Ansible Modules for Dell EMC VPLEX

Product Guide

Version 1.0

October 2020

DELL EMC

Notes, cautions, and warnings

NOTE: A NOTE indicates important information that helps you make better use of your product.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

Contents

1

Introduction

This chapter contains the following topics:

Topics:

Product overview

The Ansible Modules for Dell EMC VPLEX are used to automate and orchestrate the configuration of resources and provision storage from the VPLEX system. The Ansible modules are capable of managing Storage Views, Initiators, Ports, Consistency Groups, Virtual Volumes, Devices, Extents, Storage Volumes and also able to get information of currently configured resources through the gather facts module . The Ansible modules are called by tasks within the Ansible playbooks. The **Idempotency** feature is enabled for all the modules. The Idempotency feature enables the playbook to be run multiple times. The modules use VPLEX Python SDK to interface with the VPLEX.

List of Ansible Modules for Dell EMC VPLEX

- Gather facts
- Storage volume
- Extent
- Device
- Virtual volume
- Consistency group
- Port
- Initiator
- Storage view
- Rediscover array

The following parameters are the common parameters for all the modules:

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
vplexhost		str	Mandatory	The IP/FQDN of VPLEX server.
vplexuser		str	Mandatory	Name of the user used to authenticate with the VPLEX.
vplexpassword		str	Mandatory	Password of the user used to authenticate with the VPLEX.
verifycert		str	Mandatory	Specifies whether or not to verify the SSL certificate for VPlex Ansible commands.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file to be verified when verifycert is set to True. It is required only when verifycert is specified as True.

2

Configure Ansible

This chapter contains the following topics:

Topics:

Software prerequisites

This table provides information about the software prerequisites for the Ansible Modules for Dell EMC VPLEX.

Ansible modules	VPLEX version	Red Hat Enterprise Linux	Python version	VPLEX Python SDK version	Ansible version
v1.0	6.2	7.5	2.7.18 3.6.9	6.2	2.7 2.8 2.9

Steps to Install the Ansible module

The host server must be configured to run Ansible playbooks.

About this task

Do the following before you run playbooks on Ansible modules for Dell EMC VPLEX:

Steps

1. Make **ansible-vplex** as the current working directory **cd ansible-vplex**.
2. To determine the current Ansible and python versions, run the command `ansible --version`.
3. Based on the listed python version along with location displayed in the earlier command, configure the Ansible modules with the following steps:

```
[root@<user>~]# mkdir -p /usr/lib/python2.7/site-packages/ansible/modules/storage/
dellemc
[root@<user>~]# mkdir -p /usr/lib/python2.7/site-packages/ansible/module_utils/
storage/dell
[root@<user>~]# touch /usr/lib/python2.7/site-packages/ansible/modules/storage/
dellemc/__init__.py
[root@<user>~]# touch /usr/lib/python2.7/site-packages/ansible/module_utils/storage/
__init__.py
[root@<user>~]# touch /usr/lib/python2.7/site-packages/ansible/module_utils/storage/
dell/__init__.py
[root@<user>~]# cp -rf dellemc_ansible/utils/dellemc_ansible_vplex_utils.py /usr/lib/
python2.7/site-packages/ansible/module_utils/storage/dell/
dellemc_ansible_vplex_utils.py
[root@<user>~]# cp -rf dellemc_ansible/vplex/library/* /usr/lib/python2.7/site-
packages/ansible/modules/storage/dellemc/
For ansible 2.7 version,
[root@<user>~]# cp -rf dellemc_ansible/doc_fragments/dellemc_vplex.py /usr/lib/
python2.7/site-packages/ansible/utils/module_docs_fragments/dellemc_vplex.py
For ansible 2.8 or higher,
```

```
[root@<user>~]# cp -rf dellemc_ansible/doc_fragments/dellemc_vplex.py /usr/lib/
python2.7/site-packages/ansible/plugins/doc_fragments/dellemc_vplex.py
```

Install Dell EMC VPLEX Python SDK

About this task

To install VPLEX Python SDK in the host machine, follow these steps:

Steps

1. Download the tar from https://github.com/dell/python-vplex into the corresponding host system. For end customers, it is in git hub which customer can access.
2. Use the command `tar -xvf vplexapi-master.tar` to untar the file which creates a `vplexapi-master` directory.
3. Export the python path with VPLEX API.

   `export PYTHONPATH="{$PYTHONPATH}:<path of above untar'd vplexpi-1.0.0.0>`

```
[root@<user>~]# export PYTHONPATH="{$PYTHONPATH}:/root/vplexapi-master/
vplexapi-1.0.0.0"
[root@<user>~]# echo $PYTHONPATH
{}:/root/vplexapi-master/vplexapi-1.0.0.0
[root@<user>~]#
```

NOTE: This command works only on the current execution terminal. In order to make it persistent, update the same export command in `$HOME/.bashrc` file followed by system reboot.

Steps to retrieve CA certificate from VPLEX

About this task

The following are the steps to retrieve CA certificate from VPLEX and copy it to the Ansible host machine:

Steps

Log in into VPLEX setup CLI with valid credentials.

- `cd /etc/ipsec.d/cacerts`
- `scp -r strongswanCert.pem user@<ansible-host>:/execution_directory_path`
- Copy the file `strongswanCert.pem` into the Ansible host machine.

```
service@satellite-1:~> cd /etc/ipsec.d/cacerts/
service@satellite-1:/etc/ipsec.d/cacerts> ll
total 4
-rw-rw---- 1 root groupSvc 1655 Mar 14  2020 strongswanCert.pem
service@satellite-1:/etc/ipsec.d/cacerts> scp -r strongswanCert.pem root@10.227.50.57:/
root/
root@10.227.50.57's password:
strongswanCert.pem
service@satellite-1:/etc/ipsec.d/cacerts>
```

3

Ansible modules for Dell EMC VPLEX

This chapter contains the following topics:

Topics:

Gather Facts module

The Gather Facts module displays a list of specific entities in VPLEX. The Gather facts module is used with Ansible to register values that are used in conditional statements within the playbooks.

The Gather Facts module supports to access an inventory of Dell EMC storage objects.

Objects in the inventory include:

- Storage Arrays
- Storage Volumes
- Ports
- Storage Views
- Virtual Volumes
- Consistency Groups
- Devices or Distributed devices
- Distributed Consistency Groups
- Distributed Virtual Volumes
- Array Management Providers (AMP)

Get list of storage arrays

To get the list of connected storage arrays from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Storage Arrays
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - stor_array
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of storage volumes

To get the list of storage volumes from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Storage Volumes
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - stor_vol
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of ports

To get the list of front end ports from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Ports
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - port
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of initiators

To get the list of initiator ports from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Initiators
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - initiator
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of storage views

To get the list of storage views from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Storage Views
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - stor_view
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of virtual volumes

To get the list of virtual volumes from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Virtual Volumes
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - virt_vol
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of consistency groups

To get the list of consistency groups from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Consistency Groups
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - cg
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of extents

To get the list of extents from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Extents
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
```

```
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - extent
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of devices

To get the list of devices from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Devices
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - device
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of distributed devices

To get the list of distributed devices present in the VPLEX metro setup, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Distributed Devices
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    gather_subset:
      - dist_device
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of distributed consistency groups

To get the list of distributed consistency groups present in the VPLEX metro setup, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Distributed Consistency Groups
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    gather_subset:
      - dist_cg
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of distributed virtual volumes

To get the list of distributed virtual volumes present in the VPLEX metro setup, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Distributed Virtual Volumes
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    gather_subset:
      - dist_virt_vol
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get list of array management providers

To get the list of array management providers from the specific VPLEX cluster, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get list of Array Management Provider
  dellemc_vplex_gatherfacts:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    gather_subset:
      - amp
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Gather Facts module parameters

The following table provides information about the parameters that are displayed on the console when the user runs the playbook using the Gather facts module:

Parameter name	Choices or Default	Type	Mandatory/Optional Parameter	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The username to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Optional	Name of the cluster. **NOTE:** The `cluster_name` is not required for Distributed Devices, Distributed

Parameter name	Choices or Default	Type	Mandatory/Optional Parameter	Description
				Consistency Groups, and Distributed Virtual Volumes. If the user does not specify the `cluster_name` for the storage elements, excluding for the above specified distributed entries the gather facts module returns basic information of clusters.
gather_subset	• stor_array • stor_vol • port • initiator • stor_view • virt_vol • cg • device • extent • dist_device • dist_cg • dist_virt_vol • amp	array	Optional	List of string variables to specify the VPLEX entities for which the information is required. If gather_subset is not provided, the gather facts module returns list of clusters. • stor_array - storage arrays • stor_vol - storage volumes • port - ports • initiator - initiators • stor_view - storage views • virt_vol - virtual volumes (local) • cg - consistency groups (local) • device - devices (local) • extent - extents • dist_device - distributed devices • dist_cg - distributed consistency groups • dist_virt_vol - distributed virtual volumes • amp - array management providers

Storage volume module

The Storage Volume module manages the storage volumes in the VPLEX.

The module has following capabilities:

- Claim Storage Volume
- Unclaim Storage Volume
- Update/Modify Storage Volume
- Set thin rebuild
- List ITLs

Claim storage volume

To claim volume, run appropriate playbook.

The syntax of task is shown as follows:

Claim Storage Volume by name

```
- name: Claim Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    claimed_state: claimed
    state: present
```

Claim Storage Volume by ID

```
- name: Claim Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
    claimed_state: claimed
    state: present
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Unclaim storage volume

To unclaim volume, run appropriate playbook.

The syntax of task is shown as follows:

Unclaim Storage Volume by name

```
- name: Unclaim Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    claimed_state: unclaimed
    state: present
```

Unclaim Storage Volume by ID

```
- name: Unclaim Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
    claimed_state: unclaimed
    state: present
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Update or modify storage volume

To update name of storage volume, run appropriate playbook.

The syntax of task is shown as follows:

Update or modify Storage Volume by name

```
- name: Update Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    new_storage_volume_name: "{{ new_storage_volume_name }}"
    claimed_state: claimed
    state: present
```

Update or modify Storage Volume by ID

```
- name: Update Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
    new_storage_volume_name: "{{ new_storage_volume_name }}"
    claimed_state: claimed
    state: present
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Set thin rebuild

To enable thin_rebuild of storage volume, run appropriate playbook.

The syntax of task is shown as follows:

Set thin rebuild to true by name

```
- name: Set thin rebuild Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    thin_rebuild: true
    state: present
```

Set thin rebuild to true by ID

```
- name: Set thin rebuild Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
```

```
    thin_rebuild: true
    state: present
```

Set thin rebuild to false by name

```
- name: Set thin rebuild Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    thin_rebuild: false
    state: present
```

Set thin rebuild to false by ID

```
- name: Set thin rebuild Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
    thin_rebuild: false
    state: present
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

List ITLs

To see storage volume details with or without ITLs list, run appropriate playbook.

The syntax of task is shown as follows:

Get ITLs list in volume details by name

```
- name: List ITL's of Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    get_itls: true
    state: present
```

List ITLs - Get ITLs list in volume details by Id

```
- name: List ITL's of Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
```

```
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
    get_itls: true
    state: present
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get storage volume details by storage volume name without ITL's

```
- name: List ITL's of Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_name: "{{ storage_volume_name }}"
    get_itls: false
    state: present
```

Get storage volume details by storage volume ID without ITL's

```
- name: List ITL's of Storage Volume
  dellemc_vplex_storage_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    storage_volume_id: "{{ storage_volume_id }}"
    get_itls: false
    state: present
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Storage volume module parameters

The parameters for the Storage volume module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate.If it is True it verifies the SSL certificate.If it is False it do not verify the SSL certificate.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
storage_volume_name		str	Optional	Name of specific instance of the resource.
storage_volume_id		str	Optional	ID of specific storage volume.

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
new_storage_volume_name		str	Optional	The new name for renaming storage volume.
get_itls	• True • False	bool	Optional	To get the ITL's list of the storage volume.
thin_rebuild	• True • False	bool	Optional	This parameter allows to change the value of thin_rebuild.
claimed_state	• claimed • unclaimed	str	Optional	The state of specific storage volume either claimed or unclaimed.
state	• present • absent	str	Mandatory	The state of specific storage volume.

Extent module

The extent module manages the extents in VPLEX.

The Manage extents module has the following functions:

- Create an extent
- Get extent from cluster
- Rename an extent
- Delete an extent

Create extent with storage volume name

To create an extent, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Create an extent with storage_volume_name
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_volume_name: "ansible_storvol"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Create extent with storage volume id

To create an extent, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Create an extent with storage_voume_id
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
```

```
    cluster_name: "cluster-1"
    storage_volume_id: "VPD83T3:60000970000197200581533030353735"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get extent

To get the extent details, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Get extent
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    extent_name: "extent_ansible_storvol_1"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename extent

To rename the extent, run the appropriate playbook.

The syntax of the task is shown as follows:

Rename extent with extent name

```
- name: Rename extent with extent name
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    extent_name: "extent_ansible_storvol_1"
    new_extent_name: "ansible_ext_update_name"
    state: "present"
```

Rename extent with storage volume name

```
- name: Rename extent
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_volume_name: "ansible_storvol_1"
    new_extent_name: "ansible_ext_update_name"
    state: "present"
```

Rename extent with storage volume id

```
- name: Rename extent
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_volume_id: "VPD83T3:60000970000197200581533031 43"
    new_extent_name: "ansible_ext_update_id"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete extent with extent name

To delete the extent, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Delete an extent with extent_name
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    extent_name: "extent_ansible_storvol_1"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete extent with storage volume name

To delete an extent, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Delete an extent with storage_volume_name
  dellemc_vplex_extent:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_volume_name: "ansible_storvol"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Extent module parameters

The parameters for the extent module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
storage_volume_name		str	Optional	Storage volume name to create the extent. **NOTE:** Any one of the parameters storage_volume_name or storage_volume_id or extent_name is required
extent_name		str	Optional	The name of a specific instance of the resource.
storage_volume_id		str	Optional	Storage volume ID to create the extent.
new_extent_name		str	Optional	The value to be used while renaming the extent.
state	present/ absent	str	Mandatory	The state of the extent.

Device module

The Device module manages the local devices in the VPLEX.

The Manage local Device module has the following functions:

- Create a device
- Get device from cluster
- Add extent to the device
- Remove extent from the device
- Rename a device
- Delete a device

Create a raid-1 device

To create a device, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Create raid-1 device
  dellemc_vplex_device:
    vplexhost: "{{ vplexhost}}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    geometry: "raid-1"
    device_name: "ansible-test"
    extents: ["extent_1","extent_2"]
    extent_state: "present-in-device"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get device from cluster

To get details of a device using device_name, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Get device from cluster
  dellemc_vplex_device:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    device_name: "ansible-test"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add an extent to the device

To add an extent to the device, run the appropriate playbook.

NOTE: This task is supported only for raid-1 device. It is not supported for raid-0 and raid-c devices.

The syntax of the task is shown as follows:

```
- name: Add an extent to device
  dellemc_vplex_device:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    device_name: "ansible-test"
    extents: ["extent_1"]
    extent_state: "present-in-device"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove an extent from the device

To remove an extent to the device, run the appropriate playbook.

NOTE: This task is supported only for raid-1 device. It is not supported for raid-0 and raid-c devices.

The syntax of the task is shown as follows:

```
- name: Remove an extent from Device
  dellemc_vplex_device:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    device_name: "ansible-test"
    extents: ["extent_1"]
    extent_state: "absent-in-device"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename device

To rename the device, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Rename a local device
  dellemc_vplex_device:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    device_name: "ansible-test"
    new_device_name: "ansible-test-new"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete device

To delete the device, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Delete device from cluster
  dellemc_vplex_device:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    device_name: "ansible-test"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Manage local device module parameters

The parameters for the manage local device module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
cluster_name		str	Mandatory	Name of the cluster.
device_name		str	Mandatory	Name of the device. Device name can only contains letters, numbers _ or - and less than 60 characters.
geometry	• raid-1 • raid-0 • raid-c default: raid-1	str		Geometry for the new device. If no geometry specified then raid-1 set by default.
stripe_depth		str		Size of the stripe_depth if geometry is raid-0. It must be specified while creating raid-0 device.
extents		list	Optional	Extent names while creating a new device. It is required to specified while create a device, add/ remove extent from the device.
extent_state	• present-in-device • absent-in-device	str	Optional	To determine whether add /remove extent from device. It is required to specified while create a device, add/remove extent from the device. • present-in-device- Add extent to the device • absent-in-device- Remove extent from the device
new_device_name		str	Optional	The new name of the device. It is required to be specified while re-naming the device. New device name can only contains letters, numbers _ or - and less than 60 characters.
state	• present • absent	str	Mandatory	To determine whether device will exist or not. • present - The device must be present in the system • absent - The device must not be present in the system

Virtual volume module

The Virtual Volume module manages the virtual volumes in the VPLEX.

The Manage Virtual Volumes module has the following functions:

- Create a virtual volume
- Get virtual volume by using name/System ID
- Add local mirror to virtual volume by using name/System ID
- Remove local mirror from the virtual volume by using name/System ID
- Enable remote access of virtual volume by using name/System ID
- Disable remote access of virtual volume by using name/System ID
- Rename virtual volume by using name/System ID
- Delete virtual volume by using name/System ID
- Expand virtual volume by devices

Create virtual volume

To create a virtual volume, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Create Virtual volume
  dellemc_vplex_virtual_volume:
```

```
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    supporting_device_name: "{{ device_name }}"
    thin_enable: true
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get virtual volume using name or System ID

To get the details of virtual volume, run the appropriate playbook.

The syntax of the task is as follows:

Get details of virtual volume using name

```
- name: Get virtual Volume by using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_vol_name}}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get details of virtual volume using System ID

```
- name: Get virtual Volume by using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_vol_id }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add local mirror to virtual volume using name or System ID

To add local mirror to a virtual volume, run the appropriate playbook.

NOTE: This task is supported only for virtual volume created over raid-1 device. It is not supported for raid-0 and raid-c devices.

The syntax of the task is as follows:

Add local mirror using name

```
- name: Add local mirror by using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name}}"
    mirroring_device_name: "{{ mirroring_device_name }}"
    mirroring_flag: true
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add local mirror using System ID

```
- name: Add local mirror by using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    mirroring_device_name: "{{ mirroring_device_name }}"
    mirroring_flag: True
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove local mirror from virtual volume using name or System ID

To remove local mirror from a virtual volume, run the appropriate playbook.

NOTE: This task is supported only for virtual volume created over raid-1 device. It is not supported for raid-0 and raid-c devices.

The syntax of the task is as follows:

Remove local mirror using name

```
- name: Remove local mirror using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name }}"
    mirroring_device_name: "{{ mirroring_device_name }}"
    mirroring_flag: False
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove local mirror using System ID

```
- name: Remove local mirror using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    mirroring_device_name: "{{ mirroring_device_name }}"
    mirroring_flag: False
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Enable remote access using name or System ID

To enable remote access for a virtual volume, run the appropriate playbook.

The syntax of the task is as follows:

Enable remote access using name

```
- name: Enable remote access using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name }}"
    remote_access: "enable"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Enable remote access using System ID

```
- name: Enable remote access using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    remote_access: "enable"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Disable remote access using name or System ID

To disable remote access for a virtual volume, run the appropriate playbook.

The syntax of the task is as follows:

Disable remote access using name

```
- name: Disable remote access using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name }}"
    remote_access: "disable"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Disable remote access using System ID

```
- name: Disable remote access using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    remote_access: "disable"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename virtual volume using name or System ID

To rename a virtual volume, run the appropriate playbook.

The syntax of the task is as follows:

Rename virtual volume using name

```
- name: Rename virtual volume using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name }}"
    new_virtual_volume_name: "{{ new_virtual_volume_name }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename virtual volume using System ID

```
- name: Rename virtual volume using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
```

```
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    new_virtual_volume_name: "{{ new_virtual_volume_name }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Expand virtual volume with device using name or System ID

To expand a virtual volume with a device, run the appropriate playbook.

The syntax of the task is as follows:

Expand virtual volume with device using name

```
- name: Expand virtual volume with Device
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name }}"
    additional_devices: "{{ additional_devices }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Expand virtual volume with device using System ID

```
- name: Expand virtual volume with Device
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    additional_devices: "{{ additional_devices }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete virtual volume using name or System ID

To delete a virtual volume, run the appropriate playbook.

The syntax of the task is as follows:

Delete virtual volume using name

```
- name: Delete virtual volume using name
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
```

```
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_name: "{{ virtual_volume_name }}"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete virtual volume using System ID

```
- name: Delete virtual volume using ID
  dellemc_vplex_virtual_volume:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    virtual_volume_id: "{{ virtual_volume_id }}"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Manage virtual volume module parameters

The parameters for the Manage virtual volume module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. If it is True it verifies the SSL certificate. If it is False it do not verify the SSL certificate.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
virtual_volume_name		str	Optional	Name of specific instance of the resource virtual volume.
virtual_volume_id		str	Optional	ID of specific virtual volume.
new_virtual_volume_name		str	Optional	The new name for renaming virtual volume.
supporting_device_name		str	Optional	The name of supporting device on which virtual volume is created.
thin_enable	• True • False The default value is True	bool	Optional	To update thin enable value, while creating virtual volume. It is used in creating virtual volume.
mirroring_device_name		str	Optional	The name of the mirror device for add/remove mirror.

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
mirroring_flag	• True • False	bool	Optional	To Specify whether to add or remove mirror device.
remote_access	• enable • disable	str	Optional	To specify either to enable or disable remote access.
additional_devices		list	Optional	Target device list to expand virtual volume. **NOTE:** Virtual Volume Expand operation is not supported in release 1.0
state	• present • absent	str	Mandatory	The state of specific virtual volume. For delete virtual volume state is absent. For remaining operations state should be present.

Consistency group module

The Consistency group module manages the consistency group in VPLEX.

The Consistency group module has the following functionalities:

- Create a consistency group
- Add virtual volumes to consistency group
- Remove virtual volumes from consistency group
- Rename consistency group
- Delete a consistency group
- Get consistency group

Create a consistency group

To create a consistency group, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Create CG
  dellemc_vplex_consistency_group:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    cg_name: "{{ cg_name }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add virtual volumes to consistency group

To add the virtual volumes to the consistency group, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Add virtual volumes to CG
  dellemc_vplex_consistency_group:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
```

```
    cluster_name: "{{ cluster_name }}"
    cg_name: "{{ cg_name }}"
    virtual_volumes: "{{ virtual_volumes }}"
    virtual_volume_state: "present-in-cg"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove virtual volumes from consistency group

To remove the virtual volumes from the consistency group, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Remove virtual volumes from CG
  dellemc_vplex_consistency_group:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    cg_name: "{{ cg_name }}"
    virtual_volumes: "{{ virtual_volumes }}"
    virtual_volume_state: "absent-in-cg"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename consistency group

To rename the consistency group, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Rename CG
  dellemc_vplex_consistency_group:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    cg_name: "{{ cg_name }}"
    new_cg_name: "{{ new_cg_name }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete consistency group

To delete the consistency group, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Delete CG
   dellemc_vplex_consistency_group:
     vplexhost: "{{ vplexhost }}"
     vplexuser: "{{ vplexuser }}"
     vplexpassword: "{{ vplexpassword }}"
     verifycert: "{{ verifycert }}"
     cluster_name: "{{ cluster_name }}"
```

```
    cg_name: "{{ cg_name }}"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get consistency group

To get the consistency group, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get CG from cluster
  dellemc_vplex_consistency_group:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "{{ cluster_name }}"
    cg_name: "{{ cg_name }}"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Consistency group module parameters

The parameters for the Consistency group module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
cg_name		str	Mandatory	Name of the consistency group. The consistency group name can only contains letters, numbers _ or - and less than 63 characters.
virtual_volumes		list		List of virtual volumes to add/remove from the consistency group.
virtual_volume_state	• present-in-cg • absent-in-cg	str		To determine whether add /remove virtual volumes. • present-in-cg - Add virtual volumes to the consistency group. • absent-in-cg - Remove virtual volumes from the consistency group.
new_cg_name		str	Optional	The new name of the consistency group. It is required to be specified while re-naming the consistency

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
				group. The new_cg_name can only contains letters, numbers _ or - and less than 63 characters.
state	• present • absent	str	Mandatory	To determine whether consistency group will exist or not. • present - The consistency group must be present in the system. • absent - The consistency group must not be present in the system.

Port module

The port module manages the FE ports in the VPLEX.

The Manage ports module has the following functions:

- Get port
- Enable port
- Disable port

Get port

To get the port details, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get port details
  dellemc_vplex_port:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    port_name: "P0000000046E0124B-A0-FC02"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Enable port

To enable the port details, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Enable a port
  dellemc_vplex_port:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    port_name: "P0000000046E0124B-A0-FC02"
    state: "present"
    enabled: true
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Disable port

To disable the port details, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Disable a port
  dellemc_vplex_port:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    port_name: "P0000000046E0124B-A0-FC02"
    state: "present"
    enabled: false
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Port module parameters

The parameters for the Port module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
port_name		str	Mandatory	Name of the port.
enabled	true, false, None default: None	bool	Optional	The status of the port.
state	present/absent	str	Mandatory	Presence of the port.

Initiator module

The Initiator module manages the initiators available in VPLEX.

The initiator module has the following functionalities:

- Register an initiator (auto or manual) in cluster
- Get details of an initiator from a cluster
- Rename an initiator present in cluster
- Unregister an initiator in cluster
- Rediscover Initiators from a cluster

Register an initiator

To register an initiator that are visible to VPLEX port (auto-register) and not visible to VPLEX port (manual register) using port_wwn, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Register Initiator with port_wwn
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    initiator_name: "ansible_init"
    port_wwn: "0x21000024ff30ae28"
    host_type: "hpux"
    registered: true
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get details of an initiator

To get the details of an initiator, run the appropriate playbook.

The syntax of the task is as follows:

Get details of an Initiator using the initiator name

```
- name: Get details of an Initiator
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    initiator_name: "ansible_init"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get details of an Initiator using the port_wwn

```
- name: Get details of an Initiator
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    port_wwn: "0x21000024ff30ae28"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename initiator

To rename the initiator, run the appropriate playbook.

The syntax of the task is as follows:

Rename initiator using the initiator name

```
- name: Rename a registered Initiator name
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    initiator_name: "ansible_init"
    new_initiator_name: "ansible_init_update_name"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename initiator using the port_wwn

```
- name: Rename a registered Initiator name
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    port_wwn: "0x21000024ff30ae28"
    new_initiator_name: "ansible_init_update_wwn"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Unregister an initiator

To unregister an initiator, run the appropriate playbook.

The syntax of the task is as follows:

Unregister an initiator using the initiator name

```
- name: Unregister Initiator
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    initiator_name: "ansible_init"
    registered: false
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Unregister an initiator using the port_wwn

```
- name: Unregister Initiator
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    port_wwn: "0x21000024ff30ae28"
    registered: false
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rediscover initiators

To rediscover the initiators, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Rediscover Initiator
  dellemc_vplex_initiator:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Initiator module parameters

The parameters for the Initiator module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The user name to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate. • False - Specified that the SSL certificate should not be verified.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
initiator_name		str	Optional	The name of the initiator. • Do not use special characters other than '_-' and not more than 36 character.

Parameter name	Choice or default	Type	Mandatory/Optional Parameter	Description
new_initiator_name		str	Optional	The name to be used while renaming the initiator. • Do not use special characters other than '_-' and not more than 36 characters
host_type	• default • hpux • sun-vcs • aix • recoverpoint	str	Optional	Type of host associated with initiator. For registering the initiator manually, host_type should be specified along with port_wwn or iscsi_name. The supported values are as follows: • default • hpux • sun-vcs • aix • recoverpoint The default value is 'default'. **NOTE:** The host_type 'recoverpoint' is not supported in this release.
port_wwn		str	Optional	WWN of the port to register. For registering the initiator as FC port, port_wwn should be specified. This parameter is optional for all the operations except register initiator.
registered	• True • False	bool	Optional	Defines whether the initiator state is to be registered or not. Valid values are True/False/None. • True - Register • False - Unregister The default value is None.
state	• absent • present	str	Mandatory	Defines whether the initiator must be present in VPLEX. • absent - The initiator must not be present in VPLEX • present - The initiator must be present in VPLEX Valid values - Present (It is always assumed as initiators are visible in VPLEX)

Storage View module

The Storage view module manages the storage views available in VPLEX.

The Storage view module has the following functionalities:

- Get details of a storage view
- Create a storage view
- Delete a storage view
- Rename a storage view
- Add ports to a storage view
- Remove ports from a storage view
- Add initiators to a storage view
- Remove initiators from a storage view
- Add virtual volumes to a storage view
- Remove virtual volumes from a storage view

Get details of a storage view

To get the details of a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Get storage view details
   dellemc_vplex_storage_view:
       vplexhost: "{{ vplexhost }}"
       vplexuser: "{{ vplexuser }}"
       vplexpassword: "{{ vplexpassword }}"
       verifycert: "{{ verifycert }}"
       cluster_name: "cluster-1"
       storage_view_name: "ansible_storview"
       state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Create a storage view

To create a storage view in VPLEX with ports, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Create a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    ports: ["P0000000046E0124B-A0-FC00", "P0000000046E0124B-A0-FC01"]
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Delete a storage view

To delete a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Delete a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    state: "absent"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rename a storage view

To rename a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Rename a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    new_storage_view_name: "ansible_storview_new"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add ports to a storage view

To add ports to a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Add ports to a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    ports: ["P0000000046E0124B-A0-FC00", "P0000000046E0124B-A0-FC01"]
    port_state: "present-in-view"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add initiators to a storage view

To add initiators to a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Add initiators to a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    initiators : ["ansible_init_1", "ansible_init_2"]
    initiator_state: "present-in-view"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Add virtual volumes to a storage view

To add virtual volumes to a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Add virtual volumes to a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    virtual_volumes: ["ansible_vir_1", "ansible_vir_2"]
    virtual_volume_state: "present-in-view"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove ports from a storage view

To remove ports from a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Remove ports from a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    ports: ["P00000000046E0124B-A0-FC00", "P00000000046E0124B-A0-FC01"]
    port_state: "absent-in-view"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove initiators from a storage view

To remove initiators from a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Remove initiators from a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    initiators : ["ansible_init_1", "ansible_init_2"]
    initiator_state: "absent-in-view"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Remove virtual volumes from a storage view

To remove virtual volumes from a storage view in VPLEX, run the appropriate playbook.

The syntax of the task is as follows:

```
- name: Remove virtual volumes from a storage view
  dellemc_vplex_storage_view:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: "cluster-1"
    storage_view_name: "ansible_storview"
    virtual_volumes: ["ansible_vir_1", "ansible_vir_2"]
    virtual_volume_state: "absent-in-view"
    state: "present"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Storage view module parameters

The parameters for the Storage view module are listed as follows with an example:

Parameter name	Choice or Default	Type	Mandatory/Optional Parameter	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host.
vplexuser		str	Mandatory	The username to access the VPLEX server.
vplexpassword		str	Mandatory	The password to access the VPLEX server.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster.
storage_view_name		str	Mandatory	Name of the storage view used by the CRUD operations. • Do not use special characters other than '_-' and not more than 36 characters
new _storage_view_name		str	Optional	Name to be used for renaming the storage view.

Parameter name	Choice or Default	Type	Mandatory/Optional Parameter	Description
				• Do not use special characters other than '_-' and not more than 36 characters
ports		list	Optional	Ports list to add or remove to storage view.
initiators		list	Optional	Initiators list to add or remove to storage view.
virtual_volumes		list	Optional	Virtual volumes list to add or remove to storage view.
port_state	• present-in-view • absent-in-view	str	Optional	Decides the presence of the ports in the storage view. • absent-in-view - The ports must not be present in the storage view. • present-in-view - The ports must be present in the storage view.
Initiator_state	• present-in-view • absent-in-view	str	Optional	Decides the presence of the initiators in the storage view. • absent-in-view - The initiators must not be present in the storage view • present-in-view - The initiators must be present in the storage view
virtual_volume_state	• present-in-view • absent-in-view	str	Optional	Decides the presence of the virtual volumes in the storage view. • absent-in-view - The virtual volumes must not be present in the storage view • present-in-view - The virtual volumes must be present in the storage view
state	• absent • present	str	Mandatory	Decides the presence of the storage view in VPLEX.

Parameter name	Choice or Default	Type	Mandatory/Optional Parameter	Description
				• absent - The storage view must not be present in VPLEX • present - The storage view must be present in VPLEX

Rediscover array module

The array module rediscovers the LUNs in the storage array.

The array module has the following functions:

- Rediscover Array
- Get Array

Rediscover array

To rediscover the array, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Rediscover the StorageArray
  dellemc_vplex_array:
    vplexhost: "{{ vplexhost }}"
    vplexuser: "{{ vplexuser }}"
    vplexpassword: "{{ vplexpassword }}"
    verifycert: "{{ verifycert }}"
    cluster_name: " {{ cluster_name }}"
    array_name: " {{ array_name }}"
    rediscover: true
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Get array

To get the array, run the appropriate playbook.

The syntax of the task is shown as follows:

```
- name: Get StorageArray
  dellemc_vplex_array:
    vplexhost: "{{ vplexhost}}"
    vplexuser: "{{ vplexuser}}"
    vplexpassword: "{{ vplexpassword}}"
    verifycert: "{{ verifycert }}"
    cluster_name: " {{ cluster_name }}"
    array_name: " {{ array_name }}"
```

The parameters must be set before the user runs the playbook. For more information about the parameters, see the Parameters table.

Rediscover array module parameters

The parameters for the Rediscover array module are listed as follows with an example:

Parameter name	Choice or default	Type	Mandatory/Optional Parameters	Description
vplexhost		str	Mandatory	IP or FQDN of the VPLEX host. This parameter is mandatory.
vplexuser		str	Mandatory	The user name to access the VPLEX server. This parameter is mandatory.
vplexpassword		str	Mandatory	The password to access the VPLEX server. This parameter is mandatory.
verifycert	• True • False	bool	Mandatory	To validate the SSL certificate. • True - Verifies the SSL certificate • False - Specified that the SSL certificate should not be verified This parameter is mandatory.
ssl_ca_cert		str	Optional	Path of SSL CA certificate file specified in .pem format. It is required only when verifycert is set to "True".
cluster_name		str	Mandatory	Name of the cluster. This is Mandatory parameter for all operations.
array_name		str	Mandatory	Name of the array. This is Mandatory parameter for all operations.
rediscover	• True • False	bool	Optional	• True - Rediscover Array. • False - Get array details.

4

Sample playbooks

Sample playbooks illustrate the proper usage and some advance capabilities of the existing modules.

The list of sample playbooks included in this release is as follows:

Playbook name	Operations
dellemc_vplex_array.yml	Rediscovers array present in the specified cluster in VPLEX.
dellemc_vplex_consistency_group_tests.yml	Performs basic consistency group operations (Create/Get/Delete/Modify: name of consistency group)
dellemc_vplex_device_tests.yml	Handles basic device operations in local VPLEX setup (Create/Get/Delete/Modify: name of device/Add and Remove extent in raid-1 device)
dellemc_vplex_extent_tests.yml	Handles extent operations on a VPLEX cluster (Create/Get/Delete/Modify: name of extent)
dellemc_vplex_gatherfacts_tests.yml	Gathers list of storage objects from DELL EMC VPLEX Array (Storage Arrays, Storage Volumes, Ports, Initiators, local Consistency Goups, local Devices, local Virtual Volumes, Distributed devices, Distributed Consistency Groups, Distributed Virtual Volumes and Array Management Providers)
dellemc_vplex_initiator_tests.yml	Performs host initiator port operations that are visible to VPLEX (Register/Unregister/Get/Modify: name of initiator/Rediscover Initiators)
dellemc_vplex_port_tests.yml	Performs VPLEX port operations (Enable/Disable/Get port details)
dellemc_vplex_storage_view_tests.yml	Handles Storage view operations in VPLEX (Create/Get/Delete/Modify: name of storage view, add/remove host initiator ports, add/remove VPLEX ports and add/remove virtual volumes)
dellemc_vplex_storage_volume_tests.yml	Performs Storage Volume operations (Claim/Unclaim/Get details of Storage Volume with and without itl's/Modify: name of Storage volume)
dellemc_vplex_virtual_volume_tests.yml	Handles basic virtual volume operations (Create/Get/Delete/Modify: name of virtual volume, add/remove local mirror in virtual volume created on raid-1 device, enable/disable remote access)