各種Ultra-M元件的備份和恢復過程

目錄

簡介

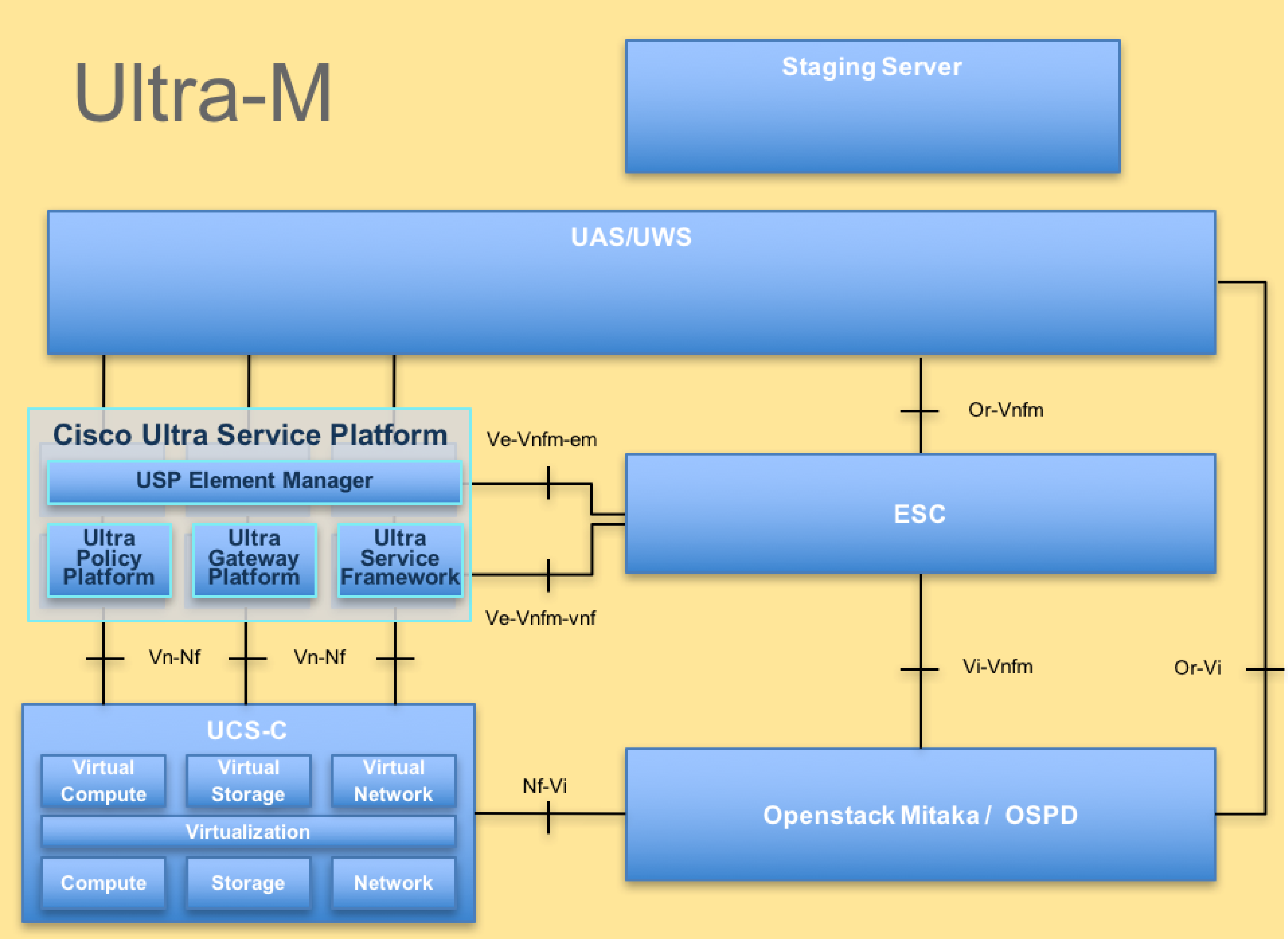
本文檔介紹在託管StarOS虛擬網路功能(VNF)的Ultra-M設定中備份和還原虛擬機器(VM)所需的步驟。

背景資訊

Ultra-M是經過預先打包和驗證的虛擬化移動資料包核心解決方案，旨在簡化VNF的部署。Ultra-M解決方案由以下虛擬機器(VM)型別組成：

- 自動IT
- 自動部署
- Ultra自動化服務(UAS)
- 元素管理器(EM)
- 彈性服務控制器(ESC)
- 控制功能(CF)
- 作業階段功能(SF)

Ultra-M的高級體系結構及涉及的元件如下圖所示：



本文檔適用於熟悉Cisco Ultra-M平台的思科人員。

附註：Ultra M 5.1.x版本用於定義本文檔中的過程。

縮寫

VNF	虛擬網路功能
CF	控制功能
SF	服務功能
ESC	彈性服務控制器
澳門幣	程式方法
OSD	對象儲存磁碟
硬碟	硬碟驅動器
固態硬碟	固態驅動器
VIM	虛擬基礎架構管理員
虛擬機器	虛擬機器
EM	元素管理器
UAS	Ultra自動化服務
UUID	通用唯一識別符號

備份過程

OSPD備份

1.檢查OpenStack堆疊和節點清單的狀態。

```
[stack@director ~]$ source stackrc
[stack@director ~]$ openstack stack list --nested
[stack@director ~]$ ironic node-list
[stack@director ~]$ nova list
```

2.從OSP-D節點檢查所有Undercloud服務是否處於已載入、活動和運行狀態。

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

UNIT                                        LOAD   ACTIVE SUB     DESCRIPTION

neutron-dhcp-agent.service                  loaded active running OpenStack Neutron DHCP 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-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-heat-api-cfn.service              loaded active running Openstack Heat CFN-compatible
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-ironic-api.service                loaded active running OpenStack Ironic API service
openstack-ironic-conductor.service          loaded active running OpenStack Ironic Conductor
service
openstack-ironic-inspector-dnsmasq.service loaded active running PXE boot dnsmasq service for
Ironic Inspector
openstack-ironic-inspector.service          loaded active running Hardware introspection service
for OpenStack Ironic
openstack-mistral-api.service               loaded active running Mistral API Server
openstack-mistral-engine.service            loaded active running Mistral Engine Server
openstack-mistral-executor.service          loaded active running Mistral Executor Server
openstack-nova-api.service                  loaded active running OpenStack Nova API Server
openstack-nova-cert.service                 loaded active running OpenStack Nova Cert Server
openstack-nova-compute.service              loaded active running OpenStack Nova Compute Server
openstack-nova-conductor.service            loaded active running OpenStack Nova Conductor Server
openstack-nova-scheduler.service            loaded active running OpenStack Nova Scheduler Server
openstack-swift-account-reaper.service      loaded active running OpenStack Object Storage
(swift) - Account Reaper
openstack-swift-account.service             loaded active running OpenStack Object Storage
(swift) - Account Server
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-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
openstack-zaqar.service                    loaded active running OpenStack Message Queuing
Service (code-named Zaqar) Server
openstack-zaqar@1.service                  loaded active running OpenStack Message Queuing
Service (code-named Zaqar) Server Instance 1
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.

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

3.在執行備份過程之前，請確認有足夠的可用磁碟空間。此彈珠應至少為3.5 GB。

```
[stack@director ~]$df -h
```

4.以根使用者身份執行這些命令，以便將資料從Undercloud節點備份到名為**undercloud-backup-[timestamp].tar.gz**的檔案中，然後將其傳輸到備份伺服器。

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

自動部署備份

1. AutoDeploy要求備份此資料：

- 自動部署配置數據庫(CDB)
- 自動部署配置
- 自動部署日誌
- 系統日誌配置

2.每次啟用/取消啟用後，都需要備份AutoDeploy Confd CDB資料和運行配置，並確保將資料傳輸到備份伺服器。

3.自動部署在獨立模式下執行，如果資料丟失，您將無法正常停用部署。因此，必須備份配置和CDB資料。

```
ubuntu@auto-deploy-iso-2007-uas-0:~$ sudo -i
root@auto-deploy-iso-2007-uas-0:~# service uas-confd stop
uas-confd stop/waiting

root@auto-deploy:/home/ubuntu# service autodeploy status
autodeploy start/running, process 1313

root@auto-deploy:/home/ubuntu# service autodeploy stop
autodeploy stop/waiting
```

```
root@auto-deploy-iso-2007-uas-0:~# cd /opt/cisco/usp/uas/confd-6.3.1/var/confd
root@auto-deploy-iso-2007-uas-0:/opt/cisco/usp/uas/confd-6.3.1/var/confd# tar cvf
autodeploy_cdb_backup.tar cdb/

cdb/
cdb/O.cdb
cdb/C.cdb
cdb/aaa_init.xml
cdb/A.cdb
```

4.將**autodeploy_cdb_backup.tar**複製到備份伺服器。

5.備份自動部署中的運行配置，並將其傳輸到備份伺服器。

```
root@auto-deploy:/home/ubuntu# confd_cli -u admin -C

Welcome to the ConfD CLI

admin connected from 127.0.0.1 using console on auto-deploy

auto-deploy#show running-config | save backup-config-$date.cfg à Replace the $date to
appropriate date and POD reference
auto-deploy#
```

6.啟動自動部署配置服務。

```
root@auto-deploy-iso-2007-uas-0:~# service uas-confd start
uas-confd start/running, process 13852


root@auto-deploy:/home/ubuntu# service autodeploy start
autodeploy start/running, process 8835
```

*7.*導航到指令碼目錄並從AutoDeploy VM收集日誌。

```
cd /opt/cisco/usp/uas/scripts
```

8.啟動**collect-uas-logs.sh**指令碼以收集日誌。

```
sudo ./collect-uas-logs.sh
```

9.從AutoDeploy進行ISO映像備份，並將其傳輸到備份伺服器。

```
root@POD1-5-1-7-2034-auto-deploy-uas-0:/home/ubuntu# /home/ubuntu/isos
root@POD1-5-1-7-2034-auto-deploy-uas-0:/home/ubuntu/isos# ll

total 4430888
drwxr-xr-x 2 root   root          4096 Dec 20 01:17 ./
drwxr-xr-x 5 ubuntu ubuntu        4096 Dec 20 02:31 ../
-rwxr-xr-x 1 ubuntu ubuntu 4537214976 Oct 12 03:34 usp-5_1_7-2034.iso*
```

10.收集系統日誌配置並將其儲存在備份伺服器上。

```
ubuntu@auto-deploy-vnf-iso-5-1-5-1196-uas-0:~$sudo su
root@auto-deploy-vnf-iso-5-1-5-1196-uas-0:/home/ubuntu#ls /etc/rsyslog.d/00-autodeploy.conf
00-autodeploy.conf
root@auto-deploy-vnf-iso-5-1-5-1196-uas-0:/home/ubuntu#ls /etc/rsyslog.conf
rsyslog.conf
```

AutoIT-VNF備份

AutoIT-VNF是無狀態虛擬機器，因此沒有需要備份的資料庫(DB)。AutoIT-VNF負責與Ultra-M的配置管理儲存庫一起執行軟體包管理，因此，執行這些備份至關重要。

1.備份第0天StarOS配置，並將其傳輸到備份伺服器。

```
root@auto-it-vnf-iso-5-8-uas-0:/home/ubuntu# cd /opt/cisco/usp/uploads/
root@auto-it-vnf-iso-5-8-uas-0:/opt/cisco/usp/uploads# ll
total 12
drwxrwxr-x  2 uspadmin usp-data 4096 Nov  8 23:28 ./
drwxr-xr-x 15 root     root     4096 Nov  8 23:53 ../
-rw-rw-r--  1 ubuntu   ubuntu    985 Nov  8 23:28 system.cfg
```

2.導航到scripts目錄並從AutoIT VM收集日誌。

```
cd /opt/cisco/usp/uas/scripts
```

3.啟動**collect-uas-logs.sh**指令碼以收集日誌。

```
sudo ./collect-uas-logs.sh
```

4.收集系統日誌配置備份並將其儲存在備份伺服器中。

```
ubuntu@auto-it-vnf-iso-5-1-5-1196-uas-0:~$sudo su

root@auto-it-vnf-iso-5-1-5-1196-uas-0:/home/ubuntu#ls /etc/rsyslog.d/00-autoit-vnf.conf

00-autoit-vnf.conf

root@auto-it-vnf-iso-5-1-5-1196-uas-0:ls /etc/rsyslog.conf

rsyslog.conf
```

AutoVNF備份

AutoVNF負責啟動單個VNFM和VNF。AutoDeploy將例項化VNFM和VNF所需的配置傳送到AutoVNF，AutoVNF執行此操作。為了啟動VNFM，AutoVNF將直接與VIM/OpenStack通訊，在VNFM啟動後，AutoVNF使用VNFM啟動VNF。

AutoVNF有1:N冗餘，在Ultra-M設定中，有三台AutoVNF虛擬機器正在運行。Ultra-M支援單個AutoVNF故障，並且可以進行恢復。

> **附註**：如果出現多個故障，則不受支援，可能需要重新部署系統。

AutoVNF備份詳細資訊：

- 運行配置
- ConfD CDB DB
- AutoVNF日誌（來自AutoVNF的每個例項）
- 系統日誌配置

建議在給定站點上執行任何啟用/取消啟用並上傳到備份伺服器之前進行備份。

1.登入到主AutoVNF並驗證其是否為**confd-master**。

```
root@auto-testautovnf1-uas-1:/home/ubuntu# confd_cli -u admin -C

Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on auto-testautovnf1-uas-1
auto-testautovnf1-uas-1#show uas
uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.57.11.101

INSTANCE IP   STATE  ROLE
-----------------------------------
172.57.12.6   alive  CONFD-SLAVE
172.57.12.7   alive  CONFD-MASTER
172.57.12.13  alive  NA

auto-testautovnf1-uas-1#exit

root@auto-testautovnf1-uas-1:/home/ubuntu# ip a

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
      valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host
      valid_lft forever preferred_lft forever

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen
1000
   link/ether fa:16:3e:c7:dc:89 brd ff:ff:ff:ff:ff:ff
   inet 172.57.12.7/24 brd 172.57.12.255 scope global eth0
      valid_lft forever preferred_lft forever
   inet6 fe80::f816:3eff:fec7:dc89/64 scope link
      valid_lft forever preferred_lft forever

3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen
1000
   link/ether fa:16:3e:10:29:1b brd ff:ff:ff:ff:ff:ff
   inet 172.57.11.101/24 brd 172.57.11.255 scope global eth1
      valid_lft forever preferred_lft forever
   inet6 fe80::f816:3eff:fe10:291b/64 scope link
      valid_lft forever preferred_lft forever
```

2.備份運行配置，並將檔案傳輸到備份伺服器。

```
root@auto-testautovnf1-uas-1:/home/ubuntu# confd_cli -u admin -C
Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on auto-testautovnf1-uas-1
auto-testautovnf1-uas-1#show running-config | save running-autovnf-12202017.cfg
auto-testautovnf1-uas-1#exit
root@auto-testautovnf1-uas-1:/home/ubuntu# ll running-autovnf-12202017.cfg
-rw-r--r-- 1 root   root   18181 Dec 20 19:03 running-autovnf-12202017.cfg
```

3.備份CDB並將檔案傳輸到備份伺服器。

```
root@auto-testautovnf1-uas-1:/opt/cisco/usp/uas/confd-6.3.1/var/confd# tar cvf
autovnf_cdb_backup.tar cdb/
cdb/
cdb/O.cdb
cdb/C.cdb
cdb/aaa_init.xml
cdb/vpc.xml
cdb/A.cdb
cdb/gilan.xml

root@auto-testautovnf1-uas-1:/opt/cisco/usp/uas/confd-6.3.1/var/confd#
root@auto-testautovnf1-uas-1:/opt/cisco/usp/uas/confd-6.3.1/var/confd# ll autovnf_cdb_backup.tar
-rw-r--r-- 1 root root 1198080 Dec 20 19:08 autovnf_cdb_backup.tar
```

4.導航到scripts目錄，收集日誌並傳輸到備份伺服器。

```
cd /opt/cisco/usp/uas/scripts

sudo ./collect-uas-logs.sh
```

5.登入到AutoVNF的備用例項並執行這些步驟，以收集日誌並將其傳輸到備份伺服器。

6.在主用和備用AutoVNF虛擬機器上備份系統日誌配置，並將其傳輸到備份伺服器。

```
ubuntu@auto-testautovnf1-uas-1:~$sudo su
root@auto-testautovnf1-uas-1:/home/ubuntu#ls /etc/rsyslog.d/00-autovnf.conf
00-autovnf.conf

root@auto-testautovnf1-uas-1:/home/ubuntu#ls /etc/rsyslog.conf
rsyslog.conf
```

ESC備份

1. AutoVNF負責通過直接與VIM互動在Ultra-M解決方案中啟動ESC。AutoVNF/EM將VNF特定配置傳遞到ESC，而ESC則通過與VIM互動來啟動VNF。

2. Ultra-M解決方案中的ESC有1:1冗餘。部署了兩個ESC虛擬機器，它們支援Ultra-M中的單個故障。也就是說，如果系統中存在單個故障，您可以恢復系統。

附註：如果出現多個故障，則不受支援，可能需要重新部署系統。

ESC備份詳細資訊：

- 運行配置
- ConfD CDB DB
- ESC日誌
- 系統日誌配置

3. ESC資料庫備份的頻率很棘手，在ESC監視和維護所部署的各種VNF虛擬機器的各種狀態機時需要仔細處理。建議您在指定的VNF/POD/站點中執行活動後執行這些備份。

4.使用**health.sh**指令碼驗證ESC的運行狀況是否良好。

```
[root@auto-test-vnfm1-esc-0 admin]# escadm status
```

```
0 ESC status=0 ESC Master Healthy

[root@auto-test-vnfm1-esc-0 admin]# health.sh
esc ui is disabled -- skipping status check
esc_monitor start/running, process 836
esc_mona is up and running ...
vimmanager start/running, process 2741
vimmanager start/running, process 2741
esc_confd is started
tomcat6 (pid 2907) is running...                           [  OK  ]
postgresql-9.4 (pid  2660) 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
```

5.備份運行配置，並將檔案傳輸到備份伺服器。

```
[root@auto-test-vnfm1-esc-0 admin]# /opt/cisco/esc/confd/bin/confd_cli -u admin -C

admin connected from 127.0.0.1 using console on auto-test-vnfm1-esc-0.novalocal
auto-test-vnfm1-esc-0# show running-config | save /tmp/running-esc-12202017.cfg
auto-test-vnfm1-esc-0#exit

[root@auto-test-vnfm1-esc-0 admin]# ll /tmp/running-esc-12202017.cfg
-rw-------. 1 tomcat tomcat 25569 Dec 20 21:37 /tmp/running-esc-12202017.cfg
```

備份資料庫

1.將ESC設定為維護模式。

2.登入到ESC VM並在進行備份之前執行此命令。

```
[admin@auto-test-vnfm1-esc-0 admin]# sudo bash
[root@auto-test-vnfm1-esc-0 admin]# cp /opt/cisco/esc/esc-scripts/esc_dbtool.py
/opt/cisco/esc/esc-scripts/esc_dbtool.py.bkup
[root@auto-test-vnfm1-esc-0 admin]# sudo sed -i "s,'pg_dump,'/usr/pgsql-9.4/bin/pg_dump,"
/opt/cisco/esc/esc-scripts/esc_dbtool.py

#Set ESC to mainenance mode
[root@auto-test-vnfm1-esc-0 admin]# escadm op_mode set --mode=maintenance
```

3.檢查ESC模式並確保它處於維護模式。

```
[root@auto-test-vnfm1-esc-0 admin]# escadm op_mode show
```

4.使用ESC中提供的資料庫備份還原工具備份資料庫。

```
[root@auto-test-vnfm1-esc-0 admin]# sudo /opt/cisco/esc/esc-scripts/esc_dbtool.py backup --file
 scp://<username>:<password>@<backup_vm_ip>:<filename>
```

5.將ESC設定回「操作模式」並確認模式。

```
[root@auto-test-vnfm1-esc-0 admin]# escadm op_mode set --mode=operation

[root@auto-test-vnfm1-esc-0 admin]# escadm op_mode show
```

6.導航到指令碼目錄並收集日誌。

```
[root@auto-test-vnfm1-esc-0 admin]# /opt/cisco/esc/esc-scripts

sudo ./collect_esc_log.sh
```

7.在備用ESC虛擬機器上重複相同步驟，並將日誌傳輸到備份伺服器。

8.收集兩個ESC VM上的系統日誌配置備份，並將其傳輸到備份伺服器。

```
[admin@auto-test-vnfm2-esc-1 ~]$ cd /etc/rsyslog.d
[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.d/00-escmanager.conf
00-escmanager.conf

[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.d/01-messages.conf
01-messages.conf

[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.d/02-mona.conf
02-mona.conf

[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.conf
rsyslog.conf
```

EM備份

1. VNFM/ESC啟動後，AutoVNF使用ESC啟動EM群集。EM集群啟動後，EM將與ESC進行互動以啟動VNF(VPC/StarOS)。

2. EM在Ultra-M解決方案中具有1:N冗餘。有一個包含三個EM虛擬機器的群集，Ultra-M支援恢復單個虛擬機器故障。

> **附註**：如果出現多個故障，則不受支援，可能需要重新部署系統。

EM備份詳細資訊：

- 運行配置
- NCS DB
- EM日誌（來自NCS主用/備用虛擬機器）
- 系統日誌配置

3. EM DB備份的頻率非常棘手，在ESC監控和維護所部署的各種VNF VM的各種狀態機時需要仔細處理。建議您在執行給定VNF/POD/站點中的活動之後執行這些備份。

4.備份EM運行配置並將檔案傳輸到備份伺服器。

```
ubuntu@vnfd1deploymentem-0:~$ sudo -i
root@vnfd1deploymentem-0:~# ncs_cli -u admin -C

admin connected from 127.0.0.1 using console on vnfd1deploymentem-0
admin@scm# show running-config | save em-running-12202017.cfg

root@vnfd1deploymentem-0:~# ll em-running-12202017.cfg
-rw-r--r-- 1 root root 19957 Dec 20 23:01 em-running-12202017.cfg
```

5.對EM NCS DB進行備份，並將檔案傳輸到備份伺服器。

```
ubuntu@vnfd1deploymentem-0:~$ sudo -i
root@vnfd1deploymentem-0:~# cd /opt/cisco/em/git/em-scm/ncs-cdb
root@vnfd1deploymentem-0:/opt/cisco/em/git/em-scm/ncs-cdb# ll

total 472716
drwxrwxr-x 2 root root      4096 Dec 20 02:53 ./
drwxr-xr-x 9 root root      4096 Dec 20 19:22 ../
-rw-r--r-- 1 root root       770 Dec 20 02:48 aaa_users.xml
-rw-r--r-- 1 root root     70447 Dec 20 02:53 A.cdb
-rw-r--r-- 1 root root 483927031 Dec 20 02:48 C.cdb
-rw-rw-r-- 1 root root        47 Jul 27 05:53 .gitignore
-rw-rw-r-- 1 root root       332 Jul 27 05:53 global-settings.xml
-rw-rw-r-- 1 root root       621 Jul 27 05:53 jvm-defaults.xml
-rw-rw-r-- 1 root root      3392 Jul 27 05:53 nacm.xml
-rw-r--r-- 1 root root      6156 Dec 20 02:53 O.cdb
-rw-r--r-- 1 root root     13041 Dec 20 02:48 startup-vnfd.xml

root@vnfd1deploymentem-0:/opt/cisco/em/git/em-scm/ncs-cdb#
root@vnfd1deploymentem-0:/opt/cisco/em/git/em-scm# tar cvf em_cdb_backup.tar ncs-cdb

ncs-cdb/
ncs-cdb/O.cdb
ncs-cdb/C.cdb
ncs-cdb/nacm.xml
ncs-cdb/jvm-defaults.xml
ncs-cdb/A.cdb
ncs-cdb/aaa_users.xml
ncs-cdb/global-settings.xml
ncs-cdb/.gitignore
ncs-cdb/startup-vnfd.xml

root@vnfd1deploymentem-0:/opt/cisco/em/git/em-scm# ll em_cdb_backup.tar
-rw-r--r-- 1 root root 484034560 Dec 20 23:06 em_cdb_backup.tar
```

6.導航到scripts目錄，收集日誌並將其傳輸到備份伺服器。

```
/opt/cisco/em-scripts

sudo ./collect-em-logs.sh
```

- 重複相同過程，從NCS備用虛擬機器收集日誌
- 備份所有EM虛擬機器上的系統日誌配置，並將其傳輸到備份伺服器

```
root@vnfd1deploymentem-0:/etc/rsyslog.d# pwd
/etc/rsyslog.d
```

```
root@vnfd1deploymentem-0:/etc/rsyslog.d# ll
total 28

drwxr-xr-x  2 root root 4096 Jun  7 18:38 ./
drwxr-xr-x 86 root root 4096 Jun  6 20:33 ../
-rw-r--r--  1 root root  319 Jun  7 18:36 00-vnmf-proxy.conf
-rw-r--r--  1 root root  317 Jun  7 18:38 01-ncs-java.conf
-rw-r--r--  1 root root  311 Mar 17  2012 20-ufw.conf
-rw-r--r--  1 root root  252 Nov 23  2015 21-cloudinit.conf
-rw-r--r--  1 root root 1655 Apr 18  2013 50-default.conf



root@vnfd1deploymentem-0:/etc/rsyslog.d# ls /etc/rsyslog.conf
rsyslog.conf
```

StarOS備份

對於StarOS，需要備份此資訊。

- 需要定期在遠端伺服器上備份計費資料記錄(CDR)
- 部署系統時應用第0天配置
- N日配置
- 固態硬碟
- 系統日誌配置

還原過程

OSPD復原

OSPD恢復程式基於這些假設來執行

- OSPD備份可從舊OSPD伺服器獲得

- OSPD恢復將在新伺服器上完成，該伺服器將替換系統中的舊OSPD伺服器。

自動部署恢復

1.當VM處於錯誤或關閉狀態時，可恢復自動部署VM，執行硬重新啟動以啟動受影響的虛擬機器。執行這些檢查以瞭解這是否有助於恢復自動部署。

```
Checking AutoDeploy Processes
Verify that key processes are running on the AutoDeploy VM:

root@auto-deploy-iso-2007-uas-0:~# initctl status autodeploy
autodeploy start/running, process 1771

root@auto-deploy-iso-2007-uas-0:~# ps -ef | grep java
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:
```

```
root@auto-deploy-iso-2007-uas-0:~# initctl start autodeploy
autodeploy start/running, process 11094

#To stop the AutoDeploy process:
root@auto-deploy-iso-2007-uas-0:~# initctl stop autodeploy
autodeploy stop/waiting

#To restart the AutoDeploy process:

root@auto-deploy-iso-2007-uas-0:~# initctl restart autodeploy
autodeploy start/running, process 11049

#If the VM is in ERROR or shutdown state, hard-reboot the AutoDeploy VM

[stack@pod1-ospd ~]$ nova list |grep auto-deploy
| 9b55270a-2dcd-4ac1-aba3-bf041733a0c9 | auto-deploy-ISO-2007-uas-
0                                      | ACTIVE | -          | running     | mgmt=172.16.181.12,
10.84.123.39

[stack@pod1-ospd ~]$ nova reboot –hard 9b55270a-2dcd-4ac1-aba3-bf041733a0c9
```

2.如果「自動部署」不可恢復，請按照以下步驟操作，將其還原到之前的狀態。使用之前進行的備份。

```
[stack@pod1-ospd ~]$ nova list |grep auto-deploy
| 9b55270a-2dcd-4ac1-aba3-bf041733a0c9 | auto-deploy-ISO-2007-uas-
0                                      | ACTIVE | -          | running     | mgmt=172.16.181.12,
10.84.123.39 [stack@pod1-ospd ~]$ cd /opt/cisco/usp/uas-installer/scripts

[stack@pod1-ospd ~]$ ./auto-deploy-booting.sh --floating-ip 10.1.1.2 --delete
```

3.刪除AutoDeploy後，請使用相同的floatingip地址重**新創**建它。

```
[stack@pod1-ospd ~]$ cd /opt/cisco/usp/uas-installer/scripts

[stack@pod1-ospd scripts]$ ./auto-deploy-booting.sh --floating-ip 10.1.1.2

2017-11-17 07:05:03,038 - INFO: Creating AutoDeploy deployment (1 instance(s)) on
'http://10.1.1.2:5000/v2.0' tenant 'core' user 'core', ISO 'default'
2017-11-17 07:05:03,039 - INFO: Loading image 'auto-deploy-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2' from '/opt/cisco/usp/uas-installer/images/usp-uas-1.0.1-1504.qcow2'
2017-11-17 07:05:14,603 - INFO: Loaded image 'auto-deploy-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2'
2017-11-17 07:05:15,787 - INFO: Assigned floating IP '10.1.1.2' to IP '172.16.181.7'
2017-11-17 07:05:15,788 - INFO: Creating instance 'auto-deploy-ISO-5-1-7-2007-uas-0'
2017-11-17 07:05:42,759 - INFO: Created instance 'auto-deploy-ISO-5-1-7-2007-uas-0'
2017-11-17 07:05:42,759 - INFO: Request completed, floating IP: 10.1.1.2]
```

4.將**Autodeploy.cfg**檔案、ISO和**confd_backup tar**檔案從備份伺服器複製到AutoDeploy VM。

5.從備份tar檔案中恢復confd cdb檔案。

```
ubuntu@auto-deploy-iso-2007-uas-0:~# sudo -i
ubuntu@auto-deploy-iso-2007-uas-0:#  service uas-confd stop
uas-confd stop/waiting

root@auto-deploy-iso-2007-uas-0:# cd /opt/cisco/usp/uas/confd-6.3.1/var/confd
root@auto-deploy-iso-2007-uas-0:/opt/cisco/usp/uas/confd-6.3.1/var/confd# tar xvf
/home/ubuntu/ad_cdb_backup.tar

cdb/
```

```
cdb/O.cdb
cdb/C.cdb
cdb/aaa_init.xml
cdb/A.cdb

root@auto-deploy-iso-2007-uas-0~# service uas-confd start
uas-confd start/running, process 2036

#Restart AutoDeploy process

root@auto-deploy-iso-2007-uas-0~# service autodeploy restart
autodeploy start/running, process 2144


#Check that confd was loaded properly by checking earlier transactions.

root@auto-deploy-iso-2007-uas-0:~# confd_cli -u admin -C
Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on auto-deploy-iso-2007-uas-0

auto-deploy-iso-2007-uas-0#show transaction

                                  SERVICE
       SITE

                                  DEPLOYMENT
SITE  TX    AUTOVNF  VNF  AUTOVNF

TX ID          TX TYPE             ID          DATE AND TIME
STATUS              ID    ID    ID       ID   TX ID

------------------------------------------------------------------------------------------------
-------------------------------------
1512571978613  service-deployment  tb5bxb      2017-12-06T14:52:59.412+00:00  deployment-success
```

6.如果虛擬機器成功恢復並運行；確保從以前成功的已知備份還原所有系統日誌特定配置。

```
ubuntu@auto-deploy-vnf-iso-5-1-5-1196-uas-0:~$sudo su
root@auto-deploy-vnf-iso-5-1-5-1196-uas-0:/home/ubuntu#ls /etc/rsyslog.d/00-autodeploy.conf
00-autodeploy.conf

root@auto-deploy-vnf-iso-5-1-5-1196-uas-0:/home/ubuntu#ls /etc/rsyslog.conf
rsyslog.conf
```

AutoIT-VNF復原

1. AutoIT-VNF VM可恢復，如果VM處於錯誤或關閉狀態，請執行硬重新啟動以啟動受影響的虛擬機器。執行這些步驟以恢復AutoIT-VNF。

```
Checking AutoIT-VNF Processes

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

root@auto-it-vnf-iso-5-1-5-1196-uas-0:~# service autoit status
AutoIT-VNF is running.

#Stopping/Restarting AutoIT-VNF Processes

root@auto-it-vnf-iso-5-1-5-1196-uas-0:~# service autoit stop
AutoIT-VNF API server stopped.
```

```
#To restart the AutoIT-VNF processes:

root@auto-it-vnf-iso-5-1-5-1196-uas-0:~# service autoit restart
AutoIT-VNF API server stopped.
Starting AutoIT-VNF

/opt/cisco/usp/apps/auto-it/vnf
AutoIT API server started.

#If the VM is in ERROR or shutdown state, hard-reboot the AutoDeploy VM

[stack@pod1-ospd ~]$ nova list |grep auto-it
| 1c45270a-2dcd-4ac1-aba3-bf041733d1a1 | auto-it-vnf-ISO-2007-uas-
0                                    | ACTIVE | -          | running     | mgmt=172.16.181.13,
10.84.123.40

[stack@pod1-ospd ~]$ nova reboot –hard 1c45270a-2dcd-4ac1-aba3-bf041733d1a1
```

2.如果AutoIT-VNF不可恢復，請按照以下過程將其還原到其之前的狀態。使用備份檔案。

```
 [stack@pod1-ospd ~]$ nova list |grep auto-it
| 580faf80-1d8c-463b-9354-781ea0c0b352 | auto-it-vnf-ISO-2007-uas-
0                                    | ACTIVE | -          | running     | mgmt=172.16.181.3,
10.84.123.42 [stack@pod1-ospd ~]$ cd /opt/cisco/usp/uas-installer/scripts

[stack@pod1-ospd ~]$ ./ auto-it-vnf-staging.sh --floating-ip 10.1.1.3 --delete
```

3.通過運行**auto-it-vnf staging**指令碼重新建立自動IT，並確保使用以前使用的相同浮動IP。

```
[stack@pod1-ospd ~]$ cd /opt/cisco/usp/uas-installer/scripts

[stack@pod1-ospd scripts]$ ./auto-it-vnf-staging.sh --floating-ip 10.1.1.3

2017-11-16 12:54:31,381 - INFO: Creating StagingServer deployment (1 instance(s)) on
'http://10.1.1.3:5000/v2.0' tenant 'core' user 'core', ISO 'default'
2017-11-16 12:54:31,382 - INFO: Loading image 'auto-it-vnf-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2' from '/opt/cisco/usp/uas-installer/images/usp-uas-1.0.1-1504.qcow2'
2017-11-16 12:54:51,961 - INFO: Loaded image 'auto-it-vnf-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2'
2017-11-16 12:54:53,217 - INFO: Assigned floating IP '10.1.1.3' to IP '172.16.181.9'
2017-11-16 12:54:53,217 - INFO: Creating instance 'auto-it-vnf-ISO-5-1-7-2007-uas-0'
2017-11-16 12:55:20,929 - INFO: Created instance 'auto-it-vnf-ISO-5-1-7-2007-uas-0'
2017-11-16 12:55:20,930 - INFO: Request completed, floating IP: 10.1.1.3
```

4.需要在AutoIT-VNF上重新載入POD中使用的ISO映像。

```
[stack@pod1-ospd ~]$ cd images/5_1_7-2007/isos

[stack@pod1-ospd isos]$ curl -F file=@usp-5_1_7-2007.iso http://10.1.1.3:5001/isos

{
 "iso-id": "5.1.7-2007"
}

Note: 10.1.1.3 is AutoIT-VNF IP in the above command.

#Validate that ISO is correctly loaded.

[stack@pod1-ospd isos]$ curl http://10.1.1.3:5001/isos

{
 "isos": [
   {
     "iso-id": "5.1.7-2007"
   }
 ]
}
```

5.將VNF **system.cfg**檔案從遠端伺服器複製到AutoIT-VNF虛擬機器。在本示例中，它從AutoDeploy複製到AutoIT-VNF VM。

```
[stack@pod1-ospd autodeploy]$ scp system-vnf* ubuntu@10.1.1.3:.
ubuntu@10.1.1.3's password:

system-
vnf1.cfg
       100% 1197     1.2KB/s   00:00
system-
vnf2.cfg
       100% 1197     1.2KB/s   00:00

ubuntu@auto-it-vnf-iso-2007-uas-0:~$ pwd
/home/ubuntu

ubuntu@auto-it-vnf-iso-2007-uas-0:~$ ls
system-vnf1.cfg  system-vnf2.cfg
```

6.將檔案複製到AutoIT-VNF上的適當位置，如AutoDeploy配置中所參考。請參閱此處；

```
ubuntu@auto-it-vnf-iso-2007-uas-0:~$ sudo -i
root@auto-it-vnf-iso-2007-uas-0:~$ cp –rp system-vnf1.cfg system-vnf2.cfg
/opt/cisco/usp/uploads/
root@auto-it-vnf-iso-2007-uas-0:~$ls /opt/cisco/usp/uploads/
system-vnf1.cfg  system-vnf2.cfg
```

7.如果VM成功還原並運行，請確保從以前成功的已知備份還原所有系統日誌特定配置。

```
root@auto-deploy-vnf-iso-5-1-5-1196-uas-0:/home/ubuntu#ls /etc/rsyslog.d/00-autoit-vnf.conf
00-autoit-vnf.conf

root@auto-deploy-vnf-iso-5-1-5-1196-uas-0:ls /etc/rsyslog.conf
rsyslog.conf
```

AutoVNF恢復

1.如果AutoVNF VM處於錯誤或關閉狀態，則該虛擬機器可恢復。執行硬重新啟動，以啟動受影響的虛擬機器。執行這些步驟以恢復AutoVNF。

2.確定處於「錯誤」或「關閉」狀態的VM。硬重新啟動AutoVNF VM。

在本範例中，reboot **auto-testautovnf1-uas-2**。

```
[root@tb1-baremetal scripts]# nova list | grep "auto-testautovnf1-uas-[0-2]"

| 3834a3e4-96c5-49de-a067-68b3846fba6b | auto-testautovnf1-uas-
0                                        | ACTIVE | -          | running     | auto-
testautovnf1-uas-orchestration=172.57.12.6; auto-testautovnf1-uas-
management=172.57.11.8
                |
| 0fbfec0c-f4b0-4551-807b-50c5fe9d3ea7 | auto-testautovnf1-uas-
```

```
1                                        | ACTIVE | -          | running    | auto-
testautovnf1-uas-orchestration=172.57.12.7; auto-testautovnf1-uas-management=172.57.11.12
                                                                                   |
| 432e1a57-00e9-4e58-8bef-2a20652df5bf | auto-testautovnf1-uas-
2                                        | ACTIVE | -          | running    | auto-
testautovnf1-uas-orchestration=172.57.12.13; auto-testautovnf1-uas-
management=172.57.11.4
            |

[root@tb1-baremetal scripts]# nova reboot --hard 432e1a57-00e9-4e58-8bef-2a20652df5bf
Request to reboot server <Server: auto-testautovnf1-uas-2> has been accepted.

[root@tb1-baremetal scripts]#
```

3.一旦虛擬機器啟動，驗證它是否重新加入群集。

```
root@auto-testautovnf1-uas-1:/opt/cisco/usp/uas/scripts# confd_cli -u admin -C
Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on auto-testautovnf1-uas-1

auto-testautovnf1-uas-1#show uas

uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.57.11.101

INSTANCE IP   STATE  ROLE
-----------------------------------

172.57.12.6   alive  CONFD-SLAVE
172.57.12.7   alive  CONFD-MASTER
172.57.12.13  alive  NA
```

4.如果上述過程無法恢復AutoVNF VM，您需要藉助這些步驟進行恢復。

```
[stack@pod1-ospd ~]$ nova list | grep vnf1-UAS-uas-0
| 307a704c-a17c-4cdc-8e7a-3d6e7e4332fa | vnf1-UAS-uas-
0                                                 | ACTIVE | -          | running     | vnf1-
UAS-uas-orchestration=172.168.11.10; vnf1-UAS-uas-management=172.168.10.3

[stack@pod1-ospd ~]$ nova delete vnf1-UAS-uas-0
Request to delete server vnf1-UAS-uas-0 has been accepted.
```

5.為了恢復**autovnf-uas** VM，請執行uas-**check指令碼以**檢查狀態。它必須報告錯誤。然後使用**—fix選項**再次執行，以重新建立缺失的UAS VM。

```
[stack@pod1-ospd ~]$ cd /opt/cisco/usp/uas-installer/scripts/
[stack@pod1-ospd scripts]$ ./uas-check.py auto-vnf vnf1-UAS

2017-12-08 12:38:05,446 - INFO: Check of AutoVNF cluster started
2017-12-08 12:38:07,925 - INFO: Instance 'vnf1-UAS-uas-0' status is 'ERROR'
2017-12-08 12:38:07,925 - INFO: Check completed, AutoVNF cluster has recoverable errors

[stack@tb3-ospd scripts]$ ./uas-check.py auto-vnf vnf1-UAS --fix

2017-11-22 14:01:07,215 - INFO: Check of AutoVNF cluster started
2017-11-22 14:01:09,575 - INFO: Instance vnf1-UAS-uas-0' status is 'ERROR'
2017-11-22 14:01:09,575 - INFO: Check completed, AutoVNF cluster has recoverable errors
2017-11-22 14:01:09,778 - INFO: Removing instance vnf1-UAS-uas-0'
2017-11-22 14:01:13,568 - INFO: Removed instance vnf1-UAS-uas-0'
2017-11-22 14:01:13,568 - INFO: Creating instance vnf1-UAS-uas-0' and attaching volume ‘vnf1-
```

```
UAS-uas-vol-0'
2017-11-22 14:01:49,525 - INFO: Created instance ‘vnf1-UAS-uas-0'

[stack@tb3-ospd scripts]$ ./uas-check.py auto-vnf vnf1-UAS

2017-11-16 13:11:07,472 - INFO: Check of AutoVNF cluster started
2017-11-16 13:11:09,510 - INFO: Found 3 ACTIVE AutoVNF instances
2017-11-16 13:11:09,511 - INFO: Check completed, AutoVNF cluster is fine
```

6.登入到主AutoVNF虛擬機器。恢復後幾分鐘內，新建立的例項必須加入群集並處於活動狀態。

```
tb3-bxb-vnf1-autovnf-uas-0#show uas

uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.17.181.101

INSTANCE IP   STATE  ROLE
-----------------------------------
172.17.180.6  alive  CONFD-SLAVE
172.17.180.7  alive  CONFD-MASTER
172.17.180.9  alive  NA

#if uas-check.py --fix fails, you may need to copy this file and execute again.

[stack@tb3-ospd]$ mkdir -p /opt/cisco/usp/apps/auto-it/common/uas-deploy/
[stack@tb3-ospd]$ cp /opt/cisco/usp/uas-installer/common/uas-deploy/userdata-uas.txt
/opt/cisco/usp/apps/auto-it/common/uas-deploy/
```

7.如果VM成功還原並運行，請確保從以前成功的已知備份還原所有系統日誌特定配置。確保它在所有AutoVNF虛擬機器中恢復。

```
ubuntu@auto-testautovnf1-uas-1:~$sudo su
root@auto-testautovnf1-uas-1:/home/ubuntu#ls /etc/rsyslog.d/00-autovnf.conf
00-autovnf.conf

root@auto-testautovnf1-uas-1:/home/ubuntu#ls /etc/rsyslog.conf
rsyslog.conf
```

ESC恢復

1.如果VM處於錯誤或關閉狀態，ESC虛擬機器可恢復。執行硬重新啟動，以啟動受影響的虛擬機器。執行這些步驟以恢復ESC。

2.確定處於「錯誤」或「關閉」狀態的VM，一旦確定硬重新啟動ESC VM。在此示例中，自動測試vnfm1-ESC-0被重新啟動。

```
[root@tb1-baremetal scripts]# nova list | grep auto-test-vnfm1-ESC-

| f03e3cac-a78a-439f-952b-045aea5b0d2c | auto-test-vnfm1-ESC-
0                                       | ACTIVE | -          | running     | auto-testautovnf1-
uas-orchestration=172.57.12.11; auto-testautovnf1-uas-
management=172.57.11.3
                         |
| 79498e0d-0569-4854-a902-012276740bce | auto-test-vnfm1-ESC-
1                                       | ACTIVE | -          | running     | auto-testautovnf1-
uas-orchestration=172.57.12.15; auto-testautovnf1-uas-
management=172.57.11.5
            |
[root@tb1-baremetal scripts]# [root@tb1-baremetal scripts]# nova reboot --hard f03e3cac-a78a-
439f-952b-045aea5b0d2c\
Request to reboot server <Server: auto-test-vnfm1-ESC-0> has been accepted.

[root@tb1-baremetal scripts]#
```

3.如果刪除了ESC VM並且需要重新啟動，請按照以下步驟執行。

```
[stack@pod1-ospd scripts]$ nova list |grep ESC-1
| c566efbf-1274-4588-a2d8-0682e17b0d41 | vnf1-ESC-ESC-
1                                                  | ACTIVE | -          | running     | vnf1-
UAS-uas-orchestration=172.168.11.14; vnf1-UAS-uas-
management=172.168.10.4
                        |

[stack@pod1-ospd scripts]$ nova delete vnf1-ESC-ESC-1
Request to delete server vnf1-ESC-ESC-1 has been
accepted.
```

4.從AutoVNF-UAS中查詢ESC部署事務，並在事務的日誌中查詢boot_vm.py命令列以建立ESC例項。

```
ubuntu@vnf1-uas-uas-0:~$ sudo -i
root@vnf1-uas-uas-0:~# confd_cli -u admin -C
Welcome to the ConfD CLI

admin connected from 127.0.0.1 using console on vnf1-uas-uas-0
vnf1-uas-uas-0#show transaction

TX ID                                 TX TYPE          DEPLOYMENT ID
TIMESTAMP                         STATUS
------------------------------------------------------------------------------------------------
------------------------------
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b  vnf-deployment   vnf1-DEPLOYMENT  2017-11-
29T02:01:27.750692-00:00  deployment-success
73d9c540-d4a8-11e7-bb72-fa163ef8df2b  vnfm-deployment  vnf1-ESC         2017-11-
29T01:56:02.133663-00:00  deployment-success

vnf1-uas-uas-0#show logs 73d9c540-d4a8-11e7-bb72-fa163ef8df2b | 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>73d9c540-d4a8-11e7-bb72-fa163ef8df2b</tx-id>
   <log>2017-11-29 01:56:02,142 - VNFM Deployment RPC triggered for deployment: vnf1-ESC,
deactivate: 0

2017-11-29 01:56:02,179 - Notify deployment
..
2017-11-29 01:57:30,385 - Creating VNFM 'vnf1-ESC-ESC-1' with [python //opt/cisco/vnf-
staging/bootvm.py vnf1-ESC-ESC-1 --flavor vnf1-ESC-ESC-flavor --image 3fe6b197-961b-4651-af22-
dfd910436689 --net vnf1-UAS-uas-management --gateway_ip 172.168.10.1 --net vnf1-UAS-uas-
orchestration --os_auth_url http://10.1.1.5:5000/v2.0 --os_tenant_name core --os_username ******
--os_password ****** --bs_os_auth_url http://10.1.1.5:5000/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 172.168.10.7 --ipaddr 172.168.10.6 dhcp --ha_node_list 172.168.10.3 172.168.10.6
--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]...
```

5.將**boot_vm.py**行儲存到Shell指令碼檔案(esc.sh)中，並使用正確的資訊(通常為core/Cisco@123)更新所有使用者名稱*****和密碼*****行。 您也需要移除 — encrypt_key選項。對於user_pass和user_confd_pass，需要使用 — user_passwd username:password(示例 — admin:Cisco@123)格式。

現在，從running-config查詢**bootvm.py**的URL，然後將**bootvm.py**檔案獲取到autovnf-uas VM。在本例中，10.1.1.3是自動IT。

```
root@vnf1-uas-uas-0:~# confd_cli -u admin -C
Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on vnf1-uas-uas-0

vnf1-uas-uas-0#show running-config autovnf-vnfm:vnfm
…
configs bootvm
 value http://10.1.1.3:80/bundles/5.1.7-2007/vnfm-bundle/bootvm-2_3_2_155.py
!

root@vnf1-uas-uas-0:~# wget http://10.1.1.3:80/bundles/5.1.7-2007/vnfm-bundle/bootvm-
2_3_2_155.py

--2017-12-01 20:25:52--  http://10.1.1.3/bundles/5.1.7-2007/vnfm-bundle/bootvm-2_3_2_155.py
Connecting to 10.1.1.3:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 127771 (125K) [text/x-python]
Saving to: ‘bootvm-2_3_2_155.py’

100%[===============================================================================================
===========>] 127,771     --.-K/s   in 0.001s


2017-12-01 20:25:52 (173 MB/s) - ‘bootvm-2_3_2_155.py’ saved [127771/127771
Create a /tmp/esc_params.cfg file.

root@vnf1-uas-uas-0:~# echo "openstack.endpoint=publicURL" > /tmp/esc_params.cfg
```

6.使用選項執行執行**bootvm.py** python指令碼的shell指令碼。

```
root@vnf1-uas-uas-0:~# /bin/sh esc.sh
+ python ./bootvm.py vnf1-ESC-ESC-1 --flavor vnf1-ESC-ESC-flavor --image 3fe6b197-961b-4651-
af22-dfd910436689 --net vnf1-UAS-uas-management --gateway_ip 172.168.10.1 --net vnf1-UAS-uas-
orchestration --os_auth_url http://10.1.1.5:5000/v2.0 --os_tenant_name core --os_username core -
-os_password Cisco@123 --bs_os_auth_url http://10.1.1.5:5000/v2.0 --bs_os_tenant_name core --
bs_os_username core --bs_os_password Cisco@123 --esc_ui_startup false --esc_params_file
/tmp/esc_params.cfg --user_pass admin:Cisco@123 --user_confd_pass admin:Cisco@123 --kad_vif eth0
--kad_vip 172.168.10.7 --ipaddr 172.168.10.6 dhcp --ha_node_list 172.168.10.3 172.168.10.6 --
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


+-----------------------------------+------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------+
| Property                          |
Value


                                                               |
+--------------------------------------+-------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------+
| OS-DCF:diskConfig                    |
MANUAL


                                   |
| OS-EXT-AZ:availability_zone          |
mgmt


                                                     |
| OS-EXT-SRV-ATTR:host                 | tb5-ultram-osd-compute-
1.localdomain


                                       |
| OS-EXT-SRV-ATTR:hypervisor_hostname  | tb5-ultram-osd-compute-
1.localdomain


                         |
| OS-EXT-SRV-ATTR:instance_name        | instance-
000001eb

                                                                                              |
| OS-EXT-STS:power_state               |
1


                             |
| OS-EXT-STS:task_state                | -


               |
| OS-EXT-STS:vm_state                  |
active


                                 |
| OS-SRV-USG:launched_at               | 2017-12-
02T13:28:32.000000


                   |
| OS-SRV-USG:terminated_at             | -


     |
| accessIPv4                           |


                                                                                       |
| accessIPv6
|
         |
| addresses                            | {"vnf1-UAS-uas-orchestration": [{"OS-EXT-IPS-
MAC:mac_addr": "fa:16:3e:d7:c6:19", "version": 4, "addr": "172.168.11.14", "OS-EXT-IPS:type":
"fixed"}], "vnf1-UAS-uas-management": [{"OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:31:ee:cd",
"version": 4, "addr": "172.168.10.6", "OS-EXT-IPS:type": "fixed"}]}
| config_drive                         | True


                                                                              |
| created                              | 2017-12-
02T13:27:49Z

                                                                                                |
| flavor                               | {"id": "457623b6-05d5-403c-b2e4-aa3b6a0c9d32", "links":
[{"href": "http://10.1.1.5:8774/flavors/457623b6-05d5-403c-b2e4-aa3b6a0c9d32", "rel":
"bookmark"}]}
                                                      |
| hostId                               |
f5d2bbf0c5a7df34cf2e6f62ae0702ef120ff82f81c3f7664ffb35e9


       |
| id                                   | 2601b8ec-8ff8-4285-810a-
e859f6642ab6

                                                                                      |
| image                                | {"id": "3fe6b197-961b-4651-af22-dfd910436689", "links":
[{"href": "http://10.1.1.5:8774/images/3fe6b197-961b-4651-af22-dfd910436689", "rel":
"bookmark"}]}
                                                       |
| key_name                             | -


                                                          |
| metadata                             |
{}


                                                                            |
| name                                 | vnf1-esc-esc-
1


                                                              |
| os-extended-volumes:volumes_attached | []


                                                |
| progress                             |
0


                                                                  |
| security_groups                      | [{"name": "default"}, {"name":
"default"}]


                                                    |
| status                               |
ACTIVE


                                      |
| tenant_id                            |
fd4b15df46c6469cbacf5b80dcc98a5c



                                                        |
| updated                              | 2017-12-
02T13:28:32Z



                                          |
| user_id                              |
d3b51d6f705f4826b22817f27505c6cd



                           |
```

7.在OSPD中，檢查新的ESC虛擬機器是否處於活動/運行狀態。

```
[stack@pod1-ospd ~]$ nova list|grep -i esc
| 934519a4-d634-40c0-a51e-fc8d55ec7144 | vnf1-ESC-ESC-
0                                                  | ACTIVE | -          | running     | vnf1-
UAS-uas-orchestration=172.168.11.13; vnf1-UAS-uas-
management=172.168.10.3
                        |
| 2601b8ec-8ff8-4285-810a-e859f6642ab6 | vnf1-ESC-ESC-
1                                                  | ACTIVE | -          | running     | vnf1-
UAS-uas-orchestration=172.168.11.14; vnf1-UAS-uas-
management=172.168.10.6
                        |

#Log in to new ESC and verify Backup state.  You may execute health.sh on ESC Master too.

ubuntu@vnf1-uas-uas-0:~$ ssh admin@172.168.11.14
…
  ####################################################################
  #   ESC on vnf1-esc-esc-1.novalocal is in BACKUP state.
  ####################################################################

[admin@vnf1-esc-esc-1 ~]$ escadm status
0 ESC status=0 ESC Backup Healthy

[admin@vnf1-esc-esc-1 ~]$ health.sh
============== ESC HA (BACKUP) =================
=======================================

ESC HEALTH PASSED
[admin@vnf1-esc-esc-1 ~]$ cat /proc/drbd
version: 8.4.7-1 (api:1/proto:86-101)
GIT-hash: 3a6a769340ef93b1ba2792c6461250790795db49 build by mockbuild@Build64R6, 2016-01-12
13:27:11

 1: cs:Connected ro:Secondary/Primary ds:UpToDate/UpToDate C r-----
    ns:0 nr:504720 dw:3650316 dr:0 al:8 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:0
```

8.如果ESC VM不可恢復並且需要還原資料庫，請從以前備份中還原資料庫。

9.對於ESC資料庫還原，請確保在還原資料庫之前停止ESC服務；對於ESC HA，先在輔助VM中執行，然後在主VM中執行。

```
# service keepalived stop
```

10.檢查ESC服務狀態，並確保在HA的主和輔助VM中一切都已停止。

```
# escadm status
```

11.執行指令碼以恢復資料庫。作為將資料庫恢復到新建立的ESC例項的一部分，該工具還將將其中一個例項升級為主ESC，將其資料庫資料夾裝載到DRBD裝置並將啟動PostgreSQL資料庫。

```
# /opt/cisco/esc/esc-scripts/esc_dbtool.py restore --file
scp://<username>:<password>@<backup_vm_ip>:<filename>
```

12.重新啟動ESC服務以完成資料庫還原。

13.對於在兩個虛擬機器中執行的HA，請重新啟動keepalive服務。

```
# service keepalived start
```

14.虛擬機器成功恢復並運行後；確保從以前成功的已知備份還原所有系統日誌特定配置。確保它在所有ESC虛擬機器中恢復。

```
[admin@auto-test-vnfm2-esc-1 ~]$
[admin@auto-test-vnfm2-esc-1 ~]$ cd /etc/rsyslog.d
[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.d/00-escmanager.conf
00-escmanager.conf

[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.d/01-messages.conf
01-messages.conf

[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.d/02-mona.conf
02-mona.conf

[admin@auto-test-vnfm2-esc-1 rsyslog.d]$ls /etc/rsyslog.conf
rsyslog.conf
```

EM恢復

1.如果EM VM由於某個條件或其他條件而處於「無/錯誤」狀態，使用者可以按照給定的順序恢復受影響的EM VM。

2. ESC/VNFM是監視EM虛擬機器的元件，因此，在EM處於錯誤狀態的情況下，ESC將嘗試自動恢復EM虛擬機器。出於任何原因，如果ESC無法成功完成恢復，ESC會將該VM標籤為錯誤狀態。

3.在這種情況下，一旦基本基礎架構問題得到解決，使用者就可以手動恢復EM虛擬機器。只有在基本問題得到解決之後，才能執行此手動恢復，這一點非常重要。

4.確定虛擬機器處於錯誤狀態。

```
[stack@pod1-ospd ~]$ source corerc
[stack@pod1-ospd ~]$ nova list --field name,host,status |grep -i err
| c794207b-a51e-455e-9a53-3b8ff3520bb9 | vnf1-DEPLOYMENT-_vnf1-D_0_a6843886-77b4-4f38-b941-
74eb527113a8 | None                                    | ERROR   |
```

5.登入到ESC主伺服器，對每個**受影響的EM和**CF **VM**執行恢復虛擬機器操作。耐心點。ESC將安排恢復操作，此操作可能在幾分鐘內不會發生。

```
ubuntu@vnf1-uas-uas-1:~$ ssh admin@172.168.10.3
```

```
…

[admin@vnf1-esc-esc-0 ~]$ sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO
vnf1-DEPLOYMENT-_vnf1-D_0_a6843886-77b4-4f38-b941-74eb527113a8
[sudo] password for admin:

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

6.監控**/var/log/esc/yangesc.log**，直到命令完成。

```
[admin@vnf1-esc-esc-0 ~]$ tail -f /var/log/esc/yangesc.log

…

14:59:50,112 07-Nov-2017 WARN  Type: VM_RECOVERY_COMPLETE
14:59:50,112 07-Nov-2017 WARN  Status: SUCCESS
14:59:50,112 07-Nov-2017 WARN  Status Code: 200
14:59:50,112 07-Nov-2017 WARN  Status Msg: Recovery: Successfully recovered VM [vnf1-DEPLOYMENT-
_vnf1-D_0_a6843886-77b4-4f38-b941-74eb527113a8]

#Log in to new EM and verify EM state is up.

ubuntu@vnf1vnfddeploymentem-1:~$ /opt/cisco/ncs/current/bin/ncs_cli -u admin -C

admin connected from 172.17.180.6 using ssh on vnf1vnfddeploymentem-1

admin@scm# show ems

EM            VNFM

ID  SLA  SCM  PROXY
---------------------
2   up   up   up
3   up   up   up
```

當ESC無法啟動VM時

1.在某些情況下，ESC由於意外狀態而無法啟動VM。解決方法是重新啟動主ESC來執行ESC切換。ESC切換大約需要一分鐘。在新主ESC上執行**health.sh**以驗證它是否啟動。當ESC成為主時，ESC可能會修復VM狀態並啟動VM。由於此操作已計畫，您必須等待5-7分鐘才能完成。

2.您可以監控**/var/log/esc/yangesc.log**和**/var/log/esc/escmanager.log**。如果您看到在5-7分鐘之後沒有恢復虛擬機器，則使用者需要手動恢復受影響的虛擬機器。

3.一旦虛擬機器成功恢復並運行；確保從以前成功的已知備份還原所有系統日誌特定配置。確保它在所有ESC虛擬機器中恢復。

```
root@abautotestvnfm1em-0:/etc/rsyslog.d# pwd
/etc/rsyslog.d

root@abautotestvnfm1em-0:/etc/rsyslog.d# ll

total 28
drwxr-xr-x  2 root root 4096 Jun  7 18:38 ./
```

```
drwxr-xr-x 86 root root 4096 Jun  6 20:33 ../]
-rw-r--r--  1 root root  319 Jun  7 18:36 00-vnmf-proxy.conf
-rw-r--r--  1 root root  317 Jun  7 18:38 01-ncs-java.conf
-rw-r--r--  1 root root  311 Mar 17  2012 20-ufw.conf
-rw-r--r--  1 root root  252 Nov 23  2015 21-cloudinit.conf
-rw-r--r--  1 root root 1655 Apr 18  2013 50-default.conf

root@abautotestvnfm1em-0:/etc/rsyslog.d# ls /etc/rsyslog.conf
rsyslog.conf
```

StarOS復原

1.如果某個StarOS VM由於某個條件或其他條件而處於「無/錯誤」狀態，使用者可以按照此順序恢復受影響的StarOS VM。

2. ESC/VNFM是監控StarOS虛擬機器的元件，因此在CF/SF虛擬機器處於錯誤狀態的情況下，ESC將嘗試自動恢復CF/SF虛擬機器。無論出於何種原因，如果ESC無法成功完成恢復，ESC會將該VM標籤為錯誤狀態。

3.在這種情況下，一旦基本基礎架構問題得到解決，使用者就可以手動恢復CF/SF虛擬機器。只有在修復了底層問題之後，才能執行此手動恢復，這一點非常重要。

4.確定處於「錯誤」狀態的VM。

```
[stack@pod1-ospd ~]$ source corerc

[stack@pod1-ospd ~]$ nova list --field name,host,status |grep -i err
| c794207b-a51e-455e-9a53-3b8ff3520bb9 | vnf1-DEPLOYMENT-_s4_0_c2b19084-26b3-4c9c-8639-
62428a4cb3a3 | None                                 | ERROR   |
```

5.登入到ESC主伺服器，對每個受影響的EM和CF VM執行**recovery-vm-action**。Be patient。ESC將安排恢復操作，此操作可能在幾分鐘內不會發生。

```
ubuntu@vnf1-uas-uas-1:~$ ssh admin@172.168.10.3
…

[admin@vnf1-esc-esc-0 ~]$ sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO
vnf1-DEPLOYMENT-_s4_0_c2b19084-26b3-4c9c-8639-62428a4cb3a3
[sudo] password for admin:

Recovery VM Action

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

##Monitor the /var/log/esc/yangesc.log until command completes.

[admin@vnf1-esc-esc-0 ~]$ tail -f /var/log/esc/yangesc.log
…
14:59:50,112 07-Nov-2017 WARN  Type: VM_RECOVERY_COMPLETE
14:59:50,112 07-Nov-2017 WARN  Status: SUCCESS
14:59:50,112 07-Nov-2017 WARN  Status Code: 200
14:59:50,112 07-Nov-2017 WARN  Status Msg: Recovery: Successfully recovered VM [vnf1-DEPLOYMENT-
_s4_0_c2b19084-26b3-4c9c-8639-62428a4cb3a3]
```

6.另外，通過在StarOS上運行**show card**頁籤來驗證相同內容。如果恢復的VM是SF，使用者可能需要在需要時將其啟用。進行必要的StarOS配置更改。

```
[local]VNF1# show card tab
Saturday December 02 14:40:20 UTC 2017
Slot         Card Type                               Oper State     SPOF  Attach

-----------  --------------------------------------  -------------  ----  ------

 1: CFC      Control Function Virtual Card           Active         No
2: CFC      Control Function Virtual Card           Standby        -
 3: FC       4-Port Service Function Virtual Card    Active         No
 4: FC       4-Port Service Function Virtual Card    Active         No
 5: FC       4-Port Service Function Virtual Card    Active         No
 6: FC       4-Port Service Function Virtual Card    Standby        -
 7: FC       4-Port Service Function Virtual Card    Active         No
 8: FC       4-Port Service Function Virtual Card    Active         No
 9: FC       4-Port Service Function Virtual Card    Active         No
10: FC       4-Port Service Function Virtual Card    Active         No
```

當ESC無法啟動VM時

在某些情況下，ESC由於意外狀態而無法啟動VM。解決方法是重新啟動主ESC來執行ESC切換。ESC切換大約需要一分鐘。在新主ESC上執行**health.sh**以驗證它是否啟動。當ESC成為主時，ESC可能會修復VM狀態並啟動VM。由於此操作已計畫，您必須等待5-7分鐘才能完成。您可以監控**/var/log/esc/yangesc.log**和**/var/log/esc/escmanager.log**。如果您在5-7分鐘後仍未看到虛擬機器被恢復，則您需要執行受影響虛擬機器的手動恢復。