



Red Hat OpenStack Platform 13

director のアンダークラウドのバックアップとリストア

director のアンダークラウドのバックアップとリストア

Red Hat OpenStack Platform 13 director のアンダークラウドのバックアップとリストア

director のアンダークラウドのバックアップとリストア

OpenStack Team
rhos-docs@redhat.com

法律上の通知

Copyright © 2023 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.

概要

本ガイドでは、Red Hat OpenStack Platform director のアンダークラウドのバックアップとリストアの方法について説明します。

目次

第1章 アンダークラウドのバックアップ	3
1.1. バックアップに関する留意事項	3
1.2. アンダークラウドノードの高可用性	3
1.3. ベアメタルアンダークラウドのバックアップ	3
1.4. 作成完了したバックアップの検証	5
パート I. アンダークラウドのリストア	6
第2章 アンダークラウドのリストア	7
第3章 オーバークラウドノード用イメージのリストア	12
第4章 完了したリストアの検証	13
4.1. IDENTITY サービス (KEYSTONE) の動作の確認	13

第1章 アンダークラウドのバックアップ

本ガイドでは、Red Hat OpenStack Platform director で使用するアンダークラウドのバックアップ方法を説明します。アンダークラウドとは、通常 OpenStack 環境のデプロイメントおよび管理に使用する単一の物理ノードのことを指します(ただし、仮想マシンで director を実行する、2 ノードタイプの Pacemaker クラスタを使用した高可用性オプションも存在します)。

1.1. バックアップに関する留意事項

データ損失やシステムのダウンタイムを最小限に抑えるため、強力なバックアップおよびリカバリーポリシーを策定します。バックアップストラテジーを決定するにあたっては、以下の質問事項への回答を明確にしておく必要があります。

- **データ損失からどの程度迅速に復旧する必要がありますか。** データ損失が一切許容されない場合には、デプロイメントストラテジーとして、バックアップの使用に加えて、高可用性に焦点を当てるべきです。物理バックアップメディアを取得する際にかかる時間(例: オフサイトの場所を利用している場合はそのサイトからのメディア)やリストアの操作に利用可能なテープドライブがいくつあるかを考慮する必要があります。
- **保管が必要なバックアップの数はいくつですか。** データの保管年数に影響を与える法的かつ規制上の要件を考慮する必要があります。
- **バックアップはオフサイトに保管する必要がありますか。** バックアップメディアをオフサイトに保管すると、物理的な場所に降りかかる災害のリスクを軽減するのに役立ちます。
- **バックアップをテストする頻度はどの程度ですか。** 強固なバックアップストラテジーには、バックアップデータを定期的に戻すテストが含まれます。これは、正しいデータが依然としてバックアップされており、バックアップやリストアプロセス中にデータの破損が発生していないことを検証するのに役立ちます。これらのテストは、実際の災害復旧の条件下で実行することを想定すべきです。
- **バックアップの対象は何ですか。** 以下の項では、コンポーネントのデータベースとファイルシステムのバックアップのほか、バックアップの復旧についても説明します。

1.2. アンダークラウドノードの高可用性

Red Hat は、アンダークラウドノードの高可用性(HA)オプションに特定の要件を規定していないので、希望に応じて自由に検討することができます。たとえば、Red Hat Virtualization (RHV) 内の高可用性の仮想マシンとしてアンダークラウドノードを実行する設定を検討してください。また、必要なサービスを HA で提供する Pacemaker をインストールした物理ノードを使用することも検討することができます。

アンダークラウドノードの高可用性について検討する場合は、お使いの環境に最も有効であると判断したソリューションのドキュメントやグッドプラクティスを参照してください。

1.3. ベアメタルアンダークラウドのバックアップ

完全なアンダークラウドのバックアップには、以下のデータベースおよびファイルが含まれます。

- アンダークラウドノード上の MariaDB データベース
- (データベースを正確に戻すできるように) アンダークラウド上の MariaDB 設定ファイル
- 設定データ: `/etc`

- ログデータ: `/var/log`
- イメージデータ: `/var/lib/glance`
- 証明書生成データ (SSL を使用している場合): `/var/lib/certmonger`
- コンテナイメージデータ: `/var/lib/docker`、`/var/lib/registry`
- swift の全データ: `/srv/node`
- stack ユーザーのホームディレクトリー内の全データ: `/home/stack`



注記

バックアッププロセスを実行する前に、アンダークラウドに利用可能なディスク容量が十分であることを確認します。アーカイブファイルは、少なくとも 3.5 GB となることが予想され、それ以上になる可能性があります。

手順

1. アンダークラウドに **root** ユーザーとしてログインします。
2. データベースのバックアップを作成します。

```
[root@director ~]# mysqldump --opt --all-databases > /root/undercloud-all-databases.sql
```

3. **backup** ディレクトリーを作成して、そのディレクトリーを所有するユーザーを **stack** ユーザーに変更します。

```
[root@director ~]# mkdir /backup
[root@director ~]# chown stack: /backup
```

このディレクトリーを使用して、アンダークラウドのデータベースおよびファイルシステムを含むアーカイブを保存します。

4. **バックアップ** ディレクトリーに移動します。

```
[root@director ~]# cd /backup
```

5. データベースのバックアップと設定ファイルをアーカイブします。

```
[root@director ~]# tar --xattrs --xattrs-include='*.*' --ignore-failed-read -cf \
  undercloud-backup-$(date +%F).tar \
  /root/undercloud-all-databases.sql \
  /etc \
  /var/log \
  /var/lib/glance \
  /var/lib/certmonger \
  /var/lib/docker \
  /var/lib/registry \
  /srv/node \
  /root \
  /home/stack
```


- **--ignore-failed-read** オプションを指定すると、アンダークラウドに適用されないディレクトリーはスキップされます。
- **--xattrs** および **--xattrs-include=*.*** オプションには、Object Storage (swift) および SELinux のメタデータを保存するために必要な拡張属性が含まれます。

これで、**undercloud-backup-<date>.tar.gz** という名前のファイルが作成されます。ここで、**<date>** はシステムの日付けです。この **tar** ファイルをセキュアな場所にコピーします。

1.4. 作成完了したバックアップの検証

リストアッププロセスを実行/検証することでバックアッププロセスが正常に完了したことを検証できます。バックアップからのリストアの詳しい情報は、次の項を参照してください。

パート I. アンダークラウドのリストア

本項では、Red Hat OpenStack Platform director で使用するアンダークラウドのリストア方法を説明します。



注記

このプロセスでは、OpenStack Platform director のバックアップのデータをアンダークラウドの新規インストールにリストアする手順を説明します。これにより、リストアされたアンダークラウドでは最新のパッケージが使用されます。

第2章 アンダークラウドのリストア

以下の復元の手順は、お使いのアンダークラウドノードでエラーが発生して、回復不能な状態であることを前提としています。この手順では、新規インストール環境でデータベースおよびクリティカルなファイルシステムの復元を行う必要があります。以下が前提条件です。

- Red Hat Enterprise Linux 7 の最新版を再インストール済みであること
- ハードウェアレイアウトが同じであること
- マシンのホスト名とアンダークラウドの設定が同じであること
- バックアップアーカイブが **root** ディレクトリーにコピー済みであること

手順

1. お使いのアンダークラウドに **root** ユーザーとしてログインします。
2. コンテンツ配信ネットワークにシステムを登録します。プロンプトが表示されたら、カスタマーポータルของผู้ーザー名とパスワードを入力します。

```
[root@director ~]# subscription-manager register
```

3. Red Hat OpenStack Platform のエンタイトルメントをアタッチします。

```
[root@director ~]# subscription-manager attach --pool=Valid-Pool-Number-123456
```

4. デフォルトのリポジトリーをすべて無効にしてから、必要な Red Hat Enterprise Linux リポジトリーを有効にします。

```
[root@director ~]# subscription-manager repos --disable=*  
[root@director ~]# subscription-manager repos --enable=rhel-7-server-rpms --enable=rhel-7-  
server-extras-rpms --enable=rhel-7-server-rh-common-rpms --enable=rhel-ha-for-rhel-7-  
server-rpms --enable=rhel-7-server-openstack-13-rpms
```

5. システムで更新を実行して、ベースシステムパッケージを最新の状態にします。

```
[root@director ~]# yum update -y  
[root@director ~]# reboot
```

6. アンダークラウドの時刻が同期されていることを確認します。以下に例を示します。

```
[root@director ~]# yum install -y ntp  
[root@director ~]# systemctl start ntpd  
[root@director ~]# systemctl enable ntpd  
[root@director ~]# ntpdate pool.ntp.org  
[root@director ~]# systemctl restart ntpd
```

7. アンダークラウドのバックアップアーカイブをアンダークラウドの **root** ディレクトリーにコピーします。これ以降のステップでは、ファイル名に **undercloud-backup-\$TIMESTAMP.tar** を使用しています。ここで、\$TIMESTAMP はアーカイブのタイムスタンプの Bash 変数です。
8. データベースサーバーとクライアントツールをインストールします。

```
[root@director ~]# yum install -y mariadb mariadb-server
```

9. データベースを起動します。

```
[root@director ~]# systemctl start mariadb  
[root@director ~]# systemctl enable mariadb
```

10. データベースのバックアップのサイズに対応するように、許可されるパケット数を増やします。

```
[root@director ~]# mysql -uroot -e"set global max_allowed_packet = 1073741824;"
```

11. アーカイブからデータベースおよびデータベース設定を抽出します。

```
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar  
etc/my.cnf.d/*server*.cnf  
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar root/undercloud-all-  
databases.sql
```

12. データベースのバックアップをリストアします。

```
[root@director ~]# mysql -u root < /root/undercloud-all-databases.sql
```

13. root 設定ファイルの一時バージョンを抽出します。

```
[root@director ~]# tar -xvf undercloud-backup-$TIMESTAMP.tar root/.my.cnf
```

14. データベースの古い root パスワードを取得します。

```
[root@director ~]# OLDPASSWORD=$(sudo cat root/.my.cnf | grep -m1 password | cut -d=''  
-f2 | tr -d '"')
```

15. データベースの root パスワードをリセットします。

```
[root@director ~]# mysqladmin -u root password "$OLDPASSWORD"
```

16. root 設定ファイルを一時ディレクトリーから **root** ディレクトリーに移動します。

```
[root@director ~]# mv ~/root/.my.cnf ~/.  
[root@director ~]# rmdir ~/root
```

17. 古いユーザー権限の一覧を取得します。

```
[root@director ~]# mysql -e 'select host, user, password from mysql.user;'
```

18. リストされた各ホストの古いユーザー権限を削除します。以下に例を示します。

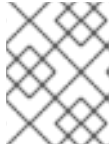
```
[root@director ~]# HOST="192.0.2.1"  
[root@director ~]# USERS=$(mysql -Nse "select user from mysql.user WHERE user !=  
\'root\' and host = \'$HOST\';" | uniq | xargs)  
[root@director ~]# for USER in $USERS ; do mysql -e "drop user \'$USER\'@\'$HOST\'" ||
```

```

true ;done
[root@director ~]# for USER in $USERS ; do mysql -e "drop user $USER" || true ;done
[root@director ~]# mysql -e 'flush privileges'

```

ホスト IP および任意のホスト ("%") からアクセスするすべてのユーザーに対して、この手順を実施します。



注記

HOST パラメーターの IP アドレスは、コントロールプレーン内のアンダークラウドの IP アドレスです。

19. データベースを再起動します。

```

[root@director ~]# systemctl restart mariadb

```

20. **stack** ユーザーを作成します。

```

[root@director ~]# useradd stack

```

21. ユーザーのパスワードを設定します。

```

[root@director ~]# passwd stack

```

22. **sudo** を使用する場合にパスワードを要求されないようにします。

```

[root@director ~]# echo "stack ALL=(root) NOPASSWD:ALL" | tee -a /etc/sudoers.d/stack
[root@director ~]# chmod 0440 /etc/sudoers.d/stack

```

23. **stack** ユーザーのホームディレクトリーをリストアします。

```

# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar home/stack

```

24. **polycoreutils-python** パッケージをインストールします。

```

[root@director ~]# yum -y install polycoreutils-python

```

25. **openstack-glance** パッケージをインストールして、そのデータおよびファイルパーミッションをリストアします。

```

[root@director ~]# yum install -y openstack-glance
[root@director ~]# tar --xattrs --xattrs-include='*.*' -xvC / -f undercloud-backup-
$TIMESTAMP.tar var/lib/glance/images
[root@director ~]# chown -R glance: /var/lib/glance/images
[root@director ~]# restorecon -R /var/lib/glance/images

```

26. **openstack-swift** パッケージをインストールして、そのデータおよびファイルパーミッションをリストアします。

```

[root@director ~]# yum install -y openstack-swift
[root@director ~]# tar --xattrs --xattrs-include='*.*' -xvC / -f undercloud-backup-
$TIMESTAMP.tar srv/node

```

```
[root@director ~]# chown -R swift: /srv/node
[root@director ~]# restorecon -R /srv/node
```

27. **openstack-keystone** パッケージをインストールして、その設定データをリストアします。

```
[root@director ~]# yum -y install openstack-keystone
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/keystone
[root@director ~]# restorecon -R /etc/keystone
```

28. **openstack-heat** をインストールして、設定をリストアします。

```
[root@director ~]# yum install -y openstack-heat*
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/heat
[root@director ~]# restorecon -R /etc/heat
```

29. Puppet をインストールして、その設定データをリストアします。

```
[root@director ~]# yum install -y puppet hiera
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/puppet/hieradata/
```

30. アンダークラウドで SSL を使用する場合は、CA 証明書を更新します。アンダークラウドの設定に応じて、ユーザー提供の証明書の手順または自動生成された証明書の手順のいずれかを使用します。

- アンダークラウドがユーザー提供の証明書で設定されている場合は、次の手順を実行します。

- a. 証明書を抽出します。

```
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/instack-certs/undercloud.pem
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/ca-trust/source/anchors/*
```

- b. SELinux コンテキストを復元し、ファイルシステムのラベル付けを管理します。

```
[root@director ~]# restorecon -R /etc/pki
[root@director ~]# semanage fcontext -a -t etc_t "/etc/pki/instack-certs(/.*)?"
[root@director ~]# restorecon -R /etc/pki/instack-certs
```

- c. 証明書を更新します。

```
[root@director ~]# update-ca-trust extract
```

- **certmonger** を使用してアンダークラウドの証明書を自動生成する場合は、以下の手順を実行します。

- a. 証明書、CA 証明書、および certmonger ファイルを抽出します。

```
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar var/lib/certmonger/*
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/tls/*
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/ca-trust/source/anchors/*
```

- b. SELinux コンテキストを復元します。

```
[root@director ~]# restorecon -R /etc/pki
[root@director ~]# restorecon -R /var/lib/certmonger
```

- c. `/var/lib/certmonger/lock` ファイルを削除します。

```
[root@director ~]# rm -f /var/lib/certmonger/lock
```

31. **stack** ユーザーに切り替えます。

```
[root@director ~]# su - stack
[stack@director ~]$
```

32. **python-tripleoclient** パッケージをインストールします。

```
$ sudo yum install -y python-tripleoclient
```

33. アンダークラウドのインストールコマンドを実行します。このコマンドは、**stack** ユーザーのホームディレクトリーから実行するようにしてください。

```
[stack@director ~]$ openstack undercloud install
```

インストールが完了すると、アンダークラウドは、オーバークラウドへの接続を自動的にリストアします。ノードは、保留中のタスクに対して、OpenStack Orchestration (heat) のポーリングを続けます。

34. root ユーザーに変更します。

```
$ sudo su -
```

35. コンテナのデータをバックアップのコンテンツと同期します。

```
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar var/lib/docker/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar var/lib/registry/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/docker/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/docker-distribution/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/sysconfig/docker*
[root@director ~]$ systemctl restart docker docker-distribution
```

第3章 オーバークラウドノード用イメージのリストア

新しいオーバークラウドノードのプロビジョニング用に、director には最新のディスクイメージが必要です。以下の手順に従って、これらのイメージをリストアします。

手順

1. source コマンドで **stackrc** ファイルを読み込み、director のコマンドラインツールを有効にします。

```
[stack@director ~]$ source ~/stackrc
```

2. **rhosp-director-images** および **rhosp-director-images-ipa** パッケージをインストールします。

```
(undercloud) [stack@director ~]$ sudo yum install rhosp-director-images rhosp-director-images-ipa
```

3. イメージのアーカイブを、**stack** ユーザーのホーム下の **images** ディレクトリー (**/home/stack/images**) に展開します。

```
(undercloud) [stack@director ~]$ cd ~/images
(undercloud) [stack@director images]$ for i in /usr/share/rhosp-director-images/overcloud-full-latest-13.0.tar /usr/share/rhosp-director-images/ironic-python-agent-latest-13.0.tar; do tar -xvf $i; done
```

4. これらのイメージを director にインポートします。

```
(undercloud) [stack@director images]$ cd ~/images
(undercloud) [stack@director images]$ openstack overcloud image upload --image-path /home/stack/images/
```

5. 新しいイメージを使用するように、環境内のノードを設定します。

```
(undercloud) [stack@director images]$ for NODE in $(openstack baremetal node list -c UUID -f value) ; do openstack overcloud node configure $NODE ; done
```


第4章 完了したリストアの検証

以下のコマンドを使用して、新しくリストアした環境のヘルスチェックを実行します。

4.1. IDENTITY サービス (KEYSTONE) の動作の確認

このステップでは、ユーザーの一覧をクエリーで取得して、Identity サービスの動作を検証します。

```
# source stackrc
# openstack user list
```

コントローラーから実行する場合は、このコマンドの出力には、この環境で作成されたユーザーの一覧が含まれているはずですが、このアクションでは、keystone が実行中でユーザーの要求を正常に認証していることが分かります。以下に例を示します。

```
# openstack user list
+-----+-----+-----+-----+
|      id      | name | enabled |   email   |
+-----+-----+-----+-----+
| 9e47bb53bb40453094e32eccce996828 | admin | True | root@localhost |
| 9fe2466f88cc4fa0ba69e59b47898829 | ceilometer | True | ceilometer@localhost |
| 7a40d944e55d422fa4e85daf47e47c42 | cinder | True | cinder@localhost |
| 3d2ed97538064f258f67c98d1912132e | demo | True |           |
| 756e73a5115d4e9a947d8aadc6f5ac22 | glance | True | glance@localhost |
| f0d1fcee8f9b4da39556b78b72fdafb1 | neutron | True | neutron@localhost |
| e9025f3faeee4d6bb7a057523576ea19 | nova | True | nova@localhost |
| 65c60b1278a0498980b2dc46c7dcf4b7 | swift | True | swift@localhost |
+-----+-----+-----+-----+
```