作者：李晓辉

联系方式：

1. 微信：Lxh_Chat

2. 邮箱：939958092@qq.com

有时候我们得限制Pod之间的访问，这就得靠配置网络策略啦。Kubernetes的网络策略超有意思，它用标签来控制Pod之间的网络流量，而不是靠IP地址呢，这样就能很方便地管理进出的流量。

我们来试试一个实验吧：

1. 在一个叫zhangsan的项目里，先搞两个Pod，让它们互相访问一下，看看能不能成功。
2. 再整一个叫lixiaohui的项目，也创建两个Pod，互相访问，测试一下。
3. 然后让这两个项目里的Pod互相访问，看看能不能搞定。
4. 最后，我们新建一个网络策略，再验证一下同项目和不同项目里的Pod互访能不能成功。

新建zhangsan project的资源

```
oc new-project zhangsan
```

在zhangsan的namespace中，新建一个名为hello开头的pod

```
oc new-app --name hello --image registry.ocp4.example.com:8443/redhattraining/hello-world-r
```

在zhangsan的namespace中，新建一个名为test开头的pod

```
oc new-app --name test --image registry.ocp4.example.com:8443/redhattraining/hello-world-ng
```

查看一下pod的ip

```
[root@workstation ~]# oc -n zhangsan get pod -o wide
NAME                     READY   STATUS    RESTARTS   AGE     IP          NODE       NOMINA
hello-7c5959664f-4mwg7   1/1     Running   0          8m22s   10.8.0.81   master01   <none>
test-54d78b7-gwc8g       1/1     Running   0          6s      10.8.0.84   master01   <none>
```

测试zhangsan project中的互访

从test的pod中发起对hello的pod访问，发现可以成功，证明同project互访ok

```
[root@workstation ~]# oc rsh test-54d78b7-gwc8g curl 10.8.0.81:8080
<html>
  <body>
    <h1>Hello, world from nginx!</h1>
  </body>
</html>
```

新建lixiaohui project的资源

```
oc new-project lixiaohui
```

```
oc new-app --name sample-app --image registry.ocp4.example.com:8443/redhattraining/hello-wo
```

检查是否能运行

```
[root@workstation ~]# oc get pod -o wide
NAME                          READY   STATUS    RESTARTS   AGE   IP           NODE       NOM
sample-app-564fdf8b8c-psjf8   1/1     Running   0          57s   10.8.0.85    master01   <no
```

验证跨project的访问

在新建网络策略之前，我们先试试他们跨project目前是否能访问

结果显示，跨project访问没问题

```
[root@workstation ~]# oc rsh sample-app-564fdf8b8c-psjf8 curl 10.8.0.81:8080
<html>
  <body>
    <h1>Hello, world from nginx!</h1>
  </body>
</html>
```

开启白名单

经过测试，上面不管是否跨project，都能互相访问，我们来试试，让hello这个pod除lixiaohui这个project外，不让所有人访问，开启白名单方式

先看看hello这个pod有什么标签

```
[root@workstation ~]# oc describe pod -n zhangsan hello-7c5959664f-4mwg7 | grep -A 2 Labels
Labels:           deployment=hello
                  pod-template-hash=7c5959664f
Annotations:      k8s.ovn.org/pod-networks:
```

在zhangsan的namespace下， 创建除网络策略

```
cat > only-allow-lixiaohui-project.yml <<-EOF
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: only-allow-lixiaohui-project
  namespace: zhangsan
spec:
  podSelector:
    matchLabels:
      deployment: hello
  policyTypes:
  - Ingress
  ingress:
  - from:
    - namespaceSelector:
        matchLabels:
          kubernetes.io/metadata.name: lixiaohui
    ports:
    - protocol: TCP
      port: 8080
EOF
```

```
[root@workstation ~]# oc create -f only-allow-lixiaohui-project.yml
[root@workstation ~]# oc get networkpolicies.networking.k8s.io -n zhangsan
NAME                          POD-SELECTOR       AGE
only-allow-lixiaohui-project   deployment=hello   18s
```

测试一下是否只允许lixiaohui的project访问

好的，看上去访问成功

```
[root@workstation ~]# oc get pod -n lixiaohui -o wide
NAME                          READY   STATUS    RESTARTS   AGE   IP          NODE       NOM
sample-app-564fdf8b8c-psjf8   1/1     Running   0          15m   10.8.0.85   master01   <no
[root@workstation ~]#
[root@workstation ~]#
[root@workstation ~]# oc get pod -n zhangsan -o wide
NAME                     READY   STATUS    RESTARTS   AGE   IP          NODE       NOMINATE
hello-7c5959664f-4mwg7   1/1     Running   0          31m   10.8.0.81   master01   <none>
test-54d78b7-gwc8g       1/1     Running   0          23m   10.8.0.84   master01   <none>
[root@workstation ~]#
[root@workstation ~]#
[root@workstation ~]#
[root@workstation ~]# oc rsh sample-app-564fdf8b8c-psjf8 curl 10.8.0.81:8080
<html>
  <body>
    <h1>Hello, world from nginx!</h1>
  </body>
</html>
```

不过在新建网络策略之前，它就是成功的，所以我们试试从同一个project中的test这个pod中访问试试

发现卡住不动了，无法访问

```
[root@workstation ~]# oc -n zhangsan rsh test-54d78b7-gwc8g curl 10.8.0.81:8080
```

但是跨project访问lixiaohui是可以的

```
[root@workstation ~]# oc -n zhangsan rsh test-54d78b7-gwc8g curl 10.8.0.85:8080
<html>
  <body>
    <h1>Hello, world from nginx!</h1>
  </body>
</html>
```