

Ultra-M AutoVNF群集故障的恢復過程 — vEPC

目錄

[簡介](#)

[背景資訊](#)

[縮寫](#)

[MoP的工作流程](#)

[案例1. UAS集群單個故障的恢復](#)

[狀態檢查](#)

[嘗試連線到UAS時無法連線到Confd伺服器](#)

[將UAS從錯誤狀態中恢復](#)

[案例2. 所有三個UAS\(AutoVNF\)均處於錯誤狀態](#)

[使用uas-check.py指令碼檢查UAS運行狀況](#)

[檢查OpenStack級別上的VM狀態](#)

[檢查Zookeeper檢視](#)

[AutoVNF故障排除 — 流程和任務](#)

[修復處於錯誤狀態的多個UAS](#)

簡介

本文檔介紹在託管StarOS虛擬網路功能(VNF)的Ultra-M設定中恢復Ultra自動化服務(UAS)或AutoVNF群集故障所需的步驟。

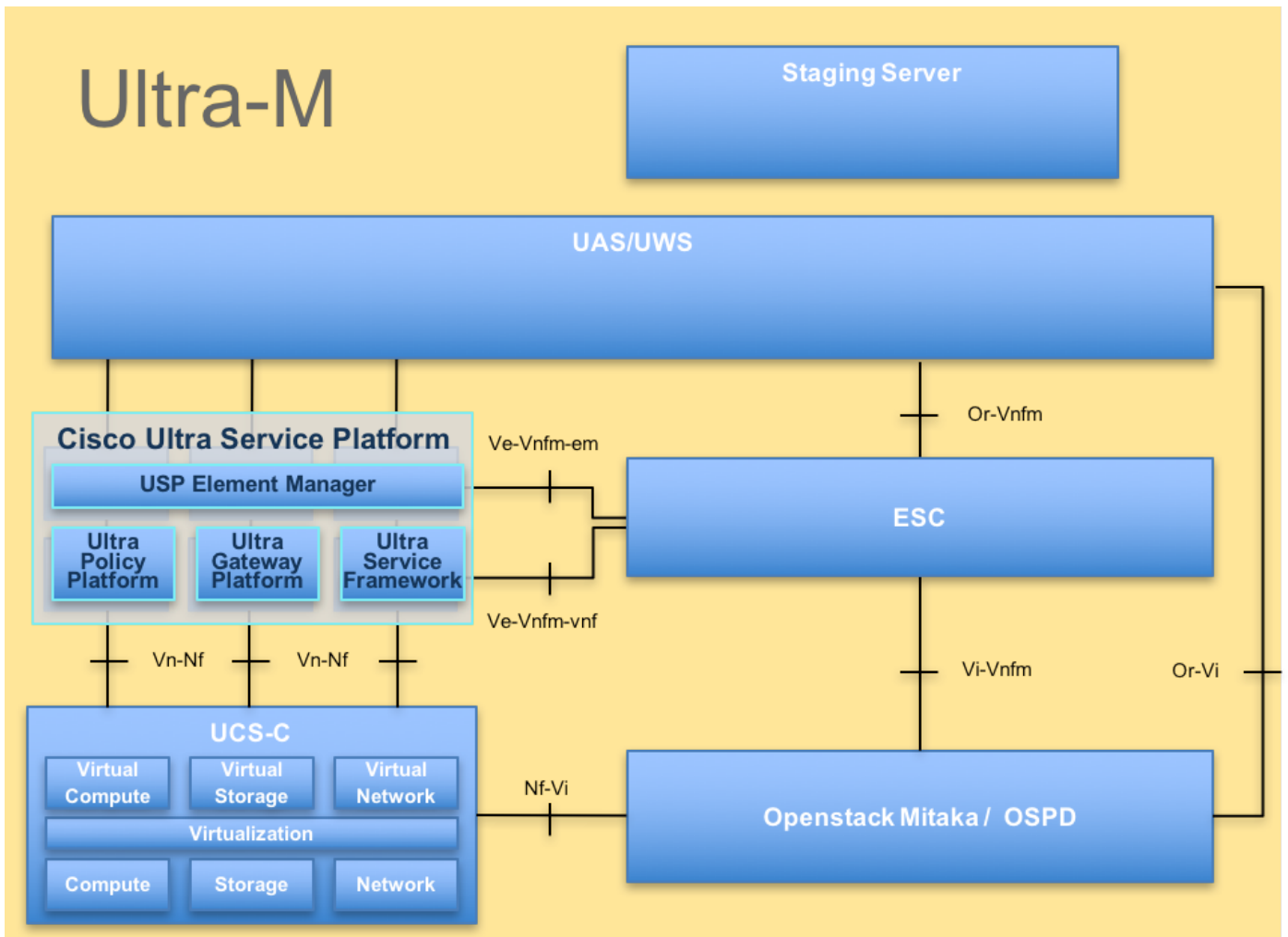
背景資訊

Ultra-M是經過預先打包和驗證的虛擬化移動資料包核心解決方案，旨在簡化VNF的部署。

Ultra-M解決方案包括指定的虛擬機器(VM)型別：

- 自動IT
- 自動部署
- UAS或AutoVNF
- 元素管理器(EM)
- 彈性服務控制器(ESC)
- 控制功能(CF)
- 作業階段功能(SF)

Ultra-M的高級體系結構及涉及的元件如下圖所示：



UltraM體系結構

本文檔適用於熟悉Cisco Ultra-M平台的思科人員。

附註：Ultra M 5.1.x版本用於定義本文檔中的過程。

縮寫

VNF	虛擬網路功能
CF	控制功能
SF	服務功能
ESC	彈性服務控制器
澳門幣	程式方法
OSD	對象儲存磁碟
硬碟	硬碟驅動器
固態硬碟	固態驅動器
VIM	虛擬基礎架構管理員
虛擬機器	虛擬機器
EM	元素管理器
UAS	Ultra自動化服務
UUID	通用唯一識別符號

MoP的工作流程



案例1. UAS集群單個故障的恢復

狀態檢查

1. Ultra-M Manager執行Ultra-M節點的運行狀況檢查。導航到reports/var/log/cisco/ultrum-health/directory和grep以檢視UAS報表。

```
[stack@pod1-ospd ultram-health]$ more ultram_health_uas.report
```

```
-----  
-----  
VNF ID          | UAS Node | Status  | Error Info, if any  
-----  
-----  
172.21.201.122  | autovnf  | XXX     | AutoVNF Cluster FAILED : Node: 172.16.180.12,  
Status: error, Role: NA  
172.21.201.122  | vnf-em   | :-)     |  
172.21.201.122  | esc      | :-)     |  
-----  
-----
```

2. UAS集群的預期狀態將如圖所示，即所有三個UAS都處於活動狀態。

```
[stack@pod1-ospd ~]# ssh ubuntu@10.1.1.1  
password:  
  
ubuntu@autovnf1-uas:~$ ncs_cli -u admin -C
```

```
autovnf1-uas-0#show uas  
uas version 1.0.1-1  
uas state ha-active  
uas ha-vip 172.16.181.101  
INSTANCE IP STATE ROLE  
-----  
172.16.180.3 alive CONFD-MASTER  
172.16.180.7 alive CONFD-SLAVE  
172.16.180.12 alive NA
```

嘗試連線到UAS時無法連線到Confd伺服器

1.在某些情況下，您無法連線到confd服務器。

```
ubuntu@autovnf1-uas-0:/opt/cisco/usp/uas/manager$ confd_cli -u admin -C  
Failed to connect to server
```

2.檢查uas-confd進程的狀態。

```
ubuntu@autovnf1-uas-0:/opt/cisco/usp/uas/manager$ sudo initctl status uas-confd  
uas-confd stop/waiting
```

3.如果confd伺服器未運行，請重新啟動該服務。

```
ubuntu@autovnf1-uas-0:/opt/cisco/usp/uas/manager$ sudo initctl start uas-confd  
uas-confd start/running, process 7970  
ubuntu@autovnf1-uas-0:/opt/cisco/usp/uas/manager$ confd_cli -u admin -C  
Welcome to the Confd CLI  
admin connected from 172.16.180.9 using ssh on autovnf1-uas-0
```

將UAS從錯誤狀態中恢復

1.如果群集中的一個AutoVNF出現故障，UAS群集將顯示一個UAS處於錯誤狀態。

```
[stack@pod1-ospd ~]# ssh ubuntu@10.1.1.1
password:
```

```
ubuntu@autovnf1-uas:~$ ncs_cli -u admin -C
```

```
autovnf1-uas-0#show uas
uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.16.181.101
INSTANCE IP STATE ROLE
-----
172.16.180.3 alive CONFID-MASTER
172.16.180.7 alive CONFID-SLAVE
172.16.180.12 alive error
```

2.將OSPD伺服器中的coreerrc檔案(VNF的rc檔案)從/home/stack複製到AutoDeploy並將其來源。

3.使用uas-check.py指令碼檢查UAS/AutoVNF的狀態。autovnf1是AutoVNF名稱。

```
ubuntu@auto-deploy-iso-590-uas-0:~$ /opt/cisco/usp/apps/auto-it/scripts/uas-check.py auto-vnf
autovnf1
2017-11-17 14:52:20,186 - INFO: Check of AutoVNF cluster started
2017-11-17 14:52:22,172 - INFO: Found 2 AutoVNF instance(s), 3 expected
2017-11-17 14:52:22,172 - INFO: Instance 'autovnf1-uas-2' is missing
2017-11-17 14:52:22,172 - INFO: Check completed, AutoVNF cluster has recoverable errors
```

4.使用uas-check.py指令碼和add-fix關鍵字恢復UAS。

```
ubuntu@auto-deploy-iso-590-uas-0:~$ /opt/cisco/usp/apps/auto-it/scripts/uas-check.py auto-vnf
autovnf1 --fix
2017-11-17 14:52:27,493 - INFO: Check of AutoVNF cluster started
2017-11-17 14:52:29,215 - INFO: Found 2 AutoVNF instance(s), 3 expected
2017-11-17 14:52:29,215 - INFO: Instance 'autovnf1-uas-2' is missing
2017-11-17 14:52:29,215 - INFO: Check completed, AutoVNF cluster has recoverable errors
2017-11-17 14:52:29,386 - INFO: Creating instance 'autovnf1-uas-2' and attaching volume
'autovnf1-uas-vol-2'
2017-11-17 14:52:47,600 - INFO: Created instance 'autovnf1-uas-2'
```

5.您將看到新建立的UAS處於活動狀態，並且是群集的一部分。

```
autovnf1-uas-0#show uas
uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.16.181.101
INSTANCE IP STATE ROLE
-----
172.16.180.3 alive CONFID-MASTER
172.16.180.7 alive CONFID-SLAVE
172.16.180.13 alive NA
```

案例2.所有三個UAS(AutoVNF)均處於錯誤狀態

1. Ultra-M Manager執行Ultra-M節點的運行狀況檢查。

```
[stack@pod1-ospd ultram-health]$ more ultram_health_uas.report
```

```
-----
-----
```

```

VNF ID          | UAS Node | Status | Error Info, if any
-----
-----
172.21.201.122 | autovnf | XXX    | AutoVNF Cluster FAILED : Node: 172.16.180.12, Status:
error, Role: NA,Node: 172.16.180.9, Status: error, Role: NA,Node: 172.16.180.10, Status: error,
Role: NA
172.21.201.122 | vnf-em  | :- )   |
172.21.201.122 | esc     | :- )   |
-----
-----

```

2.如輸出中所示，Ultra-M manager報告AutoVNF有故障，它顯示群集的所有三個UAS都處於「錯誤」狀態。

使用uas-check.py指令碼檢查UAS運行狀況

1.登入到「自動部署」，然後檢查是否可以訪問AutoVNF UAS並獲取狀態。

```

ubuntu@auto-deploy-iso-590-uas-0:~$ /opt/cisco/usp/apps/auto-it/scripts$ ./uas-check.py auto-vnf
autovnf1 --os-tenant-name core

```

```

2017-12-05 11:41:09,834 - INFO: Check of AutoVNF cluster started
2017-12-05 11:41:11,342 - INFO: Found 3 ACTIVE AutoVNF instances
2017-12-05 11:41:11,343 - INFO: Check completed, AutoVNF cluster is fine

```

2.從自動部署、安全殼層(SSH)到AutoVNF節點並進入confd模式。使用show uas檢查狀態。

```

ubuntu@auto-deploy-iso-590-uas-0:~$ ssh ubuntu@172.16.180.9
password:
autovnf1-uas-1#show uas
uas version 1.0.1-1
uas state ha-active
uas ha-vip 172.16.181.101
INSTANCE IP      STATE  ROLE
-----

```

```

172.16.180.9    error  NA
172.16.180.10  error  NA
172.16.180.12  error  NA

```

3.建議檢查所有三個UAS節點中的狀態。

檢查OpenStack級別上的VM狀態

檢查AutoVNF VM在新星清單中的狀態。如果需要，執行nova start以啟動關閉VM。

```

[stack@pod1-ospd ultram-health]$ nova list | grep autovnf
| 83870eed-b4e9-47b3-976d-cc3eddecf866 | autovnf1-uas-
0                                     | ACTIVE | -           | Running    |
orchestr=172.16.180.12;

```

mgmt=172.16.181.6

```
| 201d9ce5-538c-42f7-a46c-fc8cdefleabf | autovnf1-uas-  
1 | ACTIVE | - | Running |  
orchestr=172.16.180.10;  
mgmt=172.16.181.5
```

```
| 6c6d25cd-21b6-42b9-87ff-286220faa2ff | autovnf1-uas-  
2 | ACTIVE | - | Running |  
orchestr=172.16.180.9; mgmt=172.16.181.13
```

檢查Zookeeper檢視

1.檢查縮放管理器的狀態，以驗證模式為引線。

```
ubuntu@autovnf1-uas-0:/var/log/upstart$  
/opt/cisco/usp/packages/zookeeper/current/bin/zkServer.sh status  
ZooKeeper JMX enabled by default  
Using config: /opt/cisco/usp/packages/zookeeper/current/bin/./conf/zoo.cfg  
Mode: leader
```

2.動物飼養員一般應處於開啟狀態。

AutoVNF故障排除 — 流程和任務

1.確定節點錯誤狀態的原因。要運行AutoVNF，必須運行一組進程，如下所示：

AutoVNF

uws-ae

uas-confd

cluster_manager

uas_manager

```
ubuntu@autovnf1-uas-0:~$ sudo initctl list | grep uas
```

```
uas-confd stop/waiting ==> this is not good, the uas-confd process is not running
```

```
uas_manager start/running, process 2143
```

```
root@autovnf1-uas-1:/home/ubuntu# sudo initctl list
```

....

```
uas-confd start/running, process 1780
```

....

```
autovnf start/running, process 1908
```

....

....

```
uws-ae start/running, process 1909
```

....

....

```
cluster_manager start/running, process 1827
```

....

.....

```
uas_manager start/running, process 1697
```

.....
.....

2. 驗證這些python進程是否正在運行：

uas_manager.py

cluster_manager.py

usp_autovnf.py

```
root@autovnf1-uas-1:/home/ubuntu# ps -aef | grep pyth
root      1819   1697   0 Jun13 ?          00:00:50 python /opt/cisco/usp/uas/manager/uas_manager.py
root      1858   1827   0 Jun13 ?          00:09:21 python
/opt/cisco/usp/uas/manager/cluster_manager.py
root      1908     1   0 Jun13 ?          00:01:00 python /opt/cisco/usp/uas/autovnf/usp_autovnf.py
root      25662  24750   0 13:16 pts/7    00:00:00 grep --color=auto pyth
```

3. 如果任何預期進程未處於**啟動/運行**狀態，請重新啟動該進程並檢查狀態。如果它仍然顯示為「錯誤」狀態，則按照下一節中提到的步驟操作，以解決此問題。

修復處於錯誤狀態的多個UAS

1. **nova** — 從OSPD硬重新啟動<VM名稱>，在進入下一個UAS之前給此虛擬機器恢復一些時間。在所有UAS虛擬機器上執行該操作。

或

2. 登入到每個UAS並使用**sudo**重新引導。等待恢復，然後繼續到其他UAS虛擬機器。

對於事務日誌，請檢查：

```
/var/log/upstart/autovnf.log
```

```
show logs xxx | display xml
```

這將解決問題並從「錯誤」狀態中恢復UAS。

1. 使用ultrum_health_check報告進行驗證。

```
[stack@pod1-ospd ultram-health]$ more ultram_health_uas.report
```

```
-----
-----
```

VNF ID	UAS Node	Status	Error Info, if any
172.21.201.122	autovnf	: -)	
172.21.201.122	vnf-em	: -)	
172.21.201.122	esc	: -)	

```
-----
-----
```
