<https://www.ibm.com/mysupport/s/question/0D50z000062l4HICAY/how-do-i-configure-ssl-tls-between-java-client-and-mq-queue-manager?language=en_US>

<https://www.ibm.com/developerworks/websphere/library/techarticles/0510_fehners/0510_fehners.html>

1. Set server side keystore

1.1. Go to: qmgrs/<qmanger name>/ssl directory

1.2. Create keystore repository:

runmqckm -keydb -create -db key.kdb -type cms -pw passw0rd -stash

1.3 Create personal certificate (qm1.req) in repository:

runmqckm -certreq -create -db key.kdb -pw passw0rd -label

ibmwebspheremqqm1 -dn "CN=qm1,O=IBM,C=US" -size 2048 -file

qm1.req -sig\_alg SHA256\_WITH\_RSA

Note qmgr personal cerificate label naming: **<ibmwebspheremq> + <qmanagername>** all lowercase.

1.4 Send the file qm1.req to CA (Citi) to get signed

1.5 CA provides signed cert request + CA signer certificates, which need to be added to the keystore, and exchanged to remote partner (java client)

1.6 Add CA root and any CA intermediate signer certificates to the keystore:

runmqckm -cert -add -db key.kdb -pw passw0rd -label label -file

1.7 Receive signed cert request file (from CA) to the keystore :this merges existing private key in keystore with the signed cert request from CA:

runmqckm -cert -receive -file -db key.kdb -pw passw0rd

1.8 By this step, keystore should contain:

Personal signed certificate (1.7)

All CA singer certificates (CA root and any CA intermediate signer certificates) (1.6)

List personal certs:

runmqckm -cert -list personal -db key.kdb -pw passw0rd

List ca-signer certs

runmqckm -cert -list ca -db key.kdb -pw passw0rd

2. Set java client keystore:

2.1 In some directory, create keystore repository:

runmqckm -keydb -create -db javaclient.jks -type jks -pw passw0rd

2.2 Create personal certificate (client.req) in repository:

runmqckm -certreq -create -db javaclient.jks -type jks -pw passw0rd -

label javaclientcert -dn "CN=client,O=IBM,C=US" -size 2048 -file

client.req -sig\_alg SHA256\_WITH\_RSA

Note: Unlike creating a Queue Manager personal Certificate, there is no restriction on the Key Label that must be used.

2.3 Send file client.req to CA (Citi) to get signed

2.4 CA provides signed cert request + CA signer certificates, which need to be added to the keystore, and exchanged to remote partner (mq queue manager)

2.5 Add CA root and any CA intermediate signer certificates to the keystore:

runmqckm -cert -add -db javaclient.jks -pw passw0rd -type jks -label -file

2.6 Receive signed cert request file (from CA) to the keystore :this merges existing private key in keystore with the signed cert request from CA:

runmqckm -cert -receive -file -db javaclient.jks -pw passw0rd -type jks

2.7 By this step, keystore should contain:

Personal signed certificate (1.7)

All CA singer certificates (CA root and any CA intermediate signer certificates) (1.6)

List personal certs:

runmqckm -cert -list personal -db key.kdb -pw passw0rd

List ca-signer certs

runmqckm -cert -list ca -db key.kdb -pw passw0rd

3. Exchange certificates

3.1 Exchange CA Singer (Intermediate and Root certificates) between qmgr and Java client’s keystore

Notes:

Java requires a truststore for this

You can use the same .jks file for both the truststore and keystore

… or you can make separate keystore and truststore .jks files

3.2 At this point, the keystores and certificates should be correct