

Remplacement des composants défectueux sur le serveur UCS C240 M4 - vEPC

Contenu

[Introduction](#)

[Informations générales](#)

[Abréviations](#)

[Processus du balai](#)

[Conditions préalables](#)

[Sauvegarde](#)

[Composant RMA - Noeud Compute/OSD-Compute](#)

[Identifiez les VMs hébergées dans le noeud Compute/OSD-Compute](#)

[Gracieux mettez hors tension](#)

[Le noeud de calcul de l'affaire 1. héberge seulement la VM SF](#)

[Hôtes CF/ESC/EM/UAS de noeud de l'affaire 2. Compute/OSD-Compute](#)

[Remplacez le composant défectueux du noeud Compute/OSD-Compute](#)

[Restaurez les VMs](#)

[Le noeud de calcul de l'affaire 1. héberge seulement la VM SF](#)

[CF, ESC, EM et UAS d'hôtes de noeud de l'affaire 2. Compute/OSD-Compute](#)

[Panne de reprise d'ESC de traitement](#)

[Automatique-déployez la mise à jour de la configuration](#)

[Composant RMA - Noeud de contrôleur](#)

[Prévérification](#)

[Déplacez la batterie de contrôleur au mode maintenance](#)

[Remplacez le composant défectueux du noeud de contrôleur](#)

[Mettez sous tension le serveur](#)

Introduction

Ce document décrit l'étape nécessaire pour remplacer des composants défectueux mentionnés ici dans un serveur de l'Unified Computing System (UCS) dans un Ultra-M installé qui héberge des fonctions réseau virtuelles de StarOS (VNFs).

- BALAI à double rangée de connexions de rechange du module de mémoire (DIMM)
- Panne de contrôleur de FlexFlash
- Panne semi-conductrice d'entraînement (disque transistorisé)
- Panne de confiance du module de plate-forme (TPM)
- Panne de cache d'incursion
- Panne chaude de l'adaptateur de bus de contrôleur d'incursion (HBA)
- Panne de canalisation verticale PCI
- Panne d'Intel X520 10G d'adaptateur de PCIe
- Modulaire Réseau local-sur la panne de la carte mère (MLOM)

- Module de ventilation RMA
- Panne CPU

Informations générales

Ultra-M est une solution mobile virtualisée préemballée et validée de noyau de paquet qui est conçue afin de simplifier le déploiement de VNFs. OpenStack est le gestionnaire virtualisé d'infrastructure (SCORE) pour Ultra-M et se compose de ces types de noeud :

- Calcul
- Disque de mémoire d'objet - Calcul (OSD - Calcul)
- Contrôleur
- Plate-forme d'OpenStack - Directeur (OSPD)

L'architecture de haut niveau d'Ultra-M et les composants impliqués sont dépeints dans cette image :

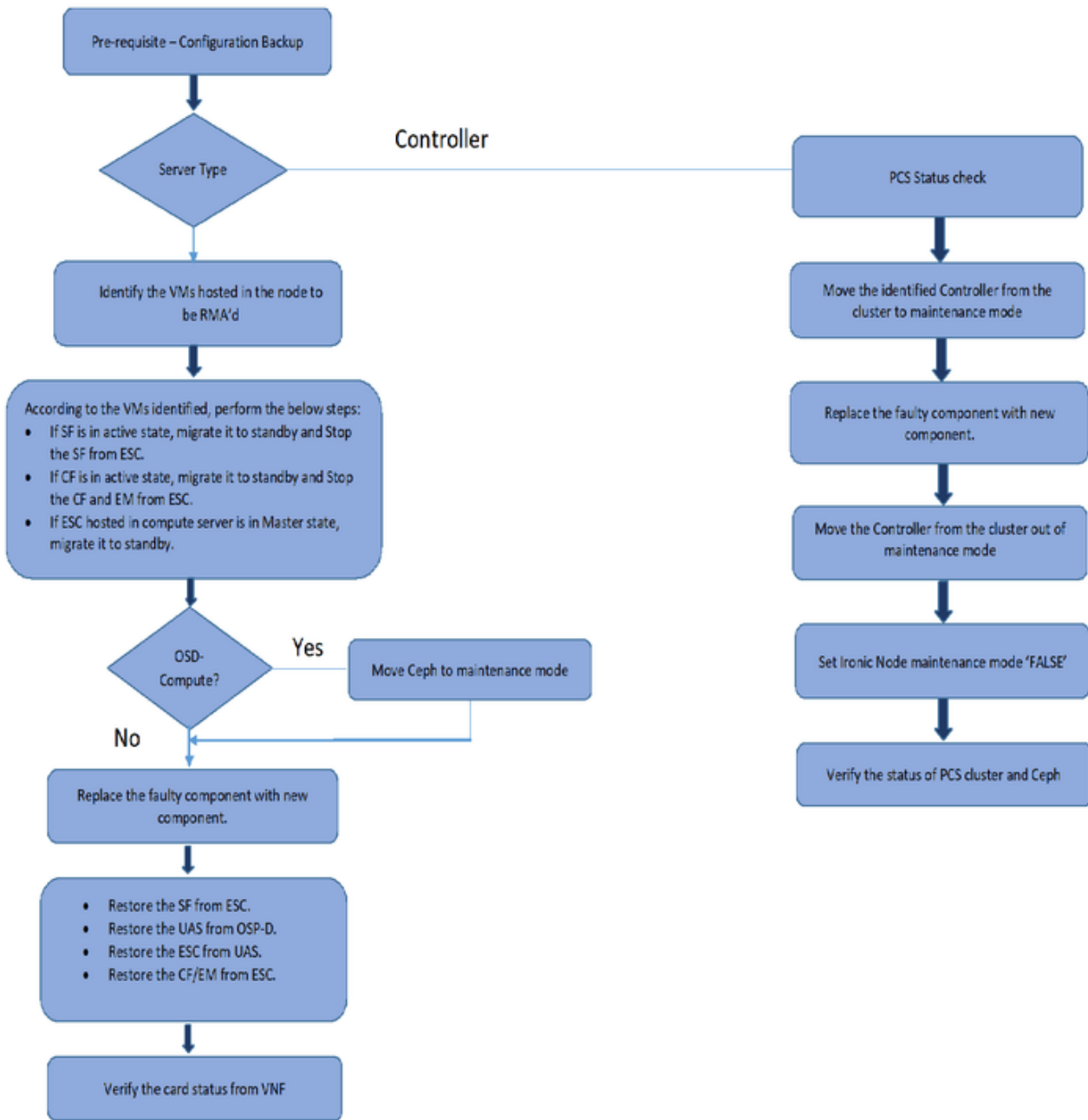
Ce document est destiné pour le personnel de Cisco qui sont au courant de la plate-forme de Cisco Ultra-M et elle détaille l'étape nécessaire à effectuer au niveau d'OpenStack et de StarOS VNF au moment du remplacement composant dans le serveur.

Remarque: Ultra la release M 5.1.x est considérée comme afin de définir les procédures dans ce document.

Abréviations

VNF	Fonction réseau virtuelle
CF	Fonction de contrôle
SF	Fonction de service
ESC	Contrôleur élastique de service
BALAI	Méthode de procédure
OSD	Disques de mémoire d'objet
HDD	Lecteur de disque dur
Disque transistorisé	Lecteur semi-conducteur
SCORE	Gestionnaire virtuel d'infrastructure
VM	Virtual machine
EM	Gestionnaire d'éléments
UAS	Services d'automatisation ultra
UUID	Universellement identifiant unique

Processus du balai



Conditions préalables

Sauvegarde

Avant que vous remplaciez un composant défectueux, il est important de vérifier l'état actuel de votre environnement de plate-forme d'OpenStack de Red Hat. Il vous est recommandé de contrôler l'état actuel afin d'éviter des complications quand le processus de remplacement est allumé. Il peut être réalisé par cet écoulement de remplacement.

En cas de reprise, Cisco recommande de prendre une sauvegarde de la base de données OSPD avec l'utilisation de ces étapes :

```
[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
```

Ce processus s'assure qu'un noeud peut être remplacé sans affecter la Disponibilité de tous les exemples. En outre, c'est sauvegarde recommandée la configuration de StarOS particulièrement si le noeud compute/OSD-compute à remplacer héberge le virtual machine de fonction de contrôle (CF) (VM).

Remarque: Si le serveur est le noeud de contrôleur, poursuivez au "" de section, autrement continuez la section suivante.

Composant RMA - Noeud Compute/OSD-Compute

Identifiez les VMs hébergées dans le noeud Compute/OSD-Compute

Identifiez les VMs qui sont hébergées sur le serveur. Il peut y avoir deux possibilités :

- Le serveur contient seulement la VM de la fonction de service (SF) :

```
[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 |
```

- Serveur contient fonction de contrôle (CF) /Elastic services contrôleur (ESC)/gestionnaire d'éléments) (d'EM/ultra combinaison des services d'automatisation (UAS) des VMs :

```
[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
| f5bd7b9c-476a-4679-83e5-303f0aae9309 | VNF2-UAS-uas-0
|
pod1-compute-8.localdomain |
pod1-compute-8.localdomain |
```

Remarque: Dans le résultat présenté ici, la première colonne correspond universellement à l'identifiant unique (UUID), la deuxième colonne est le nom VM et la troisième colonne est l'adresse Internet où la VM est présente. Les paramètres de cette sortie seront utilisés dans les parties suivantes.

Gracieux mettez hors tension

Le noeud de calcul de l'affaire 1. héberge seulement la VM SF

Migrez la carte SF vers l'état de réserve

- Ouvrez une session au StarOS VNF et identifiez la carte qui correspond à la VM SF. Utilisez l'UUID de la VM SF identifiée de la section « identifiez les VMs hébergées dans le noeud Compute/OSD-Compute », et identifiez la carte qui correspond à l'UUID :

```
[local]VNF2# show card hardware
Tuesday might 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
```

- Vérifiez l'état de la carte :

```
[local]VNF2# show card table
Tuesday might 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      -
```

- Si la carte est dans l'état active, déplacez la carte à l'état de réserve :

```
[local]VNF2# show card table
Tuesday might 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      -
```

VM de l'arrêt SF d'ESC

- Ouvrez une session au noeud d'ESC qui correspond au VNF et vérifiez l'état de la VM SF :

```
[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>
    VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229
    VM_ALIVE_STATE
    VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
    VM_ALIVE_STATE</state>
<snip>
```

- Arrêtez la VM SF avec l'utilisation de son nom VM. (Le nom VM remarquable de la section « identifient les VMs hébergées dans le noeud Compute/OSD-Compute ») :

```
[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>
    VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229
    VM_ALIVE_STATE
    VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
    VM_ALIVE_STATE</state>
<snip>
```

- Une fois qu'il est arrêté, la VM doit entrer dans l'état d'INTERRUPTION :

```
[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>
    VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229
    VM_ALIVE_STATE
    VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
    VM_ALIVE_STATE
    VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
    VM_SHUTOFF_STATE</state>
```

Hôtes CF/ESC/EM/UAS de noeud de l'affaire 2. Compute/OSD-Compute

Migrez la carte CF vers l'état de réserve

- Ouvrez une session au StarOS VNF et identifiez la carte qui correspond à la VM de CF. Utilisez l'UUID de la VM de CF identifiée de la section « identifient les VMs hébergées dans le noeud », et trouvent la carte qui correspond à l'UUID :

```
[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>
    VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229
    VM_ALIVE_STATE
```

```
VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
VM_ALIVE_STATE
VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
  VM_SHUTOFF_STATE</state>
```

- Vérifiez l'état de la carte :

```
[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>
  VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229
  VM_ALIVE_STATE
  VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
  VM_ALIVE_STATE
  VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
  VM_SHUTOFF_STATE</state>
```

- Si la carte est dans l'état active, déplacez la carte à l'état de réserve :

```
[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>
  VNF2-DEPLOYM_c1_0_df4be88d-b4bf-4456-945a-3812653ee229
  VM_ALIVE_STATE
  VNF2-DEPLOYM_c3_0_3e0db133-c13b-4e3d-ac14-
  VM_ALIVE_STATE
  VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
  VM_SHUTOFF_STATE</state>
```

CF d'arrêt et VM d'EM d'ESC

- Ouvrez une session au noeud d'ESC qui correspond au VNF et vérifiez l'état des VMs :

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

<snip>

- Arrêtez les CF et la VM d'EM un par un avec l'utilisation de son nom VM. (Le nom VM remarquable de la section « identifiant les VMs hébergées dans le noeud Compute/OSD-Compute ») :

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

- Après qu'il arrête, les VMs doivent entrer dans l'état d'INTERRUPTION :

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

Migrez l'ESC vers le mode standby

- Ouvrez une session à l'ESC hébergé dans le noeud et vérifiez s'il est dans l'état principal. Si oui, commutez l'ESC au mode standby :

```
[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>
```

```

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

```

Remarque: Si le composant défectueux doit être remplacé sur le noeud d'OSD-calcul, mettez le **Ceph** dans la maintenance sur le serveur avant que vous poursuiviez le remplacement composant.

```

[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance

```

```

[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout

```

```

[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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

```

Remplacez le composant défectueux du noeud Compute/OSD-Compute

Mettez hors tension le serveur spécifié. Les étapes afin de remplacer un composant défectueux sur le serveur UCS C240 M4 peuvent être référées de :

[Remplacer les composants de serveur](#)

Restaurez les VMs

Le noeud de calcul de l'affaire 1. héberge seulement la VM SF

Reprise VM SF d'ESC

- La VM SF serait dans l'état d'erreur dans la liste de nova :

```

[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance

```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

- Récupérez la VM SF de l'ESC :

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

- Surveillez yangesc.log :

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
```

```
controller-1,tb3-ultram-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
```

- Assurez-vous que la carte SF monte comme standby SF dans le VNF

CF, ESC, EM et UAS d'hôtes de noeud de l'affaire 2. Compute/OSD-Compute

Reprise de VM UAS

- Vérifiez l'état de la VM UAS dans la liste de nova et supprimez-le :

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
  election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

- Afin de récupérer la VM d'autovnf-uas, exécutez le script d'uas-contrôle afin de vérifier l'état. Il doit signaler une erreur. Puis passage de nouveau avec **--réparez** l'option afin de recréer la VM des disparus UAS :

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
  election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

- Procédure de connexion aux autovnf-uas. Attendez quelques minutes et UAS doit retourner au bon état :

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
  election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

Remarque: Si uas-check.py --la difficulté échoue, vous pourriez devoir copier ces fichier et passage de nouveau.

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
  election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

Reprise de VM d'ESC

- Vérifiez l'état de la VM d'ESC de la liste de nova et supprimez-le :

```
[admin@osd-compute-0 ~]$ sudo ceph osd set norebalance
```

```
set norebalance
```

```
[admin@osd-compute-0 ~]$ sudo ceph osd set noout
set noout
```

```
[admin@osd-compute-0 ~]$ 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 {tb3-ultram-pod1-controller-0=11.118.0.40:6789/0,tb3-ultram-pod1-
controller-1=11.118.0.41:6789/0,tb3-ultram-pod1-controller-2=11.118.0.42:6789/0}
election epoch 58, quorum 0,1,2 tb3-ultram-pod1-controller-0,tb3-ultram-pod1-
controller-1,tb3-ultram-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
```

- **D'AutoVNF-UAS**, trouvez la transaction de déploiement d'ESC et dans le log pour la transaction trouvez la ligne de **commande boot_vm.py** afin de créer l'exemple d'ESC :

```
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	TX TYPE	DEPLOYMENT ID	
TIMESTAMP	STATUS		
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b	vnf-deployment	VNF2-DEPLOYMENT	2017-11-29T02:01:27.750692-00:00
73d9c540-d4a8-11e7-bb72-fa163ef8df2b	vnfm-deployment	VNF2-ESC	2017-11-29T01:56:02.133663-00:00

```
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
```

Sauvegardez la ligne **boot_vm.py** à un fichier script de shell (esc.sh) et mettez à jour toutes les

lignes de ***** de nom d'utilisateur et de ***** de mot de passe avec les informations correctes (typiquement core/<PASSWORD>). Vous devez enlever – l'option d'encrypt_key aussi bien. Pour des user_pass et des user_confid_pass, vous devez utiliser le format – nom d'utilisateur : mot de passe (exemple - admin : <PASSWORD>).

- Trouvez l'URL bootvm.py de running-config et de wget le bootvm.pyfile à la VM d'autovnf-uas. Dans ce cas, 10.1.2.3 est l'IP de la VM d'Automatique-service informatique :

```
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                                TX TYPE                                DEPLOYMENT ID
TIMESTAMP                            STATUS
-----
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b vnf-deployment                          VNF2-DEPLOYMENT  2017-11-
29T02:01:27.750692-00:00 deployment-success
73d9c540-d4a8-11e7-bb72-fa163ef8df2b vnf-deployment                          VNF2-ESC         2017-11-
29T01:56:02.133663-00:00 deployment-success

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

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                                TX TYPE                                DEPLOYMENT ID
TIMESTAMP                            STATUS
-----
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b vnf-deployment                          VNF2-DEPLOYMENT  2017-11-
29T02:01:27.750692-00:00 deployment-success
73d9c540-d4a8-11e7-bb72-fa163ef8df2b vnf-deployment                          VNF2-ESC         2017-11-
29T01:56:02.133663-00:00 deployment-success
```

```
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_confd_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
```

- Créez un fichier de /tmp/esc_params.cfg :

```
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	TX TYPE	DEPLOYMENT ID	TIMESTAMP	STATUS
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b	vnf-deployment	VNF2-DEPLOYMENT	2017-11-29T02:01:27.750692-00:00	deployment-success
73d9c540-d4a8-11e7-bb72-fa163ef8df2b	vnfm-deployment	VNF2-ESC	2017-11-29T01:56:02.133663-00:00	deployment-success

```
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_confd_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

- Exécutez le script de shell afin de déployer l'ESC du noeud UAS :

```
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                                TX TYPE                                DEPLOYMENT ID
TIMESTAMP                            STATUS
-----
-----
35eefc4a-d4a9-11e7-bb72-fal63ef8df2b  vnf-deployment                        VNF2-DEPLOYMENT  2017-11-
29T02:01:27.750692-00:00  deployment-success
73d9c540-d4a8-11e7-bb72-fal63ef8df2b  vnf-deployment                        VNF2-ESC         2017-11-
29T01:56:02.133663-00:00  deployment-success
```

```
VNF2-uas-uas-0#show logs 73d9c540-d4a8-11e7-bb72-fal63ef8df2b | 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-fal63ef8df2b</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
```

- Ouvrez une session au nouvel ESC et vérifiez l'état de sauvegarde :

```
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                                TX TYPE                                DEPLOYMENT ID
TIMESTAMP                            STATUS
-----
-----
35eefc4a-d4a9-11e7-bb72-fal63ef8df2b  vnf-deployment                        VNF2-DEPLOYMENT  2017-11-
29T02:01:27.750692-00:00  deployment-success
73d9c540-d4a8-11e7-bb72-fal63ef8df2b  vnf-deployment                        VNF2-ESC         2017-11-
29T01:56:02.133663-00:00  deployment-success
```

```
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
```

Récupérez les VMs de CF et d'EM de l'ESC

- Vérifiez l'état des VMs de CF et d'EM de la liste de nova. Ils doivent être dans l'état d'ERREUR :

```
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	TX TYPE	DEPLOYMENT ID	STATUS
35eefc4a-d4a9-11e7-bb72-fa163ef8df2b	vnf-deployment	VNF2-DEPLOYMENT	2017-11-29T02:01:27.750692-00:00 deployment-success
73d9c540-d4a8-11e7-bb72-fa163ef8df2b	vnfm-deployment	VNF2-ESC	2017-11-29T01:56:02.133663-00:00 deployment-success

```
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
```

- Ouvrez une session au maître d'ESC, exécutez la reprise-VM-action pour chaque EM et VM de CF affectés. Soyez patient. L'ESC programmerait la reprise-action et il ne pourrait pas se produire pendant quelques minutes. Surveillez yangesc.log :

```
sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO <VM-Name>
```

```
[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]
```

- Ouvrez une session au nouvel EM et vérifiez l'état d'EM est :

```
sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO <VM-Name>
```

```
[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]
```

- Connectez-vous dans le StarOS VNF et vérifiez que la carte CF est dans l'état de réserve

Panne de reprise d'ESC de traitement

Dans les cas où l'ESC ne commence pas la VM due à un état inattendu, Cisco recommande comment exécuter un basculement d'ESC en redémarrant l'ESC principal. Le basculement d'ESC prendrait environ une minute. Exécutez le script « **health.sh** » sur le nouvel ESC principal afin de vérifier si l'état est en hausse. ESC principal pour commencer la VM et pour réparer l'état VM. Cette tâche de reprise prendrait à 5 minutes pour se terminer.

Vous pouvez surveiller `/var/log/esc/yangesc.log` et `/var/log/esc/escmanager.log`. Si vous ne voyez pas que la VM obtient récupéré après 5-7 minutes, l'utilisateur devrait aller faire la reprise manuelle de la VM affectée.

Automatique-déployez la mise à jour de la configuration

- De la VM d'AutoDeploy, éditez l'**autodeploy.cfg** et remplacez le vieux serveur de calcul par le neuf. Alors le chargement remplace dans le `confd_cli`. **Cette** étape est exigée pour la mise hors fonction de déploiement réussi plus tard :

```
sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO <VM-Name>
```

```
[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.ZpRCG1ieuW
<?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]
```

- Redémarrez l'**uas-confd** et les services **autodeploy** après la modification de configuration :

```
sudo /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO <VM-Name>
```

```
[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.ZpRCG1ieuW
<?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]
```

Composant RMA - Noeud de contrôleur

Pré vérification

- D'OSPD, la procédure de connexion au contrôleur et vérifient des PCs est dans un bon état – chacun des trois contrôleurs exposition en ligne et de Galera chacun des trois contrôleurs comme maître.

Remarque: Une batterie saine exige 2 contrôleurs actifs ainsi vérifiez que les deux contrôleurs qui restent sont en ligne et en activité.

```
[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-pod1-controller-0 (stonith:fence_ipmilan): Started pod1-controller-0
my-ipmilan-for-pod1-controller-1 (stonith:fence_ipmilan): Started pod1-controller-0
my-ipmilan-for-pod1-controller-2 (stonith:fence_ipmilan): Started pod1-controller-0
```

Daemon Status:

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

Déplacez la batterie de contrôleur au mode maintenance

- Utilisez les **PCs groupent** sur le contrôleur qui est mis à jour en état d'alerte :

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

- Vérifiez l'**état de PCs** de nouveau et assurez-vous que les PCs groupent arrêté sur ce noeud :

```
[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-pod1-controller-0 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-pod1-controller-1 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-pod1-controller-2 (stonith:fence_ipmilan): Started pod1-controller-2
```

Daemon Status:

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

En outre, l'état de PCs sur les 2 autres contrôleurs devrait afficher le noeud comme standby.

Remplacez le composant défectueux du noeud de contrôleur

Mettez hors tension le serveur spécifié. Les étapes afin de remplacer un composant défectueux sur le serveur UCS C240 M4 peuvent être référées de :

[Remplacer les composants de serveur](#)

Mettez sous tension le serveur

- Mettez sous tension le serveur et vérifiez le serveur monte :

```
[stack@tb5-ospd ~]$ source stackrc
[stack@tb5-ospd ~]$ nova list |grep pod1-controller-0
| 1ca946b8-52e5-4add-b94c-4d4b8a15a975 | pod1-controller-0 | ACTIVE | - | Running
| ctlplane=192.200.0.112 |
```

- Ouvrez une session au contrôleur affecté, retirez le mode standby avec l'utilisation d'**unstandby**. Vérifiez que le contrôleur est livré en ligne avec la batterie et Galera affiche chacun des trois contrôleurs comme maître. Ceci pourrait prendre quelques minutes :

```
[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.e17_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-pod1-controller-0 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-pod1-controller-1 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-pod1-controller-2 (stonith:fence_ipmilan): Started pod1-controller-2
```

Daemon Status:

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

- Vous pouvez vérifier certains des services de moniteur tels que le **ceph** qu'ils sont dans un état sain :

```
[heat-admin@pod1-controller-0 ~]$ sudo ceph -s
cluster eb2bb192-b1c9-11e6-9205-525400330666
health HEALTH_OK
monmap e1: 3 mons at {pod1-controller-0=11.118.0.10:6789/0,pod1-controller-
```

```
1=11.118.0.11:6789/0,pod1-controller-2=11.118.0.12:6789/0}
  election epoch 70, quorum 0,1,2 pod1-controller-0,pod1-controller-1,pod1-controller-2
osdmap e218: 12 osds: 12 up, 12 in
  flags sortbitwise,require_jewel_osds
pgmap v2080888: 704 pgs, 6 pools, 714 GB data, 237 kobjects
  2142 GB used, 11251 GB / 13393 GB avail
    704 active+clean
client io 11797 kB/s wr, 0 op/s rd, 57 op/s wr
```