OpenShift 4.3 Disconnected Install

1. 준비환경

1) Repository 노드

OS : RHEL 7.6
dhcp - dhcp 를 이용한 네트워크 부팅
tftp-server - iso 등 설치파일 마운트
httpd - 설치파일, ignition, pxeboot
yum repo upload
ntp
bind(named) - DNS 서버

2) Registry 서버
OS : RHEL 7.6
haproxy - 로드 밸런싱
mirror-registry
OCP image upload
/opt/registry/data/docker/registry/v2/reposi tor ies/ocp4/openshi f t4

3) bootstrap
OS : RHCOS
openshift cluster operator 를 통한 클러스터 설치
최소사양 : 4vCPU / 16GB

4) master 3 대
OS : RHCOS
최소사양 : 4vCPU / 16GB
5) worker 2 대
OS : RHCOS or RHEL 7.6
최소사양 : 2vCPU / 8GB

2. Repository 서버 구성

1) Yum repo 구성
vi /etc/yum.repos.d/ocp4.repo

```
[rhel-7-server-rpms]
name=rhel-7-server-rpms
baseurl=file:///var/www/html/repos/rhel-7-server-rpms
enabled=1
gpgcheck=0
[rhel-7-server-extras-rpms]
name=rhel-7-server-extras-rpms
baseurl=file:///var/www/html/repos/rhel-7-server-extras-rpms
enabled=1
gpgcheck=0
[rhel-7-server-ansible-2.8-rpms]
name=rhel-7-server-ansible-2.8-rpms
baseurl=file:///var/www/html/repos/rhel-7-server-ansible-2.8-rpms
```

```
enabled=1
gpgcheck=0
[rhel-7-server-ose-4.2-rpms]
name=rhel-7-server-ose-4.2-rpms
baseurl=file:///var/www/html/repos/repo/rhel-7-server-ose-4.3-rpms
enabled=1
gpgcheck=0

----------------------------

[rhel-7-server-rpms]
name=rhel-7-server-rpms
baseurl=http://192.168.50.200:8080/repo/rhel-7-server-rpms
enabled=1
gpgcheck=0
[rhel-7-server-extras-rpms]
name=rhel-7-server-extras-rpms
baseurl=http://192.168.50.200:8080/repo/rhel-7-server-extras-rpms
enabled=1
gpgcheck=0
[rhel-7-server-ansible-2.8-rpms]
name=rhel-7-server-ansible-2.8-rpms
baseurl=http://192.168.50.200:8080/repo/rhel-7-server-ansible-2.8-rpms
enabled=1
gpgcheck=0
[rhel-7-server-ose-4.2-rpms]
name=rhel-7-server-ose-4.2-rpms
baseurl=http://192.168.50.200:8080/repo/rhel-7-server-ose-4.3-rpms
enabled=1
gpgcheck=0
```

2) Install Package

```
repo server isntal|
yum -y install syslinux tftp-server vsftpd dhcp xinetd haproxy httpd-tools httpd
podman
bind-utils bind

repo server isntall, reg server install
yum -y install vim wget git net-tools yum-utils iptables-services bridge-utils
bashcompletion kexec-tools sos psacct jq

reg server install
yum -y install syslinux haproxy httpd-tools podman bind-utils bind
```

- selinux 설정

```
vi /etc/selinux/config
SELINUX=permissive
setenforce 0

설정 완료 후 실행
systemctl enable httpd
systemctl start httpd dhcpd xinetd


systemct| start haproxy
restorecon -vR /var/www/html
chmod -R 755 /var/www/html
```

3) pxeboot 구성

```
경로 :
/var/lib/tftpboot/pxelinux.cfg/default
/var/ib/tftpboot/ocp43/

mkdir -p /var/lib/tftpboot/pxelinux.cfg
mkdir -p /var/lib/tftpboot/ocp43

cp -R /usr/share/syslinux/* /var/lib/tftpboot/
```

```
vi /var/lib/tftpboot/pxelinux.cfg/default

default menu.c32
prompt 0
timeout 1000
menu title ###HHHHH PXE Boot Menu ###HHHHH

label 1
  menu label ^1 - Boot from bootstrap.ign
    KERNEL http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.50.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz
coreos.inst.ignition_url=http://192.168.50.200:8080/ign/bootstrap.ign

label 2
    menu label ^2 - Boot from master.ign
    KERNEL http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.50.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz coreos.inst.ignition_url=http://192.168.50.200:8080/ign/master.ign

label 3
    menu label ^3 - Boot from worker.ign
    KERNEL http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
```

```
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.50.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz coreos.inst.ignition_url=http://192.168.50.200:8080/ign/worker.ign
```

```
worker 가 다른 IP 대역대에 존재 할 경우 대여개 별로 label 생성
repository 노드에 대역대 별 IP 지정

default menu.c32
prompt 0
timeout 1000
menu title ###HHHHH PXE Boot Menu ###HHHHH

label 1
  menu label ^1 - Boot from bootstrap.ign
    KERNEL http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.50.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz
coreos.inst.ignition_url=http://192.168.50.200:8080/ign/bootstrap.ign

label 2
    menu label ^2 - Boot from master.ign
    KERNEL http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.50.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz coreos.inst.ignition_url=http://192.168.50.200:8080/ign/master.ign

label 3
    menu label ^3 - Boot from worker.ign
    KERNEL http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.50.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.50.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz coreos.inst.ignition_url=http://192.168.50.200:8080/ign/worker.ign

label 4
    menu label ^4 - Boot from worker.ign
    KERNEL http://192.168.10.200:8080/ocp43/rhcos-4.3.0-x86_64-installer-kernel
    APPEND ip=dhcp rd.neednet=1 initrd=http://192.168.10.200:8080/ocp43/rhcos-
4.3.0-x86_64-installer-initramfs.img console=tty0 console=ttyS0 coreos.inst=yes
coreos.inst.install_dev=vda
coreos.inst.image_url=http://192.168.10.200:8080/ocp43/rhcos-4.3.0-x86_64-
metal.raw.gz coreos.inst.ignition_url=http://192.168.10.200:8080/ign/worker.ign

chmod -R 755 /var/lib/tftpboot/pxelinx.cfg
```

```
cd /root/paas_work/ instal |_file/kb-rhocp4/
```

```
cp rhcos-4.3.0-x86_64-installer-initramfs.img /var/lib/tftpboot/ocp43/
cp rhcos-4.3.0-x86_64-installer-kernel /var/lib/tftpboot/ocp43/
cp rhcos-4.3.0-x86_64-installer.iso /var/lib/tftpboot/ocp43/
cp rhcos-4.3.0-x86_64-metal.raw.gz /var/lib/tftpboot/ocp43/

chmod -R 755 /var/lib/tftpboot/ocp43

ls -arlt /var/lib/tftpboot/ocp43/
rhcos-4.3.0-x86_64-installer-initramfs.img
rhcos-4.3.0-x86_64-installer-kernel
rhcos-4.3.0-x86_64-installer.iso
rhcos-4.3.0-x86_64-metal.raw.gz
```

4) xinetd 구성

경로 : /etc/xinetd.d/tftp

```
service tftp
{
    socket_type     = dgram
    protocol        = udp
    wait            = yes
    user            = root
    server          = /usr/sbin/in.tftpd
    server_args     = -s /var/lib/tftpboot
    disable         = no
    per_source      = 11
    cps         = 100 2
    flags           = IPv4
}
```

5) haproxy (로드밸런서) ### registry server / 신규 구성 시 Server IP 확인

경로 : /etc/haproxy/haproxy.cfg

```
#---------------------------------------------------------------------
# Example configuration for a possible web application.  See the
# full configuration options online.
#
#   http://haproxy.1wt.eu/download/1.4/doc/configuration.txt
#
#---------------------------------------------------------------------

#---------------------------------------------------------------------
# Global settings
#---------------------------------------------------------------------
global
    # to have these messages end up in /var/log/haproxy.log you will
    # need to:
    #
    # 1) configure syslog to accept network log events.  This is done
    #    by adding the '-r' option to the SYSLOGD_OPTIONS in
    #    /etc/sysconfig/syslog
```

```
    #
    # 2) configure local2 events to go to the /var/log/haproxy.log
    #   file. A line like the following can be added to
    #   /etc/sysconfig/syslog
    #
    #    local2.*                       /var/log/haproxy.log
    #
    log         127.0.0.1 local2

    chroot      /var/lib/haproxy
    pidfile     /var/run/haproxy.pid
    maxconn     4000
    user        haproxy
    group       haproxy
    daemon

    # turn on stats unix socket
    stats socket /var/lib/haproxy/stats

#---------------------------------------------------------------------
# common defaults that all the 'listen' and 'backend' sections will
# use if not designated in their block
#---------------------------------------------------------------------
defaults
    mode                    http
    log                     global
    option                  httplog
    option                  dontlognull
    option http-server-close
    option forwardfor       except 127.0.0.0/8
    option                  redispatch
    retries                 3
    timeout http-request    10s
    timeout queue           1m
    timeout connect         10s
    timeout client          1m
    timeout server          1m
    timeout http-keep-alive 10s
    timeout check           10s
    maxconn                 3000

#---------------------------------------------------------------------
# main frontend which proxys to the backends
#---------------------------------------------------------------------
frontend openshift-api-server
    bind *:6443
    default_backend openshift-api-server
    mode tcp
    option tcplog

backend openshift-api-server
    balance source
    mode tcp
    server bootstrap 192.168.50.100:6443 check
    server master01 192.168.50.101:6443 check
    server master02 192.168.50.102:6443 check
    server master03 192.168.50.103:6443 check
```

```
frontend machine-config-server
    bind *:22623
    default_backend machine-config-server
    mode tcp
    option tcplog

backend machine-config-server
    balance source
    mode tcp
    server bootstrap 192.168.50.100:22623 check
    server master01 192.168.50.101:22623 check
    server master02 192.168.50.102:22623 check
    server master03 192.168.50.103:22623 check

frontend ingress-http
    bind *:80
    default_backend ingress-http
    mode tcp
    option tcplog

backend ingress-http
    balance source
    mode tcp
    server worker01 192.168.50.104:80 check
    server worker02 192.168.50.105:80 check
    server worker3 192.168.50.180:80 check

frontend ingress-https
    bind *:443
    default_backend ingress-https
    mode tcp
    option tcplog

backend ingress-https
    balance source
    mode tcp
    server worker01 192.168.50.104:443 check
    server worker02 192.168.50.105:443 check
    server worker3 192.168.50.180:443 check
```

6) bind 구성 ### DNS Server

경로 :

/etc/named.conf

/var/named/demo.ocp42.com.zone

```
named.conf

//
// named.conf
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
```

```
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//
// See the BIND Administrator's Reference Manual (ARM) for details about the
// configuration located in /usr/share/doc/bind-{version}/Bv9ARM.html

options {
    listen-on port 53 { any; };
    listen-on-v6 port 53 { none; };
    directory   "/var/named";
    dump-file   "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    recursing-file  "/var/named/data/named.recursing";
    secroots-file   "/var/named/data/named.secroots";
    allow-query     { any; };

    /*
     - If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.
     - If you are building a RECURSIVE (caching) DNS server, you need to enable
       recursion.
     - If your recursive DNS server has a public IP address, you MUST enable
access
       control to limit queries to your legitimate users. Failing to do so will
       cause your server to become part of large scale DNS amplification
       attacks. Implementing BCP38 within your network would greatly
       reduce such attack surface
    */
    recursion yes;

    dnssec-enable yes;
    dnssec-validation yes;

    /* Path to ISC DLV key */
    bindkeys-file "/etc/named.root.key";

    managed-keys-directory "/var/named/dynamic";

    pid-file "/run/named/named.pid";
    session-keyfile "/run/named/session.key";
};

logging {
        channel default_debug {
                file "data/named.run";
                severity dynamic;
        };
};

zone "." IN {
    type hint;
    file "named.ca";
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";

zone "demo.ocp42.com" IN {
        type master;
```

```
        file "demo.ocp42.com.zone";
        allow-update { none; } ;
};
```

```
demo.ocp42.com.zone

$TTL 86400
@   IN  SOA @   root. (
                                        2019071000  ; serial
                                        21600       ; refresh
                                        3600        ; retry
                                        604800      ; expire
                                        86400 )     ; minimum TTL
            IN    NS   dns.demo.ocp42.com.

; base infra
bastion      IN    A    192.168.50.200
registry     IN    A    192.168.50.200
dns          IN    A    192.168.50.200
router       IN    A    192.168.50.200
gitlab       IN    A    192.168.50.200

prg          IN    A    192.168.50.106

; routes
*.apps       IN    A    192.168.50.104

; Kubernetes API
api-int      IN    A    192.168.50.200
api          IN    A    192.168.50.200

; etcd
etcd-0       IN    A    192.168.50.101
etcd-1       IN    A    192.168.50.102
etcd-2       IN    A    192.168.50.103

; bootstrap
bootstrap    IN    A    192.168.50.100

; master
master01     IN    A    192.168.50.101
master02     IN    A    192.168.50.102
master03     IN    A    192.168.50.103

; worker
worker01     IN    A    192.168.50.104
worker02     IN    A    192.168.50.105

; SRV DNS records for etcd
_etcd-server-ssl._tcp.demo.ocp42.com 86400 IN    SRV 0        10     2380 etcd-
0.demo.ocp42.com
_etcd-server-ssl._tcp.demo.ocp42.com  86400 IN    SRV 0        10     2380 etcd-
1.demo.ocp42.com
_etcd-server-ssl._tcp.demo.ocp42.com  86400 IN    SRV 0        10     2380 etcd-
2.demo.ocp42.com
```

7) httpd 구성

경로

/etc/httpd

/var/www/html/ignition

/var/www/html/ocp43/

```
mkdir -p /var/www/html/ocp43
cd /root/paas_work/ instal |_file/ko-rhocp4
cp rhcos-4.3.0-x86_64-installer-initramfs.img /var/www/html/ocp43
cp rhcos-4.3.0-x86_64-installer-kernel /var/www/html/ocp43
cp rhcos-4.3.0-x86_64-installer.iso /var/www/html/ocp43
cp rhcos-4.3.0-x86_64-metal.raw.gz /var/www/html/ocp43
```

8) DHCP 구성

vi /etc/dhcp/dhcpd.conf

```
authoritative;
ddns-update-style interim;
default-lease-time 14400:
max-lease-time 14400;

    option routers                     10.37.68.1;
    option broadcast-address           10.37.68.255;
    option subnet-mask                 255.255.255.0;
    option domain-name-servers         10.37.68.13;
    option domain-name                 "intpg.kbstar.local";
    option domain-scarch               "intpg.kbstar.local";

    subnet 10.37.68.0 netmask 255.255.255.0 {
    pool {
        range 10.37.68.10 10.37.68.38;

        host nclbt301 { hardware ethernet 00:50:56:9c:ce:38; fixed-address
10.37.68.37; option host-name "nclbt301.intpg.kbstar.local"; }
        host nclmt301 { hardware ethernet 00:50:56:9c:ce:39; fixed-address
10.37.68.15; option host-name "nclmt301.intpg.kbstar.local"; }
        host nclmt302 { hardware ethernet 00:50:56:9c:ce:40; fixed-address
10.37.68.16; option host-name "nclmt302.intpg.kbstar.local"; }
        host nclmt303 { hardware ethernet 00:50:56:9c:ce:41; fixed-address
10.37.68.17; option host-name "nclmt303.intpg.kbstar.local"; }
        host nclps301 { hardware ethernet 00:50:56:9c:ce:42; fixed-address
10.37.68.18; option host-name "nclps301.intpg.kbstar.local"; }
        host nclps302 { hardware ethernet 00:50:56:9c:ce:43; fixed-address
10.37.68.19; option host-name "nclps302.intpg.kbstar.local"; }

        # this will not give out
        deny unknown-clients;

        # this is PXE specific
        filename "pxelinux.0";
```

```
        #PXE Boot Server IP
        next-server 10.37.68. 12:
        }
}
```

9) mirror-registry 구성

```
Tip. 인증서 생성 시 서버간 시간이 다를 경우 인증오류 발생 / repo, reg 디렉토리 모두 생성
/ reg에서 crt 생성 / 인증서는 10년으로 생성

mkdir -p /paas/opt/registry/{auth,certs,data}
cd /paas/opt/registry/certs
openssl req -newkey rsa:4096 -nodes -sha256 -keyout utilityvm.example.com.key -
x509 -days 3650 -out utilityvm.example.com.crt

Generating a 4096 bit RSA private key
..............++
.........................++
writing new private key to 'utilityvm.example.com.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:KR
State or Province Name (full name) []:Seoul
Locality Name (eg, city) [Default City]:Seoul
Organization Name (eg, company) [Default Company Ltd]:RedHat
Organizational Unit Name (eg, section) []:RedHat
Common Name (eg, your name or your server's hostname) []:utilityvm.example.com
Email Address []:admin@redhat.com

cp /paas/opt/registry/certs/utilityvm.example.com.crt /etc/pki/ca-trust/
source/anchors/
update-ca-trust
htpasswd -bBc /paas/opt/registry/auth/htpasswd paasadm paasadm
```

```
registry 이미지 등록
- 이미지 저장
podman save --quiet -o registry~image.tar docker .io/library/registry:2

registry 노드에 업로드 후 podman load 실행
podman load -i registry-image.tar

mirror-registry 삭제 명령어
podman rm mirror-registry
```

```
vi /etc/containers/registries.conf

# This is a system-wide configuration file used to
# keep track of registries for various container backends.
```

```
# It adheres to TOML format and does not support recursive
# lists of registries.

# The default location for this configuration file is
/etc/containers/registries.conf.

# The only valid categories are: 'registries.search', 'registries.insecure',
# and 'registries.block'.

[registries.search]
registries = ['utilityvm.example.com:5000']

# If you need to access insecure registries, add the registry's fully-qualified
name.
# An insecure registry is one that does not have a valid SSL certificate or only
does HTTP.
[registries.insecure]
registries = ['utilityvm.example.com:5000']


# If you need to block pull access from a registry, uncomment the section below
# and add the registries fully-qualified name.
#
# Docker only
[registries.block]
registries = []
```

```
/etc/cni/net.d/87-podman-bridge.conflist     ### subnet 변경 (192.168.0.0/16)

{
    "cniVersion": "0.3.0",
    "name": "podman",
    "plugins": [
      {
        "type": "bridge",
        "bridge": "cni0",
        "isGateway": true,
        "ipMasq": true,
        "ipam": {
            "type": "host-local",
            "subnet": "192.168.0.0/16",
            "routes": [
                { "dst": "0.0.0.0/0" }
            ]
        }
      },
      {
        "type": "portmap",
        "capabilities": {
          "portMappings": true
        }
      }
    ]
}
```

```
- selinux 사용 시
```

```
podman run -d --name mirror-registry -p 5000:5000 --restart=always W
    -v /paas/opt/registry/data:/var/lib/registry:z W
    -v /paas/opt/registry/auth: /auth:z W
    -e "REGISTRY_AUTH=htpasswd" W
    -e "REGISTRY_AUTH_HTPASSWD_REALM=Registry Realm" W
    -e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd W
    -e /paas/opt/registry/certs:/certs:z W
    -e REGISTRY_HTTP_TLS_CERTIFICATE=/certs/registry.dmzsd.kbstar.local.crt W
    -e REGGISTRY_HTTP_TLS_KEY=/cer ts/registry.dmzsd.kbstar.local .key W
    docker.io/library/registry:2

- selinux 미사용 시
podman run -d --name mirror-registry -p 5000:5000 --restart=always \
    -v /paas/opt/registry/data:/var/lib/registry \
    -v /paas/opt/registry/auth:/auth \
    -e "REGISTRY_AUTH=htpasswd" \
    -e "REGISTRY_AUTH_HTPASSWO_REALM=Regislry Realm" \
    -e "REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd" \
    -v /paas/opt/registry/certs:/certs \
    -e REGISTRY_HTTP_TLS_CERTIFICATE=/certs/registry.dmzsd.kbstar.local.crt \
    -e REGISTRY_HTTP_TLS_KEY=/certs/registry.dmzsd.kbstar.local.key \
    docker.io/library/registry:2
```

```
- 레지스트리 확인
curl -u openshift:redhat -k https://utilityvm.example.com:5000/v2/_catalog

echo -n 'passadm:paasadm' | base64 -w0
cGFzc2FkbTpwYWFzYWRt

cat /paas/opt/registry/pull-secret-config.json
{
        "auths": {
                "utilityvm.example.com:5000": {
                        "auth": "b3BlbnNoaWZ0OnJlZGhhdA=="
                }
        }
}
```

```
cat pull-secret-config.json

{
  "auths": {
    "cloud.openshift.com": {
      "auth":
"b3BlbnNoaWZ0LXJlbGVhc2UtZGV2K3N1bmdraW1yZWRoYXRjb20xc2o3Z25uem9zaHRmcmRmY2gwaHh
2d3Ria2o6NUg3WU5FVFU4MzJBMUdLSFNKWExVRzA0V0EwODRJQ1pJVlZKQ1dTMjk3MOcyTzlWQlhVVkd
MQU1HSkVFMzU2Ug==",
      "email": "sungkim@redhat.com"
    },
    "quay.io": {
      "auth":
"b3BlbnNoaWZ0LXJlbGVhc2UtZGV2K3N1bmdraW1yZWRoYXRjb20xc2o3Z25uem9zaHRmcmRmY2gwaHh
2d3Ria2o6NUg3WU5FVFU4MzJBMUdLSFNKWExVRzA0V0EwODRJQ1pJVlZKQ1dTMjk3MOcyTzlWQlhVVkd
MQU1HSkVFMzU2Ug==",
      "email": "sungkim@redhat.com"
    },
```

```
    "registry.connect.redhat.com": {
      "auth":
"NTI4NTEzMDZ8dWhjLTFTSjdnTk5ab3NoVGZSZEZjSDBoeFZXdEJLSjpleUpoYkdjaU9pSlNVelV4TWl
KOS5leUp6ZFdJaU9pSm1ZbUkxTkRWaFpqRTNZekUwWVRkaE9HVTVOVFZqWldFMFptWTNZVGt3TlNKOS5
QQXVzamdfRllEZ194WDNndXJJbkE1bnNHNUU1U1NGQ25JckR3bVo2Y3hPOVk2TllWYnRuV0dXZlA2Y3U
1OHMxbUJKUUJyY2NyeVhuei1NdW5ST1dOZ2hsM3U1MVpmUkdnNTRYdGxiRkczX2VQYOU3VVBlSU1UWUF
GcXFIZHdoSOpEczRYUUt6LXc4c21USnBoaC1rYnhsX21GdjBmXzlnOEJYTVc5R25jVTZ1VFZuTVlaVnl
wT0VZbkFDa2pMTF8OSUhJNFdPWlvabGlOT0hDUzJ6SmV6Z3RkUGpZVTFXNE1wVk9Demx3YlhfdWw0dmN
ORlpQMnRWaUs4MFVUSXNXX3l6Vkh6MDdNQ2ZWTlQ2Nkh2YmJRN3QxV2dMQno5TzFyZOtyM3VESnE1dkM
1UmtUNU1aQ1M5YUFNaGx5ZEFWRS1ESTFHcDFTYm5RQ19XYU42MnlILWVhZEdJQzhyRHBHRGNnendtZHp
YdVluRlA4ZlQySTJPTlpTdEhCMi1ZNWk1R29DUVIwRGl6NOsydlNqUkYwUjlkYmtXdzBkTO52VGdiQ3g
4cWxhZUpacUFlalpiQTl6LW91cXpDSWx6UWY1WDlSblRSNXBwWG1vSHAtNUhJak85emVmUV9iYWF6SUh
WS3FBMVdIVOxQUlhobll6MVFsbnRfdU16cjl2Q1JWNU1TMlhhaFFUODdBSlpZT2NpVzFoUC1jdTRNTHQ
3andpbVdrV2hTMGJCckNqVDlha1lfaFI4TU9DR1cOdVBYaVVGZFFscGhPamY4b2xjdHFzZnRMdGZrSFN
TWTZlck8wSDhQcFV2MXdxZW1jUE93bE96WnhwbWxEVUtocDdlUVlIRGZYVHphYkNxaUhmSzFFbUx1dVp
hMHlaXzdEc29qROUySOFhVXIwLWlzZw==",
      "email": "sungkim@redhat.com"
    },
    "registry.redhat.io": {
      "auth":
"NTI4NTEzMDZ8dWhjLTFTSjdnTk5ab3NoVGZSZEZjSDBoeFZXdEJLSjpleUpoYkdjaU9pSlNVelV4TWl
KOS5leUp6ZFdJaU9pSm1ZbUkxTkRWaFpqRTNZekUwWVRkaE9HVTVOVFZqWldFMFptWTNZVGt3TlNKOS5
QQXVzamdfRllEZ194WDNndXJJbkE1bnNHNUU1U1NGQ25JckR3bVo2Y3hPOVk2TllWYnRuV0dXZlA2Y3U
1OHMxbUJKUUJyY2NyeVhuei1NdW5ST1dOZ2hsM3U1MVpmUkdnNTRYdGxiRkczX2VQYOU3VVBlSU1UWUF
GcXFIZHdoSOpEczRYUUt6LXc4c21USnBoaC1rYnhsX21GdjBmXzlnOEJYTVc5R25jVTZ1VFZuTVlaVnl
wT0VZbkFDa2pMTF8OSUhJNFdPWlvabGlOT0hDUzJ6SmV6Z3RkUGpZVTFXNE1wVk9Demx3YlhfdWw0dmN
ORlpQMnRWaUs4MFVUSXNXX3l6Vkh6MDdNQ2ZWTlQ2Nkh2YmJRN3QxV2dMQno5TzFyZOtyM3VESnE1dkM
1UmtUNU1aQ1M5YUFNaGx5ZEFWRS1ESTFHcDFTYm5RQ19XYU42MnlILWVhZEdJQzhyRHBHRGNnendtZHp
YdVluRlA4ZlQySTJPTlpTdEhCMi1ZNWk1R29DUVIwRGl6NOsydlNqUkYwUjlkYmtXdzBkTO52VGdiQ3g
4cWxhZUpacUFlalpiQTl6LW91cXpDSWx6UWY1WDlSblRSNXBwWG1vSHAtNUhJak85emVmUV9iYWF6SUh
WS3FBMVdIVOxQUlhobll6MVFsbnRfdU16cjl2Q1JWNU1TMlhhaFFUODdBSlpZT2NpVzFoUC1jdTRNTHQ
3andpbVdrV2hTMGJCckNqVDlha1lfaFI4TU9DR1cOdVBYaVVGZFFscGhPamY4b2xjdHFzZnRMdGZrSFN
TWTZlck8wSDhQcFV2MXdxZW1jUE93bE96WnhwbWxEVUtocDdlUVlIRGZYVHphYkNxaUhmSzFFbUx1dVp
hMHlaXzdEc29qROUySOFhVXIwLWlzZw==",
      "email": "sungkim@redhat.com"
    },
    "utilityvm.example.com:5000": {
      "auth": "b3BlbnNoaWZ0OnJlZGhhdA=="
    }
  }
}
```

```
vi install-conig.yaml

apiVersion: v1
baseDomain: kbstar.local
compute:
- hyperthreading: Enabled
  name: worker
  replicas: 2
controlPlane:
  hyperthreading: Enabled
  name: master
  replicas: 3
metadata:
  name: dmzsd
networking:
```

```
  clusterNetwork:
  - cidr: 172.31.0.0/16
    hostPrefix: 24
  networkType: OpenShiftSDN
  serviceNetwork:
  - 172.30.0.0/16
platform:
  none: {}
pullSecret: '{"auths":{"nclrpa01.dmzsd.kbstar.local:5000":
{"auth":"cGFhc2FkbTpwYWFzZYWAt", "email": "noemail@localhost"}}}'
sshKey: |
  'ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDKu2Vc7qRrv51CtGoCPtPd0JTOxEjdWSb4LyprsQfFYcDxElVM
KWdEFaprvRapvCfRtpKbkp5LqGiE7EdU/9mruToFVGxA+A9bT3dka5JmVOQiwJG9VVtcpoPpoNS7NLDe
LZtH4SMWqw3nPLwfqra1z5sPAv5cImj55JuDT8PC8Ywpl1+XGJAr2GnTB23/ErYXhhzgYh7R8E5p1FrS
TC0ACO5moHy+L1crEocLRDVVVQu+DceIkt2lMmZmexwvdFQ2f5BFB2/fl0HlZHIPJ7oCFxzV5+BwWpVD
KrxHE51o9mpgU2epBTMNRaxulOfaUUnIt+mAOdwhR0q6817kwYHh
root@nclrpa01.dmzsd.kbstar.local '
additionalTrustBundle: |
    -----BEGIN CERTIFICATE-----
    MIIFvTCCA6WgAwIBAgIJALbZi8oxd+Y5MA0GCSqGSIb3DQEBCwUAMHUxCzAJBgNV
    BAYTAlVTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0dGxlMRAw
    DgYDVQQKDAdSZWQgSGF0MQ0wCwYDVQQLDARHUFRFMR4wHAYDVQQDDBV1dGlsaXR5
    dm0uZXhhbXBsZS5jb20wHhcNMjAwMzE4MDEyMjQ1WhcNMjEwMzE4MDEyMjQ1WjB1
    MQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGluZ3RvbjEQMA4GA1UEBwwHU2Vh
    dHRsZTEQMA4GA1UECgwHUmVkIEhhdDENMAsGA1UECwwER1BURTEeMBwGA1UEAwwV
    dXRpbGl0eXZtLmV4YW1wbGUuY29tMIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIIC
    CgKCAgEApa8rHlHMdKu47gXVap8E1jZAFDZeaxxsGXFEWYMip6bxhrvGy3KzNBp5
    iwBw5b5exRqzKt3gTwkQsTgX4L3OBg7BDf+SN1YKNDwl65YLpCJjEzRIwZs8aP+Z
    3wto3PO0+BJuH7JXFBtPF7v80Cayzk/qwnE4zVjIlG6BLQ6H6TofY43TccNUPm9Y
    wbLF5f3+klyRGqKmWsJa31AKdRnjRqQeIKActXcqDx66amOSCs7+jtxuEq8zWhA8
    2+zQYGaShDHzbot8C7jsKv9UInvD9WWv+W1A6QfM3zgeA/574n6U3ByFj2GkpRUn
    8FXj1wIHEFIGqgileMTW5r1L7kTrZ2O7ulIOgQDO54DLbIXzlySaCIaINmUZ663E
    KmuapJTeq5IcW8s9ADm29RtihMtG9LbrBLYlVAVxYKFKdbS2VYvET2VQ8HVOLGc1
    Hh0du079kfcUsOZAVktDPM/5F8ZLKj6gjel+fhvny4mihC3wPuPoVtGsoSSoGGIc
    mRvqUI/eh3E4pqTMTqj2+1DItU4oM+jMW0VA38fRhEdh/GBpB77BcDawiX6Ru4hh
    6BtDwkx9OEXNa0lS5lV2zEAxcBPSnScDh0k0DVhRK+fO5jdFO8S2sn5qkcOU+3pp
    PvBlSeg4CANNAJEoOQqNo2IB73l1FehirGJGcFMx+uqS9q0lDKECAwEAAaNQME4w
    HQYDVR0OBBYEFHJOl7bWhwo+6oNOm1lWAf+Xp/MaMB8GA1UdIwQYMBaAFHJOl7bW
    hwo+6oNOm1lWAf+Xp/MaMAwGA1UdEwQFMAMBAf8wDQYJKoZIhvcNAQELBQADggIB
    AIAassPXUQLworstfQaMIGro4//iKe4pO2gNQePlyKWbBzjAJ8DTrb50D2N6diTA
    HpUDrQcwhYd/HtFMpcUTimFtxfOHRKqXO8lyoVduol+5dcmw5YdKCFP8o8ON46bU
    JHQiyqcUYhbvv6YWHWYRgPMX9GPXpXbt/EJ6xJ+Mnp6o7oAW5CeOc7lFwsiJ6Fnn
    j8HlMMw1giuy7BN/4mccwfrwQArXE0atURZbKFgPyLVPJRQUizOxHPSMhPuU6sfV
    PU2wbQwTpQr2R6em1v9Co4fsOzfQQ3rtIhyuTSqnkmqMsnEs43XfggK6NvcSOuKW
    TKRCZ3MXDELFyBKHnPYUh6A+ThwwLucJgSG3aSwcbp2eaeQDzqjzTlbZBJb3MyM1
    v7hoVpdb9qslyhwT2yT0bObRHqjJk8eIu3jykpjuf+1y+4Hs6egBHbE2F1d7eUA3
    mAmZ4QRdf8BLaiBkL1z5yCQe7Clh7jSqcHLJWu8gRpfqoSVx5ehokoiAgpOvJtua
    9rLeSC8nEALvkLCm9fCNWFd6+uY6cEQXwVILHT/7ryVFBUS3+8vilCIgMaTGFglm
    dko74LiX8wK/WzflWs64PvEhEHW+f78NyEmvp9X5EdMZsLuPcN8jJ4QMnllaO8af
    6vjTL64RfDOd2enBEoCTFtwu3qTK6h7UshURwpyM4Jhp
    -----END CERTIFICATE-----
imageContentSources:
- mirrors:
  - registry.dmzsd.kbstar.local:5000/ocp4/openshift4
  source: quay.io/openshift-release-dev/ocp-release
- mirrors:
  - registry.dmzsd.kbstar.local:5000/ocp4/openshift4
```

```
  source: quay.io/openshift-release-dev/ocp-v4.0-art-dev
```

```
oc adm release info -a $HOME/merged_pullsecret.json
"${LOCAL_REGISTRY}/${LOCAL_REPOSITORY}:${OCP_RELEASE}-x86_64"

oc adm release info -a /paas/opt/registry/pull-secret-config.json
registry.intsd.kbstar.local:5000/ocp4/openshift4:4.3.1-x86_64
```

10) ssh-key 생성

```
ssh-keygen -t rsa -b 4096 -N '' -f ~/.ssh/id_rsa
eval "$(ssh-agent -s)"
ssh-add ~/.ssh/id_rsa
```

11) oc 설치파일 및 client 압축 해제

```
mkdir /ocp-install
mkdir /var/www/html/ignition/
tar xvf openshift-client-linux-4.3.1.tar.gz -C /usr/local/bin
tar xvf openshift-install-linux-4.3.1.tar.gz -C /ocp-install
```

12) install-config.yaml 생성

```
- metadata:
    name: dmzsd                  : 클러스터 이름 변경
- pullSecret                     : 레지스트리 접속을 위한 base64 인코딩 패스워드
- additionalTrustBundle          : id_rsa.pub 값
- imageContentSources            : 레지스트리 주소
- additionalTrustBundle          : crt 파일
```

13) openshift 설치 디렉토리 생성 및 ignition 생성 ### ignition 인증서는 24시간 후 만료

```
cp /paas/opt/registry/install-config.yaml /ocp-install
./openshift-install create install-config --dir=/ocp-install
./openshift-install create manifests --dir=/ocp-install
vi /ocp-install/manifests/cluster-scheduler-02-config.yml
  mastersschedulable: false

./openshift-install create ignition-configs --dir=/ocp-install
cp /ocp-install/*.ign /var/www/html/ignition/
chmod -R 755 /var/www/html/ignition/*
restorecon /var/www/html/ignition/*
```

14) bastion 준비 완료 후 bootstrap 부팅 시작

```
# bootstarp 설치 과정 확인

./openshift-install --dir=/ocp-install wait-for bootstrap-complete --log-
level=debug

DEBUG OpenShift Installer v4.3.0
DEBUG Built from commit 2055609f95b19322ee6cfdd0bea73399297c4a3e
INFO Waiting up to 30m0s for the Kubernetes API at
https://api.6320.blue.osp.opentlc.com:6443...
INFO API v1.16.2 up
INFO Waiting up to 30m0s for bootstrapping to complete...
DEBUG Bootstrap status: complete
INFO It is now safe to remove the bootstrap resources

# bootstrap 준비 완료 후 master, worker 차례로 구동하여 부팅시작
```

```
csr 확인, pending 시 approve
export KUBECONFIG=/ocp-install/auth/kubeconfig
oc get csr
oc adm certificate approve csr_name
```

```
clusteroperator 상택 확인 / AVAILABLE 이 모두 True 상태가 되야함

oc get clusteroperators

NAME                                       VERSION   AVAILABLE   PROGRESSING
DEGRADED   SINCE
authentication                             4.3.5     True        False
False      47h
cloud-credential                           4.3.5     True        False
False      2d
cluster-autoscaler                         4.3.5     True        False
False      2d
console                                    4.3.5     True        False
False      21h
dns                                        4.3.5     True        False
False      2d
image-registry                             4.3.5     True        False
False      21h
ingress                                    4.3.5     True        False
False      21h
insights                                   4.3.5     True        False
False      2d
kube-apiserver                             4.3.5     True        False
False      2d
kube-controller-manager                    4.3.5     True        False
False      2d
kube-scheduler                             4.3.5     True        False
False      2d
machine-api                                4.3.5     True        False
False      2d
machine-config                             4.3.5     True        False
False      2d
marketplace                                4.3.5     True        False
False      21h
```

```
monitoring                                 4.3.5     True        False
False      21h
network                                    4.3.5     True        False
False      2d
node-tuning                                4.3.5     True        False
False      21h
openshift-apiserver                        4.3.5     True        False
False      21h
openshift-controller-manager               4.3.5     True        False
False      2d
openshift-samples                          4.3.5     True        False
False      45h
operator-lifecycle-manager                 4.3.5     True        False
False      2d
operator-lifecycle-manager-catalog         4.3.5     True        False
False      2d
operator-lifecycle-manager-packageserver   4.3.5     True        False
False      21h
service-ca                                 4.3.5     True        False
False      2d
service-catalog-apiserver                  4.3.5     True        False
False      2d
service-catalog-controller-manager         4.3.5     True        False
False      2d
storage                                    4.3.5     True        False
False      45h
```

```
oc patch configs.imageregistry.operator.openshift.io cluster --type merge --
patch '{"spec":{"storage":{"emptyDir":{}}}}'

#### 4.3 버전은 emptyDir 없이 OCP 설치가 진행되기 때문에 아래 patch 명령어 필수 실행!!!
oc patch configs.imageregistry.operator.openshift.io cluster --type=merge --
patch '{"spec":{"managementState": "Managed" }}'
```

```
# bootstarp complete
./openshift-install --dir=./ wait-for install-complete

INFO Waiting up to 30m0s for the cluster at
https://api.6320.blue.osp.opentlc.com:6443 to initialize...
INFO Waiting up to 10m0s for the openshift-console route to be created...
INFO router-ca resource not found in cluster, perhaps you are not using default
router CA
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/sungkim-redhat.com/openstack-upi/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.6320.blue.osp.opentlc.com
INFO Login to the console with user: kubeadmin, password: t3JMf-ukkBH-qbPCq-
TSB3K
```

* 완료 후 설정

```
haproxy 에서 bootstarp 제거
DHCP / xinetd down
```

```
각 노드 시간 동기화 설정 / CHRONYD

/etc/chrony.conf
server 10.10.10.10 iburst
```

```
각 노드 locale 설정
timedatectl set-timezone Asia/Seoul
timedatectl status
```

```
각 노드 nmcli 설정 / 설정 후 지개동

ssh core@nclef301.intpg.kbstar.local
sudo -i
hostnamectl set-hostname nclef301.intpg.kbstar.local
nmcli con show
nmcli con mod ens192 ipv4.addresses 10.37.68.24/24
nmc|i con mod ens192 ipv4.gateway 10.37.68. 1
nmcli con mod ensi92 ipv4.dns 10.37.68. 13, 10.37.68.14
nmcli con mod ensi92 ipv4.dns-search intpg.kbstar.local
nmcli con mod ens192 ipv4.method manual
nmcli con mod ens192 ipv6.method ignore
nmcli con reload

## ens224 가 존재하는 노드일 경우 아래 명령어 실행

nmcli con show ens224
nmcti con mod ens224 ipv4.addresses 10.37.69.185/24
nmcli con mod ens224 ipv4.never-default yes
nmcli con mod ens224 ipv4.method manual
nmcli con mod ens224 ipv6.method ignore
nmcli con reload
```