



# Red Hat OpenStack Platform 13

## Block Storage バックアップガイド

OpenStack での Block Storage のバックアップサービスの理解、使用、管理



# Red Hat OpenStack Platform 13 Block Storage バックアップガイド

---

OpenStack での Block Storage のバックアップサービスの理解、使用、管理

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Block\_Storage\_Backup\_Guide.ent file |.

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<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

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

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> 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<sup>®</sup> 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.

## 概要

本書では、OpenStack Block Storage のバックアップサービスをデプロイする方法を説明します。ここで説明する手順は、オーバークラウドのデプロイメントに固有のもので、OpenStack director は、バックエンドとして Red Hat Ceph Storage、NFS、および Object Storage (swift) を設定できます。

## 目次

はじめに .....	3
<b>第1章 RED HAT OPENSTACK PLATFORM BLOCK STORAGE バックアップサービスの概要</b> .....	<b>4</b>
1.1. バックアップとは	4
1.2. バックアップおよび復元の仕組み	5
1.2.1. ボリュームバックアップのワークフロー	5
1.2.2. ボリュームの復元のワークフロー	6
<b>第2章 BLOCK STORAGE バックアップサービスのデプロイメント</b> .....	<b>9</b>
2.1. バックエンドオプションの設定	9
2.1.1. Object Storage (swift)。	9
2.1.2. Red Hat Ceph Storage	10
2.1.3. NFS	10
2.2. BLOCK STORAGE バックアップサービスのデプロイ	11
<b>第3章 BLOCK STORAGE バックアップサービスの使用</b> .....	<b>12</b>
3.1. 完全バックアップ	12
3.1.1. フルボリュームバックアップの作成	12
3.1.2. 管理者としてのボリュームバックアップの作成	12
3.1.3. ボリュームからのバックアップメタデータのエクスポート	13
3.1.4. 使用中のボリュームのバックアップ	13
3.1.5. スナップショットのバックアップ	14
3.2. 増分バックアップ	14
3.2.1. パフォーマンスに関する考慮事項	14
3.2.1.1. スナップショットの影響からのバックアップ	15
3.2.2. 増分バックアップの実行	15
3.3. バックアップのキャンセル	15
3.4. テナントのバックアップクォータの表示および変更	16
3.5. バックアップからの復元	16
3.5.1. バックアップからのボリュームの復元	16
3.5.2. Block Storage データベースの損失後のボリュームの復元	17
3.5.3. 復元のキャンセル	18
3.6. トラブルシューティング	18
3.6.1. サービスの確認	19
3.6.2. トラブルシューティングのヒント	20
3.6.3. Pacemaker	20
<b>付録A 高度な BLOCK STORAGE のバックアップ設定オプション</b> .....	<b>21</b>



# はじめに

# 第1章 RED HAT OPENSTACK PLATFORM BLOCK STORAGE バックアップサービスの概要

Red Hat OpenStack Platform (RHOSP) は、Red Hat Enterprise Linux 上にプライベートまたはパブリックの Infrastructure-as-a-Service (IaaS) クラウドを構築するための基盤を提供します。RHOSP は、クラウド対応のワークロード開発向けのスケーラビリティおよび耐障害性に優れたプラットフォームです。

RHOSP Dashboard またはコマンドラインクライアントメソッドのどちらかを使用して、バックアップサービスのほとんど機能を管理することができます。ただし、一部のより高度な手順を実施するには、コマンドラインを使用する必要があります。



## 注記

Red Hat OpenStack Platform の全ドキュメントスイートは [Red Hat OpenStack Platform の製品ドキュメント](#) で参照してください。

Block Storage Service (cinder) には、cinder ボリュームのバックアップをさまざまなストレージバックエンドに提供するために使用できる、水平方向にスケーラブルなバックアップサービスが同梱されています。Block Storage バックアップサービスを使用して、完全バックアップまたは増分バックアップを作成し、これらのバックアップを復元できます。このサービスはボリュームアレイに依存しません。

Red Hat OpenStack Platform director は、完全な OpenStack 環境のインストールおよび管理を行うためのツールセットです。Red Hat OpenStack director は、最小限の手動設定で機能するエンタープライズレベルの OpenStack デプロイメントをオーケストレーションします。これは、個別の OpenStack コンポーネントを手動で設定することに固有の問題の多くに対応するのに役立ちます。

director によりデプロイされた終了結果は、オーバークラウドと呼ばれます。オーバークラウドには、Block Storage など、エンドユーザーにサービスを提供するすべてのコンポーネントが含まれます。Block Storage バックアップサービスは、コントローラーノードにデプロイされるオプションのサービスです。

本書では、特定のバックエンドを使用するために Block Storage バックアップサービスをデプロイする方法についてのガイダンスを提供します。本ガイドでは、Block Storage バックアップサービスのプランニング、インストール、設定および使用について説明します。

## 1.1. バックアップとは

ボリュームバックアップは、ボリュームのコンテンツの永続コピーです。ボリュームバックアップは通常、オブジェクトストアとして作成されます。デフォルトでは、OpenStack Object Storage サービス (swift) で管理されます。オプションで、バックアップの代替バックエンドとして Red Hat Ceph Storage および NFS を設定できます。

ボリュームバックアップを作成すると、バックアップメタデータはすべて Block Storage サービスデータベースに保存されます。cinder-backup は、メタデータを使用してバックアップからボリュームを復元します。致命的なデータベースの損失からデータを復元する場合は、バックアップからボリュームを復元する前に、Block Storage サービスデータベースを復元する必要があります。このリカバリーのシナリオでは、Block Storage サービスデータベースが元のすべてのボリュームバックアップメタデータそのまま復元されることを想定しています。

ボリュームバックアップをデータのサブセットのみに設定する場合は、ボリュームのバックアップメタデータをエクスポートする必要があります。ボリュームメタデータのバックアップを使用すると、REST API または cinder クライアントを使用して Block Storage データベースにメタデータを再インポートし、通常どおりボリュームのバックアップを復元できます。



ボリュームのバックアップはスナップショットとは異なります。バックアップはボリュームに含まれるデータを保持し、データの損失を防ぐために使用されます。スナップショットは、特定の時点でボリュームの状態を保持し、クローン作成を容易にするために使用されます。スナップショットが存在している場合にはボリュームを削除することはできません。

クローン作成中のレイテンシーを最小限に抑えるために、スナップショットバックエンドは通常、ボリュームのバックエンドと共存します。一般的なエンタープライズデプロイメントでは、バックアップリポジトリは、異なるノード、物理ストレージ、地理的な場所など、ボリュームバックエンドとは別の場所にあります。これにより、ボリュームのバックエンドに発生する可能性のある破損からバックアップリポジトリを保護します。

ボリュームスナップショットの詳細は、ストレージガイドの [ボリュームスナップショットの作成、使用、または削除](#) を参照してください。

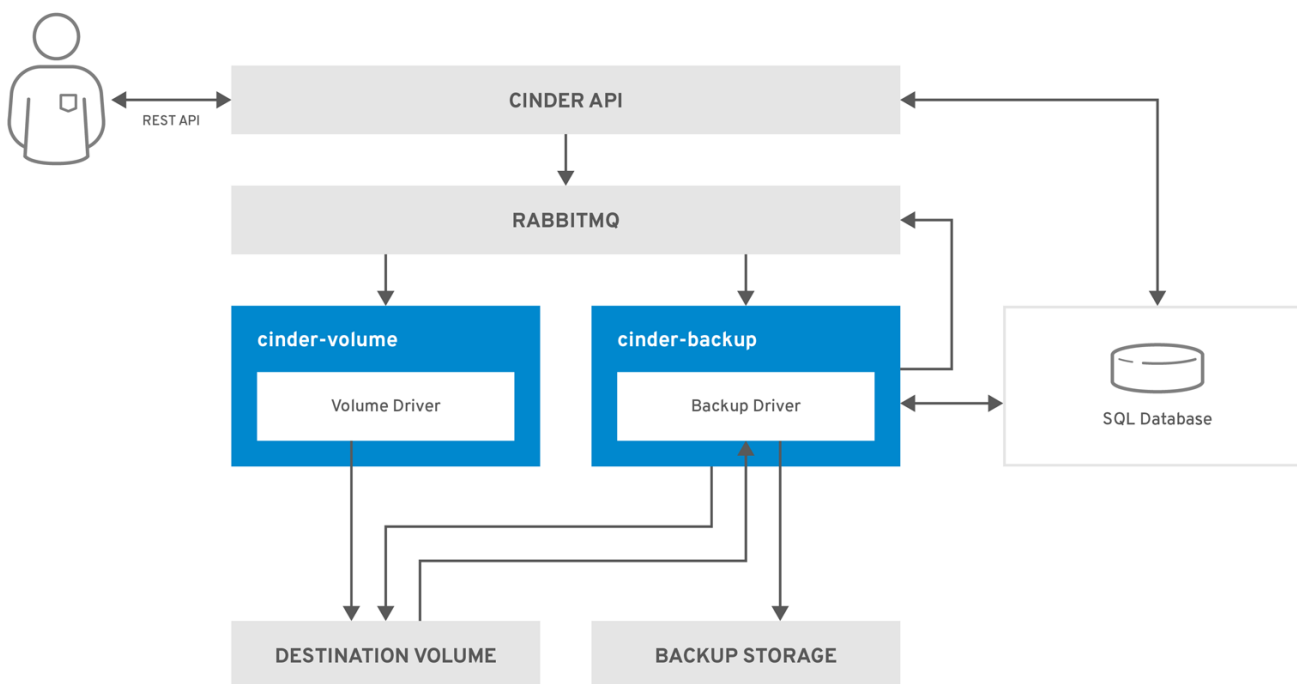
## 1.2. バックアップおよび復元の仕組み

ボリュームのバックアップおよび復元には同様のワークフローがあります。

### 1.2.1. ボリュームバックアップのワークフロー

Block Storage バックアップサービスがバックアップを実行すると、cinder API から目的のボリュームのバックアップに対する要求を受け取り、バックアップコンテンツをバックエンドに保存します。

以下の図は、要求が cinder サービスと対話してバックアップを実行する方法を示しています。



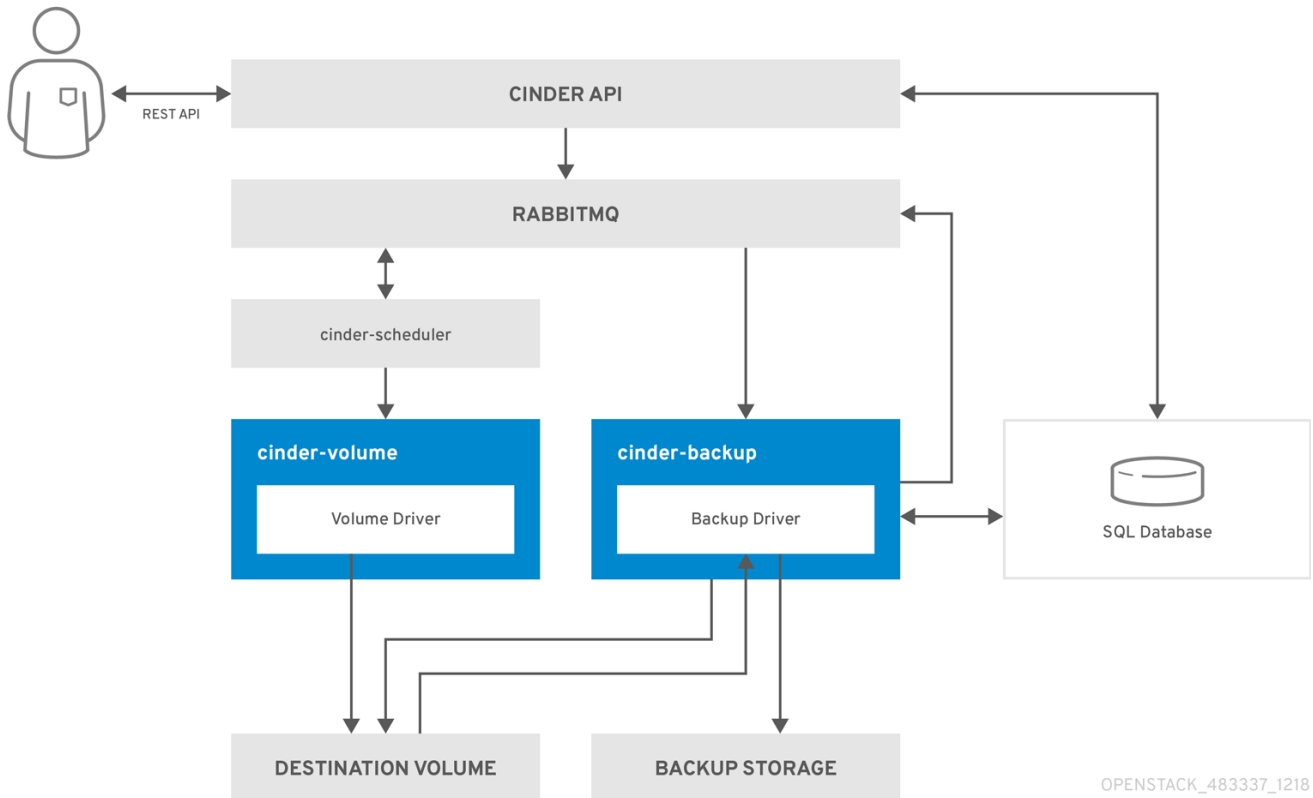
OPENSTACK\_483337\_1218

1. クライアントは cinder REST API を呼び出して cinder ボリュームをバックアップするリクエストを発行します。
2. cinder API サービスは HAProxy から要求を受信し、要求、ユーザー認証情報などを検証します。
  - a. SQL データベースにバックアップレコードを作成します。

- b. AMQP を介して、cinder-backup への非同期 RPC 呼び出しを行い、ボリュームのバックアップを作成します。
  - c. API 呼び出し元に、現在のバックアップレコード (ID) を返します。
3. RPC 作成メッセージは、バックアップサービスのいずれかに届きます。
4. cinder-backup は、get\_backup\_device への同期 RPC 呼び出しを行います。
5. cinder-volume は、正しいデバイスが呼び出し元に返されるようにします。通常は同じボリュームですが、ボリュームが使用中の場合は、設定によっては一時クローンボリュームまたは一時スナップショットが返されます。
6. cinder-backup は、別の同期 RPC を発行して、ソースデバイスを公開するように cinder-volume します。
7. cinder-volume サービスは、ソースデバイス (ボリュームまたはスナップショット) をエクスポートしてマッピングし、適切な接続情報を返します。
8. cinder-backup は、接続情報を使用してソースボリュームを接続します。
9. cinder-backup は、デバイスが接続されている状態でバックアップドライバーを呼び出し、バックアップ先へのデータ転送を開始します。
10. ボリュームがバックアップホストから切り離されている。
11. cinder-backup は、同期 RPC を cinder-volume に発行して、ソースデバイスの接続を解除します。
12. cinder-volume サービスは、デバイスのマッピングを解除し、エクスポートを削除します。
13. 一時ボリュームまたは一時スナップショットが作成された場合、cinder-backup は cinder-volume を呼び出してそのボリュームを削除します。
14. cinder-volume により、一時ボリュームが削除されます。
15. バックアップが完了すると、データベースのバックアップレコードが更新されます。

### 1.2.2. ボリュームの復元のワークフロー

以下の図は、ユーザーが Block Storage バックアップを復元する際に発生する手順を示しています。



1. クライアントは CinderREST API を呼び出して cinder バックアップを復元するリクエストを発行します。
2. cinder API は HAProxy から要求を受信し、要求、ユーザー認証情報などを検証します。
3. 要求に宛先として既存のボリュームが含まれていない場合、API は非同期 RPC 呼び出しを行って新しいボリュームを作成し、ボリュームのステータスをポーリングして利用可能になります。
4. cinder-scheduler がボリュームサービスを選択し、RPC 呼び出しを実行してボリュームを作成します。
5. 選択した cinder-ボリュームにより、音量が作成されます。
6. cinder-api が利用可能なボリュームを検出すると、バックアップレコードがデータベースに作成されます。
7. cinder-api は、AMQP 経由でバックアップサービスへの非同期 RPC 呼び出しを行い、バックアップを復元し、現在のボリューム ID、バックアップ ID、およびボリューム名を API 呼び出し元に返します。
8. RPC 作成メッセージは、バックアップサービスのいずれかに届きます。
9. cinder-backup は、cinder-volume への同期 RPC 呼び出しを実行して、宛先ボリュームを公開します。
10. cinder-volume は、適切な接続情報を返す宛先ボリュームをエクスポートしてマッピングします。
11. cinder-backup は、接続情報を使用してソースボリュームを接続します。
12. cinder-backup サービスは、デバイスが接続されている状態でドライバーを呼び出し、ボリュームデスティネーションへのデータの復元を開始します。

13. ボリュームがバックアップホストから切り離されている。
14. cinder-backup は、同期 RPC を cinder-volume に発行して、ソースデバイスの接続を解除します。
15. cinder-volume サービスは、デバイスのマッピングを解除し、エクスポートを削除します。
16. バックアップが完了すると、データベースのバックアップレコードが更新されます。

## 第2章 BLOCK STORAGE バックアップサービスのデプロイメント

Block Storage バックアップサービスはオプションです。これはデフォルトではインストールされないため、オーバークラウドのデプロイメントに追加する必要があります。

バックアップサービスをデプロイするには、以下が必要です。

- 新規または既存の OpenStack インストール
- 互換性のあるバックアップドライバーを備えた利用可能なストレージソース：Object Storage (swift)、Red Hat Ceph Storage、または NFS。

本セクションの例では、デフォルトの Pacemaker インストールを使用する標準の OpenStack 環境にバックエンドサービスをデプロイすることを前提としています。

### 2.1. バックエンドオプションの設定

バックアップサービスは、`/usr/share/openstack-tripleo-heat-templates/environments/` ディレクトリに `cinder-backup.yaml` 環境ファイルを追加して有効化されます。

このファイルのデフォルト設定は、Pacemaker を使用した Block Storage バックアップサービスの swift バックエンドを設定します。

#### 手順

カスタム環境ファイル(`cinder-backup-settings.yaml` など)を作成します。このファイルには、ドライバーのバックアップサービスおよび設定オプション用のパラメーター設定が含まれます。

1. `cinder-backup.yaml` ファイルのコピーを作成し、これを他のカスタムテンプレートと同じ場所に保存します。

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

2. 使用しているバックアップバックエンドに適切なオプションを変更します（以下のセクションの手順を参照）。
3. 変更をファイルに保存します。

#### 2.1.1. Object Storage (swift)。

Swift は CinderBackupBackend オプションのデフォルト値です。swift を使用している場合、追加の変更は必要ありません。

#### 例

```
resource_registry:  
  OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-  
  templates/docker/services/pacemaker/cinder-backup.yaml  
  # For non-pcmk managed implementation  
  # OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-  
  templates/docker/services/cinder-backup.yaml  
  
parameter_defaults:  
  CinderBackupBackend: swift
```

設定	オプション	値
<b>CinderBackupBackend</b>	Swift (デフォルト)	swift は、 <b>cinder-backup.yaml</b> テンプレートのデフォルト選択です。

## 2.1.2. Red Hat Ceph Storage

Red Hat Ceph Storage をバックアップバックエンドとして使用する場合は、バックアップに使用される RBD プール名を変更できます。デフォルト値は **backups** です。

### 例

```
resource_registry:
  OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-
  templates/docker/services/pacemaker/cinder-backup.yaml
  # For non-pcmk managed implementation
  # OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-
  templates/docker/services/cinder-backup.yaml

parameter_defaults:
  CinderBackupBackend: ceph
  CinderBackupRbdPoolName: backups
```

設定	オプション	値
<b>CinderBackupBackend</b>	ceph	必須。値を <b>ceph</b> に変更します。
<b>CinderBackupRbdPoolName</b>	backups (デフォルト名)	オプション。カスタム RBD プール名を使用しない限り、その他の設定を更新する必要はありません。

## 2.1.3. NFS

バックアップサービスのバックエンドとして NFS を使用するには、マウントする NFS 共有を指定する必要があります。

### 例

```
resource_registry:
  OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-
  templates/docker/services/pacemaker/cinder-backup.yaml
  # For non-pcmk managed implementation
  # OS::TripleO::Services::CinderBackup: /usr/share/openstack-tripleo-heat-
  templates/docker/services/cinder-backup.yaml

parameter_defaults:
  CinderBackupBackend: nfs
  CinderBackupNfsShare: '192.168.122.1:/export/cinder/backups'
  CinderBackupNfsMountOptions: ''
```

設定	オプション	値
<b>CinderBackupBackend</b>	nfs	必須。 <b>nfs</b> を値として設定します。
<b>CinderBackupNfsShare</b>		必須。マウントする NFS 共有を入力します。デフォルト値は空です。
<b>CinderBackupNfsMountOptions</b>		オプション。バックアップ NFS マウントオプションは空白のままにすることができます。マウントオプションを指定する必要がある場合は、ここに追加します。

## 2.2. BLOCK STORAGE バックアップサービスのデプロイ

`/home/stack/templates/` に環境ファイルを作成したら、`stack` ユーザーとしてログインして、以下のコマンドを実行して設定をデプロイします。

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



### 重要

オーバークラウドの作成時に追加の環境ファイルを渡した場合には、予定外の変更がオーバークラウドに加えられないように、ここで **-e** オプションを使用して環境ファイルを再度渡します。

詳細は、Director インストールおよび使用ガイドの [Overcloud Creation に環境ファイルを含める](#) および [高度なオーバークラウドカスタマイズガイドの環境ファイル](#) を参照してください。

## 第3章 BLOCK STORAGE バックアップサービスの使用

本章では、Block Storage バックアップサービスを使用して完全バックアップまたは増分バックアップを実行する方法、ボリュームにバックアップを復元する方法、および基本的なトラブルシューティングのヒントを説明します。

### 3.1. 完全バックアップ

#### 3.1.1. フルボリュームバックアップの作成

ボリュームのバックアップを作成するには、**backup-create** コマンドを使用します。デフォルトでは、このコマンドはボリュームの完全バックアップを作成します。ただし、ボリュームに既存のバックアップがある場合は、増分バックアップを作成できます。増分バックアップの作成に関する詳細は、ストレージガイドの [セクション 2.4.1.2 増分バックアップの作成](#) を参照してください。

アクセス可能なボリュームのバックアップを作成できます。つまり、管理者特権を持つユーザーは、所有者に関係なく、どのボリュームでもバックアップを作成できます。詳細は「[管理者としてのボリュームバックアップの作成](#)」を参照してください。

#### 手順

1. バックアップを作成するボリュームの ID または表示名を表示します。

```
# cinder list
```

2. ボリュームをバックアップします。

```
# cinder backup-create --name <full_backup_name> _VOLUME_
```

**VOLUME** の箇所は、バックアップするボリュームの **ID** または **Display Name** に置き換えます。以下に例を示します。

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

作成されるバックアップの **volume\_id** は、ソースボリュームの ID と同じです。

3. ボリュームのバックアップが完了したことを確認します。

```
# cinder backup-list
```

ボリュームバックアップの作成は、バックアップエントリーの **Status** が利用可能になると完了します。

#### 3.1.2. 管理者としてのボリュームバックアップの作成



デフォルトの admin アカウントなどの管理者権限を持つユーザーは、OpenStack が管理するボリュームをバックアップできます。管理者が管理者以外のユーザーが所有するボリュームのバックアップを作成すると、デフォルトでボリュームの所有者からバックアップが非表示になります。

## 手順

ボリュームをバックアップし、特定のテナントでバックアップを利用できるようにするには、以下を実行します。

```
# cinder --os-tenant-name _TENANTNAME_ backup-create _VOLUME_
```

詳細は以下のようになります。

- <TENANTNAME> は、バックアップを利用可能にするテナントの名前です。
- <VOLUME> は、バックアップを作成するボリュームの名前または ID です。

この操作を実行する場合、バックアップのサイズは、管理者のテナントではなく **TENANTNAME** のクォータに対してカウントされます。

### 3.1.3. ボリュームからのバックアップメタデータのエクスポート

ボリュームからバックアップメタデータをエクスポートおよび保存できます。ボリュームからバックアップメタデータをエクスポートして保存することで、Block Storage データベースで致命的な損失が発生した場合でも、ボリュームを復元できます。これは、ボリュームのサブセットの一部のみの完全なバックアップが必要な場合に役立ちます。

ボリュームのバックアップを作成し、そのメタデータをエクスポートすると、別の Block Storage データベースまたはクラウドサービスにボリュームを復元できます。ボリュームの作成時に UUID 暗号化を指定すると、ボリュームの暗号化はバックアップメタデータに保持されます。復元された暗号化されたボリュームは、認証情報を使用してアクセスできます。

## 手順

以下のコマンドを実行して、バックアップメタデータをエクスポートおよび保存します。

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

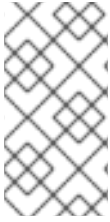
**BACKUPID** は、ボリュームバックアップの ID または名前に置き換えます。以下に例を示します。

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

ボリュームバックアップメタデータは、**backup\_service** 値と **backup\_url** 値で構成されます。

### 3.1.4. 使用中のボリュームのバックアップ

**cinder backup-create** コマンドを使用して、**--force** オプションを追加して、使用中のボリュームのバックアップを作成することができます。



## 注記

**--force** オプションは、Block Storage バックエンドスナップショットのサポートに依存しており、ほとんどのドライバーでサポートされる必要があります。使用しているバックエンドのドキュメントを確認することで、スナップショットのサポートを確認できます。

**--force** を使用することで、ドライブを静止していないことを確認し、バックアップを実行します。この方法を使用すると、クラッシュの一貫性がありますが、アプリケーションの一貫性はありません。バックアップが作成されます。つまり、バックアップの実行時に実行していたアプリケーションを認識しません。ただし、データはそのままになります。

## 手順

使用中のボリュームのバックアップを作成するには、次のコマンドを実行します。

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

### 3.1.5. スナップショットのバックアップ

スナップショットから完全バックアップを作成する場合は、そのスナップショットに関連付けられたボリューム ID を使用します。

## 手順

1. **cinder snapshot list** を使用して、バックアップを作成するスナップショットのスナップショット ID を特定します。

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

2. スナップショットの名前が の場合は、以下の例を使用して **ID** を特定できます。

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

3. スナップショットのバックアップを作成します。

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



## 注記

**--snapshot-id** を使用すると、NFS ボリュームのスナップショットベースのバックアップが失敗します。これは既知の問題です。

## 3.2. 増分バックアップ

Block Storage バックアップサービスは、増分バックアップを実行するオプションを提供します。

### 3.2.1. パフォーマンスに関する考慮事項

増分やデータ圧縮などの一部のバックアップ機能は、パフォーマンスに影響を与える可能性があります。増分バックアップは、ボリューム内のデータをすべて読み込み、完全バックアップと各増分バックアップの両方でサムチェックを行う必要があるため、パフォーマンスに影響を及ぼします。

データ圧縮は、Ceph 以外のバックエンドで使用できます。データ圧縮を有効にするには、追加の CPU 電力が必要になりますが、使用するネットワーク帯域幅とストレージ領域は全体で少なくなります。

マルチパスの設定はパフォーマンスにも影響します。マルチパスを有効にせずに複数のボリュームが接続されている場合は、ボリュームに接続できないか、完全なネットワーク機能が発生する可能性があります、パフォーマンスに影響が出る可能性があります。

詳細設定オプションを使用して、圧縮の有効化または無効化、プロセス数の定義、およびその他の CPU リソースの追加を行うことができます。

### 3.2.1.1. スナップショットの影響からのバックアップ

バックエンドの中には、スナップショットからのバックアップの作成に対応するものもあります。この機能をサポートするドライバーは、スナップショットをボリュームに直接アタッチすることができます。これは、スナップショットをボリュームにクローンするよりも速くなります。通常、この機能はスナップショットからボリュームを作成する手順が追加されたため、パフォーマンスに影響を与える可能性があります。

### 3.2.2. 増分バックアップの実行

デフォルトでは、**cinder backup-create** はボリュームのフルバックアップを作成します。ボリュームに既存のバックアップがある場合は、代わりに増分バックアップを作成できます。

増分バックアップは、NFS、オブジェクトストレージ (swift)、および Red Hat Ceph ストレージバックアップリポジトリで完全にサポートされています。Ceph バックアップは現在、**--incremental** オプションを無視します。Ceph バックアップは、ソースが Ceph ボリュームである場合に増分バックアップの実行を試みるため、複数の完全バックアップを実行することはできません。



#### 注記

増分 Ceph バックアップは、Ceph 以外のボリュームに対しては実行できません。

増分バックアップは、最後の完全バックアップまたは増分バックアップ以降のボリュームへの変更をキャプチャーします。ボリュームのサイズは時間とともに増加するため、ボリュームの多数の定期的なフルバックアップを実行すると、リソースを大量に消費できます。増分バックアップを使用すると、リソースの使用量を最小限に抑えながらボリュームへの定期的な変更をキャプチャーできます。

#### 手順

増分ボリュームバックアップを作成するには、**--incremental** オプションを使用します。

```
# cinder backup-create --name <incremental_backup_name> --incremental _VOLUME_
```

**VOLUME** の箇所は、バックアップするボリュームの **ID** または **Display Name** に置き換えます。



#### 注記

すでに増分バックアップがある場合、完全バックアップを削除することはできません。フルバックアップに複数の増分バックアップがある場合は、最新の増分バックアップのみを削除できます。

## 3.3. バックアップのキャンセル

バックアップをキャンセルするには、管理者がバックアップで強制削除を要求する必要があります。

```
# cinder backup-delete --force BACKUP ID
```

バックアップがすぐに削除され、リストに表示されなくなります。ソースリソースのステータスをチェックして、ステータスがbacking-up でなくなるかどうかを確認します。



### 注記

Red Hat OpenStack バージョン 12 以前では、スナップショットのバックアップを作成している場合でも、backing-up の状態はボリュームに保存されていました。したがって、スナップショットのバックアップを作成する際に、スナップショットに対して削除操作を行ってからキャンセルした場合に、スナップショットが依然としてマッピングされているとエラーが発生することがありました。Red Hat OpenStack Platform バージョン 13 以降では、対応しているバックアップドライバーのいずれかで、継続中の復元操作をキャンセルできます。

## 3.4. テナントのバックアップクォータの表示および変更

通常、ダッシュボードを使用して、ボリューム、ボリュームストレージ、スナップショット、テナントが持つことができるその他の操作制限など、テナントのストレージのクォータを変更できます。ただし、ダッシュボードを介してバックアップクォータを変更することはできません。

**cinder quota-update** コマンドを使用してバックアップクォータを変更するには、コマンドラインインターフェイスを使用する必要があります。

### 手順

1. 特定のテナント(TENANT\_ID)のストレージクォータを表示するには、以下を実行します。

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

2. 特定のテナントで作成できるバックアップの最大数(MAXNUM)を更新するには、以下を実行します。

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

3. 特定のテナント内のすべてのバックアップ(MAXGB)の最大サイズを更新するには、以下を実行します。

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

4. 特定のテナントのストレージクォータの使用状況を表示するには、以下を実行します。

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

## 3.5. バックアップからの復元

### 3.5.1. バックアップからのボリュームの復元

以下の手順は、バックアップから新しいボリュームを作成します。

### 手順

1. 使用するボリュームバックアップの ID を検索します。

```
# cinder backup-list
```

ボリューム ID は、復元するボリュームの ID と一致する必要があります。

2. ボリュームのバックアップを復元します。

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

**BACKUP\_ID** は、使用するボリュームバックアップの ID に置き換えます。

3. バックアップが不要になった場合は、削除します。

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

4. バックアップボリュームを特定タイプのボリュームに復元する必要がある場合は、**--volume** を使用して、バックアップを特定ボリュームに復元します。

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



#### 注記

暗号化したバックアップからボリュームを復元する場合は、復元先ボリュームの種類も暗号化する必要があります。

### 3.5.2. Block Storage データベースの損失後のボリュームの復元

通常、Block Storage データベースが失われると、ボリュームバックアップを復元できなくなります。これは、Block Storage データベースにボリュームバックアップサービス **openstack-cinder-backup** で必要なメタデータが含まれているためです。このメタデータは、`backup_service` および `backup_url` の値で設定されます。この値は、ボリュームバックアップの作成後にエクスポートできます。詳細は、「[フルボリュームバックアップの作成](#)」を参照してください。

このメタデータをエクスポートして保存した場合は、新しい Block Storage データベースにインポートして、ボリュームのバックアップを復元できます。



#### 注記

増分バックアップの場合は、増分バックアップのいずれかを復元する前に、エクスポートしたデータをすべてインポートする必要があります。

#### 手順

1. 管理者権限を持つユーザーとして以下を実行します。

```
# cinder backup-import _backup_service_ _backup_url_
```

**backup\_service** および **backup\_url** は、エクスポートしたメタデータからのものです。たとえば、「[フルボリュームバックアップの作成](#)」からエクスポートしたメタデータを使用する場合は、次のコマンドを実行します。

```
# cinder backup-import cinder.backup.drivers.swift eyJzdGF0dXMi...c2l6ZSI6IDF9
```

Property	Value
id	77951e2f-4aff-4365-8c64-f833802eaa43
name	None

- メタデータが Block Storage サービスデータベースにインポートされたら、通常どおりボリュームを復元できます。詳細は、「[バックアップからのボリュームの復元](#)」を参照してください。

### 3.5.3. 復元のキャンセル

#### 手順

- バックアップの復元をキャンセルするには、ステータスを **復元** 以外のものに変更します。**error** ステータスを使用すると、復元の成功の有無に関する混同を最小限にとどめることができます。または、値を **available** に変更できます。

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

#### 注記

バックアップのキャンセルは非同期のアクションです。バックアップドライバーは、バックアップをキャンセルする前にステータスの変更を検出する必要があります。宛先ボリュームで状況が **available** に変更すると、キャンセルが完了します。

#### 注記

この機能は、現在、RBD バックアップでは使用できません。



#### 警告

復元操作を開始した後に取り消すと、宛先ボリュームが実際に復元されたデータ量 (存在する場合) を把握できなくなるため、宛先ボリュームは役に立ちません。

## 3.6. トラブルシューティング

多くの問題を診断するには、サービスが利用可能であることを確認し、ログファイルでエラーメッセージを検索します。

バックアップサービスに関する多くの問題については、2つのシナリオアカウントがあります。

- cinder-backup サービスが起動すると、設定済みのバックエンドに接続し、バックアップのターゲットとしてバックエンドを使用します。この接続に問題があると、サービスが停止する可能性があります。
- バックアップが要求されると、バックアップサービスはボリュームサービスに接続し、要求されたボリュームを割り当てます。バックアップ時間まで接続の問題は認識されません。

いずれの場合も、ログメッセージのログを確認します。

ログの場所とトラブルシューティングの提案に関する詳細は、[Logging, Monitoring and Troubleshooting Guide](#) を参照してください。ログファイルとサービスは、[OpenStack サービスのログファイル セクション](#)に一覧表示されます。

### 3.6.1. サービスの確認

必要なサービスが利用可能であることを確認し、ログメッセージのログを確認します。「[バックアップおよび復元の仕組み](#)」は、キーサービスとその相互作用を示しています。



#### 注記

アクティブ/パッシブ設定のトラブルシューティングを行う場合は、必ずアクティブなコントローラーノード上のログファイルを確認してください。

サービスの状態を確認したら、**cinder-backup.log** を確認します。Block Storage Backup サービスログは、`/var/log/containers/cinder]/cinder-backup.log` にあります。

#### 手順

1. ボリュームで **cinder show** を実行して、ボリュームがホストに保存されているかどうかを確認します。

```
# cinder show
```

2. **cinder service-list** を実行して、実行中のサービスを表示します。

```
# 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. **pcs resource** を実行して、cinder バックアップリソースが実行されているコントローラーを確認します。

```
$ pcs resource
```

```
openstack-cinder-volume (systemd:openstack-cinder-volume): Started controller-0
```

### 3.6.2. トラブルシューティングのヒント

バックアップは非同期です。ブロックストレージバックアップサービスは、API リクエストを受信すると、不正なボリュームリファレンス (**missing**) や、インスタンスに in-use またはアタッチされているボリュームの確認など、少数の静的チェックを実行します。in-use の場合は、**--force** を使用する必要があります。



#### 注記

**--force** を使用すると、I/O が静止せず、ボリュームイメージが破損する可能性があります。

API が要求を受け入れると、バックアップがバックグラウンドで発生します。バックアップが失敗したか、または失敗することが近い場合でも、CLI は即座に返します。バックアップのステータスは、cinder バックアップ API を使用してクエリーできます。エラーが発生した場合は、ログで原因を確認します。

### 3.6.3. Pacemaker

デフォルトでは、Block Storage のバックアップサービスは、Pacemaker を使用して active/passive モードでデプロイされます。Pacemaker: Pacemaker は、仮想 IP アドレス、コンテナー、サービス、その他の機能をクラスター内のリソースとして設定することにより、定義済みの OpenStack クラスターリソースが確実に実行され、利用できるようにします。クラスター内のサービスノードまたは全ノードが停止した場合には、Pacemaker はリソースの再起動、クラスターからのノードの削除、ノードの再起動を実行することができます。これらのサービスの大半に対する要求は、HAProxy 経由で行われます。

トラブルシューティングに Pacemaker を使用方法については、**High Availability Deployment and Usage** の [Managing high availability services with Pacemaker](#) を参照してください。



## 付録A 高度な BLOCK STORAGE のバックアップ設定オプション

director でデプロイされるインストール以前は、cinder.conf ファイルを使用して Block Storage サービスおよびバックアップサービスを設定していました。cinder.conf の値に同等の Heat テンプレートがない場合は、カスタム環境を使用して director から値を渡すことができます。この値は、カスタム環境ファイル(**cinder-backup-settings.yaml**など)の **parameter\_defaults** セクションの **ExtraConfig** セクションに追加されます。

**ExtraConfig** は、追加の hiera 設定を全ノードのクラスターに挿入する方法を提供します。この設定は、**ExtraConfig** など、専用のバックアップノードに含まれます。**ExtraConfig** の代わりに **ControllerExtraConfig** を使用した場合、これらの設定はコントローラーノードにのみインストールされ、専用のバックアップノードにはインストールされません。

cinder.conf ファイルの **DEFAULT** セクションで使用される設定は、**DEFAULT/[cinder.conf setting]** に置き換えることができます。以下の例は、**ExtraConfig** およびエントリーが YAML ファイルにどのように表示されるかを示しています。

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

以下のオプションは、バックアップ関連のオプションの例です。

表A.1 ブロックストレージバックアップサービスの設定オプション

オプション	タイプ	デフォルト値	説明
backup_service_inithost_offload	任意	True	バックアップサービスの起動時に、バックアップの削除が保留中であることをオフロードします。false の場合、バックアップサービスは、保留中のバックアップがすべて削除されるまでダウンしたままになります。
use_multipath_for_image_xfer	任意	False	バックアップおよび復元の手順中にマルチパスを使用してボリュームを接続できる場合は、接続します。これは、イメージからのボリュームの作成、一般的なコールド移行、その他の操作など、すべての cinder の接続操作に影響します。
num_volume_device_scan_tries	任意	3	接続時にボリュームを検出するためにターゲットを再スキャンする回数の上限です。

オプション	タイプ	デフォルト値	説明
backup_workers	任意	1	実行するバックアッププロセスの数。複数の同時バックアップまたは圧縮で復元を実行すると、パフォーマンスが大幅に向上します。
backup_native_threads_pool_size	任意	60	バックアップ用のネイティブスレッドプールのサイズ。ほとんどのバックアップドライバーは、これに大きく依存しています。このオプションに依存しない特定のドライバーでは、の値を減らすことができます。
backup_share	必須		<b>HOST:EXPORT_PATH_</b> に設定します。
backup_container	任意	なし	(文字列) バックアップに使用するカスタムディレクトリー。
backup_enable_progress_timer	任意	True	ボリュームをバックエンドストレージにバックアップする際に、Ceilometer に定期的な進捗通知を送信するタイマーを有効または無効にします。
backup_mount_options	任意		backup_share で指定した NFS エクスポートをマウントする際に指定するオプションのコンマ区切りリスト。
backup_mount_point_base	任意	\$state_path/backup_mount	(文字列) NFS 共有のマウントポイントを含むベースディレクトリー。
backup_compression_algorithm	任意	zlib	バックアップデータをリポジトリーに送信するときに使用する圧縮アルゴリズム。有効な値は、 <b>zlib</b> 、 <b>bz2</b> 、および <b>None</b> です。

オプション	タイプ	デフォルト値	説明
backup_file_size	任意	1999994880	これよりも大きな cinder ボリュームからのデータは、バックアップリポジトリに複数のファイルとして保存されます。このオプションは、backup_sha_block_size_bytes の倍数にする必要があります。
backup_sha_block_size_bytes	任意	32768	デジタル署名の計算用の cinder ボリュームブロックのサイズ