Installer un certificat dans Oracle REST Data Services (ORDS) :

Pour installer un certificat dans Oracle REST Data Services (ORDS), vous pouvez suivre ces étapes générales :

1. **Générer ou obtenir un certificat SSL/TLS :** Si vous n'avez pas déjà un certificat SSL/TLS, vous devez en générer un ou en obtenir un auprès d'une autorité de certification (CA).
2. **Convertir le certificat au bon format :** Assurez-vous que votre certificat est dans un format pris en charge par ORDS, comme PKCS12 (.p12 ou .pfx).
3. **Configurer ORDS pour utiliser le certificat :**
   * Ouvrez le fichier de configuration ORDS (**defaults.xml** généralement).
   * Recherchez la section de configuration SSL/TLS.
   * Configurez les chemins vers votre certificat (fichier keystore) et spécifiez le mot de passe si nécessaire.
4. **Redémarrez ORDS :** Une fois que vous avez configuré le certificat, redémarrez ORDS pour appliquer les modifications.

Voici un exemple de configuration dans **defaults.xml** :

xmlCopy code

<entry key="jetty.ssl.keyStore">/chemin/vers/votre/keystore.p12</entry>

<entry key="jetty.ssl.keyStorePassword">votre\_mot\_de\_passe</entry>

<entry key="jetty.ssl.keyStoreType">PKCS12</entry>

<entry key="jetty.ssl.keyStoreProvider">SunJSSE</entry>

Assurez-vous de remplacer **/chemin/vers/votre/keystore.p12** par le chemin réel vers votre keystore et **votre\_mot\_de\_passe** par le mot de passe de votre keystore

Pour convertir un certificat au format .crt en un format .p12 ou .pfx, vous pouvez suivre ces étapes en utilisant l'outil OpenSSL :

1. **Assurez-vous d'avoir OpenSSL installé :** Si vous ne l'avez pas déjà, vous pouvez télécharger et installer OpenSSL à partir de la source officielle ou via un gestionnaire de paquets adapté à votre système d'exploitation.
2. **Convertissez le certificat .crt en un fichier .pem :** Si votre certificat est déjà dans un format PEM, vous pouvez passer à l'étape suivante. Sinon, vous devez d'abord convertir le certificat .crt en un fichier .pem. Vous pouvez le faire en utilisant la commande suivante :

bashCopy code

openssl x509 -inform PEM -in Certificate.crt -out certificat.pem

Assurez-vous de remplacer **certificat.crt** par le nom de votre fichier de certificat.

1. **Convertissez le certificat PEM en un fichier PKCS12 (.p12 ou .pfx) :** Maintenant que vous avez un certificat au format PEM, vous pouvez le convertir en un fichier PKCS12 (.p12 ou .pfx) en utilisant la commande suivante :

bashCopy code

openssl pkcs12 -export -out certificat.p12 -inkey Private.key -in certificat.pem -certfile certificat-intermediaire.crt

openssl pkcs12 -export -out certificat.p12 -in certificat.pem -certfile certificat-intermediaire.crt

Assurez-vous de remplacer :

* + **certificat.pem** par le nom de votre fichier de certificat au format PEM.
  + **certificat.p12** par le nom que vous souhaitez donner au fichier PKCS12 de sortie.

Vous pouvez également inclure **-inkey cléprivée.key** si vous avez besoin d'ajouter une clé privée associée au certificat.

1. **Suivez les instructions :** Lorsque vous exécutez la commande de conversion en PKCS12, OpenSSL peut vous demander de fournir un mot de passe pour le nouveau fichier .p12. Assurez-vous de le spécifier et de le conserver en sécurité.

Une fois ces étapes terminées, vous devriez avoir un fichier .p12 (ou .pfx) contenant votre certificat, prêt à être utilisé avec ORDS.

Pour configurer Oracle REST Data Services (ORDS) pour utiliser le certificat SSL/TLS et redémarrer Apache sur CentOS 7.9, suivez ces étapes :

**Configuration d'ORDS pour utiliser le certificat :**

1. Localisez le fichier de configuration ORDS, généralement nommé **defaults.xml**. Vous pouvez le trouver dans le répertoire d'installation d'ORDS.
2. Ouvrez **defaults.xml** dans un éditeur de texte.
3. Recherchez la section de configuration SSL/TLS. Vous devriez voir des entrées telles que **jetty.ssl.keyStore**, **jetty.ssl.keyStorePassword**, etc.
4. Configurez les paramètres SSL/TLS en fonction de votre certificat. Voici un exemple de configuration :

xmlCopy code

<entry key="jetty.ssl.keyStore">/chemin/vers/votre/certificat.p12</entry> <entry key="jetty.ssl.keyStorePassword">votre\_mot\_de\_passe</entry> <entry key="jetty.ssl.keyStoreType">PKCS12</entry>

Assurez-vous de remplacer **/chemin/vers/votre/certificat.p12** par le chemin complet vers votre fichier de certificat et **votre\_mot\_de\_passe** par le mot de passe de votre certificat.

1. Enregistrez les modifications apportées à **defaults.xml**.

**Redémarrage d'Apache :**

1. Ouvrez un terminal sur votre serveur CentOS 7.9.
2. Redémarrez Apache en utilisant la commande suivante en tant qu'utilisateur root :

bashCopy code

sudo systemctl restart httpd

Cela redémarrera le service Apache, ce qui appliquera les modifications de configuration et permettra à ORDS d'utiliser le nouveau certificat.

Une fois ces étapes effectuées, votre serveur CentOS 7.9 avec ORDS devrait être configuré pour utiliser le certificat SSL/TLS et Apache devrait avoir été redémarré avec succès pour appliquer les modifications. Assurez-vous de tester votre configuration en accédant à vos ressources ORDS via HTTPS pour vérifier que le certificat est correctement appliqué.

Voici un exemple de fichier **defaults.xml** pour ORDS avec une configuration SSL/TLS de base :

xmlCopy code

<?xml version="1.0"?> <ords-config xmlns="http://xmlns.oracle.com/apex/ords/config" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.com/apex/ords/config https://apex.oracle.com/pls/apex/ords/r/ords\_config.xsd" xmlns:xs="http://www.w3.org/2001/XMLSchema">

<settings> <entry key="debug.debugger">false</entry>

<entry key="misc.compress">true</entry>

<entry key="jdbc.InitialLimit">5</entry>

<entry key="jdbc.MaxConnectionReuseCount">1000</entry>

<entry key="jdbc.MaxTotalConnections">25</entry>

<entry key="jdbc.MinLimit">0</entry>

<entry key="jdbc.statementTimeout">900</entry>

<entry key="misc.compression.threshold">0</entry>

<entry key="misc.defaultPage">apex</entry> <entry key="misc.escapePattern">/f?p=</entry>

<entry key="security.verifySSL">false</entry>

<entry key="sqldev.connections">1</entry>

<entry key="sqldev.db.hostname"></entry>

<entry key="sqldev.db.password"></entry>

<entry key="sqldev.db.port"></entry>

<entry key="sqldev.db.servicename"></entry>

<entry key="sqldev.db.sid"></entry>

<entry key="sqldev.db.username"></entry>

<entry key="sqldev.enabled">false</entry>

<entry key="studio.beautify">true</entry>

<entry key="studio.editor.theme">ambiance</entry>

<entry key="studio.enabled">false</entry>

<entry key="studio.oracle.defaultDBConn">false</entry>

<entry key="studio.oracle.enabled">false</entry>

<entry key="studio.oracle.workspace"></entry>

<entry key="studio.schema.enabled">false</entry>

<entry key="studio.standalone.enabled">false</entry>

<entry key="debugger.enabled">false</entry>

<entry key="debugger.printPayload">true</entry>

<entry key="jetty.http.port">8080</entry>

<entry key="jetty.ssl.port">8443</entry>

<entry key="jetty.ssl.keyStore"></entry>

<entry key="jetty.ssl.keyStorePassword"></entry>

<entry key="jetty.ssl.keyStoreType"></entry>

<entry key="jetty.ssl.keyStoreProvider"></entry>

<entry key="jdbc.connectionsDefaultReadOnly">false</entry>

<entry key="jdbc.default.autoCommit">true</entry>

<entry key="jdbc.default.connectionTimeout">30</entry> <entry key="jdbc.default.idleTimeout">600</entry>

<entry key="jdbc.default.maxLifetime">1800</entry>

<entry key="jdbc.default.minimumIdle">1</entry>

<entry key="jdbc.default.poolName">Oracle REST Data Services</entry>

<entry key="jdbc.default.validateConnectionOnBorrow">true</entry>

<entry key="jdbc.maxConnections">25</entry>

<entry key="jdbc.statementTimeout">900</entry>

<entry key="jdbc.threadPoolSize">25</entry>

<entry key="jdbc.validateQuery">select 1 from dual</entry>

<entry key="jdbc.fetchSize">100</entry>

<entry key="jdbc.default.autoCommit">true</entry>

<entry key="misc.defaultPage">apex</entry>

<entry key="misc.escapePattern">/f?p=</entry>

<entry key="misc.compress">true</entry> </settings> </ords-config>

Vous devez remplacer les valeurs vides ou par défaut par vos propres valeurs pour la configuration SSL/TLS. Voici les entrées de configuration SSL/TLS que vous devez remplir :

* **jetty.ssl.keyStore**: Chemin complet vers votre fichier de keystore contenant le certificat SSL/TLS.
* **jetty.ssl.keyStorePassword**: Mot de passe pour accéder au keystore.
* **jetty.ssl.keyStoreType**: Type de keystore (par exemple, **PKCS12**).
* **jetty.ssl.keyStoreProvider**: Fournisseur de keystore (facultatif).

Remplacez ces valeurs par celles correspondant à votre configuration SSL/TLS spécifique. Une fois que vous avez mis à jour le fichier **defaults.xml** avec vos informations, enregistrez-le et redémarrez ORDS pour appliquer les modifications.

Pour redémarrer ords :

java -jar ords.war standalone