

Deploy application in monolithic and microservices architecture.

1).Deploying WordPress and MySQL in a Monolithic Architecture on AWS:-

Go to AWS cloud Services Create EC2 instance

Requirements for Ec2 instance are:-

*) Give a Name “ “

*) Use Ubuntu AMI (Ubuntu server 24.04 Lts SSD volume Type (free tire eligible)

*) select Instance Type t2. micro (Free tire eligible)

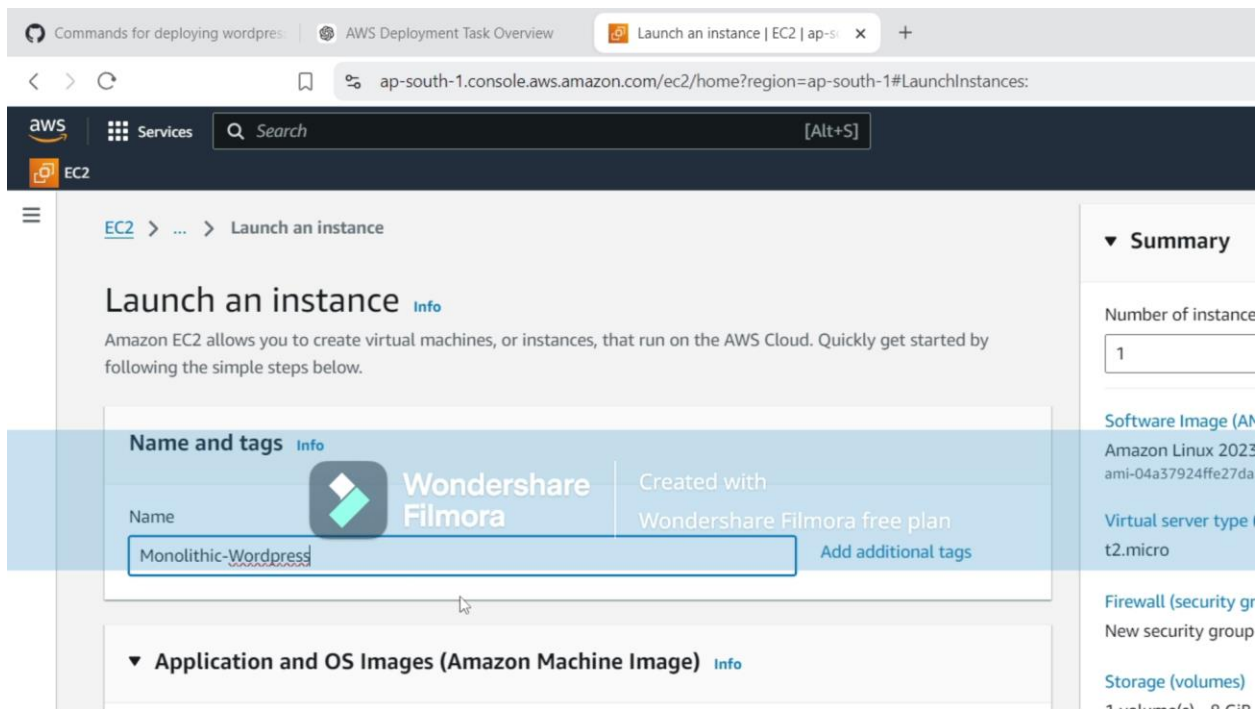
*) key Pair use your own or create one key-pair here i have used (jenkins-server)

*) network settings:- Edit here vpc i have used default, and subnet No preference and Enable Auto-assign public IP

Network configuration Add Security group rule:-

Type SSH, source type Anywhere ,Type Http, Anywhere, Mysql/Aurora

And Connect Through SSH I have Used Putty to connect



Type	Info	Protocol	Info	Port range	Info
ssh		TCP		22	
Source type	Info	Source	Info	Description - optional	Info
Anywhere		Add CIDR, prefix list or security group		e.g. SSH for admin desktop	
		0.0.0.0/0			

Security group rule 2 (TCP, 3306, 0.0.0.0/0)						Remove
Type	Info	Protocol	Info	Port range	Info	
MYSQL/Aurora		TCP		3306		
Source type	Info	Source	Info	Description - optional	Info	
Anywhere		Add CIDR, prefix list or security group		e.g. SSH for admin desktop		

Summary

Number of instances

1

Software Image (AMI)

Canonical, Ubuntu, ami-0dee22c13ea7a9a

Virtual server type

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) 8 GiB

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type

Free tier eligible

ami-0dee22c13ea7a9a67 (64-bit (x86)) / ami-0c8eea98010057bd0 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture	AMI ID	Username	
64-bit (x86)	ami-0dee22c13ea7a9a67	ubuntu	Verified provider

Summary

Number of instances

Software Image (AMI)

Canonical, Ubuntu, ami-0dee22c13ea7a9a67

Virtual server type

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) 0 GiB



Commands for deploying wordpress



AWS Deployment Task Overview



Instances | EC2 | ap-south-1



ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:v=3;\$case=tags:true%5C,client:false;\$



Services



Search

[Alt+S]



EC2

Dashboard



EC2 Global View

Events

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity

Reservations **New**

Instances (1/1) Info

Last updated

1 minute ago



Connect

Insta



Find Instance by attribute or tag (case



Name



Instance ID



Monolithic-Wo...

i-05fff5973b6aa4264



Wondershare
Filmora

Created with

Wondershare Filmora free plan

i-05fff5973b6aa4264 (Monolith

Details

Status and alarms

Mo



PuTTY Configuration



Category:

Logging

Terminal

Keyboard

Bell

Features

Window

Appearance

Behaviour

Translation

Selection

Colours

Connection

Data

Proxy

SSH

Kex

Host keys

Cipher

Auth

Credentials

GSSAPI

TTY

Credentials to authenticate with

Public-key authentication

Private key file for authentication:

Browse...

Certificate to use with the private key (optional):

Browse...

Plugin to provide authentication responses

Plugin command to run



Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

Last login: Tue Nov 5 06:39:57 2024 from 103.255.145.106

ubuntu@ip-172-31-8-201:~\$ history

```
1 sudo apt update
2 sudo apt upgrade -y
3 sudo apt install apache2 -y
4 apache2 --version
5 apache2 -- version
6 sudo systemctl enable apache2
7 sudo systemctl start apache2
8 sudo apt install mysql-server
9 sudo mysql -u root
10 sudo apt install php libapache2-mod-php php-mysql
11 sudo systemctl restart apache2
12 ls
13 ls -a
14 cd /var/www/html
15 ls
16 sudo rm index.html
17 ls
18 wget https://wordpress.org/latest.zip
19 sudo apt install unzip
20 unzip latest.zip
21 ls
22 ls -a
23 ls -a
24 ls
25 pwd
26 unzip latest.zip
27 wget https://wordpress.org/latest.zip
28 ls
29 cd ~
30 wget https://wordpress.org/latest.zip
31 sudo mv latest.zip /var/www/html/
32 cd /var/www/html
33 sudo unzip latest.zip
34 ls
35 ls /var/www/html
36 cd wordpress
37 ls
```

ubuntu@ip-172-31-8-201: ~

```
13 ls -a
14 cd /var/www/html
15 ls
16 sudo rm index.html
17 ls
18 wget https://wordpress.org/latest.zip
19 sudo apt install unzip
20 unzip latest.zip
21 ls
22 ls-a
23 ls -a
24 ls
25 pwd
26 unzip latest.zip
27 wget https://wordpress.org/latest.zip
28 ls
29 cd ~
30 wget https://wordpress.org/latest.zip
31 sudo mv latest.zip /var/www/html/
32 cd /var/www/html
33 sudo unzip latest.zip
34 ls
35 ls /var/www/html
36 cd wordpress
37 ls
38 cd .
39 cd ..
40 ls
41 sudo mv /var/www/html/wordpress/* /var/www/html/
42 cd wordpress
43 ls
44 cd ..
45 ls
46 cd /var/www/html
47 ls
48 sudo rmdir /var/www/html/wordpress
49 ls
50 sudo chown -R www-data
51 sudo chown -R www-data:www-data /var/www/html
52 sudo chmod -R 775 /var/www/html
53 ls
54 sudo mv wp-config-sample.php wp-config.php
55 sudo nano wp-config.php
56 sudo systemctl restart apache2
57 cd /var/www/html
58 ls
59 sudo mv wordpress/* /var/www/html/
60 history
```

ubuntu@ip-172-31-8-201:~\$



Search



ENG
IN



12:15 PM
06-11-2024



```
ubuntu@ip-172-31-14-211: /var/www/html
inflating: wordpress/wp-admin/js/set-post-thumbnail.min.js
inflating: wordpress/wp-admin/js/postbox.min.js
inflating: wordpress/wp-admin/js/color-picker.js
inflating: wordpress/wp-admin/js/password-strength-meter.js
inflating: wordpress/wp-admin/js/customize-nav-menus.js
inflating: wordpress/wp-admin/js/editor-expand.js
inflating: wordpress/wp-admin/js/code-editor.min.js
inflating: wordpress/wp-admin/js/set-post-thumbnail.js
inflating: wordpress/wp-admin/options-permalink.php
inflating: wordpress/wp-admin/widgets.php
inflating: wordpress/wp-admin/setup-config.php
inflating: wordpress/wp-admin/install.php
inflating: wordpress/wp-admin/admin-header.php
inflating: wordpress/wp-admin/post-new.php
inflating: wordpress/wp-admin/themes.php
inflating: wordpress/wp-admin/options-reading.php
inflating: wordpress/wp-trackback.php
inflating: wordpress/wp-comments-post.php
ubuntu@ip-172-31-14-211: /var/www/html$ ls
latest.zip  wordpress
ubuntu@ip-172-31-14-211: /var/www/html$ ls /var/www/html
latest.zip  wordpress
ubuntu@ip-172-31-14-211: /var/www/html$ cd wordpress
ubuntu@ip-172-31-14-211: /var/www/html/wordpress$ ls
index.php  readme.html  wp-admin  wp-comments-post.php  wp-content  wp-includes  wp-load.php  wp-mail.php  wp-signup.php  xmlrpc.php
license.txt  wp-activate.php  wp-blog-header.php  wp-config-sample.php  wp-cron.php  wp-links-opml.php  wp-login.php  wp-settings.php  wp-trackback.php
ubuntu@ip-172-31-14-211: /var/www/html/wordpress$ cd ..
ubuntu@ip-172-31-14-211: /var/www/html$ ls
latest.zip  wordpress
ubuntu@ip-172-31-14-211: /var/www/html$ cd /var/www/html
ubuntu@ip-172-31-14-211: /var/www/html$ ls
latest.zip  wordpress
ubuntu@ip-172-31-14-211: /var/www/html$ sudo mv /var/www/html/wordpress/* /var/www/html/
ubuntu@ip-172-31-14-211: /var/www/html$ cd wordpress
ubuntu@ip-172-31-14-211: /var/www/html/wordpress$ ls
index.php  readme.html  wp-admin  wp-config-sample.php  wp-includes  wp-login.php  wp-signup.php
latest.zip  wordpress  wp-blog-header.php  wp-content  wp-links-opml.php  wp-mail.php  wp-trackback.php
license.txt  wp-activate.php  wp-comments-post.php  wp-cron.php  wp-load.php  wp-settings.php  xmlrpc.php
ubuntu@ip-172-31-14-211: /var/www/html/wordpress$ cd ..
ubuntu@ip-172-31-14-211: /var/www/html$ ls
index.php  readme.html  wp-admin  wp-config-sample.php  wp-includes  wp-login.php  wp-signup.php
latest.zip  wordpress  wp-blog-header.php  wp-content  wp-links-opml.php  wp-mail.php  wp-trackback.php
ubuntu@ip-172-31-14-211: /var/www/html$ sudo rmdir /var/www/html/wordpress
ubuntu@ip-172-31-14-211: /var/www/html$ ls
index.php  license.txt  wp-activate.php  wp-blog-header.php  wp-config-sample.php  wp-cron.php  wp-links-opml.php  wp-login.php  wp-settings.php  wp-trackback.php  xmlrpc.php
latest.zip  readme.html  wp-admin  wp-comments-post.php  wp-content  wp-includes  wp-load.php  wp-mail.php  wp-signup.php  xmlrpc.php
ubuntu@ip-172-31-14-211: /var/www/html$ sudo chown -R www-data
chown: missing operand after 'www-data'
Try 'chown --help' for more information.
ubuntu@ip-172-31-14-211: /var/www/html$ sudo chown -R www-data /var/www/html
```

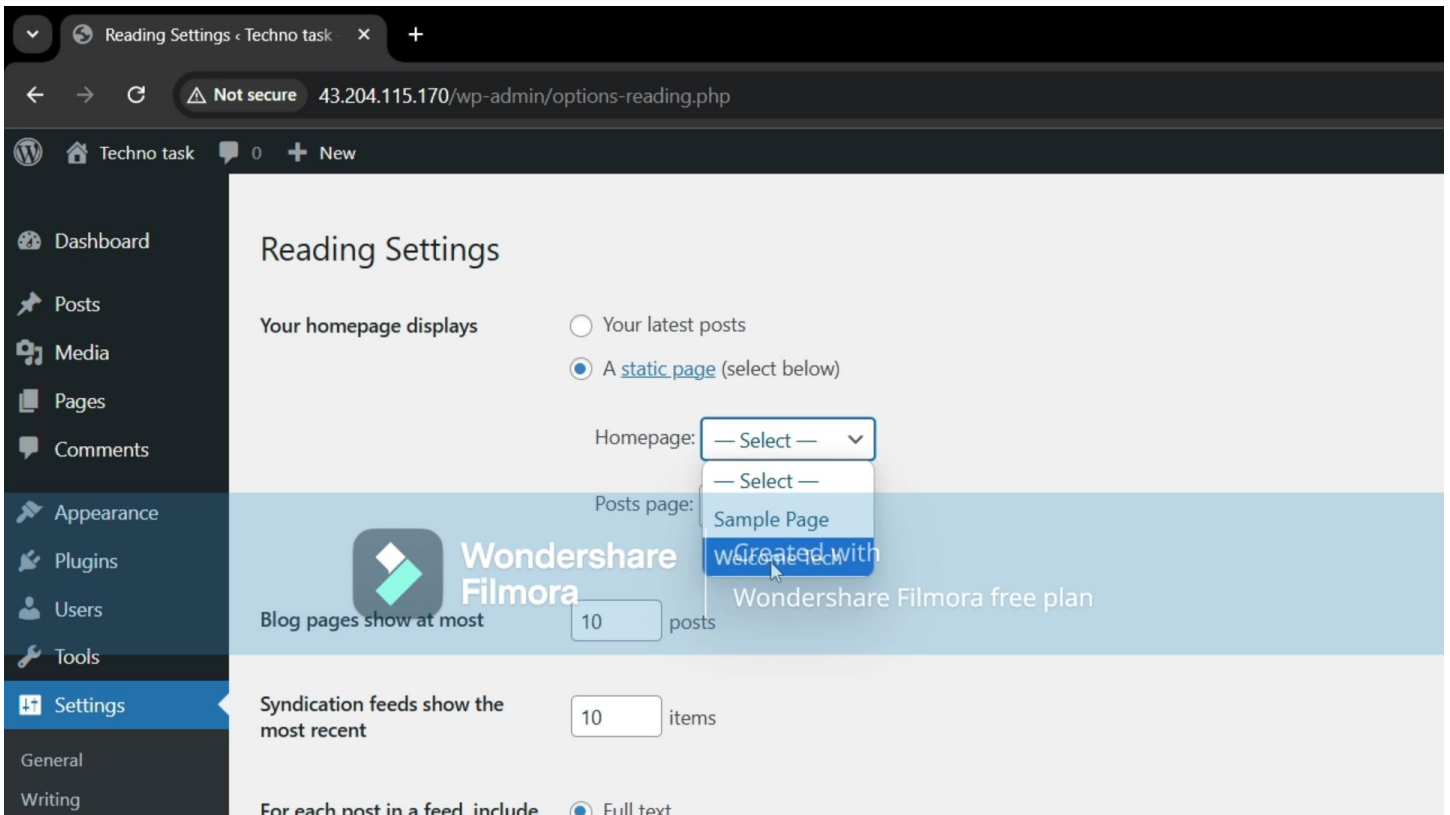
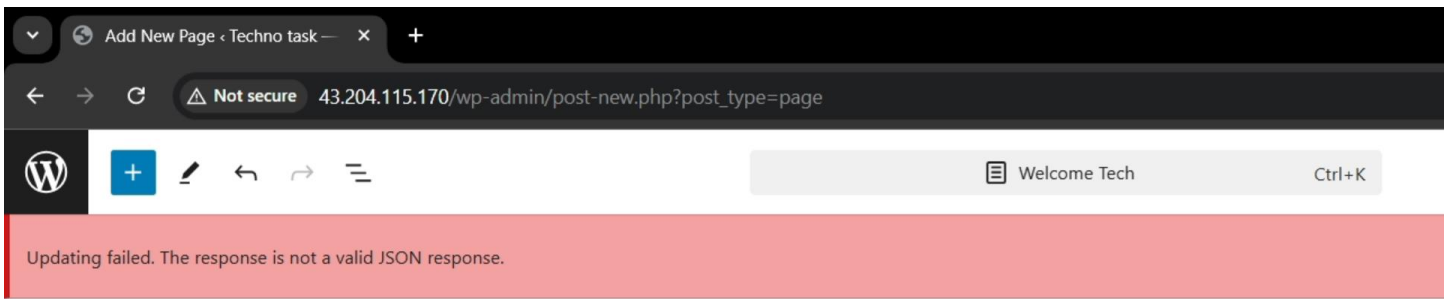
```
ubuntu@ip-172-31-8-201: /var/www/html
GNU nano 7.2 wp-config.php *
i<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the installation.
 * You don't have to use the website, you can copy this file to "wp-config.php"
 * and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * Database settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://developer.wordpress.org/advanced-administration/wordpress/wp-config/
 *
 * @package WordPress
 */

// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress_db' );

/** Database username */
define( 'DB_USER', 'wp_user' );

/** Database password */
define( 'DB_PASSWORD', 'Abhi' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );
```

Dashboard < Techno task — Wo

Techno task

+

← → ↻

Not secure 43.204.115.170

Techno task

Edit site

0

+ New

Edit Page

Techno task



Wondershare
Filmora

Created with

Welcome Tech

Wondershare Filmora free plan

This is My GitHub URL:- <https://github.com/Abhisheksgp>

Here I have Installed a Mysql Server, Apache2, php and configure all in the single Ec2 instance I have Explained in Next Microservice Architecture There we will connect by common same Vpc Here there will be not there Remaining all same

2).Microservices Architecture on AWS: Deploying WordPress and MySQL on Separate EC2 Instances:-

Requirements:- Same as Above Ec2 Instances Here Create Two Ec2 Instances:-

*) For Mysql (Here sss,Mysql/Aurora same as above)

*) Wordpress(Here ssh, http, https, Mysql/Aurora)
connect this Through Putty SSH one by one Both

*)And install And configure the Mysql And Create DataBase Name , UserName, Password, And Do FLUSH PRIVILEGES and Exit Mysql Server and Assign Security Configuration for this And Assign The Permissions For (RWE).. Here
Configure Mysql to allow external connections
like sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
here Change the Ip Address of example Bind-address =127.0.0.34 to 0.0.0.0 to allow connection from other services

*)For Wordpress Instance Insatll and configure Apache2, Php for wordpress and here configure Wodpress To connect To the database: like wp-config.php
here you can use Nano Editor/Vim/vi..
i have used nano editor and change the Same What you have created in Mysql Server Database_Name, DB_user, DB_Password, DB_Host, And Ctrl+O to save and Confirm the Name for your File and exit (Ctrl+x)

*)After Insatlling apache2, Do
Sudo systemctl enable apache2
sudo systemctl restart apache2

For secure The mysql and root password confirming Use
sudo mysql_secure_installation

Finally go To ec2 instance of Wordpress And copy Ip address and paste to Any browser It will open WordPress if any errors troubleshoot by database connection Check the DB_name, user, password, and host as private address for Microservices and for Monolithic use localhost later do login and create own homepage and save and open in browser

ubuntu@ip-172-31-9-174: ~

mysql> exit

Bye

ubuntu@ip-172-31-9-174:~\$ sudo mysql -u root -P

mysql: [ERROR] mysql: option '-P' requires an argument.

ubuntu@ip-172-31-9-174:~\$ sudo mysql -u root -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 11

Server version: 8.0.39-0ubuntu0.24.04.2 (Ubuntu)

Copyright (c) 2000, 2024, 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> CREATE DATABASE wordpress;

Query OK, 1 row affected (0.01 sec)

mysql> CREATE USER 'wp_users'@'%' IDENTIFIED BY 'Abhipassword@80882';

Query OK, 0 rows affected (0.11 sec)

mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'wp_users'@'%';

Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;

Query OK, 0 rows affected (0.00 sec)

mysql> sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf

-> sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf

-> ^C

mysql> ^C

mysql> exit

Bye

ubuntu@ip-172-31-9-174:~\$ sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf

ubuntu@ip-172-31-9-174:~\$ sudo systemctl restart mysql

ubuntu@ip-172-31-9-174:~\$ history

- 1 sudo apt update
- 2 sudo apt install mysql-server
- 3 sudo mysql_secure_installation
- 4 sudo mysql -u root -p
- 5 sudo mysql -u root -P
- 6 sudo mysql -u root -p
- 7 sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
- 8 sudo systemctl restart mysql
- 9 history

ubuntu@ip-172-31-9-174:~\$



Search



ENG
IN



08:42 PM
06-11-2024



Not secure 3.110.155.32

Welcome To Intern



Wondershare
Filmora

Cloud Computing

Created with

Wondershare Filmora free plan

The Future For Cloud is Increasing so Software Development And services with a passion for innovation also increasing so I am Looking Forward to

Enhance And upskill in the Cloud And DevOps Looking Forward to work in Company like TECHPLEMENT

I AM ABHISHEK.K