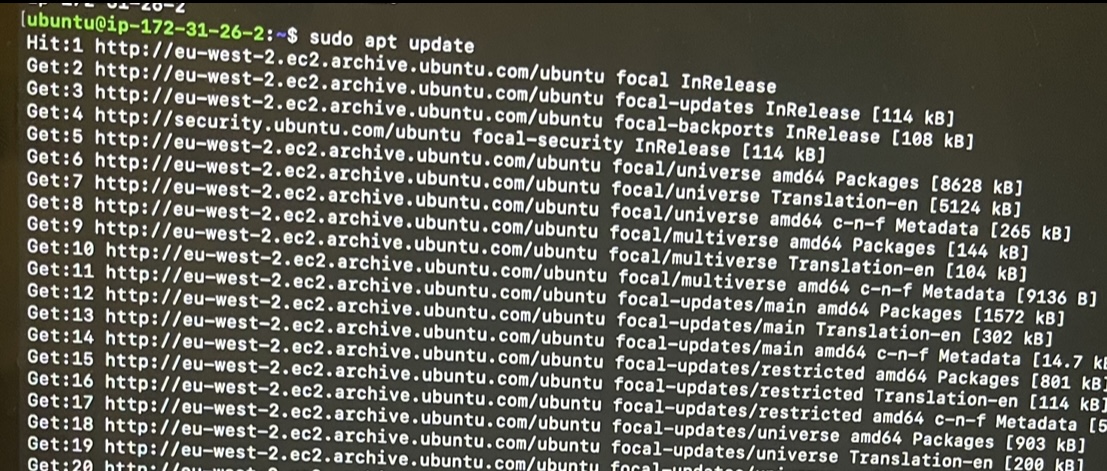
**WEB STACK IMPLEMENTATION (LAMP STACK) IN AWS**

## step 1 - Installing Apache and updating the firewall

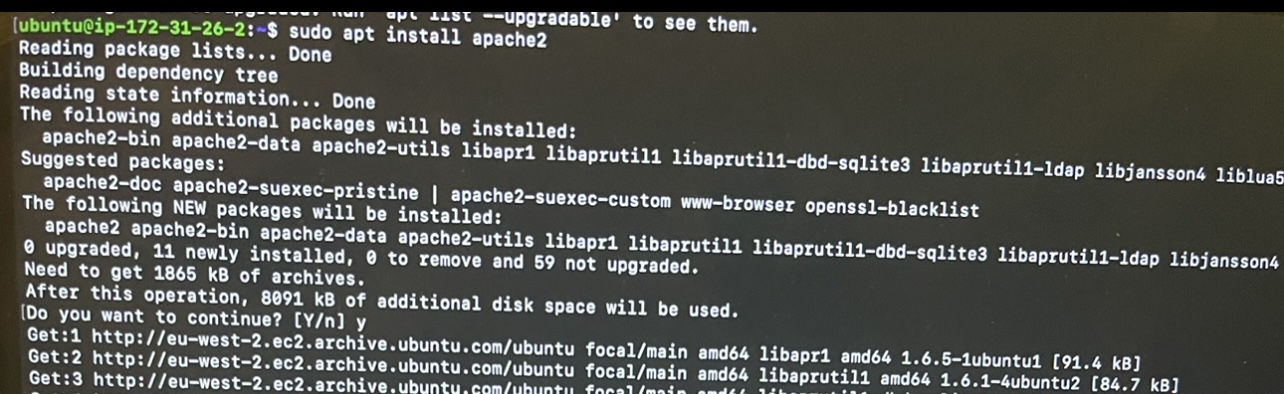
The fisrt step involves updating a list of packages in the package manager.

Command: $ sudo apt update

[](https://user-images.githubusercontent.com/93732510/153726394-dbc1ad7b-33cb-43da-bb34-bdf6f5be45b9.jpg)

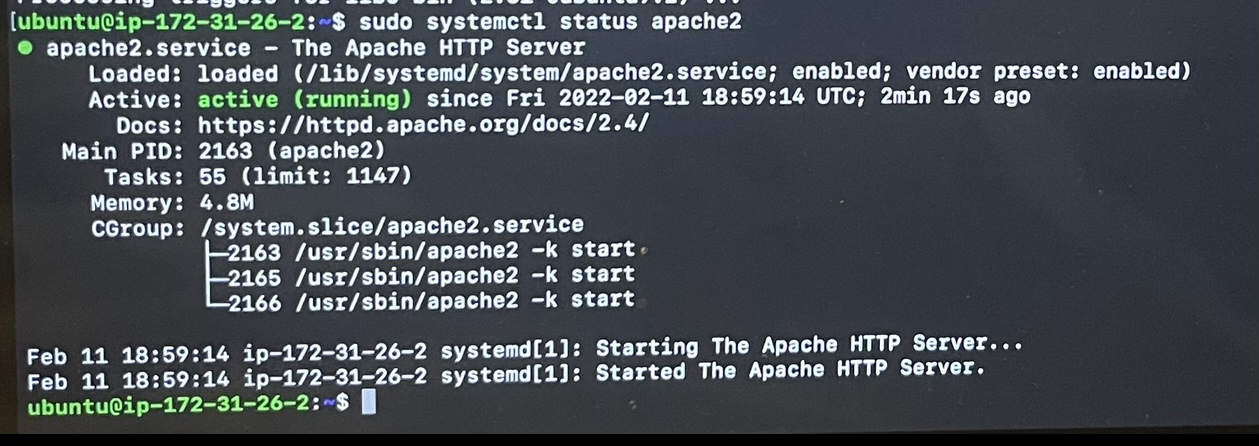
The below command runs the installation of apache2

Command: $ sudo apt install apache

[](https://user-images.githubusercontent.com/93732510/153728784-8e51dda7-e010-4003-b599-25cce299d76c.jpg)

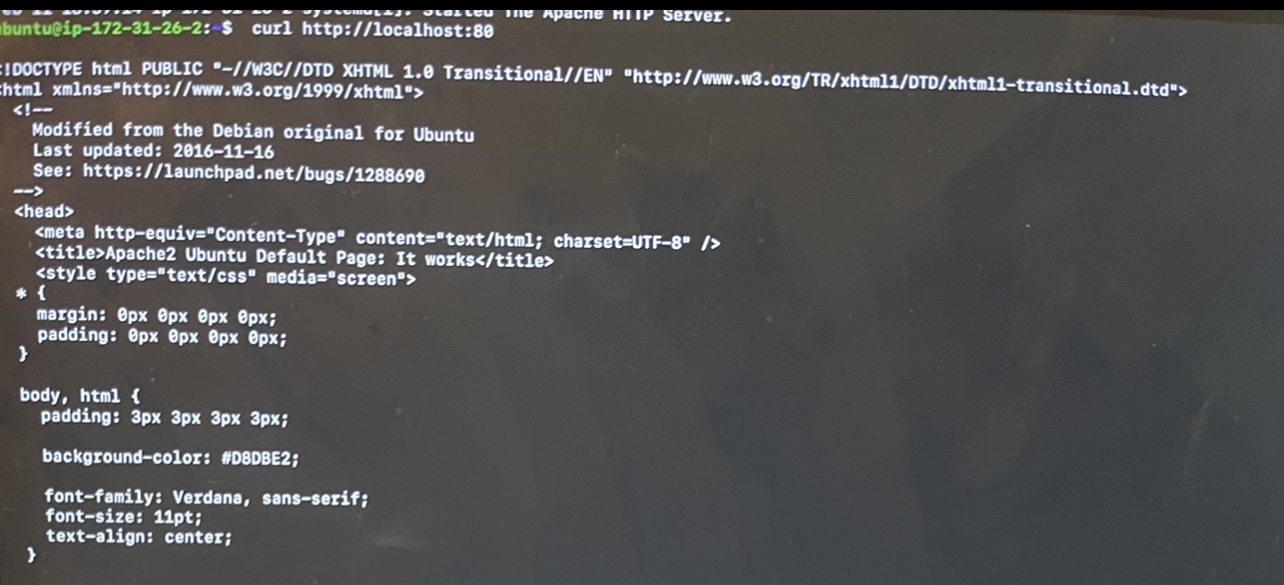
Next is to verify that the apache2 is running in the guest OS using below command

Command: $ sudo systemctl status apache2

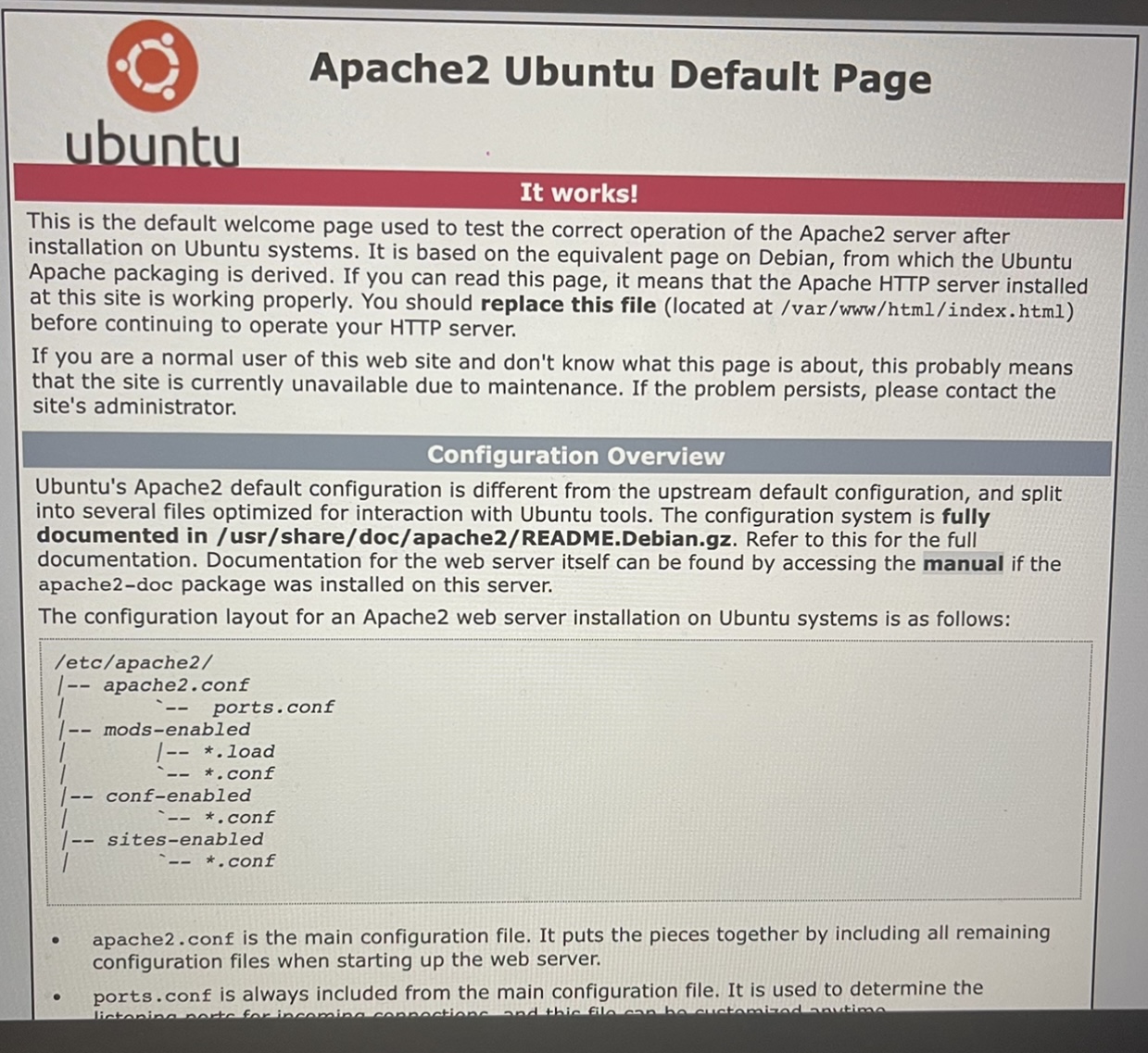
[](https://user-images.githubusercontent.com/93732510/153728943-888319c4-5048-4566-af21-d5960f25a7c2.jpg)

Now that the apache is running and active, there is need to try and access it via local host and via public IP.

Local Access - Command: curl [http://localhost:80](http://localhost/)

[](https://user-images.githubusercontent.com/93732510/153730058-44f55264-420b-4b89-919b-f6c8279ff571.jpg)

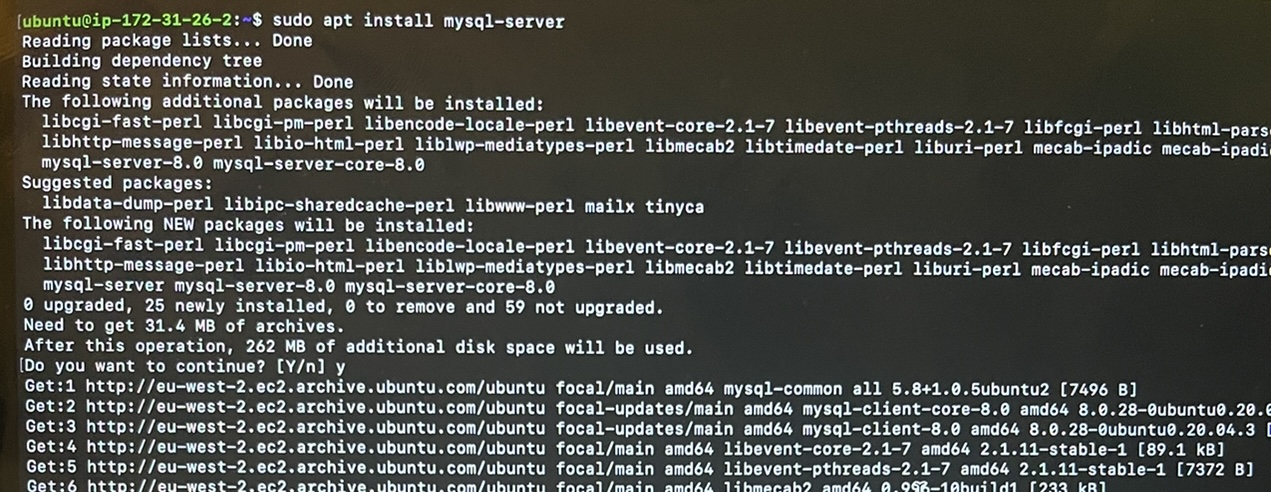
Public Access - Command: [http://172.31.26.2:80](http://172.31.26.2/)

[](https://user-images.githubusercontent.com/93732510/153730141-3c045b3d-f267-4f82-886a-5dcebc2b6026.jpg)

## Step 2 - Installing MYSQL

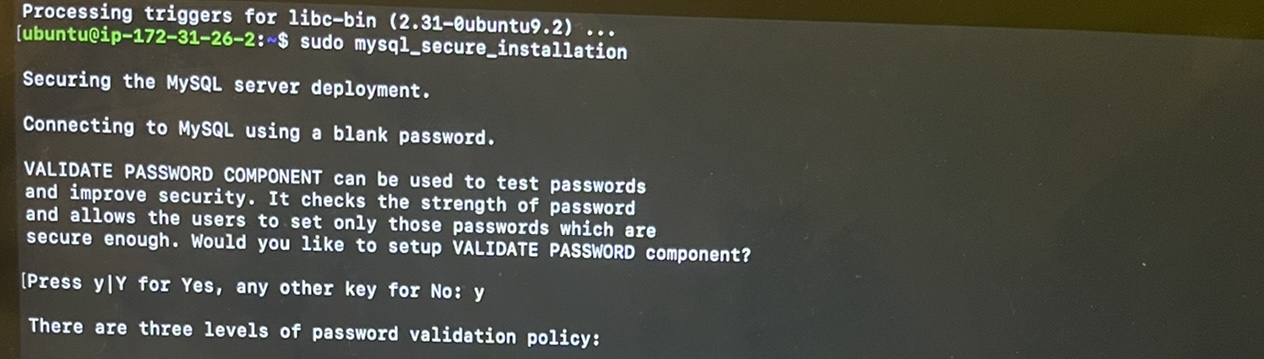
To install mysql on the server, the below command is used.

Command: sudo apt install mysql

[](https://user-images.githubusercontent.com/93732510/153730385-5d0fd77e-93da-4c6f-934e-3d2846c2936f.jpg)

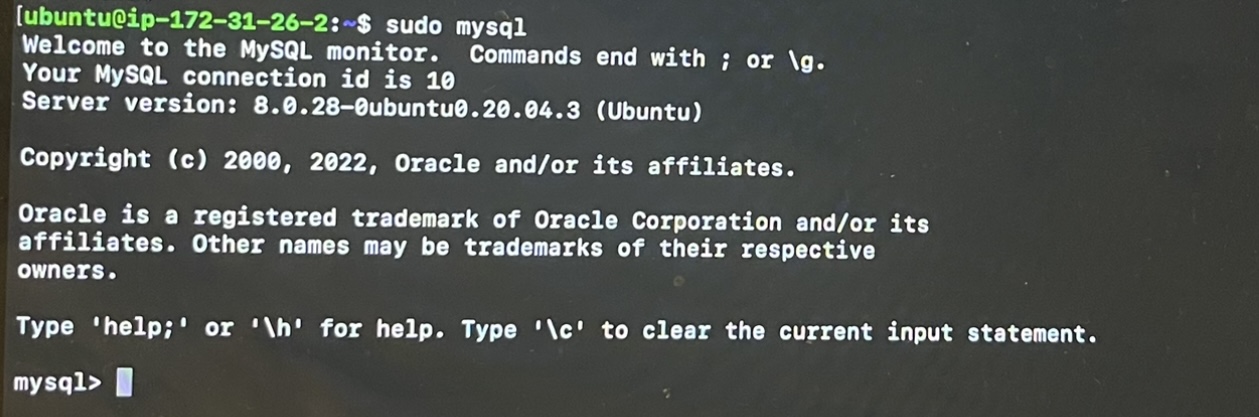
Next is to run a security script to remove some insecure default settings.

Command: $ sudo mysql\_secure\_installation

[](https://user-images.githubusercontent.com/93732510/153730597-5b1c9c26-509b-43f8-8c6b-ecf6b0c1f9c6.jpg)

Now that mysql is installed, i would log in to the console using;

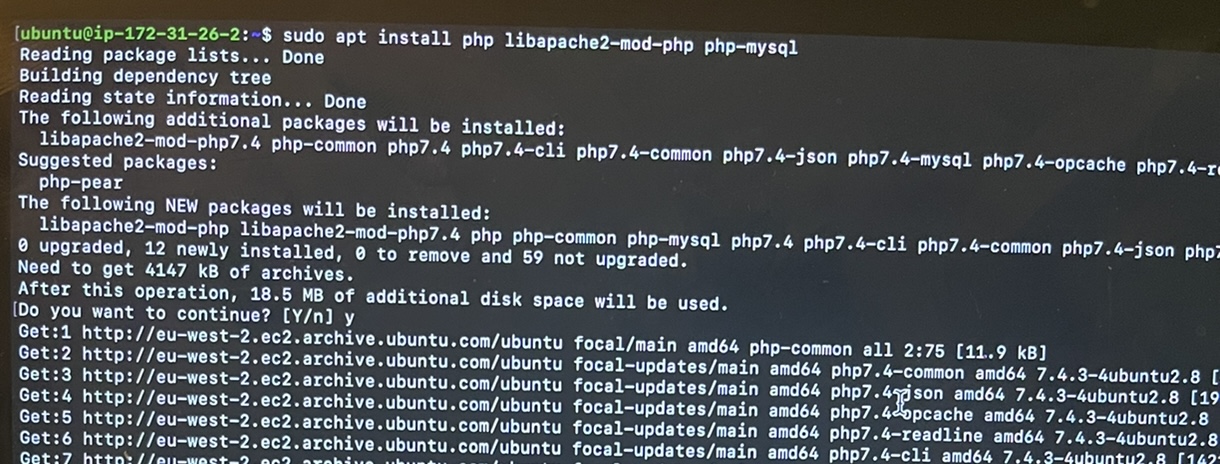
Command: $ sudo mysql

[](https://user-images.githubusercontent.com/93732510/153730753-a5e4d5c1-ffcf-4ab3-a29e-4a474657d0ce.jpg)

## Step 3 - Installing PHP

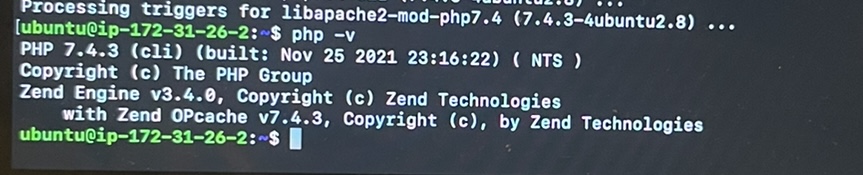
For this step, the PHP package will be installed along with PHP module that allows PHP to communicate with the mysql database and then a PHP file enabler.

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

[](https://user-images.githubusercontent.com/93732510/153746693-82c2cfc5-8990-4aab-a29f-5417cd43d07c.jpg)

Next is to check the PHP version insatlled.

$ php -v

[](https://user-images.githubusercontent.com/93732510/153746840-fd7d47f3-ece3-46bf-a05d-d5150cea46e4.jpg)

## Step 4 - Creating a virtual host for the website using Apache

The domain name will be called projectlamp and a directory is created for it.

Command: $ sudo mkdir /var/www/projectlamp

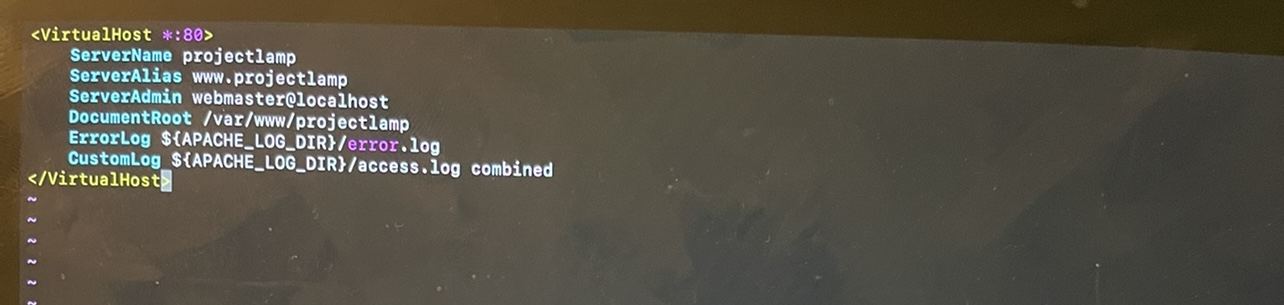
To create ownership of the directory, below command is used;

Command: sudo chown -R $USER:$USER /var/www/projectlamp

Next is to create a new configuration in apache's sites-available directory using command-line editor.

Command: $ sudo vi /etc/apache2/sites-available/projectlamp.conf

Bare-bones configuration is then pasted in the blank file created, as shown below.

[](https://user-images.githubusercontent.com/93732510/153748266-24bc8908-9288-4590-b968-0652c894edcb.jpg)

we can use ls to check the new file in sites-available directory.

Command: $ sudo ls /etc/apache2/sites-available

To enable the virtual host we use a2ensite command.

Command: $ sudo a2ensite projectlamp

There is need to disable apache's default website using a2dissite command.

Command: $ sudo a2dissite 000-default

Then to make sure the configuration file does not contain syntax error we run:

Command: $ sudo apache2ctl configtest

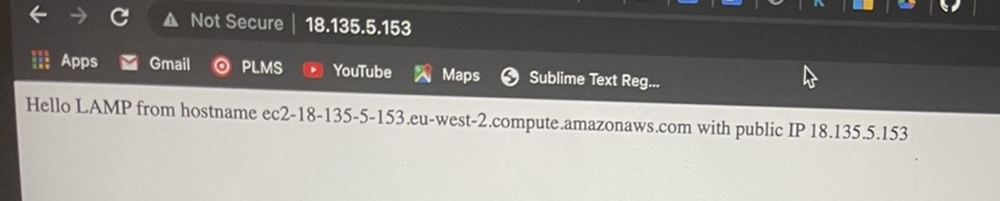
Next is to ensure the changes take effect by reloading apache

Command: $ sudo systemctl reload apache2

To check that the virtual host works, we create an index.html in the web root /var/www/projectlamp

Command: $ sudo echo 'Hello LAMP from hostname' $(curl -s <http://169.254.169.254/latest/meta-data/public-hostname>) 'with public IP' $(curl -s <http://169.254.169.254/latest/meta-data/public-ipv4>) > /var/www/projectlamp/index.html

we can check that the html file has been created using IP address. Below is the output displayed in the browser.

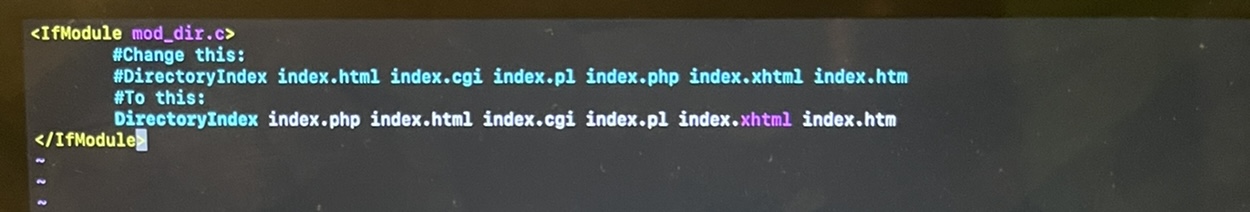
[](https://user-images.githubusercontent.com/93732510/153748872-e363e97f-6ead-46d2-a86d-962c308637ec.jpg)

## Step 5 - Enabling PHP on the website.

Becuase by default in the DirectoryIndex setting on apache, a file named index.html will take precedence over an index.php file, we need to change the order in the directoryindex.

Command: $ sudo vim /etc/apache2/mods-enabled/dir.conf

Then paste the rule is pasted in the blank file created as shown below.

[](https://user-images.githubusercontent.com/93732510/153749628-b8de5bdd-5bd2-4f37-a783-f42ef5c9b6ff.jpg)

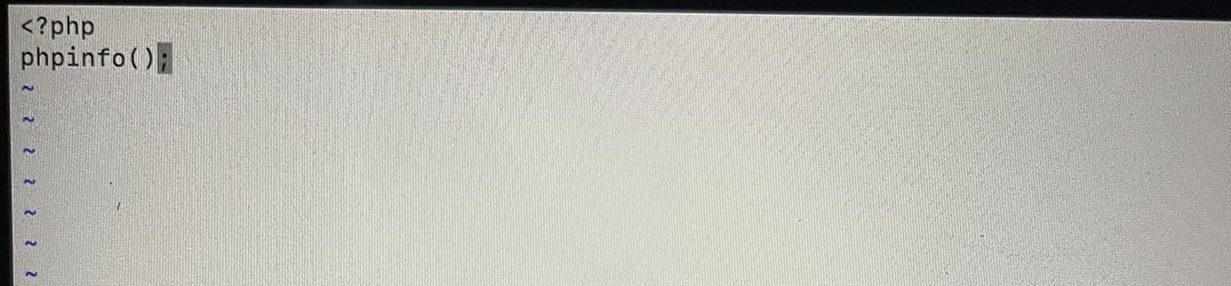
For the changes to take effect, we reload apache.

Command: $ sudo systemctl reload apache2

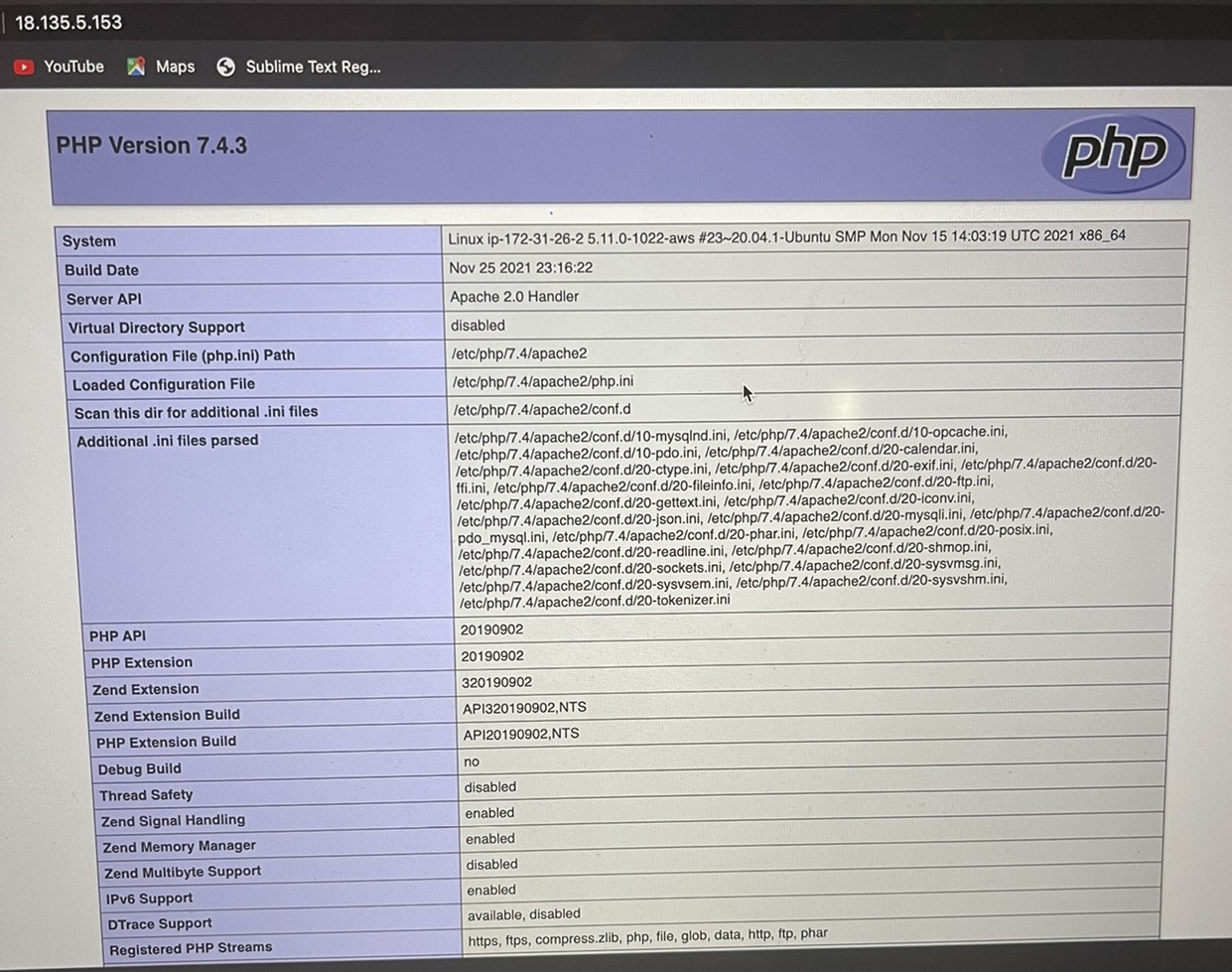
Next is to test that the apache can process requests for PHP files by creating PHP test script.

Command: vim /var/www/projectlamp/index.php

Below PHP code is then pasted in the blank file.

[](https://user-images.githubusercontent.com/93732510/153750006-2e39121f-950e-49e7-840c-da53f73dd79b.jpg)

Finally, we check the browser to see that the PHP file has been created.

[](https://user-images.githubusercontent.com/93732510/153750116-eccfc3a6-d089-437a-a5c8-168a60a6623e.jpg)

To remove the file as it contains private information, we can use the rm command.

Command: $ sudo rm /var/www/projectlamp/index.php