DEVEOPS CAPSTONE : PROJECT II

You are hired as a DevOps Engineer for Analytics Pvt Ltd. This company is a product based organization which uses Docker for their containerization needs within the company. The final product received a lot of traction in the first few weeks of launch. Now with the increasing demand, the organization needs to have a platform for automating deployment, scaling and operations of application containers across clusters of hosts. As a DevOps Engineer, you need to implement a DevOps lifecycle such that all the requirements are implemented without any change in the Docker containers in the testing environment.

Up until now, this organization used to follow a monolithic architecture with just 2 developers. The product is present on: https://github.com/hshar/website.git

Following are the specifications of the lifecycle:

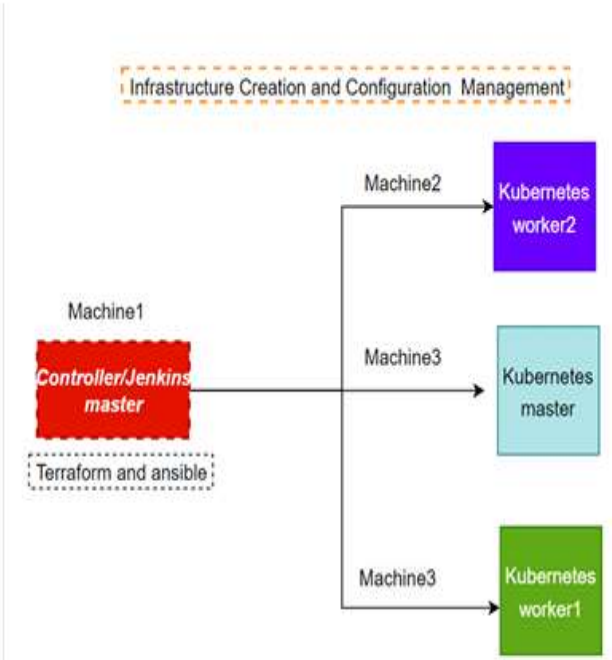
1. Git workflow should be implemented. Since the company follows a monolithic architecture of development, you need to take care of version control. The release should happen only on the 25th of every month.
2. CodeBuild should be triggered once the commits are made in the master branch.
3. The code should be containerized with the help of the Dockerfile. The Dockerfile should be built every time if there is a push to GitHub. Create a custom Docker image using a Dockerfile.
4. As per the requirement in the production server, you need to use the Kubernetes cluster and the containerized code from Docker Hub should be deployed with 2 replicas. Create a NodePort service and configure the same for port 30008.
5. Create a Jenkins Pipeline script to accomplish the above task.
6. For configuration management of the infrastructure, you need to deploy the configuration on the servers to install necessary software and configurations.
7. Using Terraform, accomplish the task of infrastructure creation in the AWS cloud provider.

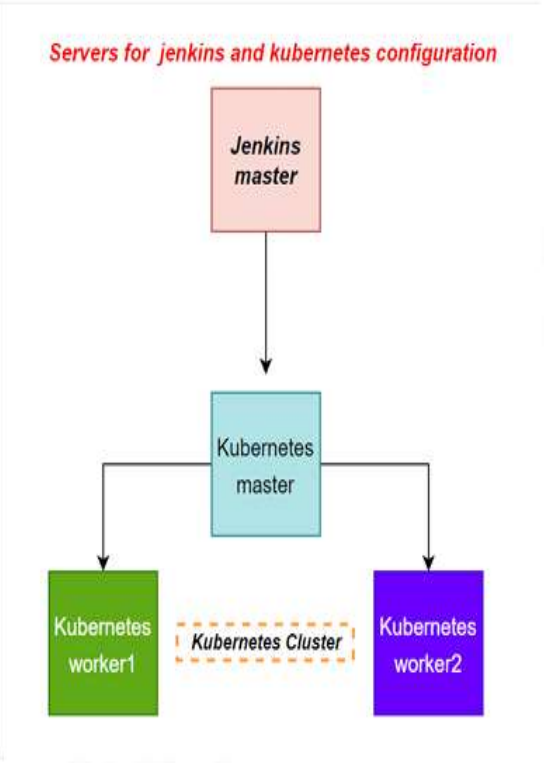
Architectural Advice

Architectural Advice:
Softwares to be installed on the respective machines using configuration management.

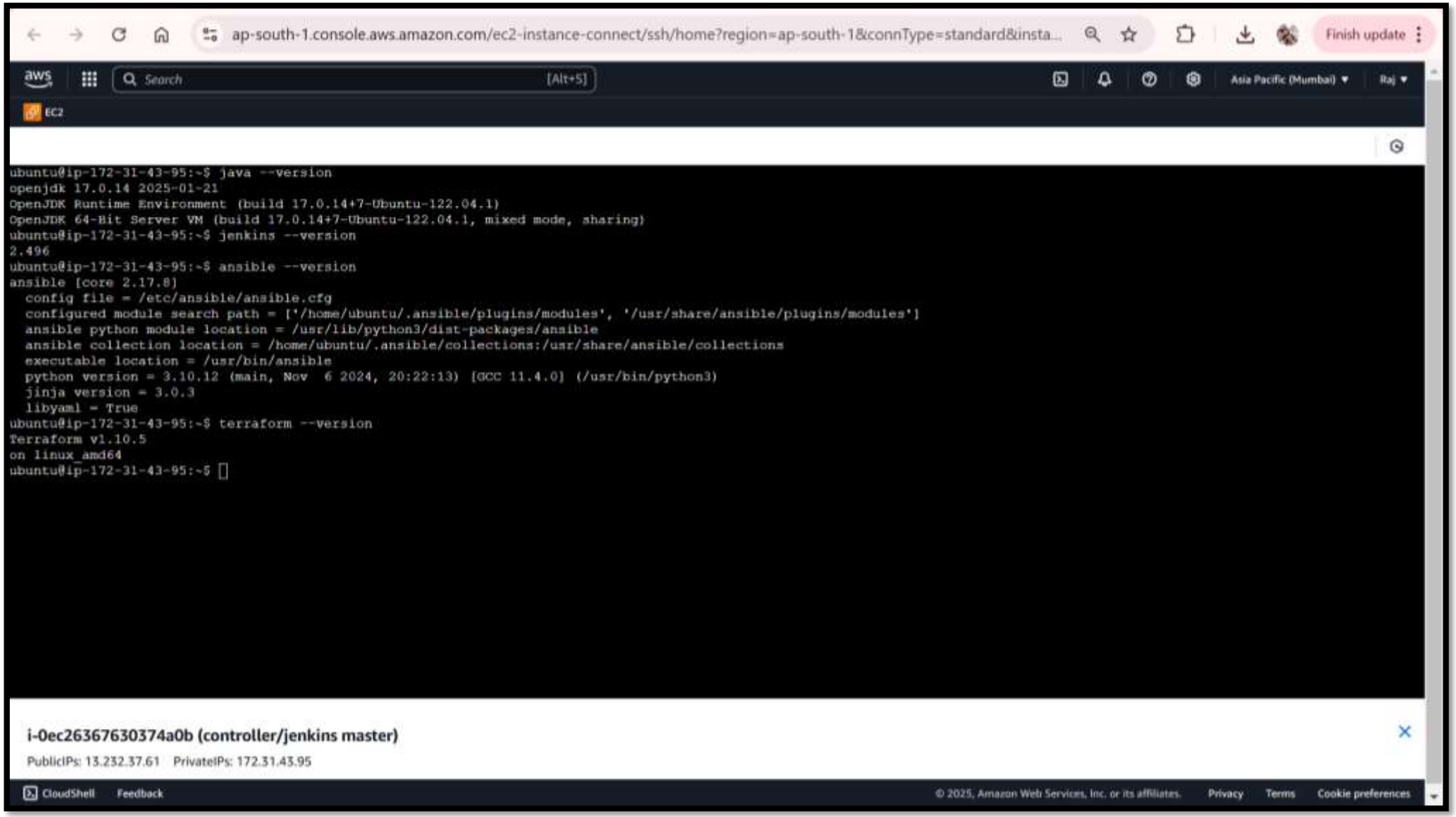
Worker1: Jenkins, Java

Worker2: Docker, Kubernetes
Worker3: Java, Docker, Kubernetes
Worker4: Docker, Kubernetes

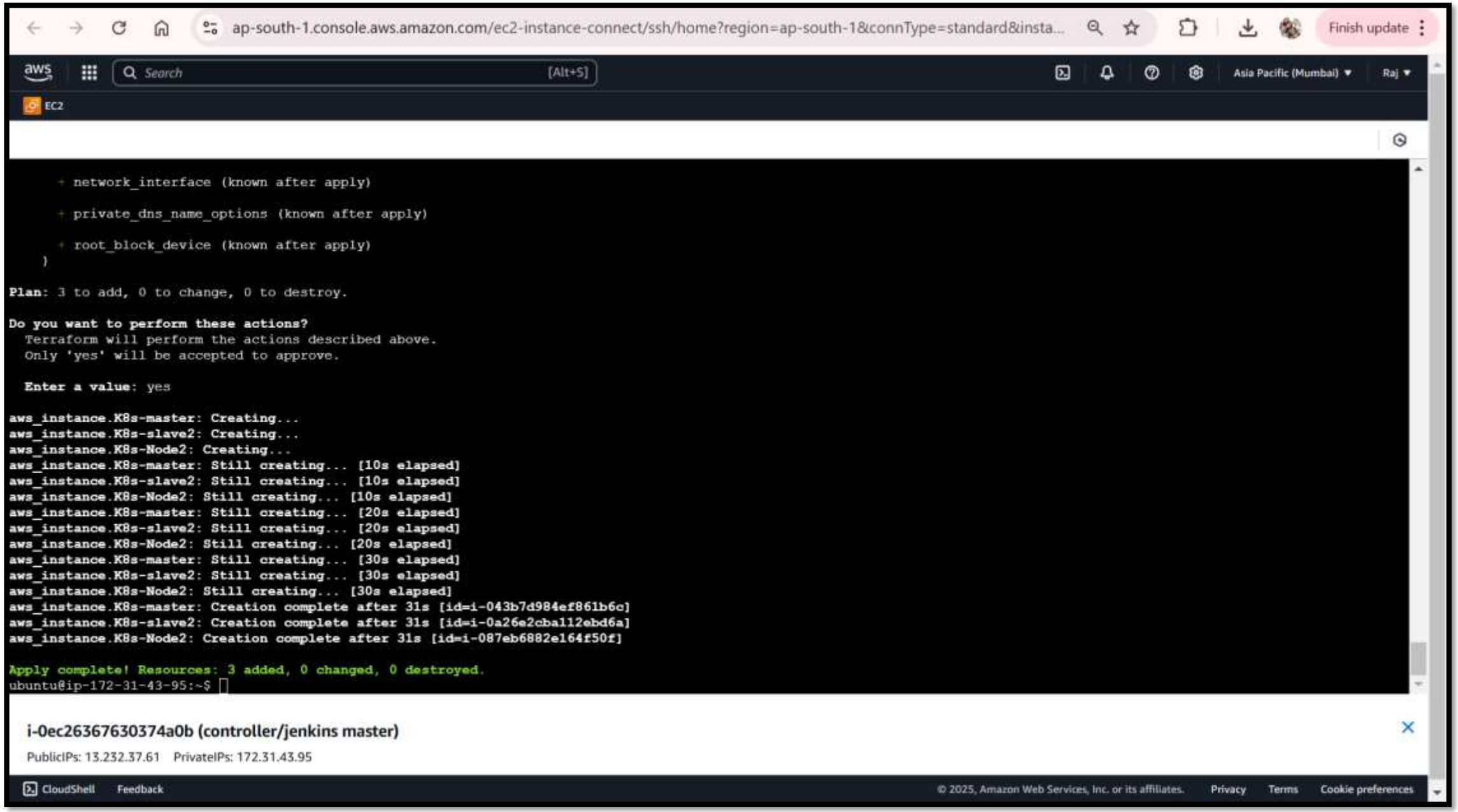




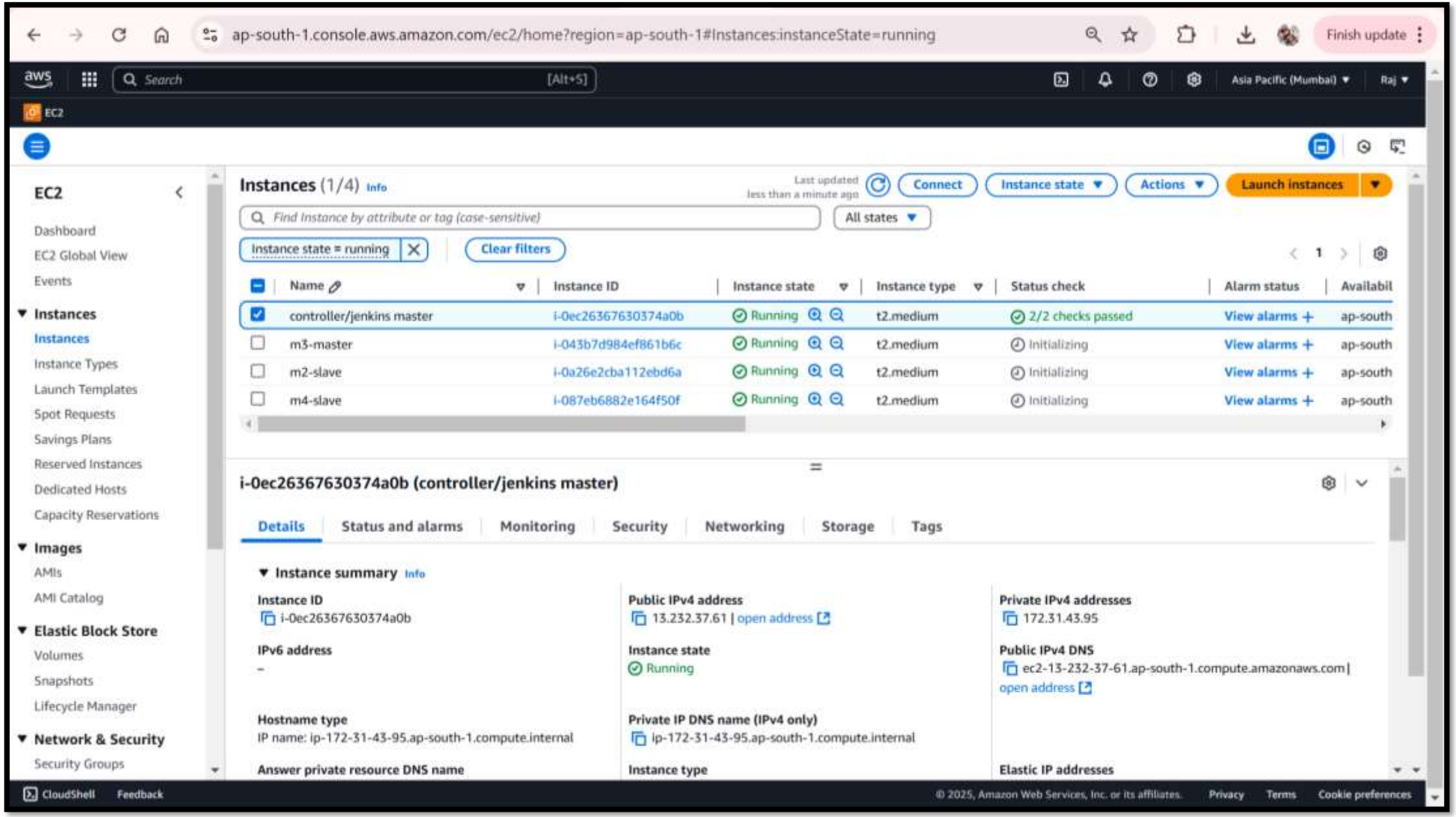
In Master-Jenkins Instance
Jenkins, Java, Ansible, Terraform Install



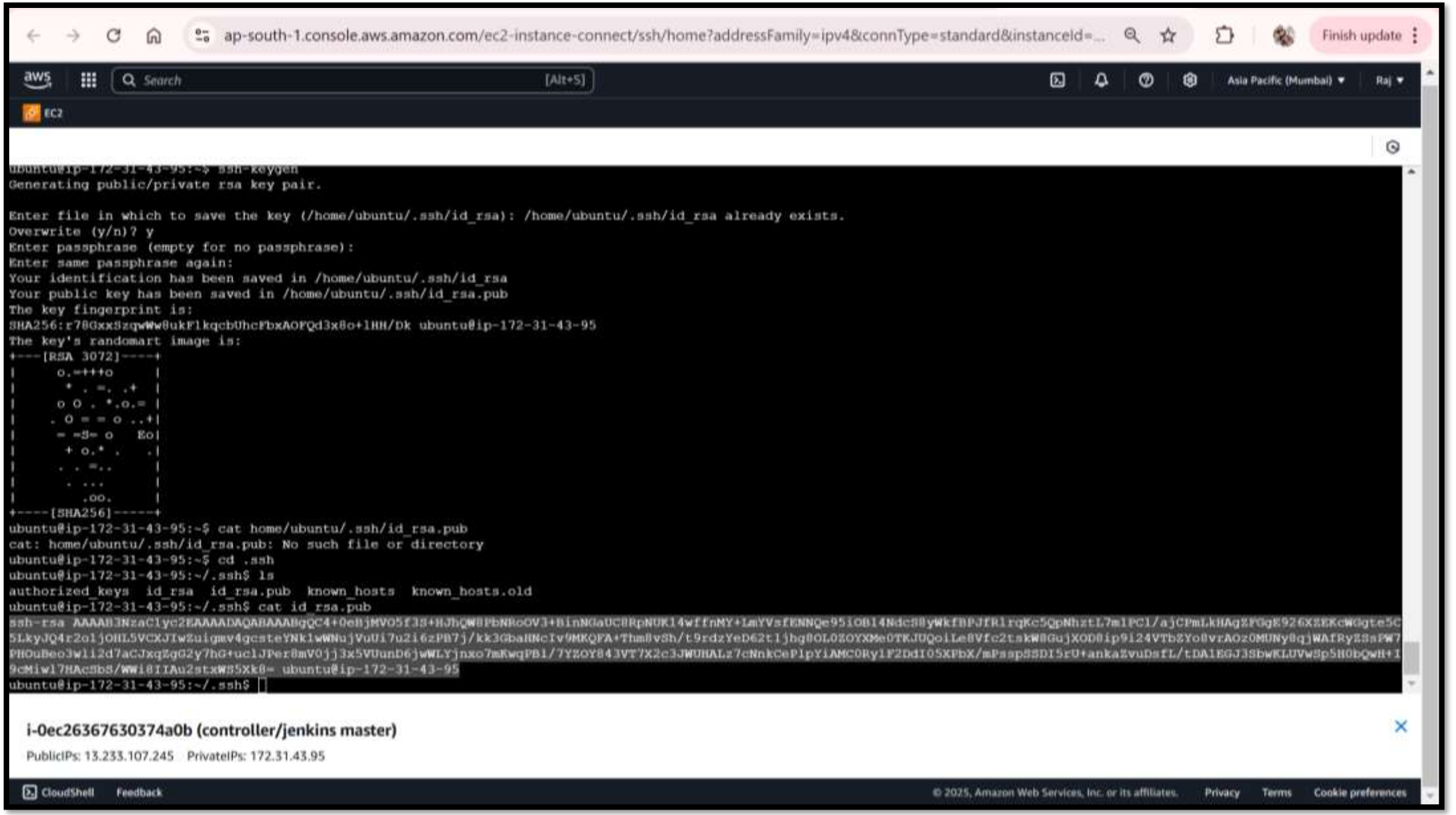
Terraform apply



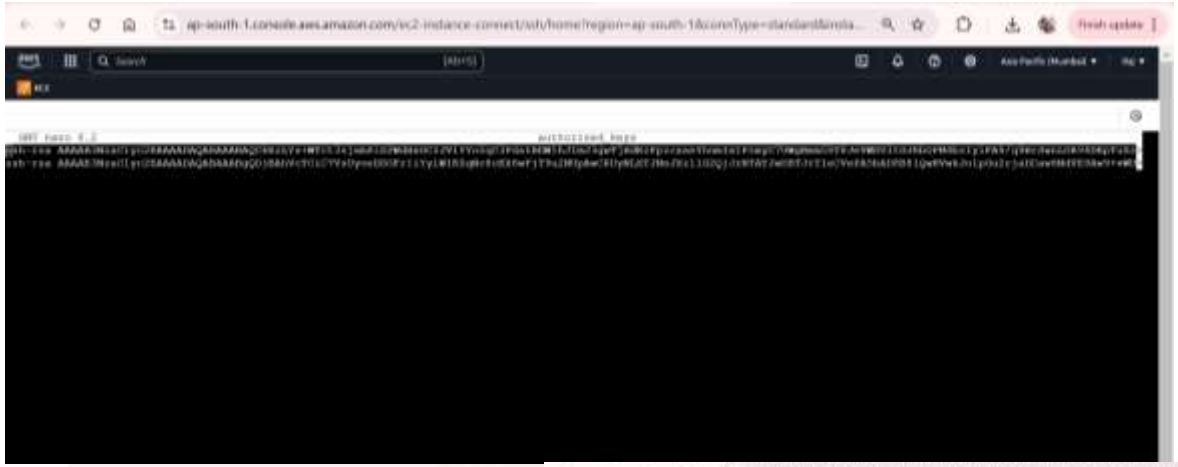
Master & worker Machine

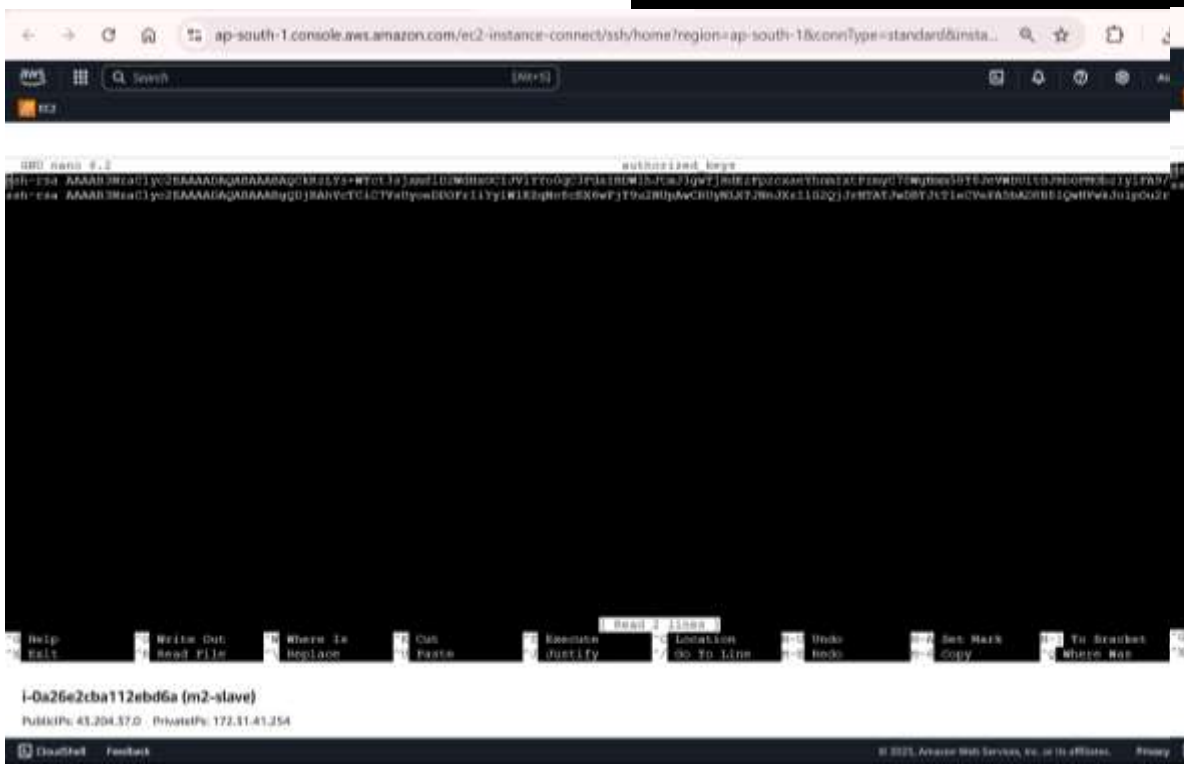


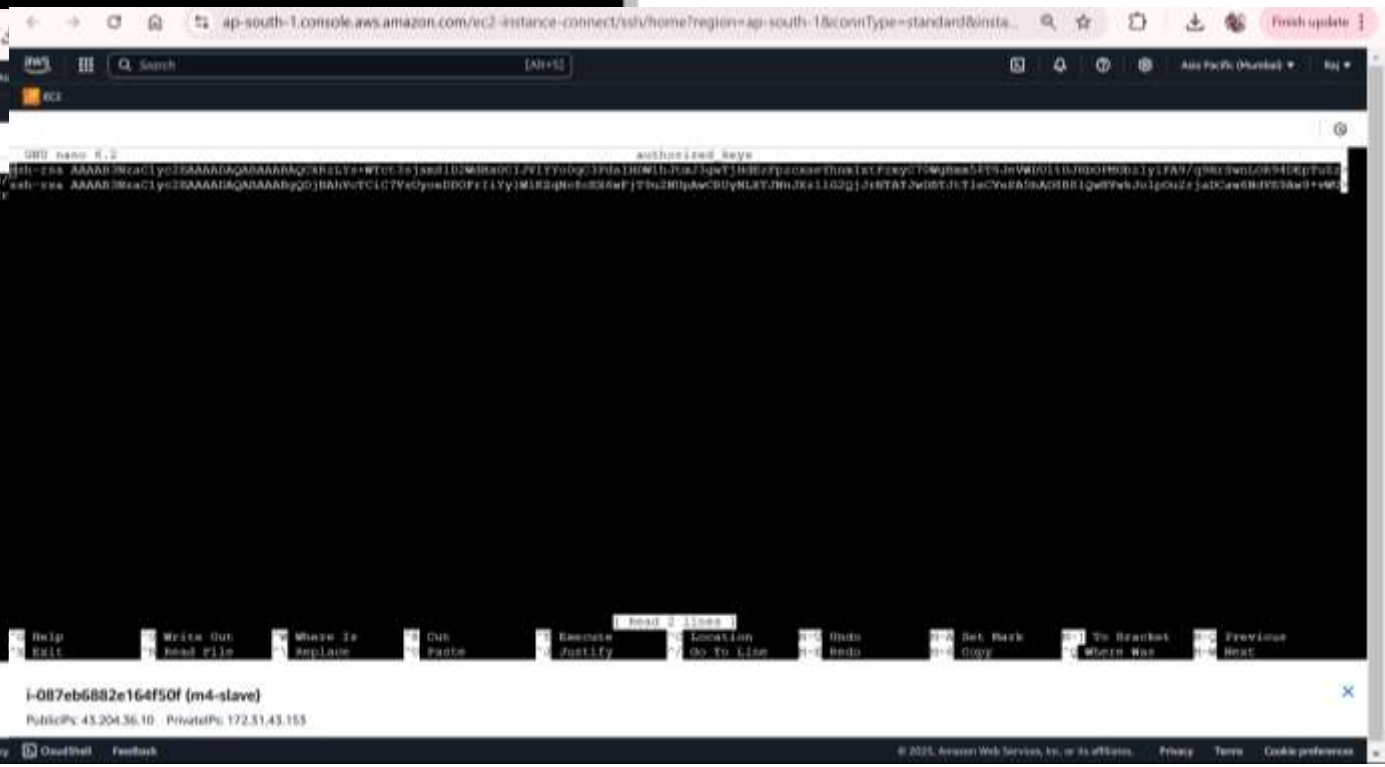
Generating Ssh public key



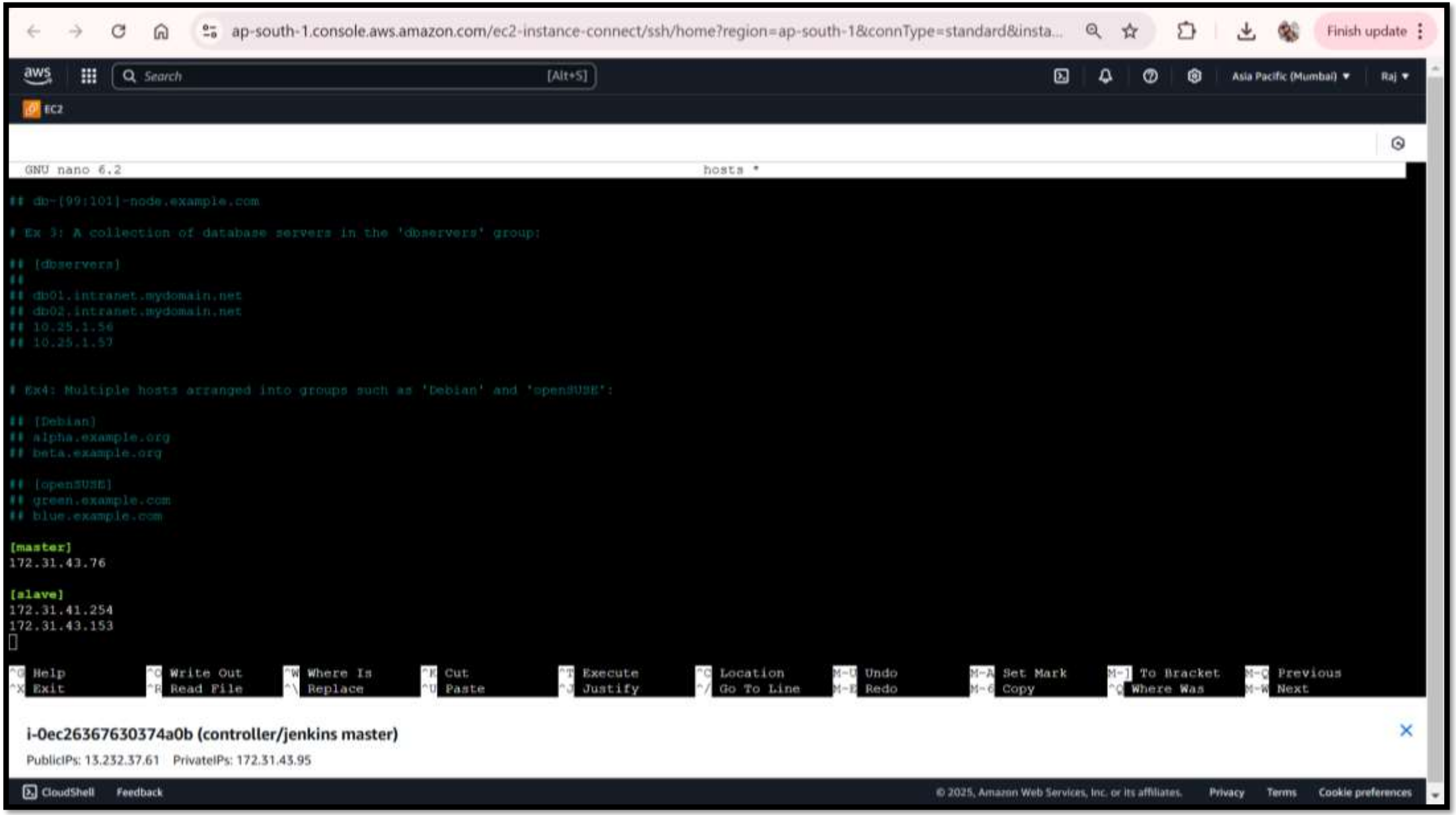
Ssh connection master with slaves machine



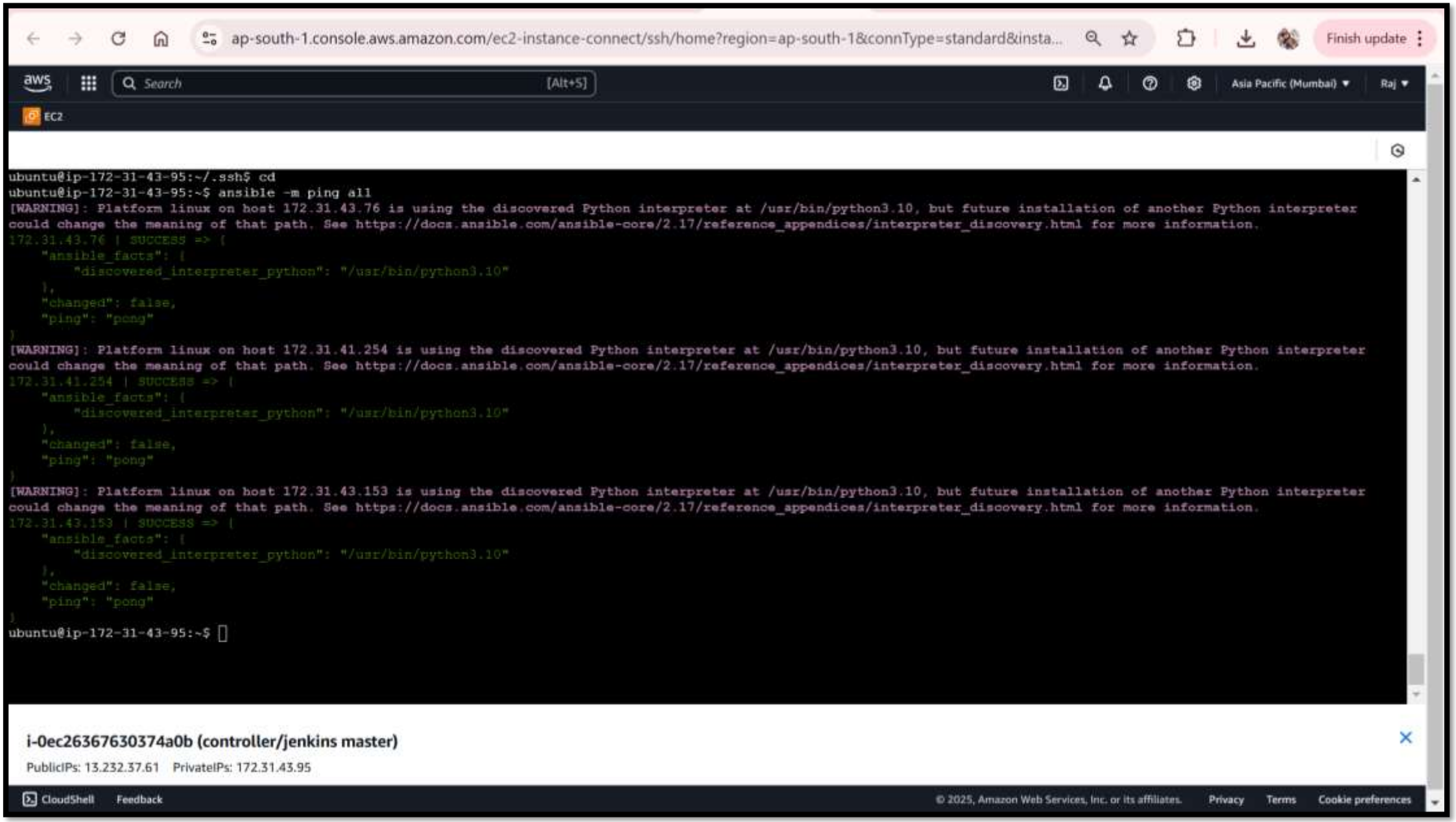




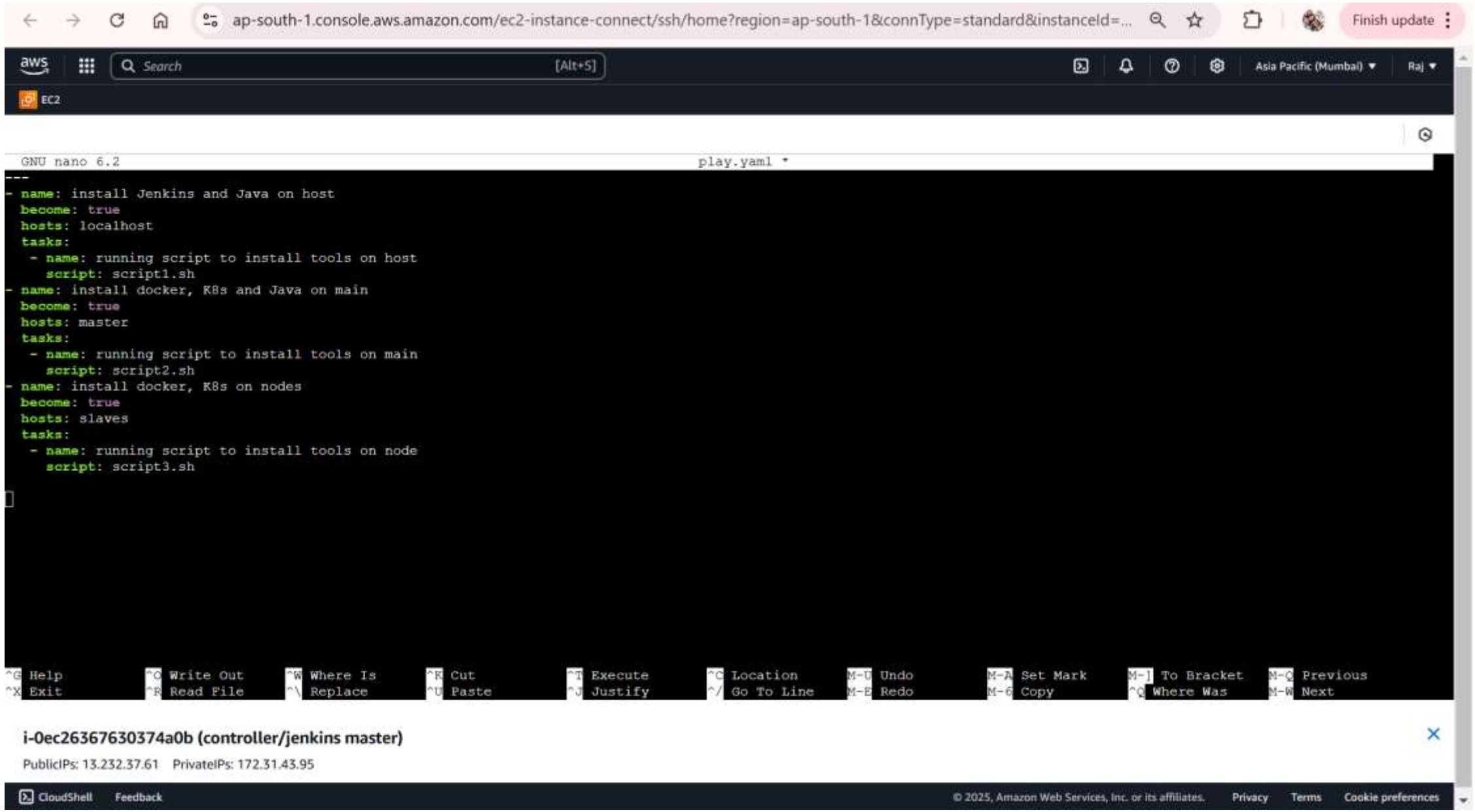
Adding private Ips of masters and slaves machine in hosts

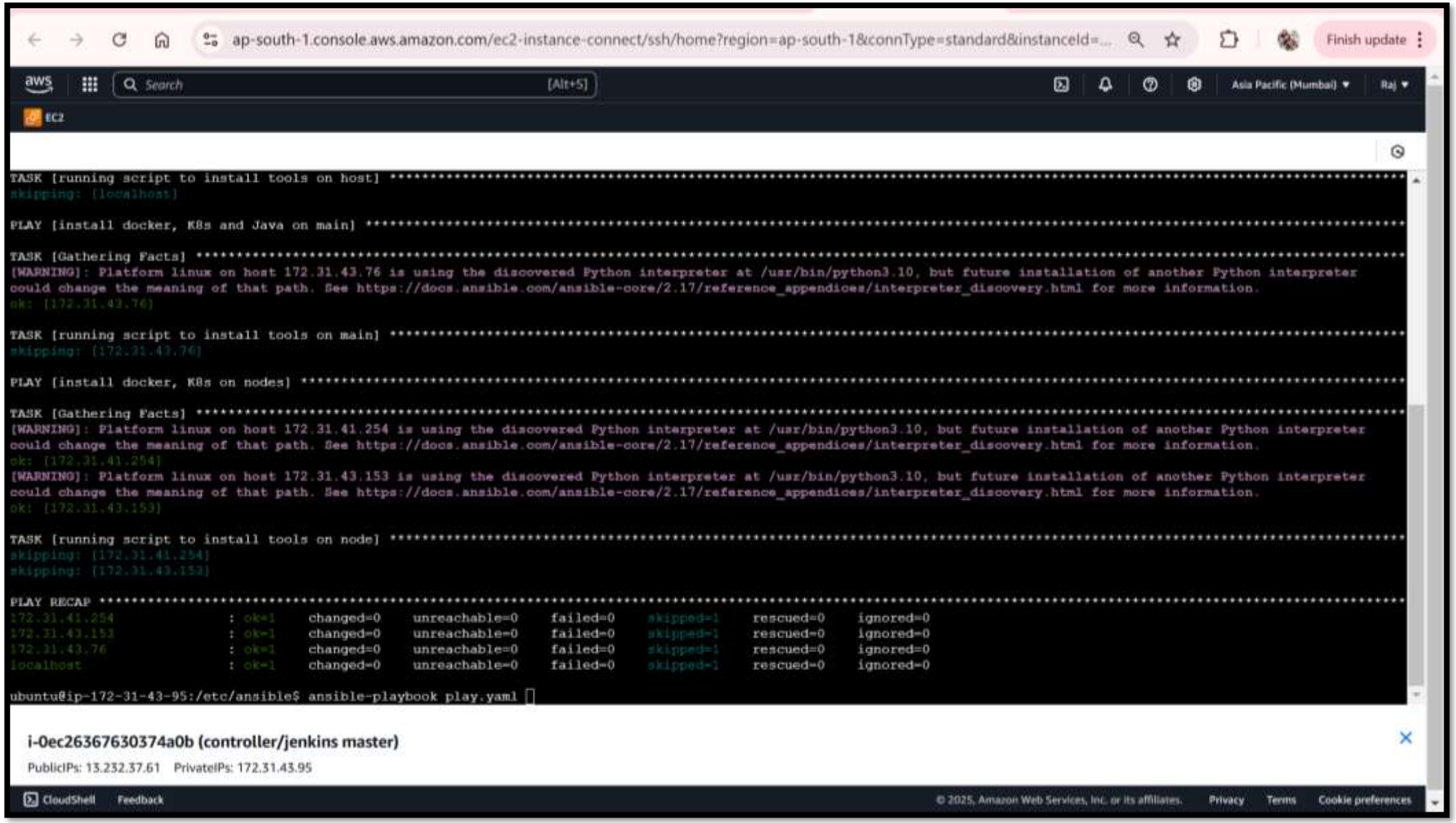


All machine are up and running

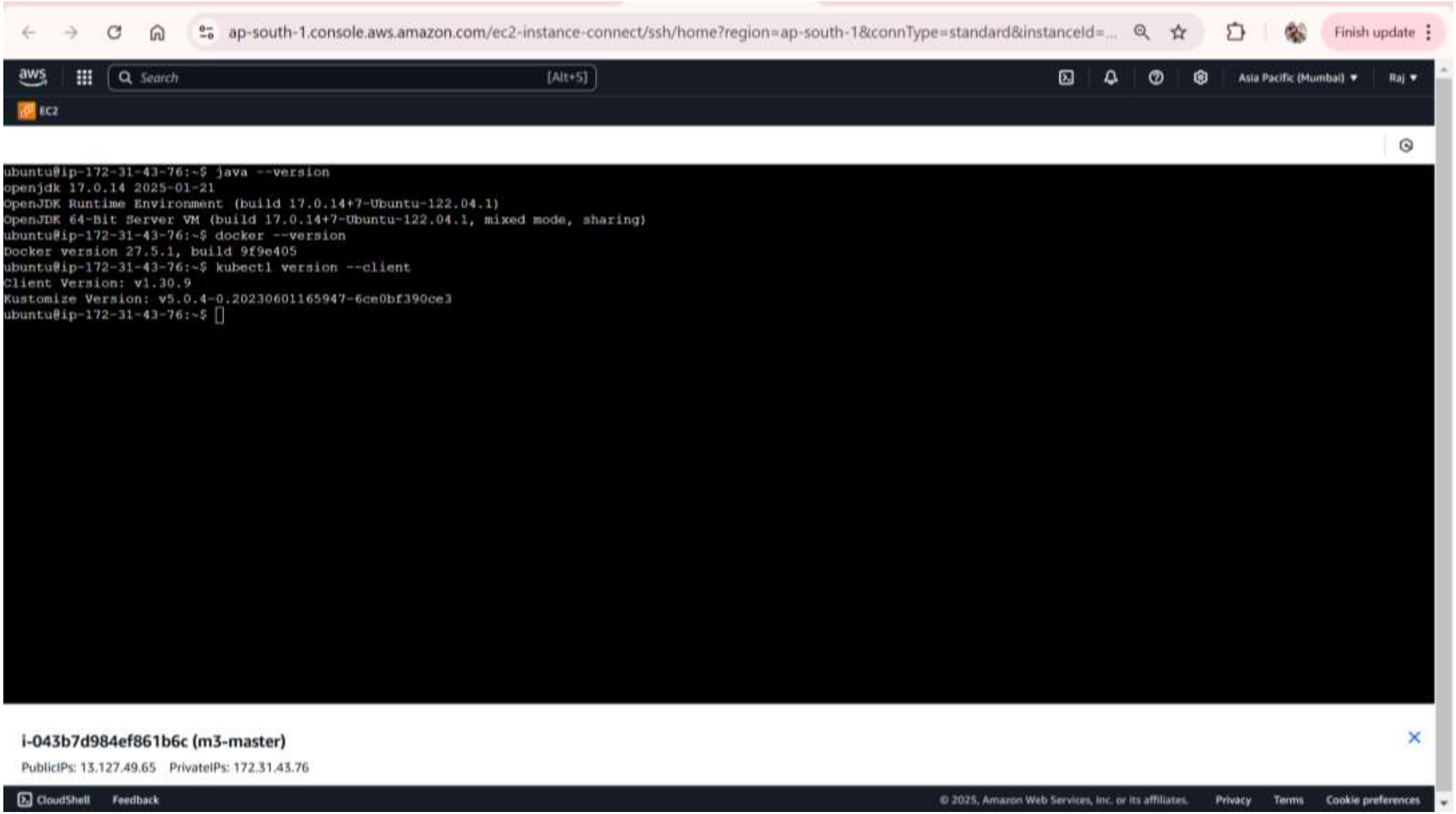


Ansible playbook play.yaml



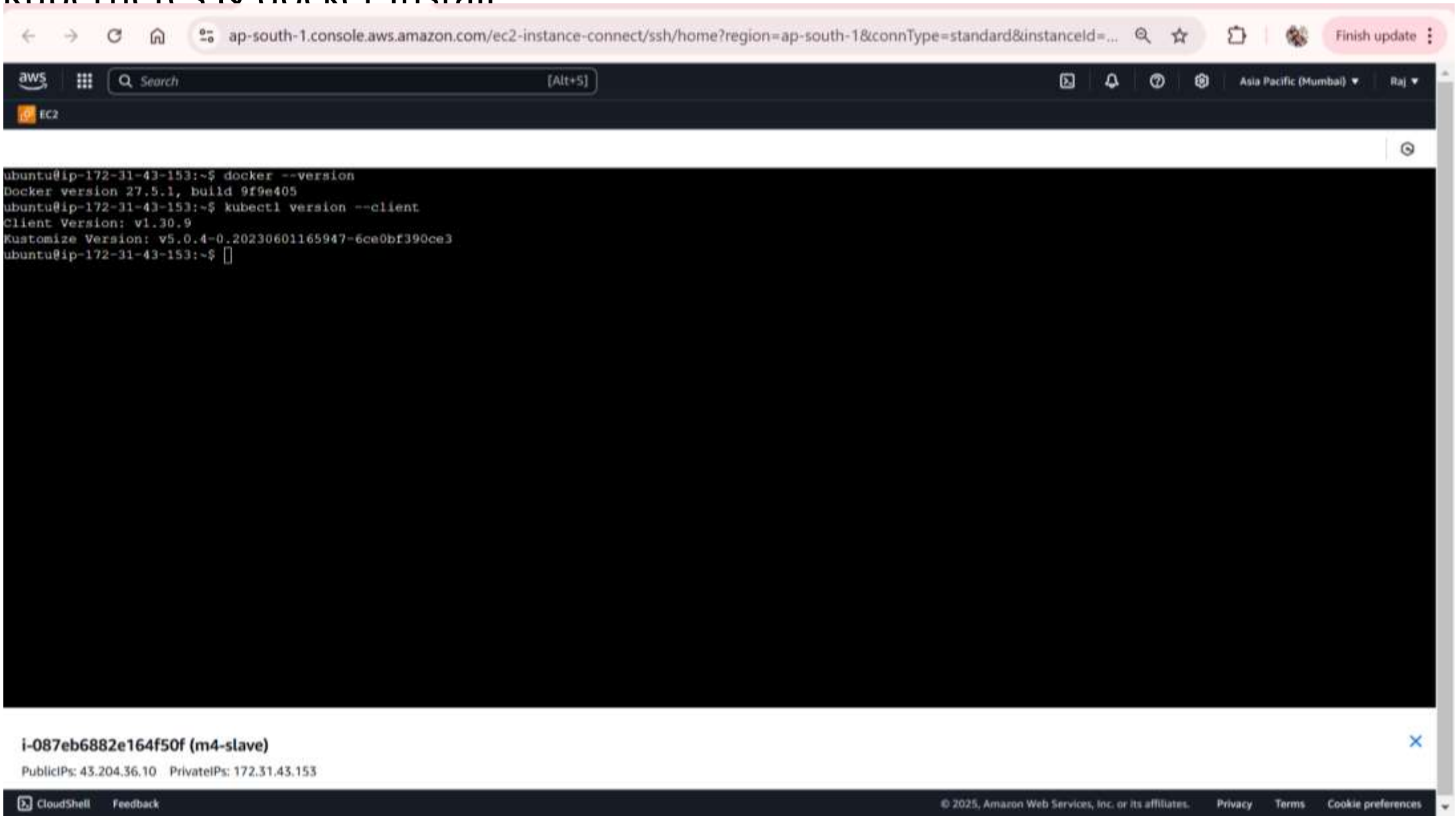

ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=...
Finish update
aws
Search
[Alt+5]
Asia Pacific (Mumbai)
Raj
EC2
TASK [running script to install tools on host] ********
skipping: [localhost]
PLAY [install docker, K8s and Java on main] ********
TASK [Gathering Facts] ********
[WARNING]: Platform linux on host 172.31.43.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.43.76]
TASK [running script to install tools on main] ********
skipping: [172.31.43.76]
PLAY [install docker, K8s on nodes] ********
TASK [Gathering Facts] ********
[WARNING]: Platform linux on host 172.31.41.254 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.41.254]
[WARNING]: Platform linux on host 172.31.43.153 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.43.153]
TASK [running script to install tools on node] ********
skipping: [172.31.41.254]
skipping: [172.31.43.153]
PLAY RECAP ********
172.31.41.254 : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
172.31.43.153 : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
172.31.43.76 : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
localhost : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
ubuntu@ip-172-31-43-95:/etc/ansible$ ansible-playbook play.yaml
i-0ec26367630374a0b (controller/jenkins master)
PublicIPs: 13.232.37.61 PrivateIPs: 172.31.43.95
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

m3-k8s-master Kubernetes , docker & Java Installed



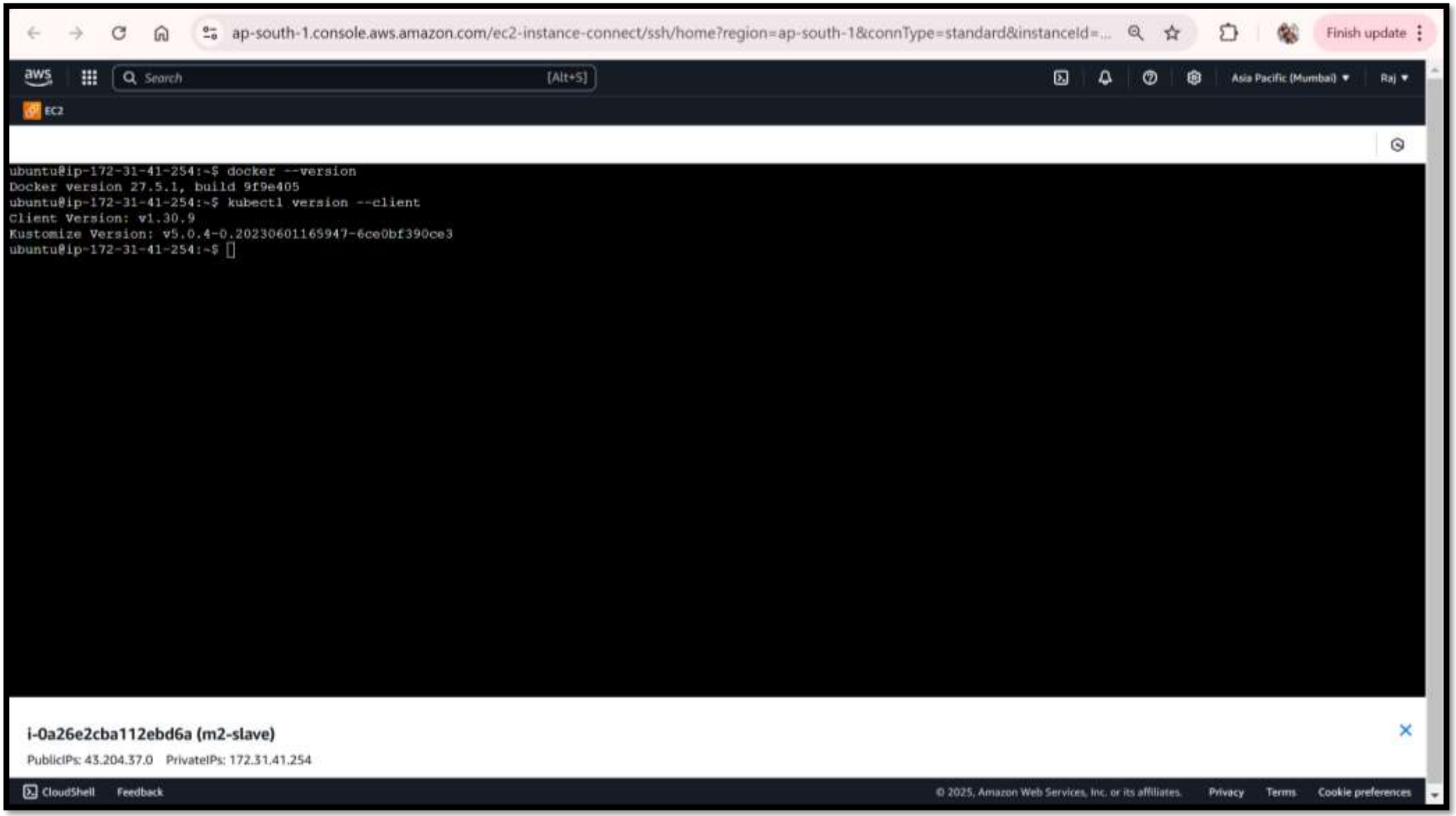
m4-k8s-slave

Kubernetes & docker Install

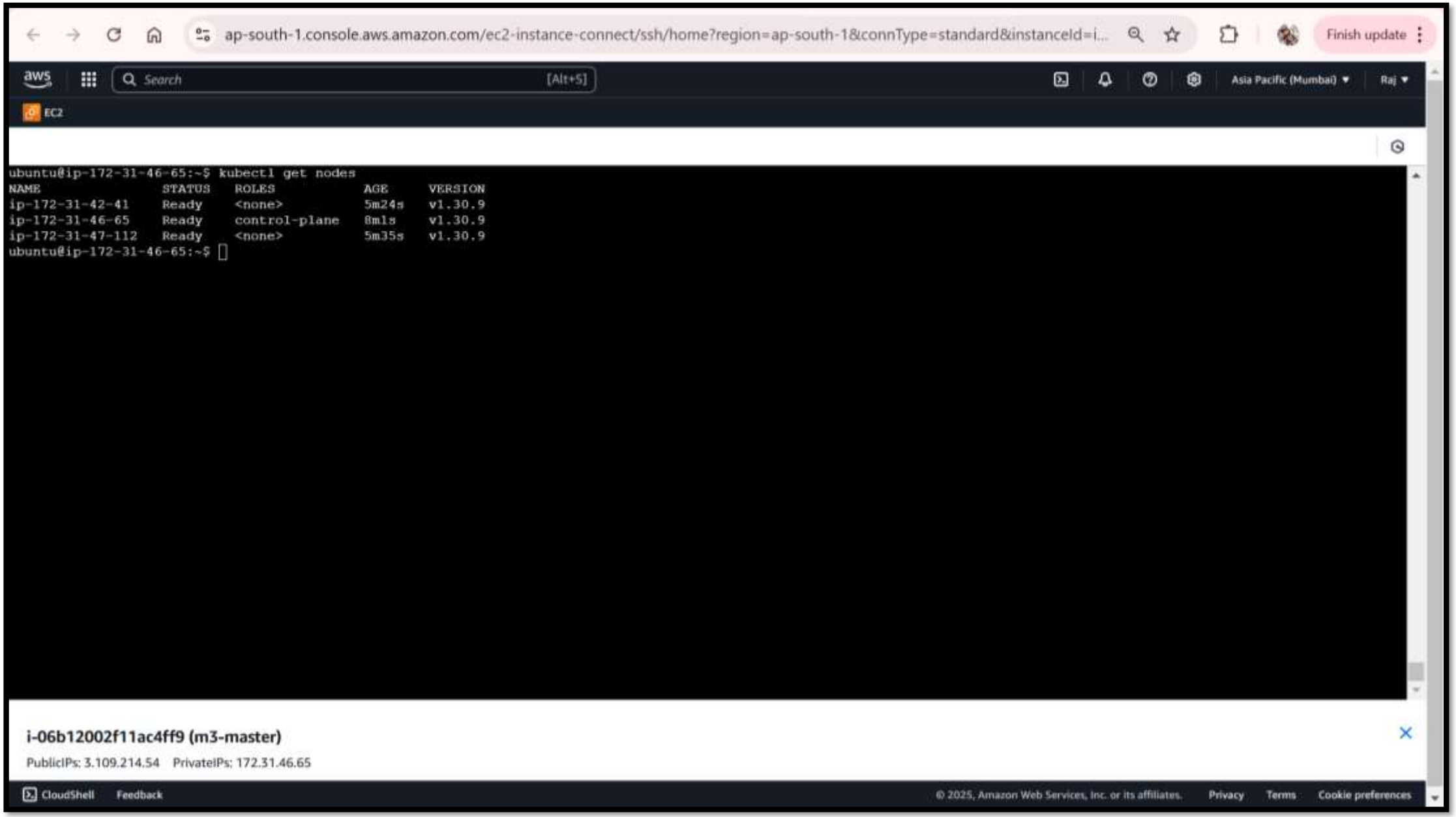


m2-k8s-slave

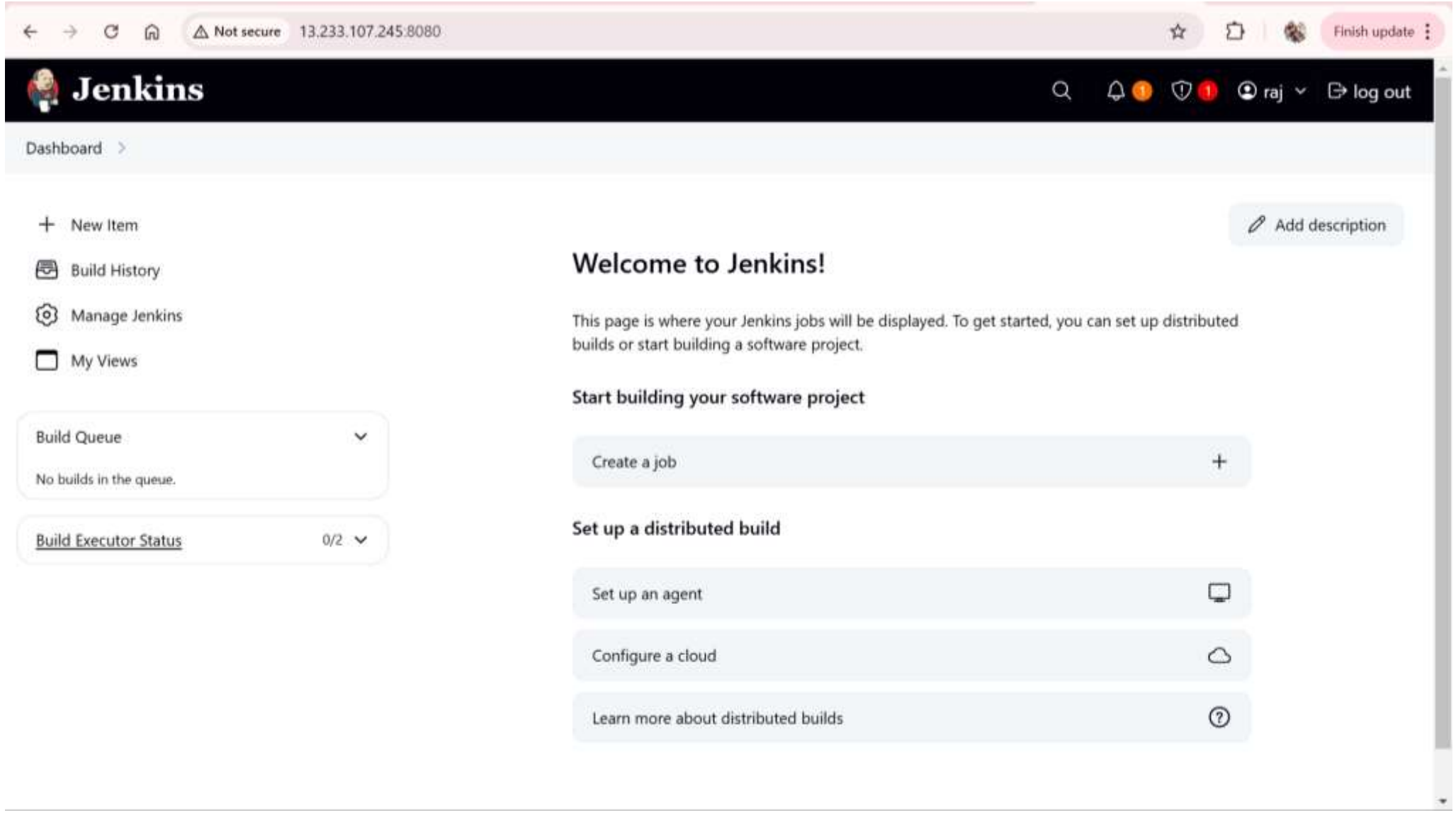
Kubernetes & docker Install



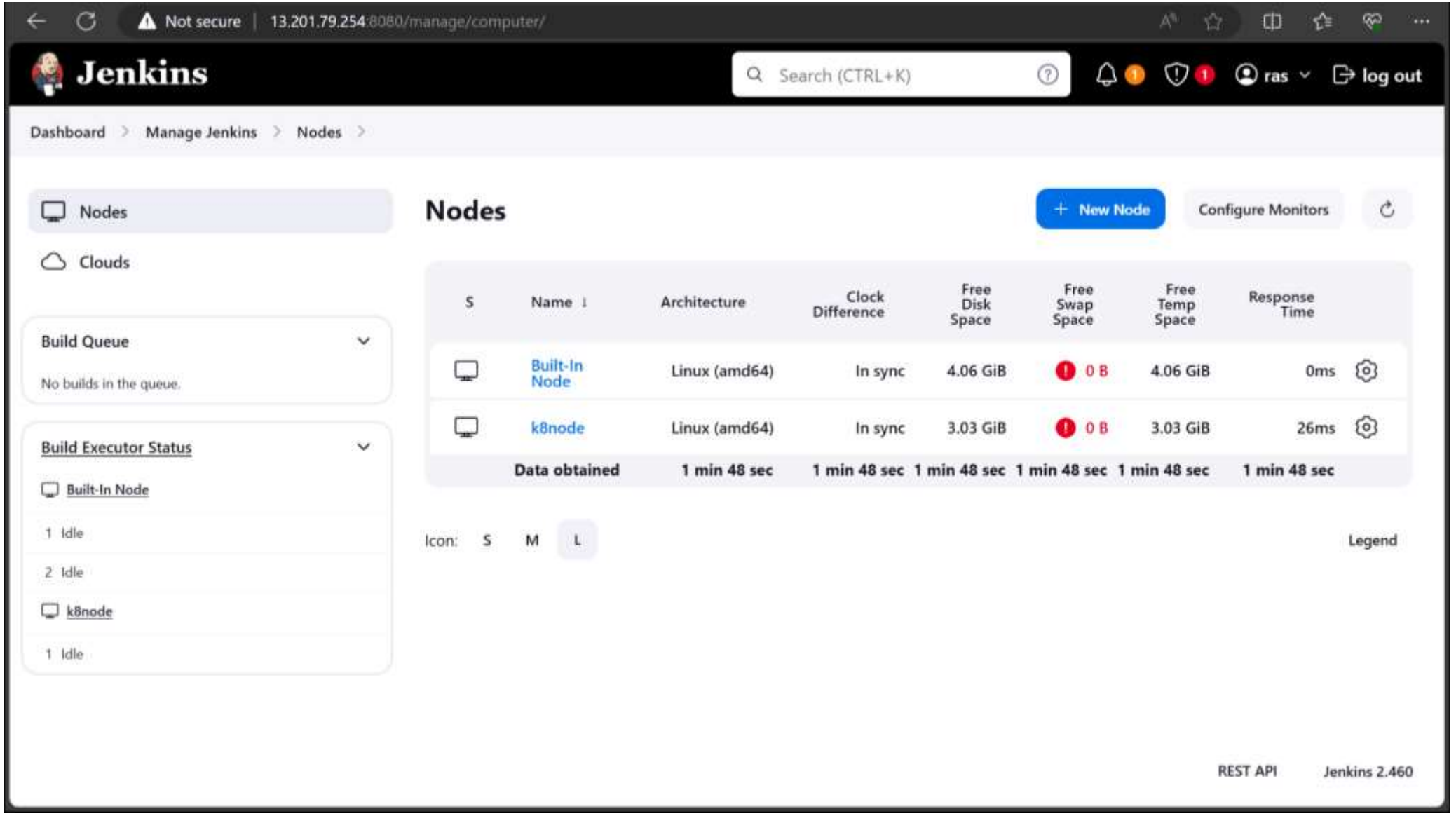
Kubernetes master nodes



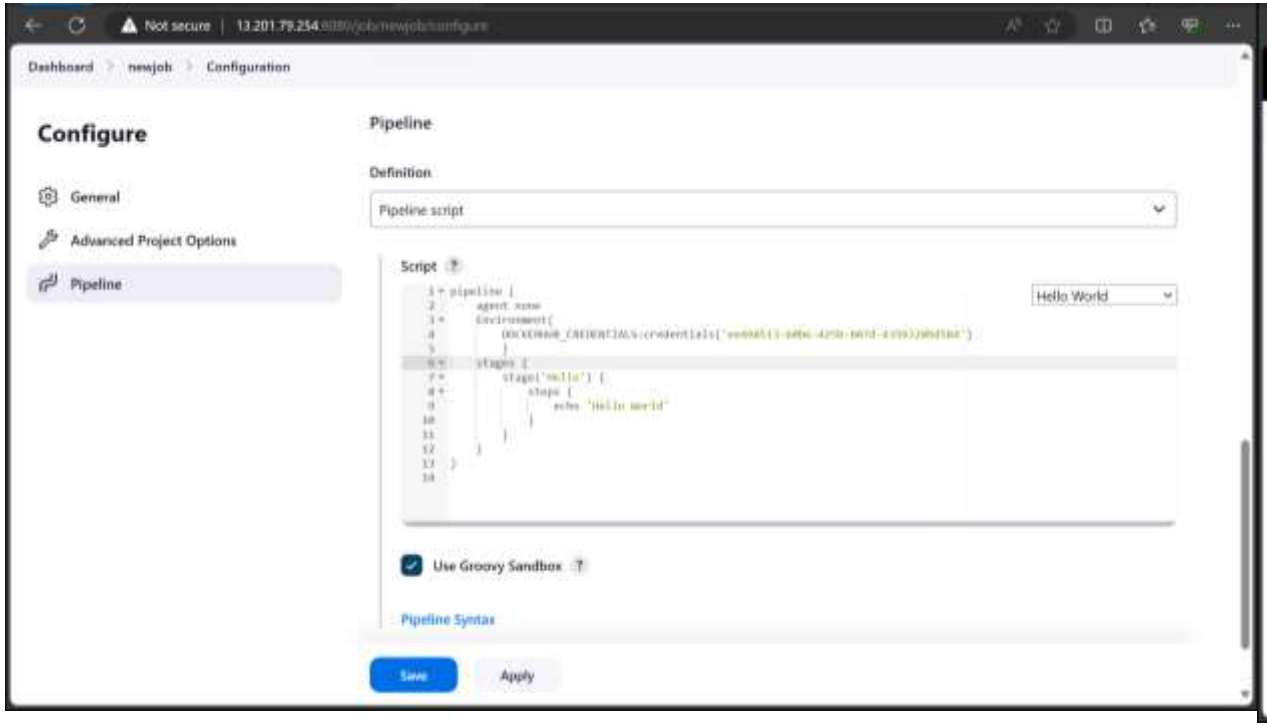
Connection to Jenkins port using public Ips of master-Jenkins machine

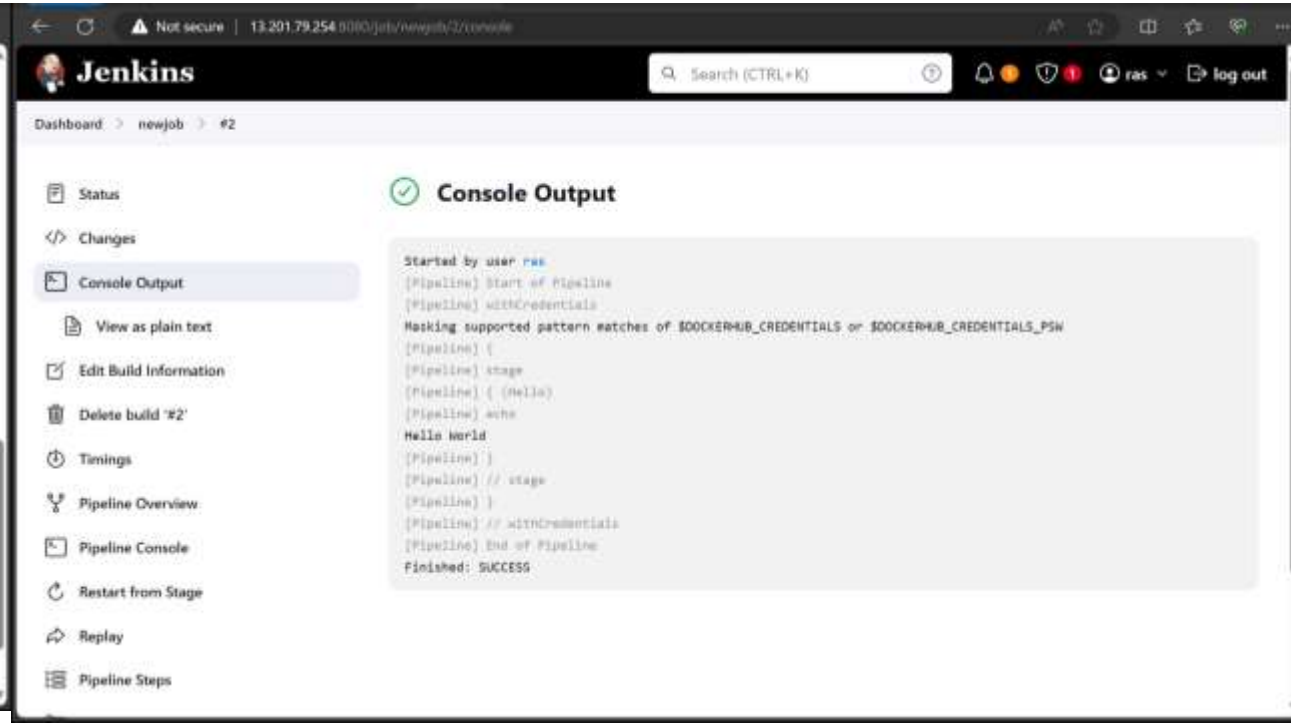


Created a node for **m3-k8s-master** named as “k8node”

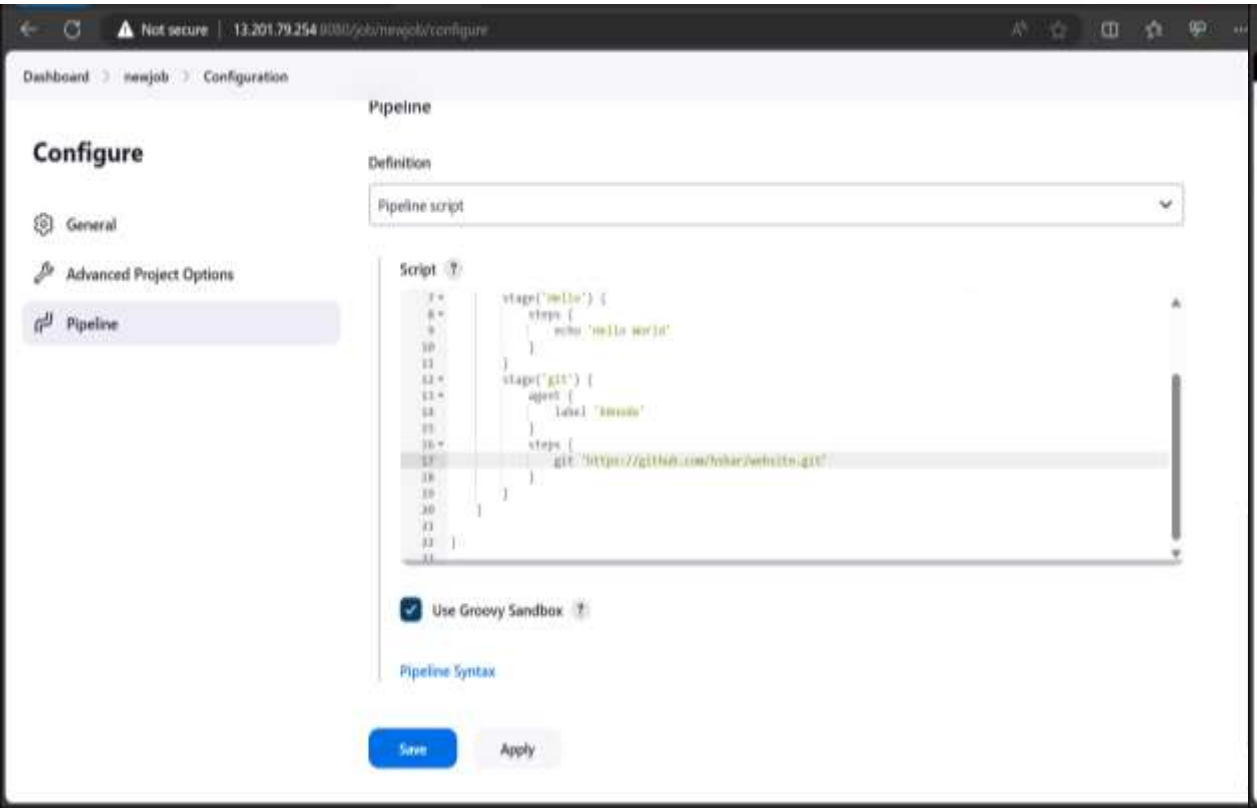


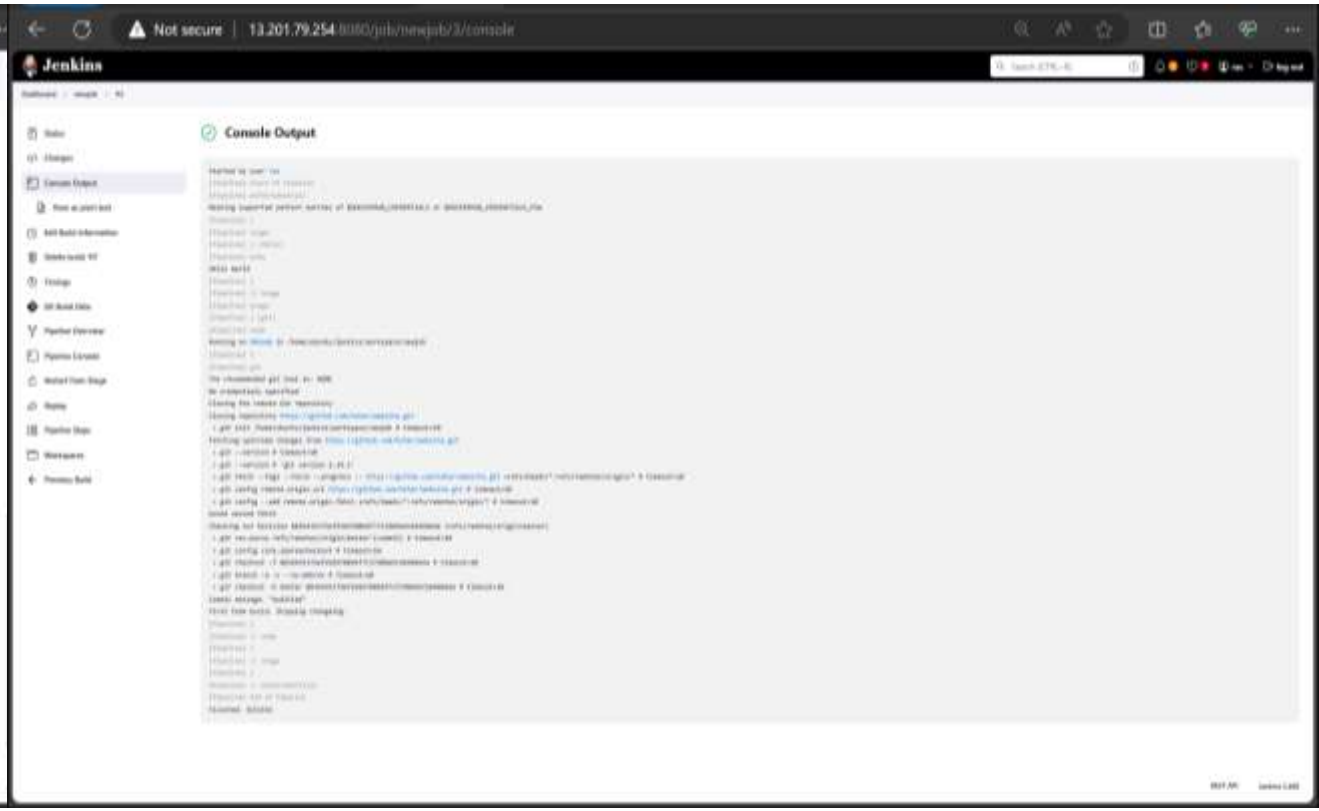
Created a newjob



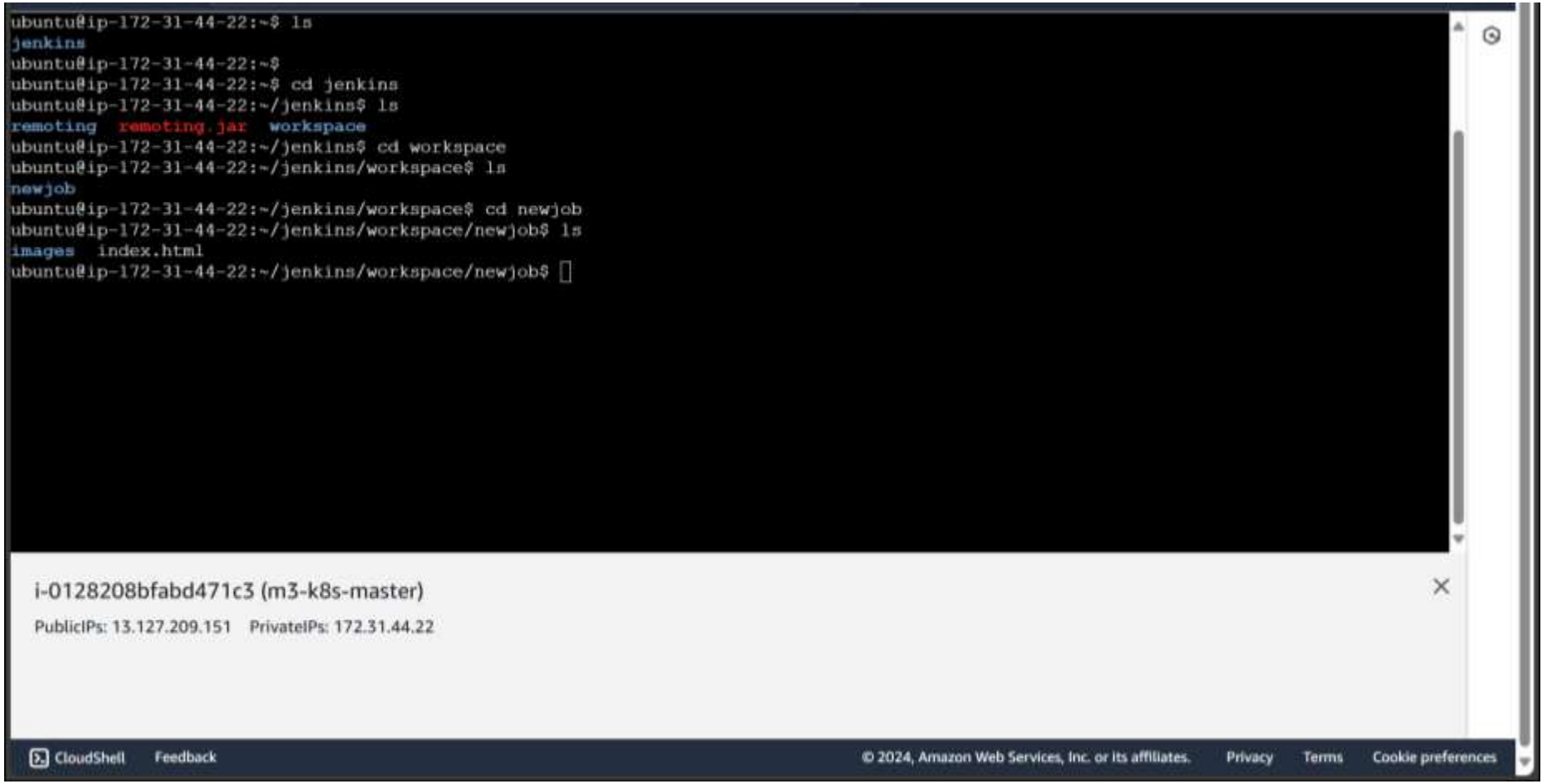


Adding git stage

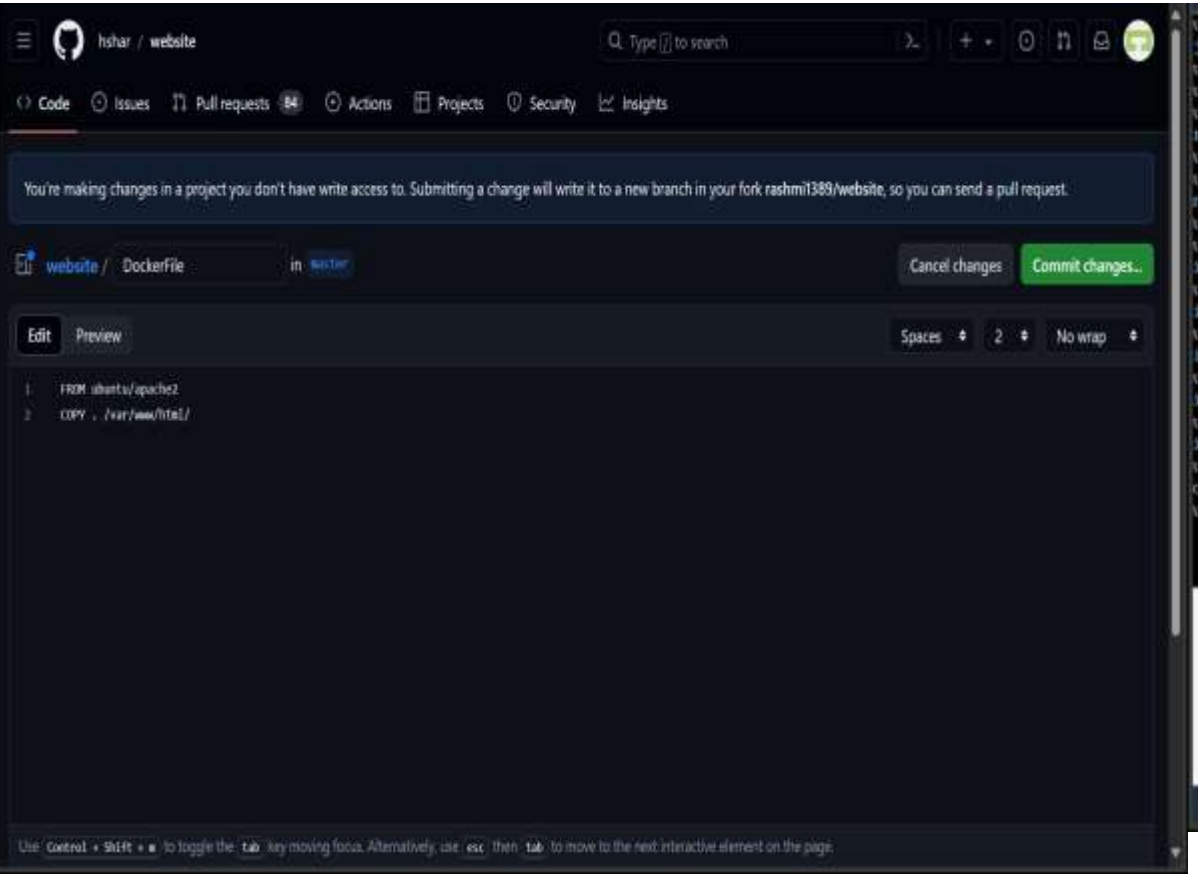


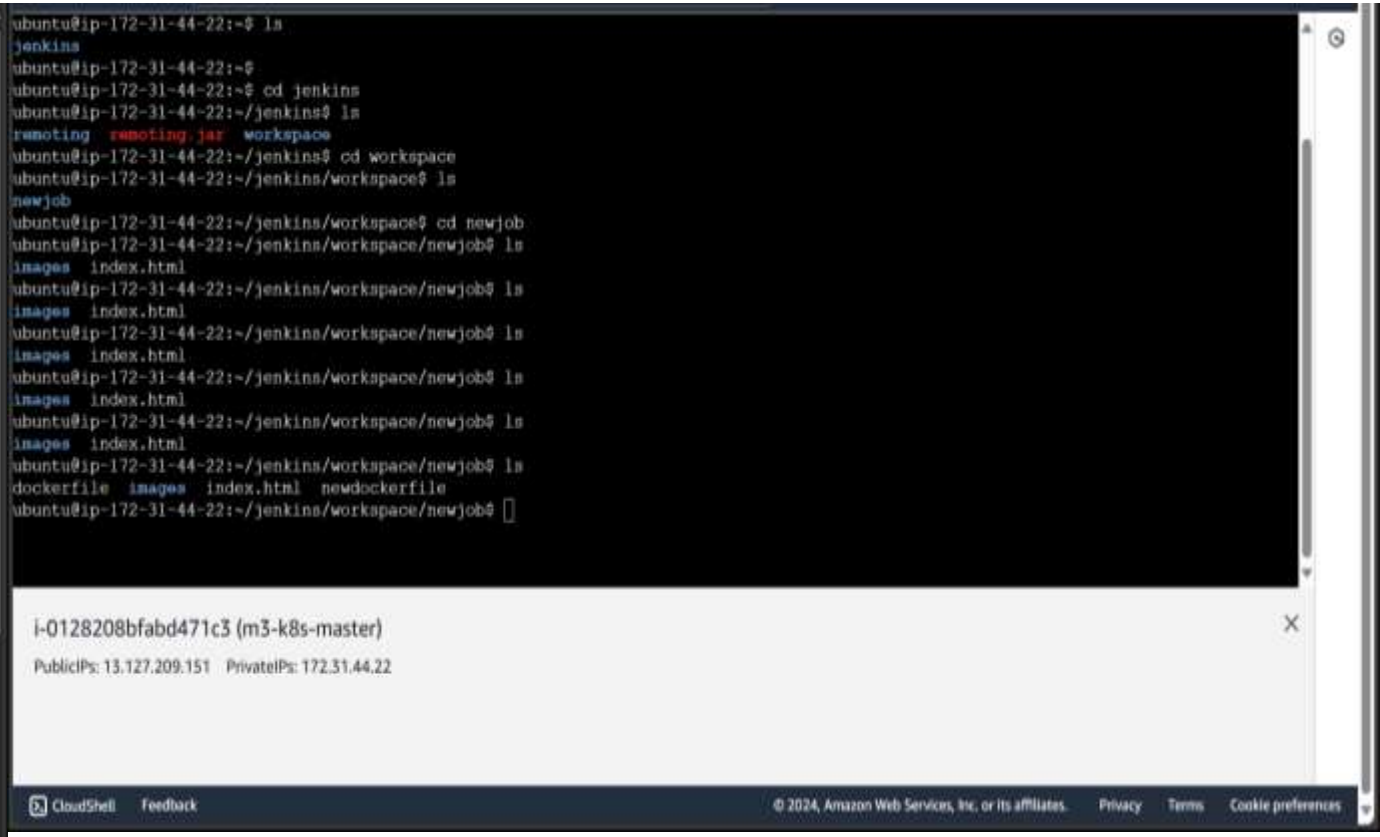


Reflecting all the file from Jenkins and git in m3-k8s-master machine

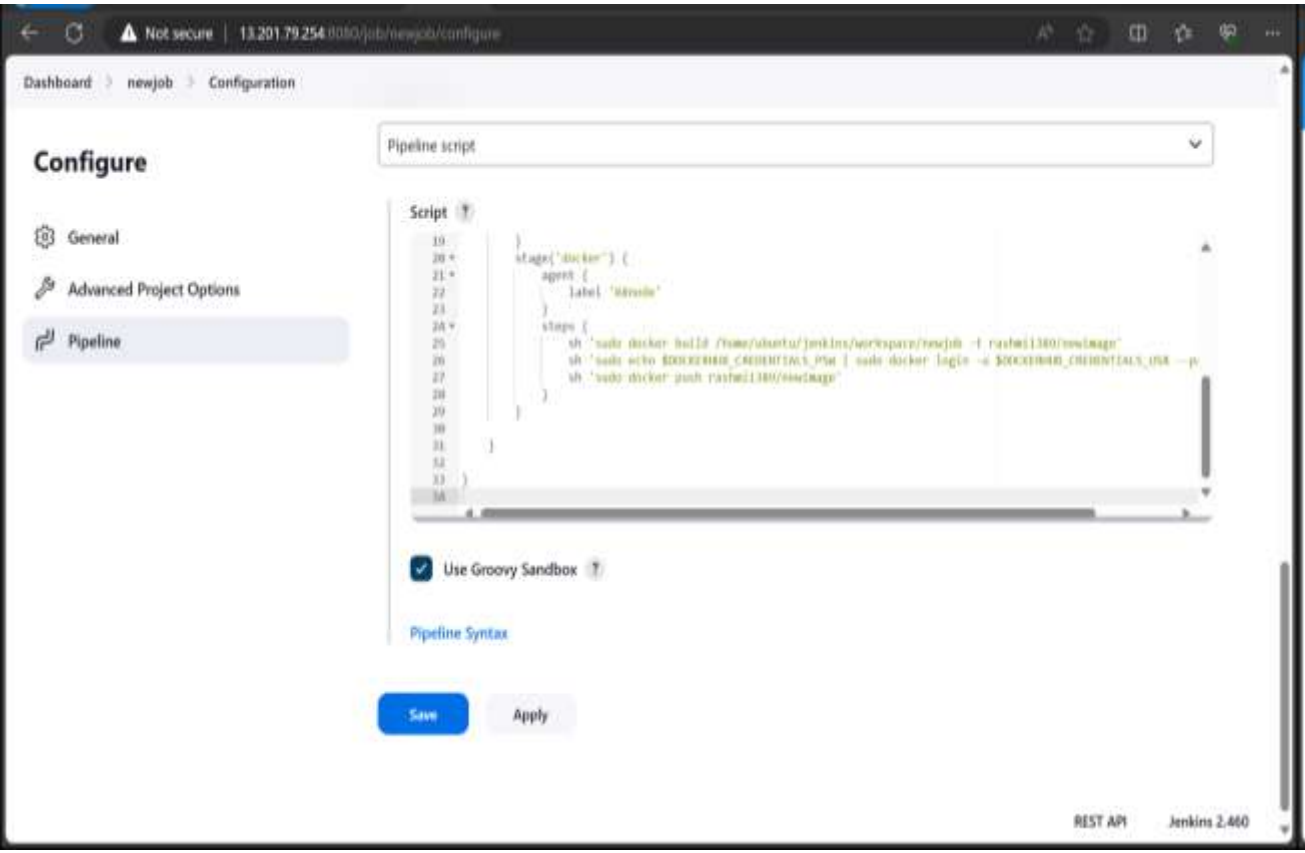


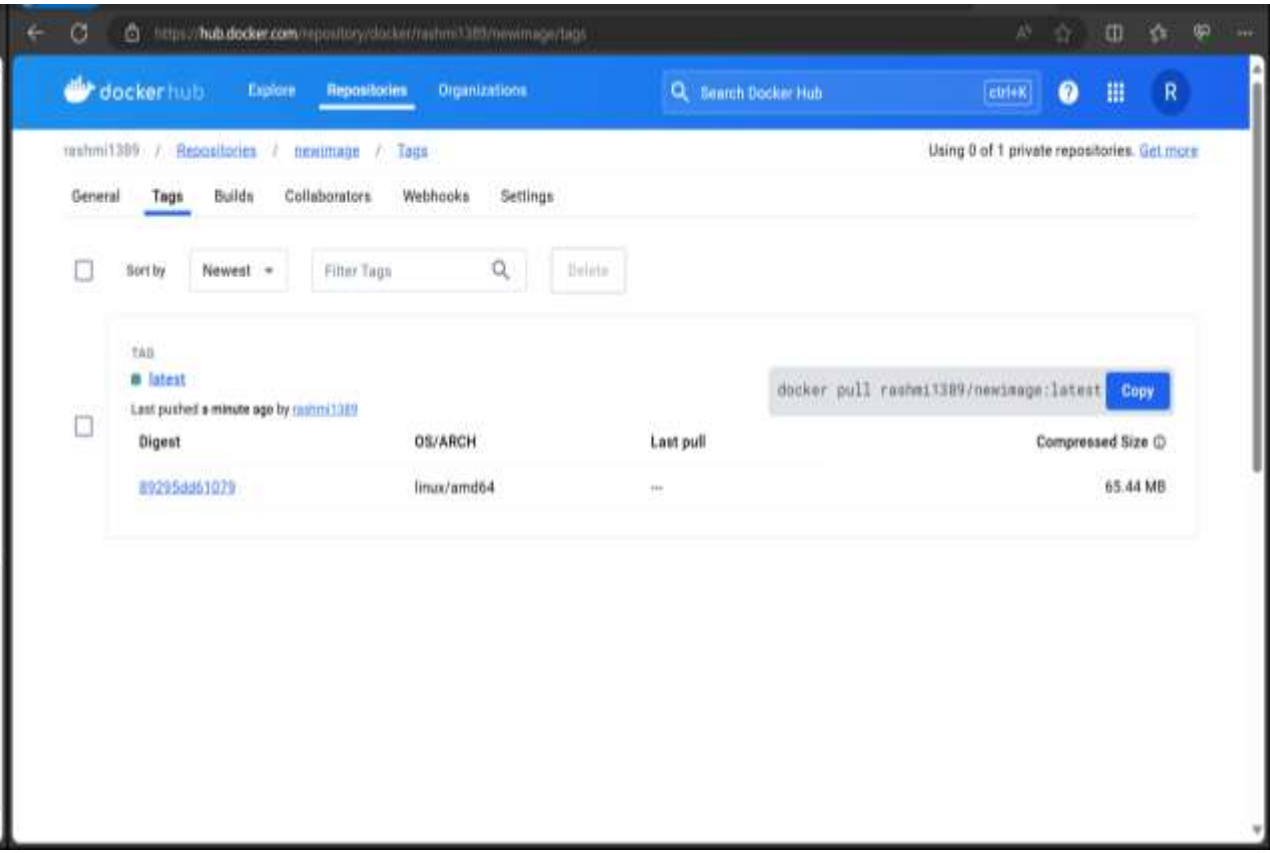
Create a Dockerfile in git and commit changes



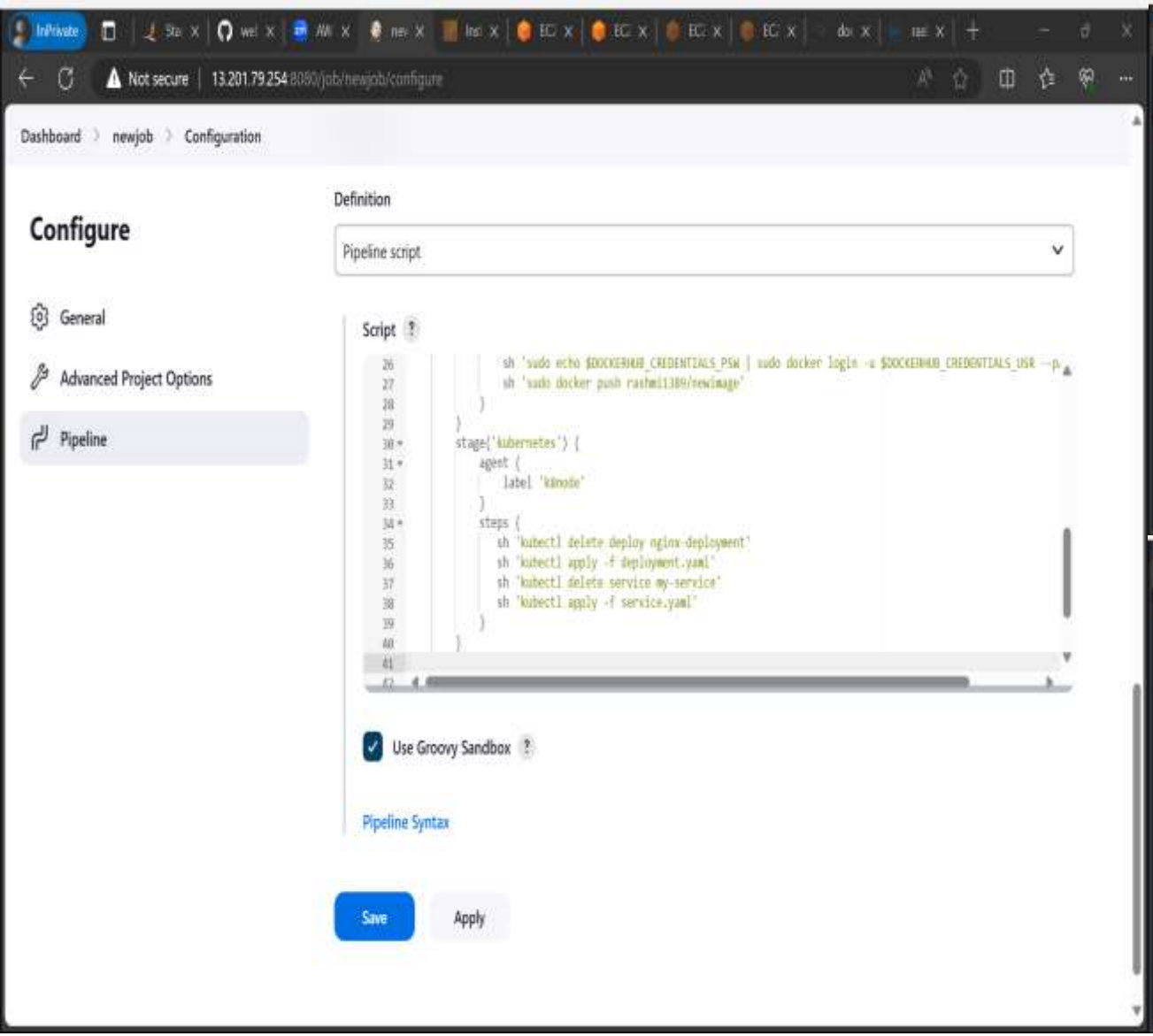


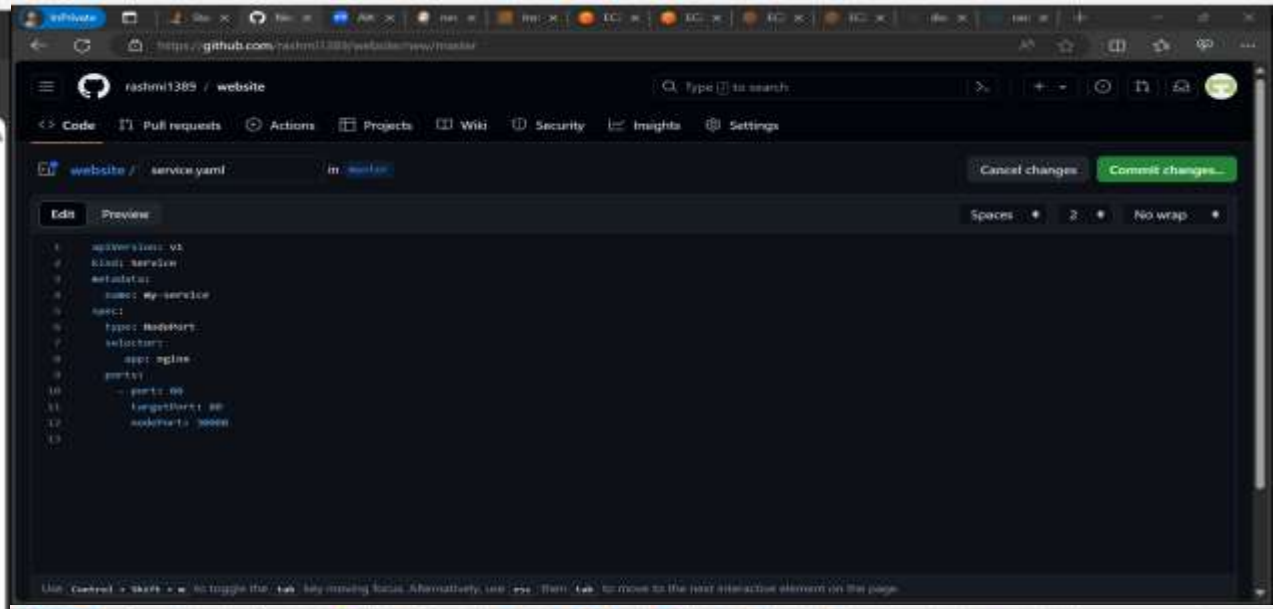
Configuration for building a docker image and pushing to docker hub

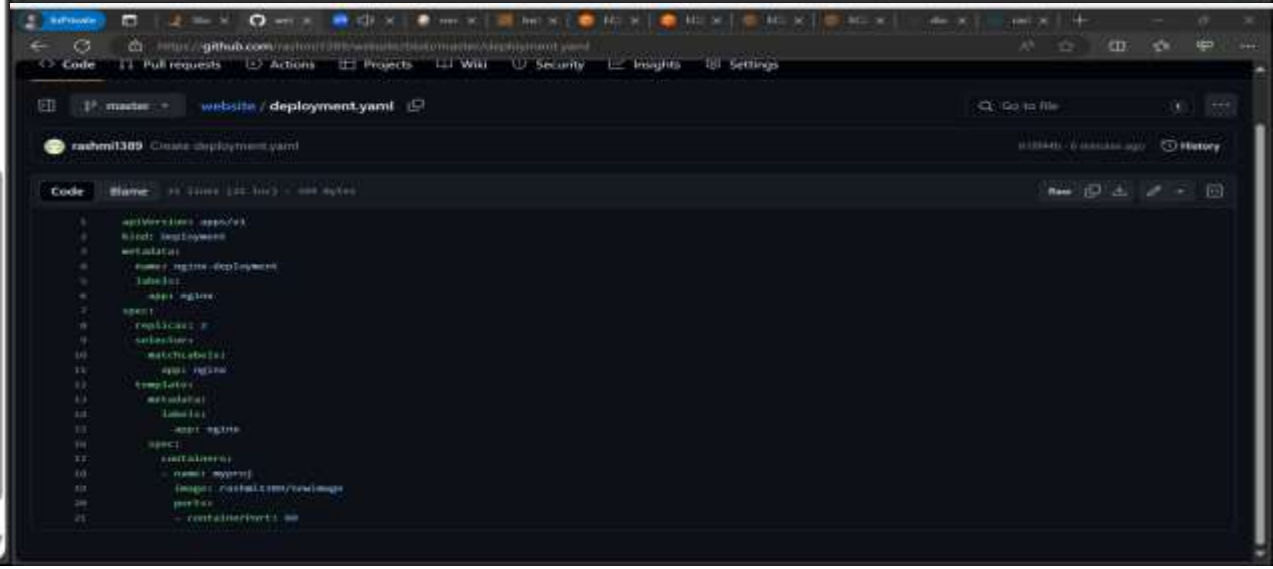




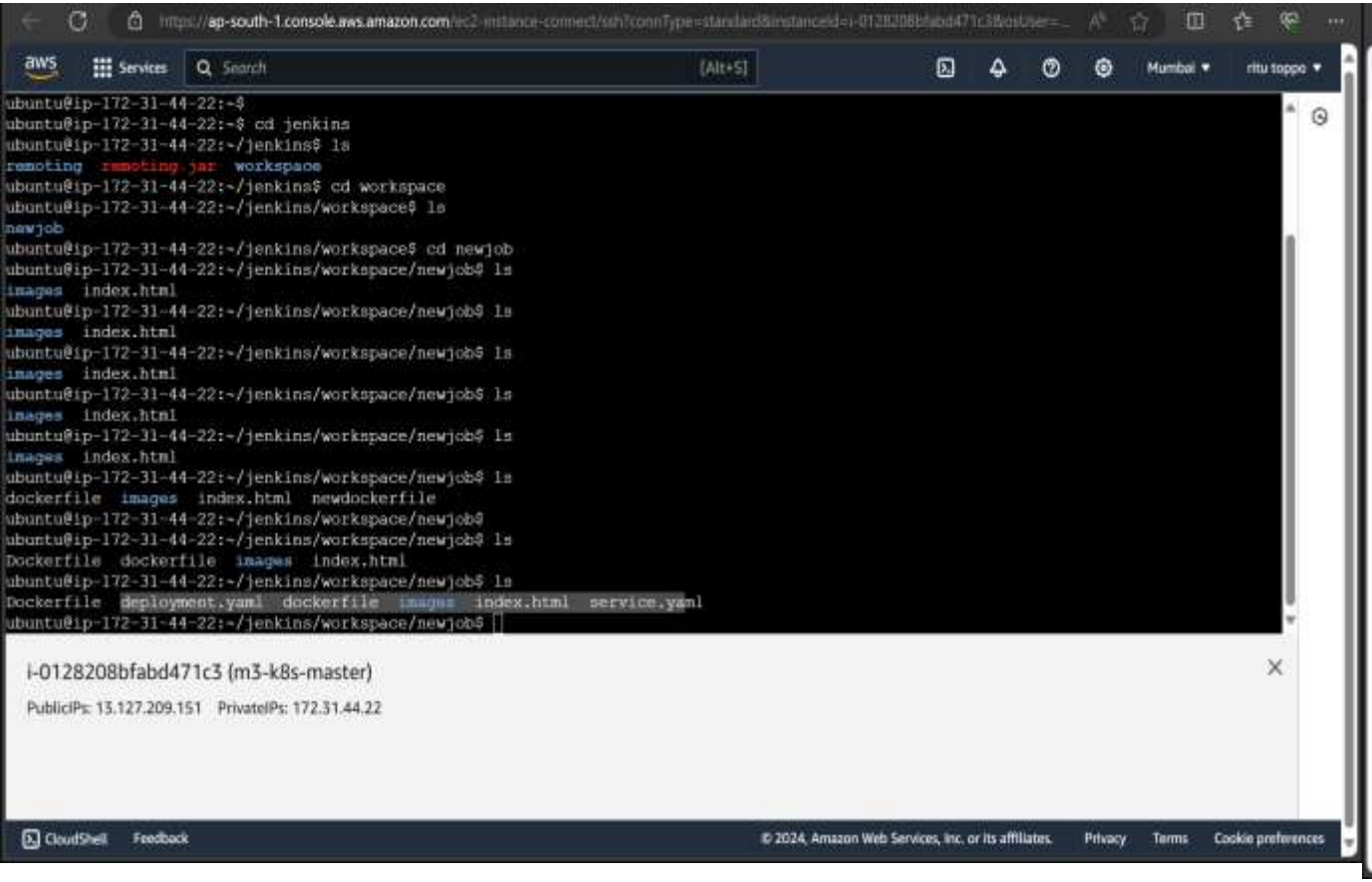
Configuring Kubernetes in jenkins

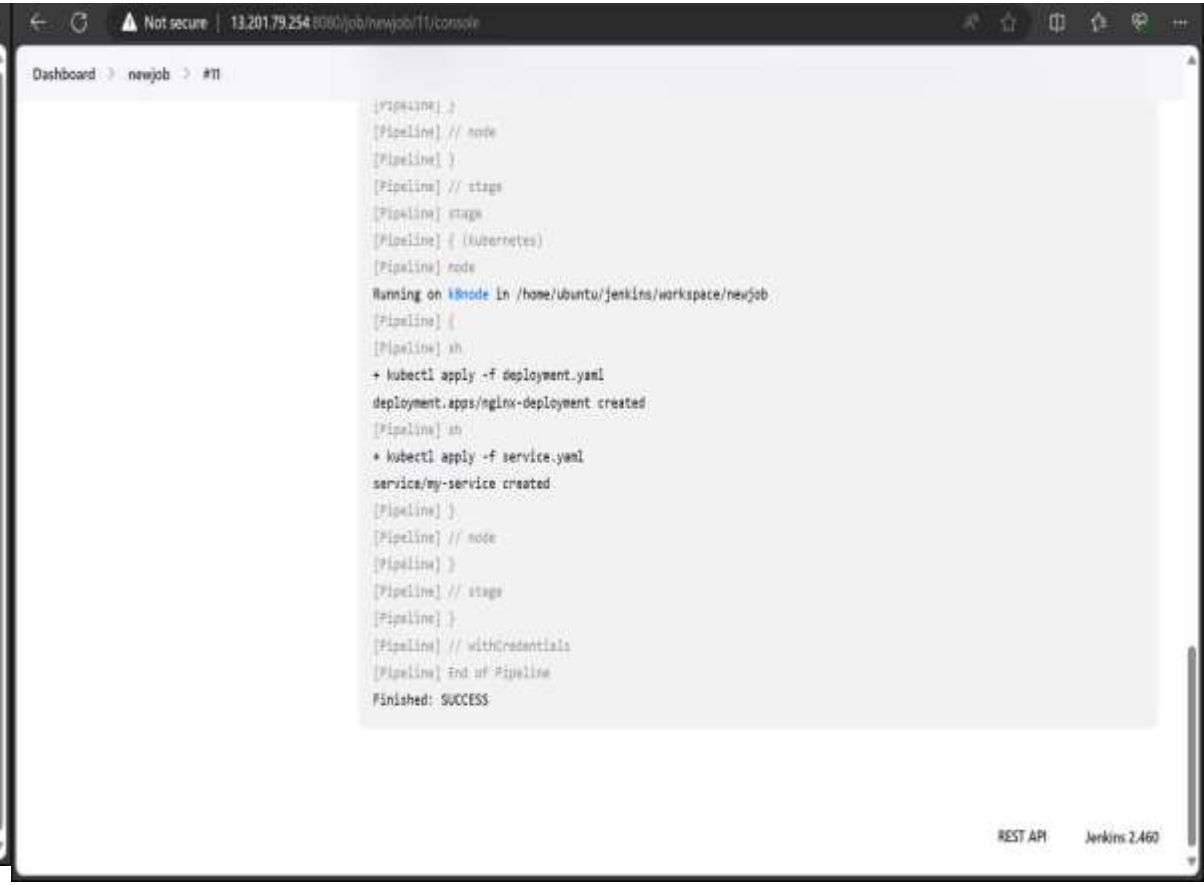






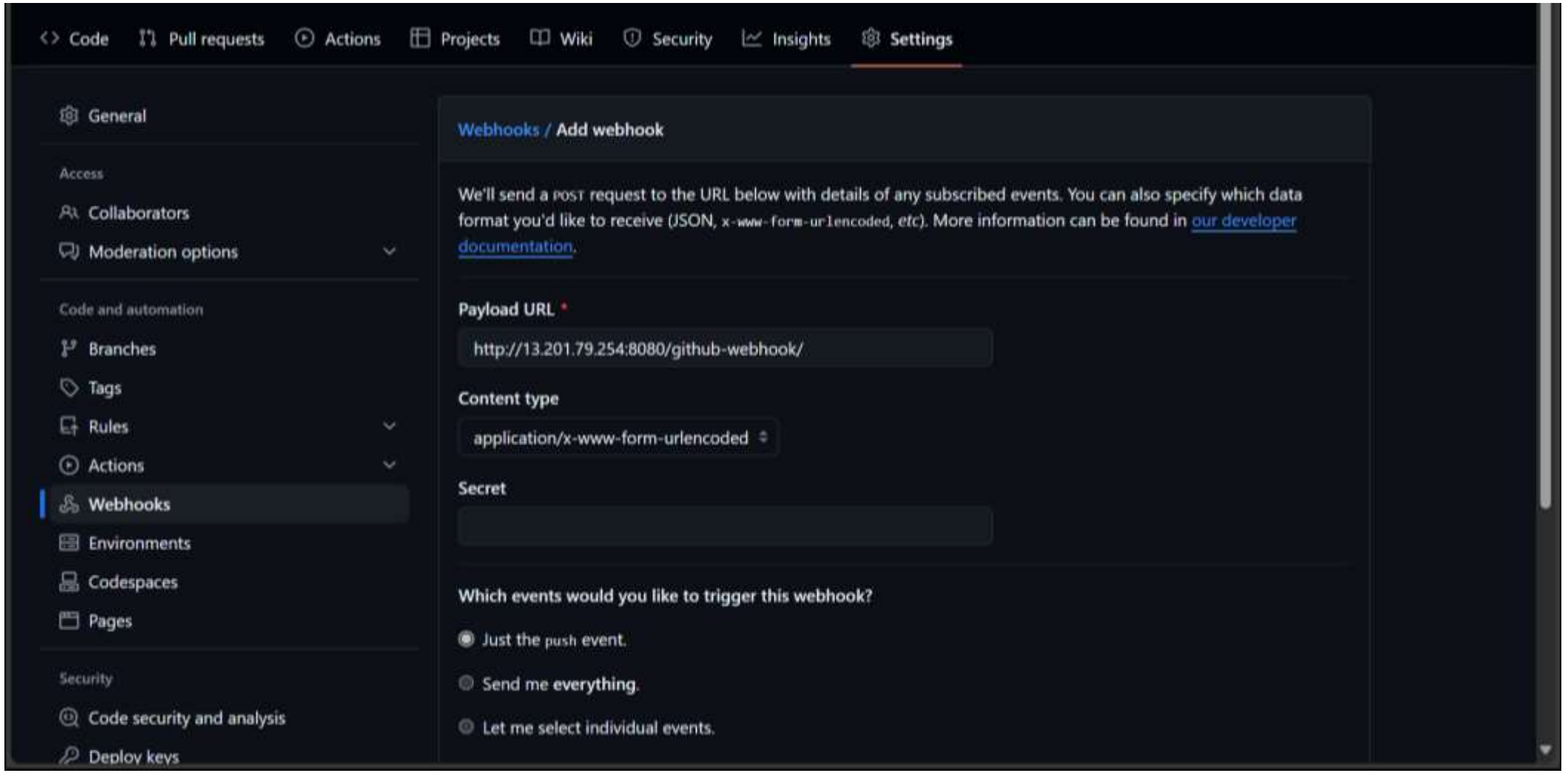
Reflecting deployment.yaml and service.yaml file in machine m3-k8s-master



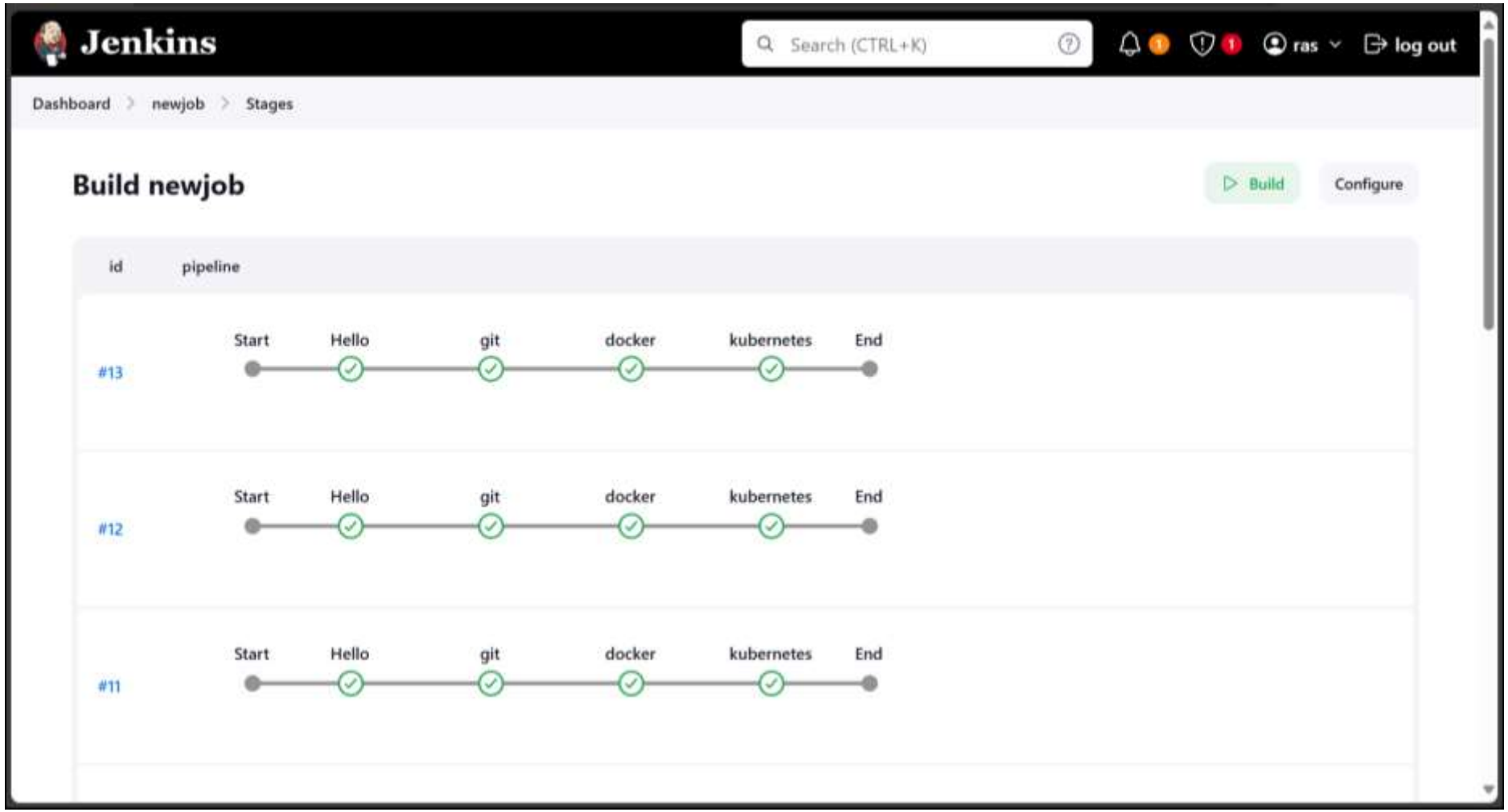


Able to access the web page using slave machine in port 30008

Connecting Jenkins to github webhook to automate workflow

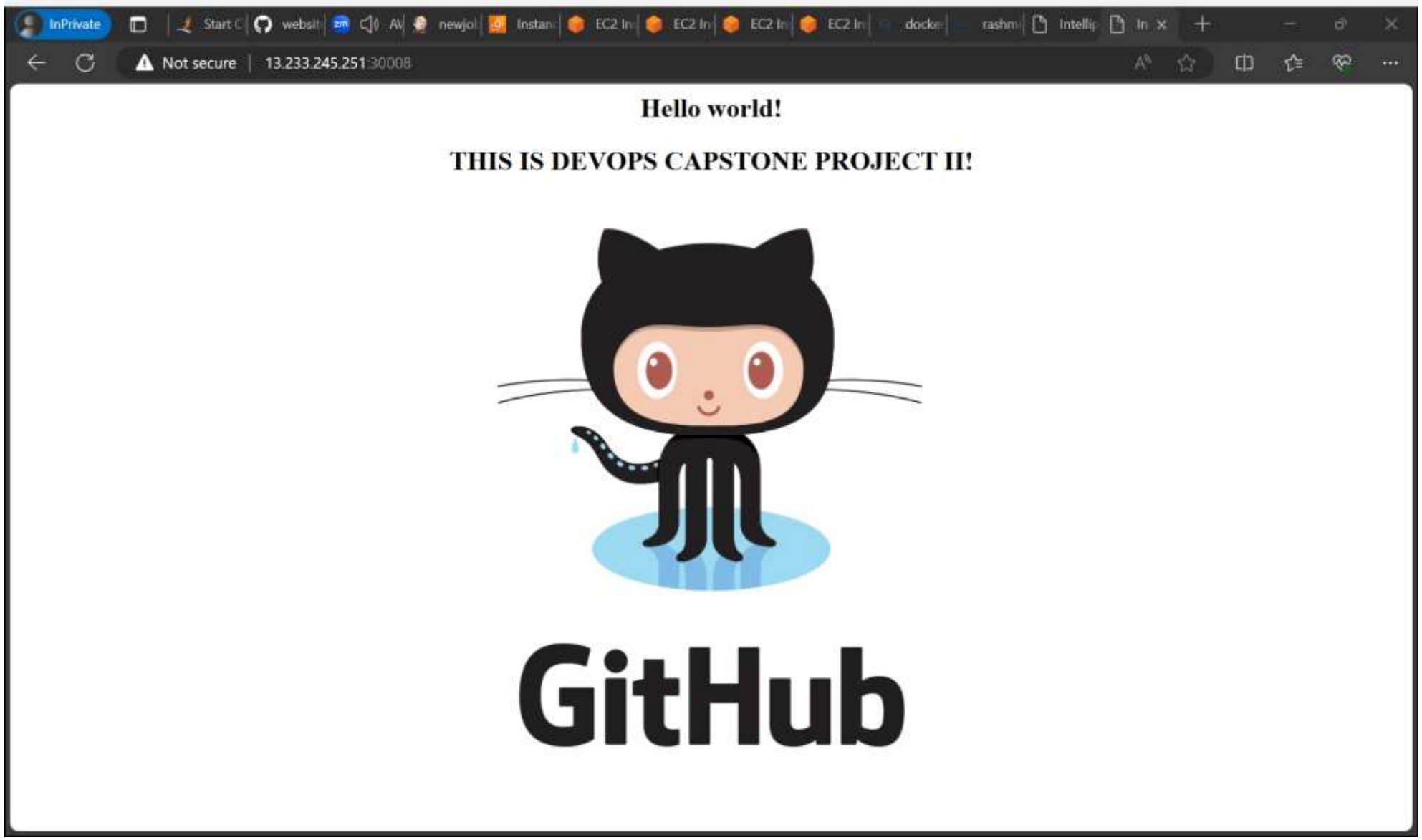


Build stage


Jenkins
Search (CTRL+K)
ras
log out
Dashboard
newjob
Stages
Build newjob
Build
Configure
id
pipeline
#13
Start
Hello
git
docker
kubernetes
End
#12
Start
Hello
git
docker
kubernetes
End
#11
Start
Hello
git
docker
kubernetes
End

Jenkins Auto build job is running Successfully in **m2-k8s-slave)**

Jenkins Auto build job is running Successfully in **m4-k8s-slave**



Pipeline Console