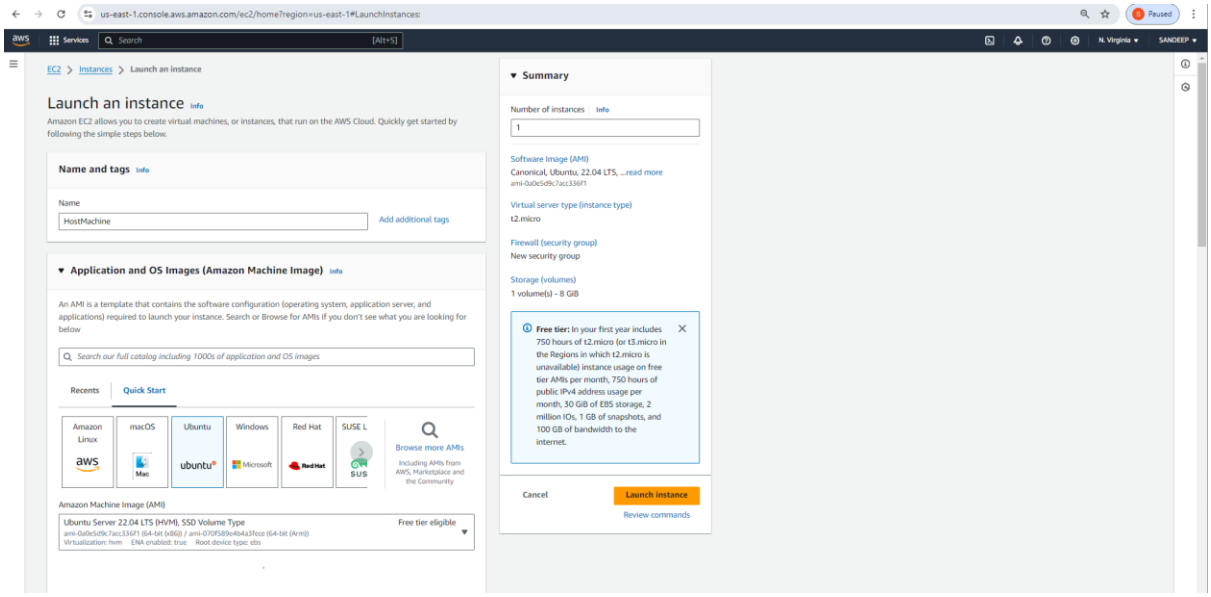
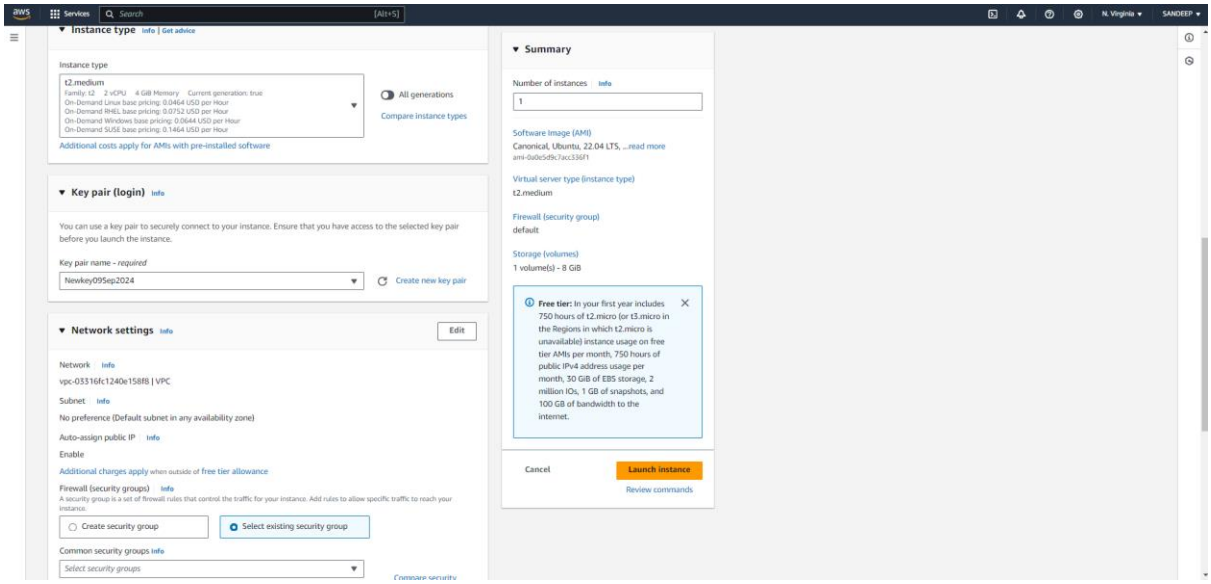
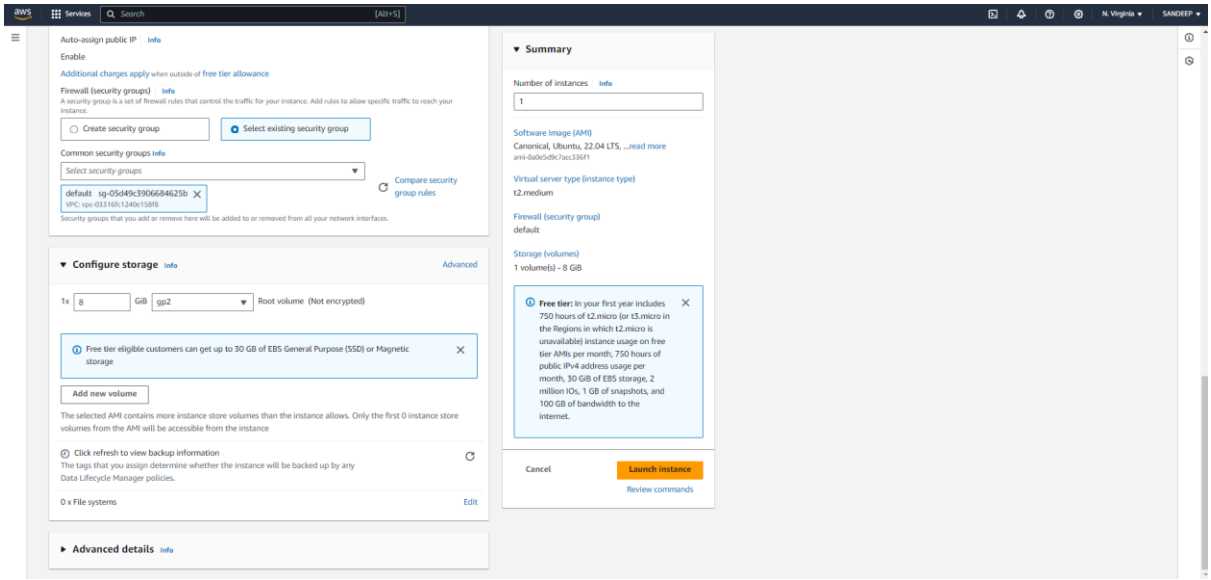
Devops Capstone Project2

Create a host Machine







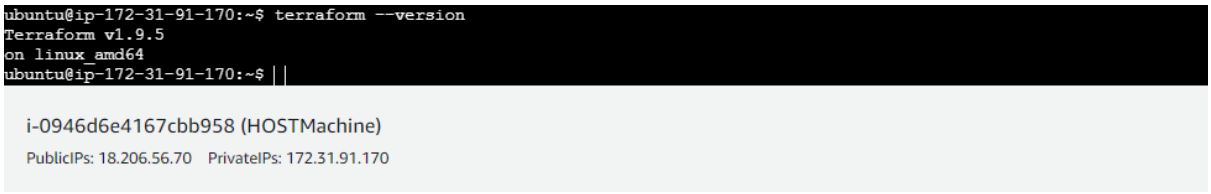
Sudo apt-get update

Install Terraform

wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg

echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list

sudo apt update && sudo apt install terraform



Sudo nano main.tf

```
provider "aws" {
  region = "us-east-1"
  access_key = "AKIA3FLD5QD2AMOVAPXG"
  secret_key = "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"
}
resource "aws_instance" "K8s-Slave1" {
  ami = "ami-0a0e5d9c7acc336f1"
```

```
  instance_type = "t2.medium"
  key_name = "Newkey09Sep2024"
  tags = {
    Name = "m2-k8s-slave1"
  }
}

resource "aws_instance" "K8s-master" {
  ami = "ami-0a0e5d9c7acc336f1"
  instance_type = "t2.medium"
  key_name = "Newkey09Sep2024"
  tags = {
    Name = "m3-k8s-master"
  }
}

resource "aws_instance" "K8s-Slave2" {
  ami= "ami-0a0e5d9c7acc336f1"
  instance_type = "t2.medium"
  key_name = "Newkey09Sep2024"
  tags = {
    Name = "m4-k8s-slave2"
  }
}
```

save and exit

5- terraform init
6-terraform plan
7- terraform apply
8- sudo apt install software-properties-common (install ansible on ubuntu)
9- sudo add-apt-repository --yes --update ppa:ansible/ansible
10- sudo apt install ansible

Three Slave Machine has been created through terraform script.

Now Go to Host Machine

Now connect m2-k8s-slave1 & m3-k8s-&master & m4-k8s-slave2
run below command

1-sudo apt update

Now go to Sandeep-master

cd .ssh
ssh-keygen
enter
enter
enter
ls
authorized_keys id_rsa id_rsa.pub

2- sudo cat id_rsa.pub
it will display id code

3- Now in m2-k8s slave1
cd .ssh
sudo nano authorized_keys
Paste the id_rsa.pub code

Now in m3-k8s-master
cd .ssh
sudo nano authorized_keys
Paste the id id_rsa.pub code

Now in m4-k8s-slave2
cd .ssh
sudo nano authorized_keys
Paste the id_rsa.pub code

CTRL+s CTRL+ X

now in sandeep-master
1-cd
2-cd /etc/ansible
3-ls
4- sudo nano hosts

```
ubuntu@ip-172-31-91-170:~/.ssh$ cd /etc/ansible
ubuntu@ip-172-31-91-170:/etc/ansible$ ls
ansible.cfg  hosts  roles
ubuntu@ip-172-31-91-170:/etc/ansible$
```

i-0946d6e4167cbb958 (HOSTMachine)

PublicIPs: 18.206.56.70 PrivateIPs: 172.31.91.170

we will make two group one mater and other slaves
paste all three private ip of m2 m3 & m4

```
aws   Services   Search
  GNU nano 6.2
[master]
172.31.91.136
[slaves]
172.31.81.83
172.31.91.84
```

```
ubuntu@ip-172-31-91-170:/etc/ansible$ sudo nano hosts
ubuntu@ip-172-31-91-170:/etc/ansible$ ansible -m ping all
The authenticity of host '172.31.81.83 (172.31.81.83)' can't be established.
ED25519 key fingerprint is SHA256:LJ/U8o1fuzTW1LsD/TzpjLqbUyV/Ljh/3DPfx2Fm9LI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? The authenticity of host '172.31.91.136 (172.31.91.136)' can't be established.
ED25519 key fingerprint is SHA256:Fh+mE5PoFeAGXeVfmdJ+0rxbeSZafSHTOIwF2J52bbk.
This key is not known by any other names
The authenticity of host '172.31.91.84 (172.31.91.84)' can't be established.
ED25519 key fingerprint is SHA256:LvpoYWuL2+NJkAQ+rn24029F/crX+hivXnVEMWyIv48.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
172.31.81.83 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
yes
172.31.91.136 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
yes
172.31.91.84 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-91-170:/etc/ansible$
```

i-0946d6e4167cbb958 (HOSTMachine)
PublicIPs: 18.206.56.70 PrivateIPs: 172.31.91.170

All three machines connected

ansible-playbook play.yaml --syntax -check
ansible-playbook play.yaml –-check
ansible-playbook play.yaml

```
ubuntu@ip-172-31-91-170:/etc/ansible$ sudo nano script1.sh
ubuntu@ip-172-31-91-170:/etc/ansible$ sudo nano script2.sh
ubuntu@ip-172-31-91-170:/etc/ansible$ sudo nano script3.sh
ubuntu@ip-172-31-91-170:/etc/ansible$ sudo nano play.yaml
ubuntu@ip-172-31-91-170:/etc/ansible$ ansible-playbook play.yaml --syntax -check

playbook: play.yaml
ubuntu@ip-172-31-91-170:/etc/ansible$ ansible-playbook play.yaml --check
```

i-0946d6e4167cbb958 (HOSTMachine)

PublicIPs: 18.206.56.70 PrivateIPs: 172.31.91.170

aws Services Search [Alt+S]

```
ubuntu@ip-172-31-91-170:/etc/ansible$ ansible-playbook play.yaml --syntax -check

playbook: play.yaml
ubuntu@ip-172-31-91-170:/etc/ansible$ ansible-playbook play.yaml --check

PLAY [install Jenkins and Java and docker on host] *****************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [localhost]

TASK [running script to install tools on host] *********************************************************************
skipping: [localhost]

PLAY [install K8s and Java on main] ********************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [172.31.91.136]

TASK [running script to install tools on main] *********************************************************************
skipping: [172.31.91.136]

PLAY [install K8s on nodes] ****************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [172.31.91.84]
ok: [172.31.81.83]

TASK [running script to install tools on node] *********************************************************************
skipping: [172.31.81.83]
skipping: [172.31.91.84]

PLAY RECAP *********************************************************************************************************
172.31.81.83               : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
172.31.91.136              : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
172.31.91.84               : 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-172-31-91-170:/etc/ansible$ ansible-playbook play.yaml

PLAY [install Jenkins and Java and docker on host] *****************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [localhost]

TASK [running script to install tools on host] *********************************************************************
changed: [localhost]

PLAY [install K8s and Java on main] ********************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [172.31.91.136]

TASK [running script to install tools on main] *********************************************************************
changed: [172.31.91.136]

PLAY [install K8s on nodes] ****************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [172.31.91.84]
ok: [172.31.81.83]

TASK [running script to install tools on node] *********************************************************************
changed: [172.31.91.84]
changed: [172.31.81.83]

PLAY RECAP *********************************************************************************************************
172.31.81.83               : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
172.31.91.136              : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
172.31.91.84               : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
localhost                  : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-91-170:/etc/ansible$
```

i-0946d6e4167cbb958 (HOSTMachine)

PublicIPs: 18.206.56.70 PrivateIPs: 172.31.91.170

Now on m3-K8-master – Run below commands

sudo kubeadm config images pull
sudo kubeadm init
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config

```
Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.91.136:6443 --token 2nzhbm.n0i7hb22etk9kpyb \
        --discovery-token-ca-cert-hash sha256:442d8319bcd3cbf30e1b4b426372a3ed38168c0572d6d7762b03d29630118569
ubuntu@ip-172-31-91-136:~$
```

i-040f177639d06c3d5 (m3-k8s-master)

PublicIPs: 52.90.136.233 PrivateIPs: 172.31.91.136

Install the "Calico Network" to run the cluster using this command: curl https://raw.githubusercontent.com/projectcalico/calico/v3.27.2/manifests/calico.yaml -O

Run this command: kubectl apply -f calico.yaml

Join slave machine through below command

kubeadm join 172.31.91.136:6443 --token 2nzhbm.n0i7hb22etk9kpyb \
    --discovery-token-ca-cert-hash
sha256:442d8319bcd3cbf30e1b4b426372a3ed38168c0572d6d7762b03d29630118569 –v=5

```
This node has joined the cluster:
* Certificate signing request was sent to apiserver and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

root@ip-172-31-81-83:/home/ubuntu#
```

i-0e54c3e9a1ce6158d (m2-k8s-slave1)

PublicIPs: 18.208.167.49 PrivateIPs: 172.31.81.83

```
This node has joined the cluster:
* Certificate signing request was sent to apiserver and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

root@ip-172-31-91-84:/home/ubuntu#
```

i-0f6ea9918b6791df6 (m4-k8s-slave2)

PublicIPs: 100.26.101.84 PrivateIPs: 172.31.91.84

```
ubuntu@ip-172-31-91-136:~$ kubectl get no
NAME                STATUS   ROLES           AGE     VERSION
ip-172-31-81-83     Ready    <none>          2m20s   v1.29.0
ip-172-31-91-136    Ready    control-plane   18m     v1.29.0
ip-172-31-91-84     Ready    <none>          2m36s   v1.29.0
ubuntu@ip-172-31-91-136:~$
```

i-040f177639d06c3d5 (m3-k8s-master)

PublicIPs: 52.90.136.233 PrivateIPs: 172.31.91.136

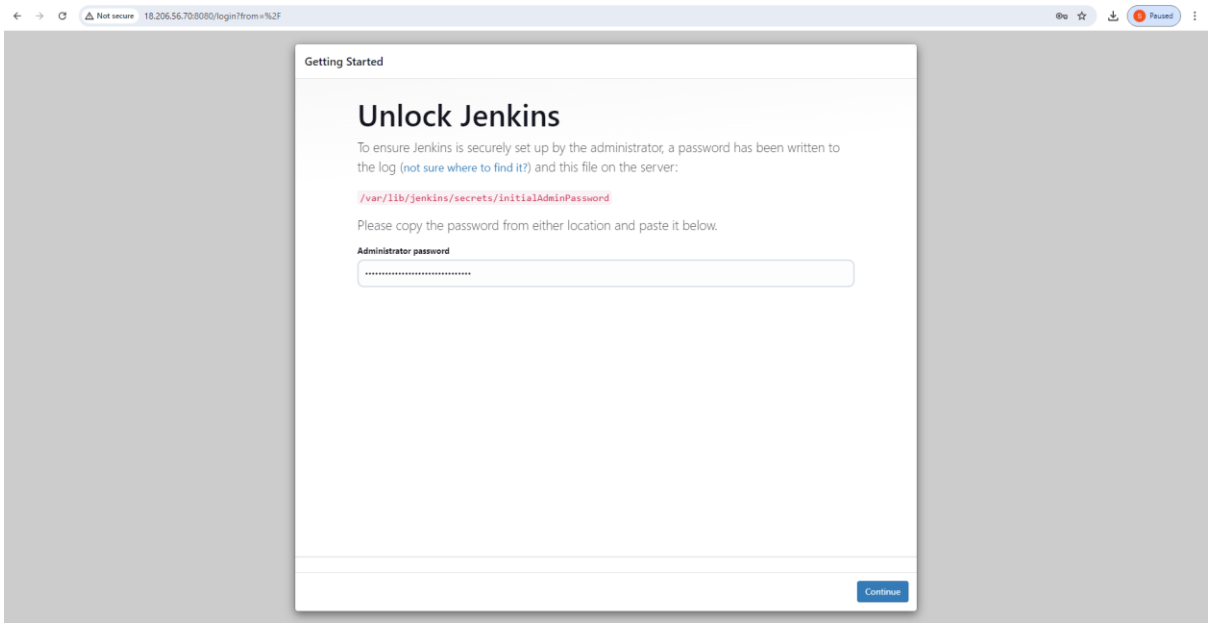
Paste IP of Host Machine on browser with 8080

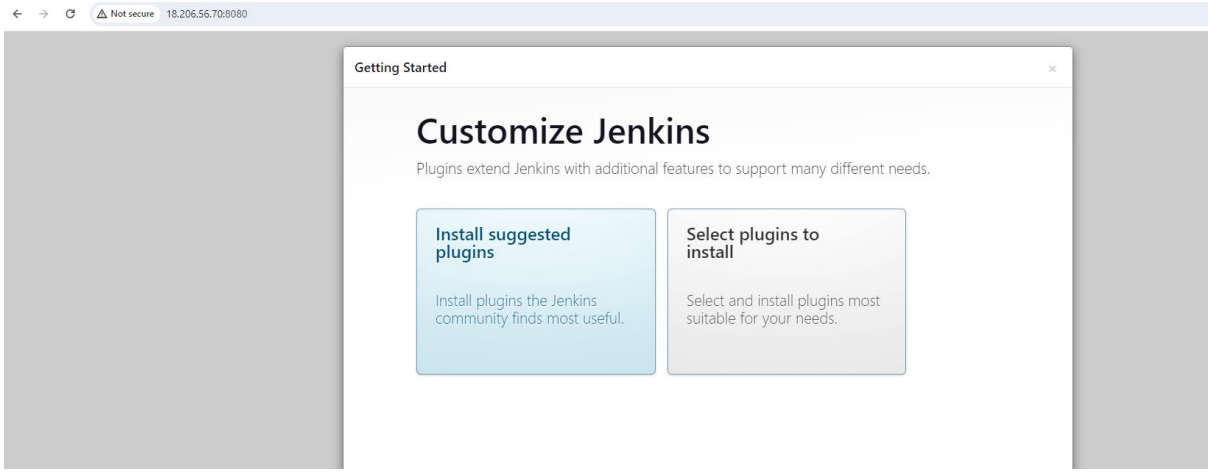
```
ubuntu@ip-172-31-91-170:~$ jenkins --version
2.476
ubuntu@ip-172-31-91-170:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
e4e942c48ccf4f50b4e0c0917690bb94
ubuntu@ip-172-31-91-170:~$
```

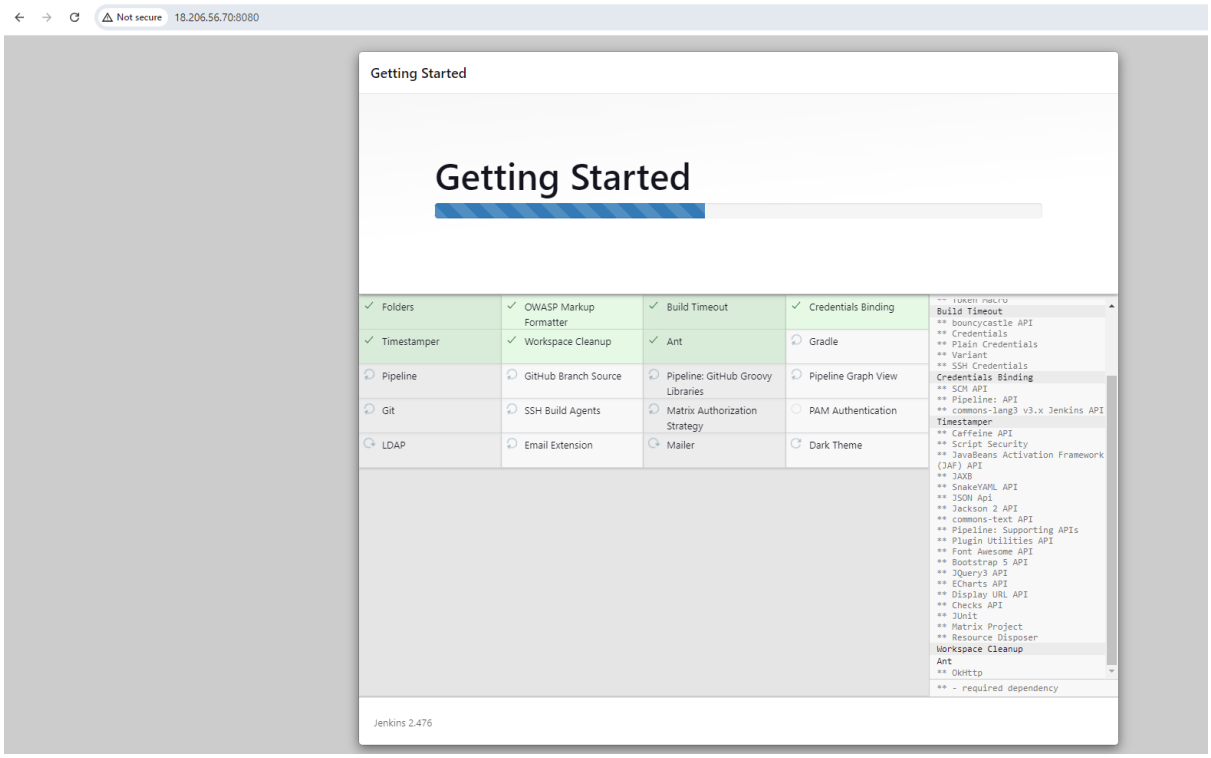
i-0946d6e4167cbb958 (HOSTMachine)

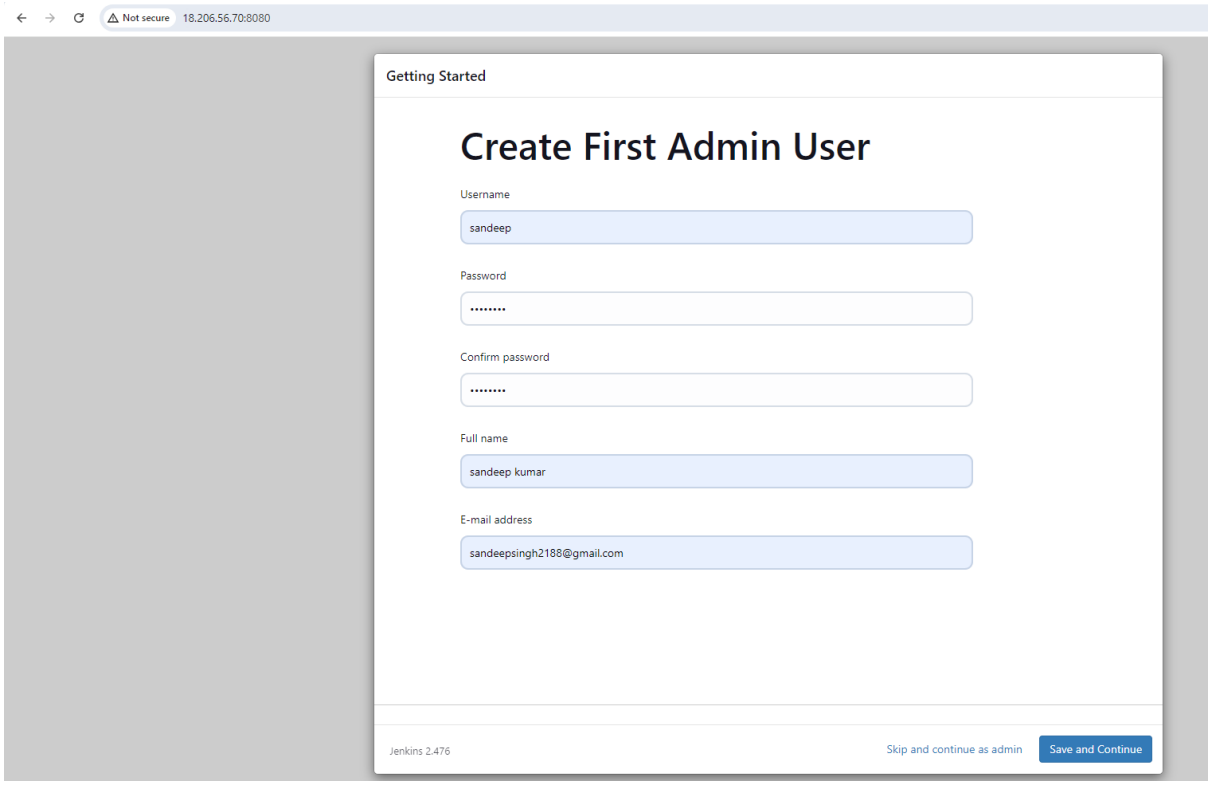
PublicIPs: 18.206.56.70 PrivateIPs: 172.31.91.170

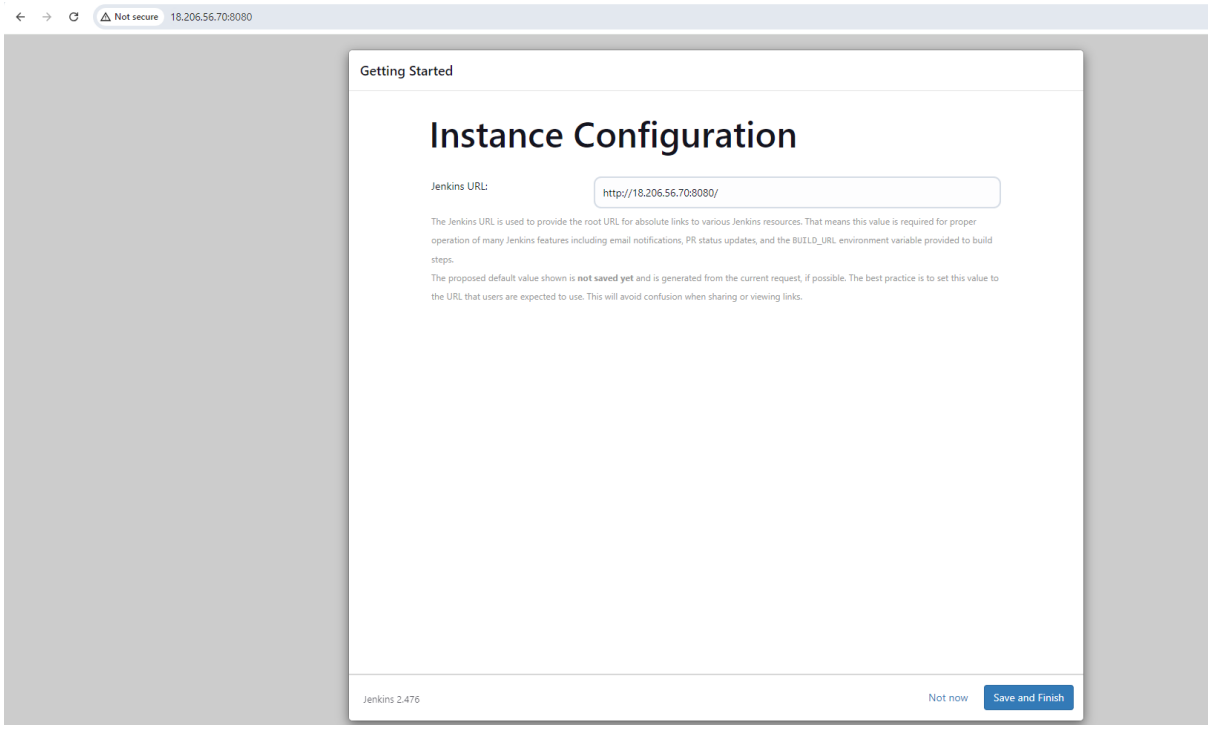
Run on terminal and copy paste the password and again paste on Jenkins page

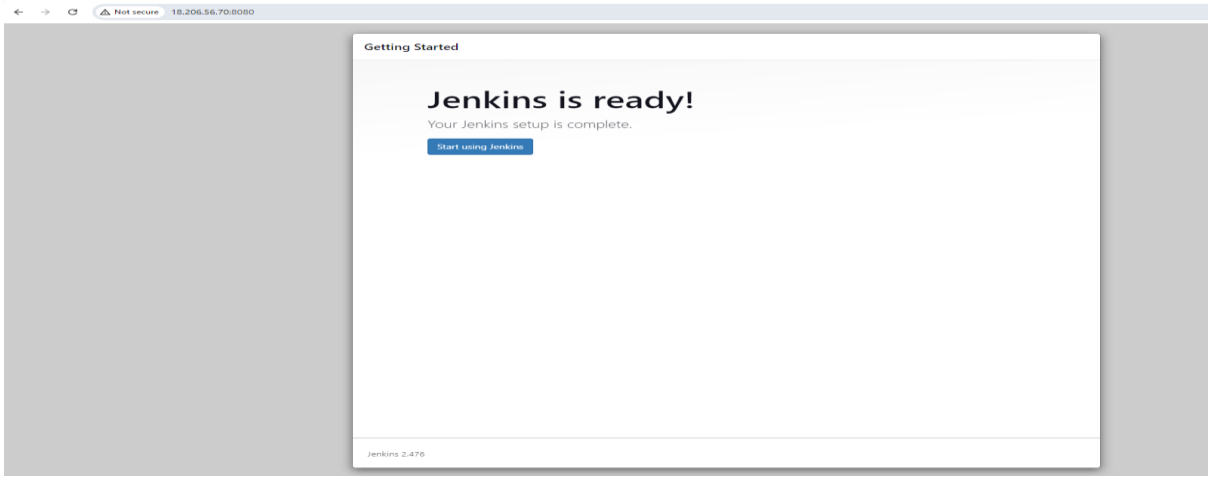





Not secure 18.206.56.70:8080
Getting Started
Getting Started
Folders
OWASP Markup Formatter
Build Timeout
Credentials Binding
Timestamper
Workspace Cleanup
Ant
Gradle
Pipeline
GitHub Branch Source
Pipeline: GitHub Groovy Libraries
Pipeline Graph View
Git
SSH Build Agents
Matrix Authorization Strategy
PAM Authentication
LDAP
Email Extension
Mailer
Dark Theme
Build Timeout
** bouncycastle API
** Credentials
** Plain Credentials
** Variant
** SSH Credentials
Credentials Binding
** SCM API
** Pipeline: API
** commons-lang3 v3.x Jenkins API
Timestamper
** Caffeine API
** Script Security
** JavaBeans Activation Framework (JAF) API
** JAXB
** SnakeYAML API
** JSON Api
** Jackson 2 API
** commons-text API
** Pipeline: Supporting APIs
** Plugin Utilities API
** Font Awesome API
** Bootstrap 5 API
** JQuery3 API
** ECharts API
** Display URL API
** Checks API
** JUnit
** Matrix Project
** Resource Disposer
Workspace Cleanup
Ant
** OkHttp
** - required dependency
Jenkins 2.476

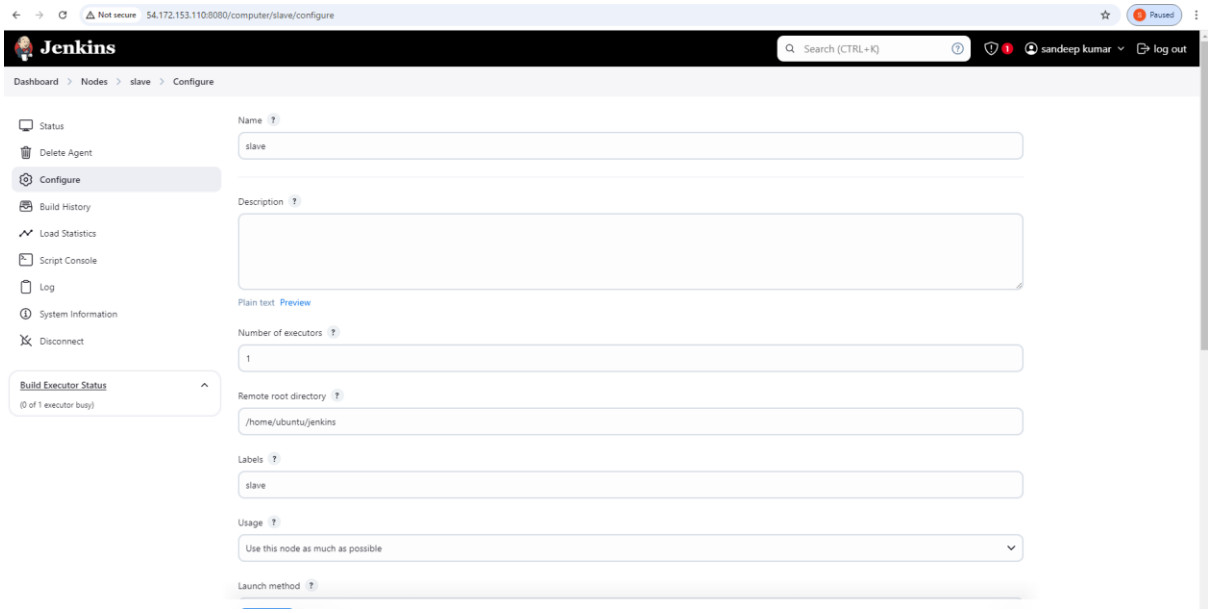

Not secure 18.206.56.70:8080
Getting Started
Create First Admin User
Username
sandeep
Password
Confirm password
Full name
sandeep kumar
E-mail address
sandeepsingh2188@gmail.com
Jenkins 2.476
Skip and continue as admin
Save and Continue



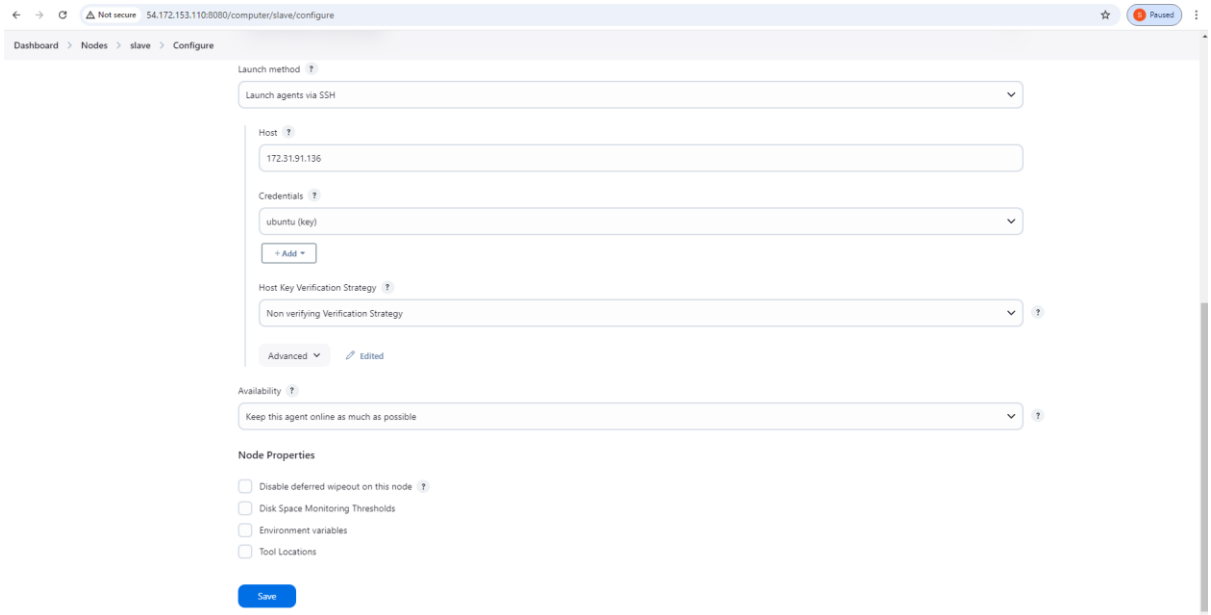


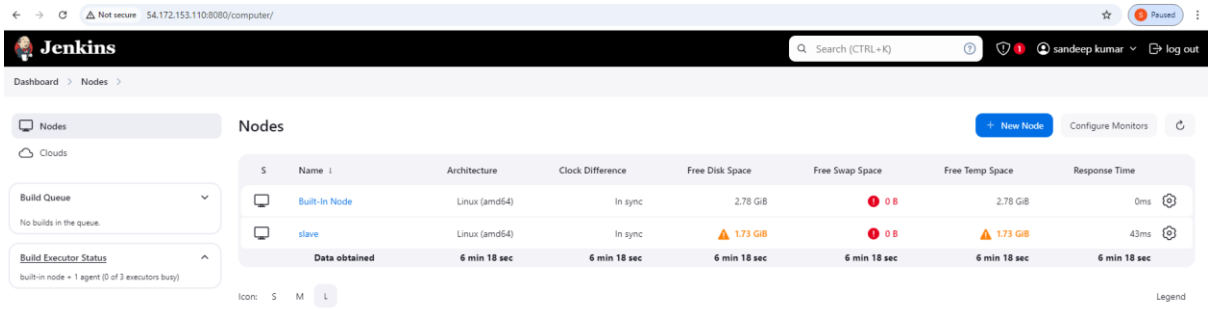
Install SSH Agent

Now Create a Node



Add credential with .pem key





Create item

Not secure 54.172.153.110:8080/job/Testpipeline/configure

Jenkins

Search (CTRL+K) sandeep kumar log out

Dashboard > Testpipeline > Configuration

Configure

- General
- Advanced Project Options
- Pipeline

General

Enabled

Description

Testpipeline

Plain text Preview

- [ ] Discard old builds
- [ ] Do not allow concurrent builds
- [ ] Do not allow the pipeline to resume if the controller restarts
- [ ] GitHub project
- [ ] Pipeline speed/durability override
- [ ] Preserve stashes from completed builds
- [ ] This project is parameterized
- [ ] Throttle builds

Build Triggers

- [ ] Build after other projects are built

Not secure 54.172.153.110:8080/job/Testpipeline/configure

Dashboard > Testpipeline > Configuration

Configure

- General
- Advanced Project Options
- Pipeline

- [ ] Build after other projects are built
- [ ] Build periodically
- [x] GitHub hook trigger for GITScm polling
- [ ] Poll SCM
- [ ] Quiet period
- [ ] Trigger builds remotely (e.g., from scripts)

Advanced Project Options

Advanced

Pipeline

Definition

Pipeline script

Script

```
pipeline {
    agent none
    environment {
        DOCKERHUB_CREDENTIALS = credentials('412ec374-fc79-4d99-9008-d66a856e1af6')
    }
    stages {
        stage('Hello') {
            steps {
                echo 'Hello World'
            }
        }
        stage('git') {
            agent {
                label 'slave'
            }
            steps {
                git 'https://github.com/Sandeepsingh2188/website.git'
```

Save Apply

Not secure 54.172.153.110:8080/me/my-views/view/all/

Jenkins

Search (CTRL+K) sandeep kumar log out

Dashboard > sandeep kumar > My Views > All >

- New Item
- Build History
- Project Relationship
- Check File Fingerprint

Build Queue

No builds in the queue.

Build Executor Status

built-in node + 2 agents (0 of 5 executors busy)

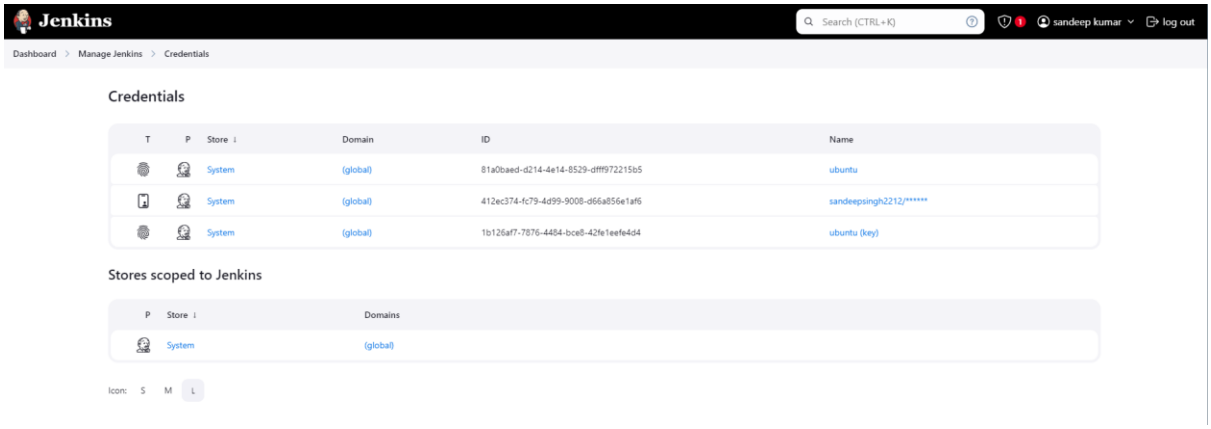
Add description

All +

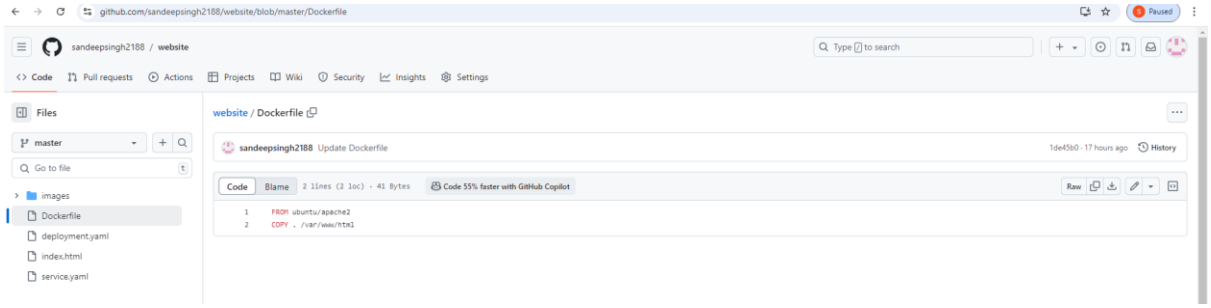
S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☁	Testpipeline	17 hr #16	17 hr #13	18 sec

Icon: S M L

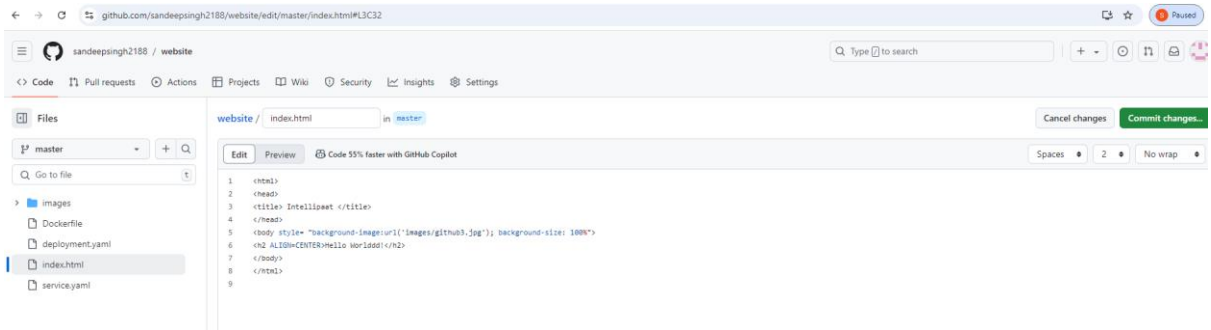
Created docker hub credential



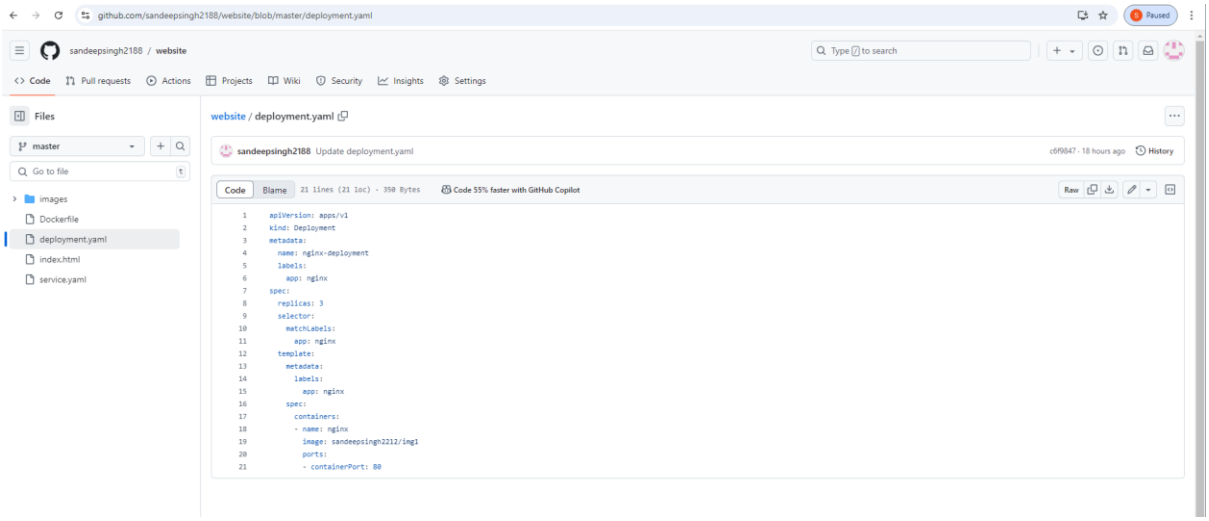
Create dockerfile and committed



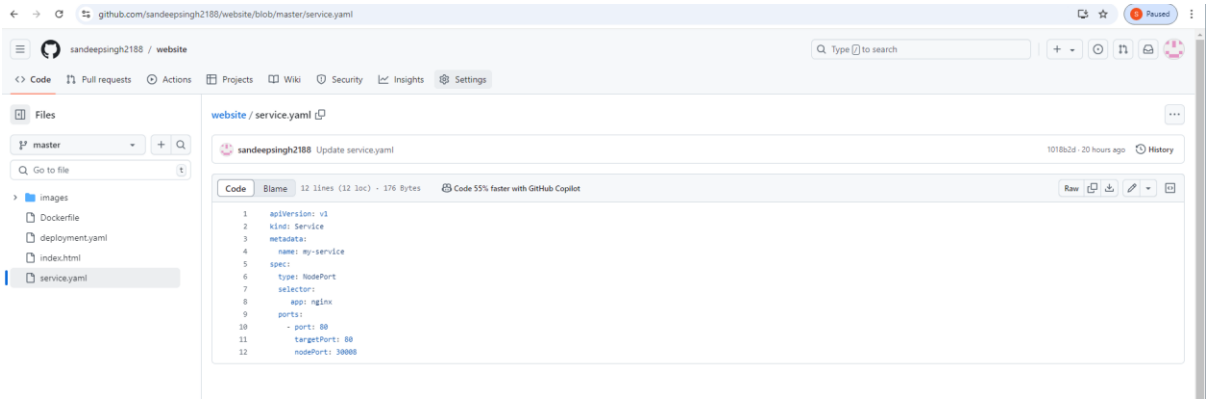
Index.html



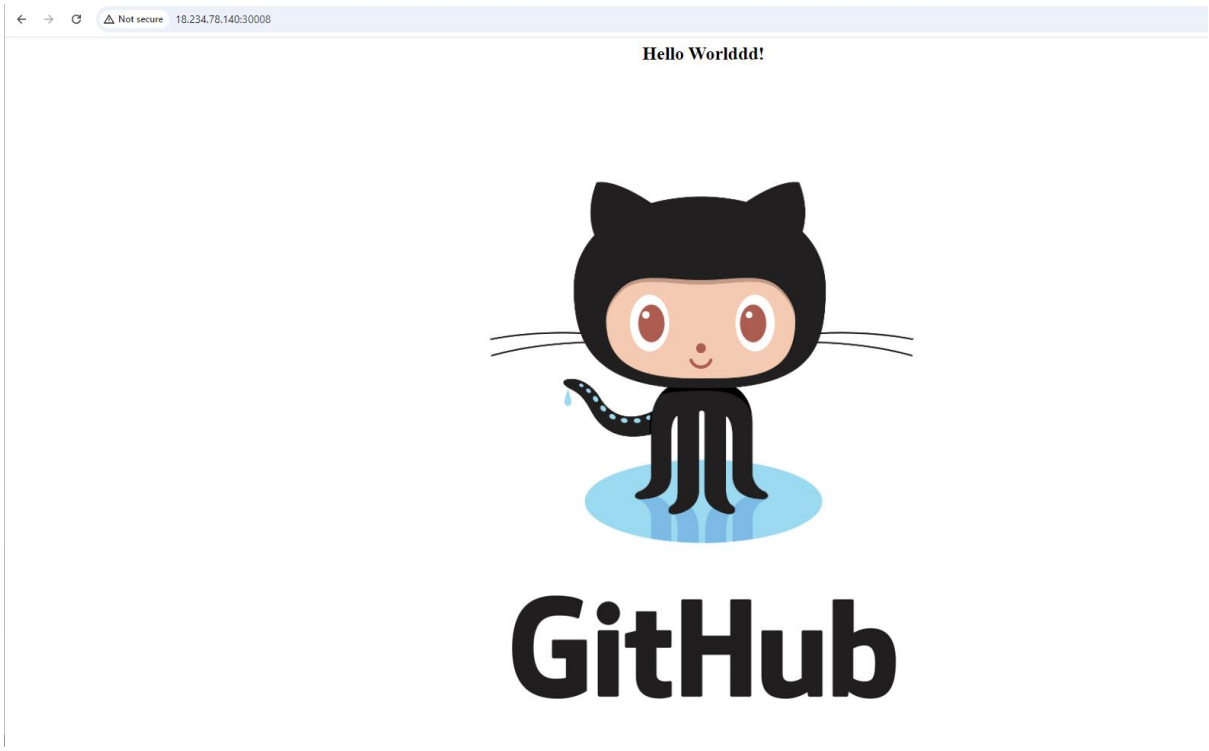
Deployment.yaml

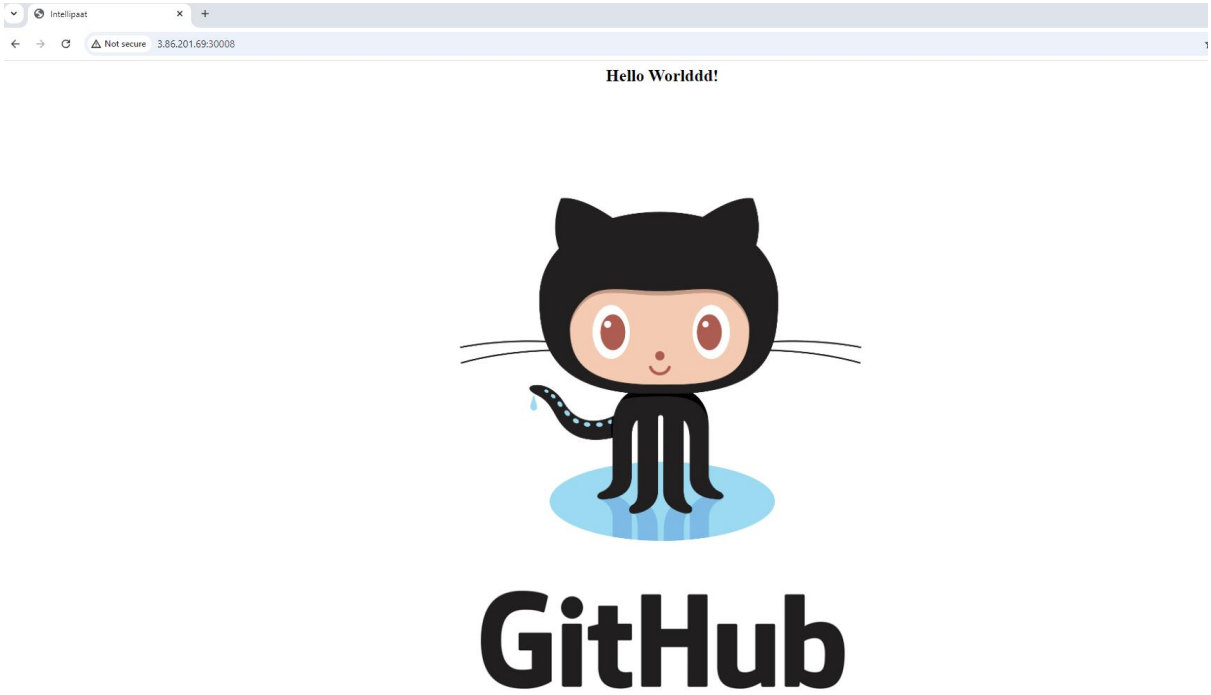


Service.yaml



Copy Slave 1 and slave 2 public ip with 30008
Output attached below





Project 2 Completed

By
SANDEEP KUMAR
9453743921