

컴퓨팅 서버 UCS C240 M4의 PCRF 교체

목차

[소개](#)

[배경 정보](#)

[상태 점검](#)

[백업](#)

[컴퓨팅 노드에서 호스팅되는 VM 식별](#)

[종료할 VM에 상주하는 PCRF 서비스 비활성화](#)

[Nova Aggregate 목록에서 컴퓨팅 노드 제거](#)

[컴퓨팅 노드 삭제](#)

[오버클라우드에서 삭제](#)

[서비스 목록에서 컴퓨팅 노드 삭제](#)

[Neutron 에이전트 삭제](#)

[아이러니한 데이터베이스에서 삭제](#)

[새 컴퓨팅 노드 설치](#)

[오버클라우드에 새 컴퓨팅 노드 추가](#)

[VM 복원](#)

[Nova Aggregate 목록 추가](#)

[ESC\(Elastic Services Controller\)에서 VM 복구](#)

[VM에 상주하는 Cisco Policy and Charging Rules Function\(PCRF\) Services\(Cisco 정책 및 PCRF\(Charging Rules Function\) 서비스 확인\)](#)

[ESC 복구가 실패할 경우 하나 이상의 VM 삭제 및 재구축](#)

[사이트에 대한 최신 ESC 템플릿 가져오기](#)

[파일 수정 절차](#)

[1단계. 템플릿 내보내기 파일을 수정합니다.](#)

[2단계. 수정된 내보내기 템플릿 파일을 실행합니다.](#)

[3단계. VM을 추가하려면 내보내기 템플릿 파일을 수정합니다.](#)

[4단계. 수정된 내보내기 템플릿 파일을 실행합니다.](#)

[5단계. VM에 있는 PCRF 서비스를 확인합니다.](#)

[6단계. 진단을 실행하여 시스템 상태를 확인합니다.](#)

[관련 정보](#)

소개

이 문서에서는 Cisco CPS(Policy Suite) VNF(Virtual Network Functions)를 호스팅하는 Ultra-M 설정에서 결함이 있는 컴퓨팅 서버를 교체하는 데 필요한 단계에 대해 설명합니다.

배경 정보

이 문서는 Cisco Ultra-M 플랫폼에 익숙한 Cisco 직원을 대상으로 하며, 컴퓨팅 서버 교체 시 OpenStack 및 CPS VNF 레벨에서 수행해야 하는 단계를 자세히 설명합니다.

참고:Ultra M 5.1.x 릴리스는 이 문서의 절차를 정의하기 위해 고려됩니다.

상태 점검

컴퓨팅 노드를 교체하기 전에 Red Hat OpenStack Platform 환경의 현재 상태를 확인하는 것이 중요합니다. 컴퓨팅 교체 프로세스가 켜져 있을 때 합병증을 피하려면 현재 상태를 확인하는 것이 좋습니다.

1단계. OSPD(OpenStack Deployment)에서

```
[root@director ~]$ su - stack
[stack@director ~]$ cd ansible
[stack@director ansible]$ ansible-playbook -i inventory-new openstack_verify.yml -e
platform=pcrf
```

2단계. 15분마다 생성되는 종합 상태 보고서에서 시스템의 상태를 확인합니다.

```
[stack@director ~]# cd /var/log/cisco/ultram-health
```

3단계. `ultram_health_os.report` 파일을 확인합니다. XXX 상태로 표시되는 유일한 서비스는 `Neutron-sriov-nic-agent.service`입니다.

4단계. OSPD에서 모든 컨트롤러에 대해 `rabbitmq`가 실행되는지 확인합니다.

```
[stack@director ~]# for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo rabbitmqctl
eval 'rabbit_diagnostics:maybe_stuck().'" ) & done
```

5단계. 스톤이 활성화되었는지 확인합니다.

```
[stack@director ~]# sudo pcs property show stonith-enabled
```

6단계. 모든 컨트롤러에서 PCS 상태를 확인합니다.

- 모든 컨트롤러 노드가 `hauproxy-clone`에서 시작됩니다.
- 모든 컨트롤러 노드는 `Galera`에서 활성 상태입니다.
- 모든 컨트롤러 노드는 `RabbitQ` 아래에서 시작됩니다.
- 컨트롤러 노드 1개는 활성 및 2개는 빨간색입니다.

7단계. OSPD에서

```
[stack@director ~]$ for i in $(nova list | grep controller | awk '{print $12}' | sed
's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo pcs status"
) ;done
```

8단계. OSPD에서 이 명령을 실행하여 모든 openstack 서비스가 활성인지 확인합니다.

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

9단계. 컨트롤러의 CEPH 상태가 `HEALTH_OK`인지 확인합니다.

```
[stack@director ~]# for i in $(nova list | grep controller | awk '{print $12}' | sed 's/ctlplane=//g') ; do (ssh -o StrictHostKeyChecking=no heat-admin@$i "hostname;sudo ceph -s" ) ;done
```

10단계. OpenStack 구성 요소 로그를 확인합니다.오류가 있는지 확인합니다.

Neutron:

```
[stack@director ~]# sudo tail -n 20 /var/log/neutron/{dhcp-agent,l3-agent,metadata-agent,openvswitch-agent,server}.log
```

Cinder:

```
[stack@director ~]# sudo tail -n 20 /var/log/cinder/{api,scheduler,volume}.log
```

Glance:

```
[stack@director ~]# sudo tail -n 20 /var/log/glance/{api,registry}.log
```

11단계. OSPD에서 API에 대해 이러한 확인을 수행합니다.

```
[stack@director ~]$ source
```

```
[stack@director ~]$ nova list
```

```
[stack@director ~]$ glance image-list
```

```
[stack@director ~]$ cinder list
```

```
[stack@director ~]$ neutron net-list
```

12단계. 서비스 상태를 확인합니다.

Every service status should be "up":

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

Every service status should be " :-)":

```
[stack@director ~]$ neutron agent-list
```

Every service status should be "up":

```
[stack@director ~]$ cinder service-list
```

백업

복구 시 다음 단계를 사용하여 OSPD 데이터베이스를 백업하는 것이 좋습니다.

```
[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
```

이 프로세스에서는 인스턴스 가용성에 영향을 주지 않고 노드를 교체할 수 있습니다. 또한 CPS 구성을 백업하는 것이 좋습니다.

Cluster Manager VM에서 CPS VM을 백업하려면 다음을 수행합니다.

```
[root@CM ~]# config_br.py -a export --all /mnt/backup/CPS_backup_$(date +%Y-%m-%d).tar.gz
```

or

```
[root@CM ~]# config_br.py -a export --mongo-all --svn --etc --grafanadb --auth-htpasswd --haproxy /mnt/backup/$(hostname)_backup_all_$(date +%Y-%m-%d).tar.gz
```

컴퓨팅 노드에서 호스팅되는 VM 식별

컴퓨팅 서버에서 호스팅되는 VM을 식별합니다.

```
[stack@director ~]$ nova list --field name,host,networks | grep compute-10  
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-  
10e75d0e134d | pod1-compute-10.localdomain | Replication=10.160.137.161;  
Internal=192.168.1.131; Management=10.225.247.229; tb1-orch=172.16.180.129
```

참고:여기에 표시된 출력에서 첫 번째 열은 UUID(Universally Unique Identifier)에 해당하고, 두 번째 열은 VM 이름이고, 세 번째 열은 VM이 있는 호스트 이름입니다. 이 출력의 매개변수는 후속 섹션에서 사용됩니다.

종료할 VM에 상주하는 PCRF 서비스 비활성화

1단계. VM의 관리 IP에 로그인합니다.

```
[stack@XX-ospd ~]$ ssh root@
```

```
[root@XXXSM3 ~]# monit stop all
```

2단계. VM이 SM, OAM 또는 중재자인 경우 sessionmgr 서비스를 중지합니다.

```
[root@XXXSM3 ~]# cd /etc/init.d  
[root@XXXSM3 init.d]# ls -l sessionmgr*  
-rwxr-xr-x 1 root root 4544 Nov 29 23:47 sessionmgr-27717  
-rwxr-xr-x 1 root root 4399 Nov 28 22:45 sessionmgr-27721  
-rwxr-xr-x 1 root root 4544 Nov 29 23:47 sessionmgr-27727
```

3단계. sessionmgr-xxxxx라는 모든 파일에 대해 service sessionmgr-xxxxx stop을 실행합니다.

```
[root@XXXSM3 init.d]# service sessionmgr-27717 stop
```

Nova Aggregate 목록에서 컴퓨팅 노드 제거

1단계. nova 집계를 나열하고, 이 집계가 호스팅하는 VNF를 기반으로 컴퓨팅 서버에 해당하는 집계를 식별합니다. 일반적으로 <VNFNAME>-SERVICE<X> 형식입니다.

```
[stack@director ~]$ nova aggregate-list
```

```
+-----+-----+-----+
| Id | Name                | Availability Zone |
+-----+-----+-----+
| 29 | POD1-AUTOIT        | mgmt              |
| 57 | VNF1-SERVICE1     | -                 |
| 60 | VNF1-EM-MGMT1     | -                 |
| 63 | VNF1-CF-MGMT1     | -                 |
| 66 | VNF2-CF-MGMT2     | -                 |
| 69 | VNF2-EM-MGMT2     | -                 |
| 72 | VNF2-SERVICE2    | -                 |
| 75 | VNF3-CF-MGMT3     | -                 |
| 78 | VNF3-EM-MGMT3     | -                 |
| 81 | VNF3-SERVICE3     | -                 |
+-----+-----+-----+
```

이 경우 대체할 컴퓨팅 서버는 VNF2에 속합니다. 따라서 해당 집계 목록은 VNF2-SERVICE2입니다.

2단계. 식별된 집계에서 컴퓨팅 노드를 제거합니다(섹션에서 설명하는 호스트 이름으로 제거 컴퓨팅 노드에서 호스팅되는 VM을 식별합니다) ◆◆

```
nova aggregate-remove-host
```

```
[stack@director ~]$ nova aggregate-remove-host VNF2-SERVICE2 pod1-compute-10.localdomain
```

3단계. 컴퓨팅 노드가 집계에서 제거되었는지 확인합니다. 이제 호스트가 집계 아래에 나열되지 않아야 합니다.

```
nova aggregate-show
```

```
[stack@director ~]$ nova aggregate-show VNF2-SERVICE2
```

컴퓨팅 노드 삭제

이 섹션에 언급된 단계는 컴퓨팅 노드에서 호스팅되는 VM과 상관없이 일반적입니다.

오버클라우드에서 삭제

1단계. `delete_node.sh`라는 스크립트 파일을 여기에 표시된 내용과 함께 만듭니다. 언급된 템플릿은 스택 배포에 사용되는 `deploy.sh` 스크립트와 동일한지 확인합니다.

```
delete_node.sh
```

```
openstack overcloud node delete --templates -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 /home/stack/custom-templates/network.yaml -e
/home/stack/custom-templates/ceph.yaml -e /home/stack/custom-templates/compute.yaml -e
/home/stack/custom-templates/layout.yaml -e /home/stack/custom-templates/layout.yaml --stack
```

```
[stack@director ~]$ source stackrc
[stack@director ~]$ /bin/sh delete_node.sh
+ openstack overcloud node delete --templates -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 /home/stack/custom-templates/network.yaml -e
/home/stack/custom-templates/ceph.yaml -e /home/stack/custom-templates/compute.yaml -e
/home/stack/custom-templates/layout.yaml -e /home/stack/custom-templates/layout.yaml --stack
pod1 49ac5f22-469e-4b84-badc-031083db0533
Deleting the following nodes from stack pod1:
- 49ac5f22-469e-4b84-badc-031083db0533
Started Mistral Workflow. Execution ID: 4ab4508a-c1d5-4e48-9b95-ad9a5baa20ae

real    0m52.078s
user    0m0.383s
sys     0m0.086s
```

2단계. OpenStack 스택 작업이 COMPLETE 상태로 이동할 때까지 기다립니다.

```
[stack@director ~]$ openstack stack list
+-----+-----+-----+-----+
| ID                | Stack Name | Stack Status | Creation Time |
Updated Time      |
+-----+-----+-----+-----+
| 5df68458-095d-43bd-a8c4-033e68ba79a0 | pod1 | UPDATE_COMPLETE | 2018-05-08T21:30:06Z | 2018-
05-08T20:42:48Z |
+-----+-----+-----+-----+
```

서비스 목록에서 컴퓨팅 노드 삭제

서비스 목록에서 계산 서비스를 삭제합니다.

```
[stack@director ~]$ source corerc
[stack@director ~]$ openstack compute service list | grep compute-8
| 404 | nova-compute | pod1-compute-8.localdomain | nova | enabled | up | 2018-
05-08T18:40:56.000000 |
```

```
openstack compute service delete
```

```
[stack@director ~]$ openstack compute service delete 404
```

Neutron 에이전트 삭제

연결된 이전 중성자 에이전트를 삭제하고 컴퓨팅 서버용 vswitch 에이전트를 엽니다.

```
[stack@director ~]$ openstack network agent list | grep compute-8
| c3ee92ba-aa23-480c-ac81-d3d8d01dcc03 | Open vSwitch agent | pod1-compute-8.localdomain |
None | False | UP | neutron-openvswitch-agent |
| ec19cb01-abbb-4773-8397-8739d9b0a349 | NIC Switch agent | pod1-compute-8.localdomain |
None | False | UP | neutron-sriov-nic-agent |

openstack network agent delete
```

```
[stack@director ~]$ openstack network agent delete c3ee92ba-aa23-480c-ac81-d3d8d01dcc03
[stack@director ~]$ openstack network agent delete ec19cb01-abbb-4773-8397-8739d9b0a349
```

아이러니한 데이터베이스에서 삭제

아이러니한 데이터베이스에서 노드를 삭제하고 확인합니다.

```
[stack@director ~]$ source stackrc
```

```
nova show
```

```
[stack@director ~]$ nova show pod1-compute-10 | grep hypervisor
| OS-EXT-SRV-ATTR:hypervisor_hostname | 4ab21917-32fa-43a6-9260-02538b5c7a5a
```

```
ironic node-delete
```

```
[stack@director ~]$ ironic node-delete 4ab21917-32fa-43a6-9260-02538b5c7a5a
[stack@director ~]$ ironic node-list (node delete must not be listed now)
```

새 컴퓨팅 노드 설치

새 UCS C240 M4 서버를 설치하는 단계 및 초기 설정 단계는 다음 위치에서 참조할 수 있습니다
[Cisco UCS C240 M4 서버 설치 및 서비스 가이드](#)

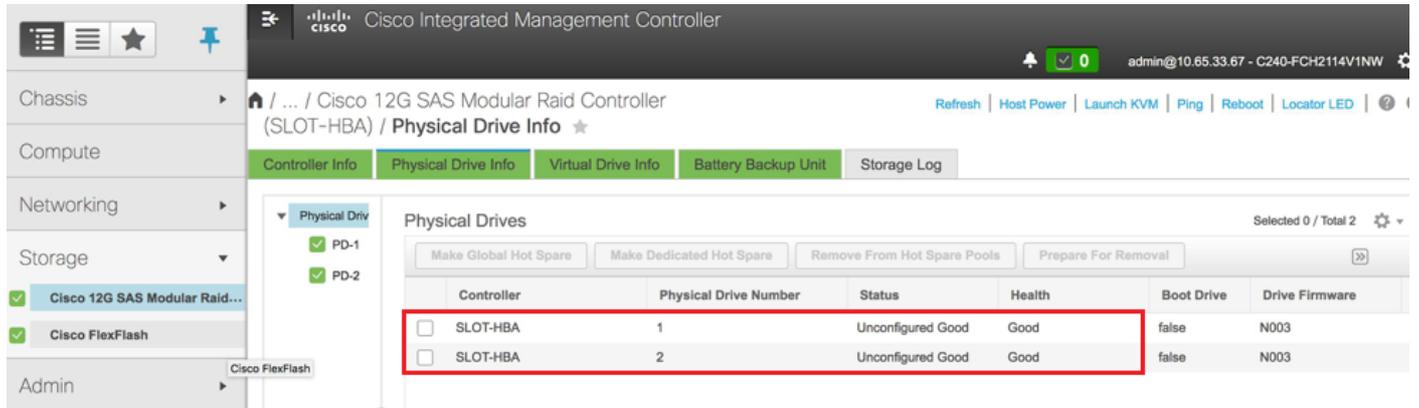
1단계. 서버를 설치한 후 하드 디스크를 각 슬롯에 이전 서버로 삽입합니다.

2단계. CIMC IP를 사용하여 서버에 로그인합니다.

3단계. 펌웨어가 이전에 사용한 권장 버전에 따라 다르다면 BIOS 업그레이드를 수행합니다. BIOS 업그레이드 단계는 다음과 같습니다. [Cisco UCS C-Series 랙 마운트 서버 BIOS 업그레이드 가이드](#)

4단계. 물리적 드라이브의 상태를 확인하려면 **Storage(스토리지) > Cisco 12G SAS Modular Raid Controller(SLOT-HBA) > Physical Drive Info(물리적 드라이브 정보)**로 이동합니다.구성되지 않은 양이어야 합니다.

여기에 표시된 스토리지는 SSD 드라이브일 수 있습니다.



The screenshot shows the Cisco IMC interface for a Cisco 12G SAS Modular Raid Controller (SLOT-HBA). The 'Physical Drive Info' tab is selected, displaying 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	Good	false	N003
<input type="checkbox"/> SLOT-HBA	2	Unconfigured Good	Good	false	N003

5단계. RAID 레벨 1을 사용하여 물리적 드라이브에서 가상 드라이브를 생성하려면 **Storage(스토리지) > Cisco 12G SAS Modular Raid Controller(SLOT-HBA) > Controller Info(컨트롤러 정보) > Create Virtual Drive from Unused Physical Drives(사용되지 않은 물리적 드라이브에서 가상 드라이브 생성)**로 이동합니다.

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

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

DG [1,2]

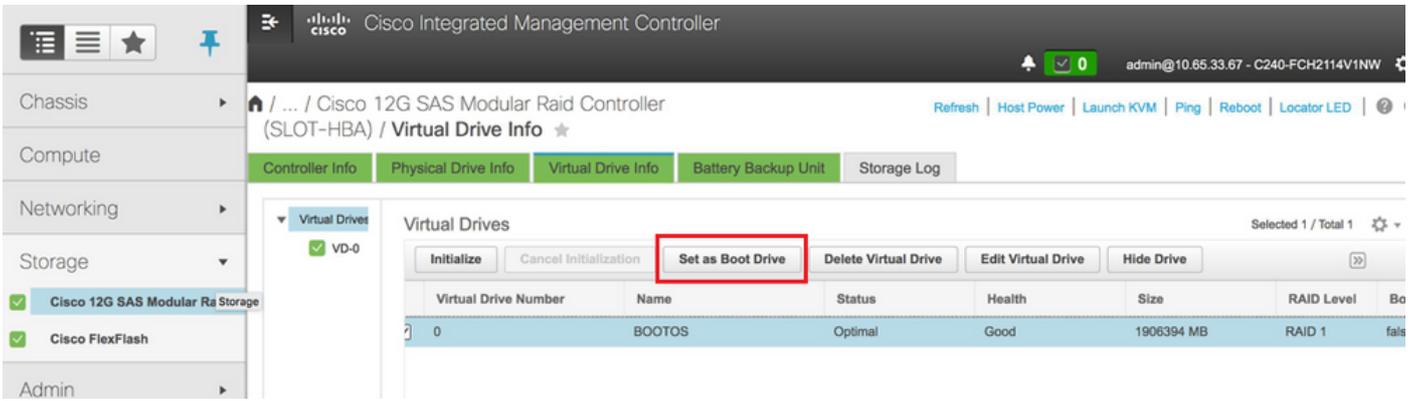
Virtual Drive Properties

Name:

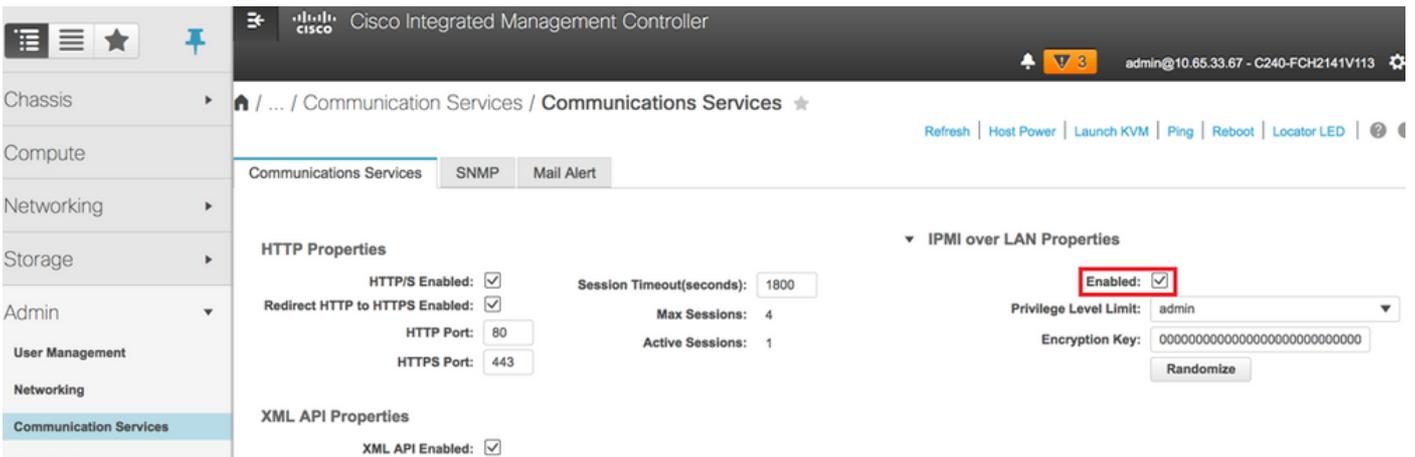
Access Policy: Read Write
 Read Policy: No Read Ahead
 Cache Policy: Direct IO

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

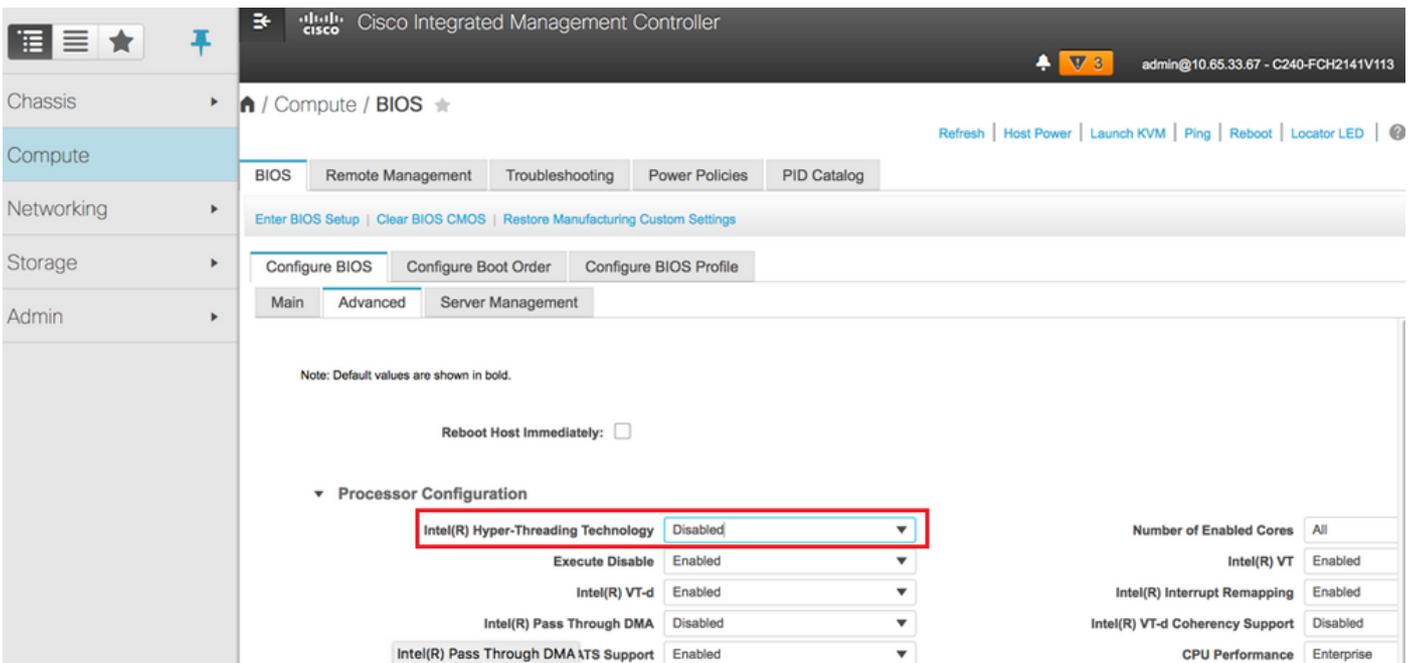
6단계. VD를 선택하고 이미지에 표시된 대로 Set as Boot Drive(부팅 드라이브로 설정)를 구성합니다.



7단계. IPMI over LAN을 활성화하려면 이미지에 표시된 대로 Admin(관리) > Communication Services(통신 서비스) > Communication Services(통신 서비스)로 이동합니다.



8단계. 이미지에 표시된 대로 하이퍼스레딩을 비활성화하려면 Compute(컴퓨팅) > BIOS > Configure BIOS(BIOS 구성) > Advanced(고급) > Processor Configuration(프로세서 컨피그레이션)으로 이동합니다.



참고: 여기에 표시된 이미지와 이 섹션에 언급된 컨피그레이션 단계는 펌웨어 버전 3.0(3e)에 대한 참조이며, 다른 버전에서 작업하는 경우 약간의 차이가 있을 수 있습니다

오버클라우드에 새 컴퓨팅 노드 추가

이 섹션에 언급된 단계는 컴퓨팅 노드에서 호스팅되는 VM과 상관없이 일반적입니다.

1단계. 다른 인덱스로 컴퓨팅 서버를 추가합니다.

추가할 새 컴퓨팅 서버의 세부 정보만 사용하여 `add_node.json` 파일을 생성합니다. 새 컴퓨팅 서버의 인덱스 번호가 이전에 사용되지 않는지 확인합니다. 일반적으로 다음으로 높은 컴퓨팅 값을 증가시킵니다.

예: 가장 높은 우선순위는 컴퓨팅-17이므로 2-vnf 시스템의 경우 컴퓨팅-18을 생성했습니다.

참고: json 형식에 유의하십시오.

```
[stack@director ~]$ cat add_node.json
{
  "nodes": [
    {
      "mac": [
        ""

      ],
      "capabilities": "node:compute-18,boot_option:local",
      "cpu": "24",
      "memory": "256000",
      "disk": "3000",
      "arch": "x86_64",
      "pm_type": "pxe_ipmitool",
      "pm_user": "admin",
      "pm_password": "<PASSWORD>",
      "pm_addr": "192.100.0.5"
    }
  ]
}
```

2단계. json 파일을 가져옵니다.

```
[stack@director ~]$ openstack baremetal import --json add_node.json
Started Mistral Workflow. Execution ID: 78f3b22c-5c11-4d08-a00f-8553b09f497d
Successfully registered node UUID 7eddfa87-6ae6-4308-b1d2-78c98689a56e
Started Mistral Workflow. Execution ID: 33a68c16-c6fd-4f2a-9df9-926545f2127e
Successfully set all nodes to available.
```

3단계. 이전 단계에서 설명한 UUID를 사용하여 노드 자체 검사를 실행합니다.

```
[stack@director ~]$ openstack baremetal node manage 7eddfa87-6ae6-4308-b1d2-78c98689a56e
[stack@director ~]$ ironic node-list |grep 7eddfa87
```

```
| 7eddfa87-6ae6-4308-b1d2-78c98689a56e | None | None | power off
| manageable | False |
```

```
[stack@director ~]$ openstack overcloud node introspect 7eddfa87-6ae6-4308-b1d2-78c98689a56e --
provide
```

```
Started Mistral Workflow. Execution ID: e320298a-6562-42e3-8ba6-5ce6d8524e5c
```

```
Waiting for introspection to finish...
```

```
Successfully introspected all nodes.
```

```
Introspection completed.
```

```
Started Mistral Workflow. Execution ID: c4a90d7b-ebf2-4fcb-96bf-e3168aa69dc9
```

```
Successfully set all nodes to available.
```

```
[stack@director ~]$ ironic node-list |grep available
```

```
| 7eddfa87-6ae6-4308-b1d2-78c98689a56e | None | None | power off
| available | False |
```

4단계. ComputeIPs의 **custom-templates/layout.yml**에 IP 주소를 추가합니다. 각 유형에 대한 목록 끝에 주소를 추가합니다. 예를 들면 다음과 같습니다. compute-0.

ComputeIPs:

```
internal_api:
```

```
- 11.120.0.43
```

```
- 11.120.0.44
```

```
- 11.120.0.45
```

```
- 11.120.0.43 <<< take compute-0 .43 and add here
```

```
tenant:
```

```
- 11.117.0.43
```

```
- 11.117.0.44
```

```
- 11.117.0.45
```

```
- 11.117.0.43 << and here
```

```
storage:
```

```
- 11.118.0.43
```

```
- 11.118.0.44
```

```
- 11.118.0.45
```

```
- 11.118.0.43 << and here
```

5단계. 이전에 스택을 구축하는 데 사용한 `deploy.sh` 스크립트를 실행하여 오버클라우드 스택에 새 컴퓨팅 노드를 추가합니다.

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

```
++ openstack overcloud deploy --templates -r /home/stack/custom-templates/custom-roles.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
/home/stack/custom-templates/network.yaml -e /home/stack/custom-templates/ceph.yaml -e
/home/stack/custom-templates/compute.yaml -e /home/stack/custom-templates/layout.yaml --stack
ADN-ultram --debug --log-file overcloudDeploy_11_06_17__16_39_26.log --ntp-server 172.24.167.109
--neutron-flat-networks phys_pcie1_0,phys_pcie1_1,phys_pcie4_0,phys_pcie4_1 --neutron-network-
vlan-ranges datacentre:1001:1050 --neutron-disable-tunneling --verbose --timeout 180
```

```
...
Starting new HTTP connection (1): 192.200.0.1
"POST /v2/action_executions HTTP/1.1" 201 1695
HTTP POST http://192.200.0.1:8989/v2/action_executions 201
Overcloud Endpoint: http://10.1.2.5:5000/v2.0
Overcloud Deployed
clean_up DeployOvercloud:
END return value: 0
```

```
real 38m38.971s
user 0m3.605s
sys 0m0.466s
```

6단계. openstack 스택 상태가 Complete(완료)가 될 때까지 기다립니다.

```
[stack@director ~]$ openstack stack list
+-----+-----+-----+-----+
| ID | Stack Name | Stack Status | Creation Time |
Updated Time |
+-----+-----+-----+-----+
| 5df68458-095d-43bd-a8c4-033e68ba79a0 | ADN-ultram | UPDATE_COMPLETE | 2017-11-02T21:30:06Z |
2017-11-06T21:40:58Z |
+-----+-----+-----+-----+
```

7단계. 새 컴퓨팅 노드가 활성 상태인지 확인합니다.

```
[stack@director ~]$ source stackrc
[stack@director ~]$ nova list |grep compute-18
| 0f2d88cd-d2b9-4f28-b2ca-13e305ad49ea | pod1-compute-18 | ACTIVE | - | Running
| ctlplane=192.200.0.117 |

[stack@director ~]$ source corerc
[stack@director ~]$ openstack hypervisor list |grep compute-18
| 63 | pod1-compute-18.localdomain |
```

VM 복원

Nova Aggregate 목록 추가

종합 호스트에 컴퓨팅 노드를 추가하고 호스트가 추가되었는지 확인합니다.

```
nova aggregate-add-host
```

```
[stack@director ~]$ nova aggregate-add-host VNF2-SERVICE2 pod1-compute-18.localdomain
nova aggregate-show
```

```
[stack@director ~]$ nova aggregate-show VNF2-SERVICE2
```

ESC(Elastic Services Controller)에서 VM 복구

1단계. VM이 nova 목록에서 오류 상태입니다.

```
[stack@director ~]$ nova list |grep VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
| 49ac5f22-469e-4b84-badc-031083db0533 | VNF2-DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d
| ERROR | - | NOSTATE |
```

2단계. ESC에서 VM을 복구합니다.

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

Recovery VM Action

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

3단계. 양esc.log를 모니터링합니다.

```
admin@VNF2-esc-esc-0 ~]$ tail -f /var/log/esc/yangesc.log
```

...

```
14:59:50,112 07-Nov-2017 WARN Type: VM_RECOVERY_COMPLETE
14:59:50,112 07-Nov-2017 WARN Status: SUCCESS
14:59:50,112 07-Nov-2017 WARN Status Code: 200
14:59:50,112 07-Nov-2017 WARN Status Msg: Recovery: Successfully recovered VM [VNF2-
DEPLOYM_s9_0_8bc6cc60-15d6-4ead-8b6a-10e75d0e134d].
```

VM에 상주하는 Cisco Policy and Charging Rules Function(PCRF) Services(Cisco 정책 및 PCRF(Charging Rules Function) 서비스 확인)

참고: VM이 종료 상태인 경우 Esc에서 `esc_nc_cli`를 사용하여 전원을 켜십시오.

클러스터 관리자 VM에서 `diagnostics.sh`를 확인하고 복구된 VM에 대해 오류가 발견되면 선택합니다.

1단계. 각 VM에 로그인합니다.

```
[stack@XX-ospd ~]$ ssh root@
```

```
[root@XXXSM03 ~]# monit start all
```

2단계. VM이 SM, OAM 또는 중재자인 경우 먼저 중지된 sessionmgr 서비스를 시작합니다.

sessionmgr-xxxxx라는 모든 파일에 대해 `service sessionmgr-xxxxx start`를 실행합니다.

```
[root@XXXSM03 init.d]# service sessionmgr-27717 start
```

그래도 진단이 명확하지 않은 경우 Cluster Manager VM에서 `build_all.sh`를 수행한 다음 응답 VM에서 VM-init를 수행합니다.

```
/var/gps/install/current/scripts/build_all.sh
```

```
ssh VM e.g. ssh pcrfclient01
```

```
/etc/init.d/vm-init
```

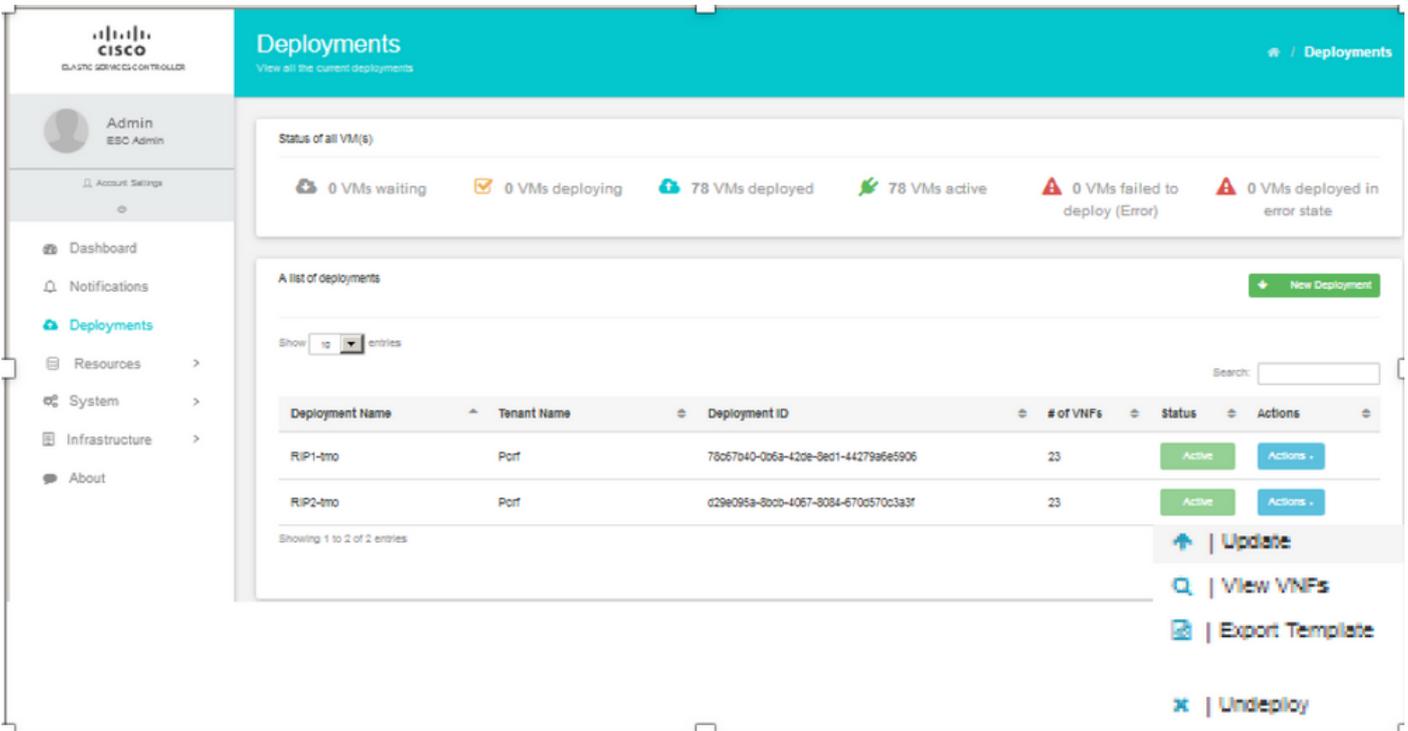
ESC 복구가 실패할 경우 하나 이상의 VM 삭제 및 재구축

ESC 복구 명령(위)이 작동하지 않으면(VM_RECOVERY_FAILED) 개별 VM을 삭제하고 다시 읽습니다.

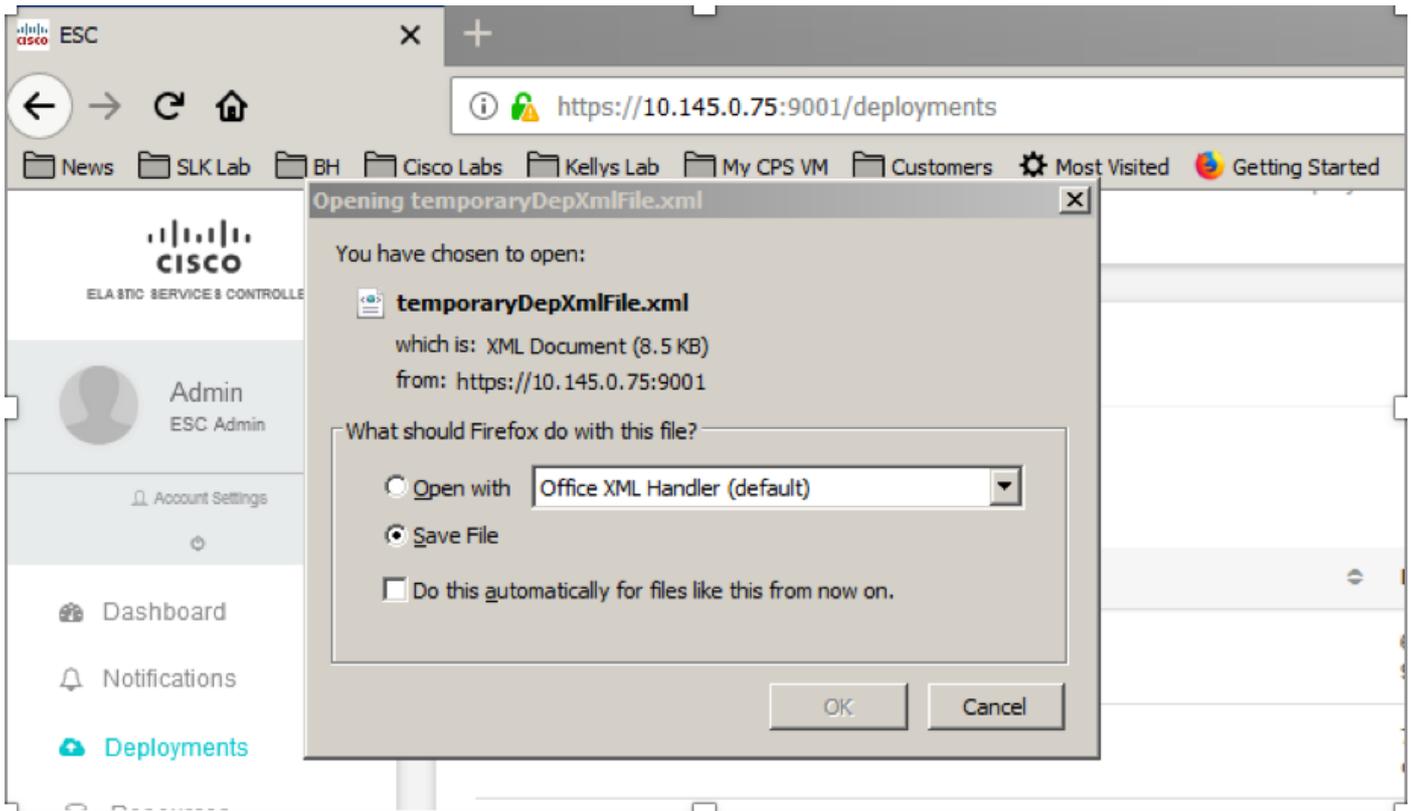
사이트에 대한 최신 ESC 템플릿 가져오기

ESC 포털에서:

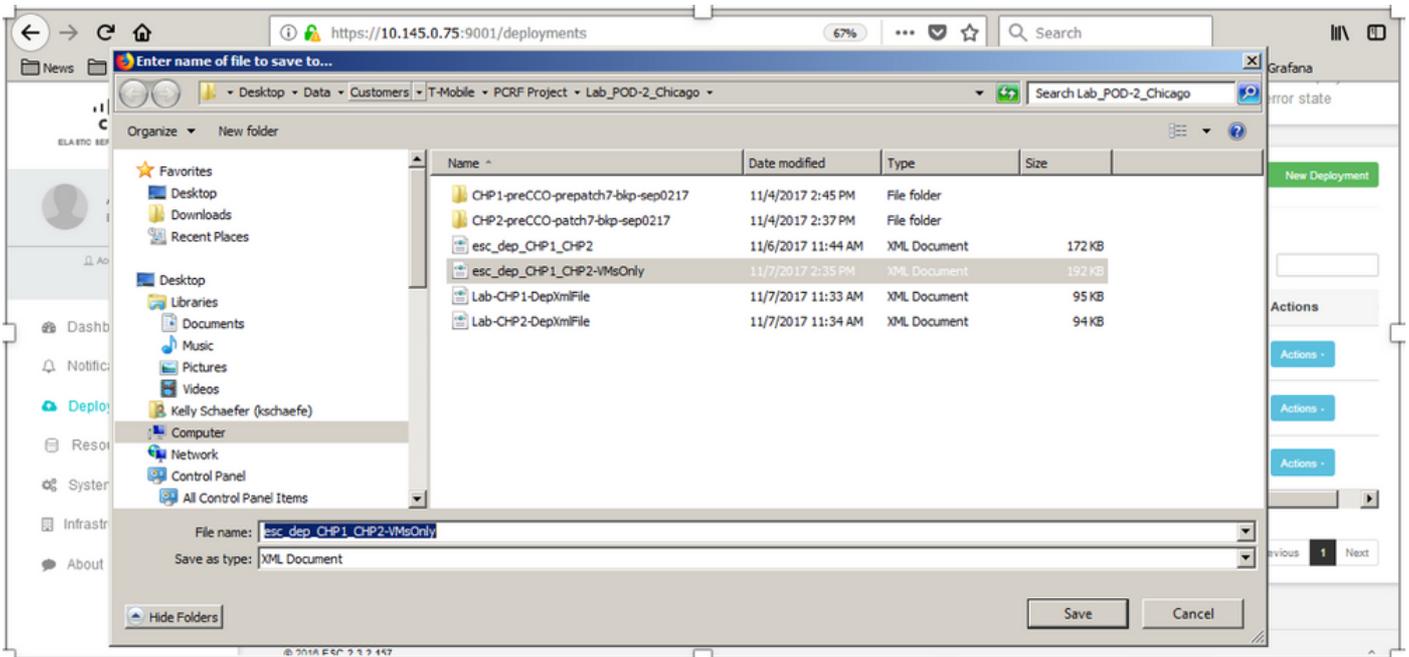
1단계. 커서를 파란색 **작업** 버튼 위에 놓으면 팝업 창이 열리고 이미지에 표시된 대로 템플릿 **내보내기**를 클릭합니다.



2단계. 로컬 시스템에 템플릿을 다운로드하는 옵션이 나타납니다. 이미지에 표시된 대로 파일 저장을 선택합니다.



3단계. 이미지에 표시된 대로 위치를 선택하고 나중에 사용할 수 있도록 파일을 저장합니다.



4단계. 사이트를 삭제하려면 Active ESC에 로그인하고 이 디렉토리의 ESC에 저장된 위 파일을 복사합니다.

```
/opt/cisco/esc/cisco-cps/config/gr/tmo/gen
```

5단계. 디렉토리를 /opt/cisco/esc/cisco-cps/config/gr/tmo/gen으로 변경합니다.

```
cd /opt/cisco/esc/cisco-cps/config/gr/tmo/gen
```

파일 수정 절차

1단계. 템플릿 내보내기 파일을 수정합니다.

이 단계에서는 복구해야 하는 VM과 연결된 VM 그룹 또는 그룹을 삭제하도록 내보내기 템플릿 파일을 수정합니다.

내보내기 템플릿 파일은 특정 클러스터에 대한 파일입니다.

해당 클러스터 내에 여러 vm_groups가 있습니다. 각 VM 유형(PD, PS, SM, OM)에 대해 하나 이상의 vm_groups가 있습니다.

참고: 일부 vm_groups에 둘 이상의 VM이 있습니다. 해당 그룹 내의 모든 VM이 삭제되고 다시 추가됩니다.

해당 구축 내에서 삭제할 하나 이상의 vm_groups에 태그를 지정해야 합니다.

예:

```
<vm_group>
```

```
<name>cm</name>
```

<vm_group>을 <vm_group nc:operation="delete">로 변경하고 변경 내용을 저장합니다.

2단계. 수정된 내보내기 템플릿 파일을 실행합니다.

ESC 실행에서 다음을 수행합니다.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli edit-config /opt/cisco/esc/cisco-cps/config/gr/tmo/gen/
```

ESC 포털에서 구축 취소 상태로 이동한 다음 완전히 사라진 하나 이상의 VM을 볼 수 있어야 합니다.

ESC의 /var/log/esc/yangesc.log에서 진행 상황을 추적할 수 있습니다.

예:

```
09:09:12,608 29-Jan-2018 INFO ===== UPDATE SERVICE REQUEST RECEIVED(UNDER TENANT) =====
09:09:12,608 29-Jan-2018 INFO Tenant name: Pcrf
09:09:12,609 29-Jan-2018 INFO Deployment name: WSP1-tmo
09:09:29,794 29-Jan-2018 INFO
09:09:29,794 29-Jan-2018 INFO ===== CONFID TRANSACTION ACCEPTED =====
09:10:19,459 29-Jan-2018 INFO
09:10:19,459 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:10:19,459 29-Jan-2018 INFO Type: VM_UNDEPLOYED
09:10:19,459 29-Jan-2018 INFO Status: SUCCESS
09:10:19,459 29-Jan-2018 INFO Status Code: 200
|
|
|
09:10:22,292 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:10:22,292 29-Jan-2018 INFO Type: SERVICE_UPDATED
09:10:22,292 29-Jan-2018 INFO Status: SUCCESS
09:10:22,292 29-Jan-2018 INFO Status Code: 200
```

3단계. VM을 추가하려면 내보내기 템플릿 파일을 수정합니다.

이 단계에서는 복구 중인 VM과 연결된 VM 그룹 또는 그룹을 다시 추가하기 위해 내보내기 템플릿 파일을 수정합니다.

내보내기 템플릿 파일은 두 구축(cluster1/cluster2)으로 분할됩니다.

각 클러스터 내에 vm_group이 있습니다. 각 VM 유형(PD, PS, SM, OM)에 대해 하나 이상의 vm_groups가 있습니다.

참고: 일부 vm_groups에 둘 이상의 VM이 있습니다. 해당 그룹 내의 모든 VM이 다시 추가됩니다.

예:

```
<vm_group nc:operation="delete">
  <name>cm</name>
```

<vm_group nc:operation="delete">를 <vm_group>으로 변경합니다.

참고:호스트를 교체했기 때문에 VM을 재구축해야 하는 경우 호스트의 호스트 이름이 변경되었을 수 있습니다. HOST의 호스트 이름이 변경된 경우 vm_group의 배치 섹션 내 호스트 이름을 업데이트해야 합니다.

<배치>

```
<type>zone_host</type>

<enforcement>strict</enforcement>

<host>wsstackovs-compute-4.localdomain</host>
```

</placement>

이 MOP를 실행하기 전에 Ultra-M 팀에서 제공하는 대로 이전 섹션에 표시된 호스트의 이름을 새 호스트 이름으로 업데이트합니다.새 호스트를 설치한 후 변경 사항을 저장합니다.

4단계. 수정된 내보내기 템플릿 파일을 실행합니다.

ESC 실행에서 다음을 수행합니다.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli edit-config /opt/cisco/esc/cisco-cps/config/gr/tmo/gen/
```

ESC 포털에서 하나 이상의 VM이 다시 나타난 다음 Active(활성) 상태로 표시될 수 있습니다.

ESC의 /var/log/esc/yangesc.log에서 진행 상황을 추적할 수 있습니다.

예:

```
09:14:00,906 29-Jan-2018 INFO ===== UPDATE SERVICE REQUESTRECEIVED (UNDER TENANT) =====
09:14:00,906 29-Jan-2018 INFO Tenant name: Pcrf
09:14:00,906 29-Jan-2018 INFO Deployment name: WSP1-tmo
09:14:01,542 29-Jan-2018 INFO
09:14:01,542 29-Jan-2018 INFO ===== CONFID TRANSACTION ACCEPTED =====
09:16:33,947 29-Jan-2018 INFO
09:16:33,947 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:16:33,947 29-Jan-2018 INFO Type: VM_DEPLOYED
09:16:33,947 29-Jan-2018 INFO Status: SUCCESS
09:16:33,947 29-Jan-2018 INFO Status Code: 200
|
|
|
09:19:00,148 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:19:00,148 29-Jan-2018 INFO Type: VM_ALIVE
09:19:00,148 29-Jan-2018 INFO Status: SUCCESS
09:19:00,148 29-Jan-2018 INFO Status Code: 200
```

```
|
|
|
09:19:00,275 29-Jan-2018 INFO ===== SEND NOTIFICATION STARTS =====
09:19:00,275 29-Jan-2018 INFO Type: SERVICE_UPDATED
09:19:00,275 29-Jan-2018 INFO Status: SUCCESS
09:19:00,275 29-Jan-2018 INFO Status Code: 200
```

5단계. VM에 있는 PCRF 서비스를 확인합니다.

PCRF 서비스가 다운되었는지 확인하고 서비스를 시작합니다.

```
[stack@XX-ospd ~]$ ssh root@
```

```
[root@XXXSM03 ~]# monsum
[root@XXXSM03 ~]# monit start all
```

VM이 SM, OAM 또는 중재자인 경우 이전에 중지한 sessionmgr 서비스를 시작합니다.

sessionmgr-xxxxx 실행 서비스 sessionmgr-xxxxx라는 이름의 모든 파일의 시작:

```
[root@XXXSM03 init.d]# service sessionmgr-27717 start
```

그래도 진단이 명확하지 않은 경우 Cluster Manager VM에서 **build_all.sh**를 수행한 다음 각 VM에서 VM-init를 수행합니다.

```
/var/qps/install/current/scripts/build_all.sh
```

```
ssh VM e.g. ssh pcrfclient01
/etc/init.d/vm-init
```

6단계. 진단을 실행하여 시스템 상태를 확인합니다.

```
[root@XXXSM03 init.d]# diagnostics.sh
```

관련 정보

- https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installati..
- https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installati..
- [기술 지원 및 문서 - Cisco Systems](#)