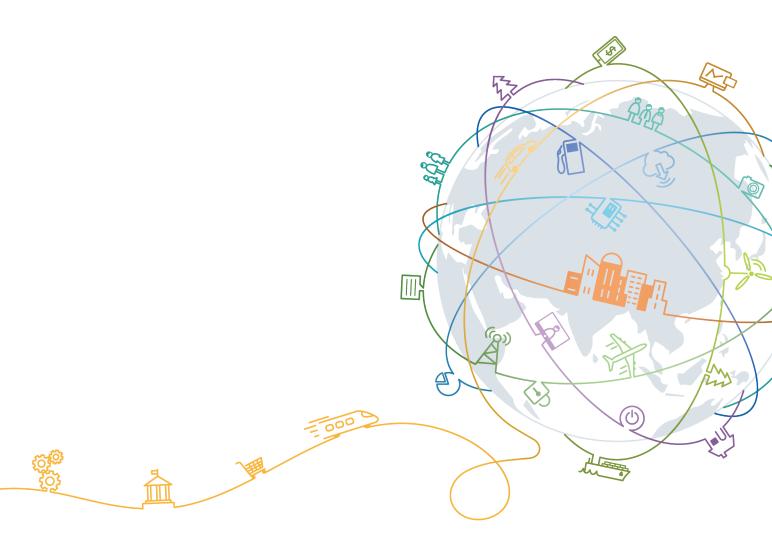
FusionStorage OpenStack Driver

Configuration Guide

Issue 01

Date 2020-04-10





Copyright © Huawei Technologies Co., Ltd. 2020. 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
3.3 Deploying FusionStorage Driver for the Cinder Volume Service in Containerized Mode	6
4 Configuring Basic Properties	7
4.1 Configuring a conf File	7
4.2 conf Configuration File Parameters	
4.3 Configuring iSCSI Multipathing	
4.4 Configuring the QoS Property	9
5 FAQ	12
5.1 Create Volume From Image Failed when verify glance signatures=enabled	12

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
Pike	FusionStorage V100R006C30
Queens	FusionStorage V100R008C00
Rocky	OceanStor 100D 8.0.2
Stein	
Train	

◯ NOTE

Pike and Queens only supports iSCSI protocol.

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

Feature	Pike	Queens	Rocky	Stein	Train
Create Volume	√	√	√	√	√
Delete Volume	√	√	√	√	√
Attach Volume	√	√	√	√	√
Detach Volume	√	√	√	√	√
Extend Volume	√	√	√	√	√
Create Snapshot	√	√	√	√	√
Delete Snapshot	√	√	√	√	√

Feature	Pike	Queens	Rocky	Stein	Train
Create Volume from Snapshot	√	√	√	√	√
Create Volume from Image	√	√	√	√	√
Create Volume from Volume	√	√	√	√	✓
Create Image from Volume	√	√	√	√	→
SmartThin	√	√	√	√	✓
Manage/Unmanage Volume	√	√	√	√	√
Manage/Unmanage Snapshot	√	√	√	√	√
Multipath	√	√	√	√	√
QoS	√	√	√	√	√
Retype	√	√	√	√	√
iSCSI	√	√	√	√	√
SCSI	х	х	√	√	√
MultiAttach	х	√	√	√	√
Revert to Snapshot	√	√	√	√	√
Backup Volume	√	√	√	√	√
HyperMetro	х	х	x	х	х
Replication V2.1	х	х	х	х	х
HyperMetro Consistency Group	x	х	х	x	х
Backup Snapshot	х	х	х	х	х
Snapshot Consistency Group	х	х	х	х	х
Consistency Group	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.

----End

□ 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

Red Hat OpenStack deployment steps are as follows:

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

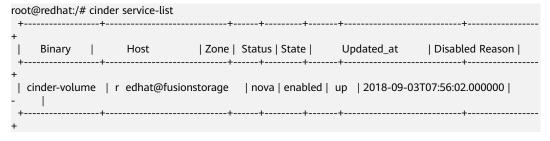
On the host, the version of Python is 2.7. If another version is used, use the correct version number. You can obtain the Cinder Driver installation directory by running the following commands:

root@redhatL004:~# find / -name dsware.py
/usr/lib/python2.7/dist-packages/cinder/volume/drivers/fusionstorage/dsware.py

- Step 2 Copy OpenStack Cinder Driver to the Cinder Driver installation directory.
- **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.



----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.

◯ 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 Huawei Driver in the container.
- **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.

----End

4 Configuring Basic Properties

This chapter describes how to configure the FusionStorage Cinder Driver.

- 4.1 Configuring a conf File
- 4.2 conf Configuration File Parameters
- 4.3 Configuring iSCSI Multipathing
- 4.4 Configuring the QoS Property

4.1 Configuring a conf File

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 an FSM, san_login and san_password are the login information of the FSM, and dsware_storage_pools indicates the storage pools name on the FSM. For details, see Table 4-1.

□ NOTE

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 4-1**.

□ NOTE

Once ISCSI mode is used, "volume_driver" needs to be configured as "cinder.volume.drivers.fusionstorage.dsware.DSWAREISCSIDriver" [fusionstorage]

...

 $volume_driver = cinder.volume.drivers.fusionstorage.dsware.DSWAREISCSIDriver$

...

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

[DEFAULT]

enabled_backends = fusionstorage

----End

4.2 conf Configuration File Parameters

Table 4-1 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 of FusionStorage is IPV6 format, please configure https: // [ipv6_ip]: 28443, where ipv6_ip is the management IP of FusionStorage in IPV6 format).	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 IP host is separated using commas (,). Add spaces at the start of each row.	Yes when selecting SCSI mode
target_ips	Management IP of the node where the service IP is located.	Yes when selecting iSCSI mode

Parameter	Description	Mandatory
scan_device_ti meout	The waitting time for scanning the device disk on the host. Default value is 3, the type is Int, the unit is seconds.	Option when selecting SCSI mode

4.3 Configuring iSCSI Multipathing

Step 1 Configure multiple IP addresses in the **target_ips** item of the Cinder Driver conf, and separate them with a comma (,).

[fusionstorage]
...
target_ips = x.x.x.x,y.y.y.y

Step 2 Enable multipathing for the OpenStack Nova module.

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 3 Restart the nova-compute service.

----End

4.4 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-gos.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 4-2 qos_key parameters

Parameter	Description	Remarks	Mandatory
maxIOPS	Maximum IOPS.	The value is a positive integer.	Yes
maxMBPS	Maximum bandwidth.	The value is a positive integer, expressed in MB/s.	Yes
minBaselineIOPS	Minimum IOPS.	The value is a positive integer.	No
minBaselineMBPS	Minimum bandwidth.	The value is a positive integer, expressed in MB/s.	No

Table 4-3 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

Parameter	Description	Remarks	Mandatory
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 Create Volume From Image Failed when verify_glance_signatures=enabled

5.1 Create Volume From Image Failed when verify_glance_signatures=enabled

Issue Description

OpenStack executes creating a volume from a image. After copying the image metadata for a period of time, an error is reported: Failed to copy metadata to volume: Glance metadata cannot be updated, key signature_verified exists for volume id b75f957c-0ba4-4099-8706-866c68012779, which eventually causes the volume creation to fail.

Environment

Operating System: CentOS 7.6

Server: Huawei 5288 V5 OpenStack Version: Rocky

Root Cause

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

A new field verify_glance_signatures has been added to cinder.image.glance.py since Rocky. The default value is enabled. When CONF.verify_glance_signatures! = 'disabled' and signature_verification in the image also exists, then the DB API does not allow duplicate writes signature_verification flag, thereby throwing an exception.

Solution

In [DEFAULT] of the cinder.conf file, set the verify_glance_signatures field to disabled, for example:

[DEFAULT]

...

verify_glance_signatures = disabled

...