Cisco Elastic Service Controller 2.0
NETCONF API Guide

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1.3</td>
<td>Create Network</td>
<td>23</td>
</tr>
<tr>
<td>5.1.4</td>
<td>Delete Network</td>
<td>25</td>
</tr>
<tr>
<td>5.1.5</td>
<td>Create Subnet</td>
<td>27</td>
</tr>
<tr>
<td>5.1.6</td>
<td>Delete Subnet</td>
<td>29</td>
</tr>
<tr>
<td>5.1.7</td>
<td>Create Image</td>
<td>31</td>
</tr>
<tr>
<td>5.1.8</td>
<td>Delete Image</td>
<td>33</td>
</tr>
<tr>
<td>5.1.9</td>
<td>Create Flavor</td>
<td>35</td>
</tr>
<tr>
<td>5.1.10</td>
<td>Delete Flavor</td>
<td>37</td>
</tr>
<tr>
<td>5.1.11</td>
<td>Register Service with the Service Catalog</td>
<td>39</td>
</tr>
<tr>
<td>5.1.12</td>
<td>Unregister Service from the Service Catalog</td>
<td>42</td>
</tr>
<tr>
<td>5.1.13</td>
<td>Deploy Service</td>
<td>44</td>
</tr>
<tr>
<td>5.1.14</td>
<td>Un-deploy Service</td>
<td>51</td>
</tr>
<tr>
<td>5.1.15</td>
<td>Update Service</td>
<td>53</td>
</tr>
<tr>
<td>5.2</td>
<td>Operational Workflows</td>
<td>59</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Service Operations</td>
<td>59</td>
</tr>
<tr>
<td>5.2.2</td>
<td>VM operations</td>
<td>68</td>
</tr>
<tr>
<td>6.0</td>
<td>ESC TRIGGERED WORKFLOWS</td>
<td>76</td>
</tr>
<tr>
<td>6.1</td>
<td>Recovery (Healing) Workflow</td>
<td>77</td>
</tr>
<tr>
<td>6.2</td>
<td>Scale Out/Scale in Work Flows</td>
<td>79</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Scale Out Work Flow</td>
<td>79</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Scale In Work Flow</td>
<td>81</td>
</tr>
</tbody>
</table>
1 Introduction

The NETCONF API of ESC is to let the northbound clients communicate with ESC using the NETCONF protocol.

1.1 Using NETCONF
The easiest way to interact with the NETCONF agent in ConfD is to use a small python based program called netconf-console that ships with ConfD. In an ESC 1.0 VM, the netconf-console is located in:

/opt/confd-5.2.1/bin/netconf-console

There are the following operations supported by using netconf-console:

1.1.1 Get configuration
# netconf-console --user=admin --password=admin --get-config -x "'/esc_datamodel"

1.1.2 Get non-configurable data
# netconf-console --user=admin --password=admin --get -x "'/esc_datamodel/opdata"

1.1.3 Subscribe ESC notifications
# netconf-console --user=admin --password=admin --create-subscription=escEvent

1.1.4 Make RPC calls
# netconf-console --user=admin --password=admin --rpc rpc.xml

1.1.5 Push in configurations
# netconf-console --user=admin --password=admin --edit-config cfg.xml

1.2 ConfD CLI
ConfD has a CLI to enable user to read, write configurations and check operational data.

The path of ConfD CLI in ESC is in:

/opt/confd-5.2.1/bin/confd_cli

1.3 Change ConfD Admin Password
ConfD admin Password can be changed using ConfD CLI.

1) /opt/confd-5.2.1/bin/confd_cli -u admin

2) configure
3) set aaa authentication users user admin password EScT3st!234 (EScT3st!234 is the new password)

4) commit

### 1.4 NETCONF Protocol Operations

This table captures what are the NETCONF Protocol Operations supported by ESC.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Usage</th>
<th>Description</th>
<th>Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>close-session</td>
<td>:base</td>
<td>Terminate this session</td>
<td>Yes</td>
</tr>
<tr>
<td>commit</td>
<td>:base AND :candidate</td>
<td>Commit the contents of the &lt;candidate/&gt; configuration database to the &lt;running/&gt; configuration database</td>
<td>Yes</td>
</tr>
<tr>
<td>copy-config</td>
<td>:base</td>
<td>Copy a configuration database</td>
<td>No</td>
</tr>
<tr>
<td>create-subscription</td>
<td>:notification</td>
<td>Create a NETCONF notification subscription</td>
<td>Yes</td>
</tr>
<tr>
<td>delete-config</td>
<td>:base</td>
<td>Delete a configuration database</td>
<td>Yes</td>
</tr>
<tr>
<td>discard-changes</td>
<td>:base AND :candidate</td>
<td>Clear all changes from the &lt;candidate/&gt; configuration database and make it match the &lt;running/&gt; configuration database</td>
<td>No</td>
</tr>
<tr>
<td>edit-config</td>
<td>:base</td>
<td>Modify a configuration database</td>
<td>Yes</td>
</tr>
<tr>
<td>get</td>
<td>:base</td>
<td>Retrieve data from the running configuration database and/or device statistics</td>
<td>Yes</td>
</tr>
<tr>
<td>get-config</td>
<td>:base</td>
<td>Retrieve data from the running configuration database</td>
<td>Yes</td>
</tr>
<tr>
<td>kill-session</td>
<td>:base</td>
<td>Terminate another session</td>
<td>No</td>
</tr>
<tr>
<td>lock</td>
<td>:base</td>
<td>Lock a configuration database so only my session can write</td>
<td>Yes</td>
</tr>
<tr>
<td>unlock</td>
<td>:base</td>
<td>Unlock a configuration database so any session can write</td>
<td>Yes</td>
</tr>
<tr>
<td>validate</td>
<td>:base AND :validate</td>
<td>Validate the entire contents of a configuration database</td>
<td>No</td>
</tr>
</tbody>
</table>

### 1.5 NETCONF Capabilities

This table captures what are the NETCONF capabilities supported by ESC.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>:candidate</td>
<td>The server supports the <code>&lt;candidate/&gt;</code> database. It will allow this special database to be locked, edited, saved, and unlocked. The server will also support the <code>&lt;discard-changes/&gt;</code> and basic <code>&lt;commit/&gt;</code> operations.</td>
<td>No</td>
</tr>
<tr>
<td>:confirmed-commit</td>
<td>For servers that support the :candidate capability, this additional capability will also be advertised if the server supports the 'confirmed commit' feature. <strong>This special mode requires a server to send two <code>&lt;commit/&gt;</code> RPC method requests instead of one, to save any changes to the <code>&lt;running/&gt;</code> database.</strong> If the second request does not arrive within a specified time interval, the server will automatically revert the running configuration to the previous version.</td>
<td>No</td>
</tr>
<tr>
<td>:interleave</td>
<td>The server will accept <code>&lt;rpc/&gt;</code> requests (besides <code>&lt;close-session/&gt;</code>) while notification delivery is active. The :notification capability must also be present if this capability is advertised.</td>
<td>No</td>
</tr>
<tr>
<td>:notification</td>
<td>The server supports the basic notification delivery mechanisms defined in RFC 5277, e.g., the <code>&lt;create-subscription/&gt;</code> operation will be accepted by the server. Unless the :interleave capability is also supported, only the <code>&lt;close-session/&gt;</code> operation must be supported by the server while notification delivery is active.</td>
<td>Yes</td>
</tr>
<tr>
<td>:partial-lock</td>
<td>The server supports the <code>&lt;partial-lock/&gt;</code> and <code>&lt;partial-unlock/&gt;</code> operations, defined in RFC 5717. This allows multiple independent clients to each write to a different part of the <code>&lt;running/&gt;</code> configuration at the same time.</td>
<td>No</td>
</tr>
<tr>
<td>:rollback-on-error</td>
<td>The server supports the 'rollback-on-error' value for the <code>&lt;error-option/&gt;</code> parameter to the <code>&lt;edit-config/&gt;</code> operation. If any error occurs during the requested edit operation, then the target database (usually the running configuration) will be left affected. <strong>This provides an 'all-or-nothing' edit mode for a single <code>&lt;edit-config/&gt;</code> request.</strong></td>
<td>Yes</td>
</tr>
<tr>
<td>:startup</td>
<td>The server supports the <code>&lt;startup/&gt;</code> database. It will allow the running configuration to be copied to this special database. It can also be locked, and unlocked, but a server is not required to allow it to be edited.</td>
<td>No</td>
</tr>
<tr>
<td>:url</td>
<td>The server supports the <code>&lt;url/&gt;</code> parameter value form to specify protocol operation source and target parameters. The capability URI for this feature will indicate which schemes (e.g., file, https, sftp) that the server supports within a particular URL value. The 'file' scheme allows for editable local configuration databases. The other schemes allow for remote storage of configuration databases.</td>
<td>No</td>
</tr>
<tr>
<td>:validate</td>
<td>The server supports the <code>&lt;validate/&gt;</code> operation. When this operation is requested on a target database, the server will perform some amount of parameter validation and referential integrity checking. Since the standard does not define exactly what must be validated by this operation, a client cannot really rely on it for anything useful.</td>
<td>No</td>
</tr>
</tbody>
</table>
This operation is used to validate a complete database. There is no standard way to validate a single edit request against a target database, however a non-standard set-option for the `<edit-config>` operation called **test-only** has been defined for this purpose.

| :writable-running | The server allows the client to change the running configuration directly. Either this capability or the :candidate capability will be supported by the server, but usually not both. | Yes |
| :xpath | The server fully supports the XPath 1.0 specification for filtered retrieval of configuration and other database contents. The 'type' attribute within the `<filter>` parameter for `<get>` and `<get-config>` operations may be set to 'xpath'. The 'select' attribute (which contains the XPath expression) will also be supported by the server. A server may support partial XPath retrieval filtering, but it cannot advertise the :xpath capability unless the entire XPath 1.0 specification is supported. | No |
2 Configuration Data
In the ESC Data Model, any object which can be created and deleted through Netconf is represented using the Netconf Configuration datastore. ESC contains only a single Configuration datastore, the running configuration datastore. Therefore, configuration changes, once accepted, are applied immediately to the running configuration datastore, and immediately reflect the new running configuration.

2.1 Reading Configuration Data
Configuration data can be retrieved from ESC using the Netconf get-config rpc method. A Netconf subtree filter or XPath filter should be supplied to limit the results that will be retrieved (see Netconf specification for more details).

2.1.1 Fetching all config data
The example below illustrates how to fetch all existing configuration data from ESC, which in this example includes a deployed Cisco Cloud Service Router (CSR). This request uses a fully inclusive XPath filter that requests all ESC configuration data.

2.1.1.1 Netconf Request

```
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter select=" esc_datamodel " type="xpath"/>
  </get-config>
</rpc>
```

2.1.1.2 Netconf Response

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <services>
        <service_definition>
          <name>csr-reg</name>
          <version>1.2</version>
          <vm_group>
            <name>csr-vm</name>
            <disk>
              <src>file://cisco/images/CSR_Ultra_Latest.qcow2</src>
              <disk_format>qcow2</disk_format>
              <container_format>bare</container_format>
              <serial_console>true</serial_console>
              <disk_bus>virtio</disk_bus>
            </disk>
            <vm_flavor>
              <vcpus>2</vcpus>
              <memory_mb>2048</memory_mb>
              <root_disk_mb>0</root_disk_mb>
              <ephemeral_disk_mb>0</ephemeral_disk_mb>
              <swap_disk_mb>0</swap_disk_mb>
            </vm_flavor>
          </vm_group>
        </service_definition>
      </services>
    </esc_datamodel>
  </data>
</rpc-reply>
```
<tenant>
  <name>admin</name>
</tenant>
<tenant>
  <name>vnf-tenant</name>
  <services>
    <service_definition>
      <deployment_name>csr-dep</deployment_name>
      <name>csr-reg</name>
      <version>1.2</version>
    </service_definition>
    <networks>
      <network>
        <name>csr-net</name>
        <shared>false</shared>
        <admin_state>true</admin_state>
        <subnet>
          <name>csr-subnet</name>
          <ipversion>ipv4</ipversion>
          <dhcp>true</dhcp>
          <address>10.91.90.0</address>
          <netmask>255.255.255.0</netmask>
          <gateway>10.91.90.1</gateway>
        </subnet>
      </network>
    </networks>
    <vm_group>
      <name>csr-dep</name>
      <bootup_time>600</bootup_time>
      <recovery_wait_time>0</recovery_wait_time>
    </vm_group>
  </services>
</tenant>

<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>ICMP Ping</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_OVERLOADED</event_name>
    <metric_value>1</metric_value>
    <metric_cond>GE</metric_cond>
    <metric_type>UINT32</metric_type>
    <metric_collector>
      <type>Memory</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>99</metric_value>
    <metric_cond>LE</metric_cond>
    <metric_type>UINT32</metric_type>
    <metric_collector>
      <type>Memory</type>
      <nicid>0</nicid>
      <poll_frequency>3</poll_frequency>
      <polling_unit>seconds</polling_unit>
      <continuous_alarm>false</continuous_alarm>
    </metric_collector>
  </kpi>
</kpi_data>
<type>CPU</type>
<nicid>0</nicid>
<poll_frequency>3</poll_frequency>
<polling_unit>seconds</polling_unit>
<continuous_alarm>false</continuous_alarm>

</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>
    <rule>
      <event_name>VM_OVERLOADED</event_name>
      <action>ALWAYS log</action>
    </rule>
    <rule>
      <event_name>VM_UNDERLOADED</event_name>
      <action>ALWAYS log</action>
    </rule>
  </admin_rules>
</rules>
<config_data>
  <configuration>
    <dst>iosxe_config.txt</dst>
    <file>file://cisco/images/csr_config.sh</file>
  </configuration>
</config_data>
<scaling>
  <min_active>1</min_active>
  <max_active>5</max_active>
  <elastic>true</elastic>
</scaling>
</vm_group>
</service_definition>
</services>
</tenant>
</tenants>
</esc_datamodel>
</data>
</rpc-reply>
2.1.2 Example: Fetching deployed services by network name

The example below illustrates how to fetch all deployed services that contain an interface bound to the given network ‘esc-net’, using an XPath filter. Note that in the response, only elements representing key data are shown in the ancestor elements.

2.1.2.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter type="xpath" select="esc_datamodel/tenants/tenant/services/service_definition/vm_group/interfaces/interface[network='esc-net']"/>
  </get-config>
</rpc>
```

2.1.2.2 Netconf Response

```xml
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <tenants>
        <tenant>
          <name>vnf-tenant</name>
          <services>
            <service_definition>
              <deployment_name>csr-dep</deployment_name>
              <name>csr-reg</name>
              <version>1.2</version>
              <vm_group>
                <name>csr-dep</name>
                <interfaces>
                  <interface>
                    <nicid>0</nicid>
                    <network>esc-net</network>
                  </interface>
                </interfaces>
              </vm_group>
            </service_definition>
          </services>
        </tenant>
      </tenants>
    </esc_datamodel>
  </data>
</rpc-reply>
```
2.2 Writing Configuration Data (CRUD operations)

Create, Update, or Delete operations are performed on configuration data through the standard Netconf `edit-config` request. By default, only nodes that are explicitly specified in the `edit-config` request will be changed (see Netconf specification for more details).

Most configuration changes in ESC follow a two-stage model. The first stage begins when the Netconf request is received, and ends when the response is returned to the Netconf client. If the request is validated and the configuration accepted during the first stage, then the configuration workflow transitions to the second stage, which occurs asynchronously inside ESC. The second stage involves processing the workflow and activating the change in ESC and in the VIM. These two stages are described in more detail below.

2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request

A Netconf `rpc-reply` response is returned to indicate the status of the configuration request. This response will indicate whether the configuration request was accepted or rejected by ESC.

The `rpc-reply` response will contain a status body:

- A Netconf `ok` status inside the `rpc-reply` means that the configuration request was validated and accepted by ESC. E.g.:
  
  ```xml
  <rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
  </rpc-reply>
  ```

  An `ok` status indicates that the data model has been updated with the new configuration, and the workflow for activating the configuration has been submitted for asynchronous processing.

- A Netconf `rpc-error` status inside the `rpc-reply` means that the configuration request failed validation and was rejected by ESC. E.g.:
  
  ```xml
  <rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <rpc-error>
      <error-type>application</error-type>
      <error-tag>invalid-value</error-tag>
      <error-severity>error</error-severity>
      <error-path xmlns:esc="http://www.cisco.com/esc/esc">/rpc/edit-config/config/esc_datamodel/esc_tenants/esc_tenant/esc_name</error-path>
      <error-message xml:lang="en">"" is not a valid value.</error-message>
      <error-info>
        <bad-element>name</bad-element>
      </error-info>
    </rpc-error>
  </rpc-reply>
  ```

  An `rpc-error` status indicates that the configuration in ESC remains unchanged and there is no further cleanup required in ESC or in the VIM to back-out the change. As shown above, the `rpc-error` body contains information about the error that occurred.
2.2.2 Stage 2: Activation of a new Configuration
When a configuration is accepted by ESC, the workflows for activating the configuration are submitted for asynchronous processing. These workflows may update internal states within ESC, and create, update, or delete resources in the VIM. The ESC Netconf API provides two mechanisms for communicating the outcome of a configuration activation. These two mechanisms are Event Notifications and Operational Data, and are discussed below.

2.2.2.1 Event Notifications
ESC will generate escEvent Notifications for events that occur during the activation of a new configuration. A Netconf client can subscribe for these notifications in order to monitor the progress of the activation. This is the recommended way of monitoring the status of an activation. Each escEvent notification will have a status of SUCCESS or FAILURE, to report progress during various stages of activation.

2.2.2.2 Operational Data
ESC provides operational data (also called state data in Netconf) through the Netconf interface, which reflects the current operational state of ESC and the VIM resources that are being managed by ESC. During various stages of activation, the operational data may be updated to reflect the new state.

Operational data will be updated after a configuration is successfully activated. Refer to section 5.1 for details on the impact of specific configuration changes on operational data. Refer to section 3.1 for general information on reading operational data through the Netconf interface.

2.2.2.3 Activation Failures
If a failure occurs while activating the configuration change, the operational data may be left in a state that is inconsistent with the configuration. Therefore, when a failure occurs it may be necessary for the Netconf client to revert the ESC configuration, or for an operator to clean up the VIM configuration, in order to bring ESC and the VIM back to a consistent operational state. Refer to section 5.1 for instructions on handling activation failures for specific workflows.

2.2.3 Configuration Workflows
ESC uses configuration workflows to manage configuration changes. The supported workflows are defined in detail in section 5.1. In general, a separate Netconf edit-config request should be sent for each specific configuration workflow. So for example, one service deployment would map to a single workflow, and therefore a single Netconf edit-config request.

There are some workflows however that can be combined into a single Netconf edit-config request. See the next section for more details on combining workflows.

2.2.4 Combining Workflows
ESC supports combining some workflows into a single Netconf edit-config request.
As of ESC 1.0, any number of the following workflow types can be combined into a single Netconf request:

- Tenant Creation
- Tenant Deletion
- Service Registration
- Service De-registration

Any other workflows must each be sent in a separate Netconf `edit-config` request. Combining workflows other than those specified above will result in an `rpc-error` response and a configuration rejection.
3 Operational Data

3.1 Reading Operational Data
Operational data can be retrieved from ESC using the Netconf get rpc method. A Netconf subtree filter or XPath filter should be supplied to limit the results that will be retrieved (see Netconf specification for more details).

3.1.1 Fetching all operational data
In this example, we will fetch all operational data from ESC, which includes a deployed Cisco Cloud Service Router (CSR). This request uses a fully inclusive XPath filter that requests all ESC operational data. Note that specifying “esc_datamodel/opdata” in the filter will ensure that no configuration data will be returned.

3.1.1.1 Netconf Request
```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter select="esc_datamodel/opdata" type="xpath"/>
  </get>
</rpc>
```

3.1.1.2 Netconf Response
```xml
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <opdata>
        <status>OPER_UP</status>
        <stats>
          <hostname>esc-vm</hostname>
          <os_name>Linux</os_name>
          <os_release>2.6.32-504.16.2.el6.x86_64</os_release>
          <arch>x86_64</arch>
          <uptime>98960</uptime>
          <cpu>
            <cpu_num>4</cpu_num>
          </cpu>
        </stats>
        <tenants>
          <tenant>
            <name>admin</name>
            <tenant_id>6b81858d5de047e8b650589bdc8f32e</tenant_id>
          </tenant>
          <tenant>
            <name>vnf-tenant</name>
            <tenant_id>dcd22a13cc64e388a4b8d39e6a8fa7f</tenant_id>
          </tenant>
          <networks>
            <network>
              <name>csr-net</name>
              <netid>a39af67f-5004-4f2d-8b7b-d4c804645dc3</netid>
              <shared>false</shared>
              <status>active</status>
              <subnets>
                <subnet>
                  <name>csr-subnet</name>
                  <subnetid>7d5f5f55c44e85552888888888888888</subnetid>
                </subnet>
                <subnet>
                  <name>csr-subnet</name>
                  <subnetid>de4a0ed21063</subnetid>
                </subnet>
              </subnets>
            </network>
          </networks>
        </tenants>
      </opdata>
    </esc_datamodel>
  </data>
</rpc-reply>
```
<ipversion>4</ipversion>
</subnet>
</subnets>
</networks>
<deployments>
<deployment_name>csr-dep</deployment_name>
<service_name>csr-reg</service_name>
<deployment_id>vnftenantcsr-depcsr-reg1.2</deployment_id>
<service_id>csr-reg1.2</service_id>
<vm_group>
<name>csr-vm</name>
<vm_instance>
<vm_id>b25a8634-ee2f-46b5-94d1-d2cf20ed93ea</vm_id>
<name>csr-reg64600_vnf-tenant_vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0</name>
</vm_instance>
</vm_group>
<state_machine>
<state>SERVICE_ACTIVE_STATE</state>
<vm_state_machines>
<vm_name>csr-reg64600_vnf-tenant_vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0</vm_name>
<state>VM_ALIVE_STATE</state>
</vm_state_machine>
</vm_state_machines>
</state_machine>
</deployments>
</tenant>
</tenants>
</esc_datamodel>
</data>
</rpc-reply>
4 Custom RPC Methods
The ESC Netconf interface provides some custom RPC methods for controlling operational states of deployed services and VMs.

4.1 serviceAction method
This method is used to perform service actions including stop, start, reboot, disable-monitor and enable monitor.

Refer to section 0 for ESC workflows using the serviceAction method.

4.2 vmAction method
This method is used to perform VM actions including stop, start, reboot, disable-monitor and enable monitor.

Refer to section 5.2.2 for ESC workflows using the vmAction method.

5 ESC Workflows using Netconf

5.1 Configuration Workflows

5.1.1 Create Tenant

5.1.1.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <source>
      <running/>
    </source>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <tenants>
          <tenant>
            <name>mytenant</name>
          </tenant>
        </tenants>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.1.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.
5.1.1.3 Configuration Activation

On Success:

Event notification
An escEvent of type **CREATE_TENANT** with a status of **SUCCESS** will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-05T19:38:27.71+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Tenant successfully created</status_message>
    <tenant>mytenant</tenant>
    <vm_source/>
    <vm_target/>
    <event>
      <type>CREATE_TENANT</type>
    </event>
  </escEvent>
</notification>
```

Operational data
After a tenant creation configuration has been successfully activated, the operational data for the tenant will show the name and the tenant_id:

**Netconf Request**
```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter select="esc_datamodel[opdata/tenants/tenant[name='mytenant']]" type="xpath"/>
  </get>
</rpc>
```

**Netconf Response**
```xml
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <opdata>
        <tenants>
          <tenant>
            <name>mytenant</name>
            <tenant_id>dccd22a13cc64e388a4b8d39e6a8fa7f</tenant_id>
          </tenant>
        </tenants>
      </opdata>
    </esc_datamodel>
  </data>
</rpc-reply>
```
On Failure:

*Event notification*
An escEvent of type `CREATE_TENANT` with a status of `FAILURE` will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a `FAILURE` event occurs, there may be cleanup required as described below under *Handling Activation Failures*

*Operational data*
After a tenant creation configuration has failed activation, there will be no change to the operational data.

*Handling Activation Failures*
If a `failure` occurs while activating a Create Tenant configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Delete Tenant configuration change. A Delete Tenant sent at this stage should return an `<ok>` status, but will not generate a Notification. The configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.
5.1.2 Delete Tenant

5.1.2.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <tenants>
          <tenant nc:operation="delete">
            <name>mytenant</name>
          </tenant>
        </tenants>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.2.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.2.3 Configuration Activation

On Success:

Event notification

An escEvent of type DELETE_TENANT with a status of SUCCESS will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-06T16:33:27.197+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Tenant successfully deleted</status_message>
    <tenant>mytenant</tenant>
  </escEvent>
</notification>
```
Operational data
After a tenant deletion configuration has been successfully activated, there will be no operational data for the tenant.

On Failure:

Event notification
An escEvent of type DELETE_TENANT with a status of FAILURE will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. For example, this could occur if the configuration resource already exists in the VIM, but was not created by ESC. If a FAILURE event occurs, there may be cleanup required as described below under Handling Activation Failures.

Operational data
After a tenant deletion configuration has failed activation, there still will be no operational data because by default whenever a delete configuration fails then ESC cleans up the resources from its database.

Handling Activation Failures
If a failure occurs while activating a delete tenant configuration change, there are no further actions required to take as ESC will automatically clean up all the resources from its database. Operator manual intervention may be required to clean up possible leaking VIM resources.
5.1.3 Create Network

5.1.3.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <networks>
          <network>
            <name>mgmt-net</name>
          </network>
        </networks>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.3.2 Netconf Response

An `rpc-reply` response is returned with a status of `ok` if the request was accepted for activation, or a status of `rpc-error` if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.3.3 Configuration Activation

On Success:

**Event notification**

An escEvent of type `CREATE_NETWORK` with a status of `SUCCESS` will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM. The network will be created as a shared network.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Network creation completed successfully</status_message>
    <network>SystemAdminTenantIdmgmt-net</network>
    <tenant>SystemAdminTenantId</tenant>
    </escEvent>
</notification>
```
On Failure:

Event notification
An escEvent of type `CREATE_NETWORK` with a status of `FAILURE` will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a FAILURE event occurs, there may be cleanup required as described below under *Handling Activation Failures*.

Handling Activation Failures
If a *failure* occurs while activating a Create Network configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Delete Network configuration change. The configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.
5.1.4 Delete Network

5.1.4.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <networks>
          <network nc:operation="delete">
            <name>mgmt-net</name>
          </network>
        </networks>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.4.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.4.3 Configuration Activation

On Success:

**Event notification**

An escEvent of type DELETE_NETWORK with a status of SUCCESS will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T20:05:45.824+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>OpenStack Driver: Network successfully deleted</status_message>
    <network>SystemAdminTenantIdmgmt-net</network>
    <tenant>SystemAdminTenantId</tenant>
    <vm_source>
    </vm_source>
    <vm_target>
    </vm_target>
    <event>
      <type>DELETE_NETWORK</type>
    </event>
  </escEvent>
</notification>
```
On Failure:

**Event notification**
An escEvent of type `DELETE_NETWORK` with a status of `FAILURE` will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. For example, this could occur if the configuration resource was already deleted in the VIM, and was not done from ESC. If a FAILURE event occurs, there may be cleanup required as described below under *Handling Activation Failures*.

**Handling Activation Failures**
If a failure occurs while activating a delete network configuration change, there are no further actions required to take as ESC will automatically clean up all the resources from its database. Operator manual intervention may be required to clean up possible leaking VIM resources.
5.1.5 Create Subnet

5.1.5.1 Netconf Request

```
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns:ns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <networks>
          <network>
            <name>mgmt-net</name>
            <subnet>
              <name>mgmt-net-subnet</name>
              <ipversion>ipv4</ipversion>
              <dhcp>false</dhcp>
              <address>10.20.0.0</address>
              <gateway>10.20.0.1</gateway>
              <netmask>255.255.255.0</netmask>
            </subnet>
          </network>
        </networks>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

For creating the subnet, the network `mgmt-net` in this example, must already be created in a separate workflow before applying this configuration. Since this combination of creation of network and subnet in same workflow is not supported as mentioned in section 3.2.4.

5.1.5.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.5.3 Configuration Activation

On Success:

Event notification

An escEvent of type `CREATE_SUBNET` with a status of SUCCESS will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T20:24:32.57+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Subnet creation completed successfully, subnet ID: [SystemAdminTenantIdmgmt-netmgmt-net-subnet]</status_message>
  </escEvent>
</notification>
```
On Failure:

**Event notification**

An escEvent of type `CREATE_SUBNET` with a status of `FAILURE` will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a `FAILURE` event occurs, there may be cleanup required as described below under *Handling Activation Failures*.

**Handling Activation Failures**

If a `failure` occurs while activating a Create Subnet configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding `Delete Subnet` configuration change. A `Delete Subnet` sent at this stage should return an `<ok>` status, and will also generate a `DELETE_SUBNET` Success Notification. The configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.
5.1.6 Delete Subnet

5.1.6.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <target>
            <running/>
        </target>
        <config>
            <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
                <networks>
                    <network>
                        <name>mgmt-net</name>
                        <subnet nc:operation="delete">
                            <name>mgmt-net-subnet</name>
                        </subnet>
                    </network>
                </networks>
            </esc_datamodel>
        </config>
    </edit-config>
</rpc>
```

5.1.6.2 Netconf Response

An `rpc-reply` response is returned with a status of `ok` if the request was accepted for activation, or a status of `rpc-error` if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.6.3 Configuration Activation

On Success:

**Event notification**

An escEvent of type `DELETE_SUBNET` with a status of `SUCCESS` will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2015-05-19T21:57:06.526+00:00</eventTime>
    <escEvent xmlns="http://www.cisco.com/esc/esc">
        <status>SUCCESS</status>
        <status_message>OpenStack Driver: Subnet successfully deleted</status_message>
        <network>SystemAdminTenantIdmgmt-net</network>
        <subnet>SystemAdminTenantIdmgmt-net-subnet</subnet>
        <tenant>SystemAdminTenantId</tenant>
        <vm_source></vm_source>
        <vm_target></vm_target>
        <event>
            <type>DELETE_SUBNET</type>
        </event>
    </escEvent>
</notification>
```
On Failure:

*Event notification*
An escEvent of type `DELETE_SUBNET` with a status of `FAILURE` will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. For example, this could occur if the configuration resource was already deleted from the VIM and not from the ESC. If a FAILURE event occurs, there may be cleanup required as described below under *Handling Activation Failures*.

*Handling Activation Failures*
If a failure occurs while activating a delete subnet configuration change, there are no further actions required to take as ESC will automatically clean up all the resources from its database. Operator manual intervention may be required to clean up possible leaking VIM resources.
5.1.7 Create Image

5.1.7.1 Netconf Request
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
     <edit-config xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
        <target>
          <running/>
        </target>
        <config>
          <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
            <images>
              <image>
                <name>cirros-image-indep</name>
                <src>file://cisco/images/cirros-0.3.3-x86_64-disk.img</src>
                <disk_format>qcow2</disk_format>
                <container_format>bare</container_format>
                <serial_console>true</serial_console>
                <disk_bus>virtio</disk_bus>
              </image>
            </images>
          </esc_datamodel>
        </config>
      </edit-config>
    </rpc>

5.1.7.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.7.3 Configuration Activation
On Success:

Event notification
An escEvent of type CREATE_IMAGE with a status of SUCCESS will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"
               <escEvent xmlns="http://www.cisco.com/esc/esc">
                 <status>SUCCESS</status>
                 <status_message>Image creation completed successfully.</status_message>
                 <image>cirros-image-indep</image>
               </escEvent>
            </notification>
**Operational data**

After an image creation configuration has been successfully activated, the operational data for the image will show the name, image_id and visibility of the image:

**Netconf Request**

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter select="esc_datamodel/opdata/images/image[name='cirros-image-indep']"/>
  </get>
</rpc>
```

**Netconf Response**

```xml
<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <opdata>
        <images>
          <image>
            <name>cirros-image-indep</name>
            <image_id>660088a1-c38a-46b4-9dce-5b7f543ac137</image_id>
            <public>true</public>
          </image>
        </images>
      </opdata>
    </esc_datamodel>
  </data>
</rpc-reply>
```

On Failure:

**Event notification**

An escEvent of type **CREATE_IMAGE** with a status of **FAILURE** will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a **FAILURE** event occurs, there may be cleanup required as described below under **Handling Activation Failures**

**Operational data**

After an image creation configuration has failed activation, there will be no change to the operational data.

**Handling Activation Failures**

If a **failure** occurs while activating a Create Image configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Delete Image configuration change. The configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.
5.1.8 Delete Image

5.1.8.1 Netconf Request

```
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config
    target="running">
    <target>
      <running/>
    </target>
    <config
      xmlns="http://www.cisco.com/esc/esc">
      <esc_datamodel
        xmlns="http://www.cisco.com/esc/esc">
        <image
          nc:operation="delete">
          <name>cirros-image-indep</name>
        </image>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.8.2 Netconf Response

An **rpc-reply** response is returned with a status of **ok** if the request was accepted for activation, or a status of **rpc-error** if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.8.3 Configuration Activation

On Success:

**Event notification**

An escEvent of type **DELETE_IMAGE** with a status of **SUCCESS** will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully deleted in the VIM.

```
<notification
    xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-06T16:33:27.197+00:00</eventTime>
  <escEvent
    xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Image deletion completed successfully.</status_message>
    <image>cirros-image-indep</image>
    <vm_source/>
    <vm_target/>
    <event>
      <type>DELETE_IMAGE</type>
    </event>
  </escEvent>
</notification>
```

**Operational data**

After an image deletion configuration has been successfully activated, there will be no operational data for the image.
On Failure:

Event notification
An escEvent of type DELETE_IMAGE with a status of FAILURE will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. For example, this could occur if the configuration resource was already deleted in the VIM, and was not done from ESC. If a FAILURE event occurs, there may be cleanup required as described below under Handling Activation Failures.

Operational data
After an image deletion configuration has failed activation, there still will be no operational data because by default whenever a delete configuration fails then ESC cleans up the resources from its database.

Handling Activation Failures
If a failure occurs while activating a delete image configuration change, there are no further actions required to take as ESC will automatically clean up all the resources from its database. Operator manual intervention may be required to clean up possible leaking VIM resources.
5.1.9 Create Flavor

5.1.9.1 Netconf Request

```
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <target>
    <running/>
  </target>
  <config>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <flavors>
        <flavor>
          <name>myflavor</name>
          <vcpus>1</vcpus>
          <memory_mb>512</memory_mb>
          <root_disk_mb>0</root_disk_mb>
          <ephemeral_disk_mb>0</ephemeral_disk_mb>
          <swap_disk_mb>0</swap_disk_mb>
        </flavor>
      </flavors>
    </esc_datamodel>
  </config>
</edit-config>
</rpc>
```

5.1.9.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.9.3 Configuration Activation

On Success:

Event notification

An escEvent of type CREATE_FLAVOR with a status of SUCCESS will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Flavor creation completed successfully.</status_message>
    <flavor>myflavor</flavor>
    <event>
      <type>CREATE_FLAVOR</type>
    </event>
  </escEvent>
</notification>
```
**Operational data**

After a flavor creation configuration has been successfully activated, the operational data for the flavor will show the name, flavor_id and visibility of the flavor:

**Netconf Request**

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter select=" esc_datamodel/opdata/flavors/flavor[name='myflavor']">
    </get>
  </rpc>
</rpc>
```

**Netconf Response**

```xml
<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <opdata>
        <flavors>
          <flavor>
            <name>myflavor</name>
            <flavor_id>db0209a5-cf8a-4eaa-89a1-64511519486a</flavor_id>
            <public>true</public>
          </flavor>
        </flavors>
      </opdata>
    </esc_datamodel>
  </data>
</rpc-reply>
```

**On Failure:**

**Event notification**

An escEvent of type `CREATE_FLAVOR` with a status of `FAILURE` will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a `FAILURE` event occurs, there may be cleanup required as described below under *Handling Activation Failures*.

**Operational data**

After a flavor creation configuration has failed activation, there will be no change to the operational data.

**Handling Activation Failures**

If a `failure` occurs while activating a Create Flavor configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Delete Flavor configuration change. The configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.
5.1.10 Delete Flavor

5.1.10.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <flavors>
          <flavor nc:operation="delete">
            <name>myflavor</name>
          </flavor>
        </flavors>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.10.2 Netconf Response

An **rpc-reply** response is returned with a status of **ok** if the request was accepted for activation, or a status of **rpc-error** if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.10.3 Configuration Activation

On Success:

**Event notification**

An escEvent of type **DELETE_FLAVOR** with a status of **SUCCESS** will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully deleted in the VIM.

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"
  eventTime="2015-05-06T16:33:27.197+00:00"
  status="SUCCESS">
  <status_message>Flavor deletion completed successfully.</status_message>
  <flavor>myflavor</flavor>
  <vm_source/>
  <vm_target/>
  <event>
    <type>DELETE_FLAVOR</type>
  </event>
</notification>
```

**Operational data**

After a flavor deletion configuration has been successfully activated, there will be no operational data for the flavor.
On Failure:

*Event notification*
An escEvent of type **DELETE_FLAVOR** with a status of **FAILURE** will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. For example, this could occur if the configuration resource was already deleted in the VIM, and was not done from ESC. If a FAILURE event occurs, there may be cleanup required as described below under *Handling Activation Failures*.

*Operational data*
After a flavor deletion configuration has failed activation, there still will be no operational data because by default whenever a delete configuration fails then ESC cleans up the resources from its database.

*Handling Activation Failures*
If a **failure** occurs while activating a delete flavor configuration change, there are no further actions required to take as ESC will automatically clean up all the resources from its database. Operator manual intervention may be required to clean up possible leaking VIM resources.
5.1.11 Register Service with the Service Catalog

5.1.11.1 Netconf Request

```
<edit-config xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <target>
    <running/>
  </target>
  <config>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <services>
        <service_definition>
          <name>csr-reg</name>
          <version>1.2</version>
          <vm_group>
            <name>csr-vm</name>
          </vm_group>
          <disk>
            <src>/cisco/images/csr_ultra_latest.qcow2</src>
            <disk_format>qcow2</disk_format>
            <container_format>bare</container_format>
            <serial_console>true</serial_console>
            <disk_bus>virtio</disk_bus>
          </disk>
          <vm_flavor>
            <vcpus>2</vcpus>
            <memory_mb>2048</memory_mb>
            <root_disk_mb>0</root_disk_mb>
            <ephemeral_disk_mb>0</ephemeral_disk_mb>
            <swap_disk_mb>0</swap_disk_mb>
          </vm_flavor>
        </service_definition>
      </services>
    </esc_datamodel>
  </config>
</edit-config>
```

5.1.11.2 Netconf Response

An `rpc-reply` response is returned with a status of **ok** if the request was accepted for activation, or a status of **rpc-error** if the request was rejected:

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <rpc-error>
    <error-type>application</error-type>
    <error-tag>operation-failed</error-tag>
    <error-severity>error</error-severity>
    <error-message xml:lang="en">Internal id: csr_reg1.2 is already in use with another resource type</error-message>
  </rpc-error>
</rpc-reply>
```

See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.
5.1.11.3 Configuration Activation

On Success:

Event notification

A single escEvent notification of type SERVICE_REGISTRATION with a SUCCESS status will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T19:18:46.342+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Register service completed successfully</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <svcid>5a96dcfd-a404-43fa-8b00-01bd68af5048</svcid>
    <vm_source></vm_source>
    <vm_target></vm_target>
  </escEvent>
</notification>
```

Operational data

Not applicable. There is no operational data for service registration.

On Failure:

Event notification

An escEvent notification of type SERVICE_REGISTRATION with a FAILURE status will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a FAILURE event occurs, there may be cleanup required as described below under Error! Reference source not found..

Notification example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T19:18:48.342+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>Register service failed</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <svcid>5a96dcfd-a404-43fa-8b00-01bd68af5048</svcid>
    <vm_source></vm_source>
    <vm_target></vm_target>
  </escEvent>
</notification>
```
Operational data

Not applicable. There is no operational data for service registration.

Handling Activation Failures
If a failure occurs while activating a service configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Unregister service configuration change. The Unregister service configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.
5.1.12 Unregister Service from the Service Catalog

5.1.12.1 Netconf Request

```
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <services>
          <service_definition nc:operation="delete">
            <name>csr-reg</name>
            <version>1.2</version>
          </service_definition>
        </services>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.12.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected.

See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.12.3 Configuration Activation

On Success:

**Event notification**

A single escEvent notification of type SERVICE_UNREGISTRATION with a SUCCESS status will be sent to Netconf subscribers once the configuration activation has completed. This indicates that the activation workflow is complete and the configuration resource has been successfully created in the VIM.

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T19:56:00.167+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Unregister service completed successfully</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <svcid>5a96dcfd-a404-43fa-8b00-01bd68af5048</svcid>
    <vm_source/>
    <vm_target/>
    <event>
      <type>SERVICE_UNREGISTRATION</type>
    </event>
  </escEvent>
</notification>
```
Operational data

Not applicable. There is no operational data for service registration.

On Failure:

Event notification

A single escEvent notification of type SERVICE_UNREGISTRATION with a FAILURE status will be sent to Netconf subscribers if the configuration action cannot be complete for some reason. If a FAILURE event occurs, there may be cleanup required as described below under Error! Reference source not found..

Notification example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T19:59:00.167+00:00</eventTime>
  <escEvent
    xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>Unregister service failed</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <svcid>5a96dcfd-a404-43fa-8b00-01bd68af5048</svcid>
    <vm_source/>
    <vm_target/>
  </escEvent>
</notification>
```

Operational data

Not applicable. There is no operational data for service registration.

Handling Activation Failures

If a failure occurs while activating an unregister service configuration change, there are still no further actions required to take as ESC will automatically clean up all the resources from its database. Operator manual intervention may be required to clean up possible leaking VIM resources.
5.1.13 Deploy Service

A Deploy Service configuration request may contain multiple dependent workflows, in which case it can be referred to as a Unified Deployment. These workflows are:

- Create zero or one Tenant (if it doesn’t already exist in ESC)
- Create zero or more Networks (if any networks are specified in the deployment)
- Create zero or more Subnets (if specified for any networks in the deployment)

Networks that are created as part of a Deploy Service workflow are referred to as “ephemeral networks”, and the lifecycle of these networks is bound to the lifecycle of the Deployment in ESC. An Un-deploy Service workflow will automatically delete any networks and subnets that were originally created during the Deploy Service workflow.

Tenants that are created during a Deploy Service workflow are NOT deleted when the service is undeployed.

5.1.13.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
        <target>
            <running/>
        </target>
        <config>
            <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
                <tenants>
                    <tenant>
                        <name>vnf-tenant</name>
                        <services>
                            <service_definition>
                                <deployment_name>csr-dep</deployment_name>
                                <name>csr-reg</name>
                                <version>1.2</version>
                                <networks>
                                    <network>
                                        <name>mk-yesc-net-u</name>
                                        <shared>false</shared>
                                        <admin_state>true</admin_state>
                                        <subnet>
                                            <name>mk-yesc-subnet-u</name>
                                            <ipversion>ipv4</ipversion>
                                            <dhcp>true</dhcp>
                                            <address>10.91.90.0</address>
                                            <netmask>255.255.255.0</netmask>
                                            <gateway>10.91.90.1</gateway>
                                        </subnet>
                                    </network>
                                </networks>
                            </service_definition>
                        </services>
                        <bootup_time>600</bootup_time>
                        <recovery_wait_time>0</recovery_wait_time>
                        <interfaces>
                            <interface>
                                <nicid>0</nicid>
                                <network>esc-net</network>
                            </interface>
                            <interface>
                                <nicid>1</nicid>
                                <network>mk-yesc-net-u</network>
                            </interface>
                        </interfaces>
                    </tenant>
                </tenants>
            </esc_datamodel>
        </config>
    </edit-config>
</rpc>
```
<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>
    </metric_collector>
  </kpi>
  <kpi>
    <event_name>VM_OVERLOADED</event_name>
    <metric_value>1</metric_value>
    <metric_cond>GE</metric_cond>
    <metric_type>UINT32</metric_type>
    <metric_collector>
      <type>Memory</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>99</metric_value>
    <metric_cond>LE</metric_cond>
    <metric_type>UINT32</metric_type>
    <metric_collector>
      <type>CPU</type>
      <nicid>0</nicid>
      <poll_frequency>3</poll_frequency>
      <polling_unit>seconds</polling_unit>
      <continuous_alarm>false</continuous_alarm>
    </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>
    <rule>
      <event_name>VM_OVERLOADED</event_name>
      <action>ALWAYS log</action>
    </rule>
    <rule>
      <event_name>VM_UNDERLOADED</event_name>
      <action>ALWAYS log</action>
    </rule>
  </admin_rules>
</rules>

<config_data>
  <configuration>
    <dst>iocxse_config.txt</dst>
    <file>file://cisco/images/csr_config.sh</file>
  </configuration>
</config_data>

<scaling>
  <min_active>1</min_active>
  <max_active>5</max_active>
  <elastic>true</elastic>
</scaling>
</vm_group>
5.1.13.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.13.3 Configuration Activation
Workflow Aggregation:
For each dependent workflow specified in the configuration, a separate workflow is submitted. Refer to the specific workflow description for information on configuration activation of each of these optional workflows. As required, dependent workflows will be processed in the following order:

- Section 5.1.1: Create a Tenant (zero or one)
- Section 5.1.3: Create a Network for a Tenant (zero or more)
- Section 0: Create a Subnet for a Tenant (zero or more)

Multiple workflows of the same type are executed in parallel, and a failure in any workflow results in a Deploy Service activation failure once the remaining workflows of the same type have finished.

Workflows of different types are processed in sequence, and a failure in any of these workflows will result in an immediate Deploy Service activation failure. Once all of these dependent workflows have been successfully activated, the Deploy Service workflow is activated.

On Success:
A successful Deploy Service workflow will trigger events and operational data changes for each of the dependent workflows, and for the Deploy Service workflow.

Event notification
A series of escEvent notifications with a SUCCESS status will be sent to Netconf subscribers once the individual workflows in the configuration activation have completed.

If the tenant was created, then an escEvent of type CREATE_TENANT will be sent. Refer to section 5.1.1 for details.

If any ephemeral networks were created, then an escEvent of type CREATE_NETWORK will be sent for each network. Refer to section 5.1.3 for details.
If any subnets for ephemeral networks were created, then an escEvent of type CREATE_SUBNET will be sent for each subnet. Refer to section 0 for details.

Once each VM in the deployment is successfully deployed in the VIM, an escEvent of type VM_DEPLOYED with a SUCCESS status will be sent (one per VM):

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:38:04.741+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>OpenStack Driver: VM successfully created, VM Name: [csr-reg__12605__vnf-tenant__vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0]</status_message>
    <svcname>csr-reg</svcname>
    <svccversion>1.2</svccversion>
    <depname>csr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcid>e20f8ed6-a9cd-4d30-ac5b-6c3be35e38ba</svcid>
    <depid>3e0d269a-c3a4-454a-832f-44e0528ed2c1</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source>
      <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
    </vm_source>
    <vm_target/>
    <event>
      <type>VM_DEPLOYED</type>
    </event>
  </escEvent>
</notification>
```

Once each VM in the deployment can be reached by the VM Monitor, an escEvent of type VM_ALIVE with a SUCCESS status will be sent (one per VM):

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:38:21.479+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM_Alive event received, VM ID: [csr-reg__12605__vnf-tenant__vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0]</status_message>
    <svcname>csr-reg</svcname>
    <svccversion>1.2</svccversion>
    <depname>csr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcid>e20f8ed6-a9cd-4d30-ac5b-6c3be35e38ba</svcid>
    <depid>3e0d269a-c3a4-454a-832f-44e0528ed2c1</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source>
      <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
    </vm_source>
    <vm_target/>
    <event>
      <type>VM_ALIVE</type>
    </event>
  </escEvent>
</notification>
```

Once all the VM’s in the deployment are ALIVE, an escEvent of type SERVICE_ALIVE with a SUCCESS status will be sent:

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:38:21.479+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Service Alive event received. Service Name: [csr-reg__12605__vnf-tenant__vnf-tenantcsr-depcsr-reg1.2__0__service_vm__0]</status_message>
    <svcname>csr-reg</svcname>
    <svccversion>1.2</svccversion>
    <depname>csr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcid>e20f8ed6-a9cd-4d30-ac5b-6c3be35e38ba</svcid>
    <depid>3e0d269a-c3a4-454a-832f-44e0528ed2c1</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source>
      <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
    </vm_source>
    <vm_target/>
    <event>
      <type>SERVICE_ALIVE</type>
    </event>
  </escEvent>
</notification>
```
Once the SERVICE_ALIVE event is sent, the Deploy Service configuration has been successfully activated.

**Operational data**

If the tenant was created, then it will show up in operational data. Refer to section 5.1.1 for details.

If any ephemeral networks were created, then they will show up in operational data. Refer to section 5.1.3 for details. **NOTE:** it is not possible to differentiate in operational data between regular Tenant networks, and ephemeral (Deployment) networks.

If any subnets for ephemeral networks were created, then they will show up in operational data. Refer to section 0 for details.

When the Deploy Service completes successfully, the deployment will appear in the operational data. A Netconf request can be sent using the **get** method with an appropriate filter to query the operational data of the deployment:

**Netconf Request**

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter select="esc_datamodel/opdata/tenants/tenant[name='vnf-tenant']/deployments[deployment_name='csr-dep']" type="xpath"/>
  </get>
</rpc>
```

**Netconf Response**

```xml
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">  
      <opdata>
        <tenants>
          <tenant>
            <name>vnf-tenant</name>
            <deployments>
              <deployment_name>csr-dep</deployment_name>
              <service_name>csr-reg</service_name>
              <service_version>1.2</service_version>
            </deployments>
          </tenant>
        </tenants>
      </opdata>
    </esc_datamodel>
  </data>
</rpc-reply>
```
<deployment_id>vnf-tenantcsr-depcsr-reg1.2</deployment_id>
<service_id>csr-reg1.2</service_id>
<vm_group>
<name>csr-vm</name>
<vm_instance>
<vm_id>0bee67cc-f415-4f6a-95fe-e00e5d8d7c57</vm_id>
<name>csr-reg__12605__vnf-tenant__vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0</name>
<host_id>109ec6a9b7c613e5b73942e8cc998ebd9434a17c552e5e01847e402</host_id>
<hostname>compute-host-1</hostname>
<interfaces>
@interface>
<nicid>0</nicid>
<port_id>66ba846f-c02a-4c2b-964b-a12a1607656</port_id>
<network>7af5c7df-6246-4d53-91bd-4572d938b5e4</network>
<subnet>7cb681e-3023-4420-87d8-2b10efcbe14e</subnet>
<ip_address>192.168.0.58</ip_address>
</interface>
</interfaces>
</vm_instance>
</vm_group>
<state_machine>
<state>SERVICE_ACTIVE_STATE</state>
<vm_state_machines>
<vm_state_machine>
<vm_name>csr-reg__12605__vnf-tenant__vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0</vm_name>
<state>VM_ALIVE_STATE</state>
</vm_state_machine>
</vm_state_machines>
</state_machine>
</deployments>
</tenants>
</opdata>
</esc_datamodel>
</data>
</rpc-reply>

On Failure:

**Event notification**

A series of escEvent notifications may be sent to Netconf subscribers once the individual workflows in the configuration activation have completed.

Workflows of different types in the configuration activation run in sequence. If an activation failure occurs in one of these workflows, and the appropriate escEvent with a FAILURE status is sent, then the Service Deploy workflow is aborted and no more workflows will be processed.
Multiple workflows of the same type will be processed in parallel, and a failure in any such workflow results in the Service Deploy workflow being aborted once the remaining parallel workflows have finished.

If a tenant activation was attempted, then a CREATE_TENANT escEvent with either a SUCCESS or FAILURE status will be sent describing the outcome of that workflow. Refer to section 5.1.1 for details.

If any ephemeral network activations were attempted, then a CREATE_NETWORK escEvent with either a SUCCESS or FAILURE status will be sent for each network, describing the outcome of each workflow. Refer to section 0 for details.

If any subnet activations for ephemeral networks were attempted, then a CREATE_SUBNET escEvent with either a SUCCESS or FAILURE status will be sent for each subnet, describing the outcome of each workflow. Refer to section 0 for details.

An escEvent notification of type SERVICE_ALIVE with a FAILURE status will be sent to Netconf subscribers if the configuration action cannot be completed for some reason. If a FAILURE event occurs, there may be cleanup required as described below under Handling Activation Failures.

**Operational data**

After a deploy service configuration has failed activation, there may still be operational data for any dependent workflows that completed successfully.

If the tenant was successfully created, then it will show up in operational data. Refer to section 5.1.1 for details.

If any ephemeral networks were created, then they will show up in operational data. Refer to section 5.1.3 for details. NOTE: it is not possible to differentiate in operational data between regular Tenant networks, and ephemeral (Deployment) networks.

If any subnets for ephemeral networks were created, then a CREATE_SUBNET escEvent will be sent for each subnet. Refer to section 0 for details.

There still will be operational data for the deployment itself but will have missing information.

**Handling Activation Failures**

If a FAILURE event occurs while activating a deploy service configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Undeploy Service configuration change. An Undeploy Service configuration change sent at this stage should return an <ok> status, and may generate Notifications for any resource that requires cleanup. The Deploy Service configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.

Note that a tenant will never be removed by an Undeploy Service workflow, even if it was created initially by the corresponding Deploy Service workflow.
5.1.14 Un-deploy Service
An un-deploy Service configuration request may contain multiple dependent workflows, in which case it can be referred to as a Unified un-deployment. These workflows are:

- Delete zero or more Subnets (if specified for any networks in the deployment)
- Delete zero or more Networks (if any networks are specified in the deployment)

Tenants that are created during a Deploy Service workflow are NOT deleted when the service is undeployed.

5.1.14.1 Netconf Request
```
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <tenants>
          <tenant>
            <name>vnf-tenant</name>
            <services>
              <service_definition nc:operation="delete">
                <deployment_name>csr-dep</deployment_name>
                <name>csr-reg</name>
                <version>1.2</version>
              </service_definition>
            </services>
          </tenant>
        </tenants>
      </esc_datamodel>
    </config>
  </edit-config>
</rpc>
```

5.1.14.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.14.3 Configuration Activation
Workflow Aggregation:
For each dependent delete workflow specified in the configuration, a separate workflow is submitted. Refer to the specific workflow description for information on configuration activation of each of these optional workflows. The undeployment starts with undeploying the service VMs first then it moves onto deleting the dependent resource objects and will be processed in the following order:

Section 06: Delete a Subnet for a Tenant (zero or more)

Section 5.1.34: Delete a Network for a Tenant (zero or more)
Multiple workflows of the same type are executed in parallel, and a failure in any workflow results in a Undeploy Service activation failure once the remaining workflows of the same type have finished.

Workflows of different types are processed in sequence, and a failure in any of these workflows will result in an immediate undeploy Service activation failure.

**On Success:**
A successful Undeploy Service workflow will trigger events and operational data changes for each of the dependent workflows, and for the undeploy Service workflow.

**Event notification**
A series of escEvent notifications with a SUCCESS status will be sent to Netconf subscribers once the individual workflows in the configuration activation have completed.

Once each VM in the deployment is successfully undeployed from the VIM, an escEvent of type VM_UNDEPLOYED with a SUCCESS status will be sent (one per VM):

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T20:48:23.908+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>OpenStack Driver: VM successfully deleted</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <depname>csr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcid>a4d6abc8-adaf-4e3c-a7af-382846499a58</svcid>
    <depid>62983966-ea7d-40ea-ba83-994b0b9a6357</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source><vmid>56c4a68f-e79f-4ada-9a59-7bf2b30e3e9e</vmid></vm_source>
    <hostid>c3efa32d511913f8af331959886a94d3c378501266e4fa931915a6d9</hostid>
  </escEvent>
</notification>
```

If any subnets for ephemeral networks were deleted, then an escEvent of type DELETE_SUBNET will be sent for each subnet. Refer to section 06 for details.

If any ephemeral networks were deleted, then an escEvent of type DELETE_NETWORK will be sent for each network. Refer to section 5.1.34 for details.

Once all the VMs in the deployment and dependent resource objects are undeployed and deleted respectively, an escEvent of type SERVICE_UNDEPLOYED with a SUCCESS status will be sent:

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T20:48:24.467+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>OpenStack Driver: Service successfully deleted</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <depname>csr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcid>a4d6abc8-adaf-4e3c-a7af-382846499a58</svcid>
    <depid>62983966-ea7d-40ea-ba83-994b0b9a6357</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source><vmid>56c4a68f-e79f-4ada-9a59-7bf2b30e3e9e</vmid></vm_source>
    <hostid>c3efa32d511913f8af331959886a94d3c378501266e4fa931915a6d9</hostid>
  </escEvent>
</notification>
```
Once the SERVICE_UNDEPLOYED event is sent, the undeploy Service configuration has been successfully activated.

**Operational data**
All of the resource objects and service VMs that have been deleted as part of undeployment will be removed from operational data.

**On Failure:**

**Event notification**
Workflows of different types in the configuration activation run in sequence. If an activation failure occurs in one of these workflows, the Service undeploy workflow is then aborted and no more workflows will be processed. Along with the abort, an escEvent of type SERVICE_UNDEPLOYED with a FAILURE status is sent.

**Operational data**
After an undeploy service configuration has failed activation, there will be no operational data for any dependent workflows nor the deployment itself (See Handling Activation Failures).

**Handling Activation Failures**
If a FAILURE event occurs while activating a undeploy service configuration change there is no need to take any further actions as ESC has already cleaned up its database. Operator may need to manually remove the resources from the VIM as the VIM may not have deleted them successfully.

### 5.1.15 Update Service
Update service allows you to incrementally add VNFs to a deployment. Update service requires a service registration with all of the potential VNFs that may potentially be used in the future even when they will not be used right away. Whenever a VNF is to be deployed then the min active under the desired VNF (vm_group) must be set to 1 and a configuration file has to be provided. The VNFs that are not going to be deployed must have their min active set to 0 and their interfaces to N/A as well as their configuration file set to N/A.
Refer to the deployment section to do an initial deployment before doing the update deployment. The original deployment will only have one active VNF and that can be seen in the operational data. Once update service is completed, another VNF will be added to the operational data.

5.1.15.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <running/>
    </target>
    <config>
      <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
        <tenants>
          <tenant>
            <name>vnf-tenant</name>
            <services>
              <service_definition>
                <deployment_name>csr-dep</deployment_name>
                <name>csr-reg</name>
                <version>1.2</version>
              </service_definition>
              <vm_group>
                <name>csr-vm</name>
                <bootup_time>600</bootup_time>
                <recovery_wait_time>0</recovery_wait_time>
              </vm_group>
              <interfaces>
                <interface>
                  <nicid>0</nicid>
                  <network>7af5c7df-6246-4d53-91bd-aa12a1607656</network>
                </interface>
              </interfaces>
            </services>
            <kpi_data>
              <kpi>
                <event_name>VM_ALIVE</event_name>
                <metric_value>50</metric_value>
                <metric_cond>GT</metric_cond>
                <metric_type>UINT32</metric_type>
                <metric_occurrences_true>3</metric_occurrences_true>
                <metric_occurrences_false>3</metric_occurrences_false>
              </kpi>
              <kpi_data>
                <rules>
                  <admin_rules>
                    <rule>
                      <event_name>VM_ALIVE</event_name>
                      <action>ALWAYS log</action>
                      <action>FALSE recover autohealing</action>
                    </rule>
                  </admin_rules>
                  <rules>
                    <config_data>
                      <configuration>
                        <dst>iosxe_config.txt</dst>
                      </configuration>
                    </config_data>
                    <scaling>
                      <min_active>1</min_active>
                    </scaling>
                  </rules>
                </rules>
              </kpi_data>
            </kpi_data>
          </tenant>
        </tenants>
      </config>
    </edit-config>
  </rpc>
```
<metric_occurrences_true>3</metric_occurrences_true>
<metric_occurrences_false>3</metric_occurrences_false>

<metric_collector>
  <type>ICMPPing</type>
  <nicid>0</nicid>
  <poll_frequency>15</poll_frequency>
  <polling_unit>seconds</polling_unit>
  <continuous_alarm>false</continuous_alarm>
</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>
  <configuration>
    <dst>iosxe_config.txt</dst>
    <file>N/A</file>
  </configuration>
</config_data>
<scaling>
  <min_active>0</min_active>
  <max_active>0</max_active>
  <elastic>true</elastic>
</scaling>
</vm_group>
</service_definition>
</services>
</tenant>
</tenants>
</esc_datamodel>
</config>
</edit-config>
</rpc>

5.1.15.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.1.15.3 Configuration Activation

On Success:
A successful update Service workflow will trigger events and operational data changes for the added VNF.

Event notification
The process for deploying a VM for the added VNF is the same as the deployment section including the notifications sent (see deployment section) such as a VM_DEPLOYED and VM_ALIVE escEvent with a success status.
Once all the VM’s in the deployment are deployed, an escEvent of type SERVICE_UPDATED with a SUCCESS status will be sent:

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T20:48:24.467+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Service group update completed successfully</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <depname>csr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcid>a4d6abc8-adaf-4e3c-a7af-382846499a58</svcid>
    <depid>62983966-ea7d-40ea-ba83-994b0b9a6357</depid>
    <vm_source> </vm_source>
    <vm_target> </vm_target>
    <event>
      <type>SERVICE_UPDATE</type>
    </event>
  </escEvent>
</notification>
```

Once the SERVICE_UPDATED event is sent, the update Service configuration has been successfully activated.

**Operational data**

When the update Service completes successfully, information about the newly added VNF will appear under the deployment in the operational data. A Netconf request can be sent using the `get` method with an appropriate filter to query the operational data of the updated deployment:

**Netconf Request**

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter select="esc_datamodel/opdata/tenants/tenant[name='vnf-tenant']/deployments[deployment_name='csr-dep']" type="xpath"/>
  </get>
</rpc>
```

**Netconf Response**

```xml
<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <esc_datamodel xmlns="http://www.cisco.com/esc/esc">
      <opdata>
        <tenants>
          <tenant>
            <name>adouha_new</name>
            <tenant_id>5272fcce63c10465cb8e9255f55248c</tenant_id>
            <deployments>
              <deployment_name>adouha_deployment</deployment_name>
              <service_name>adouha_regi</service_name>
              <service_version>1.2</service_version>
              <deployment_id>adouha_newadouha_depadouha_regi1.2</deployment_id>
              <service_id>adouha_regi1.2</service_id>
            </deployments>
          </tenant>
        </tenants>
      </opdata>
    </esc_datamodel>
  </data>
</rpc-reply>
```
<vm_group>
  <vm_instance>
    <name>ADOUHA_CSR</name>
    <vm_id>6d828cac-4ae2-4155-bbb3-aa8c1ec9d8ba</vm_id>
    <host_id>db373de8835b05cecf1f8129f9570bfe4c058d6ec01696e8bd42241a</host_id>
    <hostname>my-ucs-3</hostname>
    <interfaces>
      <interface>
        <nicid>0</nicid>
        <port_id>3d47b2ee-2464-4f5e-832e-b3afaf86ec3d</port_id>
        <network>7af5c7df-6246-4d53-91bd-a12a1607656</network>
        <subnet>7cb6815e-3023-4420-87d8-2b10efce14e</subnet>
        <ip_address>192.168.0.22</ip_address>
        <mac_address>fa:16:3e:46:02</mac_address>
      </interface>
    </interfaces>
  </vm_instance>
</vm_group>

<vm_group>
  <vm_instance>
    <name>ADOUHA_WSA</name>
    <vm_id>9925dbc0-5888-4a48-add0-b1babe05d6c</vm_id>
    <host_id>02d2b1e68a0eaca50b7f7a5aafe331f319e15320a4fe2ca45f4197937</host_id>
    <hostname>my-ucs-5</hostname>
    <interfaces>
      <interface>
        <nicid>0</nicid>
        <port_id>88de389e-fe73-45d9-b20d-fea5ac9560df</port_id>
        <network>7af5c7df-6246-4d53-91bd-a12a1607656</network>
        <subnet>7cb6815e-3023-4420-87d8-2b10efce14e</subnet>
        <ip_address>192.168.0.221</ip_address>
        <mac_address>fa:16:3e:10:7d:1e</mac_address>
      </interface>
    </interfaces>
  </vm_instance>
</vm_group>

<state_machine>
  <state>SERVICE_ACTIVE_STATE</state>
  <vm_state_machines>
    <vm_state_machine>
      <vm_name>adouha_reg__79421__adouha_new__adouha_newadouha_depadouha_reg1.2__0__ADOUHA_CSR__0</vm_name>
      <state>VM_ALIVE_STATE</state>
    </vm_state_machine>
    <vm_state_machine>
      <vm_name>adouha_reg__90903__adouha_new__adouha_newadouha_depadouha_reg1.2__0__ADOUHA_WSA__0</vm_name>
      <state>VM_ALIVE_STATE</state>
    </vm_state_machine>
  </vm_state_machines>
</state_machine>

</deployments>
</tenant>
</tenant>
</opdata>
</esc_datamodel>
</data>
</rpc-reply>
On Failure:

Event notification
A series of escEvents notification may be sent during an activation failure. The notifications will appear exactly the same as the deployment’s failure cause (See the deployment’s section for failure attempt). The workflow will be aborted and a final escEvent notification of type SERVICE_UPDATED with a FAILURE status is sent.

Operational data
After an update service configuration has failed activation, the operational data may or may not change depending on the timing of the failure.

Handling Activation Failures
If a FAILURE event occurs while activating an update service configuration change, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Undeploy Service configuration change. An Undeploy Service configuration change sent at this stage should return an <ok> status, and may generate Notifications for any resource that requires cleanup. The Deploy and Update Service configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved.

5.2 Operational Workflows
There are 5 operations that can be done to the entire service or deployment.

- STOP
- START
- REBOOT
- DISABLE_MONITOR
- ENABLE_MONITOR

5.2.1 Service Operations

5.2.1.1 Stop Service

5.2.1.1.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <serviceAction xmlns="http://www.cisco.com/esc/esc">
        <actionType>STOP</actionType>
        <tenantName>sampleTenant</tenantName>
        <depName>sampleDep</depName>
        <serviceName>SampleSvc</serviceName>
        <serviceVersion>1.1</serviceVersion>
      </serviceAction>
    </config>
  </edit-config>
```
5.2.1.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.1.3 Event Notifications

All VMs are stopped first. Each one succeeds with a notification:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.262+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM successfully stopped</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c66a</vmid>
      <hostid>c64e652a4561392ae20ae702eceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event>
        <type>VM_STOPPED</type>
        <event>
          <escEvent>
            </escEvent>
          </event>
        </event>
      </vm_target>
    </escEvent>
  </notification>
```

When all VMs have been stopped, there is a final notification:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Service group suspension completed successfully</status_message>
    <status_message>Service group suspension completed successfully</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c66a</vmid>
      <hostid>c64e652a4561392ae20ae702eceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event>
        <type>SERVICE_STOPPED</type>
        <event>
          <escEvent>
            </escEvent>
          </event>
        </event>
      </vm_target>
    </escEvent>
  </notification>
```
If there are errors during stopping the service, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escevent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>Service group suspension failed</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1A75-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_source/>
    <vm_target/>
    <event>
      <type>SERVICE_STOPPED</type>
    </event>
  </escevent>
</notification>
```

### 5.2.1.2 Start Service

#### 5.2.1.2.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <serviceAction xmlns="http://www.cisco.com/esc/esc">
        <actionType>START</actionType>
        <tenantName>sampleTenant</tenantName>
        <depName>sampleDep</depName>
        <serviceName>SampleSvc</serviceName>
        <serviceVersion>1.1</serviceVersion>
      </serviceAction>
    </config>
  </edit-config>
</rpc>
```

#### 5.2.1.2.2 Netconf Response

An **rpc-reply** response is returned with a status of **ok** if the request was accepted for activation, or a status of **rpc-error** if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

#### 5.2.1.2.3 Event Notifications

All VMs are started first. Each one succeeds with a notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.262+00:00</eventTime>
  <escevent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
  </escevent>
</notification>
```
<status_message>VM successfully started</status_message>
<svcname>CIRROS_HEZH</svcname>
<svccversion>1.6</svccversion>
<depname>dep</depname>
<tenant>hezh-yesc-0520</tenant>
<svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
<vm_group>CSR</vm_group>
<vm_source>
  <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</vmid>
  <hostid>c64e652a4561392ae20ae702eceaee53e417302e0d2d63e41d66a15cc</hostid>
</vm_source>
<vm_target>
</vm_target>
<event>
  <type>VM_STARTED</type>
</event>
</escEvent>
</notification>

When a VM has been started, it will be monitored using ICMP ping. When it gets alive, ESC will send the notification indicating VM is alive:

<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>
      VM_Alive event received after started, VM ID: [CIRROS_HEZH_614_hezh-yesc-0520_hezh-yesc-0520depCIRROS_HEZH1.6.0_CSR__0]
    </status_message>
  </escEvent>
</notification>

When all VMs are alive, there is a final notification:

<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.487+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Service group started successfully</status_message>
  </escEvent>
</notification>
If there are errors during starting the service, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>Service group start failed</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcrid>90044227-1475-45c2-83ff-aed132f77256</svcrid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_source>
      <vm_target>
        <event>
          <type>SERVICE_ALIVE</type>
        </event>
      </vm_target>
    </vm_source>
    <event>
      <type>SERVICE_ALIVE</type>
    </event>
  </escEvent>
</notification>
```

5.2.1.3 Reboot Service

5.2.1.3.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <serviceAction xmlns="http://www.cisco.com/esc/esc">
        <actionType>REBOOT</actionType>
        <tenantName>sampleTenant</tenantName>
        <depname>sampleDep</depname>
        <serviceName>SampleSvc</serviceName>
        <serviceVersion>1.1</serviceVersion>
      </serviceAction>
    </config>
  </edit-config>
</rpc>
```

5.2.1.3.2 Netconf Response

An `rpc-reply` response is returned with a status of `ok` if the request was accepted for activation, or a status of `rpc-error` if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.1.3.3 Event Notifications

All VMs are rebooted first. Each one succeeds with a notification:
When a VM has been rebooted, it will be monitored using ICMP ping. When it gets alive, ESC will send the notification indicating VM is alive:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM_Alive event received after rebooted, VM ID: [CIRROS_HEZH__614__hezh-yesc-0520__hezh-yesc-0520depCIRROS_HEZH1.6__0__CSR__0]</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702e6eae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <type>VM_ALIVE</type>
    </vm_target>
  </escEvent>
</notification>
```

When all VMs are alive, there is a final notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.487+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Service group rebooted successfully</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
  </escEvent>
</notification>
```
If there are errors during rebooting the service, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
    <escEvent xmlns="http://www.cisco.com/esc/esc">
        <status>FAILURE</status>
        <status_message>Service group reboot failed</status_message>
        <svcname>CIRROS_HEZH</svcname>
        <svcversion>1.6</svcversion>
        <depname>dep</depname>
        <tenant>hezh-yesc-0520</tenant>
        <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
        <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    </escEvent>
</notification>
```

### 5.2.1.4 Disable Service Monitor

#### 5.2.1.4.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <serviceAction xmlns="http://www.cisco.com/esc/esc">
                <actionType>DISABLE_MONITOR</actionType>
                <tenantName>sampleTenant</tenantName>
                <depName>sampleDep</depName>
                <serviceName>SampleSvc</serviceName>
                <serviceVersion>1.1</serviceVersion>
            </serviceAction>
        </config>
    </edit-config>
</rpc>
```

#### 5.2.1.4.2 Netconf Response

An `rpc-reply` response is returned with a status of `ok` if the request was accepted for activation, or a status of `rpc-error` if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.
5.2.1.4.3 Event Notifications

Disabling monitor for each VM succeeds with a notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.262+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Unset Monitoring VM successfully.</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vmid>5283bcab-c1c3-4699-9b8e-cf143b26c6aa</vmid>
    <hostid>c64e652a4561392ae2c0e13730e41d66a15cc</hostid>
    <vm_target>
      <type>VM_MONITOR_UNSET</type>
    </vm_target>
    </escEvent>
</notification>
```

When all VMs have done unssetting monitors, there is a final notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Service group suspension completed successfully</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <type>VM_MONITOR_UNSET</type>
    </vm_source>
    <vm_target>
      <type>SERVICE_MONITOR_UNSET</type>
    </vm_target>
    </escEvent>
</notification>
```

If there are errors during disabling the monitors of the service, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>Service group disable monitor failed</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <tenant>hezh-yesc-0520</tenant>
    </escEvent>
</notification>
```
5.2.1.5 Enable Service Monitor

<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <serviceAction xmlns="http://www.cisco.com/esc/esc">
        <actionType>ENABLE_MONITOR</actionType>
        <tenantName>sampleTenant</tenantName>
        <depName>sampleDep</depName>
        <serviceName>SampleSvc</serviceName>
        <serviceVersion>1.1</serviceVersion>
      </serviceAction>
    </config>
  </edit-config>
</rpc>

5.2.1.5.1 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.1.5.2 Event Notifications

All VMs have their monitors first. When one VM gets alive, ESC will send the notification indicating VM is alive:

<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM_Alive event received, VM ID: [CIRROS_HEZH_614__hezh-yesc-0520__hezh-yesc-0520depCIRROS_HEZH1.6_0__CSR__0]</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702eceeae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
When all VMs are alive, there is a final notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2015-05-19T23:46:36.487+00:00</eventTime>
    <escEvent xmlns="http://www.cisco.com/esc/esc">
        <status>SUCCESS</status>
        <status_message>Service group set monitor successfully</status_message>
        <svcname>CIRROS_HEZH</svcname>
        <svccversion>1.6</svccversion>
        <depname>dep</depname>
        <tenant>hezh-yesc-0520</tenant>
        <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
        <depid>187bc0b5-9d0b-4c01-8353-89454a61a6a6</depid>
    </escEvent>
</notification>
```

If there are errors during setting the monitor to the service, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
    <escEvent xmlns="http://www.cisco.com/esc/esc">
        <status>FAILURE</status>
        <status_message>Service group set monitor failed</status_message>
        <svcname>CIRROS_HEZH</svcname>
        <svccversion>1.6</svccversion>
        <depname>dep</depname>
        <tenant>hezh-yesc-0520</tenant>
        <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
        <depid>187bc0b5-9d0b-4c01-8353-89454a61a6a6</depid>
    </escEvent>
</notification>
```

5.2.2 VM operations

5.2.2.1 Stop VM

5.2.2.1.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>...<target>
```
The “force” option in the request decides if ESC considers the min_vm requirement for the service. Since stopping a VM would affect the service availability, after the VM is stopped the min_vm requirement specified in the datamodel may not be satisfied. If the “force” option is true, ESC will ignore the requirement for min_vm.

5.2.2.1.2 Netconf Response

An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.2.1.3 Event Notifications

When the VM is successfully stopped, ESC will send a notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.262+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM successfully stopped</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89434a61da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcb-clc3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702eceeae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event>
        <type>VM_STOPPED</type>
      </event>
    </vm_target>
  </escEvent>
</notification>
```

If there are errors during stopping the VM, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
</notification>
```
<eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
<escEvent xmlns="http://www.cisco.com/esc/esc">
<status>FAILURE</status>
<status_message>Service group suspension failed</status_message>
<svcname>CIRROS_HEZH</svcname>
<svcversion>1.6</svcversion>
<depname>dep</depname>
<tenant>hezh-yesc-0520</tenant>
<svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
<depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
<vm_source/>
<vm_target/>
<event>
<type>VM_STOPPED</type>
</event>
</escEvent>
</notification>

5.2.2.2 Start VM

5.2.2.2.1 Netconf Request
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
<target>
<running/>
</target>
<config>
<vmAction xmlns="http://www.cisco.com/esc/esc">
/actionType>START</actionType>
<vmName>sampleVmName</vmName>
<force>false</force>
</vmAction>
</config>
</edit-config>
</rpc>

5.2.2.2.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.2.3 Event Notifications

When the VM is successfully started, ESC will send a notification:

<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2015-05-19T21:21:00.262+00:00</eventTime>
<escEvent xmlns="http://www.cisco.com/esc/esc">
<status>SUCCESS</status>
<status_message>VM successfully started</status_message>
<svcname>CIRROS_HEZH</svcname>
<svcversion>1.6</svcversion>
<depname>dep</depname>
<tenant>hezh-yesc-0520</tenant>
<svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
<depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
<vm_source>CSR</vm_source>
When the VM has been started, it will be monitored using ICMP ping. When it gets alive, ESC will send the notification indicating VM is alive:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM_Alive event received after started, VM ID: [CIRROS_HEZH_614__hezh-yesc-0520__hezh-yesc-0520depCIRROS_HEZH1.6__0__CSR__0]</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a61da6</depid>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <svcmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</svcmid>
    <hostid>c64e652a4561392ae20ae702ec6ec53e417302e0d2d63e41d66a15cc</hostid>
  </escEvent>
</notification>
```

If there are errors during starting the VM, there will be a final notification indicating the error:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>VM start failed</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a61da6</depid>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <svcmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</svcmid>
    <hostid>c64e652a4561392ae20ae702ec6ec53e417302e0d2d63e41d66a15cc</hostid>
  </escEvent>
</notification>
```
5.2.2.3 Reboot VM

5.2.2.3.1 Netconf Request
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <vmAction xmlns="http://www.cisco.com/esc/esc">
        <actionType>REBOOT</actionType>
        <vmName>sampleVmName</vmName>
        <force>true/false</force>
      </vmAction>
    </config>
  </edit-config>
</rpc>

The “force” option in the request decides if ESC considers the min_vm requirement for the service. Since stopping a VM would affect the service availability, after the VM is rebooted the min_vm requirement specified in the datamodel may not be satisfied. If the “force” option is true, ESC will ignore the requirement for min_vm.

5.2.2.3.2 Netconf Response
An rpc-reply response is returned with a status of ok if the request was accepted for activation, or a status of rpc-error if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.2.3.3 Event Notifications

When the VM is successfully rebooted, ESC will send a notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.262+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM successfully rebooted</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depid>187bc0b5-9d0b-4c01-8353-89454a61da6</depid>
    <vm_group>CSR</vm_group>
    <vmid>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmid>
    <hostid>c64e652a4561392ae20ae702eceae53e417302e0d2d63e41d66a15cc</hostid>
  </escEvent>
</notification>
```
When the VM has been rebooted, it will be monitored using ICMP ping. When it gets alive, ESC will send the notification indicating VM is alive:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM_Alive event received after rebooted, VM ID: [CIRROS_HEZH_614_hezh-yesc-0520 hezh-yesc-0520depCIRROS_HEZH1.6_0_CSR_0]</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a61da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702eceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
    </vm_target>
    <event>
      <type>VM_ALIVE</type>
    </event>
  </escEvent>
</notification>
```

If there are errors during rebooting the VM, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T21:21:00.527+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <status_message>VM reboot failed</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a61da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
    </vm_source>
    <vm_target>
    </vm_target>
    <event>
      <type>VM_REBOOTED</type>
    </event>
  </escEvent>
</notification>
```

### 5.2.2.4 Disable VM Monitor

#### 5.2.2.4.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
```
5.2.2.4.2 Netconf Response

An **rpc-reply** response is returned with a status of **ok** if the request was accepted for activation, or a status of **rpc-error** if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.2.4.3 Event Notifications

When the monitor of the VM is disabled successfully, ESC will send a notification:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventtime>2015-05-20T00:47:06.591+00:00</eventtime>
  <esc_event xmlns="http://www.cisco.com/esc/esc">
    <status>Success</status>
    <status_message>Unset monitor completed successfully</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702e9ceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event type="VM_MONITOR_UNSET"/>
    </vm_target>
  </esc_event>
</notification>
```

If there are errors during disabling the monitor of the VM, there will be a final notification indicating the error:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventtime>2015-05-20T00:47:06.591+00:00</eventtime>
  <esc_event xmlns="http://www.cisco.com/esc/esc">
    <status>Failure</status>
    <status_message>Unset monitor failed</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702e9ceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
  </esc_event>
</notification>
```
5.2.2.5 Enable VM Monitor

5.2.2.5.1 Netconf Request

```xml
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <vmAction xmlns="http://www.cisco.com/esc/esc">
        <actionType>ENABLE_MONITOR</actionType>
        <vmName>sampleVmName</vmName>
        <force>false</force>
      </vmAction>
    </config>
  </edit-config>
</rpc>
```

5.2.2.5.2 Netconf Response

An **rpc-reply** response is returned with a status of **ok** if the request was accepted for activation, or a status of **rpc-error** if the request was rejected. See section 2.2.1 Stage 1: Acceptance or Rejection of a Configuration Request for more details.

5.2.2.5.3 Event Notifications

When the monitor of the VM is enabled successfully, ESC will send a notification:

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-20T00:50:51.601+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Set monitor completed successfully</status_message>
    <svcname>CIRROS_HEZH</svcname>
    <svcversion>1.6</svcversion>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-c1c3-4699-9bce-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702eceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <vmid>5283bcab-c1c3-4699-9bce-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702eceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_target>
  </escEvent>
</notification>
```

When the VM has been set monitor, it will be monitored using ICMP ping. When it gets alive, ESC will send the notification indicating VM is alive:
If there are errors during enabling the monitor of the VM, there will be a final notification indicating the error:

If there are errors during enabling the monitor of the VM, there will be a final notification indicating the error:

6 ESC Triggered Workflows
6.1 Recovery (Healing) Workflow

A VM recovery is triggered in the following scenarios.

- During a service deployment if VM_ALIVE event from any or all of the VMs being deployed by the service is not received before the timeout value. 
- After the service is deployed, if any of the VMs deployed goes down for some reason and there are rules defined to activate VM recovery for these cases.

Recovery Initialize

When a VM recovery is triggered, an escEvent of type VM_RECOVERY_INIT with a status of SUCCESS will be sent to Netconf subscribers. This indicates that the recovery workflow has been started. This will hard reboot the VM instance and checks if the VM comes back up. If the VM comes up successfully, an escEvent of type VM_RECOVERY_COMPLETE with a status of SUCCESS will be sent.

Attempts for recovery

If the VM did not come up after this attempt, then ESC tries to undeploy and redeploy the VM and corresponding escEvent of type VM_RECOVERY_UNDEPLOYED and VM_RECOVERY_DEPLOYED will be sent. If the undeploy and redeploy also fails to get the VM back up, this undeploy and redeploy cycle will be retried to a maximum value of “vm_recovery_retries_max” times which is configurable.

Recovery complete

If the VM comes alive at any attempt during the undeploy and deploy, an escEvent of type VM_RECOVERY_COMPLETE with a status of SUCCESS will be sent. But if the VM does not come up after all the redeploy attempts, then an escEvent of type VM_RECOVERY_COMPLETE with a status of FAILURE will be sent.

The various event notifications generated during VM recovery are given below.

Success Event notifications

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:38:45.107+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>Recovery event for VM [crr-reg__12605__vnf-tenant__vnf-tenantcsr-depcrr-reg1.2__0__crr-vm__0] triggered. Processing Auto healing. Proceeding with Recovery. </status_message>
    <svcname>crr-reg </svcname>
    <svcsversion>1.2</svcsversion>
    <depname>crr-dep</depname>
    <tenant>vnf-tenant</tenant>
    <svcident2f8ed6-a9cd-4d30-ac5b-6c3be35e38ba></svcident2f8ed6-a9cd-4d30-ac5b-6c3be35e38ba>
    <depid3ed269a-c3a4-454a-832f-44e0528ed2c1></depid3ed269a-c3a4-454a-832f-44e0528ed2c1>
    <vmgroup>csr-vnc</vmgroup>
    <vm>crr-vm</vm>
    <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
    <hostid>a2f7615e78330dd28697588ddfb9516504b19642f53d39d7c6c8f6ab</hostid>
</vm_source>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:38:52.073+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM [csr-reg_12605__vnf-tenant_vnf-tenantcsr-depcsr-reg1.2_0__csr-vm__0] has been undeployed.</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <depcname>csr-dep</depcname>
    <tenant>vnf-tenant</tenant>
    <svcid>e20f8ed6-a9cd-4d30-ac5b-6c3be35e38ba</svcid>
    <depid>3e0d269a-c3a4-454a-832f-44e0528ed2c1</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source>
      <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
      <hostid>a2f7615e78330dd28697588ddf9516504b19642f53d39d7cff4f6ab</hostid>
    </vm_source>
    <vm_target/>
    <event>
      <type>VM_RECOVERY_UNDEPLOYED</type>
    </event>
  </escEvent>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:39:03.36+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM [csr-reg_12605__vnf-tenant_vnf-tenantcsr-depcsr-reg1.2_0__csr-vm__0] has been deployed as part of recovery.</status_message>
    <svcname>csr-reg</svcname>
    <svcversion>1.2</svcversion>
    <depcname>csr-dep</depcname>
    <tenant>vnf-tenant</tenant>
    <svcid>e20f8ed6-a9cd-4d30-ac5b-6c3be35e38ba</svcid>
    <depid>3e0d269a-c3a4-454a-832f-44e0528ed2c1</depid>
    <vm_group>csr-vm</vm_group>
    <vm_source>
      <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
      <hostid>a2f7615e78330dd28697588ddf9516504b19642f53d39d7cff4f6ab</hostid>
    </vm_source>
    <vm_target>
      <vmid>250e515b-978f-4358-b455-124b72608ba2</vmid>
      <hostid>a2f7615e78330dd28697588ddf9516504b19642f53d39d7cff4f6ab</hostid>
    </vm_target>
    <event>
      <type>VM_RECOVERY_DEPLOYED</type>
    </event>
  </escEvent>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-04-28T23:39:21.125+00:00</eventTime>
  <escEvent/>
</notification>
<escEvent xmlns="http://www.cisco.com/esc/esc">
  <status>SUCCESS</status>
  <status_message>Successfully recovered VM [csr-reg__12605__vnf-tenant__vnf-tenantcsr-depcsr-reg1.2__0__csr-vm__0].</status_message>
  <svcname>csr-reg</svcname>
  <svcversion>1.2</svcversion>
  <depname>vnf-tenant</depname>
  <tenant>vnf-tenant</tenant>
  <svcid>e20f8ed6-a9cd-4d30-ac5b-6c3be35e38ba</svcid>
  <depid>3e0d269a-c3a4-454a-832f-44e0528ed2c1</depid>
  <vm_group>csr-vm</vm_group>
  <vm_source>
    <vmid>9b252ece-7973-4b1f-832e-fe53c68fb963</vmid>
    <hostid>a2f7615e78330dd28697588d5fb9516504b19642f53d39d7cff4f6ab</hostid>
  </vm_source>
  <vm_target>
    <vmid>250e515b-978f-4358-b455-124b72608ba2</vmid>
    <hostid>a2f7615e78330dd28697588dd5fb9516504b19642f53d39d7cff4f6ab</hostid>
  </vm_target>
  <event>
    <type>VM_RECOVERY_COMPLETE</type>
  </event>
</escEvent>
</notification>

On Failure:

Event notification
An escEvent of type VM_RECOVERY_COMPLETE with a status of FAILURE will be sent to Netconf subscribers if the recovery workflow results in a failure. If a FAILURE event occurs, there may be cleanup required as described below under Handling Recovery Failures.

Handling Recovery Failures
If a failure occurs while attempting a recovery, the Netconf client should explicitly bring the configuration back to a consistent state by sending the corresponding Undeploy service configuration change. Refer to section 6.1.11 for Undeploy service configuration. The Undeploy would send escEvents of type VM_UNDEPLOYED, DELETE_SUBNET (if any subnets were created during deployment), DELETE_NETWORK (if any networks were created during deployment) and SERVICE_UNDEPLOYED. The Deploy service configuration change can then be retried explicitly by the Netconf client once the underlying issue is resolved. Refer to section 6.1.9 for Deploy service configuration.

6.2 Scale Out/Scale in Work Flows

6.2.1 Scale Out Work Flow
A scale-out work flow will be triggered if the current VNF is overloaded. From the user side, no request is required to triggered a scale-out. The monitors attached to VNF will trigger it if necessary.

6.2.1.1 Event Notifications
There are three notifications sent to NB during a successful scale out:

1. When a scale-out work flow is initiated, the following notifications will be sent from ESC:
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM scale out triggered</status_message>
    <svcname>CIRROS.HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702e0ceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event>
        <type>VM_SCALE_OUT_INIT</type>
      </event>
    </vm_target>
  </escEvent>
</notification>

2. The scale-out deploys a new VM. When the new VM is deployed, a notification will be sent:

<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM deployed</status_message>
    <svcname>CIRROS.HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702e0ceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event>
        <type>VM_SCALE_OUT_DEPLOYED</type>
      </event>
    </vm_target>
  </escEvent>
</notification>

3. When the VM is alive, the final notification will be sent:

<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_message>VM scaled out successfully</status_message>
    <svcname>CIRROS.HEZH</svcname>
    <svccversion>1.6</svccversion>
    <depname>dep</depname>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vm_group>CSR</vm_group>
    <vm_source>
      <vmid>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmid>
      <hostid>c64e652a4561392ae20ae702e0ceae53e417302e0d2d63e41d66a15cc</hostid>
    </vm_source>
    <vm_target>
      <event>
        <type>VM_SCALE_OUT_COMPLETED</type>
      </event>
    </vm_target>
  </escEvent>
</notification>
If there are errors during scaling out, a notification indicating the failure will be sent:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escalatedEvent xmlns="http://www.cisco.com/esc/esc">
    <status>FAILURE</status>
    <statusMessage>VM scaled out failed</statusMessage>
    <svcName>CIRROS_HEZH</svcName>
    <svcVersion>1.6</svcVersion>
    <depmName>dep</depmName>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vmGroup>CSR</vmGroup>
    <vmSource>
      <vmId>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmId>
      <hostId>c64e652a4561392ae20ae702eceeae53e417302e0d2d63e41d66a15cc</hostId>
    </vmSource>
    <vmTarget>
      <vmId>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmId>
      <hostId>c64e652a4561392ae20ae702eceeae53e417302e0d2d63e41d66a15cc</hostId>
    </vmTarget>
  </escalatedEvent>
</notification>
```

### 6.2.2 Scale In Work Flow

A scale-in work flow will be triggered if the current VNF is underloaded. From the user side, no request is required to triggered a scale-in. The monitors attached to VNF will trigger it if necessary.

#### 6.2.2.1 Event Notifications

There are two notifications sent to NB during a successful scale in:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escalatedEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <statusMessage>VM scale in triggered</statusMessage>
    <svcName>CIRROS_HEZH</svcName>
    <svcVersion>1.6</svcVersion>
    <depmName>dep</depmName>
    <tenant>hezh-yesc-0520</tenant>
    <svcid>90044227-1475-45c2-83ff-aed132f77256</svcid>
    <depid>187bc0b5-9d0b-4c01-8353-89454a6a1da6</depid>
    <vmGroup>CSR</vmGroup>
    <vmSource>
      <vmId>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmId>
      <hostId>c64e652a4561392ae20ae702eceeae53e417302e0d2d63e41d66a15cc</hostId>
    </vmSource>
    <vmTarget>
      <vmId>5283bcab-clc3-4699-9bbe-cf143b26c6aa</vmId>
      <hostId>c64e652a4561392ae20ae702eceeae53e417302e0d2d63e41d66a15cc</hostId>
    </vmTarget>
  </escalatedEvent>
</notification>
```
3. When the VM is undeployed, the final notification will be sent:

```xml
<?xml version='1.0' encoding='UTF-8'?>
<notification xmlns='urn:ietf:params:xml:ns:netconf:notification:1.0'>
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns='http://www.cisco.com/esc/esc'>
    <status>SUCCESS</status>
    <status_message>VM scaled in successfully</status_message>
  </escEvent>
</notification>
```

If there are errors during scaling in, a notification indicating the failure will be sent:

```xml
<?xml version='1.0' encoding='UTF-8'?>
<notification xmlns='urn:ietf:params:xml:ns:netconf:notification:1.0'>
  <eventTime>2015-05-19T23:46:36.276+00:00</eventTime>
  <escEvent xmlns='http://www.cisco.com/esc/esc'>
    <status>FAILURE</status>
    <status_message>VM scaled in failed</status_message>
  </escEvent>
</notification>
```