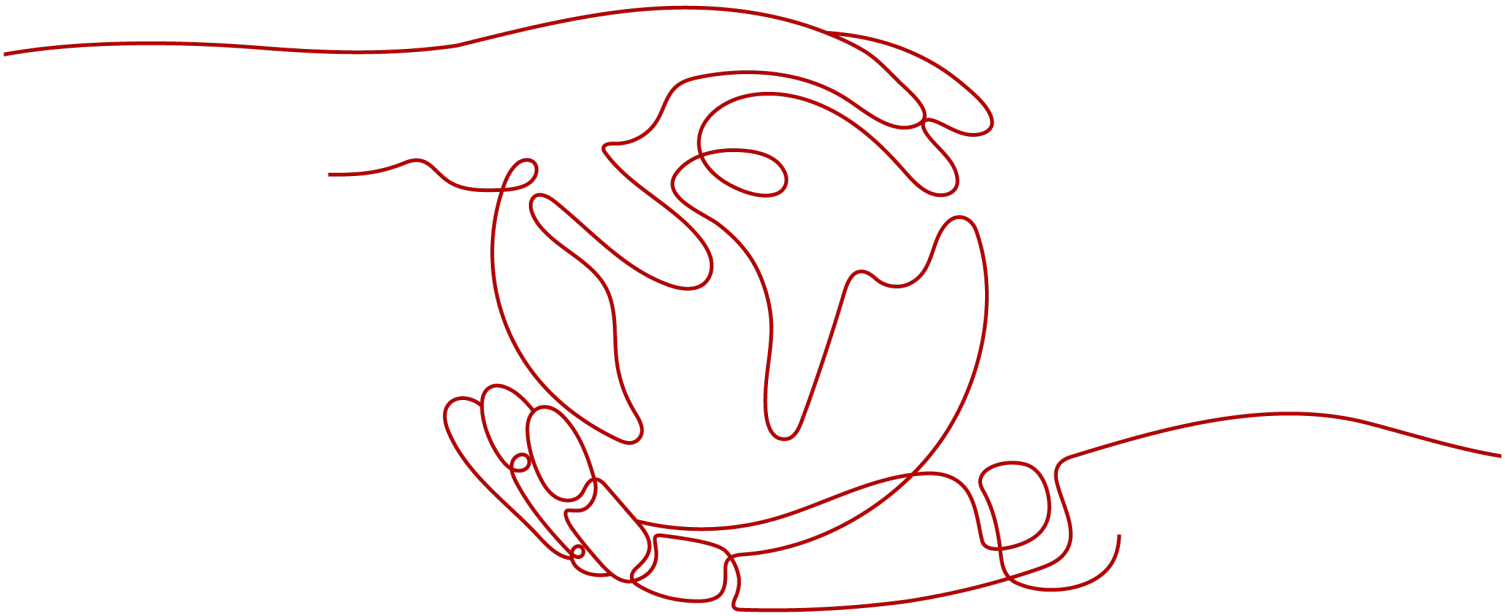
eSDK Enterprise Storage Plugins
2.5.1

OpenStack Manila Driver Configuration Guide

Issue 01

Date 2022-07-15



HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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://e.huawei.com

About This Document

Intended Audience

This document is intended for:

- Technical support engineers
- O&M engineers
- Engineers with basic knowledge of storage and OpenStack

Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
DANGER	Indicates a hazard with a high level of risk which, if not avoided, will result in death or serious injury.
WARNING	Indicates a hazard with a medium level of risk which, if not avoided, could result in death or serious injury.
CAUTION	Indicates a hazard with a low level of risk which, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a potentially hazardous situation which, if not avoided, could result in equipment damage, data loss, performance deterioration, or unanticipated results. NOTICE is used to address practices not related to personal injury.
NOTE	Supplements the important information in the main text. NOTE is used to address information not related to personal injury, equipment damage, and environment deterioration.

Change History

Issue	Date	Description
01	2022-07-15	This issue is the first official release.

Contents

1 Overview

This chapter describes the definition of Manila Driver.

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

2 Version Mapping

This chapter describes versions and functions supported by Manila Driver, and its version mappings with Huawei storage systems and OpenStack.

Table 2-1 Version mappings among the Manila Driver, Huawei storage system and OpenStack

OpenStack	Huawei Storage System
Train/Ussuri/ Victoria/Wallaby/ Xena/Yoga	• OceanStor V3/F V3 V300R006C60/V300R006C61 • OceanStor V5/F V5 V500R007C61/V500R007C70 • OceanStor V6 6.1.3 • OceanStor Dorado V3 V300R002C10/V300R002C20 • OceanStor Dorado V6 6.1.0/6.1.2/6.1.3

Table 2-2 Mappings among Manila Driver, features and the OpenStack version (√: supported, x: not supported)

Feature	Normal Share	HyperMetro Share	Remarks
OpenStack	Train/Ussuri/ Victoria/Wallaby/ Xena/Yoga	Train/Ussuri/ Victoria/Wallaby/ Xena/Yoga	NA
Create Share	√	√	NA
Delete Share	√	√	NA
Allow access	√	√	NA
Deny access	√	√	NA

Feature	Normal Share	HyperMetro Share	Remarks
Create Snapshot	√	√	HyperMetro shares support snapshot creation only on the local end of HyperMetro.
Delete Snapshot	√	√	HyperMetro shares support snapshot deletion only on the local end of HyperMetro.
Revert Snapshot	√	√	HyperMetro shares support snapshot rollback only on the local end of HyperMetro.
Extend Share	√	√	NA
Shrink Share	√	√	NA
Create Share from Snapshot	√	√	HyperMetro shares support snapshot-based share creation only on the local end of HyperMetro.
Manage/ Unmanage Share	√	x	The file system and sharing relationship managed by manila must be one-to-one. Only non-multi-tenant file systems and shares can be managed.
Manage/ Unmanage Snapshot	√	x	NA
Replication	√	x	NA

Feature	Normal Share	HyperMetro Share	Remarks
QoS	√	√	HyperMetro shares support SmartQoS configuration only on the local end of HyperMetro.
Multi-tenancy	√	x	NA
SmartThin/ SmartThick	√	√	NA
SmartCompression	√	√	NA
SmartDedupe	√	√	NA
Ensure Share	√	√	NA
Create a share on a certain storage pool	√	√	NA
Create a share with a certain disk type	√	√	NA
SectorSize	√	√	NA
SmartPartition	√	√	HyperMetro shares support SmartPartition configuration only on the local end of HyperMetro.
SmartCache	√	√	HyperMetro shares support SmartCache configuration only on the local end of HyperMetro.

NOTE

The OceanStor Dorado NAS architecture is changed. The features involved on the local end of HyperMetro are subject to the actual storage.

For OceanStor Dorado V6 6.1.2 and later versions, HyperMetro shares can be created and deleted, snapshots can be created, deleted, and rolled back, and share capacity can be expanded reduced after a primary/secondary switchover.

3 Specifications and Restrictions

Feature	Sub-feature	Description	Remarks
Mapping platform	Native OpenStack platform	OpenStack versions: Train/Ussuri/Victoria/ Wallaby/Xena/Yoga	-
Configuration	XML configuration file	Ensure that the storage pool used for configuring Cinder Driver exists on Huawei storage. Otherwise, create a storage pool. In addition, the type of this storage pool must be file storage service.	-
		All parameter values in the Cinder Driver configuration file cannot contain the following special characters: <>&'"	-
		Set **Product** based on the actual back-end storage model.	-
Policy	SmartQoS	Protection policies and restriction policies are mutually exclusive. If they are configured together, shares will fail to be created.	• Limit policies: **maxIOPS** and **maxBandWidth** • Protection policies: **minIOPS**, **minBandWidth**, and **latency**

		IOType is mandatory. If it is not configured in a QoS policy, shares will fail to be created.	
		OceanStor Dorado V6 6.1.0 supports only restriction policies and does not support protection policies.	

4 Deployment

4.1 Obtaining Manila Driver

Step 1 Open a browser and access the repository address at **https://github.com/Huawei/OpenStack_Driver**.

Step 2 Click **Download ZIP** to download the Driver package.

Step 3 Decompress the package.

Step 4 Find the **Manila** directory in the directory generated upon decompression. It contains Huawei Drivers for multiple OpenStack versions.

----End

NOTE

Huawei's OpenStack Driver repository maintains six stable versions to ensure long-term stable running of historical versions.

4.2 Deploying Manila Driver for the Manila Share Service

The OpenStack standard deployment steps are as follows:

Step 1 Search for the original Huawei Driver code in the system. The relative path is **.../manila/share/drivers/huawei**.

CAUTION

The complete path may vary depending on the operating system. You can search for the Huawei Driver code directory as follows:

- Method 1:

  Run the following command. **/usr/lib/python2.7/site-packages/manila/share/drivers/huawei** in the command output is the code directory.

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

- Method 2:

  Run the following command. **/usr/lib/python3.7/site-packages/manila/share/drivers/huawei** in the command output is the code directory.

  ```
  # python3 -c "from manila.share.drivers import huawei; print (huawei.__path__)"
  ['/usr/lib/python3.7/site-packages/manila/share/drivers/huawei']
  ```

Step 2 Copy the obtained OpenStack Manila Driver to the driver installation directory of the Manila node. For the default path, see **Step 1**. The following uses version X as an example. (Only copy and overwrite. Do not delete it.)

```
# ls -l
-rw-r--r-- 1 root root  4453 Mar 30 10:51 constants.py
-rw-r--r-- 1 root root 41592 Mar 30 10:51 helper.py
-rw-r--r-- 1 root root 10467 Mar 30 10:51 huawei_config.py
-rw-r--r-- 1 root root 92785 Mar 30 10:51 huawei_nas.py
-rw-r--r-- 1 root root  9791 Mar 30 10:51 huawei_utils.py
-rw-r--r-- 1 root root  7875 Mar 30 10:51 hypermetro.py
-rw-r--r-- 1 root root    23 Mar 30 10:51 __init__.py
-rw-r--r-- 1 root root  3380 Mar 30 10:51 manager.py
-rw-r--r-- 1 root root 13173 Mar 30 10:51 replication.py
-rw-r--r-- 1 root root  8171 Mar 30 10:51 rpcapi.py
-rw-r--r-- 1 root root  4491 Mar 30 10:51 smartx.py
```

----End

5 Configuring Basic Properties

This chapter describes how to configure Huawei Manila Driver.

NOTICE

- In OpenStack Ocata Manila, when you create a share without a share type, the default share type in the **/etc/manila/manila.conf** file will be used.
- Ensure that the storage pool used for configuring Cinder Driver exists in Huawei storage. Otherwise, create a storage pool. In addition, the type of this storage pool must be file storage service.
- All of the parameter values cannot include XML special character < > & ' "
- Snapshot and Replication can't be configured at the same time

5.1 Configuring Manila Driver

Procedure

Step 1 In **/etc/manila**, create a Huawei Manila Driver configuration file in **.xml** format. You can customize the configuration file name, for example, **manila_huawei_conf.xml**.

Step 2 Edit the **manila_huawei_conf.xml** file and set **mandatory** parameters in the Huawei Driver configuration file.

```
<?xml version='1.0' encoding='UTF-8'?>
 <config>
   <Storage>
     <Product>V3</Product>
     <LogicalPortIP>x.x.x.x</LogicalPortIP>
     <RestURL>https://x.x.x.x:8088/deviceManager/rest/</RestURL>
     <UserName>xxx</UserName>
     <UserPassword>xxx</UserPassword>
   </Storage>
   <Filesystem>
     <StoragePool>xxx</StoragePool>
```



```
  </Filesystem>
</config>
```

Table 5-1 Parameters in the configuration file

Parameter	Default Value	Description	Remarks
Product	V3	Storage product model. The value can be **V3**, **V5**, or **Dorado**.	-
LogicalPortIP	-	Logical port IP address. You can configure multiple IP addresses separated by semicolons (;).	This parameter is mandatory when DNS is not configured. If they are both configured, the priority of the parameter is lower than that of DNS.
DNS	-	DNS domain name. You can configure multiple DNSs separated by semicolons (;).	This parameter is mandatory when **LogicalPortIP** is not configured. If they are both configured, the priority of the parameter is higher than that of **LogicalPortIP**.
RestURL	-	Access address of the REST interface.	-
UserName	-	User name of an administrator.	-
UserPassword	-	Password of an administrator.	-
StoragePool	-	Name of a storage pool to be used.	-

Step 3 Edit the **manila_huawei_conf.xml** file and configure **optional** parameters in the Huawei Driver configuration file.

```
<?xml version='1.0' encoding='UTF-8'?>
 <Config>
   <Storage>
     <Port>abc;CTE0.A.H1</Port>
     <SnapshotSupport>True</SnapshotSupport>
     <ReplicationSupport>False</ReplicationSupport>
   </Storage>
   <Filesystem>
     <SectorSize>64</SectorSize>
     <WaitInterval>3</WaitInterval>
     <Timeout>60</Timeout>
     <SnapshotReserve>20</SnapshotReserve>
```



```
    <NFSClient>
      <IP>x.x.x.x</IP>
    </NFSClient>
    <CIFSClient>
      <UserName>xxx</UserName>
      <UserPassword>xxx</UserPassword>
    </CIFSClient>
  </Filesystem>
</Config>
```

Table 5-2 Parameters in the configuration file

Parameter	Default Value	Description	Type
Port	-	Port name list of bond port or ETH port, used to create vlan and logical port.	If <Port> is not configured, then will choose an online port on the array.
SectorSize	64	The size of the disk blocks, optional value can be "4", "8", "16", "32" or "64", and the units is KB.	Optional.
SnapshotSupport	True	Support snapshot or not.	This parameter cannot be enabled together with ReplicationSupport.
ReplicationSupport	False	Support replication or not.	This parameter cannot be enabled together with SnapshotSupport.
WaitInterval	3	Interval for querying file system status. The unit is second.	Optional.
Timeout	60	Timeout interval for waiting command execution of a storage device to complete. The unit is second.	Optional.
SnapshotReserve	20	The ratio of the snapshot size to the file system size(%). The range is (0~50).	Optional.
NFSClient\IP	-	Backend IP in admin network to use for mounting NFS share.	Mandatory when create a share from snapshot.

Parameter	Default Value	Description	Type
CIFSClient \UserName	-	Backend user name in admin network to use for mounting CIFS share.	Mandatory when create a share from snapshot. If OceanStor Dorado V6 is used, the user name must be added to a local authentication user group with administrator rights.
CIFSClient \UserPassword	-	Backend password in admin network to use for mounting CIFS share.	Mandatory when create a share from snapshot.

Step 4 Configure the **/etc/manila/manila.conf** file. Add the following backend configuration to the end of the file:

```
[huawei_backend]
share_driver = manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver
manila_huawei_conf_file = /etc/manila/manila_huawei_conf.xml
share_backend_name = huawei_backend
driver_handles_share_servers = False
```

NOTE

- Backend name **huawei_backend** can be customized.
- **share_driver** indicates the type of the driver to be used. Set this parameter to **manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver**.
- **manila_huawei_conf_file** indicates the path of the Manila Driver configuration file.
- **driver_handles_share_servers** indicates whether to enable multi-tenant.

In the **DEFAULT** area, add **huawei_backend** to **enabled_share_backends** item.

```
[DEFAULT]
...
enabled_share_backends=xxx,huawei_backend
```

Step 5 Change the permission on the **/etc/manila/manila_huawei_conf.xml** file to be the same as that on **/etc/manila/manila.conf**.

```
# ls /etc/manila/
...
-rw-r-----  1 manila manila 82179 Jan 20 14:34 manila.conf
-rw-r-----. 1 manila manila  1005 Jan 20 10:55 manila_huawei_conf.xml
...
```

Step 6 Restart the Manila Share service.

```
systemctl restart openstack-manila-share.service
```

----End

5.2 Verifying the Manila Share Service

Procedure

Step 1 Create a common share type. **general** indicates the name of the share type, which is specified by users. **False** indicates that multi-tenancy is not supported. To configure multi-tenancy, set this parameter to **True**.

```
# manila type-create general False
```

Step 2 Set the backend of the share type. **huawei_backend** indicates the backend name of the share, which is the value of **share_backend_name** in step 4 in **5.1 Configuring Manila Driver**.

```
# manila type-key general set share_backend_name=huawei_backend
```

Step 3 Set snapshot properties.

```
# manila type-key general set snapshot_support=True
```

Step 4 Set snapshot rollback properties.

```
# manila type-key general set revert_to_snapshot_support=True
```

Step 5 Sets the properties for creating a share using a snapshot.

```
# manila type-key general set create_share_from_snapshot_support=True
```

Step 6 Run the **manila create --name test001 NFS 2 --share-type general** command to create a share that supports the preceding properties.

----End

NOTE

Snapshot properties: This feature is mutually exclusive with remote replication and they cannot be configured at the same time (excluding OceanStor Dorado V6 NAS).

6 Configuring Advanced Properties

This chapter describes how to configure advanced storage properties.

Huawei storage supports advanced properties, such as Smartx. By associating with specified share types, these properties can be used in OpenStack.

6.1 Configuring Thin/Thick

Step 1 Configure Thin property: Run the following command to configure the key-value pair whose Thin property is **true**:

```
root@ubuntu:~manila type-key test_share_type set capabilities:thin_provisioning='<is> true'
```

Configure Thick property: Run the following command to configure the key-value pair whose Thin property is **false**:

```
root@ubuntu:~manila type-key test_share_type set capabilities:thin_provisioning='<is> false'
```


Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

NOTE

- If **capabilities:thin_provisioning** is set in the share type extra-specs, it will be used (thin share will be created if **capabilities:thin_provisioning=<is> true**, thick share will be created if **capabilities: thin_provisioning=<is> false**).
- If **capabilities:thin_provisioning** is not set in the share type extra-specs, thin share will be created.

6.2 Configuring SmartDedupe

Step 1 Run the following command to configure the key-value pair whose SmartDedupe property is **true**:

```
root@ubuntu:~#manila type-key test_share_type set capabilities:dedupe='<is> true'
```

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

NOTE

Only **Thin** share supports SmartDedupe.

Refer to section 4.1 to configure thin property.

6.3 Configuring SmartCompression

Step 1 Run the following command to configure the key-value pair whose SmartCompression is **true**:

```
root@ubuntu:~#manila type-key test_share_type set capabilities:compression='<is> true'
```

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

NOTE

Only **Thin** share supports SmartCompression.

Refer to section 4.1 to configure thin property.

6.4 Configuring SmartCache

Step 1 Run the following command to configure the key-value pair whose SmartCache property is **true**:

```
root@ubuntu:~#manila type-key test_share_type set capabilities:huawei_smartcache='<is> true'
```

Configure a name for the SmartCache existing on the storage device.

```
root@ubuntu:~#manila type-key test_share_type set huawei_smartcache:cachename='test_name'
```


Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.5 Configuring SmartPartition

Step 1 Run the following command to configure the key-value pair whose SmartPartition property is **true**:

```
root@ubuntu:~#manila type-key test_share_type set capabilities:huawei_smartpartition='<is> true'
```

Configure a name for the SmartPartition existing on the storage device.

```
root@ubuntu:~#manila type-key test_share_type set huawei_smartpartition:partitionname='test_name'
```

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.6 Configuring SmartQoS

About This Chapter

Huawei supports the following QoS properties. One or multiple properties in the same class can be configured in one QoS property simultaneously.

Protection policies: latency, minIOPS, minBandWidth

Restriction policies: maxIOPS, maxBandWidth

CAUTION

- Protection policies and restriction policies are mutually exclusive. If they are configured together, shares will fail to be created.
- OceanStor Dorado V6 6.1.0 and later versions support only restriction policies and do not support protection policies.
- **IOType** is mandatory. If it is not configured in a QoS policy, shares will fail to be created.

6.6.1 Configuring the Maximum Control IOPS

This section describes how to configure the maximum Control IOPS.

Procedure

Step 1 Run the following command to configure the key-value pair whose QoS property is **true**.

```
root@ubuntu:~# manila type-key test_share_type set capabilities:qos='<is> true'
```

Run the following command to configure QoS control property parameters.


```
root@ubuntu:~# manila type-key test_share_type set qos:IOType=0 qos:maxIOPS=50
```

- **maxIOPS**: indicates the maximum IOPS. The value is an integer larger than **0**. For OceanStor Dorado series, the valid value is an integer greater than or equal to 100.
- **IOType (**mandatory**)**: indicates the read and write type.**0** indicates the control read I/Os, **1** indicates the control write I/Os, **2** indicates the control read and write I/Os.

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.6.2 Configuring the Minimum Control IOPS

This section describes how to configure the minimum Control IOPS.

Procedure

Step 1 Run the following command to configure the key-value pair whose QoS property is **true**.

```
root@ubuntu:~# manila type-key test_share_type set capabilities:qos='<is> true'
```

Run the following command to configure QoS control property parameters.

```
root@ubuntu:~# manila type-key test_share_type set qos:IOType=0 qos:minIOPS=50
```

- **minIOPS**: indicates the minimum IOPS. The valid value is an integer larger than **0**. For OceanStor Dorado series, the valid value is an integer greater than or equal to 100.
- **IOType (**mandatory**)**: indicates the read and write type.**0** indicates the control read I/Os, **1** indicates the control write I/Os, **2** indicates the control read and write I/Os.

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.6.3 Configuring the Maximum Control Bandwidth

This section describes how to configure the maximum control bandwidth.

Procedure

Step 1 Run the following command to configure the key-value pair whose QoS property is **true**.

```
root@ubuntu:~# manila type-key test_share_type set capabilities:qos='<is> true'
```

Run the following command to configure QoS control property parameters.

```
root@ubuntu:~# manila type-key test_share_type set qos:IOType=0 qos:maxBandWidth=50
```

- **maxBandWidth**: indicates the maximum BANDWIDTH. The value is an integer larger than **0** and expressed in MB/s.

- **IOType (**mandatory**)**: indicates the read and write type.**0** indicates the control read I/Os, **1** indicates the control write I/Os, **2** indicates the control read and write I/Os.

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.6.4 Configuring the Minimum Control Bandwidth

This section describes how to configure the minimum control bandwidth.

Procedure

Step 1 Run the following command to configure the key-value pair whose QoS property is **true**.

```
root@ubuntu:~# manila type-key test_share_type set capabilities:qos='<is> true'
```

Run the following command to configure QoS control property parameters.

```
root@ubuntu:~# manila type-key test_share_type set qos:IOType=0 qos:minBandWidth=50
```

- **minBandWidth**: indicates the minimum BANDWIDTH. The value is an integer larger than **0** and expressed in MB/s.
- **IOType (**mandatory**)**: indicates the read and write type.**0** indicates the control read I/Os, **1** indicates the control write I/Os, **2** indicates the control read and write I/Os.

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.6.5 Configuring the Control Latency

This section describes how to configure the minimum control latency.

Procedure

Step 1 Run the following command to configure the key-value pair whose QoS property is **true**.

```
root@ubuntu:~# manila type-key test_share_type set capabilities:qos='<is> true'
```

Run the following command to configure QoS control property parameters.

```
root@ubuntu:~# manila type-key test_share_type set qos:IOType=0 qos:latency=50
```

- **minBandWidth**: indicates the minimum bandwidth. The value is an integer larger than **0** and expressed in MB/s.
- **IOType (**mandatory**)**: indicates the read and write type.**0** indicates the control read I/Os, **1** indicates the control write I/Os, **2** indicates the control read and write I/Os.

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.6.6 Configuring Multiple Control Policies

This section describes how to configure multiple control policies.

Procedure

Step 1 Run the following command to configure the key-value pair whose QoS property is **true**.

```
root@ubuntu:~# manila type-key test_share_type set capabilities:qos='<is> true'
```

Run the following command to configure QoS control property parameters.

```
root@ubuntu:~# manila type-key test_share_type set qos:IOType=0 qos:latency=50
```

- **latency**: indicates the latency. The value is an integer larger than **0**.
- **minIOPS**: indicates the minimum IOPS. The value is an integer larger than **0**.
- **minBandWidth**: indicates the minimum bandwidth. The value is an integer larger than **0**.
- **IOType (**mandatory**)**: indicates the read and write type.**0** indicates the control read I/Os, **1** indicates the control write I/Os, **2** indicates the control read and write I/Os.

Step 2 Run **manila create --name test001 NFS 2 --share-type test_share_type** to create a share that supports the preceding properties.

----End

6.7 Creating a Share on a Specified Storage Pool

This section describes how to create a share on a certain storage pool.

Procedure

Step 1 Run the **manila type-create target_pool_type False** command to create a share type. "target_pool_type"indicates the name of a share type. "False"indicates that the multi-tenant mode is not supported, when configured to "True" means that the multi-tenant mode is supported.

Step 2 Run the following command to configure the target storage pool.

- Configure a single target storage pool.
  ```
  root@ubuntu:~# manila type-key target_pool_type set pool_name=StoragePool001
  ```
- Configure multiple target storage pools.
  ```
  root@ubuntu:~# manila type-key target_pool_type set pool_name="<or> StoragePool001 <or> StoragePool002"
  ```

Step 3 Run **manila create --name test001 NFS 2 --share-type target_pool_type** to create a share that supports the preceding properties.

----End

6.8 Creating a Share with a Specified Disk Type

This section describes how to create a share with a certain disk type.

Procedure

Step 1 Run the **manila type-create disk_type False** command to create a share type. "disk_type"indicates the name of a share type. "False"indicates that the multi-tenant mode is not supported, when configured to "True" means that the multi-tenant mode is supported.

Step 2 Run the following command to configure the assign disk type.

- Configure one type of disk type.
  ```
  root@ubuntu:~# manila type-key disk_type set huawei_disk_type=sas
  ```
- Configure multiple types of disk type.
  ```
  root@ubuntu:~# manila type-key disk_type set huawei_disk_type="<or> sas <or> ssd"
  ```

Optional values of disk type can be "ssd", "sas", "nl_sas" or "mix", and the "mix" is a mixture of two or more in "ssd", "sas", "nl_sas".

Step 3 Run **manila create --name test001 NFS 2 --share-type disk_type** to create a share that supports the preceding properties.

----End

6.9 Configuring SectorSize

This section describes how to configure the size of blocks of filesystem.

Procedure

Step 1 Run the **manila type-create sectorsize_type False** command to create a share type. "sectorsize_type"indicates the name of a share type. "False"indicates that the multi-tenant mode is not supported, when configured to "True" means that the multi-tenant mode is supported.

Step 2 Run the following command to configure the key-value pair whose SectorSize property is **true**:

```
root@ubuntu:~# manila type-key sectorsize_type set capabilities:huawei_sectorsize='<is> true'
```

Configure the value of sectorsize:

```
root@ubuntu:~# manila type-key sectorsize_type set huawei_sectorsize:sectorsize=4
```

NOTE

`SectorSize` is the size of the disk blocks, optional value can be "4", "8", "16", "32" or "64", and the units is KB.

Step 3 Run **manila create --name test001 NFS 2 --share-type sectorsize_type** to create a share that supports the above attributes.

----End

NOTE

- If "sectorsize" is configured in both share_type and xml file, the value of sectorsize in the share_type will be used.
- If there is no value available in share-type and SectorSize is configured in the xml file at the same time. Take the SectorSize value in the xml file.
- If "sectorsize" is configured in neither share_type nor xml file, huawei storage backends will provide a default value(64) when creating a share.
- Rocky version and later versions do not support SectorSize

6.10 Configuring Replication

This section describes how to configure replication.

Prerequisites

Step 1 In file **/etc/manila/manila.conf**, configure two back ends with the remote replication relationship.

```
[DEFAULT]
...
enabled_share_backends = huawei_manila_1, huawei_manila_2
...

[huawei_manila_1]
share_driver = manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver
manila_huawei_conf_file = /etc/manila/manila_huawei_conf_1.xml
share_backend_name = huawei_manila_1
driver_handles_share_servers = False
replication_domain = huawei_domain
local_replication = False
replica_backend = host@huawei_manila_2

[huawei_manila_2]
share_driver = manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver
manila_huawei_conf_file = /etc/manila/manila_huawei_conf_2.xml
share_backend_name = huawei_manila_2
driver_handles_share_servers = False
replication_domain = huawei_domain
local_replication = False
replica_backend = host@huawei_manila_1
```

NOTE

- The replication_domain option is a backend specific option to be used within manila.conf. The value can be any ASCII string. Two backends that can replicate between each other would have the same replication_domain.
- The replica_backend option is set when the remote replication function is required. The value is a user-defined character string. The backends of the remote replication relationship are replica_backend.
- **manila_huawei_conf_1.xml** and **manila_huawei_conf_2.xml** are user-defined configuration files of the driver. For details, see **5 Configuring Basic Properties**.
- The 'local_replication' option should set to 'True' when using replication within array, and configure using the same array in 'manila_huawei_conf_1.xml' and 'manila_huawei_conf_2.xml'. If 'local_replication' option is not configured or its value is False, it means replication between arrays.

Step 2 Restart the Manila services.

----End

Procedure

Step 1 Run the **manila type-create replication_type False** command to create a share type. "replication_type" indicates the name of a share type. "False" indicates that the multi-tenant mode is not supported, when configured to "True" means that the multi-tenant mode is supported.

Step 2 Run the following command to configure the type of replication.

```
root@ubuntu:~# manila type-key replication_type set replication_type=dr
```

Step 3 Run **manila create --name test001 NFS 2 --share-type replication_type** to create a share that supports the preceding properties.

Step 4 Run **manila share-replica-create test001** to create replication for share test001.

NOTE

Now Huawei Manila Driver only supports **dr** style replication. For more details about **dr**, please refer to **http://docs.openstack.org/developer/manila/devref/share_replication.html**.

----End

6.11 Configuring the Owning Controller of an FS

This section describes how to configure the owning controller of an FS.

Procedure

Step 1 In this command, **controller_type** indicates the type of the shared controller, which is specified by the user. **False** indicates that multi-tenant is not supported. If you want to enable multi-tenant, change this value to **True**.

```
root@ubuntu:~# manila type-create controller_type_A False
```

Step 2 Run the following command to configure the key-value pair whose **Huawei_controller** attribute is **true**.

```
root@ubuntu:~# manila type-key controller_type_A set capabilities:huawei_controller='<is> true'
```

Step 3 Configure **controllername** that exists on the storage device and associate **controller_type_A** to the share type.

```
root@ubuntu:~# manila type-key controller_type_A set huawei_controller:controllername='CTE0.A'
```

Step 4 Run the **manila create --name test001 NFS 2 --share-type controller_type_A** command to create a share with the owning controller attribute specified.

----End

6.12 Configuring the NFS Client Permission

This section describes how to configure the NFS client permission.

Procedure

Step 1 In this command, **share_privilege_type** indicates the name of the share type, which is specified by the user. **False** indicates that multi-tenant is not supported. If you want to enable multi-tenant, change this value to **True**.


```
root@ubuntu:~# manila type-create share_privilege_type False
```

Step 2 Run the following command to configure the key-value pair whose **huawei_share_privilege** attribute is **true**.

```
root@ubuntu:~# manila type-key share_privilege_type set capabilities:huawei_share_privilege='<is> true'
```

Step 3 Configure the attribute name that exists on the storage device and associate to the share type.

Table 6-1

Name	Description	Value
sync	Write mode	0: synchronous 1: asynchronous
allsquash	Permission restriction	0: all_squash 1: no_all_squash
rootsquash	Root permission restriction	0: root_squash 1: no_root_squash
secure	Source Port Verification	0: secure 1: insecure

```
root@ubuntu:~# manila type-key share_privilege_type set huawei_share_privilege:sync=0
root@ubuntu:~# manila type-key share_privilege_type set huawei_share_privilege:allsquash=0
root@ubuntu:~# manila type-key share_privilege_type set huawei_share_privilege:rootsquash=0
root@ubuntu:~# manila type-key share_privilege_type set huawei_share_privilege:secure=0
```

Step 4 Run the **manila create --name test001 NFS 2 --share-type share_privilege_type** command to create a share with the owning controller attribute specified.

----End

----End

6.13 Configuring HyperMetro

This section describes how to configure HyperMetro.

Prerequisites

You have configured HyperMetro domains, vStores, vStore pairs, and logical ports available to vStores on the local and remote storage arrays.

Procedure

Step 1 In file **/etc/manila/manila.conf**, configure two backends with the NAS HyperMetro relationship.

```
[DEFAULT]
...
enabled_share_backends = huawei_manila_1, huawei_manila_2
```



```
...

[huawei_manila_1]
share_driver = manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver
manila_huawei_conf_file = /etc/manila/manila_huawei_conf_1.xml
share_backend_name = huawei_manila_1
driver_handles_share_servers = False
metro_info =
    metro_domain: huawei_domain,
    local_vStore_name: local_vstore,
    remote_vStore_name: remote_vstore,
    remote_backend: host@huawei_manila_2,
    metro_logic_ip: metro_logic_ip

[huawei_manila_2]
share_driver = manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver
manila_huawei_conf_file = /etc/manila/manila_huawei_conf_2.xml
share_backend_name = huawei_manila_2
driver_handles_share_servers = False
metro_info =
    metro_domain: huawei_domain,
    local_vStore_name: local_vstore,
    remote_vStore_name: remote_vstore,
    remote_backend: host@huawei_manila_1,
    metro_logic_ip: metro_logic_ip
```

NOTE

- **metro_info** is set when the NAS HyperMetro function is required. Its value is a user-defined list composed of multiple elements. Each element contains the **metro_domain**, **local_vStore_name**, **remote_vStore_name**, **remote_backend**, and **metro_logic_ip** fields, which are separated by commas (,). For details, see **Table 6-2**.
- **manila_huawei_conf_1.xml** and **manila_huawei_conf_2.xml** are user-defined configuration files of the driver. For details, see **5 Configuring Basic Properties**.
- Currently, shares cannot be created from the snapshots of HyperMetro shares.
- For OceanStor Dorado V6 6.1.2 and later versions, HyperMetro shares can be created and deleted, snapshots can be created, deleted, and rolled back, shares can be created using snapshots, and share capacity can be expanded reduced after a primary/secondary switchover.
- For OceanStor Dorado V6 6.1.2 and later versions, if HyperMetro is used and a backend is forcibly specified in **volume_type**, set the backend name in **type** to: **<or>** *backend1* **<or>** *backend2* (*backend1* and backend2 indicate the names of HyperMetro backends). The names of the configured backend storage pools must be the same. Otherwise, shares cannot be created using snapshots after a primary/secondary switchover.

Table 6-2 Parameter description

Parameter	Description
metro_domain	HyperMetro domain name. You must use the same value for backends with the HyperMetro relationship.
local_vStore_name	Local vStore name of the HyperMetro vStore pair.
remote_vStore_name	Remote vStore name of the HyperMetro vStore pair.
remote_backend	Name of the remote backend which belongs to the Manila backends composing a HyperMetro pair. Two backends with the NAS HyperMetro relationship are remote backends for each other. The format is *Host name*@*Backend name*.

Parameter	Description
metro_logic_ip	Logical IP address of the vStore configured for NAS HyperMetro. Two backends with the NAS HyperMetro relationship must be configured with the same IP address. You can configure multiple IP addresses separated by semicolons (;).

Step 2 Run the following command to restart the Manila service:

```
systemctl restart openstack-manila-volume.service
```

Step 3 Run the **manila type-create HyperMetro False** command to create the share type. **HyperMetro** indicates the name of the share type, which is specified by the user. **False** indicates that multi-tenant is not supported. If you want to enable multi-tenant, change this value to **True**.

Step 4 Run the following command to configure the key-value pair for the HyperMetro type:

```
root@ubuntu:~# manila type-key HyperMetro set capabilities:hypermetro='<is> true'
```

Step 5 Run the **manila create --name metro001 NFS 2 --share-type HyperMetro** command to create a share.

----End

6.14 Configuring filesystem:mode

This section describes how to configure the distribution algorithm for a shared source file system. Only OceanStor Dorado v6 supports the configuration.

Procedure

Step 1 Run the **manila type-create filesystem_mode_type False** command to create a share type. **filesystem_mode_type** indicates the name of the share type, which is specified by users. **False** indicates that multi-tenant is not supported. To configure multi-tenant, change it to **True**.

Step 2 Run the following command to set the value of **filesystem:mode**:

```
root@ubuntu:~# manila type-key filesystem_mode_type set filesystem:mode=0
```

NOTE

filesystem:mode=0 indicates the performance mode, and **filesystem:mode=2** indicates the directory balance mode.

Step 3 Run the **manila create --name test001 NFS 2 --share-type filesystem_mode_type** command to create a share that supports the **filesystem:mode** attribute.

----End

7 Best Practices

7.1 Quick Interconnection with Huawei Storage

Configuration Process

This section demonstrates how to configure Huawei Manila Driver on OpenStack to interconnect with Huawei Storage.

Step 1 Obtain Manila Driver (see chapter 3.1 for details).

Step 2 Create or query the file storage pool which will be used in Huawei storage.

Step 3 Configure file **manila.conf** and Huawei-defined configuration file of Driver. (For details, see chapter 4).

1. In **/etc/manila**, create a Huawei-defined Driver configuration file in **.xml** format. In this example, **manila_huawei_conf.xml** is used as the file name that can be changed based on actual conditions.
2. Set parameters for the created file.

```
<?xml version='1.0' encoding='UTF-8'?>
<Config>
  <Storage>
    <Product>V3</Product>
    <LogicalPortIP>x.x.x.x</LogicalPortIP>
    <RestURL>https://x.x.x.x:8088/deviceManager/rest/</RestURL>
    <UserName>xxx</UserName>
    <UserPassword>xxx</UserPassword>
    <SnapshotSupport>True</SnapshotSupport>
  </Storage>
  <Filesystem>
    <StoragePool>xxx</StoragePool>
  </Filesystem>
</Config>
```

3. Configure file **manila.conf**.

   At the end of file **/etc/manila/manila.conf**, add the following configuration item. In this configuration item, **volume_driver** indicates the loaded Driver file, and **manila_huawei_conf_file** indicates the Huawei-defined configuration file.


```
[huawei]
share_driver = manila.share.drivers.huawei.huawei_nas.HuaweiNasDriver
manila_huawei_conf_file = /etc/manila/manila_huawei_conf.xml
share_backend_name = huawei
driver_handles_share_servers = False
```

In the **[DEFAULT]** area, modify the configuration as follows to enable the huawei back-end:

```
[DEFAULT]
...
enabled_share_backends=huawei
```

Step 4 Restart the Manila service.

Step 5 Check the service status.

```
root@openstack:~# manila service-list
 +----+------------------+------------------+------+---------+-------+----------------------------+
 | Id | Binary           | Host             | Zone | Status  | State | Updated_at                 |
 +----+------------------+------------------+------+---------+-------+----------------------------+
 | 1  | manila-scheduler | openstack        | nova | enabled | up    | 2020-10-30T01:26:35.000000 |
 | 2  | manila-data      | openstack        | nova | enabled | up    | 2020-10-30T01:26:34.000000 |
 | 3  | manila-share     | openstack@huawei | nova | enabled | up    | 2020-10-30T01:26:34.000000 |
 +----+------------------+------------------+------+---------+-------+----------------------------+
```

In this example, the service status is **up**, indicating that the service is started correctly.

----End

