

환경설정 관리- ConfigMap/Secret

쿠버네티스 배포단계(YAML 레벨)에서 환경설정 정보를 마이크로서비스가 어떻게 주입받아 활용하는지 Lab 을 통해 확인한다. -이미지 불변성의 원칙(Image Immutability)

Instruction

ConfigMap, Secret

configmap 조회

```
kubectl get configmap
```

- configmap 객체 조회

ConfigMap 폴더로 이동

- /container-orchestration/yaml# 경로에서 아래의 명령어를 입력

```
cd configmap
```

- /container-orchestration/yaml/configmap#

ConfigMap 생성

```
kubectl create configmap hello-cm --from-literal=language=java
kubectl get cm
kubectl get cm hello-cm -o yaml
```

- 위의 명령어 입력시
- 아래 내용 순으로 출력 되는지 확인해 주세요

```
configmap/hello-cm created
```

```
-----
NAME      DATA   AGE
hello-cm  1       34s
-----
```

```
apiVersion: v1
data:
  language: java
kind: ConfigMap
metadata:
  creationTimestamp: "2021-05-06T06:37:47Z"
  name: hello-cm
  namespace: default
  resourceVersion: "552648"
  selfLink: /api/v1/namespaces/default/configmaps/hello-cm
  uid: 893443ab-e2a1-4b0b-aeef-3c93134f2114
```

마이크로서비스 빌드/배포 (AWS)

- 도커이미지 빌드 & Push

```
docker build -t (ACCOUNT-UUID.dkr.ecr.REGION_CODE.amazonaws.com)/cm-sandbox:v1 .
aws ecr create-repository --repository-name cm-sandbox --region REGION_CODE
docker push (ACCOUNT-UUID.dkr.ecr.REGION_CODE.amazonaws.com)/cm-sandbox:v1
```

- 인증 오류 발생시, 로그인 수행

```
docker login --username AWS -p $(aws ecr get-login-password --region (Region-Code)) (Account-Id).dkr.ecr.(Region-Code).amazonaws.com/
```

- Push 확인

AWS : 매니지먼트 콘솔 > 서비스 > 컨테이너 > Elastic Container Registry

마이크로서비스 빌드/배포 (Azure)

- 도커이미지 빌드 & Push

```
docker build -t (uengineorg).azurecr.io/cm-sandbox:v1 .
docker push (uengineorg).azurecr.io/cm-sandbox:v1
```

- 인증 오류 발생시, 로그인 수행

```
az acr login --name (Azure Container Registry 명)
```

- Push 확인

Azure : 매니지먼트 콘솔 > 컨테이너 > Azure Container Registry

컨테이너 및 서비스 생성 (AWS, Azure 공통)

cm-deployment.yaml 파일에서 **Image** 정보를 컨테이너 레지스트리 **Push** 한 정보로 수정

```
kubectl create -f cm-deployment.yaml
kubectl create -f cm-service.yaml
```

Secret

- `pwd` 로 현 위치가 `/container-orchestration/yaml/secret/` 인지 확인
- `kubectl create secret generic my-password --from-literal=password=mysqlpassword`
- Pod 에서 Secret 파일 마운트 사용하기 내용을 참고하여 배포 및 서비스 확인해 보기

Secret 생성

- base64 형태로 인코딩하여 YAML 파일내에 직접 생성 가능

```
echo java | base64
```

```
echo `amF2YQo=` | base64 --decode
```

- 위의 명령어 입력시
- 아래 내용 순으로 출력 되는지 확인해 주세요
`amF2YQo=`

CheckPoints

1. 모든 요구사항을 만족하는가 ☐