

## **Exercise 13**

### **OpenShift: Working with Pods**

#### **Q1.**

Take access to the cluster using the provided credentials.

Create a new project with your name and then choose that project as your current.

- a. Create a Pod with Nginx image through CLI method.
- b. Create a Pod with image: docker.io/dtg2468/custom-mysql:latest image using YAML file with Dry Run.
- c. Create a Pod with image: docker.io/dtg2468/my-apache:1.0 using YAML without Dry Run and creating the file from scratch yourself.
- d. Create a Pod with the Pod name “pod-console” and container name “con-console” using the already provided httpd image and leaving the remaining parameters as it is.

## Solutions

### Q1.

```
[root@workstation ~]# oc login -u admin -p redhatocp https://api.ocp4.example.com:6443
Login successful.

You have access to 69 projects, the list has been suppressed. You can list all projects with 'oc projects'
'

Using project "default".
Welcome! See 'oc help' to get started.
```

```
[root@workstation ~]# oc new-project deep
Now using project "deep" on server "https://api.ocp4.example.com:6443".

You can add applications to this project with the 'new-app' command. For example, try:

  oc new-app rails-postgresql-example

to build a new example application in Ruby. Or use kubectl to deploy a simple Kubernetes application:

  kubectl create deployment hello-node --image=k8s.gcr.io/e2e-test-images/agnhost:2.33 -- /agnhost serv
e-hostname

[root@workstation ~]# oc project deep
Already on project "deep" on server "https://api.ocp4.example.com:6443".
```

#### a.

```
[root@workstation ~]# oc run my-pod --image nginx
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "my-pod" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "my-pod" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "my-pod" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "my-pod" must set securityContext.seccompProfile.type to "RuntimeDefault" or "Localhost")
pod/my-pod created
[root@workstation ~]# oc get pods
NAME      READY   STATUS        RESTARTS   AGE
my-pod    0/1    ContainerCreating   0          9s
[root@workstation ~]# oc get pods
NAME      READY   STATUS        RESTARTS   AGE
my-pod    1/1    Running     0          2m13s
```

**b.**

```
[student@workstation ~]$ oc run my-pod2 --image docker.io/dtg2468/custom-mysql:latest --dry-run=client -o
yaml > mysql-pod.yaml
[student@workstation ~]$ cat mysql-pod.yaml
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: my-pod2
  name: my-pod2
spec:
  containers:
  - image: docker.io/dtg2468/custom-mysql:latest
    name: my-pod2
    resources: {}
  dnsPolicy: ClusterFirst
  restartPolicy: Always
status: {}
```

```
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: my-pod2
    name: my-pod2
spec:
  containers:
  - image: docker.io/dtg2468/custom-mysql:latest
    name: my-pod2
    env:
      - name: MYSQL_ROOT_PASSWORD
        value: redhat
    resources: {}
  dnsPolicy: ClusterFirst
  restartPolicy: Always
status: {}
```

```
[student@workstation ~]$ oc create -f mysql-pod.yaml
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "my-pod2" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "my-pod2" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "my-pod2" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "my-pod2" must set securityContext.seccompProfile.type to "RuntimeDefault" or "Localhost")
pod/my-pod2 created
[student@workstation ~]$ oc get pods
NAME      READY   STATUS    RESTARTS   AGE
my-pod    1/1     Running   2          175m
my-pod2   0/1     ContainerCreating   0          5s
[student@workstation ~]$ oc get pods
NAME      READY   STATUS    RESTARTS   AGE
my-pod    1/1     Running   2          176m
my-pod2   1/1     Running   0          16s
```

c.

```
apiVersion: v1
kind: Pod
metadata:
  name: new-pod
spec:
  containers:
    - name: new-con
      image: docker.io/dtg2468/my-apache:1.0
```

```
[root@workstation ~]# oc create -f my-pod.yaml
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "new-con" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "new-con" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "new-con" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "new-con" must set securityContext.seccompProfile.type to "RuntimeDefault" or "Localhost")
pod/new-pod created
[root@workstation ~]# oc get pods
NAME      READY   STATUS    RESTARTS   AGE
my-pod    1/1     Running   2          23h
my-pod2   1/1     Running   0          20h
new-pod   0/1     ContainerCreating   0          4s
```

d.

**Overview**

**Cluster**

**Getting started resources**

- [Set up your cluster](#)
- [Build with guided documentation](#)
- [Explore new admin features](#)

Finish setting up your cluster with recommended configurations.

Configure alert receivers →

View all steps in documentation →

**Status**

Cluster	Control Plane	Operators	Insights
✓ Cluster	✓ Single master	✓	Disabled
✓ Dynamic Plugins			

**Activity**

Ongoing

Recent events

View details

View alerts

View events

See what's new in OpenShift 4.12 →

**Details**

Cluster API address  
https://api.ocp4.example.com:6443

Cluster ID  
26f5365f-fd3e-4723-adca-afc106c4d58

OpenShift Cluster Manager →

Infrastructure provider  
None

OpenShift version  
4.12.0

Update channel  
Not available

Control plane high availability  
No (single master)

**Status**

**Activity**

**Recent events**

**View details**

**SamplesImagestreamImportFailing**

Oct 1, 2023, 6:08 AM

Samples operator is detecting problems with imagestream image imports. You can look at the "openshift-samples" ClusterOperator object for details. Most likely there are issues with the external image registry hosting the images that needs to be investigated. Or you can consider marking samples operator Removed if you do not care about having sample imagestreams available. The list of ImageStreams for which samples operator is reusing imports:

2:57 AM ⚡ Status for clusteroperator/auth... →

2:57 AM ⚡ Liveness probe failed: Get "... →

2:57 AM ⚡ Readiness probe error: Get ... →

Red Hat OpenShift

Administrator

Project: deep

## Pods

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
my-pod	Running	1/1	2	No owner	8.9 MiB	-	Oct 1, 2023, 2:55 AM
my-pod2	Running	1/1	0	No owner	382.3 MiB	0.004 cores	Oct 1, 2023, 5:51 AM
new-pod	Running	1/1	0	No owner	31.2 MiB	0.000 cores	Oct 2, 2023, 2:48 AM

Create Pod

Red Hat OpenShift

Administrator

Project: deep

## Create Pod

Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the editor.

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-console
  labels:
    app: httpd
  namespace: deep
spec:
  securityContext:
    runAsNonRoot: true
    seccompProfile:
      type: RuntimeDefault
  containers:
    - name: con-console
      image: image-registry.openshift-image-registry.svc:5000/openshift/httpd:latest
      ports:
        - containerPort: 8080
      securityContext:
        allowPrivilegeEscalation: false
        capabilities:
          drop:
            - ALL
```

**Pod**

**Schema**

Pod is a collection of containers that can run on a host. This resource is created by clients and scheduled onto hosts.

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

[View details](#)

Create Cancel Download

Red Hat OpenShift

Administrator

Home >

Operators >

Workloads

Pods

Deployments

DeploymentConfigs

StatefulSets

Secrets

ConfigMaps

CronJobs

Jobs

DaemonSets

ReplicaSets

ReplicationControllers

HorizontalPodAutoscalers

PodDisruptionBudgets

Networking >

Project: deep ▾

Pods > Pod details

P pod-console ⚡ Running

Details Metrics YAML Environment Logs Events Terminal Actions ▾

**Pod details**

Name	Status
pod-console	Running
Namespace	Restart policy
NS deep	Always restart
Labels	Edit ⚡ Active deadline seconds
app=httpd	Not configured
Node selector	Pod IP
No selector	10.8.0.171
Tolerations	Host IP
2 tolerations ⚡	192.168.50.10
Annotations	Node
5 annotations ⚡	master01
Created at	Image pull secret
Oct 2, 2023, 3:11 AM	default-dockercfg-6qcwf
Owner	PodDisruptionBudgets
No owners	No PodDisruptionBudgets