1. Setting up Wordpress and MySQL in Same EC2 Instances

In this document I am going to show, how to setup Wordpress and MySQL in EC2 Instances.

WordPress is the most popular open-source blogging system and CMS on the Web. It is based on PHP and MySQL. Its features can be extended with thousands of free plugins and themes

Prerequisites:

1. Visit aws.amazon.com and click on "Create an AWS Account."
2. You'll need to enter your email address, password, and account name.
3. Fill in your contact details and agree to the AWS Customer Agreement.
4. You'll need to provide a valid credit card and billing address. AWS may charge a small amount to verify your card, but this charge will be reversed.
5. AWS may ask for a phone number to verify your identity.
6. Once you've completed these steps, you'll receive an email confirming your account creation.

Step 1:Launch Instance

1. Go to the EC2 dashboard and click on "Launch Instance."

2. Give the instance name.

3. Select the Ubuntu AMI.

4. Pick the t2-micro instance type.

5. Proceed with default settings or customize as needed.

6. Set storage and add tags if necessary.

7. Create a new security group allowing HTTP (port 80) and MySQL (port 3306) access.

8. Create an existing key pair for SSH access.

9. Launche an EC2 instance.
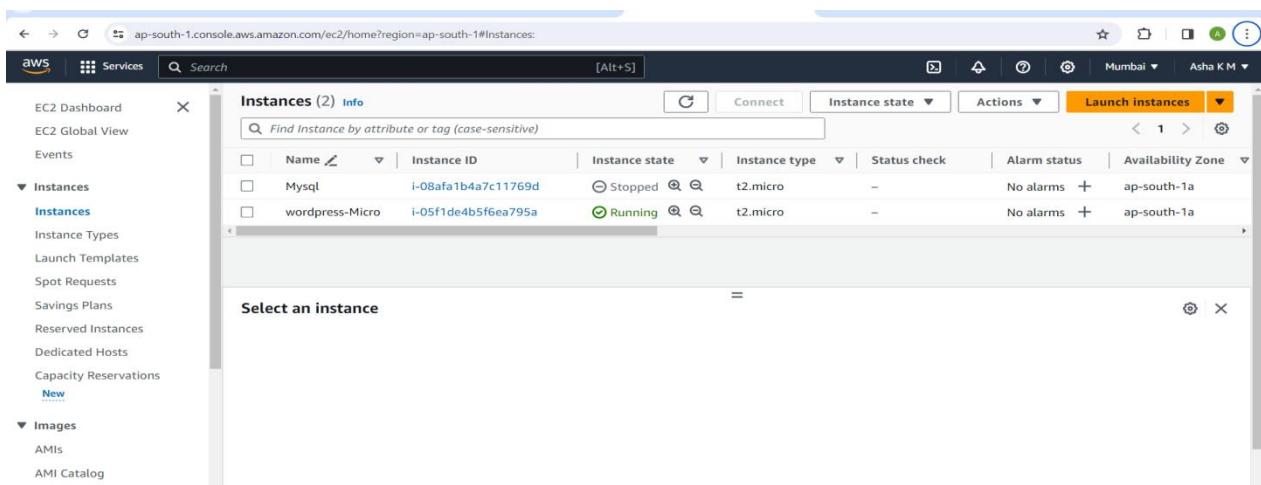
Step 2: Install MySQL on the EC2 instance

1. Use a terminal or SSH client to connect to the instance using the provided key pair.

2. Update repositories and install MySQl.

```
Sudo apt update
sudo apt install mysql-server
```

```
Expanded Security Maintenance for Applications is not enabled.

9 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


Last login: Fri Jan  5 06:50:48 2024 from 13.233.177.5
ubuntu@ip-172-31-37-121:~$ sudo apt update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1268 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [260 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1257 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [205 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1021 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [227 kB]
0% [9 Packages store 0 B]
```

3. Install Apache, PHP, and other required packages for WordPress

Sudo apt install apache2 php php-mysql php-curl php-gd php-mbstring php-xml php-xmlprc php-soap php-intl php-zip

4. Secure database server : Follow the prompts to set a root password, remove anonymous users, disallow remote root login, and remove test databases.

Sudo mysql secure installation

1. Type Y to set a password, and type a secure password twice.
2. Type Y to remove the anonymous user accounts.
3. Type N to enable the remote root login.
4. Type Y to remove the test database.
5. Type Y to reload the privilege tables and save your changes.

Step 3: Create Database and user
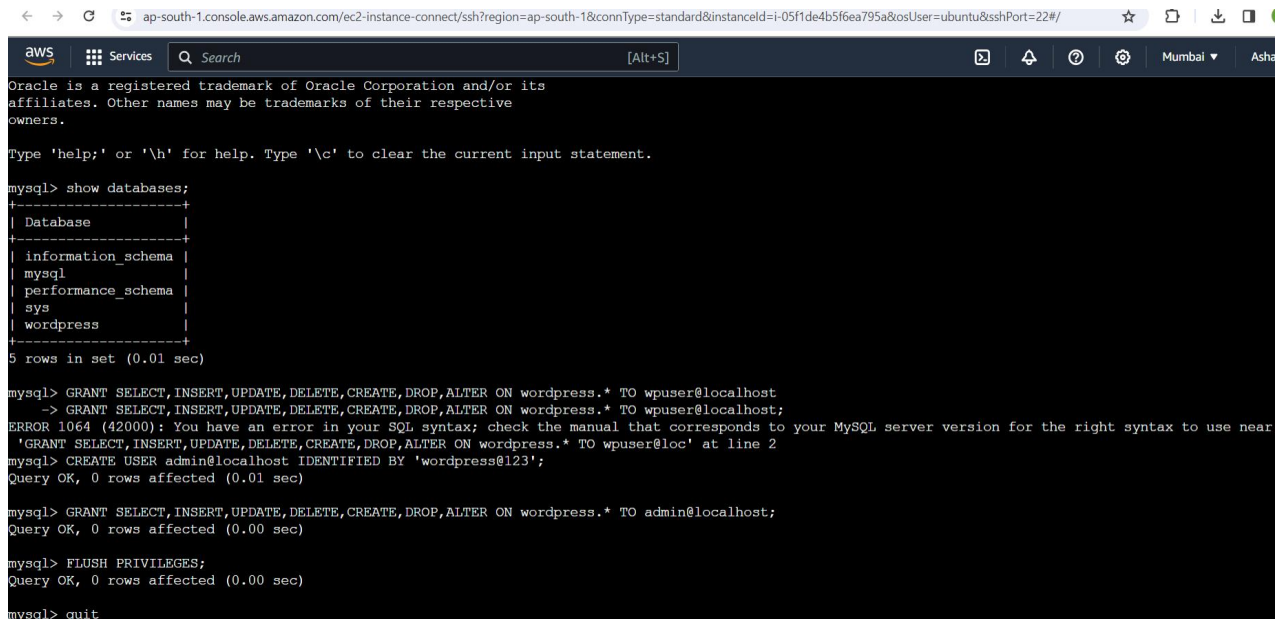
1. Login to MySQL

Sudo mysql -u root -p

2. Create user and database for wordpress

```
CREATE USER 'admin'@'localhost' IDENTIFIED BY 'my password'
CREATE DATABASE `wordpress-db`;

GRANT ALL PRIVILEGES ON *.* TO 'admin'@'localhost' IDENTIFIED
BY 'my password' WITH GRANT OPTION;

FLUSH PRIVILEGES

Quit
```


Step 4: Install WordPress on same Instance

1. Create the installation directory and download the file from WordPress.org.
2. Download the latest WordPress installation package and Unzip and unarchive the installation package.
3. The installation folder is unzipped to a folder called wordpress.

```
cd  /var/www/html/

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

Sudo tar -xzf latest.tar.gz
```

```
Reading state information... Done
php is already the newest version (2:8.1+92ubuntu1).
php-mysql is already the newest version (2:8.1+92ubuntu1).
apache2 is already the newest version (2.4.52-1ubuntu4.7).
mysql-client is already the newest version (8.0.35-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 12 not upgraded.
ubuntu@ip-172-31-37-121:~$ wget -c http://wordpress.org/latest.tar.gz
--2024-01-05 10:53:56--  http://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://wordpress.org/latest.tar.gz [following]
--2024-01-05 10:53:57--  https://wordpress.org/latest.tar.gz
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24479697 (23M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz                           20%[==============>
```
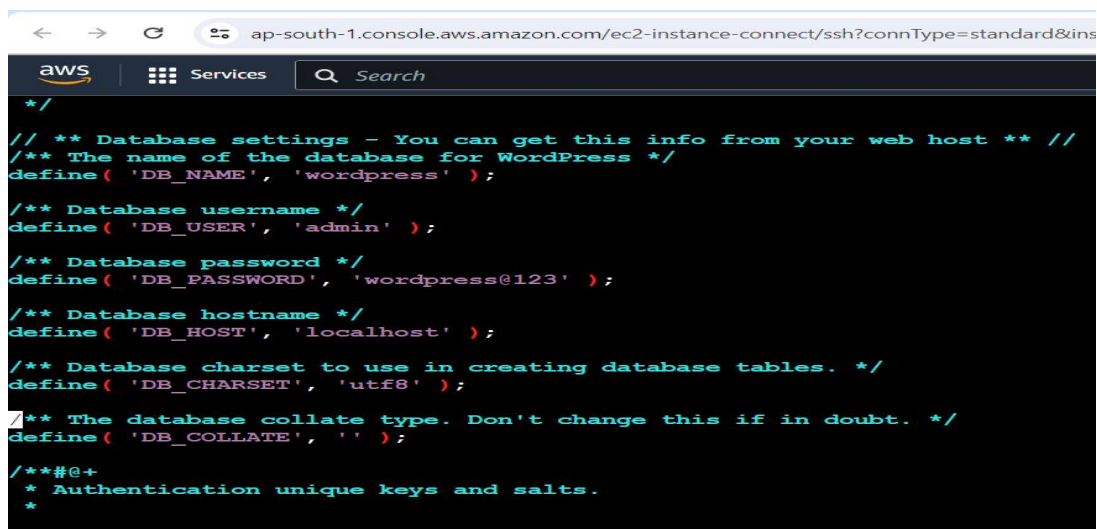
4. Copy the wp-config-sample.php file to a file called wp-config.php. This creates a new configuration file and keeps the original sample file intact as a backup.

> **cd wordpress**
>
> **Sudo mv wp-config-sample.php wp-config.php**
>
> **Sudo nano wp-config.php**

5. Edit the host, database name, user and password as shown in the following picture.

6. In the same file find the section called Authentication Unique Keys and Salts. Visit https://api.wordpress.org/secret-key/1.1/salt/ to randomly generate a set of key values that you can copy and paste into your wp-config.php file as follows:

```
*
* You can change these at any point in time to invalidate all existing cookies.
* This will force all users to have to log in again.
*
* @since 2.6.0
*/
define('AUTH_KEY',         '2$_e3+G:-FQ(}odyq6CW*,[OSE&@v`_[KYRIWDr^8y>$.@Z&]HCa<J#V;N:0#M(?');
define('SECURE_AUTH_KEY',  '_o=)e|bOOSDduc2-[(=m.cR=_<ejg:dQv;|pGtWCv&?q.p{h]Fs^BDFS+#]~qP<4');
define('LOGGED_IN_KEY',    '0)jyuwdq_4$-7|i{Oo{ vUTq-S-FE1ObA(^<I/c<&gFP$zX-n*mgL&XBWE[?Qt6z');
define('NONCE_KEY',        '(+`m`;/Eo64U=:l-0#r1<V>(lDVaoS{7{m*4xQSX)+Tai$7v&8W-JI9N.-?:Bhz)');
define('AUTH_SALT',        'k%d7#=AT[IAEcZ91XfEg3er+%dB8i41l -bl{3(DCC7Sz:!q%7a=|_Mo2tRkaqx]');
define('SECURE_AUTH_SALT', 'A@,Lj/IlT.8J?RKA@oSxz-QJ;zCEW}}@]K<aD,uq+kks3D| g-dPt&R*}i~#Qfkk');
define('LOGGED_IN_SALT',   'pDUlm$#jIAahWTfm8}C^qSDcp6QKr(pD[l[?- m339R++`@% wu{o7GwU|b&=ryg');
define('NONCE_SALT',       '5@pH<lcz:7`@lkkCrAs$W(K-&u++=IP`}+CwU_or0hx-]K%o-Sb),pT>U@?5?RbY');

/**#@-*/

/**
* WordPress database table prefix.
*
```

7. Set correct permissions:

> sudo chown -R www-data:www-data /var/www/html/wordpress
>
> sudo chmod -R 755 /var/www/html/wordpress

Step5 : Wordpress installation

1. Open a web browser and navigate to your server's IP address or domain name.
2. You'll be prompted to set up WordPress. Select your language and proceed.
3. Fill in the required information such as Site Title, Username, Password, and Email.

4. Click "Install WordPress" and then "Login" with your chosen credentials.
5. Once logged in, you'll enter the WordPress dashboard.

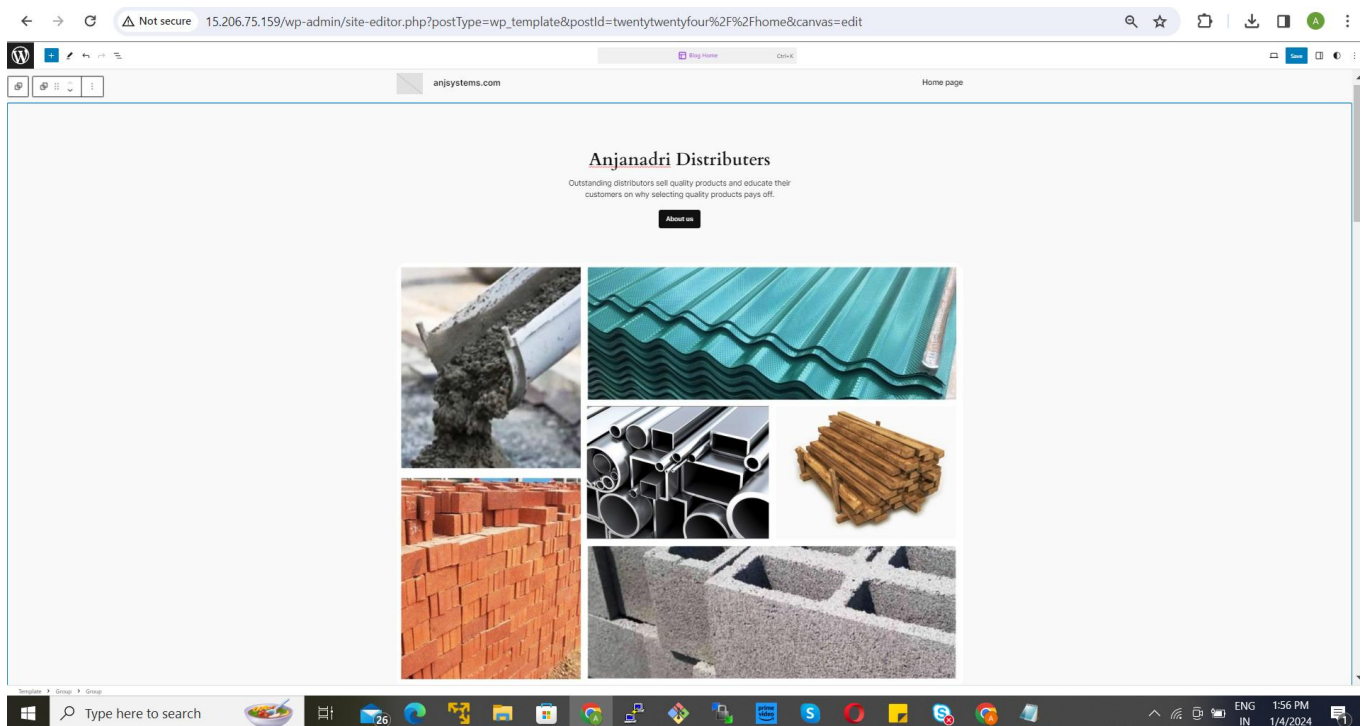6. You will notice the "Hello world!" post. Let's delete it and write something more interesting…
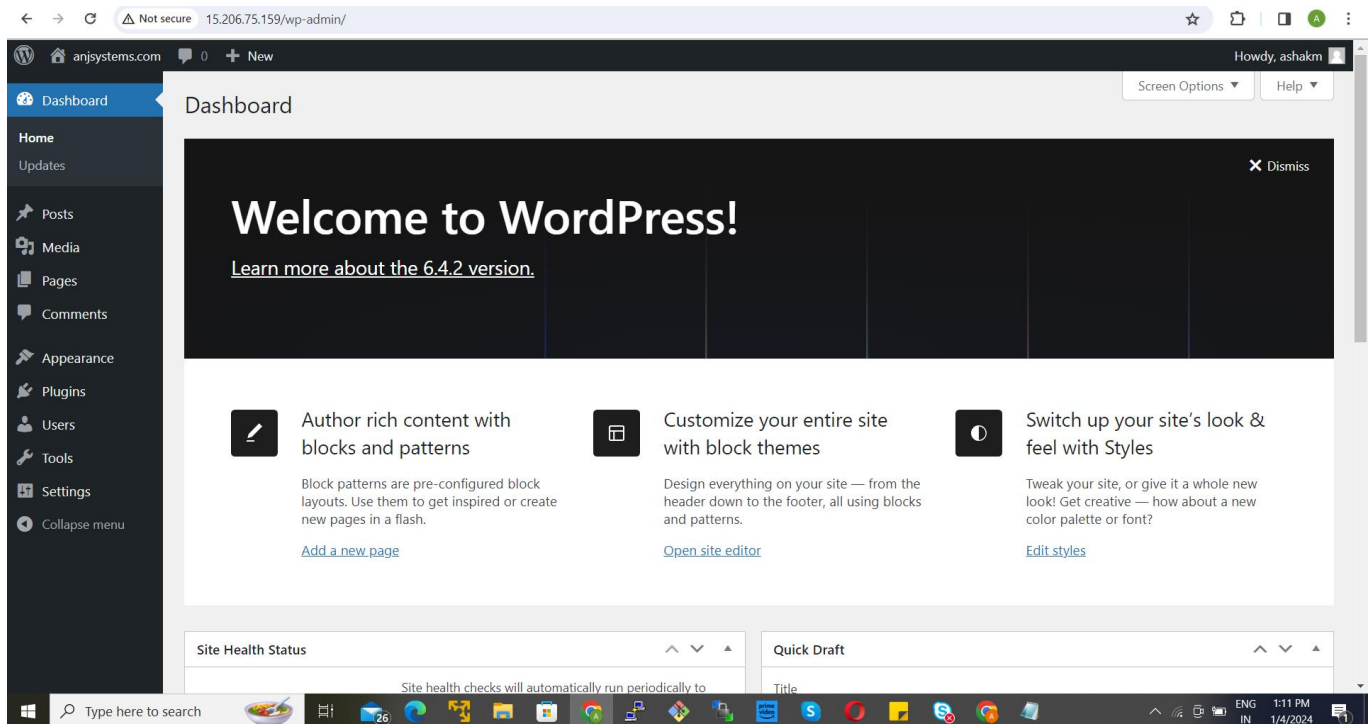
7. Configure your site settings, install themes, plugins, and create content as needed

8. Explore the settings available in the dashboard to customize your site further.

2. Setting up Wordpress and MySQL in two different EC2 Instances

In this document I am going to show, how to setup Wordpress and MySQL in 2 different EC2 Instances.

WordPress is the most popular open-source blogging system and CMS on the Web. It is based on PHP and MySQL. Its features can be extended with thousands of free plugins and themes

Prerequisites:

1) Visit aws.amazon.com and click on "Create an AWS Account."
2) You'll need to enter your email address, password, and account name.
3) Fill in your contact details and agree to the AWS Customer Agreement.
4) You'll need to provide a valid credit card and billing address. AWS may charge a small amount to verify your card, but this charge will be reversed.
5) AWS may ask for a phone number to verify your identity.
6) Once you've completed these steps, you'll receive an email confirming your account creation.

Step 1:Launch Instance for Mysql

1. Go to the EC2 dashboard and click on "Launch Instance."

2. Give the instance name.

3. Select the Ubuntu AMI.

4. Pick the t2-micro instance type.

5. Proceed with default settings or customize as needed.

6. Set storage and add tags if necessary.

7. Create a new security group allowing HTTP (port 80) and MySQL (port 3306) access.

8. Create an existing key pair for SSH access.

9. Launch an EC2 instance.
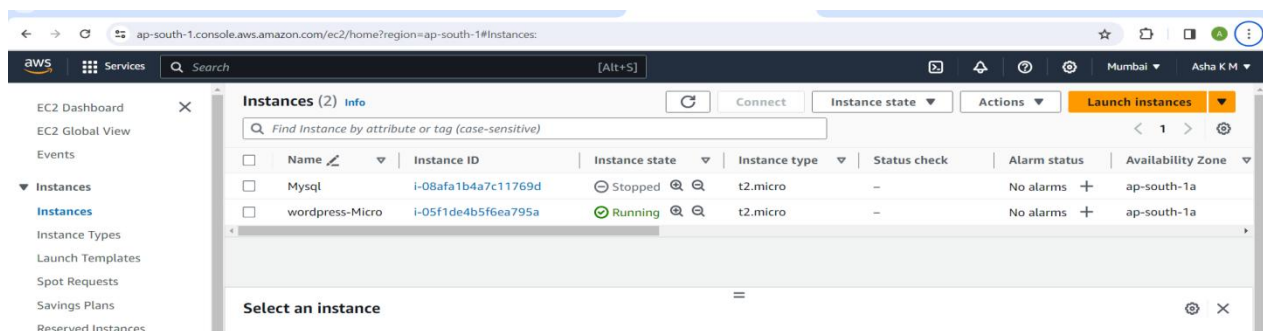
Step 2: Launch EC2 instance for wordpress

1. Follow similar steps as above but choose an AMI suitable for hosting WordPress (such as Ubuntu Server).

2. Create a new security group allowing HTTP (port 80) and HTTPS (port 443) access.

3. Create an existing key pair for SSH access.

4. Launch an EC2 instance

Step 3: Configuring Mysql Instance

1. Use a terminal or SSH client to connect to the Mysql instance using the provided key pair.

2. Update repositories and install MySql

```
Sudo apt update
sudo apt install mysql-server
```

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


Last login: Fri Jan  5 06:50:48 2024 from 13.233.177.5
ubuntu@ip-172-31-37-121:~$ sudo apt update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1268 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [260 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1257 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [205 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1021 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [227 kB]
0% [9 Packages store 0 B]
```

3. Secure database server : Follow the prompts to set a root password, remove anonymous users, disallow remote root login, and remove test databases.

Sudo mysql secure installation

1. Type Y to set a password, and type a secure password twice.
2. Type Y to remove the anonymous user accounts.
3. Type N to enable the remote root login.
4. Type Y to remove the test database.
5. Type Y to reload the privilege tables and save your changes.

Step 4: Create Database and user for wordpress in Mysql Instances

1. Login to MySQL

Sudo mysql -u root -p

2. Create user and database for wordpress

CREATE USER 'wordpress-db'@'wordpress instance private ip' IDENTIFIED BY 'my password'

CREATE DATABASE `wordpress-db`;

GRANT ALL PRIVILEGES ON *.* TO 'wordpress-db'@'wordpress instance private ip' IDENTIFIED BY 'my password' WITH GRANT OPTION;

FLUSH PRIVILEGES

Quit

```
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> CREATE USER 'wordpressuser'@'wordpress_instance_private_ip' IDENTIFIED BY 'password';
ERROR 1819 (HY000): Your password does not satisfy the current policy requirements
mysql> CREATE USER 'wordpressuser'@'wordpress_instance_private_ip' IDENTIFIED BY 'password';
ERROR 1819 (HY000): Your password does not satisfy the current policy requirements
mysql> CREATE USER 'wordpress-db'@'172.31.37.121' IDENTIFIED BY 'wordpress@123';
ERROR 1819 (HY000): Your password does not satisfy the current policy requirements
mysql> ^C
mysql>
mysql>  CREATE USER 'wordpress-db'@'172.31.37.121' IDENTIFIED BY 'Ashakm@123';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpress-db'@'172.31.37.121';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
ubuntu@ip-172-31-40-42:~$
```

Step 4: Configure the wordpress Instance

1. Use SSH to connect to the WordPress instance similarly as before.

2. Install necessary packages (like Apache, PHP, etc.) and download WordPress.

3. Copy the wp-config-sample.php file to a file called wp-config.php. This creates a new configuration file and keeps the original sample file intact as a backup.

> **Sudo apt update**
>
> **Sudo apt install apache2 php mysql-client php-mysql**
>
> **Sudo wget https://wordpress.org/latest.tar.gz**
>
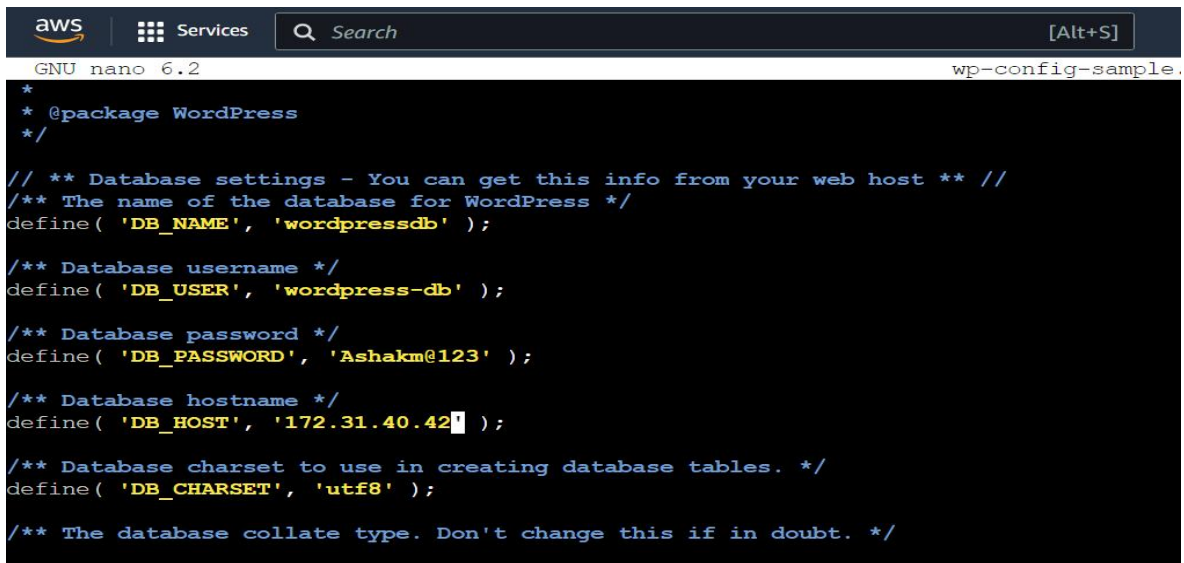> **Sudo tar -xzf latest.tar.gz**
>
> **Cd /var/www/html/wordpress**
>
> **Sudo mv wp-config-sample.php wp-config.php**
>
> **Sudo nano wp-config.php**

4. Edit the host, database name, user and password as shown in the following picture.

define('DB_NAME', 'wordpress');

define('DB_USER', 'wordpress-db');

define('DB_PASSWORD', 'password');

define('DB_HOST', 'mysql_instance_private_ip');

5. For the WordPress instance to connect to the MySQL instance, use the private IP of the MySQL instance as the DB_HOST in the WordPress configuration (wp-config.php). This ensures secure communication within the AWS network.

6. In the same file find the section called Authentication Unique Keys and Salts. Visit https://api.wordpress.org/secret-key/1.1/salt/ to randomly generate a set of key values that you can copy and paste into your wp-config.php file as folows
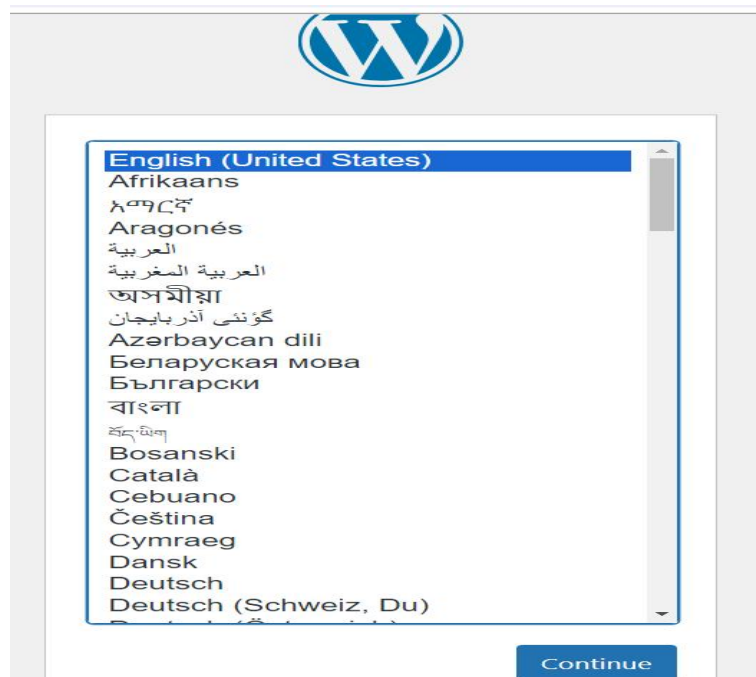

7. Set the correct permissions.

```
sudo chown -R www-data:www-data /var/www/html/wordpress

sudo chmod -R 755 /var/www/html/wordpress
```

Step 5 : Wordpress installation

1. Restart apache or relevant services

```
Sudo systemctl restart apache2
```

2. Open a web browser and go to the public IP or domain associated with your WordPress instance.

3. You should see the WordPress installation page.Select your language and proceed.

4. Fill in the required information such as Site Title, Username, Password, and Email

9. Click "Install WordPress" and then "Login" with your chosen credentials.

10. Once logged in, you'll enter the WordPress dashboard.

11. You will notice the "Hello world!" post. Let's delete it and write something more interesting…
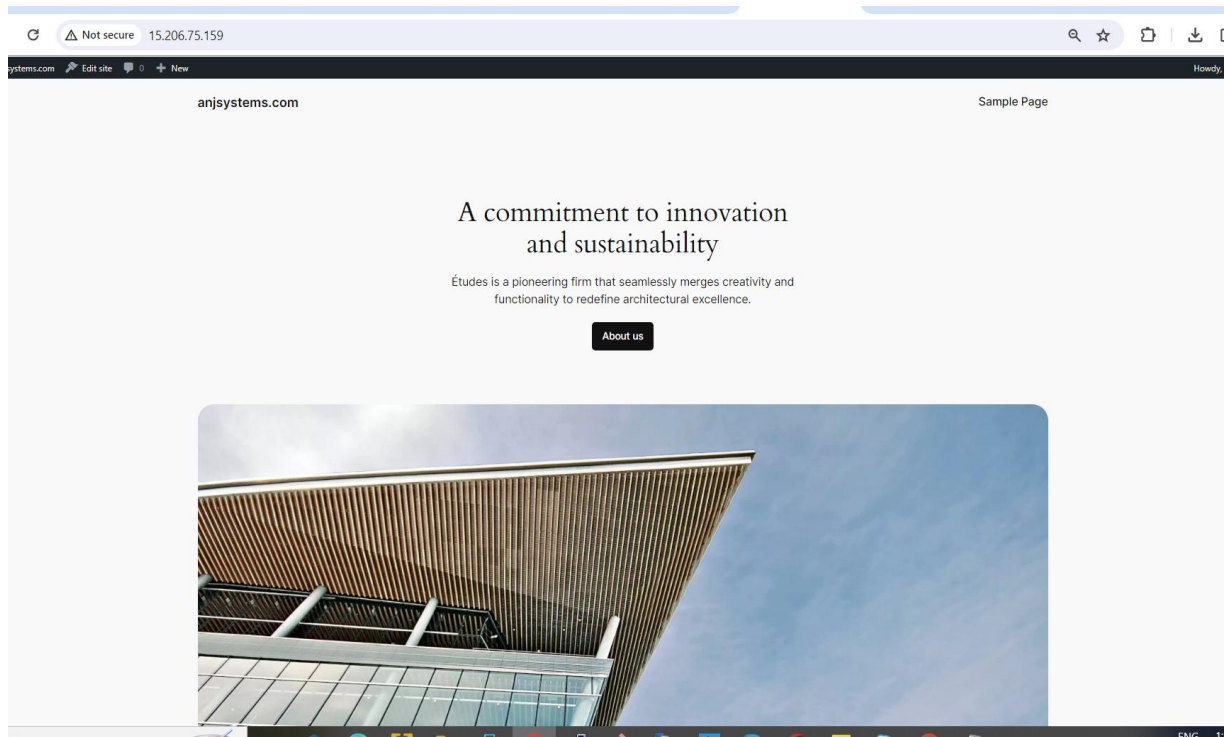
12. Configure your site settings, install themes, plugins, and create content as needed

13. Explore the settings available in the dashboard to customize your site further.

SUMMARY

1. With these steps, you'll have a WordPress instance connected to a MySQL instance on AWS using EC2 instances and configured security groups for access control.

2. This setup separates the WordPress application and the MySQL database, ensuring better security and scalability. Remember, these steps may vary depending on your specific AWS setup and configurations. Always prioritize security by properly configuring security groups and using private IPs for internal communication.

Reference:

1. https://www.linkedin.com/pulse/setting-up-wordpress-mysql-two-different-ec2-mohit-agarwal/

2. https://youtu.be/18rfWZYbS7o?si=iLOgv6YiXP3O9R-q

3. https://ubuntu.com/tutorials/install-and-configure-wordpress#5-configure-database

4. https://chat.openai.com/share/66dabda3-3f4d-4a37-863c-bdbeb6b31cf4