**Deploy a microservice application on AKS cluster and access it using public internet**

Deploying a microservice application on Azure Kubernetes Service (AKS) and making it accessible from the public internet.

There are some steps as follow:

Step 1: First I’ll create an AKS Cluster, for this

First, creating a resource group and AKS cluster:

$ az login

$ az group create --name myResourceGroup --location eastus

$ az aks create \

--resource-group myResourceGroup \

--name myAKSCluster \

--node-count 2 \

--enable-addons monitoring \

--generate-ssh-keys

Connecting to my cluster:

$ az aks get-credentials --resource-group myResourceGroup --name myAKSCluster

# Verify connection

$ kubectl get nodes

Step 2: Deploying a Sample Microservice Application

# app-deployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

name: sample-app

spec:

replicas: 3

selector:

matchLabels:

app: sample-app

template:

metadata:

labels:

app: sample-app

spec:

containers:

- name: web

image: nginx:latest

ports:

- containerPort: 80

resources:

requests:

memory: "128Mi"

cpu: "100m"

limits:

memory: "256Mi"

cpu: "200m"

Deploy the application:

$ kubectl apply -f app-d ployment.yaml

Step 3: Now I’m Exposing the Application to Public Internet using LoadBalancer Service

Creating a LoadBalancer service:

# app-service.yaml

apiVersion: v1

kind: Service

metadata:

name: sample-app-service

spec:

type: LoadBalancer

selector:

app: sample-app

ports:

- port: 80

targetPort: 80

protocol: TCP

Apply the service:

$ kubectl apply -f app-service.yaml

Get the external IP:

$ kubectl get service sample-app-service

Step 4: Accessing our Application

Once we have the external IP address, we can access our application as follow:

Using curl

$ curl http://<EXTERNAL-IP>

Or open in browser

http://<EXTERNAL-IP>