Guía de Seguridad de las TIC
CCN-STIC 1001

Procedimiento de Empleo Seguro
WSO2 Identity Server



Noviembre 2026

Edita:

CCN
c=ES, l=MADRID, o=CCN,
ou=SELLO ELECTRONICO,
2.5.4.97=VATES-S2800155J,
serialNumber=S2800155J,
cn=CCN
2026.11.04 14:07:01 +01:00

Fecha de Edición: Noviembre de 2026

LIMITACIÓN DE RESPONSABILIDAD

El presente documento se proporciona de acuerdo con los términos en él recogidos, rechazando expresamente cualquier tipo de garantía implícita que se pueda encontrar relacionada. En ningún caso, el Centro Criptológico Nacional puede ser considerado responsable del daño directo, indirecto, fortuito o extraordinario derivado de la utilización de la información y software que se indican incluso cuando se advierta de tal posibilidad.

AVISO LEGAL

Quedan rigurosamente prohibidas, sin la autorización escrita del Centro Criptológico Nacional, bajo las sanciones establecidas en las leyes, la reproducción parcial o total de este documento por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares del mismo mediante alquiler o préstamo públicos.

Esta publicación se distribuye bajo licencia CC BY-NC-ND 4.0 que permite compartir el material en cualquier medio o formato con la condición de reconocer adecuadamente la procedencia «Edita: Ministerio de Defensa. Secretaría General Técnica.» No se puede modificar. No se puede utilizar con fines comerciales.

ÍNDICE

1 INTRODUCCIÓN 4
2 OBJETO Y ALCANCE 4
3 ORGANIZACIÓN DEL DOCUMENTO 4
4 FASE PREVIA A LA INSTALACIÓN 5
4.1 ENTREGA SEGURA DEL PRODUCTO 5
4.1.1 ENTREGA SEGURA UTILIZANDO UPDATE TOOL 5
4.2 ENTORNO DE INSTALACIÓN SEGURO 6
4.3 REGISTRO Y LICENCIAS 7
4.4 CONSIDERACIONES PREVIAS 7
4.5 COMPONENTES DEL ENTORNO DE OPERACIÓN 7
5 FASE DE INSTALACIÓN 8
6 FASE DE CONFIGURACIÓN 10
6.1 MODO DE OPERACIÓN SEGURO 10
6.1.1 CONFIGURACIÓN JAVA_HOME 10
6.1.2 CONFIGURACIÓN DE LA BASE DE DATOS 10
6.1.3 CONFIGURACIÓN DE LOS ALMACENES DE USUARIOS 12
6.1.4 CONFIGURACIÓN Y ALMACENAMIENTO DE CLAVES SSL 17
6.1.5 CAMBIO DE CONTRASEÑA POR DEFECTO 20
6.1.6 CONFIGURACIÓN DE TLS 21
6.1.7 ACTUALIZACIÓN DEL SERVER NAME EN LA CABECERA DE RESPUESTAS HTTP 22
6.1.8 DESHABILITAR TRANSPORTE HTTP/HTTPS 23
6.1.9 CIPHER TOOL 23
6.1.10 RESTRINGIR LOS ENDPOINTS /CARBON Y /SERVICES 24
6.1.11 DESHABILITAR PUERTO 10711 (THIFT) 25
6.1.12 ACTUALIZACIÓN DE LA CONSOLA DE ADMINISTRACIÓN 25
6.2 AUTENTICACIÓN 26
6.3 ADMINISTRACIÓN DEL PRODUCTO 27
6.3.1 ADMINISTRACIÓN LOCAL Y REMOTA 27
6.3.2 CONFIGURACIÓN DE ADMINISTRADORES 27
6.4 CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS 29
6.5 GESTIÓN DE CERTIFICADOS 30
6.6 SERVIDORES DE AUTENTICACIÓN 30
6.7 SINCRONIZACIÓN 30
6.8 ACTUALIZACIONES 30
6.9 AUDITORÍA 30
7 REFERENCIAS 32
8 ABREVIATURAS 33

ÍNDICE DE TABLAS

Tabla 1: Prerrequisitos software 7
Tabla 2: Variables 10
Tabla 3: Configuración de las fuentes de datos 11
Tabla 4: Configuraciones básicas 11
Tabla 5: Almacenes de usuarios 13

Tabla 6: Mapeo de atributos 15
Tabla 7: Comandos a ejecutar 19
Tabla 8: Versiones y cipher suites de TLS soportadas 27

1 INTRODUCCIÓN

WSO2 Identity Server (WSO2 IS), versión 7.0.0, nivel de actualización 126 en adelante, se denominará TOE, es una solución potente y moderna de gestión de identidades y accesos para entornos locales o en la nube. Permite a las organizaciones ofrecer experiencias digitales excepcionales y fiables a todo tipo de usuarios.

El TOE puede gestionar las solicitudes de autenticación de los proveedores de servicios integrados y redirigirlas a los servicios de autenticación adecuados de forma remota o local. WSO2 Identity Server garantiza una fácil integración con diversas aplicaciones para facilitar el inicio de sesión único (SSO), el inicio de sesión social, la federación de identidades, la seguridad de las API, la autenticación fuerte, la gestión de cuentas y el cumplimiento de la normativa de privacidad.

2 OBJETO Y ALCANCE

Este Procedimiento de Empleo Seguro (PES) está desarrollado para el producto WSO2 Identity Server (WSO2 IS), versión 7.0.0, update 126.

Toda la información utilizada para la realización de este PES ha sido obtenida de la documentación de la evaluación Common Criteria: [ST] y [AGD].

El producto WSO2 IS (en adelante TOE) ha sido incluido en el catálogo CPSTIC, para conocer el listado, la categoría o nivel y la familia consulte el sitio web del CPSTIC [CPSTIC].

3 ORGANIZACIÓN DEL DOCUMENTO

a) Apartado 4. En este apartado se recogen aspectos y recomendaciones a considerar, antes de proceder a la instalación del producto.

b) Apartado 5. En este apartado se recogen recomendaciones a tener en cuenta durante la fase de instalación del producto.

c) Apartado 0. En este apartado se recogen las recomendaciones a tener en cuenta durante la fase de configuración del producto, para lograr una configuración segura.

4 FASE PREVIA A LA INSTALACIÓN

4.1 ENTREGA SEGURA DEL PRODUCTO

La siguiente sección describe todos los pasos necesarios para garantizar que los integradores de sistemas aseguren la aceptación del TOE entregado de acuerdo con los procedimientos de entrega del desarrollador.

La verificación satisfactoria tanto del procedimiento de entrega y aceptación del TOE como de la documentación de orientación de apoyo posterior se describe a lo largo de esta sección y debe ser coherente. Por lo tanto, antes de continuar, es importante tener en cuenta que, si alguno de los pasos descritos a lo largo de esta sección difiere o falla, el proceso se considerará poco fiable.

El TOE se puede descargar desde el repositorio oficial de GitHub [REPOSITORIO]. Utilizando el navegador que prefiera, introduzca el enlace en la barra de direcciones para acceder a la página de descarga de TOE en GitHub.

Se debe comprobar la validez del sitio web para verificar que se ha accedido al sitio web deseado, por lo que se debe validar el certificado digital antes de descargar el TOE.

Esta acción se lleva a cabo inspeccionando el certificado del sitio web. El certificado se considerará válido siempre que indique que ha sido emitido para «github.com» y no haya caducado.

Una vez verificada correctamente la validez del certificado, vaya a la sección «Assets», situada en la parte inferior de la página.

El siguiente paso es descargar el archivo «wso2is-7.0.0.zip», ya que este archivo zip contiene el TOE. Una vez descargado, es necesario validar el archivo zip para asegurarse de que se ha obtenido una copia auténtica del TOE antes de abrir el archivo.

El proceso para validar el valor hash es el siguiente:

a) Generar un valor hash SHA256 para el archivo «wso2is-7.0.0.zip».

b) Comparar el valor hash generado con el siguiente valor: *47df36faee3ff8f851d3d2064952968bbdd0b077f8b7367ddd872047e948c2c9*

4.1.1 ENTREGA SEGURA UTILIZANDO UPDATE TOOL

La WSO2 Update Tool proporciona un mecanismo seguro y fiable para aceptar y aplicar actualizaciones al TOE. La herramienta garantiza la autenticidad, integridad y aceptación controlada de las actualizaciones mediante las siguientes medidas:

a) Distribución autorizada: Las actualizaciones solo están disponibles para los clientes con suscripción a WSO2 (WSO2 ofrece una suscripción de prueba de 90 días para los usuarios con un correo electrónico empresarial válido. Durante el periodo de prueba, los usuarios pueden recibir actualizaciones y probar nuestro producto EULA). El acceso requiere credenciales de suscripción válidas, lo que impide descargas no autorizadas.

b) Puntos finales de confianza y canales seguros: Las actualizaciones se recuperan de los puntos finales del servidor del sistema de actualizaciones de WSO2 (api.updates.wso2.com, cdn.updates.wso2.com, cproduct-dist.wso2.com)

exclusivamente a través de conexiones HTTPS cifradas con TLS 1.2, lo que garantiza la confidencialidad e integridad de los artefactos de parcheo en tránsito. Estos puntos finales están protegidos con Cloudflare WAF.

Cuando se invoca el comando de actualización, la herramienta de actualización se comunica con el servicio de actualización de WSO2 para obtener detalles de las actualizaciones que deben descargarse y aplicarse, como los siguientes:

i. Artefactos que deben descargarse.

ii. Valores MD5sum, SHA1 y SHA256 de los artefactos.

Después de obtener la información sobre los archivos actualizados, la herramienta de actualización realiza una invocación autenticada al CDN para descargar las actualizaciones necesarias. La comunicación del cliente con el servicio de actualización WSO2 y el CDN está protegida mediante HTTPS.

c) Validación de integridad: Durante la aplicación de la actualización, la herramienta de actualización valida los artefactos con respecto a estos valores. Si se detecta una discrepancia, el proceso se aborta, lo que impide la aceptación de binarios manipulados.

   La herramienta de actualización valida los artefactos descargados de la siguiente manera:

   i. Para artefactos Java (`jar` y `war`): la herramienta de actualización comprueba los valores `md5` y `sha1`.

   ii. Para artefactos que no sean Java: la herramienta de actualización comprueba los valores `sha256`.

   La herramienta de actualización tiene un comando para comprobar el estado actual del producto, que mostrará el nivel de actualización y las revisiones aplicadas. Consulte la sección «Installing and running the udpate tool» de [AGD].

   Si le interesa la validación interna, vaya a su página de inicio de usuario. Allí encontrará una carpeta llamada «.wso2-updates». Dentro de esta carpeta, navegue hasta el directorio «updates», donde podrá ver las actualizaciones que ha ejecutado. En cada carpeta de actualización hay un archivo llamado «update.json». Este archivo contiene los hashes y los detalles de los archivos que se han modificado.

d) Aplicación controlada: Antes de aplicar las actualizaciones, la herramienta de actualización crea automáticamente una copia de seguridad del estado actual del TOE. Esto permite volver a la línea base evaluada anteriormente si la actualización falla en la verificación o introduce problemas.

e) Verificación por parte del usuario de la versión evaluada: El TOE proporciona un comando en la herramienta de actualización para mostrar el nivel de actualización actual y las revisiones aplicadas, lo que permite verificar que la versión implementada coincide con la línea base evaluada.

4.2 ENTORNO DE INSTALACIÓN SEGURO

La siguiente sección establece el entorno operativo adecuado para un uso seguro del TOE.

4.3 REGISTRO Y LICENCIAS

El TOE es un producto de código abierto, por lo que no se necesita realizar ningún registro. No obstante, el acuerdo de licencia puede revisarse en [LICENCIA].

4.4 CONSIDERACIONES PREVIAS

Antes de instalar el producto, es necesario tener instalado en el sistema el software previo adecuado.

A continuación, se detallan los requisitos del sistema operativo y los hardware compatibles:

a) CPU: 4 vCPU (arquitectura x86_64)

b) Memoria: 4 GB de RAM

c) Disco: 10 GB de espacio en disco, sin incluir el espacio asignado a los archivos de registro y las bases de datos.

d) Sistema operativo: Windows Server 2016 / Windows OS 11

NOTA: Las versiones del sistema operativo (SO) pueden ser posteriores a la versión indicada anteriormente.

4.5 COMPONENTES DEL ENTORNO DE OPERACIÓN

La siguiente tabla detalla los componentes software adicionales necesarios para el entorno operativo del TOE.

SOFWARE	DESCRIPCIÓN	VERSIÓN
Oracle Java JDK	Para garantizar la compatibilidad con el entorno, el TOE requiere que Oracle Java JDK se descargue e instale en el equipo.	21 o superior
MySQL	Se utiliza para sustituir la base de datos predeterminada, ya que la base de datos predeterminada utilizada por el TOE debe cambiarse para que la configuración del TOE se realice correctamente.	8.4 o superior
Apache Ant	Es un requisito para la herramienta de cifrado, por lo que es necesario instalar este software para que la herramienta de cifrado pueda realizar su función.	1.10.17 o superior

Tabla 1: Prerrequisitos software

5 FASE DE INSTALACIÓN

Las siguientes instrucciones deben usarse para instalar el producto sobre un equipo Windows. Siga los siguientes pasos:

a) Descargue WSO2 Identity Server 7.0.0 siguiendo las instrucciones de la sección 2.

b) Extraiga el archivo comprimido en un directorio dedicado para WSO2 Identity Server, al que en adelante nos referiremos como <IS_HOME>.

c) Configure la variable de entorno CARBON_HOME apuntando al directorio donde descargó WSO2 Identity Server. Los pasos siguientes indican cómo realizar esta tarea en un equipo con sistema operativo Windows 11:

   i. En Buscar, busque y seleccione: Sistema (Panel de control)

   ii. Haga clic en el enlace Configuración avanzada del sistema.

   iii. Haga clic en Variables de entorno. En la sección Variables del sistema, busque la variable de entorno CARBON_HOME y selecciónela. Haga clic en Editar. Nota: Si la variable de entorno CARBON_HOME no existe, haga clic en Nuevo.

d) En la ventana Editar variable del sistema (o Nueva variable del sistema), especifique el valor de la variable de entorno CARBON_HOME, el directorio donde descargó WSO Identity Server. Haga clic en Aceptar. Cierre todas las ventanas restantes haciendo clic en Aceptar.

La herramienta de actualización de WSO2 proporciona un mecanismo seguro para obtener y aplicar actualizaciones al servidor de identidad WSO2 (TOE). Siga los pasos que se indican a continuación para asegurarse de que el TOE se actualiza de forma segura a la línea base evaluada o al último nivel de parche.

a) Descargue el paquete del producto

   i. Visite la página Versiones anteriores del servidor de identidad WSO2.

   ii. Seleccione 7.0.0 en el menú desplegable.

   iii. Proporcione su dirección de correo electrónico corporativa, marque «Acepto la licencia de software de WSO2» y haga clic en el enlace «Descargar» situado junto a la opción Binario.

   iv. Compruebe su bandeja de entrada para ver si ha recibido un correo electrónico con un enlace de descarga de un solo uso y utilícelo para descargar el paquete de productos.

b) Creación de una cuenta de usuario WSO2

   i. Compruebe su bandeja de entrada de correo electrónico para ver si ha recibido un mensaje con el asunto «Detalles de la cuenta de <su nombre> para acceder a los servicios WSO2» de no-reply@wso2.com.

   ii. Haga clic en la URL de un solo uso y restablezca las credenciales.

   iii. Introduzca las nuevas credenciales, actualice cualquier otra información que necesite según sea necesario, rellene el captcha y haga clic en «guardar» para aplicar los cambios.

iv. Una vez que vea el mensaje «los cambios se han aplicado», puede cerrar la página.

c) Descarga la herramienta de actualización:

i. Opción 1: Descargar la herramienta de actualización directamente de [UPDATE].

1. Extraiga el fichero zip descargado.
2. Copie el contenido de la carpeta «Update» en el directorio «bin» de la carpeta «Identity Server» (<IS-HOME>).

ii. Opción 2: Descargar el paquete EULA y copiar la herramienta de actualización del paquete EULA al paquete OSS.

1. Descargue el paquete EULA del correo electrónico que ha recibido.
2. Extraiga el archivo zip descargado.
3. Copie el contenido de la carpeta Update en el directorio bin de la carpeta Identity Server (<IS-HOME>).
4. Ejecutar la herramienta de actualización para obtener las actualizaciones del producto.
5. Vaya a la ubicación <IS-HOME>/bin.
6. Ejecute el comando de actualización correspondiente en función de su sistema operativo:
    1. Para Windows:

```
./wso2update_windows.exe --level 126
```

7. Cuando se le solicite, introduzca las credenciales de su cuenta WSO2. Una vez autenticado, la herramienta se conectará al servicio de actualización de WSO2, descargará las últimas actualizaciones disponibles y las aplicará a la distribución del producto.
8. Reinicie el servidor para que las actualizaciones surtan efecto.

6 FASE DE CONFIGURACIÓN

6.1 MODO DE OPERACIÓN SEGURO

Para activar la configuración específica del producto para que este opere en el modo de cumplimiento *Common Criteria* se deben seguir las siguientes configuraciones.

6.1.1 CONFIGURACIÓN JAVA_HOME

Es necesario configurar la variable de entorno JAVA_HOME para que apunte al directorio donde está instalado el Java Development Kit (JDK) en el ordenador.

Nota: Normalmente, el JDK se instala en un directorio dentro de C:/Archivos de programa/Java, como C:/Archivos de programa/Java/jdk-11.0.14.

A continuación, debe configurar JAVA_HOME mediante las Propiedades del sistema, tal y como se describe a continuación. El proceso para configurar las Propiedades del sistema es el siguiente:

a) Paso 1: Haga clic con el botón derecho del ratón en el icono Mi PC del escritorio y seleccione Propiedades.

b) Paso 2: En la ventana Propiedades del sistema, haga clic en Avanzadas y, a continuación, en Variables de entorno.

c) Paso 3: Haga clic en Nuevo en Variables del sistema (para todos los usuarios) o en Variables de usuario (solo para el usuario que ha iniciado sesión actualmente).

d) Paso 4: Introduzca la siguiente información tal y como se muestra en la Tabla 2

CAMPO	DESCRIPCIÓN	VALOR
Nombre de la variable	Variable usada para definir JAVA_HOME	JAVA_HOME
Valor de la variable	Ruta de instalación del JDK	c:/Program Files/Java

Tabla 2: Variables

Nota: Una vez configurada la variable JAVA_HOME, se aplicará a todas las ventanas del símbolo del sistema que se abran posteriormente. Tenga en cuenta que, si hay alguna ventana del símbolo del sistema abierta, deberá cerrarla y volver a abrirla para que la variable JAVA_HOME surta efecto.

e) Paso 5: El último paso es verificar que la variable JAVA_HOME esté configurada correctamente. Para ello, abra una ventana de comandos (en el menú Inicio, haga clic en Ejecutar, escriba CMD y pulse Intro y ejecute el comando: set JAVA_HOME. El sistema mostrará la ruta de instalación del JDK si la configuración se ha realizado correctamente.

6.1.2 CONFIGURACIÓN DE LA BASE DE DATOS

De forma predeterminada, WSO2 Identity Server utiliza la base de datos H2 integrada como base de datos para almacenar los datos de gestión de usuarios y registro. Por lo

tanto, las fuentes de datos predeterminadas deben cambiarse para que apunten a una base de datos MySQL establecida.

Para cambiar la base de datos predeterminada y utilizar MySQL, es necesario seguir los siguientes pasos.

6.1.2.1 CONFIGURACIÓN DE LAS FUENTES DE DATOS

Una fuente de datos se utiliza para establecer la conexión con una base de datos. De forma predeterminada, las fuentes de datos WSO2_IDENTITY_DB y WSO2_SHARED_DB se utilizan para conectarse a la base de datos H2 predeterminada:

FUENTE DE DATOS	DESCRIPCION
WSO2_SHARED_DB	La Fuente de datos que almacena los datos del registro y la gestión de usuarios.
WSO2_IDENTITY_DB	La Fuente de datos específica del servidor de identidades que almacena los datos relacionados con la identidad

Tabla 3: Configuración de las fuentes de datos

Todas las configuraciones se realizan editando las configuraciones predeterminadas ubicadas en: `<IS-HOME>/repository/conf/deployment.toml`

> **Nota**: Al menos una de las fuentes de datos que se muestran en la tabla debe apuntar ahora a la base de datos MySQL; tenga en cuenta que también se pueden utilizar ambas fuentes.

6.1.2.2 CONFIGURACIÓN MÍNIMA PARA CAMBIAR LAS FUENTES DE DATOS POR DEFECTO

Consulte la Tabla 4 para llevar a cabo las configuraciones básicas.

ELEMENTO	DESCRIPCION
username and password	Nombre y contraseña del usuario de la base de datos
type	Tipo de base de datos
hostname	Nombre del host donde se aloja la base de datos
port	Puerto de la base de datos
name	Nombre de la base de datos

Tabla 4: Configuraciones básicas

Opción 1: Configuración mínima de muestra para WSO2_IDENTITY_DB:

```
[database.identity_db]
type = "mysql"
hostname = "localhost"
name = "regdb"
username = "regadmin"
password = "regadmin"
port = "3306"
```

Para inicializar esta configuración, vaya a <IS-HOME>/dbscripts. A continuación, ejecute los scripts de los siguientes archivos en la base de datos creada.

```
<IS-HOME>/dbscripts/consent/mysql.sql
```

Opción 2: Configuración mínima de ejemplo para WSO2_SHARED_DB:

```
[database.shared_db]
type = "mysql"
hostname = "localhost"
name = "regdb"
username = "regadmin"
password = "regadmin"
port = "3306"
```

Para inicializar esta configuración, vaya a <IS-HOME>/dbscripts. A continuación, ejecute los scripts del siguiente archivo en la base de datos creada.

```
<IS-HOME>/dbscripts/mysql.sql
```

Controlador JDBC de MySQL:

a) Descargue el controlador JDBC de MySQL y copie el archivo jar (mysql-connector-j-x.x.x.jar) en la carpeta <IS_HOME>/repository/components/lib.

b) Visite el sitio web oficial de descargas de MySQL [MySQL]. Compruebe que la conexión es fiable verificando el certificado, tal y como se indica en la sección 2.

6.1.3 CONFIGURACIÓN DE LOS ALMACENES DE USUARIOS

En esta etapa, se configurará el almacén de usuarios principal. Este es el almacén de usuarios principal que se comparte en el sistema. Tenga en cuenta que solo se debe configurar un almacén de usuarios como almacén de usuarios principal, que se configura en el archivo <IS_HOME>/repository/conf/deployment.toml.

De forma predeterminada, WSO2 Identity Server utiliza la base de datos H2 integrada como almacén de usuarios principal. Esto se cambiará para utilizar LDAP o AD como almacén de usuarios principal.

La consola de administración permite crear cualquier número de almacenes de usuarios secundarios, tenga en cuenta que estos también se pueden crear manualmente. Los almacenes de usuarios secundarios se guardan en un archivo xml con las configuraciones correspondientes.

Para configurar el almacén de usuarios principal, localice y abra el archivo deployment.toml ubicado en la siguiente dirección: <IS_HOME>/repository/conf/deployment.toml.

Hay dos pasos para configurar el almacén de usuarios principal:

a) Paso 1: Seleccione una de las siguientes opciones como administrador del almacén de usuarios:

ALMACEN	DESCRIPCION
LDAP	Se utiliza para que los almacenes de usuarios LDAP realicen operaciones de lectura y escritura. Esta es la configuración predeterminada del almacén de usuarios principal en el archivo deployment.toml para WSO2 Identity Server.
	org.wso2.carbon.user.core.ldap.UniqueIDReadWriteLDAPUserStoreManager
Active Directory	Se utiliza para configurar un servicio de dominio de Active Directory (AD DS) o un servicio de directorio ligero de Active Directory (AD LDS). Nota: Solo se puede utilizar para operaciones de lectura/escritura. Si necesita utilizar AD como solo lectura, deberá utilizar LDAP. org.wso2.carbon.user.core.ldap.UniqueIDActiveDirectoryUserStoreManager
	org.wso2.carbon.user.core.ldap.UniqueIDActiveDirectoryUserStoreManager

Tabla 5: Almacenes de usuarios

b) Paso 2: Configurar las propiedades del administrador del almacén de usuarios:

i. En el archivo deployment.toml, se configurará el administrador del almacén de usuarios añadiendo las propiedades pertinentes del tipo de administrador del almacén de usuarios seleccionado. Dependiendo de la opción seleccionada anteriormente (LDAP o AD), se seguirán los siguientes pasos.

ii. Opción 1: En caso de utilizar LDAP, se debe configurar un almacén de usuarios LDAP de lectura y escritura con las siguientes propiedades:

```
[user_store]
type = "read_write_ldap_unique_id"
base_dn = "ou=system"
connection_url = "ldap://localhost:10389"
connection_name = "uid=admin,ou=system"
connection_password = "admin"
```

Nota: Configure el tipo ldap read_write_ldap_unique_id. Los demás parámetros de la imagen son genéricos, cámbielos con la información válida de su almacén de usuarios.

Los atributos operativos whenCreated y whenChanged para los atributos creados y modificados también se incluirán en este archivo. Por lo tanto, se debe añadir lo siguiente al archivo deployment.toml:

```
[user_store]
timestamp_attributes = "modifyTimestamp,createTimestamp"
immutable_attributes = "modifyTimestamp,createTimestamp"
```

iii. Opción 2: En caso de utilizar AD como almacén de usuarios, introduzca esta configuración mínima en el archivo deployment.toml para cambiar el almacén de usuarios principal a un almacén de usuarios de Active Directory de lectura y escritura.

```
[user_store]
type = "active_directory_unique_id"
base_dn = "DC=<domain_component_1>,DC=<domain_component_2>
connection_url = "ldaps://<ldap_server_fqdn>:<ldaps_port>"
connection_name                                                    =
"cn=<bind_user>,OU=<service_accounts_ou>,DC=<domain_component_1>,DC=<
domain_component_2>"
```

```
connection_password = "$secret{user_store_connection_password}"
user_search_base                                                    =
"OU=<users_ou>,DC=<domain_component_1>,DC=<domain_component_2>"
user_entry_object_class = "user"
user_name_attribute = "cn"
user_id_Attribute = "objectGuid"
user_id_search_filter = "(&amp;(objectClass=user)(objectGuid=?))"
user_name_search_filter = "(&amp;(objectClass=user)(cn=?))"
user_name_list_filter = "(objectClass=user)"
group_search_base                                                   =
"OU=<groups_ou>,DC=<domain_component_1>,DC=<domain_component_2>"
group_entry_object_class = "group"
group_name_attribute = "cn"
group_id_attribute = "objectGuid"
group_name_search_filter = "(&amp;(objectClass=group)(cn=?))"
group_name_list_filter = "(objectClass=group)"
timestamp_attributes = "objectGuid,whenCreated,whenChanged"
immutable_attributes = "whenCreated,whenChanged"
```

Nota: Configure el tipo de directorio activo active_directory_unique_id. Los demás parámetros de la imagen son ejemplos de configuración; cámbielos con información válida de su almacén de usuarios.

Complete el parámetro connection_name con el nombre distinguido (DN) que identifica de forma única al usuario administrador del servidor. El DN se obtiene combinando el nombre común (CN) y los componentes de dominio (DC). Además, la unidad organizativa (OU) donde reside el objeto se puede añadir al DN.

Añada la siguiente línea debajo de la configuración existente de user_store para que el IS se alinee con los formatos de marca de tiempo de AD y evite posibles errores de análisis:

```
date_and_time_pattern          =          "[uuuuMMddHHmmss[,SSS][.SSS]X]
[uuuuMMddHHmmss[,SS][.SS]X][uuuuMMddHHmm[,S][.S]X][uuuuMMddHHmmss[,S]
[.S]X]"
```

Los atributos operativos whenCreated y whenChanged para los atributos creados y modificados también se incluirán en este archivo. Por lo tanto, se debe añadir lo siguiente al archivo deployment.toml:

```
[user_store]
timestamp_attributes = "whenCreated,shenChanged"
immutable_attributes = “objectGuid,whenCreated,whenChanged”
```

Además de lo anterior, asegúrese de que el certificado público del almacén de usuarios de Active Directory también se importe al almacén de confianza del cliente de WSO2 Identity Server. Para realizar esta acción, navegue hasta el directorio <IS_HOME>repository/resources/security y ejecute el siguiente comando para importar el certificado al almacén de confianza del cliente de WSO2 Identity Server.

```
keytool -import -alias <certalias> -file <certificate>.<cert_extension>
-keystore <client-truststore.jks> -storepass <password>
```

Nota: El usuario deberá indicar en <cert_extension> la extensión de certificado AD correspondiente, es decir, el formato pem. Tenga en cuenta que el formato del

certificado debe coincidir con la extensión del archivo, ya que a menudo se producen errores debido a una desalineación entre la extensión y el formato real del certificado.

La contraseña del almacén de confianza del cliente también debe cambiarse después de este paso. La contraseña predeterminada es «wso2carbon».

6.1.3.1 ACTUALIZACIÓN DE LOS CLAIM MAPPINGS

Es necesario actualizar las asignaciones de reclamaciones en <carbon_home>/repository/conf/claim-config.xml. Tenga en cuenta que esta acción se realiza antes de iniciar el servidor.

Las siguientes reclamaciones obligatorias deben asignarse a los atributos del almacén de usuarios:

ATRIBUTO LOCAL	ATRIBUTO MAPEADO
http://wso2.org/claims/location	streetAddress
http://wso2.org/claims/resourceType	unixHomeDirectory
http://wso2.org/claims/im	userWorkstations
http://wso2.org/claims/created	whenCreated
http://wso2.org/claims/userid	objectGuid
http://wso2.org/claims/modified	whenChanged
http://wso2.org/claims/username	cn
http://wso2.org/claims/groups	groups
http://wso2.org/claims/country	co

Tabla 6: Mapeo de atributos

Consulte [AGD] para ver cada atributo y configurar las asignaciones de atributos tal y como se explica en la sección «Gestionar atributos».

6.1.3.2 CONFIGURACIÓN DEL TENANT MANAGER Y ADMINISTRADOR

El siguiente paso es configurar el tenant manager para que utilice CommonHybridLDAPTenantManager. También es necesario configurar la propiedad RootPartition, ya que determina la raíz del árbol LDAP. Añada la configuración al archivo <IS_HOME>/repository/conf/deployment.toml como se muestra a continuación:

```
tenant_manager.ldap]
enabled=true
class                                                                    =
"org.wso2.carbon.user.core.tenant.CommonHybridLDAPTenantManager"
[tenant_manager.ldap.properties]
RootPartition = “dc=example,dc=org”
```

El usuario Admin (super_admin) es el superinquilino que podrá gestionar todos los demás usuarios, roles y permisos del sistema. Por lo tanto, el usuario que debe tener permisos de administrador debe almacenarse en el almacén de usuarios cuando se inicia el sistema por

primera vez. Dado que se puede escribir en el almacén de usuarios LDAP, tiene la opción de crear un nuevo usuario Admin en el almacén de usuarios cuando inicie el sistema por primera vez. También puede utilizar un ID de usuario que ya exista en el LDAP.

a) Opción 1: Si utiliza un usuario que ya está en el LDAP, busque un usuario válido que ya resida en el almacén de usuarios.

```
super_admin]
username = "AdminSOA"
admin_role = "admin"
create_admin_account = false
```

Nota: no es necesario actualizar el elemento de contraseña, ya que ya está configurado en el almacén de usuarios. El nombre de usuario «AdminSOA» es un ejemplo, utilice un nombre de usuario que ya exista en el LDAP.

b) Opción 2: si se va a crear un nuevo usuario administrador en el almacén de usuarios al iniciar el sistema, añada el usuario superinquilino al almacén de usuarios.

```
super_admin]
username = "AdminSOA"
admin_role = "admin"
create_admin_account = false
password = "admin"
```

Nota: el nombre de usuario «AdminSOA» y la contraseña «admin» son un ejemplo.

6.1.3.3 SECURIZAR COMUNICACIONES LDAP

Para garantizar una comunicación segura entre el TOE y el almacén de usuarios, LDAPS debe configurarse para imponer el uso de protocolos TLS y conjuntos de cifrado robustos, garantizando que solo se acepten los protocolos TLS 1.2 y TLS 1.3 y los conjuntos de cifrado aprobados por la ENISA.

Para garantizar que el servidor AD (LDAPS) solo funcione con cifrados seguros:

a) Paso 1: Abra Windows PowerShell en modo administrador.

b) Paso 2: Ejecute el siguiente comando para obtener la lista de cifrados habilitados:

```
Get-TlsCipherSuite | ForEach-Object { $_.Name}
```

c) Paso 3: Ejecute el siguiente script (en modo administrador de PowerShell) para deshabilitar los conjuntos de cifrado débiles. Puede cambiar la lista de cifrados si desea añadir o eliminar cifrados.

```
$weakSuites = @(
  "TLS_RSA_WITH_AES_256_GCM_SHA384",
  "TLS_RSA_WITH_AES_128_GCM_SHA256",
  "TLS_RSA_WITH_AES_256_CBC_SHA256",
  "TLS_RSA_WITH_AES_128_CBC_SHA256",
  "TLS_RSA_WITH_AES_256_CBC_SHA",
  "TLS_RSA_WITH_AES_128_CBC_SHA",
  "TLS_RSA_WITH_NULL_SHA256",
  "TLS_RSA_WITH_NULL_SHA",
  "TLS_RSA_WITH_3DES_EDE_CBC_SHA",
  "TLS_PSK_WITH_AES_256_GCM_SHA384",
```

```
  "TLS_PSK_WITH_AES_128_GCM_SHA256",
  "TLS_PSK_WITH_AES_256_CBC_SHA384",
  "TLS_PSK_WITH_AES_128_CBC_SHA256",
  "TLS_PSK_WITH_NULL_SHA384",
  "TLS_PSK_WITH_NULL_SHA256"
)
foreach ($suite in $weakSuites) {
  try {
    Disable-TlsCipherSuite -Name $suite
    Write-Output "Disabled: $suite"
  } catch {
    Write-Warning "Could not disable ${suite}: ${_}"
  }
}
```

d) Paso 4: Ejecute de nuevo el siguiente comando para verificar:

```
Get-TlsCipherSuite | ForEach-Object { $_.Name}
```

e) Paso 5: Reinicie el servidor para que los cambios surtan efecto.

Para garantizar que el cliente LDAP solo funcione con cifrados seguros, asegúrese de que el TOE se haya configurado de acuerdo con las instrucciones de refuerzo de TLS y conjuntos de cifrado.

6.1.4 CONFIGURACIÓN Y ALMACENAMIENTO DE CLAVES SSL

El siguiente paso es cambiar el almacén de claves predeterminado y crear nuevas claves SSL. En la siguiente sección se detallan los pasos a seguir para cambiar tanto el almacén de claves principal como el secundario.

6.1.4.1 CONFIGURACIÓN DE LA PRIMARY KEYSTORE

El almacén de claves principal se utiliza para la firma y el cifrado de datos confidenciales, que están expuestos externamente. El certificado de clave pública del almacén de claves principal debe tener el uso de clave de cifrado de datos para permitir el cifrado directo de datos sin procesar utilizando su clave pública.

> **Nota**: Este uso de clave ya está incluido en el certificado autofirmado incluido en el almacén de claves wso2carbon.jks predeterminado.

En este paso se utilizará la herramienta keytool que se incluye con la instalación de JDK:

a) Navegue hasta el directorio <IS_HOME>/repository/resources/security/ en un símbolo del sistema.

> **Nota**: Todos los almacenes de claves deben guardarse aquí.

b) Para crear el almacén de claves que incluye la clave privada, ejecute el siguiente comando:

```
keytool -genkey -alias <newcert> -keyalg RSA -keysize 3072 -keystore
<newkeystore>.jks -dname "CN=<testdomain.org>,
OU=<Home>,O=<Home>,L=<SL>,S=<WS>,C=<LK>" -storepass <mypassword> -
keypass <mypassword> -ext KeyUsage=dataEncipherment
```

> **Nota**: Los parámetros entre ángulos son ejemplos para mayor claridad de las directrices.

Este comando creará un almacén de claves con los siguientes detalles:

a) Nombre del almacén de claves: newkeystore.jks.

b) Alias del certificado público: newcert.

c) Contraseña del almacén de claves: mypassword.

d) Contraseña de la clave privada: mypassword.

La contraseña de la clave privada debe ser la misma que la contraseña del almacén de claves.

Abra el directorio <IS_HOME>/repository/resources/security/ y compruebe que se ha creado el nuevo archivo de almacén de claves. A continuación, cree una copia de seguridad de este archivo y muévalo a una ubicación segura.

Nota: Este paso es importante, ya que es el único lugar donde se encuentra su clave privada.

En este punto, se habrá creado el almacén de claves (archivo .jks) con una clave privada y un certificado de clave pública autofirmado.

6.1.4.2 CONFIGURACIÓN DE LA SECONDARY KEYSTORE

El almacén de claves secundario se utiliza para autenticar la comunicación a través de SSL/TLS.

Esto se realiza repitiendo el proceso que se muestra en la sección anterior y cambiando el valor de las opciones alias, keystore y CN. Este almacén de claves debe tener la misma contraseña que su clave privada. Añada el certificado firmado por la CA al almacén de claves.

a) Paso 1: Genere una solicitud de firma de certificado (CSR) para el almacén de claves (archivo.jks). Este archivo CSR puede ser certificado por una autoridad de certificación (CA), que es una entidad que emite certificados digitales. Estos certificados certifican la propiedad de una clave pública.

b) Paso 2: Ejecute el siguiente comando para generar el CSR.

Nota: Los parámetros entre ángulos son ejemplos para mayor claridad de las directrices.

```
Keytool -certreq -alias <secondarycertalias> -file <newcertreq>.csr -
keystore <secondarykeystore>.jks
```

c) Paso 3: En este punto, aparecerá un mensaje solicitando que se proporcione una contraseña para el almacén de claves. Una vez introducida la contraseña, el comando generará el archivo newcertreq.csr en el directorio <IS_HOME>/repository/resources/security/. Este es el CSR que debe enviar a una CA.

Nota: En este punto, es obligatorio proporcionar este archivo CSR a una CA.

Después de aceptar la solicitud, se proporciona un certificado firmado junto con un certificado raíz y varios certificados intermedios (dependiendo de la CA) como un paquete (archivo .zip).

d) Paso 4: Para añadir el certificado raíz de la CA y los dos certificados intermedios (relacionados), ejecute los siguientes comandos.

Nota: Los parámetros entre ángulos son ejemplos para mayor claridad de las directrices.

COMANDO	DESCRIPCION
Command 1	keytool -import -v -trustcacerts -alias <ExternalCARoot> -file <AddTrustExternalCARoot>.crt -keystore <secondarykeystore>.jks -storepass <mypassword>
Command 2	keytool -import -v -trustcacerts -alias <TrustCA> -file <COMODORSAAddTrustCA>.crt -keystore <secondarykeystore>.jks -storepass <mypassword>
Command 3	keytool -import -v -trustcacerts -alias <SecureServerCA> -file <COMODORSADomainValidationSecureServerCA>.crt -keystore <secondarykeystore>.jks -storepass <mypassword>

Tabla 7: Comandos a ejecutar

e) Paso 5: Para añadir el certificado SSL firmado por la CA al almacén de claves, ejecute el siguiente comando.

Nota: Los parámetros entre ángulos son ejemplos para mayor claridad de las directrices.

```
keytool -import -v -alias <secondarycertalias> -file <test_sampleapp_org>.crt -keystore <secondarykeystore>.jks -keypass <mypassword> -storepass <mypassword>
```

En este punto, se configurará el almacén de claves Java, que incluye un certificado de clave pública firmado por la CA que se puede utilizar para SSL en un entorno de producción.

f) Paso 6: Es posible que sea necesario añadir el mismo certificado de clave pública firmado por la CA al archivo client-truststore.jks. Esto proporcionará seguridad y confianza para la comunicación backend/inter-sistemas de WSO2 Identity Server a través de SSL.

Siga los pasos que se indican a continuación para importar el mismo certificado de clave pública firmado por la CA (que obtuvo en el paso anterior) al almacén de confianza predeterminado de WSO2 Identity Server (client-truststore.jks):

i. Obtenga una copia del archivo client-truststore.jks del directorio <IS_HOME>/repository/resources/security/.

ii. Para exportar la clave pública de su archivo .jks, ejecute el siguiente comando.

Nota: Los parámetros entre ángulos son ejemplos para mayor claridad de las directrices

iii. Importe la clave pública que extrajo en el paso anterior al archivo client-truststore.jks utilizando el siguiente comando.

Nota: Los parámetros entre ángulos son ejemplos para mayor claridad de las directrices

```
keytool -import -alias <certalias> -file <public key name>.pem -keystore <client-truststore>.jks -storepass <password>
```

Nota: Ahora debe cambiar la contraseña de client-truststore; la contraseña predeterminada es «wso2carbon».

6.1.4.3 REMPLAZO DE LA KEYSTORE POR DEFECTO

En este punto, se almacenará un certificado SSL en un almacén de claves Java y se añadirá una clave pública al archivo client-truststore.jks. Tenga en cuenta que ambos archivos deben estar ubicados en el directorio <IS_HOME>/repository/resources/security/.

El almacén de claves predeterminado wso2carbon.jks de la instancia de WSO2 Identity Server se sustituirá ahora por el almacén de claves recién creado. Para ello, actualice los archivos de configuración pertinentes como se muestra a continuación:

a) El almacén de claves principal se configura especificando las opciones de configuración en el archivo deployment.toml ubicado en el directorio <IS_HOME>/repository/conf/.

b) Para configurar el nuevo almacén de claves, añada el siguiente bloque de configuración a la etiqueta keystore.internal del archivo deployment.toml en el directorio <IS_HOME>/repository/conf.

Nota: Estas contraseñas se cambiarán por una contraseña actualizada.

```
[keystore.internal]
file_name = "internal.jks"
type = "JKS"
password = "wso2carbon"
alias = "wso2carbon"
key_password = "wso2carbon"
```

c) El último paso es configurar el almacén de claves secundario actualizando el mismo archivo y añadiendo la sección [transport.https.sslHostConfig.certificate.properties].

Nota: Los parámetros de la imagen son referenciales a los datos reales. Establezca los datos correspondientes.

```
[transport.https.sslHostConfig.certificate.properties]
certificateKeystoreFile =
"${carbon.home}/repository/resources/security/$ref{keystore.tls.file_
name}"
certificateKeystorePassword = "$ref{keystore.tls.password}"
certificateKeystoreType = "$ref{keystore.tls.type}"
certificateKeyAlias = "$ref{keystore.tls.alias}"
certificateKeyPassword = "$ref{keystore.tls.key_password}"
```

6.1.5 CAMBIO DE CONTRASEÑA POR DEFECTO

Para proteger el almacén de confianza, ahora es necesario cambiar la contraseña predeterminada de client-truststore.jks. Para ello, abra un terminal, vaya a <IS_HOME> y ejecute el siguiente comando.

Nota: También es necesario cambiar la contraseña del almacén de confianza en el archivo deployment.toml.

```
keytool -storepasswd -keystore repository/resources/security/client-
truststore.jks
```

Se le solicitarán la contraseña predeterminada y la nueva contraseña. La contraseña predeterminada es «wso2carbon».

Además, el TOE utiliza una contraseña de aplicación interna (internal_app_password) para la autenticación entre componentes. Esta contraseña debe sustituirse por un valor seguro y complejo.

a) Paso 1: En «<IS_HOME>/repository/conf/deployment.toml», añada un secreto que se utilizará más adelante como «app_password». Se recomienda utilizar una contraseña segura y fuerte.

```
[secrets]
internal_app_password= "[<new-passwords>]"
```

b) Paso 2: Vaya a <IS_HOME>/bin/ y ejecute:

```
ciphertool.bat -Dconfigure
```

c) Paso 3: En «<IS_HOME>/repository/conf/deployment.toml», añada la propiedad «app_password» que apunta al secreto creado en el paso 1.

```
[identity.auth_framework.endpoint]
app_password= "$secret{internal_app_password}"
```

d) Paso 4: En «<IS_HOME>/repository/conf/deployment.toml», establezca la propiedad «hash» y proporcione el hash SHA-256 de «app_password» como su valor. Recuerde que deberá generar el hash SHA-256 de la contraseña que se utilizó en el paso 1.

```
[account_recovery.endpoint.auth]
hash= "<SHA-256 hash of the newly added app_password property value>"
```

6.1.6 CONFIGURACIÓN DE TLS

En esta sección, se mejorará la seguridad TLS desactivando los cifrados débiles y eliminando las versiones antiguas de TLS.

Para configurar los protocolos TLS y los conjuntos de cifrado en WSO2 Identity Server, actualice el archivo de configuración <IS_HOME>/repository/conf/deployment.toml como se describe a continuación.

a) Paso 1: Vaya al archivo deployment.toml en el directorio <IS_HOME>/repository/conf.

b) Paso 2: Haga una copia de seguridad del archivo deployment.toml y detenga el servidor WSO2 IS.

c) Paso 3: Añada la siguiente configuración al archivo deployment.toml:

```
[transport.https.sslHostConfig.properties]
protocols= "TLSv1.2+TLSv1.3"
ciphers=
"TLS_AES_128_GCM_SHA256,TLS_AES_256_GCM_SHA384,TLS_ECDHE_RSA_WITH_AES
_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256"
```

d) Paso 4: Cuando se inicie el servidor, se aplicarán los ajustes.

A nivel de la máquina virtual Java (JVM), el TOE aplica restricciones a los protocolos TLS, los conjuntos de cifrado y los algoritmos de validación de certificados para garantizar que los algoritmos débiles u obsoletos se desactiven globalmente para todos los canales de comunicación seguros.

Actualice la siguiente configuración en la ubicación {JavaHome}/conf/security/java.security:

```
jdk.tls.disabledAlgorithms=SSLv3, TLSv1, TLSv1.1, \
    TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_256_CBC_SHA, \
    TLS_RSA_WITH_AES_128_CBC_SHA256,  TLS_RSA_WITH_AES_256_CBC_SHA256,
\
    TLS_RSA_WITH_AES_128_GCM_SHA256,  TLS_RSA_WITH_AES_256_GCM_SHA384,
\
    TLS_DHE_RSA_WITH_AES_128_CBC_SHA,
TLS_DHE_RSA_WITH_AES_256_CBC_SHA, \
    TLS_DHE_RSA_WITH_AES_128_CBC_SHA256,
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256, \
    TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384, \
    TLS_DHE_DSS_WITH_AES_128_CBC_SHA,
TLS_DHE_DSS_WITH_AES_256_CBC_SHA, \
    TLS_DHE_DSS_WITH_AES_128_CBC_SHA256,
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256, \
    TLS_DHE_DSS_WITH_AES_128_GCM_SHA256,
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384, \
    TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, \
    TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384, \
    TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, \
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, \
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384, \
    TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA,
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA, \
    TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256,
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384, \
    TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384, \
    TLS_ECDH_RSA_WITH_AES_128_CBC_SHA,
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA, \
    TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256,
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384, \
    TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256,
TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384, \
    TLS_CHACHA20_POLY1305_SHA256,
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256, \
    TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256, \
    TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256, \
    TLS_EMPTY_RENEGOTIATION_INFO_SCSV, \
    RC4, DES, 3DES_EDE_CBC, MD5withRSA, anon, NULL, \
    DH keySize < 2048, EC keySize < 224
```

6.1.7 ACTUALIZACIÓN DEL SERVER NAME EN LA CABECERA DE RESPUESTAS HTTP

De forma predeterminada, WSO2 Identity Server pasa «WSO2 WSO2 IS server» como valor del servidor en los encabezados HTTP al enviar respuestas HTTP. Esto significa que la

información sobre la pila de WSO2 Identity Server quedará expuesta a través de las respuestas HTTP.

Esto se debe cambiar configurando el nombre del servidor en el archivo deployment.toml mediante los siguientes pasos:

a) Abra el archivo <IS_HOME>/repository/conf/deployment.toml.

b) Añada un nuevo nombre de servidor añadiendo las siguientes propiedades en la configuración del conector https Tomcat:

```
jdk.certpath.disabledAlgorithms=MD2, MD5, SHA1, \
    RSA keySize < 2048, DSA keySize < 2048, EC keySize < 224, \
    include jdk.disabled.namedCurves

[transport.https.properties]
server="WSO2 WSO2 IS server"
```

6.1.8 DESHABILITAR TRANSPORTE HTTP/HTTPS

De forma predeterminada, los conectores http y https están habilitados. Para deshabilitar el conector http y evitar el acceso sin cifrar, añada la configuración correspondiente a <IS_HOME>/repository/conf/deployment.toml como se muestra a continuación:

```
[transport.http]
enabled = false
```

Nota: Esta sección debe aparecer antes de cualquier subsección de transporte que figure en la configuración anterior.

6.1.9 CIPHER TOOL

En esta etapa, todas las contraseñas configuradas previamente en los archivos de configuración se protegerán mediante su cifrado con Cipher Tool.

Las contraseñas que se cifrarán son las siguientes:

a) super_admin_password

b) user_store connection_password

c) database.identity_db_password

d) database.shared_db_password

e) keystore.primary_password

f) certificateKeystorePassword

g) certificateKeyPassword

h) truststore_password

i) password

j) key_password

Nota: Estos son ejemplos utilizados para aclarar las directrices. Cifre las contraseñas correspondientes.

Para cifrar contraseñas en WSO2 Identity Server, realice las siguientes acciones:

a) Paso 1: Añada las siguientes configuraciones [secrets] al final del archivo deployment.toml en el directorio <IS_HOME>/repository/conf/.

A continuación, asigne un alias al tipo de contraseña seguido de la contraseña real. El siguiente ejemplo enumera las contraseñas más comunes en los archivos de configuración.

Nota: Las contraseñas entre corchetes deben ser las contraseñas reales.

```
[secrets]
admin_password = "[password_1]"
keystore_password = "[password_2]"
key_password = "[password_3]"
truststrore_password = "[password_4]"
"log4j.appender.LOGEVENT.password" = "[password_5]"
```

b) Paso 2: A continuación, abra un terminal y navegue hasta el directorio <IS_HOME>/bin/. Habilite la herramienta Cipher para el producto ejecutando el comando -Dconfigure con el script de la herramienta cipher:

```
ciphertool.bat -Dconfigure
```

c) Paso 3: Vuelva al archivo deployment.toml y compruebe que las contraseñas de los alias están cifradas; el archivo debería tener un aspecto similar al del siguiente ejemplo:

```
[secrets]
admin_password =
"GeNld2aZkydnIZGtkZYOnXlVzl8WBtZzAQ8kIoR5c7aHmyUkWTag7w4…"
keystore_password =
"brClL1SOHdezXTvBz1/76b/DnHQgxjNGtzhaBr3DnhHw32NWY484…"
key_password =
"CFAaISaI19dHLApEM3usNSDXXdhdicHbVncrVwuLDJp6Rhp8B3Qy3PnBh…"
truststrore_password =
"DKnecEw+mJ8JhTUrqxpTZxwXrOdtcoAl2hD3LHtH+yJXNogum…"
"log4j.appender.LOGEVENT.password" =
"kydnIZGtkZYOnXlVzl8WBtZzAQ8kIoR5c7a…"
```

d) Paso 4: Reemplace las contraseñas del archivo deployment.toml con la referencia a los secretos creados anteriormente. A continuación, se muestra un ejemplo de la referencia para la contraseña de administrador:

```
[super_admin]
username = "adminSOA"
admin_role = "admin"
create_admin_account = true
password = "$secret{admin_password}"
```

6.1.10 RESTRINGIR LOS ENDPOINTS /CARBON Y /SERVICES

Para restringir el acceso a los puntos finales /carbon y /services desde el propio producto:

a) Paso 1: Añada las siguientes configuraciones al archivo «<IS_HOME>/repository/resources/conf/templates/repository/conf/tomcat/catalina-server.xml.j2».

Esta configuración debe añadirse entre los segmentos de código.

```
<Host   name="localhost"   unpackWARs="true"   deployOnStartup="false"
autoDeploy="false"appBase="${carbon.home}/repository/deployment/serve
r/webapps/">
{% if catalinaValves.remoteIpValve.enable is sameas true %}

<Context      path="/services"      docBase=""      reloadable="false"
antiResourceLocking="true" privileged="true">
    <!-- Restricting /services endpoint-->
    <Valve className="org.apache.catalina.valves.RemoteAddrValve"
           deny=".*" />
    <Valve className="org.apache.catalina.valves.ErrorReportValve"
           showServerInfo="false"
           showReport="false"
           errorCode.403="404" />
</Context>

<Context      path="/carbon"      docBase=""      reloadable="false"
antiResourceLocking="true" privileged="true">
    <!-- Restricting /carbon endpoint-->
    <Valve className="org.apache.catalina.valves.RemoteAddrValve"
           deny=".*" />
    <Valve className="org.apache.catalina.valves.ErrorReportValve"
           showServerInfo="false"
           showReport="false"
           errorCode.403="404" />
</Context>
```

Nota: Asegúrese de que las sangrías se mantengan correctamente. Consulte la Figura 9 a continuación (la nueva configuración aparece resaltada).

b) Paso 2: Guarde la configuración y reinicie el servidor.

A partir de este momento, cualquier intento de acceder a los puntos finales /carbon o /services devolverá una respuesta 403 Not Found.

6.1.11 DESHABILITAR PUERTO 10711 (THIFT)

En «<IS_HOME>/repository/conf/identity/thrift-authentication.xml», comente el puerto thrift de <Port>10711</Port> a <!--<Port>10711</Port>-->.

Reinicie el servidor Identity.

6.1.12 ACTUALIZACIÓN DE LA CONSOLA DE ADMINISTRACIÓN

Para mostrar el nivel de actualización actual de TOE en la consola de administración, añada la siguiente configuración a «<IS_HOME>/bin/wso2server.bat» justo antes de que comience el bloque de código «:runJava».

```
rem ----- Update the update.config.js with the update level from
updates/config.json -----

rem Update startup-config.js with update level from updates/config.json
if exist "%CARBON_HOME%\updates\config.json" if exist ^
"%CARBON_HOME%\repository\deployment\server\webapps\console\update.co
nfig.js" (

    rem Extract update level value using findstr and for loop
```

```
    for /f "tokens=2 delims=:" %%a in ('findstr /c:"update-level" ^
    "%CARBON_HOME%\updates\config.json"') do (

        for /f "tokens=1 delims=," %%b in ("%%a") do (
            set "UPDATE_LEVEL=%%b"
            set "UPDATE_LEVEL=!UPDATE_LEVEL: =!"
            set "UPDATE_LEVEL=!UPDATE_LEVEL:"=!"
        )
    )

    if defined UPDATE_LEVEL (
        set
"STARTUP_CONFIG_FILE=%CARBON_HOME%\repository\deployment\server\webap
ps\console\update.config.js"
        set "TEMP_FILE=%CARBON_HOME%\tmp\update_config_temp.js"

        rem Create temp directory if it doesn't exist
        if not exist "%CARBON_HOME%\tmp" mkdir "%CARBON_HOME%\tmp"

        rem Process the file line by line
        for /f "usebackq delims=" %%c in ('!STARTUP_CONFIG_FILE!') do
(
            set "LINE=%%c"
            echo !LINE! | findstr /c:"updateLevel:" >nul

            if !errorlevel! equ 0 (
                rem Replace the updateLevel value
                for /f "tokens=1,2* delims=:" %%d in ('!LINE!') do (
                    echo %%d: "!UPDATE_LEVEL!" >> "!TEMP_FILE!"
                )
            ) else (
                echo !LINE!>> "!TEMP_FILE!"
            )
        )

        rem Replace the original file with the temp file
        move /Y "!TEMP_FILE!" "!STARTUP_CONFIG_FILE!" >nul
        echo Updated Console update.config.js with update level:
        !UPDATE_LEVEL!
    )
)
```

6.2 AUTENTICACIÓN

El TOE utiliza los siguientes mecanismos de autenticación:

a) Autenticación de usuarios: el TOE proporciona LDPA y Active Directory para la autenticación de usuarios mediante atributos.

b) Autenticación entre componentes del producto: el producto utiliza HTTPS junto a TLS1.2 y TLS1.3 con las siguientes ciphersuites para proporcionar autenticación entre componentes.

TLS1.2	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
TLS1.3	TLS_AES_128_GCM_SHA256
	TLS_AES_256_GCM_SHA384

Tabla 8: Versiones y cipher suites de TLS soportadas

6.3 ADMINISTRACIÓN DEL PRODUCTO

6.3.1 ADMINISTRACIÓN LOCAL Y REMOTA

El TOE proporciona administración remota mediante LDAP y AD para sus usuarios.

El acceso remoto se realiza utilizando protocolos seguros como HTTPS y TLS con versiones 1.2 y 1.3 (ya mencionados en este documento).

Aunque por defecto el TOE proporciona HTTP, el procedimiento de empleo seguro obliga a la desactivación de este servicio como indica la sección 6.18 y obliga a la utilización de HTTPS y ciphersuites seguras de acuerdo con la guía CCN-STIC-807 como indica la sección 6.1.6 de este mismo documento.

6.3.2 CONFIGURACIÓN DE ADMINISTRADORES

Para el correcto uso del TOE la siguiente configuración debe realizarse:

a) Configuración de credenciales del producto. Estas deberán contener las siguientes características:
   a. Para credenciales basadas en contraseñas:
      i. Mínimo 15 caracteres
      ii. Conjunto formado por mínimo una letra minúscula [a-z], una letra mayúscula [A-Z], un números[0-9] y un caracteres especiales [“!”, “@”, "#", "$", "%", "^", "&", "*", “- ”, “_”]
      iii. El administrador deberá indicar el número de contraseñas almacenadas en el histórico para no reutilizar las mismas, en este caso 5.
   b. Para credenciales no basadas en contraseñas:
      i. La probabilidad de que un atacante pueda obtener un secreto durante la vigencia del secreto es inferior a 2^{-20}.

b) Roles de usuarios de administración. El TOE diferencia dos roles predeterminados para la administración del producto. Estos son los siguientes:
   a. Admin. Este rol se encuentra configurado por defecto con la instalación del TOE y contiene los siguientes permisos:
      i. Gestión de datos de autenticación tanto para usuarios interactivos como para entidades de TI autorizadas (si son gestionadas por el TSF)

ii. Definición de datos de identidad y credenciales que pueden asociarse a los usuarios (activar, suspender, revocar credenciales, etc.)

iii. Gestión del estado de las credenciales.

iv. Inscripción de usuarios.

v. Configuración de las circunstancias en las que se realiza la transmisión de datos de identidad y credenciales (y atributos de objetos, si procede).

vi. Configuración de eventos auditables.

vii. Configuración de la ubicación de almacenamiento de auditorías externas.

viii. Gestión del umbral de intentos de autenticación fallidos.

ix. Gestión de las medidas que se deben tomar en caso de fallo de autenticación.

x. Gestión de la métrica utilizada para verificar los secretos.

xi. Definición de los atributos de seguridad predeterminados del sujeto, modificación de los atributos de seguridad del sujeto.

xii. Gestión de los conjuntos de usuarios que pueden interactuar con las funciones de seguridad.

xiii. Gestión de los usuarios que pertenecen a una función concreta.

xiv. Configuración del periodo de inactividad para la terminación de la sesión.

xv. Mantenimiento del banner.

b. Administrator. Este rol se encuentra configurado por defecto con la instalación del TOE y contiene los siguientes permisos:

i. El conjunto de usuarios con acceso a las funciones de seguridad reservadas al rol de administrador es fijo (es decir, al usuario individual asociado inicialmente al rol de administrador durante la instalación de TOE).

ii. Los atributos de seguridad del rol de administrador están bajo el control exclusivo de dicho rol (no son accesibles para los administradores).

iii. La gestión del estado de las credenciales del rol de administrador está bajo el control exclusivo de dicho rol (no es accesible para los administradores).

iv. Los datos de identidad y credenciales para el rol de administrador se definen durante la instalación del TOE y posteriormente solo pueden ser modificados por el rol de administrador (no son accesibles para los administradores).

v. El rol de administrador no está autorizado a gestionar los datos de autenticación del rol de administrador.

vi. Gestión de datos de autenticación tanto para usuarios interactivos como para entidades de TI autorizadas (si son gestionadas por el TSF)

vii. Definición de datos de identidad y credenciales que pueden asociarse a los usuarios (activar, suspender, revocar credenciales, etc.)

viii. Gestión del estado de las credenciales.

ix. Inscripción de usuarios.

x. Configuración de las circunstancias en las que se realiza la transmisión de datos de identidad y credenciales (y atributos de objetos, si procede).

xi. Configuración de eventos auditables.

xii. Configuración de la ubicación de almacenamiento de auditorías externas.

xiii. Gestión del umbral de intentos de autenticación fallidos.

xiv. Gestión de las medidas que se deben tomar en caso de fallo de autenticación.

xv. Gestión de la métrica utilizada para verificar los secretos.

xvi. Definición de los atributos de seguridad predeterminados del sujeto, modificación de los atributos de seguridad del sujeto.

xvii. Gestión de los conjuntos de usuarios que pueden interactuar con las funciones de seguridad.

xviii. Gestión de los usuarios que pertenecen a una función concreta.

xix. Configuración del periodo de inactividad para la terminación de la sesión.

xx. Mantenimiento del banner.

xxi. Parámetros de sesión. El administrador deberá configurar:

1. Número máximo de intentos fallidos para iniciar sesión por los usuarios con un valor mínimo de 5.
2. El tiempo de bloqueo de sesión tras el número máximo de fallos de intentos de sesión con un valor mínimo de 5 minutos.
3. Tiempo de inactividad de sesión para usuarios con un valor mínimo de 60 segundos.
4. Banner de inicio de sesión. Se debe crear un banner de inicio de sesión a través de cualquier rol de administración.

6.4 CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS

Se deberá inhabilitar los siguientes puertos y servicios:

a) Puerto 10711, revisar sección 6.1.11

b) Restringir los endpoint /carbon y /services, revisar sección 6.1.10

c) Deshabilitar el transporte HTTP, revisar sección 6.1.8

d) Actualización de las HTTP Headers, revisar sección 6.1.7

e) Deshabilitar las ciphersuites no seguras de acuerdo con la guía CCN-STIC-807, revisar sección 6.1.6.

6.5 GESTIÓN DE CERTIFICADOS

La gestión de certificados se debe realizar de acuerdo con la sección 6.1.4.

Los algoritmos y funciones criptográficas que utiliza el producto cumplen con las especificaciones de la guía CCN-STIC-807 de acuerdo con lo establecido en este documento.

6.6 SERVIDORES DE AUTENTICACIÓN

Para la autenticación el producto utiliza el servidor externo LDAP, revisar la sección 6.1.3.

6.7 SINCRONIZACIÓN

El producto deberá utilizar como fuente de tiempo confiable la proporcionada por el entorno operacional, en este caso el sistema operativo Windows 11 donde el TOE es instalado.

6.8 ACTUALIZACIONES

Las actualizaciones del TOE se deberán realizar de acuerdo con la herramienta que incorpora WSO2 Update Tool, revisar las secciones 4.1.1 y 5 de este documento.

6.9 AUDITORÍA

Los eventos que registra el producto se describen en la siguiente lista.

a) Todo uso del mecanismo de autenticación.

b) Creación o modificación de datos de identidad y credenciales: los atributos modificados.

c) Inscripción o modificación del sujeto: el sujeto creado o modificado; los

d) atributos modificados (si procede).

e) Todos los intentos de transmitir información: el destino al que se

f) intentó realizar la transmisión.

g) Establecimiento y establecimiento de comunicaciones con el servidor de auditoría:

h) Identificación del servidor de auditoría

i) Todas las modificaciones del comportamiento de la función TSF

j) Uso de las funciones de gestión: función de gestión realizada

k) Todo uso de las funciones del canal de confianza: identidad del iniciador y del destino del canal de confianza

l) Todos los intentos de uso de las funciones de la ruta de confianza: identificación del usuario asociado a todas las funciones de la ruta de confianza, si está disponible

Para cada uno de estos registros el TOE deberá proporcionar la siguiente información:

a) Fecha y hora del evento

b) Tipo de evento

c) Identidad del sujeto (si procede)

d) Resultado (éxito o fracaso) del evento.

Los registros de auditoría serán almacenados localmente en el producto. El producto ofrece la posibilidad de almacenarlos de forma externa utilizando un canal seguro.

De forma predeterminada, los registros se almacenan localmente, configurados en el archivo log4j2.properties.

7 REFERENCIAS

ST	WSO2 Identity Server-ST-v1.7
AGD	WSO2 Identity Server-PRE-v1.6
REPOSITORIO	https://github.com/wso2/product-is/releases/tag/v7.0.0
LICENCIA	https://wso2.com/licenses/eula/3.4
UPDATE	https://product-dist.wso2.com/downloads/update-tool/release/update.zip
MySQL	https://dev.mysql.com/downloads/connector/j/
CPSTIC	https://cpstic.ccn.cni.es/es/catalogo-productos-servicios-stic

8 ABREVIATURAS

AD	Active Directory
API	Application Programming Interface
CA	Certificate Authority
CDN	Content Delivery Network
CMD	Command Prompt
CN	Common Name
CPSTIC	Catálogo de Productos y Servicios de Seguridad de las Tecnologías de la Información y la Comunicación.
CPU	Central Processing Unit
CSR	Certificate Signing Request
DC	Domain Component
DN	Distinguished Name
ENS	Esquema Nacional de Seguridad.
EULA	End User License Agreement
GB	Gigabyte
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol over TLS
IS	Identity Server
IS	Information Systems
JDBC	Java Database Connectivity
JDK	Java Development Kit
LDAP	Lightweight Directory Access Protocol
MD5	Message Digest Algorithm 5
OS	Operating System
OU	Organizational Unit
PC	Personal Computer
PES	Procedimiento de Empleo Seguro
RAM	Random Access Memory
SHA	Secure Hash Algorithm
SSL	Secure Sockets Layer
SSO	Single Sign-On
TLS	Transport Layer Security
TOE	Target of Evaluation
WAF	Web Application Firewall