

다양한 Ultra-M 구성 요소의 백업 및 복원 절차 - CPS

목차

- [소개](#)
- [배경 정보](#)
- [약어](#)
- [백업 절차](#)
 - [OSPD 백업](#)
 - [ESC 백업](#)
 - [CPS 백업](#)
- [복원 절차](#)
 - [OSPD 복구](#)
 - [ESC 복구](#)
 - [CPS 복구](#)

소개

이 문서에서는 Calls CPS Virtual Network Functions를 호스팅하는 Ultra-M 설정에서 가상 머신을 백업하고 복원하는 데 필요한 단계를 설명합니다.

배경 정보

Ultra-M은 VNF(Virtual Network Function)의 구축을 간소화하도록 설계된 사전 패키지 및 검증된 가상화 모바일 패킷 코어 솔루션입니다. Ultra-M 솔루션은 다음과 같은 VM(Virtual Machine) 유형으로 구성됩니다.

- Elastic Services Controller(ESC)
- CPS(Cisco Policy Suite)

Ultra-M의 고급 아키텍처 및 관련 구성 요소는 이 그림과 같습니다.

Ultra-M

Staging Server

UAS/UWS

Cisco Ultra Service Platform

USP Element Manager

Ultra Policy Platform

Ultra Gateway Platform

Ultra Service Framework

Ve-Vnfm-em

Ve-Vnfm-vnf

Vn-Nf

Vn-Nf

UCS-C

Virtual Compute

Virtual Storage

Virtual Network

Virtualization

Compute

Storage

Network

Nf-Vi

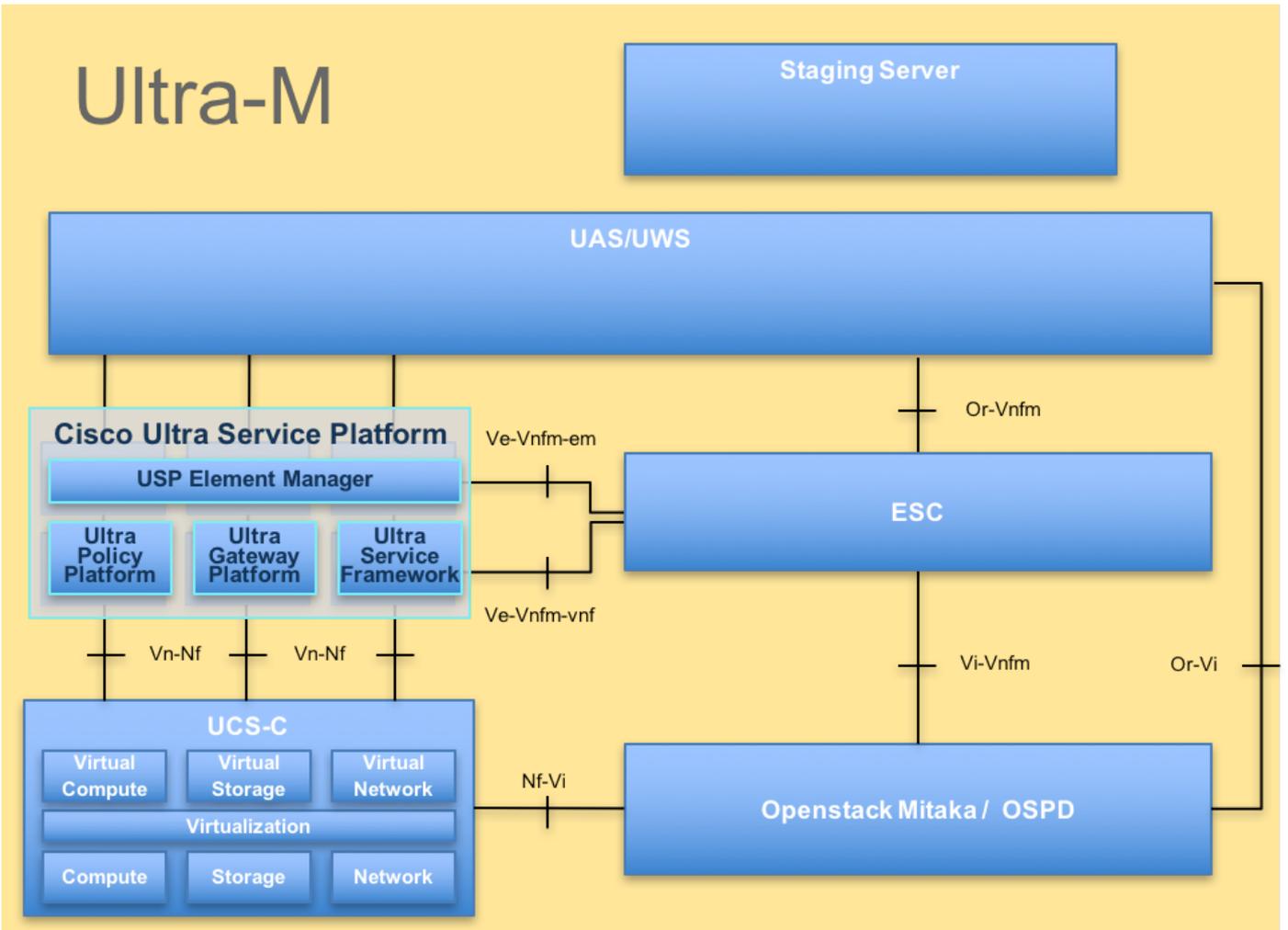
ESC

Or-Vnfm

Vi-Vnfm

Openstack Mitaka / OSPD

Or-Vi



참고: 이 문서의 절차를 정의하기 위해 Ultra M 5.1.x 릴리스가 고려됩니다. 이 문서는 Cisco Ultra-M 플랫폼에 대해 잘 알고 있는 Cisco 직원을 대상으로 합니다.

약어

VNF	가상 네트워크 기능
Esc 키	Elastic Service Controller
자루걸레	절차 방법
OSD	개체 스토리지 디스크
HDD	하드 디스크 드라이브
SSD	SSD(Solid State Drive)
빔	가상 인프라 관리자
VM	가상 머신
UUID	보편적으로 고유한 식별자

백업 절차

OSPD 백업

1. OpenStack 스택 및 노드 목록의 상태를 확인합니다.

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

2. 모든 언더클라우드 서비스가 OSP-D 노드에서 로드됨, 활성 및 실행 중인지 확인합니다.

```
[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
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)
openstack-glance-registry.service	loaded	active	running	OpenStack Image Service (code-named GLANCE)
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)
openstack-zaqar@1.service	loaded	active	running	OpenStack Message Queuing Service (code-named Zaqar)

```
openvswitch.service                                loaded active exited Open vSwitch
```

LOAD = Reflects whether the unit definition was properly loaded.

ACTIVE = The high-level unit activation state, for example, 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. 백업 프로세스를 수행하기 전에 사용 가능한 디스크 공간이 충분한지 확인합니다. 이 tarball은 최소 3.5GB로 예상됩니다.

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

4. 루트 사용자로 이러한 명령을 실행하여 언더클라우드 노드의 데이터를 undercloud-backup-[timestamp].tar.gz라는 이름의 파일에 백업한 다음 백업 서버에 전송합니다.

```
[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  
/etc/my.cnf.d/server.cnf /var/lib/glance/images /srv/node /home/stack
```

```
tar: Removing leading `/' from member names
```

ESC 백업

1. ESC를 누르면 VIM과 상호 작용하여 VNF(Virtual Network Function)가 나타납니다.

2. ESC는 Ultra-M 솔루션에서 1:1 이중화를 제공합니다. 구축된 ESC VM이 2개 있으며 Ultra-M에서 단일 장애를 지원합니다. 예를 들어 시스템에 단일 장애가 있는 경우 시스템을 복구합니다.

참고: 단일 오류 이상인 경우 지원되지 않으며 시스템을 재구축해야 할 수 있습니다.

ESC 백업 세부 정보:

- 실행 중인 컨피그레이션
- 회의자료 CDB DB
- ESC 로그
- Syslog 컨피그레이션

3. ESC DB 백업의 빈도는 까다롭고 ESC가 구축된 다양한 VNF VM의 다양한 상태 시스템을 모니터링하고 유지 관리하므로 신중하게 처리해야 합니다. 이러한 백업은 지정된 VNF/POD/사이트에서 이러한 작업 후에 수행되는 것이 좋습니다.

4. health.sh 스크립트를 사용하여 ESC의 상태가 양호한지 확인합니다.

```
[root@auto-test-vnfm1-esc-0 admin]# escadm status
0 ESC status=0 ESC Primary 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 (Primary) with DRBD =====
```

```
DRBD_ROLE_CHECK=0
MNT_ESC_DATABASE_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. 실행 중인 컨피그레이션의 백업을 가져와서 백업 서버에 파일을 전송합니다.

```
[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
```

ESC 데이터베이스 백업

1. ESC VM에 로그인하고 백업을 수행하기 전에 이 명령을 실행합니다.

```
[admin@esc ~]# sudo bash
[root@esc ~]# cp /opt/cisco/esc/esc-scripts/esc_dbtool.py /opt/cisco/esc/esc-scripts/esc_dbtool.py.bkup
[root@esc esc-scripts]# sudo sed -i "s,'pg_dump','usr/pgsq1-9.4/bin/pg_dump,'" /opt/cisco/esc/esc-script

#Set ESC to mainenance mode
[root@esc esc-scripts]# escadm op_mode set --mode=maintenance
```

2. ESC 모드를 선택하고 유지 보수 모드에 있는지 확인합니다.

```
[root@esc esc-scripts]# escadm op_mode show
```

3. ESC에서 사용할 수 있는 데이터베이스 백업 복원 도구를 사용하여 데이터베이스를 백업합니다.

```
[root@esc scripts]# sudo /opt/cisco/esc/esc-scripts/esc_dbtool.py backup --file scp://<username>:<password>
```

4. ESC를 작동 모드로 설정하고 모드를 확인합니다.

```
[root@esc scripts]# escadm op_mode set --mode=operation
```

```
[root@esc scripts]# escadm op_mode show
```

5. scripts 디렉토리로 이동하여 로그를 수집합니다.

```
[root@esc scripts]# /opt/cisco/esc/esc-scripts
```

```
sudo ./collect_esc_log.sh
```

6. ESC의 스냅샷을 생성하려면 먼저 ESC를 종료합니다.

```
shutdown -r now
```

7. OSPD에서 이미지 스냅샷을 만듭니다.

```
nova image-create --poll esc1 esc_snapshot_27aug2018
```

8. 스냅샷이 생성되었는지 확인합니다.

```
openstack image list | grep esc_snapshot_27aug2018
```

9. OSPD에서 ESC를 시작합니다.

```
nova start esc1
```

10. 대기 ESC VM에서 동일한 절차를 반복하고 백업 서버에 로그를 전송합니다.

11. ESC VM에서 syslog 컨피그레이션 백업을 수집하여 백업 서버로 전송합니다.

```
[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
```

CPS 백업

1단계. CPS Cluster-Manager의 백업을 만듭니다.

nova 인스턴스를 보고 클러스터 관리자 VM 인스턴스의 이름을 기록하려면 다음 명령을 사용합니다.

```
nova list
```

Esc에서 Cluman을 중지합니다.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action STOP <vm-name>
```

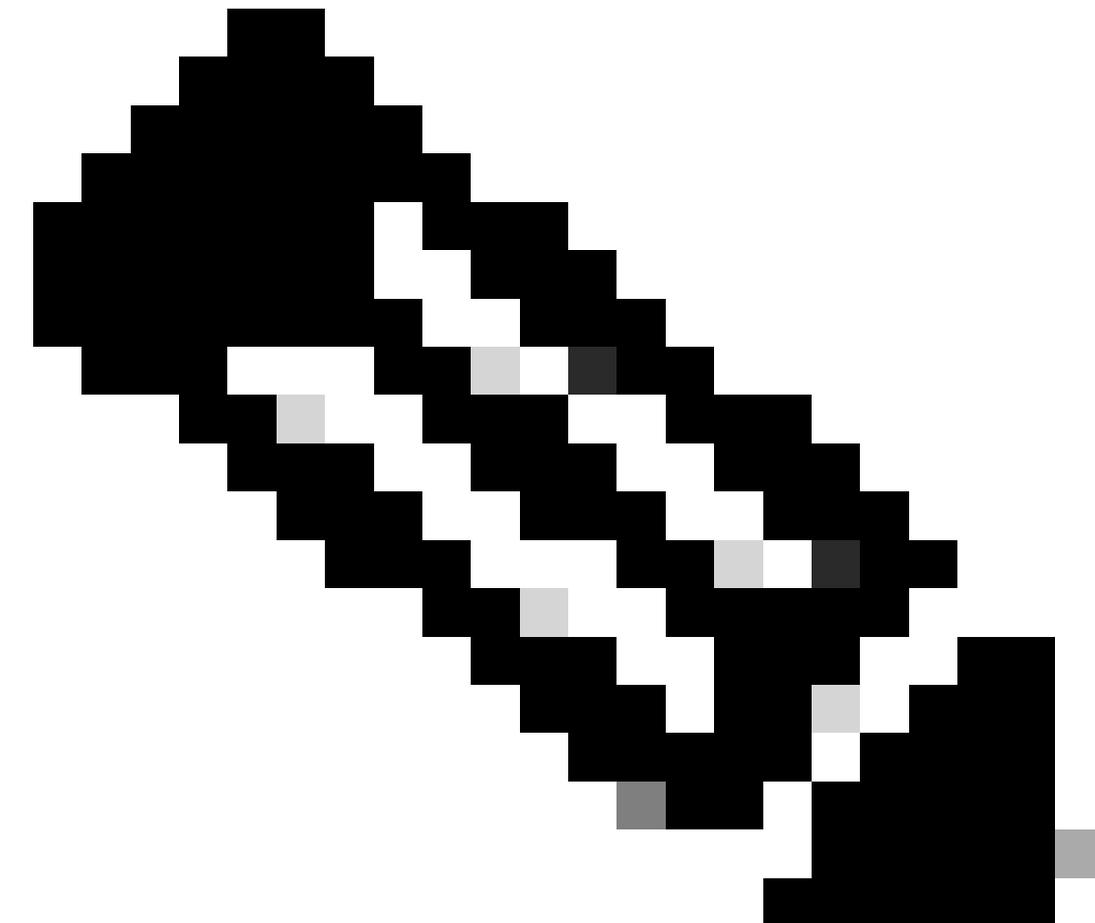
2단계. 클러스터 관리자가 차단 상태인지 확인합니다.

```
admin@esc1 ~]$ /opt/cisco/esc/confd/bin/confd_cli
```

```
admin@esc1> show esc_datamodel opdata tenants tenant Core deployments * state_machine
```

3단계. 다음 명령에 표시된 대로 nova 스냅샷 이미지를 생성합니다.

```
nova image-create --poll <cluman-vm-name> <snapshot-name>
```



참고: 스냅샷을 위한 충분한 디스크 공간이 있는지 확인하십시오.

.Important(중요) - 스냅샷 생성 후 VM에 연결할 수 없는 경우 nova list 명령을 사용하여 VM의 상태를 확인합니다. SHUTOFF 상태인 경우 VM을 수동으로 시작해야 합니다.

4단계. nova image-list 명령을 사용하여 이미지 목록을 봅니다.

이미지 1: 출력 예

ID	Name	Status	Server
146719e8-d8a0-4d5a-9b15-2a669cfab81f	CPS_10.9.9_20160803_100301_112.iso	ACTIVE	
1955d56e-4ecf-4269-b53d-b30e73ad57f0	base_vm	ACTIVE	
2bbfb51c-cd05-4b7c-ad77-8362d76578db	cluman_snapshot	ACTIVE	4842ae5a-83a3-48fd-915b-6ca6361adb2c

5단계. 스냅샷이 생성되면 스냅샷 이미지가 OpenStack Glance에 저장됩니다. 원격 데이터 저장소에 스냅샷을 저장하려면 스냅샷을 다운로드하고 OSPD의 파일을 (/home/stack/CPS_BACKUP)으로 전송합니다.

이미지를 다운로드하려면 OpenStack에서 다음 명령을 사용합니다.

```
glance image-download --file For example: glance image-download --file snapshot.raw 2bbfb51c-cd05-4b7c-
```

6단계. 이 명령에 표시된 대로 다운로드된 이미지를 나열합니다.

```
ls -ltr *snapshot*
```

```
Example output: -rw-r--r--. 1 root root 10429595648 Aug 16 02:39 snapshot.raw
```

7단계. 나중에 복원할 클러스터 관리자 VM의 스냅샷을 저장합니다.

2. 구성 및 데이터베이스를 백업합니다.

1. config_br.py -a export --all /var/tmp/backup/ATP1_backup_all_\$(date +%Y-%m-%d).tar.gz OR
2. config_br.py -a export --mongo-all /var/tmp/backup/ATP1_backup_mongoall\$(date +%Y-%m-%d).tar.gz
3. config_br.py -a export --svn --etc --grafanadb --auth-htpasswd --haproxy /var/tmp/backup/ATP1_backup
4. mongodump - /var/qps/bin/support/env/env_export.sh --mongo /var/tmp/env_export_\$(date).tgz
5. patches - cat /etc/broadhop/repositories, check which patches are installed and copy those patches to
6. backup the cronjobs by taking backup of the cron directory: /var/spool/cron/ from the Pcrfclient01/C

다른 백업이 필요한 경우 crontab-l에서 확인합니다.

모든 백업을 OSPD /home/stack/CPS_BACKUP으로 전송합니다.

3. ESC Primary에서 yaml 파일을 백업합니다.

```
/opt/cisco/esc/confd/bin/netconf-console --host 127.0.0.1 --port 830 -u <admin-user> -p <admin-password>
```

OSPD /home/stack/CPS_BACKUP에서 파일을 전송합니다.

4. crontab -l 항목을 백업합니다.

crontab -l을 사용하여 txt 파일을 만들고 원격 위치에 ftp를 보냅니다(OSPD /home/stack/CPS_BACKUP).

5. LB 및 PCRF 클라이언트에서 경로 파일을 백업합니다.

```
Collect and scp the configurations from both LBs and Pcrfclients  
route -n /etc/sysconfig/network-script/route-*
```

복원 절차

OSPD 복구

OSPD 복구 절차는 이러한 가정을 기반으로 수행됩니다.

1. OSPD 백업은 이전 OSPD 서버에서 사용할 수 있습니다.
2. OSPD 복구는 시스템의 기존 OSPD 서버를 대체하는 새 서버에서 수행할 수 있습니다..

ESC 복구

1. ESC VM은 VM이 오류 상태이거나 종료 상태일 경우 하드 리부팅을 수행하여 영향을 받은 VM을 불러오는 경우 복구할 수 있습니다. 다음 단계를 실행하여 ESC를 복구합니다.
2. ESC VM을 하드 리부팅하여 ERROR 또는 Shutdown 상태인 VM을 식별합니다. 이 예에서는 auto-test-vnfm1-ESC-0을 재부팅합니다.

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

```
| f03e3cac-a78a-439f-952b-045aea5b0d2c | auto-test-vnfm1-ESC-0
```

```
| 79498e0d-0569-4854-a902-012276740bce | auto-test-vnfm1-ESC-1
```

```
[root@tb1-baremetal scripts]# [root@tb1-baremetal scripts]# nova reboot --hard f03e3cac-a78a-439f-952b-  
Request to reboot server <Server: auto-test-vnfm1-ESC-0> has been accepted.
```

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

3. ESC VM이 삭제되어 다시 불러와야 하는 경우. 다음 단계를 사용하십시오.

```
[stack@pod1-ospd scripts]$ nova list |grep ESC-1  
| c566efbf-1274-4588-a2d8-0682e17b0d41 | vnf1-ESC-ESC-1
```

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

4. ESC VM을 복구할 수 없고 데이터베이스를 복원해야 하는 경우 이전에 수행한 백업에서 데이터베이스를 복원합니다.

5. ESC 데이터베이스 복원의 경우 데이터베이스를 복원하기 전에 esc 서비스가 중지되었는지 확인해야 합니다. ESC HA의 경우, 먼저 보조 VM에서 실행한 다음 기본 VM에서 실행합니다.

```
# service keepalived stop
```

6. ESC 서비스 상태를 확인하고 HA의 기본 및 보조 VM에서 모두 중지되었는지 확인합니다.

```
# escadm status
```

7. 스크립트를 실행하여 데이터베이스를 복원합니다. DB를 새로 만든 ESC 인스턴스로 복원하는 과정에서 이 도구는 인스턴스 중 하나를 기본 ESC로 승격하고 해당 DB 폴더를 drbd 장치에 마운트하며 PostgreSQL 데이터베이스를 시작할 수도 있습니다.

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

8. ESC 서비스를 다시 시작하여 데이터베이스 복원을 완료합니다. 두 VM에서 HA가 실행되는 경우, keepalive 서비스를 다시 시작합니다.

```
# service keepalived start
```

9. VM이 성공적으로 복원되어 실행 중인 경우, 이전에 성공한 알려진 백업에서 모든 syslog 관련 구성이 복원되었는지 확인합니다. 모든 ESC VM에서 복원되었는지 확인합니다.

```
[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
```

10. ESC를 OSPD 스냅샷에서 다시 작성해야 하는 경우 이 명령을 사용하여 백업 중에 만든 스냅샷을 사용하십시오.

```
nova rebuild --poll --name esc_snapshot_27aug2018 esc1
```

11. 재구축이 완료된 후 ESC의 상태를 확인합니다.

```
nova list --fileds name,host,status,networks | grep esc
```

12. 이 명령을 사용하여 ESC 상태를 선택합니다.

```
health.sh
```

```
Copy Datamodel to a backup file
```

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli get esc_datamodel/opdata > /tmp/esc_opdata_`date +%Y%m%d%H%M%S`.sh
```

ESC가 VM을 시작하지 못하는 경우

- 경우에 따라 ESC가 예기치 않은 상태로 인해 VM을 시작하지 못할 수 있습니다. 해결 방법은 기본 ESC를 재부팅하여 ESC 전환을 수행하는 것입니다. ESC 전환은 1분 정도 걸릴 수 있습니다. 새 주 ESC에서 health.sh를 실행하여 작동 중인지 확인합니다. ESC가 Primary가 되면 ESC는 VM 상태를 수정하고 VM을 시작할 수 있습니다. 이 작업이 예약되어 있으므로 완료될

때까지 5~7분 정도 기다려야 합니다.

- /var/log/esc/yangesc.log 및 /var/log/esc/escmanager.log을 모니터링할 수 있습니다. 5~7분 후에 VM이 복구되지 않는 경우, 사용자는 직접 영향받는 VM의 수동 복구를 수행해야 합니다.
- VM이 성공적으로 복원되어 실행 중이면 이전에 성공한 알려진 백업에서 모든 syslog 관련 컨피그레이션이 복원되었는지 확인합니다. 모든 ESC VM에서 복원되었는지 확인합니다.

```
root@abautotestvnm1em-0:/etc/rsyslog.d# pwd
/etc/rsyslog.d
```

```
root@abautotestvnm1em-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@abautotestvnm1em-0:/etc/rsyslog.d# ls /etc/rsyslog.conf
rsyslog.conf
```

CPS 복구

OpenStack에서 클러스터 관리자 VM을 복원합니다.

1단계. 다음 명령에 표시된 대로 클러스터 관리자 VM 스냅샷을 컨트롤러 블레이드에 복사합니다.

```
ls -ltr *snapshot*
```

```
Example output: -rw-r--r--. 1 root root 10429595648 Aug 16 02:39 snapshot.raw
```

2단계. 데이터 저장소에서 OpenStack에 스냅샷 이미지를 업로드합니다.

```
glance image-create --name --file --disk-format qcow2 --container-format bare
```

3단계. 다음 예제와 같이 스냅샷이 Nova 명령으로 업로드되었는지 확인합니다.

nova image-list

이미지 2: 출력 예

ID	Name	Status	Server
146719e8-d8a0-4d5a-9b15-2a669cfab81f	CPS_10.9.9_20160803_100301_112.iso	ACTIVE	
1955d56e-4ecf-4269-b53d-b30e73ad57f0	base_vm	ACTIVE	
2bbfb51c-cd05-4b7c-ad77-8362d76578db	cluman_snapshot	ACTIVE	4842ae5a-83a3-48fd-915b-6ca6361adb2c
5eebff44-658a-49a5-a170-1978f6276d18	imported_image	ACTIVE	

4단계. 클러스터 관리자 VM이 있는지 여부에 따라, 클러스터를 생성하거나 클러스터를 재구축하도록 선택할 수 있습니다.

· Cluster Manager VM 인스턴스가 없는 경우 다음 예에 표시된 대로 Heat 또는 Nova 명령을 사용하여 Cluster VM을 생성합니다.

ESC를 사용하여 클러스터 VM을 생성합니다.

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

PCRF 클러스터는 이전 명령의 도움으로 생성한 다음 config_br.py 복원으로 수행한 백업에서 클러스터 관리자 컨피그레이션을 복원하고, 백업에서 수행한 덤프에서 mongorestore를 복원할 수 있습니다.

```
delete - nova boot --config-drive true --image "" --flavor "" --nic net-id=",v4-fixed-ip=" --nic net-id
```

· Cluster Manager VM 인스턴스가 있는 경우, 다음과 같이 nova rebuild 명령을 사용하여 업로드된 스냅샷으로 Cluman VM 인스턴스를 재구축합니다.

```
nova rebuild <instance_name> <snapshot_image_name>
```

예를 들면 다음과 같습니다.

```
nova rebuild cps-cluman-5f3tujqvbi67 cluman_snapshot
```

5단계. 표시된 대로 모든 인스턴스를 나열하고 새 클러스터 관리자 인스턴스가 생성되어 실행 중인

지 확인합니다.

nova list

이미지 3. 출력 예

ID	Name	Status	Task State	Power State	Networks
ac3d2dbc-7b0e-4df4-a690-7f84ca3032bd	cluman	ACTIVE	-	Running	management=172.20.67.34; internal=172.20.70.34

시스템에서 최신 패치를 복원합니다.

1. Copy the patch files to cluster manager which were backed up in OSPD /home/stack/CPS_BACKUP
2. Login to the Cluster Manager as a root user.
3. Untar the patch by executing this command: tar -xvzf [patch name].tar.gz
4. Edit /etc/broadhop/repositories and add this entry: file:/// \$path_to_the plugin/[component name]
5. Run build_all.sh script to create updated QPS packages: /var/qps/install/current/scripts/build_all.sh
6. Shutdown all software components on the target VMs: runonall.sh sudo monit stop all
7. Make sure all software components are shutdown on target VMs: statusall.sh

 참고: 소프트웨어 구성 요소는 모두 현재 상태로 Not Monitored(모니터링되지 않음)로 표시되어야 합니다.

8. Update the qns VMs with the new software using reinit.sh script: /var/qps/install/current/scripts/reinit.sh
9. Restart all software components on the target VMs: runonall.sh sudo monit start all
10. Verify that the component is updated, run: about.sh

Cronjobs를 복원합니다.

1. 백업된 파일을 OSPD에서 Cluman/Pcrfclient01로 이동합니다.
2. 명령을 실행하여 백업에서 cronjob을 활성화합니다.

<#root>

#crontab Cron-backup

3. 이 명령으로 cronjobs가 활성화되었는지 확인합니다.

```
#crontab -l
```

클러스터의 개별 VM을 복원합니다.

pcrfclient01 VM을 재구축하려면

1단계. 클러스터 관리자 VM에 루트 사용자로 로그인합니다.

2단계. 다음 명령을 사용하여 SVN 리포지토리의 UUID를 기억하십시오.

```
<#root>
```

```
svn info http://pcrfclient02/repos | grep UUID
```

이 명령은 저장소의 UUID를 출력할 수 있습니다.

예: 저장소 UUID: ea50bbd2-5726-46b8-b807-10f4a7424f0e

3단계. 다음 예에 표시된 대로 클러스터 관리자에서 백업 정책 구성기 컨피그레이션 데이터를 가져옵니다.

```
<#root>
```

```
config_br.py -a import --etc-oam --svn --stats --grafanadb --auth-htpasswd --users /mnt/backup/oam_backu
```

 참고: 많은 배포에서 정기적으로 컨피그레이션 데이터를 백업하는 cron 작업을 실행합니다. 자세한 내용은 하위 버전 저장소 백업을 참조하십시오.

4단계. 최신 컨피그레이션을 사용하여 클러스터 관리자에서 VM 아카이브 파일을 생성하려면 다음 명령을 실행합니다.

```
<#root>
```

```
/var/qps/install/current/scripts/build/build_svn.sh
```

5단계. pcrfclient01 VM을 구축하려면 다음 중 하나를 수행합니다.

OpenStack에서 HEAT 템플릿 또는 Nova 명령을 사용하여 VM을 다시 생성합니다. 자세한 내용은 OpenStack용 CPS 설치 가이드를 참조하십시오.

6단계. 이러한 일련의 명령을 실행하여 pcrfclient01을 기본으로 하여 pcrfclient01과 pcrfclient02 간의 SVN 기본/보조 동기화를 다시 설정합니다.

SVN이 이미 동기화된 경우 이러한 명령을 실행하지 마십시오.

SVN이 동기화되었는지 확인하려면 pcrfclient02에서 이 명령을 실행합니다.

값이 반환되면 SVN이 이미 동기화됩니다.

<#root>

```
/usr/bin/svn propget svn:sync-from-url --revprop -r0 http://pcrfclient01/repos
```

pcrfclient01에서 다음 명령을 실행합니다.

<#root>

```
/bin/rm -fr /var/www/svn/repos
```

```
/usr/bin/svnadmin create /var/www/svn/repos
```

```
/usr/bin/svn propset --revprop -r0 svn:sync-last-merged-rev 0 http://pcrfclient02/repos-proxy-sync
```

```
/usr/bin/svnadmin setuuid /var/www/svn/repos/ "Enter the UUID captured in step 2"
```

```
/etc/init.d/vm-init-client /
```

```
var/qps/bin/support/recover_svn_sync.sh
```

7단계. pcrfclient01도 종재자 VM인 경우 다음 단계를 실행합니다.

a) 시스템 컨피그레이션을 기반으로 mongodb 시작/중지 스크립트를 생성합니다. 모든 구축에 이러한 데이터베이스가 모두 구성된 것은 아닙니다.

 참고: 어떤 데이터베이스를 설정해야 하는지 확인하려면 /etc/broadhop/mongoConfig.cfg을 참조하십시오.

<#root>

```
cd /var/qps/bin/support/mongo
```

```
build_set.sh --session --create-scripts
```

```
build_set.sh --admin --create-scripts
```

```
build_set.sh --spr --create-scripts
build_set.sh --balance --create-scripts
build_set.sh --audit --create-scripts
build_set.sh --report --create-scripts
```

b) mongo 프로세스를 시작합니다.

<#root>

```
/usr/bin/systemctl start sessionmgr-XXXXX
```

c) 종재자가 시작될 때까지 기다린 다음 diagnostics.sh --get_replica_status를 실행하여 복제본 세트의 상태를 확인합니다.

pcrfclient02 VM을 재구축하려면

1단계. 클러스터 관리자 VM에 루트 사용자로 로그인합니다.

2단계. 최신 컨피그레이션을 사용하여 클러스터 관리자에서 VM 아카이브 파일을 생성하려면 다음 명령을 실행합니다.

<#root>

```
/var/qps/install/current/scripts/build/build_svn.sh
```

3단계. pcrfclient02 VM을 구축하려면 다음 중 하나를 수행합니다.

OpenStack에서 HEAT 템플릿 또는 Nova 명령을 사용하여 VM을 다시 생성합니다. 자세한 내용은 OpenStack용 CPS 설치 가이드를 참조하십시오.

4단계. pcrfclient01에 대한 보안 셀:

<#root>

```
ssh pcrfclient01
```

5단계. 다음 스크립트를 실행하여 pcrfclient01에서 SVN 재배포를 복구합니다.

<#root>

```
/var/qps/bin/support/recover_svn_sync.sh
```

sessionmgr VM을 재배포하려면

1단계. 클러스터 관리자 VM에 루트 사용자로 로그인합니다.

2단계. sessionmgr VM을 배포하고 실패하거나 손상된 VM을 교체하려면 다음 중 하나를 수행합니다.

OpenStack에서 HEAT 템플릿 또는 Nova 명령을 사용하여 VM을 다시 생성합니다. 자세한 내용은 OpenStack용 CPS 설치 가이드를 참조하십시오.

3단계. 시스템 컨피그레이션을 기반으로 mongodb 시작/중지 스크립트를 생성합니다.

모든 구축에 이러한 데이터베이스가 모두 구성된 것은 아닙니다. 설정해야 하는 데이터베이스를 확인하려면 /etc/broadhop/mongoConfig.cfg을 참조하십시오.

```
cd /var/qps/bin/support/mongo
```

```
build_set.sh --session --create-scripts  
build_set.sh --admin --create-scripts  
build_set.sh --spr --create-scripts  
build_set.sh --balance --create-scripts  
build_set.sh --audit --create-scripts  
build_set.sh --report --create-scripts
```

4단계. sessionmgr VM에 대해 셸을 보안하고 mongo 프로세스를 시작합니다.

```
<#root>
```

```
ssh sessionmgrXX
```

```
/usr/bin/systemctl start sessionmgr-XXXXXX
```

5단계. 멤버가 시작되고 보조 멤버가 동기화될 때까지 기다린 다음 diagnostics.sh —get_replica_status를 실행하여 데이터베이스의 상태를 확인합니다.

6단계. 세션 관리자 데이터베이스를 복원하려면 백업이 —mongo-all 또는 —mongo 옵션으로 수행되었는지 여부에 따라 다음 명령 예 중 하나를 사용합니다.

```
<#root>
```

-

```
config_br.py -a import --mongo-all --users /mnt/backup/Name of backup
```

or

```
· config_br.py -a import --mongo --users /mnt/backup/Name of backup
```

정책 디렉터(로드 밸런서) VM을 재배포하려면

1단계. 클러스터 관리자 VM에 루트 사용자로 로그인합니다.

2단계. 클러스터 관리자에서 백업 정책 구성기 컨피그레이션 데이터를 가져오려면 다음 명령을 실행합니다.

```
config_br.py -a import --network --haproxy --users /mnt/backup/lb_backup_27102016.tar.gz
```

3단계. 최신 컨피그레이션을 사용하여 클러스터 관리자에서 VM 아카이브 파일을 생성하려면 다음 명령을 실행합니다.

<#root>

```
/var/qps/install/current/scripts/build/build_svn.sh
```

4단계. lb01 VM을 구축하려면 다음 중 하나를 수행합니다.

OpenStack에서 HEAT 템플릿 또는 Nova 명령을 사용하여 VM을 다시 생성합니다. 자세한 내용은 OpenStack용 CPS 설치 가이드를 참조하십시오.

QNS(Policy Server) VM을 재배포하려면

1단계. 클러스터 관리자 VM에 루트 사용자로 로그인합니다.

2단계. 다음 예에 표시된 대로 클러스터 관리자에서 백업 정책 구성기 컨피그레이션 데이터를 가져옵니다.

<#root>

```
config_br.py -a import --users /mnt/backup/qns_backup_27102016.tar.gz
```

3단계. 최신 컨피그레이션을 사용하여 클러스터 관리자에서 VM 아카이브 파일을 생성하려면 다음

명령을 실행합니다.

```
<#root>
```

```
/var/qps/install/current/scripts/build/build_svn.sh
```

4단계. qns VM을 구축하려면 다음 중 하나를 수행합니다.

OpenStack에서 HEAT 템플릿 또는 Nova 명령을 사용하여 VM을 다시 생성합니다. 자세한 내용은 OpenStack용 CPS 설치 가이드를 참조하십시오.

데이터베이스 복원의 일반 절차.

1단계. 이 명령을 실행하여 데이터베이스를 복원합니다.

```
<#root>
```

```
config_br.py -a import --mongo-all /mnt/backup/backup_$(date +%Y%m%d%H%M%S).tar.gz where $date is the timestamp when the backup was taken
```

예를 들면 다음과 같습니다.

```
<#root>
```

```
config_br.py -a import --mongo-all /mnt/backup/backup_27092016.tgz
```

2단계. 데이터베이스에 로그인하여 데이터베이스가 실행 중이고 액세스할 수 있는지 확인합니다.

1. 세션 관리자에 로그인합니다.

```
<#root>
```

```
mongo --host sessionmgr01 --port $port
```

여기서 \$port는 확인할 데이터베이스의 포트 번호입니다. 예를 들어 27718은 기본 Balance 포트입니다.

2. 다음 명령을 실행하여 데이터베이스를 표시합니다.

```
<#root>
```

```
show dbs
```

3. 다음 명령을 실행하여 mongo 셸을 데이터베이스로 전환합니다.

```
<#root>
```

```
use $db
```

여기서 \$db는 이전 명령에 표시된 데이터베이스 이름입니다.

use 명령은 mongo 셸을 해당 데이터베이스로 전환합니다.

예를 들면 다음과 같습니다.

```
<#root>
```

```
use balance_mgmt
```

4. 모음을 표시하려면 다음 명령을 실행합니다.

```
<#root>
```

```
show collections
```

5. 모음의 레코드 수를 표시하려면 다음 명령을 실행합니다.

```
<#root>
```

```
db.$collection.count()
```

For example, db.account.count()

앞의 예에서는 잔액 데이터베이스(balance_mgmt)의 수집 계정에 있는 레코드 수를 표시할 수 있습니다.

하위 버전 저장소 복원.

백업에서 정책 작성기 컨피그레이션 데이터를 복원하려면 다음 명령을 실행합니다.

```
<#root>
```

```
config_br.py -a import --svn /mnt/backup/backup_$(date +%Y%m%d).tgz where, $(date +%Y%m%d) is the date when the cron created
```

Grafana 대시보드를 복원합니다.

다음 명령을 사용하여 Grafana 대시보드를 복원할 수 있습니다.

```
<#root>
```

```
config_br.py -a import --grafanadb /mnt/backup/
```

복원 유효성을 검사하는 중입니다.

데이터를 복원한 후 다음 명령을 실행하여 작업 시스템을 확인합니다.

```
<#root>
```

```
/var/qps/bin/diag/diagnostics.sh
```

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.