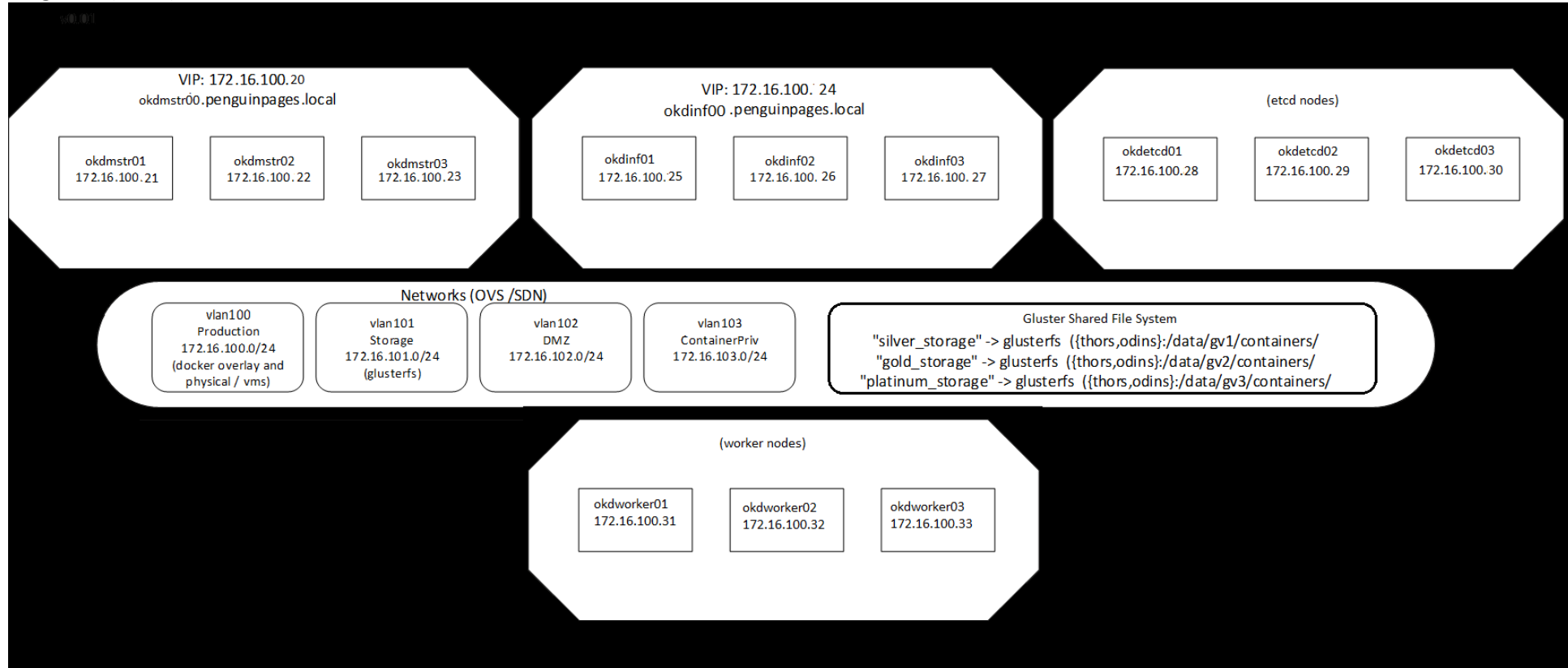


Goal: Build a NSPOF cluster with three masters, three infrastructure/proxy and two workers then later add two more workers (which have gpu cards) and add tiers for storage.

Diagram: (v0.02)



Documentation:

Video Guide: (has some issues toward end lacking updated detail as some was combined into scripts) [Openshift Origin \(3.11\) Installation On Centos7](#)

Command examples: <https://subhakarkotta.github.io/okd-installation-centos/>

Step 1: All nodes install base packages and OS settings

Assumption is you have deployed cluster nodes with base CentOS and all nodes have shared ssh keys and common /etc/hosts file and selinux enabled.

Install Packages for deployment on all nodes

```
yum update -y
yum install wget envsubst figlet git zile nano net-tools docker bind-utils iptables-services bridge-utils bash-completion kexec-tools sos psacct openssl-devel
httpd-tools NetworkManager python-cryptography python2-pip python-devel python-passlib java-1.8.0-openjdk-headless "@Development Tools" epel-release
-y
```

Validate NetworkManager running

```
systemctl enable NetworkManager
systemctl start NetworkManager
```

Create origin user

```
useradd origin
usermod -aG wheel origin
echo -e "password\npassword" | passwd origin
```

Not sure if this needed but they not to disable EPEL repository

```
sed -i -e "s/^enabled=1/enabled=0/" /etc/yum.repos.d/epel.repo
```

Step 2: Prep work on first master

Ex: kmstr01

```
yum -y --enablerepo=epel install ansible pyOpenSSL
curl -o ansible.rpm
```

Modify cluster configuration file.. NOTE: no etcd nodes defined for this example!!!!

```
cd /
git clone https://github.com/SubhakarKotta/okd-installation-centos.git
vi okd-installation-centos/provisioning/settings.sh
```

```
#!/bin/bash
```

```
#The below configuration can be edited up on your needs and and please note the it's just an example configuration.
```

#We are going to create an OKD cluster with one master and 3 worker nodes.

#OKD Version

export OKD_VERSION=3.11

#OKD Master Node 1 Configuration

export OKD_MASTER_1_IP=172.16.100.21

export OKD_MASTER_1_HOSTNAME=okdmstr01

#OKD Master Node 2 Configuration

export OKD_MASTER_2_IP=172.16.100.22

export OKD_MASTER_2_HOSTNAME=okdmstr02

#OKD Master Node 3 Configuration

export OKD_MASTER_3_IP=172.16.100.23

export OKD_MASTER_3_HOSTNAME=okdmstr03

#OKD Worker Node 1 Configuration

export OKD_WORKER_NODE_1_IP=172.16.101.31

export OKD_WORKER_NODE_1_HOSTNAME=okdworker01

#OKD Worker Node 2 Configuration

export OKD_WORKER_NODE_2_IP=172.16.101.32

export OKD_WORKER_NODE_2_HOSTNAME=okdworker02

#OKD Worker Node 3 Configuration

export OKD_WORKER_NODE_3_IP=172.16.101.33

export OKD_WORKER_NODE_3_HOSTNAME=okdworker03

#OKD Infra Node 1 Configuration

export OKD_INFRA_NODE_1_IP=172.16.100.25

export OKD_INFRA_NODE_1_HOSTNAME=okdinf01

#OKD Infra Node 2 Configuration

export OKD_INFRA_NODE_2_IP=172.16.100.26

export OKD_INFRA_NODE_2_HOSTNAME=okdinf02

#OKD Infra Node 3 Configuration

export OKD_INFRA_NODE_3_IP=172.16.100.27

```
export OKD_INFRA_NODE_3_HOSTNAME=okdinf03

#The below setting will be used to access OKD console https://console.\$DOMAIN:\$API\_PORT"
#By default we can login using the URL https://console.okd.nip.io:8443
#To access URL from your local system we need to configure master host in C:\Windows\System32\drivers\etc\hosts file as below
#100.10.10.100 console.okd.nip.io
#export DOMAIN=okd.nip.io
export DOMAIN=okd.penguinpages.local
export API_PORT=8443

#OKD Login Credentials
#By default admin/admin operator will be created and can be used to login to OKD console.
export OKD_USERNAME=admin
export OKD_PASSWORD=admin

#OKD Add-Ons
#Enable "True" only if one of the VM has 4GB memory.
export INSTALL_METRICS=False

# Enable "True" only one of the VM 16GB memory.
export INSTALL_LOGGING=False

chmod +x -R /okd-installation-centos
# Copy directory to every node in the cluster
for vm in $(cat /media/sw2_usb_A2/penguinpages_local_cluster/okd_cluster_nodes); do ; scp -r /okd-installation-centos/ root@$vm:/ ; done
```

Disable epel repo on nodes

```
for vm in $(cat /media/sw2_usb_A2/penguinpages_local_cluster/okd_cluster_nodes); do sed -i -e "s/^enabled=1/enabled=0/" /etc/yum.repos.d/epel.repo; done
```

On all nodes

```
yum -y --enablerepo=epel install ansible pyOpenSSL
```

Note: do not need to run requeue script on nodes as all it does is populate hosts files and install packages and steps done above

Ex: All nodes share sync copy of /etc/hosts

```
[root@okdmstr01 provisioning]# cat /etc/hosts
# Version: 20190730a
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

# Network Devices
172.16.100.1 sw2.penguinpages.local sw2
172.16.100.253 sw0.penguinpages.local sw0

# Cluster VM for OKD
172.16.100.21 okdmstr01.penguinpages.local okdmstr01
172.16.101.21 okdmstr01s.penguinpages.local okdmstr01s
172.16.100.20 okdmstr00.penguinpages.local okdmstr00
172.16.100.22 okdmstr02.penguinpages.local okdmstr02
172.16.101.22 okdmstr02s.penguinpages.local okdmstr02s
172.16.100.23 okdmstr03.penguinpages.local okdmstr03
172.16.101.23 okdmstr03s.penguinpages.local okdmstr03s
172.16.100.25 okdinf01.penguinpages.local okdinf01
172.16.101.25 okdinf01s.penguinpages.local okdinf01s
172.16.100.24 okdinf00.penguinpages.local okdinf00
172.16.100.26 okdinf02.penguinpages.local okdinf02
172.16.101.26 okdinf02s.penguinpages.local okdinf02s
172.16.100.27 okdinf03.penguinpages.local okdinf03
172.16.101.27 okdinf03s.penguinpages.local okdinf03s
172.16.100.28 okdetcd01.penguinpages.local okdetcd01
172.16.101.28 okdetcd01s.penguinpages.local okdetcd01s
172.16.100.29 okdetcd02.penguinpages.local okdetcd02
172.16.101.29 okdetcd02s.penguinpages.local okdetcd02s
172.16.100.30 okdetcd03.penguinpages.local okdetcd03
172.16.101.30 okdetcd03s.penguinpages.local okdetcd03s
172.16.100.31 okdworker01.penguinpages.local okdworker01
172.16.101.31 okdworker01s.penguinpages.local okdworker01s
172.16.100.32 okdworker02.penguinpages.local okdworker02
172.16.101.32 okdworker02s.penguinpages.local okdworker02s
172.16.100.33 okdworker03.penguinpages.local okdworker03
172.16.101.33 okdworker03s.penguinpages.local okdworker03s
```

Step 3: Run installation on Master node

Ex: okdmstr01

/okd-installation-centos/provisioning/install_master.sh

Shell output logs

```
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: reflector.westga.edu
* epel: reflector.westga.edu
* extras: reflector.westga.edu
* updates: reflector.westga.edu
Package ansible-2.8.4-1.el7.noarch already installed and latest version
Package pyOpenSSL-0.13.1-4.el7.x86_64 already installed and latest version
Nothing to do
% Total % Received % Xferd Average Speed Time Time Time Current
          Dload Upload Total Spent Left Speed
100 9.9M 100 9.9M 0 0 10.4M 0 --:--:-- --:--:-- --:--:-- 10.4M
Loaded plugins: fastestmirror, langpacks
Examining ansible.rpm: ansible-2.6.5-1.el7.ans.noarch
ansible.rpm: does not update installed package.
Error: Nothing to do
Cloning into 'openshift-ansible'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 143511 (delta 0), reused 6 (delta 0), pack-reused 143503
Receiving objects: 100% (143511/143511), 39.31 MiB | 19.78 MiB/s, done.
Resolving deltas: 100% (90094/90094), done.
Branch release-3.11 set up to track remote branch release-3.11 from origin.
Switched to a new branch 'release-3.11'
```

PLAY [Fail openshift_kubelet_name_override for new hosts]

```
*****
*****
```

TASK [Gathering Facts]

```
*****
*****
```

```
fatal: [openshift_ip=]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: ssh: Could not resolve hostname openshift_ip=:
Name or service not known", "unreachable": true}
ok: [172.16.100.25]
fatal: [172.16.101.31]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: ssh: connect to host 172.16.101.31 port 22:
Connection timed out", "unreachable": true}
fatal: [172.16.101.33]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: ssh: connect to host 172.16.101.33 port 22:
Connection timed out", "unreachable": true}
fatal: [172.16.101.32]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: ssh: connect to host 172.16.101.32 port 22:
Connection timed out", "unreachable": true}
```

TASK [Fail when openshift_kubelet_name_override is defined]

```
*****
*****
```

skipping: [172.16.100.25]

PLAY [Initialization Checkpoint Start]

```
*****
*****
```

TASK [Set install initialization 'In Progress']

```
*****
*****
```

ok: [172.16.100.25]

PLAY [Populate config host groups]

```
*****
*****
```

TASK [Load group name mapping variables]

```
*****
*****
```

ok: [localhost]

TASK [Evaluate groups - g_nfs_hosts is single host]

```
*****
*****
```

skipping: [localhost]

TASK [Evaluate oo_all_hosts]

ok: [localhost] => (item=openshift_ip=)
ok: [localhost] => (item=172.16.101.31)
ok: [localhost] => (item=172.16.101.32)
ok: [localhost] => (item=172.16.101.33)
ok: [localhost] => (item=172.16.100.25)

TASK [Evaluate oo_masters]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_first_master]

ok: [localhost]

TASK [Evaluate oo_new_etcd_to_config]

TASK [Evaluate oo_masters_to_config]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_etcd_to_config]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_first_etcd]

ok: [localhost]

TASK [Evaluate oo_etcd_hosts_to_upgrade]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_etcd_hosts_to_backup]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_nodes_to_config]

ok: [localhost] => (item=openshift_ip=)

ok: [localhost] => (item=172.16.101.31)

ok: [localhost] => (item=172.16.101.32)

ok: [localhost] => (item=172.16.101.33)

ok: [localhost] => (item=172.16.100.25)

TASK [Evaluate oo_lb_to_config]

TASK [Evaluate oo_nfs_to_config]

TASK [Evaluate oo_glusterfs_to_config]

TASK [Evaluate oo_etcd_to_migrate]

ok: [localhost] => (item=openshift_ip=)

[WARNING]: Could not match supplied host pattern, ignoring: oo_lb_to_config

[WARNING]: Could not match supplied host pattern, ignoring: oo_nfs_to_config

PLAY [Ensure that all non-node hosts are accessible]

PLAY [Initialize basic host facts]

TASK [Gathering Facts]

ok: [172.16.100.25]

TASK [openshift_sanitize_inventory : include_tasks]

included: /okd-installation-centos/provisioning/openshift-ansible/roles/openshift_sanitize_inventory/tasks/deprecations.yml for 172.16.100.25

TASK [openshift_sanitize_inventory : Check for usage of deprecated variables]

ok: [172.16.100.25]

TASK [openshift_sanitize_inventory : debug]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : set_stats]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : set_fact]

ok: [172.16.100.25]

TASK [openshift_sanitize_inventory : Standardize on latest variable names]

ok: [172.16.100.25]

TASK [openshift_sanitize_inventory : Normalize openshift_release]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Abort when openshift_release is invalid]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : include_tasks]

included: /okd-installation-centos/provisioning/openshift-ansible/roles/openshift_sanitize_inventory/tasks/unsupported.yml for 172.16.100.25

TASK [openshift_sanitize_inventory : set_fact]

TASK [openshift_sanitize_inventory : Ensure that dynamic provisioning is set if using dynamic storage]

**

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure the hosted registry's GlusterFS storage is configured correctly]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure the hosted registry's GlusterFS storage is configured correctly]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Check for deprecated prometheus/grafana install]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure clusterid is set along with the cloudprovider]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure ansible_service_broker_remove and ansible_service_broker_install are mutually exclusive]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure template_service_broker_remove and template_service_broker_install are mutually exclusive]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure that all requires vsphere configuration variables are set]

**

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : ensure provider configuration variables are defined]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure removed web console extension variables are not set]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : Ensure that web console port matches API server port]

skipping: [172.16.100.25]

TASK [openshift_sanitize_inventory : At least one master is schedulable]

skipping: [172.16.100.25]

TASK [Detecting Operating System from ostree_booted]

ok: [172.16.100.25]

TASK [set openshift_deployment_type if unset]

skipping: [172.16.100.25]

TASK [initialize_facts set fact openshift_is_atomic]

ok: [172.16.100.25]

TASK [Determine Atomic Host Docker Version]

skipping: [172.16.100.25]

TASK [assert atomic host docker version is 1.12 or later]

skipping: [172.16.100.25]

PLAY [Retrieve existing master configs and validate]

PLAY [Initialize special first-master variables]

PLAY [Disable web console if required]

PLAY [Setup yum repositories for all hosts]

TASK [rhel_subscribe : fail]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Install Red Hat Subscription manager]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Is host already registered?]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Register host using user/password]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Register host using activation key]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Determine if OpenShift Pool Already Attached]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Attach to OpenShift Pool]

skipping: [172.16.100.25]

TASK [rhel_subscribe : Satellite preparation]

skipping: [172.16.100.25]

TASK [openshift_repos : Ensure libselenium-python is installed]

ok: [172.16.100.25]

TASK [openshift_repos : Remove openshift_additional.repo file]

ok: [172.16.100.25]

TASK [openshift_repos : Create any additional repos that are defined]

changed: [172.16.100.25] => (item={'gpgcheck': '0', 'enabled': '1', 'id': 'centos-paas', 'baseurl': 'https://buildlogs.centos.org/centos/7/paas/x86_64/openshift-origin311', 'name': 'centos-paas'})

TASK [openshift_repos : include_tasks]

skipping: [172.16.100.25]

TASK [openshift_repos : include_tasks]

included: /okd-installation-centos/provisioning/openshift-ansible/roles/openshift_repos/tasks/centos_repos.yml for 172.16.100.25

TASK [openshift_repos : Configure origin gpg keys]

changed: [172.16.100.25]

TASK [openshift_repos : Configure correct origin release repository]

changed: [172.16.100.25]

TASK [openshift_repos : Ensure clean repo cache in the event repos have been changed manually]


```
changed: [172.16.100.25] => {  
  "msg": "First run of openshift_repos"  
}
```

TASK [openshift_repos : Record that openshift_repos already ran]

ok: [172.16.100.25]

[WARNING]: flush_handlers task does not support when conditional

RUNNING HANDLER [openshift_repos : refresh cache]

changed: [172.16.100.25]

PLAY [Install packages necessary for installer]

TASK [Gathering Facts]

ok: [172.16.100.25]

TASK [Determine if chrony is installed]

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

changed: [172.16.100.25]

TASK [Install ntp package]

skipping: [172.16.100.25]

TASK [Start and enable ntpd/chronyd]

changed: [172.16.100.25]

TASK [Ensure minimum kernel version]

skipping: [172.16.100.25]

TASK [Ensure openshift-ansible installer package deps are installed]

changed: [172.16.100.25]

PLAY [Initialize cluster facts]

TASK [Gathering Facts]

ok: [172.16.100.25]

TASK [get openshift_current_version]

ok: [172.16.100.25]

TASK [set_fact openshift_portal_net if present on masters]

skipping: [172.16.100.25]

TASK [Gather Cluster facts]

changed: [172.16.100.25]

TASK [Set fact of no_proxy_internal_hostnames]

skipping: [172.16.100.25]

TASK [Initialize openshift.node.sdn_mtu]

changed: [172.16.100.25]

TASK [set_fact |_kubelet_node_name]

ok: [172.16.100.25]

PLAY [Initialize etcd host variables]

PLAY [Determine openshift_version to configure on first master]

PLAY [Set openshift_version for etcd, node, and master hosts]

TASK [Gathering Facts]

ok: [172.16.100.25]

TASK [set_fact]

fatal: [172.16.100.25]: FAILED! => {"msg": "The task includes an option with an undefined variable. The error was: 'ansible.vars.hostvars.HostVarsVars object' has no attribute 'openshift_version'\n\nThe error appears to be in '/okd-installation-centos/provisioning/openshift-ansible/playbooks/init/version.yml': line 20, column 5, but may\nbe elsewhere in the file depending on the exact syntax problem.\n\nThe offending line appears to be:\n\n tasks:\n - set_fact:\n ^ here\n"}
^

PLAY [Verify Requirements]

PLAY [Verify Node Prerequisites]

PLAY [Validate Aci deployment variables]

PLAY [Initialization Checkpoint End]

PLAY [Validate node hostnames]

PLAY [Configure os_firewall]

PLAY [oo_nodes_to_config]

PLAY [oo_nodes_to_config]

PLAY RECAP

172.16.100.25	:ok=29	changed=10	unreachable=0	failed=1	skipped=34	rescued=0	ignored=0
172.16.101.31	:ok=0	changed=0	unreachable=1	failed=0	skipped=0	rescued=0	ignored=0
172.16.101.32	:ok=0	changed=0	unreachable=1	failed=0	skipped=0	rescued=0	ignored=0
172.16.101.33	:ok=0	changed=0	unreachable=1	failed=0	skipped=0	rescued=0	ignored=0
localhost	:ok=11	changed=0	unreachable=0	failed=0	skipped=5	rescued=0	ignored=0
openshift_ip=	:ok=0	changed=0	unreachable=1	failed=0	skipped=0	rescued=0	ignored=0

INSTALLER STATUS

Initialization : In Progress (0:00:41)

PLAY [Initialization Checkpoint Start]

TASK [Set install initialization 'In Progress']

ok: [openshift_ip=]

PLAY [Populate config host groups]

TASK [Load group name mapping variables]

ok: [localhost]

TASK [Evaluate groups - g_nfs_hosts is single host]

skipping: [localhost]

TASK [Evaluate oo_all_hosts]

ok: [localhost] => (item=openshift_ip=
ok: [localhost] => (item=172.16.101.31)
ok: [localhost] => (item=172.16.101.32)
ok: [localhost] => (item=172.16.101.33)
ok: [localhost] => (item=172.16.100.25)

TASK [Evaluate oo_masters]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_first_master]

ok: [localhost]

TASK [Evaluate oo_new_etcd_to_config]

TASK [Evaluate oo_masters_to_config]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_etcd_to_config]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_first_etcd]

ok: [localhost]

TASK [Evaluate oo_etcd_hosts_to_upgrade]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_etcd_hosts_to_backup]

ok: [localhost] => (item=openshift_ip=)

TASK [Evaluate oo_nodes_to_config]

ok: [localhost] => (item=openshift_ip=)

ok: [localhost] => (item=172.16.101.31)

ok: [localhost] => (item=172.16.101.32)

ok: [localhost] => (item=172.16.101.33)

ok: [localhost] => (item=172.16.100.25)

TASK [Evaluate oo_lb_to_config]

TASK [Evaluate oo_nfs_to_config]

TASK [Evaluate oo_glusterfs_to_config]

TASK [Evaluate oo_etcd_to_migrate]

ok: [localhost] => (item=openshift_ip=)

[WARNING]: Could not match supplied host pattern, ignoring: oo_lb_to_config

[WARNING]: Could not match supplied host pattern, ignoring: oo_nfs_to_config

PLAY [Ensure that all non-node hosts are accessible]

TASK [Gathering Facts]

fatal: [openshift_ip=]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: ssh: Could not resolve hostname openshift_ip=: Name or service not known", "unreachable": true}

NO MORE HOSTS LEFT

PLAY RECAP

localhost :ok=11 changed=0 unreachable=0 failed=0 skipped=5 rescued=0 ignored=0
openshift_ip= :ok=1 changed=0 unreachable=1 failed=0 skipped=0 rescued=0 ignored=0

INSTALLER STATUS

Initialization : In Progress (0:00:01)
Adding password for user admin

```
/okd-installation-centos/provisioning/install_master.sh: line 26: oc: command not found
% Total % Received % Xferd Average Speed Time Time Time Current
      Dload Upload Total Spent Left Speed
100 7034 100 7034 0 0 24040 0 --:--:-- --:--:-- --:--:-- 24089
Downloading https://get.helm.sh/helm-v2.14.3-linux-amd64.tar.gz
Preparing to install helm and tiller into /usr/local/bin
helm installed into /usr/local/bin/helm
tiller installed into /usr/local/bin/tiller
Run 'helm init' to configure helm.
/okd-installation-centos/provisioning/install_master.sh: line 34: kubectl: command not found
/okd-installation-centos/provisioning/install_master.sh: line 35: kubectl: command not found
Creating /root/.helm
Creating /root/.helm/repository
Creating /root/.helm/repository/cache
Creating /root/.helm/repository/local
Creating /root/.helm/plugins
Creating /root/.helm/starters
Creating /root/.helm/cache/archive
Creating /root/.helm/repository/repositories.yaml
Adding stable repo with URL: https://kubernetes-charts.storage.googleapis.com
Adding local repo with URL: http://127.0.0.1:8879/charts
$HELM_HOME has been configured at /root/.helm.
Error: error installing: Post http://localhost:8080/apis/extensions/v1beta1/namespaces/kube-system/deployments: dial tcp [::1]:8080: connect: connection
refused
#####
* Your console is https://console.okd.penguinpages.local:8443
* Your username is admin
* Your password is admin
*
* Login using:
*
$ oc login -u admin -p admin https://console.okd.penguinpages.local:8443/
#####
/okd-installation-centos/provisioning/install_master.sh: line 49: oc: command not found
```

Test UI is working

Local Shell Test	Windows Remote System Test
oc login -u admin -p admin https://console.okd.penguinpages.local:8443/	# Note you may need to add to windows hosts file "172.16.100.70 console.penguinpages.local console"

Step 4:

Master nodes

Step 5: Login and basic setup

Step 6: Add an additional Worker node

Step 8: Setup advanced disk with overlay2 and disk tiers

Goal is to setup different gluster file systems with

Ex: gv0 = basic containers and storage with SATA drives label "bronze_storage" , gv1 = higher and low latency storage on ssd label "silver_storage" , gv2 highest storage with low latency via NVMe "gold_storage" , gv3 = demo of RAMDisk based storage with sync setup from gv3 to gv2 at intervals to protect data "platinum_storage"