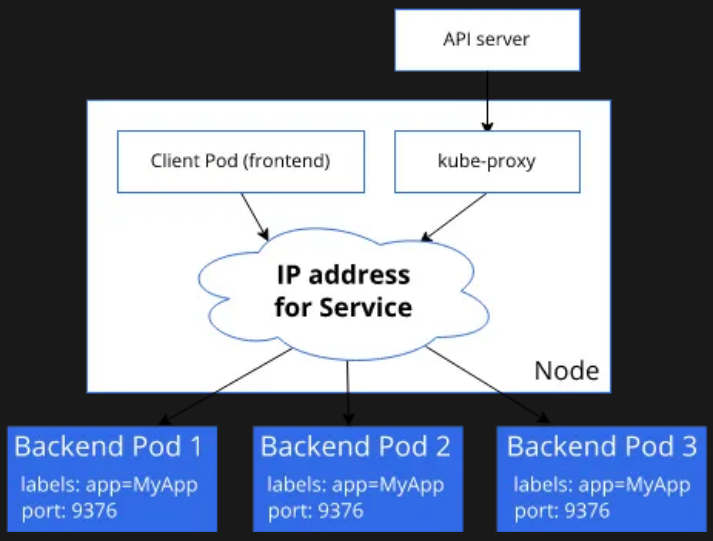
Kubernets Service

1. Kubernetes Service

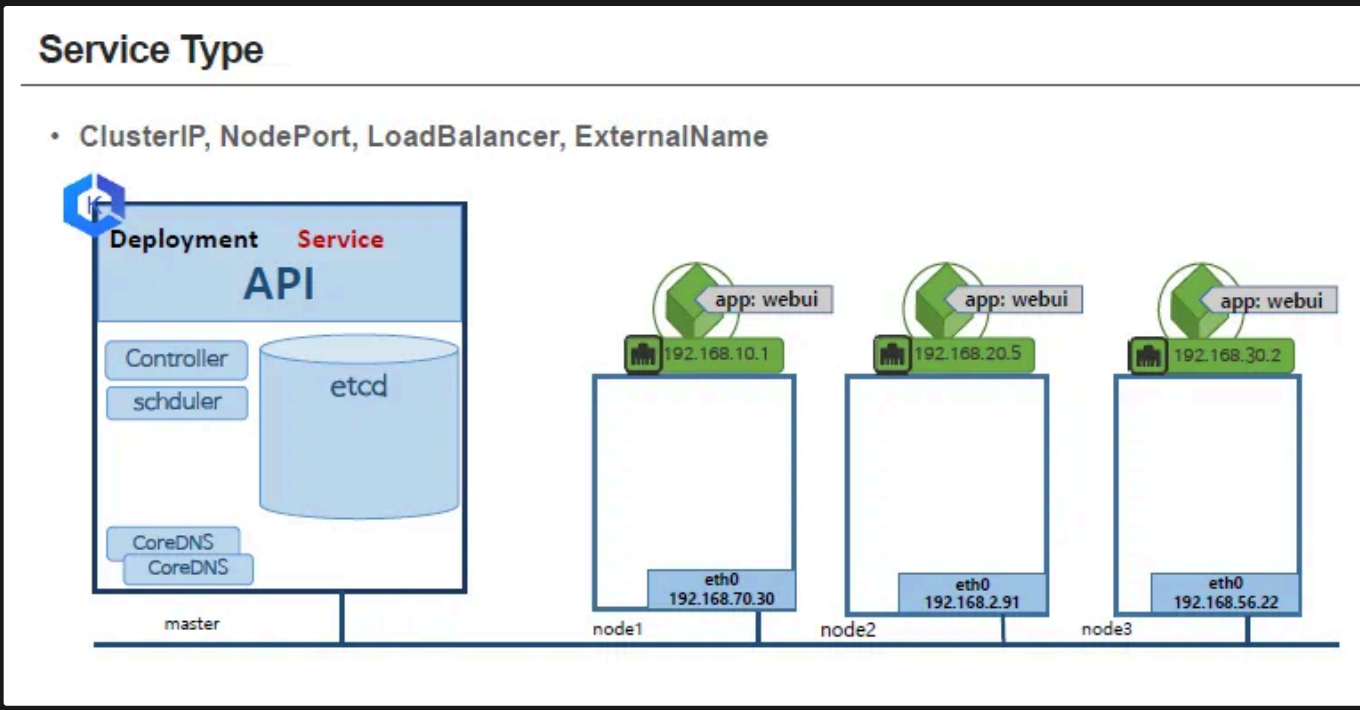
Service 동작원리

- kubernetes - kube-proxy 동작 원리



Service Type

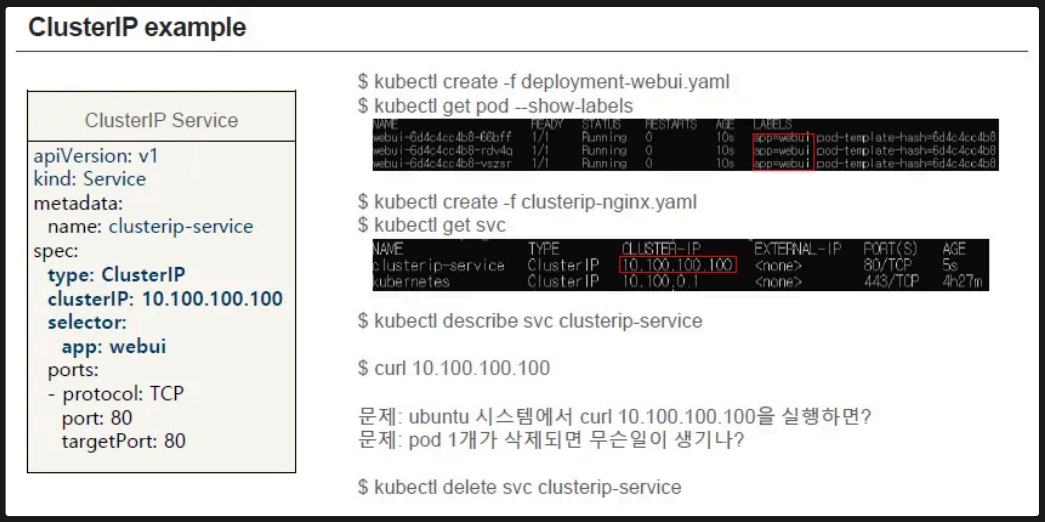
- 쿠버네티스 서비스의 4 가지 타입
  - **ClusterIP(default) : Pod 그룹(동일한 서비스를 지원하는 Pod 모음)의 단일 진입점 (Virtual IP:LB) 생성**
  - **NodePort : ClusterIP 가 생성된 후 모든 Worker Node 에 외부에서 접속 가능 한 포트가 예약**
  - LoadBalancer : 클라우드 인프라스트럭처 (AWS, Azure, GCP)에 적용
    LoadBalancer를 자동으로 프로 비전하는 기능 지원
  - ExternalName : 클러스터 안에서 외부에 접속 시 사용할 도메인을 등록해서 사용
    클러스터 도메인이 실제 외부 도메인으로 치환되어 동작



ClusterIP 타입의 서비스 운영

- 동일한 서비스를 제공하는 Pod 그룹에 단일 진입점
  - deployment name: `web-ui` , image: `nginx` , port: `80` , replicas: `2`
  - service name: `web-ui` , type: `clusterIP` , port: `80`

```
# CLI kubectl create deployment web-ui --image=public.ecr.aws/nginx/nginx:1.26 --port=80 --replicas=2 kubectl expose deployment web-ui --type ClusterIP --port 80 --target-port 80 --name web-ui-svc
```

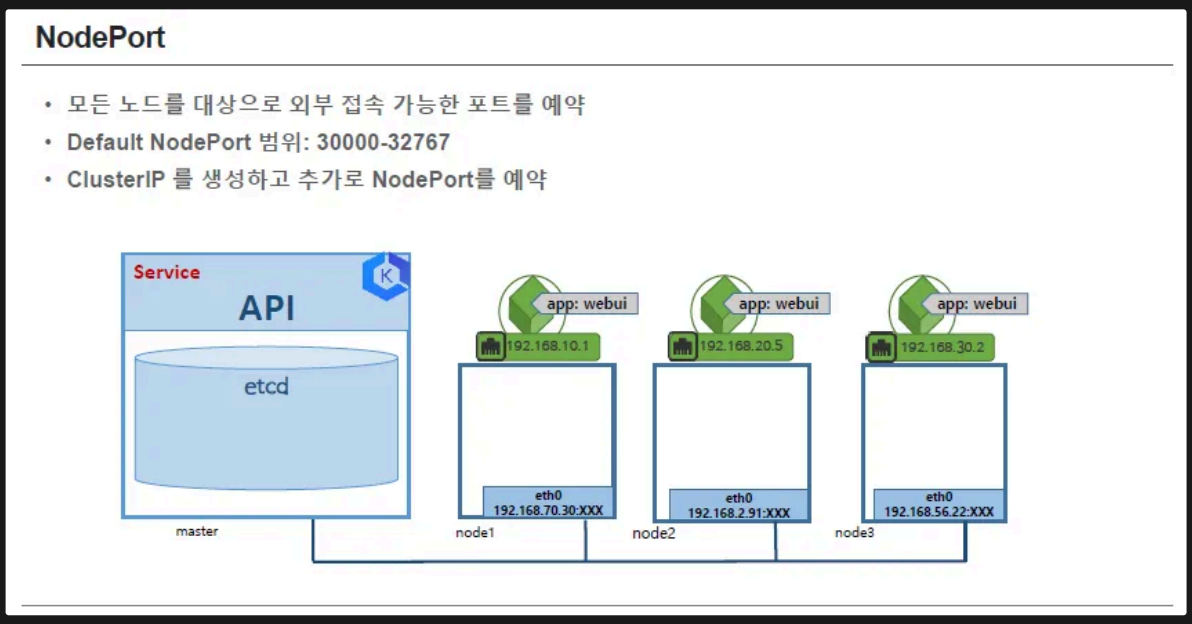



```
$ kubectl expose deployment web-ui --type=clusterIP --port 80 --target-port 80 --dry-run=client -o yaml > svc-webserver.yaml $ cat svc-webserver.yaml apiVersion: v1 kind: Service metadata: creationTimestamp: null name: webserver spec: ports: - port: 80 protocol: TCP targetPort: 80 selector: app: webserver type: ClusterIP $ kubectl apply -f svc-webserver.yaml $ kubectl get svc NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 2d webserver ClusterIP 10.107.89.60 <none> 80/TCP 36s k8s-master $ curl 10.107.89.60
```

- named port
  - 포트에 이름을 붙여서 사용
  - `web-main` 이라는 pod 생성 : `image=smlinux/appjs` , `port=8080, name: http` , `port=8443, name:https`
  - `web-main` pod에 접근할 수 있는 `web-main-svc` 생성 : `type=ClusrerIP` , `cluster port: 80` →http(8080), 443→https(8443)
  - ▶ yaml

NodePort 타입의 서비스 운영

- 외부 클라이언트에 Pod set을 노출




```
$ kubectl edit svc webserver ... name: webserver namespace: default resourceVersion: "41244"
uid: 20b512d5-82c6-4632-85e2-d551f599c7cd spec: clusterIP: 10.107.89.60 clusterIPs: -
10.107.89.60 internalTrafficPolicy: Cluster ipFamilies: - IPv4 ipFamilyPolicy: SingleStack
ports: - port: 80 protocol: TCP targetPort: 80 NodePort: 30100 selector: app: webserver
sessionAffinity: None type: NodePort ... ## NodePort $ cat svc-webserver.yaml apiVersion: v1
kind: Service metadata: name: webserver spec: type: NodePort ports: - port: 80 protocol: TCP
targetPort: 80 nodePort: 30000 selector: app: webserver
```

2. Kubernetes Service - coreDNS

coreDNS : docs 검색 키워드- "DNS"

- CoreDNS는 쿠버네티스 클러스터의 DNS 역할을 수행할 수 있는, 유연하고 확장 가능한 DNS 서버이다
- kubernetes Master 에 생성되는 add-on
- service 생성시 servicename → cluster IP Addresss 매핑 정보를 자동으로 저장
- Kubernetes service domain
  - service

    serviceName.**namespace**.svc.cluster.local

    `order-svc.default.svc.cluster.local` # default 네임스페이스에 동작중인 order-svc 서비스

    `login-svc.devops.svc.cluster.local` #devops 네임스페이스에 동작중인 login-svc 서비스
  - pod

    pod-ip-address.namespace.pod.cluster.local

    192-168-126-18.default.pod.cluster.local # web-main 이라는 파드는 192.168.126.18 사용

- 모든 pod는 내부적으로 dns 서버를 coreDNS로 정의해서 사용하고 있음
  - pod 내부 /etc/resolv.conf 파일에 등록된 DNS 서버 질의

```
kubectl create deployment login --image public.ecr.aws/nginx/nginx:1.26 --port 80 --replicas 2 kubectl get pods -o wide NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES login-6d7896c5bf-kvf2c 1/1 Running 0 22s 10.36.0.3 k8s-worker2 <none> <none> login-6d7896c5bf-kvxtk 1/1 Running 0 22s 10.44.0.14 k8s-worker1 <none> <none> kubectl get pods --show-labels login-6d7896c5bf-kvf2c 1/1 Running 0 104s app=login,pod-template-hash=6d7896c5bf login-6d7896c5bf-kvxtk 1/1 Running 0 104s app=login,pod-template-hash=6d7896c5bf # service 생성될때 coreDNS 에는 # service 등록 : login-svc.default.svc.cluser.local는 10.100.164.73 등록 # pod 등록 : 10-36-0-3.default.pod.cluster.local는 10.36.0.3 kubectl expose deployment login --name login-svc --type ClusterIP --port 80 --target-port 80 kubectl get svc NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE login-svc ClusterIP 10.100.164.73 <none> 80/TCP 17s # DNS 조회 : login-svc.default.svc.cluster.local # TEST 할수 있는 POD(busybox:1.28)만들어서 DNS 조회 kubectl run dns-test --rm --image busybox:1.28 -it -- /bin/sh / # cat /etc/resolv.conf search default.svc.cluster.local svc.cluster.local cluster.local nameserver 10.96.0.10 options ndots:5 / # nslookup login-svc.default.svc.cluster.local / # nslookup 10-36-0-3.default.pod.cluster.local / # exit Session ended, resume using 'kubectl attach dns-test -c dns-test -i -t' command when the pod is running kubectl delete deployments.apps login kubectl delete svc login-svc nslookup 10-40-0-1.default.pod.cluster.local
```

3. 기출 문제 풀이

❓ 1. Deploy and Service

- 작업 클러스터 : **k8s**
- Reconfigure the existing deployment `front-end` and add a port specification named `http` exposing port