



Red Hat OpenStack Platform 16.2

ネイティブ CephFS を使用する Shared File Systems サービスのデプロイ

Red Hat OpenStack Platform における、ネイティブ CephFS を使用する Shared File Systems サービスの理解、使用および管理

Red Hat OpenStack Platform 16.2 ネイティブ CephFS を使用する Shared File Systems サービスのデプロイ

Red Hat OpenStack Platform における、ネイティブ CephFS を使用する Shared File Systems サービスの理解、使用および管理

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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

Red Hat OpenStack Platform 環境におけるネイティブ Red Hat Ceph File System (CephFS) バックエンドを使用する Shared File Systems サービス (manila) のインストール、設定、および検証について

目次

多様性を受け入れるオープンソースの強化	3
RED HAT ドキュメントへのフィードバック (英語のみ)	4
第1章 ネイティブ CEPHFS を使用する SHARED FILE SYSTEMS サービス	5
1.1. ネイティブドライバを使用する CEPHFS	5
1.2. ネイティブ CEPHFS バックエンドのセキュリティ	6
第2章 ネイティブ CEPHFS デプロイメント	8
2.1. 要件	8
2.2. ファイル共有	9
2.3. ネイティブ CEPHFS で使用される分離ネットワーク	9
2.4. CEPH-ANSIBLE パッケージのインストール	9
2.5. 環境のデプロイ	10
第3章 デプロイメント後の設定	13
3.1. ストレージプロバイダーネットワークの作成	13
3.2. ストレージプロバイダーネットワークの設定	14
3.3. ストレージプロバイダーネットワークにおけるロールベースアクセス制御の設定	14
3.4. デフォルトのファイル共有種別の設定	15
第4章 ネイティブ CEPHFS バックエンドがデプロイされたことの検証	17
4.1. 分離ストレージネットワークが作成されていることの確認	17
4.2. CEPH MDS サービスの確認	17
4.3. CEPH クラスターのステータスの確認	18
4.4. MANILA-SHARE サービスのステータスの確認	19
4.5. 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	19

多様性を受け入れるオープンソースの強化

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

RED HAT ドキュメントへのフィードバック (英語のみ)

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

Jira でドキュメントのフィードバックを提供する

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

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

第1章 ネイティブ CEPHFS を使用する SHARED FILE SYSTEMS サービス

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 Storage 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 Storage クラスタ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、ネイティブ Ceph File System プロトコルを使用してそのファイル共有にアクセスすることができます。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。

今回のリリースでは、director はネイティブ CephFS バックエンドを使用する Shared File Systems をオーバークラウドにデプロイすることができます。



重要

本書では、ネイティブ CephFS NAS プロトコルを通じて、Red Hat OpenStack Platform Cloud にセルフサービスの Shared File Systems サービス (manila) を提供するためのネイティブ CephFS のデプロイメントおよび使用方法について説明します。このタイプのデプロイメントでは、ゲスト仮想マシンが Ceph パブリックネットワークおよびインフラストラクチャーにアクセスできる必要があります。汎用の OpenStack Platform デプロイメントには適していない許容信頼モデルが必要なため、信頼済み OpenStack Platform テナントでのみネイティブ CephFS をデプロイしてください。



重要

従来のテナント信頼モデルを使用する汎用の OpenStack Platform デプロイメントでは、NFS プロトコルを介して CephFS をデプロイすることができます。信頼モデルの詳細は、[ネイティブ CephFS バックエンドのセキュリティ](#) を参照してください。NFS バックエンドに CephFS を使用する設定についての詳細は、[Deploying the Shared File Systems service with CephFS through NFS](#) を参照してください。

1.1. ネイティブドライバを使用する CEPHFS

CephFS ネイティブドライバは、OpenStack Shared File Systems サービス (manila) と Red Hat Ceph Storage を結び付けます。Red Hat OpenStack (RHOSP) director を使用する場合、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) などの Ceph デモンならびに Shared File Systems サービスをホストします。

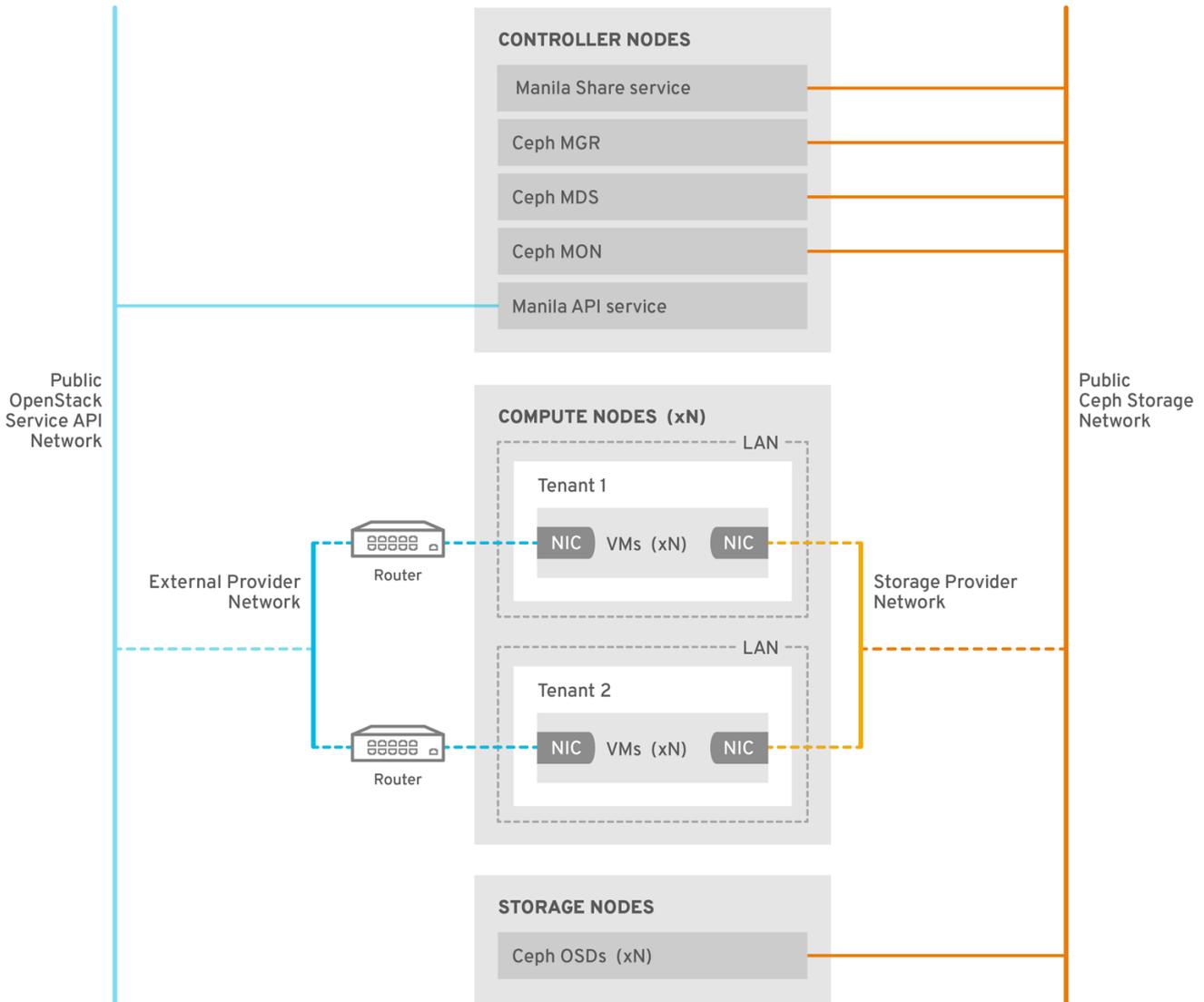
コンピュータノードは1つまたは複数のプロジェクトをホストすることができます。以前はテナントと呼ばれていたプロジェクトは、以下の図では白い長方形で表示されます。プロジェクトには、ユーザー管理の仮想マシンが含まれています。これは、2つの NIC を持つグレーのボックスで表されます。ceph および manila デモンプロジェクトにアクセスするには、パブリック Ceph ストレージネットワーク経由でデモンに接続します。

このネットワーク上で、Ceph オブジェクトストレージデモン (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ができます。プロジェクトがホストするインスタンスまたは仮想マシン (VM) 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プロジェク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、プロジェクト上で動作する仮想マシンをパブリック Ceph ス

トレイ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、確実なプロジェクト間の分離、およびセキュリティー確保のために、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。この設定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

1.2. ネイティブ CEPHFS バックエンドのセキュリティー

ネイティブ CephFS バックエンドには、Red Hat OpenStack Platform (RHOSP) テナントに許容信頼モデルが必要です。この信頼モデルは、OpenStack Platform が提供するサービスの背後にあるインフラストラクチャーにユーザーが直接アクセスするのを意図的にブロックする汎用の OpenStack Platform クラウドには適切ではありません。

ネイティブ CephFS を使用する場合、ユーザーコンピュートインスタンスは Ceph サービスのデーモンが公開される Ceph パブリックネットワークに直接接続されます。ユーザー仮想マシンで実行される CephFS クライアントは Ceph サービスデーモンと連携し、ファイルデータブロックの読み取りおよび書き込みのために RADOS と直接対話します。

Shared File Systems (manila) 共有サイズを適用する CephFS クォータは、Red Hat OpenStack Platform (RHOSP) ユーザーが所有する仮想マシンなど、クライアント側で適用されます。ユーザー仮想マシンのクライアント側ソフトウェアは最新ではない可能性があるため、重要なクラウドインフラストラクチャーが Ceph サービスポートをターゲットにする悪意のあるまたは意図的ではないが有害なソフトウェアに対して脆弱になる可能性があります。

信頼できるユーザーがクライアント側ソフトウェアを最新の状態に維持する環境でのみ、ネイティブ CephFS をバックエンドとしてデプロイします。Ceph Storage インフラストラクチャーに影響を与える可能性のあるソフトウェアが仮想マシンで実行されないようにしてください。

数多くの信頼できないユーザーに対応する汎用 Red Hat OpenStack Platform (RHOSP) デプロイメントについては、NFS バックエンドに CephFS を使用してください。NFS バックエンドに CephFS を使用する設定についての詳細は、[NFS 経由での CephFS を使用した Shared File Systems サービスのデプロイ](#) を参照してください。

ユーザーがクライアント側のソフトウェアを最新版に維持できない、また仮想マシンから有害なソフトウェアを除外できない状況でも、NFS バックエンドに CephFS を使用する設定を使用すると、ユーザーは NFS サーバーの公開側にだけアクセスでき、Ceph インフラストラクチャー自体にはアクセスできません。NFS は同様の協調的なクライアントを必要としないので、最悪の場合でも、ユーザー仮想マシンからの攻撃により、NFS ゲートウェイはダメージを受ける可能性がありますが、その背後にある Ceph ストレージインフラストラクチャーはダメージを受けません。

ネイティブ CephFS バックエンドを信頼できる全ユーザーに公開できますが、以下のセキュリティー対策を実施する必要があります。

- ストレージネットワークをプロバイダーネットワークとして設定する。
- ロールベースのアクセス制御 (RBAC) ポリシーを適用して、Storage プロバイダーネットワークのセキュリティーを保護する。
- プライベートファイル共有種別を作成します。

第2章 ネイティブ CEPHFS デプロイメント

Red Hat OpenStack Platform (RHOSP) 環境への一般的なネイティブ Ceph ファイルシステム (CephFS) インストールには、以下のコンポーネントが含まれます。

- 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)、および Shared File Systems (manila) サービスを実行する RHOSP 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、1つまたは複数の専用のノードを持つ場合があります。
- Ceph Storage ノードで実行され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 Storage クラスタ。
- クライアントが Ceph サービスデーモンと通信できる Ceph パブリックネットワークとして機能する分離ストレージネットワーク。これを容易にするため、ユーザーはストレージネットワークをプロバイダーネットワークとして使用し、仮想マシンを接続して CephFS 共有をマウントします。



重要

CephFS ネイティブドライバと共に Shared File Systems サービス (manila) を使用して、Manila CSI で OpenShift Container Platform にファイル共有を提供することはできません。Red Hat ではこのタイプのデプロイメントをサポートしないためです。詳細は、Red Hat のサポートにお問い合わせください。

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 System サービスはネイティブ CephFS をバックエンドとして使用することができます。director の管理する統合デプロイメントにおいて、ネイティブ CephFS をインストールすることができます。

director がオーバークラウドに CephFS バックエンドを持つ Shared File Systems サービスをデプロイする場合、これにより必要なデータセンターのストレージネットワークが自動的に作成されます。ただし、オーバークラウド上に対応するストレージプロバイダーネットワークを作成する必要があります。詳細は、[デプロイメント後の設定](#) を参照してください。

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 オーバークラウドネットワーク](#) を参照してください。

ノードの `/var/lib/config-data/puppet-generated/manila/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。Red Hat は、director が管理する Shared File Systems サービスのデプロイメントのみをサポートします。

2.1. 要件

以下の要件を満たす場合には、新規または既存の Red Hat OpenStack Platform (RHOSP) 環境と共にネイティブ CephFS バックエンドをデプロイすることができます。

- Red Hat OpenStack Platform バージョン 16.1 以降を使用する。
- ネイティブ CephFS バックエンドと同時に、新しい Red Hat Ceph Storage クラスタを設定する。Ceph Storage のデプロイ方法に関する情報は、[Deploying an overcloud with containerized Red Hat Ceph](#) を参照してください。



重要

RHOSP Shared File Systems サービス (manila) とネイティブ CephFS バックエンドの組み合わせは、Red Hat Ceph Storage バージョン 4.1以降との使用でサポートされます。システムにインストールされている Ceph Storage のバージョンを確認する方法についての詳細は、[What are the Red Hat Ceph Storage releases and corresponding Ceph package versions?](#) を参照してください。

- Shared File Systems サービスをコントローラーノードにインストールします。これがデフォルトの動作です。
- Shared File Systems サービスには、CephFS バックエンドのインスタンスを1つだけ使用してください。

2.2. ファイル共有

Shared File Systems サービス (manila)、Ceph File System (CephFS)、および NFS バックエンドに CephFS を使用する設定では、ファイル共有の取り扱いが異なります。

Shared File Systems サービスが提供するそれぞれのファイル共有は、個別のファイルシステムの名前空間であり、また特定のサイズを持つストレージのユニットです。元来、共有ファイルシステムのストレージでは、複数のクライアントが任意のファイル共有に接続してデータの読み取りおよび書き込みが可能ですが、クライアントが接続できるためには、Shared File Systems サービスのアクセス制御 API を通じて各クライアントにファイル共有へのアクセス権限を付与する必要があります。

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。

ネイティブ CephFS 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 CephFS プロトコルが使用されます。アクセス制御は、CephFS ユーザー名を使用する CephX 認証スキームで実行されます。

2.3. ネイティブ CEPHFS で使用される分離ネットワーク

ネイティブ CephFS デプロイメントでは、director が Ceph パブリックネットワークとしてデプロイする分離ストレージネットワークを使用されます。クライアントはこのネットワークを使用して、さまざまな Ceph インフラストラクチャーサービスデーモンと通信します。ネットワークの分離の詳細は、[オーバークラウドの高度なカスタマイズガイドの基本的なネットワークの分離](#) を参照してください。

2.4. CEPH-ANSIBLE パッケージのインストール

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. **ceph-ansible** パッケージをインストールします。

```
[stack@undercloud-0 ~]$ sudo dnf install -y ceph-ansible
```

```
[stack@undercloud-0 ~]$ sudo dnf list ceph-ansible
...
Installed Packages
ceph-ansible.noarch 4.0.23-1.el8cp @rhelosp-ceph-4-tools
```

2.5. 環境のデプロイ

環境をデプロイする準備が整ったら、ネイティブ CephFS バックエンドの設定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。

openstack overcloud deploy 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。

アクション	オプション	追加情報
network_data.yaml を使用してネットワーク設定を指定します。	<code>[filename] -n /usr/share/openstack-tripleo-heat-templates/network_data.yaml</code>	カスタム環境ファイルを使用して、このネットワークデータ環境ファイルで指定したデフォルトネットワークの値を上書きすることができます。これは、分離ネットワークを使用する際に利用可能なデフォルトのネットワークデータファイルです。このファイルは、簡潔にするために openstack overcloud deploy コマンドから省略することができます。
ceph-ansible.yaml を使用して Ceph デーモンをデプロイする。	<code>-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-ansible.yaml</code>	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の オーバークラウドデプロイメントの開始
ceph-mds.yaml を使用して Ceph メタデータサーバーをデプロイする。	<code>-e /usr/share/openstack-tripleo-heat-templates/environments/ceph-ansible/ceph-mds.yaml</code>	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の オーバークラウドデプロイメントの開始
ネイティブ CephFS バックエンドを使用する manila サービスをデプロイする	<code>-e /usr/share/openstack-tripleo-heat-templates/environments/manila-cephfsnative-config.yaml</code>	環境ファイル

以下の例は、Ceph クラスター、Ceph MDS、ネイティブ CephFS バックエンド、および Ceph クラスターに必要なネットワークをデプロイするオプションが含まれる **openstack overcloud deploy command** コマンドを示しています。

```
[stack@undercloud ~]$ openstack overcloud deploy \
...
-n /usr/share/openstack-tripleo-heat-templates/network_data.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.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-cephfsnative-config.yaml
```

openstack overcloud deploy コマンドについての詳しい情報は、**Director のインストールと使用方法** の **デプロイメントコマンド** を参照してください。

2.5.1. 環境ファイル

ネイティブ CephFS バックエンドを定義する統合環境ファイルは、アンダークラウドノードのパス (**/usr/share/openstack-tripleo-heat-templates/environments/manila-cephfsnative-config.yaml**) に置かれます。

manila-cephfsnative-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例で示します。

```
[stack@undercloud ~]$ cat /usr/share/openstack-tripleo-heat-templates/environments/manila-
cephfsnative-config.yaml

# A Heat environment file which can be used to enable a
# a Manila CephFS Native driver backend.
resource_registry:
  OS::TripleO::Services::ManilaApi: ../deployment/manila/manila-api-container-puppet.yaml
  OS::TripleO::Services::ManilaScheduler: ../deployment/manila/manila-scheduler-container-
puppet.yaml
  # Only manila-share is pacemaker managed:
  OS::TripleO::Services::ManilaShare: ../deployment/manila/manila-share-pacemaker-puppet.yaml
  OS::TripleO::Services::ManilaBackendCephFs: ../deployment/manila/manila-backend-cephfs.yaml

parameter_defaults:
  ManilaCephFSBackendName: cephfs ①
  ManilaCephFSDriverHandlesShareServers: false ②
  ManilaCephFSCephFSAuthId: 'manila' ③
  ManilaCephFSCephFSEnableSnapshots: true ④
  ManilaCephFSCephVolumeMode: '0755' ⑤
  # 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: 'CEPHFS' ⑥
```

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。

- ① **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。
- ② **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。CephFS バックエンドでサポートされるオプションはこれだけです。
- ③ **ManilaCephFSCephFSAuthId**: Ceph クラスターにアクセスするために **director** が manila サービ

スに作成する Ceph 認証 ID を定義します。

- 4 **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。Ceph Storage 4.1 以降ではスナップショットがサポートされますが、このパラメーターのデフォルト値は **false** です。この値を **true** に設定すると、ドライバーは **snapshot_support** 機能を manila スケジューラーに報告します。
- 5 **ManilaCephFSCephVolumeMode** は、ネイティブ CephFS バックエンド上に作成される manila 共有に設定する UNIX パーミッションを制御します。デフォルト値は **755** です。
- 6 ネイティブ CephFS ドライバーを使用するには、**ManilaCephFSCephFSProtocolHelperType** を **CEPHFS** に設定する必要があります。

環境ファイルについての詳細は、Director のインストールと使用方法の [環境ファイル](#) を参照してください。

第3章 デプロイメント後の設定

NFS 共有を作成し、ユーザーにアクセス権限を付与し、NFS 共有をマウントする前に、いくつかのデプロイメント後設定タスクを完了する必要があります。

- Networking サービス (neutron) ストレージネットワークを分散データセンターストレージネットワークにマッピングします。
- カスタムロールベースアクセス制御 (RBAC) のみ、ストレージプロバイダーネットワークを信頼済みテナントで利用できるようにします。ストレージプロバイダーネットワークはグローバルに共有しないでください。
- プライベートファイル共有種別を作成します。
- 特定の信頼済みテナントにアクセス権限を付与します。

これらのステップを完了したら、テナントコンピューティングインスタンスはネイティブ CephFS 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。

3.1. ストレージプロバイダーネットワークの作成

新しい分離ストレージネットワークを Networking (neutron) 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。ネイティブ CephFS 共有のエクスポート場所にアクセスするために、コンピューティングノードの仮想マシンを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情報は、[セキュリティおよび強化ガイドの Shared File System \(Manila\) の強化](#) を参照してください。

手順

openstack network create コマンドにより、ストレージ neutron ネットワークの設定を定義します。

1. アンダークラウドノードから、以下のコマンドを入力します。

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

2. アンダークラウドノードで、ストレージネットワークを作成します。

```
(overcloud) [stack@undercloud-0 ~]$ openstack network create Storage --provider-network-type vlan --provider-physical-network datacentre --provider-segment 30
```

以下のオプションを設定してこのコマンドを入力することができます。

- **--provider-physical-network** オプション: 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** オプション: ネットワーク環境ファイルで Storage 分離ネットワークに設定した値を使用します。これがカスタマイズされない場合、デフォルトの環境ファイルは `/usr/share/openstack-tripleo-heat-templates/network_data.yaml` になります。分離ネットワークの定義を変更していない限り、Storage ネットワークに関連付けられた VLAN の値は **30** です。
- **--provider-network-type** オプション: 値 **vlan** を使用します。

3.2. ストレージプロバイダーネットワークの設定

neutron プロバイダーネットワーク上に対応する **StorageSubnet** を作成します。アンダークラウドの **storage_subnet** と同じサブネットを設定します。また、ストレージ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。

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. アンダークラウドノードから、以下のコマンドを入力します。

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

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。値を実際の環境に応じて更新します。

```
(overcloud) [stack@undercloud-0 ~]$ openstack subnet create \
--allocation-pool start=172.17.3.10,end=172.17.3.149 \
--dhcp \
--network Storage \
--subnet-range 172.17.3.0/24 \
--gateway none StorageSubnet
```

- **--allocation-pool** オプションの場合は、**start=172.17.3.10,end=172.17.3.149** IP の値を、実際のネットワークの IP の値に置き換えます。
- **--subnet-range** オプション: **172.17.3.0/24** のサブネット範囲を実際のネットワークのサブネット範囲に置き換えます。

3.3. ストレージプロバイダーネットワークにおけるロールベースアクセス制御の設定

ストレージネットワークを使用することができる信頼済みテナントまたはプロジェクトを特定した後に、Networking サービス (neutron) を使用してそれらのロールベースアクセス制御 (RBAC) ルールを設定します。

要件

ストレージネットワークへのアクセスを必要とするプロジェクトの名前

手順

1. アンダークラウドノードから、以下のコマンドを入力します。

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

2. アクセスが必要なプロジェクトを特定します。

```
(overcloud) [stack@undercloud-0 ~]$ openstack project list
```

```
+-----+
| ID           | Name |
+-----+
| 06f1068f79d2400b88d1c2c33eacea87 | demo |
| 5038dde12dfb44fdaa0b3ee4bfe487ce | service |
| 820e2d9c956644c2b1530b514127fd0d | admin |
+-----+
```

3. 必要なプロジェクトでネットワーク RBAC ルールを作成します。

```
(overcloud) [stack@undercloud-0 ~]$ openstack network rbac create \
--action access_as_shared Storage \
--type network \
--target-project demo
```

ストレージネットワークへのアクセスを必要とするすべてのプロジェクトについて、このステップを繰り返します。

3.4. デフォルトのファイル共有種別の設定

Shared File Systems サービス (manila) を使用して、特定の設定で共有を作成するために使用できる共有タイプを定義できます。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、追加の仕様などの関連する設定があります。ファイル共有の作成時に種別を呼び出すと、その設定が共有ファイルシステムに適用されます。

信頼できないユーザーからネイティブ CephFS バックエンドを保護するために、Red Hat ではデフォルトの共有種別を作成しないことを推奨します。デフォルトの共有種別が存在しない場合、ユーザーは共有種別を指定することを強制され、信頼できるユーザーは排他的アクセス権限を持つカスタムのプライベート共有種別を使用できます。

信頼されていないテナントのデフォルトの共有種別を作成する必要がある場合、ネイティブ CephFS バックエンドからプロビジョニングを分離することができます。

手順

1. アンダークラウドノードから、以下のコマンドを入力します。

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

2. 共有種別に追加の仕様を設定します。

```
(overcloud) [stack@undercloud-0 ~] manila type-create default False
(overcloud) [stack@undercloud-0 ~] manila type-key default set share_backend_name='s!=cephfs'
```

3. プライベートの共有種別を作成して、信頼済みテナントにこの共有種別へのアクセス権限を付与します。

```
(overcloud) [stack@undercloud-0 ~]$ manila type-create --is-public false nativecephfstype false
(overcloud) [stack@undercloud-0 ~]$ manila type-key nativecephfstype set share_backend_name='cephfs'
(overcloud) [stack@undercloud-0 ~]$ manila type-access-add nativecephfstype <trusted_tenant_project_id>
```

ファイル共有種別に関する詳しい情報は、ストレージガイドの [共有タイプの作成](#) を参照してください。

第4章 ネイティブ CEPHFS バックエンドがデプロイされたことの検証

Shared File Systems サービス (manila) のバックエンドとしてネイティブ CephFS をデプロイすると、オーバークラウド環境に以下に示す新たな要素が追加されます。

- ストレージプロバイダーネットワーク
- コントローラーノード上の Ceph MDS サービス

ネイティブ CephFS で共有ファイルシステムサービスを使用する方法の詳細は、[ストレージガイドの Shared File Systems サービスの設定 \(manila\)](#) を参照してください。

ユーザーに利用を許可する前に、クラウド管理者はネイティブ CephFS 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

- [ネイティブ CephFS デプロイメント](#) の手順を完了している。

4.1. 分離ストレージネットワークが作成されていることの確認

Shared File Systems サービスのバックエンドとしてネイティブ CephFS をデプロイするのに使用する **network_data.yaml** ファイルにより、ストレージ VLAN が作成されます。以下の手順を使用して、ストレージ VLAN が正常に作成されていることを確認します。

手順

1. オーバークラウドのコントローラーノードのいずれかにログインします。
2. 接続されたネットワークを確認し、**network_data.yaml** ファイル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
8: vlan30: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UNKNOWN group default qlen 1000
    link/ether 52:9c:82:7a:d4:75 brd ff:ff:ff:ff:ff:ff
    inet 172.17.3.144/24 brd 172.17.3.255 scope global vlan30
        valid_lft forever preferred_lft forever
    inet6 fe80::509c:82ff:fe7a:d475/64 scope link
        valid_lft forever preferred_lft forever
```

4.2. 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

- すべてのコントローラーノードで以下のコマンドを入力して、MDS 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。

```
$ systemctl status ceph-mds@controller-0.service

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 (common)
  Tasks: 16 (limit: 204320)
  Memory: 38.2M
  CGroup: /system.slice/system-ceph\x2dmds.slice/ceph-mds@controller-0.service
          └─60921 /usr/bin/podman run --rm --net=host --memory=32000m --cpus=4 -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/localtime:ro>
```

4.3. 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し、クラスターがアクティブであることを確認します。

手順

1. 任意のコントローラーノードにログインします。
2. Ceph monitor デーモンから、以下のコマンドを入力します。

```
$ sudo podman exec ceph-mon-controller-0 ceph -s
cluster:
  id: 670dc288-cd36-4772-a4fc-47287f8e2ebf
  health: HEALTH_OK

services:
  mon: 3 daemons, quorum controller-1,controller-2,controller-0 (age 14h)
  mgr: controller-1(active, since 8w), standbys: controller-0, controller-2
  mds: cephfs:1 {0=controller-2=up:active} 2 up:standby
  osd: 15 osds: 15 up (since 8w), 15 in (since 8w)

task status:
  scrub status:
    mds.controller-2: idle

data:
  pools: 6 pools, 192 pgs
  objects: 309 objects, 1.6 GiB
  usage: 21 GiB used, 144 GiB / 165 GiB avail
  pgs: 192 active+clean
```



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 があります。

3. Ceph File System の詳細ステータスを表示するには、以下のコマンドを入力します。

```
$ sudo ceph fs ls
```

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



注記

以下の出力例では、**cephfs** は、ユーザーが Shared File Systems サービスを使用して作成する CephFS ファイル共有をホストするために director が作成する Ceph File System の名前です。

4.4. MANILA-SHARE サービスのステータスの確認

manila-share サービスのステータスを確認します。

手順

1. いずれかのコントローラーノードから、**openstack-manila-share** が起動していることを確認します。

```
$ sudo pcs status resources | grep manila
```

```
* Container bundle: openstack-manila-share [cluster.common.tag/rhosp16-openstack-manila-share:pcmklatest]:
```

```
* openstack-manila-share-podman-0 (ocf::heartbeat:podman): Started controller-0
```

4.5. 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

以下の手順を実施して、**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を確認します。

手順

1. アンダークラウドにログインします。
2. 以下のコマンドを入力します。

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

3. 以下のコマンドを入力して、**manila-scheduler** および **manila-share** 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 |
```

