CoreDNS is now the new default DNS plugin for Kubernetes. In this lesson, we’ll go over the hostnames for pods and services. We will also discover how you can customize DNS to include your own nameservers.

View the CoreDNS pods in the kube-system namespace:

kubectl get pods -n kube-system

View the CoreDNS deployment in your Kubernetes cluster:

kubectl get deployments -n kube-system

View the service that performs load balancing for the DNS server:

kubectl get services -n kube-system

Spec for the busybox pod:

apiVersion: v1

kind: Pod

metadata:

name: busybox

namespace: default

spec:

containers:

- image: busybox:1.28.4

command:

- sleep

- "3600"

imagePullPolicy: IfNotPresent

name: busybox

restartPolicy: Always

View the resolv.conf file that contains the nameserver and search in DNS:

kubectl exec -it busybox -- cat /etc/resolv.conf

Look up the DNS name for the native Kubernetes service:

kubectl exec -it busybox -- nslookup kubernetes

Look up the DNS names of your pods:

kubectl exec -ti busybox -- nslookup [pod-ip-address].default.pod.cluster.local

Look up a service in your Kubernetes cluster:

kubectl exec -it busybox -- nslookup kube-dns.kube-system.svc.cluster.local

Get the logs of your CoreDNS pods:

kubectl logs [coredns-pod-name]

YAML spec for a headless service:

apiVersion: v1

kind: Service

metadata:

name: kube-headless

spec:

clusterIP: None

ports:

- port: 80

targetPort: 8080

selector:

app: kubserve2

YAML spec for a custom DNS pod:

apiVersion: v1

kind: Pod

metadata:

namespace: default

name: dns-example

spec:

containers:

- name: test

image: nginx

dnsPolicy: "None"

dnsConfig:

nameservers:

- 8.8.8.8

searches:

- ns1.svc.cluster.local

- my.dns.search.suffix

options:

- name: ndots

value: "2"

- name: edns0