HP Operations Orchestration

Pour les systèmes d'exploitation Windows et Linux

Version du logiciel : 10.10

Manuel de configuration et de sécurisation du système

Date de publication du document : Mai 2014

Date de lancement du logiciel : Mai 2014

Mentions légales

Garantie

Les seules garanties applicables aux produits et services HP sont celles figurant dans les déclarations de garantie expresse accompagnant les dits produits et services. Aucun terme de ce document ne peut être interprété comme constituant une garantie supplémentaire. HP ne peut en aucun cas être tenu pour responsable des erreurs ou omissions techniques ou rédactionnelles du présent document.

Les informations contenues dans le présent document sont susceptibles d'être modifiées sans préavis.

Légende de restriction des droits

Logiciel confidentiel. Licence HP valide requise pour la détention, l'utilisation ou la copie. En accord avec les articles FAR 12.211 et 12.212, les logiciels informatiques, la documentation des logiciels et les informations techniques commerciales sont concédés au gouvernement américain sous licence commerciale standard du fournisseur.

Copyright

Marques

Adobe™ est une marque déposée de Adobe Systems Incorporated.

Ce produit inclut une interface de la bibliothèque de compression d'usage général 'zlib', Copyright © 1995 - 2002 Jean-loup Gailly et Mark Adler.

AMD et le logo AMD avec la flèche sont des marques déposées d'Advanced Micro Devices, Inc.

Google ™ et Google Maps ™ sont des marques commerciales de Google Inc.

Intel ® Xeon ®, Itanium ®, Pentium ® et Intel ® sont des marques déposées d'Intel Corporation aux États-Unis et/ou dans d'autres pays.

Java est une marque déposée de Oracle Corporation et/ou de ses sociétés liées.

Microsoft®, Windows®, Windows NT®, Windows® XP, et Windows Vista® sont des marques déposées de Microsoft Corporation aux États-Unis.

Oracle est une marque déposée de Oracle Corporation et/ou de ses sociétés liées.

UNIX® est une marque déposée de The Open Group.

Mises à jour de la documentation

La page de titre du présent document contient les informations d'identifications suivantes :

- le numéro de version du logiciel ;
- la date de publication du document, qui change à chaque mise à jour de ce dernier ;
- la date de lancement du logiciel.

Pour obtenir les dernières mises à jour ou vérifier que vous disposez de l'édition la plus récente d'un document, accédez à la page : **http://h20230.www2.hp.com/selfsolve/manuals**

Pour accéder à ce site, vous devez créer un compte HP Passport et vous connecter comme tel. Pour obtenir un identifiant HP Passport, accédez à l'adresse : **http://h20229.www2.hp.com/passport-registration.html**

Vous pouvez également cliquer sur le lien **New users - please register** dans la page de connexion de HP Passport.

En vous abonnant au service d'assistance du produit approprié, vous recevrez en outre les dernières mises à jour ou les nouvelles éditions. Pour plus d'informations, contactez votre revendeur HP.

Assistance

Visitez le site d'assistance HP Software à l'adresse : **http://www.hp.com/go/hpsoftwaresupport**

Ce site fournit les informations de contact et les détails sur les offres de produits, de services et d'assistance HP Software.

L'assistance en ligne de HP Software propose des fonctions de résolution autonome. Le site constitue un moyen efficace d'accéder aux outils interactifs d'assistance technique nécessaires à la gestion de votre activité. En tant que client privilégié de l'assistance, vous pouvez depuis ce site :

- rechercher des documents de connaissances présentant un réel intérêt ;
- soumettre et suivre des demandes d'assistance et des demandes d'améliorations ;
- télécharger des correctifs logiciels ;
- gérer des contrats d'assistance ;
- rechercher des contacts de l'assistance HP ;
- consulter les informations sur les services disponibles ;
- participer à des discussions avec d'autres utilisateurs d'un même logiciel ;
- rechercher des cours de formation sur les logiciels et vous y inscrire.

Pour accéder à la plupart des offres d'assistance, vous devez vous enregistrer en tant qu'utilisateur disposant d'un compte HP Passport et vous identifier comme tel. De nombreuses offres nécessitent en outre un contrat d'assistance. Pour obtenir un identifiant HP Passport, accédez à l'adresse suivante :

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

Les informations relatives aux niveaux d'accès sont détaillées à l'adresse suivante :

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

HP Software Solutions Now accède au site Web du portail HPSW Solution and Integration. Ce site vous permet d'explorer les pages de HP Product Solutions qui comprendent une liste complète des intégrations entre produits HP, ainsi qu'une liste des processus ITIL. L'URL de ce site Web est **http://h20230.www2.hp.com/sc/solutions/index.jsp**

Table des matières

Configuration et sécurisation du système

Ce document décrit la configuration et la sécurisation de HP Operations Orchestration.

Authentification via certificat de serveur et de client

Les certificats Secure Socket Layer (SSL)/Transport Layer Security (TLS) associent numériquement une clé de chiffrement aux détails d'une organisation, se qui permet d'établir des connexions sécurisées entre un serveur Web et un navigateur.

HP OO gère les clés de chiffrement et les certificats de confiance à l'aide de l'utilitaire Keytool. Cet utilitaire se trouve dans le dossier d'installation de HP OO, dans **<Répertoire d'installation>/java/bin/keytool**. Pour plus d'informations sur l'utilitaire Keytool, consultez http://docs.oracle.com/javase/7/docs/technotes/tools/solaris/keytool.html.

Les installations de HP OO Central contiennent deux fichiers pour la gestion des certificats :

- **<répertoire d'installation>/central/var/security/client.truststore** : contient la liste des certificats de confiance.
- **<répertoire d'installation>/central/var/security/key.store** : contient le certificat HP OO.

Il est conseillé de remplacer le certificat HP OO après une nouvelle installation de HP OO ou si votre certificat actuel a expiré.

Authentification par certificat de serveur

Remplacement du certificat de serveur SSL/TLS de Central

Vous pouvez utiliser un certificat signé par une société bien connue ou un certificat de serveur personnalisé.

Remplacez les paramètres qui sont mis en évidence en <jaune> pour adapter l'emplacement du fichier **key.store** et autres détails à votre ordinateur.

Remarque : La procédure suivante utilise l'utilitaire Keytool qui se trouve dans **<répertoire d'installation>/java/bin/keytool**.

1. Arrêtez Central et réalisez une sauvegarde du fichier **key.store** original qui se trouve dans **<répertoire d'installation>/central/var/security/key.store**.
2. Ouvrez une ligne de commande dans **<répertoire d'installation>/central/var/security**.
3. Supprimez le certificat de serveur existant dans le fichier **key.store** de Central à l'aide de la

commande suivante :

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

4. Si vous possédez déjà un certifcat avec l'extension **.pfx** ou **.p12**, passez à l'étape suivante. Dans le cas contraire, il faudra exporter le certificat avec la clé privée au format PKCS12 (.pfx, .p12). Par exemple, si le certificat est au format PEM :

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

   Si le certificat est au format DER, ajoutez le paramètre `–inform DER` après pkcs12. Par exemple :

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

   Remarque : Prenez note du mot de passe que vous fournissez. Vous aurez besoin de ce mot de passe pour la clé privée lorsque vous devrez saisir la phrase secrète du keystore plus loin dans cette procédure.

5. Extrayez l'alias de l'alias du certificat via la commande suivante :

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

   L'alias est affiché. Dans l'exemple ci-dessous, il s'agit de la quatrième ligne à partir du bas.

   ```
   c:\Program Files\Hewlett-Packard\oo-sam\central\var\security>keytool -list -keystore server.pfx -v -storetype PKCS12
   Enter keystore password:

   Keystore type: PKCS12
   Keystore provider: SunJSSE

   Your keystore contains 1 entry

   Alias name: le-775fb32c-269c-499b-bae8-fe7077479ec6
   Creation date: 24/04/2014
   Entry type: PrivateKeyEntry
   Certificate chain length: 2
   ```

6. Importez le certifcat de serveur au format PKSC12 dans le fichier Central **key.store** :

   keytool -importkeystore -srckeystore <chemin d'accès au certificat au format PKCS12> -destkeystore key.store -srcstoretype pkcs12 -deststoretype JKS -alias <alias du certificat> -destalias tomcat

7. Il est recommandé de modifier le mot de passe par défaut « changeit » dans le keystore généré automatiquement dans le serveur Central. Voir "Modification du mot de passe du keystore/truststore" , page 11.

8. Démarrez Central.

Remplacement du certificat de serveur SSL/TLS de Central par un certificat auto-signé

Vous pouvez créer un certificat auto-signé à l'aide de l'utilitaire Keytool.

Remarque : Après la mise à niveau à HP OO 10.10 :

- Si un nouveau Central est installé sur le même ordinateur que l'installation précédente, vous pouvez utiliser le certificat auto-signé existant.
- Si de nouvelles instances de Central sont installées sur des machines différentes, vous devez créer un certifcat auto-signé pour chacune d'entre elles, même si vous aviez un certificat pour la version antérieure.

Remarque : La procédure suivante utilise l'utilitaire Keytool qui se trouve dans **<répertoire d'installation>/java/bin/keytool**.

Remplacez les paramètres qui sont mis en évidence en <jaune> pour adapter l'emplacement du fichier **key.store** et autres détails à votre ordinateur.

1. Arrêtez Central et réalisez une sauvegarde du fichier **key.store** original qui se trouve dans **<répertoire d'installation>/central/var/security/key.store**.
2. Ouvrez une ligne de commande dans **<répertoire d'installation>/central/var/security**.
3. Supprimez le certificat de serveur existant dans le fichier **key.store** de Central à l'aide de la commande suivante :

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

4. Créez un certificat auto-signé :

   ```
   keytool -genkey -alias tomcat -keyalg RSA -keypass <changeit >-keystore
   <path/for/new/Keystore> -storepass <changeit>-storetype pkcs12 -dname
   "CN=<NOM DE DOMAINE COMPLET DE CENTRAL>, OU=<UNITÉ_ORGANISATIONELLE>,
   O=<ORGANISATION>, L=<LOCALITÉ>, C=<PAYS>"
   ```

   Remarque : Si vous ne saisissez pas un chemin d'accès pour générer le nouveau keystore, il est créé dans le dossier où vous avez saisi la commande, par exemple **<répertoire d'installation>/central/var/security**.

5. Importez le certifcat auto-signé dans le fichier Central **key.store** :

   ```
   keytool -v -importkeystore -srckeystore <new/path/created/Keystore> -
   srcstoretype PKCS12 -srcstorepass <changeit> -destkeystore key.store -
   deststoretype JKS -deststorepass <changeit>
   ```

6. Démarrez Central.

Importation d'un certificat dans un truststore RAS

Après avoir installé un RAS, si vous utilisez un certifcat racine personnalisé pour Central et que vous n'aviez pas désigné ce certificat pendant l'installation du RAS, il faudra importer l'autorité de certificat racine de confiance dans le **client.truststore** du RAS. Si vous utilisez un certificat racine signé standard, il n'est pas nécessaire de réaliser la procédure suivante car le certificat se trouvera déjà dans le fichier **client.truststore**.

Par défaut, HP OO prend en charge tous les certificats auto-signés. Toutefois, dans un environnement de production, il est conseillé de changer cette valeur par défaut pour des raisons de sécurité.

Remplacez les paramètres qui sont mis en évidence en <jaune>.

Remarque : La procédure suivante utilise l'utilitaire Keytool qui se trouve dans **<répertoire d'installation>/java/bin/keytool**.

1. Arrêtez RAS et réalisez une sauvegarde du fichier **client.truststore** original qui se trouve dans **<répertoire d'installation>/ras/var/security/client.truststore**.
2. Ouvrez une ligne de commande dans **<répertoire d'installation>/ras/var/security**.
3. Ouvrez le fichier **<répertoire d'installation> ras/conf/ras-wrapper.conf** et attribuez la valeur **false** à `-Dssl.support-self-signed`. Ceci active l'autorité de certificat racine de confiance.

   Par exemple :

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

4. Ouvrez le fichier **<répertoire d'installation> ras/conf/ras-wrapper.conf** et attribuez la valeur **true** à `-Dssl.verifyHostName`. Ceci vérifie le nom d'hôte.

   Par exemple :

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

5. Importez l'autorité de certificat de confiance racine dans le fichier **client.truststore** du RAS :

   ```
   keytool -importcert -alias <alias_quelconque> –keystore client.truststore -
   file <nom_certificat.cer> –storepass <changeit>
   ```

6. Démarrez RAS.

Importation d'un certificat dans le truststore de OOSH

Si vous utilisez un certificat racine personnalisé pour Central, il faudra importer l'autorité de certificat racine de confiance dans le **client.truststore** d'OOSH. Si vous utilisez un certificat racine

signé standard, il n'est pas nécessaire de réaliser la procédure suivante car le certificat se trouvera déjà dans le fichier **client.truststore**.

Par défaut, HP OO prend en charge tous les certificats auto-signés. Toutefois, dans un environnement de production, il est conseillé de changer cette valeur par défaut pour des raisons de sécurité.

Remplacez les paramètres qui sont mis en évidence en <jaune>.

Remarque : La procédure suivante utilise l'utilitaire Keytool qui se trouve dans **<répertoire d'installation>/java/bin/keytool**.

1. Arrêtez Central et réalisez une sauvegarde du fichier **client.truststore** original qui se trouve dans **<répertoire d'installation>/central/var/security/client.truststore**.
2. Modifiez le fichier **oosh.bat** dans **<répertoire d'installation>/central/bin**.
3. Attribuez la valeur **false** à `-Dssl.support-self-signed`. Ceci active l'autorité de certificat racine de confiance.

   Par exemple :

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

4. Attribuez la valeur **true** à `-Dssl.verifyHostName`. Ceci vérifie le nom d'hôte.

   Par exemple :

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

5. Importez l'autorité de certificat racine de confiance dans le fichier **client.truststore** :

   ```
   keytool -importcert -alias <alias_quelconque> –keystore client.truststore -file <nom_certificat.cer> –storepass <changeit>
   ```

6. Exécutez OOSH.

Importation d'un certificat dans le truststore de Studio Debugger

Une fois que Studio a été installé, si vous utilisez un certificat racine personnalisé pour Studio, il faudra importer l'autorité de certificat racine de confiance dans le **client.truststore** de Studio. Si vous utilisez un certificat racine signé standard, il n'est pas nécessaire de réaliser la procédure suivante car le certificat se trouvera déjà dans le fichier **client.truststore**.

Par défaut, HP OO prend en charge tous les certificats auto-signés. Toutefois, dans un environnement de production, il est conseillé de changer cette valeur par défaut pour des raisons de sécurité.

Remplacez les paramètres qui sont mis en évidence en <jaune>.

Remarque : La procédure suivante utilise l'utilitaire Keytool qui se trouve dans **<répertoire d'installation>/java/bin/keytool**.

1. Quittez Studio et réalisez une sauvegarde du fichier **client.truststore** original qui se trouve dans **<répertoire d'installation>/studio/var/security/client.truststore**.
2. Modifiez le fichier **Studio.l4j.ini** dans **<rép_installation>/studio**.
3. Attribuez la valeur **false** à `-Dssl.support-self-signed`. Ceci active l'autorité de certificat racine de confiance.

   Par exemple :

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

4. Attribuez la valeur **true** à `-Dssl.verifyHostName`. Ceci vérifie le nom d'hôte.

   Par exemple :

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

5. Importez l'autorité de certificat racine de confiance dans le fichier **client.truststore** de Studio :

   ```
   keytool -importcert -alias <alias_quelconque> –keystore client.truststore -
   file <nom_certificat.cer> –storepass <changeit>
   ```

6. Démarrez Studio.

Pour plus d'informations, voir « Debugging a Remote Central with Studio » dans le manuel *Studio Authoring Guide*.

Modification du mot de passe du keystore/truststore

- **Pour modifier le mot de passe Central** :

  a. Modifiez le fichier **server.xml** qui se trouve dans **<rép_ installation>/central/tomcat/conf/server.xml**.

  b. Localisez le connecteur HTTPS Par exemple :

  ```
  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" truststoreFile="C:/Program
  Files/Hewlett-Packard/HP Operations
  Orchestration/central/var/security/client.truststore"
  truststorePass="changeit" truststoreType="JKS"/>
  ```

c. Modifiez le mot de passe requis.

- `keyPass` : mot de passe utilisé pour accéder au certificat de serveur depuis le fichier keystore indiqué. La valeur par défaut est « `changeit` ».
- `keystorePass` : le mot de passe pour accéder au fichier keystore indiqué. La valeur par défaut est la valeur de l'attribut `keyPass`.
- `truststorePass` : le mot de passe pour accéder au truststore. La valeur par défaut est la valeur de la propriété système **javax.net.ssl.trustStorePassword**. Si la valeur de cette propriété est null, aucun mot de passe truststore ne sera configuré. Si un mot de passe truststore non valide est proposé, un avertissement sera consigné et une tentative d'accès au truststore sans mot de passe sera réalisée, qui ignorera la validation du contenu du truststore.

d. Enregistrez le fichier.

e. Ouvrez **central-wrapper.conf**, sous **central/conf** et modifiez :

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

f. Redémarrez Central.

- **Pour modifier le mot de passe du truststore RAS** : Modifiez le fichier **ras-wrapper.conf** et modifiez le paramètre `changeit` du truststore.
- **Pour modifier le mot de passe du truststore OOSH** : Modifiez le fichier **oosh.bat** et modifiez le paramètre `changeit` du truststore.
- **Pour modifier le mot de passe du truststore Studio** : Modifiez le fichier **<rép_ installation>/studio/Studio.l4j.ini** et modifiez le paramètre `changeit` du truststore.

Suppression de l'algorithme de chiffrement RC4 des algorithmes pris en charge par SSL

L'hôte distant prend en charge l'utilisation de l'algorithme de chiffrement RC4. Cet algorithme présente un défaut dans la création d'un flux d'octets pseudo-aléatoire qui entraîne l'insertion d'un large éventail de petits écarts dans le flux, ce qui réduit son caractère aléatoire.

Si du texte brut est chiffré à plusieurs reprises (par exemple, des cookies HTTP) et qu'un attaquant parvient à obtenir plusieurs (à savoir, des dizaines de millions) de textes de chiffrement, il pourrait arriver à découvrir le texte brut.

Désactiver l'algorithme de chiffrement RC4 au niveau du JRE (à partir de Java 7) :

1. Ouvrez le fichier **$JRE_HOME/lib/security/java.security**.
2. Modifiez la propriété **jdk.tls.disabledAlgorithms** pour désactiver l'algorithme de chiffrement RC4.

Pour plus d'informations, consultez http://stackoverflow.com/questions/18589761/restict-cipher-suites-on-jre-level.

Modification ou fermeture des ports HTTP/HTTPS

Le fichier **server.xml** sous **[OO_HOME]\central\Tomcat\conf** contient deux éléments baptisés **<Connector>** sous l'élément **<Service>**. Ces connecteurs définissent ou activent les ports que le serveur écoute.

La configuration de chaque connecteur est définie via ses attributs. Le premier connecteur définit un connecteur HTTP régulier tandis que le deuxième définit un connecteur HTTPS.

Par défaut, les connecteurs ressemblent à ceci.

Connecteur HTTP :

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

Connecteur HTTPS :

```
<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" truststoreFile="C:/Program Files/Hewlett-
Packard/HP Operations Orchestration/central/var/security/client.truststore"
truststorePass="changeit" truststoreType="JKS"/>
```

Les deux sont activés par défaut.

Modification des valeurs du port

Pour modifier les valeurs d'un des ports :

1. Modifiez le fichier **server.xml** qui se trouve dans **<rép_ installation>/central/tomcat/conf/server.xml**.
2. Localisez le connecteur HTTP ou HTTPS et modifiez la valeur **port** dans la ligne.

> **Remarque :** Si HTTP et HTTPS sont tous les deux actifs et que vous souhaitez modifier le port HTTPS, il faudra modifier la valeur **redirectPort** pour le connecteur HTTP.

3. Enregistrez le fichier.
4. Redémarrez Central.

Désactivation d'un mort

Pour désactiver un des ports :

1. Modifiez le fichier **server.xml** qui se trouve dans **<rép_ installation>/central/tomcat/conf/server.xml**.
2. Localisez le connecteur HTTP ou HTTPS et supprimez la ligne ou désactivez-la à l'aide d'un commentaire.
3. Enregistrez le fichier.
4. Redémarrez Central.

Authentification par certificat client (authentification mutuelle)

L'authentification par certificat X.509 est la plus souvent utilisée pour vérifier l'identité d'un serveur avec le protocole SSL/TLS, généralement dans le cadre de l'utilisation du protocole HTTPS depuis un navigateur. Le navigateur vérifie automatiquement si le certificat présenté par un serveur a été émis par une des autorités de certification de confiance qui figure sur la liste que le serveur maintient.

Vous pouvez utilisez également SSL/TLS avec l'authentification mutuelle. Le serveur sollicite un certificat valide du client dans le cadre de la liaison SSL/TLS. Le serveur authentifie le client en confirmant que son certificat a été signé par une autorité acceptable. Si un certificat valide a été fourni, il peut être obtenu via l'API du servlet dans une application.

Configuration de l'authentification par certificat du client dans Central

Avant de configurer l'authentification par certificat de client dans Central, confirmez que vous avez configuré le certificat de serveur SSL, conformément à la description de la section "Authentification via certificat de serveur et de client" , page 6.

Attribuez la valeur `true` à l'attribut `clientAuth` si vous souhaitez que la pile SSL demande au client une chaîne de certificat valide avant d'accepter une connexion. Attribuez la valeur `want` si vous souhaitez que la pile SSL demande un certificat au client, sans prévoir d'échec si aucun certificat n'est présenté. La valeur `false` (par défaut) ne requiert aucune chaîne de certificat sauf si le client sollicite une ressource protégée par une contrainte de sécurité qui utilise l'authentification CLIENT-CERT. (Pour plus d'informations, voir le manuel Apache Tomcat Configuration Reference).

Définissez le fichier **Liste de révocation des certificats (CRL)**. Il peut contenir plusieurs CRL. Dans certains systèmes de chiffrement, en général des infrastructures à clé publique (PKI), une

liste de révocation des certificats désigne une liste de certificats (ou plus spécialement, une liste de numéros de série pour certificats) qui ont été révoqués et par conséquent, il ne faut plus faire confiance aux entités qui présentent ces certificats (révoqués).

Remarque : La procédure suivante utilise l'utilitaire Keytool qui se trouve dans **<répertoire d'installation>/java/bin/keytool**.

1. Arrêtez le serveur Central.
2. Importer le certificat racine (CA) approprié dans Central **cllient.trustore** : **<répertoire d'installation>/central/var/security/client.truststore**, par exemple :

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

3. Modifiez le fichier **server.xml** qui se trouve dans **<rép_ installation>/central/tomcat/conf/server.xml**.
4. Attribuez la valeur `want` ou `true` à l'attribut `clientAuth` dans la balise `Connector`. La valeur par défaut est `false`.

   Remarque : Il est recommandé de démarrer les serveur à la fin de cette procédure, mais vous pouvez le démarrer maintenant.

5. Ajoutez l'attribut `crlFile` pour définir le fichier de liste de révocation de certificats pour la validation du certificat SSL/TLS, par exemple :

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

   Le fichier peut porter l'extension `.crl` pour une seule liste de révocation de certificats ou l'extension `.pem` (format PEM CRL) pour une ou plusieurs listes de révocation de certificats. Le format PEM CRL utilise l'en-tête et le pied de page suivant :

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

   Exemple de structure de fichier `.pem` pour une liste de révocation de certificats (pour plusieurs listes, ajoutez un autre bloc CRL) :

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

```
GXg+gCuTjTKUFfKRnWz7O7RyiJKKim0jtAF4RkCpLebNChY=
-----END X509 CRL-----
```

6. Démarrez le serveur Central.

Remarque : Pour chaque certificat de client, vous devez définir un utilisateur, soit interne, soit utilisateur LDAP. Le nom de l'utilisateur doit être défini dans les attributs du certificat. La valeur par défaut est l'attribut CN. Pour plus d'informations, voir Traitement du principal de certificat.

Sachez que même si HP OO est configuré avec plusieurs configurations LDAP, il est uniquement possible d'authentifier l'utilisateur à l'aide des attributs de certificat de client avec le LDAP par défaut. Central essaiera d'abord d'authentifier l'utilisateur avec le LDAP par défaut et, en cas d'échec, il tentera l'authentification au sein du domaine interne HP OO.

Mise à jour de la configuration d'un certificat de client dans RAS

Le certificat de client est configuré lors de l'installation du RAS. Toutefois, si vous devez actualiser le certificat, vous pouvez réaliser l'opération manuellement dans le fichier **ras-wrapper.conf**.

Prérequis : vous devez importer le certificat racine de l'autorité CA de Central dans le truststore de RAS. Voir "Importation d'un certificat dans un truststore RAS" , page 9.

Pour actualiser la configuration du certificat de client dans un RAS externe :

1. Arrêtez le serveur RAS.
2. Ouvrez le fichier **ras-wrapper.conf** dans **<répertoire d'installation>ras/var/conf/ras-wrapper.conf**.
3. Modifiez la ligne suivante en fonction de votre certificat de client :

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

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

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

4. Démarrez le serveur RAS.

Remarques importantes ! Le certificat de client X.509 doit avoir le nom principal du RAS, qui est l'identifiant du RAS (cf. Traitement d'un principal de certifcat).

L'identifiant du RAS figure sous l'onglet **Topologie** dans Central. Voir la rubrique « Configuration de la topologie – Travailleurs » dans le *Manuel de l'utilisateur de HP OO Central*.

Configuration d'un certificat de client dans le débogueur à distance de Studio

Prérequis : vous devez importer le certificat racine de l'autorité CA de Central dans le truststore de Studio Debugger. Voir "Importation d'un certificat dans le truststore de Studio Debugger" , page 10.

Pour configurer le certificat de client dans le débogueur à distance de Studio.

1. Fermez Studio.
2. Modifiez le fichier **Studio.l4j.ini** dans **<rép_installation>/studio**.
3. Modifiez la ligne suivante en fonction de votre certificat de client :

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

-Djavax.net.ssl.keyStorePassword=changeit

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

4. Démarrez Studio.

Remarque : Pour le certificat de client, vous devez définir un utilisateur, soit interne, soit utilisateur LDAP. Le nom de l'utilisateur doit être défini dans les attributs du certificat. La valeur par défaut est l'attribut CN. Pour plus d'informations, voir Traitement du principal de certificat.

Sachez que même si HP OO est configuré avec plusieurs configurations LDAP, il est uniquement possible d'authentifier l'utilisateur à l'aide des attributs de certificat de client avec le LDAP par défaut. Central essaiera d'abord d'authentifier l'utilisateur avec le LDAP par défaut et, en cas d'échec, il tentera l'authentification au sein du domaine interne HP OO.

Configuration d'un certificat de client dans OOSH

Prérequis : vous devez importer le certificat racine de l'autorité CA de Central dans le truststore de OOSH. Voir "Importation d'un certificat dans le truststore de OOSH" , page 9.

1. Arrêtez OOSH.
2. Modifiez le fichier **oosh.bat** dans **<répertoire d'installation>/central/bin**.
3. Modifiez la ligne suivante en fonction de votre certificat de client :

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

-Djavax.net.ssl.keyStorePassword=changeit
```

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

4. Démarrez OOSH.

Remarque : Pour le certificat de client, vous devez définir un utilisateur, soit interne, soit utilisateur LDAP. Le nom de l'utilisateur doit être défini dans les attributs du certificat. La valeur par défaut est l'attribut CN. Pour plus d'informations, voir Traitement du principal de certificat.

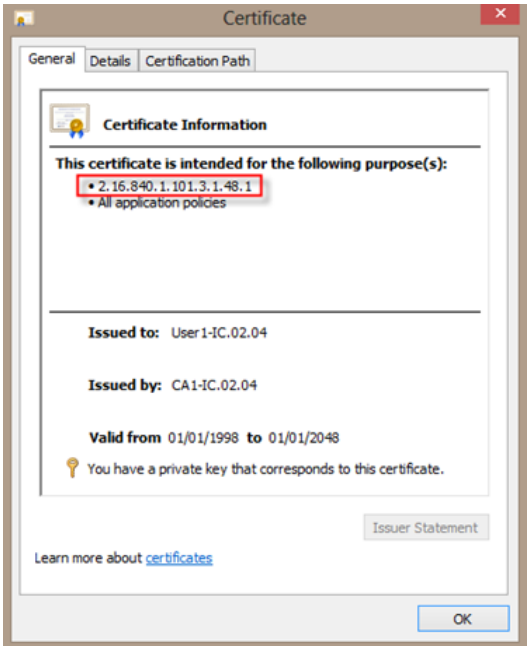
Sachez que même si HP OO est configuré avec plusieurs configurations LDAP, il est uniquement possible d'authentifier l'utilisateur à l'aide des attributs de certificat de client avec le LDAP par défaut. Central essaiera d'abord d'authentifier l'utilisateur avec le LDAP par défaut et, en cas d'échec, il tentera l'authentification au sein du domaine interne HP OO.

Traitement des stratégies de certificat

HP OO gère le traitement des stratégies de certificat pour le certificat final.

- Vous pouvez définir la chaîne d'objectif dans le certificat.
- HP OO vous permet d'ajouter la ou les chaînes de stratégie en tant qu'élément de configuration et de vérifier chaque chaîne de stratégie pour chaque certificat final. En l'absence de correspondance, rejetez le certificat.
- Activez ou désactivez la vérification de la stratégie de certificat en ajoutant l'élément de configuration suivant : `x509.certificate.policy.enabled=true/false` (la valeur par défaut est `false`).
- Définissez la liste de stratégie en ajoutant l'élément de configuration suivant : `x509.certificate.policy.list=<liste_séparée_par_une_virgule>` (la valeur par défaut

est une liste vide).



Traitement d'un principal de certifcat

Vous pouvez définir la manière d'obtenir le principal d'un certificat à l'aide d'une équivalence d'expression régulière sur `Subject`. L'expression régulière doit compter un seul groupe. L'expression par défaut `CN=(.?)` établit l'équivalence avec le champ nom commun. Par exemple, `CN=Jimi Hendrix, OU=` affecte un nom d'utilisateur de `Jimi Hendrix`.

- Les équivalences sont sensibles à la casse.
- Le principal du certificat est le nom d'utilisateur dans HP OO (utilisateur LDAP ou interne).
- Pour changer l'expression régulière, changez l'élément de configuration : `x509.subject.principal.regex`.

Dépannage

Si le serveur ne démarre pas, ouvrez le fichier **wrapper.log** et recherchez une erreur dans `ProtocolHandler ["http-nio-8443"]`.

Ceci peut se produire lorsque Tomcat s'initialise ou lance le connecteur. Il existe plusieurs versions, mais le message d'erreur peut fournir des informations.

Tous les paramètres du connecteur HTTPS se trouve dans le fichier de configuration Tomcat qui se trouve à l'emplacement **C:\HP\oo\central\tomcat\conf\server.xml**.

Ouvrez le fichier et parcourez-le jusqu'à ce que vous trouviez le connecteur HTTPS :

```
<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"/>
```

Vérifiez s'il y a des incohérences dans les paramètres en les comparant aux paramètres saisis lors des étapes antérieures.

Norme FIPS (Federal Information Processing Standard)

Configuration de HP OO pour la mise en conformité avec la norme FIPS 140-2

Cette section explique comment configurer HP Operations Orchestration afin de garantir la conformité à la norme Federal Information Processing Standards (FIPS) 140-2.

La norme FIPS 140-2 est une norme qui porte sur les exigences en matière de sécurité applicables aux modules de chiffrements définies par le National Institute of Standards Technology (NIST). Pour consultez la publication de cette norme, rendez-vous à : csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf.

Après que vous avez configuré HP OO en vue de la conformité avec FIPS 140-2, HP OO utilise l'algorithme de sécurité suivant :

- Algorithme à clé symétrique : AES
- Algorithme de hachage : SHA1

HP OO utilise le fournisseur de sécurité suivant : logiciel RSA BSAFE Crypto version 6.1.Il s'agit du seul fournisseur de sécurité pris en charge pour FIPS 140-2.

Remarque : Une fois que vous aurez configuré HP OO pour le rendre conforme à la norme FIPS 140-2, la seule manière de revenir à la configuration standard consiste à réinstaller HP OO.

Prérequis

Remarque : Si vous effectuez la mise à niveau à partir d'une installation de HP OO 10.10 (et ultérieure) déjà configurée avec FIPS, vous devez répéter les étapes 4 et 5 ci-dessous, puis répéter les étapes de la section « Configurer les propriétés du fichier de sécurité java » dans "Configuration de HP OO pour respecter la norme FIPS 140-2" , page 23.

Avant de configurer HP OO pour le rendre conforme à la norme FIPS 140-2; réalisez les opérations suivantes :

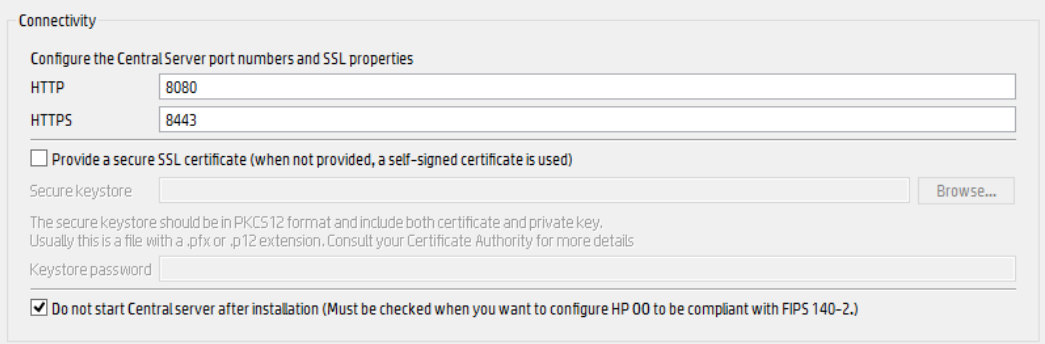
Remarque : Pour la conformité FIPS 140-2, il faut désactiver LWSSO.

1. Pour respecter la norme FIPS 140-2, vérifiez que vous êtes en train de configurer une nouvelle installation de HP OO version 10.10 ou ultérieure, et que celle-ci n'est pas en cours d'utilisation.

Vous ne pouvez pas configurer une installation de HP OO en cours d'utilisation (quelle que soit la version, 9.x ou 10.x).

2. Confirmez après l'installation de HP OO qu'il a été configuré pour ne pas démarrer le serveur Central après l'installation :

   - Dans une installation silencieuse, la valeur **no** a été attribuée au paramètre `should.start.central`.
   - Dans l'installation via un Assistant, à l'étape **Connectivité**, la case **Ne pas démarrer le serveur Central après l'installation** a été cochée.



3. Sauvegardez les répertoires suivants :

   - **<répertoire d'installation>\central\tomcat\webapps\oo.war**
   - **<répertoire d'installation>\central\tomcat\webapps\PAS.war**
   - **<répertoire d'installation>\central\conf**
   - **<oo_jre>\lib\security** (où **<oo_jre>** est le répertoire dans lequel le JRE utilisé par HP OO est installé. Par défaut, il s'agit de **<rép_installation>\java**)

4. Téléchargez et installez les fichiers Java Cryptographic Extension (JCE) Unlimited Strength Jurisdiction Policy depuis le site suivant : http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html.

   Remarque : Voir le fichier **ReadMe.txt** du contenu téléchargé pour savoir comment déployer les fichiers et mettre le JRE utilisé par HP OO à jour.

5. Installez les fichiers du logiciel RSA BSAFE Crypto. Sur le système où HP OO est installé, copiez les éléments suivants dans **<oo_jre>\lib\ext** (où **<oo_jre>** est le répertoire dans lequel le JRE utilisé par HP OO est installé. Par défaut, il s'agit de **<rép_installation\java**).

   - **<répertoire d'installation>\central\lib\cryptojce-6.1.jar**
   - **<répertoire d'installation>\central\lib\cryptojcommon-6.1.jar**

- **<répertoire d'installation>\central\lib\jcmFIPS-6.1.jar**

Remarque : Si vous effectuez la mise à niveau à partir d'une installation de HP OO 10.10 (et ultérieure) déjà configurée avec FIPS, vous devez répéter les étapes 4 et 5 de la section « Prérequis » ci-dessus, puis répéter les étapes de la section « Configurer les propriétés du fichier de sécurité java » dans "Configuration de HP OO pour respecter la norme FIPS 140-2" , ci-dessous.

Configuration de HP OO pour respecter la norme FIPS 140-2

La liste suivante décrit les procédures à exécuter pour mettre HP OO en conformité avec la norme FIPS 140-2 :

- Configurer les propriétés du fichier de sécurité java
- Configurer le fichier encryption.properties et activez le mode FIPS.
- Créer un chiffrement HP OO conforme avec la norme FIPS.
- Remplacer le mot de passe de la base de données
- Démarrer HP OO.

Configurer les propriétés du fichier de sécurité java

Modifiez le fichier de sécurité Java pour le JRE afin d'ajouter des fournisseurs de sécurité complémentaires et configurez les propriétés pour garantir la conformité à la norme FIPS 140-2.

Remarque : La mise à niveau à HP OO 10.10 remplace complètement les fichiers JRE installés. Par conséquent, les étapes suivantes doivent être réalisées après la mise à niveau à 10.10.

Remarque : Si vous effectuez la mise à niveau à partir d'une installation de HP OO 10.10 (et ultérieure) déjà configurée avec FIPS, vous devez répéter les étapes 4 et 5 de la section « Prérequis » dans " Norme FIPS (Federal Information Processing Standard)" , page 21, puis répéter les étapes ci-dessous.

Ouvrez le fichier **<oo_jre>\lib\security\java.security** dans un éditeur et réalisez les étapes suivantes :

1. Pour chaque fournisseur indiqué au format **security.provider.<nn>=<nom_du_fournisseur>**, augmentez le numéro d'ordre de préférence `<nn>` de 2 unités.

Par exemple, modifiez une entrée de fournisseur de :

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

en

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

2. Ajoutez un nouveau fournisseur par défaut (RSA JCE) Ajoutez le fournisseur suivant en haut de la liste des fournisseurs :

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

3. Ajoutez le fournisseur Java Secure Sockets Extension (JSSE) de RSA BSAFE SSL-J.

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

4. Copiez et collez la ligne suivante dans le fichier **java.security** pour confirmer que **RSA BSAFE** est utilisé dans un mode conforme à FIPS 140-2 :

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

   Vous pouvez copier cette ligne n'importe où dans le fichier **java.security**.

5. Étant donné que l'algorithme par défaut ECDRBG128 de DRGB n'est pas sûr (selon NIST), définissez la propriété de sécurité **com.rsa.crypto.default** sur **HMACDRBG**, en copiant la ligne suivante dans le fichier **java.security** :

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

   Vous pouvez copier cette ligne n'importe où dans le fichier **java.security**.

6. Enregistrez et fermez le fichier **java.security**.

Configurer le fichier encryption.properties et activez le mode FIPS.

Le fichier de propriétés de chiffrement de HP OO doit être mis à jour afin d'être conforme à la norme FIPS 140-2.

1. Sauvegardez le fichier **encryption.properties**, situé dans **<répertoire_ installation>\central\var\security**.

2. Ouvrez le fichier **encryption.properties** dans un éditeur de texte. Par exemple, modifiez le fichier suivant :

   C:\Program Files\Hewlett-Packard\HP Operations Orchestration\central\var\security\encryption.properties.

3. Localisez `keySize=128` et remplacez-le par `keySize=256`.

4. Localisez `secureHashAlgorithm=SHA1` et remplacez-le par `secureHashAlgorithm=SHA256`.
5. Localisez `FIPS140ModeEnabled=false` et remplacez-le par `FIPS140ModeEnabled=true`.

   Remarque : Si `FIPS140ModeEnabled=false` n'existe pas, ajoutez `FIPS140ModeEnabled=true` en tant que nouvelle ligne à la fin du fichier.

6. Enregistrez et fermez le fichier.

Créer un chiffrement HP OO conforme avec la norme FIPS

Pour créer ou remplacer le fichier de stockage de chiffrement de HP OO afin de le rendre conforme à FIPS, voir "Remplacement du chiffrement FIPS" , ci-dessous.

Remarque : AES possède trois longueurs de clé approuvés : 128/192/256 selon la publication NIST SP800-131A

Les algorithmes de hachage sécurisés suivants sont pris en charge dans la norme FIPS : SHA1, SHA256, SHA384, SHA512.

Remarque : Il est recommandé de modifier les mots de passe du keystore (ainsi que sa clé privée) et du truststore. Voir "Modification du mot de passe du keystore/truststore" , page 11.

Remarque : Il est recommandé de supprimer tous les certificats racine par défaut de l'autorité CA du truststore HP OO. (Le fichier client.truststore est situé dans <rép_ installation>/central/var/security.)

Remplacer le mot de passe de la base de données

Remplacez le mot de passe de la base de données tel que décrit dans "Modification du mot de passe de la base de données" , page 30.

Démarrer HP OO.

Démarrez HP OO comme décrit dans le manuel *HP OO 10.10 Installation Guide*.

Remplacement du chiffrement FIPS

HP OO, Central et RAS adhèrent à la norme Federal Information Processing Standard 140-2 (FIPS 140-2) qui définit les exigences techniques que les organismes fédéraux doivent respecter lorsqu'ils mettent en place des systèmes de sécurité à chiffrement pour la protection des données sensibles ou de valeur.

Après une installation directe de HP OO 10.10, vous avez la possibilité de modifier l'algorithme de chiffrement FIPS.

Remarque : Cette procédure concerne uniquement les nouvelles installations. Elle ne peut être exécutée après une mise à jour.

Modification de l'algorithme de chiffrement FIPS sur Central

1. Accédez au dossier **<rép_installation_Central>/var/security**.
2. Réalisez une sauvegarde du fichier **encryption_repository** et supprimez-le.
3. Accédez au dossier **<rép_installation_Central>/bin/**.
4. Exécutez le script **generate-keys**.

   Accédez au dossier **<rép_installation_Central>/var/security/encryption_repository**.

Modification des propriétés de chiffrement de RAS

Si l'installation du RAS se trouve dans un nouvel emplacement, il faudra réaliser toutes les étapes ci-dessous.

Remarque : Ces modifications sont uniquement valides si vous travaillez sur une nouvelle installation RAS après que vous avez modifié les propriétés de chiffrement de Central.

Pour modifier les propriétés de chiffrement du RAS :

1. Réalisez toutes les étapes décrites dans la section « Prérequis » de " Norme FIPS (Federal Information Processing Standard)" , page 21.
2. Réalisez toutes les étapes décrites dans la section « Configurer les propriétés du fichier de sécurité java » de "Configuration de HP OO pour respecter la norme FIPS 140-2" , page 23.
3. Copiez le fichier actuel **encryption.properties** depuis `<rép_installation>\ras\var\security` vers le dossier `<rép_installation>\ras\bin`
4. À l'aide d'un éditeur de texte, modifiez le contenu du fichier **encryption.properties** en fonction des besoins.

   Pour plus d'informations, voir « Configurer le fichier encryption.properties et activer le mode FIPS » dans "Configuration de HP OO pour respecter la norme FIPS 140-2" , page 23.
5. Enregistrez les modifications.
6. Ouvrez une invite de ligne de commande dans le dossier `<répertoire d'installation>\ras\bin`.

7. Exécutez **oosh.bat**.
8. Exécutez la commande OOShell : `replace-encryption --file encryption.properties`

   Remarque : Si vous aviez copié le fichier **encryption.properties** dans un autre dossier, confirmez que vous avez saisi l'emplacement correct dans la commande OOShell.

9. Redémarrez le service RAS.

Configuration des paramètres LWSSO

Quand vous installez HP OO 10.10, si vous décidez de réaliser la mise à niveau des paramètres LWSSO depuis HP OO 9.x, ces paramètres LWSSO seront migrés, mais LWSSO sera désactivé dans HP OO 10.10, même s'il avait été activé dans HP OO 9.x.

Quand vous activez LWSSO par la suite, il se peut que vous receviez des avertissements dans certains scénarios. Pour supprimer les avertissements du journal, suivez les étapes ci-après pour définir la propriété de l'URL de gestion à l'aide du nom de domaine complet.

- Quand Central et un RAS sont installés sur la même machine et que les paramètres LWSSO ont été activés, vous devez définir la propriété de l'URL de gestion à l'aide du nom de domaine complet.

  a. Arrêtez le processus RAS.

  b. Dans le fichier **ras/conf/ras-wrapper.conf**, changez

  ```
  wrapper.java.additional.<x>=-Dmgmt.url=<protocol>://localhost:<port>/oo
  ```

  en

  ```
  wrapper.java.additional.<x>=-
  Dmgmt.url=<protocol>://<FullyQualifiedDomainName>:<port>/oo
  ```

  c. Lancez le processus RAS.

- Quand le RAS est installé sur une machine différente de Central et que les paramètres LWSSO sont activés, vous devez définir l'URL de gestion de Central à l'aide du nom de domaine complet utilisé pendant l'installation RAS au lieu de l'adresse IP.

- Lors de la connexion d'une autre application à Central via LWSSO, vous devez désigner l'URL de gestion de Central à l'aide d'un nom de domaine complet.

  a. Arrêtez le processus Central.

  b. Dans le fichier **central/conf/central-wrapper.conf**, changez

  ```
  wrapper.java.additional.<x>=-Dmgmt.url=<protocol>://localhost:<port>/oo
  ```

en

```
wrapper.java.additional.<x>=-
Dmgmt.url=<protocol>://<FullyQualifiedDomainName>:<port>/oo
```

c. Lancez le processus Central.

Configuration de la stratégie XSS

HP OO utilise la protection XSS avec AntiSamy. La stratégie de protection AntiSamy autorise la plupart des éléments HTML et peut être utile si les utilisateurs soumettent des pages HTML complètes.

Cette stratégie peut être configurée dans une des stratégies prises en charge par AntiSamy. Pour plus d'informations, voir

https://www.owasp.org/index.php/Category:OWASP_AntiSamy_Project#Stage_2_-_Choosing_a_base_policy_file

La stratégie peut être configurée via une propriété de configuration système appelée `xss.policy`. Les valeurs possibles sont : antisamy (par défaut), antisamy-slashdot, antisamy-myspace, antisamy-ebay, antisamy-anythinggoes, antisamy-tinymc.

Pour vérifier la stratégie qui est configurée, rendez-vous sur **https://host/oo/reports/sysinfo** et recherchez le paramètre `xss.policy` dans la section **system configuration**.

La manière la plus simple de modifier la stratégie Slashdot par défaut est via l'utilitaire HP Operations Orchestration Shell.

1. Double-cliquez sur le fichier de commande `oosh.bat` afin de lancer l'utilitaire OOSH
2. Sur la ligne de commande, tapez par exemple :

```
ssc --url https://host/oo --key  xss.policy --value antisamy-cequetuveux
```

Pour plus d'informations sur l'utilitaire shell de HP Operations Orchestration, voir le manuel *HP Operations Orchestration Shell User Guide*.

Configuration de la localisation

Définition de la locale du système dans Central-wrapper.conf

Si votre système HP OO est localisé, il faudra définir les propriétés suivantes pour rendre compte de la locale du système dans le fichier **central-wrapper.conf** :

```
set.LANG=

set.LC_ALL=
```

```
set.LANGUAGE=

wrapper.java.additional.<x>=-Duser.language=

wrapper.java.additional.<x>=-Duser.country=
```

Par exemple, pour le japonais : `set.LANG=ja_JP` et `set.LC_ALL=ja_JP`

Configuration du système

Modification du mot de passe de la base de données

1. Si Central est opérationnel, arrêtez Central Service.
2. Exécutez le script encrypt-password avec l'option -e -p <mot de passe> où mot de passe est le mot de passe de la base de données.
3. Copiez le résultat qui devrait ressembler à ceci :

   ```
   ${ENCRYPTED}<des_caractères>.
   ```

4. Accédez au dossier **<Dossier d'installation de Central>/conf** et ouvrez le fichier **database.properties**.
5. Remplacez la valeur `db.password` par celle que vous avez copiée.

Modification de l'IP de la base de données

Cette section vous intéresse si vous devez configurer HP OO afin qu'il puisse fonctionner avec une autre instance de base de données. Tous les paramètres de la base de données tels que les informations d'identificatio, le nom du schéma, les tables, etc. doivent être identiques.

1. Modifiez le fichier **\HP Operations Orchestration\central\conf\database.properties.**
2. Recherchez le paramètre `jdbc.url`. Par exemple :

   ```
   jdbc.url=jdbc\:jtds\:sqlserver\://16.60.185.109\:1433/schemaName;sendStringP
   arametersAsUnicode\=true
   ```

3. Changez l'adresse IP\nom de domaine complet du serveur de base de données.
4. Enregistrez le fichier.
5. Redémarrez Central.

Réglage des niveaux de consignation

Il est possible de régler le niveau de détail des informations fournies dans le journal et ce, de manière séparée pour la consignation normale, le déploiement et l'exécution.

Les options sont les suivantes :

- `INFO` : consignation d'information par défaut
- `DEBUG` : consignation de plus d'informations

- `ERROR/WARNING` : consignation de moins d'informations

Pour régler le niveau de consignation :

1. Ouvrez le fichier **log4j.properties** (sous **/<installation oo>/central/conf/log4j.properties**).
2. Remplacez `INFO` par `DEBUG` ou `ERROR/WARNING` aux endroits suivants dans le fichier **log4j.properties**.

   Par exemple :

```
log.level=INFO
execution.log.level=DEBUG
deployment.log.level=DEBUG
```

Réglage de la planification des travaux Quartz

Dans le système HP OO, les travaux quartz sont exécutés à intervalle régulier pour la maintenance du système.

Chaque travail est exécuté pendant un certain temps et cette durée est répétée selon les intervalles définis. Voici des exemples de déclencheurs de travail :

Nom du déclencheur	Intervalle de répétition actuel	Effet
onRolling:OO_EXECUTION_STATES_Trigger	4,5 minutes	Rouler la table des états pour la purge
queueCleanerTrigger	1 minute	Purge des tables de file d'attente
queueRecoveryTrigger	2 minutes	Vérifie si le système doit récupérer
recoveryVersionTrigger	0,5 minute	Compteur de la version à utiliser pour la récupération
splitJoinTrigger	1 seconde	Jointure de scission terminée
onRolling:OO_EXECUTION_EVENTS_Trigger **Remarque :** Ce déclencheur est obsolète.	12 heures	Rouler la table des événements pour la purge

Si vous souhaitez optimiser la planification de ce travail pour améliorer les performances, procédez comme suit :

Remarque : Toute modification de la planification peut avoir une incidence importante sur le système. Consultez l'assistance HP avant de modifier ces déclencheurs.

1. Saisissez la page Jminix à l'aide de l'URL : **{OO_HOST}:{OO_PORT}/oo/jminix/**

   Remarque : Vous devez avoir l'autorisation **Gérer les paramètres système** pour saisir **jminix**.

2. Ouvrez l'onglet OO. Sous **MBeans**, vous trouverez l'opération **JobTriggersMBean**.
3. Utilisez cette opération, et saisissez les valeurs dans l'onglet de droite, en utilisant le nom du déclencheur que vous voulez changer. Utilisez exactement le même nom que la table, avec la nouvelle valeur de l'intervalle de répétition.

Les moments de déclenchements du travail sont modifiés.

Remarque : Le mécanisme de persistance des événements est obsolète (voir **onRolling:OO_EXECUTION_EVENTS_Trigger**). Vous pouvez configurer ce travail si vous utilisez Remote Debugger ou si vous avez activé l'indicateur **events.persistency**. Voir "Activation du mécanisme du journal des événements" , page suivante.

Modification de l'URL d'un Central/Répartiteur de charge du côte RAS

Bien que la meilleure pratique consiste à configurer l'URK d'un Central/répartiteur de charge via le programme d'installation, vous pouvez modifier le fichier **ras-wrapper.conf** si vous devez modifier l'URL quand le RAS a déjà été installé.

Par exemple, cette opération est nécessaire si vous avez installé un RAS par rapport à un Central/répartiteur de charge et que le nom de domaine complet de celui-ci a changé. Il faudra modifier l'URL de Central/répartiteur de charge stockée au niveau du RAS afin que le RAS puisse communiquer à nouveau avec le Central/répartiteur de charge.

1. Arrêtez le RAS.
2. Ouvrez le fichier **ras-wrapper.conf** qui se trouve dans **<dossier d'installation>\ras\conf**.
3. Modifiez l'URL de la ligne suivante :

   ```
   wrapper.java.additional.<x>=-Dmgmt.url=http://localhost:8080/oo
   ```

4. Redémarrez le RAS.

Activation du mécanisme du journal des événements

Le mécanisme du journal des événements est devenu obsolète dans HP OO 10.10 et sera supprimé dans une version future.

HP OO est déployé sans le mécanisme de journal des événements. Vous pouvez toutefois l'activer via l'indicateur **events.persistency**. Notez que dans les scénarios de mise en cluster et de renforcement des performances, il est recommandé de laisser cet indicateur désactivé.

Pour activer cet indicateur :

1. Arrêtez le processus et mettez à jour le fichier **wrapper.conf** sur chaque nœud (Central/RAS) du système.

   Sur Central, accédez à **<rép_installation>\central\conf\central-wrapper.conf**

   Sur RAS, accédez à **<rép_installation>\ras\conf\ras-wrapper.conf**

2. Dans le fichier **wrapper.conf**, recherchez le paramètre **-Devents.persistency** et modifiez sa valeur en **true**.

3. Redémarrez le processus.


Java™
COMPATIBLE