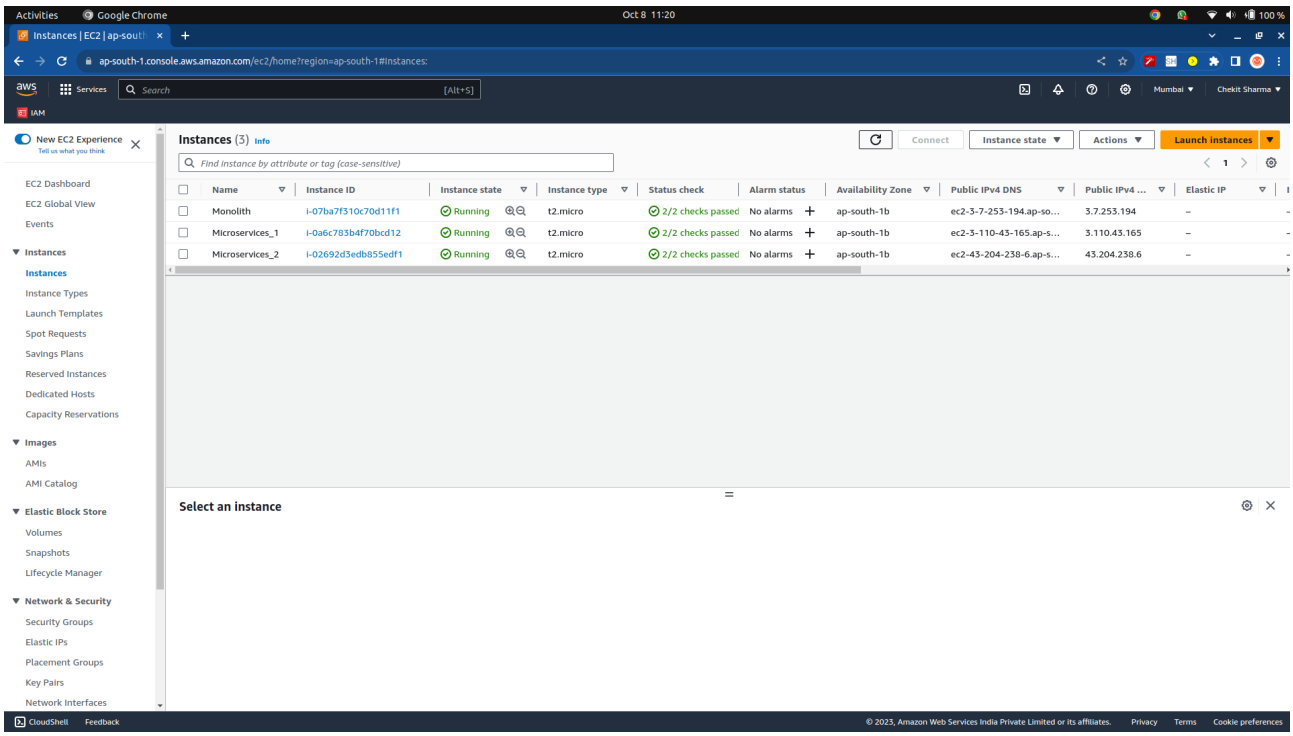
MICROSERVICES

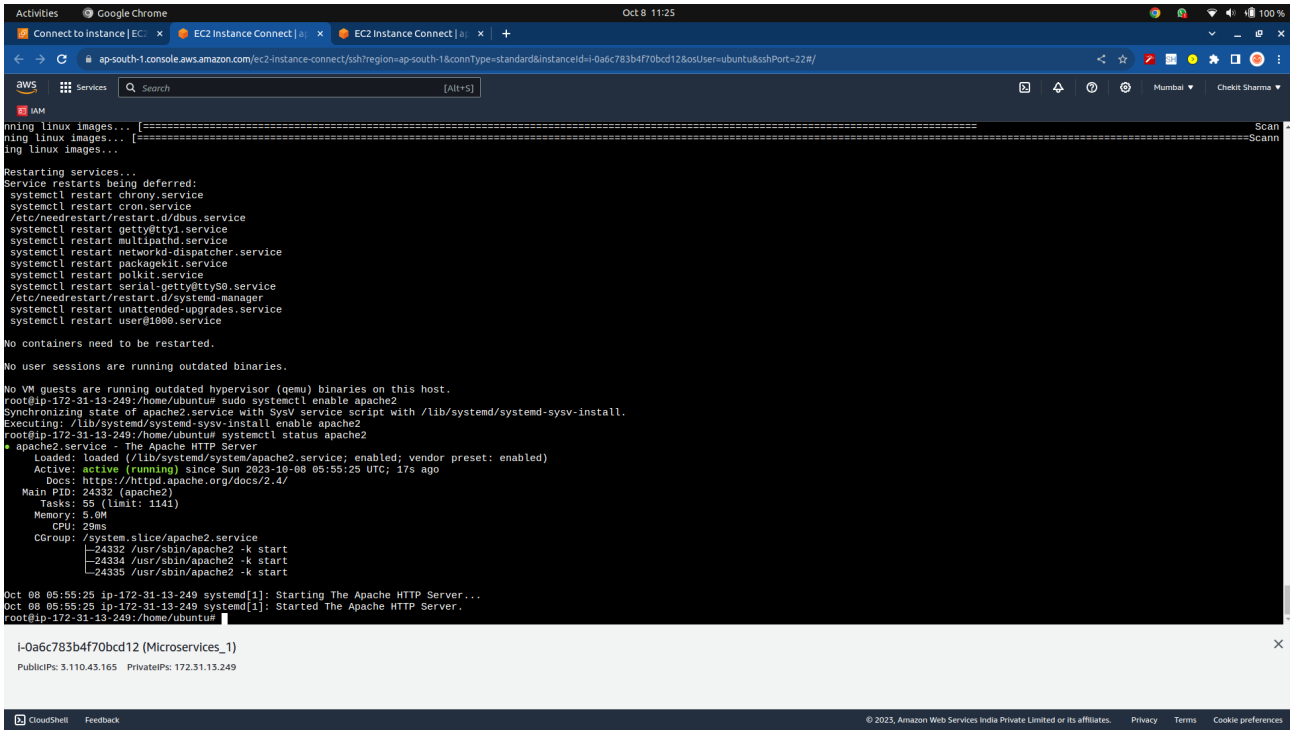
steps -

1. Create 2 EC2 Instances



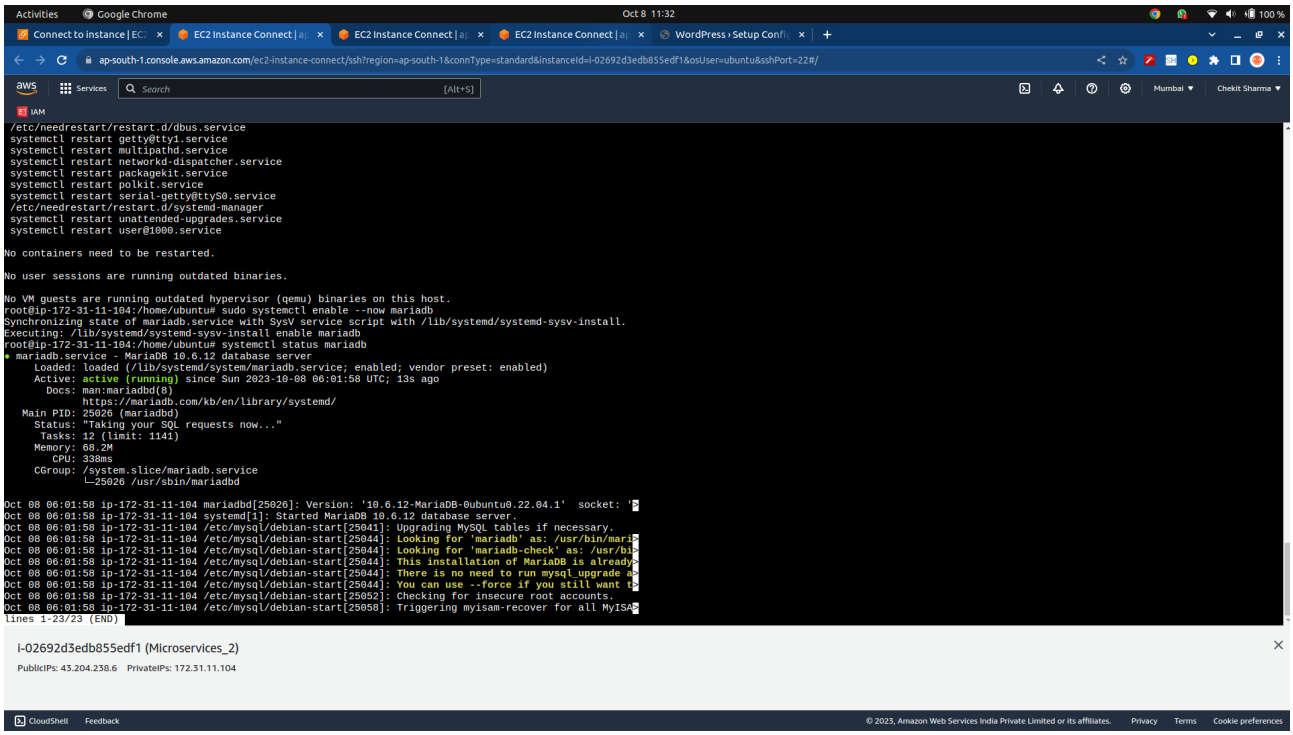
2. Perform Ubuntu 22.04 update on both

3. Install Apache & PHP in (Instance 1) for Wordpress


Activities
Google Chrome
Oct 8 11:25
Connect to instance | EC2
EC2 Instance Connect | ap
EC2 Instance Connect | ap
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0a6c783b4f70bcd12&osUser=ubuntu&sshPort=22#/
aws
Services
Search
[Alt+S]
Mumbai
Chekit Sharma
IAM
nning linux images... [
ning linux images... [
ing linux images...
Restarting services...
Service restarts being deferred:
systemctl restart chrony.service
systemctl restart cron.service
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart multipathd.service
systemctl restart networkd-dispatcher.service
systemctl restart packagekit.service
systemctl restart polkit.service
systemctl restart serial-getty@ttyS0.service
/etc/needrestart/restart.d/systemd-manager
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-13-249:/home/ubuntu# sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
root@ip-172-31-13-249:/home/ubuntu# systemctl status apache2
apache2.service - The Apache HTTP Server
Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
Active: active (running) since Sun 2023-10-08 05:55:25 UTC; 17s ago
Docs: https://httpd.apache.org/docs/2.4/
Main PID: 24332 (apache2)
Tasks: 55 (limit: 1141)
Memory: 5.0M
CPU: 29ms
CGroup: /system.slice/apache2.service
├─24332 /usr/sbin/apache2 -k start
├─24334 /usr/sbin/apache2 -k start
└─24335 /usr/sbin/apache2 -k start
Oct 08 05:55:25 ip-172-31-13-249 systemd[1]: Starting The Apache HTTP Server...
Oct 08 05:55:25 ip-172-31-13-249 systemd[1]: Started The Apache HTTP Server.
root@ip-172-31-13-249:/home/ubuntu#
i-0a6c783b4f70bcd12 (Microservices_1)
PublicIPs: 3.110.43.165 PrivateIPs: 172.31.13.249
CloudShell
Feedback
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences


Activities
Google Chrome
Oct 8 10:51
Instances | EC2 | ap-south
EC2 Instance Connect | ap
Apache2 Ubuntu Default
Not secure
3.7.253.194
Apache2 Default Page
Ubuntu
It works!
This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should replace this file (located at /var/www/html/index.html) before continuing to operate your HTTP server.
If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.
Configuration Overview
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is fully documented in /usr/share/doc/apache2/README.Debian.gz. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the manual if the apache2-doc package was installed on this server.
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:
/etc/apache2/
|-- apache2.conf
| `-- ports.conf
|-- mods-enabled
| |-- *.load
| `-- *.conf
|-- conf-enabled
| `-- *.conf
|-- sites-enabled
| `-- *.conf
apache2.conf is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
ports.conf is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
Configuration files in the mods-enabled/, conf-enabled/ and sites-enabled/ directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
They are activated by symlinking available configuration files from their respective *-available/ counterparts. These should be managed by using our helpers a2enmod, a2dismod, a2ensite, a2dissite, and a2enconf, a2disconf . See their respective man pages for detailed information.
The binary is called apache2 and is managed using systemd, so to start/stop the service use systemctl start apache2 and systemctl stop apache2, and use systemctl status apache2 and journalctl -u apache2 to check status. system and apache2ctl can also be used for service management if desired. Calling /usr/bin/apache2 directly will not work with the default configuration.
Document Roots

4. Install MariaDB or MySQL in (Instance 2)

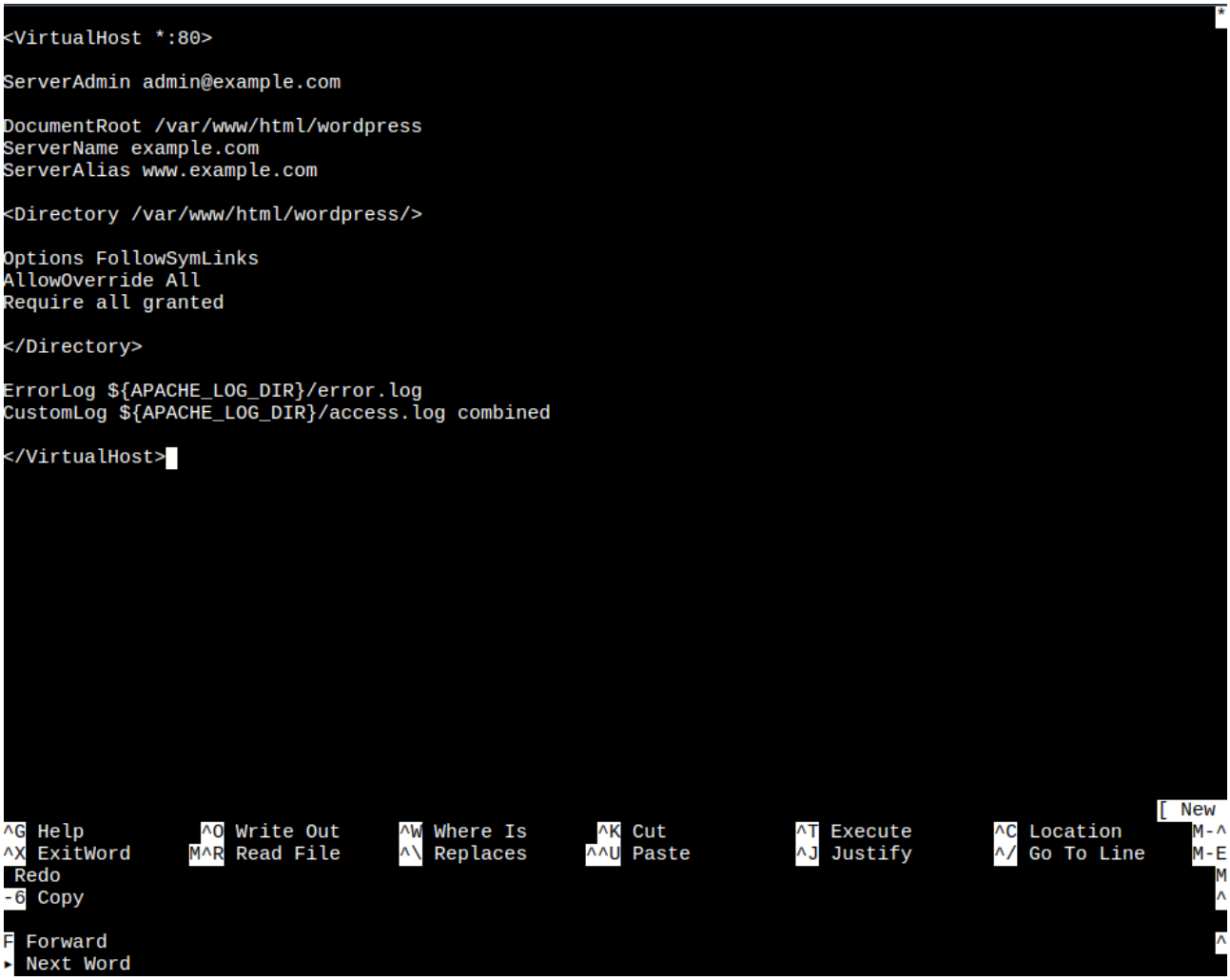


5. Create Database for WordPress (Instance 2)

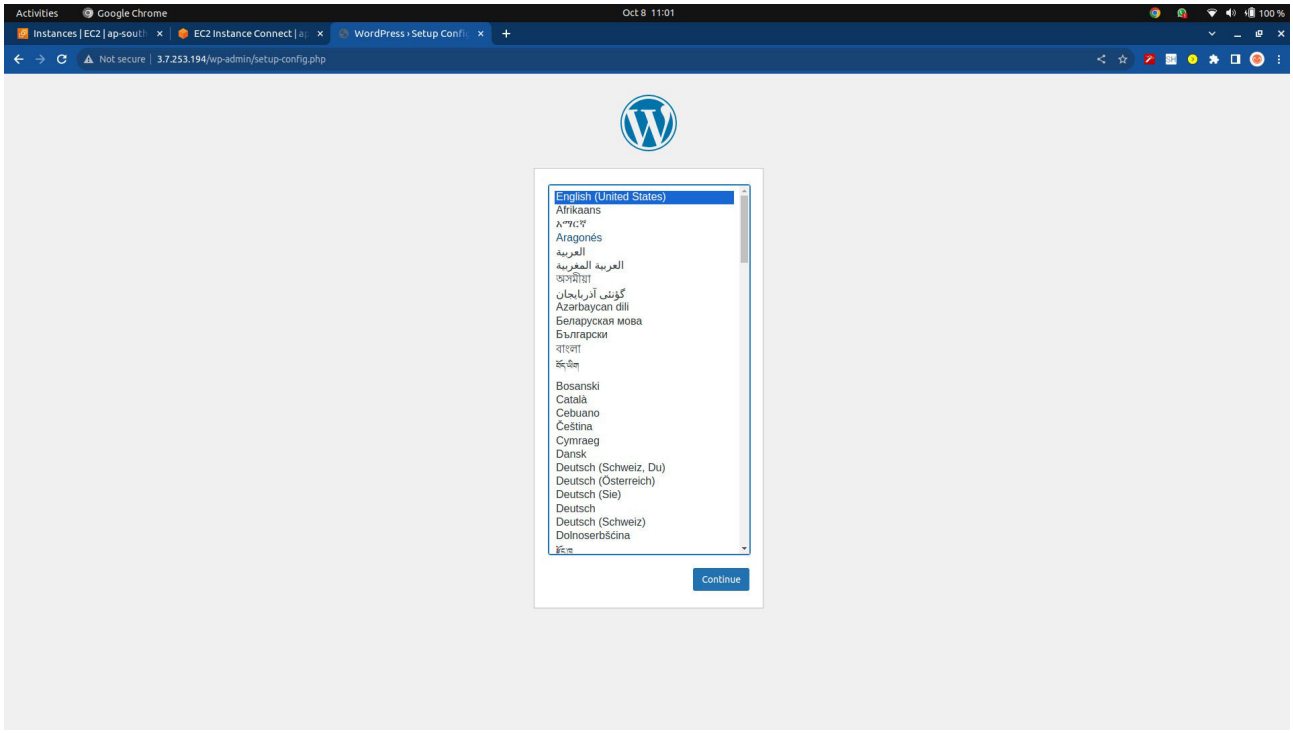
```
Remove test database and access to it? [Y/n] y
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
 ... Success!

Cleaning up...

All done!  If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
root@ip-172-31-3-210:/home/ubuntu# sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.6.12-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> CREATE USER 'new_user'@'localhost' IDENTIFIED BY 'your_password';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> CREATE DATABASE new_db;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON new_db.* TO 'new_user'@'localhost';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> Exit;
Bye
root@ip-172-31-3-210:/home/ubuntu#
```

6. Install WordPress on Ubuntu 22.04 (Instance 1)

7. Configure Apache on Ubuntu 22.04 (Instance 1)

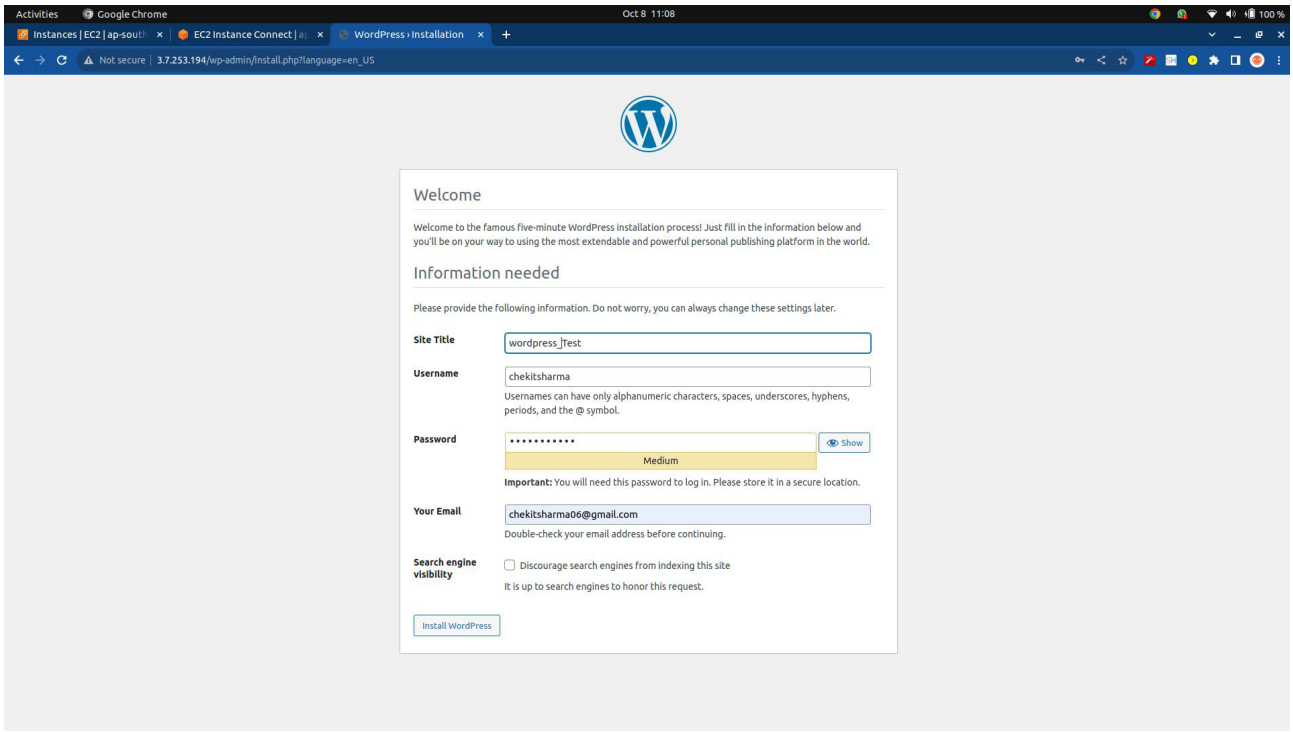


8. WordPress CMS web interface setup (Instance 1 Ip)

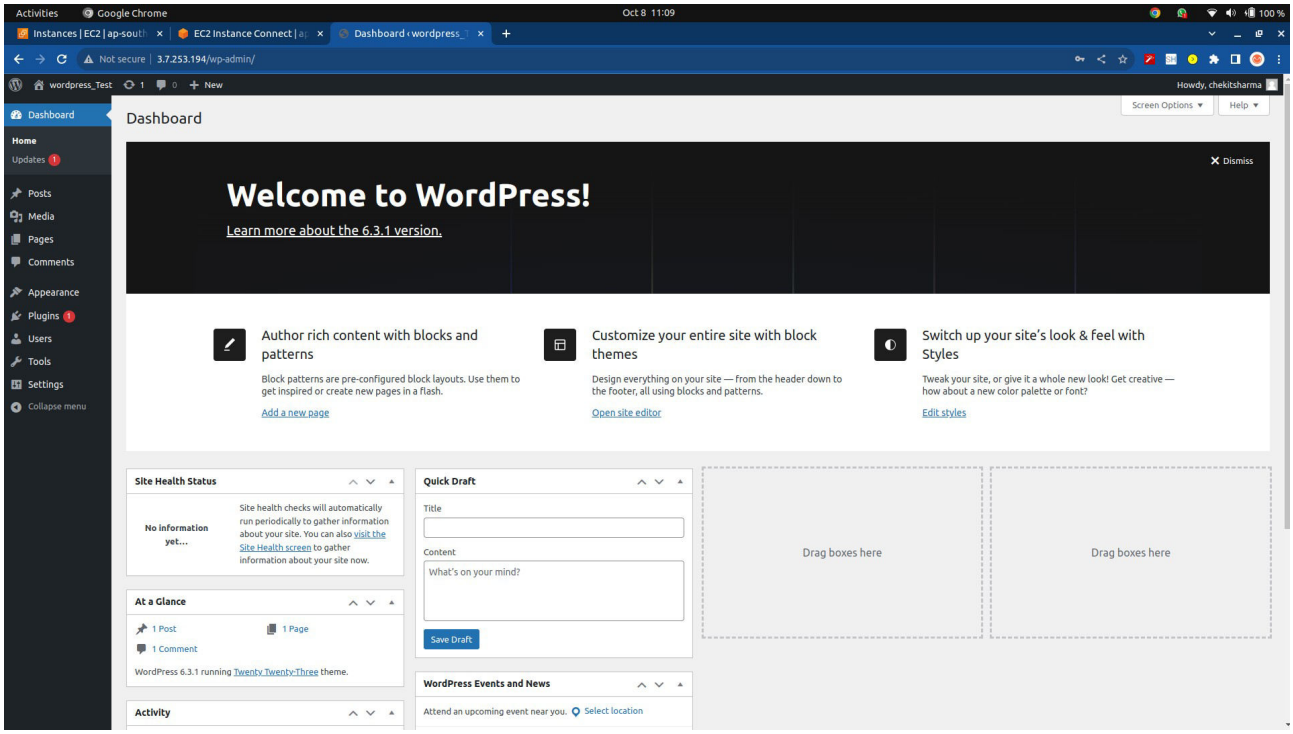


9. Add Database information

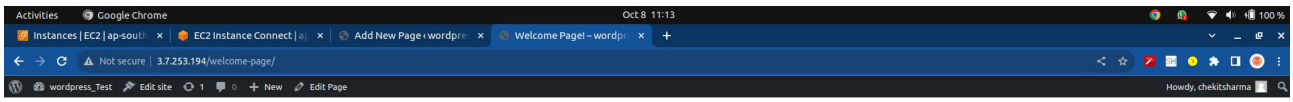
10. Create Admin user and password



11. Login to the backend (Instance 1 Ip)



12. Create Welcome Page (URL Provided by Wordpress)



Welcome Page!

Hello! Chekit Sharma here, This is a Test Page to see if i have successfuly deployed wordpress and created a welcome page.