

The Database Experimental Ranch: MySQL Lab Setup

Amit Jain

Introduction

Each user will run their own MySQL database server. So you will act as the Database administrator, Database programmer and Database user! The various MySQL database servers will be kept separate by using a different TCP/IP port and a different socket for each user. For the rest of this document, we will assume that the login name for our user is `dbkool`, and that their assigned port is 3333. The socket that is created will be `/tmp/mysql-dbkool.sock`, where `dbkool` is your login name.

The MySQL database will only run on `onyx`. However you will be able to connect it from any machine in the lab (as well as from home by tunneling via Secure Shell).

Setup

1. Get a port number in the range 3309-3345 from your instructor. Suppose you have been assigned 3333.
2. Log in to `onyx`.
3. Run the local setup script with your assigned port number as an argument.

```
mysql-setup-local 3333
```

The script will remove any existing MySQL directory from your account and re-initialize a new setup. It will also ask for a root/user password for the MySQL database. Please use a good password (but not the same as the one for your login account). For the curious, the script is located at `/usr/local/bin/mysql-setup-local`.

4. The database main directory will be `$HOME/mysql`. You will have full access to it, so be careful if you go peeking around there!
5. Note that the `mysql-setup-local` script will modify your `$HOME/.bash_rc` file. It also adds a file `$HOME/.mysqlrc` that is invoked when you login to set up MySQL environment variables.
6. Note that the setup script also creates a config file `$HOME/.my.cnf` that contains your password. If you ever change your password in MySQL, you will need to change this file as well. This file is readable only by you. Having this file eliminates having to type your password again and again for certain MySQL clients.
7. **Using the Administrator GUI.** Use the GUI tool `mysql-administrator` and check what privileges root and you have. **Don't forget to add an account for your instructor and give them appropriate privileges.**
8. The following commands are available after the setup script is run.

- `mysql-check` Check if the database server is running.
 - `mysqlshow` Check what tables are in your database.
 - `mysql-stop` Stop the database server.
 - `mysql-start` Start the database server.
9. **What if I want to setup MySQL again in my account from scratch?** Just remove the lines added by the setup-script in `.bashrc`. Then run the `mysql-setup-local` script again as shown earlier.
 10. **How do I restart the MySQL server.** Stop the server with the command `mysql-stop` and then start the server with the command `mysql-start`.
 11. **Does the MySQL database server keep running when I log out?** Yes. You should normally leave the server running at all times so you can connect from home or other places.

Using MySQL

- **Using the query browser GUI.** Use the GUI query-browser `mysql-query-browser` for running mysql queries and other operations on your databases. Remember to give your assigned port number and (under the Details button) also specify the socket file, which is `/tmp/mysqlld-dbkool.sock` with `dbkool` replaced by your login name.
- **Using the command line interface.** The client `mysql` gives you a command line interface to MySQL. This is convenient if you are coming over the network to `onyx`. It can also be used to run SQL statements in batch mode. For example, if you have all your SQL statements typed in a file, say `queries.sql`, then you can run them with the command `source queries.sql;` in the `mysql` command line interface. See an example session below.

```
[amit@onyx soln]$ mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4 to server version: 4.1.10-standard-log
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

```
mysql> use company;
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> source queries.sql;
...
<lots of output>
...
mysql> show warnings;
...

```

```
<hopefully, not much output :-)>
...
mysql> quit
```

You can even redirect them in as:

```
mysql < queries.sql
```

- **The command line administrator.** There is also a command line admin program called `mysqladmin`. For example, you can shutdown the server with the command:
`mysqladmin shutdown`
- **How to access your database server from any workstation in the lab.** Log in to one of the lab workstations. In one terminal window, login to `onyx` to start the server if it isn't already running. Now startup the `mysql-query-browser` and use `ws00` (the name for `onyx` on our private network in the lab), your password and your socket file name. Then you can work on your databases from any workstation in the lab!
- **Dumping your database in SQL.** Use the `mysqldump` command to export your entire database into SQL statements. This can be very convenient in migrating your database to another database server. An example is shown below.

```
[amit@onyx amit] mysqldump company > company.sql
```

- **Making a hot copy of your database.** The term “hot copy” refers to making a consistent copy of a database that is being used. Use the `mysqlhotcopy` command on `onyx`. An example is shown below.

```
[amit@onyx amit] mysqlhotcopy company mysql.backup/
Locked 6 tables in 0 seconds.
Flushed tables ('company'.'department', 'company'.'dependent',
'company'.'dept_locations', 'company'.'employee', 'company'.'project',
'company'.'works_on') in 0 seconds.
Copying 19 files...
Copying indices for 0 files...
Unlocked tables.
mysqlhotcopy copied 6 tables (19 files) in 0 seconds (0 seconds overall).
[amit@onyx amit]
```

Installing on your computer

MS Windows

Go the MySQL website (mysql.com) and download the self-installing package. Also don't forget to download the `mysql-administrator` and `mysql-query-browser`.

Linux

These instructions are for Fedora Core 5.

Linux usually comes installed with MySQL server software as well as the command line software. You can check if you it installed with the command.

```
yum list installed mysql*
```

If that doesn't show a bunch of software packages, then install it as follows:

```
yum install mysql*
```

This assumes that you are connected to the internet.

The GUI tools are available from mysql.com. We have a downloaded copy available on [onyx](#) under [~amit/tmp/mysql-gui-tools-5.0r9-fc5-i386.tar.gz](#)

Download that to your system. In order to install these, you will typically need to add some additional software. The steps that you need are shown below.

```
tar xzvf mysql-gui-tools-5.0r9-fc5-i386.tar.gz
su
yum install gtkmm*
yum install mysql*rpm
```

Remote Access to MySQL

Normally MySQL runs on port 3306, which is blocked by the firewall (a good thing). Your MySQL on [onyx](#) will be running on some other port in the range [3309-3345](#), which are also blocked by the firewall (another good thing). However, you can still use the tools [mysql-query-browser](#) and [mysql-administrator](#) from home to connect to your server on [onyx](#) by tunneling via Secure Shell.

First you will need to use the [mysql-administrator](#) and add the host [onyx.boisestate.edu](#) under your user name (by right-clicking on the user name and then choosing the *Add Host* option) and grant the appropriate privileges.

Linux

Suppose your assigned port is 3333 and login id is [dbkool](#). Then login to [onyx](#) with the following command.

```
ssh -L 3333:onyx:3333 dbkool@onyx
```

Now tunneling has been established. Use [mysql-query-browser](#) or [mysql-administrator](#) and specify the host address as [127.0.0.1](#) (instead of [localhost](#)), specify the correct port and socket file name and viola you are now sitting at home but using the MySQL server at [onyx](#). If you get "access denied" message, then you need to use the [mysql-administrator](#) and add the host [onyx.boisestate.edu](#) under your user name and grant the appropriate privileges. And yet our system admin is happy since he does not have to worry about holes in MySQL. How cool is that?

MS Windows

This depends upon which [ssh](#) client you are using. Tunneling can be setup with either the client from [ssh.com](#) or with [putty](#). See this article on how to set it up:

<http://www.vbmysql.com/articles/security/gui-tunnel.html>

Once you have established tunneling, then use [mysql-query-browser](#) or [mysql-administrator](#) and specify the host address as [127.0.0.1](#) (instead of [localhost](#)), specify the correct port and socket file name and viola you are now sitting at home but using the MySQL server at [onyx](#).

References

1. The MySQL website. <http://www.mysql.com>. Downloads, documentations, user forums etc. Lots of good stuff here.