2024.07.23

드라이브 참고하여 zoom 07.22꺼 못한거 마무리짓기

어제꺼

kubectl get configmap --namespace mymetallb
**metallb 폴더에서

vim my-config.yaml**

```
**
apiVersion: metallb.io/v1beta1
kind: IPAddressPool
metadata:
  name: my-metallb-config
  namespace: mymetallb
spec:
  addresses:
  - 192.168.56.100-192.168.56.120
  autoAssign: true
---
apiVersion: metallb.io/v1beta1
kind: L2Advertisement
metadata:
  name: my-metallb-config
  namespace: mymetallb
spec:
  ipAddressPools:
  - my-metallb-config
**
```

kubectl apply -f my-config.yaml
k get ipaddresspool.metallb.io --namespace mymetallb
k get all --namespace mynginx
k describe service/nginx-ingress-controller-1721618748 --namespace mynginx
vim ingress01-deploy.yaml

```
**apiVersion: apps/v1
kind: Deployment
metadata:
  name: ingress-deploy-test01
spec:
  replicas: 3
  selector:
    matchLabels:
      app.kubernetes.io/name: web-deploy01
  template:
    metadata:
      labels:
        app.kubernetes.io/name: web-deploy01
    spec:
      containers:
      - name: nginx
        image: nginx:1.25**
```

vim ingress01-ingress.yaml

```
**apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress-test01
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  ingressClassName: nginx
  rules:
  - http:
      paths:
      - path: /test01
        pathType: Prefix
        backend:
          service:
            name: ingress-service-test01
            port:
              number: 80**
```

vim ingress01-service.yaml

```
**apiVersion: v1
kind: Service
metadata:
  name: ingress-service-test01
spec:
  selector:
    app.kubernetes.io/name: web-deploy01
  type: ClusterIP
  ports:
  - protocol: TCP
    port: 80
    targetPort: 80
**
```

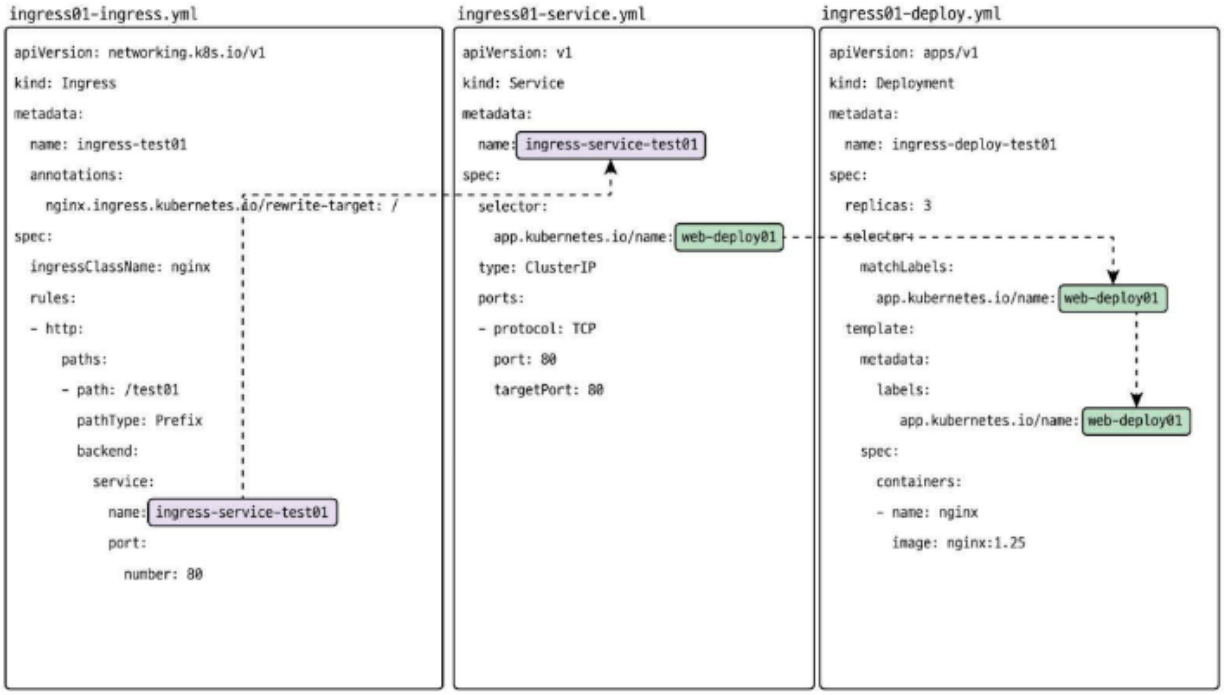
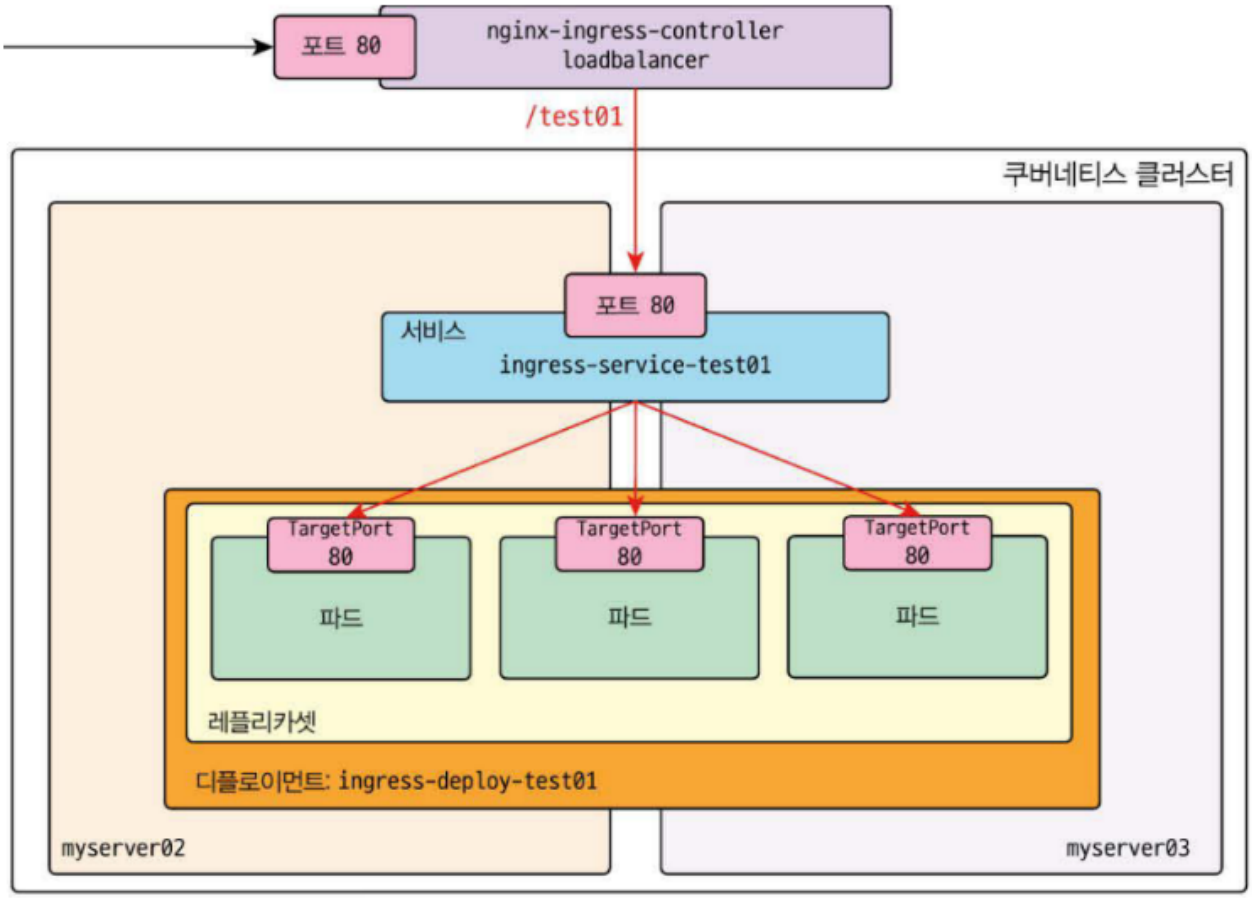


그림 9-36 인그레스 yml 구조

**k apply -f ingress01-deploy.yaml
k apply -f ingress01-service.yaml
k apply -f ingress01-ingress.yaml

- browser에서 http://192.168.56.100/test01**



mkdir metric-server && cd metric-server
helm repo add metrics-server https://kubernetes-sigs.github.io/metrics-server
helm repo update
helm search repo metric

helm pull metrics-server/metrics-server
tar xvfz metrics-server-3.12.1.tgz
cd metrics-server
cp ./values.yaml ./my-values.yaml

vim my-values.yaml
**- vim에서 찾기
명령어 -> /defaultArgs
:

- 아래 내용 추가**
- **- --kubelet-insecure-tls**
  - --kubelet-preferred-address-types=InternalIP
  **- kube-system namespace 안의 pod 보기

kubectl get all --namespace kube-system	**

helm install --namespace kube-system --generate-name metrics-server/metrics-server -f my-values.yaml
**
kubectl top nodes

% kubectl top nodes
NAME CPU(cores) CPU% MEMORY(bytes) MEMORY%
myserver01 160m 4% 1528Mi 40%
myserver02 82m 2% 1709Mi 44%
myserver03 92m 2% 2299Mi 60%

kubectl top pod --namespace kube-system
**
**cpu에서 M(Milliunits) : 1000m은 1개의 cpu를 의미

- 10m은 cpu 0.01 사용**
  **

HPA

kubectl edit service hpa-hname-svckubectl create deployment hpa-hname-pods --image=sysnet4admin/echo-hname
kubectl expose deployment hpa-hname-pods --type=LoadBalancer --name=hpa-hname-svc --port=9999
kubectl edit service hpa-hname-svc

- 80으로 변경
  targetPort: 80
  vim ingress01-ingress.yaml
- 추가
- path: /
  pathType: Prefix
  backend:
  service:
  name: hpa-hname-svc
  port:

number: 9999
kubectl apply -f ingress01-ingress.yaml
kubectl get services
kubectl edit service hpa-hname-svc

- 80으로 변경
  targetPort: 80
  kubectl edit deployment hpa-hname-pods
  resources:
  requests:
  cpu: "10m"
  limits:
  cpu: "50m"
  kubectl autoscale deployment hpa-hname-pods --min=1 --max=30 --cpu-percent=50
  watch -n 1 "kubectl top pods"
  수정할거면... kubectl delete hpa hpa-hname-pods 지우고
  다시 autoscale하기
- cpu 10m으로 설정 cpu-percent=50는 50%가 넘으면 증설 (5m)
  34m -> 7 pods 가 생성

**# 프로메테우스

- 프로메테우스와 그라파나
  - 그라파나는 시각화 도구

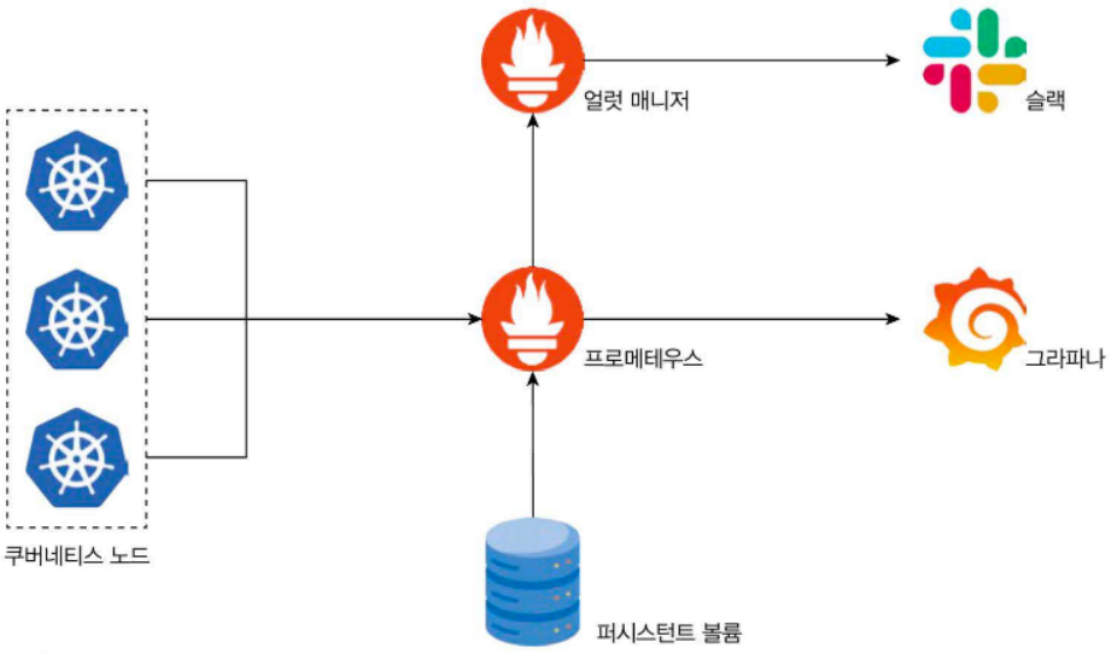


그림 12-1 프로메테우스 구조

컨트롤 g 라인명

mkdir prometheus && cd prometheus
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo update
helm search repo prometheus | grep prometheus-community/kube-prometheus-stack
helm pull prometheus-community/kube-prometheus-stack
tar xvfz kube-prometheus-stack-61.3.2.tgz
cd kube-prometheus-stack
cp ./values.yaml ./my-values.yaml
vim my-values.yaml
2614 type: NodePort
3585 serviceMonitorSelectorNilUsesHelmValues: false
3676 retentionSize: "1GiB"
cd chart
cd grafana
vim values.yaml
226 type: NodePort**

cd ../../
kubectl create namespace mymonitoring
helm install --namespace mymonitoring --generate-name prometheus-community/kube-prometheus-stack -f ./my-values.yaml
kubectl get all --namespace mymonitoring
kubectl get svc --namespace mymonitoring
kubectl port-forward --address 0.0.0.0 svc/kube-prometheus-stack-1721710504-prometheus-node-exporter 8080:9100 --namespace mymonitoring

http://192.168.56.200:8080/
kubectl get svc --namespace mymonitoring
vim ingress01-ingress.yaml
- path: /
pathType: Prefix
backend:
service:
name: kube-prometheus-stack-1721-prometheus
port:
number: 9090
kubectl apply -f ingress01-ingress.yaml
kubectl edit svc kube-prometheus-stack-1721-prometheus --namespace mymonitoring
**type: LoadBalancer
status:

loadBalancer:
ingress:
- ip: 192.168.56.102

grafana

vim ingress01-ingress.yaml
- path: /
pathType: Prefix
backend:
service:
name: kube-prometheus-stack-1721700847-grafana
port:
number: 80
kubectl edit svc kube-prometheus-stack-1721700847-grafana -n mymonitoring
type: LoadBalancer
status:
loadBalancer:
ingress:
- ip: 192.168.56.103
**

kubectl get secrets kube-prometheus-stack-1721700847-grafana -n mymonitoring -o jsonpath="{.data.admin-password}" | base64 -d
id / pw

**

해결

새로운 ova