Lesson 07 Demo 06

Analyzing Pod Logs

Objective: To analyze the logs of a pod hosting a container within a Kubernetes cluster to view the application or service's output

Tools required: kubeadm, kubectl, kubelet, and containerd

Prerequisites: A Kubernetes cluster (refer to Demo 01 from Lesson 01 for setting up a cluster)
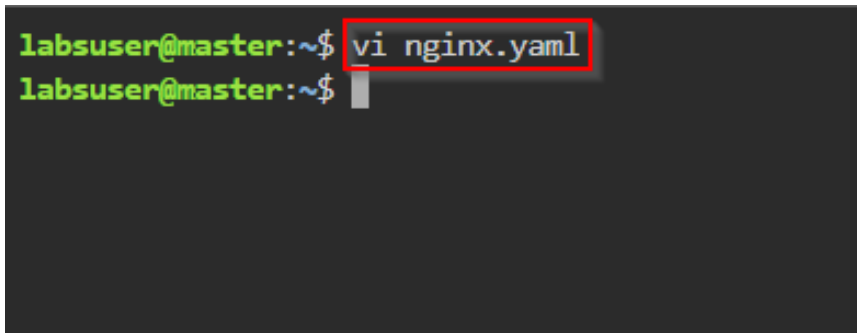
Steps to be followed:
1. Configure and verify Nginx deployment

Step 1: Configure and verify Nginx deployment

1.1 Initiate the creation of a configuration file for the Nginx deployment using the following command:

vi nginx.yaml

1.2 Add the following code to the **nginx.yaml** file:

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
 labels:
   app: nginx
  name: nginx
spec:
 replicas: 1
 selector:
   matchLabels:
     app: nginx
 template:
  metadata:
   labels:
     app: nginx
  spec:
   containers:
   - image: nginx
     name: nginx
```

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
      - image: nginx
        name: nginx
        resources: {}
~
```

1.3 Run the following command to create the Nginx deployment
   kubectl create -f nginx.yaml

```
labsuser@master:~$ kubectl create -f nginx.yaml
deployment.apps/nginx created
labsuser@master:~$
```

1.4 Use the following commands to verify deployments and pods:
   kubectl get deployments
   kubectl get pods

```
labsuser@master:~$ kubectl create -f nginx.yaml
deployment.apps/nginx created
labsuser@master:~$ kubectl get deployments
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
admin   0/1     1            0           41h
nginx   1/1     1            1           90s
labsuser@master:~$ kubectl  get pods
NAME                     READY   STATUS           RESTARTS      AGE
admin-56d684dff9-zjfhc   0/1     ImagePullBackOff 0             41h
counter                  1/1     Running          3 (14m ago)   21h
nginx-7854ff8877-mvrtr   1/1     Running          0             112s
pod-demo                 1/1     Running          7 (14m ago)   5d14h
labsuser@master:~$
```

1.5 Use the following command to view the logs of the Nginx pod, replacing **nginx-7854ff8877-mvrtr** with the pod name from step 1.4:
kubectl logs nginx-7854ff8877-mvrtr

```
labsuser@master:~$ kubectl create -f nginx.yaml
deployment.apps/nginx created
labsuser@master:~$ kubectl get deployments
NAME     READY   UP-TO-DATE   AVAILABLE   AGE
admin    0/1     1            0           41h
nginx    1/1     1            1           90s
labsuser@master:~$ kubectl  get pods
NAME                      READY   STATUS            RESTARTS      AGE
admin-56d684dff9-zjfhc    0/1     ImagePullBackOff  0             41h
counter                   1/1     Running           3 (14m ago)   21h
nginx-7854ff8877-mvrtr    1/1     Running           0             112s
pod-demo                  1/1     Running           7 (14m ago)   5d14h
labsuser@master:~$ kubectl logs nginx-7854ff8877-mvrtr
/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: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/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
2023/10/12 04:56:50 [notice] 1#1: using the "epoll" event method
2023/10/12 04:56:50 [notice] 1#1: nginx/1.25.2
2023/10/12 04:56:50 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2023/10/12 04:56:50 [notice] 1#1: OS: Linux 6.2.0-1013-aws
2023/10/12 04:56:50 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1024:524288
2023/10/12 04:56:50 [notice] 1#1: start worker processes
2023/10/12 04:56:50 [notice] 1#1: start worker process 29
2023/10/12 04:56:50 [notice] 1#1: start worker process 30
labsuser@master:~$
```

By following these steps, you have successfully monitored the Nginx pod's logs following the creation of its deployment in Kubernetes.