Openshift Installation On IBM Cloud Infrastructure

This document will help you to install Openshift on a VMWare based Infrastructure, It provides detailed steps you need to follow to access IBM Cloud, to provision the needed Infrastructure, and to install Openshift 3.11 on top of it.

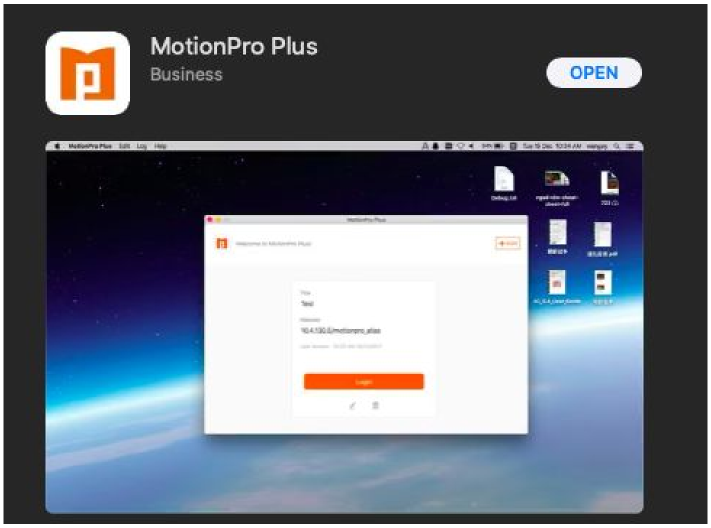
This document is divided into three main sections:
1- How to access IBM Cloud using VPN.
2- How to provision ESXi Host. and how to use it to create the required VMs.
3- How to install Openshift 3.11
    3.1- Preparing the hosts (CentOS7 VMs will be used here)
    3.2- Installing Openshift using Ansible Playbook
    3.3- Troubleshooting

Section 01- IBM Cloud Access

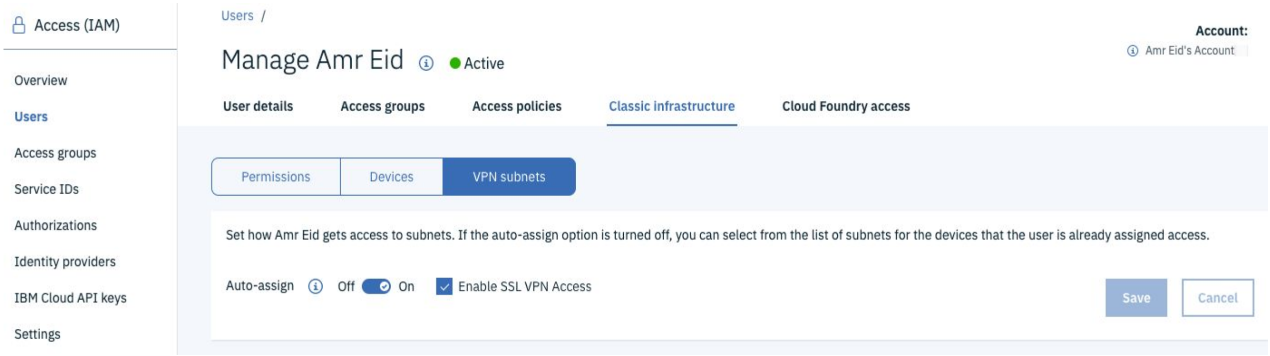
When you provision an ESXi host on top of IBM Cloud, In order to access it, you will need to setup a VPN.
You will need to establish a VPN connection to IBM Cloud private network via SSL, or IPSec.
Pre-Req:
Identify which Endpoint you will use, Which DC you will provision your BM on. Here is a full list of end points
The environment I will use in here will be based on Frankfurt , which have endpoint vpn.fra02.softlayer.com

Based on your local machine, install your VPN Client
For Windows Users
For Mac Users

From a Mac machine, install MotionPro Plus using App Store



On IBM Cloud Portal, Enable SSL VPN Access for the user and manage its password

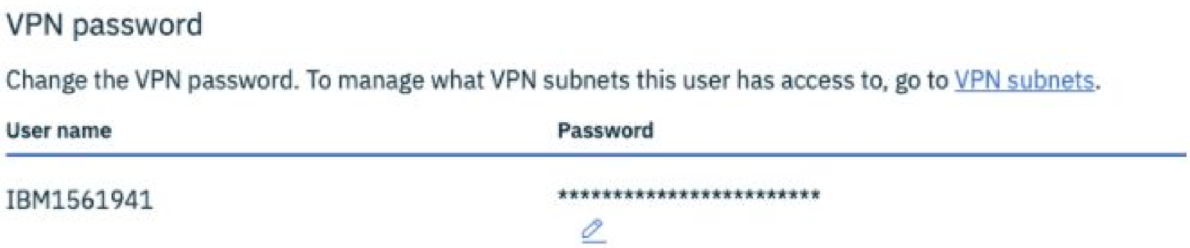


Manage Amr Eid ● Active

Set how Amr Eid gets access to subnets. If the auto-assign option is turned off, you can select from the list of subnets for the devices that the user is already assigned access.

Auto-assign Off ● On ☑ Enable SSL VPN Access

Preferably change the password. If not you need to capture the user/password from IAM/Users/User Page

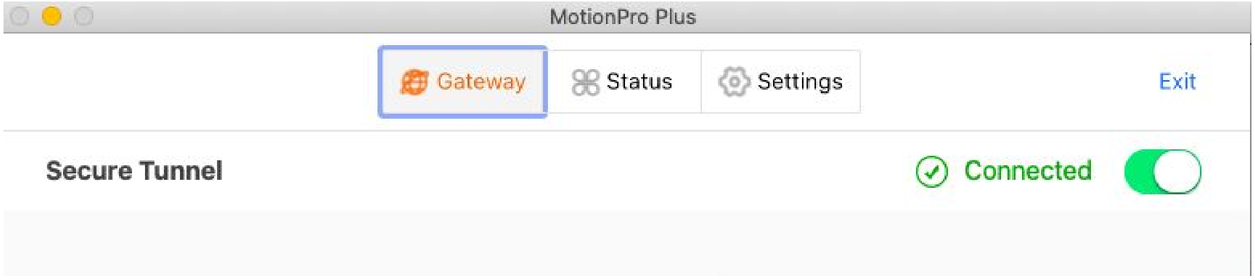


VPN password

Change the VPN password. To manage what VPN subnets this user has access to, go to VPN subnets.

User name	Password
IBM1561941	*************************

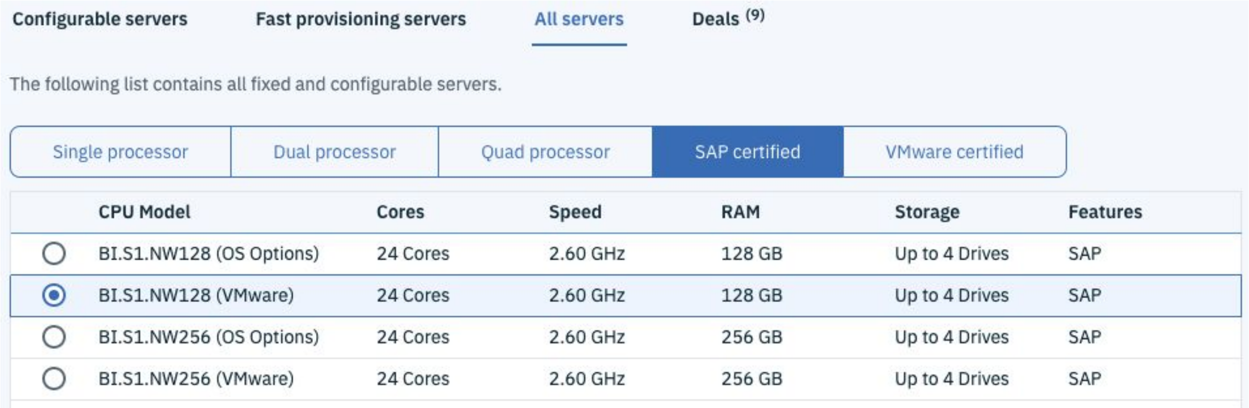
Get the Client up and running
You will need to feed MotionPro with the endpoint, In our case “vpn.fra02.softlayer.com”
You will also need to provide it with the user/password after you hit login.



Secure Tunnel ✓ Connected

Section 02- ESXi Host Provisioning

Provisioning the Bare Metal usually takes from 8 to 24 hours, so plan ahead.
Form the portal, Pick the vmWare BM version you like.



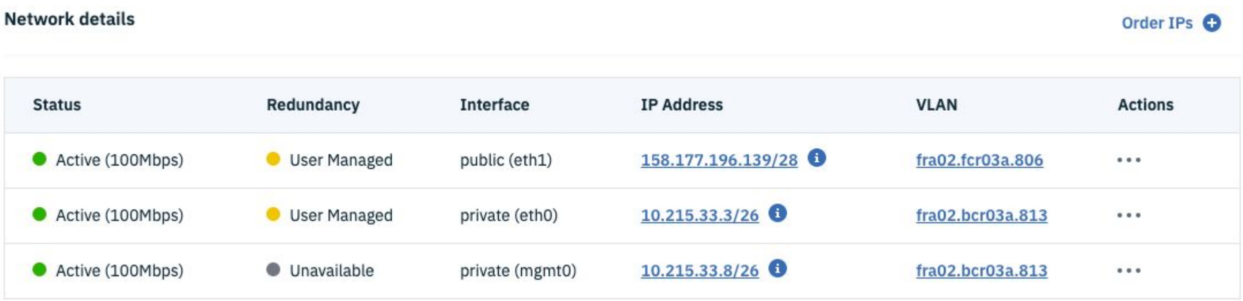
Configurable servers | Fast provisioning servers | All servers | Deals (9)

The following list contains all fixed and configurable servers.

Single processor | Dual processor | Quad processor | SAP certified | VMware certified

	CPU Model	Cores	Speed	RAM	Storage	Features
○	BI.S1.NW128 (OS Options)	24 Cores	2.60 GHz	128 GB	Up to 4 Drives	SAP
●	BI.S1.NW128 (VMware)	24 Cores	2.60 GHz	128 GB	Up to 4 Drives	SAP
○	BI.S1.NW256 (OS Options)	24 Cores	2.60 GHz	256 GB	Up to 4 Drives	SAP
○	BI.S1.NW256 (VMware)	24 Cores	2.60 GHz	256 GB	Up to 4 Drives	SAP

Check the Subnets / IP Addresses attached to the BM, Pick the private IP (usually attached to eth0), the other private IP usually is used for the management and could be accessed through KVM.

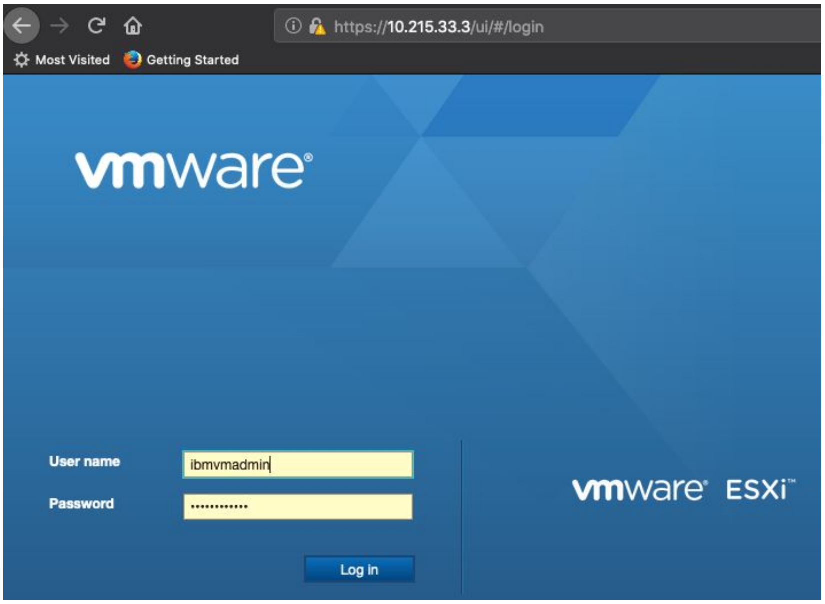


Network details

Order IPs

Status	Redundancy	Interface	IP Address	VLAN	Actions
● Active (100Mbps)	● User Managed	public (eth1)	158.177.196.139/28	fra02.fcr03a.806	•••
● Active (100Mbps)	● User Managed	private (eth0)	10.215.33.3/26	fra02.bcr03a.813	•••
● Active (100Mbps)	● Unavailable	private (mgmt0)	10.215.33.8/26	fra02.bcr03a.813	•••

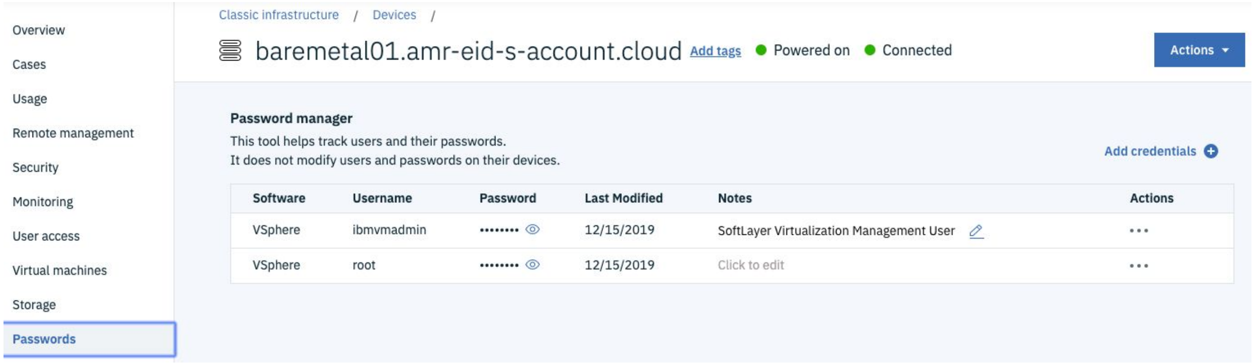
From any browser type the private IP and hit enter, then accept the default certificates you may face. You should receive the login page like below.



User name: ibmvmadmin

Password: ••••••••••••

From your BM page on IBM Cloud portal, navigate to Passwords to get the vSphere username and password. (ibmvmadmin is the ESXi user).



baremetal01.amr-eid-s-account.cloud

Password manager

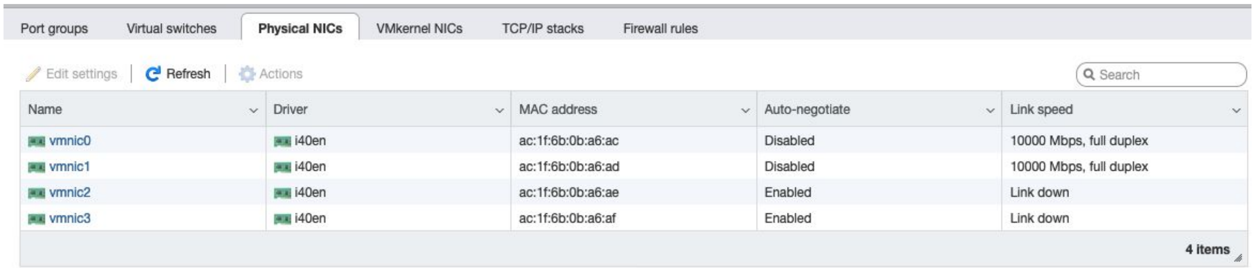
This tool helps track users and their passwords.
It does not modify users and passwords on their devices.

Software	Username	Password	Last Modified	Notes	Actions
VSphere	ibmvmadmin	••••••••	12/15/2019	SoftLayer Virtualization Management User	•••
VSphere	root	••••••••	12/15/2019	Click to edit	•••

Configuring ESXi

In the ESXi portal , You need to know

- Physical NICs on the BM are reflected on the ESXi physical NICs view.
- You should have validate that you do have an up   and running NICs like the below screenshot.



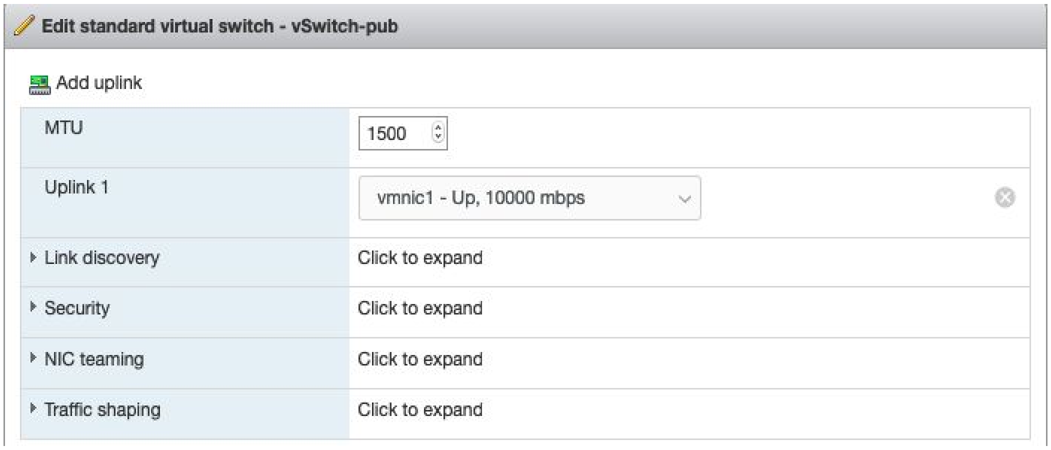
Name	Driver	MAC address	Auto-negotiate	Link speed
vmnic0	i40en	ac:1f:6b:0b:a6:ac	Disabled	10000 Mbps, full duplex
vmnic1	i40en	ac:1f:6b:0b:a6:ad	Disabled	10000 Mbps, full duplex
vmnic2	i40en	ac:1f:6b:0b:a6:ae	Enabled	Link down
vmnic3	i40en	ac:1f:6b:0b:a6:af	Enabled	Link down

4 items

Here, There are two NICs down and the other two are up and running.
From the mapping you can easily detect that NIC0 is the private NIC and NIC1 is the Public.

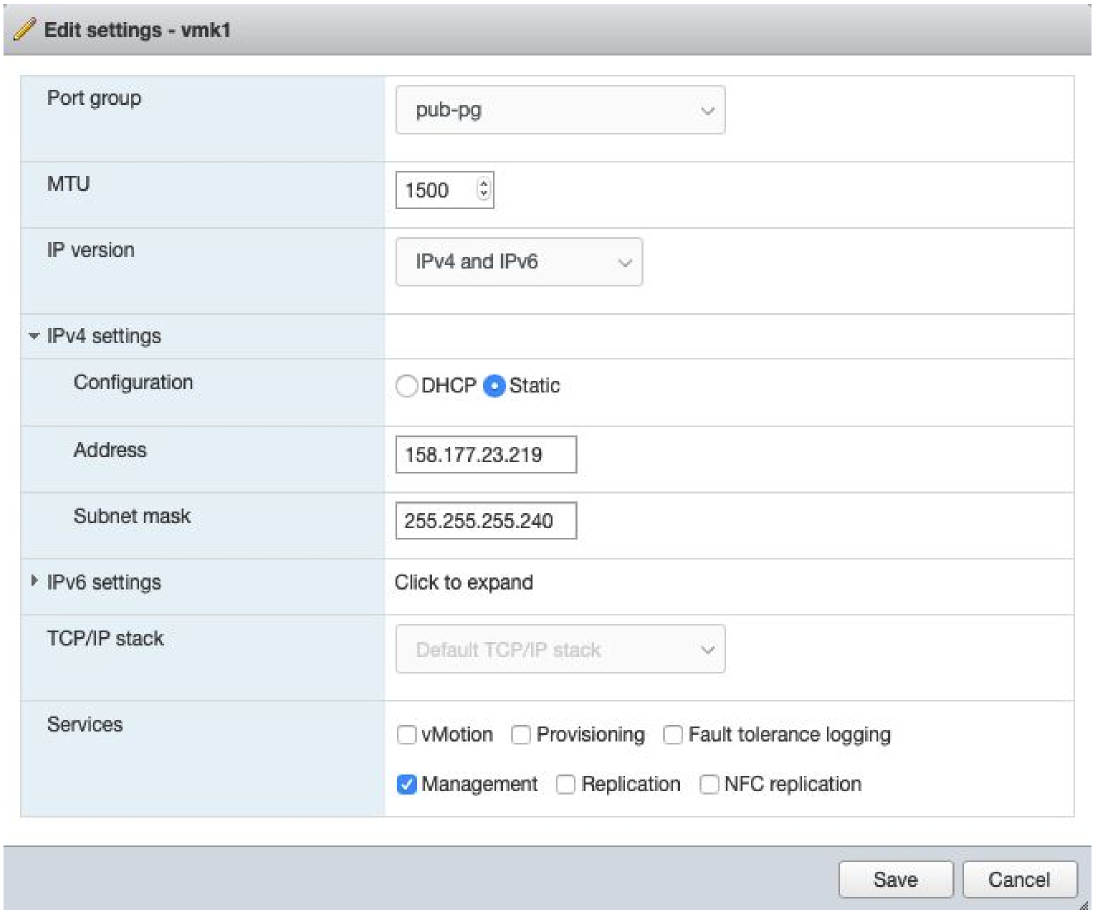
In order to proceed , you just need to do the below actions

- Create Port Groups (one for the private and one for the public.)
- link each vSwitch with its related Physical NIC.


Edit standard virtual switch - vSwitch-pub

MTU	1500
Uplink 1	vmnic1 - Up, 10000 mbps
Link discovery	Click to expand
Security	Click to expand
NIC teaming	Click to expand
Traffic shaping	Click to expand

Below screenshot shows how the public looks, attached to vmnic1 which is the public interface.
Apply the same for the private and make sure it is connected to vmnic0
VMKernel NICs will have one for private connection, you will need to create one more.



Port group	pub-pg
MTU	1500
IP version	IPv4 and IPv6
IPv4 settings	
Configuration	○ DHCP ● Static
Address	158.177.23.219
Subnet mask	255.255.255.240
IPv6 settings	Click to expand
TCP/IP stack	Default TCP/IP stack
Services	□ vMotion □ Provisioning □ Fault tolerance logging ☑ Management □ Replication □ NFC replication

Run `esxcfg-route -a 10.0.0.0/8 [your servers private gateway ip]`

```
[root@baremetal01:~] esxcfg-route -a 10.0.0.0/8 10.194.251.65
```

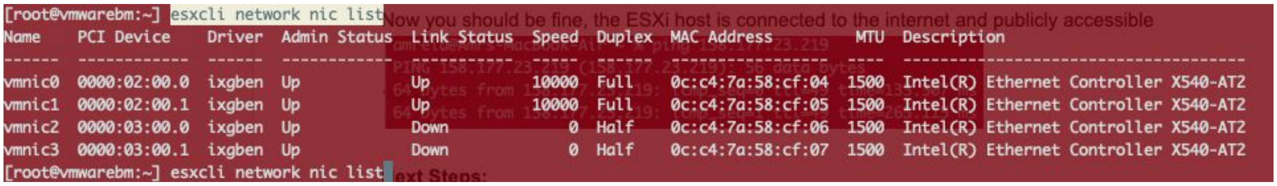
Run `esxcfg-route [your servers public gateway]` to ensure that the public gateway is the default

```
[root@baremetal01:~] esxcfg-route 158.177.23.209
VMkernel default gateway set to 158.177.23.209
```

Troubleshooting:

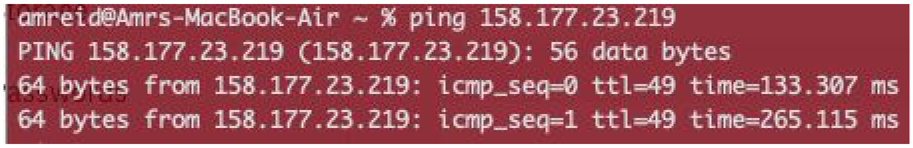
```
esxcli network nic list
```

Make sure you have the right physical NICs , if your machine is not ordered dual, one public and one private are up.
So typically you should find 4 entries, 2 of them are up. Like below



```
[root@vmwarebm:~] esxcli network nic list
Name    PCI Device    Driver  Admin Status  Link Status  Speed  Duplex  MAC Address        MTU   Description
------  ------------  ------  ------------  -----------  -----  ------  -----------------  ----  ------------------------------------
vmnic0  0000:02:00.0  ixgben  Up            Up           10000  Full    0c:c4:7a:58:cf:04  1500  Intel(R) Ethernet Controller X540-AT2
vmnic1  0000:02:00.1  ixgben  Up            Up           10000  Full    0c:c4:7a:58:cf:05  1500  Intel(R) Ethernet Controller X540-AT2
vmnic2  0000:03:00.0  ixgben  Up            Down             0  Half    0c:c4:7a:58:cf:06  1500  Intel(R) Ethernet Controller X540-AT2
vmnic3  0000:03:00.1  ixgben  Up            Down             0  Half    0c:c4:7a:58:cf:07  1500  Intel(R) Ethernet Controller X540-AT2
[root@vmwarebm:~] esxcli network nic list
```

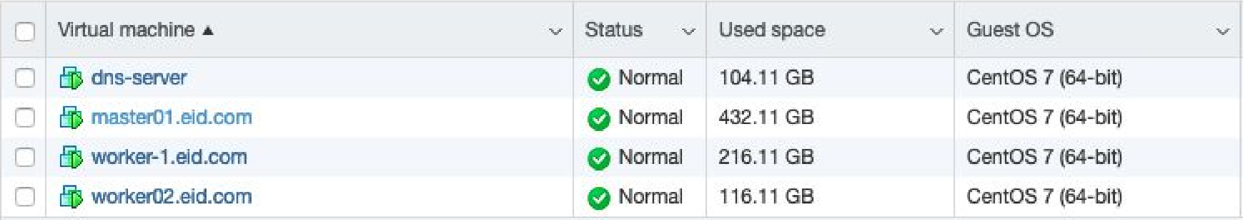
Now you got all set in the ESXi host, the ESXi host is connected to the internet and publicly accessible



```
amreid@Amrs-MacBook-Air ~ % ping 158.177.23.219
PING 158.177.23.219 (158.177.23.219): 56 data bytes
64 bytes from 158.177.23.219: icmp_seq=0 ttl=49 time=133.307 ms
64 bytes from 158.177.23.219: icmp_seq=1 ttl=49 time=265.115 ms
```

Provisioning the environment

The list appears in the screenshot below is what we need to achieve, I will explain how to create the first VM only.



□	Virtual machine ▲	Status	Used space	Guest OS
□	dns-server	✓ Normal	104.11 GB	CentOS 7 (64-bit)
□	master01.eid.com	✓ Normal	432.11 GB	CentOS 7 (64-bit)
□	worker-1.eid.com	✓ Normal	216.11 GB	CentOS 7 (64-bit)
□	worker02.eid.com	✓ Normal	116.11 GB	CentOS 7 (64-bit)

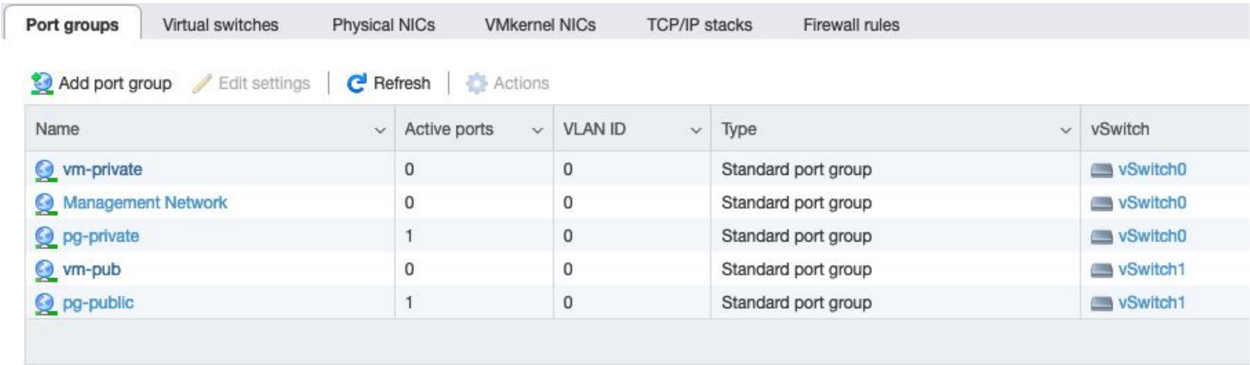
As a **prerequisite**,, we need to get a CentOS image to have it on the datastore.

```
cd vmfs/volumes/datastore1/iso/
wget
http://mirror.fra10.de.leaseweb.net/centos/7/isos/x86_64/CentOS-7-x86_64-Minimal-1908.iso
```

Secondly: Let's us create a VM

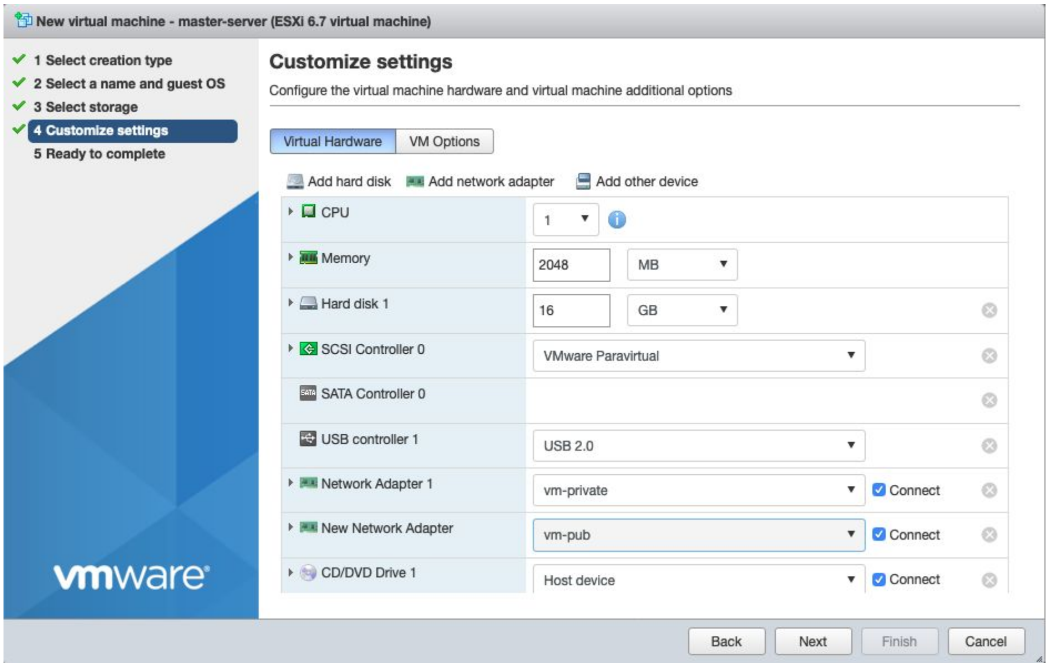
First we need to define if the VM will have private / public network adapters or both.

So, we need to add the related port-groups like below



Name	Active ports	VLAN ID	Type	vSwitch
vm-private	0	0	Standard port group	vSwitch0
Management Network	0	0	Standard port group	vSwitch0
pg-private	1	0	Standard port group	vSwitch0
vm-pub	0	0	Standard port group	vSwitch1
pg-public	1	0	Standard port group	vSwitch1

Then , go ahead and create the VM using the wizard, default setting till you reach "Customize Settings", Add Network Adapter , In the network adapter pick the related port-group for each adapter.


New virtual machine - master-server (ESXi 6.7 virtual machine)

✓ 1 Select creation type
✓ 2 Select a name and guest OS
✓ 3 Select storage
✓ 4 Customize settings
5 Ready to complete

Customize settings

Configure the virtual machine hardware and virtual machine additional options

CPU	1	
Memory	2048 MB	
Hard disk 1	16 GB	
SCSI Controller 0	VMware Paravirtual	
SATA Controller 0		
USB controller 1	USB 2.0	
Network Adapter 1	vm-private	☑ Connect
New Network Adapter	vm-pub	☑ Connect
CD/DVD Drive 1	Host device	☑ Connect

Following the same wizard, From the above step, From CD/DVD pick the proper iso for your OS. Complete the steps and power the VM on.

You can either configure the network/DNS before starting the real installation using CentOS setup wizard. For the Network, you will only be asked to provide an IP/Netmask and Gateway.
Or
After installing the OS, you can manually configure the network by accessing the VM from ESXi host, and using the CLi you can configure the network with a tool like nmtui

The below is to validate your network settings

- You can try to ping a domain like www.ibm.com
- If it worked, then you got all set. If not, look to DNS setup
- For now, you an use any public DNS like 8.8.8.8
- Try `cat /etc/resolv.conf`, if it is blank, you need to set up a DNS resolver. I would suggest editing the resolv.conf with `nameserver 8.8.8.8`
- Try ping `www.ibm.com` or `yum update` again.

One final step, as you will have to install many VMs, it is good to change the hostname for every VM to be descriptive, for CentOS, you can do it from ESXi installation wizard and you can do it later through

```
hostnamectl set-hostname "eidns.amreid.com"
exec bash
```

Section 03- Installing Openshift

03-01 Host Preparation - DNS Node Setup
DNSMASQ: How to install and configure

Install

```
yum install dnsmasq bind-utils
```

Configure

Edit those two files, and add the below line to both of them
/etc/resolv.conf and /etc/resolv.dnsmasq

```
nameserver 127.0.0.1
```

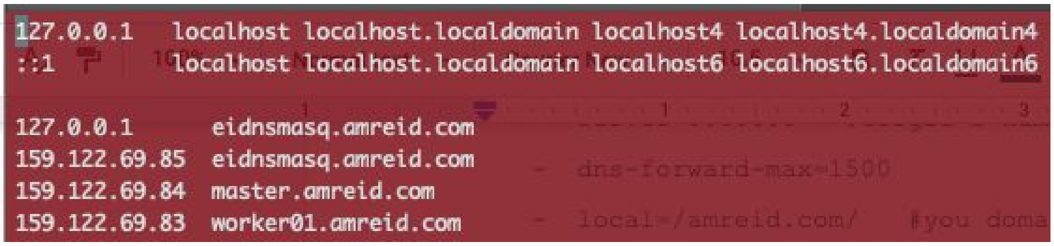
You will need to edit those two files 1- /etc/dnsmasq.conf and 2-/etc/hosts as follows

```
cp /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
vi /etc/dnsmasq.conf
```

At the end of the file you can just add

```
server=8.8.8.8   #Google’s nameserver
local=/amreid.com/   #you domain servers to be looked locally
#Log-queries         # In case you need to enable logs
#log-facility=/var/log/dnsmasq.log
```

Now, edit vi /etc/hosts and add your hosts like below

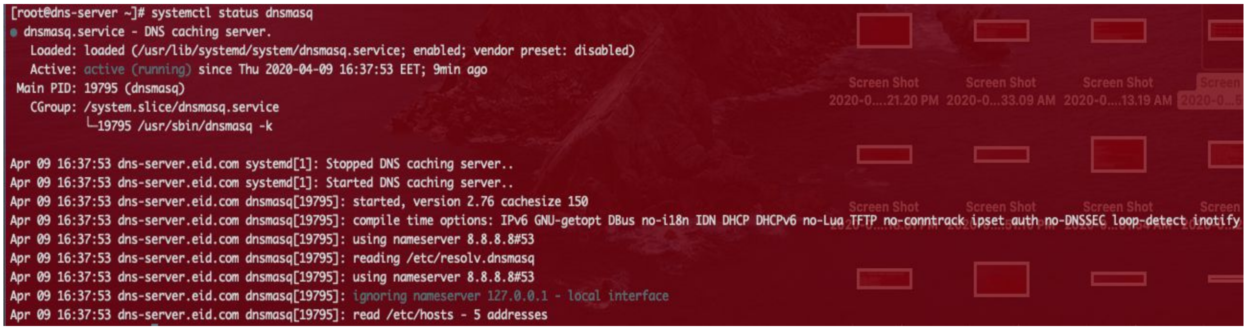


```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6

127.0.0.1      eidnsmasq.amreid.com
159.122.69.85  eidnsmasq.amreid.com
159.122.69.84  master.amreid.com
159.122.69.83  worker01.amreid.com
```

Then, start and enable the dns service.

```
systemctl start dnsmasq
systemctl enable dnsmasq
systemctl status dnsmasq
```

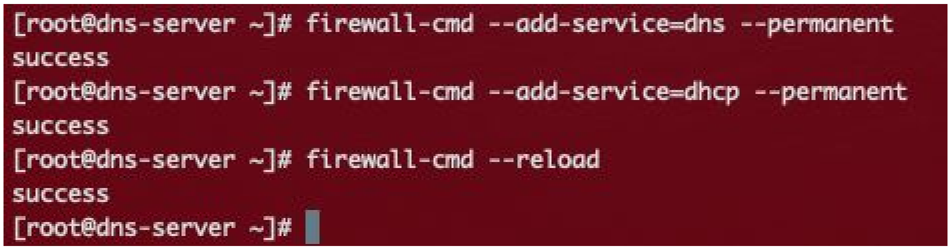
You should expect results like below saying that it is up and running.



```
[root@dns-server ~]# systemctl status dnsmasq
● dnsmasq.service - DNS caching server.
   Loaded: loaded (/usr/lib/systemd/system/dnsmasq.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-04-09 16:37:53 EET; 9min ago
 Main PID: 19795 (dnsmasq)
   CGroup: /system.slice/dnsmasq.service
           └─19795 /usr/sbin/dnsmasq -k

Apr 09 16:37:53 dns-server.eid.com systemd[1]: Stopped DNS caching server..
Apr 09 16:37:53 dns-server.eid.com systemd[1]: Started DNS caching server..
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: started, version 2.76 cachesize 150
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: compile time options: IPv6 GNU-getopt DBus no-i18n IDN DHCP DHCPv6 no-Lua TFTP no-conntrack ipset auth no-DNSSEC loop-detect inotify
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: using nameserver 8.8.8.8#53
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: reading /etc/resolv.dnsmasq
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: using nameserver 8.8.8.8#53
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: ignoring nameserver 127.0.0.1 - local interface
Apr 09 16:37:53 dns-server.eid.com dnsmasq[19795]: read /etc/hosts - 5 addresses
```

Firewall

For now only, to keep the installation easier, let’s either stop the firewall if you don’t need it, Or open DNS and DHCP services in the firewall configuration, to allow requests from hosts on your LAN to pass to the dnsmasq server.



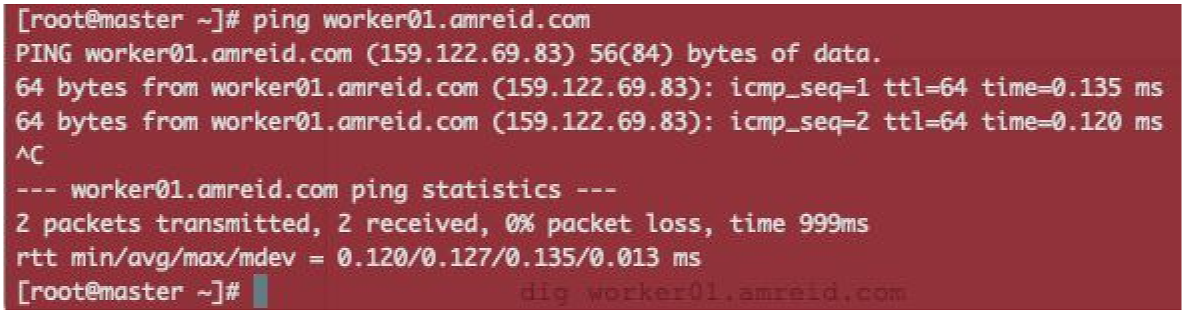
```
[root@dns-server ~]# firewall-cmd --add-service=dns --permanent
success
[root@dns-server ~]# firewall-cmd --add-service=dhcp --permanent
success
[root@dns-server ~]# firewall-cmd --reload
success
[root@dns-server ~]#
```

If you like to disable it you can use the below commands

```
systemctl status firewalld
service firewalld stop
systemctl disable firewalld
```

Validation

From worker or Master node, you can validate also by testing if you can reach to each other like

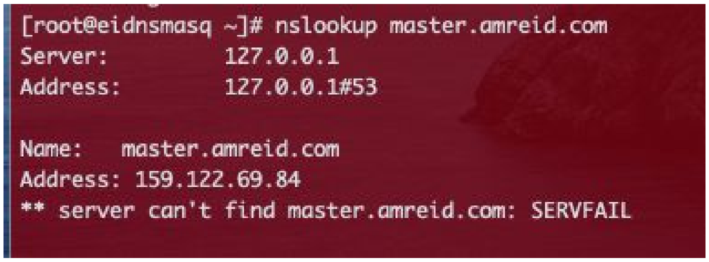


```
[root@master ~]# ping worker01.amreid.com
PING worker01.amreid.com (159.122.69.83) 56(84) bytes of data.
64 bytes from worker01.amreid.com (159.122.69.83): icmp_seq=1 ttl=64 time=0.135 ms
64 bytes from worker01.amreid.com (159.122.69.83): icmp_seq=2 ttl=64 time=0.120 ms
^C
--- worker01.amreid.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.120/0.127/0.135/0.013 ms
[root@master ~]#
```

From the DNS server , from each other host, you can check if it can access the others.

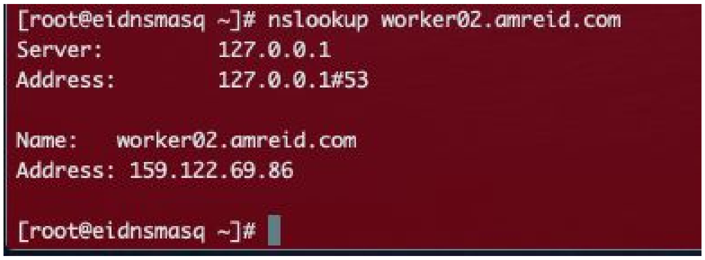
```
dig worker01.amreid.com
OR  # nslookup worker01.amreid.com
```

Preferably nslookup and make sure to read the results carefully, If it shows this last line message, then it is not going to work properly.



```
[root@eidnsmasq ~]# nslookup master.amreid.com
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   master.amreid.com
Address: 159.122.69.84
** server can't find master.amreid.com: SERVFAIL
```

The right execution should looks like



```
[root@eidnsmasq ~]# nslookup worker02.amreid.com
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   worker02.amreid.com
Address: 159.122.69.86

[root@eidnsmasq ~]#
```

03-02 Installing Openshift -3.11

Host Preparation: Stage 1

Make sure that all the prerequisites are working fine. DNS is up and running , Masters and Worker nodes can see each other on top of the DNS.

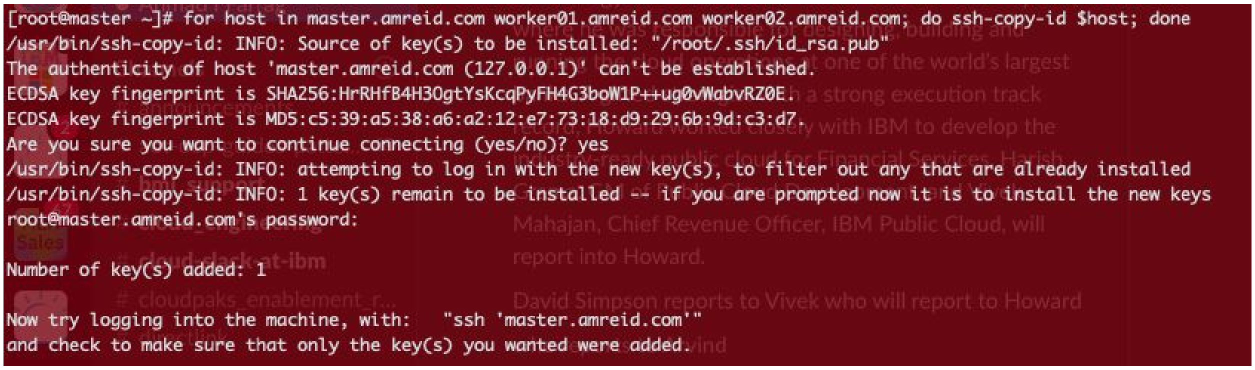
Actions:

ssh-keygen on master node
distribute the key all over the nodes. (for host in xxxxxxx; do ssh-copy-id $host done)
On the Master Node:

```
ssh-keygen
```

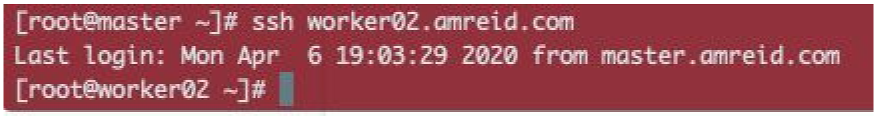
Then

```
for host in master.amreid.com worker01.amreid.com worker02.amreid.com; do
ssh-copy-id $host; done
```



```
[root@master ~]# for host in master.amreid.com worker01.amreid.com worker02.amreid.com; do ssh-copy-id $host; done
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host 'master.amreid.com (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:HrRHfB4H3OgtYsKcqPyFH4G3boW1P++ug0vWabvRZ0E.
ECDSA key fingerprint is MD5:c5:39:a5:38:a6:a2:12:e7:73:18:d9:29:6b:9d:c3:d7.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@master.amreid.com's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'master.amreid.com'"
and check to make sure that only the key(s) you wanted were added.
```

As a sanity check, you can validate the above step has been successfully by trying to ssh to each one of the worker nodes while being on the master node, you should login without any need to provide a password or a key



```
[root@master ~]# ssh worker02.amreid.com
Last login: Mon Apr  6 19:03:29 2020 from master.amreid.com
[root@worker02 ~]#
```

Host Preparation: Stage 2

Actions: Install base packages

On the master node: Install all the below packages

```
yum install wget git net-tools bind-utils yum-utils iptables-services
bridge-utils bash-completion kexec-tools sos psacct
yum install epel-release
yum install ansible pyOpenSSL
cd ~
git clone https://github.com/openshift/openshift-ansible
cd openshift-ansible
git checkout release-3.11
```

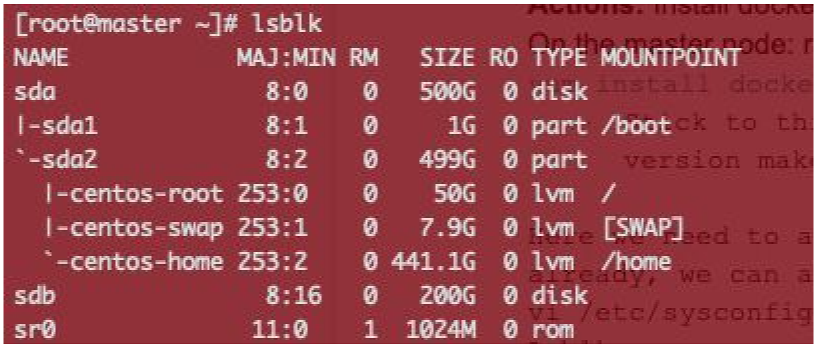
Host Preparation: Stage 3

Actions: Install docker

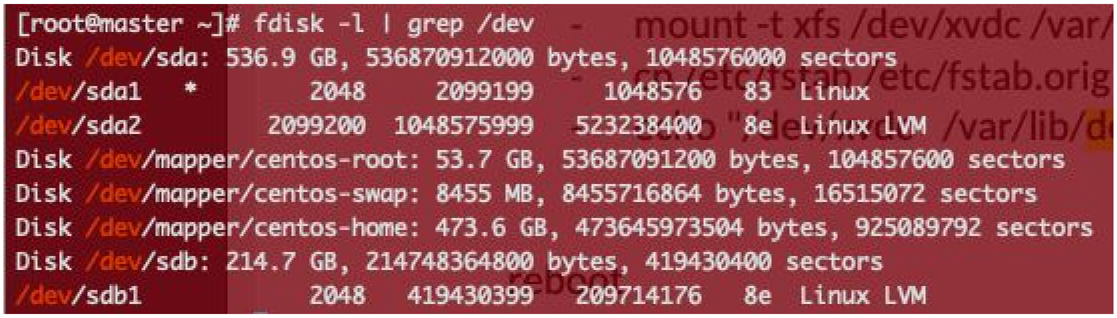
On the master node: run the below commands

```
yum install docker-1.13.1
```

This exact version is advised to be used with openshift3.11.
Here we need to add a dedicated LVM Volume Group , so if not added already, we can add a separate disk to the master VM then
`Lsblk` or `fdisk -l`



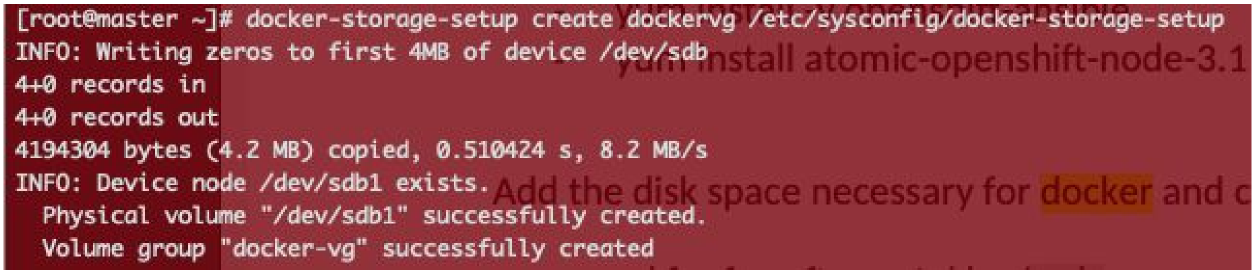
```
[root@master ~]# lsblk
NAME              MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda                 8:0    0   500G  0 disk
|-sda1              8:1    0     1G  0 part /boot
`-sda2              8:2    0   499G  0 part
  |-centos-root 253:0    0    50G  0 lvm  /
  |-centos-swap 253:1    0   7.9G  0 lvm  [SWAP]
  `-centos-home 253:2    0 441.1G  0 lvm  /home
sdb                 8:16   0   200G  0 disk
sr0                11:0    1  1024M  0 rom
```



```
[root@master ~]# fdisk -l | grep /dev
Disk /dev/sda: 536.9 GB, 536870912000 bytes, 1048576000 sectors
/dev/sda1   *        2048     2099199     1048576   83  Linux
/dev/sda2         2099200  1048575999   523238400   8e  Linux LVM
Disk /dev/mapper/centos-root: 53.7 GB, 53687091200 bytes, 104857600 sectors
Disk /dev/mapper/centos-swap: 8455 MB, 8455716864 bytes, 16515072 sectors
Disk /dev/mapper/centos-home: 473.6 GB, 473645973504 bytes, 925089792 sectors
Disk /dev/sdb: 214.7 GB, 214748364800 bytes, 419430400 sectors
/dev/sdb1            2048   419430399   209714176   8e  Linux LVM
```

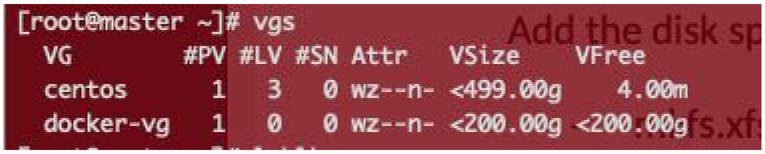
To get the name of the disk, here it is /dev/sdb

- Edit the file `vi /etc/sysconfig/docker-storage-setup`
  Add
  `STORAGE_DRIVER=overlay2`
  `DEVS=/dev/sdb`
  `VG=docker-vg`
  Run the file using the below command
- `docker-storage-setup create dockervg /etc/sysconfig/docker-storage-setup`



```
[root@master ~]# docker-storage-setup create dockervg /etc/sysconfig/docker-storage-setup
INFO: Writing zeros to first 4MB of device /dev/sdb
4+0 records in
4+0 records out
4194304 bytes (4.2 MB) copied, 0.510424 s, 8.2 MB/s
INFO: Device node /dev/sdb1 exists.
  Physical volume "/dev/sdb1" successfully created.
  Volume group "docker-vg" successfully created
```

In order to verify this phase, just run `vgs` to see the created Volume Groups. (20G is enough for the dedicated VG)



```
[root@master ~]# vgs
  VG        #PV #LV #SN Attr   VSize    VFree
  centos      1   3   0 wz--n- <499.00g   4.00m
  docker-vg   1   0   0 wz--n- <200.00g <200.00g
```

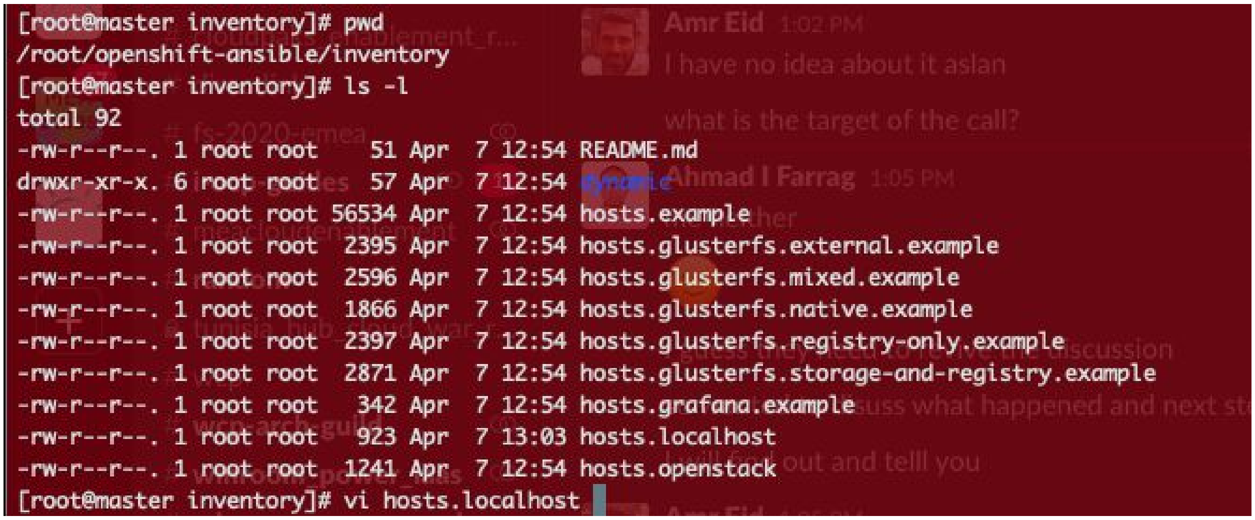
Openshift Installation: Using Ansible Installer

Actions:

- Installing Openshift using Ansible Playbook
- The initial configuration is in Ansible Inventory file /etc/ansible/hosts (on the Master Node).

Step 01

```
cd ~/openshift-ansible
cp hosts.localhost /etc/ansible/hosts
vi /etc/ansible/hosts
```

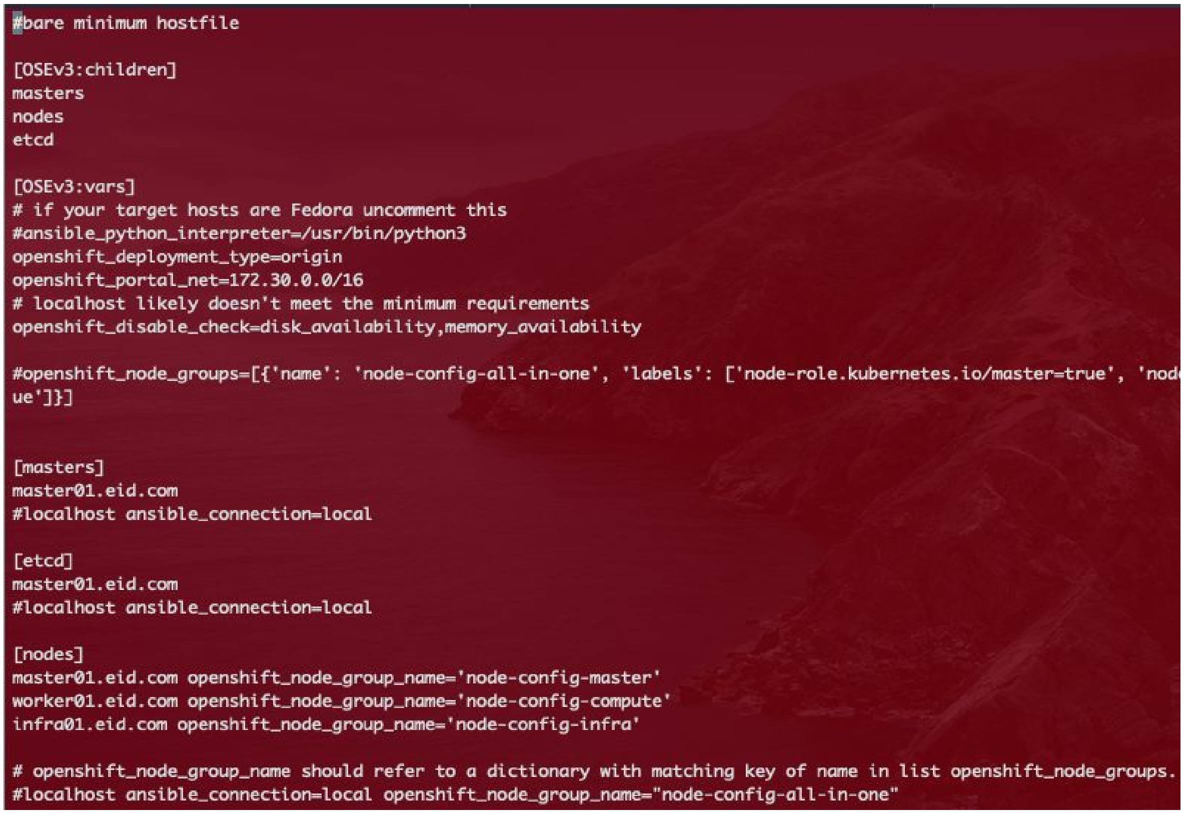


```
[root@master inventory]# pwd
/root/openshift-ansible/inventory
[root@master inventory]# ls -l
total 92
-rw-r--r--. 1 root root    51 Apr  7 12:54 README.md
drwxr-xr-x. 6 root root    57 Apr  7 12:54 dynamic
-rw-r--r--. 1 root root 56534 Apr  7 12:54 hosts.example
-rw-r--r--. 1 root root  2395 Apr  7 12:54 hosts.glusterfs.external.example
-rw-r--r--. 1 root root  2596 Apr  7 12:54 hosts.glusterfs.mixed.example
-rw-r--r--. 1 root root  1866 Apr  7 12:54 hosts.glusterfs.native.example
-rw-r--r--. 1 root root  2397 Apr  7 12:54 hosts.glusterfs.registry-only.example
-rw-r--r--. 1 root root  2871 Apr  7 12:54 hosts.glusterfs.storage-and-registry.example
-rw-r--r--. 1 root root   342 Apr  7 12:54 hosts.grafana.example
-rw-r--r--. 1 root root   923 Apr  7 13:03 hosts.localhost
-rw-r--r--. 1 root root  1241 Apr  7 12:54 hosts.openstack
[root@master inventory]# vi hosts.localhost
```

Step 02

Edit your ansible hosts file to have your nodes under the proper sections and make sure that you've added group names like below.

```
openshift_deployment_type=origin
openshift_portal_net=172.30.0.0/16
openshift_disable_check=disk_availability,memory_availability

[masters]
master01.eid.com

[etcd]
master01.eid.com

[nodes]
master01.eid.com openshift_node_group_name='node-config-master'
worker01.eid.com openshift_node_group_name='node-config-compute'
infra01.eid.com openshift_node_group_name='node-config-infra'
```



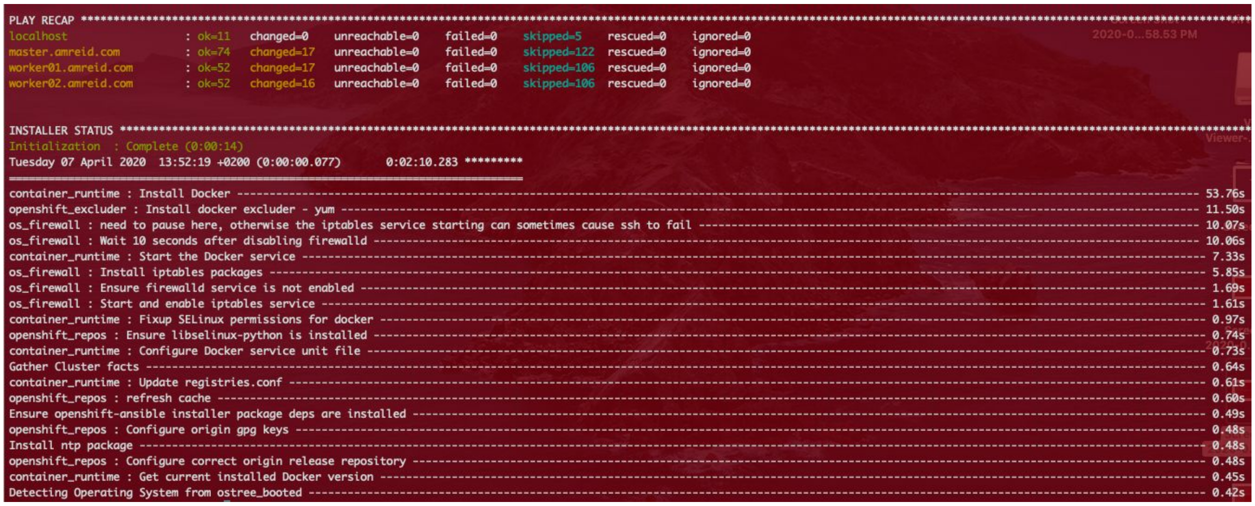
```
#bare minimum hostfile

[OSEv3:children]
masters
nodes
etcd

[OSEv3:vars]
# if your target hosts are Fedora uncomment this
#ansible_python_interpreter=/usr/bin/python3
openshift_deployment_type=origin
openshift_portal_net=172.30.0.0/16
# localhost likely doesn't meet the minimum requirements
openshift_disable_check=disk_availability,memory_availability

#openshift_node_groups=[{'name': 'node-config-all-in-one', 'labels': ['node-role.kubernetes.io/master=true', 'nod
ue']}]


[masters]
master01.eid.com
#localhost ansible_connection=local

[etcd]
master01.eid.com
#localhost ansible_connection=local

[nodes]
master01.eid.com openshift_node_group_name='node-config-master'
worker01.eid.com openshift_node_group_name='node-config-compute'
infra01.eid.com openshift_node_group_name='node-config-infra'

# openshift_node_group_name should refer to a dictionary with matching key of name in list openshift_node_groups.
#localhost ansible_connection=local openshift_node_group_name="node-config-all-in-one"
```

Step 03

Make sure that you are on the openshift-ansible root folder then run the prerequisites.yml script

```
ansible-playbook playbooks/prerequisites.yml
```

This prerequisites script is testing the environment before the installation. You should receive green results like below.

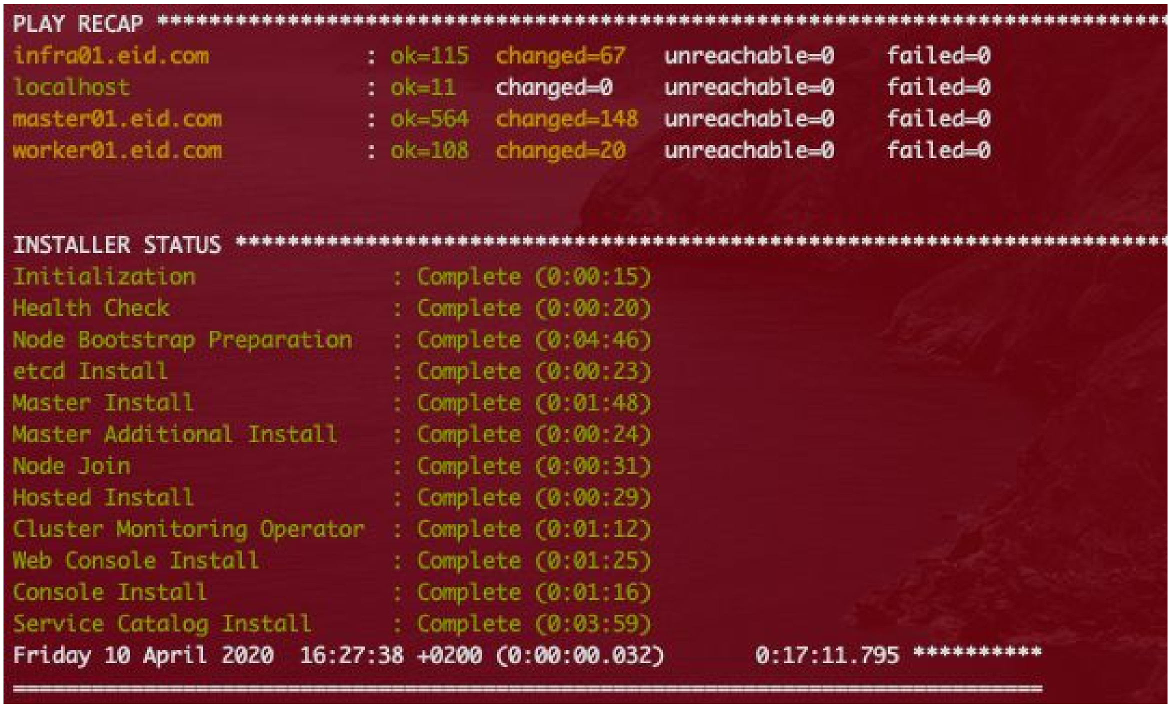


```
PLAY RECAP *******************************************************************************************************************
localhost                  : ok=11   changed=0    unreachable=0    failed=0    skipped=5    rescued=0    ignored=0
master.amreid.com          : ok=74   changed=17   unreachable=0    failed=0    skipped=122  rescued=0    ignored=0
worker01.amreid.com        : ok=52   changed=17   unreachable=0    failed=0    skipped=106  rescued=0    ignored=0
worker02.amreid.com        : ok=52   changed=16   unreachable=0    failed=0    skipped=106  rescued=0    ignored=0


INSTALLER STATUS *************************************************************************************************************
Initialization  : Complete (0:00:14)
Tuesday 07 April 2020  13:52:19 +0200 (0:00:00.077)       0:02:10.283 *********
===============================================================================
container_runtime : Install Docker ------------------------------------------------------------------------------------- 53.76s
openshift_excluder : Install docker excluder - yum -------------------------------------------------------------------- 11.50s
os_firewall : need to pause here, otherwise the iptables service starting can sometimes cause ssh to fail ------------- 10.07s
os_firewall : Wait 10 seconds after disabling firewalld --------------------------------------------------------------- 10.06s
container_runtime : Start the Docker service --------------------------------------------------------------------------- 7.33s
os_firewall : Install iptables packages -------------------------------------------------------------------------------- 5.85s
os_firewall : Ensure firewalld service is not enabled ------------------------------------------------------------------ 1.69s
os_firewall : Start and enable iptables service ------------------------------------------------------------------------ 1.61s
container_runtime : Fixup SELinux permissions for docker --------------------------------------------------------------- 0.97s
openshift_repos : Ensure libselinux-python is installed ---------------------------------------------------------------- 0.74s
container_runtime : Configure Docker service unit file ----------------------------------------------------------------- 0.73s
Gather Cluster facts --------------------------------------------------------------------------------------------------- 0.64s
container_runtime : Update registries.conf ----------------------------------------------------------------------------- 0.61s
openshift_repos : refresh cache ---------------------------------------------------------------------------------------- 0.60s
Ensure openshift-ansible installer package deps are installed ---------------------------------------------------------- 0.49s
openshift_repos : Configure origin gpg keys ---------------------------------------------------------------------------- 0.48s
Install ntp package ---------------------------------------------------------------------------------------------------- 0.48s
openshift_repos : Configure correct origin release repository ---------------------------------------------------------- 0.48s
container_runtime : Get current installed Docker version --------------------------------------------------------------- 0.45s
Detecting Operating System from ostree_booted -------------------------------------------------------------------------- 0.42s
```

Step 04

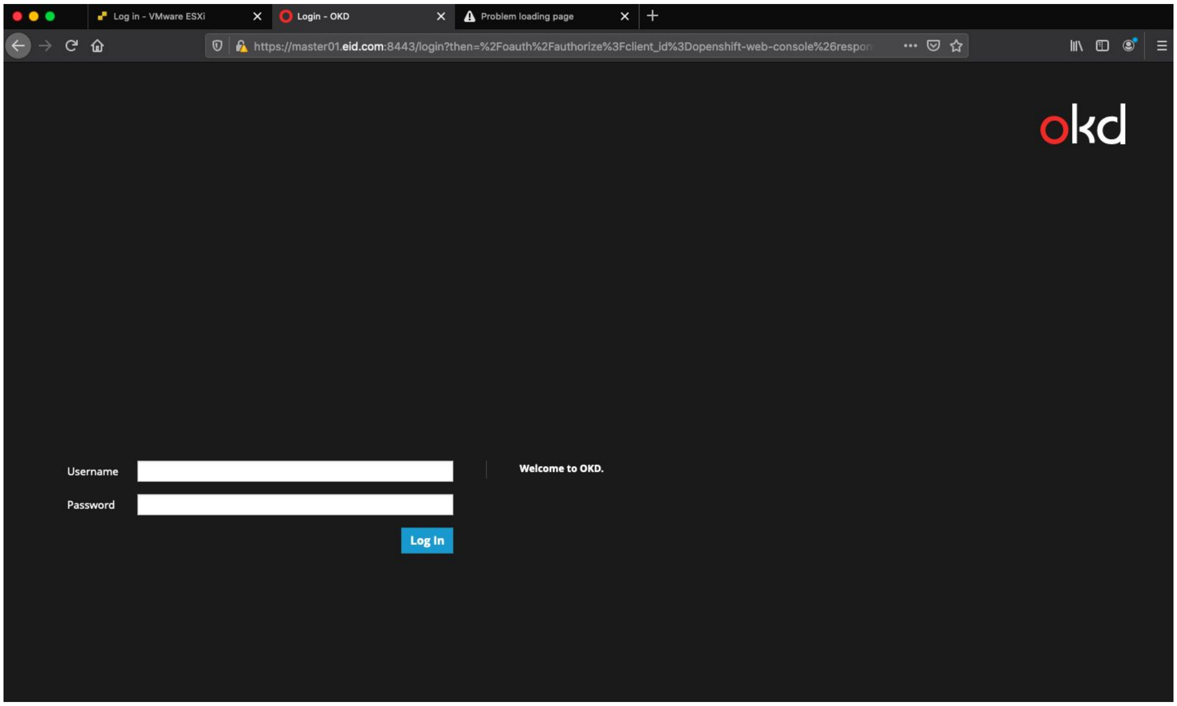
```
ansible-playbook playbooks/deploy_cluster.ym
```

This is going to take time (~15min), then you should receive green results similar to the previous step.

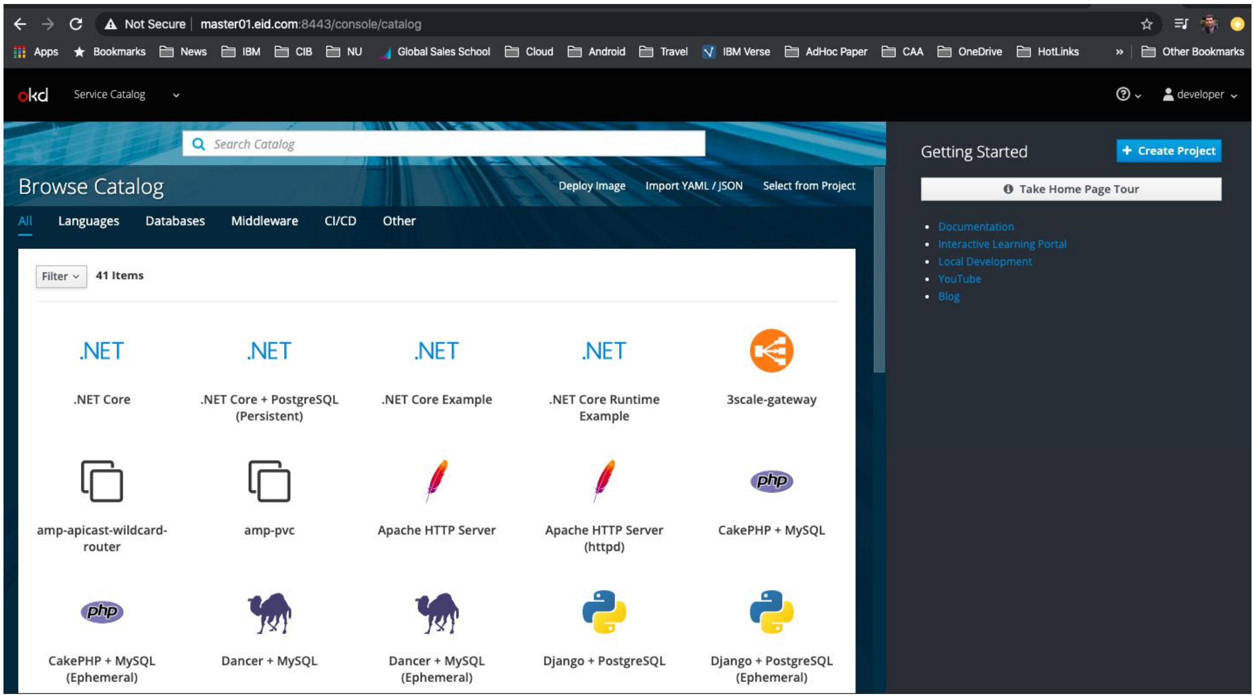


```
PLAY RECAP *******************************************************************************
infra01.eid.com            : ok=115  changed=67   unreachable=0    failed=0
localhost                  : ok=11   changed=0    unreachable=0    failed=0
master01.eid.com           : ok=564  changed=148  unreachable=0    failed=0
worker01.eid.com           : ok=108  changed=20   unreachable=0    failed=0


INSTALLER STATUS *************************************************************************
Initialization              : Complete (0:00:15)
Health Check                : Complete (0:00:20)
Node Bootstrap Preparation  : Complete (0:04:46)
etcd Install                : Complete (0:00:23)
Master Install              : Complete (0:01:48)
Master Additional Install   : Complete (0:00:24)
Node Join                   : Complete (0:00:31)
Hosted Install              : Complete (0:00:29)
Cluster Monitoring Operator : Complete (0:01:12)
Web Console Install         : Complete (0:01:25)
Console Install             : Complete (0:01:16)
Service Catalog Install     : Complete (0:03:59)
Friday 10 April 2020  16:27:38 +0200 (0:00:00.032)       0:17:11.795 **********
===============================================================================
```

Step 05 - Validating and Testing your open shift environment
With the previous good news, you can add your master node IP to your hosts file and then using https open your browser to access openshift console, note that openshift 3.11 is using port 8433.
https://master01.eid.com:8443

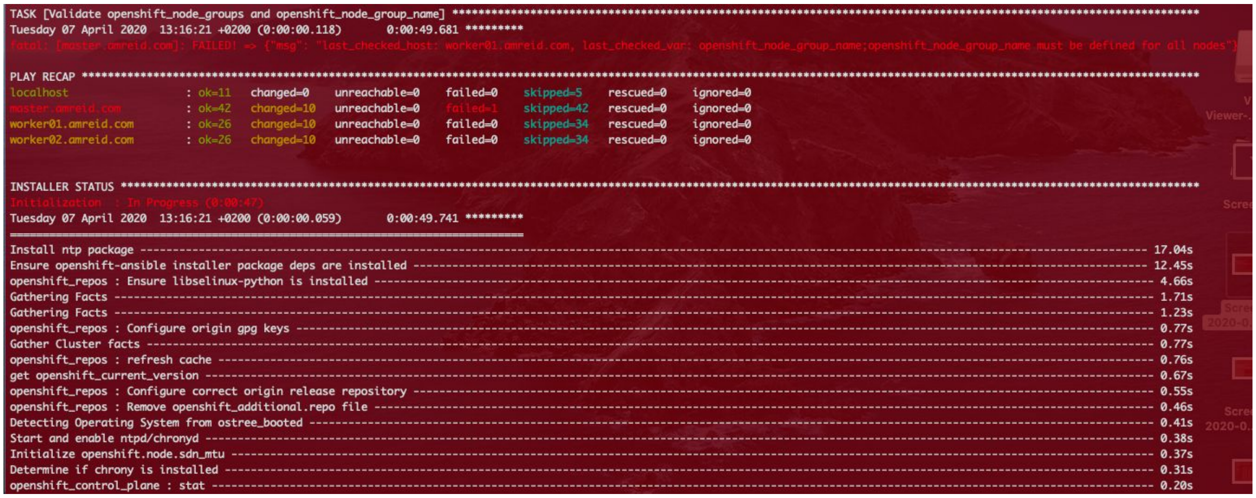

Log in - VMware ESXi
Login - OKD
Problem loading page
https://master01.eid.com:8443/login?then=%2Foauth%2Fauthorize%3Fclient_id%3Dopenshift-web-console%26respon
okd
Username
Password
Log In
Welcome to OKD.

You can try log in using developer / any password

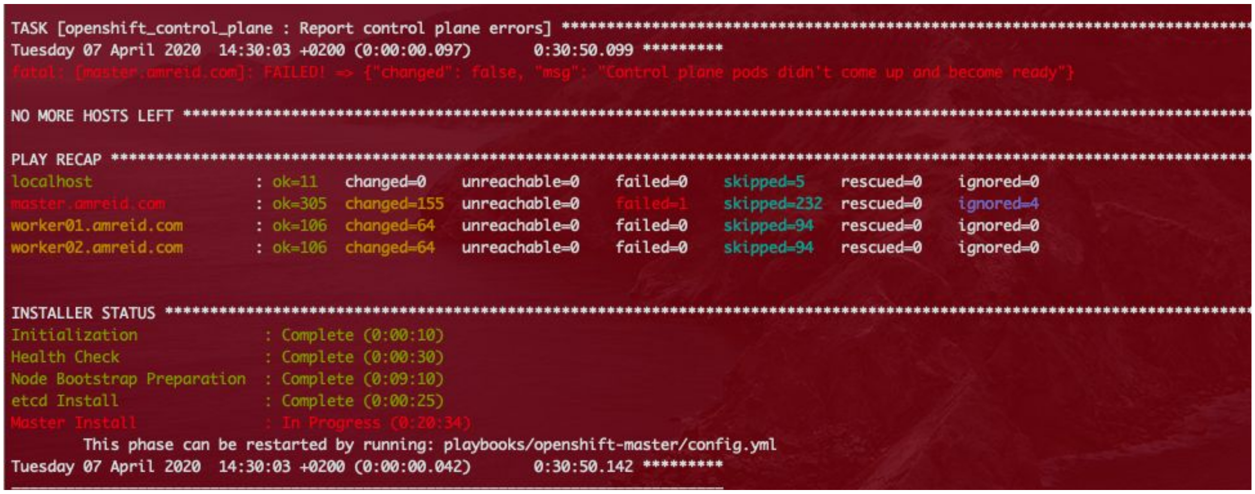


Troubleshooting

This section is trying to capture some of the errors which may happen while trying to install specially for your first time.



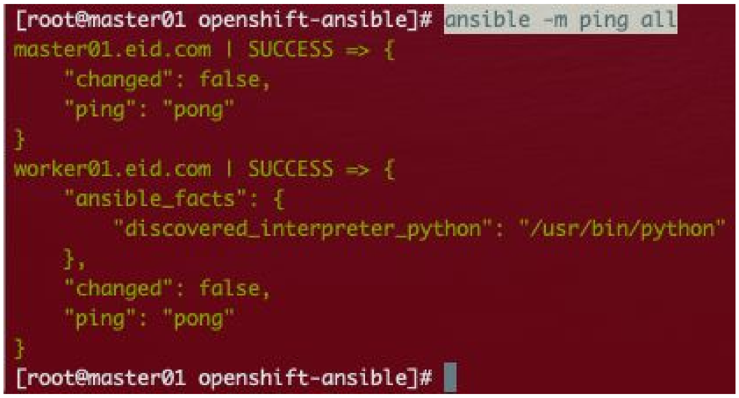
```
TASK [Validate openshift_node_groups and openshift_node_group_name] ************************************************************************************
Tuesday 07 April 2020  13:16:21 +0200 (0:00:00.118)       0:00:49.681 *********
fatal: [master.amreid.com]: FAILED! => {"msg": "last_checked_host: worker01.amreid.com, last_checked_var: openshift_node_group_name;openshift_node_group_name must be defined for all nodes"}

PLAY RECAP *********************************************************************************************************************************************
localhost                  : ok=11   changed=0    unreachable=0    failed=0    skipped=5    rescued=0    ignored=0
master.amreid.com          : ok=42   changed=10   unreachable=0    failed=1    skipped=42   rescued=0    ignored=0
worker01.amreid.com        : ok=26   changed=10   unreachable=0    failed=0    skipped=34   rescued=0    ignored=0
worker02.amreid.com        : ok=26   changed=10   unreachable=0    failed=0    skipped=34   rescued=0    ignored=0


INSTALLER STATUS ***************************************************************************************************************************************
Initialization  : In Progress (0:00:47)
Tuesday 07 April 2020  13:16:21 +0200 (0:00:00.059)       0:00:49.741 *********
===============================================================================
Install ntp package ---------------------------------------------------------------------------------------------------------------------------- 17.04s
Ensure openshift-ansible installer package deps are installed ---------------------------------------------------------------------------------- 12.45s
openshift_repos : Ensure libselinux-python is installed ---------------------------------------------------------------------------------------- 4.66s
Gathering Facts -------------------------------------------------------------------------------------------------------------------------------- 1.71s
Gathering Facts -------------------------------------------------------------------------------------------------------------------------------- 1.23s
openshift_repos : Configure origin gpg keys ---------------------------------------------------------------------------------------------------- 0.77s
Gather Cluster facts --------------------------------------------------------------------------------------------------------------------------- 0.77s
openshift_repos : refresh cache ---------------------------------------------------------------------------------------------------------------- 0.76s
get openshift_current_version ------------------------------------------------------------------------------------------------------------------ 0.67s
openshift_repos : Configure correct origin release repository ---------------------------------------------------------------------------------- 0.55s
openshift_repos : Remove openshift_additional.repo file ---------------------------------------------------------------------------------------- 0.46s
Detecting Operating System from ostree_booted -------------------------------------------------------------------------------------------------- 0.41s
Start and enable ntpd/chronyd ------------------------------------------------------------------------------------------------------------------ 0.38s
Initialize openshift.node.sdn_mtu -------------------------------------------------------------------------------------------------------------- 0.37s
Determine if chrony is installed --------------------------------------------------------------------------------------------------------------- 0.31s
openshift_control_plane : stat ----------------------------------------------------------------------------------------------------------------- 0.20s
```



```
TASK [openshift_control_plane : Report control plane errors] ***********************************************************************
Tuesday 07 April 2020  14:30:03 +0200 (0:00:00.097)       0:30:50.099 *********
fatal: [master.amreid.com]: FAILED! => {"changed": false, "msg": "Control plane pods didn't come up and become ready"}

NO MORE HOSTS LEFT *****************************************************************************************************************

PLAY RECAP *************************************************************************************************************************
localhost                  : ok=11   changed=0    unreachable=0    failed=0    skipped=5    rescued=0    ignored=0
master.amreid.com          : ok=305  changed=155  unreachable=0    failed=1    skipped=232  rescued=0    ignored=4
worker01.amreid.com        : ok=106  changed=64   unreachable=0    failed=0    skipped=94   rescued=0    ignored=0
worker02.amreid.com        : ok=106  changed=64   unreachable=0    failed=0    skipped=94   rescued=0    ignored=0


INSTALLER STATUS *******************************************************************************************************************
Initialization              : Complete (0:00:10)
Health Check                : Complete (0:00:30)
Node Bootstrap Preparation  : Complete (0:09:10)
etcd Install                : Complete (0:00:25)
Master Install              : In Progress (0:20:34)
        This phase can be restarted by running: playbooks/openshift-master/config.yml
Tuesday 07 April 2020  14:30:03 +0200 (0:00:00.042)       0:30:50.142 *********
```

In order to fix these kinds of errors, Here are some notes you can keep in mind.

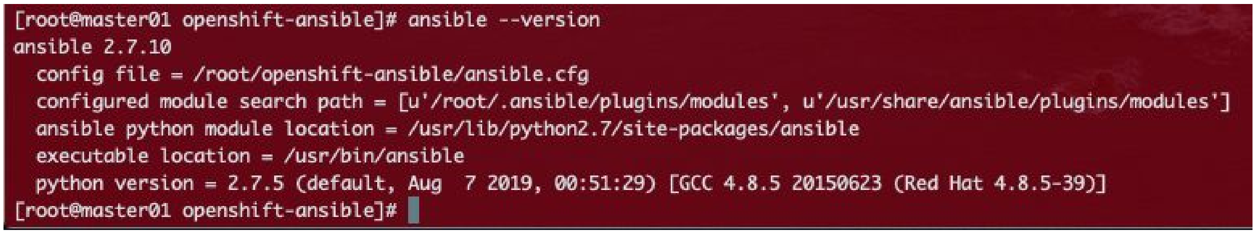
- Try using `-vvv` to run the scripts in verbose mode if you want to have more descriptive errors.
- Any connection related errors (e.x SSH connection to any node), most probably it will be related to the host preparation section in this document so, give Dnsmasq setup your main focus. I've listed it above in detail.
- A lot of ansible script issues will be related to missing some detail in Ansible hosts file. So review the hosts file carefully and make sure that the group name is setted for all the nodes.
- Before initiating the deploy script, make sure you can access all the nodes from Master Node using Ansible like below.
- `ansible -m ping all`
- The results should be looks like



```
[root@master01 openshift-ansible]# ansible -m ping all
master01.eid.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
worker01.eid.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[root@master01 openshift-ansible]# 
```

- You may need to install Ansible stable version well tested with Openshift 3.1 recommended version is: 2.7.1
  How to install Ansible 2.7.1

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
python get-pip.py
pip install ansible==2.7.10
ansible --version
```



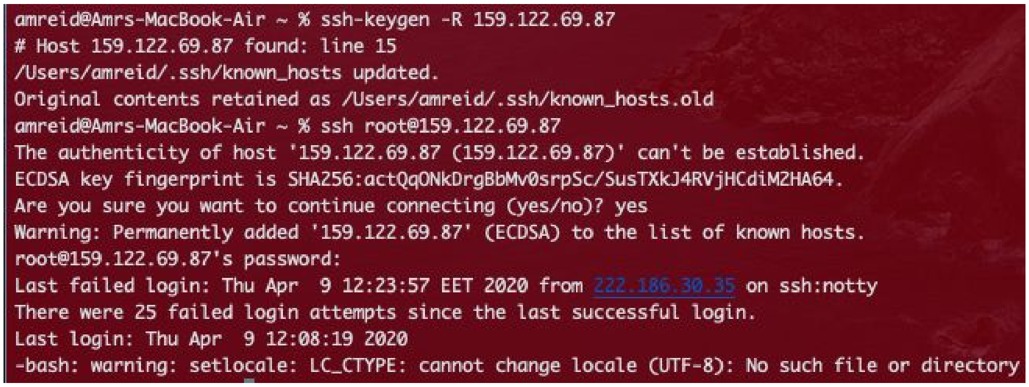
```
[root@master01 openshift-ansible]# ansible --version
ansible 2.7.10
  config file = /root/openshift-ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Aug  7 2019, 00:51:29) [GCC 4.8.5 20150623 (Red Hat 4.8.5-39)]
[root@master01 openshift-ansible]# 
```

- You may face some issues while logging into the dns server like below

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
```

How to fix it, simply run the below command.

```
ssh-keygen -R
```



```
amreid@Amrs-MacBook-Air ~ % ssh-keygen -R 159.122.69.87
# Host 159.122.69.87 found: line 15
/Users/amreid/.ssh/known_hosts updated.
Original contents retained as /Users/amreid/.ssh/known_hosts.old
amreid@Amrs-MacBook-Air ~ % ssh root@159.122.69.87
The authenticity of host '159.122.69.87 (159.122.69.87)' can't be established.
ECDSA key fingerprint is SHA256:actQqONkDrgBbMv0srpSc/SusTXkJ4RVjHCdiMZHA64.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '159.122.69.87' (ECDSA) to the list of known hosts.
root@159.122.69.87's password:
Last failed login: Thu Apr  9 12:23:57 EET 2020 from 222.186.30.35 on ssh:notty
There were 25 failed login attempts since the last successful login.
Last login: Thu Apr  9 12:08:19 2020
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
```