From a security perspective, it is often a good idea to place network-level restrictions on any communication between different parts of your infrastructure. NetworkPolicies allow you to restrict and control the network traffic going to and from your pods. In this lesson, we will discuss NetworkPolicies and demonstrate how to create a simple policy to restrict access to a pod.

Relevant Documentation

- https://kubernetes.io/docs/concepts/services-networking/network-policies/

Lesson Reference

In order to use NetworkPolicies in the cluster, we need to have a network plugin that supports them. We can accomplish this alongside an existing flannel setup using canal:

```
wget -O canal.yaml https://docs.projectcalico.org/v3.5/getting-started/kubernetes/installation/hosted/canal/canal.yaml

kubectl apply -f canal.yaml
```

Create a sample nginx pod:

```
apiVersion: v1
kind: Pod
metadata:
  name: network-policy-secure-pod
  labels:
    app: secure-app
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

Create a client pod which can be used to test network access to the Nginx pod:

```
apiVersion: v1
kind: Pod
metadata:
  name: network-policy-client-pod
spec:
  containers:
  - name: busybox
    image: radial/busyboxplus:curl
    command: ["/bin/sh", "-c", "while true; do sleep 3600; done"]
```

Use this command to get the cluster IP address of the Nginx pod:

```
kubectl get pod network-policy-secure-pod -o wide
```

Use the secure pod's IP address to test network access from the client pod to the secure Nginx pod:

```
kubectl exec network-policy-client-pod -- curl <secure pod cluster ip address>
```

Create a network policy that restricts all access to the secure pod, except to and from pods which bear the `allow-access: "true"` label:

```yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: my-network-policy
spec:
  podSelector:
    matchLabels:
      app: secure-app
  policyTypes:
  - Ingress
  - Egress
  ingress:
  - from:
    - podSelector:
        matchLabels:
          allow-access: "true"
    ports:
    - protocol: TCP
      port: 80
  egress:
  - to:
    - podSelector:
        matchLabels:
          allow-access: "true"
    ports:
    - protocol: TCP
      port: 80
```

Get information about NetworkPolicies in the cluster:

```
kubectl get networkpolicies
kubectl describe networkpolicy my-network-policy
```