Install metrics api

<https://github.com/amitopenwriteup/k8s/blob/main/components.yaml>

Copy the yaml file from above location and create yaml in your linux box

Run command : kubectl create -f <name of yaml>

**Step 1: Create a Deployment**

Let's deploy an Nginx server with CPU requests and limits:

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-deployment

labels:

app: nginx

spec:

replicas: 1

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx

resources:

requests:

cpu: "50m"

limits:

cpu: "200m"

Apply it:

kubectl apply -f nginx-deployment.yaml

**Step 2: Create an HPA**

This will autoscale the deployment when CPU usage exceeds **50%**:

apiVersion: autoscaling/v2

kind: HorizontalPodAutoscaler

metadata:

name: nginx-hpa

spec:

scaleTargetRef:

apiVersion: apps/v1

kind: Deployment

name: nginx-deployment

minReplicas: 1

maxReplicas: 5

metrics:

- type: Resource

resource:

name: cpu

target:

type: Utilization

averageUtilization: 50

Check HPA:

kubectl get hpa

Step 3: Create Service

apiVersion: v1

kind: Service

metadata:

name: nginx-service

spec:

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80

type: ClusterIP

kubectl create -f svc.yaml

Load generators

apiVersion: v1

kind: Pod

metadata:

name: load-generator

spec:

containers:

- name: busybox

image: busybox

command: ["/bin/sh", "-c", "while true; do wget -q -O- http://nginx-service.default.svc.cluster.local; done"]

kubectl apply -f load-generator.yaml

kubectl get pods

kubectl get hpa

If scaling works, you should see more pods being created.

**Cleanup**

kubectl delete deployment nginx-deployment

kubectl delete hpa nginx-deployment