What is Service Discovery in Kubernetes?

Kubernetes assigns **dynamic IPs** to pods, so you can't rely on fixed IPs to communicate. Instead, Kubernetes provides a **Service**, which gives a **stable DNS name and IP address** that routes traffic to the correct pod(s).

Service Discovery in Kubernetes

Goal:

- Deploy two apps:
  - **App A (frontend)** that calls
  - **App B (backend)** using **Service Discovery**.

Step 1: Create Backend App (App B)

backend-pod.yaml

```
yaml
---------
apiVersion: v1
kind: Pod
metadata:
  name: backend
  labels:
    app: backend
spec:
  containers:
    - name: backend
      image: hashicorp/http-echo
      args:
        - "-text=Hello from Backend"
      ports:
        - containerPort: 5678
```

Deploy it:

```
bash
---------
kubectl apply -f backend-pod.yaml
```

Step 2: Create a Service for Backend

backend-service.yaml

```
yaml
---------
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  selector:
    app: backend
  ports:
    - protocol: TCP
      port: 80
      targetPort: 5678
```

Deploy it:

```
bash
---------
kubectl apply -f backend-service.yaml
```

Now, Kubernetes gives a **stable DNS name**:

```
text
---------
http://backend-service.default.svc.cluster.local
```

> Any pod in the same namespace can reach the backend using this name.

Step 3: Create Frontend App (App A)

We'll use a simple busybox pod to **curl the backend** using service name.

frontend-pod.yaml

```
yaml
---------
apiVersion: v1
kind: Pod
metadata:
  name: frontend
spec:
  containers:
    - name: frontend
      image: busybox
      command: ['sh', '-c', 'while true; do sleep 3600; done']
```

Deploy it:

```
bash
---------
kubectl apply -f frontend-pod.yaml
```

Step 4: Test Service Discovery

Get a shell inside the frontend pod:

```
bash
---------
kubectl exec -it frontend -- sh
```

Now curl the backend:

```
bash
---------
wget -qO- http://backend-service
```

You should see:

```
text
---------
Hello from Backend
```

This confirms **Service Discovery** is working via the **DNS name** `backend-service`.

🔁 Summary

Component	Purpose
Pod (backend)	The actual service provider
Service	Provides stable name & IP
DNS Name	backend-service.default.svc...
Pod (frontend)	Client that calls the service

What is the primary purpose of Service Discovery in Kubernetes?

A. To create new pods dynamically
B. To automatically expose pods to the internet
C. To allow pods to communicate without hardcoding IPs
D. To scale the pods based on CPU usage

✅ C. To allow pods to communicate without hardcoding IPs

What happens if a pod is restarted and gets a new IP address?

A. Communication to it will fail
B. Service continues to route traffic to the new pod IP
C. The service must be recreated
D. DNS will not work anymore

✅ **B. Service continues to route traffic to the new pod IP**

What command can you use inside a pod to test service discovery?

A. `kubectl get pods`
B. `docker exec`
C. `wget` or `curl` using the service name
D. `kubectl describe service`

✅ **C.** `wget` **or** `curl` **using the service name**