

Deploy Application in Monolithic and Microservices Architecture

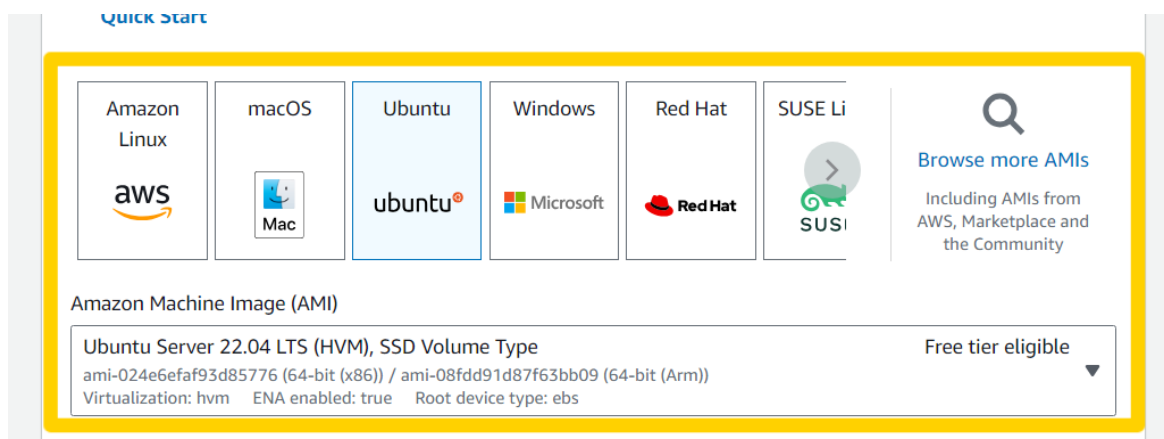
❖ Task: Deploy application in monolithic and microservices architecture

❖ For monolithic

1. In the EC2 Dashboard, click the "Launch Instance" button to start the instance creation process.
2. Choose an Amazon Machine Image (AMI)

In the "Application and OS Images" :

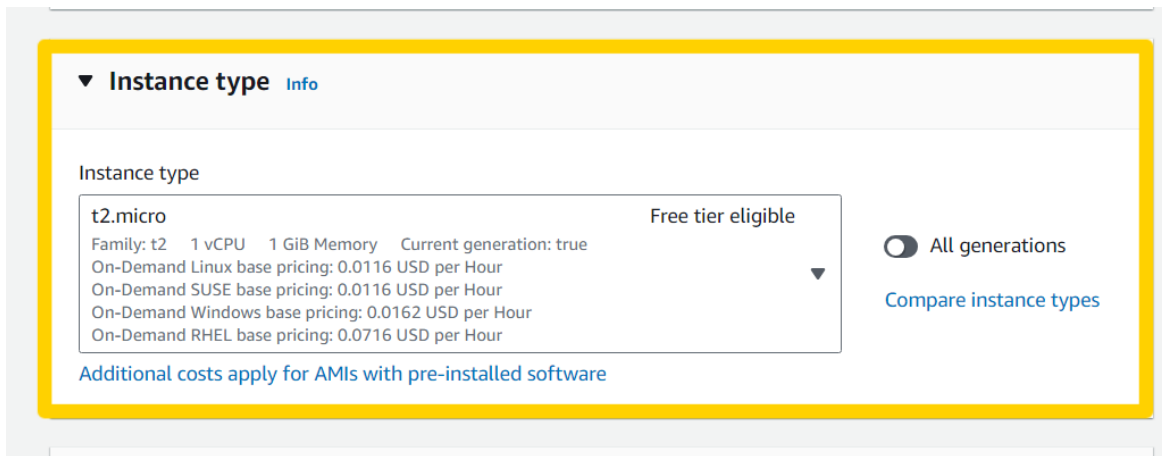
- Choose an Ubuntu AMI.
- Select the Ubuntu AMI that matches your region and desired version (e.g., Ubuntu Server 22.04 LTS).



3. Choose an Instance Type

In the "Choose an Instance Type" step:

- Select the instance type as "t2.micro"



4. In the key pair section , create a key pair(Keypair Type: RSA, File format:.pem) and download the key pair file (e.g., `your-key-pair.pem`) or choose an existing key pair. *This file is essential for SSH access to your instance*

5. Under Network Settings, **Configure Security Group**

Allow inbound traffic:

- HTTP (Port 80) from 0.0.0.0/0
- HTTPS (Port 443) from 0.0.0.0/0
- SSH (Port 22) from anywhere or a trusted IP range.

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	WordPress and MySQL Instance	i-0e4a6828b812b4070	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a

Instance: i-0e4a6828b812b4070 (WordPress and MySQL Instance)					
Name	Security group rule ID	Port range	Protocol	Source	Security groups
-	sgr-0d8ed45b0437e52a1	80	TCP	0.0.0.0/0	launch-wizard-2
-	sgr-01084498cc791fd45	22	TCP	0.0.0.0/0	launch-wizard-2
-	sgr-0a5a584f0812d4e18	443	TCP	0.0.0.0/0	launch-wizard-2

6. You can leave remaining settings at their default values(ie Configure storage and Advanced details)

6. **Review and Launch**

- Review your instance configuration to ensure it's correct.
- Click the "Launch" button.

8. Access Your Ubuntu EC2 Instance

- Once your instance is running (it may take a few minutes to initialize), select it in the list of instances.

Instance state = running X		Clear filters		< 1 > ⓘ			
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	WordPress and MySQL Instance	i-0e4a6828b812b4070	Running ⓘ	t2.micro	2/2 checks passed	No alarms +	us-east-2a

- Click the "Connect" button at the top.
- Follow the instructions provided to SSH into your Ubuntu instance using the key pair you downloaded earlier. eg: `ssh -i <your-key-pair.pem> ubuntu@<instance_public_IP>`

Connect to instance ⓘ

Connect to your instance i-0e4a6828b812b4070 (WordPress and MySQL Instance) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-0e4a6828b812b4070 (WordPress and MySQL Instance)

Connection Type

☒ Connect using EC2 Instance Connect
 Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.

☐ Connect using EC2 Instance Connect Endpoint
 Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address

3.146.178.227

User name

Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.

ubuntu

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

Cancel

Connect

9. Install MySQL and create a Database for WordPress:

- SSH into the instance:

```
ssh -i <your-key.pem> ubuntu@<instance-public-ip>
```

- Update the package list and install MySQL Server:

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

- Secure your MySQL installation by running:

```
sudo mysql_secure_installation
```

- Follow the prompts to set a root password and answer the security questions.
- Create a MySQL database for WordPress:

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

- Inside MySQL, run: To create a user with permissions for the database

```
CREATE DATABASE wordpress;  
#set a strong password in <enter-your-password>  
CREATE USER 'wordpress'@'localhost' IDENTIFIED BY '<enter-your-password>';  
GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpress'@'localhost';  
FLUSH PRIVILEGES;  
EXIT;
```

10. Install WordPress

- Install Apache with `sudo apt-get install apache2 -y`
- Install PHP with `sudo apt-get install php libapache2-mod-php php-mysql -y`
- Download and extract the latest WordPress: `wget https://wordpress.org/latest.tar.gz`
extract it `tar -xvzf latest.tar.gz`
- Move the WordPress files to the Apache web directory: `sudo mv wordpress/* /var/www/html/`

11. Configure WordPress database details by editing the WordPress configuration file located at /var/www/html/wp-config.php(rename the wp-sample.php with wp-config.php) and add the database details

12. Access your WordPress site URL (http://<instance-public-ip>/wp admin or access by the instance public ip by removing the index.html file from /var/www/html/ using `sudo rm -rf index.html`) and you get following the setup wizard. Enter the details and click on submit

Below you should enter your database connection details. If you are not sure about these, contact your host.

Database Name:
The name of the database you want to use with WordPress.

Username:
Your database username.

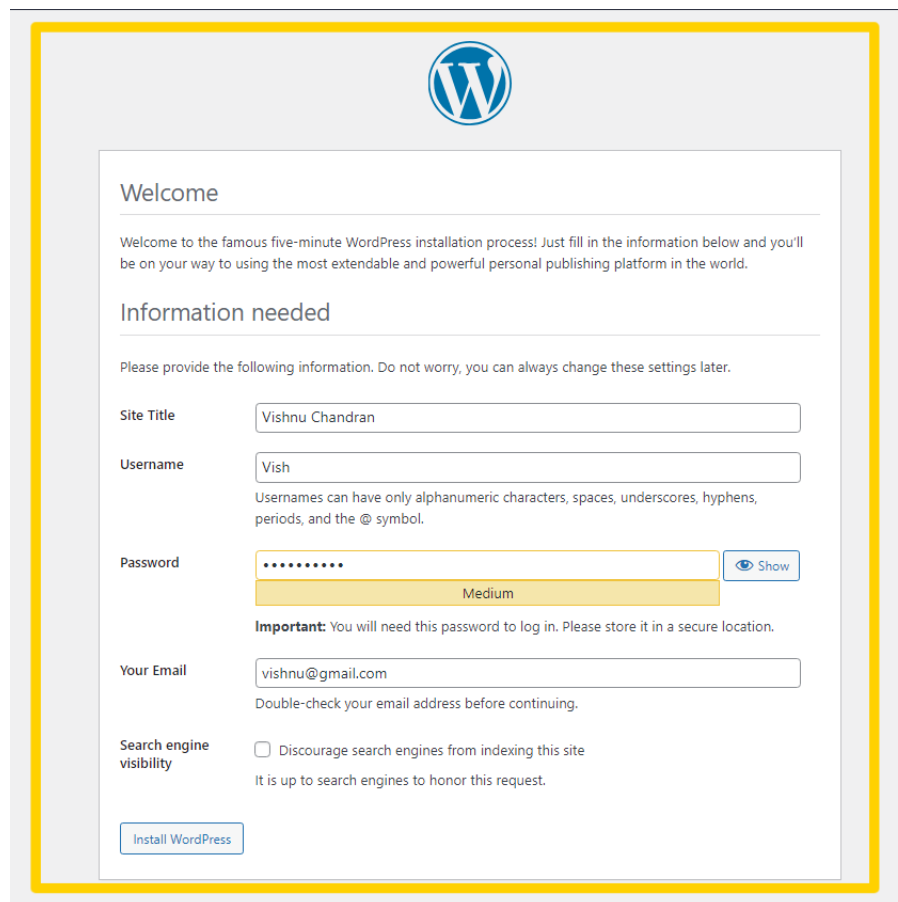
Password: [Show](#)
Your database password.

Database Host:
You should be able to get this info from your web host, if localhost does not work.

Table Prefix:
If you want to run multiple WordPress installations in a single database, change this.

[Submit](#)

13. You will get below screen and enter the required information, click on Install WordPress



The image shows the WordPress 'Welcome' screen during the installation process. At the top is the WordPress logo. Below it, the heading 'Welcome' is followed by a paragraph: 'Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.'

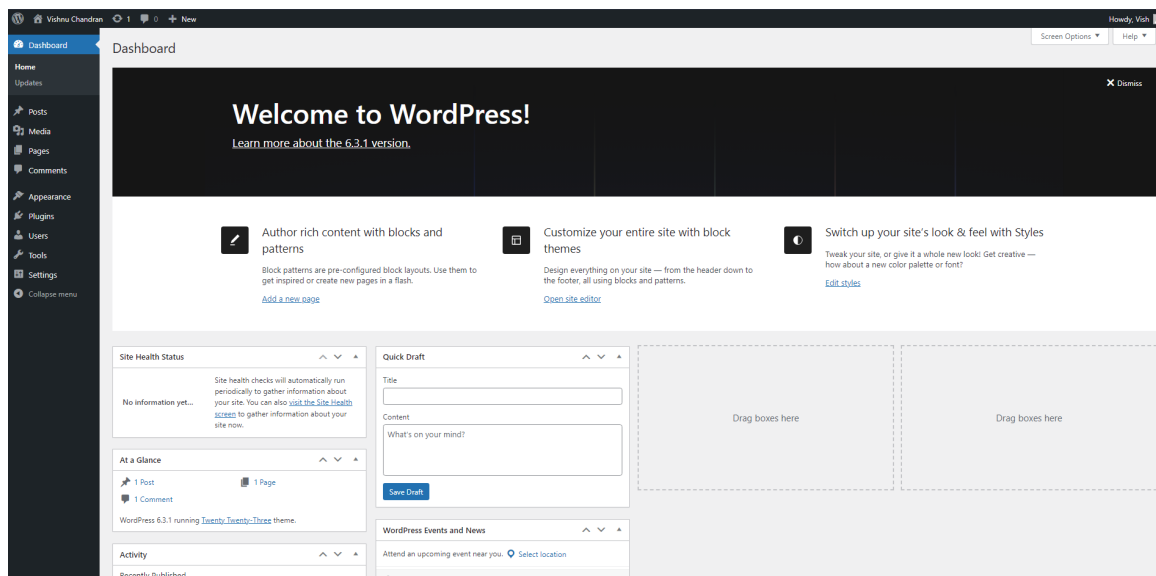
The next section is 'Information needed', with a subtext: 'Please provide the following information. Do not worry, you can always change these settings later.'

The form contains the following fields and options:

- Site Title:** A text input field containing 'Vishnu Chandran'.
- Username:** A text input field containing 'Vish'. Below it, a note states: 'Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.'
- Password:** A password input field with masked characters '.....'. To its right is a 'Show' button. Below the field is a strength indicator bar showing 'Medium' strength.
- Important:** A text block stating: 'You will need this password to log in. Please store it in a secure location.'
- Your Email:** A text input field containing 'vishnu@gmail.com'. Below it, a note states: 'Double-check your email address before continuing.'
- Search engine visibility:** A checkbox labeled 'Discourage search engines from indexing this site'. Below it, a note states: 'It is up to search engines to honor this request.'

At the bottom left of the form is a blue button labeled 'Install WordPress'.

14. Log into WordPress.



The image shows the WordPress Dashboard after logging in. The top navigation bar includes the user name 'Vishnu Chandran', a 'New' button, and a 'Howdy, Vish' greeting. The main content area features a large 'Welcome to WordPress!' banner with a link to 'Learn more about the 6.3.1 version.'

Below the banner are three featured sections:

- Author rich content with blocks and patterns:** Includes a description of block patterns and a link to 'Add a new page'.
- Customize your entire site with block themes:** Includes a description of block themes and a link to 'Open site editor'.
- Switch up your site's look & feel with Styles:** Includes a description of the Styles interface and a link to 'Edit styles'.

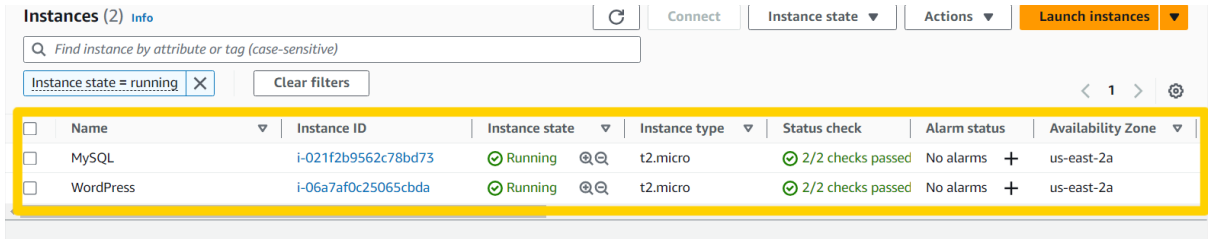
The dashboard is organized into several widgets:

- Site Health Status:** Shows site health information and a link to 'Site Health'.
- At a Glance:** Displays site statistics: 1 Post, 1 Page, 1 Comment, and the current theme 'Twenty Twenty-Three'.
- Activity:** A section for recent activity.
- Quick Draft:** A form for creating a new draft, including fields for Title and Content, and a 'Save Draft' button.
- WordPress Events and News:** A section for upcoming events and news.

On the left side, there is a sidebar menu with links to Home, Updates, Posts, Media, Pages, Comments, Appearance, Plugins, Users, Tools, and Settings.

❖ For Microservices Architecture

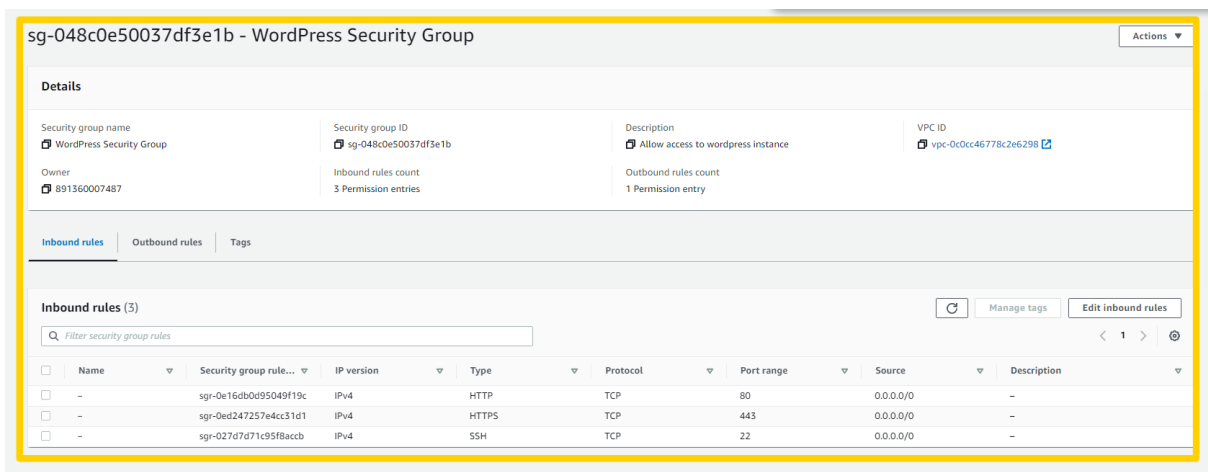
1. **Launch two EC2 instances:** Choose the **t2-micro** instance type and **Ubuntu** AMI for both. Make sure to create a new key pair and download it to your local machine.



The screenshot shows the AWS Management Console 'Instances' page. Two instances are listed: MySQL and WordPress. Both are in a 'Running' state, using the 't2.micro' instance type. The MySQL instance has ID i-021f2b9562c78bd73 and the WordPress instance has ID i-06a7af0c25065cbda. Both have passed status checks and have no alarms.

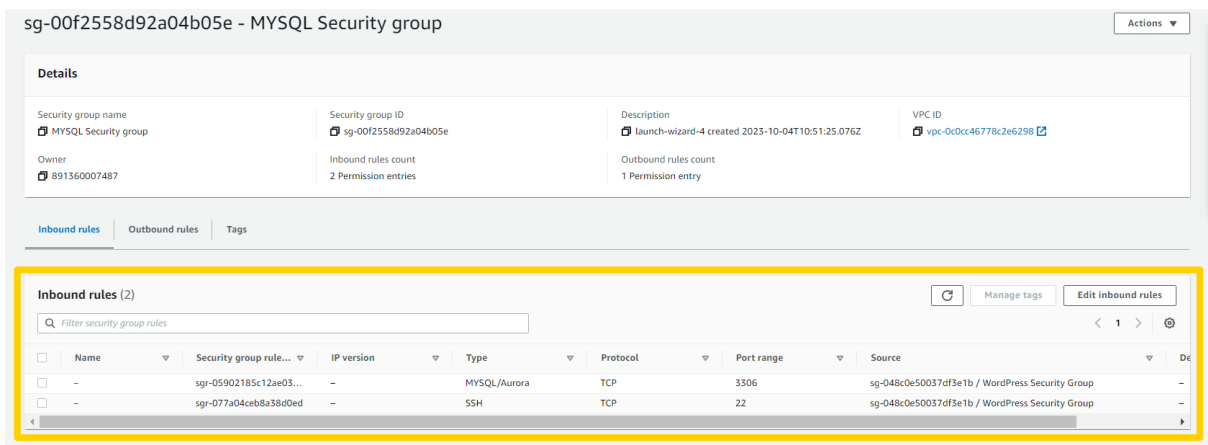
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	MySQL	i-021f2b9562c78bd73	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a
<input type="checkbox"/>	WordPress	i-06a7af0c25065cbda	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a

2. **Create two Security Groups:** One for the WordPress instance and one for the MySQL instance. For the WordPress instance, allow inbound traffic on ports 22, 80, and 443. For the MySQL instance, allow inbound traffic on port 3306 from the WordPress instance's security group.



The screenshot shows the configuration for the 'WordPress Security Group' (sg-048c0e50037df3e1b). It includes details like the security group name, ID, description, and VPC ID. The 'Inbound rules' tab is selected, showing three rules: HTTP on port 80, HTTPS on port 443, and SSH on port 22, all allowing traffic from 0.0.0.0/0.

	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sgr-0e16db0d95049f19c	IPv4	HTTP	TCP	80	0.0.0.0/0	-
<input type="checkbox"/>	-	sgr-0ed247257e4cc31d1	IPv4	HTTPS	TCP	443	0.0.0.0/0	-
<input type="checkbox"/>	-	sgr-027d7d71c95f8accb	IPv4	SSH	TCP	22	0.0.0.0/0	-



The screenshot shows the configuration for the 'MySQL Security Group' (sg-00f2558d92a04b05e). It includes details like the security group name, ID, description, and VPC ID. The 'Inbound rules' tab is selected, showing two rules: MySQL/Aurora on port 3306 from the WordPress Security Group, and SSH on port 22 from the WordPress Security Group.

	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sgr-05902185c12ae03...	-	MySQL/Aurora	TCP	3306	sg-048c0e50037df3e1b / WordPress Security Group	-
<input type="checkbox"/>	-	sgr-077a04ceb8a38d0ed	-	SSH	TCP	22	sg-048c0e50037df3e1b / WordPress Security Group	-

3. **Install WordPress on one instance and MySQL on the other:**

- SSH into your WordPress instance and install Apache and PHP as described above.
- SSH into your MySQL instance and install MySQL as described above.
- Configure WordPress to connect to the MySQL database on the other instance.

4. **Create a welcome page: Log into your WordPress site and create a new page that will serve as the homepage.** Create a welcome page. Pages>new>edit and publish it

