

Version	Author : modifications	Date
1.00	ASI: Author	30/03/2016

## Installation MySQL Server 5.7 on CentOS 7

`sudo wget http://dev.mysql.com/get/mysql57-community-release-el7-7.noarch.rpm`

`sudo yum localinstall -y mysql57-community-release-el7-7.noarch.rpm`

`sudo yum install -y mysql-server`

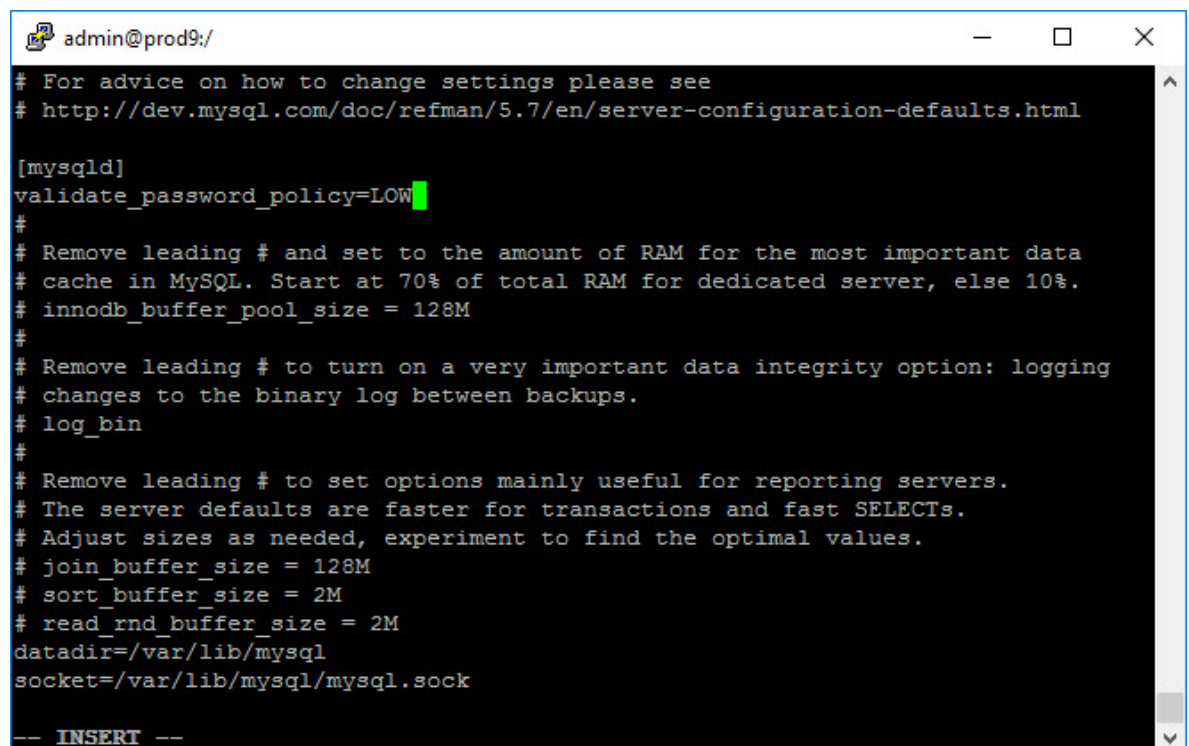
`sudo service mysqld start`

Changing password rule in file my.conf

`sudo vi /etc/my.conf`

[mysqld]

validate\_password\_policy=LOW



```
admin@prod9:/  
# For advice on how to change settings please see  
# http://dev.mysql.com/doc/refman/5.7/en/server-configuration-defaults.html  
  
[mysqld]  
validate_password_policy=LOW  
#  
# 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  
  
-- INSERT --
```

### Find MySQL temporary password for first login

```
sudo grep 'temporary password' /var/log/mysqld.log
```

### Some security configuration on MySQL server

```
sudo mysql_secure_installation
```

Enter temporary password for user root

Answer to some questions

- |   |                            |
|---|----------------------------|
| 1. Change the password for root?          | press yes and set password |
| 2. Remove anonymous users?                | press yes                  |
| 3. Disallow root login remotely?          | press yes                  |
| 4. Remove test database and access to it? | press yes                  |
| 5. Reload privilege tables now?           | press yes                  |

### Login by root in MySQL and create a user with special privileges for remote administration

```
mysql -u root -p
```

```
Create user "damaris"@"%" identified by "yourpassword";
```

```
grant all privileges on *.* to "damaris"@"%" with grant option;
```

```
exit
```

### Configuration of the remote connection to MySQL server by SSH

```
sudo vi /etc/ssh/sshd_config
```

```
Change PermitTunnel to yes
```

```
#UseDNS yes
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
PermitTunnel yes
#ChrootDirectory none
#VersionAddendum none
```

```
sudo service sshd restart
```

### Connect to server by MySQL Workbench using Connection Method:

**Standard TCP/IP over SSH**

Setup New Connection

Connection Name:  Type a name for the connection

Connection Method:  Method to use to connect to the RDBMS

Parameters SSL Advanced

SSH Hostname:  SSH server hostname, with optional port number.

SSH Username:  Name of the SSH user to connect with.

SSH Password:   SSH user password to connect to the SSH tunnel.

SSH Key File:   Path to SSH private key file.

MySQL Hostname:  MySQL server host relative to the SSH server.

MySQL Server Port:  TCP/IP port of the MySQL server.

Username:  Name of the user to connect with.

Password:   The MySQL user's password. Will be requested later if not set.

Default Schema:  The schema to use as default schema. Leave blank to select it later.

## MySQL database backup configuration

### Creating directory for backups

```
sudo mkdir /opt/mysqlbackup
```

```
sudo chmod 777 /opt/mysqlbackup
```

### Insert configurations below in file "backup.sh"

```
cd /opt/scripts/
```

```
sudo vi backup.sh
```

```
#!/bin/bash
# HOSTNAME where be backup
HOSTNAME=localhost
# Database user name
DBUSER=damaris
# directory where going backup files
DIRECTORY=/opt/mysqlbackup
# database names which be backup
DBNAME=databasename DBNAME1=databasename
# database user password
DBPASS=yourpassword
# date of backup
```

```
DATE=`date +%d.%m.%y`  
# command which doing first database backup  
mysqldump -h $HOSTNAME -u $DBUSER -p$DBPASS $DBNAME > $DIRECTORY/  
databasename -$DATE-dump.sql  
# command which doing second database backup  
mysqldump -h $HOSTNAME -u $DBUSER -p$DBPASS $DBNAME1 > $DIRECTORY/  
databasename -$DATE-dump.sql
```

### Adding backup script in crontab for starting by schedule.

Go to admin's home directory, create a file and insert the below information

```
cd /home/admin  
sudo vi file
```

```
SHELL=/bin/bash  
MAILTO=admin  
30 1 * * * /opt/scripts/backup.sh
```

Schedule will start every day 1:30 AM

Set permission for **cron** and add file in **crontab**

```
sudo chmod 755 /var/spool/cron  
crontab /home/admin/file
```

### Transfer backups to backup server by FTP

#### Install lftp

```
sudo yum install lftp  
sudo vi /etc/lftp.conf
```

Insert the bellow information to connect without displaying certificate part.

```
set ssl:verify-certificate no
```

### Create transfer script in directory /opt/scripts/

```
cd /opt/scripts/  
sudo vi ftp.sh
```

insert configuration with yours credentials

```
#!/bin/bash  
DATE=`date +%d.%m.%y`  
lftp <<SCRIPT  
lftp user:password@ftp_server_ip  
cd /data/your_path  
mkdir -p dump_ $DATE  
cd dump_ $DATE  
mput /backup/*  
exit  
SCRIPT
```

### Add transfer script in crontab for starting by schedule.

```
crontab -e
```

```
30 5 * * * /opt/scripts/ftp.sh
```

Schedule will start every day 5:30 AM