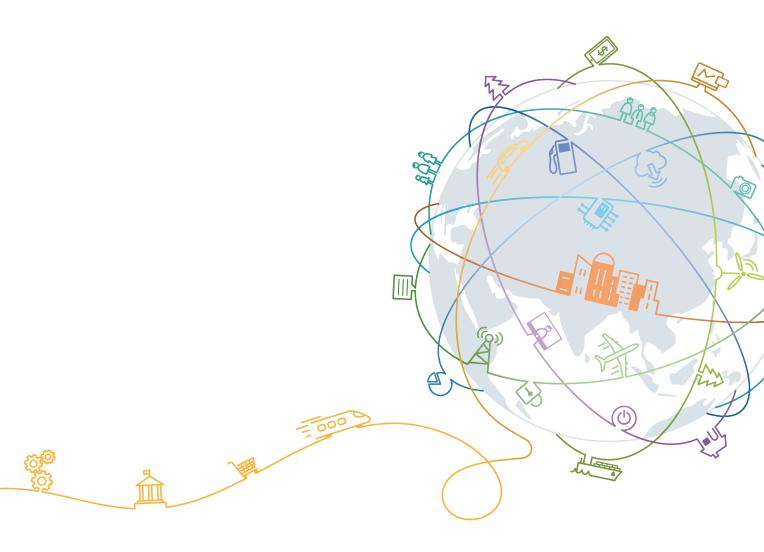
## **OpenStack Cinder Driver**

## 配置指南

文档版本 01

发布日期 2019-12-07





#### 版权所有 © 华为技术有限公司 2019。 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默示的声明或保证。

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://www.huawei.com客户服务邮箱:support@huawei.com

客户服务电话: 4008302118

## 目录

1 简介	
2 版本配套关系	2
3 Cinder Driver 安装与部署	
3.1 华为 Cinder Driver 获取	
3.2 非容器化 Cinder Volume 服务华为 Driver 部署	
3.3 容器化 Cinder Volume 服务华为 Driver 部署	
4 Cinder Driver 基本属性配置	8
4.1 iSCSI 组网 Cinder Driver 配置	
4.1.1 配置 TV1 存储 iSCSI 协议 Cinder Driver	
4.1.2 配置其他存储 iSCSI 协议 Cinder Driver	10
4.1.3 配置 iSCSI 多路径	13
4.1.4 配置 CHAP 和 ALUA 特性	14
4.2 配置 FC 组网 Cinder Driver	
4.2.1 配置 TV1 存储 FC 协议 Cinder Driver	15
4.2.2 配置其他存储 FC 协议 Cinder Driver	17
4.2.3 配置智能划域	19
4.2.4 配置 FC 多路径	
4.2.5 配置 ALUA 特性	20
4.3 配置多后端	21
4.4 配置租户用户	22
5 Cinder Driver 高级特性配置	23
5.1 配置 Thick 属性	24
5.2 配置 Thin 属性	24
5.3 配置 SmartQoS 属性	24
5.4 配置 SmartPartition 属性	25
5.5 配置 SmartCache 属性	26
5.6 配置 SmartTier 属性	26
5.7 配置双活属性	26
5.8 配置远程复制属性	27
5.9 指定存储池创建 LUN	29
5.10 指定磁盘类型创建 LUN	29
5.11 指定后端创建 LUN	29

5.12 配置 Backup Snapshot	30
5.13 配置证书认证	30
5.14 配置 LUN 归属控制器	30
5.15 配置 SmartDedupe 属性	31
5.16 配置 SmartCompression 属性	32
5.17 使用 fastclone 方式创建卷	32
5.18 配置 LUN 拷贝速率	
5.19 配置 LUN 的应用类型	33
6 FAQ	35
6.1 通过 Cinder Driver 映射卷是否需要事先在华为存储上创建主机、主机组、LUN 组等	映射需要的关联对象?
6.2 映射卷时 Nova 日志中打印 lsblk 命令执行失败导致映射失败。	35
6.3 启动器配置策略样例	35

 $m{1}$   $_{ ext{ fin}}$ 

华为Cinder Driver是部署在OpenStack Cinder模块上的一个插件,用于对接华为存储向OpenStack平台提供卷存储能力。华为Cinder Driver支持iSCSI和FC两种协议。

# 2 版本配套关系

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

表 2-1 Cinder Driver 存储产品版本支持说明

OpenStack版本	存储产品版本
Newton	T系列 V100R005/V200R002
Ocata	OceanStor V3 V300R001/V300R002/V300R003/V300R006
Pike	OceanStor 2200/2600 V300R005
Queens	OceanStor V5 V500R007
Rocky	OceanStor 18500/18800 V100R001/V300R003/V300R006/
Stein	V500R007
	OceanStor Dorado V3 V300R001/V300R002
	OceanStor Dorado V6 V600R003C00
	OceanStor F V3/V5全闪存存储

#### 山 说明

- 1. 从Rocky版本开始不再支持华为T系列存储
- 2. 华为维护最近6个OpenStack稳定发布版本的配套Driver, 再之前的版本在华为仓库上仍然可以获取, 但不再维护, 请谨慎选择

表 2-2 Cinder Driver 特性支持说明(↓:支持, x:不支持)

特性	Newt on	Ocata	Pike	Quee ns	Rock y	Stein
Create Volume	√	√	√	√	√	<b>√</b>
Delete Volume	√	√	<b>√</b>	√	<b>√</b>	√
Attach Volume	√	√	<b>√</b>	√	<b>√</b>	√
Detach Volume	√	√	√	√	<b>√</b>	<b>√</b>

特性	Newt on	Ocata	Pike	Quee ns	Rock y	Stein
Extend Volume	√	√	√	√	√	√
Create Snapshot	√	√	√	√	√	√
Delete Snapshot	√	√	√	√	√	√
Create Volume from Snapshot	√	√	√	√	√	√
Create Volume from Image	√	√	√	√	√	√
Create Volume from Volume	√	√	√	√	√	√
Create Image from Volume	√	√	√	√	√	√
Volume Migration	√	√	√	√	√	√
QoS	<b>√</b>	√	√	√	√	√
FC zoning	<b>√</b>	<b>√</b>	√	√	<b>√</b>	√
SmartTier	<b>√</b>	<b>√</b>	√	<b>√</b>	<b>√</b>	<b>√</b>
SmartCache	<b>√</b>	<b>√</b>	√	√	<b>√</b>	√
Thin Volume	√	√	√	√	√	√
Thick Volume	<b>√</b>	<b>√</b>	√	√	<b>√</b>	<b>√</b>
SmartPartition	<b>√</b>	<b>√</b>	√	√	<b>√</b>	√
HyperMetro	√	√	√	√	√	√
Retype	√	√	√	√	√	√
Manage/Unmanage Volume	<b>√</b>	<b>√</b>	√	√	<b>√</b>	√
Manage/Unmanage Snapshot	<b>√</b>	<b>√</b>	√	<b>√</b>	<b>√</b>	√
Replication V2.1	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	√
Create a volume on a certain storage pool	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	√	<b>√</b>
Create a volume with certain disk type	√	√	<b>√</b>	√	√	√
Backup Snapshot	√	√	√	√	√	√
Multipath	√	√	√	√	√	√
Multi-Attach	<b>√</b>	√	<b>√</b>	<b>√</b>	<b>√</b>	√
Consistency Group	<b>√</b>	√	√	<b>√</b>	<b>√</b>	<b>√</b>
Snapshot Consistency Group	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>
HyperMetro Consistency Group	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>	<b>√</b>

特性	Newt on	Ocata	Pike	Quee ns	Rock y	Stein
Replication Consistency Group	√	√	√	<b>√</b>	√	<b>√</b>

#### 须知

- 1. Dorado存储不支持Thick Volume、SmartTier、SmartCache和SmartPartition特性。
- 2. 低端存储不支持某些增值特性,请参考具体存储型号的支持特性规格。

# **3** Cinder Driver 安装与部署

- 3.1 华为Cinder Driver获取
- 3.2 非容器化Cinder Volume服务华为Driver部署
- 3.3 容器化Cinder Volume服务华为Driver部署

## 3.1 华为 Cinder Driver 获取

您可以通过两种途径获取到华为Cinder Driver:

- 1. 通过OpenStack社区仓库。从Kilo版本开始,华为Driver已集成到OpenStack社区仓库,安装OpenStack即会自带华为Driver,位于../cinder/cinder/volume/drivers/huawei目录下。
- 2. 通过华为仓库。

步骤1 打开浏览器,访问仓库地址: https://github.com/Huawei/OpenStack Driver。

步骤2 点击"Download ZIP"按钮,华为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Cinder目录,其下即包含多个OpenStack版本的华为Driver。

#### ----结束

#### □ 说明

- 1. OpenStack社区版本发布后,不允许合入新的特性,bug修复周期长且存在风险。华为自有 OpenStack Driver仓库版本,能够保证新特性及时发布,bug及时修复,相较社区版本更稳定 完善。
- 2. OpenStack社区仅维护两个稳定版本,华为自有OpenStack Driver仓库维护六个稳定版本,能够保障历史版本长期稳定运行。
- 3. 强烈推荐使用华为仓库版本替换社区版本。

## 3.2 非容器化 Cinder Volume 服务华为 Driver 部署

步骤1 查找系统中原有的华为Driver代码,相对路径为.../cinder/volume/drivers/huawei。

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Driver代码目录:

```
# find / -name huawei_driver.py
/usr/lib/python2.7/dist-packages/cinder/volume/drivers/huawei/huawei_driver.py

如上所示, driver的路径为:
/usr/lib/python2.7/dist-packages/cinder/volume/drivers/huawei/
```

步骤2 删除步骤1返回目录下所有的华为Driver代码文件。

步骤3 将对应版本的华为Driver代码文件全部拷贝到步骤1中返回目录中。

**步骤4** 按照**章节4**进行配置。

步骤5 重启Cinder Volume服务。

----结束

## 3.3 容器化 Cinder Volume 服务华为 Driver 部署

步骤1 使用docker save命令备份Cinder Volume容器镜像到镜像文件(可选)。

#### □说明

具体镜像版本请通过docker image Is查询。

步骤2 在任意目录下创建临时目录build并进入该目录。

# mkdir build;cd build

步骤3 将获取到的华为Driver代码文件拷贝当前目录下。

步骤4 在当前目录下新建名称为Dockerfile的文件,并编辑如下内容。

```
FROM ***: ***
COPY *.py /huawei/cinder/driver/path/
```

- \*\*\*:\*\*\*替换为原Cinder Volume容器镜像的名称和版本号。
- /huawei/cinder/driver/path/替换为华为Driver在容器中的路径。

#### **步骤**5 执行镜像构建命令。

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

#### □ 说明

\*\*\*:\*\*\*替换为原Cinder Volume容器镜像的名称和版本号。

**步骤6** 按照章节4进行配置。

步骤7 重启Cinder Volume容器。

----结束

# 4 Cinder Driver 基本属性配置

分别介绍在iSCSI协议和FC协议下如何配置华为Cinder Driver。

#### 须知

- 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"块存储服务",且存储池名不能包含中文。
- FC组网时,要使用存储资源,需要安装sg工具。
- 华为Driver配置文件所有参数值中不能含有XML特殊字符<>&'"。
- 若使用从镜像创建卷功能,需安装多路径软件,且在Cinder Volume节点/etc/cinder/cinder.conf文件里面的指定后端里增加如下配置信息:

use\_multipath\_for\_image\_xfer = True
enforce\_multipath\_for\_image\_xfer = True

- 4.1 iSCSI组网Cinder Driver配置
- 4.2 配置FC组网Cinder Driver
- 4.3 配置多后端
- 4.4 配置租户用户

## 4.1 iSCSI 组网 Cinder Driver 配置

介绍在iSCSI协议下如何配置不同华为存储产品的Cinder Driver。

#### 须知

若使用从镜像创建卷功能,需安装多路径软件,且在Cinder Volume节点/etc/cinder/cinder.conf文件里面的指定后端里增加如下配置信息:

use\_multipath\_for\_image\_xfer = True
enforce\_multipath\_for\_image\_xfer = True

### 4.1.1 配置 TV1 存储 iSCSI 协议 Cinder Driver

#### 须知

仅Rocky之前版本支持TV1存储。

**步骤1** 在 "/etc/cinder"路径下创建华为Driver配置文件,文件格式为XML。配置文件名称可自定义,例如 "cinder\_huawei\_conf.xml"。

步骤2 配置华为Driver配置文件必填参数。

```
<?xml version='1.0' encoding='UTF-8'?>
<config>
   <Storage>
       \label{eq:conduct} $$ \Product T (Product) $$
       \begin{tabular}{ll} $$ \Protocol>iSCSI</Protocol> \end{tabular}
       <ControllerIPO>*.*.*.*</ControllerIPO>
       <ControllerIP1>*. *. *. *</controllerIP1>
       <UserName>***</UserName>
       <UserPassword>***</UserPassword>
   </Storage>
   <LUN>
       <StoragePool Name="***"/>
       <StoragePool Name="***"/>
   \langle /LUN \rangle
   <iSCSI>
       <DefaultTargetIP>*.*.*.*/DefaultTargetIP>
   </iSCSI>
</config>
```

#### 表 4-1 必填参数说明

属性名称	说明
Product	存储产品类型,针对TV1存储固定填写"T"。
Protocol	连接协议类型,iSCSI场景固定填写"iSCSI"。
ControllerIP0	TV1存储设备主控制器IP地址。
ControllerIP1	TV1存储设备从控制器IP地址。
UserName	存储管理员用户名。
UserPassword	存储管理员密码。
StoragePool	需要使用的存储池名称,需提前在目标存储上创建。
DefaultTargetIP	存储提供给计算节点的默认iSCSI目标器端口IP。

#### 步骤3 配置华为Driver配置文件选填参数(可选)。

#### 表 4-2 选填参数说明

属性名称	默认值	说明
LUNType	Thick	默认创建LUN的类型,可选值为"Thick"或者"Thin"。
StripUnitSize	64	创建LUN的分条深度大小,可选值为: 4、8、 16、32、64、128、256、512,单位为KB。创建 精简LUN时,此项无效。
WriteType	1	Cache写方式,可选值为"1"(回写)、"2" (透写)。
Prefetch Type	3	Cache预取策略,可选值为 "0" (不预取)、 "1" (固定预取)、"2" (可变预取)或者 "3" (智能预取)。
Prefetch Value	0	Cache预取值。
HostIP	-	Nova compute节点的主机IP。
OSType	Linux	Nova compute节点的主机操作系统类型。

**步骤4** 修改华为Driver配置文件的拥有者及用户组与"/etc/cinder/cinder.conf"文件的拥有者及用户组一致。

```
-rw-r--r-- 1 cinder cinder 2662 Jul 29 02:13 cinder.conf
-rw-r--r-- 1 cinder cinder 778 Jul 30 02:56 cinder_huawei_conf.xml
```

步骤5 配置 "/etc/cinder/cinder.conf" 文件,在文件最后添加如下后端配置。

```
[TV1_iSCSI]

volume_driver = cinder.volume.drivers.huawei.huawei_t.HuaweiTISCSIDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf.xml

volume backend name = TV1 iSCSI
```

#### 山 说明

- 后端名TV1 iSCSI可自定义。
- volume\_driver表示待使用的Driver类型,此处填写 cinder.volume.drivers.huawei.huawei\_t.HuaweiTISCSIDriver。
- cinder\_huawei\_conf\_file表示配置的华为Driver配置文件路径。

在[DEFAULT]区块中修改enabled\_backends配置项,配置使用TV1\_iSCSI后端。

```
[DEFAULT]
...
enabled_backends=TV1_iSCSI
```

步骤6 重新启动Cinder Volume服务。

----结束

## 4.1.2 配置其他存储 iSCSI 协议 Cinder Driver

**步骤1** 在 "/etc/cinder" 路径下创建华为Driver配置文件,文件格式为XML。配置文件名称可自定义,例如 "cinder\_huawei\_conf.xml"。

步骤2 配置华为Driver配置文件必填参数。

```
<?xml version='1.0' encoding='UTF-8'?>
<config>
   <Storage>
      <Product>***</Product>
      <Protocol>iSCSI</Protocol>
      <RestURL>https://*.*.*.*:8088/deviceManager/rest/</RestURL>
      <UserName>***</UserName>
      <UserPassword>***</UserPassword>
  </Storage>
  <\!\!\text{LUN}\!\!>
      <StoragePool>***</StoragePool>
  \langle /LUN \rangle
  <iSCSI>
      <DefaultTargetIP>*.*.*
  </iSCSI>
</config>
```

#### 表 4-3 必填参数说明

属性名称	说明
Product	存储产品类型,可选值为TV2、18000、V3、V5或Dorado。
Protocol	连接协议类型,iSCSI场景固定填写"iSCSI"。
RestURL	Rest接口访问地址。如果需要配置多个RestURL,以分号";"进行分割。
UserName	存储管理员用户名。
UserPassword	存储管理员密码。
StoragePool	需要使用的存储池名称,需提前在目标存储上创建。如果需要配置 多个资源池,以分号";"进行分割。
DefaultTargetIP	存储系统提供给计算节点的默认iSCSI目标器端口IP。

#### 须知

Product配置项务必根据实际对接的后端存储型号填写。

#### 步骤3 配置华为Driver配置文件选填参数(可选):

#### 表 4-4 选填参数说明

属性名称	默认值	说明
LUNType	Thin (Dorado) Thick (其 他)	创建LUN的类型,可选值为"Thick"或者"Thin"。Dorado仅支持"Thin"。
WriteType	1	Cache写方式,可选值为"1"(回写)、"2" (透写)。Dorado普通卷只支持回写,不支持 透写。
Prefetch Type	3	Cache预取策略,可选值为"0"(不预取)、 "1"(固定预取)、"2"(可变预取)或者 "3"(智能预取)。
Prefetch Value	0	Cache预取值。
LUNcopyWaitInter val	5	插件在启动LUN拷贝后需要不断查询LUN拷贝的拷贝进度,该值用于用户指定间隔多长时间查询一次。
Timeout	2592000	等待存储设备执行LUN拷贝的超时时间,单位 为秒(s)。
LUNCopySpeed	2	LUN拷贝速率,可选值为"1"(低速)、 "2"(中速)、"3"(高速)、"4"(最 快)。
LUNCloneMode	luncopy	LUN克隆方式,可选值为 "fastclone" 或者 "luncopy"。仅Dorado支持fastclone,Dorado V300R001C20及其之后版本支持luncopy, 其他 存储仅支持luncopy。
HyperSyncSpeed	2	双活pair同步速率,可选值为"1"(低速)、 "2"(中速)、"3"(高速)、"4"(最 快)。
ReplicaSyncSpeed	2	远程复制pair同步速率,可选值为"1"(低速)、"2"(中速)、"3"(高速)、"4"(最快)。

## **步骤4** 修改华为Driver配置文件的拥有者及用户组与"/etc/cinder/cinder.conf"文件的拥有者及用户组一致。

```
-rw-r--r-- 1 cinder cinder 2662 Jul 29 02:13 cinder.conf
-rw-r--r-- 1 cinder cinder 778 Jul 30 02:56 cinder_huawei_conf.xml
```

#### 步骤5 配置 "/etc/cinder/cinder.conf" 文件,在文件最后添加如下后端配置。

```
[huawei_backend] volume_driver = cinder.volume.drivers.huawei.huawei_driver.HuaweiISCSIDriver cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf.xml volume_backend_name = huawei_backend
```

- 后端名huawei backend可自定义。
- volume\_driver表示待使用的Driver类型,此处填写 cinder.volume.drivers.huawei.huawei driver.HuaweiISCSIDriver。
- cinder huawei conf file表示配置的华为Driver配置文件路径。

在[DEFAULT]区块中修改enabled\_backends配置项,配置使用huawei\_backend后端。

```
[DEFAULT]
...
enabled_backends=huawei_backend
```

步骤6 重新启动Cinder Volume服务。

----结束

## 4.1.3 配置 iSCSI 多路径

如果需要为iSCSI配置多路径,可参考以下配置进行。

**步骤1** 通过DeviceManager界面在存储设备上创建端口组,将需要使用多路径的业务链路加入该端口组(可选)。

步骤2 在华为Driver配置文件中增加端口组设置,配置启动器端所需要使用的端口组名称。

方式一: 采用启动器名字的精确配置

方式二: 采用主机名字的正则表达式配置

```
<iscsi>
<initiator HostName="******" TargetIP="x.x.x.x y.y.y.y" TargetPortGroup="***"/>
<Initiator HostName="******" TargetIP="x.x.x.x y.y.y.y" TargetPortGroup="***"/>
</iscsi>
```

步骤3 打开OpenStack Nova模块多路径开关。

- 如果OpenStack为Mitaka及之前版本,则在Nova Compute节点"/etc/nova/nova.conf"的[libvirt]中增加"iscsi use multipath = True"。
- 如果OpenStack为Newton及以后版本,则在Nova Compute节点 "/etc/nova/nova.conf"的[libvirt]中增加 "volume\_use\_multipath = True"。

步骤4 重启nova-compute服务。

----结束

- 1. 参数说明
  - Rocky及其以后版本不支持配置TargetPortGroup参数;
  - TargetIP多个IP之间用""隔开;
  - 当同时配置TargetPortGroup和TargetIP时, TargetPortGroup优先级更高,这两个参数只需要配置一个即可;
  - 如果端口组中的端口已经加入到绑定端口中,禁止使用TargetPortGroup进行配置。
- 2. 规则说明

当且仅当配置HostName= "\*",表示的是通用配置,该条配置对任意主机名的主机生效,其余Hostname的配置均视为一般配置。

配置策略的规则:

- 以精准配置为准 (详见6.3例1)
  - 对于同一个主机,当同时通过启动器名和主机名方式都能够找到相应配置时,以启动器配置为准;
  - 对于同一个主机,当同时满足主机名的通用配置和一般配置时,以一般配置为准;
  - 优先级:启动器名配置方式>主机名一般配置>主机名通用配置
- 以满足配置的第一条为准(针对于主机名一般配置) (详见6.3例2)

## 4.1.4 配置 CHAP 和 ALUA 特性

在公共网络中,与存储系统iSCSI主机端口在同一网络的应用服务器都可以访问存储系统,并对其进行读写操作。因此,数据的安全性存在隐患。为了保证存储系统的安全,可以通过配置CHAP安全性认证,限制应用服务器对存储系统的访问权限。

#### **步骤1** 在华为Driver配置文件中添加如下配置:

#### 表 4-5 参数说明

参数	默认值	说明
Name	-	ISCSI启动器名称(可选), 和参数 "HostName" 必填写一个
HostName	-	主机名称的正则表达式(可选),和参数 "Name"必填写一个
ALUA	0	0: 默认(华为自研多路径) 1: 第三方多路径
FAILOVERMODE		1.
PAILOVERMODE	-	数才需要下发。
SPECIALMODETYPE	-	启动器的特殊模式类型,当启动器的切换模式为 特殊模式时该参数才需要下发。
РАТНТҮРЕ	-	启动器的路径类型。

参数	默认值	说明
CHAPinfo	-	CHAP认证时用户自行配置的用户名和密码,格式为"mm-user;mm-user@storage1",用户名和密码以分号隔开;
		CHAP名称长度范围是4到223位;
		密码长度范围是12到16位;
		密码必须包含大写字母、小写字母、数字和特殊符号(`~!@#\$%^*()=+\ [{}];:,./?和空格)中任意三者的组合;
		密码不能和CHAP名称或者CHAP名称的倒写一样。

由于FAILOVERMODE、SPECIALMODETYPE和PATHTYPE参数的取值范围在不同存储型号上存在差异,请参考相应存储型号的API文档进行配置。

当且仅当配置HostName= "\*",表示的是通用配置,该条配置对任意主机名的主机生效,其余Hostname的配置均视为一般配置。

配置策略的规则:

- 以精准配置为准 (详见6.3例1)
  - 对于同一个主机,当同时通过启动器名和主机名方式都能够找到相应配置时,以启动器配置为准:
  - 对于同一个主机,当同时满足主机名的通用配置和一般配置时,以一般配置为准;
  - 优先级:启动器名配置方式>主机名一般配置>主机名通用配置
- 以满足配置的第一条为准(针对于主机名一般配置) (详见6.3例2)

#### ----结束

## 4.2 配置 FC 组网 Cinder Driver

介绍在FC协议下如何配置不同华为存储产品的Cinder Driver。

## 4.2.1 配置 TV1 存储 FC 协议 Cinder Driver

#### 须知

仅Rocky之前版本支持TV1存储。

**步骤1** 在 "/etc/cinder" 路径下创建华为Driver配置文件,文件格式为XML。配置文件名称可自定义,例如 "cinder\_huawei\_conf.xml"。

步骤2 配置华为Driver配置文件必填参数。

#### 表 4-6 必填参数说明

属性名称	说明	
Product	存储产品类型,针对TV1存储固定填写"T"。	
Protocol	连接协议类型,FC场景固定填写"FC"。	
ControllerIP0	TV1存储设备主控制器IP地址。	
ControllerIP1	TV1存储设备从控制器IP地址。	
UserName	存储管理员用户名。	
UserPassword	存储管理员密码。	
StoragePool	需要使用的存储池名称,需提前在目标存储上创建。	

#### 步骤3 配置华为Driver配置文件选填参数(可选)。

#### 表 4-7 选填参数说明

属性名称	默认值	说明	
LUNType	Thick	默认创建LUN的类型,可选值为"Thick"或者"Thin"。	
StripUnitSize	64	创建LUN的分条深度大小,可选值为: 4、8、 16、32、64、128、256、512,单位为KB。创建 精简LUN时,此项无效。	
WriteType	1	Cache写方式,可选值为"1"(回写)、"2" (透写)。	
Prefetch Type	3	Cache预取策略,可选值为"0"(不预取)、 "1"(固定预取)、"2"(可变预取)或者 "3"(智能预取)。	
Prefetch Value	0	Cache预取值。	
HostIP	-	Nova compute节点的主机IP。	

属性名称	默认值	说明
OSType	Linux	Nova compute节点的主机操作系统类型。

**步骤4** 修改华为Driver配置文件的拥有者及用户组与"/etc/cinder/cinder.conf"文件的拥有者及用户组一致。

```
-rw-r--r- 1 cinder cinder 2662 Jul 29 02:13 cinder.conf
-rw-r--r- 1 cinder cinder 778 Jul 30 02:56 cinder_huawei_conf.xml
```

步骤5 配置 "/etc/cinder/cinder.conf" 文件,在文件最后添加如下后端配置。

```
[TV1_FC]
volume_driver = cinder.volume.drivers.huawei.huawei_t.HuaweiTFCDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf.xml
volume_backend_name = TV1_FC
```

#### □ 说明

- 后端名TV1\_FC可自定义。
- volume\_driver表示待使用的Driver类型,此处填写 cinder.volume.drivers.huawei.huawei t.HuaweiTFCDriver。
- cinder huawei conf file表示配置的华为Driver配置文件路径。

在[DEFAULT]区块中修改enabled backends配置项,配置使用TV1 FC后端。

```
[DEFAULT]
...
enabled_backends=TV1_FC
```

步骤6 重新启动Cinder Volume服务。

----结束

## 4.2.2 配置其他存储 FC 协议 Cinder Driver

**步骤1** 在 "/etc/cinder" 路径下创建华为Driver配置文件,文件格式为XML。配置文件名称可自定义,例如 "cinder\_huawei\_conf.xml"。

#### 步骤2 配置华为Driver配置文件必填参数。

#### 表 4-8 必填参数说明

属性名称	说明	
Product	存储产品类型,可选值为TV2、18000、V3、V5或Dorado。	
Protocol	连接协议类型,FC场景固定填写"FC"。	

属性名称	说明
RestURL	Rest接口访问地址。如果需要配置多个RestURL,以分号";"进行分割。
UserName	存储管理员用户名。
UserPassword	存储管理员密码。
StoragePool	需要使用的存储池名称,需提前在目标存储上创建。如果需要配置 多个资源池,以分号";"进行分割。

#### 须知

Product配置项务必根据实际对接的后端存储型号填写。

#### 步骤3 配置华为Driver配置文件选填参数(可选):

#### 表 4-9 选填参数说明

属性名称	默认值	说明	
LUNType	Thin (Dorado)	创建LUN的类型,可选值为"Thick"或者 "Thin"。Dorado仅支持"Thin"。	
	Thick(其 他)		
WriteType	1	Cache写方式,可选值为"1"(回写)、"2" (透写)。	
Prefetch Type	3	Cache预取策略,可选值为 "0" (不预取)、 "1" (固定预取)、"2" (可变预取)或者 "3" (智能预取)。	
Prefetch Value	0	Cache预取值。	
LUNcopyWaitInter val	5	插件在启动LUN拷贝后需要不断查询LUN拷贝的拷贝进度,该值用于用户指定间隔多长时间查询一次。	
Timeout	2592000	等待存储设备执行LUN拷贝的超时时间,单位 为秒(s)。	

属性名称	默认值	说明	
LUNCopySpeed	2	LUN拷贝速率,可选值为"1"(低速)、 "2"(中速)、"3"(高速)、"4"(最 快)。	
LUNCloneMode	luncopy	LUN克隆方式,可选值为"fastclone"或者 "luncopy"。仅Dorado支持fastclone,Dorado V300R001C20及其之后版本支持luncopy, 其他 存储仅支持luncopy。	
HyperSyncSpeed	2	双活pair同步速率,可选值为"1"(低速)、 "2"(中速)、"3"(高速)、"4"(最 快)。	
ReplicaSyncSpeed	2	远程复制pair同步速率,可选值为"1"(低速)、"2"(中速)、"3"(高速)、"4" (最快)。	

**步骤4** 修改华为Driver配置文件的拥有者及用户组与"/etc/cinder/cinder.conf"文件的拥有者及用户组一致。

```
-rw-r--r-- 1 cinder cinder 2662 Jul 29 02:13 cinder.conf
-rw-r--r-- 1 cinder cinder 778 Jul 30 02:56 cinder_huawei_conf.xml
```

步骤5 配置 "/etc/cinder/cinder.conf" 文件,在文件最后添加如下后端配置。

```
[huawei_backend]
volume_driver = cinder.volume.drivers.huawei.huawei_driver.HuaweiFCDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf.xml
volume_backend_name = huawei_backend
```

#### □ 说明

- 后端名huawei backend可自定义。
- volume\_driver表示待使用的Driver类型,此处填写 cinder.volume.drivers.huawei.huawei\_driver.HuaweiFCDriver。
- cinder huawei conf file表示配置的华为Driver配置文件路径。

在[DEFAULT]区块中修改enabled\_backends配置项,配置使用huawei\_backend后端。

```
[DEFAULT]
...
enabled_backends=huawei_backend
```

步骤6 重新启动Cinder Volume服务。

----结束

## 4.2.3 配置智能划域

智能划域功能为可选项,当需要使用智能划域功能时,进行如下配置。

**步骤1** 配置cinder.conf 文件。参考https://docs.openstack.org/cinder/latest/configuration/block-storage/fc-zoning.html配置交换机参数。

步骤2 重新启动Cinder Volume服务。

----结束

#### 须知

使用租户用户不支持自动划域特性。

## 4.2.4 配置 FC 多路径

如果需要为FC配置多路径,可参考以下配置进行。

步骤1 打开OpenStack Nova模块多路径开关。

- 如果OpenStack为Mitaka及之前版本,则在Nova Compute节点 "/etc/nova/nova.conf" 的[libvirt]中增加 "iscsi use multipath = True"。
- 如果OpenStack为Newton及以后版本,则在Nova Compute节点"/etc/nova/nova.conf"的[libvirt]中增加"volume use multipath = True"。

步骤2 非自动划域场景下配置FC最小链路数量

在华为Driver配置文件中添加如下配置:

#### 表 4-10 参数说明

参数	默认值	说明	
MinOnlineFCInitiator	0	可用FC启动器最小数量,类型为阿拉伯数字。默 认值0表示需要所有启动器在线	

步骤3 重启nova-compute服务。

----结束

## 4.2.5 配置 ALUA 特性

如果需要为FC配置ALUA特性,可参考以下配置进行。

步骤1 在华为Driver配置文件中添加如下配置:

#### 表 4-11 参数说明

参数	默认值	说明	
Name	-	FC启动器WWN(可选), 和参数 "HostName"必填写一个	
HostName	-	主机名称的正则表达式(可选),和参数 "Name"必填写一个	
ALUA	0	0: 默认(华为自研多路径)	
		1: 第三方多路径	
FAILOVERMODE	-	启动器的切换模式,当开启第三方多路径时该参 数才需要下发。	
SPECIALMODETYPE	-	启动器的特殊模式类型,当启动器的切换模式为 特殊模式时该参数才需要下发。	
РАТНТҮРЕ	-	启动器的路径类型。	

#### □ 说明

由于FAILOVERMODE、SPECIALMODETYPE和PATHTYPE参数的取值范围在不同存储型号上存在差异,请参考相应存储型号的API文档进行配置。

当且仅当配置HostName= "\*",表示的是通用配置,该条配置对任意主机名的主机生效,其余Hostname的配置均视为一般配置。

配置策略的规则:

- 以精准配置为准 (详见6.3 例1)
  - 对于同一个主机,当同时通过启动器名和主机名方式都能够找到相应配置时,以启动器 配置为准·
  - 对于同一个主机, 当同时满足主机名的通用配置和一般配置时, 以一般配置为准;
  - 优先级:启动器名配置方式>主机名一般配置>主机名通用配置
- 以满足配置的第一条为准(针对于主机名一般配置) (详见6.3例2)

#### ----结束

## 4.3 配置多后端

在"/etc/cinder/cinder.conf"中配置多个存储系统,举例如下:

```
[DEFAULT]
...
enabled_backends = v3_fc, v5_fc
[v3_fc]
volume_driver = cinder.volume.drivers.huawei.huawei_driver.HuaweiFCDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf_v3_fc.xml
volume_backend_name = v3_fc
[v5_fc]
volume_driver = cinder.volume.drivers.huawei.huawei_driver.HuaweiFCDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf_v5_fc.xml
volume_backend_name = v5_fc
```

## 4.4 配置租户用户

在华为Driver配置文件中配置租户用户对接后端存储,配置方式如下:

#### 表 4-12 租户参数说明

参数名称	说明
UserName	存储租户的用户名
UserPassword	存储租户的用户密码
vStoreName	vstore名

#### 须知

- 其他相关参数请参考前面章节。
- 仅对接具有多租户功能的存储版本时支持配置租户用户。

# 5 Cinder Driver 高级特性配置

华为存储支持一系列高级特性,可以通过配置特定的卷类型,创建出具有高级特性属性的卷。

#### 须知

华为存储上需要提前导入相应高级特性的license。

- 5.1 配置Thick属性
- 5.2 配置Thin属性
- 5.3 配置SmartQoS属性
- 5.4 配置SmartPartition属性
- 5.5 配置SmartCache属性
- 5.6 配置SmartTier 属性
- 5.7 配置双活属性
- 5.8 配置远程复制属性
- 5.9 指定存储池创建LUN
- 5.10 指定磁盘类型创建LUN
- 5.11 指定后端创建LUN
- 5.12 配置Backup Snapshot
- 5.13 配置证书认证
- 5.14 配置LUN归属控制器
- 5.15 配置SmartDedupe属性
- 5.16 配置SmartCompression属性
- 5.17 使用fastclone方式创建卷
- 5.18 配置LUN拷贝速率

#### 5.19 配置LUN的应用类型

## 5.1 配置 Thick 属性

介绍配置Thick属性的操作步骤。

#### 须知

Dorado存储不支持Thick卷。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 设置Thick卷类型,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thick\_provisioning\_support='<is> true'

步骤3 设置创建Thick卷时调度器的调度机制, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set provisioning:type='thick'

步骤4 使用步骤1的卷类型创建卷。

----结束

## 5.2 配置 Thin 属性

介绍配置Thin属性的操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 设置Thin卷类型, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'

步骤3 使用步骤1的卷类型创建卷。

----结束

## 5.3 配置 SmartQoS 属性

OpenStack的QoS特性主要依赖于前端Hypervisor和后端存储来实现。

前端QoS配置可参考社区文档https://docs.openstack.org/cinder/latest/admin/blockstorage-basic-volume-gos.html。

本章节主要介绍配置华为存储后端QoS的操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 开启QoS属性, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:QoS\_support='<is> true'

步骤3 创建QoS规格, <name>表示自定义QoS规格名称。

# cinder qos-create <name> IOType=\* <qos\_key>=\*\*\*

#### 表 5-1 IOType 参数说明

参数名称	说明	备注
ЮТуре	控制读写类型。	有效值如下:
		● 0: 读IO
		● 1: 写IO
		● 2: 读写IO

#### 表 5-2 qos\_key 可选参数说明

可选参数名称	说明	备注
maxIOPS	最大IOPS限制策略。	有效值为>0的整数。
minIOPS	最小IOPS保护策略。	有效值为>0的整数。
maxBandWidth	最大带宽限制策略。	单位MB/s,有效值为>0的整数。
minBandWidth	最小带宽保护策略。	单位MB/s,有效值为>0的整数。
latency	最大时延保护策略。	单位ms,有效值为>0的整数。

#### 山 说明

- 1. 同属保护策略或者同属限制策略的多个参数可以同时指定,但是保护策略和限制策略参数不能同时指定。
- 2. Dorado存储仅支持IOType设置为"2", 仅支持限制策略参数。

**步骤4** 关联QoS规格和卷类型, <qos\_specs>表示步骤3创建的QoS规格ID, <volume\_type\_id>表示步骤1创建的卷类型ID。

# cinder qos-associate <qos\_specs> <volume\_type\_id>

步骤5 使用步骤1的卷类型创建卷。

----结束

## 5.4 配置 SmartPartition 属性

介绍配置SmartPartition属性的操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 开启SmartPartition属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:smartpartition='<is> true'

步骤3 配置后端存储上已经存在的SmartPartition的名称。

# cinder type-key <vtype> set smartpartition:partitionname='\*\*\*'

步骤4 使用步骤1的卷类型创建卷。

----结束

## 5.5 配置 SmartCache 属性

介绍配置SmartCache属性的操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 开启SmartCache属性, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:smartcache='<is> true'

步骤3 配置后端存储上已经存在的SmartCache名称。

# cinder type-key <vtype> set smartcache:cachename='\*\*\*

步骤4 使用步骤1的卷类型创建卷。

----结束

## 5.6 配置 SmartTier 属性

介绍配置SmartTier属性的操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 开启SmartTier属性, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:smarttier='<is> true'

步骤3 配置SmartTier迁移策略。

# cinder type-key <vtype> set smarttier:policy=\*\*\*

#### □说明

policy表示迁移策略。可选参数值为:

- 0: 不迁移
- 1: 自动迁移
- 2: 向高性能迁移
- 3: 向低性能迁移

步骤4 使用步骤1的卷类型创建卷。

----结束

## 5.7 配置双活属性

介绍配置双活的操作步骤。

步骤1 在 "/etc/cinder/cinder.conf" 的指定后端中配置远端设备信息。

hypermetro\_device =
 storage\_pool:\*\*\*,
 san\_address:https://\*.\*.\*.\*:8088/deviceManager/rest/,
 san\_user:\*\*\*

```
san_password:***,
vstore_name:***,
iscsi_default_target_ip:*.*.*.*,
metro_domain:***,
metro_sync_completed:***
```

#### 表 5-3 参数说明

参数名称	说明
storage_pool	双活远端存储池名称
san_address	Rest接口访问地址。
san_user	双活远端存储管理员用户名或租户用户名
san_password	双活远端存储管理员密码或租户用户密码
vstore_name	双活远端存储vstore名(仅在使用租户用户场景下需要配置)
metro_domain	双活域名称
iscsi_default_target_ip	双活远端业务端口IP。
metro_sync_completed	双活创建时是否需要等待同步完成,可选值为True和 False; 默认值为True

如果双活远端需要使用iscsi多路径,在远端设备配置信息里面增加如下配置:

#### □说明

iscsi info各参数说明和配置规则可参考4.1.3和4.1.4章节。

步骤2 重启Cinder Volume服务。

步骤3 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤4 开启双活属性, <vtype>表示步骤3的卷类型名称。

# cinder type-key <vtype> set capabilities:hypermetro='<is> true'

步骤5 使用步骤3的卷类型创建卷。

----结束

## 5.8 配置远程复制属性

介绍配置远程复制的操作步骤。

步骤1 在 "/etc/cinder/cinder.conf"的指定后端中配置复制远端设备信息。

```
replication_device =
  backend_id:***,
  storage_pool:***,
  san_address:https://*.*.*.*:8088/deviceManager/rest/,
  san_user:***,
  san_password:******,
  vstore_name:***,
  iscsi_default_target_ip:*.*.*.*
```

#### 表 5-4 参数说明

参数名称	说明
backend_id	目标设备id,failover时需要指定的远端设备。
storage_pool	复制failover目标后端存储池名称。
san_address	Rest接口访问地址。
san_user	复制远端存储管理员用户名或租户用户名。
san_password	复制远端存储管理员密码或租户用户密码。
vstore_name	复制远端存储vstore名(仅在使用租户用户场景下需要配置)。
iscsi_default_target_ip	复制远端业务端口IP。

如果复制远端需要使用iscsi多路径,在远端设备配置信息里面增加如下配置:

#### 山 说明

iscsi info各参数说明和配置规则可参考4.1.3和4.1.4章节。

步骤2 重启Cinder Volume服务。

步骤3 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤4 启用远程复制属性, <vtype>表示步骤3的卷类型名称。

# cinder type-key <vtype> set capabilities:replication\_enabled='<is> true'

步骤5 (可选)设置复制类型。

# cinder type-key <vtype> set replication\_type='<in> \*\*\*'

#### □ 说明

\*\*\*可选值如下:

sync: 同步复制async: 异步复制

如果未显示配置replication type, 默认为异步复制。

步骤6 使用步骤3的卷类型创建卷。

----结束

## 5.9 指定存储池创建 LUN

介绍指定存储池创建卷操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 配置单个存储池, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set pool\_name=StoragePool001

或配置多个存储池。

# cinder type-key <vtype> set pool\_name="<or> StoragePool001 <or> StoragePool002"

**步骤3** 使用步骤1的卷类型创建卷。

----结束

## 5.10 指定磁盘类型创建 LUN

介绍指定存储池磁盘类型创建卷操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 配置一种磁盘类型,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set disk\_type=sas

或配置多种磁盘类型。

# cinder type-key <vtype> set disk\_type="<or> sas <or> ssd"

#### □说明

disk\_type可选值为 "ssd"、 "sas"、 "nl\_sas" 和 "mix", 其中 "mix"表示是由 "ssd"、 "sas"、 "nl sas" 中的2种及以上混合。

步骤3 使用步骤1的卷类型创建卷。

----结束

## 5.11 指定后端创建 LUN

介绍指定后端创建卷操作步骤。

步骤1 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 配置后端名称,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set volume\_backend\_name=\*\*\*

#### 山 说明

volume\_backend\_name参数需要设置为 "/etc/cinder/cinder.conf" 配置文件中相应后端的volume backend name参数值。

步骤3 使用步骤1的卷类型创建卷。

----结束

## 5.12 配置 Backup Snapshot

介绍配置快照备份功能。

**步骤1** 参考社区文档https://docs.openstack.org/ocata/config-reference/block-storage/backup-drivers.html配置备份driver,并确保Cinder Backup服务正常启动。

步骤2 在/etc/cinder/cinder.conf配置文件的对应后端增加如下配置。

backup\_use\_temp\_snapshot = True

步骤3 重启Cinder Volume和Cinder Backup服务。

----结束

## 5.13 配置证书认证

介绍配置通过证书认证方式连接后端存储。

**步骤1** 配置华为Driver配置文件。在<Storage>配置段中增加SSLCertVerify和SSLCertPath配置:

#### □ 说明

- SSLCertVerify表示是否开启证书认证,有效值为True或False。未显示配置默认为False。
- SSLCertPath指定用于认证的证书路径,仅在SSLCertVerify为True的情况下有效。

步骤2 重启Cinder Volume服务。

----结束

## 5.14 配置 LUN 归属控制器

介绍配置LUN归属控制器操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 开启huawei controller属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:huawei\_controller='<is> true'

步骤3 配置归属控制器名称。

# cinder type-key <vtype> set huawei\_controller:controllername=\*\*\*

归属控制器名称格式形式为"CTE0.A"。

步骤4 使用步骤1的卷类型创建卷。

----结束

## 5.15 配置 SmartDedupe 属性

介绍配置SmartDedupe属性操作步骤。仅Thin卷支持。

#### 开启 SmartDedupe 操作步骤

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 对于Newton及之后版本,执行如下命令开启SmartDedupe属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:dedup='<is> true'

对于Mitaka及之前版本,执行如下命令开启SmartDedupe属性,<vtype>表示步骤1的卷类型名称。

 $\label{thm:cond} \mbox{$\#$ cinder type-key $$\langle vtype$\rangle set capabilities:thin_provisioning_support='$\langle is$\rangle true' capabilities:dedup='$\langle in$\rangle true' }$ 

**步骤3** 使用步骤1的卷类型创建卷。

#### ----结束

#### □ 说明

Dorado存储仅支持Thin卷, 步骤2中capabilities:thin\_provisioning\_support配置可选。

## 关闭 SmartDedupe 操作步骤

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 对于Newton及之后版本,执行如下命令关闭SmartDedupe属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:dedup='<is> false'

对于Mitaka及之前版本,执行如下命令关闭SmartDedupe属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:dedup='<in> false'

**步骤3** 使用步骤1的卷类型创建卷。

#### ----结束

#### □ 说明

Dorado存储仅支持Thin卷, 步骤2中capabilities:thin provisioning support配置可选。

## 5.16 配置 SmartCompression 属性

介绍配置SmartCompression属性操作步骤。仅Thin卷支持。

### 开启 SmartCompression 操作步骤

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 对于Newton及之后版本,执行如下命令开启SmartCompression属性,<vtype>表示步骤1 的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:compression='<is> true'

对于Mitaka及之前版本,执行如下命令开启SmartCompression属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:compression='<in> true'

步骤3 使用步骤1的卷类型创建卷。

#### ----结束

#### □ 说明

Dorado存储仅支持Thin卷, 步骤2中capabilities:thin provisioning support配置可选。

## 关闭 SmartCompression 操作步骤

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

**步骤2** 对于Newton及之后版本,执行如下命令关闭SmartCompression属性,<vtype>表示步骤1 的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:compression='<is> false'

对于Mitaka及之前版本,执行如下命令关闭SmartCompression属性,<vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:thin\_provisioning\_support='<is> true'
capabilities:compression='<in> false'

步骤3 使用步骤1的卷类型创建卷。

#### ----结束

#### □ 说明

Dorado存储仅支持Thin卷,步骤2中capabilities:thin\_provisioning\_support配置可选。

## 5.17 使用 fastclone 方式创建卷

本节介绍如何配置fastclone方式实现从快照创建卷和克隆卷。

步骤1 从快照创建卷和克隆卷时,在"--metadata"参数指定"fastclone"选项,如下所示:

# cinder create --snapshot-id \*\*\* --metadata fastclone=True
# cinder create --source-volid \*\*\* --metadata fastclone=True

#### ----结束

#### □ 说明

- 1. 如果fastclone未显示设置,默认使用LUN拷贝方式。
- 2. 仅Dorado存储支持fastclone特性。

## 5.18 配置 LUN 拷贝速率

从快照创建卷和克隆卷默认使用LUN拷贝方式,本节介绍如何配置LUN拷贝速度。

步骤1 从快照创建卷和克隆卷时,为"--metadata"参数指定"copyspeed"选项,如下所示:

```
# cinder create --snapshot-id *** --metadata copyspeed=2
# cinder create --source-volid *** --metadata copyspeed=2
```

#### ----结束

#### 表 5-5 copyspeed 参数说明

参数值	参数说明
1	低速
2	中速
3	高速
4	最快

#### □ 说明

如果copyspeed未显示设置, 默认使用中速。

## 5.19 配置 LUN 的应用类型

介绍配置LUN的应用类型操作步骤。

**步骤1** 创建卷类型, <name>表示自定义卷类型名称。

# cinder type-create <name>

步骤2 开启huawei\_application\_type属性, <vtype>表示步骤1的卷类型名称。

# cinder type-key <vtype> set capabilities:huawei\_application\_type='<is> true'

步骤3 配置LUN的应用类型名称。

# cinder type-key <vtype> set huawei\_application\_type:applicationname='\*\*\*

步骤4 使用步骤1的卷类型创建卷。

----结束

- 该配置仅适用于Dorado V300R001C20及其以上版本的存储;
- 该配置中也包含了compression和dedupe的配置,优先级高于SmartDedupe和SmartCompression。一旦同时配置,以应用类型的配置为准。

 $6_{\text{FAQ}}$ 

- 6.1 通过Cinder Driver映射卷是否需要事先在华为存储上创建主机、主机组、LUN组等映射需要的关联对象?
- 6.2 映射卷时Nova日志中打印lsblk命令执行失败导致映射失败。
- 6.3 启动器配置策略样例

## 6.1 通过 Cinder Driver 映射卷是否需要事先在华为存储上创建主机、主机组、LUN 组等映射需要的关联对象?

不需要,因为Driver会自动创建这些卷映射需要的关联对象,以及映射视图。相反,如果手动在存储上创建了这些对象,反而会影响Driver的映射逻辑,请确保在使用Cinder映射前删除这些对象。

## 6.2 映射卷时 Nova 日志中打印 lsblk 命令执行失败导致映射失败。

Nova在卷映射流程中会调用lsblk命令查询映射卷的信息,出现此错误一般是未安装 lsblk工具,请确保Nova系统中安装了此工具并且有相应的执行权限。

## 6.3 启动器配置策略样例

例1. 配置文件如下:

```
<iSCSI>
<Initiator Name="iqn" ALUA="xxx" FAILOVERMODE="XXX" PATHTYPE="xxx" CHAPinfo="xxx"
TargetPortGroup="***"/>
<Initiator HostName="*" ALUA="xxx" FAILOVERMODE="XXX" PATHTYPE="xxx" CHAPinfo="xxx"
TargetPortGroup="***"/>
<Initiator HostName="host" ALUA="xxx" FAILOVERMODE="XXX" PATHTYPE="xxx" CHAPinfo="xxx"
TargetPortGroup="***"/>
</iSCSI>
```

对于主机名为"host",启动器名为"iqn"的主机而言,上述三条配置都能对启动器进行ALUA、端口组等信息进行配置。根据规则(以精准配置为准),我们肯定会选择配置启动器名的配置参数(第一条),其次(当第一条不存在时)是配置主机名一般

配置的参数(第三条),最后(第一条和第三条不存在时)才是我们的主机名通用配置(第二条)。

#### 例2.配置文件如下:

```
<iSCSI>
<Initiator HostName="Host[0-9]" ALUA="xxx" FAILOVERMODE="XXX" PATHTYPE="xxx" CHAPinfo="xxx"
TargetPortGroup="***"/>
<Initiator HostName="Host[5-7]" ALUA="xxx" FAILOVERMODE="XXX" PATHTYPE="xxx" CHAPinfo="xxx"
TargetPortGroup="***"/>
</iSCSI>
```

对于主机名为"Host6"的主机,上述两条配置均能满足配置规则,根据规则(以满足配置的第一条为准(针对于主机名非通用配置方式)),会选择第一条配置参数进行启动器的设置。

#### 山 说明

配置规则可参考4.1.3和4.1.4章节。