



## Red Hat OpenStack Platform 13

# Shared File Systems サービスの NFS バックエンドに CephFS を使用した場合のガイド

NFS バックエンドに CephFS を使用した OpenStack Shared File Systems サービス  
についての理解、およびその使用と管理



# Red Hat OpenStack Platform 13 Shared File Systems サービスの NFS バックエンドに CephFS を使用した場合のガイド

---

NFS バックエンドに CephFS を使用した OpenStack Shared File Systems サービスについての理解、およびその使用と管理

OpenStack Team  
rhos-docs@redhat.com

## 法律上の通知

Copyright © 2021 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

本ガイドでは、NFS バックエンドに Red Hat Ceph File System (CephFS) を使用した、Red Hat OpenStack Platform 環境の Shared File Systems サービス (manila) について詳細に説明します。これには、そのインストール、設定、および検証に関するさまざまな手順が含まれます。

## 目次

多様性を受け入れるオープンソースの強化 .....	3
<b>第1章 NFS バックエンドに CEPHFS を使用した OPENSTACK SHARED FILE SYSTEMS サービス .....</b>	<b>4</b>
1.1. NFS バックエンドに CEPHFS を使用する構成の概要	4
1.2. NFS バックエンドに CEPHFS を使用した SHARED FILE SYSTEMS サービス使用のメリット	4
1.3. CEPH FILE SYSTEM のアーキテクチャー	5
1.3.1. ネイティブドライバを使用する CephFS	5
1.3.2. NFS バックエンドへの CephFS の使用	6
1.3.2.1. Ceph サービスとクライアントアクセス	7
1.3.2.2. NFS バックエンドに CephFS を使用した Shared File Systems サービスの耐障害性	8
<b>第2章 NFS バックエンドに CEPHFS を使用する構成のインストール .....</b>	<b>9</b>
2.1. NFS-GANESHA を持つ CEPHFS のデプロイメント	9
2.1.1. 要件	9
2.1.2. ファイル共有	10
2.1.3. NFS バックエンドに CephFS を使用する構成で使用される分離ネットワーク	10
2.2. NFS バックエンドに CEPHFS を使用するためのオプションおよびカスタム NETWORK_DATA ファイルを使用した OPENSTACK のインストール	10
2.2.1. ceph-ansible パッケージのインストール	11
2.2.2. オーバークラウドコンテナイメージの準備	11
2.2.2.1. カスタムロールファイルの生成	12
2.2.3. 更新された環境のデプロイ	12
2.2.3.1. StorageNFS および network_data_ganesha.yaml ファイル	14
2.2.3.2. manila-cephfsganesha-config.yaml	14
2.2.4. デプロイメント後設定の完了	15
2.2.4.1. 分離ネットワークの設定	15
2.2.4.2. 共有プロバイダー StorageNFS ネットワークのセットアップ	17
2.2.4.3. デフォルトファイル共有種別のセットアップ	17
<b>第3章 NFS バックエンドに CEPHFS を使用する構成が正常にデプロイされたことの検証 .....</b>	<b>19</b>
3.1. STORAGE NFS 分離ネットワークが作成されていることの確認	19
3.2. CEPH MDS サービスの確認	20
3.3. CEPH クラスターのステータスの確認	20
3.4. NFS-GANESHA および MANILA-SHARE サービスのステータスの確認	21
3.5. MANILA-API サービスがスケジューラーおよびファイル共有サービスを認識していることの確認	21



## 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、[弊社の CTO、Chris Wright のメッセージ](#) を参照してください。

# 第1章 NFS バックエンドに CEPHFS を使用した OPENSTACK SHARED FILE SYSTEMS サービス

Red Hat OpenStack Platform (RHOSP) は、Red Hat Enterprise Linux 上にプライベートまたはパブリックの Infrastructure-as-a-Service (IaaS) クラウドを構築するための基盤を提供します。RHOSP は、クラウド対応のワークロード開発向けのスケーラビリティおよび耐障害性に優れたプラットフォームです。

NFS バックエンドに Ceph File System (CephFS) を使用した Shared File Systems サービス (manila) により、クラウド管理者はブロックストレージおよびオブジェクトストレージ用と同じ Ceph クラスタを使用し、NFS プロトコルを介したファイル共有を提供することができます。補足情報については、『[ストレージガイド](#)』の「[Shared File Systems サービス](#)」の章を参照してください。

本ガイドでは、NFS バックエンドに CephFS を使用する構成のインストール、設定、デプロイ、およびテストに関する概念および手順について説明します。

NFS バックエンドに Ceph File System (CephFS) を使用した OpenStack Shared File Systems サービス (manila) は、Red Hat OpenStack Platform に耐障害性を持つ NFS 共有サービスを提供します。補足情報については、『[ストレージガイド](#)』の「[Shared File Systems サービス](#)」の章を参照してください。

## 1.1. NFS バックエンドに CEPHFS を使用する構成の概要

CephFS は、高度にスケーラブルな Ceph のオープンソース分散ファイルシステムのコンポーネントで、統一された分散ストレージプラットフォームです。Ceph は、Reliable Autonomic Distributed Object Store (RADOS) を使用してオブジェクトストレージ、ブロックストレージ、およびファイルストレージを実装します。POSIX に対応した CephFS は、Ceph ストレージクラスターへのファイルアクセスを提供します。

Shared File Systems サービスでは、ユーザーは CephFS にファイル共有を作成し、NFS 4.1 を使用して NFS-Ganesha 経由でそのファイル共有にアクセスすることができます。NFS-Ganesha はファイル共有へのアクセスをコントロールし、NFS 4.1 プロトコルを使用してファイル共有をクライアントにエクスポートします。Shared File Systems サービスは、OpenStack 内からこれらのファイル共有のライフサイクルを管理します。NFS バックエンドに CephFS を使用するようサービスをセットアップした場合には、これらのファイル共有は CephFS クラスタから提供されますが、通常の NFS 共有として作成およびアクセスされます。

## 1.2. NFS バックエンドに CEPHFS を使用した SHARED FILE SYSTEMS サービス使用のメリット

NFS バックエンドに CephFS を使用した Shared File Systems サービス (manila) により、クラウド管理者はブロックストレージおよびオブジェクトストレージ用と同じ Ceph クラスタを使用し、一般的な NFS プロトコル (ほとんどのオペレーティングシステムでデフォルトで利用可能) を介したファイル共有を提供することができます。CephFS は、すでに OpenStack クラウドの他のサービス (Block Storage (cinder)、オブジェクトストレージ等) でストレージバックエンドとして使用されている Ceph クラスタの機能を最大限に活用します。

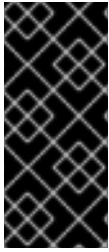


### 注記

Red Hat OpenStack director により設定されていない、外部にデプロイされた Ceph クラスタに CephFS を追加する構成は、現在サポートされていません。director で定義することのできる CephFS バックエンドは、現状1つだけです。



テクノロジープレビュー機能の CephFS ネイティブドライバーとは異なり、CephFS NFS ドライバー (NFS-Ganesha) は、Red Hat OpenStack Platform 13 で完全にサポートされます。



## 重要

Red Hat CephFS ネイティブドライバーは **テクノロジープレビュー** としてのみ利用可能であり、したがって、Red Hat による完全なサポートは提供されません。

テクノロジープレビュー機能についての詳しい情報は、[「対象範囲の詳細」](#) を参照してください。

NFS バックエンドに CephFS を使用する構成のデプロイメントでは、Ceph ストレージバックエンドはユーザーのネットワークから独立しています。このため、ベースとなる Ceph ストレージでは、悪意のある攻撃や誤操作への耐性が向上します。

データプレーントラフィック用のネットワークとコントロールプレーンサービス (Shared File Systems サービス) との通信用 API ネットワークが分離されるので、ファイルストレージがよりセキュアになります。

Ceph クライアントはシステムの管理下に置かれます。エンドユーザーが制御する NFS クライアント (例: 独立したユーザー仮想マシン) は、Ceph クラスターのストレージバックエンドに直接アクセスすることはありません。

## 1.3. CEPH FILE SYSTEM のアーキテクチャー

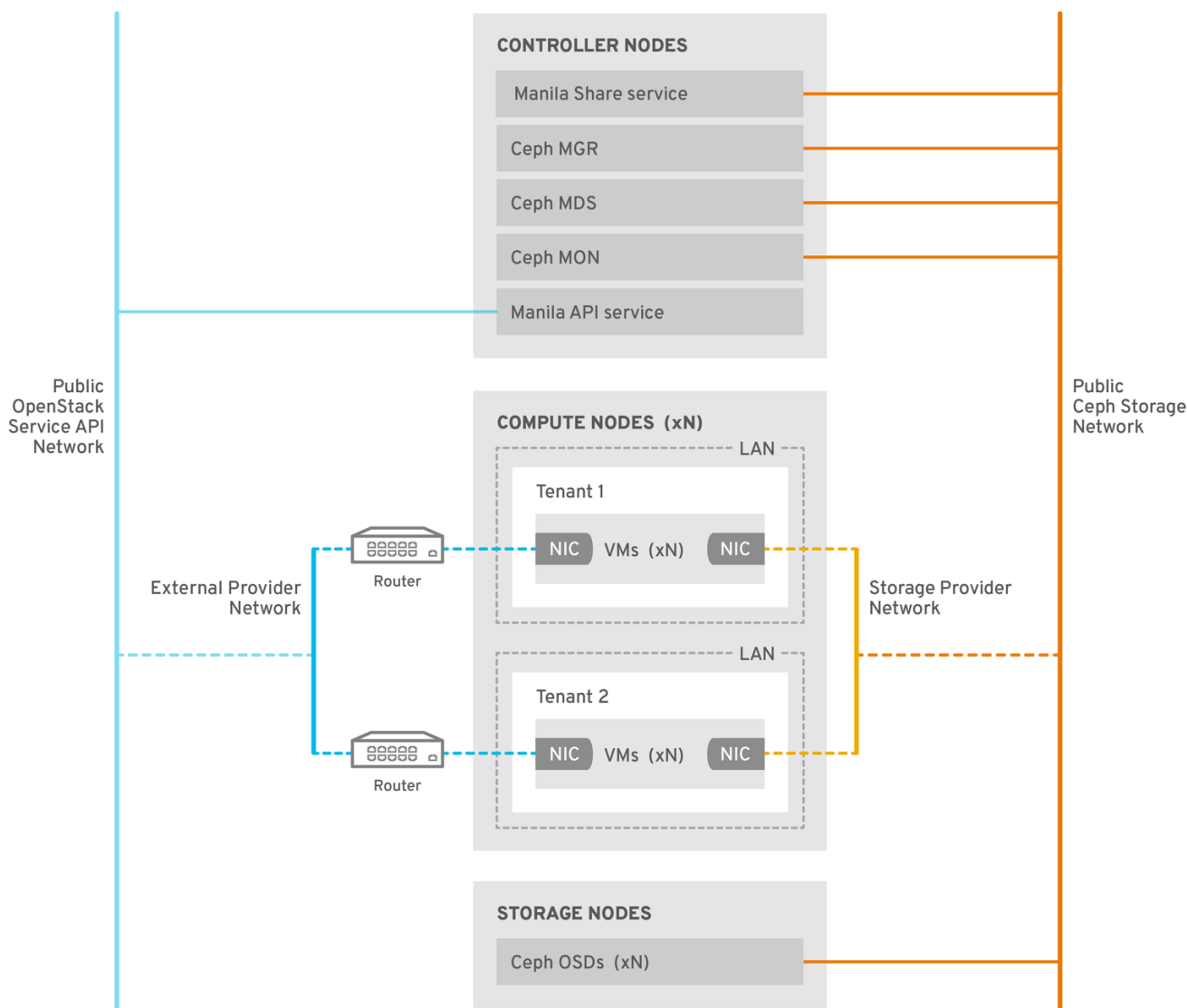
Ceph File System (CephFS) は分散ファイルシステムで、NFS v4 プロトコルを使用する NFS-Ganesha (サポート対象) または CephFS ネイティブドライバー (テクノロジープレビュー) のいずれかと共に使用することができます。

### 1.3.1. ネイティブドライバーを使用する CephFS

CephFS ネイティブドライバーは、OpenStack Shared File Systems サービス (manila) と Red Hat Ceph Storage を結び付けます。director によりデプロイされる際に、コントローラーノードはマネージャー、メタデータサーバー (MDS)、および モニター (MON) 等の Ceph デーモンと共に Shared File Systems サービスをホストします。

コンピュータノードは1つまたは複数のテナントをホストします。ユーザー管理の仮想マシン (2つの NIC を持つ灰色の長方形) が含まれるテナント (白い長方形で表される) は、パブリック Ceph ストレージネットワーク経由で **ceph** および **manila** デーモンに接続してアクセスします。このネットワークにより、Ceph オブジェクトストレージデーモン (OSD) の提供するストレージノードに保管されたデータにアクセスすることもできます。テナントがホストするインスタンス (仮想マシン) は、2つの NIC と共に起動します。1つの NIC はストレージプロバイダーネットワーク専用で、2つ目の NIC は外部プロバイダーネットワークに接続するためのテナント所有のルーター専用です。

ストレージプロバイダーネットワークは、テナント上で動作する仮想マシンをパブリック Ceph ストレージネットワークに接続します。Ceph パブリックネットワークは、Ceph オブジェクトストレージノード、メタデータサーバー (MDS)、およびコントローラーノードへのバックエンドアクセスを提供します。ネイティブドライバーを使用する場合、CephFS では、クォータの適用、確実なテナント間の分離、およびセキュリティ確保のために、クライアントおよびサーバーとの協調が必要になります。ネイティブドライバーを使用する CephFS は、プライベートクラウド上の信頼済みエンドユーザーが定義された環境で適切に機能します。この構成での協調および適切な動作のためには、ソフトウェアはユーザーの管理下で動作している必要があります。



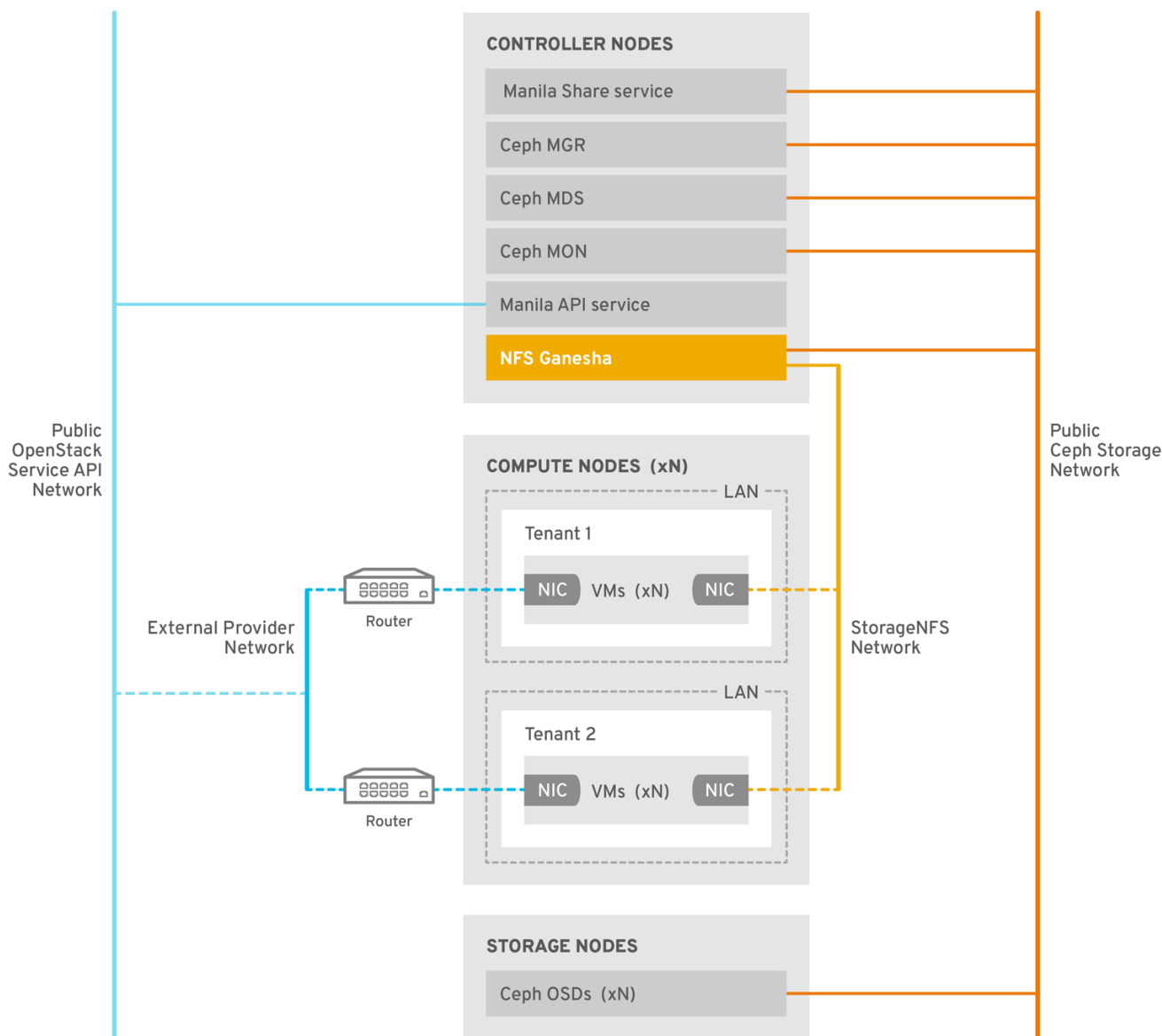
OPENSTACK\_476233\_0818

### 1.3.2. NFS バックエンドへの CephFS の使用

OpenStack Shared File Systems サービス (manila) の NFS バックエンドに CephFS を使用する構成には、Ceph メタデータサーバー (MDS)、NFS バックエンドに CephFS を使用するためのゲートウェイ (NFS-Ganesha)、および Ceph クラスターのサービスコンポーネントが含まれます。Shared File Systems サービスの CephFS NFS ドライバーは、NFS-Ganesha ゲートウェイを使用して NFSv4 プロトコルによる CephFS ファイル共有へのアクセスを提供します。Ceph MDS サービスは、ファイルシステムのディレクトリーおよびファイル名を RADOS クラスターに保管されたオブジェクトにマッピングします。NFS ゲートウェイは、Ceph 等の異なるストレージバックエンドを使用して NFS ファイル共有を提供します。NFS-Ganesha サービスは、Ceph サービスと共にコントローラーノード上で実行されます。

インスタンスは、少なくとも 2 つの NIC と共に起動します。1 つの NIC はテナントのルーターに接続され、2 つ目の NIC は StorageNFS ネットワークに接続され、そこから直接 NFS-Ganesha ゲートウェイに接続されます。インスタンスは、NFS プロトコルを使用してファイル共有をマウントします。Ceph OSD ノードがホストする CephFS ファイル共有は、NFS ゲートウェイを通じて提供されます。

NFS-Ganesha により、ユーザーのインスタンスは MDS および他の Ceph サービスに直接アクセスしなくなるので、セキュリティが向上します。インスタンスは、Ceph デーモンに直接アクセスしません。



OPENSTACK\_476233\_0818

### 1.3.2.1. Ceph サービスとクライアントアクセス

Ceph がオブジェクトストレージおよびブロックストレージを提供する際にデプロイされるモニター、OSD、Rados Gateway (RGW)、およびマネージャーサービスに加えて、CephFS には Ceph メタデータサービス (MDS) が必要です。また、NFS プロトコルを使用するネイティブ CephFS へのゲートウェイとして NFS-Ganesha サービスが必要です。(ユーザーが直接アクセスできるオブジェクトストレージの場合には、RGW サービスもデプロイされます。)ゲートウェイは Ceph パブリックネットワークにアクセスするために CephFS クライアントを実行し、エンドユーザーではなくシステムの管理下に置かれます。

NFS-Ganesha は、Ceph パブリックネットワークおよび新たな分離ネットワーク (StorageNFS) の両方とインターフェースする専用の docker コンテナ内で実行されます。この分離ネットワークをデプロイしてコントローラーノードに接続するのに、OpenStack director のコンポーザブルネットワーク機能が使用されます。その後、クラウド管理者はネットワークを neutron プロバイダーネットワークとして設定します。

NFS-Ganesha は Ceph パブリックネットワークを通じて CephFS にアクセスし、StorageNFS ネットワーク上のアドレスを使用してその NFS サービスをバインドします。

NFS 共有にアクセスするには、StorageNFS ネットワークに接続される新たな NIC と共に、ユーザー仮

想マシン (nova インスタンス) をプロビジョニングします。CephFS ファイル共有のエクスポート場所は、StorageNFS ネットワーク上の NFS-Ganesha サーバーの仮想 IP を使用して、標準的な NFS IP: <path> タプルとして表示されます。ユーザー仮想マシンのアクセス制御は、そのネットワーク上の仮想マシンの IP を使用して行われます。

Neutron セキュリティーグループが、テナント 1 に属するユーザー仮想マシンが StorageNFS ネットワークを通じてテナント 2 に属するユーザー仮想マシンにアクセスするのを防ぎます。テナントは同じ CephFS ファイルシステムを共有しますが、ユーザー仮想マシンはエクスポートツリー下のファイル (/path/to/share1/.....、/path/to/share2/.....) にしかアクセスできないので、テナントデータパスの分離が確保されます。

### 1.3.2.2. NFS バックエンドに CephFS を使用した Shared File Systems サービスの耐障害性

OpenStack director が Ceph サービスデーモンを起動すると、これらのデーモンは自己の高可用性 (HA) 状態を管理し、一般的に、これらのデーモン用に複数のインスタンスが実行されます。これとは対照的に、本リリースでは、ファイル共有を提供することのできる NFS-Ganesha 用インスタンスは、常に 1 つだけです。

NFS バックエンドに CephFS を使用したファイル共有では、データパスに単一障害点が生じるのを避けるために、NFS-Ganesha はアクティブ-パッシブ設定 (Pacemaker-Corosync クラスタが管理) の OpenStack コントローラーノード上で実行されます。NFS-Ganesha は、複数のコントローラーノードに渡って仮想サービス IP アドレスを持つ仮想サービスとして機能します。

コントローラーに障害が発生した (あるいは、特定のコントローラーノード上のサービスに障害が発生し、そのノードで復帰できない) 場合には、Pacemaker-Corosync が同じ仮想 IP を使用して新たな NFS-Ganesha インスタンスを別のコントローラー上で起動します。既存クライアントのマウントはファイル共有のエクスポート場所の仮想 IP を使用するので、これらのマウントは維持されます。

デフォルトの NFS マウントオプション設定および NFS 4.1 以降を使用している場合には、障害発生後に TCP 接続がリセットされ、クライアントが再接続されます。フェイルオーバー中は I/O 操作が一時的に応答しなくなりますが、機能は失われません。アプリケーション I/O も応答しなくなりますが、フェイルオーバーが完了すると処理が再開されます。

最大 90 秒の猶予期間 (クライアントがロックを再要求するのをサーバーが待機する期間) が経過するまで、新規接続や新たなロック状態などは拒否されます。NFS-Ganesha はクライアントの一覧を維持し、すべてのクライアントがロックを再要求したことを確認すると、これより早く猶予期間を終了する場合があります。



#### 注記

猶予期間のデフォルト値は 90 秒です。この値は、NFSv4 **Grace\_Period** 設定オプションにより変更することが可能です。

## 第2章 NFS バックエンドに CEPHFS を使用する構成のインストール

### 2.1. NFS-GANESHA を持つ CEPHFS のデプロイメント

OpenStack 環境で NFS バックエンドに Ceph File System (CephFS) を使用する一般的な構成は、以下のとおりです。

- コンテナ化された Ceph メタデータサーバー (MDS)、Ceph モニター (MON)、manila、および NFS-Ganesha サービスを実行する OpenStack コントローラーノード。これらのサービスのいくつかは、同じノードに共存する場合や、専用のノードを持つ場合があります。
- Ceph ストレージノード上で動作するコンテナ化されたオブジェクトストレージデーモン (OSD) を持つ Ceph ストレージクラスター
- NFS 共有のプロビジョニング用に、テナントから NFS-Ganesha サービスへのアクセスを提供する StorageNFS 分離ネットワーク

Shared File Systems (manila) サービスの提供する API により、テナントはファイルシステムの共有を要求することができます。これは、ドライバーモジュールにより実施されます。Red Hat CephFS のドライバー (`manila.share.drivers.cephfs.driver.CephFSDriver`) により、Shared File Systems サービスは CephFS をバックエンドとして使用することができます。Red Hat OpenStack Platform director は、NFS-Ganesha ゲートウェイをデプロイするようにドライバーを設定します。これにより、NFS 4.1 プロトコルを使用して CephFS ファイル共有が提供されます。本書では、これを NFS バックエンドに CephFS を使用する構成と呼びます。

OpenStack director を使用してオーバークラウドに CephFS バックエンドを持つ Shared File Systems をデプロイすると、(heat テンプレートで定義された) 必要なストレージネットワークが自動的に作成されます。ネットワークプランニングに関する詳しい情報は、『[director のインストールと使用方法](#)』の「[ネットワークのプランニング](#)」セクションを参照してください。

ノードの `/etc/manila/manila.conf` ファイルを編集して Shared File Systems サービスを手動で設定することができますが、今後のオーバークラウド更新時に、Red Hat OpenStack Platform director を使用して任意の設定を上書きすることができます。Shared File Systems のバックエンドを設定する場合には、director を使用する方法を推奨します。

本項では、director の管理する統合デプロイメントにおいて NFS バックエンドに CephFS を使用する構成をインストールする方法について説明します。



#### 注記

Red Hat OpenStack director により設定されていない、外部にデプロイされた Ceph クラスターに CephFS を追加する構成は、現在サポートされていません。director で一度に定義することのできる CephFS バックエンドは、現状1つだけです。

#### 2.1.1. 要件

NFS バックエンドに CephFS を使用する構成には、Red Hat OpenStack Platform バージョン 13 以降の既存または新規環境が必要です。CephFS は、Red Hat Ceph Storage バージョン 3 で機能します。この環境をデプロイする手順については、『[コンテナ化された Red Hat Ceph を持つオーバークラウドのデプロイ](#)』を参照してください。

本書の前提は以下のとおりです。

- Shared File Systems サービスは、デフォルトの動作と同様にコントローラーノードにインストールされる。
- NFS-Ganesha ゲートウェイサービスは、コントローラーノードの Pacemaker クラスタにインストールされる。
- CephFS バックエンドの1つのインスタンスだけが、Shared File Systems サービスにより使用される。その他の CephFS 以外のバックエンドは、1つの CephFS バックエンドと共に使用することができる。
- ストレージトラフィック用に、OpenStack Platform director により新たなネットワーク (StorageNFS) が作成される。
- NFS バックエンドに CephFS を使用する構成と同時に、新たな Red Hat Ceph Storage バージョン 3 クラスタが設定される。

### 2.1.2. ファイル共有

OpenStack Shared File Systems サービス (manila)、Ceph File System (CephFS)、および NFS バックエンドに Ceph を使用する構成では、ファイル共有の取り扱いが若干異なります。

Shared File Systems サービスの提供するファイル共有は、個別のファイルシステムの名前空間であり、またストレージあるいは共有のユニットで、特定のサイズを持ちます (例: クォータが設定されたサブディレクトリー)。アクセスが要求される前にファイルシステムがセットアップされるので、Shared File Systems のストレージはマルチクライアントに対応します (一方、ブロックストレージでは、アクセスが要求された時にファイルシステムがセットアップされます)。

CephFS では、ファイル共有は、特定のストレージプールまたは名前空間をポイントする、クォータおよびレイアウトが定義されたディレクトリーとして扱われます。CephFS のクォータは、ディレクトリーのサイズを Shared File Systems サービスが作成するファイル共有のサイズに制限します。Ceph ファイル共有へのアクセス可否は、MDS の認証機能により決定されます。

NFS バックエンドに CephFS を使用する構成では、ファイル共有のプロビジョニングおよびアクセスに NFS プロトコルが使用されます。NFS プロトコルはセキュリティーにも対応します。

### 2.1.3. NFS バックエンドに CephFS を使用する構成で使用される分離ネットワーク

NFS バックエンドに CephFS を使用する構成のデプロイメントでは、新たな分離ネットワーク StorageNFS が使用されます。このネットワークをデプロイすることで、ユーザーはそのネットワーク上に表示されるファイル共有を NFS プロトコルを使用してマウントすることができます。ストレージまたはストレージ管理ネットワークにアクセスする必要はありません。ストレージ管理ネットワークは、インフラトラフィック用に確保されます。

ネットワークの分離に関する補足情報は、『[director のインストールと使用方法](#)』の「[基本的なネットワーク分離](#)」を参照してください。

## 2.2. NFS バックエンドに CEPHFS を使用するためのオプションおよびカスタム NETWORK\_DATA ファイルを使用した OPENSTACK のインストール

NFS バックエンドに CephFS を使用する構成をインストールするには、以下の操作が必要です。

1. ceph-ansible パッケージをインストールする。
2. **openstack overcloud image prepare** コマンドを使用して、オーバークラウドコンテナイメージを準備する。



3. カスタムロールファイル (**roles\_data.yaml**) および **network\_data.yaml** ファイルを生成する。
4. カスタムロールおよび環境ファイルと共に **openstack overcloud deploy** コマンドを使用して、Ceph、Shared File Systems サービス (manila)、および CephFS をデプロイする。
5. StorageNFS 分離ネットワークを設定し、デフォルトのファイル共有種別を作成する。

以下の例では、OpenStack 環境の標準の **stack** ユーザーが使用されています。

これらのタスクは、OpenStack のインストールまたは環境の更新と共に実施する必要があります。

### 2.2.1. ceph-ansible パッケージのインストール

OpenStack director でコンテナ化された Ceph をデプロイするには、アンダークラウドノードに **ceph-ansible** パッケージをインストールする必要があります。

#### 手順

1. アンダークラウドノードにログインします。
2. root 権限で **yum install** を使用して、ceph-ansible パッケージをインストールします。

```
[stack@undercloud-0 ~]$ sudo yum install -y ceph-ansible
[stack@undercloud-0 ~]$ sudo yum list ceph-ansible
...
Installed Packages
ceph-ansible.noarch 3.1.0-0.1.el7 rhelosp-13.
```

### 2.2.2. オーバークラウドコンテナイメージの準備

OpenStack ではすべてのサービスがコンテナ化されているので、**openstack overcloud image prepare** コマンドを使用してオーバークラウド用に Docker イメージを準備する必要があります。追加のオプションを指定して以下のコマンドを実行すると、**ceph** および **manila** サービスのデフォルトイメージが docker レジストリーに追加されます。Ceph MDS および NFS-Ganesha サービスは、同じ Ceph のベースコンテナイメージを使用します。

コンテナイメージに関する補足情報は、『**director のインストールと使用方法**』の「[追加のサービス用のコンテナイメージ](#)」セクションを参照してください。

#### 手順

1. **-e** オプションにより以下の環境ファイルを追加して、アンダークラウドから **openstack overcloud image prepare** コマンドを実行します。

```
$ openstack overcloud container image prepare \
...
-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-ansible.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/services-docker/manila.yaml \
...
```

2. **grep** コマンドを使用して、**ceph** および **manila** サービスのデフォルトイメージが **containers-default-parameters.yaml** ファイルに含まれていることを確認します。

```
[stack@undercloud-0 ~]$ grep -E 'ceph|manila' composable_roles/docker-images.yaml
DockerCephDaemonImage: 192.168.24.1:8787/rhceph:3-12
DockerManilaApilImage: 192.168.24.1:8787/rhosp13/openstack-manila-api:2018-08-22.2
DockerManilaConfigImage: 192.168.24.1:8787/rhosp13/openstack-manila-api:2018-08-22.2
DockerManilaSchedulerImage: 192.168.24.1:8787/rhosp13/openstack-manila-
scheduler:2018-08-22.2
DockerManilaShareImage: 192.168.24.1:8787/rhosp13/openstack-manila-share:2018-08-
22.2
```

### 2.2.2.1. カスタムロールファイルの生成

StorageNFS 分離ネットワークのセットアップには、**ControllerStorageNFS** カスタムロールを使用します。このロールはデフォルトの **Controller.yaml** ロールファイルに類似していますが、StorageNFS ネットワークおよび **CephNfs** サービス (**OS::TripleO::Services:CephNfs** で表される) が追加されています。

```
[stack@undercloud ~]$ cd /usr/share/openstack-tripleo-heat-templates/roles
[stack@undercloud roles]$ diff Controller.yaml ControllerStorageNfs.yaml
16a17
> - StorageNFS
50a45
> - OS::TripleO::Services::CephNfs
```

**openstack overcloud roles generate** コマンドに関する情報は、『**オーバークラウドの高度なカスタマイズ**』の「**ロール**」セクションを参照してください。

#### 手順

**openstack overcloud roles generate** コマンドにより、**-o** 以降に指定したサービスが含まれるカスタム **roles\_data.yaml** ファイルが作成されます。以下の例では、作成される **roles\_data.yaml** ファイルには、**ControllerStorageNfs**、**Compute**、および **CephStorage** のサービスが含まれます。



#### 注記

既存の **roles\_data.yaml** ファイルがある場合には、それを変更して設定ファイルに **ControllerStorageNfs**、**Compute**、および **CephStorage** サービスを追加します。『**オーバークラウドの高度なカスタマイズ**』の「**ロール**」セクションを参照してください。

1. アンダークラウドノードにログインします。
2. **openstack overcloud roles generate** コマンドを使用して、**roles\_data.yaml** ファイルを作成します。

```
[stack@undercloud ~]$ openstack overcloud roles generate --roles-path
/usr/share/openstack-tripleo-heat-templates/roles -o /home/stack/roles_data.yaml
ControllerStorageNfs Compute CephStorage
```

### 2.2.3. 更新された環境のデプロイ

環境をデプロイする準備が整ったら、NFS-Ganesha と共に CephFS を実行するのに必要なカスタム環境およびロールを指定して、**openstack overcloud deploy** コマンドを使用します。これらの環境およびロールについて、以下で説明します。



実際のオーバークラウドデプロイコマンドでは、その他の必要なオプションに加えて以下のオプションを指定します。

機能	オプション	補足情報
<b>overcloud container image prepare</b> コマンドで更新したデフォルトコンテナイメージを追加する	<b>-e /home/stack/containers-default-parameters.yaml</b>	「オーバークラウドコンテナイメージの準備」
<b>network_data_ganesha.yaml</b> を使用して新たな StorageNFS ネットワークを追加する。	<b>-n /usr/share/openstack-tripleo-heat-templates/network_data_ganesha.yaml</b>	「StorageNFS および <a href="#">network_data_ganesha.yaml</a> ファイル」
前のセクションで作成した <b>roles_data.yaml</b> ファイルにより定義したカスタムロールを追加する。	<b>-r /home/stack/roles_data.yaml</b>	「カスタムロールファイルの生成」
<b>ceph-ansible.yaml</b> を使用して Ceph デーモンをデプロイする。	<b>-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-ansible.yaml</b>	『 <a href="#">コンテナ化された Red Hat Ceph を持つオーバークラウドのデプロイ</a> 』の「 <a href="#">オーバークラウドデプロイメントの開始</a> 」
<b>ceph-mds.yaml</b> を使用して Ceph メタデータサーバーをデプロイする。	<b>-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-mds.yaml</b>	『 <a href="#">コンテナ化された Red Hat Ceph を持つオーバークラウドのデプロイ</a> 』の「 <a href="#">オーバークラウドデプロイメントの開始</a> 」
NFS バックエンドに CephFS を使用した manila サービスをデプロイする。director で NFS-Ganesha を設定する。	<b>-e /usr/share/openstack-tripleo-heat-templates/environments/manila-cephfs-ganesha-config.yaml</b>	「 <a href="#">manila-cephfs-ganesha-config.yaml</a> 」

以下は、CephFS を使用するための NFS-Ganesha、Ceph クラスター、Ceph MDS、および StorageNFS 分離ネットワークをデプロイするオプションを取り込んだ **openstack overcloud deploy command** コマンドの例です。

```
[stack@undercloud ~]$ openstack overcloud deploy \
--templates /usr/share/openstack-tripleo-heat-templates \
-n /usr/share/openstack-tripleo-heat-templates/network_data_ganesha.yaml \
-r /home/stack/roles_data.yaml \
-e /home/stack/containers-default-parameters.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/net-single-nic-with-vlans.yaml \
-e /home/stack/network-environment.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-ansible.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-mds.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/manila-cephfs-ganesha-config.yaml
```

openstack overcloud deploy コマンドに関する補足情報は、『director のインストールと使用方法』の「CLI ツールを使用したオーバークラウドの作成」セクションを参照してください。

### 2.2.3.1. StorageNFS および network\_data\_ganesha.yaml ファイル

コンポーザブルネットワークを使用すると、カスタムネットワークを定義して、それを任意のロールに割り当てることができます。StorageNFS コンポーザブルネットワークは、標準の **network\_data.yaml** ファイルではなく、**network\_data\_ganesha.yaml** ファイルを使用して設定します。これらのロールは、共に **/usr/share/openstack-tripleo-heat-templates** ディレクトリーから利用可能です。

**network\_data\_ganesha.yaml** ファイルには、StorageNFS 分離ネットワークを定義する追加のセクションが含まれます。デフォルトの設定がほとんどのインストールで機能しますが、YAML ファイルを編集してご自分のネットワーク設定 (VLAN ID、サブネット等) を追加しなければならない場合があります。

```
name: StorageNFS
enabled: true
vip: true
name_lower: storage_nfs
vlan: 70
ip_subnet: '172.16.4.0/24'
allocation_pools: [{'start': '172.16.4.4', 'end': '172.16.4.250'}]
ipv6_subnet: 'fd00:fd00:fd00:7000::/64'
ipv6_allocation_pools: [{'start': 'fd00:fd00:fd00:7000::10', 'end': 'fd00:fd00:fd00:7000:ffff:ffff:ffff:ffff'}]
```

コンポーザブルネットワークに関する詳しい情報は、『オーバークラウドの高度なカスタマイズ』の「コンポーザブルネットワーク」の章を参照してください。

### 2.2.3.2. manila-cephfsganesha-config.yaml

CephFS バックエンドを定義するための統合環境ファイルは、アンダークラウドノードの以下のパスにあります。

```
/usr/share/openstack-tripleo-heat-templates/environments/manila-cephfsganesha-config.yaml
```

**manila-cephfsganesha-config.yaml** 環境ファイルには、Shared File Systems サービスのデプロイに関する設定が含まれます。バックエンドのデフォルト設定は、ほとんどの環境で機能するはずですが。Shared File Systems サービスをデプロイする際に director が使用するデフォルト値を、以下の例で示します。

```
[stack@undercloud ~]$ cat /usr/share/openstack-tripleo-heat-templates/environments/manila-cephfsganesha-config.yaml
# A Heat environment file which can be used to enable a
# a Manila CephFS-NFS driver backend.
resource_registry:
  OS::TripleO::Services::ManilaApi: ../docker/services/manila-api.yaml
  OS::TripleO::Services::ManilaScheduler: ../docker/services/manila-scheduler.yaml
  # Only manila-share is pacemaker managed:
  OS::TripleO::Services::ManilaShare: ../docker/services/pacemaker/manila-share.yaml
  OS::TripleO::Services::ManilaBackendCephFs: ../puppet/services/manila-backend-cephfs.yaml
  # ceph-nfs (ganesha) service is installed and configured by ceph-ansible
  # but it's still managed by pacemaker
  OS::TripleO::Services::CephNfs: ../docker/services/ceph-ansible/ceph-nfs.yaml
```

```
parameter_defaults:
  ManilaCephFSBackendName: cephfs ❶
  ManilaCephFSDriverHandlesShareServers: false ❷
  ManilaCephFSCephFSAuthId: 'manila' ❸
  ManilaCephFSCephFSEnableSnapshots: false ❹
  # manila cephfs driver supports either native cephfs backend - 'CEPHFS'
  # (users mount shares directly from ceph cluster), or nfs-ganesha backend -
  # 'NFS' (users mount shares through nfs-ganesha server)
  ManilaCephFSCephFSProtocolHelperType: 'NFS'
```

`parameter_defaults` ヘッダーから設定が始まります。具体的には、このヘッダーからの設定により、`resource_registry` で設定したデフォルト値を上書きすることができます。これには、CephFS バックエンドのデフォルトを設定する `OS::Tripleo::Services::ManilaBackendCephFs` で定義した値も含まれます。

- ❶ **ManilaCephFSBackendName:** CephFS バックエンドの manila 設定の名前を定義します。ここでは、デフォルトのバックエンド名は `cephfs` です。
- ❷ **ManilaCephFSDriverHandlesShareServers:** 共有サーバーのライフサイクルをコントロールします。`false` に設定すると、ドライバーはライフサイクルを処理しません。サポートされるオプションはこれだけです。
- ❸ **ManilaCephFSCephFSAuthId:** Ceph クラスタにアクセスするために director が `manila` サービス用に作成する Ceph 認証 ID を定義します。
- ❹ **ManilaCephFSCephFSEnableSnapshots:** スナップショットのアクティブ化をコントロールします。値が `false` であれば、スナップショットが有効ではないことを意味します。この機能は現在サポートされていません。

環境ファイルについての詳細は、『[オーバークラウドの高度なカスタマイズ](#)』の「[環境ファイル](#)」を参照してください。

## 2.2.4. デプロイメント後設定の完了

ユーザーにアクセスを許可する前に、以下に示す 2 つのデプロイメント後設定項目を完了する必要があります。

- neutron StorageNFS ネットワークをデータセンターの StorageNFS 分離ネットワークにマッピングする。
- デフォルトのファイル共有種別を作成する。

これらのステップが完了したら、テナントコンピュートインスタンスは NFS 共有を作成し、そのアクセスを許可し、マウントすることができます。

### 2.2.4.1. 分離ネットワークの設定

新しい StorageNFS 分離ネットワークを neutron 共有プロバイダーネットワークにマッピングする必要があります。NFS-Ganesha ゲートウェイの提供するファイル共有のエクスポート場所にアクセスするために、コンピュートノードの仮想マシンをこの neutron ネットワークにアタッチします。

Shared File Systems サービスのネットワークセキュリティに関する一般情報は、『[Security and Hardening Guide](#)』の「[Hardening the Shared File System \(Manila\)](#)」の章を参照してください。

## 手順

**openstack network create** コマンドにより、StorageNFS neutron ネットワークの設定を定義します。以下のオプションを設定してこのコマンドを実行します。

- **--provider-physical-network**: tripleo-heat-templates の NeutronBridgeMappings で別途 br-isolated ブリッジのタグを設定していない限り、デフォルト値 **datacentre** を使用します。
- **--provider-segment**: Heat テンプレート (/usr/share/openstack-tripleo-heat-templates/network\_data\_ganeshia.yaml) で StorageNFS 分離ネットワークに設定した vlan の値を使用します。デプロイ時に分離ネットワークの定義を変更していない限り、この値は 70 です。
- **--provider-network-type**: 値 **vlan** を使用します。

このコマンドを使用するには、以下の手順を実施します。

1. アンダークラウドノードから以下のコマンドを実行します。

```
[stack@undercloud ~]$ source ~/overcloudrc
```

2. アンダークラウドノードで **openstack network create** コマンドを実行して、StorageNFS ネットワークを作成します。

```
[stack@undercloud-0 ~]$ openstack network create StorageNFS --share --provider-network-type vlan --provider-physical-network datacentre --provider-segment 70
+-----+
| Field          | Value
+-----+
| admin_state_up | UP
| availability_zone_hints |
| availability_zones |
| created_at      | 2018-09-17T21:12:49Z
| description     |
| dns_domain      | None
| id              | cd272981-0a5e-4f3d-83eb-d25473f5176e
| ipv4_address_scope | None
| ipv6_address_scope | None
| is_default      | False
| is_vlan_transparent | None
| mtu             | 1500
| name            | StorageNFS
| port_security_enabled | True
| project_id      | 3ca3408d545143629cd0ec35d34aea9c
| provider-network-type | vlan
| provider-physical-network | datacentre
| provider-segment | 70
| qos_policy_id   | None
| revision_number | 3
| router:external | Internal
| segments        | None
| shared          | True
| status          | ACTIVE
| subnets        |
| tags            |
| updated_at      | 2018-09-17T21:12:49Z
+-----+
```

### 2.2.4.2. 共有プロバイダー StorageNFS ネットワークのセットアップ

neutron 共有プロバイダーネットワークに対応する StorageNFSSubnet を作成します。アンダークラウドの storage\_nfs\_subnet と同じサブネットを設定します。ただし、このサブネットと対応するアンダークラウドのサブネットの割り当て範囲が重複しないようにしてください。このサブネットは NFS 共有提供専用なので、ゲートウェイは必要ありません。

#### 要件

- 割り当てプールの IP 範囲 (開始および終了アドレス)
- サブネットの IP 範囲

#### 手順

1. オーバークラウドノードにログインします。
2. 下記のコマンド例を使用して、ネットワークをプロビジョニングします。必要に応じて値を更新してください。
  - a. **start=172.16.4.150,end=172.16.4.250** を、実際のネットワークの IP の値に置き換えます。
  - b. **172.16.4.0/24** を、実際のネットワークの正しいサブネット範囲に置き換えます。

```
[stack@undercloud-0 ~]$ openstack subnet create --allocation-pool
start=172.16.4.150,end=172.16.4.250 --dhcp --network StorageNFS --subnet-range 172.16.4.0/24 --
gateway none StorageNFSSubnet
```

```
+-----+
| Field      | Value
+-----+
| allocation_pools | 172.16.4.150-172.16.4.250
| cidr        | 172.16.4.0/24
| created_at   | 2018-09-17T21:22:14Z
| description   |
| dns_nameservers |
| enable_dhcp   | True
| gateway_ip   | None
| host_routes  |
| id          | 8c696d06-76b7-4d77-a375-fd2e71e3e480
| ip_version   | 4
| ipv6_address_mode | None
| ipv6_ra_mode | None
| name         | StorageNFSSubnet
| network_id   | cd272981-0a5e-4f3d-83eb-d25473f5176e
| project_id   | 3ca3408d545143629cd0ec35d34aea9c
| revision_number | 0
| segment_id   | None
| service_types |
| subnetpool_id | None
| tags         |
| updated_at   | 2018-09-17T21:22:14Z
+-----+
```

### 2.2.4.3. デフォルトファイル共有種別のセットアップ

Shared File Systems サービスでは、ファイル共有の種別を定義することができます。これを使用して、特定の設定のファイル共有を作成することができます。ファイル共有の種別は、Block Storage のボリューム種別に類似した機能を持ちます。それぞれの種別には関連する設定 (追加の仕様) があり、ファイル共有の作成時に種別を呼び出すと、それらの設定が適用されます。

OpenStack director ではデフォルトのファイル共有種別が必要です。ユーザーにクラウドへのアクセスを許可する前に、このデフォルトのファイル共有種別を作成する必要があります。NFS バックエンドに CephFS を使用する構成では、**manila type-create** コマンドを使用します。

```
manila type-create default false
```

ファイル共有種別に関する情報は、『ストレージガイド』の「[共有種別の作成と管理](#)」の章を参照してください。

## 第3章 NFS バックエンドに CEPHFS を使用する構成が正常にデプロイされたことの検証

OpenStack Shared File Systems サービス (manila) の NFS バックエンドに CephFS を使用する構成をデプロイすることで、オーバークラウド環境に新たな要素が追加されます。

新たなオーバークラウド要素は以下のとおりです。

- StorageNFS ネットワーク
- コントローラー上の Ceph MDS サービス
- コントローラー上の NFS-Ganesha サービス

NFS バックエンドに CephFS を使用した Shared File Systems サービスの使用に関する補足情報は、『ストレージガイド』の「[Shared File Systems サービス](#)」の章を参照してください。

ユーザーに利用を許可する前に、クラウド管理者は NFS バックエンドに CephFS を使用する環境が安定して動作することを確認する必要があります。

### 前提条件

- 「[2章 NFS バックエンドに CephFS を使用する構成のインストール](#)」の手順が完了していること

### 3.1. STORAGENFS 分離ネットワークが作成されていることの確認

Shared File Systems サービスの NFS バックエンドに CephFS を使用する構成をデプロイするのに使用する `network_data_ganesha.yaml` ファイルにより、StorageNFS VLAN が作成されます。

```
- name: StorageNFS
  enabled: true
  vip: true
  name_lower: storage_nfs
  vlan: 310
  ip_subnet: '172.16.4.0/24'
  allocation_pools: [{'start': '172.16.4.4', 'end': '172.16.4.250'}]
  ipv6_subnet: 'fd00:fd00:fd00:7000::/64'
  IPv6_allocation_pools: [{'start': 'fd00:fd00:fd00:7000::10', 'end': 'fd00:fd00:fd00:7000:ffff:ffff:ffff:ffff'}]
```

StorageNFS 分離ネットワークが存在することを確認するには、以下の手順を実施します。

### 手順

1. オーバークラウドのコントローラーのいずれかにログインします。
2. 以下のコマンドを実行して接続されたネットワークを確認し、`network_data_ganesha.yaml`により設定したとおりの VLAN が存在することを確認します。

```
$ ip a
15: vlan310: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UNKNOWN group default qlen 1000
    link/ether 32:80:cf:0e:11:ca brd ff:ff:ff:ff:ff:ff
    inet 172.16.4.4/24 brd 172.16.4.255 scope global vlan310
```

```

valid_lft forever preferred_lft forever
inet 172.16.4.7/32 brd 172.16.4.255 scope global vlan310
valid_lft forever preferred_lft forever
inet6 fe80::3080:cfff:fe0e:11ca/64 scope link
valid_lft forever preferred_lft forever

```

### 3.2. CEPH MDS サービスの確認

Ceph MDS サービスのステータスを確認するには、**systemctl status** コマンドを使用します。

#### 手順

1. すべてのコントローラーで以下のコマンドを実行して、MDS docker コンテナのステータスを確認します。

```
$ systemctl status ceph-mds@<CONTROLLER-HOST>
```

以下に例を示します。

```

ceph-mds@controller-0.service - Ceph MDS
  Loaded: loaded (/etc/systemd/system/ceph-mds@.service; enabled; vendor preset:
disabled)
  Active: active (running) since Tue 2018-09-18 20:11:53 UTC; 6 days ago
  Main PID: 65066 (docker-current)
  CGroup: /system.slice/system-ceph\x2dmads.slice/ceph-mds@controller-0.service
          └─65066 /usr/bin/docker-current run --rm --net=host --memory=4g --cpu-
quota=100000 -v /var/lib/ceph:/var/lib/ceph:z -v /etc/ceph:/etc/ceph:z -v
/var/run/ceph:/var/run/ceph:z -v /etc/localtime:/etc/...

```

### 3.3. CEPH クラスターのステータスの確認

Ceph クラスターのステータスを確認するには、以下の手順を実施します。

#### 手順

1. アクティブなコントローラーにログインします。
2. 次のコマンドを実行します。

```

$ sudo ceph -s

cluster:
  id: 3369e280-7578-11e8-8ef3-801844eeec7c
  health: HEALTH_OK

services:
  mon: 3 daemons, quorum overcloud-controller-1,overcloud-controller-2,overcloud-
controller-0
  mgr: overcloud-controller-1(active), standbys: overcloud-controller-2, overcloud-controller-0
  mds: cephfs-1/1/1 up {0=overcloud-controller-0=up:active}, 2 up:standby
  osd: 6 osds: 6 up, 6 in

```





## 注記

1つのアクティブな MDS と 2つのスタンバイ状態の MDS があるはずです。

3. Ceph ファイルシステムのステータスをより詳細に確認するには、以下のコマンドを実行します。<b>cephfs</b> は、Ceph ファイルシステムの名前に置き換えてください。

```
$ sudo ceph fs ls
```

```
name: cephfs, metadata pool: manila_metadata, data pools: [manila_data]
```

## 3.4. NFS-GANESHA および MANILA-SHARE サービスのステータスの確認

NFS-Ganesha および manila-share サービスのステータスを確認するには、以下の手順を実施します。

### 手順

1. いずれかのコントローラーから以下のコマンドを実行して、<b>ceph-nfs</b> および <b>openstack-manila-share</b> が起動していることを確認します。

```
$ pcs status
```

```
ceph-nfs (systemd:ceph-nfs@pacemaker): Started overcloud-controller-1
```

```
Docker container: openstack-manila-share [192.168.24.1:8787/rhosp13/openstack-manila-share:pcmklatest]
```

```
openstack-manila-share-docker-0 (ocf::heartbeat:docker): Started overcloud-controller-1
```

## 3.5. MANILA-API サービスがスケジューラーおよびファイル共有サービスを認識していることの確認

<b>manila-api</b> サービスがスケジューラーおよびファイル共有サービスを認識していることを確認するには、以下の手順を実施します。

### 手順

1. アンダークラウドにログインします。
2. 次のコマンドを実行します。

```
$ source /home/stack/overcloudrc
```

3. 以下のコマンドを実行して、<b>manila-scheduler</b> および <b>manila-share</b> が有効であることを確認します。

```
$ manila service-list
```

```
| Id | Binary          | Host          | Zone | Status | State | Updated_at |
```

```
| 2 | manila-scheduler | hostgroup    | nova | enabled | up   | 2018-08-08T04:15:03.000000 |
```

```
| 5 | manila-share    | hostgroup@cephfs | nova | enabled | up   | 2018-08-08T04:15:03.000000 |
```

