



Red Hat OpenStack Platform 17.1

OVN メカニズムドライバーへの移行

Red Hat OpenStack Platform Networking サービス (neutron) を ML2/OVS メカニズムドライバーから ML2/OVN メカニズムドライバーに移行

Red Hat OpenStack Platform 17.1 OVN メカニズムドライバーへの移行

Red Hat OpenStack Platform Networking サービス (neutron) を ML2/OVS メカニズムドライバーから ML2/OVN メカニズムドライバーに移行

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

Open vSwitch メカニズムドライバー (ML2/OVN) を備えた Modular Layer 2 プラグインから、Open Virtual Networking (ML2/OVN) を備えた Modular Layer 2 プラグインに、Red Hat OpenStack Platform Networking サービス (neutron) を移行する手順

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 OVS から OVN への ML2 メカニズムドライバーの移行計画	5
1.1. OVN および OVS メカニズムドライバーでの機能サポート	6
1.2. ML2/OVS から ML2/OVN へのインプレースマイグレーション: 検証済みのシナリオおよび禁止されるシナリオ	8
第2章 ML2 メカニズムドライバーの OVS から OVN への移行	9
2.1. OVN メカニズムドライバーを移行するための環境の準備	9
2.2. OVS から OVN への ML2 メカニズムドライバーの移行用コンテナイメージの準備	13
2.3. VXLAN OVS デプロイメントから移行するための MTU 削減	15
2.4. ML2 メカニズムドライバーの OVS から OVN への移行	17
第3章 ML2/OVS から ML2/OVN への移行を元に戻す	19
3.1. ML2/OVN 移行を元に戻すためにコントローラーノードの復元	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、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

問題の作成 フォームを使用して、Red Hat OpenStack Services on OpenShift (RHOSO) または Red Hat OpenStack Platform (RHOSP) の以前のリリースのドキュメントに関するフィードバックを提供します。RHOSO または RHOSP ドキュメントの問題を作成すると、その問題は RHOSO Jira プロジェクトに記録され、フィードバックの進行状況を追跡できるようになります。

問題の作成 フォームを完了するには、Jira にログインしていることを確認してください。Red Hat Jira アカウントをお持ちでない場合は、<https://issues.redhat.com> でアカウントを作成できます。

1. 次のリンクをクリックして、**問題の作成** ページを開きます (**問題の作成**)。
2. **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。
3. **Create** をクリックします。

第1章 OVS から OVN への ML2 メカニズムドライバーの移行計画

RHOSP 15.0 以降のすべての新規デプロイメントについて、Red Hat では ML2/OVN をデフォルトのメカニズムドライバーとして選択しました。これは、今日のほとんどのお客様にとって ML2/OVS メカニズムドライバー以上のメリットが即座に得られるためです。継続して ML2/OVN 機能セットの拡張および改善を行っているため、これらのメリットはリリースと共に拡大します。

ML2/OVS メカニズムドライバーは、RHOSP 17.0 で非推奨になりました。いくつかのリリースで、Red Hat は ML2/OVS を ML2/OVN に置き換えています。

非推奨の ML2/OVS メカニズムドライバーは、RHOSP 17 リリースでサポートされます。この間、ML2/OVS ドライバーはメンテナンスモードのままで、バグ修正と通常のサポートを受けられます。ほとんどの新機能開発は ML2/OVN メカニズムドライバーで行われます。

RHOSP 18.0 では、Red Hat は ML2/OVS メカニズムドライバーを完全に削除し、サポートを停止する予定です。

既存の Red Hat OpenStack Platform (RHOSP) デプロイメントで ML2/OVS メカニズムドライバーが使用されている場合、ML2/OVS メカニズムドライバーを ML2/OVN メカニズムドライバーに置き換えるメリットおよび現実性の評価を今すぐ開始してください。移行は、RHOSP 16.2 および RHOSP 17.1 でサポートされています。



注記

ML2/OVS から ML2/OVN への移行を試みる前に、プロアクティブケースを作成する必要があります。プロアクティブケースを作成しない場合、Red Hat では移行をサポートしません。link:<https://access.redhat.com/solutions/2186261>] を参照してください。

この評価の初期段階で、Red Hat Technical Account Manager または Red Hat Global Professional Services との連携を開始してください。移行を選択した場合、Red Hat は必要なプロアクティブケースの作成支援に加えて、以下の基本的な疑問に対する回答や、計画および準備に対するサポートを提供できます。

いつ移行すべきか？

タイミングは、お客様のビジネスニーズや Red Hat による ML2/OVN オファリングに対する継続的な改善のステータスなど、多くの要素に依存します。[OVN および OVS メカニズムドライバーでの機能サポート](#) および [ML2/OVS から ML2/OVN へのインプレースマイグレーション: 検証済みのシナリオおよび禁止されるシナリオ](#) を参照してください。

インプレースマイグレーションか並列移行か？

さまざまな要因に応じて、以下の移行の基本アプローチのいずれかを選択できます。

- 並列移行: ML2/OVN を使用する新しい並列デプロイメントを作成し、運用をそのデプロイメントに移動します。
- インプレースマイグレーション: このドキュメントで説明しているように、`ovn-migration.sh` スクリプトを使用します。Red Hat は、RHOSP director が管理するデプロイメントでのみ `ovn_migration.sh` スクリプトをサポートする点に注意してください。



警告

ML2/OVS から ML2/OVN に移行すると、環境が変更され、完全には元に戻せなくなる可能性があります。適切なバックアップ手順と元に戻す手順に従えば、失敗または中断された移行を元に戻すことができますが、元に戻された OVS 環境は元の環境から変更される可能性があります。実稼働環境で移行を行う前に、プロアクティブケースを作成してください。次に、Red Hat Technical Account Manager または Red Hat Global Professional Services と連携してバックアップおよび移行プランを作成し、実稼働環境に極めて近いステージ環境で移行をテストします。潜在的な移行の元に戻しに備えてバックアップを準備することを選択した場合は、ステージ環境で移行の元に戻しをテストする必要があります。

1.1. OVN および OVS メカニズムドライバーでの機能サポート

OVS から OVN メカニズムドライバーへの移行計画の一環として、Red Hat OpenStack Platform (RHOSP) 機能の可用性を確認します。

機能	OVN RHOS P 16.2	OVN RHOS P 17.1	OVS RHOS P 16.2	OVS RHOS P 17.1	Additional information
OVN と DHCP の組み合わせでのベアメタルマシンのプロビジョニング	いいえ	いいえ	はい	はい	OVN 上の組み込み型 DHCP サーバーは、現状ベアメタルノードをプロビジョニングできません。プロビジョニングネットワーク用に、DHCP を提供できません。iPXE のチェーンブートにはタグ付け (dnsmasq の --dhcp-match) が必要ですが、OVN DHCP サーバーではサポートされていません。 https://bugzilla.redhat.com/show_bug.cgi?id=1622154 を参照してください。
VLAN プロジェクト (テナントネットワーク) の VF (ダイレクト) ポートでのノース/サウスルーティング	いいえ	いいえ	はい	はい	OVN に関する主な制約。Bug #1875852 を参照してください。
内部 DNS レコードの逆引き DNS	いいえ	はい	はい	はい	https://bugzilla.redhat.com/show_bug.cgi?id=2211426 を参照してください。
分離ネットワークの内部 DNS 解決	いいえ	いいえ	はい	はい	OVN は、DNS サービスにポートを割り当てないため、分離ネットワークの内部 DNS 解決をサポートしません。OVS は dnsmasq を使用するため、これは OVS デプロイメントには影響しません。 https://issues.redhat.com/browse/OSP-25661 を参照してください。

機能	OVN RHOS P 16.2	OVN RHOS P 17.1	OVS RHOS P 16.2	OVS RHOS P 17.1	Additional information
セキュリティーグループのロギング	テクノロジープレビュー	はい	いいえ	いいえ	RHOSP は、OVS メカニズムドライバーを使用したセキュリティーグループのロギングをサポートしません。
ステートレスセキュリティーグループ	いいえ	はい	いいえ	いいえ	セキュリティーグループの設定 を参照してください。
負荷分散サービスの分散仮想ルーティング (DVR)	はい	はい	いいえ	いいえ	DVR が有効になっている場合でも、OVS メカニズムドライバーは、負荷分散サービストラフィックをコントローラーノードまたはネットワークノード経由でルーティングします。OVN メカニズムドライバーは、負荷分散サービストラフィックをコンピュータード経由で直接ルーティングします。
IPv6 DVR	はい	はい	いいえ	いいえ	OVS メカニズムドライバーを使用すると、DVR が有効になっている場合でも、RHOSP は IPv6 トラフィックをコンピュータードに分散しません。すべての ingress/egress トラフィックは、集中型のコントローラーノードまたはネットワークノードを通過します。IPv6 DVR が必要な場合は、OVN メカニズムドライバーを使用します。
DVR およびレイヤー 3 高可用性 (L3 HA)	はい	はい	いいえ	いいえ	OVS メカニズムドライバーを使用した RHOSP デプロイメントは、DVR と L3 HA の組み合わせをサポートしません。RHOSP director で DVR を使用する場合、L3 HA は無効になります。これは、Networking サービスが引き続きネットワークノード上のルーターをスケジュールし、L3 エージェント間で負荷分散することを意味します。ただし、1つのエージェントに障害が発生すると、このエージェントがホストするすべてのルーターにも障害が発生します。この影響を受けるのは SNAT トラフィックだけです。Red Hat は、このような場合に allow_automatic_l3agent_failover 機能を使用することを推奨しています。そうすることで、1つのネットワークノードに障害が発生してもルーターが別のノードに再スケジュールされます。

1.2. ML2/OVS から ML2/OVN へのインプレースマイグレーション: 検証済みのシナリオおよび禁止されるシナリオ

Red Hat では、インプレースマイグレーションのシナリオを継続的にテストおよび改良しています。Red Hat Technical Account Manager または Global Professional Services と連携して、OVS デプロイメントが有効なインプレースマイグレーションのシナリオの条件を満たしているかどうかを判断します。

1.2.1. 検証済みの ML2/OVS から ML2/OVN への移行シナリオ

Red Hat は以下の移行パスをテストしました。

- 分散仮想ルーター (DVR) から DVR へ
- 集中ルーティング (DVR なし) から DVR なしへ
- DVR なしから DVR へ

成功したテストには、次のポート設定のワークロードが含まれていました。

- 標準ポート
- SR-IOV ポート
- トランクポート

成功したテストには、iptables_hybrid および Open vSwitch ファイアウォールドライバーも含まれていました。



注記

各テストでは、移行前の環境が greenfield ML2/OVS デプロイメントとして作成されました。

1.2.2. 検証されていない ML2/OVS から ML2/OVN へのインプレースマイグレーションのシナリオ

Red Hat から根本の問題が解決されたと通知があるまで、以下のシナリオでは ML2/OVS から ML2/OVN へのインプレースマイグレーションを行うことはできません。

- GRE ネットワークを使用した OVS 移行前のデプロイメント
- VXLAN を使用する OVS から VXLAN を使用する OVN へ移行
- VLAN プロジェクトネットワークと DVR を備えた OVS 移行前のネットワーク
- SR-IOV および DVR を使用する OVS の OVN への移行前デプロイメント

1.2.3. ML2/OVS から ML2/OVN へのインプレースマイグレーションおよびセキュリティーグループルール

元の ML2/OVS デプロイメントのカスタムセキュリティーグループルールがターゲットの ML2/OVN デプロイメントと互換性があることを確認します。

第2章 ML2 メカニズムドライバーの OVS から OVN への移行

2.1. OVN メカニズムドライバーを移行するための環境の準備

環境の評価と準備は、移行を成功させるために重要です。Red Hat Technical Account Manager または Global Professional Services は、以下の手順での実施を案内します。

注記

非 SR-IOV 環境における `tripleo_controller_group_name` のデフォルト値は **Controller** です。SR-IOV 環境では、`tripleo_controller_group_name` の値は **ControllerSriov** である可能性があります。オプションの移行前のバックアップ手順で使用される Ansible Playbook には **Controller** の値が必要です。値が **Controller Sriov** またはその他の値の場合には、バックアップ操作が失敗する可能性があります。

オプションのバックアップステップを実行する予定がある場合は、ファイル `/usr/share/ansible/roles/backup_and_restore/defaults/main.yml` および `/usr/share/ansible/neutron-ovn-migration/playbooks/roles/recovery-backup/defaults/main.yml` を確認してください。**Controller** 以外の `tripleo_controller_group_name` が含まれている場合は、Red Hat サポートにお問い合わせください。

この問題の修正ステータスを追跡するには、https://bugzilla.redhat.com/show_bug.cgi?id=2158396 を参照してください。

前提条件

- デプロイメントは、最新の RHOSP 17.1 バージョンを使用している。つまり、OpenStack バージョンのアップグレードまたは更新が必要な場合は、まずアップグレードまたは更新を実行し、続いて ML2/OVS から ML2/OVN への移行を実施します。
- 各サブネットプールで少なくとも1つの IP アドレスが使用可能である。OVN メカニズムドライバーは、サブネットごとにメタデータポートを作成します。各メタデータポートは、IP アドレスプールから IP アドレスを要求します。
- Red Hat Technical Account Manager または Global Professional Services と連携して移行を計画し、プロアクティブケースを作成している。[How to submit a Proactive Case](#) を参照してください。
- ML2/OVS デプロイメントで VXLAN プロジェクトネットワークを使用する場合は、「[VXLAN OVS デプロイメントから移行するための MTU 削減](#)」のセクションで説明されているような調整が必要になる場合があるため、このセクションを参照してください。

手順

1. ML2/OVN ステージデプロイメントを作成し、ターゲット ML2/OVN デプロイメントのベースライン設定を取得し、ターゲットデプロイメントの実現可能性をテストします。計画された移行後の実稼働環境要デプロイメントと同じ基本的なロール、ルーティング、およびトポロジーを使用して、ステージデプロイメントを設計します。完全な **openstack overcloud deploy** コマンドをすべてのデプロイメント引数とともに **overcloud-deploy.sh** というファイルに保存します。環境ファイルなど、**openstack overcloud deploy** コマンドが参照するファイルもすべて保存します。この手順の後半で、移行のターゲット ML2/OVN 環境を設定するためにこれらのファイルが必要になります。



注記

これらのファイルは、ステージデプロイメントの作成および移行でのみ使用してください。移行後に再利用しないでください。

2. **openstack-neutron-ovn-migration-tool** をインストールします。

```
sudo dnf install openstack-neutron-ovn-migration-tool
```

3. ステップ 1 で作成した `overcloud-deploy.sh` スクリプトをコピーし、コピーの名前を `overcloud-migrate-ovn.sh` に変更します。 `overcloud-migrate-ovn.sh` 内の `overcloud deploy` コマンドのすべてのパスがまだ正しいことを確認します。後続の手順で、`overcloud-migrate-ovn.sh` スクリプトの引数をカスタマイズします。
4. 以下の一覧で移行シナリオを確認し、**overcloud-migrate-ovn.sh** の **openstack deploy** コマンドをカスタマイズするための適切な手順を実施します。



警告

デプロイメントコマンドでは、環境ファイルを追加する `-e` 引数の順序に注意してください。一般的なデフォルト設定を含む環境ファイル (`neutron-ovn-dvr-ha.yaml` など) は、ブリッジマッピングなどのカスタムネットワーク環境設定を含むファイルを指定する `-e` の引数の前に配置する必要があります。

シナリオ 1: DVR から DVR への移行で、コンピューターノードが外部ネットワークに接続されている場合

- **overcloud-migrate-ovn.sh** で、カスタム heat テンプレートファイル引数を **openstack overcloud deploy** コマンドに追加します。これらの引数はコアテンプレートファイルの引数の後に追加します。

次のコマンド例では、デフォルトの **neutron-ovn-dvr-ha.yaml** heat テンプレートファイルを使用します。実際のデプロイメントでは、OVN 環境を定義するために複数の heat ファイルを使用する場合があります。それぞれを別々の **-e** 引数を使用して追加してください。

```
openstack overcloud deploy \  
--templates /usr/share/openstack-tripleo-heat-templates \  
...  
-e /usr/share/openstack-tripleo-heat-templates/environments/services/ \  
neutron-ovn-dvr-ha.yaml
```

シナリオ 2: 集中ルーティングから集中ルーティングへ (DVR なし)

- デプロイメントで SR-IOV およびその他の NFV 機能を使用する場合は、**overcloud-migrate-ovn.sh** で **-e** 引数を使用して SR-IOV 環境パラメーターを **openstack overcloud deploy** コマンドに追加します。SR-IOV 環境ファイルは、コアテンプレート環境ファイル引数およびその他のカスタム環境ファイル引数の後に追加します。SR-IOV 環境ファイルの例については、`/usr/share/openstack-tripleo-heat-templates/environments/services/neutron-ovn-sriov.yaml` を参照してください。

- カスタムのネットワーク変更は、移行前と同じままにします。

シナリオ 3: 集中ルーティングから DVR への移行で、コンピューティングノードが **br-ex** 経由で外部ネットワークに接続されている場合

- コンピュートノードが **br-ex** ブリッジを介して外部ネットワークに接続されていることを確認します。たとえば、**compute-dvr.yaml** などの環境ファイルで、次のパラメーターを設定します。次に、**-e** を使用して、スクリプト **overcloud-migrate-ovn.sh** の **openstack overcloud deploy** コマンドに環境ファイルを追加します。

```

type: ovs_bridge
  # Defaults to br-ex, anything else requires specific # bridge mapping entries for it to
  be used.
  name: bridge_name
  use_dhcp: false
  members:
    -
      type: interface
      name: nic3
      # force the MAC address of the bridge to this interface
      primary: true

```

5. **overcloud-migrate-ovn.sh** の **overcloud deploy** コマンドの最後に次の引数を追加します。

```

-e /usr/share/openstack-tripleo-heat-templates/environments/disable-container-manage-
clean-orphans.yaml \
-e $HOME/ovn-extras.yaml

```

6. **router** が環境ファイルまたはテンプレートの **NeutronServicePlugins** または **NeutronPluginExtensions** の値として表示される場合は、その値 (**router**) を **ovn-router** に置き換えます。たとえば、**tripleo-heat-templates/environments/services/neutron-ovn-dvr-ha.yaml** では次のようになります。

```

parameter_defaults:
  NeutronServicePlugins: "ovn-router,trunk,qos,placement"

```

7. すべてのユーザーに **overcloud-migrate-ovn.sh** ファイルの実行権限があることを確認します。このスクリプトには、移行プロセス中に実行権限が必要です。

```

$ chmod a+x ~/overcloud-migrate-ovn.sh

```

8. **export** コマンドを使用して、以下の移行関連の環境変数を設定します。以下に例を示します。

```

$ export OVERCLOUDRC_FILE=~/myovercloudrc

```

STACKRC_FILE

アンダークラウドの stackrc ファイル。
デフォルト: ~/stackrc

OVERCLOUDRC_FILE

アンダークラウドの overcloudrc ファイル
デフォルト: ~/overcloudrc

OVERCLOUD_OVN_DEPLOY_SCRIPT

デプロイメントスクリプト。
デフォルト: ~/overcloud-migrate-ovn.sh

DHCP_RENEWAL_TIME

DHCP エージェント設定ファイルで設定する DHCP 更新時間 (秒単位)。
デフォルト: 30

BACKUP_MIGRATION_IP

バックアップが保存されているサーバーの IP アドレス。
デフォルト: 192.168.24.1

BACKUP_MIGRATION_CTL_PLANE_CIDRS

バックアップされるすべてのノードのコントロールプレーンサブネットを CIDR 表記でコマンド区切りで表した文字列。
デフォルト: 192.168.24.0/24

/usr/bin/ovn_migraton.sh ファイルの先頭に、関連するすべての環境変数のリストが表示されます。

9. ovn-migration ディレクトリーにいることを確認して、**ovn_migration.sh generate-inventory** コマンドを実行し、インベントリーファイル **hosts_for_migration** および **ansible.cfg** ファイルを生成します。

```
$ ovn_migration.sh generate-inventory | sudo tee -a /var/log/ovn_migration_output.txt
```

10. **hosts_for_migration** ファイルで正確性を確認してください。
 - a. リストがお使いの環境と一致していることを確認します。
 - b. 各ノードに ovn コントローラーがあることを確認します。
 - c. リスト見出し ([ovn-controllers] など) がリスト項目に含まれていないことを確認します。
 - d. ovn 移行ディレクトリーから、**ansible -i hosts_for_migration -m ping all** コマンドをすべて ping します。
11. (オプション) 移行中に予期しない事態が発生した場合に備えて、移行を元に戻す準備をするためにデプロイメントをバックアップします。

- a. setup_rear_extra_vars.yaml ファイルに次の行が存在する場合は削除します。

```
USER_INPUT_TIMEOUT: 5
KERNEL_CMDLINE: unattended
ISO_RECOVER_MODE: unattended
```

- b. コントロールプレーンのバックアップ選択したバックアップメカニズムを使用して、コントローラーノードをバックアップします。サポートされている選択肢は、ReaR バックアップツールを使用するデフォルトの **ovn-migration.sh backup** コマンドです。

```
$ ovn_migration.sh backup
```


- c. オーバークラウドのデプロイに使用されたテンプレートと環境ファイルをバックアップします。**ovn-migration.sh backup** コマンドはオーバークラウドをバックアップしません。移行が部分的にまたは失敗した後にコントローラーノードを元に戻す必要がある場合は、OVS オーバークラウドを復元するためにこのバックアップが必要になります。
- d. 元の RHOSP 16.2 ML2/OVS デプロイメントをデプロイするのに使用したスクリプトをコピーします。たとえば、元のスクリプトの名前は **overcloud_deploy.sh** になります。コピーに **overcloud-revert-ovs.sh** という名前を付けます。
- e. 次の内容を含む **/home/stack/ovs-extra.yml** ファイルを作成します。

```
parameter_defaults:
  ForceNeutronDriverUpdate: true
```

- f. **overcloud-revert-ovs.sh** の最終的な環境ファイル引数が次のようになっていることを確認します。

```
-e /home/stack/ovs-extra.yml
```

- g. **overcloud-revert-ovs.sh** を安全に保存します。失敗した移行を元に戻す場合に必要になります。

12. 「[OVS から OVN への ML2 メカニズムドライバーの移行用コンテナイメージの準備](#)」に進みます。

2.2. OVS から OVN への ML2 メカニズムドライバーの移行用コンテナイメージの準備

環境の評価と準備は、移行を成功させるために重要です。Red Hat Technical Account Manager または Global Professional Services は、以下の手順での実施を案内します。

前提条件

- [OVS から OVN への ML2 メカニズムドライバーの移行のための環境の準備](#) の手順が完了している。

手順

1. ML2/OVN への移行後に使用する新しいコンテナイメージを準備します。
 - a. ホームディレクトリーに **containers-prepare-parameter.yaml** ファイルがない場合は作成します。

```
$ test -f $HOME/containers-prepare-parameter.yaml || sudo openstack tripleo container image prepare default \
--output-env-file $HOME/containers-prepare-parameter.yaml
```
 - b. **containers-prepare-parameter.yaml** が **\$HOME/overcloud-migrate-ovn.sh** ファイルおよび **\$HOME/overcloud-deploy.sh** ファイルの末尾にあることを確認します。
 - c. **containers-prepare-parameter.yaml** ファイルの **neutron_driver** を **ovn** に変更します。

```
$ sed -i -E 's/neutron_driver:([ ]w+)/neutron_driver: ovn/' $HOME/containers-prepare-parameter.yaml
```

d. neutron_driver への変更を確認します。

```
$ grep neutron_driver $HOME/containers-prepare-parameter.yaml
neutron_driver: ovn
```

e. イメージを更新します。

```
$ sudo openstack tripleo container image prepare \
--environment-file /home/stack/containers-prepare-parameter.yaml
```



注記

containers-prepare-parameter.yaml ファイルへの完全パスを指定します。それ以外の場合、コマンドはイメージリストを更新したりエラーメッセージを表示したりすることなく、非常に迅速に完了します。

2. アンダークラウドにおいて、更新されたイメージを検証します。

```
. Log in to the undercloud as the user `stack` and source the stackrc file.
$ source ~/stackrc
$ openstack tripleo container image list | grep `'-ovn'
```

リストは以下の例のようになります。これには、OVN データベース、OVN コントローラー、メタデータエージェント、および neutron サーバーエージェント用のコンテナが含まれます。

```
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp17-openstack-ovn-
northd:17.1_20240725.1
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp17-openstack-ovn-sb-db-
server:17.1_20240725.1
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp17-openstack-ovn-
controller:17.1_20240725.1
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp17-openstack-neutron-server-
ovn:17.1_20240725.1
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp17-openstack-ovn-nb-db-
server:17.1_20240725.1
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp17-openstack-neutron-
metadata-agent-ovn:17.1_20240725.1
```

RHOSP 16.2 の OVN メカニズムドライバーに移行する場合、リストは次のようなものです。

```
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp16-openstack-ovn-
northd:16.2_20211110.2
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp16-openstack-ovn-sb-db-
server:16.2_20211110.2
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp16-openstack-ovn-
controller:16.2_20211110.2
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp16-openstack-neutron-server-
ovn:16.2_20211110.2
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp16-openstack-ovn-nb-db-
server:16.2_20211110.2
docker://undercloud-0.ctlplane.redhat.local:8787/rh-osbs/rhosp16-openstack-neutron-
metadata-agent-ovn:16.2_20211110.2
```

- 元のデプロイメントで VXLAN を使用している場合は、最大伝送単位 (MTU) 値の調整が必要になる場合があります。[「VXLAN OVS デプロイメントから移行するための MTU 削減」](#)に進みます。
元のデプロイメントで VLAN ネットワークを使用している場合は、MTU 調整をスキップして [「ML2 メカニズムドライバーの OVS から OVN への移行」](#) の手順に進むことができます。

2.3. VXLAN OVS デプロイメントから移行するための MTU 削減

移行前の OVS デプロイメントで VXLAN トンネリングプロトコルを使用している場合は、Geneve トンネリングプロトコルを使用する OVN に移行する前に、ネットワーク最大伝送単位 (MTU) を 8 バイト削減する必要がある場合があります。



注記

移行前の専用のメンテナンス期間にこの手順を実行することを検討してください。

VXLAN パケットは、ヘッダーコンテンツ用に 50 バイトのデータを予約します。これには、42 バイトの標準外部ヘッダーと 8 バイトの VXLAN ヘッダーが含まれます。物理ネットワークが 1500 バイトの標準イーサネット MTU を使用している場合、VXLAN ネットワークの MTU を 1450 に設定すると、トラフィックは断片化せずに通過できます。

Geneve パケットは、ヘッダーコンテンツ用に 58 バイトのデータを予約します。これには、42 バイトの標準外部ヘッダーと 16 バイトの Geneve ヘッダーが含まれます。したがって、物理ネットワークの MTU が 1508 未満の場合は、断片化の必要性を避けるために、移行前の OpenStack VXLAN ネットワーク上の MTU を 8 バイト削減する必要があります。



注記

物理ネットワークが断片化せずに OpenStack VXLAN ネットワーク MTU より少なくとも 58 バイト多く送信できる場合は、この手順をスキップして [「ML2 メカニズムドライバーの OVS から OVN への移行」](#) の手順に進んでください。たとえば、物理ネットワークが 9000 バイトのジャンボフレーム用に設定されており、オープンスタックネットワークの MTU が 8942 以下の場合、この手順を省略できます。

RHOSP OVN 移行ツールは、VXLAN および GRE オーバークラウドネットワーク上の MTU を自動的に 8 バイト下げます。次の手順では、ツールを使用して次のことを行います。

- DHCP T1 タイマーを 30 秒に短縮して、DHCP 更新の頻度を増やします。
- 既存の VXLAN ネットワーク上の MTU サイズを 8 バイト削減します。

デプロイですべての VM インスタンスの設定に DHCP を使用しない場合は、除外されたインスタンスの MTU を手動で減らす必要があります。

前提条件

- [「OVN メカニズムドライバーを移行するための環境の準備」](#) の手順を完了している。
- [「OVS から OVN への ML2 メカニズムドライバーの移行用コンテナイメージの準備」](#) の手順を完了している。
- 移行前のデプロイメントは、VXLAN を備えた Red Hat OpenStack Platform (RHOSP) 17.1 以降である。

手順

1. `ovn_migration.sh reduce-dhcp-t1` を実行します。これにより、DHCP エージェントが実行しているすべてのノードで、`/var/lib/config-data/puppet-generated/neutron/etc/neutron/dhcp_agent.ini` 内の `dhcp_renewal_time` を設定する内部 neutron DHCP サーバーの T1 パラメーターが短くなります。

```
$ ovn_migration.sh reduce-dhcp-t1 | sudo tee -a /var/log/ovn_migration_output.txt
```

2. T1 パラメーターが既存の仮想マシンに伝播されていることを確認します。プロセスに最大 4 時間かかる場合があります。

- コンピュートノードのいずれかにログインします。
- プロジェクトネットワークに接続された仮想マシンタップのいずれかで `tcpdump` を実行します。
T1 の伝播が成功すると、リクエストは約 30 秒ごとに発生することが予想されます。

```
[heat-admin@overcloud-novacompute-0 ~]$ sudo tcpdump -i tap52e872c2-e6 port 67 or port 68 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on tap52e872c2-e6, link-type EN10MB (Ethernet), capture size 262144 bytes
13:17:28.954675 IP 192.168.99.5.bootpc > 192.168.99.3.bootps: BOOTP/DHCP, Request from fa:16:3e:6b:41:3d, length 300
13:17:28.961321 IP 192.168.99.3.bootps > 192.168.99.5.bootpc: BOOTP/DHCP, Reply, length 355
13:17:56.241156 IP 192.168.99.5.bootpc > 192.168.99.3.bootps: BOOTP/DHCP, Request from fa:16:3e:6b:41:3d, length 300
13:17:56.249899 IP 192.168.99.3.bootps > 192.168.99.5.bootpc: BOOTP/DHCP, Reply, length 355
```



注記

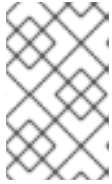
cirros 仮想マシンでは、この検証はできません。cirros `udhcpd` 実装は DHCP オプション 58 (T1) に応答しません。完全な Linux 仮想マシンに属するポートで、この検証を試みます。Red Hat では、Windows や Linux ディストリビューションのバリエーションなど、ワークロードに含まれるさまざまなオペレーティングシステムをすべて確認することを推奨します。

3. DHCP の T1 パラメーターへの変更を反映するように仮想マシンインスタンスが更新されていない場合は、再起動します。
4. 移行前の VXLAN ネットワークの MTU を減らします。

```
$ ovn_migration.sh reduce-mtu | sudo tee -a /var/log/ovn_migration_output.txt
```

このステップでは、ネットワークごとに MTU を減らし、`adapted_mtu` で完了したネットワークにタグを付けます。このツールは、VXLAN ネットワークでのみ動作します。デプロイメントに VLAN プロジェクトネットワークしかない場合、この手順では値は変更されません。

5. VXLAN プロジェクトネットワーク上に静的 IP が割り当てられているインスタンスがある場合は、インスタンスの MTU を手動で 8 バイト減らします。たとえば、VXLAN ベースの MTU が 1450 の場合は、これを 1442 に変更します。



注記

このステップは、VXLAN プロジェクトネットワークに静的 IP の割り当てと MTU 設定を手動で提供している場合にのみ行います。デフォルトでは、DHCP が IP 割り当てと MTU 設定を提供します。

6. 「ML2 メカニズムドライバーの OVS から OVN への移行」に進みます。

2.4. ML2 メカニズムドライバーの OVS から OVN への移行

`ovn-migration` スクリプトは、OVS から OVN への ML2 メカニズムドライバーのインプレース移行に関連する環境セットアップ、移行、およびクリーンアップタスクを実行します。

前提条件

- [OVS から OVN への ML2 メカニズムドライバーの移行のための環境の準備](#) の手順が完了している。
- 元のデプロイメントで VXLAN または GRE を使用している場合は、[OVS メカニズムドライバーから OVN メカニズムドライバーへの移行のための MTU の調整](#) の手順も完了している。
- また、[OVS メカニズムドライバーから OVN メカニズムドライバーへの移行のためのコンテナイメージの準備](#) を通じて、必要な移行手順をすべて完了している。

手順

1. 新しいネットワーク、サブネット、ルーターまたはインスタンスの作成、コンピュータード間のインスタンスの移行など、Networking Service (neutron) API と対話するすべての操作を停止します。
移行中の Networking API とのやり取りにより、未定義の動作が発生する可能性があります。移行が完了したら、API 操作を再開できます。
2. `ovn_migration.sh start-migration` を実行して移行プロセスを開始します。`tee` コマンドは、トラブルシューティング目的のスクリプト出力のコピーを作成します。

```
$ ovn_migration.sh start-migration | sudo tee -a /var/log/ovn_migration_output.txt
```

Result

スクリプトは以下のアクションを実行します。

- `br-int` ではなく、一時ブリッジ `br-migration` を使用して、参照実装サービスと共に OVN をデプロイするために、オーバークラウドスタックを更新します。一時ブリッジは、移行時のダウンタイムを抑えるのに役立ちます。
- `neutron-ovn-db-sync-util` を実行して OVN ノースバウンドデータベースを生成します。このユーティリティーは、OVN ノースバウンドデータベースで等価なリソースを作成するために Neutron データベースを調べます。
- `br-migration` ではなく `br-int` に `ovn-controller` を再度割り当てます。
- 以下に示す ML2/OVN が使用していないノードリソースを削除します。
 - ネットワーク名前空間 (`fip`、`snat`、`qrouter`、`qdhcp`) をクリーンアップします。

- **br-int** の不要なパッチポートを削除します。
- **br-tun** および **br-migration** ovs ブリッジを削除します。
- **br-int** から **qr-**、**ha-**、 および **qg-** で始まるポートを削除します (neutron-netns-cleanup を使用)。
- Networking サービス API を使用して、Networking サービス (neutron) エージェントおよび Networking サービス HA の内部ネットワークをデータベースから削除します。

第3章 ML2/OVS から ML2/OVN への移行を元に戻す

OVN メカニズムドライバーへの移行の準備で説明されているとおり、[ovn-migration.sh](#) のバックアップを使用してコントローラーノードをバックアップし、ML2/OVS から ML2/OVN への移行を開始する前にオーバークラウドテンプレートをバックアップした場合は、これらの基本的な手順を実行して移行を元に戻すことができます。

1. コントローラーノードを復元します。
2. `revert.yaml` を実行して、ML2/OVN アーティファクトを削除します。
3. バックアップした ML2/OVS テンプレートを使用してオーバークラウドを再デプロイします。



注記

移行を元に戻してコンピューターノードを復元すると、20 分以上の大幅なネットワークダウンタイムが発生する可能性があります。



警告

ML2/OVS から ML2/OVN に移行すると、環境が変更され、完全には元に戻せなくなる可能性があります。

適切なバックアップ手順と元に戻す手順に従えば、失敗した移行や中断した移行を元に戻すことができますが、元に戻された OVS 環境は元の環境から変更されている可能性があります。たとえば、OVN メカニズムドライバーに移行し、インスタンスを別のコンピューターノードに移行してから、OVN 移行を元に戻すと、インスタンスは元のコンピューターノード上に置かれます。また、元に戻す操作により、データプレーンへの接続が中断されます。

実稼働環境で移行を行う前に、プロアクティブケースを作成してください。次に、Red Hat Technical Account Manager または Red Hat Global Professional Services と連携してバックアップおよび移行プランを作成し、実稼働環境に極めて近いステージ環境で移行をテストします。

潜在的な移行の元に戻しに備えてバックアップを準備することを選択した場合は、ステージ環境で移行の元に戻しをテストする必要があります。

移行の元に戻しは逆の移行ではありません。これは、移行が失敗した場合の最後の手段としてのみ使用することを目的としています。移行が検証に合格し、移行後の環境で運用上の問題が判明した場合は、移行後の環境でそれらの問題をバグとして対処します。

3.1. ML2/OVN 移行を元に戻すためにコントローラーノードの復元

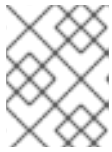
移行前に `ovn-migration.sh backup` を使用してコントローラーノードをバックアップした場合は、移行後の環境で障害が発生した場合に、Relax and Recover ツール (ReaR) を使用してそれらを復元できます。

前提条件

- 失敗した移行についてサポートケースを提出し、Red Hat から指示を受けている。
- 移行前に **ovn-migration.sh backup** を使用してコントローラーノードをバックアップしている。
- バックアップノードにアクセスできる。
- ML2/OVS オーバークラウドを再デプロイするために必要なオーバークラウドテンプレートをバックアップしている。

手順

1. 新しいネットワーク、サブネット、ルーターの作成や、コンピュータード間での仮想マシンインスタンスの移行など、コントロールプレーン API と対話するすべての操作を停止します。
2. 各コントロールプレーンノードの電源をオフにします。次のステップに進む前に、コントロールプレーンノードの電源が完全にオフになっていることを確認します。
3. 対応するバックアップの ISO イメージで各コントロールプレーンノードをブートします。
4. **Relax-and-Recover** ブートメニューが表示されたら、各コントロールプレーンノードで **Recover <control_plane_node>** を選択します。<control_plane_node> を対応するコントロールプレーンノードの名前に置き換えます。



注記

システムで UEFI を使用している場合は、**Relax-and-Recover (no Secure Boot)** オプションを選択します。

5. それぞれのコントロールプレーンノードで **root** ユーザーとしてログインし、ノードを復元します。以下のメッセージが表示されます。

```
Welcome to Relax-and-Recover. Run "rear recover" to restore your system!
RESCUE <control_plane_node>:~ # rear recover
```

コントロールプレーンノードの復元プロセスが完了すると、コンソールに以下のメッセージが表示されます。

```
Finished recovering your system
Exiting rear recover
Running exit tasks
```

6. ノードの電源を切ります。

```
RESCUE <control_plane_node>:~ # poweroff
```

7. ブートシーケンスを通常のブートデバイスに設定します。ノードをブートすると、以前の状態で再開されます。
8. サービスが正常に実行されていることを確認するには、pacemaker のステータスを確認します。**root** ユーザーとしてコントローラーノードにログインし、以下のコマンドを入力します。

```
# pcs status
```


9. Ansible を使用して revert.yml ファイルを実行します。

```
ansible-playbook -vv /usr/share/ansible/\
neutron-ovn-migration/playbooks/revert.yml \
-i hosts_for_migration
```

10. **ovn-router** が環境ファイルまたはテンプレートの **NeutronServicePlugins** または **NeutronPluginExtensions** の値として表示される場合は、その値 (**ovn-router**) を **router** に置き換えます。パラメーターは複数のファイルに表示される場合があります。たとえば、そのようなファイルの1つは **tripleo-heat templates/environments/services/neutron-ovn-dvr-ha.yaml** です。

```
parameter_defaults:
  NeutronServicePlugins: "router,trunk,qos,placement"
```

11. OVS テンプレートを使用するようにオーバークラウドを更新します。

```
bash ~/overcloud-revert-ovs.sh
```

12. 元の ML2/OVS デプロイメントにトランクポートを使用するインスタンスが含まれていた場合は、次のコマンドを実行してそれらのインスタンスへの接続を復元します。

```
ansible-playbook -vv /usr/share/ansible/neutron-ovn-migration/playbooks/revert-rewire.yml \
-i hosts_for_migration
```

トラブルシューティング

- **pcs status** で表示されるリソースアラームを以下のコマンドで解除します。

```
# pcs resource clean
```

- **pcs status** で表示される STONITH フェンシングの動作エラーを以下のコマンドで解除します。

```
# pcs resource clean
# pcs stonith history cleanup
```