FusionStorage OpenStack Driver 2.2.RC4

Configuration Guide

Issue 01

Date 2021-04-15





Copyright © Huawei Technologies Co., Ltd. 2021. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions

HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base

Bantian, Longgang Shenzhen 518129

People's Republic of China

Website: https://www.huawei.com

Email: support@huawei.com

Contents

1 Overview	1
2 Version Mappings	2
3 Deployment	4
3.1 Obtaining Cinder Driver	4
3.2 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode	5
3.2.1 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode (Min SCSI Mode)	
3.2.2 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode (C	
3.3 Deploying FusionStorage Driver for the Cinder Volume Service in Containerized Mode	
4 Configuring Basic Properties	9
4.1 Configuring the conf File (Mitaka in SCSI Mode)	g
4.2 Configuring the conf File (Others)	11
4.3 Configuring Host Multipathing (iSCSI Mode)	13
4.3.1 Installing the Multipathing Tool Package	13
4.3.2 Configuring the Multipathing Service	14
4.4 Configuring OpenStack Multipathing(iSCSI Mode)	14
4.4.1 Configuring Multipathing for OpenStack Nova Nodes	14
4.4.2 Configuring Multipathing for OpenStack Cinder Nodes	14
4.5 Configuring the QoS Property	14
5 FAQ	18
5.1 Failed to Create a Volume from an Image When verify_glance_signatures Is Set to enabled	18
5.2 Failed to Create a VM on the Specified Host If iscsi_manager_groups Is Used When the iSCSI New Is Used to Connect to Distributed Storage	
5.3 When the SCSI Network Is Used to Connect to Distributed Storage, the number of concurrent I/disks on VMs is limited to less than 64	
6 Appendix	21
6.1 Appendix 1: Configuring Volume-backend image	21
6.2 Appendix 2: Configuring Image-Volume Cache	22

1 Overview

This chapter describes the definition of Cinder Driver.

Cinder Driver is a plug-in that is deployed on the OpenStack Cinder module. The plug-in can be used to provide functions such as the logical volume and snapshot for virtual machines (VMs) in OpenStack.

2 Version Mappings

This chapter describes the version mappings among Cinder Driver, FusionStorage, and OpenStack.

Version mapping between OpenStack and storage products

OpenStack Version	Storage Product Version
Mitaka	FusionStorage V100R006C30
Queens	FusionStorage Block 8.0.0/8.0.1
Rocky	OceanStor Pacific 8.1.RC2
Stein	
Train	
Ussuri	
Victoria	

◯ NOTE

Queens only support iSCSI protocol.

Mitaka only supports OceanStor Pacific 8.1.RC2.

Mappings among Cinder Driver, features, and the OpenStack version ($\sqrt{}$: supported, x: not supported)

Feature	Mita ka	Quee ns	Rock y	Stein	Train	Ussur i	Victo ria
Create Volume	√	√	√	√	√	√	√
Delete Volume	√	√	√	√	√	√	√
Attach Volume	√	√	√	√	√	√	√
Detach Volume	√	√	√	√	√	√	√
Extend Volume	√	√	√	√	√	√	√

Feature	Mita ka	Quee ns	Rock y	Stein	Train	Ussur	Victo ria
Create Snapshot	√	√	√	√	√	√	√
Delete Snapshot	√	√	√	√	√	√	√
Create Volume from Snapshot	√	√	√	√	√	√	√
Create Volume from Image	√	√	√	√	√	√	√
Create Volume from Volume	√	√	√	√	√	√	√
Create Image from Volume	√	√	√	√	√	√	√
SmartThin	√	√	√	√	√	√	√
Manage/Unmanage Volume	√	√	√	√	√	√	√
Manage/Unmanage Snapshot	х	√	√	√	√	√	√
Multipath	√	√	√	√	√	√	√
QoS	√	√	√	√	√	√	√
Retype	√	√	√	√	√	√	√
iSCSI	√ (Restr icted)	√	√	√	√	√	√
SCSI	√	√	√	√	√	√	√
MultiAttach	х	√	√	√	√	√	√
Revert to Snapshot	х	√	√	√	√	√	√
Backup Volume	√	√	√	√	√	√	√
HyperMetro	х	х	х	х	х	х	х
Replication V2.1	х	х	х	х	х	х	х
HyperMetro Consistency Group	х	х	х	х	х	х	х
Backup Snapshot	х	х	х	х	х	х	х
Snapshot Consistency Group	х	х	х	х	х	х	х
Consistency Group	х	х	х	х	х	х	х

3 Deployment

- 3.1 Obtaining Cinder Driver
- 3.2 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode
- 3.3 Deploying FusionStorage Driver for the Cinder Volume Service in Containerized Mode

3.1 Obtaining Cinder Driver

You can obtain FusionStorage Cinder Driver in either of the following ways:

- OpenStack community warehouse. Since the Rocky version, FusionStorage Cinder Driver has been integrated into the OpenStack community warehouse. After OpenStack is installed, Cinder Driver is provided. Cinder Driver is stored in the ../cinder/cinder/volume/drivers/fusionstorage directory.
- FusionStorage OpenStack Driver warehouse
- Step 1 Open a browser and enter https://github.com/Huawei/FusionStorage_OpenStack_Driver in the address box.
- **Step 2** Click the **Download ZIP** button to download the Cinder Driver package to a local host.
- **Step 3** Decompress the package.
- **Step 4** In the decompressed directory, find the **Cinder** directory, which contains Cinder Driver of multiple OpenStack versions.

□ NOTE

- After OpenStack community versions are released, new features cannot be integrated.
 The bug fixing period is long and risks exist. FusionStorage Cinder Driver warehouse
 versions ensure that new features are released and bugs are fixed in time. These
 versions are more stable than OpenStack community versions.
- The OpenStack community maintains only two stable versions. FusionStorage Cinder Driver warehouse maintains six stable versions, ensuring long-term stable running of historical versions.
- It is strongly recommended that OpenStack community versions be replaced by FusionStorage Cinder Driver warehouse versions.

3.2 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode

3.2.1 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode (Mitaka in SCSI Mode)

- Step 1 Log in to all OpenStack controller nodes and run the mkdir -p oceanstor_100d_fsc_cli command to create the oceanstor_100d_fsc_cli directory.
- Step 2 Upload OceanStor_100D_Fsc_Cli_Mitaka.tar.gz to the oceanstor_100d_fsc_cli directory and run the tar zxvf OceanStor_100D_Fsc_Cli_Mitaka.tar.gz command to decompress the installation package.
- **Step 3** Run the **sh appctl.sh install** command to install fsc_cli.
- Step 4 Copy all files in the Mitaka directory to the /usr/lib/python2.7/site-packages/cinder/volume/drivers directory. If the /usr/lib/python2.7/site-packages/cinder/volume/drivers directory does not exist, create it manually.
 - **◯** NOTE

Replace /usr/lib/python2.7/site-packages/cinder/volume/drivers with the actual path.

Step 5 Change the owner of the **dsware.py** and **fspythonapi.py** files in the **/usr/lib/python2.7/site-packages/cinder/volume/drivers** directory to **root:root** and the permission on them to **644**.

chown root:root /usr/lib/python2.7/site-packages/cinder/volume/drivers/dsware.py **chown root:root** /usr/lib/python2.7/site-packages/cinder/volume/drivers/fspythonapi.py **chmod 644** /usr/lib/python2.7/site-packages/cinder/volume/drivers/dsware.py **chmod 644** /usr/lib/python2.7/site-packages/cinder/volume/drivers/fspythonapi.py

- **Step 6** Make configuration by referring to **4 Configuring Basic Properties**.
- **Step 7** After configuration, restart the Cinder-Volume service by running the following command:

systemctl restart openstack-cinder-volume.service

Step 8 Check the status of service restart by running the **cinder service-list** command. If **State** is **up**, the Cinder-Volume service has been restarted.

root@redhat:/				+	
+ Binary Reason	•	Host	Zone Status State	Updated_at	Disabled

----End

3.2.2 Deploying FusionStorage Driver for the Cinder Volume Service in Non-Containerized Mode (Others)

Step 1 Before installation, delete all the installation files of Huawei OpenStack Driver. The default installation path is /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage.

□ NOTE

On the host, the version of Python is 2.7. If another version is used, use the correct version number. You can obtain the Cinder Driver installation directory by running the following commands:

```
# python -c "from cinder.volume import drivers; print (drivers.__path__)"
['/usr/lib/python2.7/site-packages/cinder/volume/drivers']
```

As shown above is the directory of all drivers. So the fusionstorage driver installation directory is as follows:

/usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage

If the directory does not exist, create it manually. And change the owner the permission of the directory:

mkdir -p /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage chown root:root /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage chmod 755 /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage

□ NOTE

Replace /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage with the actual path

Step 2 Copy OpenStack Cinder Driver to the Cinder Driver installation directory. The default installation path is described in **Step 1**.

```
# ls -l
-rw-r--r- 1 root root 1138 May 15 08:23 constants.py
-rw-r--r- 1 root root 22174 May 15 08:23 dsware.py
-rw-r--r- 1 root root 18880 May 15 08:23 fs_client.py
-rw-r--r- 1 root root 4799 May 15 08:23 fs_conf.py
-rw-r--r- 1 root root 14419 May 15 08:23 fs_flow.py
-rw-r--r- 1 root root 2248 May 15 08:23 fs_qos.py
-rw-r--r- 1 root root 2988 May 15 08:23 fs_utils.py
-rw-r--r- 1 root root 0 May 15 08:23 __init__.py
```

- **Step 3** Make configuration by referring to **4 Configuring Basic Properties**.
- **Step 4** After configuration, restart the Cinder-Volume service by running the following command:

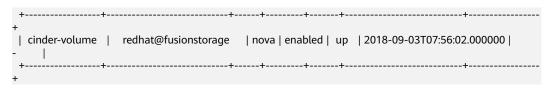
systemctl restart openstack-cinder-volume.service

Step 5 Check the status of service restart by running the **cinder service-list** command. If **State** is **up**, the Cinder-Volume service has been restarted.

```
root@redhat:/# cinder service-list

+-----+

+ Binary | Host | Zone | Status | State | Updated_at | Disabled Reason |
```



----End

3.3 Deploying FusionStorage Driver for the Cinder Volume Service in Containerized Mode

Step 1 (Optional) Run the **docker save** command to back up the Cinder Volume container image.

Ⅲ NOTE

To obtain the image version, run the docker image ls command.

Step 2 Run the following command to create temporary directory **build** in any directory and go to the created directory:

mkdir ***/build;cd ***/build

Step 3 Run the following commands to copy the obtained FusionStorage Driver code files to the current directory:

```
# ls -l
-rw-rw-r-- 1 root root 1138 May 15 08:23 constants.py
-rw-rw-r-- 1 root root 22174 May 15 08:23 dsware.py
-rw-rw-r-- 1 root root 18880 May 15 08:23 fs_client.py
-rw-rw-r-- 1 root root 4799 May 15 08:23 fs_conf.py
-rw-rw-r-- 1 root root 14419 May 15 08:23 fs_flow.py
-rw-rw-r-- 1 root root 2988 May 15 08:23 fs_utils.py
-rw-rw-r-- 1 root root 0 May 15 08:23 __init__.py
```

Step 4 Create the **Dockerfile** file in the current directory and edit the following content:

FROM ***:***
COPY *.py /fusionstorage/cinder/driver/path/

□ NOTE

- Replace ********* with the name and version of the original Cinder Volume container image.
- Replace /fusionstorage/cinder/driver/path with the path of Driver in the container. You can obtain the Cinder Driver code directory by running the following commands:
 # python -c "from cinder.volume import drivers; print (drivers.__path__)"
 ['/usr/lib/python2.7/site-packages/cinder/volume/drivers']
- As shown above is the directory of all drivers. So the fusionstorage driver installation directory is as follows:
 - /usr/lib/python2.7/site-packages/cinder/volume/drivers/**fusionstorage**
- If the directory does not exist, create it manually. And change the owner the permission
 of the directory:

mkdir -p /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage chown root:root /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage chmod 755 /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage

■ NOTE

Replace /usr/lib/python2.7/site-packages/cinder/volume/drivers/fusionstorage with the actual path

Step 5 Run the following command to build an image:

docker build -t ***:*** .

□ NOTE

Replace ***:*** with the name and version of the original Cinder Volume container image.

- **Step 6** Make configuration by referring to **4 Configuring Basic Properties**.
- **Step 7** Restart the Cinder Volume container.

4 Configuring Basic Properties

This chapter describes how to configure the FusionStorage Cinder Driver.

- 4.1 Configuring the conf File (Mitaka in SCSI Mode)
- 4.2 Configuring the conf File (Others)
- 4.3 Configuring Host Multipathing (iSCSI Mode)
- 4.4 Configuring OpenStack Multipathing(iSCSI Mode)
- 4.5 Configuring the QoS Property

4.1 Configuring the conf File (Mitaka in SCSI Mode)

Step 1 Configure /etc/cinder/cinder.conf.

Modify the **enabled_backends**, **dsware_manager**, and **fusionstorageagent** fields under **[DEFAULT]**. The following is an example:

```
[DEFAULT]
enabled_backends=fusionstorage_8_scsi
dsware_manager = 192.168.211.99
fusionstorageagent = 192.168.211.82,192.168.211.85,192.168.211.86
```

Add the following fields to the end of the file:

```
[fusionstorage_8_scsi]
volume_backend_name = fusionstoragescsi
volume_driver = cinder.volume.drivers.dsware.DSWARELocalDriver
pool_id_list = 0,1
over_ratio = 3.0
manager_ips=
node1.hw.com:192.168.160.177,
node2.hw.com:192.168.160.178
```

Table 4-1 describes the parameters in the configuration file.

Table 4-1 Parameter description

Parameter	Description
enabled_backe nds	Name of the storage node that needs to take effect in the system. For example, fusionstorage_8_scsi .
	If multiple storage nodes need to take effect, separate their names with commas (,).
dsware_manag er	Floating IP address of FusionStorage.
fusionstoragea gent	Management IP address of the OpenStack compute node (VBS) used for the communication between OpenStack and FusionStorage. If there are multiple IP addresses, separate them using commas (,).
volume_backe nd_name	Name of the back-end storage for the volume. For example, fusionstoragescsi.
volume_driver	Path of the storage system driver. The value is fixed to cinder.volume.drivers.dsware.DSWARELocalDriver.
pool_id_list	Storage pool list. This parameter specifies the storage pool where the volume is created. If there are multiple storage pools, you can use this parameter to create volumes in the storage pool with the specified ID.
over_ratio	Thin provisioning ratio. In thin provisioning mode, you can set the storage overcommitment rate. You can set this parameter to 1 or higher. If you do not set this parameter, default value 1 is used. 3.0 or higher is recommended. This parameter is optional.
manager_ips	Host name of OpenStack and the management IP address used for creating a VBS. Multiple IP addresses are separated by commas (,) and line breaks. The information about the last VBS is not followed by a comma (,).
	In this example, node1.hw.com is the host name of OpenStack, and 192.168.160.177 is the management IP address displayed for a VBS after the VBS is created for the OpenStack node.

Step 2 Test the connectivity of fsc_cli.

Run the following command:

fsc_cli --op getDSwareIdentifier --manage_ip x.x.x.x --ip y.y.y.y

x.x.x.x indicates the floating IP address of FusionStorage, and *y.y.y.y* indicates the IP address of a VBS node.

4.2 Configuring the conf File (Others)

At the end of /etc/cinder/cinder.conf, configure the FusionStorage back end with DSWAREDriver. volume_driver indicates the loaded driver file, volume_backend_name indicates the name of the back end, each row of manager_ips indicates the name and IP address of an FSA host, dsware_rest_url indicates the IP address of a storage array, san_login and san_password are the login information of the storage array, and dsware_storage_pools indicates the storage pool name on the storage array. For details, see Table 4-2.

Ensure that both the owner and user group of /etc/cinder/cinder.conf are cinder.

-rw-r--r- 1 cinder cinder 2839 Aug 29 15:29 cinder.conf

Step 1 Add the FusionStorage back end. Configure parameters for the back end according to **Table 4-2**.

If the iSCSI mode is used, **volume_driver** needs to be set to **cinder.volume.drivers.fusionstorage.dsware.DSWAREISCSIDriver**.

[fusionstorage]
...
volume_driver = cinder.volume.drivers.fusionstorage.dsware.DSWAREISCSIDriver

Step 2 In the [DEFAULT] area, configure the FusionStorage back end.

[DEFAULT]
...
enabled_backends=fusionstorage

Step 3 If the Cinder Volume service is deployed on multiple nodes, you need to configure a distributed lock in the [coordination] area. For details about <user>, <password>, <host>, and <database>, see the value of connection under [database] in the configuration file.

[coordination] backend_url = mysql://<user>:<password>@<host>/<database>?charset=utf8

For example, [database] in the configuration file is configured as follows:

[database] connection = mysql+pymysql://cinder:302824058e9a4f31@127.0.0.1/cinder

Then configure [coordination] as follows:

[coordination] backend_url = mysql://cinder:302824058e9a4f31@127.0.0.1/cinder

Table 4-2 Parameter description

Parameter	Description	Mandatory
volume_backe nd_name	Back end name of the default driver.	Yes
volume_driver	Default driver.	Yes
dsware_rest_ur l	URL and port number used by the Cinder node to access FusionStorage (If the management IP address of FusionStorage is an IPv6 IP address, configure https://[ipv6_ip]:28443, where ipv6_ip is the management IPv6 IP address of FusionStorage).	Yes
san_login	User name used by the Cinder node to access FusionStorage.	Yes
san_password	Password used by the Cinder node to access FusionStorage.	Yes
dsware_storag e_pools	Names of existing storage pools on FusionStorage.	Yes
manager_ips	Management host name and its corresponding IP address. ':' is preceded by a host name, and followed by the corresponding host IP address. Each host is separated using commas (,). Add spaces at the start of each row.	Mandatory when selecting the SCSI mode
scan_device_ti meout	Timeout for scanning disks when the iSCSI mode is selected. The default value is 3 , the type is Int, and the unit is seconds.	Mandatory when selecting the SCSI mode
target_ips	Management IP addresses of nodes when the iSCSI mode is selected. They are separated by commas (,). You must specify either target_ips or iscsi_manager_groups.	Conditionall y mandatory when selecting the iSCSI mode

Parameter	Description	Mandatory
iscsi_manager_ groups	IP address group formed by management IP addresses when the iSCSI mode is selected. IP addresses in an IP address group are separated by semicolons (;), and IP address groups are separated by commas (,). You must specify either target_ips or iscsi_manager_groups.	Conditionall y mandatory when selecting the iSCSI mode
use_ipv6	Whether to use an IPv6 address as the service IP address. The default value is False .	Optional when selecting the SCSI mode

4.3 Configuring Host Multipathing (iSCSI Mode)

Multipathing is configured to improve the link reliability of LUNs of SAN storage. If the multipathing configuration is incorrect, an I/O error occurs when a single link is faulty. As a result, the VM file system is read-only or faulty, affecting VM service delivery. When configuring multipathing, install the multipathd service on the host and configure proper parameters. Then, configure multipathing parameters on OpenStack.

4.3.1 Installing the Multipathing Tool Package

Step 1 Install the Multipath tool package based on the operating system.

CentOS:

yum install -y device-mapper-multipath

Ubuntu:

apt-get install -y multipath-tools apt-get install -y multipath-tools-boot

Step 2 Enable the host multipathing service.

CentOS:

/sbin/mpathconf --enable systemctl start multipathd.service systemctl enable multipathd.service systemctl restart multipathd.service

Ubuntu:

systemctl restart multipath-tools.service

4.3.2 Configuring the Multipathing Service

Add the following content to the **devices** field in the multipathing configuration file (/etc/multipath.conf). For details, see the Configuring Multipathing for an Application Server (Red Hat or CentOS) on OceanStor 100D.

```
devices {
  device {
          vendor
                              "Huawei"
          product
                               "VBS fileIO"
          path_grouping_policy
                                  multibus
          path_checker
                             const
          prio
          path_selector
                                "service-time 0"
          failback
                              immediate
          no_path_retry
                                 "10"
```

After the configuration is complete, restart the multipathd service.

systemctl restart multipathd.service

4.4 Configuring OpenStack Multipathing(iSCSI Mode)

4.4.1 Configuring Multipathing for OpenStack Nova Nodes

Step 1 Configure the /etc/nova/nova.conf file.

Add volume_use_multipath = True in [libvirt] of /etc/nova/nova.conf.

□ NOTE

If "/etc/nova/nova-cpu.conf" exists in the Nova Compute node, add "volume_use_multipath = True" to its [libvirt].

Step 2 Restart the nova-compute service.

systemctl restart openstack-nova-compute.service

----Fnd

4.4.2 Configuring Multipathing for OpenStack Cinder Nodes

Step 1 Add the following content to the end of BACKEND in the /etc/cinder/cinder.conf file:

```
[fusionstorage]
...
use_multipath_for_image_xfer = true
enforce_multipath_for_image_xfer = true
```

Step 2 Restart the cinder-volume service.

systemctl restart openstack-cinder-volume.service

----End

4.5 Configuring the QoS Property

QoS in OpenStack mainly depends on front-end Hypervisor and back-end storage.

For details about how to configure front-end QoS, see https://docs.openstack.org/cinder/

latest/admin/blockstorage-basic-volume-qos.html.

This section describes how to configure back-end QoS for Huawei storage devices.

- **Step 1** Run the following command to create a volume type. <name> indicates the name of thevolume type.
 - # cinder type-create <name>
- **Step 2** Run the following command to enable QoS. <vtype> indicates the volume type name configured in **Step 1**.
 - # cinder type-key <vtype> set capabilities:QoS_support='<is> true'
- **Step 3** Run the following command to create a QoS specification. <name> indicates the name of a customized QoS specification.

cinder qos-create <name> <qos_key>=*** <qos_trigger_key>=***

Table 4-3 qos_key parameters

Parameter	Description	Remarks	Mandatory	
maxIOPS	Maximum IOPS.	The value is a positive integer. The default value is 999999999.	No, but one must be specified. The priority of maxIOPS is higher	
maxMBPS	Maximum bandwidth.	The value is a positive integer, expressed in MB/s. The default value is 999999.	than total_iops_sec; the priority of maxMBPS is higher than total_bytes_sec.	
total_iops_sec	Maximum IOPS.	The value is a positive integer. The default value is 999999999.		
total_bytes_sec	Maximum bandwidth.	The value is a positive integer, expressed in byte/s. The default value is 999999 * 1024 * 1024.		

Table 4-4 qos_trigger_key parameters

Parameter	Description	Remarks	Mandatory
scheduleType	QoS triggered policy type.	Available values are:0(Always), 1(Oneoff), 2(Daily), 3(Weekly). The default is 0 when not set.	No
startDate	QoS triggered start date.	Not earlier than the current date, the format of the date is: "xx-xx-xx". Such as "2019-06-01".	Yes when the scheduleType is not 0
startTime	QoS triggered start time.	In 24-hour format, the time format is: "xx-xx", such as "08:00".	Yes when the scheduleType is not 0
durationTime	QoS daily duration.	The configuration range is 30 minutes to 24 hours. The format is: "xx:xx", for example "24:00", "0:30".	Yes when the scheduleType is not 0
dayOfWeek	QoS weekly effective date.	Available values are: "Mon", "Tue", "Wed", "Thur", "Fri", "Sat", "Sun", which means Monday to Sunday respectively. Support for configuring multiple parameters, separated by spaces(" "). For example, "Mon Tue Thur Sun" means that QOS will take effect on Monday, Tuesday, Thursday and Sunday.	Yes when the scheduleType is 3

Step 4 Associate the QoS specification with the volume type. <qos_specs> indicates the QoS specification ID created in **3** and <volume_type_id> indicates the volume type ID created in **Step 1**.

cinder qos-associate <qos_specs> <volume_type_id>

Step 5 Use the volume type in **Step 1** to create a volume.

----End

■ NOTE

When the qos_trigger_key parameter is configured, the storage version is required to be FusionStorage V100R008C00 or later.

When the qos_trigger_key parameter is configured, the user role is required to be a super administrator.

When the qos_trigger_key parameter is configured, change the environment time of OpenStack to the storage time.

5 FAQ

5.1 Failed to Create a Volume from an Image When verify_glance_signatures Is Set to enabled

5.2 Failed to Create a VM on the Specified Host If iscsi_manager_groups Is Used When the iSCSI Network Is Used to Connect to Distributed Storage

5.3 When the SCSI Network Is Used to Connect to Distributed Storage, the number of concurrent I/Os to disks on VMs is limited to less than 64.

5.1 Failed to Create a Volume from an Image When verify_glance_signatures Is Set to enabled

Symptom

OpenStack creates a volume from an image. After the image is copied for a period of time, the following error message is displayed: Failed to copy metadata to volume: Glance metadata cannot be updated, key signature_verified exists for volume id b75f957c-0ba4-4099-8706-866c68012779. As a result, the volume fails to be created.

Environment

Operating system: CentOS 7.6

Server: Huawei 5288 V5OpenStack version: Rocky

Root Cause

BUG Link: https://bugs.launchpad.net/cinder/+bug/1823445

In Rocky and later versions, the **verify_glance_signatures** field is added to **cinder.image.glance.py**. Its default value is **enabled**. When **verify_glance_signatures** in the **cinder.conf** file is **disabled** and **signature_verification** exists in the image, the DB API does not allow repeated writing of **signature_verification** and reports an exception.

Workaround

In the **[DEFAULT]** area of the **cinder.conf** file, set **verify_glance_signatures** to **disabled**. The following is an example:

```
[DEFAULT]
...
verify_glance_signatures = disabled
...
```

5.2 Failed to Create a VM on the Specified Host If iscsi_manager_groups Is Used When the iSCSI Network Is Used to Connect to Distributed Storage

Symptom

After **iscsi_manager_groups** is configured, OpenStack mounts volumes to a VM. At the beginning, the mounting is successful. However, volumes cannot be mounted to VMs from a certain time point, the Cinder driver does not report any error logs, and the iSCSI-based link fails to be established.

The target IP address returned to the Nova host from the Cinder volume log is obtained. On the Nova host, the **ping** *x.x.x.x* command is executed to check the IP address connectivity. In addition, whether the **iscsiadm** command can be used to establish a link is checked. The target IP address failed to be be pinged from the host.

Environment

Operating system: CentOS 7.6

Server: Huawei 5288 V5

OpenStack version: Pike and later

Root Cause

The data plane network between the Nova host and storage device is faulty.

Workaround

Rectify the data plane network fault and deliver a new VM.

5.3 When the SCSI Network Is Used to Connect to Distributed Storage, the number of concurrent I/Os to disks on VMs is limited to less than 64.

Symptom

When the SCSI network is used to connect to distributed storage, and the management plane is normal. However, on the service plane, the I/O size of the VM is limited to less than 64, and the VM cannot be pressed.

Environment

The Mode of OpenStack Networking: SCSI

Root Cause

The disk need to use libvirt for virtualization after mapping to a host machine, and then need to be generated a configuration file. When the SCSI network is used, the Nova does not have the "driver_io = native" configuration item. As a result, the disk I/O mode is threads (default), affecting the concurrent size.

Workaround

1. On each Nova node, add conf.driver_io = "native" to the get_config method of LibvirtVolumeDriver in nova.virt.libvirt.volume.volume.py.

2. Restart the nova-compute service.

systemctl restart openstack-nova-compute.service

6 Appendix

6.1 Appendix 1: Configuring Volume-backend image

6.2 Appendix 2: Configuring Image-Volume Cache

6.1 Appendix 1: Configuring Volume-backend image

This function is provided by the native OpenStack platform. It is used to store images in the storage backend of Cinder to accelerate the speed of creating volumes from images. Official link: https://docs.openstack.org/cinder/stein/admin/blockstorage-volume-backed-image.html

Step 1 Edit the /etc/glance/glance-api.conf file and complete the following configurations:

Configure Cinder as the storage backend of glance and configure the following content in [glance store]:

```
[glance_store]
...
stores = file,http,swift,cinder
default_store = cinder
```

Configure **show_multiple_locations** in [DEFAULT] to parse the locations URL information of Cinder.

```
[DEFAULT]
...
show_multiple_locations = True
```

Step 2 Edit the /etc/cinder/cinder.conf file and complete the following configurations:

Configure **allowed_direct_url_schemes** and **glance_api_version** to use Cinder as the backend of glance.

```
[DEFAULT]
...
glance_api_version=2
allowed_direct_url_schemes = cinder
```

Configure **image_upload_use_cinder_backend** at the backend, for example, [fusionstorage].

[fusionstorage]

image_upload_use_cinder_backend = True

Step 3 After the preceding configuration is complete, restart the cinder and glance services.

systemctl restart openstack-glance-* systemctl restart openstack-cinder-*

----End

6.2 Appendix 2: Configuring Image-Volume Cache

This function is provided by the native OpenStack platform. Image volume cache can significantly improve the performance of creating EVS disks from images. Official link: https://docs.openstack.org/cinder/stein/admin/blockstorage-image-volume-cache.html

Step 1 Obtain the project ID, for example, admin.

openstack project list | grep admin | 366d9f386ecd4defb91df5b0830e7267 | admin |

Step 2 Obtain a Cinder user ID.

openstack user list | grep cinder | 1b2a016cb4a74326a469834dfbc7a841 | cinder

Step 3 Edit the /etc/cinder/cinder.conf file.

Add cinder_internal_tenant_project_id and cinder_internal_tenant_user_id to [DEFAULT].

[DEFAULT]

cinder_internal_tenant_project_id = 366d9f386ecd4defb91df5b0830e7267 cinder_internal_tenant_user_id = 1b2a016cb4a74326a469834dfbc7a841

Configure **image_volume_cache_enabled** at the backend, for example, [fusionstorage].

[fusionstorage]

•••

image_volume_cache_enabled = True

Step 4 After the preceding configuration is complete, restart the cinder and glance services.

systemctl restart openstack-cinder-*