Services

» Cluster IP

Overview

Use `kubectl explain` to see available keys/fields for a service manifest.

```
kubectl explain service --recursive=true
kubectl explain service.spec.ports --recursive=true
```

Expose the `my-first-replica-set` ReplicaSet to the cluster with a ClusterIP service.

Create the ClusterIP Service object manifest.

```
cat > service-clusterip.yaml <<EOF
kind: Service
apiVersion: v1
metadata:
  name: myfirstservice
spec:
  type: ClusterIP
  selector:
    app: myfirstapp
  ports:
  - protocol: TCP
    port: 8080
    targetPort: 80
EOF
```

Create the ClusterIP Service.

```
kubectl create -f service-clusterip.yaml
```

You could have also used the `kubectl expose` command to create the service.

```
# kubectl expose replicaset myfirstreplicaset --name=myfirstservice --port 8080 --target-port 80
```

Confirm the ClusterIP service was successfully created.

```
kubectl get svc
```

Creating a service automatically generates an Endpoints object that manages pod IP addresses.

```
kubectl get endpoints
```

View the manifest for the Service resource.

```
kubectl get svc myfirstservice -o yaml
```

Run a separate pod to access the service.

```
kubectl run --image=busybox busybox --restart=Never -- sleep 6000
```

```
kubectl exec -it busybox -- /bin/sh
```

Environment Variables

All services are advertised as variables within the Pod. The downside to variables is that they do not automatically update when new services are created. You must restart the pod to see new variables.

```
env
```

Connecting to a ClusterIP Service

Verify that the service can be accessed using the IP address specified in the environment variable.

```
wget -qO - http://$MYFIRSTSERVICE_SERVICE_HOST:$MYFIRSTSERVICE_SERVICE_PORT
```

DNS - ClusterIP

Verify that the service can be accessed with a hostname using DNS.

```
cat /etc/resolv.conf
```

```
nslookup myfirstservice
```

Verify that the service can be accessed using the IP address specified in the environment variable.

```
wget -qO - myfirstservice:8080
```

You can also use the FQDN for the service

```
wget -qO - myfirstservice.default.svc.cluster.local:8080
```

Exit out of the pod

```
exit
```

Clean Up

Let's delete the `nginx` ClusterIP service

```
kubectl delete svc myfirstservice
```