7th Week

일곱 번째 뵙겠습니다 ?!

▷ 잠시만 기다렸다가 30분 되면 시작하겠습니다~^^

▷ Kubernetes 공부 어떠세요!?

- 조금만 더 같이 파고 들어가 보자구요 !!!

▷ Camera는 가급적 켜 주시면 대단히 감사하겠습니다 !!!

- 너무 부끄러우면 Snap Camera를 사용하시는 것 까지는~ ^^

▷ 오늘 수업 자료는 아래 링크에서 다운로드 받으실 수 있어요.

- https://github.com/whatwant-school/kubernetes

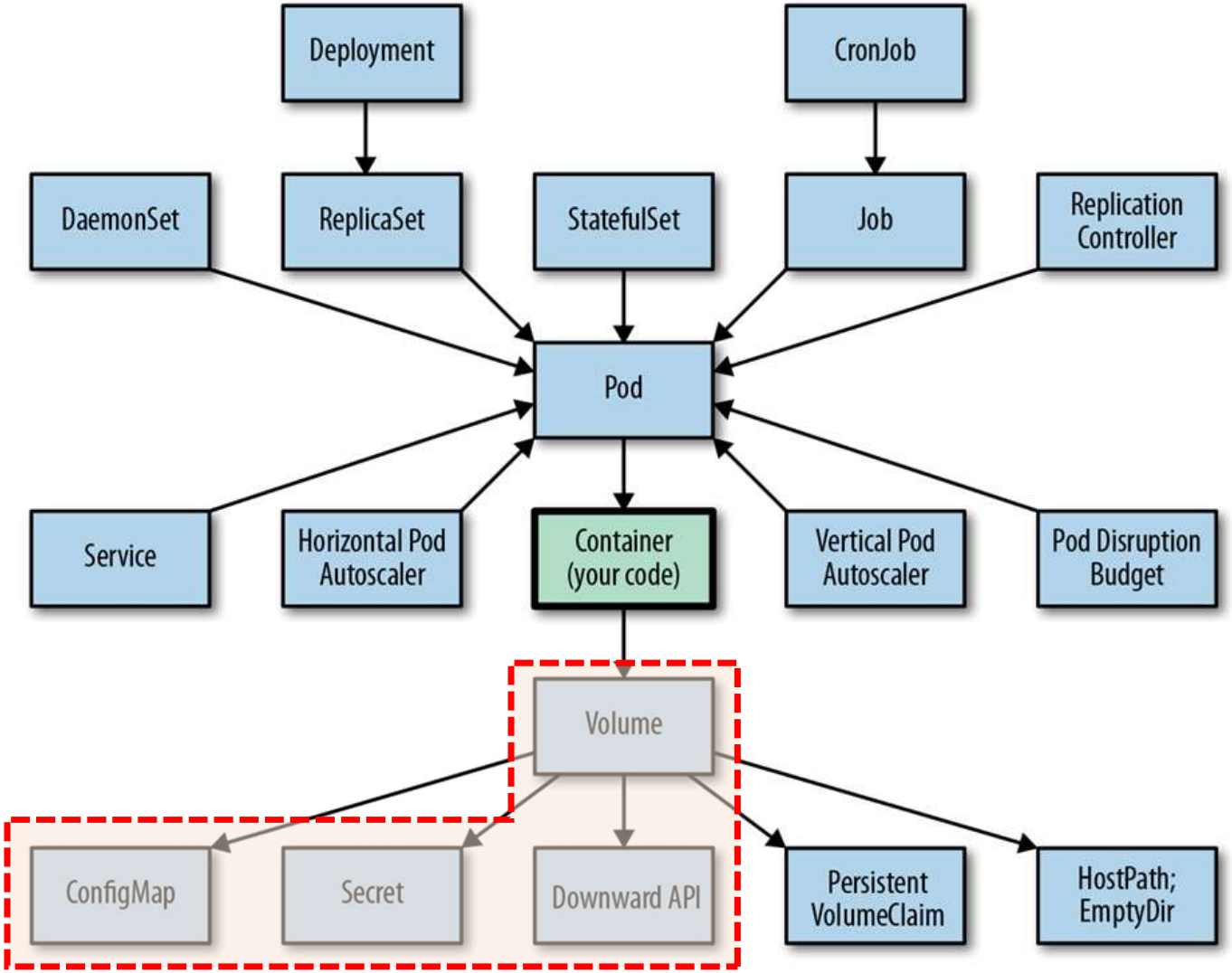
///

지난 수업 기억 나시나요?

https://kahoot.it/

///

Volume



※ 참고 : https://www.oreilly.com/library/view/kubernetes-patterns/9781492050278/ch01.html

Flip Learning

(Volume - ConfigMap/Secret/downwardAPI)

/// 님

///

Arguments(Parameters) in Container

Dockerfile

- 주어진 시간만큼 sleep 하는 실습용 script를 만들어 보자.

fortuneloop.sh

```
#!/bin/bash
trap "exit" SIGINT
INTERVAL=$1
echo generate every $INTERVAL seconds
mkdir -p /var/htdocs

while :
do
  echo $(date) Writing to /var/htdocs/index.html
  /usr/games/fortune > /var/htdocs/index.html
  sleep $INTERVAL
done
```

Dockerfile

```
FROM ubuntu:latest

RUN apt-get update
RUN apt-get -y install fortune

ADD fortuneloop.sh /bin/fortuneloop.sh
RUN chmod +x /bin/fortuneloop.sh

ENTRYPOINT ["/bin/fortuneloop.sh"]
CMD ["10"]
```

- fortune: 오늘의 운세를 알려주는 명령어 패키지

※ 참고 : https://github.com/luksa/kubernetes-in-action/tree/master/Chapter07

docker build & execute

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ docker build -t fortune:v1.0 .

...

remote ❯ docker run -it --name fortune fortune:v1.0

generate every 10 seconds
Tue Aug 30 13:21:14 UTC 2022 Writing to /var/htdocs/index.html
Tue Aug 30 13:21:24 UTC 2022 Writing to /var/htdocs/index.html
Tue Aug 30 13:21:34 UTC 2022 Writing to /var/htdocs/index.html
^C%

( DockerHub에서 fortune repository 생성 )

remote ❯ docker login
remote ❯ docker tag fortune:v1.0 whatwant/fortune:v1.0
remote ❯ docker push whatwant/fortune:v1.0
```

명령어	RUN	ENTRYPOINT	CMD
실행 시점	이미지 빌드	컨테이너 실행	컨테이너 실행
설명	주로 이미지 안에 특정 패키지 설치 용도로 사용	항상 (무조건) 실행	ENTRYPOINT 파라미터 전달 또는 선택적 실행 용도

Environment Variable in Pod

env

- container 내부 환경 변수를 Pod 정의할 때 선언할 수 있다

pod-env.yaml

```yaml
apiVersion: v1
kind: Pod

metadata:
  name: pod-env

spec:
  containers:
  - name: print-env
    image: bash

    env:
    - name: FIRST
      value: "One"

    - name: SECOND
      value: "Two"

    - name: THIRD
      value: "Three"

    command: ["echo"]
    args: ["$(FIRST), $(SECOND), $(THIRD)"]
```

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ kubectl create -f pod-env.yaml

pod/pod-env created


remote ❯ kubectl get pods -o wide

NAME     READY  STATUS     RESTARTS  AGE  IP              NODE      NOMINATED NODE   READINESS GATES
pod-env  0/1    Completed  0         9s   10.233.103.104  worker2   <none>           <none>


remote ❯ kubectl logs pod-env

One, Two, Three
```

명령어	command	args
설명	실행	파라미터 전달

※ 참고 : https://kubernetes.io/ko/docs/tasks/inject-data-application/define-environment-variable-container/

Arguments(Parameters) in Pod

YAML

- 앞에서 만든 10초 간격의 container를 이용해 5초 간격으로 실행하는 Pod를 구성해보자.

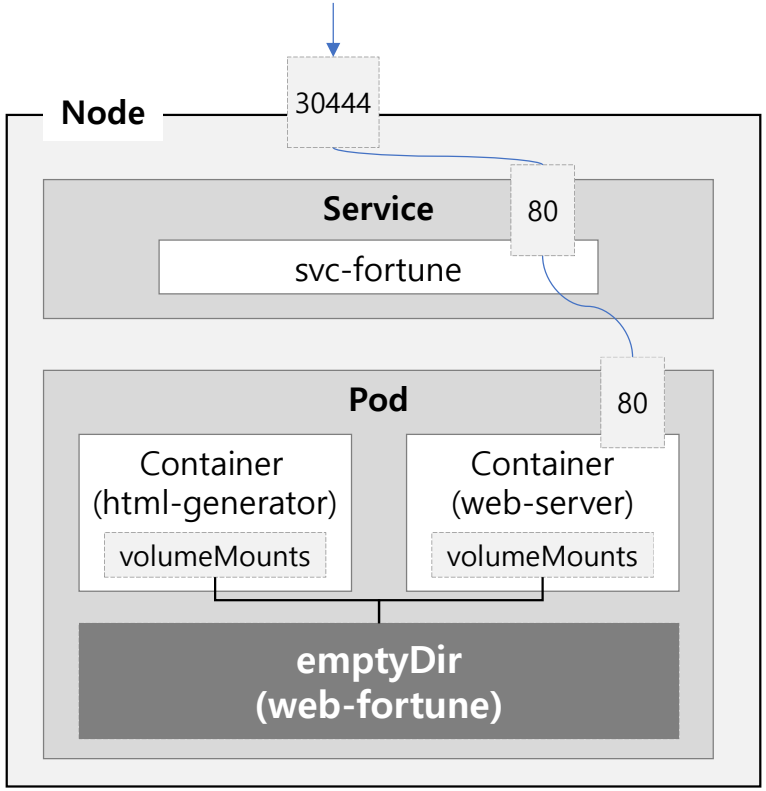
pod-fortune.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: pod-fortune
  labels:
    app: fortune
spec:
  containers:
  - name: html-generator
    image: whatwant/fortune:v1.0
    args: ["5"]
    volumeMounts:
    - name: web-fortune
      mountPath: /var/htdocs

  - name: web-server
    image: nginx:alpine
    volumeMounts:
    - name: web-fortune
      mountPath: /usr/share/nginx/html
      readOnly: true
    ports:
    - containerPort: 80

  volumes:
  - name: web-fortune
    emptyDir: {}
```

svc-fortune.yaml

```yaml
apiVersion: v1
kind: Service

metadata:
  name: svc-fortune

spec:
  type: NodePort

  ports:
  - port: 80
    targetPort: 80
    nodePort: 30444

  selector:
    app: fortune
```

설명	도커 필드 이름	쿠버네티스 필드 이름
컨테이너에서 실행되는 커맨드	Entrypoint	command
커맨드에 전달되는 인자들	Cmd	arg

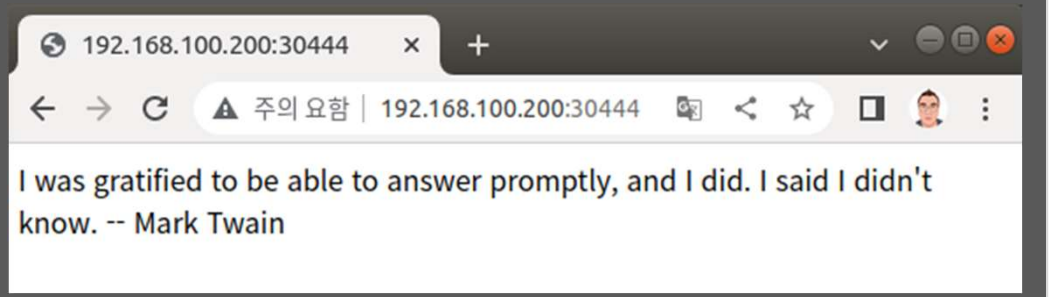


※ 참고 : https://kubernetes.io/ko/docs/tasks/inject-data-application/define-command-argument-container/

Execute

- Container의 CMD로 지정된 "10"이라는 값이, K8s의 args로 정의된 "5"라는 값으로 덮어써진 것을 확인할 수 있다.

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ kubectl create -f pod-fortune.yaml

pod/pod-fortune created


remote ❯ kubectl logs pod-fortune -c html-generator

generate every 5 seconds
Fri Sep 2 15:05:25 UTC 2022 Writing to /var/htdocs/index.html
Fri Sep 2 15:05:30 UTC 2022 Writing to /var/htdocs/index.html
Fri Sep 2 15:05:35 UTC 2022 Writing to /var/htdocs/index.html


remote ❯ kubectl create -f svc-fortune.yaml

service/svc-fortune created


remote ❯ kubectl get services -o wide

NAME          TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE   SELECTOR
kubernetes    ClusterIP   10.233.0.1      <none>        443/TCP        42d   <none>
svc-fortune   NodePort    10.233.62.145   <none>        80:30444/TCP   12s   app=fortune
```



///

Break

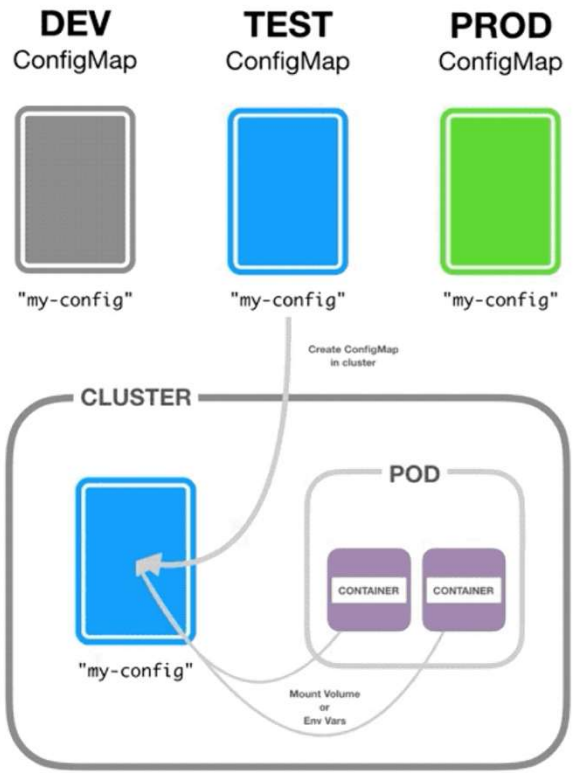
///

Volume - ConfigMap

Why ConfigMap ?

- 독립적인 리소스로 동일한 Pod라도 여러 개의 ConfigMap을 선택하여 사용 가능

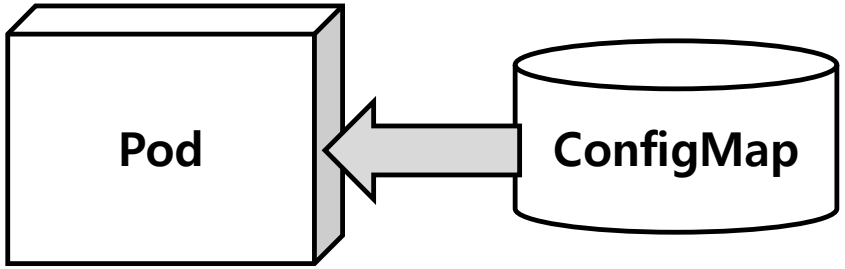
  . ConfigMap만 변경하여 Pod를 손쉽게 테스트 할 수 있다.



※ 참고 : https://timewizhan.tistory.com/entry/Kubernetes-ConfigMap

How ?

- ConfigMap을 생성해 놓으면, Pod에서 가져다 사용하는 방식



env

envFrom

volumes

ConfigMap을 **설정**하는 방법

ConfigMap을 **생성**하는 방법

from directory : --from-file=/dir

from file : --from-file=/dir/file

from literal : --from-literal=key=value

YAML

cm-envFrom.yaml

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: cm-envfrom

data:
  map-hash-bucket-size: "128"
  ssl-protocols: SSLv2
```

cm-volume.yaml

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: cm-volume

data:
  index.html: |-
     <html>
     <h1>Hello from ConfigMap</h1>
     </html>
```

pod-ConfigMap.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: pod-configmap
  labels:
    app: nginx

spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
    ports:
    - containerPort: 80

    envFrom:
      - configMapRef:
          name: cm-envfrom

    volumeMounts:
    - name: nginx-volume
      subPath: index.html
      mountPath: /usr/share/nginx/html/index.html

  volumes:
  - name: nginx-volume
    configMap:
      name: cm-volume
```

svc-ConfigMap.yaml

```yaml
apiVersion: v1
kind: Service
metadata:
  name: svc-configmap

spec:
  type: NodePort
  ports:
  - port: 80
    targetPort: 80
    nodePort: 30501

  selector:
    app: nginx
```

※ 참고 : https://github.com/tncad/k8s-app-cpd/blob/a41111ae84c45fecb5c21ba96fec0ee113beff81/021-kustomize/full/deployment.yaml

Execute - YAML

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ kubectl create -f cm-envFrom.yaml
remote ❯ kubectl create -f cm-volume.yaml
remote ❯ kubectl create -f pod-ConfigMap.yaml
remote ❯ kubectl create -f svc-ConfigMap.yaml

remote ❯ kubectl get pods -o wide

NAME            READY   STATUS    RESTARTS   AGE   IP              NODE      NOMINATED NODE   READINESS GATES
pod-configmap   1/1     Running   0          31s   10.233.103.110   worker2   <none>           <none>


remote ❯ kubectl exec -it pod-configmap -- printenv

...
NJS_VERSION=1.14.2.0.2.6-1~stretch
ssl-protocols=SSLv2
map-hash-bucket-size=128
KUBERNETES_SERVICE_PORT=443
...


remote ❯ kubectl get services -o wide

NAME            TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)        AGE    SELECTOR
kubernetes      ClusterIP   10.233.0.1     <none>        443/TCP        43d    <none>
svc-configmap   NodePort    10.233.15.73   <none>        80:30501/TCP   117s   app=nginx
```

Execute - CLI

- 앞에서 실습한 것들을 모두 지우고, CLI로 ConfigMap을 생성하는 실습을 해보자.

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ cat ./cm-env/map-hash-bucket-size

128


remote ❯ cat ./cm-env/ssl-protocols

SSLv2


remote ❯ cat index.html

<html>
<h1>Hello from ConfigMap</h1>
</html>


remote ❯ kubectl create configmap cm-envfrom --from-file=./cm-env/

configmap/cm-envfrom created


remote ❯ kubectl create configmap cm-volume --from-file=index.html

configmap/cm-volume created
```

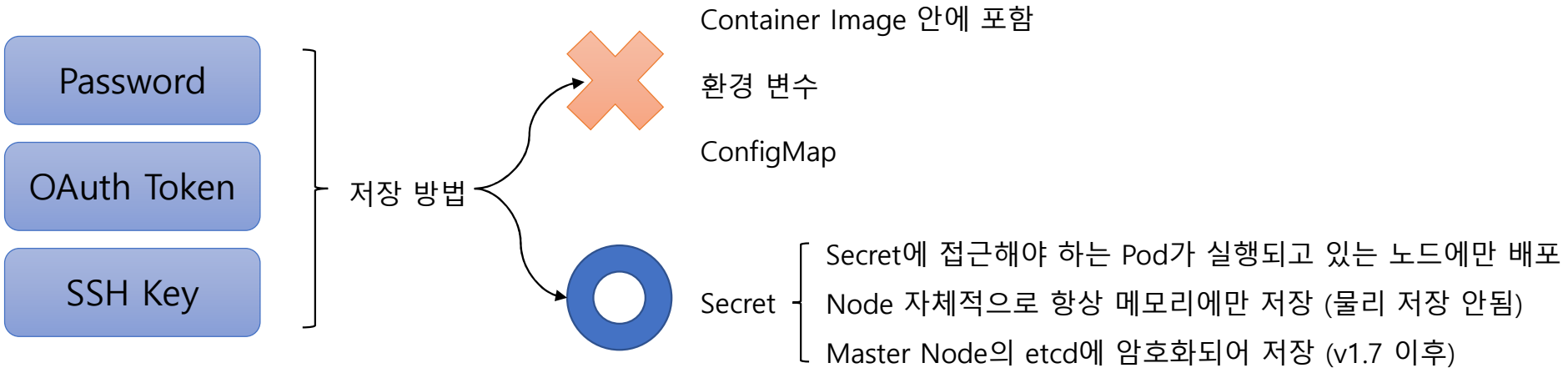
```
remote ❯ kubectl get configmaps

NAME               DATA   AGE
cm-envfrom         2      89s
cm-volume          1      71s
kube-root-ca.crt   1      43d


remote ❯ kubectl create -f pod-ConfigMap.yaml
remote ❯ kubectl create -f svc-ConfigMap.yaml


remote ❯ kubectl exec -it pod-configmap -- printenv

...
NJS_VERSION=1.14.2.0.2.6-1~stretch
ssl-protocols=SSLv2
map-hash-bucket-size=128
KUBERNETES_SERVICE_PORT=443
...
```

///

Volume - Secret

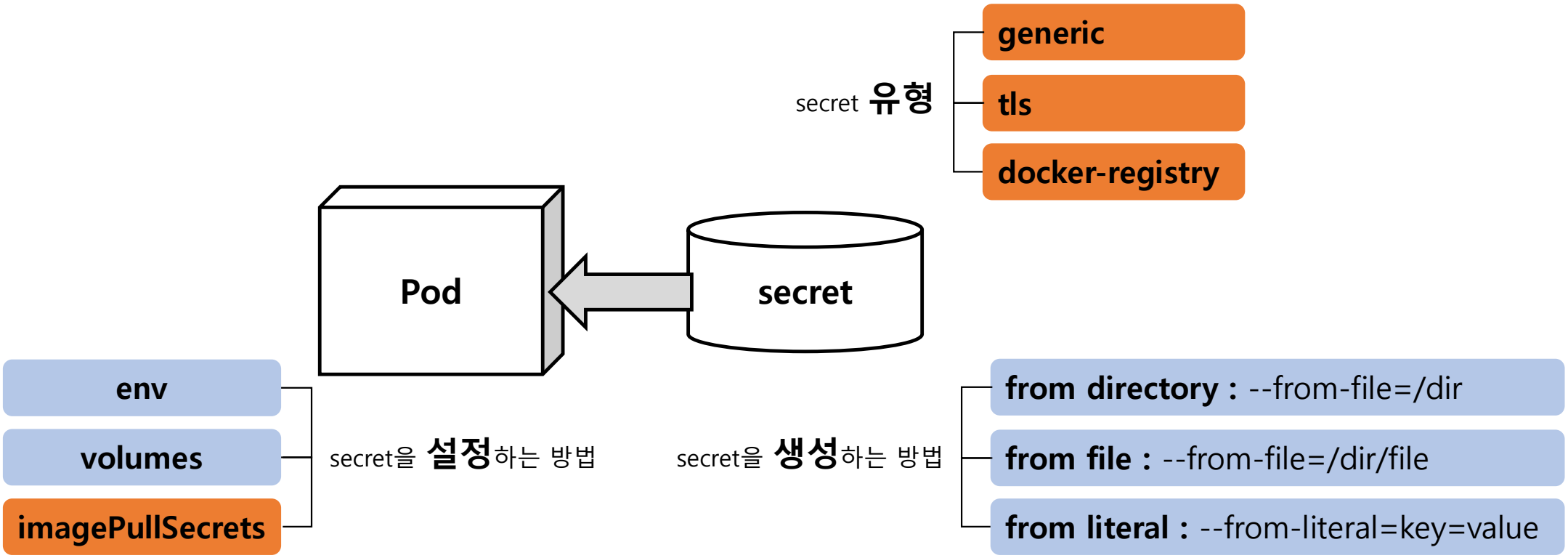
Why secret ?

- 노출되면 안되는 정보를 저장하기 위한 방법 필요



How secret ?

- configMap 사용법과 유사



default token

- 기본적으로 생성되어 있는 secret을 확인해보자

```
remote ❯ kubectl get secrets -o wide

NAME                  TYPE                                  DATA   AGE
default-token-lq4gd   kubernetes.io/service-account-token   3      43d

remote ❯ kubectl describe secrets default-token-lq4gd

Name:         default-token-lq4gd
Namespace:    default
Labels:       <none>
Annotations:  kubernetes.io/service-account.name: default
              kubernetes.io/service-account.uid: ff77cdc5-4260-406f-bbd9-cd24031259f2

Type:  kubernetes.io/service-account-token

Data
====
ca.crt:     1099 bytes
namespace:  7 bytes
token:
eyJhbGciOiJSUzI1NiIsImtpZCI6ImVicWlURzJTdWJuWnVXUHV5MzhWN0dGc000bUVvVUt3dHpSZG1LSDk3S2cifQ.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9uYW1lc3BhY2UiOiJkZWZhdWx0Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9zZWNyZXQubmFtZSI6ImRlZmF1bHQtdG9rZW4tbHE0Z2QiLCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC5uYW1lIjoiZGVmYXVsdCIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VydmljZS1hY2NvdW50LnVpZCI6ImZmNzdjZGM1LTQyNjAtNDA2Zi1iYmQ5LWNkMjQwMzEyNTlmMiIsInN1YiI6InN5c3RlbTpzZXJ2aWNlYWNjb3VudDpkZWZhdWx0OmRlZmF1bHQifQ.drHZ8HdFzTCrWYoFWflqs2XAvxOLAYFMfZcXmuFihlHVlErudYLp0bLVj2F7JenzbCFPHD8mPxBX5Z-
kl2y9HCEQpFLfF7TqShLe4uGt5fEi8HWEOOoxgZToQ1jqGzXeq7uNNntuIQC3STq7ZNsg2Lk__xZZ6ugbPquccAvcQbOfXDx_vq3fv03rf88G5q6ILGjSW3hgzzfAO88in3bZUTIXHrMW6B2CVfb-
wStWnUsFEQy0QB3cpVOOPWefzyDEglBqsy2iNh8OeSKGq453MnoI_FCLUWdKI0fP4eOEA0WL7dwpvhid8XbpoAcWPY_i23XkszAdrsas3Pu39wiW0A
```

3개의 secret 데이터 존재

Kubernetes API 통신을 위한 3종 데이터 정보

default token in Pod

- Pod를 생성하면 기본적으로 API Server와 통신하기 위한 secret이 mount 된다.

```
remote ❯ kubectl run web --image=nginx:latest

pod/web created


remote ❯ kubectl describe pods web

Name:         web
Namespace:    default
Priority:     0
Node:         worker2/192.168.100.202
Start Time:   Tue, 15 Feb 2022 00:11:11 +0900
Labels:       run=web
Annotations:  cni.projectcalico.org/containerID: 6...
              cni.projectcalico.org/podIP: 10.233....
              cni.projectcalico.org/podIPs: 10.23....
Status:       Running
IP:           10.233.103.39
IPs:
  IP:  10.233.103.39
Containers:
  web:
    Container ID:   containerd://48f16fc7f85e54703...
    Image:          nginx:latest
    Image ID:       docker.io/library/nginx@sha256...
    Port:           <none>
    Host Port:      <none>
...
```

```
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-ggr4n (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-ggr4n:
    Type:                    Projected (a volume that contains injected data from multiple ...)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:
  Type    Reason     Age   From               Message
  ----    ------     ----  ----               -------
  Normal  Scheduled  17s   default-scheduler  Successfully assigned default/web to worker2
  Normal  Pulling    17s   kubelet            Pulling image "nginx:latest"
  Normal  Pulled     7s    kubelet            Successfully pulled image "nginx:latest" in 9.1...
  Normal  Created    7s    kubelet            Created container web
  Normal  Started    7s    kubelet            Started container web
```

///

Ready - Private Image

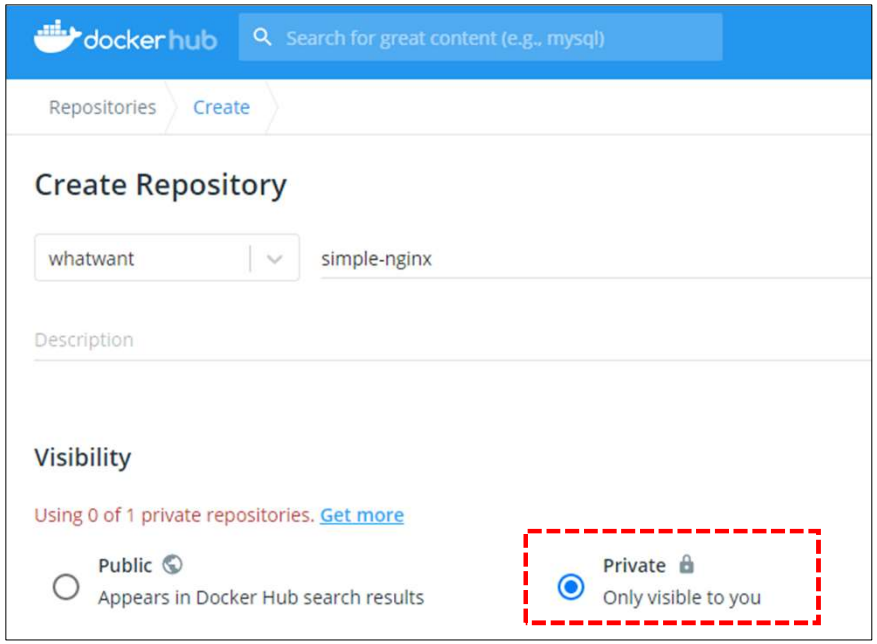
- Private Image를 하나 만들어서 Docker Hub에 업로드 해놓자.

index.html

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Docker Nginx</title>
</head>
<body>
  <h2>Hello from Nginx container</h2>
</body>
</html>
```

Dockerfile

```
FROM nginx:latest

COPY ./index.html /usr/share/nginx/html/index.html
```




```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on/private

remote ❯ docker login

remote ❯ docker build -t whatwant/simple-nginx:v0.1 -f Dockerfile .

remote ❯ docker push whatwant/simple-nginx:v0.1
```

Execute - Failure

pod-private-fail.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: pod-private
  labels:
    app: nginx

spec:
  containers:
  - name: nginx
    image: whatwant/simple-nginx:v0.1
```

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ kubectl create -f pod-private-fail.yaml

pod/pod-private created


remote ❯ kubectl get pods -o wide

NAME          READY   STATUS         RESTARTS   AGE   IP              NODE      NOMINATED NODE   READINESS GATES
pod-private   0/1     ErrImagePull   0          10s   10.233.103.112  worker2   <none>           <none>
```

```
remote ❯ kubectl describe pods pod-private

...
Events:
  Type     Reason     Age                From               Message
  ----     ------     ----               ----               -------
  Normal   Scheduled  46s                default-scheduler  Successfully assigned default/pod-private to worker2
  Normal   Pulling    29s (x2 over 47s)  kubelet            Pulling image "whatwant/simple-nginx:v0.1"
  Warning  Failed     26s (x2 over 43s)  kubelet            Failed to pull image "whatwant/simple-nginx:v0.1": rpc error: code = Unknown desc = failed to
pull and unpack image "docker.io/whatwant/simple-nginx:v0.1": failed to resolve reference "docker.io/whatwant/simple-nginx:v0.1": pull access denied,
repository does not exist or may require authorization: server message: insufficient_scope: authorization failed
  Warning  Failed     26s (x2 over 43s)  kubelet            Error: ErrImagePull
  Normal   BackOff    12s (x2 over 43s)  kubelet            Back-off pulling image "whatwant/simple-nginx:v0.1"
  Warning  Failed     12s (x2 over 43s)  kubelet            Error: ImagePullBackOff
```

docker-registry

pod-private-success.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: pod-private
  labels:
    app: nginx

spec:
  containers:
  - name: nginx
    image: whatwant/simple-nginx:v0.1

  imagePullSecrets:
    - name: my-docker-hub
```

```
kubectl create secret docker-registry <secret-name> \
                    [ --docker-server=<your-registry-server> \ ]
                      --docker-username=<your-name> \
                      --docker-password=<your-password> \
                      --docker-email=<your-email>
```

```
remote ❯ kubectl create secret docker-registry my-docker-hub \
                            --docker-username=whatwant \
                            --docker-password='xxx' \
                            --docker-email='whatwant@gmail.com'
secret/my-docker-hub created


remote ❯ kubectl get secrets -o wide

NAME                  TYPE                                  DATA   AGE
default-token-lq4gd   kubernetes.io/service-account-token   3      43d
my-docker-hub         kubernetes.io/dockerconfigjson        1      22s
```

```
remote ❯ kubectl create -f pod-private-success.yaml

pod/pod-private created


remote ❯ kubectl get pods

NAME          READY   STATUS    RESTARTS   AGE
pod-private   1/1     Running   0          49s
```

Type of Secret

빌트인 타입	사용처
Opaque	임의의 사용자 정의 데이터
kubernetes.io/service-account-token	서비스 어카운트 토큰
kubernetes.io/dockercfg	직렬화 된(serialized) ~/.dockercfg 파일
kubernetes.io/dockerconfigjson	직렬화 된 ~/.docker/config.json 파일
kubernetes.io/basic-auth	기본 인증을 위한 자격 증명(credential)
kubernetes.io/ssh-auth	SSH를 위한 자격 증명
kubernetes.io/tls	TLS 클라이언트나 서버를 위한 데이터
bootstrap.kubernetes.io/token	부트스트랩 토큰 데이터

※ 참고 : https://kubernetes.io/ko/docs/concepts/configuration/secret/

///

git-sync

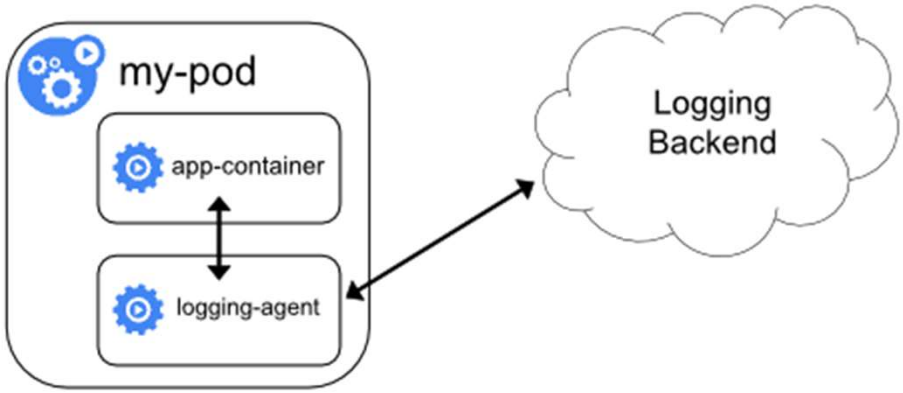
volume gitRepo: Deprecated → ‘git-sync’ Sidecar Container

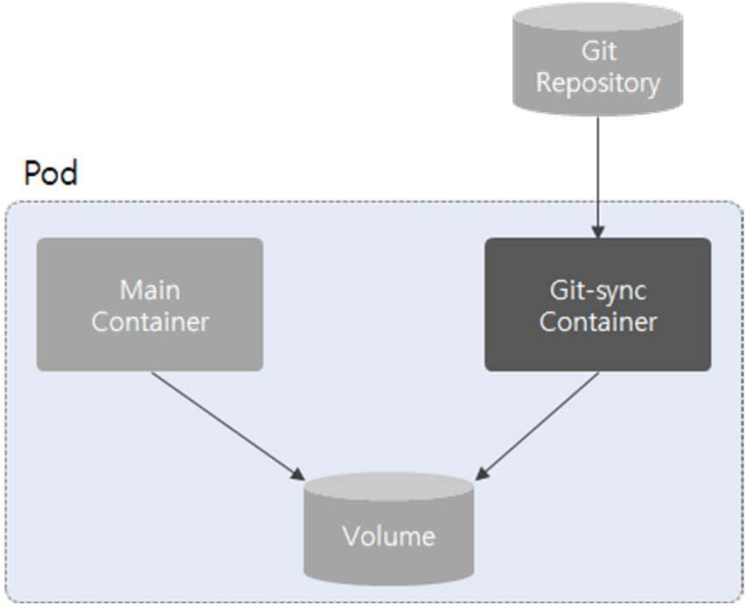
- 사이드카 패턴 (Sidecar Pattern)

  . 기본 컨테이너의 기능을 확장하거나 강화하는 용도의 컨테이너를 추가하는 패턴

  . 기본 컨테이너에는 원래 목적의 기능에만 충실하고 나머지 부가적인 공통 기능들은 사이드카 컨테이너를 추가해서 사용

- deprecate 된 gitRepo volume 대신 git-sync의 sidecar container를 구성하는 것으로 구현해야 한다





※ 참고 : https://arisu1000.tistory.com/27863

※ 참고 : https://github.com/kubernetes/git-sync

Ready - SSH Key

- ssh-key 정보와 known_hosts 정보를 secret으로 등록해보자

  . ssh-key를 아직 생성하지 않았다면, `ssh-keygen`으로 생성 후 진행하면 된다.

```
remote ❯ ls -al ~/.ssh

합계 20
drwx------  2 chani chani 4096  7월 11  2020 .
drwxr-xr-x 22 chani chani 4096  9월  3 07:12 ..
-rw-------  1 chani chani 1679  7월 11  2020 id_rsa
-rw-r--r--  1 chani chani  396  7월 11  2020 id_rsa.pub
-rw-r--r--  1 chani chani 2434  8월 15 11:48 known_hosts


remote ❯ ssh-keyscan github.com > ./known_hosts

# github.com:22 SSH-2.0-babeld-a3d15bff
# github.com:22 SSH-2.0-babeld-a3d15bff
# github.com:22 SSH-2.0-babeld-a3d15bff
```

github.com 접근을 위해 미리 known_hosts에 등록해야 한다.
그렇지 않으면, 처음 접근하는 서버의 등록 과정으로 문제가 발생하게 된다.

```
remote ❯ kubectl create secret generic my-ssh --from-file=ssh=$HOME/.ssh/id_rsa --from-file=known_hosts=./known_hosts


remote ❯ kubectl get secrets -o wide

NAME                  TYPE                                  DATA   AGE
default-token-xf884   kubernetes.io/service-account-token   3      24d
my-ssh                Opaque                                2      12s
```

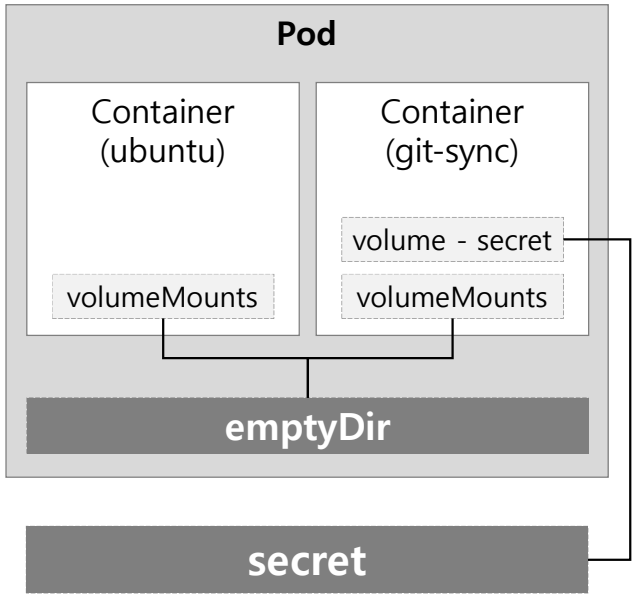
YAML

pod-git-sync.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: git-sync

spec:
  containers:
  - name: ubuntu
    image: ubuntu:latest
    args:
    - sleep
    - "370000"

    volumeMounts:
    - name: myrepo
      mountPath: "/repo"

  - name: git-sync
    image: k8s.gcr.io/git-sync/git-sync:v3.6.1
    args:
    - -ssh
    - -repo=git@github.com:whatwant-school/private-repo.git
    - -root=/repo
    - -dest=private-repo
    - -branch=main
    - -depth=1
    volumeMounts:
    - name: myrepo
      mountPath: "/repo"

    - name: git-secret
      mountPath: /etc/git-secret
      readOnly: true

    securityContext:
      runAsUser: 65533

  securityContext:
    fsGroup: 65533

  restartPolicy: Never

  volumes:
  - name: myrepo
    emptyDir: {}

  - name: git-secret
    secret:
      secretName: my-ssh
      defaultMode: 0400
```

※ 참고 : https://github.com/kubernetes/git-sync/blob/release-3.x/docs/ssh.md

Execute




```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on

remote ❯ kubectl create -f pod-git-sync.yaml


remote ❯ kubectl get pods -o wide

NAME       READY   STATUS    RESTARTS   AGE   IP              NODE      NOMINATED NODE   READINESS GATES
git-sync   2/2     Running   0          18s   10.233.103.114   worker2   <none>           <none>


remote ❯ kubectl exec -it git-sync -c ubuntu -- /bin/bash

groups: cannot find name for group ID 65533
root@git-sync:/# ls -al /repo
total 16
drwxrwsrwx 4 root  65533 4096 Sep  2 22:33 .
drwxr-xr-x 1 root  root  4096 Sep  2 22:33 ..
drwxr-sr-x 9 65533 65533 4096 Sep  2 22:33 .git
drwxr-sr-x 2 65533 65533 4096 Sep  2 22:33 56af7aa4197fee5da30ccdd562cc2ea5e751af76
lrwxrwxrwx 1 65533 65533   40 Sep  2 22:33 private-repo -> 56af7aa4197fee5da30ccdd562cc2ea5e751af76


root@git-sync:/# ls -al /repo/private-repo/
total 16
drwxr-sr-x 2 65533 65533 4096 Sep  2 22:33 .
drwxrwsrwx 4 root  65533 4096 Sep  2 22:33 ..
-rw-r--r-- 1 65533 65533   67 Sep  2 22:33 .git
-rw-r--r-- 1 65533 65533   60 Sep  2 22:33 README.md
```

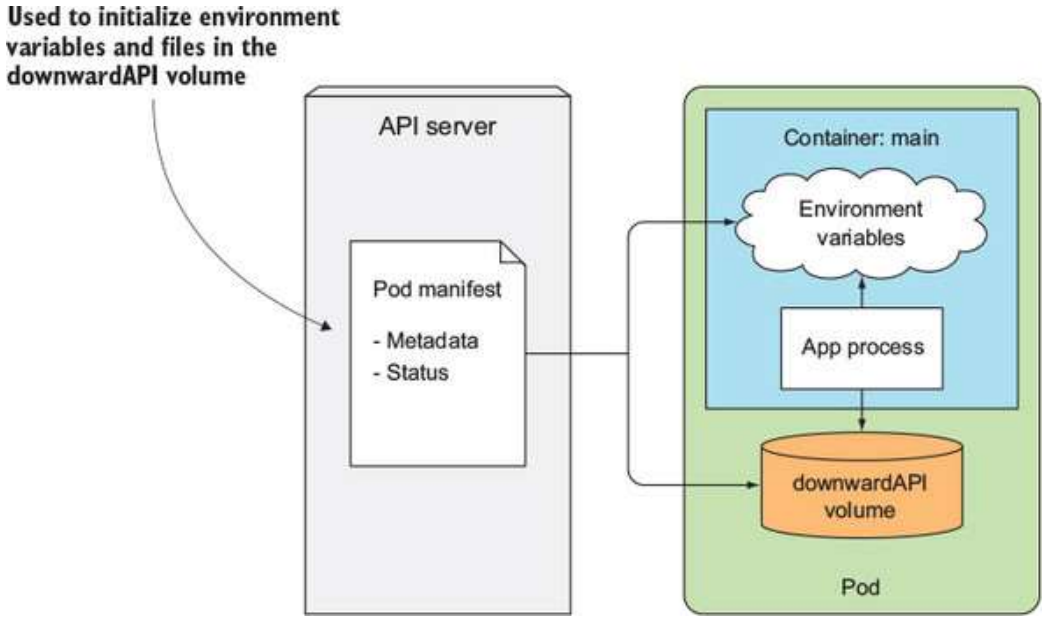
///

Volume - downwardAPI

Why downwardAPI

- Pod의 IP, 호스트 노드 이름, Pod 자체의 이름과 같이 실행 시점까지 알려지지 않은 데이터를 얻기 위한 방법

  . downwardAPI는 애플리케이션이 호출해서 데이터를 가져오는 REST Endpoint와는 다르다.



※ 참고 : https://livebook.manning.com/book/kubernetes-in-action/chapter-8/11

capabilities of downwardAPI

Information available via	item	comment
fieldRef:	metadata.name	the pod's name
	metadata.namespace	the pod's namespace
	metadata.uid	the pod's UID
	metadata.labels['<KEY>']	the value of the pod's label <KEY> (for example, metadata.labels['mylabel'])
	metadata.annotations['<KEY>']	the value of the pod's annotation <KEY> (for example, metadata.annotations['myannotation'])
resourceFieldRef:	A Container's CPU limit	
	A Container's CPU request	
	A Container's memory limit	
	A Container's memory request	
	A Container's hugepages limit	providing that the DownwardAPIHugePages feature gate is enabled
	A Container's hugepages request	providing that the DownwardAPIHugePages feature gate is enabled
	A Container's ephemeral-storage limit	
	A Container's ephemeral-storage request	
downwardAPI volume fieldRef:	metadata.labels	all of the pod's labels, formatted as label-key="escaped-label-value" with one label per line
	metadata.annotations	all of the pod's annotations, formatted as annotation-key="escaped-annotation-value" with one annotation per line
environment variables:	status.podIP	the pod's IP address
	spec.serviceAccountName	the pod's service account name, available since v1.4.0-alpha.3
	spec.nodeName	the node's name, available since v1.4.0-alpha.3
	status.hostIP	the node's IP, available since v1.7.0-alpha.1

※ 참고 : https://kubernetes.io/docs/tasks/inject-data-application/downward-api-volume-expose-pod-information/#capabilities-of-the-downward-api

YAML

pod-downwardAPI.yaml

```
apiVersion: v1
kind: Pod

metadata:
  name: kubernetes-downwardapi-volume-example
  labels:
    zone: us-est-coast
    cluster: test-cluster1
    rack: rack-22
  annotations:
    build: two
    builder: john-doe

spec:
  containers:
    - name: client-container
      image: k8s.gcr.io/busybox
      command: ["sh", "-c"]

      args:
      - while true; do
          if [[ -e /etc/podinfo/labels ]]; then
            echo -en '₩n₩n'; cat /etc/podinfo/labels; fi;
          if [[ -e /etc/podinfo/annotations ]]; then
            echo -en '₩n₩n'; cat /etc/podinfo/annotations; fi;
          sleep 5;
        done;
      volumeMounts:
        - name: podinfo
          mountPath: /etc/podinfo

  volumes:
    - name: podinfo
      downwardAPI:
        items:
          - path: "labels"
            fieldRef:
              fieldPath: metadata.labels
          - path: "annotations"
            fieldRef:
              fieldPath: metadata.annotations
```

※ 참고 : https://k8s.io/examples/pods/inject/dapi-volume.yaml

Execute

```
remote ❯ cd kubernetes/07-ConfigMap-Secret-downwardAPI/hands-on


remote ❯ kubectl create -f pod-downwardAPI.yaml


remote ❯ kubectl get pods -o wide

NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE      NOMINATED NODE   READINESS GATES
kubernetes-downwardapi-volume-example   1/1     Running   0          8s    10.233.103.115   worker2   <none>           <none>


remote ❯ kubectl logs kubernetes-downwardapi-volume-example

cluster="test-cluster1"
rack="rack-22"
zone="us-est-coast"

build="two"
builder="john-doe"
kubernetes.io/config.seen="2022-09-03T07:45:58.731423677+09:00"
kubernetes.io/config.source="api"
...
```

///

KubeCraftAdmin

https://eric-jadi.medium.com/minecraft-as-a-k8s-admin-tool-cf16f890de42

https://github.com/erjadi/kubecraftadmin

///

GCP - GKE (Google Kubernetes Engine)

Go to console.cloud.google.com and log in.

Go to and enable the Kubernetes Engine API.

```
> gcloud container clusters create whatwant-school --num-nodes 3 --machine-type g1-small --no-enable-autoupgrade --zone us-central1-a
```

```
> gcloud container clusters delete whatwant-school --zone us-central1-a
```

///

쿠버네티스 동화

※ 참고 : https://www.youtube.com/watch?v=4ht22ReBjno

///