MQ Security with CA ACF2

Roger Lacroix
roger.lacroix@capitalware.biz
http://www.capitalware.biz

Overview

- This session is focused on CA ACF2 commands that enable security for WebSphere MQ for z/OS
- In all of the examples, the queue manager name used is 'CSQ1'

OS/390 Services for Access Control

- WebSphere MQ uses *Switch Profiles* to implement security switches. The queue manager will made a SAF (System Access Facility) call to check for the existences of the switch profiles. Normal SAF calls check for the access rights that a particular UserID has to the profile.
- All switch profiles are managed by the external security manager (i.e. ACF2) rather than by WebSphere MQ (vs distributed platforms which have OAM).

Gotchas

- The ***subsys*** value can be masked in CA ACF2 resource rules
- Switch profiles are supported through use of GSO SAFDEF records
- Resource grouping cannot be used for resource MQQUEUE or any other validation invoked through RACROUTE FASTAUTH
- REFRESH SECURITY is required for MQADMIN RESLEVEL changes
- Audit records are reported as ACFRPTRV trace records

CA ACF2 UserIds

- To create the required 2 userids for each queue manager (MSTR & CHIN), the following CA ACF2 records need to be added to ACF2:

```
INSERT CSQ1MSTR MUSASS NON-CNCL STC
INSERT CSQ1CHIN MUSASS NON-CNCL STC
```

Enabling WebSphere MQ Security

- On z/OS, security is NOT enabled when WebSphere MQ is installed. To activate WebSphere MQ security checking, you need to issue the following ACF2 commands:

```
INSERT SAFDEF.CSQ1 ID(CSQ1) FUNCRET(8) -
RETCODE(4) MODE(IGNORE) -
RACROUTE(REQUEST=EXTRACT,CLASS=MQADMIN) REP
```

CA ACF2 and RESLEVEL

- The RESLEVEL resource specifies the level of the WebSphere MQ security in effect for a job.
- The level of access granted to the MQADMIN resource named 'CSQ1.RESLEVEL' is used to determine the level of MQ security for that user or CICS region.
- Giving READ or no-access to the 'CSQ1.RESLEVEL' resource means the user or CICS region will follow full/normal MQ security checking. Giving ALTER authority to this resource on the other hand will exempt the user or CICS region from all further MQ security checking.

CA ACF2 and RESLEVEL

- This may result in a problem for CICS regions in which the region logonid has NON-CNCL specified.
- NON-CNCL will always indicate ALTER authority to the RESLEVEL check.
- MQ processing will then bypass security checking for that region.

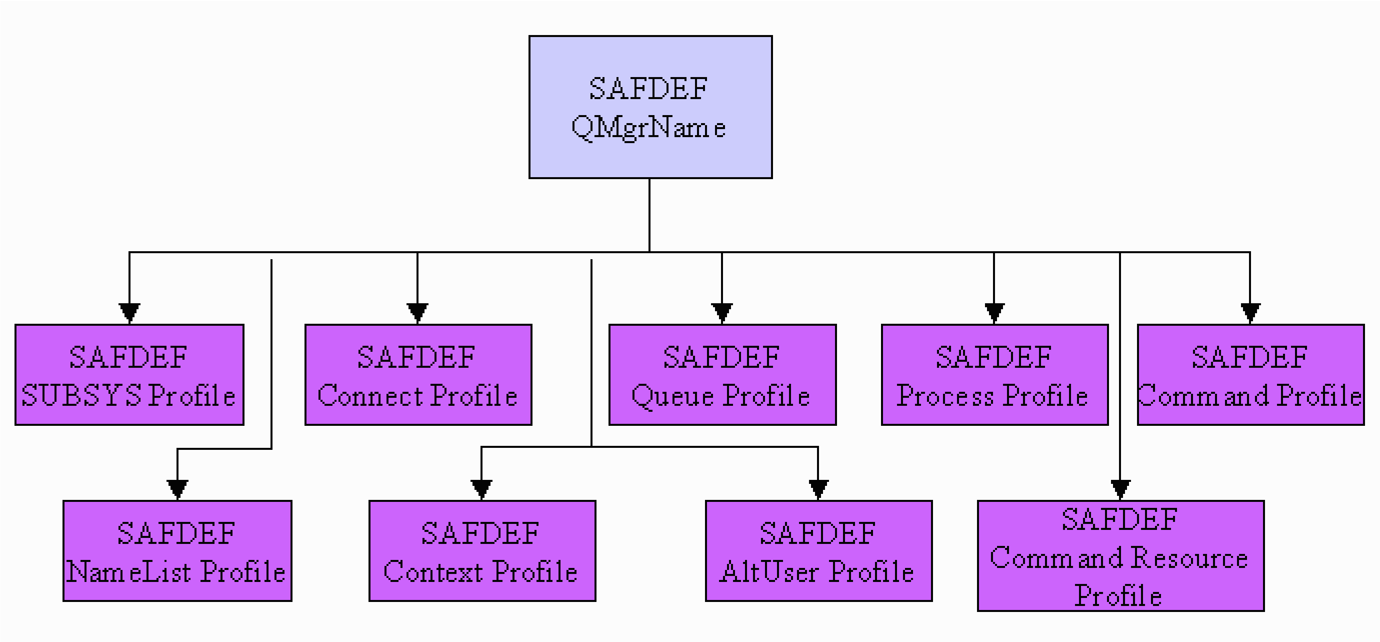
CA ACF2 and RESLEVEL

To avoid this, you can insert a SAFDEF record that will override the validation return codes for the RESLEVEL check.

```
INSERT SAFDEF.mqmresl ID(mqmresl) MODE(IGNORE) JOB(cicsjob) -
RETCODE(0) FUNCRET(20) FUNCRSN(n) -
RACROUTE(REQUEST=AUTH,CLASS=MQADMIN,ENTITYX=CSQ1.RESLEVEL) REP

where 'n' is:
      0 - no access
      4 - READ access
      8 - UPDATE access
      12 - CONTROL access
      16 - ALTER access
Note: 'cicsjob' represents the job(s) for which this access is
desired. Omit this parameter if all jobs require the same
access level.
```

ACF2 SAFDEF Layout Overview



SAFDEF Gotchas

- Each switch profile that WebSphere MQ detects turns on the checking for that type of resource. Switch profiles are activated during startup of the queue manager.
- These SAFDEF records return a NOT FOUND condition to MQM. The negative logic is that if no.subsys.security is not found, subsys security is active; if no.connect.checks is not found, connection security is active.

Note: This technique is completely contradictory to how RACF performs its switch profile checking.

Switch Profile Names

Switch Profile Name	Type of resource checking
ssid.NO.SUBSYS.SECURITY	Subsystem security
ssid.NO.CONNECT.CHECKS	Connection security
ssid.NO.QUEUE.CHECKS	Queue security
ssid.NO.PROCESS.CHECKS	Process security
ssid.NO.NLIST.CHECKS	Namelist security
ssid.NO.CONTEXT.CHECKS	Context security
ssid.NO.ALTERNATE.USER.CHECKS	Alternate user security
ssid.NO.CMD.CHECKS	Command security
ssid.NO.CMD.RESC.CHECKS	Command resource security
ssid.NO.TOPICS.CHECKS	Topic security

The 'ssid' refers to the queue manager name.

MQ Security Startup Checklist

On startup of a queue manager or someone refreshing the MQADMIN class (via REFRESH SECURITY command), the queue manager first checks the status of ACF2 and the MQADMIN class. The queue manager will set the subsystem security flag off if it discovers one of these conditions:

- ACF2 is inactive or not installed
- The MQADMIN class is not defined
- The MQADMIN class has not been activated

MQ Security Startup Checklist

If both ACF2 and the MQADMIN class are active then:

- The queue manager checks the MQADMIN class to see if any of the switch profiles have been defined.
- The queue manager first checks for the ssid.NO.SUBSYS.SECURITY profile. If this profile is defined, the queue manager sets the subsystem security flag on and then searches for any of the other switch profiles. If this profile is not defined, the queue manager sets its subsystem security flag off, and does NOT perform any further checks.

MQ Security Startup Checklist

- If any WebSphere MQ switch is set on, the queue manager checks the status of the RACF class associated with the type of security corresponding to the WebSphere MQ switch. If the class is not installed or not active, the WebSphere MQ switch is set off. For example, process security checks are not carried out if the MQPROC class has not been activated. The class not being active is equivalent to NOT defining a ssid.NO.PROCESS.CHECKS for every queue manager that uses this ACF2 database.
- If you change the switch profiles while the queue manager is running, you can get WebSphere MQ to recognize the changes by issuing the WebSphere MQ command REFRESH SECURITY.

SADEF Turning On Security Checking

```
INSERT SAFDEF.CSQ11 ID(CSQ11) FUNCRET(8) RETCODE(4) -
MODE(IGNORE) RACROUTE(REQUEST=EXTRACT,CLASS=MQADMIN, -
ENTITYX=CSQ1.NO.SUBSYS.SECURITY) REP

INSERT SAFDEF.CSQ12 ID(CSQ12) FUNCRET(8) RETCODE(4) –
MODE(IGNORE) RACROUTE(REQUEST=EXTRACT,CLASS=MQADMIN, -
ENTITYX=CSQ1.NO.CONNECT.CHECKS) REP

INSERT SAFDEF.CSQ13 ID(CSQ13) FUNCRET(8) RETCODE(4) –
MODE(IGNORE) RACROUTE(REQUEST=EXTRACT,CLASS=MQADMIN,-
ENTITYX=CSQ1.NO.QUEUE.CHECKS) REP

INSERT SAFDEF.CSQ18 ID(CSQ18) FUNCRET(8) RETCODE(4) –
MODE(IGNORE) RACROUTE(REQUEST=EXTRACT,CLASS=MQADMIN, -
ENTITYX=CSQ1.NO.CMD.CHECKS) REP

INSERT SAFDEF.CSQ19 ID(CSQ19) FUNCRET(8) RETCODE(4) –
MODE(IGNORE) RACROUTE(REQUEST=EXTRACT,CLASS=MQADMIN, -
ENTITYX=CSQ1.NO.CMD.RESC.CHECKS) REP
```

CA ACF2 CLASMAP Records

A CLASMAP ties an ACF2 resource class to an ACF2 resource rule.

- The resource type is any user-defined token. You only need to define the resource classes that you want the queue manager to check security on.
- ACF2 resource rules consist of a 3-character resource type and a 39-byte resource class name. The ACF2 resource class type corresponds to the RACF class.

CA ACF2 CLASMAP Records

Resource Class	Resource Rule	Description
MQADMIN	MQA	Administration security
MQCONN	MQK	Connection security
MQUEUE	MQQ	Queue security
MQPROC	MQP	Process security
MQNLIST	MQN	Namelist security
MQCMDS	MQC	Command security
MQTOPIC	MQT	Topics security

Examples of CLASMAP Records

```
INSERT CLASMAP.MQADMIN RESOURCE(MQADMIN) RSRCTYPE(MQA) ENTITYLN(62)

INSERT CLASMAP.MQCONN  RESOURCE(MQCONN)  RSRCTYPE(MQK) ENTITYLN(10)

INSERT CLASMAP.MQCMDS  RESOURCE(MQCMDS)  RSRCTYPE(MQC) ENTITYLN(22)

INSERT CLASMAP.MQQUEUE RESOURCE(MQQUEUE) RSRCTYPE(MQQ) ENTITYLN(53)

INSERT CLASMAP.MQPROC  RESOURCE(MQPROC)  RSRCTYPE(MQP) ENTITYLN(53)

INSERT CLASMAP.MQNLIST RESOURCE(MQNLIST) RSRCTYPE(MQN) ENTITYLN(53)

INSERT CLASMAP.MQTOPIC RESOURCE(MQTOPIC) RSRCTYPE(MQT) ENTITYLN(53)
```

Resources Rules for MQADMIN Class

- To enable security so that only the MQ Administrator is allowed to issue MQ administrator commands or modify MQ resources, the following rules are needed:

```
SET R(MQA)
COMPILE *

$KEY(CSQ1) TYPE(MQA)
- UID(*JOHNDOE) SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
UID(-) SERVICE(READ) ALLOW

STORE
```

Resources Rules for MQCMDS Class

- To enable security so that only the MQ Administrator is allowed to issue regular MQ commands, the following rules are needed:

```
SET R(MQC)

COMPILE *
$KEY(CSQ1) TYPE(MQC)
- UID(*JOHNDOE) SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
- UID(-) SERVICE(READ,ADD) ALLOW

STORE
```

Resources Rules for MQCONN Class

- To control whom can connect to a queue manager, the following rules are needed:

```
SET R(MQK)

COMPILE *
$KEY(CSQ1) TYPE(MQK)
 BATCH UID(-) ALLOW
 CICS UID(-) ALLOW

STORE
```

Resources Rules for MQQUEUE Class

- To control who can access which queues on this queue manager:

```
SET R(MQQ)

COMPILE *
$KEY(CSQ1) TYPE(MQQ)
 - UID(*JOHNDOE) SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 KMQ.- UID(-) SERVICE(READ,ADD,UPDATE,DELETE) LOG
 SYSTEM.DEFAULT.- UID(-) SERVICE(READ) ALLOW
 SYSTEM.COMMAND.REPLY.MODEL UID(-) SERVICE(READ) ALLOW
 SYSTEM.COMMAND.INPUT UID(-) SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 SYSTEM.CSQOREXX.- UID(-)    SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 SYSTEM.CSQUTIL.- UID(-)     SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 SYSTEM.CSQXCMD.- UID(-)     SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 SYSTEM.DEAD.LETTER.QUEUE UID(-) SERVICE(READ,ADD,UPDATE,DELETE)
ALLOW

STORE
```

Resources Rules for MQNLIST Class

- This resource rule controls who can access which namelist object on this queue manager. The following rules are needed:

```
SET R(MQN)

COMPILE *
$KEY(CSQ1) TYPE(MQN)
 - UID(*JOHNDOE) SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 SYSTEM.DEFAULT.- UID(-) SERVICE(READ) ALLOW

STORE
```

Resources Rules for MQTOPIC Class

- To control who can access which topics on this queue manager:

```
SET R(MQT)

COMPILE *
$KEY(CSQ1) TYPE(MQT)
 - UID(*JOHNDOE) SERVICE(READ,ADD,UPDATE,DELETE) ALLOW
 HQ.PAYROLL.- UID(-) SERVICE(READ,ADD,UPDATE,DELETE) LOG
 ACCT.DETAILS.- UID(-) SERVICE(READ,ADD,UPDATE,DELETE) LOG


STORE
```

Questions & Answers