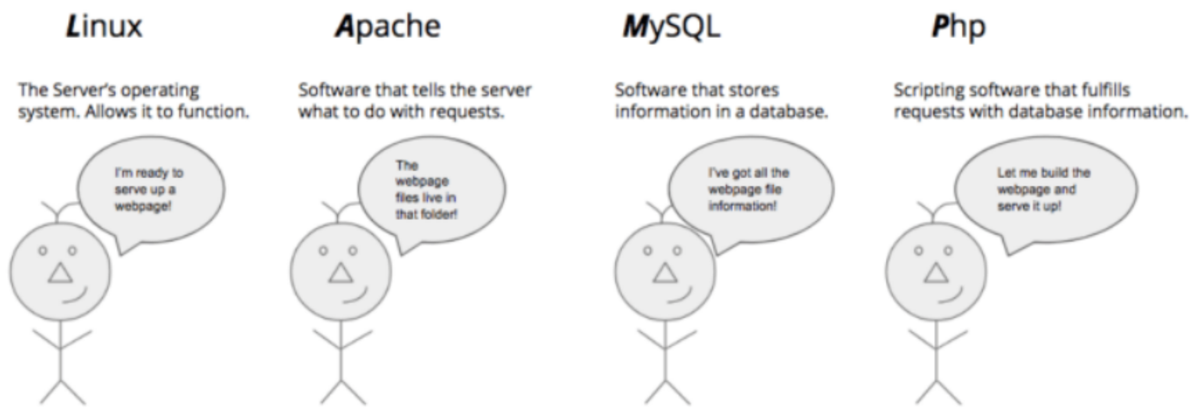
PROJECT 1: LAMP STACK IMPLEMENTATION

Ever heard of the word LAMP Stack?

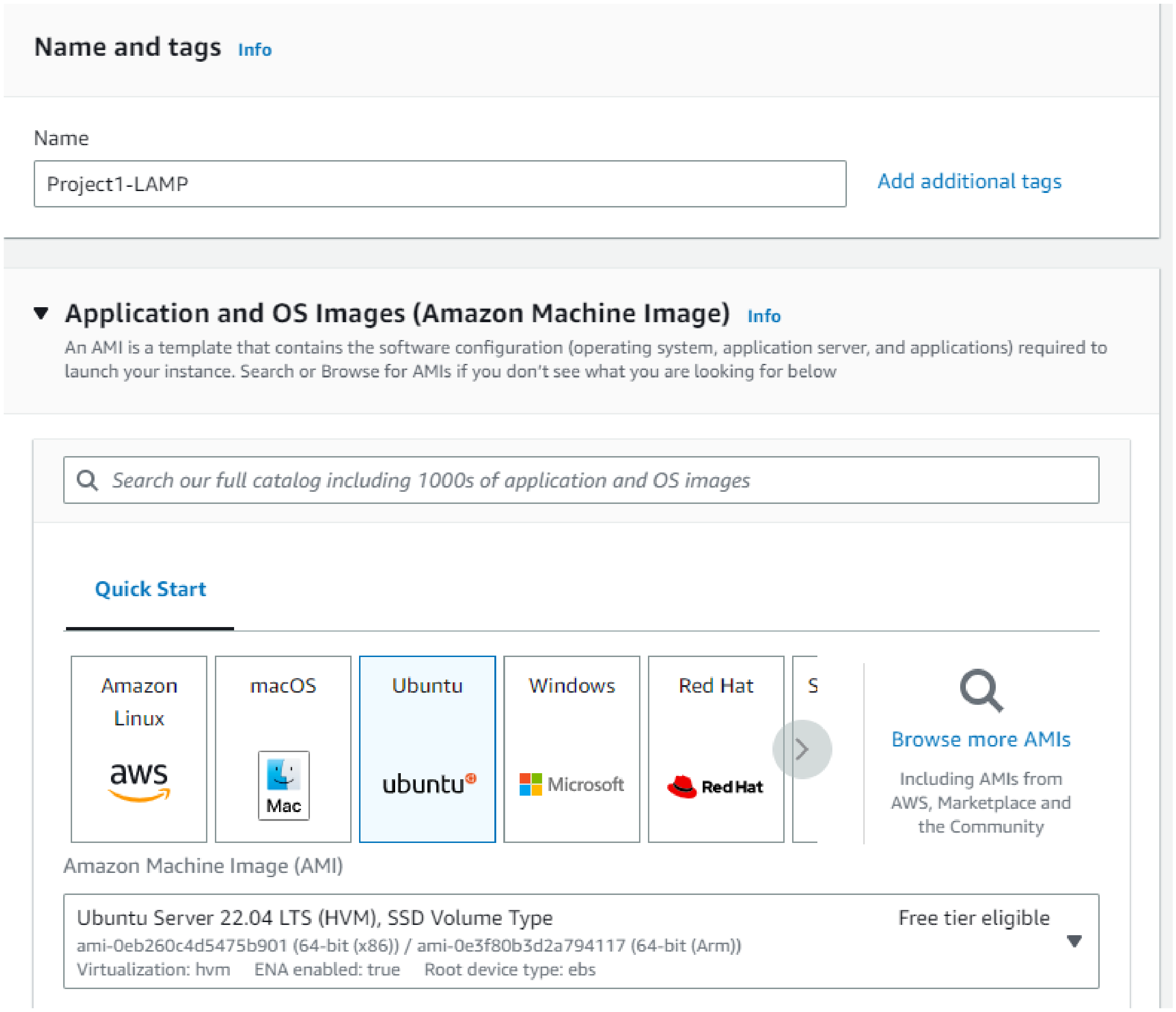
LAMP is an open-source web development framework that combines the Linux operating system, Apache as the web server, MySQL as the database system, and PHP/Perl as the back-end programming language. It is one of the most popular technologies that work together to create a platform for executing web applications. LAMP offers complete server administration and remote access, making it possible to execute administrative tasks on a Linux server from anywhere. A typical illustration of the different layers can be found in the diagram below.


Linux
The Server's operating system. Allows it to function.
I'm ready to serve up a webpage!
Apache
Software that tells the server what to do with requests.
The webpage files live in that folder!
MySQL
Software that stores information in a database.
I've got all the webpage file information!
Php
Scripting software that fulfills requests with database information.
Let me build the webpage and serve it up!

Creating an EC2 Instance

The first step in the project implementation is to create an AWS account which will be used to provision an Ubuntu Server to enable us to connect to an EC2 instance to get the work started. I signed up to the free tier account which gives me access to 750 hours for a year, so I better use the hours wisely. Let's get started 🙂…

Select a name for the instance, here I used Project 1-LAMP and I selected the Ubuntu 22.04 OS Image.



Name and tags Info

Name

Project1-LAMP

Add additional tags

Application and OS Images (Amazon Machine Image) Info

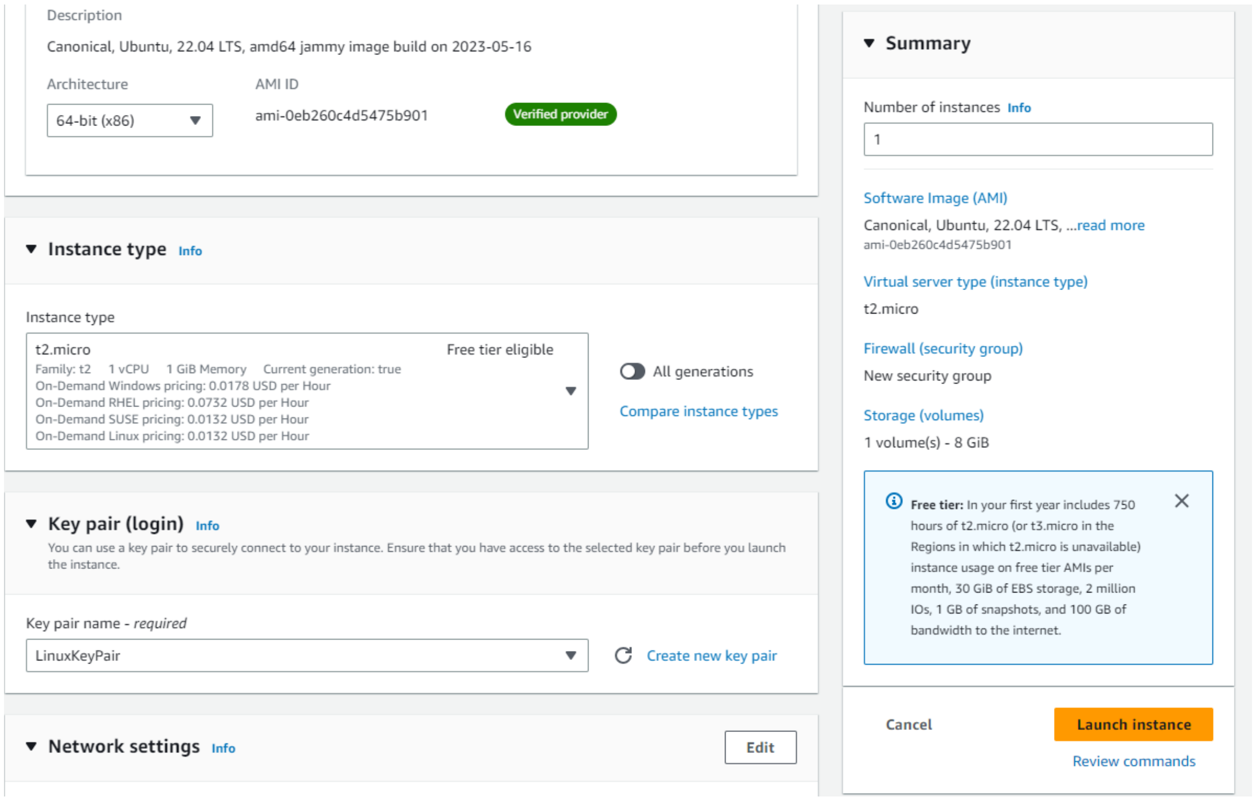
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type — Free tier eligible
ami-0eb260c4d5475b901 (64-bit (x86)) / ami-0e3f80b3d2a794117 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

The instance type will be left at default and then I generated a keypair named - LinuxKeyPair (steps to generate a keypair can be found online). For now, I will leave the default security group and revert to it later and then I clicked on Launch instance.



Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-05-16

Architecture: 64-bit (x86)

AMI ID: ami-0eb260c4d5475b901 — Verified provider

Instance type Info

Instance type

t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows pricing: 0.0178 USD per Hour
On-Demand RHEL pricing: 0.0732 USD per Hour
On-Demand SUSE pricing: 0.0132 USD per Hour
On-Demand Linux pricing: 0.0132 USD per Hour

All generations

Compare instance types

Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*: LinuxKeyPair

Create new key pair

Network settings Info — Edit

Summary

Number of instances Info: 1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more
ami-0eb260c4d5475b901

Virtual server type (instance type)
t2.micro

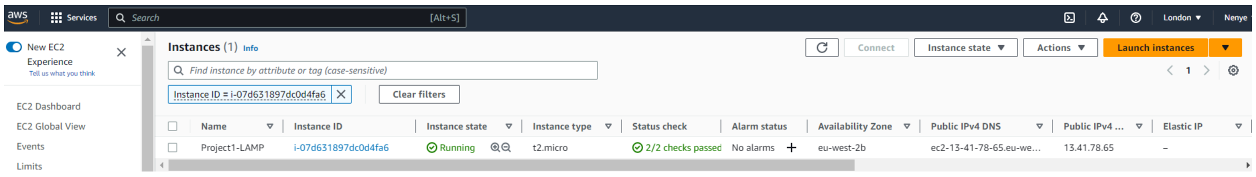
Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel — Launch instance — Review commands

Instance state should appear as running once we have everything sorted.



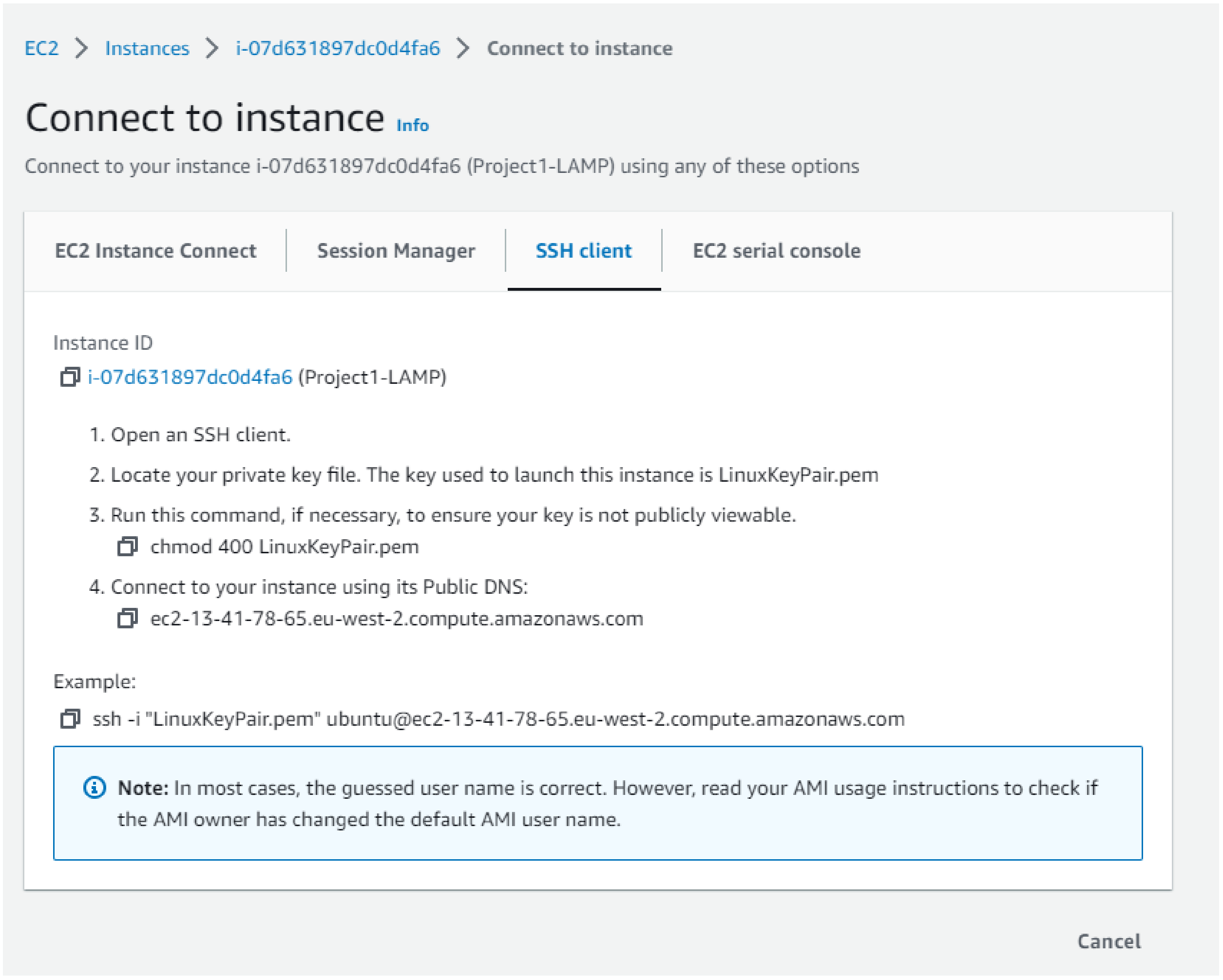
Instances (1) Info

Find instance by attribute or tag (case-sensitive)

Instance ID = i-07d631897dc0d4fa6

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Project1-LAMP	i-07d631897dc0d4fa6	Running	t2.micro	2/2 checks passed	No alarms	eu-west-2b	ec2-13-41-78-65.eu-we...	13.41.78.65	–

To connect to an instance, I opened the ssh client and copied the command to be run in my local terminal.


EC2 > Instances > i-07d631897dc0d4fa6 > Connect to instance

Connect to instance Info

Connect to your instance i-07d631897dc0d4fa6 (Project1-LAMP) using any of these options

Instance ID

i-07d631897dc0d4fa6 (Project1-LAMP)

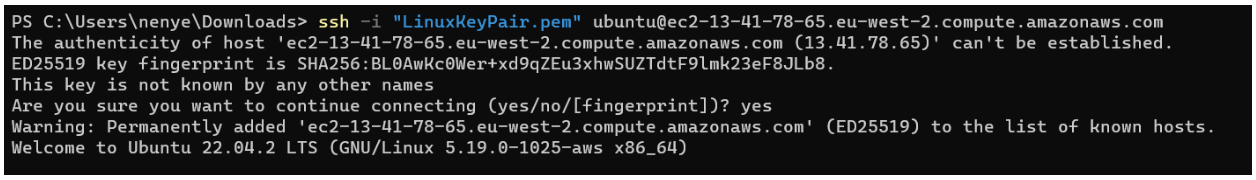
1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is LinuxKeyPair.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
   chmod 400 LinuxKeyPair.pem
4. Connect to your instance using its Public DNS:
   ec2-13-41-78-65.eu-west-2.compute.amazonaws.com

Example:

ssh -i "LinuxKeyPair.pem" ubuntu@ec2-13-41-78-65.eu-west-2.compute.amazonaws.com

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

Remember to cd into the folder where you have saved your Key pair. For me, it is the download folder. I ran the ssh command copied from the ssh client and connected successfully.



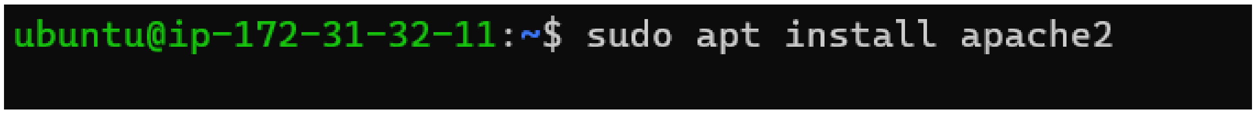
```
PS C:\Users\nenye\Downloads> ssh -i "LinuxKeyPair.pem" ubuntu@ec2-13-41-78-65.eu-west-2.compute.amazonaws.com
The authenticity of host 'ec2-13-41-78-65.eu-west-2.compute.amazonaws.com (13.41.78.65)' can't be established.
ED25519 key fingerprint is SHA256:BL0AwKc0Wer+xd9qZEu3xhwSUZTdtF9lmk23eF8JLb8.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-41-78-65.eu-west-2.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)
```

Remember it is the LAMP Stack Implementation so the next step will be to have Apache running.

Installing Apache

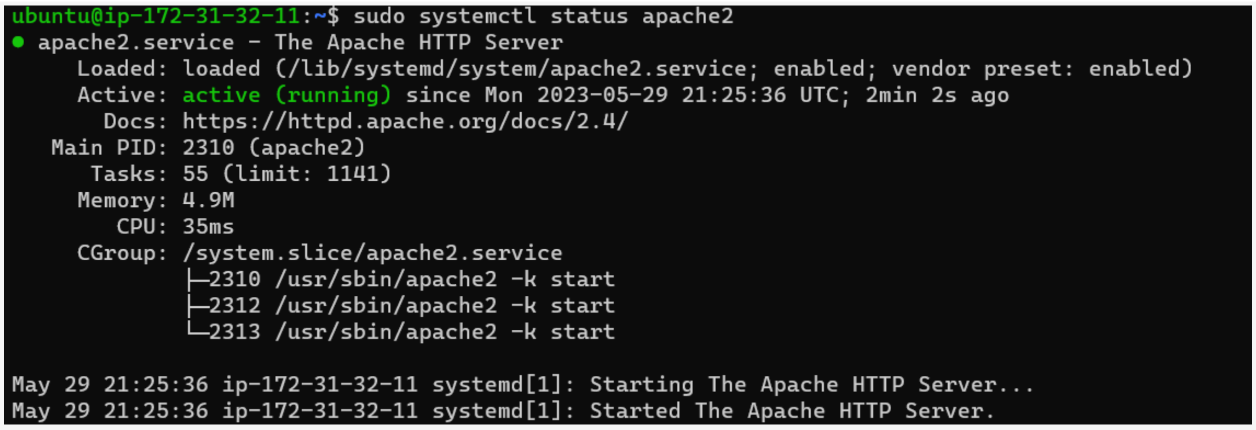
Apache is the most commonly used webserver application, it is an open-source software that may be downloaded for free. As a best practice, before commencing with any installation, it is always important to run the sudo apt update command to update the list of packages for Ubuntu once an ssh connection has been established.

To install the apache2 package run the below command and type y when prompted. Alternatively, you can run the command as sudo apt install apache2 -y



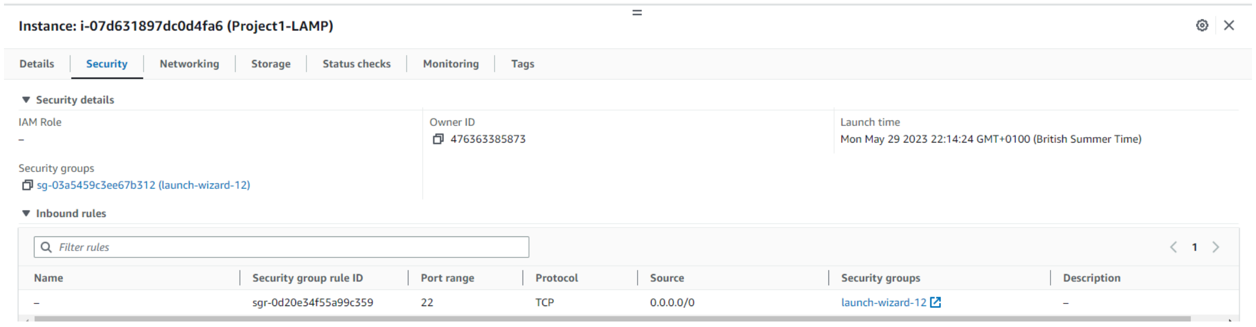
```
ubuntu@ip-172-31-32-11:~$ sudo apt install apache2
```

I ensured that the apache2 service is running by running the following command. The status started, active and running already validates that the service is up.



```
ubuntu@ip-172-31-32-11:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2023-05-29 21:25:36 UTC; 2min 2s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2310 (apache2)
      Tasks: 55 (limit: 1141)
     Memory: 4.9M
        CPU: 35ms
     CGroup: /system.slice/apache2.service
             ├─2310 /usr/sbin/apache2 -k start
             ├─2312 /usr/sbin/apache2 -k start
             └─2313 /usr/sbin/apache2 -k start

May 29 21:25:36 ip-172-31-32-11 systemd[1]: Starting The Apache HTTP Server...
May 29 21:25:36 ip-172-31-32-11 systemd[1]: Started The Apache HTTP Server.
```

The next step now will be to open our TCP Port 80 on our EC2 configuration to be able to receive traffic on our web server as unencrypted web pages are sent and received on this network port by default. To do this, select the security groups and edit inbound rules. Choose to Add rule, select HTTP from the drop-down and then include 80 as the port number



Instance: i-07d631897dc0d4fa6 (Project1-LAMP)

Security details

IAM Role: –

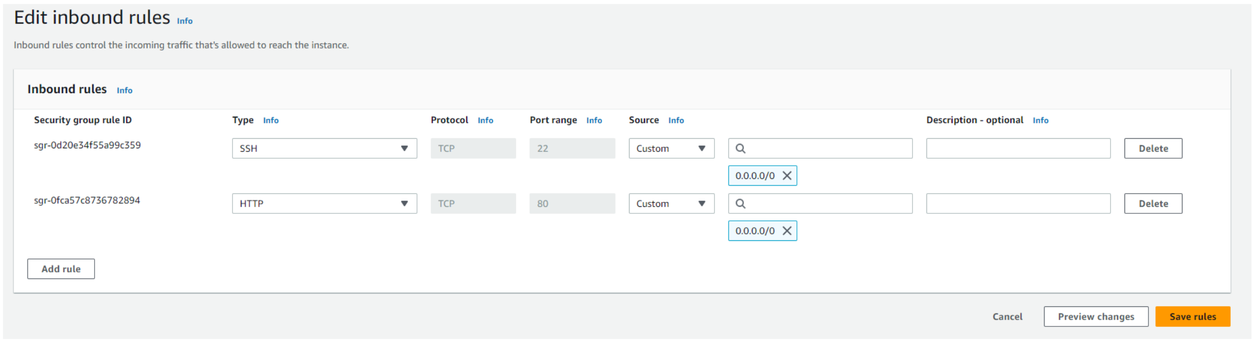
Owner ID: 476363385873

Launch time: Mon May 29 2023 22:14:24 GMT+0100 (British Summer Time)

Security groups: sg-03a5459c3ee67b312 (launch-wizard-12)

Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
–	sgr-0d20e34f55a99c359	22	TCP	0.0.0.0/0	launch-wizard-12	–



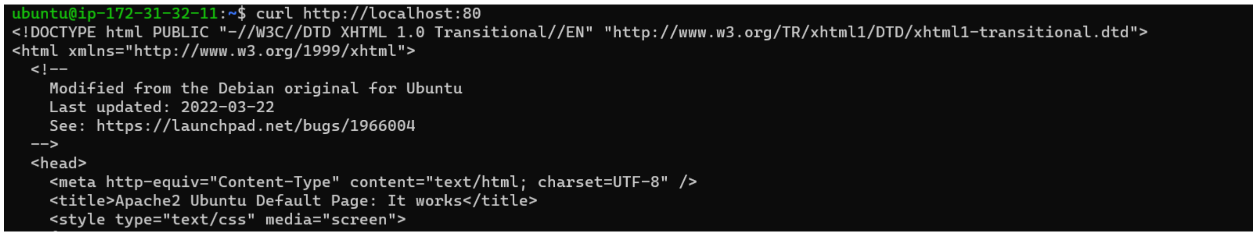
Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

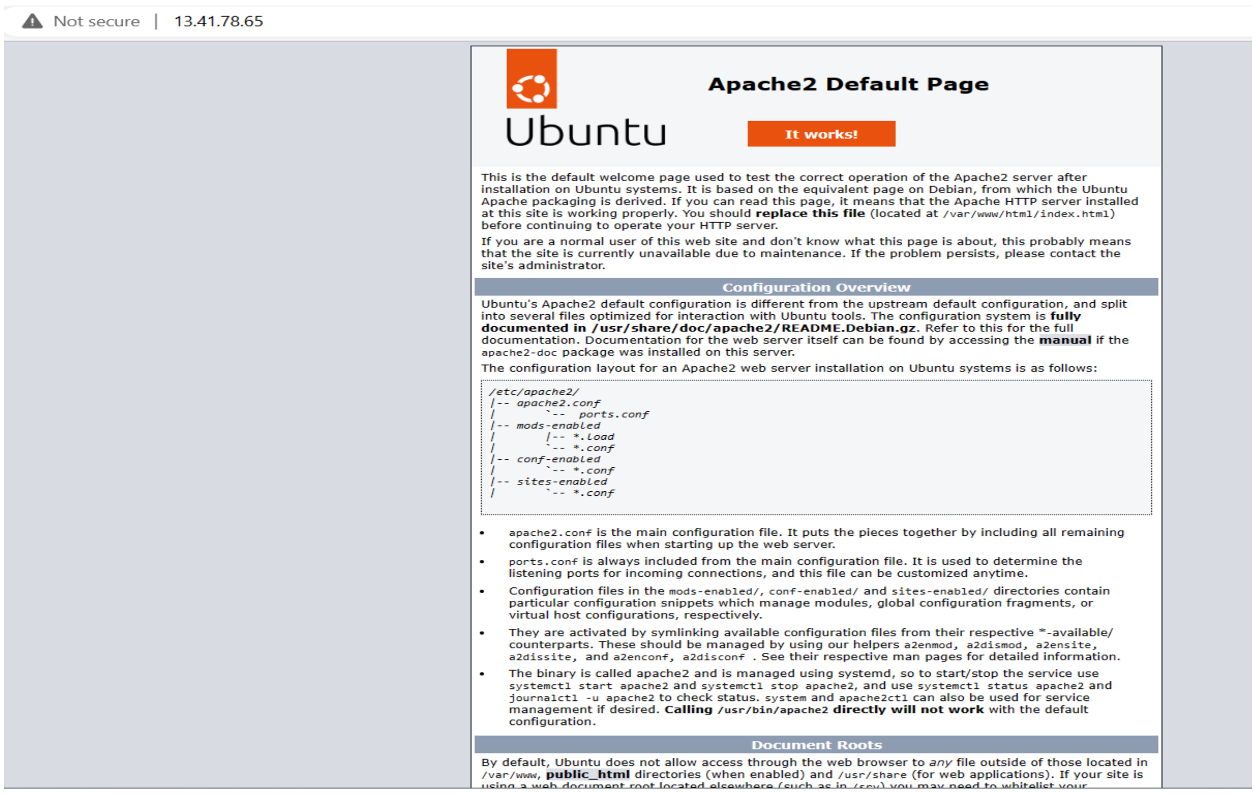
Inbound rules Info

Security group rule ID	Type	Protocol	Port range	Source		Description - optional
sgr-0d20e34f55a99c359	SSH	TCP	22	Custom	0.0.0.0/0	
sgr-0fca57c8736782894	HTTP	TCP	80	Custom	0.0.0.0/0	

We will now need to run the below command to see if we can access this remotely in Ubuntu, and then use the public IP address to confirm this as well on our browser while including the port number e.g. http://13.41.78.65:80. The content of the page should appear similar to what we have on the terminal.



```
ubuntu@ip-172-31-32-11:~$ curl http://localhost:80
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2022-03-22
    See: https://launchpad.net/bugs/1966004
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
```



Not secure | 13.41.78.65

Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

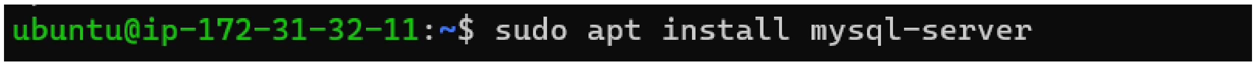
- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective *-available/ counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf` . See their respective man pages for detailed information.
- The binary is called apache2 and is managed using systemd, so to start/stop the service use `systemctl start apache2` and `systemctl stop apache2`, and use `systemctl status apache2` and `journalctl -u apache2` to check status. `system` and `apache2ctl` can also be used for service management if desired. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

Document Roots

By default, Ubuntu does not allow access through the web browser to *any* file outside of those located in `/var/www`, **public_html** directories (when enabled) and `/usr/share` (for web applications). If your site is

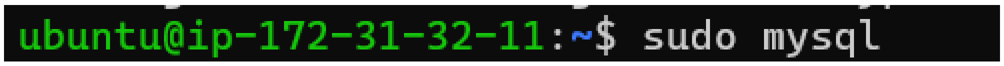
Installing MySQL

An open-source relational database management system is called MySQL. To install this, I ran the sudo apt install command. As always type y for yes when prompted during the installation.



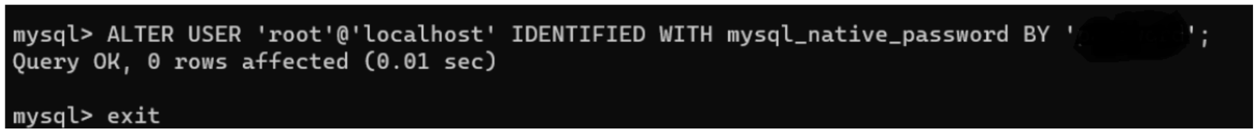
```
ubuntu@ip-172-31-32-11:~$ sudo apt install mysql-server
```

To connect to the MySQL console run



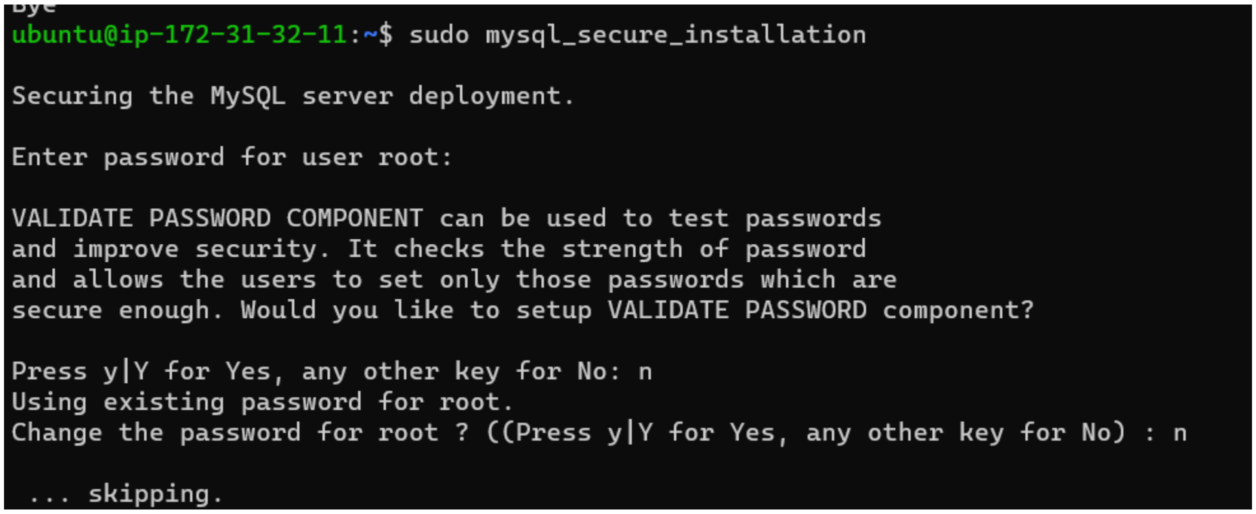
```
ubuntu@ip-172-31-32-11:~$ sudo mysql
```

To guarantee that our database is protected using mysql_native_password as the default authentication mechanism, I will need to set up a password for our root user.



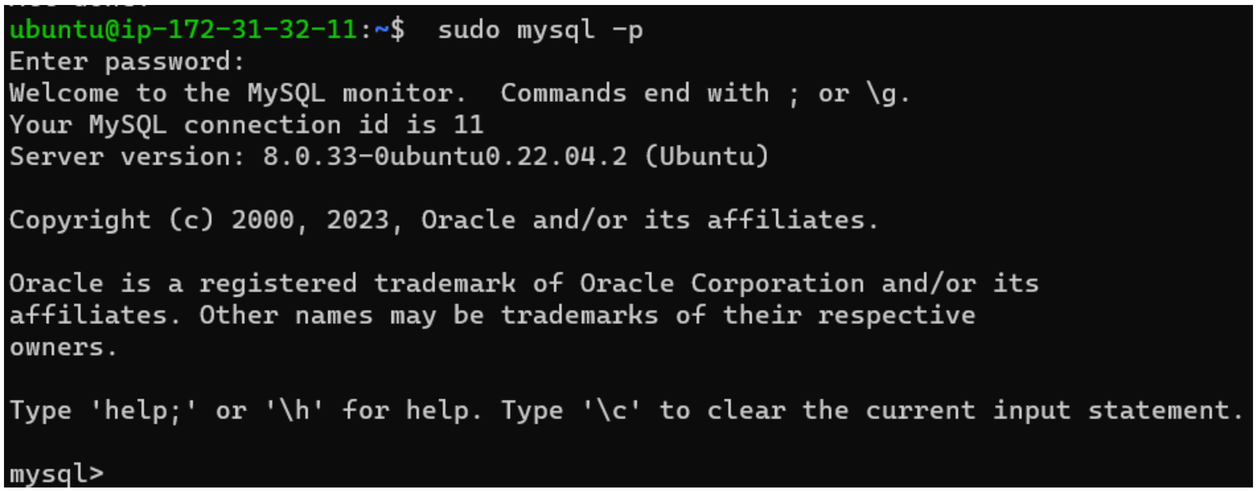
```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '[redacted]';
Query OK, 0 rows affected (0.01 sec)

mysql> exit
```

Now exit the MySQL environment and run the interactive command to validate our password component (type y and n where applicable).



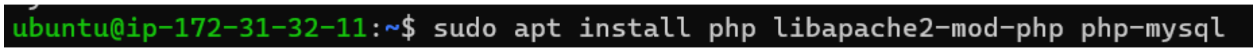
```
ubuntu@ip-172-31-32-11:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: n
Using existing password for root.
Change the password for root ? ((Press y|Y for Yes, any other key for No) : n

 ... skipping.
```

To validate that you can log in to the MySQL console type the below command and put in your password for the root user once prompted. This concludes the process for MySQL installation

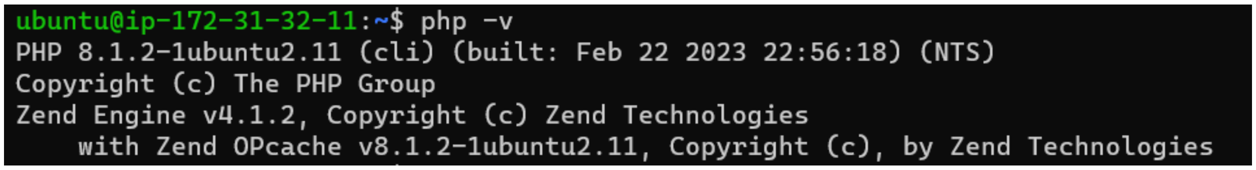

```
ubuntu@ip-172-31-32-11:~$  sudo mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.33-0ubuntu0.22.04.2 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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>
```

Installing PHP

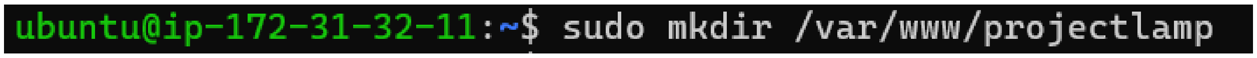
PHP is a commonly used open-source scripting language mainly used for web development**.** To install this, we will need to run all 3 packages at once.

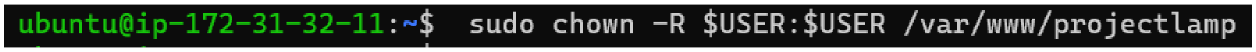

```
ubuntu@ip-172-31-32-11:~$ sudo apt install php libapache2-mod-php php-mysql
```

Once the installation is finished, you can run the following command to confirm your PHP version

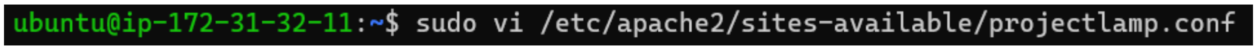

```
ubuntu@ip-172-31-32-11:~$ php -v
PHP 8.1.2-1ubuntu2.11 (cli) (built: Feb 22 2023 22:56:18) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.1.2, Copyright (c) Zend Technologies
    with Zend OPcache v8.1.2-1ubuntu2.11, Copyright (c), by Zend Technologies
```

Now it’s time to set up a domain, for this project, I will use projectlamp as the domain name. I will create a directory and then assign ownership to it with the following commands.

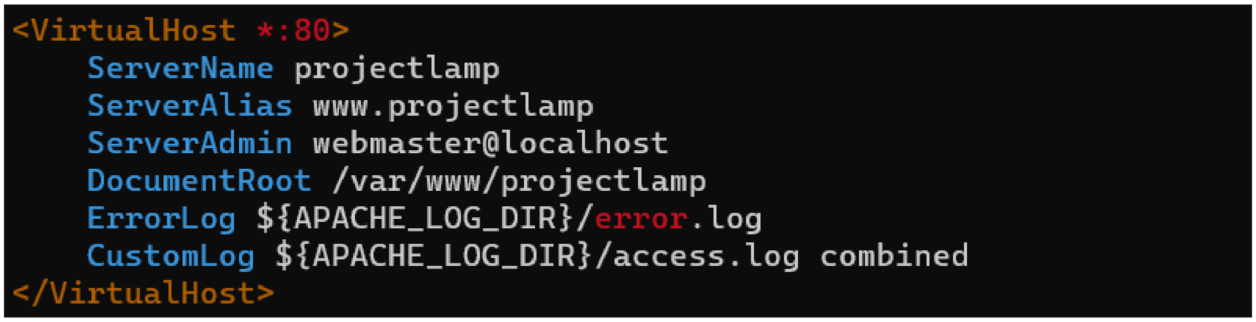

```
ubuntu@ip-172-31-32-11:~$ sudo mkdir /var/www/projectlamp
```


```
ubuntu@ip-172-31-32-11:~$  sudo chown -R $USER:$USER /var/www/projectlamp
```

I ran the ‘vi’ command to open a new configuration file in Apache’s sites-available directory.

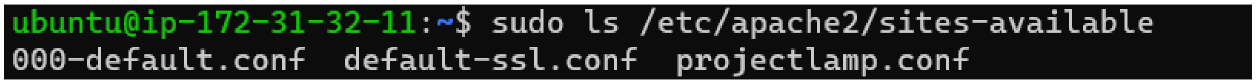

```
ubuntu@ip-172-31-32-11:~$ sudo vi /etc/apache2/sites-available/projectlamp.conf
```

Following that, I pasted in the following basic configuration and then saved the changes.



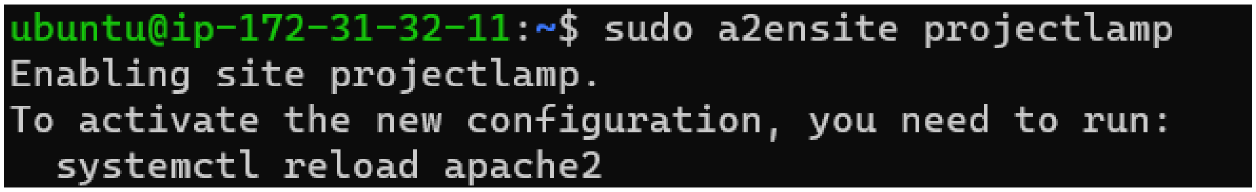
```
<VirtualHost *:80>
    ServerName projectlamp
    ServerAlias www.projectlamp
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/projectlamp
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

To confirm that the Projectlamp exists I ran



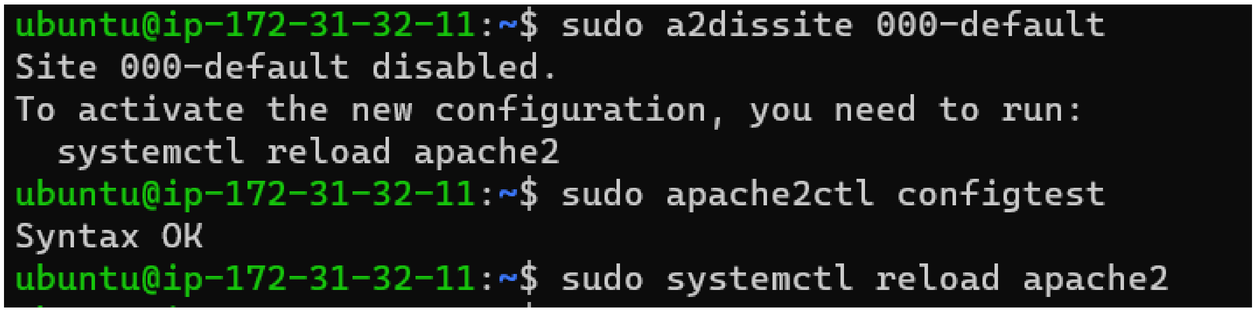
```
ubuntu@ip-172-31-32-11:~$ sudo ls /etc/apache2/sites-available
000-default.conf  default-ssl.conf  projectlamp.conf
```

To enable the new virtual host I used the a2ensite command



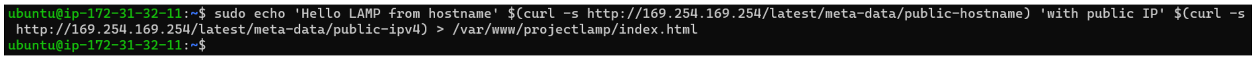
```
ubuntu@ip-172-31-32-11:~$ sudo a2ensite projectlamp
Enabling site projectlamp.
To activate the new configuration, you need to run:
  systemctl reload apache2
```

This in turn tells me the command to run to activate the config but before running the second command, it is best to disable the default website since I am not using a custom domain name, I ensured that the syntax is ok and then reloaded apache2.



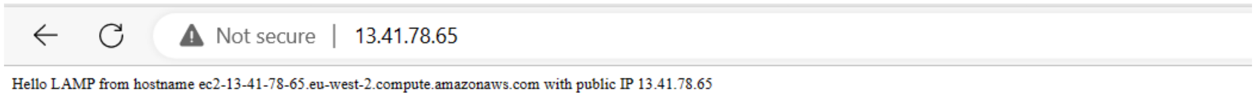
```
ubuntu@ip-172-31-32-11:~$ sudo a2dissite 000-default
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
ubuntu@ip-172-31-32-11:~$ sudo apache2ctl configtest
Syntax OK
ubuntu@ip-172-31-32-11:~$ sudo systemctl reload apache2
```

To create an index.html file to test that the virtual host works as expected run



```
ubuntu@ip-172-31-32-11:~$ sudo echo 'Hello LAMP from hostname' $(curl -s http://169.254.169.254/latest/meta-data/public-hostname) 'with public IP' $(curl -s
 http://169.254.169.254/latest/meta-data/public-ipv4) > /var/www/projectlamp/index.html
ubuntu@ip-172-31-32-11:~$
```

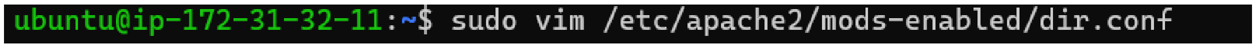
I then entered the IP address http://13.41.78.65/80 on my browser which displayed the text from the 'echo' command above indicating that my Apache virtual host is operating as intended.



Hello LAMP from hostname ec2-13-41-78-65.eu-west-2.compute.amazonaws.com with public IP 13.41.78.65

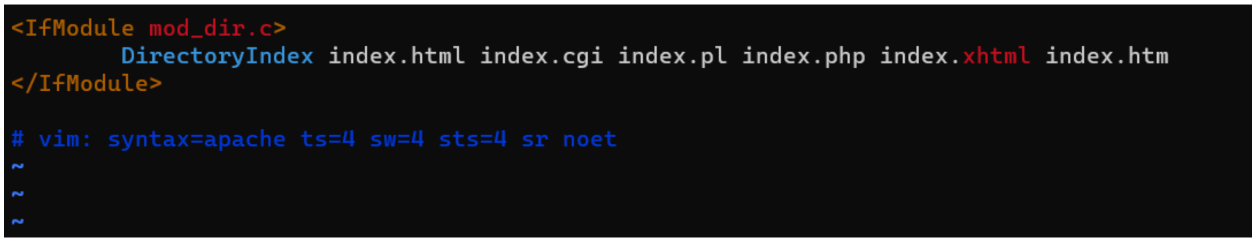
Enabling PHP on the Website

A file called index.html will always take priority over a file named index.php due to default DirectoryIndex settings on Apache. I modified its behaviour for this project's needs by using the Vim command to view and edit the file.



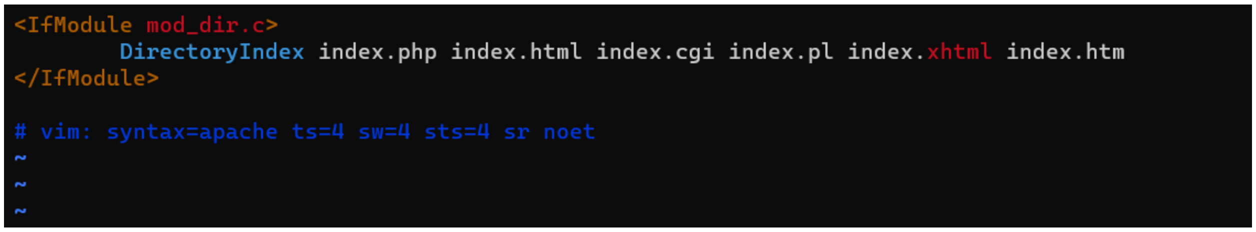
```
ubuntu@ip-172-31-32-11:~$ sudo vim /etc/apache2/mods-enabled/dir.conf
```

The display should appear as this.



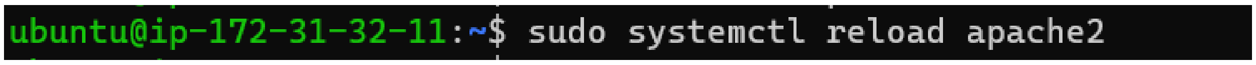
```
<IfModule mod_dir.c>
        DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
~
~
~
```

To allow index.php to take first place, I updated this to the following.



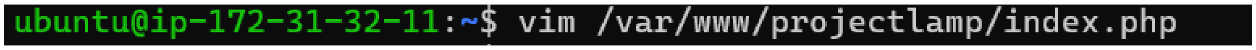
```
<IfModule mod_dir.c>
        DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
~
~
~
```

It is important to reload Apache after changes have been made and saved.



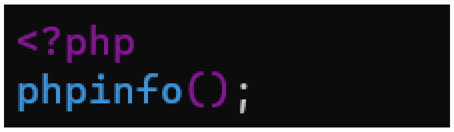
```
ubuntu@ip-172-31-32-11:~$ sudo systemctl reload apache2
```

To conclude the LAMP Stack Implementation, there is a need to create a PHP script to confirm that everything is correctly configured and running. I created an index.php file inside the custom web root folder.



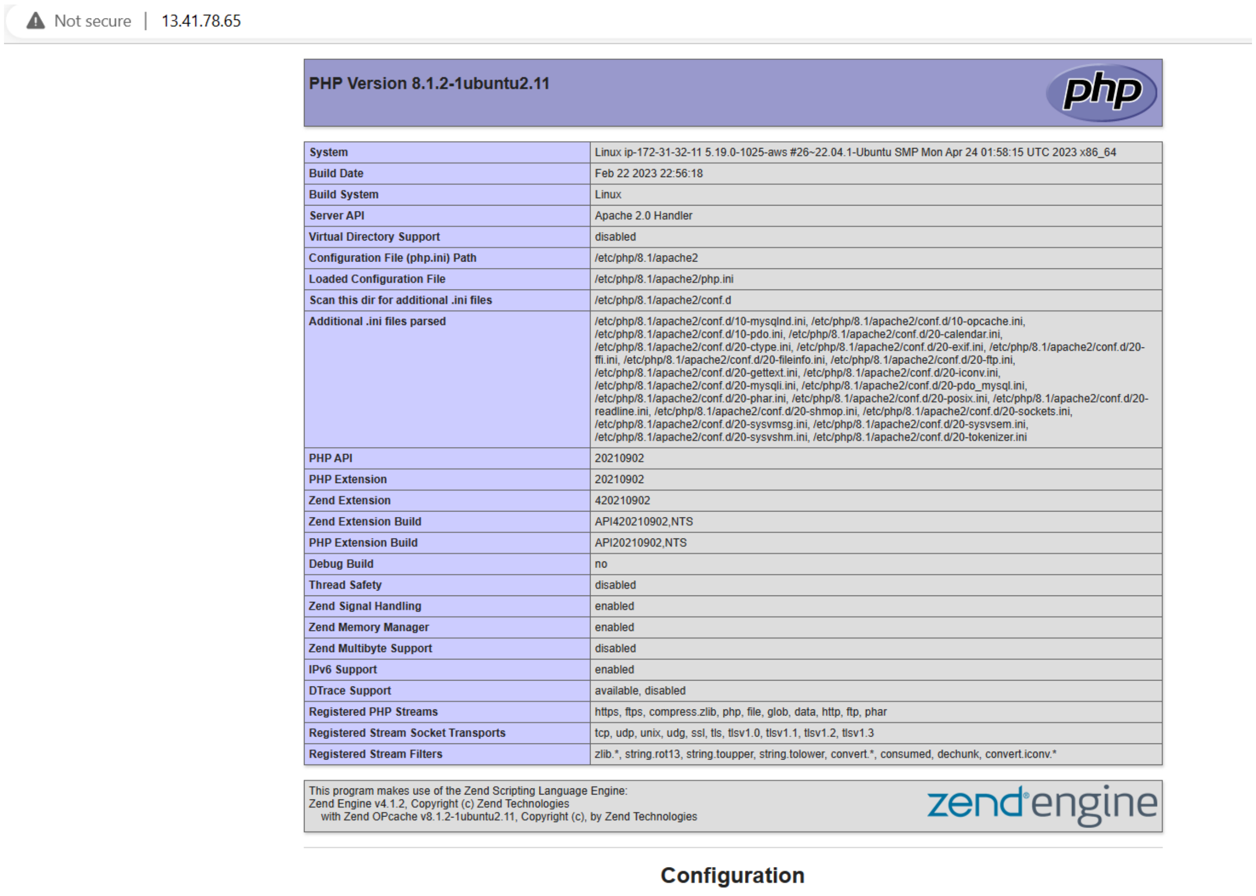
```
ubuntu@ip-172-31-32-11:~$ vim /var/www/projectlamp/index.php
```

I pasted the below PHP code into the file and saved the changes.



```php
<?php
phpinfo();
```

Once this has been completed, I browsed my webpage http://13.41.78.65/80 which then generated the below content.



Not secure | 13.41.78.65

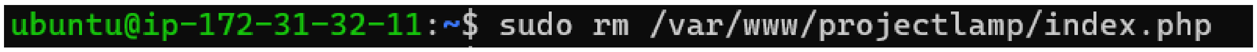
PHP Version 8.1.2-1ubuntu2.11

System	Linux ip-172-31-32-11 5.19.0-1025-aws #26~22.04.1-Ubuntu SMP Mon Apr 24 01:58:15 UTC 2023 x86_64
Build Date	Feb 22 2023 22:56:18
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/apache2
Loaded Configuration File	/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/10-mysqlnd.ini, /etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/conf.d/10-pdo.ini, /etc/php/8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20-ctype.ini, /etc/php/8.1/apache2/conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-ffi.ini, /etc/php/8.1/apache2/conf.d/20-fileinfo.ini, /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/20-gettext.ini, /etc/php/8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-mysqli.ini, /etc/php/8.1/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.1/apache2/conf.d/20-phar.ini, /etc/php/8.1/apache2/conf.d/20-posix.ini, /etc/php/8.1/apache2/conf.d/20-readline.ini, /etc/php/8.1/apache2/conf.d/20-shmop.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.1/apache2/conf.d/20-sysvsem.ini, /etc/php/8.1/apache2/conf.d/20-sysvshm.ini, /etc/php/8.1/apache2/conf.d/20-tokenizer.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902,NTS
PHP Extension Build	API20210902,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v4.1.2, Copyright (c) Zend Technologies
with Zend OPcache v8.1.2-1ubuntu2.11, Copyright (c), by Zend Technologies

Configuration

As a best practice, it is important I removed the index.php file I created as it contains sensitive data by running:



```
ubuntu@ip-172-31-32-11:~$ sudo rm /var/www/projectlamp/index.php
```

I have completed a LAMP Stack Implementation! 😀

REFERENCES

Dharmaratna, N.S. and Disanayake, C., 2021. A Mini Security Framework for LAMP Stack Deployments on the Cloud-Research Proposal. *Methodology*, *1*, p.8.

Karanjit, Arpana, "MEAN vs. LAMP Stack" (2016). Culminating Projects in Computer Science and Information Technology. 11. Available at: https://repository.stcloudstate.edu/cgi/viewcontent.cgi?article=1039&amp;context=csit_etds (Accessed: 30 May 2023).

Ludwig, L. (2022) *Lamp Stack explained – what it stands for & how it's used*, *Larry Ludwig*. Available at: https://larryludwig.com/lamp-stack/ (Accessed: 30 May 2023).