



# Red Hat OpenStack Platform 17.1

## Red Hat OpenStack Platform のマイナー更新の 実行

最新のバグ修正とセキュリティ強化を Red Hat OpenStack Platform に適用する



# Red Hat OpenStack Platform 17.1 Red Hat OpenStack Platform のマイナー更新の実行

---

最新のバグ修正とセキュリティー強化を Red Hat OpenStack Platform に適用する

OpenStack Team  
rhos-docs@redhat.com

## 法律上の通知

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<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

Red Hat OpenStack Platform (RHOSP) 環境のマイナー更新を実行して、最新のパッケージとコンテナで最新の状態に保つことができます。

## 目次

多様性を受け入れるオープンソースの強化 .....	3
RED HAT ドキュメントへのフィードバック (英語のみ) .....	4
<b>第1章 マイナー更新の準備 .....</b>	<b>5</b>
更新を妨げる可能性のある既知の問題 .....	6
手順 .....	6
1.1. ロングライフリリースのアップグレードパス .....	7
1.2. 環境の RED HAT ENTERPRISE LINUX リリースへのロック .....	7
1.3. RED HAT OPENSTACK PLATFORM リポジトリーの更新 .....	8
1.4. コンテナイメージ準備ファイルの更新 .....	10
1.5. オーバークラウドでのフェンシングの無効化 .....	10
<b>第2章 アンダークラウドの更新 .....</b>	<b>12</b>
前提条件 .....	12
2.1. アンダークラウドの更新前の RHOSP の検証 .....	12
2.2. SSH キーのサイズの検証 .....	13
2.3. コンテナ化されたアンダークラウドのマイナー更新を実施する .....	13
2.4. オーバークラウドイメージの更新 .....	14
<b>第3章 オーバークラウドの更新 .....</b>	<b>16</b>
前提条件 .....	16
手順 .....	16
3.1. オーバークラウドの更新準備タスクの実施 .....	16
3.2. コンテナイメージ準備タスクの実行 .....	18
3.3. オプション: すべてのオーバークラウドサーバーでの OVN-CONTROLLER コンテナの更新 .....	18
3.4. PACEMAKER 制御サービスのコンテナイメージ名の更新 .....	19
3.5. すべてのコントローラーノードを更新する .....	19
3.6. PACEMAKER 以外のサービスでのコンポーザブルロールの更新 .....	20
3.7. すべての COMPUTE ノードを更新する .....	20
3.8. すべての HCI COMPUTE ノードを更新する .....	21
3.9. すべての DISTRIBUTEDCOMPUTEHCI ノードの更新 .....	22
3.10. すべての CEPH STORAGE ノードの更新 .....	22
3.11. RED HAT CEPH STORAGE クラスターの更新 .....	23
3.12. データベースのオンライン更新の実施 .....	24
3.13. オーバークラウドでのフェンシングの再有効化 .....	25
<b>第4章 オーバークラウドの再起動 .....</b>	<b>26</b>
4.1. コントローラーノードおよびコンポーザブルノードの再起動 .....	26
4.2. CEPH STORAGE (OSD) クラスターの再起動 .....	27
4.3. COMPUTE ノードの再起動 .....	28
4.4. オーバークラウドの更新後の RHOSP の検証 .....	30



## 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

## RED HAT ドキュメントへのフィードバック (英語のみ)

Red Hat ドキュメントに対するご意見をお聞かせください。ドキュメントの改善点があればお知らせください。

### Jira でドキュメントのフィードバックを提供する

ドキュメントに関するフィードバックを提供するには、[Create Issue](#) フォームを使用します。Red Hat OpenStack Platform Jira プロジェクトで Jira Issue が作成され、フィードバックの進行状況を追跡できます。

1. Jira にログインしていることを確認してください。Jira アカウントをお持ちでない場合は、アカウントを作成してフィードバックを送信してください。
2. [Create Issue](#) をクリックして、**Create Issue** ページを開きます。
3. **Summary** フィールドと **Description** フィールドに入力します。**Description** フィールドに、ドキュメントの URL、章またはセクション番号、および問題の詳しい説明を入力します。フォーム内の他のフィールドは変更しないでください。
4. **Create** をクリックします。



## 第1章 マイナー更新の準備

お使いの Red Hat OpenStack Platform (RHOSP) 17.1 環境を、最新のパッケージおよびコンテナで更新された状態に維持してください。

次のバージョンのアップグレードパスを使用します。

更新前の RHOSP バージョン	新規 RHOSP バージョン
Red Hat OpenStack Platform 17.0.z	Red Hat OpenStack Platform 17.1 (最新)
Red Hat OpenStack Platform 17.1.z	Red Hat OpenStack Platform 17.1 (最新)

### マイナー更新のワークフロー

RHOSP 環境のマイナー更新には、アンダークラウドおよびオーバークラウドホスト上の RPM パッケージとコンテナ、および必要に応じてサービス設定の更新が含まれます。データプレーンとコントロールプレーンは、マイナー更新中に完全に利用可能になります。RHOSP 環境を更新するには、次の各手順を完了する必要があります。

更新手順	説明
アンダークラウドの更新	Director パッケージが更新され、コンテナが置き換えられ、アンダークラウドが再起動されます。
オプションの <b>ovn-controller</b> 更新	すべての <b>ovn-controller</b> コンテナは、すべてのコンピューティングホストとコントローラーホスト上で並行して更新されます。
ha-image-update 外部	Pacemaker 制御サービスのコンテナイメージ名を更新します。サービスの中断はありません。この手順は、システムをバージョン 17.0.z から最新の 17.1 リリースに更新するお客様にのみ該当します。
Pacemaker サービスを含むコントローラーノードとコンポーザブルノードのオーバークラウド更新	オーバークラウドの更新中、Pacemaker サービスは各ホストで停止します。Pacemaker サービスが停止している間、ホスト上の RPM、コンテナ設定データ、およびコンテナが更新されます。Pacemaker サービスが再起動すると、ホストが再度追加されます。
Pacemaker サービスを使用しないコンポーザブルノードのオーバークラウド更新	Networker、ObjectStorage、BlockStorage、または Pacemaker サービスを含まないその他のロールは、一度に 1 ノードずつ更新されます。
コンピューターノードのオーバークラウド更新	複数のノードが並行して更新されます。ノードを並列実行する場合のデフォルト値は 25 です。
Ceph ノードのオーバークラウド更新	Ceph ノードは一度に 1 ノードずつ更新されます。

更新手順	説明
Ceph クラスターの更新	Ceph サービスは、 <b>cephadm</b> を使用して更新されます。更新はデーモンごとに行われ、 <b>CephMgr</b> 、 <b>CephMon</b> 、 <b>CephOSD</b> から始まり、その後追加のデーモンが続きます。



### 注記

インフラストラクチャーがマルチスタックの場合は、各オーバークラウドスタックを一度に1つずつ完全に更新します。分散コンピュートノード (DCN) インフラストラクチャーがある場合は、中央のロケーションでオーバークラウドを完全に更新してから、各エッジサイトでオーバークラウドを一度に1つずつ更新します。

さらに、管理者はマイナー更新中に次の操作を実行できます。

- 仮想マシンの移行
- 仮想マシンネットワークの作成
- 追加のクラウド操作の実行

マイナー更新中は、次の操作はサポートされません。

- Controller ノードの置き換え
- ロールのスケールインまたはスケールアウト

### RHOSP 環境を更新する前の考慮事項

更新プロセス中のガイドとして、次の情報を考慮してください。

- Red Hat は、アンダークラウドおよびオーバークラウドの制御プレーンをバックアップすることを推奨しています。ノードのバックアップについては、[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バックアップと復元](#) を参照してください。
- 更新を妨げる可能性のある既知の問題を把握してください。
- 更新する前に、可能な更新パスとアップグレードパスを把握してください。詳細は、「[ロングライフリリースのアップグレードパス](#)」を参照してください。
- 現在のメンテナンスリリースを確認するには、**\$ cat /etc/rhosp-release** を実行してください。環境を更新した後にこのコマンドを実行して、更新を検証することもできます。

### 更新を妨げる可能性のある既知の問題

現在、既知の問題はありません。

### 手順

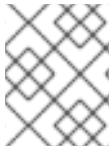
マイナー更新用に RHOSP 環境を準備するには、以下の手順を実行します。

1. [「環境の Red Hat Enterprise Linux リリースへのロック」](#)
2. [「Red Hat Openstack Platform リポジトリの更新」](#)

3. 「コンテナイメージ準備ファイルの更新」
4. 「オーバークラウドでのフェンシングの無効化」

## 1.1. ロングライフリリースのアップグレードパス

更新を開始する前に、可能な更新およびアップグレードパスを把握する必要があります。



### 注記

RHOSP および RHEL の現行バージョンは、`/etc/rhosp-release` および `/etc/redhat-release` ファイルで確認できます。

表1.1 バージョンの更新パス

現行バージョン	更新後のバージョン
RHEL 9.0 / RHOSP 17.0.x	最新の RHEL 9.0 / 最新の RHOSP 17.0
RHEL 9.2 / RHOSP 17.1.x	最新の RHEL 9.2 / 最新の RHOSP 17.1

表1.2 バージョンのアップグレードパス

現行バージョン	更新後のバージョン
RHEL 7.7 上の RHOSP 10	最新の RHEL 7.9 における最新の RHOSP 13
RHEL 7.9 上の RHOSP 13	最新の RHEL 8.2 における最新の RHOSP 16.1
RHEL 7.9 上の RHOSP 13	最新の RHEL 8.4 における最新の RHOSP 16.2
RHEL 8.4 の RHOSP 16	最新の RHEL 9.0 における最新の RHOSP 17.1

詳細は、[16.2 から 17.1 へのアップグレードフレームワーク](#) を参照してください。

## 1.2. 環境の RED HAT ENTERPRISE LINUX リリースへのロック

Red Hat OpenStack Platform (RHOSP) 17.1 は Red Hat Enterprise Linux 9.2 (RHEL) でサポートされています。更新を実行する前に、アンダークラウドおよびオーバークラウドのリポジトリを RHEL 9.2 リリースにロックして、オペレーティングシステムが新しいマイナーリリースにアップグレードされないようにします。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **RhsmVars** パラメーターが含まれるオーバークラウドのサブスクリプション管理用環境ファイルを編集します。通常、このファイルのデフォルト名は **rhsm.yml** です。
4. サブスクリプション管理の設定で、**rhsm\_release** パラメーターが含まれているかどうかを確認します。**rhsm\_release** パラメーターが存在しない場合は、追加して 9.2 に設定します。

```
parameter_defaults:
  RhsmVars:
    ...
    rhsm_username: "myusername"
    rhsm_password: "p@55w0rd!"
    rhsm_org_id: "1234567"
    rhsm_pool_ids: "1a85f9223e3d5e43013e3d6e8ff506fd"
    rhsm_method: "portal"
    rhsm_release: "9.2"
```

5. オーバークラウドのサブスクリプション管理用環境ファイルを保存します。
6. すべてのノードでオペレーティングシステムのバージョンを RHEL 9.2 にロックするタスクを含む Playbook を作成します。

```
$ cat > ~/set_release.yaml <<'EOF'
- hosts: all
  gather_facts: false
  tasks:
    - name: set release to 9.2
      command: subscription-manager release --set=9.2
      become: true
EOF
```

7. **set\_release.yaml** Playbook を実行します。

```
$ ansible-playbook -i ~/overcloud-deploy/<stack>/tripleo-ansible-inventory.yaml -f 25
~/set_release.yaml --limit <undercloud>, <Controller>, <Compute>
```

- **<stack>** をスタックの名前に置き換えます。
- **--limit** オプションを使用して、コンテンツをすべての RHOSP ノードに適用します。**<undercloud>**、**<Controller>**、**<Compute>** は、それらのノードを含む環境内の Ansible グループに置き換えます。これらのノードには別のサブスクリプションがある可能性があるため、Ceph Storage ノードに対してこの Playbook を実行しないでください。



#### 注記

手動でノードを特定のバージョンにロックするには、ノードにログインして **subscription-manager release** コマンドを実行します。

```
$ sudo subscription-manager release --set=9.2
```

## 1.3. RED HAT OPENSTACK PLATFORM リポジトリの更新

Red Hat OpenStack Platform (RHOSP) 17.1 を使用するようリポジトリを更新します。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **RhsmVars** パラメーターが含まれるオーバークラウドのサブスクリプション管理用環境ファイルを編集します。通常、このファイルのデフォルト名は **rhsm.yml** です。
4. サブスクリプション管理の設定で **rhsm\_repos** パラメーターを確認します。**rhsm\_repos** パラメーターで RHOSP 17.1 リポジトリを使用している場合は、リポジトリを正しいバージョンに変更します。

```
parameter_defaults:
  RhsmVars:
    rhsm_repos:
      - rhel-9-for-x86_64-baseos-eus-rpms
      - rhel-9-for-x86_64-appstream-eus-rpms
      - rhel-9-for-x86_64-highavailability-eus-rpms
      - openstack-17.1-for-rhel-9-x86_64-rpms
      - fast-datapath-for-rhel-9-x86_64-rpms
```

5. オーバークラウドのサブスクリプション管理用環境ファイルを保存します。
6. すべてのノードで、リポジトリを RHOSP 17.1 に設定するタスクを含む Playbook を作成します。

```
$ cat > ~/update_rhosp_repos.yaml <<'EOF'
- hosts: all
gather_facts: false
tasks:
  - name: change osp repos
    command: subscription-manager repos --enable=openstack-17.1-for-rhel-9-x86_64-rpms
    become: true
EOF
```

7. **update\_rhosp\_repos.yaml** Playbook を実行します。

```
$ ansible-playbook -i ~/overcloud-deploy/<stack>/tripleo-ansible-inventory.yaml -f 25
~/update_rhosp_repos.yaml --limit <undercloud>,<Controller>,<Compute>
```

- **<stack>** をスタックの名前に置き換えます。
  - **--limit** オプションを使用して、コンテンツをすべての RHOSP ノードに適用します。**<undercloud>**、**<Controller>**、**<Compute>** は、それらのノードを含む環境内の Ansible グループに置き換えます。通常、Ceph Storage ノードは異なるサブスクリプションを使用するため、この Playbook を Ceph Storage ノードに対して実行しないでください。
8. すべての Ceph Storage ノードで、リポジトリを RHOSP 17.1 に設定するタスクが含まれる Playbook を作成します。

```
$ cat > ~/update_ceph_repos.yaml <<'EOF'
- hosts: all
gather_facts: false
```

```

tasks:
  - name: change ceph repos
    command: subscription-manager repos --enable=openstack-17.1-deployment-tools-for-
rhel-9-x86_64-rpms
    become: true
EOF

```

9. `update_ceph_repos.yaml` Playbook を実行します。

```

$ ansible-playbook -i ~/overcloud-deploy/<stack>/tripleo-ansible-inventory.yaml -f 25
~/update_ceph_repos.yaml --limit CephStorage

```

`--limit` オプションを使用して、コンテンツを Ceph Storage ノードに適用します。

## 1.4. コンテナイメージ準備ファイルの更新

コンテナ準備ファイルは、`ContainerImagePrepare` パラメーターが含まれるファイルです。このファイルを使用して、アンダークラウドおよびオーバークラウドのコンテナイメージを取得する際のルールを定義します。

環境を更新する前に、ファイルを確認して正しいイメージバージョンを取得するようにしてください。

### 手順

1. コンテナ準備ファイルを編集します。通常、このファイルのデフォルト名は `containers-prepare-parameter.yaml` です。
2. 各ルールセットの `tag` パラメーターが `17.1` に設定されていることを確認します。

```

parameter_defaults:
  ContainerImagePrepare:
    - push_destination: true
    set:
      ...
      tag: '17.1'
      tag_from_label: '{version}-{release}'

```



### 注記

更新に特定のタグ (`17.1` や `17.1.1` など) を使用しない場合は、`tag` キーと値のペアを削除し、`tag_from_label` のみを指定します。これにより、更新プロセスの一部として使用するタグの値を決定する際に、インストールされた Red Hat OpenStack Platform バージョンが使用されます。

3. このファイルを保存します。

## 1.5. オーバークラウドでのフェンシングの無効化

オーバークラウドを更新する前に、フェンシングが無効になっていることを確認します。

コントローラーノードの更新プロセス中にフェンシングが環境にデプロイされると、オーバークラウドは特定ノードが無効であることを検出し、フェンシング操作を試みる場合があります。これにより、意図しない結果が生じる可能性があります。

オーバークラウドでフェンシングを有効にした場合、更新中はフェンシングを一時的に無効にする必要があります。

## 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 各コントローラーノードにログインし、Pacemaker コマンドを実行してフェンシングを無効にします。

```
$ ssh tripleo-admin@<controller_ip> "sudo pcs property set stonith-enabled=false"
```

- **<controller\_ip>** は、コントローラーノードの IP アドレスに置き換えます。コントローラーノードの IP アドレスは、**/etc/hosts** または **/var/lib/mistral** で確認できます。
4. **fencing.yaml** 環境ファイルで、**EnableFencing** パラメーターを **false** に設定し、更新プロセス中にフェンシングが無効のままとなるようにします。

## 関連情報

- [STONITH を使用したコントローラーノードのフェンシング](#)

## 第2章 アンダークラウドの更新

director を使用して、アンダークラウドノード上の主要なパッケージを更新します。アンダークラウドとそのオーバークラウドイメージを最新の Red Hat OpenStack Platform (RHOSP) 17.1 バージョンに更新するには、以下の手順を実行します。

1. [「コンテナ化されたアンダークラウドのマイナー更新を実施する」](#)
2. [「オーバークラウドイメージの更新」](#)

### 前提条件

- アンダークラウドを最新の RHOSP 17.1 バージョンに更新する前に、すべての更新準備手順を完了している。詳細は、[1章 マイナー更新の準備](#) を参照してください。

### 2.1. アンダークラウドの更新前の RHOSP の検証

Red Hat OpenStack Platform (RHOSP) 環境を更新する前に、**tripleo-validations** Playbook を使用してアンダークラウドを検証してください。

検証の詳細は、[director を使用した Red Hat OpenStack Platform のインストールと管理の 検証フレームワークの使用](#) を参照してください。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 検証用にパッケージをインストールします。

```
$ sudo dnf -y update openstack-tripleo-validations python3-validations-libs validations-common
```

4. 検証を実行します。

```
$ validation run -i ~/overcloud-deploy/<stack>/tripleo-ansible-inventory.yaml --group pre-update
```

- <stack> をスタックの名前に置き換えます。

### 検証

1. 検証レポートの結果を表示するには、[director を使用した Red Hat OpenStack Platform のインストールと管理の 検証履歴の表示](#) を参照してください。





## 注記

検証の実行時にホストが見つからない場合、コマンドはステータスを **SKIPPED** と報告します。**SKIPPED** のステータスは、検証が実行されないことを意味しますが、これは想定内です。さらに、検証の合格基準が満たされていない場合、コマンドはステータスを **FAILED** と報告します。検証が **FAILED** であっても、更新された RHOSP 環境を使用できないことはありません。ただし、検証が **FAILED** の場合は、環境に問題があることを示している可能性があります。

## 2.2. SSH キーのサイズの検証

Red Hat Enterprise Linux (RHEL) 9.1 以降では、最小 2048 ビットの SSH キーサイズが必要です。Red Hat OpenStack Platform (RHOSP) director 上の現在の SSH キーが 2048 ビット未満の場合、オーバークラウドにアクセスできなくなる可能性があります。SSH キーが必要なビットサイズを満たしていることを確認する必要があります。

### 手順

1. SSH キーのサイズを検証します。

```
ssh-keygen -l -f ~/.ssh/id_rsa.pub
```

出力例:

```
1024 SHA256:Xqz0Xz0/aJua6B3qRD7VsLr6n/V3zhmnGskcFR6FIJw
stack@director.example.local (RSA)
```

2. SSH キーが 2048 ビット未満の場合は、次に進む前に、SSH キーをローテーションする必要があります。詳細は、[Red Hat OpenStack Platform の強化の OpenStack 環境での SSH キーの更新](#) を参照してください。

## 2.3. コンテナ化されたアンダークラウドのマイナー更新を実施する

director では、アンダークラウドノード上の主要なパッケージを更新するためのコマンドが提供されています。director を使用して、RHOSP 環境の現在のバージョン内でマイナー更新を実行します。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **dnfupdate** コマンドを使用して director メインパッケージを更新します。

```
$ sudo dnf update -y python3-tripleoclient ansible-*
```

4. アンダークラウド環境を更新します。

```
$ openstack undercloud upgrade
```

5. アンダークラウドの更新プロセスが完了するまで待ちます。

6. アンダークラウドをリブートして、オペレーティングシステムのカーネルとその他のシステムパッケージを更新します。

```
$ sudo reboot
```

7. ノードがブートするまで待ちます。

## 2.4. オーバークラウドイメージの更新

director がノードをイントロスペクトして最新バージョンの RHOSP ソフトウェアでプロビジョニングできるようにするには、現在のオーバークラウドイメージを新しいバージョンに置き換える必要があります。

### 前提条件

- アンダークラウドノードが最新バージョンに更新されている。詳細は、[「コンテナ化されたアンダークラウドのマイナー更新を実施する」](#) を参照してください。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **stack** ユーザーのホーム下の **images** ディレクトリー (**/home/stack/images**) から既存のイメージを削除します。

```
$ rm -rf ~/images/*
```

4. アーカイブをデプロイメントします。

```
cd ~/images
for i in /usr/share/rhosp-director-images/ironic-python-agent-latest-17.1.tar /usr/share/rhosp-director-images/overcloud-hardened-uefi-full-latest-17.1.tar; do tar -xvf $i; done
cd ~
```

5. director に最新のイメージをインポートします。

```
$ openstack overcloud image upload --update-existing --image-path /home/stack/images/
```

6. ノードが新しいイメージを使用するように設定します。

```
$ openstack overcloud node configure $(openstack baremetal node list -c UUID -f value)
```

7. 新規イメージが存在することを確認します。

```
$ ls -l /var/lib/ironic/httpboot /var/lib/ironic/images
```



## 重要

- オーバークラウドノードをデプロイする際には、オーバークラウドイメージのバージョンが該当する heat テンプレートバージョンに対応している状態にします。たとえば、RHOSP 17.1 heat テンプレートでは RHOSP 17.1 イメージのみを使用します。
- Red Hat カスタマーポータルまたは Red Hat Satellite Server を使用する接続環境をデプロイした場合、オーバークラウドのイメージとパッケージリポジトリのバージョンが同期していない可能性があります。オーバークラウドのイメージとパッケージリポジトリのバージョンが一致していることを確認するには、**virt-customize** ツールを使用できます。詳細は、Red Hat ナレッジベースで [Modifying the Red Hat Linux OpenStack Platform Overcloud Image with virt-customize](#) のソリューションを参照してください。
- 新しい **overcloud-full** イメージは、古い **overcloud-full** イメージを置き換えます。古いイメージに変更を加えた場合、特に今後新規ノードをデプロイする場合は、新しいイメージで変更を繰り返す必要があります。

## 第3章 オーバークラウドの更新

アンダークラウドを更新したら、オーバークラウドとコンテナイメージの準備コマンドを実行し、ノードを更新することで、オーバークラウドを更新できます。コントロールプレーン API は、マイナー更新中もすべて利用できます。

### 前提条件

- アンダークラウドノードが最新バージョンに更新されている。詳細は、[2章アンダークラウドの更新](#)を参照してください。
- **stack** ユーザーのホームディレクトリーでコアテンプレートのローカルセットを使用する場合は、テンプレートを更新し、**Red Hat OpenStack Platform デプロイメントのカスタマイズガイド**の [heat テンプレートについて](#) で推奨されるワークフローを使用してください。オーバークラウドをアップグレードする前に、ローカルコピーを更新する必要があります。
- **GlanceApiInternal** サービスをコントローラーロールに追加します。

```
OS::TripleO::Services::GlanceApiInternal
```

これは、Image サービス (glance) API の内部インスタンスのサービスで、位置データを管理者や、Block Storage サービス (cinder) や Compute サービス (nova) などの位置データを必要とする他のサービスに提供します。

### 手順

オーバークラウドを更新するには、次の手順を実行する必要があります。

1. [「オーバークラウドの更新準備タスクの実施」](#)
2. [「コンテナイメージ準備タスクの実行」](#)
3. [「オプション: すべてのオーバークラウドサーバーでの ovn-controller コンテナの更新」](#)
4. [「Pacemaker 制御サービスのコンテナイメージ名の更新」](#)
5. [「すべてのコントローラーノードを更新する」](#)
6. [「すべての Compute ノードを更新する」](#)
7. [「すべての HCI Compute ノードを更新する」](#)
8. [「すべての DistributedComputeHCI ノードの更新」](#)
9. [「すべての Ceph Storage ノードの更新」](#)
10. [「Red Hat Ceph Storage クラスターの更新」](#)
11. [「データベースのオンライン更新の実施」](#)
12. [「オーバークラウドでのフェンシングの再有効化」](#)

### 3.1. オーバークラウドの更新準備タスクの実施

更新プロセスに向けてオーバークラウドを準備するには、**openstack overcloud update prepare** コマンドを実行する必要があります。このコマンドは、オーバークラウドプランを Red Hat OpenStack Platform (RHOSP) 17.1 に更新して、更新用にノードを準備します。

## 前提条件

- Ceph のサブスクリプションを使用し、Ceph ストレージノード用に **overcloud-minimal** イメージを使用するように **director** を設定している場合、**roles\_data.yaml** ロール定義ファイルで **rhsm\_enforce** パラメーターが **False** に設定されていることを確認する。
- カスタム NIC テンプレートをレンダリングした場合は、オーバークラウドのバージョンとの非互換性を回避するために、**openstack-tripleo-heat-templates** コレクションの更新バージョンでテンプレートを再生成する。カスタム NIC テンプレートの詳細は、**director** を使用した **Red Hat OpenStack Platform のインストールと管理** ガイドの [カスタムネットワークインターフェイステンプレートの定義](#) を参照してください。



## 注記

OVN デプロイメントを使用する分散コンピューティングノード (エッジ) アーキテクチャーの場合は、[すべてのオーバークラウドサーバーでの ovn-controller コンテナの更新](#) セクションに進む前に、Compute、DistributedCompute、または DistributedComputeHCI ノードを含むスタックごとにこの手順を完了する必要があります。

## 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 更新準備コマンドを実行します。

```
$ openstack overcloud update prepare \
  --templates \
  --stack <stack_name> \
  -r <roles_data_file> \
  -n <network_data_file> \
  -e <environment_file> \
  -e <environment_file> \
  ...
```

以下のオプションの中で、お使いの環境に適切なオプションを追加します。

- オーバークラウドスタックの名前がデフォルトの名前 **overcloud** とは異なる場合は、更新の準備コマンドに **--stack** オプションを追加し、**<stack\_name>** を実際のスタック名に置き換えます。
  - 独自のカスタムロールを使用する場合は、**-r** オプションを使用してカスタムロール (**<roles\_data\_file>**) ファイルを含めます。
  - カスタムネットワークを使用する場合は、**-n** オプションを使用して、設定可能なネットワークを (**<network\_data\_file>**) ファイルに含めます。
  - 高可用性クラスターをデプロイする場合は、更新の準備コマンドに **--ntp-server** オプションを追加するか、環境ファイルに **NtpServer** パラメーターおよび値を追加します。
  - **-e** オプションを使用して、カスタム設定環境ファイルを含めます。
4. 更新の準備プロセスが完了するまで待ちます。

## 3.2. コンテナイメージ準備タスクの実行

オーバークラウドを更新する前に、環境に必要なすべてのコンテナイメージ設定を準備して、最新の RHOSP 17.1 コンテナイメージをアンダークラウドにプルする必要があります。

コンテナイメージの準備を完了するには、**container\_image\_prepare** タグの付いたタスクに対して **openstack overcloud external-update run** コマンドを実行する必要があります。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **container\_image\_prepare** タグの付いたタスクに対して **openstack overcloud external-update run** コマンドを実行します。

```
$ openstack overcloud external-update run --stack <stack_name> --tags
container_image_prepare
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。

## 3.3. オプション: すべてのオーバークラウドサーバーでの OVN-CONTROLLER コンテナの更新

Modular Layer 2 Open Virtual Network メカニズムドライバー (ML2/OVN) を使用してオーバークラウドをデプロイした場合は、**ovn-controller** コンテナを最新の RHOSP 17.1 バージョンに更新します。更新は、**ovn-controller** コンテナを実行するすべてのオーバークラウドサーバーで行われます。

- 次の手順では、コントローラーのロールが割り当てられているサーバーの **ovn-northd** サービスを更新する前に、コンピューターのロールが割り当てられているサーバーの **ovn-controller** コンテナを更新します。
- 分散型コンピューターノード (エッジ) アーキテクチャーの場合は、[すべてのコントローラーノードの更新](#) セクションに進む前に、Compute、DistributedCompute、または DistributedComputeHCI ノードを含むスタックごとにこの手順を完了する必要があります。この手順を実行する前に誤って **ovn-northd** サービスを更新した場合、仮想マシンに接続したり、新しい仮想マシンや仮想ネットワークを作成したりできない可能性があります。次の手順で接続を復元します。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **ovn** タグを持つタスクに対して **openstack overcloud external-update run** コマンドを実行します。

```
$ openstack overcloud external-update run --stack <stack_name> --tags ovn
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。

4. **ovn-controller** コンテナの更新が完了するまで待ちます。

### 3.4. PACEMAKER 制御サービスのコンテナイメージ名の更新

システムを Red Hat Openstack Platform (RHOSP) 17 から RHOSP 17.1 に更新する場合は、Pacemaker で制御されるサービスのコンテナイメージ名を更新する必要があります。Pacemaker 制御サービスの新しいイメージ命名スキーマに移行するには、この更新を実行する必要があります。

システムを RHOSP 17.1 のバージョンから最新バージョンの RHOSP 17.1 に更新する場合、Pacemaker が制御するサービスのコンテナイメージ名を更新する必要はありません。

#### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **ha\_image\_update** タグを指定して **openstack overcloud external-update run** コマンドを実行します。

```
$ openstack overcloud external-update run --stack <stack_name> --tags ha_image_update
```

- アンダークラウドスタックの名前がデフォルトのスタック名 **undercloud** と異なる場合は、**--stack** オプションを使用してスタック名を設定し、**<stack\_name>** をスタックの名前に置き換えます。

### 3.5. すべてのコントローラーノードを更新する

すべてのコントローラーノードを最新の RHOSP 17.1 バージョンに更新します。**--limit Controller** オプションを指定して **openstack overcloud update run** コマンドを実行し、操作をコントローラーノードのみに制限します。コントロールプレーン API は、マイナー更新中もすべて利用できます。

#### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 更新コマンドを実行します。

```
$ openstack overcloud update run --stack <stack_name> --limit Controller
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。

4. コントローラーノードの更新が完了するまで待ちます。

### 3.6. PACEMAKER 以外のサービスでのコンポーザブルロールの更新

Pacemaker 以外のサービスでコンポーザブルロールを最新の RHOSP 17.1 バージョンに更新します。各コンポーザブルロールのノードを1つずつ更新します。

#### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 更新コマンドを実行します。

```
$ openstack overcloud update run --stack <stack_name> --limit <non_pcs_role_0>
$ openstack overcloud update run --stack <stack_name> --limit <non_pcs_role_1>
$ openstack overcloud update run --stack <stack_name> --limit <non_pcs_role_2>
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。
- **<non\_pcs\_role\_0>**, **<non\_pcs\_role\_1>**, and **<non\_pcs\_role\_2>** を、Pacemaker 以外のサービスと共にコンポーザブルロールの名前に置き換えます。

4. 更新が完了するまで待ちます。

### 3.7. すべての COMPUTE ノードを更新する

すべての Compute ノードを最新の RHOSP 17.1 バージョンに更新します。Compute ノードを更新するには、**openstack overcloud update run** コマンドに **--limit Compute** オプションを指定して、操作を Compute ノードのみに制限して実行する必要があります。

#### 並列処理に関する考慮事項

多数の Compute ノードを更新する場合には、パフォーマンス向上のため、バックグラウンドで複数の更新タスクを実行し、ノードが 20 個含まれる別個のグループを更新するように各タスクを設定できます。たとえば、デプロイメントに 80 の Compute ノードがある場合、次のコマンドを実行して、Compute ノードを並行して更新できます。

```
$ openstack overcloud update run -y --limit 'Compute[0:19]' > update-compute-0-19.log 2>&1 &
$ openstack overcloud update run -y --limit 'Compute[20:39]' > update-compute-20-39.log 2>&1 &
$ openstack overcloud update run -y --limit 'Compute[40:59]' > update-compute-40-59.log 2>&1 &
$ openstack overcloud update run -y --limit 'Compute[60:79]' > update-compute-60-79.log 2>&1 &
```

ノード領域分割方法はランダムで、更新されるノードを制御することはできません。ノードの選択は、**tripleo-ansible-inventory** コマンドの実行時に生成するインベントリーファイルに基づきます。



特定の Compute ノードを更新するには、バッチで更新するノードをコンマ区切りリストで指定します。

```
$ openstack overcloud update run --limit <Compute0>,<Compute1>,<Compute2>,<Compute3>
```

## 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 更新コマンドを実行します。

```
$ openstack overcloud update run --stack <stack_name> --limit Compute
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。

4. Compute ノードの更新が完了するまで待ちます。

## 3.8. すべての HCI COMPUTE ノードを更新する

ハイパーコンバージドインフラストラクチャー (HCI) コンピュートノードを最新の RHOSP 17.1 バージョンに更新します。

### 前提条件

- **ceph-mon** サービスを実行している Ceph Monitor または Controller ノードで、Red Hat Ceph Storage クラスターのステータスが正常であり、pg ステータスが **active+clean** であることを確認する。

```
$ sudo cephadm -- shell ceph status
```

Ceph クラスターが正常な場合、**HEALTH\_OK** のステータスが返されます。

Ceph クラスターのステータスが異常な場合、**HEALTH\_WARN** または **HEALTH\_ERR** のステータスが返されます。トラブルシューティングのガイダンスは、[Red Hat Ceph Storage 5 トラブルシューティングガイド](#) または [Red Hat Ceph Storage 6 トラブルシューティングガイド](#) を参照してください。

## 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 更新コマンドを実行します。

-

```
$ openstack overcloud update run --stack <stack_name> --limit ComputeHCI
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。

4. ノードの更新が完了するまで待ちます。

### 3.9. すべての DISTRIBUTEDCOMPUTEHCI ノードの更新

分散コンピューターノードのアーキテクチャーに固有のロールを更新します。分散コンピューターノードをアップグレードするときは、まず **DistributedComputeHCI** ノードを更新し、その後 **DistributedComputeHCIScaleOut** ノードを更新します。

#### 前提条件

- **ceph-mon** サービスを実行している Ceph Monitor または Controller ノードで、Red Hat Ceph Storage クラスターのステータスが正常であり、pg ステータスが **active+clean** であることを確認する。

```
$ sudo cephadm -- shell ceph status
```

Ceph クラスターが正常な場合、**HEALTH\_OK** のステータスが返されます。

Ceph クラスターのステータスが異常な場合、**HEALTH\_WARN** または **HEALTH\_ERR** のステータスが返されます。トラブルシューティングのガイダンスは、[Red Hat Ceph Storage 5 トラブルシューティングガイド](#) または [Red Hat Ceph Storage 6 トラブルシューティングガイド](#) を参照してください。

#### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

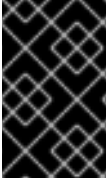
3. 更新コマンドを実行します。

```
$ openstack overcloud update run --stack <stack_name> --limit DistributedComputeHCI
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。
4. **DistributedComputeHCI** ノードの更新が完了するまで待ちます。
  5. 同じプロセスを使用して、**DistributedComputeHCIScaleOut** ノードを更新します。

### 3.10. すべての CEPH STORAGE ノードの更新

Red Hat Ceph Storage ノードを最新の RHOSP 17.1 バージョンに更新します。



## 重要

RHOSP 17.1 は RHEL 9.2 でサポートされています。ただし、Ceph Storage ロールにマップされているホストは、最新のメジャー RHEL リリースに更新されます。詳細は、[Red Hat Ceph Storage: サポートされる設定](#) を参照してください。

### 前提条件

- **ceph-mon** サービスを実行している Ceph Monitor または Controller ノードで、Red Hat Ceph Storage クラスターのステータスが正常であり、pg ステータスが **active+clean** であることを確認する。

```
$ sudo cephadm -- shell ceph status
```

Ceph クラスターが正常な場合、**HEALTH\_OK** のステータスが返されます。

Ceph クラスターのステータスが異常な場合、**HEALTH\_WARN** または **HEALTH\_ERR** のステータスが返されます。トラブルシューティングのガイダンスは、[Red Hat Ceph Storage 5 トラブルシューティングガイド](#) または [Red Hat Ceph Storage 6 トラブルシューティングガイド](#) を参照してください。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. 更新コマンドを実行します。

```
$ openstack overcloud update run --stack <stack_name> --limit CephStorage
```

- オーバークラウドスタックの名前がデフォルトのスタック名 **overcloud** と異なる場合は、スタック名を **--stack** オプションで設定し、**<stack\_name>** をスタックの名前に置き換えます。

4. ノードの更新が完了するまで待ちます。

## 3.11. RED HAT CEPH STORAGE クラスターの更新

**cephadm** コマンドを使用して、**director** を使用してデプロイされた Red Hat Ceph Storage クラスターを Red Hat OpenStack Platform (RHOSP) 17.1 と互換性のある最新バージョンに更新します。

次のいずれかのシナリオが使用中の環境に当てはまる場合は、Red Hat Ceph Storage クラスターを更新します。

- RHOSP 16.2 から RHOSP 17.1 にアップグレードした場合は、Red Hat Ceph Storage 5 を実行し、Red Hat Ceph Storage 5 の新しいバージョンに更新することになります。
- RHOSP 17.1 を新たにデプロイした場合は、Red Hat Ceph Storage 6 を実行し、Red Hat Ceph Storage 6 の新しいバージョンに更新することになります。

### 前提条件

- 「[コンテナイメージ準備タスクの実行](#)」でコンテナイメージの準備を完了します。

## 手順

1. コントローラーノードにログインします。
2. クラスターの正常性を確認します。

```
$ sudo cephadm shell -- ceph health
```



### 注記

Ceph Storage クラスターが正常な場合、コマンドは **HEALTH\_OK** の結果を返します。コマンドが別の結果を返す場合は、更新を続行する前にクラスターのステータスを確認し、Red Hat サポートに連絡してください。詳細は、[Red Hat Ceph Storage アップグレードガイド](#) の [cephadm を使用した Red Hat Ceph Storage クラスターのアップグレード](#) または [Red Hat Ceph Storage 6 アップグレードガイド](#) の [cephadm を使用した Red Hat Ceph Storage クラスターのアップグレード](#) を参照してください。

3. オプション: Ceph Storage クラスターの更新に含める必要があるイメージを確認します。

```
$ openstack tripleo container image list -f value | awk -F '/' '{print $2}'
```

4. クラスターを最新の Red Hat Ceph Storage バージョンに更新します。

```
$ sudo cephadm shell -- ceph orch upgrade start --image <image_name>: <version>
```

- **<image\_name>** を Ceph Storage クラスターイメージの名前に置き換えます。
- **<version>** を、Ceph Storage クラスターを更新するターゲットバージョンに置き換えます。

5. Ceph Storage コンテナの更新が完了するまで待ちます。更新ステータスを監視するには、次のコマンドを実行します。

```
sudo cephadm shell -- ceph orch upgrade status
```

## 3.12. データベースのオンライン更新の実施

一部のオーバークラウドコンポーネントでは、データベーステーブルのオンライン更新 (または移行) が必要です。オンラインでのデータベース更新を実行するには、**online\_upgrade** タグが付いたタスクに対して **openstack overcloud external-update run** コマンドを実行します。

データベースのオンライン更新は、次のコンポーネントに適用されます。

- OpenStack Block Storage (cinder)
- OpenStack Compute (nova)

## 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。

2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. **online\_upgrade** のタグを使用するタスクに対して **openstack overcloud external-update run** コマンドを実行します。

```
$ openstack overcloud external-update run --stack <stack_name> --tags online_upgrade
```

### 3.13. オーバークラウドでのフェンシングの再有効化

オーバークラウドをアップグレードする前に、[オーバークラウドでのフェンシングの無効化](#)の手順で、フェンシングを無効にしています。オーバークラウドを更新した後、ノードに障害が発生した場合にデータを保護するためにフェンシングを再度有効にします。

#### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

3. コントローラーノードにログインし、Pacemaker コマンドを実行してフェンシングを再度有効にします。

```
$ ssh tripleo-admin@<controller_ip> "sudo pcs property set stonith-enabled=true"
```

- **<controller\_ip>** は、コントローラーノードの IP アドレスに置き換えます。コントローラーノードの IP アドレスは、**openstack server list** コマンドで確認できます。

4. **fencing.yaml** 環境ファイルで、**EnableFencing** パラメーターを **true** に設定します。

#### 関連情報

- [STONITH を使用したコントローラーノードのフェンシング](#)

## 第4章 オーバークラウドの再起動

最新の 17.1 バージョンへの Red Hat OpenStack Platform (RHOSP) のマイナー更新実行後、オーバークラウドを再起動します。リブートにより、関連付けられたカーネル、システムレベル、およびコンテナコンポーネントの更新と共にノードがリフレッシュされます。これらの更新により、パフォーマンスとセキュリティ上のメリットが得られます。ダウンタイムを計画して、再起動手順を実施します。

以下のガイドを使用して、さまざまなノードのタイプを再起動する方法を説明します。

- 1つのロールで全ノードを再起動する場合は、各ノードを個別に再起動します。ロールの全ノードを同時に再起動すると、その操作中サービスにダウンタイムが生じる場合があります。
- 次の順序でノードの再起動の手順を完了します。
  1. 「[コントローラーノードおよびコンポーザブルノードの再起動](#)」
  2. 「[Ceph Storage \(OSD\) クラスターの再起動](#)」
  3. 「[Compute ノードの再起動](#)」

### 4.1. コントローラーノードおよびコンポーザブルノードの再起動

設定可能なロールに基づいて Controller ノードとスタンドアロンノードを再起動し、Compute ノードと Ceph ストレージノードを除外します。

#### 手順

1. リブートするノードにログインします。
2. オプション: ノードが Pacemaker リソースを使用している場合は、クラスターを停止します。

```
[tripleo-admin@overcloud-controller-0 ~]$ sudo pcs cluster stop
```

3. ノードをリブートします。

```
[tripleo-admin@overcloud-controller-0 ~]$ sudo reboot
```

4. ノードがブートするまで待ちます。

#### 検証

1. サービスが有効になっていることを確認します。
  - a. ノードが Pacemaker サービスを使用している場合は、ノードがクラスターに再度加わったか確認します。

```
[tripleo-admin@overcloud-controller-0 ~]$ sudo pcs status
```

- b. ノードが Systemd サービスを使用している場合は、すべてのサービスが有効化されていることを確認します。

```
[tripleo-admin@overcloud-controller-0 ~]$ sudo systemctl status
```

- c. ノードがコンテナ化されたサービスを使用している場合は、ノード上の全コンテナがアクティブであることを確認します。

```
[tripleo-admin@overcloud-controller-0 ~]$ sudo podman ps
```

## 4.2. CEPH STORAGE (OSD) クラスターの再起動

Ceph Storage (OSD) ノードのクラスターを再起動するには、以下の手順を実施します。

### 前提条件

- **ceph-mon** サービスを実行している Ceph Monitor または Controller ノードで、Red Hat Ceph Storage クラスターのステータスが正常であり、pg ステータスが **active+clean** であることを確認する。

```
$ sudo cephadm -- shell ceph status
```

Ceph クラスターが正常な場合、**HEALTH\_OK** のステータスが返されます。

Ceph クラスターのステータスが異常な場合、**HEALTH\_WARN** または **HEALTH\_ERR** のステータスが返されます。トラブルシューティングのガイダンスは、[Red Hat Ceph Storage 5 トラブルシューティングガイド](#) または [Red Hat Ceph Storage 6 トラブルシューティングガイド](#) を参照してください。

### 手順

1. **ceph-mon** サービスを実行している Ceph Monitor または Controller ノードにログインし、Ceph Storage クラスターのリバランスを一時的に無効にします。

```
$ sudo cephadm shell -- ceph osd set noout
$ sudo cephadm shell -- ceph osd set norebalance
```



### 注記

マルチスタックまたは分散コンピュートノード (DCN) アーキテクチャーを使用している場合は、**noout** フラグと **norebalance** フラグの設定時に Ceph クラスター名を指定する必要があります。例: **sudo cephadm shell -c /etc/ceph/<cluster>.conf -k /etc/ceph/<cluster>.client.keyring**。

2. 再起動する最初の Ceph Storage ノードを選択し、そのノードにログインします。
3. ノードをリブートします。

```
$ sudo reboot
```

4. ノードがブートするまで待ちます。
5. ノードにログインし、Ceph クラスターのステータスを確認します。

```
$ sudo cephadm -- shell ceph status
```

**pgmap** により、すべての **pgs** が正常な状態 (**active+clean**) として報告されることを確認します。

6. ノードからログアウトして、次のノードを再起動し、ステータスを確認します。全 Ceph Storage ノードが再起動されるまで、このプロセスを繰り返します。
7. 完了したら、**ceph-mon** サービスを実行している Ceph Monitor または Controller ノードにログインし、クラスターのリバランスを有効にします。

```
$ sudo cephadm shell -- ceph osd unset noout
$ sudo cephadm shell -- ceph osd unset norebalance
```



### 注記

マルチスタックまたは分散コンピュートノード (DCN) アーキテクチャーを使用している場合は、**noout** フラグと **norebalance** フラグの設定解除時に Ceph クラスター名を指定する必要があります。例: **sudo cephadm shell -c /etc/ceph/<cluster>.conf -k /etc/ceph/<cluster>.client.keyring**。

8. 最終のステータスチェックを実行して、クラスターが **HEALTH\_OK** を報告していることを確認します。

```
$ sudo cephadm shell ceph status
```

## 4.3. COMPUTE ノードの再起動

Red Hat OpenStack Platform (RHOSP) 環境でのインスタンスのダウンタイムを最小限に抑えるために、[インスタンスの移行ワークフロー](#)では、再起動するコンピュートノードからインスタンスを移行する時に必要な手順を概説します。

### インスタンスの移行ワークフロー

1. Compute ノードを再起動する前に、インスタンスを別のノードに移行するか決定します。
2. 再起動する Compute ノードを選択して無効にし、新規インスタンスをプロビジョニングしないようにします。
3. インスタンスを別の Compute ノードに移行します。
4. 空の Compute ノードを再起動します。
5. 空の Compute ノードを有効にします。

### 前提条件

- Compute ノードを再起動する前に、ノードの再起動中にインスタンスを別の Compute ノードに移行するか決定する。  
Compute ノード間で仮想マシンインスタンスを移行する際に発生する可能性のある移行の制約リストを確認する。詳細は、[インスタンス作成のための Compute サービスの設定の 移行の制約](#)を参照してください。





## 注記

Multi-RHEL 環境があり、RHEL 9.2 を実行しているコンピュータードから RHEL 8.4 を実行しているコンピュータードに仮想マシンを移行する場合は、コールドマイグレーションのみがサポートされます。コールドマイグレーションの詳細は、[インスタンス作成のための Compute サービスの設定の インスタンスのコールドマイグレーション](#) を参照してください。

- インスタンスを移行できない場合は、以下のコアテンプレートパラメーターを設定して、Compute ノード再起動後のインスタンスの状態を制御する。

### NovaResumeGuestsStateOnHostBoot

リブート後の Compute ノードで、インスタンスを同じ状態に戻すか定義します。**False** に設定すると、インスタンスは停止した状態を維持し、手動で起動する必要があります。デフォルト値は **False** です。

### NovaResumeGuestsShutdownTimeout

再起動する前に、インスタンスのシャットダウンを待つ秒数。この値を **0** に設定することは推奨されません。デフォルト値は **300** です。

オーバークラウドパラメーターおよびその使用方法の詳細は、[オーバークラウドのパラメーター](#) を参照してください。

## 手順

1. アンダークラウドに **stack** ユーザーとしてログインします。
2. コンピュータードのリストを取得して、再起動するノードのホスト名を特定します。

```
(undercloud)$ source ~/overcloudrc
(overcloud)$ openstack compute service list
```

再起動するコンピュータードのホスト名を特定します。

3. 再起動するコンピュータード上のコンピュータードサービスを無効にします。

```
(overcloud)$ openstack compute service list
(overcloud)$ openstack compute service set <hostname> nova-compute --disable
```

- **<hostname>** は、コンピュータードのホスト名に置き換えます。

4. Compute ノード上の全インスタンスをリスト表示します。

```
(overcloud)$ openstack server list --host <hostname> --all-projects
```

5. オプション: インスタンスを別のコンピュータードに移行するには、次の手順を実行します。
  - a. インスタンスを別の Compute ノードに移行する場合は、以下のコマンドのいずれかを使用します。

- インスタンスを別のホストに移行するには、次のコマンドを実行します。

```
(overcloud) $ openstack server migrate <instance_id> --live <target_host> --wait
```

- **<instance\_id>** は、インスタンス ID に置き換えます。

- **<target\_host>** は、インスタンスの移行先のホストに置き換えます。
- **nova-scheduler** がターゲットホストを自動的に選択できるようにします。

```
(overcloud) $ nova live-migration <instance_id>
```

- すべてのインスタンスを一度にライブマイグレーションします。

```
$ nova host-evacuate-live <hostname>
```



### 注記

**nova** コマンドで非推奨の警告が表示される可能性があります。無視しても問題ありません。

- 移行が完了するまで待ちます。
- 移行が正常に完了したことを確認します。

```
(overcloud) $ openstack server list --host <hostname> --all-projects
```

- Compute ノードのインスタンスがなくなるまで、移行を続けます。

6. コンピュートノードにログインして、ノードをリブートします。

```
[tripleo-admin@overcloud-compute-0 ~]$ sudo reboot
```

7. ノードがブートするまで待ちます。
8. Compute ノードを再度有効にします。

```
$ source ~/overcloudrc
(overcloud) $ openstack compute service set <hostname> nova-compute --enable
```

9. Compute ノードが有効であることを確認します。

```
(overcloud) $ openstack compute service list
```

## 4.4. オーバークラウドの更新後の RHOSP の検証

Red Hat OpenStack Platform (RHOSP) 環境を更新した後、**tripleo-validations** Playbook を使用してオーバークラウドを検証します。

検証の詳細は、**director** を使用した Red Hat OpenStack Platform のインストールと管理の [検証フレームワークの使用](#) を参照してください。

### 手順

1. アンダークラウドホストに **stack** ユーザーとしてログインします。
2. **stackrc** アンダークラウド認証情報ファイルを入手します。

```
$ source ~/stackrc
```

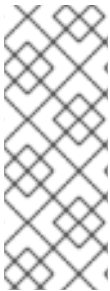
3. 検証を実行します。

```
$ validation run -i ~/overcloud-deploy/<stack>/tripleo-ansible-inventory.yaml --group post-update
```

- <stack> をスタックの名前に置き換えます。

## 検証

1. 検証レポートの結果を表示するには、**director** を使用した Red Hat OpenStack Platform のインストールと管理の [検証履歴の表示](#) を参照してください。



### 注記

検証の実行時にホストが見つからない場合、コマンドはステータスを **SKIPPED** と報告します。**SKIPPED** のステータスは、検証が実行されないことを意味しますが、これは想定内です。さらに、検証の合格基準が満たされていない場合、コマンドはステータスを **FAILED** と報告します。検証が **FAILED** であっても、更新された RHOSP 環境を使用できないことはありません。ただし、検証が **FAILED** の場合は、環境に問題があることを示している可能性があります。