Virtual Network Function Operations

- VNF Operations, on page 1
- Managing Individual and Composite VNFs, on page 2

VNF Operations

You can start, stop and reboot VNFs. Start, stop and reboot operations are performed using the RESTful interface.

A payload is required for VNF operations:

```
POST ESCManager/v0/{internal_tenant_id}/deployments/service/{internal_deployment_id}
```

Example,

```
<?xml version='1.0' encoding='UTF-8'?>
<service_operation xmlns='urn:ietf:params:xml:ns:netconf:base:1.0'>
   <operation>stop</operation>
</service_operation>
```

You must mention start, stop or reboot in the operation field.

- Start VNF: Starts all VMs, enables monitoring, and reassigns thresholds according the KPI details. The VMs start running and move to VM_ALIVE_STATE. The service will be in service_active_state. Only undeploy can interrupt the start VNF workflow.

- Stop VNF: Once the service is stopped, monitoring is disabled and all the VM services are stopped. The VMs are no longer available. The service will be in service_stopped_state. VM will be in shutoff_state. You cannot perform any recovery, scale out, scale in. You can only undeploy the VNFs.

- Reboot VNF: Disables monitoring, reboots all VMs, that is stop and then start in OpenStack, enables monitoring, and reassigns thresholds according to KPI details. The VM is in VM_ALIVE_STATE and the service is in service_alive_state. Only undeploy can interrupt the reboot operation.

You cannot start monitoring a VNF which is already running. After a reboot, logging back into the VM must indicate the reboot, update and monitoring details. It must also indicate recovery.

VM Operations

Similar to VNF operations, you can start, stop and reboot individual VMs.

A payload is required for VM operations:

```
POST ESCManager/v0/{internal_tenant_id}/deployments/vm/{vm_name}
```

Example,

```xml
<?xml version='1.0' encoding='UTF-8'?>
<vm_operation xmlns='urn:ietf:params:xml:ns:netconf:base:1.0'>
   <operation>stop</operation>
   <force>true/false</force>
</vm_operation>
```

You must mention start, stop or reboot in the operation field.

Managing Individual and Composite VNFs

An individual service consists of a single VNF. A coupled service or a composite VNF consists of several VMs of different types. The ESC interface receives VM interdependency information from the northbound system, and uses this information during VM and VNF creation, and life cycle management. Interdependency could include VM specific workflow in the group of VMs in a single VNF, VNF monitoring and scalability and so on.

Create, read, update and delete operations are allowed on the VMs. To add more VM instances to a deployed VNF using static IP, you must provide additional IP addresses into the static IP pool. If you are using an existing static IP deployment, the minimum number of VMs is altered.

If the new minimum value, which is the number of VMs is greater than the active VMs, a new VM is added to the service. If the value is greater than the max value, the update is rejected.