Namespace

Nos permite separar nuestro cluster físico en cluster virtuales separados de forma lógica. Los namespace no permiten agrupar recursos(Pods, servicios, deployments, etc.) por espacios de nombres. De esta forma podemos tener diferentes aplicaciones, usuarios, etc. trabajando en el mismo cluster de kubernetes pero trabajando en espacios de nombre separados.

Un ejemplo de uso de namespace sería dividir nuestro cluster físico en dos cluster lógicos, uno de pre y otro de pro

Kubernetes arranca con tres espacios de nombres inicialmente:

- default El espacio de nombres por defecto para aquellos objetos que no especifican ningún espacio de nombres
- kube-system El espacio de nombres para aquellos objetos creados por el sistema de Kubernetes
- kube-public Este espacio de nombres se crea de forma automática y es legible por todos los usuarios (incluyendo aquellos no autenticados)

```
apiVersion: v1
kind: Namespace
metadata:
  name: development
  labels:
    name: desarrollo
```

Comandos

Crear un namespace

```
kubectl create namespace <nombrenamespace>
```

```
kubectl get namespaces
```

ver lo objetos de un namespace

```
kubectl get all -n <nombrenamespace>
```

Establecer un namespace como predeterminado

```
kubectl config set-context --current —namespace=default
```

Especificar el namespace para un pod

```
kubectl apply -f pod.yaml --namespace=<nombre_namespace>
```

Ejemplo

https://github.com/ricardoandre97/k8s-resources/blob/master/namespaces/envs-ns.yaml

```
---
apiVersion: v1
kind: Namespace
metadata:
  name: dev               #crea un namespace de nombre dev y con la etiqueta
dev
  labels:
    name: dev
---
apiVersion: v1
kind: Namespace
metadata:
  name: prod           #crea un namespace de nombre prod y con la etiqueta
prod
  labels:
    name: prod
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment-dev
  namespace: dev           # indica en que namespace debe de crear el
deployment
  labels:
    app: front
spec:
  replicas: 1
  selector:
    matchLabels:
      app: front
  template:
    metadata:
      labels:
        app: front
    spec:
      containers:
      - name: nginx
        image: nginx:alpine
---
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment-prod
  namespace: prod                          # indica en que namespace debe de
crear el deployment
  labels:
    app: back
spec:
  replicas: 5
  selector:
    matchLabels:
      app: back
  template:
    metadata:
      labels:
        app: back
    spec:
      containers:
      - name: nginx
        image: nginx:alpine
```

Referencias

- https://kubernetes.io/es/docs/concepts/overview/working-with-objects/namespaces/

From:
https://wiki.intrusos.info/ - **LCWIKI**

Permanent link:
https://wiki.intrusos.info/doku.php?id=virtualizacion:kubernetes:namespace

Last update: **182023/01/ 13:37**