# Managing VIM Connectors

# VIM Connectors Overview

The ETSI API creates VIM connectors during the processing of an LCM operation or uses an existing connector.

The Grant response or the LCM operation request from the NFVO supplies new VimConnectionInfo to the VnfInstance. During the processing of the LCM operation, ETSI synchronizes the new *VimConnectionInfo* with the VIM connectors in ESC.

A *VimConnectionInfo* is new if the VnfInstance does not have an existing VimConnectionInfo with the same id. Any *VimConnectionInfo* supplied that matches an existing *VimConnectionInfo* id stored against any VnfInstance as part of an LCM request uses the existing connector and ignore any changes submitted in the new request.

ESC creates a new VIM connector only if a matching VIM connector is not available.

The ETSI API allows only the existing *VimConnectionInfo*, and the associated VIM connector, to be updated via the Modify VNF information operation.

The Grant from the NFVO specifies the vimConnectionId for each resource. This value identifies the VimConnectionInfo and the associated VIM connector for creating the locator for each resource. The VIM specific *VimConnectionInfo.accessInfo* properties are set as additional properties in the locator.

Example for VimConnectionInfo in OpenStack:

```
{
  "id": "435456",
  "vimType": "OPENSTACK_V3",
  "interfaceInfo": {
    "endpoint": "https://10.18.54.42:13001/v3/"
  },
  "accessInfo": {
    "username": "admin",
    "password": "bmkQJtyDrbPFnJT8ENdZw2Maw",
    "project": "cbamnso",
    "projectDomain": "Default",
    "userDomain": "Default",
```

```
      "vim_project": "cbamnso"
    }
}
```

Example for VimConnectionInfo in VMware Cloud Director:

```
{
  "id": "train_vim_13001",
  "vimType": "VMWARE_VCD",
  "interfaceInfo": {
    "endpoint": "https://10.85.103.150"
  },
  "accessInfo": {
    "username": "admin@cisco",
    "password": "bmkQJtyDrbPFnJT8ENdZw2Maw",
    "vim_project": "cbamnso",
    "vim_vdc": "vdc1"
  }
}
```

Example for VimConnectionInfo in VMware vCenter

```
{
  "id": "vcenter_org1_vim",
  "vimType": "VMWARE_VSPHERE",
  "interfaceInfo": {
    "endpoint": "https://10.85.103.21"
  },
  "accessInfo": {
    "username": "admin@vsphere.local",
    "password": "bmkQJtyDrbPFnJT8ENdZw2Maw",
    "vim_project": "cbamnso",
    "vim_vdc": "vdc1"
  }
}
```

For VIM Connector Status and SNMP Trap Notifications, see the Cisco Elastic Services Controller Administration Guide.

# Creating New VIM Connectors

During the ETSI LCM operation, ESC checks each *VimConnectionInfo* against the existing VIM connector records. If an existing VIM connector is not available, ESC creates a new VIM connector.

If the *VimConnectionInfo.vimId* is supplied, then this value is used as the id of the new VIM connector. If the *VimConnectionInfo.vimId* is not supplied, then an id is generated for the new VIM connector and this value is also set as the *VimConnectionInfo.vimId*.

To use an existing VIM connector, see Using an Existing VIM Connector, on page 2.

VIM connectors to many VIMs of different types are supported by a single instance of ESC.

# Using an Existing VIM Connector

During an ETSI LCM operation, ESC checks for an existing vimConnectionInfo with a matching identifier stored against any VnfInstance.

Existing VIM connectors are found by:

- Matching the *VimConnectionInfo.vimId*, if supplied, to the id of a VIM connector.

- Matching the VIM specific properties of the *VimConnectionInfo* to a VIM connector.

    - OpenStack

        - vimType

        - interfaceInfo.endpoint

        - accessInfo.project

    - VMware Cloud Director or vCenter

        - vimType

        - interfaceInfo.endpoint

If a matching VIM connector is found, and the *VimConnectionInfo.vimId* is not set, then the *VimConnectionInfo.vimId* is set to the id of the VIM connector.

If an NFVO provides a VimConnectionInfo with accessInfo to stipulate some of the connection properties, we use the following keys to configure the VIM connectors:

| accessInfo Property | OpenStack | Cloud Director | vCenter |
|---|---|---|---|
| username | Yes | Yes | Yes |
| password | Yes | Yes | Yes |
| project | Yes | | |
| vim_project | Yes | Yes | Yes |
| projectDomain | Yes | | |
| userDomain | Yes | | |
| vim_vdc | | Yes | Yes |

The ETSI specifications does not specify the keys to be used as part of the accessInfo attribute. In order to ease integration, in the event that an NFVO uses different keys, the properties file allows the user to specify a mapping from the third party keys to the ones that ESC understands.

```
mapping.vimConnectionInfo.accessInfo.username
mapping.vimConnectionInfo.accessInfo.password
mapping.vimConnectionInfo.accessInfo.project
mapping.vimConnectionInfo.accessInfo.projectDomain
mapping.vimConnectionInfo.accessInfo.userDomain
mapping.vimConnectionInfo.accessInfo.vim_project
mapping.vimConnectionInfo.accessInfo.vim_vdc
```

To create a new VIM connector, see Creating New VIM Connectors, on page 2.

# Updating the VIM Connector

The ETSI API updates the existing VimConnectionInfo, and the associated VIM connector via the Modifying Virtual Network Functions operation. The *VimConnectionInfo* in the modify request payload is compared to the existing *VimConnectionInfo* stored against the VnfInstance.

If an existing VimConnectionInfo stored against any VnfInstance with a matching id is not found, then then VimConnectionInfo is added to the VnfInstance.

If an existing *VimConnectionInfo* stored against any VnfInstance with a matching id is found, then the VimConnectionInfo is updated. If the VimConnectionInfo has been modified and it has an associated VIM connector, then the VIM connector is also updated.

To create new VIM connectors, see Creating New VIM Connectors, on page 2.