

# k8s集群核心概念Service

---

## 一、场景

---

通过前面课程的学习，播仔已经可以通过Controller创建应用，可是当我们访问应用时，发现一个问题，由于Pod的状态不是人为控制的，Pod IP是在创建时分配的，如果在Pod被误删除，被Controller重新拉起一个新的Pod时，我们发现Pod IP地址是变化着的，我们如果访问必须更换IP地址，这样对于大量的Pod运行应用来说，我们对Pod完全无法控制的，因此在k8s集群中我们引入另一个概念 Service。

## 二、学习目标

---

- ❑ 了解Service是什么
- ❑ 了解Service作用
- ❑ 了解Service类型
- ❑ 了解Service参数
- ❑ 掌握Service创建方法
- ❑ 了解Service删除方法

## 三、学习步骤

---

序号	步骤	备注
1	Service介绍	
2	Service作用	
3	Service类型	
4	Service参数	
5	Service创建方法	
6	Service删除方法	

## 四、课程内容

---

### 4.1 Service介绍

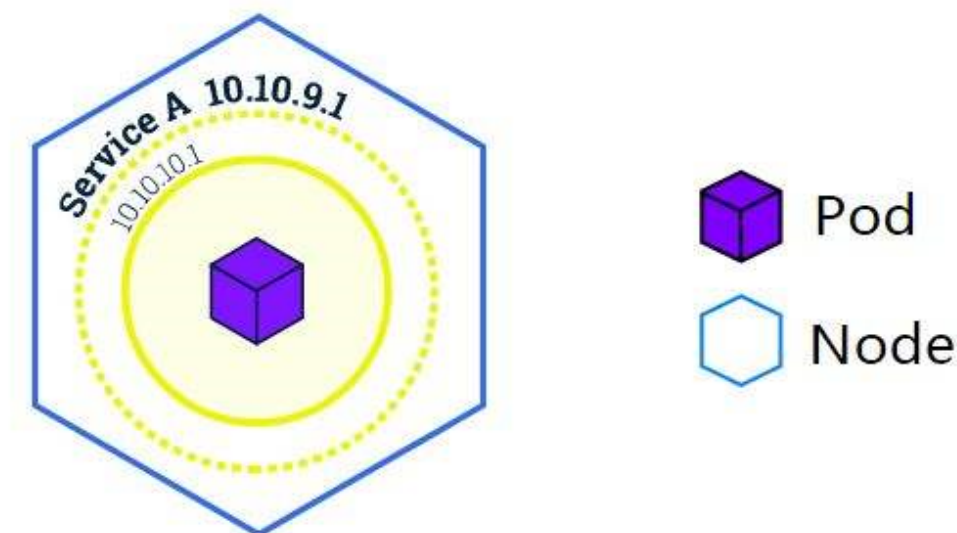
---

- 不是实体服务
- 是一条iptables或ipvs的转发规则

### 4.2 Service作用

---

- 通过Service为pod客户端提供访问pod方法，即客户端访问pod入口
- Service通过Pod标签与Pod进行关联



## 4.3 Service类型

---

- ClusterIP
  - 默认，分配一个集群内部可以访问的虚拟IP
- NodePort
  - 在每个Node上分配一个端口作为外部访问入口
- LoadBalancer
  - 工作在特定的Cloud Provider上，例如Google Cloud，AWS，OpenStack

- ExternalName
  - 表示把集群外部的服务引入到集群内部中来，即实现了集群内部pod和集群外部的服务进行通信

## 4.4 Service参数

---

- port 访问service使用的端口
- targetPort Pod中容器端口
- NodePort 通过Node实现外网用户访问k8s集群内service (30000-32767)

## 4.5 Service创建

---

Service的创建在工作中有两种方式，一是命令行创建，二是通过资源清单文件YAML文件创建

### 4.5.1 通过命令行创建

默认创建Service为ClusterIP类型

#### 4.5.1.1 创建Deployment类型应用

```
1 命令
2 [root@master1 ~]# kubectl run nginx-app --
  image=nginx:latest --image-pull-
  policy=IfNotPresent --replicas=1
```

```
1 输出
2 deployment.apps/nginx-app created
```

### 4.5.1.2 验证Deployment类型应用创建情况

```
1 命令
2 [root@master1 ~]# kubectl get deployment.apps
```

```
1 输出
2 NAME                READY    UP-TO-DATE    AVAILABLE
  AGE
3 nginx-app           1/1      1              1
  71s
```

### 4.5.1.3 创建Service与Deployment类型应用关联

```
1 命令
2 创建service
3 [root@master1 ~]# kubectl expose
  deployment.apps nginx-app --type=ClusterIP --
  target-port=80 --port=80
```

```
1 输出
2 service/nginx-app exposed
```

```
1 说明
2 expose 创建service
3 deployment.apps 控制器类型
4 nginx-app 应用名称，也是service名称
5 --type=ClusterIP 指定service类型
6 --target-port=80 指定Pod中容器端口
7 --port=80 指定service端口
```

#### 4.5.1.4 访问Service以实现访问Pod目的

```
1 命令
2 查看Service创建情况
3 [root@master1 ~]# kubectl get service
```

```
1 输出
2  NAME                TYPE                CLUSTER-IP
   EXTERNAL-IP    PORT(S)    AGE
3  nginx-app        ClusterIP    10.109.21.171
   <none>          80/TCP      5m15s
```

```
1 命令
2 访问Service
3 [root@master1 ~]# curl http://10.109.21.171
```

```
1 输出
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <title>welcome to nginx!</title>
6 <style>
7     body {
8         width: 35em;
9         margin: 0 auto;
10        font-family: Tahoma, Verdana, Arial,
    sans-serif;
11    }
12 </style>
13 </head>
14 <body>
15 <h1>welcome to nginx!</h1>
16 <p>If you see this page, the nginx web
    server is successfully installed and
```

```
17 working. Further configuration is required.
   </p>
18
19 <p>For online documentation and support
   please refer to
20 <a href="http://nginx.org/">nginx.org</a>.
   <br/>
21 Commercial support is available at
22 <a href="http://nginx.com/">nginx.com</a>.
   </p>
23
24 <p><em>Thank you for using nginx.</em></p>
25 </body>
26 </html>
```

## 4.5.2 通过资源清单YAML文件创建

### 4.5.2.1 编写用于创建Service资源清单文件

```
1 [root@master1 ~]# cat create-deployment-
   service-clusterip.yaml
2 ---
3 apiVersion: apps/v1
4 kind: Deployment
5 metadata:
6   name: nginx-apps
```



```
7   labels:
8     app: nginx
9 spec:
10  replicas: 1
11  selector:
12    matchLabels:
13      app: nginx
14  template:
15    metadata:
16      labels:
17        app: nginx
18    spec:
19      containers:
20        - name: nginxapps
21          image: nginx:latest
22          imagePullPolicy: IfNotPresent
23          ports:
24            - containerPort: 80
25  ---
26 apiVersion: v1
27 kind: Service
28 metadata:
29   name: nginx-apps
30 spec:
31   type: ClusterIP
32   selector:
33     app: nginx
34   ports:
35     - protocol: TCP
36       port: 80
37       targetPort: 80
```

### 4.5.2.2 应用用于创建Service资源清单文件

```
1 命令
2 应用create-deployment-service-clusterip.yaml
3 [root@master1 ~]# kubectl apply -f create-
  deployment-service-clusterip.yaml
```

```
1 输出
2 deployment.apps/nginx-apps created
3 service/nginx-apps created
```

### 4.5.2.3 验证资源对象创建结果

```
1 命令
2 查看deployment.apps
3 [root@master1 ~]# kubectl get deployment.apps
```

```
1 输出
2  NAME          READY    UP-TO-DATE    AVAILABLE
   AGE
3  nginx-apps    1/1      1              1
   48s
```

```
1 命令
2 [root@master1 ~]# kubectl get service
```

```
1 输出
2  NAME          TYPE          CLUSTER-IP
   EXTERNAL-IP    PORT(S)       AGE
3  nginx-apps    ClusterIP     10.105.7.229
   <none>         80/TCP        2m22s
```

```
1 命令
2 [root@master1 ~]# curl http://10.105.7.229
```

```
1 输出
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <title>welcome to nginx!</title>
6 <style>
7     body {
8         width: 35em;
```

```
9         margin: 0 auto;
10         font-family: Tahoma, Verdana, Arial,
            sans-serif;
11     }
12 </style>
13 </head>
14 <body>
15 <h1>welcome to nginx!</h1>
16 <p>If you see this page, the nginx web
            server is successfully installed and
17 working. Further configuration is required.
            </p>
18
19 <p>For online documentation and support
            please refer to
20 <a href="http://nginx.org/">nginx.org</a>.
            <br/>
21 Commercial support is available at
22 <a href="http://nginx.com/">nginx.com</a>.
            </p>
23
24 <p><em>Thank you for using nginx.</em></p>
25 </body>
26 </html>
```

## 4.5.3 基于资源清单YAML文件创建 NodePort类型Service

### 4.5.3.1 编写资源清单文件

```
1 [root@master1 ~]# cat create-deployment-  
  service-nodeport.yaml  
2 ---  
3 apiVersion: apps/v1  
4 kind: Deployment  
5 metadata:  
6   name: nginx-app-nodeport  
7   labels:  
8     app: nginx-app  
9 spec:  
10  replicas: 1  
11  selector:  
12    matchLabels:  
13      app: nginx-app  
14  template:  
15    metadata:  
16      labels:  
17        app: nginx-app4  
18    spec:  
19      containers:  
20        - name: nginx  
21          image: nginx:latest  
22          imagePullPolicy: IfNotPresent  
23          ports:  
24            - containerPort: 80  
25 ---  
26 apiVersion: v1  
27 kind: Service  
28 metadata:  
29   name: nginx-app  
30 spec:
```

```
31 selector:
32     app: nginx-app
33 type: NodePort
34 ports:
35 - protocol: TCP
36   nodePort: 30001
37   port: 80
38   targetPort: 80
```

### 4.5.3.2 应用资源清单文件

```
1 命令
2 应用 create-deployment-service-nodeport.yaml
3 [root@master1 ~]# kubectl apply -f create-
  deployment-service-nodeport.yaml
```

```
1 输出
2 deployment.apps/nginx-app-nodeport created
3 service/nginx-app created
```

### 4.5.3.3 验证资源对象创建情况

```
1 命令
2 查看service创建
3 [root@master1 ~]# kubectl get service
```

```
1 输出
2 NAME                TYPE          CLUSTER-IP
   EXTERNAL-IP      PORT(S)            AGE
3 nginx-app           NodePort       10.98.196.211
   <none>            80:30001/TCP      26s
```

```
1 命令
2 查看k8s集群所有主机30001端口侦听状态
3 [root@xxx ~]# ss -anput | grep ":30001"
```

```
1 输出
2 tcp        LISTEN      0          128        [::]:30001
           [::]:*
           users:
   (("kube-proxy",pid=4164,fd=21))
```

#### 4.5.3.4 在k8s集群外访问Service

在物理机浏览器中访问即可验证。



## 4.6 删除Service

### 4.6.1 通过命令行删除

```
1 命令
2 查看service
3 [root@master1 ~]# kubectl get service
```

```
1 输出
2 NAME          TYPE          CLUSTER-IP
   EXTERNAL-IP  PORT(S)      AGE
3 nginx-app     ClusterIP    10.109.21.171
   <none>       80/TCP       18m
```



```
1 命令
2 删除service
3 [root@master1 ~]# kubectl delete service
  nginx-app
```

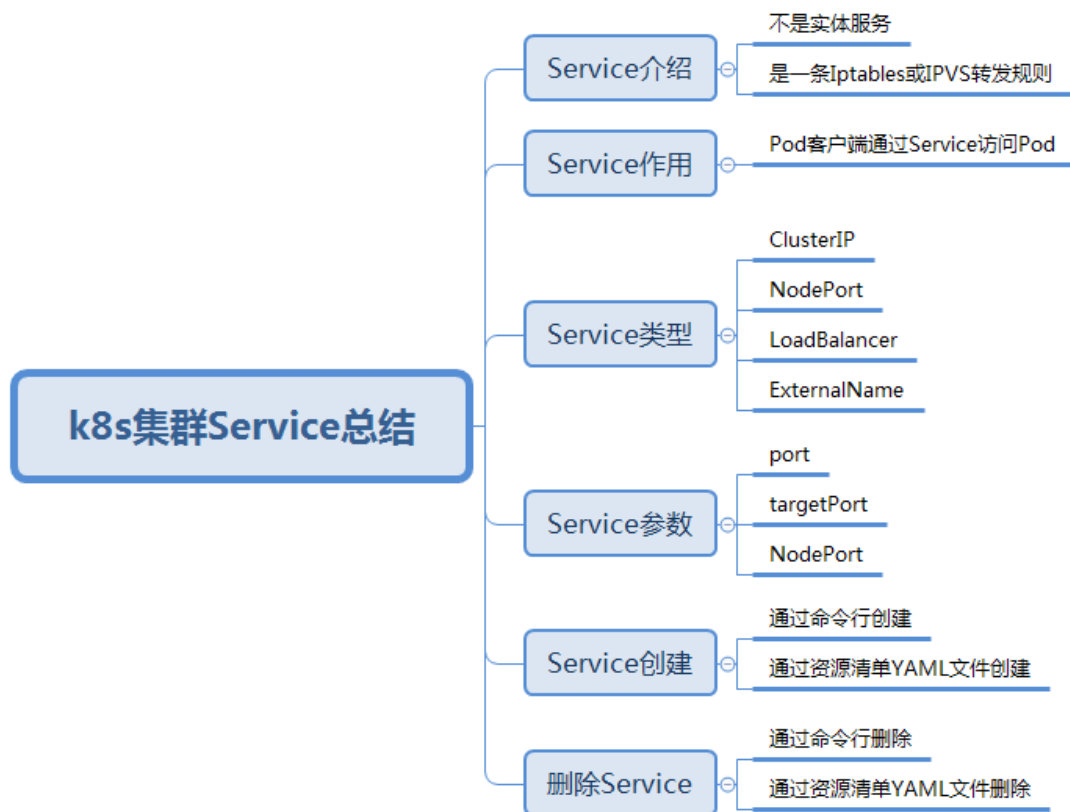
```
1 输出
2 service "nginx-app" deleted
```

## 4.6.2 通过资源清单YAML文件删除

```
1 命令
2 应用create-deployment-service-clusterip.yaml
3 [root@master1 ~]# kubectl delete -f create-
  deployment-service-clusterip.yaml
```

```
1 输出
2 deployment.apps "nginx-apps" deleted
3 service "nginx-apps" deleted
```

## 五、学习总结



## 六、课程预约

深入学习Service，可以预约《kubernetes集群从入门到企业应用实战》相关课程。