

# Sostituzione dei componenti guasti sul server UCS C240 M4 - vEPC

## Sommario

[Introduzione](#)

[Premesse](#)

[Abbreviazioni](#)

[Flusso di lavoro del piano di mobilità](#)

[Prerequisiti](#)

[Backup](#)

[Component RMA - Compute/OSD-Compute Node](#)

[Identificare le VM ospitate nel nodo di calcolo/calcolo OSD](#)

[Spegnimento regolare](#)

[Caso 1. Il nodo di calcolo ospita solo VM SF](#)

[Caso 2. Compute/OSD-Compute Node Hosts CF/ESC/EM/UAS](#)

[Sostituire il componente difettoso dal nodo di calcolo/calcolo OSD-A](#)

[Ripristino delle VM](#)

[Caso 1. Il nodo di calcolo ospita solo VM SF](#)

[Caso 2. Compute/OSD-Compute Node ospita CF, ESC, EM e UAS](#)

[Gestisci errore di ripristino ESC](#)

[Aggiornamento configurazione distribuzione automatica](#)

[RMA component - Controller Node](#)

[Verifica preliminare](#)

---

[Sposta cluster controller in modalità manutenzione](#)

[Sostituire il componente difettoso dal nodo del controller](#)

[Accendere il server](#)

## Introduzione

Questo documento descrive i passaggi necessari per sostituire i componenti guasti menzionati qui in un server UCS (Unified Computing System) in una configurazione Ultra-M che ospita funzioni di rete virtuale StarOS (VNF).

- Modulo di memoria DIMM (Dual In-line Memory Module) sostitutivo
- Errore del controller FlexFlash
- Errore unità a stato solido (SSD)
- Errore del TPM (Trusted Platform Module)
- Errore cache RAID
- Errore del controller RAID/HBA (Hot Bus Adapter)
- Errore riser PCI
- Scheda PCIe Intel X520 10G guasto
- Errore MLOM (Modular LAN-on Motherboard)

- Vassoio ventola RMA
- Errore CPU

## Premesse

Ultra-M è una soluzione di base di pacchetti mobili preconfezionata e convalidata, progettata per semplificare l'installazione di VNF. OpenStack è Virtualized Infrastructure Manager (VIM) per Ultra-M ed è costituito dai seguenti tipi di nodi:

- Calcola
- Disco Object Storage - Compute (OSD - Compute)
- Controller
- Piattaforma OpenStack - Director (OSPD)

L'architettura di alto livello di Ultra-M e i componenti coinvolti sono illustrati in questa immagine:

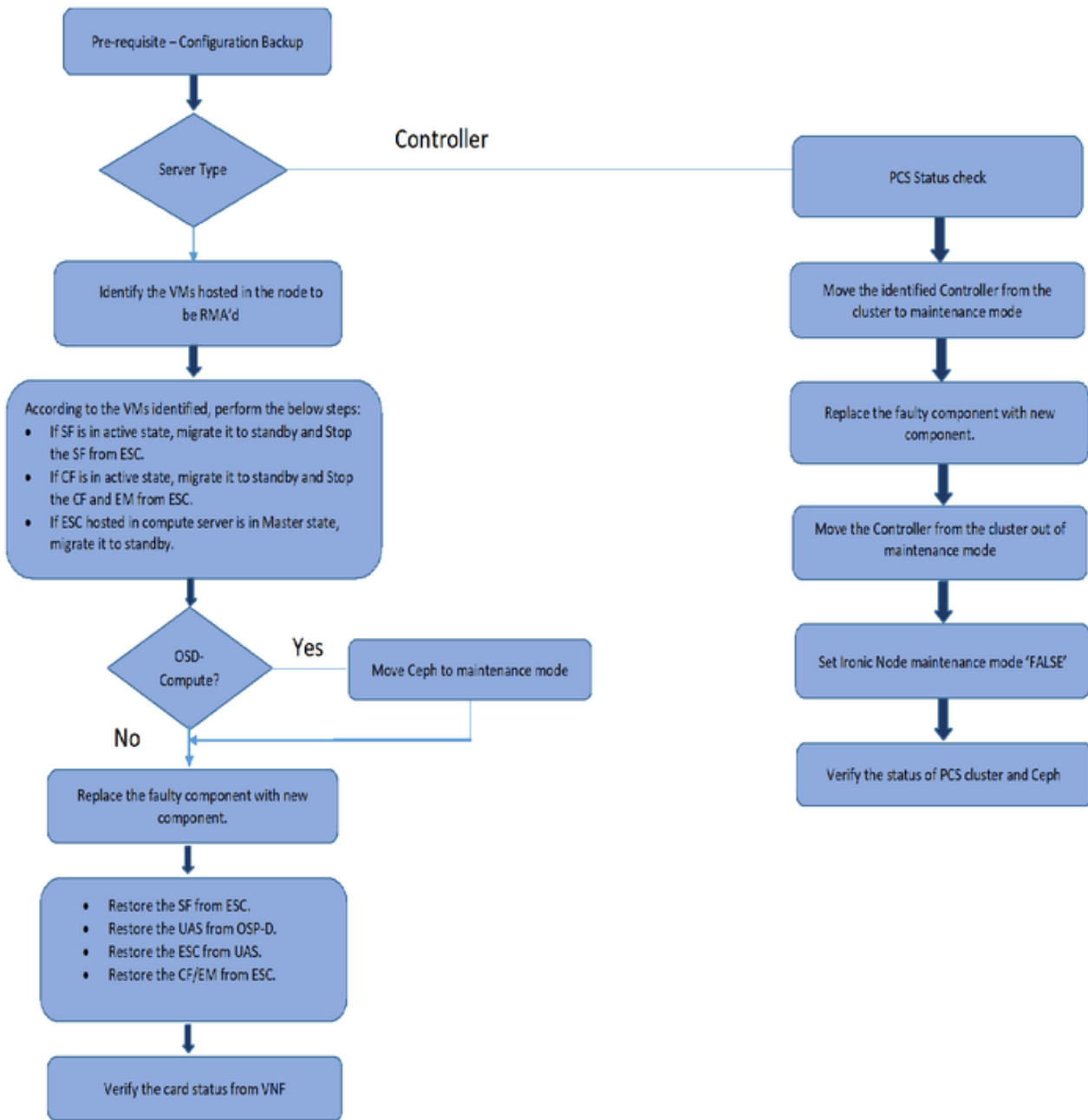
Questo documento è destinato al personale Cisco che ha familiarità con la piattaforma Cisco Ultra-M e descrive i passaggi richiesti per essere eseguiti a livello OpenStack e StarOS VNF al momento della sostituzione dei componenti nel server.

**Nota:** Per definire le procedure descritte in questo documento, viene presa in considerazione la release di Ultra M 5.1.x.

## Abbreviazioni

VNF	Funzione di rete virtuale
CF	Funzione di controllo
SF	Funzione di servizio
ESC	Elastic Service Controller
MOP	Metodo
OSD	Dischi Object Storage
HDD	Unità hard disk
SSD	Unità a stato solido
VIM	Virtual Infrastructure Manager
VM	Macchina virtuale
EM	Gestione elementi
UAS	Ultra Automation Services
UUID	Identificatore univoco universale

## Flusso di lavoro del piano di mobilità



## Prerequisiti

### Backup

Prima di sostituire un componente difettoso, è importante verificare lo stato corrente dell'ambiente della piattaforma Red Hat OpenStack. Si consiglia di controllare lo stato corrente per evitare complicazioni quando il processo di sostituzione è attivo. Questo flusso di sostituzione consente di ottenere il risultato desiderato.

In caso di ripristino, Cisco consiglia di eseguire un backup del database OSPD attenendosi alla seguente procedura:

```
[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
```

Questo processo assicura che un nodo possa essere sostituito senza influire sulla disponibilità di alcuna istanza. Inoltre, si consiglia di eseguire il backup della configurazione StarOS soprattutto se il nodo di calcolo/OSD da sostituire ospita la macchina virtuale (VM) della funzione di controllo (CF).

**Nota:** Se Server è il nodo Controller, passare alla sezione "", altrimenti passare alla sezione successiva.

## Component RMA - Compute/OSD-Compute Node

### Identificare le VM ospitate nel nodo di calcolo/calcolo OSD

Identificare le VM ospitate nel server. Esistono due possibilità:

- Il server contiene solo la VM Service Function (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 |
```

- Il server contiene una combinazione di funzioni di controllo (CF)/Elastic Services Controller (ESC)/Element Manager (EM)/Ultra Automation Services (UAS) di macchine virtuali:

```
[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 |
```

**Nota:** nell'output mostrato di seguito, la prima colonna corrisponde all'UUID (Universally Unique Identifier), la seconda colonna è il nome della macchina virtuale e la terza colonna è il nome host in cui è presente la macchina virtuale. I parametri di questo output verranno utilizzati nelle sezioni successive.

## Spegnimento regolare

## Caso 1. Il nodo di calcolo ospita solo VM SF

### Esegui migrazione della scheda SF allo stato di standby

- Accedere a StarOS VNF e identificare la scheda corrispondente alla VM SF. Utilizzare l'UUID della VM SF identificato nella sezione "Identificazione delle VM ospitate nel nodo di calcolo OSD-Compute" e identificare la scheda corrispondente all'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
```

- Controllare lo stato della scheda:

```
[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     -
```

- Se la scheda è nello stato attivo, passare allo stato standby:

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

### Arresta VM SF da ESC

- Accedere al nodo ESC corrispondente al file VNF e verificare lo stato della 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>
```

- Arrestare la VM SF utilizzando il relativo nome della VM. (Nome della VM indicato nella sezione "Identificazione delle VM ospitate nel nodo di calcolo/OSD-calcolo"):

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

- Una volta arrestata, la VM deve entrare nello stato **SHUTOFF**:

```
[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>
```

## Caso 2. Compute/OSD-Compute Node Hosts CF/ESC/EM/UAS

### Migrazione della scheda CF allo stato Standby

- Accedere alla VNF di StarOS e identificare la scheda corrispondente alla VM CF. Utilizzare l'UUID della VM CF identificato nella sezione "Identificare le VM ospitate nel nodo" e individuare la scheda corrispondente all'UUID:

```
[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 (qvpc-di-large)
UUID/Serial Number  : F9C0763A-4A4F-4BBD-AF51-BC7545774BE2
<snip>
```

- Controllare lo stato della scheda:

```
[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	-

- Se la scheda è nello stato attivo, passare allo stato standby:

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

## Arrestare CF e VM EM da ESC

- Accedere al nodo ESC corrispondente al file VNF e verificare lo stato delle macchine virtuali:

```
[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>
```

- Arrestare la VM CF e EM uno per uno utilizzando il relativo nome della VM. (Nome della VM indicato nella sezione "Identificazione delle VM ospitate nel nodo di calcolo/OSD-calcolo"):

```
[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
```

- Dopo l'arresto, le VM devono entrare nello stato SHUTOFF:

```
[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>
```

<snip>

## Esegui migrazione ESC in modalità standby

- Accedere all'ESC ospitato nel nodo e verificare se si trova nello stato master. In caso affermativo, passare all'ESC in modalità standby:

```
[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!
```

**Nota:** Se il componente difettoso deve essere sostituito su un nodo OSD-Compute, attivare la manutenzione sul server prima di procedere con la sostituzione del componente.

```
[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
```

## Sostituire il componente difettoso dal nodo di calcolo/calcolo OSD-A



Spegnere il server specificato. Per sostituire un componente guasto su un server UCS C240 M4, è possibile seguire la procedura descritta di seguito:

## [Sostituzione dei componenti server](#)

### Ripristino delle VM

#### Caso 1. Il nodo di calcolo ospita solo VM SF

Ripristino VM SF da ESC

- La VM SF si troverebbe in stato di errore nell'elenco nova:

```
[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 |
```

- Ripristinare la VM SF dalla ESC:

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

```
Recovery VM Action
/opt/cisco/esc/confd/bin/netconf-console --port=830 --host=127.0.0.1 --user=admin --
privKeyFile=/root/.ssh/confd_id_dsa --privKeyType=dsa --rpc=/tmp/esc_nc_cli.ZpRCGiieuW
<?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>
```

- Monitorare il file **yangesc.log**:

```
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].
```

- Assicurarsi che la scheda SF venga inserita come SF in standby nel VNF

#### Caso 2. Compute/OSD-Compute Node ospita CF, ESC, EM e UAS

##### Ripristino di VM UAS

- Verificare lo stato della VM UAS nell'elenco delle macchine virtuali nuove ed eliminarla:

```
[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.
```

- Per ripristinare la macchina virtuale autovnf-uas, eseguire lo script **uas-check** per controllare lo stato. Deve segnalare un errore. Eseguire nuovamente con l'opzione **—fix** per ricreare la VM UAS mancante:

```
[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'
```

- Accedere a autovnf-uas. Attendere qualche minuto e UAS deve tornare allo stato buono:

```
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
```

**Nota:** Se **uas-check.py —fix** non riesce, potrebbe essere necessario copiare il file ed eseguirlo nuovamente.

```
[stack@director ~]$ mkdir -p /opt/cisco/usp/apps/auto-it/common/uas-deploy/
[stack@director ~]$ cp /opt/cisco/usp/uas-installer/common/uas-deploy/userdata-uas.txt
/opt/cisco/usp/apps/auto-it/common/uas-deploy/
```

## Ripristino di VM ESC

- Controllare lo stato della VM ESC dall'elenco delle macchine virtuali ed eliminarlo:

```
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.
```

- Da **AutoVNF-UAS**, trovare la transazione di distribuzione ESC e nel log della transazione trovare la riga di comando **boot\_vm.py** per creare l'istanza 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	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
```

Salvare la riga **boot\_vm.py** in un file script della shell (**esc.sh**) e aggiornare tutte le righe relative al nome utente **\*\*\*\*** e alla password **\*\*\*\*\*** con le informazioni corrette (in genere **core/<PASSWORD>**). È necessario rimuovere anche l'opzione **-encrypt\_key**. Per **user\_pass** e **user\_confid\_pass**, è necessario utilizzare il formato **- nomeutente: password** (esempio **admin:<PASSWORD>**).

- Trovare l'URL per **bootvm.py** da running-config e trasferire il file **bootvm.py** nella VM **autovnf-uas**. In questo caso, **10.1.2.3** è l'IP della VM **Auto-IT**:

```
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]
```

- Creare un file `/tmp/esc_params.cfg`:

```
root@VNF2-uas-uas-0:~# echo "openstack.endpoint=publicURL" > /tmp/esc_params.cfg
```

- Eseguire lo script della shell per distribuire ESC dal nodo UAS:

```
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
```

- Accedere al nuovo ESC e verificare lo stato Backup:

```
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
```

## Ripristino di VM CF ed EM da ESC

- Controllare lo stato delle VM CF e EM dall'elenco delle macchine virtuali. Devono essere nello stato ERROR:

```
[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
```

- Accedere al master ESC, eseguire l'operazione di ripristino-vm per ogni server EM e VM CF interessati. Siate pazienti. La procedura ESC pianificherebbe l'azione di ripristino e potrebbe non essere eseguita per alcuni minuti. Monitorare il file yangesc.log:

```
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]
```

- Accedere a un nuovo EM e verificare che lo stato EM sia attivo:

```
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
```

- Accedere al VNF di StarOS e verificare che la scheda CF sia in stato di standby

## Gestisci errore di ripristino ESC

Nei casi in cui ESC non riesca ad avviare la macchina virtuale a causa di uno stato imprevisto, Cisco consiglia come eseguire il passaggio a ESC riavviando la macchina virtuale master. Il passaggio al CES richiede circa un minuto. Eseguire lo script "**health.sh**" sul nuovo Master ESC per verificare se lo stato è attivo. Master ESC per avviare la macchina virtuale e correggere lo stato della macchina virtuale. L'attività di ripristino può richiedere fino a 5 minuti.

È possibile monitorare `/var/log/esc/yangesc.log` e `/var/log/esc/escmanager.log`. Se non si rileva che la VM viene ripristinata dopo 5-7 minuti, l'utente deve eseguire il ripristino manuale delle VM interessate.

## Aggiornamento configurazione distribuzione automatica

- Da AutoDeploy VM, modificare il file **autodeploy.cfg** e sostituire il vecchio server di elaborazione con quello nuovo. Caricare quindi il comando `replace` in **confd\_cli**. Questo passaggio è necessario per la corretta disattivazione della distribuzione in seguito:

```
root@auto-deploy-iso-2007-uas-0:/home/ubuntu# 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#config
Entering configuration mode terminal
auto-deploy-iso-2007-uas-0(config)#load replace autodeploy.cfg
Loading.      14.63 KiB parsed in 0.42 sec (34.16 KiB/sec)

auto-deploy-iso-2007-uas-0(config)#commit
Commit complete.
auto-deploy-iso-2007-uas-0(config)#end
```

- Riavviare i servizi **uas-config** e **autodeploy** dopo la modifica della configurazione:

```
root@auto-deploy-iso-2007-uas-0:~# service uas-confd restart
uas-confd stop/waiting
uas-confd start/running, process 14078

root@auto-deploy-iso-2007-uas-0:~# service uas-confd status
uas-confd start/running, process 14078

root@auto-deploy-iso-2007-uas-0:~# service autodeploy restart
autodeploy stop/waiting
autodeploy start/running, process 14017
root@auto-deploy-iso-2007-uas-0:~# service autodeploy status
autodeploy start/running, process 14017
```

## RMA component - Controller Node

### Verifica preliminare

- Da OSPD, effettuare il login al controller e verificare che lo stato dei pc sia buono. Tutti e tre i controller Online e Galera presentano tutti e tre i controller come Master.

**Nota:** Un cluster integro richiede 2 controller attivi, quindi verificare che i due controller rimanenti siano online e attivi.

```
[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
```

## Sposta cluster controller in modalità manutenzione

- Utilizzare il **cluster pcs** sul controller aggiornato in standby:

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

- Controllare di nuovo lo stato del **pcs** e verificare che il cluster del pcs sia stato arrestato in questo nodo:

```
[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
```

Inoltre, lo stato del pcs sugli altri 2 controller deve indicare il nodo come standby.

## Sostituire il componente difettoso dal nodo del controller

Spegnere il server specificato. È possibile fare riferimento ai passaggi per sostituire un componente guasto su un server UCS C240 M4 da:

[Sostituzione dei componenti server](#)

## Accendere il server

- Accendere il server e verificarne l'accensione:

```
[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 |
```

- Accedere al controller interessato e rimuovere la modalità standby utilizzando **unstandby**. Verificare che il controller sia in linea con il cluster e che in Galera tutti e tre i controller siano visualizzati come Master. L'operazione potrebbe richiedere alcuni minuti:

```
[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

- È possibile verificare che alcuni servizi di monitoraggio, ad esempio **ceph**, siano in buono stato:

```
[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
```