Step-by-Step WordPress Deployment.

Introduction

This project provides a comprehensive guide to deploying a WordPress website on a server using a web server **(Apache)** and a database **(MariaDB)**. It walks through environment setup, installation of required dependencies, WordPress configuration, and final deployment steps to make the site publicly accessible. Following this guide ensures a secure, scalable, and production-ready WordPress deployment.

Feature

- **Dynamic Website** – WordPress provides a fully functional CMS for blogs, portfolios, or business websites.
- **Database-Driven** – Uses MySQL/MariaDB to store content, user data, and settings.
- **Theme and Plugin Support** – Easily customize the site with themes and plugins.
- **Scalable** – Supports multiple users and high traffic when deployed on a proper server.

Prerequisites

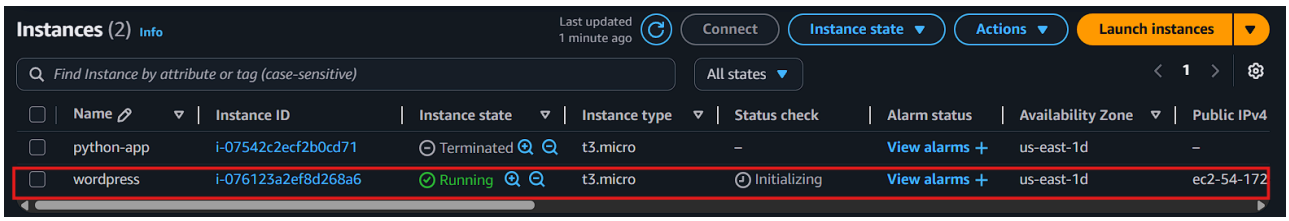
Before deploying the WordPress website, ensure the following are installed and configured:

- **Linux Server** – Amazon Linux
- **Web Server** – Apache installed and running.
- **PHP** – Along with required PHP extensions.
- **Database** – MariaDB for storing WordPress data.

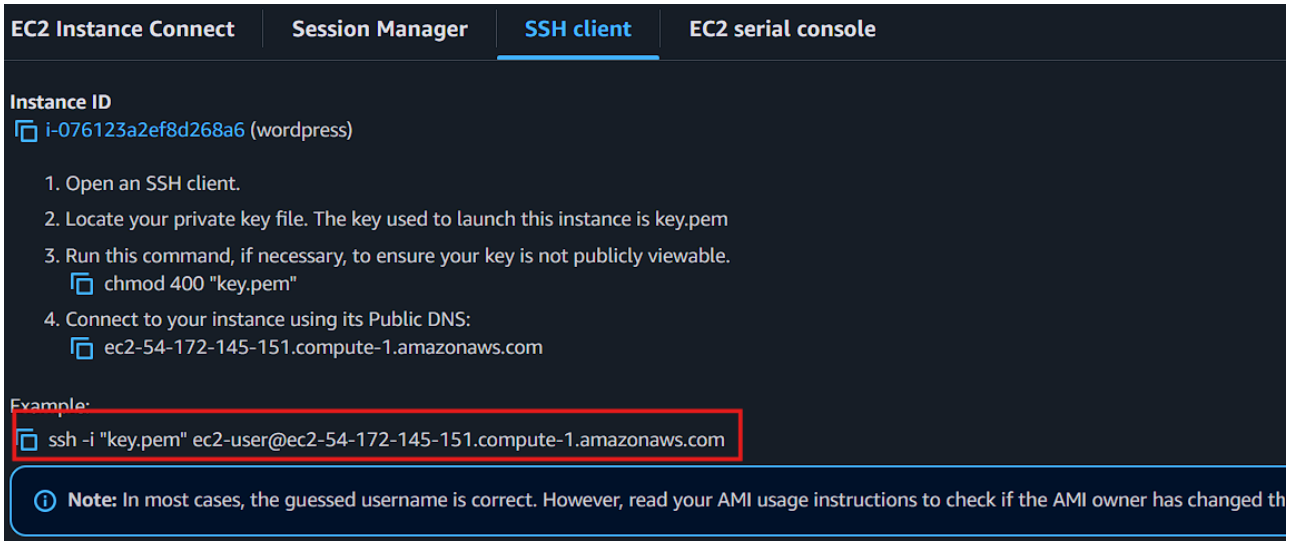
Steps for Delpoyment

Step 1: Launch EC2 instance and Establishing a secure connection to your EC2 instance

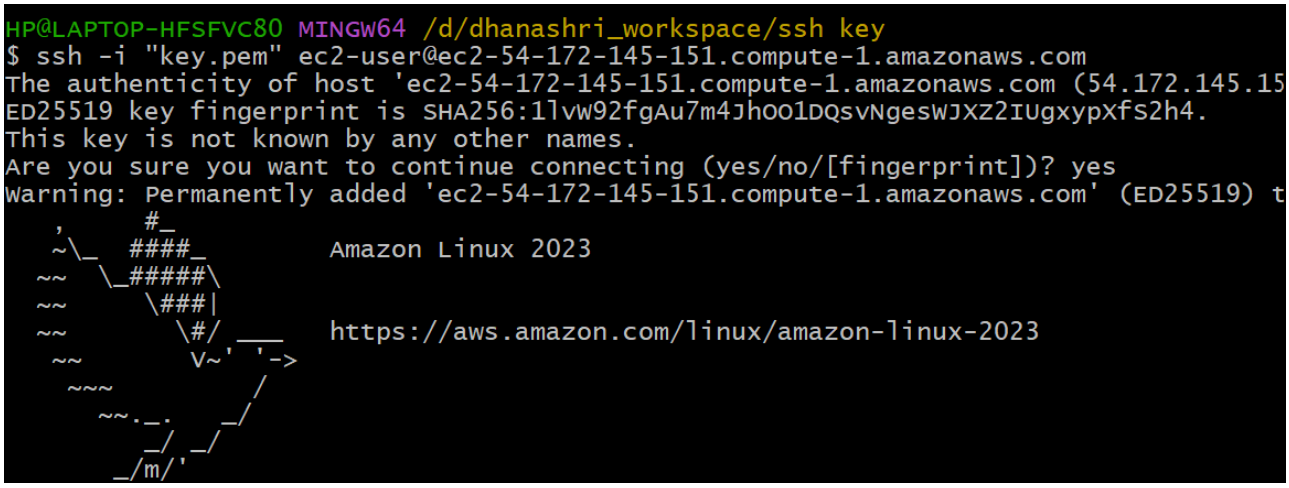
1. Launch instance



2. Copy the SSH command



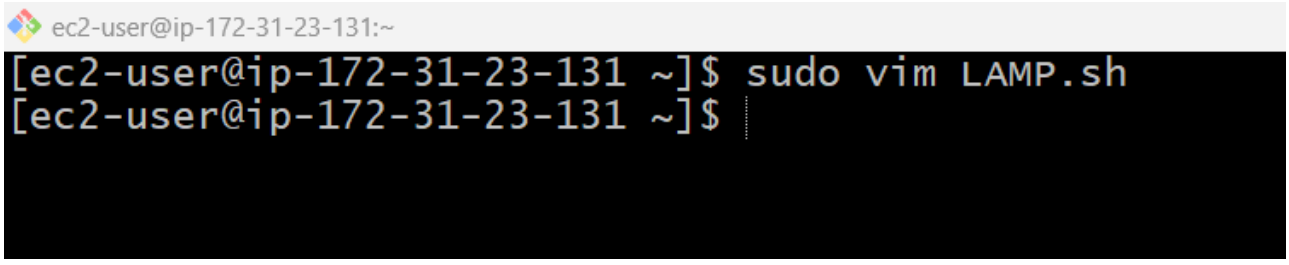
3. Paste command in Git bash



Step 2: Automating LAMP Stack Setup on AWS EC2

1. Create a LAMP.sh file

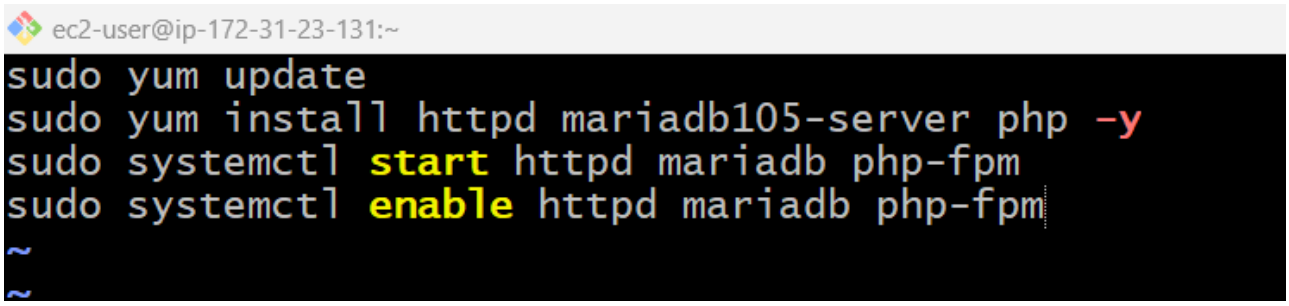
```
sudo vim LAMP.sh
```



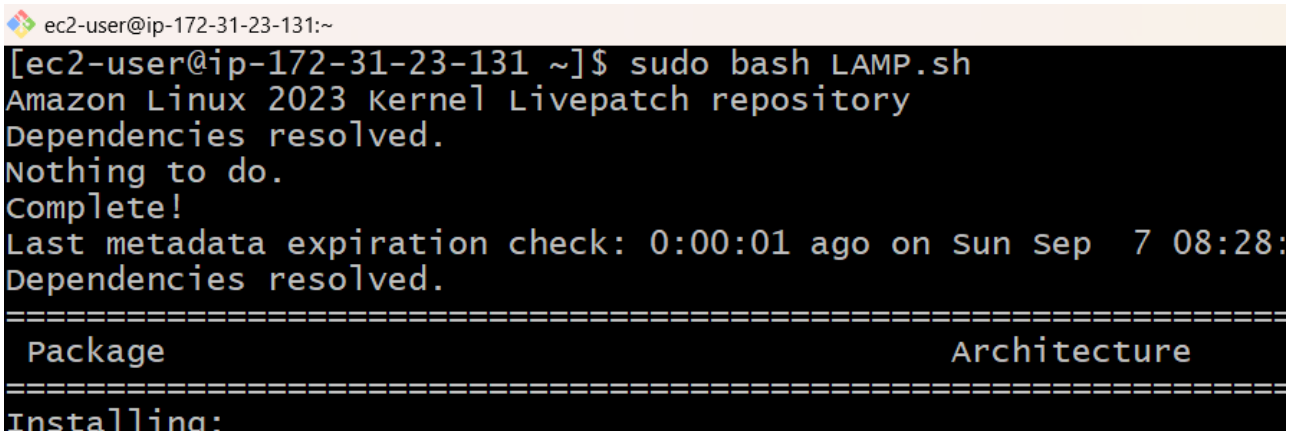
2. Insert the code for installing apache, mysql and php

```
sudo yum update
sudo yum install httpd mariadb105-server php -y
```

```
sudo systemctl start httpd mariadb php-fpm
sudo systemctl enable httpd mariadb php-fpm
```

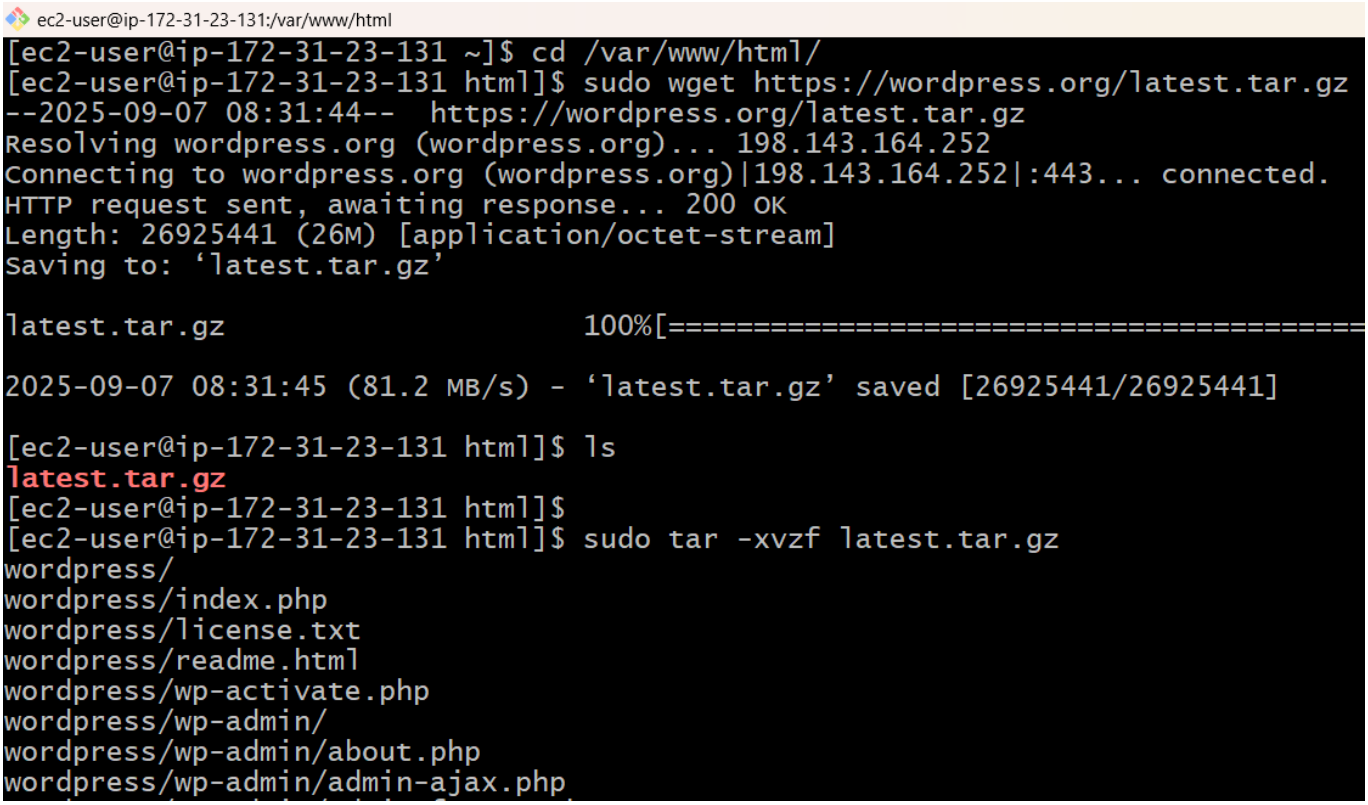


3. Run the file



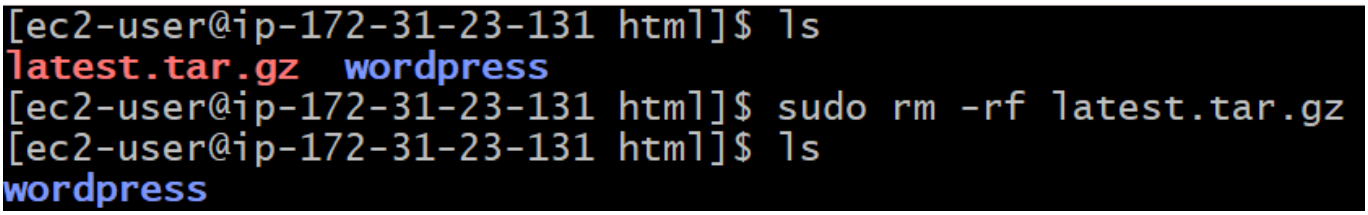
Step 3: Download and Configure WordPress

```
# Download WordPress
sudo wget https://wordpress.org/latest.tar.gz
# Extract the archive
sudo tar -xvzf latest.tar.gz
```



Step 4: Remove latest.tar.gz

```
sudo rm -rf latest.tar.gz
ls
```



Step 5: Go to the wordpress folder

```
cd wordpress/
```

Step 6: Create WordPress Database

1. Generate the username and password.

```
sudo mysql
alter user root@localhost identified by 'root';
```

```
[ec2-user@ip-172-31-23-131 wordpress]$ sudo mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.5.29-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> alter user root@localhost identified by 'root';
Query OK, 0 rows affected (0.002 sec)
```

2. Login to Mysql (mariadb105-server)

```
sudo mysql -u root -p
```

```
[ec2-user@ip-172-31-23-131 wordpress]$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 10.5.29-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input state

MariaDB [(none)]>
```

3. Create Database

```
# Create Database
create database wordpressdb;
# Show Database
show databases;
```

```
MariaDB [(none)]> create database wordpressdb;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpressdb        |
+--------------------+
4 rows in set (0.001 sec)
```

Step 7: Install Connector

```
sudo yum install php8.4-mysqlnd.x86_64
```

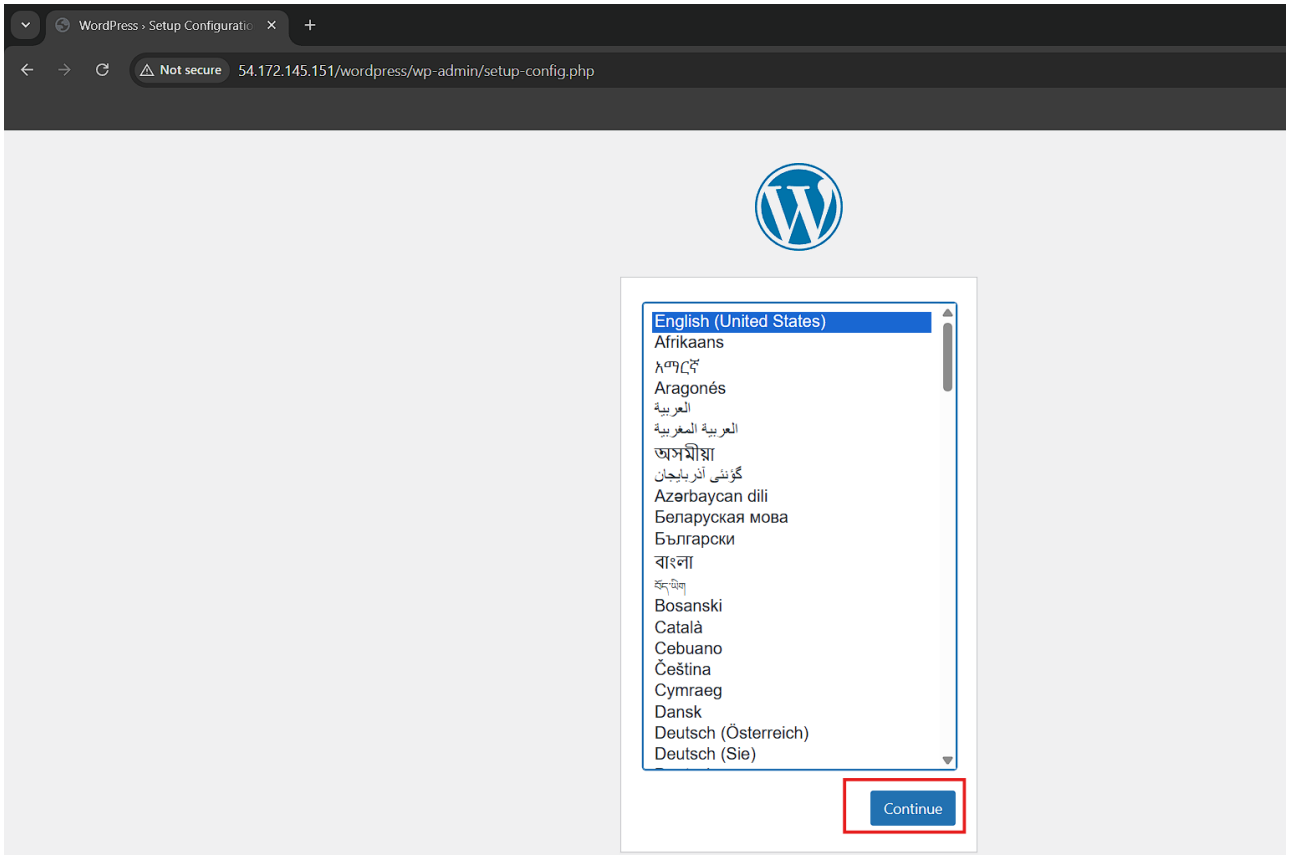
```
[ec2-user@ip-172-31-23-131 wordpress]$ sudo yum install php8.4-mysqlnd.x86_64
Last metadata expiration check: 0:17:42 ago on Sun Sep  7 08:28:43 2025.
Dependencies resolved.
================================================================================
 Package                    Architecture            Version
================================================================================
Installing:
 php8.4-mysqlnd             x86_64                  8.4.10-1.amzn2023.0.1

Transaction Summary
================================================================================
Install  1 Package
```

Step 8: Change ownership of the files

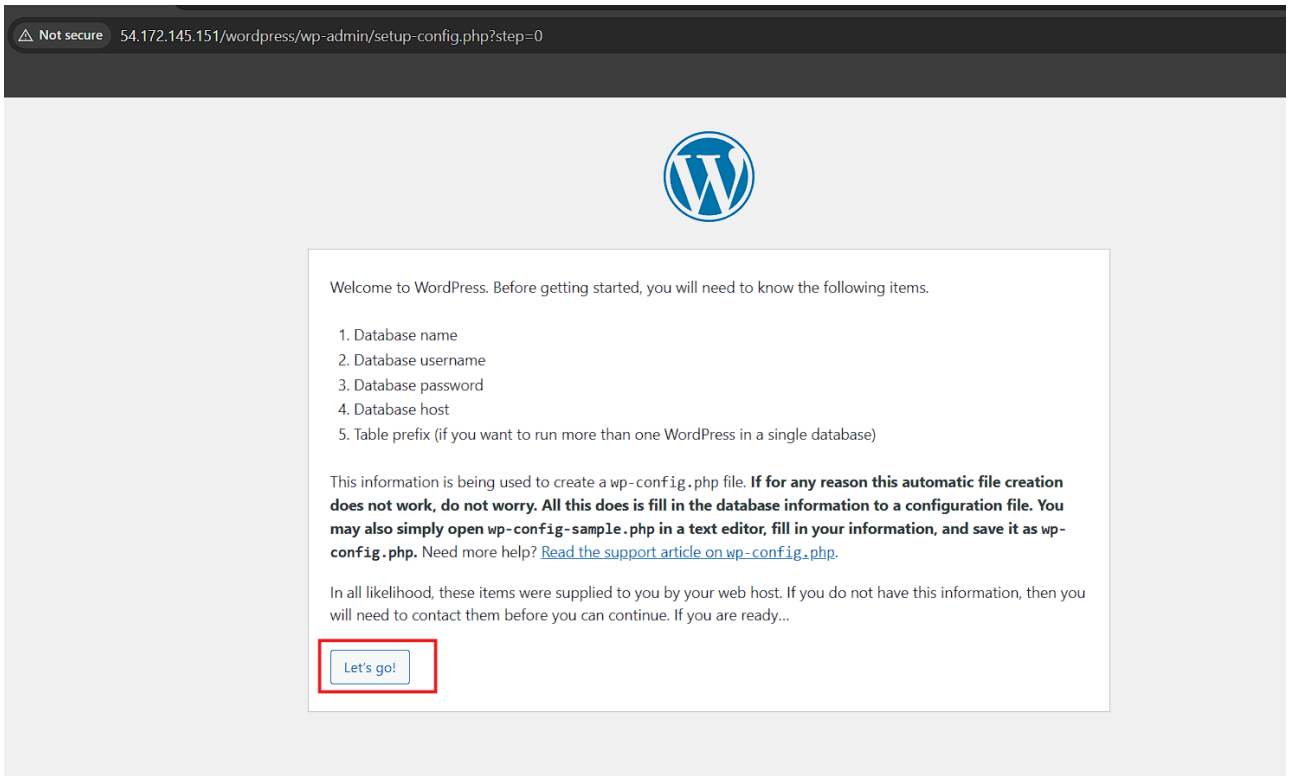
```
ec2-user@ip-172-31-23-131:/var/www/html
[ec2-user@ip-172-31-23-131 html]$ sudo chown -R apache:apache wordpress/
[ec2-user@ip-172-31-23-131 html]$
```

Step 9: Paste the Public IP and Paste it in any browser.

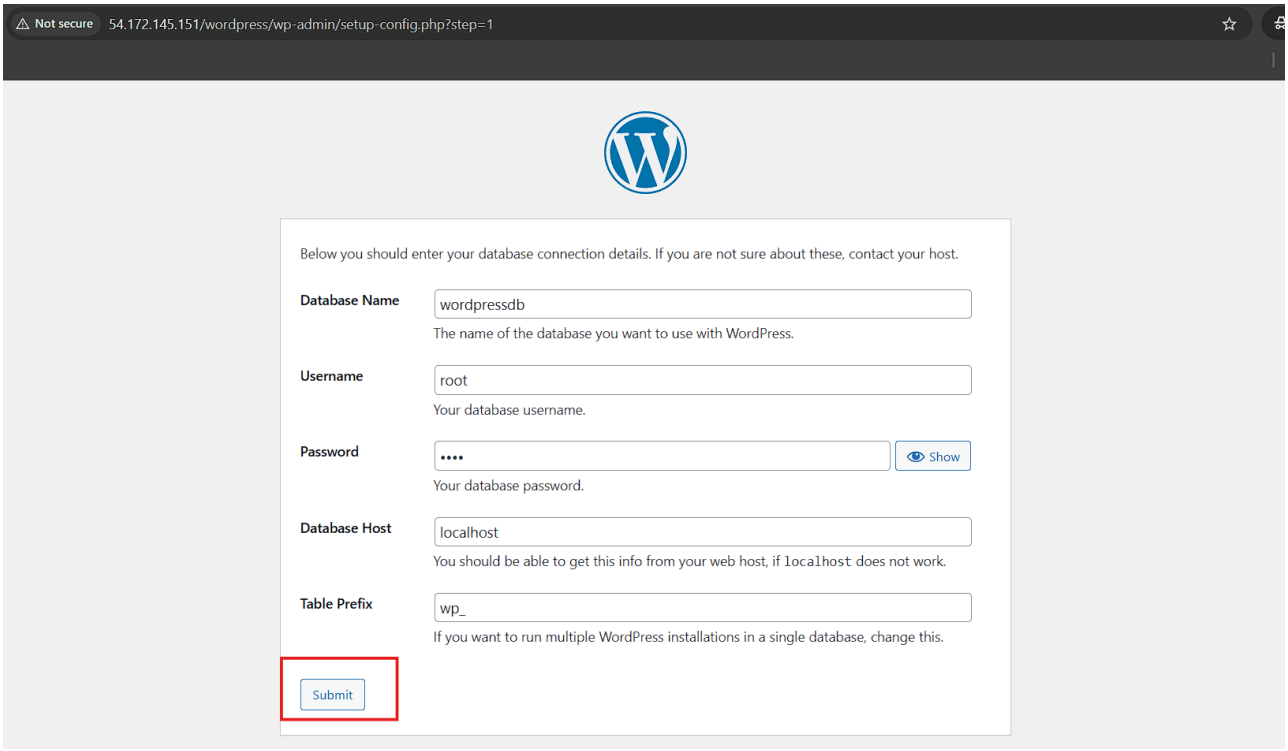
1. Click on Continue



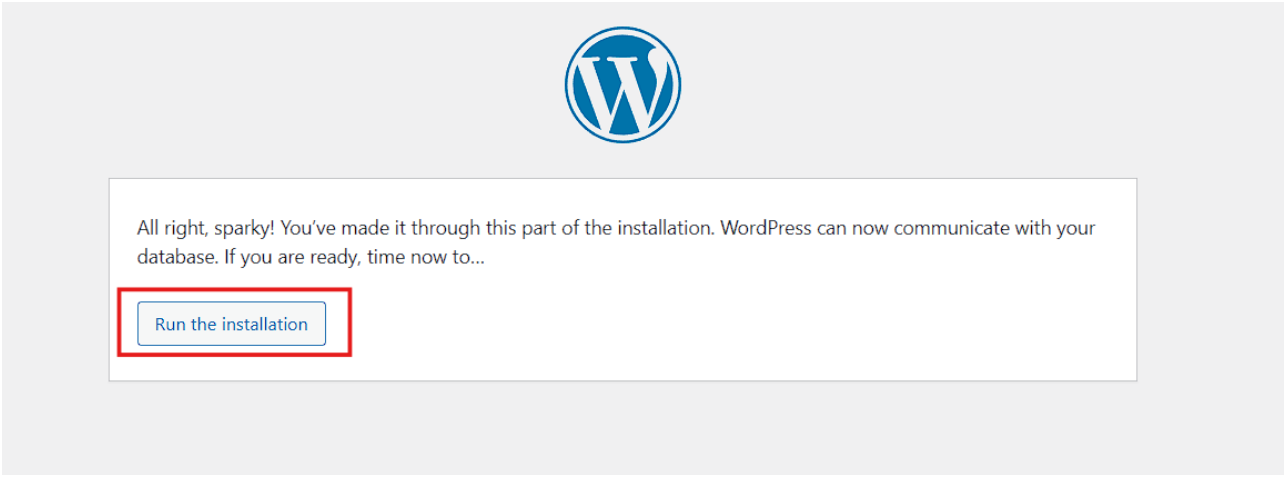
2. Click on Let,s go



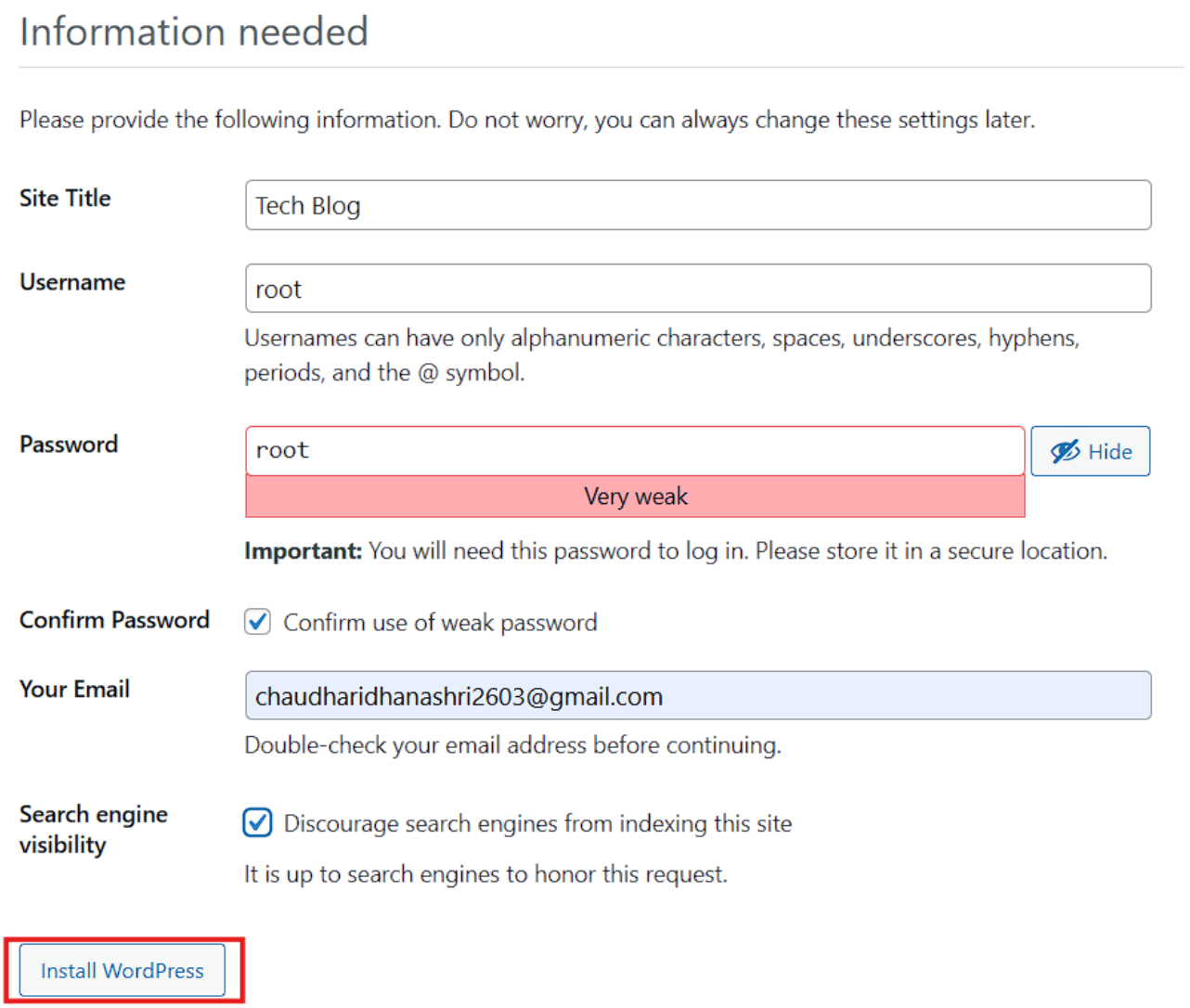
3. Fill the information and click on Submit



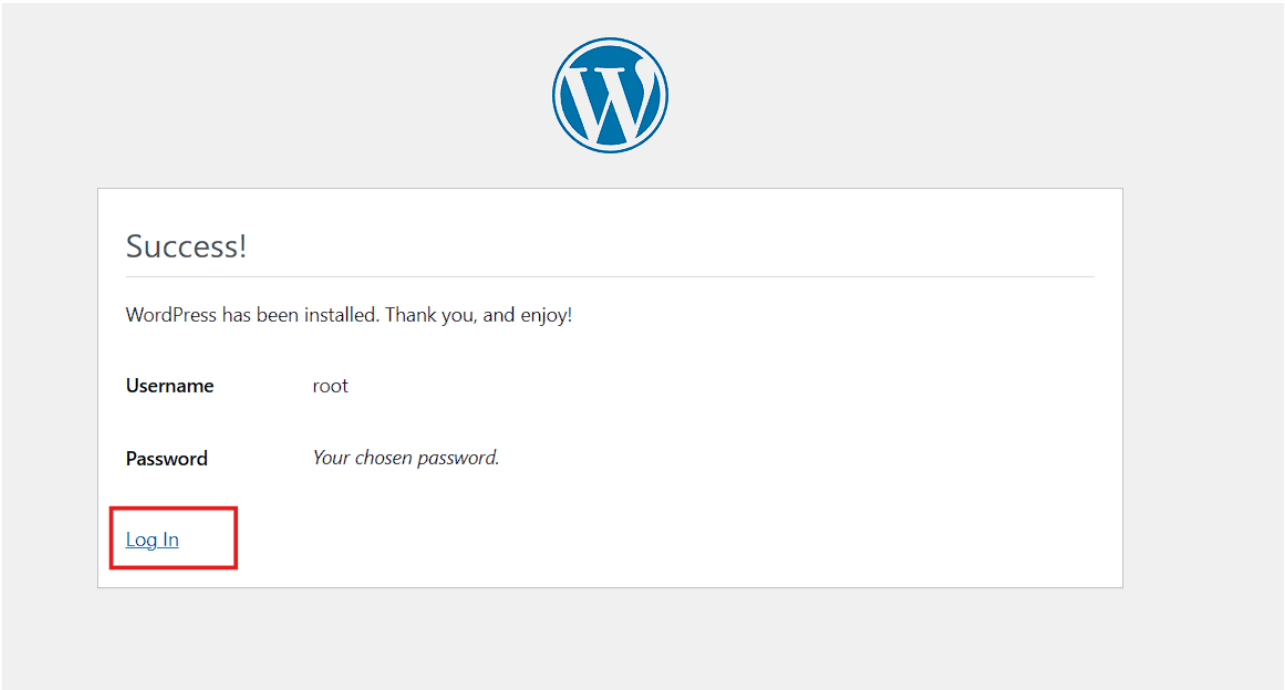
4. Run the Installation

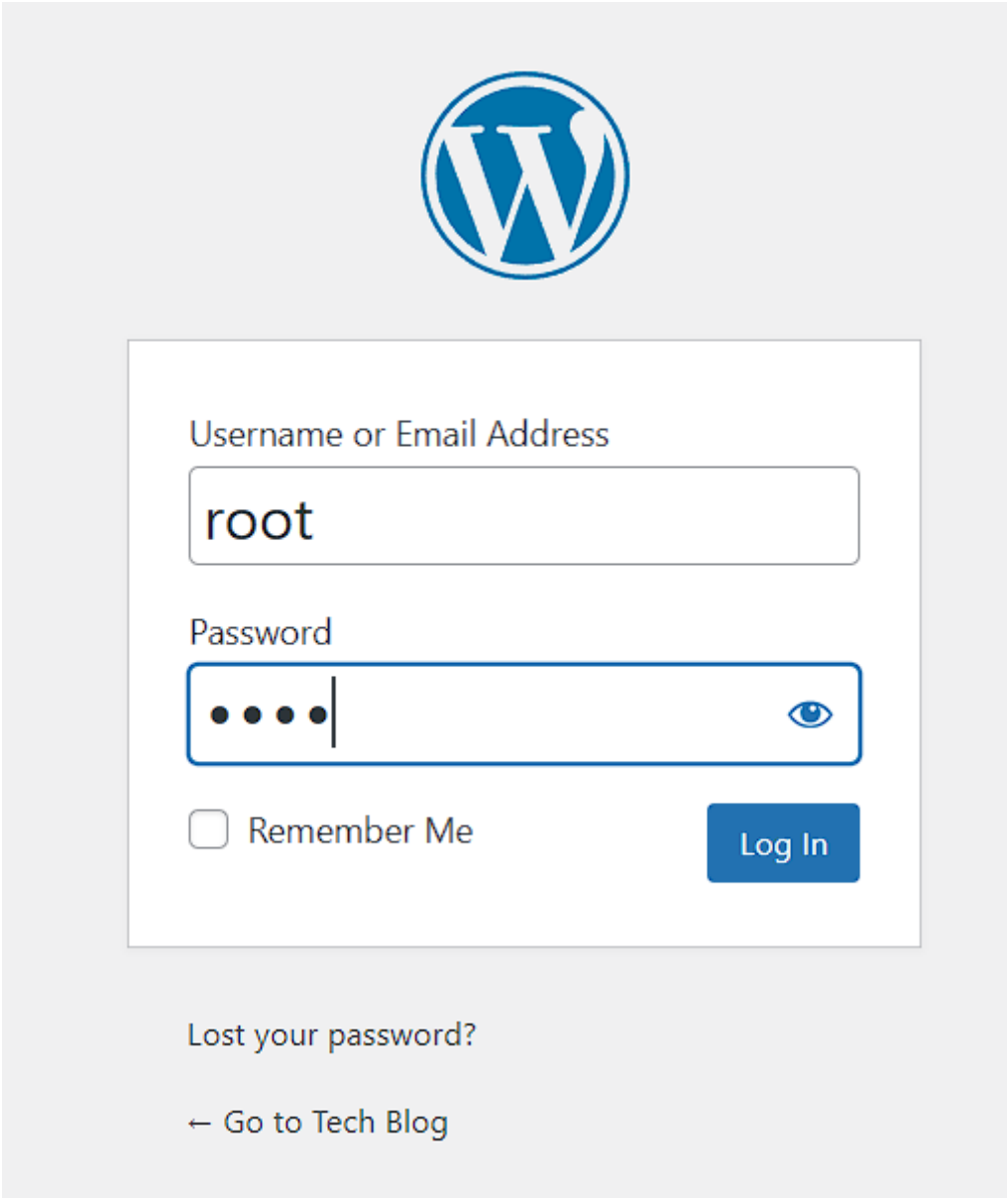


5. Fill the information and click on Install Wordpress

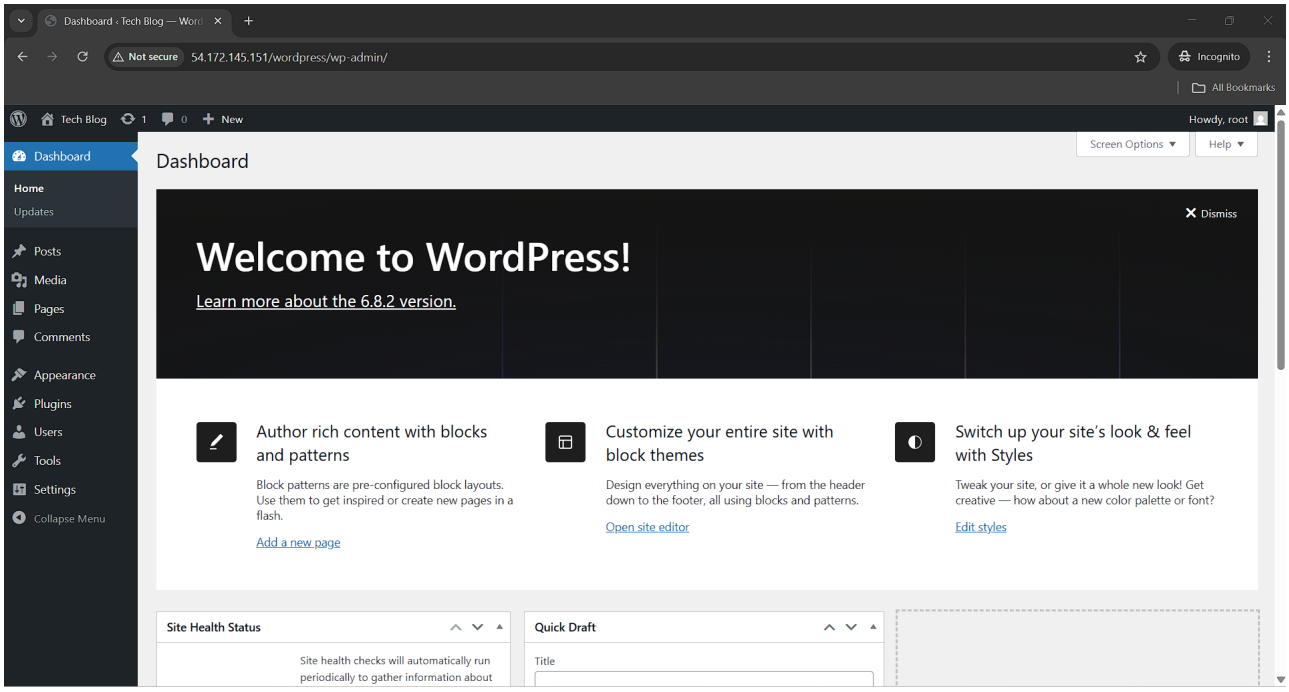


6. Login to Wordpress





7. Deployed Wordpress successfully



8. Table automatically added to Database

```
MariaDB [(none)]> use wordpressdb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [wordpressdb]> show tables;
+-----------------------+
| Tables_in_wordpressdb |
+-----------------------+
| wp_commentmeta        |
| wp_comments           |
| wp_links              |
| wp_options            |
| wp_postmeta           |
| wp_posts              |
| wp_term_relationships |
| wp_term_taxonomy      |
| wp_termmeta           |
| wp_terms              |
| wp_usermeta           |
| wp_users              |
+-----------------------+
12 rows in set (0.000 sec)
```

Summary

This project provides a complete guide to deploying a WordPress website on a Linux server using a web server (Apache) and a database (MariaDB). It covers downloading and installing WordPress, configuring file permissions, setting up the database, and configuring the web server for production use. Following this guide ensures a secure, scalable, and fully functional WordPress site ready for public access.