**Deploying Management Stack & Installing OCP 4.3 with non-root user**

**Overview:**

1. ***Deploying OpenShift Container Platform with 4.3.x on management nodes CoreOS Control plane/Master.***
   1. Creation of Installer VM.
   2. Downloading OCP bits in the installer VM.
   3. Automated to create bootstrap, three masters, haproxy VMs where vSphere rules are applied.
   4. Updating the MAC address and corresponding IP address in the configuration file within PXE.
   5. Creating the DNS entries.
   6. Create and configure Load Balance with HA Proxy server.
   7. Update the /etc/haproxy/haproxy.cfg file in the HA Proxy server and haproxy service should be up.
   8. Download and update the install-config.yaml
   9. Create the ignition files.
   10. Add the ignition files to the web/ftp server and restart service.
   11. Update the ignition file location in the tftp server and restart service.
   12. Install operating system on the bootstrap and master nodes.
   13. Install the OCP cluster.

**Deploying OpenShift Container Platform with 4.3.x on management nodes CoreOS Control plane/Master.**

**Pre-requisites:**

Ensure there is a RHEL 7.6 VM with the following configurations to be used as an installer VM.

* Memory: 8GB
* Hard Disk: 120GB
* CPU: 4
* OS: RHEL7/8
* Network adapter: PXE network

1. Minimum three vSphere6.7 hosts are required and should be attached to the same data storage under existing Data Center & Data Cluster.
2. Pyhton 3.6.x, ansible 2.9.x are required on installer VM.

Refer **APPENDIX A** for installing Python 3.6.x and Ansible 2.9.x.

1. git and PyVmomi on installer VM.

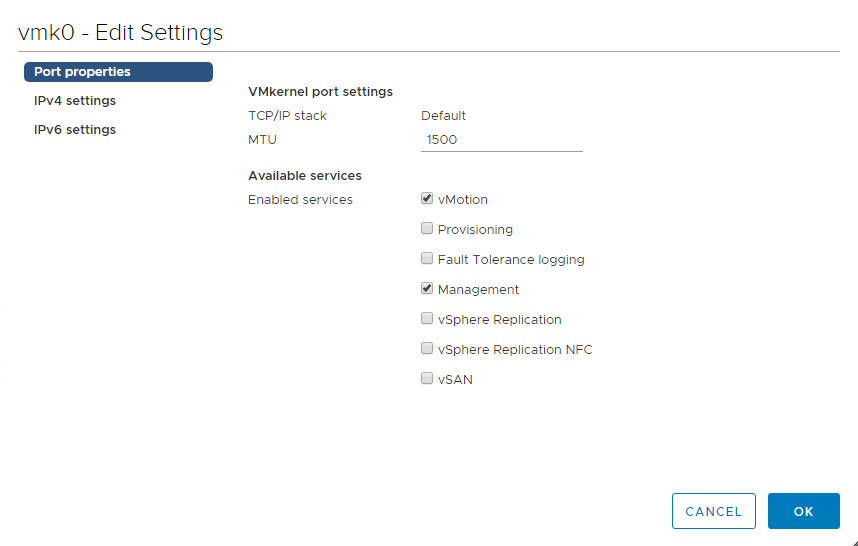
Refer **APPENDIX B** for installing PyVmomi package on installer VM.

1. Login with a non-root user on installer command prompt.
2. Use existing datacenter, data cluster and minimum three vSphere6.7 hosts.

* DRS should be enabled on cluster.

Refer **APPENDIX C** to create Data Center, Data Cluster and vSphere hosts.

* “vMotion” should be enabled on all vSphere hosts to effect vSphere rules where VMs will be migrated from one host to another host. Please follow procedure to enable “vMotion” for all vSphere hosts.
* Click <vSphere host> 🡪 ”Configure” 🡪 “Networking” 🡪 “VMkernel adapters” 🡪 “Management Network” 🡪 “Edit” and enable option as shown below.



**Repository:**

The repository https://github.com/hewlettpackard/hpe-solutions-openshift which supports deploying VMs & applying vSphere rules as well. The installer should have this repository and cloning process as follows on the installer VM

# cd /etc/ansible

# git clone [https://github.com/hewlettpackard/hpe-solutions-openshift](https://github.com/hewlettpackard/hpe-solutions-openshift/).git

**Deploying management VMs:**

Deploying VMs on vSphere hosts.

Navigate to the below path.

# cd /etc/ansible/hpe-solutions-openshift/synergy/scalable/vsphere/virtual\_nodes

*Note: Replace your variable value within the triangular (<>) brackets.*

Check and modify the below mentioned files:

* roles/deploy\_vm/vars/main.yml

*# vars file for deploy\_vm*

*# Name of the Datacenter*

*datacenter\_name: <data\_center\_name>*

*# Name of the compute clusters with the ESXi hosts for Management VMs*

*management\_cluster\_name: <data\_cluster\_name>*

*# Name of the Datastore to store the VMs*

*management\_datastore\_name: <datastore\_name>*

*# Name of the coreOS guest image*

*guest\_template: coreos64Guest*

*# Disk size in GB/GiB*

*bootstrap\_disk: 120*

*master\_disk: 120*

*lb\_disk: 50*

*# number of CPUs*

*bootstrap\_cpu: 4*

*master\_cpu: 4*

*lb\_cpu: 4*

*# Memory size in MB/MiB*

*bootstrap\_memory: 16400*

*master\_memory: 16400*

*lb\_memory: 16400*

*gateway: <replace\_with\_gateway\_ip>*

*dns\_server: <replace\_with\_dns\_server\_ip>*

*# name of the master, bootstrap and lb nodes < short names, not the FQDN >*

*bootstrap01\_name: <bootstrap01\_fqdn\_host\_name>*

*master01\_name: <master01\_fqdn\_host\_name>*

*master02\_name: <master02\_fqdn\_host\_name>*

*master03\_name: <master03\_fqdn\_host\_name>*

*lb01\_name: <lb01\_fqdn\_host\_name>*

*domain\_name: "<sub\_domain\_fqdn>"*

*# Network names for the datacenter/management*

*datacenter\_network\_name: "<network\_name>"*

*# vSphere affinity & anti-affinity rules*

*affinity\_rule\_name: "vsphere-anti-affinty-rule"*

*anti\_affinity\_rule\_name: "vsphere-affinty-rule"*

* roles/pre\_vcenter/tests/test.yml

*---*

*- hosts: localhost*

*remote\_user: <not\_root\_user>*

*becomes: true*

*roles:*

*- deploy-vm*

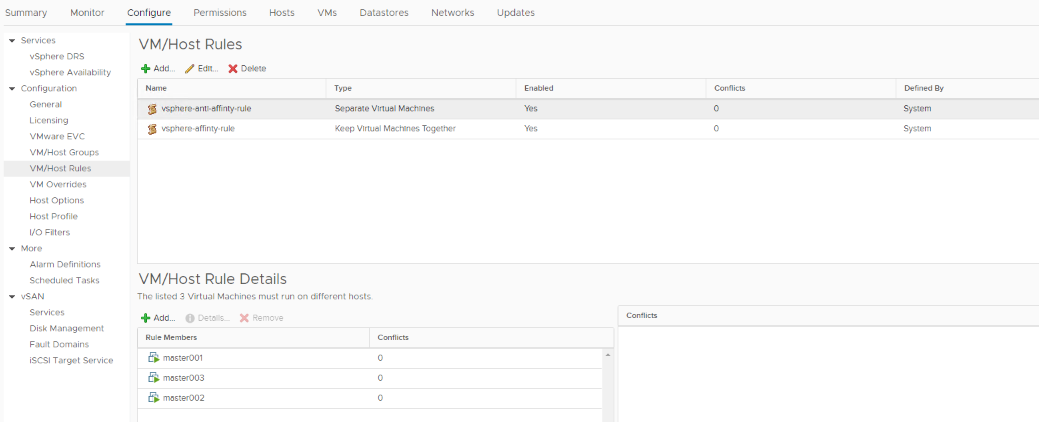
***NOTE: Check secret.yml and change variable values accordingly.***

# **ansible-playbook playbooks/deploy\_vm.yml --ask-vault-pass**

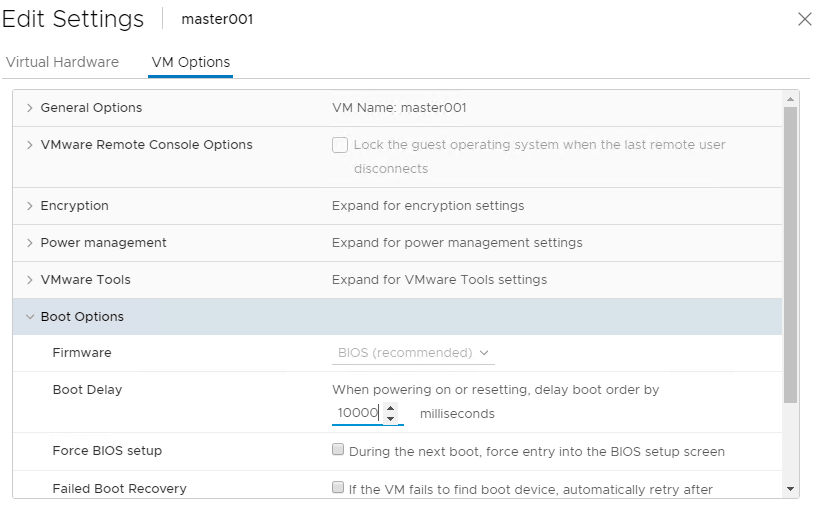
* deploy\_vm.yml – This play will deploy all Master, haproxy, bootstrap VMs with dummy guest template on hosts based on affinity and anti-affinity rules. Here, all master VMs will be deployed on different hosts whereas bootstrap & haproxy VMs will be deployed on any single host.

**NOTE:**

1. **Please wait for some time to affect vSphere rules on VMs which are deployed under vSphere hosts and can see the below “VM/HOST Rules” will be visible in vCenter. Navigation path: vCenter** 🡪 **<Datacenter>** 🡪 **<Datacluster>** 🡪 **Configure** 🡪 **“VM/Host Rules”.**
2. **Three master nodes are the part of “vsphere-anti-affinity-rule” where each master should reside in different vSphere hosts.**
3. **The “haproxy” & “bootstrap” nodes are the part of “vsphere-affinity-rule” where “haproxy” & “bootstrap” VMs should reside any of three vSphere hosts.**



Navigate “Click on VM 🡪 Actions 🡪 Edit Settings 🡪 VM Options 🡪 Boot Options 🡪 Boot Delay” = 10000. Apply the same for other VMs.



* Create DNS entries in my domain (your-sub-domain.twentynet.local) with the corresponding hostnames & IP address for the following as shown in the figure. Make sure the special characters like “-”, “\*” etc is intact and updated as is:
* Masters [hostname and alias for etcd]
* HAProxy [hostnames - haproxy, \*, api, api-int]
* Bootstrap
* SRV record entries under “\_tcp” sub-domain in DNS server.

1. Provide following details in PXE server:

Share the below mentioned VM’s MAC address, IP and FQDN

* + - 1. Masters
      2. Bootstrap
      3. Haproxy

1. Create a configuration file i.e., install-config.yaml under /root/ocp43 directory and file content as shown below:

**NOTE: /root/ocp43 is a customized directory on installer.**

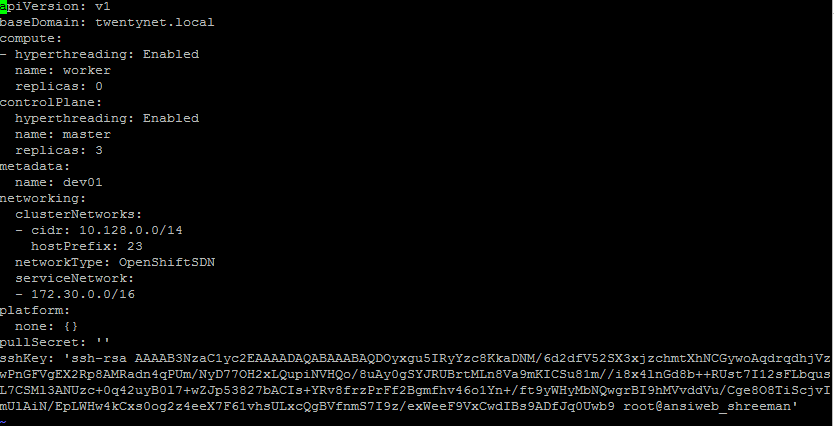
1. Generate SSH key and place in installer-config.yaml file.
   * + 1. Generate the SSH key with following command

*# ssh-keygen*

* + - 1. SSH key is present at below location. Copy that to the install-config.yaml

# *cat /root/.ssh/id\_rsa.pub*

* 1. Pull secret from the Red Hat Subscription page - <https://cloud.redhat.com/openshift/install/metal/user-provisioned>



*apiVersion: v1*

*baseDomain: <domain\_what\_you\_are\_using>*

*compute:*

*- hyperthreading: Enabled*

*name: worker*

*replicas: <num\_of\_worker\_nodes>*

*controlPlane:*

*hyperthreading: Enabled*

*name: master*

*replicas: <num\_of\_master\_nodes>*

*metadata:*

*name: <sub\_domain\_under\_baseDomain>*

*networking:*

*clusterNetworks:*

*- cidr: 10.128.0.0/14*

*hostPrefix: 23*

*networkType: OpenShiftSDN*

*serviceNetwork:*

*- 172.30.0.0/16*

*platform:*

*none: {}*

*pullSecret: ''*

*sshKey: ''*

**WARNING:**

**Pull secret valid for 24hrs only, if the pull secret is expired or has passed 24 hours of existence, the ignition and auth files created with this pull secret needs to be erased or the directory containing the same should be cleaned.**

1. Download the following files from https://mirror.open shift.com/pub/openshift-v4/clients/ocp/latest/ into /root/ocp43 directory on installer VM and unpack them using (*tar –xzvf <filename.tar.gz>*) command.

* openshift-client-linux-4.3.x.tar.gz
* openshift-install-linux-4.3.x.tar.gz

1. Generate the manifests by executing the following commands in the installer VM

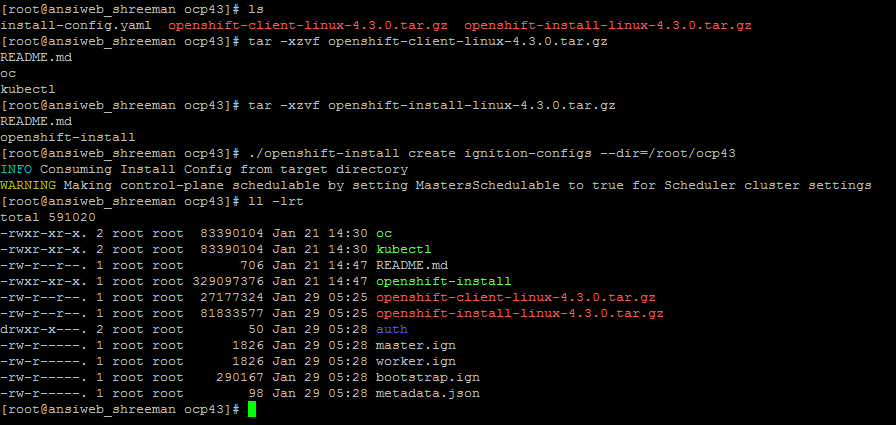
*# cd /root/ocp43*

*# ./openshift-install create manifests --dir=/root/ocp43*

8) Generate ignition files executing the command in the installer VM

*# cd /root/ocp43*

*# ./openshift-install create ignition-configs --dir=/root/ocp43*



**NOTE:**

**Have a backup of the install-config.yml since this file will be destroyed once ignition file is generated.**

1. Copy ignition files into the WEB/FTP server (in our case PXE server) using following command:

**In PXE server**

*# cd /var/www/html/ or appropriate location for FTP server (i.e., /var/ftp/pub)*

*# mkdir <your dir name>*

**In installer server**

*# cd <location of ignition files> or cd /root/ocp43*

# scp bootstrap.ign worker.ign master.ign [root@<pxe\_server\_ip>:/var/ftp/pub/<respective folder>/](mailto:root@20.0.16.251:/var/ftp/pub/%3crespective%20folder%3e/)./

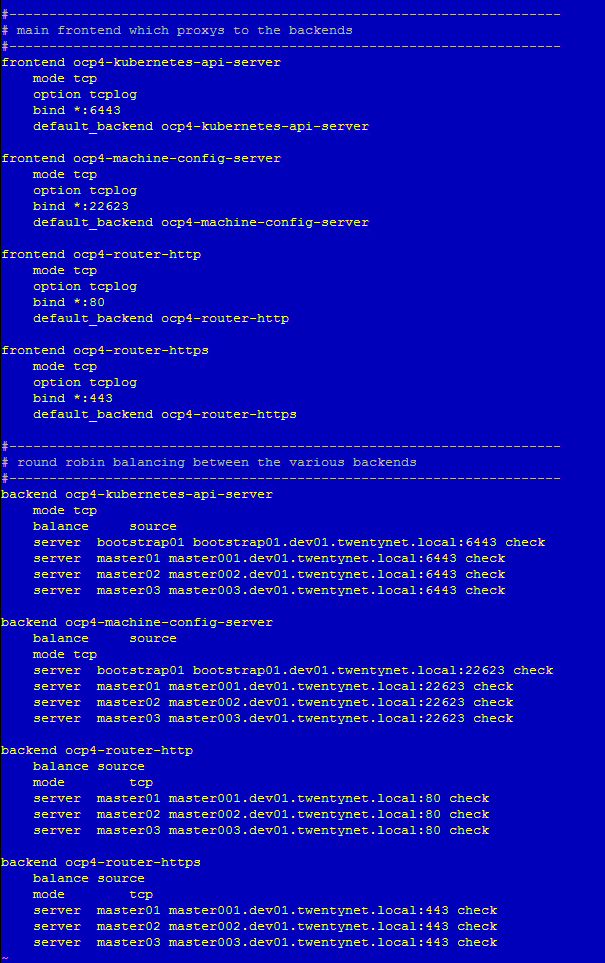
# ssh root@<pxe\_server\_ip> systemctl restart vsftpd

1. Choose centos image while re-booting haproxy VM and make sure boot should be properly done.

10.1) Install haproxy package on haproxy VM using “*yum –y install haproxy*” command.

10.2) Configure “/etc/haproxy/haproxy.cfg” file as shown below:

**NOTE: Ensure your master names and Master FQDNs should be proper.**



10.3) Start, enable and check “haproxy” service using the below commands:

*# systemctl start haproxy.service*

*# systemctl enable haproxy.service*

*# systemctl status haproxy.service*

10.4) If haproxy service is failed, follow the below procedure:

* Assign “permissive” to SELINUX in “/etc/sysconfig/selinux” file (SELINUX=permissive).
* Reboot haproxy VM and check “haproxy” service status.
* Haproxy service should be **active (running)**.

*# reboot*

*# systemctl status haproxy.service (Once haproxy VM is up).*

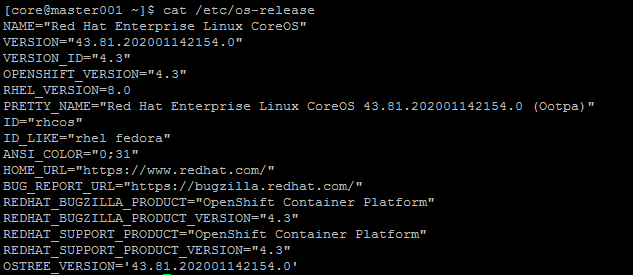
1. Choose your bootstrap image for Bootstrap VM and your master image for Master VMs while re-booting Bootstrap and Master VMs and make these VMs are properly booted.
2. **Bootstrap and all Master VMs are installed with CoreOS.**

* Login to all Master VMs from installer VM and file below command for cross checking CoreOS version:

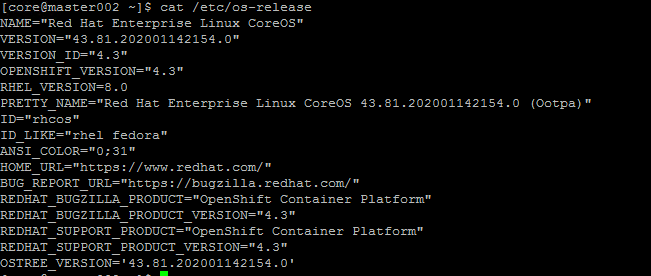
*# ssh core@<master\_vm\_ip>*

*# cat /etc/os-release*

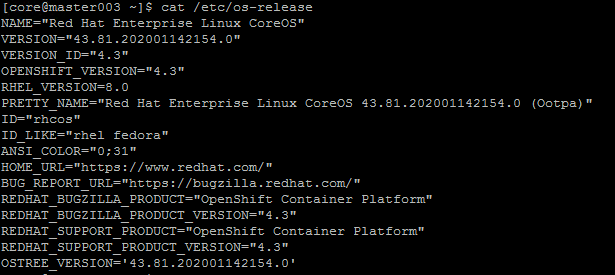
**12.1) Master1 OS Version:**



**12.2) Master2 OS Version:**

****

**12.3) Master3 OS Version:**

****

**Validation:**

1) Verify by logging into the bootstrap server from the **installer server** using the following command also verify the journal logs.

# ssh core@<bootstrap-fqdn>

# journalctl –b –f –u bootkube.service

NOTE: Since the default password is unknown for the CoreOS machines, there is no other way to login to the CoreOS other than SSH through the installer VM since we shared the SSH key.

1. Verify by logging into the servers from the installer server using the following

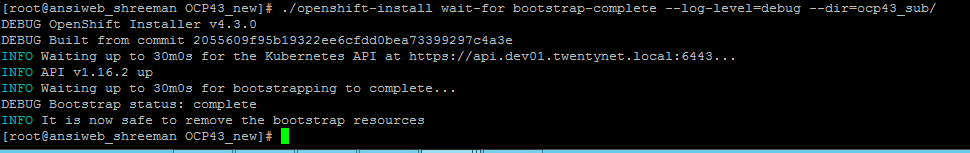
# ssh core@<server-fqdn>

NOTE:It is essential that all the nodes such as Installer, HA proxy, Bootstrap, Masters, Workers etc are in the same time zone failing which the OCP installation will result in a failure due to mismatch of certificate or similar issues.

1. On installer: Execute the command from the installer server to bootstrap the nodes

*#cd /root/ocp43*

*# ./openshift-install wait-for bootstrap-complete --log-level=debug --dir=/root/ocp43*



**NOTE:** You can shut down or remove the bootstrap node now

1. Run the below command from the install directory to set the variable

# export KUBECONFIG=/root/ocp43/auth/kubeconfig

1. On installation: provide the storage (PV) for the registry or patch it as non-production cluster, Execute the below command to set the image registry storage to an empty directory.

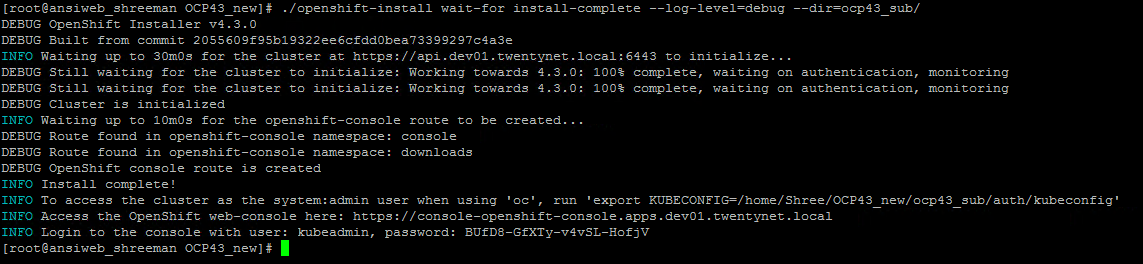
# *oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'*

1. Execute the below command from the installer VM to complete the OCP installation:

*# cd /root/ocp43*

*# ./openshift-install wait-for install-complete --log-level=debug --dir=/root/ocp43*

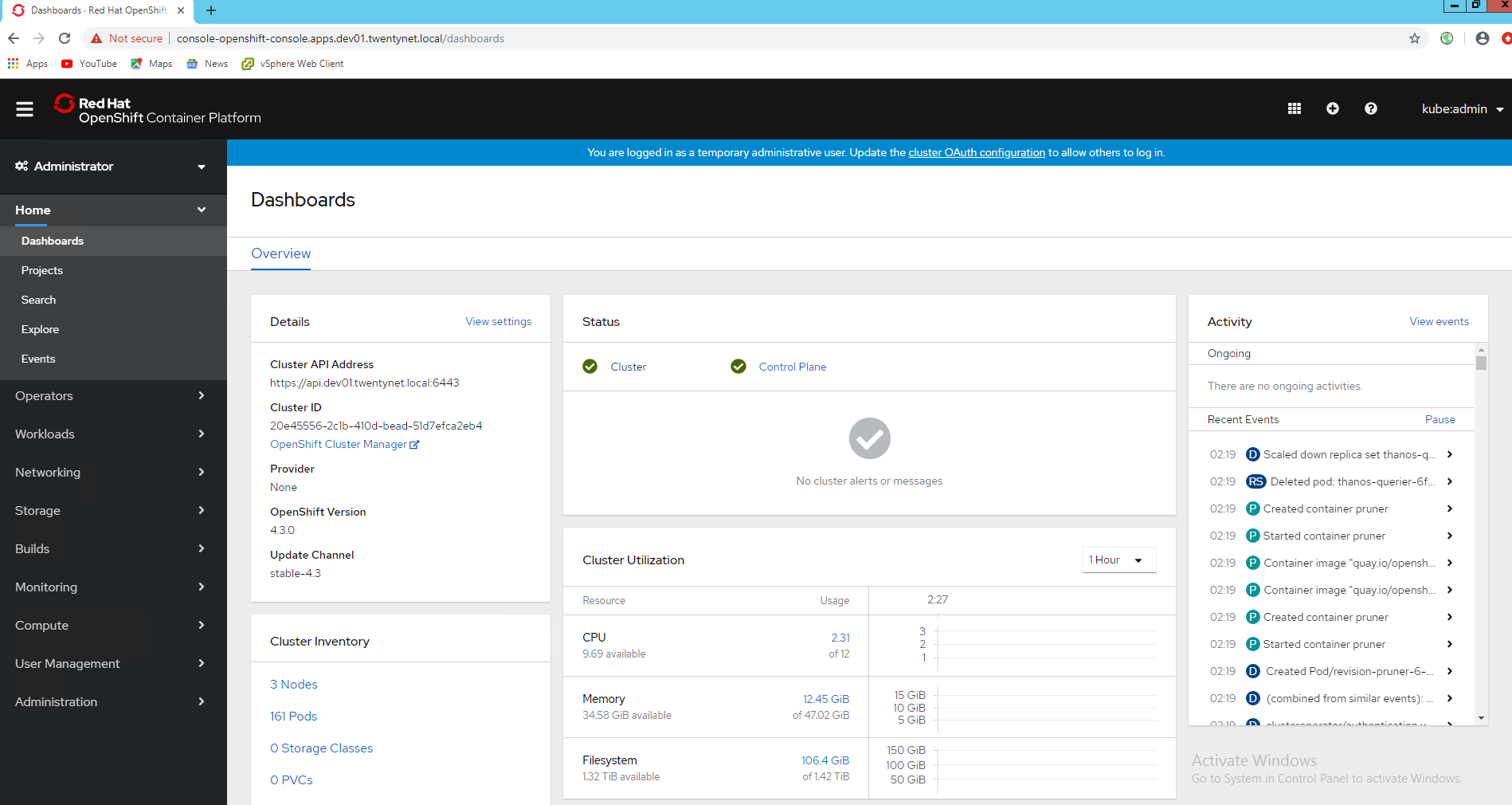
*Result* :

******

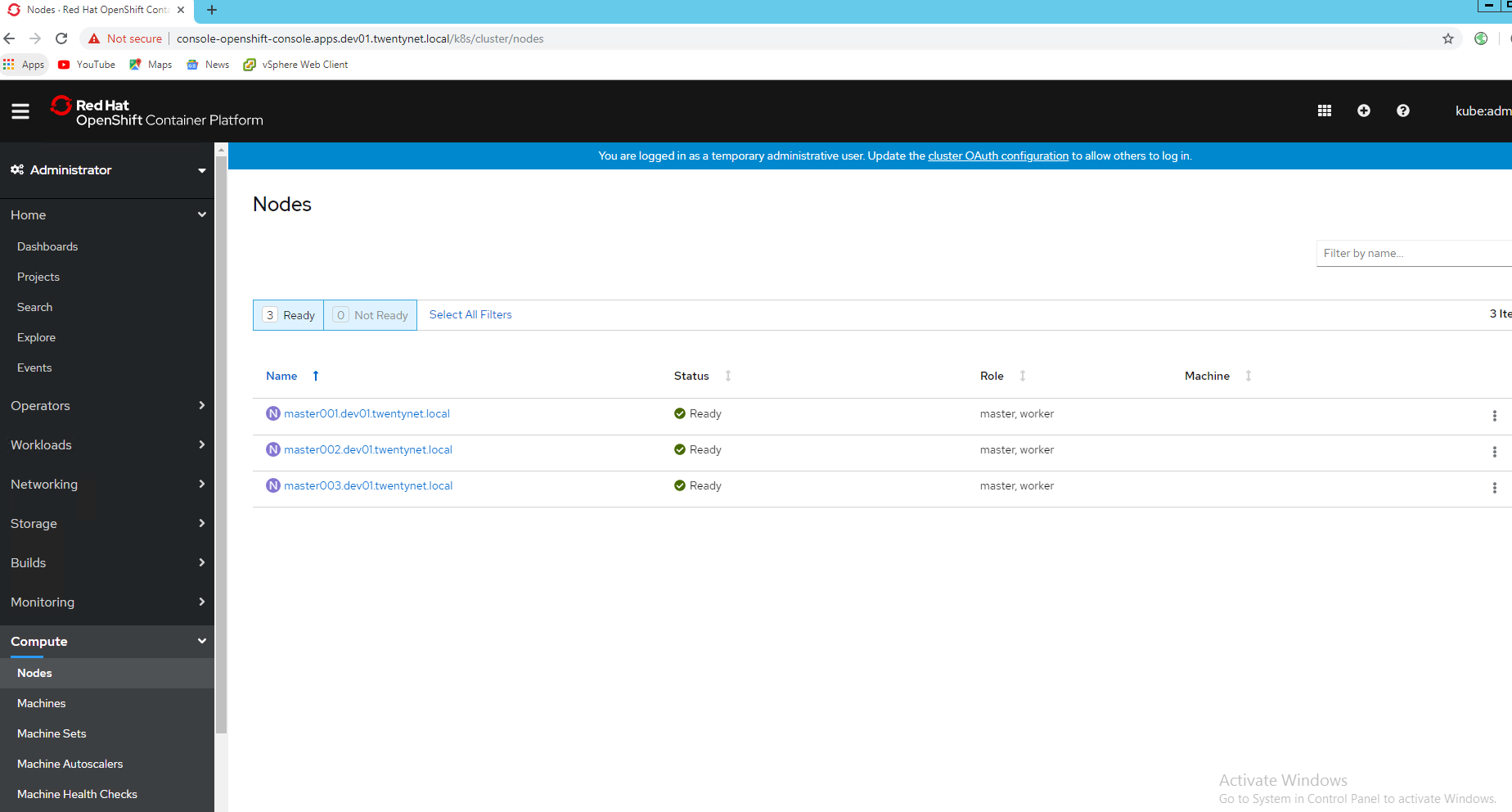
**NOTE:**

* Make a note of this OpenShift cluster URL, username and the password for future access to the OpenShift cluster.
* If in case the password is forgotten, navigate to the file auth/kubeadmin-password for viewing the password.

1. The OpenShift web console is shown below.



1. The final OpenShift web console master nodes view is shown below where validation is over.



***APPENDIX A:***

1. *Pyhton 3.6.x & ansible 2.9.x are required on installer VM.*

# yum -y install @development

# yum -y install rh-python36

# yum -y install rh-python36-numpy \

 rh-python36-scipy \

 rh-python36-python-tools \

 rh-python36-python-six

# scl enable rh-python36 bash

# python3 –V 🡪 python now also points to Python3

Python 3.6.x

$ mkdir ~/pydev

$ cd ~/pydev

  $ python3 -m venv py36-venv

$ source py36-venv/bin/activate

  (py36-venv) $ python3 -m pip install ansible==2.9.x OR pip install ansible==2.9.x

**APPENDIX B:**

Install git on installer VM.

$ yum –y install git

Install Python hpOneView

$ git clone <https://github.com/HewlettPackard/python-hpOneView.git>

$ cd python-hpOneView

$ pip install .

Install oneview-ansible

$ git clone <https://github.com/HewlettPackard/oneview-ansible.git>

$ cd oneview-ansible

$ export ANSIBLE\_LIBRARY=<path to oneview-ansible/library>

$ export ANSIBLE\_MODULE\_UTILS=<path to oneview-ansible/library/module\_utils>

Install PyVmomi should be installed on installer VM.

$ pip install PyVmomi

OR

$ yum -y install PyVmomi

**APPENDIX C:**

# cd /etc/Ansible/hpe-solutions-openshift/synergy/scalable/vsphere/vcenter

Check and modify the below mentioned variable file:

* roles/prepare\_vcenter/vars/main.yml

*# Custom name for datacenter to be created*

*datacenter\_name: <datacenter\_name>*

*# custom name of the compute clusters with the ESXi hosts for Management VMs and worker VMs*

*management\_cluster\_name: < management\_cluster\_name >*

*# hostname or IP address of the vsphere hosts*

*vsphere\_host\_01: <ESXi1\_IP>*

*vsphere\_host\_02: <ESXi2\_IP>*

*vsphere\_host\_03: <ESXi3\_IP>*

* roles/pre\_vcenter/tests/test.yml

*---*

*- hosts: localhost*

*remote\_user: <not\_root\_user>*

*becomes: true*

*roles:*

*- prepare\_vcsa*

*#* ***ansible-playbook playbooks/prepare\_vcenter.yml --ask-vault-pass***

prepare\_vcenter.yml – It will create datacenter, datacluster and vSphere hosts under cluster in vCenter.