

# 2장. 쿠버네티스 클러스터 생성

Mastering kubernetes

# OpenShift Origin (OKD) 3.10 설치

- 참고 사이트 : [https://www.server-world.info/en/note?os=CentOS\\_7&p=openshift310&f=1](https://www.server-world.info/en/note?os=CentOS_7&p=openshift310&f=1)

## 오픈시프트

위키백과, 우리 모두의 백과사전.

오픈시프트(OpenShift)는 컨테이너 기반 소프트웨어의 디플로이 및 관리를 위한 레드햇의 컴퓨터 소프트웨어 제품이다. 구체적으로 말해, 가속화된 애플리케이션 개발을 위해 도커 컨테이너와 데브옵스 도구를 사용하는 쿠버네티스의 지원 배포판이라 할 수 있다.

### 목차 [숨기기]

- 1 설명
- 2 같이 보기
- 3 각주
- 4 추가문헌
- 5 외부 링크

### 오픈시프트



**OPENSIFT**

개발자	레드햇
발표일	2011년 5월 4일 (7년 전)
최근 버전	3.5 / 2017년 4월

# 설치를 위한 준비( 중요 )

- 노드 구성

[ ctrl.srv.world ]  
(Master Node)  
(Infra Node)  
(Compute Node)

[ node01.srv.world ]  
(Compute Node)

[ node02.srv.world ]  
(Compute Node)

- 클러스터를 구성하기 위한 **필수** 요구사항
  1. Master 노드는 4vCPU, 16GB 메모리 이상
  2. Compute 노드는 1vCPU, 8GB 메모리 이상
  3. 모든 노드는 RHEL(CentOS) 7.4 이상의 운영체제

# 모든 노드에서 준비 작업

# 필요한 계정 생성

```
[root@ctrl ~]# useradd origin
```

```
[root@ctrl ~]# passwd origin
```

```
[root@ctrl ~]# echo -e 'Defaults:origin !requiretty\norigin ALL = (root) NOPASSWD:ALL' | tee  
/etc/sudoers.d/openshift
```

```
[root@ctrl ~]# chmod 440 /etc/sudoers.d/openshift
```

# 방화벽 오픈

# if FirewallD is running, allow SSH

```
[root@ctrl ~]# firewall-cmd --add-service=ssh --permanent
```

```
[root@ctrl ~]# firewall-cmd --reload
```

# 모든 노드에서 준비 작업

[root@ctrl ~]# [yum](#) -y install centos-release-openshift-origin310 epel-release docker git pyOpenSSL

[root@ctrl ~]# [systemctl](#) start docker

[root@ctrl ~]# [systemctl](#) enable docker

```
root@ygj00:~  
Updated:  
  systemd.x86_64 0:219-62.el7_6.5  
  
Dependency Updated:  
  audit.x86_64 0:2.8.4-4.el7  
  audit-libs.x86_64 0:2.8.4-4.el7  
  cryptsetup-libs.x86_64 0:2.0.3-3.el7  
  libgudev1.x86_64 0:219-62.el7_6.5  
  libselinux.x86_64 0:2.5-14.1.el7  
  libselinux-python.x86_64 0:2.5-14.1.el7  
  libselinux-utils.x86_64 0:2.5-14.1.el7  
  libsemanage.x86_64 0:2.5-14.el7  
  libsepol.x86_64 0:2.5-10.el7  
  openssl.x86_64 1:1.0.2k-16.el7  
  openssl-libs.x86_64 1:1.0.2k-16.el7  
  policycoreutils.x86_64 0:2.5-29.el7_6.1  
  selinux-policy.noarch 0:3.13.1-229.el7_6.9  
  selinux-policy-targeted.noarch 0:3.13.1-229.el7_6.9  
  systemd-libs.x86_64 0:219-62.el7_6.5  
  systemd-sysv.x86_64 0:219-62.el7_6.5  
  
Complete!  
[root@ygj00 ~]#
```

# Master 노드에서 작업

# 자동 로그인을 위한 작업

```
[origin@ctrl ~]$ ssh-keygen -q -N ""
```

Enter file in which to save the key (/home/origin/.ssh/id\_rsa):

```
[origin@ctrl ~]$ vi ~/.ssh/config
```

# create new ( define each node )

```
Host ctrl
```

```
    Hostname ctrl.srv.world
```

```
    User origin
```

```
Host node01
```

```
    Hostname node01.srv.world
```

```
    User origin
```

```
Host node02
```

```
    Hostname node02.srv.world
```

```
    User origin
```

```
[origin@ctrl ~]$ chmod 600 ~/.ssh/config
```

# transfer public-key to other nodes

```
[origin@ctrl ~]$ ssh-copy-id node01
```

origin@node01.srv.world's password:

Number of key(s) added: 1 Now try logging into the machine, with: "ssh 'node01'" and check to make sure that only the key(s) you wanted were added.

```
[origin@ctrl ~]$ ssh-copy-id node02
```

```
[origin@ctrl ~]$ ssh-copy-id ctrl
```

# Master 노드에서 작업

# 자동설치를 위한 도구 설치

```
[origin@ctrl ~]$ sudo yum -y install openshift-ansible
```

# 자동설치를 위한 옵션 설정

```
[origin@ctrl ~]$ sudo vi /etc/ansible/hosts
```

# add follows to the end

[OSEv3:children]

masters

nodes

etcd

[OSEv3:vars]

# admin user created in previous section

ansible\_ssh\_user=origin

ansible\_become=true

openshift\_deployment\_type=origin

# use HTTPasswd for authentication

openshift\_master\_identity\_providers=[{'name': 'htpasswd\_auth', 'login': 'true', 'challenge': 'true', 'kind': 'HTPasswdPasswordIdentityProvider'}]

# define default sub-domain for Master node

openshift\_master\_default\_subdomain=apps.srv.world

# allow unencrypted connection within cluster

openshift\_docker\_insecure\_registries=172.30.0.0/16

[masters]

ctrl.srv.world openshift\_schedulable=true containerized=false

[etcd]

ctrl.srv.world

[nodes]

# defined values for [openshift\_node\_group\_name] in the file below

# [/usr/share/ansible/openshift-ansible/roles/openshift\_facts/defaults/main.yml]

ctrl.srv.world openshift\_node\_group\_name='node-config-master-infra'

node01.srv.world openshift\_node\_group\_name='node-config-compute'

node02.srv.world openshift\_node\_group\_name='node-config-compute'

# Master 노드에서 작업

# 오픈슈프트 설치를 위한 준비작업을 실행

# run Prerequisites Playbook

[origin@ctrl ~]\$ ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/prerequisites.yml

```
origin@ctrl-
hifft_repos/templates/CentOS-OpenShift-Origin.repo.j2)

TASK [openshift_repos : Ensure clean repo cache in the event repos have been changed manually] ***
changed: [ctrl.srv.world] => {
  "msg": "First run of openshift_repos"
}
changed: [node01.srv.world] => {
  "msg": "First run of openshift_repos"
}
changed: [node02.srv.world] => {
  "msg": "First run of openshift_repos"
}

TASK [openshift_repos : Record that openshift_repos already ran] *****
ok: [node01.srv.world]
ok: [ctrl.srv.world]
ok: [node02.srv.world]
[WARNING]: flush_handlers task does not support when conditional

RUNNING HANDLER [openshift_repos : refresh cache] *****
changed: [ctrl.srv.world]
changed: [node02.srv.world]
changed: [node01.srv.world]

PLAY [Install packages necessary for installer] *****

TASK [Gathering Facts] *****
ok: [ctrl.srv.world]
ok: [node02.srv.world]
ok: [node01.srv.world]

TASK [Determine if chrony is installed] *****
[WARNING]: Consider using the yum, dnf or zypper module rather than running 'rpm'.
If you need to use command because yum, dnf or zypper is insufficient you can add
'warn: false' to this command task or set 'command_warnings=False' in ansible.cfg to
get rid of this message.

changed: [ctrl.srv.world]
changed: [node01.srv.world]
changed: [node02.srv.world]

TASK [Install ntp package] *****
changed: [ctrl.srv.world]
changed: [node01.srv.world]
```

```
origin@ctrl-
skipping: [node02.srv.world]

TASK [container_runtime : copy "/var/lib/docker" to "/var/lib/containers/docker"] ***
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

TASK [container_runtime : Set the selinux context on /var/lib/containers/docker] ****
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

TASK [container_runtime : restorecon the /var/lib/containers/docker] *****
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

TASK [container_runtime : ensure the unmount of top level mount point] *****
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

TASK [container_runtime : Remove the old docker location] *****
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

TASK [container_runtime : Setup the link] *****
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

TASK [container_runtime : start docker] *****
skipping: [ctrl.srv.world]
skipping: [node01.srv.world]
skipping: [node02.srv.world]

PLAY RECAP *****
ctrl.srv.world      : ok=94   changed=23   unreachable=0   failed=0
localhost          : ok=11   changed=0    unreachable=0   failed=0
node01.srv.world   : ok=69   changed=22   unreachable=0   failed=0
node02.srv.world   : ok=69   changed=22   unreachable=0   failed=0

INSTALLER STATUS *****
Initialization : Complete (-1 day, 22:23:46)
[origin@ctrl ~]$
```



# Master 노드에서 작업

# 오픈쉬프트 설치( 3개의 노드에 알아서 접속해서 자동 설치 ), 30분 정도 소요

```
# run Deploy Cluster Playbook
```

```
[origin@ctrl ~]$ ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/deploy_cluster.yml
```

오른쪽 이미지는 RHEL(CentOS) 7.4 미만의 운영체제에서 설치했때의 오류 메시지

수정방법 : Kernel 업데이트를 해보면 됨.

```

CHECK [memory_availability : node02.srv.world] *****
[fail: [node02.srv.world]: FAILED! -> {"changed": true, "checks": [{"disk_availability": [{"docker_image_availability": {"changed": false, "docker_image": "centos:7"}, {"failures": [{"openshiftcheckception", "Docker storage drivers 'overlay' and 'overlay2' are only supported beginning with kernel version 3.10.0-514; but Docker reports kernel version 3.10.0-327.el7.x86_64."}], "arg": "Docker storage drivers 'overlay' and 'overlay2' are only supported beginning with kernel version 3.10.0-514; but Docker reports kernel version 3.10.0-327.el7.x86_64."}, {"memory_availability": [{"package_availability": {"changed": false, "invocation": {"module_args": {"packages": ["pyyaml", "bind", "ceph-common", "dnsmasq", "docker", "firewalld", "flannel", "glusterfs-fuse", "iptables", "iptables-services", "iscsi-initiator-utils", "linuxflex-python", "nfs-utils", "ntp", "openssh", "origin", "origin-node", "reportd", "python-http-lib2", "yum-utils"]}}}, {"package_version": {"changed": false, "invocation": {"module_args": {"package_list": [{"check_multi": false, "name": "origin", "version": ""}, {"check_multi": false, "name": "origin-master", "version": ""}, {"check_multi": false, "name": "origin-node", "version": ""}], "package_mgr": "yum"}]}}, {"arg": "One or more checks failed"}, {"playbook_context": "install"}]}]}]

```

# Master 노드에서 작업

# 상태확인

[origin@ctrl ~]\$ oc get nodes

# show state with labels

[origin@ctrl ~]\$ oc get nodes --show-labels=true

```
origin@ctrl:~$ oc get nodes --show-labels=true
TASK [openshift_cluster_autoscaler : create the policies] *****
skipping: [ctrl.srv.world] => (item={u'name': u'cluster-role'})
skipping: [ctrl.srv.world] => (item={u'name': u'role', u'role_namespace': u'openshift-cluster-autoscaler'})

TASK [openshift_cluster_autoscaler : grant cluster-reader to cluster-autoscaler] ****
skipping: [ctrl.srv.world]

TASK [openshift_cluster_autoscaler : laydown the deployment file for cluster-autoscaler] ***
skipping: [ctrl.srv.world]

TASK [openshift_cluster_autoscaler : Ensure the cluster-autoscaler is present] *****
skipping: [ctrl.srv.world]

PLAY [Cluster Auto Scaler Install Checkpoint End] *****

TASK [Set Cluster Auto Scaler install 'Complete'] *****
skipping: [ctrl.srv.world]

PLAY RECAP *****
ctrl.srv.world      : ok=704  changed=301  unreachable=0    failed=0
localhost          : ok=11   changed=0    unreachable=0    failed=0
node01.srv.world   : ok=112  changed=42   unreachable=0    failed=0
node02.srv.world   : ok=112  changed=42   unreachable=0    failed=0

INSTALLER STATUS *****
Initialization      : Complete (0:00:52)
Health Check        : Complete (0:01:50)
Node Bootstrap Preparation : Complete (0:03:30)
etcd Install        : Complete (0:01:14)
Master Install      : Complete (0:11:12)
Master Additional Install : Complete (0:00:37)
Node Join           : Complete (0:00:31)
Hosted Install      : Complete (0:00:58)
Cluster Monitoring Operator : Complete (0:03:19)
Web Console Install : Complete (0:01:28)
Console Install     : Complete (0:01:01)
metrics-server Install : Complete (0:00:01)
Service Catalog Install : Complete (0:02:13)
[origin@ctrl ~]$
```

# Master 노드에서 작업

# cent 라는 사용자 추가

```
[origin@ctrl ~]$ sudo htpasswd /etc/origin/master/htpasswd cent
```

New password:    # set password

Re-type new password:

Adding password for user cent

# 접속 허용

```
[cent@ctrl ~]$ oc login
```

Server [https://localhost:8443]: https://ctrl.srv.world:8443

The server uses a certificate signed by an unknown authority.

You can bypass the certificate check, but any data you send to the server could be intercepted by others.

Use insecure connections? (y/n): y

Authentication required for https://ctrl.srv.world:8443 (openshift)

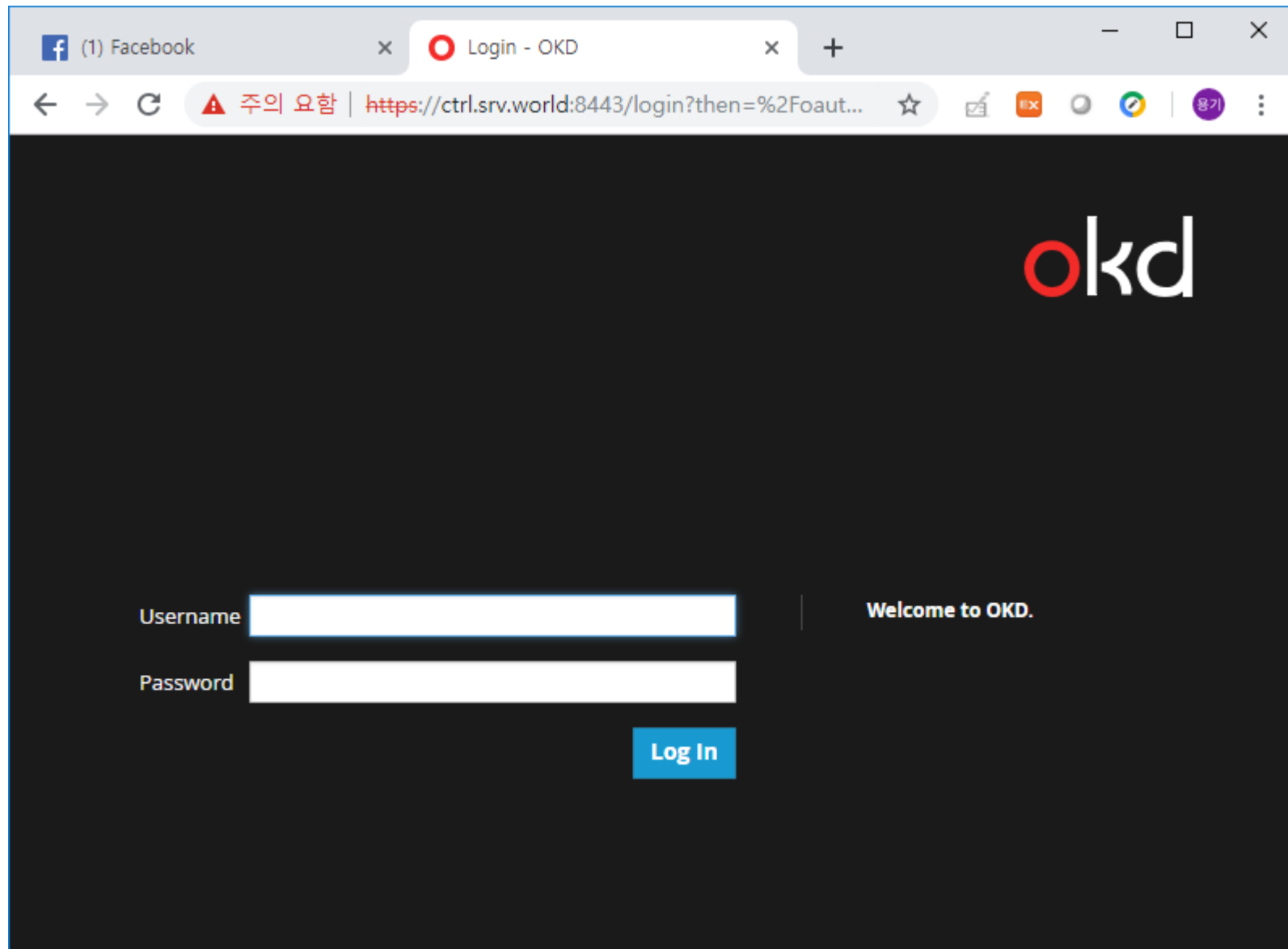
Username: cent

Password:

Login successful.

You don't have any projects. You can try to create a new project, by running

# 관리 웹화면



(1) Facebook

Login - OKD

주의 요함 | <https://ctrl.srv.world:8443/login?then=%2Foaut...>

okd

Username

Password

Log In

Welcome to OKD.

# 관리웹에 로그인된 화면

