



## OpenStack Platform 10 Director を使用した Red Hat OpenStack 向け Cisco ACI インストール ガイド

初版：2017 年 9 月 15 日

最終更新：2018 年 12 月 13 日

### シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター

0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>

【注意】 シスコ製品をご使用になる前に、安全上の注意（[www.cisco.com/jp/go/safety\\_warning/](http://www.cisco.com/jp/go/safety_warning/)）をご確認ください。本書は、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com go trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2017–2018 Cisco Systems, Inc. All rights reserved.



## 目次

---

はじめに :

[はじめに](#) v

[対象読者](#) v

[表記法](#) v

[関連資料](#) vii

[マニュアルに関するフィードバック](#) viii

[マニュアルの入手方法およびテクニカルサポート](#) viii

---

第 1 章

[新機能および変更された機能に関する情報](#) 1

[新機能および変更された機能に関する情報](#) 1

---

第 2 章

[OSP Director を使用した Red Hat OpenStack 向け Cisco ACI インストール ガイド](#) 3

[OSP Director を使用した OpenStack での ACI の概要](#) 3

[OSP Director を使用した OpenStack での Cisco ACI の要件および前提条件](#) 4

[関連資料](#) 6

[ML2、GBP または統合モードを使用した OpFlex の導入](#) 6

[OpenStack 向け ACI のインストール準備](#) 6

[APIC とネットワークの設定](#) 6

[オーバークラウドの設定](#) 10

[OpFlex オーケストレーションを備えた ACI の準備](#) 10

[OpFlex オーケストレーションを備えたアンダークラウド ACI の準備](#) 11

[オーバークラウドのインストール](#) 11

[参考リンク](#) 15

[ACI ファブリック初期化の例](#) 15

---

第 3 章	<b>ACI と OSP のアップグレード</b>	<b>19</b>
	アップグレードガイドライン	19
	アップグレード前のガイドライン	19
	agent_ovs の更新	20
	アップグレードのガイドライン	20
	アップグレード後のガイドライン	21
	ACI パッケージのアップグレード	21
	OSP メジャー リリース アップグレードの実行	22
	リリース固有の変更点	23

---

第 4 章	<b>OpenStack 外部ネットワークの追加</b>	<b>25</b>
	OpenStack 外部ネットワークの追加	25

---

付録 A :	<b>参考資料</b>	<b>29</b>
	導入テンプレートの例	29
	controller.yaml	29
	compute.yaml	32
	階層型ポート バインディングの設定	34



## はじめに

---

この前書きは、次の項で構成されています。

- [対象読者](#) (v ページ)
- [表記法](#) (v ページ)
- [関連資料](#) (vii ページ)
- [マニュアルに関するフィードバック](#) (viii ページ)
- [マニュアルの入手方法およびテクニカルサポート](#) (viii ページ)

## 対象読者

このガイドは、次の1つ以上に責任を持つ、専門知識を備えたデータセンター管理者を主な対象にしています。

- 仮想マシンのインストールと管理
- サーバ管理
- スイッチおよびネットワークの管理

## 表記法

コマンドの説明には、次のような表記法が使用されます。

表記法	説明
<b>bold</b>	太字の文字は、表示どおりにユーザが入力するコマンドおよびキーワードです。
<i>italic</i>	イタリック体の文字は、ユーザが値を入力する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角カッコで囲んで示しています。

表記法	説明
[x   y]	いずれか1つを選択できる省略可能なキーワードや引数は、角カッコで囲み、縦棒で区切って示しています。
{x   y}	必ずいずれか1つを選択しなければならない必須キーワードや引数は、波カッコで囲み、縦棒で区切って示しています。
[x {y   z}]	角カッコまたは波カッコが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。角カッコ内の波カッコと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
variable	ユーザが値を入力する変数であることを表します。イタリック体が使用できない場合に使用されます。
string	引用符を付けない一組の文字。string の前後には引用符を使用しません。引用符を使用すると、その引用符も含めて string とみなされます。

例では、次の表記法を使用しています。

表記法	説明
screen フォント	スイッチが表示する端末セッションおよび情報は、screen フォントで示しています。
太字の screen フォント	ユーザが入力しなければならない情報は、太字の screen フォントで示しています。
イタリック体の screen フォント	ユーザが値を指定する引数は、イタリック体の screen フォントで示しています。
<>	パスワードのように出力されない文字は、山カッコ (<>) で囲んで示しています。
[]	システムプロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符 (!) またはポンド記号 (#) がある場合には、コメント行であることを示します。

このマニュアルでは、次の表記法を使用しています。



(注) 「注釈」です。役立つ情報やこのマニュアルに記載されていない参照資料を紹介しています。



**注意** 「要注意」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。



**警告** 安全上の重要な注意事項

「危険」の意味です。人身事故を予防するための注意事項が記述されています。機器の取り扱い作業を行うときは、電気回路の危険性に注意し、一般的な事故防止対策に留意してください。各警告の最後に記載されているステートメント番号を基に、装置に付属の安全についての警告を参照してください。

これらの注意事項を保存しておいてください。

## 関連資料

### Application Policy Infrastructure Controller (APIC) のマニュアル

次のガイドでは、APIC のドキュメントを提供します。

- 『Cisco APIC Getting Started Guide』
- 『Cisco APIC Basic Configuration Guide』
- 『Cisco ACI Fundamentals』
- 『Cisco APIC Layer 2 Networking Configuration Guide』
- 『Cisco APIC Layer 3 Networking Configuration Guide』
- 『Cisco APIC NX-OS Style Command-Line Interface Configuration Guide』
- 『Cisco APIC REST API Configuration Guide』
- 『Cisco APIC Layer 4 to Layer 7 Services Deployment Guide』
- 『Cisco ACI Virtualization Guide』
- 『Cisco Application Centric Infrastructure Best Practices Guide』

これらすべてのドキュメントは、次の URL で入手できます。 <http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>

### シスコ アプリケーション セントリック インフラストラクチャ (ACI) のマニュアル

ACI の各種マニュアルは、次の URL から入手できます。 <http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>

シスコアプリケーションセントリックインフラストラクチャ (ACI) シミュレータのマニュアル

Cisco ACI Simulator のマニュアルは、次の URL から入手できます：<http://www.cisco.com/c/en/us/support/cloud-systems-management/application-centric-infrastructure-simulator/tsd-products-support-series-home.html>

**Cisco Nexus 9000** シリーズ スイッチのマニュアル

Cisco Nexus 9000 シリーズ スイッチのマニュアルは、次の URL で入手できます。<http://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/tsd-products-support-series-home.html>

**Cisco Application Virtual Switch** のマニュアル

Cisco Application Virtual Switch (AVS) のマニュアルは、次の URL で入手できます。<http://www.cisco.com/c/en/us/support/switches/application-virtual-switch/tsd-products-support-series-home.html>

シスコアプリケーションセントリックインフラストラクチャ (ACI) と OpenStack の統合に関するマニュアル

Cisco ACI と OpenStack の統合に関するマニュアルは、次の URL から入手できます。<http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>

## マニュアルに関するフィードバック

このマニュアルに関する技術的なフィードバック、または誤りや記載もれなどお気づきの点がございましたら、[apic-docfeedback@cisco.com](mailto:apic-docfeedback@cisco.com) までご連絡ください。ご協力をよろしくお願いいたします。

## マニュアルの入手方法およびテクニカル サポート

マニュアルの入手方法、Cisco バグ検索ツール (BST) の使用法、テクニカル サポートの依頼方法、および追加情報の収集方法については、『*What's New in Cisco Product Documentation*』 (<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>) を参照してください。

『*What's New in Cisco Product Documentation*』では、シスコの新規および改訂版の技術マニュアルの一覧を、RSS フィードとして購読できます。また、リーダーアプリケーションを使用して、コンテンツをデスクトップに直接配信することもできます。RSS フィードは無料のサービスです。



# 第 1 章

## 新機能および変更された機能に関する情報

この章の内容は、次のとおりです。

- [新機能および変更された機能に関する情報 \(1 ページ\)](#)

## 新機能および変更された機能に関する情報

次の表は、この最新リリースまでのガイドでの主な変更点の概要を示したものです。ただし、このリリースまでのこのガイドの変更点や新機能の中には一部、この表に記載されていないものもあります。

表 1: 新機能と変更された動作

Cisco APIC のリリースバージョン	機能	説明	参照先
2.3(1)	--	OpenStack 外部ネットワークの追加方法が追加されました。	詳細については、「 <a href="#">OpenStack 外部ネットワークの追加 (25 ページ)</a> 」を参照してください。
2.3(1)	--	複数の物理ネットワークのサポートが追加されました。	詳細については、「 <a href="#">オーバークラウドのインストール (11 ページ)</a> 」を参照してください。

Cisco APIC のリリースバージョン	機能	説明	参照先
2.3(1)	--	アップグレード手順が追加されました。	<p>詳細については、以下を参照してください。</p> <ul style="list-style-type: none"> <li>• <a href="#">ACI パッケージのアップグレード (21 ページ)</a></li> <li>• <a href="#">OSP メジャーリリース アップグレードの実行 (22 ページ)</a></li> <li>• <a href="#">リリース固有の変更点 (23 ページ)</a></li> </ul>
2.3(1)	--	ファブリックインターフェイスと VPC の結合に関する情報が追加されました。	<p>詳細については、「<a href="#">OSP Director を使用した OpenStack での Cisco ACI の要件および前提条件 (4 ページ)</a>」を参照してください。</p>
2.3(1)	Newton リリースで導入された tripleo コンポーザブルサービスを使用します。	このガイドがリリースされました。	--



## 第 2 章

# OSP Director を使用した Red Hat OpenStack 向け Cisco ACI インストール ガイド

この章の内容は、次のとおりです。

- [OSP Director を使用した OpenStack での ACI の概要](#) (3 ページ)
- [OSP Director を使用した OpenStack での Cisco ACI の要件および前提条件](#) (4 ページ)
- [ML2、GBP または統合モードを使用した OpFlex の導入](#) (6 ページ)
- [OpenStack 向け ACI のインストール準備](#) (6 ページ)
- [オーバークラウドの設定](#) (10 ページ)
- [オーバークラウドのインストール](#) (11 ページ)
- [ACI ファブリック初期化の例](#) (15 ページ)

## OSP Director を使用した OpenStack での ACI の概要

このマニュアルでは、Newton リリースで導入された tripleo コンポーザブル サービスを使用します。この *OSP Director* を使用した *Red Hat OpenStack* 向け *Cisco ACI* リリース 2.3(x) 以降インストール ガイドは、*OSP Director* を使用した *Red Hat OpenStack* 向け *Cisco ACI* リリース 2.2(x) インストール ガイドに代わるものであり、リリース 2.3 以降に使用する必要があります。コンポーザブル サービスの詳細については、次の URL を使用して OpenStack コンポーザブル サービス チュートリアルを参照してください。

[https://docs.openstack.org/tripleo-docs/latest/install/developer/tht\\_walkthrough/tht\\_walkthrough.html](https://docs.openstack.org/tripleo-docs/latest/install/developer/tht_walkthrough/tht_walkthrough.html)

Cisco Application Centric Infrastructure (ACI) は包括的なポリシー ベースのアーキテクチャであり、コントローラ ベースのインテリジェントなネットワーク スイッチ ファブリックを実現します。このファブリックは、OpenStack を含む複数のオーケストレーション ツール、自動化 ツール、および管理ツールに直接統合可能な API インターフェイスを通じてプログラムすることにより管理する設計となっています。ACI を OpenStack と統合することによって、ネットワーク構造体の動的な作成を OpenStack 要件に従って直接駆動するだけでなく、ACI Application Policy Infrastructure Controller (APIC) 内のさらなる可視性を個別の仮想マシン (VM) インスタンスのレベルに至るまで実現できます。

OpenStack は、クラウドコンピューティング環境を構築するための柔軟なソフトウェアアーキテクチャを明確にします。OpenStack のリファレンスソフトウェアベースの実装では、VLAN、GRE、VXLAN を含む複数のレイヤ 2 トランスポートが考慮されています。また、OpenStack の Neutron プロジェクトはソフトウェアベースのレイヤ 3 転送も提供できます。ACI とともに利用すると、ACI ファブリックは統合されたレイヤ 2 およびレイヤ 3 の VXLAN ベースのオーバーレイ ネットワーキング機能を提供して、ネットワーク カプセル化処理をコンピューティング ノードからトップオブブラック、または ACI リーフ スイッチに移行します。このアーキテクチャは、ソフトウェア オーバーレイ ネットワーキングの柔軟性を提供するとともに、ハードウェアベースのネットワークのパフォーマンス上および動作上の利点も提供します。

Cisco ACI OpenStack プラグインは、ML2 モードまたは GBP モードで導入できます。モジュラレイヤ 2 (ML2) モードでは、ネットワークの作成に標準 Neutron API が使用されます。これは OpenStack に VM およびサービスを導入するための従来の方法です。グループ ベース ポリシー (GBP) モードでは、アプリケーションをポリシーグループとして説明、作成、展開する新しい API が提供されます。これにより、ネットワーク固有の詳細を気にする必要がなくなります。詳細については、次の URL にある『*OpenStack Group-Based Policy User Guide*』を参照してください。

[http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/openstack/b\\_OpenStack\\_Group-Based\\_Policy\\_User\\_Guide.html](http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/openstack/b_OpenStack_Group-Based_Policy_User_Guide.html)

以前の OpFlex プラグインバージョン (クラシカルモードと呼ばれる) では、導入時にプラグインのモード (Neutron/ML2 または GBP) を決定する必要があり、GBP API と Neutron/ML2 API の両方を同時に使用することはできませんでした。OpFlex プラグインバージョン 2.2.1 以降では、「統合」モードでプラグインを導入できます。統合モードでは、Neutron API または GBP API のいずれかを使用してアプリケーション トポロジを作成することが可能です。統合プラグインモードの場合は、OpenStack リリース Mitaka 以降と ACI リリース 2.2(1) 以降も必要です。

このガイドでは、統合インストールモードでの OpFlex プラグインの導入について説明します。



- (注) 統合モードで GBP グループを作成する際に (auto-ptg) グループも表示されますが、これらのグループは内部的に使用するためのものであり、ユーザインタラクション (VM の接続、メンバーの追加) はサポートされていません。

## OSP Director を使用した OpenStack での Cisco ACI の要件 および前提条件

- 対象読者 : Linux、Red Hat OpenStack ディストリビューション、ACI ポリシーモデル、GUI ベースの APIC 設定に関する実際的な知識があること。加えて OpenStack アーキテクチャと導入に精通していること。

- ACI ファブリック : ACI ファブリックがインストール済みであり、1.3 以上のバージョンを使用して初期化されていること。新規 ACI ファブリックの初期化に関する基本ガイドラインについては、[ACI ファブリック初期化の例 \(15 ページ\)](#) を参照してください。



---

(注) 複数のリーフペア間の通信については、OpenStack 外部ネットワークを使用できるように、ファブリックで BGP ルート リフレクタが有効化されている必要があります。

---

- vPC にボンディングされたファブリック インターフェイスを使用する場合、ファブリック インターフェイス向けの ovs\_bond の追加はサポートされません。これは ovs ブリッジに 1 つのインターフェイスとして追加する必要があるためです。type を linux\_bond に設定してファブリック インターフェイスを集約する必要があります。nic-config テンプレートでファブリック インターフェイスを作成する簡単な例を示します。

```
type: ovs_bridge
  name: {get_input: bridge_name}
  mtu: 1500
  members:
    -
      type: linux_bond
      name: bond1
      ovs_options: {get_param: BondInterfaceOvsOptions}
      mtu: 1600
      members:
        -
          type: interface
          name: nic1
          primary: true
          mtu: 1600
        -
          type: interface
          name: nic2
          mtu: 1600
```

- ボンディングを使用する際にサポートされるのは 802.3ad のみです。
- UCS-B シリーズで導入する場合は、冗長性を実現するために、ボンディングによるデュアル vNIC のみがファブリック インターフェイスでサポートされます。



---

(注) ハードウェア フェールオーバーで単一の vNIC を使用しないでください。

---

- UCS-B シリーズでの導入では、VLAN カプセル化のみがサポートされます。
- APIC GUI で、ファブリックの OpFlex 認証を無効にします。[System] > [System Settings] > [Fabric Wide Setting] > [Fabric Wid Setting Policy] ペインで、[To enforce Opflex client certificate authentication for GOLF and Linux] がオフになっていることを確認します。

## 関連資料

詳細については、次の URL にある『*Director Installation and Usage Red Hat OpenStack Platform 10*』ドキュメントを参照してください。

[https://access.redhat.com/documentation/en-us/red\\_hat\\_openstack\\_platform/10/html/director\\_installation\\_and\\_usage/](https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installation_and_usage/)

## ML2、GBP または統合モードを使用した OpFlex の導入

ここでは、Red Hat OpenStack ディストリビューションでの ACI OpenStack プラグインのインストールと設定の方法について説明します。

次のサンプル手順は、Red Hat OpenStack の OpenStack Platform 12 リリースで検証済みです。OpenStack システムは、インストール方法によって大きく異なる可能性があります。したがって、ここで示した例は、特定のインストール状態に適応するための基本として使用してください。

Red Hat OpenStack Platform Director のインストール マニュアルに従って、OpenStack Platform Director を用意し、適切な導入ファイルとリソース ファイルを作成してください。

詳細については、[関連資料 \(6 ページ\)](#) を参照してください。

## OpenStack 向け ACI のインストール準備

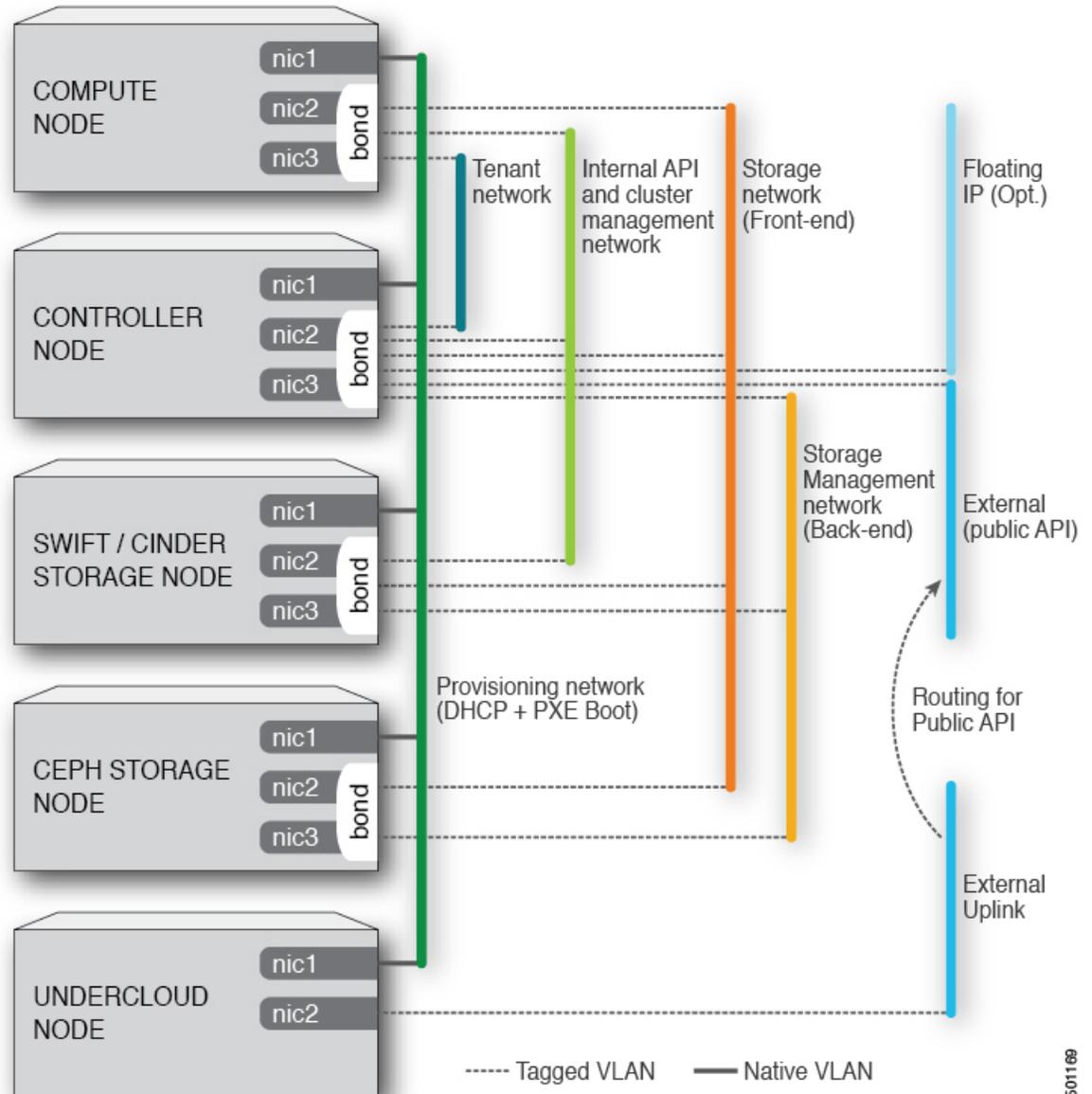
### APIC とネットワークの設定

ここでは、APIC およびネットワークのセットアップ方法について説明します。

次の図に示すようなネットワーク レイアウトについては、OpenStack Platform Director ドキュメントの「[Network Planning](#)」の項を参照してください。

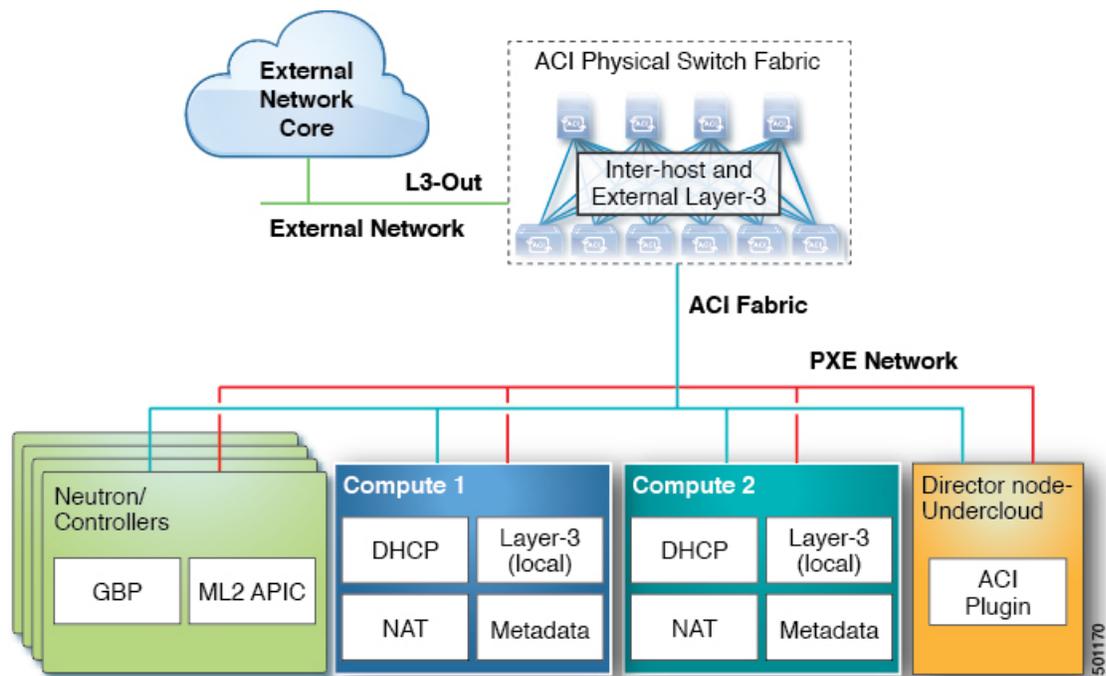
詳細については、「[関連資料 \(6 ページ\)](#)」を参照してください。

図 1: 一般的な OpenStack Platform トポロジ



501169

図 2: ACI プラグインを使用した Red Hat OpenStack Platform 8 のインストールの一般的なトポロジ



- PXE ネットワークはアウトオブバンド（OOB）であり、専用インターフェイスを使用します。
- PXE を除くすべての OpenStack Platform（OSP）ネットワークは、ACI を介するインバンド（IB）です。
  - API - VLAN 10
  - ストレージ - VLAN 11
  - ストレージ管理 - VLAN 12
  - テナント - VLAN 13
  - 外部 - VLAN 14
  - ACI インフラ - VLAN 4093
- L3-Out が事前設定されています（この例では L3-Out と呼ばれます。EPG は L3-Out-EPG です）。

ACI のインバンド設定を準備するには、これらのネットワーク用に作成された物理ドメイン、およびエンドポイントグループ（EPG）へのスタティック バインディングを使用できます。これにより、必要な物理ドメインと接続可能アクセス エンティティ プロファイル（AEP）が作成されます。AEP でインフラ VLAN を有効にする必要があることに注意してください。詳細については、以下を参照してください。

[http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/b\\_KB\\_Creating\\_AEP\\_Physical\\_Domains\\_VLANS\\_to\\_Deploy\\_an\\_EPG\\_on\\_a\\_Specific\\_Port.html](http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/b_KB_Creating_AEP_Physical_Domains_VLANS_to_Deploy_an_EPG_on_a_Specific_Port.html)

## 手順

- ステップ 1** APIC GUI（詳細モード）にログインし、OpenStack Platform のインストールに必要な VLAN の VLAN プールを作成します。
- メニューバーで **[Fabric] > [Access Policies] > [Pools]** を選択し、[VLAN] を右クリックして VLAN プールを作成します。
  - [Name] フィールドに、VLAN 範囲の名前空間ポリシー名（OSP8-infra）を入力します。
  - （任意）[Description] フィールドに、VLAN 範囲の名前空間ポリシーの説明を入力します。
  - [Encap Blocks] セクションで [+] アイコンをクリックし、encap ブロックの範囲を入力します。
  - [Submit] をクリックします。
- ステップ 2** 接続可能エンティティ プロファイルを作成して上記の物理ドメインを割り当てます。また、[Enable Infra VLAN] が選択されていることを確認します。
- メニューバーで **[Fabric] > [Access Policies] > [Global Policies]** を選択し、[Attachable Access Entity Profile] を右クリックして接続可能アクセス エンティティ プロファイルを作成します。
  - [Name] フィールドに、接続可能アクセスエンティティプロファイルの名前（OSP8-AEP）を入力します。
  - （任意）[Description] フィールドに、接続可能アクセスエンティティプロファイルの説明を入力します。
  - [Enable Infrastructure VLAN] チェックボックスをオンにしてインフラストラクチャ VLAN を有効にします。
  - [Domains (VMM, Physical or External) To Be Associated To Interfaces:] セクションで [+] アイコンをクリックし、ドロップダウンリストからドメインプロファイルを選択して [Update] をクリックします。
  - [次へ] をクリックします。
  - [Finish] をクリックします。
- ステップ 3** 物理ドメインを作成して VLAN プールを割り当てます。
- メニューバーで **[Fabric] > [Access Policies] > [Physical and External Domains]** を選択し、[Physical Domains] を右クリックして物理ドメインを作成します。
  - [Name] フィールドに、物理ドメインの名前（OSP8-Phys）を入力します。
  - [Associated Attachable Entity Profile] フィールドで、関連付けられた接続可能エンティティプロファイルを選択します。
  - [VLAN Pool] フィールドで VLAN プール（[OSP8-infra-dynamic]）を選択します。
  - [Submit] をクリックします。
- ステップ 4** 別のテナントで、[Common] を使用してアプリケーションプロファイル（例：OSP-8）を作成することもできます。OSP ネットワークの EPG、ブリッジドメイン、および VRF を作成しま

す。PXE ネットワークも ACI にアクセスする場合は、PXE の EPG と BD も作成します（この例では省略します）。

**ステップ 5** 必要な VLAN のスタティック バインディング（パス）を追加します。[:Static Binding Paths] を表示するには、EPG を展開する必要があります。

- a) 作成した物理ドメインがこの EPG に接続されていることを確認します。[Application Profiles] > [EPG] > [EPG\_name] > [Domains] を使用して物理ドメインを追加できます。
- b) メニュー バーで [Tenants] > [Tenant common] > [Application Profiles] > [ACI-OSP8] > [Application EPGs] > [EPG API] > [Static Binding Paths] を選択します。

**ステップ 6** EPG に物理ドメインが接続されていることを確認します。

(注) ACI を前述のネットワーク用にプロビジョニングする必要があります（テナント、外部、およびフローティング IP ネットワークを除く）。これには、必要な物理ドメインと接続エンティティプロファイルの作成が含まれます。接続エンティティプロファイルでインフラ VLAN を有効にする必要があることに注意してください。

ACI の OpenStack 導入に対する準備はこれで完了です。

## オーバークラウドの設定

Red Hat OpenStack Platform Director のインストール マニュアルに従って、OpenStack Platform Director を用意し、適切な導入ファイルとリソース ファイルを作成してください。



(注) オーバークラウド ノードは、書き込み時にローカルドメインの ipv6 DNS エントリの解決を試行するため、DNS サーバが NXDomain を送信する代わりに名前を解決しようとした場合に著しく速度が低下する可能性があります。著しく速度が低下した場合は、DNS サーバが正しく設定されていることを確認してください。

OpenStack Platform (OSP) Director を設定したら、導入を続行する前に ACI の TripleO オークストレーションをインストールする必要があります。

## OpFlex オークストレーションを備えた ACI の準備

オーバークラウドで ACI OpFlex をインストールして有効化するために必要な手順の概要は次のとおりです。

- 必要なソフトウェア パッケージを含むようにアンダークラウドを変更する。
- オーバークラウド イメージの一部である Neutron Puppet マニフェストに追加する。
- OpFlex Puppet マニフェストを追加する。
- アンダークラウド tripleO インフラストラクチャ上の一部のファイルを変更する。

- ACI 関連のパラメータ値を提供する HEAT 環境ファイルを作成する。
- 上記の変更後、`openstack overcloud deploy` コマンドを使用してオーバークラウドをプロビジョニングし、`openstack overcloud deploy` コマンドに新しい環境ファイルを追加できる。

## OpFlex オークストレーションを備えたアンダークラウド ACI の準備

ここでは、OpFlex オークストレーションを備えた ACI の統合パッケージをインストールする方法を説明します。



(注) 次に示すのは、ACIOpFlex RPM パッケージをホストするアンダークラウドでローカル RPM リポジトリを自動的に作成する手順です。

### 手順

**ステップ 1** ユーザ `stack` としてアンダークラウドにログインします。

**ステップ 2** `stackrc` ファイルを読み込みます。

例：

```
$ source stackrc
```

**ステップ 3** `cisco.com` から最新の ACI OSP (`tripleo ciscoaci`) rpm をダウンロードします。

詳細については、[APIC OpenStack プラグイン](#)を参照してください。

**ステップ 4** rpm をインストールします。これによって依存関係がインストールされます。rpm コマンドを使用して rpm をインストールする場合、一部の依存関係は手動によるインストールが必要な場合があります。

例：

```
$ sudo yum --nogpgcheck localinstall <rpm file>
```

## オーバークラウドのインストール

ここでは、オーバークラウドのインストール方法について説明します。

### 手順

**ステップ 1** `/opt/tripleo-ciscoaci/example_ciscoaci.yaml` ファイルを `/home/stack/templates/apic_gbp_config.yaml` ファイルにコピーします。

apic\_gbp\_config.yaml ファイルを編集し、設定の詳細を反映するように parameter\_defaults を変更します。

```
parameter_defaults:
  NeutronCorePlugin: 'ml2plus'
  NeutronServicePlugins: 'group_policy,ncp,apic_aim_13'
  NeutronEnableIsolatedMetadata: true
  EnablePackageInstall: true
  ACIYumRepo: http://10.10.250.10/acirepo
  ACIApicHosts: 172.31.218.136,172.31.218.137,172.31.218.138
  ACIApicUsername: admin
  ACIApicPassword: cisco123
  ACIApicSystemId: osp10_cs
  ACIApicEntityProfile: f-aep
  ACIApicInfraVlan: 4093
  ACIApicInfraSubnetGateway: 10.0.0.30
  ACIApicInfraAnycastAddr: 10.0.0.32
  ACIOpflexUplinkInterface: nic2
  ACIOpflexEncapMode: vxlan
  ACIOpflexVlanRange: 1200:1300
  NeutronEnableForceMetadata: true
  ACIOpflexBridgeToPatch: br-custom
```

パラメータ	説明
ACIYumRepo: http://10.10.250.10/acirepo	URL 内の IP アドレスは、ディレクタの IP アドレスに置き換える必要があります。これは OpFlex RPM のインストール元です。tripleo_ciscoaci パッケージをインストールすると、自動的にリポジトリが作成されます。
ACIApicHosts: 172.31.218.136,172.31.218.137,172.31.218.138	APIC の IP アドレスまたはホスト名のリストです。
ACIApicUsername: admin	APIC ユーザ名です。
ACIApicPassword: cisco123	APIC パスワードです。
ACIApicSystemId: osp10_cs	特定の OpenStack インスタンスを識別する一意の文字列である必要があります。
ACIApicEntityProfile: f-aep	ACI の VMM ドメインに接続する AAEP の名前です。この AEP は、オーバークラウドをインストールする前に手動で作成する必要があります。
ACIApicInfraVlan: 4093	ACI インフラ VLAN は OpFlex インフラ VLAN です。これは、ACI ファブリックの初期化時に選択されます。
ACIApicInfraSubnetGateway: 10.0.0.30	インフラ VLAN の SVI に割り当てられているエニーキャスト IP アドレスです。

パラメータ	説明
ACIApicInfraAnycastAddr: 10.0.0.32	この IP アドレスは、リーフスイッチ上でインターフェイスループバック 1023 に割り当てられたエニーキャスト IP アドレスと一致します。
ACIOpflexUplinkInterface: nic2	このインターフェイスは OpFlex に使用されません。これは、個別またはボンディングされたファブリック インターフェイスです。OSP Director テンプレートのガイドラインに従ってインターフェイス名を決定します。
ACIOpflexEncapMode: vxlan	コンピューティングノードとリーフスイッチの間で使用されるカプセル化は、vxlan または vlan です。
ACIOpflexVlanRange: 1200:1300	カプセル化用の VLAN 範囲です。vlan カプセル化を使用する場合にのみ必要です。
NeutronEnableForceMetadata: true	OpFlex の最適化されたメタデータを有効にするために必要です。
ACIOpflexBridgeToPatch: br-custom	このパラメータは、VLAN カプセル化とカスタム テンプレートを使用する場合にのみ必要です。ファブリック アップリンク インターフェイス (またはボンド インターフェイス) に接続されたブリッジの名前に設定する必要があります。Red Hat テンプレートのデフォルトブリッジは「br-ex」です。デフォルトの「br-ex」を導入に使用する場合、このパラメータは必要ありません。デフォルトブリッジを使用しない場合は値をブリッジ名に設定すると、このブリッジと統合ブリッジ「br-int」の間でパッチが作成されます。

**ステップ 2** /home/stack/templates/network-environment.yaml ファイルを編集して次の値を設定します。

```
NeutronEnableTunnelling: False
NeutronTunnelTypes: ""
NeutronEnableL3Agent: False
NeutronEnableOVSAgent: False
```

(注) これらの設定は、OpFlex エージェントによって置き換えられたデフォルト エージェントが無効になっていることを確認するために必要です。示されているとおりにこれらの設定を維持することをお勧めします。

**ステップ 3** インストール マニュアルの説明に従って、オーバークラウドを導入します。

詳細については、『*Director Installation and Usage Red Hat OpenStack Platform*』の第7章を参照してください。

a) 作成された新しい環境ファイルを展開します。

例：

```
openstack overcloud deploy --templates -e
/usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml -e
~/templates/network-environment.yaml -e
~/templates/apic_gbp_config.yaml
```

**ステップ4** 正常に展開されると、該当するVMMドメインがAPICで作成されます。OpenStackネットワークを作成する前に、必ずこのVMMドメインを適切な接続エンティティプロファイルに追加してください。

(注) 統合モードで外部接続を設定する手順は変更されました。詳細については、<https://github.com/noironetworks/opflex-documentation/wiki>の「Adding an OpenStack External Network in Unified Mode」の項を参照してください。

**ステップ5** 複数の物理ネットワークと階層型ポートバインディング (HPB) を設定します。

これで、ACIでHPBおよび複数の物理ネットワークを自動的に設定してOSPを導入できます。プラグインは、HPBの複数のメカニズムドライブの仕様をサポートします。HPBを使用する場合は、特定の物理ネットワークのNetworkVLANRanges用VLANプールを使用して、ACIでHPB物理ネットワークごとに物理ドメインを事前に作成しておく必要があります。

物理ネットワークと物理インターフェイスまたはボンドの関係を指定するには、以下を実行します。

ステップ1のapic\_gbp\_config.yamlファイルに次のパラメータを追加する必要があります。

```
NeutronPhysicalDevMappings: physnet1:ens11,physnet2:ens7,physnet3:bond1
NeutronNetworkVLANRanges: physnet1:1200:1250,physnet2:1251:1300,physnet3:1301:1350
ACIMechanismDrivers: 'sriovnicswitch,apic_aim'
```

パラメータ	説明
NeutronPhysicalDevMappings	このパラメータは、どのインターフェイスがどの物理ネットワークに属するかを指定します。このパラメータが正常に機能するように、名前が pdom_ で始まる APIC 物理ドメインを事前に作成しておく必要があります。  次に例を示します。  physnet2 の場合は、pdom_physnet2 という名前の物理ドメインを作成します。
NeutronNetworkVLANRanges	APIC の IP アドレスまたはホスト名のリストです。

パラメータ	説明
ACIMechanismDrivers	MI2構成で設定するメカニズムドライバです。HPB が正常に機能するには、リストの最後が apic_aim である必要があります。

## 参考リンク

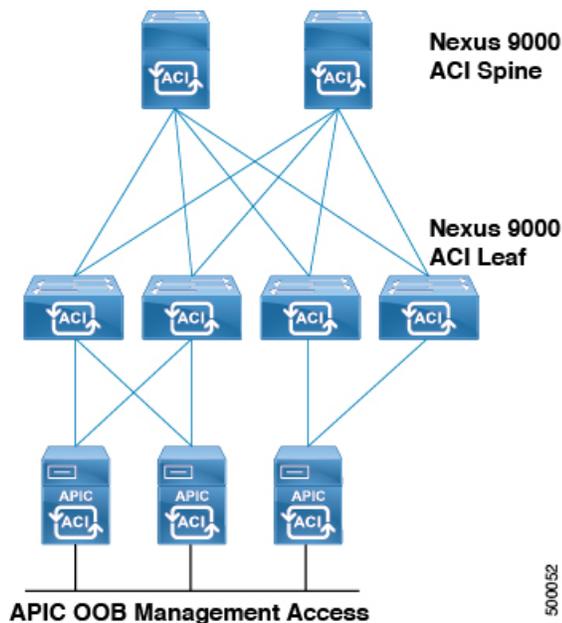
- 『*Director Installation and Usage Red Hat OpenStack Platform 8*』ドキュメント：  
<https://access.redhat.com/documentation/en/red-hat-openstack-platform/version-8/director-installation-and-usage/#chap-Introduction>
- 『*VLAN static binding to ports*』ナレッジ ベース記事：  
[http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/b\\_KB\\_Creating\\_AEP\\_Physical\\_Domains\\_VLANS\\_to\\_Deploy\\_an\\_EPG\\_on\\_a\\_Specific\\_Port.html](http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/b_KB_Creating_AEP_Physical_Domains_VLANS_to_Deploy_an_EPG_on_a_Specific_Port.html)

## ACI ファブリック初期化の例

このソリューション例は、ファブリック名とコントローラ IP アドレッシング以外のすべての設定がデフォルト状態で APIC にインストールされた基本的なスパイン/リーフ スイッチング ファブリックに基づいています。可用性の高いクラスタを形成するために、3 台の APIC が使用されています。それぞれの APIC はファブリック内の 1 台以上のリーフ スイッチに接続されています。コントローラ サービスの可用性を向上させるために、多様なリーフ スイッチを使用して複数の APIC を接続することをお勧めします。

スイッチング システムは、APIC クラスタの有無に関わらずトラフィックを転送し続けます。ファブリックのすべての設定はクラスタによって推進されるため、APIC の接続が確立されていない状態では、設定の追加、変更、削除は一切できません。ファブリックの管理制御がファブリック自体に依存することを避けるために、次の図に示すように APIC のそれぞれにアウトオブバンド (OOB) ネットワーク接続が必要です。

図 3: APIC クラスタ接続



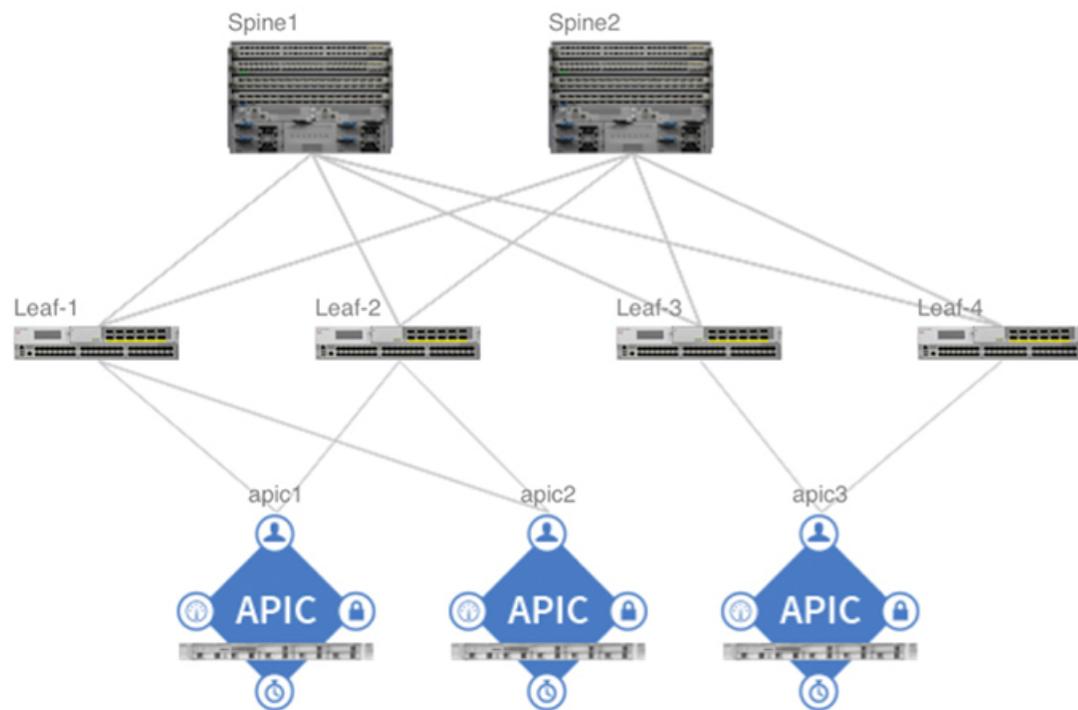
## 手順

- ステップ 1** ACIファブリックの設定では、ファブリックを検出する前にファブリック内の各スイッチのシリアル番号をメモすることをお勧めします。各スイッチのコンソールポートもターミナルサーバに接続して、ACI ファブリックの状態に関係なく常に管理制御が存在する状態が理想的です。ACI ソフトウェアイメージを実行するスイッチにログインする際にシリアル番号を取得するには、ACI スwitch の CLI で **show inventory** コマンドを入力し、プライマリ システムのシリアル番号をメモします。この番号はファブリック検出時に APIC に表示されるため、スキーム内の正しい名前とノードの番号付けをデバイスに割り当てることができます。
- ステップ 2** APIC がファブリック内のスイッチを検出および登録できるようにするために、APIC GUI（詳細モード）にログインします。
- メニューバーで、**[Fabric] > [Inventory]** を選択します。
  - [Navigation]** ペインで **[FabricMembership]** を選択します。
  - [Work]** ペインには、APIC によって検出された最初のスイッチのエントリが表示されます。
  - シリアル番号を基に、これがクラスタ内の最初の APIC で想定される最初のスイッチであることを確認します。
  - [Work]** ペインでスイッチを選択し、右クリックして **[Register Switch]** を選択します。

(注) 以降のトラブルシューティングおよび仮想ポートチャネル (vPC) のペアリングプランに適した論理ノード ID 番号とノード名を割り当てます。たとえば最初の 2 台のリーフスイッチでは、ノード ID を 101 と 102、名前を leaf1 と leaf2 にします。

- ステップ 3** 最初のリーフが検出されると、システムはそのリーフ経由でスパインスイッチを検出し、そのスパインスイッチを使用して残りのリーフスイッチを検出します。スパイン/リーフ ファブリックのレイアウトに従って、論理ノード ID 番号と名前を割り当てる追加ノードを登録します。
- ステップ 4** 想定どおりにトポロジが検出され、物理的に接続されていることを視覚的に確認し、次の操作を実行します。
- メニューバーで、**[Fabric] > [Inventory]** を選択します。
  - [Navigation] ペインで **[Topology]** を選択します。

図 4: 検出されたスパイン/リーフ トポロジ



- ファブリックが検出されたら、**[Admin] > [Firmware]** を選択し、すべての APIC とファブリック ノード（スイッチ）上で稼働しているファームウェアバージョンを検証します。必要に応じて、初期設定を開始する前に、最新バージョンまたは一貫したバージョンにアップグレードします。





## 第 3 章

# ACI と OSP のアップグレード

この章の内容は、次のとおりです。

- [アップグレード ガイドライン](#) (19 ページ)
- [ACI パッケージのアップグレード](#) (21 ページ)
- [OSP メジャー リリース アップグレードの実行](#) (22 ページ)

## アップグレード ガイドライン

ここでは、アップグレード ガイドラインについて説明します。

リリース 2.3.1 以降の OpenStack 用 Cisco ACI プラグインは、リリースに対応する ACI ファブリック ソフトウェア、および以降のすべての ACI ファブリック ソフトウェアをサポートします。OpenStack 用 ACI プラグインは、各リリースに対応するバージョンより前の ACI ファブリック ソフトウェアをサポートしません。たとえば OpenStack 用 ACI プラグイン リリース 3.1 では、ACI ファブリック ソフトウェア 3.1 以降はサポートされますが、3.0 以前のバージョンはサポートされません。

## アップグレード前のガイドライン

最初に Cisco ACI プラグインをアップグレードする必要があります。

さまざまな OpenStack バージョンとプラグインの互換性について詳しくは、次の URL にある『*Cisco ACI Virtualization Compatibility Matrix*』を参照してください。

<https://www.cisco.com/c/dam/en/us/td/docs/Website/datacenter/aci/virtualization/matrix/virtmatrix.html>



(注) OpenStack プラグインのアップグレード前に ACI ファブリックをアップグレードする必要がある例外が 1 つあります。ACI ファブリックを 2.x から 3.x リリースにアップグレードする場合は、OpenStack プラグイン 2.3.1-newton-20180419 以降の実行が必要です。下位バージョンのプラグインを実行している場合、まずはそのプラグインに付属していた `agent_ovs` を 2.3.1-newton-20180419 に付属していた `agent_ovs` にアップグレードする必要があります。詳細については、[agent\\_ovs の更新](#) (20 ページ) を参照してください。

## agent\_ovs の更新

ACI ファブリックを 2.x から 3.x リリースにアップグレードする場合は、OpenStack プラグイン 2.3.1-newton-20180419 以降の実行が必要です。下位バージョンのプラグインを実行している場合、まずはそのプラグインに付属していた agent\_ovs を 2.3.1-newton-20180419 に付属していた agent\_ovs にアップグレードする必要があります。

Openstack VMM に接続されている各サーバに root としてログインし、次の手順を実行します。

### 手順

**ステップ 1** APIC OpenStack およびコンテナ プラグインのソフトウェアダウンロードリンクから Openstack Newton ACI プラグインの RPM パッケージをダウンロードします。

a) ソフトウェア ダウンロード リンクにアクセスします。

<https://software.cisco.com/download/home/285968390/type>

b) [APIC OpenStack and Container Plugins] をクリックします。

c) [2.3(1.20180419)] を選択します。

d) **RPM packages for Openstack Newton ACI plugins dist-rpms-2.3.1-newton-20180419.tar.gz** を選択し、ダウンロードアイコンをクリックしてホーム ディレクトリに gzip ファイルをダウンロードします。

**ステップ 2** ディレクトリを作成し、そのファイルのコンテンツを抽出します。

例：

```
# mkdir newton-rpms
# cd newton-rpms
# tar -xzf ../dist-rpms-2.3.1-newton-20180419.tar.gz
```

**ステップ 3** 次の RPM を更新します。

例：

```
# rpm -U ./libmodelgbp-1.3.0-44.e17.centos.x86_64.rpm
# rpm -U ./libopflex-1.3.0-44.e17.centos.x86_64.rpm
# rpm -U ./agent-ovs-1.3.0-44.e17.centos.x86_64.rpm
```

**ステップ 4** agent-ovs を再起動します。

例：

```
# service agent-ovs restart
```

## アップグレードのガイドライン

次の URL にある『Cisco APIC Management, Installation, Upgrade, and Downgrade Guide』の情報に従って Cisco ACI ファブリックをアップグレードできます。

[https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/2-x/managing\\_ACI\\_fabric\\_upgrades\\_and\\_downgrades/b\\_Managing\\_ACI\\_Fabric\\_Upgrades\\_and\\_Downgrades/b\\_Managing\\_ACI\\_Fabric\\_Upgrades\\_and\\_Downgrades\\_chapter\\_01.html](https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/2-x/managing_ACI_fabric_upgrades_and_downgrades/b_Managing_ACI_Fabric_Upgrades_and_Downgrades/b_Managing_ACI_Fabric_Upgrades_and_Downgrades_chapter_01.html)

Cisco ACI プラグインと Cisco ACI ファブリックのリリースの組み合わせがサポートされている場合に限り、プラグインをアップグレードせずに Cisco ACI ファブリックをアップグレードすることもできます。詳細については、次の URL にある『Cisco ACI Virtualization Compatibility Matrix』を参照してください。

<https://www.cisco.com/c/dam/en/us/td/docs/Website/datacenter/aci/virtualization/matrix/virtmatrix.html>

## アップグレード後のガイドライン

Cisco ACI ファブリックをアップグレードしたら、必要に応じて、アップグレード後の ACI ファブリック コードと同等または下位のバージョンに OpenStack ACI パッケージをアップグレードできます。『OpenStack ACI Plugin Release Notes』で固有の情報も確認してください。

[https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html#Cisco\\_APIC\\_OpenStack\\_Release\\_Notes](https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html#Cisco_APIC_OpenStack_Release_Notes)

OpenStack ACI プラグインをアップグレードする方法の詳細については、[ACI パッケージのアップグレード \(21 ページ\)](#) を参照してください。

## ACI パッケージのアップグレード

ACI リリースのアップグレードはライブで実行される可能性があるため、Red Hat Director のドキュメントに従う必要があります。詳細については、次のサイトで「Director-Based Environments: Performing Updates to Minor Versions」の章を参照してください。

[https://access.redhat.com/documentation/en-us/red\\_hat\\_openstack\\_platform/10/html-single/upgrading\\_red\\_hat\\_openstack\\_platform/](https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html-single/upgrading_red_hat_openstack_platform/)



(注) 2.3 より前の ACI パッケージを 2.3 以降のリリースに更新することはできません。つまり、アップグレードできるのはコンポーザブル サービス展開のみです。

導入が完了したオーバークラウドでアップグレードを実行する手順の例を示します。

### 手順

**ステップ 1** tripleo-ciscoaci RPM の更新バージョンを CCO から OSP Director にコピーします。

**ステップ 2** yum を使用して tripleo-ciscoaci パッケージを更新します。

```
yum update tripleo-ciscoaci-<version number>.rpm
```

**ステップ 3** Red Hat ドキュメントの指定どおりに update-plan を実行します。

```
openstack overcloud deploy --update-plan-only --templates -e \
/home/stack/templates/scheduler_hints_env.yaml -e \
/usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e ~/templates/network-environment.yaml -e /home/stack/templates/aci_cs.yaml \
-e ~/templates/rhel-registration-resource-registry.yaml -e \
~/templates/environment-rhel-registration.yaml --control-scale 1 --compute-scale 2
```

**ステップ4** スタックを更新して新しいパッケージをオーバークラウドにプッシュします。

```
openstack overcloud update stack -i overcloud
```

上記のコマンドを実行すると、オーバークラウドがプラグインの新しいバージョンで更新されます。

## OSP メジャー リリース アップグレードの実行

メジャー リリース アップグレードについては、Red Hat Director の該当するドキュメントに従う必要があります。詳細については、次のサイトで「Director-Based Environments: Performing a Major Version Upgrade」の章を参照してください。

[https://access.redhat.com/documentation/en-us/red\\_hat\\_openstack\\_platform/11/html-single/upgrading\\_red\\_hat\\_openstack\\_platform/#chap-Director-Major-Upgrade](https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/11/html-single/upgrading_red_hat_openstack_platform/#chap-Director-Major-Upgrade)



(注) 2.3 以降のリリースでは、このドキュメントで説明されているように、コンポーザブルサービスによるライブアップグレードのみがサポートされます。以前のバージョンからのアップグレードはサポートされていません。

一部のテンプレートと tripleo-ciscoaci パッケージの更新が必要な場合もあります。正しい手順を使用することによって、オーバークラウドのライブアップグレードを実行できます。手順の概要は次のとおりです。

### 手順

**ステップ1** 既存の OpenStack バージョンから tripleo-ciscoaci パッケージを削除します。

**ステップ2** Red Hat Director ドキュメントの 3.4.2 項「Upgrading the Overcloud Images」の最後まで手順を実行し、アンダークラウドを停止してアップグレードします。

**ステップ3** すべてのオーバークラウドコントローラ ノードで aim プロセスを停止します。この操作を行うには、ディレクタで次のコマンドを使用します。

例：

```
source stackrc
for IP in $(nova list | grep ACTIVE | sed 's/.*ctlplane=//' | sed
's/ |//'); do ssh -o UserKnownHostsFile=/dev/null -o
StrictHostKeyChecking=no heat-admin@$IP "sudo systemctl stop
aim-event-service-rpc; sudo systemctl stop aim-aid; sudo systemctl
stop aim-event-service-polling"; done
```

**ステップ 4** Red Hat の新しいメジャー リリースの `tripleo-ciscoaci` パッケージをインストールします。

**ステップ 5** 特定のリリースに必要なテンプレートを更新します。

この手順には、ACI プラグインアップグレードに必要なリリース固有の変更点が該当する可能性があります。詳細については、[リリース固有の変更点 \(23 ページ\)](#) を参照してください。

**ステップ 6** Red Hat ドキュメントの残りの手順を実行してオーバークラウドをアップグレードします。

## リリース固有の変更点

ここでは、リリース間でアップグレードを必要とするテンプレートに対するリリース固有の変更点について説明します。リリース バージョンに対応するセクションに従ってください。

OSP 11 および Ocata リリースの OpFlex プラグインの場合、パッケージ「`neutron-ml2-cisco`」が廃止されたため、OSP 11 へのアップグレード時に削除する必要があります。OSP 10 から OSP 11 にアップグレードする場合は、`openstack-tripleo-templates` で次の変更を行う必要があります。

### 手順

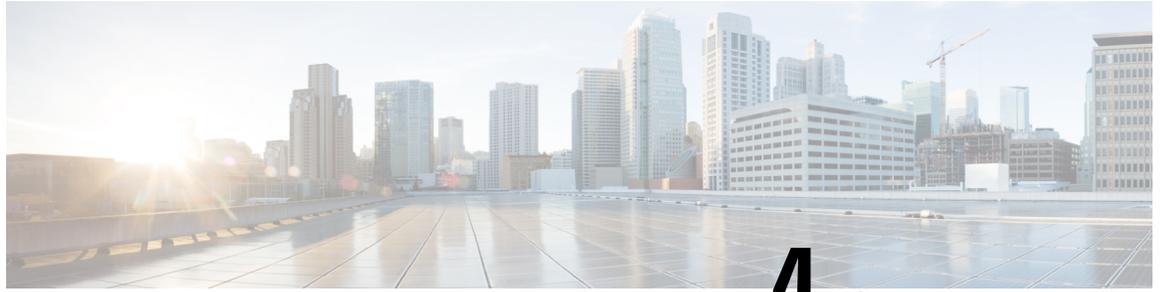
**ステップ 1** `/usr/share/openstack-tripleo-heat-templates/puppet/services/tripleo-packages.yaml` ファイルを編集し、強調表示されている行を追加します。

```
- name: Fail when rpm-python wasn't present
  fail: msg="rpm-python package was not present before this run! Check environment before re-running"
  when: rpm_python_check.changed != false
  tags: step0
- name: clean yum cache
  tags: step0
  command: yum clean all
  ignore_errors: True
- name: remove neutron-ml2-cisco
  tags: step3
  yum: name=neutron-ml2-driver-apic state=removed
- name: Update all packages
  tags: step3
  yum: name=* state=latest
```

**ステップ 2** `/usr/share/openstack-tripleo-heat-templates/extraconfig/tasks/tripleo_upgrade_node.sh` ファイルを編集し、強調表示されている行を追加します。

```
set -eu
yum -y remove neutron-ml2-driver-apic || true
NOVA_COMPUTE=""
```





## 第 4 章

# OpenStack 外部ネットワークの追加

この章の内容は、次のとおりです。

- [OpenStack 外部ネットワークの追加 \(25 ページ\)](#)

## OpenStack 外部ネットワークの追加

ここでは、OpenStack 外部ネットワークを追加する方法について説明します。

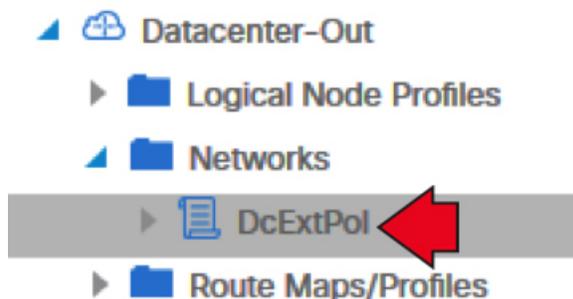


(注) 外部接続には、既存の L3Out と対応するエンドポイント グループ (EPG) が必要です。

OpenStack で外部ネットワークを設定するコマンドが変更され (新しい拡張が追加されました)、上記の既存の L3Out とその EPG に対応しています。設定には、APIC の L3Out 固有 EPG の DN が必要です。これには、次の 2 つの方法があります。

### 手順

**ステップ 1** APIC にアクセスし、API インспекタを使用して外部ネットワークの DN を取得します。



**ステップ 2** 以下に示すように、Openstack コントローラで `aimctl manager` コマンドを使用して `dn` を取得します。

(注) L3Out 「Datacenter-Out」 の例を参照してください。

例 :

```
aimctl manager external-network-find
+-----+-----+-----+
| tenant_name | l3out_name | name |
+-----+-----+-----+
| common     | Datacenter-Out | DcExtPol |
| common     | Management-Out | MgmtExtPol |
| common     | default       | MgmtNet |
+-----+-----+-----+

aimctl manager external-network-get common Datacenter-Out DcExtPol
+-----+-----+-----+
|Property          |Value          |
+-----+-----+-----+
| tenant_name      | common        |
| l3out_name       | Datacenter-Out |
| name             | DcExtPol      |
| display_name     |               |
| nat_epg_dn       |               |
| provided_contract_names | []           |
| consumed_contract_names | []           |
| monitored        | True          |
| dn               | uni/tn-common/out-Datacenter-Out/instP-DcExtPol |
+-----+-----+-----+
```

これで、上記の **dn** を使用して外部ネットワークを作成できます。

分散 NAT の場合 :

例 :

```
neutron net-create Datacenter-Out --router:external True --shared
--apic:distinguished_names \
type=dict ExternalNetwork=uni/tn-common/out-Datacenter-Out/instP-DcExtPol
```

NAT なしの場合 :

例 :

```
neutron net-create Datacenter-Out --router:external True --shared
--apic:distinguished_names \
type=dict ExternalNetwork=uni/tn-common/out-Datacenter-Out/instP-DcExtPol --apic:nat_type
""
```

フローティング IP については、目的のフローティング IP プール `cidr` を使用して外部ネットワークにサブネットを追加します。

SNAT の場合は、次のように外部ネットワークにサブネットを追加します。

例 :

```
neutron subnet-create Datacenter-Out 10.104.21.0/24 --name ext-subnet --disable-dhcp
--gateway \
10.104.21.1 --apic:snat_host_pool True
```

GBP 外部接続の場合は、最初に SNAT サブネットを使用して外部セグメントを作成します。

例 :

```
gbp external-segment-create Datacenter-Out --subnet-id <SNAT Subnet>
```

これで、このセグメントを通常どおりに使用してネットワーク サービス ポリシーを作成できるようになります。

例 :

```
gbp nat-pool-create nat-pool-0 --ip-pool 10.104.31.0/24 --external-segment Datacenter-Out
```

```
gbp network-service-policy-create --network-service-params  
type=ip_pool,name=nat-pool-0,value=nat_pool \  
net-svc-nat-0
```

---





## 付録 **A**

### 参考資料

---

この章の内容は、次のとおりです。

- [導入テンプレートの例 \(29 ページ\)](#)
- [階層型ポートバインディングの設定 \(34 ページ\)](#)

### 導入テンプレートの例

この付録では、導入ファイルの例を示します。これらは参照専用であり、実稼働システムを想定したものではありません。自身のトポロジと導入シナリオを反映した導入ファイルを作成してください。

#### **controller.yaml**

```
heat_template_version: 2015-04-30

description: >
  Software Config to drive os-net-config to configure VLANs for the
  controller role.

parameters:
  ControlPlaneIp:
    default: ''
    description: IP address/subnet on the ctlplane network
    type: string
  ExternalIpSubnet:
    default: ''
    description: IP address/subnet on the external network
    type: string
  InternalApiIpSubnet:
    default: ''
    description: IP address/subnet on the internal API network
    type: string
  StorageIpSubnet:
    default: ''
    description: IP address/subnet on the storage network
    type: string
  StorageMgmtIpSubnet:
    default: ''
    description: IP address/subnet on the storage mgmt network
    type: string
```

```

TenantIpSubnet:
  default: ''
  description: IP address/subnet on the tenant network
  type: string
ManagementIpSubnet: # Only populated when including environments/network-management.yaml

  default: ''
  description: IP address/subnet on the management network
  type: string
ExternalNetworkVlanID:
  default: 10
  description: Vlan ID for the external network traffic.
  type: number
InternalApiNetworkVlanID:
  default: 20
  description: Vlan ID for the internal_api network traffic.
  type: number
StorageNetworkVlanID:
  default: 30
  description: Vlan ID for the storage network traffic.
  type: number
StorageMgmtNetworkVlanID:
  default: 40
  description: Vlan ID for the storage mgmt network traffic.
  type: number
TenantNetworkVlanID:
  default: 50
  description: Vlan ID for the tenant network traffic.
  type: number
ManagementNetworkVlanID:
  default: 60
  description: Vlan ID for the management network traffic.
  type: number
ExternalInterfaceDefaultRoute:
  default: '10.0.0.1'
  description: default route for the external network
  type: string
ControlPlaneSubnetCidr: # Override this via parameter_defaults
  default: '24'
  description: The subnet CIDR of the control plane network.
  type: string
DnsServers: # Override this via parameter_defaults
  default: []
  description: A list of DNS servers (2 max for some implementations) that will be
added to resolv.conf.
  type: comma_delimited_list
EC2MetadataIp: # Override this via parameter_defaults
  description: The IP address of the EC2 metadata server.
  type: string

resources:
  OsNetConfigImpl:
    type: OS::Heat::StructuredConfig
    properties:
      group: os-apply-config
      config:
        os_net_config:
          network_config:
            -
              type: interface
              name: nic1
              dns_servers: {get_param: DnsServers}
              addresses:
                -

```

```

        ip_netmask:
          list_join:
            - '/'
            - - {get_param: ControlPlaneIp}
              - {get_param: ControlPlaneSubnetCidr}
    routes:
    -
      ip_netmask: 169.254.169.254/32
      next_hop: {get_param: EC2MetadataIp}
-
  type: ovs_bridge
  name: {get_input: bridge_name}
  members:
  -
    type: interface
    name: nic2
    # force the MAC address of the bridge to this interface
    primary: true
  -
    type: vlan
    vlan_id: {get_param: ExternalNetworkVlanID}
    addresses:
    -
      ip_netmask: {get_param: ExternalIpSubnet}
    routes:
    -
      default: true
      next_hop: {get_param: ExternalInterfaceDefaultRoute}
  -
    type: vlan
    vlan_id: {get_param: InternalApiNetworkVlanID}
    addresses:
    -
      ip_netmask: {get_param: InternalApiIpSubnet}
  -
    type: vlan
    vlan_id: {get_param: StorageNetworkVlanID}
    addresses:
    -
      ip_netmask: {get_param: StorageIpSubnet}
  -
    type: vlan
    vlan_id: {get_param: StorageMgmtNetworkVlanID}
    addresses:
    -
      ip_netmask: {get_param: StorageMgmtIpSubnet}
  -
    type: vlan
    vlan_id: {get_param: TenantNetworkVlanID}
    addresses:
    -
      ip_netmask: {get_param: TenantIpSubnet}
#- # Uncomment when including environments/network-management.yaml
# type: vlan
# vlan_id: {get_param: ManagementNetworkVlanID}
# addresses:
# -
#   ip_netmask: {get_param: ManagementIpSubnet}

outputs:
  OS::stack_id:
    description: The OsNetConfigImpl resource.
    value: {get_resource: OsNetConfigImpl}

```

## compute.yaml

```

heat_template_version: 2015-04-30

description: >
  Software Config to drive os-net-config to configure VLANs for the
  compute role.

parameters:
  ControlPlaneIp:
    default: ''
    description: IP address/subnet on the ctlplane network
    type: string
  ExternalIpSubnet:
    default: ''
    description: IP address/subnet on the external network
    type: string
  InternalApiIpSubnet:
    default: ''
    description: IP address/subnet on the internal API network
    type: string
  StorageIpSubnet:
    default: ''
    description: IP address/subnet on the storage network
    type: string
  StorageMgmtIpSubnet:
    default: ''
    description: IP address/subnet on the storage mgmt network
    type: string
  TenantIpSubnet:
    default: ''
    description: IP address/subnet on the tenant network
    type: string
  ManagementIpSubnet: # Only populated when including environments/network-management.yaml

    default: ''
    description: IP address/subnet on the management network
    type: string
  InternalApiNetworkVlanID:
    default: 20
    description: Vlan ID for the internal_api network traffic.
    type: number
  StorageNetworkVlanID:
    default: 30
    description: Vlan ID for the storage network traffic.
    type: number
  TenantNetworkVlanID:
    default: 50
    description: Vlan ID for the tenant network traffic.
    type: number
  ManagementNetworkVlanID:
    default: 60
    description: Vlan ID for the management network traffic.
    type: number
  ControlPlaneSubnetCidr: # Override this via parameter_defaults
    default: '24'
    description: The subnet CIDR of the control plane network.
    type: string
  ControlPlaneDefaultRoute: # Override this via parameter_defaults
    description: The default route of the control plane network.
    type: string
  DnsServers: # Override this via parameter_defaults
    default: []
    description: A list of DNS servers (2 max for some implementations) that will be

```

```

added to resolv.conf.
  type: comma_delimited_list
  EC2MetadataIp: # Override this via parameter_defaults
  description: The IP address of the EC2 metadata server.
  type: string

resources:
  OsNetConfigImpl:
    type: OS::Heat::StructuredConfig
    properties:
      group: os-apply-config
      config:
        os_net_config:
          network_config:
            -
              type: interface
              name: nic1
              dns_servers: {get_param: DnsServers}
              addresses:
                -
                  ip_netmask:
                    list_join:
                      - '/'
                      - - {get_param: ControlPlaneIp}
                        - {get_param: ControlPlaneSubnetCidr}
            routes:
              -
                ip_netmask: 169.254.169.254/32
                next_hop: {get_param: EC2MetadataIp}
            -
              type: ovs_bridge
              name: {get_input: bridge_name}
              members:
                -
                  type: interface
                  name: nic2
                  # force the MAC address of the bridge to this interface
                  primary: true
                -
                  type: vlan
                  vlan_id: {get_param: InternalApiNetworkVlanID}
                  addresses:
                    -
                      ip_netmask: {get_param: InternalApiIpSubnet}
                -
                  type: vlan
                  vlan_id: {get_param: StorageNetworkVlanID}
                  addresses:
                    -
                      ip_netmask: {get_param: StorageIpSubnet}
                -
                  type: vlan
                  vlan_id: {get_param: TenantNetworkVlanID}
                  addresses:
                    -
                      ip_netmask: {get_param: TenantIpSubnet}
            # Uncomment when including environments/network-management.yaml
            #-
            # type: vlan
            # vlan_id: {get_param: ManagementNetworkVlanID}
            # addresses:
            # -
            # ip_netmask: {get_param: ManagementIpSubnet}

```

```

outputs:
  OS::stack_id:
    description: The OsNetConfigImpl resource.
    value: {get_resource: OsNetConfigImpl}

```

## 階層型ポートバインディングの設定

ここでは、`opflex` プラグインで動作する単一のルート I/O 仮想化 (SR-IOV) エージェントや、その他の VLAN ベースの `ml2` メカニズム エージェントの設定について説明します。これは階層型ポートバインディング (HPB) を使用して行います。設定に特別な修正を加える必要はありません。`opflex` に SR-IOV を設定するために必要な基本手順は次のとおりです。

HPB を使用する場合は、OpenStack によって作成されたネットワークの EPG にスタティック VLAN バインディングを作成することで、ACI にデータパス接続が確立されます。SR-IOVNIC での VLAN の設定や OVS (LBaaS の場合はロード バランサ) の設定など、データパス用に他の設定が必要な場合があります。これは、サードパーティのエージェントまたはメカニズム ドライバ (`sriovnicswitch` など) によって実行します。

これらの資産を作成する方法：

### 始める前に

データパスを設定するには、スタティック VLAN バインディングを使用します。プラグインには次の資産が必要です。

- 物理ドメインと適切な VLAN プール。
- `host-link` 情報 (どのコンピューティングノードのファブリックイーサネットインターフェイスがどのリーフスイッチポートに接続されているかを示す情報)
- `host-link-network-label` 情報 (コンピューティングノードのどのファブリックイーサネットインターフェイスが、どの物理ネットワークの提供に使用されているかを示す情報)

この情報は、導入に複数の物理ネットワークが使用される場合にのみ必要です。

### 手順

**ステップ 1** OpenStack Platform オーバークラウドを導入する前に、必要な物理ネットワークごとに 1 つの物理ドメインが作成済みであることを確認します。作成した物理ドメインの名前に `pdom_` プレフィックスを追加します。たとえば `physnet1` の場合は、`pdom_physnet1` を作成して適切な VLAN プールを付加します。

**ステップ 2** 通常、`host-link` 情報は LLDP 自動検出によって取得されます。次の例に示すように ACI 導入テンプレートの `ACIHostLinks` パラメータを使用して、静的に取得することもできます。

また、`NeutronNetworkVLANRanges` を設定し、`ACIMechanismDrivers` パラメータを使用してサードパーティのメカニズム ドライバを有効にして、`apic_aim` をリスト内の最後のメカニズムにする必要があります。

例：

```
NeutronPhysicalDevMappings: physnet1:ens11,physnet2:ens7,physnet3:ens9
NeutronNetworkVLANRanges:physnet1:1200:1250,physnet2:1251:1300,physnet3:1301:1350
ACIMechanismDrivers: 'sriovnicswitch,apic_aim'
ACIHostLinks: '{"101": [{"host01|ens11": "1/14"}], "102": [{"host02|ens9": "1/14"}]}'
```

---

