さまざまなUltra-Mコンポーネントのバックアップおよびリストア手順：CPS

内容

[はじめに](#)

[背景説明](#)

[省略形](#)

[バックアップ手順](#)

- [OSPDバックアップ](#)
- [ESCバックアップ](#)
- [CPSバックアップ](#)

[復元手順](#)

- [OSPDリカバリ](#)
- [ESCリカバリ](#)
- [CPSリカバリ](#)

はじめに

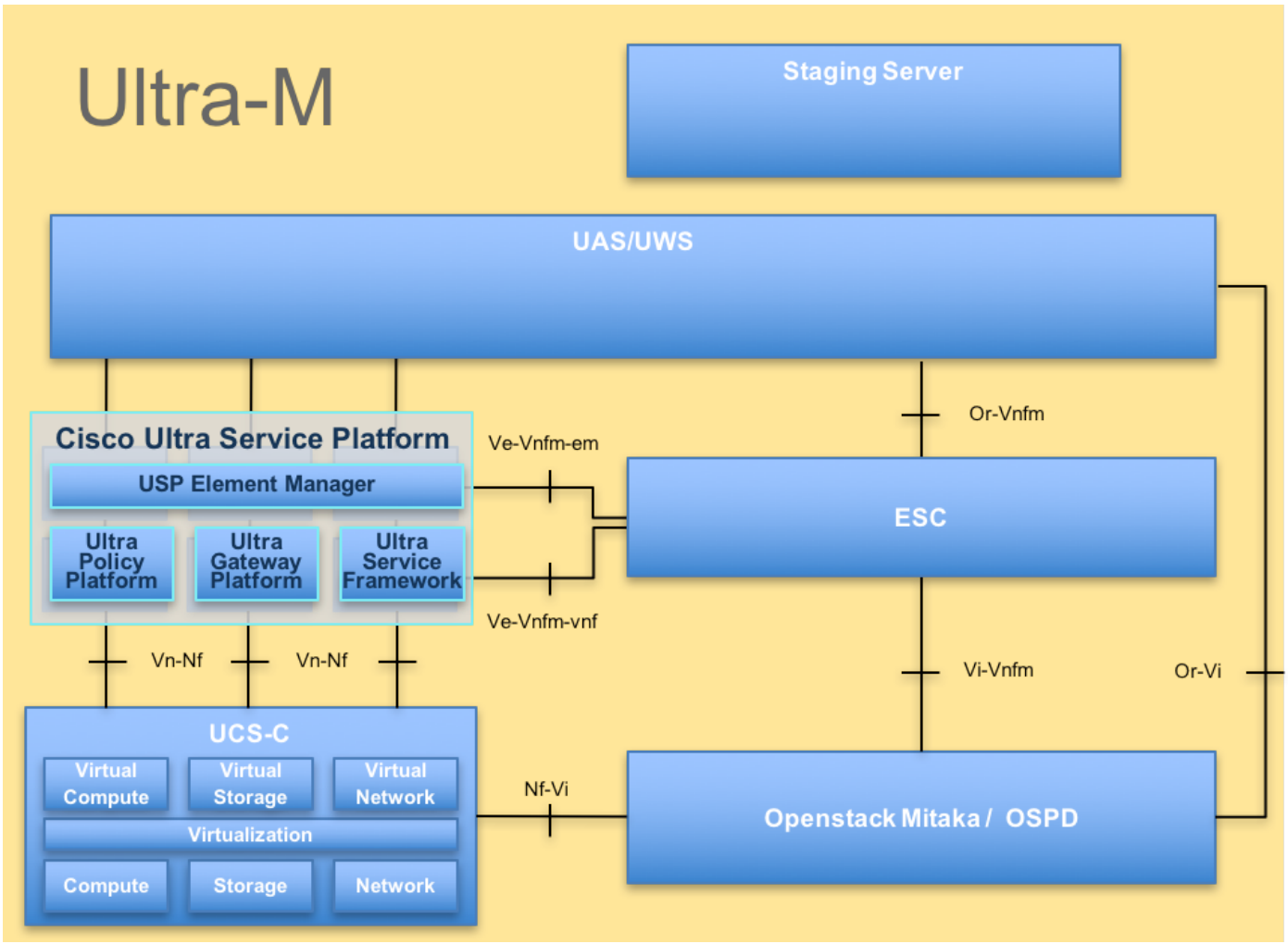
このドキュメントでは、コールCPS仮想ネットワーク機能をホストするUltra-Mセットアップで仮想マシン(VM)をバックアップおよび復元するために必要な手順について説明します。

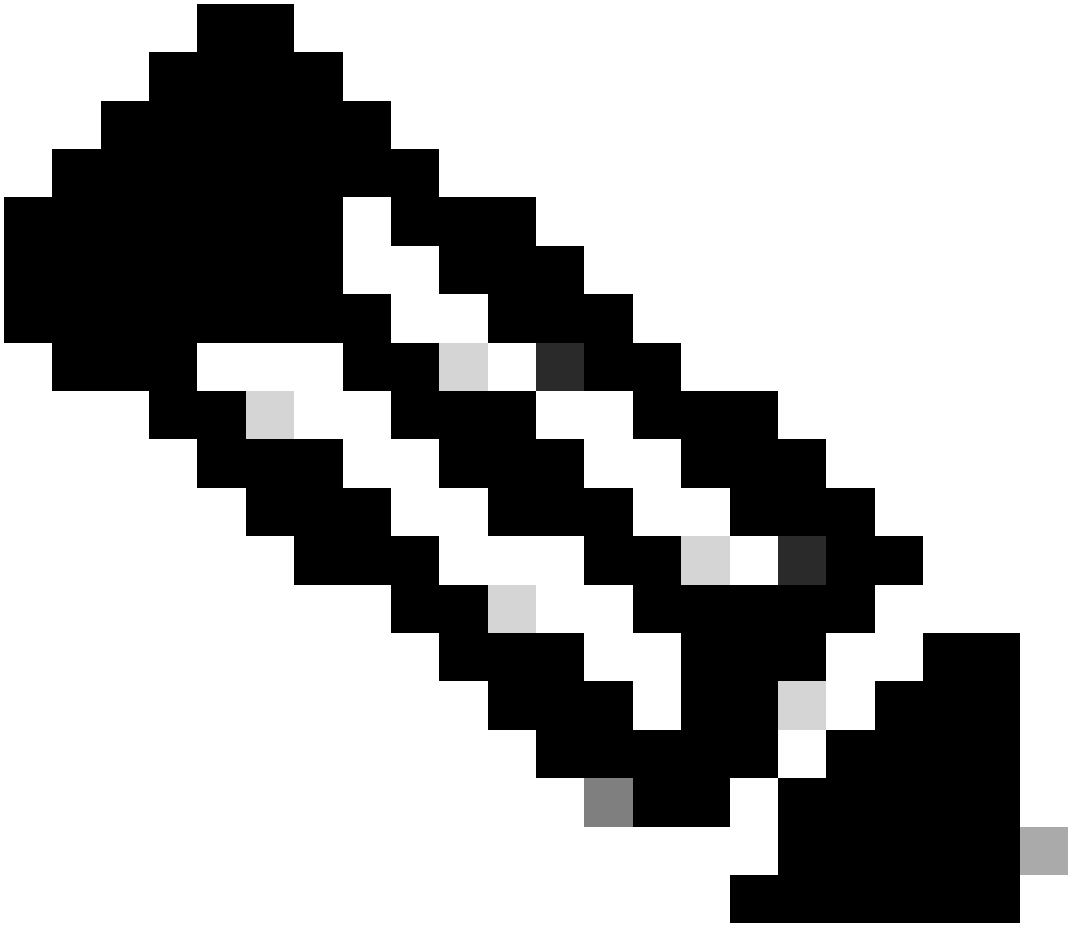
背景説明

Ultra-Mは、仮想ネットワーク機能(VNF)の導入を簡素化するように設計された、パッケージ済みで検証済みの仮想化モバイルパケットコアソリューションです。Ultra-Mソリューションは、次のタイプの仮想マシン(VM)で構成されます。

- Elastic Services Controller（ESC）
- シスコポリシースイート(CPS)

Ultra-Mのアーキテクチャの概要と関連するコンポーネントを次の図に示します。


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
Or-Vnfm
ESC
Vn-Nf
Vn-Nf
Vi-Vnfm
Or-Vi
UCS-C
Virtual Compute
Virtual Storage
Virtual Network
Virtualization
Compute
Storage
Network
Nf-Vi
Openstack Mitaka / OSPD



注：このドキュメントでは、手順を定義するためにUltra M 5.1.xリリースを考慮しています。 このドキュメントは、Cisco Ultra-Mプラットフォームに精通しているシスコの担当者を対象としています。

省略形

VNF	仮想ネットワーク機能
ESC	柔軟なサービスコントローラ
MOP	手順の方法
OSD	オブジェクトストレージディスク
HDD	ハードディスクドライブ
SSD	ソリッドステートドライブ
VIM	仮想インフラストラクチャマネージャ

仮想マシン	仮想マシン
UUID	汎用一意識別子

バックアップ手順

OSPDバックアップ

1. OpenStackスタックのステータスとノードリストを確認します。

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

2. すべてのアンダークラウドサービスが、OSP-Dノードからロード済み、アクティブ、および実行中の状態になっているかどうかを確認します。

```
[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 servic
openstack-ceilometer-notification.service   loaded active running OpenStack ceilometer notification agen
openstack-glance-api.service                loaded active running OpenStack Image Service (code-named Gl
openstack-glance-registry.service           loaded active running OpenStack Image Service (code-named Gl
openstack-heat-api-cfn.service              loaded active running Openstack Heat CFN-compatible API Serv
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 In
openstack-ironic-inspector.service          loaded active running Hardware introspection service for Ope
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) - Acc
openstack-swift-account.service             loaded active running OpenStack Object Storage (swift) - Acc
openstack-swift-container-updater.service   loaded active running OpenStack Object Storage (swift) - Con
```

```
openstack-swift-container.service          loaded active running OpenStack Object Storage (swift) - Con
openstack-swift-object-updater.service     loaded active running OpenStack Object Storage (swift) - Obj
openstack-swift-object.service             loaded active running OpenStack Object Storage (swift) - Obj
openstack-swift-proxy.service              loaded active running OpenStack Object Storage (swift) - Pro
openstack-zaqar.service                    loaded active running OpenStack Message Queuing Service (cod
openstack-zaqar@1.service                  loaded active running OpenStack Message Queuing Service (cod
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. バックアップ処理を実行する前に、十分なディスク領域があることを確認します。このtarballは3.5 GB以上になると予想されます。

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

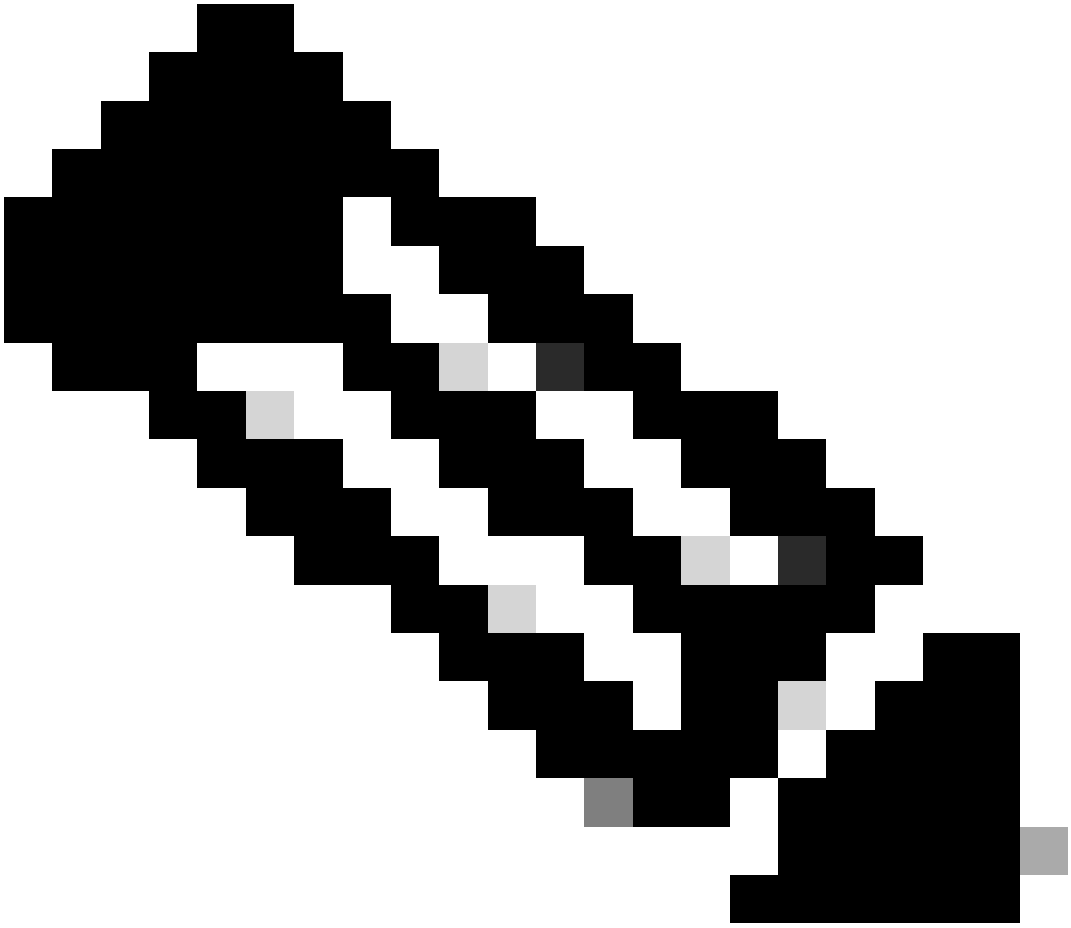
4. rootユーザーとして次のコマンドを実行し、undercloudノードからundercloud-backup-[timestamp].tar.gzというファイルにデータをバックアップし、バックアップサーバーに転送します。

```
[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バックアップ

1. ESCは、VIMと対話することで仮想ネットワーク機能(VNF)を起動します。

2. ESCはUltra-Mソリューションで1:1の冗長性を備えています。2つのESC VMが導入されており、Ultra-Mで1つの障害をサポートしています。たとえば、システム内に1つの障害が発生した場合は、システムをリカバリします。



注：複数の障害が存在する場合、その障害はサポートされておらず、システムの再導入が必要になる場合があります。

ESCバックアップ詳細：

- Running Configuration
- 会議用CDB
- ESCログ
- Syslogの設定

3. ESC DBバックアップの頻度は注意が必要で、導入されているさまざまなVNF VMのさまざまな状態マシンをESCが監視および維持するため、慎重に処理する必要があります。これらのバックアップは、特定のVNF/POD/サイトでこれらのアクティビティを実行した後に実行することをお勧めします。

4. health.shスクリプトを使用して、ESCの状態が良好であることを確認します。

```
[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_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. 実行コンフィギュレーションのバックアップを取り、ファイルをバックアップサーバに転送します。

```
[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データベースのバックアップ

1. バックアップを取る前に、ESC VMにログインし、このコマンドを実行します。

```
[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. ESCモードをチェックし、メンテナンスモードになっていることを確認します。

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

3. ESCで使用可能なデータベースバックアップ復元ツールを使用してデータベースをバックアップします。

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

4. ESCを操作モードに戻し、モードを確認します。

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

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

5. scriptsディレクトリに移動し、ログを収集します。

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

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

6. ESCのスナップショットを作成するには、最初にESCをシャットダウンします。

```
shutdown -r now
```

7. OSPDから、イメージ・スナップショットを作成します。

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

8. スナップショットが作成されたことを確認します。

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

9. OSPDからESCを起動します。

```
nova start esc1
```

10. スタンバイESC VMで同じ手順を繰り返し、ログをバックアップサーバに転送します。

11. ESC VMの両方でsyslog設定のバックアップを収集し、バックアップサーバに転送します。

```
[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バックアップ

ステップ 1：CPS Cluster-Managerのバックアップの作成

novaインスタンスを表示し、クラスタマネージャVMインスタンスの名前をメモするには、次のコマンドを使用します。

```
nova list
```

EscキーからCluманを停止します。

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

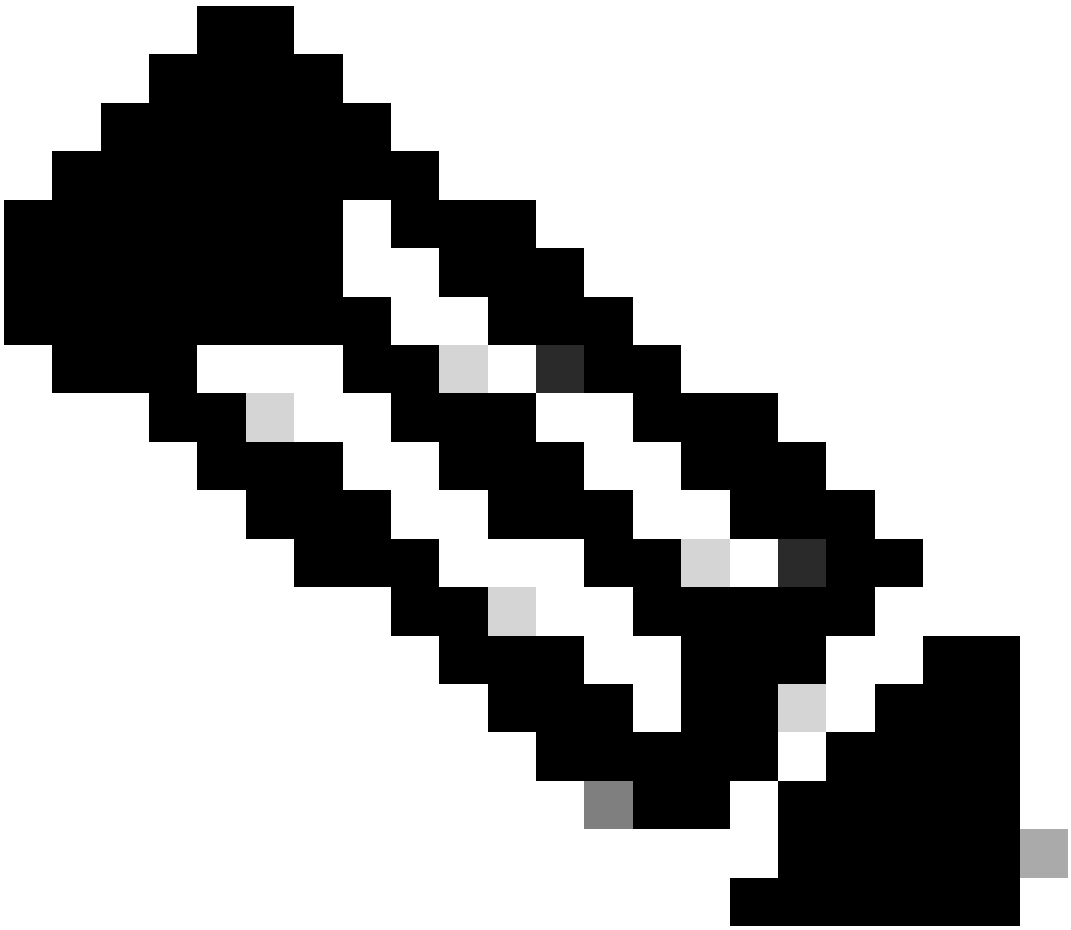
ステップ 2：Cluster Managerがシャットオフ状態であることを確認します。

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

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

ステップ 3：次のコマンドで示すように、novaスナップショット・イメージを作成します。

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



注：スナップショットに十分なディスク領域があることを確認します。

重要：スナップショットの作成後にVMが到達不能になった場合は、nova listコマンドを使用してVMのステータスを確認します。SHUTOFF状態の場合は、VMを手動で起動する必要があります。

ステップ 4：nova image-listコマンドを使用してイメージリストを表示します。

図1：出力例

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

ステップ 5：スナップショットが作成されると、スナップショットイメージがOpenStack Glanceに保存されます。スナップショットをリモートデータストアに保存するには、スナップショットをダウンロードし、OSPD内のファイルを(/home/stack/CPS_BACKUP)に転送します。

イメージをダウンロードするには、OpenStackで次のコマンドを使用します。

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

手順 6：次のコマンドで示すように、ダウンロードされたイメージをリストします。

```
ls –ltr *snapshot*
```

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

手順 7：将来リストアするCluster Manager VMのスナップショットを保存します。

2. 設定とデータベースをバックアップします。

```
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 t
6. backup the cronjobs by taking backup of the cron directory: /var/spool/cron/ from the Pcrfclient01/C
```

他のバックアップが必要かどうかをcrontab-lで確認します。

すべてのバックアップをOSPD /home/stack/CPS_BACKUPに転送します。

3. ESCプライマリからyamlファイルをバックアップします。

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

ファイルをOSPD /home/stack/CPS_BACKUPに転送します。

4. crontab -lエントリをバックアップします。

crontab -lを使用してtxtファイルを作成し、リモートの場所（OSPD /home/stack/CPS_BACKUP内）にftpで送信します。

5. LBおよびPCRFクライアントからルートファイルのバックアップを作成します。

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

復元手順

OSPDリカバリ

OSPDの回復手順は、これらの前提条件に基づいて実行されます。

1. OSPDバックアップは、古いOSPDサーバから使用できます。

2. OSPDの回復は、システム内の古いOSPDサーバを置き換える新しいサーバ上で実行できます。 .

ESCリカバリ

1. ESC VMは、VMがエラー状態またはシャットダウン状態の場合は回復可能です。影響を受けるVMを起動するには、ハードリブートを実行します。ESCを回復するには、次の手順を実行します。

2. エラー状態またはシャットダウン状態のVMを特定し、特定されたらESC VMをハードリブートします。この例では、auto-test-vnfm1-ESC-0をリブートします。

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

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

8. ESCサービスを再起動して、データベースの復元を完了します。両方のVMでHAを実行するには、キープアライブサービスを再起動します。

```
# service keepalived start
```

9. VMが正常に復元され、実行されたら、すべてのsyslog固有の設定が、以前の正常な既知のバックアップから復元されていることを確認します。すべてのESC VMで復元されていることを確認します。

```
[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. ESCをOSPDスナップショットから再構築する必要がある場合は、このコマンドを使用し、バックアップ中に取得したスナップショットを使用します。

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

11. リビルドが完了した後のESCのステータスを確認します。

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

12. 次のコマンドを使用して、ESCの状態を確認します。

```
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%
```

ESCがVMの起動に失敗した場合

- 予期しない状態が原因で、ESCがVMの起動に失敗する場合があります。回避策は、プライマリESCをリブートしてESCスイッチオーバーを実行することです。ESCスイッチオーバーには約1分かかります。 新しいプライマリESCでhealth.shを実行して、稼働していることを確認します。ESCがプライマリになると、ESCはVMの状態を修正してVMを起動できます。この操作はスケジュールされているため、完了するまで5 ~ 7分待つ必要があります。
- /var/log/esc/yangesc.logおよび/var/log/esc/escmanager.logをモニタできます。5 ~ 7分経過してもVMがリカバリされない場合は、影響を受けるVMの手動リカバリを実行する必要があります。
- VMが正常に復元されて実行されたら、syslog固有の設定がすべて、以前の正常な既知のバックアップから復元されていることを確認します。すべてのESC VMで復元されていることを確認します

```
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
```

CPSリカバリ

OpenStackでCluster Manager VMを復元します。

ステップ 1：次のコマンドに示すように、クラスタマネージャVMスナップショットをコントローラブレードにコピーします。

```
ls –ltr *snapshot*
```

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

ステップ 2：データストアからOpenStackにスナップショットイメージをアップロードします。

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

ステップ 3：次の例に示すように、スナップショットがNovaコマンドを使用してアップロードされているかどうかを確認します。

```
nova image-list
```

図2：出力例

```
+--------------------------------------+-----------------------------------+--------+--------------------------------------+
| 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 |                                      |
+--------------------------------------+-----------------------------------+--------+--------------------------------------+
```

ステップ 4：クラスタマネージャVMが存在するかどうかに応じて、クラスタを作成するか、またはクラスタを再構築するかを選択できます。

・ クラスタマネージャVMインスタンスが存在しない場合は、次の例に示すように、Heatコマンドまたはコマンドでクラスタを作成します。

ESCを使用してクラスタVMを作成します。

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

PCRFクラスタは、前のコマンドを使用して起動し、config_br.py restoreで作成したバックアップからクラスタマネージャの設定を復元できます。また、バックアップで作成したダンプからmongorestoreを復元できます。

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

・ クラスタマネージャVMインスタンスが存在する場合は、nova rebuildコマンドを使用して、アップロードされたスナップショットでクローンVMインスタンスを次のように再構築します。

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

例：

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

ステップ 5：次に示すように、すべてのインスタンスをリストし、新しいクラスタマネージャインスタンスが作成され、実行されていることを確認します。

```
nova list
```

画像 3.出力例

```
+--------------------------------------+--------+--------+------------+-------------+-----------------------------------------------------+
| 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 |
```

システムに最新のパッチを復元します。

```
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 nam
5.       Run build_all.sh script to create updated QPS packages:  /var/qps/install/current/scripts/build
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：ソフトウェアコンポーネントはすべて、現在のステータスとして「モニタされていません(Not Monitored)」と表示される必要があります。

```
8.       Update the qns VMs with the new software using reinit.sh script:  /var/qps/install/current/scri
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
```

Cronjobsを復元します。

1. バックアップファイルをOSPDからCluman/Pcrfclient01に移動します。

2. バックアップからcronjobをアクティブ化するコマンドを実行します。

<#root>

#crontab Cron-backup

3. cronjobsがこのコマンドによってアクティブになっているかどうかを確認します。

```
#crontab -l
```

クラスタ内の個々のVMを復元します。

pcrfclient01 VMを再導入するには、次の手順を実行します。

ステップ 1：Cluster Manager VMにルートユーザとしてログインします。

ステップ 2：次のコマンドを使用して、SVNリポジトリのUUIDを覚えておきます。

<#root>

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

このコマンドでは、リポジトリのUUIDを出力できます。

例：Repository UUID: ea50bbd2-5726-46b8-b807-10f4a7424f0e

ステップ 3：次の例に示すように、Cluster Managerでバックアップポリシービルダーの設定データをインポートします。

<#root>

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

---

注：多くの導入では、設定データを定期的にバックアップするcronジョブを実行しています。詳細については、「Subversionリポジトリバックアップ」を参照してください。

ステップ 4： 最新の設定を使用してCluster ManagerでVMアーカイブファイルを生成するには、

次のコマンドを実行します。

<#root>

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

ステップ 5：pcrfclient01 VMを導入するには、次のいずれかを実行します。

OpenStackでは、HEATテンプレートまたはNovaコマンドを使用してVMを再作成します。詳細については、『CPS Installation Guide for OpenStack』を参照してください。

手順 6：次の一連のコマンドを実行して、pcrfclient01をプライマリとして、pcrfclient01とpcrfclient02の間のSVNプライマリ/セカンダリ同期を再確立します。

SVNがすでに同期されている場合は、これらのコマンドを発行しないでください。

SVNが同期されているかどうかを確認するには、pcrfclient02からこのコマンドを実行します。

値が返された場合、SVNはすでに同期されています。

<#root>

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

pcrfclient01から次のコマンドを実行します。

<#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

手順 7：pcrfclient01が監視VMでもある場合は、次の手順を実行します。

a)システム構成に基づいて、mongodbの開始/停止スクリプトを作成します。すべての導入で、これらすべてのデータベースが設定されているわけではありません。

注：どのデータベースを設定する必要があるのかを判断するには、/etc/broadhop/mongoConfig.cfgを参照してください。

<#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)モンゴプロセスを開始します。

<#root>

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

c)アービターが起動するのを待ってから、diagnostics.sh —get_replica_statusを実行して、レプリカセットの状態をチェックします。

pcrfclient02 VMを再導入するには、次の手順を実行します。

ステップ 1：Cluster Manager VMにルートユーザとしてログインします。

ステップ 2：最新の設定を使用してCluster ManagerでVMアーカイブファイルを生成するには、次のコマンドを実行します。

<#root>

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

ステップ 3：pcrfclient02 VMを導入するには、次のいずれかを実行します。
OpenStackでは、HEATテンプレートまたはNovaコマンドを使用してVMを再作成します。詳細については、『CPS Installation Guide for OpenStack』を参照してください。

ステップ 4：pcrfclient01へのセキュアシェル：

<#root>

```
ssh pcrfclient01
```

ステップ 5：pcrfclient01からSVNリポジトリを回復するには、次のスクリプトを実行します。

<#root>

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

sessionmgr VMを再デプロイするには、次の手順に従います。

ステップ 1：Cluster Manager VMにルートユーザとしてログインします。

ステップ 2：sessionmgr VMを展開して、障害が発生した、または破損したVMを置き換えるには、次のいずれかを実行します。

OpenStackでは、HEATテンプレートまたはNovaコマンドを使用してVMを再作成します。詳細については、『CPS Installation Guide for OpenStack』を参照してください。

ステップ 3：システム設定に基づいてmongodb起動/停止スクリプトを作成します。

すべての導入で、これらすべてのデータベースが設定されているわけではありません。セットアップが必要なデータベースを確認するには、/etc/broadhop/mongoConfig.cfgを参照してください。

```
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
```

ステップ 4：sessionmgr VMにセキュアシェル(SSH)で接続し、mongoプロセスを開始します。

<#root>

ssh sessionmgrXX

/usr/bin/systemctl start sessionmgr-XXXXX

ステップ 5：メンバーが起動し、セカンダリメンバーが同期するまで待ってから、diagnostics.sh —get_replica_statusを実行してデータベースの状態を確認します。

手順 6：セッションマネージャデータベースを復元するには、バックアップが – mongo-allまたは – mongoオプションを使用して実行されたかどうかに応じて、次のコマンド例のいずれかを使用します。

<#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
```

ポリシーディレクタ（ロードバランサ）VMを再展開するには、次の手順を実行します。

ステップ 1：Cluster Manager VMにルートユーザとしてログインします。

ステップ 2：Cluster Managerでバックアップポリシービルダーの設定データをインポートするには、次のコマンドを実行します。

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

ステップ 3：最新の設定を使用してCluster ManagerでVMアーカイブファイルを生成するには、次のコマンドを実行します。

<#root>

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

ステップ 4：lb01 VMを導入するには、次のいずれかを実行します。

OpenStackでは、HEATテンプレートまたはNovaコマンドを使用してVMを再作成します。詳細については、『CPS Installation Guide for OpenStack』を参照してください。

ポリシーサーバ(QNS)VMを再導入するには、次の手順を実行します。

ステップ 1：Cluster Manager VMにルートユーザとしてログインします。

ステップ 2：次の例に示すように、Cluster Managerでバックアップポリシービルダーの設定データをインポートします。

<#root>

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

ステップ 3：最新の設定を使用してCluster ManagerでVMアーカイブファイルを生成するには、次のコマンドを実行します。

<#root>

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

ステップ 4：qns VMを導入するには、次のいずれかを実行します。
OpenStackでは、HEATテンプレートまたはNovaコマンドを使用してVMを再作成します。詳細については、『CPS Installation Guide for OpenStack』を参照してください。

データベースの復元の一般的な手順。

ステップ 1：データベースを復元するには、次のコマンドを実行します。

<#root>

```
config_br.py –a import --mongo-all /mnt/backup/backup_$date.tar.gz where $date is the timestamp when the
```

たとえば、

<#root>

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

ステップ 2：データベースにログインし、データベースが稼働していて、アクセス可能であるかどうかを確認します。

1. セッションマネージャにログインします。

<#root>

mongo --host sessionmgr01 --port $port

ここで、$portはチェックするデータベースのポート番号です。たとえば、27718はデフォルトのバランスポートです。

2. 次のコマンドを実行して、データベースを表示します。

<#root>

show dbs

3. 次のコマンドを実行して、mongoシェルをデータベースに切り替えます。

<#root>

use $db

$dbは前のコマンドで表示されたデータベース名です。

useコマンドは、mongoシェルをそのデータベースに切り替えます。

たとえば、

<#root>

use balance_mgmt

4. コレクションを表示するには、次のコマンドを実行します：

<#root>

show collections

5. コレクション内のレコード数を表示するには、次のコマンドを実行します：

<#root>

db.$collection.count()

For example, db.account.count()

前述の例では、Balanceデータベース(balance_mgmt)の回収勘定のレコード数を表示できます。

Subversionリポジトリの復元。

Policy Builderの設定データをバックアップから復元するには、次のコマンドを実行します。

<#root>

```
config_br.py –a import --svn /mnt/backup/backup_$date.tgz where, $date is the date when the cron created
```

Grafanaダッシュボードを復元します。

Grafanaダッシュボードは、次のコマンドを使用して復元できます。

<#root>

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

リストアの検証

データをリストアした後、次のコマンドを実行して、稼働中のシステムを確認します。

<#root>

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

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。