



## Red Hat OpenStack Platform 15

### 特定の Red Hat OpenStack Platform サービスの デプロイメントに関する推奨事項

Red Hat OpenStack Platform Telemetry サービスおよび Object Storage サービスの  
最大パフォーマンスの活用



## Red Hat OpenStack Platform 15 特定の Red Hat OpenStack Platform サービスのデプロイメントに関する推奨事項

---

Red Hat OpenStack Platform Telemetry サービスおよび Object Storage サービスの最大パフォーマンスの活用

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Deployment\_Recommendations\_for\_Specific\_Red\_Hat\_OpenStack\_Platform\_Services.ent file |.

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

director を使用して Red Hat OpenStack Platform をデプロイする際に、以下の推奨事項に従って数多くのパフォーマンスの問題に対処することができます。

---

## 目次

<b>第1章 概要</b> .....	<b>3</b>
1.1. オープンクラウドを最適化する理由	3
<b>第2章 TELEMETRY サービスの設定に関する推奨事項</b> .....	<b>4</b>
2.1. 小規模なテスト用オープンクラウド上での TELEMETRY サービスの設定	4
2.2. 大規模な実稼働用オープンクラウド上での TELEMETRY サービスの設定	4
<b>第3章 OBJECT STORAGE サービス (SWIFT) の設定に関する推奨事項</b> .....	<b>7</b>
3.1. OBJECT STORAGE サービスのディスク設定に関する推奨事項	7
3.2. OBJECT STORAGE サービスのトポロジー設定に関する推奨事項	7
3.2.1. Object Storage の専用ノードの定義	7
3.3. OBJECT STORAGE サービスにおけるパーティションのべき乗に関する推奨事項	8



---

## 第1章 概要

### 1.1. オーバークラウドを最適化する理由

大規模なオーバークラウドにスケーリングする、あるいは大規模なオーバークラウドをデプロイする予定の場合は、オーバークラウドを最適化して、ワークロードが増えるにつれてパフォーマンスの問題が発生するのを防ぎます。これらの推奨事項に従うことで、スケーリングがオーバークラウドの Telemetry サービスおよび Object Storage サービスのパフォーマンスに影響を与えるのを防ぐことが可能です。

## 第2章 TELEMETRY サービスの設定に関する推奨事項

Red Hat OpenStack Platform(RHOSP)Telemetry サービスは CPU 集約型なので、Telemetry は RHOSP 15 ではデフォルトで有効になっていません。ただし、これらのデプロイメントの推奨事項に従うことで、Telemetry を有効にした場合のパフォーマンス低下を回避することができます。

これらの手順 (1つは小規模なテストオーバークラウド用、もう1つは大規模な実稼働オーバークラウド用) には、Telemetry サービスのパフォーマンスを最大化する推奨事項が含まれています。

### 2.1. 小規模なテスト用オーバークラウド上での TELEMETRY サービスの設定

小規模なテストオーバークラウドで Red Hat OpenStack Platform (RHOSP) Telemetry サービスを有効にする場合、ファイルバックエンドを使用してそのパフォーマンスを向上させることができます。

#### 前提条件

- Telemetry サービスを設定するオーバークラウドのデプロイメントが、実稼働環境用のシステムではない。
- オーバークラウドは、100 未満のインスタンスをサポートする小規模なデプロイメントで、各コントローラーノードの物理コアは最大でも 12 (ハイパースレッディング機能が有効な場合は 24 コア) である。
- オーバークラウドデプロイメントの高可用性は **無効である**。

#### 手順

1. `/usr/share/openstack-tripleo-heat-templates/environments/enable-legacy-telemetry.yaml` 環境ファイルの `parameter_defaults` に以下を追加し、`<FILE>` を `gnocchi` 設定ファイルの名前に置き換えます。

```
parameter_defaults:
  GnocchiBackend: <FILE>
```

2. `enable-legacy-telemetry.yaml` ファイルを `openstack overcloud deploy` コマンドに追加します。

```
openstack overcloud deploy \
-e /home/stack/environment.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/enable-legacy-telemetry.yaml \
[...]
```

#### 関連資料

- 『[director のインストールと使用方法](#)』の「[オーバークラウド環境の変更](#)」

### 2.2. 大規模な実稼働用オーバークラウド上での TELEMETRY サービスの設定

大規模な実稼働用オーバークラウドで Red Hat OpenStack Platform (RHOSP) Telemetry サービスを有効にする場合、Telemetry サービスを専用のノードにデプロイしてそのパフォーマンスを向上させることができます。



Telemetry サービスは、常にストレージバックエンドとして選択された RHOSP オブジェクトストアを使用します。Red Hat Ceph Storage を有効にしない場合、Telemetry サービスは RHOSP Object Storage サービス (swift) を使用します。デフォルトでは、RHOSP director は Object Storage サービスと Telemetry サービスを Controller 上に共存させます。

## 前提条件

- Telemetry サービスをデプロイするオーバークラウドが、大規模な実稼働用のオーバークラウドである。

## 手順

1. 専用のテレメトリノードを設定するには、Controller ロールから Telemetry サービスを削除します。  
`/usr/share/openstack-tripleo-heat-templates/roles_data.yaml` を  
`/home/stack/templates/roles_data.yaml` にコピーして、Orchestration サービス (heat) カスタム環境ファイルを作成します。
2. `/home/stack/templates/roles_data.yaml` で、Controller ロールの **ServicesDefault** 一覧から以下の行を削除します。

```
- OS::TripleO::Services::CeilometerAgentCentral
- OS::TripleO::Services::CeilometerAgentNotification
- OS::TripleO::Services::GnocchiApi
- OS::TripleO::Services::GnocchiMetricd
- OS::TripleO::Services::GnocchiStatsd
- OS::TripleO::Services::AodhApi
- OS::TripleO::Services::AodhEvaluator
- OS::TripleO::Services::AodhNotifier
- OS::TripleO::Services::AodhListener
- OS::TripleO::Services::PankoApi
- OS::TripleO::Services::CeilometerAgentIpmi
```

3. 以下のスニペットを追加し、`roles_data.yaml` を保存します。

```
- name: Telemetry
  ServicesDefault:
    - OS::TripleO::Services::CACerts
    - OS::TripleO::Services::CertmongerUser
    - OS::TripleO::Services::Kernel
    - OS::TripleO::Services::Ntp
    - OS::TripleO::Services::Timezone
    - OS::TripleO::Services::Snmp
    - OS::TripleO::Services::Sshd
    - OS::TripleO::Services::Securetty
    - OS::TripleO::Services::TripleoPackages
    - OS::TripleO::Services::TripleoFirewall
    - OS::TripleO::Services::SensuClient
    - OS::TripleO::Services::FluentdClient
    - OS::TripleO::Services::AuditD
    - OS::TripleO::Services::Collectd
    - OS::TripleO::Services::MySQLClient
    - OS::TripleO::Services::Docker
    - OS::TripleO::Services::CeilometerAgentCentral
    - OS::TripleO::Services::CeilometerAgentNotification
```

```
- OS::TripleO::Services::GnocchiApi
- OS::TripleO::Services::GnocchiMetricd
- OS::TripleO::Services::GnocchiStatsd
- OS::TripleO::Services::AodhApi
- OS::TripleO::Services::AodhEvaluator
- OS::TripleO::Services::AodhNotifier
- OS::TripleO::Services::AodhListener
- OS::TripleO::Services::PankoApi
- OS::TripleO::Services::CeilometerAgentIpmi
```

4. `/home/stack/storage-environment.yaml` ファイルで、Telemetry サービスの専用ノード数を設定します。  
たとえば、**TelemetryCount: 3** を **parameter\_defaults** に追加して、3 つの専用 Telemetry ノードをデプロイします。

```
parameter_defaults:
  TelemetryCount: *3*
```

これで、カスタムの Telemetry ロールが追加されました。

このロールで、新規フレーバーを定義して、特定のテレメトリノードをタグ付けして割り当てることができます。

5. オーバークラウドをデプロイする際には、**roles\_data.yaml** および **storage-environment.yaml** を **openstack overcloud deploy** コマンドが呼び出すテンプレートおよび環境ファイルの一覧に追加します。

```
$ openstack overcloud deploy \
-r /home/stack/templates/roles_data.yaml \
-e /home/stack/templates/storage-environment.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/enable-legacy-telemetry.yaml \
[...]
```

6. 専用ノードを Telemetry サービスに割り当てることができないため、Object Storage サービスをバックエンドとして使用する必要がある場合には、コントローラーノードで Object Storage サービスを設定します。コントローラー上の Object Storage サービスを見つけると、ストレージ I/O 全体が減少します。

## 関連情報

- 『[オーバークラウドの高度なカスタマイズ](#)』の「[新規ロールの作成](#)」
- [Object Storage サービス \(swift\) の設定に関する推奨事項](#)
- 『[director のインストールと使用方法](#)』の「[オーバークラウド環境の変更](#)」

## 第3章 OBJECT STORAGE サービス (SWIFT) の設定に関する推奨事項

Red Hat Ceph Storage を使用せずに Red Hat OpenStack Platform (RHOSP) をデプロイする場合、RHOSP director は RHOSP Object Storage サービス (swift) をデプロイします。Object Store サービスは、RHOSP Telemetry サービスおよび RabbitMQ を含む複数の OpenStack サービスのオブジェクトストアです。Object Storage サービスと共に Telemetry サービスを使用する場合における、RHOSP のパフォーマンスを向上させるためのさまざまな推奨事項を以下に示します。

### 3.1. OBJECT STORAGE サービスのディスク設定に関する推奨事項

Red Hat OpenStack Platform (RHOSP) Object Storage サービス用に、1つまたは複数の独立したディスクを使用します。

デフォルトでは、RHOSP director は、Object Storage サービス用にシステムディスクの `/srv/node/d1` ディレクトリーを使用します。コントローラーではこのディスクは他のサービスにも使用され、エンタープライズ設定で Telemetry サービスがイベントの記録を開始した後に、ディスクがパフォーマンスのボトルネックになる可能性があります。

以下の例は、RHOSP Orchestration サービス (heat) のカスタム環境ファイルからの抜粋です。各コントローラーノードで、Object Storage サービスは2つの独立したディスクを使用します。両方のディスク全体には XFS ファイルシステムが含まれています。

```
parameter_defaults:
  SwiftRawDisks: {"sdb": {}, "sdc": {}}
```

**SwiftRawDisks** は、ノード上の各ストレージディスクを定義します。以下の例では、各コントローラーノードの **sdb** ディスクと **sdc** ディスクの両方を定義します。



#### 重要

複数のディスクを設定する場合は、Bare Metal サービス (ironic) が必ず目的のルートディスクを使用するようにします。

#### 関連資料

- 『[director のインストールと使用方法](#)』の「[ノード向けのルートディスクの定義](#)」

### 3.2. OBJECT STORAGE サービスのトポロジー設定に関する推奨事項

Red Hat OpenStack Platform (RHOSP) Object Storage サービス用に、専用ノードを定義します。これにより、RHOSP Telemetry サービスによるディスク I/O が、コントローラーノード上のその他のサービスに影響を与えなくなります。

#### 3.2.1. Object Storage の専用ノードの定義

ノードを Red Hat OpenStack Platform (RHOSP) Object Storage サービス専用とすることで、パフォーマンスが向上します。

#### 手順

1. (デフォルトの `/usr/share/openstack-tripleo-heat-templates/roles_data.yaml` をベースとする) カスタムの `roles_data.yaml` ファイルを作成します。

2. カスタム **roles\_data.yaml** ファイルを編集し、コントローラーノードから Object Storage サービスのエントリーを削除します。  
具体的には、**Controller** ロールの **ServicesDefault** 一覧から以下の行を削除します。

```
- OS::TripleO::Services::SwiftStorage
```

3. カスタム環境ファイルで **ObjectStorageCount** リソースを使用して、Object Storage サービスに割り当てる専用ノードの数を設定します。  
たとえば、3つのオブジェクトストレージ専用ノードをデプロイするには、環境ファイルの **parameter\_defaults** に **ObjectStorageCount: 3** を追加します。

```
parameter_defaults:
  ObjectStorageCount: 3
```

4. この設定を適用するには、その他の環境ファイルと共に **roles\_data.yaml** をスタックに追加して、オーバークラウドをデプロイします。

```
(undercloud) $ openstack overcloud deploy --templates \
-e [your environment files]
-e /home/stack/templates/roles_data.yaml
```

#### 関連資料

- 『オーバークラウドの [高度なカスタマイズ](#)』の「[コンポーザブルサービスとカスタムロール](#)」
- 『オーバークラウドの [高度なカスタマイズ](#)』の「[ロールへのサービスの追加と削除](#)」
- 『[director のインストールと使用方法](#)』の「[オブジェクトストレージノードの置き換え](#)」
- 『[director のインストールと使用方法](#)』の「[オーバークラウド環境の変更](#)」

### 3.3. OBJECT STORAGE サービスにおけるパーティションのべき乗に関する推奨事項

Red Hat OpenStack Platform (RHOSP) Object Storage サービス用に独立したノードを使用する場合には、パーティションのべき乗により大きな値を使用します。

Object Storage サービスは、変更した [ハッシュリング](#) を使用して、データをディスクとノードに分散します。デフォルトでは、アカウント用、コンテナ用、およびオブジェクト用の3つのリングがあります。各リングは、[パーティションのべき乗](#) と呼ばれる固定パラメーターを使用します。このパラメーターは、作成可能なパーティションの最大数を設定します。

パーティションのべき乗パラメーターは重要で、新規コンテナとそのオブジェクトについてしか変更できません。そのため、[初回デプロイメント](#) の前にこの値を設定することが重要になります。

RHOSP director がデプロイする環境のデフォルトのパーティションのべき乗値は **10** です。小規模なデプロイメント、特に Object Storage サービスにコントローラーノード上のディスクだけを使用する計画の場合には、これが妥当な値です。

以下の表は、3つのレプリカを使用する場合に適切なパーティションのべき乗を選択するのに役立ちます。

表3.1 利用可能なディスクの数に対する適切なパーティションのべき乗値

パーティションのべき乗	ディスクの最大数
10	35 まで
11	75 まで
12	150 まで
13	250 まで
14	500 まで



### 重要

パーティションのべき乗に過剰に大きな値を設定すると (例: 40 ディスクに対して **14**)、レプリケーション時間に悪影響を及ぼします。

パーティションのべき乗を設定するには、以下のリソースを使用します。

```
parameter_defaults:  
  SwiftPartPower: 11
```

### ヒント

新しいコンテナに追加のオブジェクトサーバーリングを設定することもできます。これは、当初小さなパーティションのべき乗値を使用する Object Storage サービスのデプロイメントにディスクを追加する場合に便利です。

### 関連資料

- 『ストレージガイド』の「[オブジェクトストレージリングの設定](#)」
- swift のアップストリームドキュメントの「[The Rings](#)」
- 『director のインストールと使用方法』の「[オーバークラウド環境の変更](#)」