.

Lesson 03 Demo 01

Creating a Kubernetes Pod

Objective: To create and describe a Kubernetes pod to understand its lifecycle

Tools required: kubeadm, kubectl, kubelet, and containerd

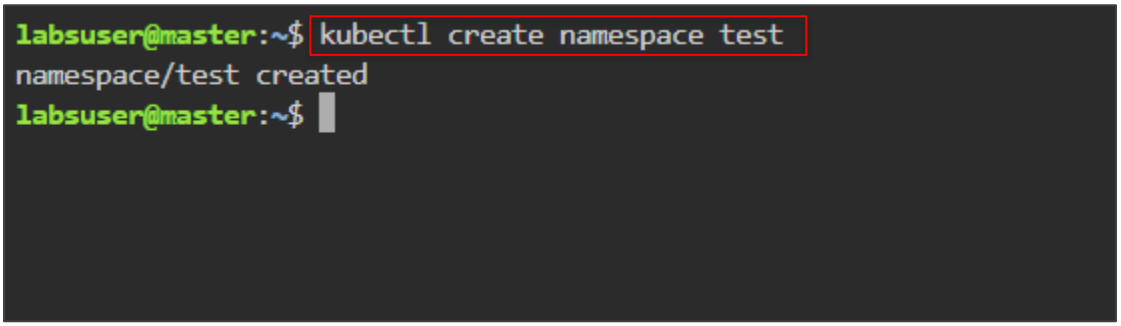
Prerequisites: A Kubernetes cluster (refer to Demo 01 from Lesson 01 for setting up a cluster)

Steps to be followed:

1. Create and describe a Kubernetes pod

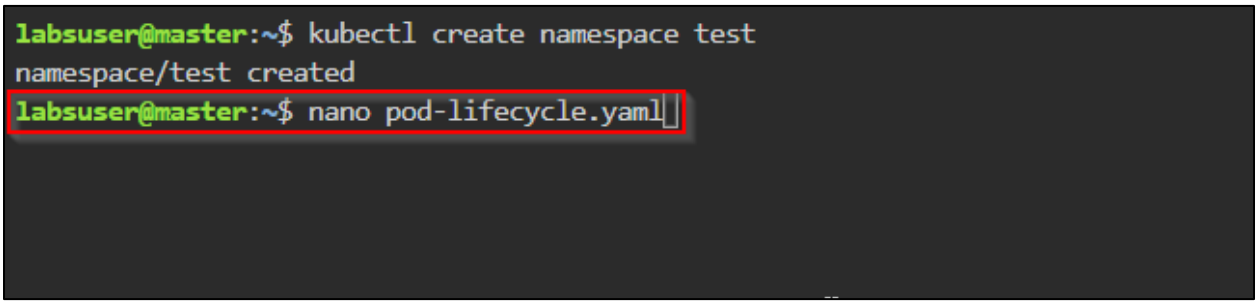
Step 1: Create and describe a Kubernetes pod

1.1 Create a namespace named **test** using the following command:
kubectl create namespace test



.

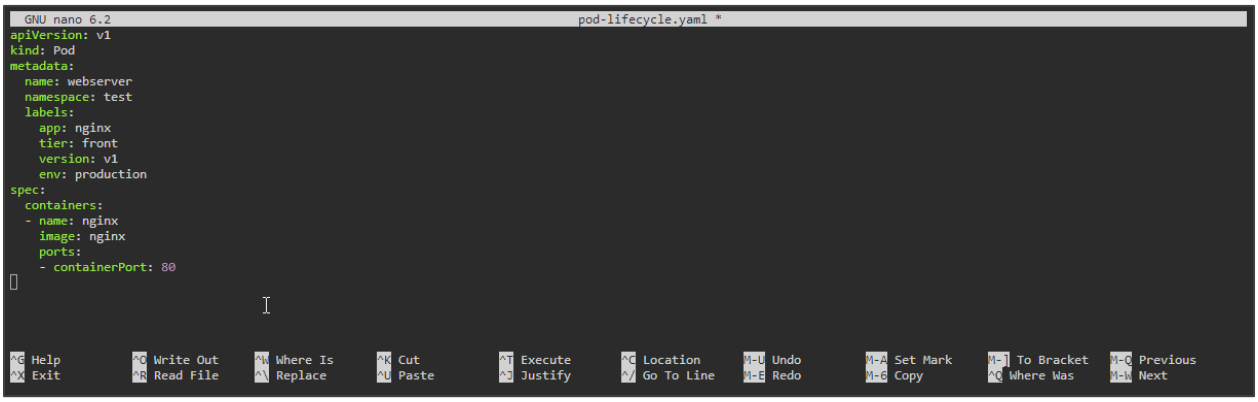
1.2 Execute the following command to create a YAML file that specifies the pod lifecycle configurations:
nano pod-lifecycle.yaml



1.3 Add the following code to the **pod-lifecycle.yaml** file:

```
apiVersion: v1
kind: Pod
metadata:
  name: webserver
  namespace: test
  labels:
    app: nginx
    tier: front
    version: v1
    env: production
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

.



1.4 Execute the following command to create a pod based on the configuration defined in the **pod-lifecycle.yaml** file:
kubectl create -f pod-lifecycle.yaml

```
labsuser@master:~$ kubectl create namespace test
namespace/test created
labsuser@master:~$ nano pod-lifecycle.yaml
labsuser@master:~$ kubectl create -f pod-lifecycle.yaml
pod/webserver created
labsuser@master:~$
```

1.5 Execute the following command to check the status of pods in **test**:
kubectl get pods -n test

```
labsuser@master:~$ kubectl create namespace test
namespace/test created
labsuser@master:~$ nano pod-lifecycle.yaml
labsuser@master:~$ kubectl create -f pod-lifecycle.yaml
pod/webserver created
labsuser@master:~$ kubectl get pods -n test
NAME        READY   STATUS    RESTARTS   AGE
webserver   1/1     Running   0          114s
labsuser@master:~$
```

.

1.6 Execute the following command to describe the specifics of the web server:
kubectl describe pod webserver -n test

```
labsuser@master:~$ kubectl describe pod webserver -n test
Name:             webserver
Namespace:        test
Priority:         0
Service Account:  default
Node:             worker-node-2.example.com/172.31.25.241
Start Time:       Tue, 17 Oct 2023 18:47:58 +0000
Labels:           app=nginx
                  env=production
                  tier=front
                  version=v1
Annotations:      cni.projectcalico.org/containerID: 6ae75bc3087e1d95cbf1bd7794d49fbb047b50d8f43eec5daae4de2a914b5f63
                  cni.projectcalico.org/podIP: 192.168.232.227/32
                  cni.projectcalico.org/podIPs: 192.168.232.227/32
Status:           Running
```

```
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
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  4m27s  default-scheduler  Successfully assigned test/webserver to worker-node-2.example.com
  Normal  Pulling    4m26s  kubelet            Pulling image "nginx"
  Normal  Pulled     4m26s  kubelet            Successfully pulled image "nginx" in 621ms (621ms including waiting)
  Normal  Created    4m26s  kubelet            Created container nginx
  Normal  Started    4m26s  kubelet            Started container nginx
labsuser@master:~$
```

This provides detailed information about the current state, events, and configuration of the specified pod in the **test** namespace.

By following these steps, you have successfully created a Kubernetes pod to understand its lifecycle.