

Sostituzione PCRf del server controller UCS C240 M4

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Backup](#)

[Controllo preliminare dello stato](#)

[Disabilita restrizione nel cluster di controller](#)

[Installare il nuovo nodo del controller](#)

[Sostituzione dei nodi di controller nel cloud](#)

[Preparazione rimozione nodo controller non riuscito](#)

[Preparazione aggiunta nuovo nodo controller](#)

[Intervento manuale](#)

[Verifica servizi cloud nel controller](#)

[Finalizzazione dei router dell'agente L3](#)

[Finalizza servizi di elaborazione](#)

[Riavviare la restrizione sui nodi del controller](#)

Introduzione

In questo documento vengono descritti i passaggi necessari per sostituire un server controller difettoso in una configurazione Ultra-M che ospita funzioni di rete virtuale (VNF, Virtual Network Functions) di CPS.

Prerequisiti

Backup

In caso di ripristino, Cisco consiglia di eseguire un backup del database OSPD (DB) 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
```

Controllo preliminare dello stato

È importante controllare lo stato attuale dell'ambiente e dei servizi OpenStack e assicurarsi che sia integro prima di procedere con la procedura di sostituzione. Consente di evitare complicazioni al momento della sostituzione del controller.

Passaggio 1. Controllare lo stato di OpenStack e l'elenco dei nodi:

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

Passaggio 2. Controllare lo stato di Pacemaker sui controller.

Accedere a uno dei controller attivi e verificare lo stato di pacemaker. Tutti i servizi devono essere in esecuzione sui controller disponibili e arrestati sul controller guasto.

```
[stack@pod1-controller-0 ~]# pcs status

<snip>
Online: [ pod1-controller-0 pod1-controller-1 ]
OFFLINE: [ pod1-controller-2 ]
Full list of resources:
ip-11.120.0.109 (ocf::heartbeat:IPaddr2): Started pod1-controller-0
ip-172.25.22.109 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
ip-192.200.0.107 (ocf::heartbeat:IPaddr2): Started pod1-controller-0

Clone Set: haproxy-clone [haproxy]
Started: [ pod1-controller-0 pod1-controller-1 ]
Stopped: [ pod1-controller-2 ]

Master/Slave Set: galera-master [galera]
Masters: [ pod1-controller-0 pod1-controller-1 ]
Stopped: [ pod1-controller-2 ]
ip-11.120.0.110 (ocf::heartbeat:IPaddr2): Started pod1-controller-0
ip-11.119.0.110 (ocf::heartbeat:IPaddr2): Started pod1-controller-1

Clone Set: rabbitmq-clone [rabbitmq]
Started: [ pod1-controller-0 pod1-controller-1 ]
Stopped: [ pod1-controller-2 ]

Master/Slave Set: redis-master [redis]
Masters: [ pod1-controller-0 ]
Slaves: [ pod1-controller-1 ]
Stopped: [ pod1-controller-2 ]

ip-11.118.0.104 (ocf::heartbeat:IPaddr2): Started pod1-controller-1
openstack-cinder-volume (systemd:openstack-cinder-volume): Started pod1-controller-0

my-ipmilan-for-controller-6 (stonith:fence_ipmilan): Started pod1-controller-1
my-ipmilan-for-controller-4 (stonith:fence_ipmilan): Started pod1-controller-0
my-ipmilan-for-controller-7 (stonith:fence_ipmilan): Started pod1-controller-0

Failed Actions:
Daemon Status:

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

Nell'esempio, Controller-2 è offline. Esso sarà pertanto sostituito. Controller-0 e Controller-1 sono operativi e eseguono i servizi cluster.

Passaggio 3. Controllare lo stato di MariaDB nei controller attivi.

| | | | | | |
|---|--------|--------|---------|--------------------------------|-----------------------|
| neutron-dhcp-agent.service | loaded | active | running | OpenStack | Neutron DHCP Agent |
| neutron-openvswitch-agent.service Agent | loaded | active | running | OpenStack | Neutron Open vSwitch |
| neutron-ovs-cleanup.service Cleanup Utility | loaded | active | exited | OpenStack | Neutron Open vSwitch |
| neutron-server.service | loaded | active | running | OpenStack | Neutron Server |
| openstack-aodh-evaluator.service service | loaded | active | running | OpenStack | Alarm evaluator |
| openstack-aodh-listener.service service | loaded | active | running | OpenStack | Alarm listener |
| openstack-aodh-notifier.service service | loaded | active | running | OpenStack | Alarm notifier |
| openstack-ceilometer-central.service agent | loaded | active | running | OpenStack | ceilometer central |
| openstack-ceilometer-collector.service service | loaded | active | running | OpenStack | ceilometer collection |
| openstack-ceilometer-notification.service notification agent | loaded | active | running | OpenStack | ceilometer |
| openstack-glance-api.service named Glance) API server | loaded | active | running | OpenStack | Image Service (code- |
| openstack-glance-registry.service named Glance) Registry server | loaded | active | running | OpenStack | Image Service (code- |
| openstack-heat-api-cfn.service API Service | loaded | active | running | Openstack | Heat CFN-compatible |
| openstack-heat-api.service | loaded | active | running | OpenStack | Heat API Service |
| openstack-heat-engine.service | loaded | active | running | Openstack | Heat Engine Service |
| openstack-ironic-api.service | loaded | active | running | OpenStack | Ironic API service |
| openstack-ironic-conductor.service service | loaded | active | running | OpenStack | Ironic Conductor |
| openstack-ironic-inspector-dnsmasq.service Ironic Inspector | loaded | active | running | PXE boot dnsmasq service for | |
| openstack-ironic-inspector.service for OpenStack Ironic | loaded | active | running | Hardware introspection service | |
| openstack-mistral-api.service | loaded | active | running | Mistral | API Server |
| openstack-mistral-engine.service | loaded | active | running | Mistral | Engine Server |
| openstack-mistral-executor.service | loaded | active | running | Mistral | Executor Server |
| openstack-nova-api.service | loaded | active | running | OpenStack | Nova API Server |
| openstack-nova-cert.service | loaded | active | running | OpenStack | Nova Cert Server |
| openstack-nova-compute.service | loaded | active | running | OpenStack | Nova Compute Server |
| openstack-nova-conductor.service | loaded | active | running | OpenStack | Nova Conductor Server |
| openstack-nova-scheduler.service | loaded | active | running | OpenStack | Nova Scheduler Server |
| openstack-swift-account-reaper.service (swift) - Account Reaper | loaded | active | running | OpenStack | Object Storage |
| openstack-swift-account.service (swift) - Account Server | loaded | active | running | OpenStack | Object Storage |
| openstack-swift-container-updater.service (swift) - Container Updater | loaded | active | running | OpenStack | Object Storage |
| openstack-swift-container.service (swift) - Container Server | loaded | active | running | OpenStack | Object Storage |
| openstack-swift-object-updater.service (swift) - Object Updater | loaded | active | running | OpenStack | Object Storage |
| openstack-swift-object.service (swift) - Object Server | loaded | active | running | OpenStack | Object Storage |
| openstack-swift-proxy.service (swift) - Proxy Server | loaded | active | running | OpenStack | Object Storage |
| openstack-zaqar.service Service (code-named Zaqar) Server | loaded | active | running | OpenStack | Message Queuing |
| openstack-zaqar@1.service Service (code-named Zaqar) Server Instance 1 | loaded | active | running | OpenStack | Message Queuing |
| openvswitch.service | loaded | active | exited | Open | vSwitch |

LOAD = Reflects whether the unit definition was properly loaded.

ACTIVE = The high-level unit activation state, i.e. generalization of SUB.

SUB = The low-level unit activation state, values depend on unit type.

37 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.

Disabilita restrizione nel cluster di controller

```
[root@pod1-controller-0 ~]# sudo pcs property set stonith-enabled=false  
[root@pod1-controller-0 ~]# pcs property show
```

```
Cluster Properties:  
cluster-infrastructure: corosync  
cluster-name: tripleo_cluster  
dc-version: 1.1.15-11.e17_3.4-e174ec8  
have-watchdog: false  
last-lrm-refresh: 1510809585  
maintenance-mode: false  
redis_REPL_INFO: pod1-controller-0  
stonith-enabled: false
```

```
Node Attributes:  
pod1-controller-0: rmq-node-attr-last-known-rabbitmq=rabbit@pod1-controller-0  
pod1-controller-1: rmq-node-attr-last-known-rabbitmq=rabbit@pod1-controller-1  
pod1-controller-2: rmq-node-attr-last-known-rabbitmq=rabbit@pod1-controller-2
```

Installare il nuovo nodo del controller

Passaggio 1. Per l'installazione di un nuovo server UCS C240 M4 e le operazioni di configurazione iniziali, consultare la [Guida all'installazione e al servizio del server Cisco UCS C240 M4](#)

Passaggio 2. Accedere al server utilizzando l'indirizzo IP CIMC.

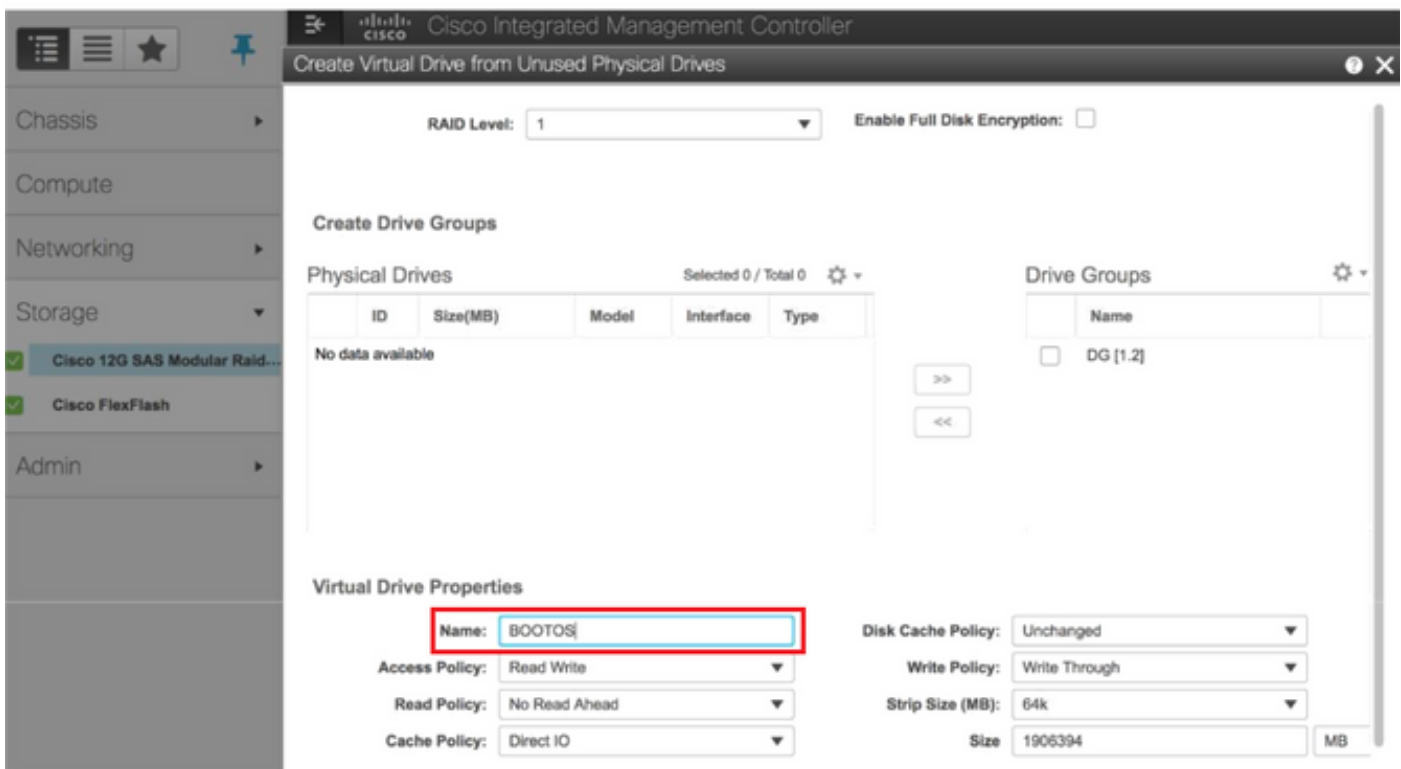
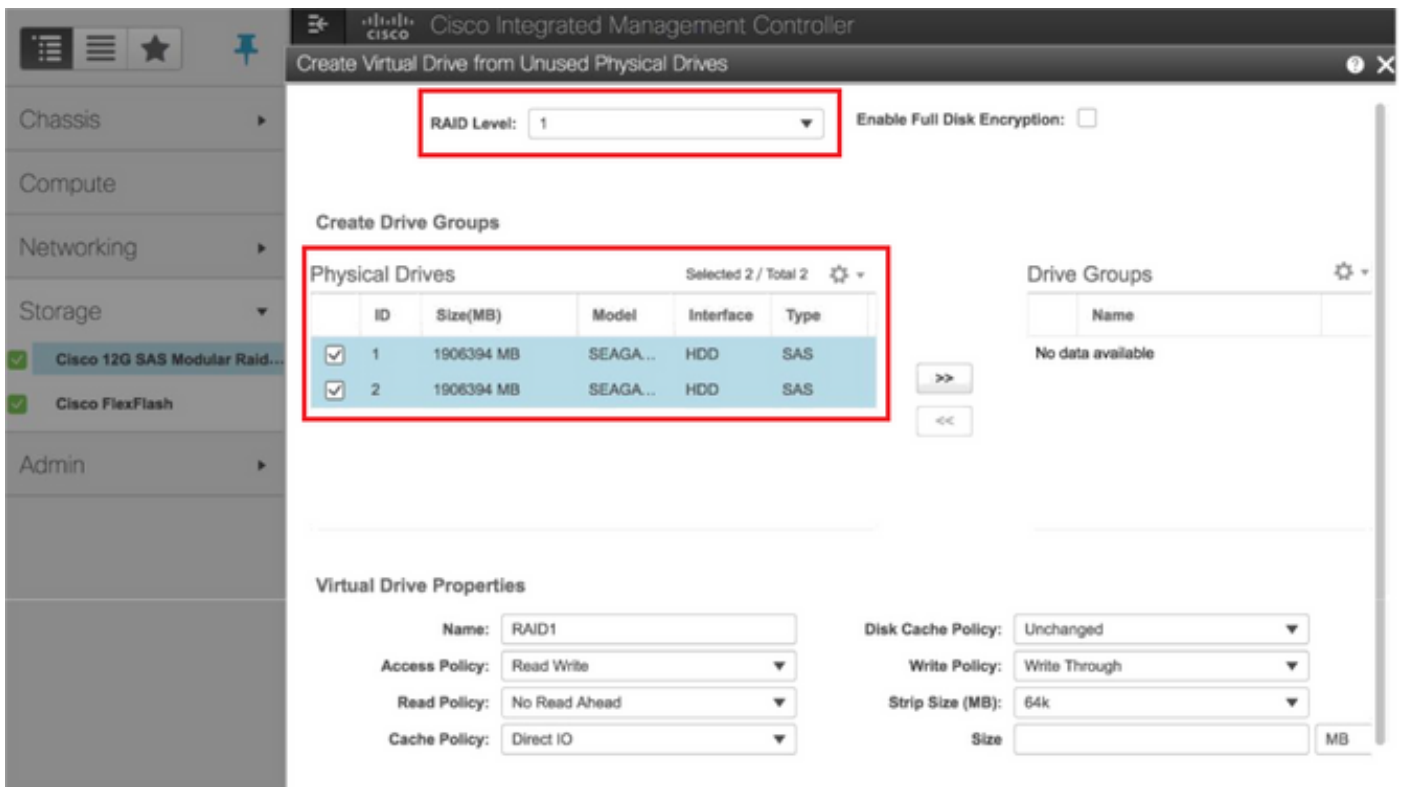
Passaggio 3. Eseguire l'aggiornamento del BIOS se il firmware non corrisponde alla versione consigliata utilizzata in precedenza. Le fasi per l'aggiornamento del BIOS sono riportate di seguito:

[Guida all'aggiornamento del BIOS dei server con montaggio in rack Cisco UCS serie C](#)

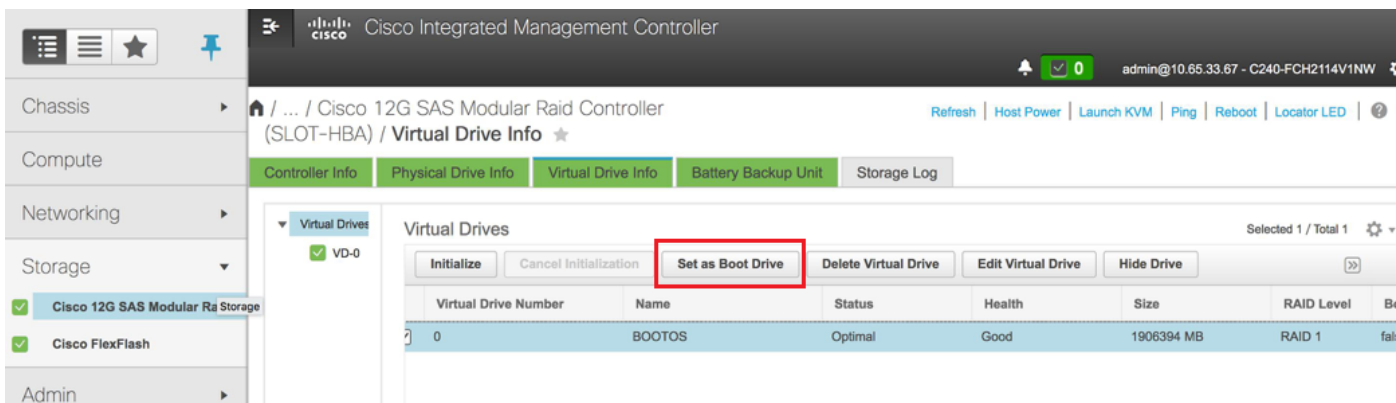
Passaggio 4. Verificare lo stato delle unità fisiche. Deve essere **Non configurato correttamente**. Selezionare **Storage > Cisco 12G SAS Modular Raid Controller (SLOT-HBA) > Physical Drive Info (Informazioni sull'unità fisica)**.

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

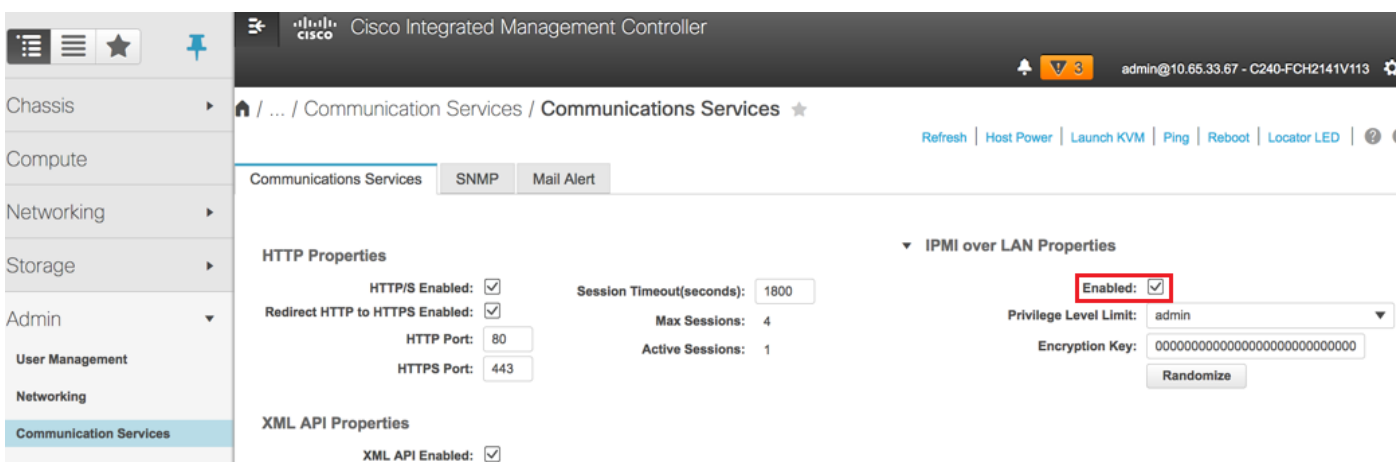
Passaggio 5. Per creare un'unità virtuale dalle unità fisiche con RAID di livello 1: Passare a **Storage > Cisco 12G SAS Modular Raid Controller (SLOT-HBA) > Informazioni controller > Crea unità virtuale da unità fisiche inutilizzate**, come mostrato nell'immagine.



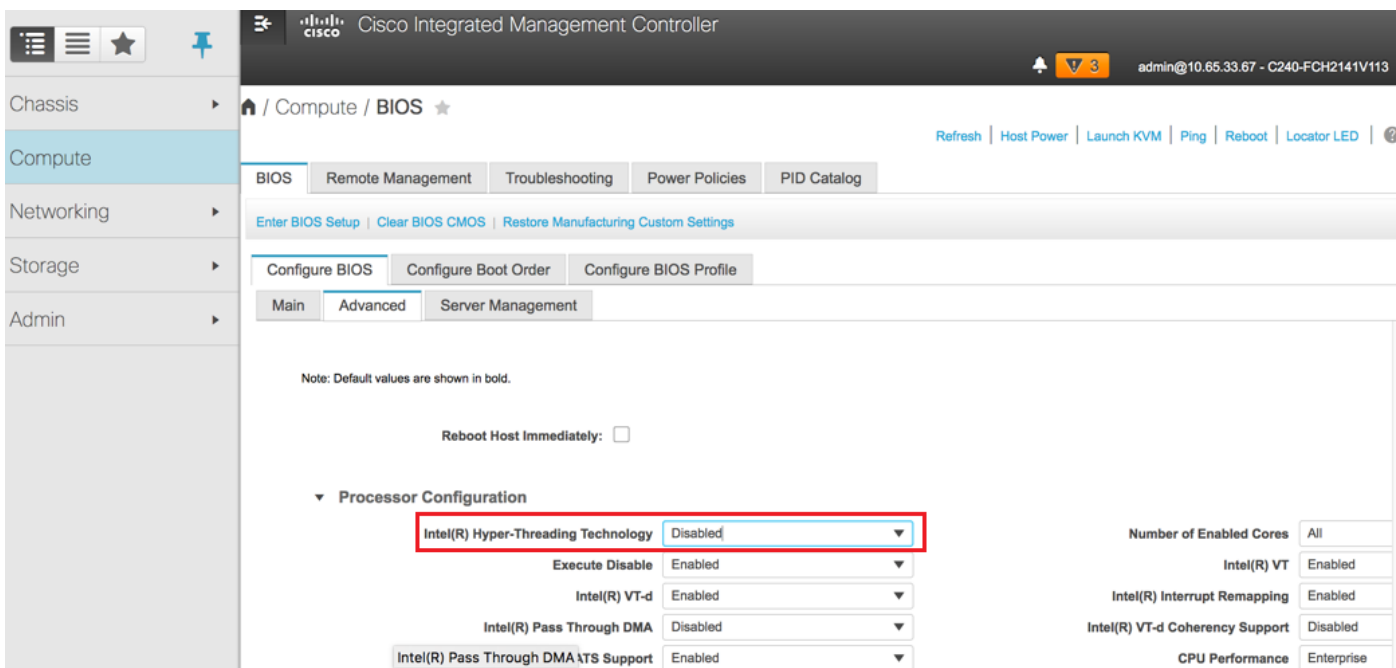
- Selezionare il disco virtuale e configurare **Set as Boot Drive**:



Passaggio 6. Per abilitare IPMI su LAN, selezionare **Admin > Communication Services > Communication Services**.



Passaggio 7. Per disabilitare l'HyperThreading, selezionare **Compute > BIOS > Configure BIOS > Advanced > Processor Configuration**, come mostrato nell'immagine.



Nota: L'immagine è mostrata qui e le procedure di configurazione menzionate in questa sezione fanno riferimento alla versione firmware 3.0(3e). Se si lavora su altre versioni, potrebbero verificarsi lievi variazioni.

Sostituzione dei nodi di controller nel cloud

In questa sezione vengono illustrati i passaggi necessari per sostituire il controller difettoso con quello nuovo nel cloud. A tale scopo, verrà riutilizzato lo script **deploy.sh** utilizzato per attivare lo stack. Al momento della distribuzione, nella fase `ControllerNodesPostDeployment`, l'aggiornamento non verrà eseguito a causa di alcune limitazioni nei moduli Puppet. È necessario un intervento manuale prima di riavviare lo script di distribuzione.

Preparazione rimozione nodo controller non riuscito

Passaggio 1. Identificare l'indice del controller guasto. L'indice è il suffisso numerico sul nome del controller nell'output dell'elenco dei server OpenStack. In questo esempio, l'indice è 2:

```
[stack@director ~]$ nova list | grep controller
| 5813a47e-af27-4fb9-8560-75decd3347b4 | pod1-controller-0 | ACTIVE | - | Running
| ctlplane=192.200.0.152 |
| 457f023f-d077-45c9-bbea-dd32017d9708 | pod1-controller-1 | ACTIVE | - | Running
| ctlplane=192.200.0.154 |
| d13bb207-473a-4e42-a1e7-05316935ed65 | pod1-controller-2 | ACTIVE | - | Running
| ctlplane=192.200.0.151 |
```

Passaggio 2. Creare un file Yaml **~templates/remove-controller.yaml** che definisca il nodo da eliminare. Utilizzare l'indice individuato nel passaggio precedente per la voce nell'elenco delle risorse.

```
[stack@director ~]$ cat templates/remove-controller.yaml
```

```
parameters:
  ControllerRemovalPolicies:
    [{'resource_list': ['2']}]
```

```
parameter_defaults:
  CorosyncSettleTries: 5
```

Passaggio 3. Creare una copia dello script di distribuzione utilizzato per installare l'overcloud e inserire una riga per includere il file **remove-controller.yaml** creato in precedenza.

```
[stack@director ~]$ cp deploy.sh deploy-removeController.sh
[stack@director ~]$ cat deploy-removeController.sh
time openstack overcloud deploy --templates \
-r ~/custom-templates/custom-roles.yaml \
-e /home/stack/templates/remove-controller.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml \
-e ~/custom-templates/network.yaml \
-e ~/custom-templates/ceph.yaml \
-e ~/custom-templates/compute.yaml \
-e ~/custom-templates/layout-removeController.yaml \
-e ~/custom-templates/rabbitmq.yaml \
--stack pod1 \
--debug \
--log-file overcloudDeploy_$(date +%m_%d_%y__%H_%M_%S).log \
--neutron-flat-networks phys_pcie1_0,phys_pcie1_1,phys_pcie4_0,phys_pcie4_1 \
--neutron-network-vlan-ranges datacentre:101:200 \
```



```
--neutron-disable-tunneling \  
--verbose --timeout 180
```

Passaggio 4. Identificare l'ID del controller da sostituire, utilizzando i comandi menzionati in questo passaggio e passare alla modalità di manutenzione.

```
[stack@director ~]$ nova list | grep controller
```

```
| 5813a47e-af27-4fb9-8560-75decd3347b4 | pod1-controller-0 | ACTIVE | - | Running  
| ctlplane=192.200.0.152 |  
  
| 457f023f-d077-45c9-bbea-dd32017d9708 | pod1-controller-1 | ACTIVE | - | Running  
| ctlplane=192.200.0.154 |  
  
| d13bb207-473a-4e42-a1e7-05316935ed65 | pod1-controller-2 | ACTIVE | - | Running  
| ctlplane=192.200.0.151 |
```

```
[stack@director ~]$ openstack baremetal node list | grep d13bb207-473a-4e42-a1e7-05316935ed65
```

```
| e7c32170-c7d1-4023-b356-e98564a9b85b | None | d13bb207-473a-4e42-a1e7-05316935ed65 | power  
off | active | False |
```

```
[stack@b10-ospd ~]$ openstack baremetal node maintenance set e7c32170-c7d1-4023-b356-e98564a9b85b
```

```
[stack@director ~]$ openstack baremetal node list | grep True
```

```
| e7c32170-c7d1-4023-b356-e98564a9b85b | None | d13bb207-473a-4e42-a1e7-05316935ed65 | power  
off | active | True |
```

Passaggio 5. Per garantire che il database venga eseguito al momento della procedura di sostituzione, rimuovere Galera dal controllo pacemaker ed eseguire questo comando su uno dei controller attivi.

```
[root@pod1-controller-0 ~]# sudo pcs resource unmanage galera
```

```
[root@pod1-controller-0 ~]# sudo pcs status
```

```
Cluster name: tripleo_cluster
```

```
Stack: corosync
```

```
Current DC: pod1-controller-0 (version 1.1.15-11.e17_3.4-e174ec8) - partition with quorum
```

```
Last updated: Thu Nov 16 16:51:18 2017
```

```
Last change: Thu Nov 16 16:51:12 2017
```

```
by root via crm_resource on pod1-controller-0
```

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

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

```
OFFLINE: [ pod1-controller-2 ]
```

```
Full list of resources:
```

```
ip-11.120.0.109 (ocf::heartbeat:IPaddr2): Started pod1-controller-0  
ip-172.25.22.109 (ocf::heartbeat:IPaddr2): Started pod1-controller-1  
ip-192.200.0.107 (ocf::heartbeat:IPaddr2): Started pod1-controller-0
```

```
Clone Set: haproxy-clone [haproxy]
```

```
Started: [ pod1-controller-0 pod1-controller-1 ]
```

```
Stopped: [ pod1-controller-2 ]
```

```
Master/Slave Set: galera-master [galera] (unmanaged)
```

```
galera (ocf::heartbeat:galera): Master pod1-controller-0 (unmanaged)
```

```
galera (ocf::heartbeat:galera): Master pod1-controller-1 (unmanaged)
```

```
Stopped: [ pod1-controller-2 ]
ip-11.120.0.110      (ocf::heartbeat:IPAddr2):      Started pod1-controller-0
ip-11.119.0.110     (ocf::heartbeat:IPAddr2):      Started pod1-controller-1
```

<snip>

Preparazione aggiunta nuovo nodo controller

Passaggio 1. Creare un file **controllerRMA.json** contenente solo i dettagli del nuovo controller. Verificare che il numero di indice sul nuovo controller non sia stato utilizzato in precedenza. In genere, passare al numero di controller successivo più alto.

Esempio: Il precedente più alto era Controller-2, quindi creare Controller-3.

Nota: Prestare attenzione al formato json.

```
[stack@director ~]$ cat controllerRMA.json
{
  "nodes": [
    {
      "mac": [
        <MAC_ADDRESS>
      ],
      "capabilities": "node:controller-3,boot_option:local",
      "cpu": "24",
      "memory": "256000",
      "disk": "3000",
      "arch": "x86_64",
      "pm_type": "pxe_ipmitool",
      "pm_user": "admin",
      "pm_password": "<PASSWORD>",
      "pm_addr": "<CIMC_IP>"
    }
  ]
}
```

Passaggio 2. Importare il nuovo nodo utilizzando il file json creato nel passaggio precedente.

```
[stack@director ~]$ openstack baremetal import --json controllerRMA.json

Started Mistral Workflow. Execution ID: 67989c8b-1225-48fe-ba52-3a45f366e7a0

Successfully registered node UUID 048ccb59-89df-4f40-82f5-3d90d37ac7dd

Started Mistral Workflow. Execution ID: c6711b5f-fa97-4c86-8de5-b6bc7013b398

Successfully set all nodes to available.

[stack@director ~]$ openstack baremetal node list | grep available
```

```
| 048ccb59-89df-4f40-82f5-3d90d37ac7dd | None | None | power
off | available | False
```

Passaggio 3. Impostare il nodo sullo stato di gestione.

```
[stack@director ~]$ openstack baremetal node manage 048ccb59-89df-4f40-82f5-3d90d37ac7dd
[stack@director ~]$ openstack baremetal node list | grep off
```

```
| 048ccb59-89df-4f40-82f5-3d90d37ac7dd | None | None | power off | manageable | False |
```

Passaggio 4. Eseguire l'introspezione.

```
[stack@director ~]$ openstack overcloud node introspect 048ccb59-89df-4f40-82f5-3d90d37ac7dd --  
provide  
Started Mistral Workflow. Execution ID: f73fb275-c90e-45cc-952b-bfc25b9b5727  
Waiting for introspection to finish...  
Successfully introspected all nodes.  
Introspection completed.  
Started Mistral Workflow. Execution ID: a892b456-eb15-4c06-b37e-5bc3f6c37c65  
Successfully set all nodes to available
```

```
[stack@director ~]$ openstack baremetal node list | grep available  
| 048ccb59-89df-4f40-82f5-3d90d37ac7dd | None | None | power  
off | available | False |
```

Passaggio 5. Contrassegnare il nodo disponibile con le nuove proprietà del controller. Assicurarsi di utilizzare l'ID controller indicato per il nuovo controller, come utilizzato nel file **controllerRMA.json**.

```
[stack@director ~]$ openstack baremetal node set --property capabilities='node:controller-  
3,profile:control,boot_option:local' 048ccb59-89df-4f40-82f5-3d90d37ac7dd
```

Passaggio 6. Nello script di distribuzione è presente un modello personalizzato denominato **layout.yaml** che, tra le altre cose, specifica quali indirizzi IP vengono assegnati ai controller per le varie interfacce. Su un nuovo stack, sono stati definiti 3 indirizzi per Controller-0, Controller-1 e Controller-2. Quando si aggiunge un nuovo controller, accertarsi di aggiungere un indirizzo IP successivo in sequenza per ciascuna subnet.

```
ControllerIPs:  
internal_api:  
- 11.120.0.10  
- 11.120.0.11  
- 11.120.0.12  
- 11.120.0.13  
tenant:  
- 11.117.0.10  
- 11.117.0.11  
- 11.117.0.12  
- 11.117.0.13  
storage:  
- 11.118.0.10  
- 11.118.0.11  
- 11.118.0.12  
- 11.118.0.13  
storage_mgmt:  
- 11.119.0.10  
- 11.119.0.11  
- 11.119.0.12  
- 11.119.0.13
```

Passaggio 7. Eseguire ora il file **deploy-removecontroller.sh** creato in precedenza, per rimuovere il vecchio nodo e aggiungerne uno nuovo.

Nota: Si prevede che questo passaggio non riesca in ControllerNodesDeployment_Step1. A questo punto, è necessario un intervento manuale.

```
[stack@b10-ospd ~]$ ./deploy-addController.sh
START with options: [u'overcloud', u'deploy', u'--templates', u'-r', u'/home/stack/custom-templates/custom-roles.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml', u'-e', u'/home/stack/custom-templates/network.yaml', u'-e', u'/home/stack/custom-templates/ceph.yaml', u'-e', u'/home/stack/custom-templates/compute.yaml', u'-e', u'/home/stack/custom-templates/layout-removeController.yaml', u'-e', u'/home/stack/custom-templates/rabbitmq.yaml', u'--stack', u'newtonoc', u'--debug', u'--log-file', u'overcloudDeploy_11_15_17__07_46_35.log', u'--neutron-flat-networks', u'phys_pcie1_0,phys_pcie1_1,phys_pcie4_0,phys_pcie4_1', u'--neutron-network-vlan-ranges', u'datacentre:101:200', u'--neutron-disable-tunneling', u'--verbose', u'--timeout', u'180']
:
DeploymentError: Heat Stack update failed
END return value: 1

real    42m1.525s
user    0m3.043s
sys     0m0.614s
```

L'avanzamento/lo stato della distribuzione può essere monitorato con questi comandi:

```
[stack@director~]$ openstack stack list --nested | grep -iv complete

+-----+-----+-----+-----+-----+-----+
| ID                                     | Stack                                     | Stack Status | Creation |
Name                                     | Parent                                   |              |         |
Time          | Updated Time          |              |         |
+-----+-----+-----+-----+-----+-----+
| c1e338f2-877e-4817-93b4-9a3f0c0b3d37 | pod1-AllNodesDeploySteps-5psegydpwxij- | UPDATE_FAILED |         |
ComputeDeployment_Step1-swnuzjixac43    |                                         |              |         |
2017-10-08T14:06:07Z | 2017-11-16T18:09:43Z | e90f00ef-2499-4ec3-90b4-d7def6e97c47 |         |
| 1db4fef4-45d3-4125-bd96-2cc3297a69ff | pod1-AllNodesDeploySteps-5psegydpwxij- | UPDATE_FAILED |         |
ControllerDeployment_Step1-             |                                         |              |         |
hmn3hpruubcn                            |                                         |              |         |
| UPDATE_FAILED | 2017-10-08T14:03:05Z | 2017-11-16T18:12:12Z | e90f00ef-2499-4ec3-90b4- |
d7def6e97c47 |                                         |              |         |
| e90f00ef-2499-4ec3-90b4-d7def6e97c47 | pod1-AllNodesDeploySteps-5psegydpwxij- | UPDATE_FAILED | 2017-10-08T13:59:25Z | 2017-11- |
16T18:09:25Z | 6c4b604a-55a4-4a19-9141-28c844816c0d |         |         |
| 6c4b604a-55a4-4a19-9141-28c844816c0d | pod1                                     | UPDATE_FAILED | 2017-10- |
08T12:37:11Z | 2017-11-16T17:35:35Z | None         |         |
+-----+-----+-----+-----+-----+-----+
```

Intervento manuale

Passaggio 1. Sul server OSP-D, eseguire il comando OpenStack server list per elencare i controller disponibili. Il controller appena aggiunto dovrebbe essere visualizzato nell'elenco.

```
[stack@director ~]$ openstack server list | grep controller
| 3e6c3db8-ba24-48d9-b0e8-1e8a2eb8b5ff | pod1-controller-3 | ACTIVE | ctlplane=192.200.0.103 |
overcloud-full |
| 457f023f-d077-45c9-bbea-dd32017d9708 | pod1-controller-1 | ACTIVE | ctlplane=192.200.0.154 |
overcloud-full |
| 5813a47e-af27-4fb9-8560-75decd3347b4 | pod1-controller-0 | ACTIVE | ctlplane=192.200.0.152 |
overcloud-full |
```

Passaggio 2. Connettersi a uno dei controller attivi (non il controller appena aggiunto) e controllare il file `/etc/corosync/corosync.conf`. Individuare l'elenco di nodi che assegna un `nodeid` a ogni controller. Individuare la voce relativa al nodo non riuscito e annotare il relativo `ID nodo`:

```
[root@pod1-controller-0 ~]# cat /etc/corosync/corosync.conf
totem {
    version: 2
    secauth: off
    cluster_name: tripleo_cluster
    transport: udpu
    token: 10000
}

nodelist {
    node {
        ring0_addr: pod1-controller-0
        nodeid: 5
    }
    node {
        ring0_addr: pod1-controller-1
        nodeid: 7
    }
    node {
        ring0_addr: pod1-controller-2
        nodeid: 8
    }
}
```

Passaggio 3. Accedere a ciascuno dei controller attivi. Rimuovere il nodo in errore e riavviare il servizio. In questo caso, rimuovere `pod1-controller-2`. Non eseguire questa azione sul controller appena aggiunto.

```
[root@pod1-controller-0 ~]# sudo pcs cluster localnode remove pod1-controller-2
pod1-controller-2: successfully removed!
[root@pod1-controller-0 ~]# sudo pcs cluster reload corosync
Corosync reloaded
```

```
[root@pod1-controller-1 ~]# sudo pcs cluster localnode remove pod1-controller-2
pod1-controller-2: successfully removed!
[root@pod1-controller-1 ~]# sudo pcs cluster reload corosync
Corosync reloaded
```

Passaggio 4. Eseguire questo comando da uno dei controller attivi per eliminare il nodo in errore dal cluster.

```
[root@pod1-controller-0 ~]# sudo crm_node -R pod1-controller-2 --force
```

Passaggio 5. Eseguire questo comando da uno dei controller attivi per eliminare il nodo in errore dal cluster **rabbitmq**.

```
[root@pod1-controller-0 ~]# sudo rabbitmqctl forget_cluster_node rabbit@pod1-controller-2  
Removing node 'rabbit@newtonoc-controller-2' from cluster ...
```

Passaggio 6. Eliminare il nodo non riuscito da MongoDB. A tale scopo, è necessario trovare il nodo Mongo attivo. Utilizzare **netstat** per trovare l'indirizzo IP dell'host.

```
[root@pod1-controller-0 ~]# sudo netstat -tulnp | grep 27017  
tcp        0      0 11.120.0.10:27017      0.0.0.0:*                LISTEN  
219577/mongod
```

Passaggio 7. Accedere al nodo e verificare se si tratta del dispositivo master con l'indirizzo IP e il numero di porta specificati nel comando precedente.

```
[heat-admin@pod1-controller-0 ~]$ echo "db.isMaster()" | mongo --host 11.120.0.10:27017  
MongoDB shell version: 2.6.11  
connecting to: 11.120.0.10:27017/test  
{  
  "setName" : "tripleo",  
  "setVersion" : 9,  
  "ismaster" : true,  
  "secondary" : false,  
  "hosts" : [  
    "11.120.0.10:27017",  
    "11.120.0.12:27017",  
    "11.120.0.11:27017"  
  ],  
  "primary" : "11.120.0.10:27017",  
  "me" : "11.120.0.10:27017",  
  "electionId" : ObjectId("5a0d2661218cb0238b582fb1"),  
  "maxBsonObjectSize" : 16777216,  
  "maxMessageSizeBytes" : 48000000,  
  "maxWriteBatchSize" : 1000,  
  "localTime" : ISODate("2017-11-16T18:36:34.473Z"),  
  "maxWireVersion" : 2,  
  "minWireVersion" : 0,  
  "ok" : 1  
}
```

Se il nodo non è il dispositivo master, accedere all'altro controller attivo ed eseguire lo stesso passaggio.

Passaggio 8. Dal dispositivo master, elencare i nodi disponibili utilizzando il comando **rs.status()**. Individuare il nodo precedente/non rispondente e identificare il nome del nodo mongo.

```
[root@pod1-controller-0 ~]# mongo --host 11.120.0.10  
MongoDB shell version: 2.6.11  
connecting to: 11.120.0.10:27017/test  
<snip>
```

```

tripleo:PRIMARY> rs.status()
{
  "set" : "tripleo",
  "date" : ISODate("2017-11-14T13:27:14Z"),
  "myState" : 1,
  "members" : [
    {
      "_id" : 0,
      "name" : "11.120.0.10:27017",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 418347,
      "optime" : Timestamp(1510666033, 1),
      "optimeDate" : ISODate("2017-11-14T13:27:13Z"),
      "electionTime" : Timestamp(1510247693, 1),
      "electionDate" : ISODate("2017-11-09T17:14:53Z"),
      "self" : true
    },
    {
      "_id" : 2,
      "name" : "11.120.0.12:27017",
      "health" : 1,
      "state" : 2,
      "stateStr" : "SECONDARY",
      "uptime" : 418347,
      "optime" : Timestamp(1510666033, 1),
      "optimeDate" : ISODate("2017-11-14T13:27:13Z"),
      "lastHeartbeat" : ISODate("2017-11-14T13:27:13Z"),
      "lastHeartbeatRecv" : ISODate("2017-11-14T13:27:13Z"),
      "pingMs" : 0,
      "syncingTo" : "11.120.0.10:27017"
    },
    {
      "_id" : 3,
      "name" : "11.120.0.11:27017",
      "health" : 0,
      "state" : 8,
      "stateStr" : "(not reachable/healthy)",
      "uptime" : 0,
      "optime" : Timestamp(1510610580, 1),
      "optimeDate" : ISODate("2017-11-13T22:03:00Z"),
      "lastHeartbeat" : ISODate("2017-11-14T13:27:10Z"),
      "lastHeartbeatRecv" : ISODate("2017-11-13T22:03:01Z"),
      "pingMs" : 0,
      "syncingTo" : "11.120.0.10:27017"
    }
  ],
  "ok" : 1
}

```

Passaggio 9. Dal dispositivo master, eliminare il nodo non riuscito utilizzando il comando **rs.remove**. Quando si esegue questo comando vengono visualizzati alcuni errori, ma controllare di nuovo lo stato per verificare che il nodo sia stato rimosso:

```

[root@pod1-controller-0 ~]$ mongo --host 11.120.0.10
<snip>
tripleo:PRIMARY> rs.remove('11.120.0.12:27017')
2017-11-16T18:41:04.999+0000 DBClientCursor::init call() failed
2017-11-16T18:41:05.000+0000 Error: error doing query: failed at src/mongo/shell/query.js:81
2017-11-16T18:41:05.001+0000 trying reconnect to 11.120.0.10:27017 (11.120.0.10) failed
2017-11-16T18:41:05.003+0000 reconnect 11.120.0.10:27017 (11.120.0.10) ok

```

```

tripleo:PRIMARY> rs.status()
{
  "set" : "tripleo",
  "date" : ISODate("2017-11-16T18:44:11Z"),
  "myState" : 1,
  "members" : [
    {
      "_id" : 3,
      "name" : "11.120.0.11:27017",
      "health" : 1,
      "state" : 2,
      "stateStr" : "SECONDARY",
      "uptime" : 187,
      "optime" : Timestamp(1510857848, 3),
      "optimeDate" : ISODate("2017-11-16T18:44:08Z"),
      "lastHeartbeat" : ISODate("2017-11-16T18:44:11Z"),
      "lastHeartbeatRecv" : ISODate("2017-11-16T18:44:09Z"),
      "pingMs" : 0,
      "syncingTo" : "11.120.0.10:27017"
    },
    {
      "_id" : 4,
      "name" : "11.120.0.10:27017",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 89820,
      "optime" : Timestamp(1510857848, 3),
      "optimeDate" : ISODate("2017-11-16T18:44:08Z"),
      "electionTime" : Timestamp(1510811232, 1),
      "electionDate" : ISODate("2017-11-16T05:47:12Z"),
      "self" : true
    }
  ],
  "ok" : 1
}
tripleo:PRIMARY> exit
bye

```

Passaggio 10. Eseguire questo comando per aggiornare l'elenco dei nodi dei controller attivi. Includere il nuovo nodo del controller nell'elenco.

```
[root@pod1-controller-0 ~]# sudo pcs resource update galera wsrep_cluster_address=gcomm://pod1-controller-0,pod1-controller-1,pod1-controller-2
```

Passaggio 11. Copiare questi file da un controller già esistente al nuovo controller:

/etc/sysconfig/clustercheck

/root/.my.cnf

On existing controller:

```
[root@pod1-controller-0 ~]# scp /etc/sysconfig/clustercheck stack@192.200.0.1:/tmp/.
[root@pod1-controller-0 ~]# scp /root/.my.cnf stack@192.200.0.1:/tmp/my.cnf
```

On new controller:

```
[root@pod1-controller-3 ~]# cd /etc/sysconfig
```

```
[root@pod1-controller-3 sysconfig]# scp stack@192.200.0.1:/tmp/clustercheck .
```



```
[root@pod1-controller-3 sysconfig]# cd /root
```

```
[root@pod1-controller-3 ~]# scp stack@192.200.0.1:/tmp/my.cnf .my.cnf
```

Passaggio 12. Eseguire il comando `add` del nodo cluster da uno dei controller già esistenti.

```
[root@pod1-controller-1 ~]# sudo pcs cluster node add pod1-controller-3
```

```
Disabling SBD service...
```

```
pod1-controller-3: sbd disabled
```

```
pod1-controller-0: Corosync updated
```

```
pod1-controller-1: Corosync updated
```

```
Setting up corosync...
```

```
pod1-controller-3: Succeeded
```

```
Synchronizing pcsd certificates on nodes pod1-controller-3...
```

```
pod1-controller-3: Success
```

```
Restarting pcsd on the nodes in order to reload the certificates...
```

```
pod1-controller-3: Success
```

Passaggio 13. Accedere a ciascun controller e visualizzare il file `/etc/corosync/corosync.conf`. Verificare che il nuovo controller sia elencato e che il **nodo assegnato al controller sia il numero successivo nella sequenza non utilizzata in precedenza. Accertarsi che la modifica venga eseguita su tutti e 3 i controller.**

```
[root@pod1-controller-1 ~]# cat /etc/corosync/corosync.conf
```

```
totem {
    version: 2
    secauth: off
    cluster_name: tripleo_cluster
    transport: udpu
    token: 10000
}
nodelist {
    node {
        ring0_addr: pod1-controller-0
        nodeid: 5
    }
    node {
        ring0_addr: pod1-controller-1
        nodeid: 7
    }
    node {
        ring0_addr: pod1-controller-3
        nodeid: 6
    }
}
quorum {
    provider: corosync_votequorum
}
logging {
    to_logfile: yes
    logfile: /var/log/cluster/corosync.log
    to_syslog: yes
}
```

Ad esempio, `/etc/corosync/corosync.conf` dopo la modifica:

```

totem {
version: 2
secauth: off
cluster_name: tripleo_cluster
transport: udpu
token: 10000
}
nodelist {
  node {
    ring0_addr: pod1-controller-0
    nodeid: 5
  }
  node {
    ring0_addr: pod1-controller-1
    nodeid: 7
  }
  node {
    ring0_addr: pod1-controller-3
    nodeid: 9
  }
}
quorum {
  provider: corosync_votequorum
}
logging {
  to_logfile: yes
  logfile: /var/log/cluster/corosync.log
  to_syslog: yes
}

```

Passaggio 14. Riavviare **corosync** sui controller attivi. Non avviare **corosync** sul nuovo controller.

```

[root@pod1-controller-0 ~]# sudo pcs cluster reload corosync
[root@pod1-controller-1 ~]# sudo pcs cluster reload corosync

```

Passaggio 15. Avviare il nuovo nodo di controller da uno dei controller attivi.

```

[root@pod1-controller-1 ~]# sudo pcs cluster start pod1-controller-3

```

Passaggio 16. Riavviare Galera da uno dei controller funzionanti.

```

[root@pod1-controller-1 ~]# sudo pcs cluster start pod1-controller-3

```

```

pod1-controller-0: Starting Cluster...

```

```

[root@pod1-controller-1 ~]# sudo pcs resource cleanup galera
Cleaning up galera:0 on pod1-controller-0, removing fail-count-galera
Cleaning up galera:0 on pod1-controller-1, removing fail-count-galera
Cleaning up galera:0 on pod1-controller-3, removing fail-count-galera
* The configuration prevents the cluster from stopping or starting 'galera-master' (unmanaged)

```

```

Waiting for 3 replies from the CRMD... OK

```

```

[root@pod1-controller-1 ~]#
[root@pod1-controller-1 ~]# sudo pcs resource manage galera

```

Passaggio 17. Il cluster è in modalità manutenzione. Disabilitare la modalità di manutenzione per avviare i servizi.

```

[root@pod1-controller-2 ~]# sudo pcs property set maintenance-mode=false --wait

```

Passaggio 18. Controllare lo stato dei PC per Galera finché tutti e 3 i controller non sono elencati come master in Galera.

Nota: Per le impostazioni di grandi dimensioni, la sincronizzazione dei database può richiedere del tempo.

```
[root@pod1-controller-1 ~]# sudo pcs status | grep galera -A1
```

```
Master/Slave Set: galera-master [galera]
```

```
Masters: [ pod1-controller-0 pod1-controller-1 pod1-controller-3 ]
```

Passaggio 19. Passare alla modalità manutenzione per il cluster.

```
[root@pod1-controller-1~]# sudo pcs property set maintenance-mode=true --wait
```

```
[root@pod1-controller-1 ~]# pcs cluster status
```

```
Cluster Status:
```

```
Stack: corosync
```

```
Current DC: pod1-controller-0 (version 1.1.15-11.e17_3.4-e174ec8) - partition with quorum
```

```
Last updated: Thu Nov 16 19:17:01 2017
```

```
Last change: Thu Nov 16 19:16:48 2017
```

```
by root via cibadmin on pod1-controller-1
```

```
*** Resource management is DISABLED ***
```

```
The cluster will not attempt to start, stop or recover services
```

```
PCSD Status:
```

```
pod1-controller-3: Online
```

```
pod1-controller-0: Online
```

```
pod1-controller-1: Online
```

Passaggio 20. Eseguire nuovamente lo script di distribuzione eseguito in precedenza. Questa volta dovrebbe avere successo.

```
[stack@director ~]$ ./deploy-addController.sh
```

```
START with options: [u'overcloud', u'deploy', u'--templates', u'-r', u'/home/stack/custom-templates/custom-roles.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/puppet-pacemaker.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/storage-environment.yaml', u'-e', u'/usr/share/openstack-tripleo-heat-templates/environments/neutron-sriov.yaml', u'-e', u'/home/stack/custom-templates/network.yaml', u'-e', u'/home/stack/custom-templates/ceph.yaml', u'-e', u'/home/stack/custom-templates/compute.yaml', u'-e', u'/home/stack/custom-templates/layout-removeController.yaml', u'--stack', u'newtonoc', u'--debug', u'--log-file', u'overcloudDeploy_11_14_17__13_53_12.log', u'--neutron-flat-networks', u'phys_pcie1_0,phys_pcie1_1,phys_pcie4_0,phys_pcie4_1', u'--neutron-network-vlan-ranges', u'datacentre:101:200', u'--neutron-disable-tunneling', u'--verbose', u'--timeout', u'180']
```

```
options: Namespace(access_key='', access_secret='***', access_token='***', access_token_endpoint='', access_token_type='', aodh_endpoint='', auth_type='', auth_url='https://192.200.0.2:13000/v2.0', authorization_code='', cacert=None, cert='', client_id='', client_secret='***', cloud='', consumer_key='', consumer_secret='***', debug=True, default_domain='default', default_domain_id='', default_domain_name='', deferred_help=False, discovery_endpoint='', domain_id='', domain_name='', endpoint='', identity_provider='', identity_provider_url='', insecure=None, inspector_api_version='1', inspector_url=None, interface='', key='', log_file=u'overcloudDeploy_11_14_17__13_53_12.log', murano_url='', old_profile=None, openid_scope='', os_alarming_api_version='2', os_application_catalog_api_version='1', os_baremetal_api_version='1.15', os_beta_command=False, os_compute_api_version='', os_container_infra_api_version='1', os_data_processing_api_version='1.1', os_data_processing_url='', os_dns_api_version='2', os_identity_api_version='', os_image_api_version='1', os_key_manager_api_version='1', os_metrics_api_version='1', os_network_api_version='', os_object_api_version='',
```

```
os_orchestration_api_version='1', os_project_id=None, os_project_name=None,
os_queues_api_version='2', os_tripleoclient_api_version='1', os_volume_api_version='',
os_workflow_api_version='2', passcode='', password='***', profile=None, project_domain_id='',
project_domain_name='', project_id='', project_name='admin', protocol='', redirect_uri='',
region_name='', roles='', timing=False, token='***', trust_id='', url='', user='',
user_domain_id='', user_domain_name='', user_id='', username='admin', verbose_level=3,
verify=None)
```

Auth plugin password selected

```
Starting new HTTPS connection (1): 192.200.0.2
"POST /v2/action_executions HTTP/1.1" 201 1696
HTTP POST https://192.200.0.2:13989/v2/action_executions 201
Overcloud Endpoint: http://172.25.22.109:5000/v2.0
Overcloud Deployed
clean_up DeployOvercloud:
END return value: 0
```

```
real    54m17.197s
user    0m3.421s
sys     0m0.670s
```

Verifica servizi cloud nel controller

Verificare che tutti i servizi gestiti vengano eseguiti correttamente sui nodi del controller.

```
[heat-admin@pod1-controller-2 ~]$ sudo pcs status
```

Finalizzazione dei router dell'agente L3

Controllare i router per verificare che gli agenti L3 siano ospitati correttamente. Assicurarsi di generare il file di overcloud quando si esegue questo controllo.

Passaggio 1. Trovare il nome del router.

```
[stack@director~]$ source corerc
[stack@director ~]$ neutron router-list
```

```

+-----+-----+-----+-----+
-----+-----+-----+-----+
| id | name | distributed | ha |
external_gateway_info |
+-----+-----+-----+-----+
-----+-----+-----+-----+
| d814dc9d-2b2f-496f-8c25-24911e464d02 | main | {"network_id": "18c4250c-e402-428c-87d6-
a955157d50b5", | False | True |

```

Nell'esempio, il nome del router è main.

Passaggio 2. Elencare tutti gli agenti L3 per trovare l'UUID del nodo non riuscito e del nuovo nodo.

```
[stack@director ~]$ neutron agent-list | grep "neutron-l3-agent"
```

```

| 70242f5c-43ab-4355-abd6-9277f92e4ce6 | L3 agent | pod1-controller-0.localdomain |
nova | :- ) | True | neutron-l3-agent |

```

```

| 8d2ffbcbb6ff42cd-b5b8-da31d8da8a40 | L3 agent | pod1-controller-2.localdomain |
nova | xxx | True | neutron-l3-agent |
| a410a491-e271-4938-8a43-458084ffe15d | L3 agent | pod1-controller-3.localdomain |
nova | :- ) | True | neutron-l3-agent |
| cb4bc1ad-ac50-42e9-ae69-8a256d375136 | L3 agent | pod1-controller-1.localdomain |
nova | :- ) | True | neutron-l3-agent |

```

Passaggio 3. In questo esempio, l'agente L3 corrispondente a `pod1-controller-2.localdomain` deve essere rimosso dal router e quello corrispondente a `pod1-controller-3.localdomain` deve essere aggiunto al router.

```
[stack@director ~]$ neutron l3-agent-router-remove 8d2ffbcbb6ff42cd-b5b8-da31d8da8a40 main
```

Removed router main from L3 agent

```
[stack@director ~]$ neutron l3-agent-router-add a410a491-e271-4938-8a43-458084ffe15d main
```

Added router main to L3 agent

Passaggio 4. Controllare l'elenco aggiornato degli agenti L3.

```
[stack@director ~]$ neutron l3-agent-list-hosting-router main
```

```

+-----+-----+-----+-----+
----+-----+
| id | host | admin_state_up |
alive | ha_state |
+-----+-----+-----+-----+
----+-----+
| 70242f5c-43ab-4355-abd6-9277f92e4ce6 | pod1-controller-0.localdomain | True | :- )
| standby |
| a410a491-e271-4938-8a43-458084ffe15d | pod1-controller-3.localdomain | True | :- )
| standby |
| cb4bc1ad-ac50-42e9-ae69-8a256d375136 | pod1-controller-1.localdomain | True | :- )
| active |
+-----+-----+-----+-----+
----+-----+

```

Passaggio 5. Elencare i servizi eseguiti dal nodo del controller rimosso e rimuoverli.

```
[stack@director ~]$ neutron agent-list | grep controller-2
```

```

| 877314c2-3c8d-4666-a6ec-69513e83042d | Metadata agent | pod1-controller-2.localdomain
| | xxx | True | neutron-metadata-agent |
| 8d2ffbcbb6ff42cd-b5b8-da31d8da8a40 | L3 agent | pod1-controller-2.localdomain
nova | xxx | True | neutron-l3-agent |
| 911c43a5-df3a-49ec-99ed-1d722821ec20 | DHCP agent | pod1-controller-2.localdomain
nova | xxx | True | neutron-dhcp-agent |
| a58a3dd3-4cdc-48d4-ab34-612a6cd72768 | Open vSwitch agent | pod1-controller-2.localdomain
| | xxx | True | neutron-openswitch-agent |

```

```
[stack@director ~]$ neutron agent-delete 877314c2-3c8d-4666-a6ec-69513e83042d
```

Deleted agent(s): 877314c2-3c8d-4666-a6ec-69513e83042d

```
[stack@director ~]$ neutron agent-delete 8d2ffbcbb6ff42cd-b5b8-da31d8da8a40
```

Deleted agent(s): 8d2ffbcbb6ff42cd-b5b8-da31d8da8a40

```
[stack@director ~]$ neutron agent-delete 911c43a5-df3a-49ec-99ed-1d722821ec20
```

Deleted agent(s): 911c43a5-df3a-49ec-99ed-1d722821ec20

```
[stack@director ~]$ neutron agent-delete a58a3dd3-4cdc-48d4-ab34-612a6cd72768
```

Deleted agent(s): a58a3dd3-4cdc-48d4-ab34-612a6cd72768

```
[stack@director ~]$ neutron agent-list | grep controller-2
[stack@director ~]$
```

Finalizza servizi di elaborazione

Passaggio 1. Controllare gli elementi dell'elenco dei servizi Nova rimasti dal nodo rimosso ed eliminarli.

```
[stack@director ~]$ nova service-list | grep controller-2
```

```
| 615 | nova-consoleauth | pod1-controller-2.localdomain | internal | enabled | down
| 2017-11-16T16:08:14.000000 | - |
| 618 | nova-scheduler | pod1-controller-2.localdomain | internal | enabled | down
| 2017-11-16T16:08:13.000000 | - |
| 621 | nova-conductor | pod1-controller-2.localdomain | internal | enabled | down
| 2017-11-16T16:08:14.000000 | - |
```

```
[stack@director ~]$ nova service-delete 615
[stack@director ~]$ nova service-delete 618
[stack@director ~]$ nova service-delete 621
```

```
stack@director ~]$ nova service-list | grep controller-2
```

Passaggio 2. Verificare che il processo **consoleauth** venga eseguito su tutti i controller o riavviarlo con questo comando: **riavvio risorse pcs openstack-nova-consoleauth**:

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

```
| 601 | nova-consoleauth | pod1-controller-0.localdomain | internal | enabled | up
| 2017-11-16T21:00:10.000000 | - |
| 608 | nova-consoleauth | pod1-controller-1.localdomain | internal | enabled | up
| 2017-11-16T21:00:13.000000 | - |
| 622 | nova-consoleauth | pod1-controller-3.localdomain | internal | enabled | up
| 2017-11-16T21:00:13.000000 | - |
```

Riavviare la restrizione sui nodi del controller

Passaggio 1. Verificare in tutti i controller la presenza di una route IP verso il cloud 192.0.0.0/8

```
[root@pod1-controller-3 ~]# ip route
default via 172.25.22.1 dev vlan101
11.117.0.0/24 dev vlan17 proto kernel scope link src 11.117.0.12
11.118.0.0/24 dev vlan18 proto kernel scope link src 11.118.0.12
11.119.0.0/24 dev vlan19 proto kernel scope link src 11.119.0.12
11.120.0.0/24 dev vlan20 proto kernel scope link src 11.120.0.12
169.254.169.254 via 192.200.0.1 dev eno1
172.25.22.0/24 dev vlan101 proto kernel scope link src 172.25.22.102
192.0.0.0/8 dev eno1 proto kernel scope link src 192.200.0.103
```

Passaggio 2. Controllare la configurazione **stonith** corrente. Rimuovere i riferimenti al vecchio nodo di controller.

```
[root@pod1-controller-3 ~]# sudo pcs stonith show --full
Resource: my-ipmilan-for-controller-6 (class=stonith type=fence_ipmilan)
Attributes: pcmk_host_list=pod1-controller-1 ipaddr=192.100.0.1 login=admin
passwd=Csco@123Starent lanplus=1
Operations: monitor interval=60s (my-ipmilan-for-controller-6-monitor-interval-60s)
```

```
Resource: my-ipmilan-for-controller-4 (class=stonith type=fence_ipmilan)
Attributes: pcmk_host_list=pod1-controller-0 ipaddr=192.100.0.14 login=admin
passwd=Csco@123Starent lanplus=1
Operations: monitor interval=60s (my-ipmilan-for-controller-4-monitor-interval-60s)
Resource: my-ipmilan-for-controller-7 (class=stonith type=fence_ipmilan)
Attributes: pcmk_host_list=pod1-controller-2 ipaddr=192.100.0.15 login=admin
passwd=Csco@123Starent lanplus=1
Operations: monitor interval=60s (my-ipmilan-for-controller-7-monitor-interval-60s)
```

```
[root@pod1-controller-3 ~]# pcs stonith delete my-ipmilan-for-controller-7
Attempting to stop: my-ipmilan-for-controller-7...Stopped
```

Passaggio 3. Aggiungere la configurazione stonith per il nuovo controller.

```
[root@pod1-controller-3 ~]# sudo pcs stonith create my-ipmilan-for-controller-8 fence_ipmilan
pcmk_host_list=pod1-controller-3 ipaddr=<CIMC_IP> login=admin passwd=<PASSWORD> lanplus=1 op
monitor interval=60s
```

Passaggio 4. Riavviare la schermatura da qualsiasi controller e verificare lo stato.

```
[root@pod1-controller-1 ~]# sudo pcs property set stonith-enabled=true
[root@pod1-controller-3 ~]# pcs status
```

<snip>

```
my-ipmilan-for-controller-1 (stonith:fence_ipmilan): Started pod1-controller-3
my-ipmilan-for-controller-0 (stonith:fence_ipmilan): Started pod1-controller-3
my-ipmilan-for-controller-3 (stonith:fence_ipmilan): Started pod1-controller-3
```