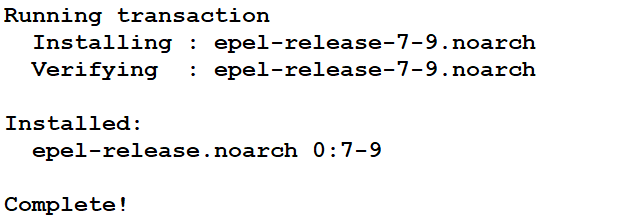
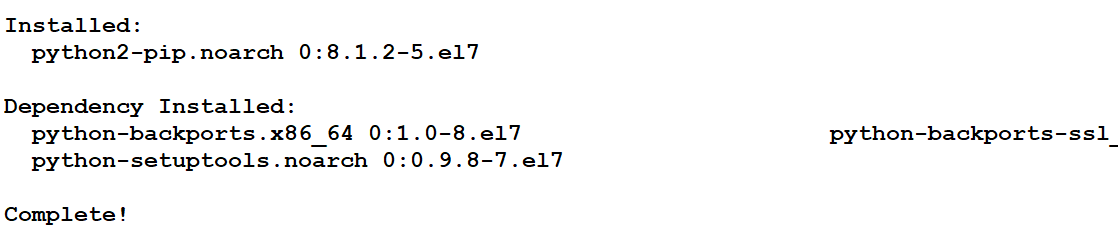
**Mysql-Wordpress compose using Docker Contaniers**

**Install the required Packages for Docker Compose**

# yum install epel-release -y

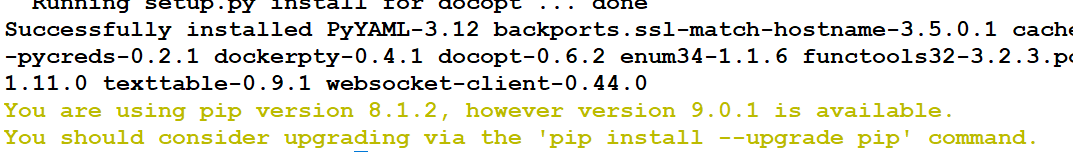


# yum install -y python-pip



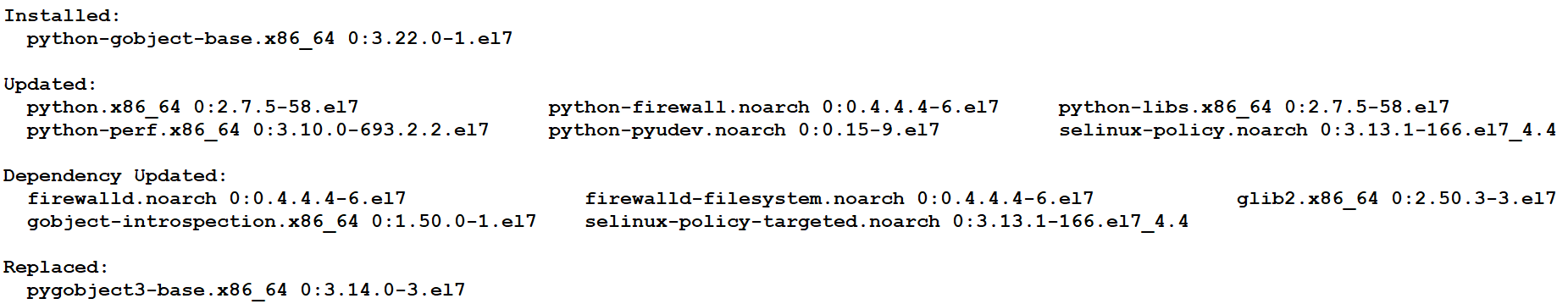
**Install Docker-Compose**

# pip install docker-compose



**Upgrade the Python Packages to get Docker-compose to run successfully**

# yum upgrade python\*



**Step 1 : Create a dir to save the file**

# mkdir my-wordpress

**Step 2 : Change dir**

# cd my-wordpress

**Step 3 : Create a docker-compose.yaml**

# vi docker-compose.yaml

version: '2'

services:

db:

image: mysql:5.7

volumes:

- db\_data:/var/lib/mysql

restart: always

environment:

MYSQL\_ROOT\_PASSWORD: wordpress

MYSQL\_DATABASE: wordpress

MYSQL\_USER: wordpress

MYSQL\_PASSWORD: wordpress

wordpress:

depends\_on:

- db

image: wordpress:latest

ports:

- "8000:80"

restart: always

environment:

WORDPRESS\_DB\_HOST: db:3306

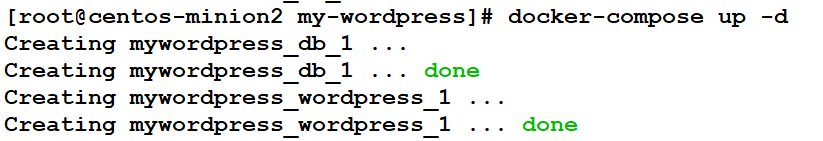
WORDPRESS\_DB\_PASSWORD: wordpress

volumes:

db\_data:

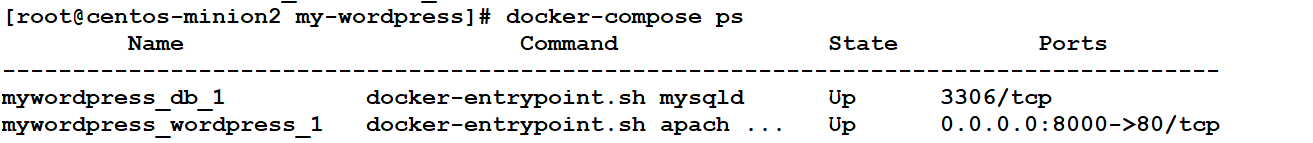
**Step 4 : Run the docker-compose command and you pwd should be /root/my-wordpress**

# docker-compose up -d



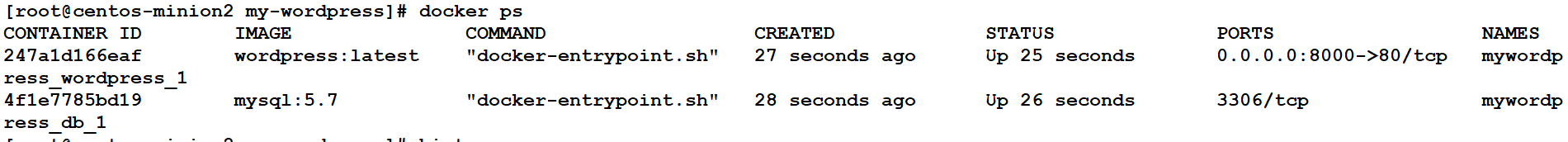
**Step 4 : Run the docker-compose ps command to check the containers**

# docker-compose ps

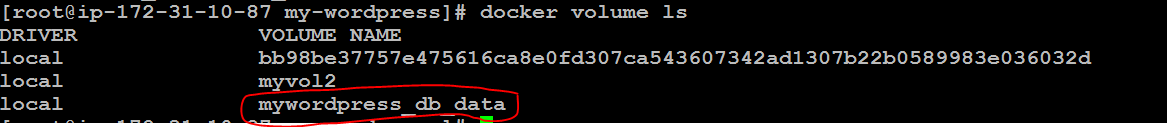


**Step 5 : Run the docker ps command and check for the containers**

# docker ps

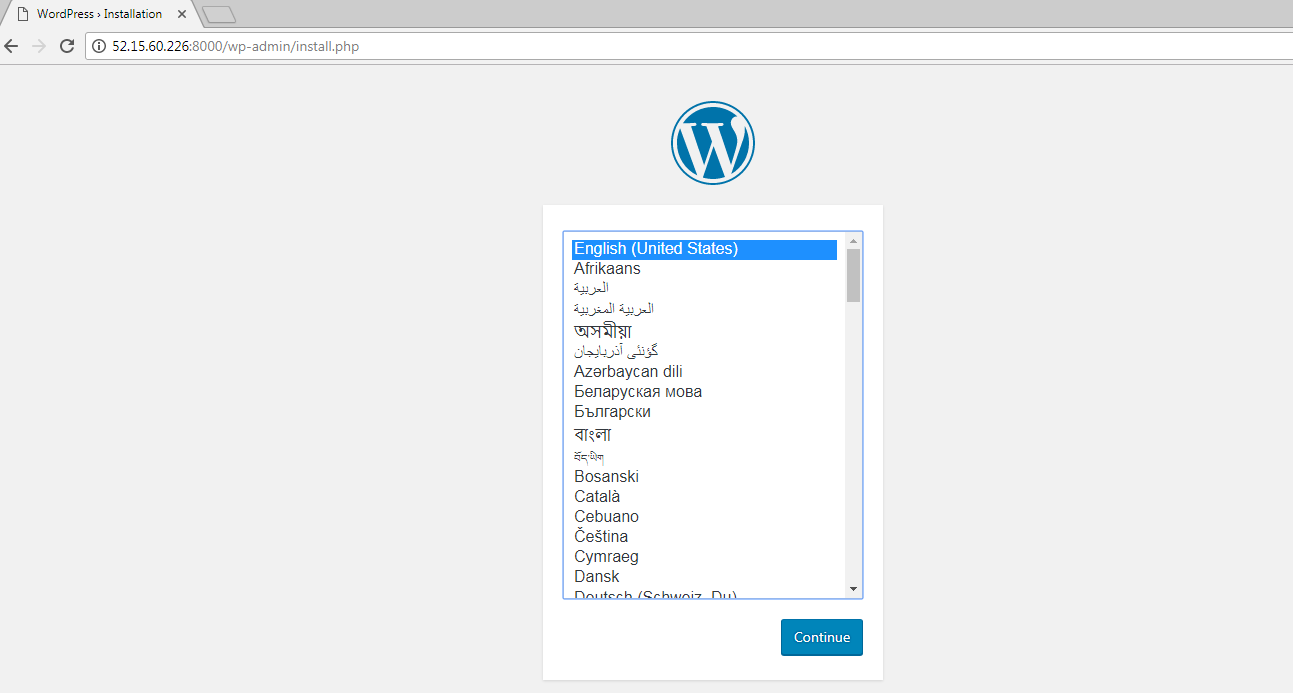


docker volume ls

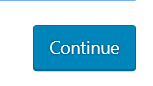


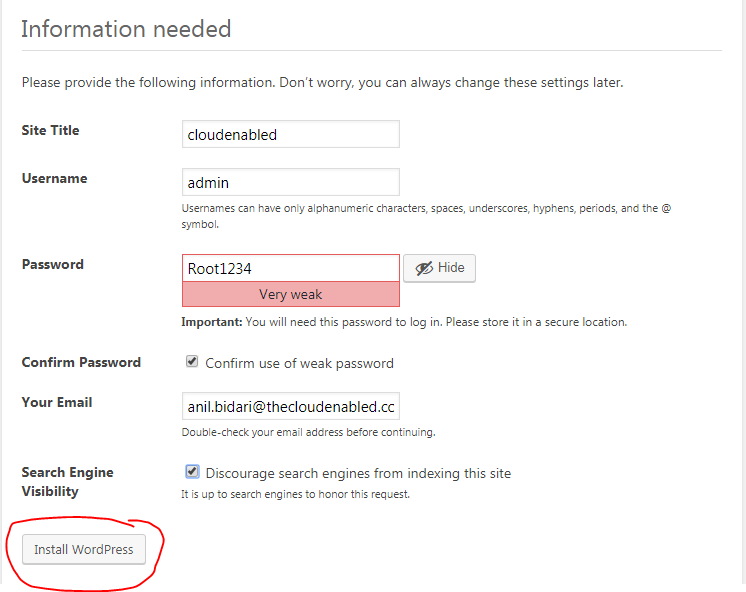
**Step 6 : Go to the web browser and check for the wordpress**

# http://publicip:8000

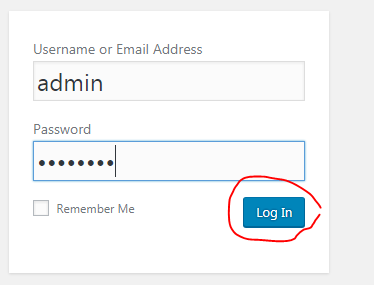


**Step 7 : Click on continue**

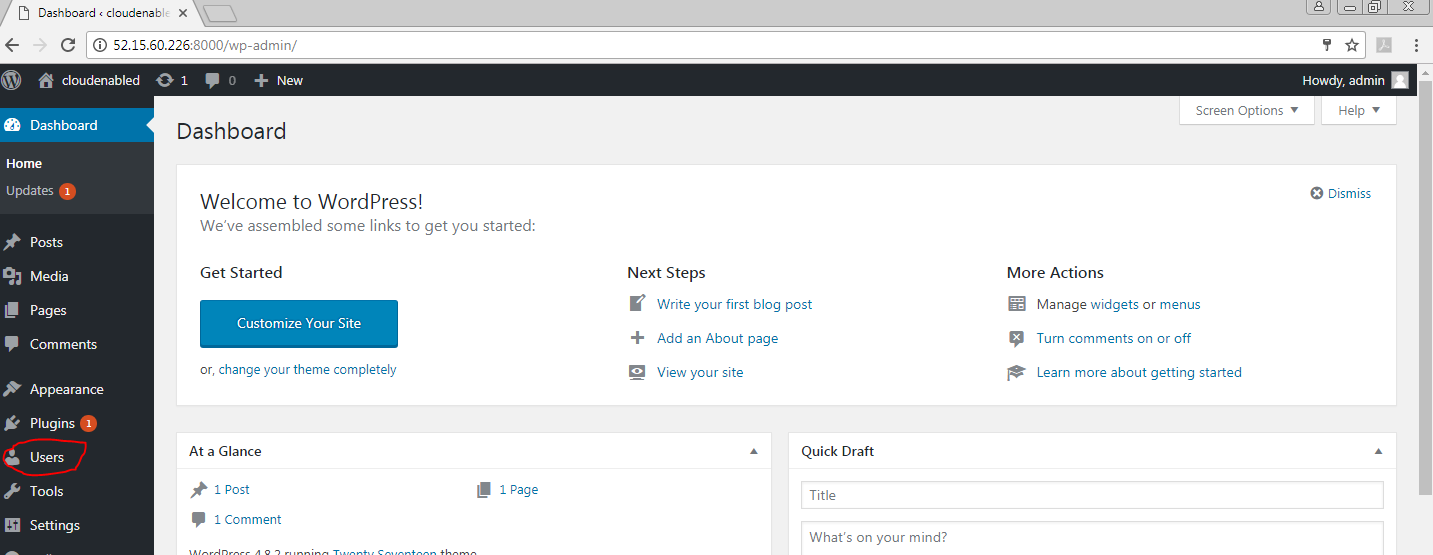


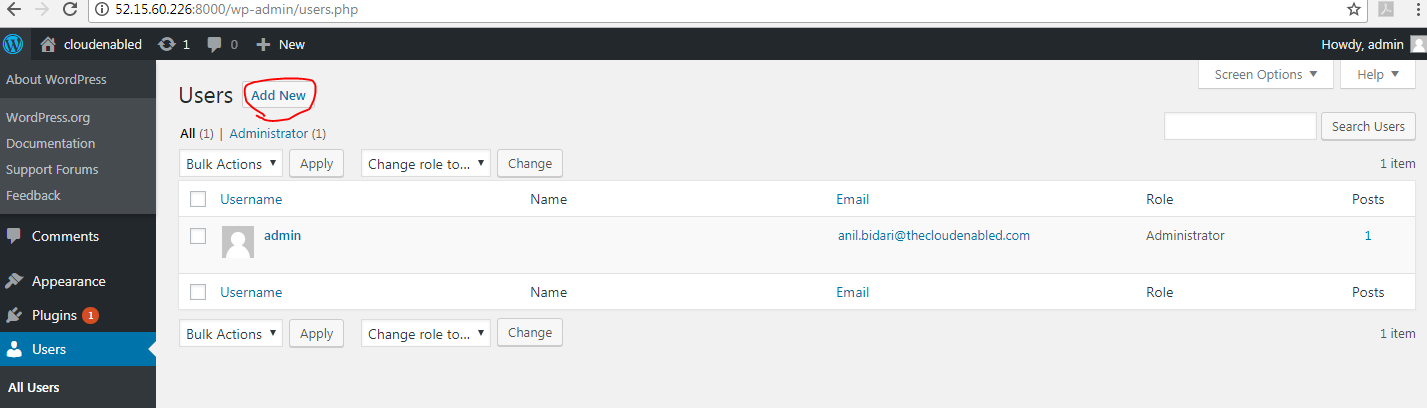


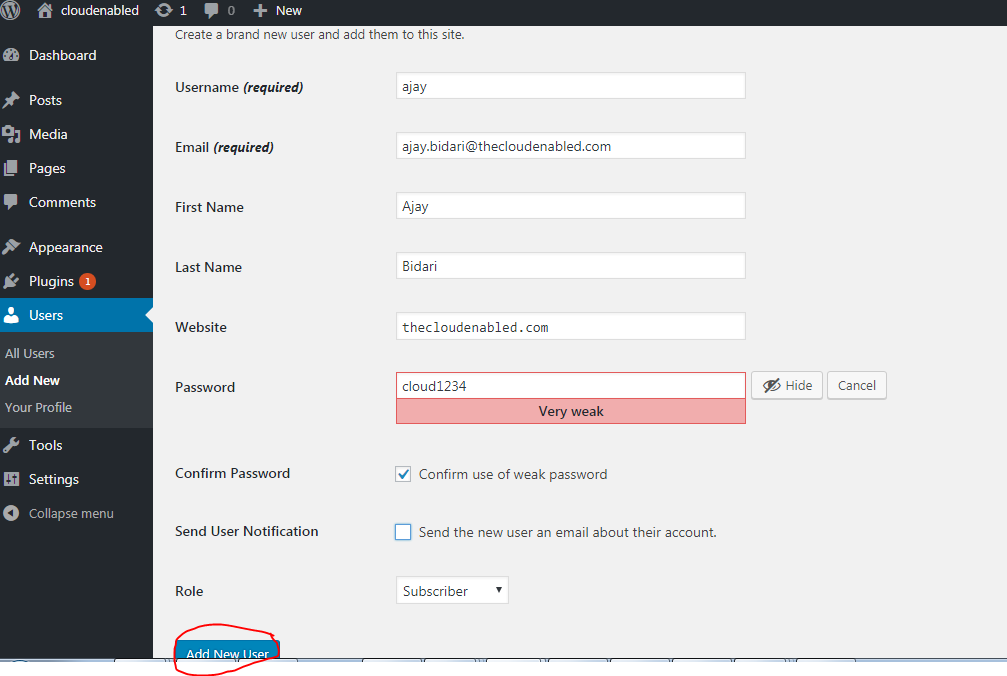
**Login to portal**



**Create a user**







**Logout of the account and login back as anjay**

