Notas de la versión de Oracle® OpenSSO Update 2

Beta

Referencia: 821–2836
Julio de 2010

ORACLE

101014@24999

Contenido

Prefacio

Las notas de la versión de Oracle OpenSSO 8.0 Update 2 proporcionan información acerca de la descarga e instalación del software de OpenSSO Update 2. Este documento contiene también información sobre los cambios realizados en el software desde la versión OpenSSO Update 1.

Quiénes deben usar esta guía

Estas notas de la versión están destinadas a los desarrolladores y administradores empresariales que ya hayan instalado e implementado Oracle OpenSSO 8.0. Debería estar familiarizado con los conceptos y los procedimientos descritos en la documentación principal del producto.

Guías relacionadas

Estas notas de la versión son un complemento de la documentación central del producto Oracle OpenSSO 8.0, que se encuentra disponible en la siguiente URL:
http://docs.sun.com/app/docs/coll/1767.1.

Referencias a sitios web de terceros relacionados

Se hace referencia a las direcciones URL de terceras partes para proporcionar información adicional relacionada.

Nota – Oracle no se hace responsable de la disponibilidad de los sitios web de terceros mencionados en este documento. Oracle no respalda ni se hace responsable de ningún contenido, anuncio, producto o cualquier otro material disponible en dichos sitios o recursos. Oracle declina toda responsabilidad sobre los posibles daños o pérdidas, reales o presuntos, causados o presuntamente causados directa o indirectamente por los contenidos, bienes o servicios citados u otros accesibles en o a través de esas páginas o recursos.

Documentación, asistencia y formación

Consulte los siguientes sitios web para obtener recursos adicionales:

- Documentación (`http://docs.sun.com`)
- Asistencia técnica (`http://www.oracle.com/us/support/systems/index.html`)
- Formación (`http://education.oracle.com`) – Haga clic en el vínculo de Sun situado en la barra de navegación de la izquierda.

Oracle valora sus comentarios

Oracle agradece sus comentarios y sugerencias sobre la calidad o utilidad de su documentación. Si encuentra algún error o desea compartir cualquier otra sugerencia de mejora, vaya a `http://docs.sun.com` y haga clic en Feedback (Comentarios). Indique el título y el número de referencia de la documentación, junto con el capítulo, la sección y el número de página, si está disponible. Indíquenos si desea recibir una respuesta.

La Red de tecnología de Oracle (`http://www.oracle.com/technetwork/index.html`) ofrece una amplia gama de recursos relacionados con el software de Oracle:

- Participe en conversaciones sobre problemas técnicos y soluciones en los foros de debate (`http://forums.oracle.com`).
- Obtenga tutoriales prácticos detallados en Oracle By Example (`http://www.oracle.com/technology/obe/start/index.html`).
- Descargue el código de muestra (`http://www.oracle.com/technology/sample_code/index.html`).

Convenciones tipográficas

En la tabla siguiente se describen las convenciones tipográficas utilizadas en este documento.

TABLA P–1 Convenciones tipográficas

Tipo de letra	Significado	Ejemplo
`AaBbCc123`	Nombres de comandos, archivos y directorios; mensajes del sistema que aparecen en la pantalla.	Edite el archivo `.login`. Utilice el comando `ls -a` para ver la lista de archivos. `nombre_máquina% ha recibido correo.`
`AaBbCc123`	Lo que escribe el usuario, frente a los mensajes del propio sistema.	`nombre_máquina%` **`su`** `Contraseña:`

TABLA P–1 Convenciones tipográficas *(Continuación)*

Tipo de letra	Significado	Ejemplo
aabbcc123	Elemento variable: se sustituye por un nombre o un valor real.	El comando para eliminar un archivo es `rm` *nombrearchivo*.
AaBbCc123	Títulos de libros, palabras o términos nuevos y palabras que deben enfatizarse.	Lea el Capítulo 6 de la *Guía de usuario*.
		Una copia en *caché* es aquella que se almacena localmente.
		No guarde el archivo.
		Nota: algunos términos enfatizados aparecen en negrita en los documentos en línea.

Indicadores de shell en los ejemplos de comandos

En la siguiente tabla se muestra el indicador del sistema UNIX predeterminado y el indicador de superusuario para shells que están incluidos en el SO Solaris de Oracle. Tenga en cuenta que el indicador del sistema predeterminado que se muestra en los comandos de ejemplo varía en función de la versión de Solaris de Oracle.

TABLA P–2 Indicadores del shell

Shell	Mensaje de petición
Shell Bash, shell Korn y shell Bourne	`$`
Shell Bash, shell Korn y shell Bourne para superusuario	`#`
C	`nombre-máquina%`
Shell de superusuario de C	`nombre_máquina#`

◆ ◆ ◆ CAPÍTULO 1

Acerca de OpenSSO 8.0 Update 2

Este capítulo contiene los temas siguientes:

Novedades de OpenSSO 8.0 Update 2

OpenSSO 8.0 Update 2 incluye mejoras en el servicio de token de seguridad y en el Fedlet de OpenSSO.

Mejoras del servicio de token de seguridad

El servicio de token de seguridad incluye ahora las siguientes características nuevas:

- Admite el tipo de token para generar un token de seguridad específico para el proveedor de servicios web.
- Admite tanto el enlace asimétrico como de transporte de X509 y los tokens de seguridad de nombre de usuario como solicitante.
- Aplica el enlace de SSL/transporte con un token de seguridad de nombre de usuario cuando el servicio STS de OpenSSO se configure con un nombre de usuario a través de SSL.
- Emite un token de seguridad de titular de clave SAML para el tipo de clave asimétrica con useKey como clave pública del cliente de servicios web y el token de seguridad X509 de cliente de servicios web.
- WSDL se actualiza de forma dinámica en función de la configuración de token de seguridad.
- Admite el cifrado mediante la clave pública del proveedor de servicios web.

- Cifra el nombre de usuario y la contraseña estáticos antes de guardarlos en el almacén de configuración.
- Admite el token de nombre de usuario como token de seguridad "En nombre de" mediante una solicitud WS-Trust.
- Admite la emisión de tokens de portador de SAML.
- El nuevo módulo de autenticación de seguridad del servicio web WSSAuth admite la validación de contraseñas implícitas.
- El nuevo módulo de autenticación OAMAuth habilita el inicio de sesión único mediante Oracle Access Manager con OpenSSO.

Para obtener más información, consulte el Capítulo 3, “Uso del servicio de token de seguridad”.

Mejoras del Fedlet

Fedlet incluye ahora las siguientes funciones nuevas:

- Admite el cifrado en el Fedlet de .NET.
- Admite las firmas en el Fedlet de .NET.
- El Fedlet de .NET admite ahora el cierre de sesión único.
- El Fedlet de .NET proporciona un inicio de sesión único iniciado por un proveedor de servicios y compatibilidad con artefactos.
- Admite varios proveedores de identidades y el servicio de detección de proveedores de identidades en el Fedlet de .NET.
- Proporciona información sobre la versión en los archivos de propiedades y configuración del Fedlet.
- Permite una nueva implementación de SPI de la contraseña.
- Admite la consulta de atributos.
- Admite el cierre de sesión único.

Para obtener más información, consulte el Capítulo 4, “Uso de Oracle OpenSSO Fedlet”.

Requisitos de hardware y software para OpenSSO 8.0 Update 2

Consulte “Hardware and Software Requirements For OpenSSO Enterprise 8.0 Update 1” de *Sun OpenSSO Enterprise 8.0 Update 1 Release Notes*.

Compatibilidad con los nuevos contenedores web

OpenSSO 8.0 Update 2 admite contenedores web, como se describe en "Support for New Web Containers" de *Sun OpenSSO Enterprise 8.0 Update 1 Release Notes*, y es compatible con los siguientes contenedores web:

- Oracle WebLogic Server 10g versión 3 (10.3)

Problemas y soluciones de OpenSSO 8.0 Update 2

CR 6959610: las muestras de OpenSSO 8.0 Update 2 deben suprimirse en el entorno de producción

Las muestras de OpenSSO 8.0 Update 2 podrían provocar posibles problemas de seguridad.

Solución. Si implementa OpenSSO 8.0 Update 2 en un entorno de producción, suprima las muestras para evitar cualquier problema de seguridad.

CR 6964648: se necesitan nuevos permisos de seguridad de Java para WebLogic Server 10.3.3

Si va a implementar OpenSSO 8.0 Update 2 en Oracle WebLogic Server 10.3.3 con el administrador de seguridad habilitado, se necesita un permiso de seguridad de Java adicional.

Solución. Agregue el siguiente permiso al archivo `weblogic.policy` de WebLogic Server 10.3.3:

```
permission java.lang.RuntimePermission "getClassLoader";
```

CR 6939443: la autenticación de certificados con comprobación LDAP u OCSP presenta errores en WebLogic Server 10.3.x

Debido a un problema en las versiones anteriores de Oracle WebLogic Server como, por ejemplo, la versión 10.3.0 y 10.3.1, la autenticación de certificados con la comprobación LDAP u OCSP habilitada presenta errores.

Solución. Este problema se ha solucionado en WebLogic Server 10.3.3. Para utilizar la autenticación de certificados con la función de comprobación LDAP u OCSP, use OpenSSO Update 2 con WebLogic Server 10.3.3.

CR 6967026: el programa de configuración no se puede conectar a una instancia del servidor de directorios habilitada para LDAPS desde GlassFish 2.1.x

Si GlassFish Enterprise Server v2.1.1 o v2.1.2 se implementa como contenedor web de OpenSSO 8.0 Update 2, el programa de configuración no se podrá conectar con la instancia del servidor de directorios habilitada para LDAPS.

Solución. Para utilizar el servidor de directorios habilitado para LDAPS con GlassFish como contenedor web, implemente GlassFish Enterprise Server v2.1.

CR 6948937: la activación de OpenSSO 8.0 Update 2 en la consola de administración de WebLogic Server 10.3.3 provoca excepciones

Si implementa OpenSSO 8.0 Update 2 (`opensso.war`) en la consola de administración de WebLogic Server 10.3.3 y hace clic en Iniciar para permitir que OpenSSO 8.0 Update 2 comience a recibir solicitudes, se generarán excepciones en la consola en la que se ha iniciado el dominio de WebLogic Server.

Nota: después de iniciar OpenSSO 8.0 Update 2, esta aplicación permanecerá activa y no se generarán de nuevo excepciones hasta que se detenga y reinicie OpenSSO 8.0 Update 2.

Solución. Copie el archivo `saaj-impl.jar` desde el archivo `opensso-client-jdk15.war` de OpenSSO 8 Update 2 en el directorio de configuración `endorsed` de WebLogic Server 10.3.3 como se indica a continuación:

1. Detenga el dominio de Oracle WebLogic Server 10.3.3.
2. Si es necesario, descomprima el archivo `opensso.zip` de OpenSSO 8.0 Update 2.
3. Cree un directorio temporal y descomprima el archivo *zip-root*`/opensso/samples/opensso-client.zip` en ese directorio, donde *zip-root* hace referencia a la ubicación en la que se ha descomprimido `opensso.zip`. Por ejemplo:

   ```
   cd zip-root/opensso/samples
   mkdir ziptmp
   cd ziptmp
   unzip ../opensso-client.zip
   ```

4. Cree un directorio temporal y extraiga el archivo `saaj-impl.jar` de `opensso-client-jdk15.war`. Por ejemplo:

   ```
   cd zip-root/opensso/samples/ziptmp/war
   mkdir wartmp
   cd wartmp
   jar xvf ../opensso-client-jdk15.war WEB-INF/lib/saaj-impl.jar
   ```

5. Cree un directorio con el nombre `endorsed` en el directorio `WEBLOGIC_JAVA_HOME/jre/lib` (si aún no existe el directorio `endorsed`), donde `WEBLOGIC_JAVA_HOME` es el JDK para cuyo uso se ha configurado WebLogic Server.
6. Copie el archivo `saaj-impl.jar` en el directorio `WEBLOGIC_JAVA_HOME/jre/lib/endorsed`.
7. Inicie el dominio de WebLogic Server.

CR 6959373: el contenedor web debe reiniciarse después de ejecutar la secuencia de comandos `updateschema`

Después de ejecutar la secuencia de comandos `updateschema.sh` o `updateschema.bat` , debe reiniciarse el contenedor web de OpenSSO 8.0 Update 2.

CR 6961419: se necesita un archivo de contraseña para ejecutar la secuencia de comandos `updateschema.bat`

La secuencia de comandos `updateschema.bat` ejecuta varios comandos `ssoadm`. Por lo tanto, antes de ejecutar `updateschema.bat` en los sistemas Windows, cree un archivo que contenga la contraseña del usuario `amadmin` en texto no cifrado. La secuencia de comandos `updateschema.bat` le solicita la ruta al archivo de contraseña. Antes de que finalice la secuencia de comando, se suprime el archivo de contraseña.

Documentación de OpenSSO 8.0 Update 2

Además de este documento, hay disponible documentación adicional de OpenSSO 8.0 en la siguiente colección:

`http://docs.sun.com/coll/1767.1`

Problemas de la documentación

La documentación de OpenSSO 8.0 Update 2 presenta los siguientes problemas:

CR 6958580: la ayuda en línea de la consola proporciona información sobre agentes de detección incompatibles

La ayuda en línea de la consola de administración de OpenSSO 8.0 Update 2 incluye información sobre los agentes de detección, incluso aunque estos no sean compatibles.

Solución. Ninguna. Omita la información sobre los agentes de detección en la ayuda en línea.

CR 6967006: la ayuda en línea de la consola no incluye información sobre los módulos de autenticación OAMAuth y WSSAuth

La ayuda en línea de la consola de administración de OpenSSO 8.0 Update 1 no incluye información sobre los módulos de autenticación de Oracle Access Manager (OAM) y de Seguridad de servicios web (WSS).

Solución. Para obtener información sobre estos módulos de autenticación, consulte el Capítulo 3, "Uso del servicio de token de seguridad".

CR 6953582: la referencia de la API de Java del Fedlet debería ser pública

La referencia pública de la API de Java del Fedlet está disponible como parte de la referencia de la API de Java de Oracle OpenSSO 8.0 Update 2, que se incluye en la siguiente colección de documentación: `http://docs.sun.com/coll/1767.1`.

Nota: OpenSSO 8.0 Update 2 no admite el método `getPolicyDecisionForFedlet`, aunque se encuentre en la referencia de la API de Java.

CR 6953579: el archivo README (Léame) de Fedlet de OpenSSO debería incluir información sobre la función de cierre de sesión único

El archivo README (Léame) de Fedlet no incluye información sobre la función de cierre de sesión único.

Solución. En Oracle OpenSSO 8.0 Update 2, la función de cierre de sesión único de Fedlet aparece documentada en el Capítulo 4, "Uso de Oracle OpenSSO Fedlet".

Información adicional y recursos

Puede encontrar también información y recursos útiles en las siguientes ubicaciones:

- "Notificaciones y anuncios de desaprobación" en la página 17
- "Comunicar problemas y enviar comentarios" en la página 18
- "Funciones de accesibilidad para usuarios con discapacidades" en la página 18
- "Sitios web de terceros relacionados" en la página 18

- Servicios de atención al cliente de Oracle para los sistemas:

  `http://www.oracle.com/us/support/systems/advanced-customer-services/index.html`
- Productos de software: `http://www.oracle.com/us/sun/sun-products-map-075562.html`
- SunSolve: `http://sunsolve.sun.com/`
- Red de desarrolladores de Sun (SDN): http://developers.sun.com/
- Servicios para desarrolladores de Sun:`http://developers.sun.com/services/`

Notificaciones y anuncios de desaprobación

- Las API del servicio de administración de servicios (SMS) (paquete `com.sun.identity.sm`) y el modelo de SMS no se incluirán en una próxima versión de OpenSSO.
- El módulo y el asistente de autenticación de Unix (`amunixd`) no se incluirán en una próxima versión de OpenSSO.
- En las notas de la versión de Sun Java System Access Manager 7.1, se indica que el paquete `com.iplanet.am.sdk` de Access Manager, conocido comúnmente como Access Manager SDK (AMSDK), y todas las API y las plantillas XML relacionadas no se incluirán en una próxima versión de OpenSSO.

  Por tanto, cuando se suprima AMSDK, también se eliminará la opción de modo tradicional y su compatibilidad.

Las opciones de migración no están disponibles actualmente y no se prevé que lo estén en el futuro. Oracle Identity Manager proporciona soluciones de aprovisionamiento de usuarios que se pueden utilizar en lugar de AMSDK. Para obtener más información sobre Identity Manager, consulte http://www.oracle.com/products/middleware/identity-management/identity-manager.html.

Comunicar problemas y enviar comentarios

Si tiene una pregunta o duda acerca de OpenSSO 8.0 Update 2 o una versión de parche posterior, póngase en contacto con el servicio de recursos de asistencia técnica en http://sunsolve.sun.com/.

Este sitio dispone de enlaces a la base de datos de soluciones, al centro de asistencia en línea y al rastreador de productos, así como a programas de mantenimiento y números de contacto de asistencia. Si solicita ayuda para un problema, incluya la siguiente información:

- Descripción del problema, incluido el momento en que se produjo y sus efectos en el funcionamiento
- El tipo de equipo, la versión del sistema operativo, el contenedor web y su versión, y la versión de JDK y OpenSSO, incluido cualquier parche o software que pueda afectar al problema
- Los pasos para reproducir el problema
- Cualquier registro de error o volcado del núcleo

Funciones de accesibilidad para usuarios con discapacidades

Para obtener las funciones de accesibilidad comercializadas desde la publicación de este medio, consulte la sección 508 de evaluaciones de productos disponible previa solicitud a fin de determinar las versiones más adecuadas para implementar las soluciones accesibles.

Para obtener información sobre el compromiso de Sun con la accesibilidad, consulte http://www.oracle.com/index.html.

Sitios web de terceros relacionados

Se hace referencia a las direcciones URL de terceras partes para proporcionar información adicional relacionada.

Nota – Oracle no se hace responsable de la disponibilidad de los sitios web de terceros que se mencionan en este documento. Oracle no respalda ni se hace responsable de ningún contenido, anuncio, producto o cualquier otro material disponible en dichos sitios o recursos. Oracle no se responsabiliza de ningún daño, real o supuesto, ni de posibles pérdidas que se pudieran derivar del uso de los contenidos, bienes o servicios que estén disponibles en dichos sitios o recursos.

◆ ◆ ◆ CAPÍTULO 2

Instalación de OpenSSO 8.0 Update 2

Este capítulo contiene los temas siguientes:

Información general de la instalación de OpenSSO 8.0 Update 2

OpenSSO 8.0 Update 2 está disponible como parche **TBS**.

Antes de instalar OpenSSO 8.0 Update 2 (o parches posteriores), consulte la información acerca de las nuevas funciones, los requisitos de hardware y software, y los problemas y soluciones que contiene este documento.

OpenSSO 8.0 Update 2 incluye un archivo `opensso.war` que puede instalar mediante los siguientes métodos:

- **Aplicar un parche en una implementación de OpenSSO 8.0 existente**: use la utilidad `ssopatch` en Update 2 para aplicar un parche en una implementación de OpenSSO 8.0 existente, como se describe en este capítulo.

  Nota: Oracle admite solo la aplicación de parches en las versiones de OpenSSO 8.0. Por ejemplo, se admite la aplicación del parche OpenSSO 8.0 Update 2 en OpenSSO 8.0.
- **Instalar una implementación de OpenSSO 8.0 Update 2:** instale y configure el archivo `opensso.war` de OpenSSO 8.0 Update 2, como se describe en *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.
- **Crear un nuevo archivo WAR especializado**: utilice la secuencia de comandos`createwar` para crear uno de los siguientes nuevos archivos WAR desde el archivo `opensso.war` de Update 2:
  - Archivo WAR solo de la consola de administración de OpenSSO
  - Archivo WAR del servidor de la IU de autenticación distribuida
  - Archivo WAR solo del servidor de OpenSSO sin la consola de administración
  - Archivo WAR del servicio de detección IDP

    Para obtener información, consulte el Capítulo 4, "Creating a Specialized OpenSSO Enterprise 8.0 Update 1 WAR File" de *Sun OpenSSO Enterprise 8.0 Update 1 Release Notes*.
- **Aplicar un parche a un archivo WAR especializado de OpenSSO**: use la utilidad `ssopatch` de Update 2 para aplicar un parche en un archivo WAR especializado de OpenSSO 8.0, como se describe en el Capítulo 23, "Patching OpenSSO Enterprise 8.0" de *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

Nota – Si ejecuta Access Manager 7.1 o Access Manager 7 2005Q4 y desea actualizar a Update 2, siga estos pasos:

1. Actualice Access Manager 7.x a OpenSSO 8.0, como se describe en *Sun OpenSSO Enterprise 8.0 Upgrade Guide* .
2. Aplique el parche de Update 2, como se describe en este capítulo.

Parches de OpenSSO 8.0 Update 2

Sun publica frecuentemente parches para OpenSSO 8.0 Update 2. Para obtener información sobre estos parches, visite esta página periódicamente.

Planificación de la aplicación del parche

▼ Para planificar la aplicación de un parche para OpenSSO 8.0

1. **Consulte la "Descripción general de la utilidad ssopatch" en la página 23.**
2. **Instale la utilidad del parche para su plataforma, como se describe en "Instalación de la utilidad ssopatch" en la página 24.**
3. **Obtenga información sobre el archivo WAR existente para determinar si se ha personalizado o modificado, como se describe en "Comparación de un archivo WAR de OpenSSO con su archivo manifest interno" en la página 26.**
4. **Compare el archivo WAR existente con el de Update 2 para comprobar los archivos personalizados en el archivo WAR original, los archivos actualizados en el nuevo archivo WAR y los archivos agregados o eliminados entre las dos versiones, como se describe en "Comparación de dos archivos WAR de OpenSSO" en la página 27.**
5. **Realice una copia de seguridad del archivo WAR de OpenSSO existente y guárdelo, como se describe en "Copia de seguridad de un archivo WAR de OpenSSO" en la página 25.**
6. **Aplique un parche al archivo WAR de OpenSSO, como se describe en "Aplicación de un parche en un archivo WAR de OpenSSO" en la página 28.**
7. **Ejecute la secuencia de comandos `updateschema`, como se describe en "Ejecución de la secuencia de comandos `updateschema`" en la página 31.**

   Nota: si va a aplicar un parche en un archivo WAR especializado que se ha generado desde un archivo `opensso.war`, como, por ejemplo, un archivo WAR solo de servidor de OpenSSO, solo de consola de administración, de servidor de la IU de la autenticación distribuida o del servicio de detección IDP, consulte "Aplicación de un parche en un archivo WAR de OpenSSO especializado" en la página 31.

Descripción general de la utilidad `ssopatch`

`ssopatch` es una utilidad de línea de comandos de Java que se encuentra disponible en los sistemas Solaris y Linux como `ssopatch` y en Windows como `ssopatch.bat`.

Nota: la sintaxis de `ssopatch` en OpenSSO 8.0 Update 2 se ha modificado considerablemente desde la versión OpenSSO 8.0. Para conocer la nueva sintaxis, consulte "Ejecución de la secuencia de comandos `updateschema`" en la página 31.

La utilidad de parche ssopatch realiza las siguientes funciones:

- Compara un archivo WAR de OpenSSO con su archivo manifest original si el archivo WAR se ha personalizado o modificado.
- Compara dos archivos WAR de OpenSSO para determinar las diferencias entre estos, incluida cualquier personalización realizada en el archivo WAR original y cualquier cambio efectuado en el nuevo archivo WAR.
- Genera un área provisional de los archivos necesarios para crear un nuevo archivo WAR de OpenSSO al que se ha aplicado un parche.

Después de descargar y descomprimir el archivo ZIP de OpenSSO 8.0 Update 2 (opensso_80U2.zip), las utilidades de aplicación de parches y los archivos relacionados están disponibles en el archivo ssoPatchTools.zip del directorio *zip-root*/opensso/tools , donde *zip-root* hace referencia a la ubicación en la que se ha descomprimido opensso_80U2.zip.

La utilidad ssopatch usa el archivo manifest para determinar el contenido de un archivo WAR de OpenSSO específico. Un archivo manifest es un archivo de texto ASCII que contiene los siguientes elementos:

- Una cadena que identifica la versión específica del archivo WAR de OpenSSO.
- Todos los archivos individuales del archivo WAR de OpenSSO, con la información de suma de comprobación de cada uno de ellos.

Por lo general, el archivo manifest recibe el nombre de OpenSSO.manifest y se almacena en el directorio META-INF del archivo WAR de OpenSSO.

La utilidad ssopatch envía los resultados a la salida estándar (stdout). Si lo prefiere, puede redireccionar la salida a un archivo para capturar la salida de ssopatch . Si ssopatch finaliza satisfactoriamente, se devolverá el código de salida cero (0). Si se producen errores, ssopatch devuelve un código de error distinto a cero.

Instalación de la utilidad ssopatch

Antes de instalar la utilidad ssopatch, realice lo siguiente:

- Descargue y descomprima el archivo ZIP de OpenSSO 8.0 Update 2 (opensso_80U2.zip).
- Establezca la variable de entorno JAVA_HOME para que señale a JDK 1.5 o posterior.

Para instalar la utilidad ssopatch

1. Busque el archivo ssoPatchTools.zip en el directorio *zip-root*/opensso/tools, donde *zip-root* hace referencia a la ubicación en la que se ha descomprimido opensso_80U2.zip.
2. Cree un nuevo directorio para descomprimir el archivo ssoPatchTools.zip. Por ejemplo: ssopatchtools

3. Descomprima el archivo ssoPatchTools.zip en el nuevo directorio.
4. Si desea ejecutar la utilidad ssopatch desde un directorio distinto al actual sin especificar la ruta completa, agregue la utilidad a la variable PATH.

En la siguiente tabla se describen los archivos de ssoPatchTools.zip.

Archivo o directorio	Descripción
README (Léame)	Archivo Readme (Léame) que proporciona una descripción de ssopatch
/lib	Archivos JAR de ssopatch necesarios
/patch	Secuencias de comandos updateschema y updateschema.bat y archivos XML relacionados
/resources	Archivos de propiedades necesarios
ssopatch y ssopatch.bat	Utilidades para los sistemas Solaris, Linux y Windows

Copia de seguridad de un archivo WAR de OpenSSO

Antes de empezar, realice una copia de seguridad del archivo WAR de OpenSSO y de los datos de configuración:

- Copie el archivo WAR de OpenSSO existente en una ubicación segura. A continuación, si, por algún motivo, necesita realizar una copia de seguridad de Update 2, puede volver a implementar la copia de seguridad del archivo WAR.
- Realice una copia de seguridad de los datos, como se describe en el Capítulo 15, "Backing Up and Restoring Configuration Data" de *Sun OpenSSO Enterprise 8.0 Administration Guide*.

Ejecución de la utilidad ssopatch

Para ejecutar la utilidad ssopatch, use esta sintaxis:

```
ssopatch
--help|-?
[--locale|-l]

ssopatch
--war-file|-o
[--manifest|-m]
[--locale|-l]
```

```
ssopatch
--war-file|-o
--war-file-compare|-c
[--staging|-s]
[--locale|-l]
[--override|-r]
[--overwrite|-w]
```

donde las opciones son:

- `-war-file|-o` especifica una ruta a un archivo WAR (por ejemplo, `opensso.war`) que se ha implementado anteriormente.
- `-manifest|-m` especifica la ruta al archivo manifest que desea crear. El archivo manifest se generará a partir del archivo WAR file indicado por `-war-file|-o` si se ha especificado esta opción.
- `-war-file-compare|-c` especifica una ruta a un archivo WAR para compararlo con el archivo WAR indicado por `-war-file|-o` .
- `-staging|-s` especifica una ruta al área provisional en el que se escribirán los archivos de un archivo WAR de OpenSSO.
- `-locale|-l` especifica la configuración regional que se utilizará. Si no se especifica esta opción, `ssopatch` utilizará la configuración regional predeterminada del sistema.
- `-override|-r` anula la comprobación de revisión para los dos archivos WAR. La comprobación de revisión determina las versiones de los archivos WAR y el proceso solo continúa en caso de las dos sean compatibles. Esta opción permite anular esta comprobación.

  El valor predeterminado es "false" (falso), lo que indica que se realizará la comprobación de revisión.
- `-overwrite|-w` sobrescribe los archivos en el área provisional existente. El valor predeterminado es "false" (falso), lo que indica que no se sobrescribirán los archivos.

Comparación de un archivo WAR de OpenSSO con su archivo manifest interno

Utilice este procedimiento para determinar si un archivo WAR de OpenSSO se ha personalizado o modificado desde que se descargó.

La utilidad `ssopatch` genera un nuevo archivo manifest interno y, a continuación, lo compara con el archivo manifest almacenado en el archivo WAR de OpenSSO original incluido en el directorio `META-INF`.

Para comparar un archivo WAR de OpenSSO con su archivo manifest interno

1. Ejecute `ssopatch` para comparar el archivo WAR de OpenSSO con su archivo manifest interno. Por ejemplo:

```
./ssopatch -o /zip-root/opensso/deployable-war/opensso.war
Generating Manifest for: /zip-root/opensso/deployable-war/opensso.war
Comparing manifest of Internal (Enterprise 8.0 Build 6(200810311055))
against /zip-root/opensso/deployable-war/opensso.war (generated-200905050855)
File not in original war (images/login-origimage.jpg)
File updated in new war (images/login-backimage.jpg)
File updated in new war (WEB-INF/classes/amConfigurator.properties)
Differences: 3
```

En este ejemplo se muestran los siguientes cambios realizados en el archivo WAR original:

- `images/login-origimage.jpg` se encuentra en `opensso.war` , pero no en el archivo manifest original.
- `images/login-backimage.jpg` se ha personalizado en el archivo `opensso.war` del archivo manifest original.
- El archivo `WEB-INF/classes/amConfigurator.properties` se ha personalizado en el archivo `opensso.war` del archivo manifest original.

Comparación de dos archivos WAR de OpenSSO

Utilice este procedimiento para comparar dos archivo WAR y mostrar los archivos que se han:

- personalizado en el archivo WAR de OpenSSO original
- actualizado en un nuevo archivo WAR de OpenSSO
- agregado o eliminado entre las dos versiones de los archivos WAR de OpenSSO

Para comparar dos archivos WAR de OpenSSO

1. Ejecute `ssopatch` para comparar los dos archivos WAR. En el ejemplo, la opción `-override` se utiliza para anular la comprobación de revisión entre los dos archivos WAR:

```
./ssopatch -o /zip-root/opensso/deployable-war/opensso.war
-c /u1/opensso/deployable-war/opensso.war --override
Generating Manifest for: /zip-root/opensso/deployable-war/opensso.war
Original manifest: Enterprise 8.0 Build 6(200810311055)
New manifest: Enterprise 8.0 Update 2 Build 6.1(200904300525)
Versions are compatible
Generating Manifest for: /u1/opensso/deployable-war/opensso.war
Comparing manifest of /zip-root/opensso/deployable-war/opensso.war
(generated-200905050919) against
    /u1/opensso/deployable-war/opensso.war (generated-200905050920)
```

```
File updated in new war(WEB-INF/classes/amClientDetection_en.properties)
File updated in new war(WEB-INF/classes/fmSAMLConfiguration_fr.properties)
...
Differences: 1821
Customizations: 3
```

En este ejemplo se muestran los archivos que se han actualizado y personalizado en el nuevo archivo WAR.

Aplicación de un parche en un archivo WAR de OpenSSO

Utilice este procedimiento para crear una nueva área de provisiones donde el archivo WAR original se combinará con uno nuevo.

En esta operación se comparan los archivos manifest de cada uno de los archivos WAR y, a continuación, se muestra lo siguiente:

- Los archivos personalizados en el archivo WAR original
- Los archivos actualizados en el archivo WAR nuevo
- Los archivos añadidos o suprimidos entre las dos versiones del archivo WAR

La utilidad `ssopatch` copia a continuación los archivos correspondientes en un directorio provisional, donde se debe agregar cualquier personalización antes de crear e implementar el nuevo archivo WAR al que se le ha aplicado un parche.

Para crear un área provisional con el objeto de aplicar un archivo al archivo WAR de OpenSSO

1. Aunque la utilidad `ssopatch` no modifica el archivo `opensso.war` original, es recomendable que realice una copia de seguridad del mismo en caso de que necesite deshacer la anulación de la aplicación del parche en `opensso.war`.
2. Ejecute `ssopatch` para crear un área provisional. Por ejemplo:

```
./ssopatch -o /zip-root/opensso/deployable-war/opensso.war
  -c /u1/opensso/deployable-war/opensso.war --override -s /tmp/staging
Generating Manifest for: /zip-root/opensso/deployable-war/opensso.war
Original manifest: Enterprise 8.0 Build 6(200810311055)
New manifest: Enterprise 8.0 Update 2 Build 6.1(200904300525)
Versions are compatible
Generating Manifest for: /u1/opensso/deployable-war/opensso.war
Comparing manifest of /zip-root/opensso/deployable-war/opensso.war
    (generated-200905051031) against /u1/opensso/deployable-war/opensso.war
    (generated-200905051032)
File was customized in original, but not found in new war.
Staging area using original war version (samples/saml2/sae/header.jsp)
File was customized in original, but not found in new war.
Staging area using original war version
```

```
    (WEB-INF/template/opends/config/upgrade/config.ldif.4517)
File was customized in original, but not found in new war.
Staging area using original war version
    (WEB-INF/template/opends/config/upgrade/schema.ldif.4517)
Differences: 1813
Customizations: 0
```

En este ejemplo, `/tmp/staging` es el área provisional en el que `ssopatch` copia los archivos.

Actualice los archivos en el área provisional según corresponda. Para ello, utilice los resultados del paso anterior.

Utilice la siguiente tabla para determinar la acción que debe realizar en cada archivo antes de aplicar el parche y generar un nuevo archivo WAR.

Resultados de `ssopatch`	Explicación y acción que debe realizarse
`File not in original war` *filename*	El archivo indicado no existe en el archivo WAR original, pero sí en la versión más reciente del mismo. **Acción**: ninguna
`File updated in new war` *filename*	El archivo indicado existe tanto en el archivo WAR original como en el nuevo y se ha actualizado en la versión más reciente del archivo WAR. No se ha realizado ninguna personalización en el archivo WAR original. **Acción**: ninguna
`File customized` *filename*	El archivo indicado existe en los dos archivos WAR, se ha personalizado en la versión original de este archivo, pero no se ha actualizado en la versión más reciente del mismo. **Acción**: ninguna
`May require manual customization` *filename*	El archivo existe en los dos archivos WAR, se ha personalizado en la versión original de este archivo y se ha actualizado en la versión más reciente del mismo. **Acción**: si desea que las personalizaciones realizadas estén presentes en el archivo, debe agregarlas manualmente al nuevo archivo actualizado en el directorio provisional.
`File was customized in original, but not found in new war`	El archivo existe en el archivo WAR original, pero no en el nuevo. **Acción**: ninguna

Pasos siguientes

1. Cree un nuevo archivo WAR de OpenSSO a partir de los archivos del área provisional. Por ejemplo:

```
cd /tmp/staging
jar cvf /patched/opensso.war *
```

donde `/patched/opensso.war` es el nuevo archivo WAR de OpenSSO al que se la ha aplicado un parche.

2. Vuelva a implementar el archivo `/patched/opensso.war` en el contenedor web mediante el URI de implementación original. Por ejemplo, `/opensso`

Cambios en la configuración de OpenSSO. Es posible que un archivo WAR de OpenSSO nuevo presente cambios de configuración que no se encontraban en el archivo WAR original. Cualquier cambio de configuración, si lo hay, se documentará por separado en cada parche. Consulte la documentación del parche y las *Notas de la versión de Sun OpenSSO Enterprise 8.0* para obtener más información sobre los cambios de configuración. (La cadena de versión del archivo manifest de OpenSSO se modificará en el nuevo archivo WAR, aunque no haya ningún cambio de configuración).

Si necesita anular la versión con el parche aplicado, anule la implementación del archivo WAR al que se ha aplicado el archivo y vuelva a implementar el archivo WAR original.

Creación de un archivo manifest para un archivo WAR de OpenSSO

Un archivo manifest de OpenSSO es un archivo de texto que identifica todos los archivos individuales de un archivo WAR de una determinada versión y que incluye la información de suma de comprobación de cada archivo.

Utilice este procedimiento para crear un archivo manifest para incluirlo en un archivo WAR de OpenSSO especializado como, por ejemplo, un archivo WAR solo de servidor, solo de la consola de administración, de servidor de la IU de la autenticación distribuida o del servicio de detección IDP.

Para crear un archivo manifest para un archivo WAR de OpenSSO

1. Ejecute `ssopatch` para crear el archivo manifest de OpenSSO. Por ejemplo:

   ```
   ./ssopatch -o zip-root/opensso/deployable-war/opensso.war --manifest /tmp/manifest
   ```

   donde `opensso.war` es un archivo WAR de OpenSSO existente.

   La utilidad `ssopatch` crea un nuevo archivo manifest denominado `manifest` en el directorio `/tmp`.

2. Para permitir la aplicación de un parche en el archivo WAR, copie este nuevo archivo manifest en el directorio `META-INF` dentro del archivo `opensso.war` . Por ejemplo:

   ```
   mkdir META-INF
   cp /tmp/manifest META-INF
   jar uf opensso.war META-INF/manifest
   ```

Aplicación de un parche en un archivo WAR de OpenSSO especializado

Si ha creado anteriormente un archivo WAR de OpenSSO especializado como, por ejemplo, un archivo de OpenSSO de solo servidor, de solo consola, del servidor de la IU de la autenticación distribuida o del servicio de detección IDP, puede aplicarle un parche mediante la utilidad `ssopatch` .

Para aplicar un parche a un archivo WAR de OpenSSO

1. Cree un archivo manifest para el archivo WAR de OpenSSO, como se describe en "Creación de un archivo manifest para un archivo WAR de OpenSSO" en la página 30.

   Nota: cree el archivo manifest en función del archivo `opensso.war` original de OpenSSO 8.0 proporcionado por Sun antes de las personalizaciones que haya podido efectuar en él. Si el archivo manifest se crea después de personalizarlo, es posible que `ssopatch` utilice los archivos de Update 2 en lugar de las personalizaciones, por lo que deberá volver a realizarlas una vez aplicado el parche.
2. Genere el archivo WAR de OpenSSO a partir del archivo `opensso.war` de OpenSSO 8.0 Update 2, como se describe en el Capítulo 4, "Creating a Specialized OpenSSO Enterprise 8.0 Update 1 WAR File" de *Sun OpenSSO Enterprise 8.0 Update 1 Release Notes*.
3. Emplee la utilidad `ssopatch` para comparar los archivos WAR antiguo y nuevo.
4. Genere un área provisional para el nuevo archivo WAR especializado, como se describe en "Para crear un área provisional con el objeto de aplicar un archivo al archivo WAR de OpenSSO" en la página 28.
5. Vuelva a implementar el archivo WAR especializado.

Ejecución de la secuencia de comandos `updateschema`

Después de ejecutar `ssopatch`, ejecute `updateschema.sh` en los sistemas Solaris o Linux, o `updateschema.bat` en Windows. La secuencia de comandos actualiza la versión del servidor de OpenSSO y agrega nuevas propiedades predeterminadas del servidor, así como los nuevos esquemas de atributos necesarios para las soluciones de errores y las mejoras de Update 2. Debe ejecutar `updateschema` para actualizar la versión del servidor.

Antes de la instalación

- La secuencia de comandos `updateschema.sh` o `updateschema.bat` requiere la versión de Update 2 (o posterior) de la utilidad de línea de comandos `ssoadm` . Por tanto, antes de ejecutar esta secuencia de comandos, instale las herramientas de administración de Update

2, como se describe en el Capítulo 3, "Installing the OpenSSO Enterprise 8.0 Update 1 Admin Tools" de *Sun OpenSSO Enterprise 8.0 Update 1 Release Notes*.

- La secuencia de comandos `updateschema.bat` ejecuta varios comandos `ssoadm`. Por tanto, antes de ejecutar`updateschema.bat` en los sistemas Windows, cree un archivo que contenga la contraseña del usuario `amadmin` en texto no cifrado. La secuencia de comandos `updateschema.bat` le solicita la ruta al archivo de contraseña. Antes de que finalice la secuencia de comando, se suprime el archivo de contraseña.

Para ejecutar la secuencia de comandos `updateschema`

1. Acceda al directorio *patch-tools*`/patch` , donde *patch-tools* hace referencia a la ubicación en la que se ha descomprimido `ssoPatchTools.zip`.
2. Ejecute `updateschema.sh` o `updateschema.bat` . Por ejemplo, en los sistemas Solaris:

   ```
   ./updateschema.sh
   ```

3. Cuando se lo solicite la secuencia de comandos, proporcione la siguiente información:
   - La ruta completa a la utilidad `ssoadm` (sin incluir `ssoadm`). Por ejemplo: `/opt/ssotools/opensso/bin`
   - La contraseña de `amadmin`

   La secuencia de comandos `updateschema.sh` o `updateschema.bat` escribe todos los mensajes o errores en la salida estándar.
4. Reinicie el contenedor web de OpenSSO 8.0 Update 2.

Anulación de una instalación con parche

Si necesita anular la instalación a la que se ha aplicado un parche, solo tiene que volver a implementar el archivo `opensso.war` original (o el archivo WAR especializado).

◆ ◆ ◆ CAPÍTULO 3

Uso del servicio de token de seguridad

Como servicio de entidad de emisora de confianza, el servicio de token de seguridad de OpenSSO emite y valida tokens de seguridad. Como proveedor de seguridad de servicios web, el servicio de token de seguridad protege la comunicación entre el cliente de servicios web y el propio servicio STS de OpenSSO. Se ha realizado un gran número de mejoras en el servicio de token de seguridad desde la versión OpenSSO 8.0 Update 2.

Este capítulo contiene los temas siguientes:

Adición de un módulo de autenticación WSSAuth

El módulo de autenticación de seguridad de servicios web permite a OpenSSO validar un nombre de usuario con una contraseña implícita recibida como token de autenticación e incluida en una solicitud de servicio del cliente de servicios web enviada a un proveedor de servicios web.

▼ Para agregar una nueva instancia del módulo de autenticación de seguridad de servicios web

1. **En la ficha Access Manager, haga clic en la ficha secundaria Autenticación.**
2. **En la sección Instancias del módulo, haga clic en Nuevo.**

3 **En el campo Nombre, escriba un nombre para la instancia del módulo de autenticación WSSAuth.**

4 **En Tipo, seleccione WSSAuth.**

5 **Configure la instancia del módulo de autenticación WSSAuth.**

▼ Para configurar una instancia del módulo de autenticación WSSAuth

1 **En la ficha Access Manager, haga clic en la ficha secundaria Autenticación.**

2 **En la sección Instancias del módulo, haga clic en el nombre de la instancia del módulo de autenticación WSSAuth que desee configurar.**

3 **Especifique valores para los atributos de dominio de la instancia del módulo de autenticación WSSAuth.**

En la siguiente tabla se proporcionan una lista y descripciones de los atributos que puede configurar.

Atributo de búsqueda de usuario	Pendiente de desarrollo
Dominio de usuario	Pendiente de desarrollo
Atributo de contraseña de usuario	Pendiente de desarrollo
Nivel de autenticación	Pendiente de desarrollo

Adición de un módulo de autenticación OAMAuth

El módulo de autenticación de Oracle permite a OpenSSO realizar una autenticación y aplicar un inicio de sesión único para un administrador que anteriormente se haya autenticado en OpenSSO en Oracle Access Manager. El administrador no tiene que proporcionar las credenciales para autenticarse en OpenSSO.

▼ Para agregar una nueva instancia del módulo de autenticación de Oracle

1. **En la ficha Access Manager, haga clic en la ficha secundaria Autenticación.**
2. **En la sección Instancias del módulo, haga clic en Nuevo.**
3. **En el campo Nombre, escriba un nombre para la instancia del módulo de autenticación de Oracle.**
4. **En Tipo, seleccione OAMAuth.**
5. **Haga clic en Aceptar.**
6. **Configure la instancia del módulo de autenticación OAMAuth.**

▼ Para configurar una instancia del módulo de autenticación de Oracle

1. **En la ficha Access Manager, haga clic en la ficha secundaria Autenticación.**
2. **En la sección Instancias del módulo, haga clic en el nombre de la instancia del módulo de autenticación OAMAuth que desee configurar.**
3. **Especifique valores para los atributos de dominio de la instancia del módulo de autenticación de Oracle.**

   En la siguiente tabla se proporcionan una lista y descripciones de los atributos que puede configurar.

Nombre de encabezado de usuario remoto	Pendiente de desarrollo
Valores de encabezado permitidos	La lista de valores actuales muestra Pendiente de desarrollo ■ Para añadir un valor de encabezado a la lista, escriba Pendiente de desarrollo en el campo Nuevo valor y, a continuación, haga clic en Añadir. ■ Para suprimir una entrada de la lista de valores actuales, seleccione la entrada y, a continuación, haga clic en Suprimir.

Nivel de autenticación	Pendiente de desarrollo

Creación de tokens de seguridad

El servicio de token de seguridad de Oracle OpenSSO (servicio STS de OpenSSO) establece una relación de confianza entre un cliente y un proveedor de servicios web y, a continuación, consolida esta confianza entre ellos. The web service can trust tokens issued by just one entity?OpenSSO STS? en lugar de tener que comunicarse con varios clientes. De esta forma, el servicio STS de OpenSSO reduce significativamente la carga adicional indirecta de la administración de puntos de confianza.

En las siguientes secciones se proporcionan instrucciones para determinar sus necesidades de token de seguridad y para configurar el servicio de token de seguridad a fin de generar y validar los tokens de seguridad que satisfagan estas necesidades.

Registro de un proveedor de servicios web en el servicio STS de OpenSSO

Al agregar un nuevo perfil de agente de seguridad del proveedor de servicios web, este proveedor se registra automáticamente en el servicio STS de OpenSSO. Consulte las siguientes secciones para obtener más información:

Una vez registrado un proveedor de servicios web en el servicio STS de OpenSSO, puede configurar este servicio a fin de generar tokens de seguridad del cliente web adecuados para el proveedor de servicios web.

Solicitud de un token de seguridad del cliente de servicios web desde el servicio STS de OpenSSO

Antes de configurar el servicio de token de seguridad para generar tokens de seguridad del cliente web, debe determinar qué tipo de token de seguridad necesita el proveedor de servicios web. El servicio STS de OpenSSO admite tokens de seguridad de Liberty Alliance Project y del perfil de seguridad básico de interoperabilidad de servicios web.

Flujo de proceso de creación de tokens de seguridad

Una vez habilitada la seguridad mediante los tokens de Liberty Alliance Project, el cliente HTTP o el navegador, se envía una solicitud de acceso mediante el cliente de servicios web al proveedor de servicios web. Un agente de seguridad de servicios web redirecciona la solicitud al

servicio de autenticación STS de OpenSSO. Una vez instalado el mecanismo de seguridad de Liberty Alliance Project, un agente de seguridad HTTP emite el redireccionamiento. Si se utiliza la seguridad WS-IBS, un agente de seguridad SOAP emitirá el redireccionamiento.

El servicio de autenticación STS de OpenSSO determina el mecanismo de seguridad registrado por el proveedor de servicios web y recupera los tokens de seguridad adecuados. Tras realizar con éxito la autenticación, el cliente de servicios web proporciona el cuerpo del mensaje SOAP, mientras que el agente de seguridad SOAP del cliente de servicios web introduce un encabezado de seguridad y un token. A continuación, se firma el mensaje antes de enviar la solicitud a WSP.

El agente de seguridad SOAP del proveedor de servicios web verifica la firma y el token de seguridad de la solicitud SOAP antes de reenviarla al propio proveedor de servicios web. A continuación, el proveedor de servicios web la procesa y devuelve una respuesta, firmada por el agente de seguridad SOAP, al cliente de servicios web. El agente de seguridad SOAP del cliente de servicios web verifica la firma antes de reenviar la respuesta al propio cliente de servicios web.

En la siguiente tabla se proporcionan una lista y descripciones breves de los tokens admitidos para las transacciones de Liberty Alliance Project.

TABLA 3–1 Tokens del solicitante - Liberty Alliance Project

Token	Cumple estos requisitos
X.509	■ El servicio web seguro utiliza una infraestructura de claves públicas (PKI) en la que el cliente de servicios web proporciona una clave pública a fin de identificar al solicitante y realizar la autenticación en el proveedor de servicios web. ■ El servicio web seguro utiliza una infraestructura de claves públicas (PKI) en la que el cliente de servicios web proporciona una clave pública a fin de identificar al solicitante y realizar la autenticación en el proveedor de servicios web.
Token de portador	■ El servicio web seguro utiliza el método de confirmación de tokens de portador de SAML (Lenguaje de marcado de aserciones de seguridad). ■ WSC proporciona una aserción SAML con información de clave pública a fin de autenticar al solicitante en el proveedor de servicios web. ■ Una segunda firma enlaza la aserción al mensaje SOAP. ■ El enlace de la segunda firma utiliza las reglas definidas por Liberty Alliance Project.

TABLA 3–1 Tokens del solicitante - Liberty Alliance Project *(Continuación)*

Token de SAML	■ El servicio web seguro utiliza el método de confirmación de titular de clave de SAML. ■ WSC agrega una aserción SAML y una firma digital a un encabezado SOAP. ■ También se proporciona una clave pública o un certificado del remitente con la firma. ■ El envío se procesa mediante las reglas definidas por Liberty Alliance Project.

En las siguientes tablas se proporcionan una lista y descripciones breves de los tokens admitidos para las transacciones de WS-IBS.

TABLA 3–2 Tokens del solicitante - WS-IBS

Token	**Cumple estos requisitos**
Nombre de usuario	■ El servicio web seguro necesita un nombre de usuario, una contraseña y, de forma opcional, una solicitud firmada. ■ El consumidor del servicio web proporciona un token de nombre de usuario a fin de identificar al solicitante. ■ El consumidor del servicio web proporciona una contraseña, un secreto compartido u otro elemento equivalente a una contraseña para autenticar la identidad en el proveedor de servicios web.
X.509	El servicio web seguro utiliza una infraestructura de claves públicas (PKI) en la que el consumidor del servicio web proporciona una clave pública a fin de identificar al solicitante y realizar la autenticación en el proveedor de servicios web.
Titular de clave de SAML	■ El servicio web seguro utiliza el método de confirmación de titular de clave de SAML. ■ El consumidor del servicio web proporciona una aserción SAML con información de clave pública a fin de autenticar al solicitante en el proveedor de servicios web. ■ Una segunda firma enlaza la aserción a los datos útiles SOAP.
Vales del remitente de SAML	■ El servicio web seguro utiliza el método de confirmación de titular de vales del remitente de SAML. ■ El consumidor del servicio web agrega una aserción SAML y una firma digital a un encabezado SOAP. También se proporciona una clave pública o un certificado del remitente con la firma.

Uso de la matriz de creación de tokens de seguridad

Use la matriz de creación de tokens de seguridad para ayudarle a configurar el servicio STS de OpenSSO a fin de generar el token de seguridad del cliente de servicios web necesario para el proveedor de servicios web. En primer lugar, en la columna titulada Token de salida del servicio STS de OpenSSO, se proporciona una descripción para cumplir los requisitos de tokens del proveedor de servicios web. A continuación, utilice los valores de parámetros de la misma fila al configurar el servicio de token de seguridad. La "Leyenda de la matriz de creación de tokens" proporciona información sobre los encabezados de la tabla y las opciones disponibles. Consulte la sección 5.2.3, "Para configurar el servicio de token de seguridad" para obtener instrucciones de configuración detalladas. Para obtener información general sobre la seguridad de servicios web y la terminología relacionada, consulte:

- http://www.oracle.com/technology/tech/standards/pdf/security.pdf
- http://download.oracle.com/docs/cd/E15523_01/web.1111/b32511/intro_security.htm#CDDHHG

En la matriz de creación de tokens de seguridad, se proporciona un resumen de la configuración de los parámetros del servicio de token de seguridad utilizados frecuentemente y los tipos de tokens de seguridad que genera el servicio STS de OpenSSO en función de estos valores.

TABLA 3–3 Matriz de creación de tokens de seguridad

Fila	Enlace de seguridad de nivel de mensaje	Token del cliente de servicios web	Tipo de clave	Token "En nombre de"	Uso de la clave	Token de salida del servicio STS de OpenSSO
1	Asimétrico	X509	Portador	Sí	No	Portador de SAML, sin clave de prueba
2	Asimétrico	Nombre de usuario	Portador	Sí	No	Portador de SAML, sin clave de prueba
3	Asimétrico	X509	Portador	No	No	Portador de SAML, sin clave de prueba
4	Asimétrico	Nombre de usuario	Portador	No	No	Portador de SAML, sin clave de prueba

TABLA 3–3 Matriz de creación de tokens de seguridad *(Continuación)*

5	Asimétrico	X509	Simétrica	Sí	No	Titular de clave de SAML, clave de prueba simétrica
6	Asimétrico	Nombre de usuario	Simétrica	Sí	No	Titular de clave de SAML, clave de prueba simétrica
7	Asimétrico	X509	Simétrica	No	No	Titular de clave de SAML, clave de prueba simétrica
8	Asimétrico	Nombre de usuario	Simétrica	No	No	Titular de clave de SAML, clave de prueba simétrica
9	Asimétrico	X509	Asimétrico	No	Clave pública del cliente de servicios web	Titular de clave de SAML, clave de prueba asimétrica
10	Asimétrico	X509	Clave patentada de Oracle para vales del remitente de SAML	Sí	No	Vales del remitente, sin clave de prueba
11	Asimétrico	Nombre de usuario	Clave patentada de Oracle para vales del remitente de SAML	Sí	No	Vales del remitente, sin clave de prueba
12	Asimétrico	X509	Clave patentada de Oracle para vales del remitente de SAML	No	No	ERROR

TABLA 3–3 Matriz de creación de tokens de seguridad *(Continuación)*

13	Asimétrico	Nombre de usuario	Clave patentada de Oracle para vales del remitente de SAML	No	No	ERROR
14	Transporte	Nombre de usuario	Portador	Sí	No	Portador de SAML, sin clave de prueba
15	Transporte	Nombre de usuario	Portador	No	No	Portador de SAML, sin clave de prueba
16	Transporte	Nombre de usuario	Simétrica	Sí	No	Titular de clave de SAML, clave de prueba simétrica
17	Transporte	Nombre de usuario	Simétrica	No	No	Titular de clave de SAML, clave de prueba simétrica
18	Transporte	Nombre de usuario	Clave patentada de Oracle para vales del remitente de SAML	Sí	No	Vales del remitente, sin clave de prueba
19	Transporte	Nombre de usuario	Clave patentada de Oracle para vales del remitente de SAML	No	No	ERROR
20	Asimétrico	Nombre de usuario	Asimétrico	No	Clave pública del cliente de servicios web	ERROR
21	Transporte	Nombre de usuario	Asimétrico	No	Clave pública del cliente de servicios web	ERROR

TABLA 3–3 Matriz de creación de tokens de seguridad *(Continuación)*

22	Asimétrico	X509	Asimétrico	Sí	No	ERROR
23	Asimétrico	Nombre de usuario	Asimétrico	Sí	No	ERROR
24	Transporte	Nombre de usuario	Asimétrico	Sí	No	ERROR
25	Asimétrico	X509	Asimétrico	No	No	Titular de clave de SAML, clave de prueba asimétrica
26	Asimétrico	X509	No	No	No	Titular de clave de SAML, clave de prueba asimétrica
27	Asimétrico	Nombre de usuario	No	No	No	Titular de clave de SAML, clave de prueba simétrica
28	Transporte	Nombre de usuario	No	No	No	Titular de clave de SAML, clave de prueba simétrica

Problemas y soluciones del servicio de token de seguridad

Pendiente de desarrollo

Problemas de configuración y soluciones

Pendiente de desarrollo

Erratas en la documentación

Pendiente de desarrollo

◆ ◆ ◆ CAPÍTULO 4

Uso de Oracle OpenSSO Fedlet

En esta sección se proporciona información sobre Oracle OpenSSO Fedlet:

Acerca de Oracle OpenSSO Fedlet

Oracle OpenSSO Fedlet es una implementación de proveedor de servicios de peso ligero (SP) que puede desplegarse en una aplicación de proveedor de servicios de Java o .NET, lo que permite a la aplicación comunicarse con un proveedor de servicios (IDP) como, por ejemplo, Oracle OpenSSO 8.0 Update 2 mediante el protocolo SAMLv2. Fedlet presenta dos versiones en función de la plataforma utilizada:

- El Fedlet de Java se publicó por primera vez en OpenSSO 8.0. Para obtener información, consulte el Capítulo 5, "Using the OpenSSO Enterprise Fedlet to Enable Identity Federation" de *Sun OpenSSO Enterprise 8.0 Deployment Planning Guide*.
- El Fedlet de .NET se lanzó en OpenSSO 8.0 Update 1. Para obtener información, consulte el Capítulo 10, "Using the ASP.NET Fedlet with OpenSSO Enterprise 8.0 Update 1" de *Sun OpenSSO Enterprise 8.0 Update 1 Release Notes*.

En Oracle OpenSSO 8.0 Update 2, el Fedlet está disponible de la siguiente forma:

- Una vez descomprimido el archivo ZIP de OpenSSO 8.0 Update 2, tanto el Fedlet de Java como el de .NET están disponibles en el siguiente archivo:

  zip-root`/opensso/fedlet/fedlet-unconfigured.zip`, donde *zip-root* es la ubicación en la que se ha descomprimido el archivo ZIP Oracle OpenSSO 8.0.
- Una vez instalado Oracle OpenSSO 8.0 Update 2, puede crear el Fedlet de Java en la consola de administración de OpenSSO 8.0 mediante el flujo de trabajo de creación de Fedlet que se encuentra debajo de Tareas comunes.

Requisitos de Oracle OpenSSO Fedlet

El Fedlet presenta los siguientes requisitos:

- Un contenedor web compatible de Oracle OpenSSO 8.0 Update 2 si tiene intención de implementar el archivo `fedlet.war` o una aplicación de proveedor de servicios de Java integrada en el Fedlet. Consulte la sección "Requisitos de hardware y software para OpenSSO 8.0 Update 2" en la página 12.
- Servicios de Internet Information Server (IIS) 7.0 de Microsoft y posterior si tiene intención de implementar el Fedlet de .NET.
- JDK 1.6.x y posterior.

Configuración de Oracle OpenSSO Fedlet

En esta sección se describe cómo configurar inicialmente el Fedlet con una aplicación de proveedor de servicios:

- "Para configurar el Fedlet de Java" en la página 46
- "Para configurar el Fedlet de .NET" en la página 48

Una vez finalizada la configuración inicial del Fedlet, continúe con cualquier tarea de configuración adicional que desee realizar. Debe tener en cuenta las siguientes consideraciones:

- Si modifica el archivo `sp.xml` del Fedlet, debe volver a importarlo en el proveedor de identidades.
- Si realiza otros cambios de configuración del Fedlet en el proveedor de servicios, proporcione esta información al administrador del proveedor de identidades para que puedan aplicarse en este proveedor.

▼ Para configurar el Fedlet de Java

1 **En el proveedor de identidades, genere los metadatos XML para este proveedor y guárdelos en un archivo denominado `idp.xml`.**

En Oracle OpenSSO 8.0 Update 2, utilice `exportmetadata.jsp`. Por ejemplo:

```
http://opensso-idp.example.com:8080/opensso/saml2/jsp/exportmetadata.jsp
```

2 **En el proveedor de servicios, descomprima el archivo ZIP del Fedlet (si es necesario).**

3 Cree el directorio principal del Fedlet, es decir, el directorio en el que el Fedlet leerá sus metadatos, el círculo de confianza y los archivos de propiedades de configuración.

La ubicación predeterminada es el subdirectorio del Fedlet en el directorio principal del usuario que ejecuta el contenedor web del Fedlet (indicado por la propiedad `user.home` de JVM). Por ejemplo, si este directorio principal es `/home/webservd`, el directorio principal del Fedlet será el siguiente:

```
/home/webservd/fedlet
```

Para cambiar el directorio principal predeterminado del Fedlet, establezca el valor de la propiedad de tiempo de ejecución `com.sun.identity.fedlet.home` de JVM en la ubicación que desee. Por ejemplo:

```
-Dcom.sun.identity.fedlet.home=/export/fedlet/conf
```

A continuación, el Fedlet lee sus metadatos, el círculo de confianza y los archivos de configuración del directorio `/export/fedlet/conf`.

4 Copie los siguientes archivos del directorio `java/conf` del Fedlet de Java en el directorio principal del Fedlet:

- `sp.xml-template`
- `sp-extended.xml-template`
- `idp-extended.xml-template`
- `fedlet.cot-template`

5 En el directorio principal del Fedlet, cambie los nombres de los archivos copiados y elimine `-template` en cada uno de ellos.

6 En los archivos del directorio principal del Fedlet que ha copiado y cuyo nombre ha cambiado, sustituya las etiquetas, tal y como se muestra en la siguiente tabla:

Etiqueta	Sustituir por
`FEDLET_COT`	El nombre del círculo de confianza (COT) del que son miembros el proveedor de identidades remoto y la aplicación del proveedor de servicios del Fedlet de Java.
`FEDLET_ENTITY_ID`	El Id. (nombre) de la aplicación del proveedor de servicios del Fedlet de Java. Por ejemplo: `fedletsp`
`FEDLET_PROTOCOL`	El protocolo del contenedor web de la aplicación del proveedor de servicios del Fedlet de Java como, por ejemplo, `fedlet.war`. Por ejemplo: `https`
`FEDLET_HOST`	El nombre de host del contenedor web de la aplicación del proveedor de servicios del Fedlet de Java como, por ejemplo, `fedlet.war`. Por ejemplo: `fedlet-host.example.com`

Etiqueta	Sustituir por
FEDLET_PORT	El número de puerto del contenedor web de la aplicación del proveedor de servicios del Fedlet de Java como, por ejemplo, `fedlet.war`. Por ejemplo: `80`
FEDLET_DEPLOY_URI	La URL de la aplicación del proveedor de servicios del Fedlet de Java. Por ejemplo: `http://fedletsp.example.com/myFedletApp`
IDP_ENTITY_ID	El Id. (nombre) del proveedor de identidades remoto. Por ejemplo: `openssoidp`

Nota: si el Id. de entidad del proveedor de servicios o identidades del Fedlet contiene un símbolo de porcentaje (%) o una coma (,), debe establecer una secuencia de escape para el carácter antes de sustituirlo en el archivo `fedlet.cot`. Por ejemplo, cambie "%"por "%25" y "," por "%2C".

7. **Copie el archivo `FedletConfiguration.properties` del directorio `java/conf` del Fedlet de Java en el directorio principal del Fedlet.**

8. **Copie el archivo XML de metadatos estándar del proveedor de identidades (indicado en el paso 1) en el directorio principal del Fedlet. A este archivo se le debe asignar el nombre `idp.xml`.**

9. **Importe el archivo de metadatos XML del Fedlet de Java (`sp.xml`) en el proveedor de identidades.**

   En Oracle OpenSSO 8.0 Update 2, utilice el flujo de trabajo de registro del proveedor de servicios remoto que se encuentra debajo de Tareas comunes en la consola de administración de OpenSSO 8.0 para importar los metadatos del proveedor de servicios del Fedlet de Java y agregar este proveedor a un círculo de confianza.

Pasos siguientes En función de los requisitos, continúe con las tareas de configuración adicionales del Fedlet de Java.

▼ Para configurar el Fedlet de .NET

1. **En el proveedor de identidades, genere los metadatos XML para este proveedor y guárdelos en un archivo denominado `idp.xml`.**

   En Oracle OpenSSO 8.0 Update 2, utilice `exportmetadata.jsp`. Por ejemplo:

   `http://opensso-idp.example.com:8080/opensso/saml2/jsp/exportmetadata.jsp`

2. **En el proveedor de servicios, descomprima el archivo ZIP del Fedlet (si es necesario).**

3. **Copie los siguientes archivos de la carpeta `asp.net/conf` del Fedlet de .NET en la carpeta `App_Data` de la aplicación.**

   - `sp.xml-template`
   - `sp-extended.xml-template`
   - `idp-extended.xml-template`

- `fedlet.cot-template`

4. **En la carpeta `App_Data`, cambie el nombre de los archivos y elimine `-template` en cada uno de ellos.**

5. **En los archivos de la carpeta `App_Data` que ha copiado y cuyo nombre ha cambiado, sustituya las etiquetas, tal y como se muestra en la siguiente tabla:**

Etiqueta	Sustituir por
`FEDLET_COT`	El nombre del círculo de confianza (COT) del que son miembros el proveedor de identidades remoto y la aplicación del proveedor de servicios del Fedlet de .NET.
`FEDLET_ENTITY_ID`	El Id. (nombre) de la aplicación del proveedor de servicios del Fedlet de .NET. Por ejemplo: `fedletsp`
`FEDLET_DEPLOY_URI`	La URL de la aplicación del proveedor de servicios del Fedlet de .NET. Por ejemplo: `http://fedletsp.example.com/myFedletApp`
`IDP_ENTITY_ID`	El Id. (nombre) del proveedor de identidades remoto. Por ejemplo: `openssoidp`

6. **Copie el archivo XML de metadatos estándar del proveedor de identidades (indicado en el paso 1) en la carpeta `App_Data` de la aplicación. A este archivo se le debe asignar el nombre `idp.xml`.**

7. **Copie los archivos `Fedlet.dll` y `Fedlet.dll.config` de la carpeta `asp.net/bin` del Fedlet de .NET en la carpeta `bin` de la aplicación.**

8. **Importe el archivo de metadatos XML del Fedlet de .NET (`sp.xml`) en el proveedor de identidades.**

   En Oracle OpenSSO 8.0 Update 2, utilice el flujo de trabajo de registro del proveedor de servicios remoto que se encuentra debajo de Tareas comunes en la consola de administración de OpenSSO 8.0 para importar los metadatos del proveedor de servicios del Fedlet de .NET y agregar este proveedor a un círculo de confianza.

Pasos siguientes En función de los requisitos, continúe con las tareas de configuración adicionales del Fedlet de .NET.

Nuevas funciones del Fedlet en OpenSSO 8.0 Update 2

Oracle OpenSSO 8.0 Update 2 incluye las siguientes nuevas funciones para el Fedlet:

Información sobre la versión del Fedlet (CR 6941387)

Oracle OpenSSO Fedlet incluye información sobre la versión. Después de extraer los archivos del paquete (archivo ZIP) del Fedlet, consulte los siguientes archivos para determinar la versión del Fedlet:

- Fedlet de Java: `java/conf/FederationConfig.properties`
- Fedlet de .NET: `asp.net/bin/Fedlet.dll.config`

Cifrado y descifrado de contraseñas del Fedlet de Java (CR 6930477)

El Fedlet de Java proporciona el archivo `fedletEncode.jsp` de `fedlet.war` para cifrar las contraseñas `storepass` y `keypass`. De manera predeterminada se genera una clave de cifrado diferente para cada Fedlet. Para cambiar esta clave de cifrado, establezca la propiedad `am.encryption.pwd` en el archivo `FederationConfig.properties` del Fedlet.

Compatibilidad del Fedlet de Java con las firmas y el cifrado

El Fedlet de Java admite la verificación de firmas XML y el descifrado de elementos `assertion` y `NameID` cifrados y sus correspondientes atributos.

▼ Para configurar el Fedlet de Java para las firmas y el cifrado

1. **Cree un archivo de almacén de claves denominado `keystore.jks` mediante la utilidad `keytool`.**
2. **Agregue las claves privadas (y los certificados públicos si corresponde) utilizadas para las firmas y el cifrado en el archivo `keystore.jks`.**
3. **Cree un archivo `.storepass`.**
4. **Agregue la contraseña al archivo `.storepass`. Para cifrar la contraseña, utilice `fedletEncode.jsp`.**
5. **Cree un archivo `.keypass`.**
6. **Agregue la contraseña al archivo `.keypass`. Para cifrar la contraseña, utilice `fedletEncode.jsp`.**
7. **Si utiliza contraseñas de texto no cifrado, incluya entre comentarios la siguiente línea del archivo `FederationConfig.properties`:**

   ```
   com.sun.identity.saml.xmlsig.passwordDecoder=
       com.sun.identity.fedlet.FedletEncodeDecode
   ```

8. **Establezca la ruta completa para los siguientes atributos del archivo `FederationConfig.properties` , donde *path* hace referencia a la ruta completa al archivo respectivo:**

   ```
   com.sun.identity.saml.xmlsig.keystore=path/keystore.jks
   com.sun.identity.saml.xmlsig.storepass=path/.storepass
   com.sun.identity.saml.xmlsig.keypass=path/.keypass
   ```

9. **Utilice `keytool` para exportar el certificado de firma. Por ejemplo:**

   ```
   keytool -export -keystore keystore.jks -rfc -alias test
   ```

   La herramienta le solicita que especifique la contraseña utilizada para acceder a `keystore.jks` y, a continuación, genera el certificado.

10. **Si necesita un certificado de cifrado, utilice `keytool` para exportarlo, como se muestra en el paso anterior (o utilice el mismo certificado para las firmas y el cifrado).**

11 Cree un bloque XML de `KeyDescriptor` y agregue el certificado de cifrado a este. Por ejemplo, tenga en cuenta la etiqueta `use="signing"` del elemento `KeyDescriptor`:

```
<KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>
MIICQDCCAakCBEeNB0swDQYJKoZIhvcNAQEEBQAwZzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExFDASBgNVBAcTC1NhbnRhIENsYXJhMQwwCgYDVQQKEwNTdW4xEDAOBgNVBAsTB09w
ZW5TU08xDTALBgNVBAMTBHRlc3QwHhcNMDgwMTE1MTkxOTM5WhcNMTgwMTEyMTkxOTM5WjBnMQsw
CQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEUMBIGA1UEBxMLU2FudGEgQ2xhcmExDDAK
BgNVBAoTA1N1bjEQMA4GA1UECxMHT3BlblNTTzENMAsGA1UEAxMEdGVzdDCBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEArSQc/U75GB2AtKhbGS5piiLkmJzqEsp64rDxbMJ+xDrye0EN/q1U5Of\+
RkDsaN/igkAvV1cuXEgTL6RlafFPcUX7QxDhZBhsYF9pbwtMzi4A4su9hnxIhURebGEmxKW9qJNY
Js0Vo5+IgjxuEWnjnnVgHTs1+mq5QYTA7E6ZyL8CAwEAATANBgkqhkiG9w0BAQQFAAOBgQB3Pw/U
QzPKTPTYi9upbFXlrAKMwtFf2OW4yvGWWvlcwcNSZJmTJ8ARvVYOMEVNbsT4OFcfu2/PeYoAdiDA
cGy/F2Zuj8XJJpuQRSE6PtQqBuDEHjjmOQJ0rV/r8mO1ZCtHRhpZ5zYRjhRC9eCbjx9VrFax0JDC
/FfwWigmrW0Y0Q==
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
</KeyDescriptor>
```

12 Cree otro bloque XML de `KeyDescriptor` y agregue el certificado de cifrado a este. Por ejemplo, tenga en cuenta la etiqueta `use="encryption"` del elemento `KeyDescriptor`:

```
<KeyDescriptor use="encryption">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
          <X509Certificate>
MIICQDCCAakCBEeNB0swDQYJKoZIhvcNAQEEBQAwZzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExFDASBgNVBAcTC1NhbnRhIENsYXJhMQwwCgYDVQQKEwNTdW4xEDAOBgNVBAsTB09w
ZW5TU08xDTALBgNVBAMTBHRlc3QwHhcNMDgwMTE1MTkxOTM5WhcNMTgwMTEyMTkxOTM5WjBnMQsw
CQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEUMBIGA1UEBxMLU2FudGEgQ2xhcmExDDAK
BgNVBAoTA1N1bjEQMA4GA1UECxMHT3BlblNTTzENMAsGA1UEAxMEdGVzdDCBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEArSQc/U75GB2AtKhbGS5piiLkmJzqEsp64rDxbMJ+xDrye0EN/q1U5Of\+
RkDsaN/igkAvV1cuXEgTL6RlafFPcUX7QxDhZBhsYF9pbwtMzi4A4su9hnxIhURebGEmxKW9qJNY
Js0Vo5+IgjxuEWnjnnVgHTs1+mq5QYTA7E6ZyL8CAwEAATANBgkqhkiG9w0BAQQFAAOBgQB3Pw/U
QzPKTPTYi9upbFXlrAKMwtFf2OW4yvGWWvlcwcNSZJmTJ8ARvVYOMEVNbsT4OFcfu2/PeYoAdiDA
cGy/F2Zuj8XJJpuQRSE6PtQqBuDEHjjmOQJ0rV/r8mO1ZCtHRhpZ5zYRjhRC9eCbjx9VrFax0JDC
/FfwWigmrW0Y0Q==
          </X509Certificate>
        </X509Data>
      </KeyInfo>
<EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc">
  <KeySize xmlns="http://www.w3.org/2001/04/xmlenc#">128</KeySize>
</EncryptionMethod>
</KeyDescriptor>
```

13 En el archivo `sp.xml` del Fedlet de Java, agregue los bloques XML con los certificados de firma y cifrado en el elemento `SPSSODescriptor`. Para obtener un elemento `SPSSODescriptor` de muestra, consulte el Ejemplo 4–1.

El atributo `AuthnRequestsSigned` se establece en `true` (verdadero), lo que configura el Fedlet de Java para que firme todas las solicitudes de autenticación.

14 En el archivo `sp-extended.xml` del Fedlet de Java, establezca los valores de los siguientes elementos:

- `signingCertAlias` contiene el alias del certificado de firma XML del almacén de claves.
- `encryptionCertAlias` contiene el alias del certificado de cifrado XML del almacén de claves.

15 Para forzar el cifrado de determinados elementos por parte del proveedor de servicios del Fedlet de Java, establezca los siguientes atributos del archivo `sp-extended.xml` en `true` (verdadero):

- `wantAssertionEncrypted`
- `wantNameIDEncrypted`
- `wantAttributeEncrypted`

16 Para forzar la firma de determinados elementos por parte del proveedor de servicios del Fedlet de Java, así como de los elementos que el proveedor desea que se firmen, establezca los siguientes atributos en `true` (verdadero):

- `wantAuthnRequestsSigned` del archivo `idp.xml` le indica al Fedlet los elementos que debe firmar.
- `AuthnRequestsSigned` y `WantAssertionsSigned` del archivo `sp.xml` le indica al proveedor de identidades los elementos que el Fedlet tiene intención de firmar.
- `wantArtifactResponseSigned` del archivo `sp-extended.xml` le indica al Fedlet los elementos que debe firmar.
- `wantPOSTResponseSigned` del archivo `sp-extended.xml`
- `wantLogoutRequestSigned` del archivo `sp-extended.xml`
- `wantLogoutResponseSigned` del archivo `sp-extended.xml`

Si el proveedor de identidades requiere que se firmen determinados mensajes, establezca los atributos respectivos en `true` (true) en el archivo `idp-extended.xml` . Por ejemplo, `wantLogoutRequestSigned` y `wantLogoutResponseSigned`.

Nota – Si establece atributos en el archivo `sp-extended.xml`, proporcione esta información al administrador del proveedor de identidades para que puedan realizarse los cambios de configuración necesarios en este.

---

17 Reinicie el contenedor web del Fedlet de Java.

18 Importe el archivo `sp.xml` del Fedlet de Java en el proveedor de identidades.

Ejemplo 4–1 Elemento `SPSSODescriptor` de muestra del Fedlet de Java

```
<EntityDescriptor entityID="fedlet"
xmlns="urn:oasis:names:tc:SAML:2.0:metadata">

<SPSSODescriptor AuthnRequestsSigned="true" WantAssertionsSigned="false"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
<b><KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>
MIICQDCCAakCBEeNB0swDQYJKoZIhvcNAQEEBQAwZzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExFDASBgNVBAcTC1NhbnRhIENsYXJhMQwwCgYDVQQKEwNTdW4xEDAOBgNVBAsTB09w
ZW5TU08xDTALBgNVBAMTBHRlc3QwHhcNMDgwMTE1MTkxOTM5WhcNMTgwMTEyMTkxOTM5WjBnMQsw
CQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEUMBIGA1UEBxMLU2FudGEgQ2xhcmExDDAK
BgNVBAoTA1N1bjEQMA4GA1UECxMHT3BlblNTTzENMAsGA1UEAxMEdGVzdDBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEArSQc/U75GB2AtKhbGS5piiLkmJzqEsp64rDxbMJ+xDrye0EN/q1U5Of\+
RkDsaN/igkAvV1cuXEgTL6RlafFPcUX7QxDhZBhsYF9pbwtMzi4A4su9hnxIhURebGEmxKW9qJNY
Js0Vo5+IgjxuEWnjnnVgHTs1+mq5QYTA7E6ZyL8CAwEAATANBgkqhkiG9w0BAQQFAAOBgQB3Pw/U
QzPKTPTYi9upbFXlrAKMwtFf2OW4yvGWWvlcwcNSZJmTJ8ARvVYOMEVNbsT4OFcfu2/PeYoAdiDA
cGy/F2Zuj8XJJpuQRSE6PtQqBuDEHjjmOQJ0rV/r8mO1ZCtHRhpZ5zYRjhRC9eCbjx9VrFax0JDC
/FfwWigmrW0Y0Q==
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>

</KeyDescriptor></b>
<b><KeyDescriptor use="encryption">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
          <X509Certificate>
MIICQDCCAakCBEeNB0swDQYJKoZIhvcNAQEEBQAwZzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExFDASBgNVBAcTC1NhbnRhIENsYXJhMQwwCgYDVQQKEwNTdW4xEDAOBgNVBAsTB09w
ZW5TU08xDTALBgNVBAMTBHRlc3QwHhcNMDgwMTE1MTkxOTM5WhcNMTgwMTEyMTkxOTM5WjBnMQsw
CQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEUMBIGA1UEBxMLU2FudGEgQ2xhcmExDDAK
BgNVBAoTA1N1bjEQMA4GA1UECxMHT3BlblNTTzENMAsGA1UEAxMEdGVzdDBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEArSQc/U75GB2AtKhbGS5piiLkmJzqEsp64rDxbMJ+xDrye0EN/q1U5Of\+
RkDsaN/igkAvV1cuXEgTL6RlafFPcUX7QxDhZBhsYF9pbwtMzi4A4su9hnxIhURebGEmxKW9qJNY
Js0Vo5+IgjxuEWnjnnVgHTs1+mq5QYTA7E6ZyL8CAwEAATANBgkqhkiG9w0BAQQFAAOBgQB3Pw/U
QzPKTPTYi9upbFXlrAKMwtFf2OW4yvGWWvlcwcNSZJmTJ8ARvVYOMEVNbsT4OFcfu2/PeYoAdiDA
cGy/F2Zuj8XJJpuQRSE6PtQqBuDEHjjmOQJ0rV/r8mO1ZCtHRhpZ5zYRjhRC9eCbjx9VrFax0JDC
/FfwWigmrW0Y0Q==
          </X509Certificate>
        </X509Data>
      </KeyInfo>

<EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc">
<KeySize xmlns="http://www.w3.org/2001/04/xmlenc#">128</KeySize>
</EncryptionMethod>
</KeyDescriptor></b>
<NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:transient</NameIDFormat
><AssertionConsumerService index="1"
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="http://server.sun.com:7070/fedlet/fedletapplication"/>
</SPSSODescriptor>
</EntityDescriptor>
```

Compatibilidad del Fedlet de Java con la consulta de atributos (CR 6930476)

El Fedlet de Java admite la consulta de atributos SAMLv2, lo que permite realizar una consulta en un proveedor de identidades como, por ejemplo, Oracle OpenSSO 8.0 Update 2 para buscar valores de atributos de identidad específicos. Puede configurar el Fedlet para que firme y cifre la consulta. La firma es necesaria para emitir la consulta del Fedlet, aunque el cifrado es opcional.

▼ Para configurar el Fedlet de Java para una consulta de atributos

1. **Habilite la firma XML para firmar la consulta de atributos, como se describe en "Compatibilidad del Fedlet de Java con las firmas y el cifrado" en la página 51.**

2. **Agregue el certificado generado en el paso anterior al elemento `RoleDescriptor` del archivo `sp.xml` del Fedlet. En el siguiente ejemplo, hay dos etiquetas `KeyDescriptor` en las que se debe pegar el certificado. Una es para la firma y la otra para el cifrado. Si no va a habilitar el cifrado, la etiqueta `KeyDescriptor use="encryption"` no es necesaria.**

```
<RoleDescriptor xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:query="urn:oasis:names:tc:SAML:metadata:ext:query"
      xsi:type="query:AttributeQueryDescriptorType"
      protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
  <KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>
            --certificate--
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
  </KeyDescriptor>
  <KeyDescriptor use="encryption">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>
            --certificate--
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    <EncryptionMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc">
<xenc:KeySize
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">128</xenc:KeySize>
    </EncryptionMethod>
  </KeyDescriptor>
</RoleDescriptor>
```

3 **En el archivo `sp-extended.xml` del Fedlet de Java, especifique un valor para el atributo `signingCertAlias` y para el atributo `encryptionCertAlias` si se ha configurado.**

Si tiene intención de configurar el proveedor de identidades para cifrar la aserción, cifre también el elemento `NameID`. De este modo, el valor del atributo `wantNameIDEncrypted` debe establecerse en `true` (verdadero). Agregue el código XMl al elemento `AttributeQueryConfig`. Por ejemplo:

```
<Attribute name="signingCertAlias">
      <Value>test</Value>
</Attribute>
<Attribute name="encryptionCertAlias">
      <Value>test</Value>
</Attribute>
<Attribute name="wantNameIDEncrypted">
      <Value>true</Value>
</Attribute>
```

En este ejemplo, `test` es el alias de la clave de ejemplo.

4 **Importe el archivo de metadatos del Fedlet de Java (`sp.xml`) en el proveedor de identidades.**

Además, realice los siguientes pasos de configuración en el proveedor de identidades para admitir la consulta de atributos del Fedlet.

Cifrado y descifrado de solicitudes y respuestas por parte del Fedlet de .NET (CR 6939005)

El Fedlet de .NET puede cifrar las solicitudes XML salientes y descifrar las respuestas entrantes de los elementos de Id. de nombre, atributo y aserción.

▼ Para configurar el Fedlet de .NET para el cifrado y el descifrado de solicitudes y respuestas

1 **Importe el certificado X.509 en la carpeta Personal de la cuenta de equipo local mediante el complemento de certificados de Microsoft Management Console. Para utilizar este complemento, consulte el siguiente artículo de Microsoft:**

`http://msdn.microsoft.com/es-es/library/ms788967.aspx`

2 **Especifique un nombre descriptivo para este certificado. Para ello, acceda al cuadro de diálogo Propiedades e introduzca un valor. (Guarde este valor para el paso 4).**

3. **Consulte los permisos adecuados para permitir el acceso de lectura al certificado para la cuenta de usuario utilizada por los Servicios de Internet Information Server (IIS), como se describe en el artículo de Microsoft. Por ejemplo:**

   a. **En el complemento de certificados, acceda a Acción, Todas las tareas y, a continuación, en Administrar claves privadas.**

   b. **Especifique los permisos de acceso de lectura para la cuenta de usuario que ejecute ISS (normalmente SERVICIO DE RED).**

4. **En el archivo de metadatos ampliado del Fedlet de .NET (`sp-extended.xml`), especifique el nombre descriptivo indicado en el paso 2 como valor del atributo `encryptionCertAlias`. Por ejemplo:**

```
<Attribute name="encryptionCertAlias">
<Value>MyFedlet</Value>
```

5. **En el archivo de metadatos del proveedor de servicios del Fedlet de .NET (`sp.xml`), agregue `KeyDescriptor` en la clave de cifrado.**

   Utilice el complemento de certificados de la instancia de Microsoft Management Console utilizada anteriormente para exportar la clave pública del certificado con codificación Base64 para incluirla en el bloque XML de `KeyDescriptor`. Esta instancia de `KeyDescriptor` debe ser el primer elemento principal en `SPSSODescriptor`. Por ejemplo:

```
<KeyDescriptor use="encryption">
           <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
             <ds:X509Data>
               <ds:X509Certificate>
MIICQDCCAakCBEeNB0swDQYJKoZIhvcNAQEEBQAwZzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExFDASBgNVBAcTC1NhbnRhIENsYXJhMQwwCgYDVQQKEwNTdW4xEDAOBgNVBAsTB09w
ZW5TU08xDTALBgNVBAMTBHRlc3QwHhcNMDgwMTE1MTkxOTM5WhcNMTgwMTEyMTkxOTM5WjBnMQsw
CQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEUMBIGA1UEBxMLU2FudGEgQ2xhcmExDDAK
BgNVBAoTA1N1bjEQMA4GA1UECxMHT3BlblNTTzENMAsGA1UEAxMEdGVzdDCBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEArSQc/U75GB2AtKhbGS5piiLkmJzqEsp64rDxbMJ+xDrye0EN/q1U5Of\+
RkDsaN/igkAvV1cuXEgTL6RlafFPcUX7QxDhZBhsYF9pbwtMzi4A4su9hnxIhURebGEmxKW9qJNY
Js0Vo5+IgjxuEWnjnnVgHTs1+mq5QYTA7E6ZyL8CAwEAATANBgkqhkiG9w0BAQQFAAOBgQB3Pw/U
QzPKTPTYi9upbFXlrAKMwtFf2OW4yvGWWvlcwcNSZJmTJ8ARvVYOMEVNbsT4OFcfu2/PeYoAdiDA
cGy/F2Zuj8XJJpuQRSE6PtQqBuDEHjjmOQJ0rV/r8mO1ZCtHRhpZ5zYRjhRC9eCbjx9VrFax0JDC
/FfwWigmrW0Y0Q==
               </ds:X509Certificate>
             </ds:X509Data>
           </ds:KeyInfo>
           <EncryptionMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc">
           <KeySize
xmlns="http://www.w3.org/2001/04/xmlenc#">128</KeySize>
           </EncryptionMethod>
</KeyDescriptor>
```

6. **Reinicie el conjunto de aplicaciones asociado a la aplicación de .NET.**

Pasos siguientes Para probar esta configuración, utilice la aplicación de muestra. Además, establezca los siguientes atributos para cifrar las solicitudes y descifrar las respuestas con el proveedor de identidades y los cambios adecuados efectuados en los metadatos configurados:

- Aserción: establezca el atributo `wantAssertionEncrypted` del archivo de metadatos `sp-extended.xml` en `true` (verdadero) para que el Fedlet de .NET descifre el elemento `EncryptedAssertion` en las respuestas entrantes del proveedor de identidades.
- Atributo: establezca el atributo `wantAttributeEncrypted` del archivo de metadatos `sp-extended.xml` en `true` (verdadero) para que el Fedlet de .NET descifre el elemento `EncryptedAttribute` en las respuestas entrantes del proveedor de identidades.
- Id. de nombre: establezca el atributo `wantNameIDEncrypted` del archivo de metadatos `idp-extended.xml` en `true` (verdadero) para que el Fedlet de .NET cifre el elemento `NameID` en las solicitudes salientes. Establezca este mismo atributo en `sp-extended.xml` para que el Fedlet de .NET descifre el elemento `EncryptedID` en las respuestas entrantes del proveedor de identidades.

Firma de solicitudes y respuestas por parte del Fedlet de .NET (CR 6928530)

El Fedlet de .NET admite la firma de solicitudes XML entrantes como, por ejemplo, las solicitudes `Authn` y de cierre de sesión.

▼ Para configurar el Fedlet de .NET para la firma de solicitudes y respuestas:

1. **Importe el certificado X.509 en la carpeta Personal de la cuenta de equipo local mediante el complemento de certificados de Microsoft Management Console. Para utilizar este complemento, consulte el siguiente artículo de Microsoft:**

   `http://msdn.microsoft.com/es-es/library/ms788967.aspx`

2. **Especifique un nombre descriptivo para este certificado. Para ello, acceda al cuadro de diálogo Propiedades e introduzca un valor. (Guarde este valor para el paso 4).**

3. **Consulte los permisos adecuados para permitir el acceso de lectura al certificado para la cuenta de usuario utilizada por los Servicios de Internet Information Server (IIS), como se describe en el artículo de Microsoft. Por ejemplo:**

   a. **En el complemento de certificados, acceda a Acción, Todas las tareas y, a continuación, a Administrar claves privadas.**

   b. **Especifique los permisos de acceso de lectura para la cuenta de usuario que ejecute ISS (normalmente SERVICIO DE RED).**

4. **En el archivo de metadatos ampliado del Fedlet de .NET (`sp-extended.xml`), especifique el nombre descriptivo indicado en el paso 2 como valor del atributo `signingCertAlias`. Por ejemplo:**

```
<Attribute name="signingCertAlias">
<Value>MyFedlet</Value>
```

5. **En el archivo de metadatos del proveedor de servicios del Fedlet de .NET (`sp.xml`), agregue `KeyDescriptor` en la clave de firma.**

   Utilice el complemento de certificados de la instancia de Microsoft Management Console utilizada anteriormente para exportar la clave pública del certificado con codificación Base64 para incluirla en el bloque XML de `KeyDescriptor`. Esta instancia de `KeyDescriptor` debe ser el primer elemento principal en `SPSSODescriptor`. Por ejemplo:

```
<KeyDescriptor use="signing">
             <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                <ds:X509Data>
                   <ds:X509Certificate>
MIICQDCCAakCBEeNB0swDQYJKoZIhvcNAQEEBQAwZzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExFDASBgNVBAcTC1NhbnRhIENsYXJhMQwwCgYDVQQKEwNTdW4xEDAOBgNVBAsTB09w
ZW5TU08xDTALBgNVBAMTBHRlc3QwHhcNMDgwMTE1MTkxOTM5WhcNMTgwMTEyMTkxOTM5WjBnMQsw
CQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEUMBIGA1UEBxMLU2FudGEgQ2xhcmExDDAK
BgNVBAoTA1N1bjEQMA4GA1UECxMHT3BlblNTTzENMAsGA1UEAxMEdGVzdDBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEArSQc/U75GB2AtKhbGS5piiLkmJzqEsp64rDxbMJ+xDrye0EN/q1U5Of\+
RkDsaN/igkAvV1cuXEgTL6RlafFPcUX7QxDhZBhsYF9pbwtMzi4A4su9hnxIhURebGEmxKW9qJNY
Js0Vo5+IgjxuEWnjnnVgHTs1+mq5QYTA7E6ZyL8CAwEAATANBgkqhkiG9w0BAQQFAAOBgQB3Pw/U
QzPKTPTYi9upbFXlrAKMwtFf2OW4yvGWWvlcwcNSZJmTJ8ARvVYOMEVNbsT4OFcfu2/PeYoAdiDA
cGy/F2Zuj8XJJpuQRSE6PtQqBuDEHjjmOQJ0rV/r8mO1ZCtHRhpZ5zYRjhRC9eCbjx9VrFax0JDC
/FfwWigmrW0Y0Q==
                   </ds:X509Certificate>
                </ds:X509Data>
             </ds:KeyInfo>
</KeyDescriptor>
```

6. **Reinicie el conjunto de aplicaciones asociado a la aplicación de .NET.**

Cierre de sesión único del Fedlet de .NET (CR 6928528 y CR 6930472)

El Fedlet de .NET admite el cierre de sesión único de los proveedores de identidades y servicios iniciados. Para implementar el cierre de sesión único, la aplicación de muestra del Fedlet de .NET incluye los archivos `logout.aspx` y `spinitiatedslo.aspx` en la carpeta `asp.net/SampleApp`. Para comprobar el funcionamiento de la función de cierre de sesión único del Fedlet, implemente la aplicación de muestra del Fedlet de .NET.

▼ Para configurar la aplicación del proveedor de servicios del Fedlet de .NET para el cierre de sesión único:

1. **Si no ha configurado el Fedlet de .NET, siga los pasos indicados en el archivo `Readme` (Léame).**

2 **Copie los archivos `logout.aspx` y `spinitiatedslo.aspx` en el contenido público de la aplicación de .NET.**

3 **Realice los siguientes cambios en los archivos de configuración de la aplicación:**

- En el archivo `sp.xml`, asegúrese de que la ruta al archivo `logout.aspx` señale a la ubicación correcta del archivo de la aplicación.
- En el archivo `idp.xml` (o durante la configuración del proveedor de identidades), asegúrese de que la ruta al archivo `spinitiatedslo.aspx` señale a la ubicación correcta del archivo de la aplicación.

4 **Si desea que se firmen la solicitud y la respuesta de cierre de sesión, establezca los siguientes atributos en `true` (verdadero) en los archivos `sp-extended.xml` y `idp-extended.xml`:**

- `wantLogoutRequestSigned`
- `wantLogoutResponseSigned`

5 **Importe el archivo de metadatos del proveedor de servicios del Fedlet (`sp.xml`) en el proveedor de identidades.**

Además, informe al administrador del proveedor de identidades de que ha configurado el cierre de sesión único para el proveedor de servicios del Fedlet a fin de que puedan realizarse los cambios adicionales necesarios en la configuración del proveedor de identidades.

Inicio de sesión único del proveedor de servicios del Fedlet de .NET iniciado (CR 6928525)

El Fedlet de .NET admite el inicio de sesión único (SSO) para el proveedor de servicios SAMLv2 iniciado. Además, se necesita disponer de compatibilidad con artefactos para permitir que el Fedlet de .NET reciba un artefacto y, a continuación, lo resuelva mediante SOAP con el servicio de resolución de artefactos del proveedor de identidades emisor.

La aplicación de muestra del Fedlet de .NET muestra cómo configurar el inicio de sesión único. Una vez que la aplicación cuente con los artefactos necesarios instalados, se requiere un URI específico para recibir un elemento HTTP POST que contenga la respuesta SAMLv2 tras una autenticación con éxito por parte del proveedor de identidades. En el siguiente código de ejemplo, se indica cómo se puede recuperar esta información en una aplicación de .NET:

EJEMPLO 4–2 Código de ejemplo para recuperar `AuthnResponse` en una aplicación del Fedlet de .NET

```
AuthnResponse authnResponse = null;
try
{
      ServiceProviderUtility spu = new ServiceProviderUtility(Context);
      authnResponse = spu.GetAuthnResponse(Context);
}
```

EJEMPLO 4–2 Código de ejemplo para recuperar `AuthnResponse` en una aplicación del Fedlet de .NET *(Continuación)*

```
catch (Saml2Exception se)
{
      // invalid AuthnResponse received
}
catch (ServiceProviderUtilityException spue)
{
      // issues with deployment (reading metadata)
}
```

Si la aplicación recibe la respuesta SAMLv2, el objeto `authnResponse` se llenará con la información de aserción. La aplicación de muestra indica cómo recuperar la información de atributos y asunto desde este objeto.

Compatibilidad del Fedlet de .NET con varios proveedores de identidades y el servicio de detección (CR 6928524)

El Fedlet de .NET admite varios proveedores de identidades y su servicio de detección.

En algunas implementaciones, es recomendable configurar el Fedlet de .NET con varios proveedores de identidades como, por ejemplo, Oracle OpenSSO 8.0 Update 2. Realice la siguiente tarea para cada proveedor de identidades adicional que desee agregar.

▼ Para configurar el Fedlet de .NET para varios proveedores de identidades

1. **Obtenga el archivo de metadatos XML del proveedor de identidades adicional.**

2. **Asigne al archivo de metadatos del proveedor de identidades adicional el nombre `idp`*n*`.xml`, donde *n* es el proveedor de identidades que va a agregar. Por ejemplo, asigne el nombre `idp2.xml` al segundo archivo del proveedor de identidades, `idp3.xml` al tercero y así sucesivamente. En este procedimiento se utiliza `idp2.xml` como nombre del archivo.**

3. **Copie el archivo `idp2.xml` del paso 2 en la carpeta `App_Data` de la aplicación.**

4. **Agregue este nuevo proveedor de identidades al círculo de confianza del Fedlet de .NET.**

   Para agregar el nuevo proveedor de identidades a un círculo de confianza existente:

   En el archivo `fedlet.cot` de la carpeta `App_Data` de la aplicación, anexe el Id. de entidad del nuevo IDP (indicado por el atributo `entityID` del archivo de metadatos `idp2.xml`) al valor del atributo `sun-fm-trusted-providers` utilizando una coma (,) como separador.

Para agregar el nuevo proveedor de identidades a un nuevo círculo de confianza:

a. **Cree un archivo nuevo denominado `fedlet2.cot` en la carpeta `App_Data` de la aplicación. Utilice el archivo `fedlet.cot` existente como plantilla. Sin embargo, cambie el valor del atributo `cot-name` por el nombre del nuevo círculo de confianza (por ejemplo, `cot2`). Incluya el Id. de entidad del nuevo proveedor de identidades y el del Fedlet como valor del atributo `sun-fm-trusted-providers` separándolos con una coma (,).**

b. **En el archivo `sp-extended.xml`, agregue el nombre del nuevo círculo de confianza al valor del atributo `cotlist`. Por ejemplo, para un círculo de confianza denominado `cot2`:**

```
<Attribute name="cotlist">
<Value>saml2cot</Value>
<Value>cot2</Value>
</Attribute>
```

5 **En la carpeta `App_Data` de la aplicación, cree un nuevo archivo `idp2-extended.xml` como metadatos ampliados del nuevo proveedor de identidades. Utilice el archivo `idp-extended.xml` existente como plantilla. Sin embargo, cambie el elemento `entityID` por el Id. de entidad del nuevo proveedor de identidades. Cambie el valor del atributo `cotlist` por el nombre del círculo de confianza si se ha creado un nuevo círculo para el proveedor de identidades. Asegúrese de que el proveedor de identidades adicional constituya una identidad remota.**

6 **Reinicie el conjunto de aplicaciones asociado a la aplicación del Fedlet de .NET.**

7 **El archivo XML de metadatos del Fedlet (`sp.xml`) debe importarse en el proveedor de identidades adicional y agregarse al mismo círculo de confianza que el de la entidad del proveedor de identidades. Importe el archivo `sp.xml` en el proveedor de identidades o indique el archivo que debe importarse al administrador del proveedor de identidades.**

Compatibilidad del Fedlet de .NET con el servicio de detección del proveedor de identidades (CR 6928524)

En este caso, el Fedlet de .NET se configura con varios proveedores de identidades en un círculo de confianza y desea configurar el Fedlet para que utilice el servicio de detección a fin de determinar el proveedor de identidades preferido.

Debe configurarse el servicio de detección para los proveedores de identidades que se están utilizando con el Fedlet de .NET. Para obtener información sobre cómo configurar el servicio de detección del proveedor de identidades en Oracle OpenSSO 8.0 Update 2, consulte la siguiente colección de documentación: `http://docs.sun.com/coll/1767.1`.

▼ Para configurar el Fedlet de .NET con el objeto de que utilice el servicio de detección del proveedor de identidades:

1. **En el archivo `fedlet.cot` del Fedlet de .NET, establezca la propiedad `sun-fm-saml2-readerservice-url` en la URL del servicio de lector SAMLv2. Por ejemplo:**

   ```
   sun-fm-saml2-readerservice-url=http://discovery.common.com/opensso/saml2reader
   ```

2. **Reinicie el conjunto de aplicaciones asociado a la aplicación del Fedlet de .NET.**

Problemas generales y soluciones de Oracle OpenSSO Fedlet

Pendiente de desarrollo

Erratas en la documentación

La referencia de la API de Java del Fedlet está disponible en la referencia de la API de Java de Oracle OpenSSO 8.0 Update 2 en la siguiente colección de documentación: `http://docs.sun.com/coll/1767.1`.

Nota – El método `getPolicyDecisionForFedlet` no se admite en la versión OpenSSO 8.0 Update 2.

◆ ◆ ◆ CAPÍTULO 5

Integración de OpenSSO 8.0 Update 2 con Oracle Access Manager

En este capítulo se proporciona información sobre cómo implementar el inicio de sesión único mediante OpenSSO 8.0 Update 2 y Oracle Access Manager 10g u 11g. Esta información complementa la información conceptual presente en el Capítulo 3, "Integrating Oracle Access Manager" de *Sun OpenSSO Enterprise 8.0 Integration Guide*. En este caso práctico, se proporciona un ejemplo de inicio de sesión único en aplicaciones protegidas por OpenSSO mediante una sesión de Oracle Access Manager. El módulo de autenticación de OpenSSO genera una sesión de OpenSSO basada en la sesión de Oracle Access Manager.

Descripción general de los pasos de integración

1. "Antes de la instalación" en la página 65
2. Unpacking the Integration Bits
3. Building source files for Oracle Access Manager in OpenSSO
4. "(Opcional) Generar un esquema de autenticación para OpenSSO en Oracle Access Manager" en la página 69
5. "Configuración del inicio de sesión único mediante Oracle Access Manager y el servicio STS de Oracle OpenSSO" en la página 70
6. "Para probar el inicio de sesión único" en la página 72
7. "(Opcional) Instalación del esquema de autenticación de Oblix en Oracle Access Manager" en la página 73

Antes de la instalación

Asegúrese de tener acceso a los siguientes componentes antes de intentar instalar OpenSSO 8.0 Update 2 para su integración con Oracle Access Manager:

opensso.zip	Este archivo zip contiene el archivo opensso.war, el código fuente de integración, los archivos de

	configuración y otras herramientas necesarias para la instalación y la configuración de OpenSSO 8.0 Update 2.
Agente de OpenSSO	El agente de OpenSSO se utiliza cuando una aplicación protegida por OpenSSO puede utilizar realmente la sesión de autenticación establecida por Oracle Access Manager.
Oracle Access Manager 10g u 11g	Descargue Oracle Access Manager desde el sitio web de Oracle. Consulte la página Descargas de software de Oracle Fusion Middleware 11gR1.
Oracle Web Gate 10g u 11g	Descargue Oracle Webgate para un contenedor compatible tanto con OpenSSO como con esta aplicación. Actualmente Sun Web Server 7.x es el único contenedor compatible con los dos productos. Consulte la página Descargas de software de Oracle Fusion Middleware 11gR1.
Oracle Access Manager SDK 10g u 11g	Descargue Oracle Access Manager. El SDK es necesario para compilar y generar los módulos de autenticación de OpenSSO para la integración con Oracle Access Manager. Consulte la página Descargas de software de Oracle Fusion Middleware 11gR1.
OpenSSO C-SDK 2.2	(Opcional) OpenSSO C-SDK es necesario para crear un módulo de autenticación en Oracle Access Manager a fin de generar una sesión de OAM. Desde la perspectiva de OpenSSO, es posible que este no sea un ejemplo de uso habitual. Consulte "Where is the C SDK?" de *Sun OpenSSO Enterprise 8.0 C API Reference for Application and Web Policy Agent Developers*.

Descomprimir los bits de integración

El directorio `opensso/integrations/oracle` contiene el código fuente y las configuraciones para compilar y generar los módulos de autenticación personalizados y otros complementos. Consulte el Capítulo 3, "Integrating Oracle Access Manager" de *Sun OpenSSO Enterprise 8.0 Integration Guide* para conocer las opciones del caso práctico y obtener información relacionada. En la siguiente tabla se ofrece un resumen de los archivos del directorio `opensso/integrations/oracle` y una descripción de cada uno de ellos.

`README.html`	Se trata del archivo que está leyendo ahora.
`build.xml`	Un archivo de compilación Ant para generar un módulo de autenticación personalizado para Oracle Access Manager en OpenSSO.
`config`	Los archivos de configuración necesarios para crear un módulo de autenticación para Oracle Access Manager en OpenSSO. ■ `OblixAuthService.xml` El archivo del servicio de autenticación para el módulo de autenticación de Oracle Access Manager. ■ `OblixAuthModule.xml` Las devoluciones de llamadas del módulo de autenticación para Oracle Access Manager. Este archivo está vacío de forma predeterminada, pero debe estar presente por motivos de configuración. ■ `OblixAuth.properties` El archivo de propiedades que almacena las claves de internacionalización para la autenticación.
`lib`	El directorio está vacío de forma predeterminada. Este directorio `lib` debe contener las siguientes bibliotecas para compilar las bibliotecas de origen. ■ `jobaccess.jar` Copie este archivo desde Oracle Access Manager SDK. ■ `openfedlib.jar`, `amserver.jar` y `opensso-sharedlib.jar` Copie estos archivos desde opensso.war. ■ `servlet.jar o javaee.jar` Copie el directorio `lib` de GlassFish. Es válido cualquier archivo que presente clases de Java EE estándar como, por ejemplo, `javax.servlet.http.Cookie`.
`source`	El directorio que contiene los siguientes archivos: ■ `com/sun/identity/authentication/oblix/OblixAuthModule.java` ■ `com/sun/identity/authentication/oblix/OblixAuthModule.java`

- `com/sun/identity/authentication/oblix/OblixPrincipal.java`
- `com/sun/identity/saml2/plugins/OAMAdapter.java`

  Esta clase es un adaptador de complementos de SAML2 para los proveedores de servicios de SAML. Esta clase realiza la autenticación remota en Oracle Access Manager mediante el servicio de sesión de OpenSSO.

oamauth (opcional) — Este directorio contiene los archivos de origen del esquema de autenticación de Oblix para OpenSSO. Se trata de un módulo de autenticación basado en C que utiliza OpenSSO C-SDK para la validación.

- oam/solaris/authn_api.c

  Este archivo implementa el esquema de autenticación personalizado de Oblix para OpenSSO.

- oam/solaris/include/*.h

  Todos los archivos de encabezados necesarios para compilar el esquema de autenticación.

- oam/solaris/AMAgent.properties

  Archivo de configuración de ejemplo del agente de OpenSSO. Es necesario para que el esquema de autenticación valide la sesión de OpenSSO.

Creación de los archivos de origen de Oracle Access Manager en OpenSSO

Utilice la secuencia de comandos Ant para generar los archivos de origen. Debe instalarse y configurarse una secuencia de comandos Ant compatible en la variable PATH.

▼ Para crear los archivos de origen de Oracle Access Manager

1 Ejecute el comando siguiente:

```
cd $openssozipdir/integrations/oracle; ant -f build.xml
```

Este comando crea los archivos de origen y genera `fam_oam_integration.jar` en el directorio `$openssozipdir/integrations/oracle/dist`.

2 Incluya el módulo de autenticación en el archivo WAR OpenSSO.

a. Cree un directorio temporal y extraiga el archivo `opensso.war`. Ejemplo:

```
# mkdir /export/tmp
# cd /export/tmp
# jar -xvf opensso.war
```

A partir de ahora, `/export/tmp` se utiliza como área provisional del archivo WAR y se representa con un marco `$WAR_DIR`.

b. Copie `$openssozipdir/integrations/oracle/dist/fam_oam_integration.jar` en `$WAR_DIR/WEB-INF/lib`.

c. Copie `$openssozipdir/integrations/oracle/config/OblixAuth.properties` en `$WAR_DIR/WEB-INF/classes`.

d. Copie `$openssozipdir/integrations/oracle/config/OblixAuthModule.xml` en `$WAR_DIR/config/auth/default` y también en el directorio `$WAR_DIR/config/auth/default_en`.

e. Vuelva a compilar el archivo `opensso.war` mediante `jar cvf opensso.war` desde `$WAR_DIR`.

Ejemplo Pendiente de desarrollo

(Opcional) Generar un esquema de autenticación para OpenSSO en Oracle Access Manager

Nota: este no es un ejemplo de uso habitual. No tiene que generar este elemento a menos que sea necesario como, por ejemplo, si se utiliza un proveedor de servicios SAML2.

Para generar el esquema de autenticación de Oblix, debe personalizar el archivo `makefile`. Además, como se trata de un módulo de autenticación basado en C, depende del sistema operativo.

▼ Para generar un esquema de autenticación para OpenSSO en Oracle Access Manager

Antes de empezar Los archivos del esquema de autenticación se encuentran en el directorio `$openssozipdir/integrations/oracle/oamauth/solaris .`

1. **Descargue y configure la versión OpenSSO C-SDK 2.2.**

   El archivo `authn_api.c` contiene una referencia al archivo `AMAgent.propeties`. Modifique el archivo según corresponda.

2. **Personalice el archivo `makefile` para su entorno.**

   Por ejemplo, especifique la ubicación de compilación de `gcc`. Además, modifique `LDFLAGS` para que señale al directorio `lib` de OpenSSO C-SDK.

3. **Ejecute el comando `make`.**

   El comando `make` debería generar el archivo `authn_api.so`.

Configuración del inicio de sesión único mediante Oracle Access Manager y el servicio STS de Oracle OpenSSO

▼ Para configurar el inicio de sesión único mediante Oracle Access Manager y Oracle OpenSSO 8.0 Update 2

Antes de empezar: Sun Java System Web Server 7.x debe estar instalado y configurado. Consulte la página wiki de documentación de Sun Java System Web Server para obtener instrucciones de instalación de Web Server.

1. **Instale OpenSSO en Sun Java System Web Server 7.x.**

2. **Instale un agente de directivas de OpenSSO en un contenedor compatible y configúrelo para que funcione con OpenSSO.**

   Consulte *Sun OpenSSO Enterprise Policy Agent 3.0 User's Guide for J2EE Agents* o *Sun OpenSSO Enterprise Policy Agent 3.0 User's Guide for Web AgentsSun OpenSSO Enterprise Policy Agent 3.0 User's Guide for Web Agents* para obtener instrucciones de instalación.

3. **Instale y configure Oracle Access Manager.**

   Consulte *Oracle Access Manager Installation Guide 10g (10.1.4.3)*

4. **Instale y configure Oracle Access Manager SDK con Oracle Access Manager.**

   Consulte *Oracle Access Manager Installation Guide 10g (10.1.4.3)*

5. **Instale Oracle Webgate en el mismo contenedor web en el que se ha instalado el servidor de OpenSSO (Sun Web Server 7.x).**

   Configure OpenSSO para que proteja solo la sección `deployURI/UI/*` de la aplicación web de OpenSSO. Ejemplo: `/opensso/UI/.../*`

   Para obtener información sobre las directivas y recursos de Oracle Access Manager y otros detalles de configuración, consulte la Guía de administración de Oracle Access Manager. Desproteja todas las demás URL en OpenSSO Enteprise. Esta acción es para un escenario de integración de inicio de sesión. No obstante, evalúe las directivas en función de la integración completa y otras dependencias de implementación.

6. **Configure el módulo de autenticación en OpenSSO.**

   a. **Acceda a la consola de OpenSSO.**

      El navegador le redireccionará a Oracle Access Manager para realizar la autenticación. Una vez realizada con éxito la autenticación, OpenSSO presenta una página de inicio de sesión. Inicie una sesión mediante el nombre de usuario y la contraseña de administrador de OpenSSO.

   b. **Importe el archivo XML del servicio de módulo de autenticación de Oracle en la configuración de OpenSSO.**

      El servicio de módulo de autenticación se puede cargar desde la utilidad `ssoadm` de la línea de comandos, así como desde el navegador basado en ssoadm.jsp.

   c. **Acceda a `http://host:port/opensso/ssoadm.jsp`.**

   d. **Seleccione la opción de creación del servicio.**

   e. **Copie y pegue el archivo XML desde `$openssozipdir/integrations/oracle/config/OblixAuthService.xml` y haga clic en Enviar.**

      Se cargará el servicio de módulo de autenticación en la configuración de OpenSSO.

   f. **Registre este módulo en el servicio central de autenticación.**

      El servicio central contiene una lista de autenticadores. Seleccione la opción `register-auth-module` en `http://host:port/opensso/ssoadm.jsp` . Especifique `com.sun.identity.authentication.oblix.OblixAuthModule` como nombre de clase del módulo de autenticación.

g. Compruebe que el módulo de autenticación se haya registrado en el dominio predeterminado.

Acceda a OpenSSO mediante la dirección URL: http://host:port/opensso. En la consola de OpenSSO, haga clic en el dominio predeterminado y, a continuación, en la ficha Autenticación. Haga clic en Nuevo para crear un nuevo módulo de autenticación con el nombre OblixAuth.

h. En la ficha Autenticación, seleccione el módulo de autenticación OblixAuth.

Configure el directorio de SDK de Oblix. Habilite Comprobar solo encabezado de usuario remoto y especifique `OAM_REMOTE_USER` como nombre de encabezado remoto. Este parámetro permite su configuración en función de la implementación.

7 (Opcional) Habilite la opción Omitir perfil en el servicio central de autenticación de OpenSSO.

En la consola de OpenSSO, vaya a Configuración > Central > Atributos de dominio > Perfil de usuario. Seleccione Omitido y, a continuación, haga clic en Guardar.

Esta configuración impide que OpenSSO busque un perfil de usuario existente después de realizarse con éxito la autenticación. Sin embargo, si los depósitos de usuarios utilizado por OpenSSO y Oracle Access Manager son exactamente iguales, este paso no es necesario. Vaya a Consola de administración -> Configuración -> Central -> Atributos de dominio -> Perfil de usuario. Seleccione Omitido y, a continuación, haga clic en Guardar.

8 Edite la secuencia de comandos de inicio del servidor web para incluir las bibliotecas compartidas de Oracle Access Manager SDK.

Actualice `LD_LIBRARY_PATH` en la secuencia de comandos `startserv` para incluir las bibliotecas compartidas desde `$ACCESSDKDIR/oblix/lib`.

9 Reinicie la instancia de Sun Web Server que contiene tanto OpenSSO como Oracle Webgate.

10 Actualice el valor de la URL de inicio de sesión del agente web como `http://openssohost:openssoport/deployURI/UI/Login?module=OblixAuth .`

Para probar el inicio de sesión único

Acceda al recurso protegido desde la aplicación protegida por OpenSSO. El navegador debería redireccionarle a la página de inicio de sesión de Oracle Access Manager si aún no se ha autenticado. Tras iniciar con éxito una sesión, se crea una sesión de OpenSSO y, por último, se redirecciona a la URL de la aplicación protegida por el agente de directivas. En función de la directiva, se le permitirá o denegará el acceso a la aplicación protegida.

(Opcional) Instalación del esquema de autenticación de Oblix en Oracle Access Manager

Este procedimiento resulta útil cuando debe generarse la sesión de Oracle Access Manager al validar la sesión de OpenSSO. Consulte el Capítulo 3, "Integrating Oracle Access Manager" de *Sun OpenSSO Enterprise 8.0 Integration Guide* para obtener información sobre los casos prácticos pertinentes.

Los esquemas de autenticación de Oblix se muestran como módulos de autenticación basados en C y este esquema de autenticación utiliza la versión OpenSSO C-SDK 2.2 para validar la sesión de OpenSSO. El esquema de autenticación de OpenSSO en Oblix utiliza una configuración para los parámetros del servidor de OpenSSO en `AMAgent.properties`. Este archivo debe personalizarse antes de configurar el módulo de autenticación. En las instrucciones de compilación se especifica la ubicación de este archivo. El archivo compilado `authn_api.so` y las demás bibliotecas de C-SDK deben copiarse en el directorio `$OAM_INSTALL_DIR/access/oblix/lib` antes de configurar el esquema de autenticación. La *Guía de integración de Sun OpenSSO 8.0* incluye una captura de pantalla en la que se muestra cómo configurar el esquema de autenticación de Oracle que debe utilizarse únicamente como referencia. Para obtener más información, consulte la documentación más reciente de Oracle Access Manager.

Integración de OpenSSO 8.0 Update 2 con Oracle Access Manager

En esta sección se proporciona información sobre cómo implementar el inicio de sesión único mediante OpenSSO 1.4 Update 0 y las versiones 10.1.4.0.1. y 11g de Oracle Access Manager. Esta información complementa la información conceptual presente en el Capítulo 3, "Integrating Oracle Access Manager" de *Sun OpenSSO Enterprise 8.0 Integration Guide*. En este caso práctico, se proporciona un ejemplo de inicio de sesión único en aplicaciones protegidas por OpenSSO mediante una sesión de Oracle Access Manager. El módulo de autenticación de OpenSSO genera una sesión de OpenSSO basada en la sesión de Oracle Access Manager.