

PCRF-Ersatz für Controller-Server UCS C240 M4

Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Sicherung](#)

[Vorläufige Statusprüfung](#)

[Deaktivierung der Funkübertragung im Controller-Cluster](#)

[Installieren des neuen Controller-Knotens](#)

[Austausch von Controller-Knoten in der Overcloud](#)

[Bereiten Sie das Entfernen des Knoten "Ausgefallener Controller" vor.](#)

[Hinzufügen eines neuen Controller-Knotens vorbereiten](#)

[Manuelle Intervention](#)

[Überprüfen der Overcloud-Services im Controller](#)

[Abschließen der L3 Agent-Router](#)

[Fertigstellen von Computing-Services](#)

[Neustarten der Videoüberwachung auf den Controller-Knoten](#)

Einführung

In diesem Dokument werden die Schritte beschrieben, die erforderlich sind, um einen fehlerhaften Controller-Server in einer Ultra-M-Konfiguration zu ersetzen, die CPS Virtual Network Functions (VNFs) hostet.

Voraussetzungen

Sicherung

Im Falle einer Wiederherstellung empfiehlt Cisco, eine Sicherung der OSPD-Datenbank (DB) mit folgenden Schritten durchzuführen:

```
[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
```

Vorläufige Statusprüfung

Es ist wichtig, den aktuellen Status der OpenStack-Umgebung und -Services zu überprüfen und sicherzustellen, dass sie fehlerfrei ist, bevor Sie mit dem Ersetzen fortfahren. Es kann beim Ersetzen des Controllers Komplikationen vermeiden.

Schritt 1: Überprüfen Sie den Status von OpenStack und die Knotenliste:

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

Schritt 2: Überprüfen Sie den Status des Schrittmachers auf Controllern.

Melden Sie sich bei einem der aktiven Controller an, und überprüfen Sie den Status des Schrittmachers. Alle Dienste sollten auf den verfügbaren Controllern ausgeführt und auf dem ausgefallenen Controller gestoppt werden.

```
[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
```

In diesem Beispiel ist Controller-2 offline. Sie wird daher ersetzt. Controller-0 und Controller-1 sind betriebsbereit und führen die Cluster-Dienste aus.

Schritt 3: Überprüfen Sie den MariaDB-Status in den aktiven Controllern.

```
[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
```

Überprüfen Sie, ob diese Leitungen für jeden aktiven Controller vorhanden sind:

wsrep_local_state_comment: Synchronisiert

wsrep_cluster_size: 2

Schritt 4: Überprüfen Sie den Rabbitmq-Status in den aktiven Controllern. Der ausgefallene Controller sollte nicht in der Liste der Knoten angezeigt werden, die ausgeführt werden.

```
[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', []}]}]
```

Schritt 5: Überprüfen Sie, ob alle unterCloud-Services über den OSP-D-Knoten im Status "load", "active" und "running" sind.

```
[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'.

Deaktivierung der Funkübertragung im Controller-Cluster

```
[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
```

Installieren des neuen Controller-Knotens

Schritt 1: Die Schritte zur Installation eines neuen UCS C240 M4 Servers und die ersten Installationsschritte können im [Cisco UCS C240 M4 Server Installations- und Serviceleitfaden beschrieben](#) werden.

Schritt 2: Melden Sie sich mit der CIMC IP-Adresse beim Server an.

Schritt 3. Führen Sie ein BIOS-Upgrade durch, wenn die Firmware nicht der zuvor verwendeten empfohlenen Version entspricht. Schritte für ein BIOS-Upgrade finden Sie hier:

[BIOS-Upgrade-Leitfaden für Rackmount-Server der Cisco UCS C-Serie](#)

Schritt 4. Überprüfen Sie den Status der physischen Laufwerke. Es muss **nicht konfiguriert** sein **Gut**. Navigieren Sie zu **Storage > Cisco 12G SAS Modular RAID Controller (SLOT-HBA) > Informationen zu physischen Laufwerken**.

The screenshot shows the Cisco IMC interface for a Cisco 12G SAS Modular Raid Controller (SLOT-HBA). The page is titled "Physical Drive Info" and displays a table of physical drives. The table has the following columns: Controller, Physical Drive Number, Status, Health, Boot Drive, and Drive Firmware. Two drives are listed: SLOT-HBA 1 and SLOT-HBA 2, both with a status of "Unconfigured" and "Good" health. The table is highlighted with a red border.

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

Schritt 5: So erstellen Sie eine virtuelle Festplatte von den physischen Festplatten mit RAID-Level 1: navigieren Sie zu **Storage > Cisco 12G SAS Modular Raid Controller (SLOT-HBA) > Controller Info > Create Virtual Drive from Unused Physical Drives** (Virtuelles Laufwerk aus nicht verwendeten physischen Laufwerken erstellen, wie im Bild gezeigt).

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]

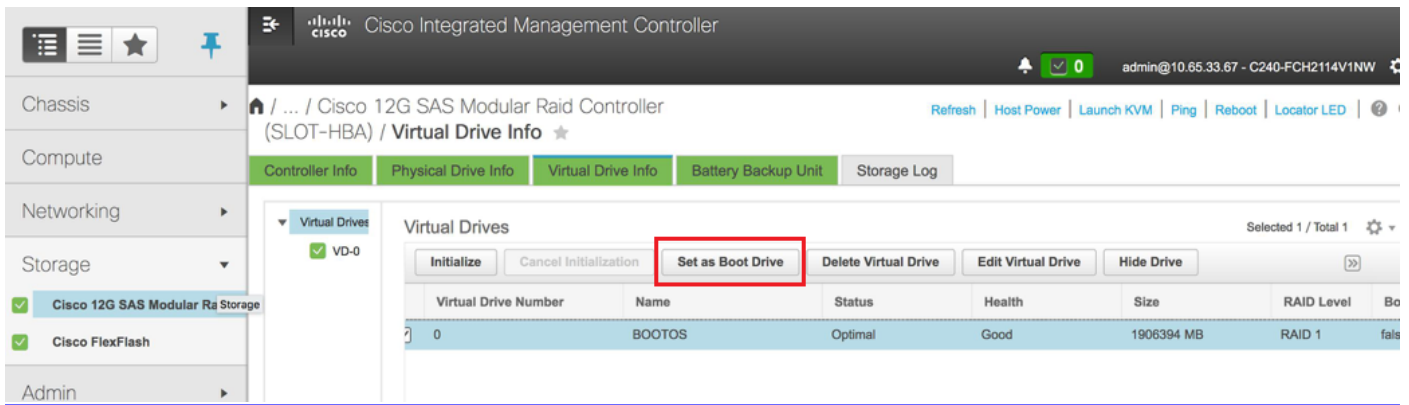
Virtual Drive Properties

Name:

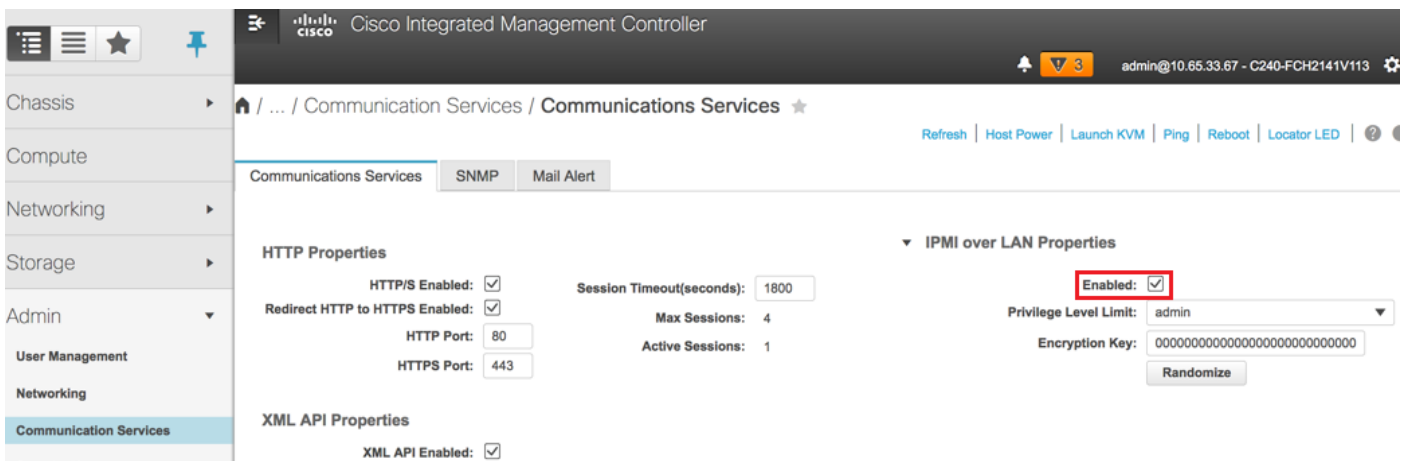
Access Policy: Read Write
 Read Policy: No Read Ahead
 Cache Policy: Direct IO

Disk Cache Policy: Unchanged
 Write Policy: Write Through
 Strip Size (MB): 64k
 Size: 1906394 MB

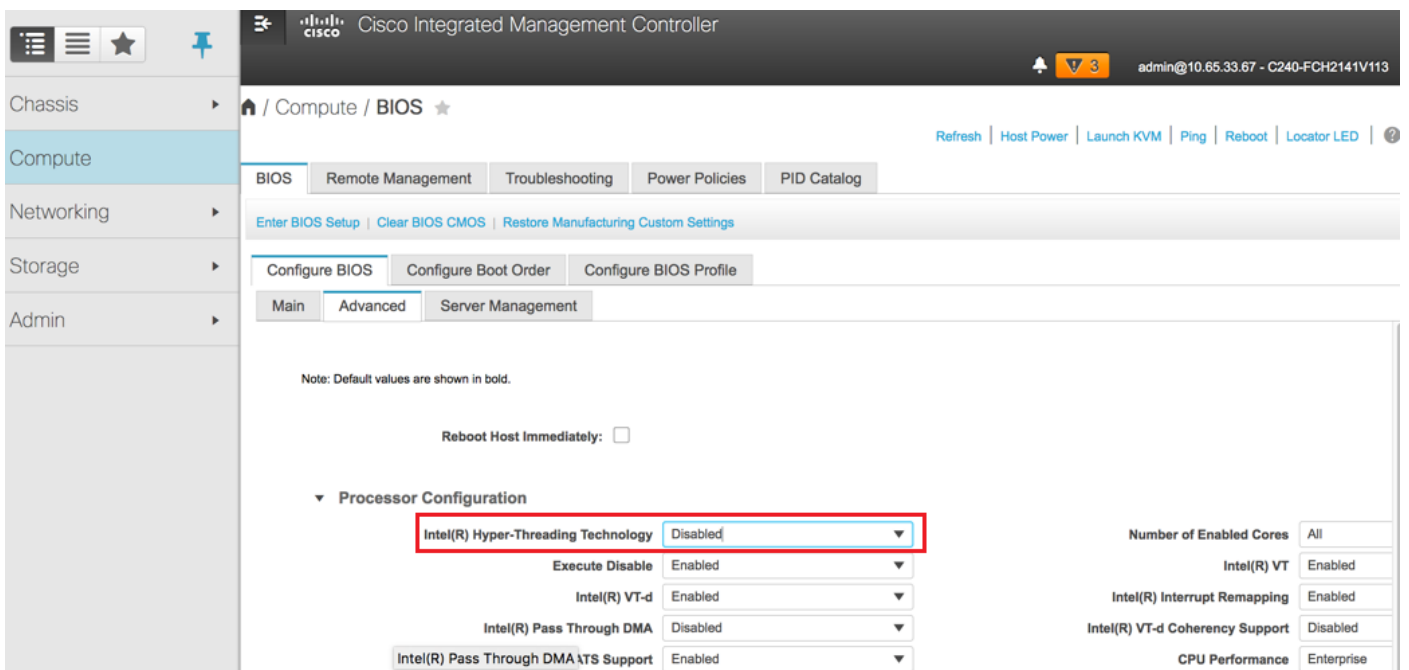
- Wählen Sie die VD aus, und konfigurieren Sie Als Startlaufwerk festlegen:



Schritt 6: Um IPMI über LAN zu aktivieren, wählen Sie **Admin > Communication Services > Communication Services** (Admin > Kommunikationsdienste > Kommunikationsdienste).



Schritt 7: Um Hyperthreading zu deaktivieren, navigieren Sie zu **Compute > BIOS > Configure BIOS > Advanced > Processor Configuration** (Compute > BIOS > Konfigurieren des BIOS > Erweitert > Prozessorkonfiguration), wie im Bild gezeigt.



Hinweis: Das Bild wird hier angezeigt, und die in diesem Abschnitt beschriebenen Konfigurationsschritte beziehen sich auf die Firmware-Version 3.0(3e). Bei anderen Versionen kann es zu geringfügigen Abweichungen kommen.

Austausch von Controller-Knoten in der Overcloud

In diesem Abschnitt werden die Schritte beschrieben, die erforderlich sind, um den fehlerhaften Controller durch den neuen in der Cloud zu ersetzen. Dazu wird das Skript **deploy.sh**, das zum Aufrufen des Stacks verwendet wurde, wiederverwendet. Zum Zeitpunkt der Bereitstellung würde das Update in der Phase der ControllerNodesPostDeployment aufgrund einiger Einschränkungen in den Puppet-Modulen fehlschlagen. Vor dem Neustart des Bereitstellungsskripts ist ein manueller Eingriff erforderlich.

Bereiten Sie das Entfernen des Knoten "Ausgefallener Controller" vor.

Schritt 1: Identifizieren Sie den Index des ausgefallenen Controllers. Der Index ist das numerische Suffix auf dem Controller-Namen in der Ausgabe der OpenStack-Serverliste. In diesem Beispiel ist der Index 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 |
```

Schritt 2: Erstellen Sie eine Yaml-Datei **~templates/remove-controller.yaml**, die den zu löschenden Knoten definiert. Verwenden Sie den im vorherigen Schritt gefundenen Index für den Eintrag in der Ressourcenliste.

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

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

```
parameter_defaults:
  CorosyncSettleTries: 5
```

Schritt 3: Erstellen Sie eine Kopie des Bereitstellungsskripts, das zur Installation der Overcloud verwendet wird, und fügen Sie eine Zeile ein, um die zuvor erstellte Datei **remove-Controller.yaml** einzuschließen.

```
[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
```

Schritt 4: Identifizieren Sie die ID des zu ersetzenden Controllers mithilfe der hier erwähnten Befehle, und verschieben Sie diesen in den Wartungsmodus.

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

```
| 5813a47e-af27-4fb9-8560-75dec3347b4 | 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 |
```

Schritt 5: Um sicherzustellen, dass die DB beim Ersetzen ausgeführt wird, entfernen Sie Galera aus der Schrittmachersteuerung und führen Sie diesen Befehl auf einem der aktiven Controller aus.

```
[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>

Hinzufügen eines neuen Controller-Knotens vorbereiten

Schritt 1: Erstellen Sie eine **ControllerRMA.json**-Datei, die nur die neuen Controller-Details enthält. Stellen Sie sicher, dass die Indexnummer auf dem neuen Controller noch nicht verwendet wurde. In der Regel wird die Erhöhung auf die nächsthöchste Controller-Nummer vorgenommen.

Beispiel: Höchste Vorgeschichte war Controller-2, also erstellen Sie Controller-3.

Hinweis: Achten Sie auf das Json-Format.

```
[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>"
    }
  ]
}
```

Schritt 2: Importieren Sie den neuen Knoten mit der im vorherigen Schritt erstellten json-Datei.

```
[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
```

Schritt 3: Stellen Sie den Knoten auf den Verwaltungsstatus ein.

```
[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 |
```

Schritt 4: Führen Sie eine Introspektion durch.

```
[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 |
```

Schritt 5: Markieren Sie den verfügbaren Knoten mit den neuen Controller-Eigenschaften. Stellen Sie sicher, dass Sie die Controller-ID verwenden, die für den neuen Controller festgelegt ist und in der Datei **controllerRMA.json** verwendet wird.

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

Schritt 6: Im Bereitstellungsskript befindet sich eine benutzerdefinierte Vorlage mit dem Namen **layout.yaml**, die unter anderem angibt, welche IP-Adressen den Controllern für die verschiedenen Schnittstellen zugewiesen werden. In einem neuen Stack sind drei Adressen für Controller-0, Controller-1 und Controller-2 definiert. Wenn Sie einen neuen Controller hinzufügen, stellen Sie sicher, dass Sie für jedes Subnetz eine nächste IP-Adresse in der Sequenz hinzufügen.

```
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
```

Schritt 7: Führen Sie nun die zuvor erstellte **deploy-removecontroller.sh** aus, um den alten Knoten zu entfernen und den neuen Knoten hinzuzufügen.

Hinweis: Dieser Schritt wird in ControllerNodesDeployment_Step1 voraussichtlich

fehlschlagen. An diesem Punkt ist ein manueller Eingriff erforderlich.

```
[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
```

Der Fortschritt/Status der Bereitstellung kann mithilfe der folgenden Befehle überwacht werden:

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

+-----+
+-----+
+-----+
+-----+

| ID                                     | Stack
Name
Time          | Updated Time          | Parent                               | Stack Status | Creation
+-----+
+-----+
+-----+
+-----+

| c1e338f2-877e-4817-93b4-9a3f0c0b3d37 | pod1-AllNodesDeploySteps-5psegydpxij-
ComputeDeployment_Step1-swnuzjixac43
| UPDATE_FAILED |
2017-10-08T14:06:07Z | 2017-11-16T18:09:43Z | e90f00ef-2499-4ec3-90b4-d7def6e97c47 |

| 1db4fef4-45d3-4125-bd96-2cc3297a69ff | pod1-AllNodesDeploySteps-5psegydpxij-
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-
5psegydpxij
| 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-
```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Manuelle Intervention

Schritt 1: Führen Sie auf dem OSP-D-Server den Befehl OpenStack-Serverliste aus, um die verfügbaren Controller aufzulisten. Der neu hinzugefügte Controller sollte in der Liste angezeigt werden.

```

[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 |

```

Schritt 2: Stellen Sie eine Verbindung zu einem der aktiven Controller her (nicht zum neu hinzugefügten Controller), und sehen Sie sich die Datei `/etc/corosync/corosync.conf` an. Suchen Sie die **Nodeliste**, die jedem Controller einen **Knoten** zuweist. Suchen Sie den Eintrag für den ausgefallenen Knoten, und notieren Sie dessen **Knoten**:

```

[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
    }
}

```

Schritt 3: Melden Sie sich bei jedem der aktiven Controller an. Entfernen Sie den ausgefallenen Knoten, und starten Sie den Dienst neu. In diesem Fall entfernen Sie **pod1-controller-2**. Führen Sie diese Aktion nicht auf dem neu hinzugefügten Controller aus.

```

[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
```

Schritt 4: Führen Sie diesen Befehl von einem der aktiven Controller aus, um den ausgefallenen Knoten aus dem Cluster zu löschen.

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

Schritt 5: Führen Sie diesen Befehl von einem der aktiven Controller aus, um den ausgefallenen Knoten aus dem **Rabbitmq**-Cluster zu löschen.

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

Schritt 6: Löschen Sie den ausgefallenen Knoten aus der MongoDB. Dazu müssen Sie den aktiven Knoten Mongo finden. Verwenden Sie **netstat**, um die IP-Adresse des Hosts zu finden.

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

Schritt 7: Melden Sie sich beim Knoten an, und überprüfen Sie, ob es sich um den Master handelt, der die IP-Adresse und die Portnummer des vorherigen Befehls verwendet.

```
[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
}
```

Wenn der Knoten nicht der Master ist, melden Sie sich beim anderen aktiven Controller an, und führen Sie den gleichen Schritt aus.

Schritt 8: Listen Sie im Master die verfügbaren Knoten mit dem Befehl **rs.status()** auf. Suchen Sie den alten/nicht reagierenden Knoten, und identifizieren Sie den Mongo-Knotennamen.

```

[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
}

```

Schritt 9: Löschen Sie den ausgefallenen Knoten mithilfe des Befehls **rs.remove** vom Master. Beim Ausführen dieses Befehls treten einige Fehler auf, überprüfen Sie jedoch den Status erneut, um festzustellen, ob der Knoten entfernt wurde:

```

[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
```

Schritt 10: Führen Sie diesen Befehl aus, um die Liste der aktiven Controller-Knoten zu aktualisieren. Integrieren Sie den neuen Controller-Knoten in dieser Liste.

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

Schritt 11: Kopieren Sie diese Dateien von einem Controller, der bereits vorhanden ist, auf den neuen 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
```

Schritt 12: Führen Sie den Befehl **Clusterknoten add** von einem der Controller aus, der bereits vorhanden ist.

```
[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

Schritt 13: Melden Sie sich bei jedem Controller an, und zeigen Sie die Datei **/etc/corosync/corosync.conf** an. Stellen Sie sicher, dass der neue Controller aufgeführt ist und dass der **Knoten**, der diesem Controller zugewiesen wurde, die nächste Nummer in der Sequenz ist, die zuvor nicht verwendet wurde. Stellen Sie sicher, dass diese Änderung auf allen 3 Controllern vorgenommen wird.

```
[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
}
```

Beispiel: `/etc/corosync/corosync.conf` nach der Änderung:

```
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
}
```

Schritt 14: Starten Sie **corosync** auf den aktiven Controllern neu. Starten Sie **corosync** nicht auf dem neuen Controller.

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

Schritt 15: Starten Sie den neuen Controller-Knoten von einem der aktiven Controller.

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

Schritt 16: Starten Sie Galera von einem der aktiven Controller neu.

```
[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
```

Schritt 17: Der Cluster befindet sich im Wartungsmodus. Deaktivieren Sie den Wartungsmodus, um die Dienste zu starten.

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

Schritt 18: Überprüfen Sie den PC-Status für Galera, bis alle 3 Controller als Master in Galera aufgeführt sind.

Hinweis: Bei großen Konfigurationen kann es einige Zeit dauern, bis DBs synchronisiert sind.

```
[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 ]
```

Schritt 19: Wechseln Sie den Cluster in den Wartungsmodus.

```
[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
```

Schritt 20: Führen Sie das zuvor ausgeführte Bereitstellungsskript erneut aus. Diesmal sollte es Erfolg haben.

```
[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

```

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

```

Überprüfen der Overcloud-Services im Controller

Stellen Sie sicher, dass alle verwalteten Dienste ordnungsgemäß auf den Controller-Knoten ausgeführt werden.

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

Abschließen der L3 Agent-Router

Überprüfen Sie die Router, um sicherzustellen, dass L3-Agenten ordnungsgemäß gehostet werden. Stellen Sie sicher, dass Sie die überlaute Datei beim Durchführen dieser Prüfung sortieren.

Schritt 1: Suchen Sie nach dem Router-Namen.

```

[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 |
```

In diesem Beispiel ist der Name des Routers Main.

Schritt 2: Listen Sie alle L3-Agenten auf, um die UUID des ausgefallenen Knotens und des neuen Knotens zu ermitteln.

```
[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 |
| 8d2ffbcf-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 |
```

Schritt 3: In diesem Beispiel sollte der L3-Agent, der der **POD1-Controller-2.localdomain** entspricht, vom Router entfernt werden, und der Agent, der **POD1-Controller-3.localdomain** entspricht, dem Router hinzugefügt werden.

```
[stack@director ~]$ neutron l3-agent-router-remove 8d2ffbcf-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

Schritt 4: Überprüfen Sie die aktualisierte Liste der L3-Agenten.

```
[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 |
+-----+-----+-----+-----+
----+-----+
```

Schritt 5: Listen Sie alle Dienste auf, die vom entfernten Controller-Knoten ausgeführt werden, und entfernen Sie sie.

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

```
| 877314c2-3c8d-4666-a6ec-69513e83042d | Metadata agent | pod1-controller-2.localdomain |
| xxx | True | neutron-metadata-agent |
| 8d2ffbcf-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 8d2ffbcfb-b6ff-42cd-b5b8-da31d8da8a40
Deleted agent(s): 8d2ffbcfb-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 ~]$
```

Fertigstellen von Computing-Services

Schritt 1: Aktivieren Sie **nova-Service-Listenelemente**, die vom entfernten Knoten übrig geblieben sind, und löschen Sie sie.

```
[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
```

Schritt 2: Stellen Sie sicher, dass der **Konsolenprozess** auf allen Controllern ausgeführt wird, oder starten Sie ihn mithilfe des folgenden Befehls neu: **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 | - |
```

Neustarten der Videoüberwachung auf den Controller-Knoten

Schritt 1: Überprüfen Sie alle Controller auf IP-Route zur Undercloud 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
```

Schritt 2: Überprüfen Sie die aktuelle Konfiguration der Stone. Entfernen Sie alle Verweise auf den alten Controller-Knoten.

```
[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
```

Schritt 3: Fügen Sie für den neuen Controller eine Stonith-Konfiguration hinzu.

```
[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
```

Schritt 4: Starten Sie die Zaunung von einem Controller neu, und überprüfen Sie den Status.

```
[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
```