By Paddy Sreenivasan
Published: 2008-04-11 17:37

MySQL Backups Using ZRM For MySQL 2.0

Zmanda Recovery Manager (ZRM) for MySQL simplifies life of a database administrator who needs an easy to use yet flexible and robust backup and recovery solution for MySQL server. Significant features are:

* Schedule full and incremental logical or raw backups of your MySQL database
* Centralized backup management
* Perform backup that is the best match for your storage engine and your MySQL configuration
* Get e-mail notification about status of your backups
* Monitor and obtain reports about your backups (including RSS feeds)
* Verify your backup images
* Compress and encrypt your backup images
* Implement Site or Application specific backup policies
* Recover database easily to any point in time or to any particular database event
* Custom plugins to tailor MySQL backups to your environment
* MySQL backup using Linux LVM and Solaris ZFS snapshots

Release 2.0 of the community project was released last week. It can be downloaded from **Zmanda downloads** page. It supports all Linux and Solaris distributions. The documentation is available on **ZRM wiki**. **ZRM forums** can be used to get questions answered about the project.

This example assumes that the ZRM server and MySQL server are the same machine. We are backing up MySQL database

myisamnetflix

to the same machine running Ubuntu 7.04.

ZRM For MySQL Installation

* Installation has to be done as super user.

* ZRM for MySQL requires perl 5.8.7 or later. Ubuntu 7.04 already has perl 5.8.8 installed.

* Install perl-DBD and perl-XML-parser modules

```
# apt-get install libxml-parser-perl libdbd-mysql-perl
```

* Download ZRM for MySQL debian packages from **Zmanda downloads** page.

* Install ZRM for MySQL (ZRM server package is sufficient because MySQL server and ZRM server are the same machine).

```
# dpkg -i mysql-zrm_2.0_all.deb
```

```
Selecting previously deselected package mysql-zrm.
(Reading database ... 108342 files and directories currently installed.)
Unpacking mysql-zrm (from mysql-zrm_2.0_all.deb) ...
Setting up mysql-zrm (2.0) ...
Updating ownership of previously backedup data sets
```

MySQL Server Configuration

* Check to see if MySQL server is running. If MySQL server is not installed, please install "mysql-server" using "apt-get" command. Update the "root" MySQL server with a password using mysqladmin command (mysqladmin --user root password boot12). We are using "boot12" as the root password. This user will be used for doing MySQL backups and restores. It is better to user a specific user with minimal privileges to do MySQL backups instead of using "root" MySQL user.

* The MySQL server has to run as "mysql" user and "mysql" OS user should belong to "mysql" group. The default installation of ZRM for MySQL requires MySQL server to run as "mysql" user.

* "ps" output shows mysql server is running using the default MySQL port

```
 mysql    22034 21995  0 14:38 pts/2    00:00:09 /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --user=mysql
--pid-file=/var/run/mysqld/mysqld.pid --skip-external-locking --port=3306 --socket=/var/run/mysqld/mysqld.sock
```

* Enable binary logging on the MySQL server. Binary logging must be enabled to do incremental backups of the MySQL server.

* Edit */etc/mysql/my.cnf* configuration file. Add "log-bin" in mysqld section.

```
[mysqld]
log-bin
```

* We have mysql database "myisamnetflix" that contains two tables. We will be backing this database. This database uses MyISAM storage engine:

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| myisamnetflix      |
| mysql              |
+--------------------+
3 rows in set (0.00 sec)

mysql> use myisamnetflix;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-------------------------+
| Tables_in_myisamnetflix |
+-------------------------+
| MovieID                 |
```

```
| MovieRatings            |
+-------------------------+
2 rows in set (0.00 sec)

mysql> select count(*) from MovieID;
+----------+
| count(*) |
+----------+
|    17770 |
+----------+
```

* MySQL client commands are installed in */usr/bin/* directory. If they are not, accordingly configure the client command location and binary log location in *mysql-zrm.conf*.

ZRM Configuration

* This should be done as *mysql* user:

```
$ id
```

```
uid=1002(mysql) gid=1001(mysql) groups=1001(mysql)
```

* Create the backup set directory. The backup set is called "netflix".

```
$ mkdir /etc/mysql-zrm/netflix
```

* Create *mysql-zrm.conf* configuration file. Backup compression is enabled and "myisamnetflix" database is being backed up. The location of MySQL binary logs are also specified ("mysql-binlog-path").

```
$ cat /etc/mysql-zrm/netflix/mysql-zrm.conf
```

```
host="localhost"
databases="myisamnetflix"
password="boot12"
user="root"
compress=1
mysql-binlog-path="/var/log/mysql"
```

Perform ZRM Backups

* This should be done as "mysql" user.

* Perform full backup of the database immediately using "mysql-zrm-scheduler".

```
$ mysql-zrm-scheduler --now --backup-set netflix --backup-level 0
```

```
schedule:INFO: ZRM for MySQL Community Edition - version 2.0
Logging to /var/log/mysql-zrm/mysql-zrm-scheduler.log
backup:INFO: ZRM for MySQL Community Edition - version 2.0
netflix:backup:INFO: START OF BACKUP
netflix:backup:INFO: PHASE START: Initialization
netflix:backup:INFO: backup-set=netflix
netflix:backup:INFO: backup-date=20080326161652
netflix:backup:INFO: mysql-server-os=Linux/Unix
netflix:backup:INFO: host=localhost
netflix:backup:INFO: backup-date-epoch=1206573412
netflix:backup:INFO: mysql-zrm-version=ZRM for MySQL Community Edition - version 2.0
netflix:backup:INFO: mysql-version=5.0.38-Ubuntu_0ubuntu1.4-log
netflix:backup:INFO: backup-directory=/var/lib/mysql-zrm/netflix/20080326161652
netflix:backup:INFO: backup-level=0
netflix:backup:INFO: backup-mode=raw
netflix:backup:INFO: PHASE END: Initialization
```

```
netflix:backup:INFO: PHASE START: Running pre backup plugin
netflix:backup:INFO: PHASE END: Running pre backup plugin
netflix:backup:INFO: PHASE START: Flushing logs
netflix:backup:INFO: PHASE END: Flushing logs
netflix:backup:INFO: PHASE START: Find table type
netflix:backup:INFO: PHASE END: Find table type
netflix:backup:INFO: PHASE START: Creating raw backup
netflix:backup:INFO: raw-databases=myisamnetflix
netflix:backup:INFO: PHASE END: Creating raw backup
netflix:backup:INFO: PHASE START: Calculating backup size & checksums
netflix:backup:INFO: next-binlog=mysql-bin.000009
netflix:backup:INFO: backup-size=122.27 MB
netflix:backup:INFO: PHASE END: Calculating backup size & checksums
netflix:backup:INFO: PHASE START: Compression/Encryption
netflix:backup:INFO: compress=
netflix:backup:INFO: backup-size-compressed=37.65 MB
netflix:backup:INFO: PHASE END: Compression/Encryption
netflix:backup:INFO: read-locks-time=00:00:01
netflix:backup:INFO: flush-logs-time=00:00:00
netflix:backup:INFO: compress-encrypt-time=00:02:20
netflix:backup:INFO: backup-time=00:00:15
netflix:backup:INFO: backup-status=Backup succeeded
netflix:backup:INFO: Backup succeeded
netflix:backup:INFO: PHASE START: Running post backup plugin
netflix:backup:INFO: PHASE END: Running post backup plugin
netflix:backup:INFO: PHASE START: Mailing backup report
netflix:backup:INFO: PHASE END: Mailing backup report
netflix:backup:INFO: PHASE START: Cleanup
netflix:backup:INFO: PHASE END: Cleanup
netflix:backup:INFO: END OF BACKUP
/usr/bin/mysql-zrm started successfully
```

** Delete some entries from the "myisamnetflix" database (so that we can do incremental backup of the database)*

```
mysql> use myisamnetflix;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed

mysql> delete from MovieID where MovieTitle = "Alien Hunter";
Query OK, 1 rows affected (0.01 sec)
```

* Perform incremental backup of the backup set.

```
$ mysql-zrm-scheduler --now --backup-set netflix --backup-level 1
```

```
schedule:INFO: ZRM for MySQL Community Edition - version 2.0
Logging to /var/log/mysql-zrm/mysql-zrm-scheduler.log
backup:INFO: ZRM for MySQL Community Edition - version 2.0
netflix:backup:INFO: START OF BACKUP
netflix:backup:INFO: PHASE START: Initialization
netflix:backup:INFO: backup-set=netflix
netflix:backup:INFO: backup-date=20080326164433
netflix:backup:INFO: mysql-server-os=Linux/Unix
netflix:backup:INFO: host=localhost
netflix:backup:INFO: backup-date-epoch=1206575073
netflix:backup:INFO: mysql-zrm-version=ZRM for MySQL Community Edition - version 2.0
netflix:backup:INFO: mysql-version=5.0.38-Ubuntu_0ubuntu1.4-log
netflix:backup:INFO: backup-directory=/var/lib/mysql-zrm/netflix/20080326164433
netflix:backup:INFO: backup-level=1
netflix:backup:INFO: PHASE END: Initialization
netflix:backup:INFO: PHASE START: Running pre backup plugin
netflix:backup:INFO: PHASE END: Running pre backup plugin
netflix:backup:INFO: PHASE START: Flushing logs
netflix:backup:INFO: PHASE END: Flushing logs
```

```
netflix:backup:INFO: PHASE START: Creating incremental backup
netflix:backup:INFO: incremental=mysql-bin.[0-9]*
netflix:backup:INFO: PHASE END: Creating incremental backup
netflix:backup:INFO: PHASE START: Calculating backup size & checksums
netflix:backup:INFO: next-binlog=mysql-bin.000013
netflix:backup:INFO: last-backup=/var/lib/mysql-zrm/netflix/20080326162210
netflix:backup:INFO: backup-size=0.03 MB
netflix:backup:INFO: PHASE END: Calculating backup size & checksums
netflix:backup:INFO: PHASE START: Compression/Encryption
netflix:backup:INFO: compress=
netflix:backup:INFO: backup-size-compressed=0.00 MB
netflix:backup:INFO: PHASE END: Compression/Encryption
netflix:backup:INFO: read-locks-time=00:00:00
netflix:backup:INFO: flush-logs-time=00:00:00
netflix:backup:INFO: compress-encrypt-time=00:00:00
netflix:backup:INFO: backup-time=00:00:00
netflix:backup:INFO: backup-status=Backup succeeded
netflix:backup:INFO: Backup succeeded
netflix:backup:INFO: PHASE START: Running post backup plugin
netflix:backup:INFO: PHASE END: Running post backup plugin
netflix:backup:INFO: PHASE START: Mailing backup report
netflix:backup:INFO: PHASE END: Mailing backup report
netflix:backup:INFO: PHASE START: Cleanup
netflix:backup:INFO: PHASE END: Cleanup
netflix:backup:INFO: END OF BACKUP
/usr/bin/mysql-zrm started successfully
```

ZRM Backup Reports

* Use "mysql-zrm-reporter" to look at the status of backups available.

```
$ /usr/bin/mysql-zrm-reporter --where backup-set=netflix --show backup-status-info
```

```
 REPORT TYPE : backup-status-info

          backup_set  backup_date                  backup_level  backup_status         comment
 ------------------------------------------------------------------------------------------------------
             netflix  Wed 26 Mar 2008 04:44:33                1  Backup succeeded      ----
                      PM PDT
             netflix  Wed 26 Mar 2008 04:16:52                0  Backup succeeded      ----
                      PM PDT
```

* ZRM reports can also provide information on impact on MySQL application.

```
$ /usr/bin/mysql-zrm-reporter --where backup-set=netflix --show  backup-app-performance-info
```

```
 REPORT TYPE : backup-app-performance-info

          backup_set  backup_date                  backup_level     backup_size  backup_time   read_locks_time
flush_logs_time
-------------------------------------------------------------------------------------------------------------------
--------
             netflix  Wed 26 Mar 2008 04:44:33                1         0.03 MB  00:00:00      00:00:00            00:00:00
                      PM PDT
             netflix  Wed 26 Mar 2008 04:16:52                0       122.27 MB  00:00:15      00:00:01            00:00:00
                      PM PDT
```

Database Recovery

* Use ZRM reporting tool to identify the location of MySQL backup images.

```
$ /usr/bin/mysql-zrm-reporter --where backup-set=netflix --show restore-info
```

```
 REPORT TYPE : restore-info
```

```
          backup_set  backup_date                  backup_level  backup_directory
backup_status         comment
-------------------------------------------------------------------------------------------------------------------
------------------------
             netflix  Wed 26 Mar 2008 04:44:33               1  /var/lib/mysql-zrm/netflix/20080326164433  Backup
succeeded      ----
                      PM PDT
             netflix  Wed 26 Mar 2008 04:16:52               0  /var/lib/mysql-zrm/netflix/20080326161652  Backup
succeeded      ----
                      PM PDT
```

* You can parse incremental backups to identify database events of interest. In our example, we will look for the "DELETE" event.

```
$ /usr/bin/mysql-zrm-parse-binlogs --source-directory /var/lib/mysql-zrm/netflix/20080326164433 | grep delete
```

```
 parse-binlogs:INFO: ZRM for MySQL Community Edition - version 2.0
 /var/lib/mysql-zrm/netflix/20080326164433/mysql-bin.000011 | 13634 | 08-03-26 16:28:03 | Query | use myisamnetflix/*!*/;
delete from MovieID where MovieTitle = "Alien Hunter"/*!*/;
```

* Restore the database from the full backup done at 16:16:52.

```
$ /usr/bin/mysql-zrm-restore --user=root --password=boot12 --source-directory=/var/lib/mysql-zrm/netflix/20080326161652
```

```
 restore:INFO: ZRM for MySQL Community Edition - version 2.0
 BackupSet1:restore:INFO: Restored database from raw backup: myisamnetflix
 BackupSet1:restore:INFO: Restore done in 9 seconds.
 MySQL server has been shutdown. Please restart after verification.

* Restart the MySQL server
 # /etc/init.d/mysql restart
 * Stopping MySQL database server mysqld                                [ OK ]
```

```
 * Starting MySQL database server mysqld                                [ OK ]
 * Checking for corrupt, not cleanly closed and upgrade needing tables.
```

* Check the database recovery.

```
mysql> use myisamnetflix;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from MovieID where MovieTitle = "Alien Hunter";
+---------+------+--------------+
| MovieID | Year | MovieTitle   |
+---------+------+--------------+
|   17770 | 2003 | Alien Hunter |
+---------+------+--------------+
1 row in set (0.02 sec)
```