## 什么是Service？

Service可以简单的理解为逻辑上的一组Pod。一种可以访问Pod的策略，而且其他Pod可以通过这个Service访问到这个Service代理的Pod。相对于Pod而言，它会有一个固定的名称，一旦创建就固定不变。

## 创建一个Service

# cat nginx-deploy.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

labels:

app: nginx

name: nginx

namespace: default

spec:

progressDeadlineSeconds: 600

replicas: 2

revisionHistoryLimit: 10

selector:

matchLabels:

app: nginx

strategy:

rollingUpdate:

maxSurge: 25%

maxUnavailable: 25%

type: RollingUpdate

template:

metadata:

creationTimestamp: null

labels:

app: nginx

spec:

containers:

- image: nginx:1.15.2

imagePullPolicy: IfNotPresent

name: nginx

resources: {}

terminationMessagePath: /dev/termination-log

terminationMessagePolicy: File

terminationMessagePath: /dev/termination-log

terminationMessagePolicy: File

dnsPolicy: ClusterFirst

restartPolicy: Always

schedulerName: default-scheduler

securityContext: {}

terminationGracePeriodSeconds: 30

# cat nginx-svc.yaml

apiVersion: v1

kind: Service

metadata:

labels:

app: nginx-svc

name: nginx-svc

spec:

ports:

- name: http # Service端口的名称

port: 80 # Service自己的端口, servicea --> serviceb http://serviceb, http://serviceb:8080

protocol: TCP # UDP TCP SCTP default: TCP

targetPort: 80 # 后端应用的端口

- name: https

port: 443

protocol: TCP

targetPort: 443

selector:

app: nginx

sessionAffinity: None

type: ClusterIP

## 使用Service代理k8s外部应用

使用场景：

* 希望在生产环境中使用某个固定的名称而非IP地址进行访问外部的中间件服务
* 希望Service指向另一个Namespace中或其他集群中的服务
* 某个项目正在迁移至k8s集群，但是一部分服务仍然在集群外部，此时可以使用service代理至k8s集群外部的服务

# cat nginx-svc-external.yaml

apiVersion: v1

kind: Service

metadata:

labels:

app: nginx-svc-external

name: nginx-svc-external

spec:

ports:

- name: http # Service端口的名称

port: 80 # Service自己的端口, servicea --> serviceb http://serviceb, http://serviceb:8080

protocol: TCP # UDP TCP SCTP default: TCP

targetPort: 80 # 后端应用的端口

sessionAffinity: None

type: ClusterIP

# cat nginx-ep-external.yaml

apiVersion: v1

kind: Endpoints

metadata:

labels:

app: nginx-svc-external

name: nginx-svc-external

namespace: default

subsets:

- addresses:

- ip: 140.205.94.189

ports:

- name: http

port: 80

protocol: TCP

## 使用Service反代域名

# cat nginx-externalName.yaml

apiVersion: v1

kind: Service

metadata:

labels:

app: nginx-externalname

name: nginx-externalname

spec:

type: ExternalName

externalName: www.baidu.com

## Service类型

* ClusterIP：在集群内部使用，也是默认值。
* ExternalName：通过返回定义的CNAME别名。
* NodePort：在所有安装了kube-proxy的节点上打开一个端口，此端口可以代理至后端Pod，然后集群外部可以使用节点的IP地址和NodePort的端口号访问到集群Pod的服务。NodePort端口范围默认是30000-32767。
* LoadBalancer：使用云提供商的负载均衡器公开服务。