**MySQL and Apache Web Server**

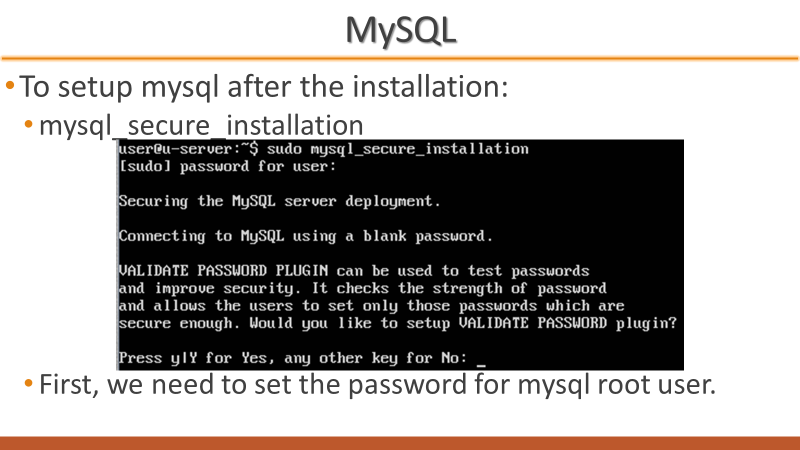
**MySQL**

Sudo mysql\_secure\_installation

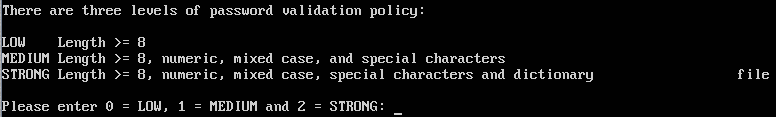
1. Run the post-installation setup to perform the following tasks.

* Set password for **root**, using the Medium validation policy.
* Remove anonymous users.
* Disallow root login remotely.
* Remove **test** database and reload privilege tables.

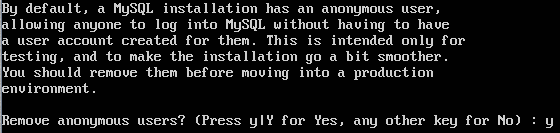
Command to run post-installation:



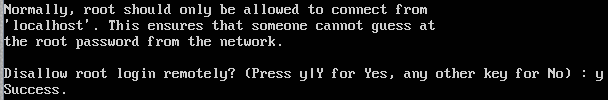
Setup password



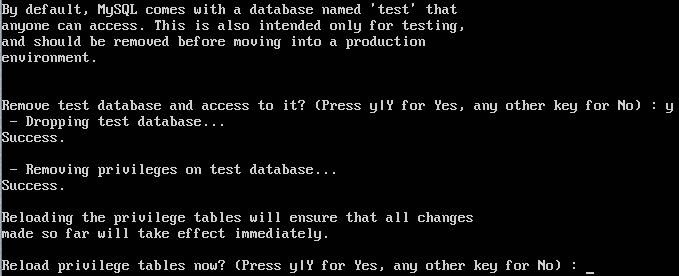
Choose 1 for medium



Remove anonoymous



Disable remote root login



Remove test database

1. Login to mysql as root.

Sudo mysql

*Self-check:*

* *Check the* ***user*** *table in* ***mysql*** *database to make sure that the anonymous users have been removed. The table should contain only root and reserved users.*

*Command: select user, host from mysql.user;*

* *Check that the* ***test*** *database has been removed.*

*Command: show databases;*

1. Create a new database named **stdb**.

Command: CREATE DATABASE stdb;

1. Create a table named **tbl\_items** with the following fields:

* id, integer type with length equal to 10, auto increment, primary key
* name, varchar with length = 30
* brand, varchar with length = 20

Command:

CREATE TABLE tbl\_items (item\_id varchar(10), name varchar(30), brand carchar(20));

1. Create a table named **tbl\_supp** with the following fields:

* id, integer type with length = 10, auto increment, as primary key
* name, varchar with length = 30
* biz\_type, varchar with length = 30

Command: CREATE TABLE tbl\_supp (item\_id varchar(10), name varchar(30), biz\_type carchar(20));

*Self-check:*

* *Check the structure of both tables to verify that they are created correctly.*

*Command: DESCRIBE tbl\_items; OR DESCRIVE tbl\_supp;*

* *How to add a field / column into existing table?*

*Command: ALTER TABLE table\_name ADD new\_field\_name type;*

* *How to remove a field / column from a table?*

*Command: ALTER TABLE table\_name DROP field\_name;*

* *How to remove a table from a database?*

*Command: DROP TABLE table\_name;*

1. Add a field / column named **supp\_id** into **tbl\_items** as a foreign key.

Command: ALTER TABLE tbl\_items ADD supp\_id, ADD foreign key (supp\_id);

1. Create a new user named **stadmin**, which can login from localhost and also from any host, with default password **stAdmin\_608**.

Command: CREATE USER ‘stadmin’@’localhost’ IDENTIFIED BY ‘stAdmin\_608’;

*Self-check:*

* *What if the password doesn’t meet the verification policy? How to change password?*

*Command:*

1. If **stadmin** logs in from localhost, all privileges will be granted on the **stdb** database.

GRANT all privileges ON stdb. : TO ‘stadmin’@’localhost’;

1. If **stadmin** logs in from any other host, only CREATE, ALTER, DELETE, DROP, SELECT, INSERT and UPDATE privileges are granted on **stdb** database.

*Self-check:*

* *How to remove privileges from user?*

*Command:* REVOKE all privileges, grant option FROM ‘user\_name’@’host’;

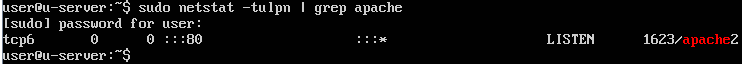
1. Create a Cron job to backup **stdb** database with **mysqldump**. (Schedule the job 2 minutes from current time to see whether it works. Remember to check the output file.

**Apache**

1. Check if **apache2** is running and which port it is listening to.

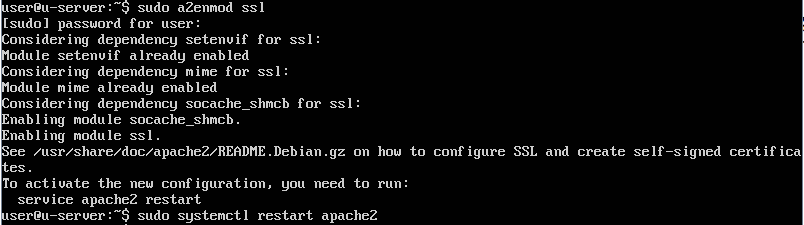
Command: systemctl status apache2

sudo netstat –tulpn | grep apache //check port



* Each virtual host consist of a configuration file, differentiate based on name or IP address.
  + The default configuration file located at /etc/apache2/sites-available, named **000-default.conf**
  + Can be duplicated and modified for different virtual hosts.

1. Enable the SSL module.
   * Command: sudo a2enmod ssl
   * sudo systemctl restart apache2



1. Create the directory **/etc/apache2/ssl-certs**.

Command: sudo mkdir /etc/apache2/ssl-certs

1. Generate self-signed certificate and key, both must be stored in **ssl-certs** directory.

Command:

Openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/certs/mysite.key – out /etc/apache2/certs/mysite.crt

1. Copy the default configuration file for virtual host with SSL and name it as **scm-ssl.conf**.

Each virtual host consist of a configuration file, differentiate based on name or IP address.

* 1. The default configuration file located at /etc/apache2/sites-available, named **000-default.conf**

Can be duplicated and modified for different virtual hosts

* Content of configuration file

<VirtualHost \*:80>

ServerAdmin webmaster@localhost

DocumentRoot /var/www/html

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

Command: sudo a2ensite scm-ssl.conf

1. Create a virtual host based on the following information and modify the configuration file accordingly.

* ServerName: www.scm.org
* ServerAdmin: webmaster@scm.org
* DocumentRoot: /var/www/scm

You need to create the necessary directory and change the ownership.

Commands:

Sudo mkdir -p /var/www/scm

Sudo chown -R $USER:$USER /var/ww/scm

Cd /etc/apache2/sites-available

Sudo cp 000-default.conf scm.conf

Nano scm.conf

ServerName www.scm.org

ServerAdmin webmaster@scm.org

DocumentRoot /var/www/scm

ErrorLog ${APACHE\_LOG\_DIR}/scm\_error.log

CustomLog ${APACHE\_LOG\_DIR}/scm\_access.log combined

1. Use nano to create index.html in /var/www/scm. Sample content of index.html:

<html>

<head>

<title> www.scm.org </title>

</head>

<body>

<h1> Welcome to scm.org! </h1>

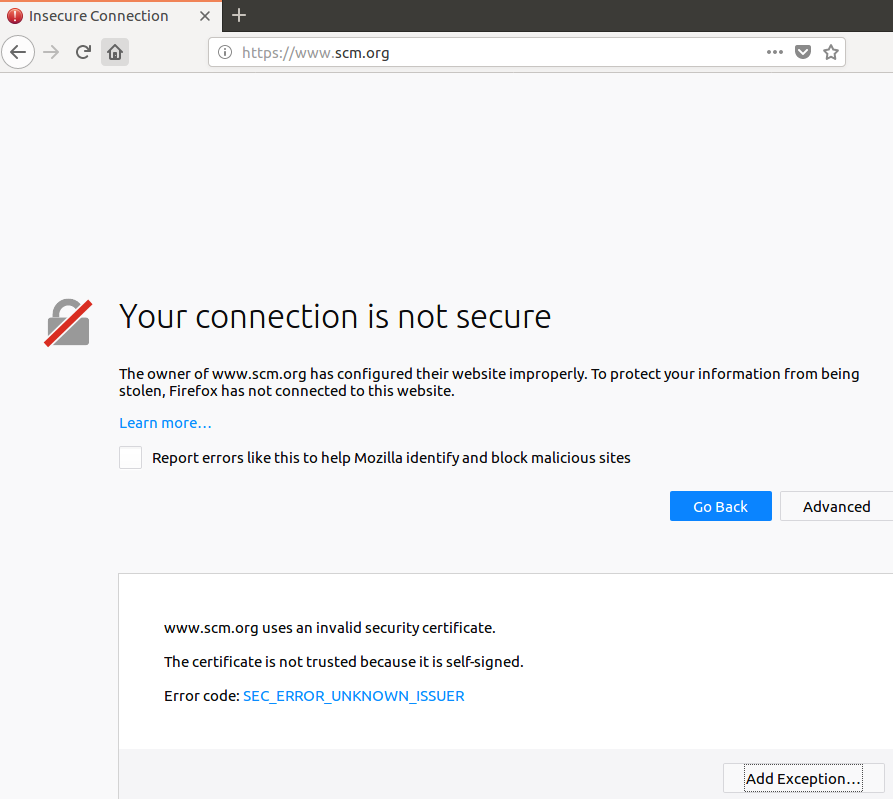
</body>

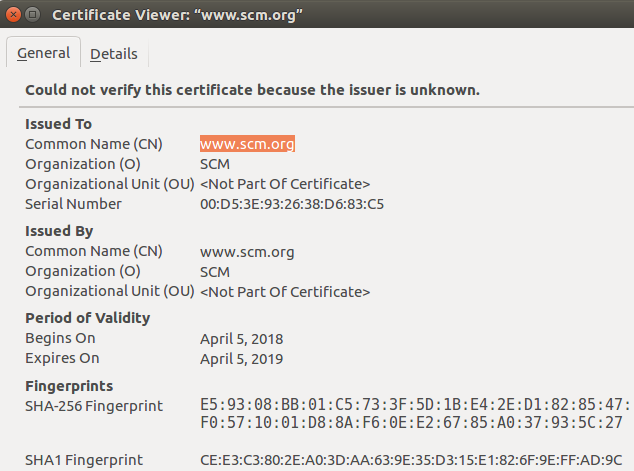
</html>

Nano index.html /var/www/scm

1. Enable the site for scm.org.
   * + Command: sudo a2ensite scm.conf
2. Reload apache service.

Command: sudo systemctl reload apache2

1. Open a browser and visit www.scm.org in Ubuntu Dektop.
2. 



*Self-check:*

* *Create another virtual host without SSL for the domain “scm.net”.*