

使用 service 访问群集中的应用程序

本文向您展示如何创建 Kubernetes Service 对象，外部客户端可以使用它来访问集群中运行的应用程序。该 Service 可以为具有两个运行实例的应用程序提供负载均衡。

目的

- 运行 Hello World 应用程序的两个实例。
- 创建一个暴露 node 节点端口的 Service 对象。
- 使用 Service 对象访问正在运行的应用程序。

为在两个 pod 中运行的应用程序创建 service

1. 在集群中运行 Hello World 应用程序：

```
kubectl run hello-world --replicas=2 --labels="run=load-balancer-example" --image=gcr.io/google-samples/node-hello:1.0 --port=8080
```

上述命令创建一个 [Deployment](#) 对象和一个相关联的 [ReplicaSet](#) 对象。该 ReplicaSet 有两个 [Pod](#)，每个 Pod 中都运行一个 Hello World 应用程序。

2. 显示关于该 Deployment 的信息：

```
kubectl get deployments hello-world  
kubectl describe deployments hello-world
```

3. 显示 ReplicaSet 的信息：

```
kubectl get replicaset  
kubectl describe replicaset
```

4. 创建一个暴露该 Deployment 的 Service 对象：

```
kubectl expose deployment hello-world --type=NodePort --  
name=example-service
```

5. 显示该 Service 的信息：

```
kubectl describe services example-service
```

输出类似于：

Name:	example-service
Namespace:	default
Labels:	run=load-balancer-example
Annotations:	<none>
Selector:	run=load-balancer-example
Type:	NodePort
IP:	10.254.216.190
Port:	<unset> 8080/TCP
TargetPort:	8080/TCP
NodePort:	<unset> 31688/TCP
Endpoints:	172.30.72.3:8080,172.30.96.2:8080
Session Affinity:	None
External Traffic Policy:	Cluster
Events:	<none>

记下服务的 `NodePort` 值。例如，在前面的输出中，`NodePort` 值为 31496。

6. 列出运行 Hello World 应用程序的

Pod：

```
kubectl get pods --selector="run=load-balancer-example" --output=wide
```

输出类似于：

NAME	READY	STATUS	...
hello-world-86cddf59c5-jn2kc	1/1	Running	...
172.30.96.2	192.168.251.103		
hello-world-86cddf59d5-rn6t6	1/1	Running	...
172.30.72.3	192.168.251.101		

7. 获取正在运行 Hello World 应用程序的 Pod 的其中一个节点的 public IP 地址。如何得到这个地址取决于您的集群设置。例如，如果您使用 Minikube，可以通过运行 `kubectl cluster-info` 查看节点地址。如果您是使用 Google Compute Engine 实例，可以使用 `gcloud compute instances list` 命令查看您的公共地址节点。

8. 在您选择的节点上，在您的节点端口上例如创建允许 TCP 流量的防火墙规则，如果您的服务 `NodePort` 值为 31568，创建防火墙规则，允许端口 31568 上的 TCP 流量。

9. 使用节点地址和节点端口访问 Hello World 应用程序：

```
curl http://<public-node-ip>:<node-port>
```

其中 `<public-node-ip>` 是您节点的 public IP 地址，而 `<node-port>` 是您服务的 NodePort 值。

对成功请求的响应是一个 hello 消息：

```
Hello Kubernetes!
```

使用 Service 配置文件

作为使用 `kubectl expose` 的替代方法，您可以使用 [service 配置文件](#) 来创建 Service。

要删除 Service，输入以下命令：

```
kubectl delete services example-service
```

删除 Deployment、ReplicaSet 和正运行在 Pod 中的 Hello World 应用程序，输入以下命令：

```
kubectl delete deployment hello-world
```

了解更多关于 [使用 service 连接应用程序](#)。