

# Day5

## Lab connection details

From the RPS cloud windows machine you need to open Remote Desktop connection and type

Server 1 ( 10.0.1.13 )

- user01 to user08
- user17 to user20

Server 3 ( 10.0.1.25 )

- user09 to user16
- user21 to user24

## Red Hat Openshift Overview

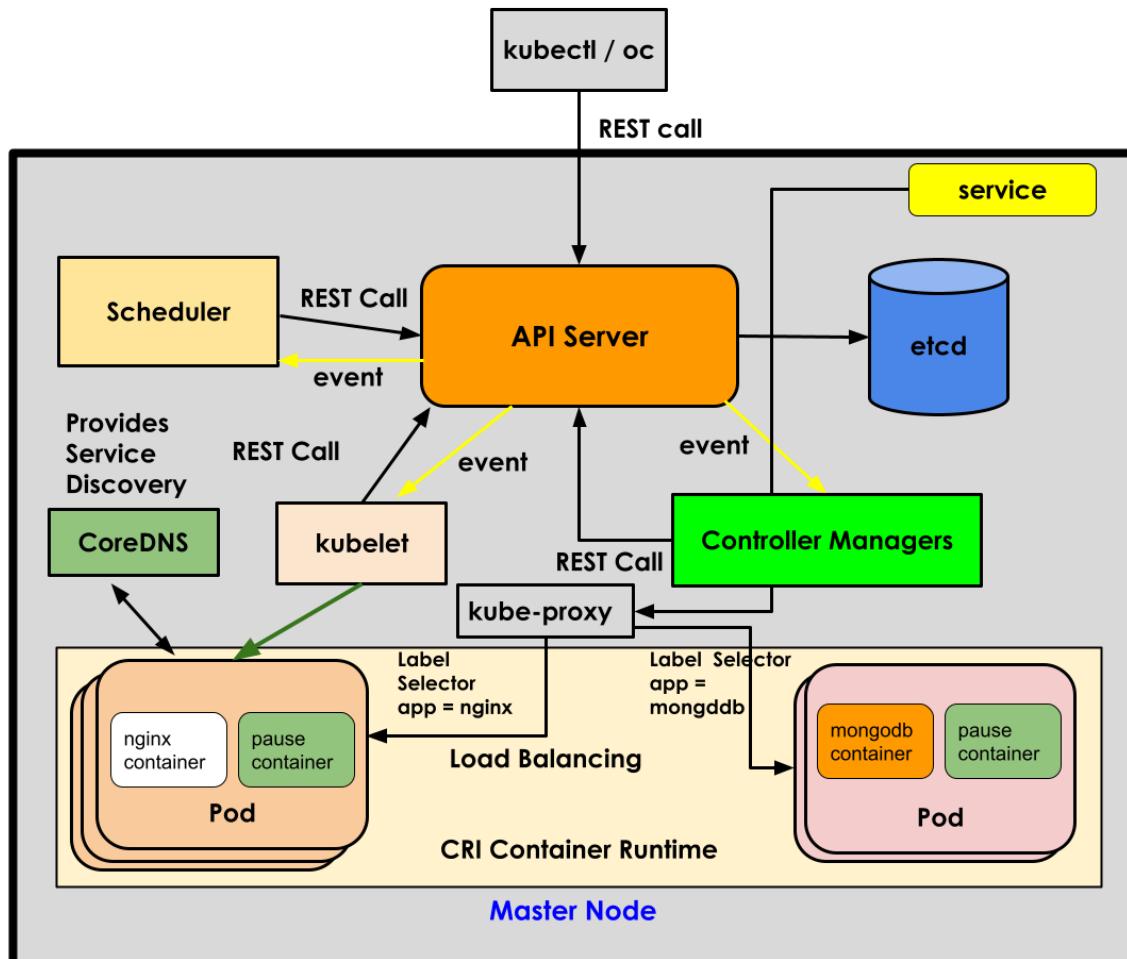
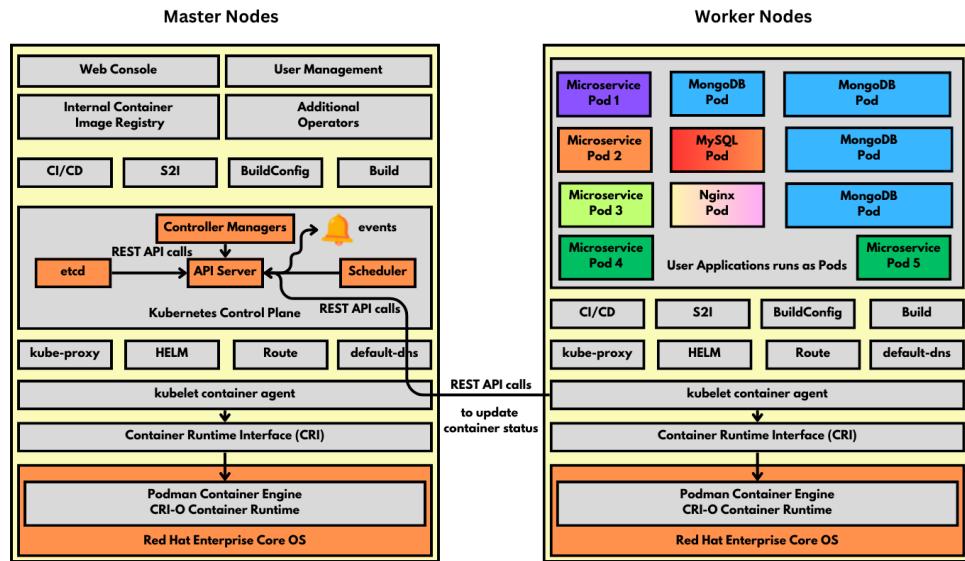
- is Red Hat's Kubernetes distribution
- developed on top of opensource Google Kubernetes with many additional features
- it is a superset of Kubernetes, hence all features of Kubernetes are also supported in Openshift
- Red Hat Openshift supports only RHCOS in master nodes and either RHCOS/RHEL in worker nodes
- supports only CRI-O Container runtime and Podman Container engine
- enterprise product that requires commercial license
- supports many additional features
  - Web console
  - Internal Openshift Image Registry
  - Source to Image (S2I)
    - deploying application from source code
    - deploying application using Dockerfile
  - supports CI/CD
  - supports routes to expose application for external access
  - supports user management
- AWS supports managed Red Hat Openshift cluster called ROSA
  - Load Balancer creates an external Load Balancer supported by AWS
- Azure supports managed Red Hat Openshift cluster called ARO
  - Load Balancer creates an external Load Balancer supported by Azure
- As Red Hat Openshift makes use of Red Hat Enterprise Core OS, it is secure already
  - Ports below 1024 are not allowed as it is reserved for internal use
  - not all applications can be deployed with root access
  - it can be made more secure by using network policy like we do in Kubernetes

- it will enforce best practices are followed which are not taken so seriously

## Info - Openshift onPrem vs Cloud

- onPrem Openshift
  - installation of openshift we need to take care
  - Red Hat Openshift license are taken care by us
  - backup is our responsibility
  - we need to decide the master/worker node hardware configuration as per our application workload and user traffic
  - add new nodes into cluster is done manually using Openstack, VMWare vSphere, etc.,
    - Metallb operator or similar operators must be used to support LoadBalancer service in a on-prem openshift setup like our lab setup
- AWS ROSA
  - installation of openshift is taken care by AWS
  - Red Hat Openshift license is taken care by AWS
  - Hardware configuration of master nodes are decided and managed by AWS
  - backup of etcd database is taken care by AWS
  - LoadBalancer service when created it will automatically create AWS ALB/ELB as it is tightly integrated with AWS

## Red Hat Openshift High-Level Architecture



## Kube config file

- the oc/kubectl client tools require a config file that has connection details to the API Server(load balancer)

- the config file is generally kept in user home directory, .kube folder and the default name of kubeconfig is config
- optionally we could also use the --kubeconfig flag with the oc command to point to a config file
- it is also possible to use a KUBECONFIG environment variable to point to the config file
- Just to give an idea, it is possible that your Kubernetes/OpenShift is running in AWS/Azure but you could install oc/kubectl client tool on your laptop with a config file and still run all the oc/kubectl commands from your laptop without going to aws/azure

To print the content of kubeconfig file

```
cat ~/.kube/config
```

## About Red Hat Enterprise Core OS ( RHCOS )

- an optimized operating system created especially for the use of Container Orchestration Platforms
- each version of RHCOS comes with a specific version of Podman Container Engine and CRI-O Container Runtime
- RHCOS enforces many best practices and security features
- it allows writing to only folders the application will have read/write access
- if an application attempts to modify a read-only folder RHCOS will not allow those applications to continue running
- RHCOS also reserves many Ports for the internal use of Openshift
- User applications will not have write access to certain reserved folders, user applications are allowed to perform things as non-admin users only, only certain special applications will have admin/root access

## Points to remember

- Red Hat Openshift uses RedHat Enterprise Linux Core OS
- RHCOS has many restrictions or insists best practices
- RHEL Core OS reserves ports under 1024 for its internal use
- Many folders within the OS are made as read only
- Any application Pod attempts to perform write operation on those restricted folders will not be allowed to run
- For detailed documentation, please refer official documentation here  
<https://docs.openshift.com/container-platform/4.8/architecture/architecture-rhcos.html>

## Info - Pod Lifecycle

- Pending - Container image gets downloaded or there are no Persistent Volume to bind and claim them
- Running - The Pod is scheduled to a node and all containers in the Pod are up and running
- Succeeded - All containers in the Pod have terminated successfully and not been restarted
- Failed - All containers in the Pod have terminated but one or more containers terminated with non-zero status or was terminated by Openshift

- Unknown - For some reason, the state of the Pod could not be obtained may be there is some problem in communicating to the node where the Pod is running

## Info - Container Lifecycle

- Waiting - pulling the container image
- Running - container is running without issues
- Terminated - container in the Terminated state began execution and then either ran to completion or failed for some reason

## Info - NodePort vs Route

- NodePort is an external service
- It is a K8s features, which is also supported in openshift
- Kubernetes/OpenShift reserve ports in range 30000-32767 for the purpose of NodePorts
- For each, NodePort service we create one of the ports from the above range will be allotted for the service
- the chosen nodeport is opened in all the nodes for the nodeport service
- if we create 100 nodeport services, we end up opening 100 firewall ports on all the nodes, which is a security concern
- also nodeport service is not end-user friendly or developer friendly as they are accessed via node hostname/ip address, ideally the end-user should not have worry about how many nodes are part of openshift
- routes is based on Kubernetes ingress, which provides an easy to access public url which is user-friendly as opposed to nodeport service
- hence, in openshift for internal service, we can create clusterip service
- for external access, we just need to expose the clusterip service as a route
- we don't have to use node-port service in openshift

## Info - Deployment vs DeploymentConfigs

- In older version of Kubernetes, we had to use ReplicationController to deploy applications into Kubernetes/OpenShift
- The Red Hat OpenShift team, at that time added DeploymentConfig to allow deploying application in the declarative style as the ReplicationController doesn't support deploying application in the declarative style
- Meanwhile, the Google Kubernetes team & community added Deployment and ReplicaSet resource as an alternate for ReplicationController
- Hence, in OpenShift the Red Hat team deprecated the use of DeploymentConfig as Deployment and DeploymentConfig pretty much does the same
- Kubernetes, deprecated the use of ReplicationController
- Hence, whenever we deploy new application we need to choose Deployment over the DeploymentConfig as DeploymentConfig internally uses ReplicationController

## Lab - List the OpenShift nodes

```
oc get nodes
kubectl get nodes
oc get nodes -o wide
kubectl get nodes -o wide
oc version
kubectl version
```

## Expected output

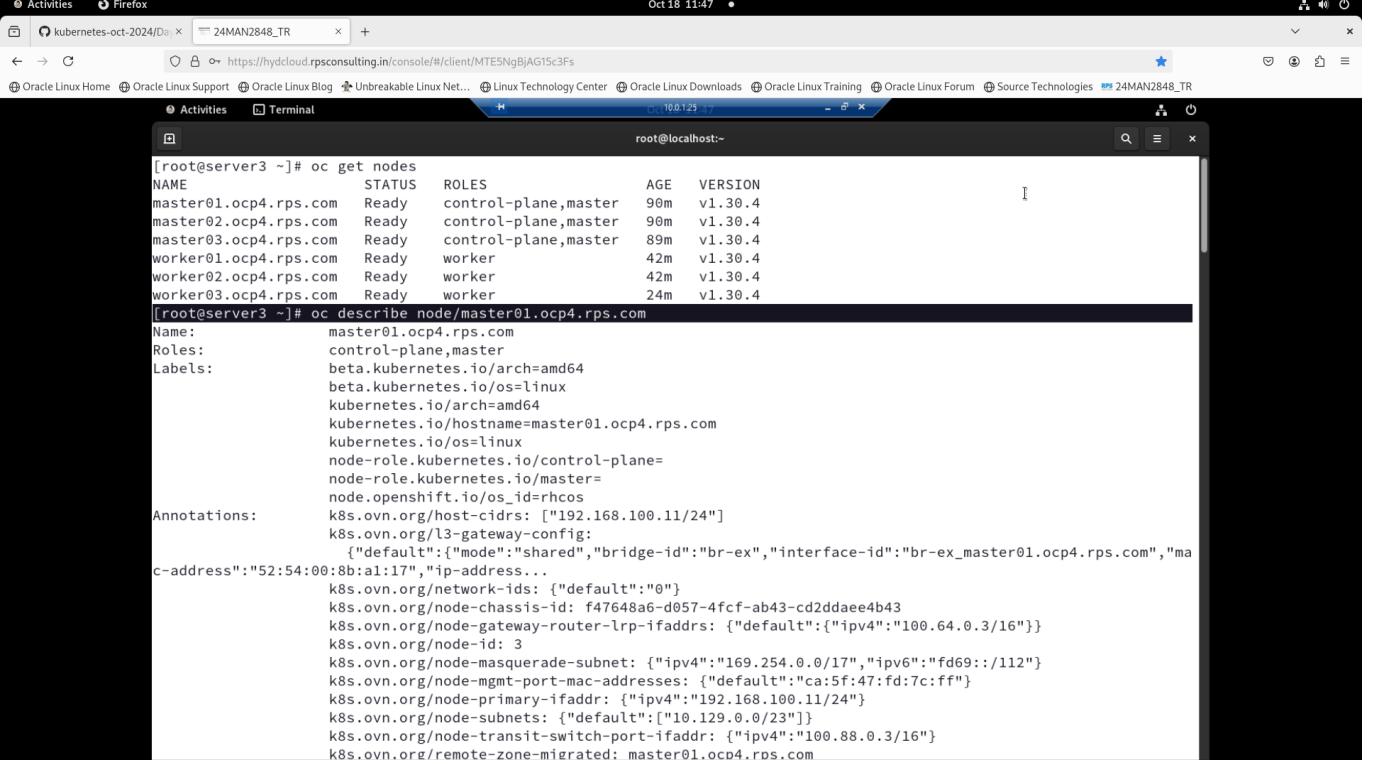
```
[root@server3 ~]# oc get nodes
NAME           STATUS   ROLES          AGE    VERSION
master01.ocp4.rps.com  Ready    control-plane,master  82m    v1.30.4
master02.ocp4.rps.com  Ready    control-plane,master  82m    v1.30.4
master03.ocp4.rps.com  Ready    control-plane,master  82m    v1.30.4
worker01.ocp4.rps.com  Ready    worker           35m    v1.30.4
worker02.ocp4.rps.com  Ready    worker           35m    v1.30.4
worker03.ocp4.rps.com  Ready    worker           16m    v1.30.4
[root@server3 ~]# kubectl get nodes
NAME           STATUS   ROLES          AGE    VERSION
master01.ocp4.rps.com  Ready    control-plane,master  82m    v1.30.4
master02.ocp4.rps.com  Ready    control-plane,master  82m    v1.30.4
master03.ocp4.rps.com  Ready    control-plane,master  82m    v1.30.4
worker01.ocp4.rps.com  Ready    worker           35m    v1.30.4
worker02.ocp4.rps.com  Ready    worker           35m    v1.30.4
worker03.ocp4.rps.com  Ready    worker           16m    v1.30.4
[root@server3 ~]# oc version
Client Version: 4.17.0
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
Server Version: 4.17.1
Kubernetes Version: v1.30.4
[root@server3 ~]#
```

```
[root@server3 ~]# oc get nodes -o wide
NAME           STATUS   ROLES          AGE    VERSION   KERNEL-VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE
CONTAINER-RUNTIME
master01.ocp4.rps.com  Ready    control-plane,master  88m    v1.30.4   5.14.0-427.40.1.el9_4.x86_64   192.168.100.11  <none>      Red Hat Enterpr
ise Linux CoreOS 417.94.202410090854-0 5.14.0-427.40.1.el9_4.x86_64   cri-o://1.30.6-3.rhaos4.17.git49b5172.el9
master02.ocp4.rps.com  Ready    control-plane,master  88m    v1.30.4   5.14.0-427.40.1.el9_4.x86_64   192.168.100.12  <none>      Red Hat Enterpr
ise Linux CoreOS 417.94.202410090854-0 5.14.0-427.40.1.el9_4.x86_64   cri-o://1.30.6-3.rhaos4.17.git49b5172.el9
master03.ocp4.rps.com  Ready    control-plane,master  88m    v1.30.4   5.14.0-427.40.1.el9_4.x86_64   192.168.100.13  <none>      Red Hat Enterpr
ise Linux CoreOS 417.94.202410090854-0 5.14.0-427.40.1.el9_4.x86_64   cri-o://1.30.6-3.rhaos4.17.git49b5172.el9
worker01.ocp4.rps.com  Ready    worker           41m    v1.30.4   5.14.0-427.40.1.el9_4.x86_64   192.168.100.21  <none>      Red Hat Enterpr
ise Linux CoreOS 417.94.202410090854-0 5.14.0-427.40.1.el9_4.x86_64   cri-o://1.30.6-3.rhaos4.17.git49b5172.el9
worker02.ocp4.rps.com  Ready    worker           41m    v1.30.4   5.14.0-427.40.1.el9_4.x86_64   192.168.100.22  <none>      Red Hat Enterpr
ise Linux CoreOS 417.94.202410090854-0 5.14.0-427.40.1.el9_4.x86_64   cri-o://1.30.6-3.rhaos4.17.git49b5172.el9
worker03.ocp4.rps.com  Ready    worker           22m    v1.30.4   5.14.0-427.40.1.el9_4.x86_64   192.168.100.23  <none>      Red Hat Enterpr
ise Linux CoreOS 417.94.202410090854-0 5.14.0-427.40.1.el9_4.x86_64   cri-o://1.30.6-3.rhaos4.17.git49b5172.el9
[root@server3 ~]#
```

## Lab - Finding more details about an openshift node

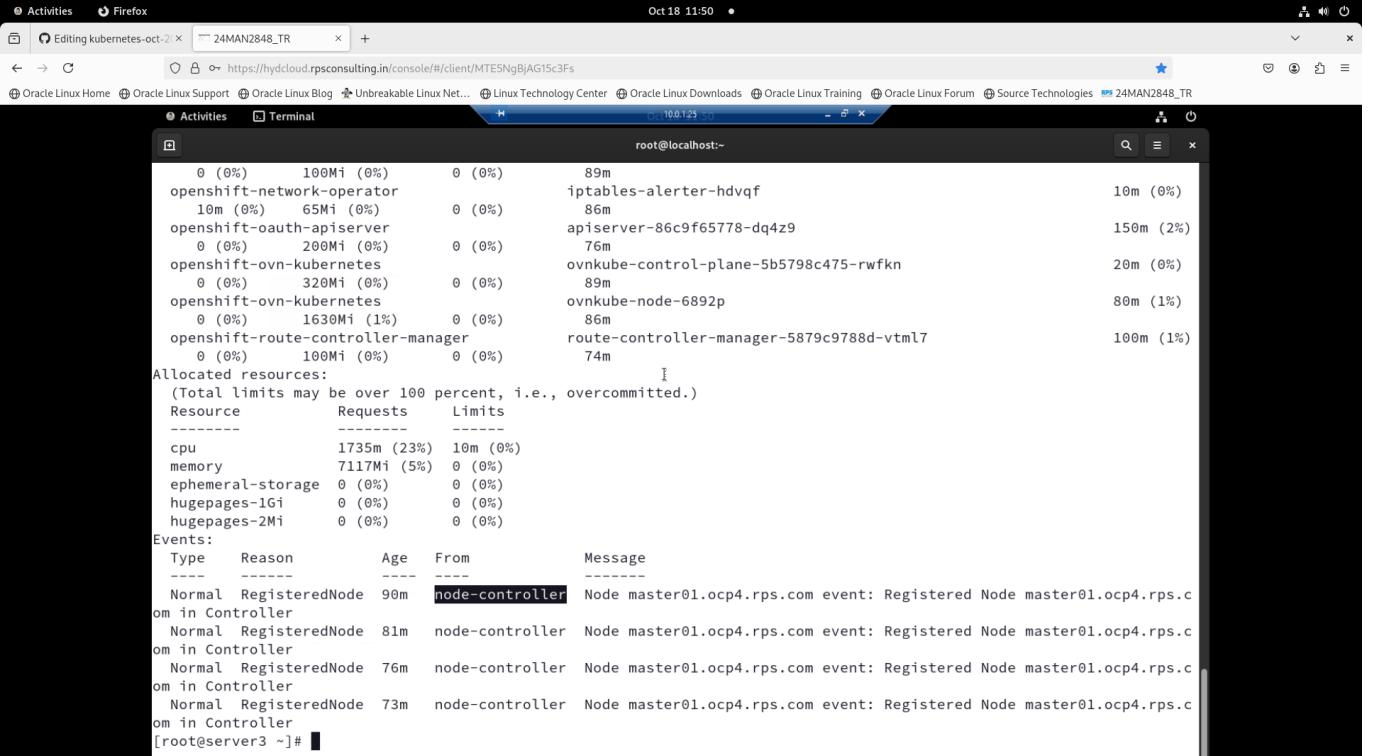
```
oc describe node/master01.ocp4.rps.com
```

## Expected output



```
[root@server3 ~]# oc get nodes
NAME        STATUS   ROLES      AGE    VERSION
master01.ocp4.rps.com  Ready    control-plane,master  90m   v1.30.4
master02.ocp4.rps.com  Ready    control-plane,master  90m   v1.30.4
master03.ocp4.rps.com  Ready    control-plane,master  89m   v1.30.4
worker01.ocp4.rps.com  Ready    worker     42m   v1.30.4
worker02.ocp4.rps.com  Ready    worker     42m   v1.30.4
worker03.ocp4.rps.com  Ready    worker     24m   v1.30.4

[root@server3 ~]# oc describe node/master01.ocp4.rps.com
Name:           master01.ocp4.rps.com
Roles:          control-plane,master
Labels:         beta.kubernetes.io/arch=amd64
                beta.kubernetes.io/os=linux
                kubernetes.io/arch=amd64
                kubernetes.io/hostname=master01.ocp4.rps.com
                kubernetes.io/os=linux
Annotations:   k8s.ovn.org/host-cidrs: ["192.168.100.11/24"]
                k8s.ovn.org/l3-gateway-config:
                  {"default":{"mode":"shared","bridge-id":"br-ex","interface-id":"br-ex_master01.ocp4.rps.com","mac-address":"52:54:00:8b:a1:17","ip-address...}}
                k8s.ovn.org/network-ids: {"default":"0"}
                k8s.ovn.org/node-chassis-id: f47648a6-d057-4fcf-ab43-cd2ddae4b43
                k8s.ovn.org/node-gateway-router-lrp-ifaddrs: {"default":{"ipv4":"100.64.0.3/16"}}
                k8s.ovn.org/node-id: 3
                k8s.ovn.org/node-masquerade-subnet: {"ipv4":"169.254.0.0/17","ipv6":"fd69::/112"}
                k8s.ovn.org/node-mgmt-port-mac-addresses: {"default":"ca:5f:47:fd:7c:ff"}
                k8s.ovn.org/node-primary-ifaddr: {"ipv4":"192.168.100.11/24"}
                k8s.ovn.org/node-subnets: {"default":["10.129.0.0/23"]}
                k8s.ovn.org/node-transit-switch-port-ifaddr: {"ipv4":"100.88.0.3/16"}
                k8s.ovn.org/remote-zone-migrated: master01.ocp4.rps.com



```
root@localhost:~#
          0 (0%) 100Mi (0%) 0 (0%) 89m iptables-alterer-hdvqf 10m (0%)
  openshift-network-operator 10m (0%) 65Mi (0%) 0 (0%) 86m apiserver-86c9f65778-dq4z9 150m (2%)
  openshift-oauth-apiserver 0 (0%) 200Mi (0%) 0 (0%) 76m ovnkube-control-plane-5b5798c475-rwfkn 20m (0%)
  openshift-ovn-kubernetes 0 (0%) 320Mi (0%) 0 (0%) 89m ovnkube-node-6892p 80m (1%)
  openshift-ovn-kubernetes 0 (0%) 1630Mi (1%) 0 (0%) 86m route-controller-manager-5879c9788d-vtml7 100m (1%)
  openshift-route-controller-manager 0 (0%) 100Mi (0%) 0 (0%) 74m
Allocated resources: [Total limits may be over 100 percent, i.e., overcommitted.]
Resource Requests Limits
----- -----
cpu      1735m (23%) 10m (0%)
memory  7117Mi (5%) 0 (0%)
ephemeral-storage 0 (0%) 0 (0%)
hugepages-1Gi 0 (0%) 0 (0%)
hugepages-2Mi 0 (0%) 0 (0%)
Events:
Type Reason Age From Message
---- ---- -- -----
Normal RegisteredNode 90m node-controller Node master01.ocp4.rps.com event: Registered Node master01.ocp4.rps.com in Controller
Normal RegisteredNode 81m node-controller Node master01.ocp4.rps.com event: Registered Node master01.ocp4.rps.com in Controller
Normal RegisteredNode 76m node-controller Node master01.ocp4.rps.com event: Registered Node master01.ocp4.rps.com in Controller
Normal RegisteredNode 73m node-controller Node master01.ocp4.rps.com event: Registered Node master01.ocp4.rps.com in Controller
[root@server3 ~]#
```


```

## Lab - Create a new project

The below command will create a new project and switches to the project

```
oc new-project jegan
```

## Expected output

```
[root@server3 ~]# oc new-project jegan
Now using project "jegan" on server "https://api.ocp4.rps.com:6443".
You can add applications to this project with the 'new-app' command. For example, try:
oc new-app rails-postgresql-example
to build a new example application in Ruby. Or use kubectl to deploy a simple Kubernetes application:
kubectl create deployment hello-node --image=registry.k8s.io/e2e-test-images/agnhost:2.43 -- /agnhost serve-hostname
[root@server3 ~]#
```

## Lab - Let's create a nginx deployment

```
oc project jegan
oc create deploy nginx --image=nginx:latest --replicas=3
oc get deploy,rs,po
```

## Expected output

```
[root@server3 ~]# # Find currently active project
[root@server3 ~]# oc project
Using project "krishna" on server "https://api.ocp4.rps.com:6443".
[root@server3 ~]#
[root@server3 ~]# # Switch to a different project
[root@server3 ~]# oc project jegan
Now using project "jegan" on server "https://api.ocp4.rps.com:6443".
[root@server3 ~]#
[root@server3 ~]# oc project
Using project "jegan" on server "https://api.ocp4.rps.com:6443".
[root@server3 ~]# oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
NAME           READY   STATUS    RESTARTS   AGE
pod/nginx-7584b6f84c-6vfz9   0/1     ImagePullBackoff   0          6m31s
pod/nginx-7584b6f84c-h8tkw   0/1     ErrImagePull    0          6m31s
pod/nginx-7584b6f84c-lkkbz   0/1     ErrImagePull    0          6m31s

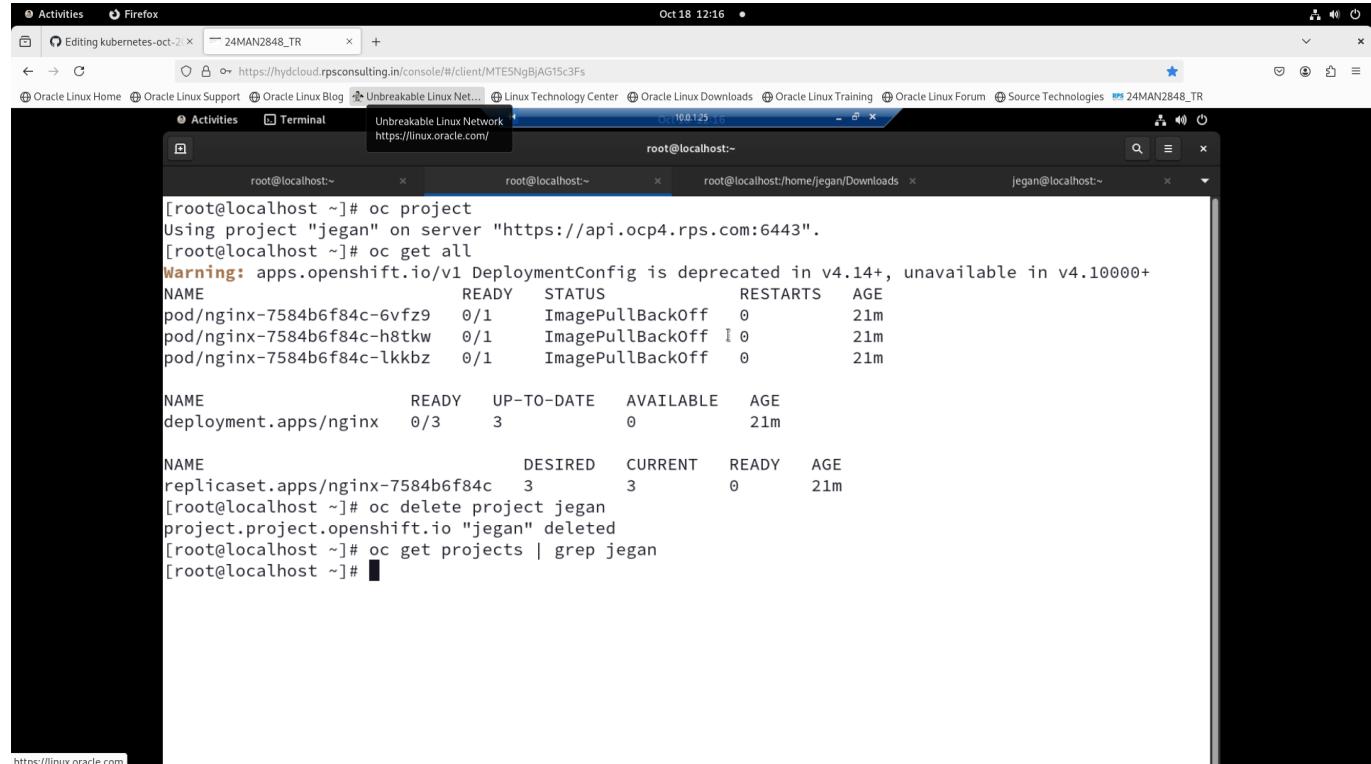
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   0/3      3           0          6m31s

NAME           DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-7584b6f84c  3         3         0          6m31s
[root@server3 ~]#
```

## Lab - Deleting a project along with all resources in it

```
oc project  
oc get all  
oc delete project jegan  
oc get projects | grep jegan
```

### Expected output



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window has four tabs: root@localhost, root@localhost, root@localhost:/home/jegan/Downloads, and jegan@localhost. The history in the terminal shows the following commands:

```
[root@localhost ~]# oc project  
Using project "jegan" on server "https://api.ocp4.rps.com:6443".  
[root@localhost ~]# oc get all  
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+  
NAME READY STATUS RESTARTS AGE  
pod/nginx-7584b6f84c-6vfz9 0/1 ImagePullBackoff 0 21m  
pod/nginx-7584b6f84c-h8tkw 0/1 ImagePullBackoff 0 21m  
pod/nginx-7584b6f84c-lkkbz 0/1 ImagePullBackoff 0 21m  
  
NAME READY UP-TO-DATE AVAILABLE AGE  
deployment.apps/nginx 0/3 3 0 21m  
  
NAME DESIRED CURRENT READY AGE  
replicaset.apps/nginx-7584b6f84c 3 3 0 21m  
[root@localhost ~]# oc delete project jegan  
project.project.openshift.io "jegan" deleted  
[root@localhost ~]# oc get projects | grep jegan  
[root@localhost ~]#
```

## Lab - Deploying an application in openshift S2I(source to Image)

```
oc project jegan  
oc new-app --name=hello https://github.com/tektutor/spring-ms.git
```

## Expected output

```
[jegan@server3 ~]$ oc project
Using project "jegan" on server "https://api.ocp4.rps.com:6443".
[jegan@server3 ~]$ oc new-app --name=hello https://github.com/tektutor/spring-ms.git
--> Found container image 303c87a (42 hours old) from registry.access.redhat.com for "registry.access.redhat.com/ubi8/openjdk-11"

  Java Applications
  -----
  Platform for building and running plain Java applications (fat-jar and flat classpath)

  Tags: builder, java

  * An image stream tag will be created as "openjdk-11:latest" that will track the source image
  * A Docker build using source code from https://github.com/tektutor/spring-ms.git will be created
    * The resulting image will be pushed to image stream tag "hello:latest"
    * Every time "openjdk-11:latest" changes a new build will be triggered

--> Creating resources ...
imagestream.image.openshift.io "openjdk-11" created
imagestream.image.openshift.io "hello" created
buildconfig.build.openshift.io "hello" created
deployment.apps "hello" created
service "hello" created
--> Success
WARNING: No container image registry has been configured with the server. Automatic builds and deployments may not function.
Build scheduled, use 'oc logs -f buildconfig/hello' to track its progress.
```

```
[jegan@server3 ~]$ oc project
Using project "jegan" on server "https://api.ocp4.rps.com:6443".
[jegan@server3 ~]$ oc new-app --name=hello https://github.com/tektutor/spring-ms.git
--> Found container image 303c87a (42 hours old) from registry.access.redhat.com for "registry.access.redhat.com/ubi8/openjdk-11"

  Java Applications
  -----
  Platform for building and running plain Java applications (fat-jar and flat classpath)

  Tags: builder, java

  * An image stream tag will be created as "openjdk-11:latest" that will track the source image
  * A Docker build using source code from https://github.com/tektutor/spring-ms.git will be created
    * The resulting image will be pushed to image stream tag "hello:latest"
    * Every time "openjdk-11:latest" changes a new build will be triggered

--> Creating resources ...
imagestream.image.openshift.io "openjdk-11" created
imagestream.image.openshift.io "hello" created
buildconfig.build.openshift.io "hello" created
deployment.apps "hello" created
service "hello" created
--> Success
WARNING: No container image registry has been configured with the server. Automatic builds and deployments may not function.
Build scheduled, use 'oc logs -f buildconfig/hello' to track its progress.
Application is not exposed. You can expose services to the outside world by executing one or more of the commands below:
  'oc expose service/hello'
  Run 'oc status' to view your app.
[jegan@server3 ~]$
```

## Lab - Importing image into OpenShift Internal registry

```
oc import-image ubi8/openjdk-11:1.20-2.1727147549 --
from=registry.access.redhat.com/ubi8/openjdk-11:1.20-2.1727147549 --confirm
```

## Expected output

```
[root@server3 ~]# oc import-image ubi8/openjdk-11:1.20-2.1727147549 --from=registry.access.redhat.com/ubi8/openjdk-11
imagestream.image.openshift.io/openjdk-11 imported

Name:          openjdk-11
Namespace:     jegan
Created:       Less than a second ago
Labels:        <none>
Annotations:   openshift.io/image.dockerRepositoryCheck=2024-10-18T10:23:13Z
Image Repository: default-route-openshift-image-registry.apps.ocp4.rps.com/jegan/openjdk-11
Image Lookup:  local=false
Unique Images: 1
Tags:          1

1.20-2.1727147549
tagged from registry.access.redhat.com/ubi8/openjdk-11:sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Less than a second ago

* registry.access.redhat.com/ubi8/openjdk-11:sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Less than a second ago

Image Name:    openjdk-11:1.20-2.1727147549
Docker Image:  registry.access.redhat.com/ubi8/openjdk-11:sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Name:          sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Created:       Less than a second ago
Annotations:   image.openshift.io/dockerLayersOrder=ascending
Image Size:    162.7MB in 2 layers
Layers:        39.37MB sha256:2384c7c17092245bda9218fee9b2ae475ee8a53cd8a66e63c1d5f37433276ff0
                123.3MB sha256:c42516183e58cf322b5770bd89b28534fc2b41208127e205c77030021d6cd9ae
Image Created: 3 weeks ago
Author:        <none>
```

```
root@localhost:~          jegan@localhost:~          root@server3:-
Oct 18 16:10 •          Oct 18 16:10 •          Oct 18 16:10 •

Image Name:    openjdk-11:1.20-2.1727147549
Docker Image:  registry.access.redhat.com/ubi8/openjdk-11:sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Name:          sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Created:       Less than a second ago
Annotations:   image.openshift.io/dockerLayersOrder=ascending
Image Size:    162.7MB in 2 layers
Layers:        39.37MB sha256:2384c7c17092245bda9218fee9b2ae475ee8a53cd8a66e63c1d5f37433276ff0
                123.3MB sha256:c42516183e58cf322b5770bd89b28534fc2b41208127e205c77030021d6cd9ae
Image Created: 3 weeks ago
Author:        <none>
Arch:          amd64
Command:       /usr/local/s2i/run
Working Dir:   /home/jboss
User:          185
Expose Ports:  8080/tcp, 8443/tcp, 8778/tcp
Docker Labels: architecture=x86_64
               build-date=2024-09-24T03:28:40
               com.redhat.component=openjdk-11-ubi8-container
               com.redhat.license_terms=https://www.redhat.com/en/about/red-hat-end-user-license-agreements#UBI
               description=Source To Image (S2I) image for Red Hat OpenShift providing OpenJDK 11
               distribution-scope=public
               io.buildah.version=1.33.8
               io.cekit.version=4.13.0.dev0
               io.fabric8.s2i.version.jolokia=1.6.2-redhat-00002
               io.fabric8.s2i.version.maven=3.8
               io.k8s.description=Platform for building and running plain Java applications (fat-jar and flat classpath)
               io.k8s.display-name=Java Applications
               io.openshift.expose-services=
               io.openshift.s2i.destination=/tmp
```

```

root@localhost:~          jegan@localhost:~          root@server3:~
release=2.1727147549
summary=Source To Image (S2I) image for Red Hat OpenShift providing OpenJDK 11
url=https://access.redhat.com/containers/#/registry.access.redhat.com/ubi8/openjdk-11/images/1.20-2.1727147549
usage=https://jboss-container-images.github.io/openjdk/
vcs-ref=f8db8e8d4a9162b6828f7d1674f58958a5bcd241
vcs-type=git
vendor=Red Hat, Inc.
version=1.20
container=oci
GECOS=JBoss user
HOME=/home/jboss
UID=185
USER=jboss
JAVA_HOME=/usr/lib/jvm/java-11
JAVA_VENDOR=openjdk
JAVA_VERSION=11
JBoss_CONTAINER_OPENJDK_JDK_MODULE=/opt/jboss/container/openjdk/jdk
AB_PROMETHEUS_JMX_EXPORTER_CONFIG=/opt/jboss/container/prometheus/etc/jmx-exporter-config.yaml
JBoss_CONTAINER_PROMETHEUS_MODULE=/opt/jboss/container/prometheus
AB_JOLOKIA_AUTH_OPENSHIFT=true
AB_JOLOKIA_HTTPS=true
AB_JOLOKIA_PASSWORD_RANDOM=true
JBoss_CONTAINER_JOLOKIA_MODULE=/opt/jboss/container/jolokia
JOLOKIA_VERSION=1.6.2
JBoss_CONTAINER_MAVEN_38_MODULE=/opt/jboss/container/maven/38/
MAVEN_VERSION=3.8
S2I_SOURCE_DEPLOYMENTS_FILTER=*.jar quarkus-app
JBoss_CONTAINER_S2I_CORE_MODULE=/opt/jboss/container/s2i/core/
JBoss_CONTAINER_JAVA_PROXY_MODULE=/opt/jboss/container/java/proxy
JBoss_CONTAINER_JAVA_JVM_MODULE=/opt/jboss/container/java/jvm
JBoss_CONTAINER_UTIL_LOGGING_MODULE=/opt/jboss/container/util/logging/
JBoss_CONTAINER_MAVEN_DEFAULT_MODULE=/opt/jboss/container/maven/default/
JBoss_CONTAINER_MAVEN_S2I_MODULE=/opt/jboss/container/maven/s2i
JAVA_DATA_DIR=/deployments/data
JBoss_CONTAINER_JAVA_RUN_MODULE=/opt/jboss/container/java/run
JBoss_CONTAINER_JAVA_S2I_MODULE=/opt/jboss/container/java/s2i
JBoss_IMAGE_NAME=ubi8/openjdk-11
JBoss_IMAGE_VERSION=1.20
LANG=C.UTF8
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/s2i

[root@server3 ~]# oc get is
NAME           IMAGE REPOSITORY
openjdk-11     default-route-openshift-image-registry.apps.ocp4.rps.com/jegan/openjdk-11   TAGS      UPDATED
ago            ago                                         1.20-2.1727147549   6 seconds

```

## Lab - Deploy your custom application using S2I source strategy

```

oc new-project jegan
oc new-app --name=hello openjdk-11:1.20-
2.1727147549~https://github.com/tektutor/spring-ms.git --strategy=source

```

## Expected output

```

root@server3:~# oc new-app --name=hello-ms openjdk-11:1.20-2.1727147549-https://github.com/tektutor/spring-ms.git --strategy=source
warning: Cannot check if git requires authentication.
--> Found image 2bb4972 (3 weeks old) in image stream "jegan/openjdk-11" under tag "1.20-2.1727147549" for "openjdk-11:1.20-2.1727147549"

Java Applications
-----
Platform for building and running plain Java applications (fat-jar and flat classpath)

Tags: builder, java

* A source build using source code from https://github.com/tektutor/spring-ms.git will be created
* The resulting image will be pushed to image stream tag "hello-ms:latest"
* Use 'oc start-build' to trigger a new build

--> Creating resources ...
imagestream.image.openshift.io "hello-ms" created
buildconfig.build.openshift.io "hello-ms" created
deployment.apps "hello-ms" created
service "hello-ms" created
--> Success
Build scheduled, use 'oc logs -f buildconfig/hello-ms' to track its progress.
Application is not exposed. You can expose services to the outside world by executing one or more of the commands below:
'oc expose service/hello-ms'
Run 'oc status' to view your app.
[root@server3 ~]# oc status
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
In project jegan on server https://api.ocp4.rps.com:6443

```

```

root@server3:~# oc new-app --name=hello-ms openjdk-11:1.20-2.1727147549-https://github.com/tektutor/spring-ms.git --strategy=source
warning: Cannot check if git requires authentication.
--> Found image 2bb4972 (3 weeks old) in image stream "jegan/openjdk-11" under tag "1.20-2.1727147549" for "openjdk-11:1.20-2.1727147549"

Java Applications
-----
Platform for building and running plain Java applications (fat-jar and flat classpath)

Tags: builder, java

* A source build using source code from https://github.com/tektutor/spring-ms.git will be created
* The resulting image will be pushed to image stream tag "hello-ms:latest"
* Use 'oc start-build' to trigger a new build

--> Creating resources ...
imagestream.image.openshift.io "hello-ms" created
buildconfig.build.openshift.io "hello-ms" created
deployment.apps "hello-ms" created
service "hello-ms" created
--> Success
Build scheduled, use 'oc logs -f buildconfig/hello-ms' to track its progress.
Application is not exposed. You can expose services to the outside world by executing one or more of the commands below:
'oc expose service/hello-ms'
Run 'oc status' to view your app.
[root@server3 ~]# oc status
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
In project jegan on server https://api.ocp4.rps.com:6443

svc/hello-ms - 172.30.83.251 ports 8080, 8443, 8778
  deployment/hello-ms deploys istag/hello-ms:latest <
    bc/hello-ms source builds https://github.com/tektutor/spring-ms.git on istag/openjdk-11:1.20-2.1727147549
      build #1 running for 5 seconds - d6f73b7: Update Dockerfile (Jeganathan Swaminathan <mail2jegan@gmail.com>)
    deployment #1 running for 5 seconds - 0/1 pods growing to 1

1 info identified, use 'oc status --suggest' to see details.
[root@server3 ~]# oc status --suggest
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
In project jegan on server https://api.ocp4.rps.com:6443

svc/hello-ms - 172.30.83.251 ports 8080, 8443, 8778
  deployment/hello-ms deploys istag/hello-ms:latest <
    bc/hello-ms source builds https://github.com/tektutor/spring-ms.git on istag/openjdk-11:1.20-2.1727147549

```

```

root@localhost:~          jegan@localhost:~          root@server3:~
deployment #1 running for 5 seconds - 0/1 pods growing to 1

1 info identified, use 'oc status --suggest' to see details.
[root@server3 ~]# oc status --suggest
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
In project jegan on server https://api.ocp4.rps.com:6443

svc/hello-ms - 172.30.83.251 ports 8080, 8443, 8778
  deployment/hello-ms deploys istag/hello-ms:latest <-
    bc/hello-ms source builds https://github.com/tektutor/spring-ms.git on istag/openjdk-11:1.20-2.1727147549
      build #1 running for 11 seconds - d6f73b7: Update Dockerfile (Jeganathan Swaminathan <mail2jegan@gmail.com>)
    deployment #1 running for 11 seconds - 0/1 pods growing to 1

Info:
  * deployment/hello-ms has no liveness probe to verify pods are still running.
    try: oc set probe deployment/hello-ms --liveness ...

View details with 'oc describe <resource>/<name>' or list resources with 'oc get all'.
[root@server3 ~]# oc get bc
NAME      TYPE      FROM      LATEST
hello-ms   Source    Git       1
[root@server3 ~]# oc describe bc/hello-ms
Name:           hello-ms
Namespace:     jegan
Created:       25 seconds ago
Labels:        app=hello-ms
               app.kubernetes.io/component=hello-ms
               app.kubernetes.io/instance=hello-ms
Annotations:  openshift.io/generated-by=OpenShiftNewApp
Latest Version: 1

```

```

root@localhost:~          jegan@localhost:~          root@server3:~
Annotations:  openshift.io/generated-by=OpenShiftNewApp
Latest Version: 1

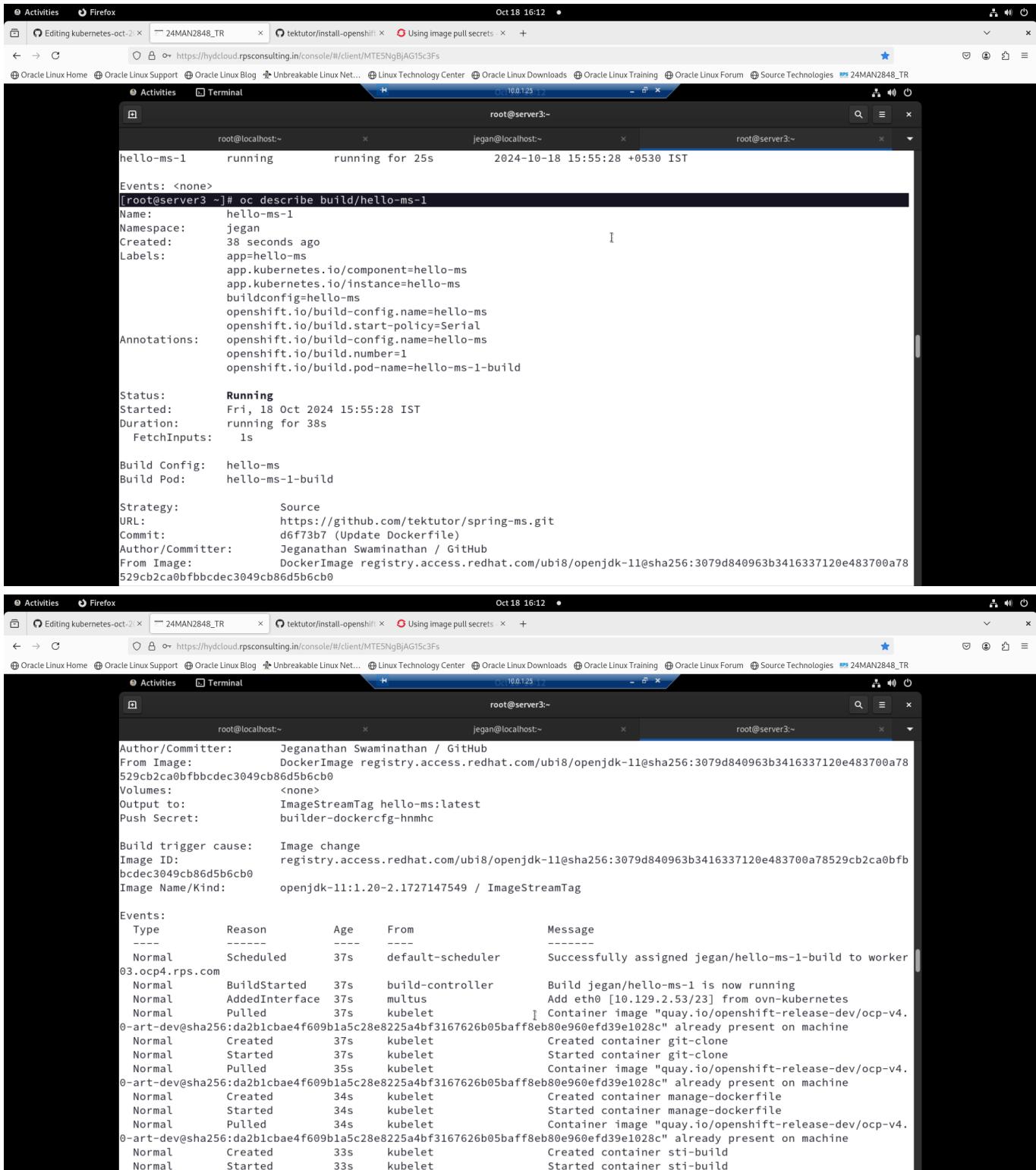
Strategy:      Source
URL:          https://github.com/tektutor/spring-ms.git
From Image:    ImageStreamTag jegan/openjdk-11:1.20-2.1727147549
Volumes:       <none>
Output to:     ImageStreamTag hello-ms:latest

Build Run Policy:  Serial
Triggered by:    Config, ImageChange
Webhook GitHub:
  URL:          https://api.ocp4.rps.com:6443/apis/build.openshift.io/v1/namespaces/jegan/buildconfigs/hello-ms/webhooks/<secret>/github
Webhook Generic:
  URL:          https://api.ocp4.rps.com:6443/apis/build.openshift.io/v1/namespaces/jegan/buildconfigs/hello-ms/webhooks/<secret>/generic
  AllowEnv:    false
Builds History Limit:
  Successful:  5
  Failed:     5

Build          Status          Duration          Creation Time
hello-ms-1     running        running for 25s    2024-10-18 15:55:28 +0530 IST

Events: <none>
[root@server3 ~]# oc describe build/hello-ms-1
Name:           hello-ms-1
Namespace:     jegan
Created:       38 seconds ago

```



```

root@server3:~# oc describe build/hello-ms-1
Name:           hello-ms-1
Namespace:      jegan
Created:        38 seconds ago
Labels:         app=hello-ms
                app.kubernetes.io/component=hello-ms
                app.kubernetes.io/instance=hello-ms
                buildconfig=hello-ms
Annotations:   openshift.io/build-config.name=hello-ms
                openshift.io/build.start-policy=Serial
                openshift.io/build-config.name=hello-ms
                openshift.io/build.number=1
                openshift.io/build.pod-name=hello-ms-1-build

Status:        Running
Started:       Fri, 18 Oct 2024 15:55:28 IST
Duration:      running for 38s
FetchInputs:   ls

Build Config:  hello-ms
Build Pod:     hello-ms-1-build

Strategy:      Source
URL:          https://github.com/tektutor/spring-ms.git
Commit:        d6f73b7 (Update Dockerfile)
Author/Committer: Jeganathan Swaminathan / GitHub
From Image:   DockerImage registry.access.redhat.com/ubi8/openjdk-11@sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0

root@server3:~# oc logs -f hello-ms-1-build
Author/Committer: Jeganathan Swaminathan / GitHub
From Image:   DockerImage registry.access.redhat.com/ubi8/openjdk-11@sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Volumes:       <none>
Output to:    ImageStreamTag hello-ms:latest
Push Secret:  builder-dockercfg-hnmhc

Build trigger cause: Image change
Image ID:      registry.access.redhat.com/ubi8/openjdk-11@sha256:3079d840963b3416337120e483700a78529cb2ca0bfbbcd3049cb86d5b6cb0
Image Name/Kind: openjdk-11:1.20-2.1727147549 / ImageStreamTag

Events:
  Type      Reason     Age     From               Message
  ----      ----      ---     ----              -----
  Normal    Scheduled  37s    default-scheduler   Successfully assigned jegan/hello-ms-1-build to worker-03.ocp4.rps.com
  Normal    BuildStarted 37s    build-controller   Build jegan/hello-ms-1 is now running
  Normal    AddedInterface 37s    multus            Add eth0 [10.129.2.53/23] from ovn-kubernetes
  Normal    Pulled       37s    kubelet          Container image "quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:da2b1cbae4f609b1a5c28e8225a4bf3167626b05baff8eb80e960efd39e1028c" already present on machine
  Normal    Created     37s    kubelet          Created container git-clone
  Normal    Started     37s    kubelet          Started container git-clone
  Normal    Pulled       35s    kubelet          Container image "quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:da2b1cbae4f609b1a5c28e8225a4bf3167626b05baff8eb80e960efd39e1028c" already present on machine
  Normal    Created     34s    kubelet          Created container manage-dockerfile
  Normal    Started     34s    kubelet          Started container manage-dockerfile
  Normal    Pulled       34s    kubelet          Container image "quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:da2b1cbae4f609b1a5c28e8225a4bf3167626b05baff8eb80e960efd39e1028c" already present on machine
  Normal    Created     33s    kubelet          Created container sti-build
  Normal    Started     33s    kubelet          Started container sti-build

```

```

root@localhost:~          jegan@localhost:~          root@server3:~
03.ocp4.rps.com
Normal BuildStarted 37s build-controller      Build jegan/hello-ms-1 is now running
Normal AddedInterface 37s multus             Add eth0 [10.129.2.53/23] from ovn-kubernetes
Normal Pulled       37s kubelet            Container image "quay.io/openshift-release-dev/ocp-v4.
0-art-dev@sha256:da2b1cbae4f609b1a5c28e8225a4bf3167626b05baff8eb80e960efd39e1028c" already present on machine
Normal Created     37s kubelet            Created container git-clone
Normal Started    37s kubelet            Started container git-clone
Normal Pulled       35s kubelet            Container image "quay.io/openshift-release-dev/ocp-v4.
0-art-dev@sha256:da2b1cbae4f609b1a5c28e8225a4bf3167626b05baff8eb80e960efd39e1028c" already present on machine
Normal Created     34s kubelet            Created container manage-dockerfile
Normal Started    34s kubelet            Started container manage-dockerfile
Normal Pulled       34s kubelet            Container image "quay.io/openshift-release-dev/ocp-v4.
0-art-dev@sha256:da2b1cbae4f609b1a5c28e8225a4bf3167626b05baff8eb80e960efd39e1028c" already present on machine
Normal Created     33s kubelet            Created container sti-build
Normal Started    33s kubelet            Started container sti-build
[root@server3 ~]# oc describe build/hello-ms-1
Name:           hello-ms-1
Namespace:      jegan
Created:        50 seconds ago
Labels:         app=hello-ms
                app.kubernetes.io/component=hello-ms
                app.kubernetes.io/instance=hello-ms
                buildconfig=hello-ms
Annotations:   openshift.io/build-config.name=hello-ms
                openshift.io/build.start-policy=Serial
                openshift.io/build-config.name=hello-ms
                openshift.io/build.number=1
                openshift.io/build.pod-name=hello-ms-1-build
Status:         Running
Started:        Fri, 18 Oct 2024 15:55:28 IST

```

Uploading image.png...

## Further references

<https://kubernetes.io/docs/concepts/cluster-administration/networking/#the-kubernetes-network-model>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-networking>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-network-policy>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-kubernetes-ingress>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-kubernetes-egress>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-kubernetes-services>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-ebpf>

<https://docs.tigera.io/calico/latest/about/kubernetes-training/about-k8s-networking>

<https://www.tigera.io/blog/everything-you-need-to-know-about-kubernetes-pod-networking-on-aws/>

<https://www.tigera.io/blog/everything-you-need-to-know-about-kubernetes-networking-on-azure/>

<https://www.tigera.io/blog/everything-you-need-to-know-about-kubernetes-networking-on-google-cloud/>

[https://docs.openshift.com/container-platform/4.8/windows\\_containers/understanding-windows-container-workloads.html#understanding-windows-container-workloads](https://docs.openshift.com/container-platform/4.8/windows_containers/understanding-windows-container-workloads.html#understanding-windows-container-workloads)