

Ultra-M UCS 240M4シングルHDD障害 – ホットスワップ対応手順 – CPAR

内容

[概要](#)

[背景説明](#)

[省略形](#)

[MoPワークフロー](#)

[シングルHDD障害](#)

[コンピューティングサーバのシングルHDD障害](#)

[コンピューティングノードでホストされるVMの特定](#)

[ヘルスチェック](#)

[コントローラサーバのシングルHDD障害](#)

[OSD-Compute ServerのシングルHDD障害](#)

[OSPDサーバのシングルHDD障害](#)

概要

このドキュメントでは、Ultra-Mセットアップでサーバの障害のあるハードディスクドライブ(HDD)を交換するために必要な手順について説明します。

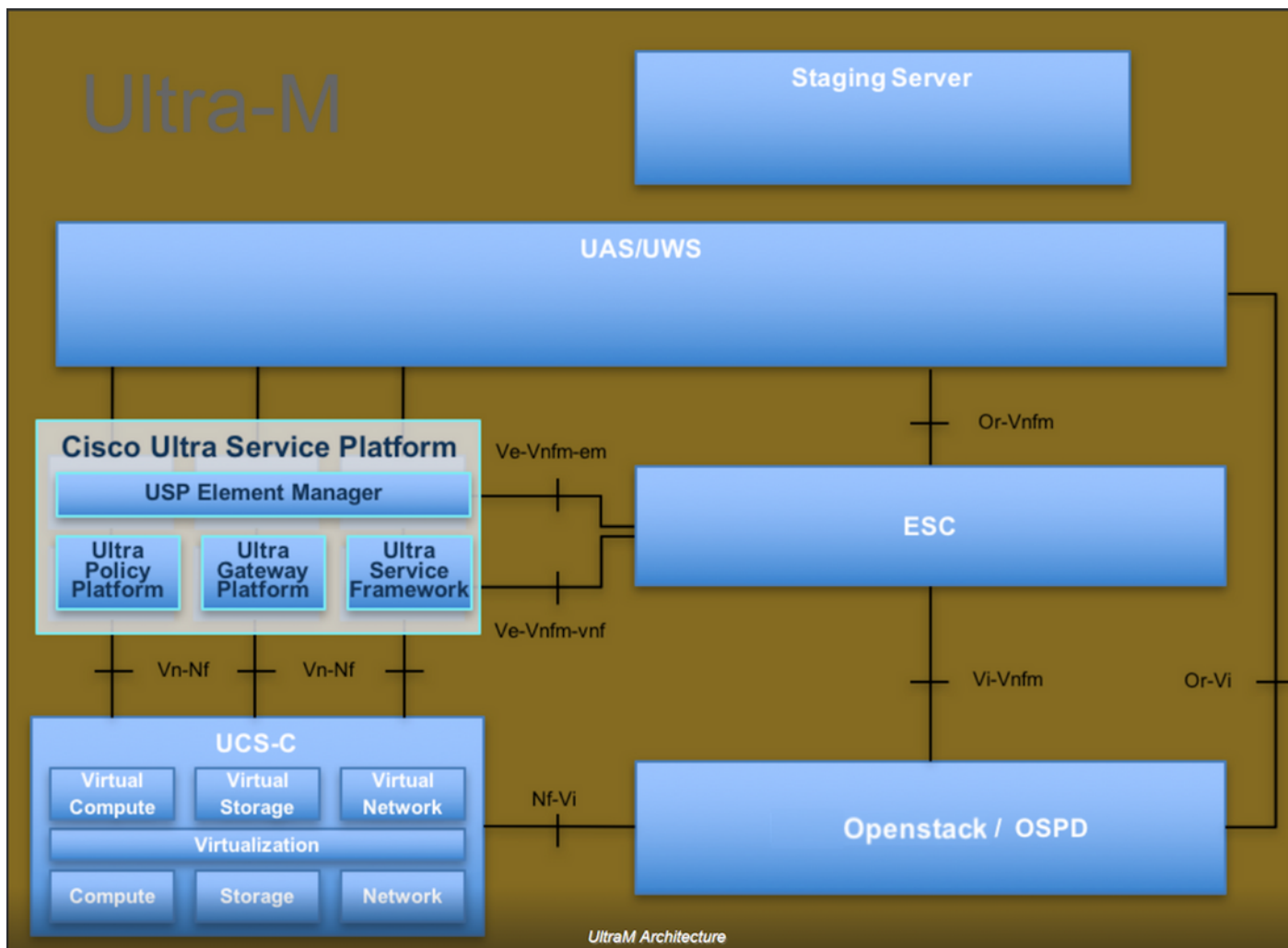
この手順は、ESCがCPARを管理せず、CPARがOpenstackに導入された仮想マシン(VM)に直接インストールされるNEWTONバージョンのOpenstack環境に適用されます。

背景説明

Ultra-Mは、仮想ネットワーク機能(VNF)の導入を簡素化するために設計された、パッケージ化および検証済みの仮想モバイルパケットコアソリューションです。OpenStackは、Ultra-MのVirtual Infrastructure Manager(VIM)であり、次のノードタイプで構成されます。

- 計算
- オブジェクトストレージディスク – コンピューティング (OSD – コンピューティング)
- コントローラ
- OpenStackプラットフォーム – Director(OSPD)

Ultra-Mのアーキテクチャと関連するコンポーネントを次の図に示します。



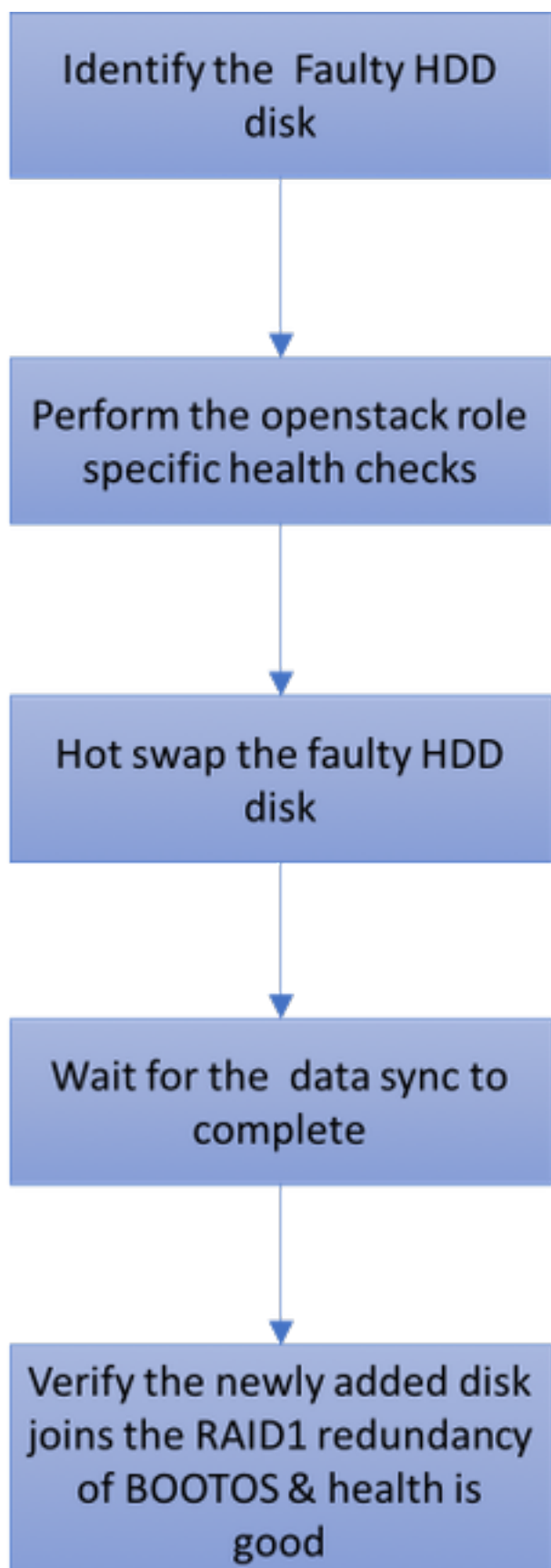
このドキュメントは、Cisco Ultra-Mプラットフォームに精通したシスコ担当者を対象としており、OSPDサーバの交換時にOpenStackレベルで実行する必要がある手順について詳しく説明しています。

注：このドキュメントの手順を定義するために、Ultra M 5.1.xリリースが検討されています。

省略形

VNF	仮想ネットワーク機能
MOP	手続きの方法
OSD	オブジェクトストレージディスク
OSPD	OpenStack Platform Director
HDD	ハードディスクドライブ
SSD	ソリッドステートドライブ
VIM	仮想インフラストラクチャマネージャ
VM	仮想マシン
EM	エレメント マネージャ
UAS	Ultra Automation Services
UUID	汎用一意識別子

MoPワークフロー



シングルHDD障害

1. 各Baremetalサーバには2つのHDDドライブがプロビジョニングされ、Raid 1構成でBOOT

DISKとして機能します。シングルHDD障害の場合、RAID 1レベルの冗長性があるため、障害のあるHDDドライブはホットスワップ可能です。

2. UCS C240 M4サーバで障害のあるコンポーネントを交換する手順については、「[サーバコンポーネントの交換](#)」を参照してください。
3. シングルHDD障害の場合は、障害のあるHDDだけがホットスワップされるため、新しいディスクの交換後にBIOSアップグレード手順は必要ありません。
4. ディスクの交換後、ディスク間のデータ同期を待つ必要があります。完了までに時間がかかることがあります。
5. Openstackベース(Ultra-M)ソリューションでは、UCS 240M4ベアメタルサーバは次のいずれかの役割を担うことができます。コンピューティング、OSD – コンピューティング、コントローラ、およびOSPD。これらの各サーバロールで単一のHDD障害を処理するために必要な手順は同じです。ここでは、ディスクのホットスワップ前に実行するヘルスチェックについて説明します。

コンピューティングサーバのシングルHDD障害

1. コンピューティングノードとして動作するUCS 240M4でHDDドライブの障害が確認された場合は、障害のあるディスクのホットスワップを実行する前にこのヘルスチェックを行います。
2. このサーバで実行されているVMを特定し、機能のステータスが良好であることを確認します。

コンピューティングノードでホストされるVMの特定

コンピューティングサーバでホストされているVMを特定し、アクティブで実行中であることを確認します。

```
[stack@director ~]$ nova list
| 46b4b9eb-a1a6-425d-b886-a0ba760e6114 | AAA-CPAR-testing-instance | pod2-stack-compute-4.localdomain |
```

ヘルスチェック

ステップ1: コマンド/opt/CSCCOar/bin/arstatusをオペレーティングシステム(OS)レベルで実行します。

```
[root@aaa04 ~]# /opt/CSCCOar/bin/arstatus
Cisco Prime AR RADIUS server running      (pid: 24834)
Cisco Prime AR Server Agent running       (pid: 24821)
Cisco Prime AR MCD lock manager running   (pid: 24824)
Cisco Prime AR MCD server running         (pid: 24833)
Cisco Prime AR GUI running                 (pid: 24836)
SNMP Master Agent running                  (pid: 24835)
[root@wscaaa04 ~]#
```

ステップ2: OSレベルでコマンド/opt/CSCCOar/bin/aregcmdを実行し、管理者クレデンシャルを入力します。CPAR Healthが10のうち10で、CPAR CLIを終了していることを確認します。

```
[root@aaa02 logs]# /opt/CSC0ar/bin/aregcmd
Cisco Prime Access Registrar 7.3.0.1 Configuration Utility
Copyright (C) 1995-2017 by Cisco Systems, Inc. All rights reserved.
Cluster:
User: admin
Passphrase:
Logging in to localhost
[ //localhost ]
```

```
LicenseInfo = PAR-NG-TPS 7.2(100TPS:)

PAR-ADD-TPS 7.2(2000TPS:)

PAR-RDDR-TRX 7.2()

PAR-HSS 7.2()
```

```
Radius/
```

```
Administrators/
```

```
Server 'Radius' is Running, its health is 10 out of 10
--> exit
```

ステップ3: コマンドnetstatを実行する | grep diameterとして、すべてのDiameter Routing Agent(DRA)接続が確立されていることを確認します。

ここで説明する出力は、Diameterリンクが必要な環境を対象としています。表示されるリンク数が少ない場合は、分析が必要なDRAからの切断を表します。

```
[root@aa02 logs]# netstat | grep diameter
tcp        0      0 aaa02.aaa.epc.:77 mp1.dra01.d:diameter ESTABLISHED
tcp        0      0 aaa02.aaa.epc.:36 tsa6.dra01:diameter ESTABLISHED
tcp        0      0 aaa02.aaa.epc.:47 mp2.dra01.d:diameter ESTABLISHED
tcp        0      0 aaa02.aaa.epc.:07 tsa5.dra01:diameter ESTABLISHED
tcp        0      0 aaa02.aaa.epc.:08 np2.dra01.d:diameter ESTABLISHED
```

ステップ4:TPSログに、CPARによって処理されている要求が表示されることを確認します。強調表示された値はTPSを表し、これらは注意が必要な値です。

TPSの値は1500を超えることはできません。

```
[root@wscaaa04 ~]# tail -f /opt/CSC0ar/logs/tps-11-21-2017.csv
11-21-2017,23:57:35,263,0
11-21-2017,23:57:50,237,0
11-21-2017,23:58:05,237,0
11-21-2017,23:58:20,257,0
11-21-2017,23:58:35,254,0
11-21-2017,23:58:50,248,0
11-21-2017,23:59:05,272,0
11-21-2017,23:59:20,243,0
11-21-2017,23:59:35,244,0
11-21-2017,23:59:50,233,0
```

ステップ5:name_radius_1_logで「error」または「alarm」メッセージを探します

```
[root@aaa02 logs]# grep -E "error|alarm" name_radius_1_log
```

ステップ6:CPARプロセスが使用するメモリ量を確認するには、次のコマンドを実行します。

```
top | grep radius
```

```
[root@sfraaa02 ~]# top | grep radius
27008 root      20    0 20.228g 2.413g 11408 S 128.3  7.7  1165:41 radius
```

この強調表示された値は、アプリケーションレベルで許可される最大値である7 Gbより小さい必要があります。

ステップ7: ディスクの使用率を確認するには、コマンドdf -hを実行します。

```
[root@aaa02 ~]# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/vg_arucsvm51-lv_root 26G    21G    4.1G  84% /
tmpfs                      1.9G    268K    1.9G   1% /dev/shm
/dev/sda1                   485M    37M    424M   8% /boot
/dev/mapper/vg_arucsvm51-lv_home 23G    4.3G    17G   21% /home
```

80 %を超える場合は、不要なファイルを特定してクリーンアップし、この全体の値を80 %未満にする必要があります。

ステップ8: 「core」ファイルが生成されていないことを確認します。

- コアファイルは、CPARが例外を処理できない場合にアプリケーションがクラッシュした場合に生成され、次の2つの場所で生成されます。

```
[root@aaa02 ~]# cd /cisco-ar/
[root@aaa02 ~]# cd /cisco-ar/bin
```

この2つの場所にコアファイルが存在してはなりません。見つかった場合は、このような例外の根本原因を特定し、デバッグ用のコアファイルを添付するために、Cisco TACケースを作成します。

- ヘルスチェックに問題がなければ、ディスクのホットスワップ手順に進み、データの同期が完了するまで時間がかかるため、待ちます（この手順は正常です）。

[サーバコンポーネントの交換](#)

- ヘルスチェックの手順を繰り返して、コンピューティングノードでホストされているVMのヘルスステータスが復元されていることを確認します。

コントローラサーバのシングルHDD障害

- HDDドライブの障害がコントローラノードとして機能するUCS 240M4で発生する場合は、障害のあるディスクのホットスワップを実行する前に、これらのヘルスチェックを行います。
- コントローラのPacemakerステータスを確認します。
- アクティブなコントローラのいずれかにログインし、パケットメーカーのステータスを確認します。すべてのサービスが使用可能なコントローラで実行され、障害が発生したコントローラで停止している必要があります。

```
[heat-admin@pod2-stack-controller-0 ~]$ sudo pcs status
Cluster name: tripleo_cluster
```

Stack: corosync

Current DC: pod2-stack-controller-2 (version 1.1.15-11.el7_3.4-e174ec8) - partition with quorum
Last updated: Tue Jul 10 10:04:15 2018Last change: Fri Jul 6 09:03:35 2018 by root via
crm_attribute on pod2-stack-controller-0

3 nodes and 19 resources configured

Online: [pod2-stack-controller-0 pod2-stack-controller-1 pod2-stack-controller-2]

Full list of resources:

ip-11.120.0.49(ocf::heartbeat:IPaddr2):Started pod2-stack-controller-1
Clone Set: haproxy-clone [haproxy]
Started: [pod2-stack-controller-0 pod2-stack-controller-1 pod2-stack-controller-2]
Master/Slave Set: galera-master [galera]
Masters: [pod2-stack-controller-0 pod2-stack-controller-1 pod2-stack-controller-2]
ip-192.200.0.110(ocf::heartbeat:IPaddr2):Started pod2-stack-controller-1
ip-11.120.0.44(ocf::heartbeat:IPaddr2):Started pod2-stack-controller-2
ip-11.118.0.49(ocf::heartbeat:IPaddr2):Started pod2-stack-controller-2
Clone Set: rabbitmq-clone [rabbitmq]
Started: [pod2-stack-controller-0 pod2-stack-controller-1 pod2-stack-controller-2]
ip-10.225.247.214(ocf::heartbeat:IPaddr2):Started pod2-stack-controller-1
Master/Slave Set: redis-master [redis]
Masters: [pod2-stack-controller-2]
Slaves: [pod2-stack-controller-0 pod2-stack-controller-1]
ip-11.119.0.49(ocf::heartbeat:IPaddr2):Started pod2-stack-controller-2
openstack-cinder-volume(systemd:openstack-cinder-volume):Started pod2-stack-controller-1

Daemon Status:

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

- アクティブコントローラのMariaDBステータスをチェックします。

```
[stack@director ~]$ nova list | grep control
| b896c73f-d2c8-439c-bc02-7b0a2526dd70 | pod2-stack-controller-0 | ACTIVE | - | Running |
ctlplane=192.200.0.113 |
| 2519ce67-d836-4e5f-a672-1a915df75c7c | pod2-stack-controller-1 | ACTIVE | - | Running |
ctlplane=192.200.0.105 |
| e19b9625-5635-4a52-a369-44310f3e6a21 | pod2-stack-controller-2 | ACTIVE | - | Running |
ctlplane=192.200.0.120 |
```

```
[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 192.200.0.110 ; do echo "*** $i ***" ; ssh
heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_st5 192.200.0.110 ; do echo
*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_st ; do echo
*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_st3 ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_st ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_s1 ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local_9 ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_local2 ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_loca. ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_loc2 ; do
echo "*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_lo0 ; do echo
*** $i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_l0 ; do echo "***
$i ***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep_. ; do echo "*** $i
***" ; ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsrep0 ; do echo "*** $i ***" ;
ssh heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsre. ; do echo "*** $i ***" ; ssh
heat-admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'wsr1 ; do echo "*** $i ***" ; ssh heat-
admin@$i "sudo mysql --exec=\"SHOW STATUS LIKE 'ws2 ; do echo "*** $i ***" ; ssh heat-admin@$i
```

```
"sudo mysql --exec=\"SHOW STATUS LIKE 'w0 ; do echo \"*** $i ***\" ; ssh heat-admin@$i \"sudo mysql --exec=\"SHOW STATUS LIKE '
```

```
*** 192.200.0.102 ***
```

```
Variable_nameValue
```

```
wsrep_local_state_commentSynced
```

```
Variable_nameValue
```

```
wsrep_cluster_size2
```

```
*** 192.200.0.110 ***
```

```
Variable_nameValue
```

```
wsrep_local_state_commentSynced
```

```
Variable_nameValue
```

```
wsrep_cluster_size2
```

- アクティブなコントローラごとに次の行があることを確認します。

```
wsrep_local_state_comment: Synced
```

```
wsrep_cluster_size: 2
```

- アクティブなコントローラのRabbitmqステータスを確認します。

```
[heat-admin@pod2-stack-controller-0 ~]$ sudo rabbitmqctl cluster_status
```

```
Cluster status of node 'rabbit@pod2-stack-controller-0' ...
```

```
[{nodes,[{disc,['rabbit@pod2-stack-controller-0',
```

```
'rabbit@pod2-stack-controller-1',
```

```
'rabbit@pod2-stack-controller-2']}]},
```

```
{running_nodes,['rabbit@pod2-stack-controller-1',
```

```
'rabbit@pod2-stack-controller-2',
```

```
'rabbit@pod2-stack-controller-0']},
```

```
{cluster_name,<<"rabbit@pod2-stack-controller-1.localdomain">>},
```

```
{partitions,[],}
```

```
{alarms,[{'rabbit@pod2-stack-controller-1',[]},
```

```
{'rabbit@pod2-stack-controller-2',[]},
```

```
{'rabbit@pod2-stack-controller-0',[]}]}]}
```

- ヘルスチェックに問題がなければ、ディスクのホットスワップ手順に進み、データの同期が完了するまで時間がかかるため、待ちます（この手順は正常です）。

サーバコンポーネントの交換

- ヘルスチェックの手順を繰り返して、コントローラのヘルスステータスが復元されたことを確認します。

OSD-Compute ServerのシングルHDD障害

- HDDドライブの障害がOSD-Computeノードとして動作するUCS 240M4で見られる場合は、障害が発生したディスクのホットスワップを実行する前に、ヘルスチェックを行います。

1. OSDコンピューティングノードでホストされるVMの特定
2. コンピューティングサーバでホストされているVMを特定します

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

```
| 46b4b9eb-a1a6-425d-b886-a0ba760e6114 | AAA-CPAR-testing-instance | pod2-stack-compute-4.localdomain |
```

- CEPHプロセスはosd-computeサーバ上でアクティブです。


```
[heat-admin@pod2-stack-osd-compute-1 ~]$ systemctl list-units *ceph*
```

```
UNIT LOAD ACTIVE SUB DESCRIPTION
var-lib-ceph-osd-ceph\x2d1.mount loaded active mounted /var/lib/ceph/osd/ceph-1
var-lib-ceph-osd-ceph\x2d10.mount loaded active mounted /var/lib/ceph/osd/ceph-10
var-lib-ceph-osd-ceph\x2d4.mount loaded active mounted /var/lib/ceph/osd/ceph-4
var-lib-ceph-osd-ceph\x2d7.mount loaded active mounted /var/lib/ceph/osd/ceph-7
ceph-osd@1.service loaded active running Ceph object storage daemon
ceph-osd@10.service loaded active running Ceph object storage daemon
ceph-osd@4.service loaded active running Ceph object storage daemon
ceph-osd@7.service loaded active running Ceph object storage daemon
system-ceph\x2ddisk.slice loaded active active system-ceph\x2ddisk.slice
system-ceph\x2dosd.slice loaded active active system-ceph\x2dosd.slice
ceph-mon.target loaded active active ceph target allowing to start/stop all ceph-mon@.service
instances at once
ceph-osd.target loaded active active ceph target allowing to start/stop all ceph-osd@.service
instances at once
ceph-radosgw.target loaded active active ceph target allowing to start/stop all ceph-
radosgw@.service instances at once
ceph.target loaded active active ceph target allowing to start/stop all ceph*@.service instances
at once
```

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.

14 loaded units listed. Pass --all to see loaded but inactive units, too.

To show all installed unit files use 'systemctl list-unit-files'.

- OSD (HDDディスク) からジャーナル(SSD)へのマッピングが良好であることを確認します
-

```
[heat-admin@pod2-stack-osd-compute-1 ~]$ sudo ceph-disk list
```

```
/dev/sda :
/dev/sda1 other, iso9660
/dev/sda2 other, xfs, mounted on /
/dev/sdb :
/dev/sdb1 ceph journal, for /dev/sdc1
/dev/sdb3 ceph journal, for /dev/sdd1
/dev/sdb2 ceph journal, for /dev/sde1
/dev/sdb4 ceph journal, for /dev/sdf1
/dev/sdc :
/dev/sdc1 ceph data, active, cluster ceph, osd.1, journal /dev/sdb1
/dev/sdd :
/dev/sdd1 ceph data, active, cluster ceph, osd.7, journal /dev/sdb3
/dev/sde :
/dev/sde1 ceph data, active, cluster ceph, osd.4, journal /dev/sdb2
/dev/sdf :
/dev/sdf1 ceph data, active, cluster ceph, osd.10, journal /dev/sdb4
```

- ceph healthおよびosd tree mappingが良好であることを確認します。

```
[heat-admin@pod2-stack-osd-compute-1 ~]$ sudo ceph -s
```

```
cluster eb2bb192-b1c9-11e6-9205-525400330666
```

```
health HEALTH_OK
```

```
monmap e1: 3 mons at {pod2-stack-controller-0=11.118.0.10:6789/0,pod2-stack-controller-1=11.118.0.11:6789/0,pod2-stack-controller-2=11.118.0.12:6789/0}
```

```
election epoch 10, quorum 0,1,2 pod2-stack-controller-0,pod2-stack-controller-1,pod2-stack-
```

```
controller-2
osdmap e81: 12 osds: 12 up, 12 in
flags sortbitwise,require_jewel_osds
pgmap v23095222: 704 pgs, 6 pools, 809 GB data, 424 kobjects
2418 GB used, 10974 GB / 13393 GB avail
704 active+clean
client io 1329 kB/s wr, 0 op/s rd, 122 op/s wr
```

```
[heat-admin@pod2-stack-osd-compute-1 ~]$ sudo ceph osd tree
ID WEIGHT TYPE NAME UP/DOWN REWEIGHT PRIMARY-AFFINITY
-1 13.07996 root default
-2 4.35999 host pod2-stack-osd-compute-0
0 1.09000 osd.0 up 1.00000 1.00000
3 1.09000 osd.3 up 1.00000 1.00000
6 1.09000 osd.6 up 1.00000 1.00000
9 1.09000 osd.9 up 1.00000 1.00000
-3 4.35999 host pod2-stack-osd-compute-1
1 1.09000 osd.1 up 1.00000 1.00000
4 1.09000 osd.4 up 1.00000 1.00000
7 1.09000 osd.7 up 1.00000 1.00000
10 1.09000 osd.10 up 1.00000 1.00000
-4 4.35999 host pod2-stack-osd-compute-2
2 1.09000 osd.2 up 1.00000 1.00000
5 1.09000 osd.5 up 1.00000 1.00000
8 1.09000 osd.8 up 1.00000 1.00000
11 1.09000 osd.11 up 1.00000 1.00000
```

- ヘルスチェックに問題がない場合は、ディスクのホットスワップ手順に進み、データの同期が完了するまで時間がかかるのを待ちます。

サーバコンポーネントの交換

- ヘルスチェックの手順を繰り返して、OSD-ComputeノードでホストされているVMのヘルスステータスが復元されることを確認します。

OSPDサーバのシングルHDD障害

- HDDドライブの障害がOSPDノードとして機能するUCS 240M4で発生する場合は、障害のあるディスクのホットスワップを実行する前にヘルスチェックを行います。
- openstackスタックのステータスとノードリストを確認します。

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

- すべてのアンダークラウドサービスがOSP-Dノードからロード済み、アクティブ、および実行中のステータスであるかどうかを確認します。

```
[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-metadata-agent.service loaded active running OpenStack Neutron Metadata Agent
neutron-openvswitch-agent.service loaded active running OpenStack Neutron Open vSwitch Agent
neutron-server.service loaded active running OpenStack Neutron Server
openstack-aodh-evaluator.service loaded active running OpenStack Alarm evaluator service
openstack-aodh-listener.service loaded active running OpenStack Alarm listener service
```

```
openstack-aodh-notifier.service loaded active running OpenStack Alarm notifier service
openstack-ceilometer-central.service loaded active running OpenStack ceilometer central agent
openstack-ceilometer-collector.service loaded active running OpenStack ceilometer collection
service
openstack-ceilometer-notification.service loaded active running OpenStack ceilometer
notification agent
openstack-glance-api.service loaded active running OpenStack Image Service (code-named Glance)
API server
openstack-glance-registry.service loaded active running OpenStack Image Service (code-named
Glance) Registry server
openstack-heat-api-cfn.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 loaded active running OpenStack Ironic Conductor service
openstack-ironic-inspector-dnsmasq.service loaded active running PXE boot dnsmasq service for
Ironic Inspector
openstack-ironic-inspector.service 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 loaded active running OpenStack Object Storage (swift) -
Account Reaper
openstack-swift-account.service loaded active running OpenStack Object Storage (swift) - Account
Server
openstack-swift-container-updater.service loaded active running OpenStack Object Storage (swift)
- Container Updater
openstack-swift-container.service loaded active running OpenStack Object Storage (swift) -
Container Server
openstack-swift-object-updater.service loaded active running OpenStack Object Storage (swift) -
Object Updater
openstack-swift-object.service loaded active running OpenStack Object Storage (swift) - Object
Server
openstack-swift-proxy.service loaded active running OpenStack Object Storage (swift) - Proxy
Server
openstack-zaqar.service loaded active running OpenStack Message Queuing Service (code-named
Zaqar) Server
openstack-zaqar@1.service 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.

lines 1-43

lines 2-44 37 loaded units listed. Pass --all to see loaded but inactive units, too.

To show all installed unit files use 'systemctl list-unit-files'.

lines 4-46/46 (END) lines 4-46/46 (END) lines 4-46/46 (END) lines 4-46/46 (END) lines 4-46/46
(END)

- ヘルスチェックに問題がない場合は、ディスクのホットスワップ手順に進み、データの同期が完了するまで時間がかかるのを待ちます。

サーバコンポーネントの交換

- ヘルスチェックの手順を繰り返して、OSPDノードのヘルスステータスが復元されたことを

確認します。