**Name: Sandra pm**

**Roll No: 34**

**Batch: B**

**Date: 02-06-2022**

**NETWORKING & SYSTEM ADMINISTRATION LAB**

**Aim**

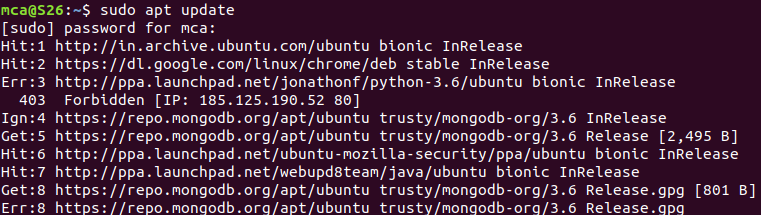
Install Linux, Apache, MySQL, PHP (LAMP) stack on Ubuntu 18.04.

**Procedure**

**Installing Apache and Updating the Firewall .**

**Step 1:** First, make sure your apt cache is updated with:

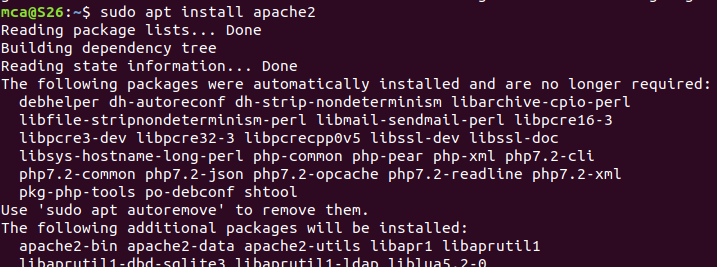
**Command : sudo apt update**

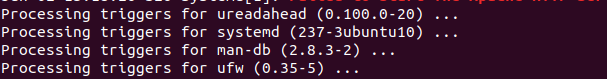


**Step 2 :** Once the cache has been updated, you can install Apache with:

**Command : Sudo apt install apache2**

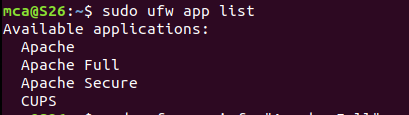
After entering this command, apt will tell you which packages it plans to install and how much extra disk space they’ll take up. Press Y and hit ENTER to confirm, and the installation will proceed.





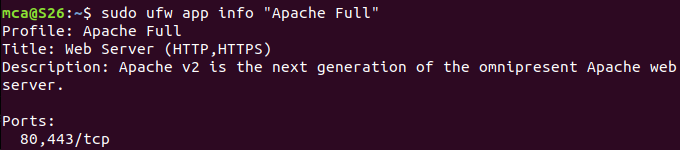
**Step 3 :** Check that UFW has an application profile for Apache:

**Command** : sudo ufw app list



**Step 4 :** If you look at the Apache Full profile details, you’ll see that it enables traffic to ports 80 and 443:

**Command : sudo ufw app info "Apache Full"**

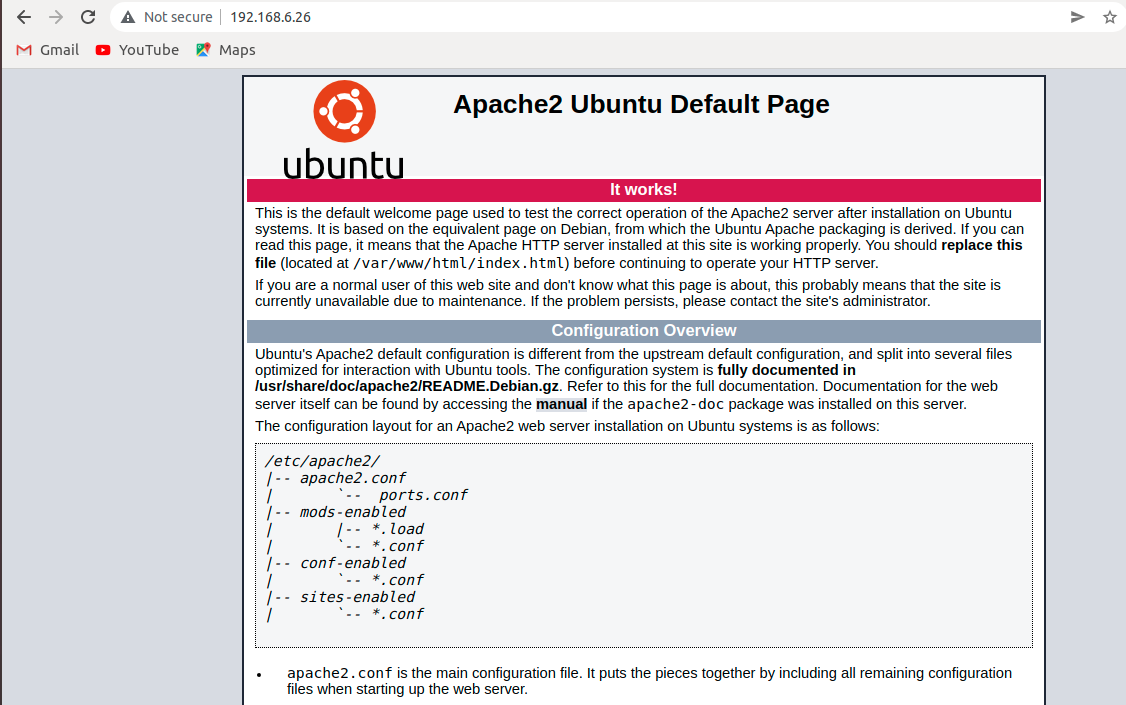
****

**Step 5 :** To allow incoming HTTP and HTTPS traffic for this server, run:

**Command : sudo ufw allow "Apache Full"**

****

**Step 6 :** You can do a spot check right away to verify that everything went as planned by visiting your server’s public IP address in your web browser:

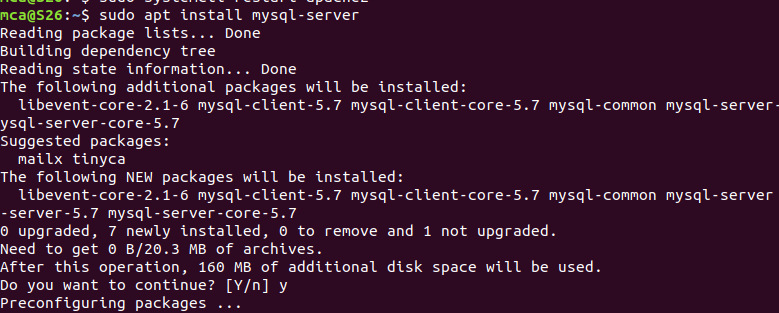
****

**Installing MySQL**

**Step 1:** Use apt to acquire and install this software.

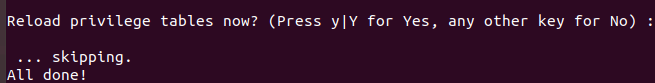
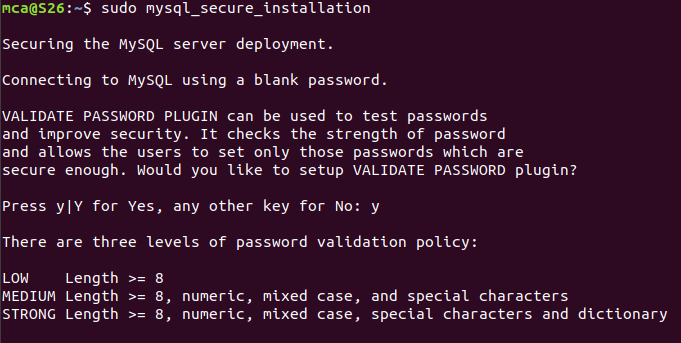
**Command : sudo apt install mysql-server**

This command, too, will show you a list of the packages that will be installed, along with the amount of disk space they’ll take up. Enter Y to continue.

****

**Step 2 :** Start the interactive script by running:

**Command : sudo mysql\_secure\_installation**

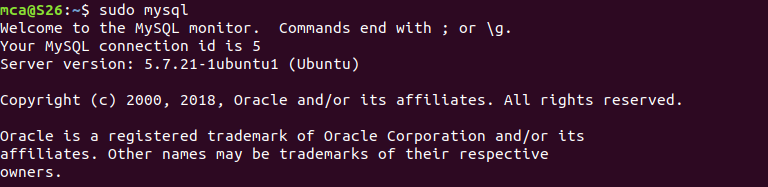
This will ask if you want to configure the VALIDATE PASSWORD PLUGIN. Answer Y for yes, or anything else to continue without enabling.

**Step 3 :** Test if you’re able to log in to the MySQL console by typing:

**Command : sudo mysql**

This will connect to the MySQL server as the administrative database user root, which is inferred by the use of sudo when running this command.

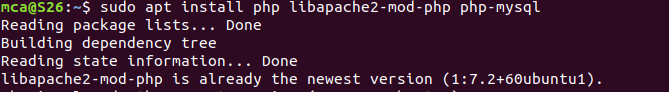
To exit the MySQL console, type: **exit**

****

**Installing PHP**

**Step 1 :** In addition to the php package, you’ll also need libapache2-mod-php to integrate PHP into Apache, and the php-mysql package to allow PHP to connect to MySQL databases. Run the following command to install all three packages and their dependencies:

**Command :** sudo apt install php libapache2-mod-php php-mysql



**Step 2 :** Restart the Apache web server in order for your changes to be recognized.

**Command :** sudo systemctl restart apache2

****

**Step 3 :** In order to test that your system is properly configured for PHP, create a PHP script called info.php. In order for Apache to find this file and serve it correctly, it must be saved to your web root directory.

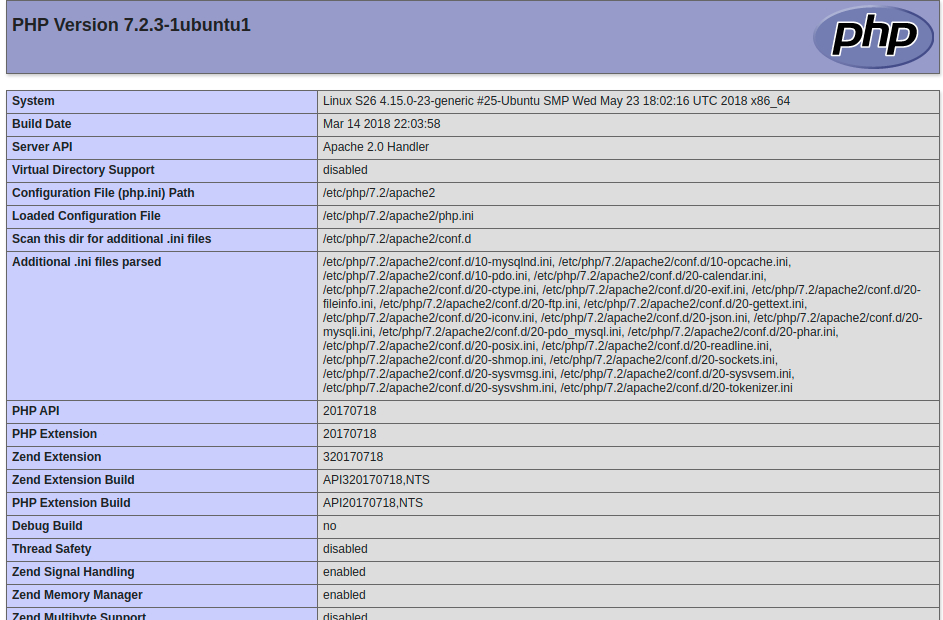
**Command : sudo nano /var/www/your\_domain/info.php**

This will open a blank file. Add the text, which is valid PHP code, inside the file. When you are finished, save and close the file.

****

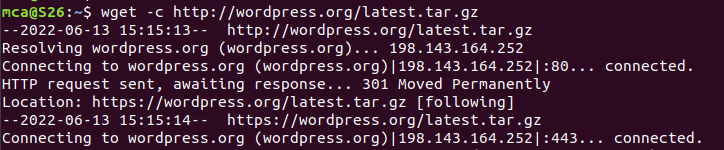
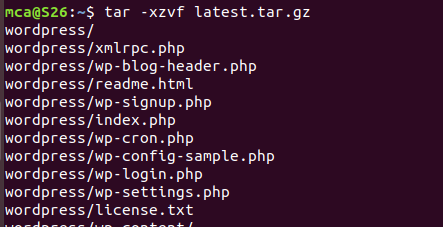
**Step 4 :** Test whether your web server is able to correctly display content generated by this PHP script. To try this out, visit this page in your web browser. You’ll need your server’s public IP address or domain name again.

**Command :** [**http://your\_domain/info.php**](http://your_domain/info.php)

****

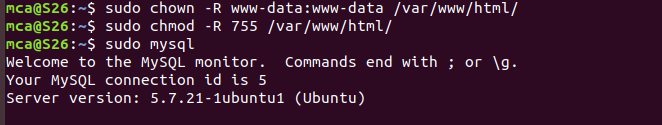
**Install WordPress with LAMP on Ubuntu**

**Step 1 – Download WordPress**

Download the latest version of the WordPress package and extract it

Then move the WordPress files from the extracted folder to the Apache default root directory, /var/www/html/:



Next, set the correct permissions on the website directory, that is give wnership of the WordPress files to the webserver

**Step 2 – Creating a MySQL Database and User for WordPress**

The first step to create a database to manage and store the user information for WordPress to use. To get started, log into the MySQL root (administrative) account

Create the database for WordPress by writing the following:

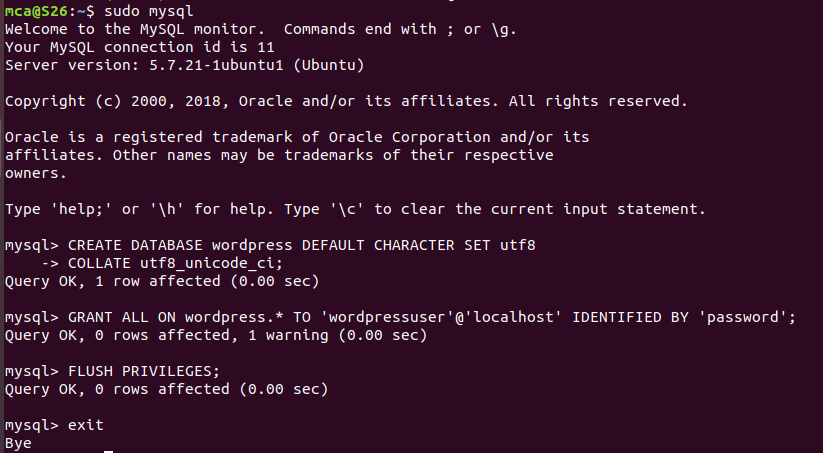
CREATE DATABASE wordpress DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;

create the account, set a password for it, and then grant it access to the database:

GRANT ALL ON wordpress.\* TO 'wordpressuser'@'localhost' IDENTIFIED BY 'password';

After creating this user, flush the privileges to ensure that the current instance of MySQL knows about: the recent changes

FLUSH PRIVILEGES;

EXIT;

Go the /var/www/html/ directory and rename existing wp-config-sample.php to wpconfig.php. Also, make sure to remove the default Apache index page:

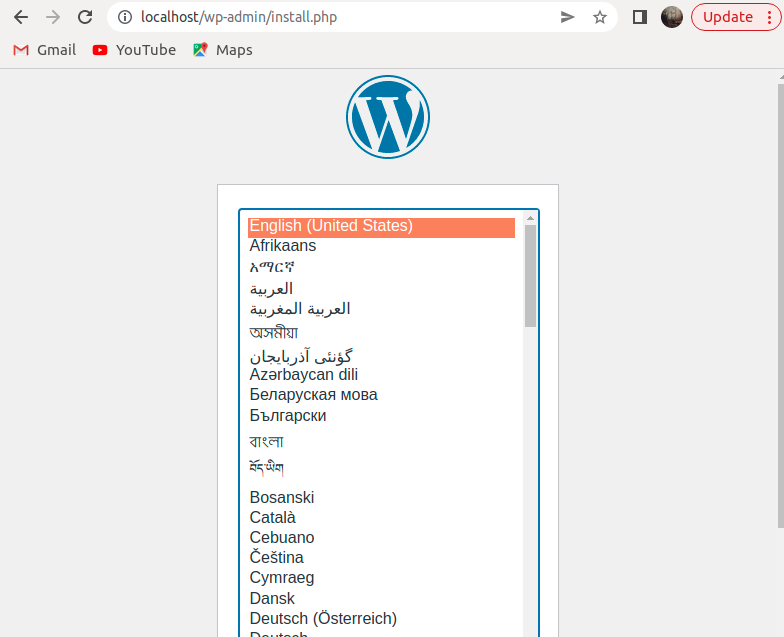


Restart the web server and mysql service

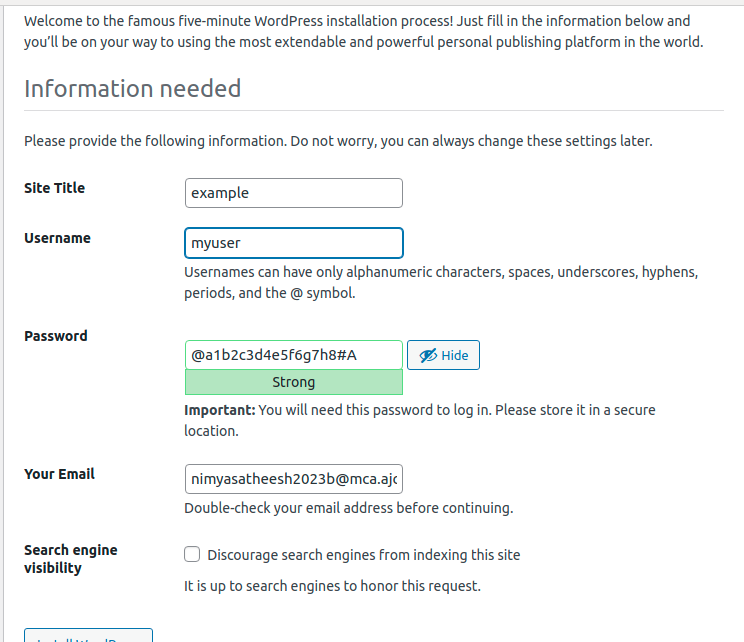


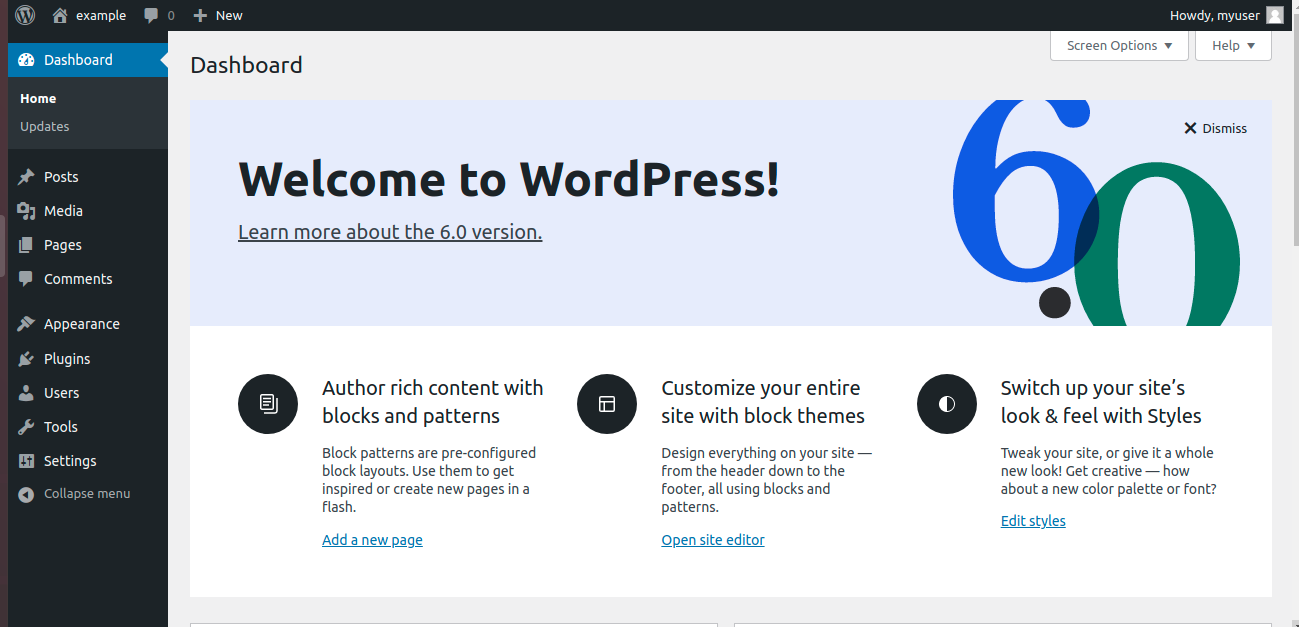
**Step 3 – Completing the Installation Through the Web Interface**

[https://server\_domain\_or\_IP](https://server_domain_or_IP/)



Select a name for your WordPress site and choose a username. A strong password is generated automatically. Save this password or select an alternative strong password.



WordPress administration dashboard: