

Taking a backup for moving files of an MQ queue manager
from one Linux host
to another Linux host
(Edition 2)

<https://www.ibm.com/support/pages/node/6114082>

Date last updated: 05-Feb-2024

Angel Rivera - rivera@us.ibm.com
IBM MQ Support

+++ Objective

The objective of this tutorial is to take a backup for moving the data files and recovery/transaction log files for an MQ queue manager from one Linux host to another Linux host.

The quick description of the procedure is to take a backup of the files from host-1 and restore them in host-2.

The procedure can only be applied for moving from one host to another that have the same type of architecture and OS, such as Linux x86 64-bit.

The basic procedure is to do a backup of the files from host-1 while the queue manager is not running. The files are stored in the default locations. Then the files are copied into host-2 and restored there, also using the default locations. It assumes that the port number is not in used by another MQ listener.

Additional considerations will be discussed, and recommendations will be provided. For example, if the queue manager to be moved is a participant in an MQ Cluster, then it is critical that the queue manager **MUST** be removed from the MQ Cluster before the move. Then after the move the queue manager can be rejoined into the MQ Cluster.

For more information on doing a backup of your queue managers, please see:
https://www.ibm.com/support/knowledgecenter/SSFKSJ_9.2.0/com.ibm.mq.con.doc/q018580_.html

IBM MQ 9.2.x / IBM MQ / Configuring / Configuring high availability, recovery and restart / Backing up and restoring IBM® MQ queue manager data

It is a Best Practice in an Enterprise to use the SAME numeric userid and groupid for the user “mqm” and group “mqm”, which has the great benefit that there is no need to worry about changing userids/groupids of the files when moving them from one host to another.

This tutorial exploits this Best Practice and in both hosts the output for the following command is the same:

```
$ id mqm
uid=501(mqm) gid=501(mqm) groups=501(mqm)
```

+++ Configuration

To move the files for queue manager QMSALES
from host1 “orizaba1” using MQ 8.0.0.9
to host2 “fortin1” using MQ 9.1.0.4 LTS

+ Source host and queue manager:

Red Hat Enterprise Linux (RHEL) 7.6 on x86-64-bit

Let’s find the release name and version:

a) For RHEL, you can use:

```
+++ROOT+++ orizaba1.fyre.ibm.com: /root
# cat /etc/redhat-release
Red Hat Enterprise Linux Server release 7.6 (Maipo)
```

b) For other Linuxes, or if the LSB component has been installed, then you can use:

```
+++ROOT+++ orizaba1.fyre.ibm.com: /root
# lsb_release -dr
Description: Red Hat Enterprise Linux Server release 7.6 (Maipo)
Release:      7.6
```

Let’s find the architecture. Notice “x86_64” which means Intel hardware, 64-bit.

```
$ uname -a
Linux orizaba1.fyre.ibm.com 3.10.0-957.12.1.el7.x86_64 #1 SMP Wed Mar 20 11:34:37 UTC
2019 x86_64 x86_64 x86_64 GNU/Linux
```

Hostname address: orizaba1.fyre.ibm.com
Queue Manager: QMSALES
MQ: 8.0.0.9
Port number: 1414
InstallationName: Installation2

The queue manager QMSALES has the following status via “dspmq”:

```
$ dspmq -o installation -s -m QMSALES
QMNAME(QMSALES)                STATUS(Running)
INSTNAME(Installation2) INSTPATH(/opt/mqm80) INSTVER(8.0.0.9)
```

The stanza for this queue manager inside the master index file /var/mqm/mqs.ini can be displayed via the command “dspmqinf”:

```
$ dspmqinf -o stanza QMSALES
```

```
QueueManager:
  Name=QMSALES
  Directory=QMSALES
  Prefix=/var/mqm
  InstallationName=Installation2
```

For more details on the Installation2 you can use “dspmqinst”:

```
$ dspmqinst -n Installation2
```

```
InstName:   Installation2
InstDesc:
Identifier:  2
InstPath:   /opt/mqm80
Version:    8.0.0.9
Primary:    No
State:      Available
```

The mqs.ini file has the default locations:

```
AllQueueManagers:
  DefaultPrefix=/var/mqm
LogDefaults:
  LogDefaultPath=/var/mqm/log
```

The above means that these are the full directory names for the following type of data for a queue manager:

- Data files: /var/mqm/qmgrs/QMSALES
- Recovery/Transaction Log files: /var/mqm/log/QMSALES

Let’s take a quick look at these directories:

```
mqm@orizaba1.fyre.ibm.com: /var/mqm
$ ls -dl /var/mqm/qmgrs/QMSALES
drwxrwsr-x 22 mqm mqm 4096 Mar 13 12:54 /var/mqm/qmgrs/QMSALES
```

```
$ ls -dl /var/mqm/qmgrs/QMSALES/*
```

```
-rw-rw-r-- 1 mqm mqm 60 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/active
-rw-rw---- 1 mqm mqm 56 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/amqalchk.fil
drwxrwsr-x 3 mqm mqm 19 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/@app
drwxrws--- 2 mqm mqm 159 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/authinfo
-rw-rw---- 1 mqm mqm 19 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/blockaddr.ini
drwxrws--- 2 mqm mqm 273 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/channel
drwxrws--- 2 mqm mqm 33 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/clntconn
drwxrws--- 2 mqm mqm 47 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/comminfo
drwxrws--- 2 mqm mqm 66 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/errors
drwxrwsr-x 3 mqm mqm 59 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/@ipcc
drwxrws--- 2 mqm mqm 57 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/listener
-rw-rw-r-- 1 mqm mqm 60 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/master
-rw-rw---- 1 mqm mqm 701 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/mqat.ini
drwxrws--- 2 mqm mqm 114 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/namelist
drwxrwsr-x 2 mqm mqm 6 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/plugincomp
drwxrws--- 2 mqm mqm 36 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/procdef
drwxrws--- 2 mqm mqm 40 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/qmanager
drwxrws--- 3 mqm mqm 19 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/@qmgr
-rw-rw---- 1 mqm mqm 1475 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/qm.ini
drwxrws--- 3 mqm mqm 19 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/@qmpersist
-rw-rw-r-- 1 mqm mqm 803 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/qmstatus.ini
drwxrws--- 2 mqm mqm 4096 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/queues
drwxrws--- 2 mqm mqm 6 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/scratch
drwxrws--- 2 mqm mqm 36 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/services
drwxrws--- 2 mqm mqm 6 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/ssl
drwxrws--- 2 mqm mqm 18 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/startprm
drwxrws--- 2 mqm mqm 167 Mar 13 09:39 /var/mqm/qmgrs/QMSALES/topic
```

```
$ ls -dl /var/mqm/log/QMSALES
```

```
drwxrwsr-x 3 mqm mqm 40 Mar 13 09:39 /var/mqm/log/QMSALES
```

```
$ ls -dl /var/mqm/log/QMSALES/*
```

```
drwxrws--- 2 mqm mqm 66 Mar 13 09:39 /var/mqm/log/QMSALES/active
-rw-rw---- 1 mqm mqm 6256 Mar 13 09:39 /var/mqm/log/QMSALES/amqhlctl.lfh
```

Notice the following subdirectories that have the name of the host where the queue manager is running (in this tutorial, the source host is: orizaba1)

Attention: these subdirectories will NOT affect the runtime of the queue manager when it is copied into the target host, because newer directories will be created, using the appropriate hostname.

```
$ cd /var/mqm/log/QMSALES/
```

```
$ ls -R | grep shmem
```

```
shmem
```

```
./@app/shmem:
```

```
./@app/shmem/orizaba1:
```

```
shmem
```

```
./@ipcc/shmem:
```

```
./@ipcc/shmem/orizaba1:
```

```
shmem
```

```
./@qmgr/shmem:
```

```
./@qmgr/shmem/orizaba1:
```

```
shmem
```

```
./@qmpersist/shmem:
```

```
./@qmpersist/shmem/orizaba1:
```

++ Target host and queue manager:

Red Hat Enterprise Linux (RHEL) 7.6 on x86-64-bit

Hostname: fortin1.fyre.ibm.com

MQ: 9.1.0.4 LTS

InstallationName: Installation1

```
mqm@fortin1.fyre.ibm.com: /home/mqm
```

```
$ dspmqinst
```

```
InstName:    Installation1
```

```
InstDesc:
```

```
Identifier:  1
```

```
InstPath:    /opt/mqm
```

```
Version:     9.1.0.4
```

```
Primary:     No
```

```
State:       Available
```

+++ Requirements/Limitations

Moving the data files and recovery log files from a queue manager in host1 to host2 ...

a) ... is **SUPPORTED** if ...

- a.1) if the operating systems are the SAME between host1 and host2, and
- a.2) if the version of MQ in host2 is at least the same as in host2 or a later version/release.

For example,

- From AIX to AIX, where you can take a complete backup of the queue manager in host1 (AIX) and restore in host2 (AIX).
- From Windows MQ 8.0 to Windows MQ 8.0.
- From Windows MQ 8.0 to Windows MQ 9.1.
- From Linux RHEL x86 64-bit to Linux RHEL x86 64-bit.

b) ... is **NOT supported** if ...

b.1) if the operating systems are different between host1 and host2 (mostly due to differences in file system format, data types, data sizes, etc.)

For example, moving from SunOS to AIX, is NOT supported.

b.2) It is NOT supported going from different architectures (hardware) within the same Operating System, such as from SunOS Solaris Sparc to SunOS Solaris Intel (x86), or from Linux on Intel (x86) 64-bit to Linux on Power PC.

b.3) It is NOT supported going to a version/release that is OLDER than the source queue manager, even if the target host has the same architecture and OS than the source.

For example, it is not supported to go from MQ 9.0 in Windows to MQ 7.5 in Windows.

b.4) The only alternative for these situations is to save the queue manager configuration using dmpmqcfg and save the messages from the application queues using dmpmqmsg, create a new queue manager and then restore the queue manager configuration and the messages.

For more details on the steps, see the following article.

This article includes information for older releases in which you need to use the utility “saveqmgr”.

Starting with MQ 7.1, the utility “dmpmqcfg” was developed based on “saveqmgr”.

<https://www-01.ibm.com/support/docview.wss?uid=ibm10738087>

Steps to export the object definitions and messages from one MQ queue manager in host-1 and import them in another queue manager in host-2?

b.5) Even if you try to force the move, the QM might fail to start and/or the queue manager might encounter damaged objects and unexpected failures.

Because this method is not supported, then the IBM MQ Support team will not help you in this situation.

+++ Basic Procedure: Part 1: Actions to be done at the source host

1) Login as user “mqm” and use “setmqenv” to setup the proper MQ environment variables. For example: in this scenario, the queue manager to be moved is associated with Installation2 which is MQ 8.0.0.9:

```
mqm@orizaba1.fyre.ibm.com: /home/mqm
$ . /opt/mqm80/bin/setmqenv -n Installation2
```

```
$ dspmqver -f2
Version: 8.0.0.9
```

Some of the preparation steps require that the queue manager to be Running:

```
$ dspmq -m QMSALES
QMNAME(QMSALES)                STATUS(Running)
```

If it is not Running, then start it:

```
$ strmqm QMSALES
```

2) Create a subdirectory to store backup copies of files generated during this moving process.

The rationale is that this will simplify the creation of a tar/zip file for all the files associated with a queue manager.

For example, create the following subdirectories under /var/mqm

```
mqm@orizaba1.fyre.ibm.com: /var/mqm
$ mkdir /var/mqm/mybackup
```

```
$ mkdir /var/mqm/mybackup/QMSALES
```

That is, the full path for the subdirectory for these backup files is:
/var/mqm/mybackup/QMSALES

3) Remove the queue manager from any MQ clusters.

For this step the queue manager needs to be running.

This step is VERY IMPORTANT! Otherwise you will have 2 different queue managers IDs (QMIDs) in the cluster, but with the same name, which will cause a lot of confusion!!!

https://www.ibm.com/support/knowledgecenter/SSFKSJ_9.1.0/com.ibm.mq.con.doc/q017520.htm

IBM MQ 9.1.x / IBM MQ / Configuring / Configuring distributed queuing / Configuring a queue manager cluster / Removing a queue manager from a cluster

4) (Optional, but recommended) Obtain a backup of the object definitions (in MQSC format) and authorities (in setmqaut format)

For this step the queue manager needs to be running.

Why is this recommended?

It will be a good baseline to have, in that way if there are questions regarding the objects and their attributes, you can review a text file, and you do not need to have running the queue manager.

Notes:

- The output file from saveqmgr or dmpmqcfg is a text file, and you can modify its contents.
- The output file that has the setmqaut commands, includes the name of the queue manager in each command. Thus, if you want to restore the commands into a different queue manager, you will need to edit the file and specify the desired queue manager name.

4.a) For MQ V6 and MQ 7.0: Use the saveqmgr command to capture the object definitions and authorities. This utility is provided with the following SupportPac:

MS03: WebSphere MQ - Save Queue Manager object definitions using PCFs

NOTE: This SupportPac has been withdrawn. Please contact MQ Support for a copy of the latest code.

Example: The following saves both the objects (-f) and the authority records (-z):

Unix:

```
saveqmgr -m QMgr -f /tmp/mq/qmgr_data.mqsc -z /tmp/mq/qmgr_auth.sh
```

Windows:

```
saveqmgr.exe -m QMgr -f C:\temp\mq\qmgr_data.mqsc -z C:\temp\mq\qmgr_auth.bat
```

4.b) For MQ 7.1 or later: The dmpmqcfg tool is new in MQ 7.1 and it is an alternative to the SupportPac MS03.

4.b.1) Capture the object definitions. Specify all attributes (includes also the AUTHRECs for setmqaut)

```
dmpmqcfg -m QMgr -a > QMgr.dmpmqcfg.all.mqsc
```


For example:

```
mqm@orizaba1.fyre.ibm.com: /var/mqm
```

```
$ dmpmqcfg -m QMSALES -a > /var/mqm/mybackup/QMSALES/QMSALES.dmpmqcfg.all.mqsc
```

```
$ ls -l /var/mqm/mybackup/QMSALES/*
```

```
-rw-rw-r-- 1 mqm mqm 92407 Mar 13 13:09 /var/mqm/mybackup/QMSALES/QMSALES.dmpmqcfg.all.mqsc
```

Note:

Some MQ Administrators line to see each object definition in a single line in the output, instead of having “stanzas” that have many lines.

In that case, you could use the flag “-o 1line”:

```
$ dmpmqcfg -m QMSALES -a -o 1line > /var/mqm/mybackup/QMSALES/QMSALES.dmpmqcfg.all.1line.mqsc
```

4.b.2) (Optional) Capture the authentication records in setmqaut format.

Even though the AUTHRECs were included in the output of the command in 4.b.1, most MQ administrators is familiar with the “setmqaut” command and it is useful for them to see the equivalent setmqaut command for the corresponding AUTHREC entries.

The output file should be considered to be a script file in Unix or a batch command file in Windows.

Windows: dmpmqcfg -m QMgr -o setmqaut > QMgr.dmpmqcfg.setmqaut.bat

Unix: dmpmqcfg -m QMgr -o setmqaut > QMgr.dmpmqcfg.setmqaut.sh

For example:

```
$ dmpmqcfg -m QMSALES -o setmqaut > /var/mqm/mybackup/QMSALES/QMSALES.setmqaut.sh
```

```
$ ls -l /var/mqm/mybackup/QMSALES/*
```

```
-rw-rw-r-- 1 mqm mqm 92407 Mar 13 13:09 /var/mqm/mybackup/QMSALES/QMSALES.dmpmqfg.all.mqsc
```

```
-rw-rw-r-- 1 mqm mqm 10224 Mar 13 13:10 /var/mqm/mybackup/QMSALES/QMSALES.setmqaut.sh
```

5) (Optional) You may want to preserve messages from certain non-SYSTEM queues.

For this step the queue manager needs to be running.

Why you may want to do this?

- You may have non-persistent messages in a queue and you do not want to lose them during the transition. Remember that non-persistent messages do not survive a restart of the queue manager.

5.a) For MQ V6, 7.0, 7.1 and 7.5: you can use "qload" to download the messages from a queue into a file, copy the file to the other system, and then use "qload" again to upload the messages from the file into a queue.

<https://github.com/ibm-messaging/mq-q-qload>

mq-q-qload

This repository contains the code from IBM MQ SupportPacs MA01 (q) and MO03 (qload)

Utility Q: A simple pipe line program (Q) which takes messages from one source and outputs to a target.

Utility QLoad: provides a simple command line application for loading and unloading messages off queues. NOTE: MQ 8.0 introduced "dmpmqcfg", developed based on QLoad.

5.b) For MQ V8, V9: you can use "dmpmqmsg" (which is based on "qload") to download the messages from a queue into a file, copy the file to the other system, and then use "dmpmqmsg" again to upload the messages from the file into a queue.

[http://www-](http://www-01.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.adm.doc/q117770_.htm?lang=en)

[01.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.adm.doc/q117770_.htm?lang=en](http://www-01.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.adm.doc/q117770_.htm?lang=en)

WebSphere MQ 8.0.0 > WebSphere MQ > Administering > Administering local WebSphere MQ objects > Using the dmpmqmsg utility between two systems > Examples of using the dmpmqmsg utility

6) If using AMS in MQ 7.0 then you may want to take a backup of the AMS Policies

6.a) For MQ 7.0: The saveqmgr command does NOT export the AMS Policies.

Thus, in case that you want to export these AMS Policies into a file that is part of backup procedures, you can use the command "dspmqspl" with the -export flag, as shown below:

```
dspmqspl -m QMgrName -export > restore_AMS_policies.sh
```

6.b) For MQ 7.1 or later:

Note that the dmpmqcfg command does export the AMS policies (in the output file search for "SET POLICY").

7) Stop the queue manager.

The reason is that the backup of the files must be done while the queue manager is stopped.

For example:

```
mqm@orizaba1.fyre.ibm.com: /var/mqm/mybackup/QMSALES
$ endmqm -i QMSALES
WebSphere MQ queue manager 'QMSALES' ending.
WebSphere MQ queue manager 'QMSALES' ended.
```

8) Obtain the stanza information from mqs.ini for the queue manager by using “dspmqinf”. Store the output in a file in the subdirectory for the backup.

8.1) The following command just prints the information from dspmqinf, without storing it in a file. This is just for illustration purposes regarding dspmqinf.

```
$ dspmqinf -o command QMSALES
addmqinf -s QueueManager -v Name=QMSALES -v Directory=QMSALES -v Prefix=/var/mqm
```

8.2) Run again the dspmqinf but this time store the output into a file

```
$ dspmqinf -o command QMSALES > /var/mqm/mybackup/QMSALES/QMSALES.dspmqinf.txt
```

```
$ ls -l /var/mqm/mybackup/QMSALES/
-rw-rw-r-- 1 mqm mqm 92407 Mar 13 13:09 QMSALES.dmpmqcfg.all.mqsc
-rw-rw-r-- 1 mqm mqm 81 Mar 13 13:12 QMSALES.dspmqinf.txt
-rw-rw-r-- 1 mqm mqm 10224 Mar 13 13:10 QMSALES.setmqaut.sh
```

9) If there are user exits in the source queue manager, then copy them into the subdirectory:

```
/var/mqm/mybackup/QMSALES/
```

10) If you have non-MQ files related to the queue manager that you want to include in the backup file (such as MQSC command files for creating objects, or scripts, or cron job instructions, etc) copy those files into the subdirectory:

```
/var/mqm/mybackup/QMSALES/
```

11) There are several ways to collect files into a single file and to compress the file.

Notice that during this gathering/compress process the subdirectories will be “relative” to the directory /var/mqm in that way, you can restore them relative to the target “/var/mqm” directory (instead of overwriting /var/mqm).

A very common way to do this task is by using the commands “tar” and “gzip”.

```
tar -zcvf CollectingFile.tar.gz subdirectory1 file2 ...
```

The options mean:

- z compress via gzip
- c create archive, which means collect files
- v verbose
- f filename

The result is a large file that has a suffix of:
tar.gz

You could separate the action into two:

a) Gather the files. The resulting archive file is NOT compressed.

```
tar -cvf CollectingFile.tar subdirectory1 file2 ...
```

b) Compress the tar file. The resulting suffix will be renamed to tar.gz.

```
gzip CollectingFile.tar
```

The command to execute is a large one and does not fit into single line in this document. There are 2 ways to handle it.

A) The following splits each logical set of tokens into a single line, ended by backslash.

Note that the LAST line does NOT have the backslash.

You will need to copy one line from this document at a time and paste it in the command prompt.

```
cd /var/mqm
tar -zcvf /tmp/QMSALES.8.0.0.9.tar.gz \
./mybackup/QMSALES \
./log/QMSALES \
./qmgrs/QMSALES
```

B) The following is just a single long command and you have to copy the whole command and paste it in the command prompt.

Even though this document shows 2 lines, it is really a single long line!

```
cd /var/mqm
tar -zcvf /tmp/QMSALES.8.0.0.9.tar.gz ./mybackup/QMSALES ./log/QMSALES
./qmgrs/QMSALES
```

In this scenario, the version-fixpack level of MQ is included in the file name, in that way, it will be easier to find out this information.

```
QmgrName.MQFixPackLevel.tar.gz
QMSALES.8.0.0.9.tar.gz
```

OK, the following is the execution of the command. There are many files that are gathered and to keep this document short, only a handful of entries are shown, to illustrate the subdirectories and files that are gathered.

```
$ cd /var/mqm
$ tar -zcvf /tmp/QMSALES.8.0.0.9.tar.gz ./mybackup/QMSALES ./log/QMSALES
./qmgrs/QMSALES
./qmgrs/QMSALES
./mybackup/QMSALES/
./mybackup/QMSALES/QMSALES.dmpmqcfg.all.mqsc
./mybackup/QMSALES/QMSALES.setmqaut.sh
./mybackup/QMSALES/QMSALES.dspmqinf.txt
./log/QMSALES/
./log/QMSALES/amqhlctl.lfh
./log/QMSALES/active/
./log/QMSALES/active/S0000000.LOG
./log/QMSALES/active/S0000001.LOG
./log/QMSALES/active/S0000002.LOG
./qmgrs/QMSALES/
./qmgrs/QMSALES/ssl/
./qmgrs/QMSALES/plugincomp/
./qmgrs/QMSALES/startprm/
./qmgrs/QMSALES/startprm/ECRC
./qmgrs/QMSALES/errors/
./qmgrs/QMSALES/errors/AMQERR01.LOG
...
./qmgrs/QMSALES/@ipcc/shmem/orizaba1/
./qmgrs/QMSALES/@qmgr/shmem/orizaba1/
./qmgrs/QMSALES/@qmpersist/shmem/orizaba1/
./qmgrs/QMSALES/@app/shmem/orizaba1/
...

$ ls -l /tmp/QMSALES.8.0.0.9.tar.gz
-rw-rw-r-- 1 mqm mqm 109000 Mar 13 13:34 /tmp/QMSALES.8.0.0.9.tar.gz
```

If you are curious about the contents of the tar.gz file you can use the following command, where instead of the “-c” flag (for create/collect), you use the “-t” (for table of contents). Notice that the file permissions and the ownership (mqm/mqm) are shown for each entry.

```
$ tar -ztvf /tmp/QMSALES.8.0.0.9.tar.gz
drwxrwsr-x mqm/mqm          0 2020-03-13 13:12 ./mybackup/QMSALES/
-rw-rw-r-- mqm/mqm        92407 2020-03-13 13:09 ./mybackup/QMSALES/QMSALES.dmpmqcfg.all.mqsc
-rw-rw-r-- mqm/mqm       10224 2020-03-13 13:10 ./mybackup/QMSALES/QMSALES.setmqaut.sh
-rw-rw-r-- mqm/mqm         81 2020-03-13 13:12 ./mybackup/QMSALES/QMSALES.dspmqinf.txt
drwxrwsr-x mqm/mqm          0 2020-03-13 12:54 ./log/QMSALES/
```

```
-rw-rw----  mqm/mqm          6256 2020-03-13 13:11 ./log/QMSALES/amqhlct1.lfh  
...
```

11) Do not delete yet the queue manager. Keep it as a backup, until the transfer is completed and verified. Keep this queue manager stopped, do not restart it.

12) It is a best practice to NOT have different queue managers with the same name! That is, AVOID having QMGR1 in host-1 and another different queue manager also named QMGR1 in host-2. This leads to many headaches!

Thus, after the move is completed and verified, then delete the queue manager from the source location to avoid accidental usage of the old queue manager.

+++ Basic Procedure: Actions to be done at the target host

1) Install the desired version of MQ in the target server machine and apply latest fix packs.

In our test scenario this is the version-fixpack level:

```
mqm@fortin1.fyre.ibm.com: /home/mqm
$ . /opt/mqm/bin/setmqenv -n Installation1
```

```
$ dspmqver -f2
Version: 9.1.0.4
```

1.a) Because the source queue manager was last running on MQ 8.0.0.9, then the target host must be at MQ 8.0 or later.

1.b) If the target host has MQ 9.0 or MQ 9.1, then after restoring the files from the source queue manager in this target host, the first time that strmqm is issued, will detect that the queue manager is not at MQ 9.x but at MQ 8.0, and strmqm will proceed with the upgrade of the queue manager to MQ 9.x.

1.c) Some useful tutorials are:

<http://www.ibm.com/support/docview.wss?uid=ibm10725975>

Installing MQ 9.1.0.0 LTS, applying Fix Pack MQ 9.1.0.1 and uninstalling MQ in Linux RHEL

<http://www.ibm.com/support/docview.wss?uid=ibm10875570>

Upgrading MQ 8.0.0.11 to MQ 9.1.0.1 LTS in Linux

<https://www.ibm.com/support/pages/node/2800923>

Installing MQ 9.1.0.3 LTS to coexist with MQ 8.0.0.9 in Linux x86 64-bit, including applying Fix Packs 8.0.0.14 and 9.1.0.4

<https://www.ibm.com/support/pages/node/2404089>

MQ Distributed: collection of articles regarding installation, upgrade and migration

2) Copy the tar.gz file from the source server onto the new target server machine.

For example, in the target host the file will be stored in:

```
+++ROOT+++ fortin1.fyre.ibm.com: /
# cd /downloads/qmgrs/

# ls -l
-rw-r--r-- 1 root root 109000 Mar 13 13:38 QMSALES.8.0.0.9.tar.gz
```

3) Login as the user “mqm”.

4) Setup the environment variables for the desired installation of MQ:

```
mqm@fortin1.fyre.ibm.com: /home/mqm
$ . /opt/mqm/bin/setmqenv -n Installation1
```

5) Move to the directory /var/mqm

```
$ cd /var/mqm
```

6) List the contents (use flag -t) of the tar.gz file stored elsewhere, but not extract yet the files (do not use yet flag -x):

```
$ tar -ztvf /downloads/qmgrs/QMSALES.8.0.0.9.tar.gz
drwxrwsr-x mqm/mqm          0 2020-03-13 13:12 ./mybackup/QMSALES/
-rw-rw-r-- mqm/mqm        92407 2020-03-13 13:09 ./mybackup/QMSALES/QMSALES.dmpmqcfg.all.mqsc
-rw-rw-r-- mqm/mqm        10224 2020-03-13 13:10 ./mybackup/QMSALES/QMSALES.setmqaut.sh
-rw-rw-r-- mqm/mqm          81 2020-03-13 13:12 ./mybackup/QMSALES/QMSALES.dspmqinf.txt
drwxrwsr-x mqm/mqm          0 2020-03-13 12:54 ./log/QMSALES/
-rw-rw---- mqm/mqm        6256 2020-03-13 13:11 ./log/QMSALES/amqhlctl.lfh
drwxrws--- mqm/mqm          0 2020-03-13 12:54 ./log/QMSALES/active/
```

Because the files inside the tar.gz file are “relative” to the current directory that is “/var/mqm” then the extraction of the files and directories will have an absolute path that begins with “/var/mqm”.

Note: Before proceeding with Step 7, ensure that the absolute paths of the extracted files will be fine. If they do not look right, perhaps you are not in the proper directory /var/mqm or perhaps you did not do the proper packaging step in the source host. It is best to double check at this point that the extracted files and directories will be in their proper place, otherwise you will have to spent time and effort in deleting incorrect entries.

7) Let’s extract the files now, by using the flag -x instead of the flag -t:

```
$ tar -zxvf /downloads/qmgrs/QMSALES.8.0.0.9.tar.gz
```

Now let’s confirm that files, directories, permissions and ownership of the extracted files look fine.

```
$ cd /var/mqm
/var/mqm
```

```
$ ls -dl qmgrs/QMSALES
drwxrwsr-x 22 mqm mqm 4096 Mar 13 13:11 qmgrs/QMSALES
```



```
$ ls -dl qmgrs/QMSALES/*
```

```
-rw-rw-r-- 1 mqm mqm 60 Mar 13 12:54 qmgrs/QMSALES/active  
-rw-rw---- 1 mqm mqm 56 Mar 13 13:11 qmgrs/QMSALES/amqalchk.fil  
drwxrwsr-x 3 mqm mqm 19 Mar 13 12:54 qmgrs/QMSALES/@app  
drwxrws--- 2 mqm mqm 159 Mar 13 12:54 qmgrs/QMSALES/authinfo
```

```
$ ls -dl log/QMSALES/*
```

```
drwxrws--- 2 mqm mqm 66 Mar 13 12:54 log/QMSALES/active  
-rw-rw---- 1 mqm mqm 6256 Mar 13 13:11 log/QMSALES/amqhlctl.lfh
```

```
$ ls -dl mybackup/QMSALES/*
```

```
-rw-rw-r-- 1 mqm mqm 92407 Mar 13 13:09 mybackup/QMSALES/QMSALES.dmpmqcfg.all.mqsc  
-rw-rw-r-- 1 mqm mqm 81 Mar 13 13:12 mybackup/QMSALES/QMSALES.dspmqinf.txt  
-rw-rw-r-- 1 mqm mqm 10224 Mar 13 13:10 mybackup/QMSALES/QMSALES.setmqaut.sh
```

8) Add the queue manager into the master index file of /var/mqm/mqs.ini by using the addmqinf command

It is strongly suggested that you do not alter manually this master index file.

If you feel comfortable though, because you are an experience MQ administrator, then it is OK to go ahead, but be careful because if you make mistakes, you may cause unforeseen problems.

Go to the subdirectory where the output of dspmqinf was captured in the source host:

```
$ cd /var/mqm/mybackup/QMSALES
```

Notice that you should see a file with the output of dspmqinf:

```
$ ls *dspmqinf*  
QMSALES.dspmqinf.txt
```

List the contents of that file:

```
$ cat QMSALES.dspmqinf.txt
```

```
addmqinf -s QueueManager -v Name=QMSALES -v Directory=QMSALES -v Prefix=/var/mqm
```

Copy the output and paste it into a Unix command prompt. You need to run the complete command.

```
$ addmqinf -s QueueManager -v Name=QMSALES -v Directory=QMSALES -v Prefix=/var/mqm  
IBM MQ configuration information added.
```

Show the bottom of the master index to show the new stanza that was added:

```
$ tail /var/mqm/mqs.ini
```

```
QueueManager:
```

```
  Name=QMSALES
```

```
  Prefix=/var/mqm
```

```
  Directory=QMSALES
```

```
  InstallationName=Installation1
```

A quick way to confirm that the update went fine is to list the status of the queue managers.

Notice that QMSALES is recognized as queue manager, but it is not running (yet).

```
$ dspmq
```

```
QMNAME(QMMI1)                STATUS(Status not available)
```

```
QMNAME(QM914)                STATUS(Ended immediately)
```

```
QMNAME(QMSALES)             STATUS(Ended immediately)
```

8.a) Let's see the shmem directories that have the old hostname as extracted from the tar.gz, BUT BEFORE the queue manager is started.

```
mqm@fortin1.fyre.ibm.com: /var/mqm/qmgrs/QMSALES
```

```
$ ls -R | grep shmem
```

```
shmem
```

```
./@app/shmem:
```

```
./@app/shmem/orizaba1:
```

```
shmem
```

```
./@ipcc/shmem:
```

```
./@ipcc/shmem/orizaba1:
```

```
shmem
```

```
./@qmgr/shmem:
```

```
./@qmgr/shmem/orizaba1:
```

```
shmem
```

```
./@qmpersist/shmem:
```

```
./@qmpersist/shmem/orizaba1:
```

9) Review the qm.ini file and if needed, delete old entries or modify existing ones or add new ones.

```
$ cd /var/mqm/qmgrs/QMSALES
```

```
$ vi qm.ini
```

10) If you have user exits in the mybackup directory that was extracted, then you can install them now in the proper directories.

11) Start the queue manager.

NOTICE that in this scenario the queue manager was last running on MQ 8.0 but the running version is MQ 9.1, therefore, the strmqm command will recognize that an upgrade needs to be done and will proceed with the upgrade.

In this example, notice the highlighted line that says that 5 objects were created during the upgrade.

```
$ strmqm QMSALES
```

```
IBM MQ queue manager 'QMSALES' starting.
```

```
The queue manager is associated with installation 'Installation1'.
```

```
5 log records accessed on queue manager 'QMSALES' during the log replay phase.
```

```
Log replay for queue manager 'QMSALES' complete.
```

```
Transaction manager state recovered for queue manager 'QMSALES'.
```

```
Migrating objects for queue manager 'QMSALES'.
```

```
Default objects statistics : 5 created. 0 replaced. 0 failed.
```

```
IBM MQ queue manager 'QMSALES' started using V9.1.0.4.
```

```
$ dspmq -m QMSALES
```

```
QMNAME(QMSALES)                                STATUS(Running)
```

11.a) Notice that the presence of the shmем directories that had the old hostname did not affect the start of the queue manager.

In fact, newer shmем directories were created, which have the new hostname:

```
mqm@fortin1.fyre.ibm.com: /var/mqm/qmgrs/QMSALES
```

```
$ ls -R | grep shmем
```

```
shmем
```

```
./@app/shmем:
```

```
./@app/shmем/fortin1:
```

```
./@app/shmем/orizaba1:
```

```
shmем
```

```
./@ipcc/shmем:
```

```
./@ipcc/shmем/fortin1:
```

```
./@ipcc/shmем/orizaba1:
```

```
shmем
```

```
./@qmgr/shmем:
```

```
./@qmgr/shmем/fortin1:
```

```
./@qmgr/shmем/orizaba1:
```

```
shmем
```

```
./@qmpersist/shmем:
```

```
./@qmpersist/shmем/fortin1:
```

```
./@qmpersist/shmем/orizaba1:
```

12) Visit the remote queue managers that use sender channels that connect to this queue manager:

- ALTER the CONNAME of those sender channels to reflect the new hostname(port)

13) Reset the sender channels of this queue manager at the new location:

- RESET channel(chan_name) seqnum(1)

.

14) Add the queue manager back to any clusters it was in:

https://www.ibm.com/support/knowledgecenter/SSFKSJ_9.1.0/com.ibm.mq.con.doc/q017370_.htm

IBM MQ 9.1.x / IBM MQ / Configuring / Configuring distributed queuing / Configuring a queue manager cluster /

Adding a queue manager to a cluster

15) Perform some verification tasks with the queue manager.

16) Later on, after you are satisfied that the move of the queue manager from the source host to the target host went fine and there are no problems, then you can delete the queue manager from the source host.

+++ end