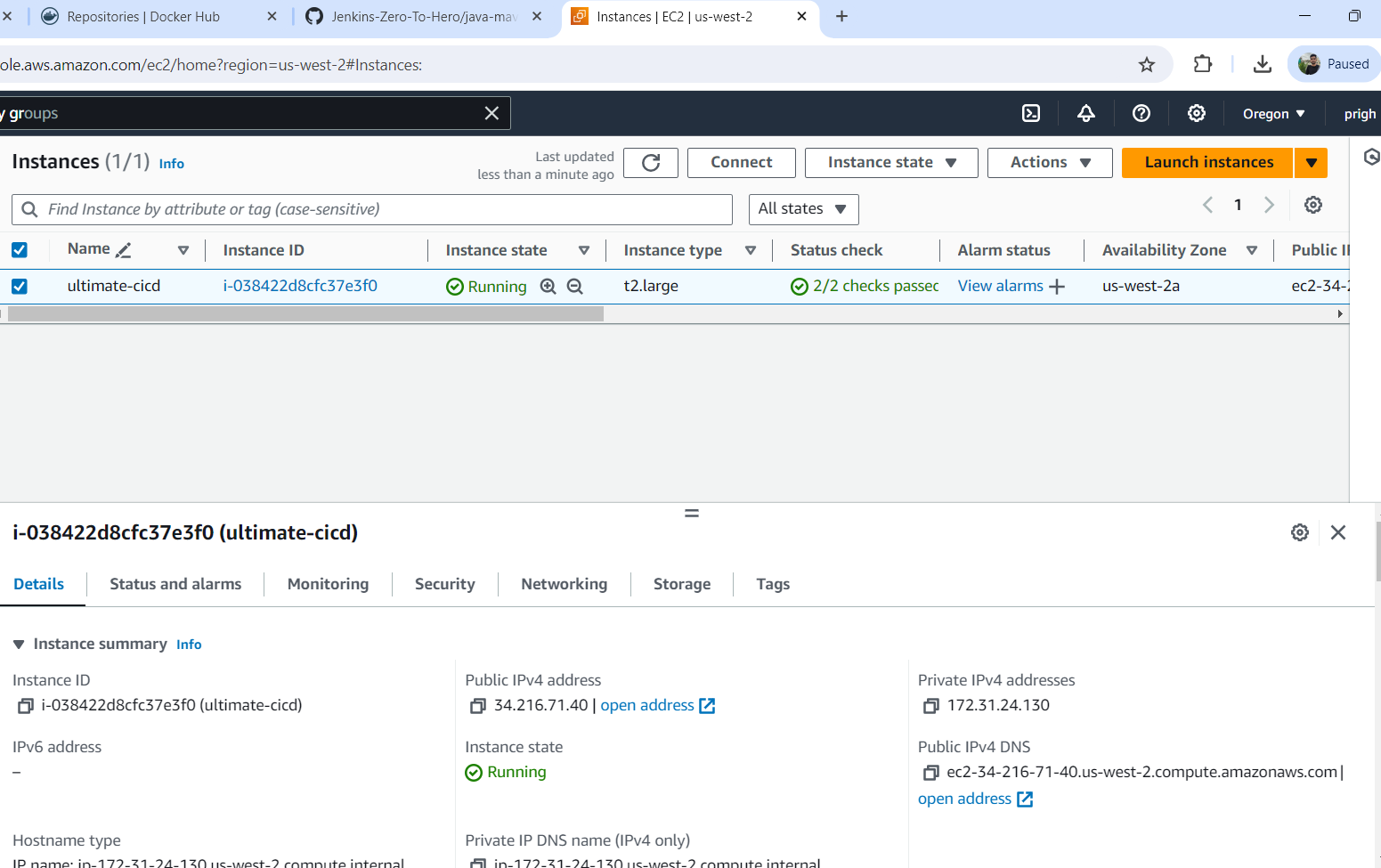
**Jenkins End to End CICD Pipeline Implementation**

1. Installation of ec2 instance

* Go to the aws console
* Create and launch the ec2 instance
* ultimate-cicd instance is running now.



1. Now I connected the ec2 instance on mobaXterm.
2. Install the Jenkins

Pre-Requisites: • Java (JDK) Run the below commands to install Java and Jenkins

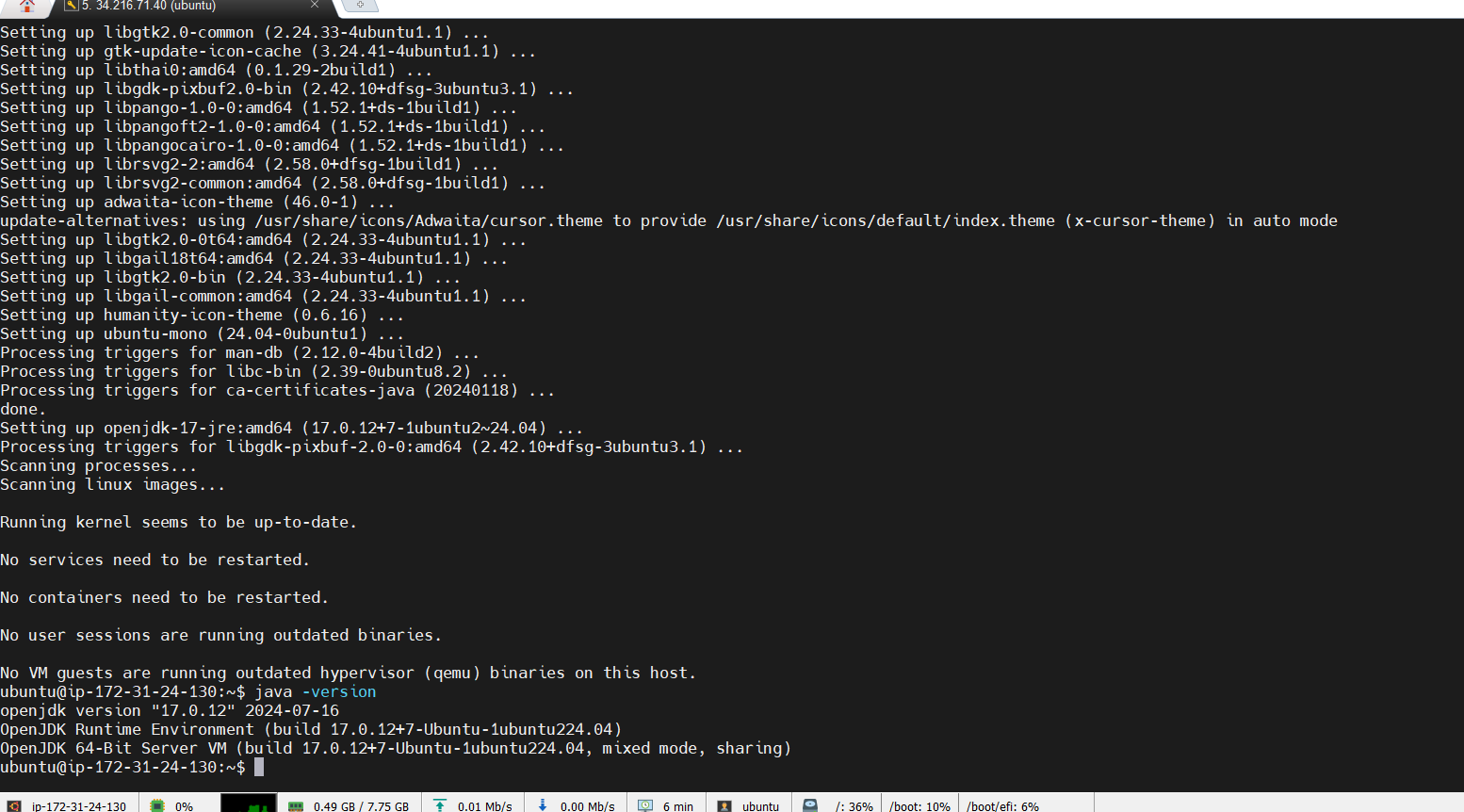
Install Java

*sudo apt update*

*sudo apt install openjdk-17-jre*

Verify Java is Installed

*java -version*



1. Now we can proceed to install Jenkins

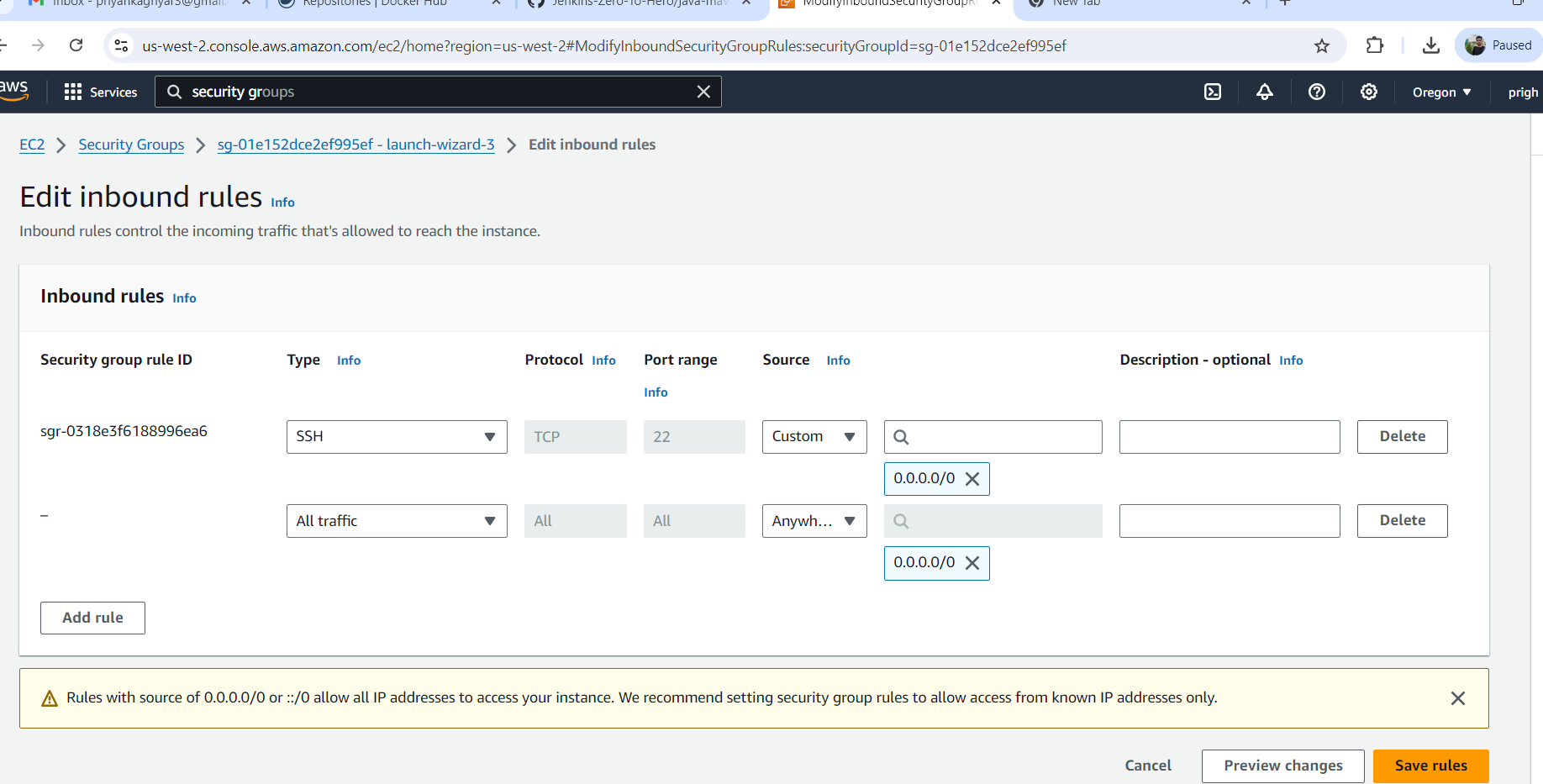
*curl -fsSL*[*https://pkg.jenkins.io/debian/jenkins.io-2023.key*](https://pkg.jenkins.io/debian/jenkins.io-2023.key)*| sudo tee  
/usr/share/keyrings/jenkins-keyring.asc > /dev/null echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]*[*https://pkg.jenkins.io/debian*](https://pkg.jenkins.io/debian)*binary/ | sudo tee  
/etc/apt/sources.list.d/jenkins.list > /dev/null*

*sudo apt-get update*

*sudo apt-get install jenkins*

1. Jenkins is installed but it will not allow inbound traffic to the external due to restriction of aws. So, we need to allow the traffic

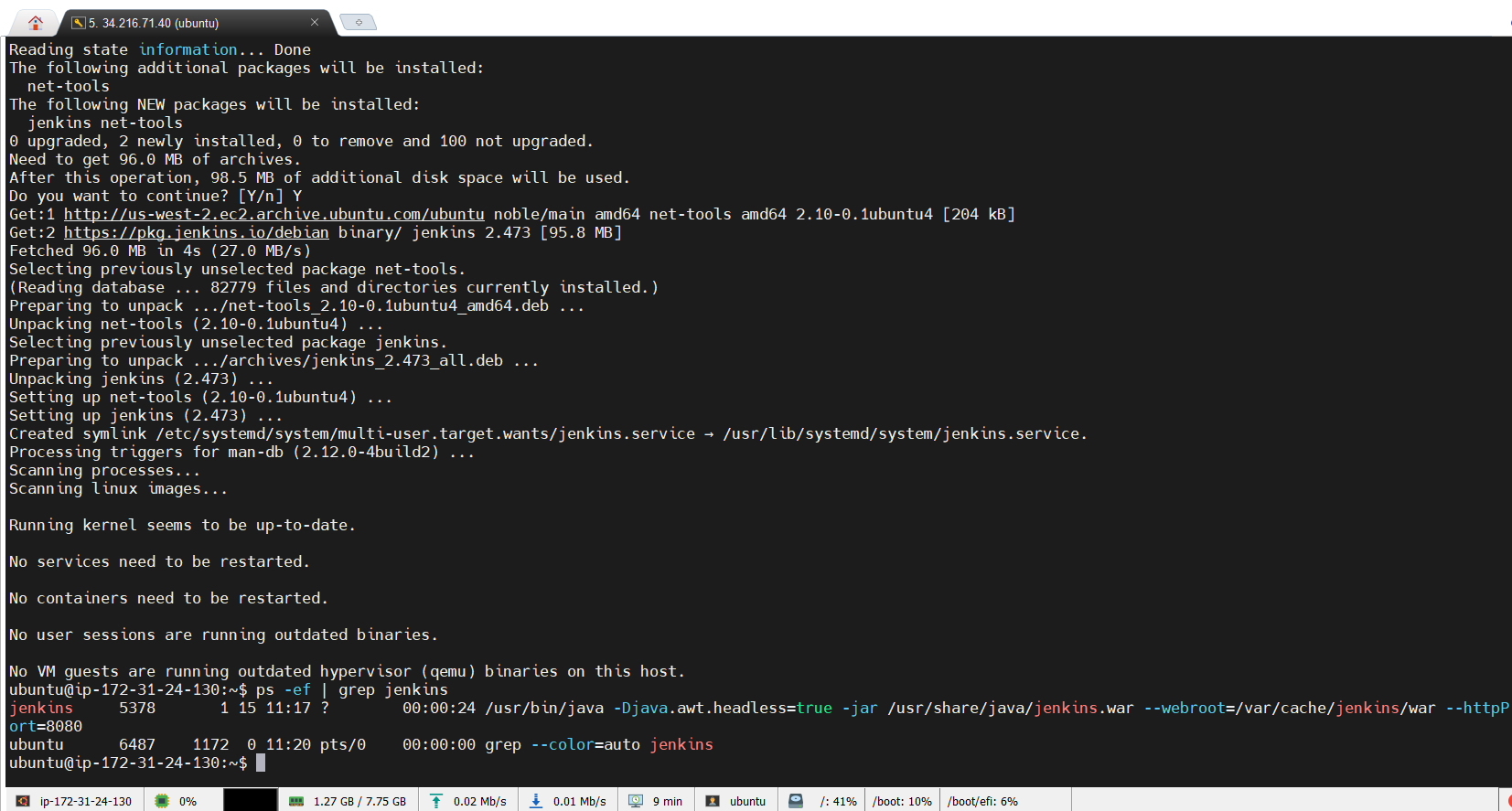
* Go to the running ec2 instance
* Go to the below tab security group -> click on that group name -> add the inbound traffic rules.



1. Check the port of jenkins

*ps -ef | grep Jenkins*

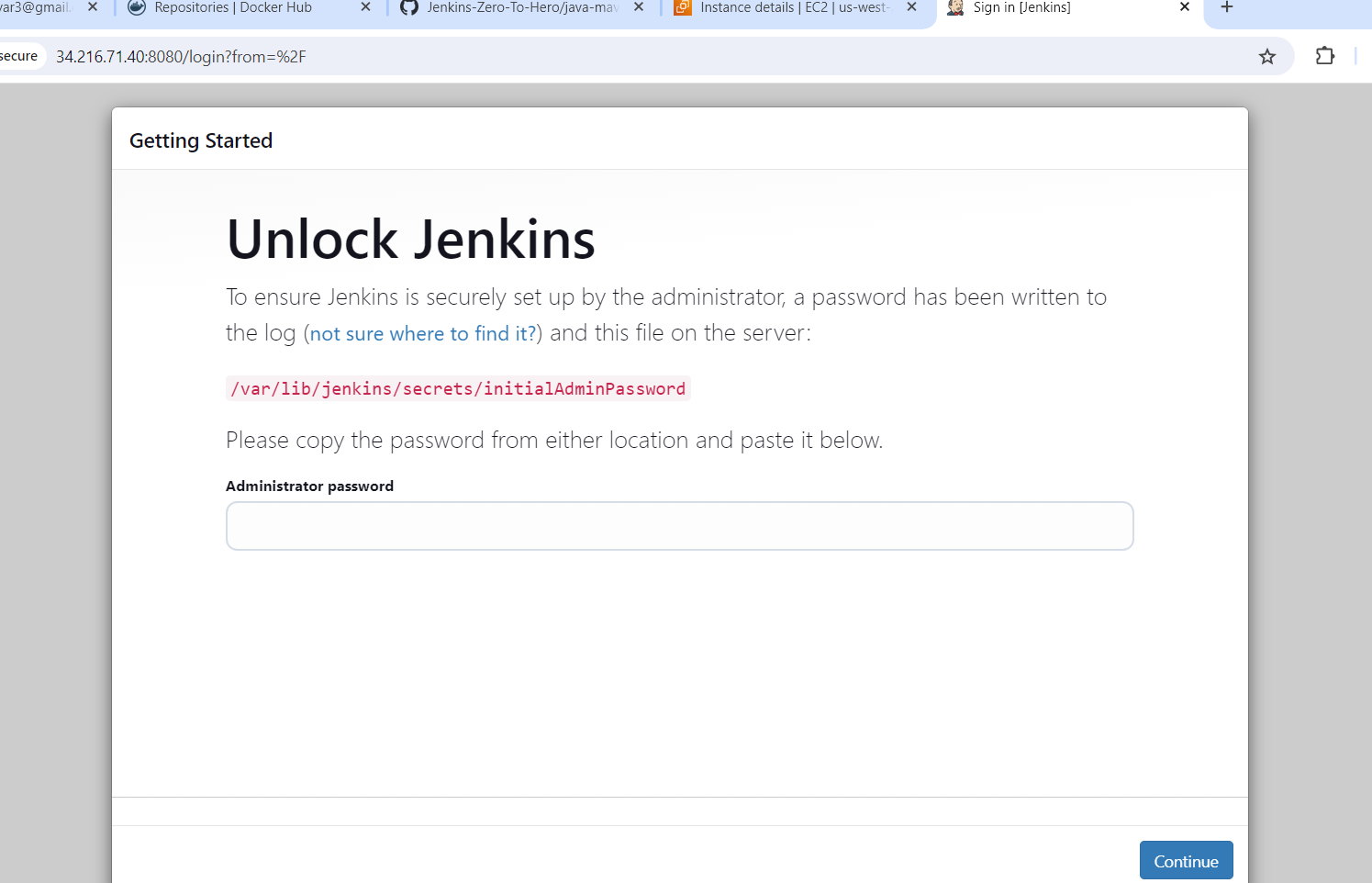
We can see the http port is 8080



1. Login the Jenkins

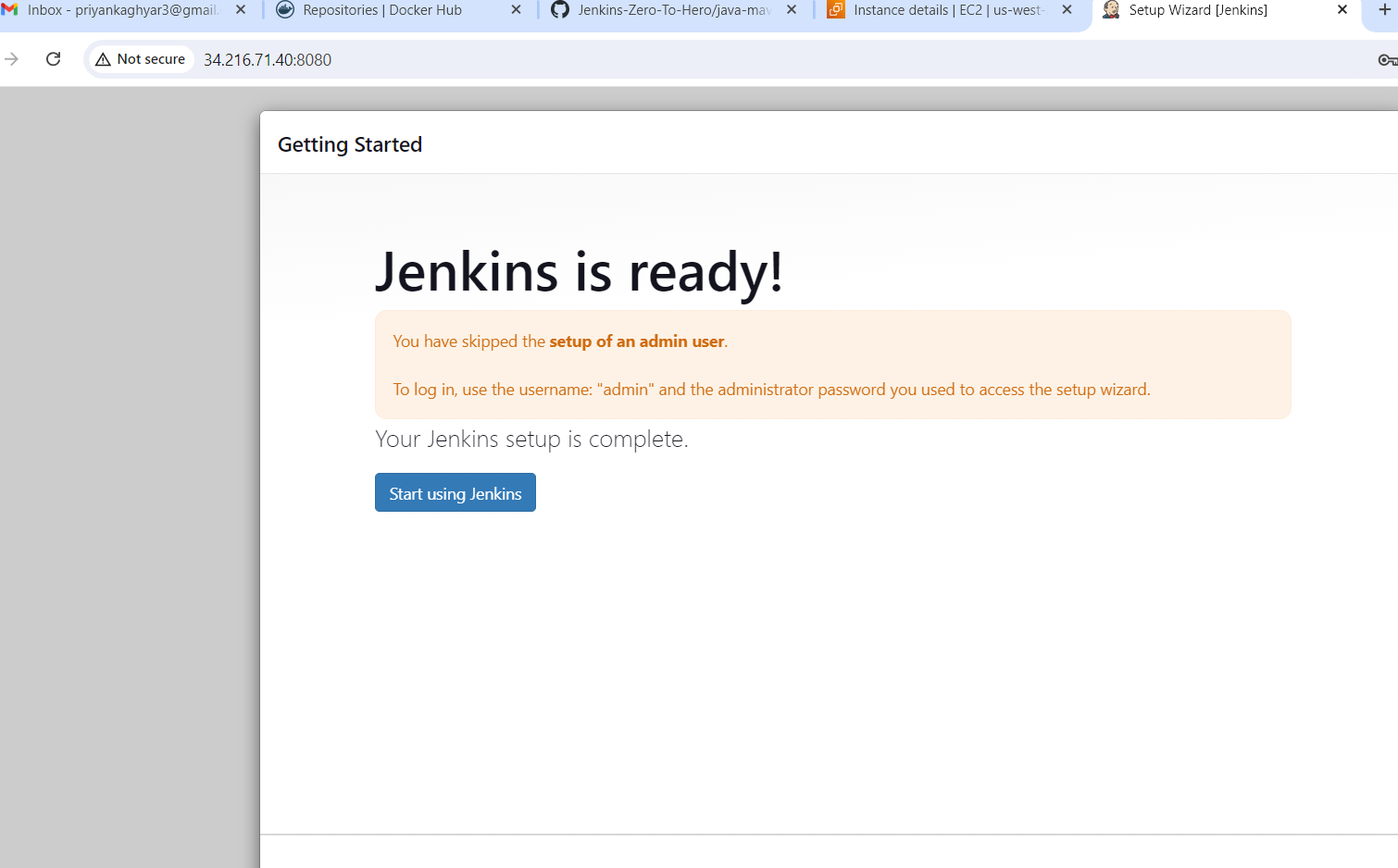
* <ec2instance Ip address>:8080

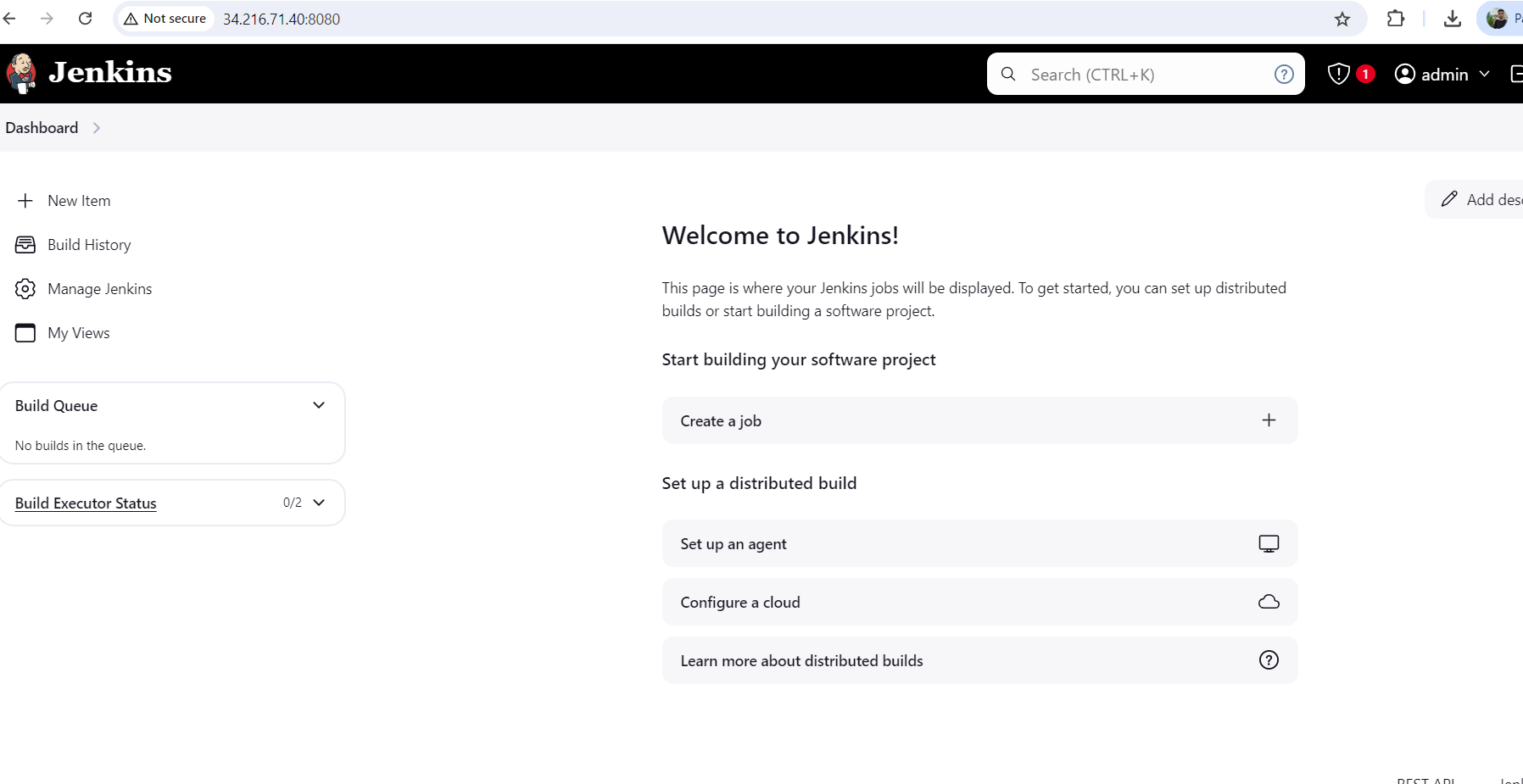
After login -> to know the password use the command. *sudo cat <path>*





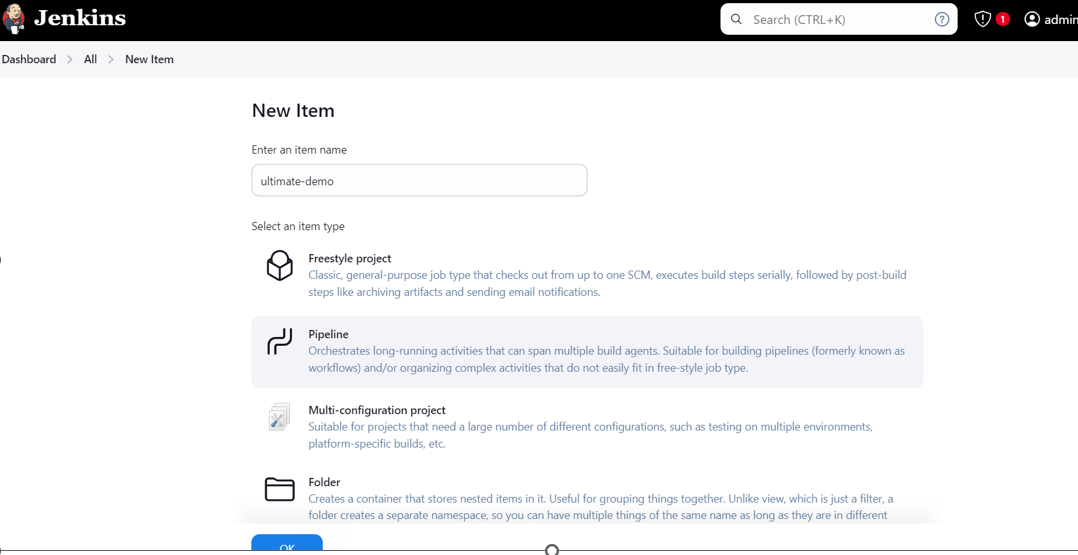
Jenkins is ready now 😊

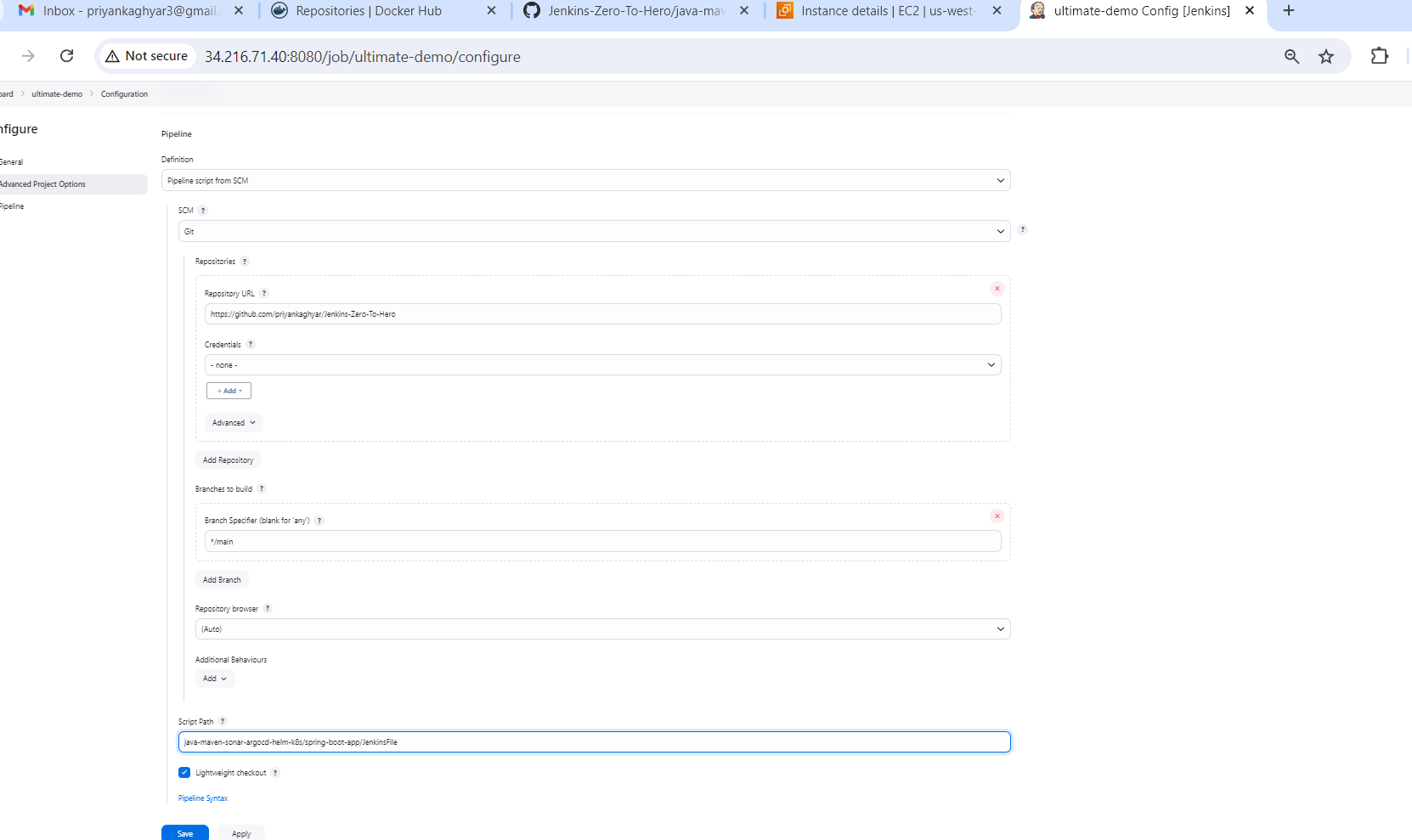




1. Click on new item. And configure the new item -> *ultimate-demo*

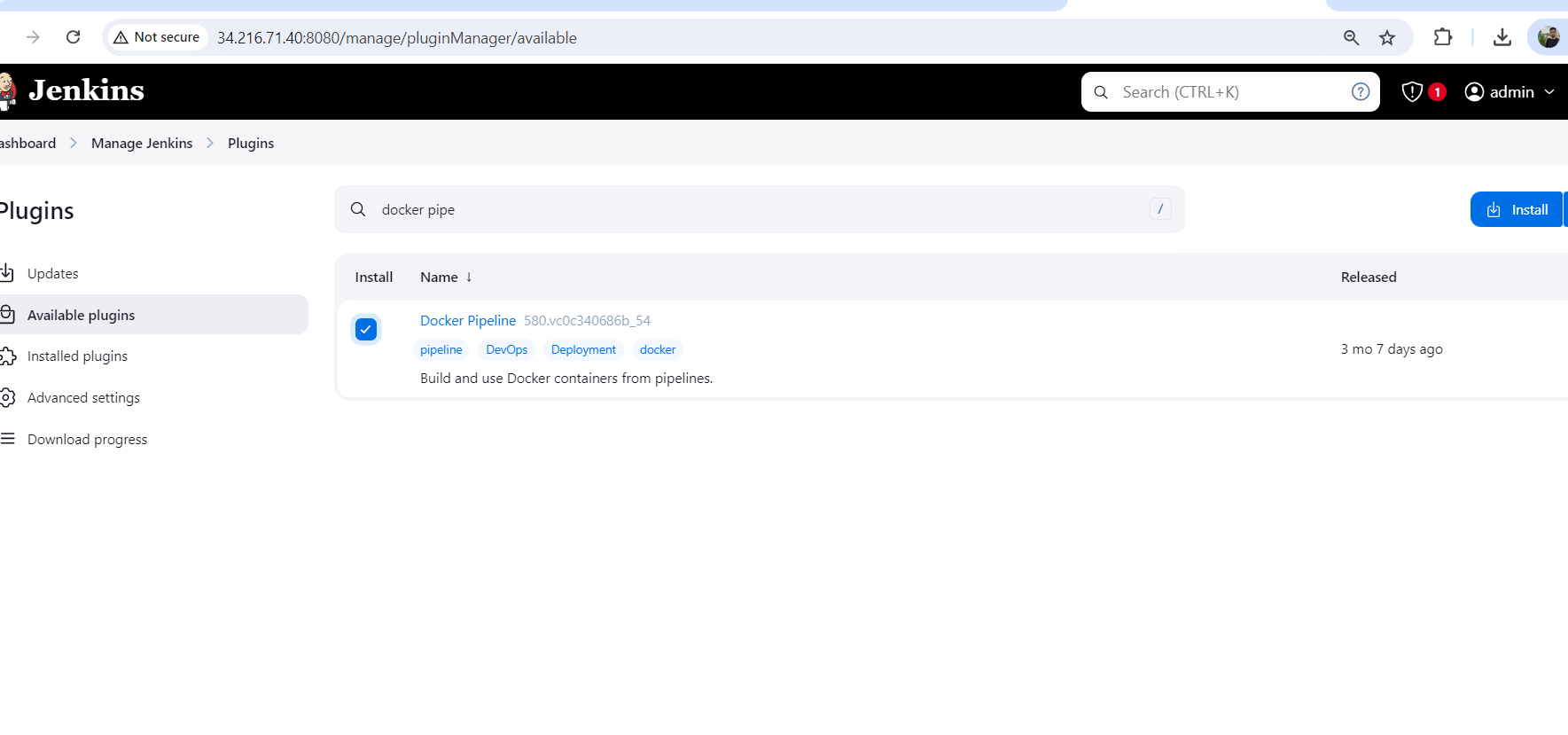
* Configure the pipeline
* Add definition, repository URL, branch, script-path and apply.

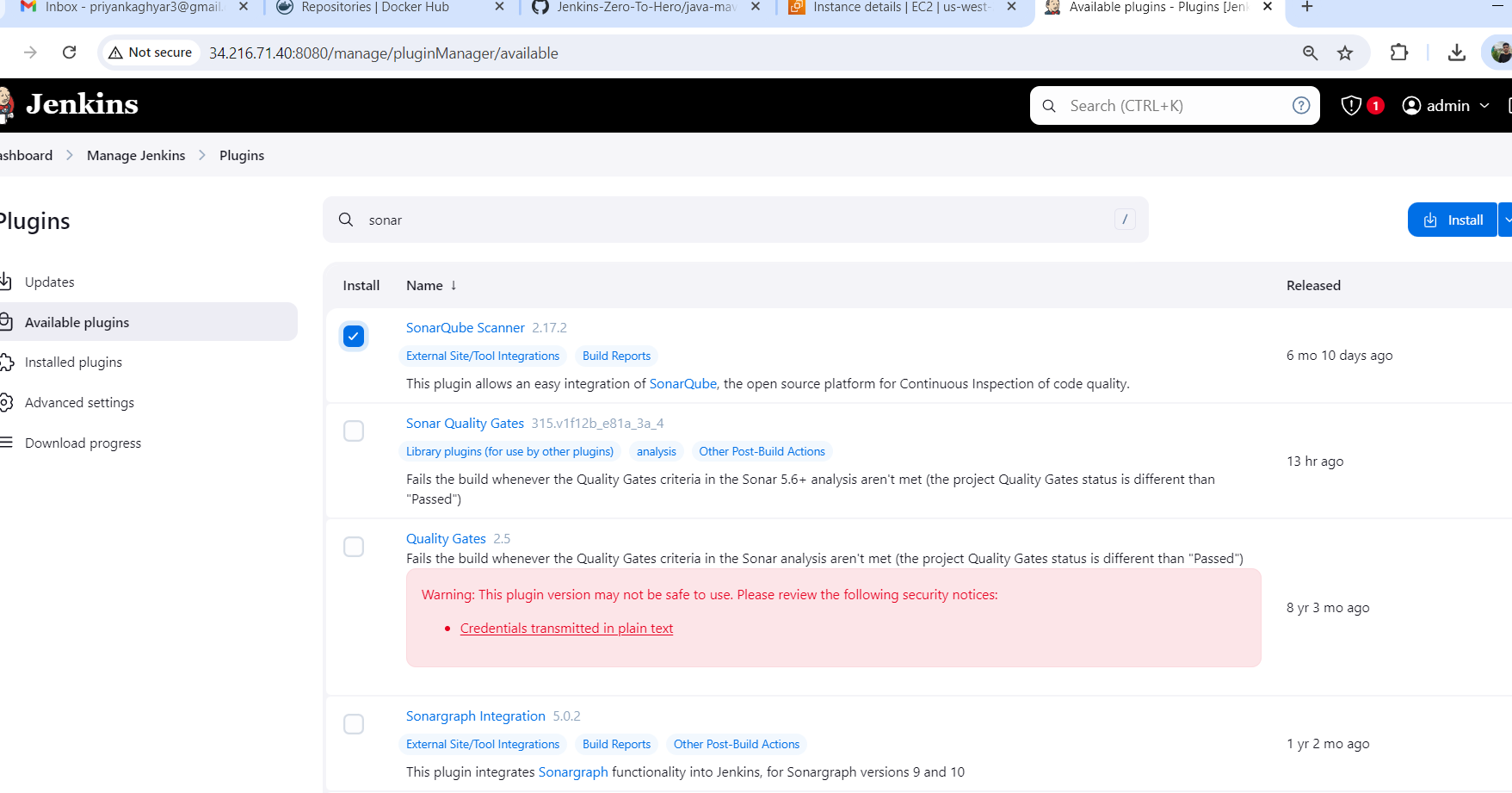




1. Manage the plugin

* Add Docker plugins
* Add Sonarqube Plugins





1. Now Configure the Sonar Server locally

*apt install unzip*

*adduser sonarqube*

*wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.4.0.54424.zip*

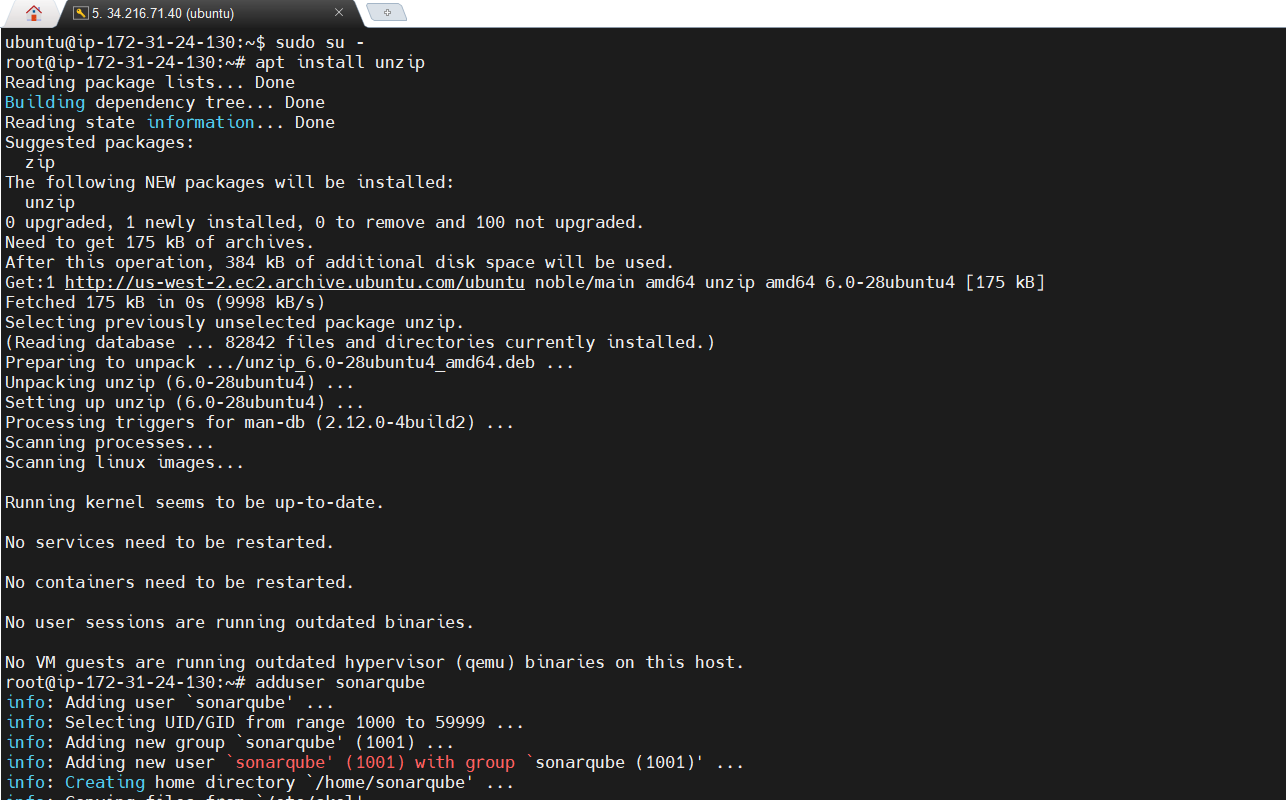
*unzip \**

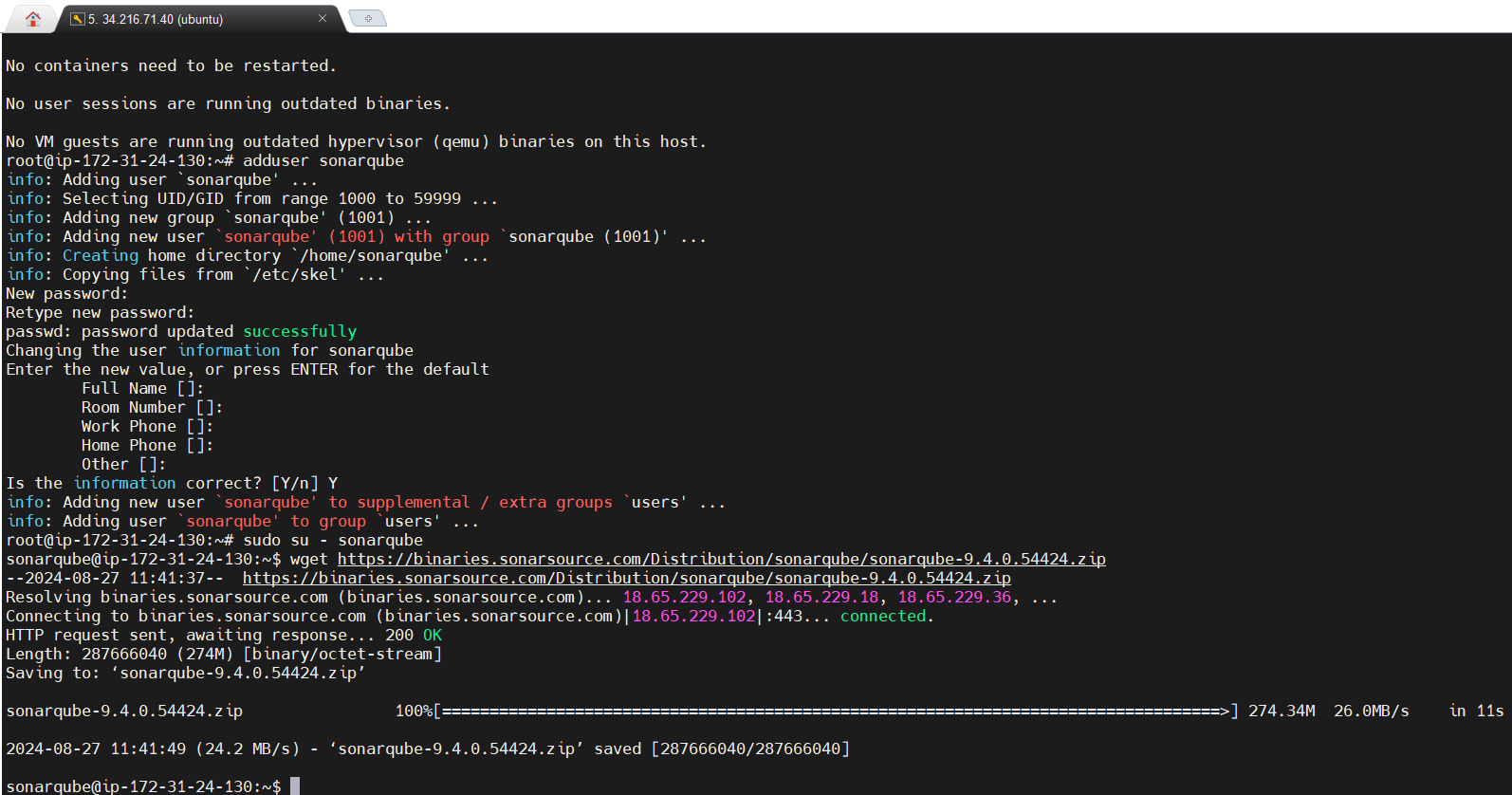
*chmod -R 755 /home/sonarqube/sonarqube-9.4.0.54424*

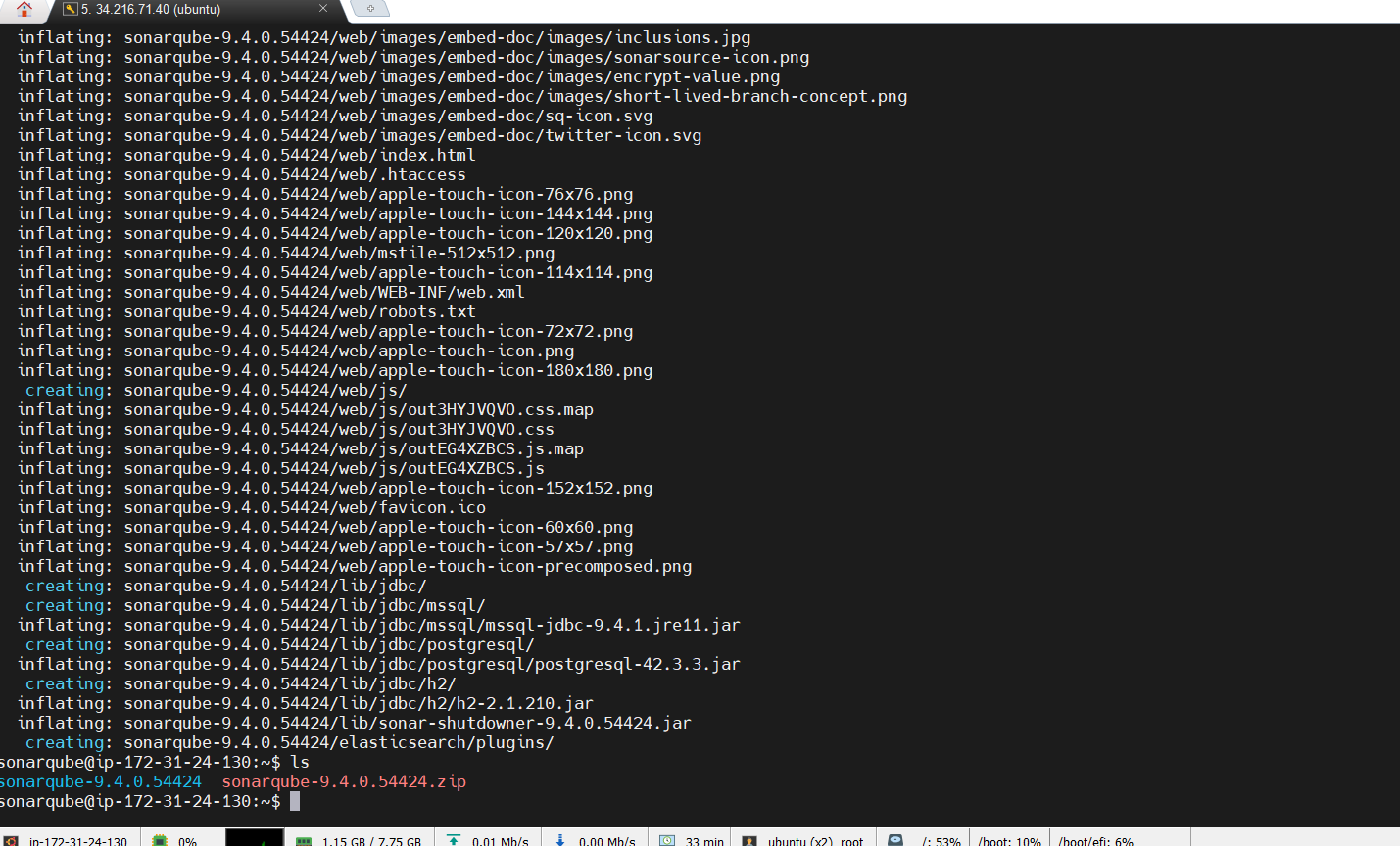
*chown -R sonarqube:sonarqube /home/sonarqube/sonarqube-9.4.0.54424*

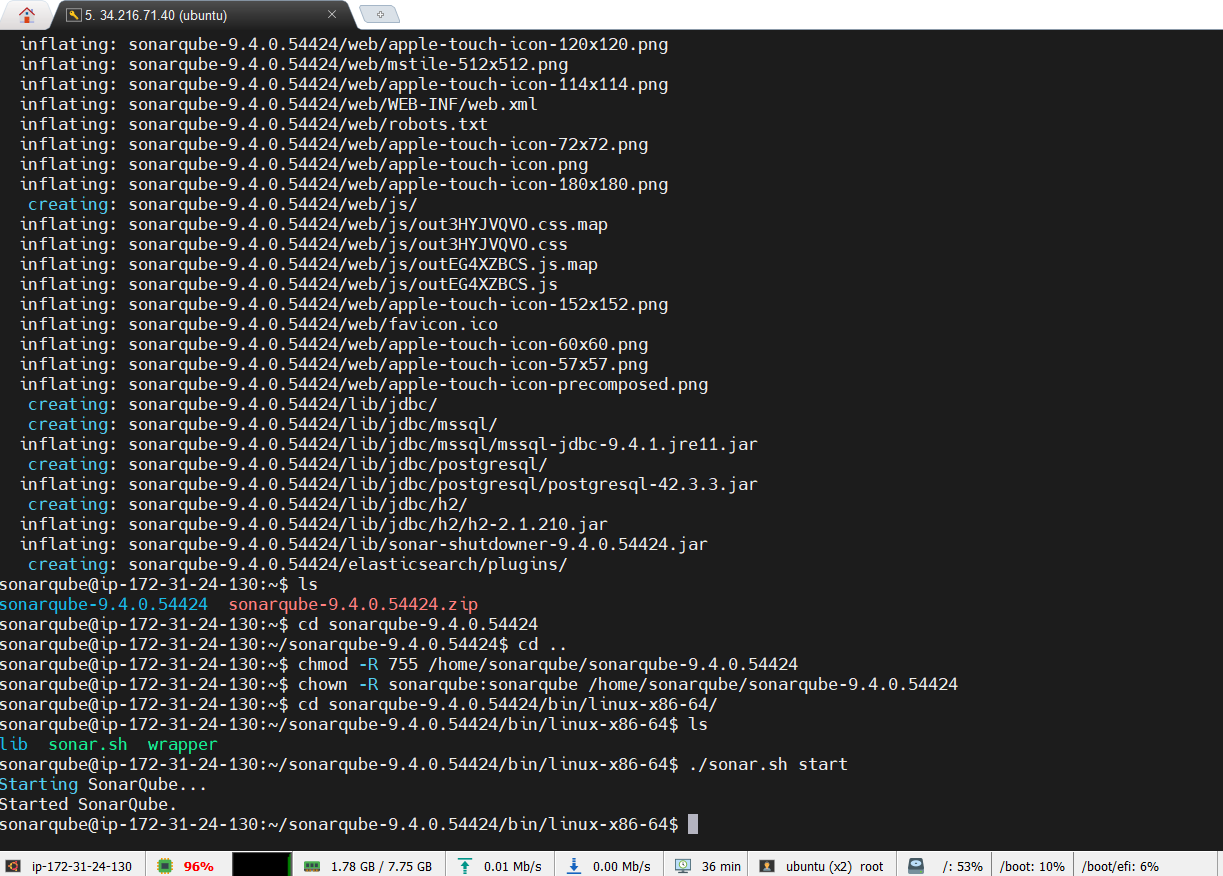
*cd sonarqube-9.4.0.54424/bin/linux-x86-64/*

*./sonar.sh start*

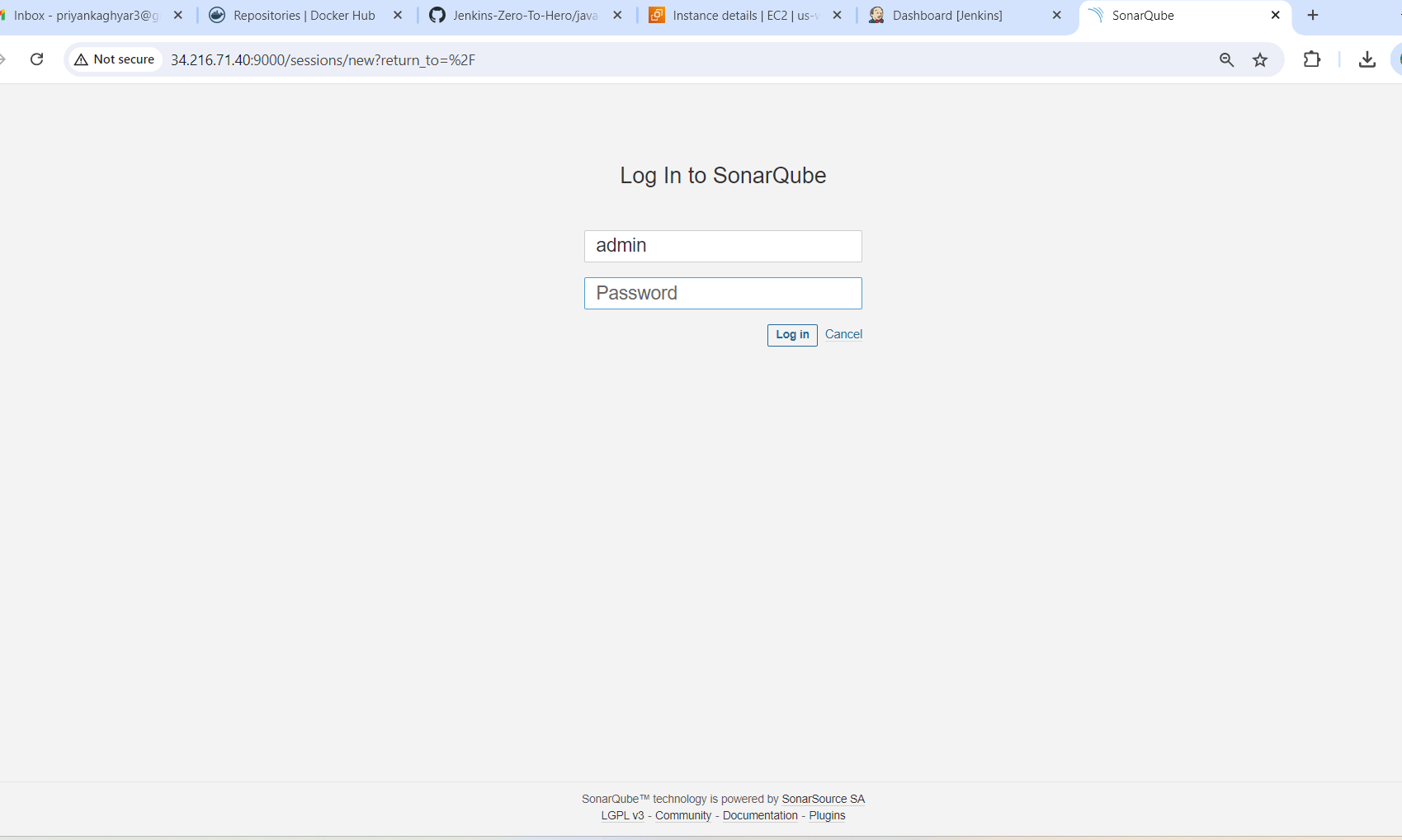




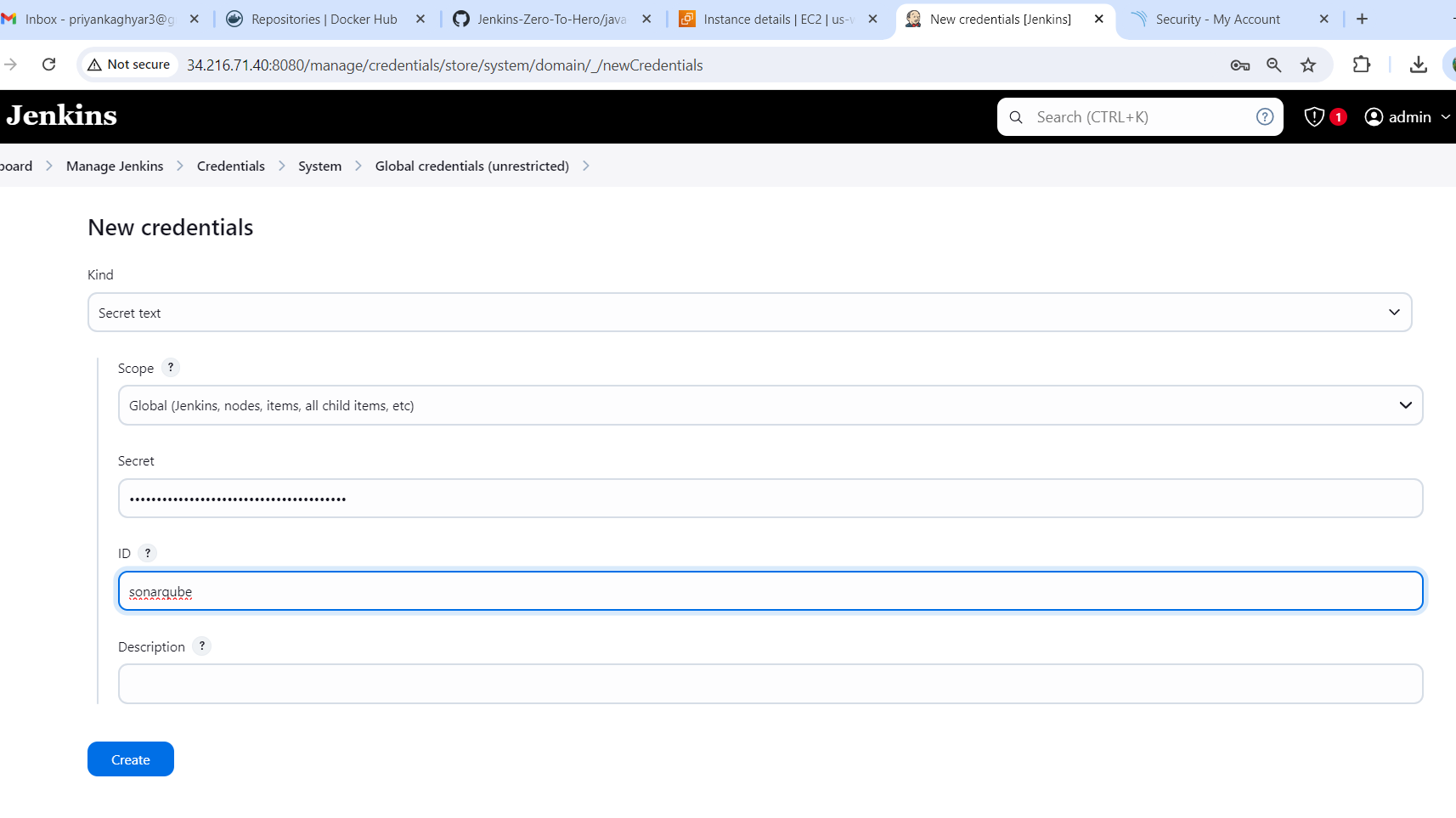




* SonarQube is started
* Login to SonarQube
* <ec2 ip>:9000
* Default username and password is admin



1. Generate the sonar token and add this token in Jenkins credential.



1. Install the docker agent on ec2 instance

*sudo apt update*

*sudo apt install docker.io*

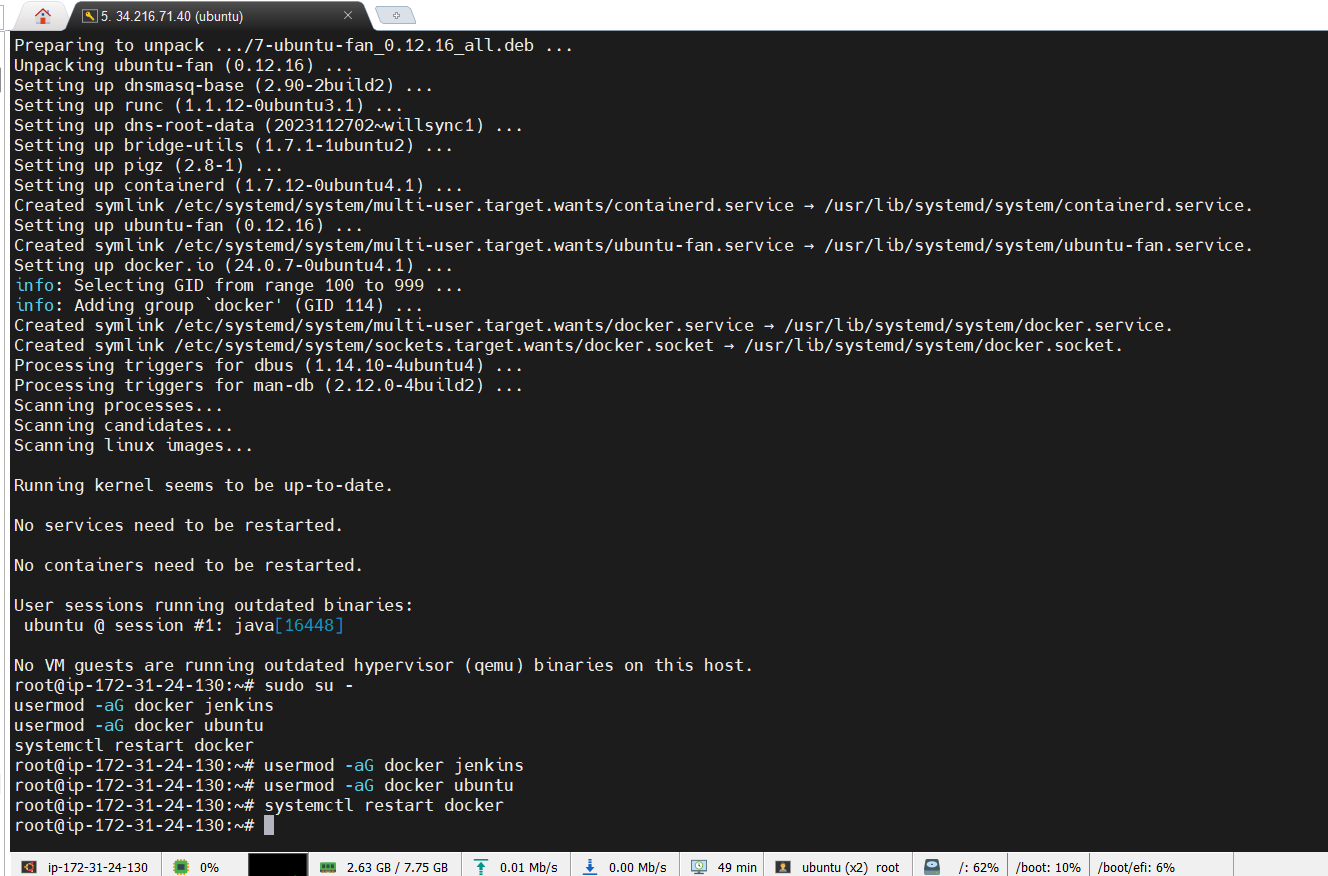
1. We need to grant the ubuntu and Jenkins user to docker daemon

*sudo su -*

*usermod -aG docker jenkins*

*usermod -aG docker ubuntu*

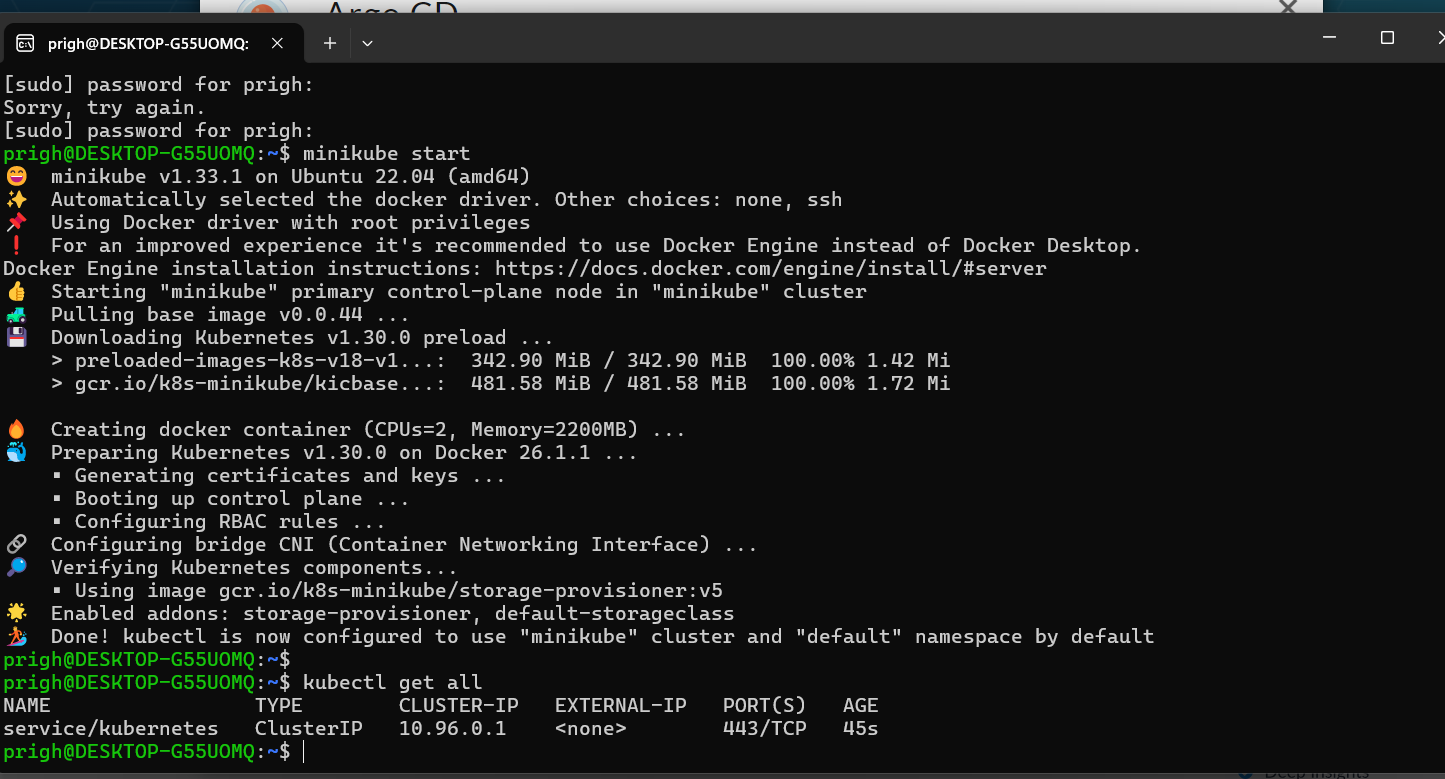
*systemctl restart docker*



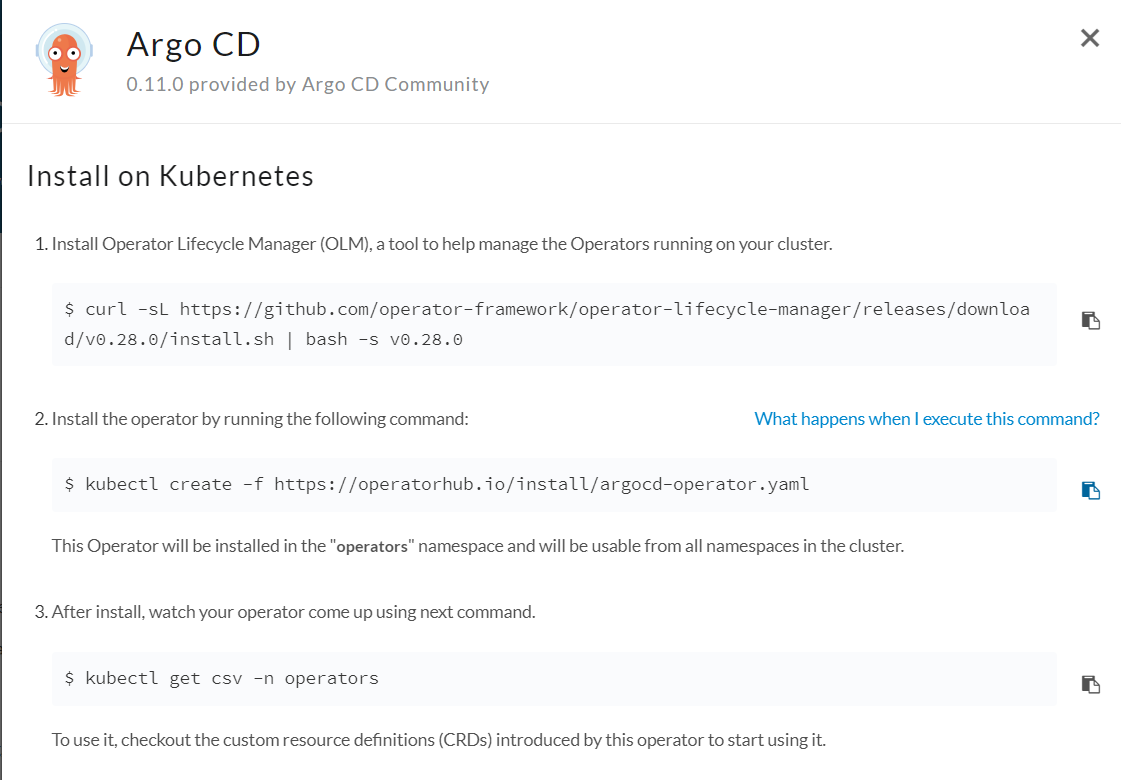
Restart Jenkins now

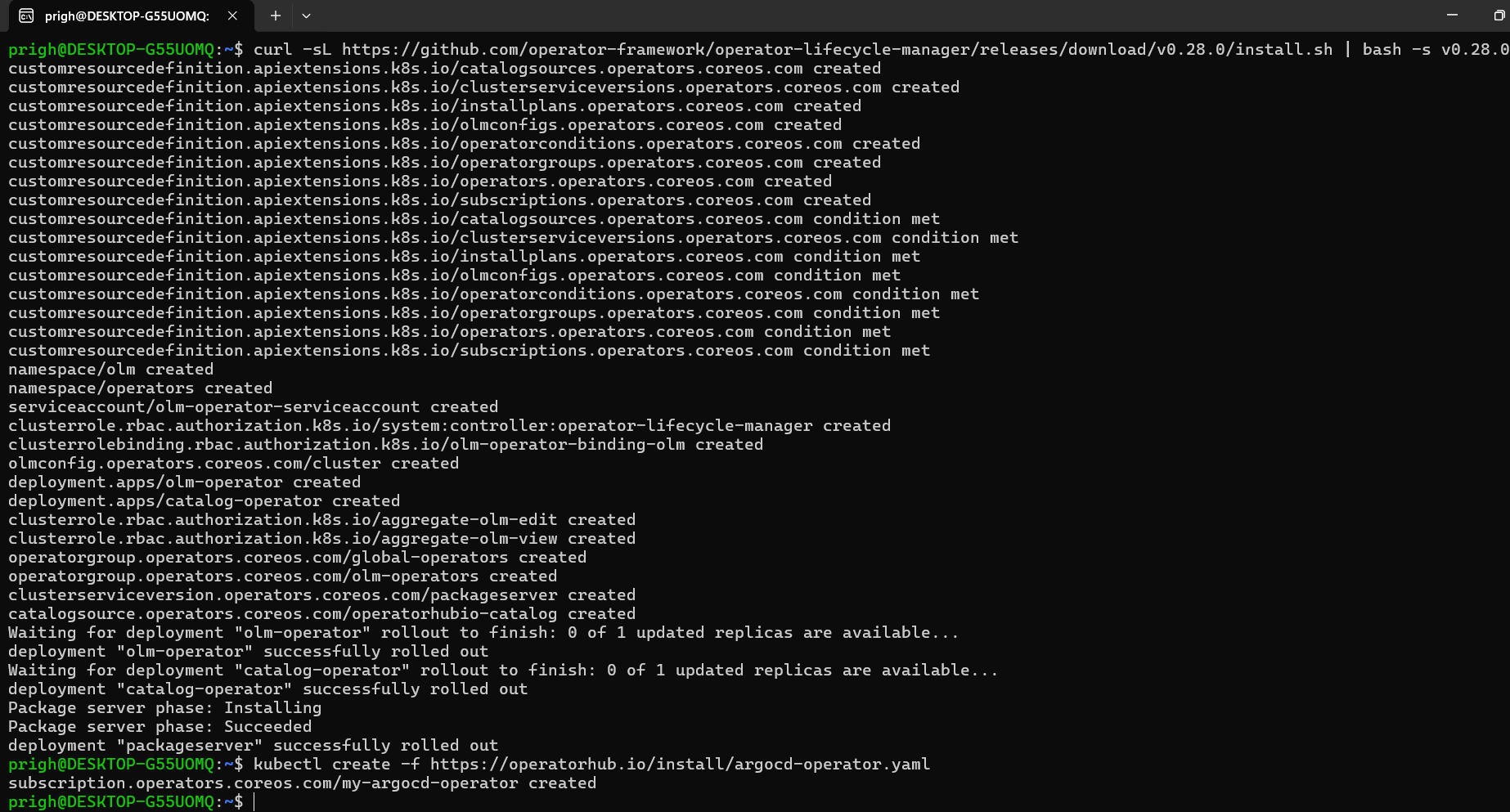
1. Install the minikube locally

* <https://minikube.sigs.k8s.io/docs/start/?arch=%2Fwindows%2Fx86-64%2Fstable%2F.exe+download>



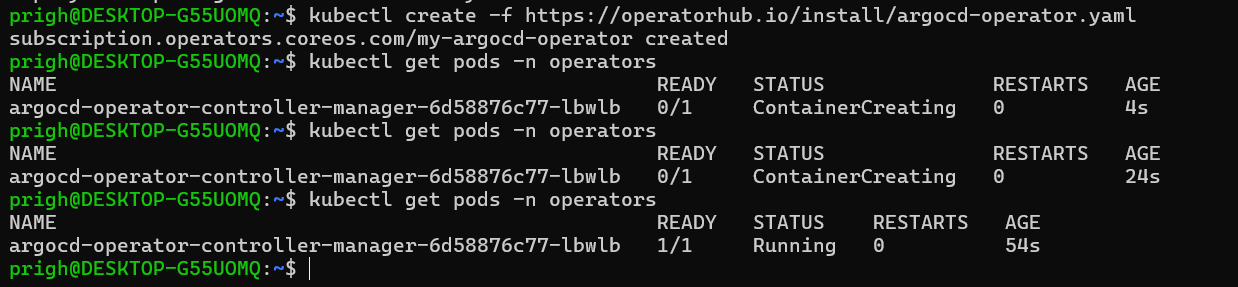
1. Install the argocd steps on operator hub site



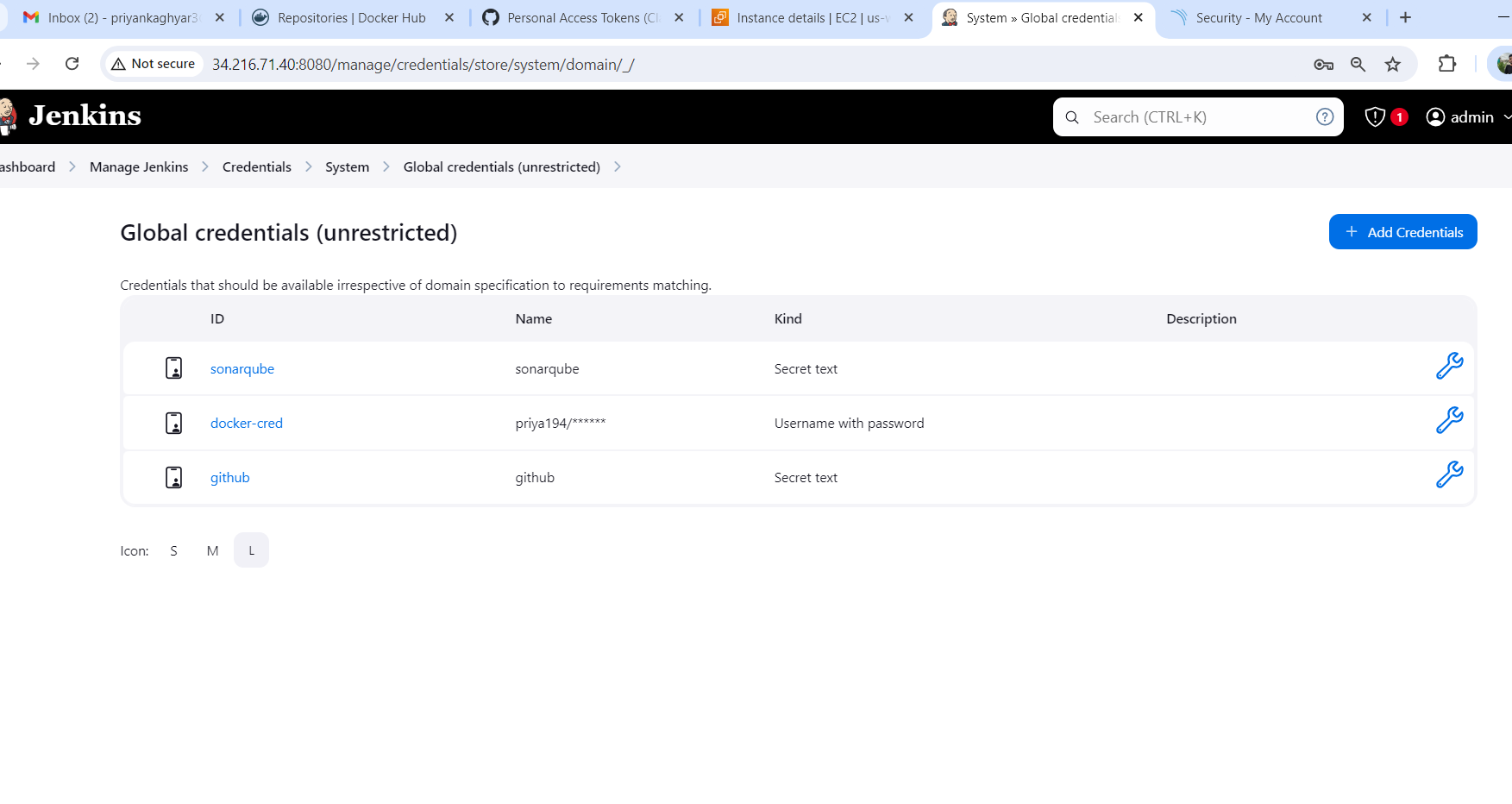


* To check the status of operator, use below command.

*kubectl get pods -n operators*

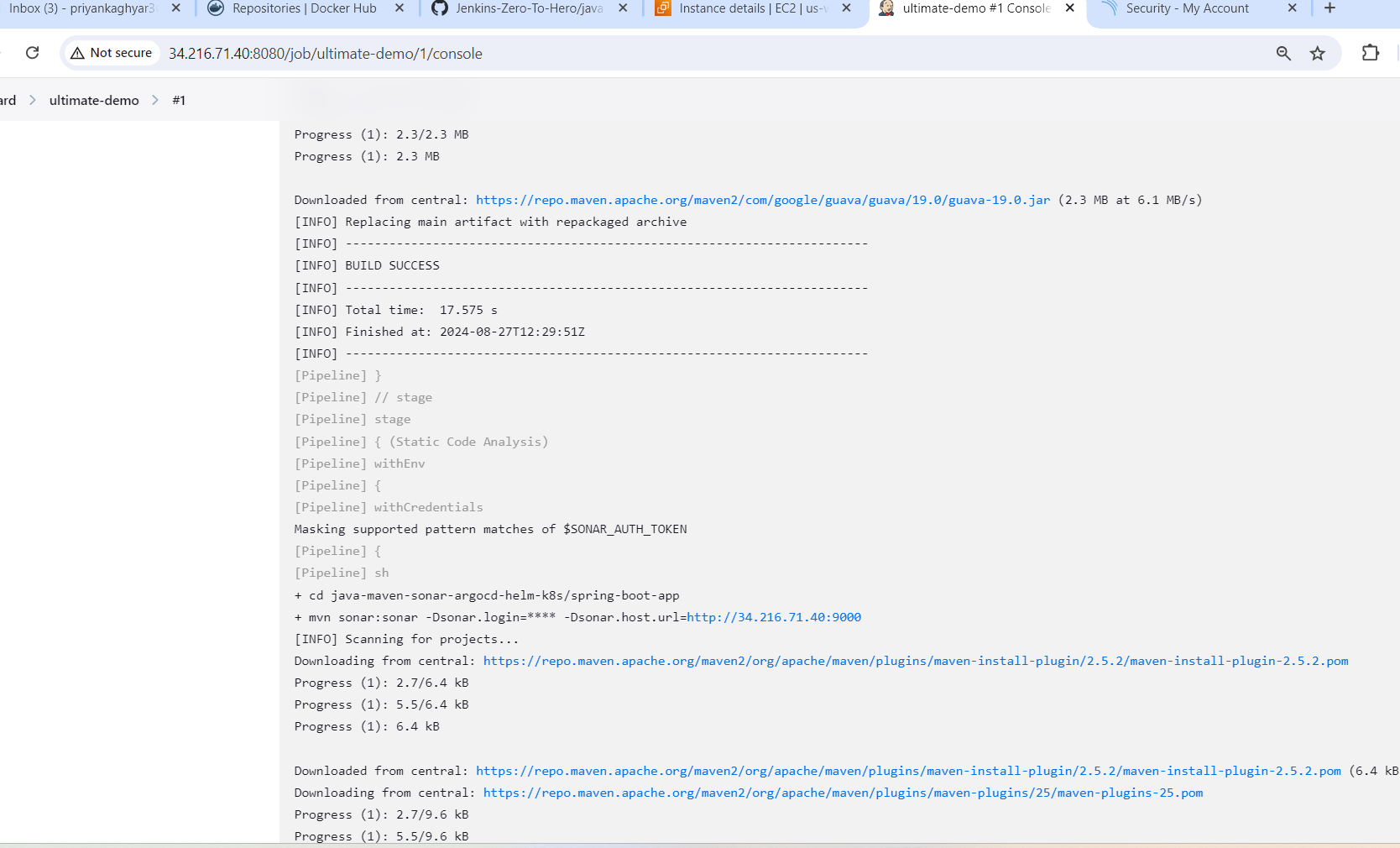


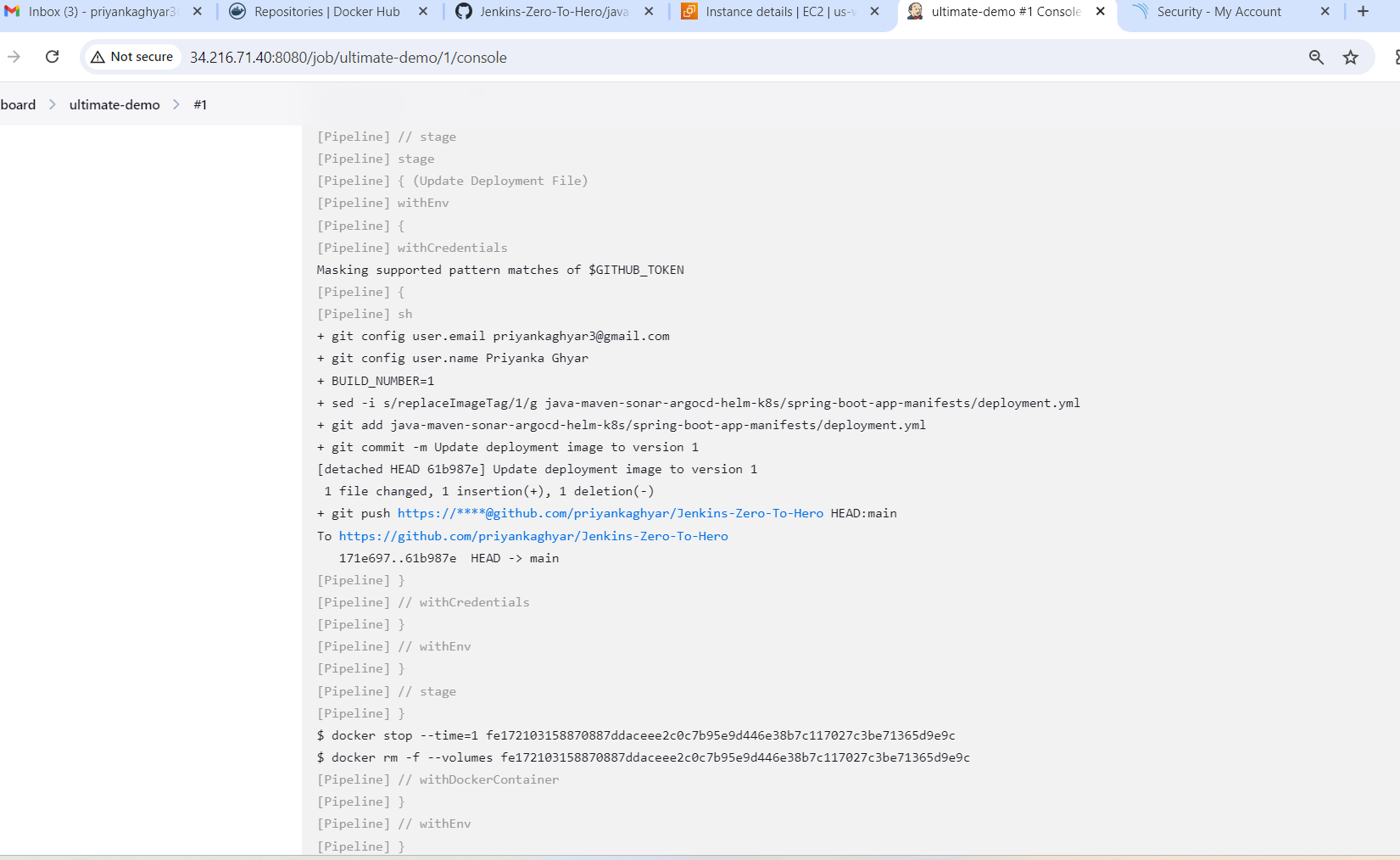
1. Added credentials on Jenkins: Docker and GitHub.



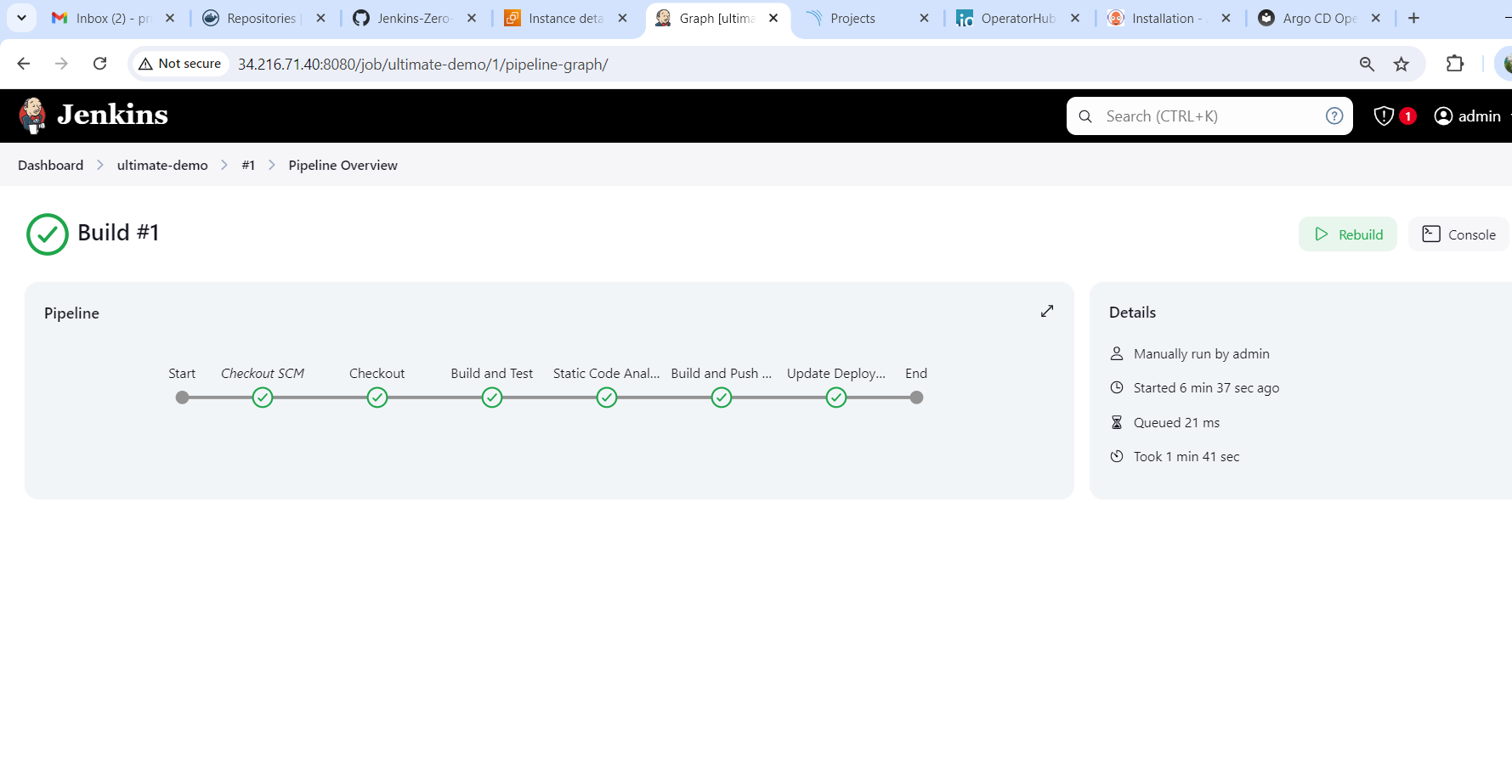
1. Restart the Jenkins and say build now 😊

* Pipeline is running it downloading dependencies, passing the stages.

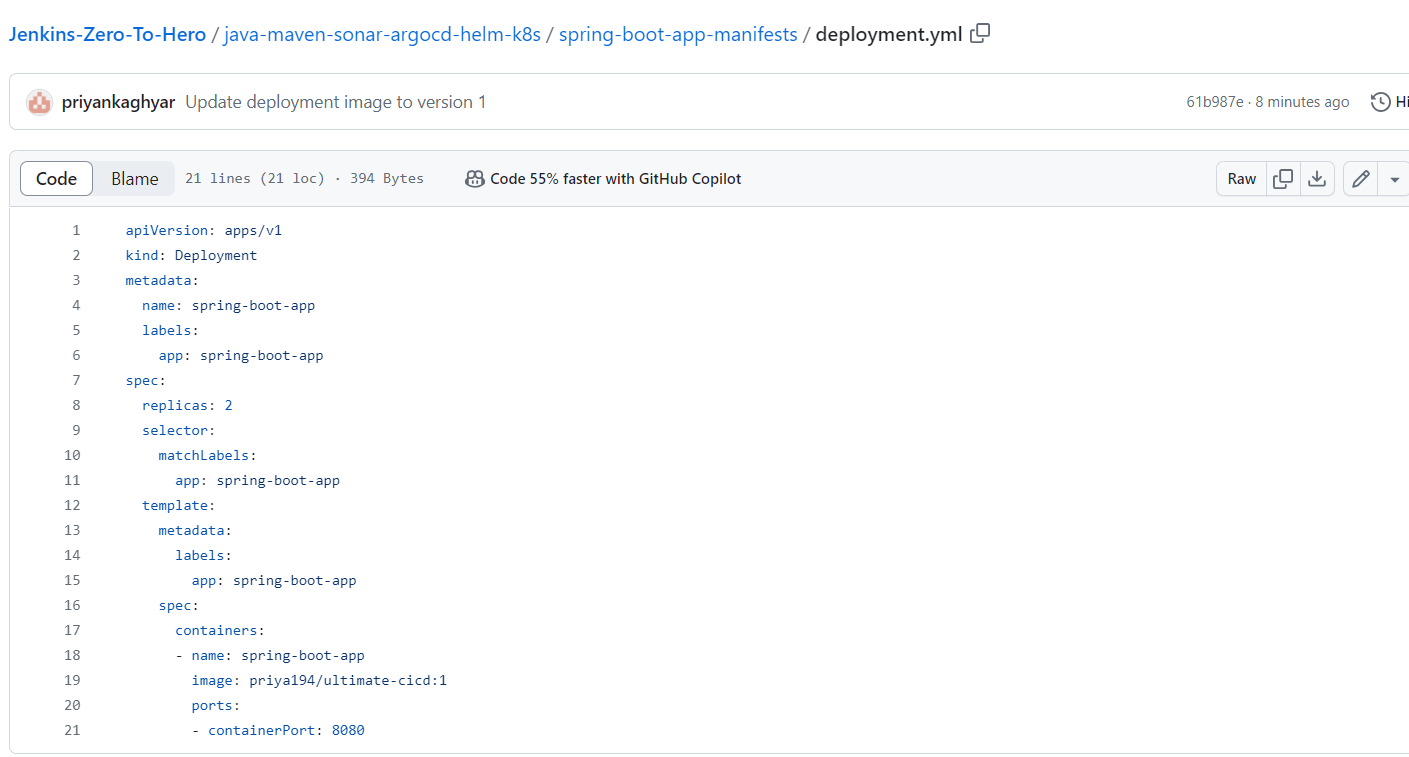
****

****

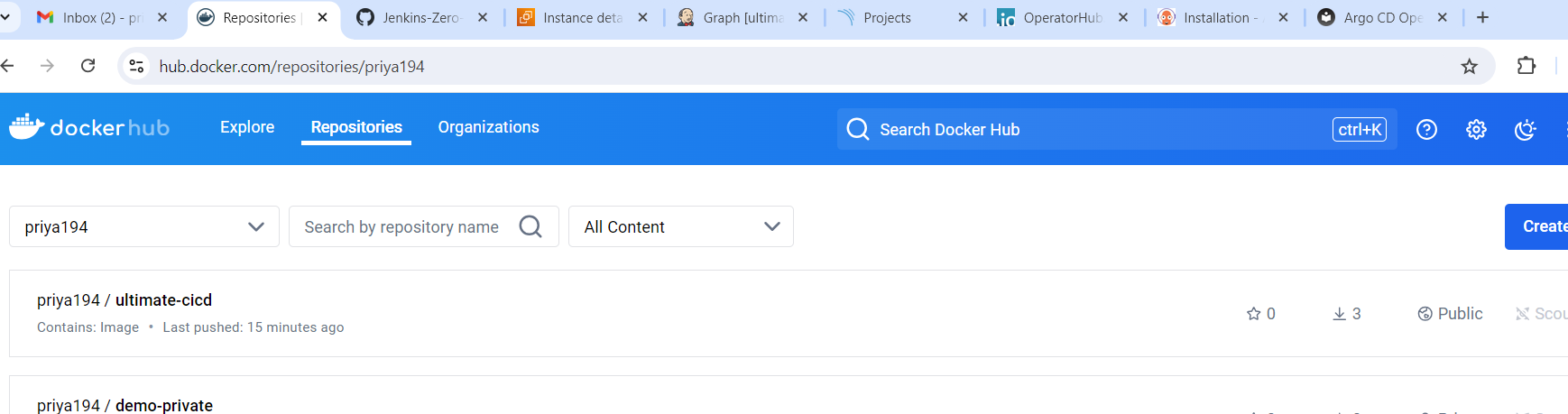
1. Stages of pipeline overview.



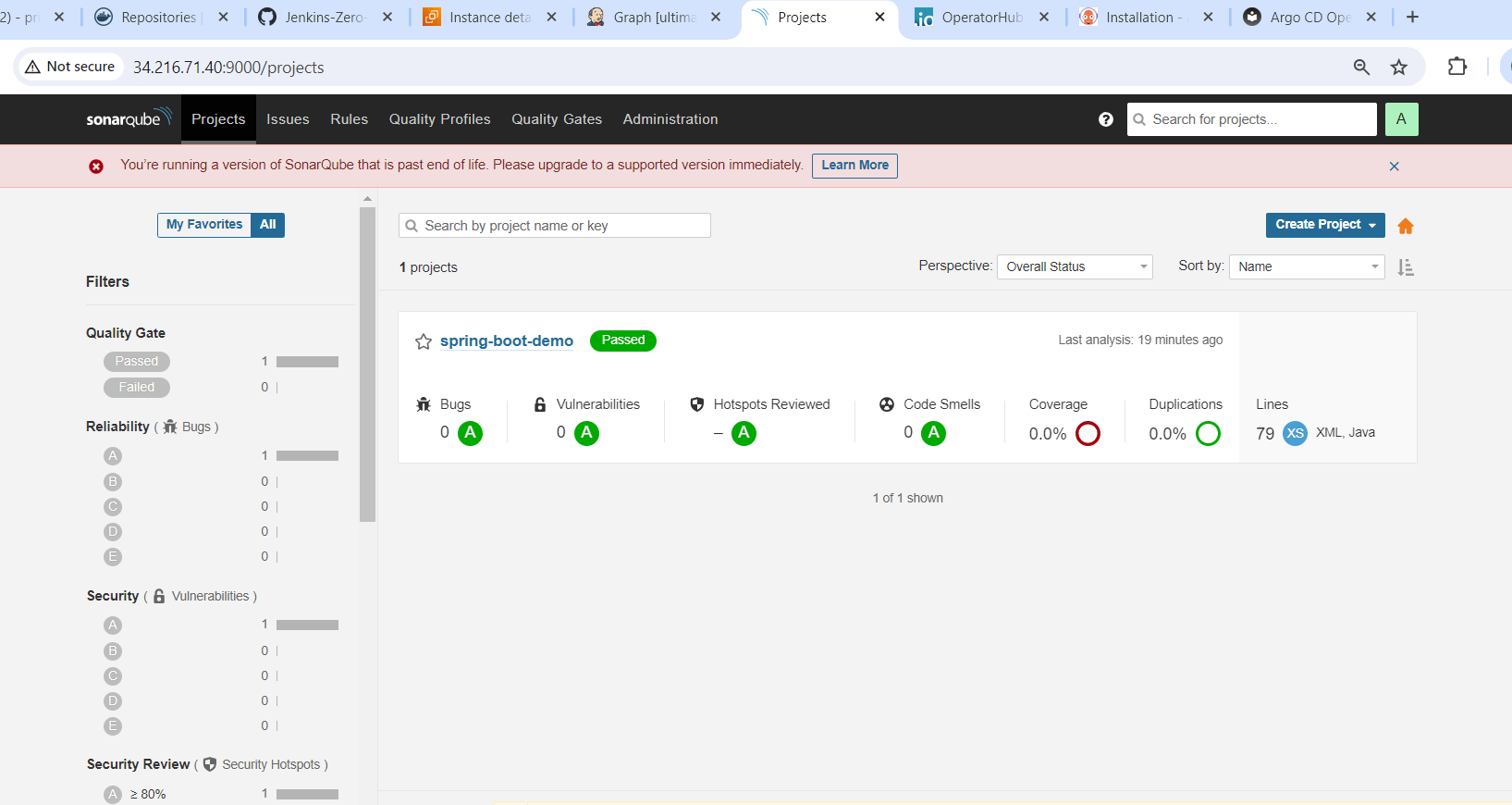
1. You can see in repository updated the deployment image to version 1. Tag is updated.

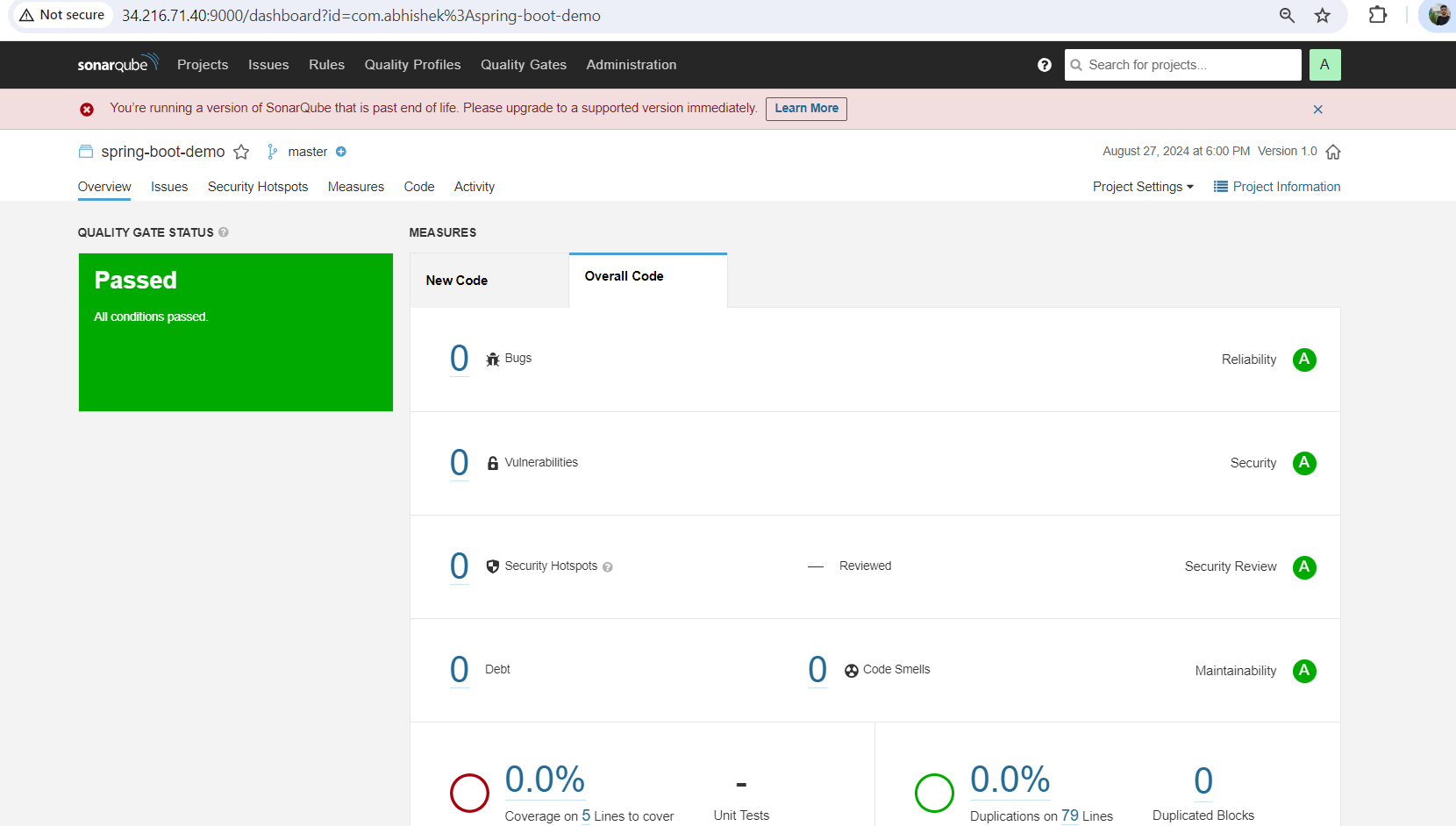


1. In the docker hub account image is created *priya194/ultimate-cicd:v1.*



1. In the SonarQube Project code is scanned.



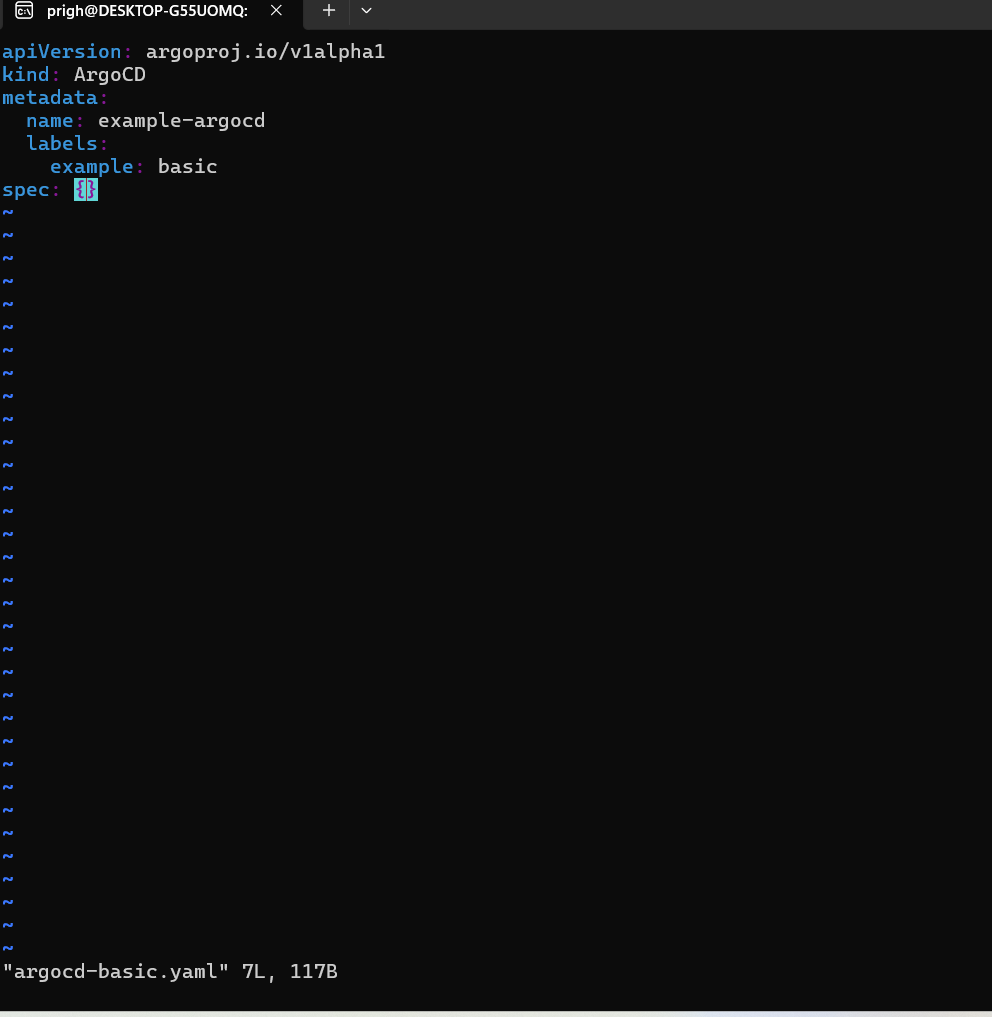


**CI part is done now. 😊**

1. **Now we will look for CD part**

* Go to the minikube
* Minikube start

*Vim basic-argocd.yaml*

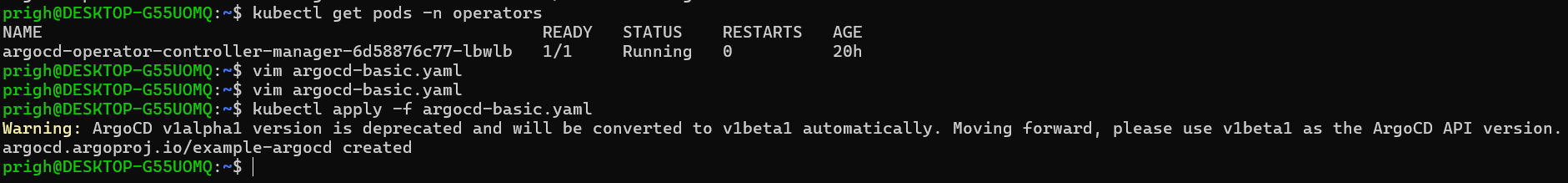


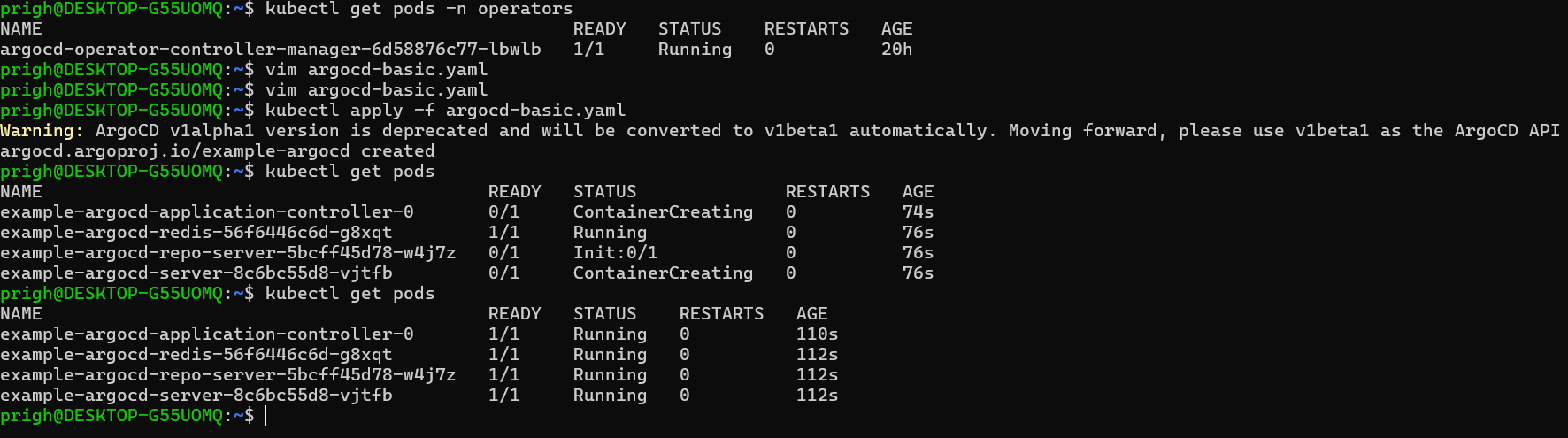
* Apply argocd-basic

*kubectl apply -f argocd-basic.yaml*

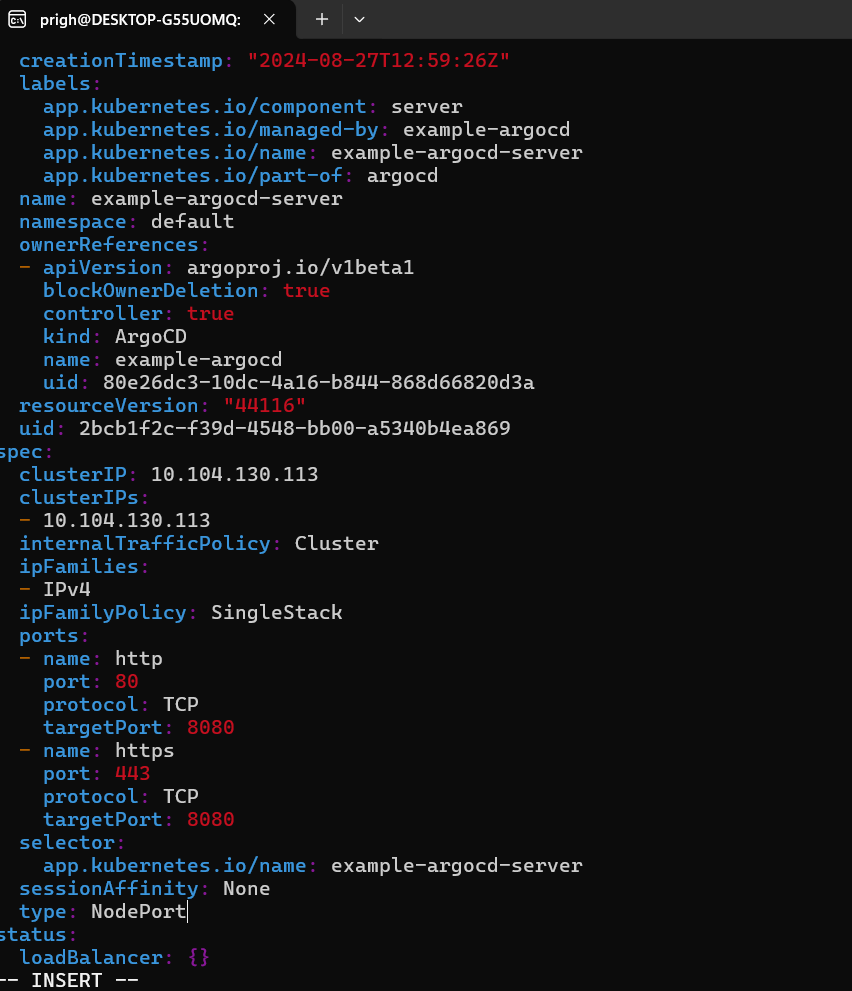
* Check the pods

*kubectl get pods*

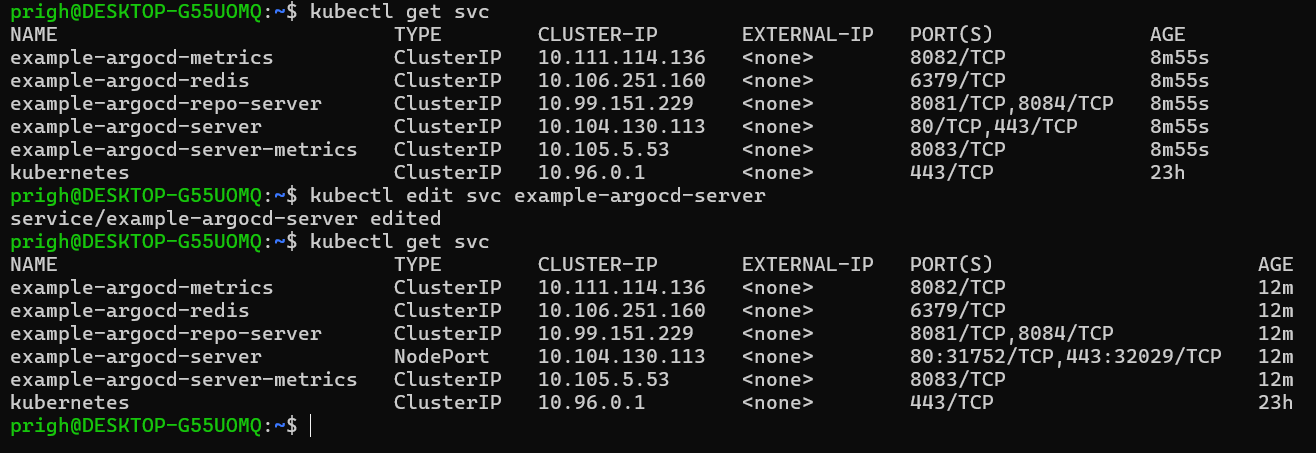


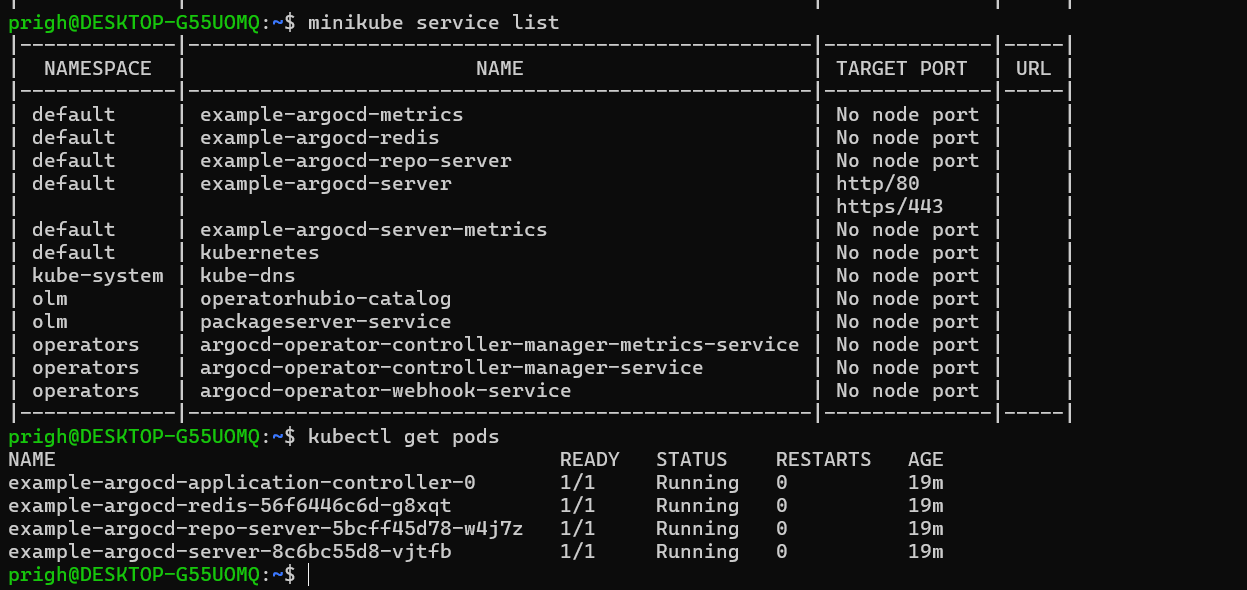


* In svc, type is clusterip change it to nodeport because we want to run on browser.



* Get the list of svc and the minikube service list.

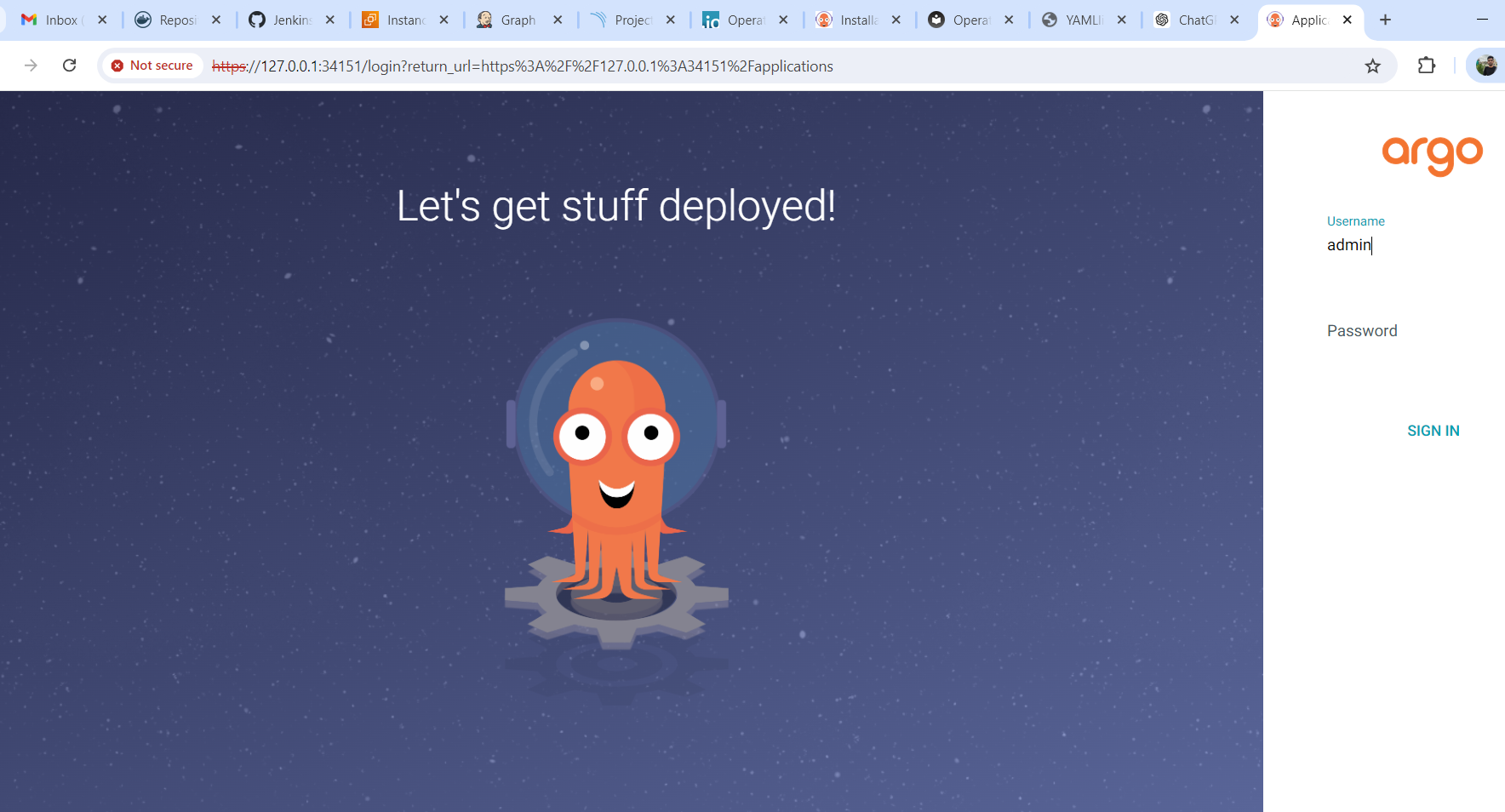




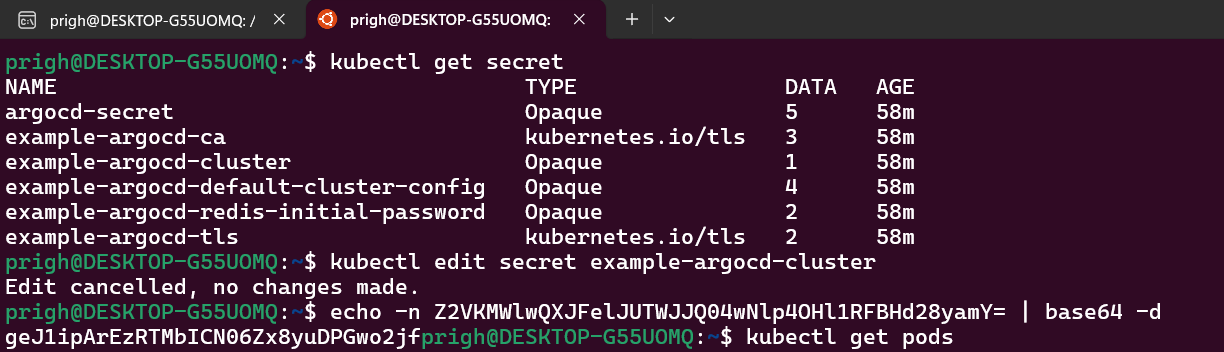
Minikube service example-argocd-server

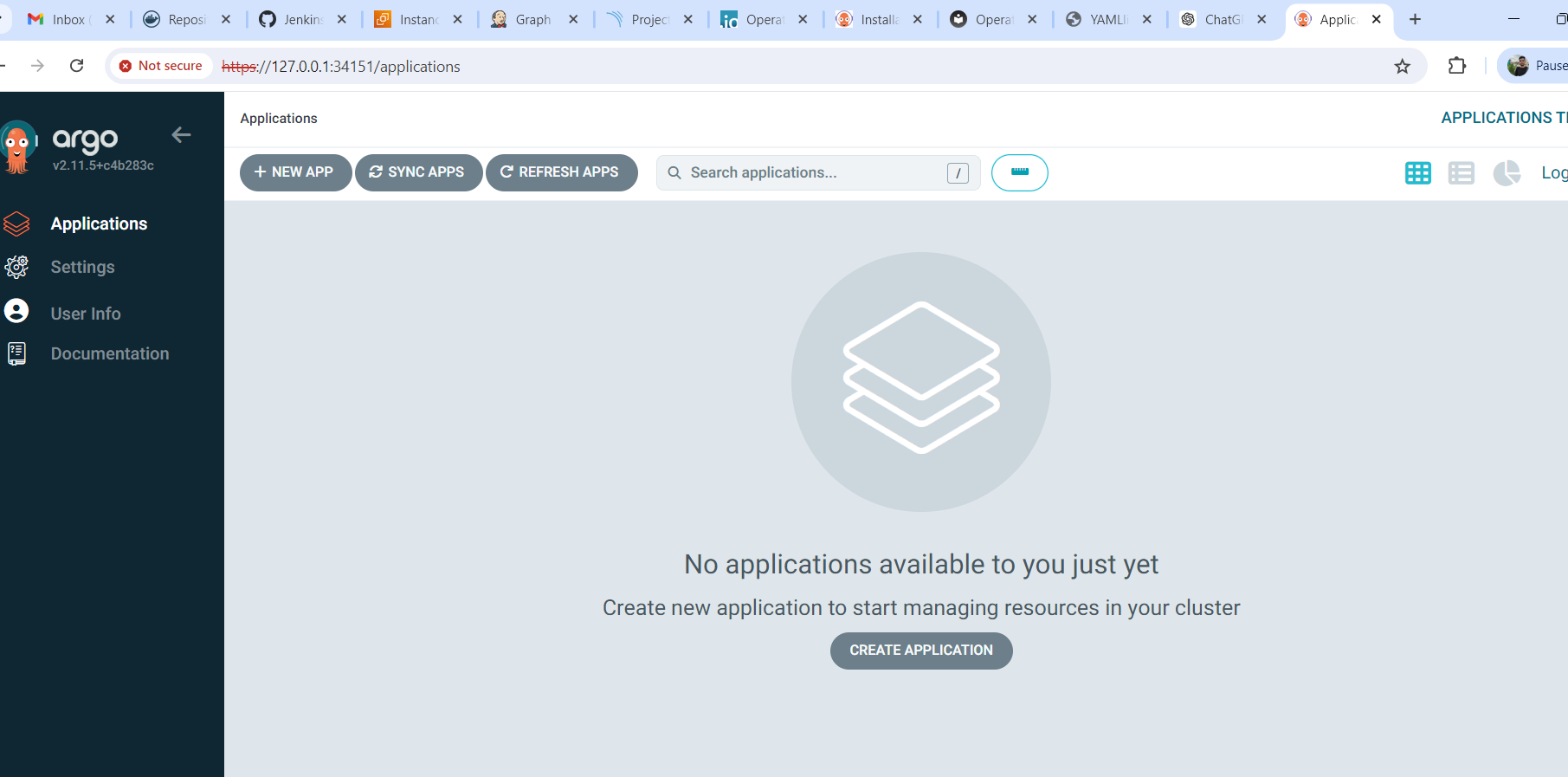


1. Copy the URL example-argocd-server and run on browser. You will get argocd started window.

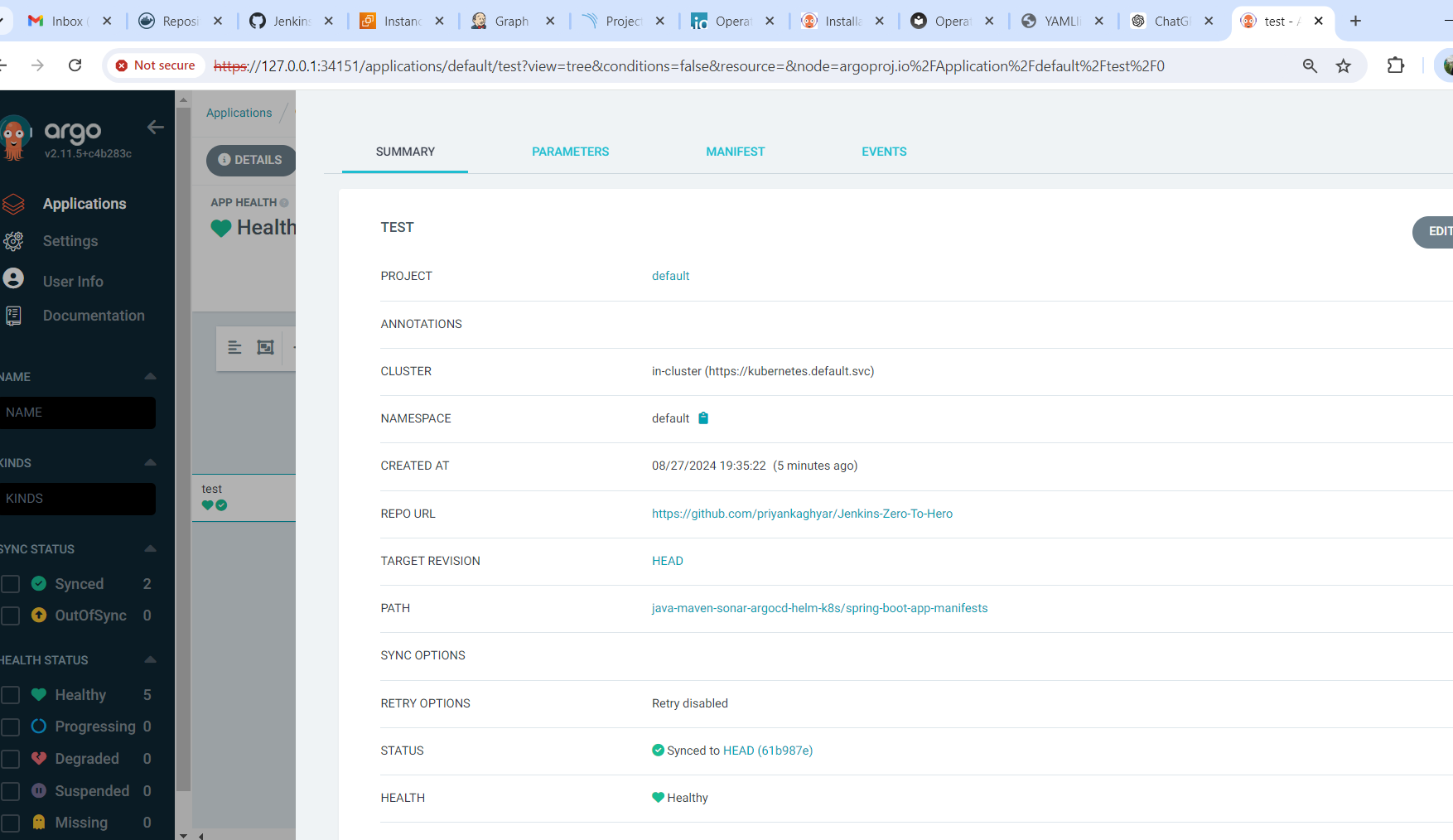


* check the secret on minikube and sign in argocd.

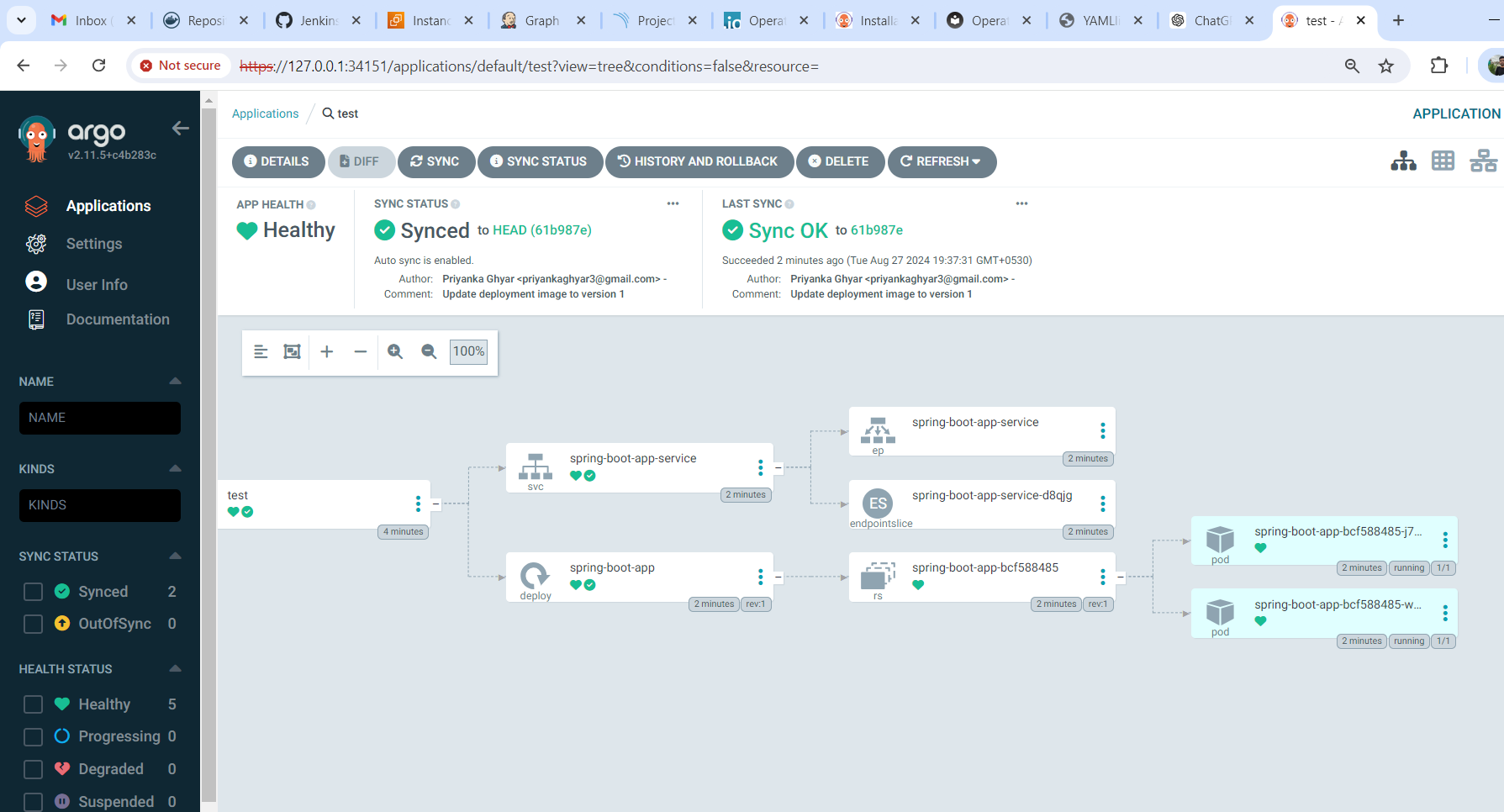




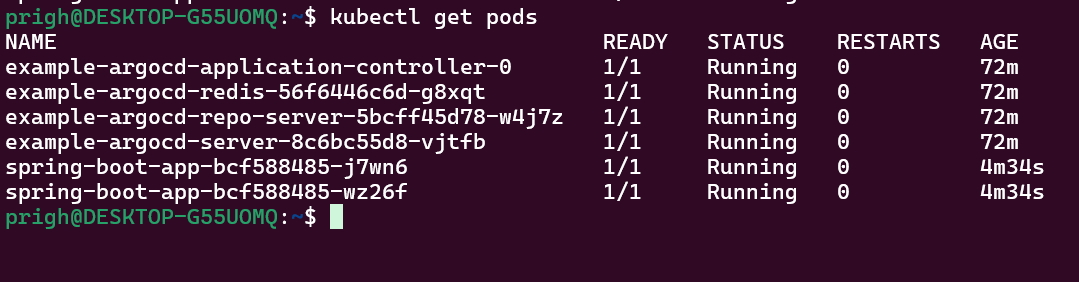
* Create the application and add the summary + create.

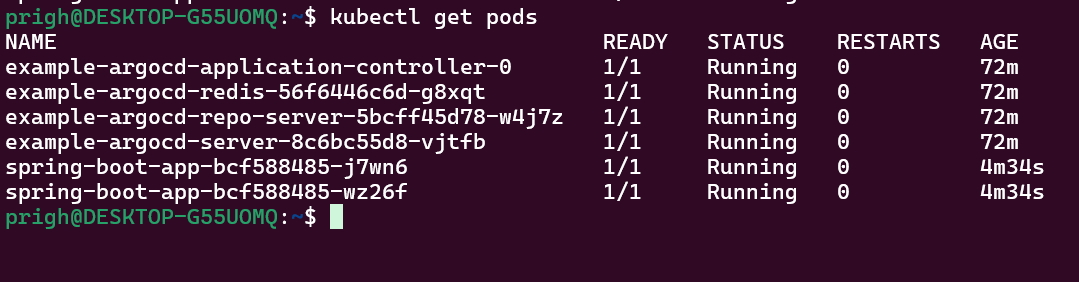


* Application is successfully deployed on argocd. App status is healthy and sync is ok.



* New 2 pods also running 😊.





**Ultimate CICD Jenkins Pipeline is successfully Done.!!**