

各种Ultra-M组件的备份和恢复过程- CPS

目录

[简介](#)

[背景信息](#)

[缩写](#)

[备份过程](#)

[OSPD备份](#)

[ESC备份](#)

[CPS备份](#)

[恢复程序](#)

[OSPD恢复](#)

[ESC恢复](#)

[CPS恢复](#)

简介

本文档介绍在托管调用CPS虚拟网络功能的Ultra-M设置中备份和恢复虚拟机所需的步骤。

背景信息

Ultra-M是经过预先封装和验证的虚拟化移动数据包核心解决方案，旨在简化虚拟网络功能(VNF)的部署。Ultra-M解决方案包括以下虚拟机(VM)类型：

- 弹性服务控制器(ESC)
- 思科策略套件(CPS)

Ultra-M的高级体系结构及其涉及的组件如下图所示。

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

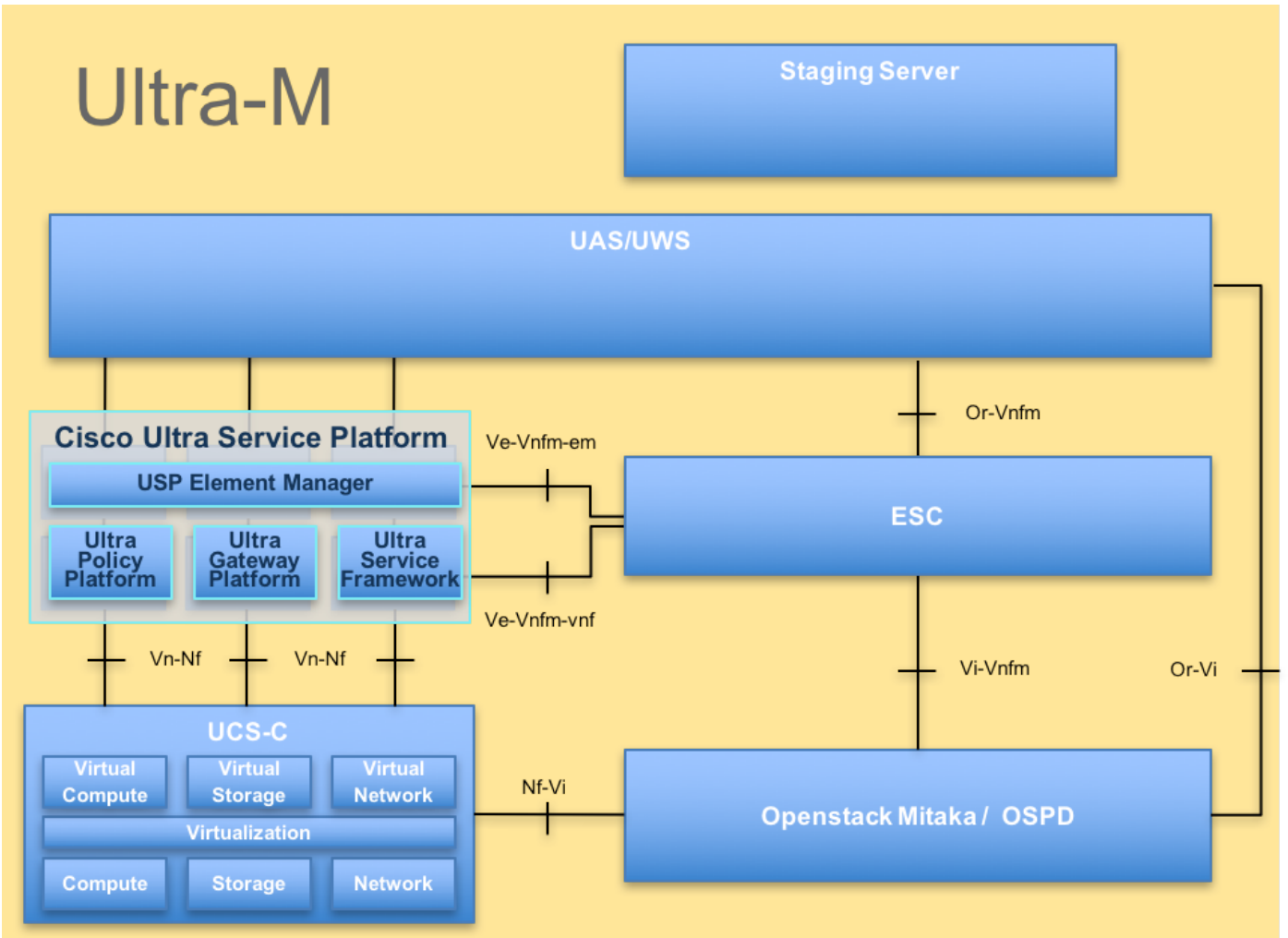
ESC

Or-Vnfm

Vi-Vnfm

Openstack Mitaka / OSPD

Or-Vi



注意：在定义本文档中的过程时需要考虑Ultra M 5.1.x版本。本文档面向熟悉Cisco Ultra-M平台的思科人员。

缩写

VNF	虚拟网络功能
ESC	弹性服务控制器
MOP	程序方法
OSD	对象存储磁盘
HDD	硬盘驱动器
SSD	固态硬盘
VIM	虚拟基础设施管理器
VM	虚拟机
UUID	通用唯一标识符

备份过程

OSPD备份

1. 检查OpenStack堆栈和节点列表的状态。

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

2. 从OSP-D节点检查所有云服务是否处于已加载、活动和运行状态。

```
[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. 在执行备份过程之前，请确认您有足够的可用磁盘空间。此tarball应至少为3.5 GB。

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

4. 以root用户身份执行这些命令，将数据从undercloud节点备份到名为undercloud-backup-[timestamp].tar.gz的文件中，然后传输到备份服务器。

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

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

ESC备份

1. ESC反过来通过与VIM交互来启用虚拟网络功能(VNF)。

2. ESC在Ultra-M解决方案中具有1:1冗余。部署了2个ESC虚拟机，并且支持Ultra-M中的单个故障。例如，如果系统中出现单个故障，则恢复系统。



注意：如果出现多个故障，则不支持这些故障，可能需要重新部署系统。

ESC备份详细信息：

- 运行配置
- ConfD CDB数据库
- ESC日志
- 系统日志配置

3. ESC数据库备份的频率很棘手，需要在ESC监视和维护所部署的各种VNF虚拟机的各种状态机时谨慎处理。建议这些备份是在给定的VNF/POD/站点中的这些活动之后执行的。

4. 使用health.sh脚本验证ESC的运行状况是否良好。

```
[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. 备份运行配置并将文件传输到备份服务器。

```
[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
```

备份ESC数据库

1. 登录ESC VM并执行此命令，然后进行备份。

```
[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/pgsql-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. 检查ESC模式并确保其处于维护模式。

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

3. 使用ESC中提供的数据库备份还原工具备份数据库。

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

4. 将ESC设置回操作模式并确认模式。

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

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

5. 导航至脚本目录并收集日志。

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

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

6. 要创建ESC的快照，请先关闭ESC。

```
shutdown -r now
```

7. 从OSPD创建映像快照。

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


8. 验证是否已创建快照。

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

9. 从OSPD启动ESC。

```
nova start esc1
```

10. 在备用ESC VM上重复相同的步骤并将日志传输到备份服务器。

11. 收集两个ESC VMS上的系统日志配置备份，并将它们传输到备份服务器。

```
[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
```

CPS备份

步骤1:创建CPS Cluster-Manager的备份。

使用此命令可以查看nova实例并记下集群管理器VM实例的名称：

```
nova list
```

从ESC停止Cluman。

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

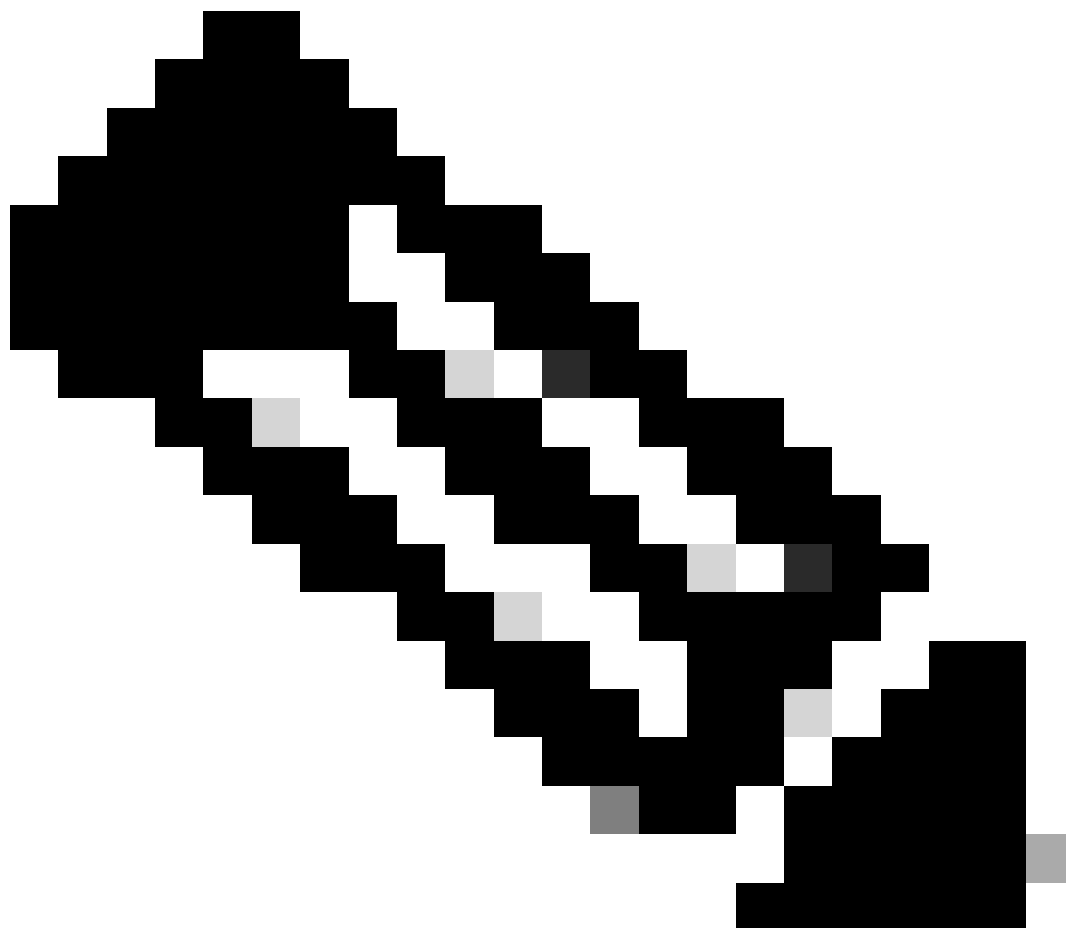
第二步：验证Cluster Manager是否处于关闭状态。

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

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

第三步：如下命令所示，创建nova快照映像：

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



注意：请确保您有足够的磁盘空间用于快照。

.重要信息-如果创建快照后无法访问VM，请使用nova list命令检查VM的状态。如果它处于关闭状态，则需要手动启动VM。

第四步：使用以下命令查看映像列表：nova image-list

图1：输出示例

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

第五步：创建快照后，快照映像将存储在OpenStack Glance中。要将快照存储在远程数据存储中，请下载快照并将文件以OSPD形式传输到(/home/stack/CPS_BACKUP)。

要下载映像，请在OpenStack中使用以下命令：

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

第六步：列出下载的映像，如以下命令所示：

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

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

步骤 7. 存储Cluster Manager VM的快照，以便将来恢复。

2. 备份配置和数据库。

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

从crontab-l中验证是否需要任何其他备份。

将所有备份传输到OSPD /home/stack/CPS_BACKUP。

3. 从ESC主映像备份yaml文件。

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

在OSPD /home/stack/CPS_BACKUP中传输文件。

4. 备份crontab -l条目。

使用crontab -l创建一个txt文件，然后将其ftp到远程位置（在OSPD /home/stack/CPS_BACKUP中）。

5. 从LB和PCRF客户端备份路由文件。

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

恢复程序

OSPD恢复

OSPD恢复程序基于这些假设执行。

1. OSPD备份可从旧OSPD服务器获得。
2. 可以在新服务器上执行OSPD恢复，即替换系统中的旧OSPD服务器。

ESC恢复

1. 如果VM处于错误或关闭状态，请通过硬重启来启动受影响的VM，ESC可以恢复VM。执行这些步骤以恢复ESC。
2. 确定处于ERROR或Shutdown状态的VM，确定后，请硬重新启动ESC VM。在本示例中，您将重新启动auto-test-vnfm1-ESC-0。

```
[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. 如果删除ESC VM并需要再次启动。按照以下步骤顺序执行。

```
[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. 如果ESC VM不可恢复并且需要恢复数据库，请从以前备份中恢复数据库。

5. 对于ESC数据库恢复，在恢复数据库之前，您必须确保esc服务已停止；对于ESC HA，请首先在辅助VM中执行，然后在主VM中执行。

```
# service keepalived stop
```

6. 检查ESC服务状态，确保在HA的主和辅助VM中都停止了所有操作。

```
# escadm status
```

7. 执行脚本以还原数据库。作为将数据库恢复到新创建的ESC实例的一部分，该工具还可以将其中一个实例升级为主用ESC，将其数据库文件夹装载到drbd设备，并且可以启动PostgreSQL数据库。

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

8. 重新启动ESC服务以完成数据库恢复。对于在两个虚拟机中执行的HA，请重新启动keepalive服务。

```
# service keepalived start
```

9. 成功还原并运行虚拟机后；请确保从以前成功的已知备份还原所有系统日志特定配置。请确保在所有ESC虚拟机中还原该配置。

```
[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. 如果要从OSPD快照重建ESC，请使用此命令并在备份过程中使用快照。

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

11. 在重建完成后，检查ESC的状态。

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

12. 使用此命令检查ESC运行状况。

```
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`.log
```

当ESC无法启动虚拟机时

- 在某些情况下，由于意外状态，ESC可能无法启动VM。解决方法是重新启动主ESC以执行ESC切换。Esc切换可能需要一分钟时间。在新的Primary ESC上执行health.sh以验证其是否已启动。当ESC变为主时，ESC可以修复虚拟机状态并启动虚拟机。由于此操作已安排，您必须等待5-7分钟才能完成。
- 您可以监控/var/log/esc/yangesc.log和/var/log/esc/escmanager.log。如果您在5-7分钟后仍未看到恢复VM，则用户需要手动恢复受影响的VM。
- VM成功恢复并运行之后；请确保从以前成功的已知备份恢复所有系统日志特定配置。确保它在所有ESC虚拟机中恢复

```
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恢复

在OpenStack中恢复群集管理器虚拟机。

步骤1:将集群管理器VM快照复制到控制器刀片，如以下命令所示：

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

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

第二步：将快照映像从Datastore上传到OpenStack：

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

第三步：验证是否已使用Nova命令上传快照，如以下示例所示：

```
nova image-list
```

图2：输出示例

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	

第四步：根据群集管理器VM是否存在，您可以选择创建群集或重建群集：

- 如果Cluster Manager VM实例不存在，请使用Heat或Nova命令创建集群虚拟机，如下例所示：
使用ESC创建集群虚拟机。

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

PCRF集群可以在先前命令的帮助下生成，然后从使用config_br.py restore进行的备份中恢复集群管理器配置，从备份中进行的转储中恢复mongorestore。

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

- 如果Cluster Manager VM实例存在，请使用nova rebuild命令重建具有已上传快照的Cluster VM实例，如下所示：

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

例如：

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

第五步：列出所有实例（如图所示），并验证新的集群管理器实例已创建且正在运行：

```
nova list
```

图 3.输出示例

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

还原系统上的最新修补程序。

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`

 注意：软件组件必须全部显示为“未监控”作为当前状态。

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`

恢复Cronjobs。

1. 将备份文件从OSPD移动到Cluman/Pcrfclient01。
2. 运行命令以从备份中激活cronjob。

```
<#root>
```

```
#crontab Cron-backup
```

- 3.检查此命令是否已激活cronjobs。

```
#crontab -l
```

恢复集群中的单个虚拟机。

重新部署pcrfclient01 VM的步骤：

步骤1:以root用户身份登录群集管理器虚拟机。

第二步：使用以下命令记住SVN存储库的UUID：


```
<#root>  
svn info http://pcrfclient02/repos | grep UUID
```

命令可以输出存储库的UUID。

例如：存储库UUID：ea50bbd2-5726-46b8-b807-10f4a7424f0e

第三步：在群集管理器上导入备份策略生成器配置数据，如本示例所示：

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

 注意：许多部署运行定期备份配置数据的cron作业。有关详细信息，请参阅Subversion存储库备份。

第四步：要使用最新配置在集群管理器上生成VM存档文件，请执行以下命令：

```
<#root>  
/var/qps/install/current/scripts/build/build_svn.sh
```

第五步：要部署pcrfclient01 VM，请执行以下操作之一：

在OpenStack中，使用HEAT模板或Nova命令重新创建VM。有关详细信息，请参阅《OpenStack CPS安装指南》。

第六步：通过执行以下系列命令，在pcrfclient01和pcrfclient02之间重新建立SVN主/辅助同步，其中pcrfclient01为主同步。

如果SVN已同步，请不要发出这些命令。

要检查SVN是否同步，请从pcrfclient02运行此命令。

如果返回值，则SVN已同步：

```
<#root>
```

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

从pcrfclient01执行以下命令：

```
<#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
```

步骤 7.如果pcrfclient01也是仲裁器VM，请执行以下步骤：

a)根据系统配置创建mongodb启动/停止脚本。并非所有部署都配置了所有这些数据库。



注意：请参阅/etc/broadhop/mongoConfig.cfg以确定需要设置哪些数据库。

```
<#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)启动蒙戈进程：

```
<#root>
```

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

c)等待仲裁器启动，然后运行diagnostics.sh —get_replica_status检查副本集的运行状况。

重新部署pcrfclient02 VM的步骤：

步骤1:以root用户身份登录群集管理器虚拟机。

第二步：要使用最新配置在集群管理器上生成VM存档文件，请执行以下命令：

```
<#root>
```

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

第三步：要部署pcrfclient02 VM，请执行以下操作之一：

在OpenStack中，使用HEAT模板或Nova命令重新创建VM。有关详细信息，请参阅《OpenStack CPS安装指南》。

第四步：将Secure Shell连接到pcrfclient01：

```
<#root>
```

```
ssh pcrfclient01
```

第五步：运行此脚本以从pcrfclient01恢复SVN回执：

```
<#root>
```

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

要重新部署sessionmgr VM，请执行以下操作：

步骤1:以root用户身份登录群集管理器虚拟机。

第二步：要部署sessionmgr VM并替换故障或损坏的VM，请执行以下操作之一：

在OpenStack中，使用HEAT模板或Nova命令重新创建VM。有关详细信息，请参阅《OpenStack CPS安装指南》。

第三步：根据系统配置创建mongodb启动/停止脚本。

并非所有部署都配置了所有这些数据库。请参阅/etc/broadhop/mongoConfig.cfg以确定需要设置哪些数据库。

```
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
```

第四步：将Shell安全连接到sessionmgr VM并启动mongo进程：

```
<#root>
```

```
ssh sessionmgrXX
```

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

第五步：等待成员启动和辅助成员同步，然后运行diagnostics.sh —get_replica_status检查数据库的运行状况。

第六步：要恢复会话管理器数据库，请使用以下示例命令之一，具体取决于是使用—mongo-all还是—mongo选项执行备份：

```
<#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
```

要重新部署策略导向器（负载均衡器）VM，请执行以下操作：

步骤1:以root用户身份登录群集管理器虚拟机。

第二步：要在集群管理器上导入备份策略生成器配置数据，请执行以下命令：

```
config_br.py -a import --network --haproxy --users /mnt/backup/1b_backup_27102016.tar.gz
```

第三步：要使用最新配置在集群管理器上生成VM存档文件，请执行以下命令：

```
<#root>
```

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

第四步：要部署lb01 VM，请执行以下操作之一：

在OpenStack中，使用HEAT模板或Nova命令重新创建VM。有关详细信息，请参阅《OpenStack CPS安装指南》。

要重新部署策略服务器(QNS) VM，请执行以下操作：

步骤1:以root用户身份登录群集管理器虚拟机。

第二步：在群集管理器上导入备份策略生成器配置数据，如本示例所示：

```
<#root>
```

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

第三步：要使用最新配置在集群管理器上生成VM存档文件，请执行以下命令：

```
<#root>
```

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

第四步：要部署qns VM，请执行以下操作之一：

在OpenStack中，使用HEAT模板或Nova命令重新创建VM。有关详细信息，请参阅《OpenStack CPS安装指南》。

数据库恢复的一般过程。

步骤1:执行以下命令以还原数据库：

```
<#root>
```

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

例如，

```
<#root>
```

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

第二步：登录数据库，验证数据库是否正在运行并且可以访问：

1. 登录会话管理器：

```
<#root>
```

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

其中\$port是要检查的数据库的端口号。例如，27718是默认平衡端口。

2. 通过执行以下命令显示数据库：

```
<#root>
```

```
show dbs
```

3. 通过执行以下命令将mongo shell切换到数据库：

```
<#root>
```

```
use $db
```

其中\$db是显示在上一个命令中的数据库名称。

use命令将mongo shell交换到该数据库。

例如，

```
<#root>
```

```
use balance_mgmt
```

4. 要显示集合，请执行以下命令：

```
<#root>
```

```
show collections
```

5. 要显示集合中的记录数，请执行以下命令：

```
<#root>
```

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

For example, db.account.count()

上一个示例可以显示余额数据库(balance_mgmt)中收款帐户中的记录数。

Subversion存储库恢复。

要从备份中恢复策略生成器配置数据，请执行以下命令：

```
<#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
```

恢复Grafana仪表盘。

您可以使用以下命令恢复Grafana仪表盘：

```
<#root>
```

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

正在验证恢复。

恢复数据后，通过执行以下命令验证工作系统：

```
<#root>
```

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


关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。