1. Create a Docker file.

chauhan2000 / midsem

<> Code   ⓘ Issues   Pull requests   ⓘ Actions   Projects   Wiki

master ▾   midsem / Dockerfile

chauhan2000 midsem

1 contributor

5 lines (4 sloc) | 111 Bytes

```
1   FROM phpmyadmin:latest
2
3   COPY ./src /var/www/html/
4   COPY apache2.conf /etc/apache2/apache2.conf
5   #ENTRYPOINT bash
```

2. Create a DockerCompose.yml file

chauhan2000 Update docker-compose.yml
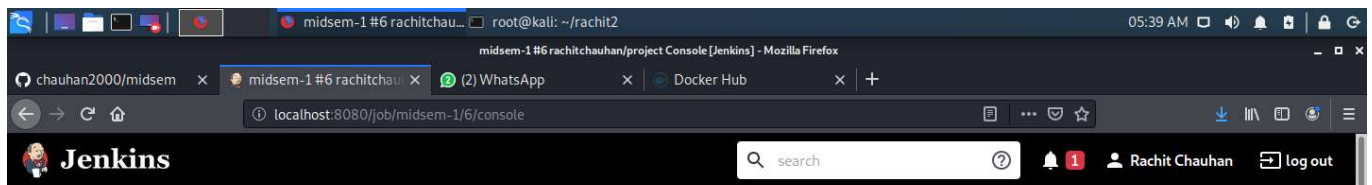
1 contributor

10 lines (7 sloc) | 125 Bytes

```
1   version: '3.3'
2
3   services:
4
5     webapp:
6       image: rachitchauhan/project
7       ports:
8         - "8008:80"
9       restart: always
10
```

3. Open Jenkins and download the cloudbees plugin from manage plugin section.

4. Create a Jenkins job (freestyle)
5. Provide the url of your github repository
6. And in the build step select the build and deploy option and give credentials of your docker hub and repository url.
7. Check that the repository is pushed to Docker hub or not

chauhan2000/midsem ✕   midsem-1 #6 rachitchau ✕   (2) WhatsApp ✕   Docker Hub ✕ | +

localhost:8080/job/midsem-1/6/console

Jenkins

search   🔔 1   👤 Rachit Chauhan   log out

Dashboard ‣ midsem-1 ‣ #6 rachitchauhan/project

- 🟢 Back to Project
- 🔍 Status
- 📝 Changes
- 🖥 Console Output
  - 📄 View as plain text
- 📝 Edit Build Information
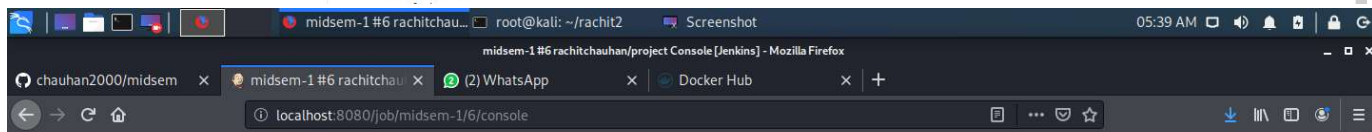- ◈ Git Build Data
- 💾 No Tags
- ⬅ Previous Build

🔵 Console Output

```
Started by user Rachit Chauhan
Running as SYSTEM
Building in workspace /root/.jenkins/workspace/midsem-1
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/chauhan2000/midsem.git # timeout=10
Fetching upstream changes from https://github.com/chauhan2000/midsem.git
 > git --version # timeout=10
 > git --version # 'git version 2.24.0'
 > git fetch --tags --force --progress -- https://github.com/chauhan2000/midsem.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision c8e3c1139f08d35c06fb2eb2ff179524cc856913 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f c8e3c1139f08d35c06fb2eb2ff179524cc856913 # timeout=10
Commit message: "midsem"
 > git rev-list --no-walk 2ab17622f55fede0124f04c6ceff0d202d1f30aa # timeout=10
[midsem-1] $ docker build -t rachitchauhan/project --pull=true /root/.jenkins/workspace/midsem-1
Sending build context to Docker daemon  111.1kB

Step 1/3 : FROM phpmyadmin:latest
latest: Pulling from library/phpmyadmin
Digest: sha256:5ae73d0a421e4bb7284640e0a7792709068f54866d7dce4d8396f58fa9e76deb
Status: Image is up to date for phpmyadmin:latest
 ---> 5fa0b0c8888e
Step 2/3 : COPY ./src /var/www/html/
 ---> Using cache
 ---> 4829967065be
Step 3/3 : COPY apache2.conf /etc/apache2/apache2.conf
 ---> Using cache
```


```
82801ee01d09: Waiting
f447060f9935: Waiting
7d55cc606fe1: Waiting
435f9e9c9e58: Waiting
fc0b35927b5a: Waiting
8c7a39edfbf0: Waiting
8888f89886ba: Waiting
f182865e86e9: Waiting
b2bb5e569df9: Waiting
d0fe97fa8b8c: Waiting
c9ec7fe9a88a: Waiting
9542cac5d52a: Layer already exists
71d45b84e621: Layer already exists
6da1c072b05a: Layer already exists
ae8cd72e3c62: Layer already exists
dd0576a7d248: Layer already exists
13d54f76b02e: Layer already exists
2f2e73839893: Layer already exists
82801ee01d09: Layer already exists
f447060f9935: Layer already exists
7d55cc606fe1: Layer already exists
c9ec7fe9a88a: Layer already exists
435f9e9c9e58: Layer already exists
8e350c691162: Pushed
fc0b35927b5a: Layer already exists
8c7a39edfbf0: Layer already exists
8888f89886ba: Layer already exists
f182865e86e9: Layer already exists
0f7cfcb05ee3: Pushed
d0fe97fa8b8c: Layer already exists
b2bb5e569df9: Layer already exists
latest: digest: sha256:e63547a17ed389d33ef064cac8d2ad1ac835dc83dcf603818183a124af8770a7 size: 4496
Finished: SUCCESS
```
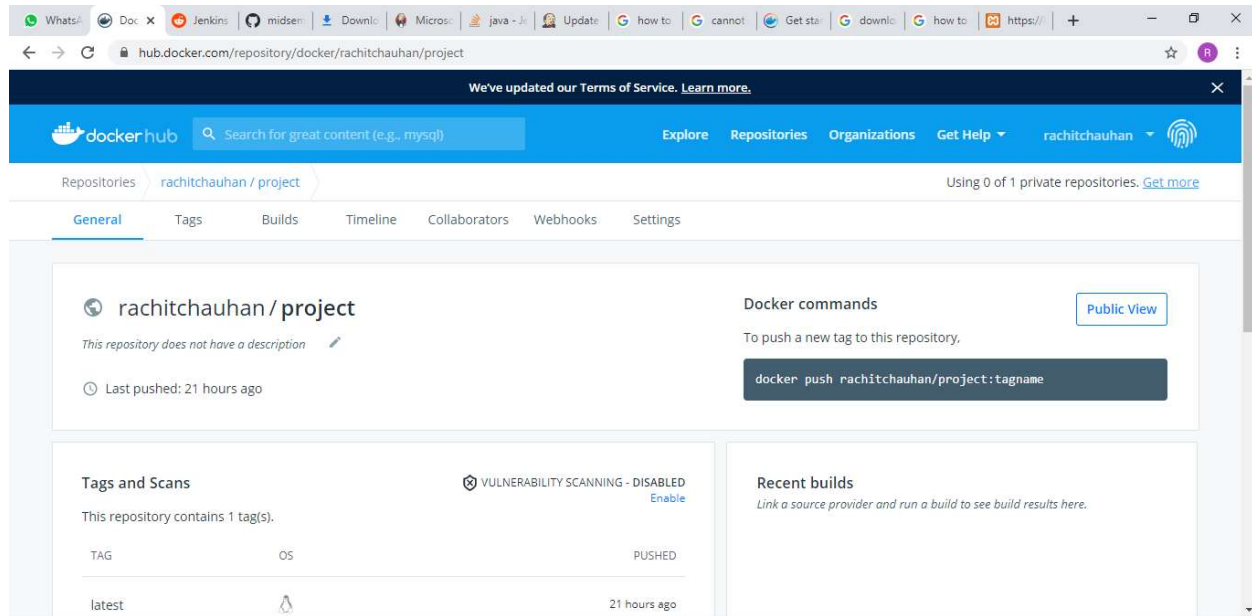
8. Go to aws and then go to IAM and then create a user

9. Enter the access key and secret key from there to the main.tf file
10. Then terraform init
11. Then terraform plan and apply
12. Open the AWS account there go to the ecs and check your cluster is present or not
13. Open the cluster and then check its state that is it running or not then copy the public ip and paste it in your browser.
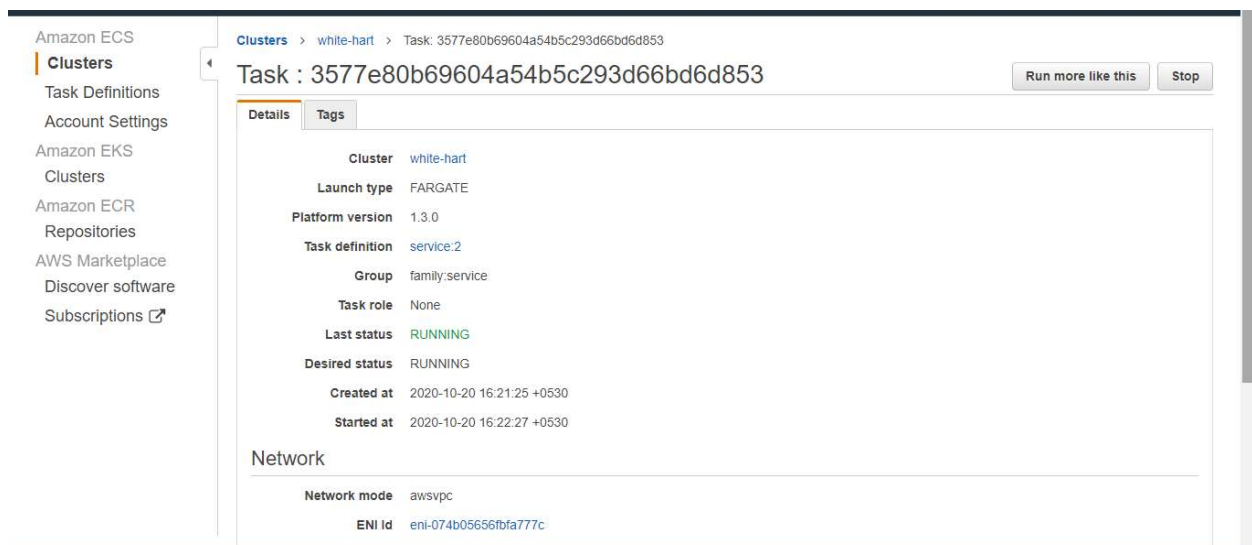
Task role   None

Last status   RUNNING

Desired status   RUNNING

Created at   2020-10-20 16:21:25 +0530

Started at   2020-10-20 16:22:27 +0530

Network

Network mode   awsvpc

ENI Id   eni-074b05656fbfa777c

Subnet Id   subnet-01dba05337d484c47

Private IP   132.0.0.191

Public IP   54.152.234.90

Mac address   06:a8:5e:8e:ca:b3

Containers

Last updated on October 21, 2020 4:02:45 PM (more than 1 hour ago)

Name	Container Runtime I...	St...	Image	Image Digest	C...	H...	E...	R...
▶ first	54334fa1c97e771775...	R...	docker.io/rachitchauhan/project		10	51...	true	f4...

🔒 Not secure | 54.152.234.90

phpMyAdmin
Welcome to phpMyAdmin

Language

English

Log in ⓘ

Username: [   ]

Password: [   ]

Go