Policy and Charging Rules Function Stop and Start Compute Node

Contents

[Introduction](#)
[Prerequisites](#)
[Identify the VMs Hosted in the Compute Node](#)
[Graceful Power Off](#)
[Compute Node Stop-Start](#)
[Restore the VMs](#)

Introduction

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

> **Note**: Ultra M 5.1.x release is considered in order to define the procedures in this document. This document is intended for the Cisco personnel who are 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 stop-start.

Prerequisites

Backup

Before you stop-start a Compute node, it is important to check the current state of your Red Hat OpenStack Platform environment. It is recommended that you check the current state in order to avoid complications.

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.

Use this configuration in order to back up CPS VMs from Cluster Manager Virtual Machine (VM).

```
[root@CM ~]# config_br.py -a export --all /mnt/backup/CPS_backup_28092016.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 will be used in subsequent sections.

Disable the PCRF Services Residing on the VM to be Shutdown

1. Login to the management IP of the VM.

```
[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
```

2. If the**VM**is an**SM**,**OAM**or**Arbiter**, in addition, stop the **sessionmgr** services.

```
[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
```

3. Forevery file titled **sessionmgr-xxxxx** run **service sessionmgr-xxxxx stop**.

```
[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
```

Graceful Power Off

Shutdown VM from ESC

1. Log in to the ESC node that corresponds to the VNF and check the status of the VM.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name> VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
```

```
                    <state>VM_ALIVE_STATE</state>
<snip>
```

2. Stop the VM with the use of its VM Name. (VM Name noted from section " Identify the VMs hosted in the Compute Node").

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name> VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                    <state>VM_ALIVE_STATE</state>
<snip>
```

3. Once it is stopped, the VM must enter the **SHUTOFF** state.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

Compute Node Stop-Start

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

Stop-Start Compute Node from the OSPD

1. Check the status and then stop-start the node.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

2. Wait for the Compute to be in Shutoff state & then start it again.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

3. Check that the new compute node is in the Active state.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>

[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

Restore the VMs

VM Recovery from ESC

1. Ideally, from OSPD if you check nova list, the VMs should be in Shut state. In this case, you need to start the VMs from ESC.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

2. Or, if the VM is in error state in the nova list, perform this configuration.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

3. Now, recover the VM from the ESC.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

4. Monitor the **yangesc.log**.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

Check the PCRF Services Residing on the VM

> **Note**: If the VM is in the SHUTOFF state, then Power it ON with the use of **esc_nc_cli** from ESC. Check the **diagnostics.sh** from cluster manager VM and if you come across any error found for the VMs which are recovered then.

1. Login to the respective VM.

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
```

```
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

2. If the**VM**is an**SM**,**OAM**or**Arbiter**, in addition, start the **sessionmgr** services which stopped earlier. Forevery file titled **sessionmgr-xxxxx,** run service **sessionmgr-xxxxx start.**

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```

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

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
                    <state>VM_ALIVE_STATE</state>
                    <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
                     <state>VM_SHUTOFF_STATE</state>
<snip>
```