Scaling Virtual Network Functions

Scaling Overview

ESC is capable of elastically scaling the service. It can be configured to do both scale in and scale out automatically. The scaling is achieved using KPI, rules and actions. These are configured during deployment. The KPI define the event name and threshold. The rules define action to trigger scale out and scale in.

For information on KPIs, Rules and Metrics, see KPIs, Rules and Metrics.

Scale In and Scale Out of VMs

Scaling workflow begins after successful deployment of a VNF. VMs are configured to monitor attributes such as CPU load, memory usage, and so on, which form the KPI data in the data model. If for any attributes, KPI reaches its threshold, based on the action defined, scale in and scale out is performed.

- During scale out, if the number of VMs is less than maximum active, a new VM deployment is triggered.
- During scale in, if the number of VMs is greater than the minimum active, the VM will be undeployed.

Note If the VM is deployed and did not receive the VM alive event, then recovery will be triggered. Any error during undeployment will be notified to the northbound user.

In the scaling section of the datamodel, the minimum and maximum values are configured. The min_active defines the number of VMs deployed. The max_active defines the number of maximum VMs that can be deployed. For example, if a VNF is deployed with a minimum 2 VMs and a maximum of 100 VMs, the below xml will define scaling under each VM group.

If the active VM was configured using a static IP address, the scaled out VMs must be assigned a static IP address. During deployment, a list of static IP addresses must be specified. The following example explains how to create a static IP pool:

```
 <scaling>
   <min_active>1</min_active>
   <max_active>2</max_active>
   <elastic>true</elastic>
   <static_ip_address_pool>
      <network>1234-5678-9123</network>
      <gateway>10.86.22.1</gateway>
      <netmask>255.255.255.0</netmask>
      <ip_address>10.86.22.227</ip_address>
      <ip_address>10.86.22.228</ip_address>
   </static_ip_address_pool>
</scaling>
```

The following example explains the method of detecting the CPU load in the KPI data section.

```
<?xml version="1.0" encoding="UTF-8"?>
<kpi>
  <event_name>VM_OVERLOADED</event_name>
  <metric_value>70</metric_value>
  <metric_cond>GT</metric_cond>
  <metric_type>UINT32</metric_type>
  <metric_occurrences_true>2</metric_occurrences_true>
  <metric_occurrences_false>4</metric_occurrences_false>
  <metric_collector>
    <type>CPU_LOAD_1</type>
    <nicid>0</nicid>
    <poll_frequency>3</poll_frequency>
    <polling_unit>seconds</polling_unit>
    <continuous_alarm>false</continuous_alarm>
  </metric_collector>
</kpi>
<kpi>
   <event_name>VM_UNDERLOADED</event_name>
   <metric_value>40</metric_value>
   <metric_cond>LT</metric_cond>
   <metric_type>UINT32</metric_type>
   <metric_occurrences_true>2</metric_occurrences_true>
   <metric_occurrences_false>4</metric_occurrences_false>
   <metric_collector>
      <type>CPU_LOAD_1</type>
      <nicid>0</nicid>
      <poll_frequency>3</poll_frequency>
      <polling_unit>seconds</polling_unit>
      <continuous_alarm>false</continuous_alarm>
   </metric_collector>
</kpi>
```

KPI rules are as follows:

```
<rule>
   <event_name>VM_OVERLOADED</event_name>
   <action>ALWAYS log</action>
   <action>TRUE servicescaleup.sh</action>
</rule>
<rule>
   <event_name>VM_UNDERLOADED</event_name>
   <action>ALWAYS log</action>
   <action>TRUE servicescaledown.sh</action>
</rule>
```

For information on scaling VNFs using ETSI API, see the *Cisco Elastic Services Controller NFV MANO Guide*.

Consistent Ordering of Resources for Scaling

ESC enables specifying resources such as ip address, mac address or day 0 configuration variables in a consistent manner in the deployment data model.

During manual and autoscaling, ESC allocates and deallocates the static IP address pool in the deployment data model in a consistent manner.

For example:

```
<scaling>
  <min_active>3</min_active>
  <max_active>6</max_active>
  <static_ip_address_pool>
    <network>jenkins-internal-vnf-net-1</network>
    <ip_address>192.168.15.3</ip_address>
    <ip_address>192.168.15.111</ip_address>
    <ip_address>192.168.15.22</ip_address>
    <ip_address>192.168.15.5</ip_address>
    <ip_address>192.168.15.4</ip_address>
    <ip_address>192.168.15.222</ip_address>
  </static_ip_address_pool>
</scaling>
```

- Manual Scaling—ESC allocates the IP addresses in the order available in the static IP pool during scale-out. During scale in, the IP addresses are released in the last in first out order.
- Autoscaling—Autoscaling uses SNMP events to indicate overload and underload of the VNFs. The overload event causes ESC to scale out, and allocates the first free IP address in the static IP pool from the order listed in the deployment data model. During scale-in, ESC deallocates the IP address, and the IP address is free for future scaling events.

For more information on day 0 configuration, ip address in the deployment data model, see Deployment Parameters.

Scaling Notifications and Events

The scaling notifications are sent to the northbound users. The notification includes status message and other details to identify the service that is undergoing scaling. Below is the list of notifications:

```
VM_SCALE_OUT_INIT
VM_SCALE_OUT_DEPLOYED
VM_SCALE_OUT_COMPLETE
VM_SCALE_IN_INIT
VM_SCALE_IN_COMPLETE
```

The following table lists the scaling scenarios and the notifications that are generated:

Scenarios	Notifications

Scale Out	ESC deploys VMs and sets KPI\Monitors and all VM Alives received. The following NETCONF notification is triggered.
	`<type>SERVICE_ALIVE</type>`
	`<status>SUCCESS</status>`
	When ESC receives a VM_OVERLOADED event, the following NetConf notification is triggered:
	`<type> VM_SCALE_OUT_INIT</type>`
	`<status>SUCCESS</status>`
	ESC checks if the max limit is reached, if not, it deploys a new VM.
	`<type> VM_SCALE_OUT_DEPLOYED</type>`
	`<status>SUCCESS</status>`
	Once the deployment is complete, the following Netconf Notification is sent,
	`<type>VM_SCALE_OUT_COMPLETE</type>`
	`<status>SUCCESS</status>`

Scale In	ESC deploys VMs and sets KPI\Monitors and all VM Alives received.
	Netconf Notification Sent
	`<type>SERVICE_ALIVE</type>`
	`<status>SUCCESS</status>`
	When ESC receives a VM_UNDERLOADED event, the following NetConf notification is triggered
	`<type> VM_SCALE_IN_INIT</type>`
	`<status>SUCCESS</status>`
	ESC check if number of VM is more than minimum active limit, if so, it undeploys one of the VM after undeployment is complete, Netconf Notification Sent.
	`<type>VM_SCALE_IN_COMPLETE</type>`
	`<status>SUCCESS</status>`

For all the error scenarios, the notification will be sent with FAILURE status. Also status message should have the corresponding failure details.