Practical - 9

AIM : Performing basics commands to interact with Kubernetes.

The objective of this lab is to familiarize yourself with basic commands to interact with a Kubernetes cluster. You will learn how to perform essential operations such as deploying pods, checking cluster status, and inspecting resources.

Step 1: Verify `kubectl` Configuration

kubectl config current-context

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl config current-context
docker-desktop
```

Step 2: List Nodes

kubectl get nodes

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl get nodes
NAME             STATUS   ROLES           AGE   VERSION
docker-desktop   Ready    control-plane   14m   v1.28.2
```

Step 3: Create a Deployment

kubectl create deployment nginx-deployment--image=nginx

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl create deployment nginx-deployment --image=nginx
deployment.apps/nginx-deployment created
```

kubectl get deployments

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   1/1     1            1           21s
```

Step 4: List Pods

kubectl get pods

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-6d6565499c-cg6bc   1/1     Running   0          66s
```

Step 5: Access Pod Logs

kubectl logs nginx-deployment-6d6565499c-rxvct

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl logs nginx-deployment-6d6565499c-cg6bc
/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
2023/12/02 06:57:21 [notice] 1#1: using the "epoll" event method
2023/12/02 06:57:21 [notice] 1#1: nginx/1.25.3
2023/12/02 06:57:21 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2023/12/02 06:57:21 [notice] 1#1: OS: Linux 5.10.102.1-microsoft-standard-WSL2
2023/12/02 06:57:21 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/12/02 06:57:21 [notice] 1#1: start worker processes
2023/12/02 06:57:21 [notice] 1#1: start worker process 29
2023/12/02 06:57:21 [notice] 1#1: start worker process 30
2023/12/02 06:57:21 [notice] 1#1: start worker process 31
2023/12/02 06:57:21 [notice] 1#1: start worker process 32
2023/12/02 06:57:21 [notice] 1#1: start worker process 33
2023/12/02 06:57:21 [notice] 1#1: start worker process 34
2023/12/02 06:57:21 [notice] 1#1: start worker process 35
2023/12/02 06:57:21 [notice] 1#1: start worker process 36
```

Step 6: Expose Deployment as a Service

kubectl expose deployment nginx-deployment--port=80--type=NodePort--name=nginx-service

kubectl expose deployment nginx-deployment --port=80 --type=NodePort

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl expose deployment nginx-deployment --port=80 --type=NodePort
service/nginx-deployment exposed
```

Step 7: List Services

kubectl get services

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl get services
NAME               TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes         ClusterIP   10.96.0.1       <none>        443/TCP        100m
nginx-deployment   NodePort    10.96.22.226    <none>        80:30584/TCP   22s
```

Step 8: Access the NGINX Service

kubectl describe service nginx-deployment

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl describe service nginx-deployment
Name:                     nginx-deployment
Namespace:                default
Labels:                   app=nginx-deployment
Annotations:              <none>
Selector:                 app=nginx-deployment
Type:                     NodePort
IP Family Policy:         SingleStack
IP Families:              IPv4
IP:                       10.96.22.226
IPs:                      10.96.22.226
LoadBalancer Ingress:     localhost
Port:                     <unset>  80/TCP
TargetPort:               80/TCP
NodePort:                 <unset>  30584/TCP
Endpoints:                10.1.0.6:80
Session Affinity:         None
External Traffic Policy:  Cluster
Events:                   <none>
```

Step 9: Delete Resources

kubectl delete deployment nginx-deployment

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl delete deployment nginx-deployment
deployment.apps "nginx-deployment" deleted
```

kubectl delete service nginx-deployment

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl delete service nginx-deployment
service "nginx-deployment" deleted
```

Step 10: Scale Deployment

kubectl scale deployment nginx-deployment--replicas=3

```
PS C:\Users\Nakul\Downloads\MLOPs\Practicals\Practical 9> kubectl scale deployment nginx-deployment --replicas=3
deployment.apps/nginx-deployment scaled
```

kubectl get pods

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl get pods
No resources found in default namespace.
```

Step 11: Update Deployment

kubectl set image deployment/nginx-deployment nginx=nginx:1.21

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl set image deployment/nginx-deployment nginx=nginx:1.21
Error from server (NotFound): deployments.apps "nginx-deployment" not found
```

kubectl rollout status deployment/nginx-deployment

```
PS C:\Users\Nakul\Downloads\MLOPs\Practicals\Practical 9> kubectl rollout status deployment/nginx-deployment
Waiting for deployment "nginx-deployment" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx-deployment" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx-deployment" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx-deployment" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nginx-deployment" rollout to finish: 1 old replicas are pending termination...
deployment "nginx-deployment" successfully rolled out
```

Step 12: Rollback Deployment

kubectl rollout undo deployment/nginx-deployment

```
PS E:\7sem\MLOPS\practicals\wordfiles\practical_9> kubectl rollout undo deployment/nginx-deployment
Error from server (NotFound): deployments.apps "nginx-deployment" not found
```

In this lab, we've learned basic Kubernetes commands to interact with a Kubernetes cluster. We've performed operations like creating deployments, services, scaling, updating, and rolling back deployments. These fundamental commands are essential for managing and working with Kubernetes resources in a real-world cluster.