HP Operations Orchestration

Softwareversion: 10.20
Betriebssysteme Windows und Linux

Optimierungshandbuch

Datum der Dokumentveröffentlichung: November 2014
Datum des Software-Release: November 2014

Rechtliche Hinweise

Garantie

Die Garantiebedingungen für Produkte und Services von HP sind in der Garantieerklärung festgelegt, die diesen Produkten und Services beiliegt. Keine der folgenden Aussagen kann als zusätzliche Garantie interpretiert werden. HP haftet nicht für technische oder redaktionelle Fehler oder Auslassungen.

Die hierin enthaltenen Informationen können ohne vorherige Ankündigung geändert werden.

Eingeschränkte Rechte

Vertrauliche Computersoftware. Gültige Lizenz von HP für den Besitz, Gebrauch oder die Anfertigung von Kopien erforderlich. Entspricht FAR 12.211 und 12.212. Kommerzielle Computersoftware, Computersoftwaredokumentation und technische Daten für kommerzielle Komponenten werden an die US-Regierung per Standardlizenz lizenziert.

Copyright-Hinweis

Markenhinweise

Adobe™ ist eine Marke von Adobe Systems Incorporated.

Microsoft® und Windows® sind in den USA eingetragene Marken der Microsoft Corporation.

UNIX® ist eine eingetragene Marke von The Open Group.

Dieses Produkt enthält eine Schnittstelle der freien Programmbibliothek zum Komprimieren, 'zlib', geschützt durch Copyright © 1995-2002 Jean-loup Gailly und Mark Adler.

Danksagungen

Aktualisierte Dokumentation

Auf der Titelseite dieses Dokuments befinden sich die folgenden identifizierenden Informationen:

- Software-Versionsnummer, die Auskunft über die Version der Software gibt.
- Datum der Dokumentveröffentlichung, das bei jeder Änderung des Dokuments ebenfalls aktualisiert wird.
- Datum des Software-Release, das angibt, wann diese Version der Software veröffentlicht wurde.

Unter der unten angegebenen Internetadresse können Sie überprüfen, ob neue Updates verfügbar sind, und sicherstellen, dass Sie mit der neuesten Version eines Dokuments arbeiten:
http://h20230.www2.hp.com/selfsolve/manuals

Für die Anmeldung an dieser Website benötigen Sie einen HP Passport. Hier können Sie sich für eine HP Passport-ID registrieren: **http://h20229.www2.hp.com/passport-registration.html**

Alternativ können Sie auf den Link **New user registration** (Neuen Benutzer registrieren) auf der HP Passport-Anmeldeseite klicken.

Wenn Sie sich beim Support-Service eines bestimmten Produkts registrieren, erhalten Sie ebenfalls aktualisierte Softwareversionen und überarbeitete Ausgaben der zugehörigen Dokumente. Weitere Informationen erhalten Sie bei Ihrem HP-Kundenbetreuer.

Support

Besuchen Sie die HP Software Support Online-Website von HP unter:
http://www.hp.com/go/hpsoftwaresupport

Auf dieser Website finden Sie Kontaktinformationen und Details zu Produkten, Services und Support-Leistungen von HP Software.

Der Online-Support von HP Software bietet Kunden mit Hilfe interaktiver technischer Support-Werkzeuge die Möglichkeit, ihre Probleme intern zu lösen. Als Valued Support Customer können Sie die Support-Website für folgende Aufgaben nutzen:

- Suchen nach interessanten Wissensdokumenten
- Absenden und Verfolgen von Support-Fällen und Erweiterungsanforderungen
- Herunterladen von Software-Patches
- Verwalten von Support-Verträgen
- Nachschlagen von HP-Support-Kontakten
- Einsehen von Informationen über verfügbare Services
- Führen von Diskussionen mit anderen Softwarekunden
- Suchen und Registrieren für Softwareschulungen

Für die meisten Support-Bereiche müssen Sie sich als Benutzer mit einem HP Passport registrieren und anmelden. In vielen Fällen ist zudem ein Support-Vertrag erforderlich. Hier können Sie sich für eine HP Passport-ID registrieren:

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

Weitere Informationen zu Zugriffsebenen finden Sie unter:

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

HP Software Solutions Now greift auf die Website von HPSW Solution and Integration Portal zu. Auf dieser Website finden Sie HP-Produktlösungen für Ihre Unternehmensanforderungen, einschließlich einer Liste aller Integrationsmöglichkeiten zwischen HP-Produkten sowie eine Aufstellung der ITIL-Prozesse. Der URL dieser Website lautet **http://h20230.www2.hp.com/sc/solutions/index.jsp**

Inhalt

Optimieren von HP Operations Orchestration

Dieses Dokument beschreibt die Optimierung der Sicherheit für HP Operations Orchestration.

Weitere Informationen zu Verwaltungsaufgaben finden Sie im *HP OO Administration Guide*.

Haftungsausschluss

Dieses Handbuch enthält Empfehlungen zum Sichern Ihrer HP OO-Bereitstellung gegen Sicherheitsrisiken oder -bedrohungen. Einige der wichtigsten Gründe für das Sichern einer Anwendung sind der Schutz der Vertraulichkeit, Integrität und Verfügbarkeit von wichtigen Informationen einer Organisation. Zum Schutz Ihrer HP OO-Daten ist es jedoch erforderlich, sowohl HP OO als auch die IT-Umgebung (z. B. die Infrastruktur), auf der die Anwendung ausgeführt wird, zu sichern.

Dieses Handbuch beschränkt sich auf Empfehlungen zum Sichern von HP OO auf Anwendungsebene. Informationen zum Sichern der Infrastruktur innerhalb der Kundenumgebung sind nicht enthalten. Für das Verständnis seiner Infrastruktur/Umgebung und das Anwenden der entsprechenden Richtlinien zum Optimieren der Sicherheit ist der Kunde allein verantwortlich.

Empfehlungen zum Optimieren der Sicherheit

1. Installieren Sie die neueste Version von HP OO. Weitere Informationen finden Sie im *HP OO Installation Guide*.
2. (Optional) Konfigurieren der FIPS 140-2-Konformität in HP OO Wenn Sie so vorgehen, muss die Konfiguration durchgeführt werden, bevor Sie den Central-Server starten. Weitere Informationen finden Sie unter " Konfigurieren der FIPS 140-2-Konformität Stufe 1 in HP OO" auf Seite 22.
3. Konfigurieren Sie das Central-Serverzertifikat für die TLS-Verschlüsselung und das Clientzertifikat für die strikte Authentifizierung (gegenseitig).

   Hinweis: Dies kann während der Installation erfolgen.

   Für RAS, Debugger und OOSH geben Sie, falls erforderlich, die Authentifizierung mit Zertifikat an (für das Serverzertifikat) und verwenden das Clientzertifikat für die Authentifizierung bei Central. Weitere Informationen finden Sie unter "Server- und Clientauthentifizierung über Zertifikate" auf der nächsten Seite.
4. Sichern Sie den HP OO Central-Server, indem Sie den HTTP-Port entfernen und die Kennwörter von KeyStore und TrustStore durch sichere Kennwörter ersetzen. Weitere Informationen finden Sie unter "Ändern oder Deaktivieren der HTTP/HTTPS-Ports" auf Seite 14 und "Ändern des Kennworts für den KeyStore/TrustStore" auf Seite 11.
5. Sichern Sie HP OO Studio, indem Sie die Kennwörter von KeyStore und TrustStore durch sichere Kennwörter ersetzen. Weitere Informationen finden Sie unter "Ändern des Kennworts für den KeyStore/TrustStore" auf Seite 11.
6. Entfernen der RC4-Verschlüsselung aus den unterstützten SSL-Verschlüsselungsverfahren Weitere Informationen finden Sie unter "Entfernen der RC4-Verschlüsselung aus den unterstützten SSL-Verschlüsselungsverfahren" auf Seite 14.

7. (Optional) Konfigurieren der TLS-Protokollversion. Weitere Informationen finden Sie unter "Konfigurieren des TLS-Protokolls" auf Seite 28.
8. Aktivieren Sie die Authentifizierung in Central. Weitere Informationen finden Sie unter "Aktivieren der Authentifizierung" im *HP OO Central-Benutzerhandbuch*.

   Da interne Benutzer nicht gesichert sind, sollten Sie ein sicheres LDAP mit einer sicheren Kennwortrichtlinie verwenden. Weitere Informationen finden Sie unter "Einrichten der Sicherheitseinstellungen – LDAP-Authentifizierung" im *HP OO Central-Benutzerhandbuch*.
9. Sichern Sie das Betriebssystem und die Datenbank.
10. Fügen Sie ein Sicherheitsbanner mit einer aussagekräftigen Meldung hinzu. Beispiel: "Sie melden sich nun bei unserer PRODUKTIONSUMGEBUNG an! Fahren Sie nur fort, wenn Sie mit den Governance-Regeln für dieses System vertraut sind und die erforderlichen Schulungen absolviert haben." Weitere Informationen finden Sie unter "Konfigurieren eines Sicherheitsbanners" im *HP OO Central-Benutzerhandbuch*.
11. In der Windows- und SQL-Serverumgebung konfigurieren Sie HP OO für die Verwendung der Windows-Authentifizierung. Weitere Informationen finden Sie unter "Konfigurieren von HP OO für die Verwendung der Windows-Authentifizierung" im *HP OO-Datenbankhandbuch*.
12. Stellen Sie sicher, dass das Audit in Central aktiviert ist. Weitere Informationen finden Sie unter "Aktivieren des Audit" im *HP OO Central-Benutzerhandbuch*.

Server- und Clientauthentifizierung über Zertifikate

TLS-Zertifikate (Transport Layer Security) binden einen kryptografischen Schlüssel digital an die Details einer Organisation und ermöglichen so sichere und verschlüsselte Verbindungen zwischen einem Webserver und einem Browser.

HP OO verwendet das Dienstprogramm Keytool zur Verwaltung kryptografischer Schlüssel und vertrauenswürdiger Zertifikate. Dieses Dienstprogramm ist im Installationsordner von HP OO enthalten; Sie finden es unter **<Installationsverzeichnis>/java/bin/keytool**. Weitere Informationen zum Dienstprogramm Keytool finden Sie unter http://docs.oracle.com/javase/7/docs/technotes/tools/solaris/keytool.html.

Hinweis: Keytool ist ein Open Source-Dienstprogramm.

Installationen von HP OO Central enthalten zwei Dateien für die Verwaltung von Zertifikaten:

- **<Installationsverzeichnis>/central/var/security/client.truststore**: Enthält die Liste der vertrauenswürdigen Zertifikate.
- **<Installationsverzeichnis>/central/var/security/key.store**: Enthält das HP OO-Zertifikat (privater Schlüssel).

Es wird empfohlen, das selbstsignierte HP OO-Zertifikat im Anschluss an eine Neuinstallation von HP OO oder nach abgelaufener Gültigkeitsdauer Ihres aktuellen Zertifikats zu ersetzen.

Verschlüsseln der Kommunikation mit einem Serverzertifikat

Ersetzen des Central-TLS-Serverzertifikats

Sie können ein von einer bekannten Zertifizierungsstelle signiertes Zertifikat oder ein benutzerdefiniertes Serverzertifikat von einer lokalen Zertifizierungsstelle verwenden.

Ersetzen Sie die in <Gelb> hervorgehobenen Parameter, um sie auf den Speicherort der Datei **key.store** und andere Details auf Ihrem Computer abzustimmen.

Hinweis: Bei dem folgenden Verfahren wird das Dienstprogramm Keytool verwendet, das sich im Ordner **<Installationsverzeichnis>/java/bin/keytool** befindet.

1. Beenden Sie Central und sichern Sie die **key.store**-Originaldatei, die sich im Ordner **<Installationsverzeichnis>/central/var/security** befindet.
2. Öffnen Sie eine Befehlszeile im Ordner **<Installationsverzeichnis>/central/var/security**.
3. Löschen Sie das vorhandene Serverzertifikat aus der Datei **key.store**, indem Sie den folgenden Befehl eingeben:

   ```
   keytool -delete -alias tomcat -keystore key.store -storepass changeit
   ```

4. Wenn Sie bereits ein Zertifikat mit der Erweiterung **.pfx** oder **.p12** besitzen, fahren Sie mit dem nächsten Schritt fort. Sollte dies nicht der Fall sein, müssen Sie das Zertifikat mit dem privaten Schlüssel in das PKCS12-Format (.pfx, .p12) exportieren. Beispiel: Das Zertifikat liegt im Format PEM vor:

```
>openssl pkcs12 –export –in <cert.pem> -inkey <.key> -out <certificate name>.p12 –name <name>
```

Wenn das Zertifikat im Format DER vorliegt, fügen Sie den Parameter `-inform DER` hinter **pkcs12** an. Beispiel:

```
>openssl pkcs12 –inform DER –export –in <cert.pem> -inkey <.key> -out <certificate name>.p12 –name <name>
```

Hinweis: Notieren Sie sich das Kennwort, das Sie angeben. Sie benötigen dieses Kennwort für den privaten Schlüssel bei der späteren Eingabe der Passphrase für den Keystore.

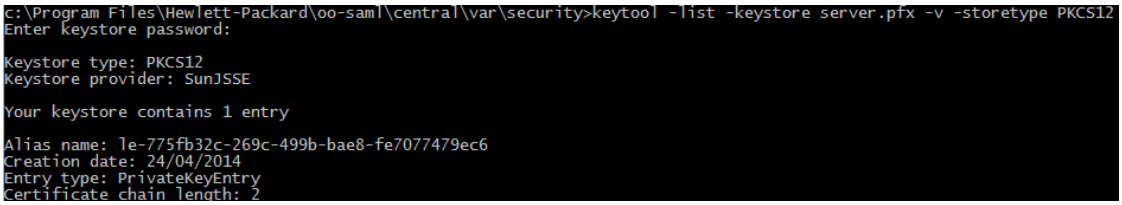
Stellen Sie sicher, dass Sie ein sicheres Kennwort wählen.

5. Listen Sie mit dem folgenden Befehl den Alias für den Alias Ihres Zertifikats auf:

   ```
   keytool -list -keystore <certificate_name> -v -storetype PKCS12
   ```

   Der Alias des Zertifikats wird angezeigt und sollte im nächsten Befehls angegeben werden.

   Im folgenden Beispiel ist dies die vierte Zeile von unten.

6. Importieren Sie mit dem folgenden Befehl das Serverzertifikat im PKCS12-Format in die Central-Datei **key.store**:

   ```
   keytool -importkeystore -srckeystore <PKCS12 format certificate path> -destkeystore key.store -srcstoretype pkcs12 -deststoretype JKS -alias <cert alias> -destalias tomcat
   ```

7. Wenn das importierte Serverzertifikat ein anderes Kennwort besitzt als das ursprüngliche Serverzertifikat, ist es wichtig, das in keyPass angegebene Kennwort zu ändern. Folgen Sie den Anweisungen unter "Ändern des Kennworts für den KeyStore/TrustStore" auf Seite 11.

   Es wird empfohlen, das Standardkennwort “changeit” im automatisch generierten Keystore des Central-Servers zu ändern. Weitere Informationen finden Sie unter "Ändern des Kennworts für den KeyStore/TrustStore" auf Seite 11.

8. Starten Sie Central.

Importieren eines CA-Stammzertifikats in einen RAS-TrustStore

Wenn Sie ein benutzerdefiniertes Stammzertifikat für Central verwenden und dieses Stammzertifikat bei der RAS-Installation nicht angegeben haben, müssen Sie nach der Installation eines RAS die vertrauenswürdige Stammzertifizierungsstelle (CA) in die RAS-Datei **client.truststore** importieren. Wenn Sie eine bekannte Stamm-CA (wie Verisign) verwenden, müssen Sie das folgende Verfahren nicht durchführen, da das Zertifikat bereits in der Datei **Client.truststore** vorhanden ist.

Standardmäßig unterstützt HP OO alle selbstsignierten Zertifikate. Allerdings ist es in einer Produktionsumgebung ratsam, diese Standardeinstellung aus Sicherheitsgründen in eine benutzerdefinierte CA oder eine bekannte CA zu ändern.

Ersetzen Sie die in <Gelb> hervorgehobenen Parameter.

Hinweis: Bei dem folgenden Verfahren wird das Dienstprogramm Keytool verwendet, das sich im Ordner **<Installationsverzeichnis>/java/bin/keytool** befindet.

1. Beenden Sie RAS und sichern Sie die **client.truststore**-Originaldatei, die sich im Ordner **<Installationsverzeichnis>/ras/var/security** befindet.
2. Öffnen Sie eine Befehlszeile im Ordner **<Installationsverzeichnis>/ras/var/security**.
3. Öffnen Sie die Datei **<Installationsverzeichnis>ras/conf/ras-wrapper.conf** und legen Sie für `-Dssl.support-self-signed`den Wert **false** fest. Dies aktiviert die vertrauenswürdige Zertifizierungsstelle (CA).

   Beispiel:

   ```
   wrapper.java.additional.<x>=-Dssl.support-self-signed=false
   ```

4. Öffnen Sie die Datei **Installationsverzeichnisras/conf/ras-wrapper.conf** und legen Sie für `-Dssl.verifyHostName` den Wert **true** fest. Hiermit wird geprüft, ob der FQDN im Zertifikat mit dem FQDN der Anforderung übereinstimmt.

   Beispiel:

   ```
   wrapper.java.additional.<x>=-Dssl.verifyHostName=true
   ```

5. Importieren Sie die vertrauenswürdige Stammzertifizierungsstelle (CA) in die RAS-Datei **Client.truststore**, wenn sie noch nicht in der CA-Liste vorhanden ist (dort befinden sich standardmäßig alle bekannten CAs):

   ```
   keytool -importcert -alias <any_alias> -keystore client.truststore -file <certificate_name.cer> -storepass <changeit>
   ```

6. Starten Sie RAS.

Importieren eines CA-Stammzertifikats in den OOSH-TrustStore

Wenn Sie ein benutzerdefiniertes Stammzertifikat für Central verwenden, müssen Sie die vertrauenswürdige Stammzertifizierungsstelle (CA) in die OOSH-Datei **client.truststore** importieren. Wenn Sie eine bekannte Stamm-CA (wie Verisign) verwenden, müssen Sie das folgende Verfahren nicht durchführen, da das Zertifikat bereits in der Datei **Client.truststore** vorhanden ist.

Standardmäßig unterstützt HP OO alle selbstsignierten Zertifikate. Allerdings ist es in einer Produktionsumgebung ratsam, diese Standardeinstellung aus Sicherheitsgründen in eine benutzerdefinierte CA oder eine bekannte CA zu ändern.

Ersetzen Sie die in <Gelb> hervorgehobenen Parameter.

Hinweis: Bei dem folgenden Verfahren wird das Dienstprogramm Keytool verwendet, das sich im Ordner **<Installationsverzeichnis>/java/bin/keytool** befindet.

1. Beenden Sie Central und sichern Sie die **client.truststore**-Originaldatei, die sich im Ordner **<Installationsverzeichnis>/central/var/security** befindet.
2. Bearbeiten Sie die Datei **oosh.bat** im Ordner **<Installationsverzeichnis>/central/bin**.
3. Legen Sie für `-Dssl.support-self-signed` den Wert **false** fest. Dies aktiviert die vertrauenswürdige Zertifizierungsstelle (CA).

   Beispiel:

   ```
   -Dssl.support-self-signed=false
   ```

4. Legen Sie für `-Dssl.verifyHostName` den Wert **true** fest. Hiermit wird geprüft, ob der FQDN im Zertifikat mit dem FQDN der Anforderung übereinstimmt.

   Beispiel:

   ```
   -Dssl.verifyHostName=true
   ```

5. Importieren Sie die vertrauenswürdige Stammzertifizierungsstelle (CA) in die Central-Datei **Client.truststore**, wenn sie noch nicht in der CA-Liste vorhanden ist (dort befinden sich standardmäßig alle bekannten CAs):

   ```
   keytool -importcert -alias <any_alias> -keystore client.truststore -file
   <certificate_name.cer> -storepass <changeit>
   ```

6. Führen Sie OOSH aus.
7. Starten Sie Central.

Importieren eines CA-Stammzertifikats in den Studio Debugger-TrustStore

Wenn Sie ein benutzerdefiniertes Stammzertifikat für Studio verwenden, müssen Sie die vertrauenswürdige Stammzertifizierungsstelle (CA) nach der Installation von Studio in die Studio-Datei **client.truststore** importieren. Wenn Sie eine bekannte Stamm-CA (wie Verisign) verwenden, müssen Sie das folgende Verfahren nicht durchführen, da das Zertifikat bereits in der Datei **Client.truststore** vorhanden ist.

Standardmäßig unterstützt HP OO alle selbstsignierten Zertifikate. Allerdings ist es in einer Produktionsumgebung ratsam, diese Standardeinstellung aus Sicherheitsgründen in eine benutzerdefinierte CA oder eine bekannte CA zu ändern.

Ersetzen Sie die in <Gelb> hervorgehobenen Parameter.

Hinweis: Bei dem folgenden Verfahren wird das Dienstprogramm Keytool verwendet, das sich im Ordner **<Installationsverzeichnis>/java/bin/keytool** befindet.

1. Beenden Sie Studio und sichern Sie die **client.truststore**-Originaldatei, die sich im Ordner **<Installationsverzeichnis>/studio/var/security** befindet.
2. Bearbeiten Sie die Datei **Studio.l4j.ini** im Ordner **<Installationsverzeichnis>/studio**.
3. Legen Sie für `-Dssl.support-self-signed` den Wert **false** fest. Dies aktiviert die vertrauenswürdige Zertifizierungsstelle (CA).

   Beispiel:

```
-Dssl.support-self-signed=false
```

4. Legen Sie für `-Dssl.verifyHostName` den Wert **true** fest. Hiermit wird geprüft, ob der FQDN im Zertifikat mit dem FQDN der Anforderung übereinstimmt.

   Beispiel:

   ```
   -Dssl.verifyHostName=true
   ```

5. Importieren Sie die vertrauenswürdige Stammzertifizierungsstelle (CA) in die Studio-Datei **Client.truststore**, wenn sie noch nicht in der CA-Liste vorhanden ist (dort befinden sich standardmäßig alle bekannten CAs):

   ```
   keytool -importcert -alias <any_alias> -keystore client.truststore -file <certificate_name.cer> -storepass <changeit>
   ```

6. Starten Sie Studio.

Weitere Informationen finden Sie unter "Debuggen einer Remote-Instanz von Central mit Studio" im *Studio-Erstellungshandbuch*.

Ändern des Kennworts für den KeyStore/TrustStore

Ändern der Kennwörter für KeyStore, Truststore und Serverzertifikat in der Central-Konfiguration

1. Stellen Sie sicher, dass Central ausgeführt wird.

   Hinweis: Stellen Sie vor diesem Schritt sicher, dass verschlüsselte Kennwörter vorhanden sind. Weitere Informationen zum Verschlüsseln eines Kennworts finden Sie unter "Encrypting Passwords" im *HP OO Administration Guide*.

   Führen Sie in OOSH den folgenden Befehl aus:

   ```
   set-sys-config --key <keyName> --value <ecryptedPassword>
   ```

   Dabei ist `<keyName>` einer der Werte aus der folgenden Tabelle:

Konfigurationselement	Aktion
`key.store.password`	Zum Festlegen des Kennworts für den Zugriff auf **key.store**. Der Standardwert ist `"changeit"`. Dieser Wert muss dem für `keystorePass` in den nachfolgenden Schritten festgelegten Wert entsprechen.
`key.store.private.key.alias.password`	Zum Festlegen des Kennworts, das für den Zugriff auf das Serverzertifikat (privater Schlüssel) in **key.store** verwendet wird. Der Standardwert ist `"changeit"`.

	Dieser Wert muss dem für `keyPass` in den nachfolgenden Schritten festgelegten Wert entsprechen.

2. Beenden Sie den Central-Dienst.
3. Ändern Sie mit Keytool die Kennwörter für KeyStore, TrustStore und Serverzertifikat.
4. Ändern Sie die Kennwörter auch in der Datei **Server.xml**, die sich im Ordner **<Installationsverzeichnis>/central/tomcat/conf** befindet.
   a. Suchen Sie den HTTPS-Connector. Beispiel:

   ```
   keyPass="changeit" keystoreFile="C:/Program Files/Hewlett-Packard/HP
   Operations Orchestration/central/var/security/key.store"
   keystorePass="changeit" keystoreType="JKS" maxThreads="200" port="8443"
   protocol="org.apache.coyote.http11.Http11NioProtocol" scheme="https"
   secure="true" sslProtocol="TLSv1.2"
   sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2" truststoreFile="C:/Program
   Files/Hewlett-Packard/HP Operations
   Orchestration/central/var/security/client.truststore"
   truststorePass="changeit" truststoreType="JKS"/>
   ```

   b. Ändern Sie das erforderliche Kennwort.
      - `keyPass` – Das Kennwort für den Zugriff auf den privaten Schlüssel des Serverzertifikats in der angegebenen Keystore-Datei. Der Standardwert ist "`changeit`".
      - `keystorePass` – Das Kennwort für den Zugriff auf die angegebene Keystore-Datei. Der Standardwert ist der Wert des Attributs `keyPass`.

        Hinweis: Es wird empfohlen, nicht das in **keyPass** angegebene Kennwort sondern ein anderes sicheres Kennwort zu verwenden.

      - `truststorePass` - Das Kennwort für den Zugriff auf den Truststore (der alle vertrauenswürdigen CAs enthält). Der Standardwert ist der Wert der Systemeigenschaft **javax.net.ssl.trustStorePassword**. Wenn diese Eigenschaft null ist, wird kein TrustStore-Kennwort konfiguriert. Wird ein ungültiges TrustStore-Kennwort angegeben, wird eine Warnung protokolliert und ein Versuch unternommen, ohne Kennwort auf den TrustStore zuzugreifen; dabei wird die Überprüfung des TrustStore-Inhalts übersprungen.

   c. Speichern Sie die Datei.
5. Bearbeiten Sie die Datei **Central-wrapper.conf**, die sich im Ordner **<Installationsverzeichnis>\central\conf** befindet, und ändern Sie:

   ```
   wrapper.java.additional.<x>=-Djavax.net.ssl.trustStorePassword=changeit
   ```

6. Starten Sie den Central-Dienst.

Ändern der TrustStore-Kennwörter für RAS, OOSH und Studio

Hinweis: Bevor Sie die folgenden Schritte ausführen, sollten Sie mit Keytool die Kennwörter für KeyStore, TrustStore und Serverzertifikat ändern.

- **So ändern Sie das TrustStore-Kennwort für eine eigenständige RAS-Instanz**: Bearbeiten Sie die Datei **Ras-wrapper.conf** und ändern Sie den Parameter `changeit` des TrustStore.
- **So ändern Sie das TrustStore-Kennwort für OOSH**: Bearbeiten Sie die Datei **Oosh.bat** und ändern Sie den Parameter `changeit` des TrustStore.
- **So ändern Sie das TrustStore-Kennwort für Studio**: Bearbeiten Sie die Datei **<Installationsverzeichnis>/studio/Studio.l4j.ini** und ersetzen Sie den Parameter `changeit` des TrustStore durch das neue Kennwort in verschlüsselter Form.

  Weitere Informationen zum Verschlüsseln eines Kennworts finden Sie unter "Encrypting Passwords" im *HP OO Administration Guide*.

Verschlüsseln der Kennwörter für Studio-KeyStore und -TrustStore

Ab HP 10.20 werden die Kennwörter von KeyStore und TrustStore aus Studio verschlüsselt. Nach einem Upgrade von 10.10 auf 10.20 werden diese Kennwörter nur dann verschlüsselt, wenn sie in der Datei **<Installationsverzeichnis>/studio/Studio.l4j.ini** unverändert geblieben sind. Alle anderen Kennwörter, die in früheren Versionen geändert wurden, werden beim Upgrade nicht automatisch verschlüsselt.

Wenn Sie das TrustStore- oder das KeyStore-Kennwort ändern möchten, können Sie dies in der Datei **Studio.l4j.ini** in verschlüsselter Form oder in Klartext vornehmen. Nach dem Ändern der Kennwörter müssen Sie sie manuell verschlüsseln, um sicherzustellen, dass sie nicht im Klartext im Task-Manager für den Studio-Prozess angezeigt werden:

1. Schließen Sie Studio.
2. Suchen Sie das Skript **encrypt-password** in **<Installationsordner>/central/bin**.
3. Verwenden Sie den folgenden Befehl, um das benutzerdefinierte Kennwort zu verschlüsseln:

   ```
   encrypt-password.bat --obfuscate <your password>
   ```

4. Ändern Sie die Kennwörter in der Datei **<Installationsverzeichnis>/studio/Studio.l4j.ini** für die folgenden Parameter:

   ```
   -Djavax.net.ssl.keyStorePassword={OBFUSCATED}<obfuscated_password>

   -Djavax.net.ssl.trustStorePassword={OBFUSCATED}<obfuscated_password>
   ```

5. Ändern Sie das KeyStore- und das TrustStore-Kennwort aus Studio mit dem keytool im Ordner **<Installationsverzeichnis>/studio/var/security/**.

   Hinweis: Wenn das Clientzertifikat für Studio Remote Debugger nicht konfiguriert wurde, wird das Pfadargument keyStore ignoriert.

6. Starten Sie Studio.

Wichtig! Löschen Sie nach der Verwendung des Skripts **encrypt-password** die Befehlshistorie.

Dies ist notwendig, da bei einem Linux-Betriebssystem der Kennwortparameter in Klartext unter **/$USER/.bash_history** gespeichert wird und über den Befehl `history` zugänglich ist.

Entfernen der RC4-Verschlüsselung aus den unterstützten SSL-Verschlüsselungsverfahren

Der Remotehost unterstützt die Verwendung der RC4-Verschlüsselung. Diese Verschlüsselung ist bei der Generierung eines pseudozufälligen Bytestroms fehlerhaft, sodass eine Vielzahl kleiner Verzerrungen in den Strom gelangt und die Zufälligkeit der Daten reduziert.

Wenn einfacher Text wiederholt verschlüsselt wird (Beispiel: HTTP-Cookies) und ein Angreifer imstande ist, viele (im zweistelligen Millionenbereich) verschlüsselte Texte in die Hände zu bekommen, kann er den Text möglicherweise entschlüsseln.

Deaktivieren Sie die RC4-Verschlüsselung auf der JRE-Ebene (beginnend mit Java 7):

1. Öffnen Sie die Datei **$JRE_HOME/lib/security/java.security**.
2. Deaktivieren Sie die RC4-Verschlüsselung, indem Sie die Kommentare entfernen und die Parameter entsprechend dem folgenden Beispiel ändern:

   ```
   jdk.certpath.disabledAlgorithms=RC4, MD2, RSA keySize < 1024

   jdk.tls.disabledAlgorithms=RC4, MD5, DSA, RSA keySize < 1024
   ```

3. Starten Sie den HP OO Central-Server neu.

Weitere Informationen finden Sie unter http://stackoverflow.com/questions/18589761/restict-cipher-suites-on-jre-level.

Hinweis: Nach einem Upgrade von einer früheren Version von HP OO 10.x müssen Sie diese Schritte wiederholen.

Ändern oder Deaktivieren der HTTP/HTTPS-Ports

Die Datei **Server.xml** im Verzeichnis **[OO_HOME]/central/tomcat/conf** enthält zwei **<Connector>**-Elemente unter dem Element **<Service>**. Diese Connector definieren oder aktivieren die Ports, die der Server überwacht.

Jede Connector-Konfiguration wird anhand von zugehörigen Attributen definiert. Der erste Connector definiert einen Standard-HTTP- und der zweite Connector einen HTTPS-Connector.

Standardmäßig sehen diese Connectoren wie folgt aus:

HTTP-Connector:

```
<Connector URIEncoding="UTF-8" compression="on" connectionTimeout="20000"
port="8080" protocol="org.apache.coyote.http11.Http11NioProtocol"
redirectPort="8443"/>
```

HTTPS-Connector:

```
<Connector SSLEnabled="true" URIEncoding="UTF-8" clientAuth="false"
compression="on" keyAlias="tomcat" keyPass="changeit" keystoreFile="C:/Program
Files/Hewlett-Packard/HP Operations
Orchestration/central/var/security/key.store" keystorePass="changeit"
keystoreType="JKS" maxThreads="200" port="8443"
protocol="org.apache.coyote.http11.Http11NioProtocol" scheme="https"
secure="true" sslProtocol="TLSv1.2" sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2"
truststoreFile="C:/Program Files/Hewlett-Packard/HP Operations
Orchestration/central/var/security/client.truststore" truststorePass="changeit"
truststoreType="JKS"/>
```

Standardmäßig sind beide Connectoren aktiviert.

Wichtig! Wenn Sie einen der Central-Ports in der Datei **Server.xml** ändern oder deaktivieren, müssen Sie auch die Datei **Central-wrapper.conf** und jede **RAS-wrapper.conf**-Datei so ändern, dass sie auf die Central-URL mit dem aktualisierten Port verweist. Andernfalls schlagen alle Ihre Flows, die in Central ausgeführt werden, fehl. Überprüfen Sie auch die Load Balancer-Konfigurationen.

Ändern der Portwerte

So ändern Sie die Werte eines Ports:

1. Bearbeiten Sie die Datei **server.xml**, die sich im Ordner **<Installationsverzeichnis>/central/tomcat/conf** befindet.
2. Suchen Sie die Zeile mit dem HTTP- oder HTTPS-Connector und ändern Sie den Wert für **Port**.

   Hinweis: Wenn Sie sowohl HTTP und HTTPS aktiv halten und den HTTPS-Port ändern möchten, müssen Sie den Wert von **redirectPort** für den HTTP-Connector und den Wert von **port** für den HTTPS-Connector ändern.
3. Speichern Sie die Datei.
4. Starten Sie Central erneut.

Deaktivieren eines Ports

Aus Sicherheitsgründen könnten Sie zum Beispiel den HTTP-Port deaktivieren, sodass der einzige Kommunikationskanal TLS verwendet und verschlüsselt wird.

So deaktivieren Sie einen der Ports:

1. Bearbeiten Sie die Datei **server.xml**, die sich im Ordner **<Installationsverzeichnis>/central/tomcat/conf** befindet.
2. Suchen Sie den HTTP- oder HTTPS-Connector und löschen Sie die Zeile oder kommentieren Sie sie aus.
3. Importieren Sie die vertrauenswürdige Stammzertifizierungsstelle (CA) in die Central-Datei **Client.truststore**, wenn sie noch nicht in der CA-Liste vorhanden ist:

```
keytool -importcert -alias <any_alias> -keystore client.truststore -file
<certificate_name.cer> -storepass <changeit>
```

Hinweis: Wenn Sie eine bekannte Stamm-CA (wie Verisign) verwenden, müssen Sie diesen Schritt nicht durchführen, da das Zertifikat bereits in der Datei **Client.truststore** vorhanden ist.

4. Speichern Sie die Datei.
5. Starten Sie Central erneut.

Hinweis: Es ist auch möglich, den HTTP-Port während der Installation zu deaktivieren.

Fehlerbehebung

Wenn der Server nicht startet, öffnen Sie die Datei **wrapper.log** und suchen nach einem Fehler in `ProtocolHandler ["http-nio-8443"]`.

Dieser Fehler kann beim Initialisieren von Tomcat oder beim Starten des Connectors auftreten. Er tritt in vielen Variationen auf, aber die Fehlermeldung enthält weitere Informationen.

Alle HTTPS-Connector-Parameter sind in der Tomcat-Konfigurationsdatei angegeben, die sich unter **C:\HP\oo\central\tomcat\conf\Server.xml** befindet.

Öffnen Sie die Datei und scrollen Sie nach unten, bis Sie den HTTPS-Connector sehen:

```
<Connector SSLEnabled="true" clientAuth="false" keyAlias="tomcat"
keystoreFile="C:/HP/oo/central/var/security/keystore.p12" keystorePass="tomcat-
keystore-password" keystoreType="PKCS12" maxThreads="200" port="8443"
protocol="org.apache.coyote.http11.Http11NioProtocol" scheme="https" secure="true"
sslProtocol="TLSv1.2" sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2"/>
```

Prüfen Sie, ob eine Nichtübereinstimmung bei den Parametern vorliegt, indem Sie sie mit den in den vorherigen Schritten eingegebenen Parametern vergleichen.

Clientzertifikatauthentifizierung (Gegenseitige Authentifizierung)

Die X.509-Zertifikat-Authentifizierung wird am häufigsten beim Überprüfen der Identität eines Servers bei Verwendung von TLS genutzt; meist sind dies HTTPS-Verbindungen eines Browsers. Der Browser überprüft automatisch, ob das von einem Server vorgelegte Zertifikat von einer vertrauenswürdigen Zertifizierungsstelle ausgegeben wurde, die sich in einer von ihm verwalteten Liste befindet.

Sie können TLS aber auch für eine gegenseitige Authentifizierung verwenden. Der Server fordert als Teil des TLS-Handshake ein gültiges Zertifikat vom Client an. Der Server authentifiziert den Client, indem er prüft, ob das Zertifikat von einer vertrauenswürdigen Authentifizierungsstelle signiert wurde. Wenn ein gültiges Zertifikat bereitgestellt wurde, können Sie es über die Servlet-API in einer Anwendung abrufen.

Konfigurieren der Clientzertifikatauthentifizierung in Central

Stellen Sie vor dem Konfigurieren der Clientzertifikatauthentifizierung in Central sicher, dass Sie das TLS-Serverzertifikat wie in "Server- und Clientauthentifizierung über Zertifikate" auf Seite 6 beschrieben konfiguriert haben.

Legen Sie für das Attribut `clientAuth` den Wert `true` fest, wenn der TLS-Stack eine gültige Zertifikatskette vom Client anfordern soll, bevor eine Verbindung akzeptiert wird. Geben Sie `want` an, um festzulegen, dass der TLS-Stack ein Clientzertifikat anfordert, aber nicht fehlschlägt, wenn kein Zertifikat vorgelegt wird. Wird `false` (Standard) angegeben, ist keine Zertifikatskette erforderlich, es sei denn der Client fordert eine Ressource an, die durch eine Sicherheitseinschränkung geschützt ist, die auf einer Clientzertifikatauthentifizierung beruht. (Weitere Informationen finden Sie in der Apache Tomcat Configuration Reference.)

Geben Sie die Datei mit der **Zertifikatsperrliste (CRL)** an. Die Datei kann mehrere CRLs enthalten. Bei einigen kryptografischen Systemen, in der Regel Public-Key-Infrastrukturen (PKIs), werden in einer Zertifikatsperrliste Zertifikate (genauer gesagt Seriennummern von Zertifikaten) erfasst, die widerrufen wurden. Entitäten, die solche (widerrufene) Zertifikate vorlegen, sollten als nicht mehr vertrauenswürdig betrachtet werden.

Hinweis: Bei dem folgenden Verfahren wird das Dienstprogramm Keytool verwendet, das sich im Ordner **<Installationsverzeichnis>/java/bin/keytool** befindet.

1. Beenden Sie den Central-Server.
2. Importieren Sie das zugehörige Stammzertifikat (CA) in Central **client.truststore**: **<Installationsverzeichnis>/central/var/security/client.truststore**, wenn es noch nicht in der CA-Liste vorhanden ist (dort befinden sich standardmäßig alle bekannten CAs). Beispiel:

   ```
   keytool -importcert -alias <any_alias> -keystore <path>/client.truststore -file <certificate_path> -storepass <changeit>
   ```

3. Bearbeiten Sie die Datei **server.xml**, die sich im Ordner **<Installationsverzeichnis>/central/tomcat/conf** befindet.
4. Legen Sie für das Attribut `clientAuth` im Tag `Connector` den Wert `want` oder `true` fest. Die Standardeinstellung ist `false`.

   Hinweis: An diesem Punkt kann der Server gestartet werden. Es wird aber empfohlen, den Server erst am Ende dieser Prozedur zu starten.

5. (Optional) Fügen Sie das Attribut `crlFile` hinzu, um die Datei mit den Zertifikatsperrlisten für die TLS-Zertifikatprüfung zu definieren. Beispiel:

   ```
   crlFile="<path>/crlname.<crl/pem>"
   ```

   Die Datei kann die Erweiterung `.crl` für eine einzelne Zertifikatsperrliste oder `.pem` (PEM CRL-Format) für eine oder mehrere Zertifikatsperrlisten aufweisen. Das PEM-CRL-Format verwendet die folgenden Kopf- und Fußzeilen:

```
-----BEGIN X509 CRL-----
-----END X509 CRL-----
```

Beispiel für die `.pem`-Dateistruktur mit einer CRL (mehrere CRLs werden mit weiteren CRL-Blöcken verkettet):

```
-----BEGIN X509 CRL-----
MIIBbzCB2QIBATANBgkqhkiG9w0BAQUFADBeMQswCQYDVQQGEwJVUzEYMBYGA1UE
ChMPVS5TLiBHb3Zlcm5tZW50MQwwCgYDVQQLEwNEb0QxEDAOBgNVBAsTB1Rlc3Rp
bmcxFTATBgNVBAMTDFRydXN0IEFuY2hvchcNOTkwMTAxMTIwMTAwWhcNNDgwMTAx
MTIwMTAwWjAiMCACAScXDTk5MDEwMTEyMDAwMFowDDAKBgNVHRUEAwoBAaAjMCEw
CgYDVR0UBAMCAQEwEwYDVR0jBAwwCoAIq5rr+cLnVI8wDQYJKoZIhvcNAQEFBQAD
gYEAC7lqZwejJRW7QvzH11/7cYcL3racgMxH3PSU/ufvyLk7ahR++RtHary/WeCv
RdyznLiIOA8ZBiguWtVPqsNysNn7WLofQIVa+/TD3T+lece4e1NwGQvj5Q+e2wRt
GXg+gCuTjTKUFfKRnWz7O7RyiJKKim0jtAF4RkCpLebNChY=
-----END X509 CRL-----
```

6. Starten Sie den Central-Server.

Hinweis: Für jedes Clientzertifikat müssen Sie entweder einen internen Benutzer oder einen LDAP-Benutzer definieren. Der Name des Benutzers sollte in den Zertifikatattributen definiert sein. Der Standardwert ist der Wert des CN-Attributs. Weitere Informationen finden Sie im Abschnitt Verarbeiten eines Zertifikatprinzipals.

Beachten Sie Folgendes: Auch wenn Sie in HP OO mehrere LDAP-Konfigurationen eingerichtet haben, kann ein Benutzer nur mit den Clientzertifikatattributen aus dem Standard-LDAP authentifiziert werden. Central versucht zuerst, den Benutzer mit dem Standard-LDAP zu authentifizieren, und unternimmt, wenn dies fehlschlägt, einen weiteren Authentifizierungsversuch in der internen HP OO-Domäne.

Aktualisieren der Konfiguration eines Clientzertifikats in RAS

Das Clientzertifikat wird bei der Installation des RAS konfiguriert. Wenn Sie das Clientzertifikat jedoch aktualisieren müssen, können Sie die Datei **ras-wrapper.conf** manuell bearbeiten.

Voraussetzung: Sie müssen das CA-Stammzertifikat von Central in den RAS-TrustStore importieren. Weitere Informationen finden Sie unter "Importieren eines CA-Stammzertifikats in einen RAS-TrustStore" auf Seite 8.

So aktualisieren Sie die Konfiguration des Clientzertifikats in einem externen RAS:

1. Beenden Sie den RAS-Server.
2. Öffnen Sie die Datei **Ras-wrapper.conf** im Ordner **<Installationsverzeichnis>/ras/conf**.
3. Ändern Sie die folgenden Angaben gemäß Ihrem Clientzertifikat:

```
wrapper.java.additional.<x>=-Djavax.net.ssl.keyStore=<installation
dir>/var/security/certificate.p12"

wrapper.java.additional.<x>=-Djavax.net.ssl.keyStorePassword=changeit
```

```
wrapper.java.additional.<x>=-Djavax.net.ssl.keyStoreType=PKCS12
```

4. Starten Sie den RAS-Server.

Wichtiger Hinweis! Das X.509-Clientzertifikat muss den Prinzipalnamen des RAS, die RAS-ID, enthalten (siehe Verarbeiten eines Zertifikatprinzipals).

Sie finden die RAS-ID auf der Registerkarte **Topologie** in Central. Weitere Informationen finden Sie unter "Einrichten der Topologie – Worker" im *HP OO Central-Benutzerhandbuch*.

Konfigurieren eines ClientZertifikats in Studio Remote Debugger

Voraussetzung: Sie müssen das CA-Stammzertifikat von Central in den Studio Debugger-TrustStore importieren. Weitere Informationen finden Sie unter "Importieren eines CA-Stammzertifikats in den Studio Debugger-TrustStore" auf Seite 10.

So konfigurieren Sie das Clientzertifikat in Studio Remote Debugger:

1. Schließen Sie Studio.
2. Bearbeiten Sie die Datei **Studio.l4j.ini** im Ordner **<Installationsverzeichnis>/studio**.
3. Ändern Sie die folgenden Angaben gemäß Ihrem Clientzertifikat:

```
-Djavax.net.ssl.keyStore="<installation
dir>/studio/var/security/certificate.p12"

-Djavax.net.ssl.keyStorePassword=changeit

-Djavax.net.ssl.keyStoreType=PKCS12
```

4. Starten Sie Studio.

Hinweise:

- Ab HP OO 10.20 wird der Parameter `keyStorePassword` in **Studio.l4j.ini** standardmäßig verschlüsselt, wenn das Kennwort als Standard beibehalten wurde. Diesen Parameter können Sie ändern und entweder in Klartext oder verschlüsselt speichern.
- Für jedes Clientzertifikat müssen Sie entweder einen internen Benutzer oder einen LDAP-Benutzer definieren. Der Name des Benutzers sollte in den Zertifikatattributen definiert sein. Der Standardwert ist der Wert des CN-Attributs. Weitere Informationen finden Sie im Abschnitt Verarbeiten eines Zertifikatprinzipals.
- Beachten Sie Folgendes: Auch wenn Sie in HP OO mehrere LDAP-Konfigurationen eingerichtet haben, kann ein Benutzer nur mit den Clientzertifikatattributen aus dem Standard-LDAP authentifiziert werden. Central versucht zuerst, den Benutzer mit dem Standard-LDAP zu authentifizieren, und unternimmt, wenn dies fehlschlägt, einen weiteren Authentifizierungsversuch in der internen HP OO-Domäne.

Konfigurieren eines Clientzertifikats in OOSH

Voraussetzung: Sie müssen das CA-Stammzertifikat von Central in den OOSH-TrustStore importieren. Weitere Informationen finden Sie unter "Importieren eines CA-Stammzertifikats in den OOSH-TrustStore" auf Seite 9.

1. Beenden Sie OOSH.
2. Bearbeiten Sie die Datei **oosh.bat** im Ordner **<Installationsverzeichnis>/central/bin**.
3. Ändern Sie die folgenden Angaben gemäß Ihrem Clientzertifikat:

```
-Djavax.net.ssl.keyStore="<installation dir>/var/security/certificate.p12"

-Djavax.net.ssl.keyStorePassword=changeit

-Djavax.net.ssl.keyStoreType=PKCS12
```

4. Starten Sie OOSH.

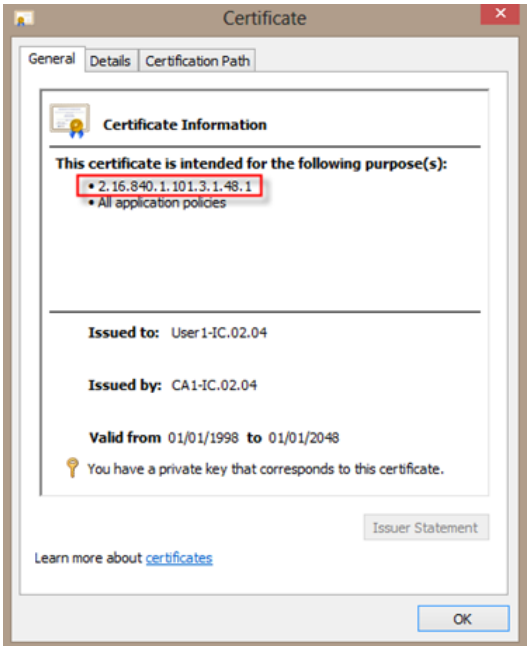
Hinweis: Für jedes Clientzertifikat müssen Sie entweder einen internen Benutzer oder einen LDAP-Benutzer definieren. Der Name des Benutzers sollte in den Zertifikatattributen definiert sein. Der Standardwert ist der Wert des CN-Attributs. Weitere Informationen finden Sie im Abschnitt Verarbeiten eines Zertifikatprinzipals.

Beachten Sie Folgendes: Auch wenn Sie in HP OO mehrere LDAP-Konfigurationen eingerichtet haben, kann ein Benutzer nur mit den Clientzertifikatattributen aus dem Standard-LDAP authentifiziert werden. Central versucht zuerst, den Benutzer mit dem Standard-LDAP zu authentifizieren, und unternimmt, wenn dies fehlschlägt, einen weiteren Authentifizierungsversuch in der internen HP OO-Domäne.

Verarbeiten der Zertifikatrichtlinien

HP OO obliegt die Verarbeitung von Zertifikatrichtlinien für das Endpunktzertifikat.

- Sie können die Zweckzeichenfolge im Zertifikat festlegen.
- In HP OO können Sie die Richtlinienzeichenfolge(n) als Konfigurationselement hinzufügen und die Richtlinienzeichenfolge eines jeden Endpunktzertifikats überprüfen. Wenn es nicht übereinstimmt, wird das Zertifikat abgelehnt.
- Aktivieren oder deaktivieren Sie die Überprüfung der Zertifikatrichtlinien, indem Sie das folgende Konfigurationselement hinzufügen: `x509.certificate.policy.enabled=true/false` (Standardeinstellung ist `false`).
- Definieren Sie die Richtlinienliste, indem Sie das folgende Konfigurationselement hinzufügen: `x509.certificate.policy.list=<comma_separated_list>` (Standardeinstellung ist eine leere Liste).



Weitere Informationen zum Ändern der HP OO-Systemeigenschaften finden Sie im *HP OO Shell Guide*.

Verarbeiten eines Zertifikatprinzipals

Sie können definieren, wie der Prinzipal aus einem Zertifikat abgerufen wird, indem Sie einen regulären Ausdruck als Vergleichskriterium für `Subject` angeben. Der reguläre Ausdruck sollte eine einzelne Gruppe enthalten. Der Standardausdruck `CN=(.?)` zieht für den Vergleich das Feld "Allgemeiner Name" (Common Name, CN) heran. Beispiel: `CN=Jimi Hendrix, OU=` weist den Benutzernamen `Jimi Hendrix` zu.

- Groß- und Kleinschreibung wird ignoriert.
- Der Prinzipal des Zertifikats ist der Benutzername in HP OO (LDAP- oder interner Benutzer).
- Um den regulären Ausdruck zu ändern, ändern Sie das Konfigurationselement: `x509.subject.principal.regex`.

Weitere Informationen zum Ändern der HP OO-Systemeigenschaften finden Sie im *HP OO Shell User Guide*.

Konfigurieren der FIPS 140-2-Konformität Stufe 1 in HP OO

In diesem Abschnitt wird erläutert, wie Sie HP Operations Orchestration konfigurieren, um Übereinstimmung mit den Federal Information Processing Standards (FIPS) 140-2 Stufe 1 zu erzielen.

FIPS 140-2 ist ein Standard, der Sicherheitsanforderungen für kryptografische Module definiert und von der US-Behörde National Institute of Standards Technology (NIST) festgelegt wurde. Der Standard wurde veröffentlicht unter: csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf.

Nachdem Sie die HP OO-Konfiguration an den FIPS 140-2-Standard angepasst haben, verwendet HP OO die folgenden Sicherheitsalgorithmen:

- Symmetrischer Schlüsselalgorithmus: AES256
- Hash-Algorithmus: SHA256

HP OO verwendet den Sicherheitsanbieter RSA BSAFE Crypto Software Version 6.1. Dies ist der einzige unterstützte Sicherheitsanbieter für FIPS 140-2.

Hinweis: Nachdem Sie die HP OO-Konfiguration an den FIPS 140-2-Standard angepasst haben, können Sie die Standardkonfiguration nur durch eine Neuinstallation von HP OO wiederherstellen.

Voraussetzungen

Hinweis: Beim Upgrade einer Installation von HP OO 10.10 (und höher), die bereits mit FIPS konfiguriert wurde, müssen Sie die Schritte 4 und 5 im folgenden Abschnitt wiederholen und dann die Schritte im Abschnitt "Konfigurieren der Eigenschaften in der Java-Sicherheitsdatei" in "Konfigurieren der FIPS 140-2-Konformität von HP OO" auf Seite 24 wiederholen.

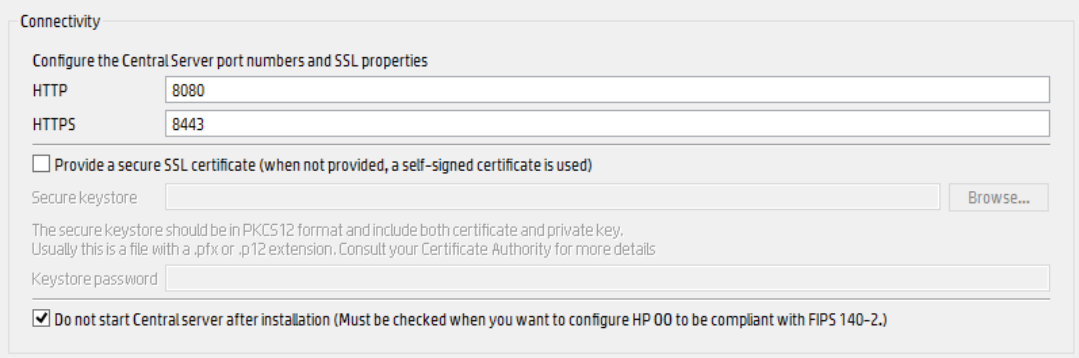
Führen Sie vor der FIPS 140-2-konformen HP OO-Konfiguration die folgenden Schritte aus:

Hinweis: Um den FIPS140-2-Standard zu erfüllen, müssen Sie LWSSO ausschalten.

1. Vergewissern Sie sich, dass Sie eine neue Installation von HP OO Version 10.10 oder höher für FIPS 140-2 konfigurieren, die gerade nicht verwendet wird.

   Eine Installation von HP OO, die gerade verwendet wird (ob Version 9.x oder 10.x), kann nicht konfiguriert werden.
2. Vergewissern Sie sich, dass HP OO bei der Installation so konfiguriert wurde, dass der Central Server nach der Installation nicht gestartet wird:
   - Bei einer Installation im Hintergrund wurde der Parameter `should.start.central` auf **no** gesetzt.
   - In einer mit dem Assistenten durchgeführten Installation wurde beim Schritt **Verbindung** das

Kontrollkästchen **Do not start Central server after installation** aktiviert.



3. Sichern Sie die folgenden Verzeichnisse:
   - **<Installationsverzeichnis>\central\tomcat\webapps\oo.war**
   - **<Installationsverzeichnis>\central\tomcat\webapps\PAS.war**
   - **<Installationsverzeichnis>\central\conf**
   - **<oo_jre>\lib\security** (**<oo_jre>** ist das Verzeichnis, in dem die von HP OO verwendete JRE installiert ist. Standardmäßig ist dies das Verzeichnis **<Installationsverzeichnis>\java**)
4. Laden Sie die Java Cryptographic Extension (JCE) Unlimited Strength Jurisdiction Policy Files von der folgenden Website herunter und installieren Sie sie: http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html.

   Hinweis: Informationen, wie Sie die Dateien verteilen und die von HP OO verwendete JRE aktualisieren, finden Sie in der Datei **ReadMe.txt**, die zu den heruntergeladenen Dateien gehört.
5. Installieren Sie die RSA BSAFE Crypto-Dateien. Kopieren Sie auf dem System, auf dem HP OO installiert ist, die folgenden Dateien in den Ordner **<oo_jre>\lib\ext** (**<oo_jre>** ist das Verzeichnis, in dem die von HP OO verwendete JRE installiert ist. Standardmäßig ist dies der Ordner **<Installationsverzeichnis>\java**).
   - **<Installationsverzeichnis>\central\lib\cryptojce-6.1.jar**
   - **<Installationsverzeichnis>\central\lib\cryptojcommon-6.1.jar**
   - **<Installationsverzeichnis>\central\lib\jcmFIPS-6.1.jar**

Hinweis: Beim Upgrade einer Installation von HP OO 10.10 (und höher), die bereits mit FIPS konfiguriert wurde, müssen Sie die Schritte 4 und 5 im obigen Abschnitt "Voraussetzungen" wiederholen und dann die Schritte im Abschnitt "Konfigurieren der Eigenschaften in der Java-Sicherheitsdatei" unter "Konfigurieren der FIPS 140-2-Konformität von HP OO" auf der nächsten Seite wiederholen.

Konfigurieren der FIPS 140-2-Konformität von HP OO

Die folgende Liste enthält die Prozeduren, die Sie durchführen müssen, um HP OO in Übereinstimmung mit FIPS 140-2 zu konfigurieren:

- Konfigurieren der Eigenschaften in der Java-Sicherheitsdatei
- Konfigurieren der Datei "encryption.properties" und Aktivieren des FIPS-Modus
- Erstellen einer FIPS-kompatiblen HP OO-Verschlüsselung
- Erneutes Verschlüsseln des Datenbankkennworts mit der neuen Verschlüsselung
- Starten von HP OO

Konfigurieren der Eigenschaften in der Java-Sicherheitsdatei

Bearbeiten Sie die Java-Security-Datei für JRE, um zusätzliche Sicherheitsanbieter hinzuzufügen, und konfigurieren Sie die Eigenschaften für die FIPS 140-2-Konformität.

Hinweis: Das Upgrade auf HP OO 10.10 ersetzt alle installierten JRE-Dateien. Deshalb müssen Sie nach dem Upgrade auf 10.10 die folgenden Schritte durchführen:

Hinweis: Beim Upgrade einer Installation von HP OO 10.10 (und höher), die bereits mit FIPS konfiguriert wurde, müssen Sie die Schritte 4 und 5 im Abschnitt "Voraussetzungen" unter " Konfigurieren der FIPS 140-2-Konformität Stufe 1 in HP OO" auf Seite 22 wiederholen und dann die Schritte im vorliegenden Abschnitt wiederholen.

Öffnen Sie die Datei **< oo_jre>\lib\security\java.security** in einem Editor und führen Sie die folgenden Schritte aus:

1. Erhöhen Sie bei jedem im Format **security.provider.nn=<provider_name>** gelisteten Anbieter die Reihenfolgennummer `<nn>` um zwei.

   Ändern Sie beispielsweise den Anbietereintrag:

   ```
   security.provider.1=sun.security.provider.Sun
   ```

   in

   ```
   security.provider.3=sun.security.provider.Sun
   ```

2. Fügen Sie einen neuen Standardanbieter hinzu (RSA JCE). Fügen Sie den folgenden Anbieter am Anfang der Anbieterliste ein:

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

3. Fügen Sie RSA BSAFE als neuen SSL-J Java Secure Sockets Extension (JSSE) Provider hinzu.

   ```
   security.provider.2=com.rsa.jsse.JsseProvider
   ```

4. Kopieren und fügen Sie die folgende Zeile in die Datei **java.security** ein, um sicherzustellen, dass **RSA BSAFE** im FIPS 140-2-konformen Modus verwendet wird:

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

Sie können diese Zeile an beliebiger Stelle in der Datei **java.security** einfügen.

5. Da der Standard-DRBG-Algorithmus ECDRBG128 (gemäß NIST) nicht sicher ist, legen Sie für die security-Eigenschaft **com.rsa.crypto.default** den Wert **HMACDRBG** fest, indem Sie die folgende Zeile in die Datei **java.security** kopieren:

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

   Sie können diese Zeile an beliebiger Stelle in der Datei **java.security** einfügen.
6. Speichern und schließen Sie die Datei **java.security**.

Konfigurieren der Datei "encryption.properties" und Aktivieren des FIPS-Modus

Die HP OO-Datei encryption.properties muss aktualisiert werden, um FIPS 140-2-konform zu sein.

1. Sichern Sie die Datei **encryption.properties**, die sich in **<Installationsverzeichnis>\central\var\security** befindet.
2. Öffnen Sie die Datei **encryption.properties** in einem Texteditor. Bearbeiten Sie beispielsweise die folgende Datei:

   C:\Programme\Hewlett-Packard\HP Operations Orchestration\central\var\security\encryption.properties.
3. Suchen Sie nach `keySize=128` und ersetzen Sie diese Angabe durch `keySize=256`.
4. Suchen Sie nach `secureHashAlgorithm=SHA1` und ersetzen Sie diese Angabe durch `secureHashAlgorithm=SHA256`.
5. Suchen Sie nach `FIPS140ModeEnabled=false` und ersetzen Sie diese Angabe durch `FIPS140ModeEnabled=true`.

   Hinweis: Wenn `FIPS140ModeEnabled=false` nicht vorhanden ist, fügen Sie `FIPS140ModeEnabled=true` als neue Zeile am Ende der Datei hinzu.
6. Speichern und schließen Sie die Datei.

Erstellen einer FIPS-kompatiblen HP OO-Verschlüsselung

Informationen dazu, wie Sie eine HP OO-Verschlüsselungsspeicherdatei erstellen oder ersetzen, sodass sie FIPS-konform ist, finden Sie unter "Ersetzen der FIPS-Verschlüsselung" auf der nächsten Seite.

Hinweis: Für AES sind drei Schlüssellängen zulässig: 128/192/256 laut NIST SP800-131A.

Die folgenden Secure-Hash-Algorithmen werden in FIPS unterstützt: SHA1, SHA256, SHA384, SHA512.

Hinweis: Es wird empfohlen, die Kennwörter für den Keystore (und den Eintrag mit dem privaten Schlüssel) und den Truststore zu ändern. Weitere Informationen finden Sie unter "Ändern des Kennworts für den KeyStore/TrustStore" auf Seite 11.

Hinweis: Es wird empfohlen, alle nicht verwendeten Standard-CA-Stammzertifikate im HP OO-Truststore zu löschen. (Die Datei **client.truststore** befindet sich unter **<Installation>/central/var/security**.)

Erneutes Verschlüsseln des Datenbankkennworts mit der neuen Verschlüsselung

Verschlüsseln Sie das Datenbankkennwort erneut. Die entsprechende Beschreibung finden Sie im *HP OO Administration Guide* unter "Ändern des Datenbankkennworts".

Starten von HP OO

Starten Sie HP OO gemäß Beschreibung im *HP OO Installation Guide*.

Ersetzen der FIPS-Verschlüsselung

HP OO, Central und RAS entsprechen dem FIPS-Standard 140-2 (Federal Information Processing Standard), der die technischen Anforderungen definiert, die von US-Bundesbehörden einzuhalten sind, wenn diese Organisationen kryptografische Sicherheitssysteme zum Schutz vertraulicher oder wertvoller Daten spezifizieren.

Nach einer Neuinstallation von HP OO 10.10 haben Sie die Möglichkeit, den FIPS-Verschlüsselungsschlüssel zu ändern.

Hinweis: Dieses Verfahren ist nur bei Neuinstallationen möglich. Sie können es nicht nach Upgradeinstallationen anwenden.

Ändern des FIPS-Verschlüsselungsschlüssels in Central

1. Wechseln Sie zu **<Central-Installationsordner>/var/security**.
2. Sichern und löschen Sie die Datei **encryption_repository**.
3. Wechseln Sie zu **<Central-Installationsordner>/bin**.
4. Führen Sie das Skript **generate-keys** aus.

   Ein neuer Masterschlüssel wird in **<Central-Installationsordner>/var/security/encryption_repository** generiert.

Ändern der RAS-Verschlüsselungseigenschaften

Wenn Sie die RAS-Installation an einem neuen Standort durchgeführt haben, müssen Sie alle folgenden Schritte ausführen.

Hinweis: Diese Änderungen sind nur gültig, wenn Sie eine neue RAS-Installation bearbeiten, nachdem Sie die Central-Verschlüsselungseigenschaften geändert haben.

So ändern Sie die RAS-Verschlüsselungseigenschaften:

1. Führen Sie alle Schritte im Abschnitt "Voraussetzungen" unter " Konfigurieren der FIPS 140-2-Konformität Stufe 1 in HP OO" auf Seite 22 aus.
2. Führen Sie alle Schritte im Abschnitt "Konfigurieren der Eigenschaften in der Java Security-Datei" unter "Konfigurieren der FIPS 140-2-Konformität von HP OO" auf Seite 24 aus.
3. Kopieren Sie die aktuelle **encryption.properties**-Datei aus dem Ordner **<Installationsverzeichnis>\ras\var\security** in den Ordner **<Installationsverzeichnis>\ras\bin**.
4. Bearbeiten und ändern Sie die Datei **encryption.properties** in einem Texteditor nach Bedarf.

   Weitere Informationen finden Sie unter "Konfigurieren der Datei encryption.properties und Aktivieren des FIPS-Modus" unter "Konfigurieren der FIPS 140-2-Konformität von HP OO" auf Seite 24.
5. Speichern Sie die Änderungen.
6. Öffnen Sie eine Befehlszeile im Ordner **<Installationsverzeichnis>\ras\bin**.
7. Führen Sie die Datei **oosh.bat** aus.
8. Führen Sie den OOShell-Befehl aus: `replace-encryption --file encryption.properties`

   Hinweis: Wenn Sie die Datei **encryption.properties** in einen anderen Ordner kopiert haben, müssen Sie den richtigen Speicherort im OOShell-Befehl angeben.
9. Starten Sie den RAS-Dienst wieder.

Konfigurieren des TLS-Protokolls

Sie können HP OO konfigurieren, um die unterstützte TLS-Protokollversion zu definieren. Standardmäßig lässt HP OO die Versionen TLS v1, TLS v1.1 und TLS v1.2 zu. Dies können aber eingrenzen.

Hinweis: SSLv3 und andere Versionen von SSL werden nicht unterstützt.

1. Öffnen Sie die Datei **<Installationsordner>/central/tomcat/conf/server.xml**.
2. Suchen Sie den SSL-Connector (am Ende der Datei).
3. Bearbeiten Sie den Standardwert von `sslEnabledProtocols`. Ändern Sie z. B. `sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2"` in `sslEnabledProtocols="TLSv1.2"`
4. Starten Sie den Server neu.


Java™
COMPATIBLE