

Pagalavan K S | DevOps-Day 05:

# Devops class guvi (DAY-5)

\*21 March 2025\*

\*Granting Jenkins Sudo Privileges\* - The jenkins ALL=(ALL) NOPASSWD: ALL entry in the sudoers file allows the Jenkins user to run any command without a password prompt.

\*Restarting SSH Services\* - Commands like `sudo systemctl restart ssh.service` and `sudo systemctl restart sshd.service` restart the SSH service, ensuring remote login functionality.

\*Installing OpenSSH Server\* - The commands `sudo apt update` and `sudo apt install openssh-server` update package lists and install the OpenSSH server for secure remote access.

\*Checking SSH Service Status\* - `sudo systemctl status ssh` checks if the SSH service is running and displays its current status.

\*Systemd Service File Lookup\* - `ls /etc/systemd/system/sshd.service` or `ls /usr/lib/systemd/system/sshd.service` helps locate the SSH daemon's systemd service file.

\*Reloading Systemd Daemon\* - `sudo systemctl daemon-reload` ensures that systemd picks up changes in service configurations without requiring a reboot.

\*Encoding Minikube Certificate\* - `cat /home/david/.minikube/ca.crt | base64 -w 0; echo` encodes the Minikube CA certificate in base64 format, likely for authentication.

**\*Changing Docker Socket Permissions\*** - `sudo chmod 666 /var/run/docker.sock` grants read and write access to all users for Docker's Unix socket, allowing non-root users to interact with Docker.

**\*Deploying Kubernetes Resources\*** - `sh 'kubectl apply -f deployment.yml --validate=false'` applies a Kubernetes deployment file, ignoring validation errors.

**\*Accessing Minikube Service\*** - `minikube service my-service --url | xargs curl` retrieves the Minikube service URL and sends an HTTP request to test its accessibility.

## Commands:

```
jenkins ALL=(ALL) NOPASSWD: ALL
sudo systemctl restart ssh.service
sudo systemctl restart sshd.service
sudo apt update
sudo apt install openssh-server
sudo systemctl restart ssh
sudo systemctl status ssh
ls /etc/systemd/system/sshd.service or ls /usr/lib/systemd/system/sshd.service
sudo systemctl daemon-reload
sudo systemctl status ssh
sudo systemctl restart ssh.service
cat /home/david/.minikube/ca.crt | base64 -w 0; echo
sudo chmod 666 /var/run/docker.sock
[https://192.168.39.226:8443](https://192.168.39.226:8443/)
sh 'kubectl apply -f deployment.yml --validate=false'
minikube service my-service --url | xargs curl
```

## Pipeline codes:

```
pipeline {
```

```
  agent any
```

```
  environment {
```

```
    DOCKER_CREDENTIALS = credentials('docker-hub-cred') // Docker Hub Credentials  
    ID
```

```
  }
```

```
  stages {
```

```
    stage('SCM') {
```

```
      steps {
```

```
        git branch: 'main', url: '<https://github.com/PagalavanS-04/guvidevopsday1.git>'
```

```
      }
```

```
    }
```

```
    stage('Build') {
```

```
      steps {
```

```
        sh "mvn clean"
```

```
        sh "mvn install"
```

```
      }
```

```
    }
```

```
    stage('Build Docker Image') {
```

```
      steps {
```

```
        script {
```

```
          sh 'docker build -t Pagalavans04/guvidevopsday1 .'
```

```

    }
  }
}

stage('Push to Docker Hub') {
  steps {
    script {
      docker.withRegistry('<https://index.docker.io/v1/>', 'docker-hub-cred') {
        sh 'docker push Pagalavans04/guvidevopsday1'
      }
    }
  }
}

}

pipeline {
  agent any

  stages {
    stage('SCM') {
      steps {
        git branch: 'main', url: '<https://github.com/PraneshC2005/DevOps_simple-web-app.git>'
      }
    }

    stage('Build-clean') {

```

```
    steps{
        sh 'mvn clean'
    }
}
stage('Build-validate') {
    steps{
        sh 'mvn validate'
    }
}
stage('Build-compile') {
    steps{
        sh 'mvn compile'
    }
}
stage('Build-test') {
    steps{
        sh 'mvn test'
    }
}
stage('Build-package') {
    steps{
        sh 'mvn package'
    }
}
stage('build to images') {
    steps {
    script{
        sh "docker build -t praneshc/webapplication ."
    }
    }
```

```
    }  
  }  
}  
stage('docker push hub') {  
  steps {  
    script{  
      withDockerRegistry(credentialsId: 'cred-2', url: '<https://index.docker.io/v1/>') {  
        sh 'docker push praneshc/webapplication'  
      }  
    }  
  }  
}  
}}
```

[illegible]

```

kubernetes@kali:~$ sudo minikube start
minikube v0.35.0 on Ubuntu 24.04 (amd64)
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.35.0 ...
Restarting existing docker container for "minikube" ...
Preparing Kubernetes v0.32.0 on Docker 27.4.1 ...
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
kubernetes@kali:~$ kubectl run my-pod --image=nginx --port=80
pod/my-pod created
kubernetes@kali:~$ kubectl get pod
NAME          READY   STATUS    RESTARTS   AGE
my-pod        0/1     ContainerCreating   0          5s
kubernetes@kali:~$ kubectl delete all --all
error: unknown shorthand flag: 'a' in -all
see 'kubectl delete --help' for usage
kubernetes@kali:~$ kubectl delete all --all
pod "my-pod" deleted
service "kubernetes" deleted
kubernetes@kali:~$ kubectl run my-pod --image=nginx --port=8080
pod/my-pod created
kubernetes@kali:~$ kubectl get pod
NAME          READY   STATUS    RESTARTS   AGE
my-pod        1/1     Running    0          56s
kubernetes@kali:~$

```



```
mugesh@DESKTOP-ELKSV5N: ~
ConfigMapOptional: <nil>
DownwardAPI: true
QoS Class: BestEffort
Node-Selectors: <none>
Tolerations: node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
               node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:
  Type      Reason      Age      From      Message
  ----      -
Normal     Scheduled   8m       default-scheduler Successfully assigned default/my-pod to minikube
Warning     Failed      7m50s    kubelet    Failed to pull image "nginx": Error response from daemon:
: Head "https://registry-1.docker.io/v2/library/nginx/manifests/latest": Get "https://auth.docker.io/token?scope=repository%3Alibrary%2Fnginx%3Apull&service=registry.docker.io": dial tcp: lookup auth.docker.io on 192.168.194.192:53: server
misbehaving
Warning     Failed      7m50s    kubelet    Error: ErrImagePull
Normal     BackOff     7m49s    kubelet    Back-off pulling image "nginx"
Warning     Failed      7m49s    kubelet    Error: ImagePullBackOff
Normal     Pulling     7m34s    kubelet    Pulling image "nginx"
Normal     Pulled      7m13s    kubelet    Successfully pulled image "nginx" in 16.224s (16.224s in
cluding waiting). Image size: 192004242 bytes.
Normal     Created     7m13s    kubelet    Created container: my-pod
Normal     Started     7m13s    kubelet    Started container my-pod

mugesh@DESKTOP-ELKSV5N:~$ docker-compose.yml snap
docker-compose.yml: command not found
mugesh@DESKTOP-ELKSV5N:~$ ls
docker-compose.yml  guvidevopsday1
mugesh@DESKTOP-ELKSV5N:~$ kubectl get pod -o wide
NAME      READY   STATUS    RESTARTS   AGE   IP           NODE      NOMINATED NODE   READINESS GATES
my-pod    1/1     Running   0           9m8s  10.244.0.7   minikube  <none>           <none>

mugesh@DESKTOP-ELKSV5N:~$

mugesh@DESKTOP-ELKSV5N:~$ ls
docker-compose.yml  guvidevopsday1
mugesh@DESKTOP-ELKSV5N:~$ kubectl get pod -o wide
NAME      READY   STATUS    RESTARTS   AGE   IP           NODE      NOMINATED NODE   READINESS GATES
my-pod    1/1     Running   0           9m8s  10.244.0.7   minikube  <none>           <none>

mugesh@DESKTOP-ELKSV5N:~$ kubectl logs my-pod
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2025/03/20 04:43:28 [notice] 1#1: using the "epoll" event method
2025/03/20 04:43:28 [notice] 1#1: nginx/1.27.4
2025/03/20 04:43:28 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2025/03/20 04:43:28 [notice] 1#1: OS: Linux 5.15.167.4-microsoft-standard-WSL2
2025/03/20 04:43:28 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/03/20 04:43:28 [notice] 1#1: start worker processes
2025/03/20 04:43:28 [notice] 1#1: start worker process 29
2025/03/20 04:43:28 [notice] 1#1: start worker process 30
2025/03/20 04:43:28 [notice] 1#1: start worker process 31
2025/03/20 04:43:28 [notice] 1#1: start worker process 32
2025/03/20 04:43:28 [notice] 1#1: start worker process 33
2025/03/20 04:43:28 [notice] 1#1: start worker process 34
2025/03/20 04:43:28 [notice] 1#1: start worker process 35
2025/03/20 04:43:28 [notice] 1#1: start worker process 36
mugesh@DESKTOP-ELKSV5N:~$
```



[illegible]