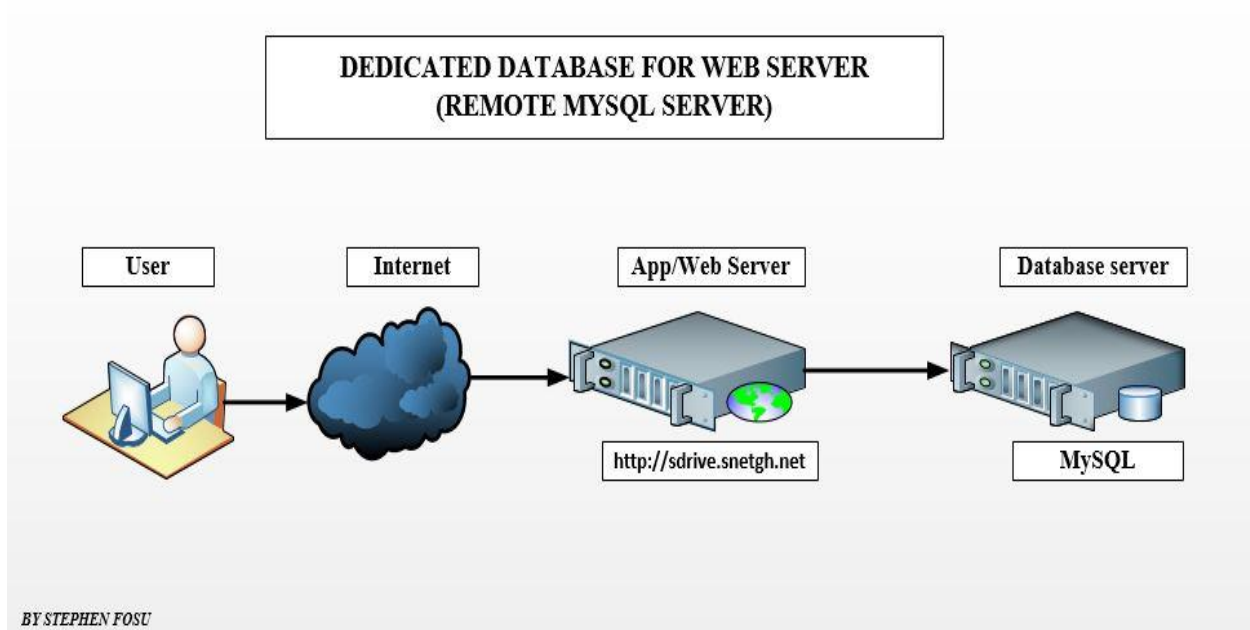


SETTING UP REMOTE/DEDICATED MYSQL SERVER FOR APPLICATION/WEB SERVER



INTRODUCTION

As your application or website grows, you may come to a point where you've outgrown your current server setup. With servers hosting large websites, resource usage is a big issue and in most cases MySQL is consuming the majority of the resources. To avoid this, setting up a separate standalone MySQL server and loading the websites from there is a very good option. MySQL server will be installed in a remote machine/server and websites loading from the local web server are tied to the databases in the remote MySQL server.

INSTALLING MYSQL SERVER ON THE UBUNTU

Step 1) First we need to install MySQL in a remote server.

Using the console command line. In your terminal enter

```
root@snet[~]# sudo apt-get install mysql-server
```

Or

```
root@snet[~]# sudo apt install -y mysql-server mysql-client
```

CONFIGURE MYSQL SERVER TO LISTEN TO ANY IP

Step 2) Make the MySQL server listen to any IP or your private or public IP, so that the web server will be able to reach the MySQL server.

Use the following command:

```
root@snet[~]# nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

Search the configuration file for ***bind-address*** and change

bind-address = 127.0.0.1 to bind-address = 0.0.0.0

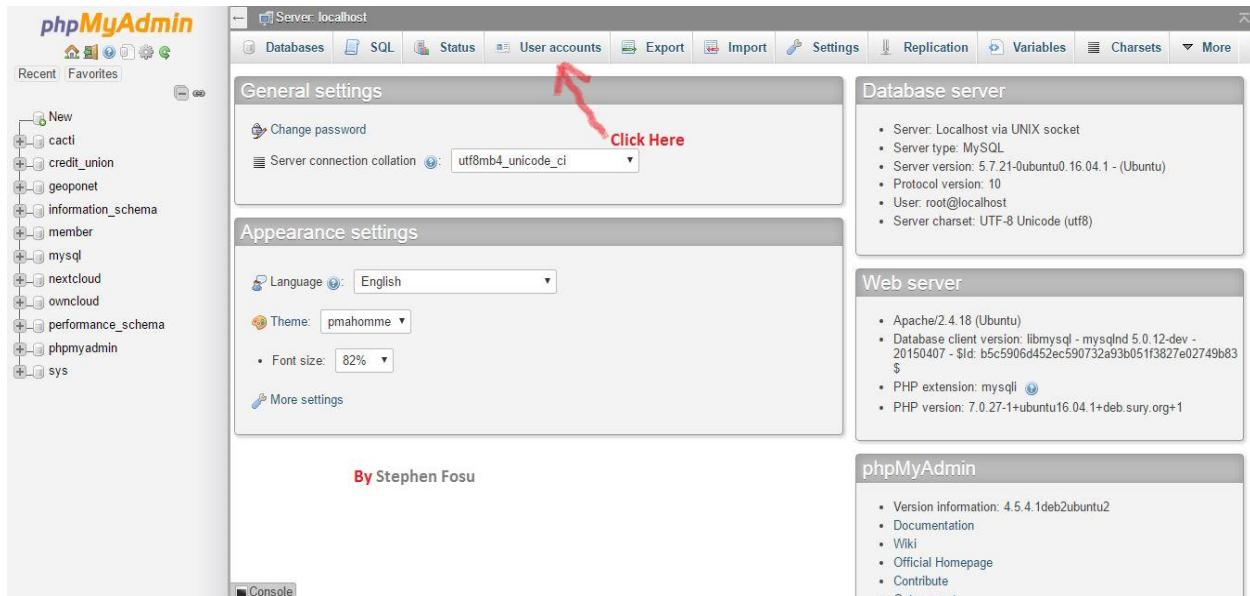
```
root@snet[~]# sudo systemctl restart mysql
```

CREATING OF MYSQL USER, DATABASE AND GRANTING PRIVILEGES

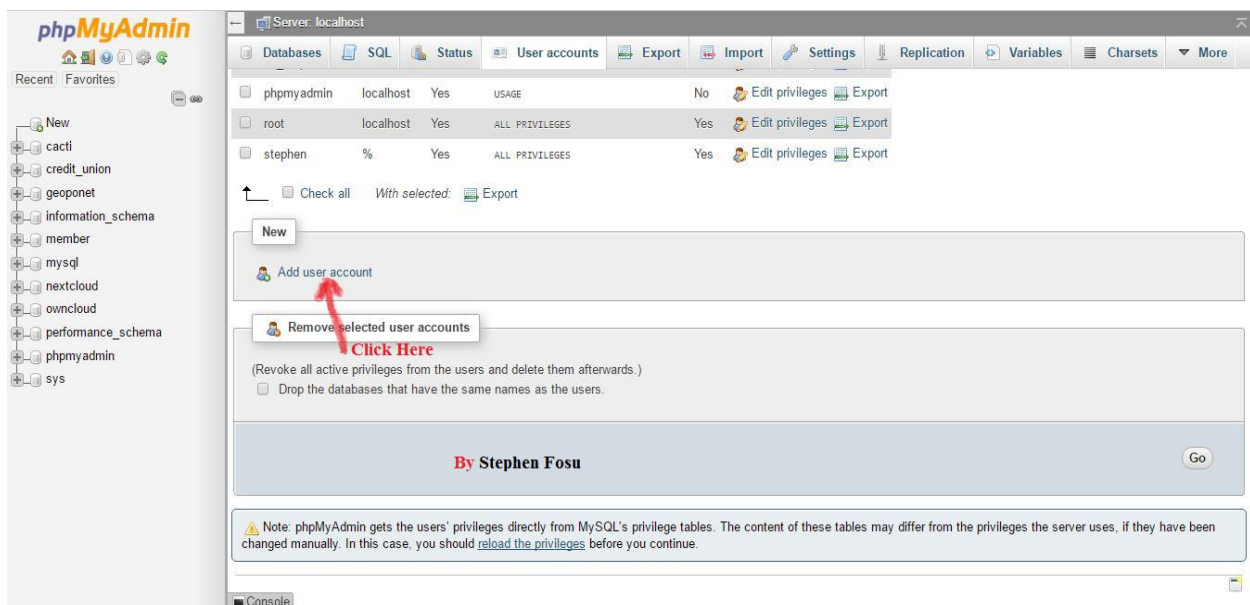
Step 3) Create the MySQL user and database and grant privileges to access this from the web server.

USING PHPMYADMIN OR ANY GRAPHICAL USER INTERFACE (WORKBENCH, ADMINER)

Login into your phpmyadmin and click on “**user accounts**”



Click on the “**Add user account**” to add new user



Add username, host name (leave it on default (any host)) and password

The screenshot shows the 'Add user account' form in phpMyAdmin. The 'Login Information' tab is active. The 'User name' field contains 'stephen', with a red arrow pointing to it and the text 'Enter the username'. The 'Host name' dropdown is set to 'Any host', with a red arrow pointing to it and the text 'Leave it on default for all hosts'. The 'Password' and 'Re-type' fields contain masked text, with a red arrow pointing to the 'Re-type' field and the text 'User Password'. A warning message states: 'An account already exists with the same username but possibly a different hostname.' The 'Authentication Plugin' is set to 'Native MySQL authentication'. The 'Database for user account' section is visible at the bottom.

Grant the user all privileges by clicking “Database for user account and Global privileges”. Then click for “Go” to finish.

The screenshot shows the 'Database for user account' and 'Global privileges' sections in phpMyAdmin. In the 'Database for user account' section, the checkbox 'Create database with same name and grant all privileges.' is checked, with a red arrow pointing to it and the text 'Click to create database with same username'. The checkbox 'Grant all privileges on wildcard name (username_%)' is also checked, with a red arrow pointing to it and the text 'Click to grant all privileges'. In the 'Global privileges' section, the 'Check all' checkbox is checked, with a red arrow pointing to it and the text 'Click to give global privileges'. Below this, there are four panels: 'Data' (SELECT, INSERT, UPDATE, DELETE, FILE), 'Structure' (CREATE, ALTER, INDEX, DROP, CREATE TEMPORARY TABLES, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EXECUTE, CREATE VIEW, EVENT, TRIGGER), 'Administration' (GRANT, SUPER, PROCESS, RELOAD, SHUTDOWN, SHOW DATABASES, LOCK TABLES, REFERENCES, REPLICATION CLIENT, REPLICATION SLAVE, CREATE USER), and 'Resource limits' (MAX QUERIES PER HOUR, MAX UPDATES PER HOUR, MAX CONNECTIONS PER HOUR, MAX USER_CONNECTIONS). The 'By Stephen Fosu' signature is at the bottom.

USING THE CL (CONSOLE COMMAND LINE)

```
root@snet[~]# mysql -u root -p
```

Enter MySQL password

Create a MySQL database for snetgh application/website present in the web server.

```
root@snet[~]# CREATE DATABASE snetgh;
```

Create MySQL user who can access this database from the web server and assign a password for it.

```
root@snet[~]# CREATE USER 'sdrive'@'%' IDENTIFIED BY 'password';
```

OR

```
root@snet[~]# CREATE USER 'sdrive'@'172.168.1.90' IDENTIFIED BY 'password';
```

Grant all privileges to the MySQL user to access this database.

```
root@snet[~]# GRANT ALL PRIVILEGES ON snetgh.* TO 'sdrive'@'%' ;
```

OR

```
root@snet[~]# GRANT ALL PRIVILEGES ON snetgh.* TO 'sdrive'@'172.168.1.90';
```

```
root@snet[~]# FLUSH PRIVILEGES;
```

```
root@snet[~]# exit
```

Note: % means any host or 172.168.1.90 is the webserver's IP. Then restart MySQL Server

```
root@snet[~]# sudo systemctl restart mysql
```

VERIFY DATABASE ACCESS FROM THE WEBSERVER

Step 4) On the web server, enter the following command to verify the connection works:

```
root@snet[~]# mysql -u sdrive -h 172.168.1.80 -p
```

Enter the MySQL user password

If the MySQL monitor displays as follows, the database is ready

```
root@webserver:~# mysql -u sdrive -h 172.168.1.80 -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 500
Server version: 5.7.20-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

The 172.168.1.80 is the MySQL server's IP

BY STEPHEN FOSU