**configmap**

* Vi sample.conf ( create one config file )
* Kubectl create configmap mymap –from-file=sample.conf ( creating one object and maping with config file )
* Kubectl get configmap
* Kubectl describe configmap mymap
* Vi deployconfigmap.yml ( create one file )

( eg:- apiVersion: v1

kind: Pod

metadata:

name: myvolconfig

spec:

containers:

- name: c1

image: centos

command: ["/bin/bash", "-c", "while true; do echo Technical-Guftgu; sleep 5 ; done"]

volumeMounts:

- name: testconfigmap

mountPath: "/tmp/config" # the config files will be mounted as ReadOnly by default here

volumes:

- name: testconfigmap

configMap:

name: mymap # this should match the config map name created in the first step

items:

- key: sample.conf

path: sample.conf )

* Kubectl apply -f deployconfigmap.yml
* Kubectl get pods
* Kubectl exec myvolconfig -it -- /bin/bash ( go inside the container )
* Cd /tmp
* Ls ( you will see config directory )
* Cd config/
* Ls ( o/p:- sample.conf now you can see sample file or config file you created before )
* Exit ( exit from container )
* Kubectl delete -f deployconfigmap.yml
* Vi deployenv.yml

( eg:- apiVersion: v1

kind: Pod

metadata:

name: myenvconfig

spec:

containers:

- name: c1

image: centos

command: ["/bin/bash", "-c", "while true; do echo Technical-Guftgu; sleep 5 ; done"]

env:

- name: MYENV # env name in which value of the key is stored

valueFrom:

configMapKeyRef:

name: mymap # name of the config created

key: sample.conf )

* Kubectl apply -f deployenv.yml
* Kubectl get pods
* Kubectl exec myenvconfig -it -- /bin/bash ( go inside the container )
* Env ( you will see that config file because you created file with env variable so it will only read )
* Echo $MYENV ( output of configmap )

**Secrets**

* Echo “root” > username.txt; echo “admin@123” > password.txt ( creating one username file and adding root in that & creating password file and adding password in that file )
* Cat username.txt ( you will see username )
* Cat password.txt ( you will see password you created in this file )
* Kubectl create secret generic mysecret –from-file=username.txt –from-file=password.txt ( creating secret file and add password and username )
* Kubectl get secret ( you can see the name of file but cannot see the contain in that file)
* Kubectl describe secret mysecret ( o/p:- you can see the size of file 7 bytes but cannot see the contain in that )
* Vi deploysecret.yml

( eg:- apiVersion: v1

kind: Pod

metadata:

name: myvolsecret

spec:

containers:

- name: c1

image: centos

command: ["/bin/bash", "-c", "while true; do echo Technical-guftgu; sleep 5 ; done"]

volumeMounts:

- name: testsecret

mountPath: "/tmp/mysecrets" # the secret files will be mounted as ReadOnly by default here

volumes:

- name: testsecret

secret:

secretName: mysecret )

* Kubectl apply -f deploysecret.yml
* Kubectl get pods
* Kubectl exec myvolsecret -it -- /bin/bash
* Cd /tmp
* Ls ( you will see my secrets file )
* Cd mysecrets/
* Ls ( now you can see password.txt and username.txt file you created in secret files )
* Cat username.txt ( now you can see the contain because you are in container )
* Cat password.txt