Red Hat
OpenShift Container Platform

User Experiences in Setting Up OpenShift on z/VM

Samuel D. Cohen
Levi, Ray & Shoup, Inc.
sam.cohen@lrs.com
(217) 862-9227

Who am I?

- Retired IBMer, now working with a Premier IBM Business Partner
- Specialist in z/VM, z/VSE and Linux
- Working with z/VM and z/VSE since 1987
- Working with Linux on z since it was released at Marist University

What I was trying to do

- Understand Openshift better
  - Learn more about the technology and terminology
- See what it takes to setup a standalone demo environment under z/VM
- Try to understand the Redhat-provided documentation
- Provide “lessons learned” for others that want to try this at home
- I can’t answer what you would do with this, other than impress your Kubernetes-centric colleagues

Credits

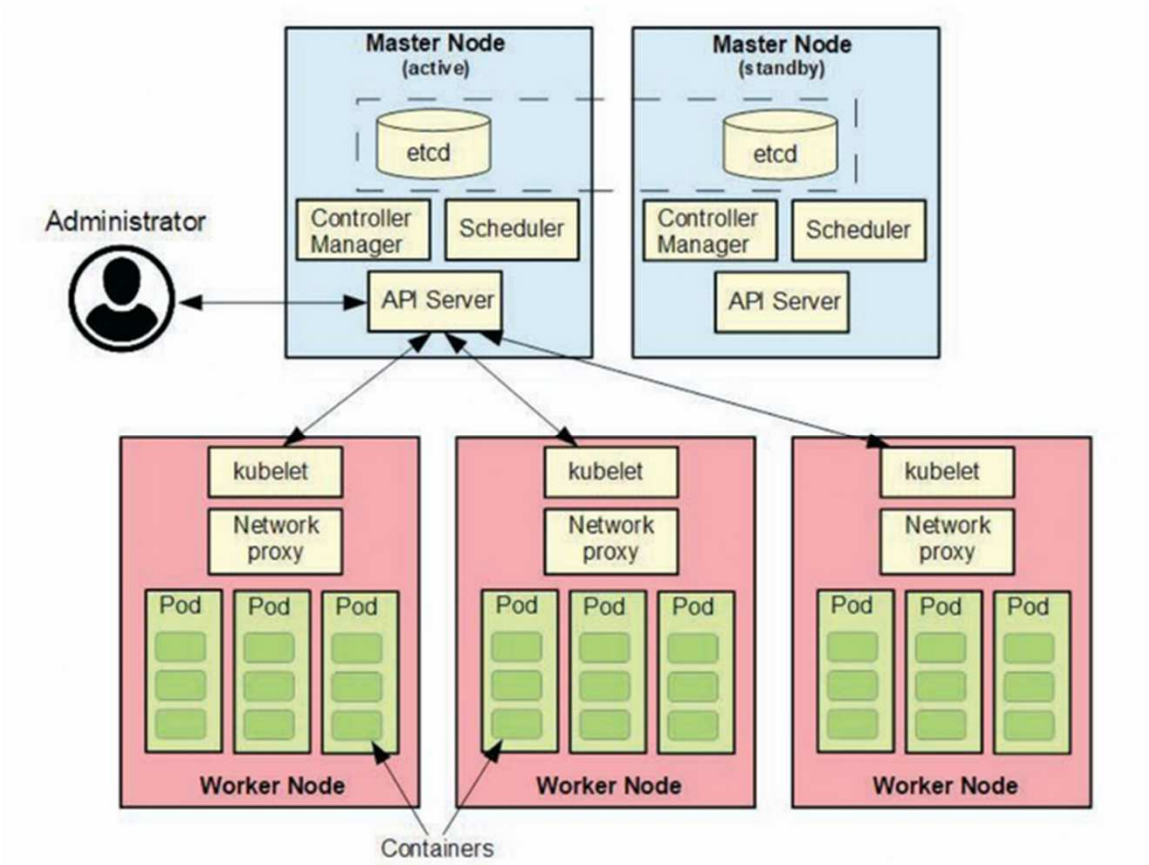
- S. Michael Benson, *Docker Swarm or Kubernetes?*, Enterprise Tech Journal, 2019 Issue 6.
- Filipe Miranda, IBM, published articles in Linkedin on his experience and provided sample settings for DNS, Load Balancer. Also provided 1:1 assistance
  - https://www.linkedin.com/pulse/red-hat-openshift-installation-process-experiences-ibm-filipe-miranda/
  - https://www.linkedin.com/pulse/understanding-network-definitions-from-openshift-4-ibm-filipe-miranda/ describes network settings used internally by Openshift

Understand OpenShift Better

- Redhat uses Kubernetes as a deployment and management engine for “pods”
- Terminology
  - Pod: a business application, made up of one or more containers
  - Service: multiple identical pods distributed throughout the cluster for load balancing and higher availability
  - Node: a virtual machine (in our case) where multiple pods can be deployed
  - Master Node: schedules and controls pods across multiple worker nodes
  - Worker Node: where pods are actually executing
  - Kubelet: an agent in each worker node for communication with the master nodes

Reference: S. Michael Benson, *Docker Swarm or Kubernetes?*, Enterprise Tech Journal, 2019 Issue 6. Used with permission

Better shown in a picture

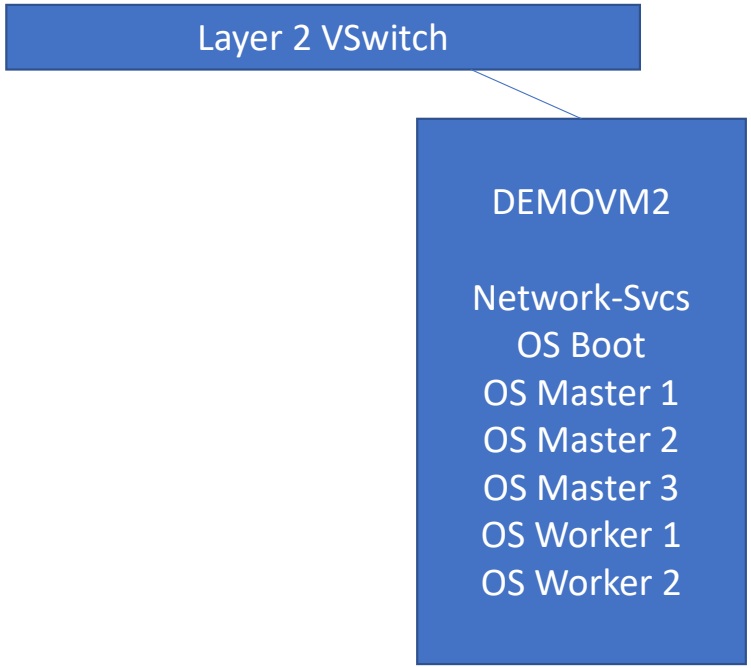


Reference: S. Michael Benson, *Docker Swarm or Kubernetes?*, Enterprise Tech Journal, 2019 Issue 6. Used with permission

Deployment process

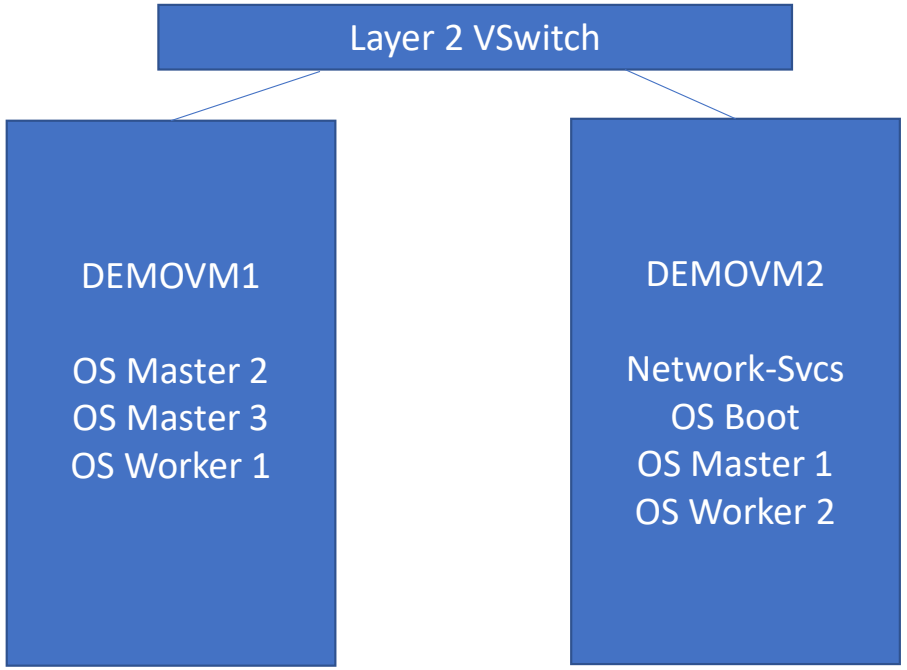
- Master nodes and worker nodes use CoreOS as the operating system
  - CoreOS is lighter weight than RHEL (even in minimal installation)
- A “boot” node is created which will create the master/worker nodes
  - Node is initially installed like RHEL via VM Reader with boot, parmfile and initrd
  - After code is laid down from a base image (one for ECKD, one for FB), CoreOS uses an “ignition file” to customize the base image
  - Ignition files are created using an Openshift-provided tool based on customer-provided instructions
    - Instructions written in YAML (recursive acronym: YAML Ain’t Markup Language)
    - Similar to XML in concept
    - Positional input, very picky

Openshift Demo Environment (Initial)



Management network:	10.96.64.192/28
DNS Domain:	osdemo.lrsinc.org
Kubernetes “service” network:	internal
DNS Domain:	democluster.osdemo.lrsinc.org

Openshift Demo Environment (Final)



Management network:	10.96.64.192/28
DNS Domain:	osdemo.lrsinc.org
Kubernetes “service” network:	internal
DNS Domain:	democluster.osdemo.lrsinc.org

Getting Started

- Ground rules:
  - No firewalls
  - Layer 2 Vswitch for this environment
    - Must fit into existing network structure (e.g. no network changes)
      - Existing network: 10.96.64.128/25, gateway 10.96.64.254, 10.96.64.250-254 reserved
  - Placed required services in a single virtual machine
    - FTP
    - DHCP
    - DDNS
    - Load Balancer
  - Using FCP-attached storage due to limited amount of ECKD DASD
    - Openshift uses single path only (at least for now)
- Download Openshift (https://try.openshift.com)
  - (https://cloud.redhat.com/openshift/install/ibmz/user-provisioned)
- Print documentation:
  - https://docs.openshift.com/container-platform/4.3/installing/installing_ibm_z/installing-ibm-z.html

Layer 2 Vswitch (from SYSTEM CONFIG)

Rdevice 0200-022F EQID ETH00200 Type OSA

Define VSwitch OSVSW Rdev 0223.P00 Ethernet VLAN Unaware

Directory Profiles for Openshift Virtual Machines

PROFILE OSMASTR
CLASS GL
MAXSTORAGE 20G
ACCOUNT OPNSHIFT
ACIGROUP OPNSHIFT
COMMAND SET VCONFIG MODE LINUX
COMMAND DEFINE CPU 00 TYPE IFL
COMMAND DEFINE CPU 01 TYPE IFL
COMMAND DEFINE CPU 02 TYPE IFL
COMMAND DEFINE CPU 03 TYPE IFL
COMMAND DEFINE STORAGE **8G** RESERVED 0M STANDBY 4G
CRYPTO APVIRT
IPL CMS
IUCV ALLOW
IUCV *IDENT RESANY GLOBAL
MACHINE ESA 4
OPTION TODENABLE APPLMON
CONSOLE 0009 3215 T OPERATOR
NICDEF 1000 TYPE QDIO LAN SYSTEM OSVSW
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
(CMS Minidisk Links)

PROFILE OSWRKR
CLASS GL
MAXSTORAGE 20G
ACCOUNT OPNSHIFT
ACIGROUP OPNSHIFT
COMMAND SET VCONFIG MODE LINUX
COMMAND DEFINE CPU 00 TYPE IFL
COMMAND DEFINE CPU 01 TYPE IFL
COMMAND DEFINE CPU 02 TYPE IFL
COMMAND DEFINE CPU 03 TYPE IFL
COMMAND DEFINE STORAGE **16G** RESERVED 0M STANDBY 4G
CRYPTO APVIRT
IPL CMS
IUCV ALLOW
IUCV *IDENT RESANY GLOBAL
MACHINE ESA 4
OPTION TODENABLE APPLMON
CONSOLE 0009 3215 T OPERATOR
NICDEF 1000 TYPE QDIO LAN SYSTEM OSVSW
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
(CMS Minidisk Links)

Four Types of Virtual Machines Defined

- Network Services

  USER OSDHCPD LBYONLY

  - IP Address = 10.96.64.193

  INCLUDE OSMASTR
  MACHINE ESA 1
  DEFINE STORAGE 2G
  (attach FCP channel)

- Bootstrap Node

  USER OSBOOT LBYONLY

  - IP Address = 10.96.64.194

  INCLUDE OSMASTR
  (attach FCP channel)

- Master Node(s)

  USER OSMASTR[0-2] LBYONLY

  - IP Addresses 10.96.64.195-197

  INCLUDE OSMASTR
  (attach FCP channel)

- Worker Node(s)

  USER OSWRKR[0-1] LBYONLY

  - IP Addresses 10.96.64.198-99

  INCLUDE OSWRKR
  (attach FCP channel)

1. Define DNS/DHCP/Load Balancer/FTP Server

- Built based on SLES15 SP1
  - Easier to setup and configure via YaST than RedHat
  - Enabled DNS with 1 new zone:
    - osdemo.lrsinc.org (10.96.64.192/26)
      - Using existing network 10.96.64.0/24 with gateway 10.96.64.254
  - Add required DNS entries to appropriate zone
  - Using nginx for load balancer
- Built alternate server based on RHEL 8.1
  - Same DNS config as on SLES
  - Using haproxy instead of nginx
  - Didn't configure DHCP server
- Download Openshift code
  - Include openshift-install tar file
  - Include pull-secret
  - Make available via anonymous FTP via bind-mount
- Define user "core" with password "corepass"
  - Defined ssh key for this user for copying "public" key to ignition file
  - Command used:

    ssh-keygen -t rsa -b 4096 -N ' ' -f /home/core/.ssh/id_rsa

2. Create installation ignition files

- Untar openshell-install in the FTP server
  - Executable and readme files
- Verify the pull-secret file
  - Go to a website like *jsonlint.com* to verify the JSON in this file.
  - Verify you aren't missing any delimiters or curly braces (easily missed in a cut-and-paste)
- Create an "install-config.yaml" file
  - Position sensitive
  - Add "architecture: s390x" under computer and controlPlane tags
    - Not documented in the Redhat instructions
  - Imbed the pull-secret file
  - Imbed the public key from /home/core/.ssh/id_rsa.pub
  - Note: the YAML file will disappear after it is used to create the manifests and ignition files
    - Openshell-install program uses the term "consumed"
    - Make a copy so you can repeat the process more easily
  - Don't change the cluster network and service network parameters

2. Create installation ignition files

```
apiVersion: v1
baseDomain: osdemo.lrsinc.org
compute:
- architecture: s390x
  hyperthreading: Enabled
  name: worker
  replicas: 0
controlPlane:
  architecture: s390x
  hyperthreading: Enabled
  name: master
  replicas: 3
metadata:
  name: democluster
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  networkType: OpenShiftSDN
  serviceNetwork:
  - 172.30.0.0/16
platform:
  none: {}
fips: false
pullSecret: 'contents of pull-secret file'
sshkey: 'contents of /home/core/.ssh/id_rsa.pub'
```

2. Create installation ignition files

- Execute *openshift-install create manifests --dir=(installation_directory)*
- Change manifests/cluster-scheduler-02-config.yml file
  - Set **mastersSchedulable** to False
- Execute *openshift-install create ignition-configs --dir=(installation_directory)*
  - 3 ignition files created: bootstrap.ign, master.ign, worker.ign
- Enable access to files via anonymous ftp
  - enable world-readable for created files
    - chmod +r *.ign
    - chmod +r -R auth/
  - mount /mnt *mount-point* --bind
    - /srv/ftp in SLES
    - /var/ftp in RHEL

Note: security certificates used in this process expire after 24 hours

- Don’t get too distracted

3. Build “bootstrap” machine

- Determine proper parameters vs. “ignition” file
  - Parm file used to define hardware (FCP) and network, points to ignition file
  - Ignition file used to customize virtual machine
  - (Redhat doc very lacking in this area)
- Parmfile that worked (spaced out for readability):

  cio_ignore=all,!condev rd.neednet=1 coreos.inst=yes
  rd.zfcp=0.0.2000,0x500507680b218626,0x0000000000000000
  coreos.inst.install_dev=sda
  coreos.inst.ignition_url=ftp://10.96.64.193/bootstrap.ign
  coreos.inst.image_url=ftp://10.96.64.193/rhcos-4.3.18-s390x-metal.s390x.raw.gz
  zfcp.allow_lun_scan=0
  rd.znet=qeth,0.0.1000,0.0.1001,0.0.1002,layer2=1,portno=0
  ip=10.96.64.194::10.96.64.254:255.255.255.128:bootstrap.osdemo.lrsinc.org::none
  nameserver=10.96.64.193
- Collapsed line-ends to fit into 480-character limit for kernel command line
- Use of a separate CONFIG file to hold parameters in RHEL doesn’t work

4. Build “master” machines

- Unique Parmfile for each machine:

  cio_ignore=all,!condev rd.neednet=1 coreos.inst=yes
  rd.zfcp=0.0.2000,0x500507680b218627,0x0000000000000000
  coreos.inst.install_dev=sda
  coreos.inst.ignition_url=ftp://10.96.64.193/master.ign
  coreos.inst.image_url=ftp://10.96.64.193/rhcos-4.3.18-s390x-metal.s390x.raw.gz
  zfcp.allow_lun_scan=0
  rd.znet=qeth,0.0.1000,0.0.1001,0.0.1002,layer2=1,portno=0
  ip=10.96.64.195::10.96.64.254:255.255.255.128:master0.osdemo.lrsinc.org::none
  nameserver=10.96.64.193

- IP address is different for each master, other parameters stay the same
  - Real FCP subchannels are unique for each v.m. but are defined with the same virtual subchannels for ease of definition

5. Build “worker” machines

- Unique Parmfile for each machine:

  cio_ignore=all,!condev rd.neednet=1 coreos.inst=yes
  rd.zfcp=0.0.2000,0x500507680b218626,0x0000000000000000
  coreos.inst.install_dev=sda
  coreos.inst.ignition_url=ftp://10.96.64.193/worker.ign
  coreos.inst.image_url=ftp://10.96.64.193/rhcos-4.3.18-s390x-metal.s390x.raw.gz
  zfcp.allow_lun_scan=0
  rd.znet=qeth,0.0.1000,0.0.1001,0.0.1002,layer2=1,portno=0
  ip=10.96.64.198::10.96.64.254:255.255.255.128:worker0.osdemo.lrsinc.org::none
  nameserver=10.96.64.193

- IP address is different for each worker, other parameters stay the same
  - Real FCP subchannels are unique for each v.m. but are defined with the same virtual subchannels for ease of definition

Progress:

On the installation server (which is the same as the DNS/DHCP/Load Balancer):

osdhcpd:/mnt # **./openshift-install create manifests --dir=/mnt**
INFO Consuming "Install Config" from target directory
WARNING Making control-plane schedulable by setting MastersSchedulable to true for Scheduler cluster settings
osdhcpd:/mnt # **vi manifests/cluster-scheduler-02-config.yml**
osdhcpd:/mnt # **./openshift-install create ignition-configs --dir=/mnt**
INFO Consuming "Common Manifests" from target directory
INFO Consuming "Openshift Manifests" from target directory
INFO Consuming "Master Machines" from target directory
INFO Consuming "Worker Machines" from target directory

Started building all servers (XAUTOLOG OSBOOT, watch for initial install to complete (via SCIF), then XAUTOLOG OSMASTRx and OSWRKRy

ssh to osboot from userid *core* and watch kubectl.service:
journalctl -b -f -u bootkube.service
May 01 23:03:41 bootstrap.osdemo.lrsinc.org bootkube.sh[1783]: Waiting for etcd cluster... (retries every 5 seconds)
May 01 23:11:01 bootstrap.osdemo.lrsinc.org bootkube.sh[1783]: https://etcd-1.democluster.osdemo.lrsinc.org:2379 is healthy: successfully committed proposal: took = 1.263431ms
May 01 23:12:09 bootstrap.osdemo.lrsinc.org bootkube.sh[1783]: https://etcd-2.democluster.osdemo.lrsinc.org:2379 is healthy: successfully committed proposal: took = 1.069063ms
May 01 23:12:39 bootstrap.osdemo.lrsinc.org bootkube.sh[1783]: https://etcd-0.democluster.osdemo.lrsinc.org:2379 is healthy: successfully committed proposal: took = 1.251888ms
....
May 01 23:17:47 bootstrap.osdemo.lrsinc.org bootkube.sh[1783]: bootkube.service complete

Meanwhile, back on the installation server:

osdhcpd:/mnt # **./openshift-install wait-for bootstrap-complete --dir=/mnt**
INFO Waiting up to 30m0s for the Kubernetes API at https://api.democluster.osdemo.lrsinc.org:6443...
INFO API v1.16.2 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO It is now safe to remove the bootstrap resources

Now shutdown OSBOOT and update haproxy to remove osboot from port forwarding

Progress:

Now check on status with the ***oc*** command:

[root@osdhcpd mnt]# **export KUBECONFIG=/mnt/auth/kubeconfig**

[root@osdhcpd mnt]# **oc status**

```
In project default on server https://api.democluster.osdemo.lrsinc.org:6443

svc/openshift - kubernetes.default.svc.cluster.local
svc/kubernetes - 172.30.0.1:443 -> 6443

View details with 'oc describe <resource>/<name>' or list everything with 'oc get all'.
```

```
[root@osdhcpd mnt]# oc get all

NAME                 TYPE           CLUSTER-IP   EXTERNAL-IP                           PORT(S)   AGE
service/kubernetes   ClusterIP      172.30.0.1   <none>                                443/TCP   24m
service/openshift    ExternalName   <none>       kubernetes.default.svc.cluster.local   <none>    17m
```

```
[root@osdhcpd mnt]# oc whoami
system:admin
```

```
[root@osdhcpd mnt]# oc get csr
NAME        AGE   REQUESTOR                                                                   CONDITION
csr-4kc4w   55m   system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Approved,Issued
csr-68df2   54m   system:node:master2.osdemo.lrsinc.org                                       Approved,Issued
csr-9rvb7   55m   system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Approved,Issued
csr-bkr9l   55m   system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Approved,Issued
csr-fwfnp   54m   system:node:master1.osdemo.lrsinc.org                                       Approved,Issued
csr-kpwn9   55m   system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Approved,Issued
csr-l5n9h   54m   system:node:master0.osdemo.lrsinc.org                                       Approved,Issued
csr-pcm6w   55m   system:node:worker0.osdemo.lrsinc.org                                       Approved,Issued
csr-phbwm   54m   system:node:worker1.osdemo.lrsinc.org                                       Approved,Issued
csr-psnvr   55m   system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Approved,Issued
```

```
[root@osdhcpd mnt]# oc get nodes
NAME                        STATUS   ROLES    AGE   VERSION
master0.osdemo.lrsinc.org   Ready    master   55m   v1.16.2
master1.osdemo.lrsinc.org   Ready    master   55m   v1.16.2
master2.osdemo.lrsinc.org   Ready    master   55m   v1.16.2
worker0.osdemo.lrsinc.org   Ready    worker   55m   v1.16.2
worker1.osdemo.lrsinc.org   Ready    worker   55m   v1.16.2
```

Progress:

```
[root@osdhcpd mnt]# watch -n5 oc get clusteroperators

Every 5.0s: oc get clusteroperators
osdhcpd.osdemo.lrsinc.org: Thu May 14 10:43:00 2020

NAME                                       VERSION   AVAILABLE   PROGRESSING   DEGRADED   SINCE
authentication                             4.3.18    True        False         False      45m
cloud-credential                           4.3.18    True        False         False      59m
cluster-autoscaler                         4.3.18    True        False         False      52m
console                                    4.3.18    True        False         False      47m
dns                                        4.3.18    True        False         False      55m
image-registry                             4.3.18    True        False         False      52m
ingress                                    4.3.18    True        False         False      52m
insights                                   4.3.18    True        False         False      53m
kube-apiserver                             4.3.18    True        False         False      54m
kube-controller-manager                    4.3.18    True        False         False      54m
kube-scheduler                             4.3.18    True        False         False      52m
machine-api                                4.3.18    True        False         False      56m
machine-config                             4.3.18    True        False         False      55m
marketplace                                4.3.18    True        False         False      51m
monitoring                                 4.3.18    True        False         False      46m
network                                    4.3.18    True        False         False      55m
node-tuning                                4.3.18    True        False         False      44m
openshift-apiserver                        4.3.18    True        False         False      52m
openshift-controller-manager               4.3.18    True        False         False      55m
openshift-samples                          4.3.18    True        False         False      52m
operator-lifecycle-manager                 4.3.18    True        False         False      52m
operator-lifecycle-manager-catalog         4.3.18    True        False         False      52m
operator-lifecycle-manager-packageserver   4.3.18    True        False         False      51m
service-ca                                 4.3.18    True        False         False      57m
service-catalog-apiserver                  4.3.18    True        False         False      52m
service-catalog-controller-manager         4.3.18    True        False         False      52m
storage                                    4.3.18    True        False         False      53m
```

Progress:

```
[root@osdhcpd mnt]# ./openshift-install wait-for install-complete --dir=/mnt
INFO Waiting up to 30m0s for the cluster at https://api.democluster.osdemo.lrsinc.org:6443 to initialize...
INFO Waiting up to 10m0s for the openshift-console route to be created...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'exportKUBECONFIG=/mnt/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-console.apps.democluster.osdemo.lrsinc.org
INFO Login to the console with user: kubeadmin, password: cqNRu-BHnTH-2e7t2-e7Lpb
[root@osdhcpd mnt]#
```

- Was now able to use my browser
  - Had to add the Openshift DNS server address to my Windows list of DNS servers.

Lessons Learned

- Couldn’t login as “core” after bootstrap machine created
  - Had to start ssh-agent from user “core” and load that agent every time I rebooted the DDNS/DHCP server:

    eval "$(ssh-agent -s)“

    ssh-add *path/filename of private key*
  - Connected to bootstrap machine and watched progress via command journalctl -b -f -u bootkube.service
    - Started to build master and worker machines once bootkube service settled down
    - My clue was to look for “SELinux: mount invalid” messages on the boot server after reboot
- To handle anonymous ftp, bind mount /mnt to /var/ftp on RHEL or /srv/ftp on SLES
- Had to change *.ign files to permissions 644 (added world-readable for anonymous ftp)
- Had to allow world-readable auth subdirectory to let userid “core” get the credentials

Lessons Learned

- CNAMEs didn't always work
  - Had to give explicit assignment to etcd-[0-2] instead of using CNAME
  - Gave explicit assignment to master[0-2], worker[0-1], CNAME to actual v.m. name
- Had many messages from SELinux:
  - SELinux: mount invalid. Same superblock, different security settings for (dev mqueue, type mqueue)
  - Not a problem, just an annoyance
- Using SCSI-disk means that IND USER won't show I/O counts
  - Harder to tell if you're stuck; had to use SCIF to monitor along with watching bootkube.service (which goes in fits and starts)
- SLES vs RHEL for DNS/Load balancer/Load source
  - haproxy (RHEL) vs. nginx (SLES), nginx didn't like to simply listen on ports 80/443
  - YaST DNS-Server dialog doesn't like "*" in the dns config file; had to manually edit the osdemo.lrsinc.org file
  - Ended up staying with RHEL due to issues with nginx for http/https
    - Probably could have gotten it to work with more knowledge of nginx

Results

- Cluster was successfully built
- Virtual machines could be relocated between SSI members
- Graphical interface worked (once I pointed to the new DNS in my Windows network settings)
- Someone else gets to figure out what to do with it

Questions?

Additional File Settings

/var/named/dynamic/osdemo.lrsinc.org

```
$TTL 2d
@               IN SOA          osdhcpd.osdemo.lrsinc.org.      root.osdhcpd.osdemo.lrsinc.org. (
                                2020030101      ; serial
                                3h              ; refresh
                                1h              ; retry
                                1w              ; expiry
                                1d )            ; minimum

osdemo.lrsinc.org.      IN NS   osdhcpd.osdemo.lrsinc.org.
$ORIGIN osdemo.lrsinc.org.
osdhcpd                 IN A    10.96.64.193
bootstrap               IN A    10.96.64.194
master0                 IN A    10.96.64.195
master1                 IN A    10.96.64.196
master2                 IN A    10.96.64.197
worker0                 IN A    10.96.64.198
worker1                 IN A    10.96.64.199
etcd-0.democluster      IN A    10.96.64.195
etcd-1.democluster      IN A    10.96.64.196
etcd-2.democluster      IN A    10.96.64.197
api.democluster         IN CNAME osdhcpd.osdemo.lrsinc.org.
api-int.democluster     IN CNAME osdhcpd.osdemo.lrsinc.org.
*.apps.democluster      IN CNAME osdhcpd.osdemo.lrsinc.org.
osboot                  IN CNAME bootstrap.osdemo.lrsinc.org.
osmastr0                IN CNAME master0.osdemo.lrsinc.org.
osmastr1                IN CNAME master1.osdemo.lrsinc.org.
osmastr2                IN CNAME master2.osdemo.lrsinc.org.
oswrkr0                 IN CNAME worker0.osdemo.lrsinc.org.
oswrkr1                 IN CNAME worker1.osdemo.lrsinc.org.
_etcd-server-ssl._tcp.democluster 86400 IN SRV 0 10 2380 etcd-0.democluster.osdemo.lrsinc.org.
_etcd-server-ssl._tcp.democluster 86400 IN SRV 0 10 2380 etcd-1.democluster.osdemo.lrsinc.org.
_etcd-server-ssl._tcp.democluster 86400 IN SRV 0 10 2380 etcd-2.democluster.osdemo.lrsinc.org.
```

Additional File Settings

/var/named/dynamic/64.96.10.in-addr.arpa

```
$TTL 2d
@               IN SOA          osdhcpd.osdemo.lrsinc.org.      root.osdhcpd.osdemo.lrsinc.org. (
                                2020041400      ; serial
                                3h              ; refresh
                                1h              ; retry
                                1w              ; expiry
                                1d )            ; minimum

64.96.10.in-addr.arpa.  IN NS           osdhcpd.osdemo.lrsinc.org.
$ORIGIN 64.96.10.in-addr.arpa.
193             IN PTR          osdhcpd.osdemo.lrsinc.org.
194             IN PTR          bootstrap.osdemo.lrsinc.org.
195             IN PTR          master0.osdemo.lrsinc.org.
196             IN PTR          master1.osdemo.lrsinc.org.
197             IN PTR          master2.osdemo.lrsinc.org.
198             IN PTR          worker0.osdemo.lrsinc.org.
199             IN PTR          worker1.osdemo.lrsinc.org.
195             IN PTR          etcd-0.democluster.osdemo.lrsinc.org.
196             IN PTR          etcd-1.democluster.osdemo.lrsinc.org.
197             IN PTR          etcd-2.democluster.osdemo.lrsinc.org.
```

Additional File Settings

/etc/vsftpd/vsftpd.conf

```
anonymous_enable=YES
local_enable=YES
write_enable=YES
local_umask=022
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
ftpd_banner=Welcome to osdhcpd FTP service
listen=NO
listen_ipv6=YES
pam_service_name=vsftpd
userlist_enable=NO
```

Additional File Settings

/etc/haproxy/haproxy.conf

```
global
 log         127.0.0.1 local2
 chroot      /var/lib/haproxy
 pidfile     /var/run/haproxy.pid
 maxconn     4000
 user        haproxy
 group       haproxy
 daemon
 stats socket /var/lib/haproxy/stats
 ssl-default-bind-ciphers PROFILE=SYSTEM
 ssl-default-server-ciphers PROFILE=SYSTEM
defaults
    mode                    http
    log                     global
    option                  tcplog
    option                  dontlognull
    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

listen ingress-http
    bind *:80
    mode tcp
    server worker0 10.96.64.198:80 check
    server worker1 10.96.64.199:80 check

listen ingress-https
    bind *:443
    mode tcp
    server worker0 10.96.64.198:443 check
    server worker1 10.96.64.199:443 check

listen api
    bind *:6443
    mode tcp
#   server bootstrap 10.96.64.194:6443 check
    server master0 10.96.64.195:6443 check
    server master1 10.96.64.196:6443 check
    server master2 10.96.64.197:6443 check

listen api-int
    bind *:22623
    mode tcp
#   server bootstrap 10.96.64.194:22623 check
    server master0 10.96.64.195:22623 check
    server master1 10.96.64.196:22623 check
    server master2 10.96.64.197:22623 check
```