella.

¿Quién debe leer esta guía?

La EDG se ha desarrollado para administradores de TI empresariales que pueden necesitar:

- Una implementación de Tableau administrada por TI
- Aplicación del cumplimiento de la industria
- Prácticas recomendadas de implementación de la industria
- Implementación segura de forma predeterminada

La EDG es una guía de implementación para implementar la arquitectura de referencia empresarial. Si bien esta versión de la EDG incluye una implementación de AWS/Linux de ejemplo, los administradores de TI empresariales experimentados pueden utilizar la guía como un recurso para implementar la arquitectura de referencia prescrita en cualquier entorno de centro de datos estándar de la industria.

Versión

Esta versión de EDG se ha desarrollado para la versión 2021.2.3 y posteriores de Tableau Server. Si bien puede usar el EDG como una referencia general para implementar versiones anteriores de Tableau Server, le recomendamos que implemente la arquitectura de referencia con Tableau Server 2021.2.3 o una versión posterior. Algunas funcionalidades y opciones no están disponibles en versiones anteriores de Tableau Server.

Para obtener las funcionalidades y mejoras más actualizadas, recomendamos implementar EDG con Tableau Server 2022.1.7 y versiones posteriores.

La arquitectura de referencia descrita en esta guía es compatible con los siguientes clientes de Tableau: creación web con navegadores compatibles, Tableau Mobile y Tableau Desktop versión 2021.2.1 o posterior. Otros clientes de Tableau (Tableau Prep, Bridge, etc.) aún no han sido validados con la arquitectura de referencia.

Resaltar funcionalidades

La primera versión de la arquitectura de referencia de Tableau Server presenta los siguientes escenarios y funcionalidades:

- Autorización previa del cliente: los clientes de Tableau (Desktop, Mobile, Creación web) se autentican con el proveedor de autenticación corporativa en el nivel web antes

de acceder al Tableau Server interno. Este proceso se gestiona mediante la configuración de un complemento de autenticación en la puerta de enlace independiente de Tableau Server que actúa como el servidor proxy inverso. Consulte Parte 5: Configuración del nivel web.

- Implementación de confianza cero: debido a que todo el tráfico a Tableau Server está previamente autenticado, toda la implementación de Tableau opera en una subred privada que no requiere una conexión de confianza.
- Repositorio externo: la arquitectura de referencia especifica la instalación del repositorio de Tableau en una base de datos PostgreSQL externa, lo que permite a los administradores de bases de datos administrar, optimizar, escalar y realizar copias de seguridad del repositorio como una base de datos genérica.
- Recuperación inicial del nodo: la EDG introduce un script que automatiza la restauración inicial del nodo en caso de error.
- Copia de seguridad y restauración basados en tar: utilice copias de seguridad .tar conocidos en hitos estratégicos de la implementación de Tableau. En el caso de un error o una configuración incorrecta de la implementación, puede revertirse rápidamente a la etapa de implementación anterior recuperando la copia de seguridad .tar asociada.
- Mejora del rendimiento: la validación del cliente y del laboratorio muestra una mejora del rendimiento del 15-20% cuando se ejecuta EDG en comparación con la implementación estándar.

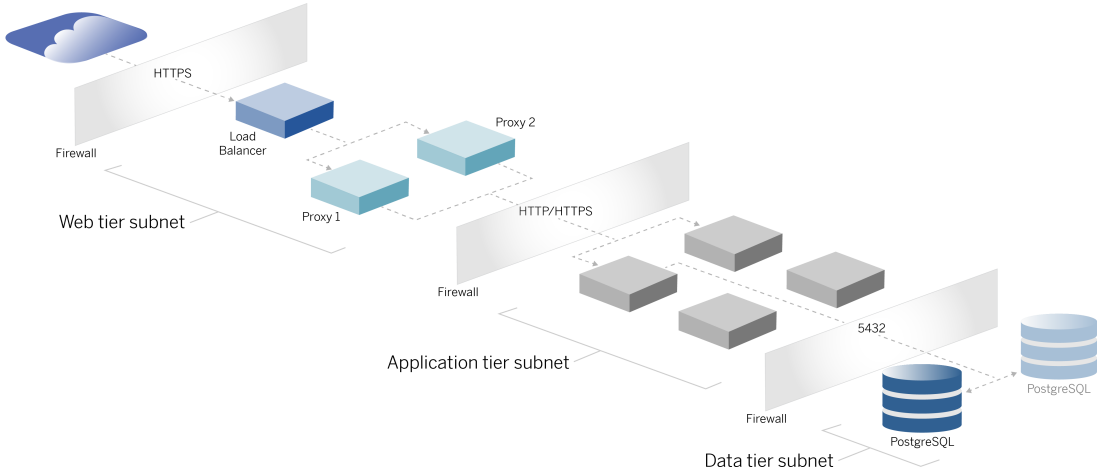
Licencias

La arquitectura de referencia de Tableau Server de esta guía requiere una licencia de Tableau Advanced Management para habilitar el repositorio externo de Tableau Server. Opcionalmente, también puede implementar el almacén de archivos externo de Tableau que también requiere la licencia de Tableau Advanced Management. Consulte *Acerca de Tableau Advanced Management en Tableau Server* (Linux).

Parte 1: comprensión de la implementación empresarial

La parte 1 describe con más detalle las características y requisitos de la implementación empresarial estándar del sector para la que se ha diseñado la Guía de implementación empresarial de Tableau Server.

El siguiente diagrama de red muestra una implementación por niveles de un centro de datos genérico con la arquitectura de referencia de Tableau Server.



Estándares del sector y requisitos de implementación

Las siguientes son funcionalidades de las implementaciones estándar del sector. Estos son los requisitos para los que se ha diseñado la arquitectura de referencia descrita:

- Un diseño de red de varios niveles: la red está unida por subredes protegidas para limitar el acceso en cada capa (capa web, capa de aplicación y capa de datos). Ninguna comunicación individual puede pasar a través de subredes, ya que todas las comunicaciones terminan en la siguiente subred.

- Puertos y protocolos bloqueados de forma predeterminada: cada subred o grupo de seguridad bloquea todos los puertos y protocolos entrantes y salientes de forma predeterminada. La comunicación se habilita, parcialmente, abriendo excepciones en la configuración del puerto o protocolo.
- Autenticación web fuera de la caja: las solicitudes de usuarios de Internet se autentican mediante un módulo de autenticación situado en el proxy inverso del nivel web. Por lo tanto, todas las solicitudes a la capa de aplicación se autentican en la web antes de pasar a la capa de aplicación protegida.
- Independiente de la plataforma: la solución se puede implementar con aplicaciones de servidor locales o en la nube.
- Independiente de la tecnología: la solución se puede implementar en un entorno de máquina virtual o en contenedores. También se puede implementar en Windows o Linux. Sin embargo, esta versión inicial de la arquitectura de referencia y la documentación se ha desarrollado para Linux que se ejecuta en AWS.
- Amplia disponibilidad: todos los componentes del sistema están implementados como un clúster y diseñados para operar en un despliegue activo/activo o activo/pasivo.
- Roles en silos: cada servidor desempeña un papel discreto. Este diseño divide todos los servidores de modo que el acceso se minimiza a los administradores específicos del servicio. Por ejemplo, los DBA administran PostgreSQL para Tableau, los administradores de identidad administran el módulo de autenticación en el nivel web, los administradores de red y nube habilitan el tráfico y la conectividad.
- Linealmente escalable: como roles discretos, puede escalar cada servicio de nivel de forma independiente según el perfil de carga.
- Compatibilidad con el cliente: la arquitectura de referencia es compatible con todos los clientes de Tableau: Tableau Desktop (versiones 2021.2 o posteriores), Tableau Mobile y Creación web de Tableau.

Medidas de seguridad

Como se ha indicado, una de las funcionalidades principales del diseño de centros de datos estándar del sector es la seguridad.

- Acceso: cada nivel está limitado por una subred que aplica el control de acceso en la capa de red mediante el filtrado de puertos. La capa de aplicación también puede imponer el acceso a la comunicación entre subredes con servicios autenticados entre procesos.

- Integración: la arquitectura está diseñada para conectarse con el proveedor de identidad (IdP) en el proxy inverso del nivel web.
- Privacidad: el tráfico hacia el nivel web se cifra desde el cliente con SSL. Opcionalmente, el tráfico hacia las subredes internas también se puede cifrar.

Nivel proxy web

El nivel web es una subred en la DMZ (a la que también se hace referencia como zona de perímetro) que actúa como un búfer de seguridad entre Internet y las subredes internas donde se implementan las aplicaciones. El nivel web aloja servidores proxy inversos que no almacenan información confidencial. Los servidores proxy inversos están configurados con un complemento AuthN para autenticar previamente las sesiones del cliente con un IdP de confianza, antes de redirigir la solicitud del cliente a Tableau Server. Para obtener más información, consulte Autenticación previa con un módulo de AuthN.

Equilibrador de carga

El diseño de implementación incluye una solución de equilibrio de carga empresarial frente a los servidores proxy inversos.

Los equilibradores de carga proporcionan importantes mejoras de seguridad y rendimiento, al

- Virtualizar la URL de front-end para los servicios de nivel de aplicación
- Aplicar el cifrado SSL
- Descargar SSL
- Aplicar la compresión entre el cliente y los servicios de nivel web
- Proteger contra ataques DOS
- Proporcionar alta disponibilidad

Nota: la versión 2022.1 de Tableau Server incluye la puerta de enlace independiente de Tableau Server. La puerta de enlace independiente es una instancia independiente del proceso de puerta de enlace de Tableau que funciona como un proxy inverso compatible

con Tableau. En el momento del lanzamiento, la puerta de enlace independiente se validó, pero no se probó por completo en la arquitectura de referencia de EDG. Una vez completadas las pruebas completas, EDG se actualizará con la guía prescriptiva de la puerta de enlace independiente de Tableau Server.

Nivel de aplicación

El nivel de aplicación está en una subred que ejecuta la lógica empresarial central de la aplicación del servidor. El nivel de aplicación consta de servicios y procesos que se configuran en los nodos distribuidos de un clúster. Solo se puede acceder al nivel de aplicación desde el nivel web y los usuarios no pueden acceder directamente a él.

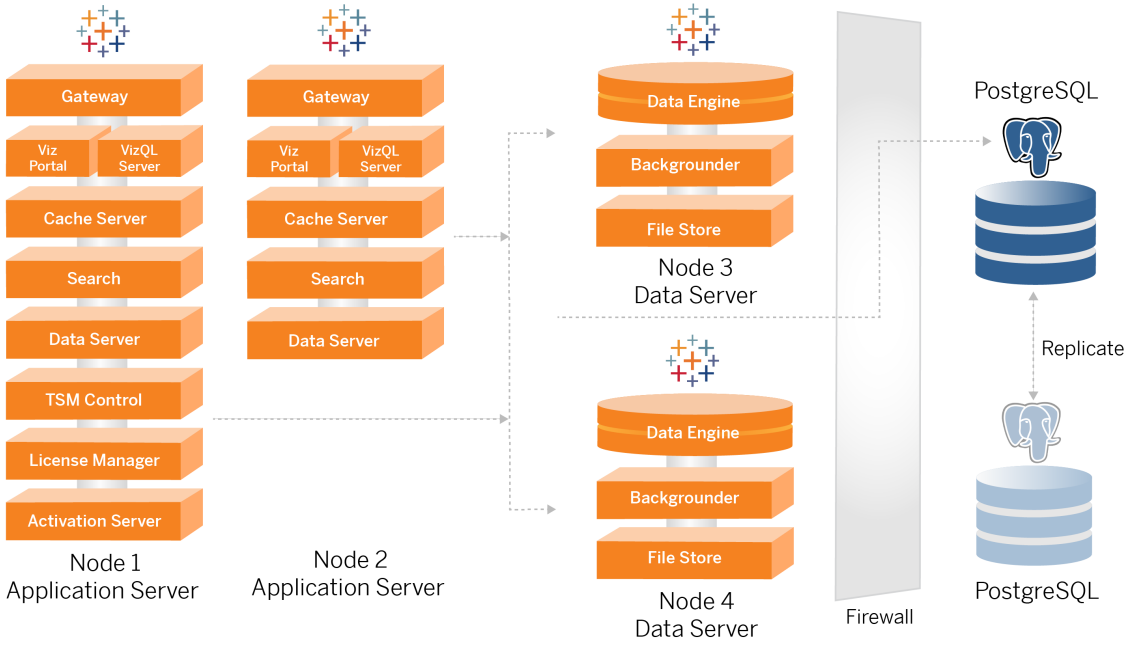
El rendimiento y la fiabilidad se mejoran mediante la configuración de los procesos de la aplicación de manera que los procesos con diferentes perfiles de uso de recursos (es decir, uso intensivo de CPU frente a uso intensivo de memoria) se ubican conjuntamente.

Nivel de datos

El nivel de datos es una subred que contiene datos valiosos. Todo el tráfico a este nivel se origina en el nivel de aplicación y, por lo tanto, ya está autenticado. Además de los requisitos de acceso en la capa de red con la configuración del puerto, esta capa debe incluir acceso autenticado y, de manera opcional, tráfico cifrado con el nivel de aplicación.

Parte 2: Comprender la arquitectura de referencia de la implementación de Tableau Server

La siguiente imagen muestra los procesos relevantes de Tableau Server y cómo se implementan en la arquitectura de referencia. Esta implementación se considera la implementación mínima de Tableau Server apropiada para la empresa.



Los diagramas de proceso de este tema pretenden mostrar los principales procesos definitorios de cada nodo. Hay muchos procesos de soporte que también se ejecutan en los nodos que no se muestran en los diagramas. Para obtener una lista de todos los procesos, consulte la sección de configuración de esta guía, Paso 4: Instalar y configurar Tableau Server.

Procesos de Tableau Server

La arquitectura de referencia de Tableau Server es una implementación de clústeres de Tableau Server de cuatro nodos con un repositorio externo en PostgreSQL:

- Nodo inicial de Tableau Server (nodo 1): ejecuta los servicios administrativos y de licencias de TSM necesarios que solo se pueden ejecutar en un único nodo del clúster. En el contexto empresarial, el nodo inicial de Tableau Server es el nodo principal del clúster. Este nodo también ejecuta servicios de aplicaciones redundantes con el nodo 2.
- Nodos de aplicación de Tableau Server (nodo 1 y nodo 2): los dos nodos atienden solicitudes de clientes, se conectan y consultan fuentes de datos y a los nodos de datos.
- Nodos de datos de Tableau Server (nodo 3 y nodo 4): dos nodos que se dedican a administrar datos.
- PostgreSQL externo: este host ejecuta el proceso del repositorio de Tableau Server. Para conseguir una implementación de alta disponibilidad, debe ejecutar un host PostgreSQL adicional para la redundancia activa/pasiva.

  También puede ejecutar PostgreSQL en Amazon RDS. Para obtener más información sobre las diferencias entre ejecutar el repositorio en RDS y una instancia EC2, consulte *Repositorio externo de Tableau Server* (Linux).

  La implementación de Tableau con un repositorio externo requiere una licencia de Tableau Advanced Management.

  Si su organización no tiene experiencia interna en administradores de bases de datos, puede ejecutar opcionalmente el proceso del repositorio de Tableau Server en la configuración interna predeterminada de PostgreSQL. En el escenario predeterminado, el repositorio se ejecuta en un nodo de Tableau con PostgreSQL insertado. En este caso, recomendamos ejecutar el repositorio en un nodo de Tableau dedicado y un repositorio pasivo en un nodo dedicado adicional para admitir la conmutación por error del repositorio. Consulte *Conmutación por error del repositorio* (Linux).

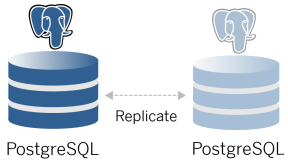
  A modo de ejemplo, la implementación de AWS descrita en esta guía explica cómo implementar el repositorio externo en PostgreSQL que se ejecuta en una instancia EC2.

- Opcional: si su organización utiliza almacenamiento externo, puede implementar el almacén de archivos de Tableau como un servicio externo. Esta guía no incluye el almacén de archivos externo en el escenario de implementación principal. Consulte *Instalar Tableau Server con el almacén de archivos externo* (Linux).

  La implementación de Tableau con un almacén de archivos externo requiere una licencia de Tableau Advanced Management.

Repositorio PostgresSQL

El repositorio de Tableau Server es una base de datos PostgresSQL que almacena datos de servidor. Estos datos incluyen información sobre usuarios, grupos y asignaciones de grupo, permisos, proyectos, fuentes de datos, metadatos de extractos e información de actualización de Tableau Server.



La implementación predeterminada de PostgresSQL consume casi el 50 % de los recursos de memoria del sistema. Según su uso (para producción y grandes implementaciones de producción), el uso de recursos puede aumentar. Por esta razón, recomendamos ejecutar el proceso del repositorio en un equipo que no esté ejecutando ningún otro componente de servidor que consuma muchos recursos, como VizQL, un procesador en segundo plano o motor de datos. La ejecución del proceso del repositorio junto con cualquiera de estos componentes creará conflictos de IO, restricciones de recursos y degradará el rendimiento general de la implementación.

Nodo 1: nodo inicial

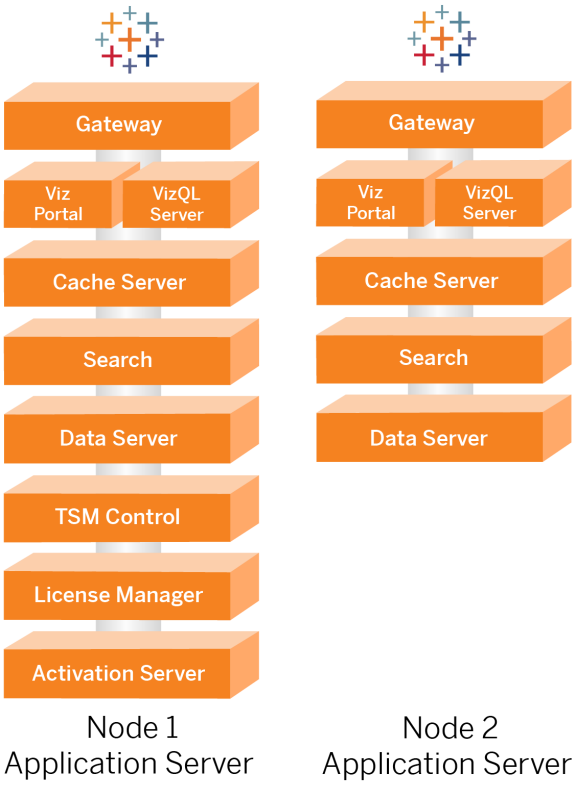
El nodo inicial ejecuta una pequeña cantidad de procesos importantes y comparte la carga de la aplicación con el nodo 2.

El primer equipo en el que instale Tableau, el "nodo inicial", tiene algunas características exclusivas. Tres procesos se ejecutan solo en el nodo inicial y no se pueden mover a cualquier otro nodo, excepto en un contexto de error, el Servicio de licencias (Administrador de licencias), el Servicio de activación y el Controlador de TSM (Controlador de administración).

Conmutación por error del nodo 1 y restauración automatizada

Los servicios de licencia, activación y controlador de TSM son fundamentales para el buen estado de una implementación de Tableau Server. En el caso de una error del nodo 1, los usuarios aún podrán conectarse a la implementación de Tableau Server, ya que una arquitectura de referencia configurada correctamente enrutará las solicitudes al nodo 2. Sin embargo, sin estos servicios básicos, la implementación estará en un estado crítico de error pendiente. Consulte Recuperación automatizada inicial del nodo.

Nodos 1 y 2: servidores de aplicaciones



Los nodos 1 y 2 ejecutan los procesos de Tableau Server que atienden las solicitudes de los clientes, consultan las fuentes de datos, generan visualizaciones, manejan el contenido y la administración, y otra lógica empresarial de Tableau. Los servidores de aplicaciones no almacenan datos de usuario.

Nota: "Servidor de aplicaciones" es un término que también hace referencia a un proceso de Tableau Server que se incluye en TSM. El proceso subyacente para el "Servidor de aplicaciones" es VizPortal.

Ejecutar en paralelo, el nodo 1 y el nodo 2 se escalan para atender las solicitudes de la lógica de equilibrio de carga que se ejecuta en los servidores proxy inversos. Como nodos redundantes, si uno de estos nodos falla, las solicitudes del cliente y el servicio quedan a cargo del nodo restante.

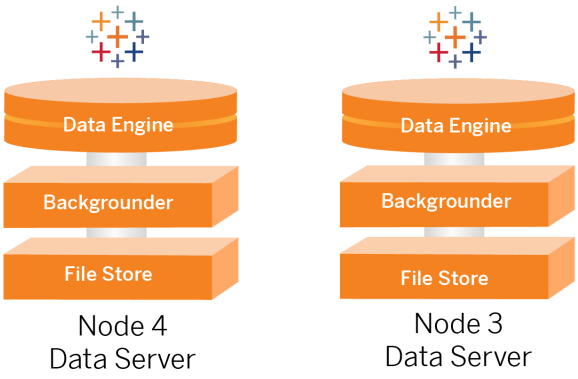
La arquitectura de referencia se ha diseñado para que los procesos de aplicación complementarios se ejecuten en el mismo equipo. Esto significa que los procesos no compiten por los recursos informáticos y no crean contención.

Por ejemplo: VizQL, un servicio de procesamiento central en los servidores de aplicaciones, está muy vinculado a la CPU y a la memoria; VizQL utiliza casi el 60-70 % de la CPU y la memoria del equipo. Por esta razón, la arquitectura de referencia está diseñada para que ningún otro proceso vinculado a la memoria o CPU esté en el mismo nodo que VizQL. Las pruebas muestran que la cantidad de carga y el número de usuarios no afectan a la memoria o el uso de la CPU en los nodos VizQL. Por ejemplo, reducir la cantidad de usuarios simultáneos en nuestra prueba de carga solo afecta al rendimiento del dashboard o al proceso de carga de visualización, pero no reduce la utilización de recursos. Por lo tanto, según la memoria disponible y la CPU durante el uso máximo, puede considerar la posibilidad de agregar más procesos VizQL. Como punto de partida para los libros de trabajo típicos, asigne 4 núcleos por proceso VizQL.

Escalar servidores de aplicaciones

La arquitectura de referencia está diseñada para escalarse en función de un modelo basado en el uso. Como punto de partida general, recomendamos un mínimo de dos servidores de aplicaciones, cada uno de los cuales con una capacidad de 1000 usuarios. A medida que aumente la base de usuarios, agregue un servidor de aplicaciones por cada 1000 usuarios adicionales. Supervise el uso y el rendimiento para ajustar la base de usuarios por host para su organización.

Nodos 3 y 4: servidores de datos



Los procesos Almacén de archivos, Motor de datos (Hyper) y Procesador en segundo plano están ubicados en los nodos 3 y 4 por las siguientes razones:

- Optimización de extracción: ejecutar el procesador en segundo plano, Hyper y el almacén de archivos en el mismo nodo optimiza el rendimiento y la fiabilidad. Durante el proceso de extracción, el procesador en segundo plano consulta la base de datos de destino, crea el archivo Hyper en el mismo nodo y luego lo carga en el almacén de archivos. Al ubicar estos procesos en el mismo nodo, el flujo de trabajo de creación de extracción no requiere copiar grandes cantidades de datos a través de la red o los nodos.
- Equilibrio de recursos gratuito: el procesador en segundo plano hace principalmente un uso intensivo de la CPU. El motor de datos es un proceso que consume mucha memoria. La combinación de estos procesos permite la máxima utilización de recursos en cada nodo.

- Consolidación de procesos de datos: dado que cada uno de estos procesos son procesos de datos de back-end, tiene sentido ejecutarlos en el nivel de datos más seguro. En versiones futuras de la arquitectura de referencia, los servidores de aplicaciones y datos se ejecutarán en niveles separados. Sin embargo, debido a las dependencias de las aplicaciones en la arquitectura de Tableau, los servidores de aplicaciones y datos deben ejecutarse en el mismo nivel en este momento.

Escalar servidores de datos

Al igual que con los servidores de aplicaciones, la planificación de los recursos necesarios para los servidores de datos de Tableau requiere un modelo basado en el uso. En general, suponga que cada servidor de datos puede admitir hasta 2000 trabajos de actualización de extracción por día. A medida que aumenten los trabajos de extracción, agregue servidores de datos adicionales sin el servicio del almacén de archivos. Generalmente, la implementación del servidor de datos de dos nodos es adecuada para implementaciones que utilizan el sistema de archivos local para el servicio del almacén de archivos. Tenga en cuenta que agregar más servidores de aplicaciones no afecta el rendimiento ni la escala de los servidores de datos de forma lineal. De hecho, con la excepción de algunos gastos generales de consultas de usuarios adicionales, el impacto de agregar más usuarios y hosts de aplicaciones es mínimo.

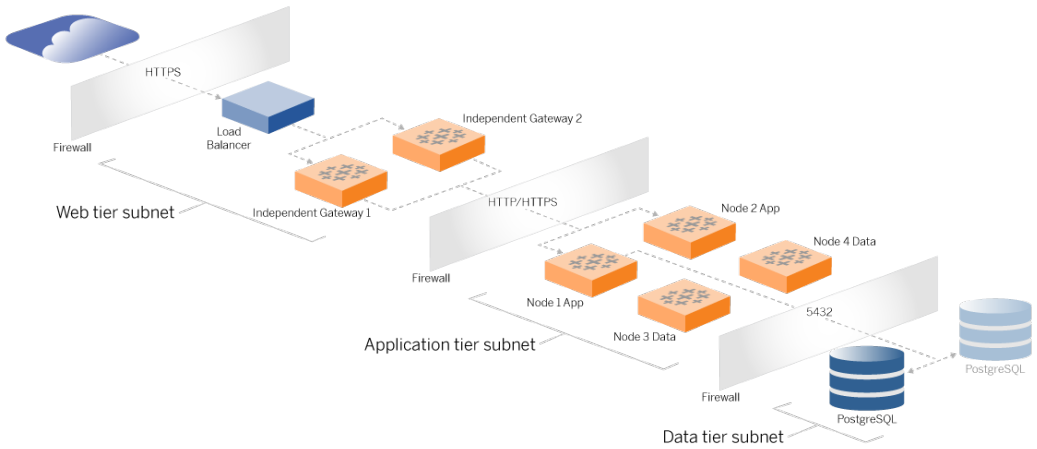
Parte 3: preparación para la implementación de Tableau Server Enterprise

La parte 3 describe los requisitos para preparar su infraestructura para implementar la arquitectura de referencia de Tableau Server. Antes de comenzar, recomendamos revisar la Parte 2: Comprender la arquitectura de referencia de la implementación de Tableau Server.

Además de las descripciones de los requisitos, este tema proporciona un ejemplo de implementación de la arquitectura de referencia en un entorno de AWS. El resto de esta guía se basa en el ejemplo de arquitectura de referencia de AWS que se inició en este tema.

Un principio fundamental de la arquitectura de referencia es la estandarización con las prácticas recomendadas de seguridad del centro de datos. En concreto, la arquitectura está diseñada para segregar servicios en subredes de red protegidas. La comunicación entre subredes está restringida a un protocolo específico y al tráfico de puertos.

El siguiente diagrama ilustra el diseño de subred de la arquitectura de referencia para una implementación local o una implementación en la nube administrada por el cliente. Para ver un ejemplo de implementación en la nube, consulte la sección siguiente, Ejemplo: configurar subredes y grupos de seguridad en AWS.



Subredes

Cree tres subredes:

- Un nivel web
- Un nivel de aplicación
- Una subred de datos.

Firewall/reglas del grupos de seguridad

Las pestañas a continuación describen las reglas de firewall para cada nivel del centro de datos. Para conocer las reglas de grupo de seguridad específicas de AWS, consulte la sección más adelante en este tema.

Nivel web

El nivel web es una subred DMZ pública que manejará las solicitudes HTTPS entrantes y enviará las solicitudes al nivel de aplicación. Este diseño proporciona una capa de defensa contra el malware que puede estar dirigido a su organización. El nivel web bloquea el acceso al nivel de aplicación/datos.

Tráfico	Tipo	Protocolo	Intervalo de puertos	Fuente
Entrante	SSH	TCP	22	Subred Bastion (para implementaciones en la nube)
Entrante	HTTP	TCP	80	Internet (0.0.0.0/0)
Entrante	HTTPS	TCP	443	Internet (0.0.0.0/0)
Saliente	Todo el tráfico	Todos	Todos	

Nivel de aplicación

La subred de la aplicación es donde reside la implementación de Tableau Server. La subred de la aplicación incluye los servidores de aplicaciones de Tableau (Nodo 1 y Nodo 2). Los servidores de aplicaciones de Tableau procesan las solicitudes de los usuarios a los servidores de datos y ejecutan la lógica empresarial central.

La subred de la aplicación también incluye los servidores de datos de Tableau (Nodo 3 y Nodo 4).

Todo el tráfico del cliente al nivel de la aplicación se autentica en el nivel web. El acceso administrativo a la subred de la aplicación se autentica y se enruta a través del host de Bastion.

Traffic	Tipo	Protocolo	Intervalo de puertos	Fuente
Entrante	SSH	TCP	22	Subred Bastion (para implementaciones en la nube)

Entrante	HTTPS	TCP	443	Subred de nivel web
Saliente	Todo el tráfico	Todos	Todos	

Nivel de datos

La subred de datos es donde reside el servidor de la base de datos PostgreSQL externa.

Traffic	**Tipo**	**Protocolo**	**Intervalo de puertos**	**Fuente**
Entrante	SSH	TCP	22	Subred Bastion (para implementaciones en la nube)
Entrante	PostgreSQL	TCP	5432	Subred de nivel de aplicación
Saliente	Todo el tráfico	Todos	Todos	

Bastion

La mayoría de los equipos de seguridad empresarial no permiten la comunicación directa desde el sistema administrativo local a los nodos implementados en la nube. En cambio, todo el tráfico SSH administrativo a los nodos de la nube se envía mediante proxy a través de un host de Bastion (también denominado "servidor de salto"). Para las implementaciones en la nube, recomendamos la conexión proxy del host de Bastion a todos los recursos de la arquitectura de referencia. Esta es una configuración opcional para entornos locales.

El host de Bastion autentica el acceso administrativo y solo permite el tráfico a través del protocolo SSH.

Traffic	Tipo	Protocolo	Intervalo de puertos	Fuente	Destino
Entrante	SSH	TCP	22	Dirección IP del equipo de administración	
Saliente	SSH	TCP	22		Subred de nivel web
Saliente	SSH	TCP	22		Subred de nivel de aplicación

Ejemplo: configurar subredes y grupos de seguridad en AWS

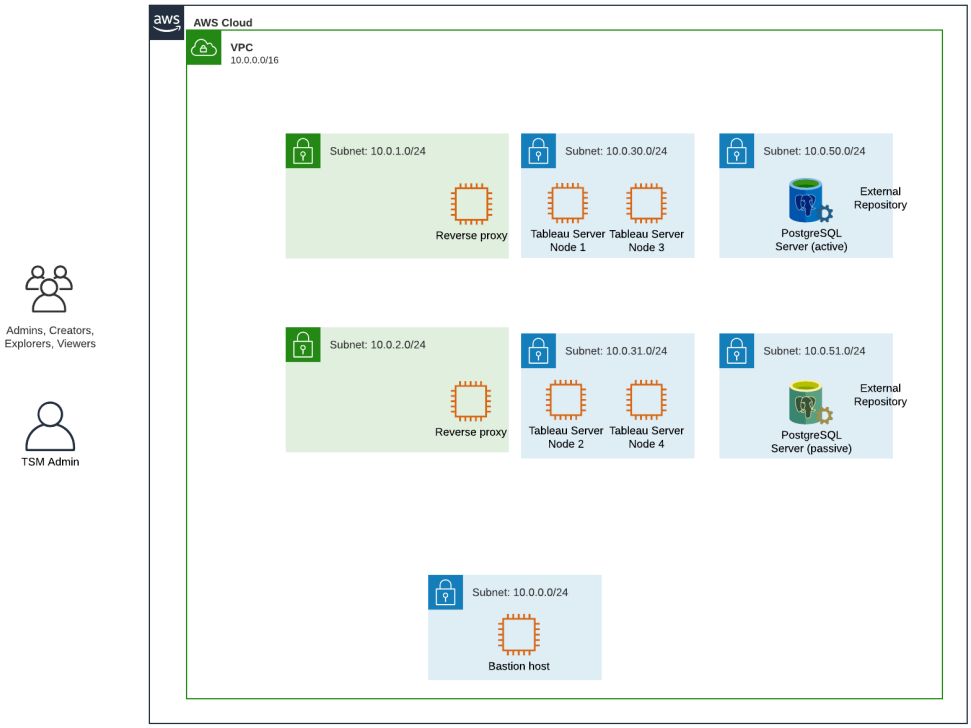
Esta sección proporciona procedimientos paso a paso para crear y configurar el entorno de red y VPC para la implementación de la arquitectura de referencia de Tableau Server en AWS.

Las diapositivas siguientes muestran la arquitectura de referencia en cuatro capas. A medida que avanza por las diapositivas, los elementos de los componentes se superponen en el mapa de topología:

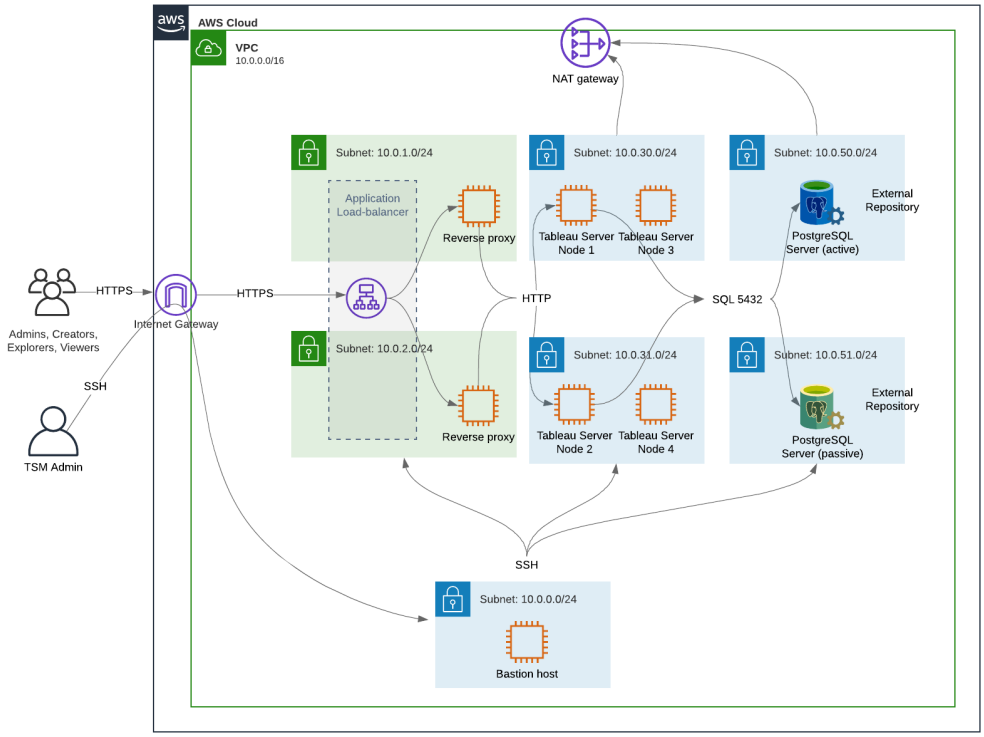
1. Topología de subred de VPC e instancias EC2: un host de Bastion, dos servidores proxy inversos, cuatro servidores Tableau y al menos un servidor PostgreSQL.
2. Flujo de protocolo y conectividad a Internet. Todo el tráfico entrante se administra a través de la puerta de enlace de Internet de AWS. El tráfico a Internet se enruta a través de NAT.
3. Zonas de disponibilidad. El proxy, Tableau Server y los hosts de PostgreSQL se implementan de manera uniforme en dos zonas de disponibilidad.
4. Grupos de seguridad. Cuatro grupos de seguridad (público, privado, de datos y Bastion) protegen cada nivel a nivel de protocolo.

Arquitectura de referencia de AWS

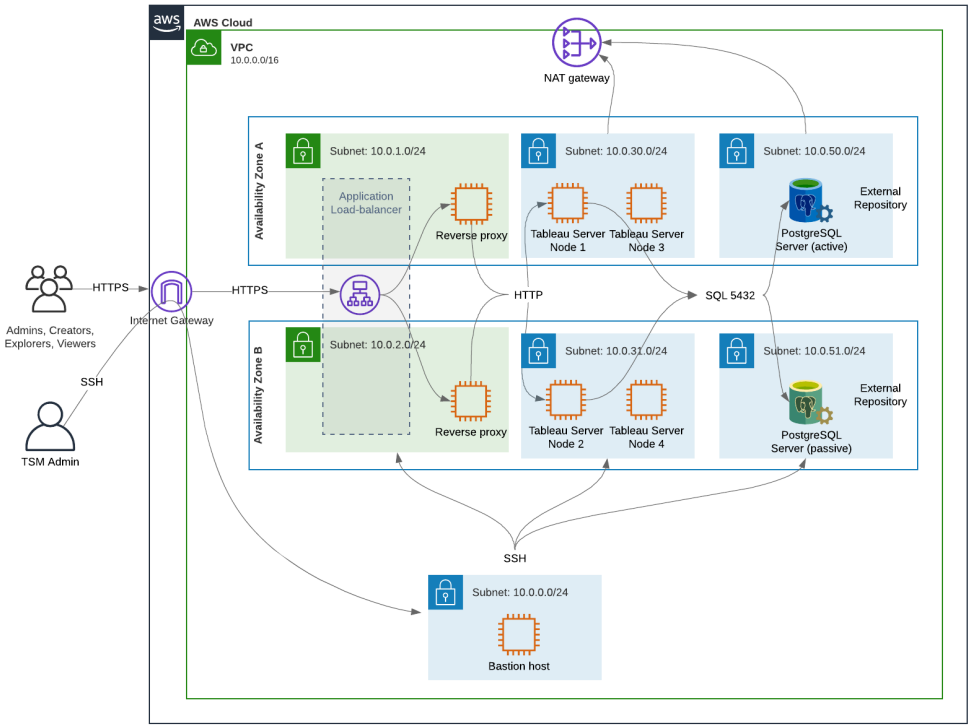
Diapositiva 1: Topología de subred de VPC e instancias EC2



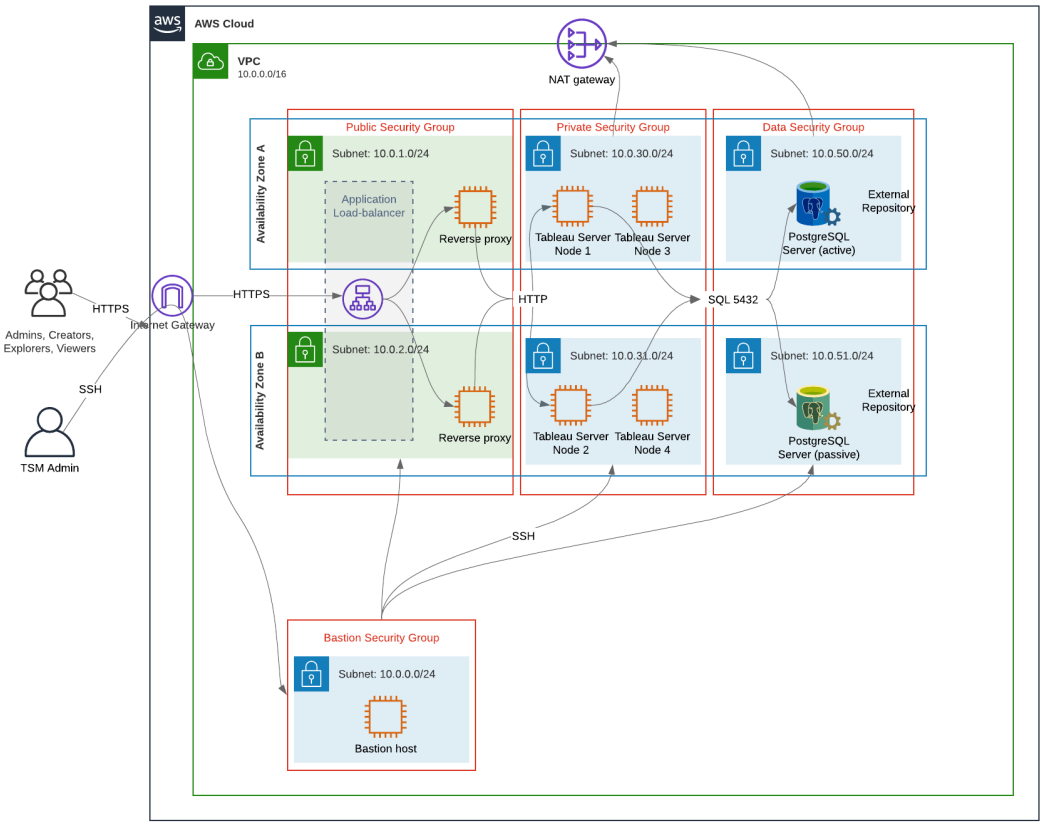
Diapositiva 2: Flujo de protocolo y conectividad



Diapositiva 3: Zonas de disponibilidad



Diapositiva 4: Grupos de seguridad



Zonas de disponibilidad de AWS y alta disponibilidad

La arquitectura de referencia que se presenta en esta guía especifica una implementación que proporciona disponibilidad a través de la redundancia cuando falla un solo host. Sin embargo, en el caso de AWS, donde la arquitectura de referencia se implementa en dos zonas de disponibilidad, la disponibilidad no sería segura en el caso muy poco común de que falle una zona de disponibilidad.

Configuración VPC

Esta sección describe cómo:

- Instalar y configurar el VPC
- Configurar la conectividad a Internet
- Configurar subredes
- Crear y configurar grupos de seguridad

Configurar VPC

El procedimiento de esta sección se asigna a la interfaz de usuario en la experiencia de VPC "clásica". Puede alternar la interfaz de usuario para mostrar la vista clásica desactivando la Nueva experiencia de VPC en la esquina superior izquierda del panel de AWS VPC.

Ejecute el asistente para VPC para crear subredes privadas y públicas predeterminadas y enrutamiento predeterminado y ACL de red.

1. Antes de configurar una VPC, debe crear una IP Elastic. Cree una asignación utilizando todos los valores predeterminados.
2. Ejecute el asistente para VPC> "VPC con subredes públicas y privadas"
3. Acepte la mayoría de los valores predeterminados. Excepto por lo siguiente:
    - Escriba un nombre de VPC.
    - Especifique el ID de asignación de IP Elastic.
    - Especifique las siguientes máscaras CIDR:
        - Subred pública IPv4 CIDR: 10.0.1.0/24, cambie el nombre de esta subred a `Public-a`.
        - Subred privada IPv4 CIDR: 10.0.30.0/24, cambie el nombre de esta subred a `Private-a`.
    - Zona de disponibilidad: para ambas subredes, seleccione la opción **a** para la región en la que se encuentra.

      Nota: Para los propósitos de este ejemplo, utilizamos **a** y **b** para distinguir entre zonas de disponibilidad en un determinado centro de datos de AWS. En AWS, es posible que los nombres de las zonas de disponibilidad no coincidan con los ejemplos que se muestran aquí. Por ejemplo, algunas zonas de disponibilidad incluyen las zonas **c** y **d** dentro de un centro de datos.

4. Haga clic en **Crear VPC**.
5. Una vez creada la VPC, cree las subredes `Public-b`, `Private-b`, `Data` y `Bastion`. Para crear una subred, haga clic en **Subredes** > **Crear subred** .

    - `Public-b`: Para Zona de disponibilidad, seleccione la opción **b** para la región en la que se encuentra. Bloque de CIDR: 10.0.2.0/24
    - `Private-b`: Para Zona de disponibilidad, seleccione la opción **b** para la región en la que se encuentra. Bloque de CIDR: 10.0.31.0/24
    - `Data`: Para Zona de disponibilidad, seleccione la zona **a** para la región en la que se encuentra. Bloque de CIDR: 10.0.50.0/24 Opcional: Si planea replicar la base de datos externa en un clúster de PostgreSQL, cree una subred Data-b en la zona de disponibilidad b con un bloque CIDR de 10.0.51.0/24.
    - `Bastion`: para Zona de disponibilidad, seleccione una de las zonas. Bloque CIDR: 10.0.0.0/24
6. Una vez creadas las subredes, edite las tablas de enrutamiento en las subredes Public y Bastion para usar la tabla de enrutamiento que está configurada para la puerta de enlace de Internet asociada (IGW). Después, edite las subredes Private y Data para usar la tabla de enrutamiento que está configurada para el traductor de direcciones de red (NAT).
    - Para determinar qué tabla de enrutamiento está configurada con IGW o NAT, haga clic en **Tablas de enrutamiento** en el panel de AWS. Seleccione uno de los dos enlaces de la tabla de rutas para abrir la página de propiedades. Mire el valor de destino en **Rutas** > **Destino** > **0.0.0.0/0**. El valor de destino diferencia el tipo de ruta y comenzará con la cadena `igw-` o `nat-`.
    - Para actualizar las tablas de enrutamiento, **VPC** > **Subredes** > [nombre_subred] > **Tabla de enrutamiento** > **Editar asociación de la tabla de enrutamiento**.

Configurar grupos de seguridad

El asistente de VPC crea un único grupo de seguridad que no utilizará. Cree los siguientes grupos de seguridad (**Grupos de seguridad** > **Crear grupo de seguridad**). Los hosts EC2 se instalarán en estos grupos en dos zonas de disponibilidad, como se muestra en el diagrama de diapositivas anterior.

- Cree un nuevo grupo de seguridad: **Privado**. Aquí es donde se instalarán los 4 nodos de Tableau Server. Más adelante durante la instalación, el grupo de seguridad privado

se asociará con las subredes 10.0.30.0/24 y 10.0.31.0/24.

- Cree un nuevo grupo de seguridad: **Público**. Aquí es donde se instalarán los servidores proxy. Más adelante durante la instalación, el grupo de seguridad público se asociará con las subredes 10.0.1.0/24 y 10.0.2.0/24.
- Cree un nuevo grupo de seguridad: **Datos**. Aquí es donde se instalará el repositorio de Tableau externo de PostgreSQL. Más adelante durante la instalación, el grupo de seguridad de datos se asociará con la subred 10.0.50.0/24 (y, opcionalmente, con la 10.0.51.0/24).
- Cree un nuevo grupo de seguridad: **Bastion**. Aquí es donde instalará el host de Bastion. Más adelante durante la instalación, el grupo de seguridad Bastion se asociará con las subredes 10.0.1.0/24 y 10.0.0.0/24.

Especificar reglas de entrada y salida

En AWS, los grupos de seguridad son análogos a los firewalls en un entorno local. Debe especificar el tipo de tráfico (por ejemplo, Https, https, etc.), el protocolo (TCP o UDP) y los puertos o intervalo de puertos (por ejemplo, 80, 443, etc.) que pueden entrar o salir del grupo de seguridad. Para cada protocolo, también debe especificar el tráfico de origen o destino.

Reglas del grupo de seguridad público

Reglas de entrada			
Tipo	**Protocolo**	**Intervalo de puertos**	**Fuente**
HTTP	TCP	80	0.0.0.0/0
HTTPS	TCP	443	0.0.0.0/0
SSH	TCP	22	Grupo de seguridad Bastion

Reglas de salida			
Tipo	**Protocolo**	**Intervalo de puertos**	**Destino**

Todo el tráfico	Todos	Todos	0.0.0.0/0

Reglas del grupo de seguridad privado

El grupo de seguridad privado incluye una regla de entrada para permitir el tráfico HTTP del grupo de seguridad público. Permita el tráfico HTTP solo durante el proceso de implementación para verificar la conectividad. Recomendamos eliminar la regla de entrada HTTP una vez que haya terminado de implementar el proxy inverso y configurar SSL en Tableau.

Reglas de entrada			
Tipo	**Protocolo**	**Intervalo de puertos**	**Fuente**
HTTP	TCP	80	Grupo de seguridad público
HTTPS	TCP	443	Grupo de seguridad público
PostgreSQL	TCP	5432	Grupo de seguridad de datos
SSH	TCP	22	Grupo de seguridad Bastion
Todo el tráfico	Todos	Todos	Grupo de seguridad privado

Regla de salida			
Tipo	**Protocolo**	**Intervalo de puertos**	**Destino**
Todo el tráfico	Todos	Todos	0.0.0.0/0
PostgreSQL	TCP	5432	Grupo de seguridad de datos
SSH	TCP	22	Grupo de seguridad Bastion

Reglas del grupo de seguridad de datos

Reglas de entrada			
Tipo	**Protocolo**	**Intervalo de puertos**	**Fuente**
PostgreSQL	TCP	5432	Grupo de seguridad privado
SSH	TCP	22	Grupo de seguridad Bastion

Reglas de salida			
Tipo	**Protocolo**	**Intervalo de puertos**	**Destino**
Todo el tráfico	Todos	Todos	0.0.0.0/0
PostgreSQL	TCP	5432	Grupo de seguridad privado
SSH	TCP	22	Grupo de seguridad Bastion

Reglas del grupo de seguridad del host de Bastion

Reglas de entrada			
Tipo	**Protocolo**	**Intervalo de puertos**	**Fuente**
SSH	TCP	22	La dirección IP y la máscara de red del equipo que utilizará para iniciar sesión en AWS (equipo de administración).
SSH	TCP	22	Grupo de seguridad privado
SSH	TCP	22	Grupo de seguridad público

Reglas de salida			
Tipo	**Protocolo**	**Intervalo de puertos**	**Destino**
SSH	TCP	22	La dirección IP y la máscara de red del equipo que utilizará para iniciar sesión en AWS (equipo de administración).
SSH	TCP	22	Grupo de seguridad privado
SSH	TCP	22	Grupo de seguridad público
SSH	TCP	22	Grupo de seguridad de datos
HTTPS	TCP	443	0.0.0.0/0 (Opcional: Cree esta regla si necesita acceder a Internet para descargar software de soporte en el host de Bastion)

Habilitar la asignación automática de IP pública

Esto le proporciona una dirección IP para conectarse a los servidores proxy y al host de Bastion.

Para subredes públicas y Bastion:

1. Seleccione la subred
2. En el menú **Acciones**, seleccione "Modificar la configuración de asignación automática de IP".
3. Haga clic en "Habilitar la asignación automática de direcciones IPv4 públicas".
4. Haga clic en **Guardar**.

Equilibrador de carga

Nota: Si está instalando en AWS y sigue la implementación de ejemplo de esta guía, debe instalar y configurar el equilibrador de carga de AWS más adelante en el proceso de implementación, como se describe en la Parte 5: Configuración del nivel web.

Para implementaciones locales, trabaje con sus administradores de red para implementar equilibradores de carga para admitir el nivel web de la arquitectura de referencia:

- Un equilibrador de carga de aplicaciones orientado a la web que acepta solicitudes HTTPS de clientes de Tableau y se comunica con los servidores proxy inversos.
- Proxy inverso:
  - Recomendamos un mínimo de dos servidores proxy para la redundancia y para manejar la carga de clientes.
  - Recibe tráfico HTTPS del equilibrador de carga.
  - Admite la sesión fija en el host de Tableau.
  - Configure el proxy para el equilibrio de carga por turnos para cada Tableau Server que ejecute el proceso de Gateway.
  - Maneja solicitudes de autenticación del IdP externo.
- Proxy de reenvío: Tableau Server requiere acceso a Internet para las licencias y la funcionalidad de mapas. Dependiendo de su entorno de proxy de reenvío, es posible que deba configurar listas seguras de proxy de reenvío para las URL de servicio de Tableau. Consulte *Comunicación con Internet* (Linux).

Configurar equipos host

Hardware mínimo recomendado

Las siguientes recomendaciones se basan en nuestras pruebas de datos del mundo real en la arquitectura de referencia.

Servidores de aplicaciones:

- CPU: 8 núcleos físicos (16 vCPU),
- RAM: 128 GB (16 GB/núcleo físico)
- Espacio en disco: 100 GB

Servidores de datos

- CPU: 8 núcleos físicos (16 vCPU),
- RAM: 128 GB (16 GB/núcleo físico)
- Espacio en disco: 1 TB. Si su implementación hará uso de almacenamiento externo para el almacén de archivos de Tableau, necesitará calcular el espacio de disco apropiado. Consulte *Instalar Tableau Server con el almacén de archivos externo* (Linux).

Servidores proxy

- CPU: 2 núcleos físicos (4 vCPU),
- RAM: 8 GB (4 GB/núcleo físico)
- Espacio en disco: 100 GB

Base de datos del repositorio externo

- CPU: 8 núcleos físicos (16 vCPU),
- RAM: 128 GB (16 GB/núcleo físico)
- El requisito de espacio en disco depende de la carga de datos y de cómo afectará a la copia de seguridad. Consulte la sección, *Procesos de copia de seguridad y restauración*, en el tema *Requisitos de espacio en disco* (Linux).

Estructura de directorios

La arquitectura de referencia recomienda instalar el paquete de Tableau Server y los datos en ubicaciones no predeterminadas:

- Instalar paquete en: `/app/tableau_server`. Cree esta ruta de directorio antes de instalar el paquete de Tableau Server y luego especifique esta ruta durante la instalación.
- Instale los datos de Tableau para: `/data/tableau_data`. No cree este directorio antes de instalar Tableau Server. En su lugar, debe especificar la ruta durante la instalación y luego Tableau Setup creará y otorgará el permiso correspondiente a la ruta.

Consulte Ejecutar el paquete de instalación e inicializar TSM para obtener detalles sobre la implementación.

Ejemplo: instalar y preparar equipos host en AWS

Esta sección explica cómo instalar hosts EC2 para cada tipo de servidor en la arquitectura de referencia de Tableau Server.

La arquitectura de referencia requiere ocho hosts:

- Cuatro instancias para Tableau Server.
- Dos instancias para servidores proxy (Apache).
- Una instancia para el host de Bastion.
- Una o dos instancias de base de datos EC2 PostgreSQL

Detalles de la instancia de host

Instale los equipos host de acuerdo con los detalles a continuación.

Tableau Server

- Amazon Linux 2
- Tipo de instancia: m5a.8xlarge
- ID de grupo de seguridad: Privado
- Almacenamiento: EBS, 150 GiB, tipo de volumen gp2. Si su implementación hará uso de almacenamiento externo para el almacén de archivos de Tableau, necesitará calcular el espacio de disco apropiado. Consulte *Instalar Tableau Server con el almacén de archivos externo* (Linux).
- Red: instale dos hosts EC2 en cada subred privada (10.0.30.0/24 y 10.0.31.0/24).
- Copie el paquete rpm de versiones de mantenimiento de Tableau Server 2021.2 (o versiones posteriores) más reciente desde la página de descargas de Tableau en cada host de Tableau.

Host de Bastion

- Amazon Linux 2
- Tipo de instancia: t3.micro
- ID de grupo de seguridad: Bastion
- Almacenamiento: EBS, 50 GiB, tipo de volumen gp2
- Red: subred de Bastion 10.0.0.0/24

Puerta de enlace independiente de Tableau Server

- Amazon Linux 2
- Tipo de instancia: t3.xlarge
- ID de grupo de seguridad: Público
- Almacenamiento: EBS, 100 GiB, tipo de volumen gp2
- Red: instale una instancia EC2 en cada subred pública (10.0.1.0/24 y 10.0.2.0/24)

Host EC2 de PostgreSQL

- Amazon Linux 2
- Tipo de instancia: r5.4xlarge
- ID de grupo de seguridad: Datos
- Almacenamiento: el requisito de espacio en disco depende de la carga de datos y de cómo afectará a la copia de seguridad. Consulte la sección, *Procesos de copia de seguridad y restauración*, en el tema *Requisitos de espacio en disco* (Linux).
- Red: subred privada 10.0.50.0/24. (Si está replicando PostgreSQL en un clúster de alta disponibilidad, instale el segundo host en la subred 10.0.51.0/24)

Verificación: conectividad VPC

Una vez que haya instalado los equipos host, verifique la configuración de la red. Compruebe la conectividad entre los hosts conectándose con SSH desde el host en el grupo de seguridad de Bastion a los hosts en cada subred.

Ejemplo: conectarse al host de Bastion en AWS

1. Configure su equipo de administración para ssh-agent. Esto le permite conectarse a hosts en AWS sin colocar su archivo de clave privada en ninguna instancia EC2.

   Para configurar ssh-agent en un Mac, ejecute el siguiente comando:

   `ssh-add -K myPrivateKey.pem`o para Mac OS más reciente, `ssh-add --apple-use-keychain myPrivateKey.pem`

   Para Windows, consulte el tema Conexión segura a instancias de Linux que se ejecutan en una Amazon VPC privada.

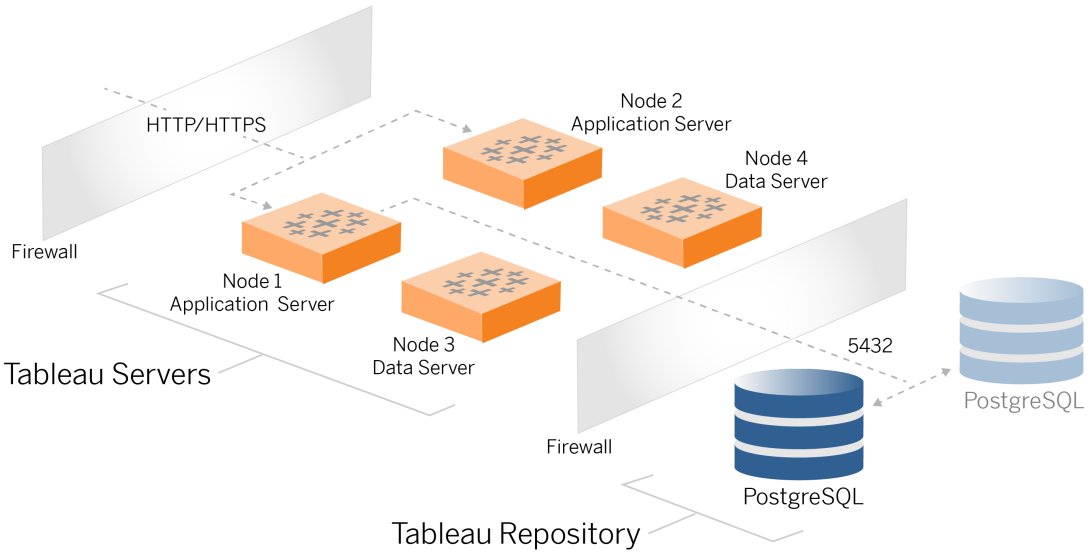
2. Conéctese al host de Bastion ejecutando el siguiente comando:

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

3. Después puede conectarse a otros hosts en la VPC desde el host de Bastion, utilizando la dirección IP privada, por ejemplo:

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

Paso 4: Instalar y configurar Tableau Server



Este tema describe cómo finalizar la instalación y configuración de la implementación de referencia de Tableau Server. El procedimiento aquí continúa con el ejemplo de arquitectura de referencia de AWS y Linux.

Los ejemplos de Linux a lo largo de los procedimientos de instalación muestran comandos para distribuciones similares a RHEL. Específicamente, los comandos aquí se han desarrollado con la distribución de Amazon Linux 2. Si está ejecutando distribuciones de Ubuntu, edite los comandos según corresponda.

Antes de empezar

Debe preparar y validar su entorno como se describe en la Parte 3: preparación para la implementación de Tableau Server Enterprise.

Instalar, configurar y convertir en .tar PostgreSQL

Esta instancia de PostgreSQL aloja el repositorio externo para la implementación de Tableau Server. Debe instalar y configurar PostgreSQL antes de instalar Tableau.

Puede ejecutar PostgreSQL en Amazon RDS o en una instancia EC2. Para obtener más información sobre las diferencias entre ejecutar el repositorio en RDS y una instancia EC2, consulte *Repositorio externo de Tableau Server* (Linux).

A modo de ejemplo, el procedimiento siguiente muestra cómo instalar y configurar Postgres en una instancia Amazon EC2. El ejemplo que se muestra aquí es una instalación y una configuración genéricas para PostgreSQL en la arquitectura de referencia. Su DBA debería optimizar su implementación de PostgreSQL según el tamaño de sus datos y las necesidades de rendimiento.

Requisitos: tenga en cuenta que debe ejecutar PostgreSQL 1.6 y debe instalar el módulo uuid-ossp.

Versiones de PostgreSQL

Debe instalar versiones principales compatibles de PostgreSQL para el repositorio externo de Tableau Server. Además, las versiones secundarias también deben cumplir con los requisitos mínimos.

Versiones de Tableau Server	Versiones mínimas compatibles con PostgreSQL
2021.2.3 - 2021.2.8 2021.3.0 - 2021.3.7 2021.4.0 - 2021.4.3	12.6
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 2021.3.14 - 2021.3.15 2021.4.9 - 2021.4.10	12.10
2021.2.17 - 2021.2.18 2021.3.16 - 2021.3.17 2021.4.11 - 2021.4.12	12.11
2021.3.26 2021.4.23	12.15
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 2022.3.0 - 2022.3.7 2023.1.0 - 2023.1.4	13.7
2022.1.17 - 2022.1.19 2022.3.8 - 2022.3.11 2023.1.5 - 2023.1.7 2023.3.0 - 2023.3.3	13.11
2024.0 - 2024.x	15.6

Instalar PostgreSQL

Este procedimiento de instalación de ejemplo describe cómo instalar PostgreSQL 13.6.

Inicie sesión en el host EC2 que creó en la parte anterior.

1. Ejecute la actualización para aplicar las últimas correcciones al sistema operativo Linux:

   ```
   sudo yum update
   ```

2. Cree y edite el archivo pgdg.repo. en el directorio `/etc/yum.repos.d/`. Complete el archivo con la siguiente información de configuración:

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

3. Instale Postgres 13.6:

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

4. Instale el módulo uuid-ossp:

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

5. Inicializar Postgres:

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

Configurar Postgres

Termine la instalación base configurando Postgres:

1. Actualice el archivo de configuración pg_hba, `/var/lib/pgsql/13/data/pg_hba.conf`, con las siguientes dos entradas. Cada entrada debe incluir la máscara de las subredes donde se ejecutarán en Tableau Server:

   ```
   host all all 10.0.30.0/24 password

   host all all 10.0.31.0/24 password
   ```

2. Actualice el archivo PostgreSQL `/var/lib/pgsql/13/data/postgresql.conf` agregando esta línea:

   ```
   listen_addresses = '*'
   ```

3. Configuración para iniciar Postgres al reiniciar:

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

4. Establecer contraseña de superusuario:

   ```
   sudo su - postgres

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

   > **Nota**: establezca una contraseña segura. No utilice '`StrongPassword`' como se muestra en el ejemplo aquí.

   ```
   exit
   ```

5. Reiniciar Postgres:

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

Realizar una copia de seguridad de tar del paso 1 de PostgreSQL

Cree una copia de seguridad .tar de la configuración de PostgreSQL. La creación de una instantánea de tar de la configuración actual le ahorrará tiempo si encuentra errores mientras continúa con la implementación.

Nos referiremos a esto como la copia de seguridad del Paso 1.

En el host PostgreSQL:

1. Detenga la instancia de la base de datos de Postgres:

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

2. Ejecute el siguiente comando para crear la copia de seguridad del archivo .tar:

   ```
   sudo su
   cd /var/lib/pgsql
   tar -cvf step1.13.bkp.tar 13
   exit
   ```

3. Inicie la base de datos de Postgres:

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

Restaurar el paso 1

Restaure el paso 1 si el nodo inicial de Tableau Server falla durante la instalación.

1. En el equipo en el que se ejecuta Tableau, ejecute el script obliterate para eliminar Tableau Server completamente del host:

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

2. Restaure el .tar de la fase 1 de PostgreSQL. En el equipo que ejecuta Postgres, ejecute los siguientes comandos:

```
sudo su

systemctl stop postgresql-13

cd /var/lib/pgsql

tar  -xvf step1.13.bkp.tar

systemctl start postgresql-13

exit
```

Reanude el proceso de instalación en Instalar el nodo inicial de Tableau Server.

Antes de instalar

Si está implementando Tableau de acuerdo con el ejemplo de implementación de AWS/Linux que se describe en esta guía, es posible que pueda ejecutar el script de instalación automatizado, TabDeploy4EDG. El script TabDeploy4EDG automatiza la instalación de ejemplo de la implementación de Tableau de cuatro nodos que se describe en los procedimientos siguientes. Consulte el Apéndice: Caja de herramientas de implementación de AWS.

Instalar el nodo inicial de Tableau Server

Este procedimiento describe cómo instalar el nodo inicial de Tableau Server según lo definido por la arquitectura de referencia. Con la excepción de la instalación del paquete y la inicialización de TSM, este procedimiento utiliza la línea de comandos de TSM siempre que sea posible. Además de ser independiente de la plataforma, el uso de la CLI de TSM permite una instalación más fluida en entornos virtualizados y desatendidos.

Ejecutar el paquete de instalación e inicializar TSM

Inicie sesión en el servidor host del Nodo 1.

1. Ejecute la actualización para aplicar las últimas correcciones al sistema operativo Linux:

   ```
   sudo yum update
   ```

2. Copie el paquete de instalación de la página de descargas de Tableau en el equipo host que ejecutará Tableau Server.

   Por ejemplo, en un equipo con el sistema operativo tipo RHEL de Linux, ejecute:

   ```
   wget http-
   s://downloads.tableau.com/esdalt/2022<version>/tableau-server-
   <version>.rpm
   ```

   donde `<version>` es el número de versión.

3. Descargue e instale las dependencias:

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

4. Cree la ruta `/app/tableau_server` en el directorio raíz.

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

5. Ejecute el programa de instalación y especifique la ruta de instalación `/app/-tableau_server`. Por ejemplo, en un sistema operativo tipo RHEL de Linux, ejecute:

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

6. Cambiae al directorio `/app/tableau_server/packages/scripts.<version_code>/` y ejecute el script `initialize-tsm` que aparece allí:

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

7. Una vez finalizada la inicialización, cierre el shell:

   ```
   exit
   ```

Activar y registrar Tableau Server

1. Inicie sesión en el servidor host del Nodo 1.
2. Proporcione las claves de producto de Tableau Server en este paso. Ejecute el siguiente comando para cada clave de licencia que haya comprado:

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

3. Cree un archivo de registro .json con el formato que se muestra aquí:

   ```
   {
   "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. Después de guardar los cambios en el archivo, páselo con la opción `--file` para regis-

trar Tableau Server:

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

Configurar el almacén de identidades

Nota: Si su implementación utilizará almacenamiento externo para el almacén de archivos de Tableau, deberá habilitar el almacén de archivos externo antes de configurar el almacén de identidades. Consulte *Instalar Tableau Server con el almacén de archivos externo* (Linux).

La arquitectura de referencia predeterminada utiliza un almacén de identidades local. Configure el host inicial con el almacén de identidades local pasando el archivo `config.json` con el comando `tsm settings import`.

Importe el archivo `config.json` de acuerdo con su sistema operativo:

El archivo `config.json` está incluido en la ruta de directorio scripts.<versión> (por ejemplo, `scripts.20204.21.0217.1203`) y está formateado para configurar el almacén de identidades.

Ejecute el siguiente comando para importar el archivo `config.json`:

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

Configurar Postgres externo

1. Cree un archivo .json de base de datos externa con los siguientes ajustes de configuración:

   ```
   {
   "flavor":"generic",
   "masterUsername":"postgres",
   ```

```
"host":"<instance ip address>",
"port":5432
}
```

2. Después de guardar los cambios en el archivo, pase el archivo con el siguiente comando:

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

   Se le solicitará que especifique la contraseña del nombre de usuario maestro de Postgres.

   La opción `--no-ssl` configura Tableau para usar SSL/TLS solo cuando el servidor de Postgres está configurado para SSL/TLS. Si Postgres no está configurado para SSL/TLS, la conexión no está cifrada. La Parte 6: configuración después de la instalación describe cómo habilitar SSL/TLS para la conexión de Postgres después de haber completado la primera fase de implementación.

3. Aplique los cambios.

   Ejecute este comando para aplicar los cambios y reiniciar Tableau Server:

   ```
   tsm pending-changes apply
   ```

4. Elimina el archivo de configuración que usaste en el paso 1.

Finalizar la instalación del nodo 1

1. Una vez instalado Tableau Server, debe inicializar el servidor.

   Ejecute el comando siguiente:

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

2. Cuando finalice la inicialización, debe crear una cuenta de administrador de Tableau Server.

A diferencia de la cuenta del equipo que está usando para instalar y administrar los componentes del sistema operativo de TSM, la cuenta de administrador es una cuenta de aplicación de Tableau Server se usa para crear usuarios, proyectos y sitios de Tableau Server. El administrador de Tableau Server también aplica permisos a los recursos de Tableau. Ejecute el siguiente comando para crear la cuenta del administrador inicial: En el siguiente ejemplo, el usuario se llama `tableau-admin`:

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

Tabcmd le pedirá que establezca una contraseña para este usuario.

Verificación: configuración del nodo 1

1. Ejecute el siguiente comando para verificar que los servicios de TSM se estén ejecutando:

   ```
   tsm status -v
   ```

   Tableau debería devolver lo siguiente:

   ```
   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 run-
   ning.
   'MessageBus Microservice 0' is running.
   'Relationship Query Microservice 0' is running.
   'Tableau Server VizQL Server 0' is running.
   ...
   ```

   Se enumerarán todos los servicios.

2. Ejecute el siguiente comando para verificar que el sitio de administración de Tableau se esté ejecutando:

   ```
   curl localhost
   ```

   Las primeras líneas deberían mostrar Vizportal html, similar a esto:

   ```
   <!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 ...
   ```

Realizar copias de seguridad del archivo .tar del paso 2

Una vez que haya verificado la instalación inicial, realice dos copias de seguridad del archivo .tar:

- PostgreSQL
- Nodo inicial de Tableau (nodo 1)

En la mayoría de los casos, puede recuperar su instalación del nodo inicial restaurando estos archivos .tar. Restaurar los archivos .tar es mucho más rápido que reinstalar y reinicializar el nodo inicial.

Crear archivos .tar del paso 2

1. En el nodo inicial de Tableau, detenga Tableau:

   ```
   tsm stop
   ```

   Espere a que Tableau se detenga antes de continuar con el siguiente paso.

2. En el host de PostgreSQL, detenga la instancia de la base de datos de Postgres:

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

3. Ejecute el siguiente comando para crear la copia de seguridad del archivo .tar:

   ```
   sudo su

   cd /var/lib/pgsql

   tar -cvf step2.13.bkp.tar 13

   exit
   ```

4. Compruebe que el archivo .tar de Postgres se crea con permisos raíz:

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

5. En el host de Tableau, detenga los servicios administrativos de Tableau:

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

6. Ejecute el siguiente comando para crear la copia de seguridad del archivo .tar:

   ```
   cd /data

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

7. En el host de Postgres, inicie la base de datos de Postgres:

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

8. Inicie los servicios administrativos de Tableau:

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

9. Ejecute el comando `tsm status` para supervisar el estado de TSM antes de reiniciar.

   En la mayoría de los casos, el comando devolverá primero un estado de DEGRADED o ERROR. Espere unos minutos y vuelva a ejecutar el comando. Si se devuelve el estado ERROR o DEGRADED, siga esperando. No intente iniciar TSM hasta que se devuelva el estado STOPPED. Después, ejecute el siguiente comando:

   ```
   tsm start
   ```

Restaurar el paso 2

Este proceso restaura el Nodo 1 de Tableau y la instancia de Postgres al paso 2. Una vez que haya restaurado este paso, puede volver a implementar los nodos de Tableau restantes.

1. Detenga el servicio tsm en el host inicial de Tableau (nodo 1):

   ```
   tsm stop
   ```

2. Detenga los servicios administrativos de Tableau en todos los nodos de la implementación de Tableau Server. Ejecute el siguiente comando en cada nodo, en orden (Nodo 1, Nodo 2 y luego Nodo 3):

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

3. Una vez que se hayan detenido los servicios de Tableau, restaure el archivo .tar del paso 2 de PostgreSQL. En el equipo que ejecuta Postgres, ejecute los siguientes comandos:

   - `sudo su`

     ```
     systemctl stop postgresql-13
     ```

```
cd /var/lib/pgsql

tar  -xvf step2.13.bkp.tar

systemctl start postgresql-13

exit
```

4. Restaure el archivo .tar del paso 2 de Tableau. En el host inicial de Tableau, ejecute los siguientes comandos:

   ```
   cd /data

   sudo rm -rf tableau_data

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

5. En el equipo del nodo 1 de Tableau, elimine los siguientes archivos:

   - `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. Inicie los servicios administrativos de Tableau:

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

7. Vuelva a cargar los archivos systmctl de Tableau y luego ejecute `start-administrative-services` de nuevo:

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

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

8. En el nodo 1, ejecute el comando `tsm status` para supervisar el estado de TSM antes de reiniciar.

   En algunos casos, obtendrá un error `Cannot connect to server...`. Este error se produce porque el servicio tabadmincontroller no se ha reiniciado. Siga ejecutando `tsm status` periódicamente. Si este error no desaparece después de 10 minutos, ejecute el comando `start-administrative-services` de nuevo.

   Después de unos momentos, el comando `tsm status` devolverá un estado de DEGRADED y luego ERROR. No inicie TSM hasta que se devuelva el estado STOPPED. Después, ejecute el siguiente comando:

   ```
   tsm start
   ```

Reanude el proceso de instalación para instalar Tableau Server en los nodos restantes.

Instalar Tableau Server en los nodos restantes

Para continuar con la implementación, copie el instalador de Tableau en cada nodo.

Descripción general de la configuración de los nodos

Esta sección describe el proceso para configurar los nodos 2-4. Las secciones siguientes proporcionan procedimientos detallados de configuración y validación para cada paso.

La instalación de los nodos 2 a 4 de Tableau Server requiere que genere, copie y haga referencia a un archivo de arranque durante la instalación del nodo.

Para generar el archivo de arranque, ejecute un comando TSM en el nodo inicial. Después, copie el archivo de arranque en el nodo de destino, donde lo ejecutará como parte de la inicialización del nodo.

El contenido json a continuación es un ejemplo de un archivo de arranque. (El certificado y los valores de cifrado se han truncado para facilitar la lectura del archivo de ejemplo).

```
{
"initialBootstrapSettings" : {
       "certificate" : "-----BEGIN CERTIFICATE-----\r\...=\r\n-----END
CERTIFICATE-----",
       "port" : 8850,
       "configurationName" : "tabsvc",
       "clusterId" : "tabsvc-clusterid",
       "cryptoKeyStore" : "zs7OzgAAAAIAAAABAAAAA...w==",
       "toksCryptoKeystore" : "LS0tLS1CRUdJTiBUT00tLS0tCjM5MDBh...L",
       "sessionCookieMaxAge" : 7200,
       "nodeId" : "node1",
       "machineAddress" : "ip-10-0-1-93.us-west-1.compute.internal",
       "cryptoEnabled" : true,
       "sessionCookieUser" : "tsm-bootstrap-user",
       "sessionCookieValue" : "eyJ-
jdHkiOiJKV1QiLCJlbmMiOiJBMTI4Q0JDLUhQ...",
       "sessionCookieName" : "AUTH_COOKIE"
       }
}
```

El archivo de arranque incluye la validación basada en la conexión para autenticar el nodo 1 y crea un canal cifrado para el proceso de arranque. La sesión de arranque tiene un tiempo limitado, y la configuración y validación de los nodos requiere mucho tiempo. Planee crear y copiar nuevos arranques mientras configura los nodos.

Después de ejecutar el archivo de arranque, inicie sesión en el nodo inicial de Tableau Server y configure los procesos para el nuevo nodo. Cuando termine de configurar los nodos, debe aplicar los cambios y reiniciar el nodo inicial. El nuevo nodo está configurado e iniciado. A medida que agregue nodos, la configuración y el reinicio de la implementación tardarán consecutivamente más en completarse.

Los ejemplos de Linux a lo largo de los procedimientos de instalación muestran comandos para distribuciones similares a RHEL. Si está ejecutando distribuciones de Ubuntu, edite los comandos según corresponda.

1. Ejecute la actualización para aplicar las últimas correcciones al sistema operativo Linux:

   ```
   sudo yum update
   ```

2. Descargue e instale las dependencias:

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

3. Cree la ruta `/app/tableau_server` en el directorio raíz.

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

4. Ejecute el programa de instalación y especifique la ruta de instalación `/app/-tableau_server`. Por ejemplo, en un sistema operativo tipo RHEL de Linux, ejecute:

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

Generar, copiar y ejecutar el archivo de arranque para inicializar TSM

El siguiente procedimiento muestra cómo generar, copiar y usar un archivo de arranque al inicializar TSM en otro nodo. En este ejemplo, el archivo de arranque se llama `boot.json`.

En este ejemplo, los equipos host se ejecutan en AWS, donde los hosts EC2 ejecutan Amazon Linux 2.

1. Conéctese al nodo inicial (nodo 1) y ejecute el siguiente comando:

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

2. Copie el archivo de arranque en el nodo 2.

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

3. Conéctese al nodo 2 y cambie al directorio de scripts de Tableau Server:

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

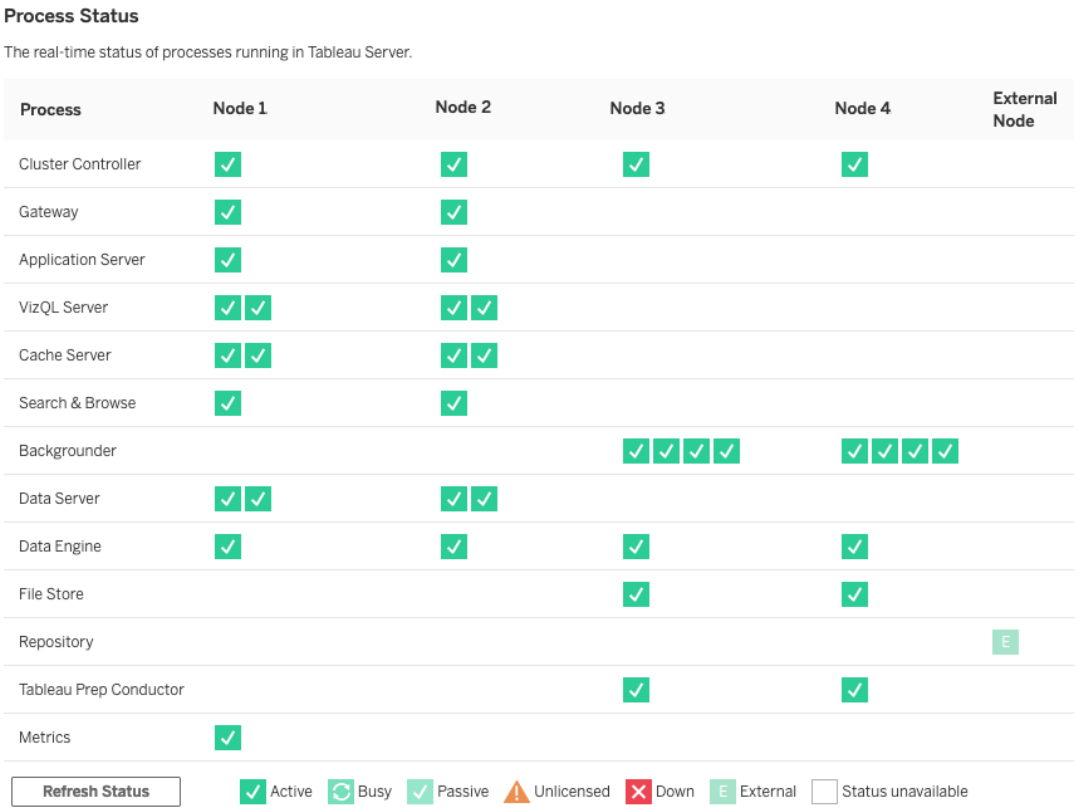
4. Ejecute el comando `initialize-tsm` y haga referencia al archivo de arranque:

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

5. Después de que `initialize-tsm` haya terminado, elimine `boot.json` y luego salga o cierre la sesión.

Configurar procesos

Debe configurar el clúster de Tableau Server en el nodo donde se ejecuta Tableau Server Administration Controller (controlador de TSM). El controlador de TSM se ejecuta en el nodo inicial.



Configurar el nodo 2

1. Una vez que haya inicializado TSM con el archivo de arranque en el nodo 2, inicie sesión en el nodo inicial.

2. En el nodo inicial (`node1`), ejecute los siguientes comandos para configurar procesos en el nodo 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
```

```
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 está instalando la versión 2022.1 o posterior, agregue también el servicio de Index and Search:

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

Si está instalando la versión 2023.3 o posterior, incluya solo el servicio de Index and Search. No agregue el servicio Buscar y examinar (searchserver)

3. Revise la configuración antes de aplicarla. Ejecute el comando siguiente:

   ```
   tsm pending-changes list
   ```

4. Una vez que haya comprobado que sus cambios están en la lista pendiente (también habrá otros servicios en la lista pendiente), aplique los cambios:

   ```
   tsm pending-changes apply
   ```

   Los cambios requieren un reinicio. La configuración y el reinicio tardarán algún tiempo.

5. Verifique la configuración del nodo 2. Ejecute el comando siguiente:

   ```
   tsm status -v
   ```

Configurar el nodo 3

Inicialice TSM con el proceso de arranque en el nodo 3 y luego ejecute los comandos `tsm topology set-process` que aparecen a continuación.

Una advertencia del Servicio de coordinación se mostrará cada vez que establezca un proceso. Puede ignorar esta advertencia mientras configura los procesos.

1. Después de inicializar TSM con el archivo de arranque en el nodo 3, inicie sesión en el nodo inicial (`node1`) y ejecute los siguientes comandos para configurar procesos:

   ```
   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 está instalando la versión 2022.1 o posterior, agregue también el servicio de Index and Search:

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

2. Revise la configuración antes de aplicarla. Ejecute el comando siguiente:

   ```
   tsm pending-changes list
   ```

3. Una vez que haya comprobado que sus cambios están en la lista pendiente (la lista incluirá otros servicios que se configuran automáticamente), aplique los cambios:

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

   Los cambios requieren un reinicio. La configuración y el reinicio tardarán algún tiempo.

4. Compruebe la configuración ejecutando el siguiente comando:

   ```
   tsm status -v
   ```

Implementar el conjunto del servicio de coordinación en los nodos 1-3

Para la implementación de cuatro nodos de arquitectura de referencia estándar, siga este procedimiento:

1. Ejecute los siguientes comandos en el nodo 1:

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

   El proceso incluye un reinicio de TSM, que llevará algún tiempo.

2. Una vez implementado el servicio de coordinación, inicie TSM:

   ```
   tsm start
   ```

Realizar copias de seguridad del archivo .tar del paso 3

Una vez que haya verificado la instalación, realice cuatro copias de seguridad del archivo .tar:

- PostgreSQL
- Nodo inicial de Tableau (nodo 1)
- Nodo 2 de Tableau
- Nodo 3 de Tableau

Crear archivos .tar del paso 3

1. En el nodo inicial de Tableau, detenga Tableau:

   ```
   tsm stop
   ```

2. Una vez que TSM se haya detenido, detenga los servicios administrativos de Tableau en cada nodo. Ejecute el siguiente comando en cada nodo, en orden (Nodo 1, Nodo 2 y luego Nodo 3):

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

3. En el host de PostgreSQL, detenga la instancia de la base de datos de Postgres:

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

4. Ejecute el siguiente comando para crear la copia de seguridad del archivo .tar:

   ```
   sudo su
   cd /var/lib/pgsql
   tar -cvf step3.12.bkp.tar 12
   exit
   ```

5. Compruebe que el archivo .tar de Postgres se crea con permisos raíz:

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

6. En el host de Postgres, inicie la base de datos de Postgres:

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

7. Cree la copia de seguridad del archivo .tar en el nodo 1, el nodo 2 y el nodo 3. Ejecute los siguientes comandos en cada nodo:

   - ```
     cd /data
     sudo tar -cvf step3.tableau_data.bkp.tar tableau_data
     ```
   - Compruebe que el archivo .tar de Tableau se crea con permisos raíz:

     ```
     ls -al
     ```

8. Inicie los servicios administrativos de Tableau en cada nodo en orden (nodo 1, nodo 2 y nodo 3):
     ```

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

9. Ejecute el comando `tsm status` para supervisar el estado de TSM antes de reiniciar.

   En la mayoría de los casos, el comando devolverá un estado DEGRADED y luego ERROR. Espere unos momentos y vuelva a ejecutar el comando. Si se devuelve el estado ERROR o DEGRADED, siga esperando. No intente iniciar TSM hasta que se devuelva el estado STOPPED. Después, ejecute el siguiente comando:

   ```
   tsm start
   ```

Restaurar paso 3

Este proceso restaura los nodos 1, 2 y 3 de Tableau. También restaura la instancia de Postgres al paso 3. Una vez que haya restaurado este paso, puede implementar el servicio de coordinación, el nodo 4 y luego las configuraciones finales del nodo.

1. Detenga el servicio tsm en el host inicial de Tableau (nodo 1):

   ```
   tsm stop
   ```

2. Una vez que TSM se haya detenido, detenga los servicios administrativos de Tableau en el nodo 1, el nodo 2 y el nodo 3. Ejecute el siguiente comando en cada nodo:

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

3. Restaure el .tar del paso 3 de PostgreSQL. En el equipo que ejecuta Postgres, ejecute los siguientes comandos:

   ```
   sudo su
   systemctl stop postgresql-12
   cd /var/lib/pgsql
   ```

```
tar -xvf step3.12.bkp.tar

systemctl start postgresql-12

exit
```

4. Restaure el archivo .tar del paso 3 de Tableau en el nodo 1, el nodo 2 y el nodo 3. Ejecute los siguientes comandos en cada nodo de Tableau:

   ```
   cd /data

   sudo rm -rf tableau_data

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

5. En el equipo del nodo 1 de Tableau, elimine los siguientes archivos:

   - `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 el shell devuelve un error de "archivo no encontrado", es posible que deba cambiar el nombre de la ruta para incrementar el número `<n>` en esta sección de la ruta: `.../appzookeeper/<n>/version-2/...`.

6. Reinicie los servicios administrativos en el nodo 1, el nodo 2 y el nodo 3. Ejecute los siguientes comandos en cada nodo:

   ```
   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>/./s-
tart-administrative-services
```

7. En el nodo 1, ejecute el comando `tsm status` para supervisar el estado de TSM antes de reiniciar.

   En algunos casos, obtendrá un error `Cannot connect to server...`. Este error se produce porque el servicio tabadmincontroller no se ha reiniciado. Siga ejecutando `tsm status` periódicamente. Si este error no desaparece después de 10 minutos, ejecute el comando `start-administrative-services` de nuevo.

   Después de unos momentos, el comando `tsm status` devolverá un estado de DEGRADED y luego ERROR. No inicie TSM hasta que se devuelva el estado STOPPED. Después, ejecute el siguiente comando:

   ```
   tsm start
   ```

Reanude el proceso de instalación para implementar el servicio de coordinación en los nodos 1-3

Configurar el nodo 4

El proceso para configurar el nodo 4 es el mismo que el del nodo 3.

Configure los mismos procesos que ha establecido para el nodo 3, ejecutando el mismo conjunto de comandos, como se muestra arriba, pero especificando`node4` en los comandos en lugar de`node3`.

Al igual que con la verificación del Nodo 3, verifique la configuración del Nodo 4 ejecutando `tsm status -v`.

Antes de continuar, espere a que el proceso de almacenamiento de archivos en el nodo 4 termine de sincronizarse. El estado del servicio del almacén de archivos devolverá `is synchronizing` hasta que termine. Cuando devuelva el estado `is running`, puede continuar.

Configuración y verificación del proceso final

El último paso para procesar la configuración es eliminar los procesos redundantes del nodo 1.

1. Conéctese al nodo inicial (`node1`).
2. Retire el almacén de archivos del nodo 1. Esto provocará una advertencia sobre la eliminación del almacén de archivos de un controlador de ubicación conjunta. Puede ignorar esta advertencia. Ejecute el comando siguiente:

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

3. Cuando se da de baja el almacén de archivos, ejecute el siguiente comando para eliminar el proceso en segundo plano del Nodo 1:

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

4. Revise la configuración antes de aplicarla. Ejecute el comando siguiente:

   ```
   tsm pending-changes list
   ```

5. Una vez que haya verificado que sus cambios están en la lista de pendientes, aplique los cambios:

   ```
   tsm pending-changes apply
   ```

   Los cambios requieren un reinicio. La configuración y el reinicio tardarán algún tiempo.

6. Compruebe la configuración:

   ```
   tsm status -v.
   ```

   Antes de continuar, espere a que el proceso de almacenamiento de archivos en el nodo 4 termine de sincronizarse. El estado del servicio del almacén de archivos devolverá `is synchronizing` hasta que termine. Cuando devuelva el estado `is running`, puede continuar.

Realizar una copia de seguridad

Una recuperación completa de Tableau Server requiere una cartera de respaldo que incluya tres componentes:

- Un archivo de copia de seguridad del repositorio y los datos del almacén de archivos. Este archivo se genera con el comando `tsm maintenance backup`.
- Un archivo de exportación de configuración y topología. Este archivo se genera con el comando `tsm settings export`.
- Certificados de autenticación, claves y archivos keytab.

Para obtener una descripción completa del proceso de copia de seguridad y restauración, consulte el tema de Tableau Server, *Realizar una copia de seguridad y restauración completa de Tableau Server* (Linux).

En esta etapa de su implementación, todos los archivos y activos relevantes que se requieren para una restauración completa se incluyen ejecutando los comandos `tsm maintenance backup` y `tsm settings export`.

1. Ejecute el siguiente comando para exportar la configuración y la topología a un archivo llamado `ts_settings_backup.json`

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

2. Ejecute el siguiente comando para crear una copia de seguridad del repositorio y los datos del almacén de archivos en un archivo llamado `ts_backup-<yyyy-mm-dd>.tsbak`. Ignore la advertencia acerca de que el almacén de archivos no se encuentra en el nodo del controlador.

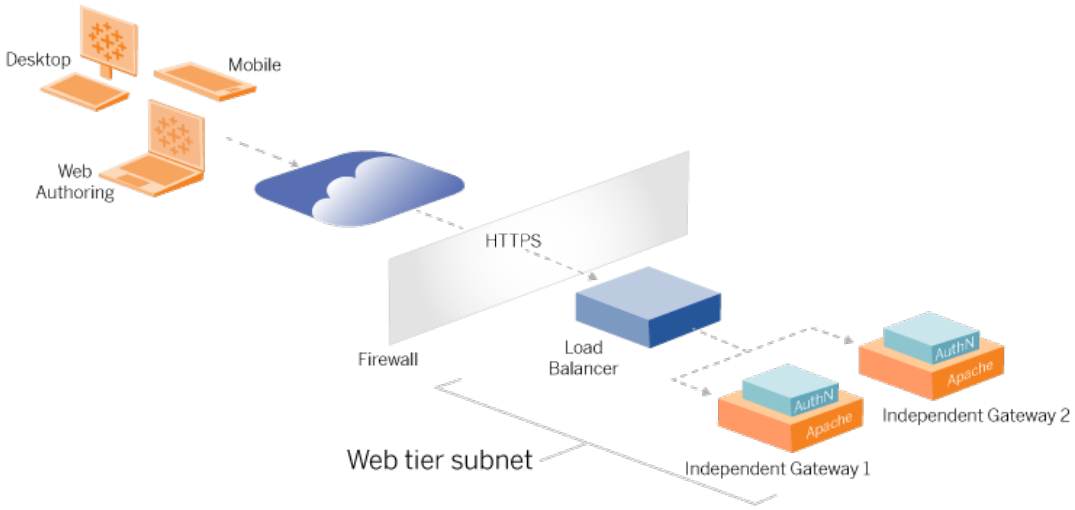
   ```
   tsm maintenance backup -f ts_backup -d --skip-compression
   ```

   Ubicación del archivo de copia de seguridad:

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

3. Copie ambos archivos y guárdelos en un activo de almacenamiento diferente que no sea compartido por su implementación de Tableau Server.

Parte 5: Configuración del nivel web



El nivel web de la arquitectura de referencia debe incluir los siguientes componentes:

- Un equilibrador de carga de aplicaciones orientado a la web que acepta solicitudes HTTPS de clientes de Tableau y se comunica con los servidores proxy inversos.
- Proxy inverso:
  - Recomendamos implementar la puerta de enlace independiente de Tableau Server.
  - Recomendamos un mínimo de dos servidores proxy para la redundancia y para manejar la carga de clientes.
  - Recibe tráfico HTTPS del equilibrador de carga.
  - Admite la sesión fija en el host de Tableau.
  - Configure el proxy para el equilibrio de carga por turnos para cada Tableau Server que ejecute el proceso de Gateway.
  - Maneja solicitudes de autenticación del IdP externo.
- Proxy de reenvío: Tableau Server requiere acceso a Internet para las licencias y la funcionalidad de mapas. Debe configurar listas de admisión de proxys de reenvío para las URL de Tableau Service. Consulte *Comunicación con Internet* (Linux).

- Todo el tráfico relacionado con el cliente se puede cifrar a través de HTTPS:
  - Equilibrador de carga de cliente a aplicación
  - Equilibrador de carga de aplicaciones para servidores proxy inversos
  - Servidor proxy para Tableau Server
  - Controlador de autenticación que se ejecuta en proxy inverso al IdP
  - Tableau Server a IdP

Puerta de enlace independiente de Tableau Server

La versión 2022.1 de Tableau Server incluyó la puerta de enlace independiente de Tableau Server. La puerta de enlace independiente es una instancia independiente del proceso de puerta de enlace de Tableau que funciona como un proxy inverso compatible con Tableau.

La puerta de enlace independiente admite el equilibrio de carga de operación por turnos simple para los servidores back-end de Tableau. Sin embargo, la puerta de enlace independiente no está diseñada para servir como equilibrador de carga de aplicaciones empresariales. Recomendamos ejecutar la puerta de enlace independiente detrás de un equilibrador de carga de aplicaciones de clase empresarial.

La puerta de enlace independiente requiere una licencia de Advanced Management.

Autenticación y autorización

La arquitectura de referencia predeterminada especifica la instalación de Tableau Server con la autenticación local configurada. En este modelo, los clientes deben conectarse a Tableau Server para ser autenticados por el proceso de autenticación local nativo de Tableau Server. No recomendamos usar este método de autenticación en la arquitectura de referencia porque el escenario requiere que los clientes no autenticados se comuniquen con el nivel de la aplicación, lo cual es un riesgo de seguridad.

En su lugar, recomendamos configurar un proveedor de identidades externo de nivel empresarial junto con un módulo de AuthN para autenticar previamente todo el tráfico al nivel de la aplicación. Cuando se configura con un IdP externo, no se utiliza el proceso de autenticación

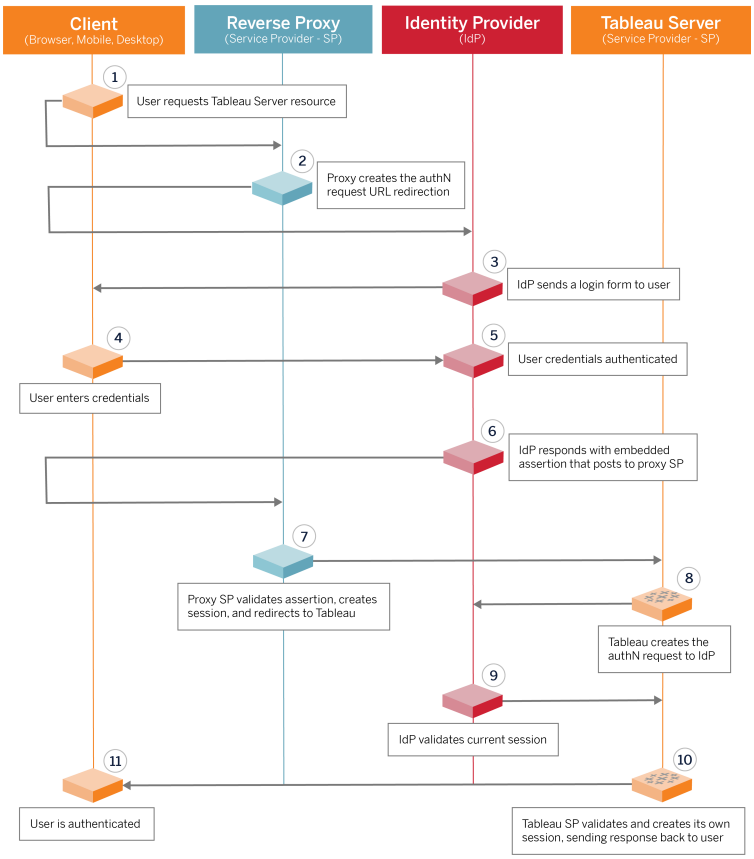
local nativo de Tableau Server. Tableau Server autoriza el acceso a los recursos en la implementación después de que el IdP haya autenticado a los usuarios.

Autenticación previa con un módulo de AuthN

En el ejemplo documentado en esta guía, el SSO de SAML está configurado, pero el proceso de autenticación previa se puede configurar con la mayoría de los proveedores de identidades externos y un módulo de AuthN.

En la arquitectura de referencia, el proxy inverso está configurado para crear una sesión de autenticación de cliente con el IdP antes de enviar esas solicitudes a Tableau Server. Nos referimos a este proceso como la fase de *preautorización.* El proxy inverso solo redirigirá las sesiones de los clientes autenticados a Tableau Server. Luego, Tableau Server creará una sesión, verificará la autenticación de la sesión con el IdP y luego devolverá la solicitud del cliente.

El siguiente diagrama muestra en detalle y paso a paso el proceso de autenticación y autorización previa con un módulo de AuthN configurado. El proxy inverso puede ser una solución genérica de terceros o la puerta de enlace independiente de Tableau Server:



Descripción general de la configuración

Esta es una descripción general del proceso para configurar el nivel web. Verifique la conectividad después de cada paso:

1. Configure dos proxies inversos para proporcionar acceso HTTP a Tableau Server.
2. Configure la lógica de equilibrio de carga con sesiones fijas en servidores proxy para conectarse a cada instancia de Tableau Server que ejecuta el proceso de la puerta de enlace.
3. Configure el equilibrio de carga de la aplicación con sesiones pegajosas en la puerta de enlace de Internet para reenviar solicitudes a los servidores proxy inversos.
4. Configure la autenticación con un IdP externo. Puede configurar SSO o SAML instalando un controlador de autenticación en los servidores proxy inversos. El módulo de AuthN administra el protocolo de enlace de autenticación entre el IdP externo y su

implementación de Tableau. Tableau también actuará como proveedor de servicios de IdP y autenticará a los usuarios con el IdP.

5. Para autenticarse con Tableau Desktop en esta implementación, sus clientes deben ejecutar Tableau Desktop 2021.2.1 o posterior.

Ejemplo de configuración de nivel web con la puerta de enlace independiente de Tableau Server

El resto de este tema proporciona un procedimiento de un extremo a otro que describe cómo implementar el nivel web en el ejemplo de la arquitectura de referencia de AWS usando la puerta de enlace independiente de Tableau Server. Para ver una configuración de ejemplo con Apache como proxy inverso, consulte el Apéndice: nivel web con implementación de ejemplo de Apache .

La configuración de ejemplo tiene los siguientes componentes:

- Equilibrador de carga de aplicaciones de AWS
- Puerta de enlace independiente de Tableau Server
- Módulo de autenticación Mellon
- IdP de Okta
- Autenticación SAML

Nota: El ejemplo de configuración de nivel web que se presenta en esta sección incluye procedimientos detallados para implementar software y servicios de terceros. Hemos hecho todo lo posible por verificar y documentar los procedimientos para habilitar el escenario del nivel web. Sin embargo, el software de terceros puede cambiar o su escenario puede diferir de la arquitectura de referencia que se describe aquí. Consulte la documentación de terceros para obtener detalles de configuración autorizados y asistencia.

Los ejemplos de Linux a lo largo de esta sección muestran comandos para distribuciones similares a RHEL. Específicamente, los comandos aquí se han desarrollado con la distribución de Amazon Linux 2. Si está ejecutando distribuciones de Ubuntu, edite los comandos según corresponda.

La implementación del nivel web en este ejemplo sigue un procedimiento de verificación y configuración paso a paso. La configuración del nivel web principal consta de los siguientes pasos para habilitar HTTP entre Tableau e Internet. La puerta de enlace independiente se ejecuta y configura para proxy inverso/equilibrio de carga detrás del equilibrador de carga de la aplicación AWS:

1. Preparar el entorno
2. Instalar la puerta de enlace independiente
3. Configurar el servidor de la puerta de enlace independiente
4. Configurar el equilibrador de carga de aplicaciones de AWS

Una vez que se configura el nivel web y se verifica la conectividad con Tableau, debe configurar la autenticación con un proveedor externo.

Preparar entorno

Complete las siguientes tareas antes de implementar la puerta de enlace independiente.

1. Cambios en el grupo de seguridad de AWS. Configure el grupo de seguridad pública para permitir el tráfico interno de entrada de la puerta de enlace independiente (TCP 21319) desde el grupo de seguridad privada.

2. Instale la versión 22.1.1 (o posterior) en un clúster de Tableau Server de cuatro nodos como se documenta en la Paso 4: Instalar y configurar Tableau Server.

3. Configure las dos instancias EC2 de proxy en el grupo de seguridad Público como se documenta en Configurar equipos host.

Instalar la puerta de enlace independiente

La puerta de enlace independiente de Tableau Server requiere una licencia de Advanced Management.

La implementación de la puerta de enlace independiente de Tableau Server consiste en instalar y ejecutar el paquete .rpm y luego configurar el estado inicial. El procedimiento incluido en esta guía proporciona orientación prescriptiva para la implementación en la arquitectura de referencia.

Si su implementación difiere de la arquitectura de referencia, consulte la documentación principal de Tableau Server, *Instalar Tableau Server con la puerta de enlace independiente* (Linux).

Importante: La configuración de la puerta de enlace independiente puede ser un proceso propenso a errores. Es muy difícil solucionar problemas de configuración en dos instancias de servidores de puerta de enlace independientes. Por este motivo, recomendamos configurar un servidor de puerta de enlace independiente a la vez. Después de configurar el primer servidor y verificar la funcionalidad, debe configurar el segundo servidor de puerta de enlace independiente.

Aunque configurará cada servidor de puerta de enlace independiente por separado, ejecute este procedimiento de instalación en ambas instancias EC2 que instaló en el grupo de seguridad público:

1. Ejecute la actualización para aplicar las últimas correcciones al sistema operativo Linux:

   ```
   sudo yum update
   ```

2. Si Apache está instalado, elimínelo:

   ```
   sudo yum remove httpd
   ```

3. Copie el paquete de instalación de la puerta de enlace independiente versión 2022.1.1 (o posterior) de la página de descargas de Tableau en el equipo host que ejecutará Tableau Server.

   Por ejemplo, en un equipo con el sistema operativo tipo RHEL de Linux, ejecute:

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

4. Ejecute el programa de instalación. Por ejemplo, en un sistema operativo tipo RHEL de Linux, ejecute:

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

5. Cambie al directorio `/opt/tableau/tableau_tsi-g/packages/scripts.<version_code>/` y ejecute el script `initialize-tsig` que aparece allí: Además de la marca `--accepteula`, debe incluir el rango de IP de las subredes donde se ejecuta la implementación de Tableau Server. Utilice la opción `-c` para especificar el rango de IP. El siguiente ejemplo muestra el comando con las subredes de AWS de ejemplo especificadas:

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

6. Una vez finalizada la inicialización, abra el archivo `tsighk-auth.conf` y copie el secreto de autenticación en el archivo. Deberá enviar este código para cada instancia de puerta de enlace independiente como parte de la configuración de Tableau Server de back-end:

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

7. Después de ejecutar los pasos anteriores en ambas instancias de Independent Gateway, prepare el archivo de configuración `tsig.json`. El archivo de configuración consta de una matriz "independentGateways". La matriz contiene objetos de

configuración que definen detalles de conexión para una instancia de puerta de enlace independiente.

Copie el siguiente JSON y personalícelo según su entorno de implementación. El ejemplo aquí muestra un archivo para una arquitectura de referencia de AWS de muestra.

El archivo JSON de ejemplo a continuación solo incluye información de conexión para una puerta de enlace independiente. Más adelante en el proceso, incluirá la información de conexión para el segundo servidor de puerta de enlace independiente.

Guardar el archivo como `tsig.json` para los trámites que siguen.

```
{
"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"`: el nombre de DNS privado de la instancia AWS EC2 que ejecuta Independent Gateway.
- `"host"`: igual que `"id"`.
- `"port"`: El puerto de limpieza, por defecto, `"21319"`.
- `"protocol"`: el protocolo para el tráfico de clientes. Deje esto como `http` para la configuración inicial.
- `"authsecret"`: el secreto que copió en el paso anterior.

Puerta de enlace independiente: conexión directa vs. retransmisión

Antes de continuar, debe decidir qué esquema de conexión configurar en su implementación: conexión directa o de retransmisión. Cada opción se describe brevemente aquí, junto con los puntos de datos de decisión relevantes.

Conexión de retransmisión: Puede configurar la puerta de enlace independiente para retransmitir la comunicación del cliente a través de un solo puerto al proceso de la puerta de enlace en Tableau Server. Nos referimos a esta comunicación como conexión *de retransmisión*:

- El proceso de retransmisión genera un salto adicional desde la puerta de enlace independiente hasta el proceso de puerta de enlace de Tableau Server de back-end. El salto adicional degrada el rendimiento en comparación con la configuración de conexión directa.
- TLS es compatible con el modo de retransmisión. Toda la comunicación en modo de retransmisión está restringida a un solo protocolo (HTTP o HTTPS) y, por lo tanto, puede cifrarse y autenticarse con TLS.

Conexión directa: La puerta de enlace independiente puede comunicarse directamente con los procesos backend de Tableau Server a través de varios puertos. Nos referimos a esta comunicación como conexión *directa*:

- Debido a que la conexión es directa al back-end de Tableau Server, el rendimiento del cliente mejora notablemente en comparación con la opción de conexión de retransmisión.
- Requiere abrir más de 16 puertos de subredes públicas a privadas para la comunicación de procesos directos desde la puerta de enlace independiente a equipos de Tableau Server.
- TLS aún no es compatible con los procesos de la puerta de enlace independiente a Tableau Server.

Configurar conexiones de retransmisión

Para ejecutar TLS entre Tableau Server y la puerta de enlace independiente, debe configurar con una conexión de retransmisión. Los escenarios de ejemplo en el EDG están configurados con conexión de relé.

1. Copie `tsig.json` al nodo 1 de su implementación de Tableau Server.
2. En el nodo 1, ejecute los siguientes comandos para habilitar la puerta de enlace independiente.

```
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
```

Configurar conexiones directas

Dado que la conexión directa no es compatible con TLS, recomendamos configurar la conexión directa solo si puede asegurar todo el tráfico de la red por otros medios. Para ejecutar TLS entre Tableau Server y la puerta de enlace independiente, debe configurar con una conexión de retransmisión. Los escenarios de ejemplo en el EDG están configurados con conexión de relé.

Si está configurando una puerta de enlace independiente para la conexión directa a Tableau Server, debe habilitar la configuración para activar la comunicación. Después de que Tableau Server se comunique con la puerta de enlace independiente, se establecerán los objetivos del protocolo. A continuación, debe recuperar el `proxy_targets.csv` desde el equipo de la puerta de enlace independiente y abrir los puertos correspondientes de los grupos de seguridad públicos a privados en AWS.

1. Copie `tsig.json` al nodo 1 de su implementación de Tableau Server.

2. En el nodo 1, ejecute los siguientes comandos para habilitar la puerta de enlace independiente.

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

3. En el equipo de la puerta de enlace independiente, ejecute el siguiente comando para ver los puertos que usa el clúster de Tableau Server:

   ```
   less /var/opt/tableau/tableau_tsig/config/httpd/proxy_targets.csv
   ```

4. Configurar grupos de seguridad de AWS. Agregue los puertos TCP enumerados en `proxy_targets.csv` para permitir la comunicación del grupo de seguridad pública al grupo de seguridad privada.

   Recomendamos automatizar la configuración de entrada de puertos, ya que los puertos pueden cambiar si cambia la implementación de Tableau Server. Añadir nodos o reconfigurar procesos en la implementación de Tableau Server activará cambios en el acceso al puerto requerido por la puerta de enlace independiente.

Verificación: configuración de topología base

Debería poder acceder a la página de administración de Tableau Server desde `http://<gateway-public-IP-address>`.

Si la página de inicio de sesión de Tableau Server no se carga o si Tableau Server no se inicia, siga estos pasos de solución de problemas:

Red:

- Verifique la conectividad entre la implementación de Tableau y la instancia de puerta de enlace independiente ejecutando el siguiente comando `wget` del nodo 1 de Tableau Server: wget http://<dirección IP interna de la puerta de enlace independiente>:21319,

por ejemplo:

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

Si la conexión se rechaza o falla, verifique que el grupo de seguridad pública esté configurado para permitir el tráfico de limpieza de puerta de enlace independiente (TCP 21319) desde el grupo de seguridad privada.

Si el grupo de seguridad está configurado correctamente, verifique que haya especificado las direcciones IP o los rangos de IP correctos durante la inicialización de la puerta de enlace independiente. Puede ver y cambiar esta configuración en el archivo `environment.bash` ubicado en `/etc/opt/tableau/tableau_tsig/environment.bash`. Si realiza un cambio en este archivo, reinicie el servicio tsig-http como se describe a continuación.

En el host del Proxy 1:

1. Sobrescriba el archivo httpd.conf con el archivo auxiliar de la puerta de enlace independiente:

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

2. Reinicie tsig-httpd como primer paso de solución de problemas:

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

En el nodo 1 de Tableau

- Compruebe el archivo `tsig.json`. Si encuentra errores, corríjalos y luego ejecute `tsm topology external-services gateway update -c tsig.json`.
- Si ejecuta una conexión directa, verifique que los puertos TCP enumerados en `proxy_targets.csv` se configuran como puertos de entrada de grupos de seguridad públicos a privados.

Configurar el equilibrador de carga de aplicaciones de AWS

Configure el equilibrador de carga como una escucha HTTP. Este procedimiento describe cómo agregar un equilibrador de carga en AWS.

Paso 1: crear un grupo de destino

Un grupo de destino es una configuración de AWS que define las instancias EC2 que ejecutan sus servidores proxy. Estos son los destinos del tráfico de LBS.

1. EC2>**Grupos de destino** > **Crear grupo de destino**

2. En la página Crear:

   - Escriba un nombre de grupo de destino, `TG-internal-HTTP` por ejemplo
   - Tipo de destino: instancias
   - Protocolo: HTTP
   - Puerto: 80
   - VPC: seleccione su VPC
   - En **Comprobaciones de estado** > **Configuración de comprobaciones de estado avanzadas** > **Códigos de éxito**, agregue la lista de códigos para leer: `200,303`.
   - Haga clic en **Crear**

3. Seleccione el grupo de destino que acaba de crear y luego haga clic en la pestaña **Destinos**:

   - Haga clic en **Editar**.
   - Seleccione las instancias EC2 (o instancia única si está configurando una a la vez) que ejecutan la aplicación de proxy y luego haga clic en **Agregar a registrado**.
   - Haga clic en **Guardar**.

Paso 2: iniciar el asistente del equilibrador de carga

1. EC2> **Equilibradores de carga** > **Crear equilibrador de carga**
2. En la página "Seleccionar tipo de equilibrador de carga", cree un equilibrador de carga de aplicaciones.

Nota: La interfaz de usuario que se muestra para configurar el equilibrador de carga no es coherente en los centros de datos de AWS. El procedimiento siguiente, "Configuración del asistente", se asigna al asistente de configuración de AWS que comienza con el **Paso 1: configurar el equilibrador de carga**.

Si su centro de datos muestra todas las configuraciones en una sola página que incluye un botón **Crear equilibrador de carga** en la parte inferior de la página, siga el procedimiento de "Configuración de una sola página" a continuación.

Configuración del asistente

1. Página **configurar equilibrador de carga**:
    - Especifique el nombre
    - Esquema: orientado a Internet (predeterminado)
    - Tipo de dirección IP: ipv4 (predeterminado)
    - Oyentes (oyentes y enrutamiento):
        a. Deje el oyente HTTP predeterminado
        b. Haga clic en **Agregar oyente** y agregue `HTTPS:443`
    - VPC: seleccione la VPC donde instaló todo
    - Zonas de disponibilidad:
        - Seleccione **a** y **b** para las regiones de su centro de datos
        - En cada selector del menú desplegable correspondiente, seleccione la subred pública (donde residen sus servidores proxy).
    - Haga clic en **Establecer configuración de seguridad**
2. Página **Configurar ajustes de seguridad**

- Suba su certificado SSL público.
- Haga clic en **Siguiente: Configurar grupos de seguridad**.

3. Página **Configurar grupos de seguridad**:

   - Seleccione el grupo de seguridad público. Si se selecciona el grupo de seguridad predeterminado, borre esa selección.
   - Haga clic en **Siguiente: Configurar el enrutamiento**.

4. Página **Configurar enrutamiento**

   - Grupo de destino: Grupo de destino existente.
   - Nombre: seleccione el grupo de destino que creó anteriormente
   - Haga clic en **Siguiente: Registrar destinos**.

5. Página **Registrar destinos**

   - Deben mostrarse las dos instancias de servidor proxy que configuró anteriormente.
   - Haga clic en **Siguiente: Revisión**.

6. Página **Revisión**

   Haga clic en **Crear**.

Configuración de una sola página

Configuración básica

- Especifique el nombre
- Esquema: orientado a Internet (predeterminado)
- Tipo de dirección IP: ipv4 (predeterminado)

Mapeo de redes

- VPC: seleccione la VPC donde instaló todo
- Mapeo:
    - Seleccione las zonas de disponibilidad **a** y **b** (o comparables) para sus regiones de centros de datos
    - En cada selector del menú desplegable correspondiente, seleccione la subred pública (donde residen sus servidores proxy).

Grupos de seguridad

Seleccione el grupo de seguridad público. Si se selecciona el grupo de seguridad predeterminado, borre esa selección.

Oyentes y enrutamiento

- Deje el oyente HTTP predeterminado. Para la **acción predeterminada**, especifique el grupo de destino que configuró anteriormente.
- Haga clic en **Agregar oyente** y agregue `HTTPS:443`. Para la **acción predeterminada**, especifique el grupo de destino que configuró anteriormente.

Configuración de oyente seguro

- Suba su certificado SSL público.

Haga clic en **Crear el equilibrador de carga**.

Paso 3: habilitar la adherencia

1. Una vez creado el equilibrador de carga, debe habilitar la adherencia en el grupo de destino.
    - Abra la página del grupo de destino de AWS (**EC2**> **Equilibradores de carga** > **Grupos de destino**), seleccione la instancia del grupo de destino que acaba de configurar. En el menú **Acciones**, seleccione **Editar atributos**.
    - En la página **Editar atributos**, seleccione **Adherencia**, especifique una duración de `1 day` y luego **Guardar cambios**.
2. En el equilibrador de carga, habilite la adherencia en el oyente HTTP. Seleccione el equilibrador de carga que acaba de configurar y luego haga clic en la pestaña **Oyentes**:

- Para **HTTP:80**, haga clic en **Ver/editar reglas**. En la página **Reglas** resultante, haga clic en el icono de edición (una vez en la parte superior de la página y luego nuevamente por la regla) para editar la regla. Elimine la regla THEN existente y reemplácela haciendo clic en **Agregar acción** > **Reenviar a...** En la configuración THEN resultante, especifique el mismo grupo de destino que ha creado. En Adherencia a nivel de grupo, habilite la adherencia y establezca la duración en 1 día. Guarde la configuración y haga clic en **Actualizar**.

Paso 4: Establezca el tiempo de espera inactivo en el equilibrador de carga

En el equilibrador de carga, actualice el tiempo de espera inactivo a 400 segundos.

Seleccione el equilibrador de carga que ha configurado para esta implementación y luego haga clic en **Acciones** > **Editar atributos**. Establezca el tiempo de **espera inactivo** en `400` segundos y luego haga clic en **Guardar**.

Paso 5: Verificar la conectividad LBS

Abra la página del equilibrador de carga de AWS (**EC2** > **Equilibradores de carga**), seleccione la instancia del equilibrador de carga que acaba de configurar.

En **Descripción**, copie el nombre del DNS y péguelo en un navegador para acceder a la página de inicio de sesión de Tableau Server.

Si aparece el error 500, probablemente necesite reiniciar sus servidores proxy.

Actualizar DNS con URL pública de Tableau

Utilice el nombre de la zona DNS de su dominio de la descripción del equilibrador de carga de AWS para crear un valor CNAME en su DNS. El tráfico a su URL (tableau.example.com) debe enviarse al nombre de DNS público de AWS.

Verificar la conectividad

Una vez finalizadas las actualizaciones de DNS, debería poder navegar a la página de inicio de sesión de Tableau Server con su URL pública, por ejemplo, `http-s://tableau.example.com`.

Ejemplo de configuración de autenticación: SAML con IdP externo

El siguiente ejemplo describe cómo instalar y configurar SAML con Okta IdP y el módulo de autenticación Mellon para una implementación de Tableau que se ejecuta en la arquitectura de referencia de AWS.

Este ejemplo se basa en la sección anterior y supone que está configurando una puerta de enlace independiente a la vez.

El ejemplo describe cómo configurar Tableau Server y la puerta de enlace independiente para usar HTTP. Okta enviará la solicitud al equilibrador de carga de AWS a través de HTTPS, pero todo el tráfico interno viajará a través de HTTP. Mientras configura para este escenario, tenga en cuenta los protocolos HTTP y HTTPS al configurar cadenas de URL.

Este ejemplo utiliza Mellon como un módulo de proveedor de servicios de autenticación previa en los servidores de la puerta de enlace independiente. Esta configuración garantiza que solo el tráfico autenticado se conecte a Tableau Server, que también actúa como proveedor de servicios con Okta IdP. Por lo tanto, debe configurar dos aplicaciones IdP: una para el proveedor de servicios Mellon y otra para el proveedor de servicios Tableau.

Crear la cuenta de administrador de Tableau

Un error común al configurar SAML es olvidarse de crear una cuenta de administrador en Tableau Server antes de habilitar SSO.

El primer paso es crear una cuenta en Tableau Server con un rol de administrador del servidor. En el caso de ejemplo de Okta, el nombre de usuario debe tener el formato de una dirección de correo electrónico válida, por ejemplo, usuario@ejemplo.com. Debe establecer una contraseña para este usuario, pero la contraseña no se utilizará después de configurar SAML.

Configurar la aplicación de autorización previa de Okta

El escenario de un extremo a otro descrito en esta sección requiere dos aplicaciones Okta:

- Solicitud de autorización previa de Okta
- Aplicación Okta de Tableau Server

Cada una de estas aplicaciones está asociada con diferentes metadatos que deberá configurar en el proxy inverso y Tableau Server, respectivamente.

Este procedimiento describe cómo crear y configurar la aplicación de autorización previa de Okta. Más adelante en este tema, creará la aplicación Okta de Tableau Server. Para obtener una cuenta de Okta de prueba gratuita con usuarios limitados, consulte la página web de desarrolladores de Okta.

Cree una integración de la aplicación SAML para el proveedor de servicios de autenticación previa de Mellon.

1. Abra el panel de administración de Okta > **Aplicaciones** > **Crear integración de aplicaciones**.
2. En la página **Crear una nueva integración de aplicaciones**, seleccione **SAML 2.0** y luego haga clic en **Siguiente**.
3. En la pestaña **Configuración general**, escriba un nombre de aplicación, como `Tableau Pre-Auth` y haga clic en **Siguiente**.
4. En la pestaña **Configurar SAML**:
    - URL de inicio de sesión único (SSO). El elemento final de la ruta en la URL de inicio de sesión único se denomina `MellonEndpointPath` en el archivo de

configuración `mellon.conf` que aparece más adelante en este procedimiento. Puede especificar cualquier punto final que desee. En este ejemplo, `sso` es el punto final. El último elemento, `postResponse`, es obligatorio:`https://tableau.example.com/sso/postResponse`.

- Desmarque la casilla de verificación: **Usar esto para la URL del destinatario y la URL de destino**.
- URL del destinatario: igual que la URL de SSO, pero con HTTP. Por ejemplo, `http://tableau.example.com/sso/postResponse`.
- URL de destino: igual que la URL del SSO, pero con HTTP. Por ejemplo, `http://tableau.example.com/sso/postResponse`.
- URI de audiencia (ID de entidad SP). Por ejemplo, `https://tableau.example.com`.
- Formato del ID de nombre: `EmailAddress`
- Nombre de usuario de la aplicación: `Email`
- Declaraciones de atributos: Nombre = `mail`; Formato de nombre = `Unspecified`; Valor = `user.email`.

Haga clic en **Siguiente**.

5. En la pestaña **Comentarios**, seleccione:
   - **Soy un cliente de Okta que agrega una aplicación interna**
   - **Esta es una aplicación interna que hemos creado**
   - Haga clic en **Finalizar**.

6. Cree el archivo de metadatos de IdP previo a la autenticación:
   - En Okta: **Aplicaciones** > **Aplicaciones** > Su nueva aplicación (p. ej., `Tableau Pre-Auth`)> **Iniciar sesión**
   - Junto a **Certificados de firma de SAML**, haga clic en **Ver instrucciones de configuración de SAML**.
   - En la página **Cómo configurar SAML 2.0 para la aplicación <pre-auth>**, desplácese hacia abajo hasta la sección **Opcional**, **Proporcionar los siguientes metadatos de IDP a su proveedor de SP** .
   - Copie el contenido del campo XML y guárdelo en un archivo llamado `pre-auth_idp_metadata.xml`.

7. (Opcional) Configure la autenticación multifactor:

- En Okta: **Aplicaciones** > **Aplicaciones** > Su nueva aplicación (p. ej., `Tableau Pre-Auth`)> **Iniciar sesión**
- **En Directiva de inicio de sesión**, haga clic en **Agregar regla**.
- En la **regla de inicio de sesión de la aplicación**, especifique un nombre y las diferentes opciones de MFA. Para probar la funcionalidad, puede dejar todas las opciones predeterminadas. Sin embargo, en **Acciones**, debe seleccionar **Solicitar factor** y luego especificar la frecuencia con la que los usuarios deben iniciar sesión. Haga clic en **Guardar**.

Crear y asignar un usuario de Okta

1. En Okta, cree un usuario con el mismo nombre de usuario que creó en Tableau (usuario@ejemplo.com): **Directorio** > **Personas** > **Agregar persona**.
2. Una vez creado el usuario, asigne la nueva aplicación Okta a esa persona: haga clic en el nombre de usuario y luego asigne la aplicación en **Asignar aplicación**.

Instalar Mellon para preautorización

Este ejemplo usa mod_auth_mellon, un popular módulo de código abierto. Algunas distribuciones de Linux empaquetan versiones obsoletas de mod_auth_mellon de un repositorio anterior. Esas versiones obsoletas pueden contener vulnerabilidades de seguridad desconocidas o problemas funcionales. Si elige usar mod_auth_mellon, verifique que esté usando una versión actual.

El módulo mod_auth_mellon es un software de terceros. Hemos hecho todo lo posible por verificar y documentar los procedimientos para habilitar este caso. Sin embargo, el software de terceros puede cambiar o su escenario puede diferir de la arquitectura de referencia que se describe aquí. Consulte la documentación de terceros para obtener detalles de configuración autorizados y asistencia.

1. En la instancia activa de EC2 que ejecuta la puerta de enlace independiente, instale una versión actual del módulo de autenticación Mellon.

2. Cree el directorio `/etc/mellon`:

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

Configurar Mellon como módulo de preautorización

Ejecute este procedimiento en la primera instancia de la puerta de enlace independiente.

Debe tener una copia de `pre-auth_idp_metadata.xml` que creó a partir de la configuración de Okta.

1. Cambie el directorio:

   ```
   cd /etc/mellon
   ```

2. Cree los metadatos del proveedor de servicios. Ejecute el script `mellon_create_metadata.sh`. Debe incluir el ID de entidad y la URL de retorno de su organización en el comando.

   La URL de retorno se denomina *URL de inicio de sesión único* en Okta. El elemento final de la ruta en la URL de retorno se denomina `MellonEndpointPath` en el archivo de configuración `mellon.conf` que aparece más adelante en este procedimiento. En este ejemplo, especificamos `sso` como la ruta del punto final.

   Por ejemplo:

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

   El script devuelve el certificado del proveedor de servicios, la clave y los archivos de metadatos.

3. Cambie el nombre de los archivos del proveedor de servicios en el directorio `mellon` para facilitar la lectura. Nos referiremos a estos archivos por los siguientes nombres en la documentación:

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

4. Copie el archivo `pre-auth_idp_metadata.xml` en el mismo directorio.

5. Cambiar la propiedad y los permisos de todos los archivos en el directorio `/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. Cree el directorio `/etc/mellon/conf.d`:

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

7. Cree el archivo `global.conf` en el directorio `/etc/mellon/conf.d`.

   Copie el contenido del archivo como se muestra a continuación, pero actualice `Mello-nCookieDomain` con su nombre de dominio raíz. Por ejemplo, si el nombre de dominio de Tableau es `tableau.example.com`, introduzca `example.com` para el dominio raíz.

   ```
   <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. Cree el archivo `mellonmod.conf` en el directorio `/etc/mellon/conf.d`.

   Este archivo contiene una única directiva que especifica la ubicación del expediente `mod_auth_mellon.so`. La ubicación en el ejemplo aquí es la ubicación predeterminada del archivo. Verifique que el archivo esté en esta ubicación o cambie la ruta en esta directiva para que coincida con la ubicación real de `mod_auth_mellon.so`:

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

Crear la aplicación de Tableau Server en Okta

1. En el panel de Okta: **Aplicaciones** > **Aplicaciones** > **Examinar catálogo de aplicaciones**
2. En **Examinar catálogo de integración de aplicaciones**, busque `Tableau`, seleccione la miniatura de Tableau Server y luego haga clic en **Agregar**.
3. En **Agregar Tableau Server** > **Configuración general**, especifique una Etiqueta y luego haga clic en **Siguiente**.
4. En Opciones de inicio de sesión, seleccione **SAML 2.0** y,después, desplácese hacia abajo hasta Configuración avanzada de inicio de sesión:
   - **ID de entidad SAML**: especifique la URL pública, por ejemplo, https://tableau.example.com.
   - **Formato de nombre de usuario de la aplicación**: correo electrónico
5. Haga clic en el enlace de **metadatos del proveedor de identidad** para iniciar un navegador. Copie el enlace del navegador. Este es el enlace que utilizará cuando configure Tableau en el procedimiento que se muestra a continuación.
6. Haga clic en **Realizado**.
7. Asigne la nueva aplicación Okta de Tableau Server a su usuario (usuario@example.com): haga clic en el nombre de usuario y luego asigne la aplicación en **Asignar aplicación**.

Establecer la configuración del módulo de autenticación en Tableau Server

Ejecute los siguientes comandos en el nodo 1 de Tableau Server. Estos comandos especifican las ubicaciones de los archivos de configuración de Mellon en el equipo remoto de la puerta de enlace independiente. Vuelva a verificar que las rutas de archivo especificadas en estos comandos se correspondan con las rutas y la ubicación del archivo en el equipo remoto de la puerta de enlace independiente.

```
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
```

Para reducir el tiempo de inactividad, no aplique cambios hasta que haya habilitado SAML como se describe en la siguiente sección.

Habilitar SAML en Tableau Server para IdP

Ejecute este procedimiento en el nodo 1 de Tableau Server.

1. Descargue los metadatos de la aplicación de Tableau Server de Okta. Utilice el enlace que guardó del procedimiento anterior:

   ```
   wget https://dev-66144217.o-
   kta.com/app/exk1egxgt1fhjkSeS5d7/sso/saml/metadata -O idp_meta-
   data.xml
   ```

2. Copie un certificado TLS y un archivo de clave relacionado en Tableau Server. El archivo de claves debe ser una clave RSA. Para obtener más información sobre la configuración y los requisitos de SAML, consulte *Requisitos de SAML* (Linux).

   Para simplificar la administración y la implementación de certificados, y como práctica recomendada de seguridad, recomendamos usar certificados generados por una

autoridad de certificación (CA) de terceros de confianza. Como alternativa, puede generar certificados autofirmados o usar certificados de una PKI para TLS.

Si no tiene un certificado TLS, puede generar un certificado autofirmado mediante el procedimiento integrado a continuación.

Generar un certificado autofirmado

Ejecute este procedimiento en el nodo 1 de Tableau Server.

a. Genere la clave de la autoridad certificadora (CA) raíz de firma:

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

b. Cree el certificado de CA raíz:

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

Se le pedirá que especifique los valores para los campos del certificado. Por ejemplo:

```
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)
[]:tableau.example.com
Email Address []:example@tableau.com
```

c. Cree el certificado y la clave relacionada (`server-saml.csr` y `server-sam-l.key` en el ejemplo siguiente). El nombre del sujeto del certificado debe

coincidir con el nombre del host público del host de Tableau. El nombre del sujeto se establece con la opción `-subj` con el formato `"/CN=<host-name>"`, por ejemplo:

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

d. Firme el nuevo certificado con el certificado CA que creó anteriormente. El siguiente comando también genera el certificado en el formato `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. Convierta el archivo de clave a RSA. Tableau requiere un archivo de clave RSA para SAML. Para convertirlo, ejecute el siguiente comando:

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

3. Configure SAML: Ejecute el siguiente comando, especificando su ID de entidad y URL de retorno, y las rutas al archivo de metadatos, archivo de certificado y archivo de clave:

```
tsm authentication saml configure --idp-entity-id "http-
s://tableau.example.com" --idp-return-url "http-
s://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 su organización ejecuta Tableau Desktop 2021.4 o posterior, debe ejecutar el siguiente comando para habilitar la autenticación a través de los servidores proxy inversos.

Las versiones de Tableau Desktop 2021.2.1 - 2021.3 funcionarán sin ejecutar este comando, siempre que su módulo de autenticación previa (p. ej., Mellon) esté configurado para permitir la conservación de cookies de dominio de nivel superior.

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

5. Aplique los cambios de configuración:

```
tsm pending-changes apply
```

Reinicie el servicio tsig-httpd

A medida que su implementación de Tableau Server aplique cambios, vuelva a iniciar sesión en el equipo de la puerta de enlace independiente de Tableau Server y ejecute los siguientes comandos para reiniciar el servicio tsig-httpd:

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

Validar la funcionalidad SAML

Para validar la funcionalidad SAML de un extremo a otro, inicie sesión en Tableau Server con la URL pública (p. ej., https://tableau.example.com) con la cuenta de administrador de Tableau que creó al comienzo de este procedimiento.

Si TSM no se inicia ("error de puerta de enlace") o si obtiene errores del navegador cuando intenta conectarse, consulte Solucionar problemas de la puerta de enlace independiente de Tableau Server.

Configurar el módulo de autenticación en la segunda instancia de la puerta de enlace independiente

Una vez que haya configurado correctamente la primera instancia de la puerta de enlace independiente, implemente la segunda instancia. El ejemplo aquí es el proceso final para instalar el escenario de AWS/Mellon/Okta descrito en este tema. El procedimiento asume que ya instaló la puerta de enlace independiente en la segunda instancia como se describe en este tema anteriormente (Instalar la puerta de enlace independiente).

El proceso para implementar la segunda puerta de enlace independiente requiere los siguientes pasos:

1. En la segunda instancia de la puerta de enlace independiente: Instale el módulo de autenticación de Mellon.

   No configure el módulo de autenticación de Mellon como se describe anteriormente en este tema. En su lugar, debe clonar la configuración como se describe en los pasos siguientes.

2. En la (primera) instancia configurada de la puerta de enlace independiente:

   Tome una copia tar de la configuración de Mellon existente. La copia de seguridad tar conservará todos los permisos y la jerarquía de directorios. Ejecute los comandos siguientes:

   ```
   cd /etc
   ```

   ```
   sudo tar -cvf mellon.tar mellon
   ```

   Copie `mellon.tar` en la segunda instancia de la puerta de enlace independiente.

3. En la segunda instancia de la puerta de enlace independiente:

Extraiga ("descomprima") el archivo tar a la segunda instancia en el directorio `/etc`. Ejecute los comandos siguientes:

```
cd /etc

sudo tar -xvf mellon.tar
```

4. En el nodo 1 de la implementación de Tableau Server: actualice el archivo de conexión (`tsig.json`) con la información de conexión de la segunda puerta de enlace independiente. Deberá recuperar la clave de autenticación como se describe en este tema anteriormente (Instalar la puerta de enlace independiente).

   Un archivo de conexión de ejemplo (`tsig.json`) se muestra aquí:

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

5. En el nodo 1 de la implementación de Tableau Server: ejecute los siguientes comandos para actualizar la configuración:

```
tsm stop
```

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

tsm start
```

6. En ambas instancias de la puerta de enlace independiente: mientras se inicia Tableau Server, reinicie el proceso `tsig-httpd`:

   ```
   sudo su - tableau-tsig

   systemctl --user restart tsig-httpd

   exit
   ```

7. En AWS **EC2>Grupos de destino**: actualice el grupo de destino para incluir la instancia EC2 que ejecuta la segunda instancia de la puerta de enlace independiente.

   Seleccione el grupo de destino que acaba de crear y luego haga clic en la pestaña Destinos:

   - Haga clic en **Editar**.
   - Seleccione la instancia EC2 del segundo equipo de la puerta de enlace independiente y luego haga clic en **Agregar a registrado**. Haga clic en **Guardar**.

Parte 6: configuración después de la instalación

Configurar SSL/TLS desde el equilibrador de carga a Tableau Server

Algunas organizaciones requieren un canal de cifrado de un extremo a otro desde el cliente hasta el servicio de back-end. La arquitectura de referencia predeterminada, como se describe hasta este punto, especifica SSL desde el cliente hasta el equilibrador de carga que se ejecuta en el nivel web de su organización.

Esta sección describe cómo configurar SSL/TLS para Tableau Server y la puerta de enlace independiente en la arquitectura de referencia de AWS de ejemplo. Para ver un ejemplo de configuración que describe cómo configurar SSL/TLS en Apache en la arquitectura de referencia de AWS, consulte Ejemplo: Configurar SSL/TLS en la arquitectura de referencia de AWS.

En este momento, TLS no es compatible con los procesos backend de Tableau Server que se ejecutan en el rango 8000-9000. Para habilitar TLS, debe configurar la puerta de enlace independiente con una conexión de retransmisión a Tableau Server.

Este procedimiento describe cómo habilitar y configurar TLS en una puerta de enlace independiente a Tableau Server y Tableau Server a una puerta de enlace independiente. El procedimiento cifra el tráfico de retransmisión a través de HTTPS/443 y el tráfico de limpieza a través de HTTPS/21319.

Los procedimientos de Linux a lo largo de este ejemplo muestran comandos para distribuciones similares a RHEL. Específicamente, los comandos aquí se han desarrollado con la distribución de Amazon Linux 2. Si está ejecutando distribuciones de Ubuntu, edite los comandos según corresponda.

La guía aquí es prescriptiva para la arquitectura de referencia de ejemplo específica de AWS como se presenta en esta guía. Por lo tanto, las configuraciones opcionales no están incluidas. Para obtener documentación de referencia completa, consulte *Configuración de TLS en una puerta de enlace independiente* (Linux).

Antes de configurar TLS

Realice las configuraciones de TLS fuera del horario comercial. La configuración requiere al menos un reinicio de Tableau Server. Si está ejecutando una implementación completa de arquitectura de referencia de cuatro nodos, el reinicio puede llevar un tiempo.

- Verifique que los clientes puedan conectarse a Tableau Server a través de HTTP. La configuración de TLS con puerta de enlace independiente es un proceso de varios pasos y puede requerir la resolución de problemas. Por lo tanto, recomendamos comenzar con una implementación de Tableau Server completamente operativa antes de configurar TLS.

- Recopile certificados TLS/SSL, claves y activos relacionados. Necesitará certificados SSL para las puertas de enlace independientes y para Tableau Server. Para simplificar la administración y la implementación de certificados, y como práctica recomendada de seguridad, recomendamos usar certificados generados por una autoridad de certificación (CA) de terceros de confianza. Como alternativa, puede generar certificados autofirmados o usar certificados de una PKI para TLS.

  La configuración de ejemplo de este tema utiliza los siguientes nombres de recursos a modo de ilustración:
  - `tsig-ssl.crt`: El certificado TLS/SSL para la puerta de enlace independiente.
  - `tsig-ssl.key`: La clave privada para `tsig-ssl.crt` en la puerta de enlace independiente.
  - `ts-ssl.crt`: El certificado TLS/SSL para Tableau Server.
  - `ts-ssl.key`: La clave privada para `tsig-ssl.crt` en Tableau Server.
  - `tableau-server-CA.pem`: El certificado de raíz para la CA que genera el certificado presentado por los equipos de Tableau Server. Por lo general, este

certificado no es necesario si utiliza certificados de terceros de confianza importantes.

- `rootTSIG-CACert.pem`: El certificado raíz de la CA que genera los certificados para los equipos de la puerta de enlace independiente. Por lo general, este certificado no es necesario si utiliza certificados de terceros de confianza importantes.
- Hay otros certificados y recursos de archivos clave necesarios para SAML que se detallan en la Parte 5 de esta guía.
- Si su implementación requiere el uso de un archivo de cadena de certificados, consulte el artículo de la base de conocimientos Configurar TLS en una puerta de enlace independiente al usar un certificado que tiene una cadena de certificados.

- Verifique que tiene acceso a IdP. Si utiliza un IdP para la autenticación, es probable que deba realizar cambios en las direcciones URL de destinatario y destino en el IdP después de haber configurado SSL/TLS.

Configurar equipos de la puerta de enlace independiente para TLS

La configuración de TLS puede ser un proceso propenso a errores. Dado que la solución de problemas en dos instancias de la puerta de enlace independiente puede llevar mucho tiempo, recomendamos habilitar y configurar TLS en la implementación de EDG con solo una puerta de enlace independiente. Una vez que haya validado que TLS funciona en toda la implementación, configure el segundo equipo de puerta de enlace independiente.

Paso 1: Distribuir certificados y claves al equipo de puerta de enlace independiente

Puede distribuir los activos a cualquier directorio arbitrario siempre que el usuario tsig-httpd tenga acceso de lectura a los archivos. Las rutas a estos archivos se mencionan en otros procedimientos. Usaremos las rutas de ejemplo bajo `/etc/ssl` , como se muestra a continuación, a lo largo del tema.

1. Crear directorio para clave privada:

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

2. Copie los archivos de certificado y de clave en las rutas `/etc/ssl`. Por ejemplo:

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

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

3. (Opcional) Si usa un certificado PKI o autofirmado para SSL/TLS en Tableau Server, también debe copiar el archivo del certificado raíz de CA en el equipo de la puerta de enlace independiente. Por ejemplo:

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

Paso 2: Actualizar las variables ambientales para TLS

Debe actualizar las variables ambientales del puerto y del protocolo para la configuración de la puerta de enlace independiente.

Cambie estos valores actualizando el archivo,`/etc/opt/tableau/tableau_tsig/environment.bash`, como se indica a continuación:

```
TSIG_HK_PROTOCOL="https"
TSIG_PORT="443"
TSIG_PROTOCOL="https"
```

Paso 3: Actualizar el archivo de configuración de código auxiliar para el protocolo HK

Edite manualmente el archivo de configuración de código auxiliar (`/var/opt/tableau/tableau_tsig/config/httpd.conf.stub`) para establecer directivas Apache httpd relacionadas con TLS para el protocolo de limpieza (HK).

El archivo de configuración de código auxiliar incluye un bloque de directivas relacionadas con TLS que se comentan con un marcador `#TLS#`. Quite los marcadores de las directivas como se muestra en el siguiente ejemplo. Tenga en cuenta que el ejemplo muestra el uso del

certificado CA raíz para el certificado SSL usado en Tableau Server con la opción `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>
```

Estos cambios se perderán si vuelve a instalar la puerta de enlace independiente. Recomendamos hacer una copia de seguridad.

Paso 4: Copiar el archivo de resguardo y reiniciar el servicio

1. Copie el archivo que actualizó en el último paso para actualizar httpd.conf con los cambios:

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

2. Reinicie el servicio de puerta de enlace independiente:

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

Después de reiniciar, la puerta de enlace independiente no estará operativa hasta que ejecute el siguiente conjunto de pasos en Tableau Server. Una vez que haya completado los pasos en Tableau Server, la puerta de enlace independiente recogerá los cambios y se conectará.

Configurar el nodo 1 de Tableau Server para TLS

Ejecute estos pasos en el nodo 1 de la implementación de Tableau Server.

Paso 1: Copiar certificados y claves y detener TSM

1. Verifique que tenga los certificados y las claves "SSL externo" de Tableau Server copiados en el nodo 1.
2. Para minimizar el tiempo de inactividad, recomendamos detener TSM, ejecutar los siguientes pasos y luego iniciar TSM después de aplicar los cambios:

   ```
   tsm stop
   ```

Paso 2: Configurar los activos del certificado y habilitar la configuración de la puerta de enlace independiente

1. Especifique la ubicación de los archivos de certificados y claves para la puerta de enlace independiente. Estas rutas hacen referencia a la ubicación en los equipos de la puerta de enlace independiente. Tenga en cuenta que este ejemplo asume que se usa el mismo certificado y par de claves para proteger HTTPS y el tráfico de mantenimiento:

   ```
   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. Habilite TLS para los protocolos HTTPS y HK para la puerta de enlace independiente:

   ```
   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. (Opcional) Si utiliza un certificado PKI o autofirmado para SSL/TLS en la puerta de enlace independiente, debe cargar el archivo del certificado raíz de CA. El archivo de

certificado raíz de CA es el certificado raíz que se usó para generar los certificados para los equipos de la puerta de enlace independiente. Por ejemplo:

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

4. (Opcional) Si usa un certificado PKI o autofirmado para SSL/TLS en Tableau Server, también debe copiar el archivo del certificado raíz de CA en el directorio `/etc/ssl/certs` de la puerta de enlace independiente. El archivo de certificado raíz de CA es el certificado raíz que se usó para generar los certificados para los equipos de Tableau Server. Una vez que haya copiado el certificado en la puerta de enlace independiente, debe especificar la ubicación del certificado en el nodo 1 con el siguiente comando tsm. Por ejemplo:

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

5. (Opcional: solo con fines de prueba) Si está utilizando el uso compartido de certificados PKI o autofirmados entre equipos y, por lo tanto, los nombres de sujetos en los certificados no coinciden con los nombres de los equipos, entonces debe deshabilitar la verificación de certificados.

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

Paso 3: Habilitar "SSL externo" para Tableau Server y aplicar los cambios

1. Habilite y configure "SSL externo" en Tableau Server:

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

2. Aplique los cambios.

   ```
   tsm pending-changes apply
   ```

Paso 4: Actualizar el archivo JSON de configuración de la puerta de enlace e iniciar tsm

1. Actualice el archivo de configuración de la puerta de enlace independiente (por ejemplo,`tsig.json`) en el lado de Tableau Server para especificar el protocolo `https` para los objetos de puerta de enlace independiente:

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

2. Elimine (o comente) la información de conexión para la segunda instancia de la puerta de enlace independiente. Asegúrese de verificar el JSON en un editor externo antes de guardarlo.

   Una vez que haya configurado y validado TLS para la instancia única de la puerta de enlace independiente, actualizará este archivo JSON con la información de conexión para la segunda instancia de la puerta de enlace independiente.

3. Ejecute el siguiente comando para actualizar la configuración de la puerta de enlace independiente:

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

4. Inicie TSM.

   ```
   tsm start
   ```

5. Mientras se inicia TSM, inicie sesión en la instancia de la puerta de enlace independiente y reinicie el servicio `tsig-httpd`:

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

Actualice las URL del módulo de autenticación de IdP a HTTPS

Si ha configurado un proveedor de identidad externo para Tableau, es probable que deba actualizar las URL de retorno en el panel administrativo del IdP.

Por ejemplo, si está utilizando una aplicación de preautenticación Okta, deberá actualizar la aplicación para usar el protocolo HTTPS para la URL del destinatario y la URL de destino.

Configurar el equilibrador de carga de AWS para HTTPS

Si está implementando con el equilibrador de carga de AWS como se documenta en esta guía, vuelva a configurar el equilibrador de carga de AWS para enviar tráfico HTTPS a los equipos que ejecutan la puerta de enlace independiente:

1. Elimine el registro del grupo de destino HTTP existente:

   En **Grupos de destino**, seleccione el grupo de destino HTTP que se ha configurado para el equilibrador de carga, haga clic en **Acciones** y, a continuación, haga clic en **Eliminar**.

2. Cree un grupo de destino HTTPS:

   Grupos de destino > **Crear grupo de destino**

   - Seleccione "Instancias"
   - Escriba un nombre de grupo de destino, `TG-internal-HTTPS` por ejemplo
   - Seleccione su VPC
   - Protocolo: HTTPS 443
   - En **Comprobaciones de estado** > **Configuración de comprobaciones de estado avanzadas** > **Códigos de éxito**, agregue la lista de códigos para leer: `200,303`.
   - Haga clic en **Crear**.

3. Seleccione el grupo de destino que acaba de crear y luego haga clic en la pestaña **Destinos**:

    - Haga clic en **Editar**
    - Seleccione la instancia EC2 que están ejecutando la puerta de enlace independiente de Tableau Server que ha configurado y luego haga clic en **Añadir a registrados**.
    - Haga clic en **Guardar**.

4. Una vez creado el grupo objetivo, debe habilitar la adherencia:

    - Abra la página del grupo de destino de AWS (**EC2**> **Equilibradores de carga** > **Grupos de destino**), seleccione la instancia del grupo de destino que acaba de configurar. En el menú **Acciones**, seleccione **Editar atributos**.
    - En la página **Editar atributos**, seleccione **Adherencia**, especifique una duración de `1 day` y luego **Guardar cambios**.

5. En el equilibrador de carga, actualice las reglas del oyente. Seleccione el equilibrador de carga que ha configurado para esta implementación y luego haga clic en la pestaña **Oyentes**.

    - Para **HTTP:80**, haga clic en **Ver/editar reglas**. En la página **Reglas** resultante, haga clic en el icono de edición (una vez en la parte superior de la página y luego nuevamente por la regla) para editar la regla. Elimine la regla THEN existente y reemplácela haciendo clic en **Agregar acción** > **Redirigir a...** En la configuración THEN resultante, especifique`HTTPS` y el puerto `443` y deje las otras opciones con la configuración predeterminada. Guarde la configuración y haga clic en **Actualizar**.
    - Para **HTTPS:443**, haga clic en **Ver/editar reglas**. En la página **Reglas** resultante, haga clic en el icono de edición (una vez en la parte superior de la página y luego nuevamente por la regla) para editar la regla. Elimine la regla THEN existente y reemplácela haciendo clic en **Agregar acción** > **Reenviar a...** Especifique el Grupo de destino para el grupo HTTPS que acaba de crear. En **Adherencia a nivel de grupo**, habilite la adherencia y establezca la duración en 1 día. Guarde la configuración y haga clic en **Actualizar**.

6. En el equilibrador de carga, actualice el tiempo de espera inactivo a 400 segundos. Seleccione el equilibrador de carga que ha configurado para esta implementación y luego haga clic en **Acciones** > **Editar atributos**. Establezca el tiempo de **espera inactivo** en `400` segundos y luego haga clic en **Guardar**.

Validar TLS

Para validar la funcionalidad TLS, inicie sesión en Tableau Server con la URL pública (p. ej., https://tableau.example.com) con la cuenta de administrador de Tableau que creó al comienzo de este procedimiento.

Si TSM no se inicia o recibe otros errores, consulte Solucionar problemas de la puerta de enlace independiente de Tableau Server.

Configurar la segunda instancia de la puerta de enlace independiente para SSL

Una vez que haya configurado correctamente la primera instancia de la puerta de enlace independiente, implemente la segunda instancia.

El proceso para implementar la segunda puerta de enlace independiente requiere los siguientes pasos:

1. En la (primera) instancia configurada de la puerta de enlace independiente: copie los siguientes archivos en las ubicaciones correspondientes en la segunda instancia de la puerta de enlace independiente:
    - `/etc/ssl/certs/tsig-ssl.crt`
    - `/etc/ssl/private/tsig-ssl.key` (Tendrá que crear el directorio `private` en la segunda instancia).
    - `/var/opt/tableau/tableau_tsig/config/httpd.conf.stub`
    - `/etc/opt/tableau/tableau_tsig/environment.bash`

2. En el nodo 1 de la implementación de Tableau Server: actualice el archivo de conexión (`tsig.json`) con la información de conexión de la segunda puerta de enlace independiente.

   Un archivo de conexión de ejemplo (`tsig.json`) se muestra aquí:

   ```
   {
   "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. En el nodo 1 de la implementación de Tableau Server: ejecute los siguientes comandos para actualizar la configuración:

   ```
   tsm stop
   ```

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

   ```
   tsm start
   ```

4. En ambas instancias de Puerta de enlace independiente: Mientras se inicia Tableau Server, reinicie el proceso `tsig-httpd` en ambas instancias de la puerta de enlace independiente:

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

```
systemctl --user restart tsig-httpd

exit
```

5. En AWS **EC2>Grupos de destino**: actualice el grupo de destino para incluir la instancia EC2 que ejecuta la segunda instancia de la puerta de enlace independiente.

   Seleccione el grupo de destino que acaba de crear y luego haga clic en la pestaña Destinos:

   - Haga clic en **Editar**.
   - Seleccione la instancia EC2 del segundo equipo de la puerta de enlace independiente y luego haga clic en **Agregar a registrado**. Haga clic en **Guardar**.

Configurar SSL para Postgres

Opcionalmente, puede configurar SSL (TLS) para la conexión de Postgres para la conexión del repositorio externo en Tableau Server.

Para simplificar la administración y la implementación de certificados, y como práctica recomendada de seguridad, recomendamos usar certificados generados por una autoridad de certificación (CA) de terceros de confianza. Como alternativa, puede generar certificados autofirmados o usar certificados de una PKI para TLS.

Este procedimiento describe cómo utilizar OpenSSL para generar un certificado autofirmado en el host de Postgres en una distribución Linux de tipo RHEL en la arquitectura de referencia de AWS de ejemplo.

Después de generar y firmar el certificado SSL, debe copiar el certificado CA en el host de Tableau.

En el host que ejecuta Postgres:

1. Genere la clave de la autoridad certificadora (CA) raíz de firma:

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

2. Cree el certificado de CA raíz:

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

   Se le pedirá que especifique los valores para los campos del certificado. Por ejemplo:

   ```
   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. Cree el certificado y la clave relacionada (`server.csr` y `server.key` en el ejemplo siguiente) para el equipo de Postgres. El nombre del sujeto del certificado debe coincidir con el nombre DNS privado de EC2 del host de Postgres. El nombre del sujeto se establece con la opción `-subj` con el formato "`/CN=<private DNS name>`", por ejemplo:

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

4. Firme el nuevo certificado con el certificado CA que creó en el paso 2. El siguiente comando también genera el certificado en el formato `crt`:

   ```
   openssl x509 -req -in server.csr -days 3650 -CA pgsql-rootCACer-
   t.pem -CAkey pgsql-rootCAKey.pem -CAcreateserial -out ser-
   ver.crt
   ```

5. Copie los archivos crt y key en la ruta `/var/lib/pgsql/13/data/` de Postgres:

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

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

6. Cambiar a usuario raíz:

```
sudo su
```

7. Establezca permisos en los archivos cer y key. Ejecute los comandos siguientes:

```
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. Actualice el archivo de configuración pg_haba `/var/lib/pgsql/13/data/pg_hba.conf` para especificar la confianza md5:

   Cambie las declaraciones de conexión existentes de

   `host all all 10.0.30.0/24 password` y

   `host all all 10.0.31.0/24 password`

   to

   `host all all 10.0.30.0/24 md5` y

   `host all all 10.0.31.0/24 md5`.

9. Actualice el archivo postgresql `/var/lib/pgsql/13/data/postgresql.conf` agregando esta línea:

```
ssl = on
```

10. Salga del modo de usuario raíz:

```
exit
```

11. Reiniciar Postgres:

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

Opcional: habilite la validación de confianza de certificados en Tableau Server para Postgres SSL

Si siguió el procedimiento de instalación en la Paso 4: Instalar y configurar Tableau Server, entonces Tableau Server está configurado con SSL opcional para la conexión de Postgres. Esto significa que la configuración de SSL en Postgres (como se describe anteriormente) dará como resultado una conexión cifrada.

Si desea solicitar la validación de confianza del certificado para la conexión, debe ejecutar el siguiente comando en Tableau Server para volver a configurar la conexión del host de Postgres:

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

Donde `<filename>.json` es el archivo de conexión descrito en Configurar Postgres externo. Y `CACert.pem` es el archivo de certificado de CA para el certificado SSL/TLS utilizado por Postgres.

Opcional: Verificar la conectividad SSL

Para verificar la conectividad SSL, debe:

- Instalar el cliente de Postgres en el nodo 1 de Tableau Server.
- Copiar el certificado raíz que creó en el procedimiento anterior en el host de Tableau.
- Conectarse al servidor Postgres desde el nodo 1

Instalar el cliente de Postgres en el nodo 1

Este ejemplo muestra cómo instalar Postgres 13.4. Instale la misma versión que está ejecutando para el repositorio externo.

1. En el Nodo 1, cree y edite el archivo pgdg.repo. en la ruta `/etc/yum.repos.d`. Complete el archivo con la siguiente información de configuración.

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

2. Instale el cliente de Postgres:

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

Copiar el certificado raíz al Nodo 1

Copie el certificado de CA (`pgsql-rootCACert.pem`) en el host de Tableau:

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

Conectarse al host de Postgres a través de SSL desde el Nodo 1

Ejecute el siguiente comando desde el Nodo 1, especificando la dirección IP del host del servidor de Postgres y el certificado de CA raíz:

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

Por ejemplo:

```
 psql "pos-
tgresql://postgres@10.0.1.189:5432/postgres?sslmode=verify-ca&ss-
lrootcert=pgsql-rootCACert.pem"
```

Postgres le pedirá la contraseña. Después de iniciar sesión correctamente, el shell devolverá:

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

Configurar SMTP y notificaciones de eventos

Tableau Server envía notificaciones de correos electrónicos a administradores y usuarios. Para habilitarlas, debe configurar Tableau Server para enviar correo a su servidor de correo electrónico. También debe especificar los tipos de eventos, los umbrales y la información de suscripción que desea enviar.

En el caso de la configuración inicial de SMTP y las notificaciones, se recomienda utilizar la siguiente plantilla de archivo de configuración para crear un archivo .json. También puede establecer cualquier clave de configuración única de las que aparecen abajo con la sintaxis descrita en *tsm configuration set* (Linux).

Ejecute este procedimiento en el nodo 1 de su implementación de Tableau Server:

1. Copie la siguiente plantilla de json en un archivo. Personalice el archivo con sus opciones de configuración SMTP y la suscripción y las notificaciones de alerta para su organización.
    - Para ver una lista y una descripción de todas las opciones de SMTP, consulte *Referencia de configuración de la CLI de SMTP* (Linux).
    - Para ver una lista y una descripción de todas las opciones de eventos de notificación, consulte la sección CLI de *Configurar la notificación de eventos del servidor* (Linux).

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

```
        "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. Ejecute el `tsm settings import -f file.json` para pasar el archivo json a Tableau Services Manager.
3. Ejecute el comando `tsm pending-changes apply` para aplicar los cambios.
4. Ejecute el comando `tsm email test-smtp-connection` para ver y verificar la configuración de la conexión.

Instalar el controlador de PostgreSQL

Para ver las vistas de administrador en Tableau Server, el controlador de PostgreSQL debe estar instalado en el nodo 1 de la implementación de Tableau Server.

1. Vaya a la página de descarga del controlador de Tableau y copie la URL del archivo .jar de PostgreSQL.
2. Ejecute el siguiente procedimiento en cada nodo de la implementación de Tableau:
   - Cree la siguiente ruta de archivo:

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

   - Desde la nueva ruta, descargue la versión más reciente del archivo .jar de PostgreSQL. Por ejemplo:

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

3. En el nodo inicial, restaure Tableau Server:

   ```
   tsm restart
   ```

Configurar una directiva de contraseñas seguras

Si no está implementando Tableau Server con un solución de autenticación de IdP, le recomendamos que refuerce la seguridad de la directiva de contraseñas predeterminada de Tableau.

Si está implementando Tableau Server con un IdP, debe administrar las directivas de contraseñas con el IdP.

El siguiente procedimiento incluye la configuración json para establecer la directiva de contraseñas en Tableau Server. Para obtener más información sobre las opciones siguientes, consulte *Autenticación local* (Linux).

1. Copie la siguiente plantilla de json en un archivo. Rellene los valores de las claves con su configuración de directiva de contraseña.

```
{
 "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,
       "vizportal.password_reset": true
 }
}
```

2. Ejecute el `tsm settings import -f file.json` para pasar el archivo json a Tableau Services Manager para configurar Tableau Server.

3. Ejecute el comando `tsm pending-changes apply` para aplicar los cambios.

Parte 7: Validación, herramientas y solución de problemas

Esta parte incluye pasos de validación posteriores a la instalación y orientación para la resolución de problemas.

Validación del sistema de conmutación por error

Una vez que haya configurado su implementación, le recomendamos que ejecute pruebas de conmutación por error simples para validar la redundancia del sistema.

Recomendamos ejecutar los siguientes pasos para validar la funcionalidad de conmutación por error:

1. Apague la primera instancia de la puerta de enlace independiente (TSIG1). Todo el tráfico entrante debe enrutarse a través de la segunda instancia de la puerta de enlace independiente (TSIG2).
2. Reinicie TSIG1 y luego apague TSIG2. Todo el tráfico entrante debe enrutarse a través de TSIG1.
3. Reinicie TSIG2.

4. Apague el nodo 1 de Tableau Server. Todo el tráfico del servicio Vizportal/Application se conmutará por error al Nodo 2.

   Nota: A partir de septiembre de 2022, la alta disponibilidad del Nodo 1 está comprometida en determinadas versiones de Tableau Server 2021.4 y posteriores. Las conexiones del cliente fallarán si el Nodo 1 está inactivo. Este problema se ha solucionado en estas versiones de mantenimiento:

- 2021.4.15 y posteriores
- 2022.1.11 y posteriores
- 2023.1.3 y posteriores

Para garantizar que su instalación de Tableau Server mediante activaciones ATR tenga un período de gracia de 72 horas después del error inicial del nodo, instale o actualice a una de estas versiones. Para obtener más información, consulte Tableau Server HA que usa ATR no tiene un período de gracia después del error inicial del nodo en la base de conocimientos de Tableau.

5. Reinicie el Nodo 1 y apague el Nodo 2. Todo el tráfico del servicio Vizportal/Application se conmutará por error al Nodo 1.
6. Reinicie el nodo 2.

En este contexto, "apagar" o "reiniciar" se realiza apagando el sistema operativo o la máquina virtual sin intentar cerrar correctamente la aplicación de antemano. El objetivo es simular una falla de hardware o máquina virtual.

El paso de validación mínimo para cada prueba de conmutación por error es autenticarse con un usuario y realizar operaciones de visualización básicas.

Es posible que obtenga un error del navegador "Solicitud incorrecta" cuando intente iniciar sesión después de una falla simulada. Es posible que vea este error incluso si borra el caché en el navegador. A menudo, este problema ocurre cuando el navegador está almacenando en caché datos de la sesión anterior de IdP. Si este error persiste incluso después de borrar la memoria caché del navegador local, valide el escenario de Tableau conectándose con un navegador diferente.

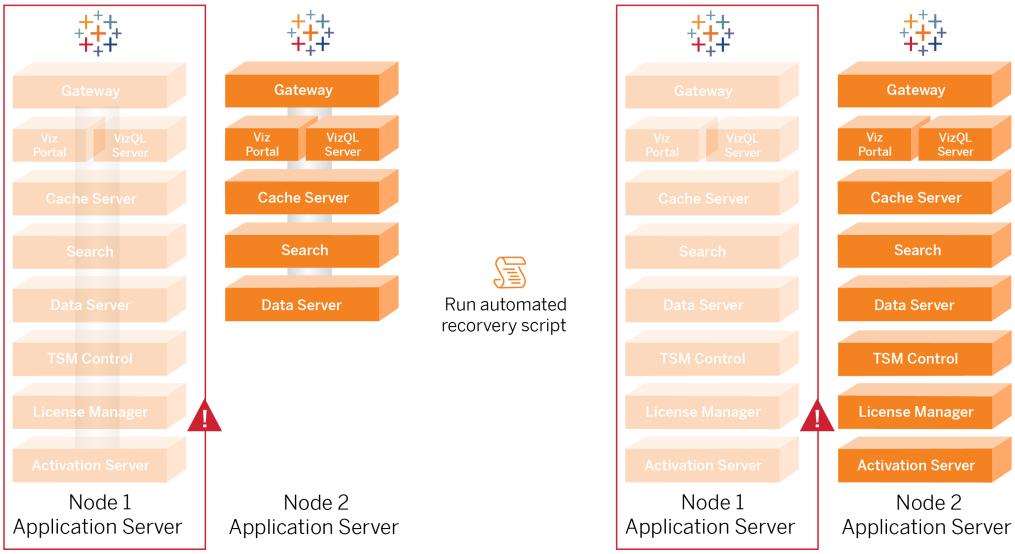
Recuperación automatizada inicial del nodo

Tableau Server 2021.2.4 y sus versiones posteriores incluyen un script de recuperación de nodo inicial automatizado, `auto-node-recovery`, en el directorio de scripts (

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

Si hay un problema en el nodo inicial y tiene procesos redundantes en el nodo 2, no hay garantía de que Tableau Server pueda seguir ejecutándose. Tableau Server puede continuar ejecutándose hasta 72 horas después de un error inicial del nodo, antes de que la falta del servicio de licencias afecte a otros procesos. Si es así, los usuarios podrán seguir iniciando sesión y ver su contenido después del fallo del nodo inicial, pero no podrán reconfigurar Tableau Server porque no tiene acceso al Controlador de administración.

Incluso cuando se configura con procesos redundantes, es posible que Tableau Server no continúe funcionando después de que falle el nodo inicial.



Para recuperar el error del nodo inicial (nodo 1):

1. Inicie sesión en el nodo 2 de Tableau Server
2. Cambie al directorio de scripts:

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

3. Ejecute el siguiente comando para iniciar el script:

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

Donde `<license keys>` es una lista separada por comas (sin espacios) de las claves de licencia para su implementación. Si no tiene acceso a sus claves de licencia, visite el Portal del cliente de Tableau para recuperarlas. Por ejemplo:

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

El script de recuperación automática de nodos ejecutará unos 20 pasos para recuperar servicios en el nodo 2. Cada paso se muestra en la terminal a medida que avanza el script. Se registra un estado más detallado en `/data/tableau_data/logs/app-controller-move.log`. En la mayoría de los entornos, el script tarda entre 35 y 45 minutos en completarse.

Solución de problemas de recuperación inicial del nodo

Si la recuperación del nodo falla, puede resultarle útil ejecutar el script de forma interactiva para permitir o no permitir determinados pasos durante el proceso. Por ejemplo, si el script falla en la mitad del proceso, puede revisar el archivo de registro, realizar cambios en la configuración y luego ejecutar el script de nuevo. Al ejecutarlo en modo interactivo, puede omitir todos los pasos hasta llegar al paso que falló.

Para ejecutar en modo interactivo, agregue el alternador `-i` al argumento del script.

Reconstrucción del nodo fallido

Una vez que haya ejecutado el script, el nodo 2 ejecutará todos los servicios que antes estaban en el host del nodo 1 que falló. Para agregar el nodo 4, debe implementar un host de Tableau Server nuevo con el archivo de arranque y configurarlo como lo hizo para el nodo 2 original, tal y como se especifica en la parte 4. Consulte Configurar el nodo 2.

switchto

Switchto es un script de Tim que facilita el cambio entre ventanas.

1. Copie el siguiente código en un archivo llamado `switchto` en el directorio de inicio de su host de 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. Actualice las direcciones IP en el script para asignarlas a sus instancias EC2 y, después, guarde el archivo.
3. Aplique los permisos al archivo del script:

   ```
   sudo chmod +x switchto
   ```

Uso:

Para cambiar a un host, ejecute el comando siguiente:

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

Por ejemplo, para cambiar al nodo 1, ejecute el siguiente comando:

```
./switchto node1
```

Solucionar problemas de la puerta de enlace independiente de Tableau Server

La configuración de la puerta de enlace independiente, Okta, Mellon y SAML en Tableau Server puede ser un proceso propenso a errores. La causa raíz más común de estos errores es un error de cadena. Por ejemplo, una barra inclinada final (`/`) en las URL de Okta especificadas durante la configuración puede causar un error de discrepancia relacionado con la aserción de SAML. Esto es solo un ejemplo. Hay muchas oportunidades durante la configuración para indicar una cadena incorrecta en cualquiera de las aplicaciones.

Reiniciar el servicio tableau-tsig

Inicie (y finalice) siempre la resolución de problemas reiniciando el servicio tableau-tsig en los equipos de la puerta de enlace independiente. Reiniciar este servicio es rápido y, a menudo, activa la configuración actualizada para que se cargue desde Tableau Server.

Ejecute los siguientes comandos en el equipo de la puerta de enlace independiente:

```
sudo su - tableau-tsig

systemctl --user restart tsig-httpd

exit
```

Encontrar cadenas incorrectas

Si cometió un error de cadena (error de copiar/pegar, cadena truncada, etc.), tómese el tiempo para revisar cada una de las configuraciones que configuró:

- Configuración de autenticación previa de Okta. Revise cuidadosamente las URL que ha establecido. Busque barras inclinadas. Verifique HTTP frente a HTTPS.
- Historial de shell para la configuración de SAML en el nodo 1. Revise el comando `tsm authentication saml configure` que ejecutó. Verifique que todas las URL coincidan con las que ha configurado en Okta. Mientras revisa el historial de shell del Nodo 1, verifique que los comandos `tsm configuration set` que especifican las rutas del archivo de configuración de Mellon se asignan exactamente a las rutas del archivo donde copió los archivos en la puerta de enlace independiente.
- Configuración de Mellon en la puerta de enlace independiente. Revise el historial de shell para verificar que creó los metadatos con la misma cadena de URL que configuró en Okta y Tableau SAML. Verifique que todas las rutas que se especifican en `/etc/-mellon/conf.d/global.conf` son correctos y que el `MellonCookieDomain` está configurado en su dominio raíz, no en su subdominio de Tableau.

Buscar registros relevantes

Si todas las cadenas parecen estar configuradas correctamente, debe inspeccionar los registros en busca de errores.

Tableau Server registra errores y eventos en docenas de archivos de registro diferentes. La puerta de enlace independiente también registra un conjunto de archivos locales. Recomendamos inspeccionar estos registros en el siguiente orden.

Archivos de registro de puerta de enlace independiente

La ubicación predeterminada de los archivos de registro de la puerta de enlace independiente es la siguiente: `/var/opt/tableau/tableau_tsig/logs`

- `access.log`: este registro es útil en la medida en que tiene entradas que muestran conexiones desde los nodos de Tableau Server. Si recibe errores de puerta de enlace (no se inicia) cuando intenta iniciar TSM y no hay entradas en el archivo `access.log`, entonces hay un problema de conectividad central. Verifique siempre la configuración del grupo de seguridad de AWS como primer paso. Otro problema común es un error tipográfico en `tsig.json`. Si realiza una actualización de tsig.json, ejecute `tsm stop` antes de ejecutar `tsm topology external-services gateway update -c tsig.json`. Después de actualizar tsig.json, ejecute `tsm start`.
- `error.log`: Entre otras entradas, este registro incluye errores SAML y Mellon.

Archivo de registro tabadminagent de Tableau Server

El conjunto de archivos `tabadminagent` (no `tabadmincontroller`) son los únicos archivos de registro relevantes para la resolución de errores relacionados con la puerta de enlace independiente.

Debe encontrar dónde se han registrado los errores de la puerta de enlace independiente en `tabdminagent`. Estos errores pueden estar en cualquier nodo, pero solo están en un nodo. Realice los siguientes pasos en cada nodo del clúster de Tableau Server hasta que encuentre la cadena "independiente":

1. Busque la ubicación del archivo de registro tabadminagent en los nodos 1 a 4 de Tableau Server en la configuración de EDG:

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

2. Abra el último registro para leer:

   ```
   less tabadminagent_nodeN.log
   ```

   (reemplace N con el número de nodo)

3. Busque todas las instancias de "Independiente" e "independiente" utilizando la siguiente cadena de búsqueda:

   ```
   /ndependent
   ```

   Si no hay coincidencias, vaya al siguiente nodo y repita los pasos 1-3.

4. Cuando obtiene una coincidencia: use `Shift + G` para moverse hacia abajo para obtener los últimos mensajes de error.

Recargar archivo auxiliar httpd

La puerta de enlace independiente gestiona la configuración de httpd para Apache. Una operación genérica que a menudo soluciona problemas transitorios es volver a cargar el archivo auxiliar httpd que genera la configuración subyacente de Apache. Ejecute los siguientes comandos en ambas instancias de la puerta de enlace independiente.

1. Copie el archivo de resguardo en httpd.conf:

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

2. Reinicie el servicio de puerta de enlace independiente:

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

Eliminar o mover archivos de registro

La puerta de enlace independiente registra todos los eventos de acceso. Deberá administrar el almacenamiento de archivos de registro para evitar llenar el espacio en disco. Si su disco se llena, la puerta de enlace independiente no podrá escribir eventos de acceso y el servicio fallará. El siguiente mensaje se registrará en error.log en la puerta de enlace independiente:

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

Este error resultará en un estado de `DEGRADED` para el nodo `external` cuando ejecuta `tsm status -v` en el nodo 1 de Tableau. El nodo `external` en la salida de estado se refiere a la puerta de enlace independiente.

Para resolver este problema, elimine o mueva los archivos access.log del disco. Los archivos Access.log se almacenan en `/var/opt/tableau/tableau_tsig/logs`. Después de borrar el disco, reinicie el servicio tableau-tsig.

Errores del navegador

Solicitud incorrecta: un error común para este escenario es un error de "Solicitud incorrecta" de Okta. A menudo, este problema ocurre cuando el navegador está almacenando en caché datos de la sesión anterior de Okta. Por ejemplo, si administra las aplicaciones de Okta como administrador de Okta y luego intenta acceder a Tableau con una cuenta diferente habilitada para Okta, los datos de sesión de los datos del administrador pueden causar el error "Solicitud incorrecta". Si este error persiste incluso después de borrar la memoria caché del navegador local, intente validar el escenario de Tableau conectándose con un navegador diferente.

Otra causa del error "Solicitud incorrecta" es un error tipográfico en una de las muchas URL que especifique durante los procesos de configuración de Okta, Mellon y SAML. Compruebe que ha introducido todos estos sin error.

A menudo el archivo `error.log` en el servidor de puerta de enlace independiente especificará qué URL está causando el error.

No encontrada: la URL solicitada no se encontró en este servidor: este error indica uno de los muchos errores de configuración.

Si el usuario está autenticado con Okta y luego recibe este error, es probable que haya cargado la aplicación de autorización previa de Okta en Tableau Server cuando configuró SAML.

Compruebe que tiene los metadatos de la aplicación Okta de Tableau Server configurados en Tableau Server, y no los metadatos de la aplicación Okta de preautorización.

Otros pasos de solución de problemas:

- Revise la configuración de la aplicación de autorización previa de Okta. Asegúrese de que los protocolos HTTP vs HTTPS estén configurados como se especifica en este tema.
- Reinicie tsig-httpd en ambos servidores de puerta de enlace independiente.
- Compruebe que `sudo apachectl configtest` devuelve "Sintaxis OK" en ambas puertas de enlace independientes.
- Verifique que el usuario de prueba esté asignado a ambas aplicaciones en Okta.
- Compruebe que stickiess esté configurado en el equilibrador de carga y los grupos de destino asociados.

Comprobar la conexión de Tableau Server a la puerta de enlace independiente

Utilice el comando `wget` para verificar la conectividad y el acceso desde Tableau Server a la puerta de enlace independiente. Las variaciones de este comando pueden ayudarlo a comprender si los problemas de certificado están causando problemas de conexión.

Por ejemplo, ejecute este comando `wget` para verificar el protocolo de limpieza (HK) de Tableau Server:

```
wget https://ip-10-0-1-38.us-west-1.compute.internal:21319
```

Construya la URL con la misma dirección de host que incluyó para la opción de host del archivo tsig.json. Especifique el protocolo `https` y agregue la URL con el puerto HK `21319`.

Para verificar la conectividad e ignorar la verificación del certificado:

```
wget https://ip-10-0-1-38.us-west-1.compute.internal:21319 --no-check-certificate
```

Para verificar que el certificado de CA raíz para TSIG sea válido:

```
wget https://ip-10-0-1-38.us-west-1.compute.internal:21319 --ca-cer-
tificate=tsigRootCA.pem
```

Si Tableau puede comunicarse, es posible que aún obtenga errores relacionados con el contenido, pero no obtendrá errores relacionados con la conexión. Si Tableau no puede conectarse en absoluto, comience por verificar la configuración del protocolo en los grupos de firewall/seguridad. Por ejemplo, las reglas de entrada para el grupo de seguridad donde reside la puerta de enlace independiente deben permitir TCP 21319.

Apéndice: Caja de herramientas de implementación de AWS

Este tema incluye herramientas y opciones de implementación alternativas para la arquitectura de referencia cuando se implementa en AWS. Específicamente, este tema describe cómo automatizar la implementación de AWS de ejemplo que se describe en toda la EDG.

Script de instalación automatizada TabDeploy4EDG

El script TabDeploy4EDG automatiza la implementación de la implementación de Tableau de cuatro nodos que se describe en la Paso 4: Instalar y configurar Tableau Server. Si sigue el ejemplo de implementación de AWS que se describe en esta guía, es posible que pueda ejecutar el TabDeploy4EDG.

Requisitos. Para ejecutar el script, debe preparar y configurar el entorno de AWS de acuerdo con la implementación de ejemplo de la Parte 3: preparación para la implementación de Tableau Server Enterprise:

- La VPC, la subred y los grupos de seguridad se han configurado como se describe. Las direcciones IP no tienen que coincidir con las que se muestran en la implementación de ejemplo.
- Cuatro instancias EC2 que ejecutan las compilaciones más recientes y actualizadas de AWS Linux 2
- PostgreSQL está instalado y configurado como se describe en Instalar, configurar y convertir en .tar PostgreSQL .
- Un archivo de copia de seguridad tar del Paso 1 se encuentra en la instancia EC2 donde está instalado PostgreSQL, como se describe en Realizar una copia de seguridad de tar del paso 1 de PostgreSQL.
- La instancia EC2 que ejecutará el nodo 1 de la implementación de Tableau Server se ha configurado para comunicarse con PostgreSQL como se describe en Paso 4: Instalar y configurar Tableau Server.

- Ha iniciado sesión en cada instancia EC2 con una sesión SSH desde el host de Bastion.

El script tarda entre 1,5 y 2 horas en instalar y configurar los cuatro servidores de Tableau. El script configura Tableau según la configuración prescrita de la arquitectura de referencia. El script realiza las siguientes acciones:

- Restaura la copia de seguridad de la etapa 1 del host de PostgreSQL si especifica una ruta al archivo tar del host de PostgreSQL.
- Elimina las instalaciones de Tableau existentes en todos los nodos.
- Ejecuta `sudo yum update` en todos los nodos.
- Descarga y copia el paquete rpm de Tableau en cada nodo.
- Descarga e instala las dependencias de cada nodo.
- Crea `/app/tableau_server` e instala el paquete en todos los nodos.
- Instala el nodo 1 con un almacén de identidades local y configura un repositorio externo con PostgreSQL.
- Realiza la instalación del archivo de arranque y la configuración inicial del nodo 2-nodo 4.
- Elimina el archivo de arranque y el archivo de configuración de TabDeploy4EDG.
- Configura los servicios en el clúster de Tableau de acuerdo con las especificaciones de la arquitectura de referencia.
- Valida la instalación y devuelve el estado de cada nodo.

Descargar y copiar el script en el host de Bastion

1. Copie el script de la página de muestras TabDeploy4EDG y pegue el código en un archivo llamado `TabDeploy4EDG`.
2. Guarde el archivo en el directorio de inicio del host EC2 que actúa como host de Bastion.
3. Ejecute el siguiente comando para cambiar el modo en el archivo y hacerlo ejecutable:

```
sudo chmod +x TabDeploy4EDG
```

Ejecutar TabDeploy4EDG

TabDeploy4EDG debe ejecutarse desde el host de Bastion. El script se ha escrito asumiendo que se está ejecutando en el contexto del agente de reenvío ssh como se describe en

Ejemplo: conectarse al host de Bastion en AWS. Si no está ejecutando el contexto del agente de reenvío ssh, se le solicitarán contraseñas durante todo el proceso de instalación.

1. Cree, edite y guarde un archivo de registro (`registration.json`). El archivo debe ser un archivo json con el formato adecuado. Copie y personalice la siguiente plantilla:

   ```
   {
          "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. Ejecute el siguiente comando para generar un archivo de configuración de plantillas:

   ```
   ./TabDeploy4EDG -g edg.config
   ```

3. Abra el archivo de configuración para editar:

   ```
   sudo nano edg.config
   ```

   Como mínimo, debe agregar las direcciones IP de cada host EC2, una ruta de acceso al archivo de registro y una clave de licencia válida.

4. Cuando haya terminado de editar el archivo de configuración, guárdelo y luego ciérrelo.

5. Para ejecutar TabDeploy4EDG, ejecute el siguiente comando:

   ```
   ./TabDeploy4EDG -f edg.config
   ```

Ejemplo: Automatice la implementación de la infraestructura de AWS con Terraform

Esta sección describe cómo configurar y ejecutar Terraform para implementar la arquitectura de referencia de EDG en AWS. El ejemplo de configuración de Terraform que se presenta aquí implementa una AWS VPC con las subredes, los grupos de seguridad y las instancias EC2 que se describen en la Parte 3: preparación para la implementación de Tableau Server Enterprise.

Las plantillas de Terraform de muestra están disponibles en el sitio web de muestras de Tableau en https://help.tableau.com/samples/en-us/edg/edg-terraform.zip. Estas plantillas deben configurarse y personalizarse para su organización. El contenido de configuración proporcionado en esta sección describe los cambios de plantilla mínimos necesarios que deben personalizar para implementar.

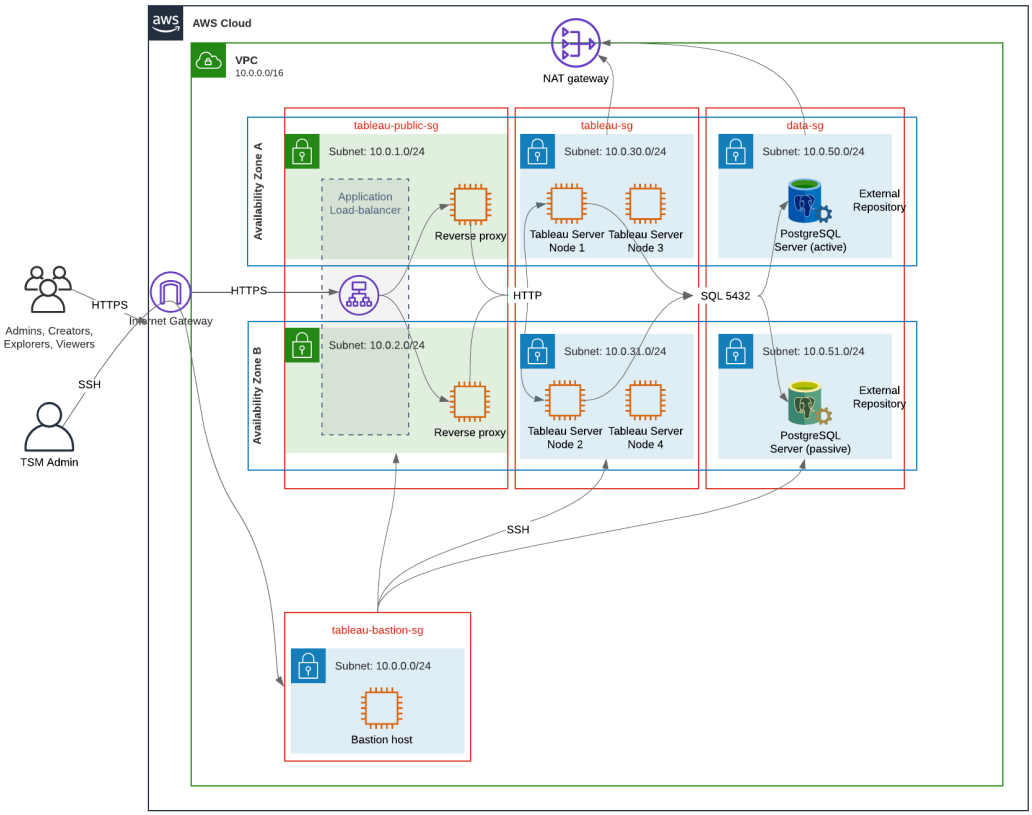
Meta

Las plantillas y el contenido de Terraform proporcionados aquí están destinados a proporcionar una muestra de trabajo que le permitirá implementar EDG rápidamente en un entorno de prueba de desarrollo.

Hemos hecho todo lo posible por probar y documentar la implementación de ejemplo de Terraform. Sin embargo, usar Terraform para implementar y mantener EDG en un entorno de producción requerirá experiencia en Terraform que está más allá del ámbito de este ejemplo. Tableau no brinda soporte para la solución Terraform de ejemplo que se documenta aquí.

Estado final

Siga el procedimiento de esta sección para configurar una VPC en AWS que sea funcionalmente equivalente a la VPC que se especifica en la Parte 3: preparación para la implementación de Tableau Server Enterprise.



Las plantillas de Terraform de muestra y el contenido de apoyo en esta sección:

- Crea una VPC con una dirección IP elástica, dos zonas de disponibilidad y una organización de subredes como se muestra arriba (las direcciones IP son diferentes)
- Crea los grupos de seguridad Bastion, Public, Private y Data.
- Establece la mayoría de las reglas de entrada y salida en los grupos de seguridad. Deberá editar los grupos de seguridad después de ejecutar Terraform.
- Crea los siguientes hosts EC2 (cada uno con AWS Linux2): bastion, proxy 1, proxy 2, nodo 1 de Tableau, nodo 2 de Tableau, nodo 3 de Tableau, nodo 4 de Tableau.
- No se crean hosts EC2 para PostgreSQL. Debe crear el EC2 manualmente en el grupo de seguridad de datos y luego instalar y configurar PostgreSQL como se describe en Instalar, configurar y convertir en .tar PostgreSQL.

Requisitos

- Cuenta de AWS: debe tener acceso a una cuenta de AWS que le permita crear VPC.
- Si está ejecutando Terraform desde un equipo con Windows, deberá instalar AWS CLI.
- Una dirección IP elástica disponible en su cuenta de AWS.
- Un dominio que está registrado en AWS Route 53. Terraform creará una zona DNS y certificados SSL relacionados en Route 53. Por lo tanto, el perfil con el que se ejecuta Terraform también debe tener los permisos adecuados en Route 53.

Antes de empezar

- Los ejemplos de línea de comandos en este procedimiento son para Terminal con Apple OS. Si está ejecutando Terraform en Windows, es posible que deba adaptar los comandos con las rutas de los archivos según corresponda.
- Un proyecto de Terraform se compone de muchos archivos de configuración de texto (extensión de archivo .tf). Configure Terraform personalizando estos archivos. Si no tiene un editor de texto fiable, instale Atom o Text++.
- Si está compartiendo el proyecto de Terraform con otras personas, le recomendamos almacenar el proyecto en Git para la gestión de cambios.

Paso 1: Preparar el entorno

A. Descargue e instale Terraform

https://www.terraform.io/downloads

B. Generar un par de claves públicas y privadas

Esta es la clave que utilizará para acceder a AWS y al entorno de VPC resultante. Cuando ejecute Terraform, incluirá la clave pública.

Abra Terminal y ejecute los siguientes comandos:

1. Create a private key. For example, `my-key.pem`:

   ```
   openssl genrsa -out my-key.pem 1024
   ```

2. Crear una clave pública. Este formato de clave no se utiliza para Terraform. La convertirá en una clave ssh más adelante en este procedimiento:

   ```
   openssl rsa -in my-key.pem -pubout > my-key.pub
   ```

3. Establecer permisos en la clave privada:

   ```
   sudo chmod 0600 my-key.pem
   ```

   Para establecer permisos en Windows:

   - Localice el archivo en el Explorador de Windows, haga clic derecho sobre él y luego seleccione **Propiedades**. Vaya a la pestaña **Seguridad** y luego haga clic en **Avanzado**.
   - Cambie el propietario por usted, deshabilite la herencia y elimine todos los permisos. Concédase **Control total** y luego haga clic en **Guardar**. Marque el archivo como de solo lectura.

4. Cree la clave pública ssh. Esta es la clave que copiará en Terraform más adelante en el proceso.

   ```
   ssh-keygen -y -f my-key.pem >my-key-ssh.pub
   ```

C. Descargar proyecto y agregar directorio de estado

1. Descargue y descomprima el proyecto EDG Terraform y guárdelo en su equipo local. Después de descomprimir la descarga, tendrá un directorio de nivel superior, edg-terraform y una serie de subdirectorios.

2. Cree un directorio llamado`state`, como un compañero del directorio de nivel superior `edg-terraform`.

Paso 2: Personalizar plantillas de Terraform

Debe personalizar las plantillas de Terraform para que se adapten a su entorno de AWS y EDG. El ejemplo aquí proporciona las personalizaciones de plantilla mínimas que la mayoría de las organizaciones necesitarán hacer. Es probable que su entorno particular requiera otras personalizaciones.

Esta sección está organizada por nombre de plantilla.

Asegúrese de guardar todos los cambios antes de continuar con el *Paso 3: Ejecutar 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. Abra la clave pública que generó en el Paso 1B y copie la clave:

   ```
   less my-key-ssh.pub
   ```

   Windows: Copie el contenido de su clave pública.

2. Copie la cadena de clave pública en el argumento public_key, por ejemplo:

   ```
   resource "aws_key_pair" "tableau" {
   key_name = "my-key"
   public_key = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQ (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. Agregue categorías según los requisitos de su organización. Por ejemplo:

```
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`.

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

Actualice el nombre de dominio raíz. Este nombre debe coincidir con el dominio que tiene registrado en Ruta 53.

```
variable "root_domain_name" {
 default = "example.com"
 }
```

De forma predeterminada, el subdominio `tableau` se especifica para el nombre de dominio DNS de la VPC. Para cambiar esto, actualice `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 es necesario, agregue categorías 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"
 }
}
```

- Según sea necesario, opcionalmente actualice su almacenamiento para manejar sus requisitos de datos:

  Volumen de la raíz:

  ```
  root_block_device {
   volume_size = 100
   volume_type = "gp3"
  }
  ```

  Volumen de aplicación:

  ```
  resource "aws_ebs_volume" "tableau" {
   availability_zone = data.aws_subnet.tableau.availability_zone
   size              = 500
   type              = "gp3"
  }
  ```

Paso 3: Ejecutar Terraform

A. Inicializar Terraform

En Terminal, cambie al directorio `edg-terraform` y ejecute el siguiente comando:

```
terraform init
```

Si la inicialización se realiza correctamente, continúe con el próximo paso. Si la inicialización falló, siga las instrucciones en la salida de Terraform.

B. Planear Terraform

Desde el mismo directorio, ejecute el comando del plan:

```
terraform plan
```

Este comando se puede ejecutar varias veces. Ejecute tantas veces como sea necesario para corregir errores. Cuando este comando se ejecute sin errores, continúe con el siguiente paso.

C. Aplicar Terraform

Desde el mismo directorio, ejecute el comando de aplicación:

```
terraform apply
```

Terraform will prompt you to verify deployment, type `Yes`.

Opcional: destruir Terraform

Puede destruir toda la VPC ejecutando el comando destroy:

```
terraform destroy
```

El comando destroy solo destruirá lo que ha creado. Si ha realizado cambios manuales en algunos objetos de AWS (es decir, grupos de seguridad, subredes, etc.), entonces el comando `destroy` fallará. Para salir de una operación de destrucción fallida/bloqueada, escriba`Control + C`. Luego, debe limpiar la VPC manualmente al estado en el que se encontraba cuando Terraform la creó originalmente. Después puede ejecutar el comando `destroy`.

Paso 4: Conéctese a bastion

Toda la conexión de la línea de comandos se realiza a través del host de bastion en TCP 22 (protocolo SSH).

1. En AWS, cree una regla de entrada en el grupo de seguridad de bastion (**AWS** > **Grupos de seguridad** > **Bastion SG** > **Editar reglas de entrada**) y cree una regla para permitir conexiones SSH (TCP 22) desde la dirección IP o la máscara de subred donde ejecutará los comandos de Terminal.

Opcional: Puede resultarle útil permitir la copia de archivos entre las instancias EC2 en los grupos Público y Privado durante la implementación. Crear reglas SSH de entrada:

- Privado: cree una regla de entrada para permitir SSH desde Público
- Público: cree una regla de entrada para permitir SSH desde Privado y desde Público

2. Use la clave pem que creó en el Paso 1.B para conectarse al host de bastion:

   En terminales Mac:

   Ejecute los siguientes comandos desde el directorio donde está almacenada la clave pem:

   `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`.

   En Windows usando PuTTY y Pageant:

   a. Cree ppk a partir de la clave pem: use el generador de claves PuTTY. Cargue la clave pem que ha creado en el paso 1.B. Después de importar la clave, haga clic en **Guardar clave privada**. Esto crea un archivo ppk.

   b. En PuTTY: abra la configuración y realice los siguientes cambios:

      - Sessions>Host Name: agregue la dirección IP del host de bastion.
      - Sessions>Port: 22
      - Connection>Data>Auto-login username: ec2-user
      - Connection>SSH>Auth>Allow agent forwarding
      - Connection>SSH>Auth> Para la clave privada, haga clic en Examinar y seleccione el archivo .ppk que acaba de crear.

   c. Instale Pageant y cargue el ppk en la aplicación.

Paso 5: Instalar PostgreSQL

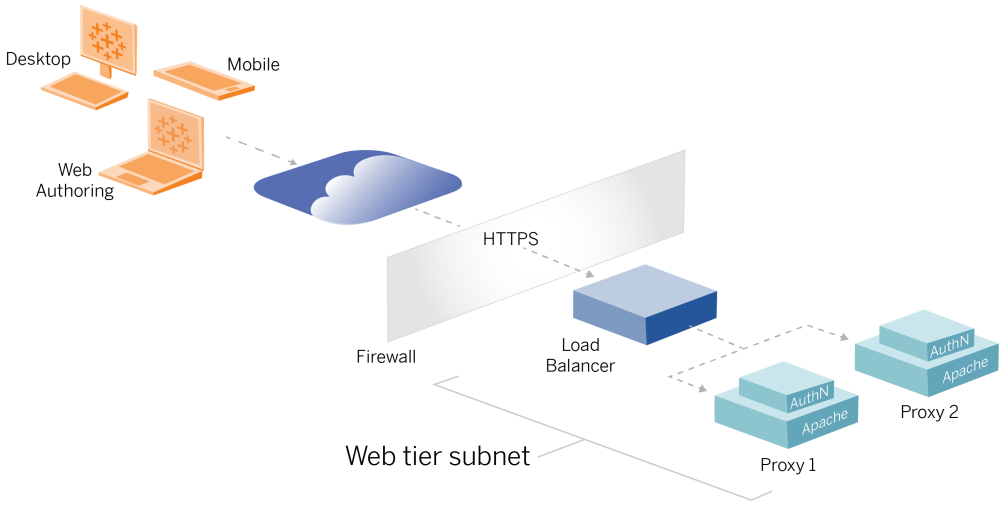
La plantilla de Terraform no instala PostgreSQL para su uso como repositorio externo. Sin embargo, se crea el grupo de seguridad y la subred asociados. Si va a instalar el repositorio externo en una instancia EC2 que ejecuta PostgreSQL, debe implementar la instancia EC2 como se describe en la Parte 3: preparación para la implementación de Tableau Server Enterprise.

Luego, instale, configure y haga una copia de seguridad tar de PostgreSQL como se describe en la Paso 4: Instalar y configurar Tableau Server.

Paso 6: (Opcional) Ejecute DeployTab4EDG

El script TabDeploy4EDG automatiza la implementación de Tableau de cuatro nodos que se describe en la Parte 4. Consulte Script de instalación automatizada TabDeploy4EDG.

Apéndice: nivel web con implementación de ejemplo de Apache



Este tema proporciona un procedimiento de un extremo a otro que describe cómo implementar el nivel web en el ejemplo de la arquitectura de referencia de AWS. La configuración de ejemplo tiene los siguientes componentes:

- Equilibrador de carga de aplicaciones de AWS
- Servidores proxy de Apache
- Módulo de autenticación Mellon
- IdP de Okta
- Autenticación SAML

Nota: El ejemplo de configuración de nivel web que se presenta en esta sección incluye procedimientos detallados para implementar software y servicios de terceros. Hemos hecho todo lo posible por verificar y documentar los procedimientos para habilitar el escenario del nivel web. Sin embargo, el software de terceros puede cambiar o su escenario puede diferir de la arquitectura de referencia que se describe aquí. Consulte la

documentación de terceros para obtener detalles de configuración autorizados y asistencia.

Los ejemplos de Linux a lo largo de esta sección muestran comandos para distribuciones similares a RHEL. Específicamente, los comandos aquí se han desarrollado con la distribución de Amazon Linux 2. Si está ejecutando distribuciones de Ubuntu, edite los comandos según corresponda.

La implementación del nivel web en este ejemplo sigue un procedimiento de verificación y configuración paso a paso. La configuración del nivel web principal consta de los siguientes pasos para habilitar HTTP entre Tableau e Internet. Apache se ejecuta y configura para proxy inverso/equilibrio de carga detrás del equilibrador de carga de la aplicación AWS:

1. Instalar Apache
2. Configurar el proxy inverso para probar la conectividad con Tableau Server
3. Configurar el equilibrio de carga en el proxy
4. Configurar el equilibrador de carga de aplicaciones de AWS

Una vez que se configura el nivel web y se verifica la conectividad con Tableau, debe configurar la autenticación con un proveedor externo.

Instalar Apache

Ejecute el siguiente procedimiento en ambos hosts EC2 (Proxy 1 y Proxy 2). Si está implementando en AWS de acuerdo con el ejemplo de arquitectura de referencia, debe tener dos zonas de disponibilidad y ejecutar un solo servidor proxy en cada zona de disponibilidad.

1. Instale Apache:

   ```
   sudo yum update -y
   sudo yum install -y httpd
   ```

2. Configuración para iniciar Apache al reiniciar:

```
sudo systemctl enable --now httpd
```

3. Verifique que la versión de httpd que ha instalado incluya `proxy_hcheck_module`.

   ```
   sudo httpd -M
   ```

   Se necesita `proxy_hcheck_module`. Si su versión de httpd no incluye este módulo, actualice a una versión de httpd que sí lo incluya.

Configurar el proxy para probar la conectividad a Tableau Server

Ejecute este procedimiento en uno de los hosts proxy (Proxy 1). El propósito de este paso es verificar la conectividad entre Internet, su servidor proxy y Tableau Server en el grupo de seguridad privado.

1. Cree un archivo llamado`tableau.conf` y agréguelo al directorio`/etc/h-ttpd/conf.d`.

   Copie el siguiente código y especifique las claves`ProxyPass` y`ProxyPassReverse` con la dirección IP privada del Nodo 1 de Tableau Server.

   > **Importante**: La configuración que se muestra a continuación no es segura y no debe usarse en producción. Esta configuración solo debe usarse durante el proceso de instalación para verificar la conectividad de un extremo a otro.

   Por ejemplo, si la dirección IP privada del Nodo 1 es`10.0.30.32` , el contenido del archivo `tableau.conf` sería:

   ```
   <VirtualHost *:80>
   ProxyPreserveHost On
   ProxyPass "/" "http://10.0.30.32:80/"
   ```

```
ProxyPassReverse "/" "http://10.0.30.32:80/"
</VirtualHost>
```

2. Reinicie httpd:

```
sudo systemctl restart httpd
```

Verificación: configuración de topología base

Debería poder acceder a la página de administración de Tableau Server desde `http://<-proxy-public-IP-address>`.

Si la página de inicio de sesión de Tableau Server no se carga en su navegador, siga estos pasos de solución de problemas en el host Proxy 1:

- Detenga y luego inicie httpd como primer paso de solución de problemas.
- Compruebe el archivo `tableau.conf`. Verifique que la IP privada del Nodo 1 sea correcta. Verifique las comillas dobles y revise cuidadosamente la sintaxis.
- Ejecute el comando `curl` en el servidor del proxy inverso con la dirección IP privada del nodo 1, por ejemplo, `curl 10.0.1.90`. Si el shell no devuelve html, o si devuelve html para la página web de prueba de Apache, verifique la configuración del puerto/protocolo entre los grupos de seguridad Público y Privado.
- Ejecute el comando `curl` con la dirección IP privada del proxy 1, por ejemplo, `curl 10.0.0.163`. Si el shell devuelve el código html para la página web de prueba de Apache, entonces el archivo proxy no está configurado correctamente.
- Reinicie siempre httpd (`sudo systemctl restart httpd`) después de cualquier cambio de configuración en el archivo proxy o en los grupos de seguridad.
- Asegúrese de que TSM se esté ejecutando en el nodo 1.

Configurar el equilibrio de carga en el proxy

1. En el mismo host de proxy (Proxy 1) donde creó `tableau.conf`, elimine la configuración del host virtual existente y edite el archivo para incluir la lógica de equilibrio de carga.

   Por ejemplo:

```
<VirtualHost *:80>
ServerAdmin admin@example.com
#Load balancing logic.
ProxyHCExpr 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. Detenga y luego inicie httpd:

```
sudo systemctl stop httpd
sudo systemctl start httpd
```

3. Verifique la configuración navegando a la dirección IP pública de Proxy 1.

Copiar la configuración al segundo servidor proxy

1. Copie el archivo `tableau.conf` de Proxy 1 y guárdelo en el directorio `/etc/h-ttpd/conf.d` en el host Proxy 2.

2. Detenga y luego inicie httpd:

```
sudo systemctl stop httpd
sudo systemctl start httpd
```

3. Verifique la configuración navegando a la dirección IP pública de Proxy 2.

Configurar el equilibrador de carga de aplicaciones de AWS

Configure el equilibrador de carga como una escucha HTTP. Este procedimiento describe cómo agregar un equilibrador de carga en AWS.

Paso 1: crear un grupo de destino

Un grupo de destino es una configuración de AWS que define las instancias EC2 que ejecutan sus servidores proxy. Estos son los destinos del tráfico de LBS.

1. EC2>**Grupos de destino** > **Crear grupo de destino**
2. En la página Crear:
   - Escriba un nombre de grupo de destino, `TG-internal-HTTP` por ejemplo
   - Tipo de destino: instancias
   - Protocolo: HTTP
   - Puerto: 80
   - VPC: seleccione su VPC
   - En **Comprobaciones de estado** > **Configuración de comprobaciones de estado avanzadas** > **Códigos de éxito**, agregue la lista de códigos para leer: `200,303`.
   - Haga clic en **Crear**
3. Seleccione el grupo de destino que acaba de crear y luego haga clic en la pestaña **Destinos**:

- Haga clic en **Editar**.
- Seleccione las instancias EC2 (o instancia única si está configurando una a la vez) que ejecutan la aplicación de proxy y luego haga clic en **Agregar a registrado**.
- Haga clic en **Guardar**.

Paso 2: iniciar el asistente del equilibrador de carga

1. EC2> **Equilibradores de carga** > **Crear equilibrador de carga**
2. En la página "Seleccionar tipo de equilibrador de carga", cree un equilibrador de carga de aplicaciones.

Nota: La interfaz de usuario que se muestra para configurar el equilibrador de carga no es coherente en los centros de datos de AWS. El procedimiento siguiente, "Configuración del asistente", se asigna al asistente de configuración de AWS que comienza con el **Paso 1: configurar el equilibrador de carga**.

Si su centro de datos muestra todas las configuraciones en una sola página que incluye un botón **Crear equilibrador de carga** en la parte inferior de la página, siga el procedimiento de "Configuración de una sola página" a continuación.

Configuración del asistente

1. Página **configurar equilibrador de carga**:
   - Especifique el nombre
   - Esquema: orientado a Internet (predeterminado)
   - Tipo de dirección IP: ipv4 (predeterminado)
   - Oyentes (oyentes y enrutamiento):
     a. Deje el oyente HTTP predeterminado
     b. Haga clic en **Agregar oyente** y agregue `HTTPS:443`

- VPC: seleccione la VPC donde instaló todo
- Zonas de disponibilidad:
  - Seleccione **a** y **b** para las regiones de su centro de datos
  - En cada selector del menú desplegable correspondiente, seleccione la subred pública (donde residen sus servidores proxy).
- Haga clic en **Establecer configuración de seguridad**

2. Página **Configurar ajustes de seguridad**
   - Suba su certificado SSL público.
   - Haga clic en **Siguiente: Configurar grupos de seguridad**.
3. Página **Configurar grupos de seguridad**:
   - Seleccione el grupo de seguridad público. Si se selecciona el grupo de seguridad predeterminado, borre esa selección.
   - Haga clic en **Siguiente: Configurar el enrutamiento**.
4. Página **Configurar enrutamiento**
   - Grupo de destino: Grupo de destino existente.
   - Nombre: seleccione el grupo de destino que creó anteriormente
   - Haga clic en **Siguiente: Registrar destinos**.
5. Página **Registrar destinos**
   - Deben mostrarse las dos instancias de servidor proxy que configuró anteriormente.
   - Haga clic en **Siguiente: Revisión**.
6. Página **Revisión**

   Haga clic en **Crear**.

Configuración de una sola página

Configuración básica

- Especifique el nombre
- Esquema: orientado a Internet (predeterminado)
- Tipo de dirección IP: ipv4 (predeterminado)

Mapeo de redes

- VPC: seleccione la VPC donde instaló todo
- Mapeo:
  - Seleccione las zonas de disponibilidad **a** y **b** (o comparables) para sus regiones de centros de datos
  - En cada selector del menú desplegable correspondiente, seleccione la subred pública (donde residen sus servidores proxy).

Grupos de seguridad

Seleccione el grupo de seguridad público. Si se selecciona el grupo de seguridad predeterminado, borre esa selección.

Oyentes y enrutamiento

- Deje el oyente HTTP predeterminado. Para la **acción predeterminada**, especifique el grupo de destino que configuró anteriormente.
- Haga clic en **Agregar oyente** y agregue `HTTPS:443`. Para la **acción predeterminada**, especifique el grupo de destino que configuró anteriormente.

Configuración de oyente seguro

- Suba su certificado SSL público.

Haga clic en **Crear el equilibrador de carga**.

Paso 3: habilitar la adherencia

1. Una vez creado el equilibrador de carga, debe habilitar la adherencia en el grupo de destino.
   - Abra la página del grupo de destino de AWS (**EC2**> **Equilibradores de carga** > **Grupos de destino**), seleccione la instancia del grupo de destino que acaba de configurar. En el menú **Acciones**, seleccione **Editar atributos**.

- En la página **Editar atributos**, seleccione **Adherencia**, especifique una duración de `1 day` y luego **Guardar cambios**.

2. En el equilibrador de carga, habilite la adherencia en el oyente HTTP. Seleccione el equilibrador de carga que acaba de configurar y luego haga clic en la pestaña **Oyentes**:
   - Para **HTTP:80**, haga clic en **Ver/editar reglas**. En la página **Reglas** resultante, haga clic en el icono de edición (una vez en la parte superior de la página y luego nuevamente por la regla) para editar la regla. Elimine la regla THEN existente y reemplácela haciendo clic en **Agregar acción** > **Reenviar a...** En la configuración THEN resultante, especifique el mismo grupo de destino que ha creado. En Adherencia a nivel de grupo, habilite la adherencia y establezca la duración en 1 día. Guarde la configuración y haga clic en **Actualizar**.

Paso 4: Establezca el tiempo de espera inactivo en el equilibrador de carga

En el equilibrador de carga, actualice el tiempo de espera inactivo a 400 segundos.

Seleccione el equilibrador de carga que ha configurado para esta implementación y luego haga clic en **Acciones** > **Editar atributos**. Establezca el tiempo de **espera inactivo** en`400` segundos y luego haga clic en **Guardar**.

Paso 5: Verificar la conectividad LBS

Abra la página del equilibrador de carga de AWS (**EC2** > **Equilibradores de carga**), seleccione la instancia del equilibrador de carga que acaba de configurar.

En **Descripción**, copie el nombre del DNS y péguelo en un navegador para acceder a la página de inicio de sesión de Tableau Server.

Si aparece el error 500, probablemente necesite reiniciar sus servidores proxy.

Actualizar DNS con URL pública de Tableau

Utilice el nombre de la zona DNS de su dominio de la descripción del equilibrador de carga de AWS para crear un valor CNAME en su DNS. El tráfico a su URL (tableau.example.com) debe

enviarse al nombre de DNS público de AWS.

Verificar la conectividad

Una vez finalizadas las actualizaciones de DNS, debería poder navegar a la página de inicio de sesión de Tableau Server con su URL pública, por ejemplo, `http-s://tableau.example.com`.

Ejemplo de configuración de autenticación: SAML con IdP externo

El siguiente ejemplo describe cómo instalar y configurar SAML con Okta IdP y el módulo de autenticación Mellon para una implementación de Tableau que se ejecuta en la arquitectura de referencia de AWS. El ejemplo describe cómo configurar Tableau Server y los servidores proxy Apache para usar HTTP. Okta enviará la solicitud al equilibrador de carga de AWS a través de HTTPS, pero todo el tráfico interno viajará a través de HTTP. Mientras configura para este escenario, tenga en cuenta los protocolos HTTP y HTTPS al configurar cadenas de URL.

Este ejemplo utiliza Mellon como un módulo de proveedor de servicios de autenticación previa en los servidores proxy inversos. Esta configuración garantiza que solo el tráfico autenticado se conecte a Tableau Server, que también actúa como proveedor de servicios con Okta IdP. Por lo tanto, debe configurar dos aplicaciones IdP: una para el proveedor de servicios Mellon y otra para el proveedor de servicios Tableau.

Crear la cuenta de administrador de Tableau

Un error común al configurar SAML es olvidarse de crear una cuenta de administrador en Tableau Server antes de habilitar SSO.

El primer paso es crear una cuenta en Tableau Server con un rol de administrador del servidor. En el caso de ejemplo de Okta, el nombre de usuario debe tener el formato de una

dirección de correo electrónico válida, por ejemplo, usuario@ejemplo.com. Debe establecer una contraseña para este usuario, pero la contraseña no se utilizará después de configurar SAML.

Configurar la aplicación de autorización previa de Okta

El escenario de un extremo a otro descrito en esta sección requiere dos aplicaciones Okta:

- Solicitud de autorización previa de Okta
- Aplicación Okta de Tableau Server

Cada una de estas aplicaciones está asociada con diferentes metadatos que deberá configurar en el proxy inverso y Tableau Server, respectivamente.

Este procedimiento describe cómo crear y configurar la aplicación de autorización previa de Okta. Más adelante en este tema, creará la aplicación Okta de Tableau Server. Para obtener una cuenta de Okta de prueba gratuita con usuarios limitados, consulte la página web de desarrolladores de Okta.

Cree una integración de la aplicación SAML para el proveedor de servicios de autenticación previa de Mellon.

1. Abra el panel de administración de Okta > **Aplicaciones** > **Crear integración de aplicaciones**.
2. En la página **Crear una nueva integración de aplicaciones**, seleccione **SAML 2.0** y luego haga clic en **Siguiente**.
3. En la pestaña **Configuración general**, escriba un nombre de aplicación, como `Tableau Pre-Auth` y haga clic en **Siguiente**.
4. En la pestaña **Configurar SAML**:
    - URL de inicio de sesión único (SSO). El elemento final de la ruta en la URL de inicio de sesión único se denomina `MellonEndpointPath` en el archivo de configuración `mellon.conf` que aparece más adelante en este procedimiento.

Puede especificar cualquier punto final que desee. En este ejemplo, `sso` es el punto final. El último elemento, `postResponse`, es obligatorio:`https://tableau.example.com/sso/postResponse`.

- Desmarque la casilla de verificación: **Usar esto para la URL del destinatario y la URL de destino**.
- URL del destinatario: igual que la URL de SSO, pero con HTTP. Por ejemplo, `http://tableau.example.com/sso/postResponse`.
- URL de destino: igual que la URL del SSO, pero con HTTP. Por ejemplo, `http://tableau.example.com/sso/postResponse`.
- URI de audiencia (ID de entidad SP). Por ejemplo, `https://tableau.example.com`.
- Formato del ID de nombre: `EmailAddress`
- Nombre de usuario de la aplicación: `Email`
- Declaraciones de atributos: Nombre = `mail`; Formato de nombre = `Unspecified`; Valor = `user.email`.

Haga clic en **Siguiente**.

5. En la pestaña **Comentarios**, seleccione:
   - **Soy un cliente de Okta que agrega una aplicación interna**
   - **Esta es una aplicación interna que hemos creado**
   - Haga clic en **Finalizar**.

6. Cree el archivo de metadatos de IdP previo a la autenticación:
   - En Okta: **Aplicaciones** > **Aplicaciones** > Su nueva aplicación (p. ej., `Tableau Pre-Auth`)> **Iniciar sesión**
   - Junto a **Certificados de firma de SAML**, haga clic en **Ver instrucciones de configuración de SAML**.
   - En la página **Cómo configurar SAML 2.0 para la aplicación <pre-auth>**, desplácese hacia abajo hasta la sección **Opcional**, **Proporcionar los siguientes metadatos de IDP a su proveedor de SP** .
   - Copie el contenido del campo XML y guárdelo en un archivo llamado `pre-auth_idp_metadata.xml`.

7. (Opcional) Configure la autenticación multifactor:

- En Okta: **Aplicaciones** > **Aplicaciones** > Su nueva aplicación (p. ej., `Tableau Pre-Auth`)> **Iniciar sesión**
- **En Directiva de inicio de sesión**, haga clic en **Agregar regla**.
- En la **regla de inicio de sesión de la aplicación**, especifique un nombre y las diferentes opciones de MFA. Para probar la funcionalidad, puede dejar todas las opciones predeterminadas. Sin embargo, en **Acciones**, debe seleccionar **Solicitar factor** y luego especificar la frecuencia con la que los usuarios deben iniciar sesión. Haga clic en **Guardar**.

Crear y asignar un usuario de Okta

1. En Okta, cree un usuario con el mismo nombre de usuario que creó en Tableau (usuario@ejemplo.com): **Directorio** > **Personas** > **Agregar persona**.
2. Una vez creado el usuario, asigne la nueva aplicación Okta a esa persona: haga clic en el nombre de usuario y luego asigne la aplicación en **Asignar aplicación**.

Instalar Mellon para preautorización

1. En las instancias EC2 que ejecuta el servidor proxy Apache, ejecute los siguientes comandos para instalar los módulos PHP y Mellon:

   ```
   sudo yum install httpd php mod_auth_mellon
   ```

2. Cree el directorio `/etc/httpd/mellon`

Configurar Mellon como módulo de preautorización

Ejecute este procedimiento en ambos servidores proxy.

Debe tener una copia de `pre-auth_idp_metadata.xml` que creó a partir de la configuración de Okta.

1. Cambie el directorio:

   ```
   cd /etc/httpd/mellon
   ```

2. Cree los metadatos del proveedor de servicios. Ejecute el script `mellon_create_metadata.sh`. Debe incluir el ID de entidad y la URL de retorno de su organización en el comando.

   La URL de retorno se denomina *URL de inicio de sesión único* en Okta. El elemento final de la ruta en la URL de retorno se denomina `MellonEndpointPath` en el archivo de configuración `mellon.conf` que aparece más adelante en este procedimiento. En este ejemplo, especificamos `sso` como la ruta del punto final.

   Por ejemplo:

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

   El script devuelve el certificado del proveedor de servicios, la clave y los archivos de metadatos.

3. Cambie el nombre de los archivos del proveedor de servicios en el directorio `mellon` para facilitar la lectura. Nos referiremos a estos archivos por los siguientes nombres en la documentación:

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

4. Copie el archivo `pre-auth_idp_metadata.xml` en el mismo directorio.

5. Cree el archivo `mellon.conf` en el directorio `/etc/httpd/conf.d`:

   ```
   sudo nano /etc/httpd/conf.d/mellon.conf
   ```

6. Copie el siguiente contenido en `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_meta-
data.xml
MellonEndpointPath /sso
MellonEnable "info"
</Location>
```

7. Añada el siguiente contenido en el archivo `tableau.conf` existente:

   Dentro del bloque `<VirtualHost *:80>`, agregue el siguiente contenido. Actualice `ServerName` con el nombre de host público en su ID de entidad:

   ```
   DocumentRoot /var/www/html
   ServerName tableau.example.com
   ServerSignature Off
   ErrorLog logs/error_sp.log
   CustomLog logs/access_sp.log combined
   LogLevel info
   ```

   Agregue el bloque Ubicación fuera del bloque `<VirtualHost *:80>`. Actualice `MellonCookieDomain` con el dominio de nivel superior para conservar la información de las cookies como se muestra:

   ```
   <Location />
   AuthType Mellon
   MellonEnable auth
   Require valid-user
   MellonCookieDomain example.com
   </Location>
   ```

   El archivo `tableau.conf` completo debería parecerse al siguiente ejemplo:

   ```
   <VirtualHost *:80>
   ServerAdmin admin@example.com
   ```

```
ProxyHCExpr 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. Verifique la configuración. Ejecute el comando siguiente:

   ```
   sudo apachectl configtest
   ```

   Si la prueba de configuración devuelve un error, corrija los errores y vuelva a ejecutar configtest. Si la configuración es correcta, aparecerá `Syntax OK`.

9. Reinicie httpd:

```
sudo systemctl restart httpd
```

Crear la aplicación de Tableau Server en Okta

1. En el panel de Okta: **Aplicaciones** > **Aplicaciones** > **Examinar catálogo de aplicaciones**
2. En **Examinar catálogo de integración de aplicaciones**, busque `Tableau`, seleccione la miniatura de Tableau Server y luego haga clic en **Agregar**.
3. En **Agregar Tableau Server** > **Configuración general**, especifique una Etiqueta y luego haga clic en **Siguiente**.
4. En Opciones de inicio de sesión, seleccione **SAML 2.0** y,después, desplácese hacia abajo hasta Configuración avanzada de inicio de sesión:
    - **ID de entidad SAML**: especifique la URL pública, por ejemplo, https://tableau.example.com.
    - **Formato de nombre de usuario de la aplicación**: correo electrónico
5. Haga clic en el enlace de **metadatos del proveedor de identidad** para iniciar un navegador. Copie el enlace del navegador. Este es el enlace que utilizará cuando configure Tableau en el procedimiento que se muestra a continuación.
6. Haga clic en **Realizado**.
7. Asigne la nueva aplicación Okta de Tableau Server a su usuario (usuario@example.com): haga clic en el nombre de usuario y luego asigne la aplicación en **Asignar aplicación**.

Habilitar SAML en Tableau Server para IdP

Ejecute este procedimiento en el nodo 1 de Tableau Server.

1. Descargue los metadatos de la aplicación de Tableau Server de Okta. Utilice el enlace que guardó del procedimiento anterior:

    ```
    wget https://dev-66144217.o-
    kta.com/app/exk1egxgt1fhjkSeS5d7/sso/saml/metadata -O idp_meta-
    data.xml
    ```

2. Copie un certificado TLS y un archivo de clave relacionado en Tableau Server. El archivo de claves debe ser una clave RSA. Para obtener más información sobre la configuración y los requisitos de SAML, consulte *Requisitos de SAML* (Linux).

   Para simplificar la administración y la implementación de certificados, y como práctica recomendada de seguridad, recomendamos usar certificados generados por una autoridad de certificación (CA) de terceros de confianza. Como alternativa, puede generar certificados autofirmados o usar certificados de una PKI para TLS.

   Si no tiene un certificado TLS, puede generar un certificado autofirmado mediante el procedimiento integrado a continuación.

Generar un certificado autofirmado

Ejecute este procedimiento en el nodo 1 de Tableau Server.

a. Genere la clave de la autoridad certificadora (CA) raíz de firma:

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

b. Cree el certificado de CA raíz:

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

Se le pedirá que especifique los valores para los campos del certificado. Por ejemplo:

```
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. Cree el certificado y la clave relacionada (`server-saml.csr` y `server-saml.key` en el ejemplo siguiente). El nombre del sujeto del certificado debe coincidir con el nombre del host público del host de Tableau. El nombre del sujeto se establece con la opción `-subj` con el formato `"/CN=<host-name>"`, por ejemplo:

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

d. Firme el nuevo certificado con el certificado CA que creó anteriormente. El siguiente comando también genera el certificado en el formato `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. Convierta el archivo de clave a RSA. Tableau requiere un archivo de clave RSA para SAML. Para convertirlo, ejecute el siguiente comando:

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

3. Configure SAML: Ejecute el siguiente comando, especificando su ID de entidad y URL de retorno, y las rutas al archivo de metadatos, archivo de certificado y archivo de clave:

```
tsm authentication saml configure --idp-entity-id "http-
s://tableau.example.com" --idp-return-url "http-
s://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 su organización ejecuta Tableau Desktop 2021.4 o posterior, debe ejecutar el siguiente comando para habilitar la autenticación a través de los servidores proxy inversos.

   Las versiones de Tableau Desktop 2021.2.1 - 2021.3 funcionarán sin ejecutar este comando, siempre que su módulo de autenticación previa (p. ej., Mellon) esté configurado para permitir la conservación de cookies de dominio de nivel superior.

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

5. Aplique los cambios de configuración:

   ```
   tsm pending-changes apply
   ```

Validar la funcionalidad SAML

Para validar la funcionalidad SAML de un extremo a otro, inicie sesión en Tableau Server con la URL pública (p. ej., https://tableau.example.com) con la cuenta de administrador de Tableau que creó al comienzo de este procedimiento.

Solución de problemas de validación

Solicitud incorrecta: un error común para este escenario es un error de "Solicitud incorrecta" de Okta. A menudo, este problema ocurre cuando el navegador está almacenando en caché datos de la sesión anterior de Okta. Por ejemplo, si administra las aplicaciones de Okta como administrador de Okta y luego intenta acceder a Tableau con una cuenta diferente habilitada para Okta, los datos de sesión de los datos del administrador pueden causar el error "Solicitud incorrecta". Si este error persiste incluso después de borrar la memoria caché del navegador local, intente validar el escenario de Tableau conectándose con un navegador diferente.

Otra causa del error "Solicitud incorrecta" es un error tipográfico en una de las muchas URL que especifique durante los procesos de configuración de Okta, Mellon y SAML. Revise todo esto cuidadosamente.

A menudo, el archivo httpd `error.log` en el servidor Apache especificará qué URL está causando el error.

No encontrada: la URL solicitada no se encontró en este servidor: este error indica uno de los muchos errores de configuración.

Si el usuario está autenticado con Okta y luego recibe este error, es probable que haya cargado la aplicación de autorización previa de Okta en Tableau Server cuando configuró SAML. Compruebe que tiene los metadatos de la aplicación Okta de Tableau Server configurados en Tableau Server, y no los metadatos de la aplicación Okta de preautorización.

Otros pasos de solución de problemas:

- Revise `tableau.conf` con cuidado en busca de errores tipográficos o de configuración
- Revise la configuración de la aplicación de autorización previa de Okta. Asegúrese de que los protocolos HTTP vs HTTPS estén configurados como se especifica en este tema.
- Reinicie httpd en ambos servidores proxy.
- Compruebe que `sudo apachectl configtest` devuelve "Sintaxis OK" en ambos servidores proxy.
- Verifique que el usuario de prueba esté asignado a ambas aplicaciones en Okta.
- Compruebe que stickiess esté configurado en el equilibrador de carga y los grupos de destino asociados

Configurar SSL/TLS desde el equilibrador de carga a Tableau Server

Algunas organizaciones requieren un canal de cifrado de un extremo a otro desde el cliente hasta el servicio de back-end. La arquitectura de referencia predeterminada, como se describe hasta este punto, especifica SSL desde el cliente hasta el equilibrador de carga que se ejecuta en el nivel web de su organización.

Para configurar SSL desde el equilibrador de carga a Tableau Server, debe:

- Instalar un certificado SSL válido tanto en Tableau como en los servidores proxy.
- Configurar SSL desde el equilibrador de carga a los servidores proxy inversos.
- Configurar SSL desde los servidores proxy a Tableau Server.
- También puede configurar SSL desde Tableau Server a la instancia de PostgreSQL.

El resto de este tema describe esta implementación en el contexto de la arquitectura de referencia de AWS de ejemplo.

Ejemplo: Configurar SSL/TLS en la arquitectura de referencia de AWS

Esta sección describe cómo configurar SSL en Tableau y configurar SSL en un servidor proxy Apache, todo en la arquitectura de referencia de AWS de ejemplo.

Los procedimientos de Linux a lo largo de este ejemplo muestran comandos para distribuciones similares a RHEL. Específicamente, los comandos aquí se han desarrollado con la distribución de Amazon Linux 2. Si está ejecutando distribuciones de Ubuntu, edite los comandos según corresponda.

Paso 1: Recopilar certificados y claves relacionadas

Para simplificar la administración y la implementación de certificados, y como práctica recomendada de seguridad, recomendamos usar certificados generados por una autoridad de certificación (CA) de terceros de confianza.

Como alternativa, puede generar certificados autofirmados o usar certificados de una PKI para TLS.

El siguiente procedimiento de cómo generar certificados autofirmados. Si está utilizando certificados de terceros como recomendamos, puede omitir este procedimiento.

Ejecute este procedimiento en uno de los hosts proxy. Después de generar el certificado y la clave asociada, lo compartirá con el otro host proxy y con el nodo 1 de Tableau Server.

1. Genere la clave de la autoridad certificadora (CA) raíz de firma:

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

2. Cree el certificado de CA raíz:

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

   Se le pedirá que especifique los valores para los campos del certificado. Por ejemplo:

   ```
   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. Cree el certificado y la clave relacionada (`serverssl.csr` y `serverssl.key` en el ejemplo siguiente). El nombre del sujeto del certificado debe coincidir con el nombre del host público del host de Tableau. El nombre del sujeto se establece con la opción `-subj` con el formato `"/CN=<host-name>"`, por ejemplo:

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

4. Firme el nuevo certificado con el certificado CA que creó en el paso 2. El siguiente comando también genera el certificado en el formato `crt`:

   ```
   openssl x509 -req -in serverssl.csr -days 3650 -CA rootCACer-
   t.pem -CAkey rootCAKey.pem -CAcreateserial -out serverssl.crt
   ```

Paso 2: configurar el servidor proxy para SSL

Ejecute este procedimiento en ambos servidores proxy.

1. Instale el módulo ssl de Apache:

   ```
   sudo yum install mod_ssl
   ```

2. Cree el directorio `/etc/ssl/private`:

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

3. Copie los archivos .crt y .key en las siguientes rutas `/etc/ssl/`:

   ```
   sudo cp serverssl.crt /etc/ssl/certs/

   sudo cp serverssl.key /etc/ssl/private/
   ```

4. Realice los siguientes cambios en el `tableau.conf`:

   - Agregue el bloque de reescritura de SSL:

     ```
     RewriteEngine on
     RewriteCond %{SERVER_NAME} =tableau.example.com
     RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI}
     [END,NE,R=permanent]
     ```

   - En el bloque de reescritura de SSL, actualice el nombre del servidor `RewriteCond`: agregue su nombre de host público, por ejemplo, `tableau.example.com`
   - Cambie `<VirtualHost *:80>` por `<VirtualHost *:443>`.
   - Envuelva el `<VirtualHost *:443>` y los bloques `<Location />` con `<IfModule mod_ssl.c>... </IfModule>`.
   - `BalancerMember`: cambie el protocolo de `http` a `https`.
   - Agregue elementos `SSL*` dentro del bloque `<VirtualHost *:443>`:

     ```
     SSLEngine on
     SSLCertificateFile /etc/ssl/certs/serverssl.crt
     SSLCertificateKeyFile /etc/ssl/private/serverssl.key
     SSLProxyEngine on
     SSLProxyVerify none
     ```

```
SSLProxyCheckPeerName off
SSLProxyCheckPeerExpire off
```

- En el elemento `LogLevel`, agregue `ssl:warn`.
- Opcional: Si ha instalado y configurado un módulo de autenticación, es posible que tenga elementos adicionales en el archivo tableau.conf. Por ejemplo, el bloque `<Location /> </Location>` incluirá elementos.

Aquí se muestra un archivo tableau.conf de ejemplo configurado para SSL:

```
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
ProxyHCExpr 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. Agregue el archivo index.html para eliminar los errores 403:

   ```
   sudo touch /var/www/html/index.html
   ```

6. Reinicie httpd:

   ```
   sudo systemctl restart httpd
   ```

Paso 3: configurar Tableau Server para SSL externo

Copie los archivos serverssl.crt y serverssl.key del host del Proxy 1 al Tableau Server inicial (Nodo 1).

Ejecute los siguientes comandos en el nodo 1:

```
tsm security external-ssl enable --cert-file serverssl.crt --key-
file serverssl.key
tsm pending-changes apply
```

Paso 4: configuración de autenticación opcional

Si ha configurado un proveedor de identidad externo para Tableau, es probable que deba actualizar las URL de retorno en el panel administrativo del IdP.

Por ejemplo, si está utilizando una aplicación de preautenticación Okta, deberá actualizar la aplicación para usar el protocolo HTTPS para la URL del destinatario y la URL de destino.

Paso 5: configurar el equilibrador de carga de AWS para HTTPS

Si está implementando con el equilibrador de carga de AWS como se documenta en esta guía, vuelva a configurar el equilibrador de carga de AWS para enviar tráfico HTTPS a los servidores proxy:

1. Anule el registro del grupo de destino HTTP existente:

   En **Grupos de destino**, seleccione el grupo de destino HTTP que se ha configurado para el equilibrador de carga, haga clic en **Acciones** y, a continuación, haga clic en **Registrar y anular el registro de la instancia**.

   En la página **Registrar y anular el registro de destinos**, seleccione las instancias que están configuradas actualmente, haga clic en **Anular registro** y luego en **Guardar**.

2. Cree un grupo de destino HTTPS:

   Grupos de destino > **Crear grupo de destino**

   - Seleccione "Instancias"
   - Escriba un nombre de grupo de destino, `TG-internal-HTTPS` por ejemplo
   - Seleccione su VPC
   - Protocolo: HTTPS 443
   - En **Comprobaciones de estado** > **Configuración de comprobaciones de estado avanzadas** > **Códigos de éxito**, agregue la lista de códigos para leer: `200,303`.
   - Haga clic en **Crear**.

3. Seleccione el grupo de destino que acaba de crear y luego haga clic en la pestaña **Destinos**:

- Haga clic en **Editar**
- Seleccione las instancias EC2 que están ejecutando la aplicación proxy y luego haga clic en **Añadir a registrados**.
- Haga clic en **Guardar**.

4. Una vez creado el grupo objetivo, debe habilitar la adherencia:
    - Abra la página del grupo de destino de AWS (**EC2**> **Equilibradores de carga** > **Grupos de destino**), seleccione la instancia del grupo de destino que acaba de configurar. En el menú **Acciones**, seleccione **Editar atributos**.
    - En la página **Editar atributos**, seleccione **Adherencia**, especifique una duración de `1 day` y luego **Guardar cambios**.
5. En el equilibrador de carga, actualice las reglas del oyente. Seleccione el equilibrador de carga que ha configurado para esta implementación y luego haga clic en la pestaña **Oyentes**.
    - Para **HTTP:80**, haga clic en **Ver/editar reglas**. En la página **Reglas** resultante, haga clic en el icono de edición (una vez en la parte superior de la página y luego nuevamente por la regla) para editar la regla. Elimine la regla THEN existente y reemplácela haciendo clic en **Agregar acción** > **Redirigir a...** En la configuración THEN resultante, especifique`HTTPS` y el puerto `443` y deje las otras opciones con la configuración predeterminada. Guarde la configuración y haga clic en **Actualizar**.
    - Para **HTTP:443**, haga clic en **Ver/editar reglas**. En la página **Reglas** resultante, haga clic en el icono de edición (una vez en la parte superior de la página y luego nuevamente por la regla) para editar la regla. En la configuración **THEN**, en **Reenviar a...** cambie el grupo de destino al grupo HTTPS que acaba de crear. En **Adherencia a nivel de grupo**, habilite la adherencia y establezca la duración en 1 día. Guarde la configuración y haga clic en **Actualizar**.

Paso 6: verificar SSL

Verifique la configuración navegando a https://tableau.example.com.