swe645

Assignment HW2

Aditya Ketanbhai Shah – G01449892
Suhas Thandaga Ramesh – G01450932
Niharika FNU LNU – G01412073
Syedfurkhan Feroz Hashmi – G01411748

Installation and Setup Instruction:

In this project swe645 HW2 has been dockerized and then deployed to self-hosted ec2 cluster in aws.

1. Rancher Node: http://54.226.23.128:80
2. Form URL: http://54.226.23.128:30000/
3. Jenkins URL: http://54.226.23.128:8080
4. Github Repo URL: https://github.com/suhastr/jenkins_dummy

Setting up GitHub:

1. Push the files that you created in HW1 to a git repo (GitHub or BitBucket). First create an empty repository and then use Visual Studio Code commands to push the code a new repository.

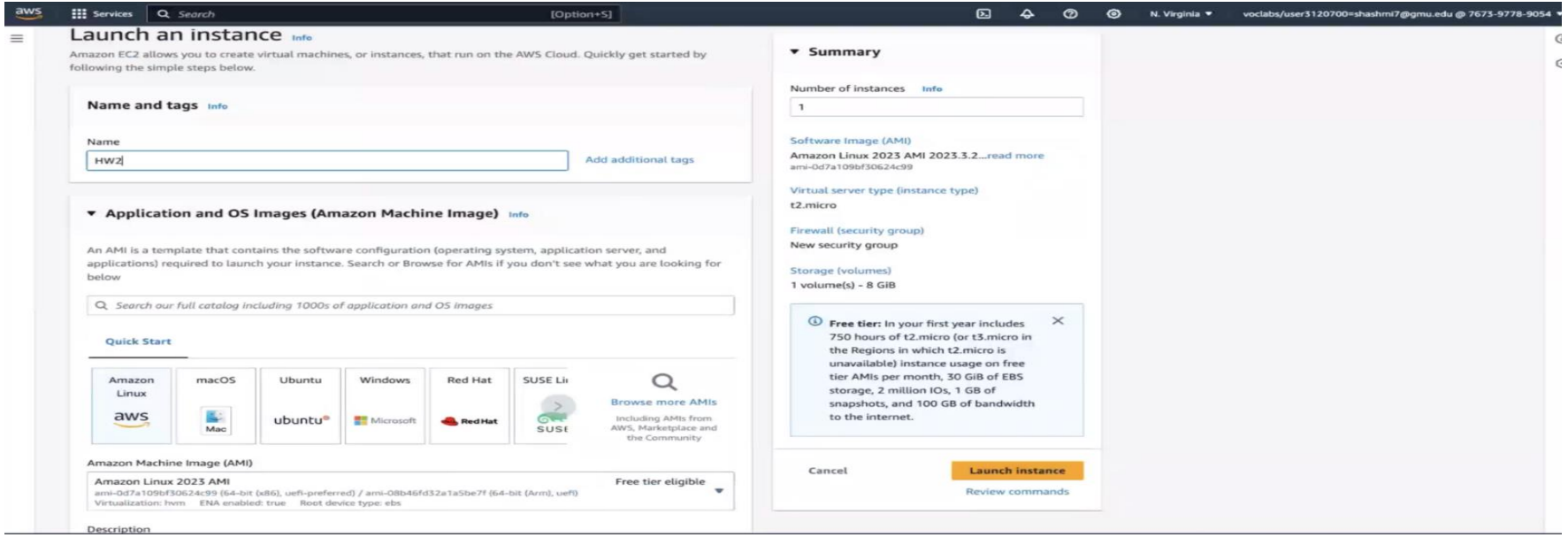
Docker installation Steps:

1. Make sure you have docker installed on your machine. I am using Docker Desktop . You will also need to create an account on https://hub.docker.com/
2. In the DockerFile, use the FROM command to get the initial image for the build.
3. On the command line, use this command: ‘docker build –tag userform’ You can use whatever name and tag you want.
4. Verify that the image is properly working by running ‘docker run -it -p 8182:8080 userform’ and open a browser at http://localhost:8182/userform

5. On the command line, login to docker using ‘docker login -u ’ i. Change the name of you image to be /: using the docker tag command. In my case it is: ‘docker tag userform hekme5/userform
6. Verify that your image is on Docker Hub. Your image is accessible from the internet.

Setup Kubernetes Cluster using Rancher:

Spin up EC2 host with atleast t2.medium as rancher consumes more memory.

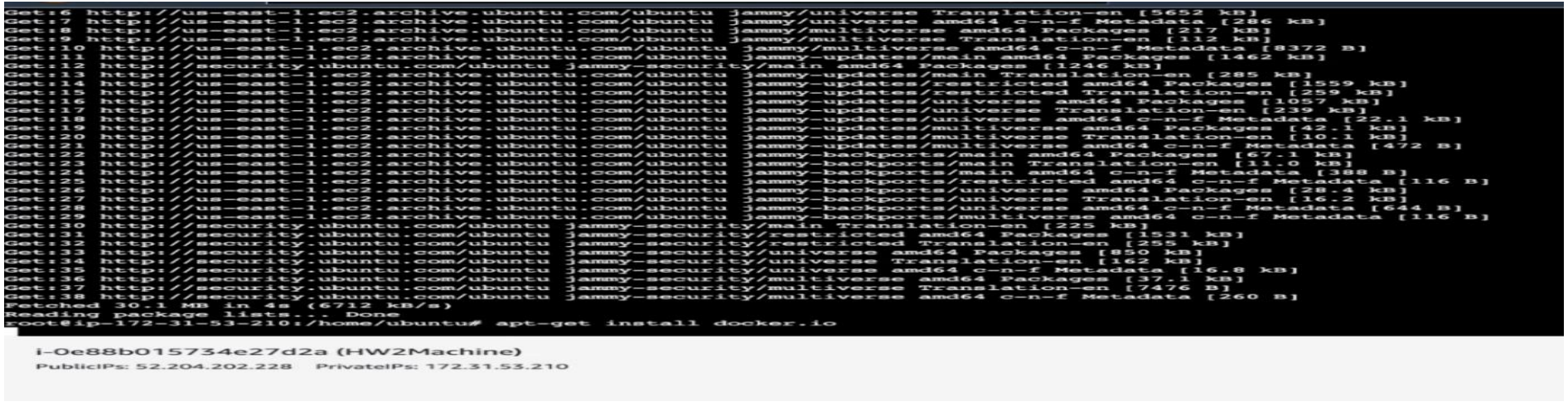


1.sudo su

2.sudo apt update

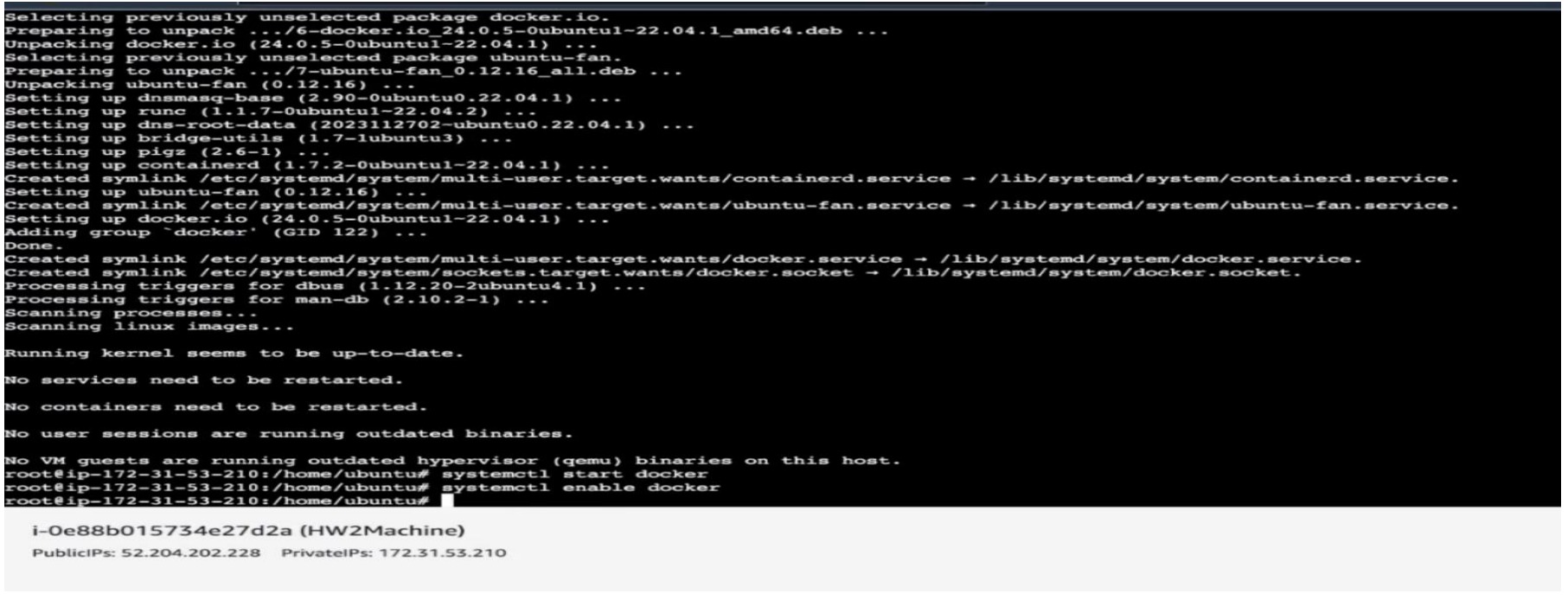
Install docker in EC2 host created for Rancher for this usecase Ubuntu image is used.

3.sudo apt install docker.io

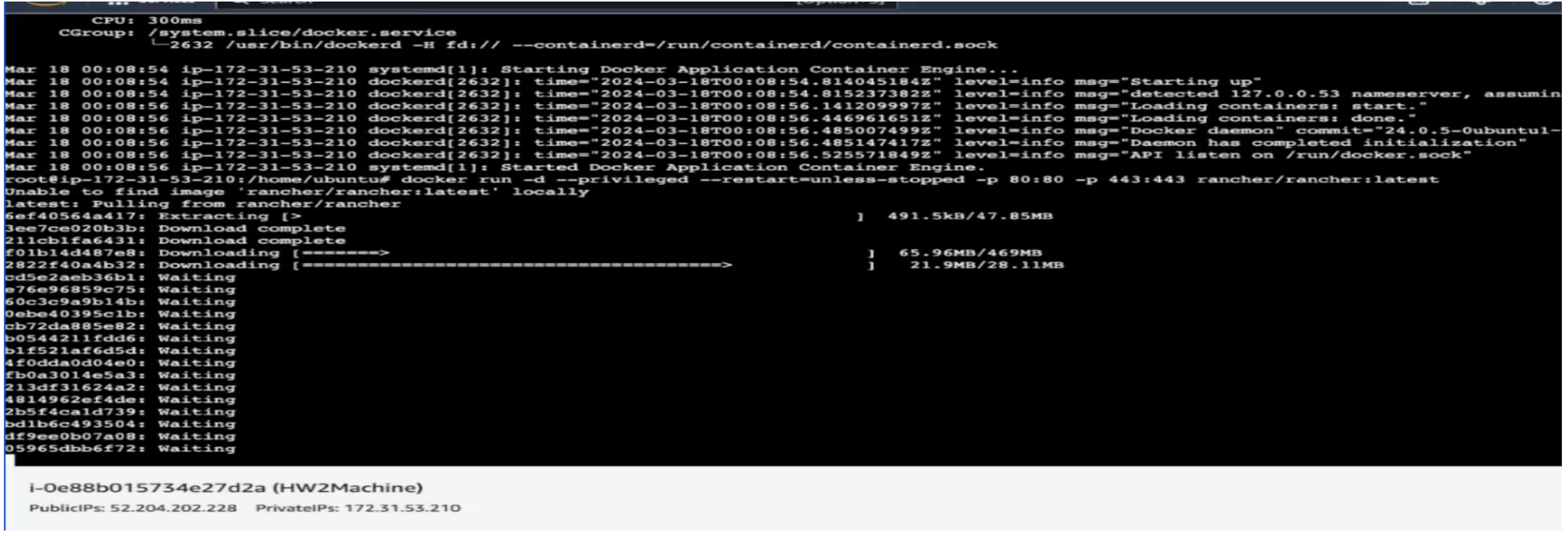


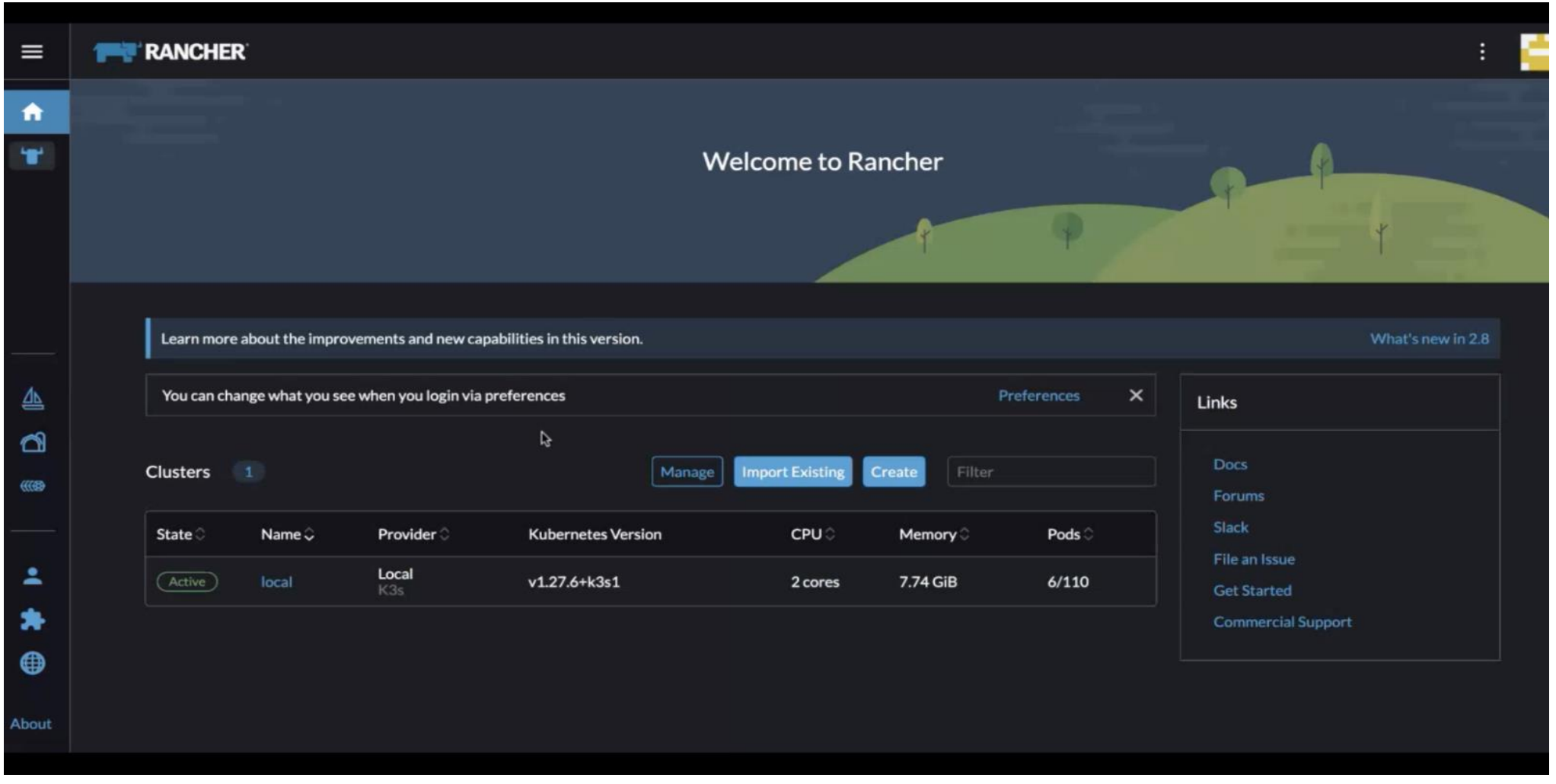
4.systemctl start docker

5.systemctl enable docker

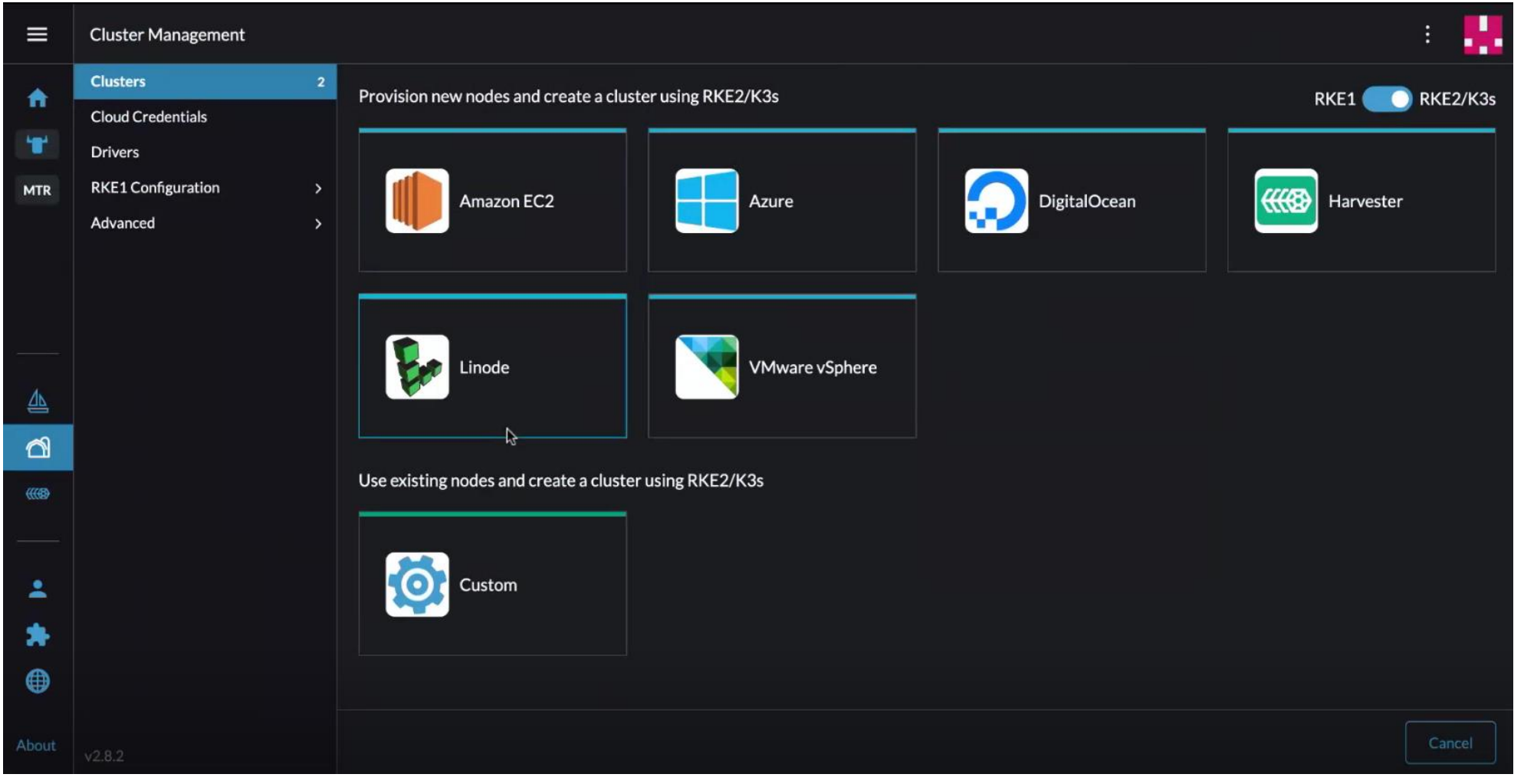


6.docker run -d --privileged --restart=unless-stopped -p 80:80 -p 443:443 rancher/rancher:latest

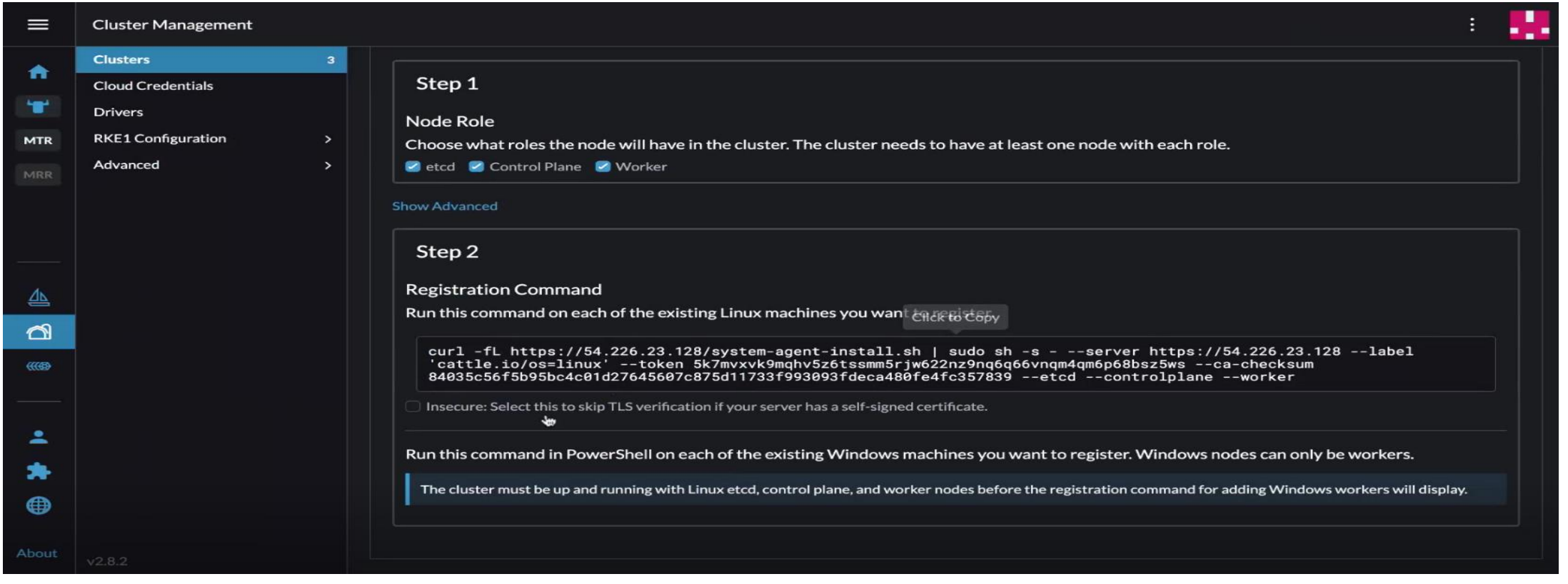



RANCHER
Welcome to Rancher
Learn more about the improvements and new capabilities in this version.
What's new in 2.8
You can change what you see when you login via preferences
Preferences
Clusters 1
Manage
Import Existing
Create
Filter
State
Name
Provider
Kubernetes Version
CPU
Memory
Pods
Active
local
Local
K3s
v1.27.6+k3s1
2 cores
7.74 GiB
6/110
Links
Docs
Forums
Slack
File an Issue
Get Started
Commercial Support
About

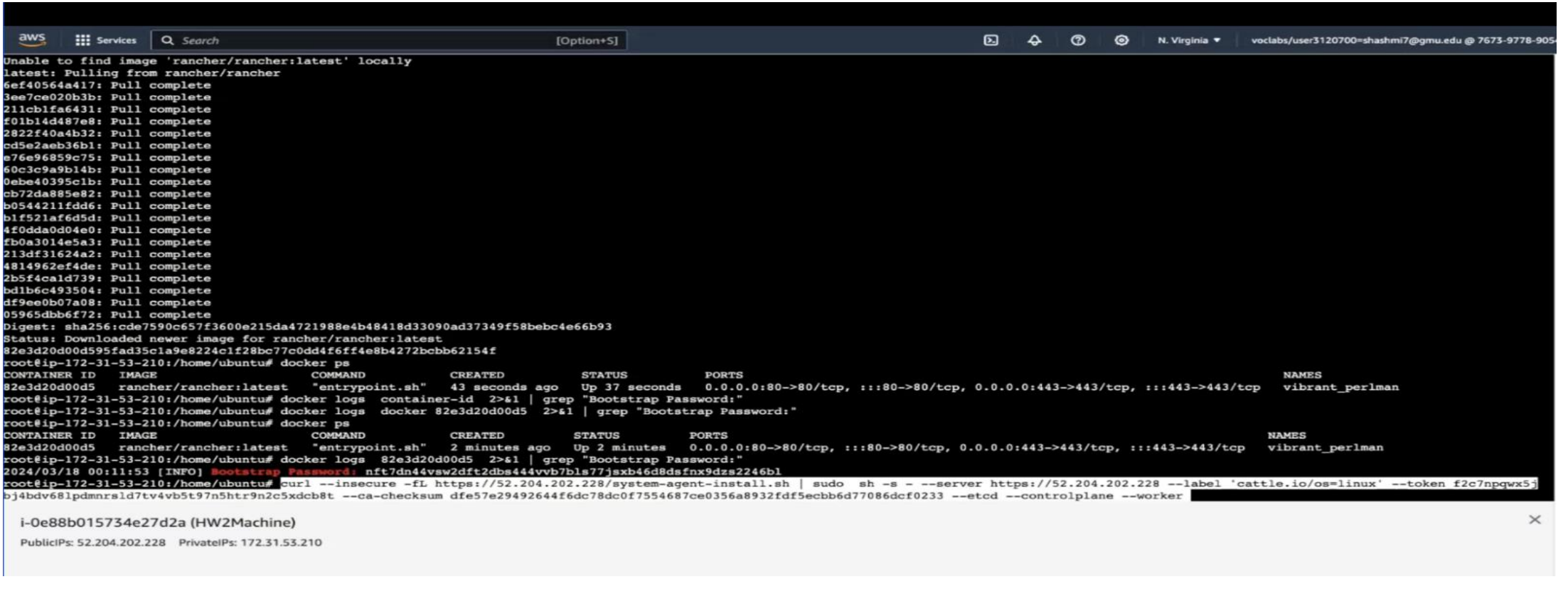
7. Local cluster is created by default. We cannot schedule our application pods in that as it has taints to it.
8. Ec2 machine with tier t2.large this will be used as a cluster for deploying application code.
9. In Racher UI navigate to cluster management for creating new cluster. There are various options present We can use different cloud providers or even use an ec2 iam user for managing the cluster. For our use case we have used custom one.

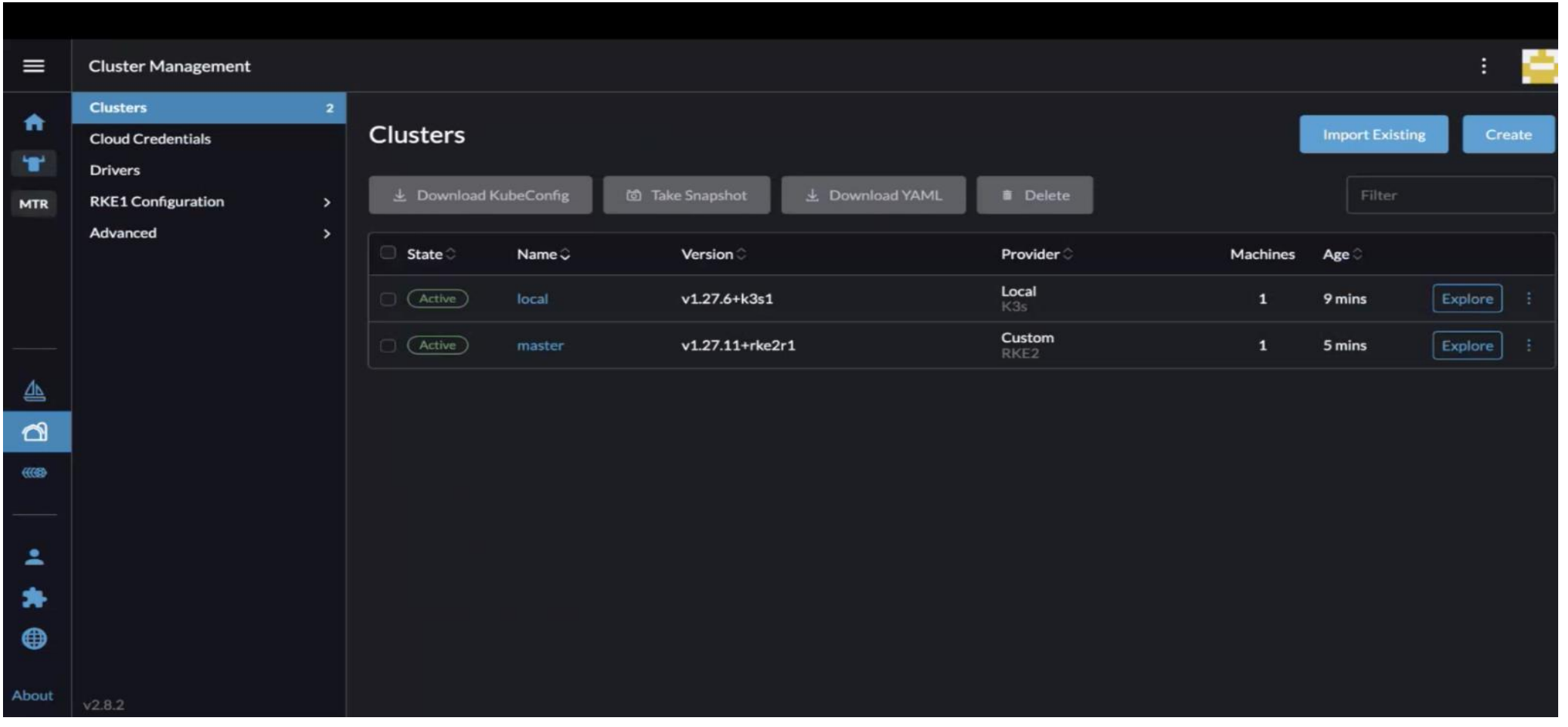

Cluster Management
Clusters 2
Cloud Credentials
Drivers
RKE1 Configuration
Advanced
MTR
About
v2.8.2
Provision new nodes and create a cluster using RKE2/K3s
RKE1
RKE2/K3s
Amazon EC2
Azure
DigitalOcean
Harvester
Linode
VMware vSphere
Use existing nodes and create a cluster using RKE2/K3s
Custom
Cancel

10. In case of custom when the cluster is created in order to make it active we have to register it with the Rancher node which was previously created. For this we have to execute the registration command in new EC2 node provided in Rancher UI. Once this is done cluster will appear as active in 4-5 mins.



11. Adding Master to the cluster:




Cluster Management
Clusters 2
Cloud Credentials
Drivers
RKE1 Configuration
Advanced
MTR
About
v2.8.2
Clusters
Import Existing
Create
Download KubeConfig
Take Snapshot
Download YAML
Delete
Filter
State
Name
Version
Provider
Machines
Age
Active
local
v1.27.6+k3s1
Local
K3s
1
9 mins
Explore
Active
master
v1.27.11+rke2r1
Custom
RKE2
1
5 mins
Explore

12. Creating deployment and service file :

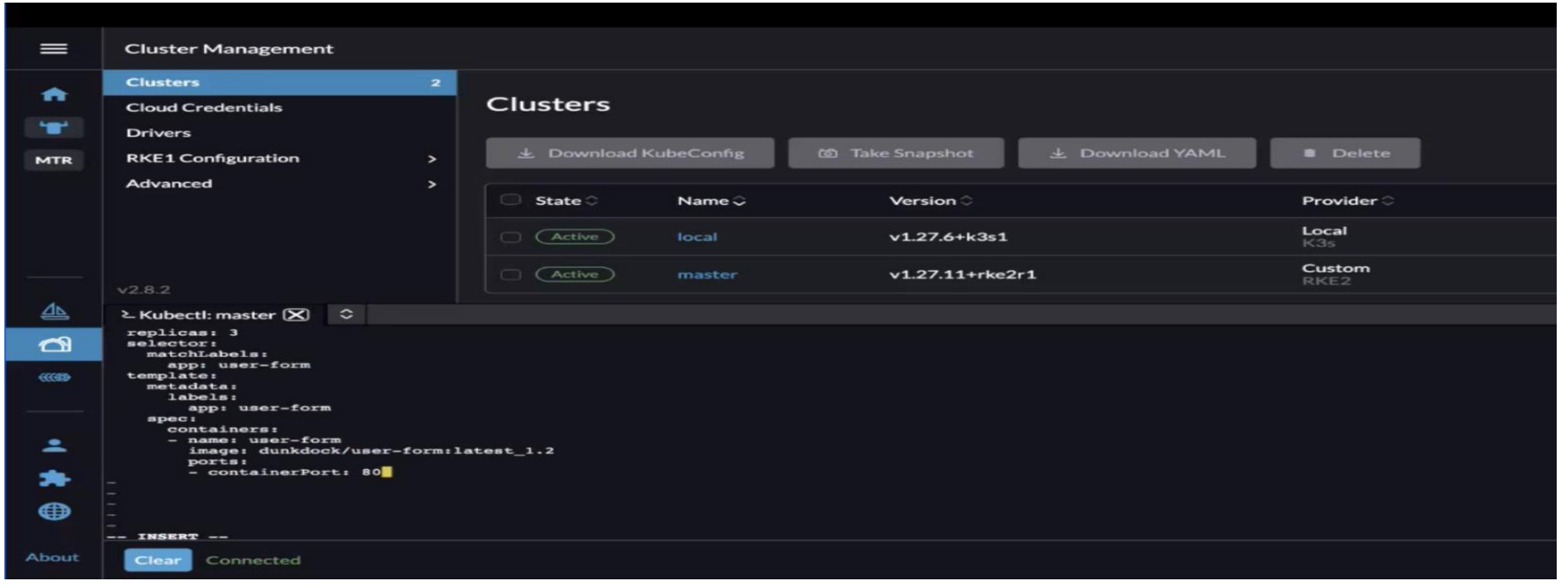


Fig. Deployment.yaml

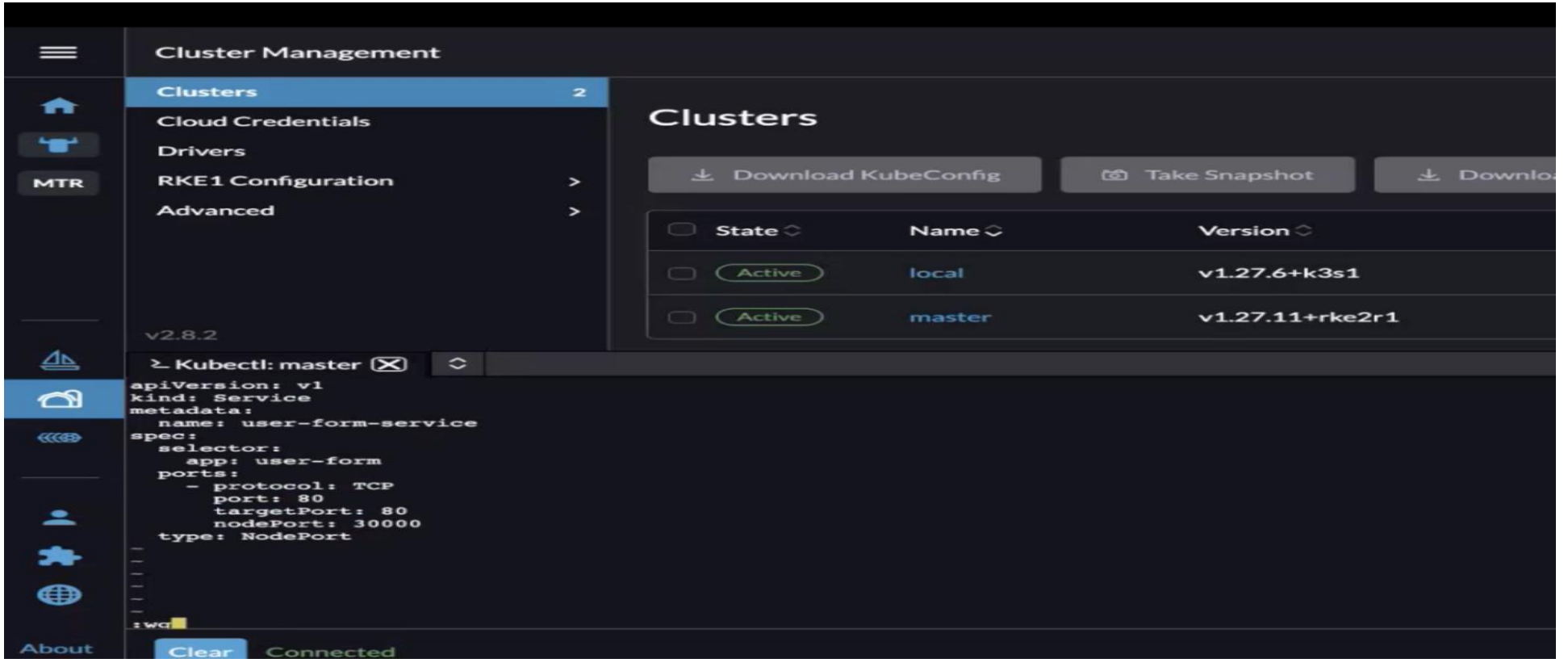
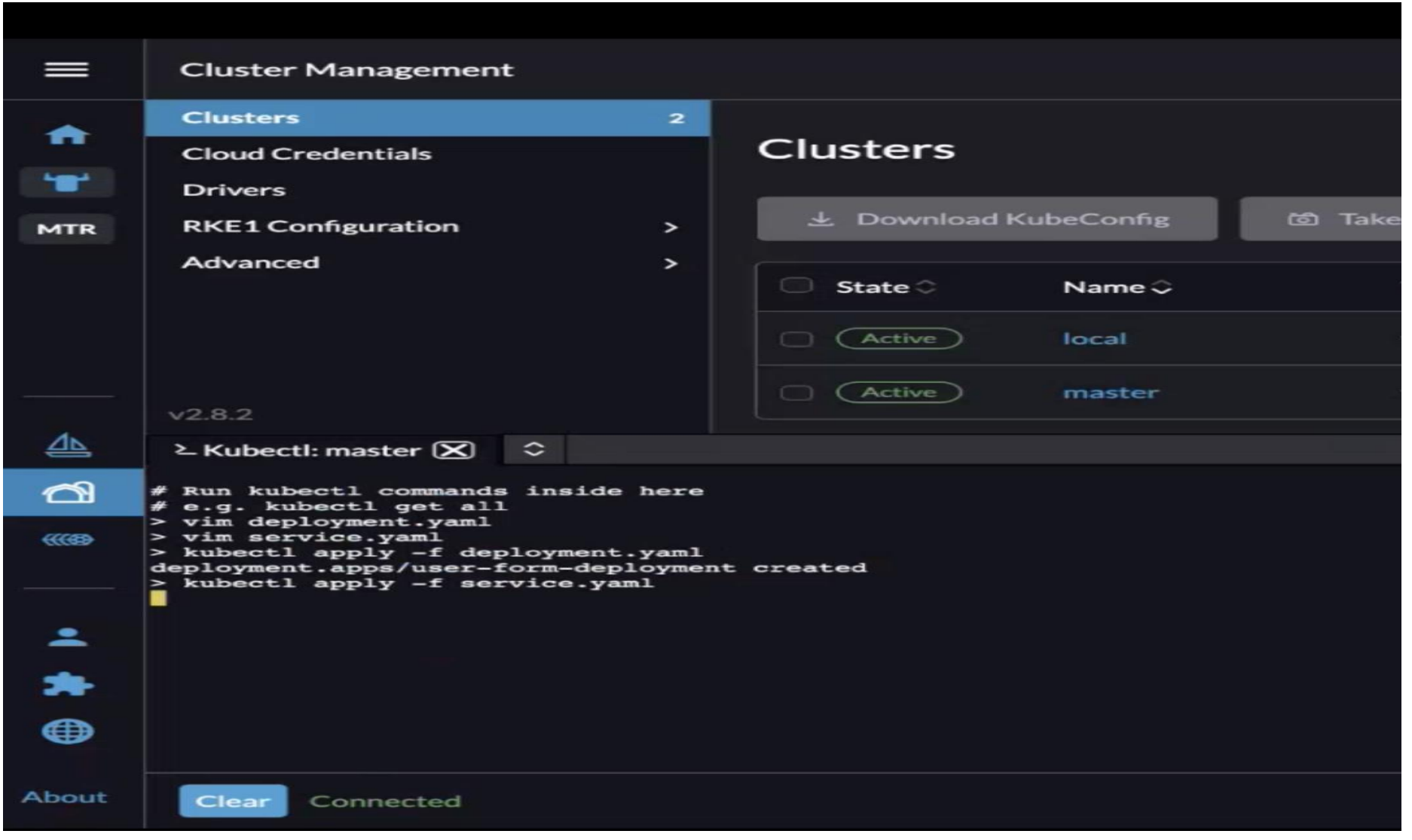
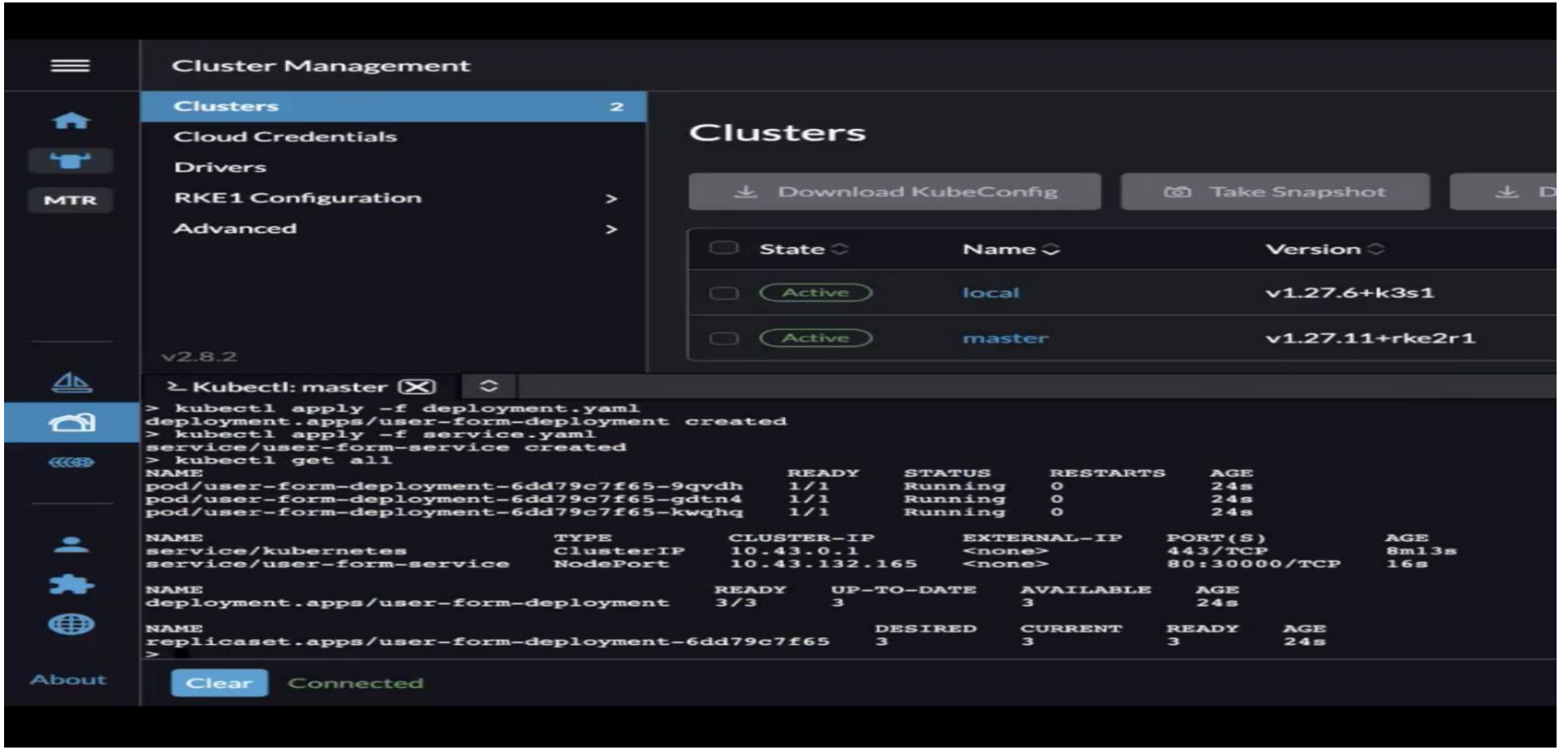


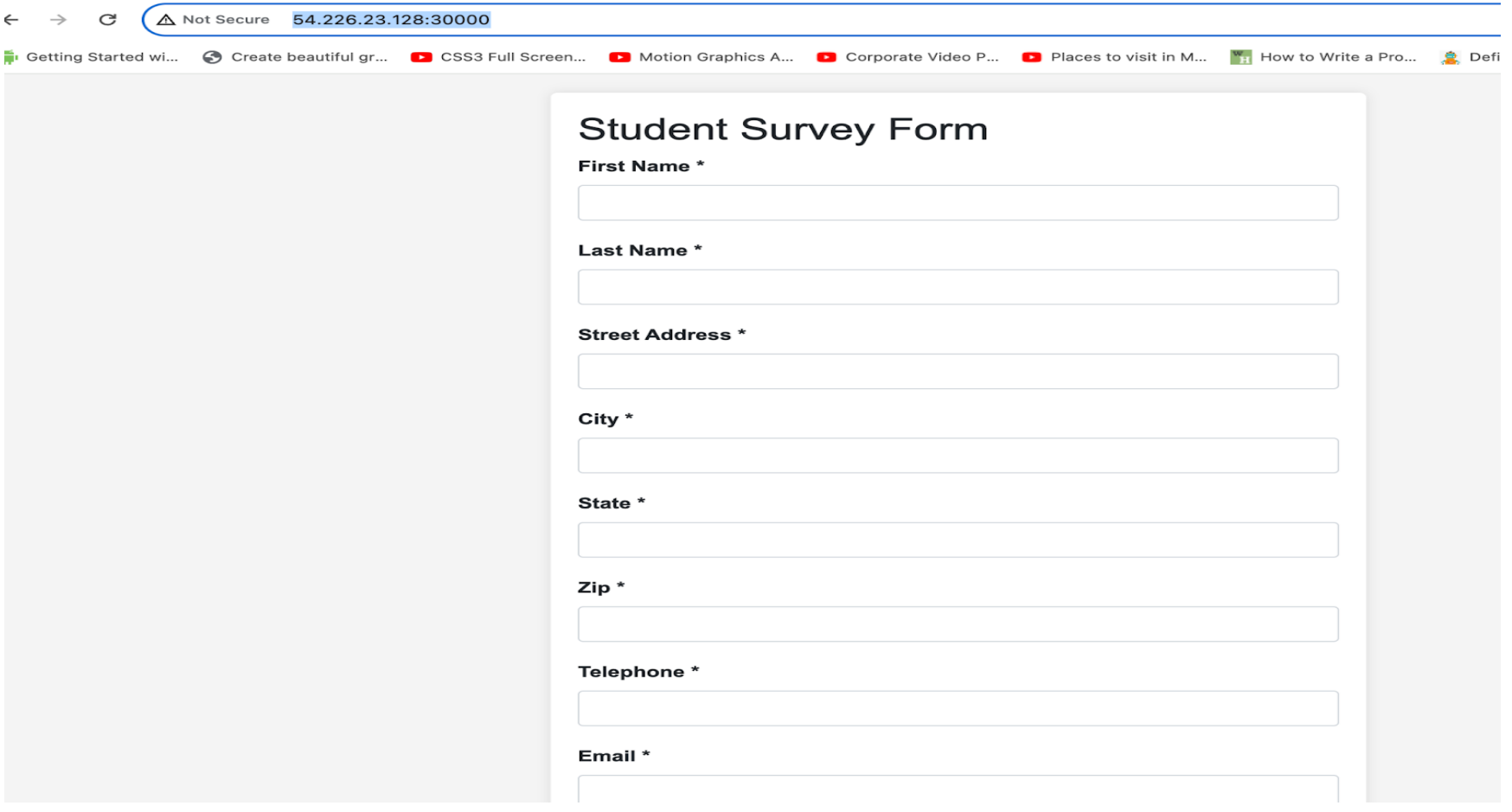
Fig. Service.yaml


Cluster Management
Clusters 2
Cloud Credentials
Drivers
RKE1 Configuration
Advanced
v2.8.2
MTR
About
Clusters
Download KubeConfig
Take
State
Name
Active
local
Active
master
Kubectl: master
Run kubectl commands inside here
e.g. kubectl get all
> vim deployment.yaml
> vim service.yaml
> kubectl apply -f deployment.yaml
deployment.apps/user-form-deployment created
> kubectl apply -f service.yaml
Clear
Connected

13. Displaying pod status:

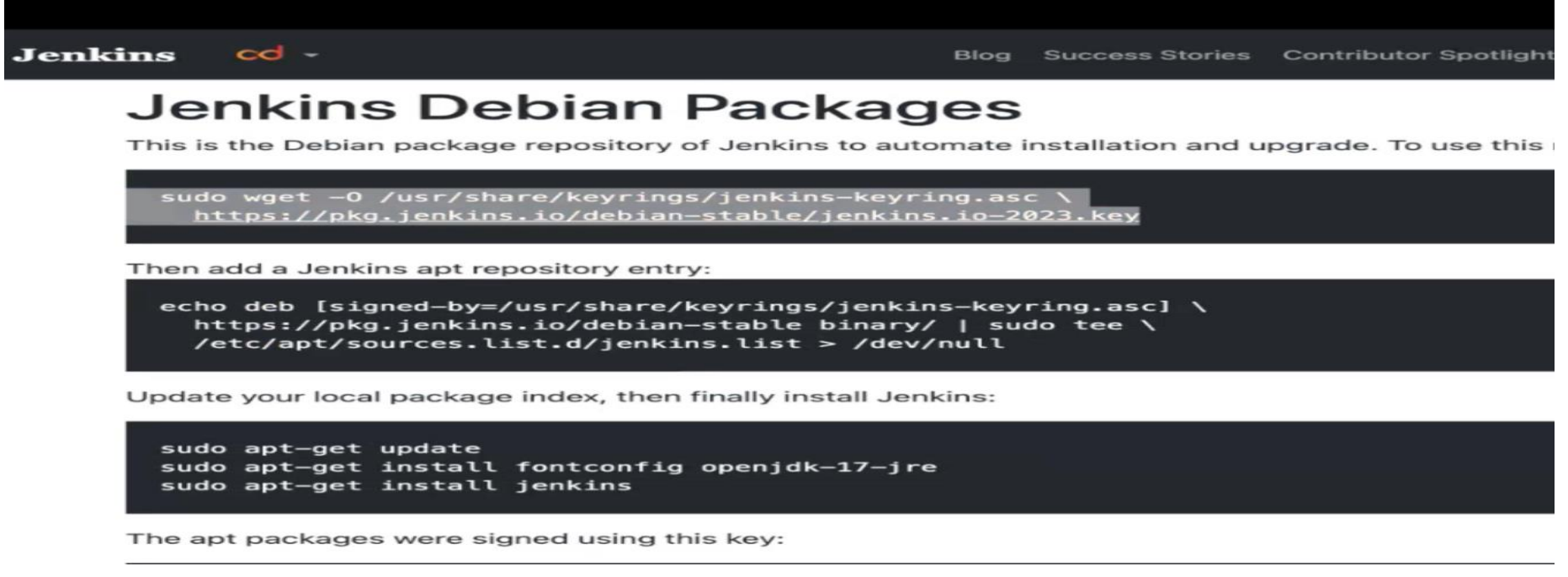


14. Now access this Form:



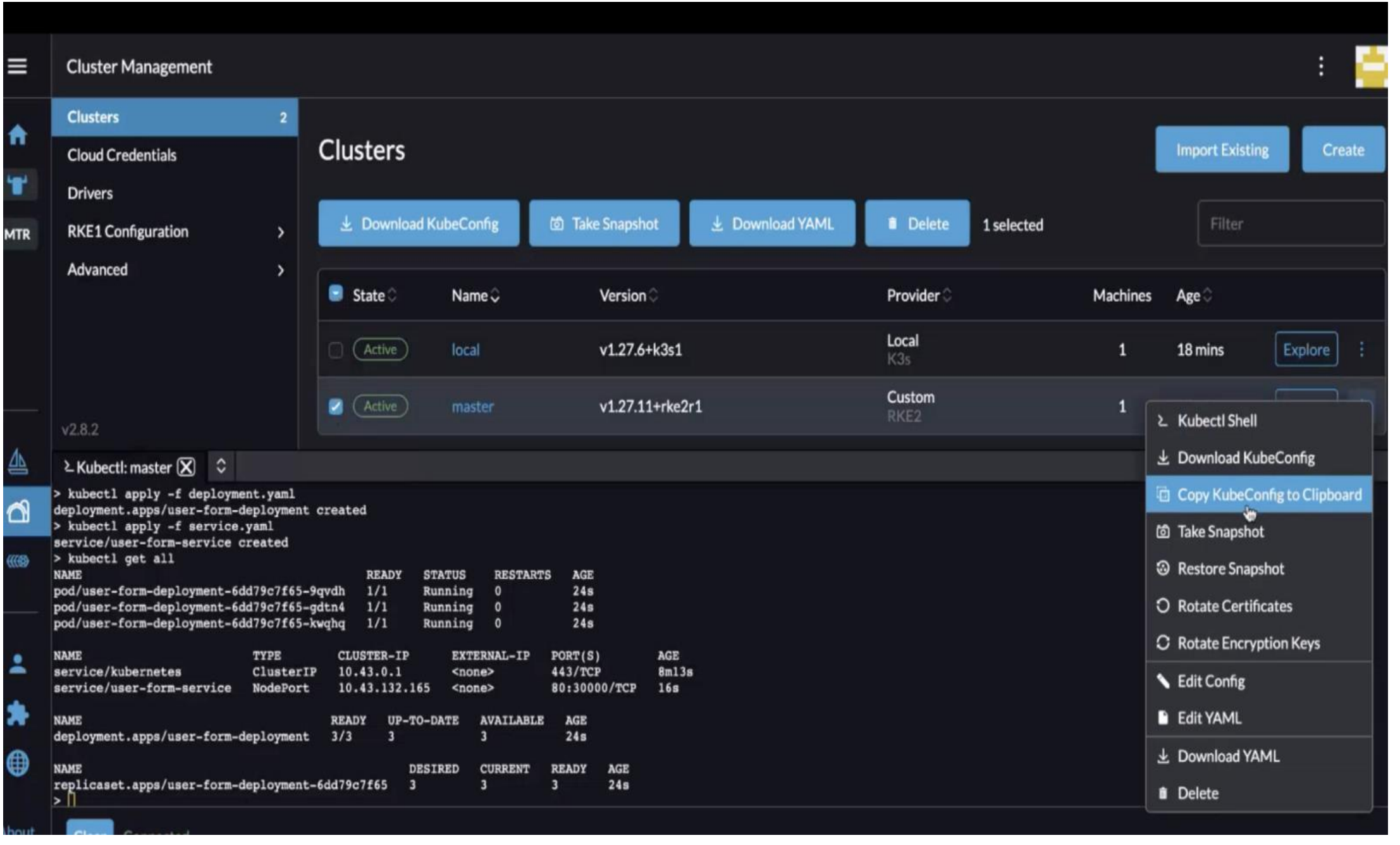
Jenkins Setup:

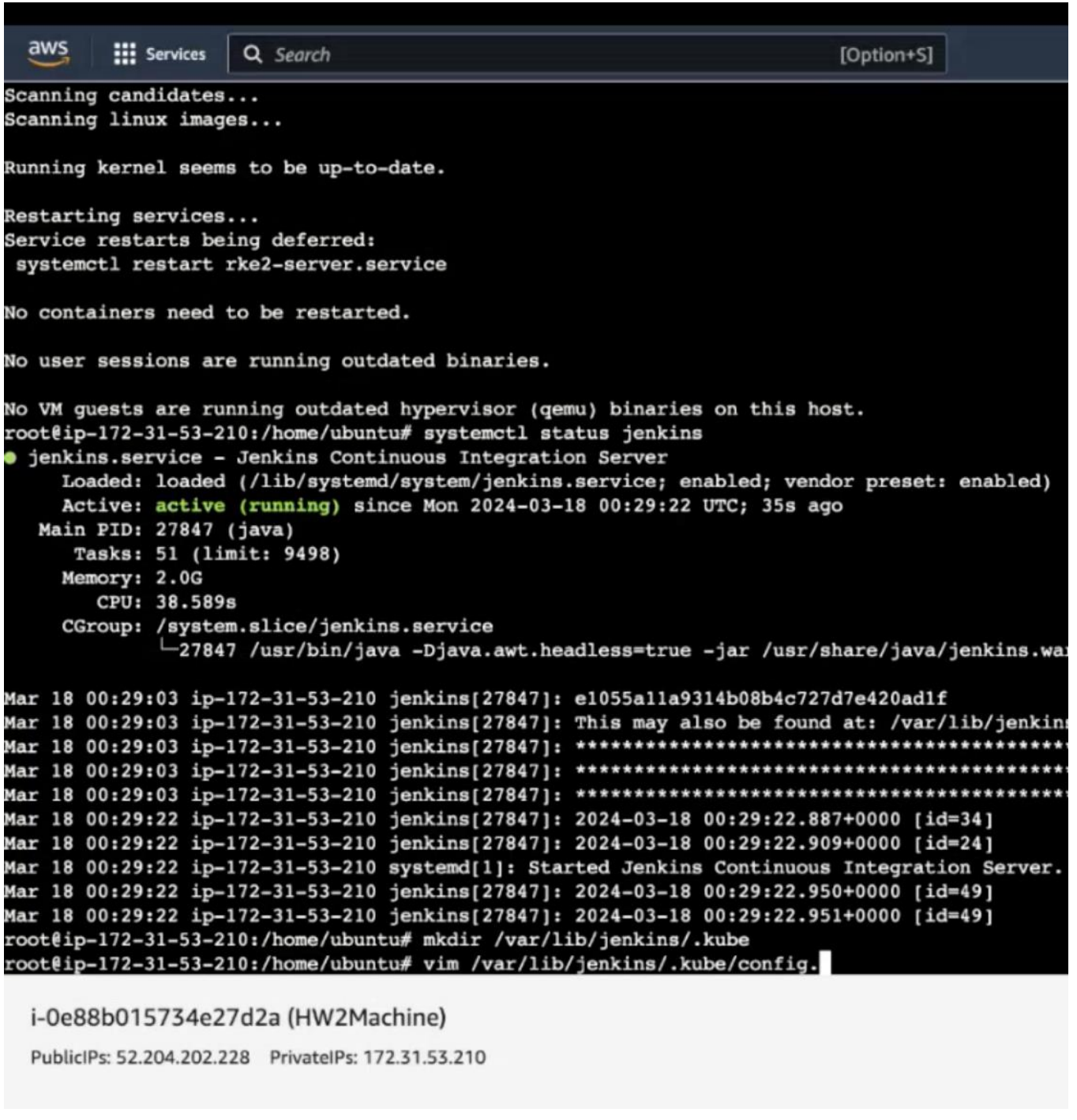
1. Setup the EC2 instance with same steps as in First assignment.
2. Connect the ec2 instance and install docker
   sudo apt get install docker.io
3. Enable docker and start docker
   sudo systemctl start docker
   sudo systemctl enable docker
4. Follow the steps outlined in below link to install jenkins
   https://pkg.jenkins.io/debian-stable/



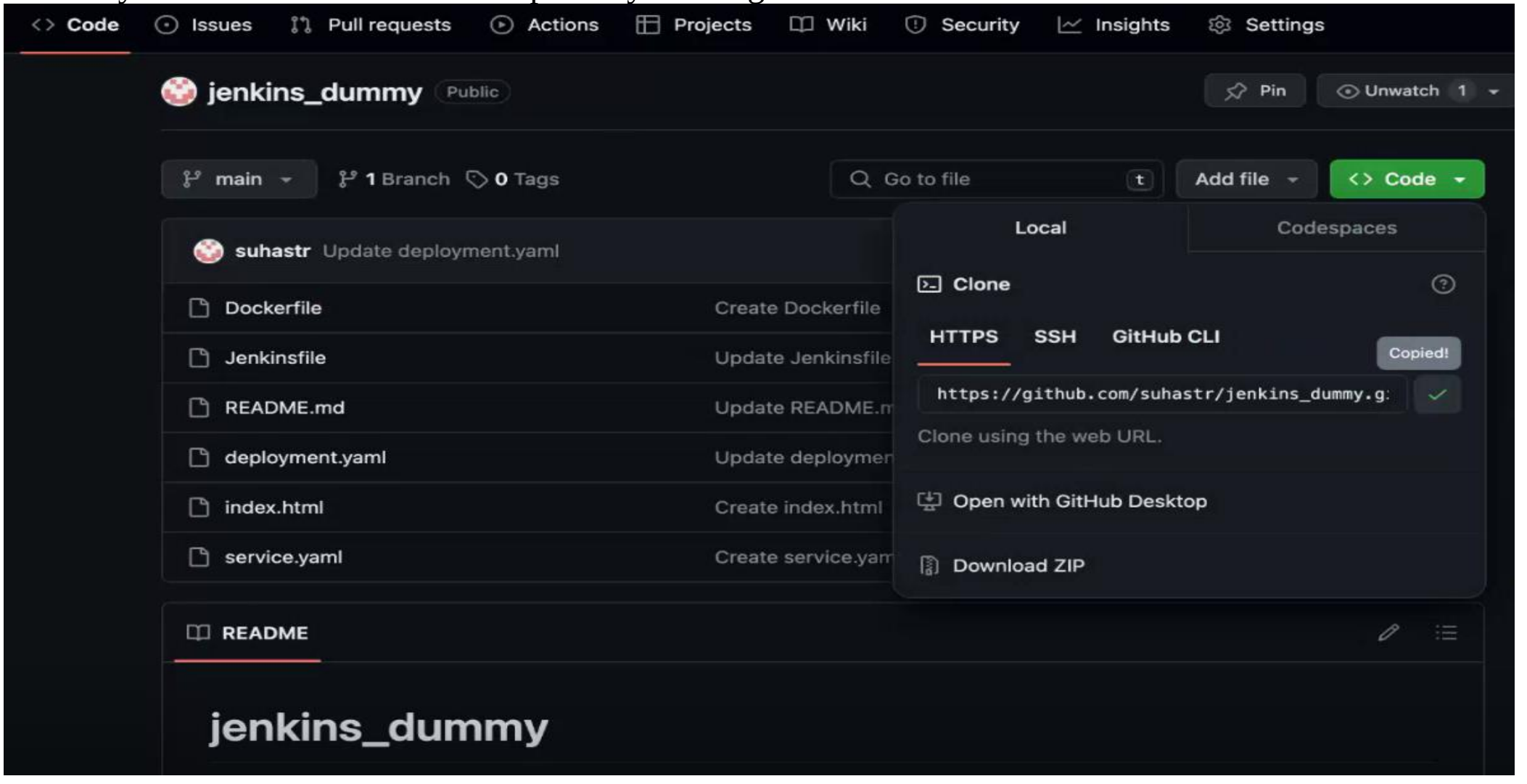
5. Using public ip address like http://54.226.23.128:8080 paste it in browser to access jenkins dashboard.
6. Where u can see it prompts user to enter password, copy that directory link and paste it in the EC2 you willget the password and again paste it in the jenkins page.
7. In EC2 instance you need to create directory and file. Also, copy KubeConfig content into config file of EC2

```
mkdir /var/lib/jenkins/.kube
touch /var/lib/jenkins/.kube/config
```

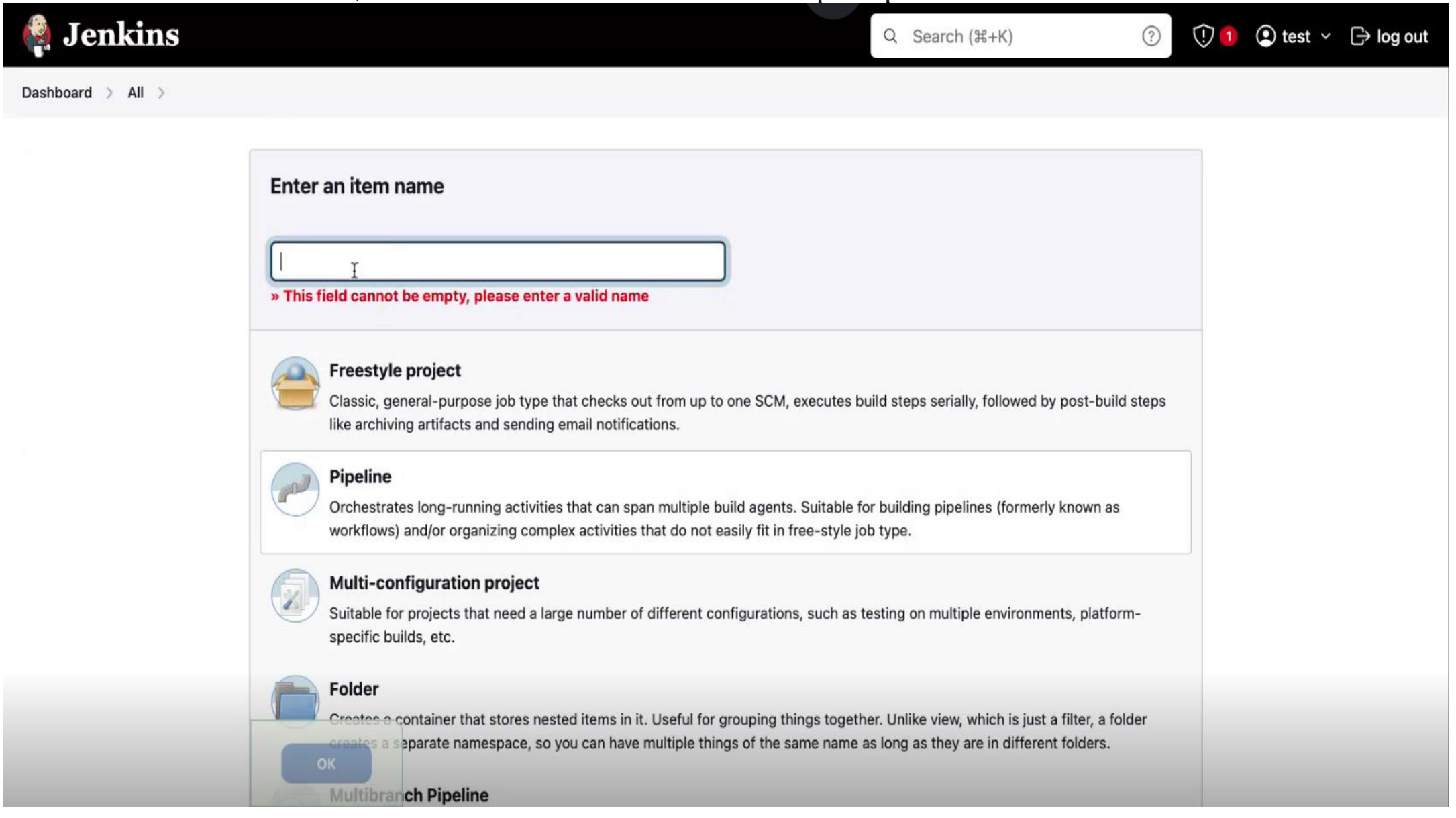

Cluster Management
Clusters 2
Cloud Credentials
Drivers
RKE1 Configuration
Advanced
v2.8.2
Clusters
Import Existing
Create
Download KubeConfig
Take Snapshot
Download YAML
Delete
1 selected
Filter
State
Name
Version
Provider
Machines
Age
Active
local
v1.27.6+k3s1
Local
K3s
1
18 mins
Explore
Active
master
v1.27.11+rke2r1
Custom
RKE2
1
Kubectl Shell
Download KubeConfig
Copy KubeConfig to Clipboard
Take Snapshot
Restore Snapshot
Rotate Certificates
Rotate Encryption Keys
Edit Config
Edit YAML
Download YAML
Delete
Kubectl: master
> kubectl apply -f deployment.yaml
deployment.apps/user-form-deployment created
> kubectl apply -f service.yaml
service/user-form-service created
> kubectl get all
NAME READY STATUS RESTARTS AGE
pod/user-form-deployment-6dd79c7f65-9qvdh 1/1 Running 0 24s
pod/user-form-deployment-6dd79c7f65-gdtn4 1/1 Running 0 24s
pod/user-form-deployment-6dd79c7f65-kwqhq 1/1 Running 0 24s
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
service/kubernetes ClusterIP 10.43.0.1 <none> 443/TCP 8m13s
service/user-form-service NodePort 10.43.132.165 <none> 80:30000/TCP 16s
NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/user-form-deployment 3/3 3 3 24s
NAME DESIRED CURRENT READY AGE
replicaset.apps/user-form-deployment-6dd79c7f65 3 3 3 24s
>


aws
Services
Search
[Option+S]
Scanning candidates...
Scanning linux images...
Running kernel seems to be up-to-date.
Restarting services...
Service restarts being deferred:
systemctl restart rke2-server.service
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-53-210:/home/ubuntu# systemctl status jenkins
jenkins.service - Jenkins Continuous Integration Server
Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2024-03-18 00:29:22 UTC; 35s ago
Main PID: 27847 (java)
Tasks: 51 (limit: 9498)
Memory: 2.0G
CPU: 38.589s
CGroup: /system.slice/jenkins.service
└27847 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.wa
Mar 18 00:29:03 ip-172-31-53-210 jenkins[27847]: e1055a11a9314b08b4c727d7e420ad1f
Mar 18 00:29:03 ip-172-31-53-210 jenkins[27847]: This may also be found at: /var/lib/jenkin
Mar 18 00:29:03 ip-172-31-53-210 jenkins[27847]: **********
Mar 18 00:29:03 ip-172-31-53-210 jenkins[27847]: **********
Mar 18 00:29:03 ip-172-31-53-210 jenkins[27847]: **********
Mar 18 00:29:22 ip-172-31-53-210 jenkins[27847]: 2024-03-18 00:29:22.887+0000 [id=34]
Mar 18 00:29:22 ip-172-31-53-210 jenkins[27847]: 2024-03-18 00:29:22.909+0000 [id=24]
Mar 18 00:29:22 ip-172-31-53-210 systemd[1]: Started Jenkins Continuous Integration Server.
Mar 18 00:29:22 ip-172-31-53-210 jenkins[27847]: 2024-03-18 00:29:22.950+0000 [id=49]
Mar 18 00:29:22 ip-172-31-53-210 jenkins[27847]: 2024-03-18 00:29:22.951+0000 [id=49]
root@ip-172-31-53-210:/home/ubuntu# mkdir /var/lib/jenkins/.kube
root@ip-172-31-53-210:/home/ubuntu# vim /var/lib/jenkins/.kube/config.
i-0e88b015734e27d2a (HW2Machine)
PublicIPs: 52.204.202.228 PrivateIPs: 172.31.53.210

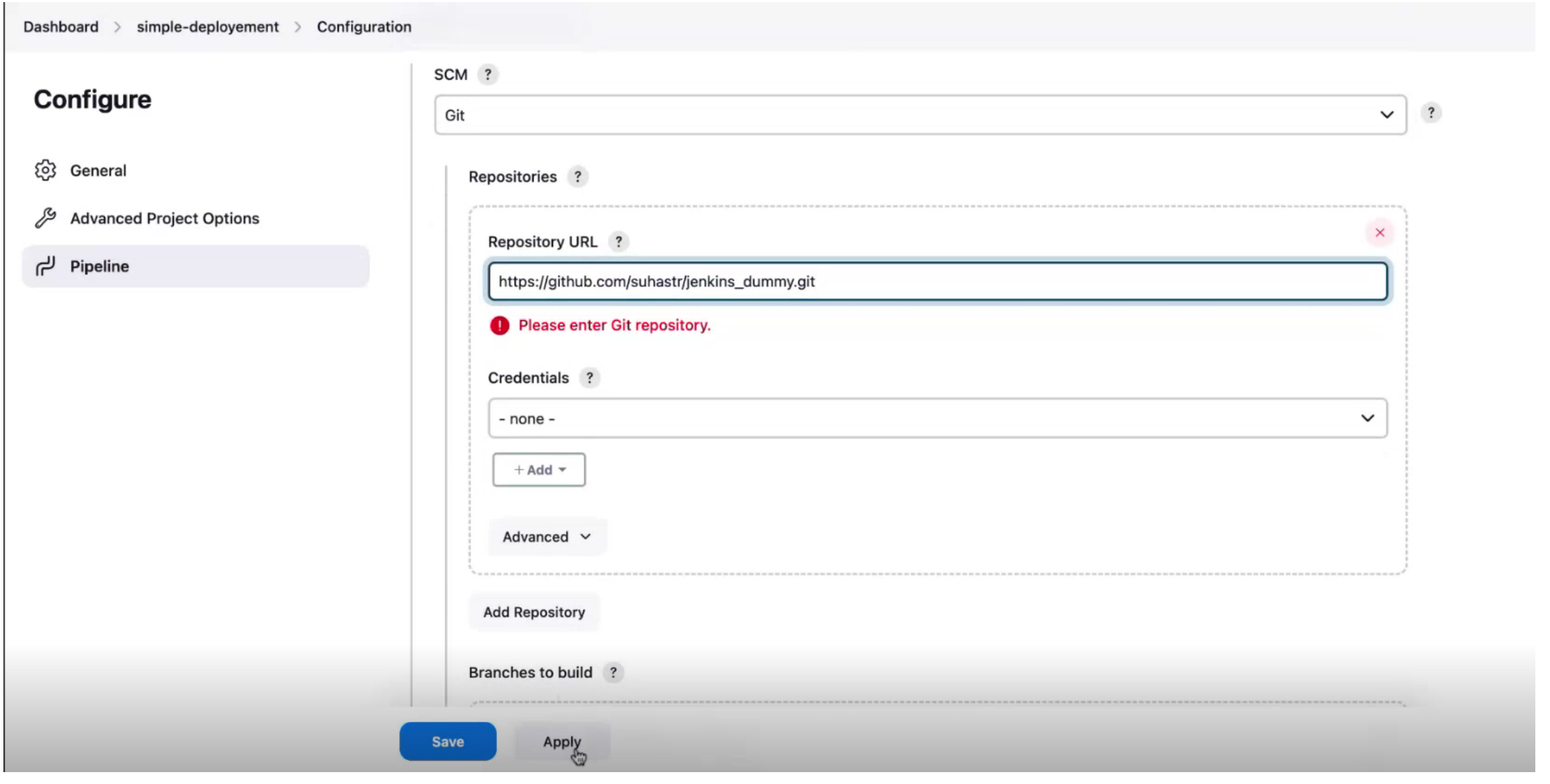
8. Now you have to create a Public repository on the github which looks like the below:



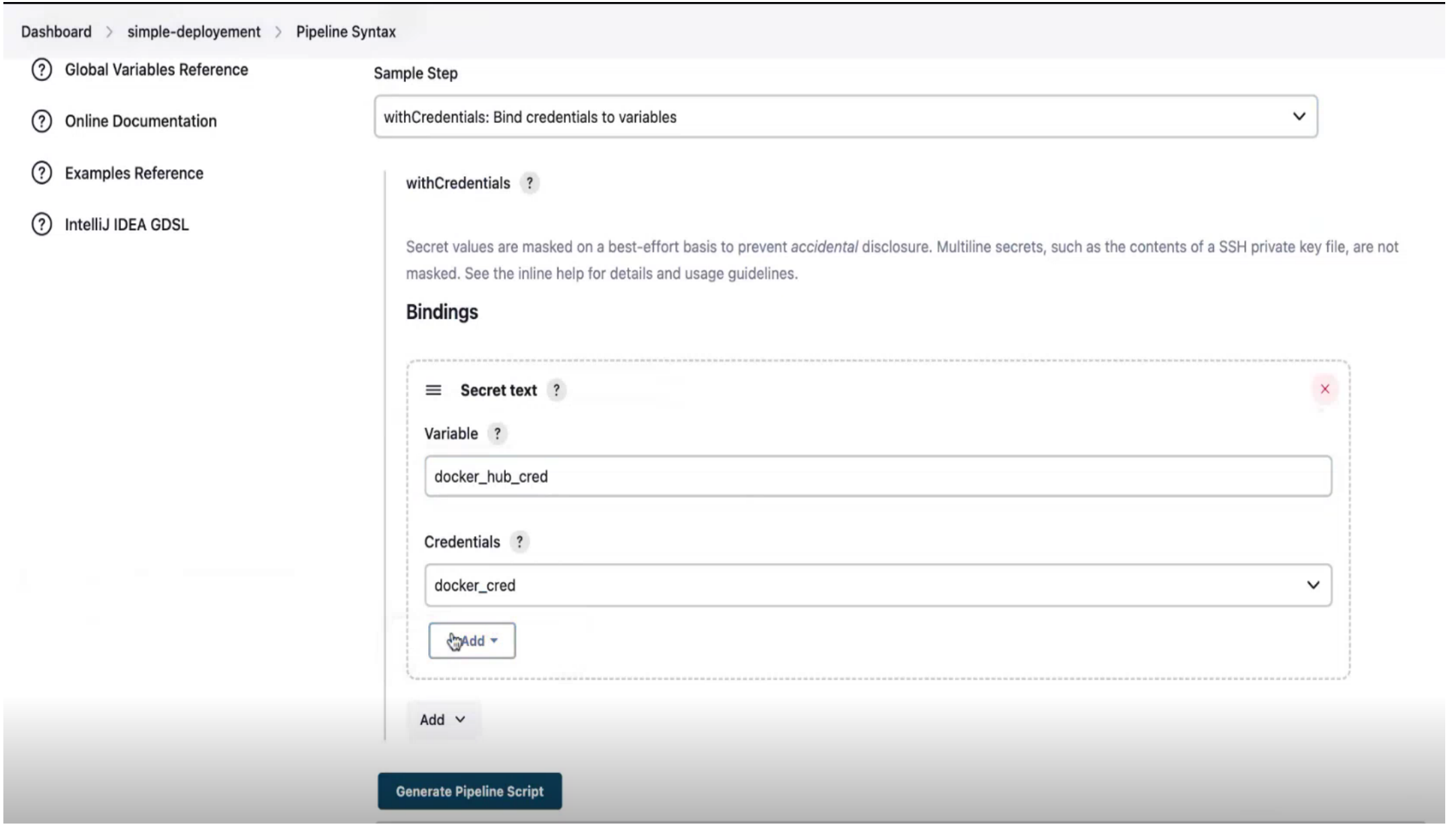
9. From Jenkins dashboard, click on “New item” and then this prompt will show:



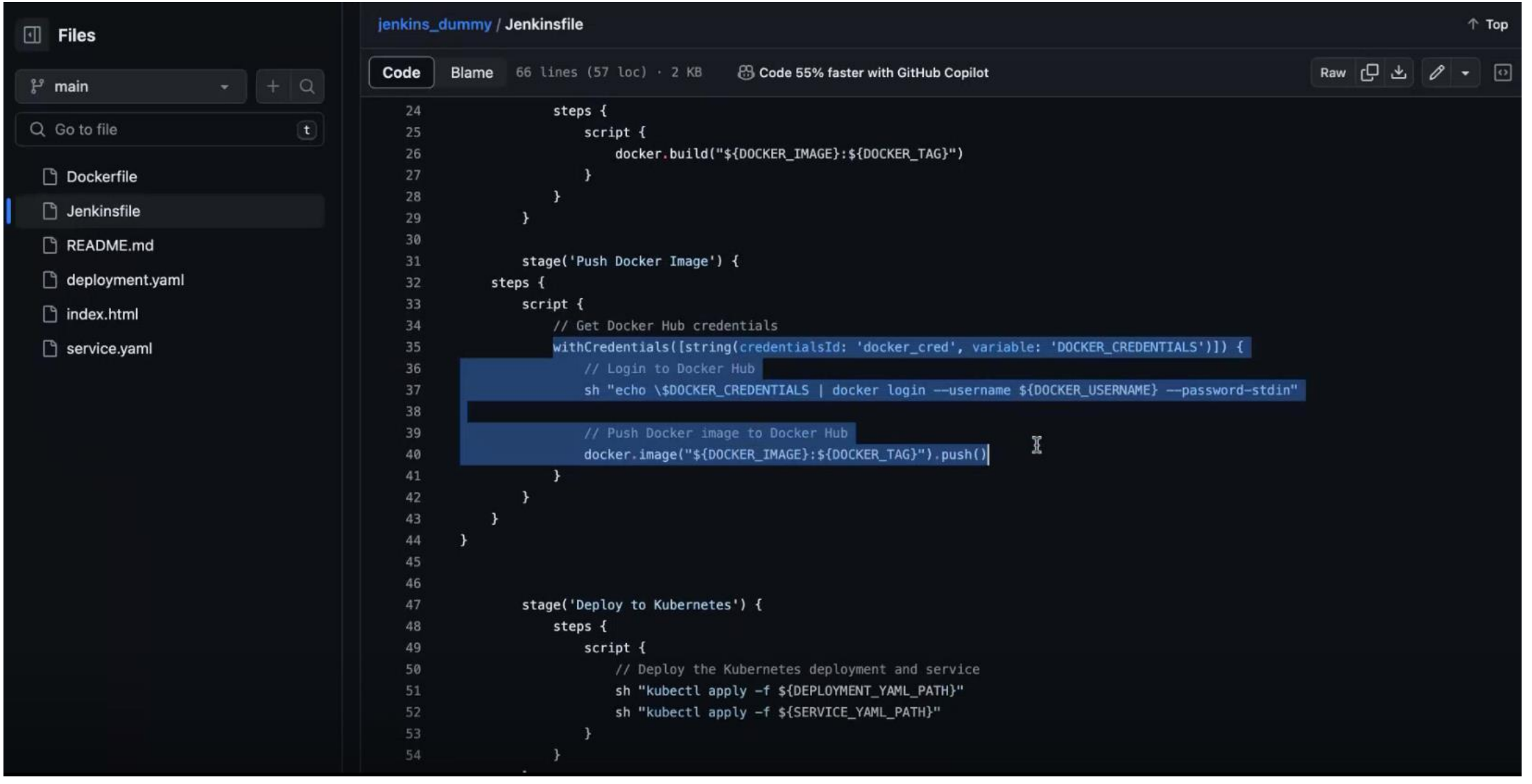
10. Enter Git Repository link in pipeline configuration. Also mention branch name



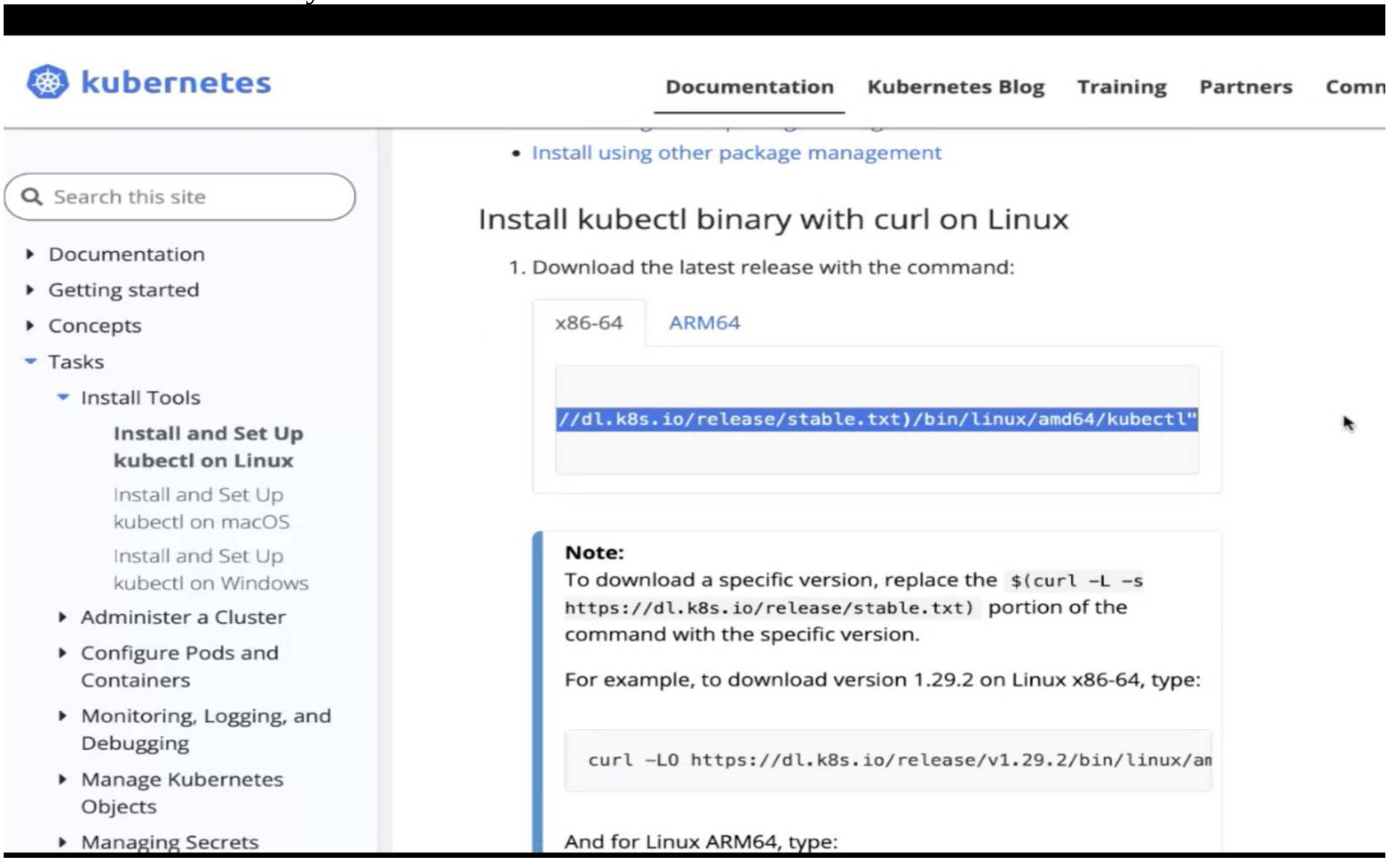
11. Go to Pipeline Syntax and generate Secret text



12. Copy the generated script into Jenkinsfile as shown below:



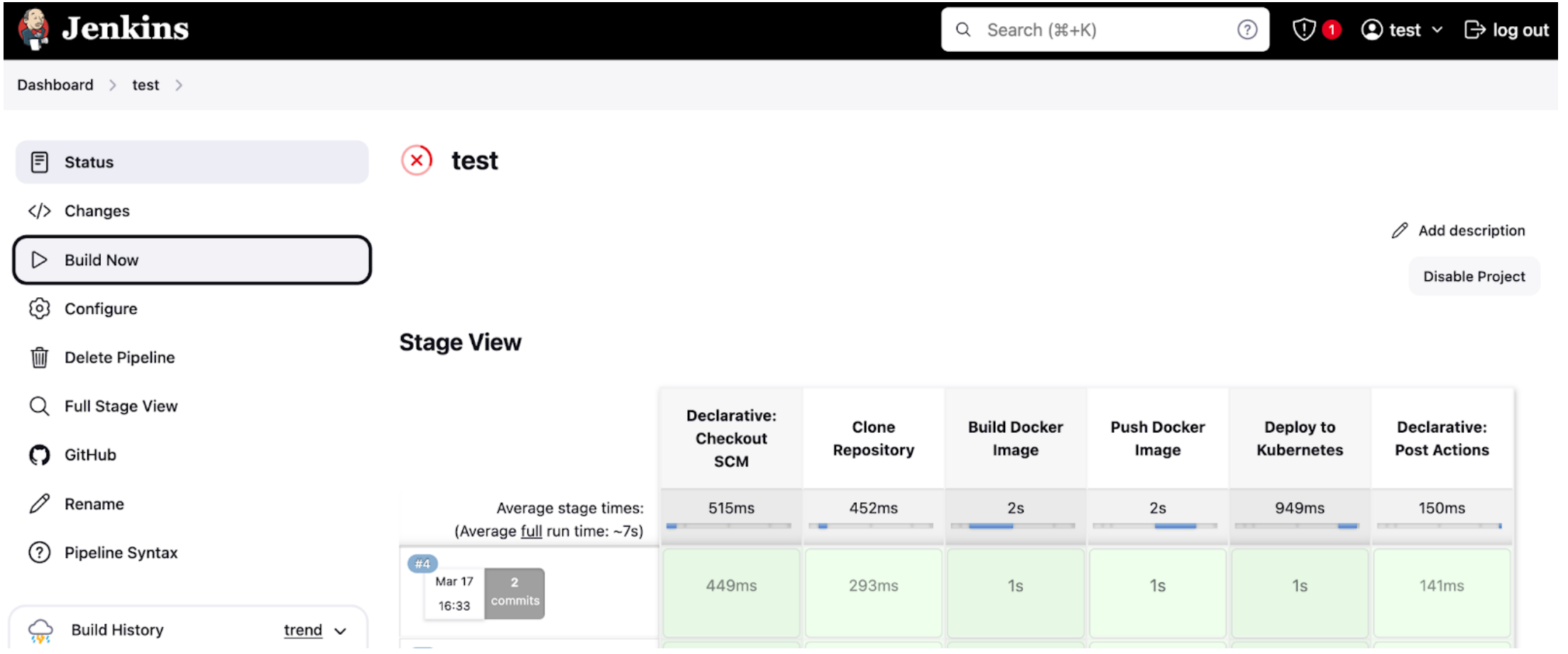
13. Install Kubectl binary with curl on linux



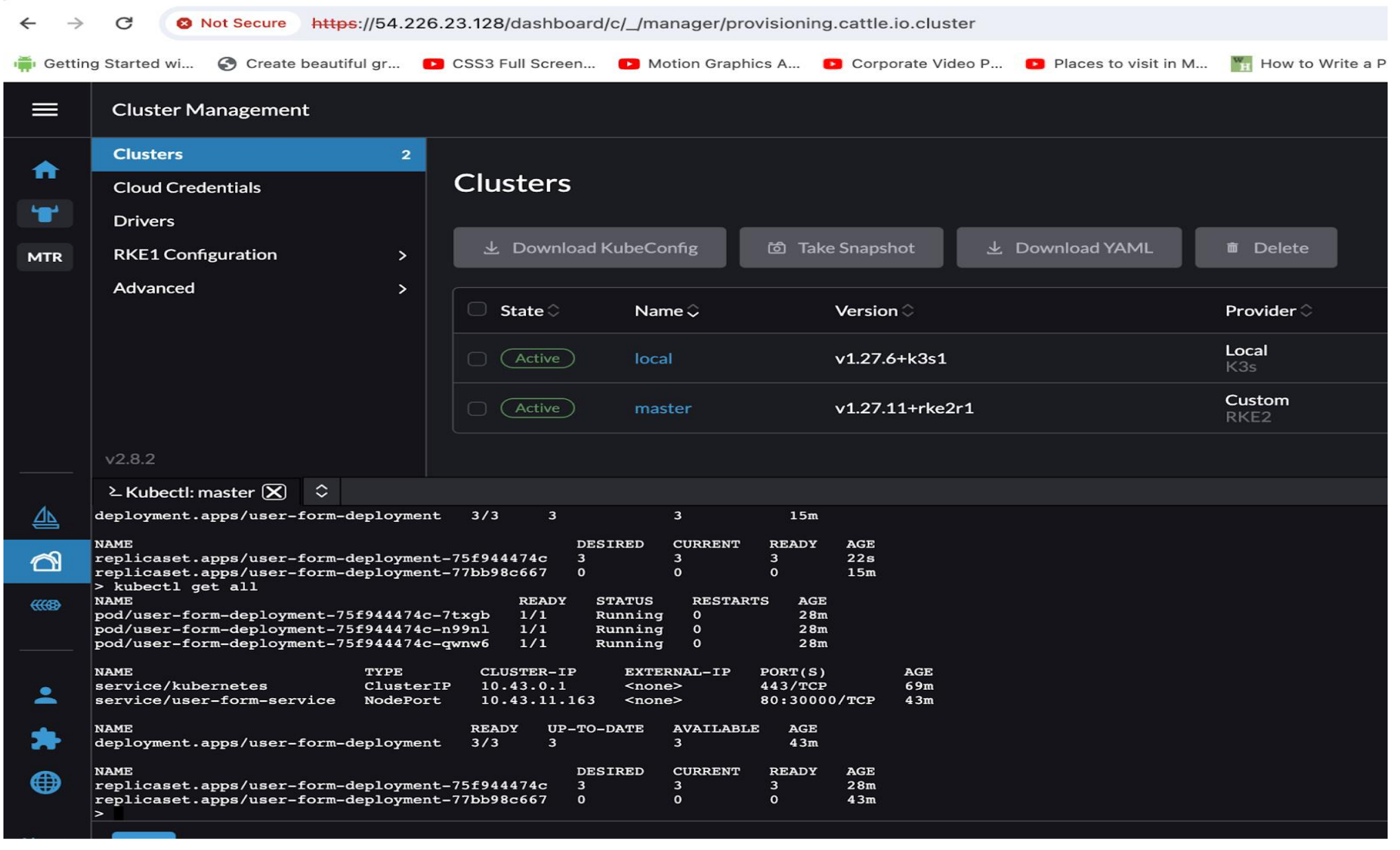
14. Now go to your ec2 instance command line, type the below command to give permission to jenkins

    sudo usermod -aG docker jenkins

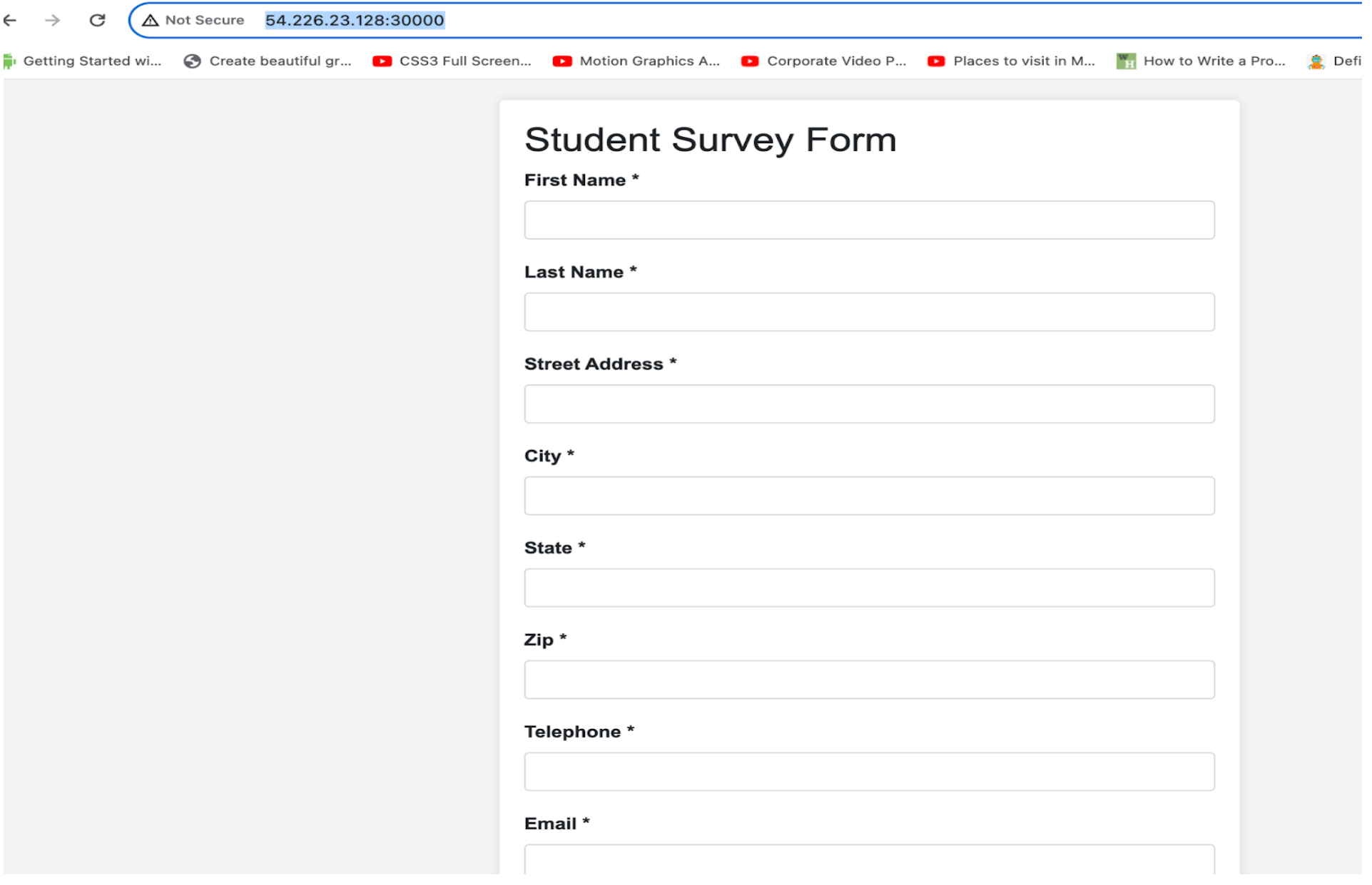
15. In the Jenkins dashboard select build now option and Jenkins will start building the processes stage by stage



16. At last you can see the deployment on your rancher like shown below:



17. Now access the form using this Form



References:

https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/#install-kubectl-binary-with-curl-on-linux
https://pkg.jenkins.io/debian-stable/

Contributions:

Aditya Ketanbhai Shah – G01449892 : 25%
Suhas Thandaga Ramesh – G01450932 : 25%
Niharika FNU LNU – G01412073 : 25%
Syedfurkhan Feroz Hashmi – G01411748 : 25%