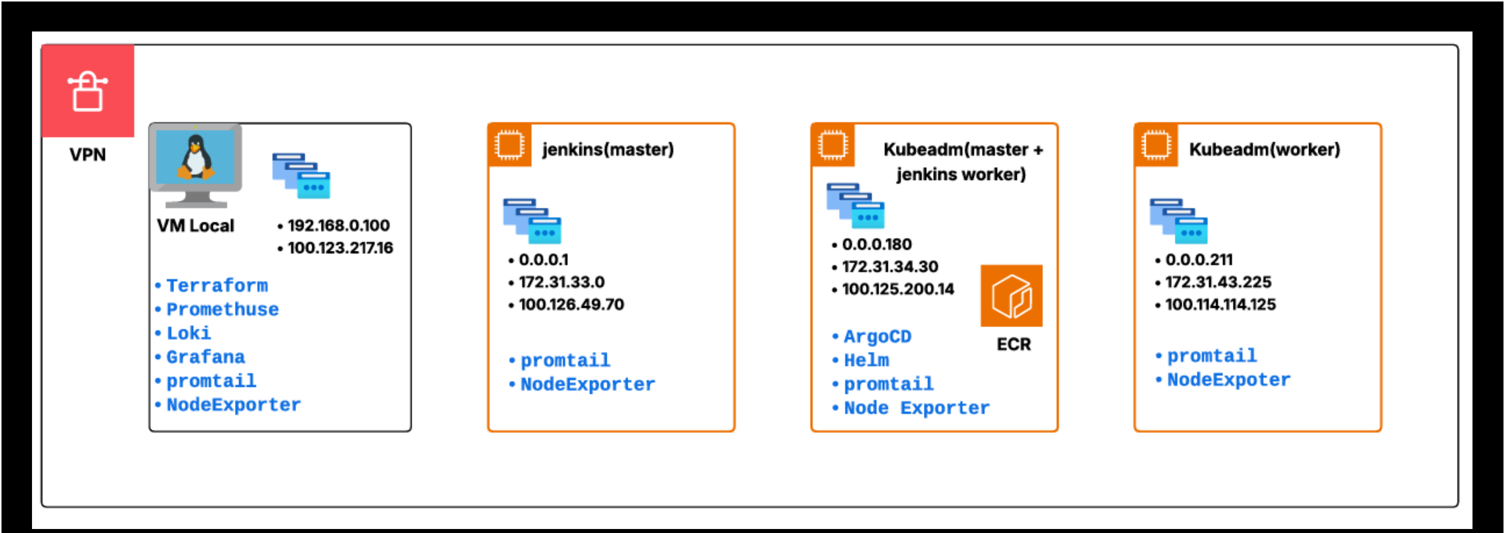
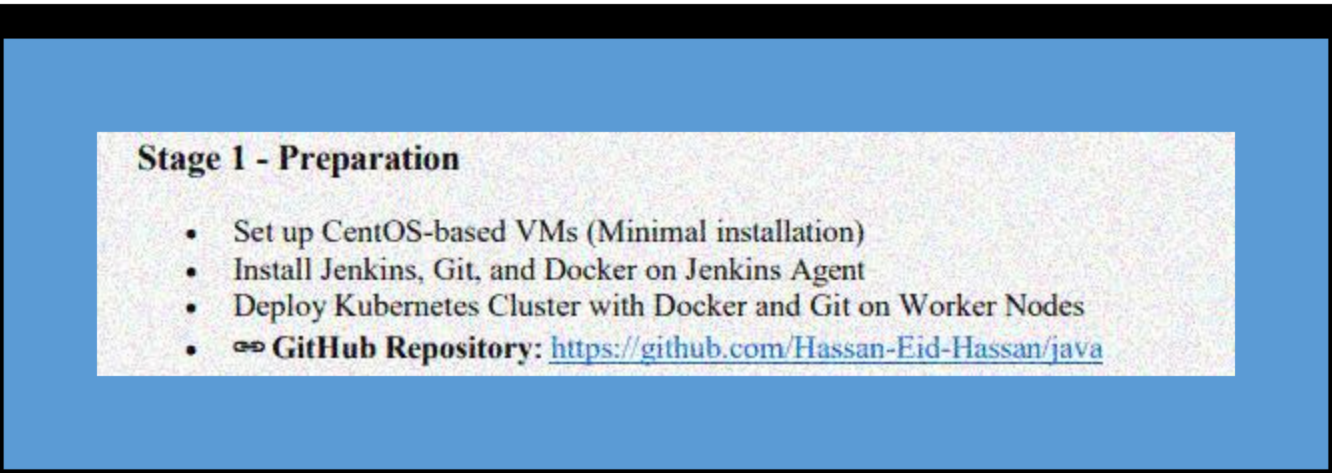
Java Application Deployment Pipeline

ITI

Diaa Qassem

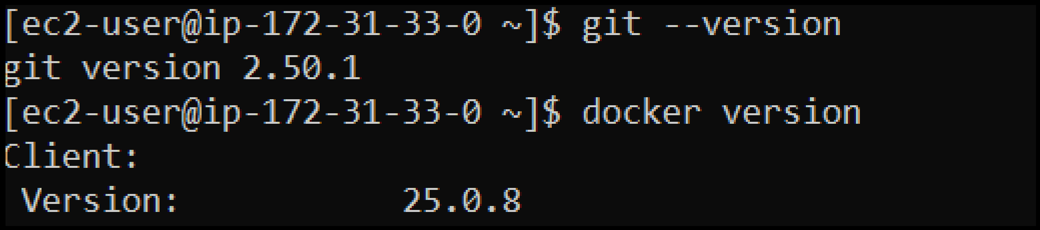




Stage 1 - Preparation

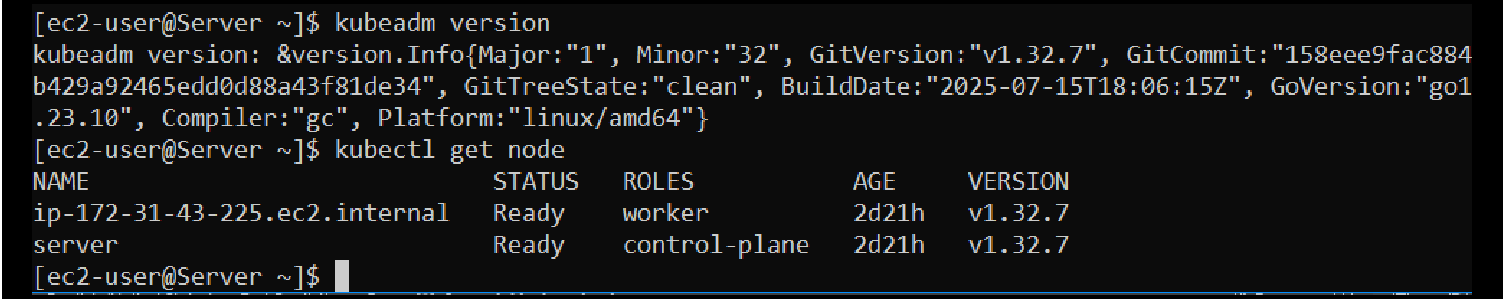
- Set up CentOS-based VMs (Minimal installation)
- Install Jenkins, Git, and Docker on Jenkins Agent
- Deploy Kubernetes Cluster with Docker and Git on Worker Nodes
- ⚭ **GitHub Repository:** https://github.com/Hassan-Eid-Hassan/java

Jenkins server



```
[ec2-user@ip-172-31-33-0 ~]$ git --version
git version 2.50.1
[ec2-user@ip-172-31-33-0 ~]$ docker version
Client:
 Version:           25.0.8
```

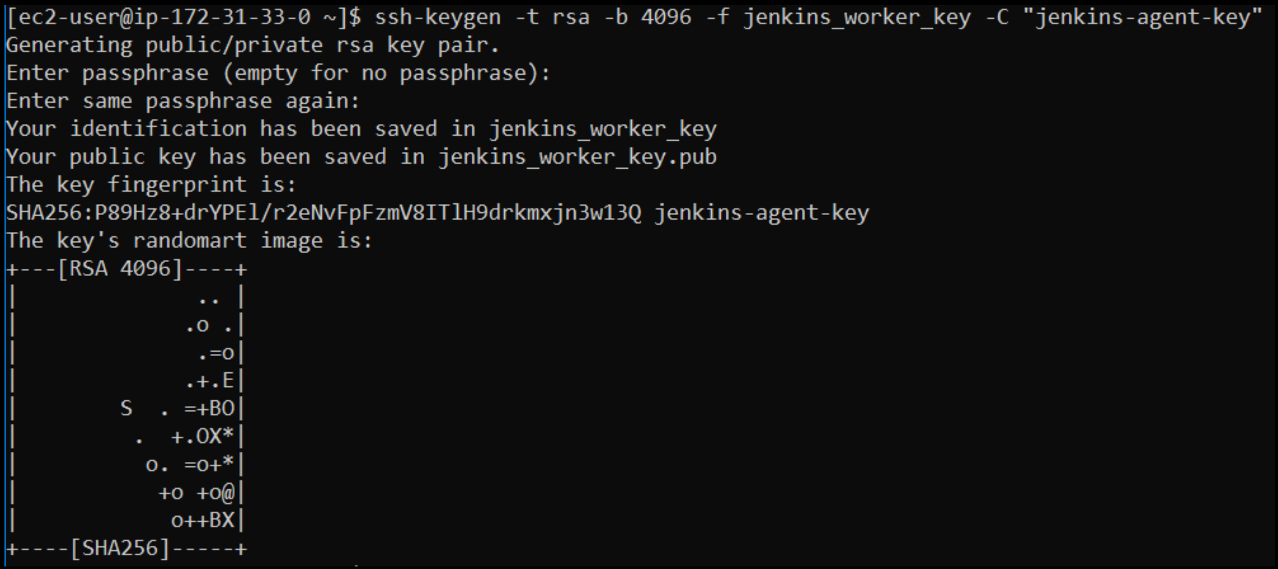
Kube cluster on worker



```
[ec2-user@Server ~]$ kubeadm version
kubeadm version: &version.Info{Major:"1", Minor:"32", GitVersion:"v1.32.7", GitCommit:"158eee9fac884
b429a92465edd0d88a43f81de34", GitTreeState:"clean", BuildDate:"2025-07-15T18:06:15Z", GoVersion:"go1
.23.10", Compiler:"gc", Platform:"linux/amd64"}
[ec2-user@Server ~]$ kubectl get node
NAME                           STATUS   ROLES           AGE     VERSION
ip-172-31-43-225.ec2.internal  Ready    worker          2d21h   v1.32.7
server                         Ready    control-plane   2d21h   v1.32.7
[ec2-user@Server ~]$
```

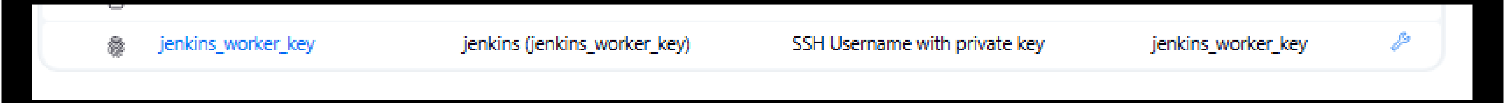
Add kubadm master as jenkins worker

1- Generate RSA key and send pub to kubeadm master(Jenkins-worker)



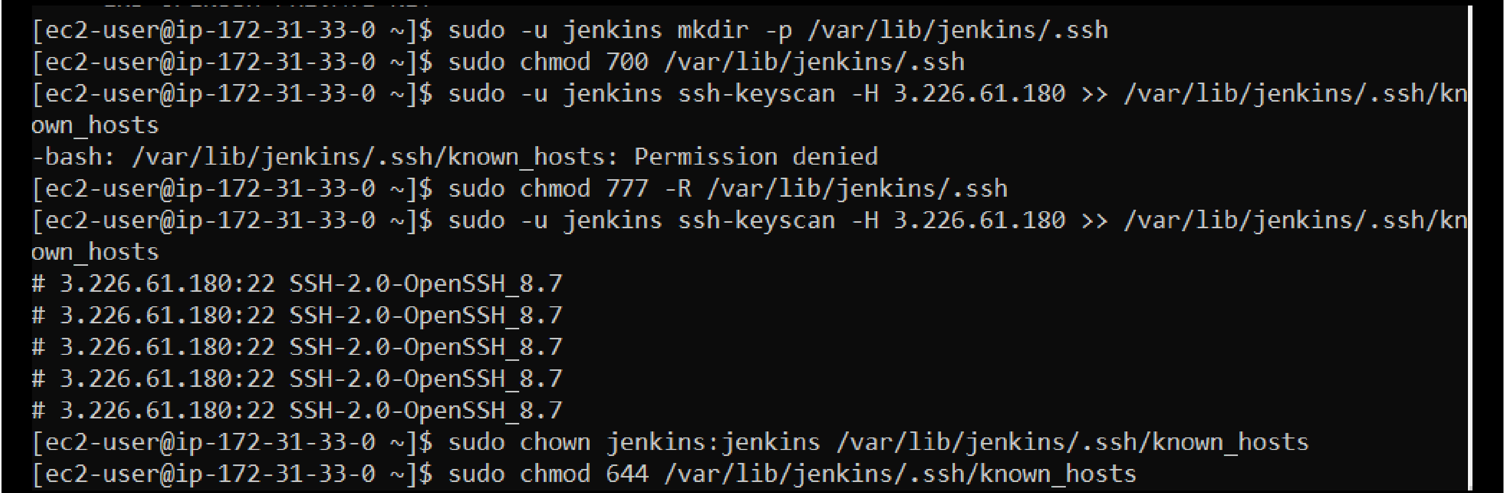
```
[ec2-user@ip-172-31-33-0 ~]$ ssh-keygen -t rsa -b 4096 -f jenkins_worker_key -C "jenkins-agent-key"
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in jenkins_worker_key
Your public key has been saved in jenkins_worker_key.pub
The key fingerprint is:
SHA256:P89Hz8+drYPEl/r2eNvFpFzmV8ITlH9drkmxjn3w13Q jenkins-agent-key
The key's randomart image is:
+---[RSA 4096]----+
|              .. |
|             .o .|
|              .=o|
|             .+.E|
|        S  . =+BO|
|         .  +.OX*|
|          o. =o+*|
|           +o +o@|
|            o++BX|
+----[SHA256]-----+
```

2- ADD priv key to credentials in Jenkins server as username and priv key



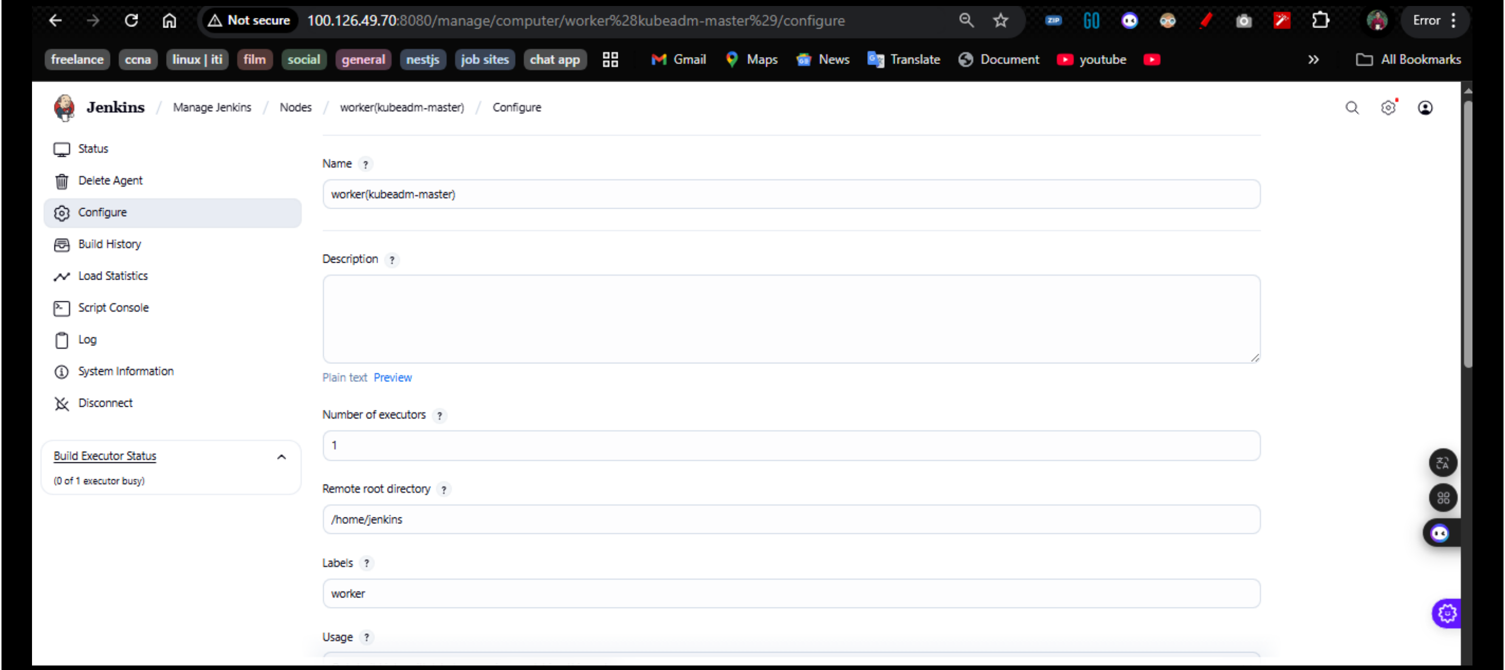
jenkins_worker_key   jenkins (jenkins_worker_key)   SSH Username with private key   jenkins_worker_key

3- Add worker ip in hosts



```
[ec2-user@ip-172-31-33-0 ~]$ sudo -u jenkins mkdir -p /var/lib/jenkins/.ssh
[ec2-user@ip-172-31-33-0 ~]$ sudo chmod 700 /var/lib/jenkins/.ssh
[ec2-user@ip-172-31-33-0 ~]$ sudo -u jenkins ssh-keyscan -H 3.226.61.180 >> /var/lib/jenkins/.ssh/kn
own_hosts
-bash: /var/lib/jenkins/.ssh/known_hosts: Permission denied
[ec2-user@ip-172-31-33-0 ~]$ sudo chmod 777 -R /var/lib/jenkins/.ssh
[ec2-user@ip-172-31-33-0 ~]$ sudo -u jenkins ssh-keyscan -H 3.226.61.180 >> /var/lib/jenkins/.ssh/kn
own_hosts
# 3.226.61.180:22 SSH-2.0-OpenSSH_8.7
# 3.226.61.180:22 SSH-2.0-OpenSSH_8.7
# 3.226.61.180:22 SSH-2.0-OpenSSH_8.7
# 3.226.61.180:22 SSH-2.0-OpenSSH_8.7
# 3.226.61.180:22 SSH-2.0-OpenSSH_8.7
[ec2-user@ip-172-31-33-0 ~]$ sudo chown jenkins:jenkins /var/lib/jenkins/.ssh/known_hosts
[ec2-user@ip-172-31-33-0 ~]$ sudo chmod 644 /var/lib/jenkins/.ssh/known_hosts
```

Add worker to Jenkins



100.126.49.70:8080/manage/computer/worker%28kubeadm-master%29/configure

Jenkins / Manage Jenkins / Nodes / worker(kubeadm-master) / Configure

Name ?

worker(kubeadm-master)

Description ?

Number of executors ?

1

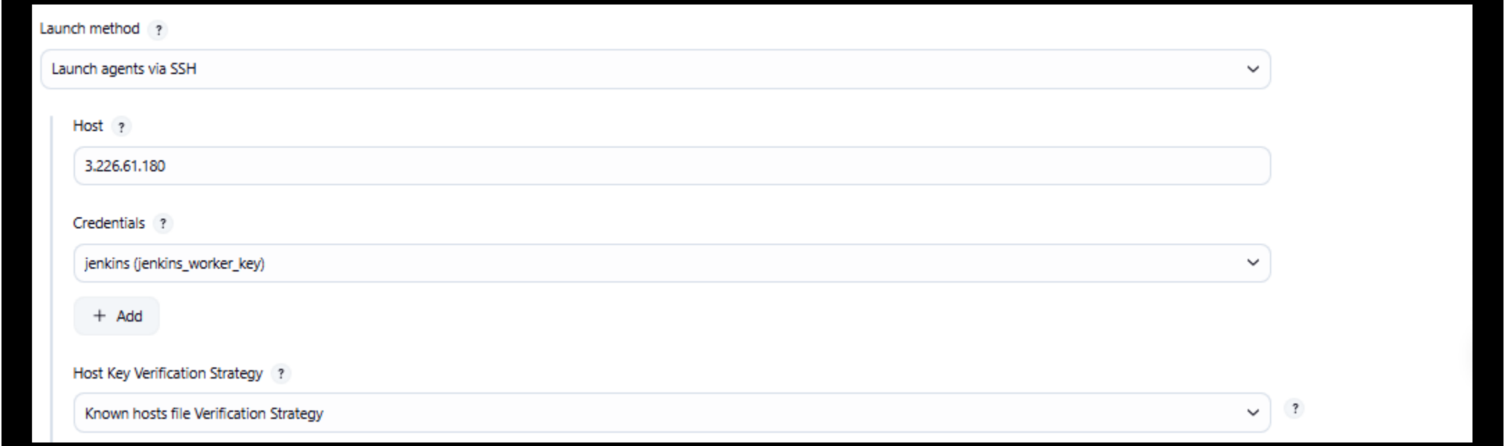
Remote root directory ?

/home/jenkins

Labels ?

worker

Usage ?



Launch method ?

Launch agents via SSH

Host ?

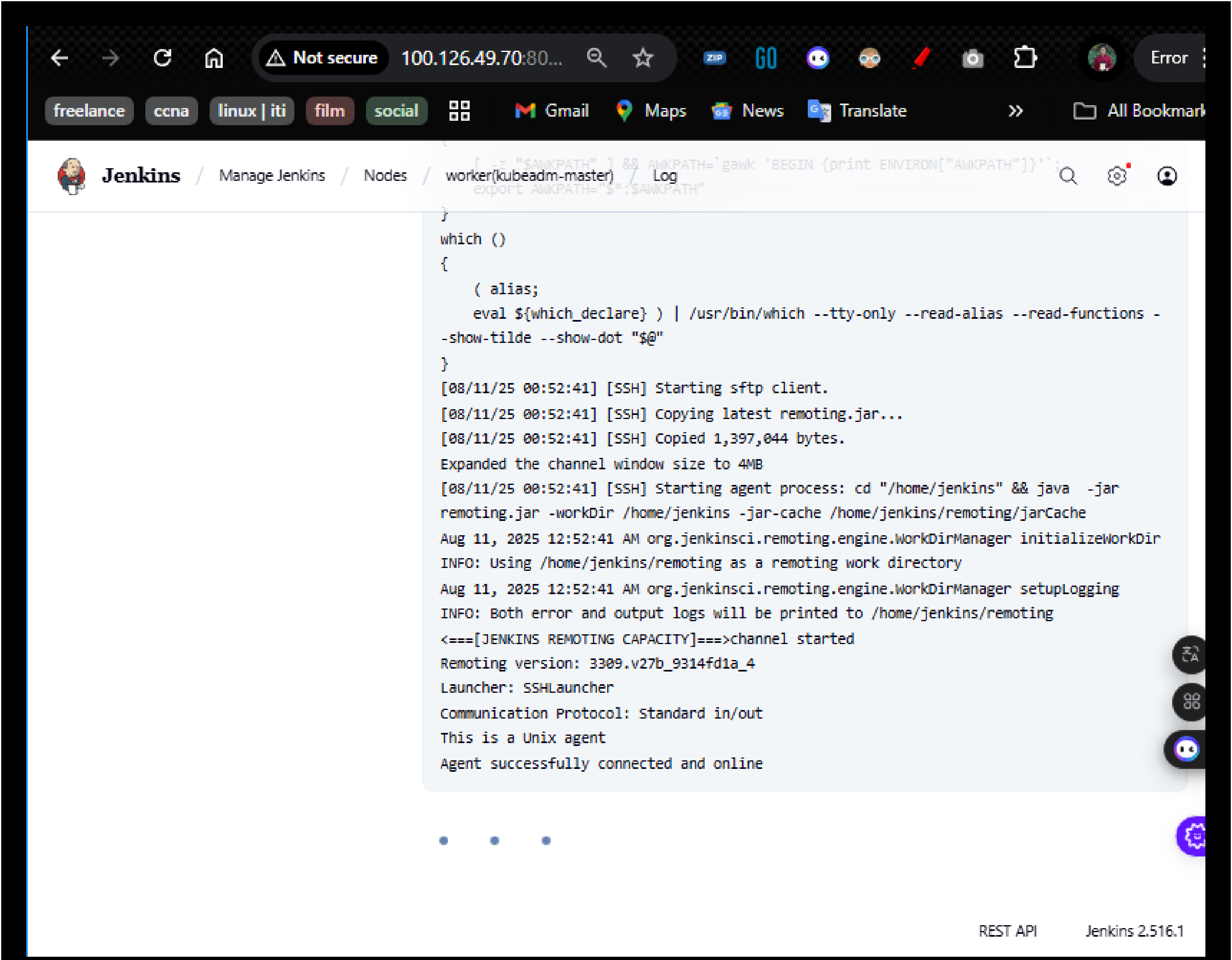
3.226.61.180

Credentials ?

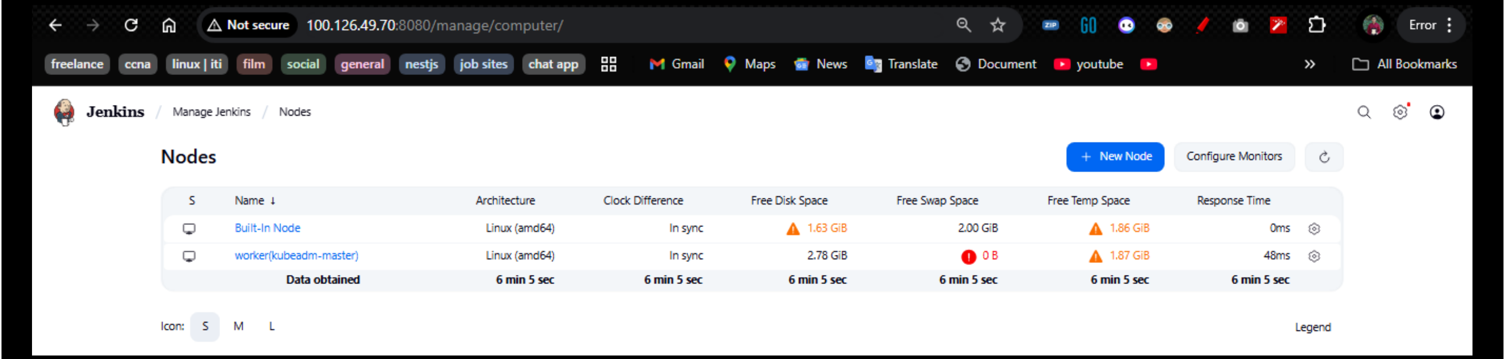
jenkins (jenkins_worker_key)

Host Key Verification Strategy ?

Known hosts file Verification Strategy



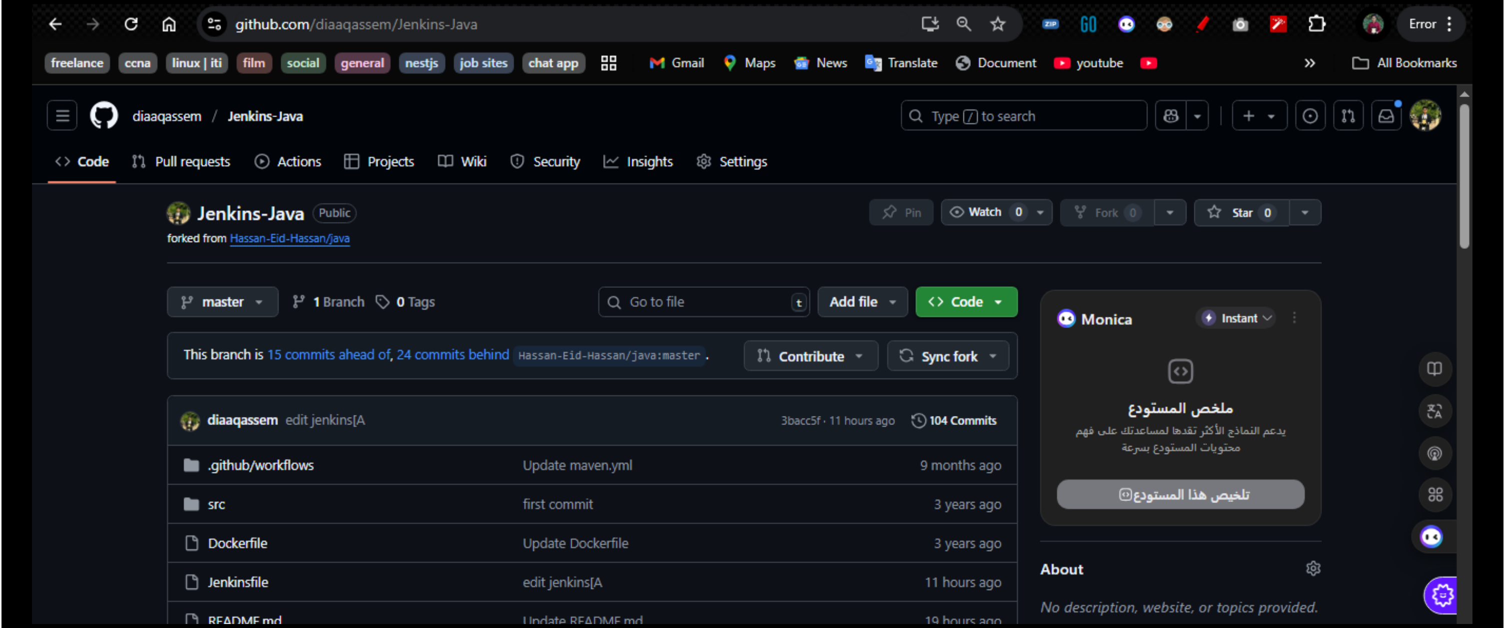
```
    [ -z "$AWKPATH" ] && AWKPATH=`gawk 'BEGIN {print ENVIRON["AWKPATH"]}'`;
    export AWKPATH="$*:$AWKPATH"
}
which ()
{
    ( alias;
    eval ${which_declare} ) | /usr/bin/which --tty-only --read-alias --read-functions -
-show-tilde --show-dot "$@"
}
[08/11/25 00:52:41] [SSH] Starting sftp client.
[08/11/25 00:52:41] [SSH] Copying latest remoting.jar...
[08/11/25 00:52:41] [SSH] Copied 1,397,044 bytes.
Expanded the channel window size to 4MB
[08/11/25 00:52:41] [SSH] Starting agent process: cd "/home/jenkins" && java  -jar
remoting.jar -workDir /home/jenkins -jar-cache /home/jenkins/remoting/jarCache
Aug 11, 2025 12:52:41 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/jenkins/remoting as a remoting work directory
Aug 11, 2025 12:52:41 AM org.jenkinsci.remoting.engine.WorkDirManager setupLogging
INFO: Both error and output logs will be printed to /home/jenkins/remoting
<===[JENKINS REMOTING CAPACITY]===>channel started
Remoting version: 3309.v27b_9314fd1a_4
Launcher: SSHLauncher
Communication Protocol: Standard in/out
This is a Unix agent
Agent successfully connected and online
```



Nodes

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
🖵	Built-In Node	Linux (amd64)	In sync	⚠ 1.63 GiB	2.00 GiB	⚠ 1.86 GiB	0ms
🖵	worker(kubeadm-master)	Linux (amd64)	In sync	2.78 GiB	● 0 B	⚠ 1.87 GiB	48ms
	Data obtained	6 min 5 sec	6 min 5 sec	6 min 5 sec	6 min 5 sec	6 min 5 sec	6 min 5 sec

Fork Java App to my repos



Jenkins-Java Public

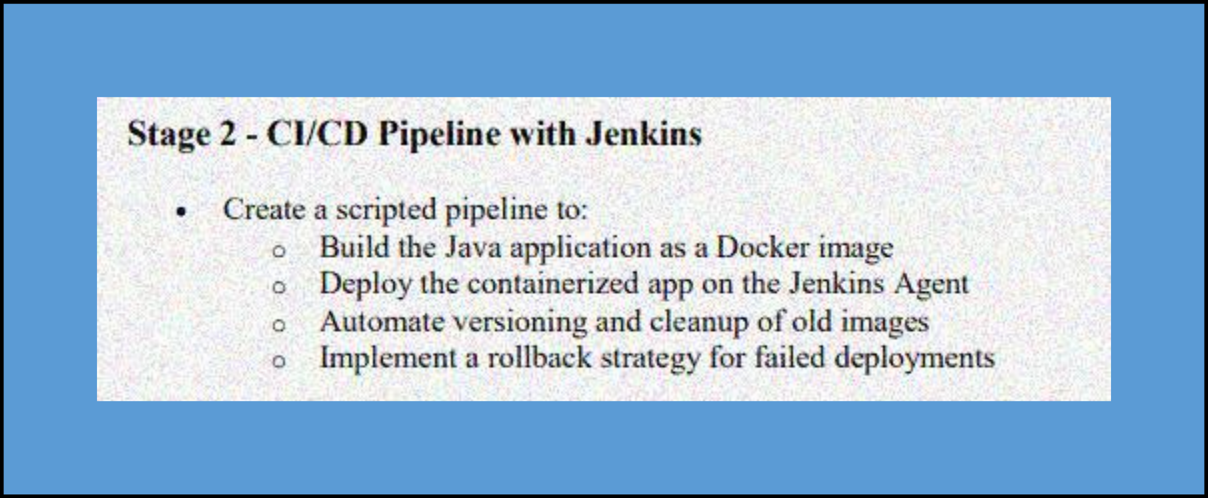
forked from Hassan-Eid-Hassan/java

This branch is 15 commits ahead of, 24 commits behind `Hassan-Eid-Hassan/java:master`.

diaaqassem edit jenkins[A 3bacc5f · 11 hours ago 104 Commits

.github/workflows	Update maven.yml	9 months ago
src	first commit	3 years ago
Dockerfile	Update Dockerfile	3 years ago
Jenkinsfile	edit jenkins[A	11 hours ago
README.md	Update README.md	19 hours ago

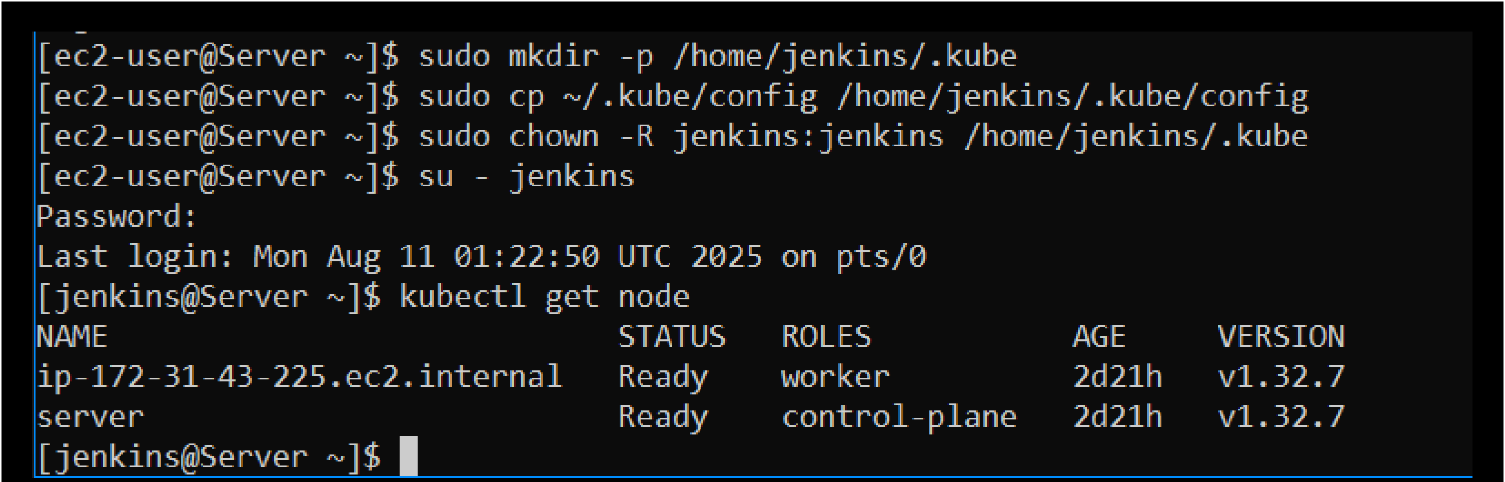
No description, website, or topics provided.



Stage 2 - CI/CD Pipeline with Jenkins

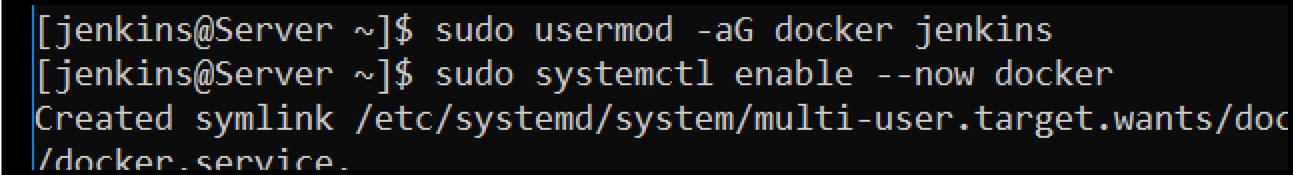
- Create a scripted pipeline to:
  - Build the Java application as a Docker image
  - Deploy the containerized app on the Jenkins Agent
  - Automate versioning and cleanup of old images
  - Implement a rollback strategy for failed deployments

- Add Permissions To Jenkins user on worker to access kube

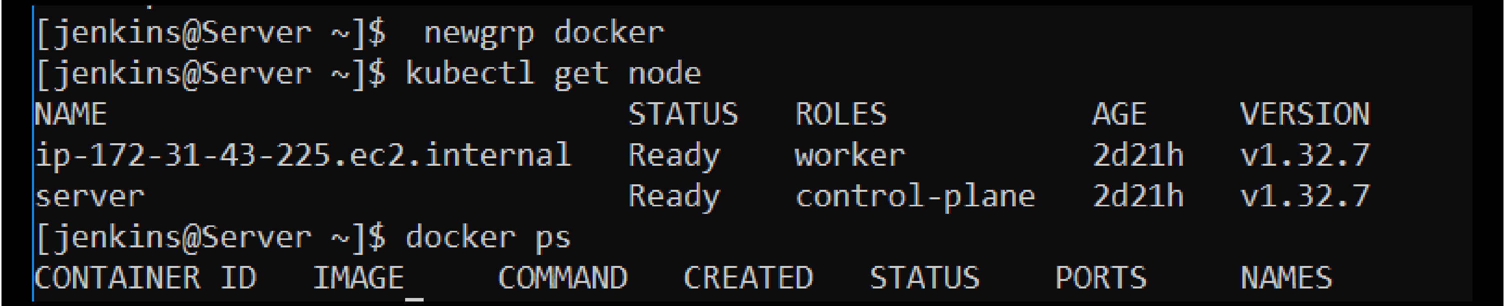


```
[ec2-user@Server ~]$ sudo mkdir -p /home/jenkins/.kube
[ec2-user@Server ~]$ sudo cp ~/.kube/config /home/jenkins/.kube/config
[ec2-user@Server ~]$ sudo chown -R jenkins:jenkins /home/jenkins/.kube
[ec2-user@Server ~]$ su - jenkins
Password:
Last login: Mon Aug 11 01:22:50 UTC 2025 on pts/0
[jenkins@Server ~]$ kubectl get node
NAME                            STATUS   ROLES           AGE     VERSION
ip-172-31-43-225.ec2.internal   Ready    worker          2d21h   v1.32.7
server                          Ready    control-plane   2d21h   v1.32.7
[jenkins@Server ~]$
```

Add Jenkins user on worker to docker group

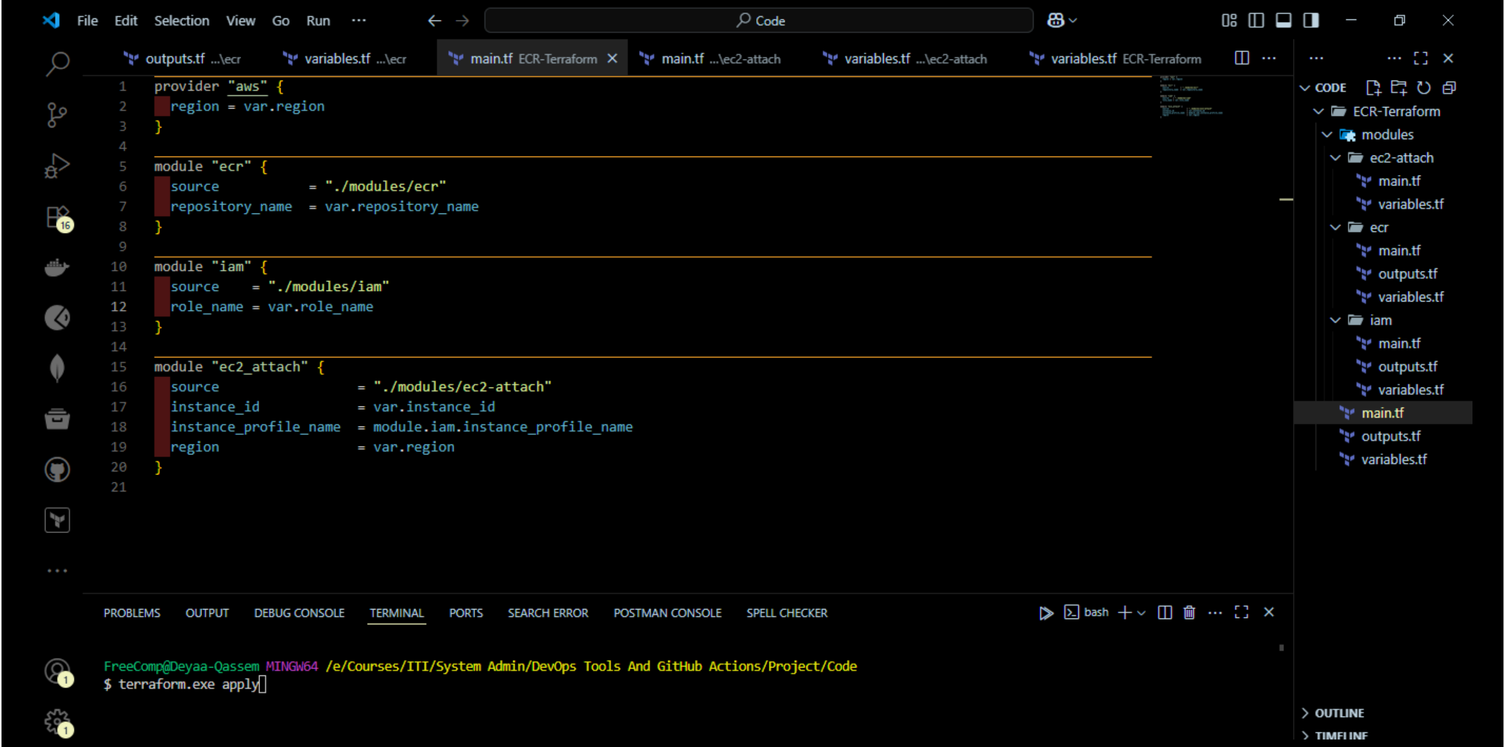


```
[jenkins@Server ~]$ sudo usermod -aG docker jenkins
[jenkins@Server ~]$ sudo systemctl enable --now docker
Created symlink /etc/systemd/system/multi-user.target.wants/doc
/docker.service.
```



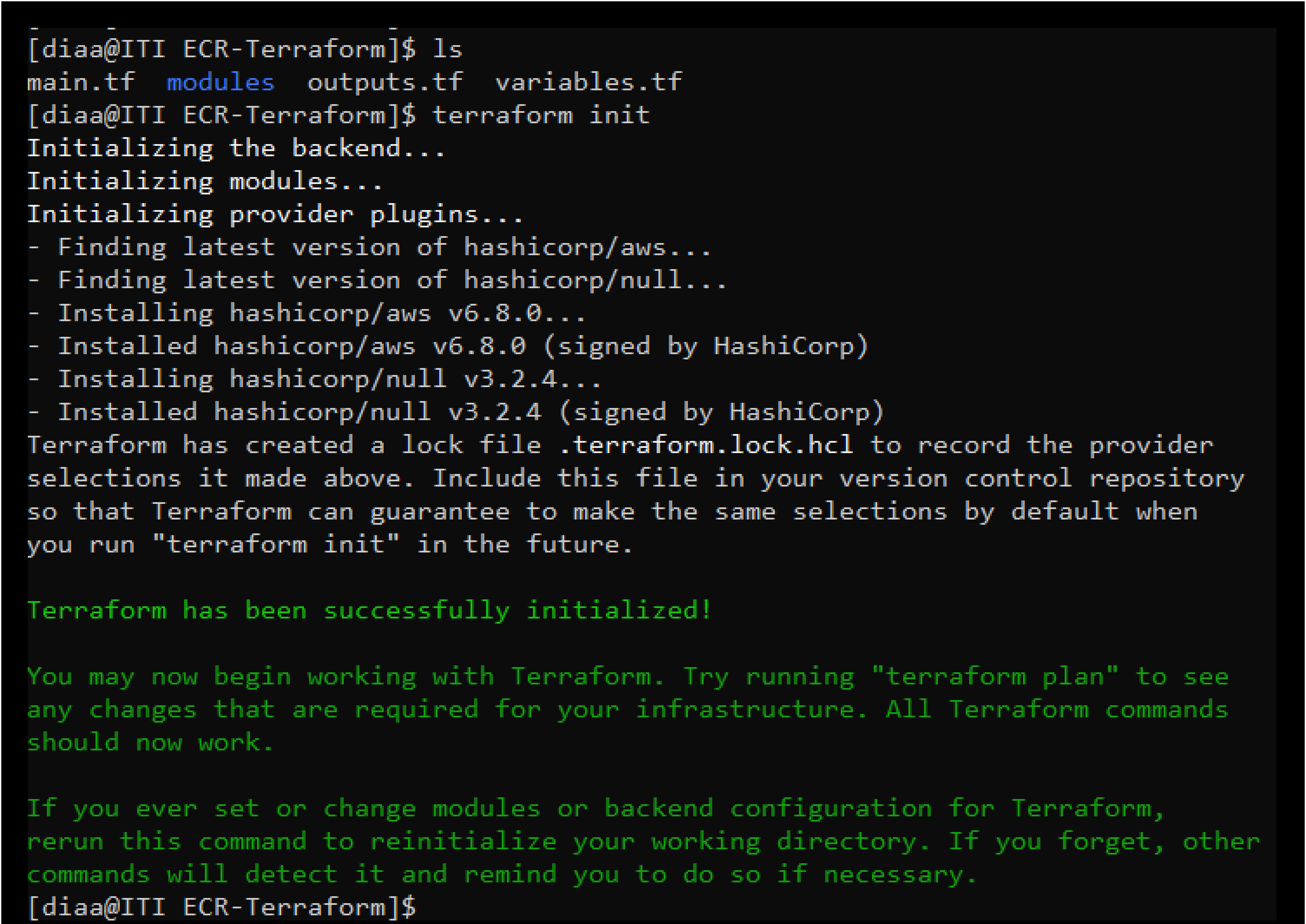
```
[jenkins@Server ~]$  newgrp docker
[jenkins@Server ~]$ kubectl get node
NAME                            STATUS   ROLES           AGE     VERSION
ip-172-31-43-225.ec2.internal   Ready    worker          2d21h   v1.32.7
server                          Ready    control-plane   2d21h   v1.32.7
[jenkins@Server ~]$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

Create AWS ECR, IAM Role To Access ECR and attach IAM with EC2(Jenkins worker) with Terraform

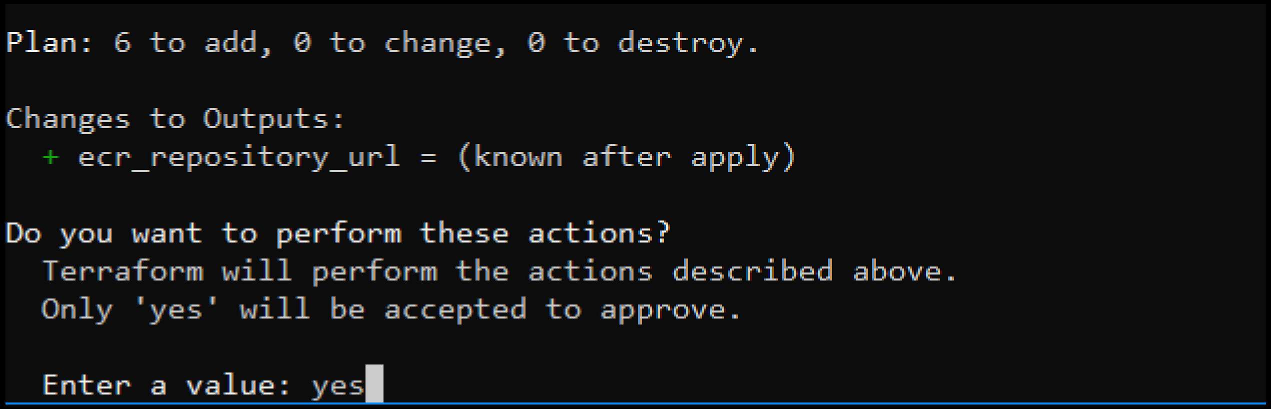


```
provider "aws" {
  region = var.region
}

module "ecr" {
  source          = "./modules/ecr"
  repository_name = var.repository_name
}

module "iam" {
  source    = "./modules/iam"
  role_name = var.role_name
}

module "ec2_attach" {
  source                = "./modules/ec2-attach"
  instance_id           = var.instance_id
  instance_profile_name = module.iam.instance_profile_name
  region                = var.region
}
```

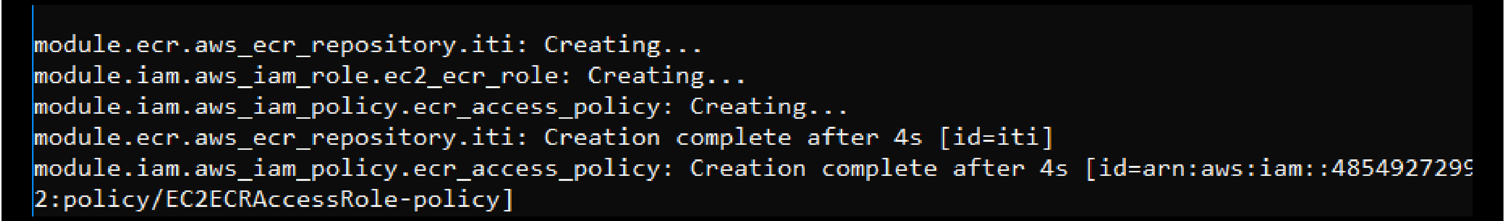
```
FreeComp@Deyaa-Qassem MINGW64 /e/Courses/ITI/System Admin/DevOps Tools And GitHub Actions/Project/Code
$ terraform.exe apply
```



```
[diaa@ITI ECR-Terraform]$ ls
main.tf  modules  outputs.tf  variables.tf
[diaa@ITI ECR-Terraform]$ terraform init
Initializing the backend...
Initializing modules...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Finding latest version of hashicorp/null...
- Installing hashicorp/aws v6.8.0...
- Installed hashicorp/aws v6.8.0 (signed by HashiCorp)
- Installing hashicorp/null v3.2.4...
- Installed hashicorp/null v3.2.4 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
[diaa@ITI ECR-Terraform]$
```

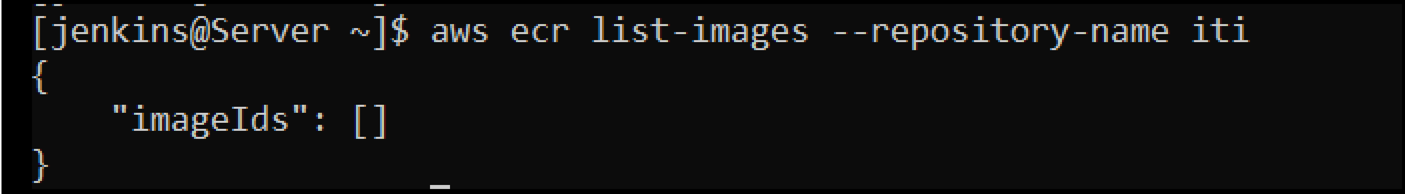


```
Plan: 6 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + ecr_repository_url = (known after apply)

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes
```



```
module.ecr.aws_ecr_repository.iti: Creating...
module.iam.aws_iam_role.ec2_ecr_role: Creating...
module.iam.aws_iam_policy.ecr_access_policy: Creating...
module.ecr.aws_ecr_repository.iti: Creation complete after 4s [id=iti]
module.iam.aws_iam_policy.ecr_access_policy: Creation complete after 4s [id=arn:aws:iam::4854927299
2:policy/EC2ECRAccessRole-policy]
```

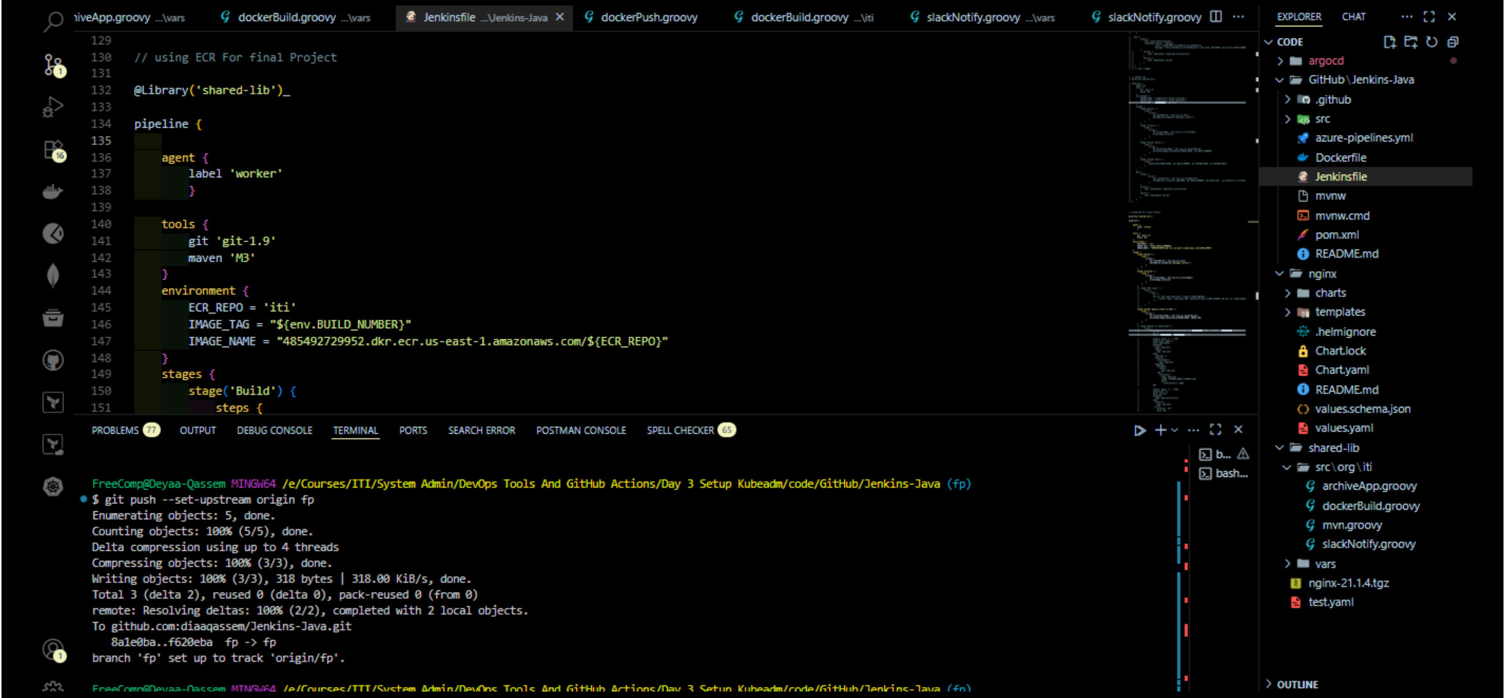
Jenkins user on worker has access to ECR



```
[jenkins@Server ~]$ aws ecr list-images --repository-name iti
{
    "imageIds": []
}
```

Create new branch fp on GitHub

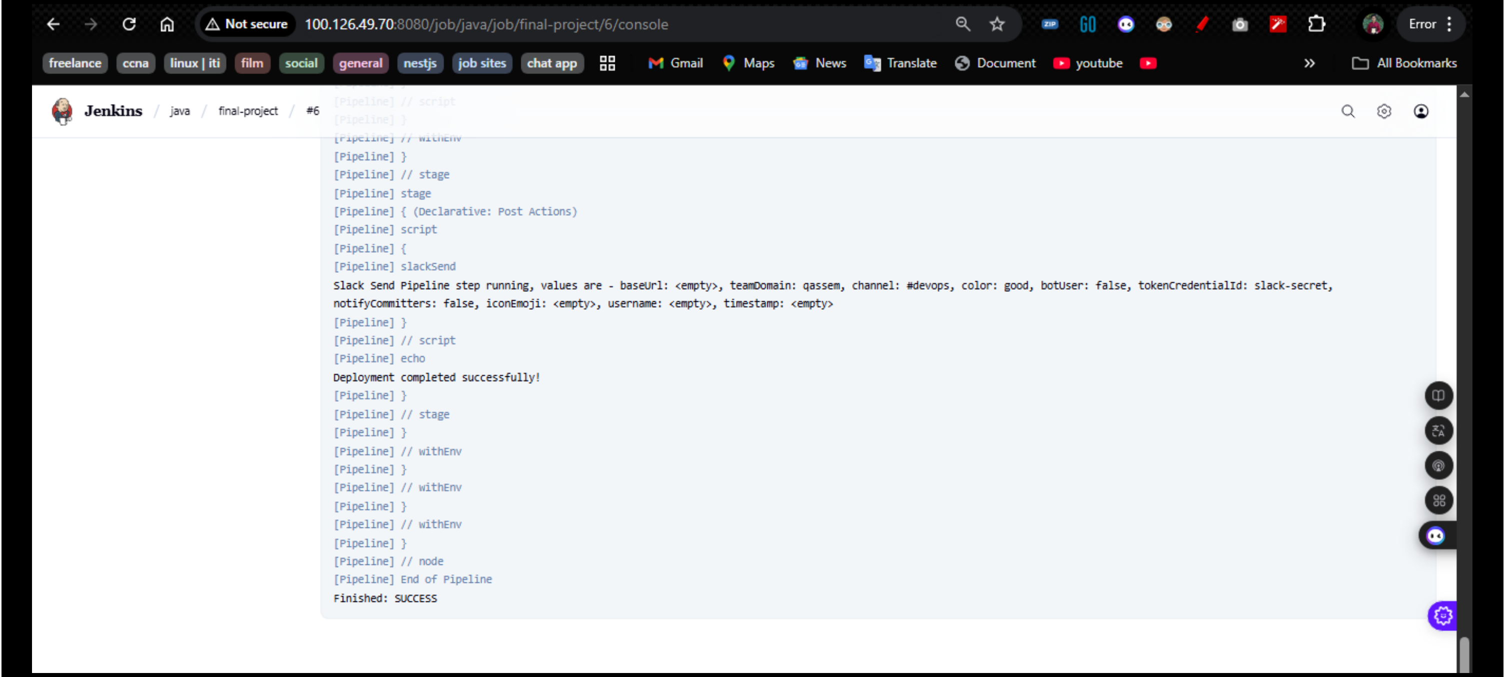
https://github.com/diaaqassem/Jenkins-Java/tree/fp



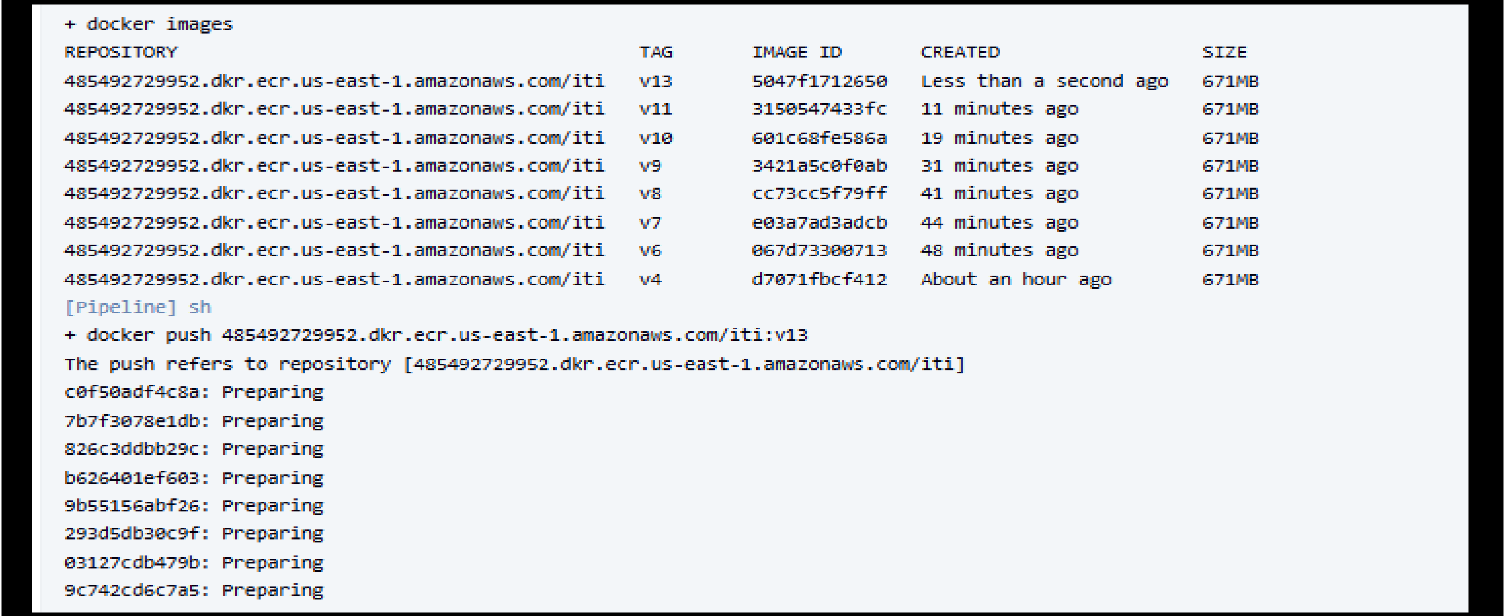
```
// using ECR For final Project

@Library('shared-lib')_

pipeline {

    agent {
        label 'worker'
        }

    tools {
        git 'git-1.9'
        maven 'M3'
    }
    environment {
        ECR_REPO = 'iti'
        IMAGE_TAG = "${env.BUILD_NUMBER}"
        IMAGE_NAME = "485492729952.dkr.ecr.us-east-1.amazonaws.com/${ECR_REPO}"
    }
    stages {
        stage('Build') {
            steps {
```

```
FreeComp@Deyaa-Qassem MINGW64 /e/Courses/ITI/System Admin/DevOps Tools And GitHub Actions/Day 3 Setup Kubeadm/code/GitHub/Jenkins-Java (fp)
$ git push --set-upstream origin fp
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 318 bytes | 318.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:diaaqassem/Jenkins-Java.git
   8a1e0ba..f620eba  fp -> fp
branch 'fp' set up to track 'origin/fp'.

FreeComp@Deyaa-Qassem MINGW64 /e/Courses/ITI/System Admin/DevOps Tools And GitHub Actions/Day 3 Setup Kubeadm/code/GitHub/Jenkins-Java (fp)
```

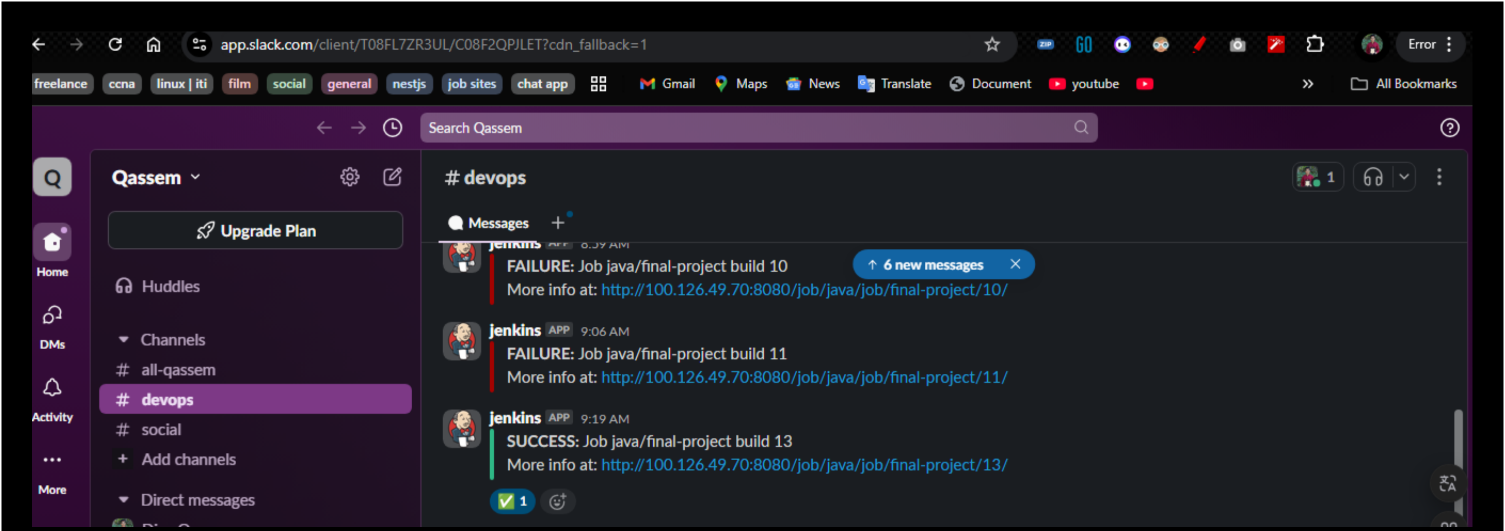
Test App



```
[Pipeline] // script
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] script
[Pipeline] {
[Pipeline] slackSend
Slack Send Pipeline step running, values are - baseUrl: <empty>, teamDomain: qassem, channel: #devops, color: good, botUser: false, tokenCredentialId: slack-secret,
notifyCommitters: false, iconEmoji: <empty>, username: <empty>, timestamp: <empty>
[Pipeline] }
[Pipeline] // script
[Pipeline] echo
Deployment completed successfully!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```



```
+ docker images
REPOSITORY                                          TAG       IMAGE ID       CREATED                  SIZE
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v13       5047f1712650   Less than a second ago   671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v11       3150547433fc   11 minutes ago           671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v10       601c68fe586a   19 minutes ago           671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v9        3421a5c0f0ab   31 minutes ago           671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v8        cc73cc5f79ff   41 minutes ago           671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v7        e03a7ad3adcb   44 minutes ago           671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v6        067d73300713   48 minutes ago           671MB
485492729952.dkr.ecr.us-east-1.amazonaws.com/iti    v4        d7071fbcf412   About an hour ago        671MB
[Pipeline] sh
+ docker push 485492729952.dkr.ecr.us-east-1.amazonaws.com/iti:v13
The push refers to repository [485492729952.dkr.ecr.us-east-1.amazonaws.com/iti]
c0f50adf4c8a: Preparing
7b7f3078e1db: Preparing
826c3ddbb29c: Preparing
b626401ef603: Preparing
9b55156abf26: Preparing
293d5db30c9f: Preparing
03127cdb479b: Preparing
9c742cd6c7a5: Preparing
```

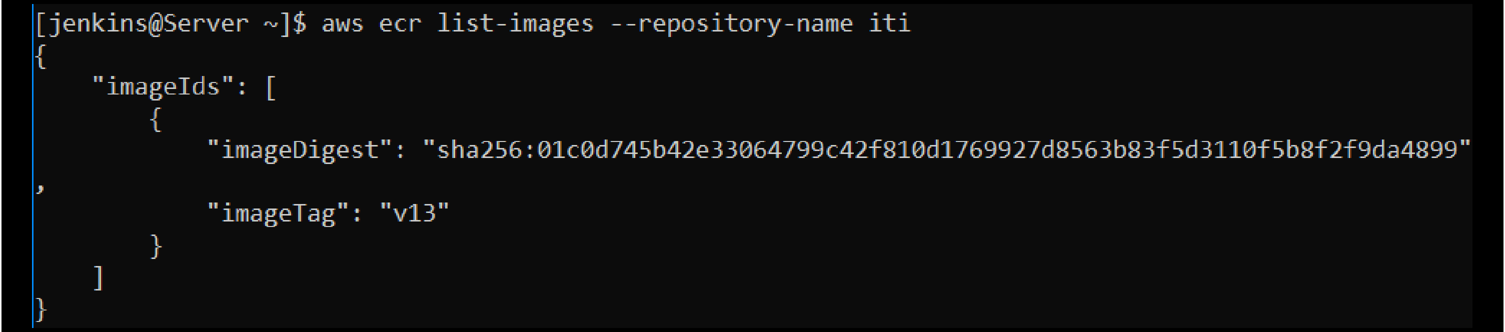


jenkins APP 8:59 AM
FAILURE: Job java/final-project build 10
More info at: http://100.126.49.70:8080/job/java/job/final-project/10/

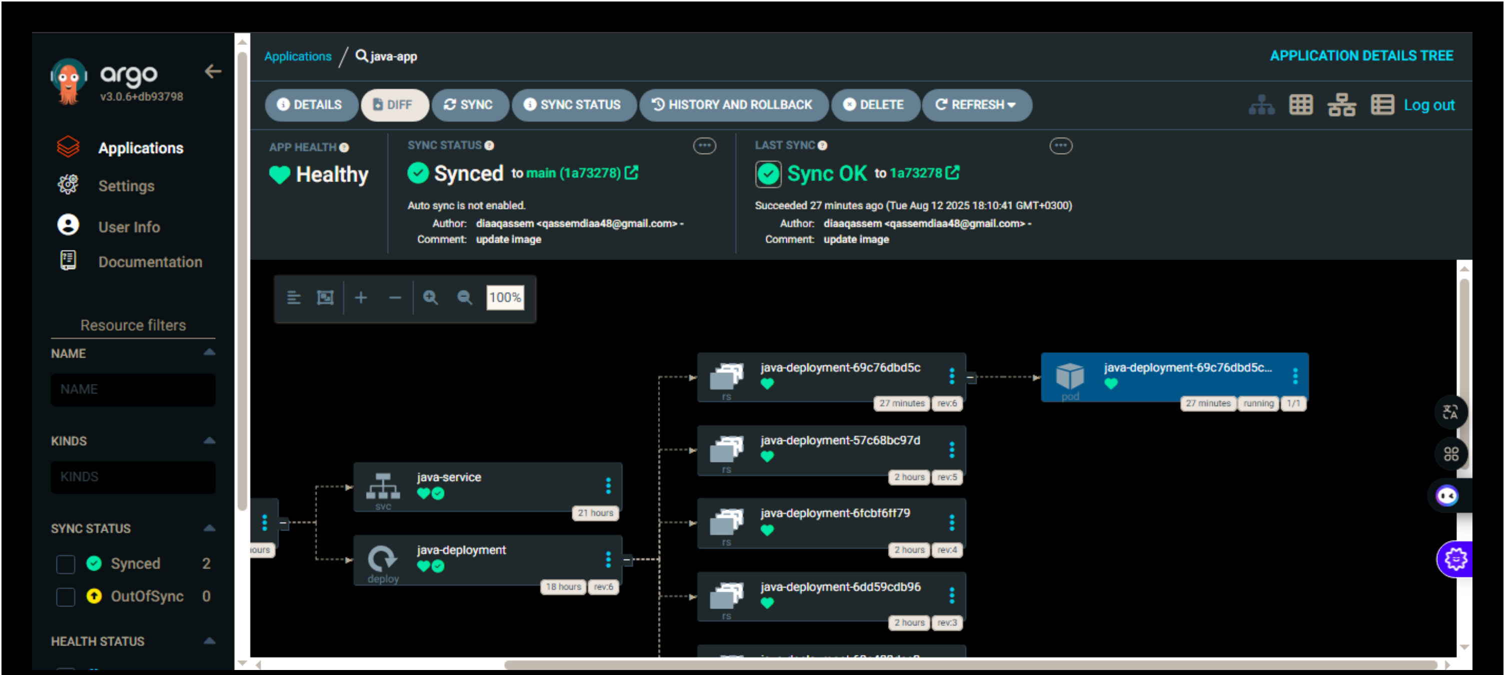
jenkins APP 9:06 AM
FAILURE: Job java/final-project build 11
More info at: http://100.126.49.70:8080/job/java/job/final-project/11/

jenkins APP 9:19 AM
SUCCESS: Job java/final-project build 13
More info at: http://100.126.49.70:8080/job/java/job/final-project/13/

✅ 1



```
[jenkins@Server ~]$ aws ecr list-images --repository-name iti
{
    "imageIds": [
        {
            "imageDigest": "sha256:01c0d745b42e33064799c42f810d1769927d8563b83f5d3110f5b8f2f9da4899"
,
            "imageTag": "v13"
        }
    ]
}
```



Applications / java-app

APP HEALTH: Healthy

SYNC STATUS: Synced to main (1a73278)

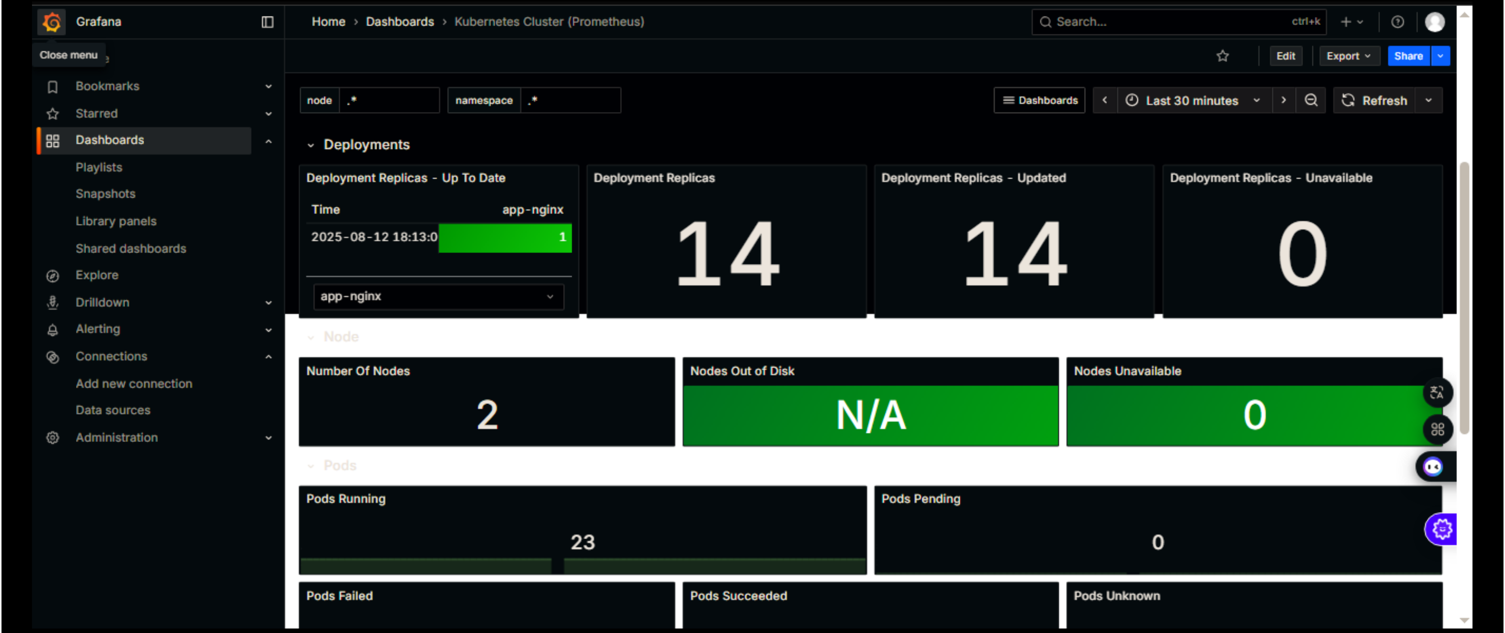
Auto sync is not enabled.
Author: diaaqassem <qassemdiaa48@gmail.com> -
Comment: update image

LAST SYNC: Sync OK to 1a73278

Succeeded 27 minutes ago (Tue Aug 12 2025 18:10:41 GMT+0300)
Author: diaaqassem <qassemdiaa48@gmail.com> -
Comment: update image

Sync status filters:

□ Synced 2
□ OutOfSync 0



Home › Dashboards › Kubernetes Cluster (Prometheus)

node .* namespace .*

Deployments

Deployment Replicas - Up To Date

Time	app-nginx
2025-08-12 18:13:0	1

app-nginx

Deployment Replicas: 14

Deployment Replicas - Updated: 14

Deployment Replicas - Unavailable: 0

Node

Number Of Nodes: 2

Nodes Out of Disk: N/A

Nodes Unavailable: 0

Pods

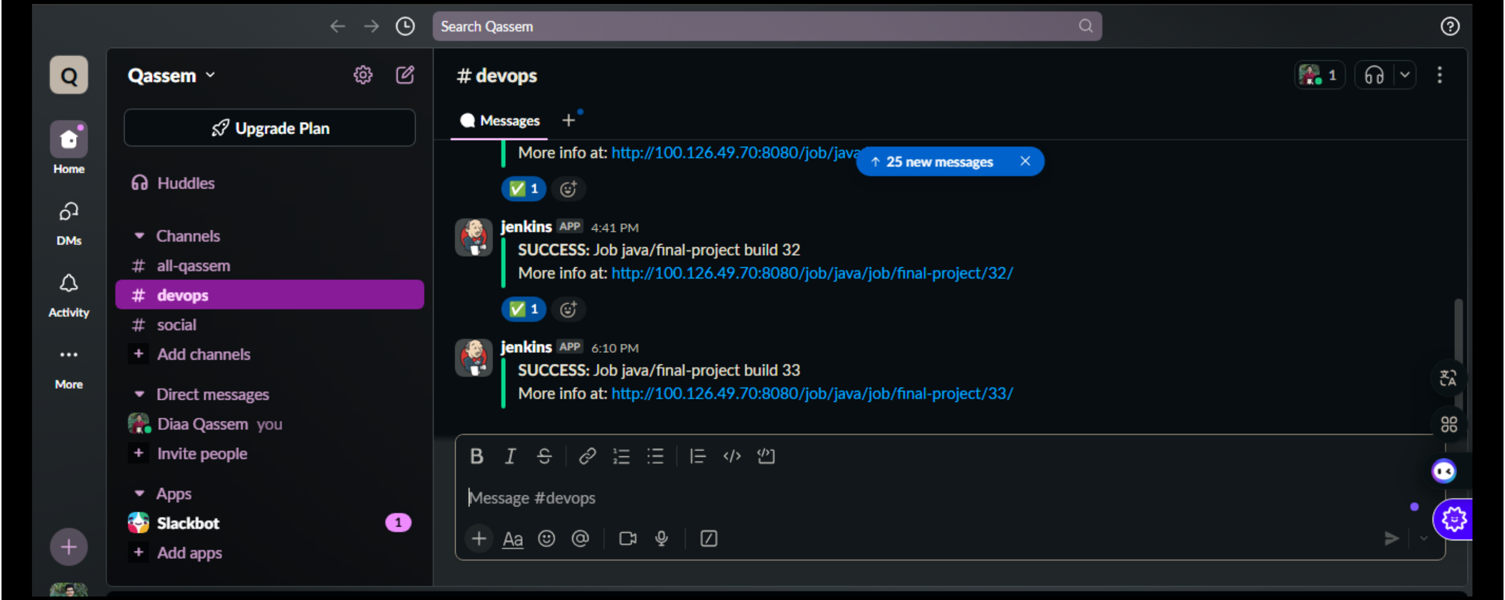
Pods Running: 23

Pods Pending: 0

Pods Failed

Pods Succeeded

Pods Unknown



devops

More info at: http://100.126.49.70:8080/job/java

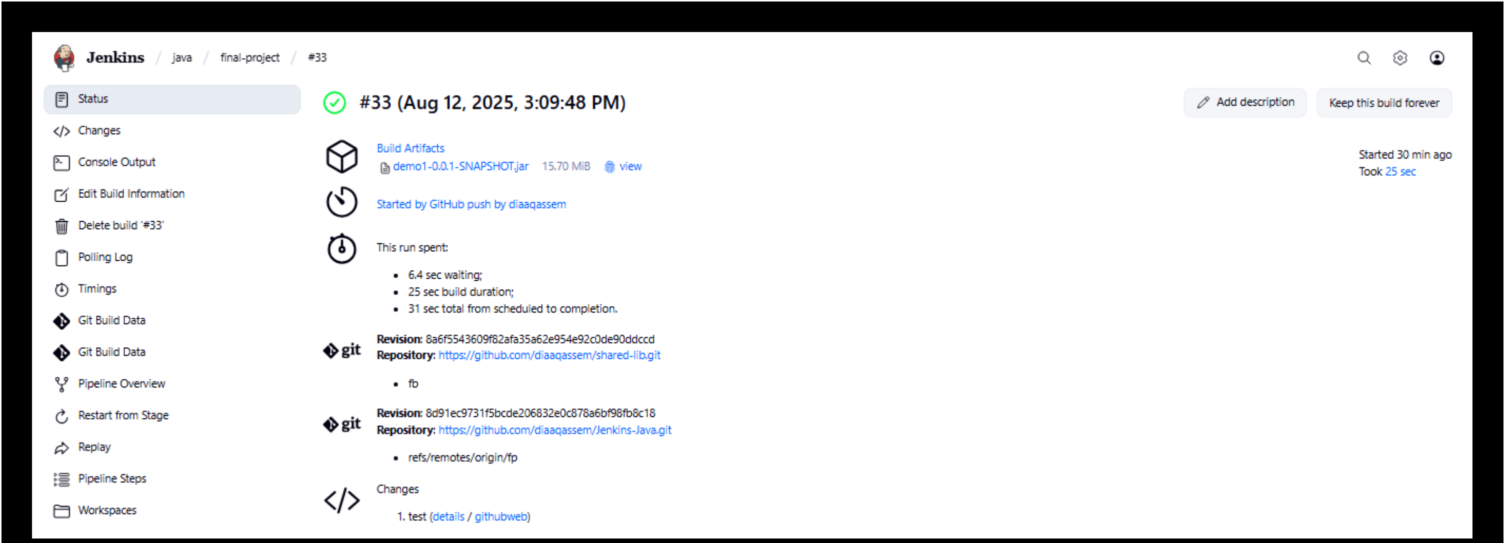
✅ 1

jenkins APP 4:41 PM
SUCCESS: Job java/final-project build 32
More info at: http://100.126.49.70:8080/job/java/job/final-project/32/

✅ 1

jenkins APP 6:10 PM
SUCCESS: Job java/final-project build 33
More info at: http://100.126.49.70:8080/job/java/job/final-project/33/

Message #devops



#33 (Aug 12, 2025, 3:09:48 PM)

Started 30 min ago
Took 25 sec

Build Artifacts
demo1-0.0.1-SNAPSHOT.jar 15.70 MiB view

Started by GitHub push by diaaqassem

This run spent:

- 6.4 sec waiting;
- 25 sec build duration;
- 31 sec total from scheduled to completion.

Revision: 8a6f5543609f82afa35a62e954e92c0de90ddccd
Repository: https://github.com/diaaqassem/shared-lib.git

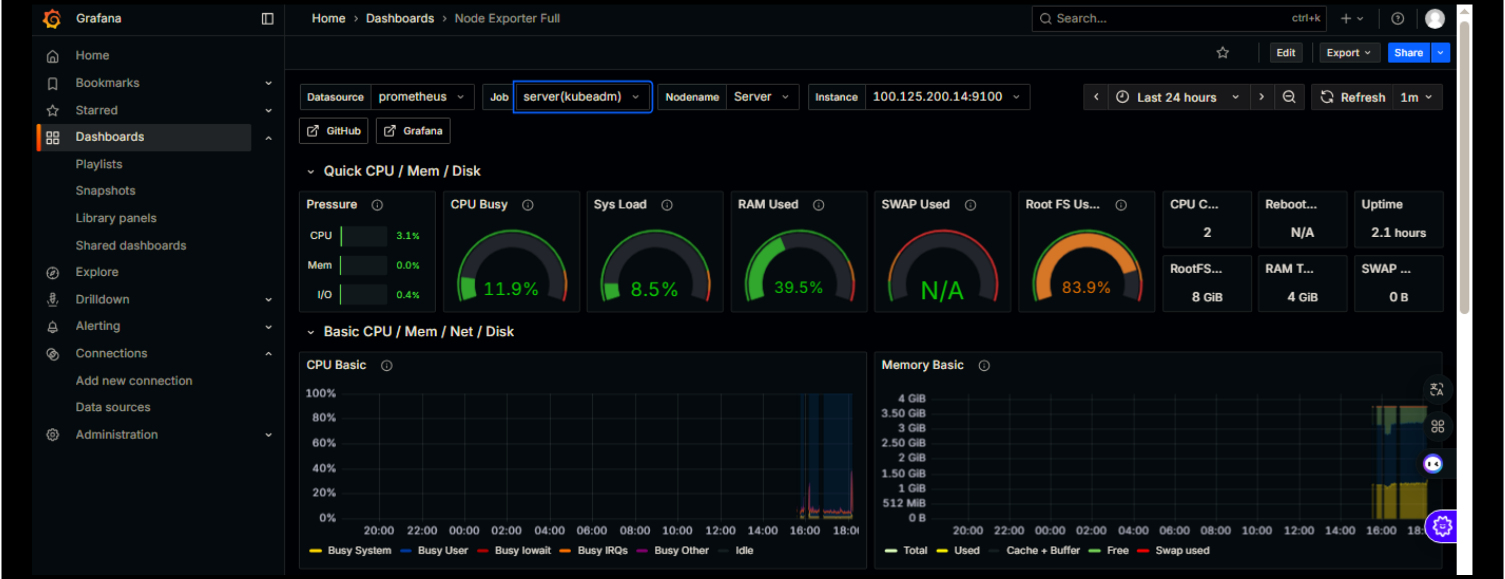
- fb

Revision: 8d91ec9731f5bcde206832e0c878a6bf98fb8c18
Repository: https://github.com/diaaqassem/Jenkins-Java.git

- refs/remotes/origin/fp

Changes

1. test (details / githubweb)



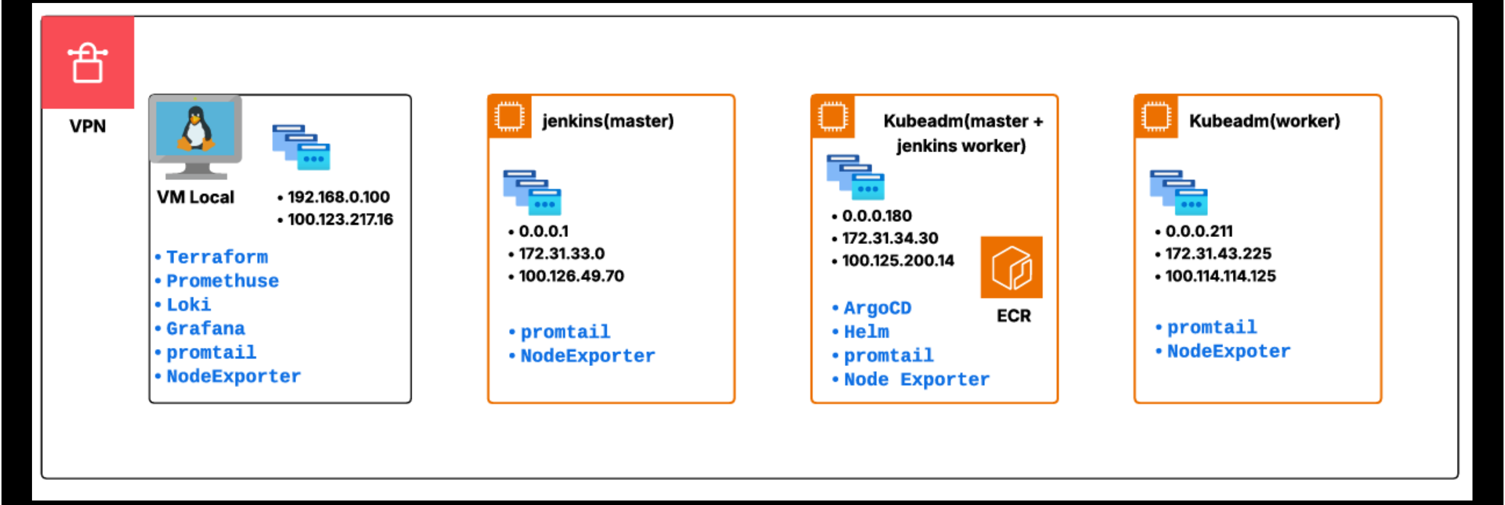
Datasource: prometheus
Job: server(kubeadm)
Nodename: Server
Instance: 100.125.200.14:9100
Last 24 hours
Refresh 1m

Quick CPU / Mem / Disk

Pressure	CPU Busy	Sys Load	RAM Used	SWAP Used	Root FS Us...	CPU C...	Reboot...	Uptime
CPU 3.1% Mem 0.0% I/O 0.4%	11.9%	8.5%	39.5%	N/A	83.9%	2	N/A	2.1 hours

RootFS...	RAM T...	SWAP ...
8 GiB	4 GiB	0 B

Basic CPU / Mem / Net / Disk



VPN

VM Local
- 192.168.0.100
- 100.123.217.16

- Terraform
- Promethuse
- Loki
- Grafana
- promtail
- NodeExporter

jenkins(master)
- 0.0.0.1
- 172.31.33.0
- 100.126.49.70

- promtail
- NodeExporter

Kubeadm(master + jenkins worker)
- 0.0.0.180
- 172.31.34.30
- 100.125.200.14

ECR

- ArgoCD
- Helm
- promtail
- Node Exporter

Kubeadm(worker)
- 0.0.0.211
- 172.31.43.225
- 100.114.114.125

- promtail
- NodeExpoter