

Install MySQL 5.7 on CentOS 7

To install MySQL 5.7, you should enable MySQL 5.7 community release yum repository on your system.

STEP 1. Enable MySQL Repository

```
sudo yum -y localinstall https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

STEP 2. Install MySQL 5.7 Server

```
sudo yum -y install mysql-community-server
```

STEP 3. Start MySQL Service

```
sudo service mysqld start
```

STEP 4. Checking temporary password

```
sudo grep 'A temporary password' /var/log/mysqld.log | tail -1
```

Then you will be able to see its result like below:

```
2020-04-20T01:50:07.931468Z 1 [Note] A temporary password is generated for root@localhost: #l+wu>PMv7vJ
```

Change password by the pre-defined password above

```
mysql_secure_installation
```

Adding a mysql user account

```
mysql -uroot -p<password>

#uninstall plugin to validate_password (not recommended)
uninstall plugin validate_password;

# below script is to create user account for local connection
CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'user_password';

# below script is to create user account can connect from 1.2.3.4
CREATE USER 'newuser'@'1.2.3.4' IDENTIFIED BY 'user_password';

# below script is to create user account for any connection
CREATE USER 'newuser'@'%' IDENTIFIED BY 'user_password';
```

Granting user permission

```
mysql -uroot -p<password>

# grant privileges on a certain database
GRANT ALL PRIVILEGES ON database_name.* TO 'database_user'@'localhost';

# grant privileges on all the database in the system
GRANT ALL PRIVILEGES ON *.* TO 'database_user'@'localhost';

# grant full privileges on all the database in the system
GRANT ALL PRIVILEGES ON *.* TO 'database_user'@'localhost' REQUIRE NONE WITH GRANT OPTION MAX_QUERIES_PER_HOUR
0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0;
```

STEP 5. Initial MySQL Configuration

```
/usr/bin/mysql_secure_installation
```

Note you can use the temporary password acquired at STEP 4.

STEP 6. /etc/my.cnf

Below is what I use in my server (i7 / RAM 32G) to run multiple server instances

```

# For advice on how to change settings please see
# http://dev.mysql.com/doc/refman/5.7/en/server-configuration-defaults.html

[mysqld]
#
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M
#
# Remove leading # to turn on a very important data integrity option: logging
# changes to the binary log between backups.
# log_bin
#
# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock

# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

# added by Chun Kang - 2020-04-20
character-set-server = utf8mb4
collation-server = utf8mb4_bin

# wait_timeout=28800 #28.800
wait_timeout=1800 #1.800

# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

# Recommended in standard MySQL setup
sql_mode=NO_ENGINE_SUBSTITUTION,STRICT_TRANS_TABLES

# Added to support Confluence
default-storage-engine=INNODB
transaction-isolation=READ-COMMITTED
max_allowed_packet=256M
open_files_limit= 100000
max_connections = 100000
table_open_cache= 100000
binlog_format=row

explicit_defaults_for_timestamp = TRUE

innodb_buffer_pool_size = 12GB          # recommended to use 60-80% of whole RAM
innodb_log_file_size = 2GB             # recommended by confluence, 20% of innodb_buffer_pool_size
innodb_file_per_table = ON
innodb_large_prefix=ON
innodb_file_format=Barracuda

```