



**Open Government Platform   
MySQL Rollback Scenarios**

Table of Contents

[Scenario 1: Rollback to the Original Master Server………………………………………………… 3](#_Toc320109258)

Steps Executed on the monitoring server (10.153.12.183)……………………………………3

Steps executed on the Original Master server (10.153.12.186)……………………………….3-4

Steps Executed on thel Slave server; promoted master (10.153.12.187)…………………… 4-8

Scenario 2: Demote Original Master as Slave of Promoted Master…………………………………7

Steps needs to be done on the monitoring server (10.153.12.183)…………………………….7

Steps needs to be done on the Promoted Master server…………………………………………7

Steps needs to be done on the demoted server; slave…………………………………………7-8

# Scenario 1: Rollback to the original master server

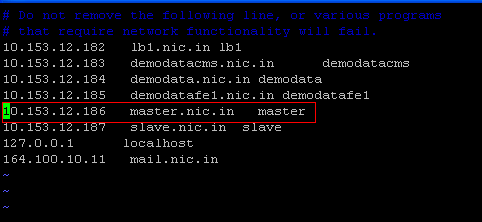
This scenario explains the steps to be followed if after F/O we wanted to rollback to original master server once it comes back again. The disadvantage of this scenario is that in live setup there are chances of data loss.

**In this setup we make the following assumptions with respect to the IP addresses:**

|  |  |
| --- | --- |
| Original Master-Slave | Master:- 10.153.12.186, Slave:- 10.153.12.187 |
| Promoted Master-Slave | Master :- 10.153.12.187, Slave:- Not available |
| Rolled back Master-Slave | Master:- 10.153.12.186, Slave:- 10.153.12.187 |

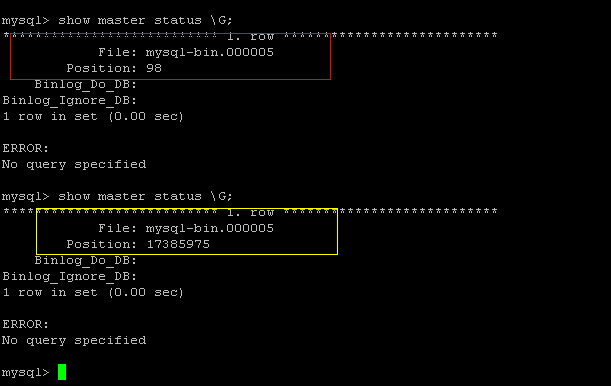
* **Steps Executed on the monitoring server (10.153.12.183)**

1. Stopped mysqld service monitoring cron from the monitoring server.
2. Changed the /etc/hosts file on the server and copied the same file on rest of the systems (Both FE, LB, master-slave)
   1. Replaced the IP address for master server from 10.153.12.187 to 10.153.12.186



* **Steps executed on the Original Master server (10.153.12.186)**

1. Started mysqld service
   1. Service mysqld start
2. Access the application with the URL <http://10.153.12.184> and confirmed that binlogs are getting updated on the original master server.
   1. mysql –u root –pdefault
   2. Show master status \G;



1. Took the backup of data from rollback master server
   1. Before backup stopped the traffic coming to master server
      1. Put the maintenance page on the server and stopped the request coming to DB server so that users will come to know that some activities are going at the background.
   2. Noted down the binlog files and positions

*mysql> show master status \G;*

*File: mysql-bin.000005*

*Position: 24914725*

* 1. [root@master mysql]# mysqldump -u root -pdefault --all-databases --master-data=2 > /mnt/master\_bk\_21march12.sql
  2. [root@master mysql]# ls -lah /mnt/master\_bk\_21march12.sql

-rw-r--r-- 1 root root 560M Nov 6 20:47 /mnt/master\_bk\_21march12.sql

* 1. Copy backedup data to the slave server
     1. [root@master mysql]# scp /mnt/master\_bk\_21march12.sql root@slave:/mnt/

root@slave's password:

master\_bk\_21march12.sql 100% 559MB 22.4MB/s 00:25

* **Steps executed on the slave; 10.153.12.187 (promoted-master)**

1. Stopped MySqld service

[root@master ~]# service mysqld stop

Stopping MySQL: [ OK ]

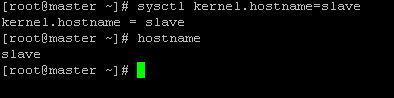
1. Checked /etc/hosts file and confirmed that IP address for “master” server has changed from 10.153.12.187 to 10.153.12.186

|  |  |
| --- | --- |
|  |  |

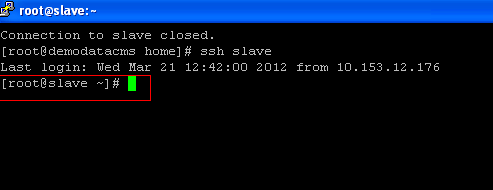
1. Made the changes in /etc/sysconfig/network file. Replaced hostname “master” with the “slave”

|  |  |
| --- | --- |
|  |  |

1. Updated the kernel parameter of the hostname



1. Closed the SSH connection for 10.153.12.187 and connected it again. Confirmed that hostnames and IP address for the promoted master server has got changed to the slave server.

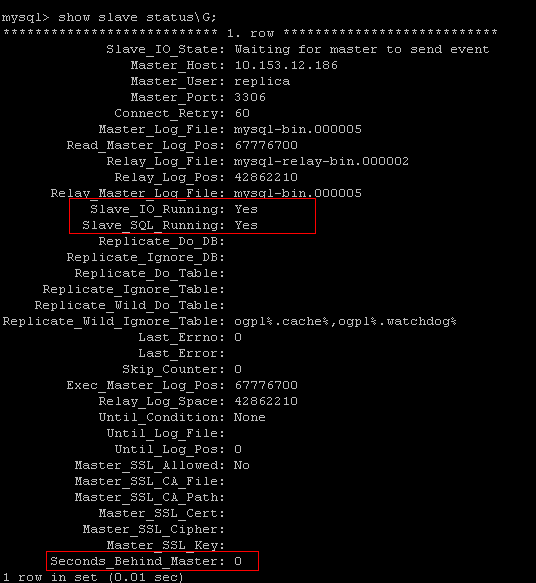


1. Reverted the my.cnf file on the system. During FO, F/O script has copied the original my.cnf file as my\_org.cnf
   1. cd /etc
   2. cp my\_org.cnf my.cnf





1. Started mysql service on the slave server
   1. Service mysqld start
2. Imported the data on the slave server
   1. Cd /var/lib/mysql
   2. mysql -u root -pdefault < /mnt/master\_bk\_21march12.sql
3. login to the slave server
   1. [root@slave mysql]# mysql -u root –pdefault
   2. Execute below commands
      1. Stop slave;
      2. Reset slave;
      3. change master to master\_host='10.153.12.186',master\_user='replica',master\_password='default',master\_port=3306,master\_log\_file='mysql-bin.000005',master\_log\_pos=24914725;
      4. start slave;
      5. show slave status \G;
4. Below screenshot confirmed that slave has got connected to the master server and it is 0 seconds behind the master.



**Scenario 2: Demote Original Master as Slave of Promoted Master**

This scenario will explain the steps to be followed once the slave server gets promoted as new master server and we demote original master server as slave of promoted master.

* **Steps needs to be done on the monitoring server (10.153.12.183)**

1. Change the IP address for the slave to the IP address of the demoted master server
   1. Vi /etc/hosts
   2. If demoted master server’s IP address is 10.153.12.186. change the entry like below in /etc/hosts file
      1. 
   3. Copy the /etc/hosts file on all server (DB, FE, Admin, LB). So that all systems will get updated with the same /etc/hosts file

* **Steps needs to be done on the Promoted Master server**

1. Stop the traffic to the promoted master server
2. **Note down the binlog file and position on the promoted master server**
3. Take the data dump with the mysqldatadump command
   1. mysqldump -u root -pdefault --all-databases --master-data=2 > /mnt/master\_bk\_21march12.sql
   2. [root@master mysql]# ls -lah /mnt/master\_bk\_21march12.sql

-rw-r--r-- 1 root root 560M Nov 6 20:47 /mnt/master\_bk\_21march12.sql

* 1. [root@master mysql]# scp /mnt/master\_bk\_21march12.sql root@slave:/mnt/

root@slave's password:

master\_bk\_21march12.sql 100% 559MB 22.4MB/s 00:25

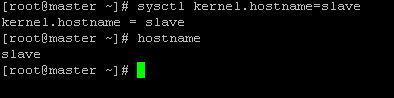
1. Start the traffic on the master server; we have binlog file/position and latest data. With the help of that we can configure a new slave in offline mode.
2. During F/O mysql promote script should have taken the backup of /etc/my.cnf file on promoted server as /etc/my\_org.cnf. copy that file as a /etc/my.cnf on the demoted server

* **Steps needs to be done on the demoted server; slave**

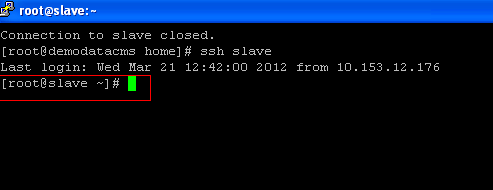
1. Make the changes in /etc/sysconfig/network file. Replace hostname “master” with the “slave”

|  |  |
| --- | --- |
|  |  |

1. Set the hostname kernel parameter and change the name of system to “slave”



1. Close the SSH connection for demoted server and connect it again. Confirmed that hostnames and IP address for the demoted server has got changed to the slave server



1. Open the /etc/my.cnf file and make sure that server-id has set to 2. Make sure that you have copied the proper file from promoted server
2. Start the mysql service on the demoted server
   1. Service mysld start
3. Login in to the mysql prompt and make sure that mysql returns empty sets for “show master status” and “show slave status”
   1. Mysql –u root –p default
   2. show master status;
   3. show slave status;
4. Import the data on the demoted sever
   1. cd /var/lib/mysql
   2. mysql -u root -pdefault < /mnt/master\_bk\_21march12.sql
5. Login to the slave server
   1. [root@slave mysql]# mysql -u root –pdefault
   2. Execute below commands
      1. Stop slave;
      2. Reset slave;
      3. change master to master\_host='10.153.12.186',master\_user='replica',master\_password='default',master\_port=3306,master\_log\_file='mysql-bin.00000X',master\_log\_pos=xxxxxx;
      4. start slave;
      5. show slave status \G;
6. Confirm that “Slave\_IO\_Running” and “Slave\_SQL\_Running” are in “yes” state and “seconds\_Behind\_master” is 0 or some value other than Null.