



Red Hat OpenStack Platform 16.2

块存储备份指南

了解、使用和管理 Red Hat OpenStack Platform 中的块存储备份服务

Red Hat OpenStack Platform 16.2 块存储备份指南

了解、使用和管理 Red Hat OpenStack Platform 中的块存储备份服务

OpenStack Team
rhos-docs@redhat.com

法律通告

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

摘要

本文档论述了如何部署 Red Hat OpenStack Platform Block Storage (cinder) 备份服务。Red Hat OpenStack Platform director 可将 Red Hat Ceph Storage、NFS 和 Object Storage (swift) 配置为后端。您还可以将 Google Cloud Storage 配置为备份后端。

目录

使开源包含更多	3
对红帽文档提供反馈	4
第 1 章 RED HAT OPENSTACK PLATFORM BLOCK STORAGE 备份服务概述	5
1.1. 备份和快照	5
1.2. 卷备份工作流	5
1.3. 卷恢复工作流	7
1.4. 云存储与本地存储	8
第 2 章 块存储备份服务部署	9
2.1. 为备份服务配置后端存储选项	9
2.2. 使用 GOOGLE CLOUD 配置部署 OVERCLOUD	10
第 3 章 使用块存储备份服务	11
3.1. 完整备份	11
3.2. 增量备份	14
3.3. 取消备份	15
3.4. 查看和修改租户备份配额	15
3.5. 从备份中恢复	16
第 4 章 对块存储备份服务进行故障排除	19
4.1. 验证服务	19
4.2. 查询失败备份的状态	19
4.3. 使用 PACEMAKER 管理资源	20
附录 A. GOOGLE CLOUD STORAGE 配置	21
A.1. 创建 GCS 凭证文件	21
A.2. 创建 CINDER-BACKUP-GCS.YAML	22
A.3. 使用 GOOGLE CLOUD 设置创建环境文件	23
A.4. 部署 OVERCLOUD	25
附录 B. 高级块存储配置	27
B.1. 高级配置选项	27

使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。详情请查看 [CTO Chris Wright 的信息](#)。

对红帽文档提供反馈

我们感谢您对文档提供反馈信息。与我们分享您的成功秘诀。

在 JIRA 中提供文档反馈

使用 [Create Issue](#) 表单对文档提供反馈。JIRA 问题将在 Red Hat OpenStack Platform Jira 项目中创建，您可以在其中跟踪您的反馈进度。

1. 确保您已登录到 JIRA。如果您没有 JIRA 帐户，请创建一个帐户来提交反馈。
2. 点击以下链接打开 **Create Issue** 页面：[Create Issue](#)
3. 完成 **Summary** 和 **Description** 字段。在 **Description** 字段中，包含文档 URL、章节或章节号以及问题的详细描述。不要修改表单中的任何其他字段。
4. 点 **Create**。

第 1 章 RED HAT OPENSTACK PLATFORM BLOCK STORAGE 备份服务概述

Red Hat OpenStack Platform (RHOSP) 提供了在 Red Hat Enterprise Linux 之上构建私有或公共基础架构即服务(DSL)云的基础。它是一个可扩展、容错的平台，用于开发启用云的工作负载。

您可以使用 RHOSP 仪表板或命令行客户端方法来管理备份服务的大部分功能，但您必须使用命令行来执行一些更高级的流程。



注意

有关 Red Hat OpenStack Platform 文档的完整套件，请参阅 [Red Hat OpenStack Platform 文档](#)。

块存储服务 (cinder) 包含了一个横向扩展备份服务，可用于将 Cinder 卷备份到不同的存储后端。您可以使用 Block Storage 备份服务创建和恢复完整备份或增量备份。该服务独立于卷。

Red Hat OpenStack Platform (RHOSP) director 是一个安装和管理完整的 RHOSP 环境的工具组，称为 overcloud。有关 director 的更多信息，请参阅 [Director 安装和使用指南](#)。overcloud 包含向最终用户提供服务的组件，包括块存储。Block Storage 备份服务是您在 Controller 节点上部署的可选服务。

1.1. 备份和快照

卷备份是卷内容的持久副本。卷备份通常作为对象存储创建，默认情况下通过 OpenStack Object Storage 服务(swift)管理。您可以使用 Red Hat Ceph 和 NFS 作为备份的替代后端。

当您创建卷备份时，所有备份元数据都存储在块存储服务数据库中。cinder-backup 服务在从备份中恢复卷时使用此元数据。这意味着，当您从灾难数据库丢失中执行恢复时，您必须首先恢复块存储服务数据库，然后才能从备份中恢复任何卷，只要块存储服务数据库的原始卷备份元数据保持不变。如果您只想将卷备份的子集配置为崩溃数据库丢失，您也可以导出备份元数据。然后，您可以使用 REST API 或 cinder 客户端将元数据重新导入到块存储数据库，然后正常恢复卷备份。

卷备份与快照不同。备份保留卷中包含的数据，快照会在特定时间保留卷的状态。如果卷已有快照，则无法删除它。卷备份可防止数据丢失，而快照有助于克隆。因此，快照后端通常与卷后端在一起，以便最小化克隆期间的延迟。相反，备份存储库通常独立于后端，可以是在不同节点或不同的物理存储中。这样可以防止备份存储库免受卷后端可能出现的任何损坏。

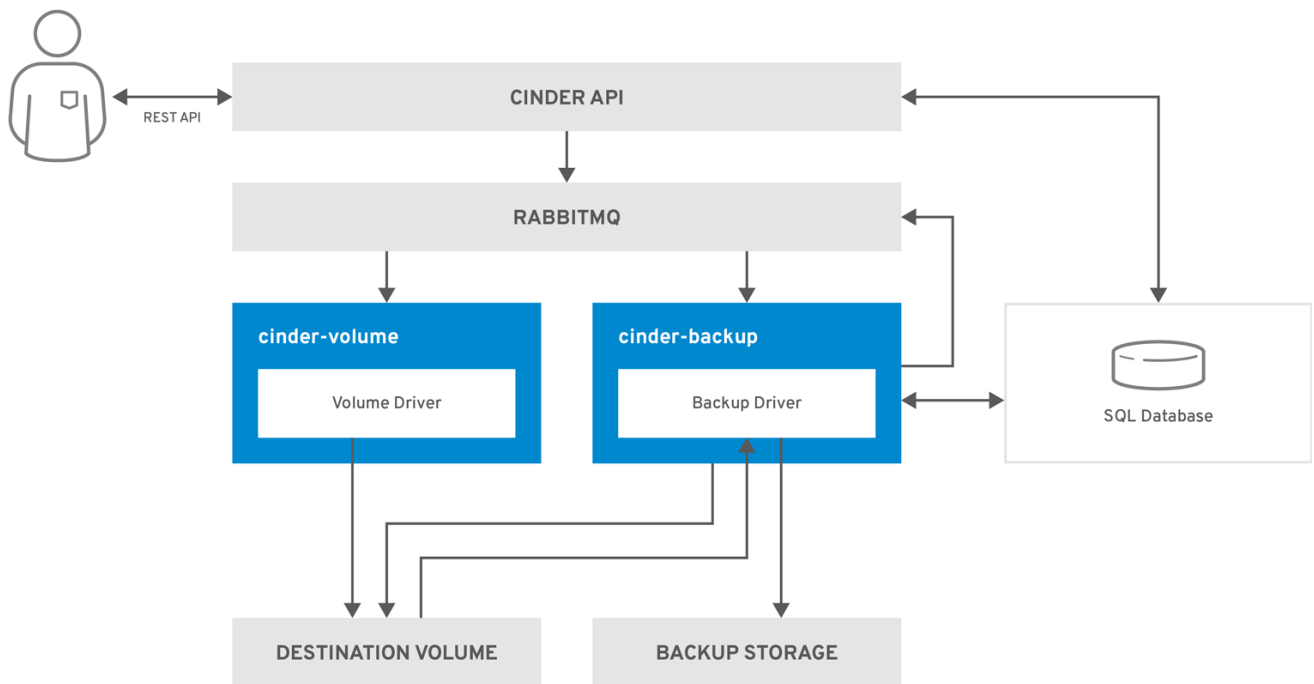
有关卷快照的更多信息，请参阅 [存储指南中的创建、使用或删除卷快照](#)。

1.2. 卷备份 workflow

当块存储备份服务执行备份时，它会从 cinder API 接收请求来备份目标卷。备份服务完成请求，并将内容存储在后端上。

下图显示了请求如何与 Block Storage (cinder) 服务交互来执行备份。

图 1.1. 使用块存储服务创建备份



OPENSTACK_483337_1218

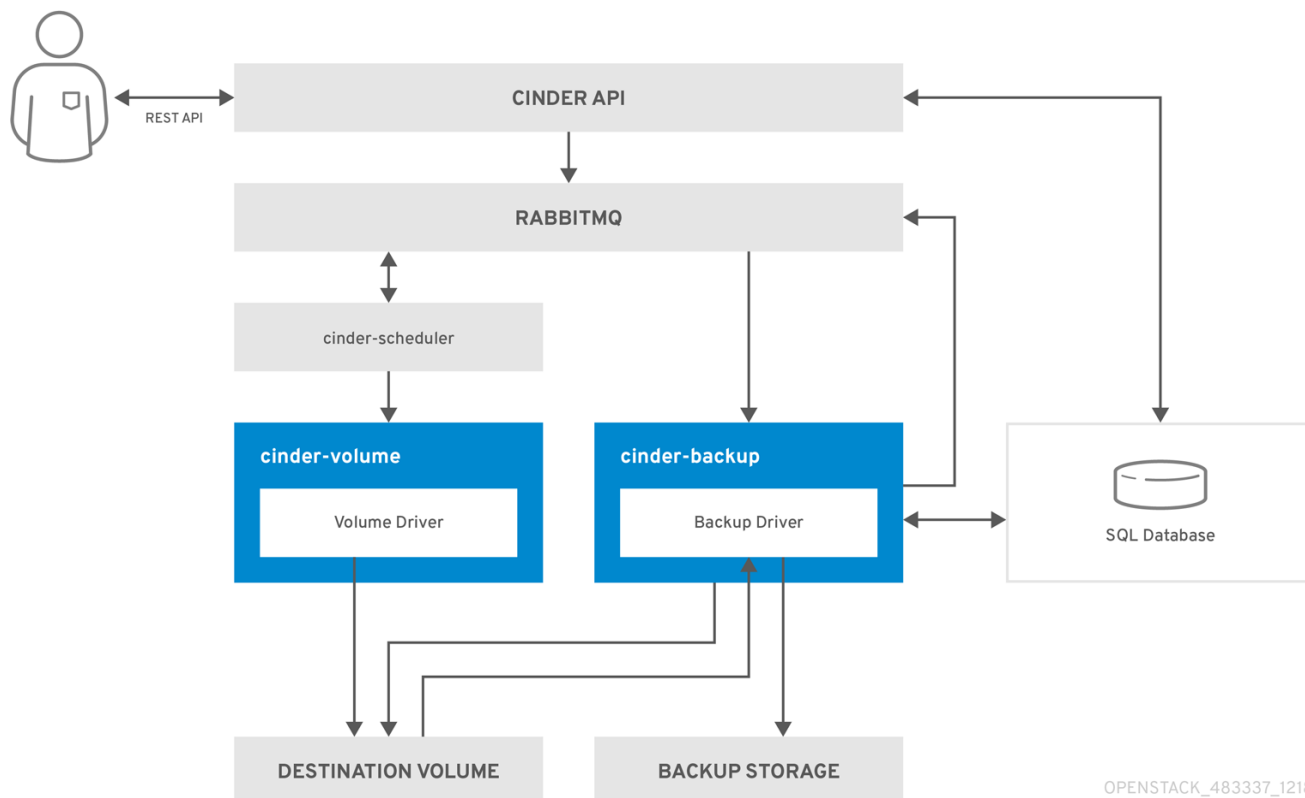
1. 客户端通过调用 cinder API 来发出备份块存储卷的请求。
2. cinder API 服务接收来自 HAProxy 的请求，并验证请求、用户凭据和其他信息。
3. 在 SQL 数据库中创建备份记录。
4. 通过 AMQP 向 **cinder-backup** 服务发出异步 RPC 调用来备份卷。
5. 将带有 ID 的当前备份记录返回到 API 调用者。
6. RPC 创建消息到达其中一个备份服务。
7. **cinder-backup** 服务对 **get_backup_device** 执行同步 RPC 调用。
8. **cinder-volume** 服务确保正确的设备返回给调用者。通常，它是相同的卷，但如果正在使用卷，服务会返回一个临时克隆的卷或临时快照，具体取决于配置。
9. **cinder-backup** 服务为 **cinder-volume** 发出另一个同步 RPC 来公开源设备。
10. **cinder-volume** 服务导出并映射源设备（卷或快照），并返回适当的连接信息。
11. **cinder-backup** 服务使用连接信息附加源卷。
12. **cinder-backup** 服务调用备份驱动程序，设备已附加，它会开始传输到备份目的地。
13. 卷与备份主机分离。
14. **cinder-backup** 服务向 **cinder-volume** 发出同步 RPC 来断开源设备的连接。
15. **cinder-volume** 服务取消映射并删除该设备的导出。
16. 如果创建了临时卷或临时快照，**cinder-backup** 会调用 **cinder-volume** 将其删除。

17. **cinder-volume** 服务移除临时卷。
18. 备份完成后，会在数据库中更新备份记录。

1.3. 卷恢复 workflow

下图演示了在用户请求恢复块存储服务(cinder)备份时发生的步骤。

图1.2. 恢复块存储备份



1. 客户端通过调用 Cinder REST API 来发出恢复块存储备份的请求。
2. cinder API 从 HAProxy 接收请求，并验证请求、用户凭据和其他信息。
3. 如果请求不包含现有卷作为目的地，API 会发出异步 RPC 调用来创建新卷并轮询卷的状态，直到卷可用为止。
4. **cinder-scheduler** 选择一个卷服务，并发出 RPC 调用来创建卷。
5. 所选 **cinder-volume** 服务创建卷。
6. 当 **cinder-api** 检测到卷可用时，会在数据库中创建备份记录。
7. 通过 AMQP 向备份服务发出异步 RPC 调用，以恢复备份。
8. 将当前卷 ID、备份 ID 和卷名称返回到 API 调用者。
9. RPC 创建消息到达其中一个备份服务。
10. **cinder-backup** 服务对 **cinder-volume** 执行同步 RPC 调用，以公开目标卷。
11. **cinder-volume** 服务导出并映射目标卷返回适当的连接信息。

12. **cinder-backup** 服务使用连接信息附加源卷。
13. **cinder-backup** 服务使用已经附加的设备调用驱动程序，它会开始恢复卷目的地。
14. 卷与备份主机分离。
15. **cinder-backup** 服务向 **cinder-volume** 发出同步 RPC 来断开源设备的连接。
16. **cinder-volume** 服务取消映射并删除该设备的导出。
17. 备份完成后，会在数据库中更新备份记录。

1.4. 云存储与本地存储

Google Cloud Storage 驱动程序是唯一由块存储备份服务支持的云驱动程序。默认情况下，Google Cloud Storage 驱动程序将最昂贵的存储解决方案 Nearline 用于这种类型的备份。

配置备份服务以优化性能。例如，如果您从 Europe 创建备份，请将备份区域更改为 Europe。如果您没有从默认更改备份区域，则由于两个区域之间的地理位置，性能可能会较慢。



注意

Google Cloud Storage 需要特殊的配置，如 [附录 A, Google Cloud Storage 配置](#) 部分中解释。

下表根据情况列出了云存储和本地存储的优点和限制。

情况	云存储	本地存储
非站点备份	云存储位于另一个公司的数据中心，因此会自动关闭。您可以从多个位置访问数据。远程副本可用于灾难恢复。	需要额外的规划和费用。
硬件控制	取决于其他服务的可用性和经验。	您已完全控制存储硬件。需要管理和经验。
成本考虑	不同的定价政策或层取决于您从供应商使用的服务。	根据需要添加额外的硬件是一个已知的成本。
网络速度和数据访问	整体数据访问速度较慢，需要访问互联网。速度和延迟依赖于多个因素。	快速、即时访问数据。不需要互联网访问。

第 2 章 块存储备份服务部署

Block Storage 备份服务是可选的。默认情况下不安装它，因此您必须将其添加到 overcloud 部署中

先决条件

- 现有 Red Hat OpenStack Platform (RHOSP) 安装。
- 带有兼容备份驱动程序的可用存储源：Object Storage (swift; default)、Ceph、NFS 或 Google Cloud Storage。



注意

Google Cloud Storage 需要额外的配置。更多信息请参阅 [附录 A, Google Cloud Storage 配置](#)。

2.1. 为备份服务配置后端存储选项

您可以将备份服务配置为使用与默认后端不同的后端。此流程描述了如何复制 `cinder-backup.yaml` 文件以及如何修改参数默认值。

流程

1. 创建 `cinder-backup.yaml` 文件的副本，它位于 `/usr/share/openstack-tripleo-heat-templates/environments/` 目录中，并将其存储在与其他自定义模板相同的位置。

```
cp /usr/share/openstack-tripleo-heat-templates/environments/cinder-backup.yaml
/home/stack/templates/cinder-backup-settings.yaml
```

2. `cinder-backup.yaml` 文件的副本包含默认设置，该设置使用 Pacemaker 为块存储备份服务配置 OpenStack Object Storage (swift) 后端。如果这是您用于备份的后端，则不需要更改此文件。如果您使用其他后端，请根据备份后端配置 `parameter_defaults`：

- 如果使用 Red Hat Ceph Storage，请使用以下方法配置 `parameter_defaults`：
 - `CinderBackupBackend`: (Required) `ceph`
 - `CinderBackupRbdPoolName`: (可选) 设置为自定义 RBD 池名称。默认：`backup`
- 如果使用 NFS，请使用以下方法配置 `parameter_defaults`：
 - `CinderBackupBackend`: (Required) `nfs`
 - `CinderBackupNfsShare` : (Required) 设置为您要挂载的 NFS 共享。默认值为空。
 - `CinderBackupNfsMountOptions` : (可选) 设置为您需要的挂载选项。

3. 保存对该文件的更改。

4. 要启用备份服务并应用此配置，请使用其他环境文件将备份设置环境文件添加到堆栈中，并部署 overcloud：

```
(undercloud) [stack@undercloud ~]$ openstack overcloud deploy --templates \
-e [your environment files]
-e /home/stack/templates/cinder-backup-settings.yaml
```

如需更多信息和其他配置选项，请参阅 [附录 A, Google Cloud Storage 配置](#)。

2.2. 使用 GOOGLE CLOUD 配置部署 OVERCLOUD

在 `/home/stack/templates/` 中创建环境文件后，部署 overcloud，然后重启 cinder-backup 服务。



重要

使用 Google Cloud Storage 的 Cinder 备份已被弃用，并将在下一个主发行版本中删除支持。

流程

1. 以 **stack** 用户身份登录。
2. 部署配置：

```
$ openstack overcloud deploy --templates \  
-e /home/stack/templates/cinder-backup-settings.yaml
```



重要

如果您在创建 overcloud 时传递任何额外的环境文件，请使用 **-e** 选项再次传递它们，以避免对 overcloud 进行不必要的更改。

3. 部署完成后重启 **cinder-backup** 服务。

如需更多信息，请参阅 Director 安装和使用指南中的 [在一个 overcloud 部署中包括的环境文件](#)，以及高级 Overcloud 自定义指南中的 [环境文件](#)。

第 3 章 使用块存储备份服务

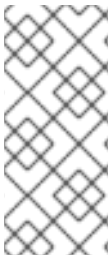
您可以使用 Block Storage 备份服务执行完整或增量备份，并将备份恢复到卷。

3.1. 完整备份

cinder backup-create 命令 默认创建卷的完整备份。您可以创建您可以访问的卷备份。这意味着，无论所有者用户如何，具有管理特权的用户可以备份任何卷。

3.1.1. 创建完整卷备份

要备份卷，请使用 **cinder backup-create** 命令。默认情况下，这个命令会创建卷的完整备份。如果卷有现有的备份，您可以选择创建增量备份。更多信息请参阅 [第 3.2.3 节“执行增量备份”](#)。



注意

在 Red Hat OpenStack Platform (RHOSP) 版本 16 之前，**cinder backup-create** 命令在第一个完整 Ceph 卷备份到 Ceph Storage 后端后创建增量备份。在 RHOSP 版本 16 及更高版本中，您必须使用 **--incremental** 选项来创建增量卷备份。如果您没有将 **--incremental** 选项与 **cinder backup-create** 命令一起使用，则默认设置会创建完整备份。更多信息请参阅 [第 3.2.3 节“执行增量备份”](#)。

您可以创建您可以访问的卷备份。这意味着，无论所有者用户如何，具有管理特权的用户可以备份任何卷。更多信息请参阅 [第 3.1.2 节“以管理员身份创建卷备份”](#)。

流程

1. 查看您要备份的卷的 ID 或显示名称：

```
# cinder list
```

2. 备份卷：

```
# cinder backup-create _VOLUME_
```

使用您要备份的卷的 **ID** 或 **显示** 名称替换 **VOLUME**。例如：

```
+-----+-----+
| Property |          Value          |
+-----+-----+
| id      | e9d15fc7-eeae-4ca4-aa72-d52536dc551d |
| name    |          None          |
| volume_id | 5f75430a-abff-4cc7-b74e-f808234fa6c5 |
+-----+-----+
```

生成的备份的 **volume_id** 与源卷的 ID 相同。

3. 验证卷备份创建已完成：

```
# cinder backup-list
```

4. 当备份条目的 **Status** 可用时，卷备份创建已完成。

3.1.2. 以管理员身份创建卷备份

具有管理特权的用户可以备份任何由 Red Hat OpenStack Platform 管理的卷。当 admin 用户备份一个非 admin 用户拥有的卷时，备份默认从卷所有者中隐藏。

流程

- 作为 admin 用户，您可以使用以下命令备份卷，并将备份提供给特定租户：

```
# cinder --os-auth-url <KEYSTONEURL> --os-tenant-name <TENANTNAME> --os-username <USERNAME> --os-password <PASSWD> backup-create <VOLUME>
```

根据您的环境要求替换以下变量：

- <TENANTNAME> 是您要提供备份的租户的名称。
- <USERNAME> 和 <ASP> 是 <TENANTNAME> 中用户的用户名和密码凭证。
- <VOLUME> 是您要备份的卷的名称或 ID。
- <KEYSTONEURL> 是 Identity 服务的 URL 端点，通常是 http://IP:5000/v2，其中 IP 是 Identity 服务的主机的 IP 地址。当您执行此操作时，生成的备份的大小会根据 **TENANTNAME** 的配额而不是租户 admin 的配额计数。

3.1.3. 导出卷备份的元数据

您可以导出并存储卷备份的元数据，以便您可以恢复卷备份，即使块存储数据库出现灾难性丢失。

流程

- 运行以下命令：

```
# cinder backup-export _BACKUPID_
```

将 <BACKUPID> 替换为卷备份的 ID 或名称：

```
+-----+-----+
| Property |      Value      |
+-----+-----+
| backup_service | cinder.backup.drivers.swift |
| backup_url | eyJzdGF0dXMiOiAiYXZhaWxhYmxlliwglm9iam...|
|           | ...4NS02ZmY4MzBhZWYwNWUiLCAic2l6ZSI6IDF9 |
+-----+-----+
```

卷备份元数据由 **backup_service** 和 **backup_url** 值组成。

3.1.4. 备份正在使用的卷

当支持 Block Storage 后端快照时，您可以使用 **--force** 选项创建 in-use 卷的备份。



注意

要使用 **--force** 选项，必须支持 Block Storage 后端快照。您可以通过检查您使用的后端的文档来验证快照支持。

通过使用 **--force** 选项，在执行备份前确认您没有静默驱动器。使用此方法会创建一个崩溃一致性，但不创建应用一致性备份。这意味着备份不知道在执行备份时运行哪个应用程序。但是，数据保持不变。

流程

- 要创建正在使用的卷的备份，请运行：

```
# cinder backup-create _VOLUME_ --incremental --force
```

3.1.5. 备份快照

您可以使用与快照关联的卷 ID 从快照创建完整备份。

流程

1. 使用 **cinder** 快照列表找到要备份的快照的快照 ID。

```
# cinder snapshot-list --volume-id _VOLUME_ID_
```

2. 如果快照被命名，您可以使用以下示例来查找 ID：

```
# cinder snapshot-show _SNAPSHOT_NAME_
```

3. 创建快照备份：

```
# cinder backup-create _VOLUME_ --snapshot-id=_SNAPSHOT_ID_
```



注意

使用 **--snapshot-id** 选项时，基于快照的 NFS 卷的备份会失败。这是个已知问题。

3.1.6. 在边缘站点间备份和恢复

您可以在边缘站点和可用区中的分布式计算节点(DCN)架构中备份和恢复块存储服务(cinder)卷。**cinder-backup** 服务在中央可用区(AZ)中运行，备份存储在中央 AZ 中。块存储服务不会在 DCN 站点存储备份。

先决条件

- 中央站点使用位于 `/usr/share/openstack-tripleo-heat-templates/environments` 中的 **cinder-backup.yaml** 环境文件进行部署。如需更多信息，请参阅[块存储备份服务部署](#)。
- Block Storage 服务(cinder) CLI 可用。
- 所有站点都必须使用通用的 **openstack** cephx 客户端名称。有关更多信息，请参阅[为外部访问创建 Ceph 密钥](#)。

流程

1. 在第一个 DCN 站点中创建卷备份：

```
$ cinder --os-volume-api-version 3.51 backup-create --name <volume_backup> --availability-zone <az_central> <edge_volume>
```

- 将 **<volume_backup>** 替换为卷备份的名称。
- 将 **<az_central>** 替换为托管 **cinder-backup** 服务的中央可用区的名称。
- 将 **<edge_volume>** 替换为您要备份的卷的名称。



注意

如果遇到 Ceph 密钥环的问题，您可能需要重启 **cinder-backup** 容器，以便主机中的密钥环成功复制到容器。

2. 将备份恢复到第二个 DCN 站点中的新卷：

```
$ cinder --os-volume-api-version 3.51 create --availability-zone <az_2> --name <new_volume> --backup-id <volume_backup> <volume_size>
```

- 将 **<az_2>** 替换为您要恢复备份的可用区的名称。
- 将 **<new_volume>** 替换为新卷的名称。
- 将 **<volume_backup>** 替换为您在上一步中创建的卷备份的名称。
- 将 **<volume_size>** 替换为等于或大于原始卷大小的值。

3.2. 增量备份

如果卷具有现有的备份，您可以使用 Block Storage 备份服务改为创建增量备份。

3.2.1. 性能考虑

增量和数据压缩等一些备份功能可能会影响性能。增量备份对性能产生影响，因为对于全和每个增量备份，卷中的所有数据都必须被读取和校验和。

您可以将数据压缩用于非 Ceph 后端。启用数据压缩需要额外的 CPU 电源，但整体使用较少的网络带宽和存储空间。

多路径配置也会影响性能。如果您在不启用多路径的情况下附加多个卷，则可能无法连接或具有影响性能的完整网络功能。

您可以使用高级配置选项启用或禁用压缩，定义进程数量，并添加额外的 CPU 资源。更多信息请参阅第 B.1 节“高级配置选项”。

3.2.2. 从快照备份的影响

有些后端支持从快照创建备份。支持这个功能的驱动程序可以直接附加快照，这比将快照克隆到卷中快，以便能够将其附加到卷中。通常，此功能可能会影响性能，因为从快照创建卷的额外步骤。

3.2.3. 执行增量备份

默认情况下，**cinder backup-create** 命令会创建卷的完整备份。但是，如果卷有现有的备份，您可以创建增量备份。

NFS、Object Storage (swift) 和 Red Hat Ceph Storage 备份软件仓库完全支持增量备份。

增量备份会捕获自上次全或增量备份以来卷的任何更改。执行大量、常规、全备份卷可能会变得大量资源，因为卷的大小会随时间增加。通过增量备份，您可以捕获对卷的定期更改，并最小化资源使用量。

流程

- 要创建增量卷备份，请使用以下命令使用 **--incremental** 选项：

```
# cinder backup-create _VOLUME_ --incremental
```

使用您要备份的卷的 **ID** 或 **显示名称** 替换 VOLUME。

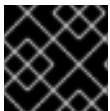


注意

如果已经有增量备份，则无法删除完整备份。如果完整备份有多个增量备份，则只能删除最新的备份。

3.3. 取消备份

要取消备份，管理员必须请求对备份进行强制删除。



重要

如果使用 Ceph 或 RBD 后端，则不支持此操作。

流程

- 运行以下命令：

```
# openstack volume backup delete --force <backup>
```

完成取消后，备份不再出现在备份列表中，备份可以成功取消。要验证备份是否已成功取消，源资源中的备份状态将停止。



注意

在 Red Hat OpenStack 版本 12 之前，备份状态存储在卷中，即使备份快照也是如此。因此，在备份快照时，如果快照仍然映射，对快照的任何删除操作都会出现错误。在 Red Hat OpenStack Platform 版本 13 及更高版本中，可以在任何受支持的备份驱动程序上取消持续恢复操作。

3.4. 查看和修改租户备份配额

通常，您可以使用控制面板来修改租户存储配额，例如，租户可以具有的卷、卷存储、快照或其他操作限制。但是，使用控制面板修改备份配额的功能还不可用。

您必须使用命令行界面修改备份配额。

流程

1. 要查看特定租户的存储配额(TENANT_ID)，请运行以下命令：

```
# cinder quota-show TENANT_ID
```

2. 要更新在特定租户中创建的备份的最大数量(MAXNUM)，请运行以下命令：

```
# cinder quota-update --backups MAXNUM TENANT_ID
```

3. 要更新特定租户中的所有备份(MAXGB)的最大总量，请运行以下命令：

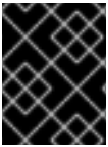
```
# cinder quota-update --backup-gigabytes MAXGB TENANT_ID
```

4. 要查看特定租户的存储配额使用情况，请运行以下命令：

```
# cinder quota-usage TENANT_ID
```

3.5. 从备份中恢复

在数据库失败或其他类型的事件后，使用您创建的备份来恢复数据。



重要

如果将 **cinder-backup** 服务配置为使用 Ceph RBD 驱动程序，则只能将备份卷恢复到基于 RBD 的块存储(cinder)后端。

3.5.1. 从备份中恢复卷

要从备份创建新卷，请完成以下步骤。

流程

1. 查找您要使用的卷备份的 ID：

```
# cinder backup-list
```

确保卷 ID 与您要恢复的卷 ID 匹配。

2. 恢复卷备份：

```
# cinder backup-restore _BACKUP_ID_
```

将 BACKUP_ID 替换为您要使用的卷备份的 ID。

3. 如果您不再需要备份，请删除它：

```
# cinder backup-delete _BACKUP_ID_
```

4. 如果您需要将备份卷恢复到特定类型的卷中，请使用 **--volume** 选项将备份恢复到特定卷：

```
# cinder backup-restore _BACKUP_ID_ --volume VOLUME_ID_
```

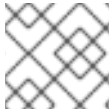
**注意**

如果您从加密备份中恢复卷，则必须加密目标卷类型。

3.5.2. 在块存储数据库丢失后恢复卷

当发生块存储数据库丢失时，您无法恢复卷备份，因为数据库包含卷备份服务所需的元数据。但是，在创建卷备份后，您可以导出并存储元数据，该元数据由 **backup_service** 和 **backup_url** 值组成，以便在发生数据库丢失时恢复卷备份。如需更多信息，请参阅 [第 3.1.1 节“创建完整卷备份”](#)。

如果您导出并存储此元数据，您可以将其导入到新的块存储数据库，允许您恢复卷备份。

**注意**

对于增量备份，您必须先导入所有导出的数据，然后才能恢复其中一个增量备份。

流程

1. 作为具有管理特权的用户，运行以下命令：

```
# cinder backup-import _backup_service _backup_url_
```

使用您导出的元数据替换 **backup_service** 和 **backup_url**。例如，使用从 [第 3.1.1 节“创建完整卷备份”](#) 导出的元数据：

```
# cinder backup-import cinder.backup.drivers.swift eyJzdGF0dXMi...c2l6ZSI6IDF9
+-----+-----+
| Property |      Value      |
+-----+-----+
| id | 77951e2f-4aff-4365-8c64-f833802eaa43 |
| name |      None      |
+-----+-----+
```

2. 将元数据导入到块存储服务数据库后，您可以正常恢复卷，请参阅 [第 3.5.1 节“从备份中恢复卷”](#)。

3.5.3. 取消备份恢复

要取消备份恢复操作，请将备份的状态更改为 **恢复** 以外的任何内容。您可以使用 **错误状态** 来最大程度降低恢复是否成功的混淆。或者，您可以将值更改为 **available**。

```
$ openstack volume backup set --state error BACKUP_ID
```

**注意**

备份取消是一个异步操作，因为备份驱动程序必须在取消备份前检测状态更改。当目标卷中状态更改为 **available** 时，取消完成。

**注意**

目前 RBD 备份上不提供此功能。



警告

如果在启动后取消恢复操作，则目标卷就无用，因为无法了解数据的数量（若有）实际恢复。

第 4 章 对块存储备份服务进行故障排除

有两种常见的场景会导致备份服务出现的许多问题：

- 当 **cinder-backup** 服务启动时，它会连接到其配置的后端，并将其用作备份的目标。这个连接的问题可能会导致服务失败。
- 当请求备份时，备份服务会连接到卷服务并附加请求的卷。这个连接的问题仅在备份期间识别。

在这两种情况下，日志都包含描述错误的消息。

有关日志文件和服务的更多信息，请参阅 *Logging, Monitoring and Troubleshooting Guide* 中的 [Location of log files for OpenStack services](#) 部分。

有关日志位置和故障排除建议的更多信息，请参阅 *Logging, Monitoring and Troubleshooting Guide* 中的 [Block Storage \(cinder\) 日志文件](#)。

4.1. 验证服务

您可以通过验证服务是否可用，并检查日志文件中的错误消息来诊断许多问题。

验证服务状态后，检查 **cinder-backup.log** 文件。Block Storage 备份服务日志位于 **/var/log/containers/cinder/cinder-backup.log** 中。

流程

1. 在卷上运行 **cinder show** 命令，以查看它是否由主机存储：

```
# cinder show
```

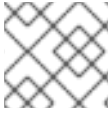
2. 运行 **cinder service-list** 命令来查看正在运行的服务：

```
# cinder service-list
+-----+-----+-----+-----+-----+-----+-----+
---+
| Binary      | Host          | Zone | Status | State | Updated_at           | Disabled
Reason |
+-----+-----+-----+-----+-----+-----+-----+
---+
| cinder-backup | hostgroup     | nova | enabled | up   | 2017-05-15T02:42:25.000000 | -
|
| cinder-scheduler | hostgroup     | nova | enabled | up   | 2017-05-15T02:42:25.000000 | -
|
| cinder-volume  | hostgroup@sas-pool | nova | enabled | down | 2017-05-14T03:04:01.000000 | -
|
| cinder-volume  | hostgroup@ssd-pool | nova | enabled | down | 2017-05-14T03:04:01.000000 | -
+-----+-----+-----+-----+-----+-----+
---+
```

3. 验证预期服务是否可用。

4.2. 查询失败备份的状态

备份是异步的。块存储备份服务在收到 API 请求时会执行少量的静态检查，如检查无效卷引用 (*missing*) 或卷处于 *in-use*，或附加到一个实例。*in-use* 要求您使用 `--force` 选项。



注意

使用 `--force` 选项意味着不会静置 I/O，且生成的卷镜像可能会损坏。

如果 API 接受请求，则备份会在后台进行。通常，即使备份失败或正在处理失败，CLI 也会立即返回。您可以使用 `cinder` 备份 API 查询备份的状态。如果发生错误，请查看日志以发现原因。

4.3. 使用 PACEMAKER 管理资源

默认情况下，Pacemaker 部署块存储备份服务。Pacemaker 将虚拟 IP 地址、容器、服务和其他功能配置为集群中的资源，以确保定义的 Red Hat OpenStack Platform 集群资源集合正在运行并可用。当集群中的服务或整个节点时，Pacemaker 可以重启资源，使节点退出，或重新引导节点。对大多数服务的请求都通过 HAProxy。

有关如何使用 Pacemaker 进行故障排除的详情，请参考 [高可用性部署和使用指南中的使用 Pacemaker 管理高可用性服务](#)。

附录 A. GOOGLE CLOUD STORAGE 配置

要将块存储服务(cinder)配置为使用 Google Cloud Storage 作为备份后端，请完成以下步骤：

1. 创建并下载 Google 帐户的服务帐户凭证：
 - [第 A.1 节“创建 GCS 凭证文件”](#)
 - [第 A.2 节“创建 `cinder-backup-gcs.yaml`”](#)
2. 创建环境文件来映射您需要的块存储设置：
 - [第 A.3 节“使用 Google Cloud 设置创建环境文件”](#)
3. 使用您创建的环境文件重新部署 overcloud：
 - [第 A.4 节“部署 overcloud”](#)



重要

使用 Google Cloud Storage 的 Cinder 备份已被弃用，并将在下一个主发行版本中删除支持。

前提条件

- 您有具有升级权限的帐户的用户名和密码。您可以使用为部署 overcloud 而创建的 **stack** 用户帐户。如需更多信息，请参阅 [Director 安装和使用指南](#)。
- 您有一个 Google 帐户，可访问 Google Cloud Platform。块存储服务使用此帐户访问和使用 Google Cloud 存储备份。

A.1. 创建 GCS 凭证文件

Block Storage 服务(cinder)需要 Google 凭证来访问和使用 Google Cloud 进行备份。您可以通过创建服务帐户密钥向块存储服务提供这些凭证。

流程

1. 使用您的 Google 帐户登录到 Google 开发人员控制台<http://console.developers.google.com>。
2. 点 **Credentials** 选项卡，然后从 **Create credentials** 下拉菜单中选择 **Service account key**。
3. 在 **Create service account key** 屏幕中，从 **Service account** 下拉菜单中选择您希望块存储服务使用的服务帐户。
4. 在同一屏幕中，从 **Key type** 部分中选择 **JSON**，再点 **Create**。浏览器将下载密钥到其默认下载位置。
5. 打开该文件并记录 **project_id** 参数的值：

```
{
  "type": "service_account",
  "project_id": "*cloud-backup-1370*",
  ...
}
```

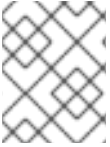
- 将 GCS JSON 凭证的副本保存到 `/home/stack/templates/Cloud-Backup.json` 中。

重要的

将文件命名为 **Cloud-Backup.json**，不会更改文件名。此 JSON 文件必须与作为您在第 A.2 节“创建 `cinder-backup-gcs.yaml`”中的步骤中创建的 `cinder-backup-gcs.yaml` 文件位于同一个目录中。

A.2. 创建 CINDER-BACKUP-GCS.YAML

使用提供的示例文件，创建 `cinder-backup-gcs.yaml` 文件。



注意

这个示例中使用的空格和格式（及您的文件中）至关重要。如果更改了空格，则该文件可能无法按预期工作。

流程

- 复制下面的文本，将其粘贴到新文件中。不要对文件内容进行任何修改。

```
heat_template_version: rocky

description: >
  Post-deployment for configuration cinder-backup to GCS

parameters:
  servers:
    type: json
  DeployIdentifier:
    type: string

resources:
  CinderBackupGcsExtraConfig:
    type: OS::Heat::SoftwareConfig
    properties:
      group: script
      config:
        str_replace:
          template: |
            #!/bin/bash
            GCS_FILE=/var/lib/config-data/puppet-generated/cinder/etc/cinder/Cloud-
Backup.json
            HOSTNAME=$(hostname -s)
            for NODE in $(hiera -c /etc/puppet/hiera.yaml cinder_backup_short_node_names | tr
-d '[]','); do
              if [ $NODE == $HOSTNAME ]; then
                cat <<EOF > $GCS_FILE
                GCS_JSON_DATA
                EOF
                chmod 0640 $GCS_FILE
                chown root:42407 $GCS_FILE
              fi
            done
        params:
          GCS_JSON_DATA: {get_file: Cloud-Backup.json}
```

```
CinderBackupGcsDeployment:
  type: OS::Heat::SoftwareDeploymentGroup
  properties:
    servers: {get_param: servers}
    config: {get_resource: CinderBackupGcsExtraConfig}
    actions: ['CREATE','UPDATE']
    input_values:
      deploy_identifier: {get_param: DeployIdentifier}
```

2. 将文件保存为 `/home/stack/templates/cinder-backup-gcs.yaml`。

A.3. 使用 GOOGLE CLOUD 设置创建环境文件

创建环境文件，使其包含您要应用到块存储服务(cinder)的设置。在这种情况下，环境文件配置块存储服务，将卷备份存储到 Google Cloud。有关环境文件的更多信息，请参阅 [Director 安装和使用指南](#)。

使用以下示例环境文件，并使用 **Cloud-Backup.json** 文件中列出的项目 ID 更新 **backup_gcs_project_id**。您还可以将 **backup_gcs_bucket_location** 位置从美国更改为更接近的位置。

有关 Google Cloud Backup Storage 备份后端的配置选项列表，请参阅 [表 A.1 “Google Cloud Storage 备份后端配置选项”](#)。

流程

1. 复制以下环境文件示例。保留空格使用情况。
2. 将内容粘贴到新文件中：`/home/stack/templates/cinder-backup-settings.yaml`。
3. 将 **backup_gcs_project_id** 的值从 **cloud-backup-1370** 更改为 **Cloud-Backup.json** 文件中列出的项目 ID。
4. 保存该文件。

环境文件示例

在环境文件中定义每个设置。使用 [表 A.1 “Google Cloud Storage 备份后端配置选项”](#) 选择可用的配置选项。

```
resource_registry:
  OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-templates/deployment/cinder/cinder-backup-pacemaker-puppet.yaml
  # For non-pcmk managed implementation
  # OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-templates/deployment/cinder/cinder-backup-container-puppet.yaml
  OS::TripleO::NodeExtraConfigPost: /home/stack/templates/cinder-backup-gcs.yaml

parameter_defaults:
  CinderBackupBackend: swift
  ExtraConfig:
    cinder::backup::swift::backup_driver: cinder.backup.drivers.gcs.GoogleBackupDriver
    cinder::config::cinder_config:
      DEFAULT/backup_gcs_credential_file:
        value: /etc/cinder/Cloud-Backup.json
      DEFAULT/backup_gcs_project_id:
```

```

value: cloud-backup-1370
DEFAULT/backup_gcs_bucket:
value: cinder-backup-gcs
DEFAULT/backup_gcs_bucket_location:
value: us

```

表 A.1. Google Cloud Storage 备份后端配置选项

参数	默认	CONFIG 描述
backup_gcs_project_id		必需。您要使用的服务帐户的项目 ID，包含在来自 第 A.1 节“创建 GCS 凭证文件” 的服务帐户密钥的 project_id 中。
backup_gcs_credential_file		您在 第 A.1 节“创建 GCS 凭证文件” 中创建的服务帐户密钥文件的绝对路径。
backup_gcs_bucket		要使用的 GCS 存储桶或对象存储存储库（可能或可能不存在）。如果您指定了不存在的存储桶，Google Cloud Storage 备份驱动程序会创建一个，并为它分配此处指定的名称。如需更多信息，请参阅 Buckets 和 Bucket 名称要求 。
backup_gcs_bucket_location	us	GCS 存储桶的位置。只有在 backup_gcs_bucket 指定不存在的存储桶时，才会使用这个值。在这种情况下，Google Cloud Storage 备份驱动程序指定它作为 GCS 存储桶位置。
backup_gcs_object_size	52428800	GCS 备份对象的大小（以字节为单位）。
backup_gcs_block_size	32768	跟踪用于增量备份的大小（以字节为单位）。这个值必须是 backup_gcs_object_size 值的倍数。
backup_gcs_user_agent	gcscinder	GCS API 的 HTTP user-agent 字符串。
backup_gcs_reader_chunk_size	2097152	GCS 对象以这个大小的块下载，以字节为单位。
backup_gcs_writer_chunk_size	2097152	GCS 对象以这个大小的块形式上传，以字节为单位。要将文件作为单个块上传，请使用值 -1。

参数	默认	CONFIG 描述
<code>backup_gcs_num_retries</code>	3	尝试的重试次数。
<code>backup_gcs_storage_class</code>	NEARLINE	GCS 存储桶的存储类。只有在 <code>backup_gcs_bucket</code> 指定不存在的存储桶时，才会使用这个值。在这种情况下，Google Cloud Storage 备份驱动程序指定它作为 GCS 存储桶类。如需更多信息，请参阅 存储类 。
<code>backup_gcs_retry_error_codes</code>	429	GCS 错误代码列表。
<code>backup_gcs_enable_progress_timer</code>	true	布尔值，用于启用或禁用卷备份期间向 Telemetry 服务 (ceilometer) 发送定期进度通知的计时器。这默认启用(True)。



警告

创建新存储桶时，Google Cloud Storage 会根据您选择的存储类收费 (`backup_gcs_storage_class`)。默认的 **NEARLINE** 类适用于备份服务。



警告

您不能在创建存储桶后编辑存储桶的位置或类。如需更多信息，请参阅[管理存储桶的存储类或位置](#)。

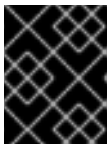
A.4. 部署 OVERCLOUD

当您在 `/home/stack/templates/` 中创建环境文件后，部署 `overcloud`，然后重启 `cinder-backup` 服务：

流程

1. 以 **stack** 用户身份登录。
2. 部署配置：

```
$ openstack overcloud deploy --templates \
-e /home/stack/templates/cinder-backup-settings.yaml
```



重要

如果您在创建 overcloud 时传递任何额外的环境文件，请使用 `-e` 选项再次传递它们，以避免对 overcloud 进行不必要的更改。

3. 部署完成后重启 `cinder-backup` 服务。

如需更多信息，请参阅 Director 安装和使用指南中的[Overcloud 创建中包括的环境文件](#)，以及高级 Overcloud 自定义指南中的[环境文件](#)部分。

附录 B. 高级块存储配置

在 director 部署的安装前，`cinder.conf` 文件配置了块存储服务和备份服务。当 `cinder.conf` 的值没有等效的编配(heat)模板时，您可以使用自定义环境文件将值传递给 director。将值添加到自定义环境文件的 `parameter_defaults` 部分中的 `ExtraConfig` 部分，如 `cinder-backup-settings.yaml` 文件。

B.1. 高级配置选项

使用 `ExtraConfig`，您可以为所有节点上的集群添加额外的层次结构配置。这些设置包含在专用备份节点上。但是，如果您使用 `ControllerExtraConfig` 而不是 `ExtraConfig`，您的设置会在 Controller 节点上安装，而不是在专用备份节点上安装。

您可以为来自 `cinder.conf` 文件的 `DEFAULT` 部分的设置替换 `DEFAULT/[cinder.conf setting]`。以下示例演示了 `ExtraConfig` 条目如何出现在 YAML 文件中：

```
parameter_defaults:
  ExtraConfig:
    cinder::config::cinder_config:
      DEFAULT/backup_compression_algorithm:
        value: None
```

表 B.1 列出了与备份相关的示例选项。

表 B.1. 块存储备份服务配置选项

选项	类型	默认值	描述
<code>backup_service_inithost_offload</code>	选填	true	在备份服务启动过程中卸载待处理的备份删除。如果为 false，则备份服务会一直保持关闭，直到所有待处理的备份都被删除。
<code>use_multipath_for_image_xfer</code>	选填	False	使用多路径在备份和恢复过程中使用多路径附加卷。这会影响所有 cinder 附加操作，如从镜像、通用冷迁移和其他操作创建卷。
<code>num_volume_device_scan_tries</code>	选填	3	在附加过程中重新扫描目标以查找卷的次数上限。
<code>backup_workers</code>	选填	1	要运行的备份进程数量。使用压缩运行多个并发备份或恢复会导致显著的性能提升。

选项	类型	默认值	描述
backup_native_threads_pool_size	选填	60	备份的原生线程池的大小。大多数备份驱动程序主要依赖于这一点。您可以减少不依赖于这个选项的特定驱动程序的值。
backup_share	必填		设置为 <code>HOST:_EXPORT_PATH_</code> 。
backup_container	选填	无	(字符串) 用于备份的自定义目录。
backup_enable_progress_timer	选填	true	启用(true)或禁用(false)计时器, 在将卷备份到后端存储时, 将定期进度通知发送到 Telemetry 服务 (ceilometer)。
backup_mount_options	选填		当您挂载在 backup_share 中指定的 NFS 导出时, 可以指定以逗号分隔的选项列表。
backup_mount_point_base	选填	<code>\$state_path/backup_mount</code>	(字符串) 包含 NFS 共享挂载点的基本目录。
backup_compression_algorithm	选填	zlib	将备份数据发送到存储库时使用的压缩算法。有效值为 zlib 、 bz2 和 None 。
backup_file_size	选填	1999994880	大于这个值的 cinder 卷的数据作为多个文件存储在备份存储库中。这个选项必须是 backup_sha_block_size_bytes 的倍数。
backup_sha_block_size_bytes	选填	32768	数字签名计算的 cinder 卷块的大小