

OKD4.7 installation - vmware

VM Overview - minimal spec

Machine	Type	OS	vCPU	RAM	Storage	Volume	Volume name	Remarks
okd4-bootstrap	Bootstrap (Temporary) (192.168.27.170)	Fedora CoreOS	16	16	200	Not Required	Bootstrap	Deleted once all vms are bootstrapped.
okd4-control-plane-1	Master (192.168.27.167)	Fedora CoreOS	16	16	200	Not Required	Master1	Production recommendation - 3 nodes.
okd4-compute1	Worker (192.168.27.168)	Fedora CoreOS	16	16	200	Add if ceph required.	Worker1	Production recommendation -3 nodes
okd4-services	DNS/LB/Web/NFS (192.168.27.169)	CentOS 8	4	8	200	-	-	-
A free floating IP(Not physical VM) (For multisite only)	VIP configuration for switchover of active and standby sites using Keepalived	-	-	-	-	-	-	-

1. Login okd4-service VM and update OS

```
sudo dnf install -y epel-release
sudo yum install git -y
sudo yum install wget -y
sudo dnf update -y
```

2. Setup XRDP

```
sudo dnf install -y xrdp tigervnc-server
sudo systemctl enable --now xrdp
sudo yum install vim-enhanced -y
sudo yum install firewalld -y
sudo systemctl enable firewalld
sudo systemctl start firewalld
sudo firewall-cmd --zone=public --permanent --add-port=3389/tcp
sudo firewall-cmd --reload
```

3. Configure okd4-services VM to host various services (Replace with your IP address)

```
cd
git clone http://rtx-swtl-git.fnc.net.local/scm/cicfwk/okd-4.5.git
cd okd-4.5/okd4_files/
```

4. Install bind (DNS)

```
sudo dnf -y install bind bind-utils
```

Copy the named config files and zones:

```
sudo cp named.conf /etc/named.conf
sudo cp named.conf.local /etc/named/
sudo mkdir /etc/named/zones
sudo cp db* /etc/named/zones
preferably keep /etc/sysconfig/network-scripts/ifcfg-ens192 with DNS1=127.0.0.1 DNS2=
```

Enable and start named:

```
sudo systemctl enable named
sudo systemctl start named
sudo systemctl status named
```

Create firewall rules:

```
sudo firewall-cmd --permanent --add-port=53/udp
sudo firewall-cmd --reload
```

Test DNS on the okd4-services host ist working as expected

dig okd.local

```
[crobins@okd4-services okd4_files]$ dig okd.local

; <<>> DiG 9.11.13-RedHat-9.11.13-3.el8 <<>> okd.local
;; global options: +cmd
;; Got answer:
;; WARNING: .local is reserved for Multicast DNS
;; You are currently testing what happens when an mDNS query is leaked to DNS
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64424
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 7009863cfd0c3194df61cc045f04c51c67e06b2f24fd0618 (good)
;; QUESTION SECTION:
;okd.local.                IN      A

;; AUTHORITY SECTION:
okd.local.                604800  IN      SOA      okd4-services.okd.local. admin.o
kd.local. 1 604800 86400 2419200 604800

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue Jul 07 14:55:24 EDT 2020
;; MSG SIZE rcvd: 122
```

dig -x 192.168.1.210

- a. Assume 192.168.1.210 is the DNS/Services-vm ip
- b. With DNS working correctly, you should see the following results:

```
[crobins@okd4-services okd4_files]$ dig -x 192.168.1.210

; <<>> DiG 9.11.13-RedHat-9.11.13-3.el8 <<>> -x 192.168.1.210
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 65007
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 23bfcaa3f3140ab1261acff65f04c4d921fa9f8011631ba6 (good)
;; QUESTION SECTION:
;210.1.168.192.in-addr.arpa.  IN      PTR

;; ANSWER SECTION:
210.1.168.192.in-addr.arpa. 604800  IN      PTR      api-int.lab.okd.local.
210.1.168.192.in-addr.arpa. 604800  IN      PTR      okd4-services.okd.local.
210.1.168.192.in-addr.arpa. 604800  IN      PTR      api.lab.okd.local.

;; AUTHORITY SECTION:
1.168.192.in-addr.arpa. 604800  IN      NS      okd4-services.okd.local.

;; ADDITIONAL SECTION:
okd4-services.okd.local. 604800  IN      A      192.168.1.210

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue Jul 07 14:54:17 EDT 2020
;; MSG SIZE rcvd: 194
```

5. (Optional, this is only for multi site) Install and configure Keepalived for switchover of active and standby clusters.

#Run the following commands in Service VM of both the sites

```
sudo yum install keepalived -y
sudo yum install gcc kernel-headers kernel-devel -y
sudo firewall-cmd --add-rich-rule='rule protocol value="vrrp" accept' --permanent
sudo firewall-cmd --reload
```

#Site-A, In service VM of Active cluster, Update configuration file /etc/keepalived/keepalived.conf, replace the virtual ip with your free floating IP choosen for virtual IP

```

vrrip_instance VI_1 {
    state MASTER
    interface ens192
    virtual_router_id 51
    priority 255
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.27.190/24
    }
}

```

#Site-B, In service VM of Standby cluster, Update configuration file /etc/keepalived/keepalived.conf , replace the virtual ip with your free floating IP choosen for virtual IP

```

vrrip_instance VI_1 {
    state BACKUP
    interface ens192
    virtual_router_id 51
    priority 254
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.27.190/24
    }
}

```

#Enable and start keepalived on both the sites

```

sudo systemctl enable keepalived
sudo systemctl start keepalived

```

Check Virtual IPs

By default virtual IP will be assigned to Active server, In case of Active server gets down, it will automatically assign to the Backup server. Use the following command to show assigned virtual IP on the interface:

ip addr show ens192

#Sample output

```

2: ens192: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:50:56:9b:47:25 brd ff:ff:ff:ff:ff:ff
    inet 192.168.27.150/24 brd 192.168.27.255 scope global noprefixroute ens192
        valid_lft forever preferred_lft forever
    inet 192.168.27.190/24 scope global secondary ens192
        valid_lft forever preferred_lft forever

```

6. Install HAProxy

```

sudo dnf install haproxy -y

```

Copy haproxy config from the git okd4_files directory :

```

sudo cp haproxy.cfg /etc/haproxy/haproxy.cfg

```

Start, enable, and verify HA Proxy service:

```
sudo setsebool -P haproxy_connect_any 1
sudo systemctl enable haproxy
sudo systemctl start haproxy
sudo systemctl status haproxy
```

Add OKD firewall ports:

```
sudo firewall-cmd --permanent --add-port=6443/tcp
sudo firewall-cmd --permanent --add-port=22623/tcp
sudo firewall-cmd --permanent --add-service=http
sudo firewall-cmd --permanent --add-service=https
sudo firewall-cmd --reload
```

7. Install Apache/HTTPD

```
sudo dnf install -y httpd
```

Change httpd to listen port to 8080:

```
sudo sed -i 's/Listen 80/Listen 8080/' /etc/httpd/conf/httpd.conf
```

Enable and Start httpd service/Allow port 8080 on the firewall:

```
sudo setsebool -P httpd_read_user_content 1
sudo systemctl enable httpd
sudo systemctl start httpd
sudo firewall-cmd --permanent --add-port=8080/tcp
sudo firewall-cmd --reload
```

Test the webserver:

```
curl localhost:8080
```

8. Download the openshift-installer and oc client

Download the 4.7 version of the oc client and openshift-install from the [OKD releases page](#).

```
cd
wget https://github.com/openshift/okd/releases/download/4.7.0-0.okd-2021-05-22-050008/openshift-client-linux-4.7.0-0.okd-2021-05-22-050008.tar.gz
wget https://github.com/openshift/okd/releases/download/4.7.0-0.okd-2021-05-22-050008/openshift-install-linux-4.7.0-0.okd-2021-05-22-050008.tar.gz
```

For OCP4.7	Binary Path
client	wget https://mirror.openshift.com/pub/openshift-v4/clients/ocp/stable/openshift-client-linux.tar.gz
installer	wget https://mirror.openshift.com/pub/openshift-v4/clients/ocp/stable/openshift-install-linux.tar.gz

Extract the okd version of the oc client and openshift-install:

```
tar -zxvf openshift-client-linux-4.7.0-0.okd-2021-05-22-050008.tar.gz
tar -zxvf openshift-install-linux-4.7.0-0.okd-2021-05-22-050008.tar.gz
```

Move the kubectl, oc, and openshift-install to /usr/local/bin and show the version:

```
sudo mv kubectl oc openshift-install /usr/local/bin/
oc version
openshift-install version
```

9. Setup the openshift-installer:

Generate an SSH key:

```
ssh-keygen
eval "$(ssh-agent -s)"
ssh-add ~/.ssh/id_rsa
```

Create an install directory and copy the install-config.yaml file:

```
cd
mkdir install_dir
cp okd-4.5/okd4_files/install-config.yaml ./install_dir
```

Edit the install-config.yaml in the install_dir, insert your pull secret(copy from [Pull Secret page](#))and ssh key(~/.ssh/id_rsa.pub), and

backup the install-config.yaml as it will be deleted in the next step:

```
vim ./install_dir/install-config.yaml
cp ./install_dir/install-config.yaml ./install_dir/install-config.yaml.bak
```

Generate the Kubernetes manifests for the cluster, ignore the warning:

```
openshift-install create manifests --dir=install_dir/
```

Modify the cluster-scheduler-02-config.yaml manifest file to prevent Pods from being scheduled on the control plane machines:

```
sed -i 's/mastersSchedulable: true/mastersSchedulable: False/' install_dir/manifests/cluster-scheduler-02-config.yml
```

Create ignition-configs:

```
openshift-install create ignition-configs --dir=install_dir/
```

Note: If you reuse the install_dir, make sure it is empty. Hidden files are created after generating the configs, and they should be removed before you use the same folder on a 2nd attempt.

10. Host ignition and Fedora CoreOS files on the webserver

Create okd4 directory in /var/www/html:

```
sudo mkdir /var/www/html/okd4
```

Copy the install_dir contents to /var/www/html/okd4 and set permissions:

```
sudo cp -R install_dir/* /var/www/html/okd4/
sudo chown -R apache: /var/www/html/
sudo chmod -R 755 /var/www/html/
```

Test the webserver:

```
curl localhost:8080/okd4/metadata.json
```

Download the Fedora CoreOS bare-metal bios image and sig files and shorten the file names:

```
cd /var/www/html/okd4/
sudo wget https://builds.coreos.fedoraproject.org/prod/streams/stable/builds/33.20210217.3.0/x86_64/fedora-coreos-33.20210217.3.0-metal.x86_64.raw.xz
sudo wget https://builds.coreos.fedoraproject.org/prod/streams/stable/builds/33.20210217.3.0/x86_64/fedora-coreos-33.20210217.3.0-metal.x86_64.raw.xz.sig
```

For OCP4.7	Binary Path
client	sudo wget
installer	sudo wget

```
sudo mv fedora-coreos-33.20210217.3.0-metal.x86_64.raw.xz fcos.raw.xz
sudo mv fedora-coreos-33.20210217.3.0-metal.x86_64.raw.xz.sig fcos.raw.xz.sig
sudo chown -R apache: /var/www/html/
sudo chmod -R 755 /var/www/html
```

11. Starting the Bootstrap node

Download the [Fedora CoreOS Bare Metal ISO](#) image and upload it in Openstack cluster. Create a new test VM using fedora live ISO image and attach it to bootstrap volume for installing configuration file.

Power on the test VM and open VM console from openstack dashboard. Press the TAB key to edit the kernel boot options and add the following:

```
ip=192.168.27.170::192.168.27.1:255.255.255.0:okd4-bootstrap.lab.okd.local:ens192:none
nameserver=192.168.27.169
nameserver=168.127.132.3
nameserver=168.127.132.4
coreos.inst.install_dev=sda
coreos.inst.image_url=http://192.168.27.169:8080/okd4/fcos.raw.xz
coreos.inst.ignition_url=http://192.168.27.169:8080/okd4/bootstrap.ign
```

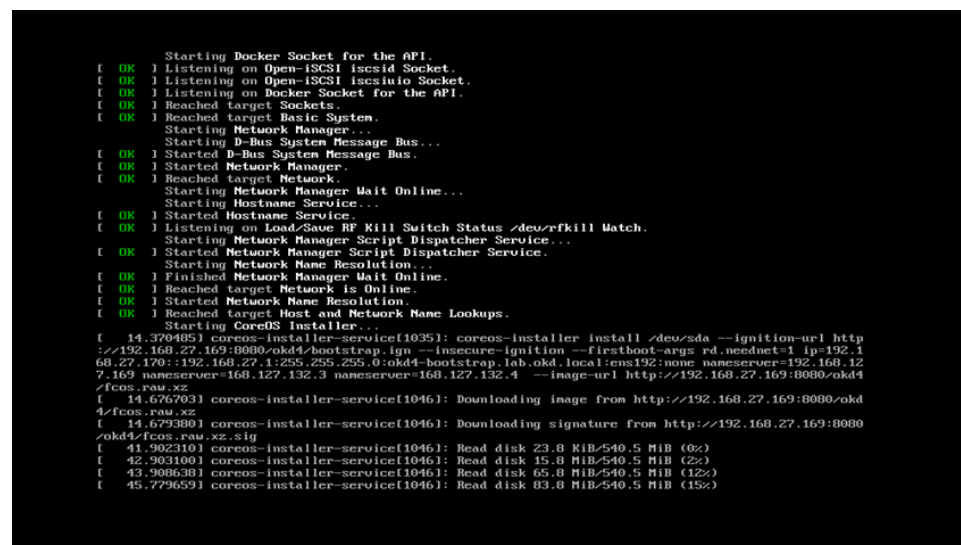


```
Fedora CoreOS

Fedora CoreOS (Live)

> /images/pxeboot/vmlinuz initrd=/images/pxeboot/initrd.img,/images/ignition.i
mg mitigations=auto,nosmt systemd.unified_cgroup_hierarchy=0 coreos.liveiso=fe
> /images/pxeboot/vmlinuz initrd=/images/pxeboot/initrd.img,/images/ignition.i
mg mitigations=auto,nosmt systemd.unified_cgroup_hierarchy=0 coreos.liveiso=fe
do/images/pxeboot/vmlinuz initrd=/images/pxeboot/initrd.img,/images/ignition.i
mg mitigations=auto,nosmt systemd.unified_cgroup_hierarchy=0 coreos.liveiso=fe
dora-coreos-33.28218217.3.0 ignition.firstboot ignition.platform.id=metal ip=1
92.168.27.170::192.168.27.1:255.255.255.0:okd4-bootstrap.lab.okd.local:ens192:
none nameserver=192.168.27.169 nameserver=168.127.132.3 nameserver=168.127.132
.4 coreos.inst.install_dev=sda coreos.inst.image_url=http://192.168.27.169:808
0/okd4/fcos.raw.xz coreos.inst.ignition_url=http://192.168.27.169:8080/okd4/b
ootstrap.ign_
```

You should see that the fcos.raw.gz image and signature are downloading:



```
Starting Docker Socket for the API.
[ OK ] Listening on Open-SCSI iscsid Socket.
[ OK ] Listening on Open-SCSI iscsiuto Socket.
[ OK ] Listening on Docker Socket for the API.
[ OK ] Reached target Sockets.
[ OK ] Reached target Basic System.
Starting Network Manager...
Starting D-Bus System Message Bus...
[ OK ] Started D-Bus System Message Bus.
[ OK ] Started Network Manager.
[ OK ] Reached target Network.
Starting Network Manager Wait Online...
Starting Hostname Service...
[ OK ] Started Hostname Service.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
Starting Network Manager Script Dispatcher Service...
[ OK ] Started Network Manager Script Dispatcher Service.
Starting Network Name Resolution...
[ OK ] Finished Network Manager Wait Online.
[ OK ] Reached target Network is Online.
[ OK ] Started Network Name Resolution.
[ OK ] Reached target Host and Network Name Lookups.
Starting CoreOS Installer...
[ 14.379485] coreos-installer-service[1035]: coreos-installer install /dev/sda --ignition-url http
://192.168.27.169:8080/okd4/bootstrap.ign --insecure-ignition --firstboot-args rd,neednet=1 ip=192.1
68.27.170::192.168.27.1:255.255.255.0:okd4-bootstrap.lab.okd.local:ens192:none nameserver=192.168.12
7.169 nameserver=168.127.132.3 nameserver=168.127.132.4 --image-url http://192.168.27.169:8080/okd4
/fcos.raw.xz
[ 14.676703] coreos-installer-service[1046]: Downloading image from http://192.168.27.169:8080/okd
4/fcos.raw.xz
[ 14.679380] coreos-installer-service[1046]: Downloading signature from http://192.168.27.169:8080
/okd4/fcos.raw.xz.sig
[ 41.902310] coreos-installer-service[1046]: Read disk 23.8 KiB/540.5 MiB (0%)
[ 42.903100] coreos-installer-service[1046]: Read disk 15.8 MiB/540.5 MiB (2%)
[ 43.908638] coreos-installer-service[1046]: Read disk 65.8 MiB/540.5 MiB (12%)
[ 45.779659] coreos-installer-service[1046]: Read disk 83.8 MiB/540.5 MiB (15%)
```

12. Starting the control-plane nodes

Power on the VM and click on Console. Press the TAB key to edit the kernel boot options and add the following, then press enter:

```
ip=192.168.27.167::192.168.27.1:255.255.255.0:okd4-control-plane-1.lab.okd.local:ens192:none
nameserver=192.168.27.169
nameserver=168.127.132.3
nameserver=168.127.132.4
coreos.inst.install_dev=sda
coreos.inst.image_url=http://192.168.27.169:8080/okd4/fcos.raw.xz
coreos.inst.ignition_url=http://192.168.27.169:8080/okd4/master.ign
```



You should see that the fcos.raw.gz image and signature are downloading:

```

Starting D-Bus System Message Bus...
[ OK ] Started D-Bus System Message Bus.
[ OK ] Started Network Manager.
[ OK ] Reached target Network.
Starting Network Manager Wait Online...
Starting Hostname Service...
[ OK ] Started Hostname Service.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
Starting Network Manager Script Dispatcher Service...
[ OK ] Started Network Manager Script Dispatcher Service.
Starting Network Name Resolution...
[ OK ] Finished Network Manager Wait Online.
[ OK ] Reached target Network is Online.
[ OK ] Started Network Name Resolution.
[ OK ] Reached target Host and Network Name Lookups.
Starting CoreOS Installer...
[ 14.666859] coreos-installer-service[1039]: coreos-installer install /dev/sda --ignition-url http
://192.168.27.169:8080/okd4/master.ign --insecure-ignition --firstboot-args rd.neednet=1 ip=192.168.
27.167::192.168.27.1:255.255.255.0:okd4-control-plane-1.lab.okd.local:ens192:none nameserver=192.168
.27.169 nameserver=168.127.132.3 nameserver=168.127.132.4 --image-url http://192.168.27.169:8080/ok
d4/fcos.raw.xz
[ 15.753071] coreos-installer-service[1050]: Downloading image from http://192.168.27.169:8080/okd
4/fcos.raw.xz
[ 15.754098] coreos-installer-service[1050]: Downloading signature from http://192.168.27.169:8080
/okd4/fcos.raw.xz.sig
[ 17.236419] coreos-installer-service[1050]: Read disk 462.2 KiB/540.5 MiB (0%)
[ 18.237298] coreos-installer-service[1050]: Read disk 17.0 MiB/540.5 MiB (3%)
[ 19.239888] coreos-installer-service[1050]: Read disk 62.4 MiB/540.5 MiB (11%)
[ 20.268897] coreos-installer-service[1050]: Read disk 83.5 MiB/540.5 MiB (15%)
[ 21.451947] coreos-installer-service[1050]: Read disk 83.9 MiB/540.5 MiB (15%)
[ 23.615566] coreos-installer-service[1050]: Read disk 85.4 MiB/540.5 MiB (15%)
[ 24.624022] coreos-installer-service[1050]: Read disk 90.3 MiB/540.5 MiB (16%)
[ 25.635924] coreos-installer-service[1050]: Read disk 99.5 MiB/540.5 MiB (18%)
[ 26.645688] coreos-installer-service[1050]: Read disk 107.7 MiB/540.5 MiB (19%)
[ 27.650223] coreos-installer-service[1050]: Read disk 115.6 MiB/540.5 MiB (21%)
[ 28.675042] coreos-installer-service[1050]: Read disk 124.6 MiB/540.5 MiB (23%)
-

```

Repeat the same process for okd4-control-plane-2 and okd4-control-plane-3 VM.

13. Starting the compute nodes

Power on the test VM and open VM console from openstack dashboard. Press the TAB key to edit the kernel boot options and add the following:

```

ip=192.168.27.168::192.168.27.1:255.255.255.0:okd4-compute-1.lab.okd.local:ens192:none
nameserver=192.168.27.169
nameserver=168.127.132.3
nameserver=168.127.132.4
coreos.inst.install_dev=sda
coreos.inst.image_url=http://192.168.27.169:8080/okd4/fcos.raw.xz
coreos.inst.ignition_url=http://192.168.27.169:8080/okd4/worker.ign

```


Fedora CoreOS

Fedora CoreOS (Live)

```
> /images/pxeboot/vmlinuz initrd=/images/pxeboot/initrd.img,/images/ignition.i
mg mitigations=auto,nosmt systemd.unified_cgroup_hierarchy=0 coreos.liveiso=fe
dora-coreos-33.20210217.3.0 ignition.firstboot ignition.platform.id=metal ip=1
92.168.27.168::192.168.27.1:255.255.255.0:okd4-compute-1.lab.okd.local:ens192:
none nameserver=192.168.27.169 nameserver=168.127.132.3 nameserver=168.127.132
.4 coreos.inst.install_dev=sda coreos.inst.image_url=http://192.168.27.169:8080
0/okd4/fcos.raw.xz coreos.inst.ignition_url=http://192.168.27.169:8080/okd4/wo
rker.ign_
```

You should see that the fcos.raw.gz image and signature are downloading:

```
[ OK ] Listening on Open-iSCSI iscsid Socket.
[ OK ] Listening on Open-iSCSI iscsiui Socket.
[ OK ] Listening on Docker Socket for the API.
[ OK ] Reached target Sockets.
[ OK ] Reached target Basic System.
Starting Network Manager...
Starting D-Bus System Message Bus...
[ OK ] Started D-Bus System Message Bus.
[ OK ] Started Network Manager.
[ OK ] Reached target Network.
Starting Network Manager Wait Online...
Starting Hostname Service...
[ OK ] Started Hostname Service.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
Starting Network Manager Script Dispatcher Service...
[ OK ] Started Network Manager Script Dispatcher Service.
Starting Network Name Resolution...
[ OK ] Finished Network Manager Wait Online.
[ OK ] Reached target Network is Online.
[ OK ] Started Network Name Resolution.
[ OK ] Reached target Host and Network Name Lookups.
Starting CoreOS Installer...
[ 9.577291] coreos-installer-service[1037]: coreos-installer install /dev/sda --ignition-url http
://192.168.27.169:8080/okd4/worker.ign --insecure-ignition --firstboot-args rd.neednet=1 ip=192.168.
27.168::192.168.27.1:255.255.255.0:okd4-compute-1.lab.okd.local:ens192:none nameserver=192.168.27.16
9 nameserver=168.127.132.3 nameserver=168.127.132.4 --image-url http://192.168.27.169:8080/okd4/fco
s.raw.xz
[ 9.828740] coreos-installer-service[1048]: Downloading image from http://192.168.27.169:8080/okd
4/fcos.raw.xz
[ 9.831414] coreos-installer-service[1048]: Downloading signature from http://192.168.27.169:8080
/okd4/fcos.raw.xz.sig
[ 11.037185] coreos-installer-service[1048]: Read disk 12.4 MiB/540.5 MiB (2%)
[ 12.048362] coreos-installer-service[1048]: Read disk 64.0 MiB/540.5 MiB (11%)
[ 13.781347] coreos-installer-service[1048]: Read disk 83.9 MiB/540.5 MiB (15%)
[ 15.635414] coreos-installer-service[1048]: Read disk 85.4 MiB/540.5 MiB (15%)
[ 16.644474] coreos-installer-service[1048]: Read disk 90.1 MiB/540.5 MiB (16%)
```

Repeat the same process for okd4-compute2 VM.

It is usual for the worker nodes to display the following until the bootstrap process complete:

Instance Console

If console is not responding to keyboard input, click the grey status bar below. [Click here to show only console](#)
To exit the fullscreen mode, click the browser's back button.

```
Connected (unencrypted) to: QEMU (instance-00000426) Send CHARSel
[ 540.515162] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #112
[ 540.527958] ignition(481): GET result: Internal Server Error
[ 540.528382] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #113
[ 540.531980] ignition(481): GET result: Internal Server Error
[ 540.532494] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #114
[ 540.536400] ignition(481): GET result: Internal Server Error
[ 540.536619] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #115
[ 540.540191] ignition(481): GET result: Internal Server Error
[ 540.540758] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #116
[ 540.544388] ignition(481): GET result: Internal Server Error
[ 540.544822] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #117
[ 540.548383] ignition(481): GET result: Internal Server Error
[ 540.548953] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #118
[ 540.552462] ignition(481): GET result: Internal Server Error
[ 540.553206] ignition(481): GET https://api-int.lab.okd.local:22623/config-war
ker: attempt #119
[ 540.556711] ignition(481): GET result: Internal Server Error
```

14. Monitor the bootstrap installation

You can monitor the bootstrap process from the okd4-services node:

`openshift-install --dir=install_dir/ wait-for bootstrap-complete --log-level=debug`

```
[kubenode@okd4-services ~]$ openshift-install --dir=install_dir/ wait-for bootstrap-complete --log-level=debug
DEBUG OpenShift Installer 4.7.0-0.okd-2021-05-22-050008
DEBUG Built from commit ba1a5fc61dd3be134a24fa6d89ae24012a93385a
INFO Waiting up to 20m0s for the Kubernetes API at https://api.lab.okd.local:6443...
INFO API v1.20.0-1073+df9c8387b2dc23-dirty up
INFO Waiting up to 30m0s for bootstrapping to complete...
DEBUG Bootstrap status: complete
INFO It is now safe to remove the bootstrap resources
DEBUG Time elapsed per stage:
DEBUG Bootstrap Complete: 10m45s
INFO Time elapsed: 10m45s
```

Once the bootstrap process is complete, which can take upwards of 30 minutes, you can shutdown your bootstrap node and delete the VM. Edit the `/etc/haproxy/haproxy.cfg`, comment out the bootstrap node, and reload the haproxy service.

`sudo sed '/okd4-bootstrap/s/^/#/' /etc/haproxy/haproxy.cfg`

`sudo systemctl reload haproxy`

15. Login to the cluster and approve CSRs

`export KUBECONFIG=~/.install_dir/auth/kubeconfig`

`oc whoami`

`oc get nodes`

`oc get csr`

`cp ~/.install_dir/auth/kubeconfig ~/.kube/config`

(To ssh worker/master nodes from service node: `sudo ssh core@IP_address_master/worker`)

Install the jq package to assist with approving multiple CSR's at once time.

`wget -O jq https://github.com/stedolan/jq/releases/download/jq-1.6/jq-linux64`

`chmod +x jq`

`sudo mv jq /usr/local/bin/`

`jq --version`

Approve all the pending certs and check your nodes:

`oc get csr -ojson | jq -r '.items[] | select(.status == {})' | xargs oc adm certificate approve`

Check status of the cluster operators and cluster version.

oc get clusteroperators # all version should be true in available tab
oc get clusterversion

```
[kubenode@okd4-services ~]$ oc get co
NAME                                     VERSION                                     AVAILABLE PROGRESSING DEGRADED SINCE
authentication                          4.7.0-0.okd-2021-05-22-050008             True      False      False  39m
baremetal                              4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
cloud-credential                        4.7.0-0.okd-2021-05-22-050008             True      False      False  19h
cluster-autoscaler                     4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
config-operator                        4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
console                                4.7.0-0.okd-2021-05-22-050008             True      False      False  15h
csi-snapshot-controller                4.7.0-0.okd-2021-05-22-050008             True      False      False  39m
dns                                     4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
etcd                                    4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
image-registry                         4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
ingress                                4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
insights                               4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
kube-apiserver                         4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
kube-controller-manager                4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
kube-scheduler                         4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
kube-storage-version-migrator           4.7.0-0.okd-2021-05-22-050008             True      False      False  4h15m
machine-api                            4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
machine-approver                       4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
machine-config                         4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
marketplace                            4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
monitoring                             4.7.0-0.okd-2021-05-22-050008             True      False      False  38m
network                                4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
node-tuning                            4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
openshift-apiserver                     4.7.0-0.okd-2021-05-22-050008             True      False      False  110m
openshift-controller-manager            4.7.0-0.okd-2021-05-22-050008             True      False      False  176m
openshift-samples                       4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
operator-lifecycle-manager              4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
operator-lifecycle-manager-catalog      4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
operator-lifecycle-manager-packageserver 4.7.0-0.okd-2021-05-22-050008             True      False      False  16h
service-ca                              4.7.0-0.okd-2021-05-22-050008             True      False      False  18h
storage                                 4.7.0-0.okd-2021-05-22-050008             True      False      False  18h

[kubenode@okd4-services ~]$
[kubenode@okd4-services ~]$ oc get clusterversion
NAME      VERSION      AVAILABLE PROGRESSING SINCE STATUS
version  4.7.0-0.okd-2021-05-22-050008 True      False      15h Cluster version is 4.7.0-0.okd-2021-05-22-050008
[kubenode@okd4-services ~]$
```

Check status of nodes

```
Every 2.0s: oc get nodes

NAME                                     STATUS    ROLES    AGE    VERSION
okd4-compute-1.lab.okd.local            Ready     worker   5m35s  v1.20.0+df9c838-1073
okd4-compute-2.lab.okd.local            Ready     worker   5m28s  v1.20.0+df9c838-1073
okd4-control-plane-1.lab.okd.local       Ready     master   32m    v1.20.0+df9c838-1073
okd4-control-plane-2.lab.okd.local       Ready     master   28m    v1.20.0+df9c838-1073
okd4-control-plane-3.lab.okd.local       Ready     master   22m    v1.20.0+df9c838-1073
```

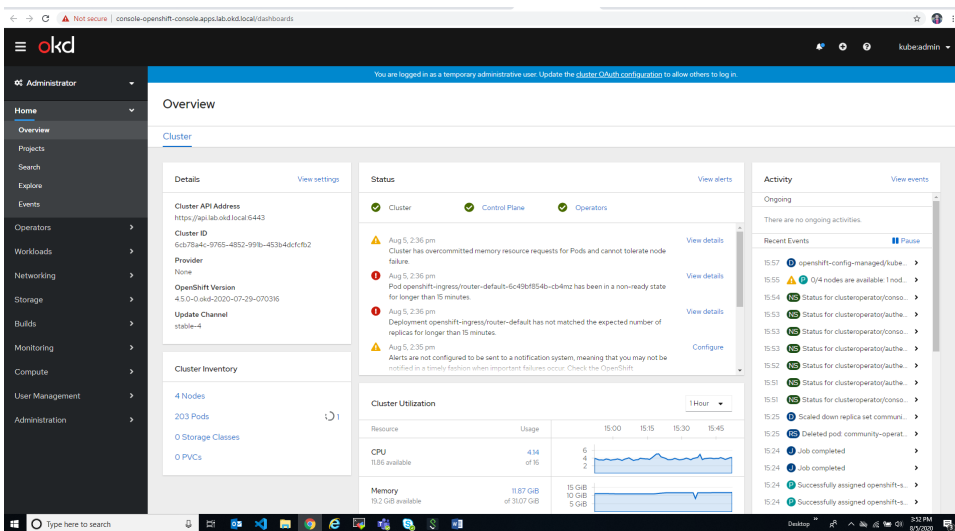
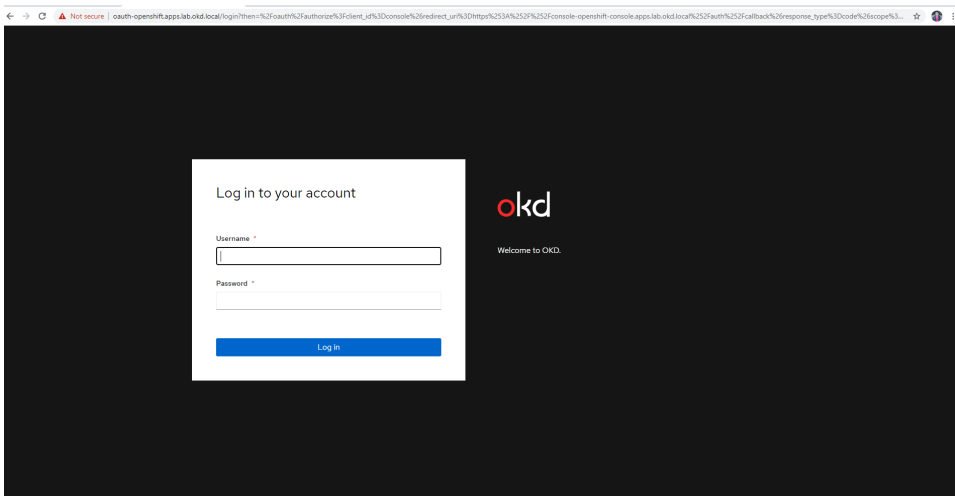
Get kubeadmin password from the install_dir/auth folder and login to the web console:

cat install_dir/auth/kubeadmin-password

Update the RDP machine /etc/hosts with below entries to access the OKD Dashboard:

192.168.27.150 console-openshift-console.apps.lab.okd.local
192.168.27.150 oauth-openshift.apps.lab.okd.local

Open web browser to <https://console-openshift-console.apps.lab.okd.local/> and login as kubeadmin with the password from above:



16. HTPasswd Setup:

The kubeadmin is a temporary user. The easiest way to set up a local user is with htpasswd.

```
cd
cd okd-4.5/okd4_files/
htpasswd -c -B -b users.htpasswd admin admin
```

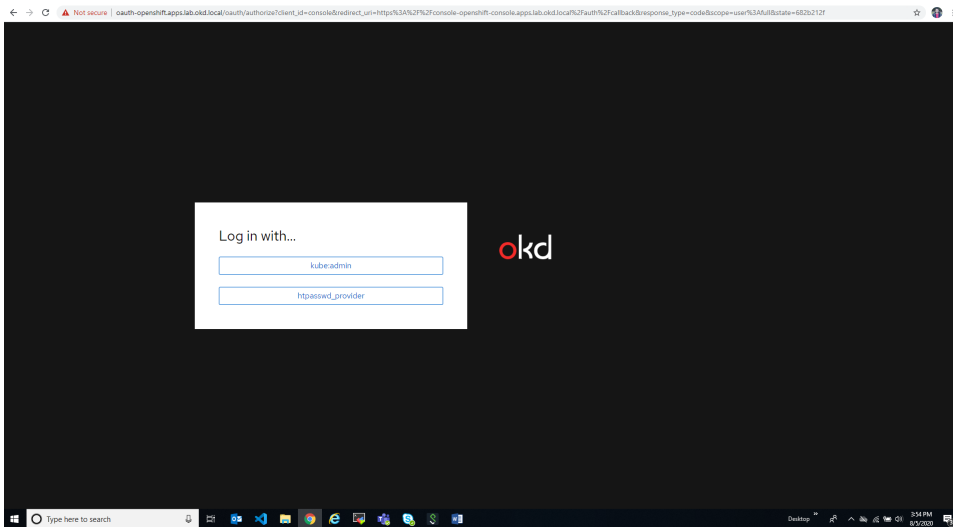
Create a secret in the openshift-config project using the users.htpasswd file you generated.

```
oc create secret generic htpass-secret --from-file=htpasswd=users.htpasswd -n openshift-config
```

Add the identity provider.

```
oc apply -f htpasswd_provider.yaml
```

Logout of the OpenShift Console. Then select htpasswd_provider and login with admin and admin credentials.



17. Check the status of installation.

`openshift-install wait-for install-complete --log-level="debug"`

```
[kubenode@okd4-services install_dir]$ openshift-install wait-for install-complete --log-level="debug"
DEBUG OpenShift Installer 4.7.0-0.okd-2021-05-22-050008
DEBUG Built from commit bala5fc61dd3be134a24faed89ae24012a93385a
DEBUG Loading Install Config...
DEBUG Loading SSH Key...
DEBUG Loading Base Domain...
DEBUG Loading Platform...
DEBUG Loading Cluster Name...
DEBUG Loading Base Domain...
DEBUG Loading Platform...
DEBUG Loading Networking...
DEBUG Loading Platform...
DEBUG Loading Pull Secret...
DEBUG Loading Platform...
DEBUG Using Install Config loaded from state file
INFO Waiting up to 40m0s for the cluster at https://api.lab.okd.local:6443 to initialize...
DEBUG Cluster is initialized
INFO Waiting up to 10m0s for the openshift-console route to be created...
DEBUG Route found in openshift-console namespace: console
DEBUG OpenShift console route is admitted
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/home/kubenode/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-console.apps.lab.okd.local
INFO Login to the console with user: "kubeadmin", and password: "nGunu-rHDrh-WM7i-XYk2x"
INFO Time elapsed: 0s
[kubenode@okd4-services install_dir]$
```

Procedure to add extra worker node to OKD4.7 cluster:

1. Boot a Fedora CoreOS with same version used for the creating OKD4.7 cluster.
2. Start the VM and move to console tab and press the TAB key to edit the kernel boot options and add the following, then press enter:

```
ip=192.168.27.171::192.168.27.1:255.255.255.0:okd4-compute-2.lab.okd.local:ens192:none
nameserver=192.168.27.169
nameserver=168.127.132.3
nameserver=168.127.132.4
coreos.inst.install_dev=sda
coreos.inst.image_url=http://192.168.27.169:8080/okd4/fcos.raw.xz
coreos.inst.ignition_url=http://192.168.27.169:8080/okd4/worker.ign
```

3. Approve all the pending certs and check your nodes:

```
oc get csr -o json | jq -r '.items[] | select(.status == {} ) | .metadata.name' | xargs oc adm certificate approve
```

VMWare OCP 4.7 Installation Binaries:

- 1) Download the openshift-installer and oc client

Download the RHCOS 4.7 version of the oc client and openshift-install from the [OKD releases page](#).

cd

wget <https://mirror.openshift.com/pub/openshift-v4/clients/ocp/stable/openshift-client-linux.tar.gz>

wget <https://mirror.openshift.com/pub/openshift-v4/clients/ocp/stable/openshift-install-linux.tar.gz>

Source:

<https://cloud.redhat.com/openshift/install/metal/user-provisioned>

https://docs.openshift.com/container-platform/4.7/installing/installing_bare_metal_ipi/ipi-install-installation-workflow.html

Extract the okd version of the oc client and openshift-install:

tar -zxvf openshift-client-linux.tar.gz

tar -zxvf openshift-install-linux.tar.gz

2) Host ignition and Fedora CoreOS files on the webserver

Create okd4 directory in /var/www/html:

sudo mkdir /var/www/html/ocp4

Copy the install_dir contents to /var/www/html/okd4 and set permissions:

sudo cp -R install_dir/* /var/www/html/ocp4/

sudo chown -R apache: /var/www/html/

sudo chmod -R 755 /var/www/html/

Test the webserver:

curl localhost:8080/ocp4/metadata.json

Download the RHCOS bare-metal bios image and sig files and shorten the file names:

cd /var/www/html/ocp4/

sudo wget https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/latest/latest/rhcos-metal.x86_64.raw.gz

sudo mv rhcos-metal.x86_64.raw.gz rhcos.raw.gz

sudo chown -R apache: /var/www/html/

sudo chmod -R 755 /var/www/html

Power on the bootstrap VM and open VM console from VMware dashboard. reboot the machine and Press the TAB key to edit the kernel boot options and add the following:

ip=192.168.27.164::192.168.27.1:255.255.255.0:okd4-bootstrap.lab.okd.local:ens192:none

nameserver=192.168.27.86

nameserver=168.127.132.3

nameserver=168.127.132.4

coreos.inst.install_dev=sda

coreos.inst.image_url=<http://192.168.27.86:8080/okd4/rhcos.raw.gz>

coreos.inst.ignition_url=<http://192.168.27.86:8080/okd4/bootstrap.ign>

coreos.inst.insecure=yes

Power on all the master VM's and open VM console from VMware dashboard. reboot the machine and Press the TAB key to edit the kernel boot options and add the following:

ip=192.168.27.165::192.168.27.1:255.255.255.0:okd4-control-plane-1.lab.okd.local:ens192:none

nameserver=192.168.27.86

nameserver=168.127.132.3

nameserver=168.127.132.4

coreos.inst.install_dev=sda

coreos.inst.image_url=<http://192.168.27.86:8080/okd4/rhcos.raw.gz>

coreos.inst.ignition_url=<http://192.168.27.86:8080/okd4/master.ign>

coreos.inst.insecure=yes

Power on all the worker VM's and open VM console from VMware dashboard. reboot the machine and Press the TAB key to edit the kernel boot options and add the following:

```
ip=192.168.27.168::192.168.27.1:255.255.255.0:okd4-compute-1.lab.okd.local:ens192:none
nameserver=192.168.27.86
nameserver=168.127.132.3
nameserver=168.127.132.4
coreos.inst.install_dev=sda
coreos.inst.image_url=http://192.168.27.86:8080/okd4/rhcos.raw.gz
coreos.inst.ignition_url=http://192.168.27.86:8080/okd4/worker.ign
coreos.inst.insecure=yes
```