

eSDK Cloud Storage Plugins
2.7.RC2

OpenStack Manila Driver 配置指南 (For Intelligent Computing)

文档版本	01
发布日期	2024-09-30



版权所有 © 华为技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： <https://e.huawei.com>

安全声明

产品生命周期政策

华为公司对产品生命周期的规定以“产品生命周期终止政策”为准，该政策的详细内容请参见如下网址：
<https://support.huawei.com/ecolumnsweb/zh/warranty-policy>

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：
<https://www.huawei.com/cn/psirt/vul-response-process>
如企业客户须获取漏洞信息，请参见如下网址：
<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

华为初始证书权责说明

华为公司对随设备出厂的初始数字证书，发布了“华为设备初始数字证书权责说明”，该说明的详细内容请参见如下网址：
<https://support.huawei.com/enterprise/zh/bulletins-service/ENEWS2000015766>

华为企业业务最终用户许可协议(EULA)

本最终用户许可协议是最终用户（个人、公司或其他任何实体）与华为公司就华为软件的使用所缔结的协议。最终用户对华为软件的使用受本协议约束，该协议的详细内容请参见如下网址：
<https://e.huawei.com/cn/about/eula>

产品资料生命周期策略

华为公司针对随产品版本发布的售后客户资料（产品资料），发布了“产品资料生命周期策略”，该策略的详细内容请参见如下网址：
<https://support.huawei.com/enterprise/zh/bulletins-website/ENEWS2000017760>

前言





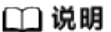
读者对象

本文档主要适用于以下读者对象：

- 技术支持工程师
- 运维工程师
- 具备存储和OpenStack基础知识的工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	表示如不避免则将会导致死亡或严重伤害的具有高等级风险的危害。
 警告	表示如不避免则可能导致死亡或严重伤害的具有中等级风险的危害。
 注意	表示如不避免则可能导致轻微或中度伤害的具有低等级风险的危害。
 须知	用于传递设备或环境安全警示信息。如不避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

修改记录

文档版本	发布日期	修改说明
01	2024-04-19	第一次正式发布。

目 录

前言..... iii

1 简介..... 1

2 版本配套关系..... 2

3 规格与限制..... 5

4 Manila Driver 安装与部署..... 7

4.1 Manila Driver 获取..... 7

4.2 Manila Share 服务华为 Manila Driver 部署..... 7

5 Manila Driver 基本属性配置..... 9

5.1 配置 Manila Driver XML 配置文件..... 9

5.2 配置 Manila Huawei Driver 后端..... 12

6 Manila Driver 升级..... 16

7 Manila Driver 特性配置..... 17

7.1 指定父 Share 创建 Share..... 17

7.2 指定分级容量创建 Share..... 18

7.3 指定分级放置策略创建 Share..... 18

7.4 指定分级迁移策略创建 Share..... 18

7.5 指定分级容量进行扩容..... 19

7.6 指定多协议创建 Share..... 19

7.7 指定协议为多协议 Share 添加授权..... 20

7.8 指定协议为多协议 Share 移除授权..... 20

1 简介

华为Manila Driver是部署在OpenStack Manila模块上的一个插件，用于对接华为存储向OpenStack平台提供共享存储能力。

2 版本配套关系

介绍华为Manila Driver与华为存储、OpenStack的版本配套关系。

表 2-1 Manila Driver 存储产品版本支持说明

OpenStack版本	存储产品版本	DME版本
中国移动苏研Pike版本	OceanStor Pacific 8.1.5/8.2.0/8.2.1	-

表 2-2 Manila Driver 特性支持说明（√：支持，x：不支持）

特性	NFS一级目录	NFS二级目录	DPC一级目录	DPC二级目录	GFS一级目录	GFS二级目录
Create Share	√	√	√	√	√	√
Delete Share	√	√	√	√	√	√
Allow Access ¹	√	√	√	√	√	√
Deny Access ¹	√	√	√	√	√	√
Update Access ¹	√	√	√	√	√	√
Extend Share	√	√	√	√	√	√
Shrink Share	√	√	√	√	x	x
Ensure Share	√	√	√	√	√	√

特性	NFS一级目录	NFS二级目录	DPC一级目录	DPC二级目录	GFS一级目录	GFS二级目录
QoS Freeze ²	√	x	√	x	√	√
QoS Thaw ²	√	x	√	x	√	√
Show QoS	√	x	√	x	x	x
Get All Share Usages	√	√	√	√	√	√
Get Share Usage	√	√	√	√	√	√
Modify Share Tier Policy	√	x	√	x	√	x
Initialize Share Tier	√	x	√	x	√	x
Get Share Tier Status	√	x	√	x	√	x
Terminate Share Tier	√	x	√	x	√	x
IPv4	√	√	√	√	√	x
IPv6	√	√	√ ³	√ ³	x	x

注释1：在DPC场景下，Allow Access、Deny Access、Update Access特性不支持只读（RO）权限，即使指定也不生效。在DPC二级目录场景下，所有的授权操作仍然是在一级目录上生效。

注释2：对于QoS Freeze、QoS Thaw特性，必须指定total_bytes_sec和total_iops_sec参数。QoS Freeze特性只支持将total_bytes_sec和total_iops_sec参数最小限制为1。

注释3：仅OceanStor Pacific 8.2.0L及之后的版本支持DPC IPv6授权。

说明

对非OpenStack社区标准的特性进行说明如下：

- **QoS Freeze**: QoS冻结，对share的带宽和IOPS进行限制。必须指定total_bytes_sec和total_iops_sec参数。QoS Freeze特性只支持将total_bytes_sec和total_iops_sec参数最小限制为1。NFS二级目录场景下不支持该特性。
- **QoS Thaw**: QoS解冻，恢复Share原有的Qos值。NFS二级目录场景下不支持该特性。
- **Show QoS**: 查询share对应的命名空间当前绑定的QoS值。
- **Get All Share Usages**: 查询当前存储所有一级和二级目录的总容量和可用容量信息后返回给上层应用进行缓存。用于获取单个目录容量时提升响应速度。
- **Get Share Usage**: 获取单个目录容量，包括总容量，已使用容量和可用容量。
- **Modify Share Tier Policy**: 修改share的分级策略。
- **Initialize Share Tier**: 创建share的预热/预冷任务
- **Get Share Tier Status**: 查询share的预热/预冷任务状态。
- **Terminate Share Tier**: 删除share的预热/预冷任务。

3 规格与限制

主特性	子特性	描述	备注
平台配套	智算项目 OpenStack平台 版本配套	中国移动苏研OpenStack版本：Pike	-
配置	XML配置文件	<ul style="list-style-type: none">Manila Driver配置使用的存储池，需要事先保证在华为存储上存在，否则需要手动创建。Product配置项当前仅支持如下两种配置<ul style="list-style-type: none">PacificPacific_GFS	华为Manila Driver配置文件所有参数值中不能含有XML特殊字符< > & ' "。
存储设备	存储池	分级场景下，一台OceanStor Pacific存储上的存储池只能有一个	-
规格	Pacific场景下的share配额	受限于OceanStor Pacific存储限制，最大支持100,000个的share的创建	包含一级目录和二级目录
share创建	Pacific_GFS场景下的share配额	<ul style="list-style-type: none">针对GFS，最大支持32个GFS的创建针对GFS的二级目录的share配额，最大支持100,000个	-
share创建	Pacific_GFS场景下的打散层级	最大支持6级目录深度打散	-
share授权	IP白名单授权配额	同一个share最大支持2,304个IP的授权	-

主特性	子特性	描述	备注
分级策略	分级策略配额	受限于OceanStor Pacific存储的限制，最大支持8,000个分级策略的创建	如果每个share一条分级放置策略，一条周期性分级迁移策略，一条一次性迁移策略, 则支持给8,000个share设置分级策略
分级迁移策略	分级迁移策略atime值	atime的取值范围为1 - 6120，单位为小时	6120为存储建议的最大值，超过6120也可以创建成功，插件不做限制
Driver后端	同一套存储支持对接的Driver后端数量	针对同一套存储，一个manila-share节点对接一个后端，最多支持三个manila-share节点	单个存储可对接的Driver后端数量与存储管理平面支持的最大并发请求相关。对接的Driver后端过多，会导致存储过载。

4 Manila Driver 安装与部署

4.1 Manila Driver获取

4.2 Manila Share服务华为Manila Driver部署

4.1 Manila Driver 获取

操作步骤

- 步骤1** 打开浏览器，访问仓库地址：https://github.com/Huawei/FusionStorage_OpenStack_Driver/releases/tag/PacificNas-V0.1
- 步骤2** 选择“Source code (zip)”，华为Manila Driver将以压缩包的形式下载到本地。
- 步骤3** 解压该压缩包。
- 步骤4** 在解压出的目录下找到“Manila/file_driver_dir/huawei”目录。
- 结束

4.2 Manila Share 服务华为 Manila Driver 部署

📖 说明

此部署操作基于社区OpenStack Pika 版本，将插件包的内容拷贝到社区指定的目录下。实际部署方式需要根据三方云平台自己定义的规则进行变化。

例如如果是使用的Docker或K8s部署的Manila Share服务，需要进入对应的容器找到华为Manila Driver代码安装目录

操作步骤

- 步骤1** 查找系统中原有的华为Manila Driver代码目录。
- 完整路径在不同的系统下可能有所不同，可通过以下方法搜索华为Manila Driver代码安装目录：
- 执行以下命令，回显中的“/usr/lib/python2.7/site-packages/manila/share/drivers/huawei”即为代码安装目录。

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

- 执行以下命令，回显中的“/usr/lib/python3.7/site-packages/manila/share/drivers/huawei”即为代码安装目录。

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

步骤2 将从[4.1 Manila Driver获取](#)获取到的OpenStack Manila Driver的huawei目录下的所有内容拷贝到Manila Share节点华为Manila Driver代码安装目录下，默认路径参考[步骤1](#)，复制后结果如下。Driver所有目录的权限最小为755，所有文件的权限最小为644。

```
[root@pnode1 huawei(keystone_admin)]# pwd  
/usr/lib/python2.7/site-packages/manila/share/drivers/huawei  
[root@pnode1 huawei(keystone_admin)]# ls -l  
drwxr-xr-x 1 root root 204 Mar 07 10:51 oceanstorPacific  
-rw-r--r-- 1 root root 23 Mar 07 10:51 __init__.py
```

----结束

5 Manila Driver 基本属性配置

介绍如何配置华为Manila Driver。

须知

- OpenStack Ocata Manila创建共享时，如未指定share type，会使用“/etc/manila/manila.conf”文件中的默认share type。
- Manila Driver配置使用的存储池，需要事先保证在华为存储上存在，否则需要手动创建，存储池类型必须是“文件存储服务”，且存储池名不能包含中文。
- 华为Manila Driver配置文件所有参数值中不能含有XML特殊字符< > & ' "。

5.1 配置Manila Driver XML配置文件

5.2 配置Manila Huawei Driver后端

5.1 配置 Manila Driver XML 配置文件

操作步骤

步骤1 在“/etc/manila”路径下创建华为Manila Driver配置文件，文件格式为xml。配置文件名称可自定义，例如“manila_huawei_conf.xml”。

步骤2 编辑“manila_huawei_conf.xml”文件，配置华为Manila Driver配置文件参数。

```
<?xml version='1.0' encoding='UTF-8'?>
<config>
  <Storage>
    <Product>Pacific</Product>
    <RestURL>https://ip:8088/api/v2/</RestURL>
    <UserName>xxx</UserName>
    <UserPassword>xxx</UserPassword>
    <Reserved_percentage>15</Reserved_percentage>
    <Max_over_subscription_ratio>1</Max_over_subscription_ratio>
    <Semaphore>10</Semaphore>
    <HotDiskType>ssd</HotDiskType>
    <WarmDiskType>sas</WarmDiskType>
    <ColdDiskType>sata</ColdDiskType>
  </Storage>
  <Filesystem>
    <StoragePool>0</StoragePool>
  </Filesystem>
</config>
```

```
<AccountName>system</AccountName>
<ClusterDomainName>nfs-ip</ClusterDomainName>
</Filesystem>
<DPC>
  <MountOption>cid={sn},rw</MountOption>
</DPC>
<NFS>
  <MountOption>nfsvers=3</MountOption>
</NFS>
</config>
```

表 5-1 参数说明

参数	说明	示例
Product	必填参数，存储产品类型，当前支持如下两种： <ul style="list-style-type: none">• Pacific• Pacific_GFS	Pacific
RestURL	必填参数，REST接口访问地址。	根据Product的取值略有差异。 <ul style="list-style-type: none">• Product的取值为Pacific时： https://ip:8088/api/v2/• Product的取值为Pacific_GFS时： https://ip:26335
UserName	必填参数，对接的后端的用户名。	-
UserPassword	必填参数，对接的后端的密码。	-
Reserved_percentage	选填参数，空间预留百分比，不填写默认为15，仅支持数字。	15
Max_over_subscription_ratio	选填参数，最大超分比，不填写默认为1，仅支持数字。 用户可通过修改该参数来实现容量超分配。	1
Semaphore	选填参数，用于在Pacific_GFS场景下控制对DME请求的并发量，不配置时默认为10，最大不能超过10	10

参数	说明	示例
HotDiskType	条件必填参数，分级场景下必填。 热硬盘池对应的硬盘类型，不配置默认为ssd, 配置值必须在以下范围内: [ssd, sas, sata] 说明 配置时以存储实际硬盘池硬盘类型对应的分级属性为准	ssd
WarmDiskType	条件必填参数，分级场景下必填。 温硬盘池对应的硬盘类型，不配置默认为sas, 配置值必须在以下范围内: [ssd, sas, sata] 说明 配置时以存储实际硬盘池硬盘类型对应的分级属性为准	sas
ColdDiskType	选填参数，冷硬盘池对应的硬盘类型，不配置默认为sata, 配置值必须在以下范围内: [ssd, sas, sata] 说明 配置时以存储实际硬盘池硬盘类型对应的分级属性为准	sata
StoragePool	必填参数，支持配置多个，使用“;” 隔开。 <ul style="list-style-type: none">Product的取值为Pacific时，填写存储池ID，仅支持数字。Product的取值为Pacific_GFS时，填写集群分类名称。	根据Product的取值略有差异。 <ul style="list-style-type: none">Product的取值为Pacific时： 0;1Product的取值为Pacific_GFS时： cluster1;cluster2
AccountName	条件必填参数，存储租户名称。 <ul style="list-style-type: none">Product的取值为Pacific时，必填。Product的取值为Pacific_GFS时，不需要填写。	system
ClusterDomainName	条件必填参数，用户挂载共享路径的业务IP或域名，NFS场景下需要配置，DPC场景下不需要配置。	-

参数	说明	示例
DPC/MountOption	选填参数，通过此配置可以使创建share时，Driver返回的export_locations会带有-o属性用于后续执行mount命令时指定mount options。 除了标准的mount options, DPC场景下支持配置cid={sn}用于支持多集群挂载。	cid={sn},rw
NFS/MountOption	选填参数，通过此配置可以使创建share时，Driver返回的export_locations会带有-o属性用于后续执行mount命令时指定mount options。 NFS场景下当前仅支持标准的mount options。	nfsvers=3

步骤3 修改 “/etc/manila/manila_huawei_conf.xml” 文件的权限，使其与 “/etc/manila/manila.conf” 文件的权限一致。

```
# 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  
...
```

----结束

5.2 配置 Manila Huawei Driver 后端

操作步骤

步骤1 配置 “/etc/manila/manila.conf” 文件，在文件最后添加如下后端配置。配置参数请参考[表5-2](#)。

表 5-2 参数说明

参数	说明	示例
share_driver	必填参数，表示对接的华为后端存储的driver类型，用于启动服务时找到对应的华为存储后端驱动。	manila.share.drivers.huawei.oceanstorPacific.customization_nas_driver.HuaweiNasDriverForSuyan
manila_huawei_conf_file	必填参数， 5.1 配置Manila Driver XML配置文件 配置的华为Manila Driver配置文件路径。	/etc/manila/manila_huawei_conf.xml
share_backend_name	必填参数，share后端名称，用于后续在创建share_type时指定share_backend_name的值。	huawei_backend

参数	说明	示例
driver_handles_share_servers	必填参数，是否开启OpenStack租户网络隔离。当前该参数不支持设置为True，仅支持设置为False。	False
pool_qos_params	<p>选填参数，此参数用于配置此存储各个资源池的类型和qos计算公式的系数，例如XML中配置了两个存储池'0,1,2,3'，则需要为这四个存储池设置资源池类型和qos系数, 详细参数说明见表5-3</p> <p>说明 不配置时，表示不限制qos，为兼容老版本，qos统一设置为无限制</p>	<pre>{resource_pool:0;pool_type:NFS_HDD;qos_coefficients:200-20-10000}; {resource_pool:1;pool_type:NFS_SSD;qos_coefficients:0-120-25000}; {resource_pool:2;pool_type:DPC_SSD;qos_coefficients:0-480-130000}; {resource_pool:3;pool_type:NFS_HDD&DPC_SSD;qos_coefficients:200-20-10000&0-480-130000}</pre>

表 5-3 参数说明

参数	说明	示例
resource_pool	必填参数 Product为Pacific时，此处填写存储池的ID	<ul style="list-style-type: none">0
pool_type	<p>必填参数</p> <p>此处填写当前资源池的类型，当前仅支持如下三种类型:</p> <p>NFS_HDD: 普通性能型，磁盘全部是HDD盘，且只使用NFS协议时配置此类型</p> <p>NFS_SSD: 高性能型，磁盘全部为SSD盘，且只使用NFS协议时配置此类型</p> <p>DPC_SSD: 高性能型增强版，磁盘全部为SSD, 且只使用DPC协议时配置此类型</p> <p>说明 针对智算三期分级存储，由于存储池只会有一个，资源池类型固定为NFS_HDD&DPC_SSD，顺序和名称不可修改，分别对应冷池和热池</p>	<ul style="list-style-type: none">NFS_HDDNFS_SSDDPC_SSDNFS_HDD&DPC_SSD

参数	说明	示例
qos_coefficients	<p>必填参数</p> <p>此处配置qos计算公式的系数，需要配置三个整数值，分别对应常量系数，容量系数以及最大限制值，不可多配或者少配，且顺序不可调换，需要和pool_type形成一一映射关系</p> <p>示例中的配置值对应的公式如下：</p> <p>200-20-10000: $\min(200 + 20 * \{share_size\}, 10000)$</p> <p>0-120-25000: $\min(120 * \{share_size\}, 25000)$</p> <p>0-480-130000: $\min(480 * \{share_size\}, 130000)$</p> <p>说明</p> <ol style="list-style-type: none">上述公式中的share_size单位为TB，表示创建的share的容量大小上述公式计算出的值表示的最大带宽，单位为MBPSIOPS 由于没有计算公式，不进行限制针对智算三期分级存储，需要配置2个，用'&'符号隔开，计算公式如下： $\min(200 + 20 * \{cold_data_size\}, 10000) + \min(480 * \{hot_data_size\}, 130000)$示例中的配置值为云能和华为侧对齐的qos容量计算公式，智算局点按照示例配置即可，其它局点可根据和华为对齐的结论自行修改	<ul style="list-style-type: none">200-20-100000-120-250000-480-130000200-20-10000&0-480-130000

```
[huawei_backend]
share_driver =
manila.share.drivers.huawei.oceanstorPacific.customization_nas_driver.HuaweiNasDriverForSuyan
manila_huawei_conf_file = /etc/manila/manila_huawei_conf.xml
share_backend_name = huawei_backend
driver_handles_share_servers = False
pool_qos_params =
    {resource_pool:0;pool_type:NFS_HDD;qos_coefficients:200-20-10000};
    {resource_pool:1;pool_type:NFS_SSD;qos_coefficients:0-120-25000};
    {resource_pool:2;pool_type:DPC_SSD;qos_coefficients:0-480-130000};
    {resource_pool:3;pool_type:NFS_HDD&DPC_SSD;qos_coefficients:200-20-10000&0-480-130000}
```

步骤2 在[DEFAULT]区块中修改enabled_share_backends配置项，新增huawei_backend后端。

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

步骤3 执行以下命令，重新启动Manila服务。

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

步骤4 等待一段时间后（一般30秒左右），执行如下命令。查看对应的华为后端share服务State的值为up，表示Manila Share服务启动成功。

```
# manila service-list
```

----结束

6 Manila Driver 升级

OpenStack Manila Driver升级场景，请参考该章节；如果是第一次安装部署Manila Driver，请跳过该章节。

操作步骤

步骤1 查找系统中原有的华为Manila Driver代码目录。

完整路径在不同的系统下可能有所不同，可通过以下方法搜索华为Manila Driver代码目录：

- 执行以下命令，回显中的“/usr/lib/python2.7/site-packages/manila/share/drivers/huawei”即为代码目录。

```
# python -c "from manila.share.drivers import huawei; print (huawei.__path__)"  
['/usr/lib/python2.7/site-packages/manila/share/drivers/huawei']
```
- 执行以下命令，回显中的“/usr/lib/python3.7/site-packages/manila/share/drivers/huawei”即为代码目录。

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

步骤2 备份**步骤1**中返回目录下所有的华为Manila Driver代码文件。

备份命令参考如下：

```
cp -rf 源目录 备份目录
```

说明

- 源目录：**步骤1**中返回的目录。
- 备份目录：用户希望备份到的目录。

步骤3 将对应版本的华为Manila Driver代码文件全部拷贝到**步骤1**中返回的目录中。

步骤4 执行如下命令，重启Manila Share服务。

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

等待一段时间后（一般30秒左右），执行**manila service-list**命令。查看对应的华为后端share服务**State**的值为**up**，表示Manila Share服务启动成功。

----结束

7 Manila Driver 特性配置

通过本节，可以直接通过命令行的方式完成Manila Driver的一些定制特性的配置。

须知

以下所有的特性，均是基于社区接口，通过在metadata中设置元数据来实现二级目录、分级、多协议的支持，因为metadata是可以直接修改的，为保证数据一致性，使用如下特性时，请不要随意修改metadata中元数据的值。

[7.1 指定父Share创建Share](#)

[7.2 指定分级容量创建Share](#)

[7.3 指定分级放置策略创建Share](#)

[7.4 指定分级迁移策略创建Share](#)

[7.5 指定分级容量进行扩容](#)

[7.6 指定多协议创建Share](#)

[7.7 指定协议为多协议Share添加授权](#)

[7.8 指定协议为多协议Share移除授权](#)

7.1 指定父 Share 创建 Share

可以在已创建的share的基础上，创建一个子share。父share为一级目录，子share为二级目录。

操作步骤

步骤1 执行**manila create**命令创建一个普通的share。

```
manila create NFS 10 --name parent_share --share-type <share_type>
```

步骤2 获取**步骤1**中创建的share的UUID, 例如 ‘4341c4ee-29ab-473a-8337-dfef652af395’。

步骤3 执行**manila create**命令创建一个子share，命令中添加**metadata**参数，**metadata**属性中指定**parent_share_id**。

```
manila create NFS 10 --name sub_share --share-type <share_type> --metadata  
parent_share_id=4341c4ee-29ab-473a-8337-dfef652af395
```

步骤4 执行**manila list**命令查看share状态。状态为available表示创建成功。

4341c4ee-29ab-473a-8337-dfef652af395	parent_share	10	NFS	available
53969233-80c6-4040-a673-6d0355a015a8	sub_share	10	NFS	available

----结束

7.2 指定分级容量创建 Share

在创建一级目录时，支持设置**hot_data_size**用于指定放置到热硬盘池的容量。

操作步骤

步骤1 执行**manila create**命令创建一个普通的share。命令中添加**metadata**参数，**metadata**属性中指定**hot_data_size**。

```
manila create NFS 10 --name parent_share --share-type <share_type> --metadata hot_data_size=5
```

步骤2 执行**manila list**命令查看share状态。状态为available表示创建成功。

4341c4ee-29ab-473a-8337-dfef652af395	parent_share	10	NFS	available
--------------------------------------	--------------	----	-----	-----------

----结束

7.3 指定分级放置策略创建 Share

在创建一级目录时，支持设置**tier_place**用于指定分级放置的策略。

说明

tier_place表示要放置的硬盘池，支持三种：

- hot：表示放置到热硬盘池
- warm：表示放置到温硬盘池
- cold：表示放置到冷硬盘池

操作步骤

步骤1 执行**manila create**命令创建一个普通的share。命令中添加**metadata**参数，**metadata**属性中指定**tier_place**。

```
manila create NFS 10 --name parent_share --share-type <share_type> --metadata tier_place=hot
```

步骤2 执行**manila list**命令查看share状态。状态为available表示创建成功。

4341c4ee-29ab-473a-8337-dfef652af395	parent_share	10	NFS	available
--------------------------------------	--------------	----	-----	-----------

----结束

7.4 指定分级迁移策略创建 Share

在创建一级目录时，支持设置**tier_migrate_expiration**用于指定分级迁移的策略。

说明

tier_migrate_expiration为数据最后访问时间，表示达到该访问时间时，数据一直没有被使用，会将数据往冷硬盘池进行迁移。单位为小时，取值范围为0 ~ 6120。

操作步骤

- 步骤1** 执行**manila create**命令创建一个普通的share。命令中添加**metadata**参数，**metadata**属性中指定**tier_migrate_expiration**。

```
manila create NFS 10 --name parent_share --share-type <share_type> --metadata  
tier_migrate_expiration=2400
```

- 步骤2** 执行**manila list**命令查看share状态。状态为available表示创建成功。

```
4341c4ee-29ab-473a-8337-dfef652af395 | parent_share | 10 | NFS | available
```

----结束

7.5 指定分级容量进行扩容

在对一级目录进行扩容时，通过修改**metadata**中的**hot_data_size**，可以实现对share的热数据容量进行扩容。

说明

对**hot_data_size**进行扩容时，新增的热数据的容量不能大于新增的总容量。

操作步骤

- 步骤1** 执行**manila metadata**命令修改**metadata**中的**hot_data_size**的值，例如需要将share从10 GB扩容到20 GB，其中热数据容量从5 GB扩容到10 GB。

```
manila metadata 4341c4ee-29ab-473a-8337-dfef652af395 set hot_data_size=10
```

- 步骤2** 执行**manila extend**命令对share进行扩容。

```
manila extend 4341c4ee-29ab-473a-8337-dfef652af395 20
```

- 步骤3** 执行**manila list**命令查看share状态。状态为available表示扩容成功。

```
4341c4ee-29ab-473a-8337-dfef652af395 | parent_share | 20 | NFS | available
```

----结束

7.6 指定多协议创建 Share

在创建share时，可创建同时支持DPC和NFS协议的share。

说明

在**metadata**中指定**share_proto**时，优先以**metadata**中指定的协议为准，当前支持设置如下三种：

- NFS
- DPC
- NFS_DPC

操作步骤

- 步骤1** 执行**manila create**命令创建一个普通的share。命令中添加**metadata**参数，**metadata**属性中指定**share_proto**。

```
manila create NFS 10 --name parent_share --share-type <share_type> --metadata share_proto=NFS_DPC
```


步骤2 执行**manila list**命令查看share状态。状态为available表示创建成功。

```
4341c4ee-29ab-473a-8337-dfef652af395 | parent_share | 10 | NFS | available
```

----结束

7.7 指定协议为多协议 Share 添加授权

在为多协议的share添加授权时，支持指定其中一种协议类型进行授权。

须知

多协议场景下，授权时如果不指定协议类型，将会将权限同时授权给多个协议，因此多协议场景下建议指定当前权限需要授权的协议，一次只能指定一个，当前支持指定如下两种：

- NFS
- DPC

操作步骤

步骤1 执行**manila metadata**命令为多协议的share指定**access_proto**，表示要授权的协议类型。

```
manila metadata 4341c4ee-29ab-473a-8337-dfef652af395 set access_proto=DPC
```

步骤2 执行**manila access-allow**命令，为share添加客户端IP权限。

```
manila access-allow 4341c4ee-29ab-473a-8337-dfef652af395 ip x.x.x.x
```

步骤3 执行**manila access-list**命令查看当前share的授权列表。授权信息存在且state为active表示授权成功。

```
id | access_type | access_to | access_level | state
+-----+-----+-----+-----+-----+
4e6ab6cd-6500-4d13-b3ef-06c3109871c0 | ip | x.x.x.x | rw | active
```

----结束

7.8 指定协议为多协议 Share 移除授权

在为多协议的share移除授权时，支持指定其中一种协议类型移除授权。

须知

多协议场景下，移除授权时，需要保证share的metadata的access_proto和要移除的授权的协议一致。

操作步骤

步骤1 执行**manila access-list**命令查看当前share的授权列表。

```
id | access_type | access_to | access_level | state
+-----+-----+-----+-----+-----+
4e6ab6cd-6500-4d13-b3ef-06c3109871c0 | ip | x.x.x.x | rw | active
```

步骤2 执行**manila metadata**命令为多协议的share指定**access_proto**，表示要移除权限的协议类型。

```
manila metadata 4341c4ee-29ab-473a-8337-dfef652af395 set access_proto=DPC
```

如果不清楚当前要移除权限对应的协议类型，执行如下命令移除metadata中的**access_proto**。

```
manila metadata 4341c4ee-29ab-473a-8337-dfef652af395 unset access_proto
```

步骤3 执行**manila access-deny**命令，为share移除客户端IP权限。

```
manila access-deny 4341c4ee-29ab-473a-8337-dfef652af395 4e6ab6cd-6500-4d13-b3ef-06c3109871c0
```

步骤4 执行**manila access-list**命令查看当前share的授权列表。授权信息不存在表示移除授权成功。

id	access_type	access_to	access_level	state
-----	-----	-----	-----	-----

----结束