M22: MQSeries MVS Security

Stuart Jones
MQSeries Technical Strategy
IBM Hursley
stuartc_jones@uk.ibm.com

Trademarks

The following terms, used in this presentation, are trademarks or registered trademarks of the IBM Corporation in the United States or other countries:

IBM AIX PC-DOS OS/2 OS/400 VTAM MVS/ESA VSE/ESA CICS
CICS/ESA CICS/MVS CICS/VSE CICS OS/2CICS/6000
CICS/400 SNA MQSeries MQ

The following terms, used in this presentation, are trademarks or registered trademarks of other companies:

Systems Strategies Inc:	ezBRIDGE Transact
Apertus Technologies Inc:	Systems Strategies
Digital Equipment Corporation:	DIGITAL VMS VAX DecNet
Tandem Computers Inc:	Tandem Guardian Himalaya
Hewlett-Packard:	HP-UX
AT&T Company:	AT&T
X/Open Company Limited:	UNIX
SUN Microsystems Inc:	SunOS Solaris
Microsoft Corp:	Windows
Novell Inc:	UnixWare
The Santa Cruz Operation Inc:	SCO
Siemens Nixdorf Information Systeme	SINIX

Agenda

Security Overview

Controlling Security for MQSeries

- Switch Profiles
- Resource Profiles

- Userids

CICS & IMS Considerations

MQSeries dataset protection

Administration

Agenda - Notes

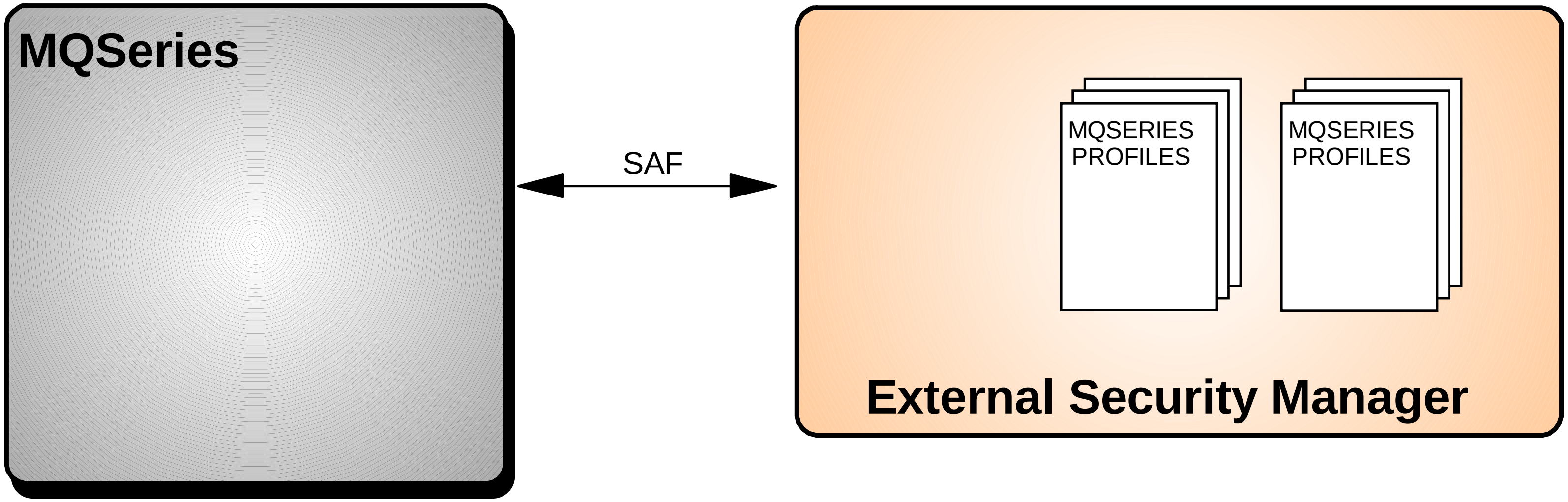
This presentation provides an overview of the way that security is implemented for the MQSeries environment on MVS/ESA. It is assumed that the audience is already familiar with MQSeries concepts, with the MVS environment and with security in general. Thus, the intent is to provide a description of the way that security facilities are provided for this particular environment. There is a description of what the security environment is, how it is activated and how access control is done. There is also an overview of the specific requirements of CICS and IMS - given that this is an MVS environment.

Although security services are provided generically on MVS via the SAF interface, there is an emphasis in these charts to RACF. All of the concepts will be easily extendible to other security manager products.

Throughout the presentation ’ssid’ is used as a prefix for profile names. This stands for ’sub-system id’ and is the name of the Queue Manager to which the profile is relevant. Because of the way in which MQSeries accesses the profiles (the ssid is used as a filter on some of the SAF calls), the ssid cannot be generic and so profiles cannot be shared across Queue Managers in an MVS environment.

For most of the information contained in this presentation, further details are available in the MQSeries for MVS/ESA System Management Guide.

Security Overview



SAF to provide choice of External Security Manager

- RACF, ACF2, Top Secret, ...
- MQSeries defines a set of classes to hold profiles
- Profiles provide access control capabilities

Features depend upon profiles used

- MVS control is more granular than other systems

Security Overview - Notes

MQSeries for MVS/ESA uses the MVS System Authorization Facility (SAF) to provide access control services within the MVS environment. This is the standard mechanism of providing security in an MVS environment and has two significant advantages; firstly, there is a security manager for the entire MVS environment and secondly there is a choice of External (to MQSeries) Security Manager (ESM) to choose from, providing greater flexibility.

Access Control Lists (ACLs) are implemented as a set of profiles. A user is granted access to a particular profile to allow access to the protected resource. To contain the profiles, MQSeries ’defines’ a set of classes, as follows:

- MQADMIN contains profiles for administration functions
- MQCONN contains profiles to limit connection to the MQSeries subsystem
- MQCMDS contains profiles for command security
- MQQUEUE controls queue resources
- MQPROC controls process resources
- MQNLIST controls namelist resources

These lists are defined to the RACF system and need to be defined to other security manager products as well.

Controlling Security - Switch Profiles

Granular control of security checking

- ssid.NO.SUBSYS.SECURITY

- Connection Security
  - ssid.NO.CONNECT.CHECKS

- MQ API Security
  - ssid.NO.QUEUE.CHECKS
  - ssid.NO.PROCESS.CHECKS
  - ssid.NO.NLIST.CHECKS
  - ssid.NO.CONTEXT.CHECKS
  - ssid.NO.ALTERNATE.USER.CHECKS

- MQ Command Security
  - ssid.NO.COMMAND.CHECKS
  - ssid.NO.CMD.RES.CHECKS

...all defined in MQADMIN class

Controlling Security - Switch Profiles - Notes

Switch profiles are RACF profiles which control the level of security checking carried out by MQSeries. These profiles are not used in the same way that profiles are normally used - for access control. They are used simply as switches to activate/deactivate access control checking for various components of Queue Manager processing. Thus, userids are not permitted various access rights to these profiles.

The default action for MQSeries is to have access control checks activated. The *presence* of a switch profile will *deactivat*e security checking for the appropriate component. As these switch profiles are not present by default, it means that explicit action is required to deactivate security processing within an MQSeries environment, once RACF is active and the MQSeries classes are defined (as no checking is possible otherwise !). If the ssid.NO.SUBSYS.SECURITY profile is present then no further checks are made for other switch profiles.

Activating security support in this way means that no security system mananagement is done from within the Queue Manager, which is deemed to be a good thing. However, the use of profiles in this way is quite unusual.

The different switches represent the different components of MQSeries to which access control checks may be applied and are split into 3 areas:

- Connecting to the Queue Manager
- MQ API
- MQ commands

Command security and Command Resource security checking (the latter part, above) were originally designed to be used together to provide a greater granularity to determine whether the issuer of the command was allowed to do so and also, if that command affected a resource, whether they were allowed to issue that particular command for the given resource. If they are both active, then all commands issued that affect resources will have both security checks carried out and must pass both checks for the command to be issued.

However each one can be used independently. If you just have Command security active, the only check performed is to determine whether the issuer of the command is authorised to do so. If authority is granted the command will be issued. If you just have Command Resource security active then anyone can issue commands that do not affect resources, but if a resource is affected a Command Resource security check will be performed to determine whether the issuer is allowed to issue that command for the named resource.

Controlling Security - RESLEVEL Profile

Controls the *number* of userids used for access control

- Based upon executing userid’s access to RESLEVEL profile

Single profile per Queue Manager

- MQADMIN class

Controlling Security - RESLEVEL Profile - Notes

While the switch profiles control *what* checks are made by MQSeries, the RESLEVEL profile controls *which* userid - or userids - will be checked. The number of userids for which checks will be carried out depends on two factors:

- The access that the adapter userid has to the ssid.RESLEVEL profile. The adapter userid is the userid of the address space in which the adapter is running.
- The environment in which the adapter is running ... batch/TSO, CICS or IMS.

The number of userids checked wil be 0, 1 or 2, though for the batch/TSO adapter it will be 0 or 1. The higher the level of access that the adapter userid has to the ssid.RESLEVEL profile, the fewer checks will be made. Because there is just one profile per Queue Manager, the use of the same userid in different environments will result in the same number of checks being caried out for that userid.

Connection Security

Profiles within MQCONN class

- One profile per adapter type
  - ssid.BATCH
  - ssid.CICS
  - ssid.IMS
  - ssid.CHIN

  ... granular control for each environment

- READ access required by (MQSeries) adapter userid

MQ API Security

Controlled by several profiles:

- MQQUEUE
  - ssid.queuename
- MQADMIN
  - ssid.CONTEXT
  - ssid.ALTERNATE.USER.alternateuserid
- MQPROC
  - ssid.processname
- MQNLIST
  - ssid.namelistname

May involve many checks

- Up to 8 !

Queues ... named queue *only* is checked

No checking for QMGR object

MQ API Security...

MQQUEUE class

- ssid.queuename

- Access required to profile is dependent upon
  - MQOPEN options
  - MQPUT1 options

  - Inquire, Browse ... READ access
  - Set ...ALTER access
  - All others require UPDATE access

- Access granularity is not very great
  - MQGET is the same as MQPUT ... different to non-MVS systems

- MQOPEN for dynamic queues requires access to multiple profiles
  - Model queue profile
  - Dynamic queue profile

- MQCLOSE checking for permanent dynamic queues

MQ API Security...

MQADMIN class

- ssid.CONTEXT

- Controls access to MQMD context fields
- Access required to profile is dependent upon access required to context fields
- One profile per Queue Manager

MQADMIN class

- ssid.ALTERNATE.USER.alternateuserid

- Controls the *use* of an alternate userid
- To use an alternate userid ... UPDATE access to appropriate profile
- One profile per Queue Manager

MQ API Security...

MQPROC class

- ssid.processname
- READ access required by userid(s)

MQNLIST class

- ssid.namelistname
- READ access required by userid(s)

Processes and Namelists are opened for inquiry only

MQ API Security - Notes

Access control checks for the MQ API are made when a queue is opened (MQOPEN or MQPUT1) and - for permanent dynamic queues - when a queue is closed (and deleted). Because of the different ways in which a queue may be opened, there are different access control checks and different access levels required. MQSeries provides 5 different profiles, depending on the object being opened and the access required. There may also be several checks made - if a queue is being opened in a specific way ... if a dynamic queue is being created **and** alternate userids are to be used **and** MQMD context fields are to be accessed. The total number of checks that might be made is 8 - if the RESLEVEL profile requires that 2 userids are checked.

For model queues, there are number of security related aspects to consider:

- Two checks are performed
  1) Authorisation to access the *Model* queue
  2) Authorisation to access the *dynamic* queue to which model queue resolves

- Dynamic queues and generic names for them...There are several things to consider here too, but the main one is to ensure that the dynamic queue names have appropriate profiles defined for them. This will, most likely, involve th use of a generic profile.

- Resource security checking performed during closure of permanent dynamic queues. This is performed if an application opens a permanent dynamic queue that it didn’t create and then attempts to delete it...an additional security check is carried out to see if it has authority to do so...

Some applications require access to the MQMD context fields. These fields are protected by the context security profile and - for some types of access - by the queue profile. The type of access required to the context fields varies and so the access required to the context security profile varies according to the access required. These fields include the UserIdentifier - this is typically used to determine the Alternate userid that is to be used for processing the message.

Alternate Userid profiles - These profiles are used to control who is allowed to perform work for another user, under that alternate user’s authority. For example a server may be receiving work from lots of different places...it needs to be able to put those messages on behalf of the users who sent them...in order to do this the server would have to have the correct authority granted against the appropriate Alternate Userid profile. Thus, if userid SERVER54 needs to do work on behalf of USER12, SERVER54 needs UPDATE access to the ssid.ALTERNATE.USER.USER12 profile.

NOTE:- MQSeries userids can be 12 characters long and all 12 characters may be used on the profile for alternate user authority. Even so, only the first 8 characters of the userid are used for any security check performed.

MQ Command Security

MQCMDS class

- ssid.verb.pkw
  - eg. ssid.DEFINE.QLOCAL
- Access required to profile is dependent upon the verb
- Allows *completely* granular control of MQ commands
  - much more than non-MVS systems

MQADMIN class

- ssid.type.localresourcename
  - eg. ssid.QUEUE.qname
- Access required to profile is dependent upon the verb
  - usually ALTER or CONTROL

MQ Command Security - Notes

These two profiles allow *very* granular control of the MQSeries commands. There is a separate profile for each MQSeries command (the verb) and target (the primary keyword), allowing each command to be controlled individually. Thus a particular userid may be able to define qlocals but not define qremotes or may be able to display queues but not define queues. It is also possible to control access to the resources accessed by these commands. Thus, a user may be authorised to use the ALTER QLOCAL command but not alter a specified queue.

Clearly, there is a price to pay with respect to this control; if this type of granular control is required then many profiles may need to be defined to facilitate this access control. The MQSeries for MVS System Management Guide provides a table (in chapter 7) showing the profiles and access required for each profile.

Userids

All access control is userid based

Userids are environment dependent

- Batch
  - Address space userid
  - TSO userid
- CICS
  - Address space userid
  - Transaction userid
- IMS
  - Address space userid
  - ’Second’ userid
- MVS Mover
  - Address space userid
  - Network userid
  - MCAUSER

RESLEVEL profile controls number of userids checked

Userids - Notes

All access control checks are based upon the availability of one or more userids. The userids used depend upon the environment (i.e. the adapter) in use.
For all environments, the address space userid is usually used (when any checking is done at all). This address space userid may not be available if the address space is a started procedure. In this case, a userid is obtained from the MVS Started Procedures Table. For batch address spaces, this is the only userid available. For CICS and IMS address spaces - where many independent transactions are running, a second userid is available. For
CICS, this is the transaction userid. For IMS, the second userid varies according to the type of IMS Dependent Region in use, as follows:

When IMS transactions connect to MQSeries, there are (as with CICS on MVS) two userids used for controlling access to MQSeries resources. These are:

1. The address space userid - for either the IMS Control region or IMS Dependent region
2. One of
   - userid associated with the IMS transaction
   - LTERM name
   - PSBNAME

   The choice of which of these to use is driven by the type of dependent region, according to the following table:

Type of dependent region	Hierarchy for determining 2nd userid
1. BMP message driven and successful GET UNIQUE issued 2. IFP and GET UNIQUE issued 3. MPP	1. Userid associated with the IMS transaction if the user id signed on or 2. LTERM name if available or 3. PSBNAME
1. BMP message driven and successful GET UNIQUE not issued 2. BMP not message driven 3. IFP and GETUNIQUE not issued	1. Userid associated with the IMS dependent region address space if this is not blanks/zeros or 2. PSBNAME

Userids...

Command checking, Cmd Resource checking

- CSQINP1 & CSQINP2
  - no checks
- System Command Queue
  - MQMD.UserIdentifier
- Console
  - Console userid
- SDSF/TSO
  - TSO, address space userid
- MGCR (SVC34) - MVS master get command routine
  - TSO, address space userid or Utoken userid
- CSQUTIL
  - address space userid
- CSQINPX
  - MVS mover address space userid

Access required to system queues

Userids - notes

The userid used for Command and Cmd Resource security checking varies according to the source of the command(s), though the same userid is used for both the Command and Cmd Resource checks. Commands can be issued from a variety of places and the userid that is checked depends on the source of that command, as follows:

CSQINP1 or CSQINP2
No check is made as these are datasets used by the Qmgr during start up and it is recommended these datasets are protected using the normal methods.

System Command Input Queue
The userid found in the UserIdentifier of the message descriptor of the message that contains the command is used. If the
message does not have anything in this field a userid of BLANKs is used and passed to the security manager.

Console
The userid signed onto the console. If the console is not signed on, the DEFAULT userid from the MQSeries subsystem
initialisation parameter module (CSQZPARM) This default is set by the CMDUSER operand on the CSQ6SYSP macro. To
issue commands from a console , the console must have the MVS SYS AUTHORITY attribute.

SDSF/TSO console ... The TSO or job userid

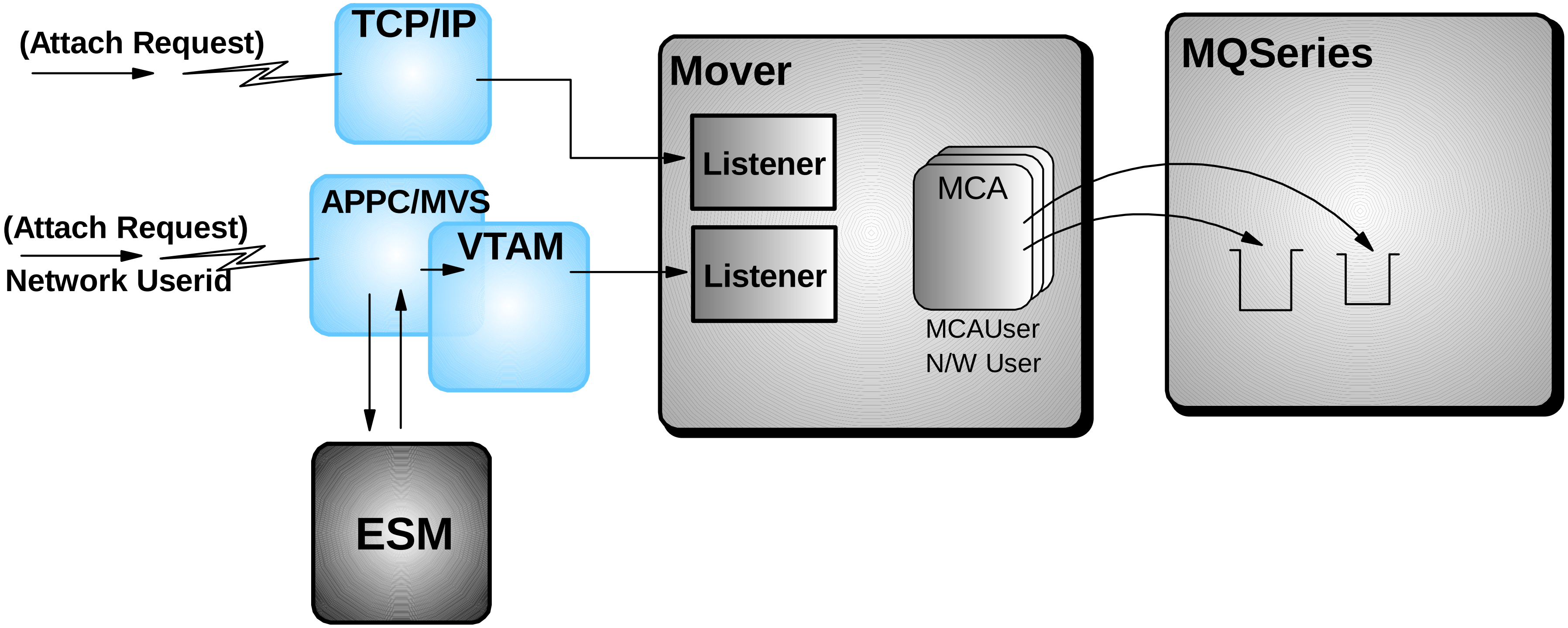
MGCR(SVC34) - MVS master get routine command
If MGCR is used with a UToken, the userid in the UToken. If MGCR is used without a UToken, the TSO or Job userid is used.

CSQUTIL ... The job userid

CSQINPX ... The userid of the channel initiator address space.

For these programs, there are a number of queues that are used and an number of queues that are dynamically created. Appropriate profiles must be created to allows the appropriate userids to access these queues. These queues are all documented in the MQSeries for MVS/ESA System Management Guide. For all of these queues, the userid running the command utillty must have UPDATE access to the apprpriate queue profile.

MQSeries Channel Security



Multiple userids are used for MQPUT authorisation

- Complex to configure
- Documentation is ’challenging’

- MQSeries for OS/390 2.1 addresses these issues
  - N/W userid may be ignored ... for authorisation
  - Configuration tables re-written

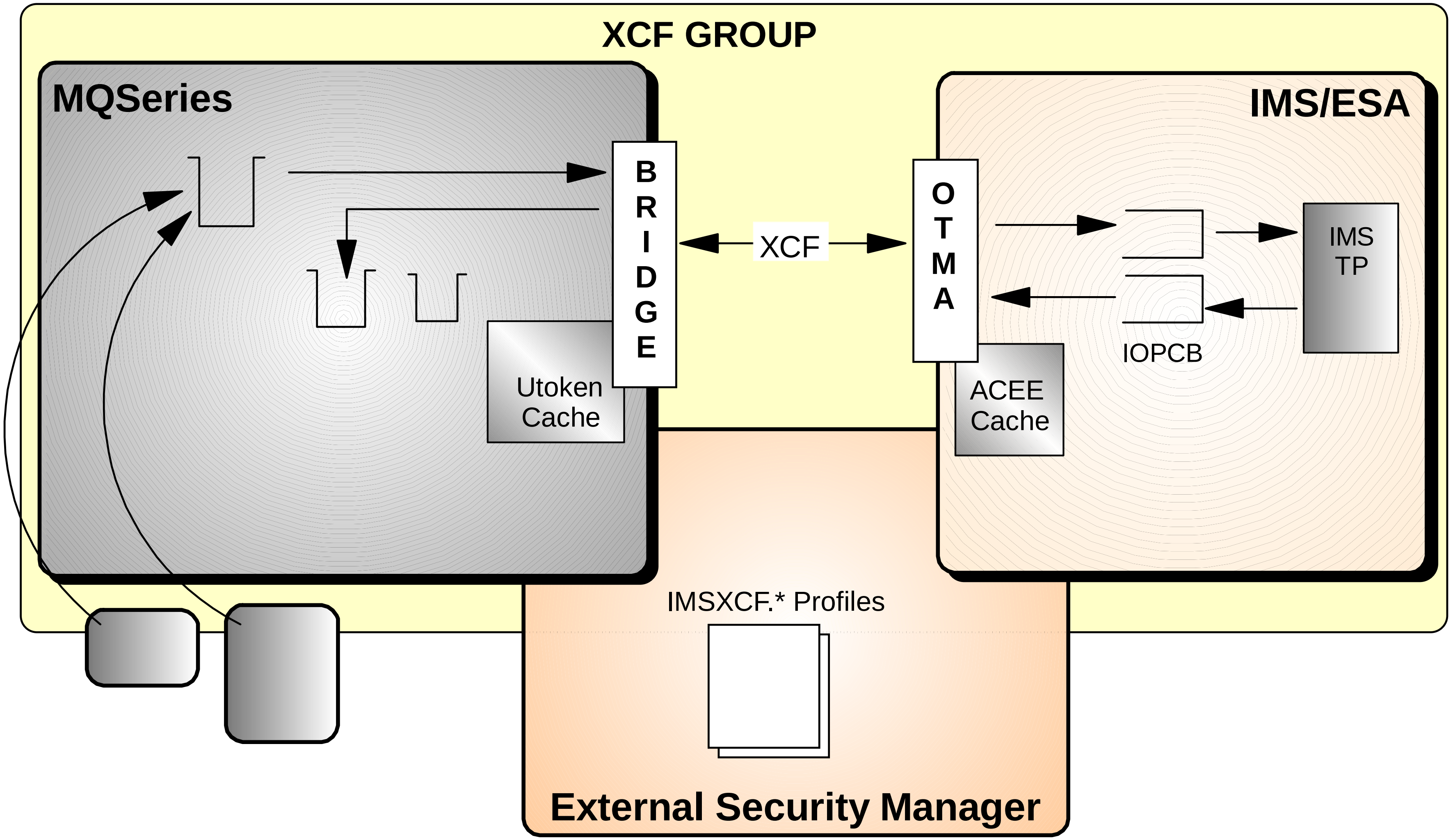
MQSeries Channel Security

The diagram shows the environment when other queue managers conenct to an OS/390 queue manager via server/server or clinet/server channels. From a security viewpoint, it may be necessary - when using APPC - to provide a N/W userid which will be verified before any connection is permitted. Once the channel is active, there are a number of userids available for authorisation checking when a channel needs to open a queue...the mover (CHINIT) address space userid, the MCAUSER and the received N/W userid. Depending upon the aceess that the CHINIT address space userid has to the RESLEVEL profile, zero, one or two userids will be used. Depending upon the availability of these userids, various substitutions are possible. Complex, isn't it !!

The complexity of this area has led to complex documentation as well, which has further complicated the configuration of the environment.

The latest relase of the queue manager for OS/390 includes some relief for this. Firstly, it is no longer necessary to include the N/W userid in the set of userids used for configuration. The Channel MCAUSER parameter has been extended to allow the N/W userid to be ignored. As well as providing simplification, this can make the OS/390 environment the same as other queue managers. Second, the documentation has been signigficantly improved to ease the configuration complexity.

IMS Bridge


XCF GROUP
MQSeries
B R I D G E
Utoken Cache
XCF
O T M A
IMS/ESA
IMS TP
IOPCB
ACEE Cache
IMSXCF.* Profiles
External Security Manager

IMS Bridge...

FACILITY class

- IMSXCF.xcfgname.xcfmname

- MQSeries/IMS connection security
  - IMSXCF.xcfgname.MQSeries_member_name
  - MQSeries userid requires READ access to this profile

- IMS level of authentication
  - IMSXCF.xcfgname.IMS_member_name
  - Security processing dependent upon MQSeries' access to this profile

/SECURE OTMA

- Controls userid processing

MQSeries system parameters

- CSQ6SYSP ... OTMACON=(,,,Age,)

IMS Bridge - Notes

When using the IMS Bridge, MQSeries messages are passed to the IMS system via IMS' OTMA facility. The userid for the message is contained in the MQMD, as usual. There is also an optional password (or PassTicket) contained in the MQSeries IMS header, associated with each message (MQIIH). MQSeries, therefore, has the capability to sign on users and can pass security information about that signed-on user to IMS via OTMA. There are several controls available within the IMS Bridge support to regulate how much security processing is carried out, as follows:

- According to the access that the MQSeries address space userid has to the appropriate IMSXCF profile, MQSeries will take action for the userid in the MQMD of each message which is targetted to the IMS Bridge:

  Access = NONE ... A **sign-on with password** is performed and no security tokens are cached for the userid

  Access = READ ... A **sign-on with password** is performed if the userid has not been previously seen by the Queue Manager. The MQSeries OTMACON.Age parameter controls how long the Queue Manager remembers userids. A Utoken is cached for the userid.

  Access = UPDATE ... A **sign-on without password** if the userid has not been previously seen by the Queue Manager. The MQSeries OTMACON.Age parameter controls how long the Queue Manager remembers userids. A Utoken is cached for this userid.

  Access = CONTROL/ALTER ... **No sign-ons** are performed and no Utokens are created/cached. This option would be used in a test or development system where no security checking is required.

- MQSeries is now able to pass security information to IMS. The /SECURE OTMA parameter controls the use to which IMS puts the security information, as follows:

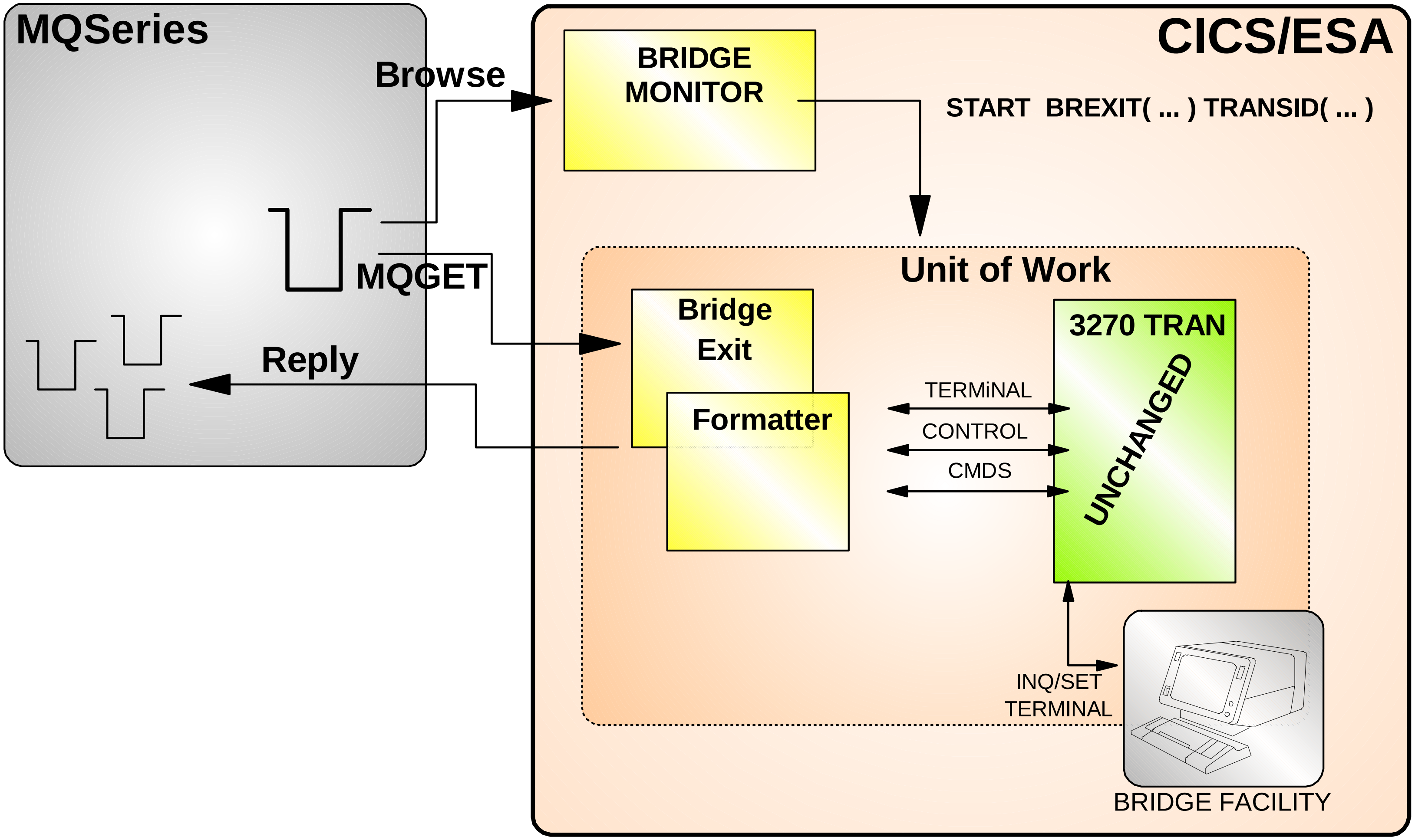
  NONE... No checks are made

  CHECK ... MQMD.UserIdentifier is passed to IMS and transaction/command security is performed by the IMSControl Region

  FULL ... As for CHECK and the MQMD.Useridentifier is passed to the IMS Dependent Region for additional security checks, for security processing that is not handled by the IMS Control Region (eg APPC).

- Connection security is also available. The same IMSXCF profile as is used above is used to control the access that MQSeries has to connect OTMA. However, in this instance the MQSeries XCF member name is used as the key - rather than the IMS XCF member name. The MQSeries address space userid requires READ access to the appropriate profile in order to be able to conect to IMS systems in the relevant XCF group.

CICS 3270 Bridge



Userid/Password supplied to 3270 transaction

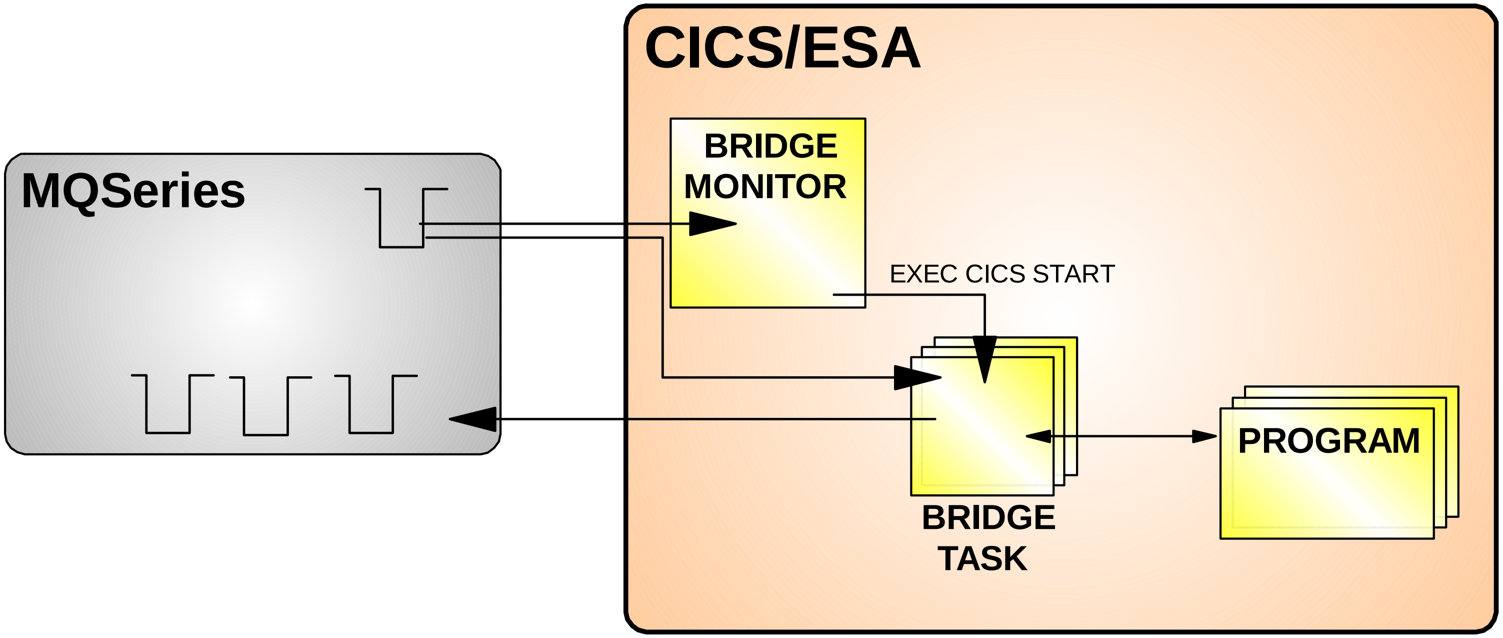
- Password verified if present
- Surrogate checking otherwise

CICS 3270 Bridge - Notes

The CICS 3270 Bridge is quite similar to the IMS Bridge. However, the CICS Bridge provides fewer controls and options for security than the IMS Bridge.

When the Bridge transactions starts up, it can set the userid and password that the 3270 transaction will run with. If a userid only is supplied, then CICS will carry out its standard surrogate checking to ensure that the userid of the Bridge transaction is authorised to use the specified userid and will then perform a **sign-on without password**. This make use of the SURROGAT class. If a password (or PassTicket) is supplied with the userid, then CICS will perform a **sign-on with password**. CICS provides a similar caching mechanism to the MQSeries OTMACON.Age parameter. The SIT option USRDELAY controls for how long unused user information is retained.

CICS DPL Bridge



Security checking depends upon CICS release

- CICS/ESA 4.1 and above
  - granular security for Bridge Tasks
- CICS/ESA 3.3 and below
  - requires ’terminals’ to provide security

CICS DPL Bridge - Notes

The CICS DPL Bridge makes extensive use of EXEC CICS START commands. Because of this, the level of security available for the bridge is dependent upon the release of CICS on which the bridge is run. CICS/ESA Version 3.3 and below do not support security for STARTed transactions unless the START is associated with a terminal. CICS/ESA Version 4.1 and above have security processing for all transactions and so there is not an issue for STARTed transactions.

The Bridge Monitor program is started via a CICS command, CKBR. One of the parameters for this command is the level of authority checking required for the Bridge Tasks. This can be one of the following:

- LOCAL
  This is the default and is the lowest level of authority available. For CICS 4.1 and above, the Bridge Task runs with the CICS DFLTUSER userid. For CICS 3.3 there is no security for the Bridge Task.

- IDENTIFY
  This is only available for CICS 4.1. The Bridge Task is started with the authority of the userid in the MQMD. In order to use this, the userid of the Bridge Monitor (which issues the START) needs the appropriate to use the userid in the MQMD. This makes use of the SURROGAT class.

- VERIFY_UOW
  For CICS 4.1 this is the same as IDENTIFY except that a password (or PassTicket) is required, which is verified using EXEC CICS VERIFY. This is done once for each UOW and is applicable to trnasactions where there are several link calls within a unit of work. For CICS 3.3, the Bridge Monitor must run as a terminal transaction and the Monitor will issue a sign-on with password for the userid in the MQMD (using EXEC CICS SIGNON). A subsequent START will propagate the userid to the STARTed Bridge Task.

- VERIFY_ALL
  This is only available for CICS 4.1 and is the same as VERIFY_UOW except that the userid/password is checked for every message, rather than every UOW.

Administration

DISPLAY SECURITY

- ALL|INTERVAL|SWITCHES|TIMEOUT

REFRESH SECURITY

- *|MQADMIN|MQQUEUE|MQPROC|MQNLIST

RVERIFY SECURITY

- Userid, Userid, ...

ALTER SECURITY

- INTERVAL() TIMEOUT()

Administration - Notes

There are four MQSeries for MVS/ESA commands that help you to administer security for your Queue Manager.

- DISPLAY SECURITY ALL|INTERVAL|SWITCHES|TIMEOUT
  This allows you to see what security is active on on your Queue Manager and how frequently the internal clearout is performed of security information held by the Queue Manager.

- REFRESH SECURITY(*|MQADMIN|MQQUEUE|MQPROC|MQNLIST)
  This command allows you to change your Queue Manager's security setup without bringing down the Queue Manager. There will be a performance impact whilst this command is being processed.

- RVERIFY SECURITY(userid,userid...)
  This allows you to change particular users access levels whilst the system is up and running...once the change has been made in RACF then you need to issue this command, specifying each userid that has had its authority changed.

- ALTER SECURITY INTERVAL() TIMEOUT()
  This allows you to alter how frequently the internal clearout occurs and the period of time that information is allowed to remain in the Queue Manager unused.

Protection of MQSeries Datasets

MVS Datasets

- Business as usual

Summary

Security Overview

Controlling Security for MQSeries

- Switch Profiles
- Resource Profiles

- Userids

CICS & IMS Considerations

MQSeries dataset protection

Administration