# JENKINS END TO END CICD Implementation

### Objective

To implement an automated CI/CD pipeline using Jenkins that builds, tests, analyzes, packages, containerizes, and deploys applications to Kubernetes clusters using ArgoCD

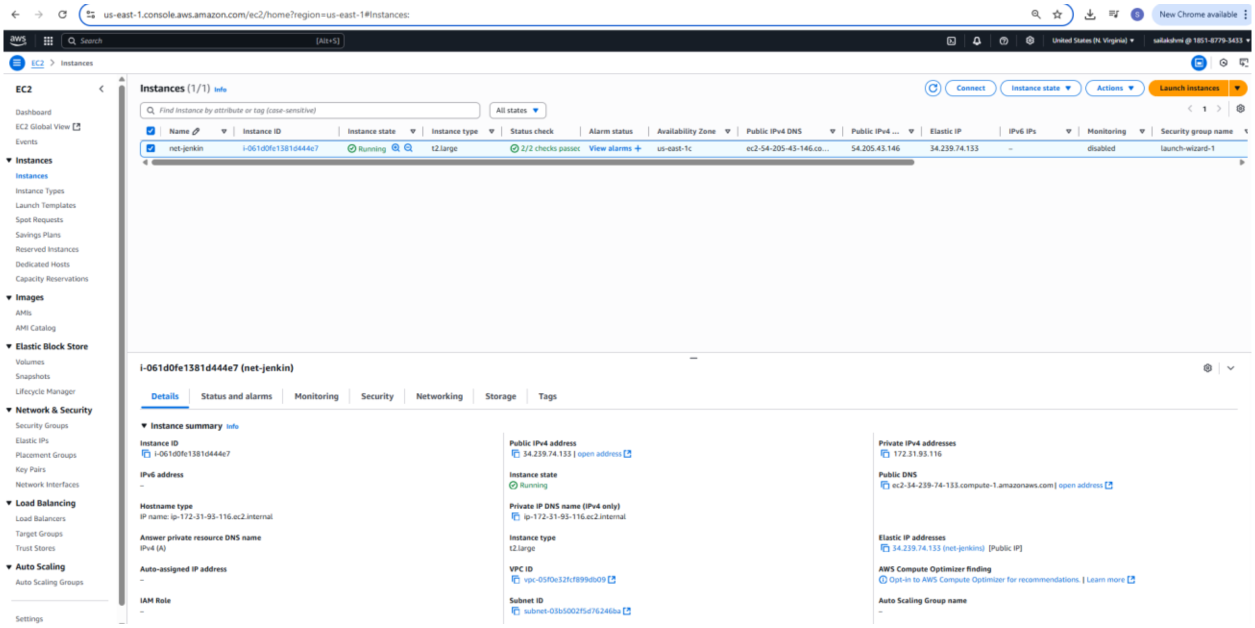
### Tools & Technologies Used

| | |
|---|---|
| Source Control | Git, GitHub |
| CI/CD | Jenkins |
| Build Tool | Maven |
| Code Quality | SonarQube |
| Container | Docker |
| Artifact Registry | DockerHub |
| Deployment | Kubernetes, ArgoCD |

### Steps to Implement

**Step 1: Launch EC2 (Ubuntu) and Install Java & Jenkins**

- Provision an EC2 instance on AWS with Ubuntu.
- Connect to the instance.



| ☑ | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS | Public IPv4 ... | Elastic IP | IPv6 IPs | Monitoring | Security group name |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ☑ | net-jenkin | i-061d0fe1381d444e7 | Running | t2.large | 2/2 checks passec | View alarms + | us-east-1c | ec2-54-205-43-146.co... | 54.205.43.146 | 34.239.74.133 | – | disabled | launch-wizard-1 |

i-061d0fe1381d444e7 (net-jenkin)

▾ Instance summary Info

Instance ID
i-061d0fe1381d444e7

Public IPv4 address
34.239.74.133 | open address

Private IPv4 addresses
172.31.93.116

IPv6 address
–

Instance state
Running

Public DNS
ec2-34-239-74-133.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-172-31-93-116.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-93-116.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.large

Elastic IP addresses
34.239.74.133 (net-jenkins) [Public IP]

Auto-assigned IP address
–

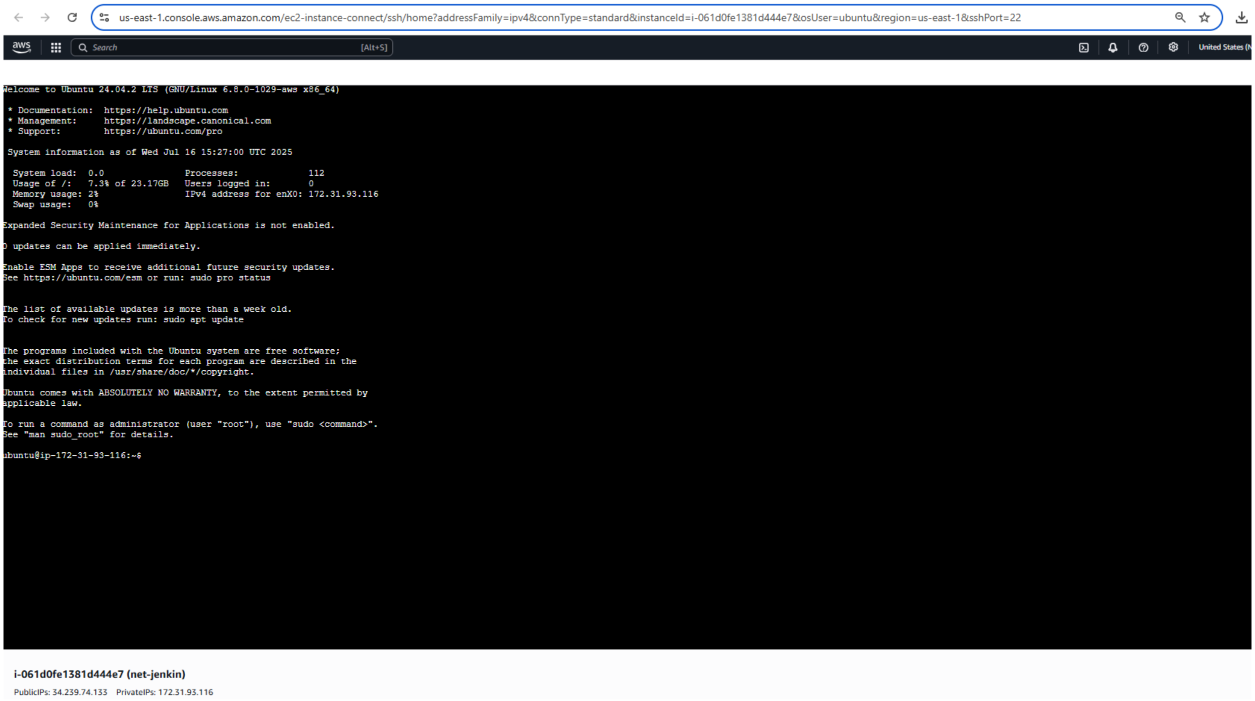
VPC ID
vpc-05f0e32fcf899db09

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

IAM Role
–

Subnet ID
subnet-03b5002f5d76246ba

Auto Scaling Group name
–


us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?addressFamily=ipv4&connType=standard&instanceId=i-061d0fe1381d444e7&osUser=ubuntu&region=us-east-1&sshPort=22
Search [Alt+S]
United States (N

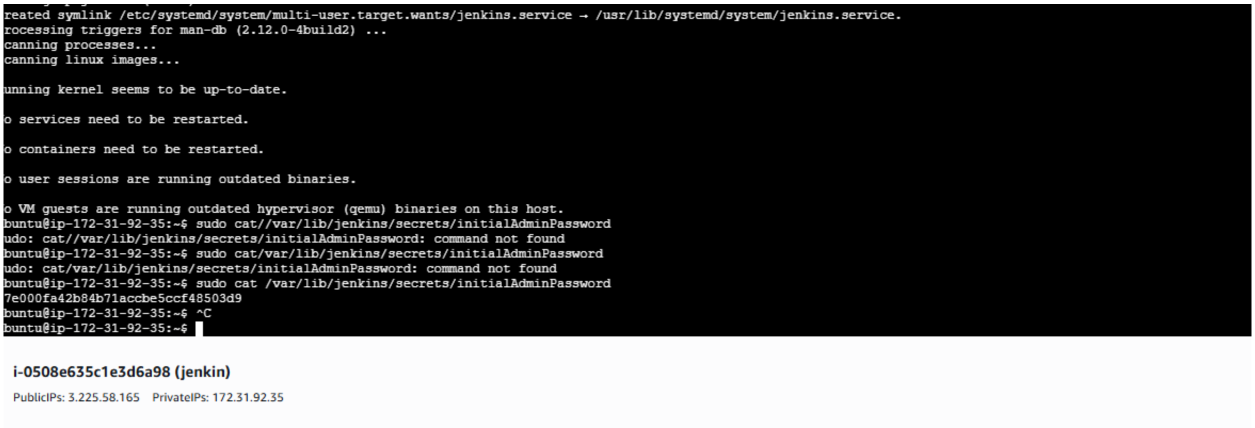
```
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Wed Jul 16 15:27:00 UTC 2025

  System load:  0.0               Processes:             112
  Usage of /:   7.3% of 23.17GB   Users logged in:       0
  Memory usage: 2%                IPv4 address for enX0: 172.31.93.116
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-93-116:~$
```

i-061d0fe1381d444e7 (net-jenkin)

PublicIPs: 34.239.74.133    PrivateIPs: 172.31.93.116

## Step 2: install Jenkins

**http://3.225.58.165:8080/**

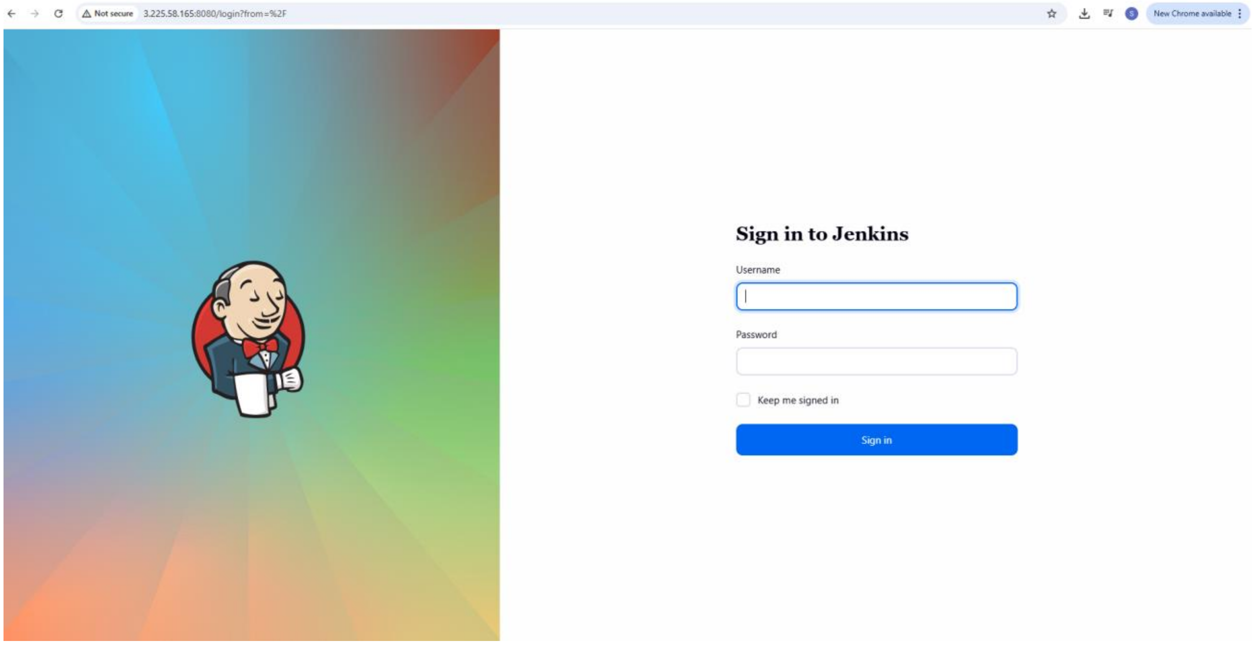


```
reated symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
rocessing triggers for man-db (2.12.0-4build2) ...
canning processes...
canning linux images...

unning kernel seems to be up-to-date.

o services need to be restarted.

o containers need to be restarted.

o user sessions are running outdated binaries.

o VM guests are running outdated hypervisor (qemu) binaries on this host.
buntu@ip-172-31-92-35:~$ sudo cat//var/lib/jenkins/secrets/initialAdminPassword
udo: cat//var/lib/jenkins/secrets/initialAdminPassword: command not found
buntu@ip-172-31-92-35:~$ sudo cat/var/lib/jenkins/secrets/initialAdminPassword
udo: cat/var/lib/jenkins/secrets/initialAdminPassword: command not found
buntu@ip-172-31-92-35:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
7e000fa42b84b71accbe5ccf48503d9
buntu@ip-172-31-92-35:~$ ^C
buntu@ip-172-31-92-35:~$
```

i-0508e635c1e3d6a98 (jenkin)

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

Access Jenkins



Sign in to Jenkins

Username

Password

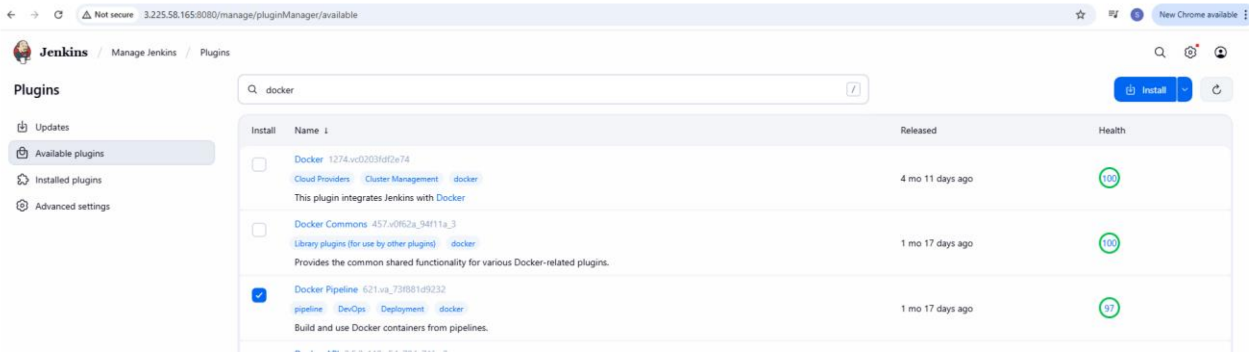
□ Keep me signed in

Sign in

**Step: 3 Install Required Jenkins Plugins**

Install the docker pipeline plugin

Go to manage jenkins plugins available plugins search for docker pipeline and install


⚠ Not secure 3.225.58.165:8080/manage/pluginManager/available

Jenkins / Manage Jenkins / Plugins

Plugins

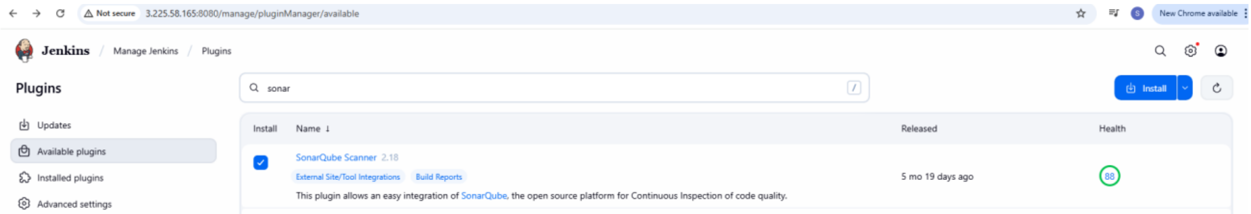
- Updates
- Available plugins
- Installed plugins
- Advanced settings

docker

Install

| Install | Name ↓ | Released | Health |
|---|---|---|---|
| □ | Docker 1274.vc0203fdf2e74<br>Cloud Providers · Cluster Management · docker<br>This plugin integrates Jenkins with Docker | 4 mo 11 days ago | 100 |
| □ | Docker Commons 457.v0f62a_94f11a_3<br>Library plugins (for use by other plugins) · docker<br>Provides the common shared functionality for various Docker-related plugins. | 1 mo 17 days ago | 100 |
| ☑ | Docker Pipeline 621.va_73f881d9232<br>pipeline · DevOps · Deployment · docker<br>Build and use Docker containers from pipelines. | 1 mo 17 days ago | 97 |

Install the sonar plugin in jenkins


⚠ Not secure 3.225.58.165:8080/manage/pluginManager/available

Jenkins / Manage Jenkins / Plugins

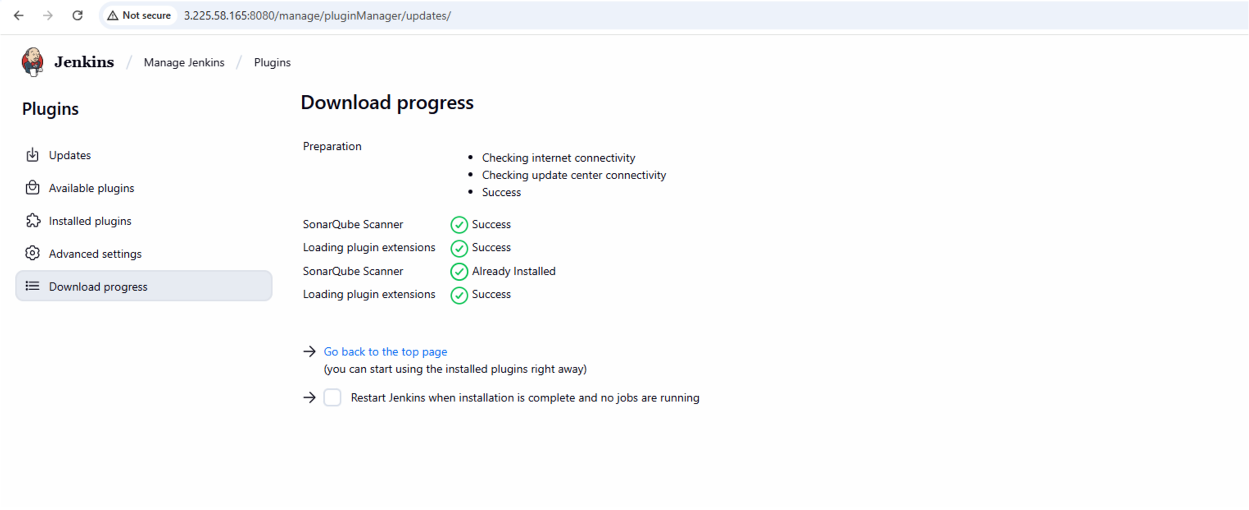
Plugins

- Updates
- Available plugins
- Installed plugins
- Advanced settings

sonar

Install

| Install | Name ↓ | Released | Health |
|---|---|---|---|
| ☑ | SonarQube Scanner 2.18<br>External Site/Tool Integrations · Build Reports<br>This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality. | 5 mo 19 days ago | 88 |


⚠ Not secure 3.225.58.165:8080/manage/pluginManager/updates/

Jenkins / Manage Jenkins / Plugins

Plugins

- Updates
- Available plugins
- Installed plugins
- Advanced settings
- Download progress

Download progress

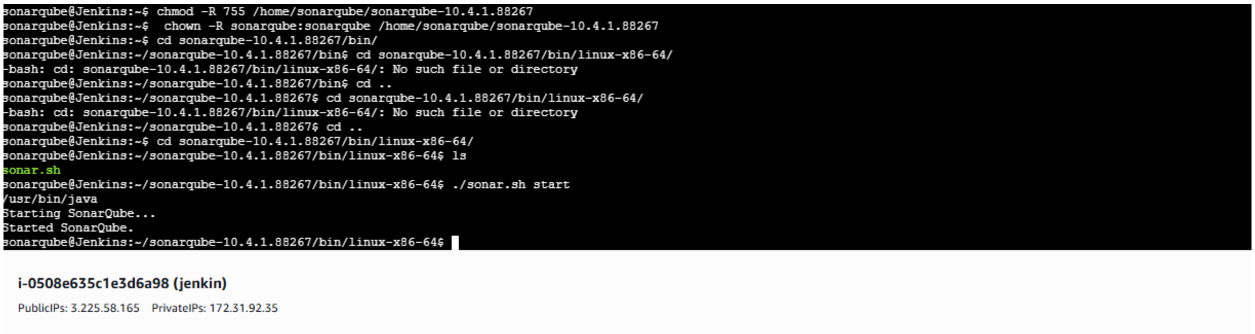
Preparation
- Checking internet connectivity
- Checking update center connectivity
- Success

| | |
|---|---|
| SonarQube Scanner | ✓ Success |
| Loading plugin extensions | ✓ Success |
| SonarQube Scanner | ✓ Already Installed |
| Loading plugin extensions | ✓ Success |

→ Go back to the top page
(you can start using the installed plugins right away)

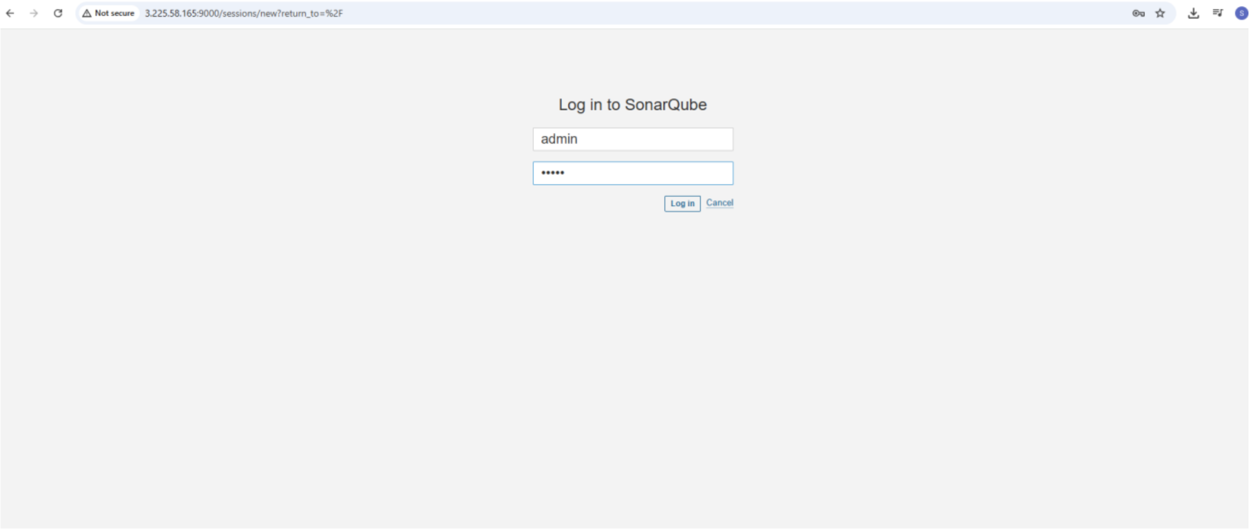
→ □ Restart Jenkins when installation is complete and no jobs are running

## Step: 4 Install sonar in ec2


```
sonarqube@Jenkins:~$ chmod -R 755 /home/sonarqube/sonarqube-10.4.1.88267
sonarqube@Jenkins:~$  chown -R sonarqube:sonarqube /home/sonarqube/sonarqube-10.4.1.88267
sonarqube@Jenkins:~$ cd sonarqube-10.4.1.88267/bin/
sonarqube@Jenkins:~/sonarqube-10.4.1.88267/bin$ cd sonarqube-10.4.1.88267/bin/linux-x86-64/
-bash: cd: sonarqube-10.4.1.88267/bin/linux-x86-64/: No such file or directory
sonarqube@Jenkins:~/sonarqube-10.4.1.88267/bin$ cd ..
sonarqube@Jenkins:~/sonarqube-10.4.1.88267$ cd sonarqube-10.4.1.88267/bin/linux-x86-64/
-bash: cd: sonarqube-10.4.1.88267/bin/linux-x86-64/: No such file or directory
sonarqube@Jenkins:~/sonarqube-10.4.1.88267$ cd ..
sonarqube@Jenkins:~$ cd sonarqube-10.4.1.88267/bin/linux-x86-64/
sonarqube@Jenkins:~/sonarqube-10.4.1.88267/bin/linux-x86-64$ ls
sonar.sh
sonarqube@Jenkins:~/sonarqube-10.4.1.88267/bin/linux-x86-64$ ./sonar.sh start
/usr/bin/java
Starting SonarQube...
Started SonarQube.
sonarqube@Jenkins:~/sonarqube-10.4.1.88267/bin/linux-x86-64$
```

i-0508e635c1e3d6a98 (jenkin)

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

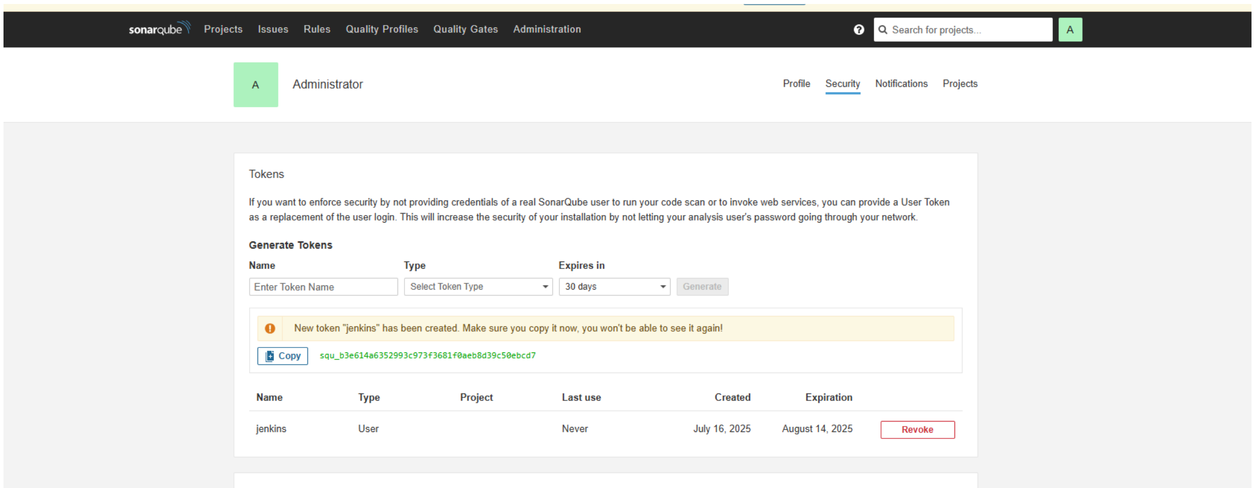


Log in to SonarQube

admin

•••••

To authineticate jenkins with sonar we need to generate a token in sonar and add it in jenkin



Administrator

Tokens

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

Generate Tokens

Name: Enter Token Name

Type: Select Token Type

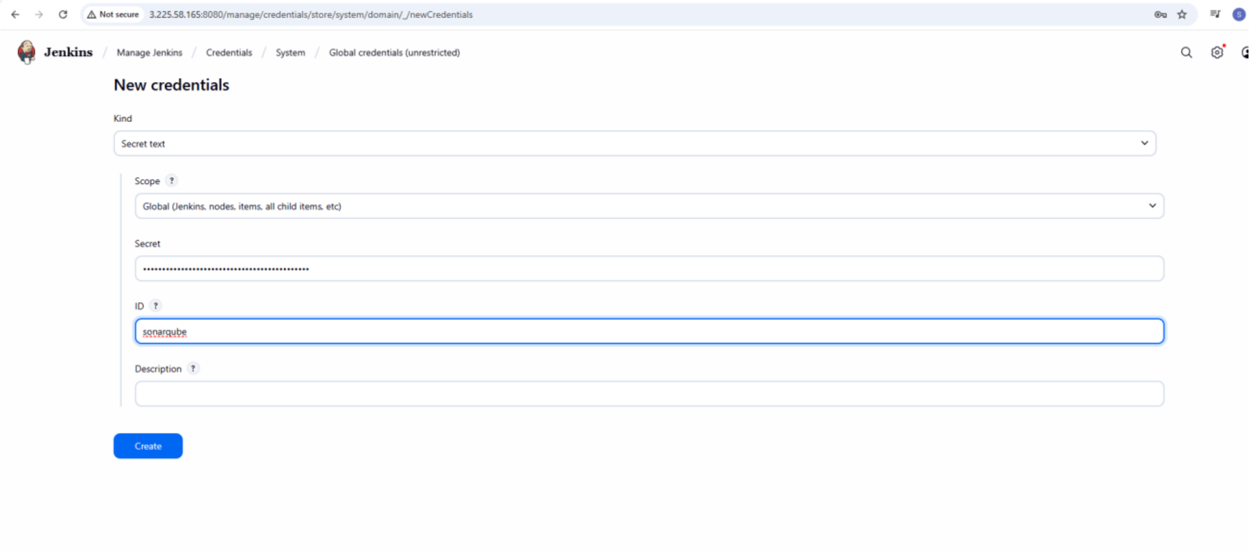
Expires in: 30 days

New token "jenkins" has been created. Make sure you copy it now, you won't be able to see it again!

```
squ_b3e614a6352993c973f3681f0aeb8d39c50ebcd7
```

| Name | Type | Project | Last use | Created | Expiration | |
|---|---|---|---|---|---|---|
| jenkins | User | | Never | July 16, 2025 | August 14, 2025 | Revoke |

Manage Jenkins → Credentials → Global:



Not secure
3.225.58.165:8080/manage/credentials/store/system/domain/_/newCredentials
Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)

New credentials

Kind: Secret text

Scope: Global (Jenkins, nodes, items, all child items, etc)

Secret: ••••••••••••••••••••••••••••••••••••••••••••••

ID: sonarqube

Description:

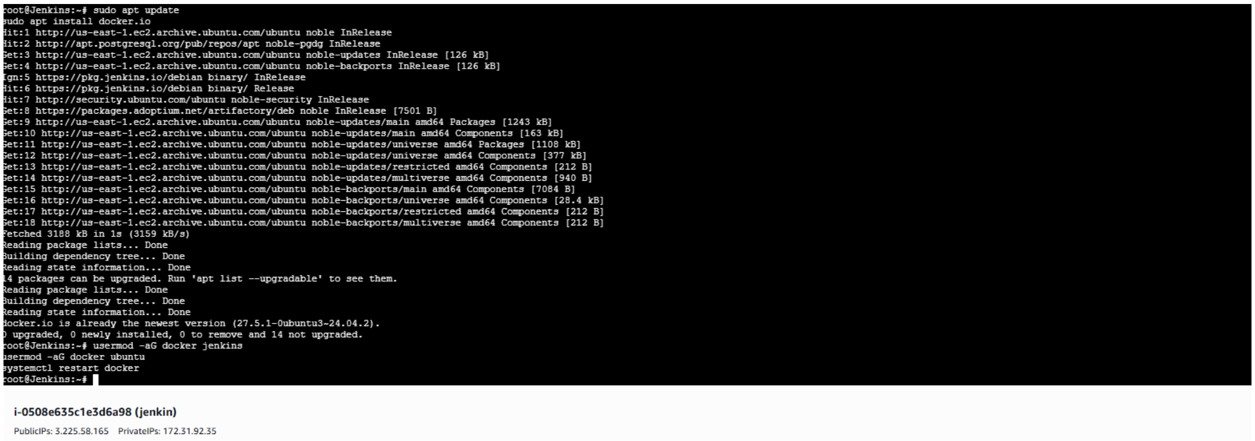
## Step: 5 Install docker in ec2

Run the below command to Install Docker

sudo apt update
sudo apt install docker.io

Grant Jenkins user and Ubuntu user permission to docker deamon.

sudo su -
usermod -aG docker jenkins
usermod -aG docker ubuntu
systemctl restart docker



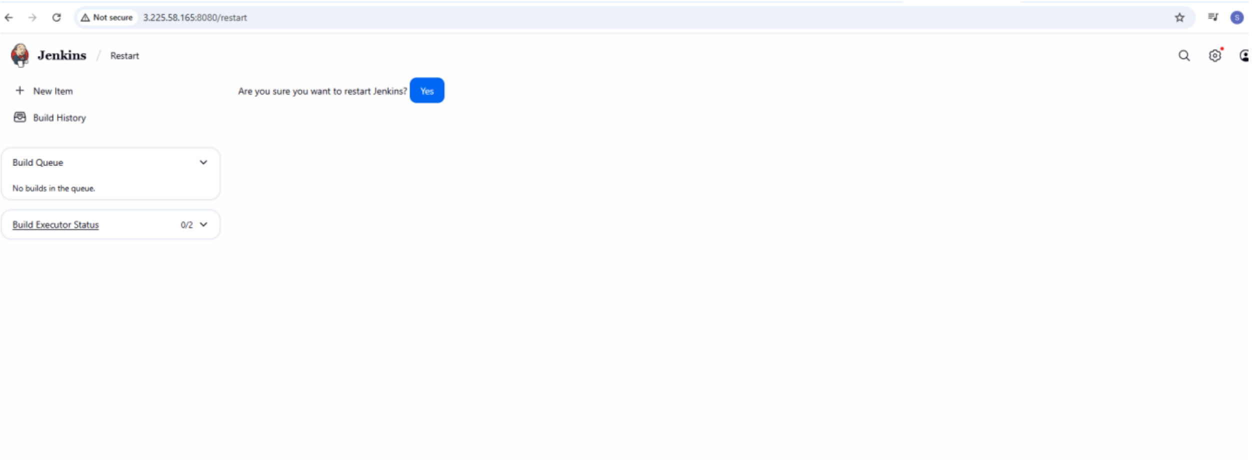
```
root@Jenkins:~# sudo apt update
sudo apt install docker.io
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://apt.postgresql.org/pub/repos/apt noble-pgdg InRelease
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Ign:5 https://pkg.jenkins.io/debian binary/ InRelease
Hit:6 https://pkg.jenkins.io/debian binary/ Release
Hit:7 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:8 https://packages.adoptium.net/artifactory/deb noble InRelease [7501 B]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1243 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [163 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1108 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [377 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7084 B]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [28.4 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Fetched 3188 kB in 1s (3159 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
14 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (27.5.1-0ubuntu3~24.04.2).
0 upgraded, 0 newly installed, 0 to remove and 14 not upgraded.
root@Jenkins:~# usermod -aG docker jenkins
usermod -aG docker ubuntu
systemctl restart docker
root@Jenkins:~# 
```

i-0508e635c1e3d6a98 (jenkin)

PublicIPs: 3.225.58.165 PrivateIPs: 172.31.92.35

Once you are done with the above steps, it is better to restart Jenkins.

http://<ec2-instance-public-ip>:8080/restart

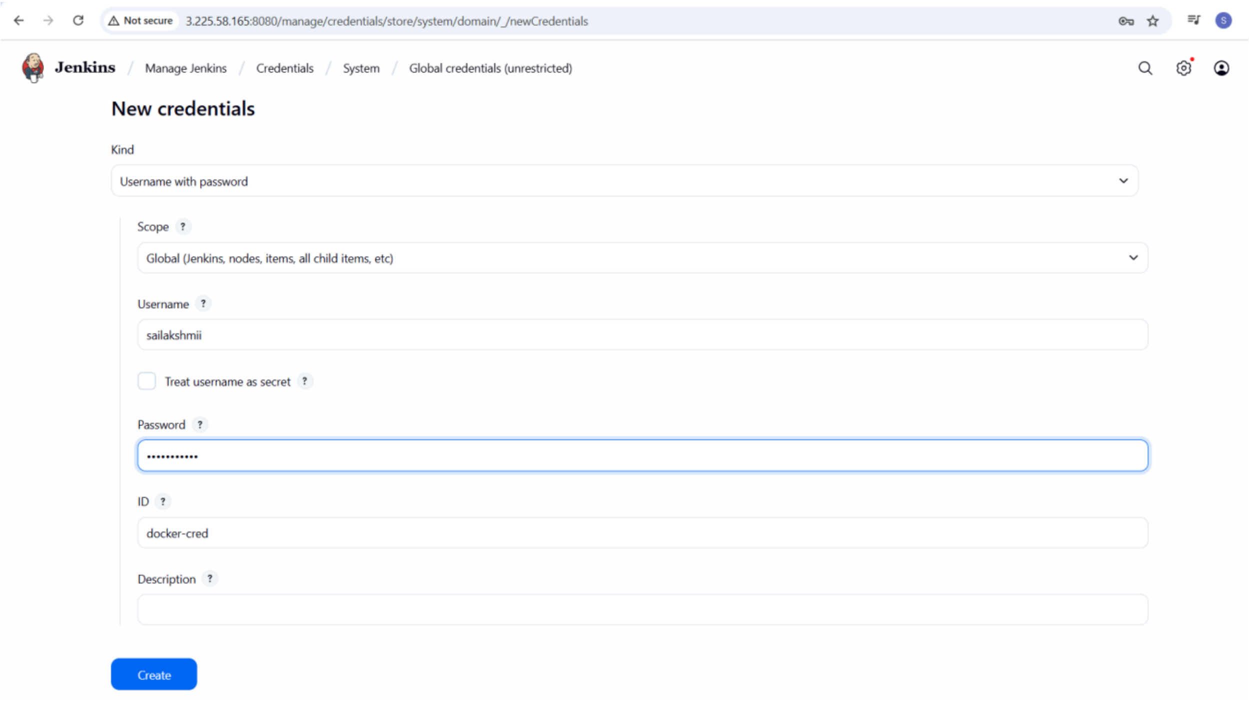


Not secure 3.225.58.165:8080/restart

Jenkins / Restart

Are you sure you want to restart Jenkins? Yes

Go to jenkins and add docker hub credentials



Not secure 3.225.58.165:8080/manage/credentials/store/system/domain/_/newCredentials

Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)

New credentials

Kind: Username with password

Scope: Global (Jenkins, nodes, items, all child items, etc)

Username: sailakshmii

□ Treat username as secret

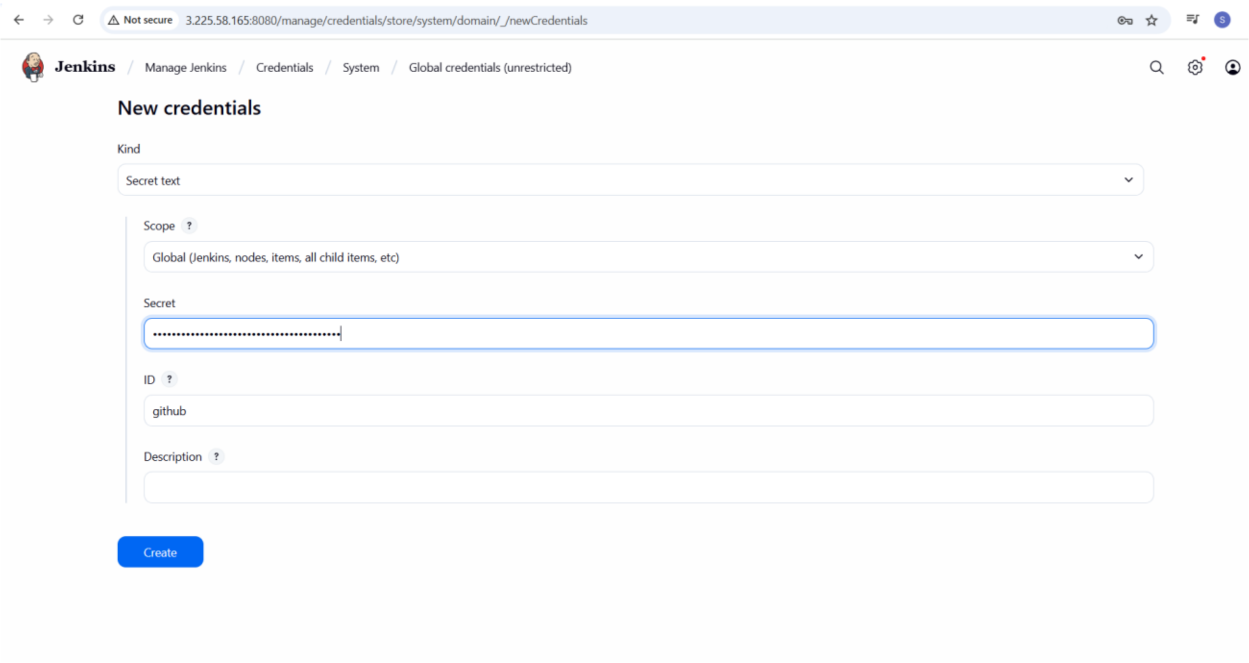
Password: •••••••••••

ID: docker-cred

Description:

Create

Now add github credentials in jenkins



3.225.58.165:8080/manage/credentials/store/system/domain/_/newCredentials

Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)

New credentials

Kind: Secret text

Scope: Global (Jenkins, nodes, items, all child items, etc)

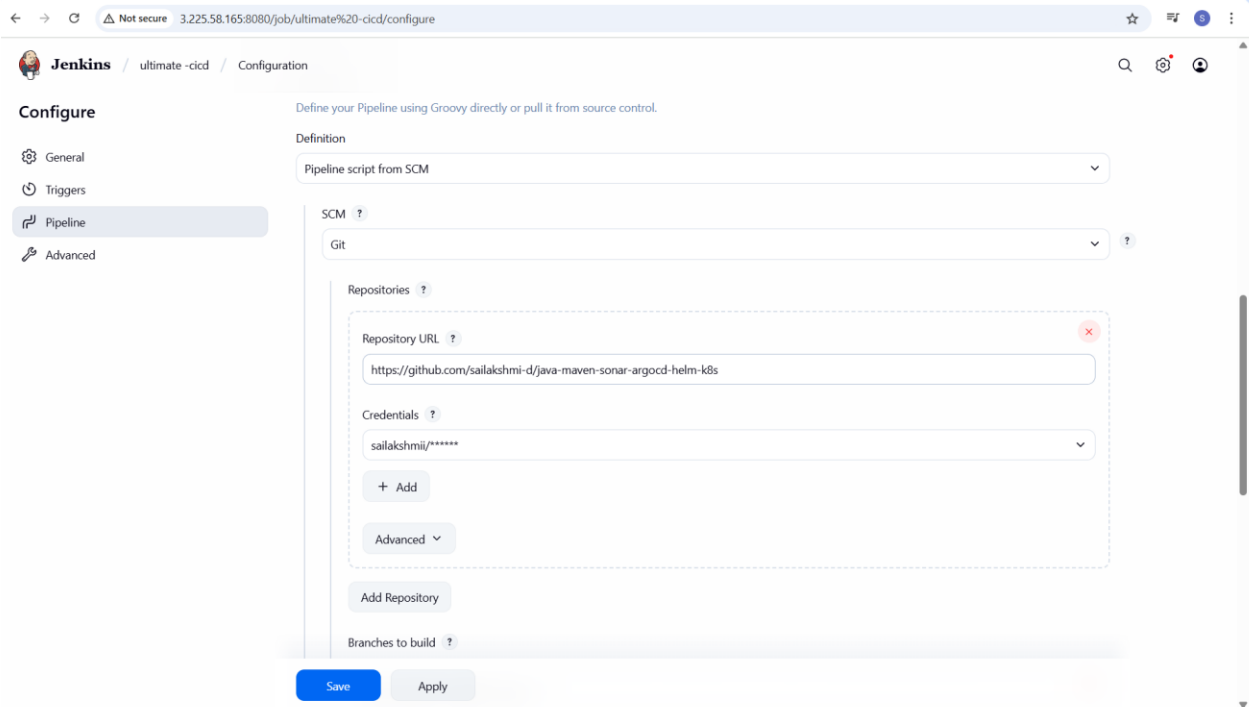
Secret: ••••••••••••••••••••••••••••••••••••••••

ID: github

Description:

## Step :6 Create a pipeline with pipeline script from ssm

Make sure you pass correct jenkins file path, github credentials



3.225.58.165:8080/job/ultimate%20-cicd/configure

Jenkins / ultimate -cicd / Configuration

Define your Pipeline using Groovy directly or pull it from source control.

Definition: Pipeline script from SCM

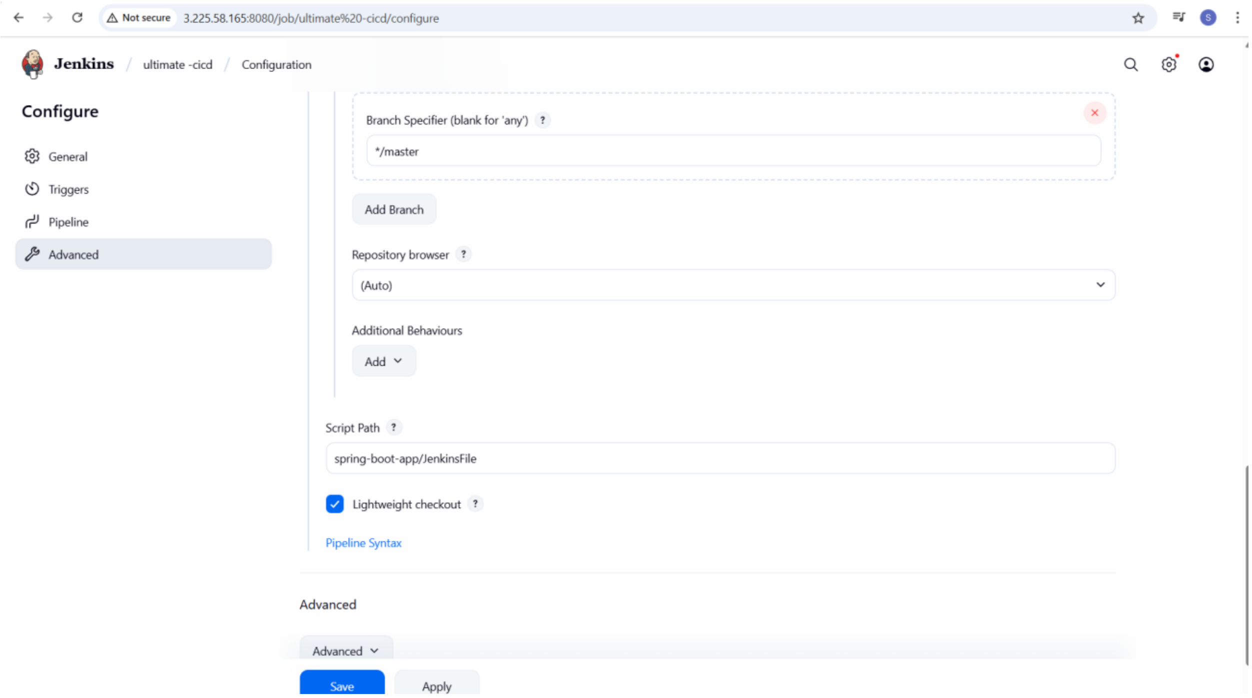
SCM: Git

Repositories

Repository URL: https://github.com/sailakshmi-d/java-maven-sonar-argocd-helm-k8s

Credentials: sailakshmii/******

Branches to build



Not secure 3.225.58.165:8080/job/ultimate%20-cicd/configure

Jenkins / ultimate -cicd / Configuration

**Configure**

Branch Specifier (blank for 'any') ?

*/master

Repository browser ?

(Auto)

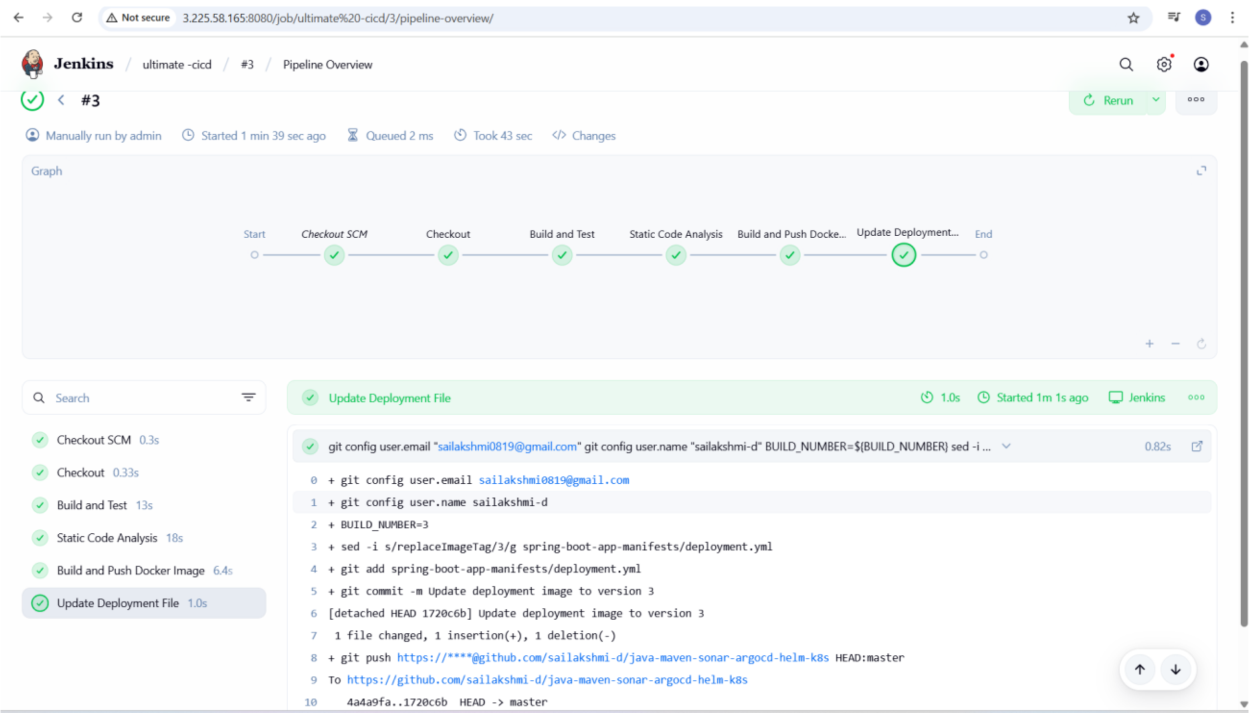
Additional Behaviours

Script Path ?

spring-boot-app/JenkinsFile

☑ Lightweight checkout ?

## Run the pipeline



Not secure 3.225.58.165:8080/job/ultimate%20-cicd/3/pipeline-overview/

Jenkins / ultimate -cicd / #3 / Pipeline Overview

#3

Manually run by admin · Started 1 min 39 sec ago · Queued 2 ms · Took 43 sec · Changes

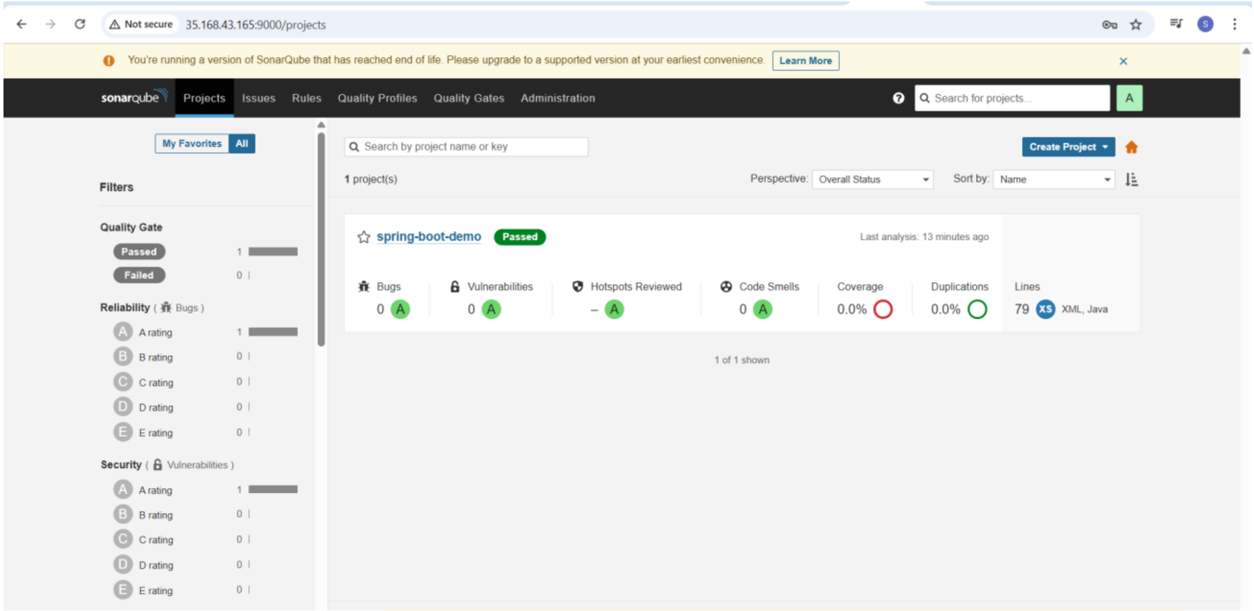
- Checkout SCM 0.3s
- Checkout 0.33s
- Build and Test 13s
- Static Code Analysis 18s
- Build and Push Docker Image 6.4s
- Update Deployment File 1.0s

Update Deployment File 1.0s Started 1m 1s ago Jenkins

git config user.email "sailakshmi0819@gmail.com" git config user.name "sailakshmi-d" BUILD_NUMBER=${BUILD_NUMBER} sed -i ... 0.82s

```
+ git config user.email sailakshmi0819@gmail.com
+ git config user.name sailakshmi-d
+ BUILD_NUMBER=3
+ sed -i s/replaceImageTag/3/g spring-boot-app-manifests/deployment.yml
+ git add spring-boot-app-manifests/deployment.yml
+ git commit -m Update deployment image to version 3
[detached HEAD 1720c6b] Update deployment image to version 3
 1 file changed, 1 insertion(+), 1 deletion(-)
+ git push https://****@github.com/sailakshmi-d/java-maven-sonar-argocd-helm-k8s HEAD:master
To https://github.com/sailakshmi-d/java-maven-sonar-argocd-helm-k8s
   4a4a9fa..1720c6b  HEAD -> master
```

## Sonarqube anylsis



You're running a version of SonarQube that has reached end of life. Please upgrade to a supported version at your earliest convenience.

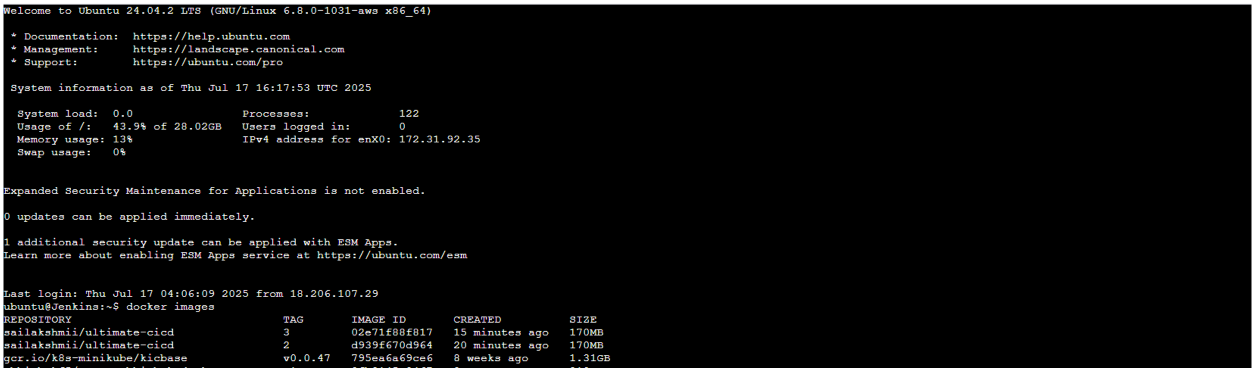
1 project(s)

spring-boot-demo Passed — Last analysis: 13 minutes ago

| Bugs | Vulnerabilities | Hotspots Reviewed | Code Smells | Coverage | Duplications | Lines |
|---|---|---|---|---|---|---|
| 0 A | 0 A | – A | 0 A | 0.0% | 0.0% | 79 XS XML, Java |

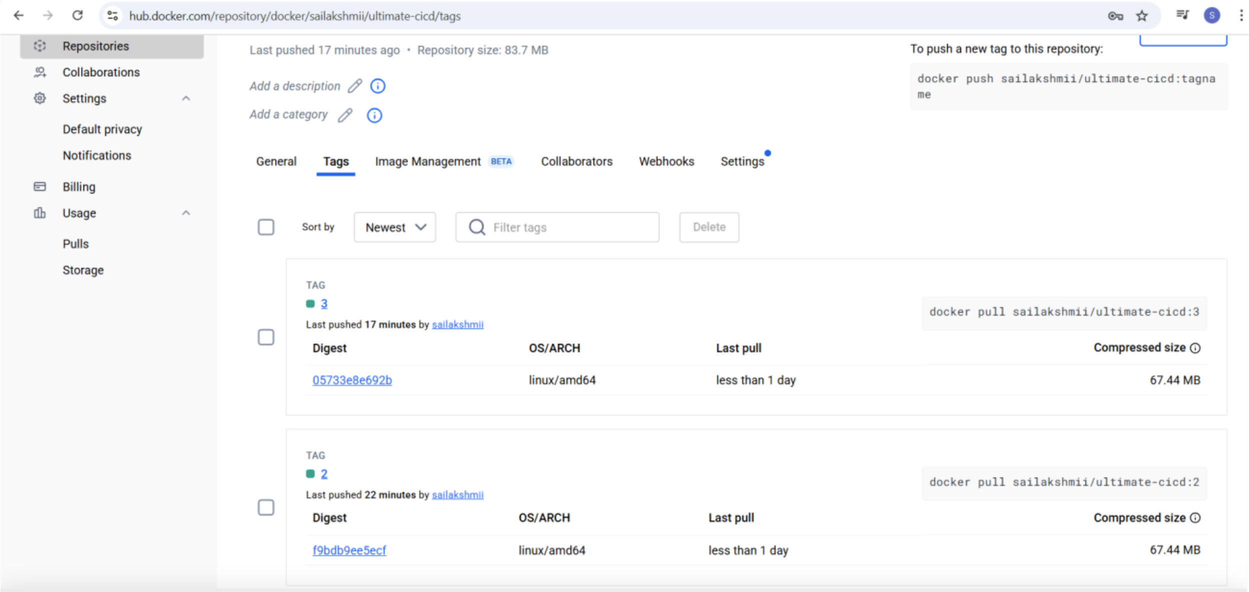
1 of 1 shown

## Docker images got created



```
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1031-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Thu Jul 17 16:17:53 UTC 2025

  System load:  0.0               Processes:             122
  Usage of /:   43.9% of 28.02GB   Users logged in:       0
  Memory usage: 13%               IPv4 address for enX0: 172.31.92.35
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


Last login: Thu Jul 17 04:06:09 2025 from 18.206.107.29
ubuntu@Jenkins:~$ docker images
REPOSITORY                        TAG       IMAGE ID       CREATED          SIZE
sailakshmii/ultimate-cicd         3         02e71f88f817   15 minutes ago   170MB
sailakshmii/ultimate-cicd         2         d939f670d964   20 minutes ago   170MB
gcr.io/k8s-minikube/kicbase       v0.0.47   795ea6a69ce6   8 weeks ago      1.31GB
```

## Check in docker hub


hub.docker.com/repository/docker/sailakshmii/ultimate-cicd/tags

Last pushed 17 minutes ago • Repository size: 83.7 MB

To push a new tag to this repository:

```
docker push sailakshmii/ultimate-cicd:tagname
```

□ TAG
3
Last pushed **17 minutes** by sailakshmii

```
docker pull sailakshmii/ultimate-cicd:3
```

| Digest | OS/ARCH | Last pull | Compressed size |
|---|---|---|---|
| 05733e8e692b | linux/amd64 | less than 1 day | 67.44 MB |

□ TAG
2
Last pushed **22 minutes** by sailakshmii

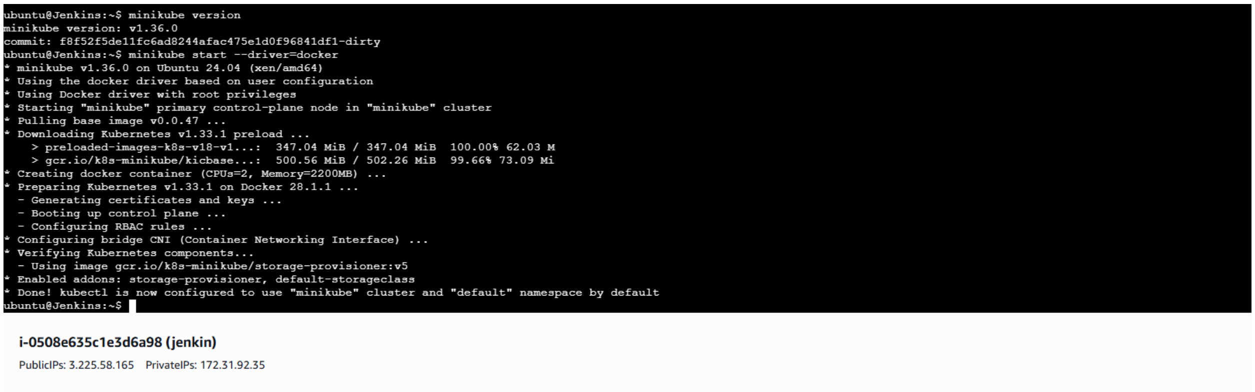
```
docker pull sailakshmii/ultimate-cicd:2
```

| Digest | OS/ARCH | Last pull | Compressed size |
|---|---|---|---|
| f9bdb9ee5ecf | linux/amd64 | less than 1 day | 67.44 MB |

CI is done

Now implemnt the Cd

### Step :7 Install kubectl and minikube in EC2



```
ubuntu@Jenkins:~$ minikube version
minikube version: v1.36.0
commit: f8f52f5de11fc6ad8244afac475e1d0f96841df1-dirty
ubuntu@Jenkins:~$ minikube start --driver=docker
* minikube v1.36.0 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on user configuration
* Using Docker driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.47 ...
* Downloading Kubernetes v1.33.1 preload ...
    > preloaded-images-k8s-v18-v1...:  347.04 MiB / 347.04 MiB  100.00% 62.03 M
    > gcr.io/k8s-minikube/kicbase...:  500.56 MiB / 502.26 MiB  99.66% 73.09 Mi
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Preparing Kubernetes v1.33.1 on Docker 28.1.1 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@Jenkins:~$
```

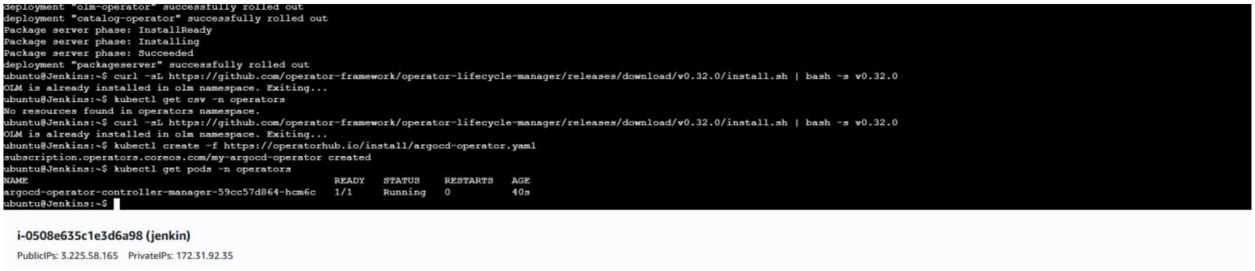
**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165 PrivateIPs: 172.31.92.35

If you want to use any kubernetes controller we need to install these using kubernetes operator

Go to operatorhub.io enter search for argocd and click on install and follow steps

### Step: 8 Install argocd

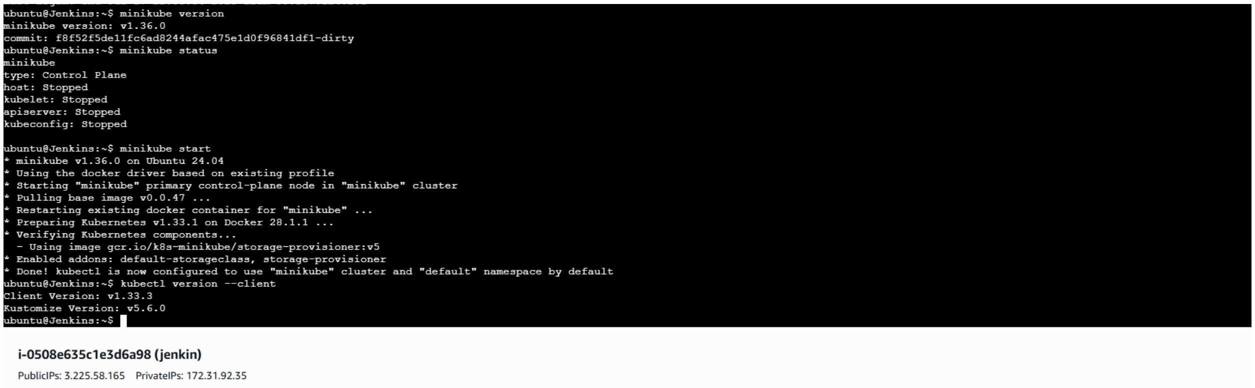


```
deployment "olm-operator" successfully rolled out
deployment "catalog-operator" successfully rolled out
Package server phase: InstallReady
Package server phase: Installing
Package server phase: Succeeded
deployment "packageserver" successfully rolled out
ubuntu@Jenkins:~$ curl -sL https://github.com/operator-framework/operator-lifecycle-manager/releases/download/v0.32.0/install.sh | bash -s v0.32.0
OLM is already installed in olm namespace. Exiting...
ubuntu@Jenkins:~$ kubectl get csv -n operators
No resources found in operators namespace.
ubuntu@Jenkins:~$ curl -sL https://github.com/operator-framework/operator-lifecycle-manager/releases/download/v0.32.0/install.sh | bash -s v0.32.0
OLM is already installed in olm namespace. Exiting...
ubuntu@Jenkins:~$ kubectl create -f https://operatorhub.io/install/argocd-operator.yaml
subscription.operators.coreos.com/my-argocd-operator created
ubuntu@Jenkins:~$ kubectl get pods -n operators
NAME                                              READY   STATUS    RESTARTS   AGE
argocd-operator-controller-manager-59cc57d864-hcm6c   1/1     Running   0          40s
ubuntu@Jenkins:~$
```

**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

Kubernetes and minikube cluster got created

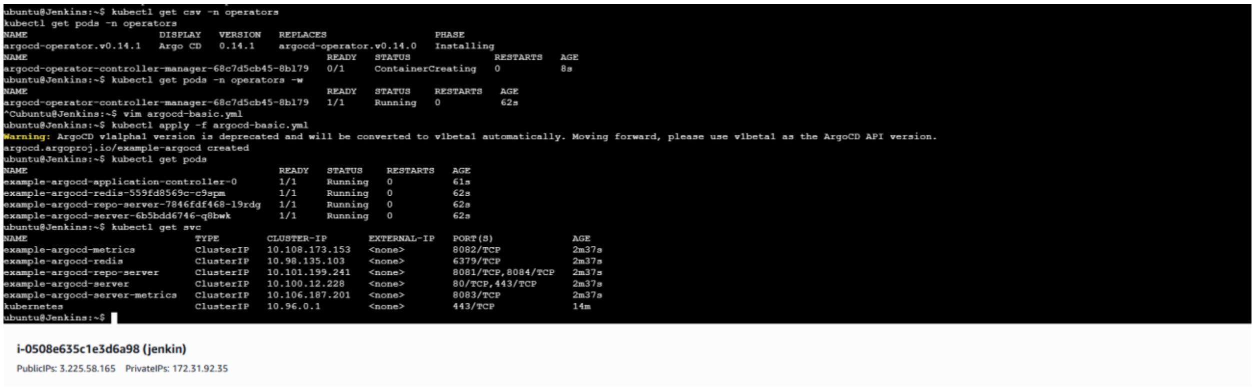


```
ubuntu@Jenkins:~$ minikube version
minikube version: v1.36.0
commit: f8f52f5de11fc6ad8244afac475e1d0f96841df1-dirty
ubuntu@Jenkins:~$ minikube status
minikube
type: Control Plane
host: Stopped
kubelet: Stopped
apiserver: Stopped
kubeconfig: Stopped

ubuntu@Jenkins:~$ minikube start
* minikube v1.36.0 on Ubuntu 24.04
* Using the docker driver based on existing profile
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.47 ...
* Restarting existing docker container for "minikube" ...
* Preparing Kubernetes v1.33.1 on Docker 28.1.1 ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: default-storageclass, storage-provisioner
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@Jenkins:~$ kubectl version --client
Client Version: v1.33.3
Kustomize Version: v5.6.0
ubuntu@Jenkins:~$
```

**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

Check argo cd is running

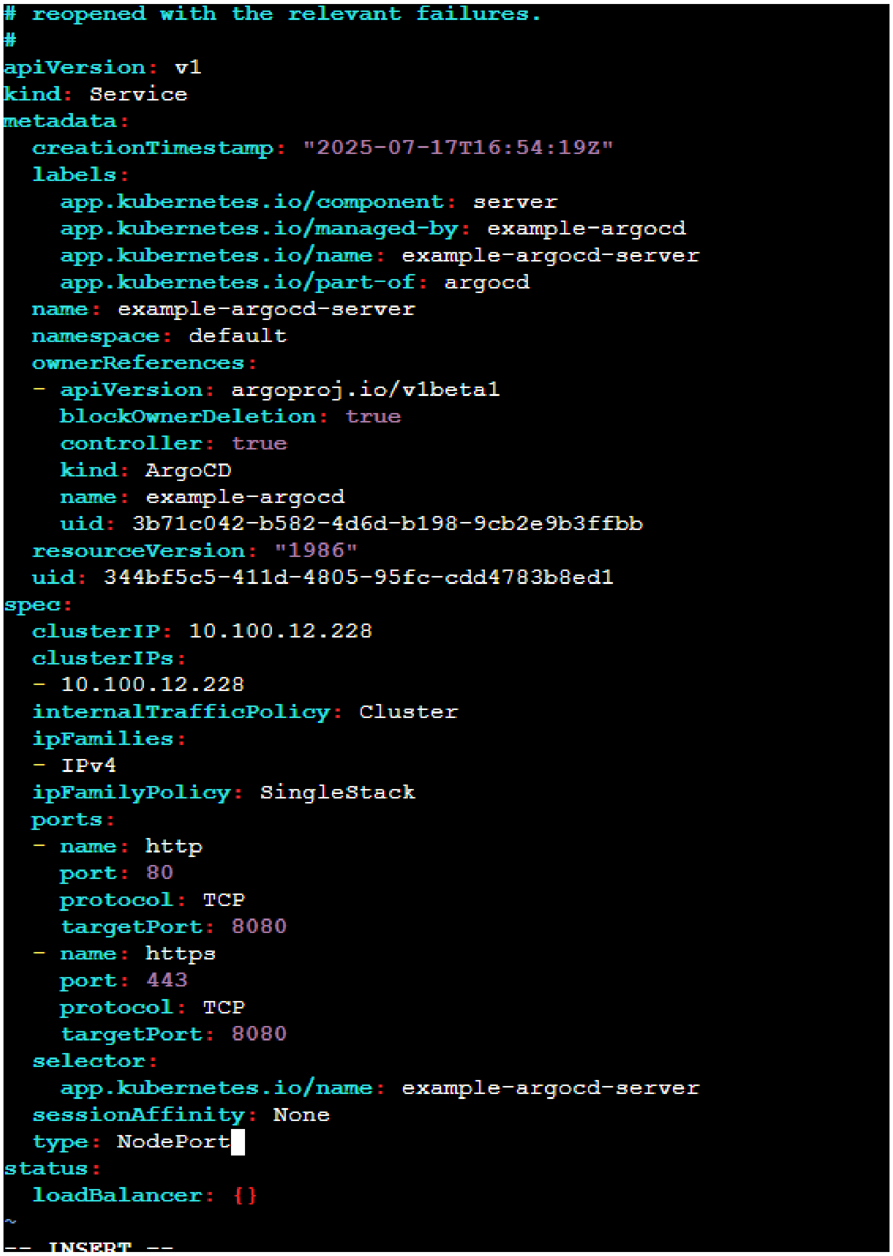


```
ubuntu@Jenkins:~$ kubectl get csv -n operators
kubectl get pods -n operators
NAME                     DISPLAY   VERSION   REPLACES                  PHASE
argocd-operator.v0.14.1   Argo CD   0.14.1    argocd-operator.v0.14.0   Installing
NAME                                              READY   STATUS              RESTARTS   AGE
argocd-operator-controller-manager-68c7d5cb45-8b179   0/1     ContainerCreating   0          8s
ubuntu@Jenkins:~$ kubectl get pods -n operators -w
NAME                                              READY   STATUS    RESTARTS   AGE
argocd-operator-controller-manager-68c7d5cb45-8b179   1/1     Running   0          62s
^Cubuntu@Jenkins:~$ vim argocd-basic.yml
ubuntu@Jenkins:~$ kubectl apply -f argocd-basic.yml
Warning: ArgoCD v1alpha1 version is deprecated and will be converted to v1beta1 automatically. Moving forward, please use v1beta1 as the ArgoCD API version.
argocd.argoproj.io/example-argocd created
ubuntu@Jenkins:~$ kubectl get pods
NAME                                        READY   STATUS    RESTARTS   AGE
example-argocd-application-controller-0      1/1     Running   0          61s
example-argocd-redis-559fd8569c-c9spm        1/1     Running   0          62s
example-argocd-repo-server-7846fdf468-19rdg   1/1     Running   0          62s
example-argocd-server-6b5bdd6746-q8bwk       1/1     Running   0          62s
ubuntu@Jenkins:~$ kubectl get svc
NAME                           TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)             AGE
example-argocd-metrics         ClusterIP   10.108.173.153   <none>        8082/TCP            2m37s
example-argocd-redis           ClusterIP   10.98.135.103    <none>        6379/TCP            2m37s
example-argocd-repo-server     ClusterIP   10.101.199.241   <none>        8081/TCP,8084/TCP   2m37s
example-argocd-server          ClusterIP   10.100.12.228    <none>        80/TCP,443/TCP      2m37s
example-argocd-server-metrics   ClusterIP   10.106.187.201   <none>        8083/TCP            2m37s
kubernetes                     ClusterIP   10.96.0.1        <none>        443/TCP             14m
ubuntu@Jenkins:~$
```

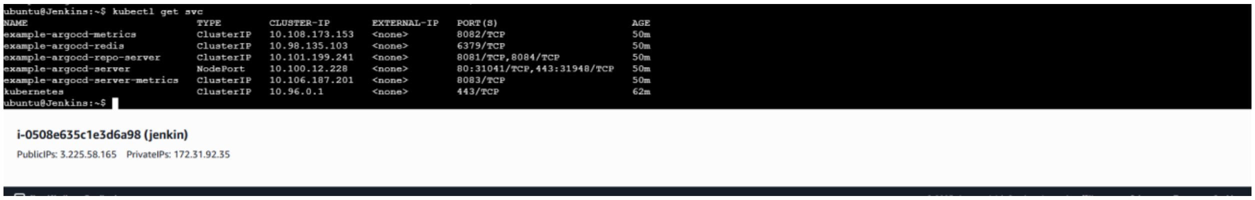
**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

Change cluster ip to node port



```
# reopened with the relevant failures.
#
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: "2025-07-17T16:54:19Z"
  labels:
    app.kubernetes.io/component: server
    app.kubernetes.io/managed-by: example-argocd
    app.kubernetes.io/name: example-argocd-server
    app.kubernetes.io/part-of: argocd
  name: example-argocd-server
  namespace: default
  ownerReferences:
  - apiVersion: argoproj.io/v1beta1
    blockOwnerDeletion: true
    controller: true
    kind: ArgoCD
    name: example-argocd
    uid: 3b71c042-b582-4d6d-b198-9cb2e9b3ffbb
  resourceVersion: "1986"
  uid: 344bf5c5-411d-4805-95fc-cdd4783b8ed1
spec:
  clusterIP: 10.100.12.228
  clusterIPs:
  - 10.100.12.228
  internalTrafficPolicy: Cluster
  ipFamilies:
  - IPv4
  ipFamilyPolicy: SingleStack
  ports:
  - name: http
    port: 80
    protocol: TCP
    targetPort: 8080
  - name: https
    port: 443
    protocol: TCP
    targetPort: 8080
  selector:
    app.kubernetes.io/name: example-argocd-server
  sessionAffinity: None
  type: NodePort
status:
  loadBalancer: {}
~
-- INSERT --
```



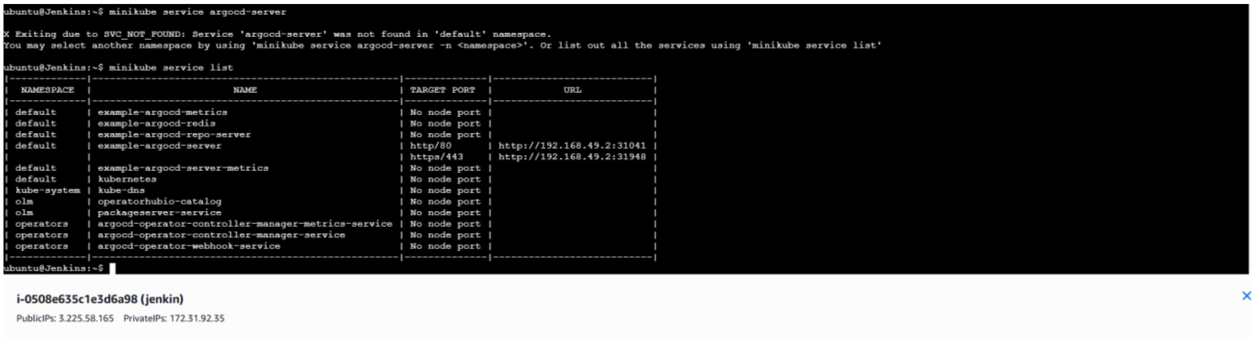
```
ubuntu@Jenkins:~$ kubectl get svc
NAME                           TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                      AGE
example-argocd-metrics         ClusterIP   10.108.173.153   <none>        8082/TCP                     50m
example-argocd-redis           ClusterIP   10.98.135.103    <none>        6379/TCP                     50m
example-argocd-repo-server     ClusterIP   10.101.199.241   <none>        8081/TCP,8084/TCP            50m
example-argocd-server          NodePort    10.100.12.228    <none>        80:31041/TCP,443:31948/TCP   50m
example-argocd-server-metrics  ClusterIP   10.106.187.201   <none>        8083/TCP                     50m
kubernetes                     ClusterIP   10.96.0.1        <none>        443/TCP                      62m
ubuntu@Jenkins:~$
```

i-0508e635c1e3d6a98 (jenkin)

PublicIPs: 3.225.58.165   PrivateIPs: 172.31.92.35

We will get the url

http://192.168.49.2:31041



```
ubuntu@Jenkins:~$ minikube service argocd-server

X Exiting due to SVC_NOT_FOUND: Service 'argocd-server' was not found in 'default' namespace.
You may select another namespace by using 'minikube service argocd-server -n <namespace>'. Or list out all the services using 'minikube service list'

ubuntu@Jenkins:~$ minikube service list
|-------------|----------------------------------------------------|--------------|---------------------------|
|  NAMESPACE  |                        NAME                        | TARGET PORT  |            URL            |
|-------------|----------------------------------------------------|--------------|---------------------------|
| default     | example-argocd-metrics                             | No node port |
| default     | example-argocd-redis                               | No node port |
| default     | example-argocd-repo-server                         | No node port |
| default     | example-argocd-server                              | http/80      | http://192.168.49.2:31041 |
|             |                                                    | https/443    | http://192.168.49.2:31948 |
| default     | example-argocd-server-metrics                      | No node port |
| default     | kubernetes                                         | No node port |
| kube-system | kube-dns                                           | No node port |
| olm         | operatorhubio-catalog                              | No node port |
| olm         | packageserver-service                              | No node port |
| operators   | argocd-operator-controller-manager-metrics-service | No node port |
| operators   | argocd-operator-controller-manager-service         | No node port |
| operators   | argocd-operator-webhook-service                    | No node port |
|-------------|----------------------------------------------------|--------------|---------------------------|
ubuntu@Jenkins:~$
```

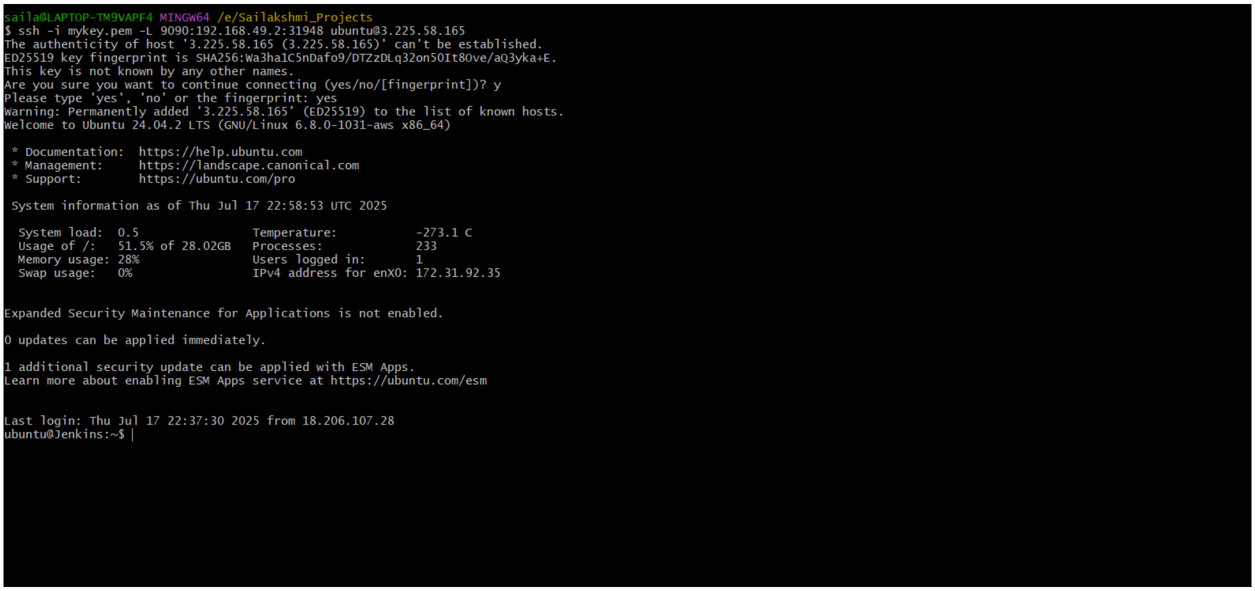
**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

Ssh into you local machine and access

ssh -i mykey.pem -L 9090:192.168.49.2:31948 ubuntu@3.225.58.165

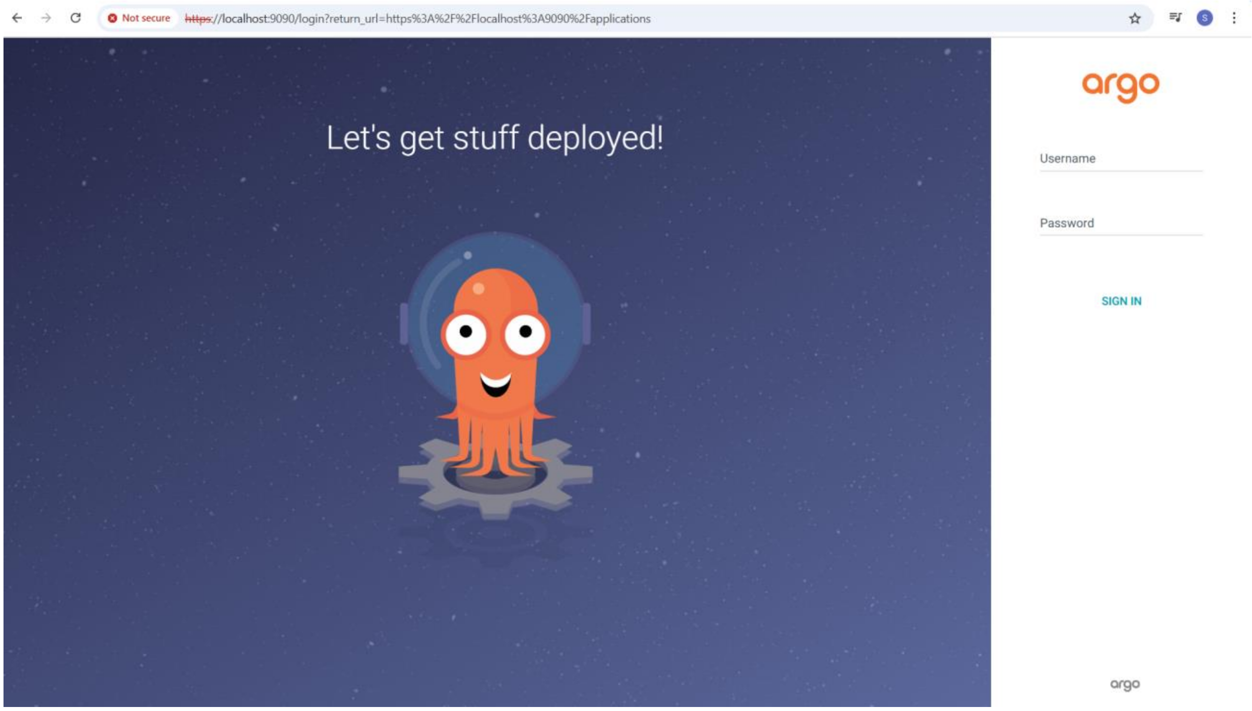
This forwards your **local port 9090** to ArgoCD's HTTPS service running on EC2 Minikube.



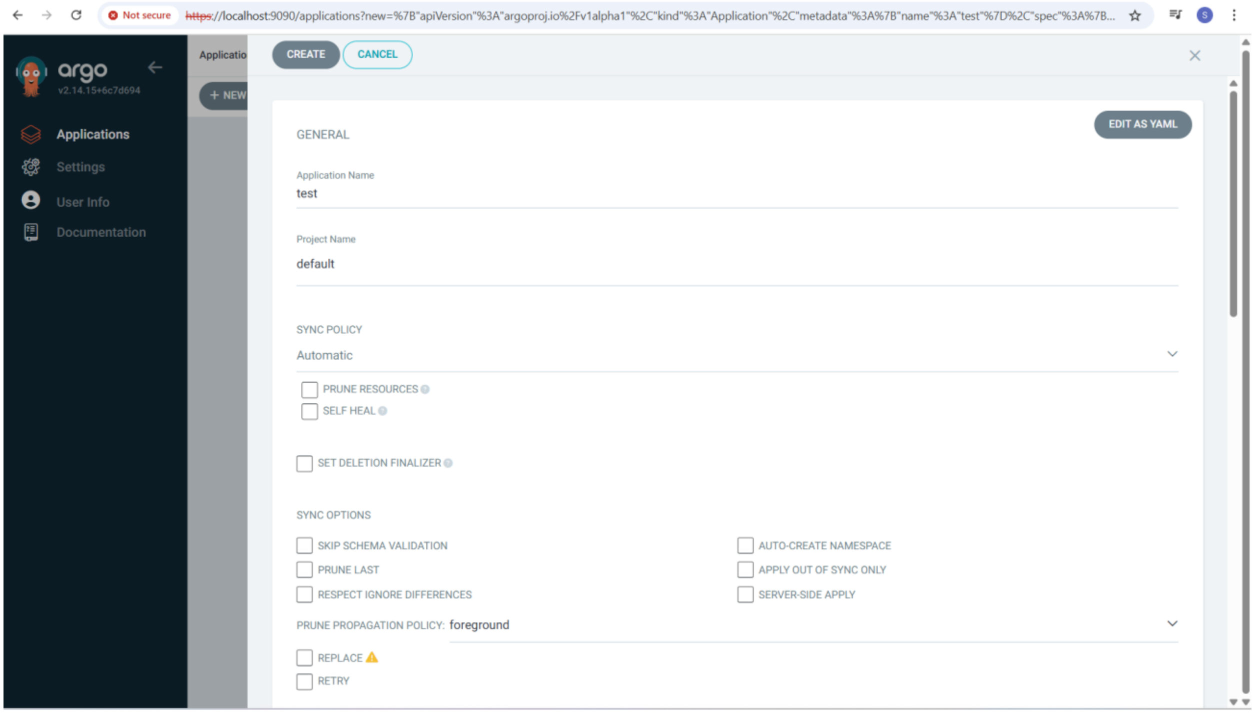
```
saila@LAPTOP-TM9VAPF4 MINGW64 /e/Sailakshmi_Projects
$ ssh -i mykey.pem -L 9090:192.168.49.2:31948 ubuntu@3.225.58.165
The authenticity of host '3.225.58.165 (3.225.58.165)' can't be established.
ED25519 key fingerprint is SHA256:Wa3ha1C5nDafo9/DTZzDLq32on5OIt8Ove/aQ3yka+E.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '3.225.58.165' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1031-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Thu Jul 17 22:58:53 UTC 2025

  System load:  0.5               Temperature:           -273.1 C
  Usage of /:   51.5% of 28.02GB   Processes:             233
  Memory usage: 28%               Users logged in:       1
  Swap usage:   0%                IPv4 address for enX0: 172.31.92.35


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


Last login: Thu Jul 17 22:37:30 2025 from 18.206.107.28
ubuntu@Jenkins:~$ |
```

Now access through browser

https://localhost:9090


Not secure
https://localhost:9090/login?return_url=https%3A%2F%2Flocalhost%3A9090%2Fapplications
Let's get stuff deployed!
argo
Username
Password
SIGN IN
argo

## Step 9: Create an application in Argocd


Not secure
https://localhost:9090/applications?new=%7B"apiVersion"%3A"argoproj.io%2Fv1alpha1"%2C"kind"%3A"Application"%2C"metadata"%3A%7B"name"%3A"test"%7D%2C"spec"%3A%7B...
argo
v2.14.15+6c7d694
Applications
Settings
User Info
Documentation
Applicatio
+ NEW
CREATE
CANCEL
EDIT AS YAML

GENERAL

Application Name
test

Project Name
default

SYNC POLICY
Automatic

□ PRUNE RESOURCES
□ SELF HEAL

□ SET DELETION FINALIZER

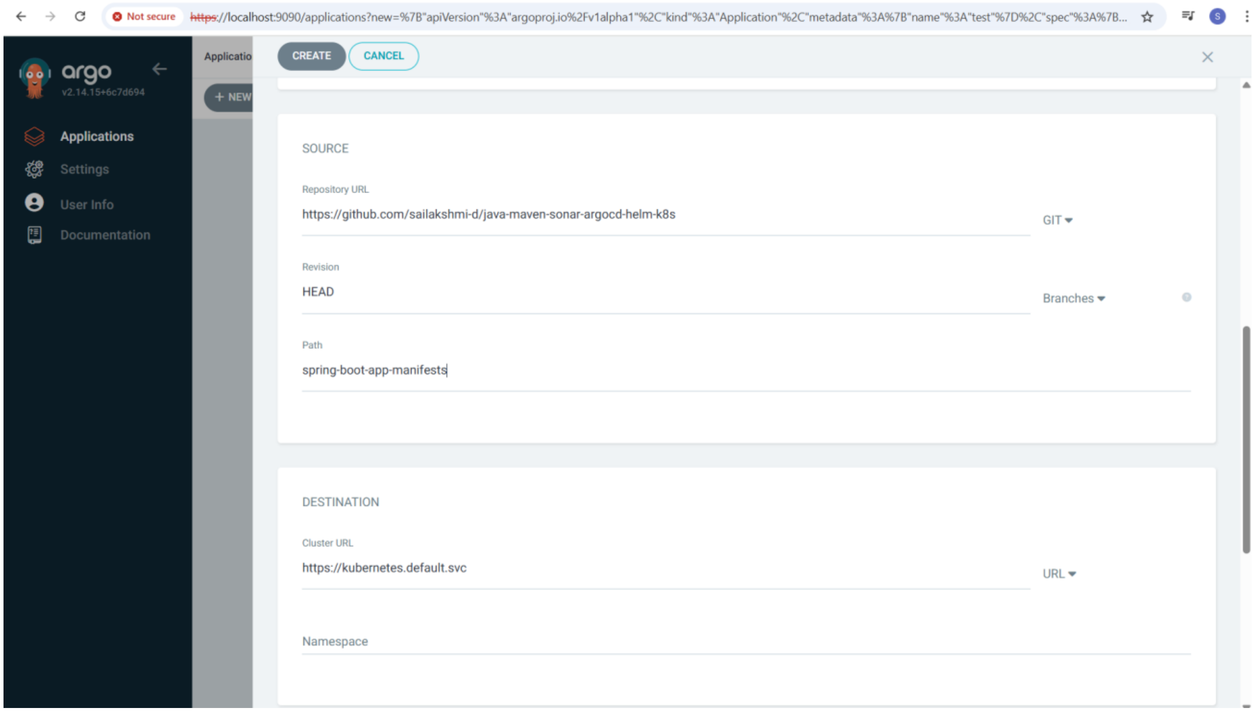
SYNC OPTIONS

□ SKIP SCHEMA VALIDATION
□ PRUNE LAST
□ RESPECT IGNORE DIFFERENCES
□ AUTO-CREATE NAMESPACE
□ APPLY OUT OF SYNC ONLY
□ SERVER-SIDE APPLY

PRUNE PROPAGATION POLICY: foreground

□ REPLACE ⚠
□ RETRY

Make sure you give the correct git repo url and path of the file


Not secure
https://localhost:9090/applications?new=%7B"apiVersion"%3A"argoproj.io%2Fv1alpha1"%2C"kind"%3A"Application"%2C"metadata"%3A%7B"name"%3A"test"%7D%2C"spec"%3A%7B...
argo
v2.14.15+6c7d694
Applications
Settings
User Info
Documentation
Applicatio
+ NEW
CREATE
CANCEL

SOURCE

Repository URL
https://github.com/sailakshmi-d/java-maven-sonar-argocd-helm-k8s
GIT ▾

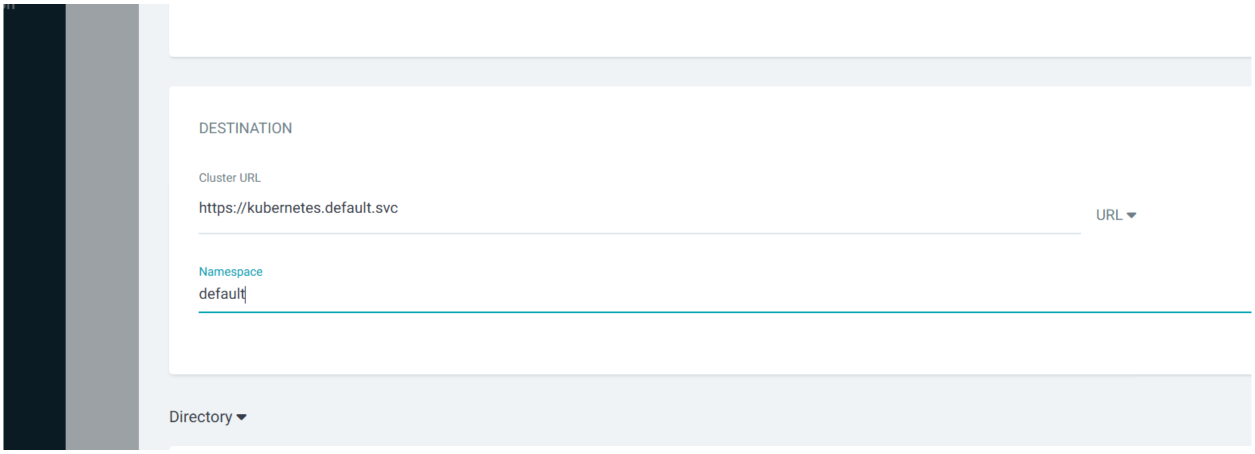
Revision
HEAD
Branches ▾

Path
spring-boot-app-manifests

DESTINATION

Cluster URL
https://kubernetes.default.svc
URL ▾

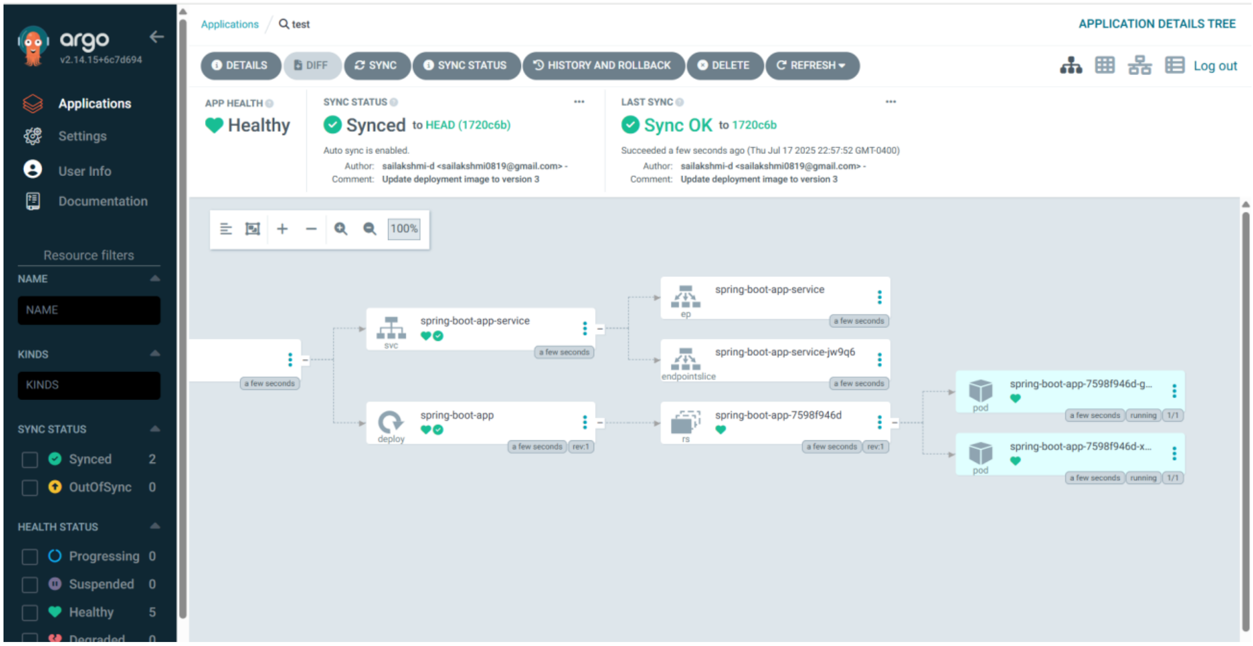
Namespace



DESTINATION

Cluster URL
https://kubernetes.default.svc
URL ▾

Namespace
default

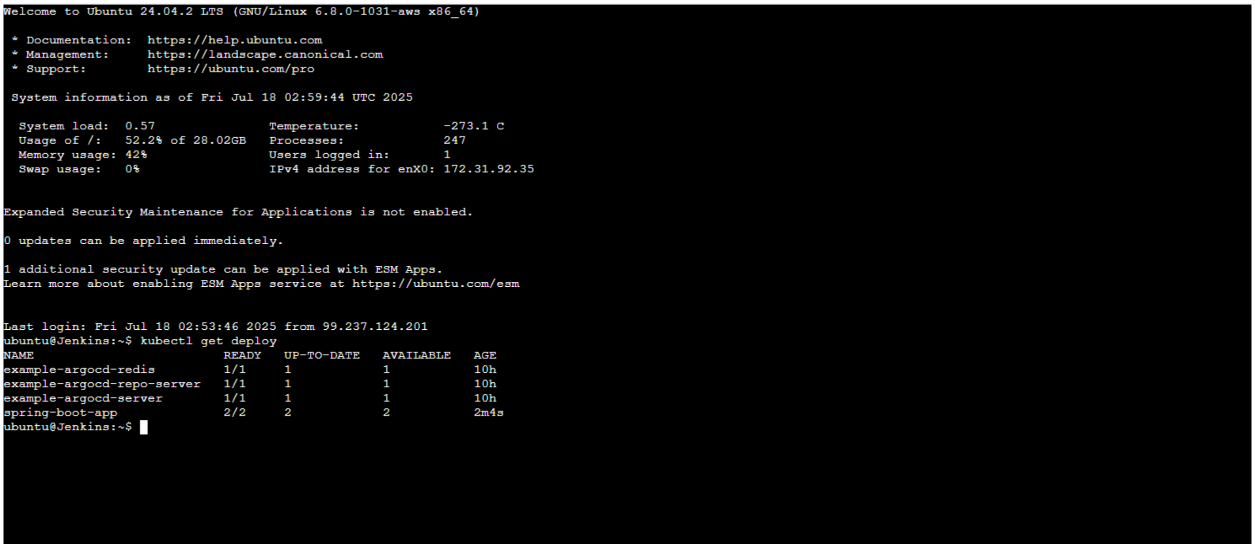


APP HEALTH
Healthy

SYNC STATUS
Synced to HEAD (1720c6b)
Auto sync is enabled.
Author: sailakshmi-d <sailakshmi0819@gmail.com> -
Comment: Update deployment image to version 3

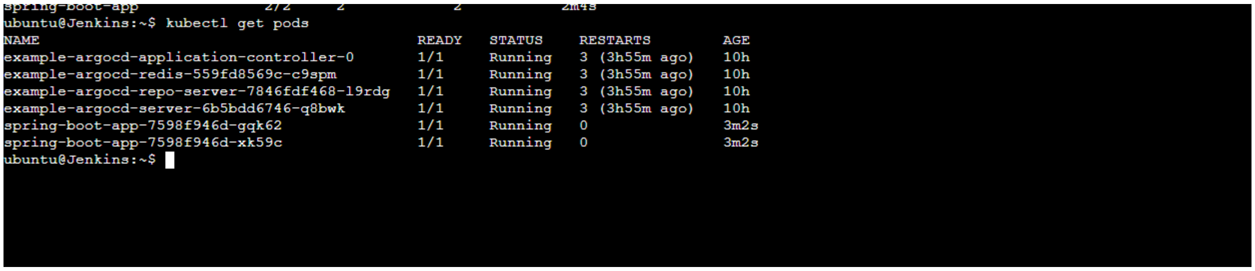
LAST SYNC
Sync OK to 1720c6b
Succeeded a few seconds ago (Thu Jul 17 2025 22:57:52 GMT-0400)
Author: sailakshmi-d <sailakshmi0819@gmail.com> -
Comment: Update deployment image to version 3

## Kubectl get deploy



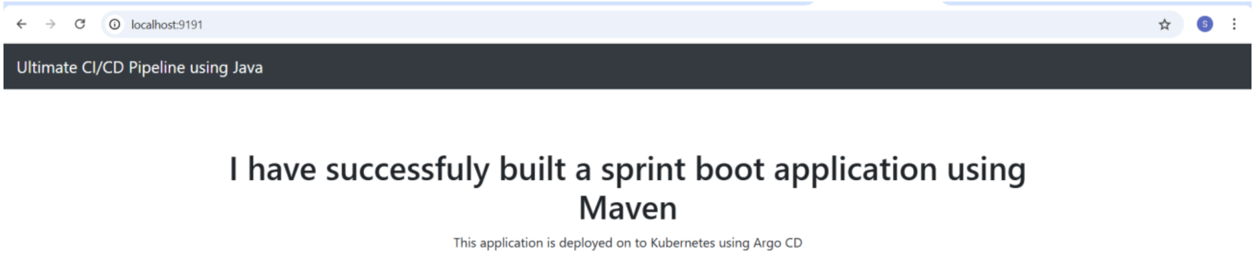
```
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1031-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Jul 18 02:59:44 UTC 2025

  System load:  0.57               Temperature:           -273.1 C
  Usage of /:   52.2% of 28.02GB   Processes:             247
  Memory usage: 42%                Users logged in:       1
  Swap usage:   0%                 IPv4 address for enX0: 172.31.92.35


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


Last login: Fri Jul 18 02:53:46 2025 from 99.237.124.201
ubuntu@Jenkins:~$ kubectl get deploy
NAME                          READY   UP-TO-DATE   AVAILABLE   AGE
example-argocd-redis          1/1     1            1           10h
example-argocd-repo-server    1/1     1            1           10h
example-argocd-server         1/1     1            1           10h
spring-boot-app               2/2     2            2           2m4s
ubuntu@Jenkins:~$
```

## Kubectl get pods



```
spring-boot-app               2/2     2            2           2m4s
ubuntu@Jenkins:~$ kubectl get pods
NAME                                                READY   STATUS    RESTARTS        AGE
example-argocd-application-controller-0             1/1     Running   3 (3h55m ago)   10h
example-argocd-redis-559fd8569c-c9spm               1/1     Running   3 (3h55m ago)   10h
example-argocd-repo-server-7846fdf468-l9rdg         1/1     Running   3 (3h55m ago)   10h
example-argocd-server-6b5bdd6746-q8bwk              1/1     Running   3 (3h55m ago)   10h
spring-boot-app-7598f946d-gqk62                     1/1     Running   0               3m2s
spring-boot-app-7598f946d-xk59c                     1/1     Running   0               3m2s
ubuntu@Jenkins:~$
```

Now access the application through browser

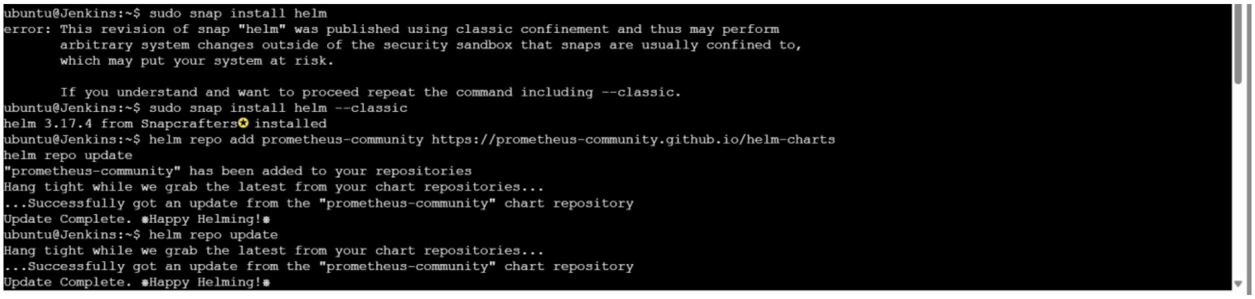


Ultimate CI/CD Pipeline using Java

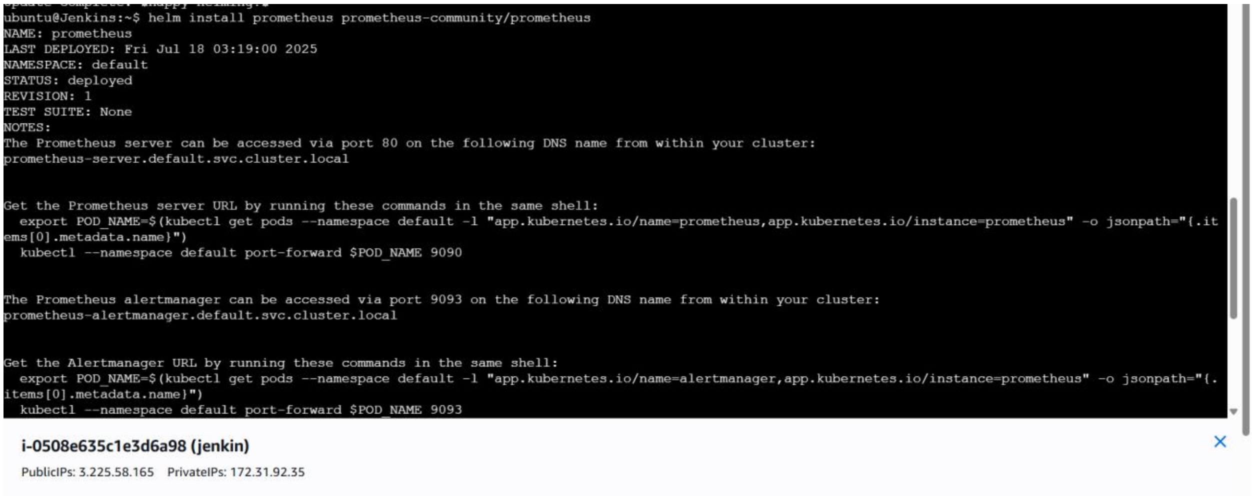
I have successfuly built a sprint boot application using Maven

This application is deployed on to Kubernetes using Argo CD

## Step:10 Moinitoring tools prometheus and grafana



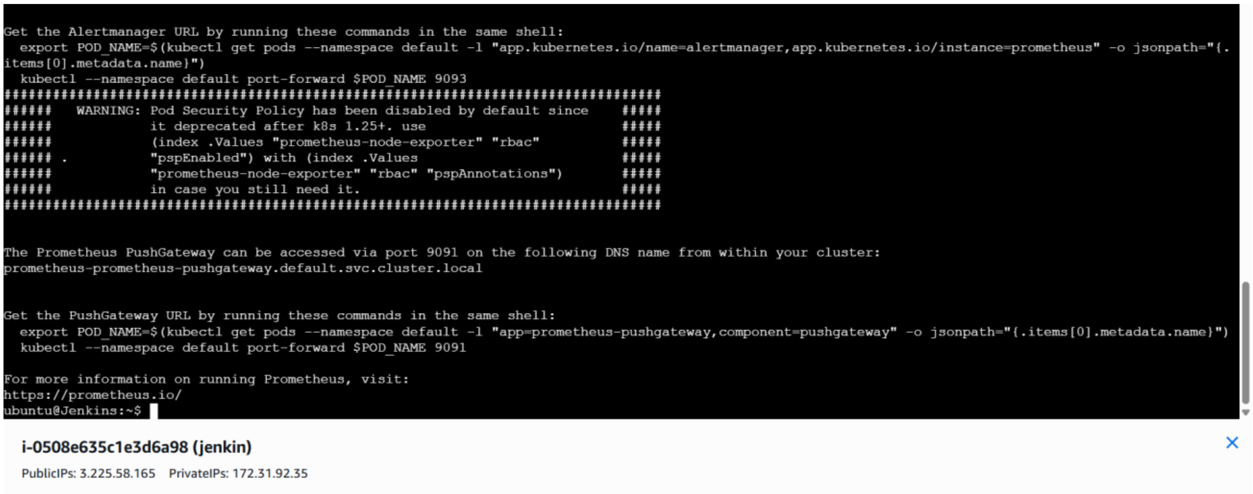
```
ubuntu@Jenkins:~$ sudo snap install helm
error: This revision of snap "helm" was published using classic confinement and thus may perform
       arbitrary system changes outside of the security sandbox that snaps are usually confined to,
       which may put your system at risk.

       If you understand and want to proceed repeat the command including --classic.
ubuntu@Jenkins:~$ sudo snap install helm --classic
helm 3.17.4 from Snapcrafters✪ installed
ubuntu@Jenkins:~$ helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo update
"prometheus-community" has been added to your repositories
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "prometheus-community" chart repository
Update Complete. ⎈Happy Helming!⎈
ubuntu@Jenkins:~$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "prometheus-community" chart repository
Update Complete. ⎈Happy Helming!⎈
```



```
ubuntu@Jenkins:~$ helm install prometheus prometheus-community/prometheus
NAME: prometheus
LAST DEPLOYED: Fri Jul 18 03:19:00 2025
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
The Prometheus server can be accessed via port 80 on the following DNS name from within your cluster:
prometheus-server.default.svc.cluster.local


Get the Prometheus server URL by running these commands in the same shell:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=prometheus,app.kubernetes.io/instance=prometheus" -o jsonpath="{.it
ems[0].metadata.name}")
  kubectl --namespace default port-forward $POD_NAME 9090


The Prometheus alertmanager can be accessed via port 9093 on the following DNS name from within your cluster:
prometheus-alertmanager.default.svc.cluster.local


Get the Alertmanager URL by running these commands in the same shell:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=alertmanager,app.kubernetes.io/instance=prometheus" -o jsonpath="{.
items[0].metadata.name}")
  kubectl --namespace default port-forward $POD_NAME 9093
```

**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

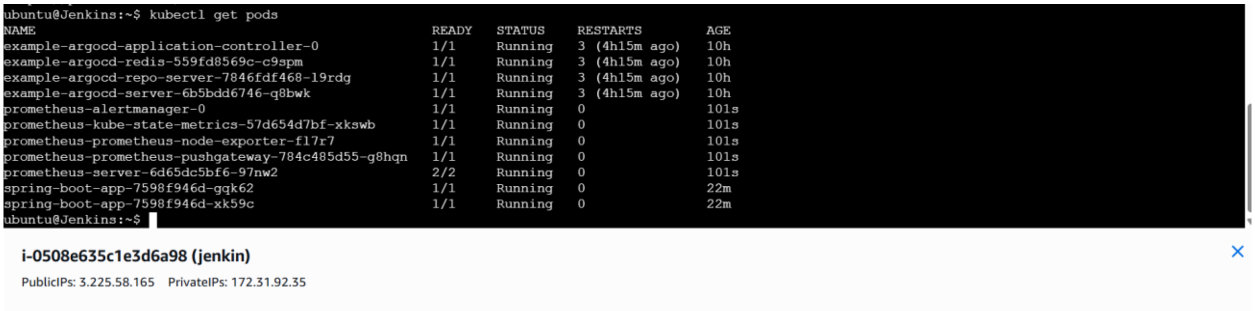


```
Get the Alertmanager URL by running these commands in the same shell:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=alertmanager,app.kubernetes.io/instance=prometheus" -o jsonpath="{.
items[0].metadata.name}")
  kubectl --namespace default port-forward $POD_NAME 9093
#################################################################################
######   WARNING: Pod Security Policy has been disabled by default since    #####
######            it deprecated after k8s 1.25+. use                        #####
######            (index .Values "prometheus-node-exporter" "rbac"          #####
###### .          "pspEnabled") with (index .Values                         #####
######            "prometheus-node-exporter" "rbac" "pspAnnotations")       #####
######            in case you still need it.                                #####
#################################################################################


The Prometheus PushGateway can be accessed via port 9091 on the following DNS name from within your cluster:
prometheus-prometheus-pushgateway.default.svc.cluster.local


Get the PushGateway URL by running these commands in the same shell:
  export POD_NAME=$(kubectl get pods --namespace default -l "app=prometheus-pushgateway,component=pushgateway" -o jsonpath="{.items[0].metadata.name}")
  kubectl --namespace default port-forward $POD_NAME 9091

For more information on running Prometheus, visit:
https://prometheus.io/
ubuntu@Jenkins:~$
```

**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165   PrivateIPs: 172.31.92.35

Kubectl get pods check prometheus is running or not

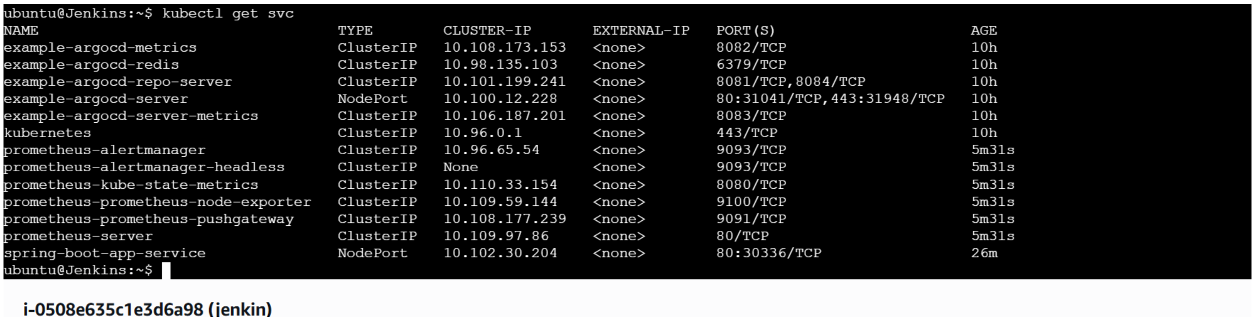


```
ubuntu@Jenkins:~$ kubectl get pods
NAME                                                READY   STATUS    RESTARTS        AGE
example-argocd-application-controller-0             1/1     Running   3 (4h15m ago)   10h
example-argocd-redis-559fd8569c-c9spm               1/1     Running   3 (4h15m ago)   10h
example-argocd-repo-server-7846fdf468-19rdg         1/1     Running   3 (4h15m ago)   10h
example-argocd-server-6b5bdd6746-q8bwk              1/1     Running   3 (4h15m ago)   10h
prometheus-alertmanager-0                           1/1     Running   0               101s
prometheus-kube-state-metrics-57d654d7bf-xkswb      1/1     Running   0               101s
prometheus-prometheus-node-exporter-f17r7           1/1     Running   0               101s
prometheus-prometheus-pushgateway-784c485d55-g8hqn  1/1     Running   0               101s
prometheus-server-6d65dc5bf6-97nw2                  2/2     Running   0               101s
spring-boot-app-7598f946d-gqk62                     1/1     Running   0               22m
spring-boot-app-7598f946d-xk59c                     1/1     Running   0               22m
ubuntu@Jenkins:~$
```

**i-0508e635c1e3d6a98 (jenkin)**

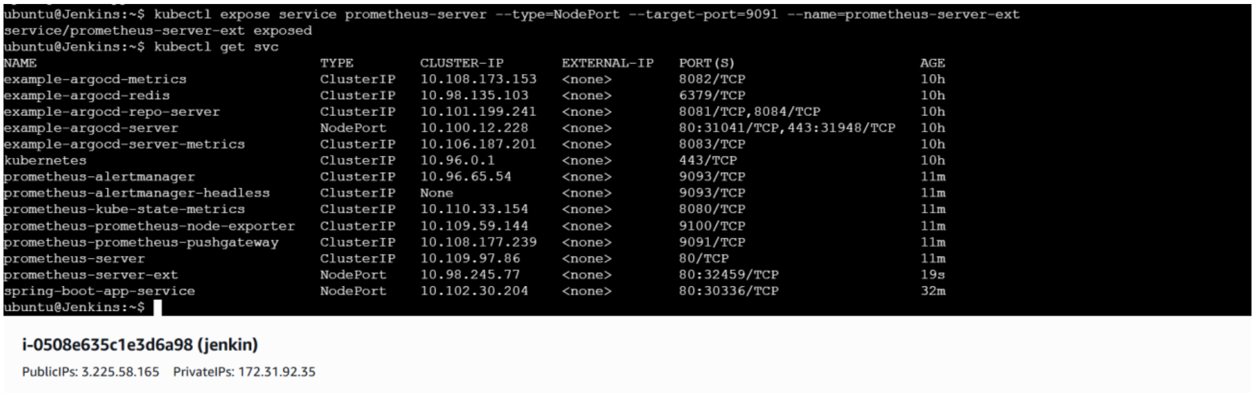
PublicIPs: 3.225.58.165   PrivateIPs: 172.31.92.35

Kubectl get svc



```
ubuntu@Jenkins:~$ kubectl get svc
NAME                                  TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                      AGE
example-argocd-metrics                ClusterIP   10.108.173.153   <none>        8082/TCP                     10h
example-argocd-redis                  ClusterIP   10.98.135.103    <none>        6379/TCP                     10h
example-argocd-repo-server            ClusterIP   10.101.199.241   <none>        8081/TCP,8084/TCP            10h
example-argocd-server                 NodePort    10.100.12.228    <none>        80:31041/TCP,443:31948/TCP   10h
example-argocd-server-metrics         ClusterIP   10.106.187.201   <none>        8083/TCP                     10h
kubernetes                            ClusterIP   10.96.0.1        <none>        443/TCP                      10h
prometheus-alertmanager               ClusterIP   10.96.65.54      <none>        9093/TCP                     5m31s
prometheus-alertmanager-headless      ClusterIP   None             <none>        9093/TCP                     5m31s
prometheus-kube-state-metrics         ClusterIP   10.110.33.154    <none>        8080/TCP                     5m31s
prometheus-prometheus-node-exporter   ClusterIP   10.109.59.144    <none>        9100/TCP                     5m31s
prometheus-prometheus-pushgateway     ClusterIP   10.108.177.239   <none>        9091/TCP                     5m31s
prometheus-server                     ClusterIP   10.109.97.86     <none>        80/TCP                       5m31s
spring-boot-app-service               NodePort    10.102.30.204    <none>        80:30336/TCP                 26m
ubuntu@Jenkins:~$
```

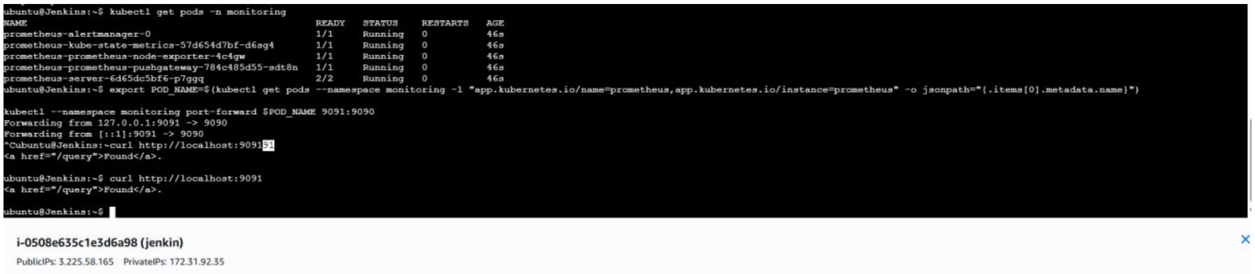
**i-0508e635c1e3d6a98 (jenkin)**



```
ubuntu@Jenkins:~$ kubectl expose service prometheus-server --type=NodePort --target-port=9091 --name=prometheus-server-ext
service/prometheus-server-ext exposed
ubuntu@Jenkins:~$ kubectl get svc
NAME                                  TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                      AGE
example-argocd-metrics                ClusterIP   10.108.173.153   <none>        8082/TCP                     10h
example-argocd-redis                  ClusterIP   10.98.135.103    <none>        6379/TCP                     10h
example-argocd-repo-server            ClusterIP   10.101.199.241   <none>        8081/TCP,8084/TCP            10h
example-argocd-server                 NodePort    10.100.12.228    <none>        80:31041/TCP,443:31948/TCP   10h
example-argocd-server-metrics         ClusterIP   10.106.187.201   <none>        8083/TCP                     10h
kubernetes                            ClusterIP   10.96.0.1        <none>        443/TCP                      10h
prometheus-alertmanager               ClusterIP   10.96.65.54      <none>        9093/TCP                     11m
prometheus-alertmanager-headless      ClusterIP   None             <none>        9093/TCP                     11m
prometheus-kube-state-metrics         ClusterIP   10.110.33.154    <none>        8080/TCP                     11m
prometheus-prometheus-node-exporter   ClusterIP   10.109.59.144    <none>        9100/TCP                     11m
prometheus-prometheus-pushgateway     ClusterIP   10.108.177.239   <none>        9091/TCP                     11m
prometheus-server                     ClusterIP   10.109.97.86     <none>        80/TCP                       11m
prometheus-server-ext                 NodePort    10.98.245.77     <none>        80:32459/TCP                 19s
spring-boot-app-service               NodePort    10.102.30.204    <none>        80:30336/TCP                 32m
ubuntu@Jenkins:~$
```

**i-0508e635c1e3d6a98 (jenkin)**

PublicIPs: 3.225.58.165   PrivateIPs: 172.31.92.35



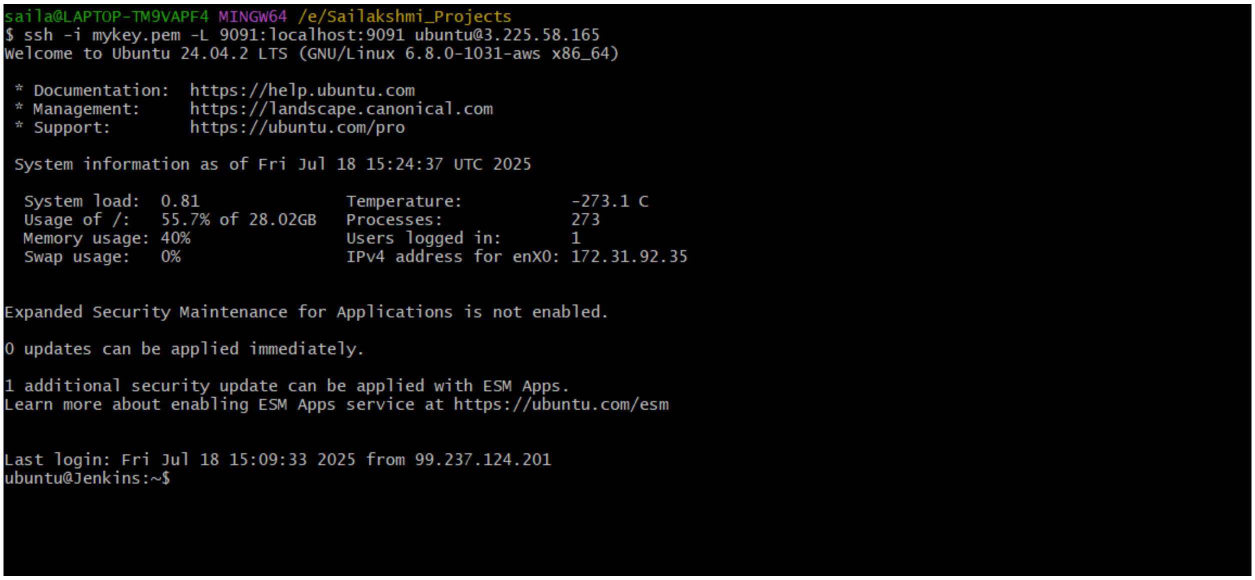
```
ubuntu@Jenkins:~$ kubectl get pods -n monitoring
NAME                                                 READY   STATUS    RESTARTS   AGE
prometheus-alertmanager-0                            1/1     Running   0          46s
prometheus-kube-state-metrics-57d654d7bf-d6sg4       1/1     Running   0          46s
prometheus-prometheus-node-exporter-4c4gw            1/1     Running   0          46s
prometheus-prometheus-pushgateway-784c485d55-sdt8n   1/1     Running   0          46s
prometheus-server-6d65dc5bf6-p7ggq                   2/2     Running   0          46s
ubuntu@Jenkins:~$ export POD_NAME=$(kubectl get pods --namespace monitoring -l "app.kubernetes.io/name=prometheus,app.kubernetes.io/instance=prometheus" -o jsonpath="{.items[0].metadata.name}")

kubectl --namespace monitoring port-forward $POD_NAME 9091:9090
Forwarding from 127.0.0.1:9091 -> 9090
Forwarding from [::1]:9091 -> 9090
^Cubuntu@Jenkins:~curl http://localhost:909191
<a href="/query">Found</a>.

ubuntu@Jenkins:~$ curl http://localhost:9091
<a href="/query">Found</a>.

ubuntu@Jenkins:~$
```

i-0508e635c1e3d6a98 (jenkin)

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35

SSH Port Forwarding Command

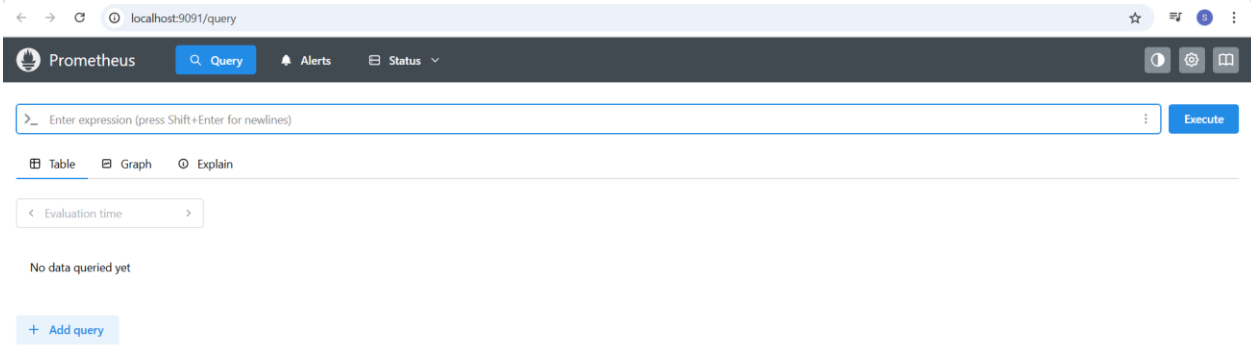
ssh -i mykey.pem -L 9091:localhost:9091 ubuntu@3.225.58.165



```
saila@LAPTOP-TM9VAPF4 MINGW64 /e/Sailakshmi_Projects
$ ssh -i mykey.pem -L 9091:localhost:9091 ubuntu@3.225.58.165
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1031-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Jul 18 15:24:37 UTC 2025

  System load:  0.81               Temperature:           -273.1 C
  Usage of /:   55.7% of 28.02GB   Processes:             273
  Memory usage: 40%                Users logged in:       1
  Swap usage:   0%                 IPv4 address for enX0: 172.31.92.35


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


Last login: Fri Jul 18 15:09:33 2025 from 99.237.124.201
ubuntu@Jenkins:~$
```

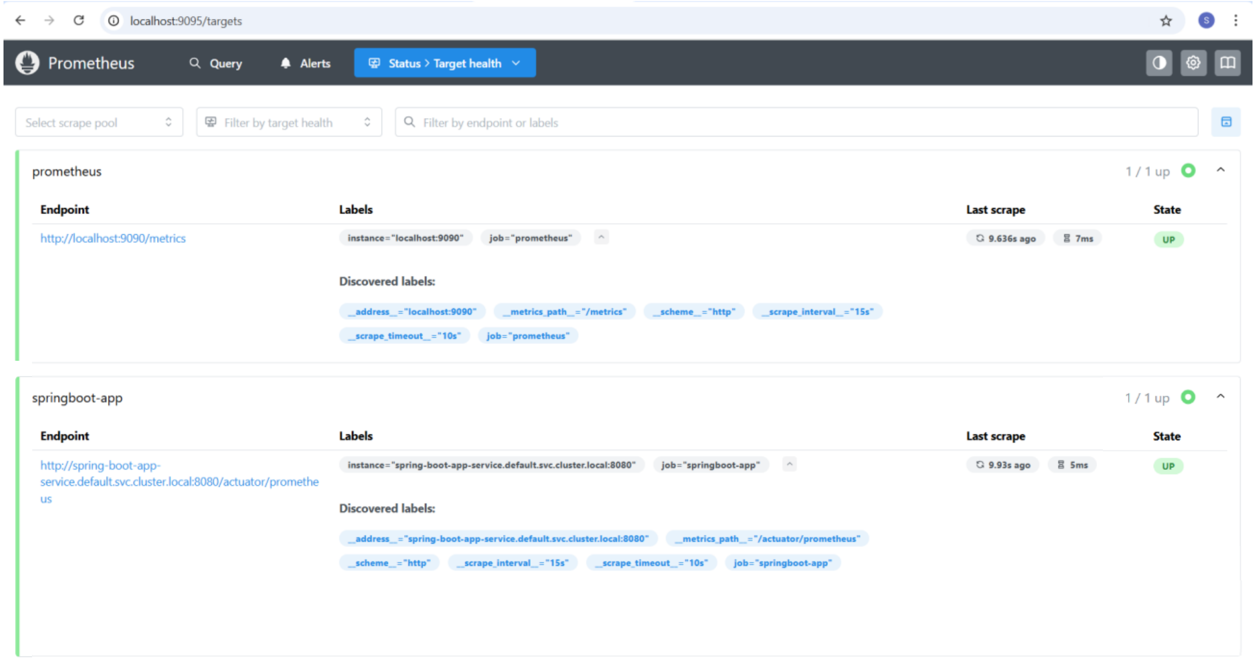
Run this url in browser

http://localhost:9091



No data queried yet

Go to status --> targets



prometheus — 1 / 1 up

| Endpoint | Labels | Last scrape | State |
|---|---|---|---|
| http://localhost:9090/metrics | instance="localhost:9090" job="prometheus" | 9.636s ago 7ms | UP |

Discovered labels:

__address__="localhost:9090" __metrics_path__="/metrics" __scheme__="http" __scrape_interval__="15s" __scrape_timeout__="10s" job="prometheus"

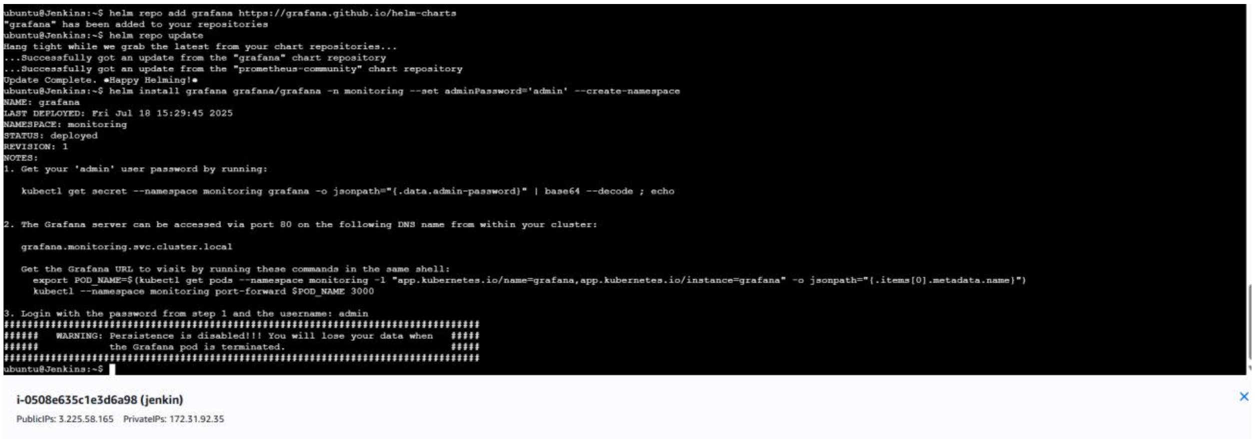
springboot-app — 1 / 1 up

| Endpoint | Labels | Last scrape | State |
|---|---|---|---|
| http://spring-boot-app-service.default.svc.cluster.local:8080/actuator/prometheus | instance="spring-boot-app-service.default.svc.cluster.local:8080" job="springboot-app" | 9.93s ago 5ms | UP |

Discovered labels:

__address__="spring-boot-app-service.default.svc.cluster.local:8080" __metrics_path__="/actuator/prometheus" __scheme__="http" __scrape_interval__="15s" __scrape_timeout__="10s" job="springboot-app"

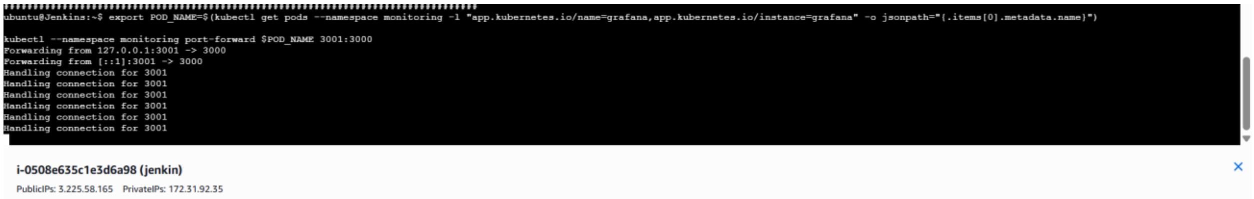
## Step 11: Install Grafana via Helm



```
ubuntu@Jenkins:~$ helm repo add grafana https://grafana.github.io/helm-charts
"grafana" has been added to your repositories
ubuntu@Jenkins:~$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "grafana" chart repository
...Successfully got an update from the "prometheus-community" chart repository
Update Complete. ●Happy Helming!●
ubuntu@Jenkins:~$ helm install grafana grafana/grafana -n monitoring --set adminPassword='admin' --create-namespace
NAME: grafana
LAST DEPLOYED: Fri Jul 18 15:29:45 2025
NAMESPACE: monitoring
STATUS: deployed
REVISION: 1
NOTES:
1. Get your 'admin' user password by running:

   kubectl get secret --namespace monitoring grafana -o jsonpath="{.data.admin-password}" | base64 --decode ; echo


2. The Grafana server can be accessed via port 80 on the following DNS name from within your cluster:

   grafana.monitoring.svc.cluster.local

   Get the Grafana URL to visit by running these commands in the same shell:
     export POD_NAME=$(kubectl get pods --namespace monitoring -l "app.kubernetes.io/name=grafana,app.kubernetes.io/instance=grafana" -o jsonpath="{.items[0].metadata.name}")
     kubectl --namespace monitoring port-forward $POD_NAME 3000

3. Login with the password from step 1 and the username: admin
#################################################################################
######   WARNING: Persistence is disabled!!! You will lose your data when   #####
######            the Grafana pod is terminated.                            #####
#################################################################################
ubuntu@Jenkins:~$
```

i-0508e635c1e3d6a98 (jenkin)

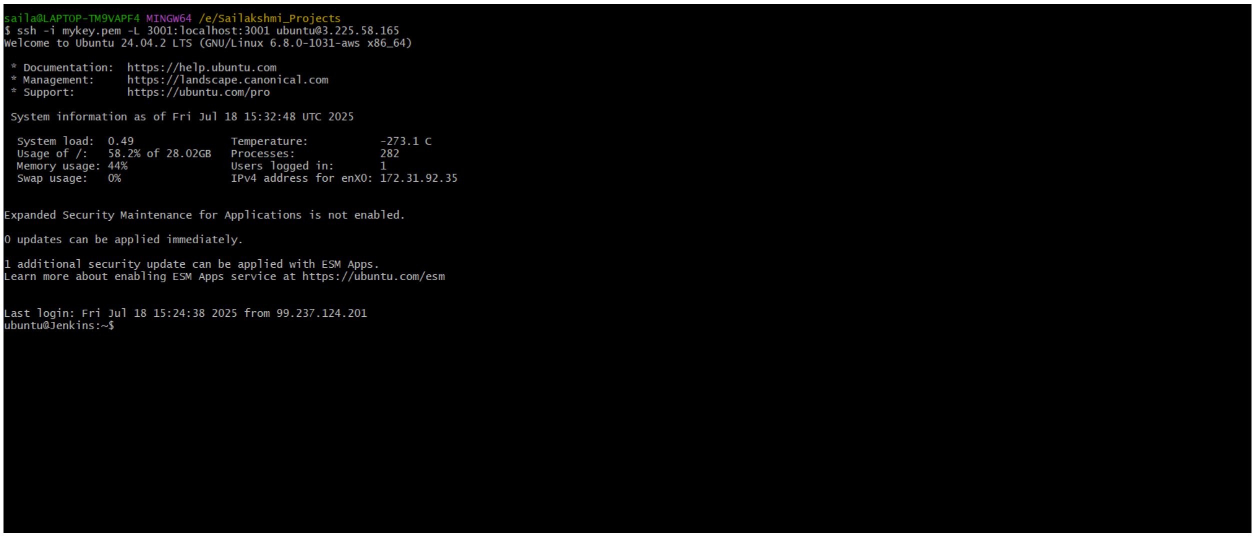
PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35



```
#################################################################################
ubuntu@Jenkins:~$ export POD_NAME=$(kubectl get pods --namespace monitoring -l "app.kubernetes.io/name=grafana,app.kubernetes.io/instance=grafana" -o jsonpath="{.items[0].metadata.name}")

kubectl --namespace monitoring port-forward $POD_NAME 3001:3000
Forwarding from 127.0.0.1:3001 -> 3000
Forwarding from [::1]:3001 -> 3000
Handling connection for 3001
Handling connection for 3001
Handling connection for 3001
Handling connection for 3001
Handling connection for 3001
Handling connection for 3001
```

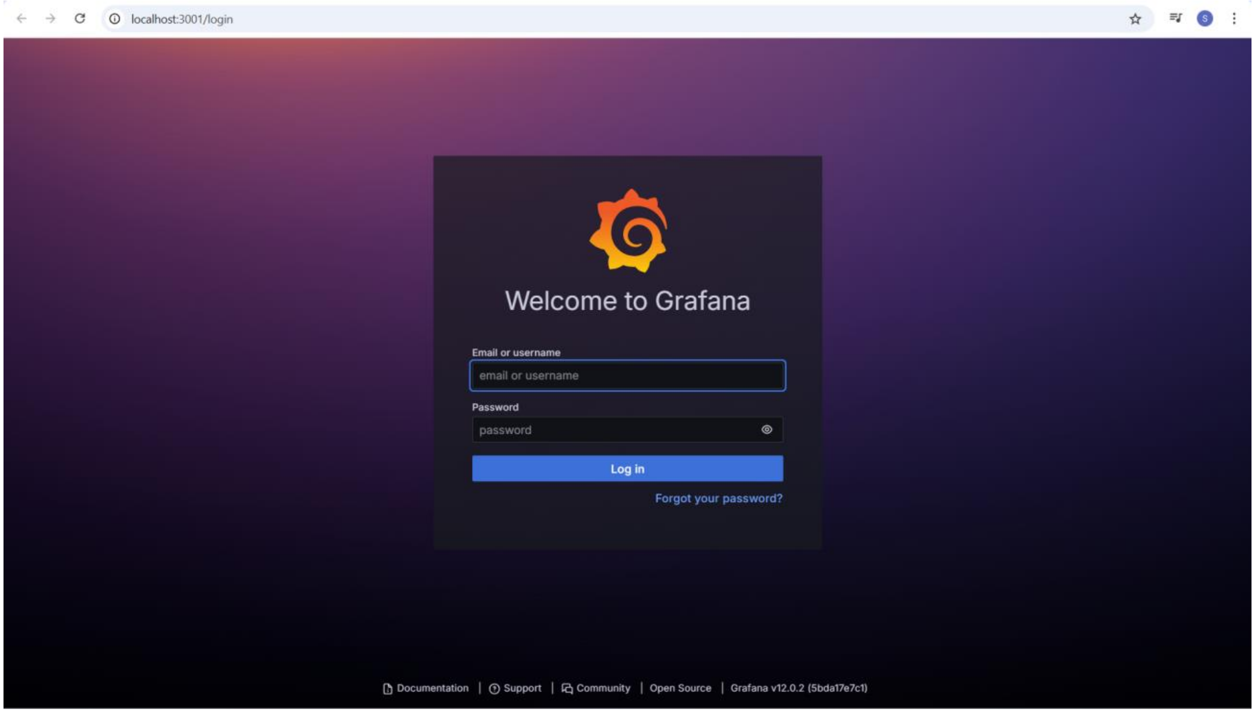
i-0508e635c1e3d6a98 (jenkin)

PublicIPs: 3.225.58.165    PrivateIPs: 172.31.92.35



```
saila@LAPTOP-TM9VAPF4 MINGW64 /e/Sailakshmi_Projects
$ ssh -i mykey.pem -L 3001:localhost:3001 ubuntu@3.225.58.165
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1031-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Jul 18 15:32:48 UTC 2025

  System load:  0.49               Temperature:           -273.1 C
  Usage of /:   58.2% of 28.02GB   Processes:             282
  Memory usage: 44%                Users logged in:       1
  Swap usage:   0%                 IPv4 address for enX0: 172.31.92.35


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


Last login: Fri Jul 18 15:24:38 2025 from 99.237.124.201
ubuntu@Jenkins:~$
```

http://localhost:3001



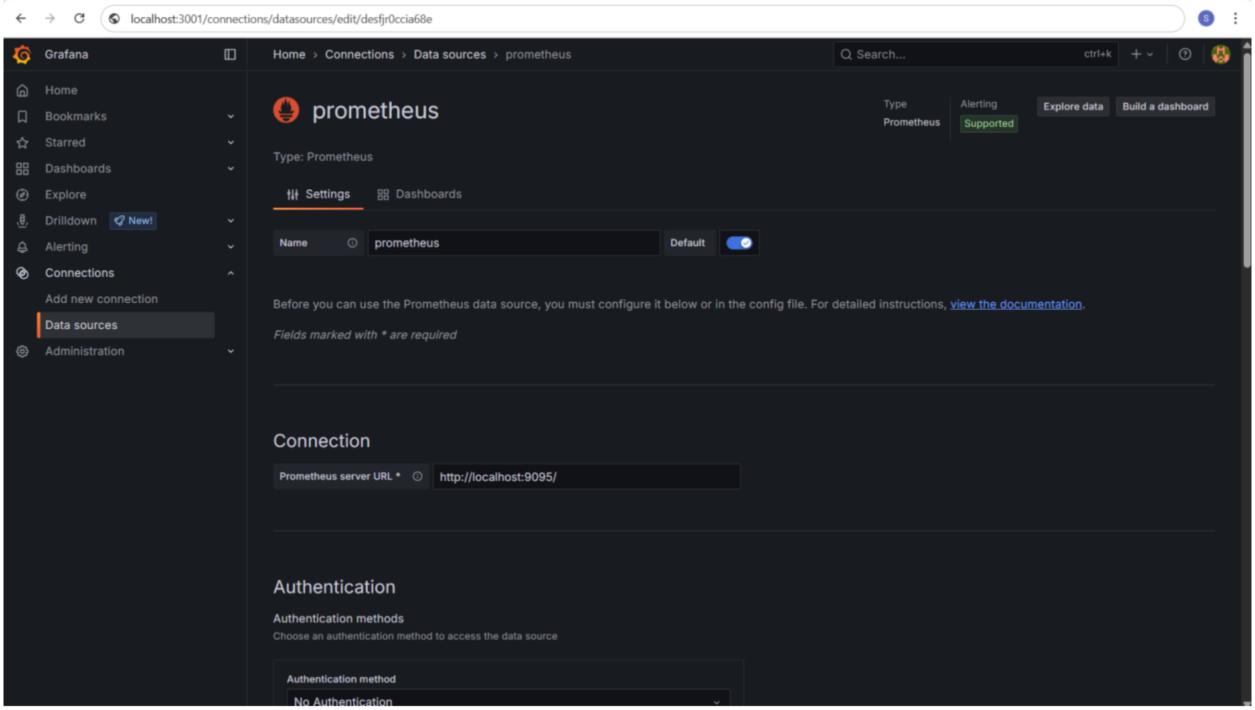
localhost:3001/login

Welcome to Grafana

Email or username: email or username

Password: password

Log in
Forgot your password?
Documentation | Support | Community | Open Source | Grafana v12.0.2 (5bda17e7c1)



localhost:3001/connections/datasources/edit/desfjr0ccia68e

Home › Connections › Data sources › prometheus

prometheus

Type: Prometheus

Type Prometheus | Alerting Supported

Name: prometheus | Default: on

Before you can use the Prometheus data source, you must configure it below or in the config file. For detailed instructions, view the documentation.

*Fields marked with * are required*

Connection

Prometheus server URL *: http://localhost:9095/

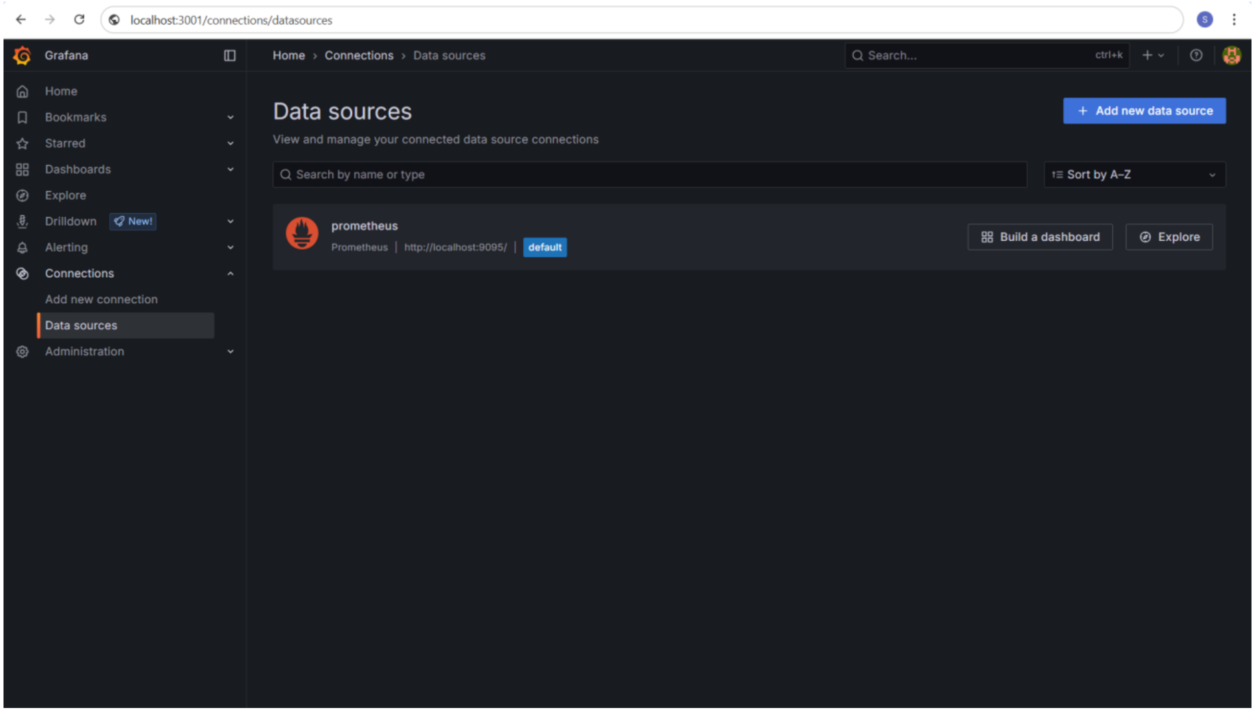
Authentication

Authentication methods

Choose an authentication method to access the data source

Authentication method: No Authentication

Save and test



Data sources

View and manage your connected data source connections

prometheus

Prometheus | http://localhost:9095/ | default