

Lesson 10

Introduction to Pod, Namespace Label, Service

1. Các lệnh thường sử dụng

2. Các thành phần trong file YAML

3. Tìm hiểu và thực hành Pod

1. Các câu lệnh thường sử dụng

`kubectl get <resource>`: Lấy danh sách các tài nguyên. Ví dụ: `kubectl get pods`.

`kubectl describe <resource> <name>`: Lấy thông tin chi tiết về một tài nguyên. Ví dụ: `kubectl describe pod my-pod`.

`kubectl create <resource> <name>`: Tạo một tài nguyên mới. Ví dụ: `kubectl create deployment my-deployment --image=my-image`.

1. Các câu lệnh thường sử dụng

`kubectl apply -f <filename>`: Áp dụng một tệp YAML hoặc JSON để tạo hoặc cập nhật các tài nguyên trong Kubernetes.

`kubectl delete <resource> <name>`: Xóa một tài nguyên. Ví dụ: `kubectl delete pod my-pod`.

`kubectl logs <pod>`: Lấy logs của một pod. Ví dụ: `kubectl logs my-pod`.

`kubectl exec <pod> <command>`: Chạy một command trong container của một pod. Ví dụ:
`kubectl exec my-pod -- ls /app`.

1. Các câu lệnh thường sử dụng

`kubectl port-forward <pod> <local-port>:<pod-port>`: Chuyển tiếp traffic từ một cổng trên máy local đến một cổng trên pod. Ví dụ: `kubectl port-forward my-pod 8080:80`.

`kubectl scale <resource> <name> --replicas=<number>`: Thay đổi số lượng replicas của một tài nguyên. Ví dụ: `kubectl scale deployment my-deployment --replicas=3`.

`kubectl rollout status <resource> <name>`: Kiểm tra trạng thái của một rollout. Ví dụ:
`kubectl rollout status deployment/my-deployment`.

2. Các thành phần trong file YAML

apiVersion: Định nghĩa phiên bản của API Kubernetes được sử dụng để triển khai đối tượng. Ví dụ: "apiVersion: v1" sẽ sử dụng phiên bản API v1 của Kubernetes.

kind: Định nghĩa loại của đối tượng Kubernetes được tạo ra. Các loại đối tượng thông dụng trong Kubernetes bao gồm Deployment, Service, Pod, ConfigMap, Secret, và PersistentVolumeClaim.

metadata: Định nghĩa các thông tin về đối tượng, bao gồm tên đối tượng, label, annotations và thông tin khác như tác giả, phiên bản, và mô tả.

spec: Định nghĩa thông tin chi tiết về đối tượng, bao gồm thông tin về cấu hình, khai báo, và các thông tin khác tùy thuộc vào loại đối tượng sử dụng.

2. Các thành phần trong file YAML

```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  containers:
    - name: container1
      image: nginx
      ports:
        - containerPort: 80
    - name: container2
      image: busybox
      command:
        [
          "/bin/sh",
          "-c",
          "while true; do echo Hello from Container 2; sleep 10; done",
        ]
```

```
apiVersion: v1
kind: Service
metadata:
  name: my-service
spec:
  selector:
    app: my-app
  ports:
    - name: http
      port: 80
      targetPort: 8080
  type: ClusterIP
```

3. Tìm hiểu về Pod ?

Pod là một khái niệm quan trọng trong Kubernetes và là đơn vị nhỏ nhất để triển khai và quản lý các Application trên Kubernetes. Pod có thể Run một hoặc nhiều Container bên trong và cùng chia sẻ Resource như Network, Storage,...

