Packaging MySQL on Andrew Linux

Stephen Hathorne
Unix Systems Administrator
Infrastructure, Systems, Applications and Middleware
Carnegie Mellon
slh+@cmu.edu

This document is a tutorial on how to install MySQL on an Andew Linux system.

Assumptions

This document assumes that you have some base familiarity with Systems Administration, administrative privileges on the system(s) that you are about to modify, and a passing familiarity with the *Andrew Package System*[1]. Finally, you have an understanding of how to modify files, and more importantly which files you **shouldn't** modify.

Conventions

$> Represents a system prompt do not type this.
Items listed after the $> prompt are to be typed:

Example:

$> ls

This example indicates for the reader to type the **ls** command after the $> prompt.

Items listed with borders and gray shaded represent file examples:

```
%define doesreallycoolstuff
```

Items listed in plain Arial Text 12 point font are explanatory in nature, and will comprise most of the document.

[1] See the Andrew Package documentation in Appendix A: *Additional Resources*

Italicized items represent filenames or special instructions

Bold elements generally represent section headings or important topics

Step by Step Installation

1. First, we will create an /etc/my.cnf file to configure the attributes of the MySQL database engine. The example file that we will use, uses the INNODB model which makes the database engine more robust.

2. Next we will need to create the data directory, which is referred to as the *datadir* in /etc/my.cnf, and in various places within the document.

3. Modify */etc/package.proto* to include the *%doesmysql* definition, and to protect our version of */etc/my.cnf* or optionally include a file from an *AFS* path.

4. Once */etc/package.proto* completes we will need to run /etc/mpp-package to add the files included in our package definition to our Andrew Linux System.

5. Become the user *database* to initialize and configure the MySQL engine.
6. Restart the MySQL database
7. Set the MySQL root (administrative) password
8. Create a simple test database and user.

I. Create an /etc/my.cnf file

The file */etc/my.cnf* is the main configuration file for the ***MySQL*** database engine. The file below is a sample currently in use.

```
[mysqld]
datadir=/usr/mysql/data/mysql
innodb_data_file_path = d1:2000M
innodb_data_home_dir = /usr/mysql/data/mysql/idb
innodb_log_group_home_dir = /usr/mysql/data/mysql/idb/log
innodb_log_arch_dir = /usr/mysql/data/mysql/idb/log
set-variable = innodb_mirrored_log_groups=1
set-variable = innodb_log_files_in_group=5
set-variable = innodb_log_file_size=200M
set-variable = innodb_log_buffer_size=100M
innodb_flush_log_at_trx_commit=1
innodb_log_archive=0
set-variable = innodb_buffer_pool_size=100M
set-variable = innodb_additional_mem_pool_size=4M
```

```
set-variable = innodb_file_io_threads=4
set-variable = innodb_lock_wait_timeout=50
```

The important items to note in the above file is the **datadir** This is the directory where the MySQL data files will live. There are also two (2) additional subdirectories defined in the file.

Innodb_data_home_dir=/usr/mysql/data/mysql/idb
Innodb_log_group_home_dir=/usr/mysql/data/mysql/idb/log

As you can see from the file these two directories are used to define a couple of other variables

Step I is now complete

II. Create the Data Directory, set ownership and permissions.

$>mkdir –p /usr/mysql/data/mysql/idb/log
$> chown –R database /usr/mysql/data

These two (2) commands create the directories necessary to hold our database data, and sets the permissions and ownership for the user database to be able to access the data. [2]

Step II is now complete

[2] mysql runs as *database*

III Modify /etc/package.proto, and run mpp-package

Our */etc/package.proto* file needs to be modified to contain the package definition to cause the package system to install MySQL locally.

Sample /etc/package.proto

```
%define doessshd
%define usesdepot
%define localdepotdir
%define contribdepotdir
%define autolinuxspec
%define doesmysql
F /etc/my.cnf
%include /afs/andrew.cmu.edu/wsadmin/public/src/public.proto
```

This package.proto file adds apache, php4, the ssh server, and MySQL to the workstation. Notice the F lines. These lines indicate to the package subsystem that the filename that follows should not be replaced. Optionally you may put a file path from AFS space to put in place:

Example:

F /usr/fnork /afs/Andrew.cmu.edu/wsadmin/public/fnork

This example tells the package subsystem to put the file /afs/Andrew.cmu.edu/wsadmin/public/ fnork in /usr/fnork.
(For additional information on package see Appendix A: Additional Resources)

The other item to notice is the ^^***%define doesmysql"*** line. This line tells the package system to grab all of the information from the wsadmin tree tht relates to the mysql database engine, and install those files locally.

Once the */etc/package.proto* file is in place, you will need to run the ***/etc/mpp-package*** script. This script will apply the definitions from the package file in order to update the machine.

$> /etc/mpp-package

Step III is now complete

IV. Become the database user and initialize the database

In this step we will need to use the UNIX **su** command to become the database user. This will allow us to initialize the database engine, and restart the MySQL program, as well as initially set the MySQL admin password.

$> su database

The above command should be issued from your root instance. Notice the command is not su – database, since the database account has no home directory.

$>mysql_install_db –datadir=/usr/mysql/data/mysql

This program initializes the database engine. The *datadir* variable tells the program where the Data Directory is.

$>/etc/init.d/mysql restart

This execution restarts the mysql database program from the /etc/init.d directory.

$>/usr/local/bin/mysqladmin –u root password 'New Admin Password'

This command allows the administrator to set the mysql administrative password, since when the database is initialized no password is set.

Note: Be sure to set a password!

$>exit

exit the database shell.

Step IV is now complete.

V. Create a simple test database, create a user and grant access

In this section we will execute a few simple SQL commands to create a database, enable a new user. Log in as that user, and create a small test table.

Login as the mysql administrator:

$> mysql –u root –p
password: ********

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 4.0.13

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

mysql>create database *yourdatabasenamehere;*
Query OK, 1 row affected (0.02 sec)

mysql> grant all on yourdatabasenamehere.* to userid@localhost identified by 'password';

Query OK, 0 rows affected (0.00 sec)
mysql> exit
Bye

In the above session, you are logging into the mysql instance as the database administrator[3]. A command is then issued to create a new database. The response "Query OK, 1 row affected (0.02 sec) tells us that the mysql engine has created our database. The next command *"grant all on yourdatabasenamehere.* to userid@localhost identified by new password"* accomplishes two (2) things at once. The first is that it creates the new user "userid@localhost", the second is that the user is granted all privileges on the newly created database. ***Carefully note the use of databasename.* here*** this tells the mysql program to not only grant the user privileges on the database but also on *any subsequent objects*. This allows the user to create tables, and indexes. We then issue the exit command which leaves the database program.

Return to your own userid by exiting both the database user, and the root instance.

$>mysql –u useridfromabove –p
Enter password: ******
mysql> use yourdbnamehere;**
Database changed
mysql> create table fnork(name varchar(30));
Query OK, 0 rows affected (0.00 sec)
mysql> insert into fnork values('James Isaac Neutron');
Query OK, 1 row affected (0.00 sec)
mysql> select * from fnork;

[3] I find it amusing that the database administrator is "root" rather than sysdba or dba…:*)

```
mysql> select * from fnork;
+--------------------------------+
| name                           |
+--------------------------------+
| James Isaac Neutron-- |
+--------------------------------+
1 row in set (0.00 sec)
mysql> drop table fnork;
Query OK, 0 rows affected (0.00 sec)
mysql> exit;
bye
```

In the brief session above we are testing our installation and configuration by creating a simple table, and inserting a value into it. Once that is done, we drop the table and exit the database.

Step V is now complete, and the database is installed.

See Appendix A: Additional Resources for additional information.

Appendix A: Additional Resources

Package Documentation

http://www.cmu.edu/computing/documentation/andrw_package/package.html

MySQL online manual

http://dev.mysql.com/doc/