

# PCRF Replacement of OSD-Compute UCS 240M4

## Contents

[Introduction](#)

[Background Information](#)

[Healthcheck](#)

[Backup](#)

[Identify the VMs Hosted in the OSD-Compute Node](#)

[Graceful Power Off](#)

[Migrate ESC to Standby Mode](#)

[Osd-Compute Node Deletion](#)

[Delete from Overcloud](#)

[Delete Osd-Compute Node from the Service List](#)

[Delete Neutron Agents](#)

[Delete from the Nova and Ironic Database](#)

[Install the New Compute Node](#)

[Add the new OSD-Compute node to the Overcloud](#)

[Restore the VMs](#)

[Addition to Nova Aggregate List](#)

[Recovery of ESC VM](#)

## Introduction

This document describes the steps required to replace a faulty osd-compute server in an Ultra-M setup that hosts Cisco Policy Suite (CPS) Virtual Network Functions (VNFs).

## Background Information

This document is intended for the Cisco personnel familiar with Cisco Ultra-M platform and it details the steps required to be carried out at OpenStack and CPS VNF level at the time of the OSD-Compute Server Replacement.

**Note:** Ultra M 5.1.x release is considered in order to define the procedures in this document.

## Healthcheck

Before you replace a Osd-Compute node, it is important to check the current state of your Red Hat OpenStack Platform environment. It is recommended you check the current state in order to avoid complications when the Compute replacement process is on.

From OSPD

```
[root@director ~]$ su - stack
[stack@director ~]$ cd ansible
[stack@director ansible]$ ansible-playbook -i inventory-new openstack_verify.yml -e
platform=pcrf
```

Step 1. Verify health of system from ultram-health report which is generated in every fifteen minutes.

```
[stack@director ~]# cd /var/log/cisco/ultram-health
Check file ultram_health_os.report.
```

The only services should show as **XXX** status are **neutron-sriov-nic-agent.service**.

Step 2. Check whether **rabbitmq** runs for all controllers, which in turn runs from OSPD.

```
[stack@director ~]# for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo rabbitmqctl
eval 'rabbit_diagnostics:maybe_stuck().'" ) & done
```

Step 3. Verify stonith is enabled.

```
[stack@director ~]# sudo pcs property show stonith-enabled
```

For all Controllers verify PCS status

- All controller nodes are **Started** under haproxy-clone
- All controller nodes are **Master** under galera
- All controller nodes are **Started** under Rabbitmq
- 1 controller node is **Master** and 2 **Slaves** under redis

From OSPD

```
[stack@director ~]$ for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo pcs status"
) ;done
```

Step 4. Verify all openstack services are Active, from OSPD run this command:

```
[stack@director ~]# sudo systemctl list-units "openstack*" "neutron*" "openvswitch*"
```

Step 5. Verify CEPH status is HEALTH\_OK for Controllers.

```
[stack@director ~]# for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo ceph -s" )
;done
```

Step 6. Verify OpenStack component logs. Look for any error:

Neutron:

```
[stack@director ~]# sudo tail -n 20 /var/log/neutron/{dhcp-agent,13-agent,metadata-
agent,openvswitch-agent,server}.log
```

Cinder:

```
[stack@director ~]# sudo tail -n 20 /var/log/cinder/{api,scheduler,volume}.log
```

Glance:

```
[stack@director ~]# sudo tail -n 20 /var/log/glance/{api,registry}.log
```

Step 7. From OSPD perform these verifications for API.

```
[stack@director ~]$ source <overcloudrc>
```

```
[stack@director ~]$ nova list
```

```
[stack@director ~]$ glance image-list
```

```
[stack@director ~]$ cinder list
```

```
[stack@director ~]$ neutron net-list
```

Step 8. Verify the health of services.

Every service status should be "up":

```
[stack@director ~]$ nova service-list
```

Every service status should be " :-)":

```
[stack@director ~]$ neutron agent-list
```

Every service status should be "up":

```
[stack@director ~]$ cinder service-list
```

## Backup

In case of recovery, Cisco recommends to take a backup of the OSPD database with the use of these steps.

Step 1. Take Mysql dump.

```
[root@director ~]# mysqldump --opt --all-databases > /root/undercloud-all-databases.sql
[root@director ~]# tar --xattrs -czf undercloud-backup-`date +%F`.tar.gz /root/undercloud-all-
databases.sql
/etc/my.cnf.d/server.cnf /var/lib/glance/images /srv/node /home/stack
tar: Removing leading `/' from member names
```

This process ensures that a node can be replaced without affecting the availability of any instances.

Step 2. To back up CPS VMs from Cluster Manager VM:

```
[root@CM ~]# config_br.py -a export --all /mnt/backup/CPS_backup_$(date +%Y-%m-%d).tar.gz
```

or

```
[root@CM ~]# config_br.py -a export --mongo-all --svn --etc --grafanadb --auth-htpasswd --
haproxy /mnt/backup/$(hostname)_backup_all_$(date +%Y-%m-%d).tar.gz
```

## Identify the VMs Hosted in the OSD-Compute Node

Identify the VMs that are hosted on the compute server:

Step 1. The compute server contains Elastic Services Controller (ESC).

```
[stack@director ~]$ nova list --field name,host,networks | grep osd-compute-1
| 50fd1094-9c0a-4269-b27b-cab74708e40c | esc | pod1-osd-compute-0.localdomain
| tb1-orch=172.16.180.6; tb1-mgmt=172.16.181.3
```

**Note:** In the output shown here, the first column corresponds to the Universally Unique Identifier (UUID), the second column is the VM name and the third column is the hostname where the VM is present. The parameters from this output will be used in subsequent sections.

**Note:** If OSD-compute node to be replaced is completely down & not accessible, then proceed to section titled "Remove the Osd-Compute Node from Nova Aggregate List". Otherwise, proceed from the next section.

Step 2. Verify that CEPH has available capacity to allow a single OSD server to be removed.

```
[root@pod1-osd-compute-0 ~]# sudo ceph df
```

GLOBAL:

SIZE	AVAIL	RAW USED	%RAW USED
<b>13393G</b>	<b>11804G</b>	<b>1589G</b>	<b>11.87</b>

POOLS:

NAME	ID	USED	%USED	MAX AVAIL	OBJECTS
rbd	0	0	0	3876G	0
metrics	1	4157M	0.10	3876G	215385
images	2	6731M	0.17	3876G	897
backups	3	0	0	3876G	0
volumes	4	399G	9.34	3876G	102373
vms	5	122G	3.06	3876G	31863

Step 3. Verify ceph osd tree status are up on the osd-compute server.

```
[heat-admin@pod1-osd-compute-0 ~]$ sudo ceph osd tree
```

ID	WEIGHT	TYPE	NAME	UP/DOWN	REWEIGHT	PRIMARY-AFFINITY
-1	13.07996	root	default			
-2	4.35999	host	pod1-osd-compute-0			
0	1.09000		osd.0	up	1.00000	1.00000

```

3 1.09000      osd.3                up 1.00000          1.00000
6 1.09000      osd.6                up 1.00000          1.00000
9 1.09000      osd.9                up 1.00000          1.00000
-3 4.35999      host pod1-osd-compute-2
1 1.09000      osd.1                up 1.00000          1.00000
4 1.09000      osd.4                up 1.00000          1.00000
7 1.09000      osd.7                up 1.00000          1.00000
10 1.09000     osd.10               up 1.00000          1.00000
-4 4.35999      host pod1-osd-compute-1
2 1.09000      osd.2                up 1.00000          1.00000
5 1.09000      osd.5                up 1.00000          1.00000
8 1.09000      osd.8                up 1.00000          1.00000
11 1.09000     osd.11               up 1.00000          1.00000

```

**Step 4. CEPH processes are active on the osd-compute server.**

```
[root@pod1-osd-compute-0 ~]# systemctl list-units *ceph*
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
var-lib-ceph-osd-ceph\x2d11.mount	loaded	active	mounted	/var/lib/ceph/osd/ceph-11
var-lib-ceph-osd-ceph\x2d2.mount	loaded	active	mounted	/var/lib/ceph/osd/ceph-2
var-lib-ceph-osd-ceph\x2d5.mount	loaded	active	mounted	/var/lib/ceph/osd/ceph-5
var-lib-ceph-osd-ceph\x2d8.mount	loaded	active	mounted	/var/lib/ceph/osd/ceph-8
ceph-osd@11.service	loaded	active	running	Ceph object storage daemon
ceph-osd@2.service	loaded	active	running	Ceph object storage daemon
ceph-osd@5.service	loaded	active	running	Ceph object storage daemon
ceph-osd@8.service	loaded	active	running	Ceph object storage daemon
system-ceph\x2ddisk.slice	loaded	active	active	system-ceph\x2ddisk.slice
system-ceph\x2dosd.slice	loaded	active	active	system-ceph\x2dosd.slice
ceph-mon.target	loaded	active	active	ceph target allowing to start/stop all
ceph-mon@.service				instances at once
ceph-osd.target	loaded	active	active	ceph target allowing to start/stop all
ceph-osd@.service				instances at once
ceph-radosgw.target	loaded	active	active	ceph target allowing to start/stop all
ceph-radosgw@.service				instances at once
ceph.target	loaded	active	active	ceph target allowing to start/stop all

ceph\*@.service instances at once

**Step 5. Disable and stop each ceph instance and remove each instance from osd and unmount the directory. Repeat for each ceph instance.**

```
[root@pod1-osd-compute-0 ~]# systemctl disable ceph-osd@11
```

```
[root@pod1-osd-compute-0 ~]# systemctl stop ceph-osd@11
```

```
[root@pod1-osd-compute-0 ~]# ceph osd out 11
```

marked out osd.11.

```
[root@pod1-osd-compute-0 ~]# ceph osd crush remove osd.11
```

removed item id 11 name 'osd.11' from crush map

```
[root@pod1-osd-compute-0 ~]# ceph auth del osd.11
```

updated

```
[root@pod1-osd-compute-0 ~]# ceph osd rm 11
```

removed osd.11

```
[root@pod1-osd-compute-0 ~]# umount /var/lib/ceph/osd/ceph-11
```

```
[root@pod1-osd-compute-0 ~]# rm -rf /var/lib/ceph/osd/ceph-11
```

(or)

**Step 6. Clean.sh** script can be used for doing the above task at once.

```
[heat-admin@pod1-osd-compute-0 ~]$ sudo ls /var/lib/ceph/osd
```

ceph-11 ceph-3 ceph-6 ceph-8

```
[heat-admin@pod1-osd-compute-0 ~]$ /bin/sh clean.sh
```

```
[heat-admin@pod1-osd-compute-0 ~]$ cat clean.sh
```

```
#!/bin/sh
```

```

set -x

CEPH=`sudo ls /var/lib/ceph/osd`

for c in $CEPH
do
    i=`echo $c |cut -d'-' -f2`

    sudo systemctl disable ceph-osd@$i || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo systemctl stop ceph-osd@$i || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo ceph osd out $i || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo ceph osd crush remove osd.$i || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo ceph auth del osd.$i || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo ceph osd rm $i || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo umount /var/lib/ceph/osd/$c || (echo "error rc:$?"; exit 1)

    sleep 2

    sudo rm -rf /var/lib/ceph/osd/$c || (echo "error rc:$?"; exit 1)

    sleep 2
done

sudo ceph osd tree

```

After all OSD processes have been migrated/deleted, the node can be removed from the overcloud.

**Note:** When CEPH is removed, VNF HD RAID goes in the to Degraded state but hd-disk must still be accessible.

## Graceful Power Off

### Migrate ESC to Standby Mode

Step 1. Login to the ESC hosted in the compute node and check if it is in the master state. If yes, switch the ESC to standby mode.

```
[admin@esc esc-cli]$ escaadm status
0 ESC status=0 ESC Master Healthy
```

```
[admin@esc ~]$ sudo service keepalived stop
Stopping keepalived: [ OK ]
```

```
[admin@esc ~]$ escaadm status
1 ESC status=0 In SWITCHING_TO_STOP state. Please check status after a while.
```

```
[admin@esc ~]$ sudo reboot
Broadcast message from admin@vnf1-esc-esc-0.novalocal
(/dev/pts/0) at 13:32 ...
The system is going down for reboot NOW!
```

## Step 2. Remove the Osd-Compute Node from Nova Aggregate List.

- List the nova aggregates and identify the aggregate that corresponds to the compute server based on the VNF hosted by it. Usually, it would be of the format <VNFNAME>-EM-MGMT<X> and <VNFNAME>-CF-MGMT<X>

```
[stack@director ~]$ nova aggregate-list
+-----+-----+-----+
| Id | Name | Availability Zone |
+-----+-----+-----+
| 3 | esc1 | AZ-esc1 |
| 6 | esc2 | AZ-esc2 |
| 9 | aaa | AZ-aaa |
+-----+-----+-----+
```

In our case, the osd-compute server belongs to esc1. So, the aggregates that correspond would be **esc1**

## Step 3. Remove the osd-compute node from the aggregate identified.

```
nova aggregate-remove-host <Aggregate> <Host>
```

```
[stack@director ~]$ nova aggregate-remove-host esc1 pod1-osd-compute-0.localdomain
```

Step 4. Verify if the osd-compute node has been removed from the aggregates. Now, ensure that the Host is not listed under the aggregates.

```
nova aggregate-show <aggregate-name>
```

```
[stack@director ~]$ nova aggregate-show esc1
[stack@director ~]$
```

# Osd-Compute Node Deletion

The steps mentioned in this section are common irrespective of the VMs hosted in the compute node.

## Delete from Overcloud

Step 1. Create a script file named delete\_node.sh with the contents as shown. Ensure that the templates mentioned are the same as the ones used in the **deploy.sh** script used for the stack



## deployment.

### **delete\_node.sh**

```
openstack overcloud node delete --templates -e /usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml -e /home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e /home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml --stack <stack-name> <UUID>
```

```
[stack@director ~]$ source stackrc
[stack@director ~]$ /bin/sh delete_node.sh
+ openstack overcloud node delete --templates -e /usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml -e /home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e /home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml --stack pod1 49ac5f22-469e-4b84-badc-031083db0533
Deleting the following nodes from stack pod1:
- 49ac5f22-469e-4b84-badc-031083db0533
Started Mistral Workflow. Execution ID: 4ab4508a-c1d5-4e48-9b95-ad9a5baa20ae
```

```
real    0m52.078s
user    0m0.383s
sys     0m0.086s
```

Step 2. Wait for the OpenStack stack operation to move to the COMPLETE state.

```
[stack@director ~]$ openstack stack list
```

ID	Stack Name	Stack Status	Creation Time	Updated Time
5df68458-095d-43bd-a8c4-033e68ba79a0	pod1	UPDATE_COMPLETE	2018-05-08T21:30:06Z	2018-05-08T20:42:48Z

## Delete Osd-Compute Node from the Service List

Delete the compute service from the service list.

```
[stack@director ~]$ source corerc
[stack@director ~]$ openstack compute service list | grep osd-compute-0
```

404	nova-compute	pod1-osd-compute-0.localdomain	nova	enabled	up	2018-05-08T18:40:56.000000
-----	--------------	--------------------------------	------	---------	----	----------------------------

```
openstack compute service delete <ID>
```

```
[stack@director ~]$ openstack compute service delete 404
```

## Delete Neutron Agents

Delete the old associated neutron agent and open vswitch agent for the compute server.

```
[stack@director ~]$ openstack network agent list | grep osd-compute-0
| c3ee92ba-aa23-480c-ac81-d3d8d01dcc03 | Open vSwitch agent | pod1-osd-compute-0.localdomain
| None | False | UP | neutron-openvswitch-agent |
| ec19cb01-abbb-4773-8397-8739d9b0a349 | NIC Switch agent | pod1-osd-compute-0.localdomain
| None | False | UP | neutron-sriov-nic-agent |
```

```
openstack network agent delete <ID>
```

```
[stack@director ~]$ openstack network agent delete c3ee92ba-aa23-480c-ac81-d3d8d01dcc03
[stack@director ~]$ openstack network agent delete ec19cb01-abbb-4773-8397-8739d9b0a349
```

## Delete from the Nova and Ironic Database

Delete a node from nova list along with the ironic database, and then verify it.

```
[stack@director ~]$ source stackrc
```

```
[stack@al01-pod1-ospd ~]$ nova list | grep osd-compute-0
| c2cfa4d6-9c88-4ba0-9970-857d1a18d02c | pod1-osd-compute-0 | ACTIVE | - | Running
| ctlplane=192.200.0.114 |
```

```
[stack@al01-pod1-ospd ~]$ nova delete c2cfa4d6-9c88-4ba0-9970-857d1a18d02c
```

```
nova show <compute-node> | grep hypervisor
```

```
[stack@director ~]$ nova show pod1-osd-compute-0 | grep hypervisor
| OS-EXT-SRV-ATTR:hypervisor_hostname | 4ab21917-32fa-43a6-9260-02538b5c7a5a
```

```
ironic node-delete <ID>
```

```
[stack@director ~]$ ironic node-delete 4ab21917-32fa-43a6-9260-02538b5c7a5a
[stack@director ~]$ ironic node-list (node delete must not be listed now)
```

## Install the New Compute Node

The steps in order to install a new UCS C240 M4 server and the initial setup steps can be referred from: [Cisco UCS C240 M4 Server Installation and Service Guide](#)

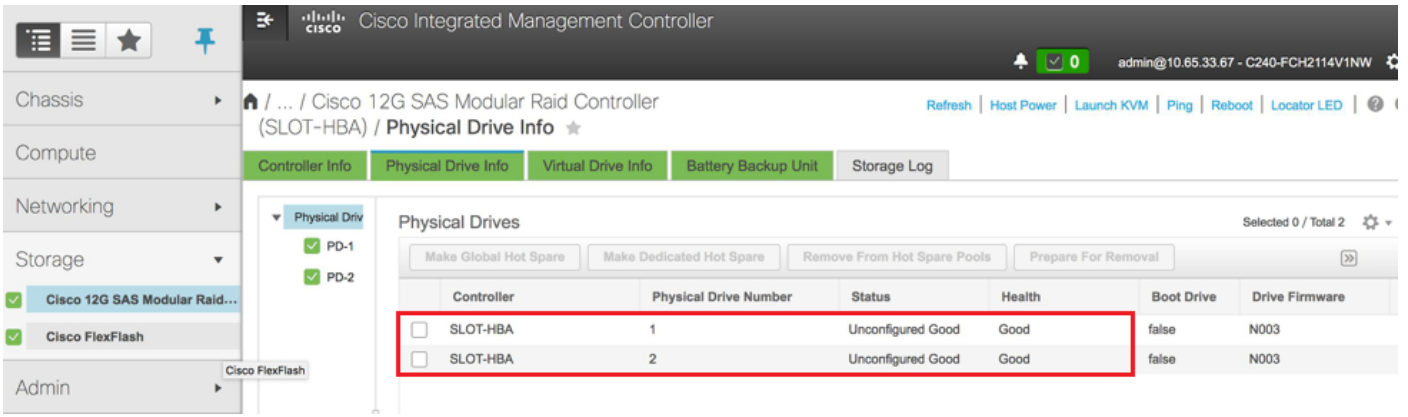
Step 1. After the installation of the server, insert the hard disks in the respective slots as the old server.

Step 2. Login to server with the use of the CIMC IP.

Step 3. Perform BIOS upgrade if the firmware is not as per the recommended version used previously. Steps for BIOS upgrade are given here: [Cisco UCS C-Series Rack-Mount Server BIOS Upgrade Guide](#)

Step 4. Verify the status of Physical Drives. It must be **Unconimaged Good**.

Step 5. Create a virtual drive from the physical drives with RAID Level 1.



Step 6. Navigate to storage section and select the Cisco 12G Sas Modular Raid Controller and verify the status and health of the raid controller as shown in the image.

**Note:** The above image is for illustration purpose only, in actual OSD-Compute CIMC you see seven physical drives in slots [1,2,3,7,8,9,10] in unconfigured Good state as no Virtual Drives are created from them.

Cisco Integrated Management Controller  
Create Virtual Drive from Unused Physical Drives

RAID Level: 1  Enable Full Disk Encryption

Create Drive Groups

Physical Drives Selected 2 / Total 2

ID	Size(MB)	Model	Interface	Type
<input checked="" type="checkbox"/> 1	1906394 MB	SEAGA...	HDD	SAS
<input checked="" type="checkbox"/> 2	1906394 MB	SEAGA...	HDD	SAS

Drive Groups

No data available

Virtual Drive Properties

Name: RAID1  
 Access Policy: Read Write  
 Read Policy: No Read Ahead  
 Cache Policy: Direct IO  
 Disk Cache Policy: Unchanged  
 Write Policy: Write Through  
 Strip Size (MB): 64k  
 Size:  MB

Cisco Integrated Management Controller  
Create Virtual Drive from Unused Physical Drives

RAID Level: 1  Enable Full Disk Encryption

Create Drive Groups

Physical Drives Selected 0 / Total 0

No data available

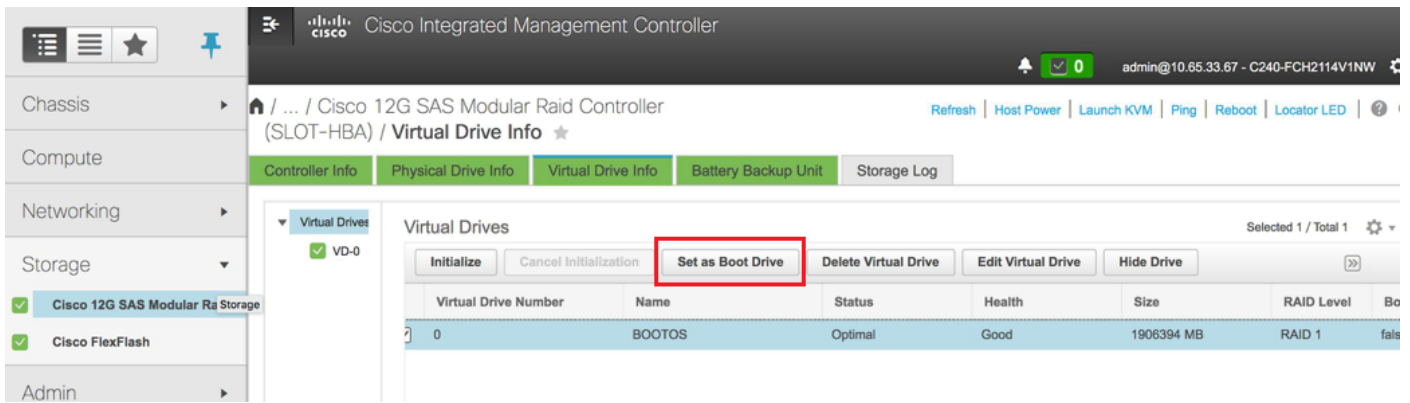
Drive Groups

DG [1,2]

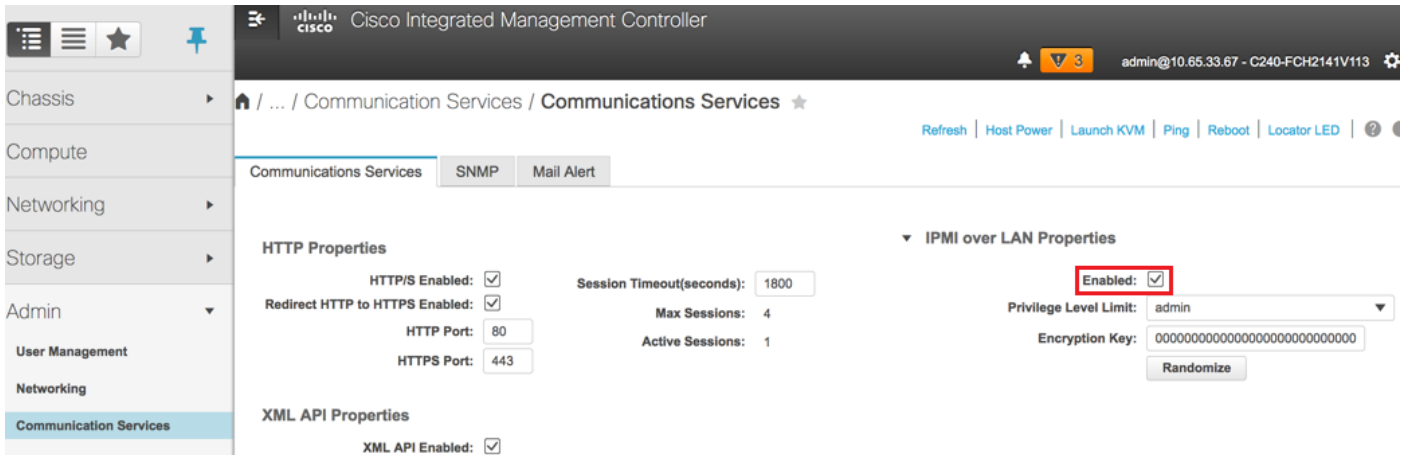
Virtual Drive Properties

Name:   
 Access Policy: Read Write  
 Read Policy: No Read Ahead  
 Cache Policy: Direct IO  
 Disk Cache Policy: Unchanged  
 Write Policy: Write Through  
 Strip Size (MB): 64k  
 Size: 1906394 MB

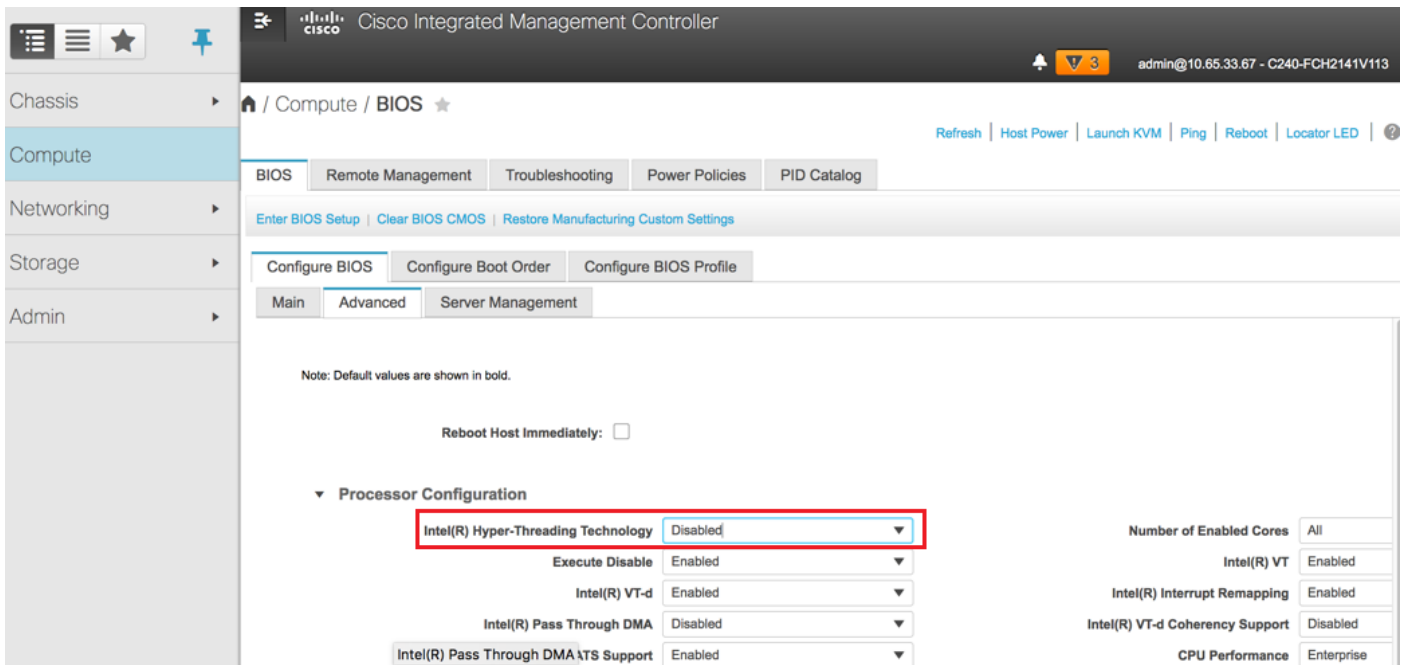
Step 7. Now create a Virtual drive from an unused physical drive from the controller info, under the **Cisco 12G SAS Modular Raid Controller**.



Step 8. Select the VD and configure **set as boot drive**.



Step 9. Enable IPMI over LAN from Communication services under Admin tab.



Step 10. Disable Hyper-Threading from the Advance BIOS configuration under the Compute node as shown in the image.

Step 11. Similar to BOOTOS VD created with physical drives 1 & 2 , create four more virtual drives as

JOURNAL - From physical drive number 3

OSD1 - From physical drive number 7

OSD2 - From physical drive number 8

OSD3 - From physical drive number 9

OSD4 - From physical drive number 10

Step 7. In the end, the physical drives and Virtual drives must be similar.

**Note:** The image shown here and the configuration steps mentioned in this section are with reference to the firmware version 3.0(3e) and there might be slight variations if you work on other versions.

## Add the new OSD-Compute node to the Overcloud

The steps mentioned in this section are common irrespective of the VM hosted by the compute node.

Step 1. Add Compute server with a different index.

Create an **add\_node.json** file with only the details of the new compute server to be added. Ensure that the index number for the new osd-compute server has not been used before. Typically, increment the next highest compute value.

Example: Highest prior was osd-compute-0 so created osd-compute-3 in case of 2-vnf system.

**Note:** Be mindful of the json format.

```
[stack@director ~]$ cat add_node.json
{
  "nodes": [
    {
      "mac": [
        "<MAC_ADDRESS>"
      ],
      "capabilities": "node:osd-compute-3,boot_option:local",
      "cpu": "24",
      "memory": "256000",
      "disk": "3000",
      "arch": "x86_64",
      "pm_type": "pxe_ipmitool",
      "pm_user": "admin",
      "pm_password": "<PASSWORD>",
      "pm_addr": "192.100.0.5"
    }
  ]
}
```

Step 2. Import the json file.

```
[stack@director ~]$ openstack baremetal import --json add_node.json
Started Mistral Workflow. Execution ID: 78f3b22c-5c11-4d08-a00f-8553b09f497d
Successfully registered node UUID 7eddfa87-6ae6-4308-b1d2-78c98689a56e
Started Mistral Workflow. Execution ID: 33a68c16-c6fd-4f2a-9df9-926545f2127e
Successfully set all nodes to available.
```

### Step 3. Run node introspection with the use of the UUID noted from the previous step.

```
[stack@director ~]$ openstack baremetal node manage 7eddfa87-6ae6-4308-b1d2-78c98689a56e
[stack@director ~]$ ironic node-list |grep 7eddfa87
| 7eddfa87-6ae6-4308-b1d2-78c98689a56e | None | None | power off
| manageable | False |
```

```
[stack@director ~]$ openstack overcloud node introspect 7eddfa87-6ae6-4308-b1d2-78c98689a56e --
provide
Started Mistral Workflow. Execution ID: e320298a-6562-42e3-8ba6-5ce6d8524e5c
Waiting for introspection to finish...
Successfully introspected all nodes.
Introspection completed.
Started Mistral Workflow. Execution ID: c4a90d7b-ebf2-4fcb-96bf-e3168aa69dc9
Successfully set all nodes to available.
```

```
[stack@director ~]$ ironic node-list |grep available
| 7eddfa87-6ae6-4308-b1d2-78c98689a56e | None | None | power off
| available | False |
```

### Step 4. Add IP addresses to custom-templates/layout.yml under OsdComputeIPs. In this case, as you replace osd-compute-0, you add that address to the end of the list for each type.

OsdComputeIPs:

```
internal_api:
- 11.120.0.43
- 11.120.0.44
- 11.120.0.45
- 11.120.0.43 <<< take osd-compute-0 .43 and add here

tenant:
- 11.117.0.43
- 11.117.0.44
- 11.117.0.45
- 11.117.0.43 << and here

storage:
- 11.118.0.43
- 11.118.0.44
- 11.118.0.45
```

```
- 11.118.0.43 << and here
```

```
storage_mgmt:
```

```
- 11.119.0.43
```

```
- 11.119.0.44
```

```
- 11.119.0.45
```

```
- 11.119.0.43 << and here
```

Step 5. Run **deploy.sh** script that was previously used to deploy the stack, in order to add the new compute node to the overcloud stack.

```
[stack@director ~]$ ./deploy.sh
++ openstack overcloud deploy --templates -r /home/stack/custom-templates/custom-roles.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml -e
/home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e
/home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml --stack
ADN-ultram --debug --log-file overcloudDeploy_11_06_17__16_39_26.log --ntp-server 172.24.167.109
--neutron-flat-networks phys_pcie1_0,phys_pcie1_1,phys_pcie4_0,phys_pcie4_1 --neutron-network-
vlan-ranges datacentre:1001:1050 --neutron-disable-tunneling --verbose --timeout 180
...
Starting new HTTP connection (1): 192.200.0.1
"POST /v2/action_executions HTTP/1.1" 201 1695
HTTP POST http://192.200.0.1:8989/v2/action_executions 201
Overcloud Endpoint: http://10.1.2.5:5000/v2.0
Overcloud Deployed
clean_up DeployOvercloud:
END return value: 0

real    38m38.971s
user    0m3.605s
sys     0m0.466s
```

Step 6. Wait for the openstack stack status to be COMPLETE.

```
[stack@director ~]$ openstack stack list
+-----+-----+-----+-----+
| ID                                     | Stack Name | Stack Status | Creation Time |
Updated Time |
+-----+-----+-----+-----+
| 5df68458-095d-43bd-a8c4-033e68ba79a0 | pod1      | UPDATE_COMPLETE | 2017-11-02T21:30:06Z | 2017-
11-06T21:40:58Z |
+-----+-----+-----+-----+
```

Step 7. Check that new osd-compute node is in the Active state.

```
[stack@director ~]$ source stackrc
[stack@director ~]$ nova list |grep osd-compute-3
| 0f2d88cd-d2b9-4f28-b2ca-13e305ad49ea | pod1-osd-compute-3 | ACTIVE | - | Running
| ctlplane=192.200.0.117 |
```



```
[stack@director ~]$ source corerc
[stack@director ~]$ openstack hypervisor list |grep osd-compute-3
| 63 | pod1-osd-compute-3.localdomain |
```

Step 8. Login to the new osd-compute server and check ceph processes. Initially, status is in HEALTH\_WARN as ceph recovers.

```
[heat-admin@pod1-osd-compute-3 ~]$ sudo ceph -s

cluster eb2bb192-b1c9-11e6-9205-525400330666

health HEALTH_WARN

    223 pgs backfill_wait

    4 pgs backfilling

    41 pgs degraded

    227 pgs stuck unclean

    41 pgs undersized

recovery 45229/1300136 objects degraded (3.479%)

recovery 525016/1300136 objects misplaced (40.382%)

monmap e1: 3 mons at {Pod1-controller-0=11.118.0.40:6789/0,Pod1-controller-1=11.118.0.41:6789/0,Pod1-controller-2=11.118.0.42:6789/0}

election epoch 58, quorum 0,1,2 Pod1-controller-0,Pod1-controller-1,Pod1-controller-2

osdmap e986: 12 osds: 12 up, 12 in; 225 remapped pgs

flags sortbitwise,require_jewel_osds

pgmap v781746: 704 pgs, 6 pools, 533 GB data, 344 kobjects

1553 GB used, 11840 GB / 13393 GB avail

45229/1300136 objects degraded (3.479%)

525016/1300136 objects misplaced (40.382%)

    477 active+clean

    186 active+remapped+wait_backfill

    37 active+undersized+degraded+remapped+wait_backfill

    4 active+undersized+degraded+remapped+backfilling
```

Step 9. However, after a short period (20 minutes), CEPH returns to a HEALTH\_OK state.

```
[heat-admin@pod1-osd-compute-3 ~]$ sudo ceph -s

cluster eb2bb192-b1c9-11e6-9205-525400330666

health HEALTH_OK
```

```
monmap e1: 3 mons at {Pod1-controller-0=11.118.0.40:6789/0,Pod1-controller-1=11.118.0.41:6789/0,Pod1-controller-2=11.118.0.42:6789/0}
```

```
election epoch 58, quorum 0,1,2 Pod1-controller-0,Pod1-controller-1,Pod1-controller-2
```

```
osdmap e1398: 12 osds: 12 up, 12 in
```

```
flags sortbitwise,require_jewel_osds
```

```
pgmap v784311: 704 pgs, 6 pools, 533 GB data, 344 kobjects
```

```
1599 GB used, 11793 GB / 13393 GB avail
```

```
704 active+clean
```

```
client io 8168 kB/s wr, 0 op/s rd, 32 op/s wr
```

```
[heat-admin@pod1-osd-compute-3 ~]$ sudo ceph osd tree
```

ID	WEIGHT	TYPE	NAME	UP/DOWN	REWEIGHT	PRIMARY-AFFINITY
-1	13.07996	root	default			
-2	0	host	pod1-osd-compute-0			
-3	4.35999	host	pod1-osd-compute-2			
1	1.09000		osd.1	up	1.00000	1.00000
4	1.09000		osd.4	up	1.00000	1.00000
7	1.09000		osd.7	up	1.00000	1.00000
10	1.09000		osd.10	up	1.00000	1.00000
-4	4.35999	host	pod1-osd-compute-1			
2	1.09000		osd.2	up	1.00000	1.00000
5	1.09000		osd.5	up	1.00000	1.00000
8	1.09000		osd.8	up	1.00000	1.00000
11	1.09000		osd.11	up	1.00000	1.00000
-5	4.35999	host	pod1-osd-compute-3			
0	1.09000		osd.0	up	1.00000	1.00000
3	1.09000		osd.3	up	1.00000	1.00000
6	1.09000		osd.6	up	1.00000	1.00000
9	1.09000		osd.9	up	1.00000	1.00000

## Restore the VMs

## Addition to Nova Aggregate List

Add the osd-compute node to the aggregate-hosts and verify if the host is added.

```
nova aggregate-add-host <Aggregate> <Host>
```

```
[stack@director ~]$ nova aggregate-add-host esc1 pod1-osd-compute-3.localdomain
```

```
nova aggregate-show <Aggregate>
```

```
[stack@director ~]$ nova aggregate-show esc1
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | Name | Availability Zone | Hosts | Metadata |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | esc1 | AZ-esc1 | 'pod1-osd-compute-3.localdomain' | 'availability_zone=AZ-esc1',
'esc1=true' |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
```

## Recovery of ESC VM

Step 1. Check the status of the ESC VM from the nova list and delete it.

```
stack@director scripts]$ nova list |grep esc
```

```
| c566efbf-1274-4588-a2d8-0682e17b0d41 | esc |
ACTIVE | - | Running | VNF2-UAS-uas-orchestration=172.168.11.14; VNF2-UAS-uas-
management=172.168.10.4
```

```
[stack@director scripts]$ nova delete esc
Request to delete server esc has been accepted.
```

If can not delete esc then use command: `nova force-delete esc`

Step 2. In OSPD, navigate to ECS-Image directory and ensure that the **bootvm.py** and **qcow2** for ESC release are present, if not move it to a directory.

```
[stack@atospd ESC-Image-157]$ ll
```

```
total 30720136
-rw-r--r--. 1 root root 127724 Jan 23 12:51 bootvm-2_3_2_157a.py
-rw-r--r--. 1 root root 55 Jan 23 13:00 bootvm-2_3_2_157a.py.md5sum
-rw-rw-r--. 1 stack stack 31457280000 Jan 24 11:35 esc-2.3.2.157.qcow2
```

Step 3. Create the image.

```
[stack@director ESC-image-157]$ glance image-create --name ESC-2_3_2_157 --disk-format "qcow2"
--container "bare" --file /home/stack/ECS-Image-157/ESC-2_3_2_157.qcow2
```

Step 4. Verify ESC image exists.

```
stack@director ~]$ glance image-list
```

```
+-----+-----+-----+-----+-----+-----+
| ID | Name |
+-----+-----+-----+-----+-----+-----+
```

```

+-----+
| 8f50acbe-b391-4433-aa21-98ac36011533 | ESC-2_3_2_157 |
| 2f67f8e0-5473-467c-832b-e07760e8d1fa | tmobile-pcrf-13.1.1.iso |
| c5485c30-45db-43df-831d-61046c5cfd01 | tmobile-pcrf-13.1.1.qcow2 |
| 2f84b9ec-61fa-46a3-a4e6-45f14c93d9a9 | tmobile-pcrf-13.1.1_cco_20170825.iso |
| 25113ecf-8e63-4b81-a73f-63606781ef94 | wscaaa01-sept072017 |
| 595673e8-c99c-40c2-82b1-7338325024a9 | wscaaa02-sept072017 |
| 8bce3a60-b3b0-4386-9e9d-d99590dc9033 | wscaaa03-sept072017 |
| e5c835ad-654b-45b0-8d36-557e6c5fd6e9 | wscaaa04-sept072017 |
| 879dfcde-d25c-4314-8da0-32e4e73ffc9f | WSP1_cluman_12_07_2017 |
| 7747dd59-c479-4c8a-9136-c90ec894569a | WSP2_cluman_12_07_2017 |
+-----+

```

```
[stack@ ~]$ openstack flavor list
```

```

+-----+
-----+
| ID | Name | RAM | Disk | Ephemeral | VCPUs | Is
Public |
+-----+
-----+
| 1e4596d5-46f0-46ba-9534-cfdea788f734 | pcrf-smb | 100352 | 100 | 0 | 8 | True
| 251225f3-64c9-4b19-a2fc-032a72bfe969 | pcrf-oam | 65536 | 100 | 0 | 10 | True
| 4215d4c3-5b2a-419e-b69e-7941e2abe3bc | pcrf-pd | 16384 | 100 | 0 | 12 | True
| 4c64a80a-4d19-4d52-b818-e904a13156ca | pcrf-qns | 14336 | 100 | 0 | 10 | True
| 8b4cbba7-40fd-49b9-ab21-93818c80a2e6 | esc-flavor | 4096 | 0 | 0 | 4 | True
| 9c290b80-f80a-4850-b72f-d2d70d3d38ea | pcrf-sm | 100352 | 100 | 0 | 10 | True
| e993fc2c-f3b2-4f4f-9cd9-3afc058b7ed1 | pcrf-arb | 16384 | 100 | 0 | 4 | True
| f2b3b925-1bf8-4022-9f17-433d6d2c47b5 | pcrf-cm | 14336 | 100 | 0 | 6 | True
+-----+
-----+

```

Step 5. Create this file under the image directory and launch the ESC instance.

```
[root@director ESC-IMAGE]# cat esc_params.conf
openstack.endpoint = publicURL
```

```
[root@director ESC-IMAGE] ./bootvm-2_3_2_157a.py esc --flavor esc-flavor --image ESC-2_3_2_157 --
net tb1-mgmt --gateway_ip 172.16.181.1 --net tb1-orch --enable-http-rest --avail_zone AZ-escl --
user_pass "admin:Cisco123" --user_confid_pass "admin:Cisco123" --bs_os_auth_url
http://10.250.246.137:5000/v2.0 --kad_vif eth0 --kad_vip 172.16.181.5 --ipaddr 172.16.181.4 dhcp
--ha_node_list 172.16.181.3 172.16.181.4 --esc_params_file esc_params.conf
```

**Note:** After the problematic ESC VM is redeployed with exactly the same **bootvm.py** command as the initial installation, ESC HA performs synchronization automatically without any manual procedure. Ensure that ESC Master is Up and runs.

Step 6. Login to new ESC and verify the Backup state.

```
[admin@esc ~]$ esccadm status
0 ESC status=0 ESC Backup Healthy
```

```
[admin@VNF2-esc-esc-1 ~]$ health.sh
```

```
===== ESC HA (BACKUP) =====
```

```
ESC HEALTH PASSED
```