



# Red Hat OpenStack Platform 17.1

## ハイパーコンバージドインフラストラクチャーの デプロイ

Red Hat OpenStack Platform オーバークラウドにおけるハイパーコンバージドイン  
フラストラクチャーの理解および設定



# Red Hat OpenStack Platform 17.1 ハイパーコンバージドインフラストラクチャーのデプロイ

---

Red Hat OpenStack Platform オーバークラウドにおけるハイパーコンバージドインフラストラクチャーの理解および設定

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

このガイドでは、Red Hat OpenStack Platform のハイパーコンバージェンスの実装について説明します。この実装では、Compute サービスと Ceph Storage サービスが同じホストに配置されます。

## 目次

多様性を受け入れるオープンソースの強化 .....	4
RED HAT ドキュメントへのフィードバック (英語のみ) .....	5
<b>第1章 RED HAT OPENSTACK PLATFORM ハイパーコンバージドインフラストラクチャーの設定およびデプロイ</b> .....	<b>6</b>
1.1. ハイパーコンバージドインフラストラクチャーの概要 .....	6
<b>第2章 HCI ハードウェアの導入</b> .....	<b>7</b>
2.1. CEPH STORAGE ノードのディスクのクリーニング .....	7
2.2. ノードの登録 .....	7
2.3. 利用可能な RED HAT CEPH STORAGE パッケージの確認 .....	10
2.4. HCI 環境用のソフトウェアイメージのデプロイ .....	10
2.5. HCI のノードの指定 .....	11
2.6. マルチディスク CEPH クラスターのルートディスクの定義 .....	14
<b>第3章 HCI 向けの RED HAT CEPH STORAGE クラスター名の設定</b> .....	<b>17</b>
3.1. デプロイメントの前提条件 .....	17
3.2. OPENSTACK OVERCLOUD CEPH DEPLOY コマンド .....	17
3.3. HCI の CEPH 設定のオーバーライド .....	17
3.4. 時刻同期の設定 .....	18
3.5. トップレベルドメイン接尾辞の設定 .....	20
3.6. RED HAT CEPH STORAGE クラスター名の設定 .....	20
3.7. ネットワークデータファイルを使用したネットワークオプションの設定 .....	21
3.8. 設定ファイルを使用したネットワークオプションの設定 .....	22
3.9. OSD の CRUSH 階層の設定 .....	23
3.10. CEPH サービス配置オプションの設定 .....	24
3.11. CEPH ノードの SSH ユーザーオプションの設定 .....	25
3.12. CEPH STORAGE コンテナへのアクセス .....	27
<b>第4章 HCI 用の RED HAT CEPH STORAGE クラスターのカスタマイズ</b> .....	<b>29</b>
4.1. 設定オプション .....	29
4.2. サービス仕様の生成 (オプション) .....	30
4.3. RED HAT OPENSTACK PLATFORM と RED HAT CEPH STORAGE の CEPH コンテナ .....	30
4.4. 高度な OSD 仕様の設定 .....	30
4.5. ノード固有のオーバーライドからの移行 .....	31
4.6. CEPH 伝送時暗号化の有効化 .....	31
<b>第5章 HCI のストレージサービスのカスタマイズ</b> .....	<b>32</b>
5.1. HCI のコンピューティングサービスリソースの設定 .....	32
5.2. カスタム環境ファイルの設定 .....	32
5.3. CEPH METADATA SERVER の有効化 .....	33
5.4. CEPH OBJECT GATEWAY オブジェクトストレージ .....	34
5.5. RED HAT OPENSTACK PLATFORM オブジェクトストレージのデプロイメントオプション .....	34
5.6. CEPH を使用するための BLOCK STORAGE BACKUP SERVICE の設定 .....	35
5.7. CEPH ノード向けの複数のボンディングされたインターフェイスの設定 .....	35
5.8. HCI のオーバークラウドデプロイメントの開始 .....	36
<b>第6章 HCI 設定の確認</b> .....	<b>38</b>
6.1. HCI 設定の確認 .....	38
<b>第7章 ハイパーコンバージドノードのスケールリング</b> .....	<b>39</b>
7.1. HCI 環境におけるハイパーコンバージドノードのスケールアップ .....	39
7.2. HCI 環境におけるハイパーコンバージドノードのスケールダウン .....	39

付録A 関連情報 .....	40
A.1. 設定ガイダンス	40
A.2. ハイパーコンバージドインフラストラクチャー環境の設定に関するガイドとリソース	41



## 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、用語の置き換えは、今後の複数のリリースにわたって段階的に実施されます。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 RED HAT OPENSTACK PLATFORM ハイパーコンバージドインフラストラクチャーの設定およびデプロイ

## 1.1. ハイパーコンバージドインフラストラクチャーの概要

Red Hat OpenStack Platform (RHOSP) ハイパーコンバージドインフラストラクチャー (HCI) は、ハイパーコンバージドノードで構成されます。RHOSP HCI では、リソースの使用を最適化するために、コンピューティングサービスとストレージサービスがこれらのハイパーコンバージドノードに配置されます。ハイパーコンバージドノードのみのオーバークラウド、またはハイパーコンバージドノードを通常のコンピュータードおよび Red Hat Ceph Storage ノードと混在させたオーバークラウドをデプロイすることが可能です。



### 注記

Red Hat Ceph Storage をストレージプロバイダーとして使用する必要があります。

### ヒント

BlueStore のメモリー処理機能を利用するには、BlueStore を HCI デプロイメントのバックエンドとして使用します。

ハイパーコンバージドインフラストラクチャーは、[director を使用した Red Hat Ceph および OpenStack のデプロイ](#) で説明されているデプロイメントプロセスのバリエーションを使用して構築されます。このデプロイメントシナリオでは、RHOSP director は、director がオーバークラウドと呼ぶクラウド環境と Red Hat Ceph Storage をデプロイします。Ceph クラスターは、オーバークラウド設定とは別に管理およびスケールリングします。



### 重要

インスタンス HA は、RHOSP HCI 環境ではサポートされません。RHOSP HCI 環境でインスタンス HA を使用するには、ComputeInstanceHA ロールを持つコンピュータードのサブセットで、インスタンス HA を使用するように指定する必要があります。Red Hat Ceph Storage サービスは、インスタンス HA をホストするコンピューティングノード上でホストしないでください。

Red Hat OpenStack Platform 17.1 は、新規デプロイメントのサポートは Red Hat Ceph Storage 6 のみです。Red Hat Ceph Storage 5 は、新しいデプロイメントシナリオではサポートされません。



### 重要

サポートされているハイパーコンバージドインフラストラクチャー環境内のすべての HCI ノードは、Red Hat OpenStack Platform コントローラーで使用されるバージョンと同じバージョンの Red Hat Enterprise Linux を使用する必要があります。同じハイパーコンバージドインフラストラクチャー環境内の HCI ノード上で複数の Red Hat Enterprise Linux バージョンをハイブリッド状態で使用する場合は、サポート例外について [Red Hat Customer Experience and Engagement](#) チームにご相談ください。

HCI 設定のガイダンスは、[設定ガイダンス](#) を参照してください。

## 第2章 HCI ハードウェアの導入

このセクションには、ハイパーコンバージドノードの準備と設定に関する手順と情報が含まれていません。

### 前提条件

- **director** を使用した Red Hat Ceph および OpenStack のデプロイの [オーバークラウド](#) および [Red Hat Ceph Storage のデプロイ](#) を確認している。

### 2.1. CEPH STORAGE ノードのディスクのクリーニング

Ceph Storage OSD とジャーナルパーティションには、ファクトリークリーンディスクが必要です。Ceph OSD サービスをインストールする前に、Bare Metal Provisioning サービス (ironic) によって、これらのディスクからすべてのデータとメタデータを消去する必要があります。

Bare Metal Provisioning サービスを使用して、デフォルトですべてのディスクデータとメタデータを削除するように、**director** を設定できます。**director** がこのタスクを実行するように設定されている場合、Bare Metal Provisioning サービスは、ノードが **available** に設定されるたびに、ノードを起動する追加の手順を実行します。



#### 警告

Bare Metal Provisioning サービスは、**wipefs --force --all** コマンドを使用します。このコマンドは、ディスク上のすべてのデータとメタデータを削除しますが、セキュアな消去は実行しません。Secure Erase には非常に長い時間がかかります。

### 手順

1. **/home/stack/undercloud.conf** を開き、次のパラメーターを追加します。

```
clean_nodes=true
```

2. **/home/stack/undercloud.conf** を保存します。
3. アンダークラウド設定を更新します。

```
openstack undercloud install
```

### 2.2. ノードの登録

**director** と通信できるように、ノードを登録します。

### 手順

1. **/home/stack** にノードインベントリ JSON ファイルを作成します。
2. ノードごとのハードウェアおよび電源管理の詳細を入力します。

以下に例を示します。

```
{
  "nodes":[
    {
      "mac":[
        "b1:b1:b1:b1:b1:b1"
      ],
      "cpu":"4",
      "memory":"6144",
      "disk":"40",
      "arch":"x86_64",
      "pm_type":"ipmi",
      "pm_user":"admin",
      "pm_password":"p@55w0rd!",
      "pm_addr":"192.0.2.205"
    },
    {
      "mac":[
        "b2:b2:b2:b2:b2:b2"
      ],
      "cpu":"4",
      "memory":"6144",
      "disk":"40",
      "arch":"x86_64",
      "pm_type":"ipmi",
      "pm_user":"admin",
      "pm_password":"p@55w0rd!",
      "pm_addr":"192.0.2.206"
    },
    {
      "mac":[
        "b3:b3:b3:b3:b3:b3"
      ],
      "cpu":"4",
      "memory":"6144",
      "disk":"40",
      "arch":"x86_64",
      "pm_type":"ipmi",
      "pm_user":"admin",
      "pm_password":"p@55w0rd!",
      "pm_addr":"192.0.2.207"
    },
    {
      "mac":[
        "c1:c1:c1:c1:c1:c1"
      ],
      "cpu":"4",
      "memory":"6144",
      "disk":"40",
      "arch":"x86_64",
      "pm_type":"ipmi",
      "pm_user":"admin",
      "pm_password":"p@55w0rd!",
      "pm_addr":"192.0.2.208"
    }
  ],
}
```

```
{
  "mac":[
    "c2:c2:c2:c2:c2:c2"
  ],
  "cpu":"4",
  "memory":"6144",
  "disk":"40",
  "arch":"x86_64",
  "pm_type":"ipmi",
  "pm_user":"admin",
  "pm_password":"p@55w0rd!",
  "pm_addr":"192.0.2.209"
},
{
  "mac":[
    "c3:c3:c3:c3:c3:c3"
  ],
  "cpu":"4",
  "memory":"6144",
  "disk":"40",
  "arch":"x86_64",
  "pm_type":"ipmi",
  "pm_user":"admin",
  "pm_password":"p@55w0rd!",
  "pm_addr":"192.0.2.210"
},
{
  "mac":[
    "d1:d1:d1:d1:d1:d1"
  ],
  "cpu":"4",
  "memory":"6144",
  "disk":"40",
  "arch":"x86_64",
  "pm_type":"ipmi",
  "pm_user":"admin",
  "pm_password":"p@55w0rd!",
  "pm_addr":"192.0.2.211"
},
{
  "mac":[
    "d2:d2:d2:d2:d2:d2"
  ],
  "cpu":"4",
  "memory":"6144",
  "disk":"40",
  "arch":"x86_64",
  "pm_type":"ipmi",
  "pm_user":"admin",
  "pm_password":"p@55w0rd!",
  "pm_addr":"192.0.2.212"
},
{
  "mac":[
    "d3:d3:d3:d3:d3:d3"
  ],
```

```

        "cpu":"4",
        "memory":"6144",
        "disk":"40",
        "arch":"x86_64",
        "pm_type":"ipmi",
        "pm_user":"admin",
        "pm_password":"p@55w0rd!",
        "pm_addr":"192.0.2.213"
    }
]
}

```

3. 新しいファイルを保存します。
4. スタックユーザーを初期化します。

```
$ source ~/stackrc
```

5. JSON インベントリーファイルを director にインポートし、ノードを登録する

```
$ openstack overcloud node import <inventory_file>
```

**<inventory\_file>** を最初の手順で作成したファイルの名前に置き換えます。

6. カーネルと ramdisk イメージを各ノードに割り当てます。

```
$ openstack overcloud node configure <node>
```

## 2.3. 利用可能な RED HAT CEPH STORAGE パッケージの確認

オーバークラウドのデプロイの失敗を回避するために、必要なすべてのパッケージが利用可能であることを確認します。

### 2.3.1. cephadm パッケージのインストールの確認

**cephadm** パッケージが1つ以上のオーバークラウドノードにインストールされていることを確認します。**cephadm** パッケージは、Ceph Storage クラスターの最初のノードをブートストラップするために使用されます。

**cephadm** パッケージは **overcloud-hardened-uefi-full.qcow2** イメージに含まれています。**tripleo\_cephadm** ロールは、Ansible パッケージモジュールを使用して、イメージ内に存在することを確認します。

## 2.4. HCI 環境用のソフトウェアイメージのデプロイ

HCI 環境用に設定されたノードは、**overcloud-hardened-uefi-full.qcow2** ソフトウェアイメージを使用する必要があります。このソフトウェアイメージを使用するには、Red Hat OpenStack Platform (RHOSP) サブスクリプションが必要です。

### 手順

1. **/home/stack/templates/overcloud-baremetal-deploy.yaml** ファイルを開きます。

2. **overcloud-hardened-uefi-full** イメージを必要とするノードの **image** プロパティを追加または更新します。特定のノードで使用するイメージ、または特定のロールを使用するすべてのノードで使用するイメージを設定できます。

### 特定のノード

```
- name: Ceph
  count: 3
  instances:
    - hostname: overcloud-ceph-0
      name: node00
      image:
        href: file:///var/lib/ironic/images/overcloud-minimal.qcow2
    - hostname: overcloud-ceph-1
      name: node01
      image:
        href: file:///var/lib/ironic/images/overcloud-hardened-uefi-full.qcow2
    - hostname: overcloud-ceph-2
      name: node02
      image:
        href: file:///var/lib/ironic/images/overcloud-hardened-uefi-full.qcow2
```

### 特定のロール用に設定されたすべてのノード

```
- name: ComputeHCI
  count: 3
  defaults:
    image:
      href: file:///var/lib/ironic/images/overcloud-hardened-uefi-full.qcow2
  instances:
    - hostname: overcloud-ceph-0
      name: node00
    - hostname: overcloud-ceph-1
      name: node01
    - hostname: overcloud-ceph-2
      name: node02
```

3. **roles\_data.yaml** ロール定義ファイルで、**rhsm\_enforce** パラメーターを **False** に設定します。

```
rhsm_enforce: False
```

4. プロビジョニングコマンドを実行します。

```
(undercloud)$ openstack overcloud node provision \
--stack overcloud \
--output /home/stack/templates/overcloud-baremetal-deployed.yaml \
/home/stack/templates/overcloud-baremetal-deploy.yaml
```

5. **overcloud-baremetal-deployed.yaml** 環境ファイルを **openstack overcloud deploy** コマンドに渡します。

## 2.5. HCI のノードの指定

HCI のノードを指定するには、新しいロールファイルを作成して **ComputeHCI** ロールを設定し、**ComputeHCI** のリソースクラスでベアメタルノードを設定する必要があります。

## 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. `source` コマンドで **stackrc** 認証情報ファイルを読み込みます。

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

3. **Controller** ロールと **ComputeHCI** ロールを含む、**roles\_data.yaml** という名前の新しいロールデータファイルを生成します。

```
(undercloud)$ openstack overcloud roles generate Controller ComputeHCI -o  
~/roles_data.yaml
```

4. **roles\_data.yaml** を開き、次のパラメーターとセクションがあることを確認します。

セクション/パラメーター	値
ロールのコメント	<b>Role: ComputeHCI</b>
ロール名	<b>name: ComputeHCI</b>
<b>description</b>	<b>HCI role</b>
<b>HostnameFormatDefault</b>	<b>%stackname%-novaceph-%index%</b>
<b>deprecated_nic_config_name</b>	<b>ceph.yaml</b>

5. オーバークラウド用の ComputeHCI ノードをノード定義のテンプレート **node.json** または **node.yaml** に追加して、そのノードを登録します。
6. ノードのハードウェアを検査します。

```
(undercloud)$ openstack overcloud node introspect --all-manageable --provide
```

7. カスタム HCI リソースクラスを使用して、HCI 用に指定する各ベアメタルノードにタグを付けます。

```
(undercloud)$ openstack baremetal node set \  
--resource-class baremetal.HCI <node>
```

**<node>** をベアメタルノードの ID に置き換えてください。

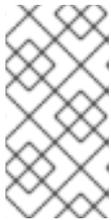
8. **ComputeHCI** ロールを **/home/stack/templates/overcloud-baremetal-deploy.yaml** ファイルに追加し、予測ノード配置、リソースクラス、またはノードに割り当てるその他の属性を定義します。

```
- name: Controller  
  count: 3
```

```
- name: ComputeHCI
  count: 1
  defaults:
    resource_class: baremetal.HCI
```

9. **baremetal.yaml** ファイルを開き、HCI に必要なネットワーク設定が含まれていることを確認します。以下に設定例を示します。

```
- name: ComputeHCI
  count: 3
  hostname_format: compute-hci-%index%
  defaults:
    profile: ComputeHCI
    network_config:
      template: /home/stack/templates/three-nics-vlans/compute-hci.j2
  networks:
    - network: ctlplane
      vif: true
    - network: external
      subnet: external_subnet
    - network: internalapi
      subnet: internal_api_subnet01
    - network: storage
      subnet: storage_subnet01
    - network: storage_mgmt
      subnet: storage_mgmt_subnet01
    - network: tenant
      subnet: tenant_subnet01
```



### 注記

**ComputeHCI** ロールのネットワーク設定には、**storage\_mgmt** ネットワークが含まれています。CephOSD ノードは、このネットワークを使用してデータの冗長コピーを作成します。**Compute** ロールのネットワーク設定には、このネットワークが含まれていません。

詳細は、[Bare Metal Provisioning サービスの設定](#) を参照してください。

10. プロビジョニングコマンドを実行します。

```
(undercloud)$ openstack overcloud node provision \
--stack overcloud \
--output /home/stack/templates/overcloud-baremetal-deployed.yaml \
/home/stack/templates/overcloud-baremetal-deploy.yaml
```

11. 別のターミナルでプロビジョニングの進捗をモニタリングします。

```
(undercloud)$ watch openstack baremetal node list
```



### 注記

デフォルトでは、**watch** コマンドは 2 秒ごとに更新されます。**-n** オプションは、更新タイマーを別の値に設定します。

12. **watch** プロセスを停止するには、**Ctrl-c** を入力します。
13. 検証: プロビジョニングが成功すると、ノードの状態が **available** から **active** に変わります。

## 関連情報

- ノードの登録の詳細は、**director** を使用した Red Hat OpenStack Platform のインストールと管理 ガイドの [オーバークラウドのノードの登録](#) を参照してください。
- ノードハードウェアの検査の詳細は、**director** を使用した Red Hat OpenStack Platform のインストールと管理 ガイドの [ベアメタルノードハードウェアのインベントリーの作成](#) を参照してください。
- **ComputeHCI** ロールおよび **storage\_mgmt** ネットワークのネットワーク設定の詳細は、[Bare Metal Provisioning サービスの設定](#) を参照してください。

## 2.6. マルチディスク CEPH クラスターのルートディスクの定義

Ceph Storage ノードは通常、複数のディスクを使用します。Director は、複数のディスク設定でルートディスクを識別する必要があります。オーバークラウドイメージは、プロビジョニングプロセス中にルートディスクに書き込まれます。

ハードウェアプロパティは、ルートディスクを識別するために使用されます。ルートディスクの識別に使用できるプロパティの詳細は、[ルートディスクを識別するプロパティ](#) を参照してください。

## 手順

1. 各ノードのハードウェアイントロスペクションからのディスク情報を確認します。

```
(undercloud)$ openstack baremetal introspection data save <node_uuid> --file
<output_file_name>
```

- **<node\_uuid>** をノードの UUID に置き換えます。
- **<output\_file\_name>** を、ノードイントロスペクションの出力を含むファイルの名前に置き換えます。  
たとえば、1つのノードのデータで3つのディスクが表示される場合があります。

```
[
  {
    "size": 299439751168,
    "rotational": true,
    "vendor": "DELL",
    "name": "/dev/sda",
    "wwn_vendor_extension": "0x1ea4dcc412a9632b",
    "wwn_with_extension": "0x61866da04f3807001ea4dcc412a9632b",
    "model": "PERC H330 Mini",
    "wwn": "0x61866da04f380700",
    "serial": "61866da04f3807001ea4dcc412a9632b"
  }
  {
    "size": 299439751168,
    "rotational": true,
    "vendor": "DELL",
    "name": "/dev/sdb",
```

```

    "wwn_vendor_extension": "0x1ea4e13c12e36ad6",
    "wwn_with_extension": "0x61866da04f380d001ea4e13c12e36ad6",
    "model": "PERC H330 Mini",
    "wwn": "0x61866da04f380d00",
    "serial": "61866da04f380d001ea4e13c12e36ad6"
  }
  {
    "size": 299439751168,
    "rotational": true,
    "vendor": "DELL",
    "name": "/dev/sdc",
    "wwn_vendor_extension": "0x1ea4e31e121cfb45",
    "wwn_with_extension": "0x61866da04f37fc001ea4e31e121cfb45",
    "model": "PERC H330 Mini",
    "wwn": "0x61866da04f37fc00",
    "serial": "61866da04f37fc001ea4e31e121cfb45"
  }
]

```

- 一意のハードウェアプロパティを使用して、ノードのルートディスクを設定します。

```
(undercloud)$ openstack baremetal node set --property root_device='{<property_value>}' <node-uuid>
```

- **<property\_value>** を、ルートディスクの設定に使用するイントロスペクションデータからの一意的ハードウェアプロパティ値に置き換えます。
- **<node\_uuid>** をノードの UUID に置き換えます。



### 注記

一意のハードウェアプロパティは、ディスクを一意的に識別するハードウェアイントロスペクションステップからの任意のプロパティです。たとえば、次のコマンドは、ディスクのシリアル番号を使用してルートディスクを設定します。

```
(undercloud)$ openstack baremetal node set --property
root_device='{"serial": "61866da04f380d001ea4e13c12e36ad6"}'
1a4e30da-b6dc-499d-ba87-0bd8a3819bc0
```

- 最初にネットワークから起動し、次にルートディスクから起動するように、各ノードの BIOS を設定します。

director は、ルートディスクとして使用する特定のディスクを把握します。**openstack overcloud node provision** コマンドを実行すると、director はオーバークラウドをプロビジョニングし、ルートディスクにオーバークラウドのイメージを書き込みます。

### 2.6.1. ルートディスクを識別するプロパティ

以下の属性を定義すると、director がルートディスクを特定するのに役立ちます。

- **model** (文字列): デバイスの ID
- **vendor** (文字列): デバイスのベンダー
- **serial** (文字列): ディスクのシリアル番号

- **hctl** (文字列): SCSI のホスト、チャンネル、ターゲット、Lun
- **size** (整数): デバイスのサイズ (GB 単位)
- **wwn** (文字列): 一意のストレージ ID
- **wwn\_with\_extension** (文字列): ベンダー拡張子を追加した一意のストレージ ID
- **wwn\_vendor\_extension** (文字列): 一意のベンダーストレージ ID
- **rotational** (ブール値): 回転式デバイス (HDD) には true、そうでない場合 (SSD) には false
- **name** (文字列): デバイス名 (例: /dev/sdb1)



### 重要

永続的な名前を持つデバイスには **name** プロパティを使用します。ノードの起動時に値が変更される可能性があるため、**name** プロパティを使用して永続的な名前を持たないデバイスのルートディスクを設定しないでください。

## 第3章 HCI 向けの RED HAT CEPH STORAGE クラスター名の設定

この章では、HCI 環境用に Red Hat Ceph Storage クラスターを設定およびデプロイする方法について説明します。

### 3.1. デプロイメントの前提条件

Red Hat Ceph Storage クラスターの設定とデプロイを試みる前に、以下が実行されていることを確認してください。

- Bare Metal Provisioning サービス (ironic) を使用したベアメタルインスタンスとそのネットワークのプロビジョニング。ベアメタルインスタンスのプロビジョニングに関する詳細は、[Bare Metal Provisioning サービスの設定](#) を参照してください。

### 3.2. OPENSTACK OVERCLOUD CEPH DEPLOY コマンド

director を使用して Ceph クラスターをデプロイする場合は、**openstack overcloud ceph deploy** コマンドを使用する必要があります。コマンドオプションおよびパラメーターの完全な一覧は、[コマンドラインインターフェイスリファレンス](#) の [openstack overcloud ceph deploy](#) を参照してください。

コマンド **openstack overcloud ceph deploy --help** は、環境で使用可能な現在のオプションとパラメーターを提供します。

### 3.3. HCI の CEPH 設定のオーバーライド

標準形式の初期化ファイルは、Ceph クラスター設定のオプションです。この初期化ファイルは、**cephadm bootstrap --config <file\_name>** または **openstack overcloud ceph deploy --config <file\_name>** コマンドで Ceph クラスターを設定するために使用されます。

ハイパーコンバージドノードに Ceph OSD と Compute サービスを配置すると、Red Hat Ceph Storage と Compute サービスの間でリソースの競合が発生するリスクがあります。これは、サービスがコリジョンを認識していないために発生します。リソースの競合によってサービスが低下し、ハイパーコンバージェンスの利点が相殺される可能性があります。

リソースの割り当ては、初期化ファイルを使用して調整し、リソースの競合を管理できます。以下では、**initial-ceph.conf** という初期化ファイルを作成し、**openstack overcloud ceph deploy** コマンドを使用して HCI デプロイメントを設定します。

```
$ cat <<EOF > initial-ceph.conf
[osd]
osd_memory_target_autotune = true
osd_numa_auto_affinity = true
[mgr]
mgr/cephadm/autotune_memory_target_ratio = 0.2
EOF
$ openstack overcloud ceph deploy --config initial-ceph.conf
```

**osd\_memory\_target\_autotune** オプションが **true** に設定されているため、OSD デーモンは **osd\_memory\_target** オプションに基づいてメモリー消費を調整します。**autotune\_memory\_target\_ratio** のデフォルトは **0.7** です。これは、システム内の合計 RAM の 70% が、自動調整されていない Ceph デーモンによって消費されるメモリーが差し引かれる開始点であることを示しています。次に、すべての OSD で **osd\_memory\_target\_autotune** が **true** に設定されていると仮定して、残りのメモリーを OSD で分割します。HCI デプロイメントの場

合、`mgr/cephadm/autotune_memory_target_ratio` を 0.2 に設定して、Compute サービスでより多くのメモリーを使用できるようにします。0.2 という値は慎重な開始点です。デプロイ後、必要に応じて `ceph` コマンドを使用してこの値を変更します。

2つの NUMA ノードシステムは、レイテンシーの影響を受けやすい Nova ワークロードを1つの NUMA ノードでホストし、Ceph OSD ワークロードを別の NUMA ノードでホストできます。Compute ワークロードで使用されていない特定の NUMA ノードを使用するように Ceph OSD を設定するには、次の Ceph OSD 設定のいずれかを使用します。

- `osd_numa_node` が numa ノードへのアフィニティーを設定する
- `osd_numa_auto_affinity` がストレージとネットワークが一致する NUMA ノードにアフィニティーを自動的に設定する

両方の NUMA ノードにネットワークインターフェイスがあり、ディスクコントローラーが NUMA ノード 0 である場合、ストレージネットワーク用に NUMA ノード 0 のネットワークインターフェイスを使用し、NUMA ノード 0 で Ceph OSD ワークロードをホストします。NUMA ノード 1 で Nova ワークロードをホストし、NUMA ノード 1 のネットワークインターフェイスを使用します。この設定を実現するには、`osd_numa_auto_affinity` を `true` に設定します。または、`osd_numa_node` を直接 0 に設定し、`osd_numa_auto_affinity` に値を設定せず、デフォルトで `false` に設定することもできます。

OSD がオフラインになった結果、ハイパーコンバージドクラスターがバックフィルされると、バックフィルプロセスが遅くなる可能性があります。回復が遅くなる代わりに、バックフィルアクティビティーは、 कोरोケーションされたコンピューティングワークロードに与える影響が少なくなります。Red Hat Ceph Storage には、バックフィルアクティビティーのレートを制御するための以下のデフォルトがあります。

- `osd_recovery_op_priority = 3`
- `osd_max_backfills = 1`
- `osd_recovery_max_active_hdd = 3`
- `osd_recovery_max_active_ssd = 10`



#### 注記

これらのデフォルトはデフォルト値であるため、初期化ファイルで渡す必要はありません。初期設定にデフォルト以外の値が必要な場合は、デプロイメント前に必要な値を使用して初期化ファイルに追加します。デプロイ後、コマンド `ceph config set osd` を使用します。

### 3.4. 時刻同期の設定

時刻同期サービス (chrony) は、デフォルトで時刻同期が有効になっています。以下のタスクを実行してサービスを設定できます。

- [区切りリストを使用した時刻同期の設定](#)
- [環境ファイルを使用した時刻同期の設定](#)
- [時刻同期の無効化](#)



## 注記

時刻同期は、区切り文字付きリストまたは環境ファイルを使用して設定されます。管理に最適な手順を使用してください。

### 3.4.1. 区切りリストを使用した時刻同期の設定

区切りリストを使用して NTP サーバーを設定するように時刻同期サービス (chrony) を設定できます。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. 区切りリストを使用して NTP サーバーを設定します。

```
openstack overcloud ceph deploy \
  --ntp-server "<ntp_server_list>"
```

**<ntp\_server\_list>** をコンマ区切りのサーバーのリストに置き換えます。

```
openstack overcloud ceph deploy \
  --ntp-server "0.pool.ntp.org,1.pool.ntp.org"
```

### 3.4.2. 環境ファイルを使用した時刻同期の設定

NTP サーバーを定義する環境ファイルを使用するように時刻同期サービス (chrony) を設定できます。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. **/home/stack/templates/ntp-parameters.yaml** などの環境ファイルを作成して、NTP サーバー設定を含めます。
3. **NtpServer** パラメーターを追加します。 **NtpServer** パラメーターには、NTP サーバーのコンマ区切りリストが含まれます。

```
parameter_defaults:
  NtpServer: 0.pool.ntp.org,1.pool.ntp.org
```

4. 環境ファイルを使用して NTP サーバーを設定します。

```
openstack overcloud ceph deploy \
  --ntp-heat-env-file "<ntp_file_name>"
```

**<ntp\_file\_name>** を、作成した環境ファイルの名前に置き換えます。

```
openstack overcloud ceph deploy \
  --ntp-heat-env-file "/home/stack/templates/ntp-parameters.yaml"
```

### 3.4.3. 時刻同期の無効化

時刻同期サービス (chrony) はデフォルトで有効になっています。サービスを使用したくない場合は、サービスを無効にすることができます。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. 時刻同期サービス (chrony) を無効にします。

```
openstack overcloud ceph deploy \
  --skip-ntp
```

### 3.5. トップレベルドメイン接尾辞の設定

トップレベルドメイン (TLD) 接尾辞を設定できます。この接尾辞は、短いホスト名に追加して、オーバークラウドノードの完全修飾ドメイン名を作成します。



#### 注記

TLS-e の設定には完全修飾ドメイン名が必要です。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. 最上位のドメイン接尾辞を設定します。

```
openstack overcloud ceph deploy \
  --tld "<domain_name>"
```

**<domain\_name>** は、必要なドメイン名に置き換えます。

```
openstack overcloud ceph deploy \
  --tld "example.local"
```

### 3.6. RED HAT CEPH STORAGE クラスター名の設定

設定した名前 Red Hat Ceph Storage クラスターをデプロイできます。デフォルト名は **ceph** です。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. 次のコマンドを使用して、Ceph Storage クラスターの名前を設定します。  
**openstack overcloud ceph deploy \ --cluster <cluster\_name>**

```
$ openstack overcloud ceph deploy \ --cluster central \
```



## 注記

キーリングファイルは、現時点では作成されません。キーリングファイルは、オーバークラウドのデプロイメント中に作成されます。キーリングファイルは、この手順で設定されたクラスター名を継承します。オーバークラウドのデプロイメントの詳細は、「[HCI のオーバークラウドデプロイメントの開始](#)」を参照してください。

上記の例では、Ceph クラスターの名前は **central** です。**central** の Ceph クラスターの設定ファイルとキーリングファイルは、デプロイプロセス中に **/etc/ceph** に作成されません。

```
[root@oc0-controller-0 ~]# ls -l /etc/ceph/
total 16
-rw-----. 1 root root 63 Mar 26 21:49 central.client.admin.keyring
-rw-----. 1 167 167 201 Mar 26 22:17 central.client.openstack.keyring
-rw-----. 1 167 167 134 Mar 26 22:17 central.client.radosgw.keyring
-rw-r--r--. 1 root root 177 Mar 26 21:49 central.conf
```

## トラブルシューティング

Ceph Storage クラスターのカスタム名を設定すると、次のエラーが表示される場合があります。

**monclient: get\_monmap\_and\_config cannot identify monitors to contact because**

このエラーが表示された場合は、Ceph のデプロイ後に次のコマンドを使用します。

**cephadm shell --config <configuration\_file> --keyring <keyring\_file>**

たとえば、クラスター名を **central** に設定したときにこのエラーが表示された場合は、次のコマンドを使用します。

```
cephadm shell --config /etc/ceph/central.conf \
--keyring /etc/ceph/central.client.admin.keyring
```

代わりに次のコマンドを使用することもできます。

```
cephadm shell --mount /etc/ceph:/etc/ceph
export CEPH_ARGS='--cluster central'
```

## 3.7. ネットワークデータファイルを使用したネットワークオプションの設定

ネットワークデータファイルには、Red Hat Ceph Storage クラスターが使用するネットワークが記述されています。

### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. **network\_data.yaml** というカスタムネットワーク属性を定義する YAML 形式のファイルを作成します。



### 重要

ネットワーク分離を使用すると、標準のネットワークデプロイメントは、2つの Ceph ネットワークにマッピングされる2つのストレージネットワークで構成されます。

- ストレージネットワーク **storage** は、Ceph ネットワーク **public\_network** にマッピングされます。このネットワークは、コンピュータードから Ceph クラスタへの RBD トラフィックなどのストレージトラフィックを処理します。
- ストレージネットワーク **storage\_mgmt** は、Ceph ネットワーク **cluster\_network** にマッピングされます。このネットワークは、Ceph OSD 間のデータレプリケーションなどのストレージ管理トラフィックを処理します。

3. **openstack overcloud ceph deploy** コマンドと **--config** オプションを使用して、設定をデプロイします。

```
openstack overcloud ceph deploy \
  deployed_metal.yaml \
  -o deployed_ceph.yaml \
  --network-data network_data.yaml
```



### 重要

**openstack overcloud ceph deploy** コマンドは、**--network-data** オプションで指定されたネットワークデータファイルを使用して、**public\_network** および **cluster\_network** として使用するネットワークを決定します。コマンドは、**--public-network-name** および **--cluster-network-name** オプションで別の名前が指定されていない限り、これらのネットワークがネットワークデータファイルで **storage** および **storage\_mgmt** という名前であると想定します。

ネットワーク分離を使用してデプロイする場合は、**--network-data** オプションを使用する必要があります。このオプションを使用しない場合、デフォルトのアンダークラウド (192.168.24.0/24) が **public\_network** と **cluster\_network** の両方に使用されます。

## 3.8. 設定ファイルを使用したネットワークオプションの設定

ネットワークオプションは、ネットワークデータファイルの代わりに設定ファイルで指定できます。



### 重要

この方法を使用してネットワークオプションを設定すると、**network\_data.yaml** で自動的に生成された値が上書きされます。このネットワーク設定方法を使用する場合は、必ず4つの値をすべて設定してください。

### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. Ceph クラスタを設定する標準形式の初期化ファイルを作成します。他の設定オプションを含むファイルをすでに作成している場合は、それにネットワーク設定を追加できます。

3. ファイルの **[global]** セクションに次のパラメーターを追加します。

- **public\_network**
- **cluster\_network**
- **ms\_bind\_ipv4**



#### 重要

**public\_network** および **cluster\_network** が **storage** および **storage\_mgmt** と同じネットワークにマップされていることを確認します。

以下は、複数のサブネットとカスタムネットワーク名を持つネットワーク設定の設定ファイルエントリーの例です。

```
[global]
public_network = 172.16.14.0/24,172.16.15.0/24
cluster_network = 172.16.12.0/24,172.16.13.0/24
ms_bind_ipv4 = True
ms_bind_ipv6 = False
```

4. コマンド **openstack overcloud ceph deploy** を **--config** オプションとともに使用して、設定ファイルをデプロイします。

```
$ openstack overcloud ceph deploy \
--config initial-ceph.conf --network-data network_data.yaml
```

## 3.9. OSD の CRUSH 階層の設定

OSD デプロイメント中にカスタム Controlled Replication Under Scalable Hashing (CRUSH) 階層を設定して、OSD **location** 属性を Ceph Storage クラスター **hosts** 仕様に追加できます。**location** 属性は、OSD が CRUSH 階層内のどこに配置されるかを設定します。



#### 注記

**location** 属性は、最初の CRUSH ロケーションのみを設定します。その後の属性変更は無視されます。

#### 手順

1. アンダークラウドノードに **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. カスタム CRUSH 階層を定義する設定ファイルを作成します (例: **crush\_hierarchy.yaml**)。
4. そのファイルに以下の設定を追加します。

```
<osd_host>:
root: default
rack: <rack_num>
```

```
<osd_host>:
  root: default
  rack: <rack_num>
<osd_host>:
  root: default
  rack: <rack_num>
```

- **<osd\_host>** を、OSD がデプロイされているノードのホスト名 (**ceph-0** など) に置き換えます。
- **<rack\_num>** を、OSD がデプロイメントされているラックの番号 (**r0** など) に置き換えます。

5. カスタム OSD レイアウトを使用して Ceph クラスタをデプロイします。

```
openstack overcloud ceph deploy \
  deployed_metal.yaml \
  -o deployed_ceph.yaml \
  --osd-spec osd_spec.yaml \
  --crush-hierarchy crush_hierarchy.yaml
```

Ceph クラスタは、カスタム OSD レイアウトで作成されます。

上記のサンプルファイルは、次の OSD レイアウトになります。

ID	CLASS	WEIGHT	TYPE	NAME	STATUS	REWEIGHT	PRI-AFF
-1		0.02939	root	default			
-3		0.00980	rack	r0			
-2		0.00980	host	ceph-node-00			
0	hdd	0.00980	osd	osd.0	up	1.00000	1.00000
-5		0.00980	rack	r1			
-4		0.00980	host	ceph-node-01			
1	hdd	0.00980	osd	osd.1	up	1.00000	1.00000
-7		0.00980	rack	r2			
-6		0.00980	host	ceph-node-02			
2	hdd	0.00980	osd	osd.2	up	1.00000	1.00000



### 注記

デバイスクラスは Ceph によって自動的に検出されますが、CRUSH ルールはプールに関連付けられています。プールは、引き続きオーバークラウドのデプロイメント中に **CephCrushRules** パラメーターを使用して定義および作成されます。

### 関連情報

詳細は、Red Hat Ceph Storage インストールガイドの [Red Hat Ceph Storage ワークロードの考慮事項](#) を参照してください。

## 3.10. CEPH サービス配置オプションの設定

カスタムロールファイルを使用して、どのノードがどの Ceph サービスを実行するかを定義できます。カスタムロールファイルは、環境が理由でデフォルトのロール割り当てが使用されない場合にのみ必要です。たとえば、ハイパーコンバージドノードをデプロイする場合、事前にデプロイされたコンピュー

トノードは、サービスタイプが `osd` の `osd` としてラベル付けされ、コンピュートインスタンスのリストを含む配置リストを持つ必要があります。

`roles_data.yaml` ファイルのサービス定義は、どのベアメタルインスタンスがどのサービスを実行するかを決定します。デフォルトでは、Controller ロールには CephMon および CephMgr サービスがあり、CephStorage ロールには CephOSD サービスがあります。ほとんどのコンポーザブルサービスとは異なり、Ceph サービスでは、サービスの設定方法を決定するために `heat` 出力は必要ありません。デプロイされた Ceph プロセスが Heat の実行前に発生する場合でも、`roles_data.yaml` ファイルが常に Ceph サービスの配置を決定します。

## 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. カスタムロールを定義する YAML 形式のファイルを作成します。
3. 設定ファイルをデプロイします。

```
$ openstack overcloud ceph deploy \
  deployed_metal.yaml \
  -o deployed_ceph.yaml \
  --roles-data custom_roles.yaml
```

## 3.11. CEPH ノードの SSH ユーザーオプションの設定

`openstack overcloud ceph deploy` コマンドはユーザーとキーを作成し、それらをホストに配布するため、このセクションの手順を実行する必要はありません。ただし、これはサポート対象のオプションです。

Cephadm は、SSH を使用してすべての管理対象のリモート Ceph ノードに接続します。Red Hat Ceph Storage クラスターのデプロイメントプロセスにより、すべてのオーバークラウド Ceph ノードでアカウントと SSH キーのペアが作成されます。その後、鍵ペアが Cephadm に渡され、ノードと通信できるようになります。

### 3.11.1. Red Hat Ceph Storage クラスター作成前の SSH ユーザーの作成

`openstack overcloud ceph user enable` コマンドを使用して、Ceph クラスターを作成する前に SSH ユーザーを作成できます。

## 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. SSH ユーザーを作成します。

```
$ openstack overcloud ceph user enable <specification_file>
```

- `<specification_file>` を、ユーザーが作成され、公開 SSH キーがインストールされているクラスターを記述した Ceph 仕様ファイルのパスと名前に置き換えます。仕様ファイルは、どのノードを変更するか、また秘密鍵が必要かどうかを決定するための情報を提供します。仕様ファイルの作成の詳細は、[サービス仕様の生成](#) を参照してください。



### 注記

デフォルトのユーザー名は **ceph-admin** です。別のユーザー名を指定するには、**--cephadm-ssh-user** オプションを使用して別のユーザー名を指定します。

```
openstack overcloud ceph user enable --cephadm-ssh-user
<custom_user_name>
```

**--cephadm-ssh-user** パラメーターを使用せずに、デフォルト名を使用することが推奨されます。

ユーザーが事前に作成されている場合は、**openstack overcloud ceph deploy** を実行するときにパラメーター **--skip-user-create** を使用します。

### 3.11.2. SSH ユーザーの無効化

SSH ユーザーを無効にすると、**cephadm** が無効になります。**cephadm** を無効にすると、Ceph クラスタを管理するサービスの機能が削除され、関連するコマンドが機能しなくなります。また、Ceph ノードのオーバークラウドのスケーリング操作も防ぎます。また、すべての公開および秘密 SSH キーも削除されます。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. **openstack overcloud ceph user disable --fsid <FSID> <specification\_file>** コマンドを使用して、SSH ユーザーを無効にします。
  - **<FSID>** を、クラスタのファイルシステム ID に置き換えます。FSID は、そのクラスタの一意的識別子です。FSID は、**deployed\_ceph.yaml** 環境ファイルにあります。
  - **<specification\_file>** を、ユーザーが作成されたクラスタを記述した Ceph 仕様ファイルのパスと名前に置き換えます。



### 重要

**cephadm** を無効にする必要がある場合を除き、**openstack overcloud ceph user disable** コマンドは推奨されません。



### 重要

SSH ユーザーと Ceph オーケストレーターサービスを無効にした後に有効にするには、**openstack overcloud ceph user enable --fsid <FSID> <specification\_file>** コマンドを使用します。



## 注記

このコマンドでは、以下を判別するために Ceph 仕様ファイルへのパスが必要です。

- SSH ユーザーを必要とするホスト。
- `_admin` ラベルがあり、プライベート SSH キーを必要とするホスト。
- 公開 SSH キーを必要とするホスト。

仕様ファイルとその生成方法の詳細は、サービス仕様の生成を参照してください。

## 3.12. CEPH STORAGE コンテナへのアクセス

`director` を使用した Red Hat OpenStack Platform のインストールと管理ガイドの [コンテナイメージの準備](#) には、コンテナイメージを使用するためにレジストリーとアンダークラウドおよびオーバークラウド設定を準備する方法に関する手順と情報が記載されています。このセクションの情報をを使用して、これらの手順を Ceph Storage コンテナにアクセスするように調整します。

オーバークラウドから Ceph Storage コンテナにアクセスするには 2 つのオプションがあります。

- [リモートレジストリーからコンテナを直接ダウンロードする](#)
- [アンダークラウド上のコンテナのキャッシュ](#)

### 3.12.1. アンダークラウド上のコンテナのキャッシュ

[準備プロセスにおけるイメージの変更](#) 手順では、次のコマンドの使用について説明しています。

```
sudo openstack tripleo container image prepare \
  -e ~/containers-prepare-parameter.yaml \
```

[リモートレジストリーからのコンテナの直接ダウンロード](#) で説明されているように、`--container-image-prepare` オプションを使用しないで `openstack overcloud ceph deploy` コマンドに認証情報を提供し、リモートレジストリーから Ceph コンテナを直接ダウンロードする場合は、Ceph をデプロイする前に `sudo openstack tripleo container image prepare` コマンドを実行する必要があります。

### 3.12.2. リモートレジストリーからコンテナを直接ダウンロードする

リモートレジストリーからコンテナを直接ダウンロードするように Ceph を設定できます。

`cephadm` コマンドは、`containers-prepare-parameter.yaml` ファイルに設定されている認証情報を使用してリモートレジストリーに認証し、Red Hat Ceph Storage コンテナをダウンロードします。

## 手順

1. `director` を使用した Red Hat OpenStack Platform のインストールと管理ガイドの [コンテナイメージの準備](#) の手順を使用して、`containers-prepare-parameter.yaml` ファイルを作成します。

2. [プライベートレジストリーからコンテナイメージを取得する](#) の説明に従って、**ContainerImageRegistryCredentials** パラメーターを使用して、リモートレジストリーの認証情報を **containers-prepare-parameter.yaml** ファイルに追加します。
3. Ceph をデプロイするときは、**openstack overcloud ceph deploy** コマンドを使用して、**containers-prepare-parameter.yaml** ファイルを渡します。

```
openstack overcloud ceph deploy \  
--container-image-prepare containers-prepare-parameter.yaml
```



### 注記

[アンダークラウドでのコンテナのキャッシュ](#) で説明されているように、アンダークラウドでコンテナをキャッシュしない場合は、Ceph をデプロイするときに同じ **containers-prepare-parameter.yaml** ファイルを **openstack overcloud ceph deploy** コマンドに渡す必要があります。これにより、コンテナがアンダークラウドにキャッシュされます。

## 第4章 HCI 用の RED HAT CEPH STORAGE クラスターのカスタマイズ

Red Hat OpenStack Platform (RHOSP) director はデフォルト設定を使用して、コンテナ化された Red Hat Ceph Storage をデプロイします。Ceph Storage をカスタマイズするには、デフォルト設定を上書きします。

### 前提条件

- サーバーが展開され、そのストレージネットワークが設定されます。
- **openstack overcloud node provision -o ~/deployed\_metal.yaml ...** による出力としてデプロイされたベアメタルファイル。

### 4.1. 設定オプション

Red Hat Ceph Storage クラスターを設定するには、複数のオプションがあります。

#### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. オプション: 標準形式の初期化 (ini) ファイルを使用して Ceph クラスターを設定します。
  - a. 設定オプションを含むファイルを作成します。  
以下は、単純な設定ファイルの例です。

```
[global]
osd_crush_chooseleaf type = 0
log_file = /var/log/ceph/$cluster-$type.$id.log

[mon]
mon_cluster_log_to_syslog = true
```

- b. 設定ファイルを作成します。
- c. **openstack overcloud ceph deploy --config <configuration\_file\_name>** コマンドを使用して設定をデプロイします。  
<configuration\_file\_name> を作成したファイルの名前に置き換えます。

```
$ openstack overcloud ceph deploy --config initial-ceph.conf
```

3. オプション: 設定値を **cephadm bootstrap** コマンドに送信します。  
**openstack overcloud ceph deploy --force \ --cephadm-extra-args '<optional\_arguments>'** \<br><optional\_arguments> を設定値に置き換えて、基になるコマンドに提供します。



#### 注記

引数 **--log-to-file** および **--skip-prepare-host** を使用する場合は、コマンド **openstack overcloud ceph deploy --force \ --cephadm-extra-args '--log-to-file --skip-prepare-host'** \ が使用されます。

## 4.2. サービス仕様の生成 (オプション)

Red Hat Ceph Storage クラスターサービス仕様は、Ceph Storage サービスのデプロイメントを記述する YAML ファイルです。Ceph Storage クラスターがデプロイされる前に、**tripleo** によって自動的に生成されます。通常は、個別に生成する必要はありません。

カスタムサービス仕様を作成して、Red Hat Ceph Storage クラスターをカスタマイズできます。

### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。
2. 仕様ファイルを生成します。

```
openstack overcloud ceph spec deployed_metal.yaml -o <specification_file>
```

- **<specification\_file>** を現在のサービス仕様で生成するファイルの名前に置き換えます。



### 注記

**deployed\_metal.yaml** は、**openstack overcloud node provision** コマンドの出力から取得されます。

3. 生成されたファイルを編集して必要な設定を追加します。
4. カスタムサービス仕様をデプロイします。

```
openstack overcloud ceph deploy \
  deployed_metal.yaml \
  -o deployed_ceph.yaml \
  --ceph-spec <specification_file>
```

- **<specification\_file>** をカスタムサービス仕様ファイルの名前に置き換えます。

## 4.3. RED HAT OPENSTACK PLATFORM と RED HAT CEPH STORAGE の CEPH コンテナ

NFS Ganesha で Red Hat Ceph Storage を使用するように、Red Hat Openstack Platform (RHOSP) を設定するには、Ceph Storage コンテナが必要です。外部 Ceph Storage クラスターがブロック (RBD 経由)、オブジェクト (RGW 経由)、またはファイル (ネイティブ CephFS 経由) ストレージのみを提供する場合、Ceph Storage コンテナは不要です。

RHOSP 17.1 は Red Hat Ceph Storage 6.x (Ceph パッケージ 17.x) をデプロイします。Ceph Storage 6.x コンテナは、認証が必要なレジストリーである **registry.redhat.io** でホスティングされます。詳細は、[コンテナイメージ準備のパラメーター](#) を参照してください。

## 4.4. 高度な OSD 仕様の設定

デフォルトの仕様では、Ceph Storage クラスターに必要な機能が提供されない場合、高度な OSD 仕様を設定します。

### 手順

1. アンダークラウドノードに **stack** ユーザーとしてログインします。

- 高度な OSD 仕様を定義する YAML 形式のファイルを作成します。  
以下は、カスタム OSD 仕様の例です。

```
data_devices:
  rotational: 1
db_devices:
  rotational: 0
```

この例では、すべてのローテーションデバイスがデータデバイスになり、すべての非ローテーションデバイスが共有デバイスとして使用される OSD 仕様を作成します。動的 Ceph サービス仕様が構築されると、**service\_type** が **osd** の場合、仕様ファイルにあるものは、すべて仕様のセクションに追加されます。

- 仕様ファイルを保存します。
- 仕様をデプロイします。  
**openstack overcloud ceph deploy \ --osd-spec <osd\_specification\_file>**

**<osd\_specification\_file>** を作成した仕様ファイルの名前に置き換えます。

```
$ openstack overcloud ceph deploy \ --osd-spec osd_spec.yaml \
```

### 関連情報

サービス仕様で OSD を設定するために使用される OSD 関連の属性のリストは、**Red Hat Ceph Storage 操作ガイド** の [OSD をデプロイするための高度なサービス仕様およびフィルター](#) を参照してください。

## 4.5. ノード固有のオーバーライドからの移行

Red Hat OpenStack Platform 17.0 より前は、ノード固有のオーバーライドを使用して、同種ではないサーバーハードウェアを管理していました。これは、カスタム OSD 仕様ファイルで実行されるようになりました。カスタム OSD 仕様ファイルの作成方法は、[高度な OSD 仕様の設定](#) を参照してください。

## 4.6. CEPH 伝送時暗号化の有効化

メッセージバージョン 2 プロトコルの **secure mode** を使用して、すべての Ceph Storage トラフィックの暗号化を有効にします。Red Hat Ceph Storage **Red Hat OpenStack Platform** の [データ強化の暗号化とキー管理](#) の説明に従って Ceph Storage を設定し、Ceph オンワイヤー暗号化を有効にします。

### 関連情報

Ceph 伝送時暗号化に関する詳しい情報は、Red Hat Ceph Storage [アーキテクチャーガイド](#) の [Ceph on-wire encryption](#) を参照してください。

## 第5章 HCI のストレージサービスのカスタマイズ

Red Hat OpenStack Platform (RHOSP) director は、基本的な Ceph Storage 設定を有効にするために必要な heat テンプレートと環境ファイルを提供します。

Director は `/usr/share/openstack-tripleo-heat-templates/environments/cephadm/cephadm.yaml` 環境ファイルを使用して、`openstack overcloud ceph deploy` によってデプロイされた Ceph クラスタに追加設定を追加します。

RHOSP のコンテナ化されたサービスの詳細は、[director を使用した Red Hat OpenStack Platform のインストールと管理の CLI ツールを使用した基本的なオーバークラウドの設定](#) を参照してください。

### 5.1. HCI のコンピューティングサービスリソースの設定

ハイパーコンバージドノードに Ceph OSD と Compute サービスを配置すると、Red Hat Ceph Storage と Compute サービスの間でリソースの競合が発生するリスクがあります。これは、サービスがコロケーションを認識していないために発生します。リソースの競合によってサービスが低下し、ハイパーコンバージェンスの利点が相殺される可能性があります。

Compute サービスが使用するリソースを設定すると、リソースの競合が軽減され、HCI のパフォーマンスが向上します。

#### 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. `NovaReservedHostMemory` パラメーターを `ceph -overrides.yaml` ファイルに追加します。以下は使用例です。

```
parameter_defaults:
  ComputeHCIParameters:
    NovaReservedHostMemory: 75000
```

`NovaReservedHostMemory` パラメーターは、`/etc/nova/nova.conf` の `reserved_host_memory_mb` のデフォルト値をオーバーライドします。このパラメーターは、Ceph OSD が必要とするメモリーを仮想マシンに与える Nova スケジューラーを停止するように設定されます。

上記の例では、ハイパーバイザー用にデフォルトで予約されているメモリーに加えて、ホストごとに 10 OSD 用に OSD ごとに 5 GB を予約しています。IOPS 最適化クラスターでは、OSD ごとにより多くのメモリーを予約することで、パフォーマンスを向上させることができます。5 GB の数値は、必要に応じてさらに絞り込むことができる開始点として提供されます。



#### 重要

`openstack overcloud deploy` コマンドを使用するときに、このファイルを含めます。

### 5.2. カスタム環境ファイルの設定

Director は、デプロイされた Red Hat Ceph Storage クラスタに基本的なデフォルト設定を適用します。カスタム環境ファイルで追加の設定を定義する必要があります。

## 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. カスタム設定を定義するファイルを作成します。  
**vi /home/stack/templates/storage-config.yaml**
3. ファイルに **parameter\_defaults** セクションを追加します。
4. カスタム設定パラメーターを追加します。パラメーター定義の詳細は、[オーバークラウドのパラメーター](#) を参照してください。

```
parameter_defaults:
  CinderEnableiscsiBackend: false
  CinderEnableRbdBackend: true
  CinderBackupBackend: ceph
  NovaEnableRbdBackend: true
  GlanceBackend: rbd
```



### 注記

カスタム設定ファイルで定義されたパラメーターは、`/usr/share/openstack-tripleo-heat-templates/environments/cephadm/cephadm.yaml` の対応するデフォルト設定をオーバーライドします。

5. ファイルを保存します。

## 関連情報

カスタム設定は、オーバークラウドのデプロイ中に、適用されます。

## 5.3. CEPH METADATA SERVER の有効化

Ceph Metadata Server (MDS) は **ceph-mds** デーモンを実行します。このデーモンは、CephFS に保存されているファイルに関連するメタデータを管理します。CephFS は、ネイティブまたは NFS プロトコルで使用できます。



### 注記

Red Hat は、Shared File Systems サービス (manila) のネイティブ CephFS および CephFS NFS バックエンドを使用した Ceph MDS のデプロイをサポートしています。

## 手順

- Ceph MDS を有効にするには、オーバークラウドのデプロイ時、以下の環境ファイルを使用します。

```
/usr/share/openstack-tripleo-heat-templates/environments/cephadm/ceph-mds.yaml
```



## 注記

デフォルトでは、Ceph MDS はコントローラーノードにデプロイされます。Ceph MDS を独自の専用ノードにデプロイできます。

### 関連情報

- [Red Hat Ceph Storage ファイルシステム ガイド](#)

## 5.4. CEPH OBJECT GATEWAY オブジェクトストレージ

Ceph Object Gateway (RGW) は、Red Hat Ceph Storage クラスタ内のオブジェクトストレージ機能にアクセスするためのインターフェイスを提供します。

director を使用して、Ceph をデプロイすると、director は RGW を自動的に有効にします。これは、Object Storage サービス (swift) を直接置き換えるものです。通常、Object Storage サービスを使用するサービスは、追加の設定なしで、代わりに RGW を使用できます。Object Storage サービスは、アップグレードされた Ceph クラスタのオブジェクトストレージオプションとして引き続き利用できます。

個別の RGW 環境ファイルを有効にする必要はありません。その他のオブジェクトストレージオプションの環境ファイルの詳細は、「[Red Hat OpenStack Platform オブジェクトストレージのデプロイメントオプション](#)」を参照してください。

デフォルトでは、Ceph Storage は Object Storage Daemon (OSD) ごとに 250 の配置グループを許可します。RGW を有効にすると、Ceph Storage は RGW に必要な次の 6 つの追加プールを作成します。

- `.rgw.root`
- `<zone_name>.rgw.control`
- `<zone_name>.rgw.meta`
- `<zone_name>.rgw.log`
- `<zone_name>.rgw.buckets.index`
- `<zone_name>.rgw.buckets.data`



## 注記

デプロイメントでは、`<zone_name>` は、プールが属するゾーンの名前に置き換えられます。

### 関連情報

- RGW の詳細は、Red Hat Ceph Storage [Object Gateway ガイド](#) を参照してください。
- Swift の代わりに RGW を使用方法は、[BLock Storage ボリュームのバックアップ ガイド](#) を参照してください。

## 5.5. RED HAT OPENSTACK PLATFORM オブジェクトストレージのデプロイメントオプション

オーバークラウドのオブジェクトストレージをデプロイするには、3 つのオプションがあります。

- Ceph オブジェクトゲートウェイ (RGW)  
「[Ceph Object Gateway オブジェクトストレージ](#)」の説明に従って、RGW をデプロイするには、オーバークラウドのデプロイ時、以下の環境ファイルを含めます。

```
-e environments/cephadm/cephadm.yaml
```

この環境ファイルは、Ceph ブロックストレージ (RBD) と RGW の両方を設定します。

- Object Storage サービス (swift)  
RGW ではなく、Object Storage サービス (swift) をデプロイするには、オーバークラウドのデプロイ時、以下の環境ファイルを含めます。

```
-e environments/cephadm/cephadm-rbd-only.yaml
```

**cephadm-rbd-only.yaml** ファイルは Ceph RBD を設定しますが、RGW は設定しません。



### 注記

Red Hat Ceph Storage クラスターをアップグレードする前に、Object Storage サービス (swift) を使用した場合は、アップグレード中に、**environments/ceph-ansible/ceph-ansible.yaml** ファイルを **environment/cephadm/cephadm-rbd-only.yaml** に置き換えると、RGW ではなく、Object Storage サービス (swift) を引き続き使用できます。詳細は、[Red Hat OpenStack Platform のマイナー更新の実行](#) を参照してください。

Red Hat OpenStack Platform は、Object Storage サービス (swift) から Ceph Object Gateway (RGW) への移行をサポートしていません。

- オブジェクトストレージなし  
RGW または Object Storage サービス (swift) ではなく、RBD で Ceph をデプロイするには、オーバークラウドのデプロイ時、以下の環境ファイルを含めます。

```
-e environments/cephadm/cephadm-rbd-only.yaml  
-e environments/disable-swift.yaml
```

**cephadm-rbd-only.yaml** ファイルは RBD を設定しますが、RGW は設定しません。**disable-swift.yaml** ファイルは、Object Storage サービス (swift) がデプロイされないようにします。

## 5.6. CEPH を使用するための BLOCK STORAGE BACKUP SERVICE の設定

Block Storage Backup サービス (cinder-backup) はデフォルトで無効になっています。Ceph で使用するには、有効にする必要があります。

### 手順

Block Storage Backup サービス (cinder-backup) を有効にするには、オーバークラウドのデプロイ時、以下の環境ファイルを使用します。

```
`/usr/share/openstack-tripleo-heat-templates/environments/cinder-backup.yaml`.
```

## 5.7. CEPH ノード向けの複数のボンディングされたインターフェイスの設定

ボンディングされたインターフェイスを使用して、複数の NIC を組み合わせ、ネットワーク接続に冗長性を追加します。Ceph ノードに十分な NIC がある場合には、各ノードに複数のボンディングされたインターフェイスを作成して冗長化機能を拡張することができます。

ノードが必要とするネットワーク接続ごとに結合インターフェイスを使用します。これにより、冗長性と各ネットワーク専用の接続の両方が提供されます。

情報と手順については **director** を使用した Red Hat OpenStack Platform のインストールと管理ガイドの [オーバークラウドネットワークのプロビジョニング](#) を参照してください。

## 5.8. HCI のオーバークラウドデプロイメントの開始

Red Hat OpenStack Platform (RHOSP) 環境に加えた変更を実装するには、オーバークラウドをデプロイする必要があります。

### 前提条件

- アンダークラウドのインストール前に、**undercloud.conf** ファイルに **generate\_service\_certificate=false** を設定します。それ以外の場合は、Red Hat OpenStack Platform の強化の [オーバークラウドパブリックエンドポイントでの SSL/TLS の有効化](#) の説明に従って、オーバークラウドで SSL/TLS を設定する必要があります。



### 注記

オーバークラウドのデプロイメント中に Ceph ダッシュボードを追加する場合は、**director** を使用した Red Hat Ceph Storage および Red Hat OpenStack Platform のデプロイの [オーバークラウドデプロイメントへの Red Hat Ceph Storage ダッシュボードの追加](#) を参照してください。

### 手順

- オーバークラウドをデプロイする。デプロイメントコマンドには、次のような追加の引数が必要です。

```
$ openstack overcloud deploy --templates -r /home/stack/templates/roles_data_custom.yaml \
\
-e /usr/share/openstack-tripleo-heat-templates/environments/cephadm/cephadm.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/cephadm/ceph-mds.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/cinder-backup.yaml \
-e /home/stack/templates/storage-config.yaml \
-e /home/stack/templates/deployed-ceph.yaml \
--ntp-server pool.ntp.org
```

上記のコマンド例は、以下のオプションを使用します。

- templates:** デフォルトの heat テンプレートコレクション (`/usr/share/openstack-tripleo-heat-templates/`) からオーバークラウドを作成します。
- r /home/stack/templates/roles\_data\_custom.yaml:** カスタマイズされたロール定義ファイルを指定します。
- e /usr/share/openstack-tripleo-heat-templates/environments/cephadm/cephadm.yaml:** 以前にデプロイされた Ceph Storage クラスターを確定するようにディレクターを設定します。この環境ファイルは、デフォルトで RGW をデプロイします。また、プール、キー、およびデーモンも作成します。

- **-e /usr/share/openstack-tripleo-heat-templates/environments/cephadm/ceph-mds.yaml**: Ceph Metadata Server を有効にします。
- **-e /usr/share/openstack-tripleo-heat-templates/environments/cinder-backup.yaml** - Block Storage Backup サービスを有効にします。
- **-e /home/stack/templates/storage-config.yaml**: Ceph Storage のカスタム設定が含まれる環境ファイルを追加します。
- **-e /home/stack/templates/deployed-ceph.yaml**: 以前に実行した **openstack overcloud ceph deploy** コマンドの出力として、Ceph クラスタ設定を含む環境ファイルを追加します。
- **--ntp-server pool.ntp.org** - NTP サーバーを設定します。



#### 注記

オプションの完全なリストを取得するには、**openstack help overcloud deploy** コマンドを実行してください。

#### 関連情報

- 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 [CLI ツールを使用した基本的なオーバークラウドの設定](#) を参照してください。

## 第6章 HCI 設定の確認

展開が完了したら、HCI 環境が適切に設定されていることを確認します。

### 6.1. HCI 設定の確認

HCI 環境の展開後、指定された設定で展開が成功したことを確認します。

#### 手順

1. ceph シェルを開始します。
2. NUMA とメモリーターゲットの設定を確認します。

```
[ceph: root@oc0-controller-0 /]# ceph config dump | grep numa
osd                                advanced osd_numa_auto_affinity      true
[ceph: root@oc0-controller-0 /]# ceph config dump | grep autotune
osd                                advanced osd_memory_target_autotune  true
[ceph: root@oc0-controller-0 /]# ceph config get mgr
mgr/cephadm/autotune_memory_target_ratio
0.200000
```

3. 特定の OSD 設定を確認します。

```
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_memory_target
4294967296
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_memory_target_autotune
true
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_numa_auto_affinity
true
```

4. 特定の OSD バックフィル設定を確認します。

```
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_recovery_op_priority
3
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_max_backfills
1
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_recovery_max_active_hdd
3
[ceph: root@oc0-controller-0 /]# ceph config get osd.11 osd_recovery_max_active_ssd
10
```

5. コンピュートノードで **reserved\_host\_memory\_mb** の設定を確認します。

```
$ sudo podman exec -ti nova_compute /bin/bash
bash-5.1$ grep reserved_host_memory_mb /etc/nova/nova.conf
```

## 第7章 ハイパーコンバージドノードのスケールリング

HCI ノードをスケールアップまたはスケールダウンする場合、コンピュータードまたは Red Hat Ceph Storage ノードのスケールリングと同じ原則および手法が適用されます。

### 7.1. HCI 環境におけるハイパーコンバージドノードのスケールアップ

HCI 環境のハイパーコンバージドノードをスケールアップするには、ハイパーコンバージドノードではないノードをスケールアップするのと同じ手順に従います。詳細は、[オーバークラウドへのノード追加](#)を参照してください。



#### 注記

新規ノードをタグ付けする場合には、必ず適切なフレーバーを使用するようにしてください。

### 7.2. HCI 環境におけるハイパーコンバージドノードのスケールダウン

HCI 環境のハイパーコンバージドノードをスケールダウンするには、HCI ノード上の Ceph OSD サービスをリバランスし、HCI ノードからインスタンスを移行し、オーバークラウドからコンピュータードを削除する必要があります。

#### 手順

1. HCI ノード上の Ceph OSD サービスを無効にして、リバランスします。HCI ノードまたは Red Hat Ceph Storage ノードを削除する際に、director は Red Hat Ceph Storage クラスタを自動的にリバランスしないので、このステップが必要となります。詳細は、[Red Hat Ceph Storage および Red Hat OpenStack Platform を director と共にデプロイするの Ceph Storage クラスタのスケールリング](#)を参照してください。
2. HCI ノードからインスタンスを移行します。詳細は、[インスタンス作成のための Compute サービスの設定の コンピュートノード間の仮想マシンインスタンスの移行](#)を参照してください。
3. オーバークラウドからコンピュータードを削除します。詳細は、[コンピュータードの削除](#)を参照してください。

## 付録A 関連情報

### A.1. 設定ガイド

次の設定ガイドは、ハイパーコンバージドインフラストラクチャー環境を作成するためのフレームワークを提供することを目的としています。このガイドは、すべての Red Hat OpenStack Platform インストールに最終的な設定パラメーターを提供することを目的としたものではありません。特定の環境に適した具体的なガイドや提案は、[Red Hat カスタマーエクスペリエンスおよびエンゲージメントチーム](#) にお問い合わせください。

- [クラスタのサイジングとスケールアウト](#)
- [容量のプランニングとサイジング](#)

#### A.1.1. クラスタのサイジングとスケールアウト

[Red Hat Ceph Storage ハードウェアガイド](#) では、IOPS、スループット、コストと容量が最適化された Ceph デプロイメントシナリオに関する推奨事項が提供されます。デプロイメントシナリオを最もよく表す推奨事項に従い、コンピュータワークロードをサポートするために必要な NIC、CPU、RAM を追加します。

最適で設置面積が小さい設定は、7つのノードで構成されます。環境内で IOPS 最適化されたパフォーマンスの要件があり、オールフラッシュストレージを使用している場合を除き、スループット最適化導入シナリオを使用する必要があります。

3 ノード Ceph Storage クラスタ設定が可能です。この設定では、次のことを行う必要があります。

- オールフラッシュストレージを使用します。
- **ceph.conf** ファイルで、**replica\_count** パラメーターを 3 に設定します。
- **ceph.conf** ファイルで **min\_size** パラメーターを 2 に設定します。

この設定でノードがサービスを終了しても、IOPS は続きます。データの 3 つのコピーを保持するために、3 番目のノードへのレプリケーションは、サービスに戻るまでキューに入れられます。その後、データは 3 番目のノードにバックフィルされます。



#### 注記

最大 64 ノードの HCI 設定がテストされています。一部の HCI 環境の例は、最大 128 ノードまで文書化されています。このような大規模なクラスタについては、サポート例外およびコンサルティングサービスの契約を検討できます。ガイドについては、[Red Hat Customer Experience and Engagement チーム](#) にお問い合わせください。

2 つの NUMA ノードを含むデプロイメントでは、1 つの NUMA ノードでレイテンシーの影響を受けやすいコンピューティングワークロードをホストし、もう 1 つの NUMA ノードで Ceph OSD サービスをホストできます。両方のノードにネットワークインターフェイスがあり、ディスクコントローラーがノード 0 にある場合は、ストレージネットワークにノード 0 のネットワークインターフェイスを使用し、ノード 0 で Ceph OSD ワークロードをホストします。ノード 1 でコンピューティングワークロードをホストし、ノード 1 のネットワークインターフェイスを使用するように設定します。導入用のハードウェアを購入するときは、どの NIC がどのノードを使用するかに注意し、ストレージとワークロードに分割するようにしてください。

#### A.1.2. 容量のプランニングとサイジング

Red Hat Ceph Storage [ハードウェアガイド](#) で定義されているスループット最適化 Ceph ソリューションは、IOPS の最適化を必要としないほとんどのデプロイメントにバランスの取れたソリューションを提供します。環境を作成するときは、ソリューションで提供される設定ガイドラインに加えて、次の点に注意してください。

- OSD ごとに 5 GB の RAM が割り当てられるため、OSD には十分な動作メモリーが確保されません。ハードウェアがこの要件をサポートできることを確認してください。
- CPU の速度は、使用している記憶媒体と一致する必要があります。SSD などの高速ストレージメディアの利点は、CPU が遅すぎてサポートできない場合に無効になる可能性があります。同様に、高速な CPU は、より高速な記憶媒体によってより効率的に使用できます。CPU とストレージの媒体速度のバランスをとり、どちらか一方が他方のボトルネックにならないようにします。

## A.2. ハイパーコンバージドインフラストラクチャー環境の設定に関するガイドとリソース

次のガイドには、ハイパーコンバージドインフラストラクチャー環境の設定に役立つ追加情報と手順が含まれています。

- [director を使用した Red Hat Ceph および OpenStack のデプロイ](#)
  - このガイドでは、Red Hat OpenStack Platform director を使用して、Red Hat Ceph Storage クラスターを持つオーバークラウドを作成する方法について説明します。これには、director を使用して Ceph クラスターをカスタマイズする手順が含まれます。
- [director を使用した Red Hat OpenStack Platform のインストールと管理](#)
  - このガイドは、Red Hat OpenStack Platform 環境のエンドツーエンドのデプロイメントに関するガイダンスを提供します。これには、director のインストール、環境のプランニング、director を使用した OpenStack 環境の構築などが含まれます。
- [Red Hat OpenStack Platform ネットワークの設定](#)
  - このガイドでは、Red Hat OpenStack Platform のネットワークタスクについて詳しく説明します。
- [Configuring persistent storage](#)
  - このガイドでは、Red Hat OpenStack Platform 環境における永続ストレージの使用/管理手順について詳しく説明します。また、各永続ストレージの種別に対応する OpenStack サービスの設定および管理の手順も記載しています。
- [Bare Metal Provisioning サービスの設定](#)
  - このガイドでは、Red Hat OpenStack Platform 環境のオーバークラウドに Bare Metal Provisioning サービスをインストールして設定し、クラウドユーザーの物理マシンをプロビジョニングおよび管理する方法について詳しく説明します。
- [Red Hat OpenStack Platform の強化](#)
  - このガイドでは、Red Hat OpenStack Platform 環境のセキュリティーを強化するためのグッドプラクティスのアドバイスと概念的な情報を提供します。
- [リリースノート](#)

- このドキュメントは、Red Hat OpenStack Platform の本リリースにおける主要機能、機能拡張、既知の問題について記載します。