Deploying Virtual Network Functions on Amazon Web Services

Deploying Virtual Network Functions on Amazon Web Services

This section describes the deployment scenario for Elastic Services Controller (ESC) and the procedure to deploy VNFs on Amazon Web Services (AWS). To install ESC on AWS, see the *Cisco Elastic Services Controller Install and Upgrade Guide*.

The following AWS resources must be created on AWS before deployment:

- Amazon Machine Images (AMI)

- Key Pairs

- Elastic IPs

- Security Groups

- Network Elements (such as VPCs, subnets, ACLs, gateways, routes and so on)

See the AWS documentation to create these resources.

For information on VIM connector configuration prior to AWS deployment, see "VIM Connector Configurations for AWS".

Scenarios	Description	Resources	Advantages
Deploying VNFs on a single VIM by creating Amazon Machine Image (AMI) and regions through ESC	The *deployment data model* refers to Amazon Machine Images (AMI), flavors, AWS regions, key pairs, security groups, network interfaces and VIM projects created, and then deploys VNFs.	Amazon Machine Images (AMI), flavors, AWS regions, key pairs, security groups, network interfaces, VIM projects (specified in the locators) and Networks created through ESC.	• You can specify the VIM (to deploy VMs) that needs to be configured in ESC within a deployment. • The images and flavors can be used in multiple VNF deployments. • You can delete resources created by ESC.
Deploying VNFs on multiple VIMs by creating AMIs and regions through ESC	The *deployment data model* refers to Amazon Machine Images (AMI), flavors, AWS regions, key pairs, security groups, network interfaces and VIM projects created and then deploys VNFs.	Images, Flavors, VIM projects (specified in the locators) and Networks created through ESC.	You can specify the VIM (to deploy VMs) that needs to be configured in ESC within a deployment.

For more details, see .

Deploying VNFs on a Single or Multiple AWS Regions

You can deploy VNFs on a single or multiple AWS regions or VIMs of the same type using ESC.

Note AWS is a Virtual Infrastructure Manager (VIM) for ESC. Further in this document, the terms AWS region and AWS VIM are used interchangeably.

To deploy VNFs on a single or multiple VIMs, you must:

- Configure the VIM connector and its credentials using the VIM connector API
- Create a tenant within ESC

A VIM connector registers the VIM to ESC. To deploy VNFs on a single or multiple AWS VIMs, you must configure the VIM connector and its credentials for each region of the VIM. You can configure a VIM connector using the VIM connector APIs. For more information, see VIM Connector Configurations for AWS.

Note A default VIM connector is not supported for AWS deployment.

ESC creates a tenant within ESC with the *vim_mapping* attribute set to false. This tenant is independent of any VIM.

```
<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>
```

For a single or multiple AWS VIM deployment, you must specify the target region for each VM group.

You must add a locator attribute to the VM group in the datamodel to enable AWS VIM deployment. The locator node consists of the following attributes:

- vim_id—the vim id of the target VIM. ESC defines the vim_id and maps it to the vim_connector id. The vim connector must exist before deploying to the VIM specified by the vim_id.

- vim_project—the tenant name created in the target VIM. This is an out-of-band tenant or project existing in OpenStack.

- vim_region—the AWS region in which the VM groups are deployed. This is optional. If the vim region is not specified, then the VMs are deployed in the aws_default_region specified in the VIM connector.

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

If the vim region is not specified,

```
<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>
```

After configuring the VIM connectors and locators, you must pass certain resources as extensions to the deployment. In the example below, the elastic IP, key pair and source destination are passed as extensions to the AWS deployment.

```
<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>
```

A sample AWS deployment is as follows:

```
<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>
                                <network>vpc-d7ee1bac</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>
```

```xml
                                <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>
```