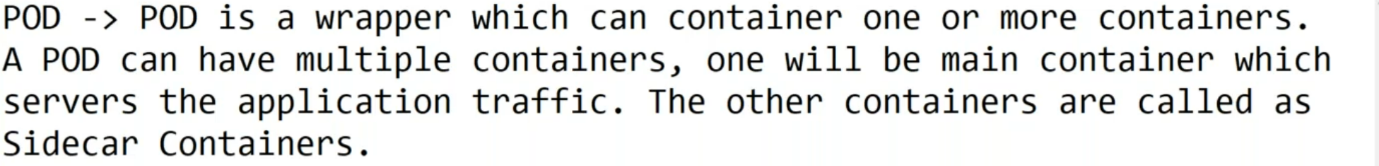
**5.Pods-ReplicaSets-Deployments**

--- in this, we will discuss about pods, replicaset, deployment, services.

--- how many containers I can put in a pod…?

You can put as many containers you want, the recommended is 1 unless you have a justification using side car container

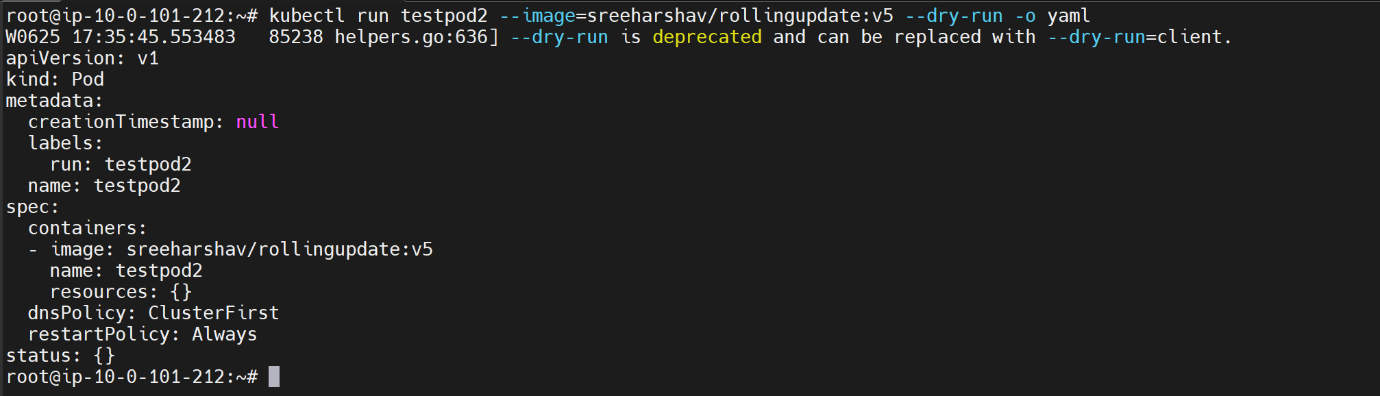
**Pods**



**Kubectl command to generate pod yml file**

**# generate pod yaml file from kubernetes command**

--- kubectl run testpod2 --image=sreeharshav/rollingupdate:v5 --dry-run -o yaml



**Container logs checking**

--- **how do you check the logs of the container…?**

It is completely depending up on pod is ready or not. If the pod is ready then we can search for the logs using kubectl logs.

if the pod is not ready then we should use the **kubectl describe pod describe <pod name>**.

**# List the pods**

--- kubectl get pods

**# Delete all pods**

--- kubectl delete pods ---all

**replication set**

--- **what replication set do…?**

At any given time, the replication set will maintain required number of pods. If any pod is not responding or crashed then replication set will replace those pods with new pods.

--- replicationSet.yml

apiVersion: apps/v1

kind: ReplicaSet

metadata:

  name: frontend

  labels:

    app: guestbook

    tier: frontend

spec:

  # modify replicas according to your case

  replicas: 3

  selector:

    matchLabels:

      tier: frontend

  template:

    metadata:

      labels:

        tier: frontend

    spec:

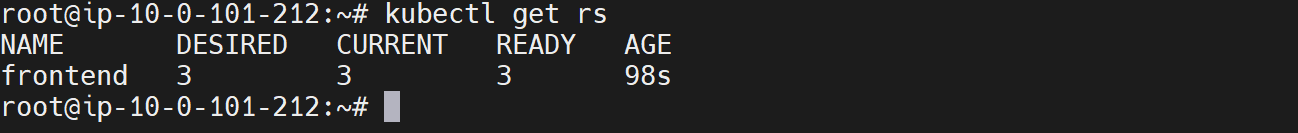
      containers:

      - name: rollingupdatev5

        image: sreeharshav/rollingupdate:v5

**# List replication set**

--- kubectl get rs



--- **note** – replication set manage pods based on their labels. If you add pod whose name is same replication set pod then it will delete one of the pods because of the same label name and it should maintain required number of pods.

**Expose replication set using NodePort**

**# Expose replication set.**

--- kubectl expose rs frontend --port=8000 --target-port=80 --type=NodePort

**# List service**

--- kubectl get svc

**# Delete all replication sets**

--- kubectl delete rs --all

**Access application form browser**

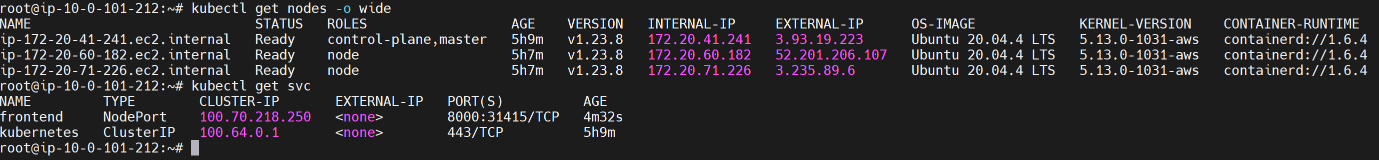
**# List the nodes in wide mode**

--- kubectl get nodes -o wide

--- **note** – copy the external ip

**# List the service**

--- kubectl get svc



**# Access application from web**

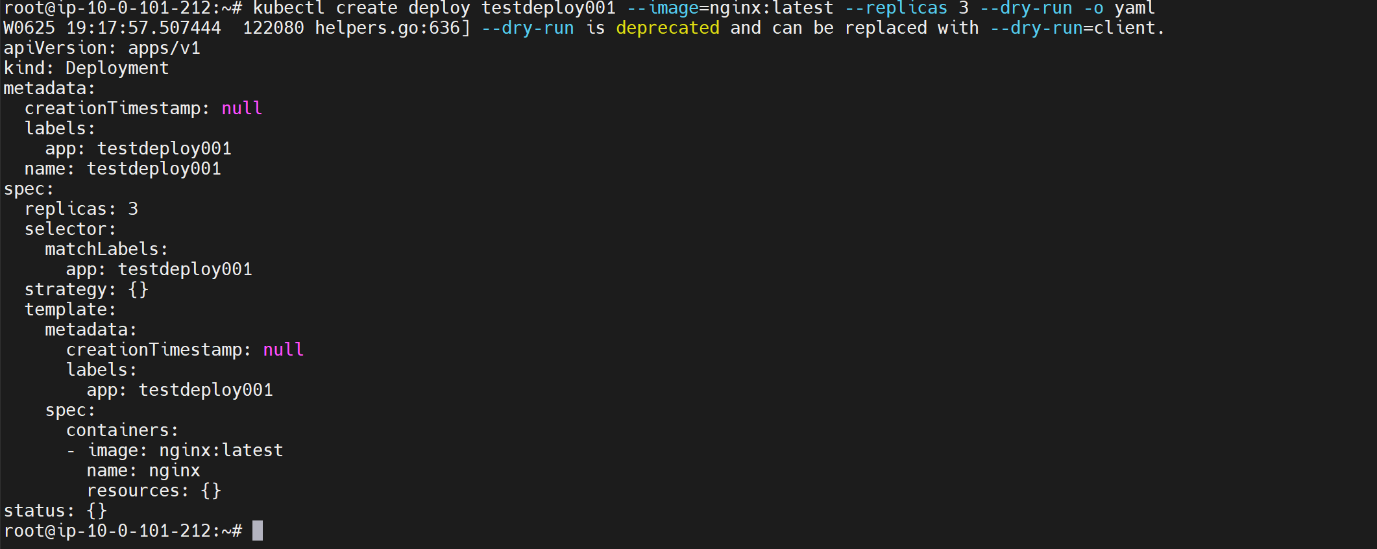
--- <http://3.235.89.6:31415/>

**Deployment**

--- **note** - deployment is superset of replication set.

**# Generate code for deployment using kubectl command.**

--- kubectl create deploy testdeploy001 --image=nginx:latest --replicas 3 --dry-run -o yaml

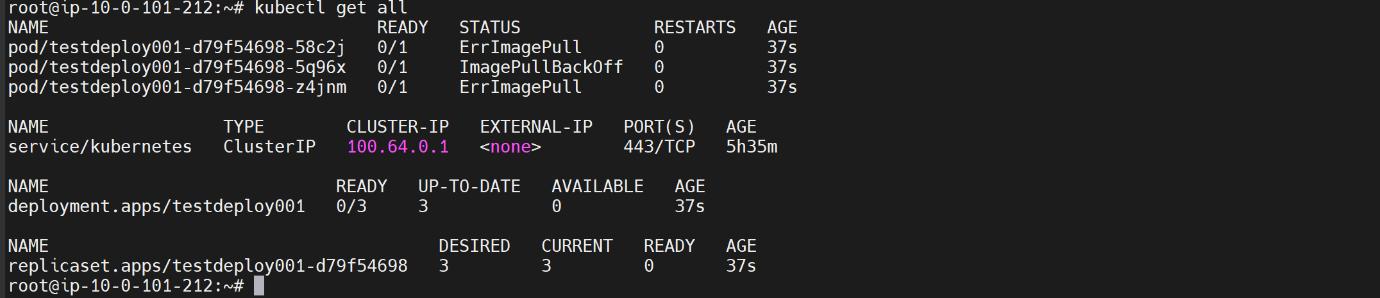


**# Create deployment using kubectl command**

--- kubectl create deploy testdeploy001 --image=nginx:latest --replicas 3

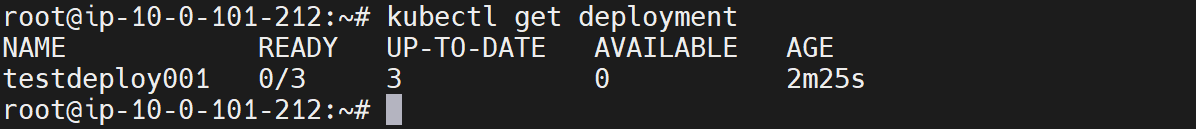
# List everything

--- kubectl get all



**# List the deployment**

--- kubectl get deployment



**# Expose deployment using NodePort**

--- kubectl expose deploy testdeploy001 --port=8000 --target-port=80 --type=NodePort

**# Want to know the status of the deployment**

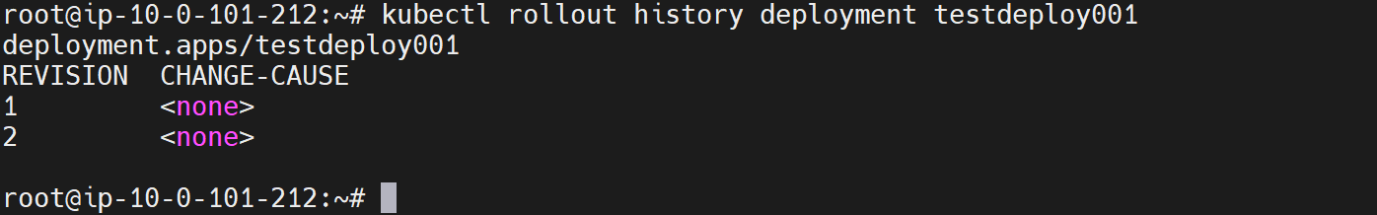
--- kubectl rollout status deployment testdeploy001

**# Pause deployment**

--- kubectl rollout pause deployment testdeploy001

**# List rollout history**

--- kubectl rollout history deployment testdeploy001



**Rollout undo**

**# Undo the deployment to previous version**

--- kubectl rollout undo deployment testdeploy001

--- **note** – the application will go to the previous version.

**Record your deployment**

**# Want to record your deployment**

--- kubectl create deploy testdeploy001 --image=nginx:latest –record=true

**Upgrade application form old to new with new image**

--- **Note** – in real time, when we want to deploy new application in the cluster then we do this procedure.

**# Upgrade application from**

--- kubectl edit deployment testdeploy001

--- **note** – replace old image with new image, by default the deployment will use rollingupdate so there is no downtime.

**Access application form browser**

**# List the nodes in wide mode**

--- kubectl get nodes -o wide

--- note – copy the external ip

**# List the service**

--- kubectl get svc

**# Access application from web**

--- <http://3.235.89.6:31415/>