

コンピューティングサーバUCS C240 M4のPCRF交換

内容

[概要](#)

[背景説明](#)

[ヘルスチェック](#)

[バックアップ](#)

[コンピューティングノードでホストされるVMの特定](#)

[シャットダウンするVM上のPCRFサービスを無効にする](#)

[Nova集約リストからの計算ノードの削除](#)

[コンピューティングノードの削除](#)

[オーバークラウドから削除](#)

[サービスリストからのコンピューティングノードの削除](#)

[Neutronエージェントの削除](#)

[Ironicデータベースから削除](#)

[新しいコンピューティングノードのインストール](#)

[新しいコンピューティングノードをオーバークラウドに追加する](#)

[VMのリストア](#)

[Nova集約リストへの追加](#)

[Elastic Services Controller\(ESC\)からのVMリカバリ](#)

—
[VMに存在するCisco Policy and Charging Rules Function\(PCRF\)サービスの確認](#)

[ESCリカバリが失敗した場合の1つ以上のVMの削除と再導入](#)

[サイトの最新のESCテンプレートを取得する](#)

[ファイルの変更手順](#)

[ステップ1: エクスポートテンプレートファイルを変更します。](#)

[ステップ2: 変更されたエクスポートテンプレートファイルを実行します。](#)

[ステップ3: VMを追加するようにエクスポートテンプレートファイルを変更します。](#)

[ステップ4: 変更されたエクスポートテンプレートファイルを実行します。](#)

[ステップ5: VMに存在するPCRFサービスを確認します。](#)

[ステップ6: 診断を実行してシステムステータスを確認します。](#)

[関連情報](#)

概要

このドキュメントでは、Cisco Policy Suite(CPS)Virtual Network Functions(VNF)をホストするUltra-Mセットアップで故障したコンピューティングサーバを交換するために必要な手順について説明します。

背景説明

このドキュメントは、Cisco Ultra-Mプラットフォームに精通したシスコ担当者を対象としており、コンピュータサーバ交換時にOpenStackおよびCPS VNFレベルで実行する必要がある手順について詳しく説明しています。

注：このドキュメントの手順を定義するために、Ultra M 5.1.xリリースが検討されています。

ヘルスチェック

コンピューティングノードを交換する前に、Red Hat OpenStack Platform環境の現在の稼働状態を確認することが重要です。コンピューティングの交換プロセスがオンの場合に複雑にならないように、現在の状態を確認することをお勧めします。

ステップ1:OpenStack Deployment(OSPD)から。

```
[root@director ~]$ su - stack
[stack@director ~]$ cd ansible
[stack@director ansible]$ ansible-playbook -i inventory-new openstack_verify.yml -e
platform=pcrf
```

ステップ2:15分ごとに生成されるultam-healthレポートからシステムの健全性を確認します。

```
[stack@director ~]# cd /var/log/cisco/ultram-health
```

ステップ3：ファイルultam_health_os.reportをチェックします。XXXステータスとして表示されるサービスはneutron-sriov-nic-agent.serviceのみです。

ステップ4:OSPDから実行されるすべてのコントローラでrabbitmqが実行されているかどうかを確認します。

```
[stack@director ~]# for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo rabbitmqctl
eval 'rabbit_diagnostics:maybe_stuck().'" ) & done
```

ステップ5:stonithが有効になっていることを確認します

```
[stack@director ~]# sudo pcs property show stonith-enabled
```

ステップ6：すべてのコントローラでPCSのステータスを確認します。

- すべてのコントローラ・ ノードがhaproxy-cloneの下で開始されます。
- すべてのコントローラノードがGaleraの下でアクティブです。
- すべてのコントローラノードがRabbitmqの下で開始されます。
- 1台のコントローラノードがアクティブで、2台のスタンバイがredisの下にあります。

ステップ7:OSPDから

```
[stack@director ~]$ for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo pcs status"
) ;done
```

ステップ8：すべてのopenstackサービスがアクティブであることを確認します。OSPDからのこの

コマンドを実行します。

```
[stack@director ~]# sudo systemctl list-units "openstack*" "neutron*" "openvswitch*"
ステップ9: コントローラのCEPHステータスがHEALTH_OKであることを確認します。
```

```
[stack@director ~]# for i in $(nova list | grep controller | awk '{print $12}' | sed 's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo ceph -s" ) ;done
```

ステップ10: OpenStackコンポーネントのログを確認します。エラーを探します。

Neutron:

```
[stack@director ~]# sudo tail -n 20 /var/log/neutron/{dhcp-agent, l3-agent, metadata-agent, openvswitch-agent, server}.log
```

Cinder:

```
[stack@director ~]# sudo tail -n 20 /var/log/cinder/{api, scheduler, volume}.log
```

Glance:

```
[stack@director ~]# sudo tail -n 20 /var/log/glance/{api, registry}.log
```

ステップ11: OSPDからAPIに対してこれらの検証を実行します。

```
[stack@director ~]$ source
```

```
[stack@director ~]$ nova list
```

```
[stack@director ~]$ glance image-list
```

```
[stack@director ~]$ cinder list
```

```
[stack@director ~]$ neutron net-list
```

ステップ12: サービスの健全性を確認します。

Every service status should be "up":

```
[stack@director ~]$ nova service-list
```

Every service status should be " :-)":

```
[stack@director ~]$ neutron agent-list
```

Every service status should be "up":

```
[stack@director ~]$ cinder service-list
```

バックアップ

リカバリの場合は、次の手順を使用してOSPDデータベースのバックアップを取ることを推奨します。

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

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

このプロセスにより、インスタンスの可用性に影響を与えることなく、ノードを確実に交換できます。また、CPS構成のバックアップも推奨されます。

CPS VMをバックアップするには、Cluster Manager VMから次の手順を実行します。

```
[root@CM ~]# config_br.py -a export --all /mnt/backup/CPS_backup_$(date +%Y-%m-%d).tar.gz
```

or

```
[root@CM ~]# config_br.py -a export --mongo-all --svn --etc --grafanadb --auth-htpasswd --haproxy /mnt/backup/$(hostname)_backup_all_$(date +%Y-%m-%d).tar.gz
```

コンピューティングノードでホストされるVMの特定

コンピューティングサーバでホストされているVMを特定します。

```
[stack@director ~]$ nova list --field name,host,networks | grep compute-10
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-
10e75d0e134d | pod1-compute-10.localdomain | Replication=10.160.137.161;
Internal=192.168.1.131; Management=10.225.247.229; tb1-orch=172.16.180.129
```

注：ここに示す出力では、最初のカラムはUniversal Unique Identifier(UUID)に対応し、2番目のカラムはVM名を表し、3番目のカラムはVMが存在するホスト名を表しています。この出力のパラメータは、以降のセクションで使用します。

シャットダウンするVM上のPCRFサービスを無効にする

ステップ1:VMの管理IPにログインします。

```
[stack@XX-ospd ~]$ ssh root@
```

```
[root@XXXSM03 ~]# monit stop all
```

ステップ2:VMがSM、OAM、またはアービタである場合は、sessionmgrサービスを停止します。

```
[root@XXXSM03 ~]# cd /etc/init.d
[root@XXXSM03 init.d]# ls -l sessionmgr*
-rwxr-xr-x 1 root root 4544 Nov 29 23:47 sessionmgr-27717
-rwxr-xr-x 1 root root 4399 Nov 28 22:45 sessionmgr-27721
-rwxr-xr-x 1 root root 4544 Nov 29 23:47 sessionmgr-27727
```

ステップ3:sessionmgr-xxxxxというタイトルのファイルごとに、service sessionmgr-xxxxx stopを実行します。

```
[root@XXXSM03 init.d]# service sessionmgr-27717 stop
```

Nova集約リストからの計算ノードの削除

ステップ1:novaアグリゲートをリストし、ホストされているVNFに基づいてコンピューティングサーバに対応するアグリゲートを特定します。通常、この形式は<VNFNAME>-SERVICE<X>です。

```
[stack@director ~]$ nova aggregate-list
+-----+-----+-----+
| Id | Name                | Availability Zone |
+-----+-----+-----+
| 29 | POD1-AUTOIT        | mgmt              |
| 57 | VNF1-SERVICE1     | -                 |
| 60 | VNF1-EM-MGMT1     | -                 |
| 63 | VNF1-CF-MGMT1     | -                 |
| 66 | VNF2-CF-MGMT2     | -                 |
| 69 | VNF2-EM-MGMT2     | -                 |
| 72 | VNF2-SERVICE2    | -                 |
| 75 | VNF3-CF-MGMT3     | -                 |
| 78 | VNF3-EM-MGMT3     | -                 |
| 81 | VNF3-SERVICE3     | -                 |
+-----+-----+-----+
```

この場合、交換するコンピューティングサーバはVNF2に属しています。したがって、対応する集約リストはVNF2-SERVICE2です。

ステップ2：識別された集約からコンピューティングノードを削除します(セクション「コンピューティングノードでホストされているVMを識別する」で示されているホスト名で削除◆◆)

```
nova aggregate-remove-host
```

```
[stack@director ~]$ nova aggregate-remove-host VNF2-SERVICE2 pod1-compute-10.localdomain
```

ステップ3：コンピューティングノードが集約から削除されているかどうかを確認します。ここで、集約の下にホストをリストすることはできません。

```
nova aggregate-show
```

```
[stack@director ~]$ nova aggregate-show VNF2-SERVICE2
```

コンピューティングノードの削除

このセクションで説明する手順は、コンピューティングノードでホストされるVMに関係なく共通

です。

オーバークラウドから削除

ステップ1: 次に示す内容のdelete_node.shという名前のスクリプトファイルを作成します。記載されているテンプレートが、スタック配置に使用されるdeploy.shスクリプトと同じであることを確認します。

```
delete_node.sh
```

```
openstack overcloud node delete --templates -e /usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml -e /home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e /home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml --stack
```

```
[stack@director ~]$ source stackrc
[stack@director ~]$ /bin/sh delete_node.sh
+ openstack overcloud node delete --templates -e /usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml -e /usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml -e /home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e /home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml -e /home/stack/custom-templates/layout.yaml --stack
pod1 49ac5f22-469e-4b84-badc-031083db0533
Deleting the following nodes from stack pod1:
- 49ac5f22-469e-4b84-badc-031083db0533
Started Mistral Workflow. Execution ID: 4ab4508a-c1d5-4e48-9b95-ad9a5baa20ae

real    0m52.078s
user    0m0.383s
sys     0m0.086s
```

ステップ2: OpenStackスタックの動作がCOMPLETE状態になるまで待ちます。

```
[stack@director ~]$ openstack stack list
+-----+-----+-----+-----+
| ID                                     | Stack Name | Stack Status   | Creation Time           |
| Updated Time                           |            |                |                          |
+-----+-----+-----+-----+
| 5df68458-095d-43bd-a8c4-033e68ba79a0 | pod1       | UPDATE_COMPLETE | 2018-05-08T21:30:06Z | 2018-05-08T20:42:48Z |
+-----+-----+-----+-----+
```

サービスリストからのコンピューティングノードの削除

サービスリストからコンピューティングサービスを削除します。

```
[stack@director ~]$ source corerc
[stack@director ~]$ openstack compute service list | grep compute-8
| 404 | nova-compute | pod1-compute-8.localdomain | nova | enabled | up | 2018-05-08T18:40:56.000000 |
```

```
openstack compute service delete
```

```
[stack@director ~]$ openstack compute service delete 404
```

Neutronエージェントの削除

古い関連付けられたNeutronエージェントを削除し、コンピューティングサーバのvswitchエージェントを開きます。

```
[stack@director ~]$ openstack network agent list | grep compute-8
| c3ee92ba-aa23-480c-ac81-d3d8d01dcc03 | Open vSwitch agent | pod1-compute-8.localdomain |
None | False | UP | neutron-openvswitch-agent |
| ec19cb01-abbb-4773-8397-8739d9b0a349 | NIC Switch agent | pod1-compute-8.localdomain |
None | False | UP | neutron-sriov-nic-agent |
```

```
openstack network agent delete
```

```
[stack@director ~]$ openstack network agent delete c3ee92ba-aa23-480c-ac81-d3d8d01dcc03
```

```
[stack@director ~]$ openstack network agent delete ec19cb01-abbb-4773-8397-8739d9b0a349
```

Ironicデータベースから削除

Ironicデータベースからノードを削除し、確認します。

```
[stack@director ~]$ source stackrc
```

```
nova show
```

```
[stack@director ~]$ nova show pod1-compute-10 | grep hypervisor
| OS-EXT-SRV-ATTR:hypervisor_hostname | 4ab21917-32fa-43a6-9260-02538b5c7a5a
```

```
ironic node-delete
```

```
[stack@director ~]$ ironic node-delete 4ab21917-32fa-43a6-9260-02538b5c7a5a
```

```
[stack@director ~]$ ironic node-list (node delete must not be listed now)
```

新しいコンピューティングノードのインストール

新しいUCS C240 M4サーバをインストールする手順と初期セットアップ手順は、次から参照できます。『[Cisco UCS C240 M4 Server Installation and Service Guide](#)』

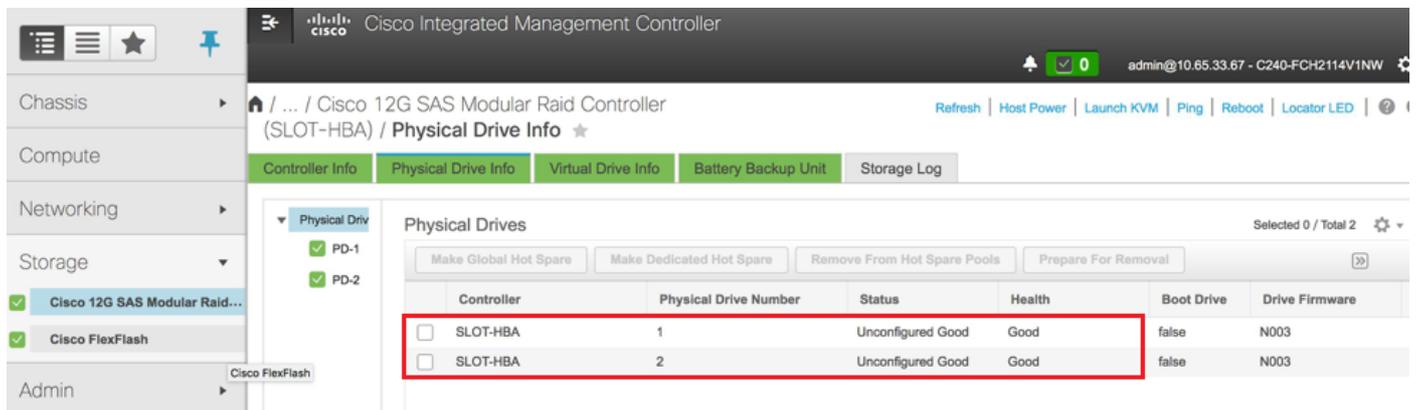
ステップ1：サーバのインストール後、ハードディスクを古いサーバとしてそれぞれのスロットに挿入します。

ステップ2:CIMC IPを使用してサーバにログインします。

ステップ3：ファームウェアが以前に使用した推奨バージョンと異なる場合は、BIOSアップグレードを実行します。BIOSアップグレードの手順は次のとおりです。[Cisco UCS CシリーズラックマウントサーバBIOSアップグレードガイド](#)

ステップ4：物理ドライブのステータスを確認するには、[Storage] > [Cisco 12G SAS Modular Raid Controller (SLOT-HBA)] > [Physical Drive Info]に移動します。構成されていない正常な必要

ここに示すストレージはSSDドライブです。



Controller	Physical Drive Number	Status	Health	Boot Drive	Drive Firmware
<input type="checkbox"/> SLOT-HBA	1	Unconfigured Good	Good	false	N003
<input type="checkbox"/> SLOT-HBA	2	Unconfigured Good	Good	false	N003

ステップ5:RAIDレベル1の物理ドライブから仮想ドライブを作成するには、[Storage] > [Cisco 12G SAS Modular Raid Controller (SLOT-HBA)] > [Controller Info] > [Create Virtual Drive from Unused Physical Drives]に移動します

Cisco Integrated Management Controller
Create Virtual Drive from Unused Physical Drives

RAID Level: 1 Enable Full Disk Encryption

Create Drive Groups

Physical Drives Selected 2 / Total 2

ID	Size(MB)	Model	Interface	Type
<input checked="" type="checkbox"/> 1	1906394 MB	SEAGA..	HDD	SAS
<input checked="" type="checkbox"/> 2	1906394 MB	SEAGA..	HDD	SAS

Drive Groups
No data available

Virtual Drive Properties

Name: RAID1
Access Policy: Read Write
Read Policy: No Read Ahead
Cache Policy: Direct IO

Disk Cache Policy: Unchanged
Write Policy: Write Through
Strip Size (MB): 64k
Size: MB

Cisco Integrated Management Controller
Create Virtual Drive from Unused Physical Drives

RAID Level: 1 Enable Full Disk Encryption

Create Drive Groups

Physical Drives Selected 0 / Total 0

No data available

Drive Groups
 DG [1,2]

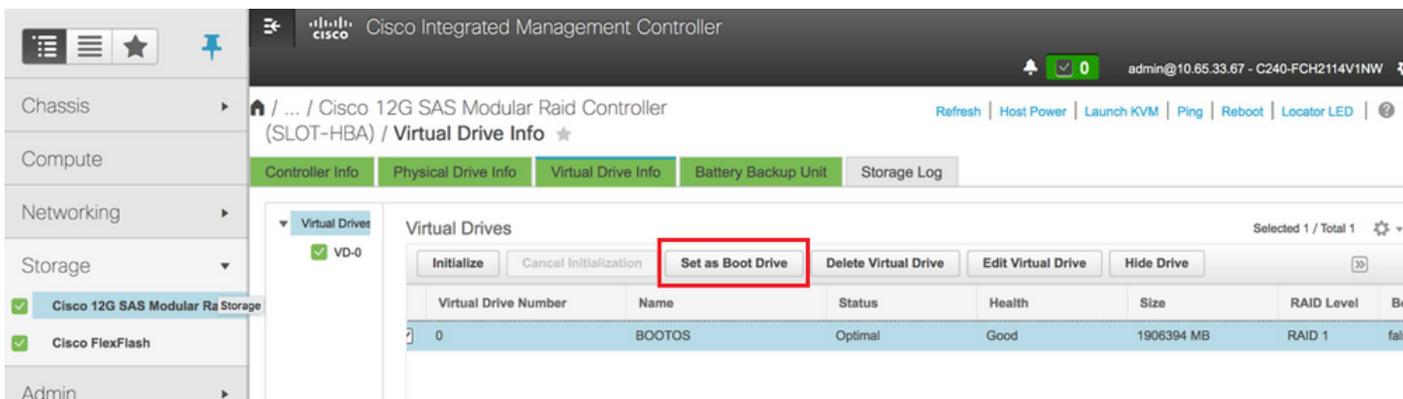
Virtual Drive Properties

Name:

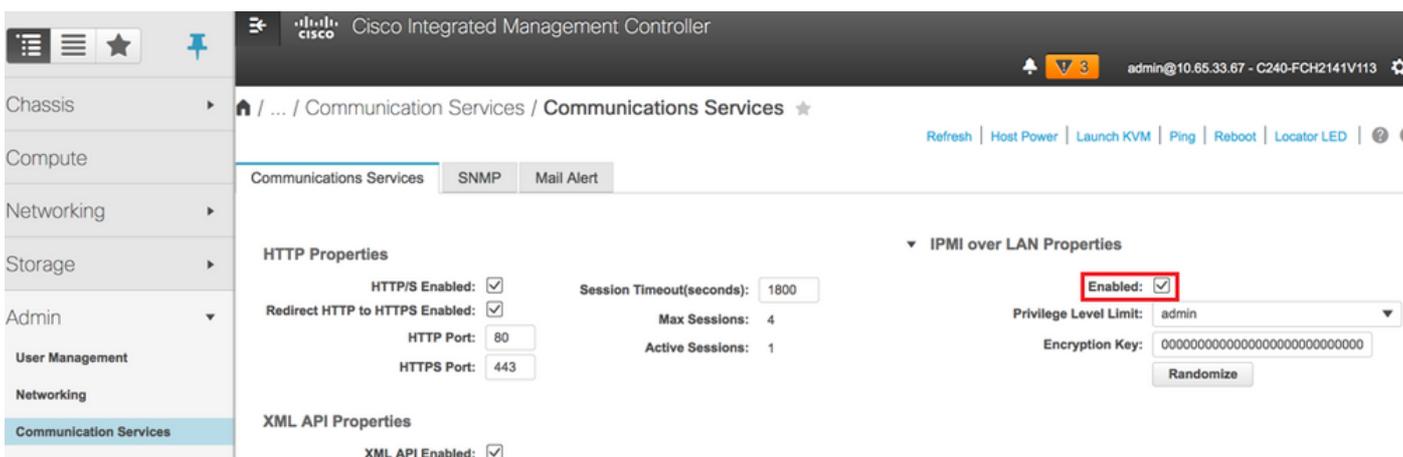
Access Policy: Read Write
Read Policy: No Read Ahead
Cache Policy: Direct IO

Disk Cache Policy: Unchanged
Write Policy: Write Through
Strip Size (MB): 64k
Size: 1906394 MB

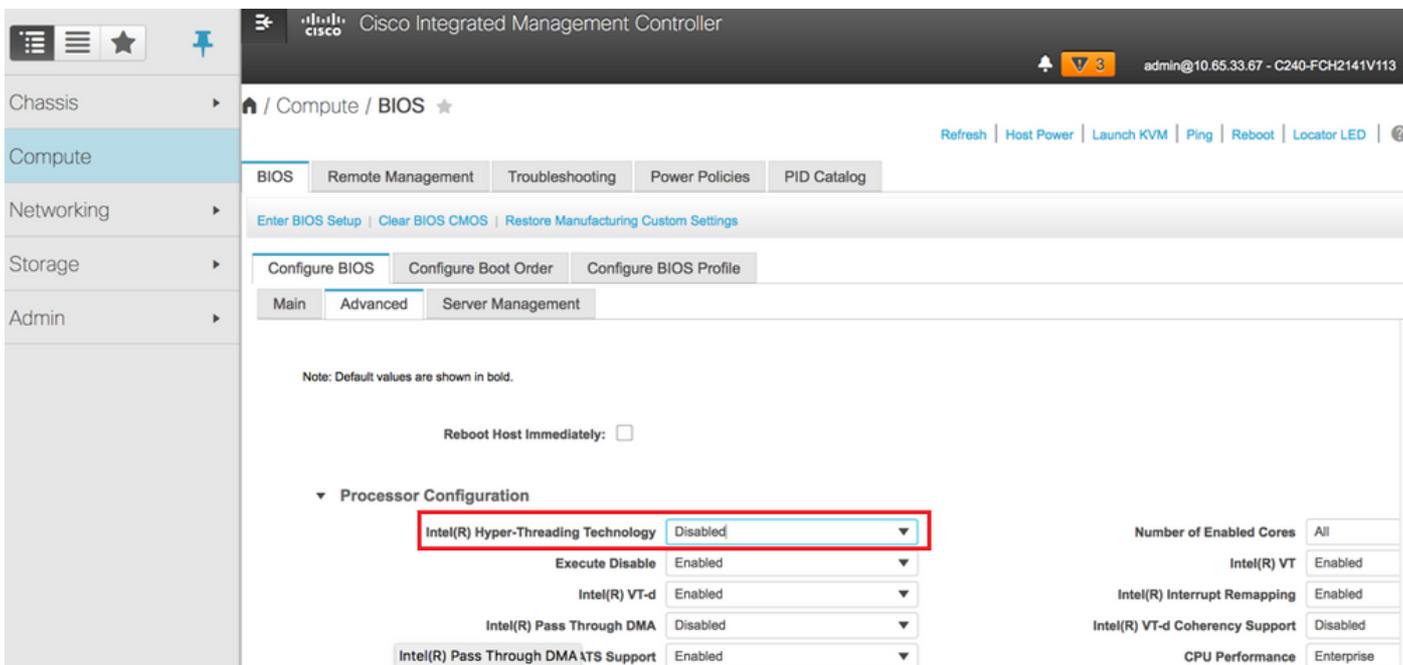
ステップ6:VDを選択し、図に示すように[Set as Boot Drive]を設定します。



ステップ7:IPMI over LANを有効にするには、図に示すように、[Admin] > [Communication Services] > [Communication Services]に移動します。



ステップ8 : 図に示すように、ハイパースレッディングを無効にするには、[Compute] > [BIOS] > [Configure BIOS] > [Advanced] > [Processor Configuration]に移動します。



注：このセクションに示す図と設定手順は、ファームウェアバージョン3.0(3e)を参照するもので、他のバージョンで作業する場合は、若干の違いがあります

新しいコンピューティングノードをオーバークラウドに追加する

このセクションで説明する手順は、コンピューティングノードによってホストされるVMに関係なく共通です。

ステップ1: 異なるインデックスを持つコンピューティングサーバを追加します。

追加する新しいコンピュートサーバの詳細のみを含むadd_node.jsonファイルを作成します。新しいコンピューティングサーバのインデックス番号が、以前は使用されていないことを確認します。通常、次に高い計算値を増分します。

例: 最も前はcompute-17だったので、2-vnfシステムの場合はcompute-18を作成しました。

注: json形式に注意してください。

```
[stack@director ~]$ cat add_node.json
{
  "nodes": [
    {
      "mac": [
        ""

      ],
      "capabilities": "node:compute-18,boot_option:local",
      "cpu": "24",
      "memory": "256000",
      "disk": "3000",
      "arch": "x86_64",
      "pm_type": "pxe_ipmitool",
      "pm_user": "admin",
      "pm_password": "<PASSWORD>",
      "pm_addr": "192.100.0.5"
    }
  ]
}
```

ステップ2: jsonファイルをインポートします。

```
[stack@director ~]$ openstack baremetal import --json add_node.json
Started Mistral Workflow. Execution ID: 78f3b22c-5c11-4d08-a00f-8553b09f497d
Successfully registered node UUID 7eddfa87-6ae6-4308-b1d2-78c98689a56e
Started Mistral Workflow. Execution ID: 33a68c16-c6fd-4f2a-9df9-926545f2127e
Successfully set all nodes to available.
```

ステップ3: 前のステップでメモしたUUIDを使用して、ノードのイントロスペクションを実行します。

```
[stack@director ~]$ openstack baremetal node manage 7eddfa87-6ae6-4308-b1d2-78c98689a56e
[stack@director ~]$ ironic node-list |grep 7eddfa87
| 7eddfa87-6ae6-4308-b1d2-78c98689a56e | None | None | power off
| manageable | False |
```

```
[stack@director ~]$ openstack overcloud node introspect 7eddfa87-6ae6-4308-b1d2-78c98689a56e --
provide
Started Mistral Workflow. Execution ID: e320298a-6562-42e3-8ba6-5ce6d8524e5c
Waiting for introspection to finish...
Successfully introspected all nodes.
Introspection completed.
Started Mistral Workflow. Execution ID: c4a90d7b-ebf2-4fcb-96bf-e3168aa69dc9
Successfully set all nodes to available.
```

```
[stack@director ~]$ ironic node-list |grep available
| 7eddfa87-6ae6-4308-b1d2-78c98689a56e | None | None | power off
| available | False |
```

ステップ4:[ComputeIPs]の下のcustom-templates/layout.ymlにIPアドレスを追加します。このアドレスをリストの最後に追加します。例としてcompute-0を示します。

ComputeIPs:

internal_api:

- 11.120.0.43
- 11.120.0.44
- 11.120.0.45
- 11.120.0.43 <<< take compute-0 .43 and add here

tenant:

- 11.117.0.43
- 11.117.0.44
- 11.117.0.45
- 11.117.0.43 << and here

storage:

- 11.118.0.43
- 11.118.0.44
- 11.118.0.45
- 11.118.0.43 << and here

ステップ5:新しいコンピュートノードをオーバークラウドスタックに追加するために、スタックの導入に以前に使用したdeploy.shスクリプトを実行します。

```
[stack@director ~]$ ./deploy.sh
++ openstack overcloud deploy --templates -r /home/stack/custom-templates/custom-roles.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml -e
```

```
/usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml -e
/usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml -e
/home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e
/home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml --stack
ADN-ultram --debug --log-file overcloudDeploy_11_06_17__16_39_26.log --ntp-server 172.24.167.109
--neutron-flat-networks phys_pcie1_0,phys_pcie1_1,phys_pcie4_0,phys_pcie4_1 --neutron-network-
vlan-ranges datacentre:1001:1050 --neutron-disable-tunneling --verbose --timeout 180
```

```
...
Starting new HTTP connection (1): 192.200.0.1
"POST /v2/action_executions HTTP/1.1" 201 1695
HTTP POST http://192.200.0.1:8989/v2/action_executions 201
Overcloud Endpoint: http://10.1.2.5:5000/v2.0
Overcloud Deployed
clean_up DeployOvercloud:
END return value: 0
```

```
real 38m38.971s
user 0m3.605s
sys 0m0.466s
```

ステップ6:openstackスタックのステータスが[Complete]になるまで待ちます。

```
[stack@director ~]$ openstack stack list
+-----+-----+-----+-----+
| ID | Stack Name | Stack Status | Creation Time |
Updated Time |
+-----+-----+-----+-----+
| 5df68458-095d-43bd-a8c4-033e68ba79a0 | ADN-ultram | UPDATE_COMPLETE | 2017-11-02T21:30:06Z |
2017-11-06T21:40:58Z |
+-----+-----+-----+-----+
```

ステップ7:新しいコンピューティングノードがアクティブ状態であることを確認します。

```
[stack@director ~]$ source stackrc
[stack@director ~]$ nova list |grep compute-18
| 0f2d88cd-d2b9-4f28-b2ca-13e305ad49ea | pod1-compute-18 | ACTIVE | - | Running
| ctlplane=192.200.0.117 |

[stack@director ~]$ source corerc
[stack@director ~]$ openstack hypervisor list |grep compute-18
| 63 | pod1-compute-18.localdomain |
```

VMのリストア

Nova集約リストへの追加

集約ホストにコンピューティングノードを追加し、ホストが追加されているかどうかを確認します。

```
nova aggregate-add-host
```

```
[stack@director ~]$ nova aggregate-add-host VNF2-SERVICE2 pod1-compute-18.localdomain
```

```
nova aggregate-show
```

```
[stack@director ~]$ nova aggregate-show VNF2-SERVICE2
```

Elastic Services Controller(ESC)からのVMリカバリ

ステップ1:VMがnovaリストでエラー状態になっている。

```
[stack@director ~]$ nova list |grep VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
| ERROR | - | NOSTATE |
```

ステップ2:ESCからVMを回復します。

```
[admin@VNF2-esc-esc-0 ~]$ sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO
VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
[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
```

ステップ3:yanesc.logを監視します。

```
admin@VNF2-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 [VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d].

VMに存在するCisco Policy and Charging Rules Function(PCRF)サービスの確認

注：VMがシャットオフ状態の場合は、ESCからesc_nc_cliを使用して電源をオンにします。

クラスタマネージャVMからdiagnostics.shを確認し、回復されたVMに関するエラーが見つかったら、

ステップ1：各VMにログインします。

```
[stack@XX-ospd ~]$ ssh root@
```

```
[root@XXXSM03 ~]# monit start all
```

ステップ2:VMがSM、OAM、またはアービターである場合は、それに加えて、先に停止したsessionmgrサービスを開始します。

sessionmgr-xxxxxというタイトルのファイルごとに、service sessionmgr-xxxxx start:

```
[root@XXXSM03 init.d]# service sessionmgr-27717 start
```

まだ診断がクリアされていない場合は、Cluster Manager VMからbuild_all.shを実行して、各VMでVM-initを実行します。

```
/var/qps/install/current/scripts/build_all.sh
```

```
ssh VM e.g. ssh pcrfclient01
```

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

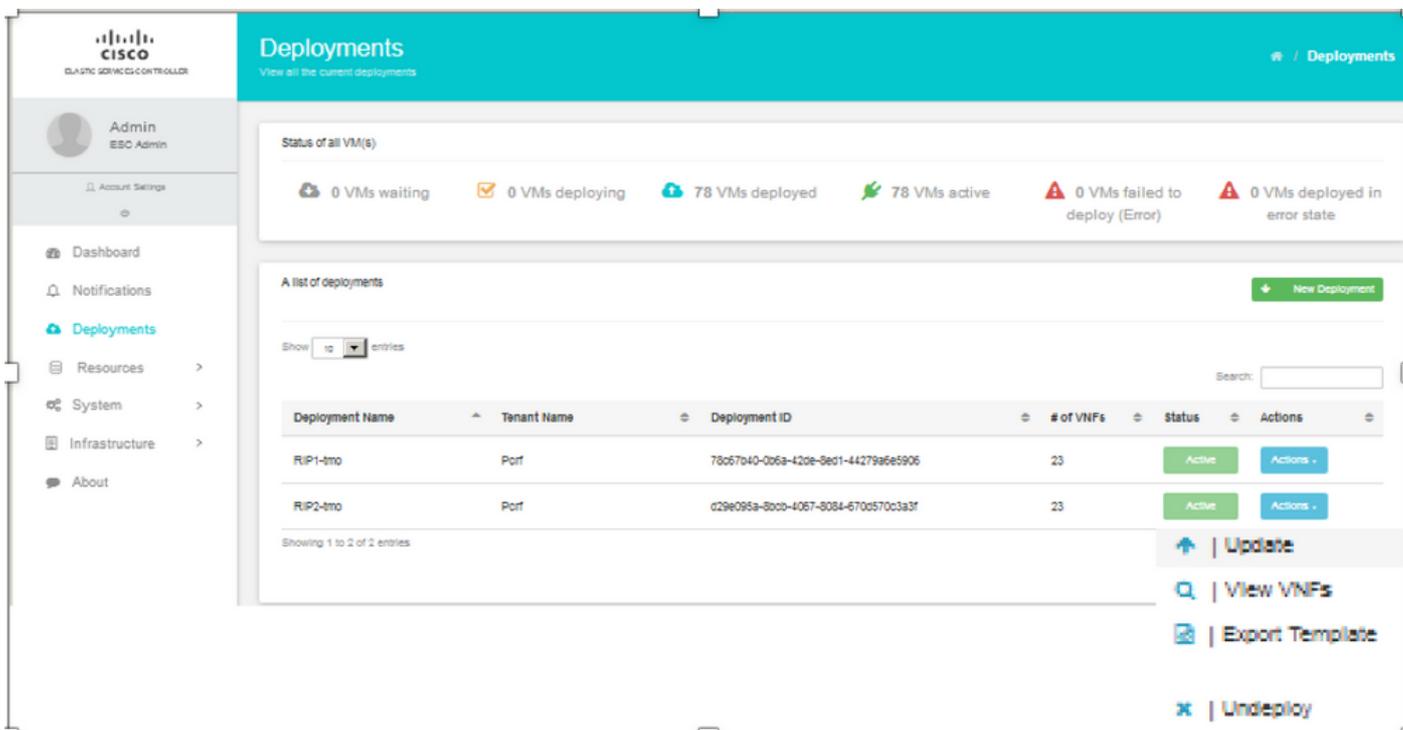
ESCリカバリが失敗した場合の1つ以上のVMの削除と再導入

ESCリカバリコマンド(上記)が機能しない場合(VM_RECOVERY_FAILED)、個々のVMを削除して再度追加します。

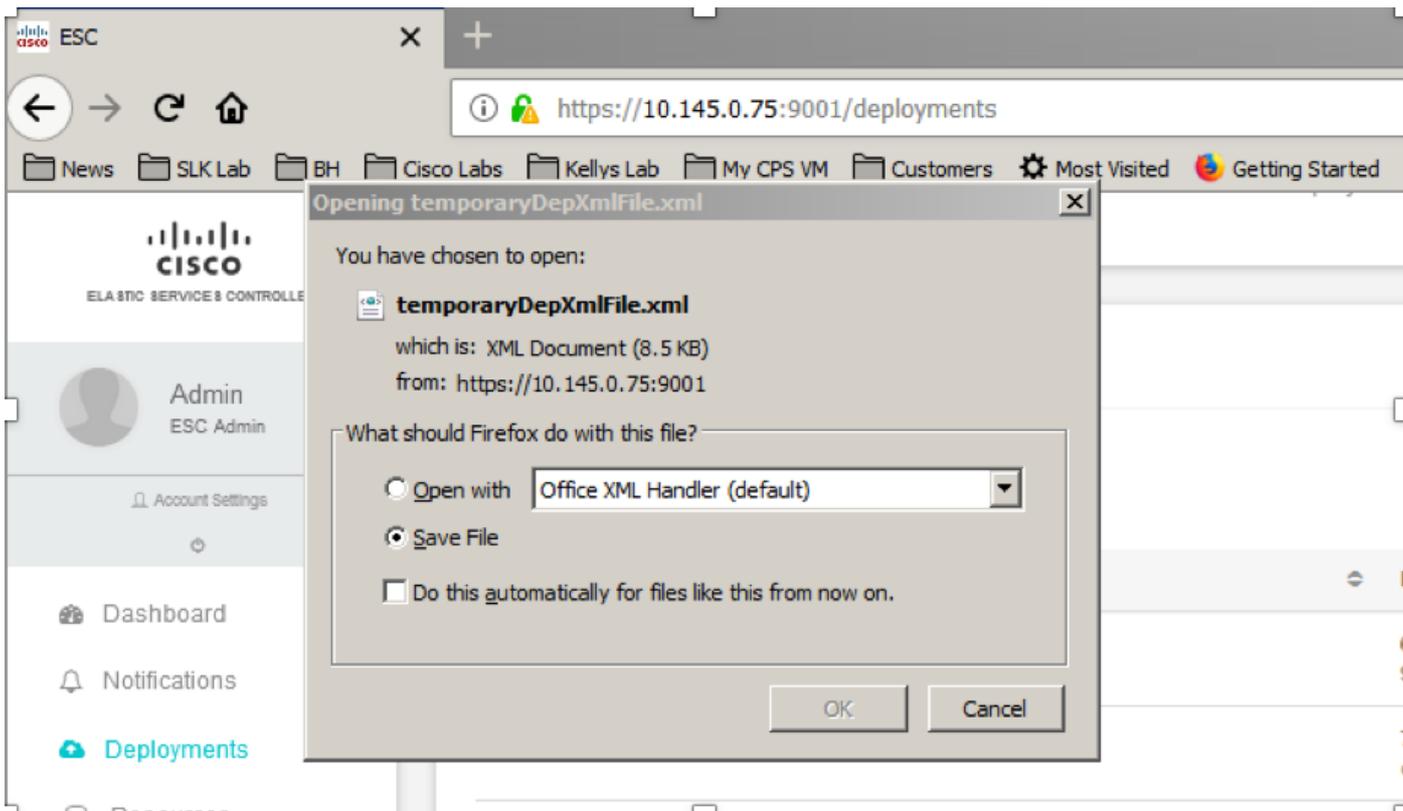
サイトの最新のESCテンプレートを取得する

ESCポータルから：

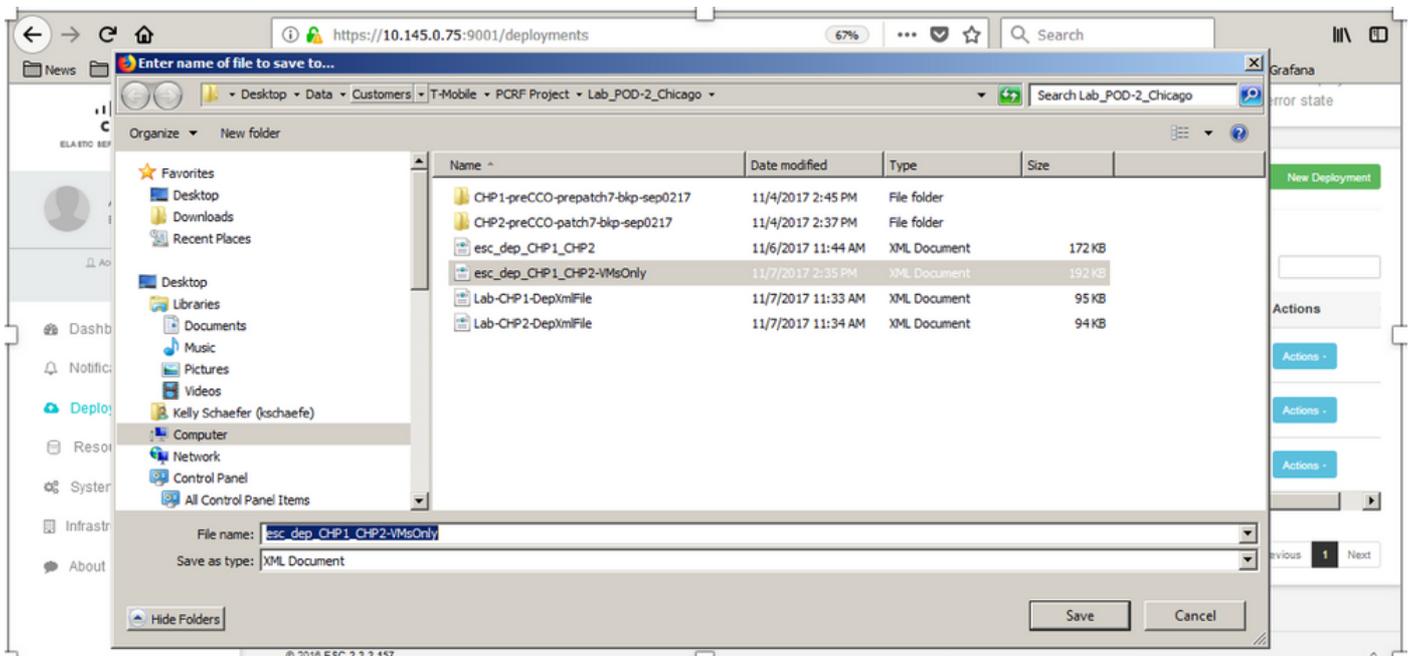
ステップ1：カーソルを青いアクションボタンの上に置くと、ポップアップウィンドウが開き、図に示すように[テンプレートの書き出し]をクリックします。



ステップ2：テンプレートをローカルマシンにダウンロードするオプションが表示されます。図に示すように、[Save File]をオンにします。



ステップ3：図に示すように、場所を選択し、後で使用できるようにファイルを保存します。



ステップ4：削除するサイトのActive ESCにログインし、このディレクトリのESCに保存した上記のファイルをコピーします。

```
/opt/cisco/esc/cisco-cps/config/gr/tmo/gen
```

ステップ5：ディレクトリを/opt/cisco/esc/cisco-cps/config/gr/tmo/genに変更します。

```
cd /opt/cisco/esc/cisco-cps/config/gr/tmo/gen
```

ファイルの変更手順

ステップ1：エクスポートテンプレートファイルを変更します。

この手順では、エクスポートテンプレートファイルを変更して、リカバリする必要があるVMに関連付けられているVMグループを1つまたは複数削除します。

エクスポートテンプレートファイルは、特定のクラスター用です。

そのクラスター内には複数のvm_groupがあります。 VMタイプ(PD、PS、SM、OM)ごとに1つ以上のvm_groupがあります。

注：一部のvm_groupには複数のVMがあります。 そのグループ内のすべてのVMが削除され、再度追加されます。

その導入内で、1つ以上のvm_groupsにタグを付けて削除する必要があります。

例：

```
<vm_group>
```

```
<name>cm</name>
```

ここで、<vm_group>を<vm_group nc:operation="delete">に変更し、変更を保存します。

ステップ2 : 変更されたエクスポートテンプレートファイルを実行します。

ESCを実行して、次の操作を実行します。

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

ESCポータルから、1つ以上のVMが展開されていない状態に移行し、完全に消えたことを確認できます。

進行状況はESCの/var/log/esc/yangesc.logで追跡できます

例 :

```
09:09:12,608 29-Jan-2018 INFO ===== UPDATE SERVICE REQUEST RECEIVED(UNDER TENANT) =====
09:09:12,608 29-Jan-2018 INFO Tenant name: Pcrf
09:09:12,609 29-Jan-2018 INFO Deployment name: WSP1-tmo
09:09:29,794 29-Jan-2018 INFO
09:09:29,794 29-Jan-2018 INFO ===== CONFID TRANSACTION ACCEPTED =====
09:10:19,459 29-Jan-2018 INFO
09:10:19,459 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:10:19,459 29-Jan-2018 INFO Type: VM_UNDEPLOYED
09:10:19,459 29-Jan-2018 INFO Status: SUCCESS
09:10:19,459 29-Jan-2018 INFO Status Code: 200
|
|
|
09:10:22,292 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:10:22,292 29-Jan-2018 INFO Type: SERVICE_UPDATED
09:10:22,292 29-Jan-2018 INFO Status: SUCCESS
09:10:22,292 29-Jan-2018 INFO Status Code: 200
```

ステップ3:VMを追加するようにエクスポートテンプレートファイルを変更します。

この手順では、エクスポートテンプレートファイルを変更して、リカバリ対象のVMに関連付けられているVMグループを再度追加します。

エクスポートテンプレートファイルは、2つの導入(cluster1/cluster2)に分かれています。

各クラスタ内にはvm_groupがあります。VMタイプ(PD、PS、SM、OM)ごとに1つ以上のvm_groupがあります。

注 :一部のvm_groupには複数のVMがあります。そのグループ内のすべてのVMが再度追加されます。

例 :

```
<vm_group nc:operation="delete">
<name>cm</name>
```

<vm_group nc:operation="delete">を単に<vm_group>に変更します。

注：ホストが交換されたためにVMを再構築する必要がある場合、ホストのホスト名が変更されている可能性があります。HOSTのホスト名が変更された場合は、vm_groupの配置セクション内のホスト名を更新する必要があります。

<配置>

```
<type>zone_host</type>
```

```
<enforcement>strict</enforcement>
```

```
<host>wsstackovs-compute-4.localdomain</host>
```

</placement>

このMOPを実行する前に、前のセクションに示したホストの名前をUltra-Mチームが提供した新しいホスト名に更新します。新しいホストのインストール後、変更を保存します。

ステップ4：変更されたエクスポートテンプレートファイルを実行します。

ESCを実行して、次の操作を実行します。

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

ESCポータルから、1つ以上のVMが再び表示され、次に[Active]状態になります。

進行状況はESCの/var/log/esc/yangesc.logで追跡できます

例：

```
09:14:00,906 29-Jan-2018 INFO ===== UPDATE SERVICE REQUESTRECEIVED (UNDER TENANT) =====
09:14:00,906 29-Jan-2018 INFO Tenant name: Pcrf
09:14:00,906 29-Jan-2018 INFO Deployment name: WSP1-tmo
09:14:01,542 29-Jan-2018 INFO
09:14:01,542 29-Jan-2018 INFO ===== CONFID TRANSACTION ACCEPTED =====
09:16:33,947 29-Jan-2018 INFO
09:16:33,947 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:16:33,947 29-Jan-2018 INFO Type: VM_DEPLOYED
09:16:33,947 29-Jan-2018 INFO Status: SUCCESS
09:16:33,947 29-Jan-2018 INFO Status Code: 200
|
|
|
09:19:00,148 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:19:00,148 29-Jan-2018 INFO Type: VM_ALIVE
09:19:00,148 29-Jan-2018 INFO Status: SUCCESS
09:19:00,148 29-Jan-2018 INFO Status Code: 200
```

```
|
|
|
09:19:00,275 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:19:00,275 29-Jan-2018 INFO Type: SERVICE_UPDATED
09:19:00,275 29-Jan-2018 INFO Status: SUCCESS
09:19:00,275 29-Jan-2018 INFO Status Code: 200
```

ステップ5:VMに存在するPCRFサービスを確認します。

PCRFサービスがダウンしているかどうかを確認し、起動します。

```
[stack@XX-ospd ~]$ ssh root@
```

```
[root@XXXSM03 ~]# monsum
[root@XXXSM03 ~]# monit start all
```

VMがSM、OAM、またはアービターの場合は、以前に停止したsessionmgrサービスを起動します。

sessionmgr-xxxxxというタイトルのファイルごとに、service sessionmgr-xxxxx start:

```
[root@XXXSM03 init.d]# service sessionmgr-27717 start
```

診断がまだクリアされていない場合は、Cluster Manager VMからbuild_all.shを実行し、それぞれのVMでVM-initを実行します。

```
/var/gps/install/current/scripts/build_all.sh
```

```
ssh VM e.g. ssh pcrfclient01
/etc/init.d/vm-init
```

ステップ6：診断を実行してシステムステータスを確認します。

```
[root@XXXSM03 init.d]# diagnostics.sh
```

関連情報

- https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installati.. にアクセスしてください。
- https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installati.. にアクセスしてください。
- [テクニカル サポートとドキュメント – Cisco Systems](#)