Documentum Server - How to setup SSL HTTPS for Java Method Server

Article ID:KB10912939

Applies to

Documentum Server 16.4, 7.3

Summary

In Documentum Server, you need to enable HTTPS SSL with Java Method Server (JMS) if you want to prevent any network data security breaches between DFC client application and JMS server. This article addresses how to configure SSL for JMS server.
This issue occurs in (but may not be limited to):
Documentum Server 7.3, 16.4

Resolution

You can follow the below procedure to enable HTTPS SSL connections between JMS and Content Server.

1. Create a certificate keystore named jms.keystore using the Java keytool utility. The keystore must be in the JKS format. A new certificate must be generated by exporting the keystore file.
   You must first set the JAVA_HOME environment variable, for example JAVA_HOME=C:\Documentum\java64\1.8.0_77
   Linux:
   $JAVA_HOME/bin/keytool -genkey -alias jms -keyalg RSA -keystore jms.keystore -dname CN=dctm1,OU=ecd,O=opentext,L=TORONTO,ST=ON,C=CA -keypass changeit -storepass changeit
   Windows:
   %JAVA_HOME%/bin/keytool -genkey -alias jms -keyalg RSA -keystore jms.keystore -dname CN=dctm1,OU=ecd,O=opentext,L=TORONTO,ST=ON,C=CA -keypass changeit -storepass changeit
2. Export the certificate with the following command.
   Linux :
   $JAVA_HOME/bin/keytool -export -alias jms -keystore jms.keystore -file jmscerts.cer -storepass changeit
   Certificate stored in file <jmscerts.cer>
   Windows:
   %JAVA_HOME%/bin/keytool -export -alias jms -keystore jms.keystore -file jmscerts.cer -storepass changeit
   Certificate stored in file <jmscerts.cer>
3. Import the certificate to JDK cacerts keystore.
   Linux:
   $JAVA_HOME/bin/keytool -import -trustcacerts -alias jms -keystore $JAVA_HOME/jre/lib/security/cacerts -file jmscerts.cer -storepass changeit -noprompt
   Certificate was added to keystore
   Windows:
   %JAVA_HOME%/bin/keytool -import -trustcacerts -alias jms -keystore %JAVA_HOME%/jre/lib/security/cacerts -file jmscerts.cer -storepass changeit -noprompt
   Certificate was added to keystore
4. Enable an HTTPS port for the JMS instance in wildfly9.01.

   1. Open $WILDFLY_HOME\server\DctmServer_MethodServer\configuration\standalone.xml,Add the following lines under <security-realms>.

      ```
      <security-realm name="JMSRealm">
      <server-identities>
      <ssl>
      <keystore path="C:/Documentum/keystore/jms.keystore" keystore-password="changeit " alias="jms" key-
      </server-identities>
      </security-realm>
      ```

2. Add the following after <server name = "default-server">.
   ```
   <http-listener name="default" socket-binding="http" redirect-socket="https"/>
   <https-listener name="https" socket-binding="https" security-realm="JMSRealm" enabled-cipher-suites
   ```

3. Use the cipher-suite values depending on the following.
   If you want to configure JMS in non-anonymous SSL mode, then cipher suite parameters are:
   TLS_RSA_WITH_AES_128_CBC_SHA,AES256-SHA256,DHE-DSS-AES256-SHA256,DHE-DSS-AES256-SHA.
   To use AES256 and SHA256 algorithms, ensure that you have the JCE Unlimited Strength Jurisdiction policy files.
   If you want to configure JMS in dual mode (anonymous and non-anonymous), then cipher suite parameter is:
   LS_DH_anon_WITH_AES_128_CBC_SHA,SSL_DH_anon_WITH_3DES_EDE_CBC_SHA,TLS_RSA_WITH_AES_128_CBC_SHA.

5. Dump the dm_jms_config object and change the base_uri to point to the new URL.
   ```
   API> retrieve,c,dm_jms_config ......080004d380000b1a
   API> set,c,l,base_uri[0]
   API> https://<host>:9082/DmMethods/servlet/DoMethod
   API> save,c,l
   ```

6. Start the repository and Java Method Server..

Keywords

SSL, HTTPS, JMS