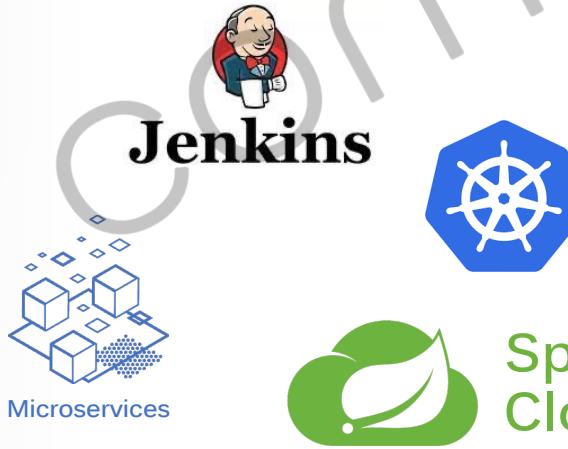


JenKins 를 이용한 CI/CD Pipeline 구축



```
public static void main(String[] args)
{
    < servlet >
        < servlet-name >BoardController</servlet-name>
        < servlet-class >com.joneconsulting.controller.BoardController</servlet-class>
        < init-param >
            < param-name >user_name</param-name>
            < param-value >Kenneth Lee</param-value>
        </init-param >
    </servlet >
```

```
    < servlet >
        < servlet-name >BookController</servlet-name>
        < servlet-class >com.joneconsulting.controller.BookController</servlet-class>
        < init-param >
            < param-name >self_title, price, author</param-name>
            < param-value >self_title
                self_price
                self_author = author
            </param-value>
        </init-param >
    </servlet >
```

```
    < servlet >
        < servlet-name >UIApplicationDelegate</servlet-name>
        < servlet-class >com.joneconsulting.UIApplicationDelegate</servlet-class>
        < init-param >
            < param-name >NEXT_INNOVATION_DELEGATE</param-name>
            < param-value >@Interface NextInnovationDelegate : NSObject < UIApplicationDelegate >
                ...
            </param-value>
        </init-param >
    </servlet >
```



프로필

Dowon Lee



지식공유자 인증

5452 ★ 4.8(420)

멘토링 활성



- 홈
- 강의
- 로드맵
- 수강후기
- 블로그

최신순 ▾

강의 (3)



Spring Cloud로 개발하는 마이크로서비스 애플리케이션(MSA)

Dowon Lee

★★★★★ (174)

학습중

+3000명 독점 할인중



Spring Boot를 이용한 RESTful Web Services 개발

Dowon Lee

★★★★★ (308)

학습중

+2700명 독점 할인중



웹 애플리케이션 개발을 위한 IntelliJ IDEA 설정

Dowon Lee

★★★★★ (228)

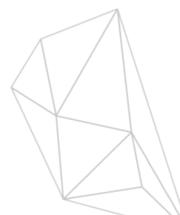
학습중

+3500명



목차

- Section 1: DevOps와 CI/CD
- Section 2: Jenkins를 이용한 CI/CD 사용
- Section 3: Jenkins + Infrastructure as Code
- **Section 4: Jenkins + Ansible + Kubernetes 연동**
- Section 5: Advanced Jenkins 사용
- Section 6: Public Cloud에 배포
- Appendix



Section 4.

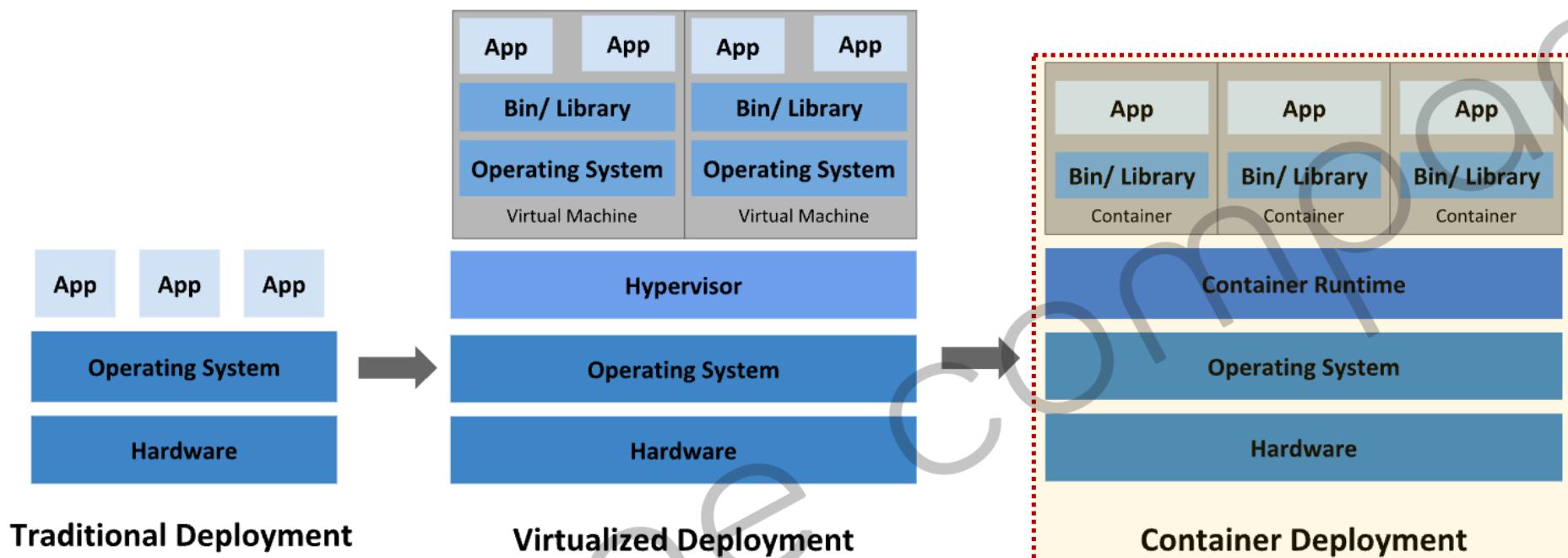
Jenkins + Ansible + Kubernetes 연동

- Kubernetes Cluster
- Kubernetes 설치 (Minikube)
- Kubernetes Cluster에 배포하기
- Ansible + Kubernetes 연동
- CI/CD 프로세스

Container Virtualization

njone company

- <https://kubernetes.io/ko/docs/concepts/overview/what-is-kubernetes/>



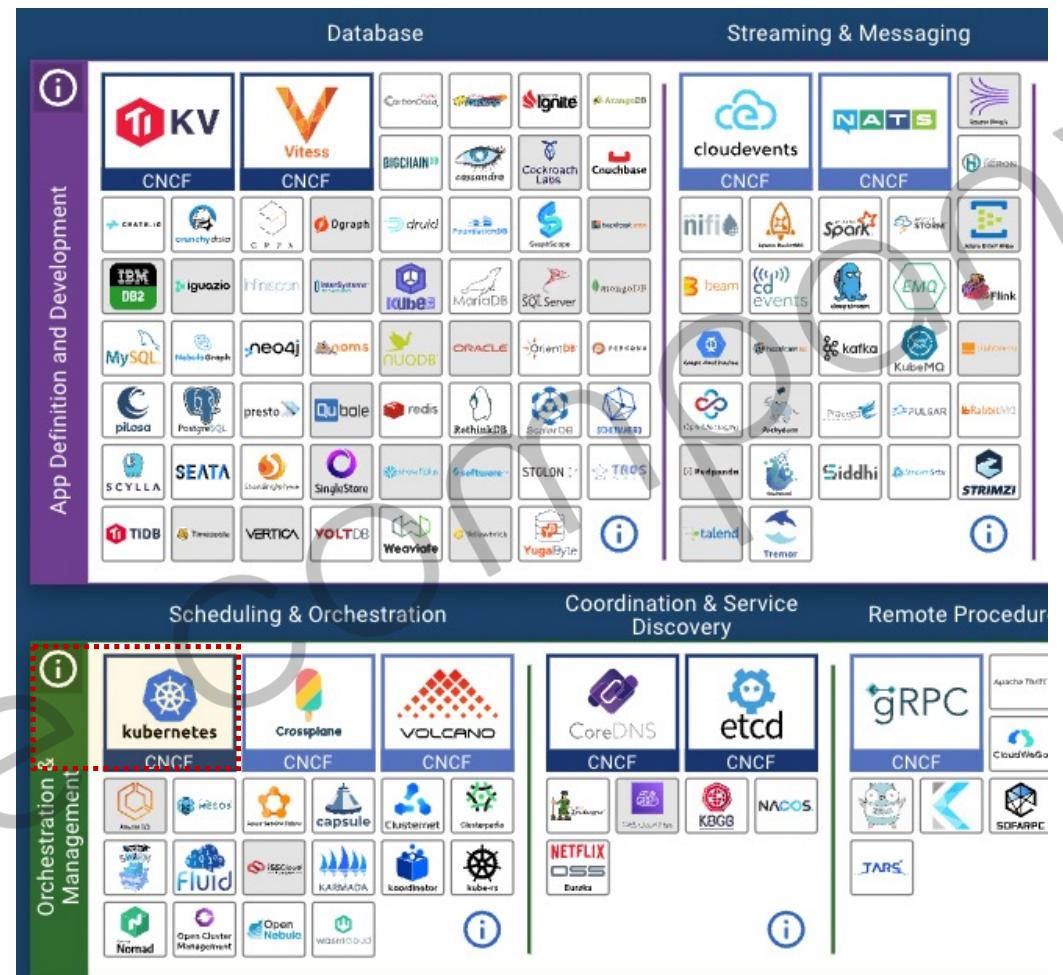
CNCF Cloud Native Interactive Landscape

njone company

- <https://landscape.cncf.io/>

- Kubernetes(K8s)

- 오픈소스 기반의 컨테이너화 된 애플리케이션(워크로드와 서비스)의 자동 배포, 스케일링 등을 제공하는 관리 플랫폼





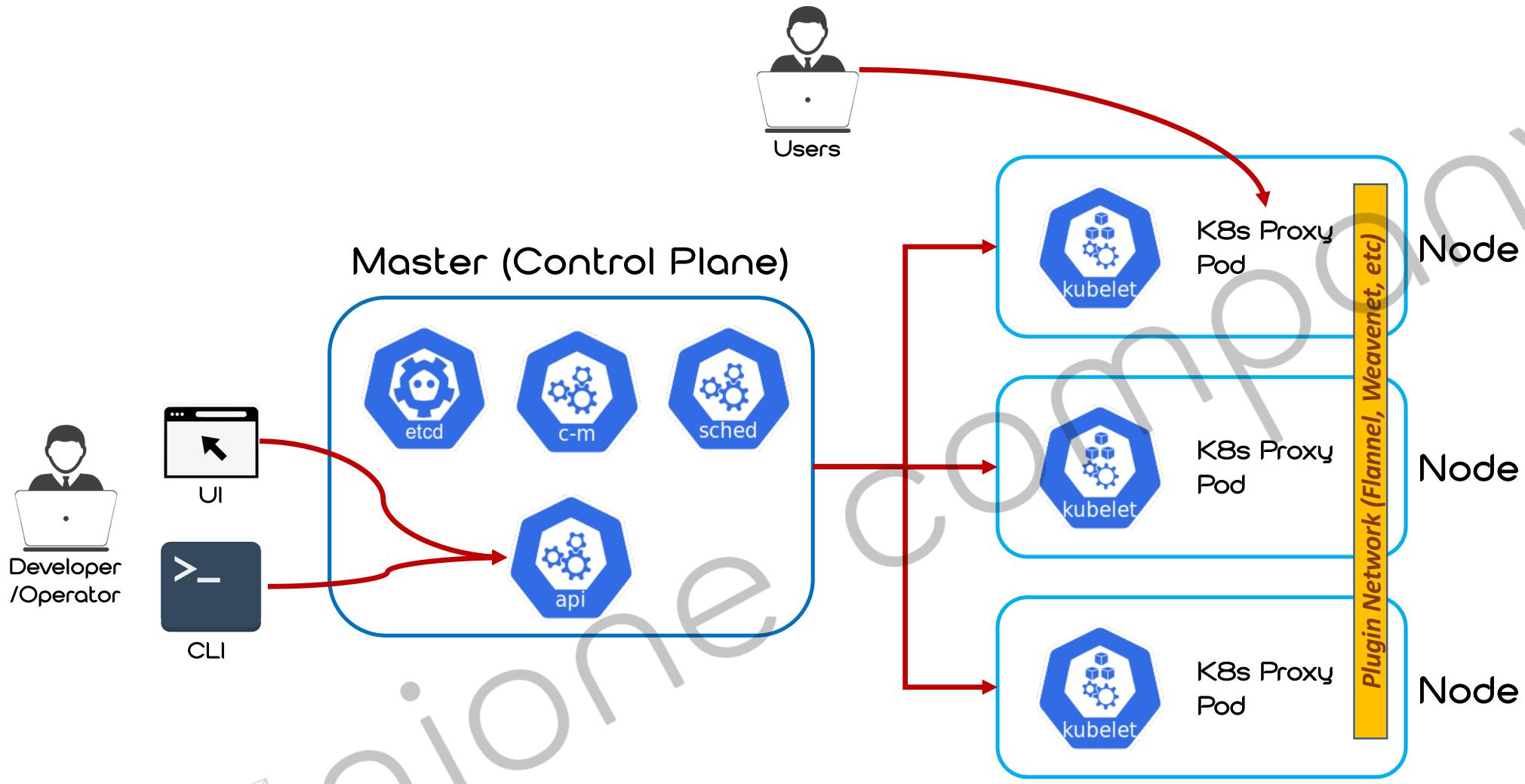
Kubernetes

njone company

- 컨테이너화 된 애플리케이션 구동
- 서비스 디스커버리와 로드 밸런싱
- 스토리지 오케스트레이션
- 자동화된 롤아웃과 롤백
- 자동화된 빈 패킹(bin packing)
- 자동화된 복구(self-healing)
- 시크릿과 구성 관리
- 소스 코드 배포 X, 빌드 X
- 애플리케이션 레벨 서비스 X
- 로깅, 모니터링 솔루션 X
- 포괄적인 머신 설정, 유지보수, 관리, 자동 복구 시스템을 제공 X

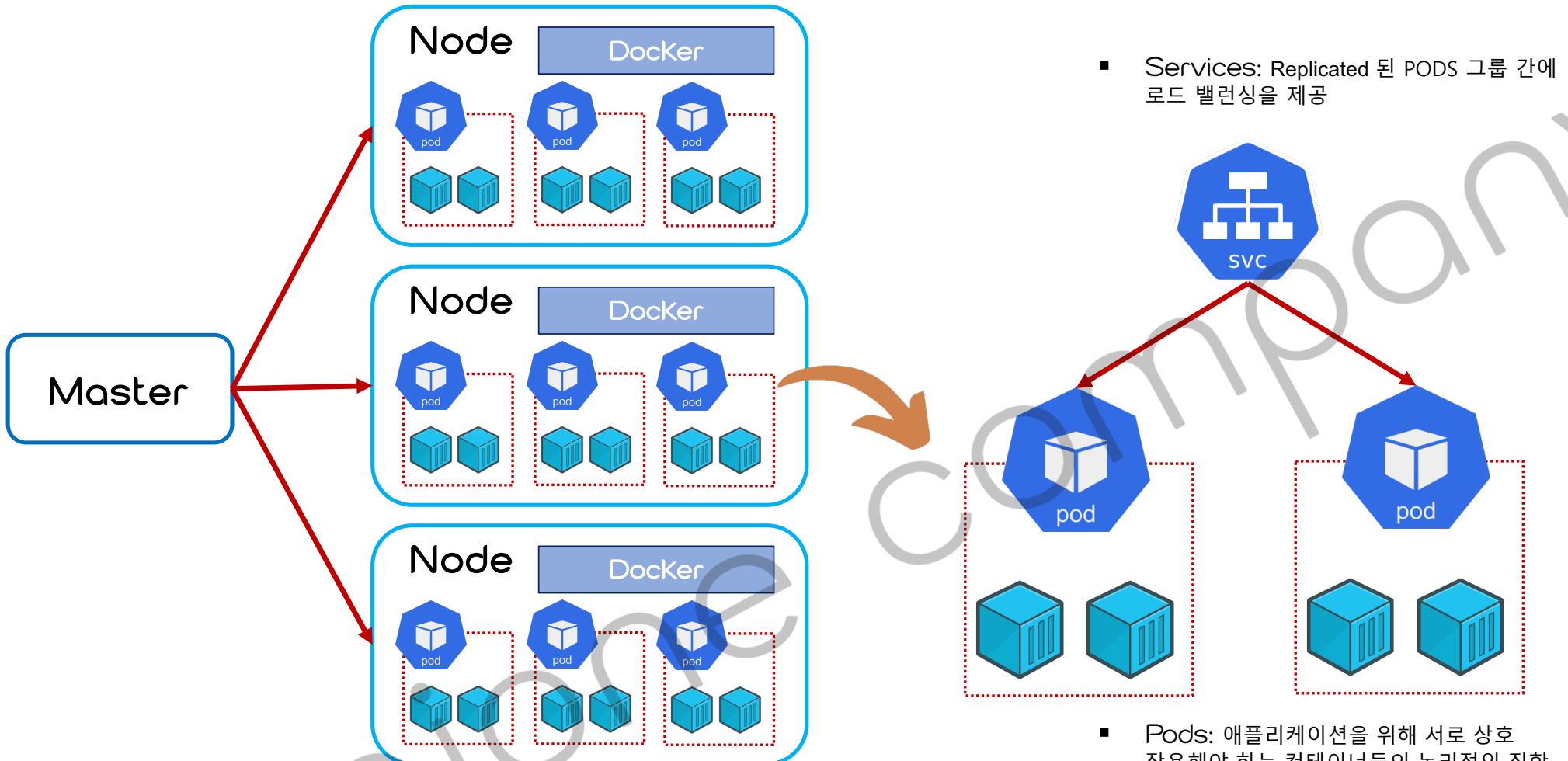
Kubernetes Cluster

njone company



Working of Kubernetes

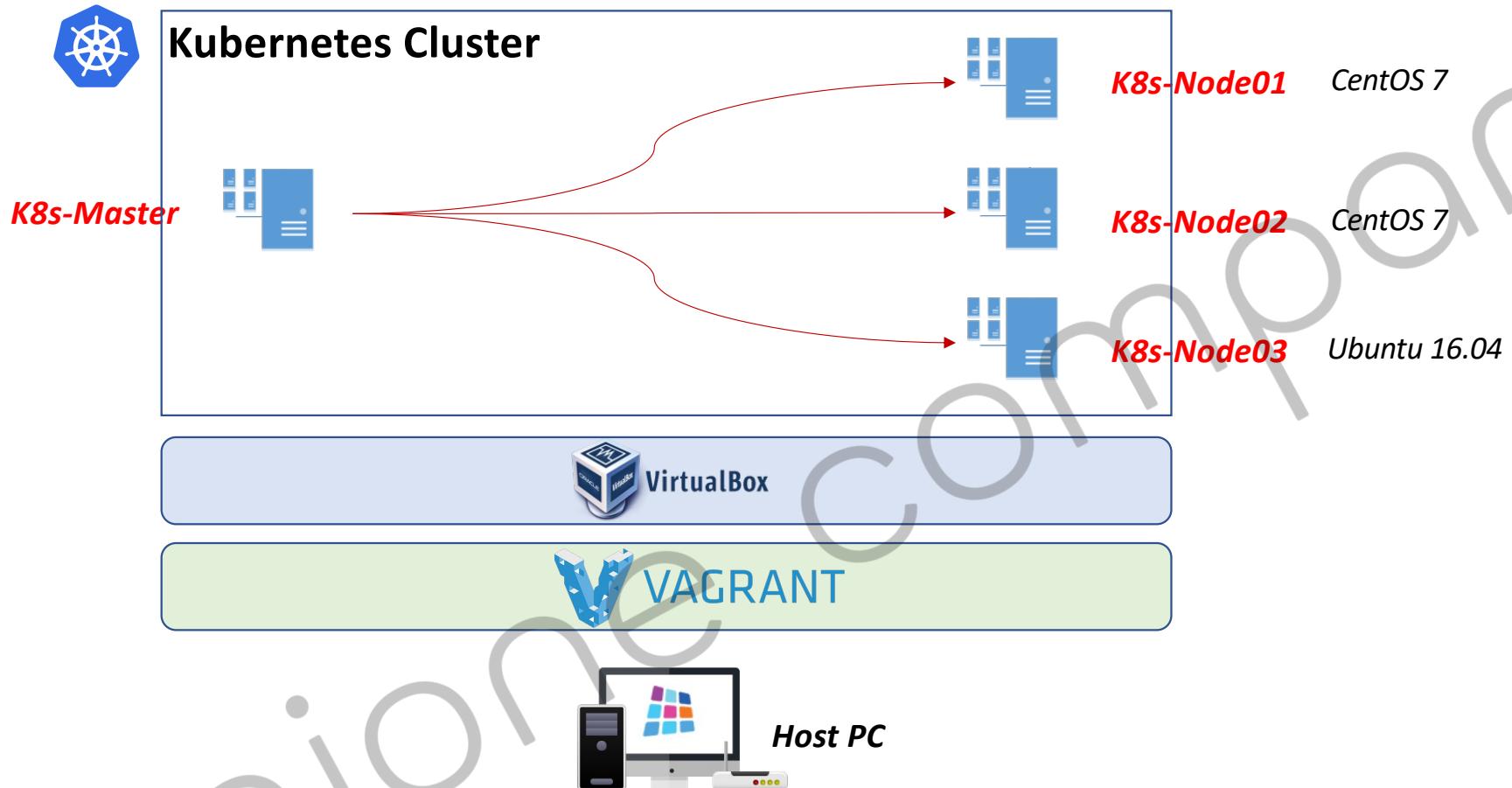
njone company



Install Kubernetes Cluster

njone company

- VM을 이용한 리눅스 설치



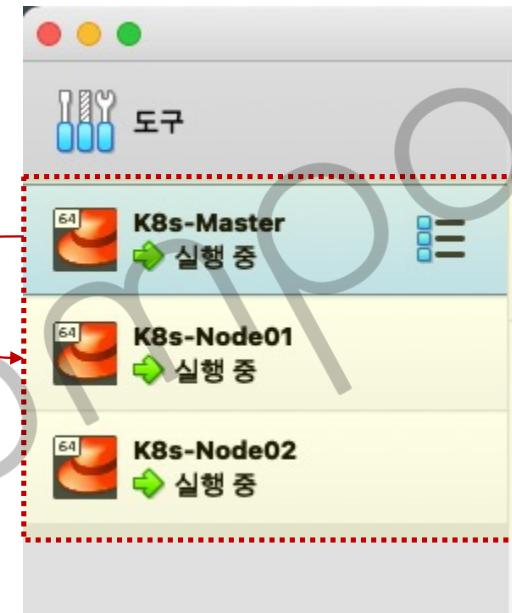
Install Kubernetes Cluster

njone company

- https://github.com/joneconsulting/k8s/blob/master/install/kubernetes_install.md

```
vagrant status  
Current machine states:  
  
k8s-node01      running (virtualbox)  
k8s-node02      running (virtualbox)  
k8s-master      running (virtualbox)
```

```
[root@k8s-master ~]# kubectl get nodes  
NAME        STATUS    ROLES      AGE     VERSION  
k8s-master   Ready     control-plane,master   24h    v1.20.5  
k8s-node01   Ready     <none>      24h    v1.20.5  
k8s-node02   Ready     <none>      24h    v1.20.5
```

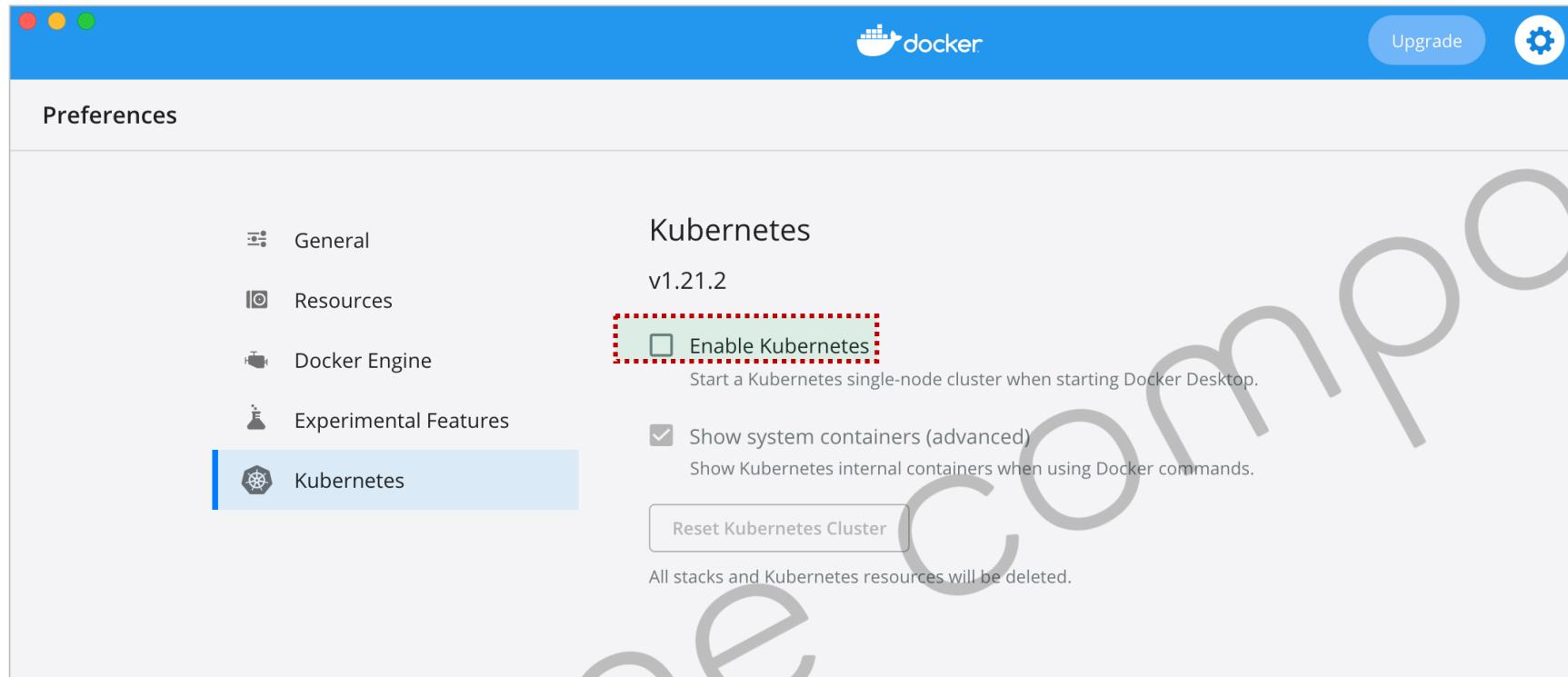




Install Minikube

njone company

- Exercise Kubernetes on *Minikube*





Kubernetes – Test

njone company

- `$ kubectl get nodes`

```
▶ kubectl get nodes
NAME        STATUS   ROLES      AGE    VERSION
docker-desktop   Ready   control-plane  55d    v1.24.0
```

- `$ kubectl get pods (or deployments, or services)`

```
▶ kubectl get pods
NAME        READY   STATUS    RESTARTS   AGE
hello-pod   1/1     Running   1 (15m ago) 54d
(base) edowon ➤ ~
▶ kubectl get deployments
No resources found in default namespace.
(base) edowon ➤ ~
▶ kubectl get services
NAME        TYPE       CLUSTER-IP      EXTERNAL-IP    PORT(S)        AGE
hello-svc   NodePort   10.100.207.206 <none>        8080:31145/TCP 54d
kubernetes  ClusterIP  10.96.0.1      <none>        443/TCP       55d
```

Kubernetes – Test

njone company

- 방법1)

```
$ kubectl run sample-nginx --image=nginx --port=80
```

- 방법2)

```
$ kubectl create deployment sample-nginx --image=nginx
```

```
$ kubectl scale deployment sample-nginx --replicas=2
```

- 방법3)

```
$ kubectl apply -f sample1.yml
```

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: nginx-deployment
5    labels:
6      app: nginx
7  spec:
8    replicas: 2
9    selector:
10   matchLabels:
11     app: nginx
12   template:
13     metadata:
14       labels:
15         app: nginx
16     spec:
17       containers:
18         - name: nginx
19           image: nginx:1.14.2
20           ports:
21             - containerPort: 80
```

Kubernetes – Test

njone company

- `$ kubectl get pods`
- `$ kubectl get pods -o wide`

```
▶ kubectl get deployments
NAME           READY  UP-TO-DATE  AVAILABLE  AGE
nginx-deployment  2/2    2          2          2m16s
(base) downonlee ➤ ~/Desktop/Work/14.online/devops/k8s
▶ kubectl get pods
NAME           READY  STATUS   RESTARTS  AGE
nginx-deployment-66b6c48dd5-bfs4l  1/1    Running  0          2m19s
nginx-deployment-66b6c48dd5-qg8zx  1/1    Running  0          2m19s
(base) downonlee ➤ ~/Desktop/Work/14.online/devops/k8s
▶ kubectl get pods -o wide
NAME           READY  STATUS   RESTARTS  AGE   IP           NODE  NOMINATED NODE  READINESS GATES
nginx-deployment-66b6c48dd5-bfs4l  1/1    Running  0          3m56s  10.1.0.149  docker-desktop  <none>        <none>
nginx-deployment-66b6c48dd5-qg8zx  1/1    Running  0          3m56s  10.1.0.148  docker-desktop  <none>        <none>
```

- `$ kubectl exec -it nginx-deployment-66b6c48dd5-qg8zx -- /bin/bash`
 - `root@nginx-deployment-66b6c48dd5-qg8zx:# apt-get update`
 - `root@nginx-deployment-66b6c48dd5-qg8zx:# apt-get install -y curl wget`
 - `root@nginx-deployment-66b6c48dd5-qg8zx:# hostname -i`
 - `root@nginx-deployment-66b6c48dd5-qg8zx:# curl -X GET http://10.10.0.148`



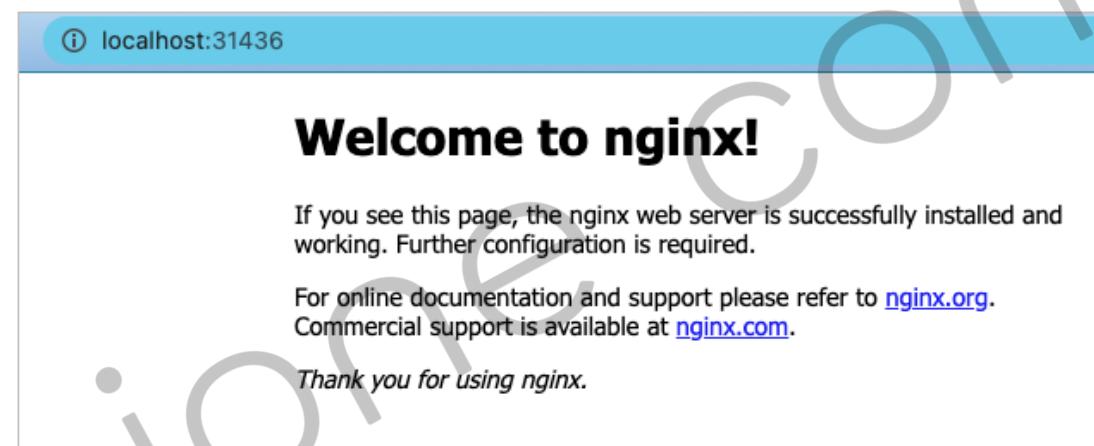
Kubernetes – Test

njone company

- Expose the deployment as service

- `$ kubectl expose deployment nginx-deployment --port=80 --type=NodePort`

```
▶ kubectl expose deployment nginx-deployment --port=80 --type=NodePort
service/nginx-deployment exposed
(base) dowonlee ~/Desktop/Work/14.online/devops/k8s
▶ kubectl get services
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
kubernetes     ClusterIP  10.96.0.1    <none>        443/TCP       43d
nginx-deployment  NodePort   10.111.25.43 <none>        80:31436/TCP  4s
```





Kubernetes – Test

njone company

- Display the pods

- *\$ kubectl get pods*

```
▶ kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-deployment-66b6c48dd5-bfs4l  1/1     Running   0          23m
nginx-deployment-66b6c48dd5-qg8zx  1/1     Running   0          23m
```

- Delete the pods

- *\$ kubectl delete pod nginx-deployment-66b6c48dd5-qg8zx*

```
▶ kubectl delete pod nginx-deployment-66b6c48dd5-qg8zx
pod "nginx-deployment-66b6c48dd5-qg8zx" deleted
(base) downonlee ~/Desktop/Work/14.online/devops/k8s▶
▶ kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-deployment-66b6c48dd5-bfs4l  1/1     Running   0          23m
nginx-deployment-66b6c48dd5-pn96x  1/1     Running   0          4s
```

- Delete the deployment

- *\$ kubectl delete deployment nginx-deployment*

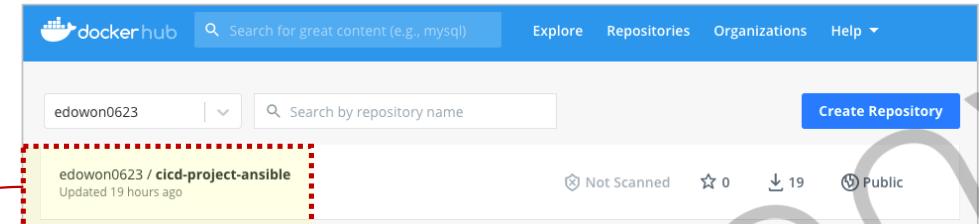


Kubernetes – Test

njone company

- Create deployment and service file

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: cicd-deployment
5  spec:
6    selector:
7      matchLabels:
8        app: cicd-devops-project
9    replicas: 2
10
11  template:
12    metadata:
13      labels:
14        app: cicd-devops-project
15    spec:
16      containers:
17        - name: cicd-devops-project
18          image: edowon0623/cicd-project-ansible
19          imagePullPolicy: Always
20          ports:
21            - containerPort: 8080
```



```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: cicd-service
5    labels:
6      app: cicd-devops-project
7  spec:
8    selector:
9      app: cicd-devops-project
10   type: NodePort
11   ports:
12     - port: 8080
13       targetPort: 8080
14       nodePort: 32000
```

Kubernetes – Test

njone company

- Create deployment and services

- `$ kubectl apply -f cicd-devops-deployment.yml`
- `$ kubectl apply -f cicd-devops-service.yml`

```
▶ kubectl apply -f cicd-devops-deployment.yml
deployment.apps/cicd-deployment created
(base) downonlee ➤ ~/Desktop/Work/14.online/devops/k8s
▶ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
cicd-deployment 1/2       2           1           5s
```

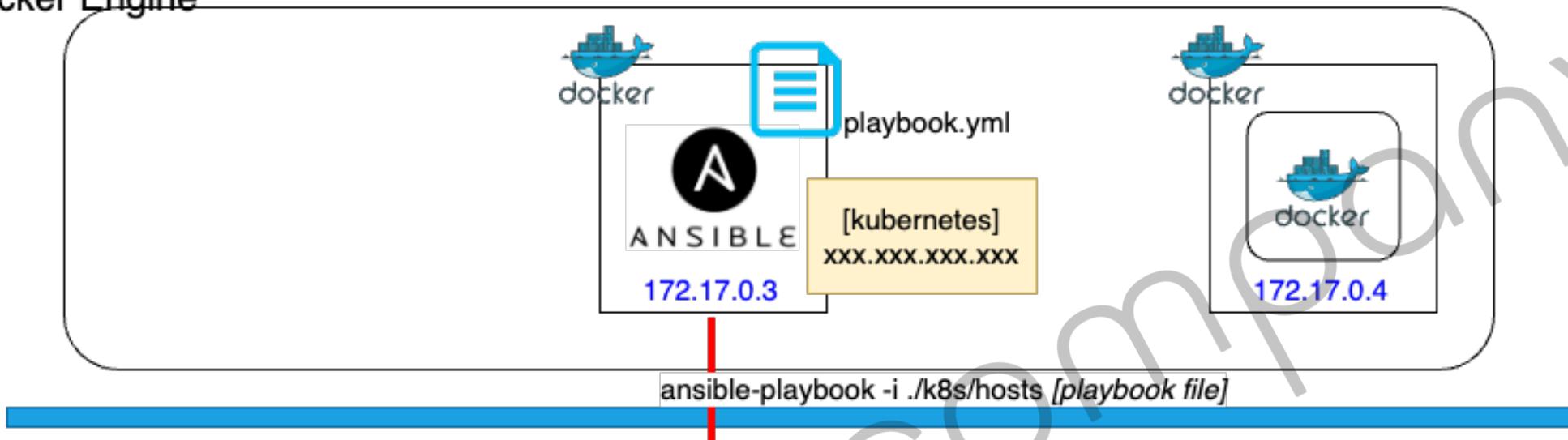
```
▶ kubectl apply -f cicd-devops-service.yml
service/cicd-service created
(base) downonlee ➤ ~/Desktop/Work/14.online/devops/k8s
▶ kubectl get services
NAME           TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
cicd-service   NodePort    10.111.0.95   <none>        8080:32000/TCP  3s
kubernetes     ClusterIP   10.96.0.1     <none>        443/TCP         43d
```



Kubernetes + Ansible

njone company

Docker Engine

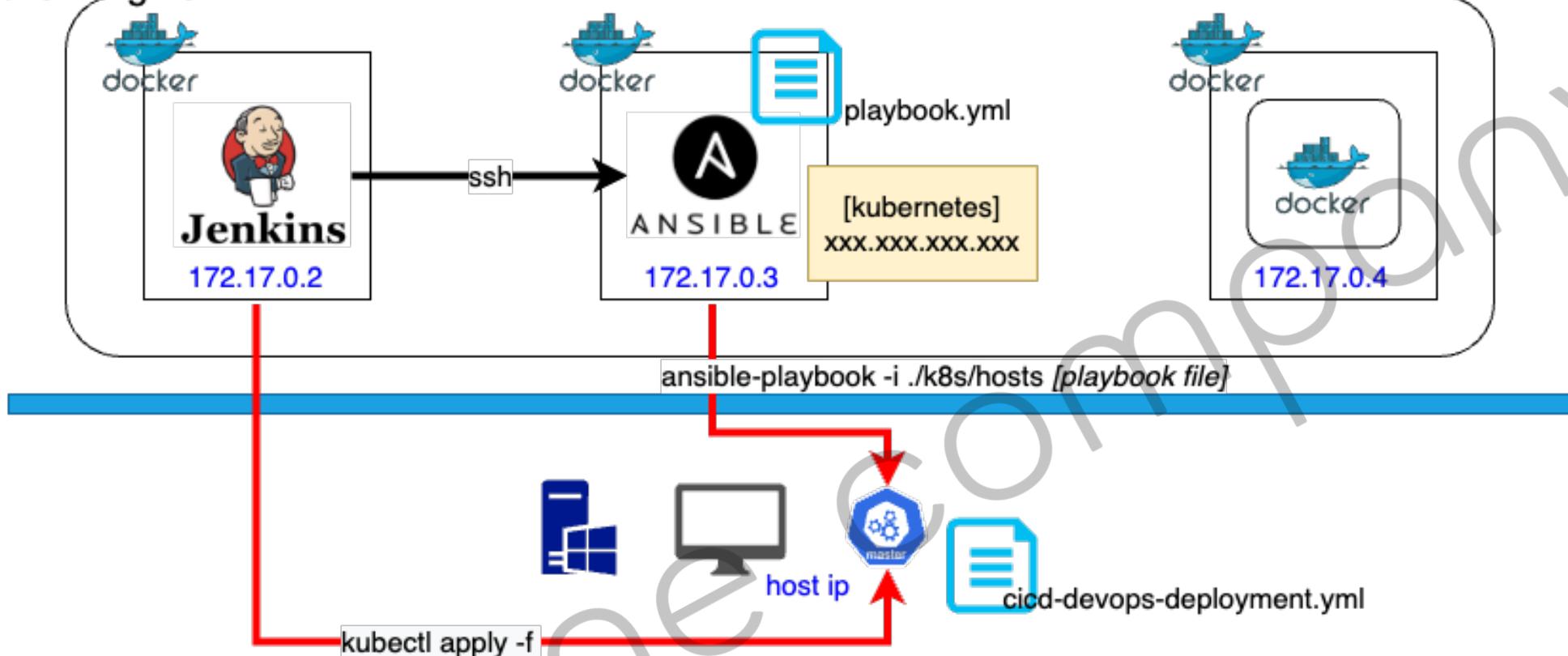


`ansible-playbook -i ./k8s/hosts [playbook file]`

Kubernetes + Ansible

njone company

Docker Engine





Kubernetes + Ansible

njone company

- Ansible Server) k8s-master 접속 테스트

```
▶ docker exec -it ansible-server bash
[root@ansible-server ~]# ssh root@192.168.32.10
root@192.168.32.10's password:
Last login: Wed Sep 15 02:01:46 2021 from 192.168.32.1
[root@k8s-master ~]# █
```

- Ansible Server) k8s/hosts 파일 작성

```
[root@ansible-server ~]# mkdir k8s
[root@ansible-server ~]# cd k8s
[root@ansible-server k8s]# vi hosts
```

```
[ansible-server]
localhost

[kubernetes]
192.168.32.10
```



Kubernetes + Ansible

njone company

- Ansible Server) ssh-keygen, ssh-copy-id

```
[root@ansible-server ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:W/dfJc0v6vkG1bCC8cxZ6nbBwMoQM28W5P/vUhU3RKY root@ansible-server
The key's randomart image is:
+---[RSA 2048]----+
|   oo. o+ |
| .=..o =o.|
| o=B E ++|
| o+.B =oo|
| S ..oo..=|
| o .+o...+|
| . . o+.o|
|     oo+.|
|     .+o.o+|
+---[SHA256]----+
```

- \$ **ansible -i ./k8s/hosts kubernetes -m ping**

```
[root@ansible-server ~]# ansible -i ./k8s/hosts kubernetes -m ping
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details
192.168.32.10 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

```
[root@ansible-server ~]# ssh-copy-id root@192.168.32.10
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@192.168.32.10's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'root@192.168.32.10'"
and check to make sure that only the key(s) you wanted were added.
```

Kubernetes + Ansible

njone company

- K8s Master) create a deployment yaml file

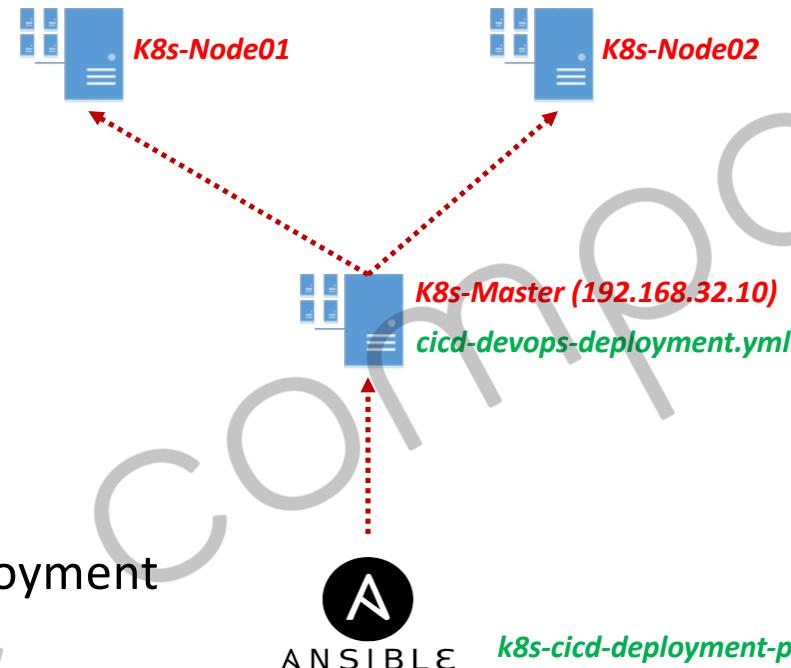
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: cicd-deployment
spec:
  selector:
    matchLabels:
      app: cicd-devops-project
  replicas: 2

  template:
    metadata:
      labels:
        app: cicd-devops-project
    spec:
      containers:
        - name: cicd-devops-project
          image: edowon0623/cicd-project-ansible
          imagePullPolicy: Always
          ports:
            - containerPort: 8080
```

- Ansible Server) create a playbook file for deployment

```
- name: Create pods using deployment
  hosts: kubernetes
  # become: true
  user: root

  tasks:
    - name: create a deployment
      command: kubectl apply -f cicd-devops-deployment.yml
```





Kubernetes + Ansible

njone company

- Ansible Server) execute a playbook file (for deployment)

```
[root@ansible-server ~]# ansible-playbook -i ./k8s/hosts k8s-cicd-deployment-playbook.yml
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details

PLAY [Create pods using deployment] ****
TASK [Gathering Facts] ****
ok: [192.168.32.10]

TASK [create a deployment] ****
changed: [192.168.32.10]

PLAY RECAP ****
192.168.32.10      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@k8s-master ~]# kubectl get all -o wide
NAME                                         READY   STATUS    RESTARTS   AGE     IP           NODE       NOMINATED NODE   READINESS GATES
pod/cicd-deployment-54f8446c9c-dhgqz   1/1     Running   0          17m    192.168.58.193   k8s-node02   <none>        <none>
pod/cicd-deployment-54f8446c9c-kskmh   1/1     Running   0          17m    192.168.85.193   k8s-node01   <none>        <none>

NAME            TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE     SELECTOR
service/cicd-service  NodePort    10.108.67.58  <none>        8080:32000/TCP  11m    app=cicd-devops-project
service/kubernetes  ClusterIP   10.96.0.1    <none>        443/TCP       34m    <none>

NAME           READY   UP-TO-DATE   AVAILABLE   AGE     CONTAINERS   IMAGES
deployment.apps/cicd-deployment  2/2     2           2          17m    cicd-devops-project   edowon0623/cicd-project-ansible   app=cicd-d
NAME           DESIRED   CURRENT   READY   AGE     CONTAINERS   IMAGES
replicaset.apps/cicd-deployment-54f8446c9c  2        2         2          17m    cicd-devops-project   edowon0623/cicd-project-ansible   app=cicd-d
```



Kubernetes + Ansible

njone company

- Ansible Server) execute a playbook file (for service)

```
[root@ansible-server ~]# ansible-playbook -i ./k8s/hosts k8s-cicd-service-playbook.yml
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details

PLAY [create service for deployment] ****
TASK [Gathering Facts] ****
ok: [192.168.32.10]

TASK [create a service] ****
changed: [192.168.32.10]

PLAY RECAP ****
192.168.32.10 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@k8s-master ~]# kubectl get service -o wide
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE   SELECTOR
cicd-service   NodePort   10.110.70.178 <none>        8080:32000/TCP   85s  app=cicd-devops-project
kubernetes     ClusterIP  10.96.0.1    <none>        443/TCP         113m  <none>
```



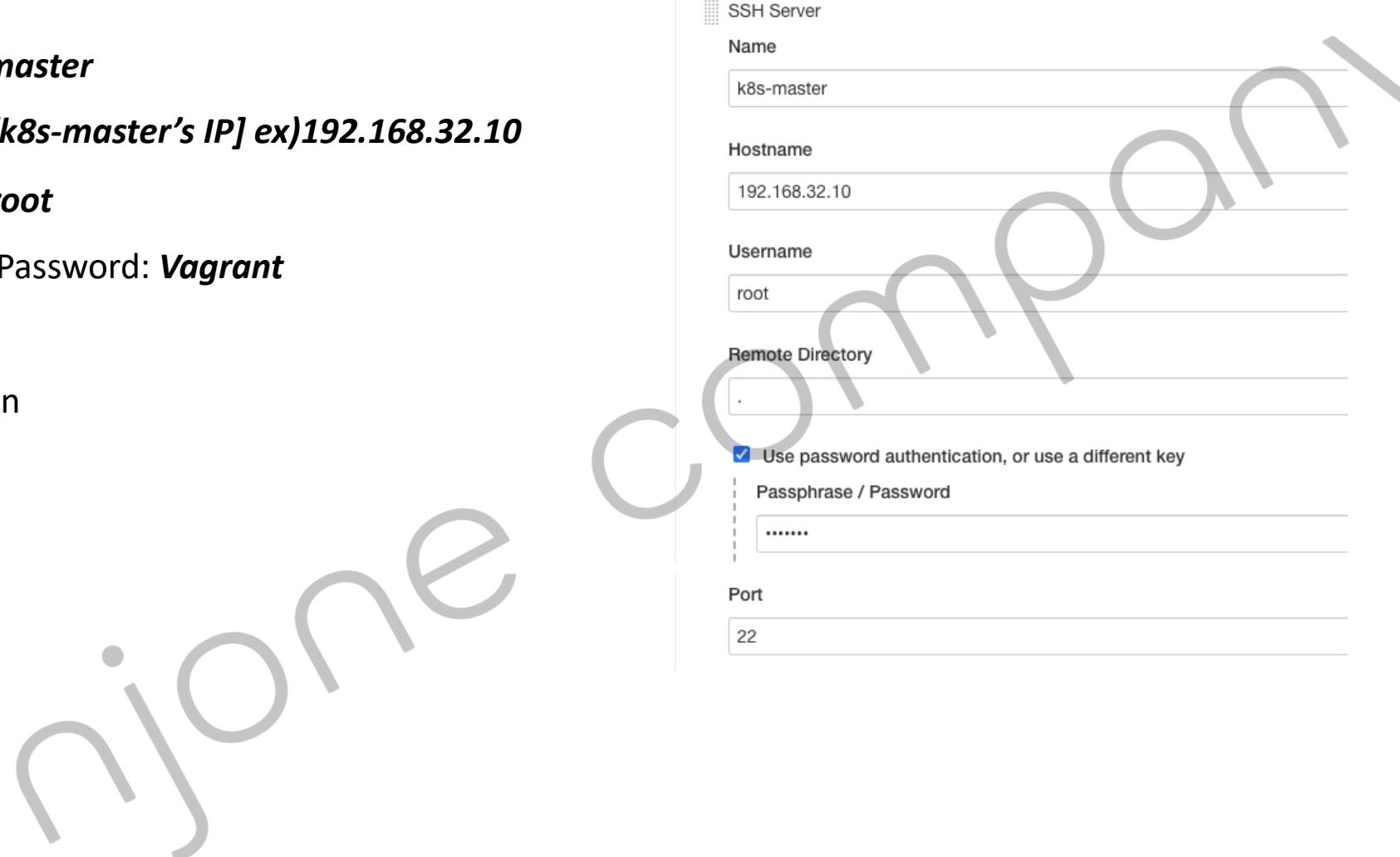
```
{
  "id": 1,
  "content": "Hello, Kenneth!",
  "version": "3.0",
  "ip": "10.1.0.36"
}
```



Setup Ansible on Jenkins

njone company

- Manage Jenkins → Configure System → Publish over SSH
 - Add SSH Servers
 - Name: ***k8s-master***
 - Hostname: **[*k8s-master's IP*] ex)192.168.32.10**
 - Username: ***root***
 - Passphrase/Password: ***Vagrant***
 - Port: **22**
 - Test Configuration



SSH Server

Name
k8s-master

Hostname
192.168.32.10

Username
root

Remote Directory
.

Use password authentication, or use a different key

Passphrase / Password
.....

Port
22

Exercise 7# JenKins Job 1/5

njone company

- Item name → ***My-K8s-Project***
 - Copy from: ***My-Ansible-Project***

- Build Triggers
 - SSH Server: ***k8s-master***

- Exec command
 - ***kubectl apply -f cicd-devops-deployment.yml***

SSH Server
Name ?
k8s-master

고급...

Transfers

Transfer Set
Source files ?
target/*.war

Remove prefix ?
target

Remote directory ?
.

Exec command ?
kubectl apply -f cicd-devops-deployment.yml

Exercise 7# JenKins Job 2/5

njone company

■ Build Now

```
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 7.200 s
[INFO] Finished at: 2021-09-15T04:20:46Z
[INFO] -----
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project/webapp/pom.xml to com.edowan0623/cicd-devops-deployment:1.0-SNAPSHOT.pom
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project/webapp/target/webapp-1.0-SNAPSHOT.war
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project/server/pom.xml to com.edowan0623/cicd-devops-deployment:1.0-SNAPSHOT.pom
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project/server/target/server-1.0-SNAPSHOT.jar
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project/pom.xml to com.example.cicd-devops-deployment:1.0-SNAPSHOT.pom
channel stopped
SSH: Connecting from host [512076bd5890]
SSH: Connecting with configuration [k8s-master] ...
SSH: EXEC: completed after 406 ms
SSH: Disconnecting c
```

```
[root@k8s-master ~]# kubectl get all -o wide
SSH: Transferred 1 files
NAME                                         READY   STATUS    RESTARTS   AGE   IP           NODE      NOMINATED NODE   READINESS GATES
pod/cicd-deployment-54f8446c9c-4rlw4       1/1     Running   0          32s   192.168.58.194   k8s-node02   <none>        <none>
pod/cicd-deployment-54f8446c9c-6tj4d       1/1     Running   0          32s   192.168.85.194   k8s-node01   <none>        <none>

NAME              TYPE    CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE   SELECTOR
service/kubernetes   ClusterIP  10.96.0.1   <none>        443/TCP   37m   <none>

NAME                           READY   UP-TO-DATE   AVAILABLE   AGE   CONTAINERS   IMAGES   SELECTOR
deployment.apps/cicd-deployment   2/2     2           2           32s   cicd-devops-project   edowan0623/cicd-project-ansible   app=cicd-devops-project

NAME                           DESIRED   CURRENT   READY   AGE   CONTAINERS   IMAGES   SELECTOR
replicaset.apps/cicd-deployment-54f8446c9c   2         2         2           32s   cicd-devops-project   edowan0623/cicd-project-ansible   app=cicd-devops-project
```



Exercise 7# JenKins Job 3/5

njone company

- Item name → ***My-K8s-Project-using-Ansible***

- Copy from: ***My-K8s-Project***

- Build Triggers

- SSH Server: ***ansible-host***

- Exec command

- ***ansible-playbook -i ./k8s/hosts k8s-cicd-deployment-playbook.yml***

My-K8s-Project-using-Ansible

General 소스 코드 관리 빌드 유발 빌드 환경

설명

Deploy on K8s master using Ansible

SSH Server
Name ?
ansible-host

Transfers
Transfer Set
Source files ?
webapp/target/*.war

Remove prefix ?
webapp/target

Remote directory ?
. .

Exec command ?
ansible-playbook -i ./k8s/hosts k8s-cicd-deployment-playbook.yml

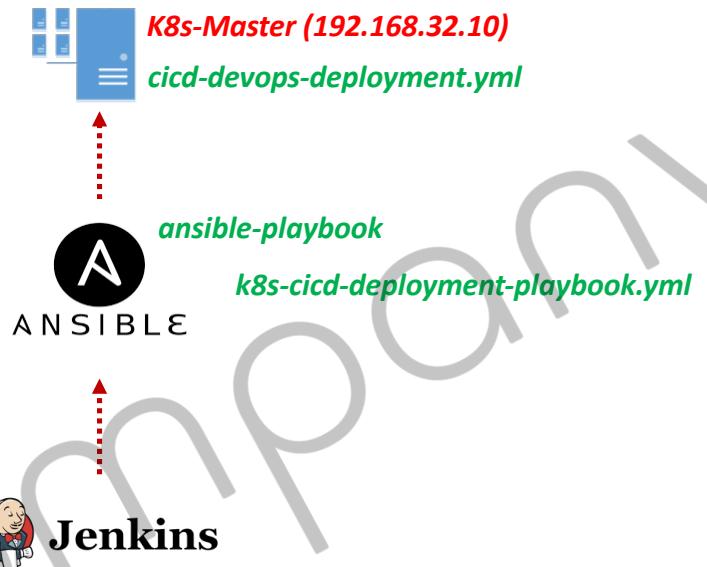
Exercise 7# JenKins Job 4/5

njone company

■ Build Now

```
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time: 5.039 s  
[INFO] Finished at: 2021-09-15T06:17:44Z  
[INFO] -----  
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project-using-Ansible/webapp/pom.xml  
SNAPSHOT/webapp-1.0-SNAPSHOT.pom  
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project-using-Ansible/webapp/target/w  
SNAPSHOT/webapp-1.0-SNAPSHOT.war  
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project-using-Ansible/server/pom.xml  
SNAPSHOT/server-1.0-SNAPSHOT.pom  
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project-using-Ansible/server/target/s  
SNAPSHOT/server-1.0-SNAPSHOT.jar  
[JENKINS] Archiving /var/jenkins_home/workspace/My-K8s-Project-using-Ansible/pom.xml to com.  
SNAPSHOT/maven-project-1.0-SNAPSHOT.pom  
channel stopped  
SSH: Connecting from host [512076bd5890]  
SSH: Connecting with configuration [ansible-host] ...  
SSH: Disconnecting configuration [ansible-host] ...  
SSH: Transferred 1 file(s)  
Finished: SUCCESS
```

```
[root@k8s-master ~]# kubectl get all -o wide  
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READIN  
pod/cicd-deployment-54f8446c9c-pxw6t 1/1 Running 0 29s 192.168.85.195 k8s-node01 <none> <none>  
pod/cicd-deployment-54f8446c9c-tdq7p 1/1 Running 0 29s 192.168.58.195 k8s-node02 <none> <none>  
  
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE SELECTOR  
service/kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 71m <none>  
  
NAME READY UP-TO-DATE AVAILABLE AGE CONTAINERS IMAGES  
deployment.apps/cicd-deployment 2/2 2 2 29s cicd-devops-project edowon0623/cicd-project-ansible  
  
NAME DESIRED CURRENT READY AGE CONTAINERS IMAGES  
replicaset.apps/cicd-deployment-54f8446c9c 2 2 2 29s cicd-devops-project edowon0623/cicd-project-an
```



Exercise 7# JenKins Job 5/5

njone company

- Append exec commands

- ***ansible-playbook -i ./k8s/hosts k8s-cicd-deployment-playbook.yml;***
- ***ansible-playbook -i ./k8s/hosts k8s-cicd-service-playbook.yml;***

The Jenkins pipeline configuration shows a Transfer Set step and an Exec command step.

Transfer Set:

- Source files: webapp/target/*.war
- Remove prefix: webapp/target
- Remote directory: .

Exec command:

```
ansible-playbook -i ./k8s/hosts k8s-cicd-deployment-playbook.yml;
ansible-playbook -i ./k8s/hosts k8s-cicd-service-playbook.yml;
```

Terminal Output (kubectl get all):

NAME	READY	STATUS	RESTARTS	AGE
pod/cicd-deployment-54f8446c9c-hkcj9	1/1	Running	0	22s
pod/cicd-deployment-54f8446c9c-lchpx	1/1	Running	0	22s

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
service/cicd-service	NodePort	10.110.52.2	<none>	8080:32000/TCP	19s
service/kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	157m

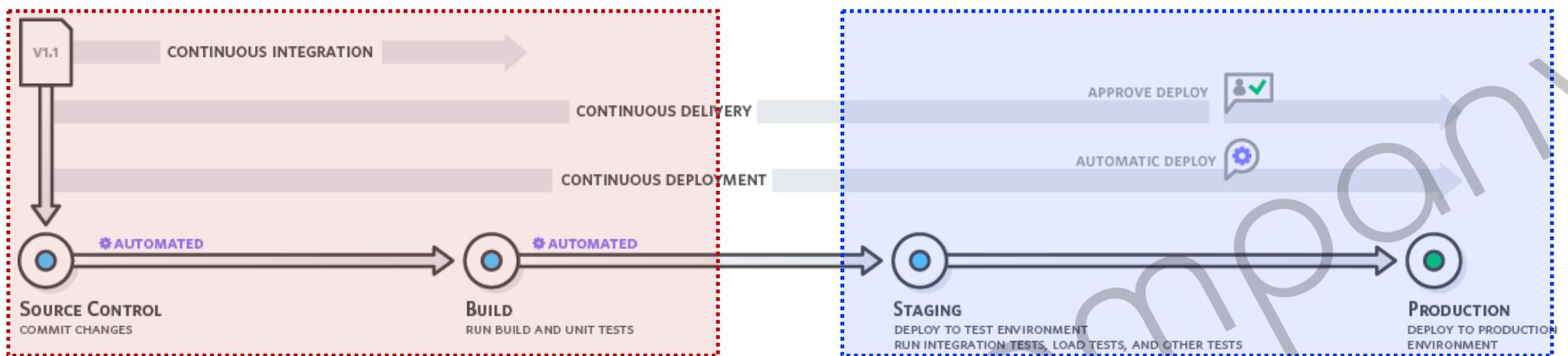
NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deployment.apps/cicd-deployment	2/2	2	2	22s

NAME	DESIRED	CURRENT	READY	AGE
replicaset.apps/cicd-deployment-54f8446c9c	2	2	2	22s

CI/CD Process

njone company

■ Jenkins CI/CD Jobs



■ CI Jobs

- *git pull*
- *create a docker image*
- *push the image to the registry (<https://hub.docker.com>)*
- *remove the image from the local*

■ CD Jobs

- *create a deployment (replicaset: 2)*
- *create a service*



Exercise 8# JenKins Job 1/6

njone company

- create-cicd-devops-image.yml

```
- hosts: all
#   become: true

tasks:
- name: create a docker image with deployed waf file
  command: docker build -t edowon0623/cicd-project-ansible .
  args:
    chdir: /root

- name: push the image on Docker Hub
  command: docker push edowon0623/cicd-project-ansible

- name: remove the docker image from the ansible server
  command: docker rmi edowon0623/cicd-project-ansible
  ignore_errors: yes
```

- Dockerfile

```
FROM tomcat:latest

LABEL org.opencontainers.image.authors="edowon0623@gmail.com"

COPY ./webapp.war /usr/local/tomcat/webapps
```

Exercise 8# JenKins Job 2/6

njone company

- Ansible Playbook을 이용한 Docker Image 생성 (CI job)

- Item name → **My-K8s-Project-for-CI**
 - Copy from: **My-K8s-Project-using-Ansible**

- Build Triggers
 - Poll SCM: * * * * *

The screenshot shows the Jenkins General configuration page for a job named "My-K8s-Project-for-CI". The "General" tab is selected. In the "설명" (Description) section, the text "Deploy on the K8s master using Ansible playbook (CI)" is displayed. Under the "빌드 유발" (Build Triggers) section, several options are listed: "Build whenever a SNAPSHOT dependency is built" (checked), "Schedule build when some upstream has no successful builds" (unchecked), "Build after other projects are built" (unchecked), "Build periodically" (unchecked), "GitHub hook trigger for GITScm polling" (unchecked), and "Poll SCM" (checked). A red dashed box highlights the "Poll SCM" option.

Exercise 8# JenKins Job 3/6

njone company

- Post-build Actions
 - **SSH Server: ansible-host**
- Exec command
 - ***ansible-playbook -i ./k8s/hosts create-cicd-devops-image.yml --limit ansible-server***



Exercise 8# JenKins Job 4/6

njone company

■ Build Now

The screenshot shows two windows. On the left is the Jenkins 'Build History' page, which lists three builds: #3 (success), #4 (success), and #5 (pending). Build #4 is highlighted with a red dashed box. An arrow points from this box to the right window, which is a Docker Hub search results page for 'edowon0623'. It shows one repository: 'edowon0623 / cicd-project-ansible', last updated 3 minutes ago. This repository is also highlighted with a red dashed box.

■ Modify code

This screenshot shows the Jenkins 'Build History' page again. It now includes build #5, which is marked as 'pending—Finished waiting'. An arrow points from this build to the middle Jenkins 'Build History' window, which displays builds #4 and #5. Build #5 is highlighted with a red dashed box. Another arrow points from this box to the right Docker Hub search results page, which now shows the same repository ('edowon0623 / cicd-project-ansible') but with a different update timestamp: 'Updated a few seconds ago'. This repository is also highlighted with a red dashed box.

Exercise 8# JenKins Job 5/6

njone company

- Configuration
- **Post-build Actions**
 - **Build other projects**
 - **Deploy on Kubernetes CD (CD job) → My-K8s-Project-using-Ansible**
 - **Trigger only if build is stable**

The screenshot shows the Jenkins post-build actions configuration. A red circle highlights the 'Build other projects' section. Within this section, a red dashed box surrounds the 'Projects to build' field, which contains 'My-K8s-Project-using-Ansible'. Below this, a red dashed box surrounds the radio button 'Trigger only if build is stable', which is selected. Other options shown are 'Trigger even if the build is unstable' and 'Trigger even if the build fails'.

Exercise 8# JenKins Job 6/6

njone company

- Build Now (again) or Modify code

The image shows two screenshots. On the left, the Jenkins 'Build History' page displays a successful build (#6) from September 15, 2021, at 12:32. On the right, the Docker Hub search results for 'edowon0623' show a public repository named 'cicd-project-ansible' with the same timestamp.

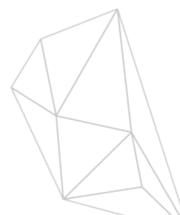
```
[root@k8s-master ~]# kubectl get all
NAME                                         READY   STATUS    RESTARTS   AGE
pod/cicd-deployment-54f8446c9c-f62gj        1/1     Running   0          2m58s
pod/cicd-deployment-54f8446c9c-sj8b2        1/1     Running   0          2m58s

NAME                  TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)
service/cicd-service  NodePort   10.105.71.210 <none>       8080:32000/TCP
```

← → ⌂ ⓘ localhost:32000/hello-world/

It's working on Tomcat server(9.0.65)

Hi, there





후속 강의 소개

- ~~Spring Cloud로 개발하는 마이크로서비스 애플리케이션~~
- Jenkins를 이용한 CI/CD Pipeline 구축
- Microservice Architecture 와 Patterns
- Spring Boot 와 WebFlux를 이용한 Reactive RESTful API 개발