2.7 MCH Advanced Configurations

Back to main MCH documentation page: Documentation

Advanced Configurations

This section describes some specific, more advanced configuration for certain MCH components.

- Configure MQ SSL connectivity

Configure MQ SSL connectivity

Following instructions are tested with mch1.1 and mq series 7.5 installed on windows

These instructions should **not** be used as guidelines for naming conventions, security/authentication

1. Install MQ Series
2. Create a new user group and a new user
   group name - mquser
   user name - mchmquser

3. Start MQExplorer - on windows it can be found at INSTALL_LOC\bin\MQExplorer.exe
4. Create a new Queue Manager - see

   Rt click Queue Managers > New > Queue Manager
   Create Queue Manager window comes up
   Fill the form with below values
   Queue manager name - SSLQM
   Dead letter queue - SYSTEM.DEAD.LETTER.QUEUE

   click Next

   click Next

   check the tick box - Create server connection channel

   click Finish

5. Create a new Queue

   Open SSLQM

   Rt click Queues > New > LocalQueue

   Enter Name - MCH_MESSAGE_QUEUE

   click Finish

6. Create a new Server-connenction Channel

   Rt click Channels > New > Server-connenction Channel

   Enter Name - MCH.DEF.SVRCONN

   click Finish

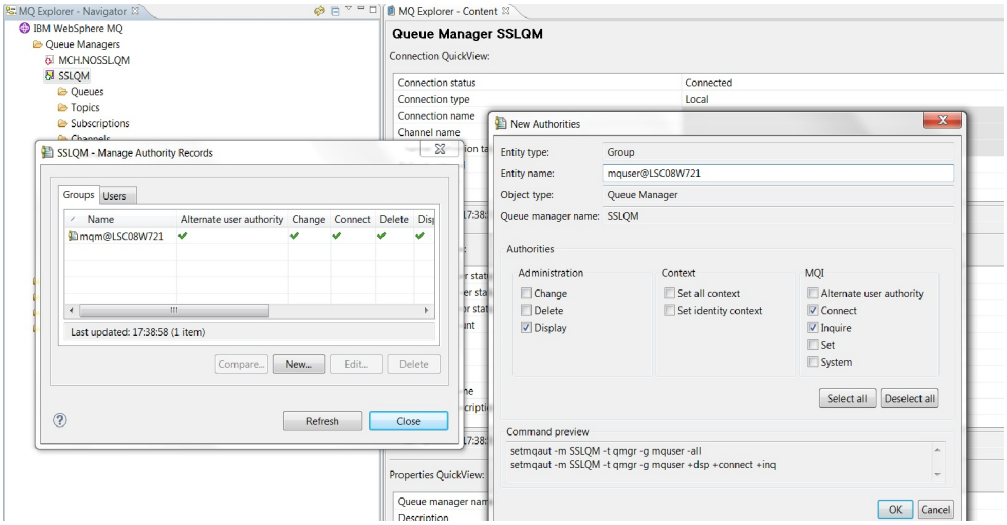
7. Grant permissions to mquser group

   Rt click SSLQM > Object Authorities > Manage Queue Manager Authority Records

   Manage Authority Records window comes up

   click New button - New Authorities window pops up

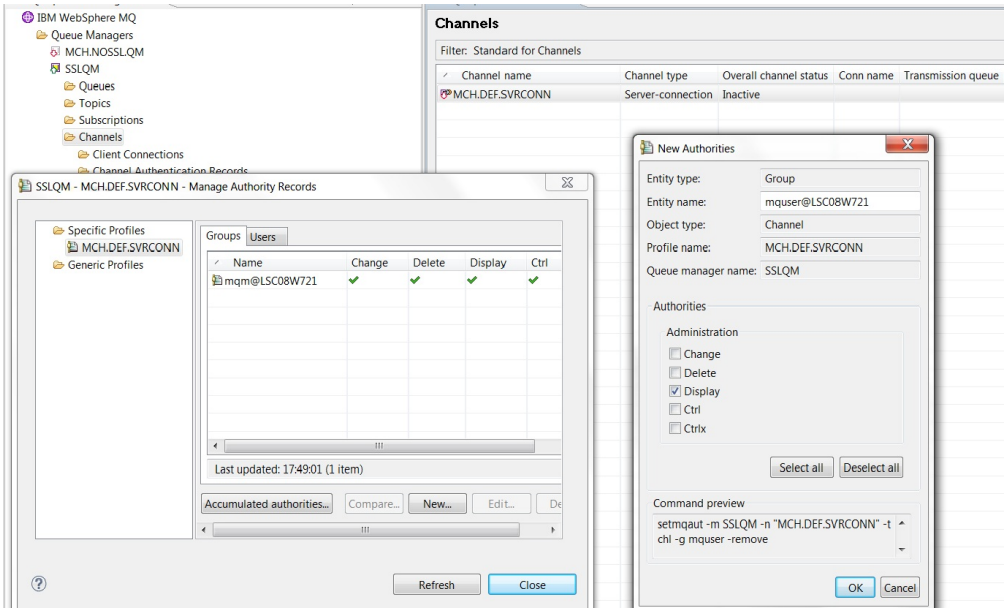
   Give Display , Connect, Inquire permissions to mquser group created in step 2



Click on Channels and open MQ Exlporer - content window

Rt click MCH.DEF.SVRCONN > Object Authorities > Manage Authority Records > Open Specific Profiles > click MCH.DEF.SVRCONN > click New button > New Authorities window opens up

Give Display permissions to mquser group



Click on Queues > MQ Exlporer - content > Rt click MCH_MESSAGE_QUEUE > Object Authorities > Manage Authority Records > Open Specific Profiles > MCH_MESSAGE_QUEUE > click New button

Give All permissions to mquser group

8. Create Channel Authentication Record

Expand Channels > Rt click Channel Authentication Records > New > Channel Authentication Record > New Channel Authentication Record window pops up

click Next > click Next > enter MCH.DEF.SVRCONN in Channel Prfile text box > click Show matching channels > select channel > click Next > Enter SSL/TSL DN pattern as **CN=www.misys.com**

> click Next > select Fixed user id> enter as user id - mchmquser > Finish

9. Create keystore and trust store with self signed certificates

Mq expects certificate alias to have below naming conventions

ibmwebspheremq<<QUEUEMANAGER_IN_LOWER_CASE>>

ibmwebspheremq<<USER_NAME_LOWER_CASE>>

In our case

queuemanager - sslqm

user - mchmquser

create a temporary folder for storing sertificates

open command prompt

cd to above created folder

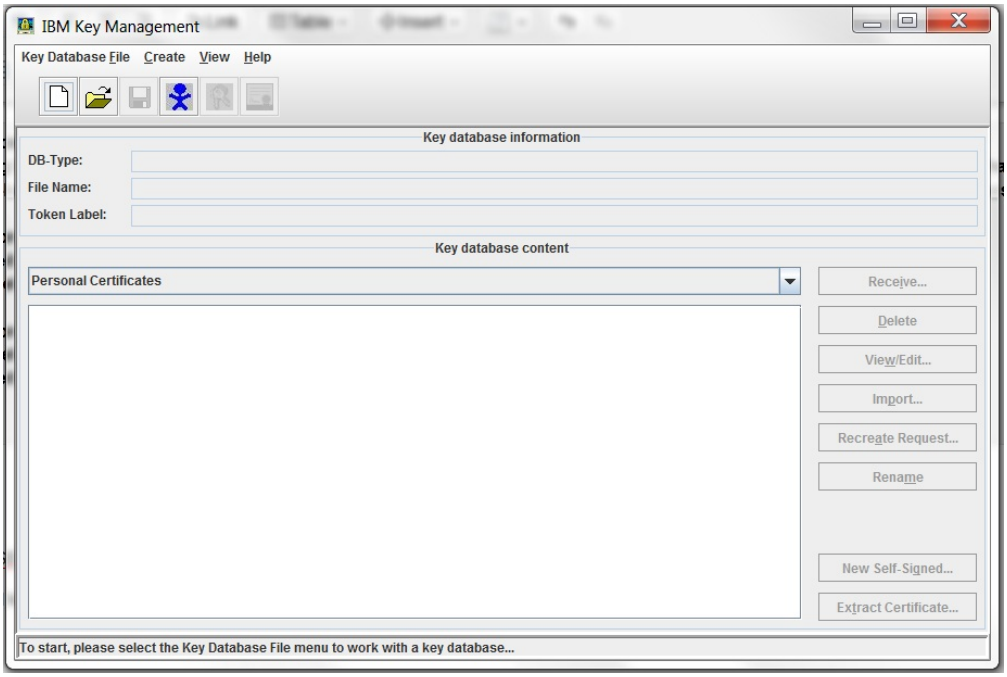
execute below commands

```
# Create the server and application client key stores and certificates
keytool -genkeypair -alias ibmwebspheremqsslqm -keyalg RSA -keysize 1024 -dname "CN=www.misys.com, OU=mch, O=cmf, L=london, ST=paddington, C=UK" -keypass changeit -storepass changeit -keystore server.jks
keytool -genkeypair -alias ibmwebspheremqmchmquser -keyalg RSA -keysize 1024 -dname "CN=www.misys.com, OU=mch, O=cmf, L=london, ST=paddington, C=UK" -keypass changeit -storepass changeit -keystore client.jks

# Copy the client's public certificate to the server's keystore
keytool -exportcert -keystore client.jks -storepass changeit -file client-public.cer -alias ibmwebspheremqmchmquser
keytool -importcert -keystore server.jks -storepass changeit -file client-public.cer -alias ibmwebspheremqmchmquser -noprompt

# Copy the server's public certificate to the client's keystore
keytool -exportcert -keystore server.jks -storepass changeit -file server-public.cer -alias ibmwebspheremqsslqm
keytool -importcert -keystore client.jks -storepass changeit -file server-public.cer -alias ibmwebspheremqsslqm -noprompt
```

10. Create Mq keydb file

Rt Click SSLQM (our Queue Manager) > Properties > select SSL in popped up window > note the path of SSL key repository, for convenience let us call it KEY_REPO > Cancel

Rt Click IBM Websphere MQ > Manage SSL certificates > IBM key management application comes up



KEY_REPO will have repo name("key" in this case) appended to folder where certificates are stored.

let the folder be called REPO_PATH

Ctrl-N > enter in location text box - REPO_PATH (D:\skakani\servers\wmq7.5\qmgrs\SSLQM\ssl\ on my system) > click Ok > enter "changeit" as password > select check box to stash password

click Import > change key file type to JKS > choose temp folder created in step 9 as location > enter "server.jks" as File name > click Ok

Import all certificates present in JKS file

11. Configure SSL cipher on channel

In MQ Explore select Channels > Rt click on MCH.DEF.SVRCONN channel (in MQ Exlporer - content view) > select Properties > select SSL > select RC4_SHA_US > click Ok

refer http://publib.boulder.ibm.com/infocenter/wmqv6/v6r0/index.jsp?topic=%2Fcom.ibm.mq.csqzaw.doc%2Fuj34740_.htm for valid values and their corresponding cipher suite names

12. Test MCH

set JVM arguments

-Djavax.net.ssl.trustStore=client.jks(filecreated in step 9)

-Djavax.net.ssl.trustStorePassword=changeitdirector.properties

-Djavax.net.ssl.keyStore=client.jks

-Djavax.net.ssl.keyStorePassword=changeit

-Djavax.net.debug=ssl:handshake

samples mch configuration - director.properties

if there are no certificate validation errors, no authentication errors then MQ SSL installation is successful.

See MQ error log located in MQ_INSTALL_FOLDER\Qmgrs\SSLQM\errors for debugging purpose.