Practical 10
HTTPd, PHP

A. HTTPd (DAEMON)
B. PHP

You may like to watch a video first: https://www.youtube.com/watch?v=PzFO-73mAkg

A. Apache 2 web server

1. Install the HTTP server the command

Update your Ubuntu

```
sudo apt-get update
```

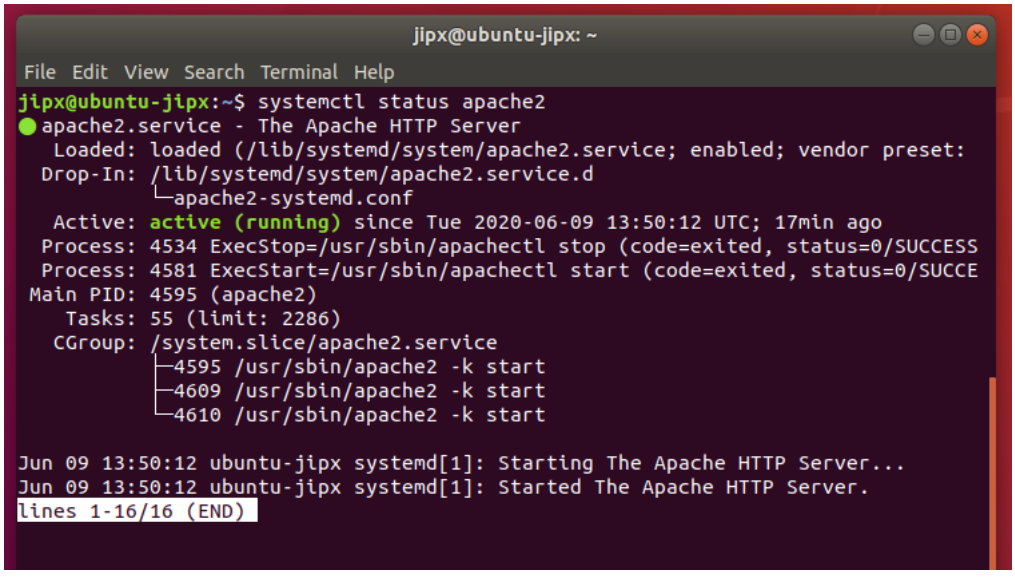
Install apache2

```
sudo apt-get install apache2
```

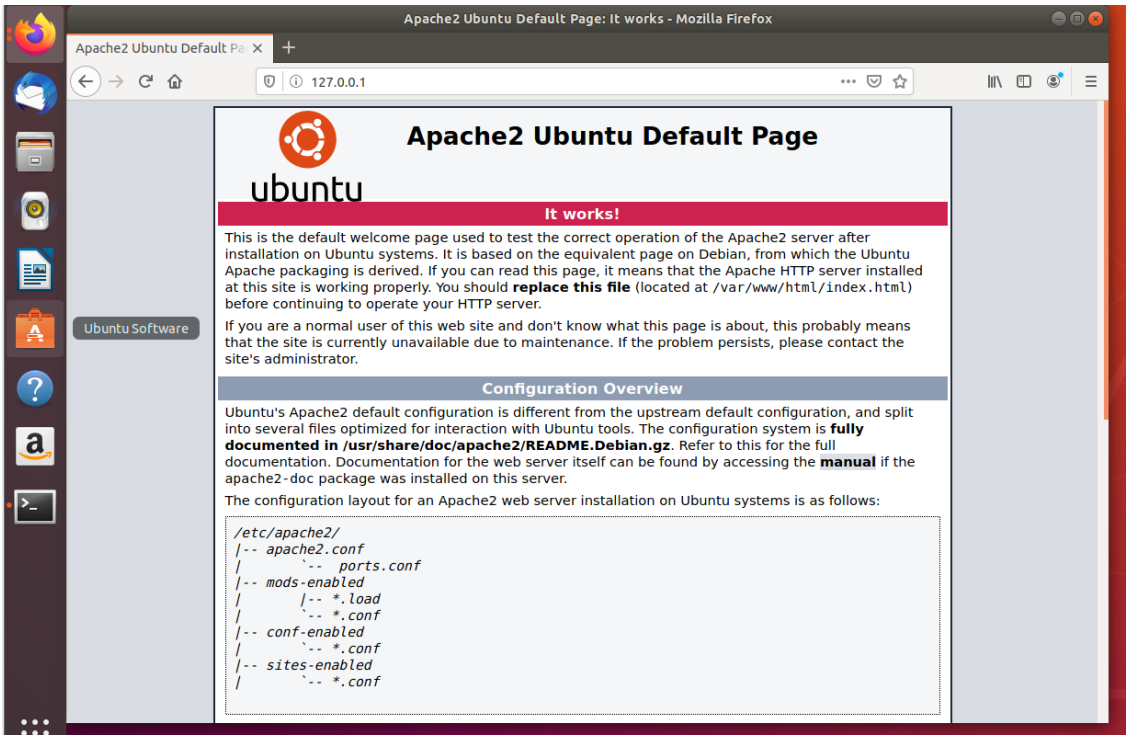
```
jipx@ubuntu-jipx:~$ sudo apt-get install apache2
```

2. Check if apache2 service is running

```
sudo systemctl status apache2
```



3. Using your “Firefox” web browser to go to http://127.0.0.1 to check out the pages.

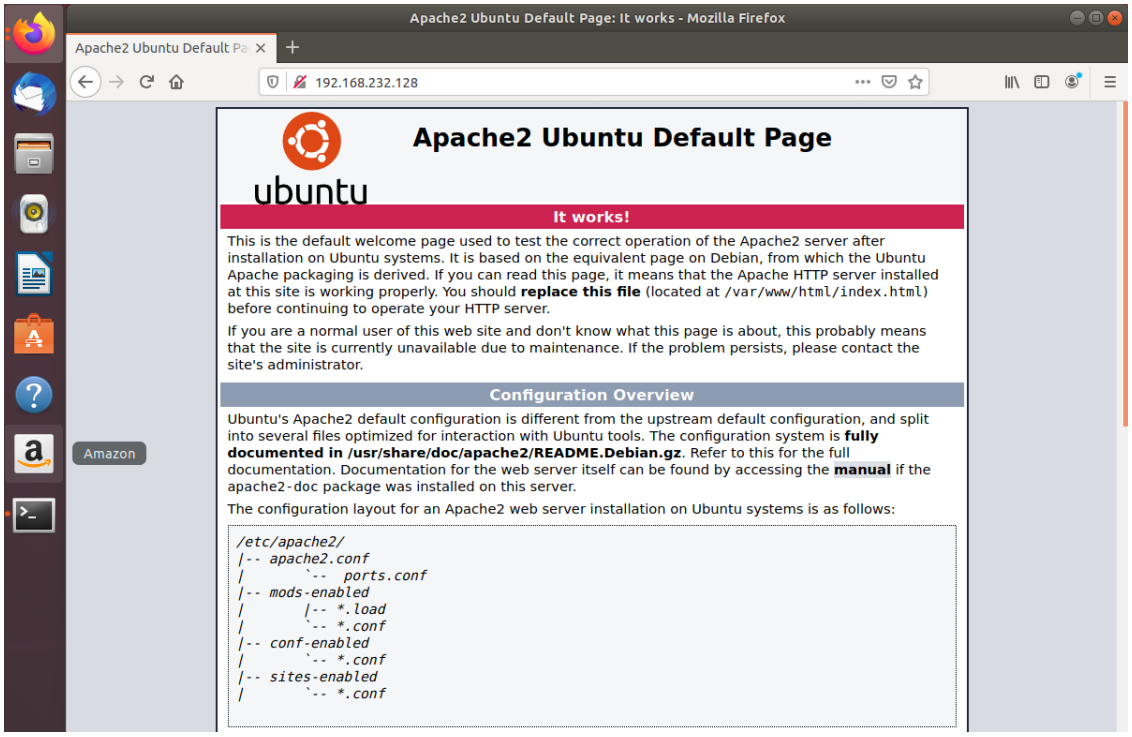


For AWS ubuntu server:
If you encounter an error, such as connection refused, please refer to the **Troubleshooting** section (page 12).

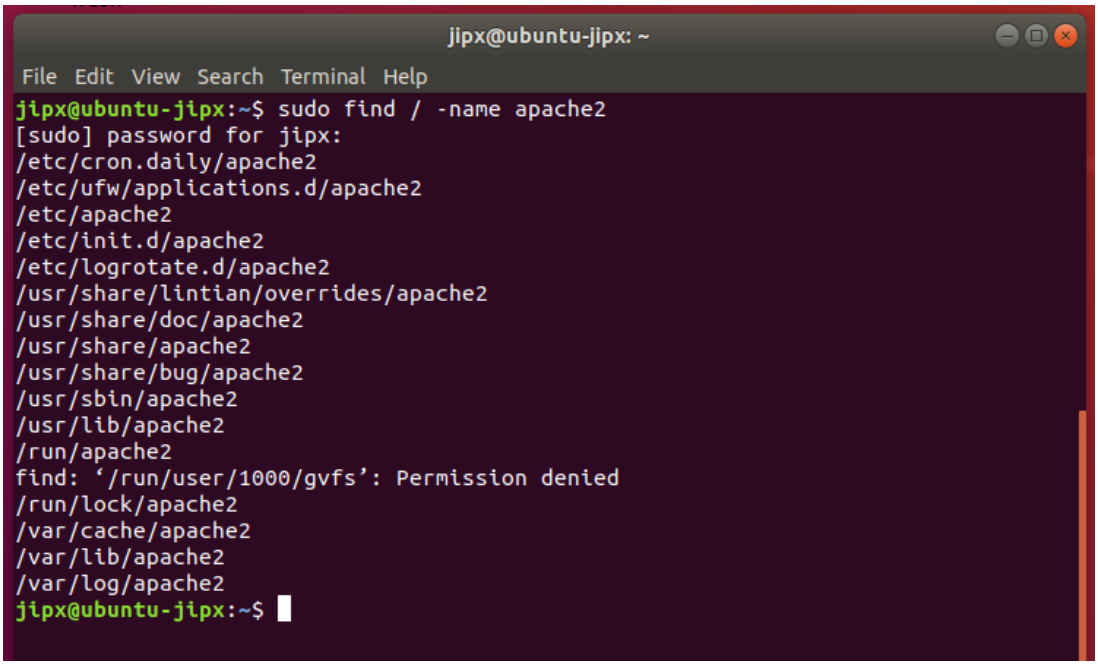
4. You can find the IP address of your server by issuing
   `ifconfig -a`

IP address (of your server)	

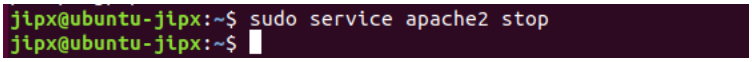
Access your web site using the above IP address



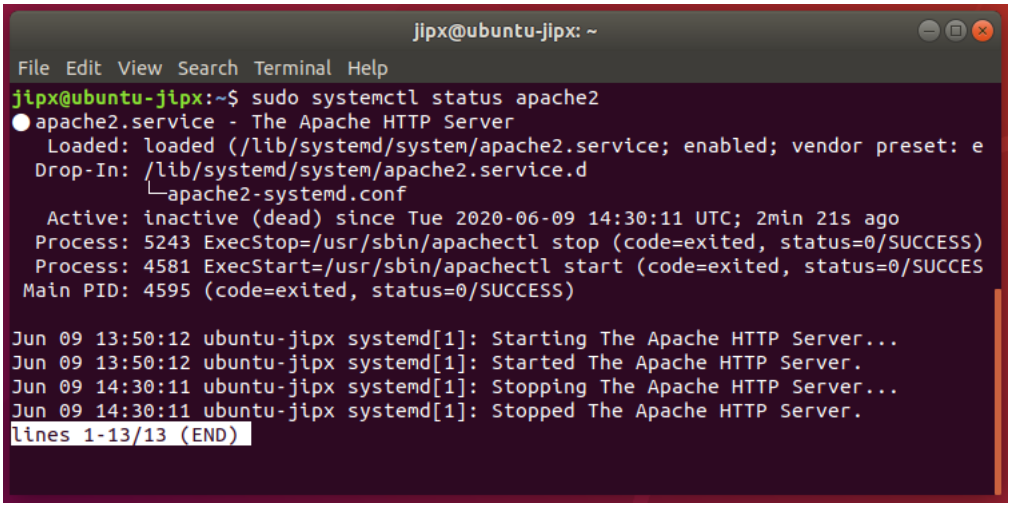
5. Where is Apache HTTP server installed?
   Issue a "**find**" command to reveal all the Apache folders and files location.



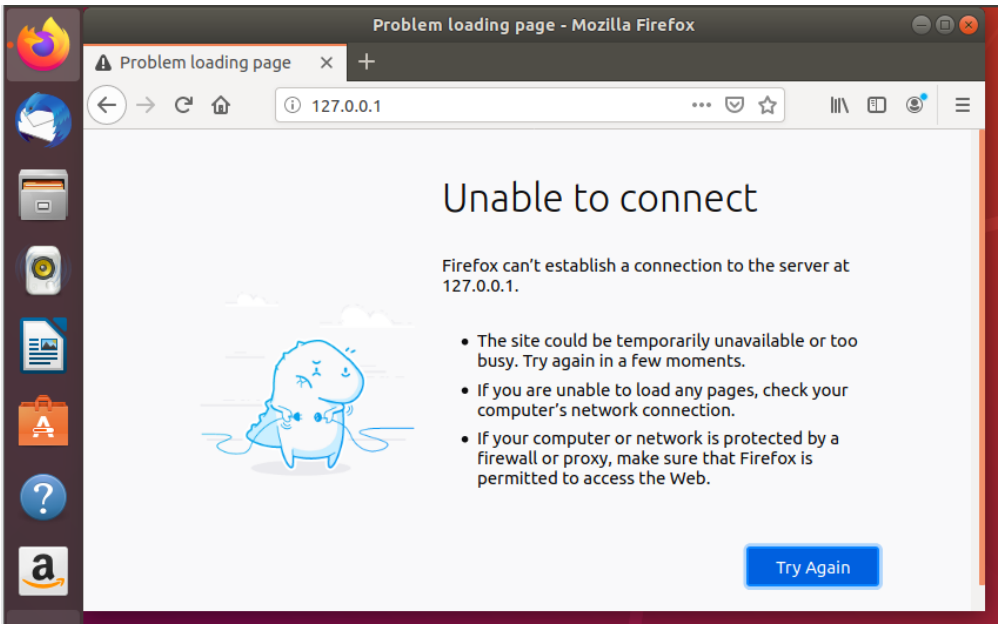
6. Stop your apache2 service using
   `sudo service apache2 stop` or
   `sudo /etc/init.d/apache2 stop`



   Check if apache2 service is stopped
   `sudo systemctl status apache2`



7. Using your “Firefox” web browser to go to http://127.0.0.1 to check out the pages.



8. Restart the service using

```
sudo service apache2 restart          or
sudo /etc/init.d/apache2 restart
```

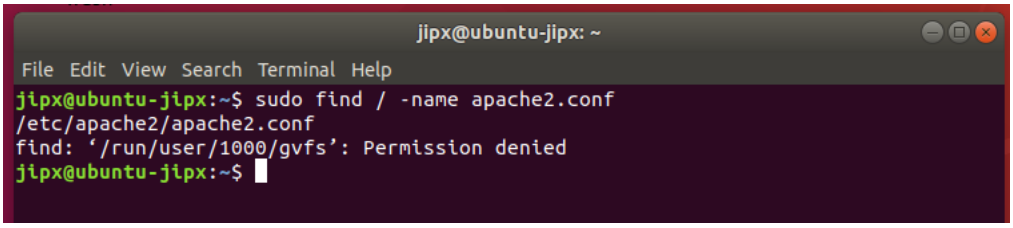
B. Create a web page

1. Check your permission on /var/www folder
   `ls -ld /var/www`

		binary	decimal
Owner: root	rwx	111	7
Group: root	r-x	101	5
other	r-x	101	5

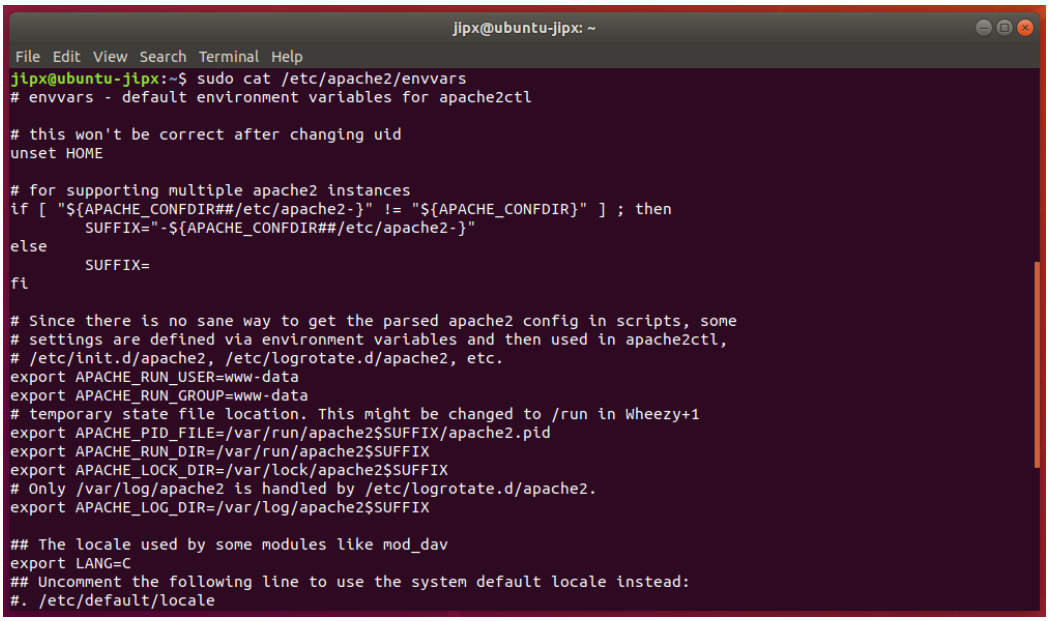
> `/var/www` is the default root folder of your local web server, you host all your website files here and access it on the browser with url like http://127.0.0.1 or http://localhost
>
> You cannot simply copy paste stuff in this folder using your file browser. It is protected for security reasons, you will get a permission denied error message because by default you don't have write access permission here.

2. Type `sudo find / -name apache2.conf` to locate the apache2.conf file



3. Type `sudo cat /etc/apache2/apache2.conf` to find out the folder/directory of Apache web server.

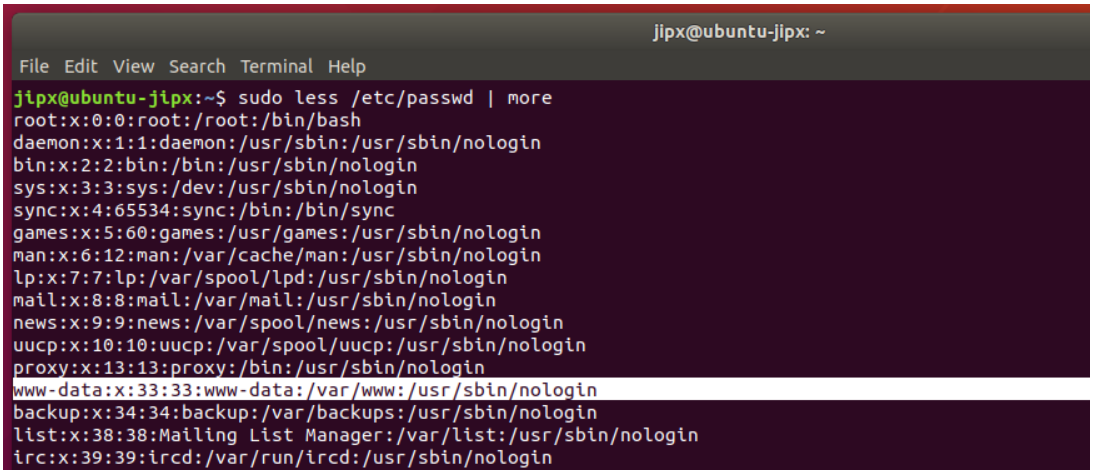
4. Type `sudo cat /etc/apache2/envvars` to find out the user and group the web server process is running under.



Record down the environment variable for `APACHE_RUN_USER` and `APCHE_RUN_GROUP`

APACHE_RUN_USER	
APCHE-RUN-GROUP	

5. Issue the following command
   `sudo less /etc/passwd | more`

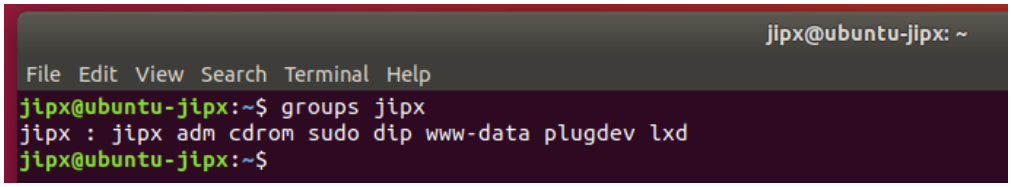


6. Change the group for the folder
   `sudo chgrp -R www-data /var/www`

> The group ownership of a file or directory may be changed with `chgrp`.
>
> In the command above, we changed the group ownership of `/var/www` from its previous group to "`www-data`". You must be the owner of the file or directory to perform a `chgrp`.

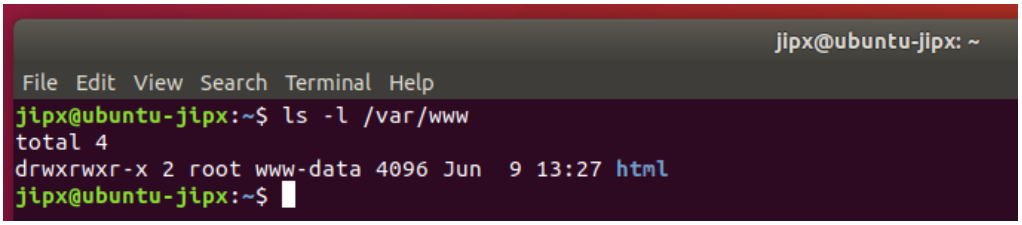
7. Add yourself to the group www-data
   `sudo usermod -a -G www-data jipx`

8. Check if above user (`jipx`) has been added to www-data group
   `groups jipx`



9. Change the document permissions for the group
   `sudo chmod -R g+w /var/www`

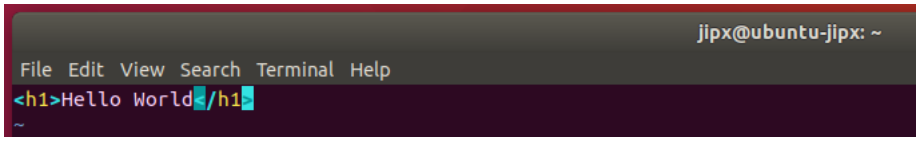
   Check the folder setting
   `ls -l /var/www`



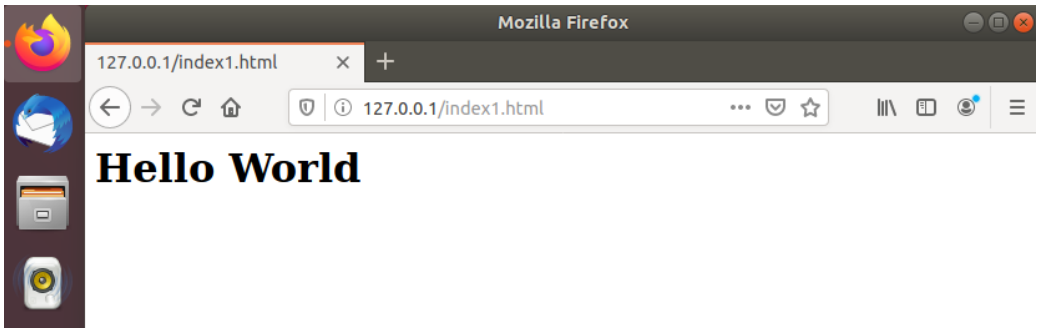
		binary	decimal
Owner: root	rwx	111	7
Group: www-data	rwx		
other	r-x	101	5

`sudo chmod -R g+w /var/www/html`

10. Create a file `sudo vi /var/www/html/index1.html` with the following content
    `<h1> Hello World </h1>`



11. Using your “Firefox” web browser to go to http://127.0.0.1/index1.html



C. Install PHP

PHP 7.2 which is the default PHP version in Ubuntu 18.04 is fully supported and recommended for WordPress.

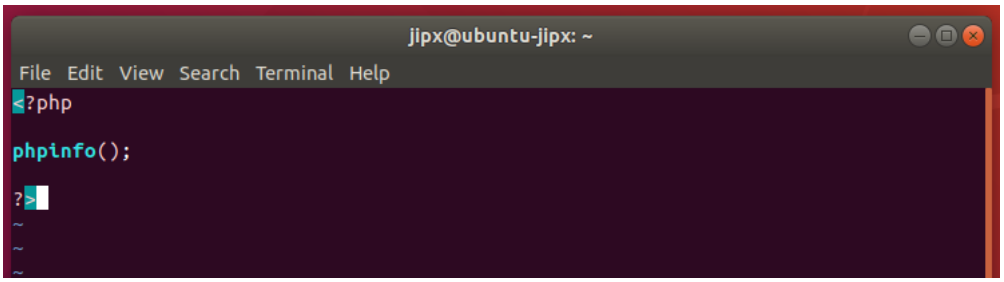
To install PHP and all required PHP extensions, run the following command:

```
sudo apt install php7.2 php7.2-cli php7.2-mysql php7.2-json
php7.2-opcache php7.2-mbstring php7.2-xml php7.2-gd php7.2-
curl
```

Restart apache2 so the newly installed PHP extensions are loaded:

```
sudo systemctl restart apache2
```

```
sudo vi /var/www/html/test.php
```

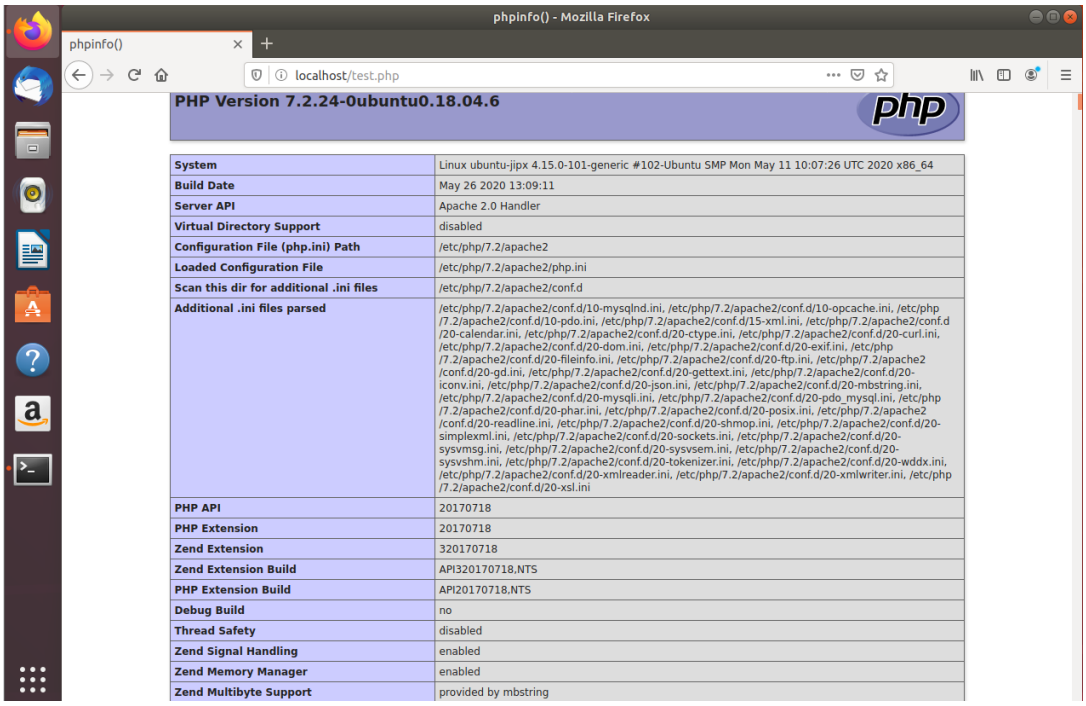


Now we can test whether our web server can correctly display content generated by a PHP script. To try this out, we just have to visit this page in our web browser. You'll need your server's public IP address again.

The address you want to visit will be:

```
http://your_server_IP_address/test.php          or
http://localhost/test.php
```

The page that you come to should look something like this:



System	Linux ubuntu-jipx 4.15.0-101-generic #102-Ubuntu SMP Mon May 11 10:07:26 UTC 2020 x86_64
Build Date	May 26 2020 13:09:11
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.2/apache2
Loaded Configuration File	/etc/php/7.2/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.2/apache2/conf.d
Additional .ini files parsed	/etc/php/7.2/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.2/apache2/conf.d/10-opcache.ini, /etc/php/7.2/apache2/conf.d/10-pdo.ini, /etc/php/7.2/apache2/conf.d/15-xml.ini, /etc/php/7.2/apache2/conf.d/20-calendar.ini, /etc/php/7.2/apache2/conf.d/20-ctype.ini, /etc/php/7.2/apache2/conf.d/20-curl.ini, /etc/php/7.2/apache2/conf.d/20-dom.ini, /etc/php/7.2/apache2/conf.d/20-exif.ini, /etc/php/7.2/apache2/conf.d/20-fileinfo.ini, /etc/php/7.2/apache2/conf.d/20-ftp.ini, /etc/php/7.2/apache2/conf.d/20-gd.ini, /etc/php/7.2/apache2/conf.d/20-gettext.ini, /etc/php/7.2/apache2/conf.d/20-iconv.ini, /etc/php/7.2/apache2/conf.d/20-json.ini, /etc/php/7.2/apache2/conf.d/20-mbstring.ini, /etc/php/7.2/apache2/conf.d/20-mysqli.ini, /etc/php/7.2/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.2/apache2/conf.d/20-phar.ini, /etc/php/7.2/apache2/conf.d/20-posix.ini, /etc/php/7.2/apache2/conf.d/20-readline.ini, /etc/php/7.2/apache2/conf.d/20-shmop.ini, /etc/php/7.2/apache2/conf.d/20-simplexml.ini, /etc/php/7.2/apache2/conf.d/20-sockets.ini, /etc/php/7.2/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.2/apache2/conf.d/20-sysvsem.ini, /etc/php/7.2/apache2/conf.d/20-sysvshm.ini, /etc/php/7.2/apache2/conf.d/20-tokenizer.ini, /etc/php/7.2/apache2/conf.d/20-wddx.ini, /etc/php/7.2/apache2/conf.d/20-xmlreader.ini, /etc/php/7.2/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.2/apache2/conf.d/20-xsl.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718,NTS
PHP Extension Build	API20170718,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring

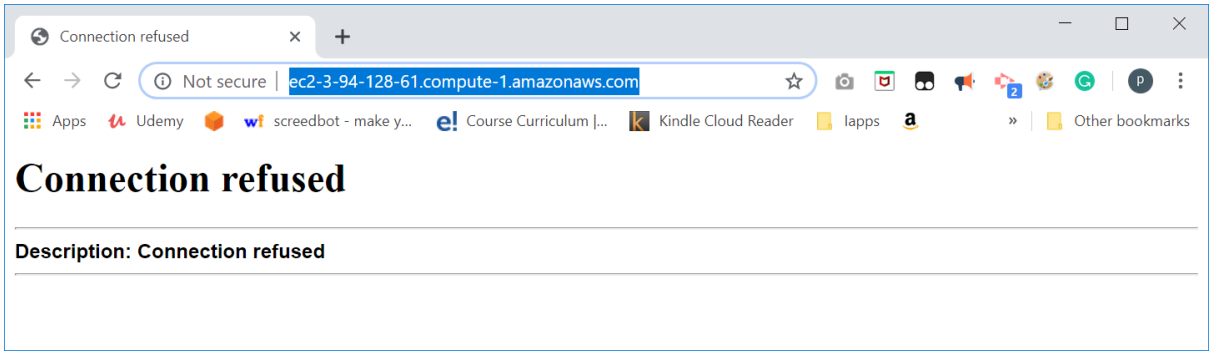
This page basically gives you information about your server from the perspective of PHP. It is useful for debugging and to ensure that your settings are being applied correctly. If this was successful, then your PHP is working as expected.

You probably want to remove this file after this test because it could actually give information about your server to unauthorized users. To do this, you can type this:

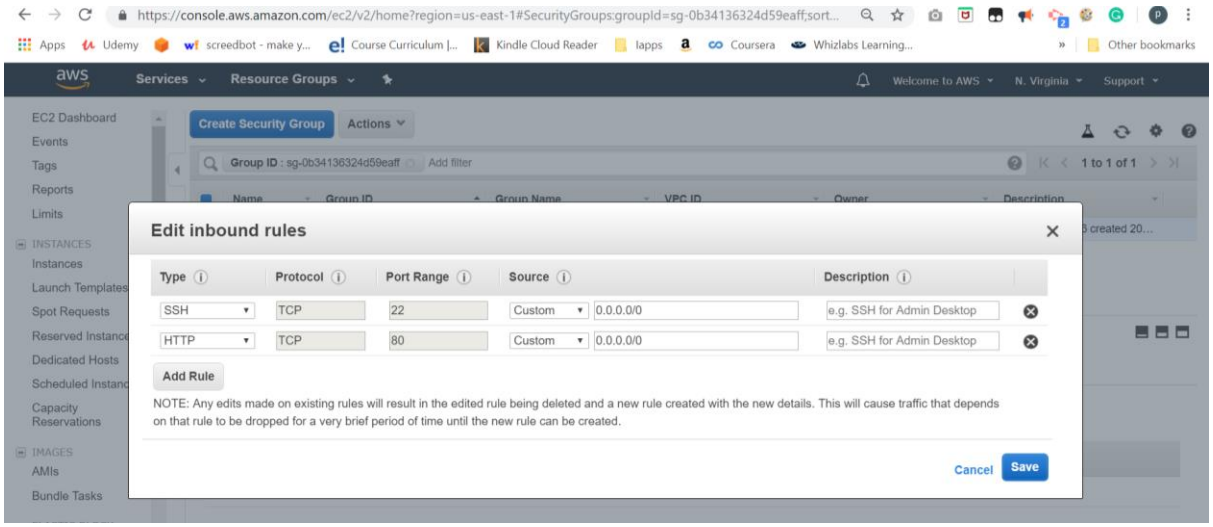
```
sudo rm /var/www/html/test.php
```

** You can always recreate this page if you need to access the information again later.

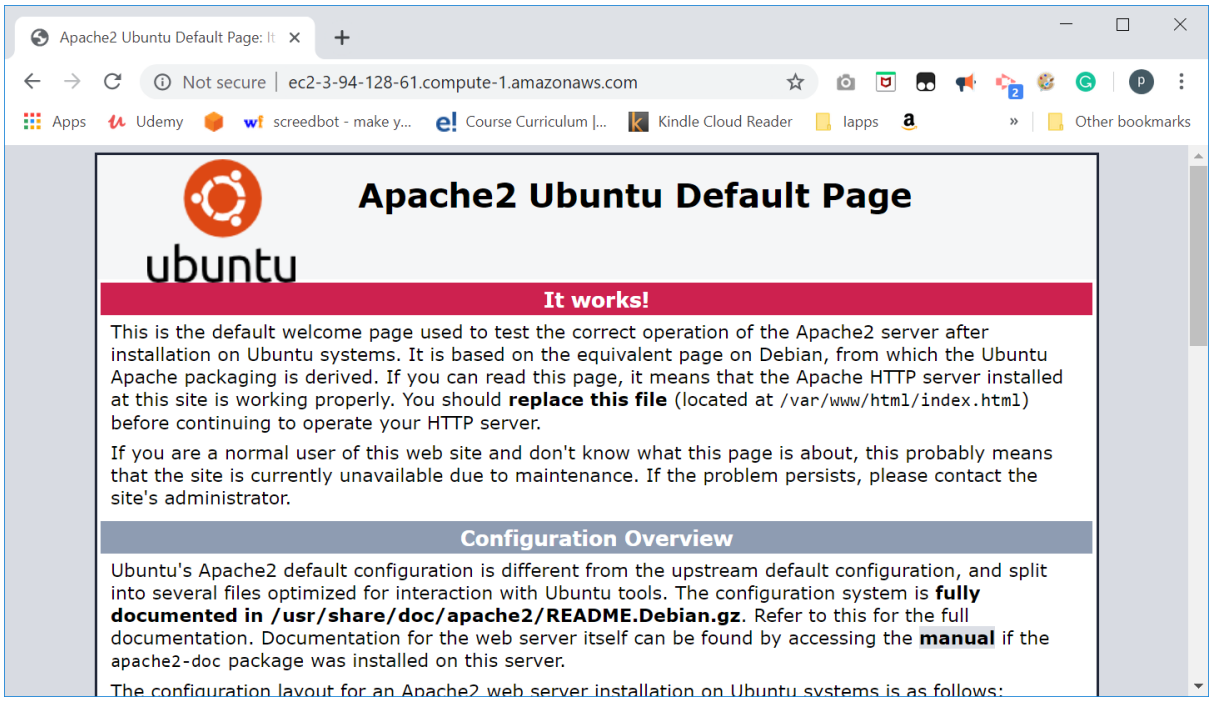
Troubleshooting



Go to AWS Console to Change the security group for the EC2 instance.



After saving the security group, it will take effect immediately.



Reference: How To Install Linux, Apache, MySQL, PHP (LAMP) stack on Ubuntu 16.04

End of Practical