

Day 1

Cloning this repository on the RPS Linux machine

```
cd ~
git clone https://github.com/tektutor/openshift-june-2024.git
cd openshift-june-2024
ls
```

Testing lab environment

```
oc version
kubectl version

oc get nodes
kubectl get nodes

cat ~/openshift.txt
```

Expected output

```
jegan@tektutor.org ~/openshift-june-2024 ✘ main ➔ oc version
Client Version: 4.15.0-202405141637.p0.g7693229.Assembly.Stream.el8-7693229
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
Server Version: 4.15.18
Kubernetes Version: v1.28.10+a2c84a5

jegan@tektutor.org ~/openshift-june-2024 ✘ main ➔ kubectl version
Client Version: v1.28.2
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
Server Version: v1.28.10+a2c84a5

jegan@tektutor.org ~/openshift-june-2024 ✘ main ➔ oc get nodes
NAME STATUS ROLES AGE VERSION
master-1.ocp4.tektutor.org.labs Ready control-plane,master,worker 3h25m v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs Ready control-plane,master,worker 3h25m v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs Ready control-plane,master,worker 3h25m v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs Ready worker 3h7m v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs Ready worker 3h8m v1.28.10+a2c84a5

jegan@tektutor.org ~/openshift-june-2024 ✘ main ➔ kubectl get nodes
NAME STATUS ROLES AGE VERSION
master-1.ocp4.tektutor.org.labs Ready control-plane,master,worker 3h25m v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs Ready control-plane,master,worker 3h25m v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs Ready control-plane,master,worker 3h25m v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs Ready worker 3h8m v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs Ready worker 3h8m v1.28.10+a2c84a5
```

What is Hypervisor ?

- virtualization technology
- it allows us running multiple os on the same machine

- many OS can be actively running side by side on the same machine
- each Virtual Machine represents one fully function operating system
- we can install windows/linux/mac in the virtual machine
- the Operating System installed inside the Virtual machine is called Guest OS
- the virtualization software is of two types
 1. Type 1 - Bare Metal Hypervisor (Installed in Servers/Workstations)
 2. Type 2 - Installed in Laptops/Desktops/Workstations
- this is considered heavy-weight virtualization as each Virtual Machine has to allocated with dedicated hardware resources
 - CPU Cores
 - RAM
 - Disk Storage
 - Graphics Card (Virtual)
 - Network Card (virtual)
- Examples
 - VMWare vSphere/vCenter - Type 1 Hypervisor (Bare Metal) - Commercial Product
 - VMWare Workstation - Type 2 - Linux/Windows - commercial product
 - VMWare Fusion - Type 2 - Mac - commerical product
 - Oracle Virtual Box - Type 2- works in Linux/Windows/Mac and it is Free
 - KVM - Type 2 - works in all Linux Distributions
 - Parallels - Type 2 - works in Mac
 - Hyper-V - Type 2 - works in Windows Server grade OS

Info - Containerization

- is an application virtualization technology
- each container represents one application process
- container are not operating sysytem, they don't their own OS Kernel
- each container either represents one fully function application or a application component
 - application component (database server, web server, application server, etc.,)
 - application - CRM, etc.,
 - an application might requires more than one container in some cases
- light-weight virtualization technology
 - containers doesn't require dedicated hardwares as they use the hardware resources available on the host operating system
- there are some similarities between containers and virtual machines
 - just like Virtual machines acquire their own IP, containers also get their own IP usually (Private IPs)
 - just like Virtual machines has a Network Card, containers also has a network card
 - just like Virtual machines has their own Network stack, containers also has their own network stack
 - containers has a file system just like virtual machine
 - containers has own port range (0 - 65535) just like virtual machines
- containers will never be able to replace virtual machine as virtual machine runs an Operating System, while container runs a single application

Info - Is it possible to run multiple application within a container?

- Yes, it is possible to run multiple applications in a container
- How docker or podman let's you run multiple applications inside a container is, the main container will run a utility called supervisord which will spin separate process to run the applications and the supervisord monitors the health and status of the child processes, which is a overhead, hence though it works it is not a recommended best practice
- recommendation is, one application per container

Info - Container Runtime Overview

- low-level software to manage container images and containers
- not user-friendly
- generally not used by end-users
- examples
 - CRI-O Container Runtime
 - runC container runtime

Info - Container Engine Overview

- high-level software used to manage containers and images
- very user-friendly
- without knowing low-level container related kernel knowledge we can easily manage containers
- internally container engines depend on Container Runtime to manage containers and images
- Examples
 - Docker is a Container Engine which depends on containerd which in turn depends on runC container runtime
 - Podman is a Container Engine which depends on CRI-O Container runtime

Info - Docker Overview

- is developed in Golang by Docker Inc organization
- comes in 2 flavours
 1. Community Edition - Docker CE (Free)
 2. Enterprise Edition - Docker EE (Paid)
- follow Client/Server Architecture
- in most cases, when we create containers they provide us root access irrespective of whether you are administrator or not
- Docker supports rootless containers

Info - Podman Overview

- is a container engine which internally depends on CRI-O Container Runtime
- Red Hat Openshift supported Docker(run-C) until v3.11
- Red Hat Openshift supports only CRI-O container runtime and Podman within Red Hat Openshift
- Red Hat Openshift v4.x onwards Docker and other container runtime support is removed

Info - Container Orchestration Platforms Overview

- though containerized application workloads can be managed manually, in real-world application companies don't manage containers directly/manually
- generally every organization uses Container Orchestration Platforms to manage their containerized application workloads
- Examples
 - Docker SWARM
 - Google Kubernetes
 - Red Hat Openshift
 - AWS - Kubernetes Managed Service called EKS (Elastic Kubernetes Service)
 - Azure - Kubernetes Managed Service called AKS (Azure Kuberentes Service)
 - AWS - Red Hat Managed Openshift service called ROSA
 - Azure - Red Hat Managed Openshift service called ARO

Docker SwARM

- Docker SWARM is Docker Inc native Container Orchestration Platform
- it supports managing only Docker containerized application workloads
- it is light weight setup, hence we can install this in a laptop with even a basic configuration
- it is easy to install, learn
- it is not production-grade, generally used for learning purpose, can be used in dev/qa environment

Google Kubernetes

- developed by Google in Golang
- robust container orchestration platform
- supports many different types of container runtimes including runc, containerd, CRI-O, etc.,
- it also supports adding additional functionality by adding your own Custom Resource and custom controllers
- to extend Kubernetes, we can develop an Kubernetes Operator and add new

functionalities

- Operators => is a combination of many Custom Resource + Custom Controllers
- is opensource
- supports command-line interface only
- time tested and robust can be used in Production
- Kubernetes Dashboard (webconsole - it is basic - it poses some security issues, so the first thing administrators does is to disable this)

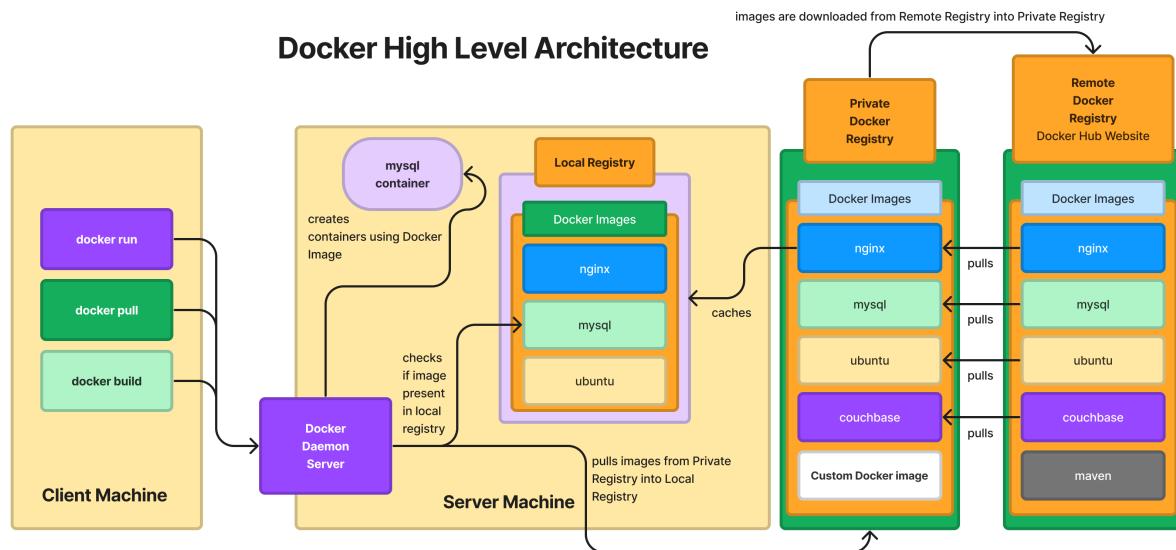
Red Hat OpenShift

- is developed on top of Google Kubernetes
- Red Hat developed many operators and extended Kubernetes to support many practical features required by IT industry
- Red Hat supports
 - User Management (Role Based Access Control - RBAC)
 - Comes with Internal Container Registry
 - Comes with S2I
 - We can deploy application from source code grabbed from Version control softwares (Not supported in K8s)
 - We build and deploy application from within OpenShift
 - CI/CD Platform
 - Supports Virtualization
 - Routes - allows us expose our application with a friendly public url

Red Hat OpenShift - Control Plane Components

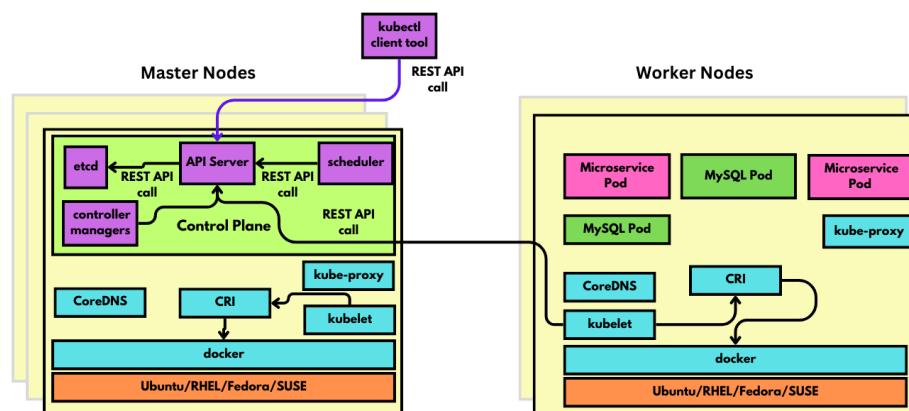
- API Server (Pod)
- etcd key-value data-store (Pod)
- controller managers (Pod)
- scheduler (Pod)

Docker - High Level Architecture

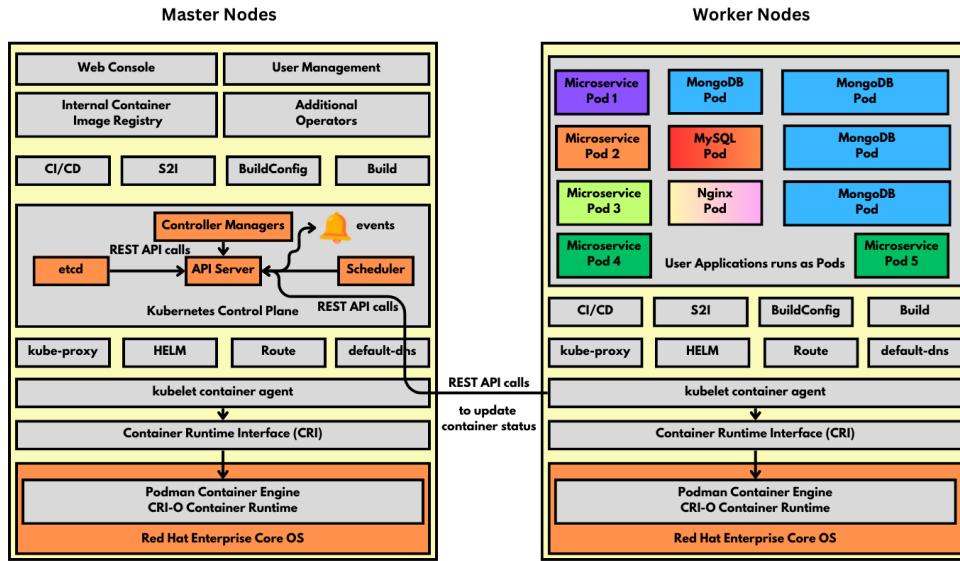


Google Kubernetes - High Level Architecture

Kubernetes High Level Architecture



Red Hat OpenShift - High Level Architecture



API Server

- support REST APIs for every feature supported in Kubernetes/OpenShift
- the entire status and application status is stored in the etcd database by the API Server
- no components in OpenShift talk to each other directly
- API Server is the only component which can update the etcd database
- all OpenShift components they only communicate with API Server via REST calls
- any time the API Server updates the etcd database it will send a broadcasting event
 - Examples
 - new deployment created
 - new replicaset created
 - new pod created
 - scaled up
 - scaled down
 - deployment updated
 - deployment delete

etcd

- key-value database
- this stores the application and cluster status
- if we backup the etcd, it is very easy to restore the same cluster elsewhere
- open source, independent project which can be used outside the scope of Kubernetes/OpenShift

controller managers

- a group of many controller which provides monitoring
- each controller manages one type of resource in Kubernetes/OpenShift
- Deployment Controller manages Deployment resource
- ReplicaSet Controller manages ReplicaSet resource
- Examples
 - Deployment Controller
 - ReplicaSet controller
 - StatefulSet controller
 - Job Controller
 - CronJob Controller
 - Endpoint Controller
 - DaemonSet Controller

scheduler

- this component recommends on which node a newly created Pod can be deployed
- scheduler can't deploy a pod directly, hence it sends its scheduling recommendations to API Server via REST calls

Info - Pod Overview

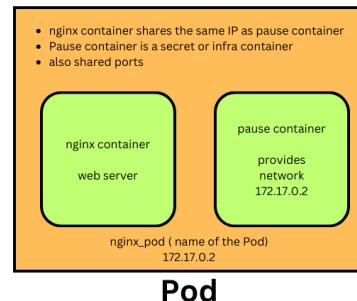
- Pod is a group of related containers
- Pod is the smallest unit that can be deployed within Kubernetes/OpenShift
- Pod is a resource stored and managed within etcd database by API Server
- Pod never runs anywhere, only the container within them runs in worker or master nodes
- One Pod can contain any number of containers
- Best practice is,
 - only one main application should run per Pod
 - one Pod must represent a single application or a single application component (microservice, webserver, app server, db server, etc.,)
- when we deploy our applications into Kubernetes/OpenShift, they run inside a container which is part of a Pod
- In case of Docker, every running container gets an IP address, but in Kubernetes/OpenShift IP address is assigned only on the Pod level not on the container level

Creating a pod with plain Docker

```
docker run -d --name nginx_pause --hostname nginx
gcr.io/google_containers/pause:latest
docker run -d --name nginx --network=container:nginx_pause nginx:latest
docker ps
docker inspect -f {{.NetworkSettings.IPAddress}} nginx_pause
docker exec -it nginx sh
hostname -i
exit
```

In the above, both the nginx_pause container and the nginx containers share the IP address.

Expected output



```
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker run -d --name nginx_pause --hostname nginx gcr.io/google_containers/pause:latest
8376403a0a931ab15c42fa7402480e29740f2d014fd3467a97e78bf79296b089
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
8376403a0a93 gcr.io/google_containers/pause:latest "/pause" 2 seconds ago Up 1 second nginx_pause
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker run -d --name nginx --network=container:nginx_pause nginx:latest
723f602c33d92c3a5c7a8d39b92005fa91d94f7eb9b8ecb5a3e6eccabbda2f5
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
723f602c33d9 nginx:latest "/docker-entrypoint..." 2 seconds ago Up 1 second nginx
8376403a0a93 gcr.io/google_containers/pause:latest "/pause" 49 seconds ago Up 48 seconds nginx_pause
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker inspect -f nginx_pause | grep IPA
"docker inspect" requires at least 1 argument.
See 'docker inspect --help'.

Usage: docker inspect [OPTIONS] NAME|ID [NAME|ID...]

Return low-level information on Docker objects
x jegan@tektutor.org ~ -/openshift-june-2024 [main] docker inspect nginx_pause | grep IPA
  "SecondaryIPAddresses": null,
  "IPAddress": "172.17.0.2",
  "IPAMConfig": null,
  "IPAddress": "172.17.0.2",
jegan@tektutor.org ~ -/openshift-june-2024 [main] docker exec -it nginx sh
# hostname -i
172.17.0.2
# exit
jegan@tektutor.org ~ -/openshift-june-2024 [main]
```

Info - Pod and container states

You may refer the official documentation here

<https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/>

Info - What is ReplicaSet ?

- Let's say 1000s of users are trying to access a web hosted in a single Pod
- a single Pod instance won't be able to serve 1000s of users, hence we may need to add more instances of the Pod
- to create and manage multiple Pod instances of a single application, Kubernetes/OpenShift supports something called ReplicaSet
- ReplicaSet is a resource which is stored in etcd database
 - the container image that must be used to create containers under the Pod
 - it has desiredCount - tells how many Pod instances the user expects to run in OpenShift
 - it has currentPod - tells how many Pod instances are actually running in the openshift
 - it has available/ready Pod - tells how many Pod are in ready state to serve end-users
- ReplicaSet is created by Deployment Controller
- ReplicaSet is accepted as an input by the ReplicaSet Controller
- ReplicaSet Controller creates so many Pods as mentioned in the desiredCount of the ReplicaSet
- ReplicaSet Controller is also responsible for scale up/down

Info - What is Deployment?

- stateless application are deployed as Deployment into Kubernetes/Openshift
- When we run the command 'oc create deployment nginx --image=bitnami/nginx --replicas=3' it creates the following in the openshift cluster
 - a Deployment resource
 - a replicaset resource
 - Pod 1
 - pause container (running in worker-1 node)
 - nginx container (running in worker-1 node)
 - Pod 2
 - pause container (running in worker-2 node)
 - nginx container (running in worker-2 node)
 - Pod 3
 - pause container (running in master-2 node)
 - nginx container (running in master-2 node)

Info - openshift project

- Kubernetes supports something called namespace
- Kubernetes/Openshift are generally shared by many teams within the organization
- Using namespace we can segregate the applications deployment by one team from the other teams
- Openshift has introduced a new feature called projects, which is based on namespace
- Using Openshift projects we can give access only the team members while denying access to other users

Info - What is an Openshift Operator?

- Kubernetes/Openshift allows adding new functionalities by creating custom resources
- Custom Resources can be added to Kubernetes/Openshift by creating Custom Resource Definitions (CRD)
- In order to manage our Custom Resources, we also need to add Custom Controllers
- As we know, each Controller manages one type of Resource, hence the already existing controllers will have no knowledge about our Custom Resources, hence we need to provide our own Controller
- The combination of Custom Resources and Custom Controller is referred as Operators
- This is supported in both Kubernetes and Openshift
- Example
 - We installed Metallb operator to support LoadBalancer service in bare-metal openshift cluster

- The metallb operator would have installed many new resources to our cluster
- The metallb operator would have also installed controllers to manage the custom metallb resources

Lab - Listing the Openshift nodes

```
oc get nodes
kubectl get nodes

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

Expected output

```
jegan@tektutor.org ~ /openshift-june-2024 [main] oc get nodes
jegan@tektutor.org ~ /openshift-june-2024 [main] kubectl get nodes
jegan@tektutor.org ~ /openshift-june-2024 [main] kubectl get nodes -o wide
```

NAME	STATUS	ROLES	AGE	VERSION
master-1.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h31m	v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h31m	v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h31m	v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs	Ready	worker	6h13m	v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs	Ready	worker	6h14m	v1.28.10+a2c84a5

NAME	STATUS	ROLES	AGE	VERSION
master-1.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h31m	v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h31m	v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h31m	v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs	Ready	worker	6h13m	v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs	Ready	worker	6h14m	v1.28.10+a2c84a5

NAME	STATUS	ROLES	AGE	VERSION	INTERNAL-IP	EXTERNAL-IP	OS-IMAGE
master-1.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h32m	v1.28.10+a2c84a5	192.168.122.77	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	control-plane,master,worker	5.14.0-284.69.1.el9_2.x86_64	cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9	192.168.122.249	<none>	Red
master-2.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h32m	v1.28.10+a2c84a5	192.168.122.172	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	control-plane,master,worker	5.14.0-284.69.1.el9_2.x86_64	cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9	192.168.122.172	<none>	Red
master-3.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h32m	v1.28.10+a2c84a5	192.168.122.185	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	control-plane,master,worker	5.14.0-284.69.1.el9_2.x86_64	cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9	192.168.122.192	<none>	Red
worker-1.ocp4.tektutor.org.labs	Ready	worker	6h14m	v1.28.10+a2c84a5	192.168.122.185	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	worker	6h15m	v1.28.10+a2c84a5	192.168.122.192	<none>	Red
worker-2.ocp4.tektutor.org.labs	Ready	worker	6h15m	v1.28.10+a2c84a5	192.168.122.192	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	worker	6h15m	v1.28.10+a2c84a5	192.168.122.192	<none>	Red

NAME	STATUS	ROLES	AGE	VERSION	INTERNAL-IP	EXTERNAL-IP	OS-IMAGE
master-1.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h32m	v1.28.10+a2c84a5	192.168.122.77	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	control-plane,master,worker	5.14.0-284.69.1.el9_2.x86_64	cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9	192.168.122.249	<none>	Red
master-2.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h32m	v1.28.10+a2c84a5	192.168.122.172	<none>	Red
Hat Enterprise Linux CoreOS 415.92.20240611137-0 (Plow)	Ready	control-plane,master,worker	5.14.0-284.69.1.el9_2.x86_64	cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9	192.168.122.172	<none>	Red
master-3.ocp4.tektutor.org.labs	Ready	control-plane,master,worker	6h32m	v1.28.10+a2c84a5	192.168.122.172	<none>	Red

```
jegan@tektutor.org ~ /openshift-june-2024 $ main $ kubectl get nodes
NAME           STATUS   ROLES      AGE    VERSION
master-1.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h31m v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h31m v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h31m v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs Ready    worker      6h13m v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs Ready    worker      6h14m v1.28.10+a2c84a5
jegan@tektutor.org ~ /openshift-june-2024 $ main $ kubectl get nodes -o wide
NAME           STATUS   ROLES      AGE    VERSION   INTERNAL-IP          EXTERNAL-IP  OS-I
MAGE
master-1.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h32m v1.28.10+a2c84a5 192.168.122.77 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
master-2.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h32m v1.28.10+a2c84a5 192.168.122.249 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
master-3.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h32m v1.28.10+a2c84a5 192.168.122.172 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
worker-1.ocp4.tektutor.org.labs Ready    worker      6h14m v1.28.10+a2c84a5 192.168.122.185 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
worker-2.ocp4.tektutor.org.labs Ready    worker      6h15m v1.28.10+a2c84a5 192.168.122.192 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
jegan@tektutor.org ~ /openshift-june-2024 $ main $ oc get nodes -o wide
NAME           STATUS   ROLES      AGE    VERSION   INTERNAL-IP          EXTERNAL-IP  OS-I
MAGE
master-1.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h32m v1.28.10+a2c84a5 192.168.122.77 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
master-2.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h32m v1.28.10+a2c84a5 192.168.122.249 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
master-3.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h32m v1.28.10+a2c84a5 192.168.122.172 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
worker-1.ocp4.tektutor.org.labs Ready    worker      6h15m v1.28.10+a2c84a5 192.168.122.185 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
worker-2.ocp4.tektutor.org.labs Ready    worker      6h15m v1.28.10+a2c84a5 192.168.122.192 <none> Red
Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow) 5.14.0-284.69.1.el9_2.x86_64 cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
jegan@tektutor.org ~ /openshift-june-2024 $ main $
```

Lab - Describe node to find more details about node

```
oc get nodes
oc describe node//master-1.ocp4.tektutor.org.labs
```

Expected output

```
jegan@tektutor.org ~ /openshift-june-2024 $ main $ oc get nodes
NAME           STATUS   ROLES      AGE    VERSION
master-1.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h35m v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h36m v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs Ready    control-plane,master,worker 6h35m v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs Ready    worker      6h18m v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs Ready    worker      6h19m v1.28.10+a2c84a5
jegan@tektutor.org ~ /openshift-june-2024 $ main $ oc describe node/master-1.ocp4.tektutor.org.labs
Name:           master-1.ocp4.tektutor.org.labs
Roles:          control-plane,master,worker
Labels:         beta.kubernetes.io/arch=amd64
                beta.kubernetes.io/os=linux
                kubernetes.io/arch=amd64
                kubernetes.io/hostname=master-1.ocp4.tektutor.org.labs
                kubernetes.io/os=linux
                node-role.kubernetes.io/control-plane=
                node-role.kubernetes.io/master=
                node-role.kubernetes.io/worker=
                node.openshift.io/os_id=rhcos
Annotations:   k8s.ovn.org/host-cidrs: ["192.168.122.77/24"]
                k8s.ovn.org/l3-gateway-config:
                  {"default":{"mode":"shared","interface-id":"br-ex_master-1.ocp4.tektutor.org.labs","mac-address":"52:54:00:57:65:a
a","ip-addresses":["192...
                  k8s.ovn.org/network-ids: {"default":"0"}
                  k8s.ovn.org/node-chassis-id: 0e81aafe-47f2-495b-8d82-4f9265bddc98
                  k8s.ovn.org/node-gateway-router-lrp-ifaddr: {"ipv4":"100.64.0.4/16"}
                  k8s.ovn.org/node-id: 4
                  k8s.ovn.org/node-mgmt-port-mac-address: 2a:35:f0:fc:c4:d0
                  k8s.ovn.org/node-primary-ifaddr: {"ipv4":"192.168.122.77/24"}
                  k8s.ovn.org/node-subnets: {"default":["10.128.0.0/23"]}
                  k8s.ovn.org/node-transit-switch-port-ifaddr: {"ipv4":"100.88.0.4/16"}
                  k8s.ovn.org/remote-zone-migrated: master-1.ocp4.tektutor.org.labs
                  k8s.ovn.org/zone-name: master-1.ocp4.tektutor.org.labs
                  machineconfiguration.openshift.io/controlPlaneTopology: HighlyAvailable
                  machineconfiguration.openshift.io/currentConfig: rendered-master-7b78469312188f8fc026e379fa589ec8
                  machineconfiguration.openshift.io/desiredConfig: rendered-master-7b78469312188f8fc026e379fa589ec8
jegan@tektutor.org ~ /openshift-june-2024 $ main $
```

```

jegan@tektutor.org
openshift-monitoring      87Mi (0%)    0 (0%)   6h22m
openshift-multus          10Mi (0%)   0 (0%)   6h34m
openshift-multus          65Mi (0%)   0 (0%)   6h34m
openshift-multus          120Mi (0%)  0 (0%)   6h34m
openshift-network-diagnostics 40Mi (0%)  0 (0%)   6h33m
openshift-network-diagnostics 15Mi (0%)  0 (0%)   6h33m
openshift-network-node-identity 100Mi (0%) 0 (0%)   6h33m
openshift-oauth-apiserver 200Mi (1%) 0 (0%)   6h20m
openshift-ovn-kubernetes 320Mi (2%) 0 (0%)   6h33m
openshift-ovn-kubernetes 1630Mi (11%) 0 (0%)  6h30m
openshift-route-controller-manager 100Mi (0%) 0 (0%)  6h20m
openshift-service-ca       120Mi (0%) 0 (0%)  6h29m
thanos-querier-779595dbdd-d98ln      15m (0%)  0 (0%) 
multus-additional-cni-plugins-vwptp     10m (0%)  0 (0%) 
multus-tqd7c                      10m (0%)  0 (0%) 
network-metrics-daemon-ckc4q          20m (0%)  0 (0%) 
network-check-source-66bf455fcb-6lnvg 10m (0%)  0 (0%) 
network-check-target-rkq7r            10m (0%)  0 (0%) 
network-node-identity-8h6c8          20m (0%)  0 (0%) 
apiserver-65dbc77c5-flnch           150m (2%) 0 (0%) 
ovnkube-control-plane-5bf6c7f49-lr49h 20m (0%)  0 (0%) 
ovnkube-node-d8j2p                  80m (1%)  0 (0%) 
route-controller-manager-56fb6bc766-wq5gg 100m (1%) 0 (0%) 
service-ca-5b58db65bf-jrxsd8        10m (0%)  0 (0%) 

Allocated resources:
(Total limits may be over 100 percent, i.e., overcommitted.)
Resource      Requests      Limits
-----        -----        -----
cpu           2063m (27%)  0 (0%) 
memory        8790Mi (60%) 0 (0%) 
ephemeral-storage 0 (0%)   0 (0%) 
hugepages-1Gi 0 (0%)   0 (0%) 
hugepages-2Mi 0 (0%)   0 (0%) 

Events: <none>
jegan@tektutor.org ~/openshift-june-2024 ✘ main

```

Lab - Using explain command to know api details and definition of any resource in Openshift/Kubernetes

```

oc explain node
oc explain deployment
oc explain replicaset
oc explain pod

```

Expected output

```

jegan@tektutor.org
hugepages-1Gi      0 (0%)    0 (0%)
hugepages-2Mi      0 (0%)    0 (0%)
Events:           <none>
jegan@tektutor.org ~/openshift-june-2024 ✘ main oc explain node
KIND:             Node
VERSION:          v1

DESCRIPTION:
  Node is a worker node in Kubernetes. Each node will have a unique identifier
  in the cache (i.e. in etcd).

FIELDS:
  apiVersion    <string>
    APIVersion defines the versioned schema of this representation of an object.
    Servers should convert recognized schemas to the latest internal value, and
    may reject unrecognized values. More info:
    https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources

  kind    <string>
    Kind is a string value representing the REST resource this object
    represents. Servers may infer this from the endpoint the client submits
    requests to. Cannot be updated. In CamelCase. More info:
    https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds

  metadata    <ObjectMeta>
    Standard object's metadata. More info:
    https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata

  spec      <NodeSpec>
    Spec defines the behavior of a node.
    https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#spec-and-status

  status     <NodeStatus>
    Most recently observed status of the node. Populated by the system.
    Read-only. More info:
    https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#spec-and-status

```

```
jegan@tektutor.org status <NodeStatus>
Most recently observed status of the node. Populated by the system.
Read-only. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#spec-and-status

jegan@tektutor.org ▶ ~/openshift-june-2024 ▶ main ▶ oc explain deployment
GROUP: apps
KIND: Deployment
VERSION: v1

DESCRIPTION:
Deployment enables declarative updates for Pods and ReplicaSets.

FIELDS:
apiVersion <string>
APIVersion defines the versioned schema of this representation of an object.
Servers should convert recognized schemas to the latest internal value, and
may reject unrecognized values. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources

kind <string>
Kind is a string value representing the REST resource this object
represents. Servers may infer this from the endpoint the client submits
requests to. Cannot be updated. In CamelCase. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds

metadata <ObjectMeta>
Standard object's metadata. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata

spec <DeploymentSpec>
Specification of the desired behavior of the Deployment.

status <DeploymentStatus>
Most recently observed status of the Deployment.
```

Lab - Listing yaml definition of a specific node

```
oc get node/master-1.ocp4.tektutor.org.labs -o yaml
```

Expected output

```
jegan@tektutor.org jegan@tektutor.org
jegan@tektutor.org ▶ ~/openshift-june-2024 ▶ main ▶ oc get nodes
NAME STATUS ROLES AGE VERSION
master-1.ocp4.tektutor.org.labs Ready control-plane,master,worker 6h41m v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs Ready control-plane,master,worker 6h42m v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs Ready control-plane,master,worker 6h41m v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs Ready worker 6h24m v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs Ready worker 6h24m v1.28.10+a2c84a5
jegan@tektutor.org ▶ ~/openshift-june-2024 ▶ main ▶ oc get node/master-1.ocp4.tektutor.org.labs -o yaml
apiVersion: v1
kind: Node
metadata:
  annotations:
    k8s.ovn.org/host-cidrs: '["192.168.122.77/24"]'
    k8s.ovn.org/l3-gateway-config: '{"default":{"mode":"shared","interface-id":"br-ex_master-1.ocp4.tektutor.org.labs","mac-address":"52:54:00:57:65:aa","ip-addresses":["192.168.122.77/24"],"ip-address":"192.168.122.77/24","next-hops":["192.168.122.1"],"next-hop":"192.168.122.1","node-port-enable":"true","vlan-id":0}}'
    k8s.ovn.org/network-ids: '{"default":0}'
    k8s.ovn.org/node-chassis-id: 0e81aa0-47f2-495b-8d82-4f9265bddc98
    k8s.ovn.org/node-gateway-router-lrp-ifaddr: '{"ipv4": "100.64.0.4/16"}'
    k8s.ovn.org/node-id: "4"
    k8s.ovn.org/node-mgmt-port-mac-address: 2a:35:f0:fc:c4:d0
    k8s.ovn.org/node-primary-ifaddr: '{"ipv4": "192.168.122.77/24"}'
    k8s.ovn.org/node-subnets: '{"default": ["10.128.0.0/23"]}'
    k8s.ovn.org/node-transit-switch-port-ifaddr: '{"ipv4": "100.88.0.4/16"}'
    k8s.ovn.org/remote-zone-migrated: master-1.ocp4.tektutor.org.labs
    k8s.ovn.org/zone-name: master-1.ocp4.tektutor.org.labs
  machineconfiguration.openshift.io/controlPlaneTopology: HighlyAvailable
  machineconfiguration.openshift.io/currentConfig: rendered-master-7b78469312188f8fc026e379fa589ec8
  machineconfiguration.openshift.io/desiredConfig: rendered-master-7b78469312188f8fc026e379fa589ec8
  machineconfiguration.openshift.io/desiredDrain: uncordon-rendered-master-7b78469312188f8fc026e379fa589ec8
  machineconfiguration.openshift.io/lastAppliedDrain: uncordon-rendered-master-7b78469312188f8fc026e379fa589ec8
  machineconfiguration.openshift.io/lastSyncedControllerConfigResourceVersion: "34850"
  machineconfiguration.openshift.io/reason: ""
  machineconfiguration.openshift.io/state: Done
  volumes.kubernetes.io/controller-managed-attach-detach: "true"
creationTimestamp: "2024-06-24T02:02:57Z"
```

```
jegan@tektutor.org
jegan@tektutor.org
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:506043f06ceecfa6ea536fa00b4b9a0c458ff803292050381788f882ed4dc519
sizeBytes: 479075651
- names:
- quay.io/openshift-release-dev/ocp-release@sha256:0f55261077557d1bb909c06b115e0c79b0025677be57ba2f045495c11e2443ee
sizeBytes: 478204279
- names:
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:9fb85269a43c579c5be076865a2a14c48affa14fc9d412b384e09c0c5965d2d6
sizeBytes: 469909493
- names:
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:a438c5196a05de97d5ab23a5c253da78dbdb78b063baaa8f710e9b287c722dc7
sizeBytes: 456767509
- names:
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:c2e43004a17edd4c3ac629e96c8a7577cd4ae480031ac2e358cc9f169e67c0f4
sizeBytes: 452296807
- names:
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:a72ccb4a9578ec6fb4e10755ce1e882a0f1e118550b2664a0d957d9aed53a97a
sizeBytes: 447305840
- names:
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:32a95f18b5d680b21af54970294f7d8cf84e5a9e3520b410c71dbc17ad56e4b
sizeBytes: 445675764
- names:
- quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:670e27c06d452137c507cb0b503cc059ad6961b3f79c1425553119101bdc59aa
sizeBytes: 443868274
nodeInfo:
architecture: amd64
bootID: de0cf256-f4f8-456a-83ec-6e229ab1fa65
containerRuntimeVersion: cri-o://1.28.7-3.rhaos4.15.gitf52304f.el9
kernelVersion: 5.14.0-284.69.1.el9_2.x86_64
kubeProxyVersion: v1.28.10+a2c84a5
kubeletVersion: v1.28.10+a2c84a5
machineID: ee397eca09f54a38835e43c86bcf5df5
operatingSystem: linux
osImage: Red Hat Enterprise Linux CoreOS 415.92.202406111137-0 (Plow)
systemUUID: ee397eca-09f5-4a38-835e-43c86bcf5df5
jegan@tektutor.org > ~/openshift-june-2024 > \ main
```

Lab - Listing all project namespaces in openshift

```
oc get projects
oc get namespaces

oc get project
oc get namespace
```

Expected output

```
jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc get projects
NAME          DISPLAY NAME   STATUS
default        Active
kube-node-lease Active
kube-public    Active
kube-system    Active
openshift      Active
openshift-apiserver Active
openshift-apiserver-operator Active
openshift-authentication Active
openshift-authentication-operator Active
openshift-cloud-controller-manager Active
openshift-cloud-controller-manager-operator Active
openshift-cloud-credential-operator Active
openshift-cloud-network-config-controller Active
openshift-cloud-platform-infra Active
openshift-cluster-csi-drivers Active
openshift-cluster-machine approver Active
openshift-cluster-node-tuning-operator Active
openshift-cluster-samples-operator Active
openshift-cluster-storage-operator Active
openshift-cluster-version Active
openshift-config Active
openshift-config-managed Active
openshift-config-operator Active
openshift-console Active
openshift-console-operator Active
openshift-console-user-settings Active
openshift-controller-manager Active
openshift-controller-manager-operator Active
openshift-dns Active
openshift-dns-operator Active
openshift-etcd Active
openshift-etcd-operator Active
openshift-host-network Active
openshift-image-registry Active
```

```
jegan@tektutor.org ~ -/openshift-june-2024 [1] $ main > oc get namespaces
NAME           STATUS   AGE
default        Active   6h53m
kube-node-lease Active   6h53m
kube-public    Active   6h53m
kube-system   Active   6h53m
openshift      Active   6h39m
openshift-apiserver Active   6h41m
openshift-apiserver-operator Active   6h53m
openshift-authentication Active   6h41m
openshift-authentication-operator Active   6h53m
openshift-cloud-controller-manager Active   6h52m
openshift-cloud-controller-manager-operator Active   6h53m
openshift-cloud-credential-operator Active   6h53m
openshift-cloud-network-config-controller Active   6h53m
openshift-cloud-platform-infra Active   6h52m
openshift-cluster-csi-drivers Active   6h53m
openshift-cluster-machine-approver Active   6h53m
openshift-cluster-node-tuning-operator Active   6h53m
openshift-cluster-samples-operator Active   6h52m
openshift-cluster-storage-operator Active   6h53m
openshift-cluster-version Active   6h53m
openshift-config Active   6h52m
openshift-config-managed Active   6h52m
openshift-config-operator Active   6h53m
openshift-console Active   6h35m
openshift-console-operator Active   6h35m
openshift-console-user-settings Active   6h35m
openshift-controller-manager Active   6h12m
```

Lab - Create a new project in your name

```
oc new-project jegan
```

Expected output

```
jegan@tektutor.org ~ -/openshift-june-2024 [1] $ main > oc new-project jegan
Now using project "jegan" on server "https://api.ocp4.tektutor.org.labs: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
```

```
jegan@tektutor.org ~ -/openshift-june-2024 [1] $ main >
```

Lab - Finding the currently active project

```
oc project
```

Expected output

```
jegan@tektutor.org
jegan@tektutor.org
jegan@tektutor.org

openshift-marketplace          Active  6h53m
openshift-monitoring           Active  6h52m
openshift-multus               Active  6h46m
openshift-network-diagnostics Active  6h46m
openshift-network-node-identity Active  6h46m
openshift-network-operator     Active  6h53m
openshift-node                 Active  6h39m
openshift-nutanix-infra        Active  6h52m
openshift-oauth-apiserver      Active  6h41m
openshift-openstack-infra       Active  6h52m
openshift-operator-lifecycle-manager Active  6h52m
openshift-operators            Active  6h52m
openshift-ovirt-infra          Active  6h52m
openshift-ovn-kubernetes       Active  6h46m
openshift-route-controller-manager Active  6h42m
openshift-service-ca            Active  6h42m
openshift-service-ca-operator  Active  6h52m
openshift-user-workload-monitoring Active  6h52m
openshift-vsphere-infra        Active  6h52m

jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc new-project jegan
Now using project "jegan" on server "https://api.ocp4.tektutor.org.labs: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

jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc project default
Now using project "default" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc project jegan
Now using project "jegan" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org > ~/openshift-june-2024 > \ main >
```

Lab - Switching between projects

```
oc project default
oc project jegan
```

Expected output

```
jegan@tektutor.org
jegan@tektutor.org
jegan@tektutor.org

openshift-marketplace          Active  6h53m
openshift-monitoring           Active  6h52m
openshift-multus               Active  6h46m
openshift-network-diagnostics Active  6h46m
openshift-network-node-identity Active  6h46m
openshift-network-operator     Active  6h53m
openshift-node                 Active  6h39m
openshift-nutanix-infra        Active  6h52m
openshift-oauth-apiserver      Active  6h41m
openshift-openstack-infra       Active  6h52m
openshift-operator-lifecycle-manager Active  6h52m
openshift-operators            Active  6h52m
openshift-ovirt-infra          Active  6h52m
openshift-ovn-kubernetes       Active  6h46m
openshift-route-controller-manager Active  6h42m
openshift-service-ca            Active  6h42m
openshift-service-ca-operator  Active  6h52m
openshift-user-workload-monitoring Active  6h52m
openshift-vsphere-infra        Active  6h52m

jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc new-project jegan
Now using project "jegan" on server "https://api.ocp4.tektutor.org.labs: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

jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc project default
Now using project "default" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org > ~/openshift-june-2024 > \ main > oc project jegan
Now using project "jegan" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org > ~/openshift-june-2024 > \ main >
```

Lab - Create your first deployment in imperative style

```
oc project jegan
oc project
oc create deployment nginx --image=nginx:latest --replicas=3
```

Listing the deployments in the active project

```
oc get deployments
oc get deployment
oc get deploy
```

Listing the replicaset in the active project

```
oc get replicaset
oc get replicaset
oc get rs
```

Listing the pods in the active project

```
oc get pods
oc get pod
oc get po
```

Expected output

```
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc project
Using project "jegan" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc create deployment nginx --image=nginx:latest --replicas=3
deployment.apps/nginx created
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get deployments
NAME READY UP-TO-DATE AVAILABLE AGE
nginx 0/3 3 0 5s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get deployment
NAME READY UP-TO-DATE AVAILABLE AGE
nginx 0/3 3 0 9s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get deploy
NAME READY UP-TO-DATE AVAILABLE AGE
nginx 0/3 3 0 11s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get replicaset
NAME DESIRED CURRENT READY AGE
nginx-56fcf95486 3 3 0 31s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get replicaset
NAME DESIRED CURRENT READY AGE
nginx-56fcf95486 3 3 0 33s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get rs
NAME DESIRED CURRENT READY AGE
nginx-56fcf95486 3 3 0 35s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get pods
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 CrashLoopBackOff 1 (17s ago) 38s
nginx-56fcf95486-mpk9t 0/1 CrashLoopBackOff 1 (17s ago) 38s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (13s ago) 38s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get pod
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 Error 2 (20s ago) 41s
nginx-56fcf95486-mpk9t 0/1 Error 2 (20s ago) 41s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (16s ago) 41s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get po
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 Error 2 (22s ago) 43s
nginx-56fcf95486-mpk9t 0/1 Error 2 (22s ago) 43s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (18s ago) 43s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔
```

```
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get deploy
NAME READY UP-TO-DATE AVAILABLE AGE
nginx 0/3 3 0 9s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get replicaset
NAME DESIRED CURRENT READY AGE
nginx-56fcf95486 3 3 0 31s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get replicaset
NAME DESIRED CURRENT READY AGE
nginx-56fcf95486 3 3 0 33s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get rs
NAME DESIRED CURRENT READY AGE
nginx-56fcf95486 3 3 0 35s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get pods
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 CrashLoopBackOff 1 (17s ago) 38s
nginx-56fcf95486-mpk9t 0/1 CrashLoopBackOff 1 (17s ago) 38s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (13s ago) 38s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get pod
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 Error 2 (20s ago) 41s
nginx-56fcf95486-mpk9t 0/1 Error 2 (20s ago) 41s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (16s ago) 41s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get po
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 CrashLoopBackOff 5 (80s ago) 4m59s
nginx-56fcf95486-mpk9t 0/1 CrashLoopBackOff 5 (77s ago) 4m59s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 5 (89s ago) 4m59s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main

```

Lab - Troubleshooting the Pod CrashLoopBackoff issue

Let's try to understand why the Pod is crashing

```
oc logs nginx-56fcf95486-85bzw
```

Expected output

```
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get pod
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 CrashLoopBackOff 1 (17s ago) 38s
nginx-56fcf95486-mpk9t 0/1 CrashLoopBackOff 1 (17s ago) 38s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (13s ago) 38s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get pod
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 Error 2 (20s ago) 41s
nginx-56fcf95486-mpk9t 0/1 Error 2 (20s ago) 41s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (16s ago) 41s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get po
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 Error 2 (22s ago) 43s
nginx-56fcf95486-mpk9t 0/1 Error 2 (22s ago) 43s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 1 (18s ago) 43s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main oc get po
NAME READY STATUS RESTARTS AGE
nginx-56fcf95486-85bzw 0/1 CrashLoopBackOff 5 (80s ago) 4m59s
nginx-56fcf95486-mpk9t 0/1 CrashLoopBackOff 5 (77s ago) 4m59s
nginx-56fcf95486-w29qd 0/1 CrashLoopBackOff 5 (89s ago) 4m59s
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: can not modify /etc/nginx/conf.d/default.conf (read-only file system?)
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/06/24 09:06:22 [warn] 1#1: the "user" directive makes sense only if the master process runs with super-user privileges, ignored in /etc/nginx/nginx.conf:2
nginx: [warn] the "user" directive makes sense only if the master process runs with super-user privileges, ignored in /etc/nginx/nginx.conf:2
2024/06/24 09:06:22 [emerg] 1#1: mkdir() "/var/cache/nginx/client_temp" failed (13: Permission denied)
nginx: [emerg] mkdir() "/var/cache/nginx/client_temp" failed (13: Permission denied)
jegan@tektutor.org ~ -/openshift-june-2024 ↵ main
```

In Kubernetes, the Pod containers can run with/without admin privileges. Hence, the Kubernetes cluster will not enforce best practices.

In Openshift, the openshift nodes are installed with Red Hat Enterprise Core OS (RHCOS). The RHCOS operating system enforces the best practices are always followed, when the container images doesn't the best practices it won't let the Pod run due to the violations.

In Openshift regular application is supposed to run with non-adminstrator privileges. Hence, not all folders are writable. In this the docker image that we used is not prepared keep openshift guidelines in mind. Hence, not all docker images will work in openshift, but the same docker image will work perfectly fine in Kubernetes.

Lab - Deleting a deployment from your project

```
oc delete deploy/nginx
oc get deploy,rs,po
```

Expected output

```
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ oc delete deploy/nginx
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ oc get deploy,rs,po
NAME      READY   STATUS    RESTARTS   AGE
nginx-56fcf95486-85bzw  0/1     Error    2 (20s ago)  41s
nginx-56fcf95486-mpk9t  0/1     Error    2 (20s ago)  41s
nginx-56fcf95486-w29qd  0/1     CrashLoopBackOff 1 (16s ago)  41s
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ oc get po
NAME      READY   STATUS    RESTARTS   AGE
nginx-56fcf95486-85bzw  0/1     Error    2 (22s ago)  43s
nginx-56fcf95486-mpk9t  0/1     Error    2 (22s ago)  43s
nginx-56fcf95486-w29qd  0/1     CrashLoopBackOff 1 (18s ago)  43s
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ oc logs nginx-56fcf95486-85bzw
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: can not modify /etc/nginx/conf.d/default.conf (read-only file system?)
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/06/24 09:06:22 [warn] 1#1: the "user" directive makes sense only if the master process runs with super-user privileges, ignored in /etc/nginx/nginx.conf:2
nginx: [warn] the "user" directive makes sense only if the master process runs with super-user privileges, ignored in /etc/nginx/nginx.conf:2
2024/06/24 09:06:22 [emerg] 1#1: mkdir() "/var/cache/nginx/client_temp" failed (13: Permission denied)
nginx: [emerg] mkdir() "/var/cache/nginx/client_temp" failed (13: Permission denied)
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ cat /etc/redhat-release
Red Hat Enterprise Linux release 9.4 (Plow)
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ oc delete deploy/nginx
deployment.apps "nginx" deleted
jegan@tektutor.org ~ -/openshift-june-2024 [main] $ oc get deploy,rs,po -n jegan
No resources found in jegan namespace.
jegan@tektutor.org ~ -/openshift-june-2024 [main] $
```

Lab - Creating a nginx deployment using bitnami/nginx container image from Docker Hub Remote Registry

```
oc project jegan
oc create deployment nginx --image=bitnami/nginx:1.18 --replicas=3
oc get deploy,rs,po
oc get po -w
```

Listing the pod with their IP and node where they are deployed

```
oc get po -o wide
```

Expected output

```
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc project
Using project "jegan" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc create deployment nginx --image=bitnami/nginx:1.18 --replicas=3
deployment.apps/nginx created
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc get deploy,rs,po
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   0/3       3           0           5s

NAME          DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-566b5879cb   3         3           0           5s

NAME          READY   STATUS      RESTARTS   AGE
pod/nginx-566b5879cb-cqcsz   0/1     ContainerCreating   0           5s
pod/nginx-566b5879cb-kj8rd   0/1     ContainerCreating   0           5s
pod/nginx-566b5879cb-r759k   0/1     ContainerCreating   0           5s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc get po -w
NAME          READY   STATUS      RESTARTS   AGE
nginx-566b5879cb-cqcsz   0/1     ContainerCreating   0           10s
nginx-566b5879cb-kj8rd   0/1     ContainerCreating   0           10s
nginx-566b5879cb-r759k   0/1     ContainerCreating   0           10s
nginx-566b5879cb-kj8rd   1/1     Running      0           13s
nginx-566b5879cb-cqcsz   1/1     Running      0           13s
nginx-566b5879cb-r759k   1/1     Running      0           13s
^C
x jegan@tektutor.org ~ /openshift x jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤
x jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤
```

```
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc project
Using project "jegan" on server "https://api.ocp4.tektutor.org.labs:6443".
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc get all
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc create deployment nginx --image=bitnami/nginx:1.18 --replicas=3
deployment.apps/nginx created
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc get deploy,rs,po
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   0/3       3           0           5s

NAME          DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-566b5879cb   3         3           0           5s

NAME          READY   STATUS      RESTARTS   AGE
pod/nginx-566b5879cb-cqcsz   0/1     ContainerCreating   0           5s
pod/nginx-566b5879cb-kj8rd   0/1     ContainerCreating   0           5s
pod/nginx-566b5879cb-r759k   0/1     ContainerCreating   0           5s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤ oc get po -w
NAME          READY   STATUS      RESTARTS   AGE
nginx-566b5879cb-cqcsz   0/1     ContainerCreating   0           10s
nginx-566b5879cb-kj8rd   0/1     ContainerCreating   0           10s
nginx-566b5879cb-r759k   0/1     ContainerCreating   0           10s
nginx-566b5879cb-kj8rd   1/1     Running      0           13s
nginx-566b5879cb-cqcsz   1/1     Running      0           13s
nginx-566b5879cb-r759k   1/1     Running      0           13s
^C
x jegan@tektutor.org ~ /openshift x jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤
oc get po -w
NAME          READY   STATUS      RESTARTS   AGE
nginx-566b5879cb-cqcsz   1/1     Running      0           3m28s
nginx-566b5879cb-kj8rd   1/1     Running      0           3m28s
nginx-566b5879cb-r759k   1/1     Running      0           3m28s
^C
openshift-june-2024 ✘ main ➤ oc get po -o wide
NAME          READY   STATUS      RESTARTS   AGE   IP           NODE   NOMINATED NODE   READINESS GATES
nginx-566b5879cb-cqcsz   1/1     Running      0           3m34s   10.130.0.210   master-3.ocp4.tektutor.org.labs   <none>   <none>
nginx-566b5879cb-kj8rd   1/1     Running      0           3m34s   10.128.2.8    worker-1.ocp4.tektutor.org.labs   <none>   <none>
nginx-566b5879cb-r759k   1/1     Running      0           3m34s   10.131.0.35   worker-2.ocp4.tektutor.org.labs   <none>   <none>
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➤
```

Lab - Getting inside the nodes where your application pod is running to understand some internal stuffs

```
oc get nodes
oc get po -o wide
oc debug node/master-3.ocp4.tektutor.org.labs
```

```
chroot /host
```

```
podman version
crictl version
```

List all container images present in the master 3 node

```
crictl images
```

List all containers running in the master node

```
crictl ps
```

List all containers that belong to a specific pod

```
crictl ps | grep nginx-566b5879cb-cqcsz
```

Expected output

```
jegan@tektutor.org ~ /openshift-june-2024 ✘ main [1] oc get nodes
NAME           STATUS   ROLES          AGE    VERSION
master-1.ocp4.tektutor.org.labs   Ready    control-plane,master,worker   7h32m   v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs   Ready    control-plane,master,worker   7h32m   v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs   Ready    control-plane,master,worker   7h32m   v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs  Ready    worker            7h15m   v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs  Ready    worker            7h15m   v1.28.10+a2c84a5
jegan@tektutor.org ~ /openshift-june-2024 ✘ main [2] oc debug node/master-3.ocp4.tektutor.org.labs
Temporary namespace openshift-debug-mtb74 is created for debugging node...
Starting pod/master-3.ocp4.tektutor.org.labs-debug-lj759 ...
To use host binaries, run `chroot /host`
Pod IP: 192.168.122.172
If you don't see a command prompt, try pressing enter.
sh-4.4# ls
bin boot dev etc home host lib lib64 lost+found media mnt opt proc root run sbin srv sys tmp usr var
sh-4.4# chroot /host
sh-5.1# ls
bin boot dev etc home lib lib64 media mnt opt ostree proc root run sbin srv sys sysroot tmp usr var
sh-5.1# podman version
Client: Podman Engine
Version: 4.4.1
API Version: 4.4.1
Go Version: go1.20.12
Built: Tue Jun 11 06:12:07 2024
OS/Arch: linux/amd64
sh-5.1# crictl version
Version: 0.1.0
RuntimeName: cri-o
RuntimeVersion: 1.28.7-3.rhaos4.15.gitf52304f.el9
RuntimeApiVersion: v1
sh-5.1# crictl images
IMAGE                      TAG        IMAGE ID      SIZE
docker.io/bitnami/nginx    1.18       fa68d866b7140  93.4MB
docker.io/library/nginx     latest     e0c9858e10ed8  192MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      0f6ba8127bcc2  491MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      ffa520813a176  501MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      0b7c0e00cb30b  866MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      85194d67d914c  503MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      2cfde0faf03b9  497MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      809a5dc3fd6ac  706MB
quay.io/openshift-release-dev/ocp-v4.0-art-dev <none>      cb923212cb1d6  538MB
quay.io/openshift-release-dev/ocn-v4.0-art-dev <none>      f5hfq0ar76fqe   814MB
```

jegan@tektutor.org					
sh-5.1# crictl ps		IMAGE	STATE	NAME	CREATED
875caa89d9867	Running	quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:829c1c6b7a2d0053f1cc07b1d8f001cd4198ff8bb221f914d799453d588a7ff8	0	3a7264b8816cc	About a minute ago
19a16a83bd369	Running	quay.io/redhat/community-operator-index	v4.15	f9d0afafe294e	1.41GB
3f17fd66a2f21	Running	quay.io/redhat/redhat-marketplace-index	v4.15	7bdf138932a8d	1.02GB
5e27c27c4fc1f	Running	quay.io/redhat/redhat-operator-index	v4.15	8f7ca7cb159f	2.196GB
61aa64c115ad3	Running	quay.io/redhat/redhat-operator-index	<none>	bf34dfaecd736	2.196GB
fc3e2dcfa3566	Running	quay.io/redhat/redhat-operator-index	<none>	8b4b872dc0467	2.196GB
b0e3a4c1b2b0c	Running	etcd-readyz	0	b3b0ff0713b91	etcd-master-3.ocp4.tektutor.org.labs
bc00f1d8fc468	Running	etcd-metrics	0	b3b0ff0713b91	etcd-master-3.ocp4.tektutor.org.labs
e2f57d62b4e5	Running	etcd	0	b3b0ff0713b91	etcd-master-3.ocp4.tektutor.org.labs
173a560126cff	Running	etcdctl	0	b3b0ff0713b91	etcd-master-3.ocp4.tektutor.org.labs
rg.labs	Running	kube-api-server-check-endpoints	0	36996412eec25	kube-api-server-master-3.ocp4.tektutor.org.labs
e8fa039b78ede	Running	kube-api-server-insecure-readyz	0	36996412eec25	kube-api-server-master-3.ocp4.tektutor.org.labs
a988980a5e72c	Running	kube-api-server-cert-regeneration-controller	0	36996412eec25	kube-api-server-master-3.ocp4.tektutor.org.labs
rr.labs	Running	kube-api-server-cert-syncer	0	36996412eec25	kube-api-server-master-3.ocp4.tektutor.org.labs
3691f1218eed83efde6834db21e46bciae267a46bcfb6f842d130cb82b182c4e7	Running	kube-controller-manager-recovery-controller	0	2cea2bc940b4b	kube-controller-manager-master-3.ocp4.tektutor.org.labs
3691f1218eed83efde6834db21e46bciae267a46bcfb6f842d130cb82b182c4e7	Running	kube-controller-manager-cert-syncer	0	2cea2bc940b4b	kube-controller-manager-master-3.ocp4.tektutor.org.labs
0f6ba8127bcc2548f3085f700427940d45a65c41b86c7732f61f716678bab765	Running	cluster-policy-controller	0	2cea2bc940b4b	kube-controller-manager-master-3.ocp4.tektutor.org.labs

jegan@tektutor.org					
sh-5.1# crictl pods					
POD ID	CREATED	STATE	RUNTIME	NAME	NAMESPACE
9559f3b0f2ecf	2 minutes ago	Ready	(default)	master-3.ocp4.tektutor.org.labs-debug-lj759	openshift-debug-mtb74
1f39b4b663aa5	12 minutes ago	Ready	(default)	nginx-566b5879cb-cqcsz	jegan
010fc3bdd9cae	2 hours ago	Ready	(default)	redhat-operators-528m2	openshift-marketplace
b3b0ff0713b91	7 hours ago	Ready	(default)	etcd-master-3.ocp4.tektutor.org.labs	openshift-etcd
8b7a16a4fd160	7 hours ago	NotReady	(default)	revision-pruner-7-master-3.ocp4.tektutor.org.labs	openshift-kube-apiserver
e2bce02fc028	7 hours ago	NotReady	(default)	installer-7-master-3.ocp4.tektutor.org.labs	openshift-etcd
36996412eec25	7 hours ago	Ready	(default)	kube-apiserver-master-3.ocp4.tektutor.org.labs	openshift-kube-apiserver
ec8ba6bf3b62e	7 hours ago	NotReady	(default)	installer-7-master-3.ocp4.tektutor.org.labs	openshift-kube-apiserver
69db8a40501de	7 hours ago	NotReady	(default)	revision-pruner-7-master-3.ocp4.tektutor.org.labs	openshift-etcd
c44234e243bef	7 hours ago	NotReady	(default)	revision-pruner-6-master-3.ocp4.tektutor.org.labs	openshift-etcd
b46d8b9f56c5d	7 hours ago	NotReady	(default)	revision-pruner-7-master-3.ocp4.tektutor.org.labs	openshift-kube-scheduler
5781d1915458e	7 hours ago	NotReady	(default)	installer-6-master-3.ocp4.tektutor.org.labs	openshift-etcd
b6df8ebf1a646	7 hours ago	NotReady	(default)	revision-pruner-7-master-3.ocp4.tektutor.org.labs	openshift-kube-controller-manager
2cea2bc940b4b	7 hours ago	Ready	(default)	kube-controller-manager-master-3.ocp4.tektutor.org.labs	openshift-kube-controller-manager

```

jegan@tektutor.org
0cd809bbc76d9    7 hours ago      Ready      authentication-operator-7946d4f4f7-k7cj4
c448ba745236c    7 hours ago      Ready      service-ca-operator-56b8b84899-7dvm9
336281ed5ebd2    7 hours ago      Ready      kube-storage-version-migrator-operator-5bdd8f66f6-jj584
grator-operator   0              (default)
2ee4928d4423f    7 hours ago      Ready      openshift-kube-scheduler-operator-5f4cfb774d-nqsbl
47ec2b760b74a    7 hours ago      Ready      insights-operator-75b84ccf57-6rxld
dae64f2e9c2d1    7 hours ago      Ready      openshift-apiserver-operator-767dc9bc97-hdg9d
b8434c5ad9e8d    8 hours ago      Ready      ovnkube-node-58kkk
47c2ed7e1e45a    8 hours ago      Ready      network-node-identity-cg8mh
6b11ad63a02c3    8 hours ago      Ready      multus-additional-cni-plugins-kzpz8
14fd5f99e368f    8 hours ago      Ready      multus-nphnh
dd7370c1251ab    8 hours ago      Ready      network-operator-659bb9fc6f-9kr2g
286dcc82e612a    8 hours ago      Ready      kube-rbac-proxy-crio-master-3.ocp4.tektutor.org.labs
sh-5.1# curlctl ps | grep nginx
19a16a83bd369   docker.io/bitnami/nginx@sha256:6b998c3cla9e411346cff07fe48bcade4ab4775130da80efc00d223dfe03ce0b
                  Running      nginx           0          1f39b4b663aa5      nginx-566b5879cb-cqcsz
sh-5.1# curlctl pods | grep nginx
1f39b4b663aa5   13 minutes ago  Ready      nginx-566b5879cb-cqcsz
sh-5.1# curlctl ps | grep nginx-566b5879cb-cqcsz
19a16a83bd369   docker.io/bitnami/nginx@sha256:6b998c3cla9e411346cff07fe48bcade4ab4775130da80efc00d223dfe03ce0b
                  Running      nginx           0          1f39b4b663aa5      nginx-566b5879cb-cqcsz
sh-5.1# exit
exit
sh-4.4# exit
exit

Removing debug pod ...
Temporary namespace openshift-debug-mtb74 was removed.
jegan@tektutor.org ~/openshift-june-2024 ↵ main

```

Lab - Getting inside a Pod shell

```

oc get deploy,po
oc rsh deploy/nginx

oc exec -it nginx-566b5879cb-kj8rd sh
hostname
hostname -i
exit

```

Expected output

```

jegan@tektutor.org ~/openshift-june-2024 ↵ main
jegan@tektutor.org ~/openshift-june-2024 ↵ main oc get po
NAME        READY  STATUS    RESTARTS  AGE
nginx-566b5879cb-cqcsz  1/1    Running   0          23m
nginx-566b5879cb-kj8rd  1/1    Running   0          23m
nginx-566b5879cb-r759k  1/1    Running   0          23m
jegan@tektutor.org ~/openshift-june-2024 ↵ main oc rsh deploy/nginx
$ hostname
nginx-566b5879cb-cqcsz
$ hostname -i
10.130.0.210
$ ls
50x.html  index.html
$ pwd
/app
$ cd ..
$ ls
app  bitnami  certs  etc  lib  media  opt  root  sbin  sys  usr
bin  boot    dev    home  lib64 mnt   proc  run   srv  tmp  var
$ exit
jegan@tektutor.org ~/openshift-june-2024 ↵ main

```

```

jegan@tektutor.org
NAME          READY   STATUS    RESTARTS   AGE
nginx-566b5879cb-cqcsz  1/1     Running   0          23m
nginx-566b5879cb-kj8rd  1/1     Running   0          23m
nginx-566b5879cb-r759k  1/1     Running   0          23m

jegan@tektutor.org > ~/openshift-june-2024 > } main > oc rsh deploy/nginx
$ hostname
nginx-566b5879cb-cqcsz
$ hostname -i
10.130.0.210
$ ls
50x.html index.html
$ pwd
/app
$ cd ..
$ ls
app  bitnami  certs  etc  lib  media  opt  root  sbin  sys  usr
bin  boot    dev    home  lib64 mnt  proc  run  srv  tmp  var
$ exit
jegan@tektutor.org > ~/openshift-june-2024 > } main > oc exec -it nginx-566b5879cb-kj8rd sh
Kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
$ pwd
/app
$ hostname
nginx-566b5879cb-kj8rd
$ hostname -i
10.128.2.8
$ exit
jegan@tektutor.org > ~/openshift-june-2024 > } main >

```

Lab - Developer testing using port-forward (not recommended in production)

The below command is a blocking command, hence to access the web page you need to open another terminal tab

```

oc get po
oc port-forward nginx-566b5879cb-cqcsz 9090:8080

```

In the above command, port 9090 is open on the local machine, while 8080 is the Pod container port where Nginx is listening. So when we access <http://localhost:9090> the call is forwarded to pod container port 8080.

In a different terminal tab, you can try this (this only works on your local linux machine)

```

curl http://127.0.0.1:9090
curl http://localhost:9090

```

To come out of the port-forward, you need to press **Ctrl + C**

Expected output

The screenshot shows two terminal windows side-by-side.

Terminal 1 (Left):

```
jegan@tektutor.org ~ /openshift-june-2024 [main] $ oc get po
NAME           READY   STATUS    RESTARTS   AGE
nginx-566b5879cb-cqcsz  1/1     Running   0          32m
nginx-566b5879cb-kj8rd  1/1     Running   0          32m
nginx-566b5879cb-r759k  1/1     Running   0          32m
jegan@tektutor.org ~ /openshift-june-2024 [main] $ oc get po -o wide
NAME           READY   STATUS    RESTARTS   AGE   IP                NODE      NOMINATED NODE   READINESS GATES
nginx-566b5879cb-cqcsz  1/1     Running   0          32m   10.130.0.210  master-3.ocp4.tektutor.org.labs  <none>
nginx-566b5879cb-kj8rd  1/1     Running   0          32m   10.128.2.8   worker-1.ocp4.tektutor.org.labs  <none>
nginx-566b5879cb-r759k  1/1     Running   0          32m   10.131.0.35  worker-2.ocp4.tektutor.org.labs  <none>
jegan@tektutor.org ~ /openshift-june-2024 [main] $ ping 10.130.0.210
PING 10.130.0.210 (10.130.0.210) 56(84) bytes of data.
^C
--- 10.130.0.210 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4120ms
x jegan@tektutor.org ~ /openshift-june-2024 [main] $ oc port-forward nginx-566b5879cb-cqcsz 9090:8080
Forwarding from 127.0.0.1:9090 -> 8080
Forwarding from [::1]:9090 -> 8080
Handling connection for 9090
```

Terminal 2 (Right):

```
jegan@tektutor.org ~ /openshift-june-2024 [main] $ curl http://127.0.0.1:9090
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>. <br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

Lab - Exposing an application only within the cluster

When to use ClusterIP Internal Service?

- A clusterip service is an internal service
- this type of service can be used in dev, qa and production
- For instance, you have a mysql database deployment with multiple pods that needs to be accessed from some microservice pod, you can expose the mysql as a clusterip service
- Normally for all db deployment, it is exposed as clusterip services as

```
they are only supposed to accessed within openshift cluster not exposed to
the end users
```

To create a clusterip service, we need to have a deployment first

```
oc project jegan
oc get deploy,rs, po

oc expose deploy/nginx --type=ClusterIP --port=8080
```

Listing the service

```
oc get services
oc get service
oc get svc
```

Finding more details about the clusterip service

```
oc describe svc/nginx
```

Expected output

```
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ oc get deploy,rs,po
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ deployment.apps/nginx 3/3 3 3 48m
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ replicaset.apps/nginx-566b5879cb 3 3 3 48m
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ pod/nginx-566b5879cb-cqcsz 1/1 Running 0 48m
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ pod/nginx-566b5879cb-kj8rd 1/1 Running 0 48m
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ pod/nginx-566b5879cb-r759k 1/1 Running 0 48m
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ oc expose deploy/nginx --type=ClusterIP --port=8080
service/nginx exposed
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ oc get services
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ nginx ClusterIP 172.30.226.252 <none> 8080/TCP 3s
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ oc get service
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ nginx ClusterIP 172.30.226.252 <none> 8080/TCP 7s
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ oc get svc
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ nginx ClusterIP 172.30.226.252 <none> 8080/TCP 9s
jegan@tektutor.org ~ -/openshift-june-2024 └─ main ┤ oc describe svc/nginx
Name:           nginx
Namespace:      jegan
Labels:         app=nginx
Annotations:   <none>
Selector:       app=nginx
Type:          ClusterIP
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             172.30.226.252
IPs:            172.30.226.252
Port:           <unset>  8080/TCP
TargetPort:     8080/TCP
Endpoints:     10.128.2.8:8080,10.130.0.210:8080,10.131.0.35:8080
Session Affinity: None
Events:        <none>
```

```
jegan@tektutor.org ~ -/openshift-june-2024 [main] oc expose deploy/nginx --type=ClusterIP --port=8080
service/nginx exposed
jegan@tektutor.org ~ -/openshift-june-2024 [main] oc get services
NAME      TYPE        CLUSTER-IP      EXTERNAL-IP    PORT(S)      AGE
nginx   ClusterIP   172.30.226.252 <none>        8080/TCP    3s
jegan@tektutor.org ~ -/openshift-june-2024 [main] oc get service
NAME      TYPE        CLUSTER-IP      EXTERNAL-IP    PORT(S)      AGE
nginx   ClusterIP   172.30.226.252 <none>        8080/TCP    7s
jegan@tektutor.org ~ -/openshift-june-2024 [main] oc get svc
NAME      TYPE        CLUSTER-IP      EXTERNAL-IP    PORT(S)      AGE
nginx   ClusterIP   172.30.226.252 <none>        8080/TCP    9s
jegan@tektutor.org ~ -/openshift-june-2024 [main] oc describe svc/nginx
Name:            nginx
Namespace:       jegan
Labels:          app:nginx
Annotations:     <none>
Selector:        app:nginx
Type:            ClusterIP
IP Family Policy: SingleStack
IP Families:    IPv4
IP:              172.30.226.252
IPs:             172.30.226.252
Port:            <unset>: 8080/TCP
TargetPort:      8080/TCP
Endpoints:      10.128.2.8:8080,10.130.0.210:8080,10.131.0.35:8080
Session Affinity: None
Events:          <none>
jegan@tektutor.org ~ -/openshift-june-2024 [main] oc get po -o wide
NAME      READY  STATUS    RESTARTS   AGE      IP           NODE      NOMINATED NODE  READINESS GATES
nginx-566b5879cb-cqcsz  1/1    Running   0          51m     10.130.0.210  master-3.ocp4.tektutor.org.labs  <none>        <none>
nginx-566b5879cb-kj8rd  1/1    Running   0          51m     10.128.2.8   worker-1.ocp4.tektutor.org.labs  <none>        <none>
nginx-566b5879cb-r759k  1/1    Running   0          51m     10.131.0.35  worker-2.ocp4.tektutor.org.labs  <none>        <none>
```

Accessing the clusterip service from a Pod that belongs to a different deployment

```
oc create deployment hello --image=tektutor/spring-ms:1.0
oc get po -w
```

Get inside the hello pod shell

```
oc rsh deploy/hello
curl http://nginx:8080
curl http://172.30.226.252:8080
```

Expected output

The screenshot shows two terminal windows side-by-side. Both windows have a title bar 'jegan@tektutor.org'.

Terminal 1 (Left):

```
jegan@tektutor.org ~/openshift-june-2024 [main] oc get deploy
NAME READY UP-TO-DATE AVAILABLE AGE
hello 0/1 1 0 5s
nginx 3/3 3 3 59m
jegan@tektutor.org ~/openshift-june-2024 [main] oc get po -w
NAME READY STATUS RESTARTS AGE
hello-56d67b65ff-pmvt2 0/1 ContainerCreating 0 9s
nginx-566b5879cb-cqcsz 1/1 Running 0 59m
nginx-566b5879cb-kj8rd 1/1 Running 0 59m
nginx-566b5879cb-r759k 1/1 Running 0 59m
hello-56d67b65ff-pmvt2 1/1 Running 0 22s
^C%
```

Terminal 2 (Right):

```
x jegan@tektutor.org ~/openshift-june-2024 [main] oc rsh deploy/hello
sh-4.4$ ls
app.jar bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
sh-4.4$ curl http://nginx:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
```

The right terminal continues:

```
</body>
</html>
sh-4.4$ curl http://172.30.226.252:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
sh-4.4$
```

Info - What is service discovery in Kubernetes/OpenShift?

- accessing a service by its name
- In Kubernetes/OpenShift, we have DNS Pods, whose job is to translate the service name into its respective service IP address
- In each pod, there will be file called /etc/resolv.conf
- When the pod containers are created by kubelet container agent running on every node, it also configures the /etc/resolv.conf with the openshift dns service IP, which helps in resolving the service name to its respective IP address

- the nameserver IP address we see 172.30.0.10 is the dns-default service IP, which represents a group of dns pods(one pod per node)

Finding the dns service IP address

```
oc get svc -n openshift-dns
```

Expected output

```
jegan@tektutor.org
jegan@tektutor.org
jegan@tektutor.org

<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
sh-4.4$ cat /etc/resolv.conf
search jegan.svc.cluster.local svc.cluster.local cluster.local ocp4.tektutor.org.labs
nameserver 172.30.0.10
options ndots:5
sh-4.4$
```

```
jegan@tektutor.org
jegan@tektutor.org
jegan@tektutor.org

options ndots:5
sh-4.4$ exit
exit
jegan@tektutor.org > ~/openshift-june-2024 > | main | oc get po
NAME          READY   STATUS    RESTARTS   AGE
hello-56d67b65ff-pmv2  1/1     Running   0          15m
nginx-566b5879cb-cqcsz  1/1     Running   0          75m
nginx-566b5879cb-kj8rd  1/1     Running   0          75m
nginx-566b5879cb-r759k  1/1     Running   0          75m
jegan@tektutor.org > ~/openshift-june-2024 > | main | oc describe pod/nginx-566b5879cb-cqcsz
Name:           nginx-566b5879cb-cqcsz
Namespace:      jegan
Priority:       0
Service Account: default
Node:          master-3.ocp4.tektutor.org.labs/192.168.122.172
Start Time:    Mon, 24 Jun 2024 14:56:10 +0530
Labels:         app=nginx
                pod-template-hash=566b5879cb
Annotations:   k8s.v1.cni.cncf.io/network-status:
                {"default":{"ip_addresses":["10.130.0.210/23"],"mac_address":"0a:58:0a:82:00:d2","gateway_ips":["10.130.0.1"]}, "route": [{"dest": "10.128.0..."}], "k8s.v1.cni.cncf.io/network-status": [{"name": "ovn-kubernetes", "interface": "eth0", "ips": ["10.130.0.210"], "mac": "0a:58:0a:82:00:d2", "default": true, "dns": {}}], "openshift.io/scc": "restricted-v2", "seccomp.security.alpha.kubernetes.io/pod": "runtime/default", "Status": "Running", "seccompProfile": "RuntimeDefault+"
Status:          Running
```

```
jegan@tektutor.org
Started: Mon, 24 Jun 2024 14:56:22 +0530
Ready: True
Restart Count: 0
Environment: <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-c85x8 (ro)
Conditions:
  Type      Status
  Initialized  True
  Ready       True
  ContainersReady  True
  PodScheduled  True
Volumes:
  kube-api-access-c85x8:
    Type:        Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:  kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI:   true
    ConfigMapName:  openshift-service-ca.crt
    ConfigMapOptional: <nil>
  QoS Class:  BestEffort
  Node-Selectors: <none>
  Tolerations:  node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                 node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type  Reason     Age   From            Message
  ----  ----      --   --              --
  Normal Scheduled  75m  default-scheduler  Successfully assigned jegan/nginx-566b5879cb-cqcsz to master-3.ocp4.tektutor.org.labs
  Normal AddedInterface  75m  multus          Add eth0 [10.130.0.210/23] from ovn-kubernetes
  Normal Pulling   75m  kubelet         Pulling image "bitnami/nginx:1.18"
  Normal Pulled    75m  kubelet         Successfully pulled image "bitnami/nginx:1.18" in 11.38s (11.38s including waiting)
  Normal Created   75m  kubelet         Created container nginx
  Normal Started   75m  kubelet         Started container nginx
jegan@tektutor.org ~ /openshift-june-2024 ↵ main ↴

jegan@tektutor.org
Environment: <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-c85x8 (ro)
Conditions:
  Type      Status
  Initialized  True
  Ready       True
  ContainersReady  True
  PodScheduled  True
Volumes:
  kube-api-access-c85x8:
    Type:        Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:  kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI:   true
    ConfigMapName:  openshift-service-ca.crt
    ConfigMapOptional: <nil>
  QoS Class:  BestEffort
  Node-Selectors: <none>
  Tolerations:  node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                 node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type  Reason     Age   From            Message
  ----  ----      --   --              --
  Normal Scheduled  75m  default-scheduler  Successfully assigned jegan/nginx-566b5879cb-cqcsz to master-3.ocp4.tektutor.org.labs
  Normal AddedInterface  75m  multus          Add eth0 [10.130.0.210/23] from ovn-kubernetes
  Normal Pulling   75m  kubelet         Pulling image "bitnami/nginx:1.18"
  Normal Pulled    75m  kubelet         Successfully pulled image "bitnami/nginx:1.18" in 11.38s (11.38s including waiting)
  Normal Created   75m  kubelet         Created container nginx
  Normal Started   75m  kubelet         Started container nginx
jegan@tektutor.org ~ /openshift-june-2024 ↵ main ↴ oc get svc -n openshift-dns
NAME        TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE
dns-default  ClusterIP  172.30.0.10  <none>           53/UDP,53/TCP,9154/TCP  8h
jegan@tektutor.org ~ /openshift-june-2024 ↵ main ↴
```

Lab - Creating a nodeport external service for nginx deployment in imperative style

First, let's delete the existing clusterip internal service

```
oc get svc
oc delete svc/nginx
oc get svc
```

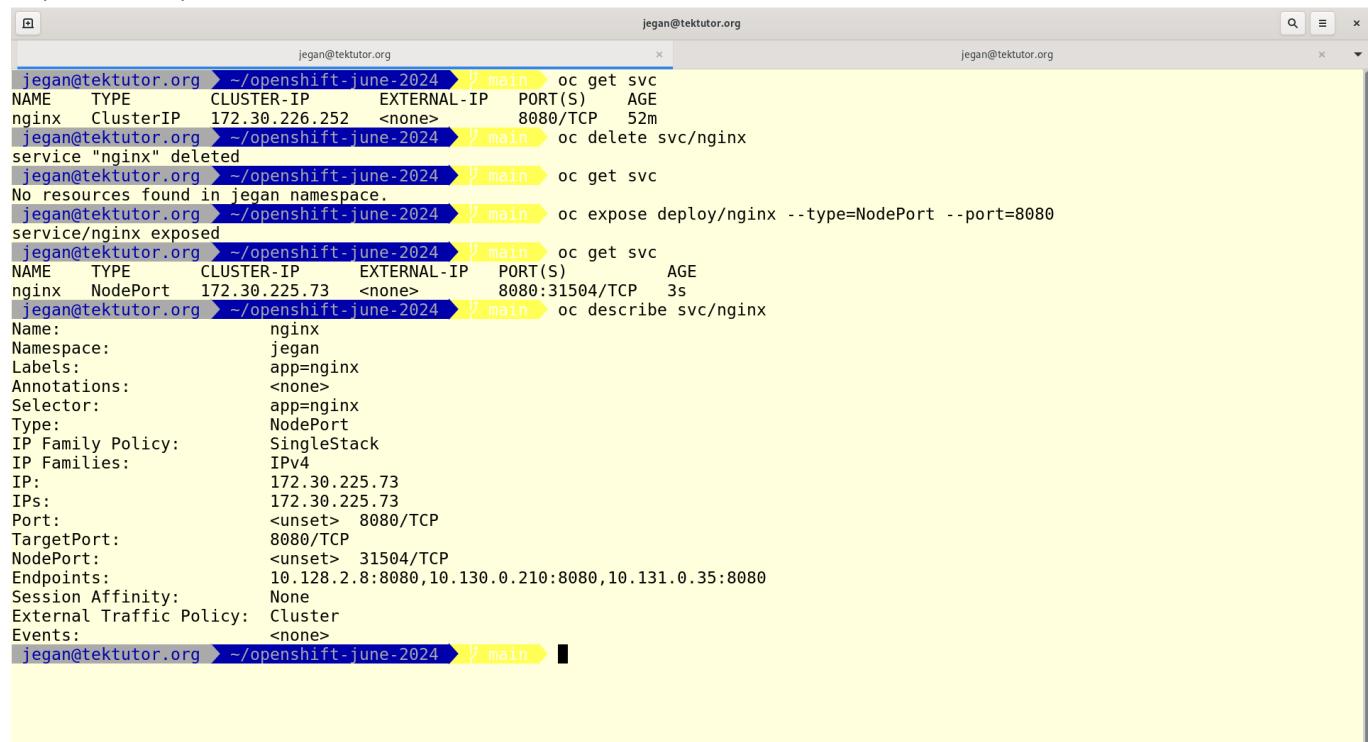
Let's create the nodeport external service

```
oc expose deploy/nginx --type=NodePort --port=8080
oc get svc
oc describe svc/nginx
```

Let's access the nodeport service

```
oc get nodes
curl http://master-1.ocp4.tektutor.org.labs:31504
curl http://master-2.ocp4.tektutor.org.labs:31504
curl http://master-3.ocp4.tektutor.org.labs:31504
curl http://worker-1.ocp4.tektutor.org.labs:31504
curl http://worker-2.ocp4.tektutor.org.labs:31504
```

Expected output



The screenshot shows a terminal window with two tabs. The left tab is titled 'jegan@tektutor.org' and contains the command history for creating and exposing the service. The right tab is also titled 'jegan@tektutor.org'. The terminal output is as follows:

```
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get svc
NAME      TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
nginx    ClusterIP   172.30.226.252 <none>        8080/TCP   52m
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc delete svc/nginx
service "nginx" deleted
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get svc
No resources found in jegan namespace.
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc expose deploy/nginx --type=NodePort --port=8080
service/nginx exposed
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc get svc
NAME      TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
nginx    NodePort    172.30.225.73  <none>        8080:31504/TCP 3s
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ oc describe svc/nginx
Name:           nginx
Namespace:      jegan
Labels:          app=nginx
Annotations:    <none>
Selector:       app=nginx
Type:           NodePort
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             172.30.225.73
IPs:            172.30.225.73
Port:           <unset>  8080/TCP
TargetPort:     8080/TCP
NodePort:       <unset>  31504/TCP
Endpoints:      10.128.2.8:8080,10.130.0.210:8080,10.131.0.35:8080
Session Affinity: None
External Traffic Policy: Cluster
Events:         <none>
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔
```

```
jegan@tektutor.org
Annotations: <none>
Selector: app=nginx
Type: NodePort
IP Family Policy: SingleStack
IP Families: IPv4
IP: 172.30.225.73
IPs: 172.30.225.73
Port: <unset> 8080/TCP
TargetPort: 8080/TCP
NodePort: <unset> 31504/TCP
Endpoints: 10.128.2.8:8080,10.130.0.210:8080,10.131.0.35:8080
Session Affinity: None
External Traffic Policy: Cluster
Events: <none>
jegan@tektutor.org ~ -/openshift-june-2024 ⌘ main ➜ oc get nodes
NAME STATUS ROLES AGE VERSION
master-1.ocp4.tektutor.org.labs Ready control-plane,master,worker 9h v1.28.10+a2c84a5
master-2.ocp4.tektutor.org.labs Ready control-plane,master,worker 9h v1.28.10+a2c84a5
master-3.ocp4.tektutor.org.labs Ready control-plane,master,worker 9h v1.28.10+a2c84a5
worker-1.ocp4.tektutor.org.labs Ready worker 8h v1.28.10+a2c84a5
worker-2.ocp4.tektutor.org.labs Ready worker 8h v1.28.10+a2c84a5
jegan@tektutor.org ~ -/openshift-june-2024 ⌘ main ➜ curl http://master-1.ocp4.tektutor.org.labs:31504
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org ~ -/openshift-june-2024 ⌘ main ➜ curl http://master-2.ocp4.tektutor.org.labs:31504
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org ~ -/openshift-june-2024 ⌘ main ➜
```

```
jegan@tektutor.org
jegan@tektutor.org

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org > ~/openshift-june-2024 > ⌂ main curl http://master-3.ocp4.tektutor.org.labs:31504
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org > ~/openshift-june-2024 > ⌂ main
```

```
jegan@tektutor.org
jegan@tektutor.org

</body>
</html>
jegan@tektutor.org > ~/openshift-june-2024 > ⌂ main curl http://worker-1.ocp4.tektutor.org.labs:31504
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org > ~/openshift-june-2024 > ⌂ main curl http://worker-2.ocp4.tektutor.org.labs:31504
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
```

```

jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ curl http://worker-2.ocp4.tektutor.org.labs:31504
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔ curl http://worker-2.ocp4.tektutor.org.labs:31504
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org ~ /openshift-june-2024 ✘ main ➔

```

Points to note

- Kubernetes & Openshift reserves port 30000-32767 port range for the purpose of nodeport external service
- Hence, when we create nodeport service any one of the ports on the above range will be randomly picked by Kubernetes/OpenShift
- the single Nodeport picked by OpenShift is opened on the nodes in the openshift cluster
- each nodeport service, it is going to open one port on every node, so the more nodeport services we create it ends up opening more number of ports in the openshift cluster, this might lead to security issues
- also the way we access the nodeport service is neither developer friendly nor end-user friendly
- In openshift, we could use route to make a service available outside the cluster scope which kind of addresses all the nodeport issues we discussed earlier

Lab - Creating a load balancer external service for nginx deployment in imperative style

We need to delete the existing nodeport external service

```

oc get svc
oc delete svc/nginx
oc get svc

```

Let's create the load balancer service for nginx deployment

```
oc expose deploy/nginx --type=LoadBalancer --port=8080
oc get svc
oc describe svc
```

In the above output, you would have noticed that the external ip is pending.

Points to note

- LoadBalancer service is used in public cloud environments like AWS, GCP, Azure, Digital Ocean, etc.,
- It is not normally used in local bare metal openshift setup like ours
- In case, you wish to use LoadBalancer external service in a bare-metal openshift cluster, we need to install metallb operator as an Openshift Administrator
- Post installing metallb operator, we need to allocate some available IP address range for the purpose of metallb load balancer
- We also need to create a metallb controller instance by creating a metallb instance
- In Kubernetes/OpenShift, there is a pod called kube-proxy which runs in every node
- kube-proxy provides load-balancing functionality for the clusterip and nodeport services
- In case of Load Balancer service, an external balancer with static ip accessible over internet is assigned for the external load balancer instance created in aws/azure/gcp, etc.,
- the metallb operator spins-off a load-balancer locally within the openshift cluster supporting a similar loadbalancing just like the public clouds

Lab - Installing Metallb Operator as a Openshift administrator

For detailed instructions on how to install Metallb operator in Openshift, you may refer my medium blog

<https://medium.com/tektutor/using-metallb-loadbalancer-with-bare-metal-openshift-onprem-4230944bfa35>

OperatorHub - Red Hat OpenShift - Google Chrome

Untitled des... OperatorHub Webex link fo... openshift-jun... Inbox (7,582) Webex link fo... 24MAN0852 Cisco Web Cluster Setti... bitnami/nginx Load Balance

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

OperatorHub

Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. You can purchase commercial software through [Red Hat Marketplace](#). You can install Operators on your clusters to provide optional add-ons and shared services to your developers. After installation, the Operator capabilities will appear in the [Developer Catalog](#) providing a self-service experience.

All Items

Search: metal

3 items

 Bare Metal Event Relay provided by Red Hat Inc	 MetalLB Operator provided by Red Hat	 OpenShift sandboxed containers Operator provided by Red Hat
This software manages the lifecycle of the hw-event-proxy container.	An operator for deploying...	OpenShift sandboxed containers, based on the Kata Containers open source project, provides an...
	 Installed	

<https://console-openshift-console.apps.ocp4.tektutor.org.labs/operatorhub/ns/jegan?keyword=metal>

Installed Operators - Red Hat OpenShift - Google Chrome

Untitled des... Installed Oper... Webex link fo... Editing oper... Inbox (7,582) Webex link fo... 24MAN0852 Cisco Web Cluster Setti... bitnami/nginx Load Balance

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the [Understanding Operators documentation](#). Or create an Operator and ClusterServiceVersion using the [Operator SDK](#).

Name	Managed Namespaces	Status	Last updated	Provided APIs
 MetalLB Operator 4.15.0-202406111806 provided by Red Hat	 jegan	 Succeeded Up to date	24 Jun 2024, 17:15	BGPPeer AddressPool BFDProfile BGPAdvertisement View 4 more...

```
jegan@tektutor.org ~/openshift-june-2024/Day1/metallb [main] oc apply -f address-pool.yml
Warning: metallb.io v1alpha1 AddressPool is deprecated
addresspool.metallb.io/tektutor-metallb-addresspool created
jegan@tektutor.org ~/openshift-june-2024/Day1/metallb [main] oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx    LoadBalancer  172.30.57.200  192.168.122.90  8080:30881/TCP  37m
jegan@tektutor.org ~/openshift-june-2024/Day1/metallb [main] curl http://192.168.122.90:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org ~/openshift-june-2024/Day1/metallb [main]
```