



Red Hat OpenStack Platform 13

アンダークラウドとコントロールプレーンのバックアップおよびリストア

更新/アップグレード時に、アンダークラウドとオーバークラウドのコントロールプレーンをバックアップおよびリストアする手順

Red Hat OpenStack Platform 13 アンダークラウドとコントロールプレーンのバックアップおよびリストア

更新/アップグレード時に、アンダークラウドとオーバークラウドのコントロールプレーンをバックアップおよびリストアする手順

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、アンダークラウドおよびオーバークラウドのコントロールプレーンノードに Relax-and-Recover (ReaR) をインストール/設定する方法、更新/アップグレードの前にアンダークラウドおよびコントロールプレーンノードをバックアップする方法、および更新またはアップグレードを実施中にエラーが発生した場合にアンダークラウドおよびコントロールプレーンノードをリストアする方法について説明します。

目次

第1章 アンダークラウドとコントロールプレーンのバックアップおよびリストア	3
1.1. 背景	3
1.2. バックアップ管理オプション	3
第2章 バックアップノードの準備	5
2.1. NFS サーバーの準備	5
2.2. バックアップディレクトリーの作成およびエクスポート	5
第3章 RELAX AND RECOVER (REAR) のインストールおよび設定	7
3.1. REAR のインストール	7
3.2. REAR 設定ファイルの作成	7
第4章 バックアップ手順の実行	9
4.1. アンダークラウドのバックアップ作成前の前提条件タスクの実行	9
4.2. アンダークラウドのバックアップ	9
4.3. コントロールプレーンのバックアップ	10
第5章 復元手順の実行	14
5.1. アンダークラウドの復元	14
5.2. コントロールプレーンの復元	15
5.3. GALERA クラスターに関するトラブルシューティング	16
5.4. コロケーションされた CEPH モニターを持つアンダークラウドおよびコントロールプレーンノードの復元	19

第1章 アンダークラウドとコントロールプレーンのバックアップおよびリストア

本ガイドの手順を使用して、更新/アップグレードの前に、Red Hat OpenStack Platform 13 のアンダークラウドおよびオーバークラウドのコントローラーノード (以下、コントロールプレーンノードと呼びます) の状態をバックアップするステップを説明します。この手順により、更新またはアップグレード時にエラーが発生した場合に、アンダークラウドおよびオーバークラウドのコントロールプレーンノードを以前の状態にリストアすることができます。

1.1. 背景

本ガイドの手順では、Bash で書かれたオープンソースの災害復旧ソリューション Relax and Recover (ReaR) を使用します。ReaR は、アンダークラウドまたはコントロールプレーンノードの最新状態で設定されるブート可能なイメージを作成します。ReaR を使用すると、システム管理者はバックアップを作成するファイルを選択することもできます。

ReaR では、以下に示すさまざまなブートメディア用フォーマットがサポートされます。

- ISO
- USB
- eSATA
- PXE

本書に記載する例は、**ISO** ブートフォーマットを使用してテストされています。

ReaR は、以下に示すさまざまなプロトコルを使用してブートイメージを転送することができます。

- HTTP/HTTPS
- SSH/SCP
- FTP/SFTP
- NFS
- CIFS (SMB)

Red Hat OpenStack Platform 13 のアンダークラウドとオーバークラウドのコントロールプレーンノードをバックアップおよびリストアする目的で、本書に記載する例は、NFS を使用してテストされています。

1.2. バックアップ管理オプション

ReaR では、内部および外部両方のバックアップ管理オプションを使用することができます。

内部バックアップ管理

内部バックアップ管理オプションを以下に示します。

- **tar**
- **rsync**

外部バックアップ管理

外部バックアップ管理オプションには、オープンソースのソリューションおよびプロプライエタリーのソリューションの両方が含まれます。オープンソースのソリューションを以下に示します。

- Bacula
- Bareos

プロプライエタリーのソリューションを以下に示します。

- EMC NetWorker (Legato)
- HP DataProtector
- IBM Tivoli Storage Manager (TSM)
- Symantec NetBackup

第2章 バックアップノードの準備

アンダークラウドまたはコントロールプレーンノードをバックアップする前に、バックアップイメージを受け入れるバックアップノードを準備します。

2.1. NFS サーバーの準備

ReaR では、複数の転送手段を使用することができます。Red Hat は、NFS を使用した ReaR によるバックアップおよびリストアをサポートします。

1. バックアップノードに NFS サーバーをインストールします。

```
[root@backup ~]# yum install -y nfs-utils
```

2. ファイアウォールに NFS サービスを追加して、ポート **111** および **2049** が開放されている状態にします。以下に例を示します。

```
[root@backup ~]# firewall-cmd --add-service=nfs
[root@backup ~]# firewall-cmd --add-service=nfs --permanent
```

3. NFS サーバーを有効にし、さらに起動します。

```
[root@backup ~]# systemctl enable nfs-server
[root@backup ~]# systemctl restart nfs-server
```

2.2. バックアップディレクトリーの作成およびエクスポート

バックアップ用 ISO イメージをアンダークラウドまたはコントロールプレーンノードからバックアップノードにコピーするには、バックアップディレクトリーを作成する必要があります。

前提条件

- NFS サーバーをインストールして有効にしている。詳しい情報は、[NFS サーバーの準備](#) を参照してください。

手順

1. バックアップディレクトリーを作成します。

```
[root@backup ~]# mkdir /ctl_plane_backups
```

2. ディレクトリーをエクスポートします。<IP_ADDRESS/24> を IP アドレスおよびネットワークのサブネットマスクに置き換えてください。

```
[root@backup ~]# cat >> /etc/exports << EOF
/ctl_plane_backups <IP_ADDRESS/24>(rw,sync,no_root_squash,no_subtree_check)
EOF
```

/etc/exports ファイルのエントリーは、スペース区切りリストの形式です。アンダークラウドおよびオーバークラウドのコントロールプレーンノードが異なるネットワークまたはサブネットを使用している場合には、以下の例に示すように、それぞれのネットワークまたはサブネットについて、このステップを繰り返します。

```
cat >> /etc/exports << EOF
/ctl_plane_backups 192.168.24.0/24(rw,sync,no_root_squash,no_subtree_check) /
ctl_plane_backups 10.0.0.0/24(rw,sync,no_root_squash,no_subtree_check) /
ctl_plane_backups 172.16.0.0/24(rw,sync,no_root_squash,no_subtree_check)
EOF
```

3. NFS サーバーを再起動します。

```
[root@backup ~]# systemctl restart nfs-server
```

4. エントリーが NFS サーバーで正しく設定されていることを確認します。

```
[root@backup ~]# showmount -e `hostname`
```

第3章 RELAX AND RECOVER (REAR) のインストールおよび設定

アンダークラウドおよびオーバークラウドのコントロールプレーンをバックアップするには、まずアンダークラウドおよび各コントロールプレーンノードに ReaR をインストール/設定する必要があります。

3.1. REAR のインストール

アンダークラウドおよびコントロールプレーンをバックアップする前に、Relax and Recover (ReaR) パッケージおよび ISO イメージを生成するためのパッケージを、アンダークラウドノードおよび各コントロールプレーンノードにインストールする必要があります。

手順

1. ReaR パッケージおよび ISO イメージを生成するためのパッケージを、アンダークラウドノードおよび各コントロールプレーンノードにインストール

```
[root@controller-x ~]# yum install rear genisoimage nfs-utils -y
```

2. 今後のアップグレードの問題を回避するには、予定されている ReaR cron タスクを削除します。

```
[root@controller-x ~]# rm -f /etc/cron.d/rear
```

3. アンダークラウドおよび各コントロールプレーンノードにバックアップディレクトリーを作成します。

```
[root@controller-x ~]# mkdir -p /ctl_plane_backups
```

4. NFS を実行するバックアップノードの **ctl_plane_backups** NFS ディレクトリーを、アンダークラウドおよび各コントロールプレーンノードにマウントします。

```
[root@controller-x ~]# mount -t nfs <BACKUP_NODE_IP_ADDRESS>:/ctl_plane_backups  
/ctl_plane_backups
```

<BACKUP_NODE_IP_ADDRESS> を NFS サーバーを実行するバックアップノードの IP アドレスに置き換えてください。

3.2. REAR 設定ファイルの作成

Relax and Recovery (ReaR) をアンダークラウドノードおよび各コントロールプレーンノードにインストールしたら、**root** ユーザーとして設定ファイルを作成します。

手順

1. ReaR 設定ファイルを作成します。

```
[root@controller-x ~]# mkdir -p /etc/rear  
[root@controller-x ~]# tee -a "/etc/rear/local.conf" > /dev/null <<'EOF'  
OUTPUT=ISO  
OUTPUT_URL=nfs://<BACKUP_NODE_IP_ADDRESS>/ctl_plane_backups  
ISO_PREFIX=<NODE_HOSTNAME>
```

```

BACKUP=NETFS
BACKUP_PROG_COMPRESS_OPTIONS=( --gzip )
BACKUP_PROG_COMPRESS_SUFFIX=".gz"
BACKUP_PROG_EXCLUDE=( '/tmp/*' '/data/*' )
BACKUP_URL=nfs://<BACKUP_NODE_IP_ADDRESS>/ctl_plane_backups
BACKUP_PROG_EXCLUDE=("${BACKUP_PROG_EXCLUDE[@]}" '/media' '/var/tmp'
'/var/crash')
BACKUP_PROG_OPTIONS+=( --anchored --xattrs-include='*.*' --xattrs )
EOF

```

- **<NODE_HOSTNAME>** をノードのホスト名に置き換えてください。たとえば、ノードのホスト名が **controller-0** の場合は、**<NODE_HOSTNAME>** を **controller-0** に置き換えます。
- **<BACKUP_NODE_IP_ADDRESS>** を、NFS サーバーを実行するバックアップノードの IP アドレスに置き換えます。これは、バックアップノードの準備時に設定した IP アドレスです。詳細は、[2章バックアップノードの準備](#)を参照してください。



重要

アンダークラウドまたはコントロールプレーンノードがブートモード UEFI を使用する場合は、設定ファイルに **USING_UEFI_BOOTLOADER=1** を追加する必要があります。

2. ネットワークインターフェイスに OVS ブリッジを使用する場合は、次の設定を **/etc/rear/local.conf** ファイルに追加して、リカバリープロセスがネットワーク経由でバックアップを取得できるようにします。

```

NETWORKING_PREPARATION_COMMANDS=( '<command_1>' '<command_2>' ...
'<command_n>' )

```

<command_1>、および **<command_n>** までのすべてのコマンドを、ネットワーク設定を設定するために必要なコマンドに置き換えます。次に例を示します。

- コントロールプレーンブリッジ名を設定するには、**ip link add br-ctlplane type bridge** コマンドを追加します。
- インターフェイスの名前を設定するには、**ip link set eth0 up** コマンドを追加します。

3. **rescue.conf** ファイルを作成します。

```

[root@controller-x ~]# tee -a "/etc/rear/rescue.conf" > /dev/null <<'EOF'
BACKUP_PROG_OPTIONS+=( --anchored --xattrs-include='*.*' --xattrs )
EOF

```

第4章 バックアップ手順の実行

Fast Forward Upgrade を実施する前に、アンダークラウドおよびオーバークラウドのコントロールプレーンノードをバックアップし、エラーが発生した場合に以前の状態に復元できるようにします。



注記

アンダークラウドおよびオーバークラウドのバックアップを作成する前に、アンダークラウドからオーバークラウドに対する操作を実施しないようにしてください。

4.1. アンダークラウドのバックアップ作成前の前提条件タスクの実行

アンダークラウドをデプロイする場合や、既存のアンダークラウドに変更を加える場合には、アンダークラウドのバックアップを実行しないでください。

データの破損を防ぐには、アンダークラウドノードをバックアップする前に、スタックのエラーや実行中のタスクがないこと、および **mariadb** を除くすべての OpenStack サービスが停止していることを確認してください。

手順

1. スタックでエラーが発生していないことを確認します。<STACKNAME> はスタック名に置き換えてください。デプロイして利用可能なすべてのスタックにコマンドを使用します。

```
(undercloud) [stack@undercloud-0 ~]$ openstack stack failures list <STACKNAME>
```

2. アンダークラウドで実行中のタスクがないことを確認します。

```
(undercloud) [stack@undercloud-0 ~]$ openstack stack list --nested | grep -v "_COMPLETE"
```

コマンドによって結果が返されなければ、実行中のタスクはありません。

3. アンダークラウド上のすべての OpenStack サービスを停止します。

```
# systemctl stop openstack-*
# systemctl stop neutron-*
# systemctl stop ironic*
# systemctl stop haproxy
# systemctl stop httpd
```

4. **mariadb** が実行していることを確認します。

```
# sudo systemctl status mariadb
```

4.2. アンダークラウドのバックアップ

アンダークラウドノードをバックアップするには、アンダークラウドノードで root ユーザーとしてログインする必要があります。データベースを復元できるように、バックアップを作成する必要がある点に注意してください。

前提条件

- バックアップディレクトリーを作成し、エクスポートしている。詳細は、[バックアップディレクトリーの作成およびエクスポート](#) を参照してください。
- アンダークラウドをバックアップする前の、前提条件のタスクを実施している。詳細は、[バックアップディレクトリーの作成およびエクスポート](#) を参照してください。
- 各コントロールプレーンノードに ReaR をインストールし、設定している。詳細は、[Relax and Recover \(ReaR\) のインストールと設定](#) を参照してください。

手順

1. データベースのパスワードを特定します。

```
[root@undercloud stack]# PASSWORD=$(sudo /bin/hiera -c /etc/puppet/hiera.yaml
mysql::server::root_password)
```

2. データベースのバックアップを作成します。

```
[root@undercloud stack]# mysql -uroot -p$PASSWORD -s -N -e "select distinct
table_schema from information_schema.tables where engine='innodb' and table_schema !=
'mysql';" | xargs mysqldump -uroot -p$PASSWORD --single-transaction --databases >
openstack-backup-mysql.sql
```

```
[root@undercloud stack]# mysql -uroot -p$PASSWORD -s -N -e "SELECT
CONCAT("\SHOW GRANTS FOR '",user,"','@','host',';') FROM mysql.user where
(length(user) > 0 and user NOT LIKE 'root')" | xargs n1 mysql -uroot -p$PASSWORD -s -N -
e | sed 's/;/;' > openstack-backup-mysql-grants.sql
```

3. **mariadb** データベースサービスを停止します。

```
[root@undercloud stack]# systemctl stop mariadb
```

4. バックアップを作成します。

```
[root@undercloud stack]# rear -d -v mkbackup
```

ReaR で作成するバックアップの ISO ファイルは、バックアップノードの **/ctl_plane_backups** ディレクトリーにあります。

5. アンダークラウドを再起動します。
 - a. アンダークラウドに stack ユーザーとしてログインします。
 - b. アンダークラウドを再起動します。

```
[stack@undercloud]$ sudo reboot
```

4.3. コントロールプレーンのバックアップ

コントロールプレーンをバックアップするには、まず Pacemaker クラスターおよびコントロールプレーンノード上で動作しているすべてのコンテナを停止する必要があります。状態の整合性確保のためにスタックを操作しないでください。バックアップ手順が完了したら、Pacemaker クラスターおよびコンテナを起動します。

Pacemaker クラスターおよびコンテナを再起動した後にデータベースを復元できるように、データベースをバックアップする必要がある点に注意してください。

コントロールプレーンノードを同時にバックアップします。

前提条件

- バックアップディレクトリーを作成し、エクスポートしている。詳細は、[バックアップディレクトリーの作成およびエクスポート](#) を参照してください。
- 各コントロールプレーンノードに ReaR をインストールし、設定している。詳細は、[Relax and Recover \(ReaR\) のインストールと設定](#) を参照してください。

手順

1. データベースのパスワードを特定します。

```
[heat-admin@overcloud-controller-x ~]# PASSWORD=$(sudo /bin/hiera -c /etc/puppet/hiera.yaml mysql::server::root_password)
```

2. データベースのバックアップを作成します。

```
[heat-admin@overcloud-controller-x ~]# mysql -uroot -p$PASSWORD -s -N -e "select distinct table_schema from information_schema.tables where engine='innodb' and table_schema != 'mysql';" | xargs mysqldump -uroot -p$PASSWORD --single-transaction --databases > openstack-backup-mysql.sql
```

```
[heat-admin@overcloud-controller-x ~]# mysql -uroot -p$PASSWORD -s -N -e "SELECT CONCAT("\nSHOW GRANTS FOR ",user,"@",host,";") FROM mysql.user where (length(user) > 0 and user NOT LIKE 'root');" | xargs -n1 mysql -uroot -p$PASSWORD -s -N -e | sed 's/$/;' > openstack-backup-mysql-grants.sql
```



注記

データベースのバックアップは、万が一を考慮しての措置です。このステップにより、復元手順の一部として Galera クラスターが自動的に復元されない場合に、手動で復元することができます。Galera クラスターの復元に関する詳細は、[Galera クラスターに関するトラブルシューティング](#) を参照してください。

3. コントロールプレーンノードのいずれかで、Pacemaker クラスターを停止します。



重要

スタックは操作しないでください。Pacemaker クラスターおよびコンテナを停止すると、コンピュータノードへのコントロールプレーンサービスが一時的に中断します。また、ネットワーク接続、Ceph、および NFS データプレーンサービスにも中断が発生します。この手順の最終ステップに従い、Pacemaker クラスターおよびコンテナがサービスに戻るまで、インスタンスの作成、インスタンスの移行、要求の認証、クラスターの健全性の監視はできません。

```
[heat-admin@overcloud-controller-x ~]# sudo pcs cluster stop --all
```

4. 各コントロールプレーンノードで、コンテナを停止します。

- a. コンテナを停止します。

```
[heat-admin@overcloud-controller-x ~]# sudo docker stop $(sudo docker ps -a -q)
```

- b. ceph-mon@controller.service コンテナを停止します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl stop ceph-mon@$(hostname -s)
```

- c. ceph-mgr@controller.service コンテナを停止します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl stop ceph-mgr@$(hostname -s)
```

5. オプション: **ganesh-nfs** を使用する場合は、1つのコントローラー上のファイルサーバーを無効にします。

```
[heat-admin@overcloud-controller-x ~]# sudo pcs resource disable ceph-nfs
```

6. オプション: ceph サービスの **ceph-mds** および **ceph-rgw** を使用する場合は、以下のサービスを停止します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl stop ceph-mds@$(hostname -s)
```

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl stop ceph-rgw@$(hostname -s)
```

7. コントロールプレーンをバックアップするには、各コントロールプレーンノードでコントロールプレーンのバックアップを実行します。

```
[heat-admin@overcloud-controller-x ~]# sudo rear -d -v mkbackup
```

ReaR で作成するバックアップの ISO ファイルは、バックアップノードの **/ctl_plane_backups** ディレクトリーにあります。

注記

バックアップコマンドを実行すると、**tar** コマンドおよび tar プロセス時に無視されるソケットに関する以下のような警告メッセージが表示されることがあります。

```
WARNING: tar ended with return code 1 and below output:
```

```
---snip---
```

```
tar: /var/spool/postfix/public/qmgr: socket ignored
```

```
...
```

```
...
```

This message indicates that files have been modified during the archiving process and the backup might be inconsistent. Relax-and-Recover continues to operate, however, it is important that you verify the backup to ensure that you can use this backup to recover your system.

8. バックアップ手順により各コントロールプレーンノードの ISO イメージが生成されたら、Pacemaker クラスターを再起動します。コントロールプレーンノードのいずれかで、以下のコマンドを入力します。

```
[heat-admin@overcloud-controller-x ~]# sudo pcs cluster start --all
```

9. 各コントロールプレーンノードで、コンテナを起動します。

- a. コンテナを起動します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl restart docker
```

- b. **ceph-mon@controller.service** コンテナを起動します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl start ceph-mon@$(hostname -s)
```

- c. **ceph-mgr@controller.service** コンテナを起動します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl start ceph-mgr@$(hostname -s)
```

10. オプション: **ceph-mds** および **ceph-rgw** を使用する場合は、以下のサービスを起動します。

```
[heat-admin@overcloud-controller-x ~]# sudo systemctl start ceph-rgw@$(hostname -s)
[heat-admin@overcloud-controller-x ~]# sudo systemctl start ceph-mds@$(hostname -s)
```

11. オプション: **ganesh-nfs** を使用する場合は、1つのコントローラー上のファイルサーバーを有効にします。

```
[heat-admin@overcloud-controller-x ~]# sudo pcs resource enable ceph-nfs
```

第5章 復元手順の実行

更新またはアップグレード中にエラーが発生した場合には、アンダークラウドもしくはオーバークラウドのコントロールプレーンノードのいずれか (またはその両方) をリストアして、以前の状態にすることができます。復元手順の一部として Galera クラスタが自動的に復元されない場合は、クラスタを手動で復元する必要があります。

कोरोケーションされた Ceph モニターを持つアンダークラウドまたはオーバークラウドのコントロールプレーンノードを復元することもできます。



注記

ISO ファイルからブートする場合には、アンダークラウドおよびオーバークラウドが NFS サーバーにアクセスできるようにします。

以下の一般的な手順を使用します。

1. ブート可能な ISO イメージを DVD に書き込むか、または ILO リモートアクセスを通じてイメージを読み込みます。
2. リカバリーメディアからリストアが必要なノードをブートします。
3. **Recover <HOSTNAME>** を選択します。<HOSTNAME> は復元するノードの名前に置き換えてください。
4. ユーザー **root** としてログオンします。
5. バックアップを復元します。

5.1. アンダークラウドの復元

Fast Forward Upgrade を実施中にエラーが発生した場合には、「[アンダークラウドのバックアップ](#)」に記載の手順により作成した ISO イメージを使用して、アンダークラウドノードを以前に保存した状態にリストアすることができます。バックアップの手順により、ISO イメージは「[バックアップディレクトリーの作成およびエクスポート](#)」のステップで作成したバックアップノードのフォルダーに保存されます。

手順

1. アンダークラウドノードをシャットダウンします。次のステップに進む前に、アンダークラウドノードが完全にシャットダウンされていることを確認します。
2. バックアッププロセス中に作成した ISO イメージを使用してアンダークラウドノードをブートし、リストアします。ISO イメージは、バックアップノードの `/ctl_plane_backups` ディレクトリーにあります。
3. **Relax-and-Recover** ブートメニューが表示されたら、**Recover <UNDERCLOUD_NODE>** を選択します。<UNDERCLOUD_NODE> は、アンダークラウドノードの名前になります。
4. **root** ユーザーとしてログインします。
以下のメッセージが表示されます。

```
Welcome to Relax-and-Recover. Run "rear recover" to restore your system!
RESCUE <UNDERCLOUD_NODE>:~ # rear recover
```

イメージのリストアがすぐに行われます。完了すると、コンソールは以下のメッセージを受信します。

```
Finished recovering your system
Exiting rear recover
Running exit tasks
```

5. コマンドラインインターフェイスが利用できる状態になったら、イメージのリストアプロセスは完了です。ノードの電源をオフにします。

```
RESCUE <UNDERCLOUD_NODE>:~ # poweroff
```

ノードをブートすると、以前の状態で再開されます。

5.2. コントロールプレーンの復元

Fast Forward Upgrade を実施中にエラーが発生した場合には、「[コントロールプレーンのバックアップ](#)」に記載の手順により作成した ISO イメージを使用して、コントロールプレーンノードを以前に保存した状態にリストアすることができます。コントロールプレーンをリストアする場合には、状態の整合性を確保するために、すべてのコントロールプレーンノードを以前の状態にリストアする必要があります。



注記

Red Hat は、Open vSwitch (OVS) およびデフォルトの Open Virtual Network (OVN) などのネイティブ SDN を使用する Red Hat OpenStack Platform のバックアップをサポートします。サードパーティーの SDN についての詳細は、サードパーティーの SDN ドキュメントを参照してください。

手順

1. 各コントロールプレーンノードをシャットダウンします。次のステップに進む前に、コントロールプレーンノードが完全にシャットダウンされていることを確認します。
2. バックアッププロセス中に作成した ISO イメージを使用してコントロールプレーンノードをブートし、リストアします。ISO イメージは、バックアップノードの `/ctl_plane_backups` ディレクトリにあります。
3. **Relax-and-Recover** ブートメニューが表示されたら、**Recover <CONTROL_PLANE_NODE>** を選択します。<CONTROL_PLANE_NODE> をコントロールプレーンノードの名前に置き換えます。以下のメッセージが表示されます。

```
Welcome to Relax-and-Recover. Run "rear recover" to restore your system!
RESCUE <CONTROL_PLANE_NODE>:~ # rear recover
```

イメージのリストアがすぐに行われます。リストアが完了すると、コンソールは以下のメッセージを受信します。

```
Finished recovering your system
Exiting rear recover
Running exit tasks
```

コマンドラインインターフェイスが利用できる状態になったら、イメージのリストアプロセスは完了です。ノードの電源をオフにします。

```
RESCUE <CONTROL_PLANE_NODE>:~ # poweroff
```

ブートシーケンスを通常のブートデバイスに設定します。ノードをブートすると、以前の状態で再開されます。

4. サービスが正常に実行されていることを確認するには、pacemaker のステータスを確認します。**root** ユーザーとしてコントローラーにログインし、以下のコマンドを実行します。

```
# pcs status
```

5. オーバークラウドのステータスを確認するには、Tempest を使用します。Tempest の詳細は、OpenStack Integration Test Suite Guide の [Configuring the OpenStack Integration Test Suite](#) を参照してください。

5.3. GALERA クラスターに関するトラブルシューティング

復元手順の一部として Galera クラスターが復元されない場合は、Galera を手動で復元する必要があります。



注記

以下の手順では、1つのコントローラーノードでいくつかのステップを実施する必要があります。手順の実施と同じコントローラーノードで、これらのステップを実施するようにしてください。

手順

1. Controller-0 で、Galera クラスターの仮想 IP を取得します。

```
$ sudo hiera -c /etc/puppet/hiera.yaml mysql_vip
```

2. すべてのコントローラーノードで、仮想 IP を通じたデータベース接続を無効にします。

```
$ sudo iptables -I INPUT -p tcp --destination-port 3306 -d $MYSQL_VIP -j DROP
```

3. Controller-0 で MySQL の root パスワードを取得します。

```
$ sudo hiera -c /etc/puppet/hiera.yaml mysql::server::root_password
```

4. Controller-0 で、Galera リソースを **unmanaged** モードに設定します。

```
$ sudo pcs resource unmanage galera-bundle
```

5. すべてのコントローラーノードで、MySQL コンテナを停止します。

```
$ sudo docker container stop $(sudo docker container ls --all --format "{{.Names}}") --filter=name=galera-bundle)
```

6. すべてのコントローラーノードで、現在のディレクトリーを移動します。

```
$ sudo mv /var/lib/mysql /var/lib/mysql-save
```

7. すべてのコントローラーノードで、新規ディレクトリー **/var/lib/mysql** を作成します。

```
$ sudo mkdir /var/lib/mysql
$ sudo chown 42434:42434 /var/lib/mysql
$ sudo chcon -t container_file_t /var/lib/mysql
$ sudo chmod 0755 /var/lib/mysql
$ sudo chcon -r object_r /var/lib/mysql
$ sudo chcon -u system_u /var/lib/mysql
```

8. すべてのコントローラーノードで、MySQL コンテナを起動します。

```
$ sudo docker container start $(sudo docker container ls --all --format "{{ .Names }}" --
filter=name=galera-bundle)
```

9. すべてのコントローラーノードで、MySQL データベースを作成します。

```
$ sudo docker exec -i $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mysql_install_db --datadir=/var/lib/mysql --
user=mysql"
```

10. すべてのコントローラーノードで、データベースを起動します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mysqld_safe --skip-networking --wsrep-on=OFF" &
```

11. すべてのコントローラーノードで、**.my.cnf** Galera 設定ファイルを移動します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mv /root/.my.cnf /root/.my.cnf.bck"
```

12. すべてのコントローラーノードで、Galera root パスワードをリセットします。

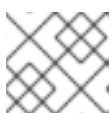
```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mysql -uroot -e'use mysql;update user set
password=PASSWORD(\"$ROOTPASSWORD\")where User='root';flush privileges;"
```

13. すべてのコントローラーノード上の Galera コンテナ内で、**.my.cnf** Galera 設定ファイルを復元します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mv /root/.my.cnf.bck /root/.my.cnf"
```

14. Controller-0 で、バックアップデータベースファイルを **/var/lib/MySQL** にコピーします。

```
$ sudo cp $BACKUP_FILE /var/lib/mysql
$ sudo cp $BACKUP_GRANT_FILE /var/lib/mysql
```



注記

これらのファイルへのパスは **/home/heat-admin/** です。

15. Controller-0 で、MySQL データベースを復元します。

```
$ sudo docker exec $(docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mysql -u root -p$ROOT_PASSWORD <
\var/lib/mysql/$BACKUP_FILE \ "

$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mysql -u root -p$ROOT_PASSWORD <
\var/lib/mysql/$BACKUP_GRANT_FILE \ "
```

16. すべてのコントローラーノードで、データベースをシャットダウンします。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "mysqladmin shutdown"
```

17. Controller-0 で、ブートストラップノードを起動します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" --
filter=name=galera-bundle) \
/usr/bin/mysqld_safe --pid-file=/var/run/mysql/mysqld.pid --
socket=/var/lib/mysql/mysql.sock --datadir=/var/lib/mysql \
--log-error=/var/log/mysql_cluster.log --user=mysql --open-files-limit=16384 \
--wsrep-cluster-address=gcomm:// &
```

18. 検証: Controller-0 で、クラスターのステータスを確認します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "clustercheck"
```

Galera cluster node is synced のメッセージが表示されるのを確認してください。表示されない場合は、ノードを再作成する必要があります。

19. Controller-0 で、設定からクラスターアドレスを取得します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "grep wsrep_cluster_address /etc/my.cnf.d/galera.cnf" |
awk '{print $3}'
```

20. 残りの各コントローラーノードでデータベースを起動し、クラスターを検証します。

- a. データベースを起動します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) /usr/bin/mysqld_safe --pid-
file=/var/run/mysql/mysqld.pid --socket=/var/lib/mysql/mysql.sock \
--datadir=/var/lib/mysql --log-error=/var/log/mysql_cluster.log --user=mysql --open-
files-limit=16384 \
--wsrep-cluster-address=$CLUSTER_ADDRESS &
```

- b. MYSQL クラスターのステータスを確認します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" \
--filter=name=galera-bundle) bash -c "clustercheck"
```

Galera cluster node is synced のメッセージが表示されるのを確認してください。表示されない場合は、ノードを再作成する必要があります。

- すべてのコントローラーノードで MySQL コンテナを停止します。

```
$ sudo docker exec $(sudo docker container ls --all --format "{{ .Names }}" --
filter=name=galera-bundle) \
/usr/bin/mysqladmin -u root shutdown
```

- すべてのコントローラーノードで以下のファイアウォールルールを削除して、仮想 IP アドレス経由のデータベース接続を許可します。

```
$ sudo iptables -D INPUT -p tcp --destination-port 3306 -d $MYSQL_VIP -j DROP
```

- すべてのコントローラーノードで MySQL コンテナを再起動します。

```
$ sudo docker container restart $(sudo docker container ls --all --format "{{ .Names }}" --
filter=name=galera-bundle)
```

- すべてのコントローラーノードで **clustercheck** コンテナを再起動します。

```
$ sudo docker container restart $(sudo docker container ls --all --format "{{ .Names }}" --
filter=name=clustercheck)
```

- Controller-0 で、Galera リソースを **managed** モードに設定します。

```
$ sudo pcs resource manage galera-bundle
```

5.4. コロケーションされた CEPH モニターを持つアンダークラウドおよびコントロールプレーンノードの復元

更新またはアップグレード中にエラーが発生した場合には、ReaR バックアップを使用して、アンダークラウドもしくはオーバークラウドのコントロールプレーンノードのいずれか (またはその両方) を以前の状態に復元することができます。

前提条件

- ReaR をインストールおよび設定していること。詳細は、[ReaR のインストールおよび設定](#) を参照してください。
- バックアップノードを準備します。詳しくは、[バックアップノードの準備](#) を参照してください。
- バックアップ手順を実行します。詳細は、[バックアップ手順の実行](#) を参照してください。

手順

- バックアップノードで、Ceph バックアップをホストする NFS ディレクトリーをエクスポートします。<IP_ADDRESS/24> を IP アドレスおよびネットワークのサブネットマスクに置き換えてください。

```
[root@backup ~]# cat >> /etc/exports << EOF
/ceph_backups <IP_ADDRESS/24>(rw,sync,no_root_squash,no_subtree_check)
EOF
```

2. アンダークラウドノードで、source コマンドでアンダークラウドの認証情報を読み込み、以下のスクリプトを実行します。

```
# source stackrc
```

```
#!/bin/bash
for i in `openstack server list -c Name -c Networks -f value | grep controller | awk -F='{' '{print $2}' | awk -F'}' '{print $1}'`; do ssh -q heat-admin@$i 'sudo systemctl stop ceph-mon@$(hostname -s) ceph-mgr@$(hostname -s)'; done
```

ceph-mgr@controller.service コンテナが停止したことを確認するには、以下のコマンドを入力します。

```
[heat-admin@overcloud-controller-x ~]# sudo docker ps | grep ceph
```

3. アンダークラウドノードで、source コマンドでアンダークラウドの認証情報を読み込み、以下のスクリプトを実行します。

```
# source stackrc
```

```
#!/bin/bash
for i in `openstack server list -c Name -c Networks -f value | grep controller | awk -F='{' '{print $2}' | awk -F'}' '{print $1}'`; do ssh -q heat-admin@$i 'sudo mkdir /ceph_backups'; done
```

```
#!/bin/bash
for i in `openstack server list -c Name -c Networks -f value | grep controller | awk -F='{' '{print $2}' | awk -F'}' '{print $1}'`; do ssh -q heat-admin@$i 'sudo mount -t nfs <BACKUP_NODE_IP_ADDRESS>:/ceph_backups /ceph_backups'; done
```

```
#!/bin/bash
for i in `openstack server list -c Name -c Networks -f value | grep controller | awk -F='{' '{print $2}' | awk -F'}' '{print $1}'`; do ssh -q heat-admin@$i 'sudo mkdir /ceph_backups/$(hostname -s)'; done
```

```
#!/bin/bash
for i in `openstack server list -c Name -c Networks -f value | grep controller | awk -F='{' '{print $2}' | awk -F'}' '{print $1}'`; do ssh -q heat-admin@$i 'sudo tar -zcv --xattrs-include=*. * --xattrs --xattrs-include=security.capability --xattrs-include=security.selinux --acls -f /ceph_backups/$(hostname -s)/$(hostname -s).tar.gz /var/lib/ceph'; done
```

4. 復元するノードで、以下のタスクを実行します。
 - a. 次に進む前にノードの電源を切ります。
 - b. バックアッププロセス中に作成した ReaR バックアップファイルでノードを復元します。ファイルは、バックアップノードの **/ceph_backups** ディレクトリーにあります。
 - c. **Relax-and-Recover** ブートメニューから、**Recover <CONTROL_PLANE_NODE>** を選択します。ここで、**<CONTROL_PLANE_NODE>** はコントロールプレーンノードの名前になります。

- d. プロンプトで以下のコマンドを入力します。

```
RESCUE <CONTROL_PLANE_NODE> :~ # rear recover
```

イメージの復元プロセスが完了すると、コンソールに以下のメッセージが表示されます。

```
Finished recovering your system
Exiting rear recover
Running exit tasks
```

5. 復元するノードに対して、Ceph バックアップを **/ceph_backups** ディレクトリーから **/var/lib/ceph** ディレクトリーにコピーします。

- a. システムのマウントポイントを特定します。

```
RESCUE <CONTROL_PLANE_NODE>:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        16G   0  16G   0% /dev
tmpfs           16G   0  16G   0% /dev/shm
tmpfs           16G  8.4M  16G   1% /run
tmpfs           16G   0  16G   0% /sys/fs/cgroup
/dev/vda2       30G  13G  18G  41% /mnt/local
```

/dev/vda2 ファイルシステムは **/mnt/local** にマウントされます。

- b. 一時ディレクトリーを作成します。

```
RESCUE <CONTROL_PLANE_NODE>:~ # mkdir /tmp/restore
RESCUE <CONTROL_PLANE_NODE>:~ # mount -v -t nfs -o rw,noatime
<BACKUP_NODE_IP_ADDRESS>:/ceph_backups /tmp/restore/
```

- c. コントロールプレーンノードで、既存の **/var/lib/ceph** ディレクトリーを削除します。

```
RESCUE <CONTROL_PLANE_NODE>:~ # rm -rf /mnt/local/var/lib/ceph/*
```

- d. 以前の Ceph マップを復元します。 **<control_plane_node>** をコントロールプレーンノードの名前に置き換えます。

```
RESCUE <CONTROL_PLANE_NODE>:~ # tar -xvC /mnt/local/ -f
/tmp/restore/<CONTROL_PLANE_NODE>/<CONTROL_PLANE_NODE>.tar.gz --xattrs -
-xattrs-include=*. * var/lib/ceph
```

- e. ファイルが復元されていることを確認します。

```
RESCUE <CONTROL_PLANE_NODE>:~ # ls -l
total 0
drwxr-xr-x 2 root 107 26 Jun 18 18:52 bootstrap-mds
drwxr-xr-x 2 root 107 26 Jun 18 18:52 bootstrap-osd
drwxr-xr-x 2 root 107 26 Jun 18 18:52 bootstrap-rbd
drwxr-xr-x 2 root 107 26 Jun 18 18:52 bootstrap-rgw
drwxr-xr-x 3 root 107 31 Jun 18 18:52 mds
drwxr-xr-x 3 root 107 31 Jun 18 18:52 mgr
drwxr-xr-x 3 root 107 31 Jun 18 18:52 mon
```

```
drwxr-xr-x 2 root 107  6 Jun 18 18:52 osd
drwxr-xr-x 3 root 107 35 Jun 18 18:52 radosgw
drwxr-xr-x 2 root 107  6 Jun 18 18:52 tmp
```

6. ノードの電源を切ります。

```
RESCUE <CONTROL_PLANE_NODE> :~ # poweroff
```

7. ノードの電源をオンにします。ノードは以前の状態で再開します。