



Amazon Web Services での仮想ネットワーク機能の展開

- [Amazon Web Services での仮想ネットワーク機能の展開 \(1 ページ\)](#)

Amazon Web Services での仮想ネットワーク機能の展開

ここでは、Elastic Services Controller (ESC) の展開シナリオと、Amazon Web Services (AWS) に VNF を展開する手順について説明します。AWS に ESC をインストールする場合は、*Cisco Elastic Services Controller* インストールおよびアップグレードガイド [英語] を参照してください。

展開前に、次の AWS リソースを AWS で作成する必要があります。

- Amazon マシンイメージ (AMI)
- キーペア
- Elastic IP
- セキュリティ グループ
- ネットワーク要素 (VPC、サブネット、ACL、ゲートウェイ、ルートなど)

これらのリソースを作成するには、AWS のマニュアルを参照してください。

AWS 展開前の VIM コネクタ設定の詳細については、「AWS の VIM コネクタ設定」を参照してください。

シナリオ	説明	リソース	利点
ESC を使用した Amazon マシンイメージ (AMI) およびリージョンの作成による単一 VIM への VNF の展開	展開データモデルは、Amazon マシンイメージ (AMI)、フレーバ、AWS リージョン、キーペア、セキュリティグループ、ネットワークインターフェイス、および作成された VIM プロジェクトを参照し、VNF を展開します。	Amazon マシンイメージ (AMI)、フレーバ、AWS リージョン、キーペア、セキュリティグループ、ネットワークインターフェイス、VIM プロジェクト (ロケータで指定)、および ESC によって作成されたネットワーク。	<ul style="list-style-type: none"> 展開内の ESC で設定する必要がある (VM を展開するための) VIM を指定できます。 イメージとフレーバは、複数の VNF 展開で使用できます。 ESC によって作成されたリソースを削除できます。
ESC を使用した AMI およびリージョンの作成による、複数の VIM への VNF の展開	展開データモデルは、Amazon マシンイメージ (AMI)、フレーバ、AWS リージョン、キーペア、セキュリティグループ、ネットワークインターフェイス、および作成された VIM プロジェクトを参照し、VNF を展開します。	イメージ、フレーバ、VIM プロジェクト (ロケータで指定) および ESC を使用して作成されたネットワーク。	展開内の ESC で設定する必要がある (VM を展開するための) VIM を指定できます。

詳細については、[単一または複数の AWS リージョンでの VNF の展開 \(2 ページ\)](#) を参照してください。

単一または複数の AWS リージョンでの VNF の展開

ESC を使用して、単一または複数の AWS リージョンまたは同じタイプの VIM に VNF を展開できます。



(注) AWS は ESC 用の仮想インフラストラクチャ マネージャ (VIM) です。このドキュメントでは、AWS リージョンと AWS VIM という用語は同じ意味で使用されています。

単一または複数の VIM に VNF を展開するには、次の手順を実行する必要があります。

- VIM コネクタ API を使用して VIM コネクタとそのログイン情報を設定する
- ESC 内にテナントを作成する

VIM コネクタは VIM を ESC に登録します。単一または複数の AWS VIM に VNF を展開するには、VIM のリージョンごとに VIM コネクタとそのログイン情報を設定する必要があります。VIM コネクタ API を使用して VIM コネクタを設定できます。詳細については、[AWS の VIM コネクタ設定](#) を参照してください。



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

ESC は、`vim_mapping` 属性が `false` に設定されている ESC 内にテナントを作成します。このテナントは、VIM から独立しています。

```
<esc_datamodel xmlns="http://www.cisco.com/esc/esc">
  <tenants>
    <tenant>
      <name>aws-sample-tenant</name>
      <vim_mapping>>false</vim_mapping>
    </tenant>
  </tenants>
</esc_datamodel>
```

単一または複数の AWS VIM 展開の場合、各 VM グループのターゲットリージョンを指定する必要があります。

AWS VIM 展開を有効にするには、データモデルの VM グループにロケータ属性を追加する必要があります。ロケータノードは、次の属性で構成されます。

- `vim_id` : ターゲット VIM の VIM ID。ESC は `vim_id` を定義し、`vim_connector` ID にマッピングします。VIM コネクタは、`vim_id` で指定された VIM に展開する前に存在している必要があります。
- `vim_project` : ターゲット VIM で作成されたテナント名。これは、OpenStack に存在するアウトオブバンドテナントまたはプロジェクトです。
- `vim_region` : VM グループが展開されている AWS リージョン。これはオプションです。VIM リージョンが指定されていない場合、VM は VIM コネクタで指定された `aws_default_region` に展開されます。

```
<locator>
  <vim_id>AWS_EAST_2</vim_id>
  <vim_region>us-east-1</vim_region>
  <!-- the deployment is going into
North Virginia -->
</locator>
```

VIM リージョンが指定されていない場合

```
<locator>
  <vim_id>AWS_EAST_2</vim_id>
  <!-- the deployment is going into the default region Ohio (us-east-2)
as defined in the VIM Connector example above -->
```

```
</locator>
```

VIM コネクタとロケータを設定したら、特定のリソースを拡張機能として展開に渡す必要があります。次の例では、Elastic IP、キーペア、および送信元の宛先が拡張機能として AWS 展開に渡されます。

```
<extensions>
  <extension>
    <name>AWS_PARAMS</name>
    <properties>
      <property>
        <name>elastic_ip</name>
        <value>13.56.148.25</value>
      </property>
      <property>
        <name>source_dest_check</name>
        <value>true</value>
      </property>
      <property>
        <name>key_pair_name</name>
        <value>esc-us-east-1</value>
      </property>
    </properties>
  </extension>
</extensions>
```

AWS の展開例は次のとおりです。

```
<esc_datamodel xmlns="http://www.cisco.com/esc/esc">
  <tenants>
    <tenant>
      <name>aws-east-1-tenant</name>
      <vim_mapping>false</vim_mapping>
      <deployments>
        <deployment>
          <name>aws-east-1-dep</name>
          <vm_group>
            <name>aws-vm-east-1</name>
            <locator>
              <vim_id>AWS_US_EAST_1</vim_id>
            </locator>
            <bootup_time>600</bootup_time>
            <recovery_wait_time>33</recovery_wait_time>
            <flavor>t2.micro</flavor>
            <image>ami-c7bfa6bd</image>
            <extensions>
              <extension>
                <name>AWS_PARAMS</name>
                <properties>
                  <property>
                    <name>key_pair_name</name>
                    <value>esc-us-east-1</value>
                  </property>
                </properties>
              </extension>
            </extensions>
            <interfaces>
              <interface>
                <nicid>0</nicid>
              </interface>
            </interfaces>
          </vm_group>
        </deployment>
      </deployments>
    </tenant>
  </tenants>
</esc_datamodel>
```

```
<network>vpc-d7eelbac</network>
<security_groups>
  <security_group>esc-sg-us-east-1</security_group>
</security_groups>
</interface>
</interfaces>
<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_collector>
      <type>ICMPping</type>
      <nicid>0</nicid>
      <poll_frequency>3</poll_frequency>
      <polling_unit>seconds</polling_unit>
      <continuous_alarm>>false</continuous_alarm>
      <monitoring_public_ip>>true</monitoring_public_ip>
    </metric_collector>
  </kpi>
</kpi_data>
<rules>
  <admin_rules>
    <rule>
      <event_name>VM_ALIVE</event_name>
      <action>ALWAYS log</action>
      <action>FALSE recover autohealing</action>
      <action>TRUE servicebooted.sh</action>
    </rule>
  </admin_rules>
</rules>
<config_data />
<scaling>
  <min_active>1</min_active>
  <max_active>1</max_active>
  <elastic>>true</elastic>
</scaling>
</vm_group>
</deployment>
</deployments>
</tenant>
</tenants>
</esc_datamodel>
```

