# Hosting a WordPress Website on Amazon EC2

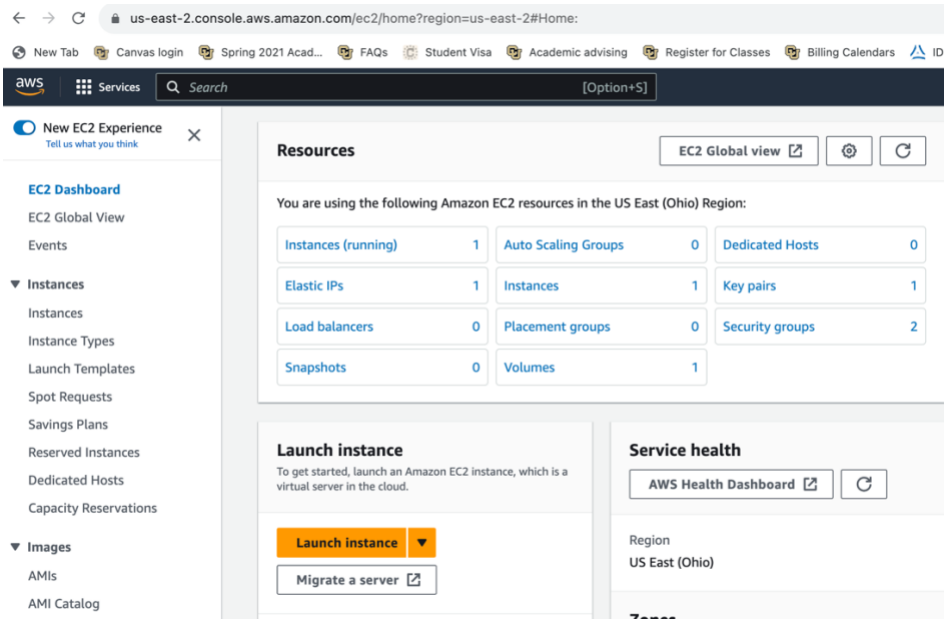
This document explains how to use Amazon EC2 to host a WordPress and connect it to a custom domain using Apache webserver and MySQL database.

**Following are the Steps to host a wordpress website on Amazon EC2 with custom domain:**

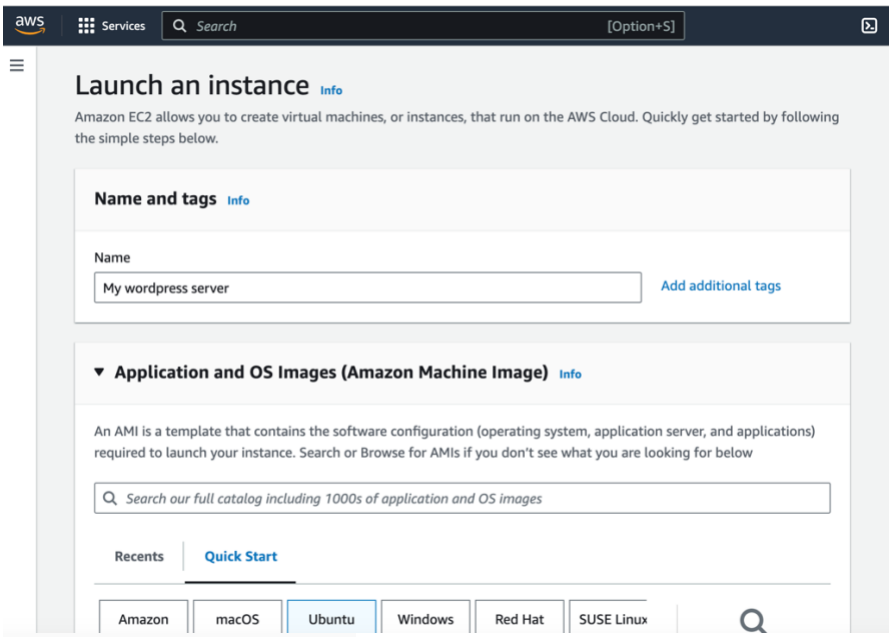
Step 1: Setup Amazon EC2 instance
Step 2: Install apache webserver
Step 3: Install MySQL database
Step 4: Download and install wordpress
Step 5: Link a custom domain to a website

## Step1: Setup Amazon EC2 instance

Open Amazon EC2 console

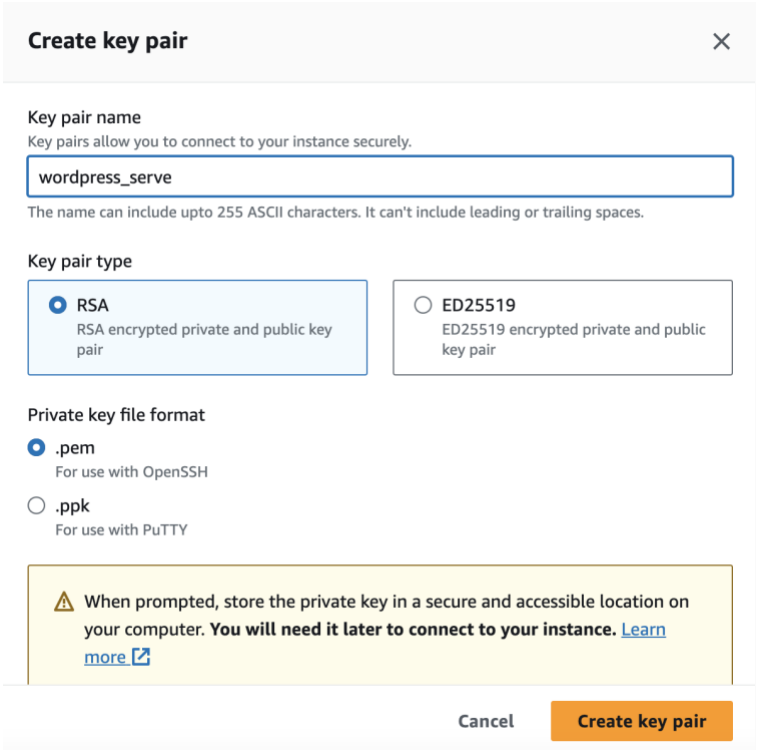



Click on “Launch instance” and enter an instance name
Name : My wordpress server
Application and OS Images (Amazon Machine Image) : Ubuntu




Instance type : t2.micro

Key pair: wordpress_server , others as default click on “create key pair”



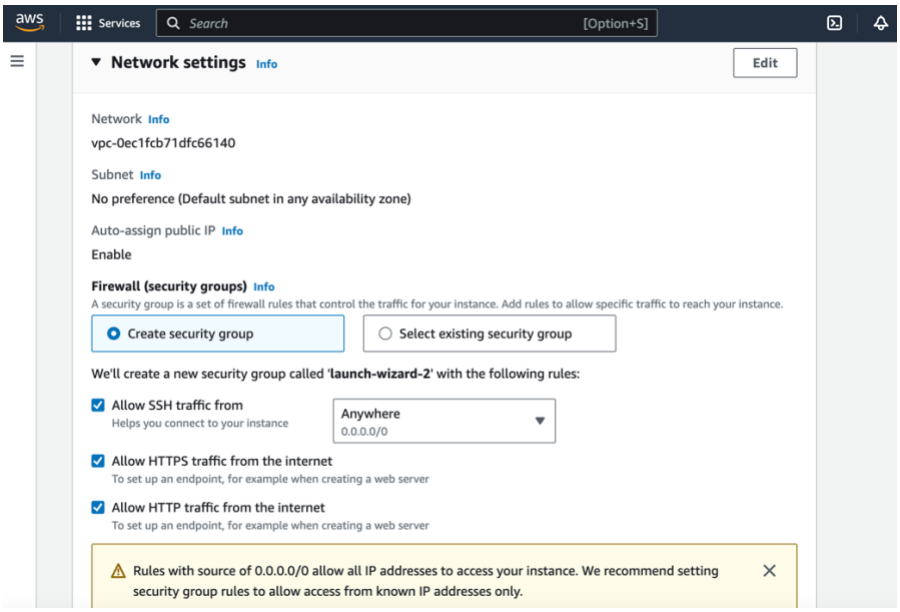

This downloads the wordpress_server.pem file (Secure file)

Under network settings, enable all the three traffic

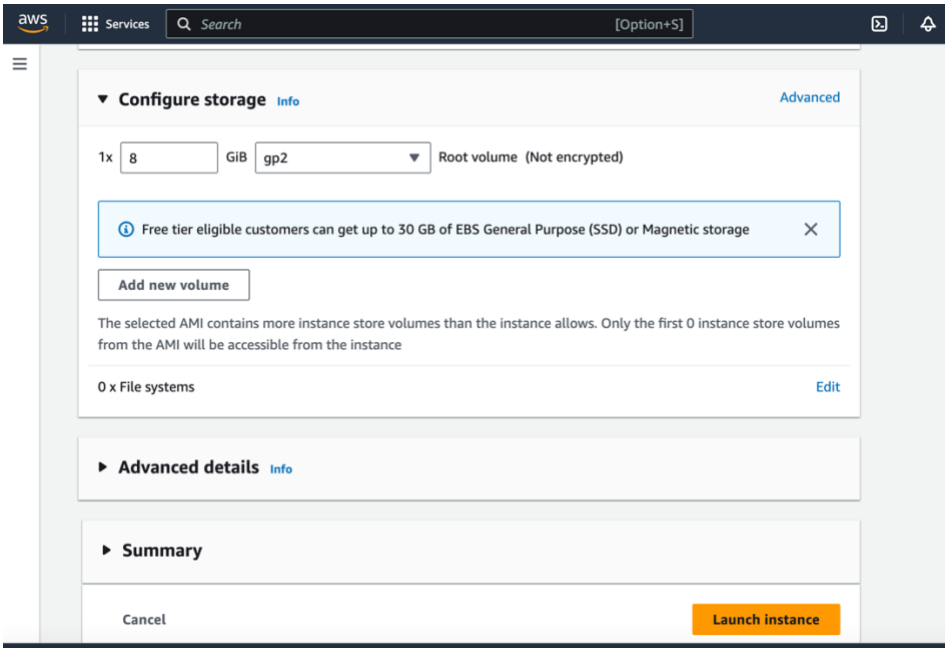
Allow SSh traffic from

Allow HTTPs traffic from the internet

Allow HTTP traffic from the internet

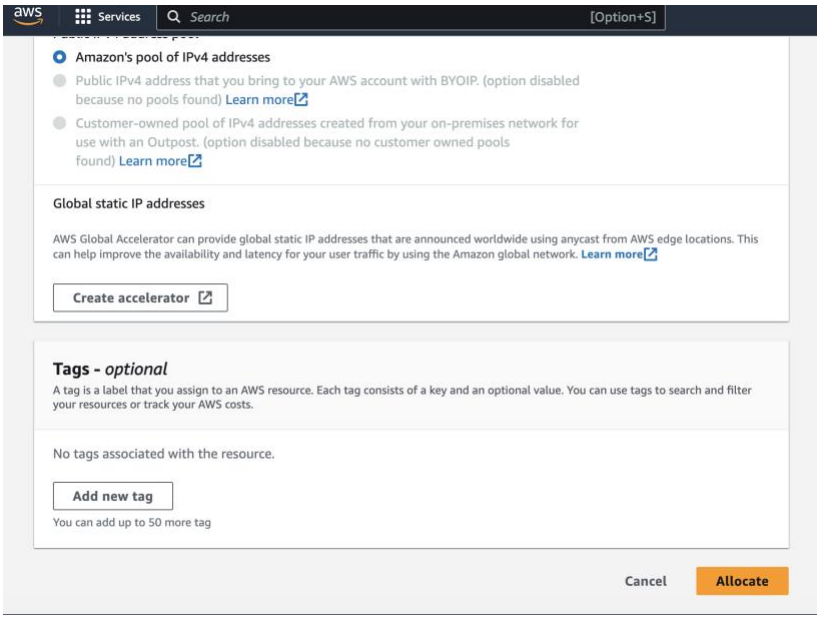



Others as default and finally click on “Launch instance”

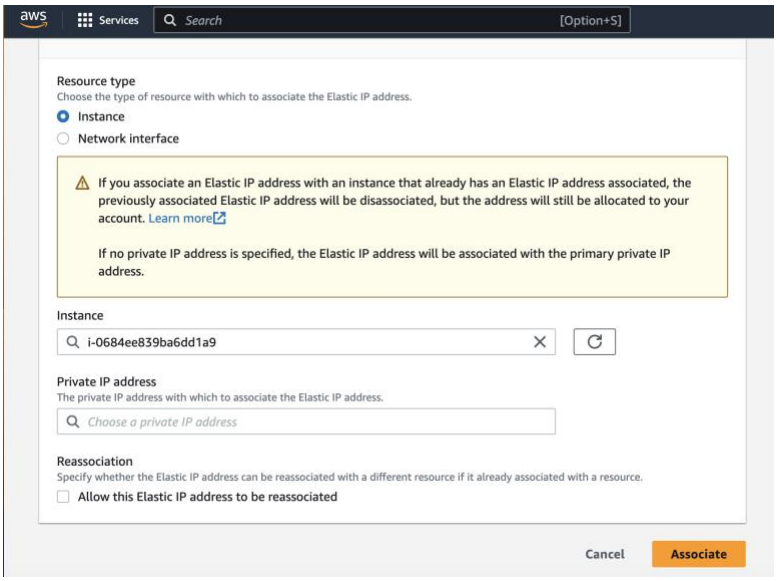



Once the instance is created, now assign the Elastic Ip address in order to have constant public IPV4 address.

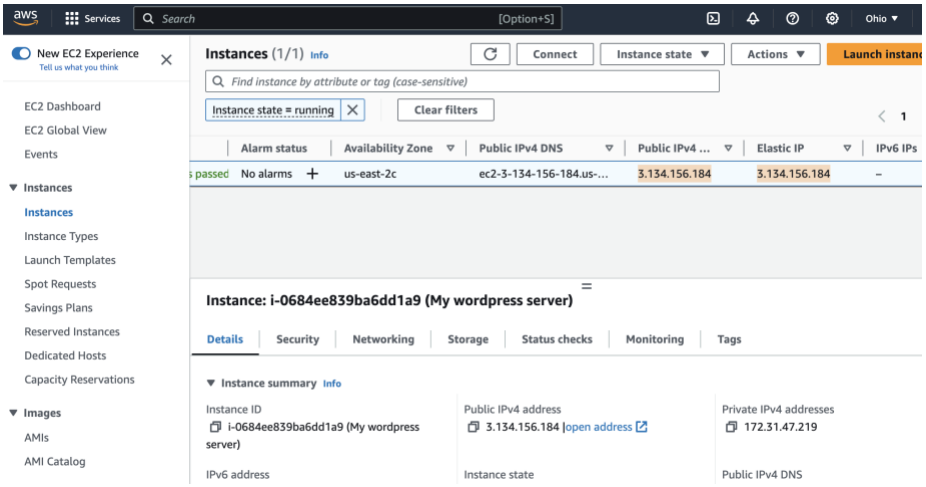
Goto EC2 dashboard, click on “Elastic IPs”, click on “Allocate Elastic IP address”, and click on “Allocate”. Now created Elastic IP address.




Select the Elastic IP address created and Goto Actions and select the Associate Elastic IP address. Here choose an EC2 instance and click on “Associate”.




Now you can see public IPV4 address is changed and is same as Elastic IP.



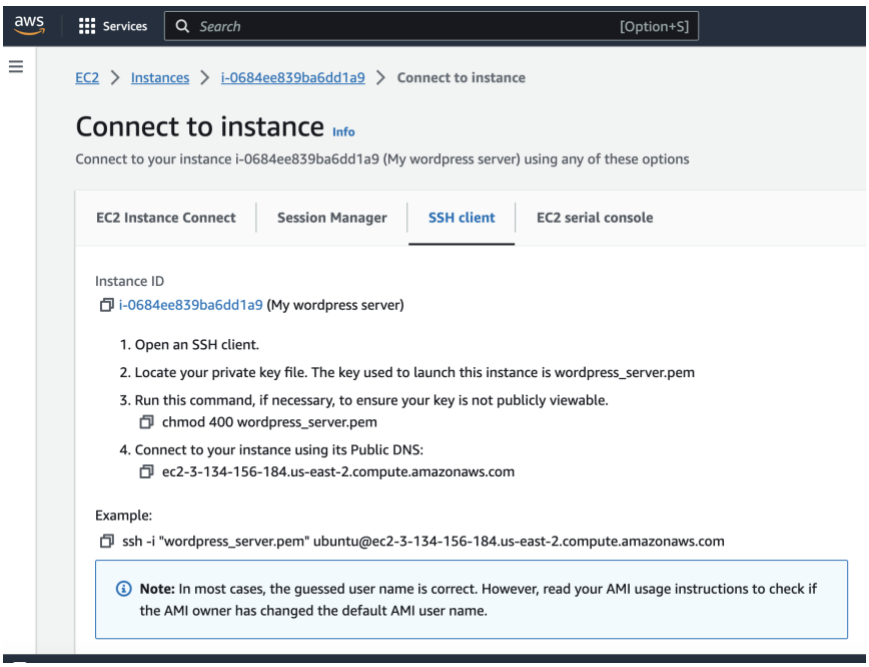

Open the SSH client (terminal)

Locate your private key file. The key used to launch this instance is wordpress_server.pem

Run this command, if necessary, to ensure your key is not publicly viewable.

chmod 400 wordpress_server.pem

Right click on the instance created, click on “connect”.
Under SSH client section, there is command to connect through SSH.



Connect to your instance in SSH client

ssh -i "wordpress_server.pem" ubuntu@ec2-3-134-156-184.us-east-2.compute.amazonaws.com

```
mallika:Downloads sharathkrishna$ ssh -i "wordpress_server.pem" ubuntu@ec2-3-134-156-184.us-east-2.compute.amazonaws.com
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)

  * Documentation:  https://help.ubuntu.com
  * Management:     https://landscape.canonical.com
  * Support:        https://ubuntu.com/advantage

   System information as of Mon Oct  2 18:27:48 UTC 2023

   System load:  0.0               Processes:             107
   Usage of /:   38.9% of 7.57GB   Users logged in:       0
   Memory usage: 71%               IPv4 address for eth0: 172.31.47.219
   Swap usage:   0%

  * Ubuntu Pro delivers the most comprehensive open source security and
    compliance features.

    https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

63 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


*** System restart required ***
Last login: Thu Sep 28 12:40:02 2023 from 76.25.165.0
ubuntu@ip-172-31-47-219:~$
```

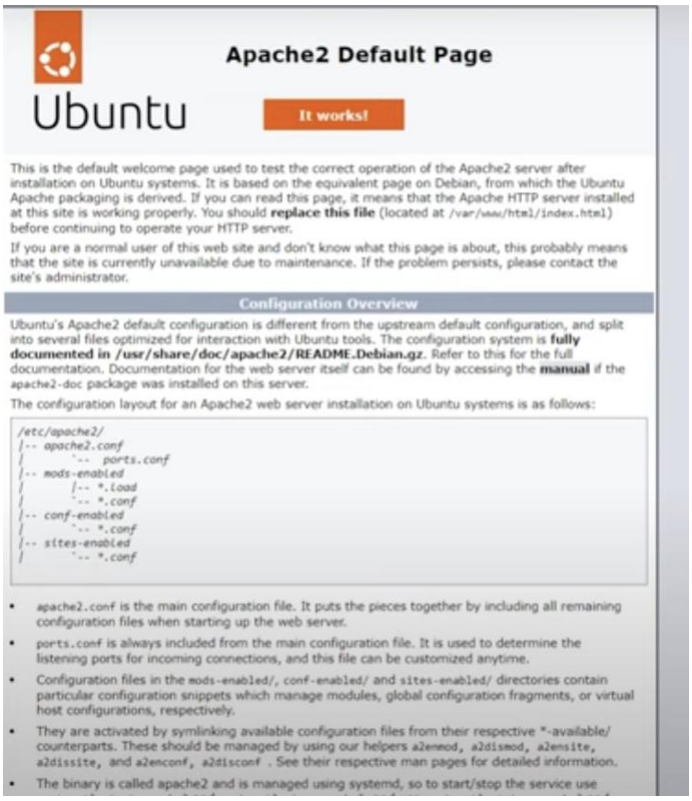
Now you are connected to your instance.

## Step2: Install Apache Webserver

sudo apt install apache2

```
ubuntu@ip-172-31-47-219:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2
0 upgraded, 1 newly installed, 0 to remove and 65 not upgraded.
Need to get 97.8 kB of archives.
After this operation, 546 kB of additional disk space will be used.
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2 amd64 2.4.52-1ubuntu4.6 [97.8 kB]
Fetched 97.8 kB in 0s (5035 kB/s)
Selecting previously unselected package apache2.
(Reading database ... 95546 files and directories currently installed.)
Preparing to unpack .../apache2_2.4.52-1ubuntu4.6_amd64.deb ...
Unpacking apache2 (2.4.52-1ubuntu4.6) ...
Setting up apache2 (2.4.52-1ubuntu4.6) ...
apache-htcacheclean.service is a disabled or a static unit not running, not starting it.
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for ufw (0.36.1-4build1) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Restarting services...
 /etc/needrestart/restart.d/systemd-manager
 systemctl restart chrony.service multipathd.service mysql.service packagekit.service polkit.service rsyslog.service ssh.service systemd-journald.service systemd-networkd.service systemd-resolved.service
systemd-udevd.service
Service restarts being deferred:
 /etc/needrestart/restart.d/dbus.service
 systemctl restart networkd-dispatcher.service
 systemctl restart systemd-logind.service
 systemctl restart unattended-upgrades.service
 systemctl restart user@1000.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-47-219:~$
```

Cross Verify if the webserver is working, In the browser open the public IPV4 address and check the apache2 default page.



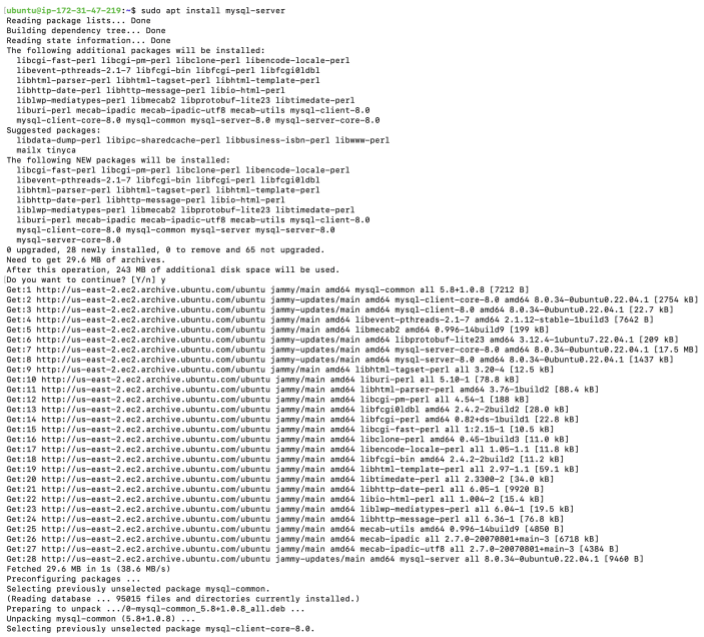

Now install php runtime on our instance and MySQL connector for php.

sudo apt install php libapache2-mod-php php-mysql

```
ubuntu@ip-172-31-47-219:~$ sudo apt install php libapache2-mod-php php-mysql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libapache2-mod-php8.1 php-common php8.1 php8.1-cli php8.1-common
  php8.1-mysql php8.1-opcache php8.1-readline
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libapache2-mod-php libapache2-mod-php8.1 php php-common php-mysql php8.1
  php8.1-cli php8.1-common php8.1-mysql php8.1-opcache php8.1-readline
0 upgraded, 11 newly installed, 0 to remove and 65 not upgraded.
Need to get 5265 kB of archives.
After this operation, 21.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 php-common all 2:92ubuntu1 [12.4 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-common amd64 8.1.2-1ubuntu2.14 [1127 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-opcache amd64 8.1.2-1ubuntu2.14 [365 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-readline amd64 8.1.2-1ubuntu2.14 [13.6 kB]
Get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-cli amd64 8.1.2-1ubuntu2.14 [1834 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libapache2-mod-php8.1 amd64 8.1.2-1ubuntu2.14 [1766 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libapache2-mod-php all 2:8.1+92ubuntu1 [2898 B]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1 all 8.1.2-1ubuntu2.14 [9158 B]
Get:9 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 php all 2:8.1+92ubuntu1 [2756 B]
Get:10 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-mysql amd64 8.1.2-1ubuntu2.14 [130 kB]
Get:11 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 php-mysql all 2:8.1+92ubuntu1 [1834 B]
Fetched 5265 kB in 0s (41.1 MB/s)
Selecting previously unselected package php-common.
(Reading database ... 94876 files and directories currently installed.)
Preparing to unpack .../00-php-common_2%3a92ubuntu1_all.deb ...
Unpacking php-common (2:92ubuntu1) ...
Selecting previously unselected package php8.1-common.
Preparing to unpack .../01-php8.1-common_8.1.2-1ubuntu2.14_amd64.deb ...
Unpacking php8.1-common (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package php8.1-opcache.
Preparing to unpack .../02-php8.1-opcache_8.1.2-1ubuntu2.14_amd64.deb ...
Unpacking php8.1-opcache (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package php8.1-readline.
Preparing to unpack .../03-php8.1-readline_8.1.2-1ubuntu2.14_amd64.deb ...
Unpacking php8.1-readline (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package php8.1-cli.
Preparing to unpack .../04-php8.1-cli_8.1.2-1ubuntu2.14_amd64.deb ...
Unpacking php8.1-cli (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package libapache2-mod-php8.1.
Preparing to unpack .../05-libapache2-mod-php8.1_8.1.2-1ubuntu2.14_amd64.deb ...
Unpacking libapache2-mod-php8.1 (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package libapache2-mod-php.
Preparing to unpack .../06-libapache2-mod-php_2%3a8.1+92ubuntu1_all.deb ...
Unpacking libapache2-mod-php (2:8.1+92ubuntu1) ...
Selecting previously unselected package php8.1.
Preparing to unpack .../07-php8.1_8.1.2-1ubuntu2.14_all.deb ...
Unpacking php8.1 (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package php.
Preparing to unpack .../08-php_2%3a8.1+92ubuntu1_all.deb ...
Unpacking php (2:8.1+92ubuntu1) ...
Selecting previously unselected package php8.1-mysql.
Preparing to unpack .../09-php8.1-mysql_8.1.2-1ubuntu2.14_amd64.deb ...
Unpacking php8.1-mysql (8.1.2-1ubuntu2.14) ...
Selecting previously unselected package php-mysql.
Preparing to unpack .../10-php-mysql_2%3a8.1+92ubuntu1_all.deb ...
Unpacking php-mysql (2:8.1+92ubuntu1) ...
Setting up php-common (2:92ubuntu1) ...
```

## Step3: Install MySQL database

```
sudo apt install mysql-server
```




Configuration to mysql server.

Change the mysql authentication plugin to mysql_native_password. So that we can login to mysql-server with normal password.

First login to MySQL Server

```
sudo mysql -u root
```

Change the authentication plugin

```
ALTER USER ‘root’@’localhost’ IDENTIFIED WITH mysql_native_password by ‘TestPassword@123’;
```

Create a new user other than root account in order to use the wordpress.

```
CREATE USER ‘wp_user’@localhost IDENTIFIED BY ‘Testpassword@123’;
```

Create a new database for wordpress

CREATE DATABASE wp;

Grant all the privileges to the database ‘wp’ to the ‘wp_user’.

GRANT ALL PRIVILEGES ON wp.* TO ‘wp_user’@localhost;

We configured the mysql-server

```
ubuntu@ip-172-31-47-219:~$ sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.34-0ubuntu0.22.04.1 (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> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'Testpassword@123';
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE USER 'wp_user'@localhost IDENTIFIED BY 'Testpassword@123';
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE DATABASE wp;
Query OK, 1 row affected (0.01 sec)

mysql> GRANT ALL PRIVILEGES ON wp.* TO 'wp_user'@localhost;
Query OK, 0 rows affected (0.01 sec)

mysql> ^DBye
ubuntu@ip-172-31-47-219:~$
```

**Step 4: Download and install wordpress.**

Goto /tmp directory

Download the latest.tar.gz

cd /tmp

wget https://wordpress.org/latest.tar.gz

unzip

tar -xvf latest.tar.gz

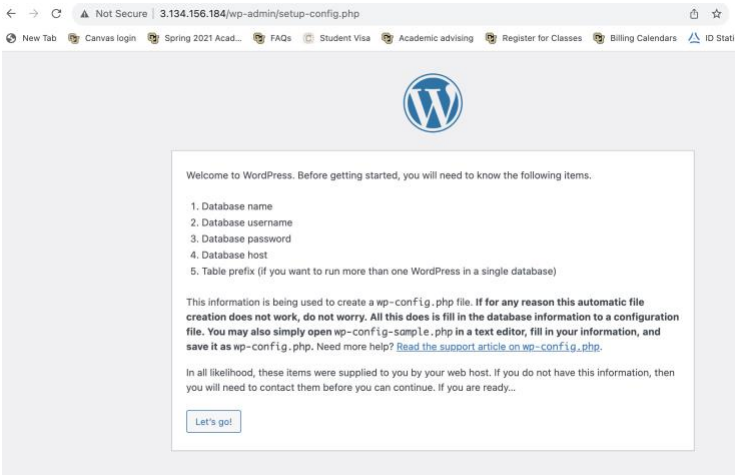
Move the wordpress folder to the apache document root

sudo mv wordpress/ /var/www/html

```
ubuntu@ip-172-31-47-219:/$ cd /tmp/
ubuntu@ip-172-31-47-219:/tmp$ wget https://wordpress.org/latest.tar.gz
--2023-10-04 05:39:26--  https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 23447259 (22M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz       100%[===================>]  22.36M  48.3MB/s    in 0.5s

2023-10-04 05:39:26 (48.3 MB/s) - 'latest.tar.gz' saved [23447259/23447259]

ubuntu@ip-172-31-47-219:/tmp$ ls
latest.tar.gz
snap-private-tmp
systemd-private-7aa8dd189256437692b9f4dd30cae343-apache2.service-5jWFuc
systemd-private-7aa8dd189256437692b9f4dd30cae343-chrony.service-nG6KiU
systemd-private-7aa8dd189256437692b9f4dd30cae343-systemd-logind.service-AtFOWu
systemd-private-7aa8dd189256437692b9f4dd30cae343-systemd-resolved.service-fJyjEu
tmp.oaaAQbP4Qk
ubuntu@ip-172-31-47-219:/tmp$ tar -xvf latest.tar.gz
wordpress/
wordpress/xmlrpc.php
wordpress/wp-blog-header.php
wordpress/readme.html
wordpress/wp-signup.php
wordpress/index.php
wordpress/wp-cron.php
wordpress/wp-config-sample.php
wordpress/wp-login.php
wordpress/wp-settings.php
wordpress/license.txt
wordpress/wp-content/
wordpress/wp-content/themes/
wordpress/wp-content/themes/twentytwentythree/
wordpress/wp-content/themes/twentytwentythree/theme.json
wordpress/wp-content/themes/twentytwentythree/parts/
wordpress/wp-content/themes/twentytwentythree/parts/footer.html
wordpress/wp-content/themes/twentytwentythree/parts/comments.html
wordpress/wp-content/themes/twentytwentythree/parts/header.html
wordpress/wp-content/themes/twentytwentythree/parts/post-meta.html
wordpress/wp-content/themes/twentytwentythree/patterns/
wordpress/wp-content/themes/twentytwentythree/patterns/hidden-404.php
wordpress/wp-content/themes/twentytwentythree/patterns/post-meta.php
wordpress/wp-content/themes/twentytwentythree/patterns/hidden-no-results.php
wordpress/wp-content/themes/twentytwentythree/patterns/call-to-action.php
wordpress/wp-content/themes/twentytwentythree/patterns/footer-default.php
```

```
ubuntu@ip-172-31-47-219:/tmp$ sudo mv wordpress/ /var/www/html/
ubuntu@ip-172-31-47-219:/tmp$ ls
latest.tar.gz
snap-private-tmp
systemd-private-7aa8dd189256437692b9f4dd30cae343-apache2.service-5jWFuc
systemd-private-7aa8dd189256437692b9f4dd30cae343-chrony.service-nG6KiU
systemd-private-7aa8dd189256437692b9f4dd30cae343-systemd-logind.service-AtFOWu
systemd-private-7aa8dd189256437692b9f4dd30cae343-systemd-resolved.service-fJyjEu
tmp.oaaAQbP4Qk
ubuntu@ip-172-31-47-219:/tmp$ cd /var/www/html/
ubuntu@ip-172-31-47-219:/var/www/html$ ls
index.html  wordpress
```

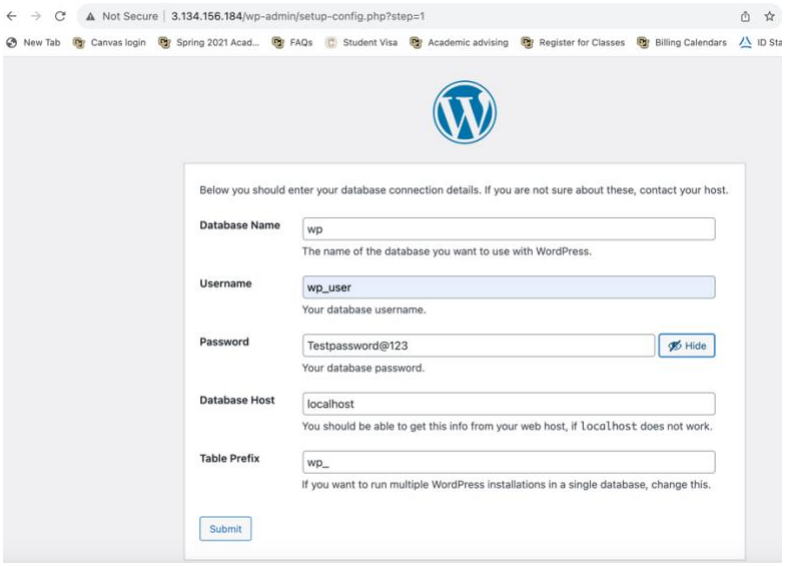
When you open the public IPV4 address in the browser(http://3.134.156.184/wordpress), then the wordpress installation page is displayed.



Install the wordpress, click on “Let’s go!”

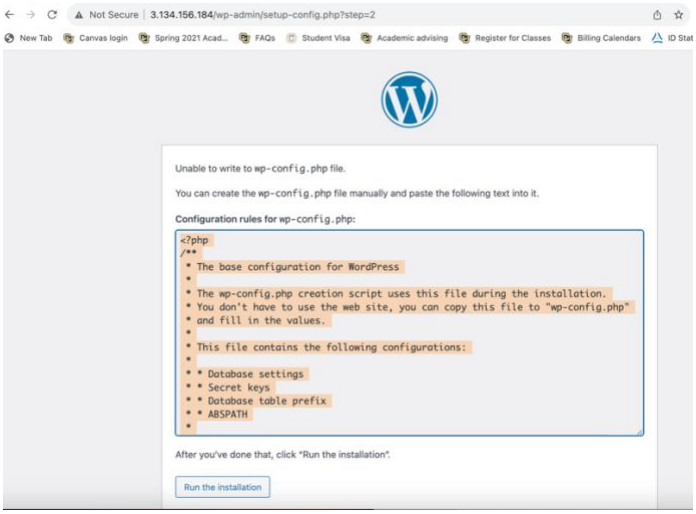
Next, configure the database.

Database Name: wp
Username: wp_user
Password: Testpassword@123
Database Host: localhost
Table Prefix: wp_
Click on “submit”




There is an error “Unable to write wp-config.php file”

So, copy the php code



Goto terminal, Goto /var/www/html/wordpress/

```
cd /var/www/html/wordpress/
```

In the path, create a new file wp-config.php

```
nano wp-config.php
```

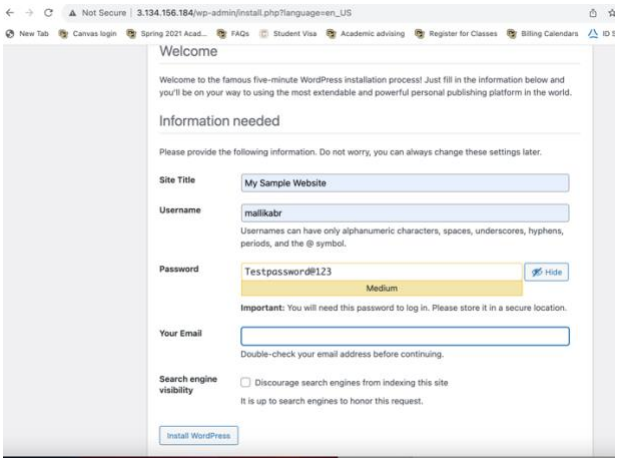
paste the php code.

```
ubuntu@ip-172-31-47-219:/$ cd /var/www/html/wordpress/
ubuntu@ip-172-31-47-219:/var/www/html/wordpress$ nano wp-config.php
ubuntu@ip-172-31-47-219:/var/www/html/wordpress$
```

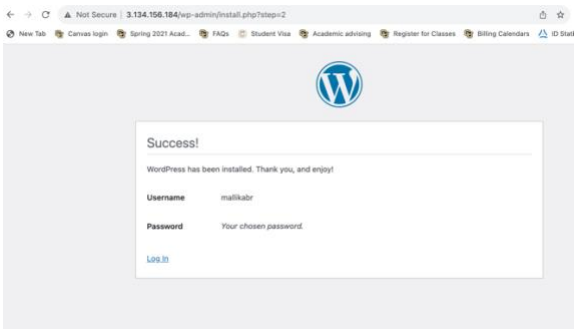
Click on “Run the installation” in the browser.

Fill the information needed

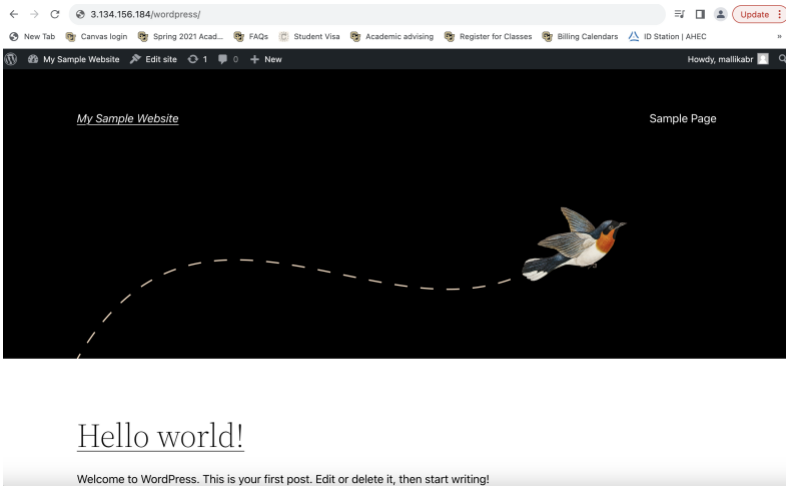
Site Title: my sample website
Username: mallikabr
Password: Testpassword@123
Your email
Click on “Install WordPress”




WordPress is now installed.




Open http://3.134.156.184/wordpress/




We don't have to go to the sub path i.e., /wordpress (http://3.134.156.184/wordpress/)

The wordpress website has to be served on the root directory (http://3.134.156.184)

For that goto terminal, goto /etc/apache2/sites-available/

```
cd /etc/apache2/sites-available/
```

Then open the 000-default.conf (configuration file)

```
sudo nano 000-default.conf
```

change the DocumentRoot from /var/www/html to /var/www/html/wordpress

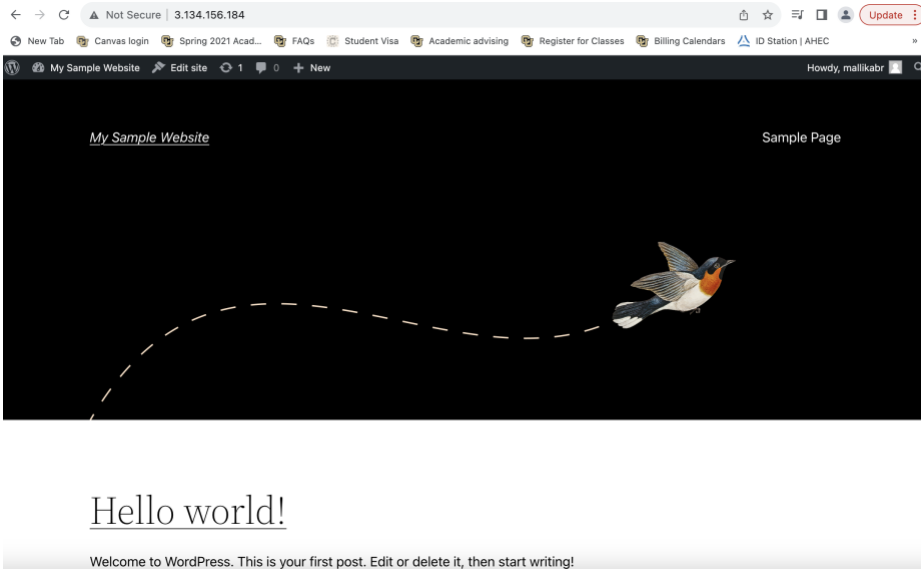
```
 GNU nano 6.2                    000-default.conf
VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port t>
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/wordpress

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn
```

To restart the apache2

sudo systemctl restart apache2

```
ubuntu@ip-172-31-47-219:/var/www/html/wordpress$ cd /etc/apache2/sites-available/
ubuntu@ip-172-31-47-219:/etc/apache2/sites-available$ ls
ubuntu@ip-172-31-47-219:/etc/apache2/sites-available$ sudo systemctl restart apache2
ubuntu@ip-172-31-47-219:/etc/apache2/sites-available$
```

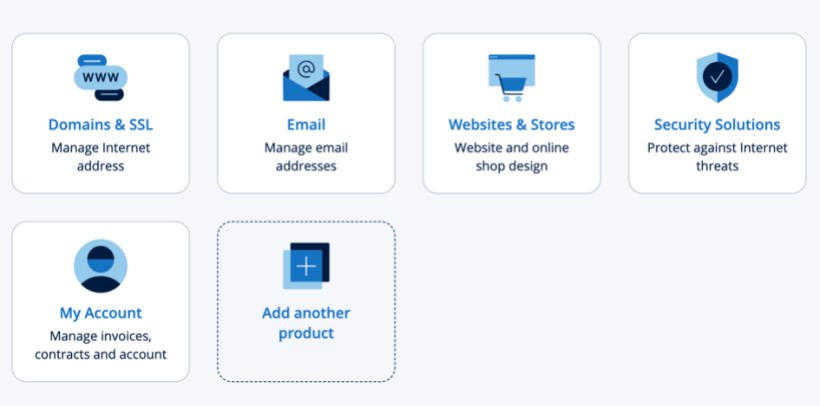
This worked!!



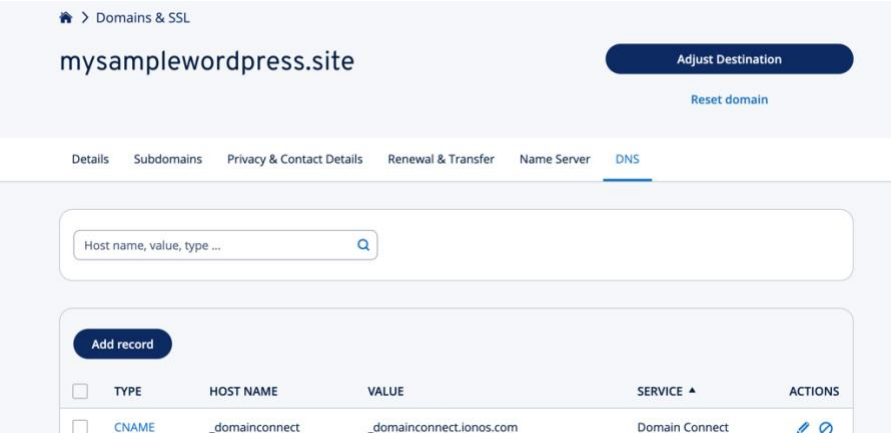
## Step5: Link a custom domain to a website

My domain Name: mysamplewordpress.site from IONOS. Now link this domain to the wordpress.

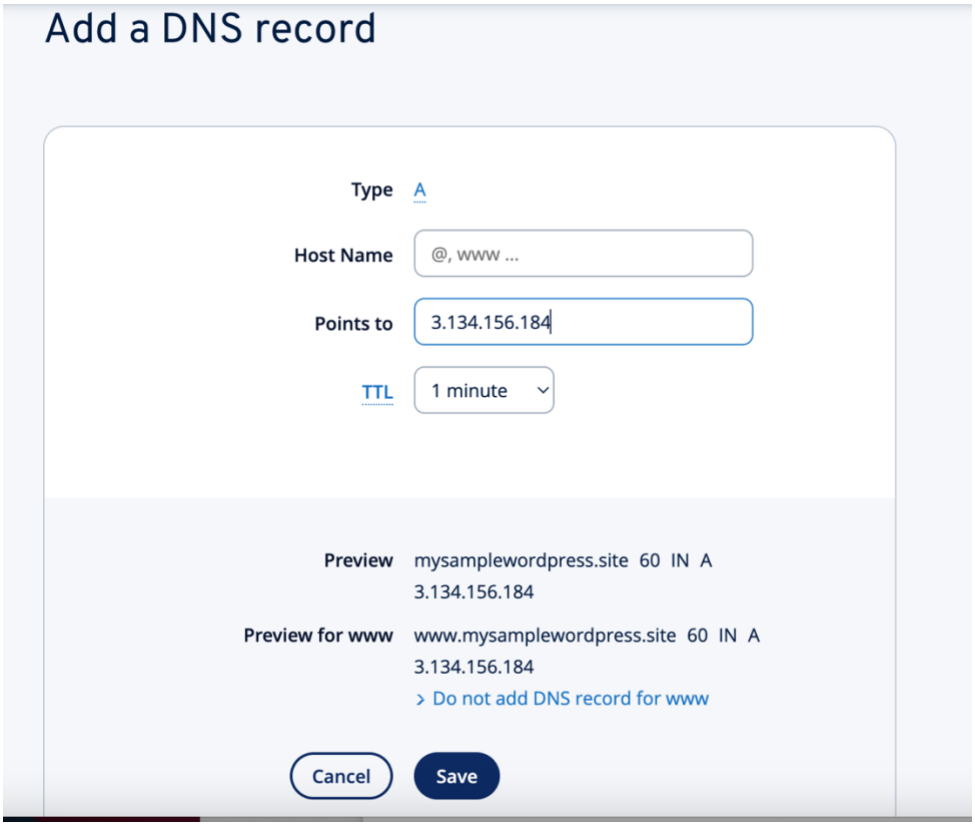
Login to IONOS portal and click on “Domains & SSL” in home page.




Click on “Adjust DNS Settings”, then click on “Add record”.


Choose A type and points to public IPV4 address i.e., 3.134.156.184 and TTL set to 60 and click on “save”.




And there are some additional changes that needs to be modified in the configuration file

cd /etc/apache2/sites-available

sudo nano 000-default.conf

add the below two lines in the .conf file

ServerName mysamplewordpress.site

ServerAlias www.mysamplewordpress.site

```
GNU nano 6.2                    000-default.conf
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/wordpress

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ServerName mysamplewordpress.site
        ServerAlias www.mysamplewordpress.site

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
```

Restart the apache2

sudo systemctl restart apache2

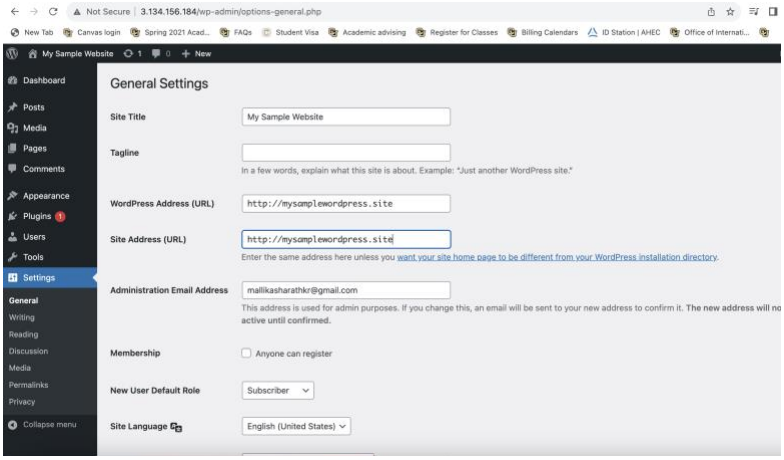
Finally, domain name is pointing to the AWS instance. Now we can access the wordpress using the domain name.

Visit the website.

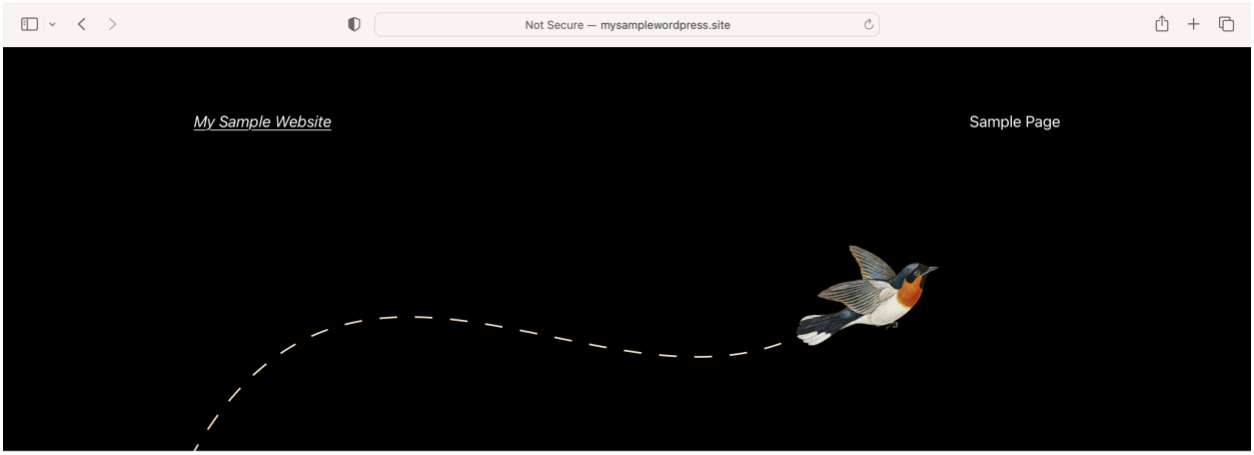
www.mysamplewordpress.site

Now, when you visit www.mysamplewordpress.site/wp-admin, it is redirecting the public IPV4 address. To fix this,

Login to the wordpress, In the Dashboard, Goto Settings, then click on General.
Modify the Wordpress Address (URL): http://mysamplewordpress.site
Site Address(URL): http://mysamplewordpress.site
Click on “Save changes”




Now, the website points to domain name.




# Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

October 4, 2023

Here the website is not secure. It has to work over https.
We need to install certbot.

sudo apt install certbot python3-certbot-apache

```
ubuntu@ip-172-31-47-219:~$ cd /etc/apache2/sites-available/
ubuntu@ip-172-31-47-219:/etc/apache2/sites-available$ sudo apt install certbot python3-certbot-apache
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  augeas-lenses libaugeas0 python3-acme python3-augeas python3-certbot
  python3-configargparse python3-icu python3-josepy python3-parsedatetime
  python3-requests-toolbelt python3-rfc3339 python3-zope.component
  python3-zope.event python3-zope.hookable
Suggested packages:
  augeas-doc python-certbot-doc python3-certbot-nginx augeas-tools
  python-acme-doc python-certbot-apache-doc
The following NEW packages will be installed:
  augeas-lenses certbot libaugeas0 python3-acme python3-augeas python3-certbot
  python3-certbot-apache python3-configargparse python3-icu python3-josepy
  python3-parsedatetime python3-requests-toolbelt python3-rfc3339
  python3-zope.component python3-zope.event python3-zope.hookable
0 upgraded, 16 newly installed, 0 to remove and 60 not upgraded.
Need to get 1552 kB of archives.
After this operation, 7681 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 augeas-lenses all 1.13.0-1 [321 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 libaugeas0 amd64 1.13.0-1 [200 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 python3-josepy all 1.10.0-1 [22.0 kB]
```

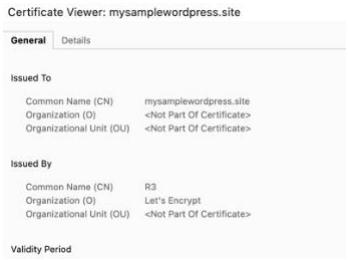
Run the command to secure the apache with SSL certificate
sudo certbot –apache

prompts for an email address and enter

```
ubuntu@ip-172-31-47-219:/etc/apache2/sites-available$ sudo certbot --apache
Saving debug log to /var/log/letsencrypt/letsencrypt.log

Which names would you like to activate HTTPS for?
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1: mysamplewordpress.site
2: www.mysamplewordpress.site
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Select the appropriate numbers separated by commas and/or spaces, or leave input
blank to select all options shown (Enter 'c' to cancel):
Requesting a certificate for mysamplewordpress.site and www.mysamplewordpress.site

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/mysamplewordpress.site/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/mysamplewordpress.site/privkey.pem
This certificate expires on 2024-01-03.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for mysamplewordpress.site to /etc/apache2/sites-available/000-default-le-ssl.conf
Successfully deployed certificate for www.mysamplewordpress.site to /etc/apache2/sites-available/000-default-le-ssl.conf
Congratulations! You have successfully enabled HTTPS on https://mysamplewordpress.site and https://www.mysamplewordpress.site

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
 * Donating to EFF:                    https://eff.org/donate-le
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
ubuntu@ip-172-31-47-219:/etc/apache2/sites-available$
```

Successfully deployed certificate for both mysamplewordpress.site and www.mysamplewordpress.site

The certificate issued by R3, Let’s Encrypt




Now the domain is secure and works over https. So, you can view wordpress website on the custom domain.