

```
rem =====
rem   One possible approach to creating the SSL key repositories
rem   for four queue managers
rem
rem   Using a certificate authority to sign certificate requests,
rem   where certreqs are created by queue manager admins.
rem
rem
rem   The commands can be run (as written here) on a single machine
rem   and then the completed key repositories moved into locations
rem   accessible by the queue managers
rem
rem   Dale Lane (http://hursleyonwmg.wordpress.com/)
rem =====

REM *****
REM *** ENVIRONMENT
REM *****
rem *** path for WebSphere MQ
set MQBASE=C:\Program Files\IBM\WebSphere MQ

set PASSWORD=passw0rd

REM *** command name (gsk7cmd on UNIX, runmqckm on Windows)
set GSK7CMD=runmqckm

REM -----
REM   On Windows, using runmqckm acts as a wrapper for the GSKit command
REM   gsk7cmd in the correct environment. Using runmqckm means you do
REM   not need the following commands.
REM   Alternatively, you could use gsk7cmd, and use the following two
REM   commands to set the environment manually.
REM -----
rem *** Set the path to the GSKit programs used to create the repository ***
rem set PATH=%PATH%;C:\Program Files\IBM\gsk7\bin
rem *** Set the path to the JRE installed by WMQ for GSKit ***
rem set JAVA_HOME=%MQBASE%\gskit\jre
REM -----

REM *****
REM   lowercase!
REM   when used in label names, we need
REM   queue manager names in lowercase,
REM   regardless of the case of the qmgr
REM   names
REM *****
set QMGR1NAME=qmgr1
set QMGR2NAME=qmgr2
set QMGR3NAME=qmgr3
set QMGR4NAME=qmgr4
```

```
REM *****
REM create repositories for use by queue managers to store keys
REM
REM these should be moved to the SSL directory of the relevant
REM queue manager, or the queue manager SSLKEYR attribute
REM altered to point at this location
REM *****
%GSK7CMD% -keydb -create -db qmgr1.kdb -pw %PASSWORD% -type cms -stash
%GSK7CMD% -keydb -create -db qmgr2.kdb -pw %PASSWORD% -type cms -stash
%GSK7CMD% -keydb -create -db qmgr3.kdb -pw %PASSWORD% -type cms -stash
%GSK7CMD% -keydb -create -db qmgr4.kdb -pw %PASSWORD% -type cms -stash

REM *****
rem CERTIFICATE REQUESTS FOR QUEUE MANAGERS
REM *****
rem *** Create a request for a certificate to be signed for QMGR1 ***
%GSK7CMD% -certreq -create -db qmgr1.kdb -pw %PASSWORD% -label ibmwebspheremq%QMGR1NAME% -dn "CN=Qmgr1,O=IBM,OU=Hursley blog,L=Hursley,C=UK"
-file qmgr1req.arm
rem *** Create a request for a certificate to be signed for QMGR2 ***
%GSK7CMD% -certreq -create -db qmgr2.kdb -pw %PASSWORD% -label ibmwebspheremq%QMGR2NAME% -dn "CN=Qmgr2,O=IBM,OU=Hursley blog,L=Hursley,C=UK"
-file qmgr2req.arm
rem *** Create a request for a certificate to be signed for QMGR3 ***
%GSK7CMD% -certreq -create -db qmgr3.kdb -pw %PASSWORD% -label ibmwebspheremq%QMGR3NAME% -dn "CN=Qmgr3,O=IBM,OU=Hursley blog,L=Hursley,C=UK"
-file qmgr3req.arm
rem *** Create a request for a certificate to be signed for QMGR4 ***
%GSK7CMD% -certreq -create -db qmgr4.kdb -pw %PASSWORD% -label ibmwebspheremq%QMGR4NAME% -dn "CN=Qmgr4,O=IBM,OU=Hursley blog,L=Hursley,C=UK"
-file qmgr4req.arm
```

```
REM *****
REM *** Create CA root certificate for signing
REM *****
rem *** Create a repository for the certificate authority ***
%GSK7CMD% -keydb -create -db ca_key.kdb -pw %PASSWORD% -type cms
rem *** Create a self-signed certificate which will be the signing certificate authority ***
%GSK7CMD% -cert -create -db ca_key.kdb -pw %PASSWORD% -label "CA_Cert" -dn "CN=WMQ Blog Certificate Authority,O=IBM,OU=Hursley
blog,L=Hursley,C=UK"
rem *** Export the public key for the self-signed certificate
%GSK7CMD% -cert -extract -db ca_key.kdb -pw %PASSWORD% -label "CA_Cert" -target ca_cert.arm

rem  at this point, you would transfer the certificate requests
rem  qmgr1req.arm qmgr2req.arm qmgr3req.arm qmgr4req.arm
rem  to the machine with the CA key repository

REM *****
REM *** Sign certificate requests with CA
REM *****
rem *** Sign the certificate request using the previously created CA certificate ***
%GSK7CMD% -cert -sign -db ca_key.kdb -pw %PASSWORD% -label "CA_Cert" -file qmgr1req.arm -target qmgr1signed.arm -expire 364
rem *** Sign the certificate request using the previously created CA certificate ***
%GSK7CMD% -cert -sign -db ca_key.kdb -pw %PASSWORD% -label "CA_Cert" -file qmgr2req.arm -target qmgr2signed.arm -expire 364
rem *** Sign the certificate request using the previously created CA certificate ***
%GSK7CMD% -cert -sign -db ca_key.kdb -pw %PASSWORD% -label "CA_Cert" -file qmgr3req.arm -target qmgr3signed.arm -expire 364
rem *** Sign the certificate request using the previously created CA certificate ***
%GSK7CMD% -cert -sign -db ca_key.kdb -pw %PASSWORD% -label "CA_Cert" -file qmgr4req.arm -target qmgr4signed.arm -expire 364

rem  at this point, you would transfer the signed certificate requests
rem  qmgr1signed.arm qmgr2signed.arm qmgr3signed.arm qmgr4signed.arm
rem  to the machines with their key repositories

rem  you also need to transfer the public key for the signing certificate authority
rem  ca_cert.arm
rem  to the machines with their key repositories
```

```
REM *****
REM *** Add certificate authority public key to qmgr key repositories
REM *****
%GSK7CMD% -cert -add -db qmgr1.kdb -pw %PASSWORD% -label "CA_Cert" -file ca_cert.arm
%GSK7CMD% -cert -add -db qmgr2.kdb -pw %PASSWORD% -label "CA_Cert" -file ca_cert.arm
%GSK7CMD% -cert -add -db qmgr3.kdb -pw %PASSWORD% -label "CA_Cert" -file ca_cert.arm
%GSK7CMD% -cert -add -db qmgr4.kdb -pw %PASSWORD% -label "CA_Cert" -file ca_cert.arm
```

```
REM *****
REM *** Receive signed certificate requests back in qmgr key repositories
REM *****
rem *** Receive the signed certificate for QMGR1 back into the repository ***
%GSK7CMD% -cert -receive -db qmgr1.kdb -pw %PASSWORD% -file qmgr1signed.arm
rem *** Receive the signed certificate for QMGR2 back into the repository ***
%GSK7CMD% -cert -receive -db qmgr2.kdb -pw %PASSWORD% -file qmgr2signed.arm
rem *** Receive the signed certificate for QMGR3 back into the repository ***
%GSK7CMD% -cert -receive -db qmgr3.kdb -pw %PASSWORD% -file qmgr3signed.arm
rem *** Receive the signed certificate for QMGR4 back into the repository ***
%GSK7CMD% -cert -receive -db qmgr4.kdb -pw %PASSWORD% -file qmgr4signed.arm
```

```
REM -----
REM QUEUE MANAGER KEY REPOSITORIES
REM   qmgr1.kdb (and associated stash file qmgr1.sth)
REM   qmgr2.kdb (and associated stash file qmgr2.sth)
REM   qmgr3.kdb (and associated stash file qmgr3.sth)
REM   qmgr4.kdb (and associated stash file qmgr4.sth)
REM   are now ready for use by the queue managers
REM -----
REM CA SIGNING AUTHORITY KEY REPOSITORY
REM   ca_key.kdb (and associated stash file ca_key.sth)
REM   is ready for signing any future qmgr certificates
REM -----
```

```
REM *****
REM delete files used to create certificates
REM *****
del ca_cert.arm
del qmgr1signed.arm qmgr2signed.arm qmgr3signed.arm qmgr4signed.arm
del qmgr1req.arm   qmgr2req.arm   qmgr3req.arm   qmgr4req.arm
```

```
rem =====
rem   END
rem =====
```