Exercise 8. Controlling access to IBM MQ

Estimated time

01;00

Overview

In this exercise, you use the IBM MQ OAM commands to set access control on a queue, and then use the IBM MQ sample programs to see the effect of attempting to breach security.

Objectives

After completing this exercise, you should be able to:

- Use the **setmqaut** command to define access control on a queue
- Use the **dspmqaut** command to display the access control on a queue
- Use IBM MQ Explorer to manage authority records
- Enable and monitor authority events
- Test security by using IBM MQ sample programs

Introduction

In this exercise you use a non-privileged user ID of “oamlabuser” that is a member or the “oamlab” group to access MQ objects, experience access failures, examine related diagnostic messages, and apply the correct rules. You use the MQ Explorer and the **`setmqaut`** command to apply OAM rules.

L **Linux**

All authorities are set at the group level by default. In Exercise 1, you created the queue manager so that you can set authorities at user-level authority by entering the **`-oa user`** option on the **`create queue manager`** command.

In Part 2 of this exercise, you create a generic profile. A generic profile associates with more than one object of the same type. You can grant authorities to a set of objects at the same time by creating an authority record against the generic profile.

Requirements

- IBM MQ and IBM MQ Explorer

- A text editor
- The queue manager QM01 and queues that are created in Exercise 1
- A user that does not belong to the “mqm” group or the “users” group (on Linux)

Exercise instructions

Some steps in this exercise require that you run MQ sample programs to put, get, and browse messages as an unauthorized user that is not a member of the **mqm** group access is required. Other steps require that you complete steps as the Administrator user.

W **Windows**

On the Windows image, the unauthorized user is **oamlabuser**. This user is assigned to the **oamlab** group. It is not a member of the **mqm** or **Administrators** group.

The Administrator user ID is **Administrator**.

L **Linux**

On the Linux image, the unauthorized user is **oamlabuser**. It is not a member of the **mqm** group.

The Administrator user ID is **localuser**.

Part 1: Authorizing a user

In this part of the exercise, you log in as a user **oamlabuser** (the user that does not have object authority) and verify that this user cannot access a queue. The unauthorized user receives a 2035 error when an attempt is made to access the queue from a sample application.

__ 1. If it is not running, start queue manager QM01.

__ 2. Verify that queue QL.A on QM01 is not PUT inhibited.

To display the queue properties in MQSC, type:

```
DIS QL(QL.A)
```

To display the queue properties in MQ Explorer, right-click the queue in the **Queues** view and then click **Properties**.

__ 3. Enable authority events on your queue manager. This method is one way to monitor authority violations.

__ a. Using MQ Explorer, right-click **QM01** in the navigator and click **Properties**.

__ b. On the **Events** tab, select **Enabled** for **Authority events**.

__ c. Click **Apply** and then click **OK**.

__ 4. Open a new command window and start a session (window) as the unauthorized user.

W **Windows**

To start a command window as **oamlabuser**, enter the following command from a Windows command prompt:

```
runas /user:oamlabuser cmd
```

The password is: **passw0rd**

Linux

Open a new terminal window and set the user to **oamlabuser**:

```
su -l oamlabuser
```

The password is: **passw0rd123**

__ 5. In the unauthorized user (**oamlabuser**) command window, run the **amqsput** sample program to attempt to put a message to queue QL.A on the queue manager QM01. Type:

```
amqsput QL.A QM01
```

You should receive an error message with a reason code 2035 (MQRC_NOT_AUTHORIZED)

__ 6. Use the **amqsevt** command to display the queue manager events.

In the administrator command window, type:

```
amqsevt -m QM01 -q SYSTEM.ADMIN.QMGR.EVENT
```

The SYSTEM.ADMIN.QMGR.EVENT queue should include a message similar to the following message:

```
Event Type                : Queue Mgr Event [44]
Reason:                   : Not Authorized [2035]
Event created             : 2016/10/26 09:52_04.54 GMT
   Queue Mgr Name         : QM01
   Reason Qualifier       : Conn Not Authorized
   User Identifier        : oamlabuser
   Appl Type              : Unix
   Appl Name              : amqsput
```

This program continues to run until you either close the window or end the program by typing Ctrl+C.

__ 7. A record of the authority violation is also included in the queue manager **errors** subdirectory.

Locate the error file and open it in a text editor. Scroll to the end of the file to find the most recent entry.

Windows

For Windows queue managers, a security violation message is written to the **`AMQERR01.LOG`** file in the **`C:\ProgramData\IBM\MQ\qmgrs\QM01\errors`** directory.

Linux

For Linux queue managers, a security violation message is written to the **`AMQERR01.LOG`** file in the**`/var/mqm/qmgrs/QM01/errors`** directory.

Error message example:

```
10/26/2016 07:14:41 - Process(2920.12) User(MUSR_MQADMIN) Program(amqzlaa0.exe)
                      Host(WS2008R2X64) Installation(Installation1)
                        VRMF(9.0.0.0) QMgr(QM01)

   AMQ8077: Entity 'oamlabuser@ws2012r2x64' has insufficient authority to
   access object'QM01'.

   EXPLANATION:
   The specified entity is not authorized to access the required object. The
   following requested permissions are unauthorized: connect

   ACTION:
   Ensure that the correct level of authority has been set for this entity
   against the required object, or ensure that the entity is a member of a
   privileged group.
```

__ 8. Use MQ Explorer or the **setmqaut** command in the **Administrator** command window to add connection authority for the user **oamlabuser** to the queue manager QM01.

If you use the **`setmqaut`** command, use the **dspmqaut** command to verify that the user has connect authority.

To use the **`setmqaut`** command to add connection authority for the user **oamlabuser**, type:

`setmqaut -m QM01 –t qmgr -p oamlabuser +connect`

To use the **dspmqaut** command to verify that the user has connect authority, type:

`dspmqaut -m QM01 –t qmgr -p oamlabuser`

To use MQ Explorer to add connection authority for the user **oamlabuser**:

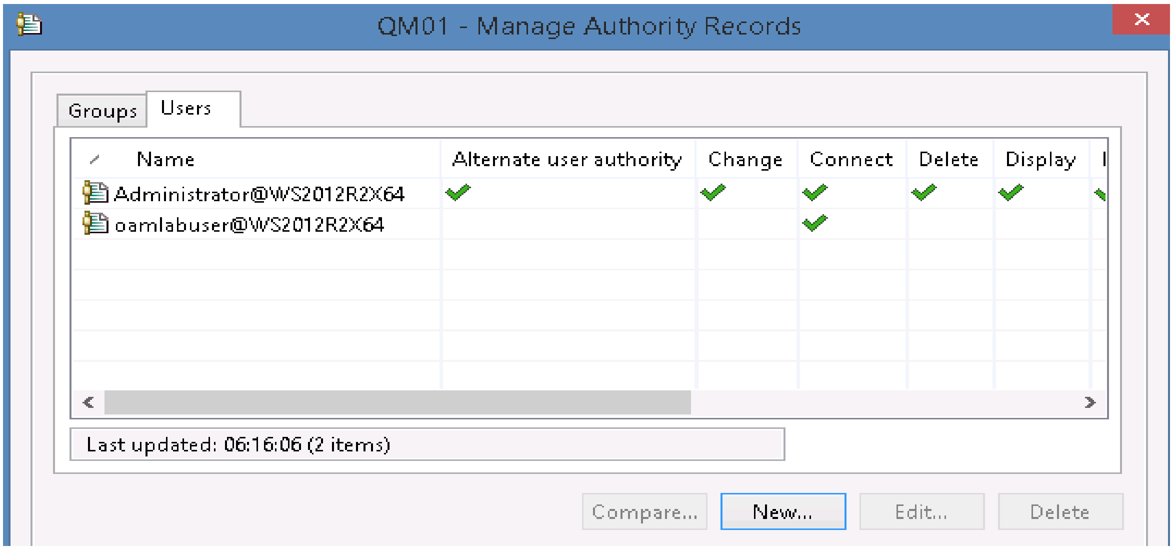
__ a. Right-click the queue manager QM01 in the **MQ Explorer - Navigator** view and then click **Object Authorities > Manage Queue Manager Authority Records**.

__ b. On the **Users** tab, click **New**. Set the **Entity name** to: **oamlabuser**

__ c. Select **Connect** under the **MQI** section. Click **OK**.

__ d. Click **OK** on the confirmation window.

__ e. Verify that the user has **Connect authority**, and then click **Close**.



Name	Alternate user authority	Change	Connect	Delete	Display	I
Administrator@WS2012R2X64	✓	✓	✓	✓	✓	✓
oamlabuser@WS2012R2X64			✓			

Last updated: 06:16:06 (2 items)

__ 9. As the **Administrator**, give the user **oamlabuser** general API access (put, get, browse, inquire, and set) to QL.A on your queue manager.

To use the **setmqaut** command, type:

```
setmqaut -m QM01 -t q -n QL.A -p oamlabuser +put +get +browse +inq +set
```

To use MQ Explorer:

__ a. Right-click the **Queues** folder under QM01 in the **MQ Explorer - Navigator** view and then click **Object Authorities > Manage Authority Records**.

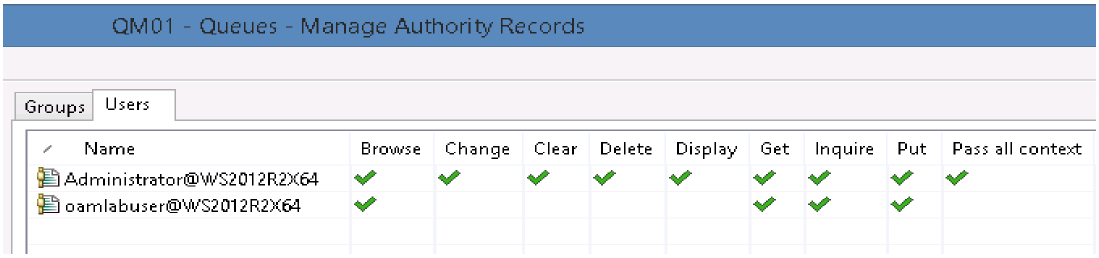
__ b. Expand **Specific Profiles**. Select **QL.A**.

__ c. On the **Users** tab, click **New** and then type **oamlabuser** as the **Entity Name**.

__ d. Select **Browse**, **Get**, **Inquire**, **Put**, and **Set** authority under the **MQI** section. Click **OK**.

__ e. Click **OK** on the confirmation window.

__ f. Verity that the user has **Browse**, **Get**, **Inquire**, **Put**, and **Set** authorities, and then click **Close**.



Name	Browse	Change	Clear	Delete	Display	Get	Inquire	Put	Pass all context
Administrator@WS2012R2X64	✓	✓	✓	✓	✓	✓	✓	✓	✓
oamlabuser@WS2012R2X64	✓					✓	✓	✓	

__ 10. In the command window that is running as **oamlabuser**, put a message on QL.A. This user should now be able to put messages to QL.A.

Type: **amqsput QL.A QM01**

__ 11. In the command window that is running as **oamlabuser**, use the **amqsbcg** sample program on QL.A to browse the queue and verify that the access is now allowed.

Type: **`amqsbcg QL.A QM01`**

Part 2: Using generic profiles

In this part of the exercise, you create a generic profile to grant authorities to a set of objects at the same time by creating an authority record against the generic profile.

__ 1. As **Administrator**, use the **`setmqaut`** command to remove the specific profile that allows user **oamlabuser** to access QL.A.

Type: **`setmqaut -m QM01 -t q -n QL.A -p oamlabuser -remove`**

__ 2. As **Administrator**, create a new generic profile that is called **`QL.*.TEST`** that allows the user **oamlabuser** to access (browse, put, get, set, and inquire) queues that match the generic queue name (**`-n`**).

Type: **`setmqaut -m QM01 -t q -n QL.*.TEST -p oamlabuser +allmqi`**

__ 3. As **Administrator**, define a queue on QM01 that is named QL.A.TEST.

Type:

```
runmqsc QM01
DEFINE QL(QL.A.TEST)
END
```

__ 4. As **oamlabuser**, try to access QL.A. You should get the 2035 error.

Type: **`amqsput QL.A QM01`**

__ 5. As **oamlabuser**, try to put a message to QL.A.TEST.

Type: **`amqsput QL.A.TEST QM01`**

This user should be able to put a message to QL.A.TEST because it matches the generic queue name of QL*.TEST.

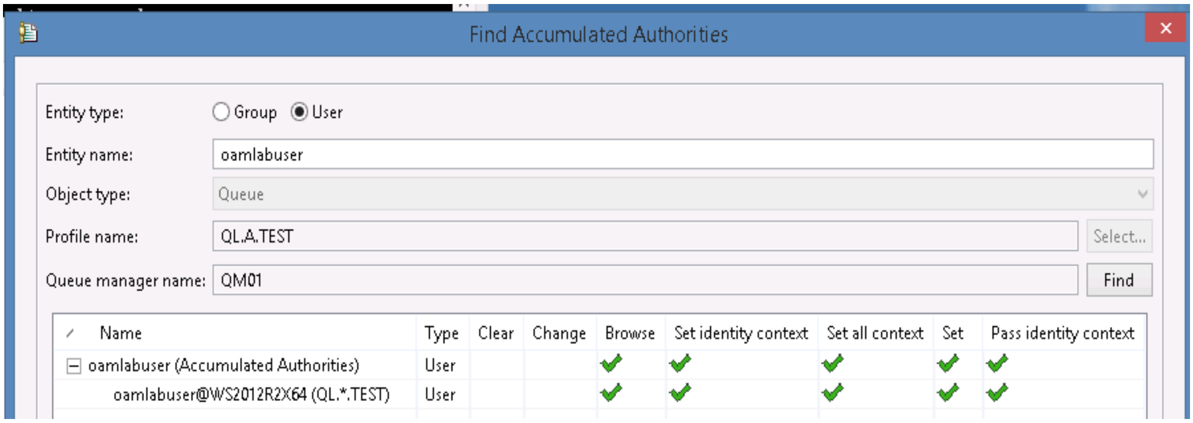
__ 6. As the **Administrator**, use MQ Explorer or the **`dmpmqaut`** command to inspect the profiles that apply to QL.A.TEST queue.

To use MQ Explorer to inspect the profiles that apply to QL.A.TEST, complete the following steps:

__ a. Right-click the queue **QL.A.TEST** in the **Queues** content view and then click **Object Authorities > Find Accumulated Authorities**.

__ b. Click **User** for the Entity Type and then type **`oamlabuser`** for the **Entity name**.

__ c. Click **Refresh**.



Entity type: ○ Group ● User

Entity name: oamlabuser

Object type: Queue

Profile name: QL.A.TEST

Queue manager name: QM01

Name	Type	Clear	Change	Browse	Set identity context	Set all context	Set	Pass identity context
⊟ oamlabuser (Accumulated Authorities)	User			✓	✓	✓	✓	✓
oamlabuser@WS2012R2X64 (QL.*.TEST)	User			✓	✓	✓	✓	✓

To use to the **dmpmqaut** command to inspect the profiles that apply to QL.A.TEST, type:

```
dmpmqaut -m QM01 -t q -p oamlabuser -n QL.A.TEST
```

It returns a summary of the profile. For example,

```
profile: QL.*.TEST
object type: queue
entity: oamlabuser@WS2012R2X64(QL.*.TEST)
entity type: principal
authority: allmqi
```

__ 7. Close the command window that is running as the user **oamlabuser**.

Exercise cleanup

__ 1. Disable authority events on the queue manager QM01.

__ a. Using MQ Explorer, right-click **QM01** in the navigator and click **Properties**.

__ b. On the **Events** tab, select **Disabled** for **Authority events**.

__ c. Click **Apply** and then click **OK**.

End of exercise

Exercise review and wrap-up

In the lab environment, you ran as an unauthorized user and enabled authorization to the queue manager and queues.

Having completed this exercise, you should be able to:

- Use the **`setmqaut`** command or MQ Explorer to create authority records for a queue
- Use the **`dspmqaut`** or MQ Explorer to display the authority records for a queue
- Use IBM MQ utilities and sample programs to test security