HP Cloud Service Automation

Software Version: 4.50
Windows ® operating systems

FIPS 140-2 Compliance Configuration Guide

Document Release Date: June 2015
Software Release Date: June 2015

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

The OpenStack® Word Mark and the Square O Design, together or apart, are trademarks or registered trademarks marks of OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

RED HAT READY™ Logo and RED HAT CERTIFIED PARTNER™ Logo are trademarks of Red Hat, Inc.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: **http://h20230.www2.hp.com/selfsolve/manuals**

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: **http://h20229.www2.hp.com/passport-registration.html**

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: **http://www.hp.com/go/hpsoftwaresupport**

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is **http://h20230.www2.hp.com/sc/solutions/index.jsp**

Contents

Chapter 1: Overview

This document provides information on how to configure HP CSA to be compliant with Federal Information Processing Standards (FIPS) 140-2.

FIPS 140-2 is a standard for security requirements for cryptographic modules defined by the National Institute of Standards and Technology (NIST). To view the publication for this standard, go to:

csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

The following information is provided in this document:

- **"Getting Started" on page 7**. Before configuring HP CSA, you must initially prepare your environment by completing tasks to back up affected directories and files and install additional applications required for configuration.
- **" Configure HP CSA for FIPS 140-2 Compliance" on page 9**. Tasks to be completed to configure HP CSA for FIPS 140-2 compliance.
- **"Common HP CSA Tasks" on page 46**. Tasks to start and restart HP CSA are different in a FIPS 140-2 compliant environment. Other common tasks, such as encrypting passwords, remain the same between a standard and FIPS 140-2 compliant HP CSA environment.
- **"Examples Used in this Document" on page 49**. This is a reference for the items and values used in the FIPS 140-2 examples.

Note: Elasticsearch is not supported in FIPS mode. Be sure it is turned off before you configure FIPS 140-2 compliance.

Refer to the following guides for more information about:

- HP CSA technical requirements for FIPS 140-2: *HP CSA FIPS 140-2 Compliance Statement*
- Supported components and versions: *HP Cloud Service Automation System and Software Support Matrix*
- Installation: *HP Cloud Service Automation Installation Guide*
- Configuration: *HP Cloud Service Automation Configuration Guide*

These guides are available from the HP Software Support Web site at http://h20230.www2.hp.com/selfsolve/manuals/ (this site requires that you register with HP Passport).

Chapter 2: Getting Started

Before configuring HP CSA to be compliant with FIPS 140-2, you must complete the following tasks, such as backing up affected directories and files and installing additional applications, to prepare your environment for configuration:

Caution: Do NOT configure any other feature of HP CSA and do not use any of the HP CSA tools before configuring HP CSA to be compliant with FIPS 140-2. If you have configured any feature or used one of the tools, you must re-install HP CSA before you can configure HP CSA to be compliant with FIPS 140-2.

Note: HP CSA that is compliant with FIPS 140-2 supports the Microsoft SQL database and Oracle JRE only. For more information about application and version requirements, refer to the *HP Cloud Service Automation System and Software Support Matrix*.

1. Verify that you are configuring a new or fresh installation of HP CSA version 4.50 to be compliant with FIPS 140-2. You cannot configure an upgraded installation of HP CSA version 4.50 or an installation of HP CSA version 4.50 that is in use. For information on upgrading FIPS 140-2, see the *HP Cloud Service Automation Upgrade Guide*.

2. Stop the global search services as follows:

   a. Right-click on the Elasticsearch 1.5.2 service and select **Stop**.

   b. Right-click on HP Search Service and select **Stop**.

3. Back up the following directories:

   - `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\`
   - `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\`
   - `%CSA_HOME%\jboss-as\standalone\configuration\`
   - `%CSA_HOME%\portal\conf\`
   - `%CSA_HOME%\node.js\`
   - *`<csa_jre>`*`\lib\security`
     (where *`<csa_jre>`* is the directory in which the JRE that is used by HP CSA is installed)

4. Download and install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files from the following sites:

   http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

Refer to the `Readme.txt` file from the downloaded content for information on how to deploy the files and upgrade the JRE used by HP CSA.

5. Download and install the Microsoft Visual C++ 2010 Redistributable Package (x86) from the following site:

   http://www.microsoft.com/en-us/download/details.aspx?id=5555

6. Install the RSA BSAFE Crypto software files. To get this library, contact HP support. Unzip the acquired ZIP file to `<csa_jre>\lib\ext\` (where `<csa_jre>` is the directory in which the JRE that is used by HP CSA is installed.

7. Install the recompiled version NodeJS needed for FIPS compliance. On the system on which HP CSA is installed, unzip the `\fips\nodejs-fips-windows.zip` file to the `%CSA_HOME%\node.js\` directory.

Chapter 3: Configure HP CSA for FIPS 140-2 Compliance

This chapter explains how to configure HP CSA to be compliant with FIPS 140-2.

After you have configured HP CSA for FIPS 140-2 compliance, HP CSA uses or complies with the following:

- RSA BSAFE Crypto software
- Keystore and truststore: PKCS #12
- Asymmetric algorithm: RSA
- Symmetric-key algorithm: AES
- Random number generation algorithm: HMAC DRBG (128-bit)
- Hashing algorithm: SHA-256

Complete the following tasks to configure HP CSA to be compliant with FIPS 140-2:

> **Caution:** Once you have configured HP CSA to be compliant with FIPS 140-2, you cannot revert back to the standard configuration unless you uninstall and re-install HP CSA.

- "Stop HP CSA" on page 47
- "Update applicationContext.xml to be FIPS 140-2 Compliant" on the next page
- "Configure Properties in the Java Security File" on page 11
- "Create an HP CSA Encryption Keystore" on page 12
- "Create a New Keystore and Truststore for Secure Communication" on page 16
- "Re-Encrypt HP CSA Passwords" on page 28
- "Configure HP CSA Properties" on page 31
- "Configure the Marketplace Portal" on page 34
- "Configure the Identity Management Component" on page 39
- "Start HP CSA" on page 44
- "Test Secure Connections" on page 45

Stop HP CSA

HP CSA should not be running while you are configuring it to be compliant with FIPS 140-2.

To stop HP CSA:

1. On the server that hosts HP Cloud Service Automation, navigate to **Start** > **Administrative Tools** > **Services**.
2. Right-click on the HP Cloud Service Automation service and select **Stop**.
3. Right-click on the HP Marketplace Portal service and select **Stop**.
4. If you installed an embedded HP Operations Orchestration instance, right-click on the HP Operations Orchestration Central service and select **Stop**.

Update applicationContext.xml to be FIPS 140-2 Compliant

The `applicationContext.xml` file for the Cloud Service Management Console must be updated to be FIPS 140-2 compliant. Do the following:

1. Open the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\applicationContext.xml` file in a text editor. For example, edit the following file:

   ```
   C:\Program Files\Hewlett-Packard\CSA\jboss-as\standalone\deployments\csa.war\
   WEB-INF\applicationContext.xml
   ```

2. Locate the `START Standard Mode Configuration` comment and comment out the following content that appears between the `START Standard Mode Configuration` and `END Standard Mode Configuration` comments:

   ```
   <bean id="simpleEncryptionConfiguration"
   class="com.hp.csa.security.CSASimplePBEConfig" init-method="init">
   </bean>

   <bean id="configurationEncryptor"
   class="org.jasypt.encryption.pbe.StandardPBEStringEncryptor">
     <property name="config" ref="simpleEncryptionConfiguration" />
   </bean>

   <bean id="propertyConfigurer" class="org.jasypt.spring.properties.
   EncryptablePropertyPlaceholderConfigurer">
     <constructor-arg ref="configurationEncryptor" />
     <property name="locations">
   ```

```
    <list>
      <value>classpath:csa.properties</value>
    </list>
  </property>
</bean>
```

3. Locate the START FIPS Mode Configuration comment and uncomment the following content that appears between the START FIPS Mode Configuration and END FIPS Mode Configuration comments:

   ```
   <bean id="configurationEncryptor"
   class="com.hp.csa.security.util.CSASecurityHelper" />

   <bean id="propertyConfigurer" class=
   "com.hp.csa.security.CSAEncryptablePropertyPlaceholderConfigurer">
     <constructor-arg ref="configurationEncryptor" />
     <property name="locations">
       <list>
         <value>classpath:csa.properties</value>
       </list>
     </property>
   </bean>
   ```

4. Save and close the file.

Configure Properties in the Java Security File

Edit the Java security file for the JRE to add additional security providers and configure properties for FIPS 140-2 compliance. Open the *<csa_jre>*\lib\security\java.security file in an editor (where *<csa_jre>* is the directory in which the JRE that is used by HP CSA is installed) and do the following:

1. For every provider listed (in the format security.provider.<*nn*>=<*provider_name*>), increment the preference order number (<*nn*>) by one. For example, change a provider entry from security.provider.**1**=sun.security.provider.Sun to security.provider.**2**=sun.security.provider.Sun.

2. Add a new default provider (RSA JCE). Add the following provider to the top of the provider list:

   ```
   security.provider.1=com.rsa.jsafe.provider.JsafeJCE
   ```

3. Update the SunJSSE provider to use packages that are compliant with FIPS 140-2.

For example, change the following entry from:

```
security.provider.<nn>=com.sun.net.ssl.internal.ssl.Provider
```

to

```
security.provider.<nn>=com.sun.net.ssl.internal.ssl.Provider JsafeJCE
```

4. Set the default keystore type to PKCS #12. Edit or add the following entry:

   ```
   keystore.type=PKCS12
   ```

5. Add the following entry to ensure RSA BSAFE is used in FIPS 140-2 compliant mode:

   ```
   com.rsa.cryptoj.fips140initialmode=FIPS140_SSL_MODE
   ```

6. Set the default random number generation algorithm to HMAC DRBG with 128-bit security strength:

   ```
   com.rsa.crypto.default.random = HMACDRBG128
   ```

7. Exit and save the `java.security` file.

Create an HP CSA Encryption Keystore

This section describes an example of how to create a keystore, referred to in this document as the HP CSA encryption keystore that is used by HP CSA to encrypt and decrypt a key. This key is used to encrypt and decrypt the data in HP CSA. The validity period assigned to the HP CSA encryption keystore is not used by HP CSA.

The examples used in this document saves the keystore in the `%CSA_HOME%\jboss-as\standalone\configuration\` directory. You may choose to store the keystore in any location; however, you must remember to use that location in any other subsequent example.

Note: In the following examples, `%CSA_HOME%` is the directory in which HP Cloud Service Automation is installed (for example, `C:\Program Files\Hewlett-Packard\CSA`), the `keytool` utility is included with the JRE, and a JRE has been installed for HP CSA in *`<csa_jre>`*.

The following is an example of how to create the HP CSA encryption keystore:

1. Open a command prompt and change directories to `%CSA_HOME%`.

2. Run the following command:

```
"<csa_jre>\bin\keytool" -genkey -alias csa_encryption_key
-validity 365 -keyalg rsa -keysize 2048 -storetype PKCS12
-keystore .\jboss-as\standalone\configuration\csa_encryption_keystore.p12
```

where *<csa_jre>* is the directory in which the JRE that is used by HP CSA is installed.

You can use different values for `-alias`, `-validity`, `-keysize` and `-keystore`. These instructions assume that you will use the `-alias` and `-keystore` values recommended here; you will have to adjust the commands accordingly if you use different values.

Because the HP CSA encryption keystore is used by HP CSA to only encrypt and decrypt a key and not to generate certificates, you can enter any value for `-validity`. The validity period assigned to the HP CSA encryption keystore is not used by HP CSA.

3. Enter a keystore password (referred to in this document as the HP CSA encryption keystore password).

   This password is used to control access to the keystore. This password must be the same as the password you enter for the key in step 5 of this task.

   Note: You must create a password file with this password whenever HP CSA is started. See "Start HP CSA" on page 46 for more information.

4. Follow the prompts to enter your first and last name, organization, and location values.

5. Enter the keystore password you supplied earlier to use as the key password.

   Although `keytool` allows you to enter different passwords for the keystore and the key, the two passwords must be the same to work with HP CSA.

Generate an Encrypted Symmetric Key

This section describes an example of how to generate an encrypted symmetric key that is used by HP CSA to encrypt and decrypt data. This key is also used to encrypt the passwords for the Cloud Service Management Console.

Caution: Do NOT generate the key more than one time.

The following is an example of how to generate an encrypted symmetric key:

1. Open a command prompt and change to the `%CSA_HOME%\Tools\PasswordUtil` directory. For example:

   ```
   C:\Program Files\Hewlett-Packard\CSA\Tools\PasswordUtil
   ```

2. Run the following command (this example uses the same example names from "Create an HP CSA Encryption Keystore" on page 12):

   ```
   "<csa_jre>\bin\java" -jar passwordUtil-standalone.jar genAndEncKey JsafeJCE
   ../../jboss-as/standalone/configuration/csa_encryption_keystore.p12
   <HP CSA encryption keystore password> csa_encryption_key
   ../../jboss-as/standalone/configuration/key.dat
   ```

   > **Note:** The path separators used in the passwordUtil-standalone.jar script options are forward slashes (/). You can also use double backward slashes (\\) as your path separators.

   In this example, the encrypted symmetric key is saved to:

   ```
   %CSA_HOME%\jboss-as\standalone\configuration\key.dat
   ```

   > **Note:** You will use this file name and location when encrypting HP CSA passwords for the Cloud Service Management Console.

   If you used different names for the keystore, alias, or encrypted symmetric key file, here is an example of the command without using the example names:

   ```
   "<csa_jre>\bin\java" -jar "%CSA_HOME%\Tools\PasswordUtil\passwordUtil-
   standalone.jar" genAndEncKey JsafeJCE <HP CSA encryption keystore>
   <HP CSA encryption keystore password>
   <HP CSA encryption keystore alias>
   <location and name of the encrypted symmetric key>
   ```

   > **Note:** If you use path separators in the passwordUtil-standalone.jar script options, use either a single forward slash (/) or double backward slashes (\\) as your path separator.

When to Regenerate the HP CSA Encryption Keystore or Encrypted Symmetric Key

You should not regenerate the HP CSA encryption keystore or encrypted symmetric key unless one of the following occurs:

- The HP CSA encryption keystore or encrypted symmetric key was deleted and is not recoverable.
- The HP CSA encryption keystore or encrypted symmetric key was regenerated and the original file is not recoverable.
- The HP CSA encryption keystore password is not retained.

Locate your situation in the table below and perform the tasks starting at the listed step.

Situation	Start at:
Lost HP CSA encryption keystore	Step 1
Lost encrypted symmetric key	Step 2
Regenerated HP CSA encryption keystore	Step 1
Regenerated encrypted symmetric key	Step 3
Forgotten HP CSA encryption keystore password	Step 1

Tasks to perform:

1. Regenerate the HP CSA encryption keystore (see "Create an HP CSA Encryption Keystore" on page 12).

2. Regenerate the encrypted symmetric key (see "Generate an Encrypted Symmetric Key" on page 13).

3. Encrypt HP CSA passwords (see "Re-Encrypt HP CSA Passwords" on page 28).

4. Configure HP CSA properties (see "Configure HP CSA Properties" on page 31). As applicable, update the `keystore`, `keyAlias`, `encryptedKeyFile`, and `csaTruststorePassword` property values.

5. Reset the password for every organization's LDAP access point:

   Update the passwords for the following users in the CSA_ACCESS_POINT table in the database.

   a. Open an SQL client to your database.

   b. Run the following: `update CSA_ACCESS_POINT set password=null;`

   c. Launch the Cloud Service Management Console by typing the following URL in a supported Web browser: `https://<csahostname>:8444/csa` where *`<csahostname>`* is the fully-qualified domain name of the system on which the Cloud Service Management Console resides.

   d. Log in to the Cloud Service Management Console as the CSA Administrator.

   e. Click the **Organizations** tile.

   f. In the left-navigation frame, select an organization.

   g. From the organization's navigation frame, select **LDAP**.

   h. Enter the password in the **Password** and **Retype Password** fields.

   i. Click **Save Changes**.

   j. Repeat steps f - i for every organization.

Create a New Keystore and Truststore for Secure Communication

To comply with FIPS 140-2, the keystore and truststore (that store the keys and certificates used and other applications) must support PKCS #12: Personal Information Exchange Syntax Standard (PKCS #12). You must create a new keystore and truststore for HP CSA for PKCS #12.

This section describes the process you should follow to obtain, install, and configure a certificate that supports PKCS #12 for use by HP CSA.

Perform the following tasks (described in more detail in the sections that follow the list below):

1. Create the HP CSA server keystore that supports PKCS #12
2. Create HP CSA's certificate, create a truststore that supports PKCS #12, and import certificate(s)
3. Configure the Web server
4. Import the HP Operations Orchestration certificate as a trusted certificate
5. Import the VMware vCenter certificate as a trusted certificate
6. Import the certificates for other applications as trusted certificates
7. Configure client browsers (optional)

> **Note:** In the following examples, `%CSA_HOME%` is the directory in which HP Cloud Service Automation is installed (for example, `C:\Program Files\Hewlett-Packard\CSA`), the `keytool` utility is included with the JRE (you may choose to use a different utility), and a JRE has been installed for HP CSA in *`<csa_jre>`*.

Step 1: Create an HP CSA Server Keystore that Supports PKCS #12

Create the HP CSA server keystore. For example, do the following:

1. Open a command prompt and change directories to `%CSA_HOME%`.

2. Run the following command:

   ```
   "<csa_jre>\bin\keytool" -genkey -alias csa_fips -validity 365
   -keyalg rsa -keysize 2048 -storetype PKCS12
   -keystore .\jboss-as\standalone\configuration\keystore_csaID.p12
   ```

You can use different values for `-alias`, `-validity`, `-keysize` and `-keystore`. These instructions assume that you will use the `-alias` and `-keystore` values recommended here; you will have to adjust the commands accordingly if you use different values.

3. Enter a keystore password (referred to in this document as the HP CSA server keystore password).

   This password is used to control access to the keystore. This password must be the same as the password you enter for the key in task 6 of this step.

4. When you are prompted for your first and last name, enter the fully qualified domain name of the HP CSA server.

5. Follow the prompts to enter the remaining organization and location values.

6. Enter the keystore password you supplied earlier to use as the key password.

   Although `keytool` allows you to enter different passwords for the keystore and the key, the two passwords must be the same to work with HP CSA.

Step 2: Create HP CSA's Certificate, Create a Truststore that Supports PKCS #12, and Import Certificate(s)

This section shows examples on how to export a self-signed certificate, create a Certificate Authority-signed certificate (optional), create the HP CSA server truststore that supports PKCS #12, and import the certificates into the truststore and keystore.

Select the type of certificate you will be using (self-signed or Certificate Authority-signed) and complete one of the applicable sections below.

Using a Self-Signed Certificate

Export a self-signed certificate, create the HP CSA server truststore that supports PKCS #12, and import the self-signed certificate into the HP CSA server truststore. For example:

1. Open a command prompt and change directories to `%CSA_HOME%`.

2. Export a self-signed certificate by exporting HP CSA's certificate:

   a. Run the following command:

      ```
      "<csa_jre>\bin\keytool" -export -alias csa_fips
      -file C:\csa_fips.crt -storetype PKCS12
      -keystore .\jboss-as\standalone\configuration\keystore_csaID.p12
      ```

   b. When you are prompted for a password, enter the HP CSA server keystore password used in step 1 (where you created the HP CSA server keystore that supports PKCS #12).

3. Create a truststore that supports PKCS #12 and import the self-signed certificate:

a. Run the following command:

```
"<csa_jre>\bin\keytool" -importcert -alias csa_fips
-file C:\csa_fips.crt -trustcacerts
-keystore .\jboss-as\standalone\configuration\csa_server_truststore.p12
```

b. When prompted, enter a truststore password (referred to in this document as the HP CSA server truststore password). You will need this password when you import the HP Operations Orchestration and other certificates.

c. Enter `yes` when prompted to trust the certificate.

Using a Certificate Authority-Signed Certificate

Create a self-signed certificate, create a Certificate Authority-signed certificate, import the Certificate Authority-signed certificate into the HP CSA server keystore, create the HP CSA server truststore that supports PKCS #12, and import the root certificate into the HP CSA server truststore. For example:

1. Open a command prompt and change directories to `%CSA_HOME%`.

2. To create a Certificate Authority-signed certificate, you must create a certificate signing request and submit the certificate signing request to a Certificate Authority:

   a. From the command prompt, run the following command:

   ```
   "<csa_jre>\bin\keytool" -certreq -alias csa_fips -file C:\csacsrfips.csr
   -keystore .\jboss-as\standalone\configuration\keystore_csaID.p12
   ```

   b. When you are prompted for a password, enter the HP CSA server keystore password used in step 1 (where you created the HP CSA server keystore that supports PKCS #12).

   c. Submit the Certificate Signing Request (`C:\csacsrfips.csr`) to the Certified Authority following the procedure used by your organization or a third-party provider. After the submission has been processed, you will receive a Certificate Authority-signed certificate (referred to as `C:\ca_signed.crt` in the example below) and a root certificate (referred to as `C:\ca_root.crt` in the example below) for the Certificate Authority.

3. Import the Certificate Authority-signed certificate into the HP CSA server keystore:

   a. Open a command prompt and change directories to `%CSA_HOME%`.

   b. From the command prompt, run the following command:

   ```
   "<csa_jre>\bin\keytool" -importcert -alias ca_signed -file C:\ca_signed.crt
   -keystore .\jboss-as\standalone\configuration\keystore_csaID.p12
   ```

   c. When you are prompted for a password, enter the HP CSA server keystore password used in step 1 (where you created the HP CSA server keystore that supports PKCS #12).

4. Create a truststore that supports PKCS #12 and import the root certificate:

a. From the command prompt, run the following command:

```
"<csa_jre>\bin\keytool" -importcert -alias ca_root
-file C:\ca_root.crt -trustcacerts
-keystore .\jboss-as\standalone\configuration\csa_server_truststore.p12
```

b. When prompted, enter a truststore password (referred to in this document as the HP CSA server truststore password). You will need this password when you import the HP Operations Orchestration and other certificates.

c. Enter `yes` when prompted to trust the certificate.

Step 3: Configure the Web Server

1. Encrypt the HP CSA server keystore password and datasource (database) password using the JBoss vault script. Do the following:

   a. Verify that the %JAVA_HOME% environment variable has been defined and that %JAVA_HOME% has been set to the directory in which the JRE that is used by HP CSA is installed (for example, `C:\Program Files\Hewlett-Packard\CSA\openjre`).

   > **Note:** Do NOT enclose the value in quotation marks, even if the path name includes a space. The vault script will fail if the JAVA_HOME variable definition contains quotation marks.

   To verify that %JAVA_HOME% has been defined, from a command prompt, type:

   ```
   echo %JAVA_HOME%
   ```

   b. Create a keystore used by vault. This vault keystore is used to store the HP CSA keystore password.

   > **Note:** This example saves the vault keystore and encrypted vault file in the `%CSA_HOME%\jboss-as\standalone\configuration\` directory (the contents of this directory are automatically backed up during an upgrade). You may choose to store the vault keystore and encrypted vault file in any location. However, you must remember to use those locations in subsequent steps in this task and, if those locations are not automatically backed up during upgrade, to manually back up the files before upgrade.

      i. Open a command prompt.

      ii. Run the following command:

```
"<csa_jre>\bin\keytool" -genkey -alias vault -validity 365 -keyalg rsa
-keysize 2048 -storetype JKS -keystore .\jboss-
as\standalone\configuration\csa_vault.keystore
```

where *<csa_jre>* is the directory in which the JRE that is used by HP CSA is installed.

You can use different values for `-alias`, `-validity`, `-keysize` and `-keystore`. These instructions assume that you will use the `-alias` and `-keystore` values recommended here; you will have to adjust the commands accordingly if you use different values.

iii. Enter the vault keystore password (for example, csavault).

This password is used to control access to the vault keystore. This password must be the same as the password you enter for the key in step e of this task.

iv. Follow the prompts to enter your first and last name, organization, and location values.

v. Enter the key password. Click **Enter** to use the vault keystore password you supplied earlier (for example, csavault).

Although `keytool` allows you to enter different passwords for the keystore and the key, the two passwords must be the same to work with HP CSA.

c. Run the vault script. The script will generate the masked password and the values to configure in the `standalone.xml` file in order to use the masked password.

i. From the command prompt, type: `%CSA_HOME%\jboss-as\bin\vault`

ii. Select **0** to start the interactive session.

iii. Enter the following information, when prompted, to configure the vault keystore:

Prompt	Description
Directory to store encrypted files	Directory in which the vault encrypted file is stored (for example, `%CSA_HOME%\jboss-as\standalone\configuration`). Verify that a vault encrypted file (`VAULT.dat`) does not already exist in this directory. If the file exists, select a different directory.
Keystore URL	The name and location of the vault keystore (for example, `%CSA_HOME%\jboss-as\standalone\configuration\csa_vault.keystore`).
Keystore password (twice)	The password to the vault keystore (for example, csavault).

Prompt	Description
8 character salt	A random number (for example, 12345678).
Iteration count as a number	The number of times the HP CSA keystore password is hashed (for example, 25).
Keystore alias	The alias used to identify the HP CSA keystore password in the vault keystore (for example, vault).

iv. Make a copy of the vault property block that is displayed. For example, copy:

```
<vault>
   <vault-option name="KEYSTORE_URL" value="%CSA_HOME%\jboss-
as\standalone\configuration\csa_vault.keystore"/>
   <vault-option name="KEYSTORE_PASSWORD" value="MASK-2PtpNyQsI1E7t"/>
   <vault-option name="KEYSTORE_ALIAS" value="vault"/>
   <vault-option name="SALT" value="12345678"/>
   <vault-option name="ITERATION_COUNT" value="25"/>
   <vault-option name="ENC_FILE_DIR" value="%CSA_HOME%\jboss-
as\standalone\configuration\"/>
</vault>
```

You will need to add this content to the `standalone.xml` file (the exact location is described in a later step).

v. Select **0** to store a secured attribute.

vi. Enter the following information, when prompted, to generate the vault entry to use for the HP CSA keystore password in the `standalone.xml` file:

Prompt	Description
Secured attribute value (twice)	Enter the HP CSA keystore password (for example, <*HP CSA server keystore password*>).
Vault Block	Enter a name for the vault block (for example, csa_keystore).
Attribute Name	Enter the attribute being stored (for example, password).

Note the VAULT entry (for example, `VAULT::csa_keystore::password::1`). You will need this value when you configure the `standalone.xml` file.

vii. Enter **2** to exit the script.

Note: The vault script converts the format of the vault keystore (for example, %CSA_HOME%\jboss-as\standalone\configuration\csa_vault.keystore) to JCEKS.

2. Open `%CSA_HOME%\jboss-as\standalone\configuration\standalone.xml` in a text editor.

3. Locate the following entry for the HP CSA server keystore password (this entry may have been modified):

   ```
   <ssl>
       <keystore keystore-password="..." path="%CSA_HOME%/jboss-
   as/standalone/configuration/.keystore"/>
   </ssl>
   ```

4. Update the entry by:

   - Adding or changing the value of the password to the encrypted value of the HP CSA server keystore password you generated in task 1 of this step.
   - Changing the value of the path to the keystore you created in step 1 (`%CSA_HOME%\jboss-as\standalone\configuration\keystore_csaID.p12`)
   - Adding the attribute provider and setting its value to `PKCS12`

   For example:

   ```
   <ssl>
       <keystore provider="PKCS12" path="%CSA_HOME%/jboss-
   as/standalone/configuration/keystore_csaID.p12" keystore-
   password="${VAULT::csa_keystore:password::1}"/>
   </ssl>
   ```

5. Locate the following entry for the datasource password (this entry may have been modified):

   ```
   <datasource jndi-name="java:jboss/datasources/csaDS" pool-name="mssqlDS">
      <connection-
   url>jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=request</connection-url>
      <driver>mssqlDriver</driver>
      <pool>
         <min-pool-size>10;</min-pool-size>
         <max-pool-size>200;</max-pool-size>
         <prefill>true;</prefill>
      </pool>
      <security>
         <security-domain>csa-encryption-sec;</security-domain>
      </security>
   </datasource>
   ```

6. Replace the security-domain entry with the datasource user name and password, setting the password value to the encrypted value of the datasource password you generated in task 1 of this step. For Microsoft SQL Server, also update the `connection-url ssl` attribute value from `request` to `authenticate` (if it has not already been updated).

For example:

```
<datasource jndi-name="java:jboss/datasources/csaDS" pool-name="mssqlDS">
   <connection-url>
      jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=~~request~~authenticate
   </connection-url>
   <driver>mssqlDriver</driver>
   <pool>
      <min-pool-size>10;</min-pool-size>
      <max-pool-size>200;</max-pool-size>
      <prefill>true;</prefill>
   </pool>
   <security>
      ~~<security-domain>csa-encryption-sec;</security-domain>~~
      <user-name>datasource_username</user-name>
      <password>
         ${VAULT::csa_keystore::password::1}
      </password>
   </security>
<datasource>
```

7. Locate and delete the following entry for the datasource password (this entry may have been modified):

```
<security-domain name="csa-encryption-sec" cache-type="default">
   <authentication>
      <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
         <module-option name="username" value="<old_user_name>"/>
         <module-option name="password" value="<old_encoded_password>"/>
         <module-option name="managedConnectionFactoryName"
value="jboss.jca:service=LocalTxCM,name=mssqlDS"/>
      </login-module>
   </authentication>
</security-domain>
```

8. Locate the following entry for the datasource password (this entry may have been modified):

```
<datasource enabled="true" jndi-name="java:jboss/datasources/idmDS"
jta="true" pool-name="IdMDS" use-ccm="true" use-java-context="true">
   <connection-
url>jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=request</connection-url>
  <driver>pqsqlDriver</driver>
   <pool>
      <min-pool-size>10;</min-pool-size>
      <max-pool-size>200;</max-pool-size>
      <prefill>true</prefill>
      <use-strict-min>false</use-strict-min>
      <flush-strategy>FailingConnectionOnly</flush-strategy>
   </pool>
   <security>
      <security-domain>idm-encryption-sec;</security-domain>
  </security>
</datasource>
```

9. Replace the `security-domain` entry with the datasource user name and password. Set the password value to the encrypted value of the datasource password you generated in task 1 of this step. For Microsoft SQL Server, also update the `connection-url` ssl attribute value from request to authenticate (if it has not already been updated).

   For example:

```
<datasource jta="true" jndi-name="java:jboss/datasources/idmDS" pool-
name="IdMDS" enabled="true" use-java-context="true" use-ccm="true">
   <connection-
url>jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=~~request~~authenticate
   </connection-url>
   <driver>mssqlDriver</driver>
   <pool>
      <min-pool-size>10</min-pool-size>
      <max-pool-size>200</max-pool-size>
      <prefill>true</prefill>
      <use-strict-min>false</use-strict-min>
      <flush-strategy>FailingConnectionOnly</flush-strategy>
   </pool>
  <security>
      ~~<security-domain>idm-encryption-sec</security-domain>~~
      <user-name>datasource_username</user-name>
      <password>${VAULT::csa_keystore::password::1}</password>
   </security>
</datasource>
```

10. Locate and delete the following entry for the datasource password (this entry may have been modified):

```
<security-domain cache-type="default" name="idm-encryption-sec">
   <authentication>
      <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
            <module-option name="username" value="<old_user_name>"/>
            <module-option name="password" value="<old_encoded_password>"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:service=LocalTxCM,name=IdMDS"/>
      </login-module>
   </authentication>
</security-domain>
```

11. In standalone.xml add new properties to system-properties section. Copy this:

```
<property name="javax.net.ssl.trustStore" value="%CSA_HOME%/jboss-
as/standalone/configuration/csa_server_truststore.p12"/>
<property name="javax.net.ssl.trustStorePassword" value="${VAULT::csa_
keystore::password::1}"/> <!-- vault encrypted password for csa_server_
truststore.p12 -->
<property name="javax.net.ssl.trustStoreType" value="PKCS12"/>
<property name="jsse.enableCBCProtection" value="false"/>
<property name="com.sun.net.ssl.enableECC" value="false"/>
```

12. Add the vault property block to `<server xmlns="urn:jboss:domain:1.3">` after the `<systemproperties>` block. For example, using the example values, enter the following:

```
<server xmlns="urn:jboss:domain:1.3">
.
.
.
<system-properties>
.
.
.
</system-properties>
<vault>
   <vault-option name="KEYSTORE_URL" value="%CSA_HOME%\jboss-
as\standalone\configuration\csa_vault.keystore"/>
   <vault-option name="KEYSTORE_PASSWORD" value="MASK-2PtpNyQsI1E7t"/>
   <vault-option name="KEYSTORE_ALIAS" value="vault"/>
   <vault-option name="SALT" value="12345678"/>
   <vault-option name="ITERATION_COUNT" value="25"/>
   <vault-option name="ENC_FILE_DIR" value="%CSA_HOME%\jboss-
as\standalone\configuration\"/>
</vault>
```

Step 4: Import the HP Operations Orchestration Certificate as a Trusted Certificate

Because the integration of HP CSA and HP Operations Orchestration requires a secure connection, you must import the HP Operations Orchestration certificate.

For each system running HP CSA, import the root certificate of each HP Operations Orchestration's Certificate Authority (you must first export HP Operations Orchestration's certificate from HP Operations Orchestration's truststore and then import it into the HP CSA server truststore).

The following is an example of how to export the HP Operations Orchestration certificate and import it into the HP CSA server truststore.

1. On the system running HP Operations Orchestration, open a command prompt and change the directory to `%ICONCLUDE_HOME%` (Windows) or `$ICONCLUDE_HOME` (Linux).

2. Run the following command:

   HP Operations Orchestration 10.x, Windows

   ```
   .\java\bin\keytool -exportcert -alias tomcat -file C:\oo.crt
   -keystore .\Central\var\security\key.store -storepass changeit
   ```

   HP Operations Orchestration 9.x, Windows

   ```
   .\jre1.6\bin\keytool -exportcert -alias pas -file C:\oo.crt
   -keystore .\Central\conf\rc_keystore -storepass bran507025
   ```

   HP Operations Orchestration 10.x, Linux

   ```
   ./java/bin/keytool -exportcert -alias tomcat -file /tmp/oo.crt
   -keystore ./Central/var/security/key.store -storepass changeit
   ```

   HP Operations Orchestration 9.x, Linux

   ```
   ./jre1.6/bin/keytool -exportcert -alias pas -file /tmp/oo.crt
   -keystore ./Central/conf/rc_keystore -storepass bran507025
   ```

   where `C:\oo.crt` and `/tmp/oo.crt` are examples of filenames and locations used to store the exported root certificate (you can choose a different filename and location).

3. If HP Operations Orchestration is not running on the same system as HP CSA, copy `oo.crt` from the HP Operations Orchestration system to the system running HP CSA (in this example, the file is copied to `C:\`).

4. On the system running HP CSA, change the directory to `%CSA_HOME%` and run the following command:

   ```
   "<csa_jre>\bin\keytool" -importcert -alias pas -file C:\oo.crt
   -keystore .\jboss-as\standalone\configuration\csa_server_truststore.p12
   -storepass <HP CSA server truststore password>
   ```

5. When prompted to trust the certificate, enter `yes`.

Step 5: Import the Provider's Certificate as a Trusted Certificate

If you configure the access point to HP Matrix Operating Environment, HP Server Automation, VMware vCenter, or any provider in the Cloud Service Management Console to use a secure connection, you must import the provider's certificate into the truststore.

For each system running HP CSA, import the root certificate of the provider's Certificate Authority into the truststore (you must first export the provider's certificate from the provider's truststore and then import it into the HP CSA server truststore).

The following is an example of how to import the VMware vCenter certificate into the HP CSA server truststore.

1. Obtain the root certificate of VMware vCenter's Certificate Authority and copy it to the system running HP Cloud Service Automation (in this example, the file is copied to `C:\vcenter.crt`).

2. On the system running HP CSA, change the directory to %CSA_HOME% and run the following command:

   ```
   "<csa_jre>\bin\keytool" -importcert -alias vcenter -file C:\vcenter.crt
   -keystore .\jboss-as\standalone\configuration\csa_server_truststore.p12
   -storepass <HP CSA server truststore password>
   ```

3. When prompted to trust the certificate, enter `yes`.

Step 6: Import the Certificates for other Applications as Trusted Certificates

If other applications, such as the database, LDAP, SMTP, HP Operations Orchestration Load Balancer, or HP Continuous Delivery Automation require a secure connection, you must import the other applications' certificates into the HP CSA server truststore.

The following is an example of how to import another application's certificate into the HP CSA server truststore.

1. Export the certificate for the application and copy the certificate file to the system running HP CSA.

2. Import this certificate into the HP CSA server truststore.

   For example, run the following command on the system running HP CSA:

   ```
   "<csa_jre>\bin\keytool" -importcert -alias <alias>
   -file <filename.crt> -trustcacerts -keystore
   "%CSA_HOME%\jboss-as\standalone\configuration\csa_server_truststore.p12"
   -storepass <HP CSA server truststore password>
   ```

Step 7: Configure Client Browsers (Optional)

If HP CSA's certificate is not signed by a Certificate Authority, when accessing the Cloud Service Management Console, warning messages are displayed in the browser (these messages do not affect normal operations of HP CSA). To avoid these warning messages, import the `csa_fips.crt` file or add an exception.

- **Microsoft Internet Explorer** and **Chrome**: From Windows Explorer, double-click on the `csa_fips.crt` file to begin the import process. Install the certificate in the Trusted Root Certification Authorities store. For information on how to import the certificate, refer to the browser's online documentation.

- **Firefox**: Add an exception by opening the browser and navigating to `https://<csahostname>:8444/csa` where *<csahostname>* is the fully-qualified domain name of the system on which HP CSA is running. When the **This Connection is Untrusted** page opens, select **I Understand the Risks**, click the **Add Exception** button, verify the Server Location, and click **Confirm Security Exception**. For information on how to import the certificate, refer to the browser's online documentation.

Re-Encrypt HP CSA Passwords

This section describes how to generate and replace the passwords used by HP CSA. You will be generating new passwords using FIPS 140-2 compliant utilities.

> **Note:** In the following instructions, `%CSA_HOME%` is the directory in which HP Cloud Service Automation is installed (for example, `C:\Program Files\Hewlett-Packard\CSA`) and a JRE has been installed for HP CSA in *<csa_jre>*.

Generate and replace the passwords for the following HP CSA properties (default passwords appear in parentheses):

- csaTruststorePassword
- securityAdminPassword (cloud)
- securityCsaReportingUserPassword (cloud)
- securityTransportPassword (csaTransportUser)
- securityOoInboundUserPassword (ooInboundUser)
- securityCdaInboundUserPassword (CDA2CSAIntegration!)
- securityIdmTransportUserPassword (idmTransportUser)
- securityCatalogAggregationTransportUserPassword (cloud)
- securityEncryptedSigningKey (cloud)
- securityCodarIntegrationUserPassword (cloud)

Generate and replace the passwords for the following tools:

- Content archive tool
- Purge tool
- Process definition tool
- Provider tool
- Schema installation tool

To generate and replace existing passwords used by HP CSA, do the following:

1. Open a command prompt and change to the `%CSA_HOME%\Tools\PasswordUtil` directory. For example:

   ```
   C:\Program Files\Hewlett-Packard\CSA\Tools\PasswordUtil
   ```

2. Generate a password by running the following command (this example uses the same example names from "Create an HP CSA Encryption Keystore" on page 12):

   ```
   "<csa_jre>\bin\java" -jar passwordUtil-standalone.jar encrypt <password>
   JsafeJCE ../../jboss-as/standalone/configuration/csa_encryption_keystore.p12
   <HP CSA encryption keystore password> csa_encryption_key
   ../../jboss-as/standalone/configuration/key.dat
   ```

   > **Note:** The path separators used in the `passwordUtil-standalone.jar` script options are forward slashes (/). You can also use double backward slashes (\\) as your path separators.

   The encrypted value of the password is displayed.

   If you used different names for the keystore, alias, or encrypted symmetric key file, here is an example of the command without using the example names:

   ```
   "<csa_jre>\bin\java" -jar "%CSA_HOME%\Tools\PasswordUtil\passwordUtil-
   standalone.jar" encrypt <password> JsafeJCE <HP CSA encryption keystore>
   <HP CSA encryption keystore password>
   <HP CSA encryption keystore alias>
   <location and name of the encrypted symmetric key>
   ```

   > **Note:** If you use path separators in the `passwordUtil-standalone.jar` script options, use either a single forward slash (/) or double backward slashes (\\) as your path separator.

3. To update HP CSA properties used by the Cloud Service Management Console, edit the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\classes\csa.properties` file. Update the password for the following properties:

- csaTruststorePassword
- securityAdminPassword
- securityCsaReportingUserPassword
- securityTransportPassword (use the same password for the Identity Management component)
- securityOoInboundUserPassword
- securityCdaInboundUserPassword
- securityIdmTransportUserPassword (use the same password for the Identity Management component and Marketplace Portal)
- securityCatalogAggregationTransportUserPassword
- securityEncryptedSigningKey (use the same password for the Identity Management component)
- securityCodarIntegrationUserPassword

See "Configure the Identity Management Component" on page 39 for more information about configuring passwords for the Identity Management component.

> **Note:** In the properties file, the encrypted password value must be preceded by `ENC` without any separating spaces and is enclosed in parentheses.

For more information about these properties, refer to the *HP Cloud Service Automation Configuration Guide*.

4. Update the password property value defined in the database property file for the following tools:

   - Content archive tool
   - Purge tool
   - Process definition tool
   - Provider tool
   - Schema installation tool

Configure HP CSA Properties

To configure HP CSA properties for FIPS 140-2 compliance:

1. Open a command prompt and change to the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\classes` directory. For example:

   ```
   C:\Program Files\Hewlett-Packard\CSA\jboss-as\standalone\deployments\csa.war\
   WEB-INF\classes
   ```

2. Open the `csa.properties` file in an editor.

   a. Verify that the `enableHPSSO` property is either set to false or is commented out.

   b. Configure the following properties:

Property	Description
useExternalProvider	Required. For FIPS 140-2 compliance, uncomment and set this property to true. When enabled, HP CSA uses the RSA BSAFE libraries to encrypt and decrypt passwords. If a password was encrypted using different libraries (for example, if the password was encrypted before this property is enabled), the resulting decrypted password will not be valid. If you cannot connect to the database after you have configured HP CSA for FIPS 140-2 compliance, try re-encrypting the database password in the database properties file. Default: commented out/disabled
securityProviderName	Required. The name of the FIPS 140-2 compliant provider. By default, HP CSA uses the RSA BSAFE provider and this property should be set to JsafeJCE.

Property	Description
keySize	Optional. The key size used for HP CSA encryption. By default, the key size is 128. If you manually enter a different key size when encrypting a password, uncomment this property and configure the value to the key size used to encrypt the passwords. **Note:** All passwords must be encrypted using the same key size. By default, the password encryption utility encrypts all passwords using a key size of 128 (even if you do not specify a key size when running the utility).
keystore	Required. The absolute path to and file name of the HP CSA encryption keystore. This is the keystore that supports PKCS #12 and stores the key used by HP CSA to encrypt and decrypt data in HP CSA. **Example** (this example uses the same example name from "Create an HP CSA Encryption Keystore" on page 12): `%CSA_HOME%/jboss-as/standalone/configuration/csa_encryption_keystore.p12` **Note:** Use only forward slashes (/) as your path separators.
keyAlias	Required. The alias used to identify the HP CSA encryption key in the HP CSA encryption keystore. **Example** (this example uses the same example name from "Create an HP CSA Encryption Keystore" on page 12): csa_encryption_key

Property	Description
keystorePasswordFile	Required. The absolute path to and file name of the HP CSA encryption keystore password. This is a temporary file that stores the HP CSA encryption keystore password in clear text. This file is required to start the HP CSA service and is automatically deleted when the service is started. The password file must contain only the following content: `keystorePassword=<HP CSA encryption keystore password>` where *<HP CSA encryption keystore password>* is the HP CSA encryption keystore password in clear text. **Note:** Use only forward slashes (/) as your path separators.
encryptedKeyFile	Required. The location of the HP CSA encrypted symmetric key. **Example** (this example uses the same example name from "Create an HP CSA Encryption Keystore" on page 12): `%CSA_HOME%/jboss-as/standalone/configuration/key.dat` **Note:** Use only forward slashes (/) as your path separators.
csaTruststore	Required. The HP CSA keystore that stores trusted Certificate Authority certificates. **Note:** This property is located in another section of the `csa.properties` file. **Example** (this example uses the same example name of the HP CSA server truststore from "Create an HP CSA Encryption Keystore" on page 12): `%CSA_HOME%/jboss-as/standalone/configuration/csa_server_truststore.p12` **Note:** Use only forward slashes (/) as your path separators.

Property	Description
csaTruststorePassword	Required. The encrypted password of the HP CSA keystore (see "Encrypt a Password" on page 47 for instructions on encrypting passwords). An encrypted password is preceded by `ENC` without any separating spaces and is enclosed in parentheses. Default: No default specified **Example** `ENC(9eC7TTnB0uGOGK5U648UITcEV5AuV5T)` **Note:** This property is located in another section of the `csa.properties` file. This is the <*HP CSA server truststore password*> from "Create an HP CSA Encryption Keystore" on page 12.

3. Copy the property values from step 2b to the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\classes\idm-security.properties` file. The property values must be the same in both files.

4. When configuring a command line tool, copy the property values from step 2b to its configuration file. Add `;ssl=authenticate` at the end of the database connection string if it is missing.

5. When executing a tool, you must add this system property `"-Djsse.enableCBCProtection=false"`.For example `"java -Djsse.enableCBCProtection=false -jar provider-tool.jar <tool parameters>"`.

> **Note:** Each time the tool is executed, the password file must be created for that execution. The content (format and password) must be the same that was used for the HP CSA startup.

Configure the Marketplace Portal

This section describes how to encrypt passwords for the Marketplace Portal.

Password Encryption

The Marketplace Portal implements password encryption via PBES2 using the NodeJS crypto library. The key is hard coded in the JavaScript (JS), but it is not directly used. Instead, the key is used to decrypt a randomly-generated key that is encrypted and saved in a keyfile, which will be protected by the file system.

Note: Make sure the file system in which the Marketplace Portal exists is protected by the operating system, so that no one without permission can read or edit files or folders.

Encrypt a Password

The Marketplace Portal provides a password utility (`passwordUtil.js`), which you use to encrypt a password and generate a keyfile.

Note: It is recommended that you use the password utility in case the keyfile is deleted or lost, or the passwords need to be re-encrypted (keyfile has changed or the password has changed).

Following is the password utility syntax.

```
cd %CSA_HOME%\portal\bin
..\..\node.js\node passwordUtil.js --help
..\..\node.js\node passwordUtil.js --password <password to encrypt>
```

Following is an example.

```
..\..\node.js\node passwordUtil.js
Please enter password to encrypt -password hidden-
Encrypted password is ENC(TPhdYjB72z+v+pHdscGSkQ==)
```

Note: If the keyfile needs to be regenerated, delete the existing keyfile, as defined in the `mpp.json` file (see next section for the exact location) and run the password utility script (it will generate a keyfile if it does not exist).

Configure Settings for Keyfile, Session ID Cookie Secret, IdM Transport User Password, and SSL Keyfile or Truststore Passphrase

1. Edit the `%CSA_HOME%\portal\conf\mpp.json` file:

```
{
   "uid": "ccue_mpp",
   "port": 8089,
   "defaultOrganizationName": "CSA_CONSUMER",
   "defaultHelpLocale": "en_US",
   "defaultHelpPage": "MarketplacePortal_HELP_CSA.htm",
    "keyfile": "%CSA_HOME%/portal/conf/keyfile",
   "rejectUnauthorized": false,
   "session": {
```

```
   "cookieSecret": "ENC(udA/d1FqxrK26qQlu5cO2w==)",
   "timeoutDuration": 1800,
   "cleanupInterval": 3600
},
"cart": {
   "thresholdQuantity": 20,
   "maximumQuantity": 100
},
"provider": {
   "url": "https://MPAVM0081.hpswlabs.adapps.hp.com:8444",
   "contextPath": "/csa/api/mpp",
   "strictSSL": true,
   "secureProtocol": "SSLv23_method",
   "ca": "C:/csa_fips.crt"
},
"idmProvider": {
   "url": "https://MPAVM0081.hpswlabs.adapps.hp.com:8444",
   "returnUrl": "https://MPAVM0081.hpswlabs.adapps.hp.com:8089",
   "contextPath": "/idm-service",
    "username": "idmTransportUser",
   "password": "ENC(Op4ZJjnG4F8b/jalqUA6WVzgBCGarmazThflGYeX8wY=)",
   "strictSSL": true,
   "secureProtocol": "SSLv23_method",
   "ca": "C:/csa_fips.crt"
},
"https": {
   "enabled": true,
   "options": {
      "passphrase": "ENC(21P/dn5zzdEAvGjEP3Su7A==)",
      "key" : "%CSA_HOME%/portal/conf/.mpp_privateKey.pem",
      "cert" : "%CSA_HOME%/portal/conf/.mpp_publicKey.pem",
      "secureProtocol" : "TLSv1_method",
      "ciphers" : "TLS_RSA_WITH_3DES_EDE_CBC_SHA:HIGH:!MD5:!aNULL:!EDH",
      "honorCipherOrder" : true
   }
},
"ha": {
   "enabled": false,
   "numWorkers": 2,
   "redis": {
      "options": {
        "host": "MPAVM0081.hpswlabs.adapps.hp.com",
        "port": 6379
      }
   }
},
"logging": {
      "console": {
         "enabled": false,
         "level": "info"
   },
   "file": {
      "enabled": true,
      "level": "info",
      "maxSizeMB": 10,
      "maxFile": 10
   },
   "cef": {
      "enabled": false,
      "address": "MPAVM0081.hpswlabs.adapps.hp.com",
      "port": 9876,
      "level": "warn"
      }
   },
   "proxy": {
      "enabled": false,
      "port": 8090,
      "contextPath": "/mpp"
   }
}
```

2. Set the following parameters:

   - `keyfile` is the location of the key file generated by the Marketplace Portal password utility (`passwordUtil.js`). When the keyfile file is not placed in the default location or with a different name, use the `--keyfile` parameter for `passwordUtil.js` and change the path in the `keyfile` parameter in the configuration.

   - `session.cookieSecret` is the secret `passphrase` to encrypt the session ID cookie on the browser. This is an encryptable field, so make sure you enclose it with `enc()`.

   - `idmProvider.password` is the transport user used to connect to Identity Management (IdM). This is an encryptable field, so make sure you enclose it with `enc()`. The default password for `idmProvider.password` is `idmTransportUser`.

   - `https.options.passphrase` is the passphrase of the SSL keyfile or truststore. This is an encryptable field, so make sure you enclose it with `enc()`. The default password for `https.options.passphrase` is `changeit`.

3. Set the correct location to the HP CSA web public certificate (in the HP CSA configuration file named `csa_fips.crt`) for the following:

   - `provider.ca`

   - `idmProvider.ca`

Note: Do not copy the encrypted password from this example, because the encryption key and salt are generated and stored in the keyfile. However, you can reuse the keyfile for multiple systems, and the encrypted password in the `mpp.json` file will be the same.

Configure TLS

The Marketplace Portal uses the NodeJS HTTPS module to enable TLS. OpenSSL is used to perform the encryption and decryption.

FIPS 140-2 supports only TLS. You must configure the Marketplace Portal to use a FIPS-compliant cipher .

To configure the Marketplace Portal to use a FIPS-compliant cipher, do the following:

1. Edit the `%CSA_HOME%\portal\conf\mpp.json` file:

   ```
   "https": {
     "enabled": true,
     "options": {
       "passphrase": "ENC(pEYj2aVNBVUyH85PDnVjZg==))"
       "key": "../conf/.mpp_privateKey.pem",
       "cert": "../conf/.mpp_publicKey.pem",
       "secureProtocol": "TLSv1_method",
       "ciphers": "TLS_RSA_WITH_3DES_EDE_CBC_SHA:HIGH:!MD5:!aNULL:!EDH",
       "honorCipherOrder": true
     }
   },
   ```

2. The `key` and `cert` files should be generated from the `pfx` file (`../conf/.mpp_keystore`).
3. Set the `secureProtocol` parameter to `TLSv1_method`.
4. Set the `ciphers` parameter to `TLS_RSA_WITH_3DES_EDE_CBC_SHA:HIGH:!MD5:!aNULL:!EDH`.
5. Set the `honorCipherOrder` parameter to `true`.

To generate `pem` files from the `.mpp_keystore` you can use these commands:

1. Generate a private key:

   ```
   openssl pkcs12 -in .mpp_keystore -out .mpp_privateKey.pem -nocerts
   ```

2. Generate a public certificate:

   ```
    openssl pkcs12 -in .mpp_keystore -out .mpp_publicKey.pem -nokeys
   ```

3. You will be asked for the password to open the `.mpp_keystore` (default is `changeit`).

4. You will be asked to set the password to secure the private key.

> **Note:** If you use a different password than the default password, encrypt this password with `passwordUtil` and replace the value of the `https.options.passphrase` with this one.

Configure the Identity Management Component

If you are using the Identity Management component, to configure the Identity Management component for FIPS 140-2 compliance, do the following:

1. Update the `applicationContext.xml` file.

2. Re-encrypt passwords.

3. "Update the idm-security.properties File" on page 43.

> **Note:** The examples in this section explain how to configure the Identity Management component that is installed on the same instance as HP CSA, where HP CSA is configured in a standalone environment. If your environment is different, files may be located in a different directory.
>
> In the following instructions, `%CSA_HOME%` is the directory in which HP Cloud Service Automation is installed (for example, `C:\Program Files\Hewlett-Packard\CSA`) and *`<csa_jre>`* is the directory in which the JRE used by HP CSA has been installed.

Update the applicationContext.xml File

The `applicationContext.xml` file for the Cloud Service Management Console must be updated to be FIPS 140-2 compliant. Do the following:

1. Open the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\spring\applicationContext.xml` file in a text editor.

2. Locate the `START Standard Mode Configuration` comment and comment out the following content that appears between the `START Standard Mode Configuration` and `END Standard Mode Configuration` comments:

```
<bean id="simpleEncryptionConfiguration"
class="com.hp.csa.security.CSASimplePBEConfig" init-method="init">
</bean>

<bean id="configurationEncryptor"
```

```
class="org.jasypt.encryption.pbe.StandardPBEStringEncryptor">
  <property name="config" ref="simpleEncryptionConfiguration" />
</bean>

<bean id="propertyConfigurer" class="org.jasypt.spring.properties.
EncryptablePropertyPlaceholderConfigurer">
  <constructor-arg ref="configurationEncryptor" />
  <property name="locations">
    <list>
      <value>classpath:csa.properties</value>
      <value>classpath:swagger.properties</value>
    </list>
  </property>
</bean>
```

3. Locate the `START FIPS Mode Configuration` comment that appears immediately after the `Standard Mode Configuration` section and uncomment the following content that appears between the `START FIPS Mode Configuration` and `END FIPS Mode Configuration` comments:

```
<bean id="configurationEncryptor"
class="com.hp.csa.security.util.CSASecurityHelper" />

<bean id="propertyConfigurer" class=
"com.hp.csa.security.CSAEncryptablePropertyPlaceholderConfigurer">
  <constructor-arg ref="configurationEncryptor" />
  <property name="locations">
    <list>
      <value>/WEB-INF/spring/applicationContext.properties</value>
    </list>
  </property>
</bean>
```

4. Locate the `START FIPS Mode Configuration` comment for the `csaTemplateFactory` bean and uncomment the following content that appears between the `START FIPS Mode Configuration` and `END FIPS Mode Configuration` comments:

```
<property name="fipsEnabled" value="true" />
```

5. Locate the `START FIPS Mode Configuration` comment for the `keystoneTemplateFactory` bean and uncomment the following content that appears between the `START FIPS Mode Configuration` and `END FIPS Mode Configuration` comments:

```
<property name="fipsEnabled" value="true" />
```

6. Save and close the file.

Re-Encrypt Passwords

This section describes how to generate and replace the passwords used by the Identity Management component. You will be generating new passwords using FIPS 140-2 compliant utilities.

Generate and replace the passwords for the following Identity Management component properties:

- idm.csa.password
- idm.encryptedSigningKey
- idm.keystone.transportPassword
- consumer
- idmTransportUser

Note: The default password values for these properties are provided in the steps below (they will appear in parentheses after the property name).

To generate and replace existing passwords used by the Identity Management component, do the following:

1. Open a command prompt and change to the `%CSA_HOME%\Tools\PasswordUtil` directory. For example:

   ```
   C:\Program Files\Hewlett-Packard\CSA\Tools\PasswordUtil
   ```

2. Generate a password by running the following command (this example uses the same example names from "Create an HP CSA Encryption Keystore" on page 12):

   ```
   "<csa_jre>\bin\java" -jar passwordUtil-standalone.jar encrypt <password>
   JsafeJCE ../../jboss-as/standalone/configuration/csa_encryption_keystore.p12
   <HP CSA encryption keystore password> csa_encryption_key
   ../../jboss-as/standalone/configuration/key.dat
   ```

   Note: The path separators used in the `passwordUtil-standalone.jar` script options are forward slashes (/). You can also use double backward slashes (\\) as your path separators.

   The encrypted value of the password is displayed.

   If you used different names for the keystore, alias, or encrypted symmetric key file, here is an example of the command without using the example names:

   ```
   "<csa_jre>\bin\java" -jar "%CSA_HOME%\Tools\PasswordUtil\passwordUtil-
   standalone.jar" encrypt <password> JsafeJCE <HP CSA encryption keystore>
   ```

```
<HP CSA encryption keystore password>
<HP CSA encryption keystore alias>
<location and name of the encrypted symmetric key>
```

> **Note:** If you use path separators in the `passwordUtil-standalone.jar` script options, use either a single forward slash (/) or double backward slashes (\\) as your path separator.

3. Open the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\spring\applicationContext.properties` file in a text editor and do the following:

   a. Update the `idm.csa.password` (csaTransportUser) property. `idm.csa.password` must be the same password you configured for the `securityTransportPassword` property (which is configured in the `csa.properties` file). See "Re-Encrypt HP CSA Passwords" on page 28 for more information about encrypting the `securityTransportPassword` password property.

   b. Update the `idm.encryptedSigningKey` (cloud) property. `idm.encryptedSigningKey` must be the same password you configured for the `securityEncryptedSigningKey` property (which is configured in the `csa.properties` file). See "Re-Encrypt HP CSA Passwords" on page 28 for more information about encrypting the `securityEncryptedSigningKey` password property.

   c. If you are using Keystone, update the `idm.keystone.transportPassword` property. `idm.keystone.transportPassword` must be the password you configured for the user defined by the `idm.keystone.transportUsername` property and is located above the `idm.keystone.transportPassword` property.

   d. Save and close the file.

4. Open the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\classes\csa-consumer-users.properties` file in a text editor and do the following:

   a. Update the `consumer` (cloud,SERVICE_CONSUMER,ROLE_REST,enabled) and `consumerAdmin` (cloud,SERVICE_CONSUMER,ROLE_REST,ROLE_ADMIN,enabled) properties.

   > **Note:** This property not only contains the password, but also the roles that control access to HP CSA and if the account is enabled.
   > This entire value must be encrypted.

   b. Save and close the file.

5. Open the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\classes\csa-provider-users.properties` file in a text editor and do the following:

   a. Update the `admin` (cloud,ROLE_REST,enabled), `csaReportingUser` (cloud,ROLE_REST,ROLE_DYNAMIC,enabled), `cdaInboundUser` (CDA2CSAIntegration!,ROLE_

REST,enabled), `codarIntegrationUser` (cloud,ROLE_REST,enabled), and `ooInboundUser` (cloud,ROLE_REST,enabled) properties.

> **Note:** This property not only contains the password, but also the roles that control access to HP CSA and if the account is enabled.
> This entire value must be encrypted.

   b. Save and close the file.

6. Open the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\classes\integrationusers.properties` file in a text editor and do the following:

   a. Update the `idmTransportUser` (idmTransportUser,ROLE_ADMIN,PERM_IMPERSONATE,enabled) property.

   > **Note:** This property not only contains the password, but also the roles that control access to HP CSA and if the account is enabled.
   > This entire value must be encrypted.

   The password in the `idmTransportUser` value must be the same password you configured for both the `securityIdmTransportUserPassword` property (configured in the `csa.properties` file) and the `password` attribute (configured in the idmProvider section of the `mpp.json` file). See "Re-Encrypt HP CSA Passwords" on page 28 for more information about encrypting the `securityIdmTransportUserPassword` password property. See "Encrypt a Marketplace Portal Password" on page 48 for more information about encrypting the `password` attribute.

   b. Save and close the file.

Update the idm-security.properties File

Enable the FIPS 140-2 security settings in the `idm-security.properties` file. Do the following:

1. Open the `%CSA_HOME%\jboss-as\standalone\deployments\idm-service.war\WEB-INF\classes\idm-service.properties` file in a text editor.

2. Verify that the FIPS 140-2 property values in this file are the same values that are configured in the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\classes\csa.properties` file. You should have already copied these values (see "Configure HP CSA Properties" on page 31 for more information about these properties).

3. Save and close the file.

Initialize the IdM client part in HP CSA

1. In the `<CSA_HOME>/jboss-as/standalone/deployments/csa.war/WEB-INF/web.xml` file, search for FIPS and uncomment the section below

   ```
   <!-- FIPS :: IDM Security Context listener -->
   <!--
   <listener>
   <listener-class>com.hp.ccue.identity.config.SecurityContextListener</listener-
   class>
   </listener>
   -->
   ```

2. Copy the configured `idm-security.properties` file from `idm-service.war/WEB-INF/classes` to `csa.war/WEB-INF/classes`.

Start HP CSA

To start HP CSA:

1. Create an HP CSA encryption keystore password file. The name and location of this file must match the value configured for the `keystorePasswordFile` property in the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\classes\csa.properties` file.

   The password file must contain only the following content:
   `keystorePassword=<`*`HP CSA encryption keystore password`*`>`

   where `<`*`HP CSA encryption keystore password`*`>` is the HP CSA encryption keystore password in clear text.

   This file is automatically deleted when the HP Cloud Service Automation service is started.

2. On the server that hosts HP CSA, navigate to **Start** > **Administrative Tools** > **Services**.

3. Right-click on the HP Cloud Service Automation service and select **Start**.

4. Right-click on the HP Marketplace Portal service and select **Start**.

5. If you installed an embedded HP Operations Orchestration instance, right-click on the HP Operations Orchestration Central service and select **Start**.

After the service has started, review the log files in *`%CSA_HOME%`*`\jboss-as\standalone\log\` and verify that no TLS or keystore errors are present.

Test Secure Connections

To test the connection to the Cloud Service Management Console, on a client system, open a supported Web browser and navigate to `https://<csahostname>:8444/csa` where *`<csahostname>`* is the fully-qualified domain name of the system that was used when the certificate was created. If the client browser is configured to accept HP CSA's certificate and the Web application opens without a certificate warning, then you have successfully configured HP CSA to use HP CSA's certificate. If you did not configure the client browser to accept HP CSA's certificate, verify that the only certificate warning relates to the certificate not being issued by a trusted authority. If any other certificate warning is displayed, review all steps in "Create a New Keystore and Truststore for Secure Communication" on page 16 to be sure they were followed as documented.

Chapter 4: Common HP CSA Tasks

This chapter provides information on how to perform common HP CSA tasks.

> **Note:** Steps for starting and restarting HP CSA that is configured for FIPS 140-2 compliance are different from the steps to start and restart the standard HP CSA product.

Tasks include:

- "Start HP CSA" below
- "Restart HP CSA" on the next page
- "Stop HP CSA" on the next page
- "Encrypt a Password" on the next page
- "Encrypt a Marketplace Portal Password" on page 48

Start HP CSA

To start HP CSA:

1. Create an HP CSA encryption keystore password file. The name and location of this file must match the value configured for the `keystorePasswordFile` property in the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\classes\csa.properties` file.

   The password file must contain only the following content:
   `keystorePassword=<HP CSA encryption keystore password>`

   where *`<HP CSA encryption keystore password>`* is the HP CSA encryption keystore password in clear text.

   This file is automatically deleted when the HP Cloud Service Automation service is started.

2. On the server that hosts HP CSA, navigate to **Start** > **Administrative Tools** > **Services**.

3. Right-click on the HP Cloud Service Automation service and select **Start**.

4. Right-click on the HP Marketplace Portal service and select **Start**.

5. If you installed an embedded HP Operations Orchestration instance, right-click on the HP Operations Orchestration Central service and select **Start**.

Restart HP CSA

To restart HP CSA:

1. Create an HP CSA encryption keystore password file. The name and location of this file must match the value configured for the `keystorePasswordFile` property in the `%CSA_HOME%\jboss-as\standalone\deployments\csa.war\WEB-INF\classes\csa.properties` file.

   The password file must contain only the following content:
   `keystorePassword=<`*`HP CSA encryption keystore password`*`>`

   where `<`*`HP CSA encryption keystore password`*`>` is the HP CSA encryption keystore password in clear text.

   This file is automatically deleted when the HP Cloud Service Automation service is started.

2. On the server that hosts HP Cloud Service Automation, navigate to **Start** > **Administrative Tools** > **Services**.

3. Right-click on the HP Cloud Service Automation service and select **Restart**.

4. Right-click on the HP Marketplace Portal service and select **Restart**.

Stop HP CSA

HP CSA should not be running while you are configuring it to be compliant with FIPS 140-2.

To stop HP CSA:

1. On the server that hosts HP Cloud Service Automation, navigate to **Start** > **Administrative Tools** > **Services**.

2. Right-click on the HP Cloud Service Automation service and select **Stop**.

3. Right-click on the HP Marketplace Portal service and select **Stop**.

4. If you installed an embedded HP Operations Orchestration instance, right-click on the HP Operations Orchestration Central service and select **Stop**.

Encrypt a Password

To encrypt a password (for use with HP CSA configuration only; see "Encrypt a Marketplace Portal Password" on the next page for information on how to encrypt a Marketplace Portal password):

1. Open a command prompt and change to the `%CSA_HOME%\Tools\PasswordUtil` directory. For example:

   ```
   C:\Program Files\Hewlett-Packard\CSA\Tools\PasswordUtil
   ```

2. Generate a password by running the following command (this example uses the same example names from "Create an HP CSA Encryption Keystore" on page 12):

   ```
   "<csa_jre>\bin\java" -jar passwordUtil-standalone.jar encrypt <password>
   JsafeJCE ../../jboss-as/standalone/configuration/csa_encryption_keystore.p12
   <HP CSA encryption keystore password> csa_encryption_key
   ../../jboss-as/standalone/configuration/key.dat
   ```

   > **Note:** The path separators used in the `passwordUtil-standalone.jar` script options are forward slashes (/). You can also use double backward slashes (\\) as your path separators.

   The encrypted value of the password is displayed.

   If you used different names for the keystore, alias, or encrypted symmetric key file, here is an example of the command without using the example names:

   ```
   "<csa_jre>\bin\java" -jar "%CSA_HOME%\Tools\PasswordUtil\passwordUtil-
   standalone.jar" encrypt <password> JsafeJCE <HP CSA encryption keystore>
   <HP CSA encryption keystore password>
   <HP CSA encryption keystore alias>
   <location and name of the encrypted symmetric key>
   ```

   > **Note:** If you use path separators in the `passwordUtil-standalone.jar` script options, use either a single forward slash (/) or double backward slashes (\\) as your path separator.

Encrypt a Marketplace Portal Password

To encrypt a password used by the Marketplace Portal:

1. Open a command prompt and change to the `%CSA_HOME%\portal\bin` directory. For example:

   ```
   C:\Program Files\Hewlett-Packard\CSA\portal\bin
   ```

2. Run the following command:

   ```
   ..\..\node.js\node passwordUtil --keyfilePath <keyfile> --password <myPassword>
   ```

   where <keyfile> is the path to (absolute or relative to the bin directory) and name of the file that contains the Marketplace Portal's encrypted symmetric key (if the file does not exist, it will create the file) and <myPassword> is the password to be encrypted.

Appendix A: Examples Used in this Document

The following table is a quick reference to the items and values used in the FIPS 140-2 examples. Also included are the names used in this document to reference the items. If you choose to use different values for these items, you must substitute the different value in all of the FIPS 140-2 examples in this document.

Item	Referenced as	Description	Value Used in Examples
Directory where HP CSA is installed	%CSA_HOME%	The directory in which the HP CSA product is installed.	`C:\Program Files\Hewlett-Packard\CSA`
Directory where the JRE used by HP CSA is installed	*<csa_jre>*	The directory in which the JRE used by HP CSA is installed. For example, `C:\Program Files\Java\CSAjre\jre`.	*<csa_jre>*
Keystore for encryption	HP CSA encryption keystore	The keystore that stores the keypair that is used to encrypt and decrypt HP CSA's symmetric key (also known as the secret key). HP CSA's symmetric key is used to encrypt and decrypt HP CSA's data.	`%CSA_HOME%\jboss-as\standalone\configuration\csa_encryption_keystore.p12`
Keystore alias for encryption	HP CSA encryption keystore alias	The alias is a name assigned to identify a keypair in the HP CSA encryption keystore. This keypair is used by HP CSA to encrypt and decrypt HP CSA's symmetric key.	csa_encryption_key
Key for encryption	HP CSA encryption keystore file or encrypted symmetric key	This is the file containing HP CSA's encrypted symmetric key and used by HP CSA to encrypt and decrypt data in HP CSA.	`%CSA_HOME%\jboss-as\standalone\configuration\key.dat`
Keystore password for encryption	HP CSA encryption keystore password	This is the password used to access the HP CSA encryption keystore.	*<HP CSA encryption keystore password>*

Item	**Referenced as**	**Description**	**Value Used in Examples**
Keystore for secure communication	HP CSA server keystore	This is a file that stores the keypair used for secure communication and is the identity of the HP CSA server.	`%CSA_HOME%\jboss-as\standalone\configuration\keystore_csaID.p12`
Keystore alias for secure communication	HP CSA server keystore alias	The alias is a name assigned to identify the HP CSA TLS keypair. When used with keytool's `-export` option, the alias is the name used by the HP CSA server keystore to identify the certificate.	csa_fips
Keystore password for secure communication	HP CSA server keystore password	This is the password used to access the HP CSA server keystore.	<*HP CSA server keystore password*>
Certificate for HP CSA	HP CSA's certificate	This is the certificate for HP CSA that must be imported into an application's truststore if HP CSA communicates with this application using TLS.	`C:\csa_fips.crt`
Truststore for secure communication	HP CSA server truststore	This is the truststore that holds all certificates for trusted applications that communicate with HP CSA using TLS.	`%CSA_HOME%\jboss-as\standalone\configuration\csa_server_truststore.p12`
Truststore alias for secure communication	HP CSA server truststore alias	When used with keytool's `-import` option, the alias is a name assigned to identify the certificate imported into the HP CSA truststore. Typically the truststore alias is identical to the keystore alias used to generate the certificate.	csa_fips (alias for HP CSA's certificate) pas (alias for the root certificate of HP Operations Orchestration's Certificate Authority)
Truststore password for secure communication	HP CSA server truststore password	This is the password used to access the HP CSA server truststore.	<*HP CSA server truststore password*>

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on FIPS 140-2 Compliance Configuration Guide (Cloud Service Automation 4.50)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to clouddocs@hp.com.

We appreciate your feedback!

Java™
COMPATIBLE