**Lab Exercise 27- Using LoadBalancer Service in Kubernetes**

This lab exercise will guide you through deploying an application in Kubernetes and exposing it externally using a **LoadBalancer** service.

**Objective**

* Deploy an Nginx application.
* Expose it to the outside world using a **LoadBalancer** service.
* Access the application from a browser or external network.

**Prerequisites**

1. A Kubernetes cluster with **LoadBalancer** support:
   * If using a cloud provider (e.g., AWS, GCP, Azure), LoadBalancer services are supported by default.
   * For Minikube, enable the metallb add-on for LoadBalancer services.
   * For Kind, install **MetalLB** manually.
2. **kubectl** installed and configured.

**Steps**

**1. Deploy Nginx**

Create a Deployment for the Nginx application.

**nginx-deployment.yaml**

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-deployment

labels:

app: nginx

spec:

replicas: 2

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx:latest

ports:

- containerPort: 80

**Apply the deployment:**

kubectl apply -f nginx-deployment.yaml

**Verify the deployment:**

kubectl get deployments

kubectl get pods

**2. Create a LoadBalancer Service**

Expose the Nginx deployment using a LoadBalancer service.

**nginx-service.yaml**:

apiVersion: v1

kind: Service

metadata:

name: nginx-service

spec:

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80

type: LoadBalancer

**Apply the service:**

kubectl apply -f nginx-service.yaml

**Check the service:**

kubectl get services

**3. Verify External Access**

1. **Obtain the External IP Address**:
   * Check the external IP of the LoadBalancer service:

kubectl get svc nginx-service

* + The external IP will appear under the EXTERNAL-IP column (it may take a few minutes to populate).

1. **Access the Application**:
   * Open a browser and navigate to http://<EXTERNAL-IP> or use curl:

curl http://<EXTERNAL-IP>

* + You should see the Nginx default welcome page.

**4. Test Load Balancing**

Since you have two replicas of the Nginx pods, the LoadBalancer will distribute traffic between them. To verify:

1. **Check the logs of Nginx pods**:

kubectl logs -l app=nginx

This will show requests being served by different pods, demonstrating load balancing.

**5. Cleanup**

Remove the resources created during this lab:

kubectl delete -f nginx-deployment.yaml

kubectl delete -f nginx-service.yaml

**Expected Output**

1. The Nginx deployment creates two pods.
2. The LoadBalancer service exposes the Nginx application to an external IP.
3. You can access the application externally using the provided IP.