



# Red Hat OpenStack Platform

## 11

# Google Cloud バックアップガイド

---

Google Cloud Storage を使用するための OpenStack Block Storage バックアップの設定

OpenStack Team



## Google Cloud Storage を使用するための OpenStack Block Storage バックアップの設定

OpenStack Team  
rhos-docs@redhat.com

## 法律上の通知

Copyright © 2017 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, 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 Software Collections 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.

## 概要

本ガイドでは、Google Cloud Storage をバックエンドとして使用するために OpenStack Block Storage のバックアップサービスをデプロイする方法について説明します。本書に記載する方法は、オーバークラウド固有のデプロイメント手順です。本リリースでは、Google Cloud Storage とのバックアップの統合は、テクノロジープレビューとして提供しており、実稼働環境でデプロイすべきではありません。テクノロジープレビューに関する詳しい情報は、以下のリンクを参照してください。

---

## 目次

1. はじめに .....	2
2. プロセスの説明 .....	2
3. GCS 認証情報ファイルの作成とダウンロード .....	3
4. 環境ファイルの作成 .....	5
5. オーバークラウドの再デプロイ .....	8

## 1. はじめに

Red Hat OpenStack Platform director は、完全な OpenStack 環境をインストールおよび管理するためのツールセットで、主に OpenStack プロジェクトの TripleO (**OpenStack-on-OpenStack**) をベースとしています。director の主な目的は、手動での設定を最低限に抑えて、機能的なエンタープライズレベルの OpenStack デプロイメントを完全にオーケストレーションすることです。director は、個別の OpenStack コンポーネントの手動設定に内在する数多くの問題に対処するのに役立ちます。

director が作成して最終的に構築される OpenStack デプロイメントは、**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コンポーネントがすべて含まれます。本書は、オーバークラウドの Block Storage サービスにカスタムのバックエンドをデプロイする方法を説明します。

Red Hat OpenStack Platform は、ブロックストレージバックエンドやデータ処理などの機能向けにサードパーティーのサービス、デバイス、およびアプリケーションを複数サポートしています。今回のリリースでは、Block Storage サービスが Google Cloud をバックアップストレージサービスとして使用するよう設定することができます。本リリースでは、Google Cloud の統合は **テクノロジープレビュー** 機能として提供しています。



### 重要

テクノロジープレビュー機能は、Red Hat では完全にサポートしていません。本ガイドで記載するデプロイメントシナリオはテスト目的のみで使用し、実稼働環境にはデプロイすべきではありません。

テクノロジープレビュー機能に関する詳しい情報は、「[対象範囲の詳細](#)」を参照してください。

本ガイドには、オーバークラウドデプロイメント上の Block Storage サービスがボリュームを Google Cloud Storage にバックアップするように設定するテストシナリオを記載しています。このテストシナリオは、以下の要件を満たす必要があります。

- ※ オーバークラウドは、『[Red Hat OpenStack Platform 11 director のインストールと使用方法](#)』の手順に従って、director を使用してデプロイ済みであること。
- ※ 管理者特権のアカウントのユーザー名とパスワードがあること。オーバークラウドをデプロイするために作成されたのと同じアカウントを使用することができます。『[Red Hat OpenStack Platform 11 director のインストールと使用方法](#)』では、**stack** という名前のユーザーはこの目的で作成されます。
- ※ Block Storage サービスが、コントローラーノードにインストールされていること。HA デプロイメントの場合には、全コントローラーノードにインストールされていること。
- ※ Google Cloud Platform にアクセスできる Google アカウントがあること。このアカウントは、Block Storage サービスが Google Cloud にアクセスして、バックアップを保管するために使用します。

## 2. プロセスの説明

Block Storage サービスが Google Cloud をバックアップサービスとして使用するよう設定する手順は、以下のステップで構成されます。

1. Google アカウントのサービスアカウントの認証情報の作成およびダウンロード（「[GCS 認証情報ファイルの作成とダウンロード](#)」）。

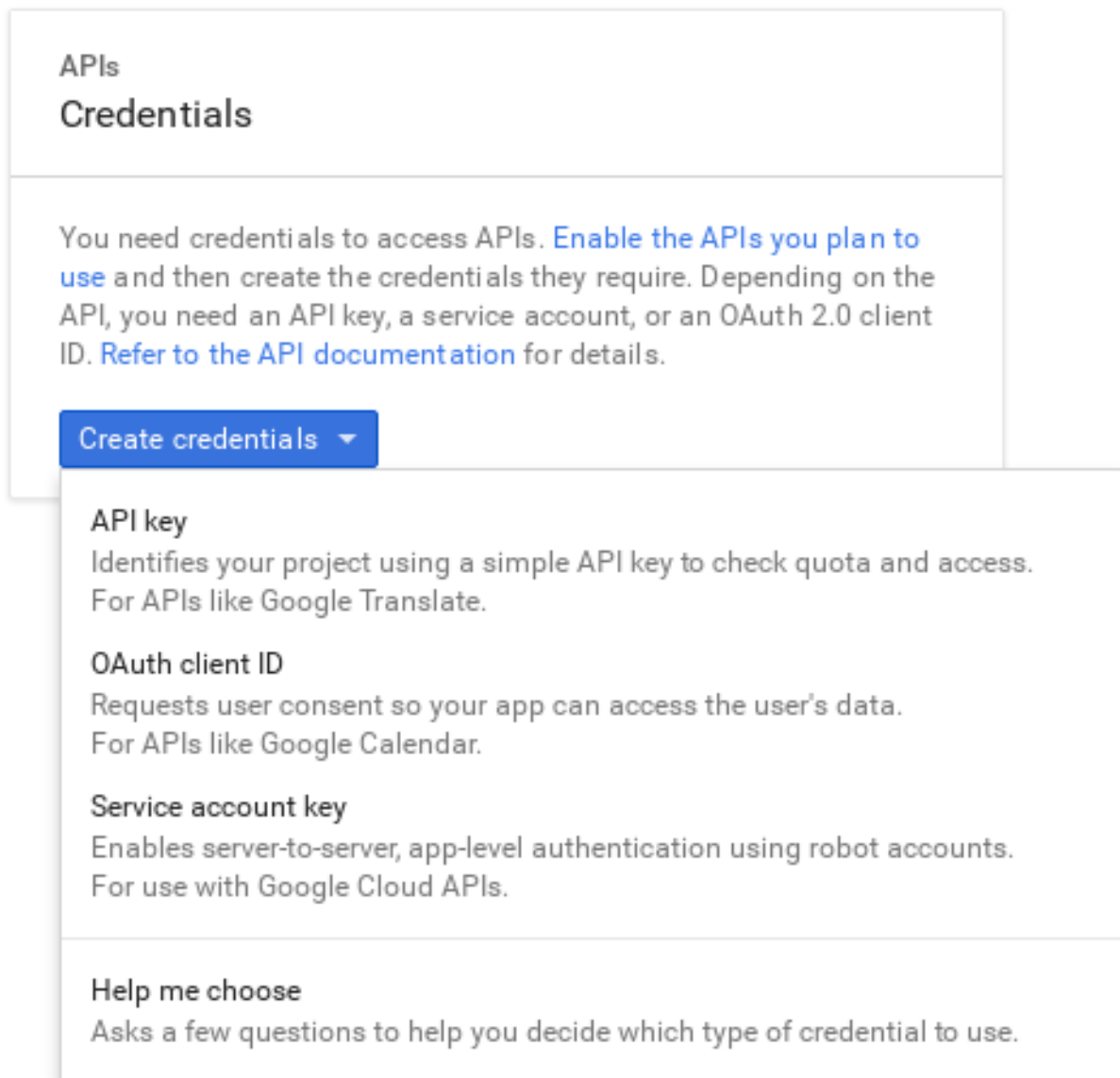
2. 必要な Block Storage の設定を定義する環境ファイルの作成 (「[環境ファイルの作成](#)」)。この環境ファイルは、前のステップで作成したサービスアカウントの認証情報も使用します。
3. 作成した環境ファイルを使用したオーバークラウドの再デプロイ (「[オーバークラウドの再デプロイ](#)」)。

以下のセクションでは、これらの各手順について詳しく説明します。

### 3. GCS 認証情報ファイルの作成とダウンロード

Block Storage サービスがバックアップのために Google Cloud にアクセス/使用するには、Google の認証情報を必要とします。以下の手順にしたがって **service account key** を作成すると、これらの認証情報を Block Storage に提供することができます。

1. Google アカウントを使用して、Google Developer Console (<http://console.developers.google.com>) にログインします。
2. **認証情報** タブをクリックします。**認証情報を作成** ドロップダウンリストから **サービス アカウント キー** を選択します。



3. 次の画面 (**サービス アカウント キーの作成**) で、**サービス アカウント** ドロップダウンリストから Block Storage サービスが使用する必要のあるサービスアカウントを選択します。

## Credentials



### Create service account key

#### Service account

Compute Engine default service account

#### Key type

Downloads a file that contains the private key. Store the file securely because this key can't be recovered if lost.

JSON

Recommended

P12

For backward compatibility with code using the P12 format

Create

Cancel

4. 同じ画面で、**キーのタイプ** のセクションから **JSON** を選択して、**作成** をクリックします。  
ブラウザにより、キーがデフォルトのダウンロード先にダウンロードされます。

### New private key

Cloud Backup-0c642522b844.json has been saved on your computer. This is the only copy of the key, so store it securely.

Close

5. ダウンロードしたファイルを開いて、**project\_id** パラメーターの値をメモしておきます。

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

`/etc/cinder/Cloud-Backup.json` キーは、後ほど「[環境ファイルの作成](#)」で使用します (具体的には、**project\_id** の値とファイルへの絶対パスを使用します)。



6. キーファイルを任意のコントローラーノード上の `/etc/cinder/` にコピーします。その場所から、以下のコマンドを実行して、キーファイルのユーザー、グループ、パーミッションが `/etc/cinder/cinder.conf` と一致するように変更します。この操作により、Block Storage サービスが確実にこのキーファイルの情報を使用することができるようになります。

```
# cp Cloud-Backup.json /etc/cinder/
# chown cinder:cinder /etc/cinder/Cloud-Backup.json
# chmod 0600 /etc/cinder/Cloud-Backup.json
```

7. 各コントローラーノードの同じ場所 (`/etc/cinder/Cloud-Backup.json`) にキーファイルをコピーします。以下のように `rsync -a` コマンドを実行して、パーミッションと所有者の設定が保持されるようにします。

```
# rsync -a /etc/cinder/Cloud-Backup.json
root@CONTROLLERHOST:/etc/cinder/
```

`CONTROLLERHOST` の箇所は、コピー先のコントローラーのホスト名に置き換えます。

## 4. 環境ファイルの作成

環境ファイルには、Block Storage サービスに適用する必要がある設定が記載されています。この場合には、Block Storage サービスは Google Cloud にボリュームのバックアップを保管するように設定されます。環境ファイルに関する詳しい情報は、『[Red Hat OpenStack Platform 11 director のインストールと使用方法](#)』を参照してください。

環境ファイルでは、各設定を以下のように定義します。

### 入力形式

```
SECT/PARAM: # ①
value: CONFIG # ②
```

1

Block Storage の設定はすべて、Block Storage サービスをホストしているノードの `/etc/cinder/cinder.conf` で設定されます。このファイルは、複数のセクションに分かれており、異なる設定を管理しやすいようになっています。`PARAM` は適用する設定に、`SECT` はその設定が属するセクションに置き換えます。

2

`CONFIG` は、`PARAM` に対して設定する値に置き換えます。

本ガイドでは、すべてのパラメーターが `DEFAULT` セクションで宣言されます。以下の表には、Google Cloud Storage (GCS) をバックアップサービスとして設定するのに必要な各設定についての説明をまとめています。

1. Google Cloud バックアップの設定

PARAM	デフォルト	CONFIG の説明
backup_driver	cinder.backup.drivers.swift	Block Storage サーバーが使用する必要のあるバックアップドライバー。Google Cloud Storage には、 <b>cinder.backup.drivers.google</b> を使用します。
backup_gcs_credential_file		<a href="#">「GCS 認証情報ファイルの作成とダウンロード」</a> のステップで作成済みのサービスアカウントのキーファイルへの絶対パス
backup_gcs_bucket		使用する GCS バケット (またはオブジェクトストレージリポジトリ)。これはすでに存在している場合とそうでない場合があります。存在しないバケットを指定する場合、Google Cloud Storage のバックアップドライバーは、ここで指定した名前を使用してバケットを作成します。詳しくは、 <a href="#">「Buckets」</a> および <a href="#">「Bucket name requirements」</a> を参照してください。
backup_gcs_bucket_location	US	GCS バケットの場所。この値は、 <b>backup_gcs_bucket</b> に存在しないバケットを指定した場合にのみ使用されます。その場合には、Google Cloud Storage のバックアップドライバーが GCS バケットの場所としてこの値を指定します。  詳しくは、 <a href="#">「Bucket Locations」</a> を参照してください。
backup_gcs_project_id		使用するサービスアカウントのプロジェクト ID。 <a href="#">「GCS 認証情報ファイルの作成とダウンロード」</a> のステップで、サービスアカウントキーの <b>project_id</b> を確認してメモした値です。

backup_gcs_object_size	52428800	GCS バックアップオブジェクトのサイズ (バイト単位)
backup_gcs_block_size	32768	差分バックアップでトラッキングされる変更のサイズ (バイト単位)。この値は、 <b>backup_gcs_object_size</b> 値の倍数です。
backup_gcs_user_agent	gcsinder	GCS API の HTTP ユーザーエージェントの文字列
backup_gcs_reader_chunk_size	2097152	GCS オブジェクトは、このサイズ (バイト単位) のチャンクでダウンロードされます。
backup_gcs_writer_chunk_size	2097152	GCS オブジェクトは、このサイズ (バイト単位) の複数のチャンクでアップロードされます。複数のチャンクではなく、単一のチャンクとしてファイルをアップロードするには、 <b>-1</b> の値を指定してください。
backup_gcs_num_retries	3	再試行回数
backup_gcs_storage_class	NEARLINE	GCS バケットのストレージクラス。この値は、 <b>backup_gcs_bucket</b> に存在しないバケットを指定した場合にのみ使用されます。その場合には、Google Cloud Storage のバックアップドライバーが GCS バケットのストレージクラスとしてこの値を指定します。詳しくは、「 <a href="#">Storage Classes</a> 」を参照してください。
backup_gcs_retry_error_codes	429	GCS エラーコード一覧

```
backup_gcs_enable_progress_time    True
r
```

ボリュームのバックアップ中に Telemetry サービス (**ceilometer**) に定期的な進捗状況通知を送信するためのタイマーを有効化または無効化するためのブール値。デフォルトでは、この設定は有効化されています (**True**)。

## 警告

Google Cloud Storage は、新規バケットを作成すると、選択したストレージクラス (**backup\_gcs\_storage\_class**) に基づいて課金します。デフォルトの **NEARLINE** クラスは、バックアップサービスに適しています。

なお、バケットは、一旦作成したら場所やクラスを変更することはできません。詳しくは、「[Managing a bucket's storage class or location](#)」を参照してください。

以下のサンプルには、GCS をバックアップサービスとして設定するための環境ファイルの一般的な内容を示しています。

### /home/stack/templates/gcs-backup.yaml

```
parameter_defaults:
  ControllerExtraConfig: # 1
    cinder::config::cinder_config:
      DEFAULT/backup_driver
        value: cinder.backup.drivers.google
      DEFAULT/backup_gcs_credential_file
        value: /etc/cinder/Cloud-Backup.json
      DEFAULT/backup_gcs_bucket
        value: mycinderbucket
      DEFAULT/backup_gcs_project_id
        value: cloud-backup-1370
      DEFAULT/backup_gcs_user_agent
        value: myuseragent
```

1

**ControllerExtraConfig** は、全コントローラーノードに適用されるカスタムを設定を定義します。**cinder::config::cinder\_config** クラスは、この設定が Block Storage (**cinder**) サービスに適用される必要のあることを意味します。このため、バックエンドの設定は最終的に、各コントローラーノードの `/etc/cinder/cinder.conf` ファイルに指定されることとなります。

環境ファイルを作成した後は、「[オーバークラウドの再デプロイ](#)」でオーバークラウドに設定をデプロイする手順を参照してください。

## 5. オーバークラウドの再デプロイ

`/home/stack/templates/` に [environment file](#) ファイルを作成したら、`stack` ユーザーとしてログインし、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/gcs-backup.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。

詳しくは、「[オーバークラウドのスケーリング](#)」および「[オーバークラウド環境の変更](#)」を参照してください。