1) Stages for deploying Jenkins using HELM

```
helm repo add jenkins https://charts.jenkins.io
helm repo update
```

In order to expose the jenkins app to the host we set the serviceType to **NodePort**
helm show values jenkins/jenkins > jenkinsValues.yml

```
# For minikube, set this to NodePort, elsewhere use LoadBalancer
# Use ClusterIP if your setup includes ingress controller
serviceType: NodePort
```

```
helm install my-jenkins jenkins/jenkins

helm upgrade my-jenkins --values jenkinsValues.yml

export NODE_PORT=$(kubectl get --namespace default -o
       jsonpath="{.spec.ports[0].nodePort}" services my-jenkins)

export NODE_IP=$(kubectl get nodes --namespace default -o
       jsonpath="{.items[0].status.addresses[0].address}")

echo http://$NODE_IP:$NODE_PORT/login

kubectl exec --namespace default -it svc/my-jenkins -c jenkins -- /bin/cat
        /run/secrets/chart-admin-password && echo
```

2) I created a pipeline job in Jenkins which runs the following

```
podTemplate(containers: [
    containerTemplate(name: 'ubuntu', image: 'ubuntu', command: 'sleep', args:
'99d'),
  ]) {

    node(POD_LABEL) {
        stage('Get an Ubuntu image') {
            git 'https://github.com/jenkinsci/kubernetes-plugin.git'
            container('ubuntu') {
                stage('Run command df -h') {
                    sh 'df -h'
                }
            }
        }

    }
}
```

3) Jenkins is deployed on K8S with k8s plugin installed on jenkins so that a job running on Jenkins will be able to create a pod with an image and run it and after running the pod can be deleted and rerun at a later stage or Jenkins job trigger

Bonus

We can see inside the file values.yml of Jenkins HELM chart that

1) Enabling persistency

```
persistence:
  enabled: true
  ## A manually managed Persistent Volume and Claim
  ## Requires persistence.enabled: true
  ## If defined, PVC must be created manually before volume will be bound
  existingClaim:
  ## jenkins data Persistent Volume Storage Class
  ## If defined, storageClassName: <storageClass>
  ## If set to "-", storageClassName: "", which disables dynamic provisioning
  ## If undefined (the default) or set to null, no storageClassName spec is
  ##   set, choosing the default provisioner.  (gp2 on AWS, standard on
  ##   GKE, AWS & OpenStack)
  ##
  storageClass:
  annotations: {}
  labels: {}
  accessMode: "ReadWriteOnce"
  size: "8Gi"
  volumes:
  #  - name: nothing
  #    emptyDir: {}
  mounts:
  #  - mountPath: /var/nothing
  #    name: nothing
  #    readOnly: true
```

2) Run as **non** root user

```
# Set runAsUser to 1000 to let Jenkins run as non-root user
# 'jenkins' which exists in 'jenkins/jenkins' docker image.
# When setting runAsUser to a different value than 0 also
# set fsGroup to the same value:
runAsUser: 1000
fsGroup: 1000
```

Jenkins

Dashboard ▸ run-ubuntu-pod ▸

- Back to Dashboard
- Status
- Changes
- Build Now
- Configure
- Delete Pipeline
- Full Stage View
- Rename
- Pipeline Syntax

Build History trend

Filter builds...

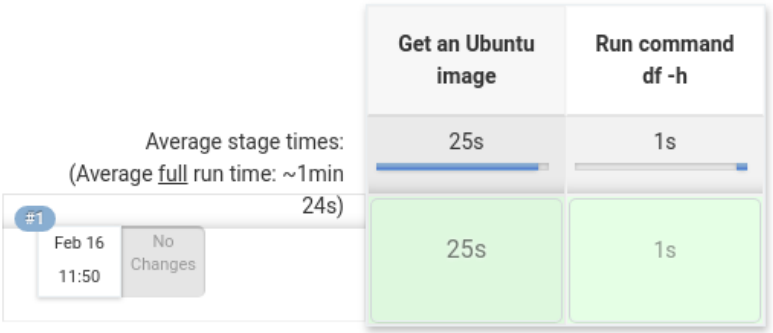
#1 Feb 16, 2022, 9:50 AM

Atom feed for all Atom feed for failures

Pipeline run-ubuntu-pod

Recent Changes

Stage View



Permalinks

- Last build (#1), 1 min 45 sec ago
- Last stable build (#1), 1 min 45 sec ago
- Last successful build (#1), 1 min 45 sec ago
- Last completed build (#1), 1 min 45 sec ago