



# VNF展開に関するトラブルシューティング

- 
- [概要 \(1 ページ\)](#)
- [トラブルシューティング用のログ \(1 ページ\)](#)
- [展開された VNF VM が VIM 関連エラーで失敗する \(2 ページ\)](#)
- [展開された VNF VM が LCM で失敗する \(3 ページ\)](#)
- [ロールアクセスの問題により、展開された VNF VM が失敗する \(ESC リリース 3.1 以降\) \(4 ページ\)](#)
- [VNF VM は展開されたが、ブートループに入る \(5 ページ\)](#)
- [VNF VM は展開されたが、稼働状態にならない \(7 ページ\)](#)
- [VNF リカバリの失敗 \(8 ページ\)](#)
- [再起動の失敗により VNF VM リカバリが失敗する \(8 ページ\)](#)
- [エラー状態の VNF VM の回復 \(9 ページ\)](#)
- [非アクティブ状態の VNF サービス \(展開\) の回復 \(13 ページ\)](#)
- [サービスステータスが正しくないために VNF リカバリが拒否される \(13 ページ\)](#)
- [VIM コネクタの問題により VNF 操作が拒否される \(15 ページ\)](#)

## 概要

このガイドでは、ESCによって展開および管理されているVNF間の問題をトラブルシューティングする方法について、順を追って説明します。

## トラブルシューティング用のログ

VNF 展開の ESC トラブルシューティングを実行する前の最初の手順は、ログの収集および確認です。ESC ログを収集するには、次の手順を実行します。

ESC リリース 2.3.2 の場合 :

```
sudo /opt/cisco/esc/esc-scripts/collect_esc_log.sh
```

ESC リリース 3.0 以降の場合 :

```
sudo escadm log collect
```

有用なエラーメッセージを含む重要なログがいくつかあります。

- YangESC ログには、着信要求と通知が含まれています。  
/var/log/esc/yangesc.log
- ESCManager ログには、ESC 処理の詳細が含まれています。  
/var/log/esc/escmanager.log
- VimManager ログには、VimManager 処理の詳細が含まれています。  
/var/log/esc/vimmanager/vimmanager.log
- Vim\_VimManager ログには、VimManager と VIM 間の未処理の要求と応答が含まれています。  
/var/log/esc/vimmanager/vim\_vimmanager.log
- Mona ログには、処理の詳細とスクリプト実行のモニタリング情報が含まれています。  
/var/log/esc/mona/mona.log

## 展開された VNF VM が VIM 関連エラーで失敗する

### 問題

ステータスコードが 200 以外の VM\_DEPLOYED 通知を（NETCONF、REST、ポータルを介して）受け取った場合、エラーが原因で展開が失敗したことを意味します。

/var/log/esc/yangesc.log に通知が見つかりました。

```
02:19:25,758 23-Jan-2018 WARN ===== SEND NOTIFICATION STARTS =====
02:19:25,758 23-Jan-2018 WARN Type: VM_DEPLOYED
02:19:25,758 23-Jan-2018 WARN Status: FAILURE
02:19:25,758 23-Jan-2018 WARN Status Code: 500
02:19:25,759 23-Jan-2018 WARN Status Msg: VIM Driver: Exception while creating VM: Error
  creating VM from template, the host [10.67.103.255] does not exist
02:19:25,759 23-Jan-2018 WARN Tenant: admin
02:19:25,759 23-Jan-2018 WARN Deployment ID: 169384d7-c67b-40a4-bcaa-dd3294305ba3
02:19:25,759 23-Jan-2018 WARN Deployment name:
Vmware-NetConf-Intra-Affinity-Anti-Affinity-With-InvalidCluster-InvalidHost
02:19:25,759 23-Jan-2018 WARN VM group name:
Group2-uLinux-Intra-Anti-Affinity-With-InvalidHost
02:19:25,759 23-Jan-2018 WARN User configs: 1
02:19:25,759 23-Jan-2018 WARN VM Name:
Vmware-NetConf-I_Group2_0_65aa6ca8-3b53-4eb3-a39f-a3f12394a190
02:19:25,759 23-Jan-2018 WARN Host ID:
02:19:25,759 23-Jan-2018 WARN Host Name:
02:19:25,760 23-Jan-2018 WARN ===== SEND NOTIFICATION ENDS =====
```

### ソリューション

失敗の理由は、ステータスメッセージ自体に記載されています。前述の例は、展開対象のホストが存在しないことを示しています。次に、失敗した VM 展開の別の例を示します。

/var/log/esc/yangesc.log に通知が見つかりました。

```
07:20:56,164 25-Jan-2018 WARN Status Msg: Failed to create VM ports for instance :
jenkins-ErrHandl_ErrorG_2_cc0f8c28-8900-4977-90d9-b9f996c8ca71. Create port operation
failed: Exception during processing:
com.cisco.esc.vimmanager.exceptions.CreatePortException: Create port failed:
ClientResponseException{message=No more IP addresses available on network
0b7965b4-c604-444c-8cbb-7c2399e912d4., status=409, status-code=CONFLICT}.
```

前述の例は、VIMからの直接応答（エラーメッセージとステータスコードあり）を含む展開失敗メッセージを示しています。このようなVIM関連の問題など、ロールアクセスの問題が原因で展開が失敗した場合は、VIMインスタンスで必要なアクションを実行するか、適切な構成でESC展開データモデルを調整します。一般的なVIM関連の問題を次に示します。

#### 1. クォータ不足エラー

1. 該当テナント/プロジェクト/ユーザーの下でESCを介して問題のリソースを削除します。または、
2. テナント/プロジェクト/ユーザーごとにVIMのリソース制限を設定します。

#### 2. 使用中エラー

1. リソース名または設定を変更します。あるいは、VIMで許可されない制限があり、設定可能な場合があります。
2. 問題のリソースを削除します。

## 展開された VNF VM が LCM で失敗する

### 問題

VNF展開データモデルにLCMアクション（ステージングスクリプトなど）が含まれる場合、アクションが完了しなかったために展開が失敗した可能性があります。この場合、`/var/log/esc/escmanager.log`に次のエラーメッセージが表示されます。

```
22:12:11,912 25-Jan-2018
VM_STATE_MACHINE-ab-auto-test-vnf_ab-aut_0_31ebad33-e12f-4772-a89c-3bdc239acf69 ERROR
[StateMachineCloudUtils.java:setupPersonalities():1081]
[tid=ffffae7af-a321-4ea5-albc-3b30c903f3a5]
com.cisco.esc.datamodel.exceptions.ESCEException: Action [GEN_VPC_ISO] failed
    at
com.cisco.esc.statemachines.utils.StateMachineCloudUtils.setupPersonalities(StateMachineCloudUtils.java:1069)
```

### 説明

ステージングスクリプトの失敗の詳細を確認するには、`/var/log/esc/mona/mona.log`ログで次のようなエントリを探します。

`/var/log/esc/mona/mona.log`

```
2018-01-25 19:34:45.751 [http-nio-127.0.0.1-8090-exec-5] Script:
[/opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh] execution in progress
2018-01-25 19:34:45.751 [http-nio-127.0.0.1-8090-exec-5] Use the original script path
and skip downloading: no protocol: /opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh
2018-01-25 19:49:45.772 [http-nio-127.0.0.1-8090-exec-5] Script execution failed, timer
expired for script: /opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh
```

## ロールアクセスの問題により、展開された VNF VM が失敗する (ESC リリース 3.1 以降)

```
2018-01-25 19:49:45.805 [http-nio-127.0.0.1-8090-exec-5] Script execution failed
com.cisco.esc.mona.exceptions.ActionExecutionException: Script execution failed, timer
expired for script:/opt/cisco/esc/esc-scripts/esc_volume_em_staging.sh
```

### ソリューション

一般的なエラーには、権限の問題や、スクリプト実行のタイムアウトなどがあります。ESC VM でスクリプトのリハーサルを実行して、スクリプトが機能することを確認します。

## ロールアクセスの問題により、展開された VNF VM が失敗する (ESC リリース 3.1 以降)

### 問題

管理者以外のユーザーとして OpenStack に VNF を展開すると、展開時に次のようなロールアクセスエラーが発生する場合があります。

```
02:19:25,758 23-Jan-2018 WARN ===== SEND NOTIFICATION STARTS =====
02:19:25,758 23-Jan-2018 WARN Type: VM_DEPLOYED
02:19:25,758 23-Jan-2018 WARN Status: FAILURE
02:19:25,758 23-Jan-2018 WARN Status Code: 500
02:19:25,759 23-Jan-2018 WARN Status Msg: VIM Driver: Exception while creating VM:
{"message": "You are not authorized to perform the requested action:
identity:create_project", "code": 403, "title": "Forbidden"}
```

```
02:19:25,759 23-Jan-2018 WARN Tenant: admin
02:19:25,759 23-Jan-2018 WARN Deployment ID: 169384d7-c67b-40a4-bcaa-dd3294305ba3
02:19:25,759 23-Jan-2018 WARN Deployment name:
Vmware-NetConf-Intra-Affinity-Anti-Affinity-With-InvalidCluster-InvalidHost
02:19:25,759 23-Jan-2018 WARN VM group name:
Group2-uLinux-Intra-Anti-Affinity-With-InvalidHost
02:19:25,759 23-Jan-2018 WARN User configs: 1
02:19:25,759 23-Jan-2018 WARN VM Name:
Vmware-NetConf-I_Group2_0_65aa6ca8-3b53-4eb3-a39f-a3f12394a190
02:19:25,759 23-Jan-2018 WARN Host ID:
02:19:25,759 23-Jan-2018 WARN Host Name:
02:19:25,760 23-Jan-2018 WARN ===== SEND NOTIFICATION ENDS =====
```

### ソリューション

ESC リリース 3.1 以降では、Neutron で 2 つの権限を付与する必要があります。

```
create_port:fixed_ips
create_port:mac_address
```

1. ESC の OpenStack に新しいロールを作成します。OpenStack Horizon ([アイデンティティ (Identity)] → [ロール (Roles)]) に移動し、**vnfm** または他の任意の名前を指定して新しいロールを作成します。
2. OpenStack Horizon の ESC によって管理されるプロジェクトに **vnfm** ロールを持つユーザーを割り当てます ([アイデンティティ (Identity)] → [プロジェクト (Projects)])。[メンバーの管理 (Manage members)] をクリックし、ESC のユーザーが **vnfm** ロールを持っていることを確認します。



```

        "boot_index" : 0,
        "delete_on_termination" : true
    } ]
}
}

```

## ソリューション

base64 で符号化されたパーソナリティコンテンツの文字列値を復号化します。

```

hostname csr
!
platform console serial
!
ip subnet-zero
no ip domain-lookup
ip domain name cisco.com
!
enable password cisco123
username admin password cisco123
username admin privilege 15
!
interface GigabitEthernet1
  description management network
  ip address dhcp
  no shut
!
interface GigabitEthernet2
  description service network
  ip address dhcp
  no shut
!
interface GigabitEthernet3
  description service network
  ip address dhcp
  no shut
!
crypto key generate rsa modulus 1024
ip ssh version 2
ip ssh authentication-retries 5
ip scp server enable
file prompt quiet
!
line con 0
  stopbits 1
line vty 0 4
  login local
  privilege level 15
  transport input ssh telnet
  transport output ssh telnet
!
snmp-server community public RO
!
end

```

コンテンツに正しいデイズロ設定が含まれているか確認します。デイズロ設定がボリュームを介して渡される場合は、そのボリュームを VNF から切り離し、別の VM に接続して内容を確認します。

VMware の場合、デイズロ設定が OVF 設定を介して渡される場合は、vCenter から確認します。

1. [VM設定 (VM settings) ]を開きます。

2. [オプション (Options)] で、[OVF設定 (OVF Settings)] を選択します。
3. [OVF環境 (OVF Enviroment)] の下にある [表示 (View)] をクリックします。

デイレゾ設定が CDROM を介して渡される場合、CDROM にマウントされた ISO ファイルから確認できます。

特定のデータストアから ISO ファイルをダウンロードし、ISO ファイルをローカルにマウントしてその内容を確認します。

## VNF VM は展開されたが、稼働状態にならない

### 問題

VNF が正常に展開されたが、VM\_ALIVE 通知を受信しなかった場合、ESC は新しく展開された VNF に到達できなかったと推測されます。主な原因はネットワークの問題です。VNF 展開データモデルの KPI セクションを確認します。

```
<kpi_data>
  <kpi>
    <event_name>VM_ALIVE</event_name>
    <metric_value>1</metric_value>
    <metric_cond>GT</metric_cond>
    <metric_type>UINT32</metric_type>
    <metric_occurrences_true>1</metric_occurrences_true>
    <metric_occurrences_false>30</metric_occurrences_false>
    <metric_collector>
      <type>ICMPPing</type>
      <nicid>2</nicid>
      <poll_frequency>10</poll_frequency>
      <polling_unit>seconds</polling_unit>
      <continuous_alarm>>false</continuous_alarm>
    </metric_collector>
  </kpi>
</kpi_data>
```

### ソリューション

VNF が稼働しているか確認するには、次に示されている特定の IP を使用して、ESC VM から VNF に ICMP ping を実行します。

```
<nicid>2</nicid>
```

ここで、nicid 2 は、ESC が ping しようとしている nicid が 2 のインターフェイスの IP を指しており、以下の内容を指しています。

```
<interface>
  <nicid>2</nicid>
  <network>NVPGW100-UAS-uas-orchestration</network>
  <allowed_address_pairs>
    <address>
      <ip_address>172.168.11.0</ip_address>
      <netmask>255.255.255.0</netmask>
    </address>
  </allowed_address_pairs>
</interface>
```

ここで 172.168.11.0 は IP です。インターフェイスが ESC と同じネットワークを共有していることを確認してください。前述の例では、ネットワークは NVPGW100-UAS-uas-orchestration です。ping が失敗した場合、ゲートウェイまたはサブネットで使用可能な別の IP に ping を実行して、問題がネットワークにあるのか確認できます。

## VNF リカバリの失敗

一般的なりカバリの問題を次に示します。

1. リカバリ動作が期待どおりに動作しない。再起動に失敗後、ESC が再展開を試みていません。
  1. XML ファイルのリカバリポリシーが REBOOT\_THEN\_REDEPLOY に設定されていて、「再起動のみ」に設定されていないことを確認します。リカバリマニュアルを読み、リカバリオプションと期待される動作を理解します。
2. ESC がリカバリを 1 回のみ試行しているか、何度も試行しています。
  1. 設定パラメータ「VM\_RECOVERY\_RETRIES\_MAX」を再確認します。デフォルト値は 3 回です。この値を確認するには、ESC VM 内で REST 呼び出しを実行します。
 

```
curl -H "accept: Application/json"
http://127.0.0.1:8080/ESCManager/v0/config/default/VM_RECOVERY_RETRIES_MAX |
python -mjson.tool
```
  2. 正しく設定されている場合は、ESC がリカバリ時に正常であり、スイッチオーバーが発生していないことを確認します。また、2 番目の ESC VM でリカバリの試行を継続している可能性があります。

## 再起動の失敗により VNF VM リカバリが失敗する

### 問題

VM の再起動に失敗したため、VNF VM のリカバリに失敗しました。失敗するかどうかは、VM リカバリポリシーの定義によって異なります。

```
<recovery_policy>
  <recovery_type>AUTO</recovery_type>
  <action_on_recovery>REBOOT_ONLY</action_on_recovery>
  <max_retries>3</max_retries>
</recovery_policy>
```

### 説明

ESC は、エラー状態の RECOVERY\_COMPLETED イベントを受信する前に、未成功状態で VNF VM の再起動を 3 回試行します。再起動操作は、他の 2 つのシステム全体の設定パラメータにも依存します。

```
VM_STATUS_POLLING_VM_OPERATION_RETRIES
VM_STATUS_POLLING_WAIT_TIME_SEC
```



ESC は、VM の再起動を VIM に要求後、VM ステータスのポーリングを継続します。VM\_STATUS\_POLLING\_VM\_OPERATION\_RETRIES では、ESC のポーリング試行回数を定義し、VM\_STATUS\_POLLING\_WAIT\_TIME\_SEC では、ポーリング間での ESC の待機時間を定義します。以下はそれぞれのデフォルト値です。

```
VM_STATUS_POLLING_VM_OPERATION_RETRIES=10
VM_STATUS_POLLING_WAIT_TIME_SEC=5
```

### ソリューション

OpenStack で VNF VM が再起動状態からアクティブ状態に移行するのに 50 秒以上かかる場合は、ESC REST API を使用して VM\_STATUS\_POLLING\_WAIT\_TIME\_SEC をより大きな数値に変更します。

```
curl -X PUT -H "accept:application/json"
http://localhost:8080/ESCManager/v0/config/openstack/vm_status_polling_wait_time_sec/20
-k | python -mjson.tool
```

成功の応答を受信したら、VM の手動リカバリを再度実行します。

## エラー状態の VNF VM の回復

VNF VM が ESC でエラー状態になっている場合、エラー状態の原因となった外部の問題（VIM の問題など）が解決した場合に稼働状態に戻すための 2 つのオプションがあります。次の 2 つのオプションのいずれかを実行する前に、同じ展開環境で実行中の操作がないことを確認してください。操作は /var/log/esc/yangesc.log で確認します。完了通知（成功または失敗）のない、以前に開始されたアクションを探します。進行中の操作が見つかった場合は、その操作が完了するのを待ってから、次のアクションを実行します。

### VNF VM の手動リカバリ

makecall ディレクトリで、次のコマンドを実行します。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO {ESC generated VM Name}
```

### VM モニタリングの手動設定解除または設定（ESC リリース 3.1 以降）

次のコマンドを実行して、モニタリングの設定を解除します。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action DISABLE_MONITOR {ESC generated VM Name}
```

次に、再度有効にします。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action ENABLE_MONITOR {ESC generated VM Name}
```

### サービス更新による VNF の削除または追加

#### スクリプトを使用したデータモデルの準備

ESC VM でスクリプトを実行してください。スクリプトにより 2 つのデータモデル xml ファイルが生成されます。1 つは VM グループを削除するためのファイルで、もう 1 つは VM グループを追加し直すためのファイルです。

```
[admin@abc-test-232 ~]$ ./genVMGroupDeletionDM.py -h
usage: genVMGroupDeletionDM.py [-h] vm_group_name [vm_group_name ...]

*****
Utility tool for generating VM group removing datamodel for ESC
Check the following wiki for details
https://confluence-eng-sjc1.cisco.com/conf/display/ESCWIKI/How+to+Use+Service+Update+to+Remove+a+VM+Group

positional arguments:
  vm_group_name <Required> VM group name(s) separate by space

optional arguments:
  -h, --help      show this help message and exit
```

### 例

```
[admin@abc-test-232 ~]$ ./genVMGroupDeletionDM.py g1 g2

Datamodel is generated:
[/home/admin/delete_g1_g2.xml]
[/home/admin/add_g1_g2.xml]
** Use on your own risk! **

[admin@abc-test-232 ~]$
```

### 手動でのデータモデルの準備

1. 現在の ESC データモデルを取得します。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli get esc_datamodel/tenants > {file name}
```

2. ステップ 1 で取得した元のファイル（および <xml> タブの前にあるすべての CLI 出力）から余分なラッパー <data> および <rpc-reply> を削除します。最終結果は次のようになります。

#### ステップ 2 後のデータモデルの例

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
```



- (注) データモデルに複数の展開が含まれている場合は、データモデルの編集時に他の展開をそのまま維持してください。テキストの書式設定は変更しないでください。他の展開セクションをデータモデルから完全に削除します。削除すると、サービス更新の発生時に、削除された展開は変更されません。たとえば、削除する VM グループが c3 の場合、データモデル編集時に EM の展開部分をデータモデルから削除できます。

3. 次に、VM グループ c3 を削除する例を示します。まず、VM グループ c3 が含まれる <policies> の下に定義済みの <placement\_group> または <placement> policy(ies) が存在するか確認します。ポリシーをマークして削除します。次の例には、8 つの配置ポリシーがあります。

```
...
      <placement>
        <target_vm_group_ref>c3</target_vm_group_ref>
        <type>anti_affinity</type>
```

```

    <enforcement>strict</enforcement>
    <vm_group_ref>c1</vm_group_ref>
    <vm_group_ref>s2</vm_group_ref>
  </placement>
</placement>
  <target_vm_group_ref>s10</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
  <vm_group_ref>s4</vm_group_ref>
  <vm_group_ref>s5</vm_group_ref>
  <vm_group_ref>s6</vm_group_ref>
  <vm_group_ref>s7</vm_group_ref>
  <vm_group_ref>s8</vm_group_ref>
  <vm_group_ref>s9</vm_group_ref>
</placement>
...
<placement>
  <target_vm_group_ref>s4</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
</placement>
<placement>
  <target_vm_group_ref>s5</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref nc:operation='delete'>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
  <vm_group_ref>s4</vm_group_ref>
</placement>
<placement>
  <target_vm_group_ref>s6</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
  <vm_group_ref>s4</vm_group_ref>
  <vm_group_ref>s5</vm_group_ref>
</placement>
<placement>
  <target_vm_group_ref>s7</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
  <vm_group_ref>s4</vm_group_ref>
  <vm_group_ref>s5</vm_group_ref>
  <vm_group_ref>s6</vm_group_ref>
</placement>
<placement>
  <target_vm_group_ref>s8</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>

```

```

    <vm_group_ref>s4</vm_group_ref>
    <vm_group_ref>s5</vm_group_ref>
    <vm_group_ref>s6</vm_group_ref>
    <vm_group_ref>s7</vm_group_ref>
  </placement>
  <placement>
    <target_vm_group_ref>s9</target_vm_group_ref>
    <type>anti_affinity</type>
    <enforcement>strict</enforcement>
    <vm_group_ref>c1</vm_group_ref>
    <vm_group_ref>c3</vm_group_ref>
    <vm_group_ref>s2</vm_group_ref>
    <vm_group_ref>s4</vm_group_ref>
    <vm_group_ref>s5</vm_group_ref>
    <vm_group_ref>s6</vm_group_ref>
    <vm_group_ref>s7</vm_group_ref>
    <vm_group_ref>s8</vm_group_ref>
  </placement>

```

`target_vm_group` が `c3` の場合、XML 要素に属性 `nc:operation='delete'` を追加して、配置ポリシー全体を削除します。

.\

```

<placement nc:operation='delete'>
  <target_vm_group_ref>c3</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
</placement>

```

`vm_group_ref` が `c3` の場合、`vm_group_ref` エントリ自体を削除し、他の関係はそのままにします。

```

<placement>
  <target_vm_group_ref>s10</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c1</vm_group_ref>
  <vm_group_ref nc:operation='delete'>c3</vm_group_ref>
  <vm_group_ref>s2</vm_group_ref>
  <vm_group_ref>s4</vm_group_ref>
  <vm_group_ref>s5</vm_group_ref>
  <vm_group_ref>s6</vm_group_ref>
  <vm_group_ref>s7</vm_group_ref>
  <vm_group_ref>s8</vm_group_ref>
  <vm_group_ref>s9</vm_group_ref>
</placement>

```

`vm_group_ref` 要素が 1 つだけある配置ポリシーの場合、`vm_group_ref` が `c3` であるか、`target_vm_group` が `c3` である場合は、ポリシー全体を削除します。これは、`c3` が削除されると、このポリシーに意味がなくなるためです。

```

<placement nc:operation='delete'>
  <target_vm_group_ref>c11</target_vm_group_ref>
  <type>anti_affinity</type>
  <enforcement>strict</enforcement>
  <vm_group_ref>c3</vm_group_ref>
</placement>

```

- 最後のステップでは、属性 `nc:operation='delete'` を XML 要素に追加して、VM グループ自体を削除対象としてマークします。

```
<vm_group nc:operation='delete'>
  <name>c3</name>
  <flavor>SFPCF101-DEPLOYMENT-control-function</flavor>
  <bootup_time>1800</bootup_time>
  <recovery_wait_time>1</recovery_wait_time>
  ...
```

5. 同じ VM グループを再び追加するためにデータモデルを準備するには、削除データモデルを取得し、すべての `nc:operation='delete'` をすべての場所から削除します。

2つのデータモデルファイルの準備ができれば、次のコマンドを使用してサービス更新を実行します。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli edit-config {deleting datamodel file}
```

サービス更新が完了するまで待ちます。次に、VNF を追加し直します。



**警告** VNF の追加し直す前に、サービスは稼働状態になっている必要があります。稼働状態になっていない場合は、該当 VM をリカバリしてください。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli edit-config {adding datamodel file}
```

## 非アクティブ状態の VNF サービス（展開）の回復

ESC リリース 3.1 以降のリリースでは、VM の停止操作が失敗すると、サービスが非アクティブ状態でスタックし、リカバリがトリガーされない場合があります。1つの VM はエラー状態ですが、サービスは非アクティブ状態になります。ENABLE MONITOR を使用して、VM とサービスを稼働状態またはエラー状態に戻すことができます。

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action ENABLE_MONITOR {VM Name}
```

この操作により、VM のモニタリングを有効にできます。VM が VIM で実行されている場合、VM ALIVE イベントは VM ステートマシンに戻す必要があります。VM は最終的に稼働状態に遷移します。VM が VIM で実行されていない場合、タイマーが期限切れになり、リカバリ手順で VM を戻すことができます。その間、サービスはアクティブまたはエラー状態に遷移しません。

## サービスステータスが正しくないために VNF リカバリが拒否される

### 問題

VNF VM の手動リカバリを実行すると、誤ったサービスステータス（ESC 3.1 以降のリリース）が原因でリクエストが拒否されます（以下を参照）。

```
$ /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO vm-name
```

```
Recovery VM Action
```

## サービスステータスが正しくないために VNF リカバリが拒否される

```

/opt/cisco/esc/confd/bin/netconf-console --port=830 --host=127.0.0.1 --user=admin
--privKeyFile=/home/admin/.ssh/confd_id_dsa --privKeyType=dsa
--rpc=/tmp/esc_nc_cli.L1WdqyIE7r
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <rpc-error>
    <error-type>application</error-type>
    <error-tag>operation-failed</error-tag>
    <error-severity>error</error-severity>
    <error-path xmlns:esc="http://www.cisco.com/esc/esc"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
      /nc:rpc/esc:recoveryVmAction
    </error-path>
    <error-message xml:lang="en">Exception from action callback: Recovery VM Operation:
recovery_do is not applicable since the service is in [SERVICE_INERT_STATE]
state.</error-message>
    <error-info>
      <bad-element>recoveryVmAction</bad-element>
    </error-info>
  </rpc-error>
</rpc-reply>

```

### ソリューション

この時点で、**opdata** をチェックして、サービスの状態と VM の状態を調べます。

```

$ /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli get esc_datamodel/opdata

<state_machine>
  <state>SERVICE_INERT_STATE</state>
  <vm_state_machines>
    <vm_state_machine>
      <vm_name>depz_g1_0_b6d19896-bc3b-400a-ad50-6d84c522067d</vm_name>
      <state>VM_MONITOR_UNSET_STATE</state>
    </vm_state_machine>
    <vm_state_machine>
      <vm_name>depz_g1_1_f8445a8a-29ba-457d-9224-c46eaaa97f72</vm_name>
      <state>VM_ALIVE_STATE</state>
    </vm_state_machine>
  </vm_state_machines>
</state_machine>

```

未設定監視状態の VM の監視を有効にします。

```

$ /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action ENABLE_MONITOR vm-name

```

しばらくしてから、**opdata** を再度確認します。サービスはアクティブ状態またはエラー状態に遷移する必要があります。

```

$ /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli get esc_datamodel/opdata

<state_machine>
  <state>SERVICE_ACTIVE_STATE</state>
  <vm_state_machines>
    <vm_state_machine>
      <vm_name>depz_g1_0_b6d19896-bc3b-400a-ad50-6d84c522067d</vm_name>
      <state>VM_ALIVE_STATE</state>
    </vm_state_machine>
    <vm_state_machine>
      <vm_name>depz_g1_1_f8445a8a-29ba-457d-9224-c46eaaa97f72</vm_name>
      <state>VM_ALIVE_STATE</state>
    </vm_state_machine>
  </vm_state_machines>
</state_machine>

```

エラー状態の VM を手動でリカバリします。

```
$ /opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli recovery-vm-action DO vm-name
```

## VIM コネクタの問題により VNF 操作が拒否される

### 問題

VNF 操作（展開、リカバリなど）が次の理由で拒否される：

```
Default VIM Connector is not set up, or is unreachable. Please check your VIM Connector credentials and VIM status.
```

### 説明

ESC が複数 VIM 用に設定されている場合は、少なくとも 1 つの VIM コネクタが「デフォルト」としてマークされていることを確認してください。それ以外の場合は、ESC VIM コネクタのステータスを確認してください。

```
[admin@leishi-test ~]$ escadm vim show
[
  {
    "status": "CONNECTION_SUCCESSFUL",
    "status_message": "Successfully connected to VIM",
    "type": "OPENSTACK",
    "id": "default_openstack_vim",
    "properties": {
      "property": [
        {
          "name": "os_project_domain_name",
          "value": "default"
        },
        {
          "name": "os_auth_url",
          "value": "http://10.85.103.143:35357/v3"
        },
        {
          "name": "os_project_name",
          "value": "admin"
        }
      ]
    }
  }
]
{
  "user": [
    {
      "credentials": {
        "properties": {
          "property": [
            {
              "name": "os_password",
              "value": "cisco123"
            },
            {
              "name": "os_user_domain_name",
              "value": "default"
            }
          ]
        }
      }
    }
  ],
}
```

```

        "vim_id": "default_openstack_vim",
        "id": "admin"
    }
}
]
}

```

## ソリューション

VIM コネクタが返されない場合は、追加します。1 つの VIM コネクタが返され、ステータスが「CONNECTION\_SUCCESSFUL」でない場合は、`/var/log/esc/vimmanager/vimmanager.log` で次のエントリを確認します。

```

2017-12-07 23:11:49.760 [http-nio-127.0.0.1-8095-exec-5] INFO
c.c.e.v.c.VimConnectionManagerService - Registering an user.

```

エントリの後に例外またはエラーが表示されている場合は、根本原因を示しています。たとえば、SSL に関連するエラーがある場合、証明書が見つからないか、間違っていることを意味します。

```

2017-12-07 23:11:49.818 [http-nio-127.0.0.1-8095-exec-5] ERROR
c.c.e.v.p.i.o.OpenStackProvider - Failed to register a user
org.openstack4j.api.exceptions.ConnectionException: javax.net.ssl.SSLHandshakeException:
sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid
certification path to requested target
    at
org.openstack4j.connectors.jersey2.HttpExecutorServiceImpl.invoke(HttpExecutorServiceImpl.java:58)

```

接続タイムアウトがある場合、またはホスト名が例外の名前でない場合は、提供された AuthUrl に対して `CURL get` 呼び出しを試し、OpenStack が ESC VM から到達可能であることを確認します。

```

curl -k https://www.xxxxx.com:5000/

```

「Register an user」エントリの後にエラーも例外も表示されない場合は、提供されたログイン情報が正しくないことを意味します。この場合、`/var/log/esc/vimmanager/vim_vimmanager.log` を確認してください。最初の認証が行われたログファイルの先頭を確認します。

```

2017-12-07 23:11:49.748 INFO os - 1 * Sending client request on thread
http-nio-127.0.0.1-8095-exec-4
1 > POST https://10.85.103.49:35357/v3/auth/tokens
1 > Accept: application/json
1 > Content-Type: application/json
1 > OS4J-Auth-Command: Tokens
{
  "auth" : {
    "identity" : {
      "password" : {
        "user" : {
          "name" : "admin",
          "domain" : {
            "name" : "default"
          },
        },
        "password" : "****"
      }
    },
    "methods" : [ "password" ]
  },
  "scope" : {
    "project" : {
      "name" : "admin",
      "domain" : {

```



```
        "name" : "default"
      }
    }
  }
}
```

authUrl、ユーザー、プロジェクトまたはテナントを再確認します。V3 認証の場合、authUrl が実際の V3 エンドポイントであることを確認してください。そうでない場合は、404 が返されます。V3 認証の場合も、ユーザードメインとプロジェクトドメインが提供されていることを確認します。Horizon の OpenRC ファイルを使用して ESC VM を起動し、OpenRC にプロジェクトドメインまたはユーザードメインが含まれていない場合は、明示的に宣言します。

```
OS_PROJECT_DOMAIN_NAME=default
OS_USER_DOMAIN_NAME=default
```

ESC が bootvm を使用して、デフォルトの VIM コネクタの正しいパスワードを取得するか確認するには、次の手順を実行します。

```
admin@leishi-test ~]$ sudo escadm reload
[sudo] password for admin:
[admin@leishi-test ~]$ cat /opt/cisco/esc/esc-config/esc_params.conf
openstack.os_auth_url= http://10.85.103.153:35357/v3
openstack.os_project_name= admin
openstack.os_tenant_name= admin
openstack.os_user_domain_name= default
openstack.os_project_domain_name= default
openstack.os_identity_api_version= 3
openstack.os_username = admin
openstack.os_password = cisco123
```

VIM コネクタの問題により VNF 操作が拒否される

## 翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。