



## 仮想ネットワーク機能の操作

- [VNF 操作 \(1 ページ\)](#)
- [VNF バックアップおよび復元操作 \(2 ページ\)](#)
- [個々の VNF と複合 VNF の管理 \(13 ページ\)](#)

### VNF 操作

VNF を起動、停止、および再起動できます。起動、停止、および再起動の操作は、RESTful インターフェイスを使用して実行されます。

VNF 操作にはペイロードが必要です。

```
POST ESCManager/v0/{internal_tenant_id}/deployments/service/{internal_deployment_id}
```

例、

```
<?xml version='1.0' encoding='UTF-8'?>  
<service_operation xmlns='urn:ietf:params:xml:ns:netconf:base:1.0'>  
  <operation>stop</operation>  
</service_operation>
```

操作フィールドで起動、停止、または再起動を指定する必要があります。

- VNF の起動：すべての VM が起動し、モニタリングが有効になり、KPI の詳細に従ってしきい値が再割り当てされます。VM の実行が開始され、VM\_ALIVE\_STATE に移動します。サービスは service\_active\_state になります。VNF の起動ワークフローを中断できるのは展開解除のみです。
- VNF の停止：サービスが停止すると、モニタリングが無効になり、すべての VM サービスが停止します。VM は使用できなくなります。サービスは service\_stopped\_state になります。VM は shutoff\_state になります。リカバリ、スケールアウト、スケールインを実行することはできません。VNF の展開解除のみ可能です。
- VNF の再起動：モニタリングが無効になり、すべての VM が再起動します。つまり、VM が停止してから OpenStack で起動し、モニタリングが有効になり、KPI の詳細に従ってしきい値が再割り当てされます。VM は VM\_ALIVE\_STATE となり、サービスは service\_alive\_state です。再起動操作を中断できるのは展開解除のみです。

すでに実行中の VNF のモニタリングを開始することはできません。再起動後、VM に再度ログインすると、再起動、更新、およびモニタリングの詳細が示されます。また、リカバリも示す必要があります。

### VM の操作

VNF 操作と同様に、個々の VM を起動、停止、および再起動できます。

VM 操作にはペイロードが必要です。

```
POST ESCManager/v0/{internal_tenant_id}/deployments/vm/{vm_name}
```

例、

```
<?xml version='1.0' encoding='UTF-8'?>
<vm_operation xmlns='urn:ietf:params:xml:ns:netconf:base:1.0'>
  <operation>stop</operation>
  <force>true/false</force>
</vm_operation>
```

操作フィールドで起動、停止、または再起動を指定する必要があります。

## VNF バックアップおよび復元操作

ここでは、VM スナップショットを使用した VNF のバックアップおよび復元操作について説明します。

### VNF バックアップ操作

#### VM スナップショットの管理

ESC は、OpenStack VIM 上にイメージ（特定の状況ではボリューム）であるスナップショットを作成します。ESC API は、ESC によって管理される VNF のスナップショットを管理します。ESC は、次の 3 つの主なスナップショット操作をサポートします。

- VM スナップショットの作成
- VM スナップショットの一覧表示
- VM スナップショットの削除

ESC は、HTTP および HTTPS プロトコルで ESC REST API を使用して VM スナップショット操作を実行します。VM スナップショットの作成および削除操作は、`esc_nc_cli` スクリプトによってサポートされます。NETCONF 通知と REST API 通知はどちらも、作成および削除操作に対するスナップショット操作のさまざまな段階で生成されます。




---

(注) VM スナップショット操作は、OpenStack VIM でのみサポートされています。

---

## VM スナップショットの作成

ESC VM によって管理される任意の VNF から（REST API または `esc_nc_cli` スクリプトを使用して）スナップショットを作成できます。スナップショットは、（ESC VM ステータスが `VM_ALIVE` または `VM_STOPPED` に変換される）アクティブな VNF または停止した VNF に対してのみ作成できます。API 呼び出しのペイロードでスナップショット名を指定できます。スナップショット名が一意でない場合、ESC スナップショット操作ペイロードを指定するときに参照として使用されるスナップショット名とともに一意の ID が生成されます。OpenStack 上にスナップショット（イメージ）が作成されます。ブート可能ボリュームを使用する VNF の場合、ボリュームスナップショットも OpenStack に作成されます。

### REST API を使用したスナップショットの作成

スナップショットを作成するには、ESCManager API に HTTP POST 操作を指定します。

```
POST: /ESCManager/v0/<tenant-id>/deployments/snapshot-vm/<generated-vm-name>
```

ペイロードには `operation` と `name` の値が含まれている必要があり、操作の値は `snapshot` である必要があります。

```
operation: snapshot
name: <snapshot-name>
```

成功すると、HTTP 200 コードが返され、ペイロードはありません。

失敗した場合（検証エラーまたは OpenStack API エラー）、適切な HTTP エラーコードとエラーメッセージが返されます。

以下は、スナップショットを作成するための API 呼び出しを示しています。

```
[admin@localhost]$ cat snapshot.json
{
  "operation": "snapshot",
  "name": "my-snapshot-name"
}

[admin@localhost]$ curl -X POST -d @snapshot.json -H 'Content-Type: application/json'
-H 'callback: http://localhost:9009' -H 'Callback-ESC-Events: http://localhost:9009'
"http://localhost:8080/ESCManager/v0/snapshot-tenant/deployments/snapshot-vm/new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba"
```

### esc\_nc\_cli スクリプトを使用したスナップショットの作成

`esc_nc_cli` スクリプトを使用してスナップショットを作成するには、生成された VM の名前と操作を指定する固定パラメータを渡します。

```
VM Backup Action : vm-backup-action vm-name backup-name [<action-type>] [<xmlfile>]
action-type := SNAPSHOT|EXPORT
```

オプションの `action-type` パラメータは、指定されていない場合、デフォルトで `SNAPSHOT` になります。以下は、スナップショットを作成するためのスクリプトの呼び出しを示しています。

```
[admin@localhost]$ esc_nc_cli vm-backup-action
new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba my-snapshot-name SNAPSHOT
VM Backup Action
/opt/cisco/esc/confd/bin/netconf-console --port=830 --host=127.0.0.1 --user=esc-nc-admin
--privKeyFile=/home/admin/.ssh/confd_id_rsa --privKeyType=rsa
--rpc=/tmp/tmp_esc_nc_cli.c8d9kAjcGf
```

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <ok/>
</rpc-reply>
```

成功した場合、単一の <ok/> 要素を含む XML ペイロードが返されます。失敗した場合（検証エラーまたは OpenStack API エラー）、適切なエラーメッセージが返されます。

### 注記

ESC REST API と `esc_nc_cli` の両方について、次の点に注意してください。

- スナップショット名の長さは 255 文字以下にする必要があります。
- 生成された VM 名は有効である必要があります。
- `action-type` は SNAPSHOT または EXPORT (`esc_nc_cli` のみ) である必要があります。
- `xmlfile` : 指定されている場合、有効な XML ドキュメントが含まれている必要があります (`esc_nc_cli` のみ)。

### [Notifications]

スナップショットの作成操作中に、NETCONF 通知と ESC REST コールバックメッセージの両方が送信されます。

表 1:

通知 (NETCONF または ESC コールバック)	通知の送信タイミング
VM_BACKUP_INIT	API が呼び出され、検証に合格した場合。
VM_BACKUP_CREATED	OpenStack がスナップショットの作成要求を正常に受信して検証した場合。
VM_BACKUP_COMPLETE	OpenStack がスナップショットの作成要求操作を完了し、成功したか、エラーが発生した場合。

次に、VM\_BACKUP\_CREATED の成功した NETCONF 通知の例を示します（他の通知も同様です）。

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2021-09-14T12:18:39.836+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_code>202</status_code>
    <status_message>Snapshot is now active.</status_message>
    <depname>snapshot-deployment-name</depname>
    <tenant>snapshot-tenant</tenant>
    <tenant_id>7d61b5de73874f88a458d486759a9b83</tenant_id>
    <depid>ae0bea05-9630-4d17-a9e7-926f1f625dc7</depid>
    <vm_group>snapshot-group</vm_group>
    <vm_source>
      <vmid>1773914c-20cd-4f50-b337-1e46be2cf295</vmid>
```

```

<vmname>new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba</vmname>

<generated_vmname>new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba</generated_vmname>

  <vim_id>default_openstack_vim</vim_id>
  <vim_project>snapshot-tenant</vim_project>
  <vim_project_id>7d61b5de73874f88a458d486759a9b83</vim_project_id>
  <hostid>95503baadecce2d33e5d924322390aee9d30c6ed24043284bf46984</hostid>
  <hostname>pf-ucs-27</hostname>
</vm_source>
<event>
  <type>VM_BACKUP_CREATED</type>
</event>
</escEvent>
</notification>

```

失敗した場合、NETCONF 通知と ESC REST コールバックメッセージは引き続き生成されますが、次のようになります。

- <status> 値は FAILURE になります。
- <status\_code> は 500 になります。
- <status\_message> は、内部で生成されるか、OpenStack から送り返される適切なメッセージになります。

## スナップショットの一覧表示

ESC REST API を使用してスナップショットを一覧表示できます。ESC で管理しているスナップショットのみ一覧表示できます。スナップショットデータのサブセットをクエリパラメータとして指定して、返されるスナップショットの数を減らすことができます。返されるスナップショットデータは、HTTP Accept ヘッダーによって制御される XML または JSON 形式にできます。Accept ヘッダーの値は、指定されていない場合、デフォルトで XML になります。

ESC REST API のみがスナップショットの一覧表示をサポートしています。esc\_nc\_cli は、ESC 管理対象エンティティの一覧表示をサポートしていません。

### ESC REST API を使用したスナップショットの一覧表示

スナップショットを一覧表示するには、ESCManager API に対して HTTP GET 操作を指定できます。

```
GET: /ESCManager/v0/snapshots
```

オプションのクエリパラメータ (internalTenantId、generatedVMName) も指定できます。

返されるスナップショットの数に関係なく、常に HTTP 200 コードが返されます。

以下は、特定の内部テナント ID と生成された VM 名のスナップショットを一覧表示する API 呼び出しを示しています。

```

[admin@localhost]$ curl -X GET --header "Accept: application/xml"
"http://localhost:8080/ESCManager/v0/snapshots?internalTenantId=snapshot-tenant&generatedVMName=new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba"
| xmllint --format -

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<snapshots>
  <snapshot xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">

```

```

<id>7813c20b-94b6-492b-ae74-0bd36c1168dc</id>
<name>my-snapshot-name</name>
<creation_start_date>2021-07-20T11:26:47.532Z</creation_start_date>
<creation_end_date>2021-07-20T11:27:53.139Z</creation_end_date>
<status>available</status>
<status_message>Snapshot image for VM [gen_vm_name] is active.</status_message>

<gen_vm_name>new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba</created_from_generated_vm_name>

<vim_id>default_openstack_vim</vim_id>
<tenant>snapshot-tenant</tenant>
<bootable_volume_snapshot_id:c3cd5d13-63bf-49f0-b864-df3bc024d5e4/>
</snapshot>
</snapshot>

```



(注) この API 呼び出し時に生成される通知はありません。

### スナップショットの削除

ESC で作成したスナップショットは、REST API または `esc_nc_cli` スクリプトを使用して削除できます。削除できるのは現在の ESCVM によって管理されているスナップショットのみで、一度に削除できるスナップショットは1つだけです。成功した場合、スナップショットはESC から削除され、OpenStack 内の関連するイメージとボリュームのスナップショット（ある場合）も削除されます。

#### REST API を使用したスナップショットの削除

ESC を介して以前に作成されたスナップショットを削除するには、ESCManager API に対して HTTP DELETE 操作を指定できます。

```
DELETE: /ESCManager/v0/snapshots/<snapshot-id|snapshot-name>
```

スナップショット ID またはスナップショット名を渡すことができます。成功すると、HTTP 200 コードが返され、ペイロードはありません。失敗した場合（検証エラーまたは OpenStack API エラー）、適切な HTTP エラーコードとエラーメッセージが返されます。以下は、スナップショットを削除するための API 呼び出しを示しています。

```
[admin@localhost]$ curl -X DELETE -H 'callback: http://localhost:9009' -H
'Callback-ESC-Events: http://localhost:9009'
"http://localhost:8080/ESCManager/v0/snapshots/7813c20b-94b6-492b-ae74-0bd36c1168dc"
```

#### esc\_nc\_cli を使用したスナップショットの削除

`esc_nc_cli` スクリプトを使用してスナップショットを削除するには、スナップショット ID またはスナップショット名のみを単一のパラメータとして渡す必要があります。

```
Snapshot Action : snapshot-action <snapshot-id|snapshot-name>
```

以下は、スナップショットを作成するためのスクリプトの呼び出しを示しています。

```
[admin@localhost]$ esc_nc_cli snapshot-action delete my-snapshot-name
```

または

```
[admin@localhost]$ esc_nc_cli snapshot-action delete my-snapshot-name-1
<?xml version="1.0" encoding="UTF-8"?>
<error xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
<error_code>404</error_code>
<error_message>Snapshot image [my-snapshot-name-1] not found.</error_message>
</error>
```

成功した場合、単一の <ok/> 要素を含む XML ペイロードが返されます。失敗した場合（検証エラーまたは OpenStack API エラー）、適切なエラーメッセージが返されます。

ESC REST API と `esc_nc_cli` の両方について、次の点に注意してください。

- スナップショット ID またはスナップショット名は有効である必要があります。
- スナップショット名を指定する場合は、一意の名前にする必要があります。

**通知**

スナップショットの削除操作中に、NETCONF 通知と ESC REST コールバックメッセージの両方が送信されます。

通知は次のとおりです。

表 2:

通知（NETCONF や ESC コールバック）	通知の送信タイミング
VM_SNAPSHOT_DELETING	送信と検証が成功した場合。
VM_SNAPSHOT_DELETED	OpenStack がスナップショットの削除操作を完了し、成功したか、エラーが発生した場合。 スナップショットにイメージスナップショットとともにボリュームスナップショットがある場合、両方が削除されるまで通知は送信されません。

次に、VM\_SNAPSHOT\_DELETED の成功した NETCONF 通知の例を示します（他の通知も同様です）。

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2021-09-14T12:18:39.836+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_code>200</status_code>
    <status_message>Snapshot image [2ffadd36-3b41-4c13-a9d6-a48c07764d1a] has been
deleted.</status_message>
    <depname>snapshot-deployment-name</depname>
    <tenant>snapshot-tenant</tenant>
    <tenant_id>7d61b5de73874f88a458d486759a9b83</tenant_id>
    <depid>ae0bea05-9630-4d17-a9e7-926f1f625dc7</depid>
    <vm_group>snapshot-group</vm_group>
    <vm_source>
      <vmid>1773914c-20cd-4f50-b337-1e46be2cf295</vmid>
      <vmname>new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba</vmname>
    </vm_source>
    <gen_vm_name>new-deployment-n_new-gr_0_af0148e2-e74c-4be7-b8c1-49bd53def6ba</generated_vmname>
    <vim_id>default_openstack_vim</vim_id>
```

```

    <vim_project>snapshot-tenant</vim_project>
    <vim_project_id>7d61b5de73874f88a458d486759a9b83</vim_project_id>
    <hostid>95503baadecce2d33e5d924322390aee9d30c6ed24043284bf46984</hostid>
    <hostname>pf-ucs-27</hostname>
  </vm_source>
  <event>
    <type>VM_SNAPSHOT_DELETED</type>
  </event>
</escEvent>
</notification>

```

失敗した場合、NETCONF 通知と ESC REST コールバックメッセージは引き続き生成されますが、次のようになります。

- <status> 値は FAILURE になります。
- <status\_code> は 500 になります。
- <status\_message> は、内部で生成されるか、OpenStack から送り返される適切なメッセージになります。

### VM スナップショット ポーリング パラメータ

VIM Manager の構成プロパティを使用して、OpenStack への呼び出しの間隔を制御して、作成操作と削除操作のステータスを確認するとともに、操作がエラーでタイムアウトするまでの最大許容時間（分）を制御できます。

次のプロパティがこれに該当します。

- vim.asyncpoller.snapshot.create.poll.secs # default 15、ポーリング間の秒数
- vim.asyncpoller.snapshot.create.timeout.mins # default 20、スナップショット作成操作の最大分数
- vim.asyncpoller.snapshot.delete.poll.secs # default 15、ポーリング間の秒数
- vim.asyncpoller.snapshot.delete.timeout.mins # default 20、スナップショット削除操作の最大分数

デフォルト値は、以下に示すように、/opt/cisco/esc/vimmanager/application.properties の下にある application.properties ファイルに設定し、VIM Manager サービスを再起動することでオーバーライドできます。

```

[admin@localhost]$ sudo cat /opt/cisco/esc/vimmanager/application.properties
vim.asyncpoller.snapshot.create.poll.secs=5
vim.asyncpoller.snapshot.create.timeout.mins=10
vim.asyncpoller.snapshot.delete.poll.secs=10
vim.asyncpoller.snapshot.delete.timeout.mins=60

```

```

[admin@localhost]$ sudo escadm vimmanager restart
Stopping vimmanager service: [OK]
Starting vimmanager service: [OK]

```

```

[admin@localhost]$ sudo escadm vimmanager show
VimManager System Configurations.
{
  "ccp.pollRetries": "200",
  "ccp.pollRetryDelaySecs": "15",

```



```

. . .

"vim.asyncpoller.snapshot.create.poll.secs": "5",
"vim.asyncpoller.snapshot.create.mins": "10",
"vim.asyncpoller.snapshot.delete.poll.secs": "10",
"vim.asyncpoller.snapshot.delete.timeout.mins": "60",

. . .

"vmware.ovftool.params": "--acceptAllEulas --disableVerification --noSSLVerify
--allowExtraConfig",
"vmware.powerOnRetry": "8"
}

```

HAセットアップでは、アプリケーションプロパティファイルを両方のノードにコピーする必要があります。

または、`escadm` スクリプトを使用して、値を動的に設定できます（ただし、それらの値は再起動後は保持されません）。

```

[admin@localhost] sudo escadm vimmanager set --config
vim.asyncpoller.snapshot.create.poll.secs=200
vim.asyncpoller.snapshot.create.timeout.mins=1
VimManager configuration [vim.asyncpoller.snapshot.create.poll.secs] has updated to
[200].
VimManager configuration [vim.asyncpoller.snapshot.create.timeout.mins] has updated to
[1].

```

### ブート可能ボリュームのある VNF のスナップショット

ブートボリュームのある ESC 管理対象 VNF のスナップショットが作成された場合、イメージスナップショットとボリュームスナップショットの両方が **OpenStack** 内に作成されます。



- (注) イメージスナップショット名は、スナップショットペイロードで指定されたスナップショット名になります。ボリュームのスナップショット名（該当する場合は、スナップショットの前に付加されます。

たとえば、ブート可能ボリュームのある VNF の ESC VM でスナップショットが作成され、そのスナップショットに `my-snapshot-name` という名前が付けられた場合、次のことが当てはまります。

```

[admin@localhost]$ openstack volume snapshot list | grep my-snapshot-name
| 52a96891-f22d-4863-bb47-bd9442ca0cb1 | snapshot for my-snapshot-name | None | available
| 2 |

[admin@localhost]$ openstack image list | grep my-snapshot-name
| c8846c14-48e4-45db-88a0-f838fc3ac29d | my-snapshot-name | active |

```

ボリュームスナップショットは、ESC 内で直接使用することも、**OpenStack** でネイティブに復元操作で使用することもできません。最初にスナップショットからブート可能ボリュームを作成する必要があります。ESC は、ボリュームスナップショットからのブート可能ボリュームの作成をサポートしています。詳細については、「VNF 復元操作」を参照してください。

## VNF 復元操作

### ブート可能ボリュームのない VNF のスナップショット

ESC がブート不能ボリュームを使用して VNF のスナップショットを取得した場合、そのスナップショットは OpenStack にスナップショットイメージとして保存されます。そのスナップショットイメージは、ESC のサービス更新機能を介して復元するために使用できます。サービス更新 XML は、元の展開 XML と同じように作成できますが、スナップショットイメージ名を使用します。このサービス更新 XML が REST または `esc_nc_cli` インターフェイスを使用して ESC に展開されると、ESC は VNF のイメージ名を内部的に更新し、次の再展開時に（通常は REST または `esc_nc_cli` インターフェイスを使用して手動でトリガーされます）、新しいイメージを使用して新しい展開が作成されます。

### ブート可能ボリュームのある VNF のスナップショット

ESC がブート可能ボリュームを使用して VNF のスナップショットを作成すると、そのスナップショットは、スナップショットイメージとボリュームスナップショットの両方として OpenStack に保存されます。ボリュームスナップショットは、ESC または OpenStack では、VNF の復元プロセス内で直接使用できません。ブート可能ボリュームは、ESC の観点からアウトオブバンドと見なされる（つまり、ESC によって直接管理されない）ボリュームスナップショットから最初に作成する必要があります。ブート可能ボリュームが作成されて利用可能になると、ESC のサービス更新機能を介して復元するために使用できます。サービス更新 XML は、元の展開 XML と同じように作成できますが、指定された元のボリュームに対する削除操作と、新しいブート可能ボリュームを指定する作成操作を実行します。このサービス更新 XML が REST または `esc_nc_cli` インターフェイスを使用して ESC に展開されると、ESC は、VNF を再展開することなく、元のボリュームを新しいボリュームに自動的にスワップアウトするため、すべての OpenStack UUID とリソースが保持されます。

元のボリューム（VNF から切り離されたボリューム）は引き続き OpenStack に残るため、手動でクリーンアップする必要があります。

### ボリュームスナップショットからのブート可能ボリュームの作成

ESC がブート可能ボリュームのある VNF のスナップショットを取得すると、OpenStack でボリュームのスナップショットが作成され、ESC を使用してボリュームスナップショットからブート可能ボリュームを作成できます。ESC 内で、または OpenStack API を直接使用して、復元操作を実行するには、ボリュームスナップショットからブート可能ボリュームを作成する必要があります。ボリューム名は、ESC REST API 呼び出しのペイロードで指定できます。

`esc_nc_cli` スクリプトは、ボリュームスナップショットからのブート可能ボリュームの作成をサポートしていません。ボリューム名は一意である必要はありません。ESC 復元操作のペイロードを指定するときに参照として使用できる名前と一緒に一意の ID が生成されます。ボリュームスナップショットから正常に作成されたボリュームは最終的に、OpenStack で新しいブート可能ボリュームになります。



- (注) OpenStack 上の新しいブート可能ボリュームは、ESC によって管理されません。このボリュームはアウトオブバンドであり、Orchestrator で直接管理する必要があります。

### REST API を使用したボリュームスナップショットからのブート可能ボリュームの作成

ボリュームスナップショットからボリュームを作成するには、ESCManager API のスナップショットエンドポイントに対して HTTP POST 操作を指定できます。

POST: /ESCManager/v0/snapshots/<snapshot-id>/volumes

ペイロードには、新しいボリュームの名前となる **name** の値が含まれている必要がありますが、オプションで **volume\_type**、**multiattach**、および **bootable** を指定できます。

```
operation: snapshot
name: <snapshot-name>
volume_type: <valid-volume-type> # defaults to the OpenStack default volume type
multiattach: <true|false> # defaults to false
bootable: <true|false> # defaults to true
```

成功すると、操作が正常に OpenStack に送信されたことを示す HTTP 202 コードが返され、ペイロードはありません。失敗した場合（検証エラーまたは OpenStack API エラー）、適切な HTTP エラーコードとエラーメッセージが返されます。

以下は、スナップショット ID を決定するために、最初に ESC 内で一覧表示した後、スナップショットを作成する API 呼び出しを示しています。

```
[admin@localhost]$ curl -s
"http://localhost:8080/ESCManager/v0/snapshots?internalTenantId=dave-2000" | xmllint
--format -
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<snapshots>
  <snapshot xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <id>171ffa7d-8318-47d1-acab-b01db4501a39</id>
    <name>my-snapshot-name</name>
    <creation_start_date>2021-09-20T08:29:22.074+01:00</creation_start_date>
    <creation_end_date>2021-09-20T08:33:41.193+01:00</creation_end_date>
    <status>active</status>
    <status_message>Snapshot image for
[new-dep-4_new-gr_0_34b9da8a-af64-4452-a8a3-8972e23e4e98] is active.</status_message>
<created_from_generated_vm_name>new-dep-4_new-gr_0_34b9da8a-af64-4452-a8a3-8972e23e4e98</created_from_generated_vm_name>
    <vim_id>my-snapshot-vim</vim_id>
    <tenant>dave-2000</tenant>
    <volume_snapshot_id>c4548ba4-0480-4b42-8229-ad98de44b3ea</volume_snapshot_id>
  </snapshot>
</snapshots>

[admin@localhost]$ cat volume_from_volume_snapshot.json
{
  "name": "my-bootable-volume-from-snapshot-volume",
  "multiattach": true
}

[admin@localhost]$ curl -X POST -d @volume_from_volume_snapshot.json -H 'Content-Type:
application/json' -H 'callback: http://localhost:9009' -H 'Callback-ESC-Events:
http://localhost:9009'
"http://localhost:8080/ESCManager/v0/snapshots/171ffa7d-8318-47d1-acab-b01db4501a39/volumes"
```

```
[admin@localhost]$ openstack volume list | grep my-bootable-volume
| c4548ba4-0480-4b42-8229-ad98de44b3ea | my-bootable-volume-from-snapshot-volume |
available | 2 | |
```

ESC REST API では、次の内容が検証されます。

- ボリューム名の長さは 255 文字以下にする必要があります。
- スナップショット ID は、ESC 管理対象スナップショットの ID である必要があります（つまり、list snapshot 操作が返すスナップショットの 1 つである必要があります）。
- name はペイロードで指定する必要があります。他の属性はすべてオプションで設定できます。
- ペイロードでサポートされていない属性名は無視されます。
- スナップショットは、ブート可能ボリュームがある VNF に対して取得されている必要があります。

### [Notifications]

この操作では、ESC REST コールバックメッセージのみが生成されます。

VM\_VOLUME\_ACCEPTED\_EVENT と VM\_VOLUME\_CREATED\_EVENT の 2 つのコールバックメッセージが生成されます。

VM\_VOLUME\_CREATED\_EVENT ESC REST コールバックメッセージの例を次に示します。

```
{
  "escTransactionId": "5acac790-9213-45c0-8fde-9dd7d3111fdb",
  "eventType": "VM_VOLUME_CREATED_EVENT",
  "eventSourceContext": null,
  "eventTargetContext": null,
  "message": "Create volume snapshot request completed",
  "stateMachineEventNBInfo": {
    "id": "de9440c0-1342-441d-a16e-5c8267231ae5",
    "message": {},
    "logNames": [],
    "keywords": {},
    "actionInfo": {},
    "stackTrace": ""
  },
  "escParameter": {
    "external_volume_id": "c3cd5d13-63bf-49f0-b864-df3bc024d5e4",
    "size": "2",
    "sizeunit": null,
    "bus": "virtio",
    "type": "LVM",
    "outOfBand": "false",
    "bootIndex": null,
    "name": "daves-ooband-bootable-volume-for-restore",
    "format": null,
    "deviceType": null,
    "storageLocation": null,
    "external_tenant_id": null,
    "internal_tenant_id": null,
    "internal_volume_id": null,
    "volid": null,
    "event_type": null,
    "image": null
  }
},
```

```

    "vmUpdateType": null,
    "requestDetails": null,
    "statusCode": "201",
    "notificationOnlyEvent": false
  }

```

### ポーリング設定パラメータ

VIM Manager の構成プロパティを使用して、OpenStack への呼び出しの間隔を制御して、ボリューム作成操作のステータスを確認するとともに、操作がエラーでタイムアウトするまでの最大許容時間（分）を制御できます。

次のプロパティがこれに該当します。

- `vim.asyncpoller.volume.create.poll.secs` # default 15、ポーリング間の秒数
- `vim.asyncpoller.volume.create.timeout.mins` # default 20、ボリューム作成操作の最大分数

```

[admin@localhost]$ sudo cat /opt/cisco/esc/vimmanager/application.properties
vim.asyncpoller.volume.create.poll.secs=5
vim.asyncpoller.volume.create.timeout.mins=10

[admin@localhost]$ sudo escadm vimmanager restart
Stopping vimmanager service: [OK]
Starting vimmanager service: [OK]

[admin@localhost]$ sudo escadm vimmanager show
VimManager System Configurations.
{
  "ccp.pollRetries": "200",
  "ccp.pollRetryDelaySecs": "15",
  . . .
  "vim.asyncpoller.volume.create.poll.secs": "5",
  "vim.asyncpoller.volume.create.mins": "10",
  . . .
  "vmware.ovftool.params": "--acceptAllEulas --disableVerification --noSSLVerify
--allowExtraConfig",
  "vmware.powerOnRetry": "8"
}

```



- (注) HA セットアップでは、アプリケーションプロパティ ファイルを両方のノードにコピーする必要があります。

## 個々の VNF と複合 VNF の管理

個々のサービスは単一の VNF で構成されます。連携サービスまたは複合 VNF は、異なるタイプの複数の VM で構成されます。ESC インターフェイスは、ノースバウンドシステムから VM 相互依存情報を受信し、VM および VNF の作成中、およびライフサイクル管理中にこの情報

を使用します。相互依存性には、単一 VNF 内の VM グループ、VNF モニタリング、拡張性などの VM 固有のワークフローが含まれます。

VM では、作成、読み取り、更新、および削除の操作が許可されます。静的 IP を使用して展開済み VNF に VM インスタンスを追加するには、追加の IP アドレスを静的 IP プールに提供する必要があります。既存の静的 IP 展開を使用している場合は、VM の最小数が変更されます。

新しい最小値（VM の数）がアクティブな VM の数より大きい場合、新しい VM がサービスに追加されます。値が最大値よりも大きい場合、更新は拒否されます。

## 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。