

# Sustitución de PCRF del servidor controlador UCS C240 M4

## Contenido

[Introducción](#)

[Prerequisites](#)

[Copia de seguridad](#)

[Comprobación de estado preliminar](#)

[Deshabilitar el envergadura en el clúster del controlador](#)

[Instalación del nuevo nodo del controlador](#)

[Sustitución del nodo del controlador en la nube](#)

[Preparación para eliminar el nodo de controlador fallido](#)

[Preparación para agregar un nuevo nodo de controlador](#)

[Intervención manual](#)

[Verificar los servicios de nube excesiva en el controlador](#)

[Finalización de los Routers de Agente L3](#)

[Finalización de servicios de cálculo](#)

[Reiniciar el vallado en los nodos del controlador](#)

## Introducción

Este documento describe los pasos necesarios para sustituir un servidor de controlador defectuoso en una configuración Ultra-M que aloja las funciones de red virtual (VNF) de CPS.

## Prerequisites

### Copia de seguridad

En caso de recuperación, Cisco recomienda realizar una copia de seguridad de la base de datos OSPD (DB) con estos pasos:

```
[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
```

### Comprobación de estado preliminar

Es importante comprobar el estado actual del entorno y los servicios de OpenStack y asegurarse de que esté sano antes de continuar con el procedimiento de reemplazo. Puede ayudar a evitar complicaciones en el momento del proceso de reemplazo del controlador.

## Paso 1. Compruebe el estado de OpenStack y la lista de nodos:

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

## Paso 2. Verifique el estado del marcapasos en los controladores.

Inicie sesión en uno de los controladores activos y verifique el estado del marcapasos. Todos los servicios deben estar ejecutándose en los controladores disponibles y se deben detener en el controlador fallido.

```
[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
```

En este ejemplo, Controller-2 está desconectado. Por lo tanto, será sustituida. El controlador 0 y el controlador 1 están funcionando y están ejecutando los servicios de clúster.

### Paso 3. Verifique el estado de MariaDB en los controladores activos.

```
[stack@director] nova list | grep control
| 4361358a-922f-49b5-89d4-247a50722f6d | pod1-controller-0 | ACTIVE | - | Running |
ctlplane=192.200.0.102 |
| d0f57f27-93a8-414f-b4d8-957de0d785fc | pod1-controller-1 | ACTIVE | - | Running |
ctlplane=192.200.0.110 |
```

```
[stack@director ~]$ for i in 192.200.0.102 192.200.0.110 ; do echo "### $i ###" ; ssh heat-
admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_state_comment'\" ; sudo mysql --
exec=\"SHOW STATUS LIKE 'wsrep_cluster_size'\""; done
```

```
*** 192.200.0.152 ***
```

```
Variable_name      Value
wsrep_local_state_comment  Synced
```

```
Variable_name      Value
```

```
wsrep_cluster_size      2
```

```
*** 192.200.0.154 ***
```

```
Variable_name      Value
```

```
wsrep_local_state_comment  Synced
```

```
Variable_name      Value
```

```
wsrep_cluster_size      2
```

Verifique que estas líneas estén presentes para cada controlador activo:

wsrep\_local\_state\_comentario: Sincronizado

wsrep\_cluster\_size: 2

### Paso 4. Verifique el estado de Rabbitmq en los controladores activos. El controlador fallido no debe aparecer en la lista de los nodos que se ejecutan.

```
[heat-admin@pod1-controller-0 ~] sudo rabbitmqctl cluster_status
Cluster status of node 'rabbit@pod1-controller-0' ...
[{nodes, [{disc, ['rabbit@pod1-controller-0', 'rabbit@pod1-controller-1',
                  'rabbit@pod1-controller-2']}]},
{running_nodes, ['rabbit@pod1-controller-1',
                  'rabbit@pod1-controller-0']},
{cluster_name, <<"rabbit@pod1-controller-2.localdomain">>},
{partitions, []},
{alarms, [{'rabbit@pod1-controller-1', []},
          {'rabbit@pod1-controller-0', []}]}]
```

```
[heat-admin@pod1-controller-1 ~] sudo rabbitmqctl cluster_status
Cluster status of node 'rabbit@pod1-controller-1' ...
[{nodes, [{disc, ['rabbit@pod1-controller-0', 'rabbit@pod1-controller-1',
                  'rabbit@pod1-controller-2']}]},
{running_nodes, ['rabbit@pod1-controller-0',
                  'rabbit@pod1-controller-1']},
{cluster_name, <<"rabbit@pod1-controller-2.localdomain">>},
{partitions, []},
{alarms, [{'rabbit@pod1-controller-0', []},
          {'rabbit@pod1-controller-1', []}]}]
```

### Paso 5. Compruebe si todos los servicios de la nube inferior están en estado cargado, activo y en ejecución desde el nodo OSP-D.

```
[stack@director ~]$ systemctl list-units "openstack*" "neutron*" "openvswitch"
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
neutron-dhcp-agent.service	loaded	active	running	OpenStack Neutron DHCP Agent
neutron-openvswitch-agent.service Agent	loaded	active	running	OpenStack Neutron Open vSwitch Agent
neutron-ovs-cleanup.service Cleanup Utility	loaded	active	exited	OpenStack Neutron Open vSwitch Cleanup Utility
neutron-server.service	loaded	active	running	OpenStack Neutron Server
openstack-aodh-evaluator.service service	loaded	active	running	OpenStack Alarm evaluator service
openstack-aodh-listener.service service	loaded	active	running	OpenStack Alarm listener service
openstack-aodh-notifier.service service	loaded	active	running	OpenStack Alarm notifier service
openstack-ceilometer-central.service agent	loaded	active	running	OpenStack ceilometer central agent
openstack-ceilometer-collector.service service	loaded	active	running	OpenStack ceilometer collection service
openstack-ceilometer-notification.service notification agent	loaded	active	running	OpenStack ceilometer notification agent
openstack-glance-api.service named Glance) API server	loaded	active	running	OpenStack Image Service (code- named Glance) API server
openstack-glance-registry.service named Glance) Registry server	loaded	active	running	OpenStack Image Service (code- named Glance) Registry server
openstack-heat-api-cfn.service API Service	loaded	active	running	Openstack Heat CFN-compatible API Service
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 service
openstack-ironic-inspector-dnsmasq.service Ironic Inspector	loaded	active	running	PXE boot dnsmasq service for Ironic Inspector
openstack-ironic-inspector.service for OpenStack Ironic	loaded	active	running	Hardware introspection service for OpenStack Ironic
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 (swift) - Account Reaper
openstack-swift-account.service (swift) - Account Server	loaded	active	running	OpenStack Object Storage (swift) - Account Server
openstack-swift-container-updater.service (swift) - Container Updater	loaded	active	running	OpenStack Object Storage (swift) - Container Updater
openstack-swift-container.service (swift) - Container Server	loaded	active	running	OpenStack Object Storage (swift) - Container Server
openstack-swift-object-updater.service (swift) - Object Updater	loaded	active	running	OpenStack Object Storage (swift) - Object Updater
openstack-swift-object.service (swift) - Object Server	loaded	active	running	OpenStack Object Storage (swift) - Object Server
openstack-swift-proxy.service (swift) - Proxy Server	loaded	active	running	OpenStack Object Storage (swift) - Proxy Server
openstack-zaqar.service Service (code-named Zaqar) Server	loaded	active	running	OpenStack Message Queuing Service (code-named Zaqar) Server
openstack-zaqar@1.service Service (code-named Zaqar) Server Instance 1	loaded	active	running	OpenStack Message Queuing Service (code-named Zaqar) Server Instance 1
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'.

## Deshabilitar el envergadura en el clúster del controlador

```
[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
```

## Instalación del nuevo nodo del controlador

Paso 1. Los pasos para instalar un nuevo servidor UCS C240 M4 y los pasos iniciales de configuración se pueden consultar en la [Guía de Instalación y Servicio del Servidor Cisco UCS C240 M4](#)

Paso 2. Inicie sesión en el servidor con la IP de CIMC.

Paso 3. Realice la actualización del BIOS si el firmware no es conforme a la versión recomendada utilizada anteriormente. Los pasos para la actualización del BIOS se indican a continuación:

[Guía de actualización del BIOS del servidor de montaje en bastidor Cisco UCS C-Series](#)

Paso 4. Verifique el estado de las unidades físicas. Debe ser **Unconfigured Good**. Vaya a **Almacenamiento > Cisco 12G SAS Modular Raid Controller (SLOT-HBA) > Información de unidad física**.

The screenshot shows the Cisco IMC interface for a Cisco 12G SAS Modular Raid Controller. The left sidebar contains navigation menus for Chassis, Compute, Networking, Storage, and Admin. The main content area is titled "Physical Drive Info" and includes a table of physical drives. The table has the following data:

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

Paso 5. Para crear una unidad virtual desde las unidades físicas con RAID Nivel 1: desplácese hasta **Almacenamiento > Controlador de raid modular SAS 12G de Cisco (SLOT-HBA) > Información del controlador > Crear unidad virtual desde unidades físicas no usadas**, como se muestra en la imagen.

Cisco Integrated Management Controller  
Create Virtual Drive from Unused Physical Drives

RAID Level: 1  Enable Full Disk Encryption

Create Drive Groups

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

Drive Groups

Name
No data available

Virtual Drive Properties

Name: RAID1  
Access Policy: Read Write  
Read Policy: No Read Ahead  
Cache Policy: Direct IO  
Disk Cache Policy: Unchanged  
Write Policy: Write Through  
Strip Size (MB): 64k  
Size:  MB

Cisco Integrated Management Controller  
Create Virtual Drive from Unused Physical Drives

RAID Level: 1  Enable Full Disk Encryption

Create Drive Groups

Physical Drives						Selected 0 / Total 0	
ID	Size(MB)	Model	Interface	Type			
No data available							

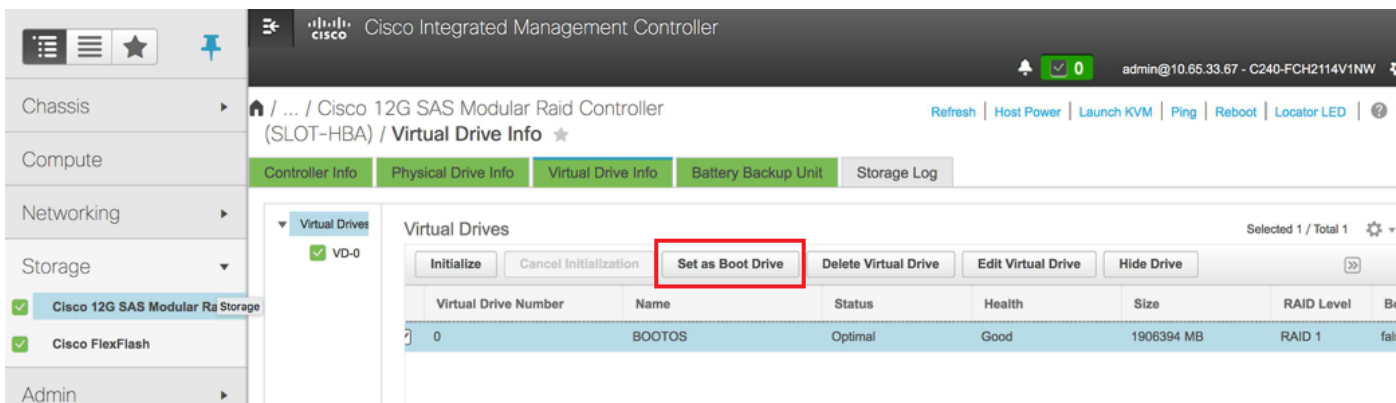
Drive Groups

Name
<input type="checkbox"/> DG [1,2]

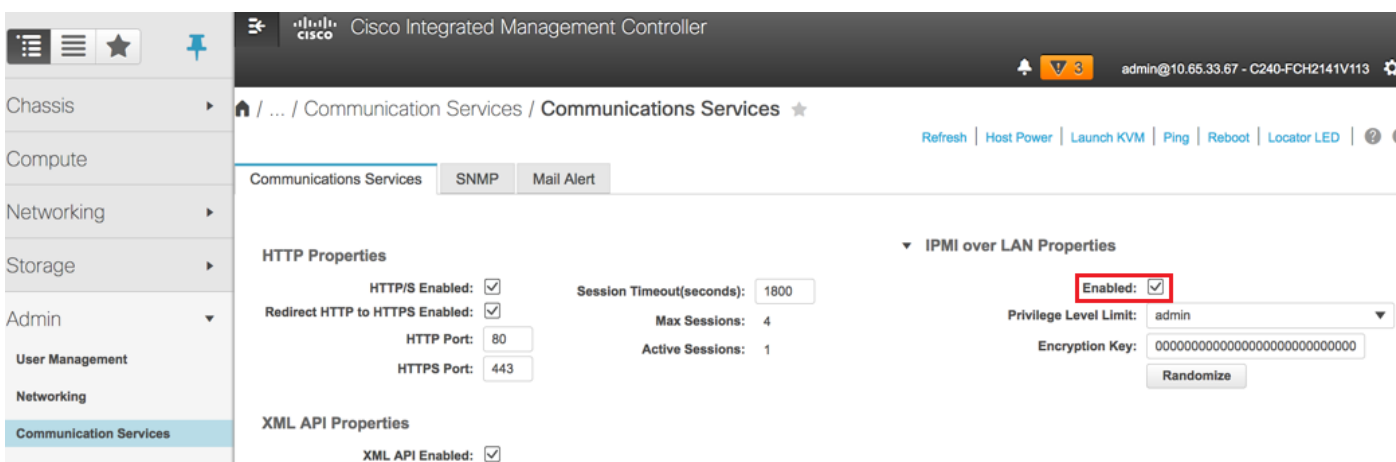
Virtual Drive Properties

Name:   
Access Policy: Read Write  
Read Policy: No Read Ahead  
Cache Policy: Direct IO  
Disk Cache Policy: Unchanged  
Write Policy: Write Through  
Strip Size (MB): 64k  
Size: 1906394 MB

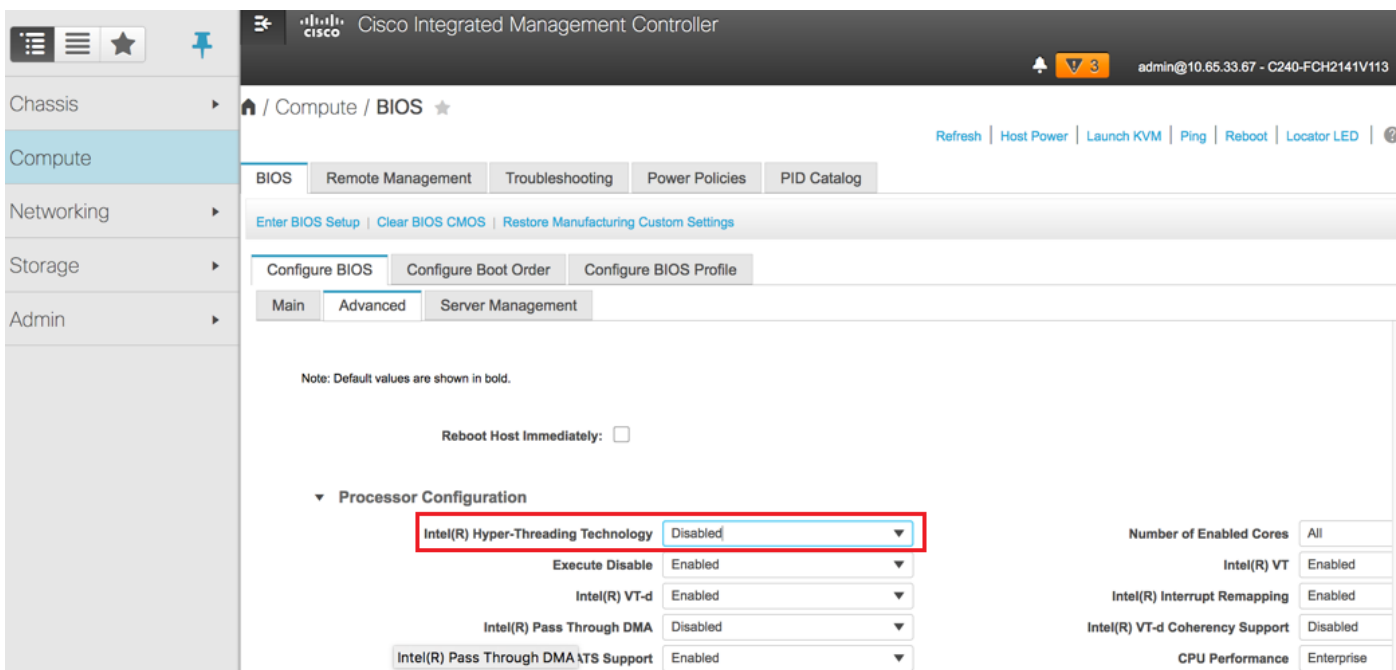
- Seleccione el VD y configure **Set as Boot Drive**:



Paso 6. Para habilitar IPMI sobre LAN, navegue hasta **Admin > Communication Services > Communication Services**.



Paso 7. Para inhabilitar el hiperprocesamiento, navegue hasta **Compute > BIOS > Configure BIOS > Advanced > Processor Configuration**, como se muestra en la imagen.



**Nota:** La imagen se muestra aquí y los pasos de configuración mencionados en esta sección se refieren a la versión de firmware 3.0(3e) y puede haber ligeras variaciones si trabaja en otras versiones.



# Sustitución del nodo del controlador en la nube

Esta sección trata los pasos necesarios para reemplazar el controlador defectuoso por el nuevo en la nube. Para esto, se reutilizaría el script **Deploy.sh** que se utilizó para activar la pila. En el momento de la implementación, en la fase **ControllerNodesPostDeployment**, la actualización fallaría debido a algunas limitaciones en los módulos Puppet. Se requiere una intervención manual antes de reiniciar el script de implementación.

## Preparación para eliminar el nodo de controlador fallido

Paso 1. Identifique el índice del controlador fallido. El índice es el sufijo numérico del nombre del controlador en el resultado de la lista de servidores OpenStack. En este ejemplo, el índice es 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 |
```

Paso 2. Cree un archivo Yaml **~templates/remove-controller.yaml** que defina el nodo que desea eliminar. Utilice el índice encontrado en el paso anterior para la entrada de la lista de recursos.

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

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

```
parameter_defaults:
  CorosyncSettleTries: 5
```

Paso 3. Haga una copia del script de implementación que se utiliza para instalar la nube excesiva e insertar una línea para incluir el archivo **remove-controller.yaml** creado anteriormente.

```
[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
```

Paso 4. Identifique el ID del controlador que se va a reemplazar, con el uso de los comandos mencionados aquí y muévelo al modo de mantenimiento.

```
[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 |
```

Paso 5. Para asegurarse de que la base de datos se ejecute en el momento del procedimiento de reemplazo, quite Galera del control de marcapasos y ejecute este comando en uno de los controladores activos.

```
[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>

## Preparación para agregar un nuevo nodo de controlador

Paso 1. Cree un archivo **controllerRMA.json** con los nuevos detalles del controlador. Asegúrese de que el número de índice del nuevo controlador no se haya utilizado antes. Normalmente, aumente al siguiente número de controlador más alto.

Ejemplo: El más alto anterior era el Controlador-2, así que cree el Controlador-3.

**Nota:** Tenga en cuenta el 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>"
    }
  ]
}
```

Paso 2. Importe el nuevo nodo con el uso del archivo json creado en el paso anterior.

```
[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
```

Paso 3. Establezca el nodo para administrar el estado.

```
[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 |
```

## Paso 4. Ejecute introspection.

```
[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 |
```

Paso 5. Marque el nodo disponible con las nuevas propiedades del controlador. Asegúrese de utilizar el ID del controlador designado para el nuevo controlador, como se utiliza en el archivo **controllerRMA.json**.

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

Paso 6. En el script de implementación, hay una plantilla personalizada llamada **Layout.yaml** que, entre otras cosas, especifica qué direcciones IP se asignan a los controladores para las diversas interfaces. En una pila nueva, hay 3 direcciones definidas para el controlador-0, el controlador-1 y el controlador-2. Cuando agrega un nuevo controlador, asegúrese de agregar una siguiente dirección IP secuencialmente para cada subred.

```
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
```

Paso 7. Ahora ejecute **Deploy-removecontroller.sh** que se creó anteriormente, para quitar el nodo antiguo y agregar el nuevo nodo.

**Nota:** Se espera que este paso falle en **ControllerNodesDeployment\_Step1**. En ese momento, es necesaria la intervención manual.

```
[stack@b10-ospd ~]$ ./deploy-addController.sh
```



## Intervención manual

Paso 1. En el servidor OSP-D, ejecute el comando OpenStack server list para enumerar los controladores disponibles. El controlador recién agregado debe aparecer en la lista.

```
[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-af27-4fb9-8560-75dec3347b4 | pod1-controller-0 | ACTIVE | ctlplane=192.200.0.152 |
overcloud-full |
```

Paso 2. Conéctese a uno de los controladores activos (no al controlador recién agregado) y vea el archivo `/etc/corosync/corosync.conf`. Busque la **lista de nodos** que asigna un **nodo** a cada controlador. Busque la entrada para el nodo fallido y observe su **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
    }
}
```

Paso 3. Inicie sesión en cada uno de los controladores activos. Quite el nodo que ha fallado y reinicie el servicio. En este caso, quite **pod1-controller-2**. No realice esta acción en el controlador recién agregado.

```
[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
```

Paso 4. Ejecute este comando desde uno de los controladores activos para eliminar el nodo fallido del clúster.

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

Paso 5. Ejecute este comando desde uno de los controladores activos para eliminar el nodo fallido del clúster **rabbitmq**.

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

Paso 6. Elimine el nodo fallido de la base de datos de mongo. Para ello, debe encontrar el nodo Mongo activo. Utilice **netstat** para buscar la dirección IP del 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
```

Paso 7. Inicie sesión en el nodo y verifique para ver si es el maestro con el uso de la dirección IP y el número de puerto del comando anterior.

```
[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  
}
```

Si el nodo no es el maestro, inicie sesión en el otro controlador activo y realice el mismo paso.

Paso 8. Desde el maestro, enumere los nodos disponibles con el uso del comando **rs.status()**. Busque el nodo antiguo/sin respuesta e identifique el nombre 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
}

```

Paso 9. Desde el maestro, elimine el nodo fallido con el uso del comando **rs.remove**. Se observan algunos errores al ejecutar este comando, pero verifique el estado una vez más para descubrir que el nodo se ha eliminado:

```

[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

```

Paso 10. Ejecute este comando para actualizar la lista de nodos del controlador activo. Incluya el nuevo nodo del controlador en esta lista.

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

Paso 11. Copie estos archivos de un controlador que ya existe en el nuevo controlador:

**/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
```

Paso 12. Ejecute el comando **cluster node add** desde uno de los controladores que ya existe.

```
[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
```

Paso 13. Inicie sesión en cada controlador y vea el archivo **/etc/corosync/corosync.conf**. Asegúrese de que el nuevo controlador esté en la lista y que el nodo asignado a ese controlador sea el siguiente número en la secuencia que no se ha utilizado previamente. Asegúrese de que este cambio se realice en los 3 controladores.

```
[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
}
```

Por ejemplo, **/etc/corosync/corosync.conf** después de la modificación:

```
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
}

```

**Paso 14.** Reinicie **corosync** en los controladores activos. No inicie **corosync** en el nuevo controlador.

```

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

```

**Paso 15.** Inicie el nuevo nodo de controlador desde uno de los controladores que actúan.

```

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

```

**Paso 16.** Reinicie Galera desde uno de los controladores de actuación.

```

[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

```

**Paso 17.** El clúster está en modo de mantenimiento. Inhabilite el modo de mantenimiento para que se inicien los servicios.

```

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

```

**Paso 18.** Compruebe el estado de los PC para Galera hasta que los 3 controladores aparezcan como maestros en Galera.

**Nota:** Para configuraciones grandes, puede tomar algún tiempo sincronizar las bases de datos.

```
[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 ]
```

**Paso 19. Cambie el clúster al modo de mantenimiento.**

```
[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.el7_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
```

**Paso 20. Vuelva a ejecutar el script de implementación que ejecutó anteriormente. Esta vez debería tener éxito.**

```
[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

```

## Verificar los servicios de nube excesiva en el controlador

Asegúrese de que todos los servicios administrados se ejecuten correctamente en los nodos del controlador.

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

## Finalización de los Routers de Agente L3

Verifique los routers para asegurarse de que los agentes L3 estén correctamente alojados. Asegúrese de generar el archivo overcloudrc cuando realice esta comprobación.

Paso 1. Busque el nombre 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 |

```

En este ejemplo, el nombre del router es main.

Paso 2. Enumera todos los agentes L3 para encontrar UUID del nodo fallido y el nuevo 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 |
| 8d2ffbcb-b6ff-42cd-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          |
```

**Paso 3.** En este ejemplo, el agente L3 que corresponde a **pod1-controller-2.localdomain** debe ser eliminado del router y el que corresponde a **pod1-controller-3.localdomain** debe ser agregado al router.

```
[stack@director ~]$ neutron l3-agent-router-remove 8d2ffbcb-b6ff-42cd-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

**Paso 4.** Verifique la lista actualizada de agentes 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 |
+-----+-----+-----+-----+
----+-----+
```

**Paso 5.** Enumere cualquier servicio que se ejecute desde el nodo de controlador eliminado y quítelos.

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

```
| 877314c2-3c8d-4666-a6ec-69513e83042d | Metadata agent          | pod1-controller-2.localdomain
|          | xxx | True          | neutron-metadata-agent          |
| 8d2ffbcb-b6ff-42cd-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-openvswitch-agent |
```

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

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

```
[stack@director ~]$ neutron agent-delete 8d2ffbcb-b6ff-42cd-b5b8-da31d8da8a40
```

Deleted agent(s): 8d2ffbcb-b6ff-42cd-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 ~]$
```

# Finalización de servicios de cálculo

Paso 1. Verifique los elementos `nova service-list` que quedan del nodo eliminado y elimínelos.

```
[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
```

Paso 2. Asegúrese de que el proceso `consoleauth` se ejecute en todos los controladores o reiniciarlo con el uso de este comando: `pcs resource restart 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 | -
```

## Reiniciar el vallado en los nodos del controlador

Paso 1. Verifique todos los controladores para la ruta IP a la nube inferior `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
```

Paso 2. Verifique la configuración actual de `Stonith`. Elimine cualquier referencia al nodo del controlador antiguo.

```
[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=Csc0@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=Csc0@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
```

**Paso 3. Agregue la configuración stonith para el nuevo controlador.**

```
[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
```

**Paso 4. Reinicie el cercado desde cualquier controlador y verifique el estado.**

```
[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
```