



VIM コネクタの設定

- [OpenStack の VIM コネクタの設定 \(1 ページ\)](#)
- [AWS の VIM コネクタ設定 \(9 ページ\)](#)
- [VMware vCloud Director \(vCD\) の VIM コネクタの設定 \(10 ページ\)](#)

OpenStack の VIM コネクタの設定

OpenStack 固有の操作用に VIM コネクタを設定できます。



(注) VIM コネクタを設定するには、[VIM コネクタの設定](#)を参照してください。

OpenStack での ESC ユーザの非管理者ロールの作成

デフォルトでは、OpenStack では ESC ユーザに管理者ロールが割り当てられます。一部のポリシーでは、特定の ESC 操作に対してデフォルトの管理者ロールの使用が制限される場合があります。ESC リリース 3.1 以降では、OpenStack で ESC ユーザの権限を制限した非管理者ロールを作成できます。

非管理者ロールを作成するには、次の手順を実行します。

1. OpenStack で非管理者ロールを作成します。
2. ESC ユーザに非管理者ロールを割り当てます。

OpenStack Horizon (アイデンティティ) で、または OpenStack コマンドラインインターフェイスを使用して、ESC ユーザロールを割り当てる必要があります。詳細については、OpenStack のマニュアルを参照してください。

ロール名は OpenStack でカスタマイズできます。デフォルトでは、OpenStack のすべての非管理者ロールに同じレベルの権限が付与されます。

3. 管理者以外のロールに必要な権限を付与します。

policy.json ファイルを変更して、必要な権限を付与する必要があります。



(注) ESC ユーザロールを動作させるには、`policy.json` ファイルの `create_port: fixed_ips` および `create_port: mac_address` パラメータに権限を付与する必要があります。

次の表に、必要な権限を取得後に非管理者ロールで実行できる ESC 操作を示します。

表 1: ESC 操作用の非管理者ロール権限

ESC VIM 操作	説明	Permission	コメント
プロジェクトの作成	OpenStack プロジェクトを作成する	<code>/etc/keystone/policy.json</code> " <code>identity:create_project</code> " " <code>identity:create_grant</code> "	ESC 管理対象 OpenStack プロジェクトの場合は、 ロールに <code>identity: create_grant</code> が必要なプロジェクトにユーザを追加します。
プロジェクトの削除	OpenStack プロジェクトを削除する	<code>/etc/keystone/policy.json</code> " <code>identity:delete_project</code> "	
イメージのクエリ	すべてのイメージのリストを取得する	不要	所有者（ターゲットプロジェクトのユーザ）がクエリを実行できます。 パブリックイメージまたは共有イメージを取得できます。
イメージの作成 (Create Image)	パブリックイメージを作成する	<code>/etc/glance/policy.json</code> " <code>publicize_image</code> "	デフォルトでは、管理者はパブリックイメージを作成できます。 イメージの公開はポリシーによって保護されます。
	プライベートイメージを作成する	不要	次の文を使用してプライベートイメージを作成できます。 <pre><image> <name>mk-test-image</name> ... <disk_bus>virtio</disk_bus> <visibility>private</visibility> </image></pre>
イメージの削除	イメージを削除する	不要	所有者はイメージを削除できます。

ESC VIM 操作	説明	Permission	コメント
フレーバーのクエリ	既存のフレーバーをクエリする	不要	所有者はフレーバーをクエリできます。 パブリックフレーバーもクエリできます。
フレーバーの作成	新しいフレーバーを作成する	/etc/nova/policy.json "os_compute_api:os-flavor-manage"	フレーバーの管理は通常、クラウドの管理者だけができます。
フレーバーの削除	フレーバーを削除する	/etc/nova/policy.json "os_compute_api:os-flavor-manage"	
ネットワークのクエリ	ネットワークのリストを取得する	/etc/neutron/policy.json "get_network"	所有者は、共有ネットワークを含むネットワークのリストを取得できます。
ネットワークの作成	通常のネットワークを作成する	不要	
	特殊なケースでネットワークを作成する	/etc/neutron/policy.json "create_network:provider:physical_network" "create_network:provider:network_type" "create_network:provider:segmentation_id" "create_network:shared"	これらのルールは、 physical_network (SR-IOV など)、 network_type (SR-IOV など)、または segmentation_id (3008 など) を使用してネットワークを作成するか、あるいはネットワークを共有用に設定する場合に必要です。 <pre>< n e t w o r k > <name>provider-network</name> <!-- <shared>>false</shared> //default is t r u e - - > <admin_state>>true</admin_state> <provider_physical_network>VAR_PHYSICAL_NET </provider_physical_network> <provider_network_type>vlan </provider_network_type> <provider_segmentation_id>2330 </provider_segmentation_id> ... </network></pre>

ESC VIM 操作	説明	Permission	コメント
ネットワークの削除	ネットワークを削除する	不要	所有者はネットワークを削除できます。
サブネットのクエリ	サブネットのリストを取得する	<code>/etc/neutron/policy.json</code> "get_subnet"	<p>ネットワークの所有者はサブネットのリストを取得できます。</p> <p>共有ネットワークからサブネットのリストを取得することもできます。</p> <pre>< n e t w o r k > <name>esc-created-network</name> <!--network must be created by E S C - - > <admin_state>>false</admin_state> < s u b n e t > <name>makulandyescextnet1-subnet1</name> <ipversion>ipv4</ipversion> < d h c p > t r u e < / d h c p > <address>10.6.0.0</address> <netmask>255.255.0.0</netmask> </subnet> </network></pre>
サブネットの作成	サブネットを作成する	不要	ネットワークの所有者はサブネットを作成できます。
サブネットの削除	サブネットを削除する	不要	ネットワークの所有者はサブネットを削除できます。
ポートのクエリ	既存のポートを取得する	不要	所有者はポートのリストを取得できます。

ESC VIM 操作	説明	Permission	コメント
ポート の作成	DHCP を 使用して ネット ワーク インター フェイス を作成す る	不要	
	MAC ア ドレスを 使用して ネット ワーク インター フェイス を作成す る	/etc/neutron/policy.json "create_port:mac_address"	<interfaces> <interface> < n i c i d > 0 < / n i c i d > <mac_address>fa:16:3e:73:19:b5</mac_address> <network>esc-net</network> </interface> </interfaces> VM リカ バリにもこの権限が必要です。
	固定 IP または共 有 IP を 使用して ネット ワーク インター フェイス を作成す る	/etc/neutron/policy.json "create_port:fixed_ips"	<subnet> <name>IP-pool-subnet</name> <ipversion>ipv4</ipversion> < d h c p > f a l s e < / d h c p > <address>172.16.0.0</address> <netmask>255.255.255.0</netmask> <gateway>172.16.0.1</gateway> < / s u b n e t > < s h a r e d _ i p > < n i c i d > 0 < / n i c i d > <static>false</static> </shared_ip> VM リカバリにもこの権限が必要で す。

ESC VIM 操作	説明	Permission	コメント
ポートの更新	ポートデバイスの所有者の更新	不要	所有者はポートを更新できます。
	アドレスペアを許可するようにポートを更新する	/etc/neutron/policy.json "update_port:allowed_address_pairs"	<pre><interface> <nicid>0</nicid> <network>VAR_MANAGEMENT_NETWORK_ID</network> <allowed_address_pairs> <network> <name>VAR_MANAGEMENT_NETWORK_ID</name> </network> <address> <ip_address>172.16.0.0</ip_address> <netmask>255.255.0.0 </netmask> </address> <address> <ip_address>172.16.6.1</ip_address> <ip_prefix>24 </ip_prefix> </address> </allowed_address_pairs> </interface></pre>
ポートの削除 (Delete Port)	ポートを削除する	不要	所有者はポートを削除できます。
ボリュームのクエリ	ボリュームのリストを取得する	不要	所有者はボリュームのリストを取得できます。
ボリュームの作成	ボリュームを作成する	不要	
ボリュームの削除	ボリュームを削除する	不要	所有者はボリュームを削除できます。
VM のクエリ	プロジェクト内のすべての VM を取得する	不要	所有者は、プロジェクト内のすべての VM のリストを取得できます。

ESC VIM 操作	説明	Permission	コメント
VM の作成	VM を作成する	不要	
	ホストのターゲット展開で VM を作成する	<code>/etc/nova/policy.json</code> <code>"os_compute_api:servers:create:forced_host"</code>	<code><placement> <type>zone_host</type></code> <code><enforcement>strict</enforcement></code> <code><host>anyHOST</host> </placement></code>
	ゾーンのターゲット展開で VM を作成する	不要	
	同じホストに VM を作成する アフィニティ/アンチアフィニティ	不要	
	サーバグループに VM を作成する アフィニティ/アンチアフィニティ	不要	このサポートは、グループ内アンチアフィニティ専用です。
[VM の削除 (Delete VM)]	VM を削除する	不要	所有者は VM を削除できます。

OpenStack でのリソース管理の詳細については、[OpenStack のリソースの管理](#)を参照してください。

OpenStack エンドポイントの上書き

デフォルトでは、ESCは認証に成功後、OpenStackが提供するエンドポイントのカタログリターンオプションを使用します。ESCはこれらのエンドポイントを使用して、OpenStackのさまざまなAPIと通信します。エンドポイントは正しく設定されていないこともあります。たとえば、OpenStackインスタンスは認証にKeyStone V3を使用するように設定されているのに、OpenStackから返されるエンドポイントがKeyStone V2用の場合があります。この問題は、OpenStack エンドポイントを上書きすることで解決できます。

VIM コネクタの設定中にOpenStack エンドポイントを上書き（設定）できます。これは、インストール時に `bootvm.py` パラメータおよび VIM コネクタ API を使用して実行できます。

次の OpenStack エンドポイントは、VIM コネクタの設定を使用して設定できます。

- OS_IDENTITY_OVERWRITE_ENDPOINT
- OS_COMPUTE_OVERWRITE_ENDPOINT
- OS_NETWORK_OVERWRITE_ENDPOINT
- OS_IMAGE_OVERWRITE_ENDPOINT
- OS_VOLUME_OVERWRITE_ENDPOINT

インストール時にOpenStack エンドポイントを上書きする場合、ユーザはESC設定パラメータファイルを作成し、ESC VMの展開中にこのファイルを引数として `bootvm.py` に渡します。

次に、`param.conf` ファイルの例を示します。

```
openstack.os_identity_overwrite_endpoint=http://www.xxxxxxxxxx.com
```

インストール時の VIM コネクタの設定の詳細については、[VIM コネクタの設定](#)を参照してください。

VIM コネクタ API (REST と NETCONF の両方) を使用して、デフォルト以外の VIM コネクタの OpenStack エンドポイントを上書き（設定）するには、新しい VIM コネクタの作成時または既存の VIM コネクタの更新時に、上書きするエンドポイントを VIM コネクタのプロパティとして追加します。

各 VIM コネクタには、独自の上書きエンドポイントを設定できます。デフォルトの上書きエンドポイントはありません。

次の例では、`os_identity_overwrite_endpoint` および `os_network_overwrite_endpoint` プロパティを追加して、エンドポイントを上書きします。

```
<esc_system_config xmlns="http://www.cisco.com/esc/esc">
  <vim_connectors>
    <!--represents a vim-->
    <vim_connector>
      <id>default_openstack_vim</id>
      <type>OPENSTACK</type>
      <properties>
        <property>
          <name>os_auth_url</name>
          <value>http://172.16.0.0:35357/v3</value>
        </property>
        <property>
```



```

        <name>os_project_domain_name</name>
        <value>default</value>
    </property>
    <property>
        <name>os_project_name</name>
        <value>admin</value>
    </property>
    <property>
        <name>os_identity_overwrite_endpoint</name>
        <value>http://some_server:some_port</value>
    </property>
    <property>
        <name>os_network_overwrite_endpoint</name>
        <value>http://some_other_server:some_other_port</value>
    </property>
</properties>
</vim_connector>
</vim_connectors>
</esc_system_config>

```

AWS の VIM コネクタ設定

VIM コネクタと VIM ユーザ API を使用して、AWS 展開の VIM クレデンシャルを設定できません。



(注) AWS 展開では、デフォルトの VIM コネクタはサポートされていません。

VIM コネクタの `aws_default_region` 値は認証を提供し、VIM ステータスを更新します。認証後にデフォルトリージョンを変更することはできません。

VIM コネクタの設定

AWS 展開用の VIM コネクタを設定するには、AWS クレデンシャルから `AWS_ACCESS_ID`、`AWS_SECRET_KEY` を指定します。

```
[admin@localhost ~]# esc_nc_cli edit-config
aws-vim-connector-example.xml
```



(注) 既存の VIM コネクタ設定を編集するには、必要な変更を加えた後、同じコマンドを使用します。

AWS VIM コネクタの例は次のとおりです。

```

<esc_system_config xmlns="http://www.cisco.com/esc/esc">
  <vim_connectors>
    <vim_connector>
      <id>AWS_EAST_2</id>
      <type>AWS_EC2</type>
      <properties>
        <property>
          <name>aws_default_region</name>
          <value>us-east-2</value>

```

```

    </property>
  </properties>
</users>
<user>
  <id>AWS_ACCESS_ID</id>
  <credentials>
    <properties>
      <property>
        <name>aws_secret_key</name>
        <encrypted_value>AWS_SECRET_KEY</encrypted_value>
      </property>
    </properties>
  </credentials>
</user>
</users>
</vim_connector>
</vim_connectors>
</esc_system_config>

```

VIM コネクタの削除

既存の VIM コネクタを削除するには、最初に展開、VIM ユーザ、次に VIM コネクタを削除する必要があります。

```
[admin@localhost ~]# esc_nc_cli delete-vimuser
AWS_EAST_2 AWS_ACCESS_ID
```

```
[admin@localhost ~]# esc_nc_cli delete-vimconnector
AWS_EAST_2
```



(注) 同じ VIM タイプに対して複数の VIM コネクタを設定できます。

AWS 展開用の VIM コネクタは、VIM コネクタの API を使用して設定する必要があります。

ESC は、VIM コネクタごとに 1 人の VIM ユーザをサポートします。

VIM コネクタとそのプロパティは、展開後に更新できません。

AWS での VNF の展開については、[単一または複数の AWS リージョンでの VNF の展開](#)を参照してください。

VMware vCloud Director (vCD) の VIM コネクタの設定

vCD 組織に接続するには、VIM コネクタを設定する必要があります。組織および組織ユーザは、VMware vCD で事前設定する必要があります。展開データモデルについては、「VMware vCloud Director (vCD) での仮想ネットワーク機能の展開」を参照してください。

VIM コネクタの詳細は次のとおりです。

```

<?xml version="1.0" encoding="UTF-8"?>
<esc_system_config xmlns="http://www.cisco.com/esc/esc">
  <vim_connectors>

```

```
<vim_connector>
  <id>vcd_vim</id>
  <type>VMWARE_VCD</type>
  <properties>
    <property>
      <name>authUrl</name>
      <!-- vCD is the vCD server IP or host name -->
      <value>https://vCD</value>
    </property>
  </properties>
  <users>
    <user>
      <!-- the user id here represents {org username}@{org name} -->
      <id>user@organization</id>
      <credentials>
        <properties>
          <property>
            <name>password</name>
            <!--the organization user's password-->
            <value>put user's password here</value>
          </property>
        </properties>
      </credentials>
    </user>
  </users>
</vim_connector>
</vim_connectors>
</esc_system_config>
```

