Practical operation Namespace

Knowledge

命名空间(Namespace)是一种资源隔离机制，将同一集群中的资源划分为相互隔离的组。

命名空间可以在多个用户之间划分集群资源（通过资源配额）。

- 例如我们可以设置**开发、测试、生产**等多个命名空间。

同一命名空间内的资源名称要唯一，但跨命名空间时没有这个要求。

命名空间作用域仅针对带有名字空间的对象，例如 Deployment、Service 等。

这种作用域对集群访问的对象不适用，例如 StorageClass、Node、PersistentVolume 等。

Kubernetes 会创建四个初始命名空间：

- `**default**` 默认的命名空间，不可删除，未指定命名空间的对象都会被分配到default中。
- `**kube-system**` Kubernetes 系统对象(控制平面和Node组件)所使用的命名空间。
- `**kube-public**` 自动创建的公共命名空间，所有用户（包括未经过身份验证的用户）都可以读取它。通常我们约定，将整个集群中公用的可见和可读的资源放在这个空间中。
- `**kube-node-lease**` 租约（Lease）对象使用的命名空间。每个节点都有一个关联的 lease 对象，lease 是一种轻量级资源。lease对象通过发送心跳，检测集群中的每个节点是否发生故障。

使用`kubectl get lease -A`查看`lease`对象

使用多个命名空间

- 命名空间是在多个用户之间划分集群资源的一种方法（通过资源配额）。
- 例如我们可以设置**开发、测试、生产**等多个命名空间。
- 不必使用多个命名空间来分隔轻微不同的资源。
- 例如同一软件的不同版本： 应该使用标签 来区分同一命名空间中的不同资源。
- 命名空间适用于跨多个团队或项目的场景。
- 对于只有几到几十个用户的集群，可以不用创建命名空间。
- 命名空间不能相互嵌套，每个 Kubernetes 资源只能在一个命名空间中。

Practical Operation

```
# 创建命名空间
controlplane $ kubectl create namespace dev
namespace/dev created
# 查看命名空间
controlplane $ kubectl get ns
NAME                   STATUS   AGE
default                Active   13h
dev                    Active   4s
kube-node-lease        Active   13h
kube-public            Active   13h
kube-system            Active   13h
local-path-storage     Active   12h
```

```
# 在命名空间里创建运行pod
controlplane $ kubectl  run nginx --image=nginx --namespace=dev
pod/nginx created
controlplane $ kubectl run my-nginx --image=nginx -n=dev
pod/my-nginx created
# 查看命名空间内的pods
controlplane $ kubectl get pods -n dev
NAME       READY   STATUS    RESTARTS   AGE
my-nginx   1/1     Running   0          29s
nginx      1/1     Running   0          64s
# 查看集群所有信息
controlplane $ kubectl get all
NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   13h
# 删除命名空间（-- 同时会删除所有的命名空间下的pod）
controlplane $ kubectl delete ns dev
namespace "dev" deleted
controlplane $ kubectl get all
NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   13h
# 创建命名空间dev
controlplane $ kubectl create namespace dev
namespace/dev created
# 创建运行命名空间 pod
controlplane $ kubectl run nginx --image nginx --namespace dev
pod/nginx created
# 查看命名空间pod
controlplane $ kubectl get pod -n dev
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          10s
# 查看当前上下文
controlplane $ kubectl config current-context
kubernetes-admin@kubernetes
controlplane $
# 修改默认命名空间为dev
controlplane $ kubectl config set-context $(kubectl config current-context) --
namespace dev
Context "kubernetes-admin@kubernetes" modified.
controlplane $ kubectl config current-context
kubernetes-admin@kubernetes
# 测试运行命名空间创建不加-n
controlplane $ kubectl run my-nginx --image nginx
pod/my-nginx created
# 查看生效
controlplane $ kubectl get all
NAME           READY   STATUS    RESTARTS   AGE
pod/my-nginx   1/1     Running   0          17s
pod/nginx      1/1     Running   0          100s
# 删除所有dev 下的命名空间 pod
controlplane $ kubectl delete ns dev
namespace "dev" deleted
controlplane $ kubectl get all
No resources found in dev namespace.
controlplane $
# 还原命名空间为默认
controlplane $ kubectl config set-context $(kubectl config current-context) --
namespace default
Context "kubernetes-admin@kubernetes" modified.
```

```
controlplane $
# 查看结果恢复原样
controlplane $ kubectl get all
NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   13h
controlplane $
# END
```