

Narendiran 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/MugeshS-04/guvidevopsday1.git>'
```

```
      }
```

```
    }
```

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

```
      steps {
```

```
        sh "mvn clean"
```

```
        sh "mvn install"
```

```
      }
```

```
    }
```

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

```
      steps {
```

```
        script {
```

```
          sh 'docker build -t mugeshs04/guvidevopsday1 .'
```

```

    }
  }
}

stage('Push to Docker Hub') {
  steps {
    script {
      docker.withRegistry('<https://index.docker.io/v1/>', 'docker-hub-cred') {
        sh 'docker push mugeshs04/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]

```
naren@DESKTOP-3ELNUUV:~$ minikube start
🔧 minikube v1.35.0 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 v0.0.46 ...
🔄 Restarting existing docker container for "minikube" ...
❗ StartHost failed, but will try again: provision: get ssh host-port: get port 22 for "minikube": do
cker container inspect -f '{{(index (index .NetworkSettings.Ports "22/tcp") 0).HostPort}}' minikube:
exit status 1
stdout:

stderr:
template parsing error: template: :1:4: executing "" at <index (index .NetworkSettings.Ports "22/tcp")
0>: error calling index: reflect: slice index out of range

🔧 Updating the running docker "minikube" container ...
📡 Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
🔍 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
naren@DESKTOP-3ELNUUV:~$ kubectl get services --all-namespaces
NAMESPACE   NAME          TYPE          CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
default     kubernetes    ClusterIP     10.96.0.1    <none>        443/TCP    24h
```


Skip to content

ns

Dashboard > java-application >

Status

</> Changes

▶ Build Now

⚙️ Configure

🗑️ Delete Pipeline

🔍 Full Stage View

📋 Stages

✎ Rename

❓ Pipeline Syntax

Builds

Filter

Today

#7 8:41 AM

#6 8:28 AM

#5 8:27 AM

#4 8:15 AM

java-application

Add description

Stage View

Average stage times:
(full run time: ~31s)

	SCM	Build-clean	Build-validate	Build-compile	Build-test	Build-package	build to images	docker push hub
#7 14:11 No Changes	823ms	1s	1s	2s	2s	2s	428ms	15s
#6 13:58 No Changes	833ms	1s	1s	2s	2s	2s	428ms	22s
#5 13:57 No Changes	1s	1s	1s	2s	2s	2s	465ms	408ms failed
#4 13:45 No Changes	813ms	1s	1s	2s	3s	3s	428ms	217ms failed
#3 13:42 No Changes	886ms	1s	1s	2s	2s	3s	685ms	237ms