

Day 3

You may like these articles - go thru when you find time

<https://mvallim.github.io/kubernetes-under-the-hood/documentation/kube-flannel.html>

<https://docs.tigera.io/calico/latest/reference/architecture/overview>

<https://yuminlee2.medium.com/kubernetes-weave-net-cni-plugin-810849203c73>

<https://www.techtarget.com/searchitoperations/tutorial/Step-by-step-guide-Get-started-with-Weave-for-Kubernetes#:~:text=Weave%20Net%2C%20often%20simply%20called,to%20talk%20to%20each%20other.>

<https://kubernetes.io/docs/tasks/administer-cluster/network-policy-provider/weave-network-policy/>

<https://granulate.io/blog/kubernetes-architecture-beginner-guide/>

References - Must read to understand Kubernetes/OpenShift Network Model

<https://www.tkng.io/>

<https://www.caseyc.net/cni-talk-kubecon-18.pdf>

<https://www.altoros.com/blog/kubernetes-networking-writing-your-own-simple-cni-plug-in-with-bash/>

<https://loggingood.github.io/kubernetes/cni/2016/05/14/netns-and-cni.html>

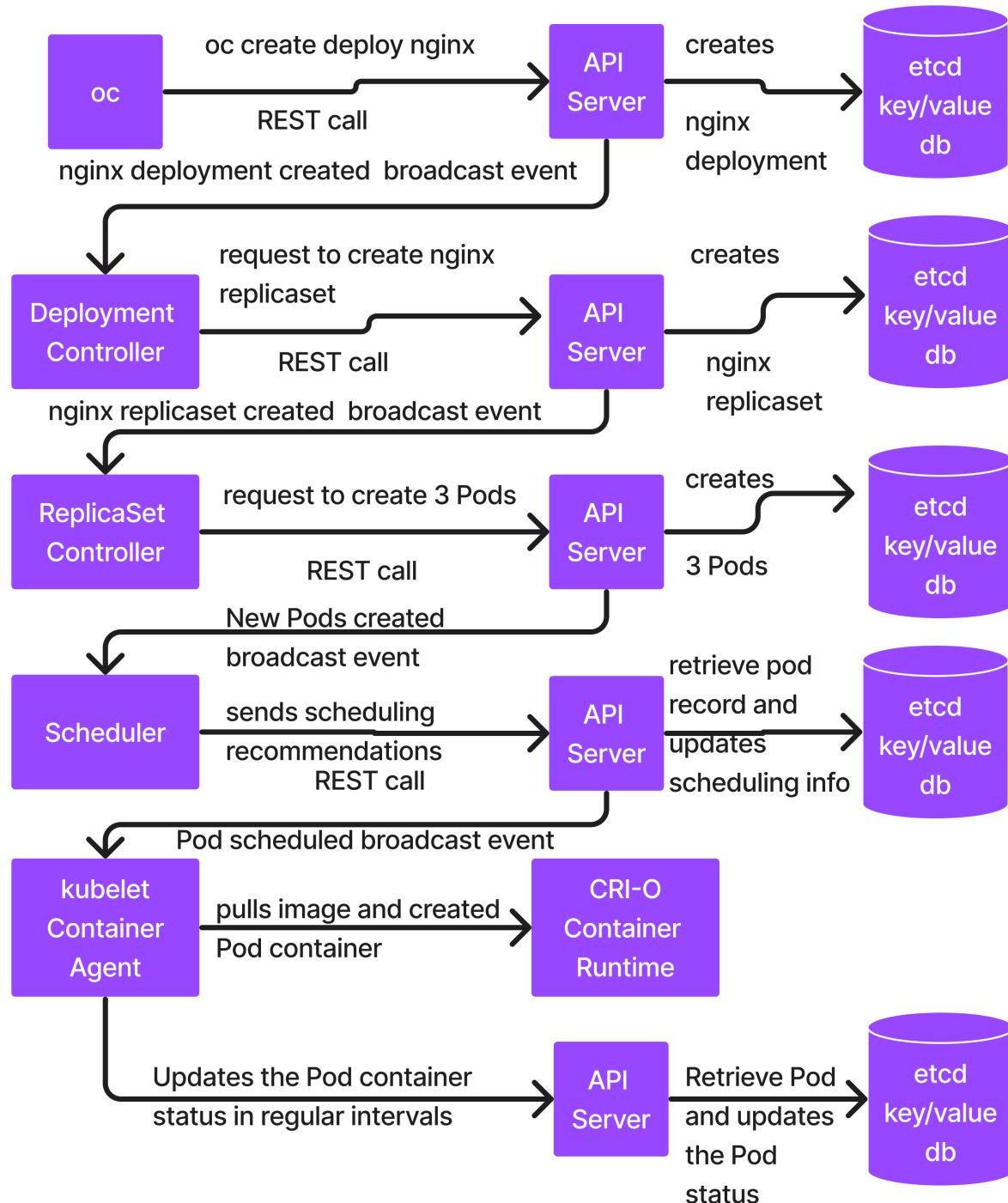
<https://loggingood.github.io/kubernetes/cni/2016/05/15/bagpipe-gobgp.html>

<https://doubtv.com/nfvpe/2017/06/22/cni-tutorial/>

Info - What happens when we deploy an application in OpenShift with the below command

```
oc create deployment nginx --image=bitnami/nginx:latest --replicas=3
```

- the oc client tool makes a REST call to API Server requesting it to create nginx deployment with image bitnami/nginx:latest with 3 Pods in it
- API Server receives the request from oc client tool, it then create a new Deployment record in etcd database
- API Server then sends a broadcasting event saying a New Deployment is created
- the event is received by Deployment Controller, it then sends a REST API call to API Server, requesting it to create a ReplicaSet for nginx deployment
- API Server receives the request from Deployment Controller and it creates a ReplicaSet record in etcd database
- API Server sends a broadcasting event saying a New ReplicaSet is created
- the event is received by ReplicaSet Controller, it then understands 3 Pods are mentioned in the Desired count, hence it makes REST call to API server to create 3 Pods
- the API Server creates 3 Pod records in etcd database and it sends broadcasting events say new Pod created. One such event will be broadcasted for every New Pod created.
- the scheduler receives the event and it sends scheduling recommendation for each Pod to the API Servers
- API Server receives the REST call from Scheduler, it then retrieves the existing Pod records from etcd and it updates the Pod records with the node details as recommended by Scheduler
- API Server then sends broadcasting events that Pod scheduler to do on nodes
- the kubelet container agent that runs in node where the Pod is scheduled receives the event, it then downloads the container images, creates the container and starts the container
- the kubelet then sends the status of the container running on that nodes to API Server via REST calls
- the API Servers updates the status of the Pod based on the status it received from kubelet
- the kubelet keeps sending this kind of container status updates to API Server like a heart beat fashion
- the API keeps the Pod status updated based on the status reported by kubelet



OpenShift Operators

- automating the human operators specific skill within OpenShift cluster
- Operators = Custom Resource(s) + Custom Controller(s)
- One Controller manages one type of Resource
- For example:
 - Deployment Controller manages Deployment
 - ReplicaSet Controller manages ReplicaSet

How Custom Resource(CR) are added to OpenShift cluster?

- We need to define Custom Resource Definition(CRD)
- CRDs introduce/register a new type of Custom Resource(CR) to your OpenShift cluster
- CRDs themselves doesn't add a Custom Resource(CR), we need to create them ourselves

What is an OpenShift Controller?

- it is an application that runs within the OpenShift cluster that monitors a specific type of Resource
- whenever it detects a Resource managed by the Controller is added/edited/deleted/updated, it acts
- Controller does 3 things
 1. monitors a specific Resource (Observe/Watch)
 2. compares desired vs current status
 3. Whenever there is a deviation of current status from desired state, Controllers act to ensure Current state matches the actual State

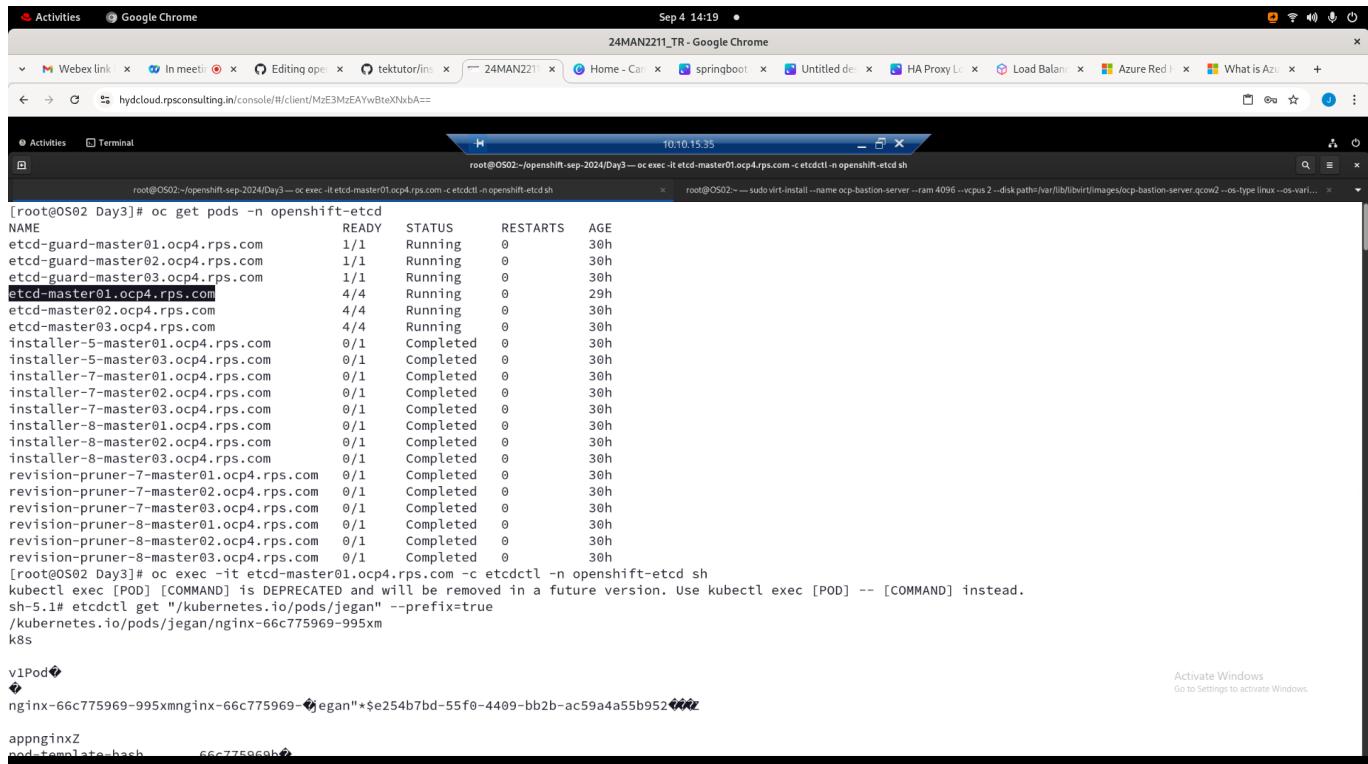
Example:

- ReplicaSet Controller monitors the ReplicaSet resources constantly
- ReplicaSet Controller registers with the APIServer (informers) for CRUD (Create, Read, Update and Delete) events of ReplicaSet
- ReplicaSet Controller will act whenever it sees the Desired Pods doesn't match the current no of Pods

Lab - Getting inside an etcd pod and understand how etcd stores deployment, replicaset, pods, etc

```
oc get pods -n openshift-etcd
oc exec -it
```

Expected output



The screenshot shows a terminal window with the following command and its output:

```
root@OS02:~/openshift-eph-2024/Day3— oc exec -it etcd-master01.ocp4.rps.com -c etcdctl -n openshift-etcd sh
```

NAME	READY	STATUS	RESTARTS	AGE
etcd-guard-master01.ocp4.rps.com	1/1	Running	0	30h
etcd-guard-master02.ocp4.rps.com	1/1	Running	0	30h
etcd-guard-master03.ocp4.rps.com	1/1	Running	0	30h
etcd-master01.ocp4.rps.com	4/4	Running	0	29h
etcd-master02.ocp4.rps.com	4/4	Running	0	30h
etcd-master03.ocp4.rps.com	4/4	Running	0	30h
installer-5-master01.ocp4.rps.com	0/1	Completed	0	30h
installer-5-master03.ocp4.rps.com	0/1	Completed	0	30h
installer-7-master01.ocp4.rps.com	0/1	Completed	0	30h
installer-7-master02.ocp4.rps.com	0/1	Completed	0	30h
installer-7-master03.ocp4.rps.com	0/1	Completed	0	30h
installer-8-master01.ocp4.rps.com	0/1	Completed	0	30h
installer-8-master02.ocp4.rps.com	0/1	Completed	0	30h
installer-8-master03.ocp4.rps.com	0/1	Completed	0	30h
revision-pruner-7-master01.ocp4.rps.com	0/1	Completed	0	30h
revision-pruner-7-master02.ocp4.rps.com	0/1	Completed	0	30h
revision-pruner-7-master03.ocp4.rps.com	0/1	Completed	0	30h
revision-pruner-8-master01.ocp4.rps.com	0/1	Completed	0	30h
revision-pruner-8-master02.ocp4.rps.com	0/1	Completed	0	30h
revision-pruner-8-master03.ocp4.rps.com	0/1	Completed	0	30h

```
[root@OS02 Day3]# oc get pods -n openshift-etcd
[root@OS02 Day3]# oc exec -it etcd-master01.ocp4.rps.com -c etcdctl -n openshift-etcd sh
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
sh-5.1# etcdctl get "/kubernetes.io/pods/jegan" --prefix=true
/kubernetes.io/pods/jegan/nginx-66c775969-995xm
k8s

v1Pod
nginx-66c775969-995xmnginx-66c775969-jegan"$e254b7bd-55f0-4409-bb2b-ac59a4a55b952
appnginxZ
pod_template.bash      66c775969

Activate Windows
Go to Settings to activate Windows.
```

Activities Google Chrome Sep 4 14:20 24MAN221_TR - Google Chrome

Webex link In meeting Editing operation tektutor/inst 24MAN221 Home - Can springboot Untitled de HA Proxy Load Balancer Azure Red What is Azu

hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNbA==

Activities Terminal 10.10.15.35

root@OS02:~/openshift-sep-2024/Day3— oc exec -it etcd-master01.ocp4.rps.com -c etcdctl -n openshift-etcd sh

```
revision-pruner-8-master02.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-8-master03.ocp4.rps.com 0/1 Completed 0 30h
[root@OS02 Day3]# oc exec -it etcd-master01.ocp4.rps.com -c etcdctl -n openshift-etcd sh
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
sh-5.1# etcdctl get "/kubernetes.io/pods/jegan" --prefix=true
/kubernetes.io/pods/jegan/nginx-66c775969-995xm
k8s

v1Pod
nginx-66c775969-995xmnginx-66c775969-nginx*$/e254b7bd-55f0-4409-bb2b-ac59a4a55b952***
```

appnginx

pod-template-hash 66c775969b*

k8s.ovn.org/pod-networks{"default": {"ip_addresses": ["10.128.2.49/23"], "mac_address": "0a:58:0a:80:02:31", "gateway_ips": ["10.128.2.1"], "routes": [{"dest": "10.128.0.0/14", "extHop": "10.128.2.1"}, {"dest": "172.30.0.0/16", "nextHop": "10.128.2.1"}, {"dest": "100.64.0.0/16", "nextHop": "10.128.2.1"}], "ip_address": "10.128.2.49/23", "gateway_ip": "10.128.2.1"}}

!k8s.v1.cni.cncf.io/network-status{}

```
  "name": "ovn-kubernetes",
  "interface": "eth0",
  "ips": [
    "10.128.2.49"
  ],
  "mac": "0a:58:0a:80:02:31",
  "default": true,
  "dns": {}
}]}!
```

restricted-v2b;c

(seccomp.security.alpha.kubernetes.io/podruntime/default)tp

Activate Windows
Go to Settings to activate Windows.

Activities Google Chrome Sep 4 14:20 24MAN221_TR - Google Chrome

Webex link In meeting Editing operation tektutor/inst 24MAN221 Home - Can springboot Untitled de HA Proxy Load Balancer Azure Red What is Azu

hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNbA==

Activities Terminal 10.10.15.35

root@OS02:~/openshift-sep-2024/Day3— oc exec -it etcd-master01.ocp4.rps.com -c etcdctl -n openshift-etcd sh

```
installer-5-master01.ocp4.rps.com 0/1 Completed 0 30h
installer-5-master03.ocp4.rps.com 0/1 Completed 0 30h
installer-7-master01.ocp4.rps.com 0/1 Completed 0 30h
installer-7-master02.ocp4.rps.com 0/1 Completed 0 30h
installer-7-master03.ocp4.rps.com 0/1 Completed 0 30h
installer-8-master01.ocp4.rps.com 0/1 Completed 0 30h
installer-8-master02.ocp4.rps.com 0/1 Completed 0 30h
installer-8-master03.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-7-master01.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-7-master02.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-7-master03.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-8-master01.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-8-master02.ocp4.rps.com 0/1 Completed 0 30h
revision-pruner-8-master03.ocp4.rps.com 0/1 Completed 0 30h
[root@OS02 Day3]# oc exec -it etcd-master01.ocp4.rps.com -c etcdctl -n openshift-etcd sh
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
sh-5.1# etcdctl get "/kubernetes.io/pods/jegan" --prefix=true
/kubernetes.io/pods/jegan/nginx-66c775969-995xm
k8s

v1Pod
nginx-66c775969-995xmnginx-66c775969-nginx*$/e254b7bd-55f0-4409-bb2b-ac59a4a55b952***
```

appnginx

pod-template-hash 66c775969b*

k8s.ovn.org/pod-networks{"default": {"ip_addresses": ["10.128.2.49/23"], "mac_address": "0a:58:0a:80:02:31", "gateway_ips": ["10.128.2.1"], "routes": [{"dest": "10.128.0.0/14", "extHop": "10.128.2.1"}, {"dest": "172.30.0.0/16", "nextHop": "10.128.2.1"}, {"dest": "100.64.0.0/16", "nextHop": "10.128.2.1"}], "ip_address": "10.128.2.49/23", "gateway_ip": "10.128.2.1"}}

!k8s.v1.cni.cncf.io/network-status{}

```
  "name": "ovn-kubernetes",
  "interface": "eth0",
  "ips": [
    "10.128.2.49"
  ],
  "mac": "0a:58:0a:80:02:31",
  "default": true,
  "dns": {}
}]}!
```

restricted-v2b;c

(seccomp.security.alpha.kubernetes.io/podruntime/default)tp

Activate Windows
Go to Settings to activate Windows.

Lab - Deploying nginx in declarative style

First of all, let's delete the existing deployments, services and routes within our project

```
oc get deploy,svc,routes
oc delete deploy/nginx deploy/hello svc/nginx svc/hello route/nginx
route/hello
oc get all
```

Expected output

Activities Google Chrome Sep 4 14:24 24MAN2211_TR - Google Chrome

Webex link fo In meeting Editing opens 24MAN2211... Home - Canvas springboot m Untitled desig HA Proxy Log Load Balance Azure Red Hat What is Azure +

hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXnbA==

Activities Terminal 10.10.15.35

root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts

root@OS02:~ -- sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-vari...

```
[root@OS02 Day3]# ls
persistent-volume README.md
[root@OS02 Day3]# mkdir declarative-manifests-scripts
[root@OS02 Day3]# cd declarative-manifests-scripts/
[root@OS02 declarative-manifests-scripts]# ls
[root@OS02 declarative-manifests-scripts]# oc delete deploy/nginx svc/nginx route/nginx
deployment.apps "nginx" deleted
service "nginx" deleted
route.route.openshift.io "nginx" deleted
[root@OS02 declarative-manifests-scripts]# oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
NAME          TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE
service/hello   ClusterIP  172.30.22.222  <none>          8080/TCP      20h

NAME          HOST/PORT      PATH        SERVICES      PORT      TERMINATION      WILDCARD
route.route.openshift.io/tekstutor-hgsc9  tekstutor.apps.ocp4.rps.com  /hello      hello      <all>      None
[root@OS02 declarative-manifests-scripts]# oc svc/hello route/hello
error: unknown command "svc/hello" for "oc"
[root@OS02 declarative-manifests-scripts]# oc delete svc/hello route/hello
service "hello" deleted
Error from server (NotFound): routes.route.openshift.io "hello" not found
[root@OS02 declarative-manifests-scripts]# oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]#
```

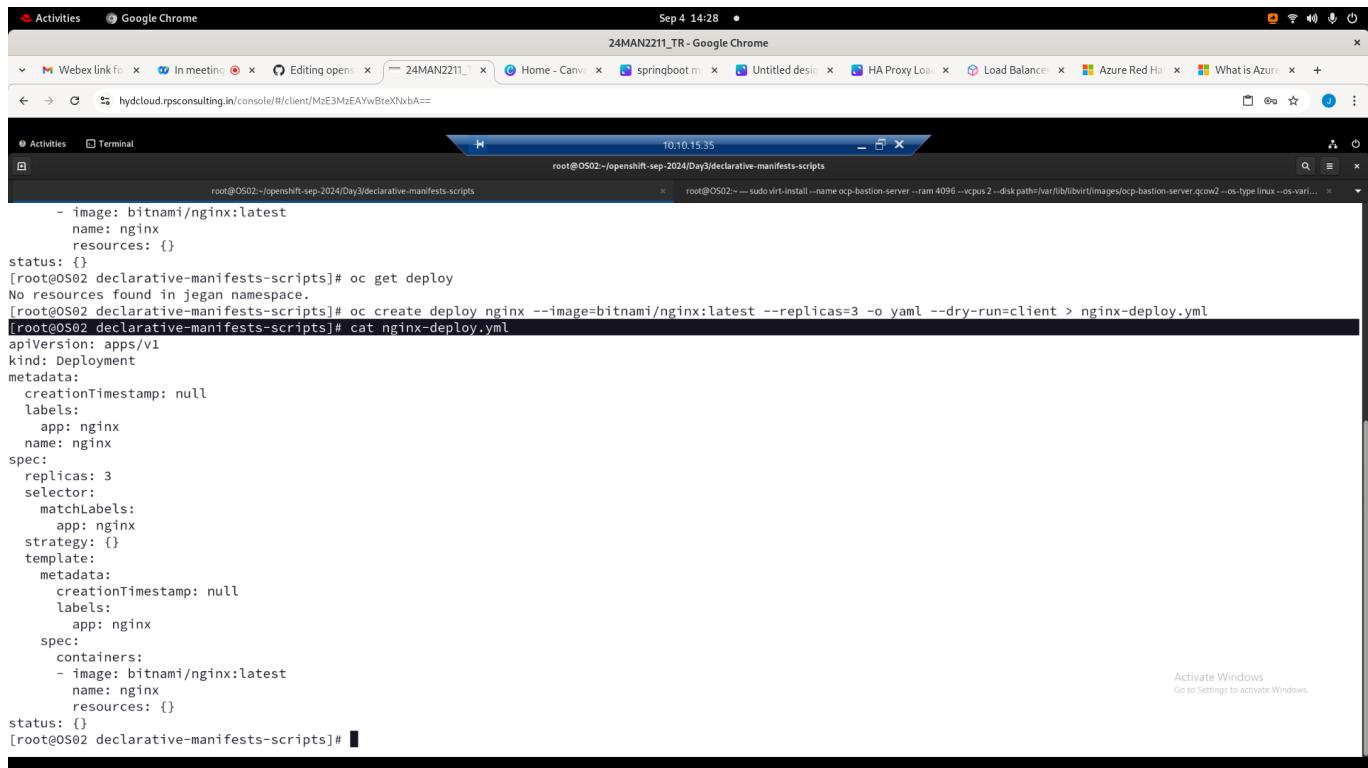
Let's generate the nginx deployment declarative script

```
oc create deployment nginx --image=bitnami/nginx:latest --replicas=3 -o yaml --dry-run=client
oc create deployment nginx --image=bitnami/nginx:latest --replicas=3 -o yaml --dry-run=client > nginx-deploy.yml
cat nginx-deploy.yml
```

Expected output

```
Activities Google Chrome Sep 4 14:27 24MAN2211_TR - Google Chrome
Webex link fo In meeting Editing opens 24MAN2211 TR - Google Chrome
hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNkbA==

Activities Terminal 10.10.15.35
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
[root@OS02 declarative-manifests-scripts]# oc create deploy nginx --image=bitnami/nginx:latest --replicas=3 -o yaml --dry-run=client
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx
          resources: {}
status: {}
[root@OS02 declarative-manifests-scripts]# oc get deploy
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]#
```



```

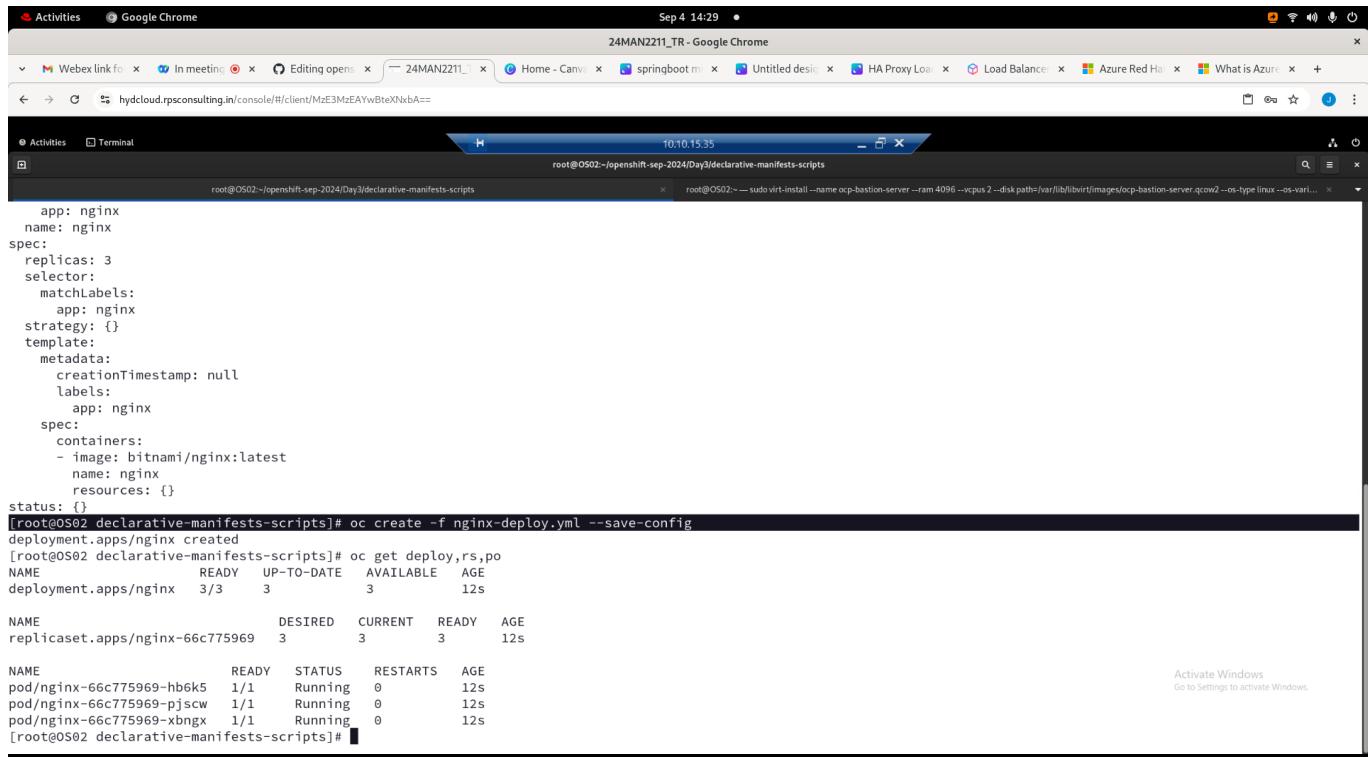
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
- image: bitnami/nginx:latest
  name: nginx
  resources: {}
status: {}
[root@OS02 declarative-manifests-scripts]# oc get deploy
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]# oc create deploy nginx --image=bitnami/nginx:latest --replicas=3 -o yaml --dry-run=client > nginx-deploy.yml
[root@OS02 declarative-manifests-scripts]# cat nginx-deploy.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx
          resources: {}
status: {}
[root@OS02 declarative-manifests-scripts]#

```

Let's create the nginx deployment in declarative style

```
oc create -f nginx-deploy.yml --save-config
```

Expected output



```

root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
- app: nginx
  name: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx
          resources: {}
status: {}
[root@OS02 declarative-manifests-scripts]# oc create -f nginx-deploy.yml --save-config
deployment.apps/nginx created
[root@OS02 declarative-manifests-scripts]# oc get deploy,rs,po
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   3/3      3           3           12s

NAME          DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-66c775969   3         3         3         12s

NAME          READY   STATUS    RESTARTS   AGE
pod/nginx-66c775969-hb6k5   1/1     Running   0          12s
pod/nginx-66c775969-pjscw   1/1     Running   0          12s
pod/nginx-66c775969-xbngx   1/1     Running   0          12s
[root@OS02 declarative-manifests-scripts]#

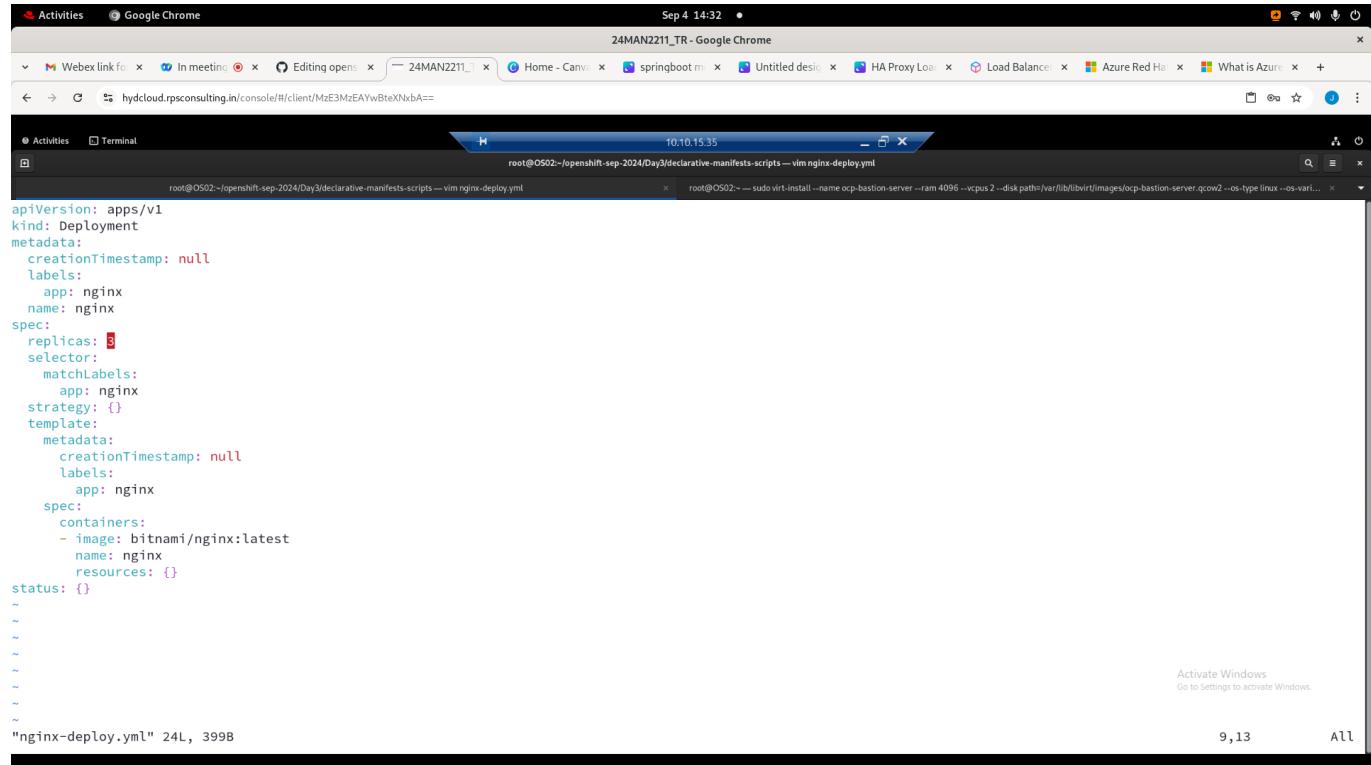
```

Lab - Scale up nginx deployment in declarative style

Let's update the nginx-deploy.yml replicas from 3 to 5 and save it before applying the changes into openshift cluster

```
cd ~/openshift-sep-2024
git pull
cd Day3/declarative-manifest-scripts
vim nginx-deploy.yml
oc apply -f nginx-deploy.yml
oc get po -w
oc get po
```

Expected output

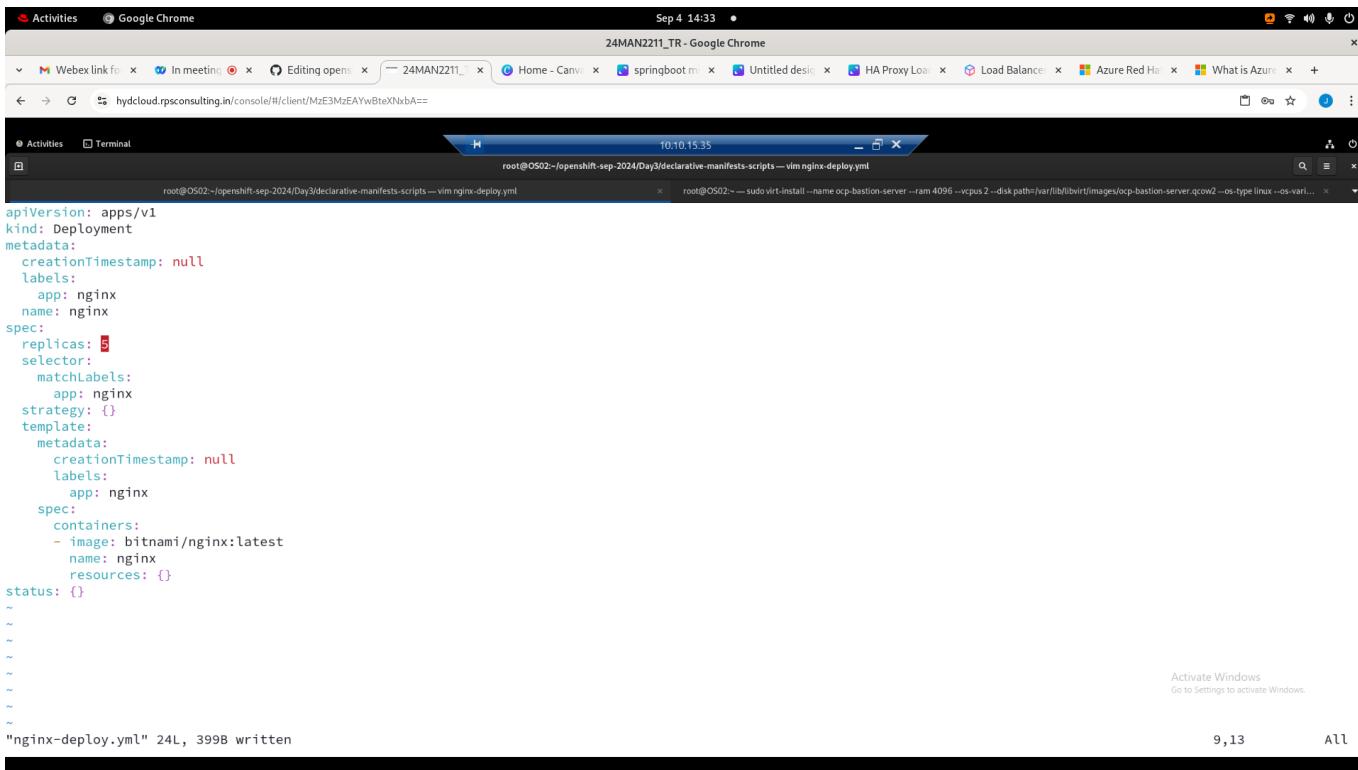


The screenshot shows a terminal window with the following content:

```
root@OS02:~/openshift-sep-2024/Day3/declarative-manifest-scripts— vim nginx-deploy.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  replicas: 5
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx
          resources: {}
status: {}
```

The terminal window is titled "root@OS02:~/openshift-sep-2024/Day3/declarative-manifest-scripts— vim nginx-deploy.yml". The status bar shows "root@OS02:~— sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=

At the bottom of the terminal, there is a status bar with "Activate Windows" and "Go to Settings to activate Windows." It also shows "9,13" and "All".

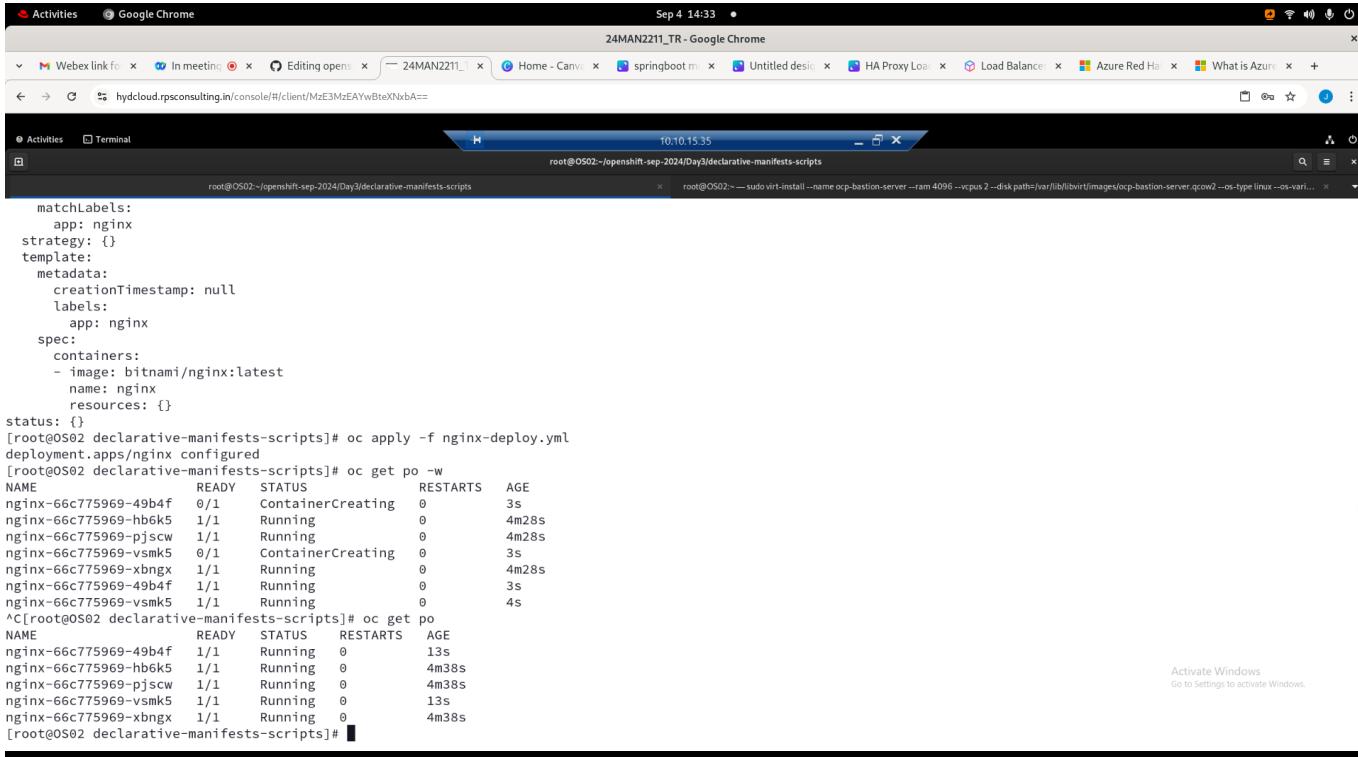


```

apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx
          resources: {}
status: {}

```

"nginx-deploy.yml" 24L, 399B written



```

matchLabels:
  app: nginx
strategy: {}
template:
  metadata:
    creationTimestamp: null
    labels:
      app: nginx
  spec:
    containers:
      - image: bitnami/nginx:latest
        name: nginx
        resources: {}
status: {}

```

[root@OS02 declarative-manifests-scripts]# oc apply -f nginx-deploy.yml

deployment.apps/nginx configured

[root@OS02 declarative-manifests-scripts]# oc get po -w

NAME	READY	STATUS	RESTARTS	AGE
nginx-66c775969-49b4f	0/1	ContainerCreating	0	3s
nginx-66c775969-hb6k5	1/1	Running	0	4m28s
nginx-66c775969-pjscw	1/1	Running	0	4m28s
nginx-66c775969-vsmk5	0/1	ContainerCreating	0	3s
nginx-66c775969-xbngx	1/1	Running	0	4m28s
nginx-66c775969-49b4f	1/1	Running	0	3s
nginx-66c775969-vsmk5	1/1	Running	0	4s

[root@OS02 declarative-manifests-scripts]# oc get po

NAME	READY	STATUS	RESTARTS	AGE
nginx-66c775969-49b4f	1/1	Running	0	13s
nginx-66c775969-hb6k5	1/1	Running	0	4m38s
nginx-66c775969-pjscw	1/1	Running	0	4m38s
nginx-66c775969-vsmk5	1/1	Running	0	13s
nginx-66c775969-xbngx	1/1	Running	0	4m38s

[root@OS02 declarative-manifests-scripts]#

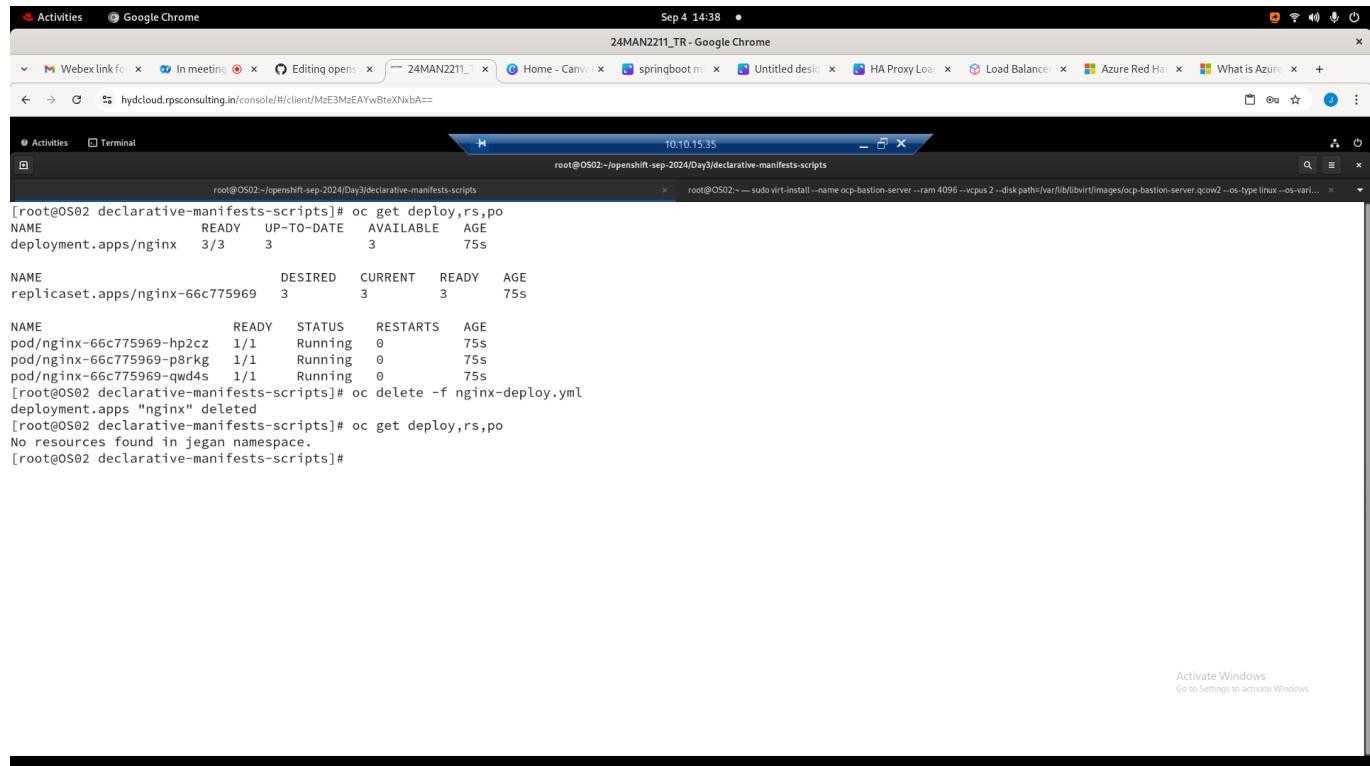
Lab - Delete a deployment in declarative style

```

oc get deploy,rs,po
oc delete -f nginx-deploy.yml
oc get deploy,rs,po

```

Expected output



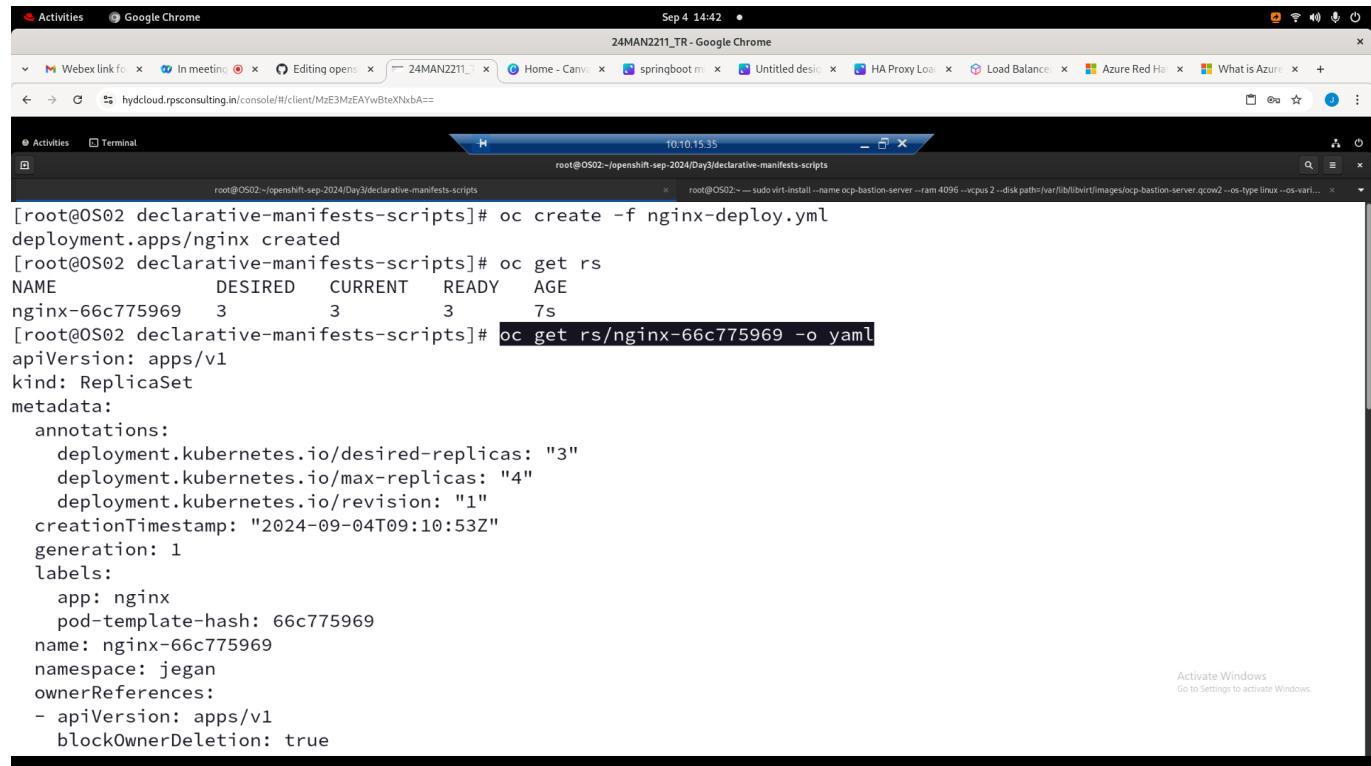
```
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts# oc get deploy,rs,po
NAME          READY  UP-TO-DATE  AVAILABLE  AGE
deployment.apps/nginx   3/3      3           3          75s
replicaset.apps/nginx-66c775969  3        3           3          75s
NAME          READY  STATUS    RESTARTS  AGE
pod/nginx-66c775969-hp2cz  1/1    Running   0          75s
pod/nginx-66c775969-p8rk  1/1    Running   0          75s
pod/nginx-66c775969-qwd4s  1/1    Running   0          75s
[root@OS02 declarative-manifests-scripts]# oc delete -f nginx-deploy.yml
deployment.apps "nginx" deleted
[root@OS02 declarative-manifests-scripts]# oc get deploy,rs,po
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]#
```

Activate Windows
Go to Settings to activate Windows.

Lab - Declarative creating replicaset without deployment

```
oc create -f nginx-deploy.yml
oc get rs
oc get rs/nginx-66c775969 -o yaml
oc get rs/nginx-66c775969 -o yaml > nginx-rs.yml
oc delete -f nginx-deploy.yml
oc get deploy,rs,po
```

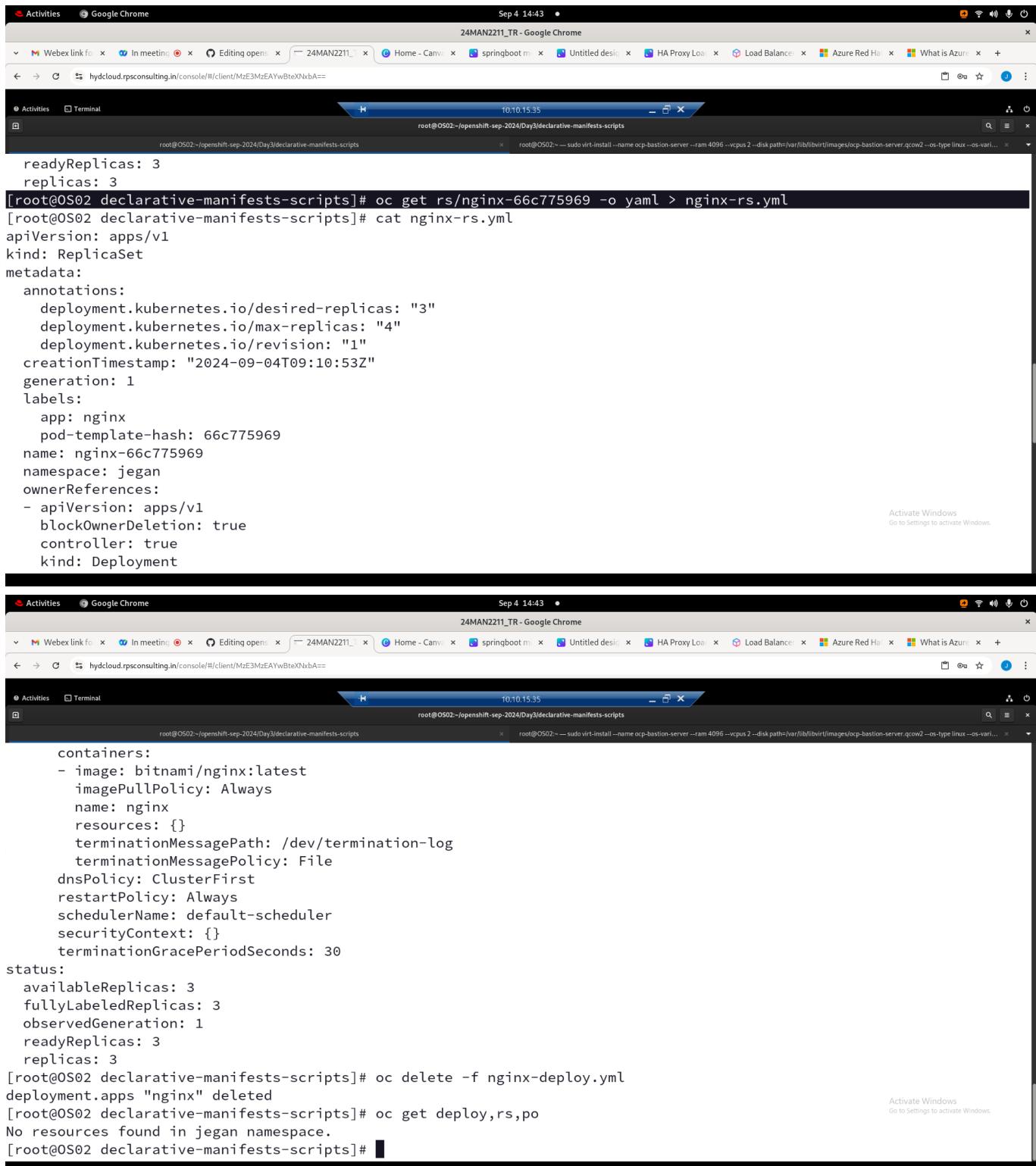
Expected output



The screenshot shows a terminal window on a Linux system (root@OS02) with the title 'Activities' and 'Terminal'. The terminal is running on port 10.10.15.35. The user is in the directory '/openshift-sep-2024/Day3/declarative-manifests-scripts'. The terminal output shows the creation of an 'nginx' deployment and its retrieval:

```
[root@OS02 declarative-manifests-scripts]# oc create -f nginx-deploy.yml
deployment.apps/nginx created
[root@OS02 declarative-manifests-scripts]# oc get rs
NAME      DESIRED   CURRENT   READY   AGE
nginx-66c775969  3         3         3       7s
[root@OS02 declarative-manifests-scripts]# oc get rs/nginx-66c775969 -o yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  annotations:
    deployment.kubernetes.io/desired-replicas: "3"
    deployment.kubernetes.io/max-replicas: "4"
    deployment.kubernetes.io/revision: "1"
  creationTimestamp: "2024-09-04T09:10:53Z"
  generation: 1
  labels:
    app: nginx
    pod-template-hash: 66c775969
  name: nginx-66c775969
  namespace: jegan
  ownerReferences:
  - apiVersion: apps/v1
    blockOwnerDeletion: true
```

At the bottom right of the terminal window, there is a watermark: 'Activate Windows' and 'Go to Settings to activate Windows'.



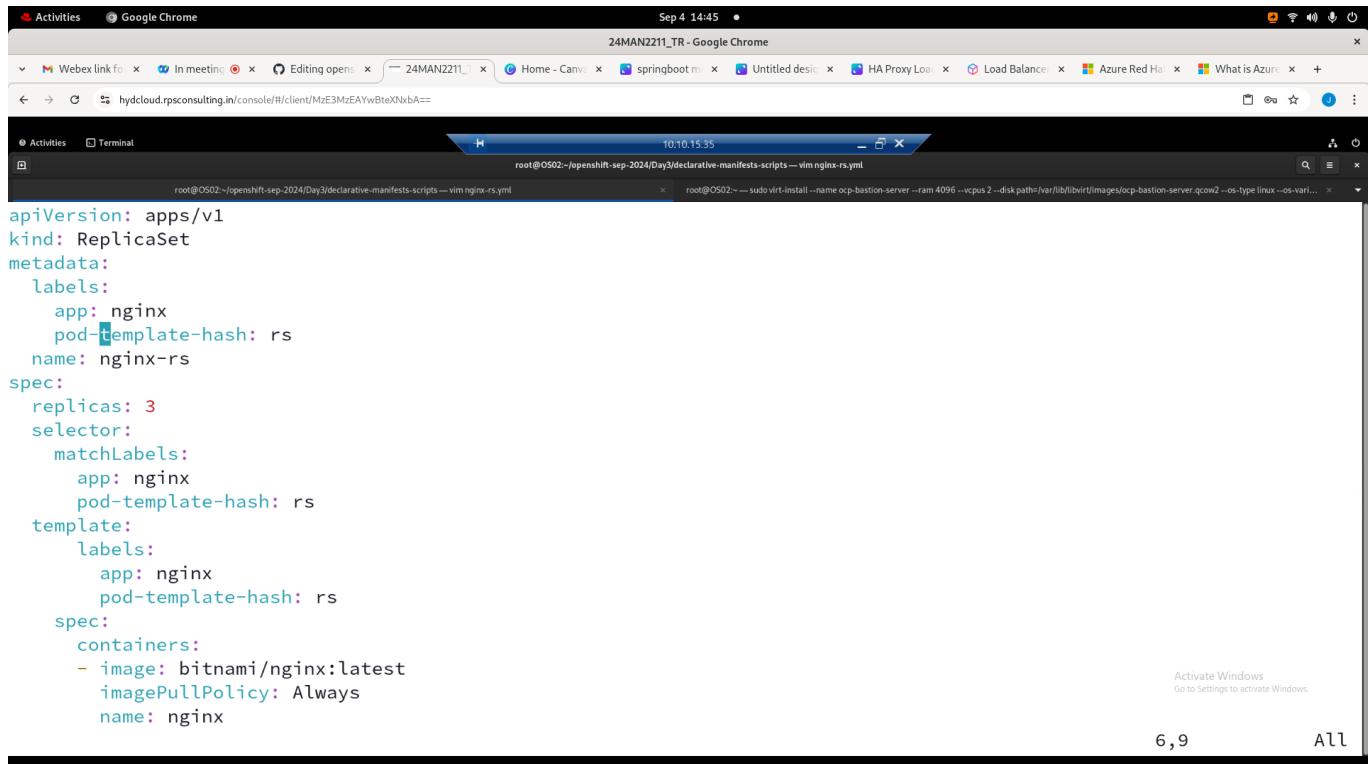
```
Sep 4 14:43 • 24MAN2211_TR - Google Chrome
Activities Google Chrome Sep 4 14:43 • 24MAN2211_TR - Google Chrome
Webex link fo In meeting Editing open 24MAN2211 TR - Google Chrome
Home - Canva springboot m Untitled desig HA Proxy Load Balance Azure Red Hat What is Azure
hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNxbA==

Activities Terminal 10.10.15.35
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
root@OS02:~ sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=redhat-8.7 --cpu host --graphics none --noautoconsole
readyReplicas: 3
replicas: 3
[root@OS02 declarative-manifests-scripts]# oc get rs/nginx-66c775969 -o yaml > nginx-rs.yml
[root@OS02 declarative-manifests-scripts]# cat nginx-rs.yml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  annotations:
    deployment.kubernetes.io/desired-replicas: "3"
    deployment.kubernetes.io/max-replicas: "4"
    deployment.kubernetes.io/revision: "1"
  creationTimestamp: "2024-09-04T09:10:53Z"
  generation: 1
  labels:
    app: nginx
    pod-template-hash: 66c775969
  name: nginx-66c775969
  namespace: jegan
  ownerReferences:
  - apiVersion: apps/v1
    blockOwnerDeletion: true
    controller: true
    kind: Deployment
Activate Windows
Go to Settings to activate Windows.

Sep 4 14:43 • 24MAN2211_TR - Google Chrome
Activities Google Chrome Sep 4 14:43 • 24MAN2211_TR - Google Chrome
Webex link fo In meeting Editing open 24MAN2211 TR - Google Chrome
Home - Canva springboot m Untitled desig HA Proxy Load Balance Azure Red Hat What is Azure
hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNxbA==

Activities Terminal 10.10.15.35
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
root@OS02:~ sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=redhat-8.7 --cpu host --graphics none --noautoconsole
containers:
- image: bitnami/nginx:latest
  imagePullPolicy: Always
  name: nginx
  resources: {}
  terminationMessagePath: /dev/termination-log
  terminationMessagePolicy: File
  dnsPolicy: ClusterFirst
  restartPolicy: Always
  schedulerName: default-scheduler
  securityContext: {}
  terminationGracePeriodSeconds: 30
status:
  availableReplicas: 3
  fullyLabeledReplicas: 3
  observedGeneration: 1
  readyReplicas: 3
  replicas: 3
[root@OS02 declarative-manifests-scripts]# oc delete -f nginx-deploy.yml
deployment.apps "nginx" deleted
[root@OS02 declarative-manifests-scripts]# oc get deploy,rs,po
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]#
```

Clean up the nginx-rs.yml as shown below



```

apiVersion: apps/v1
kind: ReplicaSet
metadata:
  labels:
    app: nginx
    pod-template-hash: rs
  name: nginx-rs
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
      pod-template-hash: rs
  template:
    labels:
      app: nginx
      pod-template-hash: rs
  spec:
    containers:
      - image: bitnami/nginx:latest
        imagePullPolicy: Always
        name: nginx

```

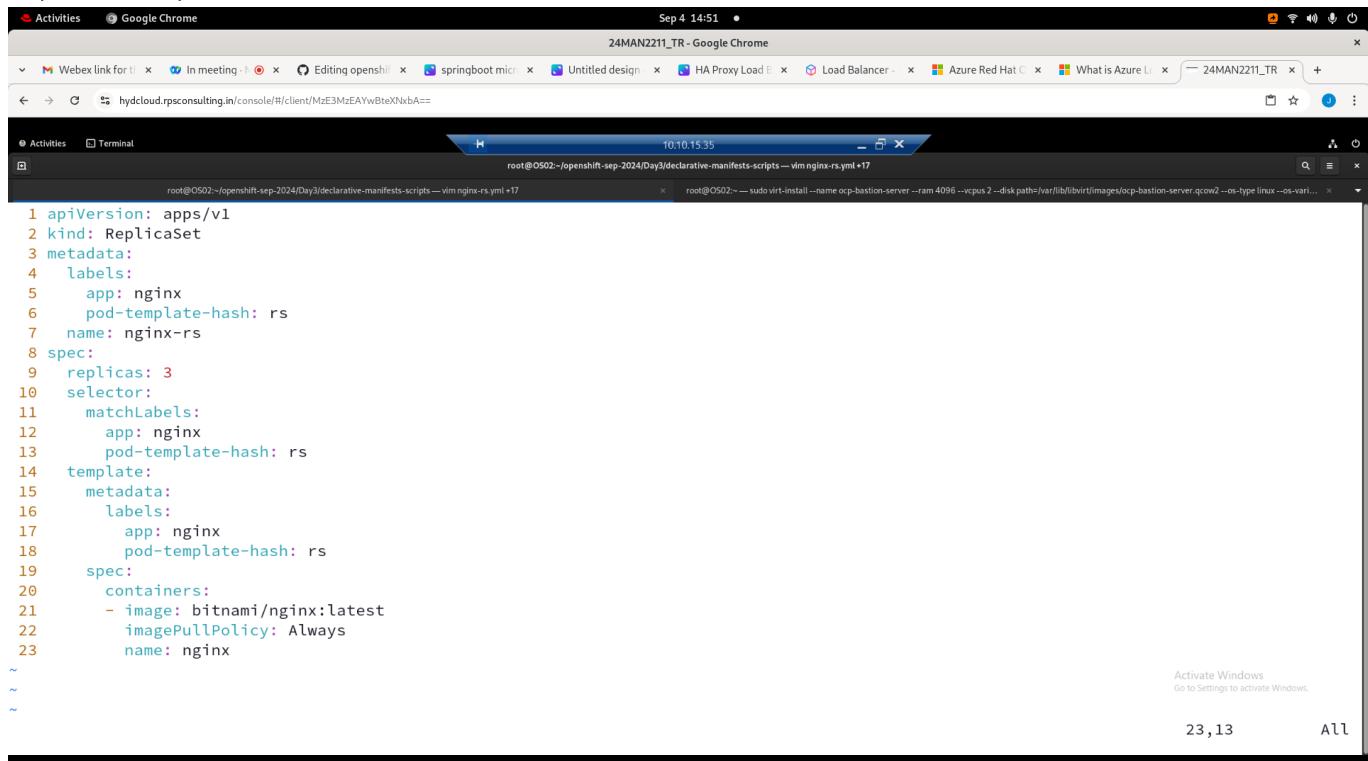
Now you may create the nginx replicaset without deployment in declarative style

```

oc create -f nginx-rs.yml --save-config
oc get deploy,rs,po

```

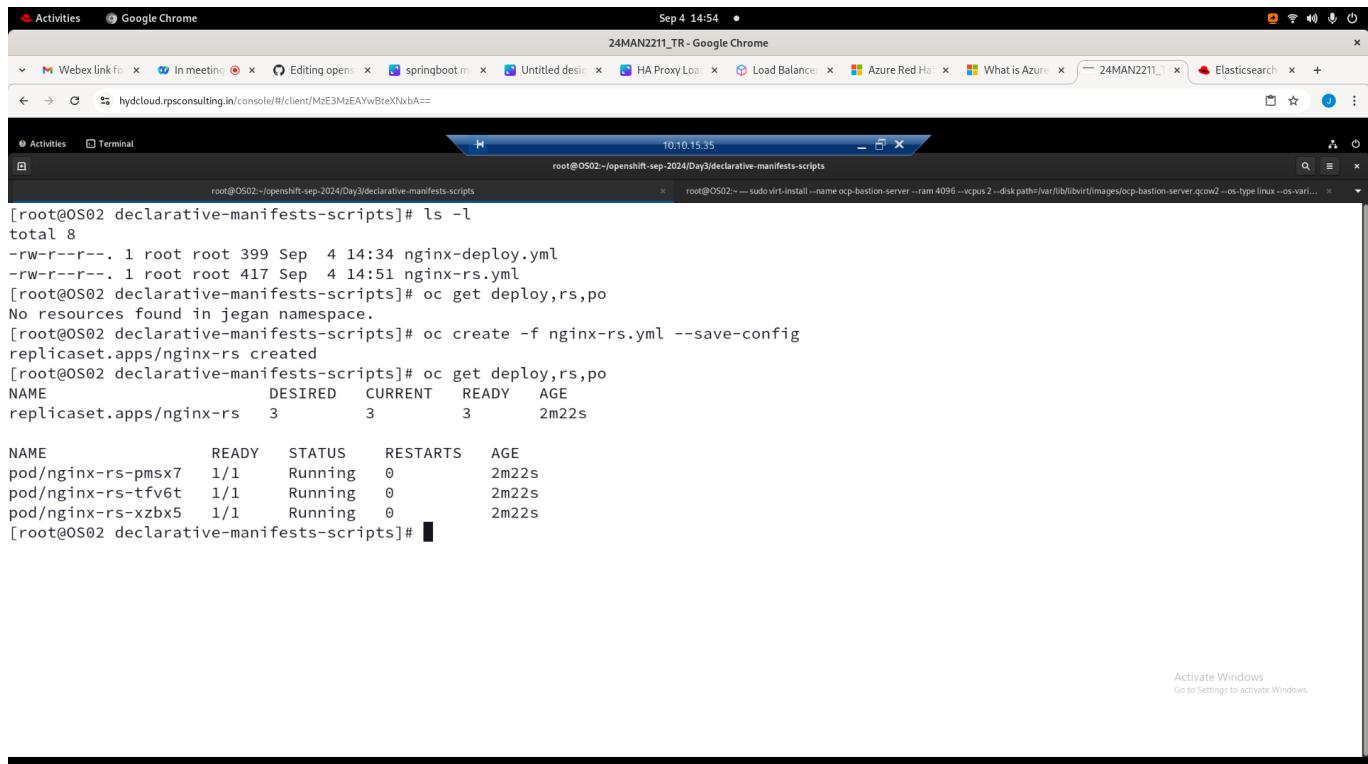
Expected output



```

1 apiVersion: apps/v1
2 kind: ReplicaSet
3 metadata:
4   labels:
5     app: nginx
6     pod-template-hash: rs
7   name: nginx-rs
8 spec:
9   replicas: 3
10  selector:
11    matchLabels:
12      app: nginx
13      pod-template-hash: rs
14  template:
15    metadata:
16      labels:
17        app: nginx
18        pod-template-hash: rs
19  spec:
20    containers:
21      - image: bitnami/nginx:latest
22        imagePullPolicy: Always
23        name: nginx

```

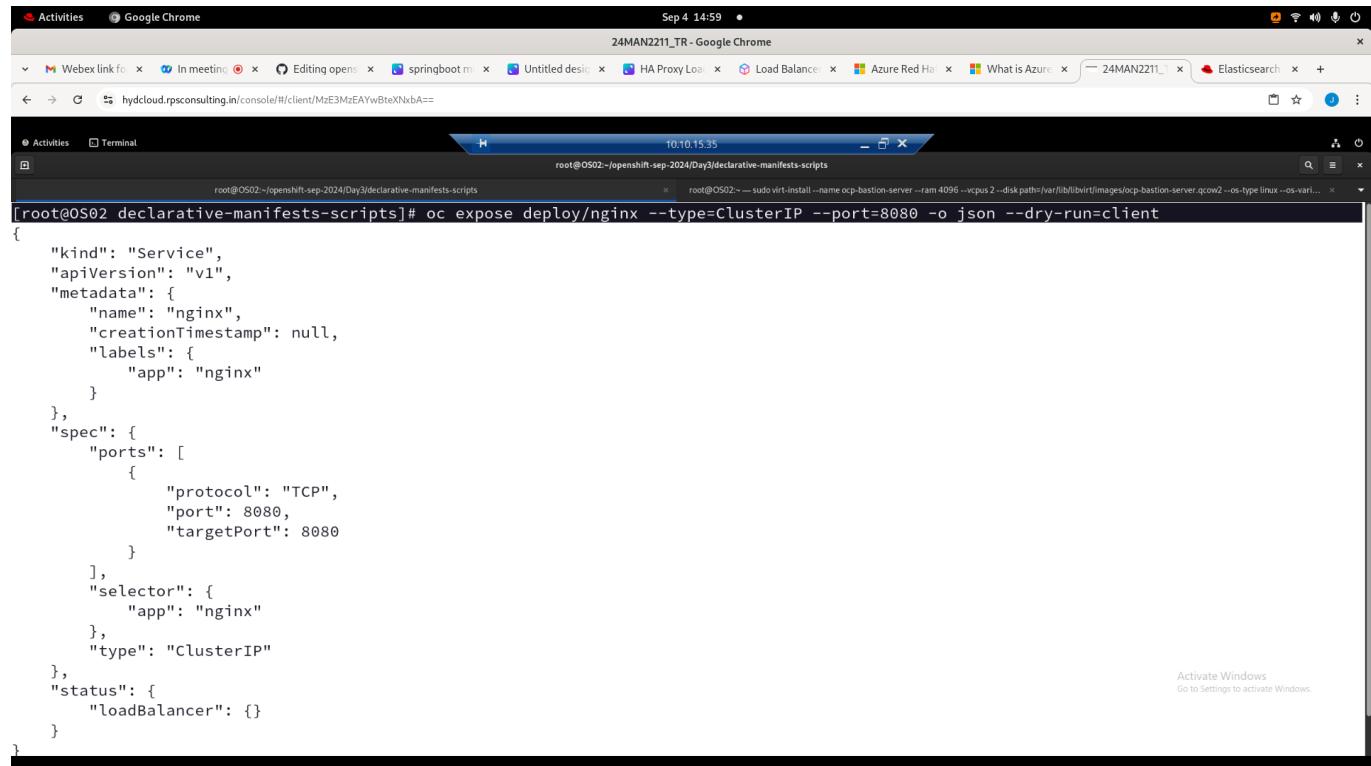


```
[root@OS02 declarative-manifests-scripts]# ls -l
total 8
-rw-r--r--. 1 root root 399 Sep  4 14:34 nginx-deploy.yml
-rw-r--r--. 1 root root 417 Sep  4 14:51 nginx-rs.yml
[root@OS02 declarative-manifests-scripts]# oc get deploy,rs,po
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]# oc create -f nginx-rs.yml --save-config
replicaset.apps/nginx-rs created
[root@OS02 declarative-manifests-scripts]# oc get deploy,rs,po
NAME      DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-rs   3         3         3        2m22s
NAME      READY     STATUS    RESTARTS   AGE
pod/nginx-rs-pmsx7   1/1      Running   0          2m22s
pod/nginx-rs-tfv6t   1/1      Running   0          2m22s
pod/nginx-rs-xzbx5   1/1      Running   0          2m22s
[root@OS02 declarative-manifests-scripts]#
```

Lab - Creating a ClusterIP internal service in declarative style

```
oc delete -f nginx-rs.yml
oc create -f nginx-deploy.yml
oc get deploy
oc expose deploy/nginx --type=ClusterIP --port=8080 -o json
oc expose deploy/nginx --type=ClusterIP --port=8080 -o yaml
oc expose deploy/nginx --type=ClusterIP --port=8080 -o yaml > nginx-
clusterip-svc.yml
oc create -f nginx-clusterip-svc.yml --save-config
oc get svc
```

Expected output



The screenshot shows a terminal window titled 'Activities' with the title bar '24MAN2211_TR - Google Chrome'. The terminal window is running on a host with IP 10.10.15.35. The command entered is:

```
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=ClusterIP --port=8080 -o json --dry-run=client
```

The output is a JSON object representing the service configuration:

```
{ "kind": "Service", "apiVersion": "v1", "metadata": { "name": "nginx", "creationTimestamp": null, "labels": { "app": "nginx" } }, "spec": { "ports": [ { "protocol": "TCP", "port": 8080, "targetPort": 8080 } ], "selector": { "app": "nginx" }, "type": "ClusterIP" }, "status": { "loadBalancer": {} } }
```

In the bottom right corner of the terminal window, there is a watermark: 'Activate Windows' and 'Go to Settings to activate Windows'.

```

root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=ClusterIP --port=8080 -o yaml --dry-run=client > nginx-clusterip-svc.yml
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=ClusterIP --port=8080 -o yaml --dry-run=client > nginx-clusterip-svc.yml
[root@OS02 declarative-manifests-scripts]# 

root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=ClusterIP --port=8080 -o yaml --dry-run=client > nginx-clusterip-svc.yml
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=ClusterIP --port=8080 -o yaml --dry-run=client > nginx-clusterip-svc.yml
[root@OS02 declarative-manifests-scripts]# oc apply -f nginx-clusterip-svc.yml
[root@OS02 declarative-manifests-scripts]# oc get svc
[root@OS02 declarative-manifests-scripts]# oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx    ClusterIP  172.30.115.97  <none>        8080/TCP    3s
[root@OS02 declarative-manifests-scripts]# 

```

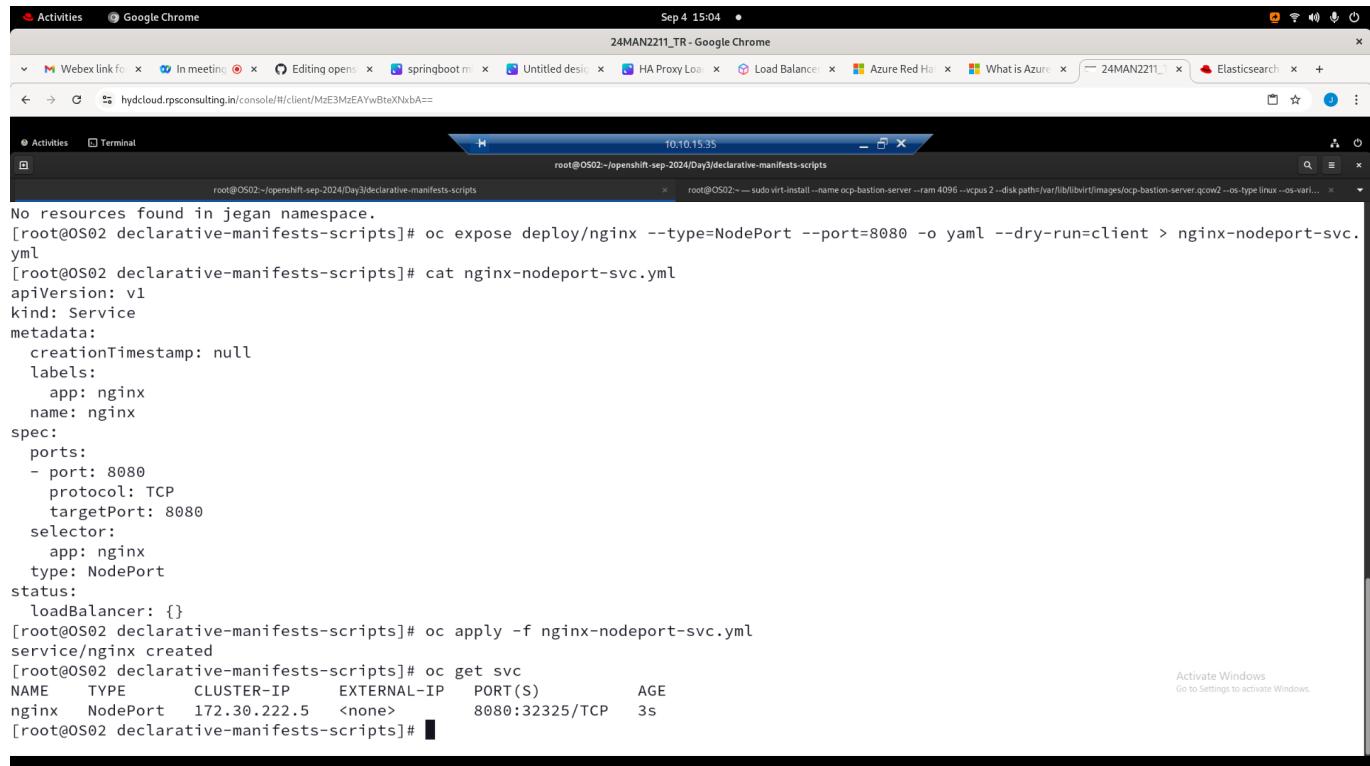
Lab - Creating a NodePort external service in declarative style

```

oc delete -f nginx-clusterip-svc.yml
oc get svc
oc expose deploy/nginx --type=NodePort --port=8080 -o yaml --dry-run=client > nginx-nodeport-svc.yml
cat nginx-nodeport-svc.yml
oc apply -f nginx-nodeport-svc.yml
oc get svc

```

Expected output



The screenshot shows a terminal window with a blue header bar. The title bar says "Activities" and "Terminal". The window title is "root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts". The terminal content shows the following commands and their output:

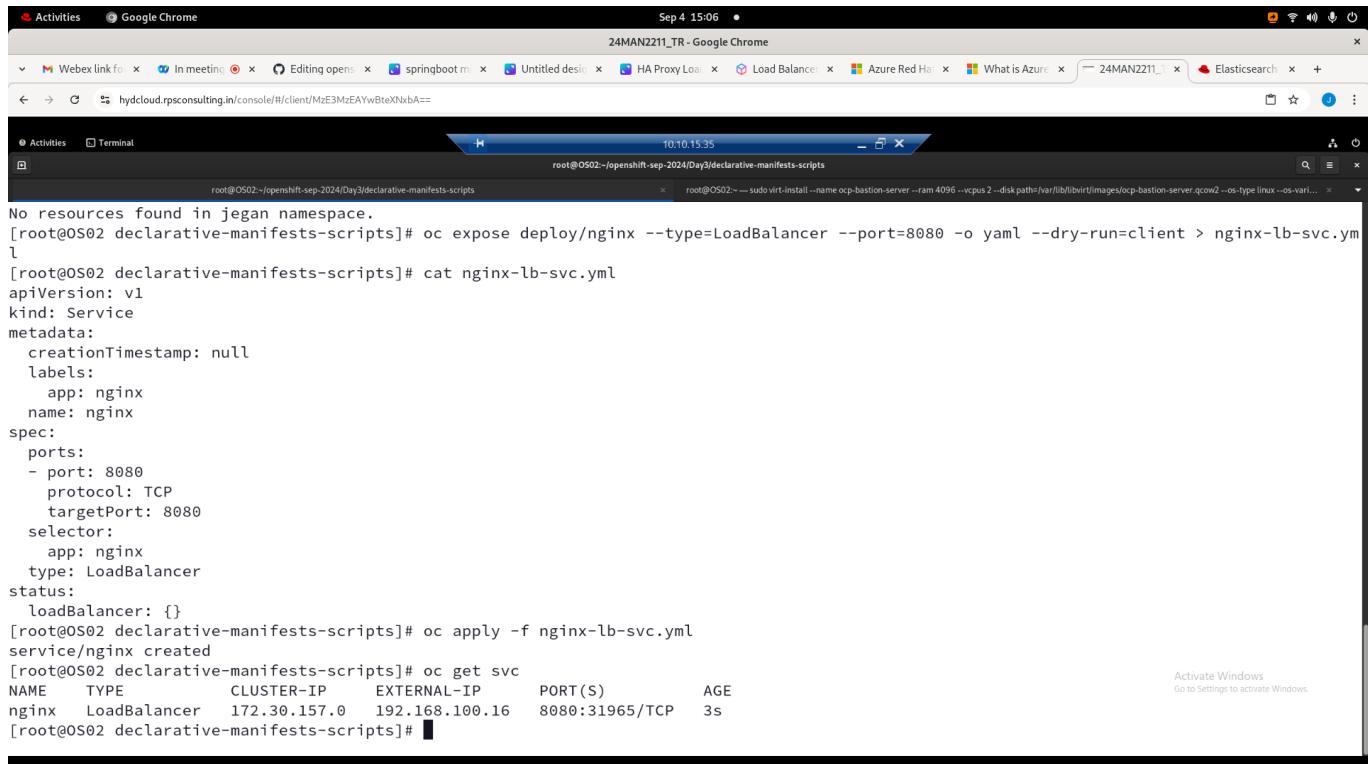
```
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=NodePort --port=8080 -o yaml --dry-run=client > nginx-nodeport-svc.yml
[root@OS02 declarative-manifests-scripts]# cat nginx-nodeport-svc.yml
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: null
  labels:
    app: nginx
  name: nginx
spec:
  ports:
  - port: 8080
    protocol: TCP
    targetPort: 8080
  selector:
    app: nginx
    type: NodePort
  status:
    loadBalancer: {}
[root@OS02 declarative-manifests-scripts]# oc apply -f nginx-nodeport-svc.yml
service/nginx created
[root@OS02 declarative-manifests-scripts]# oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx    NodePort    172.30.222.5    <none>        8080:32325/TCP   3s
[root@OS02 declarative-manifests-scripts]#
```

On the right side of the terminal window, there is a small message: "Activate Windows" and "Go to Settings to activate Windows".

Lab - Creating a LoadBalancer external service in declarative style

```
oc delete -f nginx-nodeport-svc.yml
oc get svc
oc expose deploy/nginx --type=LoadBalancer --port=8080 -o yaml --dry-
run=client -o nginx-lb-svc.yml
cat nginx-lb-svc.yml
oc apply -f nginx-lb-svc.yml
oc get svc
```

Expected output



```

Activities Google Chrome
Sep 4 15:06 •
24MAN2211_TR - Google Chrome
Webex link fo In meeting Editing open... springboot m Untitled des... HA Proxy Load Balancer Azure Red Ha What is Azure 24MAN2211_ Elasticsearch x +
hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNxbA==

Activities Terminal
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts 10:10.15.35
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts x root@OS02:~ sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=redhat-7.8 --cpu host --graphics none --network network=br0,model=virtio --location=/var/lib/libvirt/images/centos-7-1908-x86_64-disk1.qcow2 --cdrom=/var/lib/libvirt/images/centos-7-1908-x86_64-dvd.iso --extra-args=initrd=/vmlinuz-7.9.0-1602.el7.x86_64 root=/dev/sda1 ro rootdelay=5
No resources found in jegan namespace.
[root@OS02 declarative-manifests-scripts]# oc expose deploy/nginx --type=LoadBalancer --port=8080 -o yaml --dry-run=client > nginx-lb-svc.yml
l
[root@OS02 declarative-manifests-scripts]# cat nginx-lb-svc.yml
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  ports:
  - port: 8080
    protocol: TCP
    targetPort: 8080
  selector:
    app: nginx
    type: LoadBalancer
  status:
    loadBalancer: {}
[root@OS02 declarative-manifests-scripts]# oc apply -f nginx-lb-svc.yml
service/nginx created
[root@OS02 declarative-manifests-scripts]# oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx    LoadBalancer    172.30.157.0    192.168.100.16    8080:31965/TCP   3s
[root@OS02 declarative-manifests-scripts]# 
```

Lab - Creating a Pod in declarative style

Create a pod.yml with below content

```

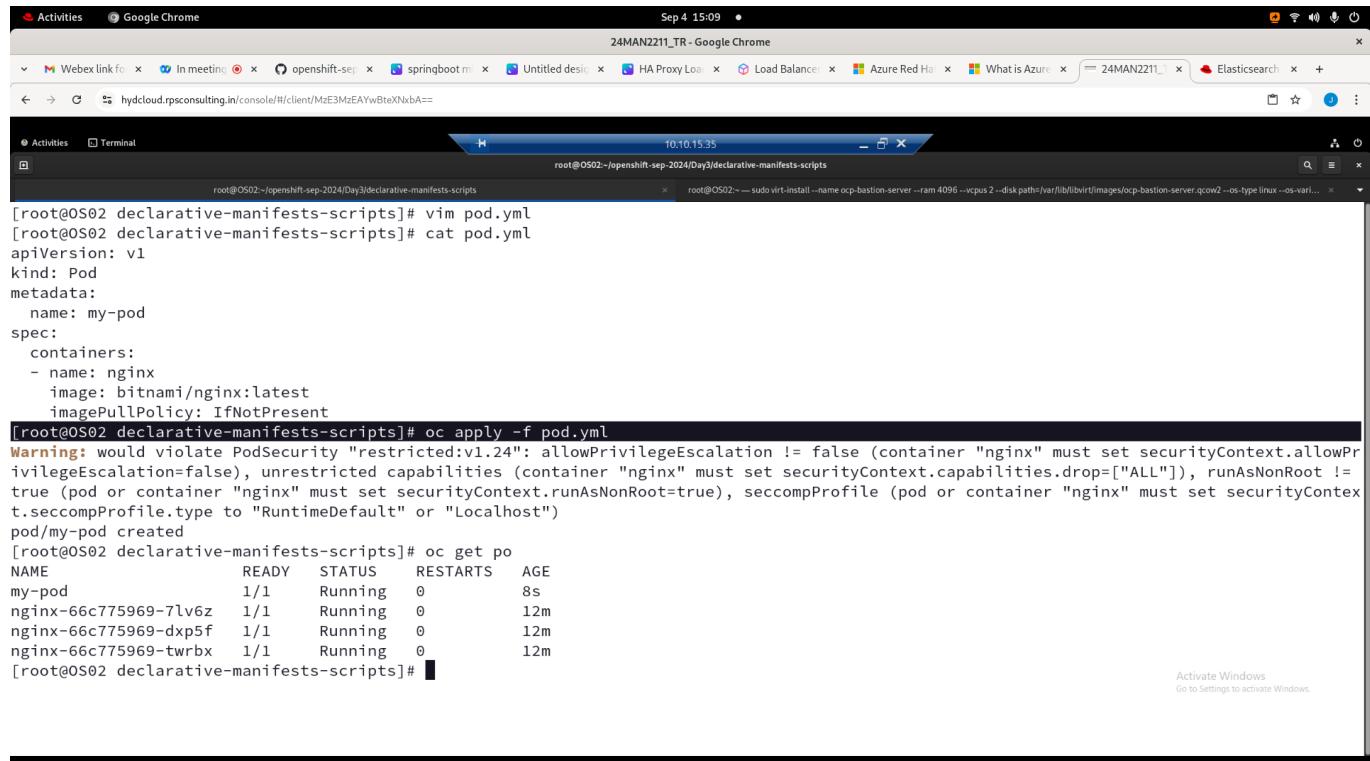
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  containers:
  - name: nginx
    image: bitnami/nginx:latest
    imagePullPolicy: IfNotPresent
  
```

You should be able to create pod in the declarative style

```

oc apply -f pod.yml
oc get po
  
```

Expected output



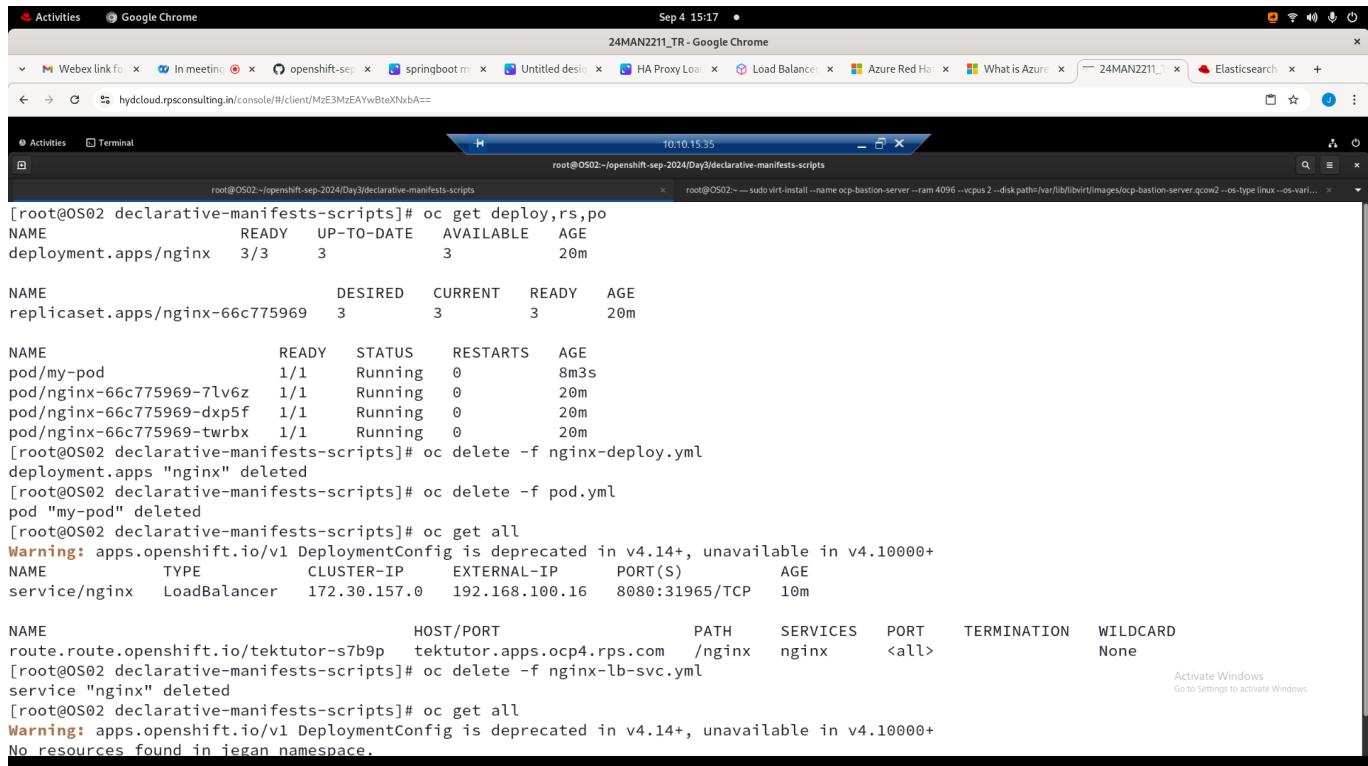
```
root@OS02 declarative-manifests-scripts]# vim pod.yml
[root@OS02 declarative-manifests-scripts]# cat pod.yml
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  containers:
    - name: nginx
      image: bitnami/nginx:latest
      imagePullPolicy: IfNotPresent
[root@OS02 declarative-manifests-scripts]# oc apply -f pod.yml
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "nginx" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "nginx" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "nginx" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "nginx" must set securityContext.seccompProfile.type to "RuntimeDefault" or "localhost")
pod/my-pod created
[root@OS02 declarative-manifests-scripts]# oc get po
NAME           READY   STATUS    RESTARTS   AGE
my-pod          1/1     Running   0          8s
nginx-66c775969-7lv6z 1/1     Running   0          12m
nginx-66c775969-dxp5f 1/1     Running   0          12m
nginx-66c775969-twrbx 1/1     Running   0          12m
[root@OS02 declarative-manifests-scripts]#
```

Lab - Rolling update in declarative style

Let's delete all the resources in our project namespace

```
oc get all
oc delete -f nginx-deploy.yml
oc delete -f nginx-lb-svc.yml
oc delete -f pod.yml
oc get all
```

Expected output



```

Sep 4 15:17 • 24MAN2211_TR - Google Chrome
Activities Google Chrome
Webex link fo In meeting openshift-sei springboot m Untitled desi HA Proxy Load Balancer Azure Red Ha What is Azure 24MAN2211 Elasticsearch
hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNxbA==

Activities Terminal 10:10.15.35
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts x root@OS02:~ sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=centos7.9 --cpu host --graphics none --network network=br0,model=virtio --location /var/lib/libvirt/images/centos-7.9.2009-disk-1.qcow2 --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=centos7.9 --cpu host --graphics none --network network=br0,model=virtio
[root@OS02 declarative-manifests-scripts]# oc get deploy,rs,po
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   3/3      3           3           20m
replicaset.apps/nginx-66c775969   3         3           3           20m
NAME           READY   STATUS    RESTARTS   AGE
pod/my-pod     1/1     Running   0          8m3s
pod/nginx-66c775969-7lv6z  1/1     Running   0          20m
pod/nginx-66c775969-dxp5f   1/1     Running   0          20m
pod/nginx-66c775969-twrbx  1/1     Running   0          20m
[root@OS02 declarative-manifests-scripts]# oc delete -f nginx-deploy.yml
deployment.apps "nginx" deleted
[root@OS02 declarative-manifests-scripts]# oc delete -f pod.yml
pod "my-pod" deleted
[root@OS02 declarative-manifests-scripts]# oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
NAME           TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE
service/nginx  LoadBalancer  172.30.157.0  192.168.100.16  8080:31965/TCP  10m
NAME           HOST/PORT      PATH        SERVICES    PORT    TERMINATION   WILDCARD
route.route.openshift.io/tektutor-s7b9p  tektutor.apps.ocp4.rps.com  /nginx  nginx    <all>  None
[root@OS02 declarative-manifests-scripts]# oc delete -f nginx-lb-svc.yml
service "nginx" deleted
[root@OS02 declarative-manifests-scripts]# oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
No resources found in jegan namespace.

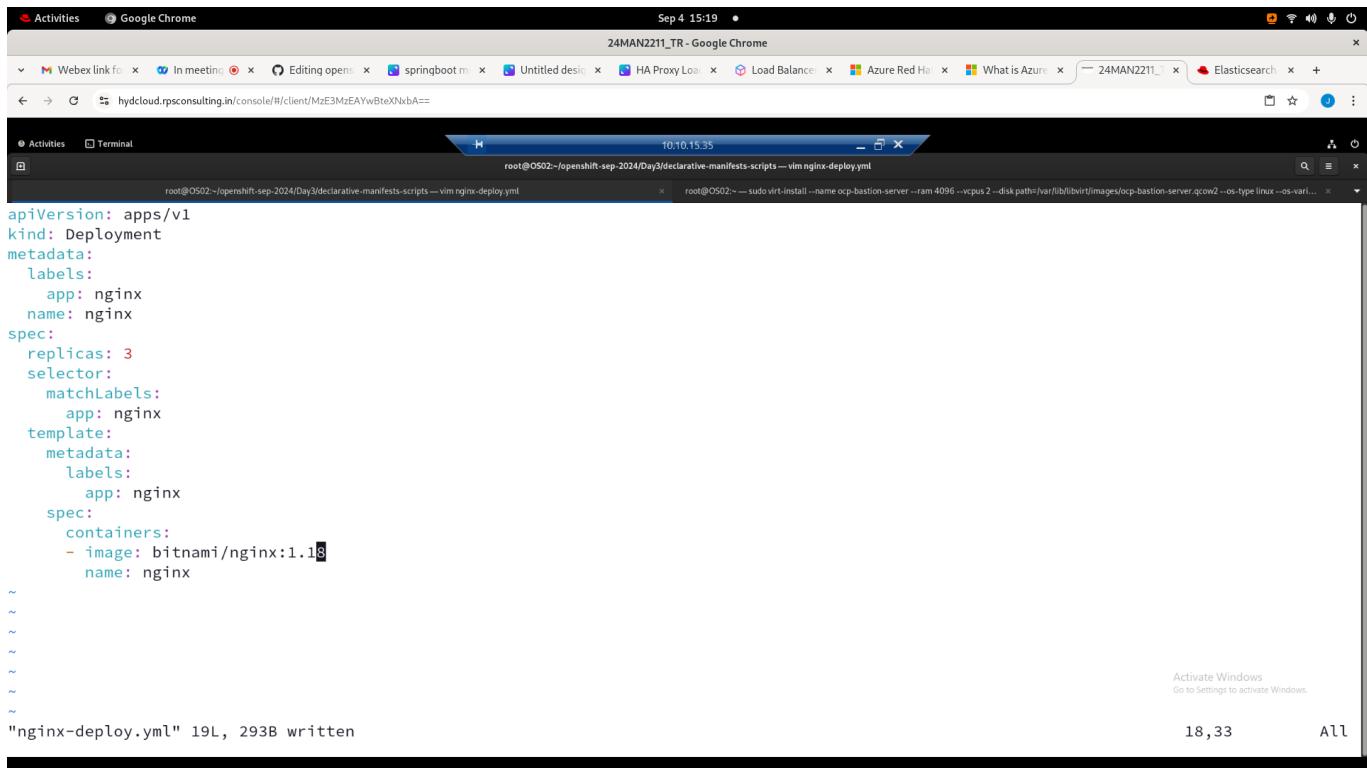
```

Let's deploy Nginx in declarative style using bitnami/nginx:1.18 docker image

```

oc create -f nginx-deploy.yml --save-config
oc get deploy,po

```

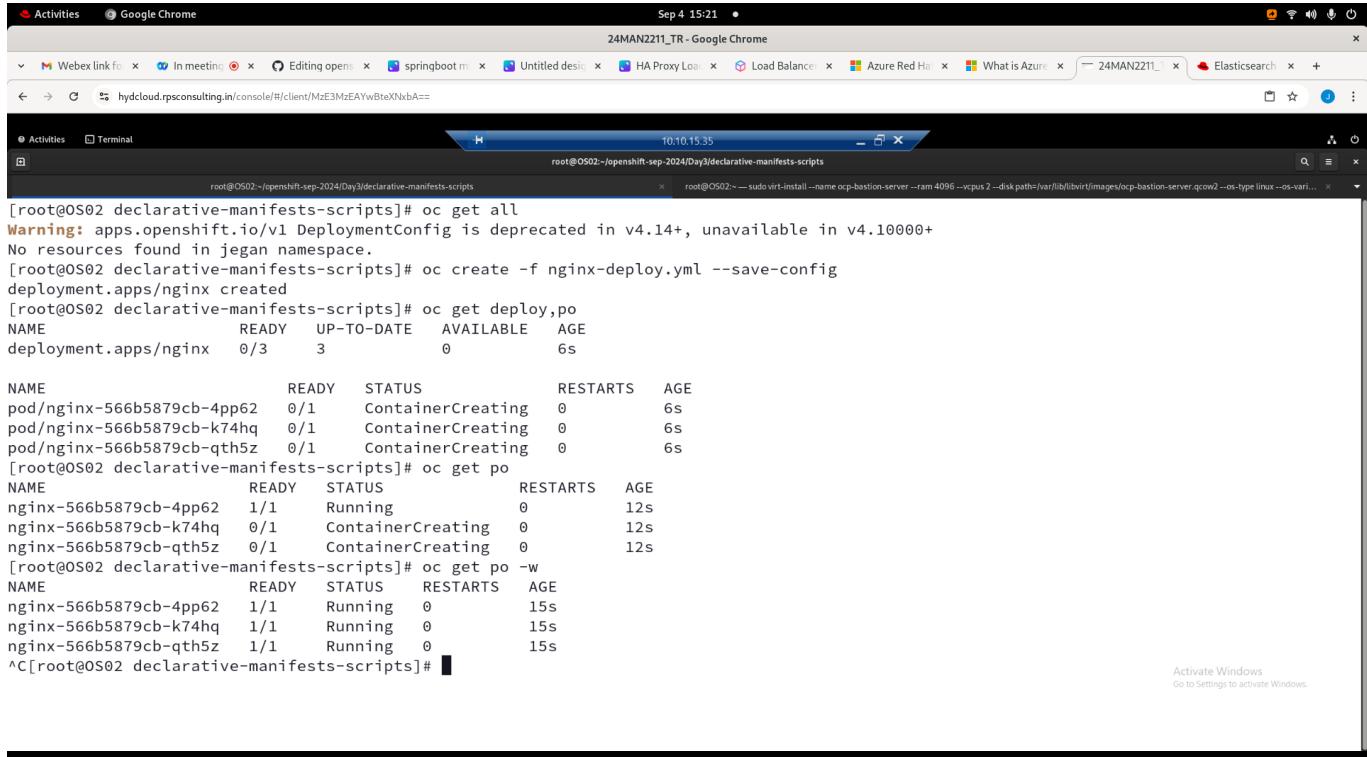


```

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:1.18
          name: nginx

```

"nginx-deploy.yml" 19L, 293B written



```

[root@OS02 declarative-manifests-scripts]# oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
No resources found in jegan namespace.

[root@OS02 declarative-manifests-scripts]# oc create -f nginx-deploy.yml --save-config
deployment.apps/nginx created

[root@OS02 declarative-manifests-scripts]# oc get deploy,po
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   0/3       3           0           6s

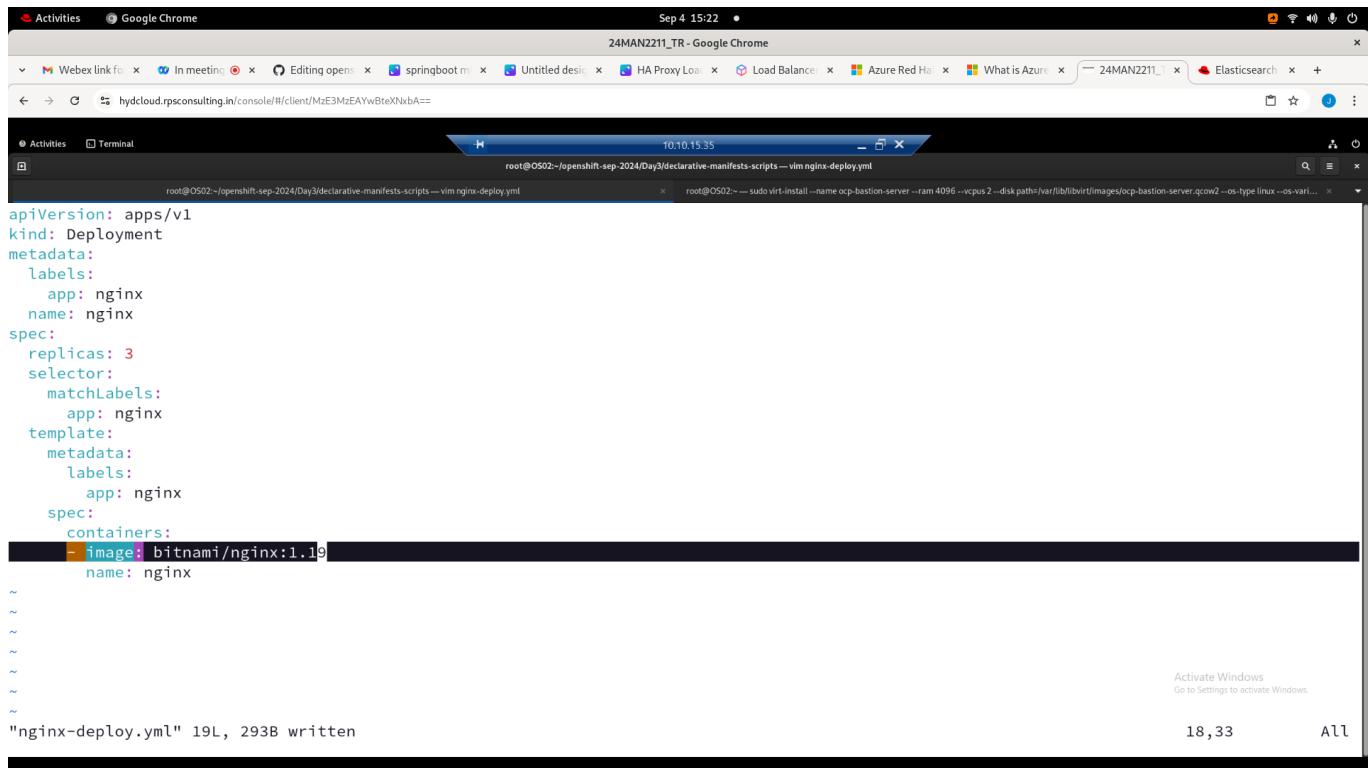
NAME          READY   STATUS        RESTARTS   AGE
pod/nginx-566b5879cb-4pp62  0/1   ContainerCreating  0          6s
pod/nginx-566b5879cb-k74hq  0/1   ContainerCreating  0          6s
pod/nginx-566b5879cb-qth5z  0/1   ContainerCreating  0          6s

[root@OS02 declarative-manifests-scripts]# oc get po -w
NAME          READY   STATUS        RESTARTS   AGE
nginx-566b5879cb-4pp62  1/1   Running      0          12s
nginx-566b5879cb-k74hq  0/1   ContainerCreating  0          12s
nginx-566b5879cb-qth5z  0/1   ContainerCreating  0          12s

[root@OS02 declarative-manifests-scripts]# oc get po -w
NAME          READY   STATUS        RESTARTS   AGE
nginx-566b5879cb-4pp62  1/1   Running      0          15s
nginx-566b5879cb-k74hq  1/1   Running      0          15s
nginx-566b5879cb-qth5z  1/1   Running      0          15s
^C[root@OS02 declarative-manifests-scripts]#

```

Now, let's edit the `nginx-deploy.yml` and update the image from `bitnami/nginx:1.18` to `bitnami/nginx:1.19` and save it



```

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:1.19
          name: nginx

```

"nginx-deploy.yml" 19L, 293B written

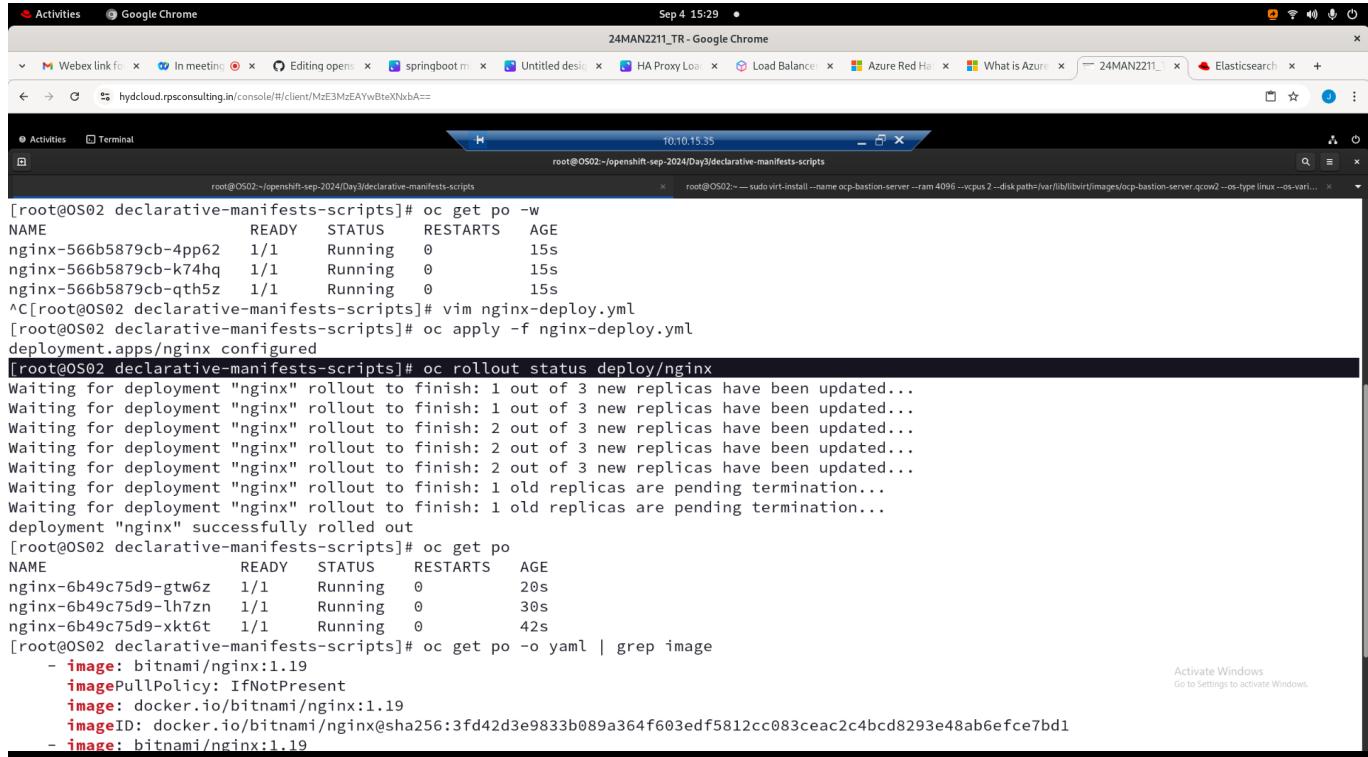
Now let's apply the nginx-deploy.yml with updated bitnami/nginx:1.19

```

oc apply -f nginx-deploy
oc get po -o yaml | grep image
oc rollout status deploy/nginx
oc get po -o yaml | grep image

```

Expected output



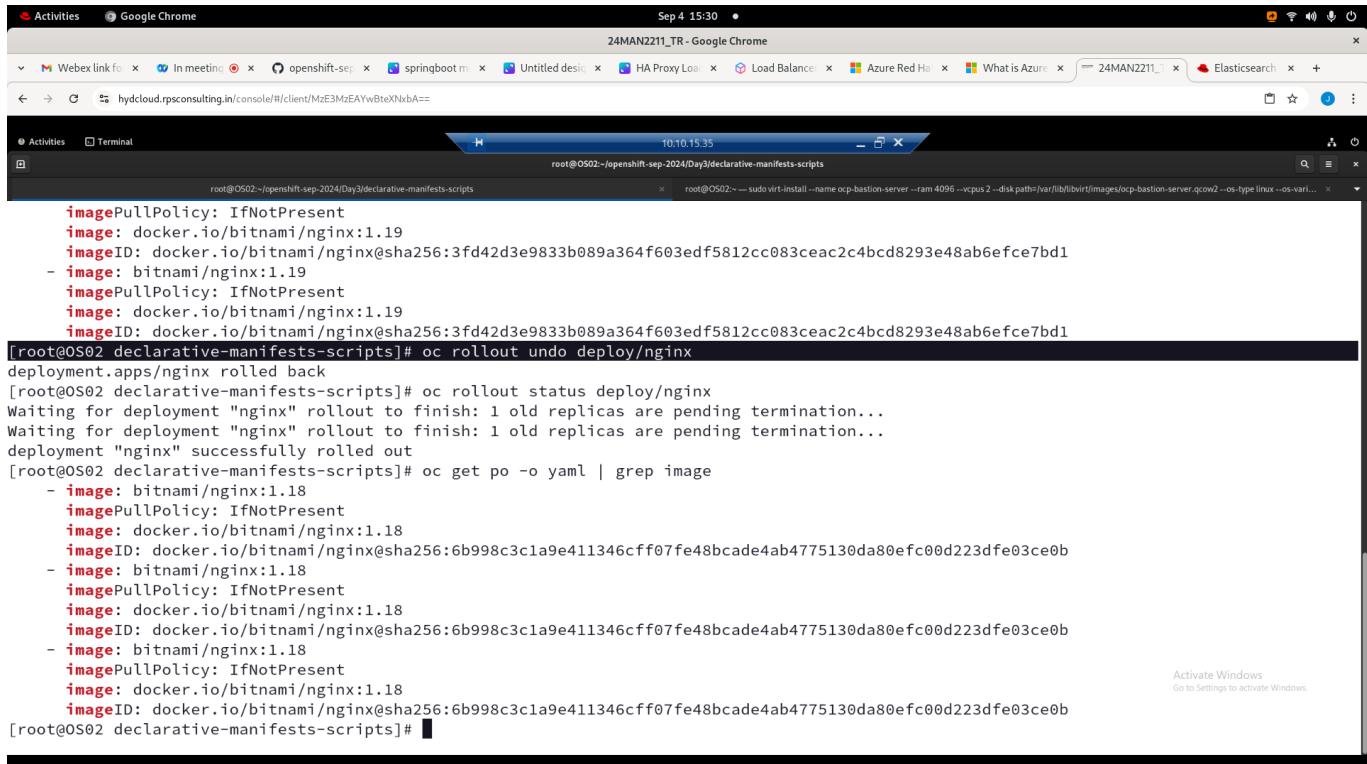
```

[root@OS02 declarative-manifests-scripts]# oc get po -w
NAME        READY   STATUS    RESTARTS   AGE
nginx-566b5879cb-4pp62  1/1     Running   0          15s
nginx-566b5879cb-k74hq  1/1     Running   0          15s
nginx-566b5879cb-qth5z  1/1     Running   0          15s
^C[root@OS02 declarative-manifests-scripts]# vim nginx-deploy.yml
[root@OS02 declarative-manifests-scripts]# oc apply -f nginx-deploy.yml
deployment.apps/nginx configured
[root@OS02 declarative-manifests-scripts]# oc rollout status deploy/nginx
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
deployment "nginx" successfully rolled out
[root@OS02 declarative-manifests-scripts]# oc get po
NAME        READY   STATUS    RESTARTS   AGE
nginx-6b49c75d9-gtw6z  1/1     Running   0          20s
nginx-6b49c75d9-lh7zn  1/1     Running   0          30s
nginx-6b49c75d9-xkt6t  1/1     Running   0          42s
[root@OS02 declarative-manifests-scripts]# oc get po -o yaml | grep image
  - image: bitnami/nginx:1.19
    imagePullPolicy: IfNotPresent
    image: docker.io/bitnami/nginx:1.19
    imageID: docker.io/bitnami/nginx@sha256:3fd42d3e9833b089a364f603edf5812cc083ceac2c4bcd8293e48ab6efce7bd1
  - image: bitnami/nginx:1.19

```

Rollback will revert the pods to use older image i.e from bitnami/nginx:1.19 to bitnami/nginx:1.18

```
oc rollout undo deploy/nginx
oc rollout status deploy/nginx
oc get po -o yaml | grep image
```



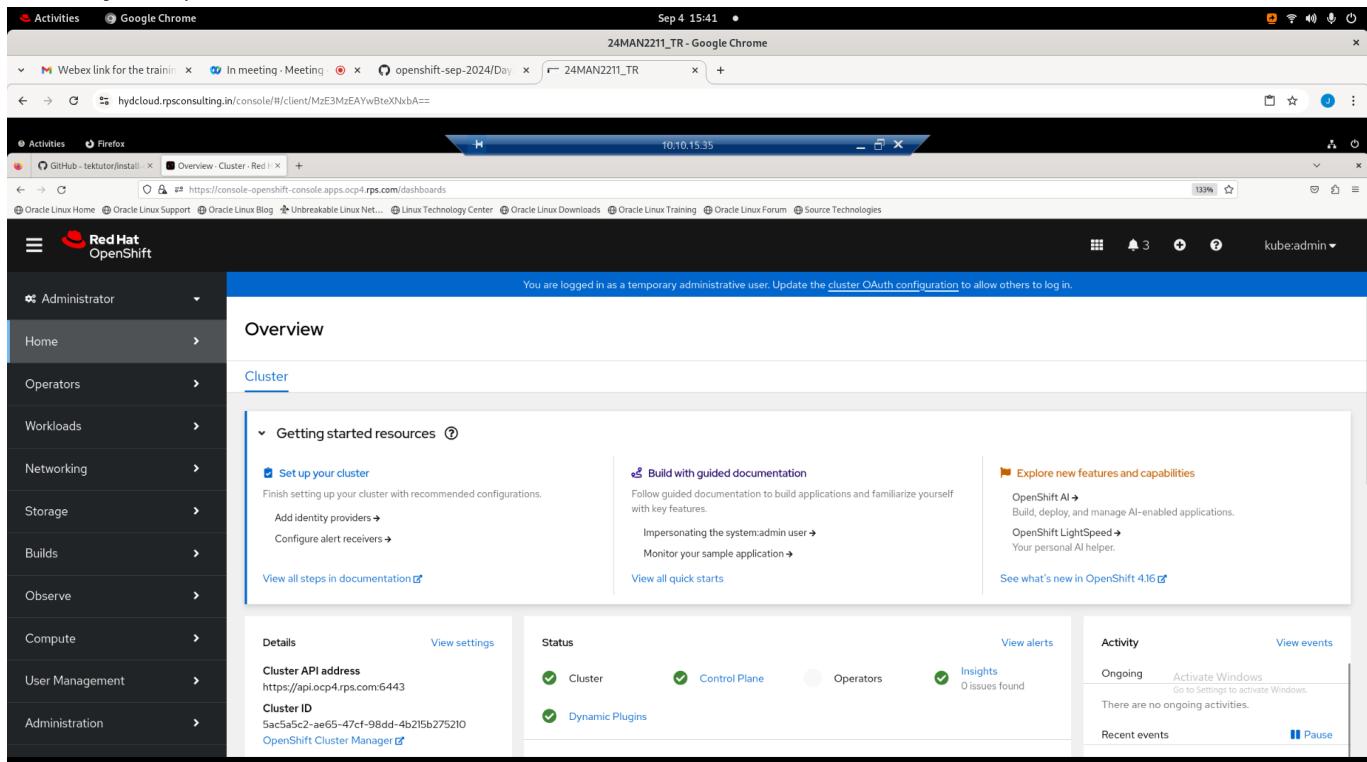
```
Activities Google Chrome Sep 4 15:30 • 24MAN2211_TR - Google Chrome
Webex link fo In meeting openshift-sep... springboot m Untitled desig HA Proxy Log Load Balance Azure Red Ha What is Azure 24MAN2211_ Elasticsearch x

hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNbA==

Activities Terminal 10:10.15.35
root@OS02:~/openshift-sep-2024/Day3/declarative-manifests-scripts
root@OS02:~ sudo virt-install --name ocp-bastion-server --ram 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/ocp-bastion-server.qcow2 --os-type linux --os-variant=centos7.9
root@OS02:~ oc rollout undo deploy/nginx
deployment.apps/nginx rolled back
[root@OS02 declarative-manifests-scripts]# oc rollout status deploy/nginx
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
deployment "nginx" successfully rolled out
[root@OS02 declarative-manifests-scripts]# oc get po -o yaml | grep image
- image: bitnami/nginx:1.18
  imagePullPolicy: IfNotPresent
  image: docker.io/bitnami/nginx:1.18
  imageID: docker.io/bitnami/nginx@sha256:3fd42d3e9833b089a364f603edf5812cc083ceac2c4bcd8293e48ab6efce7bd1
- image: bitnami/nginx:1.19
  imagePullPolicy: IfNotPresent
  image: docker.io/bitnami/nginx:1.19
  imageID: docker.io/bitnami/nginx@sha256:3fd42d3e9833b089a364f603edf5812cc083ceac2c4bcd8293e48ab6efce7bd1
[root@OS02 declarative-manifests-scripts]# oc rollout undo deploy/nginx
deployment.apps/nginx rolled back
[root@OS02 declarative-manifests-scripts]# oc rollout status deploy/nginx
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
deployment "nginx" successfully rolled out
[root@OS02 declarative-manifests-scripts]# oc get po -o yaml | grep image
- image: bitnami/nginx:1.18
  imagePullPolicy: IfNotPresent
  image: docker.io/bitnami/nginx:1.18
  imageID: docker.io/bitnami/nginx@sha256:6b998c3c1a9e411346cff07fe48bcade4ab4775130da80efc00d223dfe03ce0b
- image: bitnami/nginx:1.18
  imagePullPolicy: IfNotPresent
  image: docker.io/bitnami/nginx:1.18
  imageID: docker.io/bitnami/nginx@sha256:6b998c3c1a9e411346cff07fe48bcade4ab4775130da80efc00d223dfe03ce0b
- image: bitnami/nginx:1.18
  imagePullPolicy: IfNotPresent
  image: docker.io/bitnami/nginx:1.18
  imageID: docker.io/bitnami/nginx@sha256:6b998c3c1a9e411346cff07fe48bcade4ab4775130da80efc00d223dfe03ce0b
[root@OS02 declarative-manifests-scripts]#
```

Lab - Deploying application from Red Hat Openshift webconsole

Launch your openshift webconsole



Switch to Developer context(view)

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

Project: jegan Application: All applications

No resources found

Start building your application or visit the [Add page](#) for more details.

Activate Windows
Go to Settings to activate Windows.

Click on +Add

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

Project: jegan

Add

Getting started resources

Create applications using samples

Choose a code sample to get started creating an application with.

Basic Quarkus →
Basic Spring Boot →

View all samples

Build with guided documentation

Follow guided documentation to build applications and familiarize yourself with key features.

Get started with Quarkus using s2i →
Get started with Spring →

View all quick starts

Explore new developer features

Explore new features and resources within the developer perspective.

Discover certified Helm Charts →
Start building your application quickly in topology →

What's new in OpenShift 4.16 →

Developer Catalog

All services

Browse the catalog to discover, deploy and connect to services

Git Repository

Import from Git

Deploy code from your Git repository to be built and deployed

Container images

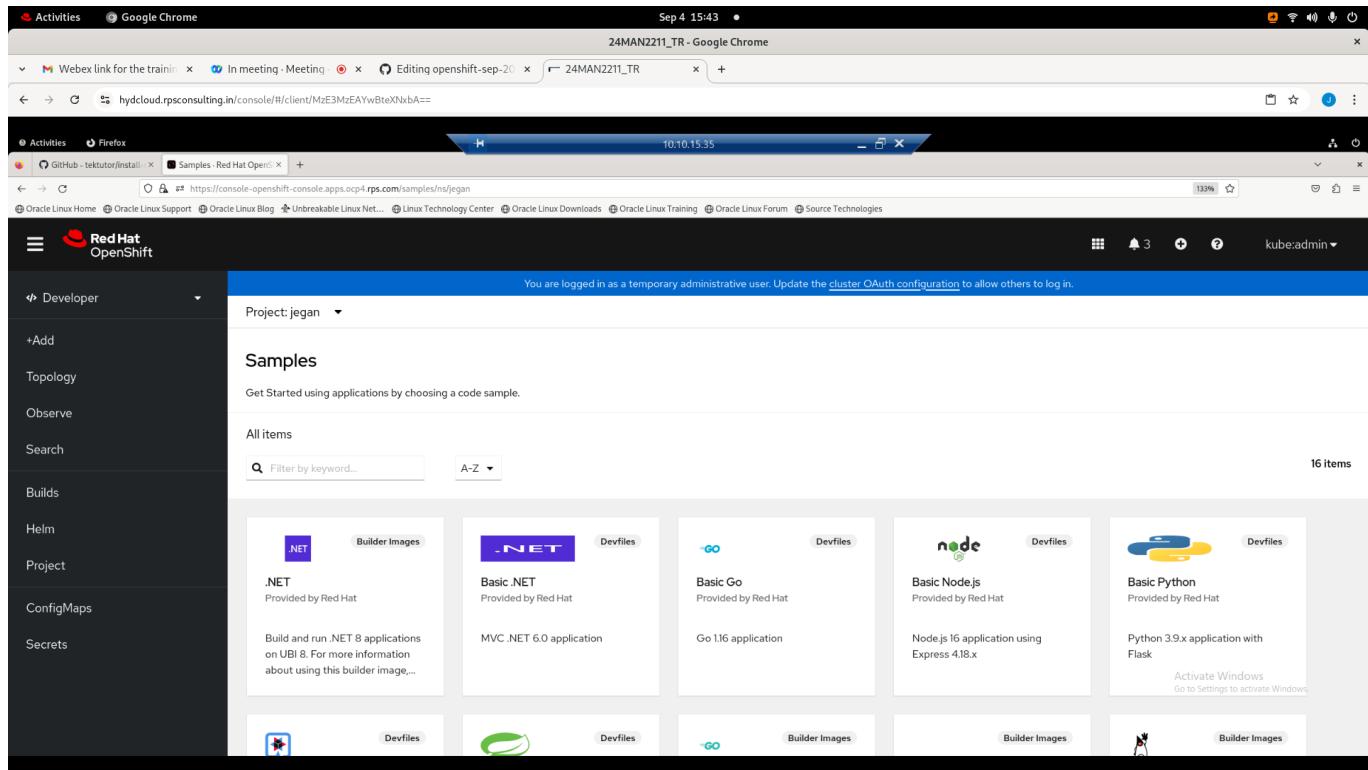
Deploy an existing Image from an Image registry or Image stream tag

Sharing

Activate Windows

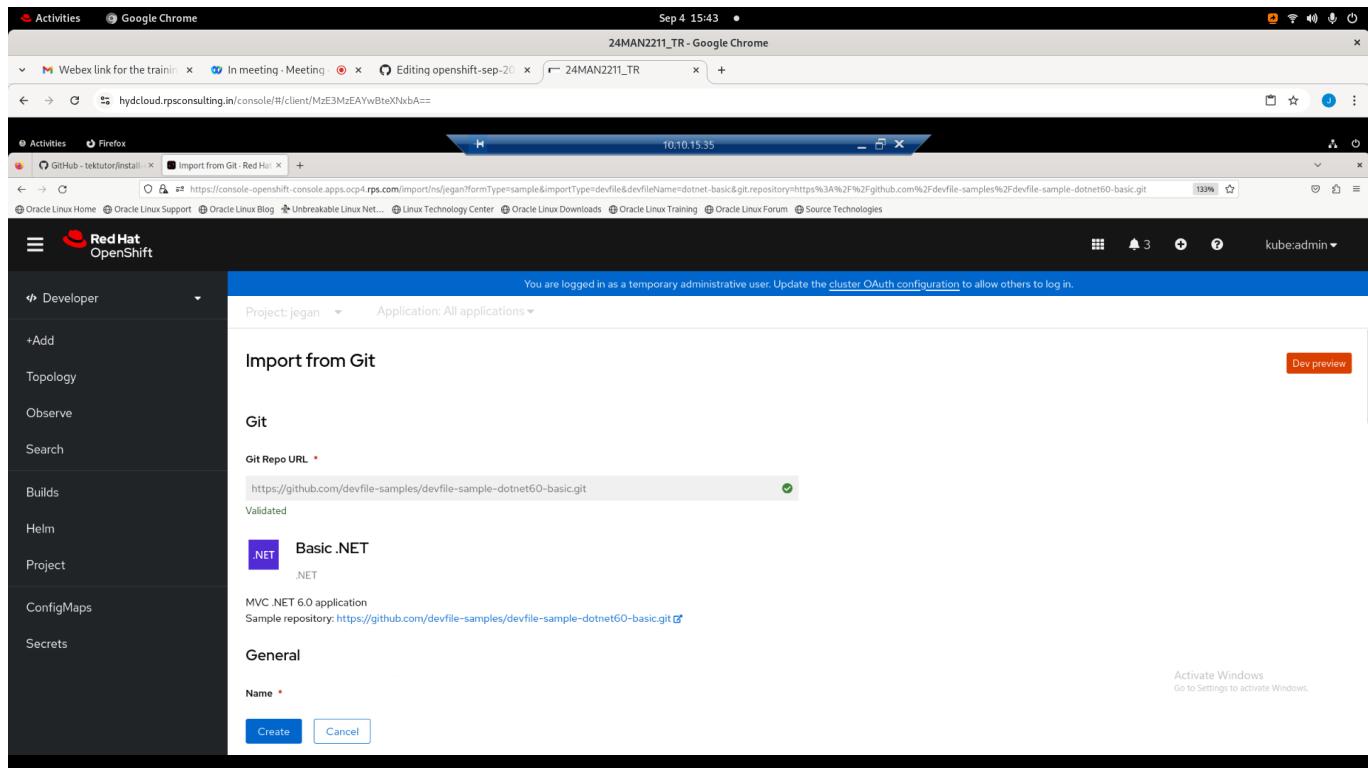
Project access allows you to add or remove a user's access to the project

Under "Get started resources", click "View all samples"



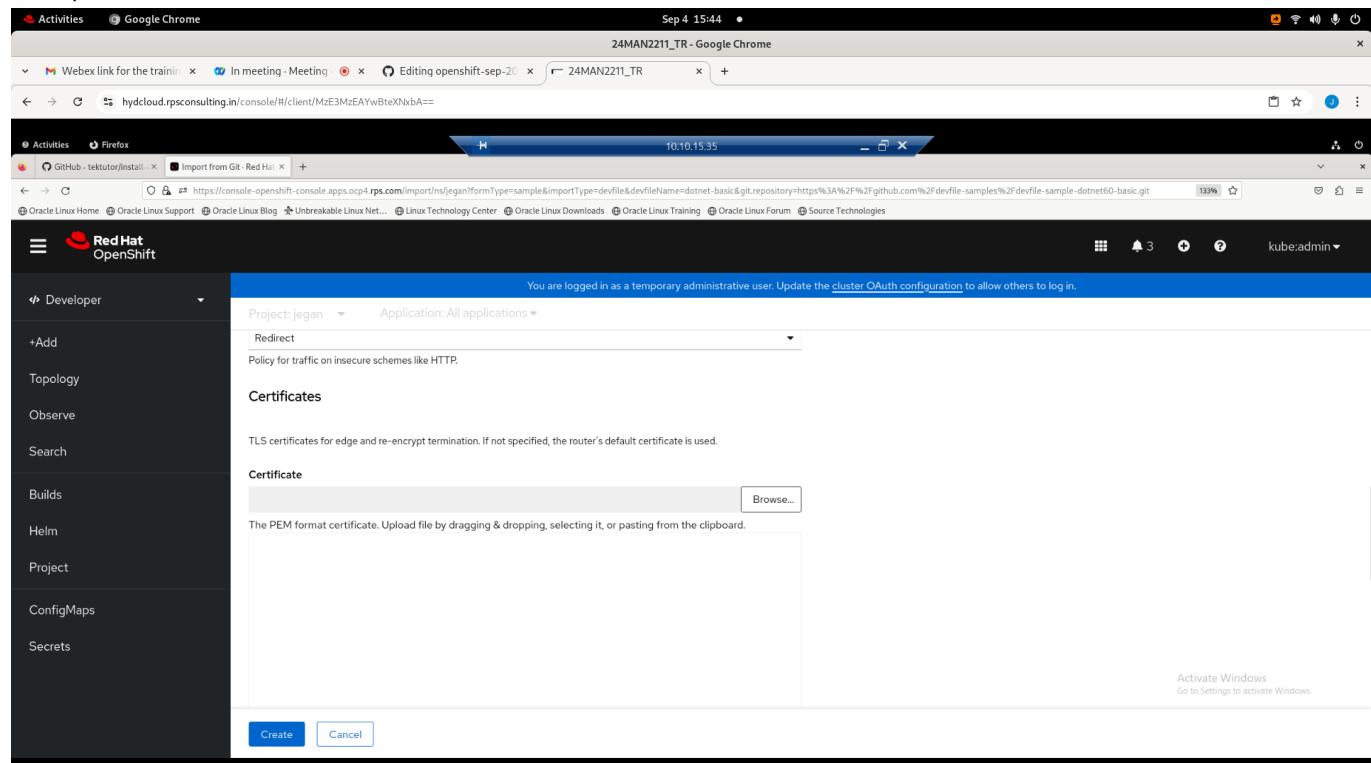
The screenshot shows the Red Hat OpenShift Samples page. The left sidebar includes options like Developer, Topology, Observe, Search, Builds, Helm, Project, ConfigMaps, and Secrets. The main content area is titled "Samples" and displays a list of 16 items. Each item has a thumbnail, a name, and a "Builder Images" or "Devfiles" label. The items listed are: .NET (Builder Images), Basic .NET (Builder Images), Basic Go (Devfiles), Basic Node.js (Devfiles), Basic Python (Devfiles), Java (Builder Images), Java (Devfiles), and Node.js (Builder Images).

Click on "Basic .Net"



The screenshot shows the Red Hat OpenShift Import from Git page for the "Basic .NET" application. The left sidebar is identical to the previous screenshot. The main content area is titled "Import from Git" and shows the "Git" section. It includes a "Git Repo URL" field with the value "https://github.com/devfile-samples/devfile-sample-dotnet60-basic.git" and a "Validated" status. Below this, there is a "Basic .NET" section with a ".NET" label and a description of "MVC .NET 6.0 application". At the bottom, there are "Create" and "Cancel" buttons.

Accept all the default and click "Create" button



You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

Project: jegan Application: All applications

Redirect

Policy for traffic on insecure schemes like HTTP.

Certificates

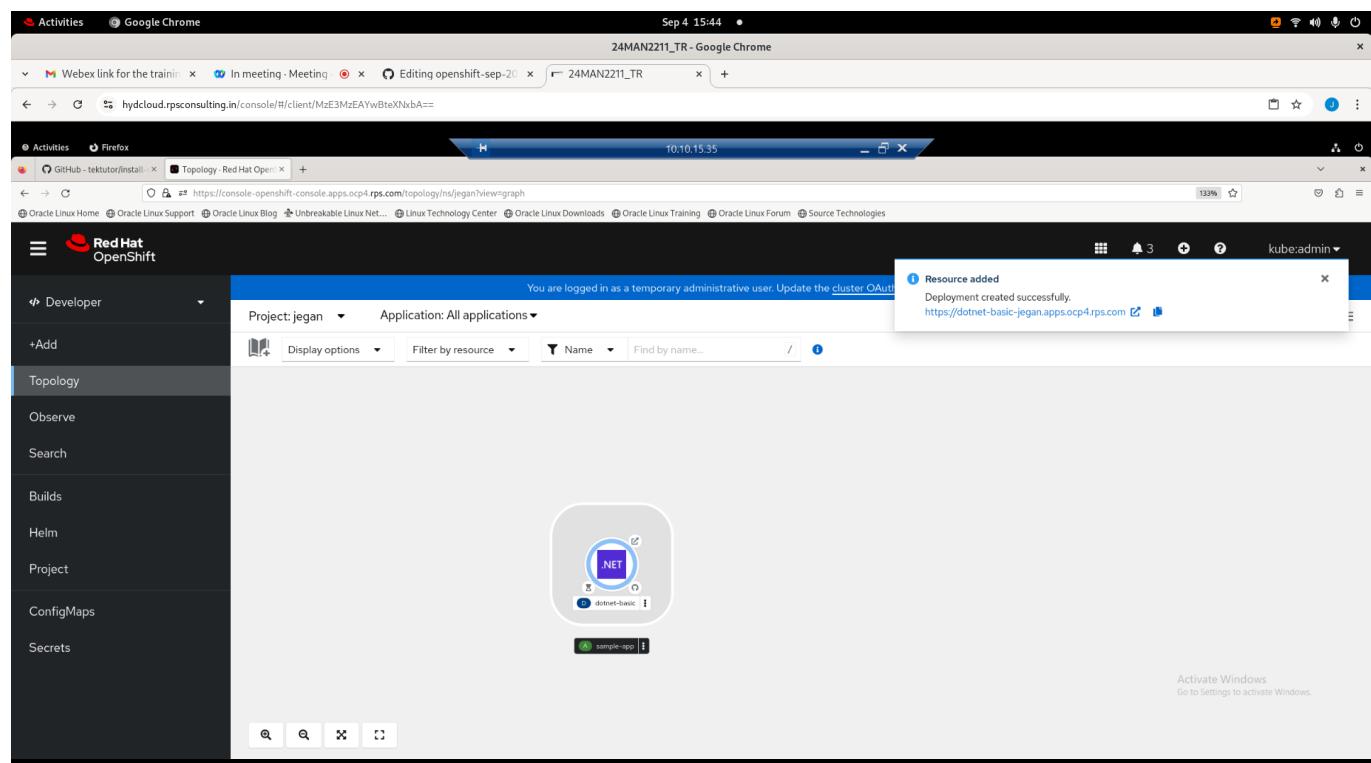
TLS certificates for edge and re-encrypt termination. If not specified, the router's default certificate is used.

Certificate

The PEM format certificate. Upload file by dragging & dropping, selecting it, or pasting from the clipboard.

Browse...

Create Cancel



Activities Google Chrome Sep 4 15:44 24MAN2211_TR - Google Chrome

Web link for the trainin In meeting - Meeting Editing openshift-sep-20 24MAN2211_TR

hydcloud.rpsconsulting.in/console/#/client/MzE3MzEAYwBteXNxbA==

Activities Firefox 10:10.15.35

GitHub - tektutor/install Import from Git - Red Hat OpenShift https://console-openshift-console.apps.ocp4.rps.com/import/ns/jegan/form?type=sample&importType=devFile&devFileName=dotnet-basic&gitRepository=https%3A%2F%2Fgithub.com%2Fdevfile-samples%2Fdevfile-sample-dotnet60-basic.git

Oracle Linux Home Oracle Linux Support Oracle Linux Blog Unbreakable Linux Net... Linux Technology Center Oracle Linux Downloads Oracle Linux Training Oracle Linux Forum Source Technologies

Red Hat OpenShift

Developer

+Add

Topology

Observe

Search

Builds

Helm

Project

ConfigMaps

Secrets

Project: jegan Application: All applications

Topology - Red Hat OpenShift

Display options Filter by resource Name Find by name...

Resource added Deployment created successfully. <https://dotnet-basic-jegan.apps.ocp4.rps.com>

.NET

dotnet-basic

sample-app

Create Cancel

Activate Windows Go to Settings to activate Windows.

Activities Google Chrome Sep 4 15:45 24MAN2211_TR - Google Chrome

Webex link for the trainin... In meeting - Meeting Editing openshift-sep-20... 24MAN2211_TR

Activities Firefox 10.10.15.35

GitHub - tektutor/install dotnet-basic-1 - Build - Logs https://console-openshift-console.apps.ocp4.rps.com/k8s/ns/jegan/builds/dotnet-basic-1/logs Oracle Linux Home Oracle Linux Support Oracle Linux Blog Unbreakable Linux Net... Linux Technology Center Oracle Linux Downloads Oracle Linux Training Oracle Linux Forum Source Technologies

Red Hat OpenShift

Developer Project: jegan

Builds > Build details dotnet-basic-1 New

Logs Events

0 lines 1

Log streaming... Search Show full log Wrap lines Raw Download Expand

Activate Windows Go to Settings to activate Windows.

Activities Google Chrome Sep 4 16:36 24MAN2211_TR - Google Chrome

Webex link for the trainin... In meeting - Meeting openshift-sep-2024/Day 24MAN2211_TR Failed to pull image during

Activities Firefox 10.10.15.35

GitHub - tektutor/install Topology - Red Hat OpenShift Home Page - app Failed to pull image during GitHub - devfile-sample

GitHub - tektutor/install Topology - Red Hat OpenShift Home Page - app Failed to pull image during GitHub - devfile-sample

Red Hat OpenShift

Project: jegan Application: All applications View shortcuts

Display options Filter by resource Name Find by name...

sample-app

Activate Windows Go to Settings to activate Windows.

Troubleshooting - Internal Openshift Image Registry not registered issue

<https://www.ibm.com/support/pages/failed-pull-image-during-manage-activation-due-integrated-container-image-registry-not-configured>

Info - What is S2I in Openshift?

- S2I - stands for Souce to Image
- In order to deploy our applications into Kubernetes we need to prepare/use Container Image
- In case of Kubernetes, the Container Image is created outside Kubernetes manually using our favourite tools
- In Openshift, we can deploy our application from source code from GitHub, BitBucket, Gitlab etc.,
- Openshift S2I
 - clones your source code from your version control
 - builds your application binary
 - prepares a custom Container Image using Dockerfile you have in GitHub
 - or prepares a custom Container Image based on your preference (this doesn't require you supplying Dockerfile)
 - pushes the image to Openshift's Internal Registry
 - deploys the application into Openshift from your custom Image cached in Openshift's Internal Registry
- Some of the popular S2I stategies are
 - source
 - you just provide your source code and mention the docker image that should be used to build and deploy your application
 - Dockerfile is not required
 - docker
 - your source code should also have Dockerfile apart from your application source code

Lab - Deploying a Custom Java springboot application using docker strategy from CLI

```
oc project
oc new-app https://github.com/tektutor/spring-ms.git --strategy=docker
oc expose svc/spring-ms
oc get bc
oc logs bc/spring-ms
```

Lab - Deploying a custom java spring boot application using source strategy from CLI

```
oc new-app redhat.access.registry.com/ubi8/openjdk-11~https://github.com/tektutor/spring-ms.git --strategy=source
oc get deploy
oc get svc
oc expose svc/spring-ms
oc logs -f bc/spring-ms
```