

# Kubeflow 설치 실습 자료

## 1. Prerequisite 개념

## 2. 설치 방법

## 3. Kubeflow 설치

### Prerequisite

### Step 1) kustomize 설정

### Step 2) minikube start

### Step 3) Git clone kubeflow/manifests

### Step 4) 각각의 kubeflow 구성 요소 순서대로 설치

## 4. Kubeflow 접속

## 5. Other Useful Tool

---

## 1. Prerequisite 개념

- kustomize
    - Helm 과 비슷한 역할을 담당
      - 여러 개의 yaml 파일들을 쉽게 관리하기 위한 도구
    - 여러 resource 들의 configuration 을 템플릿(**base**)과 커스터마이제이션한 부분 (**overlay**)을 나누어서 관리할 수 있는 도구
    - `kustomize build` 명령을 통해, base + overlay 가 merge 된 형태의 yaml 파일들을 generate 할 수 있음
- 

## 2. 설치 방법

- Kfctl
  - v1.2 이후로는 공식적으로 지원하지 않음
- Minikf
  - 아직 v1.3 까지만 릴리즈
  - kubeflow 가 이미 설치되어있는 VM 이미지를 사용하여 Vagrant 쉽게 설치 가능
- Kubeflow manifests
  - 공식 릴리즈 관리용 Repository.

- Kustomize v3 기반으로 manifests 파일 관리
  - 가장 정석적인 방법
- 

## 3. Kubeflow 설치

### Prerequisite

- Kubernetes 환경
    - 버전 : v1.17 ~ v1.21
      - v1.19.3 사용
    - Default StorageClass
      - Dynamic provisioning 지원하는 storageclass
    - TokenRequest API 활성화
      - alpha version 의 API 이므로, k8s API Server 에 해당 feature gate 를 설정 해주어야 함
  - Kustomize
    - 버전 : v3.x
      - v3.2.0 사용
- 

### Step 1) kustomize 설정

```
# 바이너리 다운 (for linux amd64)
# 이외의 os 는 https://github.com/kubernetes-sigs/kustomize/releases/tag/v3.2.0 경로에서 binary 링크 확인
wget https://github.com/kubernetes-sigs/kustomize/releases/download/v3.2.0/kustomize_3.2.0_linux_amd64

# file mode 변경
chmod +x kustomize_3.2.0_linux_amd64


# file 위치 변경
sudo mv kustomize_3.2.0_linux_amd64 /usr/local/bin/kustomize

# 버전 확인
kustomize version
```


### Step 2) minikube start

```
# minikube start
# docker driver option
# cpu 4 개 할당
# memory 7g 할당
# kubernetes version v.19.3 설정
# --extra-config 부분은 tokenRequest 활성화 관련 설정
minikube start --driver=docker \
  --cpus='4' --memory='7g' \
  --kubernetes-version=v1.19.3 \
  --extra-config=apiserver.service-account-signing-key-file=/var/lib/minikube/certs/sa.key \
  --extra-config=apiserver.service-account-issuer=kubernetes.default.svc
```

- 다음과 같이 default-storageclass 는 기본으로 addon 이 활성화되어있음

 Kubernetes 구성 요소를 확인...

- Using image gcr.io/k8s-minikube/storage-provisioner:v5

 애드온 활성화 : storage-provisioner, default-storageclass

### Step 3) Git clone kubeflow/manifests

- kubeflow/manifests Repository 를 로컬 폴더에 git clone 합니다.
  - <https://github.com/kubeflow/manifests>

```
cd ~/fast-campus-demo/kubeflow-tutorial

# git clone
git clone git@github.com:kubeflow/manifests.git

# 해당 폴더로 이동
cd manifests

# v1.4.0 태그 시점으로 git checkout
git checkout tags/v1.4.0
```


### Step 4) 각각의 kubeflow 구성 요소 순서대로 설치

이번 실습에서 사용하지 않는 일부 구성요소는 설치를 진행하지 않습니다.

- Knative, KFServing, Training Operator, MPI Operator

#### GitHub - kubeflow/manifests at v1.4.0

This repo is owned by the Manifests Working Group. If you are a contributor authoring or editing the packages please see Best Practices.

 <https://github.com/kubeflow/manifests/tree/v1.4.0>

#### kubeflow/ manifests

A repository for Kustomize manifests



 94 Contributors  95 Issues  295 Stars  465 Forks



- kustomize build 의 동작 확인해보기
  - `kustomize build common/cert-manager/cert-manager/base`
  - `|` pipe 연산자를 활용하여, kustomize build 의 결과물을 `kubectl apply -f -` 하여 적용
- 모든 구성요소가 Running 이 될 때까지 대기
  - `kubectl get po -n kubeflow -w`
    - 상당히 많은 구성요소들의 docker image 를 로컬 머신에 pull 받기 때문에, 최초 실행 시에는 네트워크 상황에 따라 약 30 분 정도까지도 소요될 수 있음
  - 여러 구성요소들의 상태가 `PodInitializing` → `ContainerCreating` 으로 진행되지만 시간이 오래걸리는 경우라면 정상적인 상황이지만, 상태가 `Error` or `CrashLoopBackOff` 라면 minikube start 시의 설정을 다시 확인해주시기 바랍니다.

## 4. Kubeflow 접속

- 포트 포워딩
  - `kubectl port-forward svc/istio-ingressgateway -n istio-system 8080:80`
  - gateway 를 포트포워딩하여 localhost:8080 으로 kubeflow 대시보드에 접속
- 접속 정보
  - kubeflow manifests 배포 시, user 접속 정보 관련 설정을 변경하지 않은 경우의 default 접속 정보
    - ID : user@example.com
    - PW : 12341234

## 5. Other Useful Tool

- kubectlx & kubens

- <https://github.com/ahmetb/kubectx>
  - Install 방법도 매우 간편
- kubernetes 의 current-context 와, kubernetes 의 current-namespace 를 변경할 수 있는 툴
  - 여러 개의 context 나 namespace 를 다루는 경우에 유용하게 사용할 수 있음
- ex)
  - `kubens kubeflow` 를 수행하면 현재 바라보고 있는 namespace 가 kubeflow로 변경됨
- kubectl-alias
  - <https://github.com/ahmetb/kubectl-aliases>
  - kubectl 관련 여러 명령어에 대한 alias 를 자동 생성
    - 자주 사용하는 명령을 쉽게 수행할 수 있음
  - ex)
    - `kubectl get pod` → `kgpo`
    - `kubectl get deployment -w` → `kgdepw`