**Jenkin assignment**

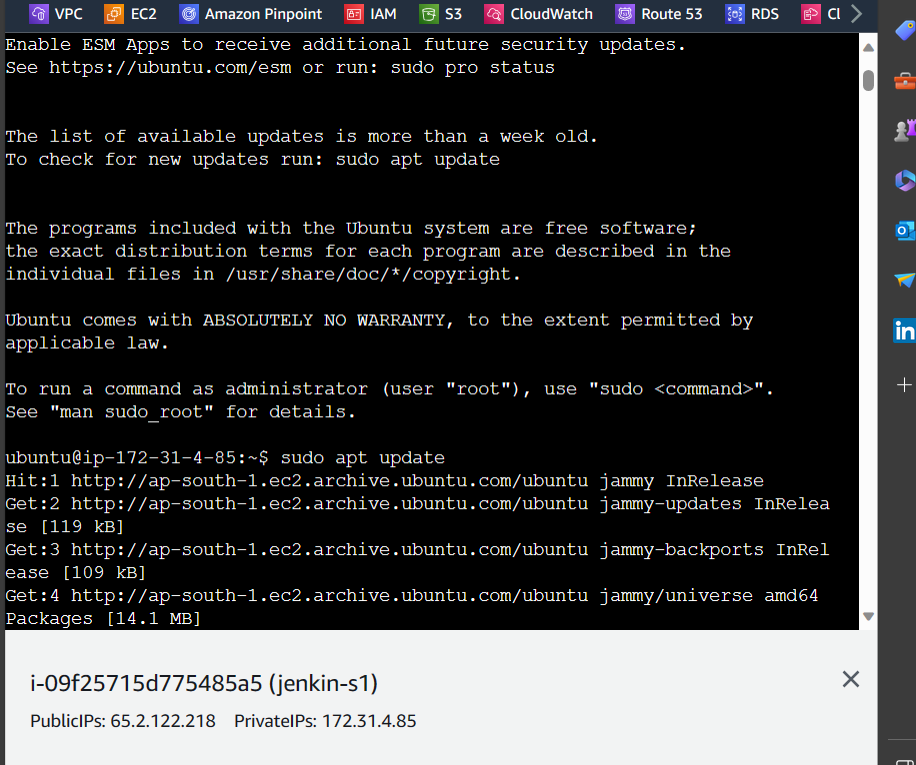
**Test:1**

Tasks To Be Performed:

1. Trigger a pipeline using Git when push on develop branch

2. Pipeline should pull Git content to a folder

Sudo apt update



Then install java

sudo apt install openjdk-11-jdk -y

[Linux (jenkins.io)](https://www.jenkins.io/doc/book/installing/linux/)

curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

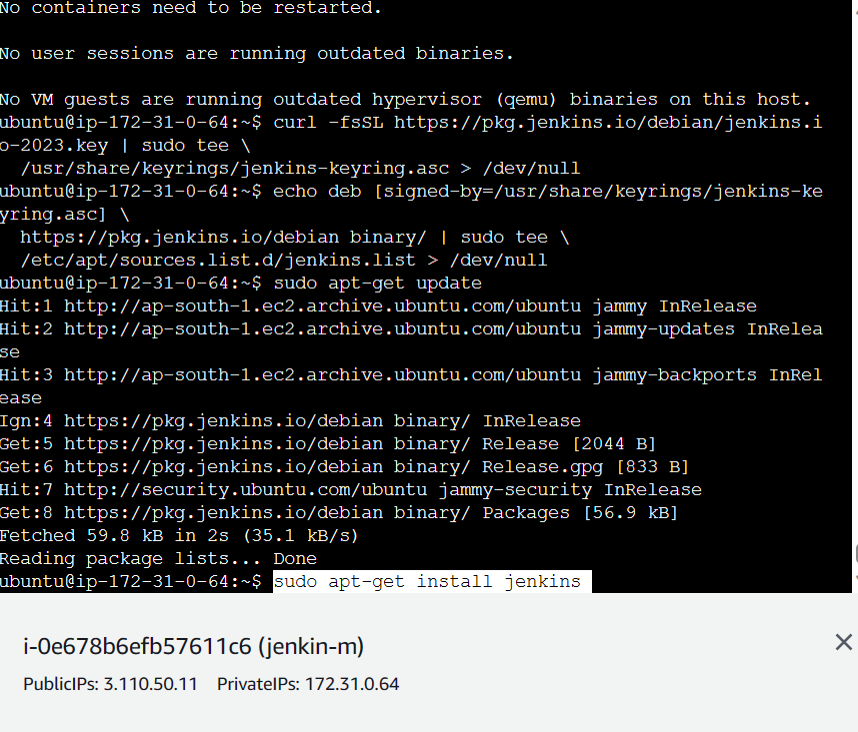
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

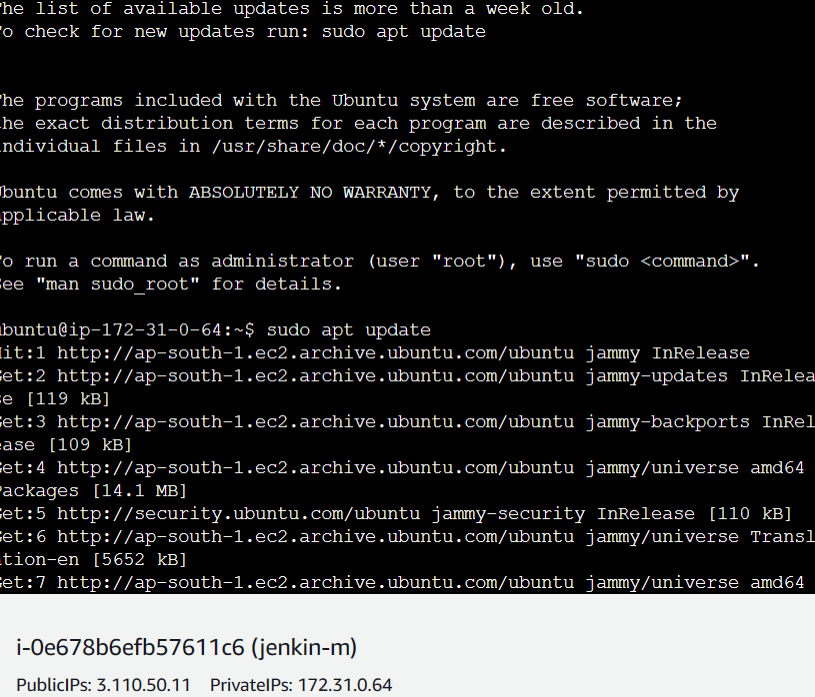
https://pkg.jenkins.io/debian binary/ | sudo tee \

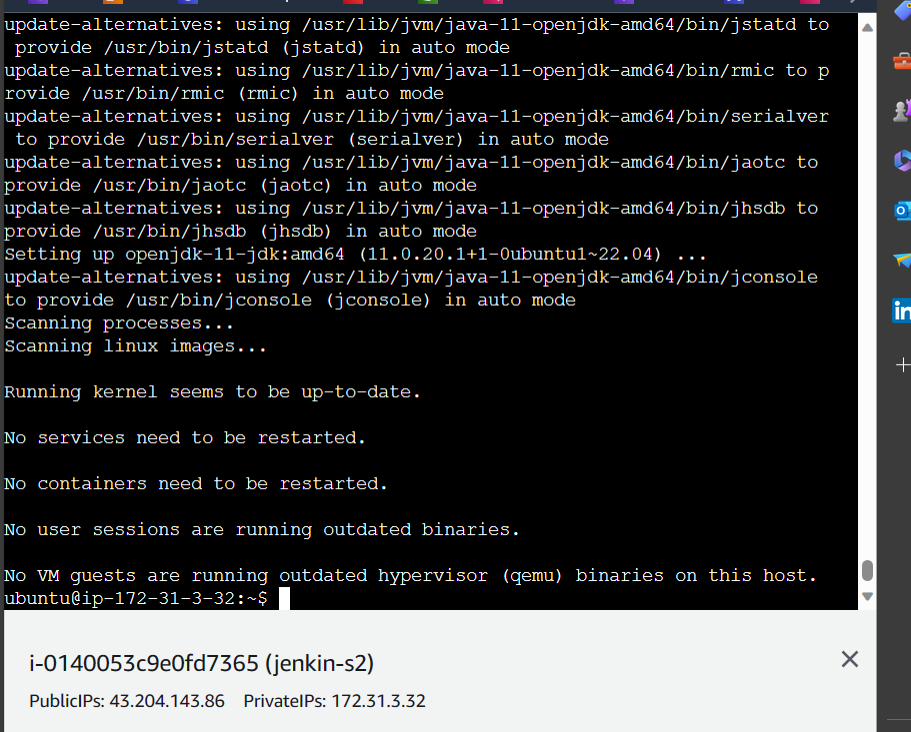
/etc/apt/sources.list.d/jenkins.list > /dev/null

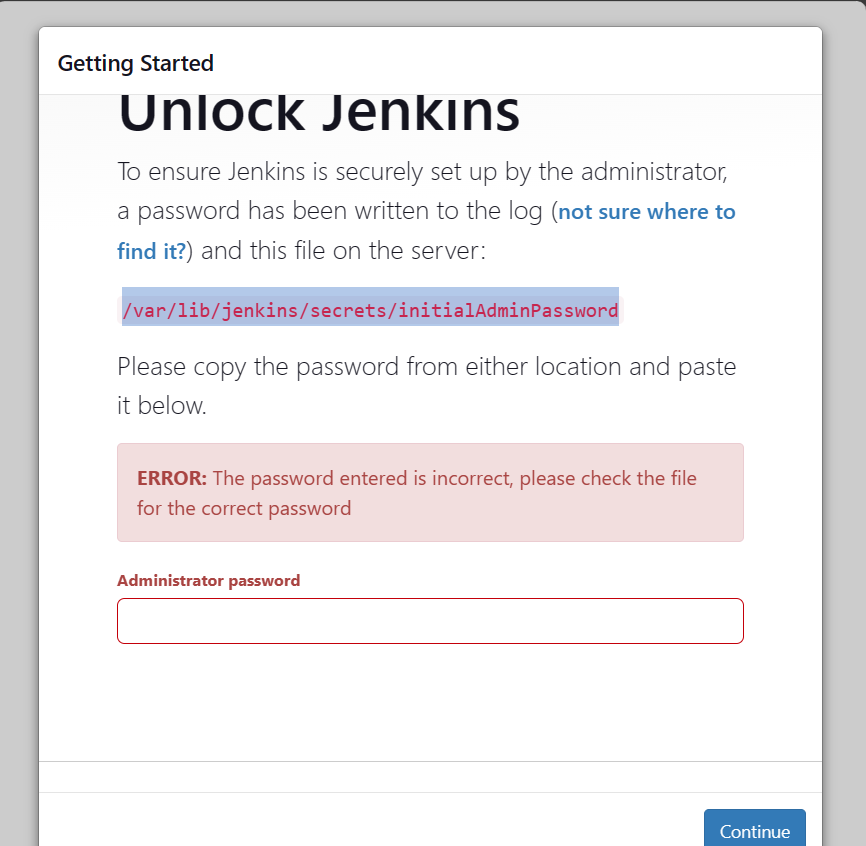
sudo apt-get update

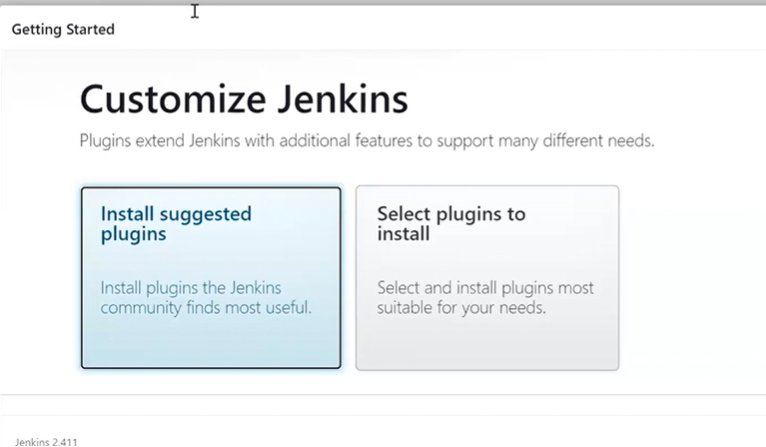
sudo apt-get install jenkins

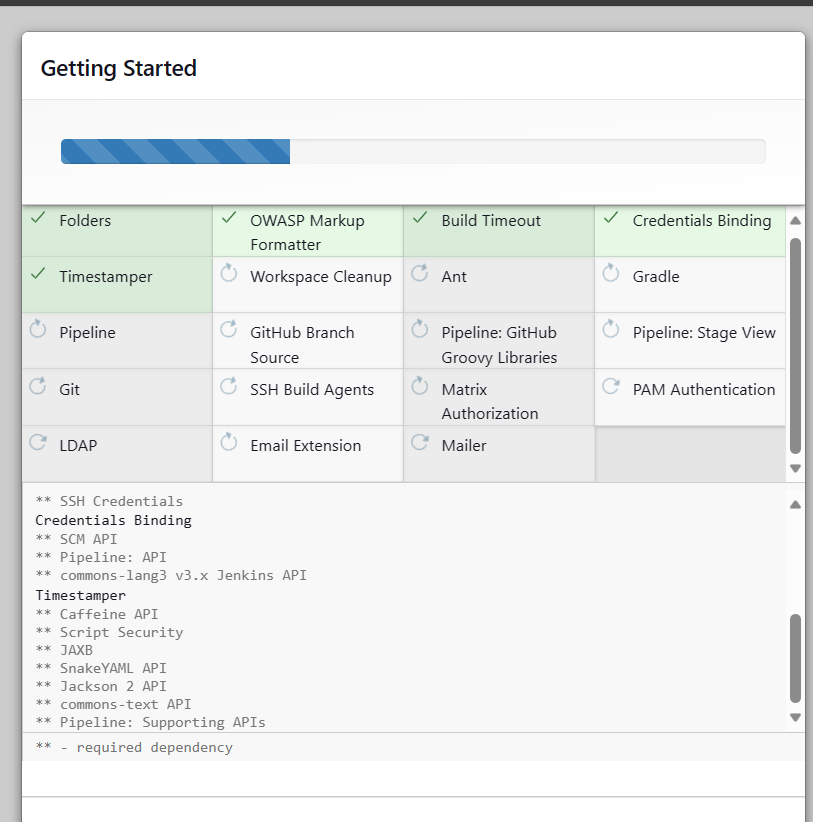


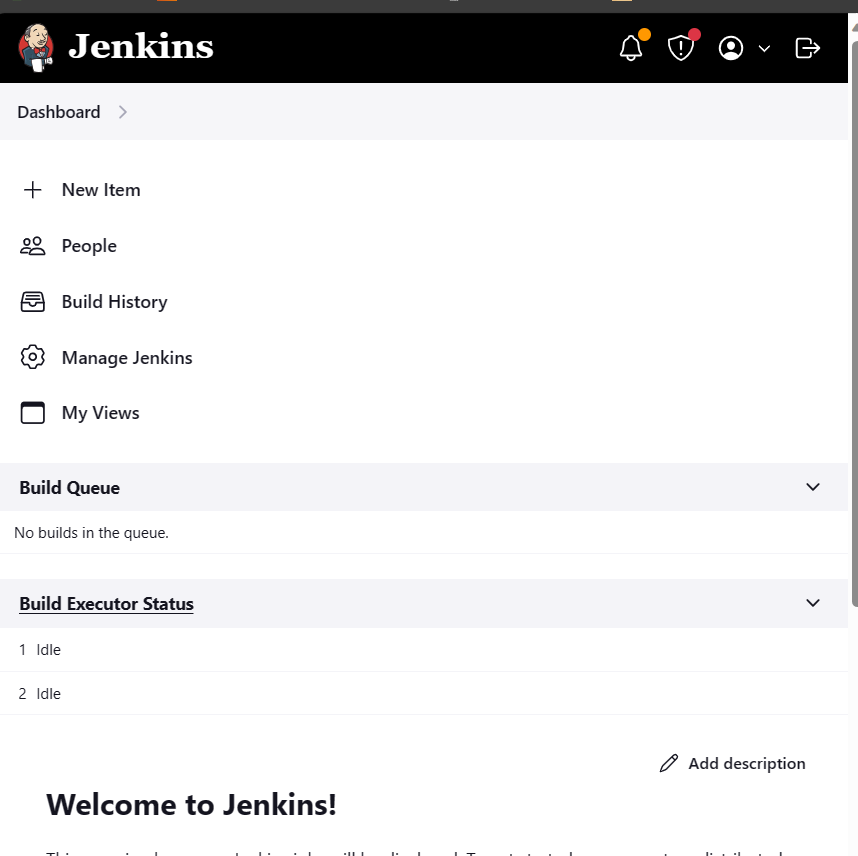


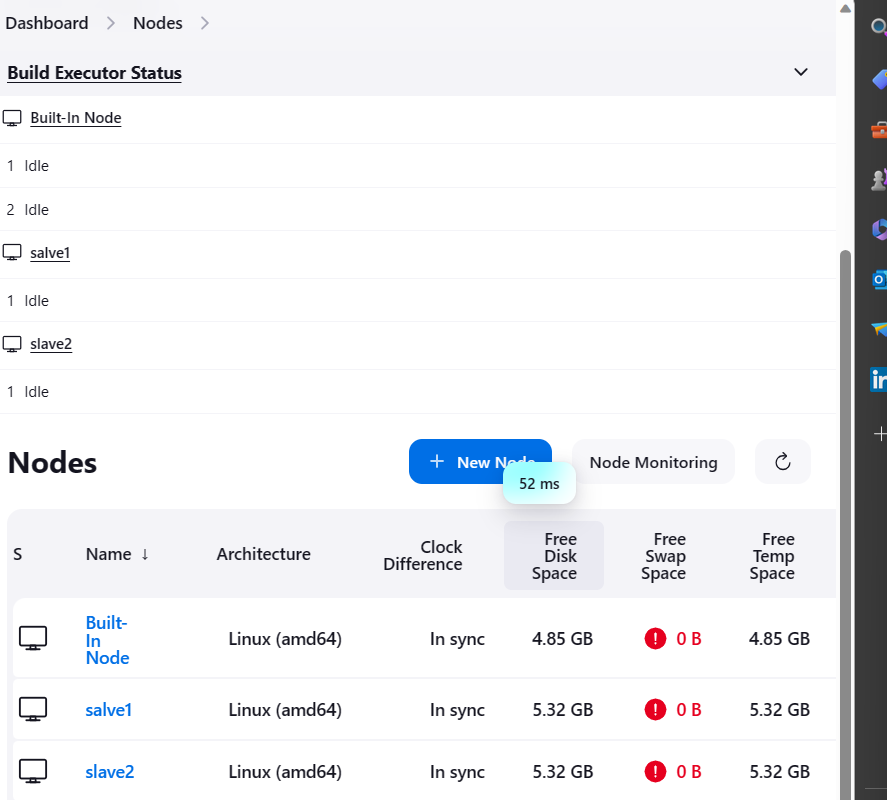


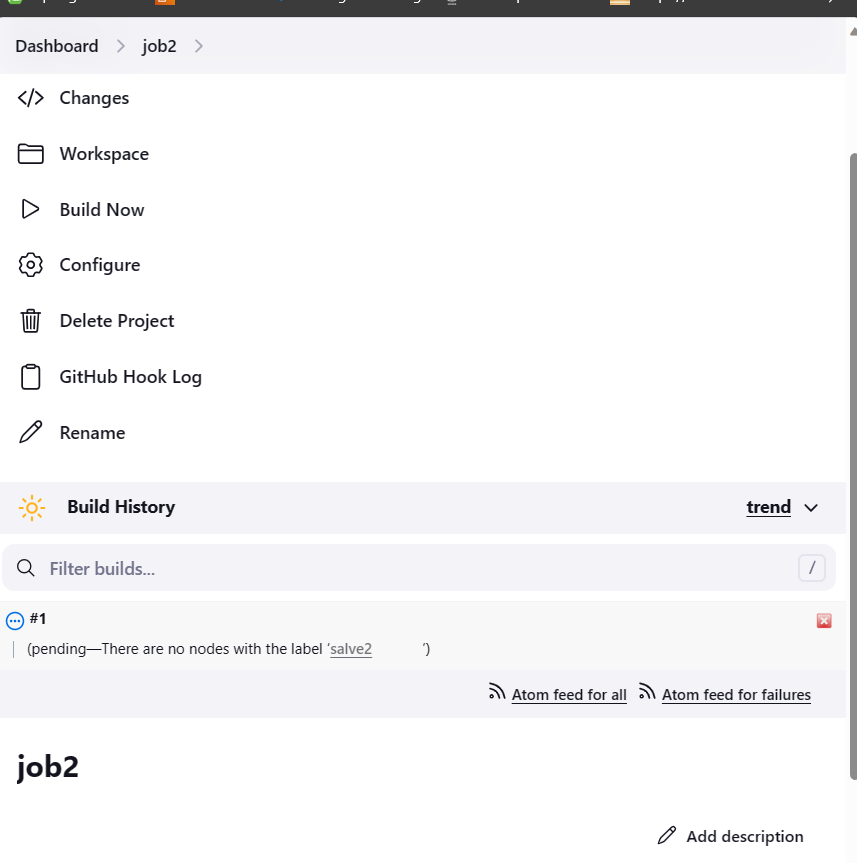


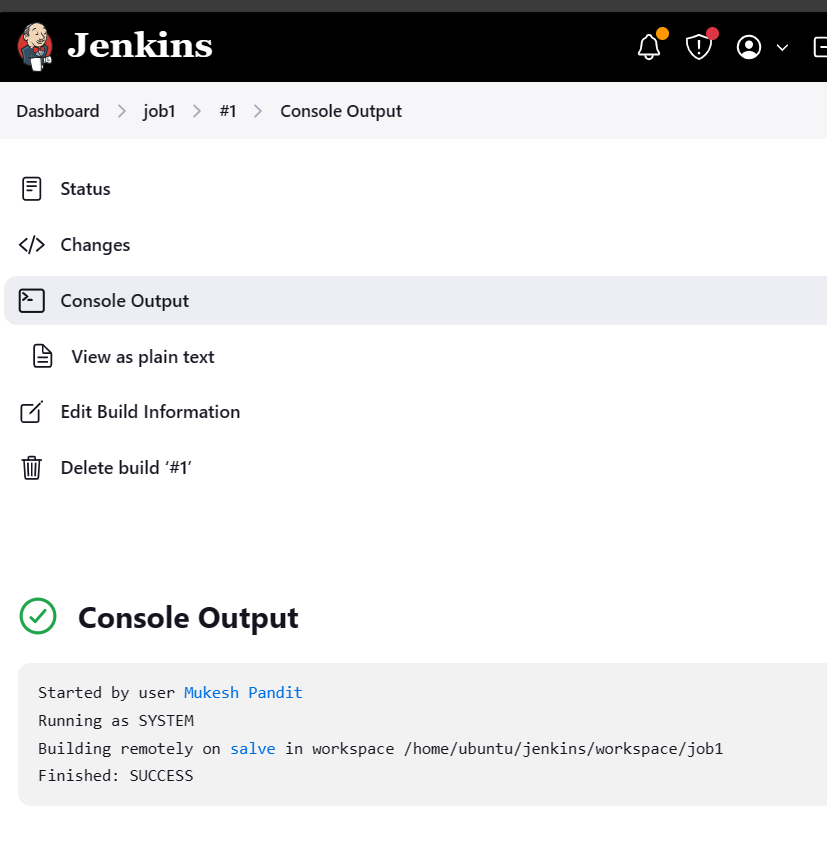


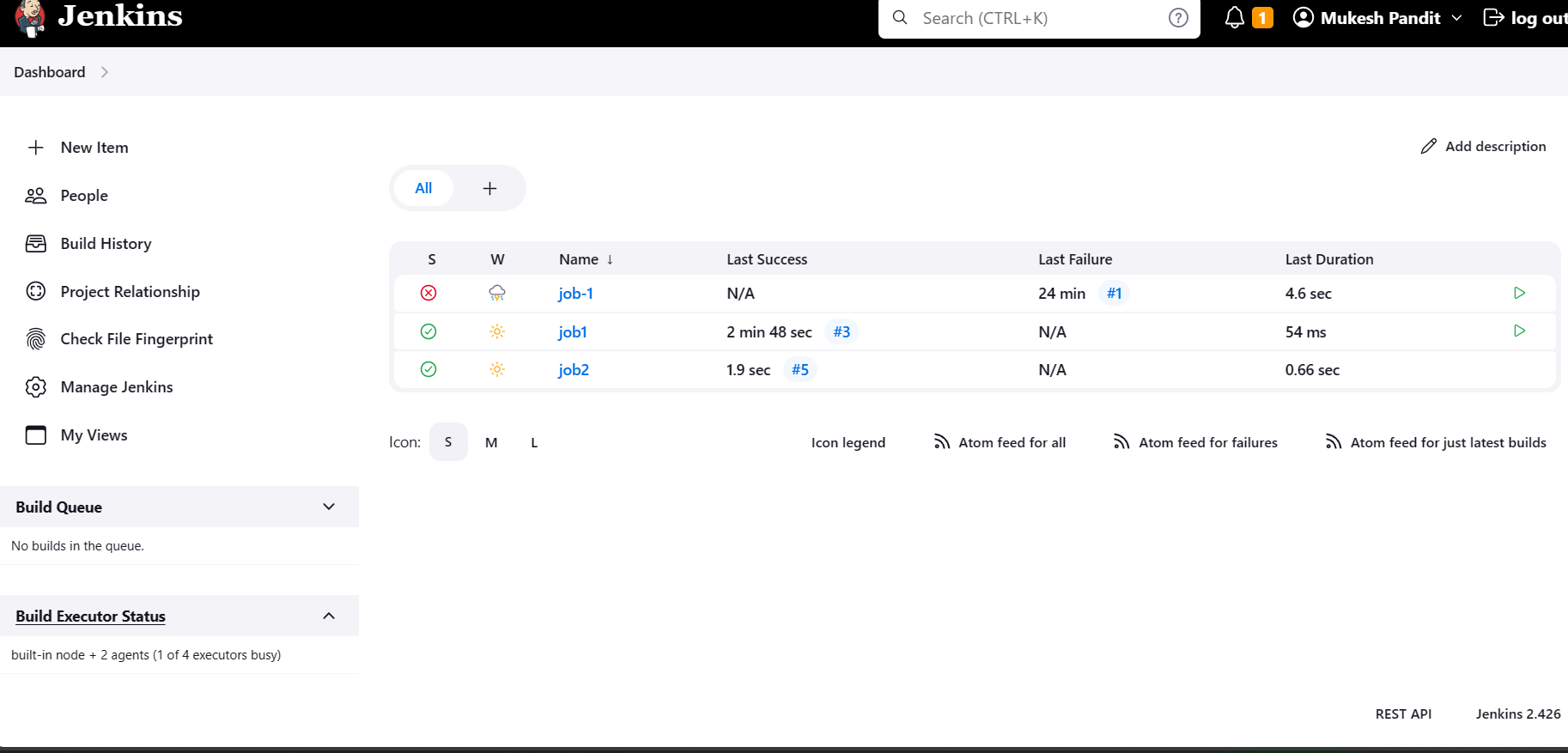












**Assignment 2**

**Test :2**

Tasks To Be Performed:

1. Add 2 nodes to Jenkins master

2. Create 2 jobs with the following jobs:

a. Push to test

b. Push to prod

3. Once a push is made to test branch, copy Git files to test server

4. Once a push is made to master branch, copy Git files to prod server

Let creat ec2 istance master and slave

Connect with java jdk +Jenkins to push the job

sudo apt-get install openjdk-11-jdk -y

sudo apt-get install openjdk-11-jdk -y

Nano install jenkin .sh

curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

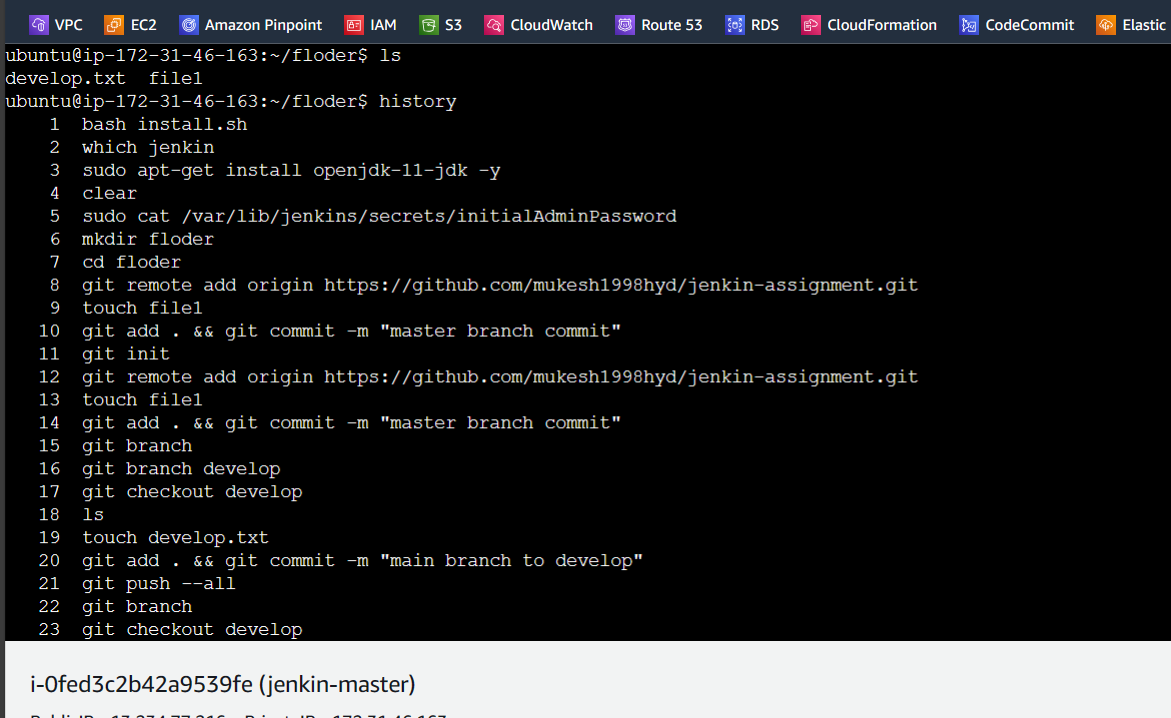
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update

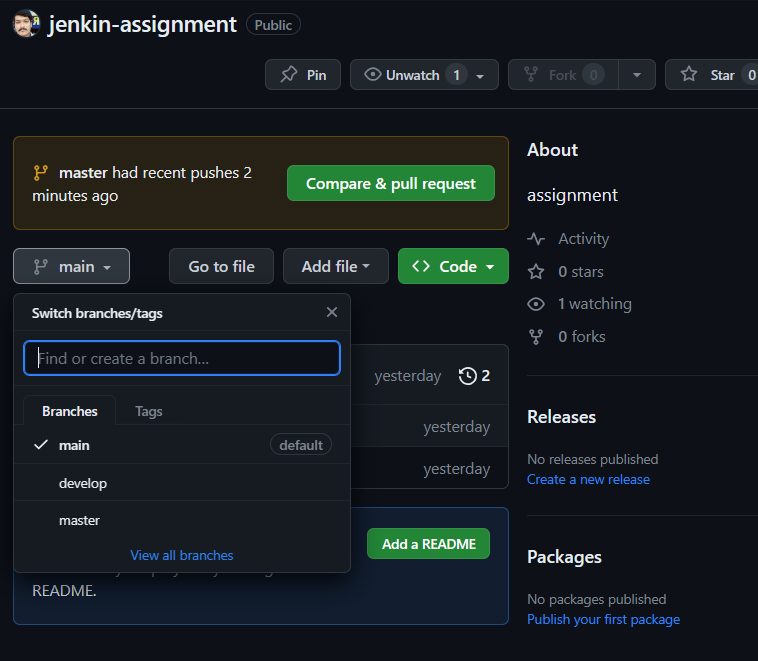
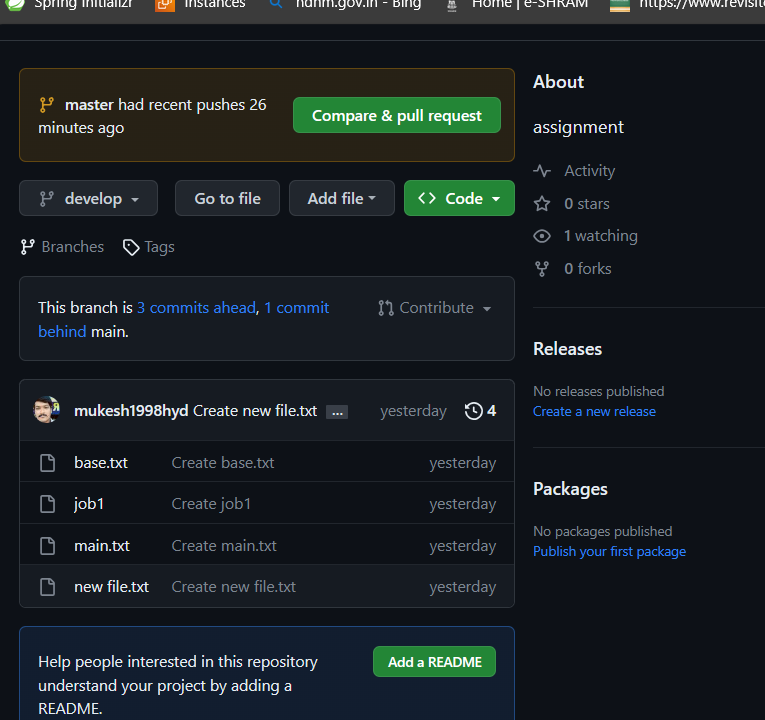
sudo apt-get install Jenkins -y

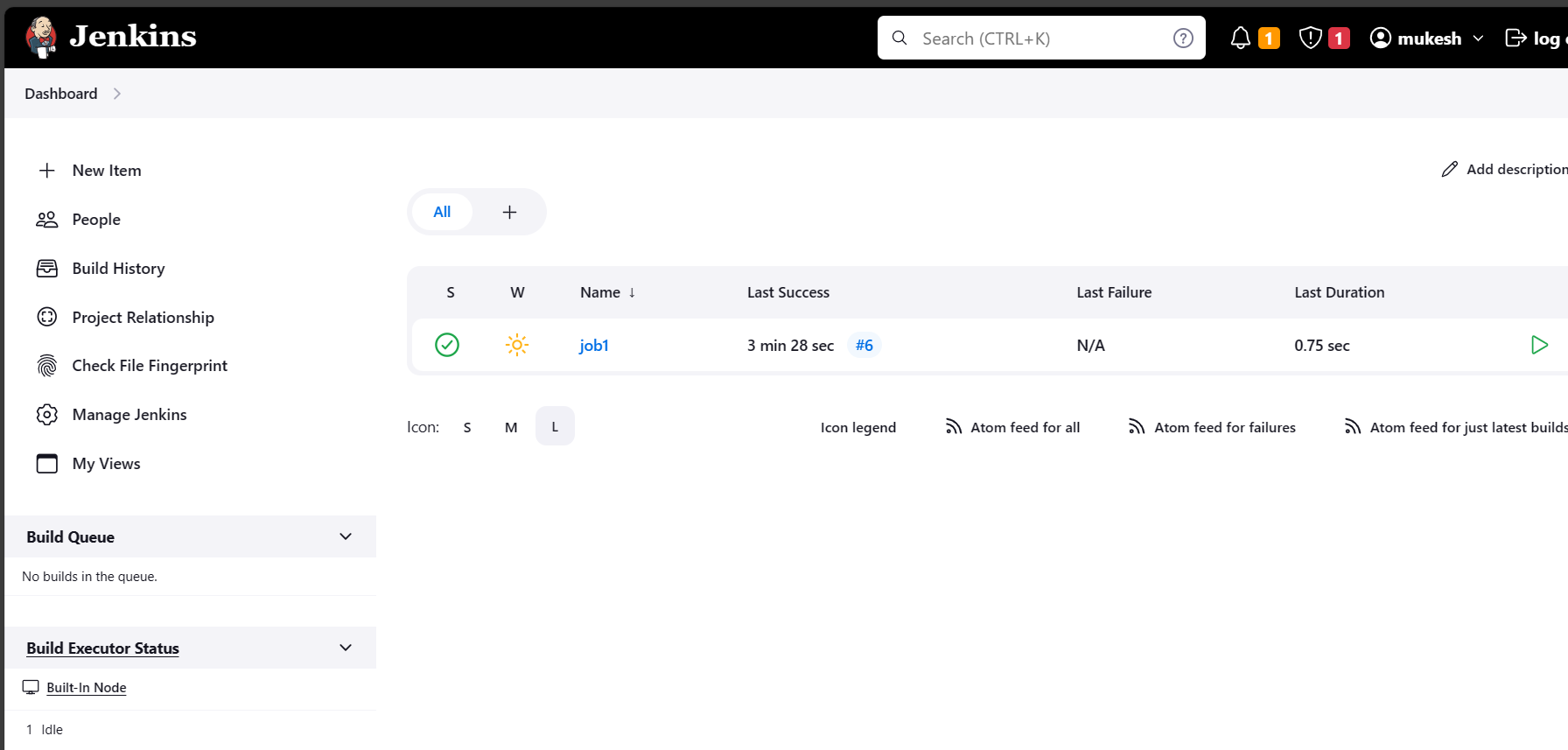


Checkout

Jenlins master

Commit the file of floder

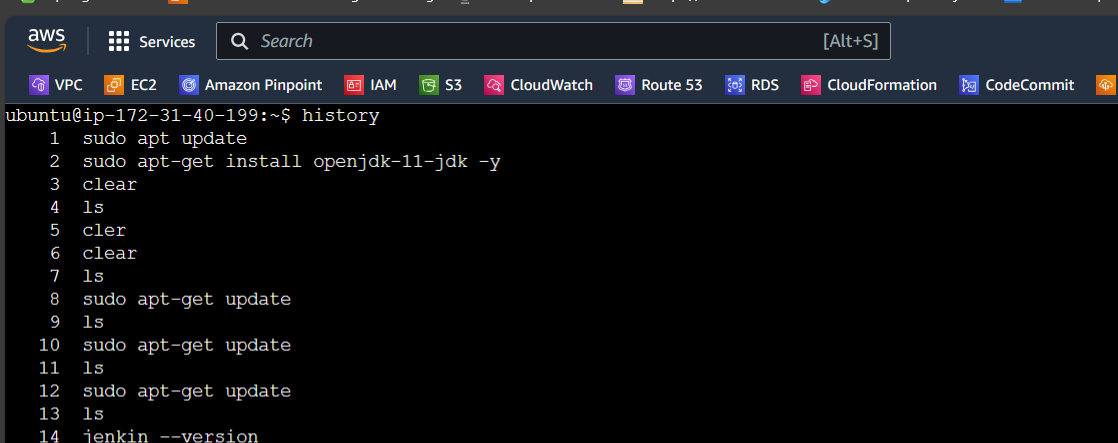


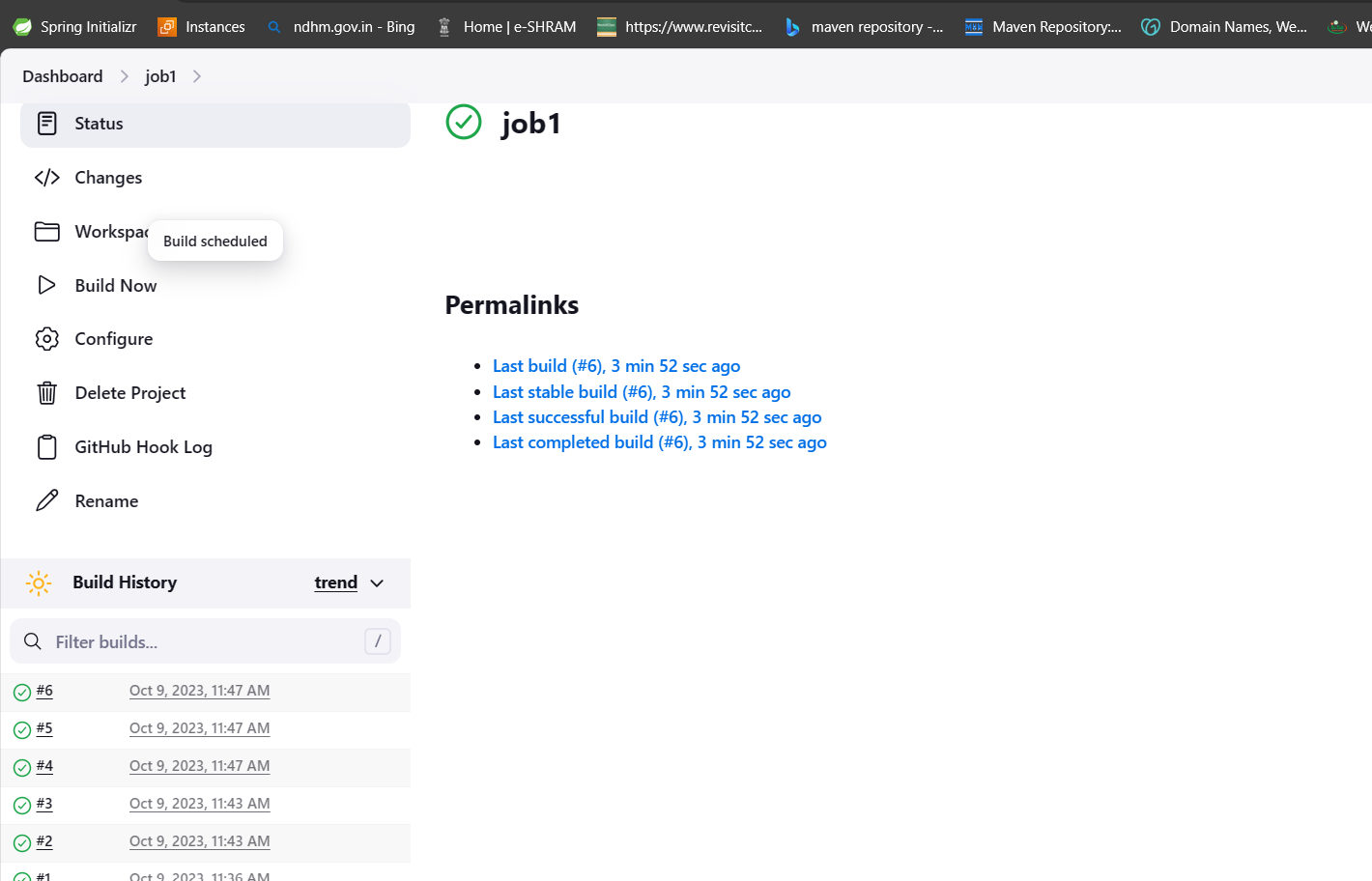


Then creat the job and slave file

Push all in master

Checkout





**Project 1**

**Capstone project**

You have been hired as a Sr. DevOps Engineer in Abode Software. They want to implement DevOps Lifecycle in their company. You have been asked to implement this lifecycle as fast as possible. Abode Software is a product-based company and their product is available on this GitHub link.

<https://github.com/hshar/website.git>

Following are the specifications of the lifecycle

: 1. Install the necessary software on the machines using a configuration management tool

2. Git workflow has to be implemented

3. CodeBuild should automatically be triggered once a commit is made to master branch or develop branch. a. If a commit is made to master branch, test and push to prod b. If a commit is made to develop branch, just test the product, do not push to prod

4. The code should be containerized with the help of a Dockerfile. The Dockerfile should be built every time there is a push to GitHub. Use the following pre-built container for your application: hshar/webapp The code should reside in '/var/www/html'

5. The above tasks should be defined in a Jenkins Pipeline with the following jobs:

a. Job1 : build

b. Job2 : test

c. Job3:prod

let create a master :git ,anisble,Jenkins,docker,java

connect to slaves1 : java,docker (anisble use)

slaves2 : java ,docker(anisble use)

Ansible installation commands

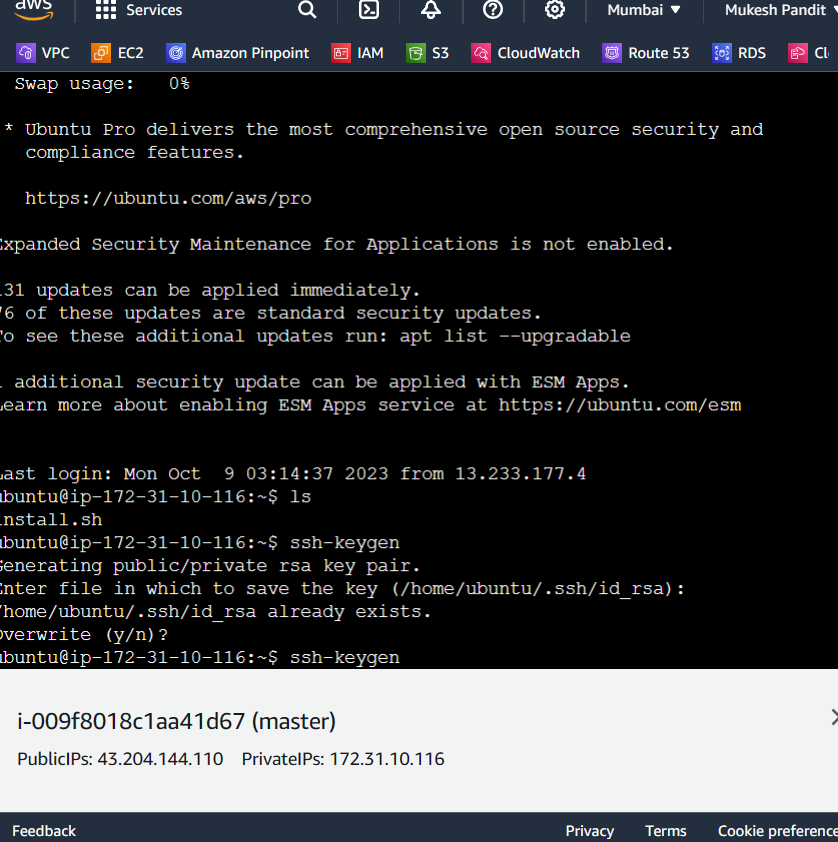
--------------------------------

sudo apt update

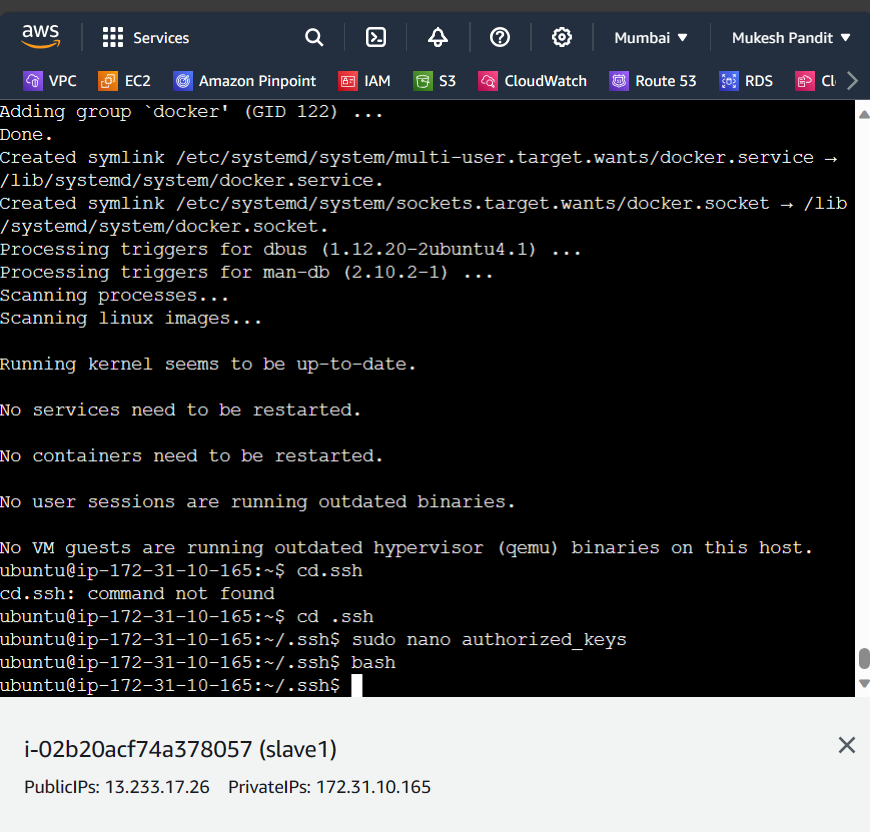
sudo apt install software-properties-common

sudo add-apt-repository --yes --update ppa:ansible/ansible

sudo apt install ansible



Master1



Slave1 and slave 2 create EC2 instance

Some command to write install in slave1 and 2

(create file slave.sh and paste commands present below)

sudo apt-get update

sudo apt-get install openjdk-11-jdk -y

sudo apt-get install docker.io -y

(create file master.sh and paste commands present below)

(please check the latest jenkins installation commands if you are using weekly release commands)

sudo apt-get update

sudo apt-get install openjdk-11-jdk -y

sudo apt-get install docker.io -y

curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

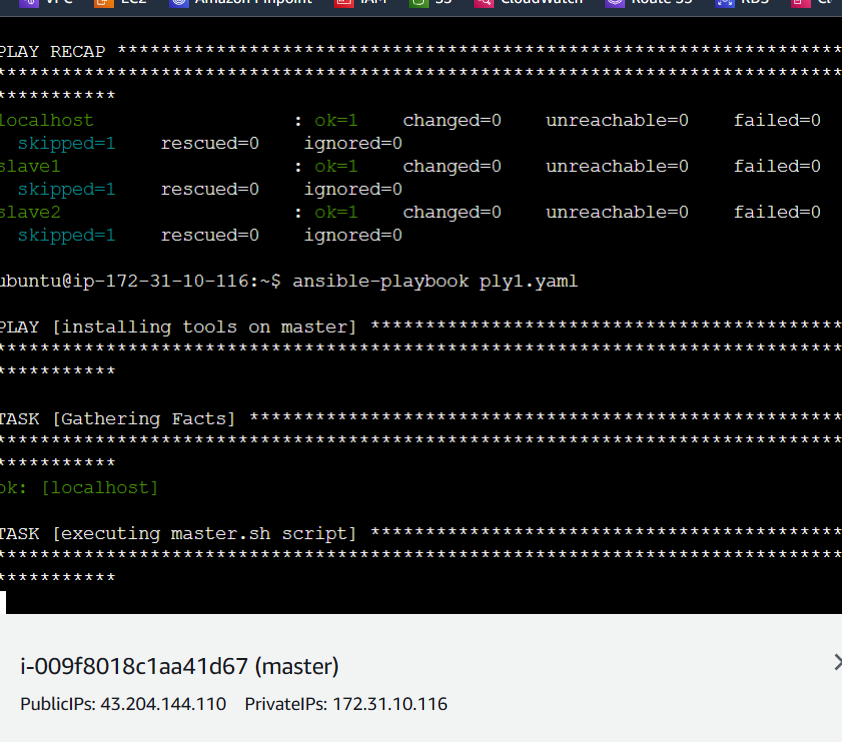
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update

sudo apt-get install jenkins -y



Install all yaml file

---

- name: installing tools on master

hosts: localhost

become: true

tasks:

- name: executing master.sh script

script: master.sh

- name: installing tools on slaves

hosts: slave1

become: true

tasks:

- name: executing slave.sh script

script: slave.sh

- name: installing tools on slaves

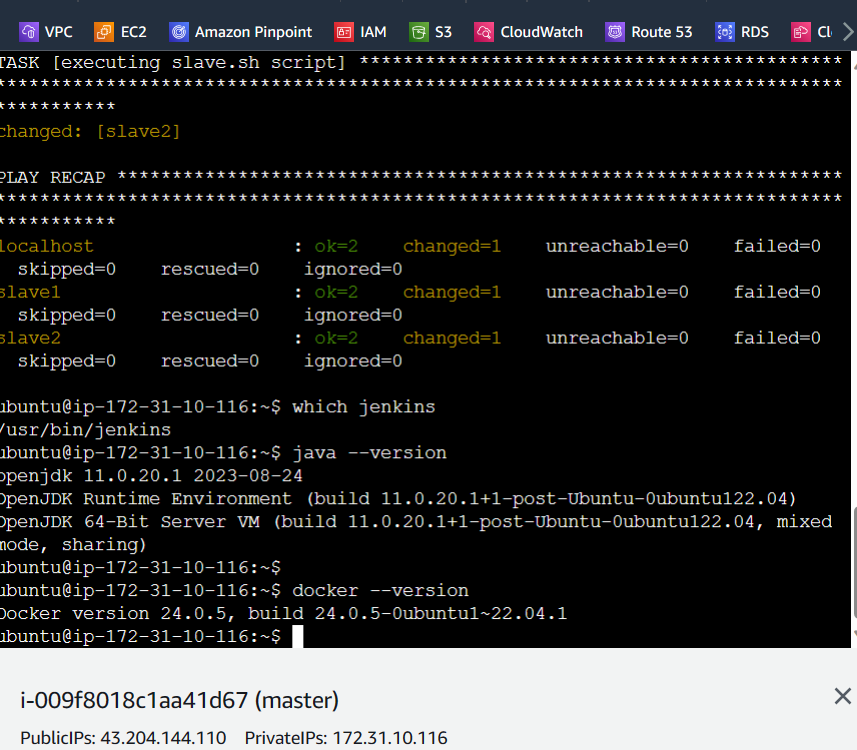
hosts: slave2

become: true

tasks:

- name: executing slave.sh script

script: slave.sh



Check it all install successful in master,slave1 and slave2

Jenkins Execute shell commands:

===============================================

sudo docker build . -t finalrelease

sudo docker run -itd -p 80:80 finalrelease

===============================================

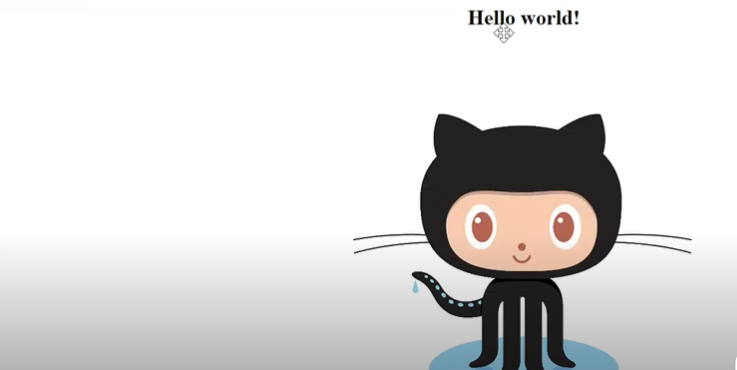
Running for second time:

===============================================

sudo docker rm -f $(sudo docker ps -a -q)

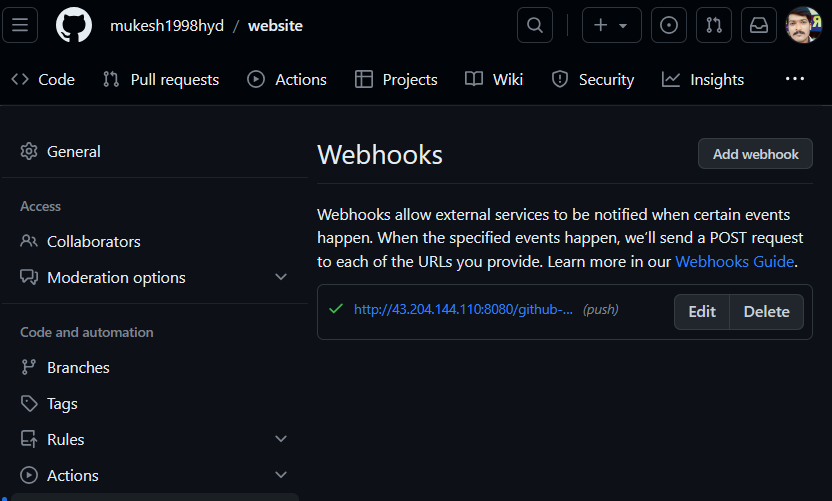
copy the ip address then paste and

out put



Afterwords go github to webhook to instance

To manual setting



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*completed project \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*