

# Infotrixxs - Cloud AWS Intern

Name: Pavan Valmiki

Github: <https://github.com/valmiki58>

LinkedIn: <https://www.linkedin.com/in/pavan-valmiki-b0a576240/>

Task 1: Monolithic:- 1 EC2 instance, deploying Wordpress and MySQL on the same instance

Steps:1

Login to aws account.

Create 1 instance

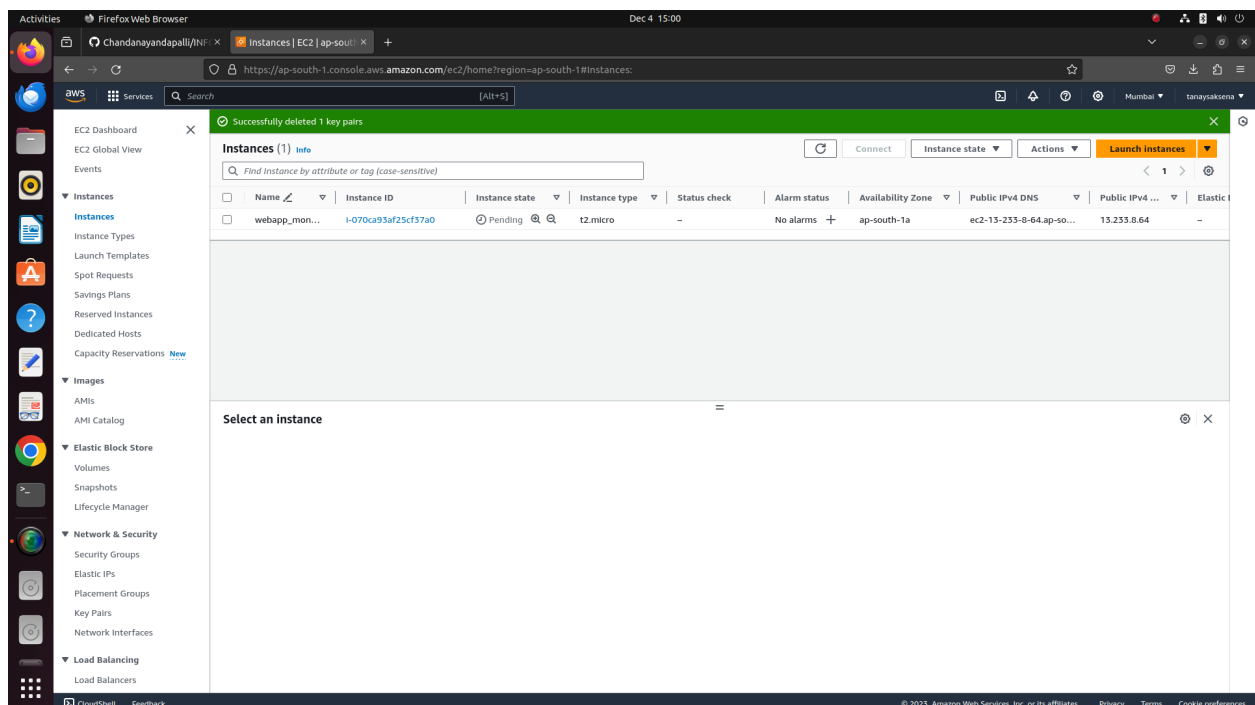
Instance Specifications

Name: webapp\_mono

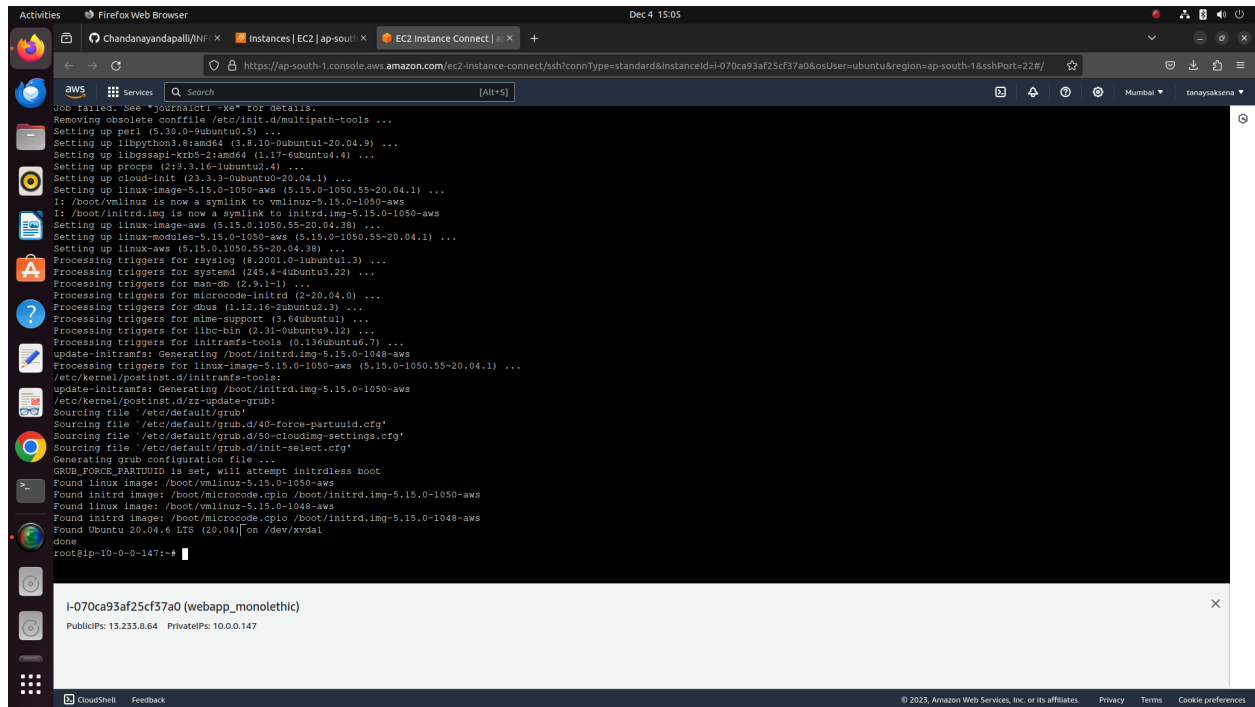
OS: Ubuntu

Instance Type: t2.micro

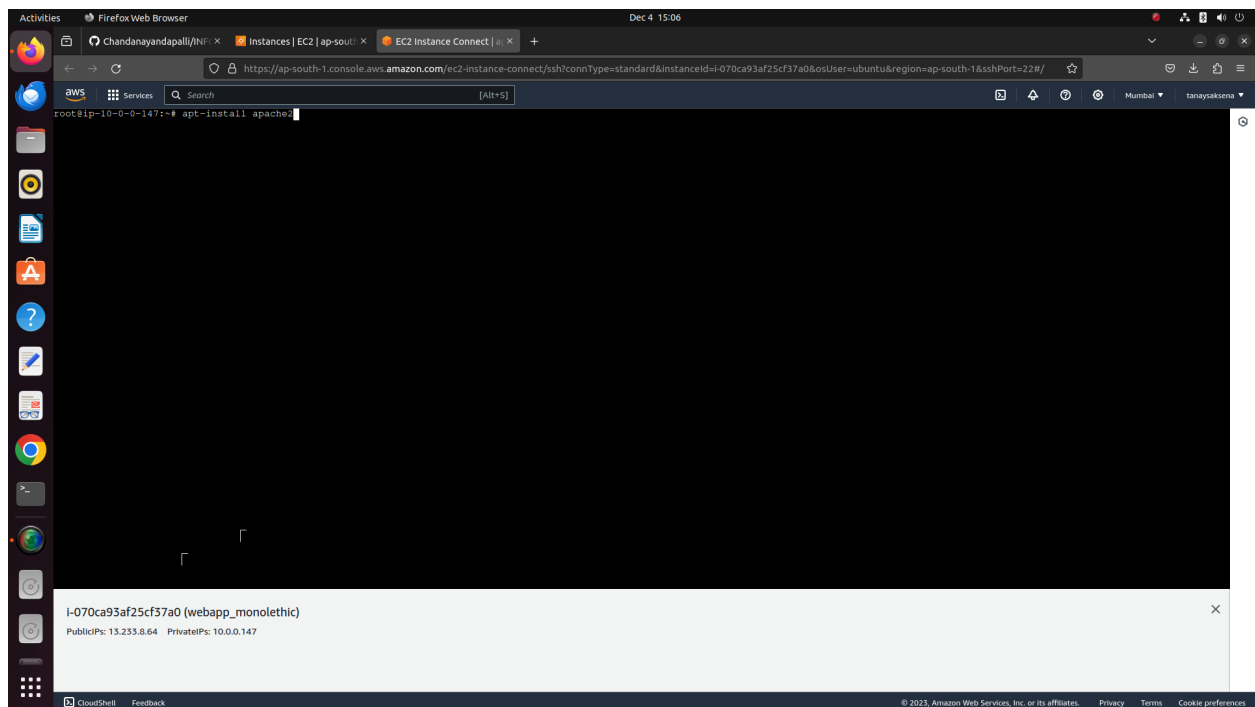
Instance is up & running



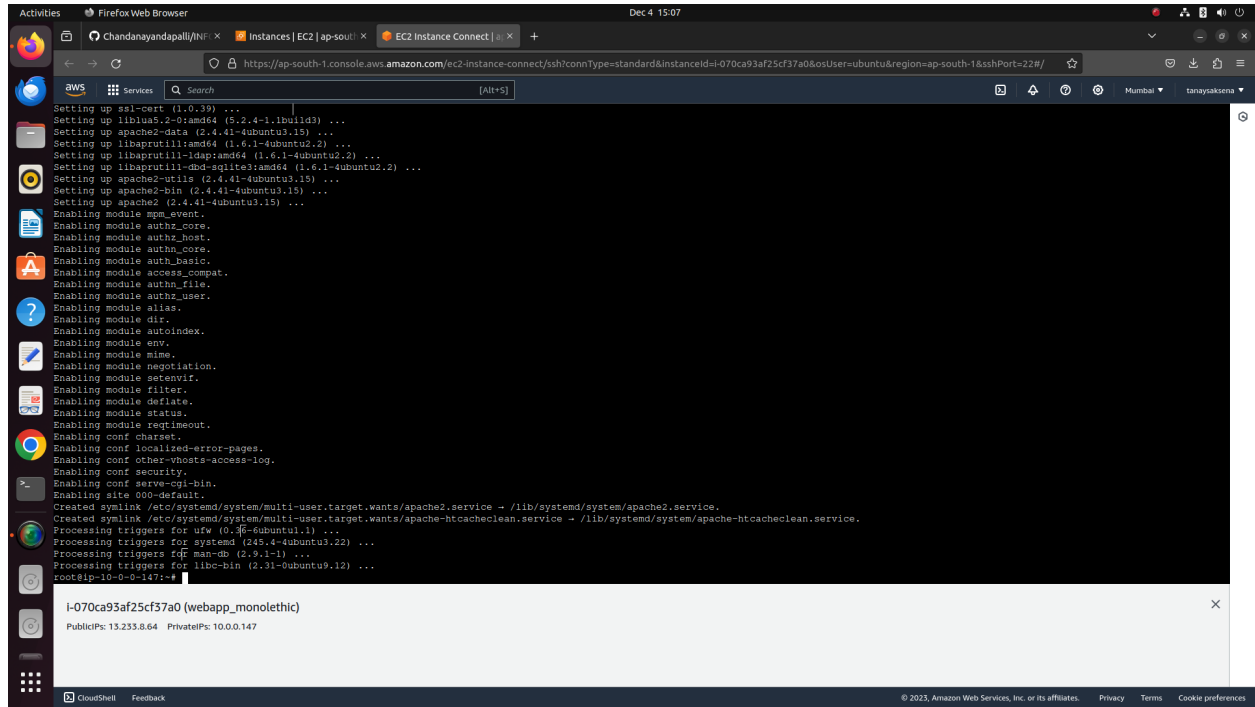
## Connected to EC2 instance via SSH & Installing Update



## Installing Apache2 server



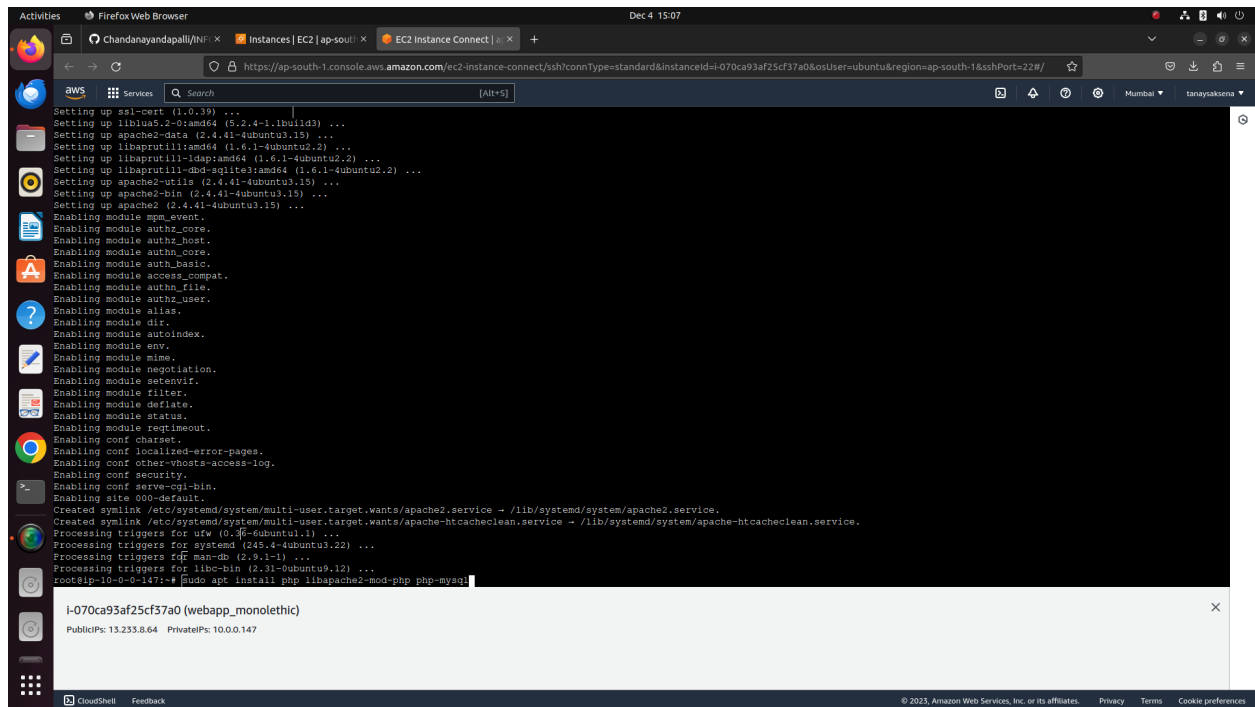
# Apache Installed



The screenshot shows the AWS Management Console interface for an EC2 instance named 'i-070ca93af25cf37a0'. The 'Instances' tab is selected, and the 'Connect' button is clicked, opening a terminal window. The terminal displays the output of the 'yum install httpd' command, showing the progress of installing Apache 2.4.41 on a CentOS 7.5.1804 instance. The output includes details about the packages being installed, the modules being enabled, and the configuration files being created. The terminal output is as follows:

```
Setting up ssl-cert (1.0.39) ...
Setting up liblua5.2-0:amd64 (5.2.4-1.1build3) ...
Setting up apache2-data (2.4.41-4ubuntu3.15) ...
Setting up libaprutil1:amd64 (1.6.1-4ubuntu2.2) ...
Setting up libaprutil1-dbd-sqlite3:amd64 (1.6.1-4ubuntu2.2) ...
Setting up apache2-utils (2.4.41-4ubuntu3.15) ...
Setting up apache2-bin (2.4.41-4ubuntu3.15) ...
Setting up apache2 (2.4.41-4ubuntu3.15) ...
Enabling module mpm_event.
Enabling module authz_core.
Enabling module authz_host.
Enabling module authn_core.
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service - /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service - /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36-6ubuntu1.1) ...
Processing triggers for systemd (245.4-4ubuntu3.22) ...
Processing triggers for mdadm (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
root@ip-10-0-0-147:~#
```

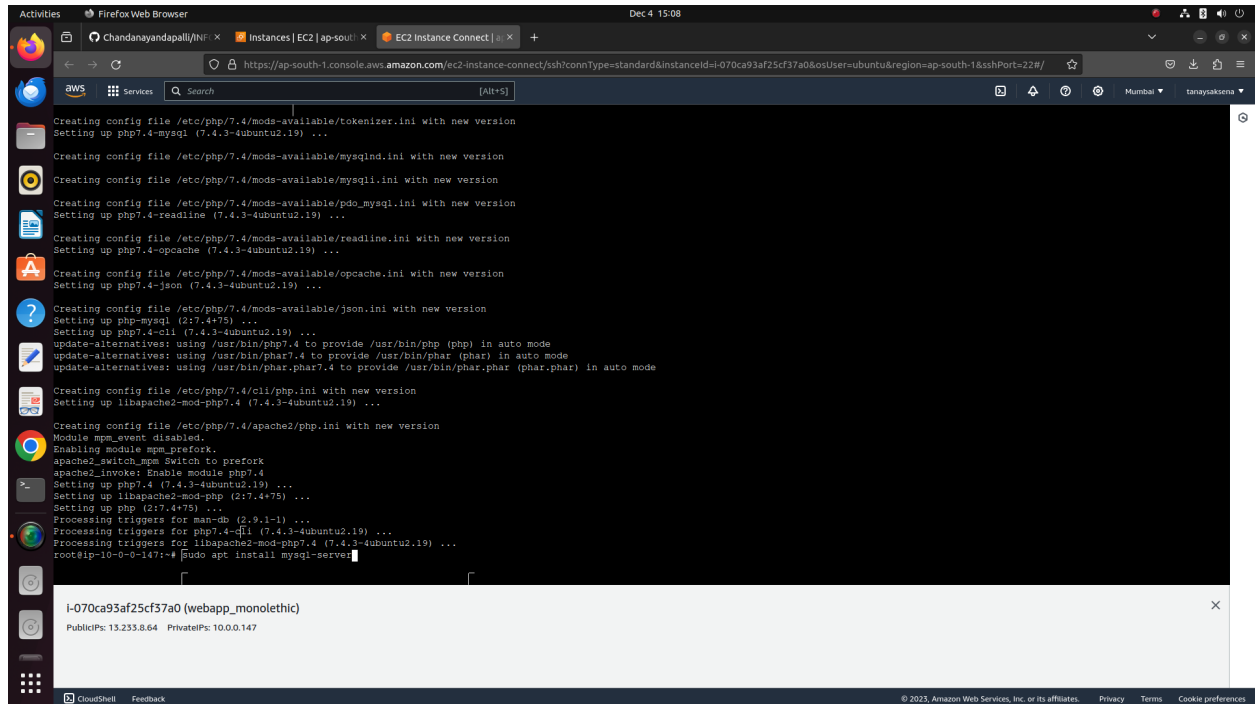
# Installing PHP Dependencies and my sql connector



The screenshot shows the AWS Management Console interface for the same EC2 instance. The terminal window now displays the output of the 'yum install php php-mysql' command, showing the progress of installing PHP 7.4.19 and its MySQL connector. The output includes details about the packages being installed, the dependencies being resolved, and the configuration files being created. The terminal output is as follows:

```
Setting up ssl-cert (1.0.39) ...
Setting up liblua5.2-0:amd64 (5.2.4-1.1build3) ...
Setting up apache2-data (2.4.41-4ubuntu3.15) ...
Setting up libaprutil1:amd64 (1.6.1-4ubuntu2.2) ...
Setting up libaprutil1-dbd-sqlite3:amd64 (1.6.1-4ubuntu2.2) ...
Setting up apache2-utils (2.4.41-4ubuntu3.15) ...
Setting up apache2-bin (2.4.41-4ubuntu3.15) ...
Setting up apache2 (2.4.41-4ubuntu3.15) ...
Enabling module mpm_event.
Enabling module authz_core.
Enabling module authz_host.
Enabling module authn_core.
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service - /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service - /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36-6ubuntu1.1) ...
Processing triggers for systemd (245.4-4ubuntu3.22) ...
Processing triggers for mdadm (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
root@ip-10-0-0-147:~#
```

# Installing MY SQL SERVER



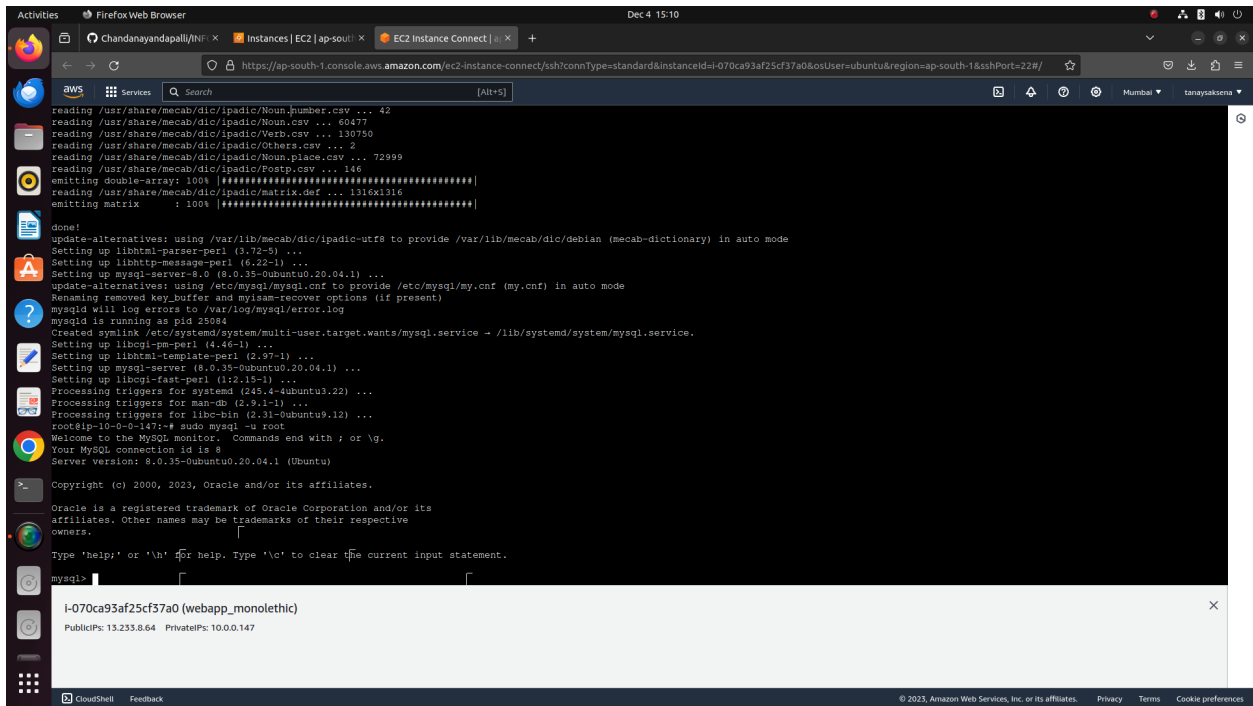
The screenshot shows a terminal window connected to an EC2 instance via AWS Instance Connect. The terminal output displays the steps for installing MySQL on an Ubuntu 20.04 instance. The steps include creating configuration files for various PHP extensions (tokenizer, mysqli, pdo\_mysql, readline, opcache, json), setting up the MySQL server, and enabling the module. The terminal output is as follows:

```
Creating config file /etc/php/7.4/mods-available/tokenizer.ini with new version
Setting up php7.4-mysql (7.4.3-4ubuntu2.19) ...
Creating config file /etc/php/7.4/mods-available/mysqli.ini with new version
Creating config file /etc/php/7.4/mods-available/pdo_mysql.ini with new version
Setting up php7.4-readline (7.4.3-4ubuntu2.19) ...
Creating config file /etc/php/7.4/mods-available/readline.ini with new version
Setting up php7.4-opcache (7.4.3-4ubuntu2.19) ...
Creating config file /etc/php/7.4/mods-available/opcache.ini with new version
Setting up php7.4-json (7.4.3-4ubuntu2.19) ...
Creating config file /etc/php/7.4/mods-available/json.ini with new version
Setting up php-mysql (2:7.4+75) ...
Setting up php7.4-cli (7.4.3-4ubuntu2.19) ...
update-alternatives: using /usr/bin/php7.4 to provide /usr/bin/php (php) in auto mode
update-alternatives: using /usr/bin/phar7.4 to provide /usr/bin/phar (phar) in auto mode
update-alternatives: using /usr/bin/phar.phar7.4 to provide /usr/bin/phar.phar (phar.phar) in auto mode
Creating config file /etc/php/7.4/cli/php.ini with new version
Setting up libapache2-mod-php7.4 (7.4.3-4ubuntu2.19) ...
Creating config file /etc/php/7.4/apache2/php.ini with new version
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
apache2_invoke: Enable module php7.4
Setting up php7.4 (7.4.3-4ubuntu2.19) ...
Setting up libapache2-mod-php (2:7.4+75) ...
Setting up php (2:7.4+75) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for php7.4-cli (7.4.3-4ubuntu2.19) ...
Processing triggers for libapache2-mod-php7.4 (7.4.3-4ubuntu2.19) ...
root@ip-10-0-0-147:~# sudo apt install mysql-server
```

Below the terminal output, a metadata box for the instance is visible:

```
I-070ca93af25cf37a0 (webapp_monoethic)
PublicIPs: 13.233.8.64 PrivateIPs: 10.0.0.147
```

## Sql Server login



```
reading /usr/share/mecab/dic/ipadic/Noun.jnumber.csv ... 42
reading /usr/share/mecab/dic/ipadic/Noun.csv ... 60477
reading /usr/share/mecab/dic/ipadic/Verb.csv ... 130750
reading /usr/share/mecab/dic/ipadic/Others.csv ... 2
reading /usr/share/mecab/dic/ipadic/Noun.place.csv ... 72999
reading /usr/share/mecab/dic/ipadic/Postp.csv ... 146
emitting double-array: 100% |#####|
reading /usr/share/mecab/dic/ipadic/matrix.def ... 1316x1316
emitting matrix : 100% |#####|

done!
update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode
Setting up libhtml-parser-perl (3.72-5) ...
Setting up libhttp-message-perl (6.22-1) ...
Setting up mysql-server-8.0 (8.0.35-0ubuntu0.20.04.1) ...
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Renaming removed key_buffer and myisam-recover options (if present)
mysql will log errors to /var/log/mysql/error.log
mysql is running as pid 25084
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.
Setting up liboci-pm-perl (4.46-1) ...
Setting up libhtml-template-perl (2.97-1) ...
Setting up mysql-server (8.0.35-0ubuntu0.20.04.1) ...
Setting up liboci-fast-perl (1:2.15-1) ...
Processing triggers for systemd (245.4-4ubuntu3.22) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
root@ip-10-0-0-147:~# sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

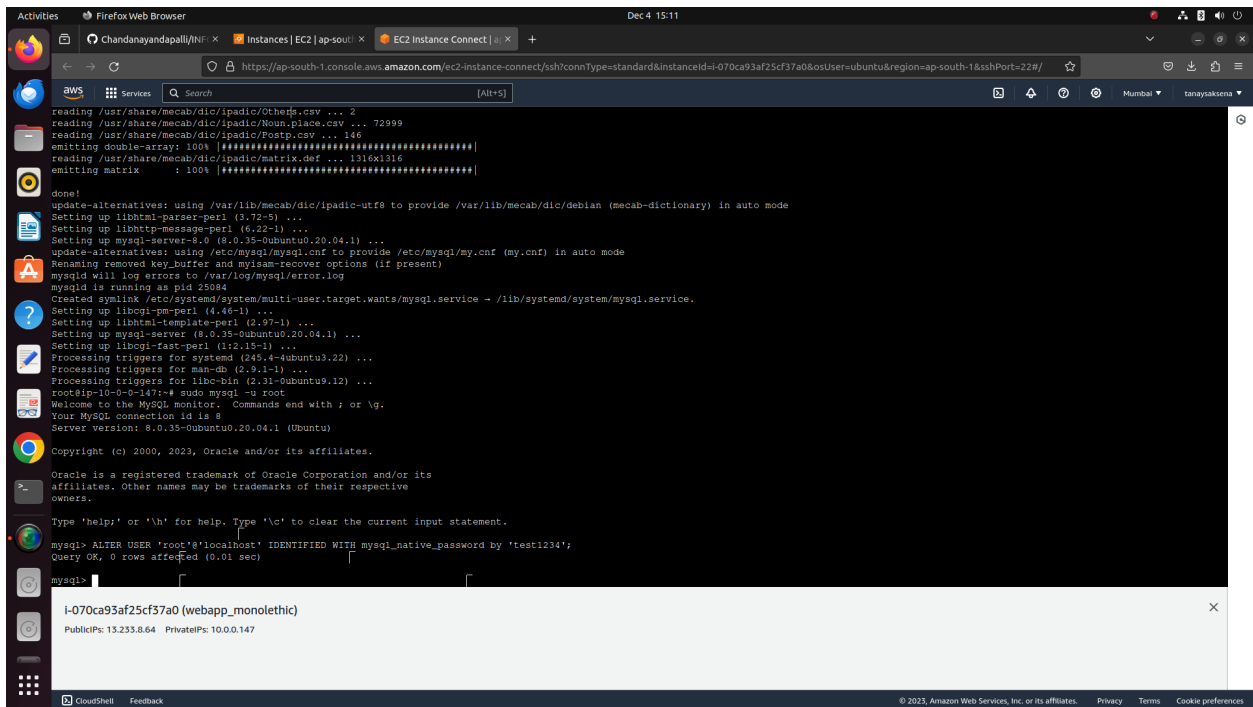
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>

I-070ca93af25cf37a0 (webapp_monoethic)
PublicIP: 13.233.8.64 PrivateIP: 10.0.0.147
```

# Root user Password changed



```
reading /usr/share/mecab/dic/ipadic/Other.csv ... 2
reading /usr/share/mecab/dic/ipadic/Noun.place.csv ... 72999
reading /usr/share/mecab/dic/ipadic/Postp.csv ... 146
emitting double-array: 100% |#####|
reading /usr/share/mecab/dic/ipadic/matrix.def ... 1316x1316
emitting matrix : 100% |#####|

done!
update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode
Setting up libhtml-parser-perl (3.72-5) ...
Setting up libhttp-message-perl (6.22-1) ...
Setting up mysql-server-8.0 (8.0.35-0ubuntu0.20.04.1) ...
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Renaming removed key_buffer and myisam-recover options (if present)
mysql will log errors to /var/log/mysql/error.log
mysql is running as pid 25084
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.
Setting up liboci-pm-perl (4.46-1) ...
Setting up libhtml-template-perl (2.97-1) ...
Setting up mysql-server (8.0.35-0ubuntu0.20.04.1) ...
Setting up liboci-fast-perl (1:2.15-1) ...
Processing triggers for systemd (245.4-4ubuntu3.22) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
root@ip-10-0-0-147:~# sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, 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> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'test1234';
Query OK, 0 rows affected (0.01 sec)

mysql>

I-070ca93af25cf37a0 (webapp_monoethic)
PublicIP: 13.233.8.64 PrivateIP: 10.0.0.147
```

# New user created in my sql

The screenshot shows a terminal window within an AWS EC2 instance. The terminal output displays the installation of MySQL on Ubuntu 20.04.1. It includes commands for updating alternatives, setting up the MySQL server, and creating a new user named 'tanay'. The user is created with the password 'test1234' and is granted all privileges. The terminal also shows the MySQL version and copyright information.

```
done!  
update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode  
Setting up libhtml-parser-perl (3.72-5) ...  
Setting up libhttp-message-perl (6.22-1) ...  
Setting up mysql-server-8.0 (8.0.35-0ubuntu0.20.04.1) ...  
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode  
Renaming removed key_buffer and myisam-recover options (if present)  
mysql will log errors to /var/log/mysql/error.log  
mysqld is running as pid 25084  
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.  
Setting up libio-nonblock-perl (4.46-1) ...  
Setting up libhtml-template-perl (2.97-1) ...  
Setting up mysql-server (8.0.35-0ubuntu0.20.04.1) ...  
Setting up libio-nonblock-perl (4.46-1) ...  
Processing triggers for systemd (245.4-4ubuntu3.22) ...  
Processing triggers for man-db (2.9.1-1) ...  
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...  
root@ip-10-0-0-147:~# sudo mysql -u root  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 8  
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)  
Copyright (c) 2000, 2023, 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> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'test1234';  
Query OK, 0 rows affected (0.01 sec)  
mysql> CREATE USER 'tanay'@'localhost' IDENTIFIED BY 'test1234';  
Query OK, 0 rows affected (0.01 sec)  
mysql>
```

## Created Database

The screenshot shows a terminal window within an AWS EC2 instance, continuing the MySQL setup. The terminal output displays the creation of a new database named 'wp'. The user 'tanay' is shown creating the database, and the terminal confirms the successful creation.

```
done!  
update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode  
Setting up libhtml-parser-perl (3.72-5) ...  
Setting up libhttp-message-perl (6.22-1) ...  
Setting up mysql-server-8.0 (8.0.35-0ubuntu0.20.04.1) ...  
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode  
Renaming removed key_buffer and myisam-recover options (if present)  
mysql will log errors to /var/log/mysql/error.log  
mysqld is running as pid 25084  
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.  
Setting up libio-nonblock-perl (4.46-1) ...  
Setting up libhtml-template-perl (2.97-1) ...  
Setting up mysql-server (8.0.35-0ubuntu0.20.04.1) ...  
Setting up libio-nonblock-perl (4.46-1) ...  
Processing triggers for systemd (245.4-4ubuntu3.22) ...  
Processing triggers for man-db (2.9.1-1) ...  
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...  
root@ip-10-0-0-147:~# sudo mysql -u root  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 8  
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)  
Copyright (c) 2000, 2023, 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> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'test1234';  
Query OK, 0 rows affected (0.01 sec)  
mysql> CREATE USER 'tanay'@'localhost' IDENTIFIED BY 'test1234';  
Query OK, 0 rows affected (0.01 sec)  
mysql> CREATE DATABASE wp;  
Query OK, 1 row affected (0.00 sec)  
mysql>
```

## Granted all Privileges to new user

The screenshot shows a terminal window connected to an AWS EC2 instance via EC2 Instance Connect. The terminal output displays the installation of MySQL on an Ubuntu 20.04.1 instance. The process includes setting up the libhtml-parser-perl, libhttp-message-perl, and libio-socket-ssl-perl packages, followed by the installation of MySQL server and client. The MySQL service is started, and the root user is created with a password. The terminal also shows the MySQL monitor output, including the MySQL version and the MySQL monitor prompt.

```
Setting up libhtml-parser-perl (3.72-5) ...
Setting up libhttp-message-perl (6.22-1) ...
Setting up mysql-server-8.0 (8.0.35-0ubuntu0.20.04.1) ...
update-alternatives: using /etc/mysql/mysql.conf to provide /etc/mysql/my.cnf in auto mode
Remaining removed key buffer and myisam-recover options (if present)
mysqld will log errors to /var/log/mysql/error.log
mysqld is running as pid 25084
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service - /lib/systemd/system/mysql.service.
Setting up libio-socket-ssl-perl (4.46-1) ...
Setting up mysql-server (8.0.35-0ubuntu0.20.04.1) ...
Setting up libio-socket-ssl-perl (4.46-1) ...
Processing triggers for systemd (245.4-ubuntu0.22) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
root@ip-10-0-0-147:~# sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, 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> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'test1234';
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE USER 'tanay'@'localhost' IDENTIFIED BY 'test1234';
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE DATABASE wp;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON wp.* TO 'tanay'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql>
```

# Download, unzip and copy wordpress to /var/www/html

The screenshot shows a terminal window connected to an AWS EC2 instance via EC2 Instance Connect. The terminal output displays the download and installation of WordPress on an Ubuntu 20.04.1 instance. The process includes downloading the latest WordPress tar.gz file from the WordPress website, extracting it, and copying the contents to the /var/www/html directory. The terminal also shows the output of the ls command, confirming the successful installation of WordPress.

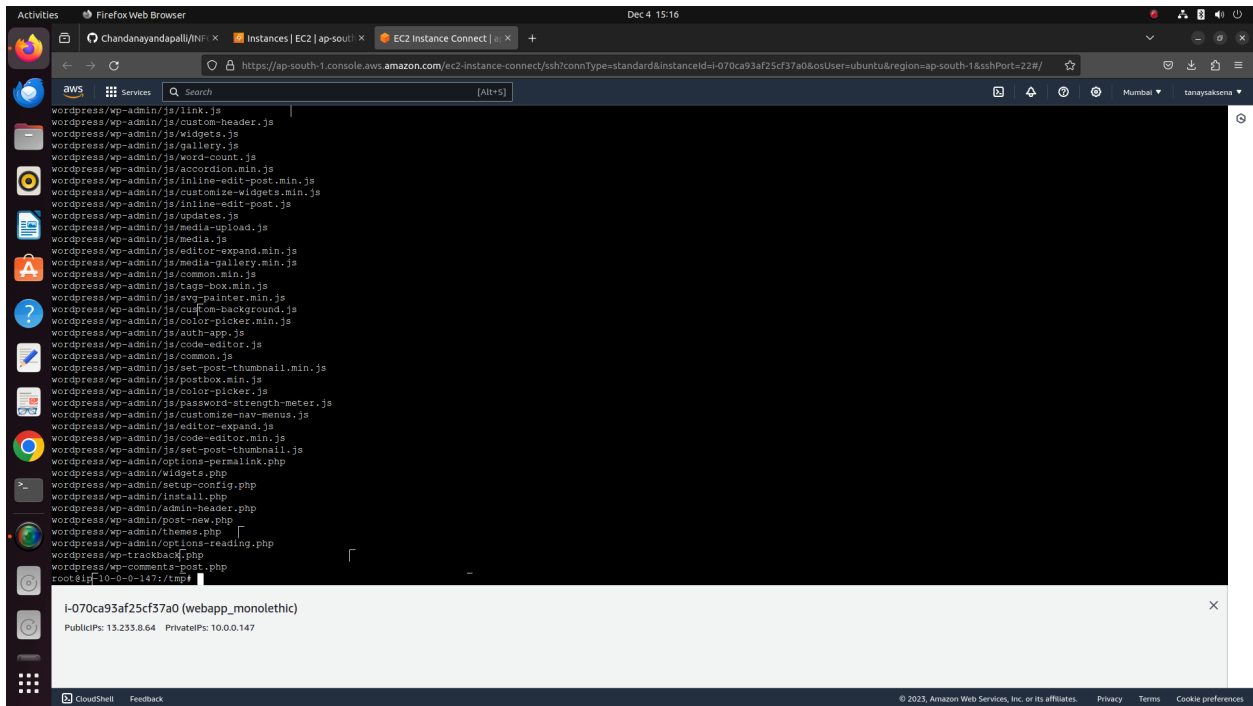
```
root@ip-10-0-0-147:~# wget https://wordpress.org/latest.tar.gz
--2023-12-04 09:44:51-- https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24479162 (23M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz 100%[=====] 23.34M 6.66MB/s in 5.0s

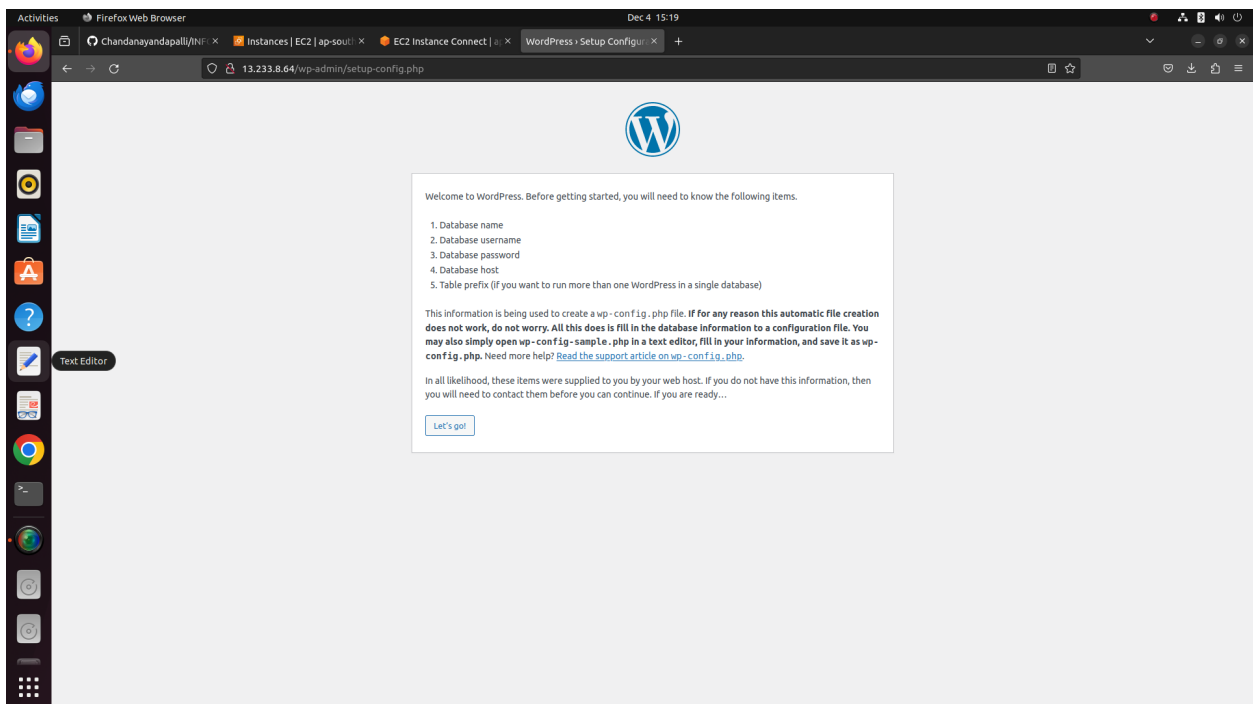
2023-12-04 09:44:57 (4.64 MB/s) - 'latest.tar.gz' saved [24479162/24479162]

root@ip-10-0-0-147:~# ls
latest.tar.gz
systemd-private-b1153704f038462c82d021daed6b8a7e-systemd-logind.service-rERBzg
systemd-private-b1153704f038462c82d021daed6b8a7e-systemd-resolved.service-8PqMf
systemd-private-b1153704f038462c82d021daed6b8a7e-systemd-timesyncd.service-9dMxj
tmp.2a4j5BdaJg

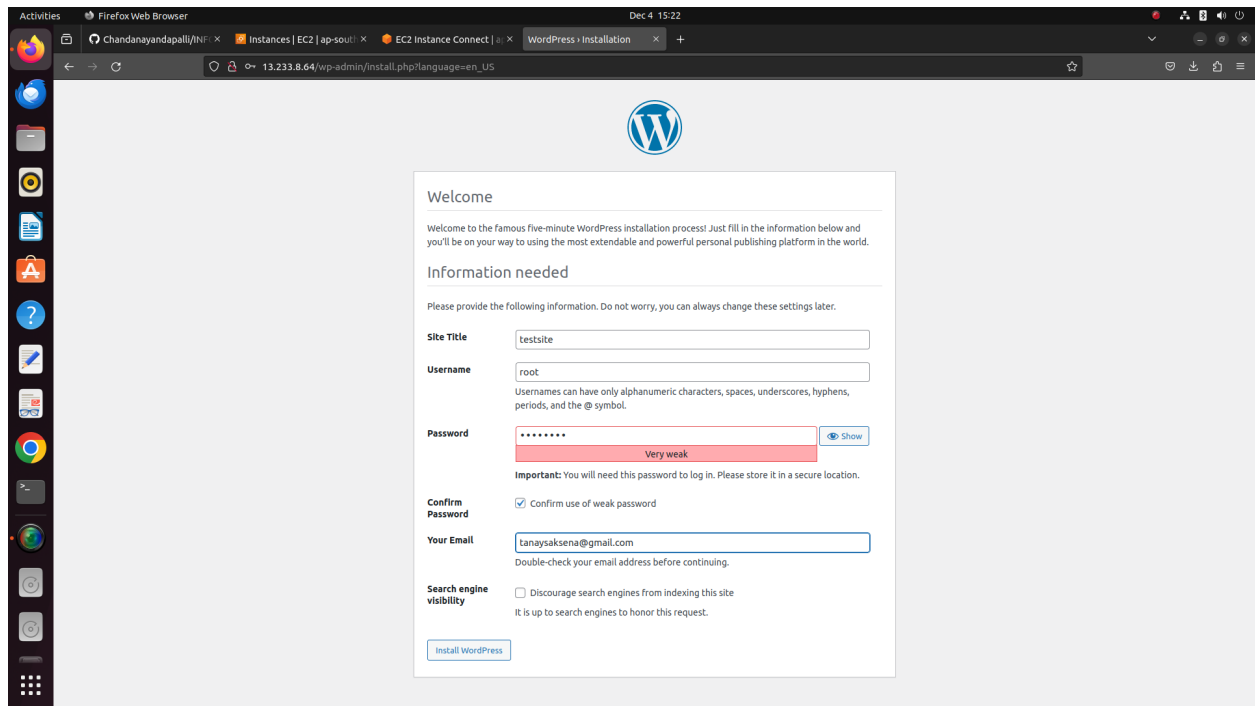
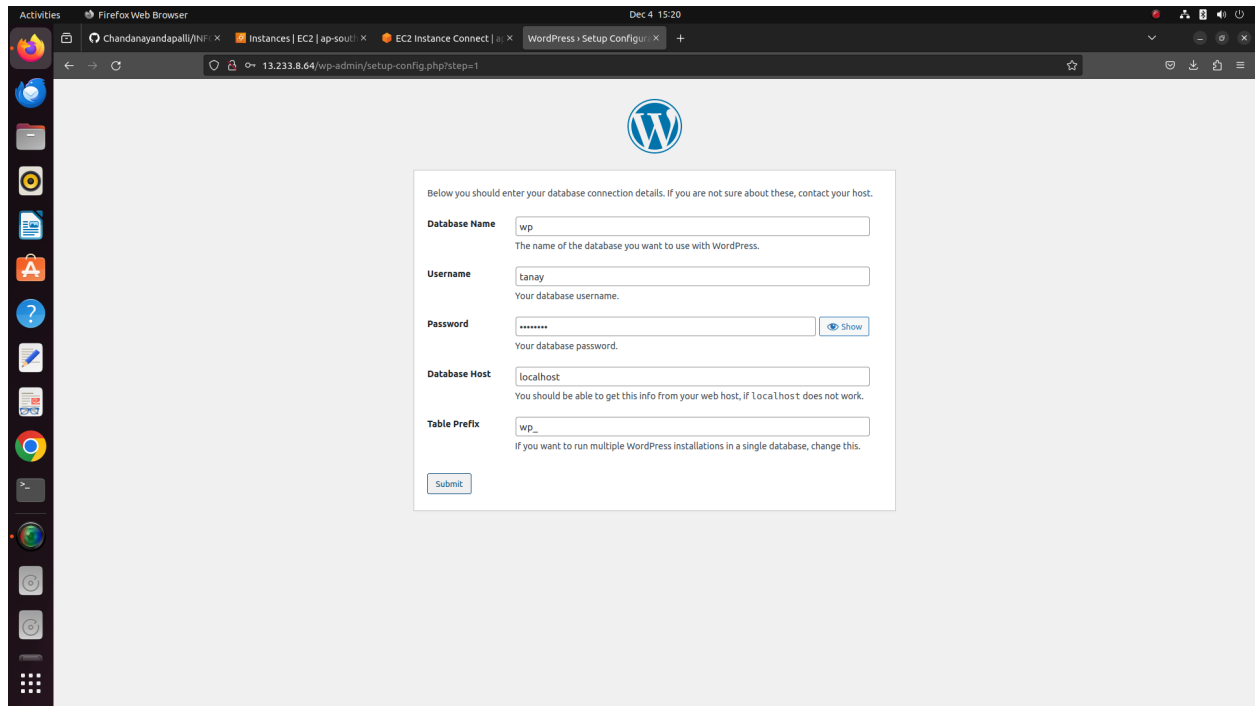
root@ip-10-0-0-147:~#
```



## Wordpress installation and Configure







Wordpress login and created new front page

