

PCRF Replacement of Compute Server UCS C240 M4

Contents

[Introduction](#)

[Background Information](#)

[Healthcheck](#)

[Backup](#)

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

[Disable the PCRF Services Residing on the VM to be Shutdown](#)

[Remove the Compute Node from Nova Aggregate List](#)

[Compute Node Deletion](#)

[Delete from Overcloud](#)

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

[Delete Neutron Agents](#)

[Delete from the Ironic Database](#)

[Install the New Compute Node](#)

[Add the New Compute Node to the Overcloud](#)

[Restore the VMs](#)

[Addition to Nova Aggregate List](#)

[VM Recovery from Elastic Services Controller \(ESC\)](#)

[Check the Cisco Policy and Charging Rules Function \(PCRF\) Services that Resides on VM](#)

[Delete and Re-Deploy One or More VMs in Case ESC Recovery Fails](#)

[Obtain the Latest ESC Template for the Site](#)

[Procedure to the Modify the File](#)

[Step 1. Modify the Export Template File.](#)

[Step 2. Run the Modified Export Template File.](#)

[Step 3. Modify the Export Template File to Add the VMs.](#)

[Step 4. Run the Modified Export Template File.](#)

[Step 5. Check the PCRF Services that Reside on the VM.](#)

[Step 6. Run the Diagnostics to Check System Status.](#)

[Related Information](#)

Introduction

This document describes the steps required to replace a faulty 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 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 Compute node, it is important to check the current health 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.

Step 1. From OpenStack Deployment (OSPD).

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

Step 2. Verify health of system from ultram-health report which is generated every fifteen minutes.

```
[stack@director ~]# cd /var/log/cisco/ultram-health
```

Step 3. Check file **ultram_health_os.report**. The only services should show as **XXX** status are **neutron-sriov-nic-agent.service**.

Step 4. To check if rabbitmq runs for all controllers run 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 5. Verify stonith is enabled

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

Step 6. For all Controllers verify PCS status.

- All controller nodes are **Started** under haproxy-clone.
- All controller nodes are **Active** under galera.
- All controller nodes are **Started** under Rabbitmq.
- 1 controller node is **Active** and 2 **Standby** under redis.

Step 7. 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 8. Verify all openstack services are Active, from OSPD run this command.

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

Step 9. 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 10. Verify OpenStack component logs. Look for any error:

Neutron:

```
[stack@director ~]# sudo tail -n 20 /var/log/neutron/{dhcp-agent,l3-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 11. 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 12. 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:

```
[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. Also, it is recommended to backup the CPS configuration.

In order 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 Compute Node

Identify the VMs that are hosted on the compute server:

```
[stack@director ~]$ nova list --field name,host,networks | grep compute-10
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-
10e75d0e134d | pod1-compute-10.localdomain | Replication=10.160.137.161;
Internal=192.168.1.131; Management=10.225.247.229; tb1-orch=172.16.180.129
```

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 are used in subsequent sections.

Disable the PCRF Services Residing on the VM to be Shutdown

Step 1. Login to management IP of the VM:

```
[stack@XX-ospd ~]$ ssh root@<Management IP>
[root@XXXSM03 ~]# monit stop all
```

Step 2. If the VM is an SM, OAM or arbiter, in addition, stop the sessionmgr services:

```
[root@XXXSM03 ~]# cd /etc/init.d
[root@XXXSM03 init.d]# ls -l sessionmgr*
-rwxr-xr-x 1 root root 4544 Nov 29 23:47 sessionmgr-27717
-rwxr-xr-x 1 root root 4399 Nov 28 22:45 sessionmgr-27721
-rwxr-xr-x 1 root root 4544 Nov 29 23:47 sessionmgr-27727
```

Step 3. For every file titled sessionmgr-xxxxx, run service sessionmgr-xxxxx stop:

```
[root@XXXSM03 init.d]# service sessionmgr-27717 stop
```

Remove the Compute Node from Nova Aggregate List

Step 1. 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>-SERVICE<X>:

```
[stack@director ~]$ nova aggregate-list
+-----+-----+-----+
| Id | Name | Availability Zone |
```

```
+-----+-----+-----+
| 29 | POD1-AUTOIT | mgmt |
| 57 | VNF1-SERVICE1 | - |
| 60 | VNF1-EM-MGMT1 | - |
| 63 | VNF1-CF-MGMT1 | - |
| 66 | VNF2-CF-MGMT2 | - |
| 69 | VNF2-EM-MGMT2 | - |
| 72 | VNF2-SERVICE2 | - |
| 75 | VNF3-CF-MGMT3 | - |
| 78 | VNF3-EM-MGMT3 | - |
| 81 | VNF3-SERVICE3 | - |
+-----+-----+-----+
```

In this case, the compute server to be replaced belongs to VNF2. Hence, the corresponding aggregate-list is VNF2-SERVICE2.

Step 2. Remove the compute node from the aggregate identified (remove by hostname noted from Section **Identify the VMs hosted in the Compute Node**

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

```
[stack@director ~]$ nova aggregate-remove-host VNF2-SERVICE2 pod1-compute-10.localdomain
```

Step 3. Verify if the compute node is removed from the aggregates. Now, the Host must not be listed under the aggregate:

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

```
[stack@director ~]$ nova aggregate-show VNF2-SERVICE2
```

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 here. Ensure that the templates mentioned are 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 -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 -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 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 compute-8
| 404 | nova-compute | pod1-compute-8.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 compute-8
| c3ee92ba-aa23-480c-ac81-d3d8d01dcc03 | Open vSwitch agent | pod1-compute-8.localdomain |
None | False | UP | neutron-openswitch-agent |
| ec19cb01-abbb-4773-8397-8739d9b0a349 | NIC Switch agent | pod1-compute-8.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 Ironic Database

Delete a Node from the Ironic Database and Verify it.

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

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

```
[stack@director ~]$ nova show pod1-compute-10 | 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. Log in to server with the use of the CIMC IP.

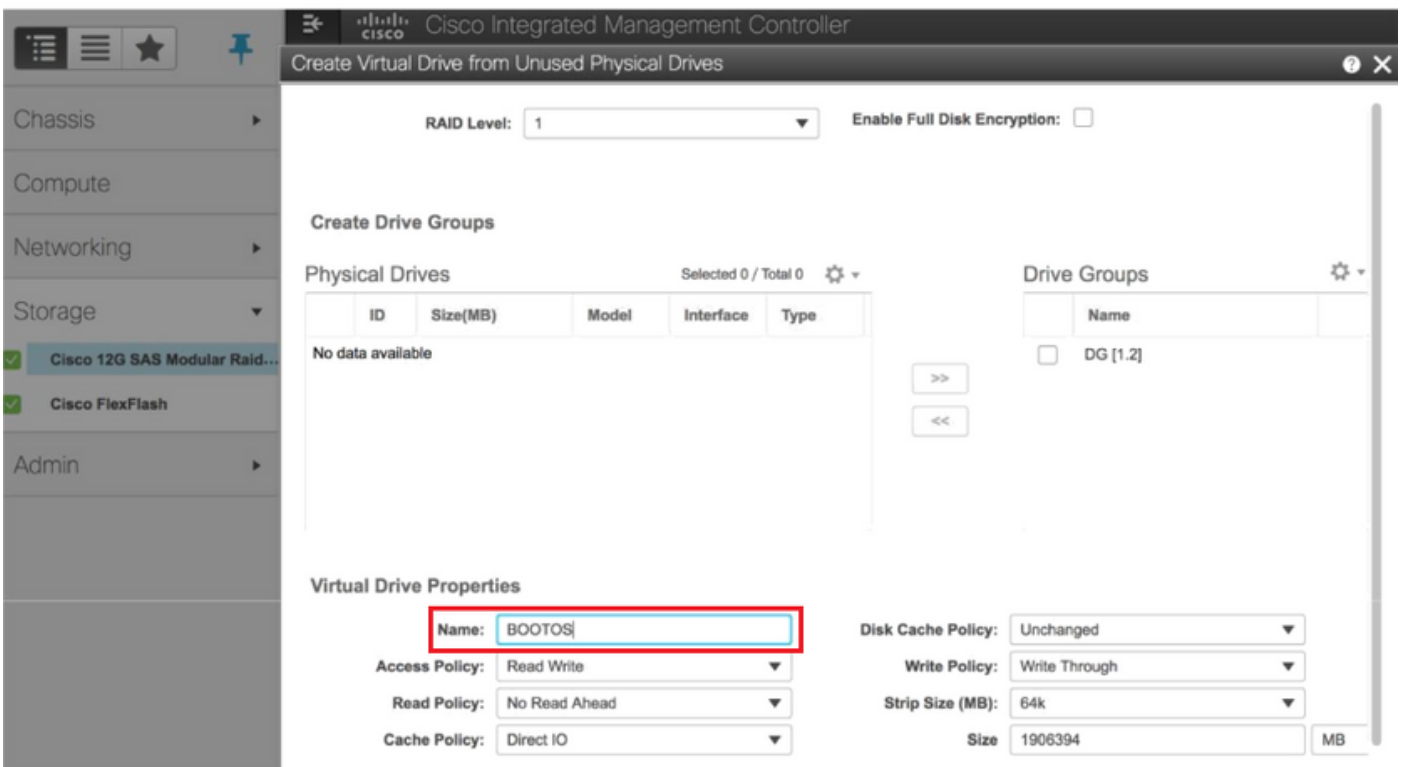
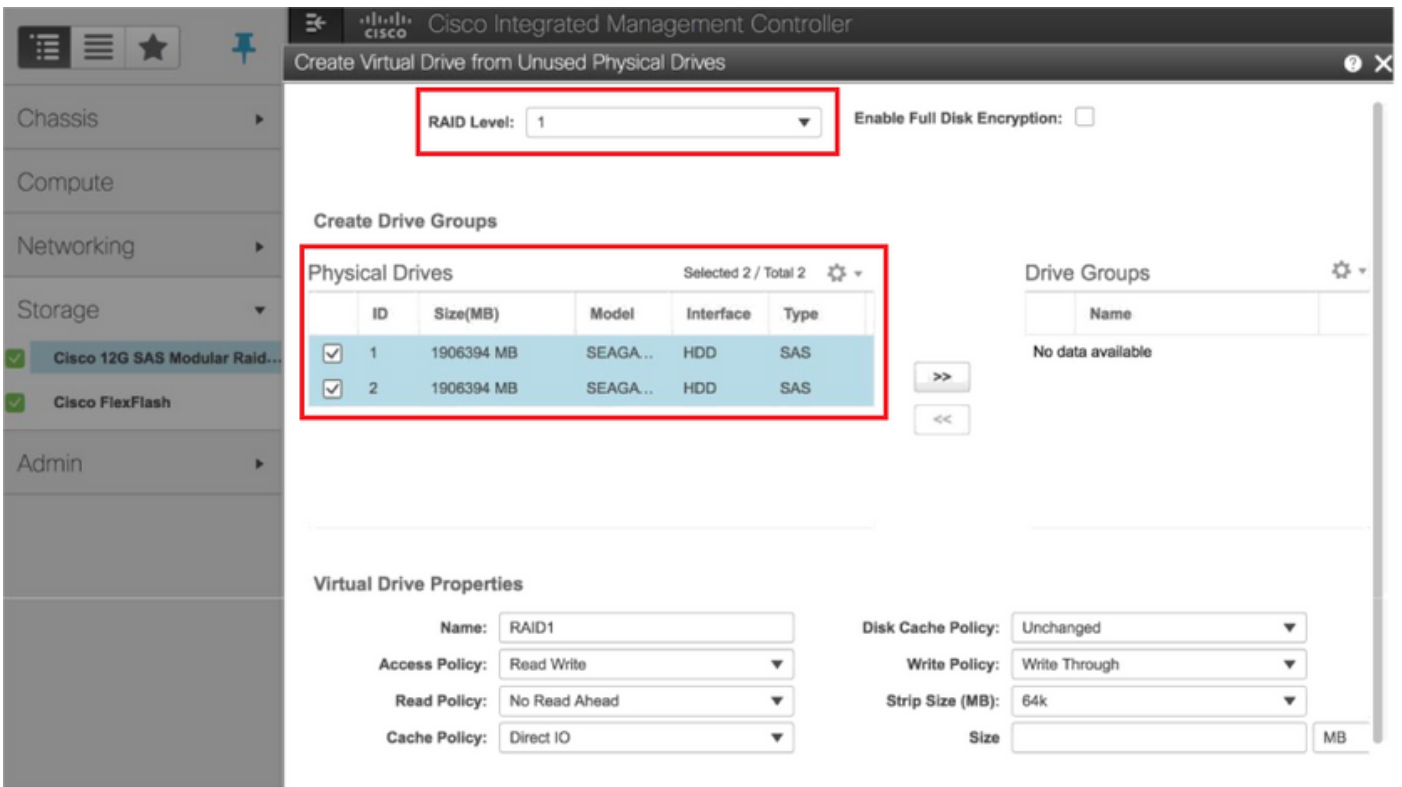
Step 3. Perform BIOS upgrade if the firmware is not as per the recommended version used previously. Steps for BIOS upgrade are given here: [Cisco UCS C-Series Rack-Mount Server BIOS Upgrade Guide](#)

Step 4. In order to verify the status of Physical drives, navigate to **Storage > Cisco 12G SAS Modular Raid Controller (SLOT-HBA) > Physical Drive Info**. It must be **Unconfigured Good**

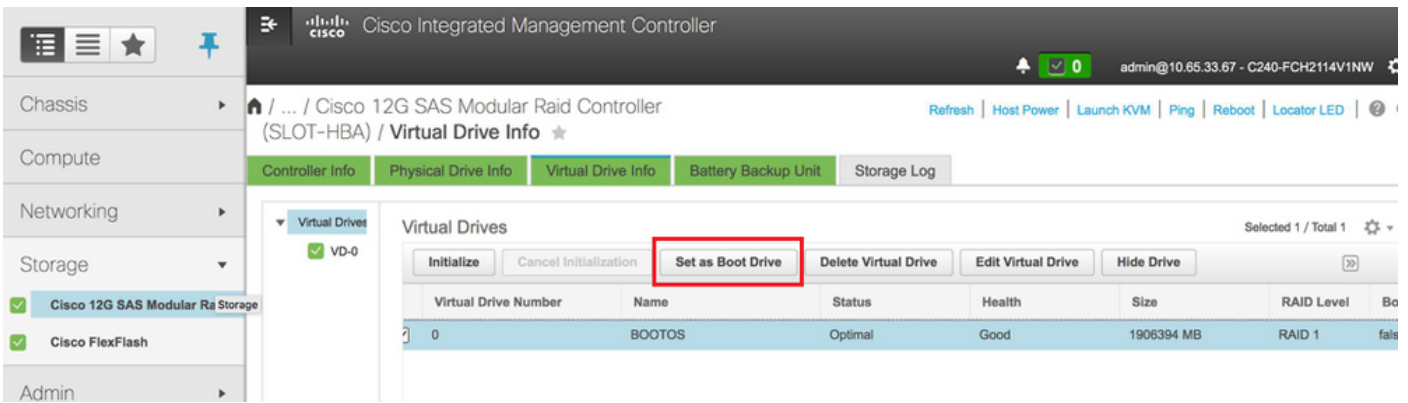
The storage shown here can be SSD drive.

Controller	Physical Drive Number	Status	Health	Boot Drive	Drive Firmware
<input type="checkbox"/> SLOT-HBA	1	Unconfigured Good	Good	false	N003
<input type="checkbox"/> SLOT-HBA	2	Unconfigured Good	Good	false	N003

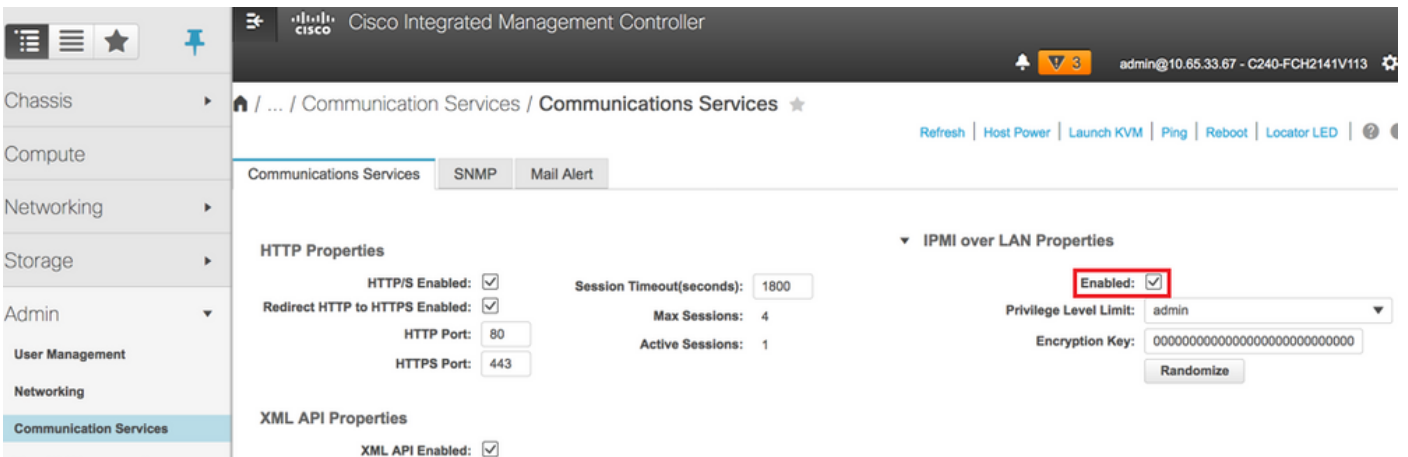
Step 5. In order to create a virtual drive from the physical drives with RAID Level 1, navigate to **Storage > Cisco 12G SAS Modular Raid Controller (SLOT-HBA) > Controller Info > Create Virtual Drive from Unused Physical Drives**



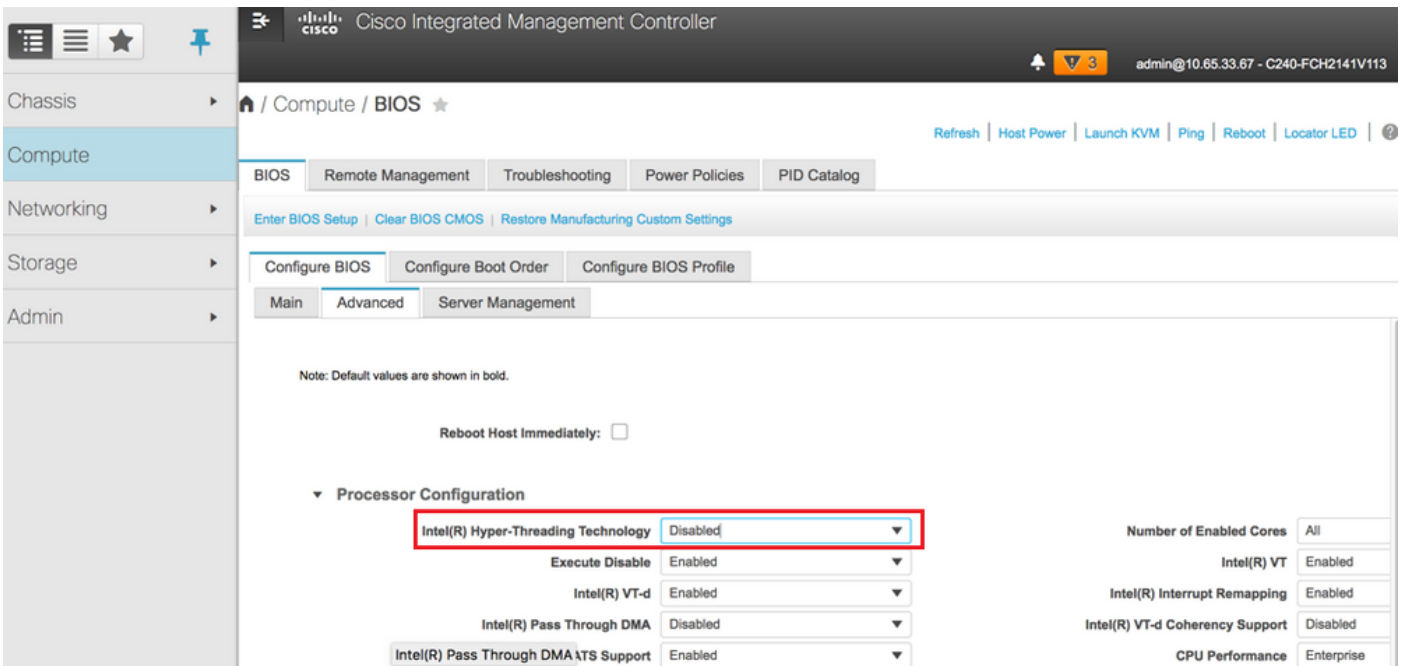
Step 6. Select the VD and configure **Set as Boot Drive**, as shown in the image.



Step 7. In order to enable IPMI over LAN, navigate to **Admin > Communication Services > Communication Services**, as shown in the image.



Step 8. In order to disable hyperthreading, as shown in the image, navigate to **Compute > BIOS > Configure BIOS > Advanced > Processor Configuration**.



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 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 compute server is not used before. Typically, increment the next highest compute value.

Example: Highest prior was compute-17, therefore, created compute-18 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:compute-18,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 ComputeIPs. You add that address to the end of the list for each type, compute-0 shown here as an example.

ComputeIPs:

internal_api:

- 11.120.0.43

- 11.120.0.44

- 11.120.0.45

- 11.120.0.43 <<< take 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

Step 5. Execute **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 | ADN-ultram | UPDATE_COMPLETE | 2017-11-02T21:30:06Z |
2017-11-06T21:40:58Z |
+-----+-----+-----+-----+
```

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

```
[stack@director ~]$ source stackrc
[stack@director ~]$ nova list |grep compute-18
| 0f2d88cd-d2b9-4f28-b2ca-13e305ad49ea | pod1-compute-18 | ACTIVE | - | Running
| ctlplane=192.200.0.117 |

[stack@director ~]$ source corerc
[stack@director ~]$ openstack hypervisor list |grep compute-18
| 63 | pod1-compute-18.localdomain |
```

Restore the VMs

Addition to Nova Aggregate List

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

```
nova aggregate-add-host <Aggregate> <Host>
[stack@director ~]$ nova aggregate-add-host VNF2-SERVICE2 pod1-compute-18.localdomain

nova aggregate-show <Aggregate>
[stack@director ~]$ nova aggregate-show VNF2-SERVICE2
```

VM Recovery from Elastic Services Controller (ESC)

Step 1. The VM is in error state in the nova list.

```
[stack@director ~]$ nova list |grep VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
| ERROR | - | NOSTATE |
```

Step 2. Recover the VM from the ESC.

```
[admin@VNF2-esc-esc-0 ~]$ sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO
VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
[sudo] password for admin:
```

Recovery VM Action

```
/opt/cisco/esc/confd/bin/netconf-console --port=830 --host=127.0.0.1 --user=admin --
privKeyFile=/root/.ssh/confd_id_dsa --privKeyType=dsa --rpc=/tmp/esc_nc_cli.ZpRCGiieuW
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <ok/>
</rpc-reply>
```

Step 3. Monitor the yangesc.log.

```
admin@VNF2-esc-esc-0 ~]$ tail -f /var/log/esc/yangesc.log
...
14:59:50,112 07-Nov-2017 WARN Type: VM_RECOVERY_COMPLETE
14:59:50,112 07-Nov-2017 WARN Status: SUCCESS
14:59:50,112 07-Nov-2017 WARN Status Code: 200
14:59:50,112 07-Nov-2017 WARN Status Msg: Recovery: Successfully recovered VM [VNF2-
DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d].
```

Check the Cisco Policy and Charging Rules Function (PCRF) Services that Resides on VM

Note: If VM is in shutoff state then Power it On using `esc_nc_cli` from ESC.

Check the `diagnostics.sh` from cluster manager VM & if any error found for the VMs which are recovered then

Step 1. Login to the respective VM.

```
[stack@XX-ospd ~]$ ssh root@<Management IP>
[root@XXXSM03 ~]# monit start all
```

Step 2. If the VM is a **SM**, **OAM** or **arbiter**, in addition to it, start the sessionmgr services which stopped earlier:

For every file titled sessionmgr-xxxxx, run service sessionmgr-xxxxx start:

```
[root@XXXSM03 init.d]# service sessionmgr-27717 start
```

If still the diagnostic is not clear then perform `build_all.sh` from Cluster Manager VM and then perform VM-init on respective VM.

```
/var/qps/install/current/scripts/build_all.sh
```

```
ssh VM e.g. ssh pcrfclient01  
/etc/init.d/vm-init
```

Delete and Re-Deploy One or More VMs in Case ESC Recovery Fails

If the ESC recovery command (above) does not work (VM_RECOVERY_FAILED) then delete and read the individual VMs.

Obtain the Latest ESC Template for the Site

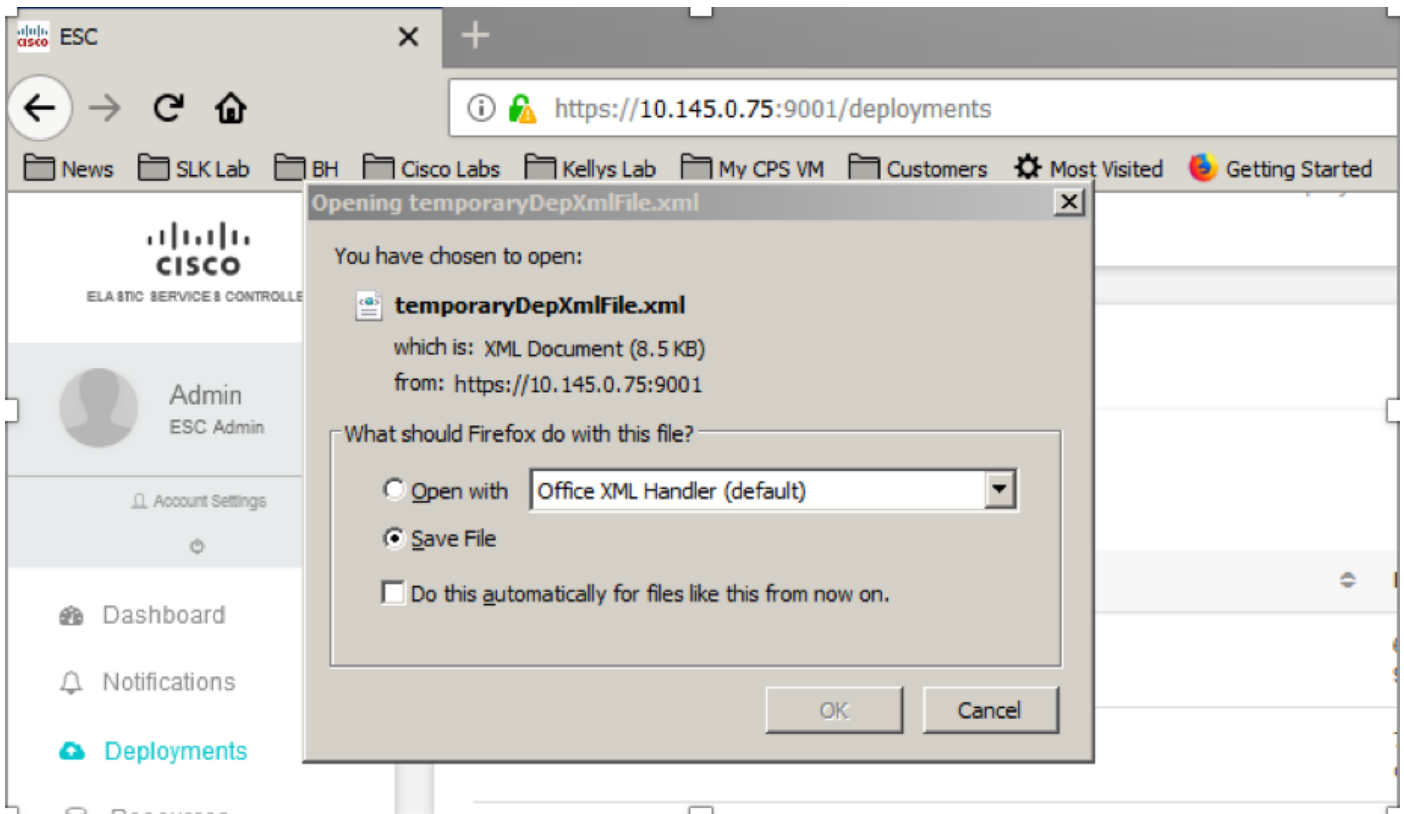
From ESC Portal:

Step 1. Place your cursor over the blue **Action** button, a pop-up window opens, now click on **Export Template**, as shown in the image.

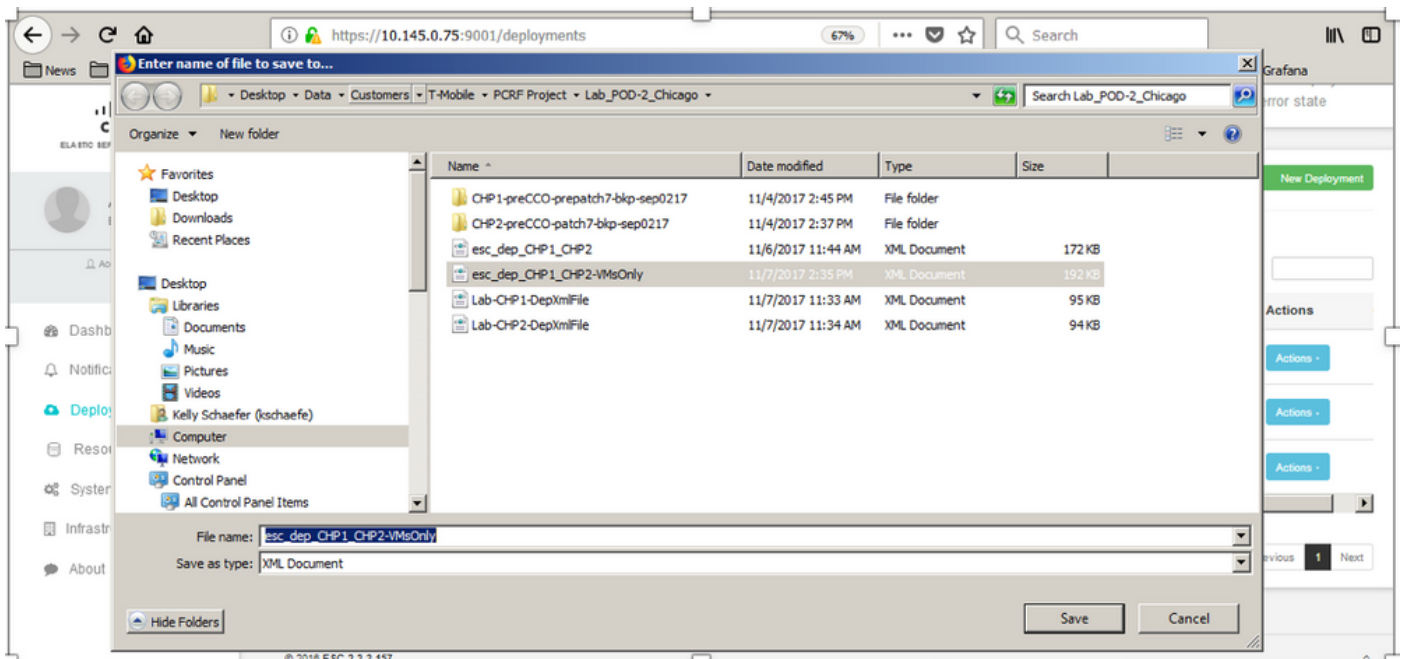
The screenshot displays the Cisco Elastic Service Controller (ESC) interface. The top navigation bar includes the Cisco logo and the title 'Deployments'. A sidebar on the left contains navigation links for Dashboard, Notifications, Deployments, Resources, System, Infrastructure, and About. The main content area shows a summary of VM status: 0 VMs waiting, 0 VMs deploying, 78 VMs deployed, 78 VMs active, 0 VMs failed to deploy (Error), and 0 VMs deployed in error state. Below this is a table of deployments with columns for Deployment Name, Tenant Name, Deployment ID, # of VNFs, Status, and Actions. Two deployments are listed: RIP1-tno and RIP2-tno, both with status 'Active'. A search bar and a 'Show 10 entries' dropdown are visible. A pop-up menu is open over the 'Actions' column, showing options: Update, View VNFs, Export Template, and Undeploy.

Deployment Name	Tenant Name	Deployment ID	# of VNFs	Status	Actions
RIP1-tno	Porf	78c67040-0b6a-420e-8e01-44279a6e6906	23	Active	Actions
RIP2-tno	Porf	d29e095a-800b-4067-8084-6700570c3a3f	23	Active	Actions

Step 2. An option to download the template to the local machine is presented, check on **Save File**, as shown in the image.



Step 3. As shown in the image, select a location and save the file for later use.



Step 4. Login to the Active ESC for the site to be deleted and copy the above-saved file in the ESC in this directory.

```
/opt/cisco/esc/cisco-cps/config/gr/tmo/gen
```

Step 5. Change Directory to **/opt/cisco/esc/cisco-cps/config/gr/tmo/gen**:

```
cd /opt/cisco/esc/cisco-cps/config/gr/tmo/gen
```

Procedure to the Modify the File

Step 1. Modify the Export Template File.

In this step, you modify the export template file to delete the VM group or groups associated with the VMs that need to be recovered.

The export template file is for a specific cluster.

Within that cluster are multiple vm_groups. There are one or more vm_groups for each VM type (PD, PS, SM, OM).

Note: Some vm_groups have more than one VM. All VMs within that group will be deleted and re-added.

Within that deployment, you need to tag one or more of the vm_groups for deletion.

Example:

```
<vm_group>
  <name>cm</name>
```

Now Change the <vm_group>to <vm_group nc:operation="delete"> and save the changes.

Step 2. Run the Modified Export Template File.

From the ESC run:

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli edit-config /opt/cisco/esc/cisco-cps/config/gr/tmo/gen/<modified_file_name>
```

From the ESC Portal, you should be able to see one or more VMs that move to the **undeploy** state and then disappeared completely.

Progress can be tracked in the ESC's **/var/log/esc/yangesc.log**

Example:

```
09:09:12,608 29-Jan-2018 INFO ===== UPDATE SERVICE REQUEST RECEIVED(UNDER TENANT) =====
09:09:12,608 29-Jan-2018 INFO Tenant name: Pcrf
09:09:12,609 29-Jan-2018 INFO Deployment name: WSP1-tmo
09:09:29,794 29-Jan-2018 INFO
09:09:29,794 29-Jan-2018 INFO ===== CONFID TRANSACTION ACCEPTED =====
09:10:19,459 29-Jan-2018 INFO
09:10:19,459 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:10:19,459 29-Jan-2018 INFO Type: VM_UNDEPLOYED
09:10:19,459 29-Jan-2018 INFO Status: SUCCESS
09:10:19,459 29-Jan-2018 INFO Status Code: 200
|
|
|
09:10:22,292 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:10:22,292 29-Jan-2018 INFO Type: SERVICE_UPDATED
09:10:22,292 29-Jan-2018 INFO Status: SUCCESS
```


Step 3. Modify the Export Template File to Add the VMs.

In this step, you modify the export template file to re-add the VM group or groups associated with the VMs that are being recovered.

The export template file is broken down into the two deployments (cluster1 / cluster2).

Within each cluster is a vm_group. There are one or more vm_groups for each VM type (PD, PS, SM, OM).

Note: Some vm_groups have more than one VM. All VMs within that group will be re-added.

Example:

```
<vm_group nc:operation="delete">
  <name>cm</name>
```

Change the **<vm_group nc:operation="delete">** to just **<vm_group>**.

Note: If the VMs need to be rebuilt because the Host was replaced, the hostname of the Host may have changed. If the hostname of the HOST has changed then the hostname within the **placement section** of the **vm_group** will need to be updated.

```
<placement>
```

```
  <type>zone_host</type>
```

```
  <enforcement>strict</enforcement>
```

```
  <host>wsstackovs-compute-4.localdomain</host>
```

```
</placement>
```

Update the name of the host shown in the preceding section to the new hostname as provided by the Ultra-M team prior to the execution of this MOP. After the installation of the new host, save the changes.

Step 4. Run the Modified Export Template File.

From the ESC run:

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli edit-config /opt/cisco/esc/cisco-  
cps/config/gr/tmo/gen/<modified_file_name>
```

From the ESC Portal, you should be able to see one or more VMs reappear, then into the Active state.

Progress can be tracked in the ESC's **/var/log/esc/yangesc.log**

Example:

```
09:14:00,906 29-Jan-2018 INFO ===== UPDATE SERVICE REQUESTRECEIVED (UNDER TENANT) =====
09:14:00,906 29-Jan-2018 INFO Tenant name: Pcrf
09:14:00,906 29-Jan-2018 INFO Deployment name: WSP1-tmo
09:14:01,542 29-Jan-2018 INFO
09:14:01,542 29-Jan-2018 INFO ===== CONFID TRANSACTION ACCEPTED =====
09:16:33,947 29-Jan-2018 INFO
09:16:33,947 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:16:33,947 29-Jan-2018 INFO Type: VM_DEPLOYED
09:16:33,947 29-Jan-2018 INFO Status: SUCCESS
09:16:33,947 29-Jan-2018 INFO Status Code: 200
|
|
|
09:19:00,148 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:19:00,148 29-Jan-2018 INFO Type: VM_ALIVE
09:19:00,148 29-Jan-2018 INFO Status: SUCCESS
09:19:00,148 29-Jan-2018 INFO Status Code: 200
|
|
|
09:19:00,275 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:19:00,275 29-Jan-2018 INFO Type: SERVICE_UPDATED
09:19:00,275 29-Jan-2018 INFO Status: SUCCESS
09:19:00,275 29-Jan-2018 INFO Status Code: 200
```

Step 5. Check the PCRF Services that Reside on the VM.

Check whether the PCRF services are down and start them.

```
[stack@XX-ospd ~]$ ssh root@<Management IP>
[root@XXXSM03 ~]# monsum
[root@XXXSM03 ~]# monit start all
```

If the **VM** is an **SM**, **OAM** or **arbiter**, in addition, start the sessionmgr services which stopped earlier:

For every file titled sessionmgr-xxxxx run service sessionmgr-xxxxx start:

```
[root@XXXSM03 init.d]# service sessionmgr-27717 start
```

If still the diagnostic is not clear, perform **build_all.sh** from Cluster Manager VM and then perform VM-init on the respective VM.

```
/var/qps/install/current/scripts/build_all.sh
```

```
ssh VM e.g. ssh pcrfclient01
/etc/init.d/vm-init
```

Step 6. Run the Diagnostics to Check System Status.

```
[root@XXXSM03 init.d]# diagnostics.sh
```

Related Information

- https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installati...
- https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installati...
- [Technical Support & Documentation - Cisco Systems](#)