Deploying VNFs Using ESC Portal

Deploying Virtual Network Functions Using ESC Portal (OpenStack Only)

You can use the ESC portal to deploy a single VNF or multiple VNFs together by deploying a datamodel XML file. You can use the ESC portal to deploy a single VNF or multiple VNFs together either by:

Procedure

Step 1 Deploying using a file—You can upload an existing datamodel file.

Step 2 Creating and deploying a datamodel using a form—A new deployment datamodel is created by filling all the appropriate fields in the ESC portal. After you create a datamodel using the ESC portal, you can either export a deployment datamodel from the ESC portal or deploy the datamodel.

The following sections explain how to deploy VNFs using the ESC portal.

Deploy Using a File (Deployment Data model)

An existing deployment data model is used to deploy VNFs. The deployment data model is preconfigured with the number of VNFs and other specifications. It is either uploaded by locating the deployment data model or you can drag and drop the existing deployment data model. The drag and drop feature allows you to grab an existing deployment data model and to reuse it by dragging the file and dropping it off to the deployment table.

Note Only XML files are accepted.

Procedure

Step 1   Choose **Deployments**.

Step 2   Drag and drop your file to the Deployments table, or click Upload XML on the table toolbar to browse and select the file.

             Note   The drag and drop feature executes a REST call as of now and does not execute NETCONF calls.

Deploying Using a Form

To create a new deployment template, do the following:

Procedure

Step 1   Choose **Deployments**.

Step 2   Click + to deploy using a form.

Step 3   Enter a **Deployment name**.

Step 4   Select the tenant name.

Step 5   In the **General** tab, enter the appropriate values for the fields.

         a) Click **Enable Smart Licensing** to enable smart licensing.

         b) Click **Enable Intragroup Rules** to enable intragroup rules and select **Affinity** to enable affinity rules.

           For more information on intragroup affinity rules, see Affinity and Anti-Affinity Rules.

Step 6   (Optional): Click the **Add VNF Intergroup Rule** tab to select VNFs for which you the affinity rules to be applicable.

         For more information on intergroup affinity rules, see Affinity and Anti-Affinity Rules.

Step 7   To specify the parameters that ESC will utilize to heal the VNFs when there is a failure, click the **Recovery** tab.

         For more information on recovery or healing, see Healing Virtual Network Functions.

Step 8   To specify the number of interfaces and properties for each interface, click the **Interfaces** tab. The order of the interfaces specified here does not correspond to the order of the interfaces in the VM.

         a) Click **Add Interface** to add interfaces.

Step 9   To specify the number of instances of a particular type of VM that needs to be instantiated and to enable elastic scale in and scale out, click the **Scaling** tab.

Step 10   To specify the monitoring rules that will be used to configure the monitor module within ESC, click the **Monitoring** tab.

         For more information on monitoring, see Monitoring Virtual Network Functions.

Step 11   In the **Config Data** tab, enter the appropriate values for the fields.

Step 12   Do one of the following:

- To export the datamodel to a XML file, click **Export Template**.

- To deploy the datamodel, click **Deploy Template**.

Deploying VNFs on VMware vCenter using ESC Portal

The ESC portal allows you deploy a single VNF or multiple VNFs together. An existing deployment data model is either uploaded through the portal, or a new deployment data model is created. A new deployment data model is created by filling all the appropriate fields in the ESC portal. ESC also allows you to export a deployment data model from the portal. The following section explains multiple ways to deploy VNFs using the ESC portal.

The following sections explain how to deploy VNFs using the ESC portal.

Procedure

Step 1	Deploy using a file.
Step 2	Deploy using a form.

Deploy Using a File (Deployment Data model)

An existing deployment data model is used to deploy VNFs. The deployment data model is preconfigured with the number of VNFs and other specifications. It is either uploaded by locating the deployment data model or you can drag and drop the existing deployment data model. The drag and drop feature allows you to grab an existing deployment data model and to reuse it by dragging the file and dropping it off to the deployment table.

Note   Only XML files are accepted.

Procedure

Step 1	Choose **Deployments**.
Step 2	Drag and drop your file to the Deployments table, or click Upload XML on the table toolbar to browse and select the file.

Note   The drag and drop feature executes a REST call as of now and does not execute NETCONF calls.

Deploying Using a Form

To create a new deployment template, do the following:

> **Note** Click **Export Template** to export a deployment data model.

Procedure

Step 1 Choose **Deployments**.

Step 2 Click + to deploy using a form.

Step 3 Enter a **Deployment name**.

Step 4 From the **Datacenter** drop-down list, choose a datacenter on which you want to deploy the VNF.

For more information on virtual datacenter, see Deploying Virtual Network Functions on VMware vCenter .

Step 5 In the **General** tab, enter the appropriate values for the fields.

a) In the **Placement** field, select the **Cluster** or **Host** radio button .

- **Cluster**—Choose the name of a cluster to deploy a VNF in the same cluster.

- **Host**— Choose a host to deploy a VNF in the same host.

- **Datastore**— Choose a datastore for the selected cluster.

- **Image** Choose an image.

Step 6 Click **Enable Smart Licensing** to enable smart licensing.

Step 7 Click **Enable Intragroup Rules** to enable intragroup rules.

a) From the **Type** drop-down list, choose **Affinity** or **Anti-Affinity** to enable affinity or anti-affinity rules.

For more information on intragroup affinity rules, Affinity and Anti-Affinity Rules.

Step 8 (Optional) Click the **Add VNF Intergroup Rule** tab to select VNFs for which you want the affinity or anti-affinity rules to be applicable.

For more information on intergroup affinity rules, see Affinity and Anti-Affinity Rules.

Step 9 To specify the parameters that ESC will utilize to heal the VNFs when there is a failure, click the **Recovery** tab.

For more information on recovery or healing, see Healing Virtual Network Functions.

Step 10 To specify the number of interfaces and properties for each interface, click the **Interfaces** tab. The order of the interfaces specified here does not correspond to the order of the interfaces in the VM.

a) Click **Add Interface** to add interfaces.

Step 11 To specify the number of instances of a particular type of VM that needs to be instantiated and to elastic scale in and scale out, click the **Scaling** tab.

a) Click **Add Static IP Pool** to add a static IP pool.

Step 12   To specify the monitoring rules that will be used to configure the monitor module within ESC, click the **Monitoring** tab.

For more information on monitoring, see Monitoring Virtual Network Functions.

Step 13   In the **Config Data** tab, enter the appropriate values for the fields.

Step 14   (Optional) In the **OVF Settings** tab, enter the appropriate values for the fields.

    a) Click **Add OVF Property** to add a list of OVF properties.

Deploying Virtual Network Functions Using a Deployment Template

You can now deploy VNFs by uploading a preconfigured deployment template through the ESC portal.

1. Navigate to **System** > **Deployment Templates**

2. Click **Upload XML**.

   You can drag and drop, or choose a preconfigured deployment template (dep.xml) and click **Confirm**. The deployment template appears in the table.

3. Select the uploaded deployment template, and Click **Deploy from Template**.

4. The deployment name and tenant name are added from the uploaded template. Modify the fields if necessary, or click **Create** to create the template.

5. A success message appears on the screen. Click **Ok**.

The new deployment template appears in the Deployments view.

Preconfigured template

You can make changes to an existing dep.xml to use as a preconfigured template. You must make the following changes to the datamodel:

- Use esc_datamodel_template tag instead of esc_datamodel.

- The esc_datamodel_template name property is unique and must be specified to identify the template.

- *param_key* is used by the portal to identify customizable values. This is a required field. This key is unique, but can appear multiple times in the template.

- *prompt* shows the input value that needs to be added by the user. This is a required field. If the prompt is different for the same param_key specified elsewhere in the document, the first prompt is used.

- *core*, is the default value, which can be left blank.

- *required* specifies if the user must enter this value. This is an optional field. The default value is true.

- *range* validates the number field. This is an optional field.

Sample preconfigured template:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```xml
<esc_datamodel_template xmlns="http://www.cisco.com/esc/esc" name="VPC Template 1">
   <tenants>
      <tenant>
         <name param_key="tenant_name" prompt="Tenant Name">core</name>
         <managed_resource>false</managed_resource>
         <deployments>
            <deployment>
               <name param_key="dep_name" prompt="Deployment
Name">vnfd3-deployment-1.0.0-1</name>
               <policies>
                  <placement>
                     <target_vm_group_ref>c2</target_vm_group_ref>
                     <type>anti_affinity</type>
                     <enforcement>strict</enforcement>
                     <vm_group_ref>c1</vm_group_ref>
                  </placement>
               </policies>
            </deployment>
         </deployments>
      </tenant>
   </tenants>
</esc_datamodel_template>
```