KUBERNETES FUNDAMENTALS (LFS258)

SUPPORT SIGN OUT

SERVICES

Services

Accessing an Application with a Service

The basic step to access a new service is to use **kubectl**. See the following commands and outputs:

```
$ kubectl expose deployment/nginx --port=80 --type=NodePort
```

```
$ kubectl get svc
```

```
NAME        TYPE       CLUSTER-IP  EXTERNAL-IP  PORT(S)       AGE
kubernetes  ClusterIP  10.0.0.1    <none>       443/TCP       18h
nginx       NodePort   10.0.0.112  <none>       80:31230/TCP  5s
```

```
$ kubectl get svc nginx -o yaml
```

```
apiVersion: v1
kind: Service
...
spec:
    clusterIP: 10.0.0.112
    ports:
    - nodePort: 31230
...
```

Open browser **`http://Public-IP:31230`**.

The **`kubectl expose`** command created a service for the **`nginx`** deployment. This service used port 80 and generated a random port on all the nodes. A particular **`port`** and **`targetPort`** can also be passed during object creation to avoid random values. The **`targetPort`** defaults to the port, but could be set to any value, including a string referring to a port on a backend Pod. Each Pod could have a different port, but traffic is still passed via the name. Switching traffic to a different port would maintain a client connection, while changing versions of software, for example.

The **`kubectl get svc`** command gave you a list of all the existing services, and we saw the **`nginx`** service, which was created with an internal cluster IP.

The range of cluster IPs and the range of ports used for the random NodePort are configurable in the API server startup options.

Services can also be used to point to a service in a different namespace, or even a resource outside the cluster, such as a legacy application not yet in Kubernetes.