

本节我们讨论 volume 的 Backup 操作。

Backup 是将 volume 备份到别的地方（备份设备），将来可以通过 restore 操作恢复。

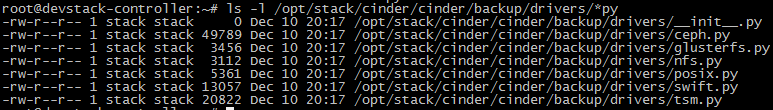
**Backup VS Snapshot**

初看 backup 功能好像与 snapshot 很相似，都可以保存 volume 的当前状态，以备以后恢复。但二者在用途和实现上还是有区别的，具体表现在：

1. Snapshot 依赖于源 volume，不能独立存在；而 backup 不依赖源 volume，即便源 volume 不存在了，也可以 restore。
2. Snapshot 与源 volume 通常存放在一起，都由同一个 volume provider 管理；而 backup 存放在独立的备份设备中，有自己的备份方案和实现，与 volume provider 没有关系。
3. 上面两点决定了 backup 具有容灾功能；而 snapshot 则提供 volume provider 内便捷的回溯功能。

**配置 cinder-backup**

Cinder 的 backup 功能是由 cinder-backup 服务提供的，devstack 默认没有启用该服务，需要手工启用。与 cinder-volume 类似，cinder-backup 也通过 driver 架构支持多种备份 backend，包括 POSIX 文件系统、NFS、Ceph、GlusterFS、Swift 和 IBM TSM。支持的driver 源文件放在 /opt/stack/cinder/cinder/backup/drivers/



本节我们将以 NFS 为 backend 来研究 backup 操作。

在实验环境中，存放 volume backup 的 NFS 远程目录为 192.168.104.11:/backup cinder-backup 服务节点上 mount point 为 /backup\_mount。

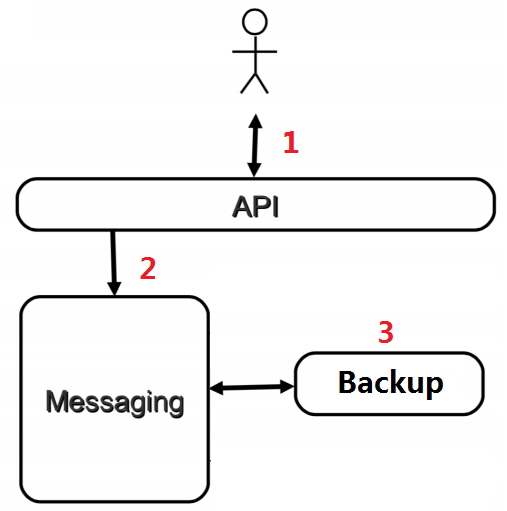
需要在 /etc/cinder/cinder.conf 中作相应配置。

http://7xo6kd.com1.z0.glb.clouddn.com/upload-ueditor-image-20160712-1468277068173082134.jpg

然后手工启动 cinder-backup 服务。

/usr/bin/python /usr/local/bin/cinder-backup --config-file /etc/cinder/cinder.conf

一切准备就绪，下面我们来看 backup 操作的流程



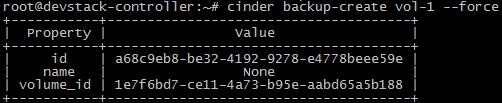
1. 向 cinder-api 发送 backup 请求
2. cinder-api 发送消息
3. cinder-backup 执行 backup 操作

下面我们详细讨论每一个步骤。

**向 cinder-api 发送 backup 请求**

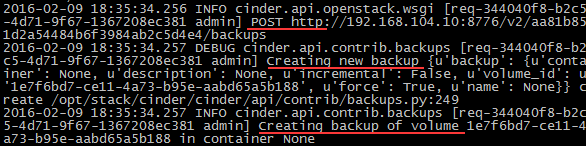
客户（可以是 OpenStack 最终用户，也可以是其他程序）向 cinder-api 发送请求：“请 backup 指定的 volume。

这里我们将 backup volume “vol-1”，目前 backup 只能在 CLI 中执行。



这里因为 vol-1 已经 attach 到 instance，需要使用 --force 选项。

cinder-api 接收到 backup volume 的请求。日志文件在 /opt/stack/logs/c-api.log。



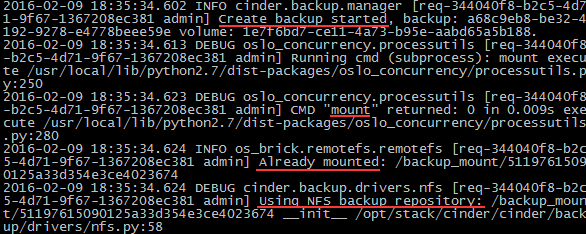
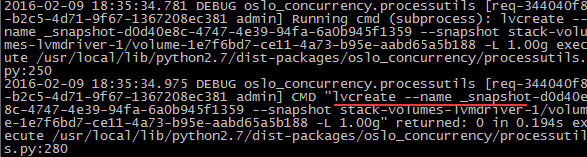
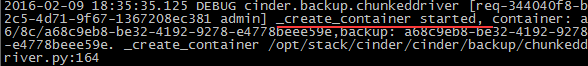
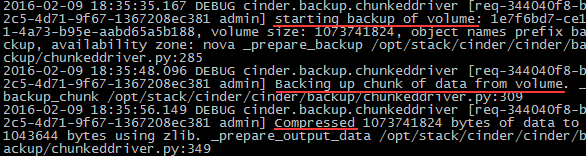
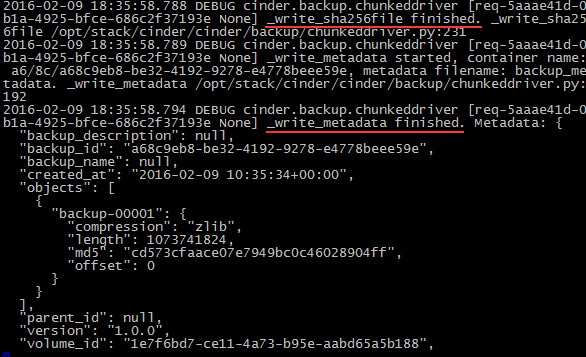
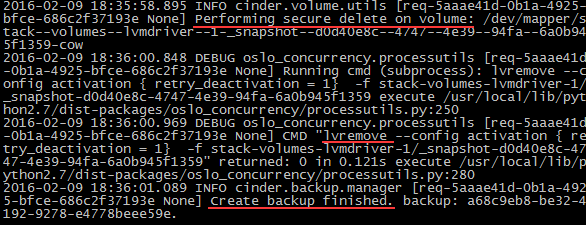
**cinder-api 发送消息**

cinder-api 发送 backup 消息。cinder-api 没有打印发送消息的日志，只能通过源代码查看 /opt/stack/cinder/cinder/backup/api.py，方法为 create。

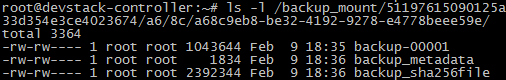
http://7xo6kd.com1.z0.glb.clouddn.com/upload-ueditor-image-20160712-1468277068553001942.jpg

**cinder-backup 执行 backup 操作**

cinder-backup 收到消息后，通过如下步骤完成 backup 操作，日志为 /opt/stack/logs/c-vol.log。

1. 启动 backup 操作，mount NFS。  
   
2. 创建 volume 的临时快照。  
   
3. 创建存放 backup 的 container 目录。  
   
4. 对临时快照数据进行压缩，并保存到 container 目录。  
   
5. 创建并保存 sha256（加密）文件和 metadata 文件。  
   
6. 删除临时快照。  
   

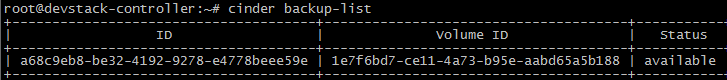
Backup 完成后，我们可以查看一下 container 目录的内容



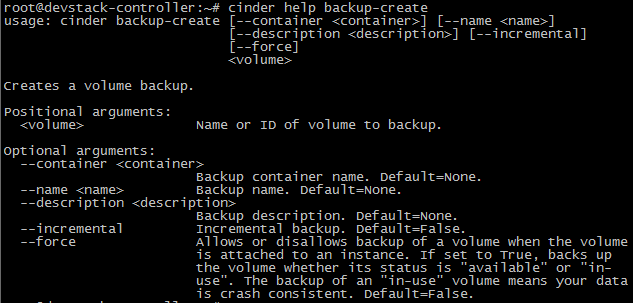
里面有三个文件，根据前面的日志我们可以知道：

1. backup-00001，压缩后的 backup 文件。
2. backup\_metadata，metadata 文件。
3. backup\_sha256file，加密文件。

可以通过 cinder backup-list 查看当前存在的 backup。



另外我们可以查看一下 cinder backup-create 的用法。



这里有 --incremental 选项，表示可以执行增量备份。

如果之前做过普通（全量）备份，之后可以通过增量备份大大减少需要备份的数据量，是个很不错的功能。增量备份的操作和日志分析留给大家做练习。  
  
以上就是 volume backup 的分析，下一节我们讨论如何通过 restore 操作恢复备份的 volume。