

Ultra-M UCS 240M4サーバのマザーボードの交換 – vEPC

内容

[概要](#)

[背景説明](#)

[省略形](#)

[MoPのワークフロー](#)

[Ultra-Mセットアップでのマザーボードの交換](#)

[コンピューティングノードでのマザーボードの交換](#)

[コンピューティングノードでホストされるVMの特定](#)

[グレースフルパワーオフ](#)

[ケース1.コンピューティングノードホストのみSF VM](#)

[ケース2.コンピューティングノードホストCF/ESC/EM/UAS](#)

[マザーボードの交換](#)

[VMのリストア](#)

[ケース1.コンピューティングノードホストのみSF VM](#)

[ケース2.コンピューティングノードホストUAS、ESC、EMおよびCF](#)

[OSDコンピューティングノードでのマザーボードの交換](#)

[Cephをメンテナンスモードにする](#)

[Osd-ComputeノードでホストされるVMの特定](#)

[グレースフルパワーオフ](#)

[ケース1. OSDコンピューティングノードホストCF/ESC/EM/UAS](#)

[ケース2. OSDコンピューティングノードホストの自動配置/自動配置/EM/UAS](#)

[自動導入のCDBのバックアップ](#)

[Auto-ITからのBackup System.cfg](#)

[マザーボードの交換](#)

[メンテナンスモードからCephを移動](#)

[VMのリストア](#)

[ケース1. OSDコンピューティングノードホストCF、ESC、EM、UAS](#)

[ケース2. OSDコンピューティングノードホストのAuto-It、Auto-Deploy、EM、およびUAS](#)

[自動IT VMのリカバリ](#)

[コントローラノードのマザーボードの交換](#)

[コントローラのステータスを確認し、クラスタをメンテナンスモードにする](#)

[マザーボードの交換](#)

[クラスタステータスの復元](#)

概要

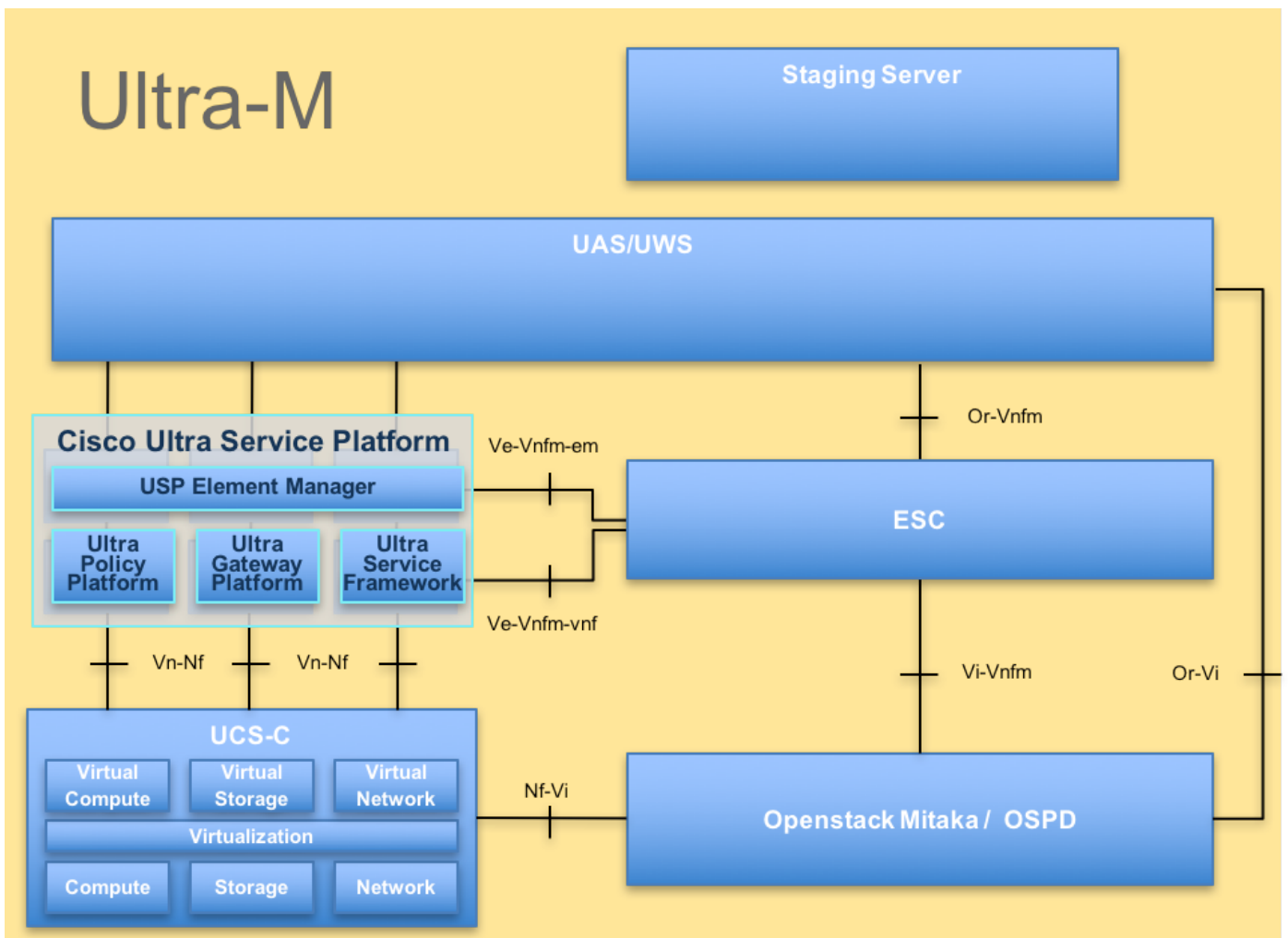
このドキュメントでは、StarOS Virtual Network Functions(VNF)をホストするUltra-Mセットアップでサーバのマザーボードの障害を交換するために必要な手順について説明します。

背景説明

Ultra-Mは、VNFの導入を簡素化するために設計された、パッケージ化および検証済みの仮想化モバイルネットワークコアソリューションです。OpenStackは、Ultra-M向けのVirtualized Infrastructure Manager(VIM)であり、次のノードタイプで構成されています。

- 計算
- オブジェクトストレージディスク – コンピューティング (OSD – コンピューティング)
- コントローラ
- OpenStackプラットフォーム – Director(OSPD)

Ultra-Mのアーキテクチャと関連するコンポーネントを次の図に示します。



UltraMアーキテクチャ

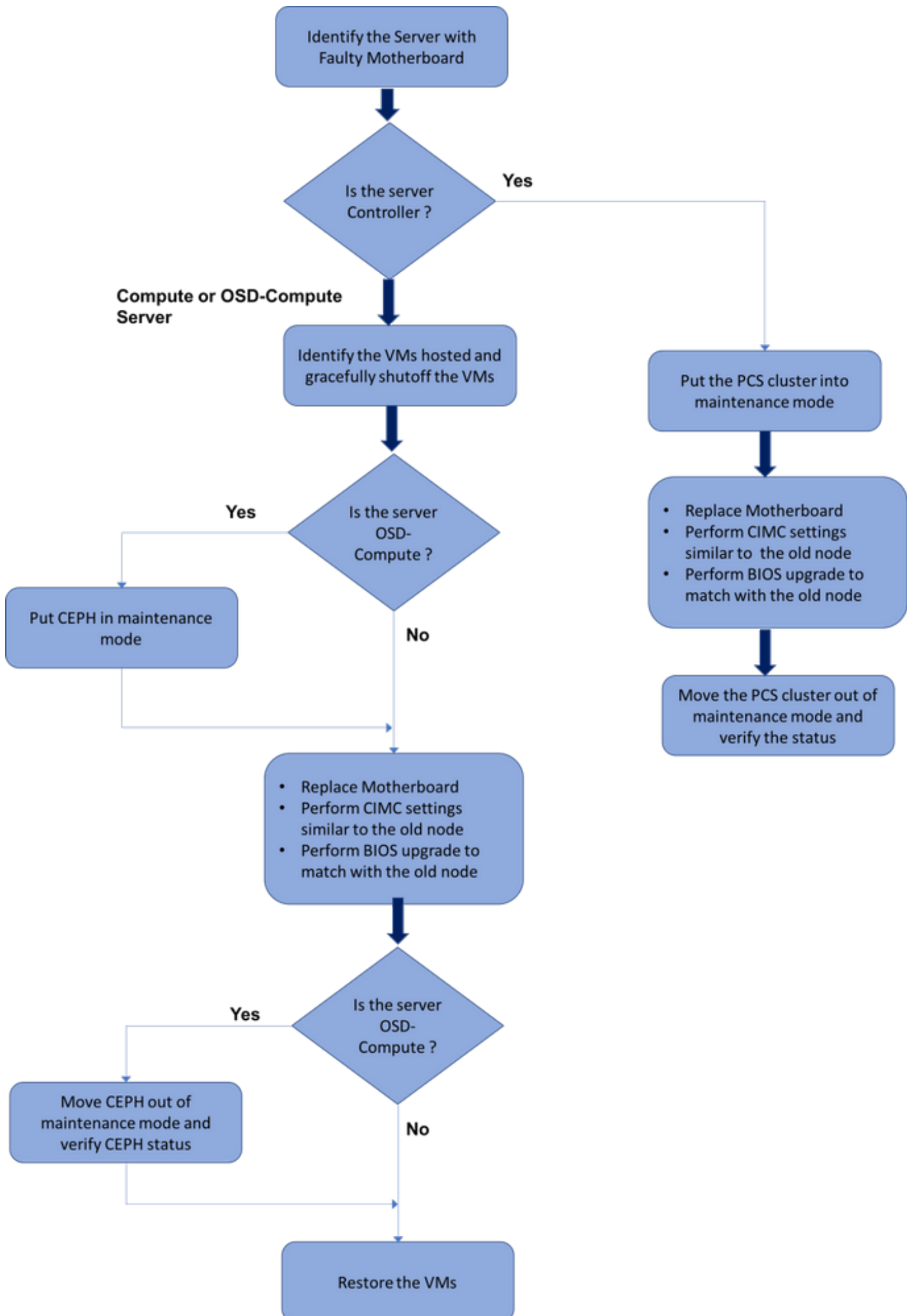
このドキュメントは、Cisco Ultra-Mプラットフォームに精通しているシスコの担当者を対象としており、サーバのマザーボード交換時にOpenStackおよびStarOS VNFレベルで実行する必要がある手順の詳細を説明しています。

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。

省略形

VNF	仮想ネットワーク機能
CF	制御機能
SF	サービス機能
ESC	Elastic Service Controller
MOP	手続きの方法
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
VM	仮想マシン
EM	エレメント マネージャ
UAS	Ultra Automation Services
UUID	汎用一意識別子

MoPのワークフロー



Ultra-Mセットアップでのマザーボードの交換

Ultra-Mセットアップでは、次のサーバタイプでマザーボードの交換が必要になる場合があります。コンピューティング、OSDコンピューティング、コントローラ

注：OpenStackをインストールしたブートディスクは、マザーボードの交換後に交換されます。したがって、ノードをオーバークラウドに追加する必要はありません。交換アクティビティの後にサーバの電源がオンになると、サーバ自体がオーバークラウドスタックに登録されます。

コンピューティングノードでのマザーボードの交換

このアクティビティの前に、コンピューティングノードでホストされているVMは正常にシャットオフされます。マザーボードを交換すると、VMが復元されます。

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。次の2つの可能性があります。

コンピューティングサーバにはSF VMのみが含まれます。

```
[stack@director ~]$ nova list --field name,host | grep compute-10
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d |
pod1-compute-10.localdomain |
```

コンピューティングサーバには、VMのCF/ESC/EM/UASの組み合わせが含まれています。

```
[stack@director ~]$ nova list --field name,host | grep compute-8
| 507d67c2-1d00-4321-b9d1-da879af524f8 | VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-88a2d6fa82ea | pod1-compute-8.localdomain |
| f9c0763a-4a4f-4bbd-af51-bc7545774be2 | VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229 | pod1-compute-8.localdomain |
| 75528898-ef4b-4d68-b05d-882014708694 | VNF2-ESC-ESC-0 | pod1-compute-8.localdomain |
| f5bd7b9c-476a-4679-83e5-303f0aae9309 | VNF2-UAS-uas-0 | pod1-compute-8.localdomain |
```

注：ここに示す出力では、最初の列がUUIDに対応し、2番目の列がVM名で、3番目の列がVMが存在するホスト名です。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。

グレースフルパワーオフ

ケース1.コンピューティングノードホストのみSF VM

StarOS VNFにログインし、SF VMに対応するカードを特定します。「コンピュータノードでホストされるVMを識別する」セクションで識別されるSF VMのUUIDを使用し、UUIDに対応するカードを識別します。

```
[local]VNF2# show card hardware
Tuesday night 08 16:49:42 UTC 2018
<snip>
Card 8:
  Card Type           : 4-Port Service Function Virtual Card
  CPU Packages        : 26 [#0, #1, #2, #3, #4, #5, #6, #7, #8, #9, #10, #11, #12, #13, #14,
#15, #16, #17, #18, #19, #20, #21, #22, #23, #24, #25]
  CPU Nodes           : 2
  CPU Cores/Threads   : 26
  Memory              : 98304M (qvpc-di-large)
  UUID/Serial Number  : 49AC5F22-469E-4B84-BADC-031083DB0533
<snip>
```

カードのステータスを確認します。

```
[local]VNF2# show card table
Tuesday night 08 16:52:53 UTC 2018
Slot      Card Type                               Oper State   SPOF  Attach
-----
1: CFC     Control Function Virtual Card             Active       No
2: CFC     Control Function Virtual Card             Standby      -
3: FC      4-Port Service Function Virtual Card     Active       No
4: FC      4-Port Service Function Virtual Card     Active       No
5: FC      4-Port Service Function Virtual Card     Active       No
6: FC      4-Port Service Function Virtual Card     Active       No
7: FC      4-Port Service Function Virtual Card     Active       No
8: FC      4-Port Service Function Virtual Card     Active       No
9: FC      4-Port Service Function Virtual Card     Active       No
10: FC     4-Port Service Function Virtual Card     Standby      -
```

カードがアクティブ状態の場合は、カードをスタンバイ状態にします。

```
[local]VNF2# card migrate from 8 to 10
```

VNFに対応するESCノードにログインし、SF VMのステータスを確認します。

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
  <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
  <state>VM_ALIVE_STATE</state>
  <vm_name> VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
  <state>VM_ALIVE_STATE</state>
<snip>
```

VM名を使用してSF VMを停止します。(VM名は「コンピュータノードでホストされるVMを特定する」セクションに記載◆◆)

```
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli vm-action STOP VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
```

停止すると、VMはシャットオフ状態になる必要があります。

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
  <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
  <state>VM_ALIVE_STATE</state>
  <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
  <state>VM_ALIVE_STATE</state>
  <vm_name>VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d</vm_name>
  <state>VM_SHUTOFF_STATE</state>
</snip>
```

ケース2.コンピュートノードホストCF/ESC/EM/UAS

StarOS VNFにログインし、CF VMに対応するカードを特定します。「コンピュートノードでホストされているVMを識別する」セクションで識別されるCF VMのUUIDを使用し、UUIDに対応するカードを見つけます。

```
[local]VNF2# show card hardware
Tuesday night 08 16:49:42 UTC 2018
<snip>
Card 2:
  Card Type           : Control Function Virtual Card
  CPU Packages        : 8 [#0, #1, #2, #3, #4, #5, #6, #7]
  CPU Nodes           : 1
  CPU Cores/Threads   : 8
  Memory              : 16384M (qvp-c-di-large)
  UUID/Serial Number  : F9C0763A-4A4F-4BBD-AF51-BC7545774BE2
</snip>
```

カードのステータスを確認します。

```
[local]VNF2# show card table
Tuesday night 08 16:52:53 UTC 2018
```

Slot	Card Type	Oper State	SPOF	Attach
1: CFC	Control Function Virtual Card	Standby	-	
2: CFC	Control Function Virtual Card	Active	No	
3: FC	4-Port Service Function Virtual Card	Active	No	
4: FC	4-Port Service Function Virtual Card	Active	No	
5: FC	4-Port Service Function Virtual Card	Active	No	
6: FC	4-Port Service Function Virtual Card	Active	No	
7: FC	4-Port Service Function Virtual Card	Active	No	
8: FC	4-Port Service Function Virtual Card	Active	No	
9: FC	4-Port Service Function Virtual Card	Active	No	
10: FC	4-Port Service Function Virtual Card	Standby	-	

カードがアクティブ状態の場合は、カードをスタンバイ状態にします。

```
[local]VNF2# card migrate from 2 to 1
```

VNFに対応するESCノードにログインし、VMのステータスを確認します。

```
[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
```

```

<snip>
<state>SERVICE_ACTIVE_STATE</state>
    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
    <state>VM_ALIVE_STATE</state>
    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
    <state>VM_ALIVE_STATE</state>
<deployment_name>VNF2-DEPLOYMENT-em</deployment_name>
    <vm_id>507d67c2-1d00-4321-b9d1-da879af524f8</vm_id>
    <vm_id>dc168a6a-4aeb-4e81-abd9-91d7568b5f7c</vm_id>
    <vm_id>9ffec58b-4b9d-4072-b944-5413bf7fcf07</vm_id>
    <state>SERVICE_ACTIVE_STATE</state>
    <vm_name>VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-88a2d6fa82ea</vm_name>
    <state>VM_ALIVE_STATE</state>
</snip>

```

VM名を使用して、CFとEM VMを1つずつ停止します。(VM名は「コンピュータノードでホストされるVMを特定する」セクションに記載◆◆)

```

[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli vm-action STOP VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229

[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli vm-action STOP VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-88a2d6fa82ea

```

停止したVMは、シャットオフ状態に入る必要があります。

```

[admin@VNF2-esc-esc-0 ~]$ cd /opt/cisco/esc/esc-confd/esc-cli
[admin@VNF2-esc-esc-0 esc-cli]$ ./esc_nc_cli get esc_datamodel | egrep --color
"<state>|<vm_name>|<vm_id>|<deployment_name>"
<snip>
<state>SERVICE_ACTIVE_STATE</state>
    <vm_name>VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229</vm_name>
    <state>VM_SHUTOFF_STATE</state>
    <vm_name>VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
    <state>VM_ALIVE_STATE</state>
<deployment_name>VNF2-DEPLOYMENT-em</deployment_name>
    <vm_id>507d67c2-1d00-4321-b9d1-da879af524f8</vm_id>
    <vm_id>dc168a6a-4aeb-4e81-abd9-91d7568b5f7c</vm_id>
    <vm_id>9ffec58b-4b9d-4072-b944-5413bf7fcf07</vm_id>
    <state>SERVICE_ACTIVE_STATE</state>
    <vm_name>VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-88a2d6fa82ea</vm_name>

```

<snip>
 コンピューティングノードでホストされているESCにログインし、マスター状態であるかどうかを確認します。存在する場合は、ESCをスタンバイモードに切り替えます。

```

[admin@VNF2-esc-esc-0 esc-cli]$ escadm status
0 ESC status=0 ESC Master Healthy

```

```

[admin@VNF2-esc-esc-0 ~]$ sudo service keepalived stop
Stopping keepalived: [ OK ]

```



```
[admin@VNF2-esc-esc-0 ~]$ escadm status
1 ESC status=0 In SWITCHING_TO_STOP state. Please check status after a while.
```

```
[admin@VNF2-esc-esc-0 ~]$ sudo reboot
Broadcast message from admin@vnf1-esc-esc-0.novalocal
(/dev/pts/0) at 13:32 ...
The system is going down for reboot NOW!
```

マザーボードの交換

UCS C240 M4サーバのマザーボードを交換する手順については、『[Cisco UCS C240 M4サーバインストールおよびサービスガイド](#)』を参照してください。

CIMC IPを使用してサーバにログインします。

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シリーズブラックマウントサーバBIOSアップグレードガイド](#)

VMのリストア

ケース1.コンピューティングノードホストのみSF VM

SF VMはnovaリストでエラー状態になります。

```
[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 |
```

ESCからSF VMをリカバリします。

```
[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.ZpRCGieUW
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:iETF:params:xml:ns:netconf:base:1.0" message-id="1">
  <ok/>
</rpc-reply>
```

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].
```

VNFでSFカードがスタンバイSFとして起動することを確認します。

ケース2.コンピュートノードホストUAS、ESC、EMおよびCF

UAS VMのリカバリ

ノバリストのUAS VMのステータスを確認し、削除します。

```
[stack@director ~]$ nova list | grep VNF2-UAS-uas-0
| 307a704c-a17c-4cdc-8e7a-3d6e7e4332fa | VNF2-UAS-uas-0
| ACTIVE | - | Running | VNF2-UAS-uas-
orchestration=172.168.11.10; VNF2-UAS-uas-management=172.168.10.3
```

```
[stack@tb5-ospd ~]$ nova delete VNF2-UAS-uas-0
Request to delete server VNF2-UAS-uas-0 has been accepted.
```

AutoVNF-UAS VMを回復するには、UAS-checkスクリプトを実行して状態を確認します。エラーを報告する必要があります。次に、`--fix`オプションを使用して再度実行し、欠落しているUAS VMを再作成します。

```
[stack@director ~]$ cd /opt/cisco/usp/uas-installer/scripts/
[stack@director scripts]$ ./uas-check.py auto-vnf VNF2-UAS
2017-12-08 12:38:05,446 - INFO: Check of AutoVNF cluster started
2017-12-08 12:38:07,925 - INFO: Instance 'vnf1-UAS-uas-0' status is 'ERROR'
2017-12-08 12:38:07,925 - INFO: Check completed, AutoVNF cluster has recoverable errors
```

```
[stack@director scripts]$ ./uas-check.py auto-vnf VNF2-UAS --fix
2017-11-22 14:01:07,215 - INFO: Check of AutoVNF cluster started
2017-11-22 14:01:09,575 - INFO: Instance VNF2-UAS-uas-0' status is 'ERROR'
2017-11-22 14:01:09,575 - INFO: Check completed, AutoVNF cluster has recoverable errors
2017-11-22 14:01:09,778 - INFO: Removing instance VNF2-UAS-uas-0'
2017-11-22 14:01:13,568 - INFO: Removed instance VNF2-UAS-uas-0'
2017-11-22 14:01:13,568 - INFO: Creating instance VNF2-UAS-uas-0' and attaching volume 'VNF2-
UAS-uas-vol-0'
2017-11-22 14:01:49,525 - INFO: Created instance 'VNF2-UAS-uas-0'
```

AutoVNF-UASにログインします。数分間待つと、UASが正常な状態に戻ります。

```
VNF2-autovnf-uas-0#show uas
uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.17.181.101
INSTANCE IP STATE ROLE
-----
172.17.180.6 alive CONFD-SLAVE
172.17.180.7 alive CONFD-MASTER
172.17.180.9 alive NA
```

ESC VMのリカバリ

ノバリストからESC VMのステータスを確認し、削除します。

```
stack@director scripts]$ nova list |grep ESC-1
| c566efbf-1274-4588-a2d8-0682e17b0d41 | VNF2-ESC-ESC-
1 | ACTIVE | - | Running | VNF2-
UAS-uas-orchestration=172.168.11.14; VNF2-UAS-uas-
```

management=172.168.10.4

```
[stack@director scripts]$ nova delete VNF2-ESC-ESC-1
Request to delete server VNF2-ESC-ESC-1 has been accepted.
```

AutoVNF-UASからESC導入トランザクションを検索し、トランザクションのログでboot_vm.pyコマンド行を見つけて、ESCインスタンスを作成します。

```
ubuntu@VNF2-uas-uas-0:~$ sudo -i
root@VNF2-uas-uas-0:~# confd_cli -u admin -C
Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on VNF2-uas-uas-0
VNF2-uas-uas-0#show transaction
```

TX ID TIMESTAMP	TX TYPE STATUS	DEPLOYMENT ID
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b 29T02:01:27.750692-00:00	vnf-deployment deployment-success	VNF2-DEPLOYMENT
73d9c540-d4a8-11e7-bb72-fa163ef8df2b 29T01:56:02.133663-00:00	vnfm-deployment deployment-success	VNF2-ESC

```
VNF2-uas-uas-0#show logs 73d9c540-d4a8-11e7-bb72-fa163ef8df2b | display xml
```

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <logs xmlns="http://www.cisco.com/usp/nfv/usp-autovnf-oper">
    <tx-id>73d9c540-d4a8-11e7-bb72-fa163ef8df2b</tx-id>
    <log>2017-11-29 01:56:02,142 - VNF Deployment RPC triggered for deployment: VNF2-ESC,
deactivate: 0
2017-11-29 01:56:02,179 - Notify deployment
..
2017-11-29 01:57:30,385 - Creating VNF 'VNF2-ESC-ESC-1' with [python //opt/cisco/vnf-
staging/bootvm.py VNF2-ESC-ESC-1 --flavor VNF2-ESC-ESC-flavor --image 3fe6b197-961b-4651-af22-
dfd910436689 --net VNF2-UAS-uas-management --gateway_ip 172.168.10.1 --net VNF2-UAS-uas-
orchestration --os_auth_url http://10.1.2.5:5000/v2.0 --os_tenant_name core --os_username *****
--os_password ***** --bs_os_auth_url http://10.1.2.5:5000/v2.0 --bs_os_tenant_name core --
bs_os_username ***** --bs_os_password ***** --esc_ui_startup false --esc_params_file
/tmp/esc_params.cfg --encrypt_key ***** --user_pass ***** --user_confid_pass ***** --kad_vif
eth0 --kad_vip 172.168.10.7 --ipaddr 172.168.10.6 dhcp --ha_node_list 172.168.10.3 172.168.10.6
--file root:0755:/opt/cisco/esc/esc-
scripts/esc_volume_em_staging.sh:/opt/cisco/usp/uas/autovnf/vnfms/esc-
scripts/esc_volume_em_staging.sh --file root:0755:/opt/cisco/esc/esc-
scripts/esc_vpc_chassis_id.py:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_vpc_chassis_id.py
--file root:0755:/opt/cisco/esc/esc-scripts/esc-vpc-di-internal-
keys.sh:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc-vpc-di-internal-keys.sh
boot_vm.py行をシェルスクリプトファイル(esc.sh)に保存し、すべてのユーザ名*****およびパス
ワード*****行を正しい情報(通常はcore/<PASSWORD>)で更新します。 --encrypt_keyオプション
も削除する必要があります。user_passとuser_confid_passの場合、username:パスワード(例
: admin:<PASSWORD>)。
```

running-configからbootvm.pyへのURLを探し、bootvm.pyファイルをAutoVNF UAS VMに取得し
ます。この場合、10.1.2.3はAutoIT VMのIPです。

```
root@VNF2-uas-uas-0:~# confd_cli -u admin -C
Welcome to the ConfD CLI
admin connected from 127.0.0.1 using console on VNF2-uas-uas-0
VNF2-uas-uas-0#show running-config autovnf-vnfm:vnfm
...
configs bootvm
  value http:// 10.1.2.3:80/bundles/5.1.7-2007/vnfm-bundle/bootvm-2_3_2_155.py
```

!

```
root@VNF2-uas-uas-0:~# wget http://10.1.2.3:80/bundles/5.1.7-2007/vnfm-bundle/bootvm-2\_3\_2\_155.py
--2017-12-01 20:25:52-- http://10.1.2.3 /bundles/5.1.7-2007/vnfm-bundle/bootvm-2_3_2_155.py
Connecting to 10.1.2.3:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 127771 (125K) [text/x-python]
Saving to: 'bootvm-2_3_2_155.py'
100%[=====>]
127,771 --.-K/s in 0.001s
2017-12-01 20:25:52 (173 MB/s) - 'bootvm-2_3_2_155.py' saved [127771/127771]

/tmp/esc_params.cfg ファイルを作成します。
```

```
root@VNF2-uas-uas-0:~# echo "openstack.endpoint=publicURL" > /tmp/esc_params.cfg
シェルスクリプトを実行して、UASノードからESCを展開します。
```

```
root@VNF2-uas-uas-0:~# /bin/sh esc.sh
+ python ./bootvm.py VNF2-ESC-ESC-1 --flavor VNF2-ESC-ESC-flavor --image 3fe6b197-961b-4651-af22-dfd910436689
--net VNF2-UAS-uas-management --gateway_ip 172.168.10.1 --net VNF2-UAS-uas-orchestration --os_auth_url http://10.1.2.5:5000/v2.0 --os_tenant_name core --os_username core --os_password <PASSWORD> --bs_os_auth_url http://10.1.2.5:5000/v2.0 --bs_os_tenant_name core --bs_os_username core --bs_os_password <PASSWORD>
--esc_ui_startup false --esc_params_file /tmp/esc_params.cfg --user_pass admin:<PASSWORD> --user_confid_pass admin:<PASSWORD> --kad_vif eth0 --kad_vip 172.168.10.7 --ipaddr 172.168.10.6 dhcp --ha_node_list 172.168.10.3 172.168.10.6 --file root:0755:/opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_volume_em_staging.sh --file root:0755:/opt/cisco/esc/esc-scripts/esc_vpc_chassis_id.py:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc_vpc_chassis_id.py --file root:0755:/opt/cisco/esc/esc-scripts/esc-vpc-di-internal-keys.sh:/opt/cisco/usp/uas/autovnf/vnfms/esc-scripts/esc-vpc-di-internal-keys.sh
新しいESCにログインし、バックアップ状態を確認します。
```

```
ubuntu@VNF2-uas-uas-0:~$ ssh admin@172.168.11.14
...
#####
# ESC on VNF2-esc-esc-1.novalocal is in BACKUP state.
#####
```

```
[admin@VNF2-esc-esc-1 ~]$ escadm status
0 ESC status=0 ESC Backup Healthy
```

```
[admin@VNF2-esc-esc-1 ~]$ health.sh
===== ESC HA (BACKUP) =====
ESC HEALTH PASSED
```

CFおよびEM VMをESCからリカバリする

novaリストからCFおよびEM VMのステータスを確認します。これらはERROR状態である必要がありません。

```
[stack@director ~]$ source corerc
[stack@director ~]$ nova list --field name,host,status |grep -i err
| 507d67c2-1d00-4321-b9d1-da879af524f8 | VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-
88a2d6fa82ea | None | ERROR|
| f9c0763a-4a4f-4bbd-af51-bc7545774be2 | VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-
3812653ee229 |None | ERROR
```

ESC Masterにログインし、影響を受ける各EMおよびCF VMに対してrecovery-vm-actionを実行します。我慢なさい。ESCはリカバリアクションをスケジュールし、数分間実行されない可能性があります。yanesc.logを監視します。

```
sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO
```

```
[admin@VNF2-esc-esc-0 ~]$ sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO
VNF2-DEPLOYMENT-_VNF2-D_0_a6843886-77b4-4f38-b941-74eb527113a8
[sudo] password for admin:
```

Recovery VM Action

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

```
[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-DEPLOYMENT-
_VNF2-D_0_a6843886-77b4-4f38-b941-74eb527113a8]
```

新しいEMにログインし、EMの状態がupであることを確認します。

```
ubuntu@VNF2vnfddeploymentem-1:~$ /opt/cisco/ncs/current/bin/ncs_cli -u admin -C
admin connected from 172.17.180.6 using ssh on VNF2vnfddeploymentem-1
admin@scm# show ems
EM          VNFM
ID  SLA  SCM  PROXY
-----
2   up   up   up
3   up   up   up
```

StarOS VNFにログインし、CFカードがスタンバイ状態であることを確認します。

ESC回復障害の処理

予期しない状態が原因でESCがVMの起動に失敗する場合は、マスターESCをリブートしてESCスイッチオーバーを実行することを推奨します。ESCスイッチオーバーには約1分かかります。

す。新しいマスターESCでスクリプトhealth.shを実行し、ステータスがupかどうかを確認します。マスターESCを使用してVMを開始し、VMの状態を修正します。この回復タスクの完了には最大5分かかります。

/var/log/esc/yangesc.logと/var/log/esc/escmanager.logを監視できます。5～7分後にVMがリカバリされない場合は、影響を受けるVMを手動でリカバリする必要があります。

OSDコンピュータノードでのマザーボードの交換

アクティビティの前に、コンピューティングノードでホストされているVMは正常にシャットオフされ、Cephはメンテナンスモードになります。マザーボードを交換すると、VMが復元され、Cephがメンテナンスモードから外れます。

Cephをメンテナンスモードにする

ceph osdツリーの状態がサーバでアップしていることを確認します。

```
[heat-admin@pod1-osd-compute-1 ~]$ sudo ceph osd tree
```

ID	WEIGHT	TYPE	NAME	UP/DOWN	REWEIGHT	PRIMARY-AFFINITY
-1	13.07996	root	default			
-2	4.35999	host	pod1-osd-compute-0			
0	1.09000		osd.0	up	1.00000	1.00000
3	1.09000		osd.3	up	1.00000	1.00000
6	1.09000		osd.6	up	1.00000	1.00000
9	1.09000		osd.9	up	1.00000	1.00000
-3	4.35999	host	pod1-osd-compute-2			
1	1.09000		osd.1	up	1.00000	1.00000
4	1.09000		osd.4	up	1.00000	1.00000
7	1.09000		osd.7	up	1.00000	1.00000
10	1.09000		osd.10	up	1.00000	1.00000
-4	4.35999	host	pod1-osd-compute-1			
2	1.09000		osd.2	up	1.00000	1.00000
5	1.09000		osd.5	up	1.00000	1.00000
8	1.09000		osd.8	up	1.00000	1.00000
11	1.09000		osd.11	up	1.00000	1.00000

OSD Computeノードにログインし、Cephをメンテナンスモードにします。

```
[root@pod1-osd-compute-1 ~]# sudo ceph osd set norebalance
[root@pod1-osd-compute-1 ~]# sudo ceph osd set noout
```

```
[root@pod1-osd-compute-1 ~]# sudo ceph status
```

```
cluster eb2bb192-b1c9-11e6-9205-525400330666
health HEALTH_WARN
noout,norebalance,sortbitwise,require_jewel_osds flag(s) set
monmap e1: 3 mons at {pod1-controller-0=11.118.0.40:6789/0,pod1-controller-1=11.118.0.41:6789/0,pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 pod1-controller-0,pod1-controller-1,pod1-controller-2
osdmap e194: 12 osds: 12 up, 12 in
flags noout,norebalance,sortbitwise,require_jewel_osds
pgmap v584865: 704 pgs, 6 pools, 531 GB data, 344 kobjects
1585 GB used, 11808 GB / 13393 GB avail
```

704 active+clean

client io 463 kB/s rd, 14903 kB/s wr, 263 op/s rd, 542 op/s wr

注：Cephを削除すると、VNF HD RAIDはDegraded状態になりますが、HDDにアクセスできる必要があります。

Osd-ComputeノードでホストされるVMの特定

OSD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。次の2つの可能性があります。

osd-computeサーバには、VMのElement Manager(EM)/UAS/Auto-Deploy/Auto-ITの組み合わせが含まれています。

```
[stack@director ~]$ nova list --field name,host | grep osd-compute-0
| c6144778-9afd-4946-8453-78c817368f18 | AUTO-DEPLOY-VNF2-uas-0 | pod1-osd-compute-0.localdomain
|
| 2d051522-bce2-4809-8d63-0c0e17f251dc | AUTO-IT-VNF2-uas-0 | pod1-osd-compute-0.localdomain |
| 507d67c2-1d00-4321-b9d1-da879af524f8 | VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-
88a2d6fa82ea | pod1-osd-compute-0.localdomain |
| f5bd7b9c-476a-4679-83e5-303f0aae9309 | VNF2-UAS-uas-0 | pod1-osd-compute-0.localdomain |
```

コンピューティングサーバには、VMの制御機能(CF)/Elastic Services Controller(ESC)/Element Manager(EM)/(UAS)の組み合わせが含まれています。

```
[stack@director ~]$ nova list --field name,host | grep osd-compute-1
| 507d67c2-1d00-4321-b9d1-da879af524f8 | VNF2-DEPLOYM_XXXX_0_c8d98f0f-d874-45d0-af75-
88a2d6fa82ea | pod1-compute-8.localdomain |
| f9c0763a-4a4f-4bbd-af51-bc7545774be2 | VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-
3812653ee229 | pod1-compute-8.localdomain |
| 75528898-ef4b-4d68-b05d-882014708694 | VNF2-ESC-ESC-
0 | pod1-compute-8.localdomain |
| f5bd7b9c-476a-4679-83e5-303f0aae9309 | VNF2-UAS-uas-
0 | pod1-compute-8.localdomain |
```

注：ここに示す出力では、最初の列がUUIDに対応し、2番目の列がVM名で、3番目の列がVMが存在するホスト名です。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。

グレースフルパワーオフ

ケース1. OSDコンピュートノードホストCF/ESC/EM/UAS

CF/ESC/EM/UAS VMの正常な電源投入手順は、VMがコンピューティングまたはOSDコンピューティングノードでホストされているかどうかに関係なく同じです。コンピューティングノードのマザーボードの交換の手順に従い、VMの電源を正常にオフにします。

ケース2. OSDコンピュートノードホストの自動配置/自動配置/EM/UAS

自動導入のCDBのバックアップ

自動デプロイ confd cdbデータを定期的またはアクティブ化/非アクティブ化ごとにバックアップし、ファイルをバックアップサーバに保存します。Autodeployは冗長ではないため、このデータが失われた場合、展開を正常に非アクティブ化することはできません。

AutoDeploy VMおよびbackup confd cdbディレクトリにログインします。

```
ubuntu@auto-deploy-iso-2007-uas-0:~ $sudo -i
```

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

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

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

注:autodeploy_cdb_backup.tarをコピーして、サーバをバックアップします。

Auto-ITからのBackup System.cfg

system.cfgファイルのバックアップを取って、サーバをバックアップします。

```
Auto-it = 10.1.1.2
Backup server = 10.2.2.2
```

```
[stack@director ~]$ ssh ubuntu@10.1.1.2
ubuntu@10.1.1.2's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-76-generic x86_64)
```

```
* Documentation:  https://help.ubuntu.com/
```

```
System information as of Wed Jun 13 16:21:34 UTC 2018
```

```
System load:  0.02                Processes:            87
Usage of /:   15.1% of 78.71GB     Users logged in:    0
Memory usage: 13%                 IP address for eth0: 172.16.182.4
Swap usage:   0%
```

```
Graph this data and manage this system at:
https://landscape.canonical.com/
```

```
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud
```

```
Cisco Ultra Services Platform (USP)
Build Date: Wed Feb 14 12:58:22 EST 2018
Description: UAS build assemble-uas#1891
shal: bf02ced
```

```
ubuntu@auto-it-vnf-uas-0:~$ scp -r /opt/cisco/usp/uploads/system.cfg root@10.2.2.2:/home/stack
```



```
root@10.2.2.2's password:
system.cfg
```

```
100% 565 0.6KB/s 00:00
ubuntu@auto-it-vnf-uas-0:~$
```

注：EM/UAS VMの正常な電源投入手順は、VMがコンピューティングまたはOSDコンピューティングノードのどちらでホストされているかに関係なく同じです。

これらのVMの電源を正常にオフにするには、「コンピュートノードのマザーボードの交換」の手順に従います。

マザーボードの交換

UCS C240 M4サーバのマザーボードを交換する手順については、『[Cisco UCS C240 M4サーバインストールおよびサービスガイド](#)』を参照してください。

CIMC IPを使用してサーバにログインします。

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](#)

メンテナンスモードからCephを移動

OSDコンピュートノードにログインし、Cephをメンテナンスモードから外します。

```
[root@pod1-osd-compute-1 ~]# sudo ceph osd unset norebalance
[root@pod1-osd-compute-1 ~]# sudo ceph osd unset noout

[root@pod1-osd-compute-1 ~]# sudo ceph status

cluster eb2bb192-b1c9-11e6-9205-525400330666
health HEALTH_OK
monmap e1: 3 mons at {pod1-controller-0=11.118.0.40:6789/0,pod1-controller-1=11.118.0.41:6789/0,pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 pod1-controller-0,pod1-controller-1,pod1-controller-2
osdmap e196: 12 osds: 12 up, 12 in
flags sortbitwise,require_jewel_osds
pgmap v584954: 704 pgs, 6 pools, 531 GB data, 344 kobjects
1585 GB used, 11808 GB / 13393 GB avail
704 active+clean
client io 12888 kB/s wr, 0 op/s rd, 81 op/s wr
```

VMのリストア

ケース1. OSDコンピュートノードホストCF、ESC、EM、UAS

CF/ESC/EM/UAS VMの復元手順は、VMがコンピューティングまたはOSDコンピューティングノードのどちらでホストされているかに関係なく、同じです。ケース2の手順に従って、コンピュー

トノードホストCF/ESC/EM/UASをVMをリストアします。

ケース2. OSDコンピュートノードホストのAuto-It、Auto-Deploy、EM、およびUAS

自動展開VMの回復

OSPDから、VMの自動配置が影響を受けていても[ACTIVE/Running]と表示されている場合は、まずVMを削除する必要があります。自動導入が影響を受けなかった場合は、「Auto-it VMのリカバリ」にスキップします。

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

```
[stack@director ~]$ ./auto-deploy-booting.sh --floating-ip 10.1.2.7 --delete
自動配置が削除されたら、同じfloatingipアドレスを使用して再び作成します。
```

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

```
[stack@director scripts]$ ./auto-deploy-booting.sh --floating-ip 10.1.2.7
```

```
2017-11-17 07:05:03,038 - INFO: Creating AutoDeploy deployment (1 instance(s)) on
'http://10.84.123.4:5000/v2.0' tenant 'core' user 'core', ISO 'default'
2017-11-17 07:05:03,039 - INFO: Loading image 'auto-deploy-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2' from '/opt/cisco/usp/uas-installer/images/usp-uas-1.0.1-1504.qcow2'
2017-11-17 07:05:14,603 - INFO: Loaded image 'auto-deploy-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2'
2017-11-17 07:05:15,787 - INFO: Assigned floating IP '10.1.2.7' to IP '172.16.181.7'
2017-11-17 07:05:15,788 - INFO: Creating instance 'auto-deploy-ISO-5-1-7-2007-uas-0'
2017-11-17 07:05:42,759 - INFO: Created instance 'auto-deploy-ISO-5-1-7-2007-uas-0'
2017-11-17 07:05:42,759 - INFO: Request completed, floating IP: 10.1.2.7
```

Autodeploy.cfgファイル、ISOおよび**confd_backup tar**ファイルをバックアップ・サーバからコピーし、VMを自動配置し、バックアップtarファイルから**confd cdb**ファイルをリストアします。

```
ubuntu@auto-deploy-iso-2007-uas-0:~# sudo -i
```

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

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

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

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

以前のトランザクションをチェックして、confdが正しくロードされたことを確認します。**autodeploy.cfg**を新しいosd-compute名で更新します。セクション – 最終手順：AutoDeploy構成の更新。

```
root@auto-deploy-iso-2007-uas-0:~# confd_cli -u admin -C
```

```
Welcome to the ConfD CLI
```

```
admin connected from 127.0.0.1 using console on auto-deploy-iso-2007-uas-0
```

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

```

                SERVICE
SITE
                DEPLOYMENT
SITE TX      AUTOVNF  VNF  AUTOVNF
TX ID      TX TYPE  ID      ID  ID      ID  TX ID
STATUS      ID      ID      ID  ID      ID  TX ID
-----
1512571978613  service-deployment  tb5bxb          2017-12-06T14:52:59.412+00:00  deployment-success
```

```
auto-deploy-iso-2007-uas-0# exit
```

自動IT VMのリカバリ

OSPDから、自動it VMが影響を受けていてもACTIVE/Runningと表示される場合は、削除する必要があります。auto-itが影響を受けない場合は、次に進みます。

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

```
[stack@director ~]$ ./ auto-it-vnf-staging.sh --floating-ip 10.1.2.8 --delete
```

Auto-IT-VNFステージングスクリプトを実行してAuto-ITを再作成する：

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

```
[stack@director scripts]$ ./auto-it-vnf-staging.sh --floating-ip 10.1.2.8
```

```
2017-11-16 12:54:31,381 - INFO: Creating StagingServer deployment (1 instance(s)) on
'http://10.84.123.4:5000/v2.0' tenant 'core' user 'core', ISO 'default'
2017-11-16 12:54:31,382 - INFO: Loading image 'auto-it-vnf-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2' from '/opt/cisco/usp/uas-installer/images/usp-uas-1.0.1-1504.qcow2'
2017-11-16 12:54:51,961 - INFO: Loaded image 'auto-it-vnf-ISO-5-1-7-2007-usp-uas-1.0.1-
1504.qcow2'
2017-11-16 12:54:53,217 - INFO: Assigned floating IP '10.1.2.8' to IP '172.16.181.9'
2017-11-16 12:54:53,217 - INFO: Creating instance 'auto-it-vnf-ISO-5-1-7-2007-uas-0'
2017-11-16 12:55:20,929 - INFO: Created instance 'auto-it-vnf-ISO-5-1-7-2007-uas-0'
2017-11-16 12:55:20,930 - INFO: Request completed, floating IP: 10.1.2.8
```

ISOイメージをリロードします。この場合、Auto-IT IPアドレスは10.1.2.8です。ロードには数分かかります。

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

```
[stack@director isos]$ curl -F file=@usp-5_1_7-2007.iso http://10.1.2.8:5001/isos
{
  "iso-id": "5.1.7-2007"
}
```

to check the ISO image:

```
[stack@director isos]$ curl http://10.1.2.8:5001/isos
```

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

OSPD Auto-DeployディレクトリからAuto-IT VMにVNF system.cfgファイルをコピーします。

```
[stack@director autodeploy]$ scp system-vnf* ubuntu@10.1.2.8:.
```

```
ubuntu@10.1.2.8's password:
```

```
system-
```

```
vnf1.cfg
```

```
100% 1197      1.2KB/s   00:00
```

```
system-vnf2.cfg
```

```
100% 1197      1.2KB/s   00:00
```

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

```
/home/ubuntu
```

```
ubuntu@auto-it-vnf-iso-2007-uas-0:~$ ls
```

```
system-vnf1.cfg system-vnf2.cfg
```

注：EMおよびUAS VMの回復手順は、VMがコンピューティングまたはOSDコンピューティングのどちらでホストされているかに関係なく同じです。これらのVMの電源を正常にオフにするには、「コンピュータノードのマザーボードの交換」の手順に従います。

コントローラノードのマザーボードの交換

コントローラのステータスを確認し、クラスタをメンテナンスモードにする

OSPDからコントローラにログインし、PCが正常な状態であることを確認します。3つのコントローラすべてがオンラインとGaleraに3つのコントローラすべてがマスターとして表示されます。

```
[heat-admin@pod1-controller-0 ~]$ sudo pcs status
```

```
Cluster name: tripleo_cluster
```

```
Stack: corosync
```

```
Current DC: pod1-controller-2 (version 1.1.15-11.e17_3.4-e174ec8) - partition with quorum
```

```
Last updated: Mon Dec 4 00:46:10 2017 Last change: Wed Nov 29 01:20:52 2017 by hacluster via crmd on pod1-controller-0
```

```
3 nodes and 22 resources configured
```

```
Online: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
```

```
Full list of resources:
```

```
ip-11.118.0.42 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
```

```
ip-11.119.0.47 (ocf::heartbeat:IPaddr2): Started pod1-controller-2
```

```
ip-11.120.0.49 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
```

```
ip-192.200.0.102 (ocf::heartbeat:IPAddr2): Started pod1-controller-2
Clone Set: haproxy-clone [haproxy]
Started: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
Master/Slave Set: galera-master [galera]
Masters: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
ip-11.120.0.47 (ocf::heartbeat:IPAddr2): Started pod1-controller-2
Clone Set: rabbitmq-clone [rabbitmq]
Started: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
Master/Slave Set: redis-master [redis]
Masters: [ pod1-controller-2 ]
Slaves: [ pod1-controller-0 pod1-controller-1 ]
ip-10.84.123.35 (ocf::heartbeat:IPAddr2): Started pod1-controller-1
openstack-cinder-volume (systemd:openstack-cinder-volume): Started pod1-controller-2
my-ipmilan-for-controller-0 (stonith:fence_ipmilan): Started pod1-controller-0
my-ipmilan-for-controller-1 (stonith:fence_ipmilan): Started pod1-controller-0
my-ipmilan-for-controller-2 (stonith:fence_ipmilan): Started pod1-controller-0
```

Daemon Status:

```
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled
```

クラスタをメンテナンスモードにします。

```
[heat-admin@pod1-controller-0 ~]$ sudo pcs cluster standby
```

```
[heat-admin@pod1-controller-0 ~]$ sudo pcs status
```

```
Cluster name: tripleo_cluster
Stack: corosync
Current DC: pod1-controller-2 (version 1.1.15-11.e17_3.4-e174ec8) - partition with quorum
Last updated: Mon Dec 4 00:48:24 2017 Last change: Mon Dec 4 00:48:18 2017 by root via
crm_attribute on pod1-controller-0
```

3 nodes and 22 resources configured

Node pod1-controller-0: standby

```
Online: [ pod1-controller-1 pod1-controller-2 ]
```

Full list of resources:

```
ip-11.118.0.42 (ocf::heartbeat:IPAddr2): Started pod1-controller-1
ip-11.119.0.47 (ocf::heartbeat:IPAddr2): Started pod1-controller-2
ip-11.120.0.49 (ocf::heartbeat:IPAddr2): Started pod1-controller-1
ip-192.200.0.102 (ocf::heartbeat:IPAddr2): Started pod1-controller-2
Clone Set: haproxy-clone [haproxy]
Started: [ pod1-controller-1 pod1-controller-2 ]
Stopped: [ pod1-controller-0 ]
Master/Slave Set: galera-master [galera]
Masters: [ pod1-controller-1 pod1-controller-2 ]
Slaves: [ pod1-controller-0 ]
ip-11.120.0.47 (ocf::heartbeat:IPAddr2): Started pod1-controller-2
Clone Set: rabbitmq-clone [rabbitmq]
Started: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
Master/Slave Set: redis-master [redis]
Masters: [ pod1-controller-2 ]
Slaves: [ pod1-controller-1 ]
Stopped: [ pod1-controller-0 ]
ip-10.84.123.35 (ocf::heartbeat:IPAddr2): Started pod1-controller-1

openstack-cinder-volume (systemd:openstack-cinder-volume): Started pod1-controller-2
my-ipmilan-for-controller-0 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-controller-1 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-controller-2 (stonith:fence_ipmilan): Started pod1-controller-2
```

マザーボードの交換

UCS C240 M4サーバのマザーボードを交換する手順については、『[Cisco UCS C240 M4サーバインストールおよびサービスガイド](#)』を参照してください。

CIMC IPを使用してサーバにログインします。

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シリーズブラックマウントサーバBIOSアップグレードガイド](#)

クラスタステータスの復元

影響を受けたコントローラにログインし、unstandbyを設定してスタンバイモードを削除します。コントローラがクラスタでオンラインになり、Galeraが3つのコントローラすべてをマスターとして表示するかどうかを確認します。これには数分かかることがあります。

```
[heat-admin@pod1-controller-0 ~]$ sudo pcs cluster unstandby

[heat-admin@pod1-controller-0 ~]$ sudo pcs status
Cluster name: tripleo_cluster
Stack: corosync
Current DC: pod1-controller-2 (version 1.1.15-11.el7_3.4-e174ec8) - partition with quorum
Last updated: Mon Dec 4 01:08:10 2017 Last change: Mon Dec 4 01:04:21 2017 by root via
crm_attribute on pod1-controller-0

3 nodes and 22 resources configured

Online: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]

Full list of resources:

ip-11.118.0.42 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
ip-11.119.0.47 (ocf::heartbeat:IPaddr2): Started pod1-controller-2
ip-11.120.0.49 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
ip-192.200.0.102 (ocf::heartbeat:IPaddr2): Started pod1-controller-2
Clone Set: haproxy-clone [haproxy]
Started: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
Master/Slave Set: galera-master [galera]
Masters: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
ip-11.120.0.47 (ocf::heartbeat:IPaddr2): Started pod1-controller-2
Clone Set: rabbitmq-clone [rabbitmq]
Started: [ pod1-controller-0 pod1-controller-1 pod1-controller-2 ]
Master/Slave Set: redis-master [redis]
Masters: [ pod1-controller-2 ]
Slaves: [ pod1-controller-0 pod1-controller-1 ]
ip-10.84.123.35 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
openstack-cinder-volume (systemd:openstack-cinder-volume): Started pod1-controller-2
my-ipmilan-for-controller-0 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-controller-1 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-controller-2 (stonith:fence_ipmilan): Started pod1-controller-2

Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled
```