# Day14 容器进阶之Kubernetes 安全实战

## 1 打卡任务

作业：

通过CCE界面在同一个namespace下创建两个工作负载，workload01和workload02。设置网络隔离策略使workload01无法访问workload02。

打卡：

在workload01容器中访问workload02，截图网络策略设置前后的访问效果

## 2 准备工作

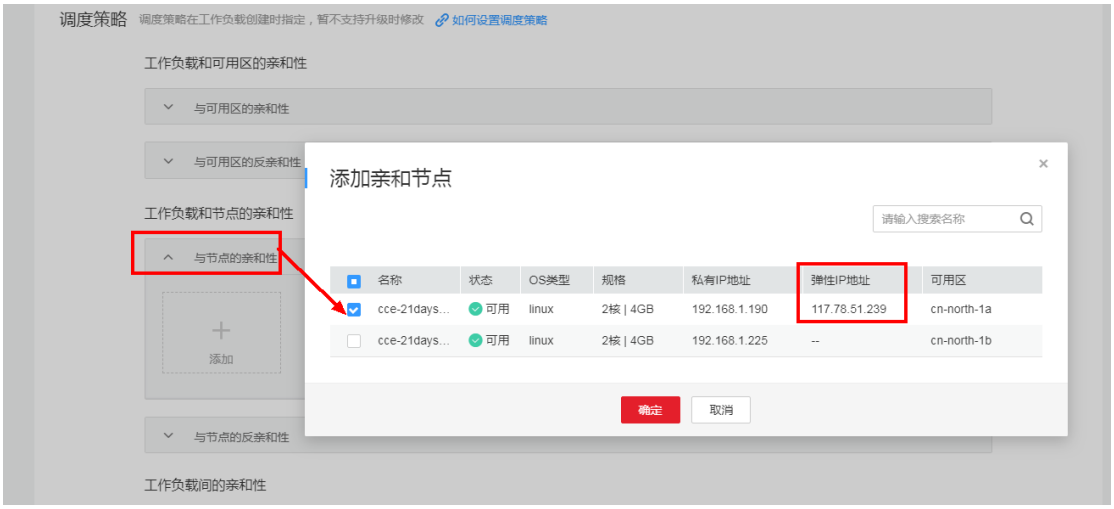
1、 私有镜像仓库中已存在入门课程中的坦克大战镜像

2、 已拥有可运行的CCE集群

## 3 通过网络隔离策略使 workload01 无法访问 workload02

1、 参考day8在CCE集群的default namespace下创建两个fc-tank应用

注意：创建最后一步时将应用都调度到有EIP的节点上。




2、 登录node节点（设置kubectl命令行的节点），通过kubectl从fc-tank-01访问fc-tank-02

1) 查询pod实例列表并获取pod的IP：

kubectl get pods -owide

2) 登录fc-tank-01的容器

```
[root@cce-21days-cluster-98379 ~]#
[root@cce-21days-cluster-98379 ~]#
[root@cce-21days-cluster-98379 ~]# kubectl get pods -owide
NAME                          READY     STATUS    RESTARTS   AGE       IP            NODE
fc-tank-01-c9849f648-9j2px    1/1       Running   0          1m        172.16.0.7    192.168.1.190
fc-tank-02-64899b7d96-6j59s   1/1       Running   0          46s       172.16.0.8    192.168.1.190
[root@cce-21days-cluster-98379 ~]#
[root@cce-21days-cluster-98379 ~]#
[root@cce-21days-cluster-98379 ~]# kubectl exec fc-tank-01-c9849f648-9j2px -ti /bin/bash
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
```

注意：由于容器中没有curl命令，执行以下命令安装即可

apt-get update

apt-get install curl -y

3) 使用curl命令访问fc-tank-02，返回数据表示访问成功

curl http://172.16.0.8:80

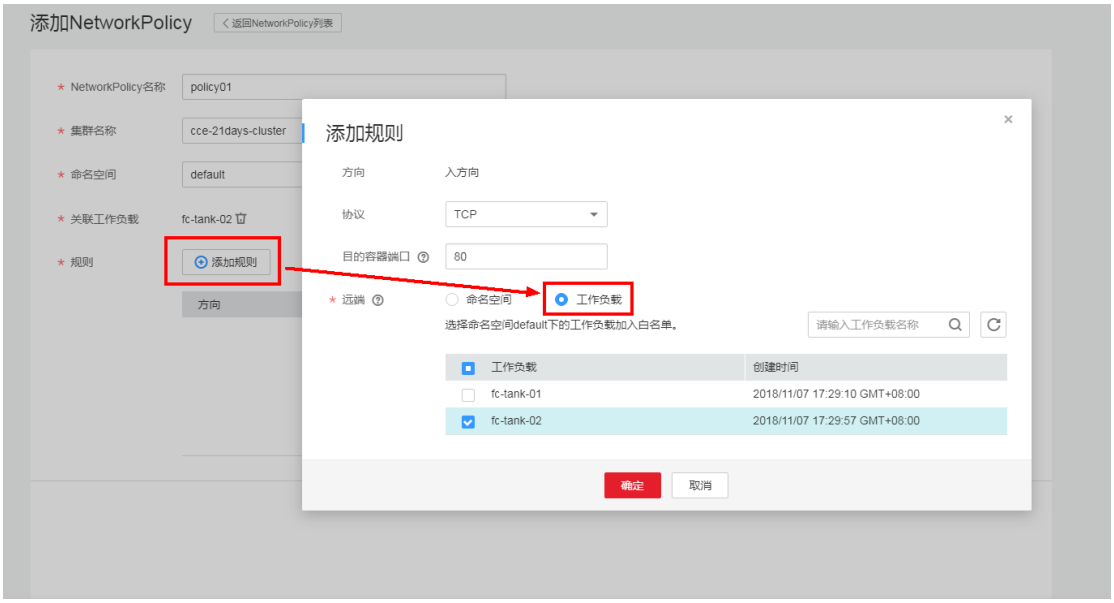
172.16.0.8为fc-tank-02的POD IP

```
root@fc-tank-01-c9849f648-9j2px:/# curl http://172.16.0.8:80
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=UTF-8">
  <title>FC Tank - a coffeescript rewritten of classic 90 tank on Nintendo Family Computer</title>
  <script type="text/javascript" src="js/coffeescript-v1.6.2.min.js"></script>
  <script type='text/javascript' src='js/jquery-v1.9.1.min.js'></script>
  <script type='text/javascript' src='js/lodash-v1.2.1.min.js'></script>
  <script type='text/javascript' src="js/kinetic-v4.5.1.min.js"></script>
  <script type='text/javascript' src="js/howler-v1.1.5.min.js"></script>
  <style type='text/css'>
    #tank_sprite { display: none; }
    #canvas {
      background: #000;
      width: 600px;
      height: 520px;
    }
    #tv_wrapper {
      width: 600px;
      height: 520px;
      padding: 192px 363px 210px 237px;
```

4) 在CCE界面上设置网络隔离策略，使fc-tank-01无法访问fc-tank-02

“资源管理” –> “网络管理” -> “NetworkPolicy” -> “添加NetworkPolicy”

添加规则中，远端类型为工作负载，白名单不选择fc-tank-01





登录node节点，再次在容器中访问fc-tank-02。可以看到已经无法访问fc-tank-02了

```
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/# curl http://172.16.0.8:80
curl: (7) Failed to connect to 172.16.0.8 port 80: Connection timed out
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
```

# 4 打卡截图

网络策略设置前

```
root@fc-tank-01-c9849f648-9j2px:/# curl http://172.16.0.8:80
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=UTF-8">
  <title>FC Tank - a coffeescript rewritten of classic 90 tank on Nintendo Family Computer</title>
  <script type="text/javascript" src="js/coffeescript-v1.6.2.min.js"></script>
  <script type='text/javascript' src='js/jquery-v1.9.1.min.js'></script>
  <script type='text/javascript' src='js/lodash-v1.2.1.min.js'></script>
  <script type='text/javascript' src="js/kinetic-v4.5.1.min.js"></script>
  <script type='text/javascript' src="js/howler-v1.1.5.min.js"></script>
  <style type='text/css'>
    #tank_sprite { display: none; }
    #canvas {
      background: #000;
      width: 600px;
      height: 520px;
    }
    #tv_wrapper {
      width: 600px;
      height: 520px;
      padding: 192px 363px 210px 237px;
```

网络策略设置后

```
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/# curl http://172.16.0.8:80
curl: (7) Failed to connect to 172.16.0.8 port 80: Connection timed out
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
root@fc-tank-01-c9849f648-9j2px:/#
```