Post Deployment Operations

Deactivating the USP Deployment

Caution It is recommended that you perform the checks identified in Pre-Deactivation/Post-Activation Health Check Summary, on page 5 before performing any deactivations. It is also recommended that you back up relevant data before proceeding. Refer to Backing Up Deployment Information for more information.

Execute the following command to deactivate the entire USP deployment:

deactivate-deployment service-deployment-id <*deployment-id*>
The output of this command is a transaction-id which can be used to monitor the deactivation progress using the following command

show logs <*transaction_id*> **log |display xml**

Example output for a successful USP deactivation:

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <log xmlns="http://www.cisco.com/usp/nfv/usp-autodeploy-oper">
    <tx-id>1495752667278</tx-id>
    <log>Thu May 25 22:51:08 UTC 2017 [Task: 1495752667278] Started service deployment
ServiceDeploymentRequest [type=DEACTIVATE, serviceDeploymentId=north-east, siteList=[]]
Thu May 25 22:51:08 UTC 2017 [Task: 1495752667278/auto-testvnfd2] Starting Vnf UnDeployment
Thu May 25 22:52:58 UTC 2017 [Task: 1495752667278/auto-testvnfd2] Successfully deactivated
 all Vnf Deployments.
Thu May 25 22:53:00 UTC 2017 [Task: 1495752667278/auto-testvnfd2] Vnf UnDeployment Successful
Thu May 25 22:53:00 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Deactivating VNFM
Thu May 25 22:53:31 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Successfully
deactivating VNFM
Thu May 25 22:53:31 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Deleted VnfmInstance
 configuration
Thu May 25 22:53:31 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Deleted Vnfm
configuration
Thu May 25 22:54:21 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-em-autovnf-mgmt2] Starting to delete
Host Aggregate.
Thu May 25 22:54:22 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-em-autovnf-mgmt2] Deleted Host Aggregate
 successfully.
Thu May 25 22:54:22 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-service2] Starting to delete Host
Aggregate.
Thu May 25 22:54:23 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-service2] Deleted Host Aggregate
successfully.
Thu May 25 22:54:23 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-cf-esc-mgmt2] Starting to delete Host
Aggregate.
Thu May 25 22:54:24 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-cf-esc-mgmt2] Deleted Host Aggregate
successfully.
Thu May 25 22:54:24 UTC 2017 [Task: 1495752667278/auto-testvnfd1] Starting Vnf UnDeployment
Thu May 25 22:56:24 UTC 2017 [Task: 1495752667278/auto-testvnfd1] Successfully deactivated
 all Vnf Deployments.
Thu May 25 22:56:26 UTC 2017 [Task: 1495752667278/auto-testvnfd1] Vnf UnDeployment Successful
Thu May 25 22:56:26 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Deactivating VNFM
Thu May 25 22:56:56 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Successfully
deactivating VNFM
Thu May 25 22:56:56 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Deleted VnfmInstance
 configuration
Thu May 25 22:56:56 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Deleted Vnfm
configuration
Thu May 25 22:57:54 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-service1] Starting to delete Host
Aggregate.
Thu May 25 22:57:55 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-service1] Deleted Host Aggregate
successfully.
Thu May 25 22:57:55 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-cf-esc-mgmt1] Starting to delete Host
Aggregate.
Thu May 25 22:57:56 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-cf-esc-mgmt1] Deleted Host Aggregate
successfully.
Thu May 25 22:57:56 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-em-autovnf-mgmt1] Starting to delete
Host Aggregate.
Thu May 25 22:57:57 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-em-autovnf-mgmt1] Deleted Host Aggregate
 successfully.
Thu May 25 22:57:58 UTC 2017 [Task: 1495752667278] Success
</log>
  </log>
</config>
```

Terminating the AutoDeploy VM

Terminating the AutoDeploy VM leverages the same *auto-deploy-booting.sh* used to instantiate the AutoDeploy VM.

Note Ensure that no changes have been made to this file since it was used to deploy AutoDeploy. Additionally, be sure to take a backup of the VM content if you are terminating the VM in order to upgrade with a new ISO.

To terminate the AutoDeploy VM:

1. Log on to the Ultra M Manager Node.
2. Terminate the AutoDeploy VM.

   ./auto-deploy-booting.sh --delete

Terminating the AutoIT-VNF VM

Terminating the AutoIT-VNF VM leverages the same *auto-it-vnf-staging.sh* used to instantiate the AutoIT-VNF VM.

Note Ensure that no changes have been made to this file since it was used to deploy AutoIT-VNF. Additionally, be sure to take a backup of the VM content if you are terminating the VM in order to upgrade with a new ISO.

To terminate the AutoDeploy VM:

1. Log on to the Ultra M Manager Node.
2. Terminate the AutoIT-VFN VM.

   ./auto-it-vnf-staging.sh --delete

Restarting the AutoIT-NFVI and AutoDeploy VMs

Within Ultra M Manager solution deployments based on the Hyper-Converged architecture, the Ultra M Manager node hosts the the AutoIT-NFVI and AutoDeploy VMs. These VMs are not designed to automatically restart after the Ultra M Manager Node is rebooted or power-cycled. In cases such as these, the AutoIT-NFVI and AutoDeploy VMs must be manually restarted.

To restart the AutoIT-NFVI and AutoDeploy VMs after the Ultra M Manager Node has rebooted:

1. Log on to the Ultra M Manager as the user **nfvi**.
2. Verify that the br-ex and br-ctlplane network bridges are up.

   ifconfig | more
3. Verify your default gateway configuration.

   route -n

4 Check the VM status.

virsh list -all
Example output:

```
Id    Name                           State
----------------------------------------------------
 1     undercloud                     running
 -     auto-deploy                    shut off
 -     nfvi                           shut off
```

5 Start AutoIT-NFVI.

virsh start nfvi
Example output:

```
Domain nfvi started
```

6 Validate that the required processes within the AutoIT-NFVI VM are up.

a Log in to the AutoIT-NFVI console as the user *ubuntu*.

virsh console nfvi

```
Connected to domain nfvi
Escape character is ^]

Ubuntu 14.04.3 LTS auto-nfvi ttyS0
```

auto-nfvi login: ubuntu
Password: <*password*>

b Verify UAS ConfD is running.

service uas-confd status
Example output:

```
uas-confd start/running, process 1223
```

c Verify the AutoIT-NFVI service is running.

service autoit-nfvi status
Example output:

```
autoit-nfvi start/running, process 1280
```

d Exit the AutoIT-NFVI console.

Ctrl+]

7 Start AutoDeploy.

virsh start auto-deploy
Example output:

```
Domain auto-deploy started
```

8 Verify the VM status.

virsh list -all

Example output:

```
Id    Name                           State
----------------------------------------------------
 1     undercloud                     running
 2     nfvi                           running
 3     auto-deploy                    running
```

9 Validate that the required processes within the AutoDeploy VM are up.

a Log in to the AutoDeploy console as the user *ubuntu*.

virsh console auto-deploy

```
Connected to domain auto-deploy
Escape character is ^]
```

```
Ubuntu 14.04.3 LTS auto-deploy ttyS0
auto-deploy login: ubuntu
Password: <password>
```

b Verify UAS ConfD is running.

service uas-confd status
Example output:

```
uas-confd start/running, process 1268
```

c Verify the AutoIT-NFVI service is running.

service autodeploy status
Example output:

```
autodeploy start/running, process 1338
```

d Exit the AutoIT-NFVI console.

Ctrl+]

Monitoring and Troubleshooting the Deployment

Pre-Deactivation/Post-Activation Health Check Summary

Table 1: Pre-deactivation/Post-activation Health Checks, on page 5 contains a summary of items to check/verify before performing a deactivation and/or after an activation.

Table 1: Pre-deactivation/Post-activation Health Checks

Item to Check	Notes
Checking OSP-D Server Health	Perform all identified checks.
Checking Controller Server Health	Perform all identified checks.
Checking OSD Compute Server Health	Perform all identified checks.
Viewing AutoDeploy Operational Data	Perform all identified checks.
Checking AutoVNF and UAS-Related Processes	Perform all identified checks.
Viewing AutoVNF Operational Data	In particular, check the outputs of the following commands: • **show uas** • **show autovnf-oper:vip-port** • **show autovnf-oper:vnf-em** • **show autovnf-oper:vnfm**
Viewing ESC Status	Perform all identified checks.

Item to Check	Notes
Viewing ESC Health	Perform all identified checks.
Viewing UEM Service Status	Perform all identified checks.
Viewing VNF Information through the Control Function	Perform all identified checks.

Checking NFVI Server Health

Checking OSP-D Server Health

Viewing Stack Status

Log on to the server on which OSP-D is running to view the stack status by executing the following command:

openstack stack list

Example output:

```
| ID                                   | Stack Name | Stack Status    | Creation Time
   | Updated Time |
+--------------------------------------+------------+-----------------+----------------------+--------------+
| db229d67-212d-4086-a266-e635b2902708 | tb3-ultram | CREATE_COMPLETE | 2017-06-20T02:31:31Z
 | None         |
+--------------------------------------+------------+-----------------+----------------------+--------------+
```

Note Prior to an update, the stack status may be “CREATE_COMPLETE” at the beginning of the update procedure. The stack status should read “UPDATE_COMPLETE” and list and update time at the successful completion of the update procedure.

Viewing the Bare Metal Node List

Log on to the server on which OSP-D is running to view the node list by executing the following command:

openstack baremetal node list

Example command output:

```
+--------------------------------------+------+--------------------------------------+-------------+--------------------+-------------+
| UUID                                 | Name | Instance UUID                        | Power
 State | Provisioning State | Maintenance |
+--------------------------------------+------+--------------------------------------+-------------+--------------------+-------------+
| 6725bb18-2895-4a8a-86ad-96b00cc9df4d | None | bc903f51-8483-4522-bcd7-ac396ac626b1 | power
 on    | active             | False       |
| f1aa6356-40a0-41de-be1b-fa6033c9affb | None | 05fbfb44-ccd9-475d-b263-58b2deaf8554 | power
 on    | active             | False       |
| f02357a3-6f9b-46ae-b31f-1a21f6d33543 | None | dd0596b1-bd35-451a-85bc-c635e7fa6d14 | power
 on    | active             | False       |
| ca1153d6-ffaf-481a-ac9b-bc2afc450152 | None | 96d2725c-9c70-4a66-9d3c-4a0356faf1c0 | power
 on    | active             | False       |
```

```
| 8f338102-c114-4a7a-94f0-9e1a54494519 | None | 85a9a708-5eae-4ea2-8b29-dc2acd6e515d | power
 on    | active              | False       |
| 5d3d3525-2528-4801-b885-6c4b340393a6 | None | 315c7aea-acef-4341-aa9e-bcd594cae592 | power
 on    | active              | False       |
| ac21208b-36fd-4404-8e68-53a90df3a29f | None | 9f0b2ff3-5234-42e9-81dd-c0ef5e454137 | power
 on    | active              | False       |
| a6d92bfc-0136-4c22-9988-0108df775a03 | None | 2a3e2086-3516-40ac-a584-3714e91858f5 | power
 on    | active              | False       |
| 5f0593b7-31de-4291-b43f-a549699cd470 | None | f4cc50d4-441e-4728-9984-53df29f0b7f7 | power
 on    | active              | False       |
| 99225e1b-085e-4ef7-8173-4687900b741a | None | 200a918e-abb3-4539-a1c4-7e30f2d8ebc2 | power
 on    | active              | False       |
| c6ec143b-a522-4d69-ab31-5b4934ad3c42 | None | 7c675ed5-17d9-47ad-a2ef-592353e27713 | power
 on    | active              | False       |
| e1026c43-f2a3-44ad-a385-4d4462552977 | None | 45b45041-656f-4ee1-8be2-976c71a35b1f | power
 on    | active              | False       |
| 122188ea-09ae-486c-b225-17cf0defe025 | None | bd38818e-36ca-4fd9-a65f-c4b0e5b34977 | power
 on    | active              | False       |
| f6ecf896-6e5e-4735-8727-942478dee58a | None | 82a79351-5520-4e89-ae19-48c7b6f6b39f | power
 on    | active              | False       |
| e6db159e-008e-4186-8967-92a9faeee368 | None | 986affe6-23ba-48ba-ae4e-0d2226aabf55 | power
 on    | active              | False       |
| 44f3a434-eaf8-4b1a-97e5-6340d277fa4e | None | 1f385454-3ddb-40bd-bc6e-a55ad69fff47 | power
 on    | active              | False       |
| 7ab70571-64ea-439b-a0f4-34147d01dfbf | None | 6f9f76ac-3cf7-4002-94ba-39bc6f0b4c40 | power
 on    | active              | False       |
| 6d478a22-874c-4611-834d-21f4809f90ce | None | 8e37407f-c784-4f5f-942f-2e2c36aa3fa4 | power
 on    | active              | False       |
| 0a57a5ad-d160-477e-807f-11997307bc9c | None | 25b53356-9f02-4810-b722-efb6fd887879 | power
 on    | active              | False       |
| 6fff3d83-ed37-4934-89e0-d632aeb37b15 | None | 0ea048c0-6f4b-460d-99b2-796dd694c226 | power
 on    | active              | False       |
| 5496919c-c269-4860-b49a-e0d103a6a460 | None | 6a8e05aa-26fe-43bb-b464-ede86b9f4639 | power
 on    | active              | False       |
| 513b936d-1c52-4b0a-9ac4-4101fe812f07 | None | b92c5720-7db9-417b-b3d5-023046788c8e | power
 on    | active              | False       |
+--------------------------------------+------+--------------------------------------+--------+-------------+---------+
```

Viewing the OpenStack Server List

Log on to the server on which OSP-D is running to ensure that stack components and verify they are active and running the same image by executing the following command:

openstack server list

Example command output:

```
+--------------------------------------+-------------------------+--------+-------------------
| ID                                   | Name                    | Status | Networks
          | Image Name                     |
+--------------------------------------+-------------------------+--------+-------------------
| 9f0b2ff3-5234-42e9-81dd-c0ef5e454137 | tb3-ultram-compute-3    | ACTIVE |
ctlplane=192.200.0.133 | overcloud-full_20170620T011048 |
| 25b53356-9f02-4810-b722-efb6fd887879 | tb3-ultram-compute-15   | ACTIVE |
ctlplane=192.200.0.131 | overcloud-full_20170620T011048 |
| 986affe6-23ba-48ba-ae4e-0d2226aabf55 | tb3-ultram-compute-11   | ACTIVE |
ctlplane=192.200.0.128 | overcloud-full_20170620T011048 |
| 45b45041-656f-4ee1-8be2-976c71a35b1f | tb3-ultram-compute-8    | ACTIVE |
ctlplane=192.200.0.130 | overcloud-full_20170620T011048 |
| bd38818e-36ca-4fd9-a65f-c4b0e5b34977 | tb3-ultram-compute-9    | ACTIVE |
ctlplane=192.200.0.127 | overcloud-full_20170620T011048 |
| 82a79351-5520-4e89-ae19-48c7b6f6b39f | tb3-ultram-compute-10   | ACTIVE |
ctlplane=192.200.0.126 | overcloud-full_20170620T011048 |
| 1f385454-3ddb-40bd-bc6e-a55ad69fff47 | tb3-ultram-compute-12   | ACTIVE |
ctlplane=192.200.0.118 | overcloud-full_20170620T011048 |
| 8e37407f-c784-4f5f-942f-2e2c36aa3fa4 | tb3-ultram-compute-14   | ACTIVE |
ctlplane=192.200.0.117 | overcloud-full_20170620T011048 |
| 315c7aea-acef-4341-aa9e-bcd594cae592 | tb3-ultram-compute-2    | ACTIVE |
ctlplane=192.200.0.114 | overcloud-full_20170620T011048 |
```

```
| 2a3e2086-3516-40ac-a584-3714e91858f5 | tb3-ultram-compute-4     | ACTIVE |
ctlplane=192.200.0.120 | overcloud-full_20170620T011048 |
| b92c5720-7db9-417b-b3d5-023046788c8e | tb3-ultram-osd-compute-2 | ACTIVE |
ctlplane=192.200.0.110 | overcloud-full_20170620T011048 |
| 7c675ed5-17d9-47ad-a2ef-592353e27713 | tb3-ultram-compute-7     | ACTIVE |
ctlplane=192.200.0.111 | overcloud-full_20170620T011048 |
| 0ea048c0-6f4b-460d-99b2-796dd694c226 | tb3-ultram-osd-compute-0 | ACTIVE |
ctlplane=192.200.0.112 | overcloud-full_20170620T011048 |
| f4cc50d4-441e-4728-9984-53df29f0b7f7 | tb3-ultram-compute-5     | ACTIVE |
ctlplane=192.200.0.108 | overcloud-full_20170620T011048 |
| dd0596b1-bd35-451a-85bc-c635e7fa6d14 | tb3-ultram-controller-2  | ACTIVE |
ctlplane=192.200.0.115 | overcloud-full_20170620T011048 |
| 85a9a708-5eae-4ea2-8b29-dc2acd6e515d | tb3-ultram-compute-1     | ACTIVE |
ctlplane=192.200.0.102 | overcloud-full_20170620T011048 |
| bc903f51-8483-4522-bcd7-ac396ac626b1 | tb3-ultram-controller-0  | ACTIVE |
ctlplane=192.200.0.105 | overcloud-full_20170620T011048 |
| 6a8e05aa-26fe-43bb-b464-ede86b9f4639 | tb3-ultram-osd-compute-1 | ACTIVE |
ctlplane=192.200.0.106 | overcloud-full_20170620T011048 |
| 200a918e-abb3-4539-a1c4-7e30f2d8ebc2 | tb3-ultram-compute-6     | ACTIVE |
ctlplane=192.200.0.109 | overcloud-full_20170620T011048 |
| 05fbfb44-ccd9-475d-b263-58b2deaf8554 | tb3-ultram-controller-1  | ACTIVE |
ctlplane=192.200.0.113 | overcloud-full_20170620T011048 |
| 96d2725c-9c70-4a66-9d3c-4a0356faf1c0 | tb3-ultram-compute-0     | ACTIVE |
ctlplane=192.200.0.107 | overcloud-full_20170620T011048 |
| 6f9f76ac-3cf7-4002-94ba-39bc6f0b4c40 | tb3-ultram-compute-13    | ACTIVE |
ctlplane=192.200.0.103 | overcloud-full_20170620T011048 |
+--------------------------------------+--------------------------+--------+-------------------------+-----------------------------+
```

Viewing the OpenStack Stack Resource List

Log on to the server on which OSP-D is running to view the stack resources and their status by executing the following command:

openstack stack resource list tb5-ultra-m

Example command output:

```
+--------------------------------------+------------------------------------------+-------------------------------------+-----------------+----------------------+
| resource_name                          | physical_resource_id                 |
resource_type                          | resource_status | updated_time         |
+--------------------------------------+------------------------------------------+-------------------------------------+-----------------+----------------------+
| UpdateWorkflow                         | 94270702-cd8b-4441-a09e-5c9da0c2d02b   |
OS::TripleO::Tasks::UpdateWorkflow      | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
| CephStorageHostsDeployment             | 196dbba7-5d66-4a9c-9308-f47ff4ddbe2d   |
OS::Heat::StructuredDeployments         | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
| OsdComputeAllNodesDeployment           | 6a5775c0-03d8-453f-92d8-be6ea5aed853   |
OS::Heat::StructuredDeployments         | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
| BlockStorageHostsDeployment            | 97b2f70a-c295-4437-9222-8248ec30badf   |
OS::Heat::StructuredDeployments         | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
| CephStorage                            | 1bc20bb0-516a-4eb5-85e2-be9d30e2f6e8   |
OS::Heat::ResourceGroup                 | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
| AllNodesDeploySteps                    | da9ead69-b83e-4cc9-86e8-8d823c02843b   |
OS::TripleO::PostDeploySteps            | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
| CephStorageAllNodesDeployment          | e5ee9df8-fae1-4641-9cfb-038c8f4eca85   |
OS::Heat::StructuredDeployments         | CREATE_COMPLETE | 2017-06-27T22:04:00Z |
```

Verifying Node Reachability

Log on to the server on which OSP-D is running to ensure the node reachability and availability by executing the following command:

for i in $(nova list| grep ACTIVE| awk '{print $12}' | sed 's\ctlplane=\\g') ; do ssh heat-admin@${i} uptime ; done

This command establishes an SSH session with each node and report the system uptime. Investigate any node that does not reply or has an unexpected uptime.

Example command output:

```
14:47:10 up 18:15,  0 users,  load average: 0.01, 0.02, 0.05
14:47:11 up 18:14,  0 users,  load average: 9.50, 9.15, 12.32
14:47:11 up 18:14,  0 users,  load average: 9.41, 9.09, 12.26
14:47:11 up 18:14,  0 users,  load average: 10.41, 10.28, 10.49
14:47:12 up 18:15,  0 users,  load average: 0.00, 0.02, 0.05
14:47:12 up 18:14,  0 users,  load average: 0.18, 0.06, 0.06
14:47:12 up 18:15,  0 users,  load average: 0.00, 0.03, 0.05
14:47:12 up 18:15,  0 users,  load average: 0.00, 0.01, 0.05
14:47:13 up 18:14,  0 users,  load average: 0.02, 0.02, 0.05
14:47:13 up 18:14,  0 users,  load average: 8.23, 8.66, 12.29
14:47:13 up 18:14,  0 users,  load average: 8.76, 8.87, 12.14
14:47:14 up 18:15,  0 users,  load average: 0.01, 0.04, 0.05
14:47:14 up 18:15,  0 users,  load average: 9.30, 9.08, 10.12
14:47:14 up 18:15,  0 users,  load average: 0.01, 0.06, 0.05
14:47:14 up 18:14,  0 users,  load average: 8.31, 8.61, 11.96
14:47:15 up 18:14,  0 users,  load average: 17.08, 12.09, 11.06
14:47:15 up 17:09,  0 users,  load average: 1.64, 1.33, 1.10
14:47:15 up 17:04,  0 users,  load average: 1.02, 0.77, 0.79
14:47:16 up 16:58,  0 users,  load average: 0.55, 0.63, 0.72
14:47:16 up 23:46,  0 users,  load average: 2.68, 3.46, 3.89
14:47:16 up 1 day, 5 min,  0 users,  load average: 4.10, 4.27, 4.44
14:47:17 up 23:53,  0 users,  load average: 1.90, 2.32, 2.24
```

Verify NTP is running

Log on to the server on which OSP-D is running to ensure that NTP is running on all nodes in the cluster by executing the following command:

for i in $(nova list| grep ACTIVE| awk '{print $12}' | sed 's\ctlplane=\\g') ; do ssh heat-admin@${i} systemctl status ntpd |grep Active; done
This command establishes an SSH session with each node and lists the ntpd status.

Example command output:

```
   Active: active (running) since Tue 2017-07-11 20:32:25 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:28 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:50 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:28 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:14 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:30 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:22 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:16 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:35 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:31 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:30 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:25 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:19 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:14 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:41 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 20:32:30 UTC; 18h ago
   Active: active (running) since Tue 2017-07-11 21:37:32 UTC; 17h ago
   Active: active (running) since Tue 2017-07-11 21:43:16 UTC; 17h ago
   Active: active (running) since Tue 2017-07-11 21:48:57 UTC; 17h ago
   Active: active (running) since Tue 2017-07-11 15:01:30 UTC; 23h ago
   Active: active (running) since Tue 2017-07-11 14:42:10 UTC; 24h ago
   Active: active (running) since Tue 2017-07-11 14:54:06 UTC; 23h ago
```

Check the NTP status on the server on which OSP-D is running by executing the following command:

systemctl status ntpd |grep Active
Investigate any node that is not actively running NTP.

Checking OSP-D Server Health

Verifying VM and Other Service Status and Quotas

Log on to the server on which OSP-D is running to verify that Overcloud VMs are active and running by executing the following commands:

cd /home/stack
source ~/<*stack_name*>**rc-core**
nova list

Note Overcloud VM status can also be checked through the Horizon GUI.

Example command output:

```
+--------------------------------------+------------------------------------+--------+------------+-------------+----------------------------------------------------------------------------------------+
| ID                                   | Name
              | Status | Task State | Power State | Networks


                          |
+--------------------------------------+------------------------------------+--------+------------+-------------+----------------------------------------------------------------------------------------+
| 407891a2-85bb-4b84-a023-bca4ff304fc5 | auto-deploy-vm-uas-0
              | ACTIVE | -          | Running     | mgmt=172.16.181.21, 10.84.123.13


                          |
| bb4c06c5-b328-47bd-ac57-a72a9b4bb496 | auto-it-vm-uas-0
              | ACTIVE | -          | Running     | mgmt=172.16.181.19, 10.84.123.12


                          |
| fc0e47d3-e59e-41a3-9d8d-99371de1c4c5 | tb3-bxb-autovnf1-uas-0
              | ACTIVE | -          | Running     |
tb3-bxb-autovnf1-uas-orchestration=172.17.180.10; tb3-bxb-autovnf1-uas-management=172.17.181.8

                                                                  |
| 8056eff1-913e-479a-ac44-22eba42ceee1 | tb3-bxb-autovnf1-uas-1
              | ACTIVE | -          | Running     |
tb3-bxb-autovnf1-uas-orchestration=172.17.180.6; tb3-bxb-autovnf1-uas-management=172.17.181.12

                                                                  |
| 4e9fab14-dad0-4789-bc52-1fac3e40b7cc | tb3-bxb-autovnf1-uas-2
              | ACTIVE | -          | Running     |
tb3-bxb-autovnf1-uas-orchestration=172.17.180.13; tb3-bxb-autovnf1-uas-management=172.17.181.3

                                                                  |
| 1a4e65e3-9f9d-429f-a604-6dfb45ef2a45 | tb3-bxb-vnfm1-ESC-0
              | ACTIVE | -          | Running     |
tb3-bxb-autovnf1-uas-orchestration=172.17.180.3; tb3-bxb-autovnf1-uas-management=172.17.181.4

                                                                   |
| 7f4ec2dc-e8a8-4f6c-bfce-8f29735e9fca | tb3-bxb-vnfm1-ESC-1
              | ACTIVE | -          | Running     |
tb3-bxb-autovnf1-uas-orchestration=172.17.180.14; tb3-bxb-autovnf1-uas-management=172.17.181.5

                                                                  |
| 1c9fc0bd-dc16-426f-b387-c2b75b3a1c16 |
tb3-bxb-vnfm1-em_tb3-bx_0_190729a1-c703-4e15-b0b3-795e2e876f55 | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.4;
tb3-bxb-autovnf1-uas-management=172.17.181.9

                    |
| 9a407a06-929a-49ce-8bae-4df35b5f8b40 |
tb3-bxb-vnfm1-em_tb3-bx_0_92c5224b-1f1f-4f3f-8ac8-137be69ce473 | ACTIVE | -          |
```

```
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.5;
tb3-bxb-autovnf1-uas-management=172.17.181.10

                |
| e4528022-6e7b-43f9-94f6-a6ab6289478d |
tb3-bxb-vnfm1-em_tb3-bx_0_d9f7ecb2-a7dc-439b-b492-5ce0402264ea | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.2;
tb3-bxb-autovnf1-uas-management=172.17.181.7

                |
| 2ca11e5b-8eec-456d-9001-1f2600605ad4 |
vnfd1-deployment_c1_0_5b287829-6a9d-4c0a-97d0-a5e0f645b767     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.16;
tb3-bxb-vnfm1-di-internal1=192.168.1.4; tb3-bxb-autovnf1-uas-management=172.17.181.15;
tb3-bxb-vnfm1-di-internal2=192.168.2.5
                  |
| 0bdbd9e3-926a-4abe-81b3-95dc42ea0676 |
vnfd1-deployment_c2_0_7074a450-5268-4c94-965b-8fb809410d14     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.15;
tb3-bxb-vnfm1-di-internal1=192.168.1.2; tb3-bxb-autovnf1-uas-management=172.17.181.18;
tb3-bxb-vnfm1-di-internal2=192.168.2.6
                  |
| 8b07a9b1-139f-4a12-b16e-d35cb17f6668 |
vnfd1-deployment_s10_0_f6d110f9-9e49-43fe-be14-4ab87ca3334c    | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.7;
tb3-bxb-vnfm1-di-internal1=192.168.1.8; tb3-bxb-vnfm1-service-network1=10.10.10.3,
10.10.10.10; tb3-bxb-vnfm1-service-network2=20.20.20.5, 20.20.20.4;
tb3-bxb-vnfm1-di-internal2=192.168.2.12      |
| 4ff0ce2e-1d97-4056-a7aa-018412c0385d |
vnfd1-deployment_s3_0_5380ef6c-6fe3-4e92-aa44-d94ef6e94235     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.19;
tb3-bxb-vnfm1-di-internal1=192.168.1.5; tb3-bxb-vnfm1-service-network1=10.10.10.7, 10.10.10.2;
 tb3-bxb-vnfm1-service-network2=20.20.20.9, 20.20.20.6; tb3-bxb-vnfm1-di-internal2=192.168.2.8
       |
| 3954cd6e-0f12-4d4b-8558-2e035c126d9a |
vnfd1-deployment_s4_0_e5ae4aa9-a90e-4bfe-aaff-82ffd8f7fe34     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.8;
tb3-bxb-vnfm1-di-internal1=192.168.1.9; tb3-bxb-vnfm1-service-network1=10.10.10.13,
10.10.10.8; tb3-bxb-vnfm1-service-network2=20.20.20.12, 20.20.20.10;
tb3-bxb-vnfm1-di-internal2=192.168.2.3     |
| 2cc6728c-2982-42bf-bb8b-198a14fdcb31 |
vnfd1-deployment_s5_0_1d57c15d-a1de-40d4-aac2-1715f01ac50a     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.17;
tb3-bxb-vnfm1-di-internal1=192.168.1.7; tb3-bxb-vnfm1-service-network1=10.10.10.5,
10.10.10.18; tb3-bxb-vnfm1-service-network2=20.20.20.11, 20.20.20.2;
tb3-bxb-vnfm1-di-internal2=192.168.2.4     |
| 876cc650-ae8b-497b-805a-24a305be6c13 |
vnfd1-deployment_s6_0_05e13a62-623c-4749-ae2a-15c70dd12e16     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.11;
tb3-bxb-vnfm1-di-internal1=192.168.1.6; tb3-bxb-vnfm1-service-network1=10.10.10.12,
10.10.10.9; tb3-bxb-vnfm1-service-network2=20.20.20.13, 20.20.20.18;
tb3-bxb-vnfm1-di-internal2=192.168.2.16   |
| 89f7245e-c2f7-4041-b5e6-1eee48641cfd |
vnfd1-deployment_s7_0_3a4d7273-e808-4b5f-8877-7aa182483d93     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.24;
tb3-bxb-vnfm1-di-internal1=192.168.1.12; tb3-bxb-vnfm1-service-network1=10.10.10.14,
10.10.10.6; tb3-bxb-vnfm1-service-network2=20.20.20.20, 20.20.20.8;
tb3-bxb-vnfm1-di-internal2=192.168.2.7    |
| 535b0bca-d3c5-4d99-ba41-9953da6339f4 |
vnfd1-deployment_s8_0_1e0f3ebf-b6e0-4bfe-9b1c-985dc32e1519     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.18;
tb3-bxb-vnfm1-di-internal1=192.168.1.14; tb3-bxb-vnfm1-service-network1=10.10.10.17,
10.10.10.11; tb3-bxb-vnfm1-service-network2=20.20.20.17, 20.20.20.15;
tb3-bxb-vnfm1-di-internal2=192.168.2.9  |
| dfdffafb-a624-4063-bae6-63c4a757473f |
vnfd1-deployment_s9_0_26db8332-8dac-43fc-84c5-71a8b975fd17     | ACTIVE | -          |
Running     | tb3-bxb-autovnf1-uas-orchestration=172.17.180.22;
tb3-bxb-vnfm1-di-internal1=192.168.1.10; tb3-bxb-vnfm1-service-network1=10.10.10.21,
10.10.10.24; tb3-bxb-vnfm1-service-network2=20.20.20.23, 20.20.20.22;
tb3-bxb-vnfm1-di-internal2=192.168.2.19 |
+--------+---------------+--+--+--+--------------------------------------------------------+
```

Checking Cinder Type

Log on to the server on which OSP-D is running to check the Cinder vilome type by executing the following commands:

cd /home/stack
source ~/<*stack_name*>**rc-core**
cinder type-list
Example command output:

```
+--------------------------------------+------+-------------+-----------+
| ID                                   | Name | Description | Is_Public |
+--------------------------------------+------+-------------+-----------+
| 208ef179-dfe4-4735-8a96-e7beee472944 | LUKS | -           | True      |
+--------------------------------------+------+-------------+-----------+
```

cinder type-show LUKS
Example command output:

```
+---------------------------------+--------------------------------------+
| Property                        | Value                                |
+---------------------------------+--------------------------------------+
| description                     | None                                 |
| extra_specs                     | {}                                   |
| id                              | bf855b0f-8b3f-42bf-9497-05013b4ddad9 |
| is_public                       | True                                 |
| name                            | LUKS                                 |
| os-volume-type-access:is_public | True                                 |
| qos_specs_id                    | None                                 |
+---------------------------------+--------------------------------------+
```

Checking Core Project (Tenant) and User Core

Log on to the server on which OSP-D is running to check the core projects and users by executing the following commands:

cd /home/stack
source~/<*stack_name*> **rc-core**
openstack project list
Example command output:

```
+----------------------------------+---------+
| ID                               | Name    |
+----------------------------------+---------+
| 271ab207a197465f9d166c2dc7304b18 | core    |
| 52547e0fca994cd682aa733b941d0f68 | service |
| 9543ad9db4dd422ea5aedf04756d3682 | admin   |
+----------------------------------+---------+
```

openstack project show core
Example command output:

```
+-------------+----------------------------------+
| Field       | Value                            |
+-------------+----------------------------------+
| description | core tenant                      |
| enabled     | True                             |
| id          | 271ab207a197465f9d166c2dc7304b18 |
| name        | core                             |
| properties  |                                  |
+-------------+----------------------------------+
```

openstack project show service
Example command output:

```
+-------------+----------------------------------+
| Field       | Value                            |
+-------------+----------------------------------+
```

```
| description | Tenant for the openstack services |
| enabled     | True                              |
| id          | 52547e0fca994cd682aa733b941d0f68  |
| name        | service                           |
| properties  |                                   |
+-------------+-----------------------------------+
```

openstack project show admin
Example command output:

```
+-------------+----------------------------------+
| Field       | Value                            |
+-------------+----------------------------------+
| description | admin tenant                     |
| enabled     | True                             |
| id          | 9543ad9db4dd422ea5aedf04756d3682 |
| name        | admin                            |
| properties  |                                  |
+-------------+----------------------------------+
```

openstack user list
Example command output:

```
+----------------------------------+------------+
| ID                               | Name       |
+----------------------------------+------------+
| 1ac7208b033a41ccba805d86bf60dbb7 | admin      |
| a6adac4ee79c4206a29de5165d7c7a6a | neutron    |
| 79da40fe88c64de7a93bc691a42926ea | heat       |
| ac3887fec44c483d8780f4500f6f856b | gnocchi    |
| aaee103013404bdeb5f9b172ac019daa | aodh       |
| 525048a99816474d91d692d9516e951c | nova       |
| 8d6688db8d19411080eeb4c84c1d586b | glance     |
| a6b9fb8312be4e4d91c9cc2e7e9ad6be | ceilometer |
| 9aadd12171474d1e8bcbacf890e070ab | cinder     |
| d2ee641a72c4493995de70a1a9671f2b | heat-cfn   |
| 7fbb088c15e1428ab6ce677aad5415f4 | swift      |
| 828cbf69cf564747a81bb313208a1c21 | core       |
| 40563efc469d4c1295de0d6d4cf545c2 | tom        |
+----------------------------------+------------+
```

openstack user show core
Example command output:

```
+------------+----------------------------------+
| Field      | Value                            |
+------------+----------------------------------+
| email      | None                             |
| enabled    | True                             |
| id         | 828cbf69cf564747a81bb313208a1c21 |
| name       | core                             |
| project_id | 271ab207a197465f9d166c2dc7304b18 |
| username   | core                             |
+------------+----------------------------------+
```

openstack role list
Example command output:

```
+----------------------------------+------------------+
| ID                               | Name             |
+----------------------------------+------------------+
| 315d3058519a4b1a9385e11aa5ffe25b | admin            |
| 585de968688e4257bc76f6dec13752cb | ResellerAdmin    |
| 9717fe8079ba49e9ba9eadd5a37689e7 | swiftoperator    |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_         |
| d75dcf507bfa4a6abee3aee3bb0323c6 | heat_stack_user  |
+----------------------------------+------------------+
```

openstack role show admin
Example command output:

```
+-----------+----------------------------------+
| Field     | Value                            |
+-----------+----------------------------------+
| domain_id | None                             |
| id        | 315d3058519a4b1a9385e11aa5ffe25b |
```

```
| name       | admin                              |
+------------+------------------------------------+
```

Checking Nova/Neutron Security Groups

Log on to the server on which OSP-D is running to check Nova and Neutron security groups by executing the following commands:

nova secgroup-list
Example command output:

```
WARNING: Command secgroup-list is deprecated and will be removed after Nova 15.0.0 is
released. Use python-neutronclient or python-openstackclient instead.
+--------------------------------------+---------+------------------------+
| Id                                   | Name    | Description            |
+--------------------------------------+---------+------------------------+
| ce308d67-7645-43c1-a83e-89d3871141a2 | default | Default security group |
+--------------------------------------+---------+------------------------+
```

neutron security-group-list
Example command output:

```
+--------------------------------------+---------+-----------------------------------------------------------------------+
| id                                   | name    | security_group_rules
                                |
+--------------------------------------+---------+-----------------------------------------------------------------------+
| 4007a7a4-e7fa-4ad6-bc74-fc0b20f0b60c | default | egress, IPv4
                                |
|                                      |         | egress, IPv6
                                |
|                                      |         | ingress, IPv4, remote_group_id:
4007a7a4-e7fa-4ad6-bc74-fc0b20f0b60c |
|                                      |         | ingress, IPv6, remote_group_id:
4007a7a4-e7fa-4ad6-bc74-fc0b20f0b60c |
| 8bee29ae-88c0-4d5d-b27a-a123f20b6858 | default | egress, IPv4
                                |
|                                      |         | egress, IPv6
                                |
|                                      |         | ingress, IPv4, 1-65535/tcp,
remote_ip_prefix: 0.0.0.0/0              |
|                                      |         | ingress, IPv4, 1-65535/udp,
remote_ip_prefix: 0.0.0.0/0              |
|                                      |         | ingress, IPv4, icmp, remote_ip_prefix:
0.0.0.0/0                        |
|                                      |         | ingress, IPv4, remote_group_id:
8bee29ae-88c0-4d5d-b27a-a123f20b6858 |
|                                      |         | ingress, IPv6, remote_group_id:
8bee29ae-88c0-4d5d-b27a-a123f20b6858 |
| b6b27428-35a3-4be4-af9b-38559132d28e | default | egress, IPv4
                                |
|                                      |         | egress, IPv6
                                |
|                                      |         | ingress, IPv4, remote_group_id:
b6b27428-35a3-4be4-af9b-38559132d28e |
|                                      |         | ingress, IPv6, remote_group_id:
b6b27428-35a3-4be4-af9b-38559132d28e |
| ce308d67-7645-43c1-a83e-89d3871141a2 | default | egress, IPv4
                                |
|                                      |         | egress, IPv6
                                |
|                                      |         | ingress, IPv4, 1-65535/tcp,
remote_ip_prefix: 0.0.0.0/0              |
|                                      |         | ingress, IPv4, 1-65535/udp,
remote_ip_prefix: 0.0.0.0/0              |
|                                      |         | ingress, IPv4, icmp, remote_ip_prefix:
0.0.0.0/0                        |
|                                      |         | ingress, IPv4, remote_group_id:
ce308d67-7645-43c1-a83e-89d3871141a2 |
|                                      |         | ingress, IPv6, remote_group_id:
ce308d67-7645-43c1-a83e-89d3871141a2 |
```

```
+-------------------------------+--------+-------------------------------------------------------------------+
```

neutron security-group-show ce308d67-7645-43c1-a83e-89d3871141a2
Example command output:

```
+----------------------+-------------------------------------------------------------------------+
| Field                | Value
  |
+----------------------+-------------------------------------------------------------------------+
| created_at           | 2017-06-03T04:57:01Z
  |
| description          | Default security group
  |
| id                   | ce308d67-7645-43c1-a83e-89d3871141a2
  |
| name                 | default
  |
| project_id           | 271ab207a197465f9d166c2dc7304b18
  |
| revision_number      | 4
  |
| security_group_rules | {
  |
|                      |      "remote_group_id": null,
  |
|                      |      "direction": "egress",
  |
|                      |      "protocol": null,
  |
|                      |      "description": null,
  |
|                      |      "ethertype": "IPv4",
  |
|                      |      "remote_ip_prefix": null,
  |
|                      |      "port_range_max": null,
  |
|                      |      "updated_at": "2017-06-03T04:57:01Z",
  |
|                      |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
  |
|                      |      "port_range_min": null,
  |
|                      |      "revision_number": 1,
  |
|                      |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
  |
|                      |      "created_at": "2017-06-03T04:57:01Z",
  |
|                      |      "project_id": "271ab207a197465f9d166c2dc7304b18",
  |
|                      |      "id": "337838dd-0612-47f8-99e8-7d4f58dc09d6"
  |
|                      | }
  |
|                      | {
  |
|                      |      "remote_group_id": null,
  |
|                      |      "direction": "ingress",
  |
|                      |      "protocol": "udp",
  |
|                      |      "description": "",
  |
|                      |      "ethertype": "IPv4",
  |
|                      |      "remote_ip_prefix": "0.0.0.0/0",
  |
|                      |      "port_range_max": 65535,
  |
|                      |      "updated_at": "2017-06-03T04:57:20Z",
  |
```

```
|                       |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
|
|                       |      "port_range_min": 1,
|
|                       |      "revision_number": 1,
|
|                       |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "created_at": "2017-06-03T04:57:20Z",
|
|                       |      "project_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "id": "48b04902-d617-4e25-ad0d-4d087128f3b9"
|
|                       | }
|
|                       | {
|
|                       |      "remote_group_id": null,
|
|                       |      "direction": "ingress",
|
|                       |      "protocol": "icmp",
|
|                       |      "description": "",
|
|                       |      "ethertype": "IPv4",
|
|                       |      "remote_ip_prefix": "0.0.0.0/0",
|
|                       |      "port_range_max": null,
|
|                       |      "updated_at": "2017-06-03T04:57:33Z",
|
|                       |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
|
|                       |      "port_range_min": null,
|
|                       |      "revision_number": 1,
|
|                       |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "created_at": "2017-06-03T04:57:33Z",
|
|                       |      "project_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "id": "68913f31-6788-4473-8b3b-90a264e9ef62"
|
|                       | }
|
|                       | {
|
|                       |      "remote_group_id": null,
|
|                       |      "direction": "ingress",
|
|                       |      "protocol": "tcp",
|
|                       |      "description": "",
|
|                       |      "ethertype": "IPv4",
|
|                       |      "remote_ip_prefix": "0.0.0.0/0",
|
|                       |      "port_range_max": 65535,
|
|                       |      "updated_at": "2017-06-03T04:57:02Z",
|
|                       |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
|
|                       |      "port_range_min": 1,
|
|                       |      "revision_number": 1,
```

```
|
|                       |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "created_at": "2017-06-03T04:57:02Z",
|
|                       |      "project_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "id": "85ece95b-d361-4986-8db0-78d1a404dd3c"
|
|                       | }
|
|                       | {
|
|                       |      "remote_group_id": null,
|
|                       |      "direction": "egress",
|
|                       |      "protocol": null,
|
|                       |      "description": null,
|
|                       |      "ethertype": "IPv6",
|
|                       |      "remote_ip_prefix": null,
|
|                       |      "port_range_max": null,
|
|                       |      "updated_at": "2017-06-03T04:57:01Z",
|
|                       |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
|
|                       |      "port_range_min": null,
|
|                       |      "revision_number": 1,
|
|                       |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "created_at": "2017-06-03T04:57:01Z",
|
|                       |      "project_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "id": "88320991-5232-44f6-b74b-8cfe934165d0"
|
|                       | }
|
|                       | {
|
|                       |      "remote_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
|
|                       |      "direction": "ingress",
|
|                       |      "protocol": null,
|
|                       |      "description": null,
|
|                       |      "ethertype": "IPv4",
|
|                       |      "remote_ip_prefix": null,
|
|                       |      "port_range_max": null,
|
|                       |      "updated_at": "2017-06-03T04:57:01Z",
|
|                       |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
|
|                       |      "port_range_min": null,
|
|                       |      "revision_number": 1,
|
|                       |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
|
|                       |      "created_at": "2017-06-03T04:57:01Z",
|
```

```
|                      |      "project_id": "271ab207a197465f9d166c2dc7304b18",
  |
|                      |      "id": "ba306ee2-d21f-48be-9de2-7f04bea5e43a"
  |
|                      | }
  |
|                      | {
  |
|                      |      "remote_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
  |
|                      |      "direction": "ingress",
  |
|                      |      "protocol": null,
  |
|                      |      "description": null,
  |
|                      |      "ethertype": "IPv6",
  |
|                      |      "remote_ip_prefix": null,
  |
|                      |      "port_range_max": null,
  |
|                      |      "updated_at": "2017-06-03T04:57:01Z",
  |
|                      |      "security_group_id": "ce308d67-7645-43c1-a83e-89d3871141a2",
  |
|                      |      "port_range_min": null,
  |
|                      |      "revision_number": 1,
  |
|                      |      "tenant_id": "271ab207a197465f9d166c2dc7304b18",
  |
|                      |      "created_at": "2017-06-03T04:57:01Z",
  |
|                      |      "project_id": "271ab207a197465f9d166c2dc7304b18",
  |
|                      |      "id": "deb7752c-e642-462e-92f0-5dff983f0739"
  |
|                      | }
  |
| tenant_id            | 271ab207a197465f9d166c2dc7304b18
  |
| updated_at           | 2017-06-03T04:57:33Z
  |
+----------------------+--------------------------------------------------------------------------+
```

Checking Tenant Project Default Quotas

Log on to the server on which OSP-D is running to check default project quotas by executing the following commands:

nova quota-show
Example command output:

```
+-----------------------------+----------+
| Quota                       | Limit    |
+-----------------------------+----------+
| instances                   | 1000     |
| cores                       | 1000     |
| ram                         | 51200000 |
| metadata_items              | 128      |
| injected_files              | 100      |
| injected_file_content_bytes | 1024000  |
| injected_file_path_bytes    | 255      |
| key_pairs                   | 100      |
| server_groups               | 10       |
| server_group_members        | 10       |
+-----------------------------+----------+
```

openstack project list | grep core

Example command output:

```
| 271ab207a197465f9d166c2dc7304b18 | core |
```

nova quota-class-show 271ab207a197465f9d166c2dc7304b18
Example command output:

```
+-----------------------------+-------+
| Quota                       | Limit |
+-----------------------------+-------+
| instances                   | 10    |
| cores                       | 20    |
| ram                         | 51200 |
| floating_ips                | 10    |
| fixed_ips                   | -1    |
| metadata_items              | 128   |
| injected_files              | 5     |
| injected_file_content_bytes | 10240 |
| injected_file_path_bytes    | 255   |
| key_pairs                   | 100   |
| security_groups             | 10    |
| security_group_rules        | 20    |
+-----------------------------+-------+
```

neutron quota-show
Example command output:

```
+---------------------+-------+
| Field               | Value |
+---------------------+-------+
| floatingip          | 100   |
| network             | 1000  |
| port                | 4092  |
| rbac_policy         | 10    |
| router              | 100   |
| security_group      | 100   |
| security_group_rule | 300   |
| subnet              | 1000  |
| subnetpool          | -1    |
| trunk               | -1    |
+---------------------+-------+
```

openstack project list | grep core
Example command output:

```
| 271ab207a197465f9d166c2dc7304b18 | core |
```

cinder quota-show 271ab207a197465f9d166c2dc7304b18
Example command output:

```
+----------------------+-------+
| Property             | Value |
+----------------------+-------+
| backup_gigabytes     | 1000  |
| backups              | 10    |
| gigabytes            | 8092  |
| gigabytes_LUKS       | -1    |
| per_volume_gigabytes | -1    |
| snapshots            | 300   |
| snapshots_LUKS       | -1    |
| volumes              | 500   |
| volumes_LUKS         | -1    |
+----------------------+-------+
```

Checking the Nova Hypervisor List

Log on to the server on which OSP-D is running to check the status of nova api on all compute nodes by executing the following command:

nova hypervisor-list

Example command output:

```
+----+--------------------------------------+-------+---------+
| ID | Hypervisor hostname                  | State | Status  |
+----+--------------------------------------+-------+---------+
| 3  | tb3-ultram-compute-7.localdomain     | up    | enabled |
| 6  | tb3-ultram-compute-6.localdomain     | up    | enabled |
| 9  | tb3-ultram-osd-compute-0.localdomain | up    | enabled |
| 12 | tb3-ultram-compute-9.localdomain     | up    | enabled |
| 15 | tb3-ultram-compute-0.localdomain     | up    | enabled |
| 18 | tb3-ultram-compute-14.localdomain    | up    | enabled |
| 21 | tb3-ultram-compute-2.localdomain     | up    | enabled |
| 24 | tb3-ultram-compute-8.localdomain     | up    | enabled |
| 27 | tb3-ultram-compute-13.localdomain    | up    | enabled |
| 30 | tb3-ultram-compute-15.localdomain    | up    | enabled |
| 33 | tb3-ultram-compute-12.localdomain    | up    | enabled |
| 36 | tb3-ultram-compute-5.localdomain     | up    | enabled |
| 39 | tb3-ultram-osd-compute-1.localdomain | up    | enabled |
| 42 | tb3-ultram-compute-10.localdomain    | up    | enabled |
| 45 | tb3-ultram-compute-11.localdomain    | up    | enabled |
| 48 | tb3-ultram-compute-3.localdomain     | up    | enabled |
| 51 | tb3-ultram-osd-compute-2.localdomain | up    | enabled |
| 54 | tb3-ultram-compute-4.localdomain     | up    | enabled |
| 57 | tb3-ultram-compute-1.localdomain     | up    | enabled |
+----+--------------------------------------+-------+---------+
```

Checking the Router Main Configuration

Log on to the server on which OSP-D is running to check the Neutron router by entering the following commands:

neutron router-list

Example command output:

```
+--------------------------------------+------+------------------------------------------+-------------+------+
| id                                   | name | external_gateway_info
                 | distributed | ha   |
+--------------------------------------+------+------------------------------------------+-------------+------+
| 2d0cdee4-bb5e-415b-921c-97caf0aa0cd1 | main | {"network_id":
"1c46790f-cab5-4b1d-afc7-a637fe2dbe08",     | False       | True |
|                                      |      | "enable_snat": true, "external_fixed_ips":
 [{"subnet_id":  |             |      |
|                                      |      | "a23a740e-3ad0-4fb1-8526-3353dfd0010f",
"ip_address":      |             |      |
|                                      |      | "10.169.127.176"}]}
                 |             |      |
+--------------------------------------+------+------------------------------------------+-------------+------+
```

`[stack@lbucs001-ospd ~]$` **neutron router-show 2d0cdee4-bb5e-415b-921c-97caf0aa0cd1**

Example command output:

```
+-------------------------+--------------------------------------------------------------------------+
| Field                   | Value
                                                         |
+-------------------------+--------------------------------------------------------------------------+
| admin_state_up          | True
                                                         |
| availability_zone_hints |
                                                         |
| availability_zones      | nova
                                                         |
| created_at              | 2017-06-03T05:05:08Z
                                                         |
| description             |
                                                         |
| distributed             | False
                                                         |
| external_gateway_info   | {"network_id": "1c46790f-cab5-4b1d-afc7-a637fe2dbe08",
"enable_snat": true, "external_fixed_ips": [{"subnet_id":         |
```

```
|                          | "a23a740e-3ad0-4fb1-8526-3353dfd0010f", "ip_address":
"10.169.127.176"}]}                                              |
| flavor_id                |
                                                        |
| ha                       | True
                                                        |
| id                       | 2d0cdee4-bb5e-415b-921c-97caf0aa0cd1
                                                        |
| name                     | main
                                                        |
| project_id               | 271ab207a197465f9d166c2dc7304b18
                                                        |
| revision_number          | 94
                                                        |
| routes                   |
                                                        |
| status                   | ACTIVE
                                                        |
| tenant_id                | 271ab207a197465f9d166c2dc7304b18
                                                        |
| updated_at               | 2017-07-28T00:44:27Z
                                                        |
+----------------+----------------------------------------------------------------+
```

Checking the External Network Using the core-project-id

Log on to the server on which OSP-D is running to check the external network configuration by entering the following commands:

neutron net-list
Example command output:

```
+--------------------------------------+----------------------------------------------------+--------------------------------------+
| id                                   | name
  | subnets                                              |
+--------------------------------------+----------------------------------------------------+--------------------------------------+
| 1236bd98-5389-42f9-bac8-433997525549 | LBUCS001-AUTOIT-MGMT
  | c63451f2-7e44-432e-94fc-167f6a31e4aa 172.16.182.0/24   |
| 1c46790f-cab5-4b1d-afc7-a637fe2dbe08 | LBUCS001-EXTERNAL-MGMT
  | a23a740e-3ad0-4fb1-8526-3353dfd0010f 10.169.127.160/27 |
| 1c70a9ab-212e-4884-b7d5-4749c44a87b6 | LBPGW101-DI-INTERNAL1
  |                                                        |
| e619b02e-84e0-48d9-9096-f16adc84f1cc | HA network tenant 271ab207a197465f9d166c2dc7304b18
 | cefd5f5f-0c97-4027-b385-ca1a57f2cfac 169.254.192.0/18  |
+--------------------------------------+----------------------------------------------------+--------------------------------------+
```

neutron net-show 1c46790f-cab5-4b1d-afc7-a637fe2dbe08
Example command output:

```
+---------------------------+--------------------------------------+
| Field                     | Value                                |
+---------------------------+--------------------------------------+
| admin_state_up            | True                                 |
| availability_zone_hints   |                                      |
| availability_zones        |                                      |
| created_at                | 2017-06-05T07:18:59Z                 |
| description               |                                      |
| id                        | 1c46790f-cab5-4b1d-afc7-a637fe2dbe08 |
| ipv4_address_scope        |                                      |
| ipv6_address_scope        |                                      |
| is_default                | False                                |
| mtu                       | 1500                                 |
| name                      | LBUCS001-EXTERNAL-MGMT               |
| port_security_enabled     | True                                 |
| project_id                | 271ab207a197465f9d166c2dc7304b18     |
| provider:network_type     | vlan                                 |
| provider:physical_network | datacentre                           |
| provider:segmentation_id  | 101                                  |
| qos_policy_id             |                                      |
```

```
| revision_number           | 6                                    |
| router:external           | True                                 |
| shared                    | False                                |
| status                    | ACTIVE                               |
| subnets                   | a23a740e-3ad0-4fb1-8526-3353dfd0010f |
| tags                      |                                      |
| tenant_id                 | 271ab207a197465f9d166c2dc7304b18     |
| updated_at                | 2017-06-05T07:22:51Z                 |
+---------------------------+--------------------------------------+
```

Note down the **provider:segmentation_id**. In this example, 101 is the vlan for the external interface.

neutron subnet-list
Example command output:

```
+--------------------------------------+-------------------------------------+-------------------+-------------------------------------------+
| id                                   | name                                                  | cidr
            | allocation_pools                          |
+--------------------------------------+-------------------------------------+-------------------+-------------------------------------------+
| a23a740e-3ad0-4fb1-8526-3353dfd0010f | LBUCS001-EXTERNAL-MGMT                                |
10.169.127.160/27 | {"start": "10.169.127.168", "end":       |
|                                      |                                                       |
            | "10.169.127.190"}                         |
| c63451f2-7e44-432e-94fc-167f6a31e4aa | LBUCS001-AUTOIT-MGMT                                  |
172.16.182.0/24   | {"start": "172.16.182.2", "end":         |
|                                      |                                                       |
            | "172.16.182.254"}                         |
| cefd5f5f-0c97-4027-b385-ca1a57f2cfac | HA subnet tenant                                      |
169.254.192.0/18  | {"start": "169.254.192.1", "end":        |
|                                      | 271ab207a197465f9d166c2dc7304b18                      |
            | "169.254.255.254"}                        |
+--------------------------------------+-------------------------------------+-------------------+-------------------------------------------+
```

neutron subnet-show a23a740e-3ad0-4fb1-8526-3353dfd0010f
Example command output:

```
+-------------------+------------------------------------------------------+
| Field             | Value                                                |
+-------------------+------------------------------------------------------+
| allocation_pools  | {"start": "10.169.127.168", "end": "10.169.127.190"} |
| cidr              | 10.169.127.160/27                                    |
| created_at        | 2017-06-05T07:22:51Z                                 |
| description       |                                                      |
| dns_nameservers   |                                                      |
| enable_dhcp       | False                                                |
| gateway_ip        | 10.169.127.163                                       |
| host_routes       |                                                      |
| id                | a23a740e-3ad0-4fb1-8526-3353dfd0010f                 |
| ip_version        | 4                                                    |
| ipv6_address_mode |                                                      |
| ipv6_ra_mode      |                                                      |
| name              | LBUCS001-EXTERNAL-MGMT                               |
| network_id        | 1c46790f-cab5-4b1d-afc7-a637fe2dbe08                 |
| project_id        | 271ab207a197465f9d166c2dc7304b18                     |
| revision_number   | 2                                                    |
| service_types     |                                                      |
| subnetpool_id     |                                                      |
| tenant_id         | 271ab207a197465f9d166c2dc7304b18                     |
| updated_at        | 2017-06-05T07:22:51Z                                 |
+-------------------+------------------------------------------------------+
```

Checking the Staging Network Configuration

Log on to the server on which OSP-D is running to check the staging network configuration by entering the following commands:

neutron subnet-show *<ext-mgmt-id>*
<ext-mgmt-id> is the ID for the external management interface as obtained through the **neutron subnet-list** command output.

Example output:

```
+-------------------+------------------------------------------------------+
| Field             | Value                                                |
+-------------------+------------------------------------------------------+
| allocation_pools  | {"start": "10.169.127.168", "end": "10.169.127.190"} |
| cidr              | 10.169.127.160/27                                    |
| created_at        | 2017-06-05T07:22:51Z                                 |
| description       |                                                      |
| dns_nameservers   |                                                      |
| enable_dhcp       | False                                                |
| gateway_ip        | 10.169.127.163                                       |
| host_routes       |                                                      |
| id                | a23a740e-3ad0-4fb1-8526-3353dfd0010f                 |
| ip_version        | 4                                                    |
| ipv6_address_mode |                                                      |
| ipv6_ra_mode      |                                                      |
| name              | LBUCS001-EXTERNAL-MGMT                               |
| network_id        | 1c46790f-cab5-4b1d-afc7-a637fe2dbe08                 |
| project_id        | 271ab207a197465f9d166c2dc7304b18                     |
| revision_number   | 2                                                    |
| service_types     |                                                      |
| subnetpool_id     |                                                      |
| tenant_id         | 271ab207a197465f9d166c2dc7304b18                     |
| updated_at        | 2017-06-05T07:22:51Z                                 |
+-------------------+------------------------------------------------------+
```

neutron subnet-show *<autoit-mgmt-id>*

<autoit-mgmt-id> is the ID for the AutoIT management interface as obtained through the **neutron subnet-list** command output.

Example output:

```
+-------------------+----------------------------------------------------+
| Field             | Value                                              |
+-------------------+----------------------------------------------------+
| allocation_pools  | {"start": "172.16.182.2", "end": "172.16.182.254"} |
| cidr              | 172.16.182.0/24                                    |
| created_at        | 2017-06-05T07:41:45Z                               |
| description       |                                                    |
| dns_nameservers   |                                                    |
| enable_dhcp       | True                                               |
| gateway_ip        | 172.16.182.1                                       |
| host_routes       |                                                    |
| id                | c63451f2-7e44-432e-94fc-167f6a31e4aa               |
| ip_version        | 4                                                  |
| ipv6_address_mode |                                                    |
| ipv6_ra_mode      |                                                    |
| name              | LBUCS001-AUTOIT-MGMT                               |
| network_id        | 1236bd98-5389-42f9-bac8-433997525549               |
| project_id        | 271ab207a197465f9d166c2dc7304b18                   |
| revision_number   | 2                                                  |
| service_types     |                                                    |
| subnetpool_id     |                                                    |
| tenant_id         | 271ab207a197465f9d166c2dc7304b18                   |
| updated_at        | 2017-06-05T07:41:45Z                               |
+-------------------+----------------------------------------------------+
```

Checking the DI-Internal and Service Network Configurations

Log on to the server on which OSP-D is running to check the DI-internal and service network configuration by entering the following commands:

neutron net-list
Example command output:

```
+------------------------+--------------------------------+----------------------------------+
| id                                   | name
  | subnets                                          |
+------------------------+--------------------------------+----------------------------------+
```

```
| 1236bd98-5389-42f9-bac8-433997525549 | LBUCS001-AUTOIT-MGMT
  | c63451f2-7e44-432e-94fc-167f6a31e4aa 172.16.182.0/24   |
| 1c46790f-cab5-4b1d-afc7-a637fe2dbe08 | LBUCS001-EXTERNAL-MGMT
  | a23a740e-3ad0-4fb1-8526-3353dfd0010f 10.169.127.160/27 |
| 1c70a9ab-212e-4884-b7d5-4749c44a87b6 | LBPGW101-DI-INTERNAL1
  |                                                        |
| e619b02e-84e0-48d9-9096-f16adc84f1cc | HA network tenant 271ab207a197465f9d166c2dc7304b18
 | cefd5f5f-0c97-4027-b385-ca1a57f2cfac 169.254.192.0/18  |
+-----------------------+--------------------------------+----------------------------------+
```

neutron net-show LBPGW101-DI-INTERNAL1

Example command output:

```
+---------------------------+--------------------------------------+
| Field                     | Value                                |
+---------------------------+--------------------------------------+
| admin_state_up            | True                                 |
| availability_zone_hints   |                                      |
| availability_zones        |                                      |
| created_at                | 2017-07-28T22:25:53Z                 |
| description               |                                      |
| id                        | 1c70a9ab-212e-4884-b7d5-4749c44a87b6 |
| ipv4_address_scope        |                                      |
| ipv6_address_scope        |                                      |
| mtu                       | 1500                                 |
| name                      | LBPGW101-DI-INTERNAL1                |
| port_security_enabled     | True                                 |
| project_id                | 271ab207a197465f9d166c2dc7304b18     |
| provider:network_type     | flat                                 |
| provider:physical_network | phys_pcie1_0                         |
| provider:segmentation_id  |                                      |
| qos_policy_id             |                                      |
| revision_number           | 3                                    |
| router:external           | False                                |
| shared                    | True                                 |
| status                    | ACTIVE                               |
| subnets                   |                                      |
| tags                      |                                      |
| tenant_id                 | 271ab207a197465f9d166c2dc7304b18     |
| updated_at                | 2017-07-28T22:25:53Z                 |
+---------------------------+--------------------------------------+
```

neutron subnet-list

Example command output:

```
+------------------------+--------------------------+------------+--------------------------+
| id                                   | name                                         | cidr
            | allocation_pools                        |
+------------------------+--------------------------+------------+--------------------------+
| 96ae7e6e-f2e9-4fa5-a816-769c5a79f8f4 | LBPGW101-DI-INTERNAL1-SUBNET                 |
192.168.1.0/24    | {"start": "192.168.1.2", "end":          |
|                                      |                                              |
            | "192.168.1.254"}                        |
| a23a740e-3ad0-4fb1-8526-3353dfd0010f | LBUCS001-EXTERNAL-MGMT                       |
10.169.127.160/27 | {"start": "10.169.127.168", "end":       |
|                                      |                                              |
            | "10.169.127.190"}                       |
| c63451f2-7e44-432e-94fc-167f6a31e4aa | LBUCS001-AUTOIT-MGMT                         |
172.16.182.0/24   | {"start": "172.16.182.2", "end":         |
|                                      |                                              |
            | "172.16.182.254"}                       |
| cefd5f5f-0c97-4027-b385-ca1a57f2cfac | HA subnet tenant                             |
169.254.192.0/18  | {"start": "169.254.192.1", "end":        |
|                                      | 271ab207a197465f9d166c2dc7304b18             |
            | "169.254.255.254"}                      |
+------------------------+--------------------------+------------+--------------------------+
```

Checking the Flavor List

Log on to the server on which OSP-D is running to check the flavor list and to by entering the following command:

nova flavor-list

Example command output:

```
+--------------------------------------+------------------------+-----------+------+-----------+------+-------+-------------+-----------+
| ID                                   | Name                   | Memory_MB | Disk | Ephemeral
 | Swap | VCPUs | RXTX_Factor | Is_Public |
+--------------------------------------+------------------------+-----------+------+-----------+------+-------+-------------+-----------+
| eff0335b-3374-46c3-a3de-9f4b1ccaae04 | DNUCS002-AUTOIT-FLAVOR | 8192      | 80   | 0
     |      | 2     | 1.0         | True      |
+--------------------------------------+------------------------+-----------+------+-----------+------+-------+-------------+-----------+
```

Checking Host Aggregate and Availability Zone Configuration

Log on to the server on which OSP-D is running to check the host aggregate and availability zone configurations for the OSD Compute and for the AutoDeploy and AutoIT-VNF VMs.

Note It is assumed that the AutoDeploy and AutoIT-VNF VMs reside on the same OSD Compute node.

This is done by executing the following commands:

cd /home/stack
source~/<*stack_name*>**rc-core**
nova aggregate-list
Example command output:

```
+-----+-------------------+-------------------+
| Id  | Name              | Availability Zone |
+-----+-------------------+-------------------+
| 108 | LBUCS001-AUTOIT   | mgmt              |
| 147 | LBPGW101-EM-MGMT1 | -                 |
| 150 | LBPGW101-SERVICE1 | -                 |
| 153 | LBPGW101-CF-MGMT1 | -                 |
+-----+-------------------+-------------------+
```

nova aggregate-show LBUCS001-AUTOIT

```
+-----+-----------------+-------------------+--------------------------------------+--------------------------------+
| Id  | Name            | Availability Zone | Hosts                                | Metadata
                                |
+-----+-----------------+-------------------+--------------------------------------+--------------------------------+
| 108 | LBUCS001-AUTOIT | mgmt              | 'newtonoc-osd-compute-0.localdomain' |
'availability_zone=mgmt', 'mgmt=true' |
+-----+-----------------+-------------------+--------------------------------------+--------------------------------+
```

Note This information can also be verified through the Horizon GUI. Login to Horizon as the user core and navigate to **Project** > **Compute** > **Instances**. Check each instance to verify that the status is Active and the power state is Running.

Correct any instance that does not meet these criteria before continuing.

Checking Controller Server Health

Note The commands in this section should be executed on any one of the Controller nodes and do not need to be repeated on the other Controller nodes unless an issue is observed.

Checking the Pacemaker Cluster Stack (PCS) Status

Log on to one of the Controller nodes and verify that the group of resources in the PCS cluster are active and in the expected state by executing the following command:

sudo pcs status
Example command output:

```
Cluster name: tripleo_cluster
Stack: corosync
Current DC: tb3-ultram-controller-0 (version 1.1.15-11.el7_3.4-e174ec8) - partition with
quorum
Last updated: Wed Jul 12 13:28:56 2017          Last change: Tue Jul 11 21:45:09 2017 by
root via crm_attribute on tb3-ultram-controller-0

3 nodes and 22 resources configured

Online: [ tb3-ultram-controller-0 tb3-ultram-controller-1 tb3-ultram-controller-2 ]

Full list of resources:

 ip-192.200.0.104       (ocf::heartbeat:IPaddr2):       Started tb3-ultram-controller-1
 ip-10.84.123.6 (ocf::heartbeat:IPaddr2):       Started tb3-ultram-controller-0
 ip-11.119.0.42 (ocf::heartbeat:IPaddr2):       Started tb3-ultram-controller-0
 Clone Set: haproxy-clone [haproxy]
     Started: [ tb3-ultram-controller-0 tb3-ultram-controller-1 tb3-ultram-controller-2 ]
 Master/Slave Set: galera-master [galera]
     Masters: [ tb3-ultram-controller-0 tb3-ultram-controller-1 tb3-ultram-controller-2 ]
 ip-11.120.0.47 (ocf::heartbeat:IPaddr2):       Started tb3-ultram-controller-1
 ip-11.118.0.49 (ocf::heartbeat:IPaddr2):       Started tb3-ultram-controller-0
 Clone Set: rabbitmq-clone [rabbitmq]
     Started: [ tb3-ultram-controller-0 tb3-ultram-controller-1 tb3-ultram-controller-2 ]
 ip-11.120.0.48 (ocf::heartbeat:IPaddr2):       Started tb3-ultram-controller-1
 Master/Slave Set: redis-master [redis]
     Masters: [ tb3-ultram-controller-0 ]
     Slaves: [ tb3-ultram-controller-1 tb3-ultram-controller-2 ]
 openstack-cinder-volume        (systemd:openstack-cinder-volume):      Started
tb3-ultram-controller-0
 my-ipmilan-for-controller-0    (stonith:fence_ipmilan):        Started
tb3-ultram-controller-0
 my-ipmilan-for-controller-1    (stonith:fence_ipmilan):        Started
tb3-ultram-controller-1
 my-ipmilan-for-controller-2    (stonith:fence_ipmilan):        Started
tb3-ultram-controller-0

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

From the output of this command, ensure that:

- All 3 controllers are listed as Online
- haproxy-clone is started on all 3 controllers
- galera-master lists all 3 controllers as Masters

- rabbitmq-clone is started on all 3 controllers
- redis-master lists one controller as master and the other 2 controllers as slaves
- openstack-cinder-volume is started on one node
- my-ipmilan/stonith is started on all 3 controllers
- Daemons corosync, pacemaker and pcsd are active and enabled

Note If the output displays any "Failed Actions", execute the **sudo pcs resource cleanup** command and then re-execute the **sudo pcs status** command.

Checking Ceph Storage Status

Log on to the Controller node and verify the health of the Ceph storage from the Controller node by executing the following command:

sudo ceph status
Example command output:

```
cluster eb2bb192-b1c9-11e6-9205-525400330666
     health HEALTH_OK
     monmap e1: 3 mons at
{tb3-ultram-controller-0=11.118.0.10:6789/0,tb3-ultram-controller-1=11.118.0.11:6789/0,
tb3-ultram-controller-2=11.118.0.12:6789/0}
            election epoch 152, quorum 0,1,2
tb3-ultram-controller-0,tb3-ultram-controller-1,tb3-ultram-controller-2
     osdmap e158: 12 osds: 12 up, 12 in
            flags sortbitwise,require_jewel_osds
      pgmap v1417251: 704 pgs, 6 pools, 321 GB data, 110 kobjects
            961 GB used, 12431 GB / 13393 GB avail
                 704 active+clean
  client io 53755 B/s wr, 0 op/s rd, 7 op/s wr
```

From the output of this command, ensure that:

- health is listed as HEALTH_OK
- The correct number of monitors are listed in the monmap
- The correct number of OSDs are listed in the osdmap

Checking Controller Node Services

Log on to the Controller node and check the status of all services by executing the following command:

sudo systemctl list-units "openstack*" "neutron*" "openvswitch*"
Example command output:

```
UNIT                                        LOAD   ACTIVE SUB     DESCRIPTION
neutron-dhcp-agent.service                  loaded active running OpenStack Neutron DHCP
Agent
neutron-l3-agent.service                    loaded active running OpenStack Neutron Layer
 3 Agent
neutron-metadata-agent.service             loaded active running OpenStack Neutron Metadata
 Agent
neutron-openvswitch-agent.service           loaded active running OpenStack Neutron Open
vSwitch Agent
neutron-ovs-cleanup.service                 loaded active exited  OpenStack Neutron Open
vSwitch Cleanup Utility
```

```
neutron-server.service                          loaded active running OpenStack Neutron Server
openstack-aodh-evaluator.service               loaded active running OpenStack Alarm evaluator
 service
openstack-aodh-listener.service                loaded active running OpenStack Alarm listener
 service
openstack-aodh-notifier.service                loaded active running OpenStack Alarm notifier
 service
openstack-ceilometer-central.service            loaded active running OpenStack ceilometer
central agent
openstack-ceilometer-collector.service          loaded active running OpenStack ceilometer
collection service
openstack-ceilometer-notification.service       loaded active running OpenStack ceilometer
notification agent
openstack-cinder-api.service                    loaded active running OpenStack Cinder API
Server
openstack-cinder-scheduler.service             loaded active running OpenStack Cinder Scheduler
 Server
openstack-cinder-volume.service                 loaded active running Cluster Controlled
openstack-cinder-volume
openstack-glance-api.service                    loaded active running OpenStack Image Service
 (code-named Glance) API server
openstack-glance-registry.service               loaded active running OpenStack Image Service
 (code-named Glance) Registry server
openstack-gnocchi-metricd.service              loaded active running OpenStack gnocchi metricd
 service
openstack-gnocchi-statsd.service                loaded active running OpenStack gnocchi statsd
 service
openstack-heat-api-cfn.service                  loaded active running Openstack Heat
CFN-compatible API Service
openstack-heat-api-cloudwatch.service          loaded active running OpenStack Heat CloudWatch
 API Service
openstack-heat-api.service                     loaded active running OpenStack Heat API Service
openstack-heat-engine.service                   loaded active running Openstack Heat Engine
Service
openstack-nova-api.service                     loaded active running OpenStack Nova API Server
openstack-nova-conductor.service                loaded active running OpenStack Nova Conductor
 Server
openstack-nova-consoleauth.service             loaded active running OpenStack Nova VNC console
 auth Server
openstack-nova-novncproxy.service               loaded active running OpenStack Nova NoVNC
Proxy Server
openstack-nova-scheduler.service                loaded active running OpenStack Nova Scheduler
 Server
openstack-swift-account-auditor.service         loaded active running OpenStack Object Storage
 (swift) - Account Auditor
openstack-swift-account-reaper.service          loaded active running OpenStack Object Storage
 (swift) - Account Reaper
openstack-swift-account-replicator.service      loaded active running OpenStack Object Storage
 (swift) - Account Replicator
openstack-swift-account.service                 loaded active running OpenStack Object Storage
 (swift) - Account Server
openstack-swift-container-auditor.service       loaded active running OpenStack Object Storage
 (swift) - Container Auditor
openstack-swift-container-replicator.service loaded active running OpenStack Object Storage
 (swift) - Container Replicator
openstack-swift-container-updater.service       loaded active running OpenStack Object Storage
 (swift) - Container Updater
openstack-swift-container.service               loaded active running OpenStack Object Storage
 (swift) - Container Server
openstack-swift-object-auditor.service          loaded active running OpenStack Object Storage
 (swift) - Object Auditor
openstack-swift-object-expirer.service          loaded active running OpenStack Object Storage
 (swift) - Object Expirer
openstack-swift-object-replicator.service       loaded active running OpenStack Object Storage
 (swift) - Object Replicator
openstack-swift-object-updater.service          loaded active running OpenStack Object Storage
 (swift) - Object Updater
openstack-swift-object.service                  loaded active running OpenStack Object Storage
 (swift) - Object Server
openstack-swift-proxy.service                   loaded active running OpenStack Object Storage
 (swift) - Proxy Server
openvswitch.service                              loaded active exited  Open vSwitch
```

```
LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

43 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
```

Check the RabbitMQ Database Status

From each of the controller nodes, determine if the rabbitmq database is in a good state by executing the following command:

sudo rabbitmqctl eval 'rabbit_diagnostics:maybe_stuck().'
Example command output:

```
2017-07-20 01:58:02 There are 11020 processes.
2017-07-20 01:58:02 Investigated 0 processes this round, 5000ms to go.
2017-07-20 01:58:03 Investigated 0 processes this round, 4500ms to go.
2017-07-20 01:58:03 Investigated 0 processes this round, 4000ms to go.
2017-07-20 01:58:04 Investigated 0 processes this round, 3500ms to go.
2017-07-20 01:58:04 Investigated 0 processes this round, 3000ms to go.
2017-07-20 01:58:05 Investigated 0 processes this round, 2500ms to go.
2017-07-20 01:58:05 Investigated 0 processes this round, 2000ms to go.
2017-07-20 01:58:06 Investigated 0 processes this round, 1500ms to go.
2017-07-20 01:58:06 Investigated 0 processes this round, 1000ms to go.
2017-07-20 01:58:07 Investigated 0 processes this round, 500ms to go.
2017-07-20 01:58:07 Found 0 suspicious processes.
ok
```

If the database is healthy, the command returns “Found 0 suspicious processes.” If the database is not healthy, the command returns 1 or more suspicious processes. Contact your local support representative if suspicious processes are found.

Checking OSD Compute Server Health

Checking Ceph Status

Log on to the OSD Compute and check the Ceph storage status by executing the following command:

sudo ceph status
Example command output:

```
sudo ceph status
    cluster eb2bb192-b1c9-11e6-9205-525400330666
     health HEALTH_OK
     monmap e1: 3 mons at
{tb3-ultram-controller-0=11.118.0.10:6789/0,tb3-ultram-controller-1=11.118.0.11:6789/0,
tb3-ultram-controller-2=11.118.0.12:6789/0}
            election epoch 152, quorum 0,1,2
tb3-ultram-controller-0,tb3-ultram-controller-1,tb3-ultram-controller-2
     osdmap e158: 12 osds: 12 up, 12 in
            flags sortbitwise,require_jewel_osds
      pgmap v1417867: 704 pgs, 6 pools, 321 GB data, 110 kobjects
            961 GB used, 12431 GB / 13393 GB avail
                 704 active+clean
  client io 170 kB/s wr, 0 op/s rd, 24 op/s wr
```

Checking OSD Compute Node Services

Log on to each OSD Compute node and check the status of all services by executing the following command:

sudo systemctl list-units "openstack*" "neutron*" "openvswitch*"

Example command output:

```
UNIT                                LOAD   ACTIVE SUB     DESCRIPTION
neutron-openvswitch-agent.service    loaded active running OpenStack Neutron Open vSwitch
Agent
neutron-ovs-cleanup.service          loaded active exited  OpenStack Neutron Open vSwitch
Cleanup Utility
neutron-sriov-nic-agent.service      loaded active running OpenStack Neutron SR-IOV NIC
Agent
openstack-ceilometer-compute.service loaded active running OpenStack ceilometer compute
agent
openstack-nova-compute.service       loaded active running OpenStack Nova Compute Server
openvswitch.service                  loaded active exited  Open vSwitch

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

6 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
```

Monitoring AutoDeploy Operations

This section identifies various commands that can be used to determine the status and health of AutoDeploy.

To use them, you must:

1. Log on to the AutoDeploy VM as *ubuntu*. Use the password that was created earlier for this user.
2. Become the root user.

   sudo -i

Viewing AutoDeploy Logs

AutoDeploy logs are available on the AutoDeploy VM in the following directory:

/var/log/upstart/autodeploy.log

Note To access the command used to view logs, you must be logged in to the Confd CLI as the admin user on the AutoDeploy VM:

confd_cli -u admin -C

AutoDeploy Transaction Logs

Execute the following command to display AutoDeploy transaction logs:

show logs $TX-ID | display xml
Example output - Activation:

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <log xmlns="http://www.cisco.com/usp/nfv/usp-autodeploy-oper">
    <tx-id>1495749896040</tx-id>
    <log>Thu May 25 22:04:57 UTC 2017 [Task: 1495749896040] Started service deployment
ServiceDeploymentRequest [type=ACTIVATE, ser-viceDeploymentId=north-east, siteList=[]]
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Uploading config file(s)
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Uploading image file(s)
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Validation of ISO called for
OS linux
```

```
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Executing /tmp mount -t iso9660
 -o loop /home/ubuntu/isos/usp-5_1_0.iso /tmp/7715990769784465243
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Command exited with return code:
 0
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Executing . ls -lah
/tmp/7715990769784465243/repo
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Command exited with return code:
 0
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Executing . python
/opt/cisco/signing/cisco_openpgp_verify_release.py -e
/tmp/7715990769784465243/repo/USP_RPM_CODE_REL_KEY-CCO_RELEASE.cer -G
/tmp/7715990769784465243/repo/rel.gpg
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Command exited with return code:
 0
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] ISO validation successful
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Executing . umount
/tmp/7715990769784465243
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Command exited with return code:
 0
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Executing . rm -r
/tmp/7715990769784465243
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Command exited with return code:
 0
Thu May 25 22:04:58 UTC 2017 [Task: 1495749896040/vnf-pkg1] Uploading ISO file
Thu May 25 22:06:32 UTC 2017 [Task: 1495749896040/vnf-pkg1] Collecting VnfPkg vnf-pkg1
details
Thu May 25 22:06:32 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf1-rack-auto-test-sjc-service1] Create Host Aggregate:
auto-test-sjc-service1
Thu May 25 22:06:33 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf1-rack-auto-test-sjc-service1] Created Host Aggregate
successfully.
Thu May 25 22:06:33 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf1-rack-auto-test-sjc-cf-esc-mgmt1] Create Host Aggregate:
auto-test-sjc-cf-esc-mgmt1
Thu May 25 22:06:34 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf1-rack-auto-test-sjc-cf-esc-mgmt1] Created Host Aggregate
success-fully.
Thu May 25 22:06:34 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf1-rack-auto-test-sjc-em-autovnf-mgmt1] Create Host Aggregate:
 auto-test-sjc-em-autovnf-mgmt1
Thu May 25 22:06:35 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf1-rack-auto-test-sjc-em-autovnf-mgmt1] Created Host Aggregate
 suc-cessfully.
Thu May 25 22:06:35 UTC 2017 [Task: 1495749896040/auto-testautovnf1] Current status of
AutoVnf auto-testautovnf1 is unknown hence send-ing request to deploy it.
Thu May 25 22:08:59 UTC 2017 [Task: 1495749896040/auto-testautovnf1] Successfully deployed
 AutoVnf auto-testautovnf1 with floating-ip 172.21.201.59.
Thu May 25 22:08:59 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm1] Starting VNFM deployment
Thu May 25 22:08:59 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm1] Current Vnfm deployment
 status is unknown
Thu May 25 22:08:59 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm1] Deploying VNFM
Thu May 25 22:13:10 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm1] VNFM deployed
successfully
Thu May 25 22:13:20 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm1] Got Vnfm HA-VIP =
172.57.11.6
Thu May 25 22:13:35 UTC 2017 [Task: 1495749896040/auto-testvnfd1] Starting Vnf Deployment
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/auto-testvnfd1] Successfully completed
all Vnf Deployments.
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Uploading config file(s)
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Uploading image file(s)
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Validation of ISO called for
OS linux
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Executing /tmp mount -t iso9660
 -o loop /home/ubuntu/isos/usp-5_1_0.iso /tmp/5099470753324893053
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Command exited with return code:
 0
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Executing . ls -lah
/tmp/5099470753324893053/repo
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Command exited with return code:
 0
Thu May 25 22:19:05 UTC 2017 [Task: 1495749896040/vnf-pkg2] Executing . python
/opt/cisco/signing/cisco_openpgp_verify_release.py -e
```

```
/tmp/5099470753324893053/repo/USP_RPM_CODE_REL_KEY-CCO_RELEASE.cer -G
/tmp/5099470753324893053/repo/rel.gpg
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] Command exited with return code:
 0
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] ISO validation successful
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] Executing . umount
/tmp/5099470753324893053
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] Command exited with return code:
 0
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] Executing . rm -r
/tmp/5099470753324893053
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] Command exited with return code:
 0
Thu May 25 22:19:06 UTC 2017 [Task: 1495749896040/vnf-pkg2] Uploading ISO file
Thu May 25 22:20:23 UTC 2017 [Task: 1495749896040/vnf-pkg2] Collecting VnfPkg vnf-pkg2
details
Thu May 25 22:20:23 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf2-rack-auto-test-sjc-em-autovnf-mgmt2] Create Host Aggregate:
 auto-test-sjc-em-autovnf-mgmt2
Thu May 25 22:20:25 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf2-rack-auto-test-sjc-em-autovnf-mgmt2] Created Host Aggregate
 suc-cessfully.
Thu May 25 22:20:25 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf2-rack-auto-test-sjc-service2] Create Host Aggregate:
auto-test-sjc-service2
Thu May 25 22:20:26 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf2-rack-auto-test-sjc-service2] Created Host Aggregate
successfully.
Thu May 25 22:20:26 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf2-rack-auto-test-sjc-cf-esc-mgmt2] Create Host Aggregate:
auto-test-sjc-cf-esc-mgmt2
Thu May 25 22:20:27 UTC 2017 [Task:
1495749896040/auto-test-sjc-vnf2-rack-auto-test-sjc-cf-esc-mgmt2] Created Host Aggregate
success-fully.
Thu May 25 22:20:27 UTC 2017 [Task: 1495749896040/auto-testautovnf2] Current status of
AutoVnf auto-testautovnf2 is unknown hence send-ing request to deploy it.
Thu May 25 22:22:44 UTC 2017 [Task: 1495749896040/auto-testautovnf2] Successfully deployed
 AutoVnf auto-testautovnf2 with floating-ip 172.21.201.64.
Thu May 25 22:22:44 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm2] Starting VNFM deployment
Thu May 25 22:22:44 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm2] Current Vnfm deployment
 status is unknown
Thu May 25 22:22:44 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm2] Deploying VNFM
Thu May 25 22:27:04 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm2] VNFM deployed
successfully
Thu May 25 22:27:14 UTC 2017 [Task: 1495749896040/ab-auto-test-vnfm2] Got Vnfm HA-VIP =
172.67.11.5
Thu May 25 22:27:29 UTC 2017 [Task: 1495749896040/auto-testvnfd2] Starting Vnf Deployment
Thu May 25 22:32:40 UTC 2017 [Task: 1495749896040/auto-testvnfd2] Successfully completed
all Vnf Deployments.
Thu May 25 22:32:40 UTC 2017 [Task: 1495749896040] Success
</log>
  </log>
</config>
```

Example output - Deactivation:

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <log xmlns="http://www.cisco.com/usp/nfv/usp-autodeploy-oper">
    <tx-id>1495752667278</tx-id>
    <log>Thu May 25 22:51:08 UTC 2017 [Task: 1495752667278] Started service deployment
ServiceDeploymentRequest [type=DEACTIVATE, serviceDeploymentId=north-east, siteList=[]]
Thu May 25 22:51:08 UTC 2017 [Task: 1495752667278/auto-testvnfd2] Starting Vnf UnDeployment
Thu May 25 22:52:58 UTC 2017 [Task: 1495752667278/auto-testvnfd2] Successfully deactivated
 all Vnf Deployments.
Thu May 25 22:53:00 UTC 2017 [Task: 1495752667278/auto-testvnfd2] Vnf UnDeployment Successful
Thu May 25 22:53:00 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Deactivating VNFM
Thu May 25 22:53:31 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Successfully
deactivating VNFM
Thu May 25 22:53:31 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Deleted VnfmInstance
 configuration
Thu May 25 22:53:31 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm2] Deleted Vnfm
configuration
Thu May 25 22:54:21 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-em-autovnf-mgmt2] Starting to delete
```

```
Host Aggregate.
Thu May 25 22:54:22 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-em-autovnf-mgmt2] Deleted Host Aggregate
 successfully.
Thu May 25 22:54:22 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-service2] Starting to delete Host
Aggregate.
Thu May 25 22:54:23 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-service2] Deleted Host Aggregate
successfully.
Thu May 25 22:54:23 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-cf-esc-mgmt2] Starting to delete Host
Aggregate.
Thu May 25 22:54:24 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf2-rack-auto-test-sjc-cf-esc-mgmt2] Deleted Host Aggregate
successfully.
Thu May 25 22:54:24 UTC 2017 [Task: 1495752667278/auto-testvnfd1] Starting Vnf UnDeployment
Thu May 25 22:56:24 UTC 2017 [Task: 1495752667278/auto-testvnfd1] Successfully deactivated
 all Vnf Deployments.
Thu May 25 22:56:26 UTC 2017 [Task: 1495752667278/auto-testvnfd1] Vnf UnDeployment Successful
Thu May 25 22:56:26 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Deactivating VNFM
Thu May 25 22:56:56 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Successfully
deactivating VNFM
Thu May 25 22:56:56 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Deleted VnfmInstance
 configuration
Thu May 25 22:56:56 UTC 2017 [Task: 1495752667278/ab-auto-test-vnfm1] Deleted Vnfm
configuration
Thu May 25 22:57:54 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-service1] Starting to delete Host
Aggregate.
Thu May 25 22:57:55 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-service1] Deleted Host Aggregate
successfully.
Thu May 25 22:57:55 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-cf-esc-mgmt1] Starting to delete Host
Aggregate.
Thu May 25 22:57:56 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-cf-esc-mgmt1] Deleted Host Aggregate
successfully.
Thu May 25 22:57:56 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-em-autovnf-mgmt1] Starting to delete
Host Aggregate.
Thu May 25 22:57:57 UTC 2017 [Task:
1495752667278/auto-test-sjc-vnf1-rack-auto-test-sjc-em-autovnf-mgmt1] Deleted Host Aggregate
 successfully.
Thu May 25 22:57:58 UTC 2017 [Task: 1495752667278] Success
</log>
  </log>
</config>
```

Viewing AutoDeploy Operational Data

View the AutoDeploy operational data by executing the following command:

show service-deploymentr
Example output (VIM Orchestrator deployment)

```
service-deploymentr north-east
siter auto-test-sjc
  nfvi-popr nfvi-deployment-status "Required Undercloud services are UP"
  nfvi-popr vim-orch status deployment-success
  nfvi-popr vim-orch steps-total 84
  nfvi-popr vim-orch steps-completed 84
  nfvi-popr vim-orch version "Red Hat OpenStack Platform release 10.0 (Newton)"
                                                    FIRMWARE  IP       BIOS
IS           PHYSNET
NFVI NODE ID               UUID  STATUS  ROLE      VERSION   ADDRESS  VERSION  ID  SIZE
JOURNAL  ID  ID
-------------------------------------------------------------------------------------------------------------
```

```
autoit-nfvi-physical-node  -     up      vim-orch  -         -         -
```

Note The deployment-status in the above output changes based on the current progress. The command can be re-issued multiple times to refresh the status.

Example output (VIM deployment)

```
PACKAGER  AUTO IT
ID        ISO ID      STATUS
------------------------------
vnf-pkg1  5.5.1-1315  alive
vnf-pkg2  5.5.1-1315  alive

  nfvi-popr nfvi-deployment-status "Stack vnf1-vim create completed"
  nfvi-popr vim-orch status deployment-success
  nfvi-popr vim-orch steps-total 84
  nfvi-popr vim-orch steps-completed 84
  nfvi-popr vim-orch version "Red Hat OpenStack Platform release 10.0 (Newton)"
  nfvi-popr vim status deployment-success
  nfvi-popr vim steps-total 16
  nfvi-popr vim steps-completed 16
  nfvi-popr vim version "Red Hat OpenStack Platform release 10.0 (Newton)"
                                                          FIRMWARE
                                        IS
NFVI NODE ID              UUID  STATUS  ROLE           VERSION   IP ADDRESS    BIOS VERSION
                  ID        SIZE     JOURNAL  ID          PHYSNET ID
-------------------------------------------------------------------------------------------------------------
autoit-nfvi-physical-node  -     up      vim-orch        -         -             -

node_1                     -     up      vim-controller  2.0(13i)  192.100.3.5
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false
node_2                     -     up      vim-controller  2.0(13i)  192.100.3.6
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false
node_3                     -     up      vim-controller  2.0(13i)  192.100.3.7
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false
node_4                     -     up      vim-compute     2.0(13i)  192.100.3.8
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                         enp10s0f1   phys_pcie1_1

                                                         enp133s0f0  phys_pcie4_0

                                                         enp133s0f1  phys_pcie4_1
node_5                     -     up      vim-compute     2.0(13i)  192.100.3.9
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                         enp10s0f1   phys_pcie1_1

                                                         enp133s0f0  phys_pcie4_0

                                                         enp133s0f1  phys_pcie4_1
node_6                     -     up      vim-compute     2.0(13i)  192.100.3.10
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                         enp10s0f1   phys_pcie1_1

                                                         enp133s0f0  phys_pcie4_0

                                                         enp133s0f1  phys_pcie4_1
node_7                     -     up      vim-compute     2.0(13i)  192.100.3.11
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                         enp10s0f1   phys_pcie1_1

                                                         enp133s0f0  phys_pcie4_0

                                                         enp133s0f1  phys_pcie4_1
node_8                     -     up      vim-compute     2.0(13e)  192.100.3.12
C240M4.2.0.13d.0.0812161132  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0
```

```
                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_9                        -     up      vim-compute     2.0(13e)  192.100.3.13
C240M4.2.0.13d.0.0812161132  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_10                       -     up      vim-compute     2.0(13e)  192.100.3.14
C240M4.2.0.13d.0.0812161132  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_11                       -     up      vim-compute     2.0(13e)  192.100.3.15
C240M4.2.0.13d.0.0812161132  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_12                       -     up      vim-compute     2.0(13i)  192.100.3.16
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_13                       -     up      vim-compute     2.0(13i)  192.100.3.17
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_14                       -     up      vim-compute     2.0(13i)  192.100.3.18
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_15                       -     up      vim-compute     2.0(13i)  192.100.3.19
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_16                       -     up      vim-compute     2.0(13i)  192.100.3.20
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0

                                                enp10s0f1   phys_pcie1_1

                                                enp133s0f0  phys_pcie4_0

                                                enp133s0f1  phys_pcie4_1
node_17                       -     up      vim-compute     2.0(13i)  192.100.3.21
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0
```

```
                                                enp10s0f1   phys_pcie1_1
                                                enp133s0f0  phys_pcie4_0
                                                enp133s0f1  phys_pcie4_1
node_18                     -     up      vim-compute     2.0(13i)  192.100.3.22
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0
                                                enp10s0f1   phys_pcie1_1
                                                enp133s0f0  phys_pcie4_0
                                                enp133s0f1  phys_pcie4_1
node_19                     -     up      vim-compute     2.0(13i)  192.100.3.23
C240M4.2.0.13g.0.1113162311  /dev/sda  1143845  false    enp10s0f0   phys_pcie1_0
                                                enp10s0f1   phys_pcie1_1
                                                enp133s0f0  phys_pcie4_0
                                                enp133s0f1  phys_pcie4_1
node_20                     -     up      vim-osd-compute 2.0(13i)  192.100.3.24
C240M4.2.0.13g.0.1113162311  /dev/sda  285245   false
                    /dev/sdb  456965   true
                    /dev/sdc  1143845  false
                    /dev/sdd  1143845  false
                    /dev/sde  1143845  false
                    /dev/sdf  1143845  false    enp10s0f0   phys_pcie1_0
                                                enp10s0f1   phys_pcie1_1
                                                enp133s0f0  phys_pcie4_0
                                                enp133s0f1  phys_pcie4_1
node_21                     -     up      vim-osd-compute 2.0(13i)  192.100.3.25
C240M4.2.0.13g.0.1113162311  /dev/sda  285245   false
                    /dev/sdb  456965   true
                    /dev/sdc  1143845  false
                    /dev/sdd  1143845  false
                    /dev/sde  1143845  false
                    /dev/sdf  1143845  false    enp10s0f0   phys_pcie1_0
                                                enp10s0f1   phys_pcie1_1
                                                enp133s0f0  phys_pcie4_0
                                                enp133s0f1  phys_pcie4_1
node_22                     -     up      vim-osd-compute 2.0(13i)  192.100.3.26
C240M4.2.0.13g.0.1113162311  /dev/sda  285245   false
                    /dev/sdb  456965   true
                    /dev/sdc  1143845  false
                    /dev/sdd  1143845  false
                    /dev/sde  1143845  false
                    /dev/sdf  1143845  false    enp10s0f0   phys_pcie1_0
                                                enp10s0f1   phys_pcie1_1
                                                enp133s0f0  phys_pcie4_0
```

```
                                                    enp133s0f1  phys_pcie4_1

  autovnfr auto-testautovnf1
   iso-id 5.5.1-1315
   endpoint-info ip-address 172.25.22.71
   endpoint-info port 2022
   status alive
   vnfmr ab-auto-test-vnfm1
    endpoint-info ip-address 172.57.11.102
    endpoint-info port 830
    status alive
   vnfr auto-testvnfd1
    iso-id 5.5.1-1315
    status alive
    vnf-deploymentr vnfd1-deployment
     em-endpoint-info ip-address 172.57.11.103
     em-endpoint-info port 2022
  autovnfr auto-testautovnf2
   iso-id 5.5.1-1315
   endpoint-info ip-address 172.25.22.77
   endpoint-info port 2022
   status alive
   vnfmr ab-auto-test-vnfm2
    endpoint-info ip-address 172.67.11.7
    endpoint-info port 830
    status alive
   vnfr auto-testvnfd2
    iso-id 5.5.1-1315
    status alive
    vnf-deploymentr vnfd2-deployment
     em-endpoint-info ip-address 172.67.11.11
     em-endpoint-info port 2022
```

Note The deployment-status in the above output changes based on the current progress. The command can be re-issued multiple times to refresh the status.

Example output (VNF deployment):

```
VNF
PACKAGER  AUTO IT
ID        ISO ID     STATUS
----------------------------
vnf-pkg1  5.1.0-662  alive
vnf-pkg2  5.1.0-662  alive

  autovnfr auto-testautovnf1
   endpoint-info ip-address 172.21.201.59
   endpoint-info port 2022
   status alive
   vnfmr ab-auto-test-vnfm1
    endpoint-info ip-address 172.57.11.6
    endpoint-info port 830
    status alive
   vnfr auto-testvnfd1
    status alive
    vnf-deploymentr vnfd1-deployment
     em-endpoint-info ip-address 172.57.11.12
     em-endpoint-info port 2022
  autovnfr auto-testautovnf2
   endpoint-info ip-address 172.21.201.64
   endpoint-info port 2022
   status alive
   vnfmr ab-auto-test-vnfm2
    endpoint-info ip-address 172.67.11.5
    endpoint-info port 830
    status alive
   vnfr auto-testvnfd2
    status alive
    vnf-deploymentr vnfd2-deployment
```

```
em-endpoint-info ip-address 172.67.11.12
em-endpoint-info port 2022
```

Checking AutoDeploy Processes

Verify that key processes are running on the AutoDeploy VM:

initctl status autodeploy
Example output:

```
autodeploy start/running, process 1771
```

ps -ef | grep java

Example output:

```
root      1788  1771  0 May24 ?        00:00:41 /usr/bin/java -jar
/opt/cisco/usp/apps/autodeploy/autodeploy-1.0.jar com.cisco.usp.autodeploy.Application
--autodeploy.transaction-log-store=/var/log/cisco-uas/autodeploy/transactions
```

Stopping/Restarting AutoDeploy Processes

To start the AutoDeploy process:

initctl start autodeploy
Example output:

```
AutoIT-VNF API server stopped.
```

To restart the AutoIT-VNF processes:

initctl stop autodeploy
Example output:

```
autodeploy stop/waiting
```

To restart the AutoDeploy process:

initctl restart autodeploy
Example output:

```
autodeploy start/running, process 11049
```

Determining the Running AutoDeploy Version

To display the version of the AutoDeploy software module that is currently operational:

ps -ef | grep java

Example output:

```
root      1788  1771  0 May24 ?        00:00:41 /usr/bin/java -jar
/opt/cisco/usp/apps/autodeploy/autodeploy-1.0.jar com.cisco.usp.autodeploy.Application
--autodeploy.transaction-log-store=/var/log/cisco-uas/autodeploy/transactions
```

Monitoring AutoIT-VNF Operations

This section identifies various commands that can be used to determine the status and health of AutoIT-VNF.

To use them, you must:

1 Log on to the AutoIT-VNF VM as *ubuntu*. Use the password that was created earlier for this user.

2 Become the *root* user.

sudo -i

Viewing AutoIT-VNF Logs

AutoIT maintains logs containing information pertaining to UAS deployment and termination transactions. The *autoit.log* file is located in the following directory on the Ultra M Manager Node:

/var/log/cisco/usp/auto-it/autoit.log

Example Deployment Log:

tail -100f /var/log/cisco/usp/auto-it/autoit.log &^C

```
2017-05-25 22:04:57,527 - INFO: Received a request to list config folder names.
2017-05-25 22:04:57,527 - INFO: config contents are:
2017-05-25 22:04:57,536 - INFO: Received a request to list config folder names.
2017-05-25 22:04:57,536 - INFO: config contents are:
2017-05-25 22:04:57,545 - INFO: Received a request to create a configuration folder.
2017-05-25 22:04:57,551 - INFO: Received a request to create a configuration folder.
2017-05-25 22:04:57,553 - INFO: Received request to download package: system.cfg from ISO
2017-05-25 22:04:57,563 - INFO: Received request to download package: system.cfg from ISO
2017-05-25 22:04:57,565 - INFO: Received request to download package: system.cfg from ISO
2017-05-25 22:04:57,566 - INFO: Received request to upload config file system.cfg to config
 named vnf-pkg1
2017-05-25 22:04:57,567 - INFO: Uploaded file system.cfg to config named vnf-pkg1

2017-05-25 22:05:54,268 - INFO: Received request to upload ISO usp-5_1_0.iso
2017-05-25 22:05:54,268 - INFO: Saving ISO to /tmp/tmpxu7MuO/usp-5_1_0.iso
2017-05-25 22:06:30,678 - INFO: Mounting ISO to /tmp/tmpxu7MuO/iso_mount
2017-05-25 22:06:30,736 - INFO: ISO version already installed, (5.1.0-662)
2017-05-25 22:06:31,355 - INFO: Received a request to list file names in config named
vnf-pkg1.
2017-05-25 22:06:31,355 - INFO: config contents are: system.cfg
2017-05-25 22:06:31,362 - INFO: Received a request to list file names in config named
vnf-pkg1-images.
2017-05-25 22:06:31,362 - INFO: config contents are:
2017-05-25 22:06:31,370 - INFO: Received request to get ISO details 5.1.0-662
2017-05-25 22:06:31,391 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:06:31,857 - INFO: Getting Host Aggregate failed: Aggregate
'auto-test-sjc-service1' not found on OpenStack setup
2017-05-25 22:06:31,872 - INFO: Received a request to deploy an Host Aggregate
2017-05-25 22:06:32,415 - INFO: Deploying Host Aggregate 'auto-test-sjc-service1' completed
2017-05-25 22:06:32,427 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:06:32,975 - INFO: Getting Host Aggregate failed: Aggregate
'auto-test-sjc-cf-esc-mgmt1' not found on OpenStack setup
2017-05-25 22:06:32,986 - INFO: Received a request to deploy an Host Aggregate
2017-05-25 22:06:33,513 - INFO: Deploying Host Aggregate 'auto-test-sjc-cf-esc-mgmt1'
completed
2017-05-25 22:06:33,524 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:06:33,881 - INFO: Getting Host Aggregate failed: Aggregate
'auto-test-sjc-em-autovnf-mgmt1' not found on OpenStack setup
2017-05-25 22:06:33,891 - INFO: Received a request to deploy an Host Aggregate
2017-05-25 22:06:34,535 - INFO: Deploying Host Aggregate 'auto-test-sjc-em-autovnf-mgmt1'
completed
2017-05-25 22:06:34,580 - INFO: Received a request to deploy AutoVnf
2017-05-25 22:06:40,340 - INFO: Creating AutoVnf deployment (3 instance(s)) on
'http://172.21.201.217:5000/v2.0' tenant 'core' user 'core', ISO '5.1.0-662'
2017-05-25 22:06:40,340 - INFO: Creating network 'auto-testautovnf1-uas-management'
2017-05-25 22:06:42,241 - INFO: Created network 'auto-testautovnf1-uas-management'
2017-05-25 22:06:42,241 - INFO: Creating network 'auto-testautovnf1-uas-orchestration'
2017-05-25 22:06:42,821 - INFO: Created network 'auto-testautovnf1-uas-orchestration'
2017-05-25 22:06:42,888 - INFO: Created flavor 'auto-testautovnf1-uas'
2017-05-25 22:06:42,888 - INFO: Loading image 'auto-testautovnf1-usp-uas-1.0.0-601.qcow2'
from '/opt/cisco/usp/bundles/5.1.0-662/uas-bundle/usp-uas-1.0.0-601.qcow2'
2017-05-25 22:06:53,927 - INFO: Loaded image 'auto-testautovnf1-usp-uas-1.0.0-601.qcow2'
```

```
2017-05-25 22:06:53,928 - INFO: Creating volume 'auto-testautovnf1-uas-vol-0' with command

[/opt/cisco/usp/apps/auto-it/vnf/../common/autoit/../autoit_os_utils/scripts/autoit_volume_staging.sh
 OS_USERNAME core OS_TENANT_NAME core OS_PASSWORD **** OS_AUTH_URL
http://172.21.201.217:5000/v2.0 ARG_TENANT core ARG_DEPLOYMENT test-uas ARG_VM_NAME
auto-testautovnf1-uas-vol-0 ARG_VOLUME_TYPE LUKS  FILE_1 /tmp/tmphsTAj6/encrypted.cfg]
2017-05-25 22:07:06,104 - INFO: Created volume 'auto-testautovnf1-uas-vol-0'
2017-05-25 22:07:06,104 - INFO: Creating volume 'auto-testautovnf1-uas-vol-1' with command

[/opt/cisco/usp/apps/auto-it/vnf/../common/autoit/../autoit_os_utils/scripts/autoit_volume_staging.sh
 OS_USERNAME core OS_TENANT_NAME core OS_PASSWORD **** OS_AUTH_URL
http://172.21.201.217:5000/v2.0 ARG_TENANT core ARG_DEPLOYMENT test-uas ARG_VM_NAME
auto-testautovnf1-uas-vol-1 ARG_VOLUME_TYPE LUKS  FILE_1 /tmp/tmphsTAj6/encrypted.cfg]
2017-05-25 22:07:17,598 - INFO: Created volume 'auto-testautovnf1-uas-vol-1'
2017-05-25 22:07:17,598 - INFO: Creating volume 'auto-testautovnf1-uas-vol-2' with command

[/opt/cisco/usp/apps/auto-it/vnf/../common/autoit/../autoit_os_utils/scripts/autoit_volume_staging.sh
 OS_USERNAME core OS_TENANT_NAME core OS_PASSWORD **** OS_AUTH_URL
http://172.21.201.217:5000/v2.0 ARG_TENANT core ARG_DEPLOYMENT test-uas ARG_VM_NAME
auto-testautovnf1-uas-vol-2 ARG_VOLUME_TYPE LUKS  FILE_1 /tmp/tmphsTAj6/encrypted.cfg]
2017-05-25 22:07:29,242 - INFO: Created volume 'auto-testautovnf1-uas-vol-2'
2017-05-25 22:07:30,477 - INFO: Assigned floating IP '172.21.201.59' to IP '172.57.11.101'
2017-05-25 22:07:33,843 - INFO: Creating instance 'auto-testautovnf1-uas-0' and attaching
volume 'auto-testautovnf1-uas-vol-0'
2017-05-25 22:08:00,717 - INFO: Created instance 'auto-testautovnf1-uas-0'
2017-05-25 22:08:00,717 - INFO: Creating instance 'auto-testautovnf1-uas-1' and attaching
volume 'auto-testautovnf1-uas-vol-1'
2017-05-25 22:08:27,577 - INFO: Created instance 'auto-testautovnf1-uas-1'
2017-05-25 22:08:27,578 - INFO: Creating instance 'auto-testautovnf1-uas-2' and attaching
volume 'auto-testautovnf1-uas-vol-2'
2017-05-25 22:08:58,345 - INFO: Created instance 'auto-testautovnf1-uas-2'
2017-05-25 22:08:58,345 - INFO: Deploy request completed
2017-05-25 22:14:07,201 - INFO: Received request to download file system.cfg from config
named vnf-pkg1
2017-05-25 22:19:05,050 - INFO: Received a request to list config folder names.
2017-05-25 22:19:05,051 - INFO: config contents are: vnf-pkg1-images,vnf-pkg1
2017-05-25 22:19:05,059 - INFO: Received a request to list config folder names.
2017-05-25 22:19:05,059 - INFO: config contents are: vnf-pkg1-images,vnf-pkg1
2017-05-25 22:19:05,066 - INFO: Received a request to create a configuration folder.
2017-05-25 22:19:05,073 - INFO: Received a request to create a configuration folder.
2017-05-25 22:19:05,076 - INFO: Received request to download package: system.cfg from ISO
2017-05-25 22:19:05,083 - INFO: Received request to download package: system.cfg from ISO
2017-05-25 22:19:05,085 - INFO: Received request to download package: system.cfg from ISO
2017-05-25 22:19:05,086 - INFO: Received request to upload config file system.cfg to config
 named vnf-pkg2
2017-05-25 22:19:05,087 - INFO: Uploaded file system.cfg to config named vnf-pkg2
2017-05-25 22:19:59,895 - INFO: Received request to upload ISO usp-5_1_0.iso
2017-05-25 22:19:59,895 - INFO: Saving ISO to /tmp/tmpWbdnxm/usp-5_1_0.iso
2017-05-25 22:20:21,395 - INFO: Mounting ISO to /tmp/tmpWbdnxm/iso_mount
2017-05-25 22:20:22,288 - INFO: ISO version already installed, (5.1.0-662)
2017-05-25 22:20:23,203 - INFO: Received a request to list file names in config named
vnf-pkg2.
2017-05-25 22:20:23,203 - INFO: config contents are: system.cfg
2017-05-25 22:20:23,211 - INFO: Received a request to list file names in config named
vnf-pkg2-images.
2017-05-25 22:20:23,211 - INFO: config contents are:
2017-05-25 22:20:23,220 - INFO: Received request to get ISO details 5.1.0-662
2017-05-25 22:20:23,251 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:20:23,621 - INFO: Getting Host Aggregate failed: Aggregate
'auto-test-sjc-em-autovnf-mgmt2' not found on OpenStack setup
2017-05-25 22:20:23,633 - INFO: Received a request to deploy an Host Aggregate
2017-05-25 22:20:24,301 - INFO: Deploying Host Aggregate 'auto-test-sjc-em-autovnf-mgmt2'
completed
2017-05-25 22:20:24,313 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:20:24,843 - INFO: Getting Host Aggregate failed: Aggregate
'auto-test-sjc-service2' not found on OpenStack setup
2017-05-25 22:20:24,853 - INFO: Received a request to deploy an Host Aggregate
2017-05-25 22:20:25,524 - INFO: Deploying Host Aggregate 'auto-test-sjc-service2' completed
2017-05-25 22:20:25,537 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:20:25,898 - INFO: Getting Host Aggregate failed: Aggregate
'auto-test-sjc-cf-esc-mgmt2' not found on OpenStack setup
2017-05-25 22:20:25,909 - INFO: Received a request to deploy an Host Aggregate
2017-05-25 22:20:26,540 - INFO: Deploying Host Aggregate 'auto-test-sjc-cf-esc-mgmt2'
```

```
completed
2017-05-25 22:20:26,584 - INFO: Received a request to deploy AutoVnf
2017-05-25 22:20:31,604 - INFO: Creating AutoVnf deployment (3 instance(s)) on
'http://172.21.201.217:5000/v2.0' tenant 'core' user 'core', ISO '5.1.0-662'
2017-05-25 22:20:31,605 - INFO: Creating network 'auto-testautovnf2-uas-management'
2017-05-25 22:20:33,720 - INFO: Created network 'auto-testautovnf2-uas-management'
2017-05-25 22:20:33,720 - INFO: Creating network 'auto-testautovnf2-uas-orchestration'
2017-05-25 22:20:34,324 - INFO: Created network 'auto-testautovnf2-uas-orchestration'
2017-05-25 22:20:34,402 - INFO: Created flavor 'auto-testautovnf2-uas'
2017-05-25 22:20:34,402 - INFO: Loading image 'auto-testautovnf2-usp-uas-1.0.0-601.qcow2'
from '/opt/cisco/usp/bundles/5.1.0-662/uas-bundle/usp-uas-1.0.0-601.qcow2'
2017-05-25 22:20:43,169 - INFO: Loaded image 'auto-testautovnf2-usp-uas-1.0.0-601.qcow2'
2017-05-25 22:20:43,169 - INFO: Creating volume 'auto-testautovnf2-uas-vol-0' with command

[/opt/cisco/usp/apps/auto-it/vnf/../common/autoit/../autoit_os_utils/scripts/autoit_volume_staging.sh
 OS_USERNAME core OS_TENANT_NAME core OS_PASSWORD **** OS_AUTH_URL
http://172.21.201.217:5000/v2.0 ARG_TENANT core ARG_DEPLOYMENT test-uas ARG_VM_NAME
auto-testautovnf2-uas-vol-0 ARG_VOLUME_TYPE LUKS  FILE_1 /tmp/tmpe1mMIL/encrypted.cfg]
2017-05-25 22:20:54,713 - INFO: Created volume 'auto-testautovnf2-uas-vol-0'
2017-05-25 22:20:54,714 - INFO: Creating volume 'auto-testautovnf2-uas-vol-1' with command

[/opt/cisco/usp/apps/auto-it/vnf/../common/autoit/../autoit_os_utils/scripts/autoit_volume_staging.sh
 OS_USERNAME core OS_TENANT_NAME core OS_PASSWORD **** OS_AUTH_URL
http://172.21.201.217:5000/v2.0 ARG_TENANT core ARG_DEPLOYMENT test-uas ARG_VM_NAME
auto-testautovnf2-uas-vol-1 ARG_VOLUME_TYPE LUKS  FILE_1 /tmp/tmpe1mMIL/encrypted.cfg]
2017-05-25 22:21:06,203 - INFO: Created volume 'auto-testautovnf2-uas-vol-1'
2017-05-25 22:21:06,204 - INFO: Creating volume 'auto-testautovnf2-uas-vol-2' with command

[/opt/cisco/usp/apps/auto-it/vnf/../common/autoit/../autoit_os_utils/scripts/autoit_volume_staging.sh
 OS_USERNAME core OS_TENANT_NAME core OS_PASSWORD **** OS_AUTH_URL
http://172.21.201.217:5000/v2.0 ARG_TENANT core ARG_DEPLOYMENT test-uas ARG_VM_NAME
auto-testautovnf2-uas-vol-2 ARG_VOLUME_TYPE LUKS  FILE_1 /tmp/tmpe1mMIL/encrypted.cfg]
2017-05-25 22:21:18,184 - INFO: Created volume 'auto-testautovnf2-uas-vol-2'
2017-05-25 22:21:19,626 - INFO: Assigned floating IP '172.21.201.64' to IP '172.67.11.101'
2017-05-25 22:21:22,762 - INFO: Creating instance 'auto-testautovnf2-uas-0' and attaching
volume 'auto-testautovnf2-uas-vol-0'
2017-05-25 22:21:49,741 - INFO: Created instance 'auto-testautovnf2-uas-0'
2017-05-25 22:21:49,742 - INFO: Creating instance 'auto-testautovnf2-uas-1' and attaching
volume 'auto-testautovnf2-uas-vol-1'
2017-05-25 22:22:16,881 - INFO: Created instance 'auto-testautovnf2-uas-1'
2017-05-25 22:22:16,881 - INFO: Creating instance 'auto-testautovnf2-uas-2' and attaching
volume 'auto-testautovnf2-uas-vol-2'
2017-05-25 22:22:43,304 - INFO: Created instance 'auto-testautovnf2-uas-2'
2017-05-25 22:22:43,304 - INFO: Deploy request completed
2017-05-25 22:28:08,865 - INFO: Received request to download file system.cfg from config
named vnf-pkg2
2017-05-25 22:40:03,550 - INFO: Received request to download file system.cfg from config
named vnf-pkg1
```

Example Termination Log:

```
2017-05-25 22:53:30,970 - INFO: Received a request to destroy AutoVnf
2017-05-25 22:53:31,310 - INFO: Destroying AutoVnf deployment on
'http://172.21.201.217:5000/v2.0' tenant 'core' user 'core', ISO '5.1.0-662'
2017-05-25 22:53:32,698 - INFO: Removed floating IP '172.21.201.64'
2017-05-25 22:53:34,114 - INFO: 3 instance(s) found with name matching 'auto-testautovnf2'
2017-05-25 22:53:34,448 - INFO: Removing volume 'auto-testautovnf2-uas-vol-2'
2017-05-25 22:53:43,481 - INFO: Removed volume 'auto-testautovnf2-uas-vol-2'
2017-05-25 22:53:43,481 - INFO: Removing instance 'auto-testautovnf2-uas-2'
2017-05-25 22:53:47,080 - INFO: Removed instance 'auto-testautovnf2-uas-2'
2017-05-25 22:53:47,283 - INFO: Removing volume 'auto-testautovnf2-uas-vol-1'
2017-05-25 22:53:56,508 - INFO: Removed volume 'auto-testautovnf2-uas-vol-1'
2017-05-25 22:53:56,508 - INFO: Removing instance 'auto-testautovnf2-uas-1'
2017-05-25 22:54:00,290 - INFO: Removed instance 'auto-testautovnf2-uas-1'
2017-05-25 22:54:00,494 - INFO: Removing volume 'auto-testautovnf2-uas-vol-0'
2017-05-25 22:54:04,714 - INFO: Removed volume 'auto-testautovnf2-uas-vol-0'
2017-05-25 22:54:04,714 - INFO: Removing instance 'auto-testautovnf2-uas-0'
2017-05-25 22:54:11,647 - INFO: Removed instance 'auto-testautovnf2-uas-0'
2017-05-25 22:54:15,107 - INFO: 1 image(s) 'auto-testautovnf2-usp-uas-1.0.0-601.qcow2'
found, removing
2017-05-25 22:54:19,289 - INFO: Removed network 'auto-testautovnf2-uas-management'
2017-05-25 22:54:20,463 - INFO: Removed network 'auto-testautovnf2-uas-orchestration'
2017-05-25 22:54:20,541 - INFO: Removed flavor 'auto-testautovnf2-uas'
```

```
2017-05-25 22:54:20,541 - INFO: Destroy request completed
2017-05-25 22:54:20,562 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:54:20,925 - INFO: Getting Host Aggregate 'auto-test-sjc-em-autovnf-mgmt2'
completed
2017-05-25 22:54:20,940 - INFO: Received a request to destroy an Host Aggregate
2017-05-25 22:54:21,564 - INFO: Destroying Host Aggregate 'auto-test-sjc-em-autovnf-mgmt2'
 completed
2017-05-25 22:54:21,575 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:54:21,930 - INFO: Getting Host Aggregate 'auto-test-sjc-service2' completed
2017-05-25 22:54:21,947 - INFO: Received a request to destroy an Host Aggregate
2017-05-25 22:54:22,456 - INFO: Destroying Host Aggregate 'auto-test-sjc-service2' completed
2017-05-25 22:54:22,468 - INFO: Received a request to get an Host Aggregate details
2017-05-25 22:54:22,826 - INFO: Getting Host Aggregate 'auto-test-sjc-cf-esc-mgmt2' completed
2017-05-25 22:54:22,840 - INFO: Received a request to destroy an Host Aggregate
2017-05-25 22:54:23,394 - INFO: Destroying Host Aggregate 'auto-test-sjc-cf-esc-mgmt2'
completed
2017-05-25 22:56:55,925 - INFO: Received a request to destroy AutoVnf
2017-05-25 22:56:56,391 - INFO: Destroying AutoVnf deployment on
'http://172.21.201.217:5000/v2.0' tenant 'core' user 'core', ISO '5.1.0-662'
2017-05-25 22:56:57,507 - INFO: Removed floating IP '172.21.201.59'
2017-05-25 22:56:58,614 - INFO: 3 instance(s) found with name matching 'auto-testautovnf1'
2017-05-25 22:56:58,949 - INFO: Removing volume 'auto-testautovnf1-uas-vol-2'
2017-05-25 22:57:08,166 - INFO: Removed volume 'auto-testautovnf1-uas-vol-2'
2017-05-25 22:57:08,166 - INFO: Removing instance 'auto-testautovnf1-uas-2'
2017-05-25 22:57:15,117 - INFO: Removed instance 'auto-testautovnf1-uas-2'
2017-05-25 22:57:15,323 - INFO: Removing volume 'auto-testautovnf1-uas-vol-1'
2017-05-25 22:57:24,501 - INFO: Removed volume 'auto-testautovnf1-uas-vol-1'
2017-05-25 22:57:24,502 - INFO: Removing instance 'auto-testautovnf1-uas-1'
2017-05-25 22:57:28,275 - INFO: Removed instance 'auto-testautovnf1-uas-1'
2017-05-25 22:57:28,722 - INFO: Removing volume 'auto-testautovnf1-uas-vol-0'
2017-05-25 22:57:37,702 - INFO: Removed volume 'auto-testautovnf1-uas-vol-0'
2017-05-25 22:57:37,703 - INFO: Removing instance 'auto-testautovnf1-uas-0'
2017-05-25 22:57:44,622 - INFO: Removed instance 'auto-testautovnf1-uas-0'
2017-05-25 22:57:47,921 - INFO: 1 image(s) 'auto-testautovnf1-usp-uas-1.0.0-601.qcow2'
found, removing
2017-05-25 22:57:52,453 - INFO: Removed network 'auto-testautovnf1-uas-management'
2017-05-25 22:57:53,677 - INFO: Removed network 'auto-testautovnf1-uas-orchestration'
2017-05-25 22:57:53,760 - INFO: Removed flavor 'auto-testautovnf1-uas'
2017-05-25 22:57:53,760 - INFO: Destroy request completed
```

Log Levels

To enable debug level logging for detailed troubleshooting:

curl -X POST http:/0.0.0.0:5001/debugs

To revert to the default logging level:

curl -X DELETE http:/0.0.0.0:5001/debugs

Checking AutoIT-VNF Processes

Verify that key processes are running on the AutoIT-VNF VM:

service autoit status

Example output:

```
AutoIT-VNF is running.
```

Stopping/Restarting AutoIT-VNF Processes

To stop the AutoIT-VNF processes:

service autoit stop

Example output:

```
AutoIT-VNF API server stopped.
```

To restart the AutoIT-VNF processes:

service autoit restart

Example output:

```
AutoIT-VNF API server stopped.
Starting AutoIT-VNF
/opt/cisco/usp/apps/auto-it/vnf
AutoIT API server started.
```

Monitoring AutoVNF Operations

This section identifies various commands that can be used to determine the status and health of AutoVNF.

To use them, you must:

1. Log on to the AutoVNF VM as *ubuntu*. Use the password that was created earlier for this user.
2. Become the root user.

   sudo -i

Checking AutoVNF VM Health

The **uas-check.py** script provides basic health-checking and recovery of VMs that are part of the AutoVNF cluster.

The script determines VM health from information retrieved from OpenStack. It then reports the health, identifies any errors and whether or not they are recoverable. If they are recoverable, the script provides you with the opportunity to correct the error.

uas-check.py is part of the UAS bundle. Upon installation, the script is located on the Ultra M Manager Node or Onboarding Server in the */opt/cisco/usp/uas-installer/scripts/* directory.

To run the script:

1. Navigate to the *scripts* directory.

   cd /opt/cisco/usp/uas-installer/scripts
2. Launch the *uas-check.py* script.

   ./uas-check.py auto-vnf <*deployment_name*>

   Example:

   ./uas-check.py auto-vnf auto-autovnf1

```
2017-05-25 10:36:15,050 - INFO: Check of AutoVNF cluster started
2017-05-25 10:36:17,611 - INFO: Found 3 ACTIVE AutoVNF instances
2017-05-25 10:36:17,611 - INFO: Check completed, AutoVNF cluster is fine
```

Note Additional arguments and options for running the script are available and described in the scripts help text. Execute the following command to access the script's help:

./uas-check.py -h

Checking AutoVNF and UAS-Related Processes

AutoVNF and UAS Processes

To ensure that processes required by the UAS are running by executing the following commands:

- **initctl status autovnf**
- **initctl status uws-ae**
- **initctl status uas-confd**
- **initctl status cluster_manager**
- **initctl status uas_manager**

For each process, you should see a message similar to the following indicating that the process is running:

```
autovnf start/running, process 2206
```

Python Processes

To verify that the Python process is running:

ps -ef | grep python

Example output:

```
root      2194  1970 81 22:28 ?        00:16:36 python
/opt/cisco/usp/uas/manager/uas_manager.py
root      2201     1  0 22:28 ?        00:00:00 python
/opt/cisco/usp/uas/autovnf/usp_autovnf.py
root      2227  2202 99 22:28 ?        00:20:22 python
/opt/cisco/usp/uas/manager/cluster_manager.py
root      3939  3920  0 22:48 pts/0    00:00:00 grep --color=auto python
```

ConfD Processes

To verify that ConfD is running:

ps -ef | grep confd

Example output:

```
root      2149  2054  0 22:28 ?        00:00:03
/opt/cisco/usp/uas/confd-6.1/lib/confd/erts/bin/confd -K false -B -MHe true -- -root
/opt/cisco/usp/uas/confd-6.1/lib/confd -progname confd -- -home / -- -smp disable
-code_path_cache -boot confd -noshell -noinput -foreground -verbose -shutdown_time 30000
-conffile /opt/cisco/usp/uas/confd-6.1/etc/confd/confd.conf -max_fds 1024
root      3945  3920  0 22:48 pts/0    00:00:00 grep --color=auto confd
```

ZooKeeper Processes

To verify that ZooKeeper is running (for HA functionality):

ps -ef | grep java

Example output:

```
root      1183     1  2 22:27 ?        00:00:34 /usr/bin/java -jar
/opt/cisco/usp/uws/ae/java/uws-ae-0.1.0.jar
zk        1388     1 18 22:27 ?        00:03:55 java
-Dzookeeper.log.dir=/var/log/cisco-uas/zookeeper -Dzookeeper.root.logger=INFO,ROLLINGFILE
-cp /opt/cisco/usp/packages/zookeeper/current/bin/../build
/classes:/opt/cisco/usp/packages/zookeeper/current/bin/../build/lib
/*.jar:/opt/cisco/usp/packages/zookeeper/current/bin/../lib/slf4j-log4j12-1.6.1.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../lib/slf4j-api-1.6.1.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../lib/netty-3.7.0.Final.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../lib/log4j-1.2.16.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../lib/jline-0.9.94.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../zookeeper-3.4.8.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../src/java/lib/*.jar:
/opt/cisco/usp/packages/zookeeper/current/bin/../conf: -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.local.only=false
org.apache.zookeeper.server.quorum.QuorumPeerMain
/opt/cisco/usp/packages/zookeeper/current/bin/../conf/zoo.cfg
root      3955  3920  0 22:48 pts/0    00:00:00 grep --color=auto java
```

Note If there are any issues seen when executing the above commands, please refer to the following sections:

- Viewing AutoVNF Logs, on page 45
- Viewing AutoVNF Operational Data, on page 50
- Viewing UAS HA Logs, on page 56
- Viewing UAS Manager Logs, on page 57
- Viewing ZooKeeper Logs, on page 57

Viewing AutoVNF Logs

General AutoVNF Logs

AutoVNF logs are available on the AutoVNF VM in the following file:

/var/log/upstart/autovnf.log

To collect AutoVNF logs:

1. Navigate to the *scripts* directory.

   cd /opt/cisco/usp/uas/scripts

2. Launch the *collect-uas-logs.sh* script to collect the logs.

   sudo ./collect-uas-logs.sh

   Example log output:

```
Creating log tarball uas-logs-2017-05-26_00.24.55_UTC.tar.bz2 ...
uas-logs/
uas-logs/autovnf/
uas-logs/autovnf/autovnf_server.log
uas-logs/autovnf/a15bf26c-41a1-11e7-b3ab-fa163eccaffc/
uas-logs/autovnf/a15bf26c-41a1-11e7-b3ab-fa163eccaffc/netconf_traces
uas-logs/autovnf/a15bf26c-41a1-11e7-b3ab-fa163eccaffc/vnfd
uas-logs/autovnf/audit.log
uas-logs/autovnf/579b4546-41a2-11e7-b3ab-fa163eccaffc/
uas-logs/autovnf/579b4546-41a2-11e7-b3ab-fa163eccaffc/netconf_traces
```

```
uas-logs/autovnf/579b4546-41a2-11e7-b3ab-fa163eccaffc/vnfd
uas-logs/ha/
uas-logs/ha/info.log
uas-logs/uas_manager/
uas-logs/uas_manager/info.log
uas-logs/zk/
uas-logs/zk/zookeeper.out
uas-logs/zk/zookeeper.log
uas-logs/upstart/
uas-logs/upstart/uas-confd.log
uas-logs/upstart/zk.log
uas-logs/upstart/autovnf.log
uas-logs/upstart/uws-ae.log
uas-logs/upstart/ensemble.log

=============== Tarball available at: /tmp/uas-logs-2017-05-26_00.24.55_UTC.tar.bz2
===============

To extract the tarball, run: "tar jxf /tmp/uas-logs-2017-05-26_00.24.55_UTC.tar.bz2"
```

AutoVNF Transaction Logs

AutoVNF server and transaction logs are available on the Ultra M Manager Node in the following directory on the UAS VM:

/var/log/cisco-uas/autovnf

Inside this directory are transaction sub-directories, VNFD information and NETCONF traces are provided for the given transaction.

Example:

```
total 3568
drwxr-xr-x 4 root root    4096 May 25 23:31 ./
drwxr-xr-x 7 root root    4096 May 25 19:39 ../
drwxr-xr-x 2 root root    4096 May 25 23:31 579b4546-41a2-11e7-b3ab-fa163eccaffc/
drwxr-xr-x 2 root root    4096 May 25 23:29 a15bf26c-41a1-11e7-b3ab-fa163eccaffc/
-rw-r--r-- 1 root root 3632813 May 26 18:33 audit.log
-rw-r--r-- 1 root root       0 May 25 23:26 autovnf_server.log

cd a15bf26c-41a1-11e7-b3ab-fa163eccaffc
total 2568
drwxr-xr-x 2 root root    4096 May 25 23:29 ./
drwxr-xr-x 4 root root    4096 May 25 23:31 ../
-rw-r--r-- 1 root root 2614547 May 25 23:37 netconf_traces
-rw-r--r-- 1 root root       0 May 25 23:29 vnfd
```

AutoVNF Event Logs

Event logs provide useful information on UAS task progress. These logs are located in the *autovnf.log* file within the following directory on the UAS VM:

/var/log/upstart

Event logs are filed by transaction ID. To view transaction IDs:

1. Login to the ConfD CLI as the admin user.

   confd_cli -u admin -C
2. List the transactions.

   show transactions
   Example output:

```
TX ID                                TX TYPE          DEPLOYMENT ID       TIMESTAMP
                     STATUS
```

```
562c18b0-4199-11e7-ad05-fa163ec6a7e4  vnf-deployment   vnfd2-deployment
2017-05-25T22:27:28.962293-00:00  deployment-success
abf51428-4198-11e7-ad05-fa163ec6a7e4  vnfm-deployment  ab-auto-test-vnfm2
2017-05-25T22:22:43.389059-00:00  deployment-success
```

To view the logs associated with a specific transaction:

show logs <*transaction_id*>**| display xml**

Example log pertaining to VNFM deployment:

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <logs xmlns="http://www.cisco.com/usp/nfv/usp-autovnf-oper">
    <tx-id>abf51428-4198-11e7-ad05-fa163ec6a7e4</tx-id>
    <log>2017-05-25 22:22:43,402 - VNFM Deployment RPC triggered for deployment:
ab-auto-test-vnfm2, deactivate: 0
2017-05-25 22:22:43,446 - Notify deployment
2017-05-25 22:22:43,472 - VNFM Transaction: abf51428-4198-11e7-ad05-fa163ec6a7e4 for
deployment: ab-auto-test-vnfm2 started
2017-05-25 22:22:43,497 - Downloading Image:
http://172.21.201.63:80/bundles/5.1.0-662/vnfm-bundle/ESC-2_3_2_143.qcow2
2017-05-25 22:22:49,146 - Image: //opt/cisco/vnf-staging/vnfm_image downloaded successfully
2017-05-25 22:22:49,714 - Checking network 'public' existence
2017-05-25 22:22:49,879 - Checking flavor 'ab-auto-test-vnfm2-ESC-flavor' non existence
2017-05-25 22:22:50,124 - Checking image 'ab-auto-test-vnfm2-ESC-image' non existence
2017-05-25 22:22:50,598 - Checking network 'auto-testautovnf2-uas-management' existence
2017-05-25 22:22:50,752 - Checking network 'auto-testautovnf2-uas-orchestration' existence
2017-05-25 22:22:50,916 - Checking instance 'ab-auto-test-vnfm2-ESC-0' non existence
2017-05-25 22:22:51,357 - Checking instance 'ab-auto-test-vnfm2-ESC-1' non existence
2017-05-25 22:22:52,084 - Creating flavor 'ab-auto-test-vnfm2-ESC-flavor'
2017-05-25 22:22:52,184 - Loading image 'ab-auto-test-vnfm2-ESC-image' from
'//opt/cisco/vnf-staging/vnfm_image'...
2017-05-25 22:23:06,444 - ESC HA mode is ON
2017-05-25 22:23:07,118 - Allocated these IPs for ESC HA: ['172.67.11.3', '172.67.11.4',
 '172.67.11.5']
2017-05-25 22:23:08,228 - Creating VNFM 'ab-auto-test-vnfm2-ESC-0' with [python
//opt/cisco/vnf-staging/bootvm.py ab-auto-test-vnfm2-ESC-0 --flavor
ab-auto-test-vnfm2-ESC-flavor --image b29e7a72-9ad0-4178-aa35-35df0a2b23b7 --net
auto-testautovnf2-uas-management --gateway_ip 172.67.11.1 --net
auto-testautovnf2-uas-orchestration
 --os_auth_url http://172.21.201.217:5000/v2.0 --os_tenant_name core --os_username ******
 --os_password ****** --bs_os_auth_url http://172.21.201.217:5000/v2.0 --bs_os_tenant_name
 core --bs_os_username ****** --bs_os_password ****** --esc_ui_startup false
 --esc_params_file /tmp/esc_params.cfg --encrypt_key ****** --user_pass ******
--user_confd_pass ****** --kad_vif eth0 --kad_vip 172.67.11.5 --ipaddr 172.67.11.3 dhcp
 --ha_node_list 172.67.11.3 172.67.11.4 --file
root:0755:/opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh:
/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_volume_em_staging.sh
 --file
root:0755:/opt/cisco/esc/esc-scripts/esc_vpc_chassis_id.py:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_vpc_chassis_id.py
 --file
root:0755:/opt/cisco/esc/esc-scripts/esc-vpc-di-internal-keys.sh:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc-vpc-di-internal-keys.sh]...
2017-05-25 22:24:13,329 - ESC started!
2017-05-25 22:24:13,803 - Creating VNFM 'ab-auto-test-vnfm2-ESC-1' with [python
//opt/cisco/vnf-staging/bootvm.py ab-auto-test-vnfm2-ESC-1 --flavor
ab-auto-test-vnfm2-ESC-flavor --image b29e7a72-9ad0-4178-aa35-35df0a2b23b7 --net
auto-testautovnf2-uas-management --gateway_ip 172.67.11.1 --net
auto-testautovnf2-uas-orchestration
 --os_auth_url http://172.21.201.217:5000/v2.0 --os_tenant_name core --os_username ******
 --os_password ****** --bs_os_auth_url http://172.21.201.217:5000/v2.0 --bs_os_tenant_name
 core --bs_os_username ****** --bs_os_password ****** --esc_ui_startup false
--esc_params_file /tmp/esc_params.cfg --encrypt_key ****** --user_pass ******
--user_confd_pass ****** --kad_vif eth0 --kad_vip 172.67.11.5 --ipaddr 172.67.11.4 dhcp
 --ha_node_list 172.67.11.3 172.67.11.4
 --file root:0755:/opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh:
/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_volume_em_staging.sh --file
root:0755:/opt/cisco/esc/esc-scripts/esc_vpc_chassis_id.py:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_vpc_chassis_id.py
 --file
root:0755:/opt/cisco/esc/esc-scripts/esc-vpc-di-internal-keys.sh:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc-vpc-di-internal-keys.sh]...
2017-05-25 22:25:12,660 - ESC started!
2017-05-25 22:25:12,677 - Waiting for VIM to declare 2 instance(s) active
2017-05-25 22:25:18,254 - Instance(s) are active
2017-05-25 22:25:18,271 - Waiting for VNFM to be ready...
```

```
2017-05-25 22:25:18,292 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:25:21,313 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:25:31,341 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:25:31,362 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:25:41,379 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:25:41,397 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:25:51,424 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:25:51,495 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:26:01,521 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:26:01,539 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:26:11,563 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:26:11,591 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:26:21,617 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:26:21,635 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:26:31,662 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:26:31,680 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:26:41,706 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:26:41,726 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:26:51,748 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:26:51,765 - Could not estabilish NETCONF session to 172.67.11.5
2017-05-25 22:27:01,791 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:27:02,204 - NETConf Sessions (Transaction/Notifications) estabilished
2017-05-25 22:27:02,507 - Notify VNFM Up
2017-05-25 22:27:02,525 - VNFM Transaction: abf51428-4198-11e7-ad05-fa163ec6a7e4 for
deployment: ab-auto-test-vnfm2 completed suc-cessfully.
2017-05-25 22:27:02,545 - Notify deployment</log>
  </logs>
</config>
```

Example log pertaining to VNF deployment:

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <logs xmlns="http://www.cisco.com/usp/nfv/usp-autovnf-oper">
    <tx-id>562c18b0-4199-11e7-ad05-fa163ec6a7e4</tx-id>
    <log>2017-05-25 22:27:29,039 - Notify deployment
2017-05-25 22:27:29,062 - Connection to VNFM (esc) at 172.67.11.5
2017-05-25 22:27:29,404 - NETConf Sessions (Transaction/Notifications) estabilished
2017-05-25 22:27:29,420 - Get Images
2017-05-25 22:27:29,435 - NETCONF get-config Request sent, waiting for reply
2017-05-25 22:27:29,560 - NETCONF Transaction success!
2017-05-25 22:27:29,570 - Get Flavors List
2017-05-25 22:27:29,582 - Adding images ...
2017-05-25 22:27:29,592 - Creating Images
2017-05-25 22:27:29,603 -   image: ab-auto-test-vnfm2-element-manager
2017-05-25 22:27:29,620 -     src:
http://172.21.201.63:80/bundles/5.1.0-662/em-bundle/em-1_0_0_532.qcow2
2017-05-25 22:27:29,630 -     disk_format: qcow2
2017-05-25 22:27:29,641 -     container_format: bare
2017-05-25 22:27:29,655 -     serial_console: True
2017-05-25 22:27:29,665 -     disk_bus: virtio
2017-05-25 22:27:29,674 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:27:29,901 - NETCONF Transaction success!
2017-05-25 22:27:29,911 - Waiting for VNFM to process CREATE_IMAGE transaction
2017-05-25 22:27:46,987 - | CREATE_IMAGE | ab-auto-test-vnfm2-element-manager | SUCCESS
 | (1/1)
2017-05-25 22:27:47,004 - NETCONF transaction completed successfully!
2017-05-25 22:27:47,749 - Creating Images
2017-05-25 22:27:47,764 -   image: ab-auto-test-vnfm2-control-function
2017-05-25 22:27:47,776 -     src:
http://172.21.201.63:80/bundles/5.1.0-662/ugp-bundle/qvpc-di-cf.qcow2
2017-05-25 22:27:47,793 -     disk_format: qcow2
2017-05-25 22:27:47,805 -     container_format: bare
2017-05-25 22:27:47,819 -     serial_console: True
2017-05-25 22:27:47,831 -     disk_bus: virtio
2017-05-25 22:27:47,841 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:27:48,317 - NETCONF Transaction success!
2017-05-25 22:27:48,331 - Waiting for VNFM to process CREATE_IMAGE transaction
2017-05-25 22:27:56,403 - | CREATE_IMAGE | ab-auto-test-vnfm2-control-function | SUCCESS
 | (1/1)
2017-05-25 22:27:56,434 - NETCONF transaction completed successfully!
2017-05-25 22:27:56,822 - Creating Images
2017-05-25 22:27:56,838 -   image: ab-auto-test-vnfm2-session-function
2017-05-25 22:27:57,267 -     src:
http://172.21.201.63:80/bundles/5.1.0-662/ugp-bundle/qvpc-di-sf.qcow2
```

```
2017-05-25 22:27:57,412 -     disk_format: qcow2
2017-05-25 22:27:57,423 -     container_format: bare
2017-05-25 22:27:57,523 -     serial_console: True
2017-05-25 22:27:57,535 -     disk_bus: virtio
2017-05-25 22:27:57,550 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:27:58,378 - NETCONF Transaction success!
2017-05-25 22:27:58,391 - Waiting for VNFM to process CREATE_IMAGE transaction
2017-05-25 22:28:06,339 - | CREATE_IMAGE | ab-auto-test-vnfm2-session-function | SUCCESS
 | (1/1)
2017-05-25 22:28:06,355 - NETCONF transaction completed successfully!
2017-05-25 22:28:06,367 - Images added successfully
2017-05-25 22:28:06,378 - Creating flavors ...
2017-05-25 22:28:06,388 - Creating flavors
2017-05-25 22:28:06,432 -   flavor: ab-auto-test-vnfm2-element-manager
2017-05-25 22:28:06,444 -     vcpus: 2
2017-05-25 22:28:06,457 -     memory_mb: 4096
2017-05-25 22:28:06,469 -     root_disk_mb: 40960
2017-05-25 22:28:06,481 -     ephemeral_disk_mb: 0
2017-05-25 22:28:06,491 -     swap_disk_mb: 0
2017-05-25 22:28:06,505 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:28:06,781 - NETCONF Transaction success!
2017-05-25 22:28:06,793 - Waiting for VNFM to process CREATE_FLAVOR transaction
2017-05-25 22:28:07,286 - | CREATE_FLAVOR | ab-auto-test-vnfm2-element-manager | SUCCESS
 | (1/1)
2017-05-25 22:28:07,298 - NETCONF transaction completed successfully!
2017-05-25 22:28:07,310 - Creating flavors
2017-05-25 22:28:07,328 -   flavor: ab-auto-test-vnfm2-control-function
2017-05-25 22:28:07,341 -     vcpus: 8
2017-05-25 22:28:07,358 -     memory_mb: 16384
2017-05-25 22:28:07,374 -     root_disk_mb: 6144
2017-05-25 22:28:07,386 -     ephemeral_disk_mb: 0
2017-05-25 22:28:07,398 -     swap_disk_mb: 0
2017-05-25 22:28:07,410 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:28:07,586 - NETCONF Transaction success!
2017-05-25 22:28:07,603 - Waiting for VNFM to process CREATE_FLAVOR transaction
2017-05-25 22:28:07,818 - | CREATE_FLAVOR | ab-auto-test-vnfm2-control-function | SUCCESS
 | (1/1)
2017-05-25 22:28:07,830 - NETCONF transaction completed successfully!
2017-05-25 22:28:07,842 - Creating flavors
2017-05-25 22:28:07,853 -   flavor: ab-auto-test-vnfm2-session-function
2017-05-25 22:28:07,865 -     vcpus: 8
2017-05-25 22:28:07,877 -     memory_mb: 16384
2017-05-25 22:28:07,889 -     root_disk_mb: 6144
2017-05-25 22:28:07,901 -     ephemeral_disk_mb: 0
2017-05-25 22:28:07,917 -     swap_disk_mb: 0
2017-05-25 22:28:07,928 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:28:08,204 - NETCONF Transaction success!
2017-05-25 22:28:08,216 - Waiting for VNFM to process CREATE_FLAVOR transaction
2017-05-25 22:28:08,455 - | CREATE_FLAVOR | ab-auto-test-vnfm2-session-function | SUCCESS
 | (1/1)
2017-05-25 22:28:08,473 - NETCONF transaction completed successfully!
2017-05-25 22:28:08,489 - Flavors created successfully
2017-05-25 22:28:08,501 - Onboarding configuration file: ('control-function',
'staros_config.txt', 'http://172.21.201.63:5001/configs/vnf-pkg2/files/system.cfg')
2017-05-25 22:28:08,547 - NETCONF get-operational Request sent, waiting for reply
2017-05-25 22:28:08,724 - NETCONF Transaction success!
2017-05-25 22:28:08,855 - Notify VDU Create Catalog for : element-manager, status: SUCCESS,
 txid: 562c18b0-4199-11e7-ad05-fa163ec6a7e4
2017-05-25 22:28:08,892 - Notify VDU Create Catalog for : control-function, status:
SUCCESS, txid: 562c18b0-4199-11e7-ad05-fa163ec6a7e4
2017-05-25 22:28:09,008 - Notify VDU Create Catalog for : session-function, status:
SUCCESS, txid: 562c18b0-4199-11e7-ad05-fa163ec6a7e4
2017-05-25 22:28:09,024 - NETCONF get-config Request sent, waiting for reply
2017-05-25 22:28:09,151 - NETCONF Transaction success!
2017-05-25 22:28:14,837 - Deployment: vnfd2-deployment started ...
2017-05-25 22:28:14,858 - Generating VNFD
2017-05-25 22:28:14,930 - VNFD generated successfully.
2017-05-25 22:28:14,966 - Generating configuration files for EM
2017-05-25 22:28:14,979 - Creating VIP Ports
2017-05-25 22:28:16,970 - VIP ports created successfully
2017-05-25 22:28:16,987 - Deploging EM
2017-05-25 22:28:17,000 - Added anti-affinity placement policy for ab-auto-test-vnfm2-em-1
2017-05-25 22:28:17,012 - Added anti-affinity placement policy for ab-auto-test-vnfm2-em-2
```

```
2017-05-25 22:28:17,025 - Added anti-affinity placement policy for ab-auto-test-vnfm2-em-3
2017-05-25 22:28:17,041 - Starting Service Deployment: ab-auto-test-vnfm2-em
2017-05-25 22:28:17,054 - Start VM: ab-auto-test-vnfm2-em-1
2017-05-25 22:28:17,066 - Start VM: ab-auto-test-vnfm2-em-2
2017-05-25 22:28:17,077 - Start VM: ab-auto-test-vnfm2-em-3
2017-05-25 22:28:17,089 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 22:28:17,721 - NETCONF Transaction success!
2017-05-25 22:28:17,733 - Waiting for VNFM to process SERVICE_ALIVE transaction
2017-05-25 22:29:37,185 - | VM_DEPLOYED | ab-auto-test-vnfm2-em-1 | SUCCESS | Waiting
for: SERVICE_ALIVE|
2017-05-25 22:29:59,679 - | VM_ALIVE | ab-auto-test-vnfm2-em-1 | SUCCESS | Waiting for:
 SERVICE_ALIVE|
2017-05-25 22:30:42,170 - | VM_DEPLOYED | ab-auto-test-vnfm2-em-2 | SUCCESS | Waiting
for: SERVICE_ALIVE|
2017-05-25 22:30:59,620 - | VM_ALIVE | ab-auto-test-vnfm2-em-2 | SUCCESS | Waiting for:
 SERVICE_ALIVE|
2017-05-25 22:31:51,510 - | VM_DEPLOYED | ab-auto-test-vnfm2-em-3 | SUCCESS | Waiting
for: SERVICE_ALIVE|
2017-05-25 22:32:13,584 - | VM_DEPLOYED | c2 | SUCCESS | Waiting for: SERVICE_ALIVE|
2017-05-25 22:32:29,639 - | VM_ALIVE | ab-auto-test-vnfm2-em-3 | SUCCESS | Waiting for:
 SERVICE_ALIVE|
2017-05-25 22:32:29,661 - | SERVICE_ALIVE | ab-auto-test-vnfm2-em | SUCCESS | (1/1)
2017-05-25 22:32:29,674 - NETCONF transaction completed successfully!
2017-05-25 22:32:29,687 - EM Online !
2017-05-25 22:32:29,699 - HA-VIP[element-manager] : 172.67.11.12
2017-05-25 22:32:29,716 - HA-VIP[control-function] : 172.67.11.13
2017-05-25 22:32:29,729 - Deployment: vnfd2-deployment completed successfully.
2017-05-25 22:32:29,742 - NETCONF get-operational Request sent, waiting for reply
2017-05-25 22:32:30,221 - NETCONF Transaction success!
2017-05-25 22:32:30,261 - Notify EM Up
2017-05-25 22:32:30,274 - VNF Transaction completed successfully!
2017-05-25 22:32:30,292 - Notify deployment</log>
  </logs>
</config>
```

Viewing AutoVNF Operational Data

AutoVNF maintains history information for all transactions, associated events, and related error/information logs in persistent storage. These logs are useful for monitoring deployment progress and for troubleshooting issues.

These logs can be retrieved at time using the “task-id” returned as well as by running ConfD “show” commands.

To access these commands, you must be logged in to the Confd CLI as the admin user on the AutoVNF VM:

confd_cli -u admin -C

Table 2: ConfD Log Descriptions, on page 50 provides a list of the available commands and describes the information in the output.

Table 2: ConfD Log Descriptions

ConfD Command	Purpose
show autovnf-oper:errors	Displays a list of any deployment errors that may have occurred.
show autovnf-oper:logs | display xml	Displays log messages for AutoVNF transactions.
show autovnf-oper:network-catalog	Displays information for the networks deployed with USP.

ConfD Command	Purpose
show autovnf-oper:transactions	Displays a list of transaction IDs that correspond to the USP deployment along with their execution date, time, and status.
show autovnf-oper:vdu-catalog	Displays information pertaining to the virtual descriptor units (VDUs) used to deploy USP.
show autovnf-oper:vip-port	Displays information port, network, and virtual IP addresses information.
show autovnf-oper:vnf-em	Displays information pertaining to the UEM VM deployment.
show autovnf-oper:vnfm	Displays information pertaining to the VNFM deployment.
show confd-state	Displays information pertaining to confd-state on AutoVNF.
show confd-state ha	Displays information pertaining to HA specific confd-state on AutoVNF.
show logs <*transaction_id*>	Displays detailed log information for a specific transaction ID.
show running-config	Displays the configuration running on the AutoVNF.
show uas	Displays information pertaining to the AutoVNF VM deployment.
show usp	Displays information pertaining to the overall USP VM deployment.

NOTES:

- Log information can be saved out of ConfD to a file for later retrieval using one of the following commands:

  show logs *transaction_id* **| save** *url*
  OR

  show autovnf-oper: *command* **| save** *url*

  Where *transaction_id* is a specific ID, *url* is a valid directory path, and *command* is one of the command operators identified in Table 2: ConfD Log Descriptions, on page 50 .

Example show autovnf-oper:errors Command Output

show autovnf-oper:errors

```
% No entries found.
```

Note If no errors are found, the resulting output will look as above.

Example show autovnf-oper:logs Command Output

show autovnf-oper:logs | display xml

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <logs xmlns="http://www.cisco.com/usp/nfv/usp-autovnf-oper">
    <tx-id>579b4546-41a2-11e7-b3ab-fa163eccaffc</tx-id>
    <log>2017-05-25 23:31:56,911 - Notify deployment
2017-05-25 23:31:56,937 - Connection to VNFM (esc) at 172.57.11.6
2017-05-25 23:31:57,346 - NETConf Sessions (Transaction/Notifications) estabilished
2017-05-25 23:31:57,356 - Get Images
2017-05-25 23:31:57,370 - NETCONF get-config Request sent, waiting for reply
2017-05-25 23:31:57,500 - NETCONF Transaction success!
2017-05-25 23:31:57,515 - Get Flavors List
2017-05-25 23:31:57,525 - Adding images ...
2017-05-25 23:31:57,539 - Creating Images
2017-05-25 23:31:57,549 -   image: ab-auto-test-vnfm1-element-manager
2017-05-25 23:31:57,560 -     src:
http://172.21.201.63:80/bundles/5.1.0-662/em-bundle/em-1_0_0_532.qcow2
2017-05-25 23:31:57,573 -     disk_format: qcow2
2017-05-25 23:31:57,582 -     container_format: bare
2017-05-25 23:31:57,592 -     serial_console: True
2017-05-25 23:31:57,602 -     disk_bus: virtio
2017-05-25 23:31:57,614 - NETCONF edit-config Request sent, waiting for reply
2017-05-25 23:31:57,838 - NETCONF Transaction success!
2017-05-25 23:31:57,850 - Waiting for VNFM to process CREATE_IMAGE transaction
2017-05-25 23:32:15,129 - | CREATE_IMAGE | ab-auto-test-vnfm1-element-manager | SUCCESS |
(1/1)
2017-05-25 23:32:15,143 - NETCONF transaction completed successfully!
2017-05-25 23:32:15,156 - Creating Images
<-- SNIP -->
```

Example show autovnf-oper:transactions Command Output

show autovnf-oper:transactions

```
auto-testautovnf1-uas-0#show autovnf-oper:transactions
TX ID                                TX TYPE          DEPLOYMENT ID       TIMESTAMP
                  STATUS
------------------------------------------------------------------------------------------
579b4546-41a2-11e7-b3ab-fa163eccaffc  vnf-deployment   vnfd1-deployment
2017-05-25T23:31:56.839173-00:00  deployment-success
a15bf26c-41a1-11e7-b3ab-fa163eccaffc  vnfm-deployment  ab-auto-test-vnfm1
2017-05-25T23:26:51.078847-00:00  deployment-success
```

Example show autovnf-oper:vdu-catalog Command Output

show autovnf-oper:vdu-catalog

```
autovnf-oper:vdu-catalog control-function
 image-source http://172.21.201.63:80/bundles/5.1.0-653/ugp-bundle/qvpc-di-cf.qcow2
 vnfm-image   ab-auto-test-vnfm3-control-function
 image-id     b6848eca-6ec1-4ee3-bf9b-df6aa4a7c1e5
 vnfm-flavor  ab-auto-test-vnfm3-control-function
 flavor-id    bf932ae5-f022-473f-a26e-5065e59d5084
 configurations staros_config.txt
  config-source http://172.21.201.63:5001/configs/vnf-pkg3/files/system.cfg
  config-used   /config/control-function/staros_config.txt
autovnf-oper:vdu-catalog element-manager
 image-source http://172.21.201.63:80/bundles/5.1.0-653/em-bundle/em-1_0_0_523.qcow2
 vnfm-image   ab-auto-test-vnfm3-element-manager
```

```
 image-id     fad22774-e244-401d-84eb-d6a06ac0402f
 vnfm-flavor  ab-auto-test-vnfm3-element-manager
 flavor-id    cd78dfd5-b26e-46f9-ba59-fbdac978c6be
autovnf-oper:vdu-catalog session-function
 image-source http://172.21.201.63:80/bundles/5.1.0-653/ugp-bundle/qvpc-di-sf.qcow2
 vnfm-image   ab-auto-test-vnfm3-session-function
 image-id     a0957201-fec3-4931-9e35-3a75f3e2484a
 vnfm-flavor  ab-auto-test-vnfm3-session-function
 flavor-id    2453c945-ad14-4376-bb2d-0561afbf92e5
```

Example show autovnf-oper:vip-port Command Output

show autovnf-oper:vip-port

```
DEPLOYMENT ID     TRANSACTION ID                          PORT ID
              NETWORK                              HA VIP         VDU REF
-------------------------------------------------------------------------------------------------------------------------------
vnfd3-deployment  346293ae-40b7-11e7-879a-fa163efa324b
auto-testautovnf3-uas-management-172.77.11.12  auto-testautovnf3-uas-management  172.77.11.12
  element-manager

auto-testautovnf3-uas-management-172.77.11.13  auto-testautovnf3-uas-management  172.77.11.13
  control-function
```

Example show autovnf-oper:vnf-em Command Output

show autovnf-oper:vnf-em

```
--SNIP--
vnf-em vnfd-deployment
state alive
transaction-id 1508009048-329005
ha-vip 30.30.61.103
vnfc-instance vnfd-deployment-em-1
compute-host tb1ano-compute-4.localdomain
interfaces eth0
ip-address 30.30.62.5
mac-address fa:16:3e:ea:67:a7
interfaces eth1
ip-address 30.30.61.5
mac-address fa:16:3e:75:62:e5
vnfc-instance vnfd-deployment-em-2
compute-host tb1ano-compute-6.localdomain
interfaces eth0
ip-address 30.30.62.6
mac-address fa:16:3e:6f:09:82
interfaces eth1
ip-address 30.30.61.6
mac-address fa:16:3e:56:58:0e
vnfc-instance vnfd-deployment-em-3
compute-host tb1ano-compute-0.localdomain
interfaces eth0
ip-address 30.30.62.8
mac-address fa:16:3e:bc:2a:30
interfaces eth1
ip-address 30.30.61.7
mac-address fa:16:3e:8a:c0:f5
--SNIP--
```

Example show autovnf-oper:vnfm Command Output

show autovnf-oper:vnfm

```
autovnf-oper:vnfm ab-auto-test-vnfm3
 state          alive
 version        3.0.1.9
 transaction-id 7dacc0f8-40b6-11e7-879a-fa163efa324b
```

```
 ha-vip         172.77.11.7
 vnfc-instance ab-auto-test-vnfm3-ESC-0
  compute-host neutonoc-compute-9.localdomain
  interfaces auto-testautovnf3-uas-management
   ip-address  172.77.11.3
   mac-address fa:16:3e:5e:c4:08
  interfaces auto-testautovnf3-uas-orchestration
   ip-address  172.77.12.9
   mac-address fa:16:3e:de:4a:ed
 vnfc-instance ab-auto-test-vnfm3-ESC-1
  compute-host neutonoc-compute-10.localdomain
  interfaces auto-testautovnf3-uas-management
   ip-address  172.77.11.5
   mac-address fa:16:3e:f6:a0:3f
  interfaces auto-testautovnf3-uas-orchestration
   ip-address  172.77.12.5
   mac-address fa:16:3e:db:52:36
```

Example show confd-state Command Output

show confd-state

```
confd-state version 6.3.1
confd-state epoll false
confd-state daemon-status started
confd-state ha mode master
confd-state ha node-id confd-master
confd-state ha connected-slave [ a2dd5178-afae-4b3a-8b2b-910216583501 ]

                     EXPORTED
NAME                         REVISION    NAMESPACE
      PREFIX         TO ALL    EXPORTED TO
-------------------------------------------------------------------------------------------------
iana-crypt-hash              2014-08-06  urn:ietf:params:xml:ns:yang:iana-crypt-hash
      ianach         X          -
ietf-inet-types              2013-07-15  urn:ietf:params:xml:ns:yang:ietf-inet-types
      inet           X          -
ietf-netconf-acm             2012-02-22  urn:ietf:params:xml:ns:yang:ietf-netconf-acm
      nacm           X          -
ietf-netconf-monitoring      2010-10-04  urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring
     ncm            X          -
<-- SNIP -->
```

Example show confd-state ha Command Output

show confd-state ha

```
confd-state ha mode master
confd-state ha node-id confd-master
confd-state ha connected-slave [ a2dd5178-afae-4b3a-8b2b-910216583501 ]
```

Example show logs Command Output

show logs *<transaction_id>* **| display xml**

Example show running-config Command Output

show running-config

```
<-- SNIP -->
autovnf:secure-token autovnf-admin
 user     $8$YQiswhu0QLpA4N2kBo7t5eZN2uUW0L19m8WaaBzkVoc=
 password $8$mSaszfxjZ8My8Y/FqLL3Sasn1b/DmRh3pdblatq49cM=
!
autovnf:secure-token autovnf-oper
```

```
 user     $8$kTEQZ4YNdV6BcnH3ggRHJPmhk6lsh5KQFqhsQnh/KV8=
 password $8$KdTBd7ZeYuHrpdkLk5m888ckE3ZGIM7RbEMJwMwCjfo=
!
autovnf:secure-token em-login
 user     $8$jVDkSMi/W1XzkZj/qx07kEfHB9PlpPlnzCKUSjWiPXA=
 password $8$52ELrKMilGT/nad5WcPgUh7cijHiizAt8A8Tly79Q/I=
!
autovnf:secure-token confd-auth
 user     $8$bHYvP179/hlGWO8qoTnJFmm8A1HqqlREsasX+GlSAPw=
 password $8$S52APq1vb9WhLjbSPNSWiBmAmaG1tzTTmSkktKs8reo=
!
volume-catalog em-volume
 volume type LUKS
 volume size 1024
 volume bus ide
 volume bootable false
!
volume-catalog cf-boot
 volume type LUKS
 volume size 16
 volume bus ide
 volume bootable true
!
volume-catalog cf-cdr
 volume type LUKS
 volume size 200
 volume bus ide
 volume bootable false
!
autovnf:network-catalog di-internal1
 pre-created di-internal1
 type        sriov-flat
 physnet     phys_pcie1_0
 ip-prefix   192.168.1.0/24
 dhcp        true
 vlan-tag    true
 vlan        2110
<-- SNIP -->

<-- SNIP -->
autovnf:vdu-catalog control-function
 ha-type               one-to-one
 health-check-frequency 10
 health-probe-max-miss  6
 recovery-type         recovery-restart
 image location http://172.21.201.63:80/bundles/5.1.0-662/ugp-bundle/qvpc-di-cf.qcow2
 neds netconf
  ned-id         cisco-staros-nc
  port-number    830
  authentication confd-auth
 !
 volumes cf-cdr
 !
 volumes cf-boot
 !
 flavor host-aggregate   auto-test-sjc-cf-esc-mgmt1
 flavor vcpus            8
 flavor ram              16384
 flavor root-disk        6
 flavor ephemeral-disk   0
 flavor swap-disk        0
 flavor anti-affinity-placement true
 configuration staros_config.txt
  apply-at   day-zero
  source-url http://172.21.201.63:5001/configs/vnf-pkg1/files/system.cfg
<-- SNIP -->
```

Example show uas Command Output

show uas

```
uas version 5.7.0
uas state ha-active
uas ha-vip 30.30.61.101
INSTANCE IP   STATE  ROLE
-------------------------------------
30.30.62.4   alive  CONFD-MASTER
30.30.62.13  alive  NA
30.30.62.14  alive  NA
```

Note In this example, 30.30.62.4 is the confd-master and the active UAS VM.

The current version of AutoVNF software can also be seen through the USP UWS – AutoVNF User Interface under –

- the Site Overview screen (Service Deployment > Site) only if the AutoVNF configuration type is a record.
- the Auto-Vnf Configuration Overview screen only if the AutoVNF configuration type is a record.
- the UWS – AutoVNF dashboard.

Example show usp Command Output

show usp

```
<-- SNIP -->
show usp
usp uwsclock systemTime 2017-05-26T18:18:26.829Z
                        NUM
NAME                ID  DEPLOYMENTS  NAME
-----------------------------------------------
USP-GILAN-TEMPLATE  -   -
USP-VPC-TEMPLATE    -   -

usp vnfrecord 494ae7b6-c26a-4549-9212-214eb3645fef
 vnfd-name          vnfd1-deployment
 operational-status start-success
 em-state           deploying
 usp-vnf            http://172.57.11.10:2022
 em-username        admin
 em-password        $8$mSaszfxjZ8My8Y/FqLL3Sasn1b/DmRh3pdblatq49cM=
 em-mgmt-ip         172.57.11.10
 tx-id              579b4546-41a2-11e7-b3ab-fa163eccaffc
<-- SNIP -->
```

Monitoring General UAS Operations

Viewing UAS HA Logs

Logs pertaining to UAS HA are located in the following directory on the UAS VM:

/var/log/cisco-uas/ha

Log information is in the *info.log* file.

Example log:

```
2017-05-24 19:23:27,527 - Started Confd Cluster Manager.
2017-05-24 19:23:27,527 - HA Reboot policy is OFF.
2017-05-24 19:23:27,539 - Trying to acquire election lock.
2017-05-24 19:23:27,558 - Acquired election lock.
2017-05-24 19:23:27,768 - Detected zookeeper follower on this node.
2017-05-24 19:23:27,768 - Trying to become master.
2017-05-24 19:23:27,785 - Attained master state
2017-05-24 19:23:27,812 - Emitted confd-master event.
2017-05-24 19:23:27,826 - AutoVNF service started successfully
2017-05-24 19:23:27,841 - bind ha vip to ha interface successful
2017-05-24 19:23:27,851 - Error in deleting default route RTNETLINK answers: No such process

2017-05-24 19:23:27,858 - Successfully set default gateway to 172.77.11.1
2017-05-24 19:23:27,860 - Setting oper data: ha-active in confd.
2017-05-24 19:23:30,562 - Setting oper data: 172.77.11.101, 1.0.0-1 in confd.
2017-05-24 19:23:38,213 - A slave joined the cluster
```

Viewing UAS Manager Logs

Logs pertaining to UAS Manager are located in the following directory on the UAS VM:

/var/log/cisco-uas/uas-manager

Log information is in the *info.log* file.

Example log:

```
2017-05-24 19:23:27,496 - Connected to Zookeeper.
2017-05-24 19:23:27,507 - Created an ephemeral node: /172.77.12.6
```

Viewing ZooKeeper Logs

Logs pertaining to ZooKeeper are located in the following directory on the UAS VM:

/var/log/cisco-uas/zookeeper

Log information is in the *zookeeper.log* and *zookeeper.out* files.

Monitoring VNFM Operations

Note The Cisco Elastic Services Controller (ESC) is the only VNFM supported in this release.

Viewing ESC Status

ESC status can be viewed from the ESC command line or by executing a REST API from AutoVNF.

Monitoring StatusThrough the ESC Command Line

Log on to the primary ESC VM and execute the following command from the command line:

escadm status

Example command output:

```
0 ESC status=0 ESC Master Healthy
```

Monitoring Status Through an AutoVNF API

Log on to the master AutoVNF VM and execute the following command:

curl -u admin:<*password*> **-k https://**<*master_vnfm_address*>**:60000/esc/health**
Example command output:

```
{"message": "ESC services are running.", "status_code": "2000"}
```

Status code and message display information about ESC health conditions as identified in Table 3: ESC Status Code Messages, on page 58. Status codes in the 2000s imply ESC is operational, 5000 status codes imply at least one of the ESC components is not in service.

Table 3: ESC Status Code Messages

Code	Message
2000	ESC services are running
2010	ESC services are running. ESC High-Availability node not reachable.
2020	ESC services are running. One or more VIM services (keystone, nova) not reachable.*
2030	ESC services are running. VIM credentials not provided.
2040	ESC services running. VIM is configured, ESC initializing connection to VIM.
2100	ESC services are running. ESC High-Availability node not reachable. One or more VIM services (nova) not reachable
5010	ESC service ESC_MANAGER not running.
5020	ESC service CONFD not running.
5030	ESC service MONA not running.
5040	ESC service VIM_MANAGER not running.
5090	More than one ESC service (confd, mona) not running.**

Viewing ESC Health

ESC health can be viewed by logging on to the primary ESC VM and executing the following command from the command line:

health.sh

Example command output:

```
esc ui is disabled -- skipping status check
esc_monitor start/running, process 840
esc_mona is up and running ...
vimmanager start/running, process 2807

vimmanager start/running, process 2807
esc_confd is started
tomcat6 (pid 2973) is running...                           [  OK  ]
postgresql-9.4 (pid  2726) is running...
ESC service is running...
Active VIM = OPENSTACK
ESC Operation Mode=OPERATION

/opt/cisco/esc/esc_database is a mountpoint
============== ESC HA (MASTER) with DRBD =================
DRBD_ROLE_CHECK=0
MNT_ESC_DATABSE_CHECK=0
VIMMANAGER_RET=0
ESC_CHECK=0
STORAGE_CHECK=0
ESC_SERVICE_RET=0
MONA_RET=0
ESC_MONITOR_RET=0
=======================================
ESC HEALTH PASSED
```

Viewing ESC Logs

ESC logs are available on the VNFM VM in the following directory:

/var/log/esc/

Two levels of logs are available for ESC:

- ESC Logs, on page 59
- ESC YANG Logs, on page 61

Refer also to the ESC user documentation for additional information on monitoring and maintaining the software.

ESC Logs

To collect ESC logs:

1. Log on to the primary VNFM VM.
2. Navigate to the scripts directory.

   cd /opt/cisco/esc/esc-scripts
3. Launch the *collect-esc-logs.sh* script to collect the logs.

   sudo ./collect-esc-logs.sh

Example log output:

```
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
```

```
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for admin: Creating log tarball:
/var/tmp/esc_log-2017-05-25_18.09.31_UTC.tar.bz2
Creating temporary working directory: /var/tmp/esc_log-2017-05-25_18.09.31_UTC

Dumping thread status of ESCManager from tomcat pid 2973 to catalina.out
escadm-output.txt
vm_info.txt
esc_version.txt
esc/
esc/vimmanager/
esc/vimmanager/operations_vimmanager.log
esc/vimmanager/vimmanager.log
esc/esc_gc.log.2.current
esc/esc_gc.log.0
esc/escmanager.log
esc/event_escmanager.log
esc/escmanager_tagged.log
esc/esc_gc.log.1
esc/custom_script/
esc/pgstartup.log
esc/mona/
esc/mona/actions_mona.log
esc/mona/mona_gc.log.0.current
esc/mona/rules_mona.log
esc/mona/mona.log
tar: esc/mona/mona.log: file changed as we read it
esc/confd/
esc/confd/global.data
esc/confd/devel.log
esc/confd/confd.log
esc/confd/browser.log
esc/confd/audit.log
esc/confd/netconf.trace
esc/confd/netconf.log
esc/spy.log
esc/error_escmanager.log
esc/esc_monitor.log
esc/esc_haagent.log
esc/yangesc.log
esc/debug_yangesc.log
esc/esc_confd.log
boot.log
secure
messages
dmesg
tomcat6/
tomcat6/localhost.2017-05-24.log
tomcat6/host-manager.2017-05-24.log
tomcat6/manager.2017-05-24.log
tomcat6/catalina.out
tomcat6/catalina.2017-05-24.log
audit/
audit/audit.log
postgresql/data/pg_log/
postgresql/data/pg_log/postgresql-Thu.log
postgresql/data/pg_log/postgresql-Wed.log
esc-config/esc-config.xml
Warning: tar completed with status: 1

Tarball file:  /var/tmp/esc_log-2017-05-25_18.09.31_UTC.tar.bz2
Symbolic link: /tmp/esc_log-2017-05-25_18.09.31_UTC.tar.bz2

Suggestions:
 1. Transfer the tarball file from the esc vm
 2. Remove the tarball and symbolic link (to save ESC disk space):
     sudo rm /var/tmp/esc_log-2017-05-25_18.09.31_UTC.tar.bz2
     sudo rm /tmp/esc_log-2017-05-25_18.09.31_UTC.tar.bz2
 3. Command to list contents of tarball:
     tar jtvf esc_log-2017-05-25_18.09.31_UTC.tar.bz2
```

```
4. Command to extract from the tarball:
    tar jxf esc_log-2017-05-25_18.09.31_UTC.tar.bz2
```

ESC YANG Logs

ESC YANG logs are stored in the following file:

/var/log/esc/yangesc.log

Monitoring VNF Operations

Viewing UEM Service Status

1. Log on to the master UEM VM as the user *ubuntu*.
2. Access the NCS CLI.

   /opt/cisco/usp/packages/nso/ncs-4.1.1/bin/ncs_cli -C -u admin
3. Check the NCS state.

   show ncs-state ha
   Example command output:

   ```
   ncs-state ha mode master
   ncs-state ha node-id 3-1501714180
   ncs-state ha connected-slave [ 4-1501714262 ]
   ```
4. Display the health of cluster.

   show ems
   Example command output:

   ```
   EM            VNFM
   ID  SLA  SCM  PROXY
   --------------------
   3   UP   UP   UP
   4   UP   UP   UP
   ```

Viewing UEM Logs

To collect UEM logs:

1. Navigate to the *scripts* directory.

   cd /opt/cisco/em-scripts
2. Launch the *collect-em-logs.sh* script to collect the logs.

   sudo ./collect-em-logs.sh

   Example log output:

   ```
   Collecting Zookeeper nodes...
   Traceback (most recent call last):
     File "/opt/cisco/em-scripts/zk_dump.py", line 2, in <module>
       from kazoo.client import KazooClient
   ImportError: No module named kazoo.client
   ```

```
Creating log tarball em-logs-2017-05-26_00.37.28_UTC.tar.bz2 ...
em-logs/
em-logs/upstart/
em-logs/upstart/proxy.log
em-logs/upstart/zk.log
em-logs/upstart/ncs.log
em-logs/scm/
em-logs/scm/audit.log.1.gz
em-logs/scm/ncserr.log.1
em-logs/scm/ncs-java-vm.log.2.gz
em-logs/scm/xpath.trace.1.gz
em-logs/scm/ncs-java-vm.log.1.gz
em-logs/scm/xpath.trace.2.gz
em-logs/scm/ncs-java-vm.log
em-logs/scm/ncserr.log.siz
em-logs/scm/xpath.trace
em-logs/scm/audit.log
em-logs/scm/audit.log.2.gz
em-logs/scm/ncserr.log.idx
em-logs/sla/
em-logs/sla/sla-mgr.log
em-logs/sla/sla-system.log
em-logs/zookeeper/
em-logs/zookeeper/zookeeper.out
em-logs/zookeeper/zookeeper.log
em-logs/vnfm-proxy/
em-logs/vnfm-proxy/vnfm-proxy.log

=============== Tarball available at: /tmp/em-logs-2017-05-26_00.37.28_UTC.tar.bz2
===============

To extract the tarball, run: "tar jxf /tmp/em-logs-2017-05-26_00.37.28_UTC.tar.bz2"
```

Viewing UEM Zookeeper Logs

The UEM maintains logs on the Zookeeper process. The logs are located in the following directories:

```
/var/log/em/zookeeper/zookeeper.log
/var/log/em/zookeeper/zookeeper.out
```

Viewing VNF Information through the Control Function

Information on the VNF deployment can be obtained by executing commands on the Control Function (CF) VNFC. To access the CF CLI:

1. Open an SSH connection to the IP address of the management interface associated with CF1.
2. Press Enter to bring up the log in prompt.
3. Enter the username and password.
4. At the Exec mode prompt, enter each of the following commands and observe the results to ensure that the VNF components have been properly deployed according to the desired configuration:

Command	Purpose
show card table	Displays all VM types (e.g. CF, SF, NF, and AF) that have been deployed.

Command	Purpose
show crash list	Displays software crash events records and associated dump files (minicore, NPU or kernel) for all crashes or a specified crash event. Verify that there are no new or unexpected crashes listed.
show emctrl vdu list	Displays card to VM mappings for the VNF. Each card should have a valid universally unique identifier (UUID).
show rct stats	Displays statistics associated with Recovery Control Task (RCT) events, including migrations, switchovers and shutdowns. RCT statistics are associated with card-to-card session recovery activities.
show session progress	Displays session progress information for the current context filtered by the options specified. Check for any active or new calls before proceeding with a deactivation.
show version verbose	Displays the software version that has been deployed.
show vdu summary	Displays general information pertaining to the virtual descriptor units (VDUs) that have been deployed.
show usf vdu all	Displays detailed information for the VDUs that have been deployed for the USF VDU.
show usf vdu-group all	Displays information for VDU groups pertaining to the USF VNF use case (if deployed).
show usf network-path all	Displays network path information for USF VNF components (if deployed).
show usf service-function-chain all	Displays SFC information for the USF VNF (if deployed).

Troubleshooting Deactivation Process and Issues

NOTES:

- The deactivate process is idempotent and can be multiple times and without error. The system will retry to remove any resources that remain.
- If a deactivation fails (a transaction failure occurs), look at the logs on various UAS software components (AutoDeploy, AutoIT-VNF, and AutoVNF), VNFM (ESC), and UEM.
- If deactivation has failed, you must ensure that a clean up is performed either using automation tools or manually if necessary.
- Activation must not be reattempted until all of the previous artifacts have been removed.

Deactivation Fails Due to Communication Errors with AutoVNF

Problem Description

During the AutoVNF deactivation process, AutoDeply indicates that it is unable to deactivate the AutoVNF. This is observed through:

- AutoDeploy transaction log
- AutoDeploy upstart log

Possible Cause(s)

- AutoDeploy is not able to communicate with AutoVNF.

Action(s) to Take

- Check network connectivity between the AutoDeploy VM and the AutoVNF VIP.
- Check the management and orchestration network.
- Address any connectivity issues.

Next Steps

- Once connectivity issues are addressed, perform the deactivate procedure again.

Deactivation Fails Because AutoDeploy Generates an Exception

Problem Description

AutoDeploy generates an exception error during the deactivation process.

Possible Cause(s)

- Connectivity issues
- Configuration issues
- OpenStack/VIM specific issues
- Hardware issues

Action(s) to Take

1. Capture logs from */var/log/upstart/autodeploy.log* along with exception error message.
2. Log on to AutoIT-VNF and collect the logs from */var/log/cisco/usp/auto-it/autoit.log* along with the exception message, if any.
3. Log on to VIP of the active (master) AutoVNF VM and perform a cleanup by running the **deactivate** command from there.
   a. Log on to the AutoVNF VM as the default user, *ubuntu*.
   b. Switch to the root user.

      sudo su

c Enter the ConfD CLI.

confd_cli -C -u admin

d Deactivate the deployment.

autovnf:deactivate-deployment deployment-name <*deployment_name*>

4 Check the last transaction log to verify that the deactivation was successful. (Transactions are auto-sorted by timestamp, so it should be the last one in the list.)

Example commands and outputs:

show transactions

```
TX ID              TX TYPE         ID          TIMESTAMP                          STATUS
              DETAIL
-------------------------------------------------------------------------------------------------------
1500605583-055162  vnf-deployment  dep-5-5  2017-07-21T02:53:03.055205-00:00
deployment-failed   -
1500606090-581863  vnf-deployment  dep-5-5  2017-07-21T03:01:30.581892-00:00
deployment-success  -
1500606127-221084  vnf-deployment  dep-5-5  2017-07-21T03:02:07.221114-00:00
deployment-success  -

show log 1500606127-221084 | display xml
<config xmlns="http://tail-f.com/ns/config/1.0">
  <log xmlns="http://www.cisco.com/usp/nfv/usp-autovnf-oper">
    <tx-id>1500606127-221084</tx-id>
    <log>2017-07-21 03:02:07,276 - Notify deployment
2017-07-21 03:02:07,297 - Connection to VNFM (esc) at 172.16.181.107
2017-07-21 03:02:07,418 - NETConf Sessions (Transaction/Notifications) estabilished
...
```

5 Manually delete the AutoDeploy VM using the information in Terminating the AutoDeploy VM, on page 3.

Next Steps

- Open a support case providing all of the log information that was collected.

Deactivation Fails Because of AutoVNF-VNFM Communication Issues

Problem Description

During the AutoVNF deactivation process, AutoVNF indicates that it is unable to deactivate the VNFM. This is observed through:

- AutoVNF transaction log
- AutoVNF upstart log

Possible Cause(s)

- AutoVNF is not able to communicate with the VNFM.

Action(s) to Take

- Check network connectivity between the master AutoVNF VM and the VNFM VIP.
- Check the management and orchestration network.
- Address any connectivity issues.

Next Steps

- Once connectivity issues are addressed, perform the deactivate procedure again.

Deactivation Fails Because of Issue at VNFM

Problem Description

During the AutoVNF deactivation process, the VNFM returns an error. his is observed through:

- AutoVNF transaction log
- AutoVNF upstart log
- ESC logs

Possible Cause(s)

- ESC health is not good due to a bug or network connectivity.
- ESC is not able to communicate with the VIM.
- ESC has an internal error.
- AutoVNF is unable to create/delete OpenStack artifacts.

Action(s) to Take

1. Check */var/log/esc/yangesc.log* for any issues or error messages.
2. Run **health.sh** to determine the health of ESC.
3. Check network connectivity and address an issues. Retry the deactivation.
4. Check network connectivity with the VIM and address any issues. Retry the deactivation.
5. Determine if ESC has a deployment configuration. From the active ESC VM:

   /opt/cisco/esc/confd/bin/confd_cli -C
   show running-config
   If a configuration is present, most likely ESC is still retrying the deactivation, allow more time for the process to continue.

   If no configuration exists, check if there are deployment artifacts still on the VIM. Retry the deactivation.
6. Collect logs by running *collect_esc_log.sh* from both the active and standby ESC VMs.
7. Perform a manual cleanup.

Note Only artifacts which UAS created need to be removed. Any pre-created artifacts must remain in place.

   a. Login on to the VIM as tenant.
   b. Remove all VMs.
   c. Remove all VIP Ports.
   d. Remove all networks.

e Remove all flavors.

f Remove all volumes.

g Remove all images.

h Remove host-aggregate created as part of automation.

Next Steps

- Open a support case providing all of the log information that was collected.

Deactivation Fails Because AutoVNF Generates an Exception

Problem Description

AutoVNF generates an exception error during the deactivation process.

Possible Cause(s)

- Connectivity issues
- Configuration issues
- OpenStack/VIM specific issues
- Hardware issues

Action(s) to Take

1 Collect all logs from */var/log/cisco-uas*.

2 Perform a manual cleanup.

Note Only artifacts which UAS created need to be removed. Any pre-created artifacts can remain in place.

a Login on to the VIM as tenant.

b Remove all VMs.

c Remove all VIP Ports.

d Remove all networks.

e Remove all flavors.

f Remove all volumes.

g Remove all images.

h Remove host-aggregate created as part of automation.

Next Steps

- Open a support case providing all of the log information that was collected.

Troubleshooting UEM Issues

This section contains information on troubleshooting UEM issues.

UEM VM Stuck in a Boot Loop

Problem Description

Processes that normally run on the UEM VM are unable to start and the VM is stuck in a boot-loop.

Possible Cause(s)

There is an error with the Zookeeper database keeping the Zookeeper process and other UEM processes from starting. (No other UEM process can be started unless the Zookeeper process has started.)

Action(s) to Take

1. Check the UEM Zookeeper logs. Refer to Viewing UEM Zookeeper Logs, on page 62.
2. Look for error messages similar to the following:

   ```
   [myid:4] - INFO  [main:FileSnap@83] - Reading snapshot
   /var/lib/zookeeper/data/version-2/snapshot.5000004ba
   [myid:4] - ERROR [main:QuorumPeer@557] - Unable to load database on disk
   java.io.EOFException
   ```

   If the above errors exist, proceed to the next step. If not, further debugging is required. Please contact your local support representative.
3. Rebuild the Zookeeper database.
   a. Check the health of Master and Slave EM instance. Execute the following commands on each instance.

   Master UEM VM:

   sudo -i
   ncs_cli -u admin -C

   ```
   admin connected from 127.0.0.1 using console on deploymentem-1
   ```

   show ems

   ```
   EM              VNFM
   ID  SLA  SCM  PROXY  VERSION
   ------------------------------
   3   UP   UP   UP     5.7.0
   6   UP   UP   UP     5.7.0
   ```

   exit

Important Only the master UEM status may be displayed in the above command because the slave UEM is in the the boot loop.

show ncs-state ha

```
ncs-state ha mode master
ncs-state ha node-id 6-1506059686
ncs-state ha connected-slave [ 3-1506059622 ]
```

Slave UEM VM:

Important The slave UEM may not be accessible if it is experiencing the boot loop issue.

sudo -i
ncs_cli -u admin -C
```
admin connected from 127.0.0.1 using console on deploymentem-1
```

show ems
```
EM            VNFM
ID  SLA  SCM  PROXY  VERSION
-----------------------------
3   UP   UP   UP     5.7.0
6   UP   UP   UP     5.7.0
```

exit
show ncs-state ha
```
ncs-state ha mode slave
ncs-state ha node-id 3-1506059622
ncs-state ha master-node-id 6-1506059686
```

b Login to the node on which Zookeeper data is corrupted.

c Enable the debug mode.

/opt/cisco/em-scripts/enable_debug_mode.sh
```
Disable EM reboot. Enable debug mode
```

d Reboot the VM in order to enter the debug mode.

e Remove the corrupted data.

cd /var/lib/zookeeper/data/
ls
```
myid  version-2  zookeeper_server.pid
```

mv version-2 version-2_old

Important This process removes the Zookeeper database by renaming it for additional debugging/recovery.

f Reboot the node instance for it to reconcile and rebuild the Zookeeper database from a healthy UEM instance.

reboot

g Login to the UEM VM upon reboot.

h Validate that the database has been successfully rebuilt on the previously failing UEM node.

sudo -i
ncs_cli -u admin -C
```
admin connected from 127.0.0.1 using console on aselvanavnfddeploymentem-0
```

show ems
```
EM            VNFM
ID  SLA  SCM  PROXY  VERSION
-----------------------------
3   UP   UP   UP     5.7.0
6   UP   UP   UP     5.7.0
```

show ncs-state ha
```
ncs-state ha mode slave
ncs-state ha node-id 3-1506093933
ncs-state ha master-node-id 6-1506093930
```

exit

cd /var/lib/zookeeper/data/
ls
```
myid  version-2  version-2_old  zookeeper_server.pid
```

cat /var/log/em/zookeeper/zookeeper.log
```
<---SNIP--->
2017-09-22 15:25:35,192 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Follower@61] - FOLLOWING - LEADER ELECTION
TOOK - 236
2017-09-22 15:25:35,194 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:QuorumPeer$QuorumServer@149] - Resolved
hostname: 30.30.62.6 to address: /30.30.62.6
2017-09-22 15:25:35,211 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Learner@329] - Getting a snapshot from leader
2017-09-22 15:25:35,224 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:FileTxnSnapLog@240] - Snapshotting: 0x200000050
 to /var/lib/zookeeper/data/version-2/snapshot.200000050
2017-09-22 15:25:37,561 [myid:3] - INFO
[NIOServerCxn.Factory:0.0.0.0/0.0.0.0:2181:NIOServerCnxnFactory@192] - Accepted socket
 connection from /30.30.62.15:58011
2017-09-22 15:25:37,650 [myid:3] - WARN
[NIOServerCxn.Factory:0.0.0.0/0.0.0.0:2181:ZooKeeperServer@882] - Connection request
 from old client /30.30.62.15:58011; will be dropped if server is in r-o mode
2017-09-22 15:25:37,652 [myid:3] - INFO
[NIOServerCxn.Factory:0.0.0.0/0.0.0.0:2181:ZooKeeperServer@928] - Client attempting
to establish new session at /30.30.62.15:58011
<---SNIP--->
```

i Disable the UEM debug mode on the VM on which the Zookeeper database was rebuilt.

/opt/cisco/em-scripts/disable_debug_mode.sh
```
Disable debug mode
```

Next Steps

Open a support case providing all of the log information that was collected.