

FUJITSU Cloud Service S5

Installation and Configuration of MySQL on CentOS Server

This guide details the steps required to install and configure MySQL on a CentOS Server

Introduction

The FUJITSU Cloud Service S5 is an open platform and in the interest of openness, this guide will form part of a series dedicated to using the FUJITSU Cloud Service S5 in an open source context. This guide will detail the steps required to install, configure and set-up MySQL ("The world's most popular open source database") from an RPM on a CentOS server on the FUJITSU Cloud Service S5.

Red Hat Package Manager or RPM Package Manager (RPM) is a file package format containing the compiled version of software for installation. This guide is based on installing MySQL from an RPM package. MySQL is a relational database and a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python). More and more organisations are using LAMP because of its lower cost and freedom from platform lock-in.

Assumptions

- User Account, Certificate and connectivity to the FUJITSU Cloud Service S5.
- Familiarity with the FUJITSU Cloud Service S5 Portal basic configuration and administration tasks.
- Familiarity with the FUJITSU Cloud Service S5 Portal System Manager and Design Studio.
- Familiarity with Linux / CentOS configuration and administration tasks.
- CentOS Economy Server on the FUJITSU Cloud Service S5 with an additional disk.
- Familiarity with MySQL and associated administration (further information is available at: <http://dev.mysql.com/doc/refman/5.5/en/>)
- Copy of the MySQL Server Generic RPM Package either 32-bit or 64-bit (MySQL Community Server 5.5.18 will be used here).

Obtaining a copy of the MySQL & Client Tools RPM Packages

1. MySQL (<http://www.mysql.com/>) Community Server Generally Available (GA) Release: 5.5.18 will be used.
2. FUJITSU Cloud Service S5 provides CentOS servers in either 32-bit and 64-bit architecture variants. When the downloading MySQL, choose a **Linux - Generic RPM Package** either **32-bit** or **64-bit** as appropriate to the architecture of the proposed CentOS server.
3. A free MySQL Community Edition server may be downloaded from <http://www.mysql.com/downloads/mysql/>.
4. In this example **Linux - Generic 2.6 (x86, 64-bit), RPM Package MySQL Server** was selected
5. When complete, download the MySQL Client Tools. In this example **Linux - Generic 2.6 (x86, 64-bit), RPM Package Client Utilities** was selected.
6. When the MySQL package has been downloaded, create a VPN Connection and remote copy the packages to the target server via SSH (see FUJITSU Cloud Service S5 - Setup and Configuration of Secure Shell to perform remote a copy with CentOS Servers).

Installing MySQL - Preparation

1. Login to the target CentOS Server as root
2. Add a dedicated mysql user account. Use `useradd <user-name>` followed by typing the ENTER key,

```
[root@localhost mysql]# useradd mysql
```

3. Add a dedicated mysql group. Use `groupadd <group-name>` followed by typing the ENTER key,

```
[root@localhost mysql]# groupadd mysql
groupadd: group mysql exists
```

4. Assign the mysql user to the mysql group using `useradd -g <group-name> <user-name>` followed by ENTER,

```
[root@localhost mysql]# useradd -g mysql mysql
useradd: user mysql exists
```

5. Set the mysql user password,

```
[root@localhost mysql]# passwd mysql
```

6. When installing any database it is good practice to keep the software binaries separate from data and log files. A CentOS Economy Server on the FUJITSU Cloud Service S5 includes a 10GB system disk, MySQL binaries will reside here. For the purpose of keeping data and logs files separate, this guide will assume an additional disk mounted under the following path, `/home/mysql`. For more information see, FUJITSU Cloud Service S5 - Installation and Configuration of MySQL on CentOS Servers.

7. Create directory structure by typing the following commands each followed by the ENTER key,

```
[root@localhost mysql]# cd /home/mysql
[root@localhost mysql]# mkdir console
[root@localhost mysql]# mkdir data
[root@localhost mysql]# mkdir log
```

8. An explanation of the directory structure is as follows:
 - **console** directory contains any MySQL ASCII type log files such as the console / stdout, slow-query logs etc.
 - **data** directory holds the MySQL data files.
 - **log** directory is where MySQL transaction logs will reside.
9. Ensure correct directory access using the syntax followed by ENTER,

```
[root@localhost mysql]# chown -R mysql:mysql console data log
```

Installing MySQL - Setup

10. Install MySQL server. This guide is based on a Generic Linux RPM install of MySQL. The MySQL-server-5.5.18-1.linux2.6.x86_64 RPM package being used is NOT relocatable. To install type, `rpm -Uvh <rpm-file-name>` followed by ENTER,

```
[root@localhost install]# rpm -Uvh MySQL-server-5.5.18-1.linux2.6.x86_64.rpm
Preparing...                ##### [100%]
 1:MySQL-server             ##### [100%]

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:

/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h localhost.localdomain password 'new-password'

Alternatively you can run:
/usr/bin/mysql_secure_installation

which will also give you the option of removing the test
databases and anonymous user created by default.  This is
strongly recommended for production servers.

See the manual for more instructions.

Please report any problems with the /usr/bin/mysqlbug script!

http://dev.mysql.com/doc/refman/5.5/en/linux-installation-rpm.html
```

10. MySQL will be installed in the following directories,

- /usr/bin, client programs and scripts.
- /usr/sbin, the mysqld server binaries. (32bit only? Cant see in 5 or 6 for)
- /var/lib/mysql, log files and databases.
- /usr/share/info, MySQL manual in Info format.
- /usr/share/man, MySQL Linux manual pages.
- /usr/include/mysql, include (header) files.
- /usr/lib64/mysql, 64 bit libraries.
- /usr/share/mysql, miscellaneous support files, including error messages, character set files, sample configuration files, SQL for database installation.
- /usr/share/sql-bench, benchmarks.

11. Ownership of Server. If necessary, ensure that the distribution contents are accessible to the mysql user and group created previously. If the installation was performed as mysql, no further action is required. If the distribution was installed as root, the contents will be owned by root. Therefore it is essential to change its ownership to mysql. This may be done by executing the following commands as in the installation directory. The first command changes the owner attribute of the files to the mysql user. The second changes the group attribute to the mysql group,

■ /usr/bin,

```
[root@localhost bin]# cd /usr/bin
[root@localhost bin]# chown mysql:mysql mysql*
```

■ /usr/sbin

```
[root@localhost bin]# cd /usr/sbin
[root@localhost sbin]# ls -l mysql*
-rwxr-xr-x 1 root root 45151874 Nov 14 00:07 mysqld
-rwxr-xr-x 1 root root 44431554 Nov 13 23:59 mysqld-debug
[root@localhost sbin]# chown mysql:mysql mysql*
```

■ /var/lib/mysql,

```
[root@localhost lib]# cd /var/lib
[root@localhost lib]# chown -R mysql:mysql mysql
[root@localhost lib]# ls -l mysql
total 20
drwx--x--x 2 mysql mysql 4096 Nov 29 15:56 mysql
drwx----- 2 mysql mysql 4096 Dec  1 11:34 performance_schema
-rw-r--r-- 1 mysql mysql  766 Dec  1 11:13 RPM_UPGRADE_HISTORY
-rw-r--r-- 1 mysql mysql  325 Dec  1 11:13 RPM_UPGRADE_MARKER-LAST
drwxr-xr-x 2 mysql mysql 4096 Nov 29 15:56 test
```

■ /usr/share/info, (Centos 5.x only) not present on 6

```
[root@localhost lib]# cd /usr/share/info
[root@localhost info]# ls -l mysql*
-rw-r--r-- 1 root root 2887931 Nov 13 23:47 mysql.info.gz
[root@localhost info]# chown mysql:mysql mysql.info.gz
```

■ /usr/share/mysql,

```
[root@localhost share]# cd /usr/share/
[root@localhost share]# chown -R mysql:mysql mysql*
```

■ /usr/lib64/mysql,

```
[root@localhost lib64]# cd /usr/lib64
[root@localhost lib64]# chown -R mysql:mysql *
```

■ /var/lib/mysql,

```
[root@localhost lib]# cd /var/lib/
[root@localhost lib]# chown -R mysql:mysql mysql*
```

■ /home/mysql,

```
[root@localhost lib]# chmod -R 777 /home/mysql/
[root@localhost lib]#
```

- Ownership and permissions of client tools,

```
[root@localhost bin]# cd /usr/bin
[root@localhost bin]# chown mysql:mysql mysql*
```

Installing MySQL - Configuration

- Set MySQL Directories. By default, MYSQL database files are created under the `/var/lib/mysql` directory. Good practice to place database data and log files on a separate disk to binaries. MySQL configuration is recorded in the `my.cnf` file. The MySQL distribution contains several sample configurations. For the purpose of this guide a high concurrency configuration is used based on the InnoDB database engine plug-in. To set the primary MySQL directories, start with a copy a sample configuration file and modify,

```
[root@localhost bin]# cp /usr/share/mysql/my-innodb-heavy-4G.cnf /usr/bin/my.cnf
```

- Set both owner and group membership of the `my.cnf` file to `mysql`,

```
[root@localhost bin]# chown mysql:mysql /usr/bin/my.cnf
```

- Using an editor of your choosing edit the sample file (vi is used in this example),

```
[root@localhost bin]# vi /usr/bin/my.cnf
```

- Add or amend as appropriate the following entries within `my.cnf`:

Section	Action	Setting
[client]	Amend	socket = /home/mysql/bin/mysql.sock
[mysqld]	Amend	socket = /home/mysql/bin/mysql.sock
	Add	basedir = /usr
	Amend	default-storage-engine = InnoDB Centos 5 called default_table_type
	Add	datadir = /home/mysql/data
	Add	pid_file = /home/mysql/log/mysql.pid
	Add	log-error = /home/mysql/log/console.out
	Amend	log-bin = /home/mysql/bin/log
	Add	log-bin-index = /home/mysql/bin/log
	Add	innodb_log_group_home_dir = /home/mysql/bin
	Add	innodb_file_format = Barracuda
	Amend	innodb_buffer_pool_size = 512M (See Note below)
[mysql]	Add	safe-updates
[mysqld_safe]	Amend	open-files-limit = 1024

NOTE: The MySQL sample configuration, `/usr/share/mysql/my-innodb-heavy-4G.cnf`, is designed for a server with 4GB of RAM. An Economy CentOS server on FUJITSU Cloud Service S5 has 1.7GB of RAM. Either switch the server to an Advance server configuration (contains 7.5GB of RAM) or tune down the following `innodb_buffer_pool_size` configuration item from 2GB to 512MB as in the above example.

- Save the changes to `my.cnf` and copy to,

```
[root@localhost bin]# cp /usr/bin/my.cnf /etc/my.cnf
```

- Before proceeding further, ensure that you are logged in as the `mysql` user. If not type the following followed by ENTER,

```
[root@localhost bin]# su - mysql
```

19. Modify the permissions on my.cnf.

```
-bash-3.2$ chmod 664 /etc/my.cnf
-bash-3.2$
```

20. Create and initialise MySQL database tables. To create and initialise MySQL database tables. Most of the install packages will initialise the grant tables as part of the install. Therefore the following step may be optional. Based on the data directory path used in Step 7 specify the `--datadir` path.

```
-bash-3.2$ cd /usr/bin
-bash-3.2$ ./usr/bin/mysql_install_db --basedir=/usr --datadir=/home/mysql/data
Installing MySQL system tables...
OK
Filling help tables...
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:

/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h localhost.localdomain password 'new-password'
```

21. Hardening the server installation is strongly recommended using the `mysql_secure_installation` utility. This utility will provide the option of removing the test databases and anonymous user created by default. This is strongly recommended for production servers. To do this run the following command followed by ENTER,

```
-bash-3.2$ /usr/bin/mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user. If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MySQL installation has an anonymous user, allowing anyone
to log into MySQL without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.
```

```

Remove anonymous users? [Y/n] Y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
... Success!

By default, MySQL comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] Y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!

```

22. Refer to the MySQL 5.5 Reference Manual (<http://dev.mysql.com/doc/refman/5.5/en/>) for more instructions and MySQL best practice.

Installing MySQL – Starting, Testing & Stopping

23. To start the MySQL daemon using `mysqld` followed by the ENTER key,

```
-bash-3.2$ /usr/bin/mysqld_safe &
```

24. Test the MySQL daemon with `mysql-test-run.pl` followed by the ENTER key,

```

-bash-3.2$ cd /usr/mysql-test
-bash-3.2$ perl mysql-test-run.pl

Please report any problems with the /usr/bin/mysqlbug script!

-bash-3.2$

```

25. Confirm the databases have been created,

```

-bash-3.2$ ls -l /home/mysql/data/
total 12
drwx----- 2 mysql mysql 4096 Dec  1 13:58 mysql
drwx----- 2 mysql mysql 4096 Dec  1 13:58 performance_schema

```

```
drwx----- 2 mysql mysql 4096 Dec  1 13:58 test
-bash-3.2$
```

26. Confirm that each table in the mysql database has a separate data file. Also confirm that the tables are owned by the mysql user and group,

```
-bash-3.2$ ls -l /home/mysql/data/mysql/
total 1020
-rw-rw---- 1 mysql mysql 8820 Dec  1 13:58 columns_priv.frm
-rw-rw---- 1 mysql mysql  0 Dec  1 13:58 columns_priv.MYD
-rw-rw---- 1 mysql mysql 4096 Dec  1 13:58 columns_priv.MYI
-rw-rw---- 1 mysql mysql 9582 Dec  1 13:58 db.frm
...
-rw-rw---- 1 mysql mysql 1024 Dec  1 13:58 time_zone_transition_type.MYI
-rw-rw---- 1 mysql mysql 10630 Dec  1 13:58 user.frm
-rw-rw---- 1 mysql mysql  344 Dec  1 13:58 user.MYD
-rw-rw---- 1 mysql mysql 2048 Dec  1 13:58 user.MYI
-bash-3.2$
```

27. Shut down MySQL

```
[root@localhost bin]# ./mysqladmin shutdown -u root -p
Enter password:
```

28. For information on best practice MySQL installation, configuration and administration refer to the MySQL 5.5 Reference Manual (<http://dev.mysql.com/doc/refman/5.5/en/>) covering both MySQL server and client tools.

MySQL is now installed, configured and is ready to use on the CentOS server.

Contact

Fujitsu Global Cloud Team
FUJITSU
E-mail: cloud_gsd@au.fujitsu.com
Website: au.fujitsu.com

All rights reserved, including intellectual property rights.
Technical data subject to modifications and delivery subject to availability. Any liability that the data and illustrations are complete, actual or correct is excluded. Designations may be trademarks and/or copyrights of the respective manufacturer, the use of which by third parties for their own purposes may infringe the rights of such owner.

© Copyright Fujitsu Limited 2012