

Techplement - Week 1 Tasks

I have attached the screenshots related to my works at the end of the document.

Overview

This document provides a detailed description of the steps I followed to deploy WordPress and MySQL on AWS using EC2 instances in both monolithic and microservices architectures. Additionally, it includes instructions on setting up a welcome page in WordPress.

Prerequisites

Before starting, I ensured the following prerequisites were met:

1. Created a free-tier AWS account.
2. Installed AWS CLI on my local machine and configured it with my AWS credentials.
3. Generated an SSH key pair for secure access to the EC2 instances.

Task Breakdown

Monolithic Architecture

Step 1: Launch an EC2 Instance

1. **Open the AWS Management Console** and navigated to the EC2 dashboard.
2. **Launch Instance:**
 - o **AMI:** Selected an Ubuntu AMI (ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-2023).
 - o **Instance Type:** Chose t2.micro.
 - o **Key Pair:** Selected my previously created key pair.
 - o **Security Group:** Created a new security group with the following rules:
 - Allowed SSH (port 22) from my IP.
 - Allowed HTTP (port 80) from anywhere.
3. Launched the instance.
4. Created Elastic IP and associated with the instance to ensure the IP doesn't change after every reboot.

Step 2: Install WordPress and MySQL

1. Connected to the EC2 instance via SSH:
2. `ssh -i /key-pair.pem ubuntu@3.209.28.142`
3. Updated the package list:

4. `sudo apt update`
5. Installed Apache, MySQL, PHP, and related packages:
6. `sudo apt install apache2 mysql-server php libapache2-mod-php php-mysql -y`
7. Secured MySQL installation:
8. `sudo mysql_secure_installation`
9. Created a MySQL database and user for WordPress:
10. `sudo mysql -u root -p`
11. `CREATE DATABASE wordpress;`
12. `GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpressuser'@'localhost' IDENTIFIED BY 'password';`
13. `FLUSH PRIVILEGES;`
14. `EXIT;`
15. Installed WordPress:
16. `cd /tmp`
17. `wget https://wordpress.org/latest.tar.gz`
18. `tar -xzf latest.tar.gz`
19. `sudo mv wordpress /var/www/html/`
20. `sudo chown -R www-data:www-data /var/www/html/wordpress`
21. `sudo chmod -R 755 /var/www/html/wordpress`
22. Configured Apache:
23. `sudo nano /etc/apache2/sites-available/wordpress.conf`

Added the following content:

```
<VirtualHost *:80>
    DocumentRoot /var/www/html/wordpress
</VirtualHost>
```

Enabled the configuration and rewrite module:

```
sudo a2ensite wordpress.conf
sudo a2enmod rewrite
sudo systemctl restart apache2
```

24. Completed the WordPress installation by navigating to `http:// 3.209.28.142` in my web browser and following the setup instructions.

Microservices Architecture

Step 1: Launch EC2 Instances

Repeated the instance launch process for two instances, one for WordPress and one for MySQL, with the same configurations but different names to identify them.

Created Elastic IP and associated with the instances to ensure the IP doesn't change after every reboot.

Step 2: Configure Security Groups

1. MySQL Instance Security Group:

- Allowed SSH (port 22) from my IP.
- Allowed MySQL (port 3306) from the WordPress instance's private IP address.

2. WordPress Instance Security Group:

- Allowed SSH (port 22) from my IP.
- Allowed HTTP (port 80) from anywhere.

Step 3: Install MySQL on MySQL Instance

1. Connected to the MySQL EC2 instance via SSH:
2. `ssh -i /path/to/your-key-pair.pem ubuntu@54.197.61.78`
3. Updated the package list and installed MySQL:
4. `sudo apt update`
5. `sudo apt install mysql-server -y`
6. Secured MySQL installation:
7. `sudo mysql_secure_installation`
8. Created a MySQL database and user for WordPress:
9. `sudo mysql -u root -p`
10. `CREATE DATABASE wordpress;`
11. `GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpressuser'@'wordpress-ec2-private-ip' IDENTIFIED BY 'password'; *`
12. `FLUSH PRIVILEGES;`
13. `EXIT;`
14. Updated MySQL configuration to allow remote access:
15. `sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf`

Changed bind-address to 0.0.0.0:

```
bind-address = 0.0.0.0
```

Restarted MySQL:

```
sudo systemctl restart mysql
```

Step 4: Install WordPress on WordPress Instance

1. Connected to the WordPress EC2 instance via SSH:
2. `ssh -i /key-pair.pem ubuntu@54.156.198.64`
3. Updated the package list:
4. `sudo apt update`

5. Installed Apache, PHP, and related packages:

```
6. sudo apt install apache2 php libapache2-mod-php php-mysql -y
```

7. Installed WordPress:

```
8. cd /tmp
```

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

```
10. tar -xzf latest.tar.gz
```

```
11. sudo mv wordpress /var/www/html/
```

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

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

14. Configured Apache:

```
15. sudo nano /etc/apache2/sites-available/wordpress.conf
```

Added the following content:

```
<VirtualHost *:80>
    DocumentRoot /var/www/html/wordpress
</VirtualHost>
```

Enabled the configuration and rewrite module:

```
sudo a2ensite wordpress.conf
sudo a2enmod rewrite
sudo systemctl restart apache2
```

16. Configured WordPress to connect to MySQL:

```
17. sudo nano /var/www/html/wordpress/wp-config.php
```

Updated the database settings:

```
define('DB_NAME', 'wordpress');
define('DB_USER', 'wordpressuser');
define('DB_PASSWORD', 'password');
define('DB_HOST', 'mysql-ec2-private-ip');
```

18. Completed the WordPress installation by navigating to [http:// 54.156.198.64](http://54.156.198.64) in my web browser and following the setup instructions.

Create a Welcome Page in WordPress

1. Logged into the WordPress admin dashboard.
2. Created a new page named "Welcome".
3. Set the "Welcome" page as the homepage:
 - o Went to Settings > Reading.

- Selected A static page and chose "Welcome" from the dropdown menu.

Screenshots:

Name and tags [Info](#)

Name

Techplement-Microservice-MySQL

Add additional tags

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Q

Browse more AMIs

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-04b70fa74e45c3917 (64-bit (x86)) / ami-0eac975a54dfee8cb (64-bit (Arm))

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

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Techplement-1

Create new key pair

▼ Network settings Info

Edit

Network Info

vpc-02942a3adc6425744

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

We'll create a new security group called 'launch-wizard-4' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance

My IP
116.73.126.163/32

☒ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

×

✔ Successfully initiated termination of i-058bb49a13a94c380

×

Instances (1/4) Info

↻ Connect Instance state ▼ Actions ▼ Launch instances ▼

Find Instance by attribute or tag (case-sensitive) All states ▼ < 1 > ⚙

| | Name ↗ | Instance ID | Instance state ▲ | Instance type ▼ | Status check | Alarm status | Availability |
|--------------------------|------------------------------------|---------------------|------------------|-----------------|---------------------|---------------|--------------|
| <input type="checkbox"/> | Techplement-Microservice-Wordpress | i-00a28140fe4af7c57 | ⌚ Pending 🔍 🔍 | t2.micro | – | View alarms + | us-east-1a |
| <input type="checkbox"/> | Techplement-Wordpress-Monolithic | i-0ba38c301c6ca7811 | ✔ Running 🔍 🔍 | t2.micro | ✔ 2/2 checks passed | View alarms + | us-east-1a |
| <input type="checkbox"/> | Techplement-Microservice-MySQL | i-05131b64e9bd3915f | ✔ Running 🔍 🔍 | t2.micro | ⌚ Initializing | View alarms + | us-east-1a |

h

[Alt+S]

N. Virginia

SandBoom_7

Elastic IP addresses (1/2)

Find resources by attribute or tag

Name

Allocated IPv4 addr...

Type

Allocation ID

-

3.209.28.142

Public IP

eipalloc-0d4

-

54.156.198.64

Public IP

eipalloc-019

Allocate Elastic IP address

View details

Release Elastic IP addresses

Associate Elastic IP address

Disassociate Elastic IP address

Update reverse DNS

Enable transfers

Disable transfers

Accept transfers

View IP address usage and recommendations to release unused IPs with [Public IP insights](#)

54.156.198.64

Summary

Tags

Elastic IP address: 54.156.198.64

Resource type

Choose the type of resource with which to associate the Elastic IP address.

Instance

Network interface

If you associate an Elastic IP address with an instance that already has an Elastic IP address associated, the previously associated Elastic IP address will be disassociated, but the address will still be allocated to your account. [Learn more](#)

If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address.

Instance

Choose an instance

Private IP address

The private IP address with which to associate the Elastic IP address.

Choose a private IP address

Reassociation

Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.

Allow this Elastic IP address to be reassociated

Cancel

Associate

| | | | |
|-------------------------|---------------|---------------|---|
| ec2-54-156-198-64.co... | 54.156.198.64 | 54.156.198.64 | - |
|-------------------------|---------------|---------------|---|

```
ubuntu@ip-172-31-28-254:~$ sudo apt update
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 Packages [1401 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main Translation-en [513 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:9 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [182 kB]
```

```
ubuntu@ip-172-31-28-254:~$ sudo apt install apache2 mysql-server php libapache2-mod-php php-mysql -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapache2-mod-php8.3 libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 libcgf-fast-perl
  libcgf-pm-perl libclone-perl libencode-locale-perl libevent-pthreads-2.1-7t64 libfcgi-bin libfcgi-perl libfcgi0t64 libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblua5.4-0 liblwp-mediatypes-perl libmecab2 libprotobuf-lite32t64
```

```
ubuntu@ip-172-31-28-254:~$ sudo mysql_secure_installation
```

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords and improve security. It checks the strength of password and allows the users to set only those passwords which are secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: ☐

VALIDATE PASSWORD COMPONENT can be used to test passwords and improve security. It checks the strength of password and allows the users to set only those passwords which are secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER_USER" command.
See <https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management> for more information.

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : ☐

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) :
... skipping.

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) :
... skipping.

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) :
... skipping.

```
ubuntu@ip-172-31-28-254:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.37-0ubuntu0.24.04.1 (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>
```

```
ubuntu@ip-172-31-28-254:~$ 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: 8.0.37-0ubuntu0.24.04.1 (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 wp;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON wp.* TO 'Techplement-Wordpress'@'localhost' IDENTIFIED BY 'password';
```

SHANTHOSH KUMAR E - CLOUD(AWS) intern

| Instance ID | Instance Name | State | Instance Type | Availability Zone | Subnet |
|---------------------|------------------------------------|------------|---------------|-------------------|---|
| i-00a28140fe4af7c57 | Techplement-Microservice-Wordpress | Running | t2.micro | us-east-1a | ec2-204-236-199-112.compute-1.amazonaws.com |
| i-058bb49a13a94c380 | Techplement-Microservice-MySQL | Terminated | t2.micro | us-east-1a | - |

i-00a28140fe4af7c57 (Techplement-Microservice-Wordpress)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

▼ Instance summary [Info](#)

| | | |
|---|---|---|
| Instance ID i-00a28140fe4af7c57 (Techplement-Microservice-Wordpress) | Public IPv4 address 204.236.199.112 open address | Private IPv4 addresses 172.31.25.165 |
| IPv6 address - | Instance state Running | Public IPv4 DNS ec2-204-236-199-112.compute-1.amazonaws.com open address |
| Hostname type IP name: ip-172-31-25-165.ec2.internal | Private IP DNS name (IPv4 only) ip-172-31-25-165.ec2.internal | |

```
mysql> CREATE USER 'Techplement-Wordpress'@54.197.61.78 identified by 'Testpassword@123';
Query OK, 0 rows affected (0.02 sec)

mysql> use wp;
Database changed
mysql> Grant all privileges on wp.* to 'Techplement-Wordpress'@54.197.61.78;
Query OK, 0 rows affected (0.00 sec)

mysql> Grant all privileges on wp.* to 'Techplement-Wordpress'@54.197.61.78;^C
mysql> exit
Bye
```

```
ubuntu@ip-172-31-25-165:/var/www/html$ sudo chown -R www-data:www-data /var/www/html/wordpress
ubuntu@ip-172-31-25-165:/var/www/html$ sudo chmod -R 755 /var/www/html/wordpress
ubuntu@ip-172-31-25-165:/var/www/html$ sudo nano /etc/apache2/sites-available/wordpress.conf
ubuntu@ip-172-31-25-165:/var/www/html$ cd /etc/apache2/sites-available/
ubuntu@ip-172-31-25-165:/etc/apache2/sites-available$ ls
000-default.conf  default-ssl.conf
ubuntu@ip-172-31-25-165:/etc/apache2/sites-available$ nano 000-default.conf
ubuntu@ip-172-31-25-165:/etc/apache2/sites-available$ sudo nano 000-default.conf
ubuntu@ip-172-31-25-165:/etc/apache2/sites-available$ sudo systemctl restart apache2
```

Welcome to WordPress. Before getting started, you will need to know the following items.

1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

This information is being used to create a `wp-config.php` file. **If for any reason this automatic file creation does not work, do not worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`.** Need more help? [Read the support article on wp-config.php.](#)

In all likelihood, these items were supplied to you by your web host. If you do not have this information, then you will need to contact them before you can continue. If you are ready...

Let's go!

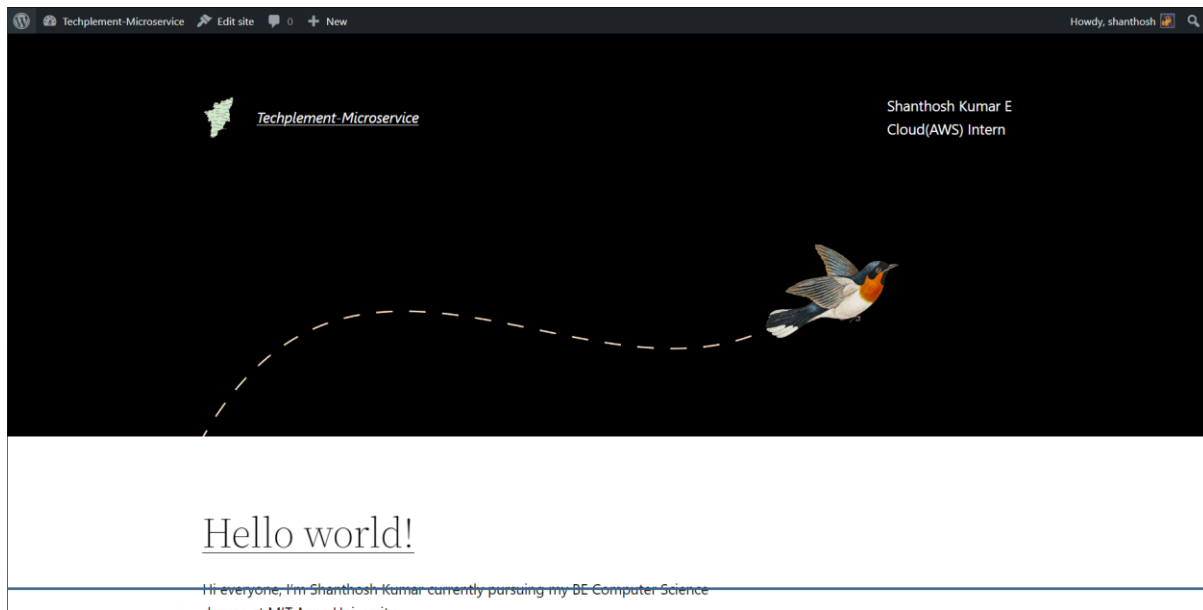
Not secure 54.156.198.64/wp-admin/setup-config.php



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

| | |
|--|---|
| Database Name | <input type="text" value="wp"/> |
| The name of the database you want to use with WordPress. | |
| Username | <input type="text" value="Techplement-Wordpress"/> |
| Your database username. | |
| Password | <input type="password" value="....."/> Show |
| Your database password. | |
| Database Host | <input type="text" value="54.197.61.78"/> |
| You should be able to get this info from your web host, if localhost does not work. | |
| Table Prefix | <input type="text" value="wp_"/> |
| If you want to run multiple WordPress installations in a single database, change this. | |
| <input type="button" value="Submit"/> | |

SHANTHOSH KUMAR E - CLOUD(AWS) intern



***Note:** I have concealed the private information like my private ip address and have placed placeholders instead.