



Cisco APIC レイヤ 4 ～ 7 サービス導入ガイド

初版：2013年10月31日

最終更新：2014年08月01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/）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。

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザー側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

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.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

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: <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. (1110R)

© 2013-2014 Cisco Systems, Inc. All rights reserved.



目次

はじめに vii

対象読者 vii

表記法 vii

関連資料 ix

マニュアルに関するフィードバック x

マニュアルの入手方法およびテクニカル サポート x

概要 1

アプリケーションセントリック インフラストラクチャのレイヤ4～7サービスの導入について 1

サービス グラフについて 3

前提条件 5

テナントについて 5

セキュリティドメインについて 6

レイヤ3 ネットワークについて 6

ブリッジドメインについて 6

アプリケーション プロファイルについて 6

コントラクトについて 6

VLAN プールの設定 7

カプセル化ブロック範囲の設定 7

物理ドメインの設定 8

VMM ドメインの設定 9

VMM クレデンシャルの設定 10

vCenter/vShield コントローラ プロファイルの設定 10

テナントの設定 11

ブリッジドメインの設定 12

レイヤ3 ネットワークの設定 13

アプリケーションプロファイルの設定	14
コントラクトの設定	18
管理エンドポイントグループの設定	19
デバイスパッケージのインポート	23
デバイスパッケージについて	23
デバイスパッケージのインストール例	24
RESTでのデバイスパッケージのインストール時の注意事項	24
デバイスパッケージのインポート	25
デバイスクラスタの設定（論理デバイス）	27
デバイスクラスタについて（論理デバイス）	27
具象デバイスについて	27
デバイスクラスタの設定	28
具象デバイスの設定	30
仮想サービスデバイスに対するデバイスクラスタの設定	32
デバイスクラスタへの接続の設定	35
デバイスクラスタへの接続の設定について	35
テナントのVRFを使用したデバイスクラスタ内のデバイスへのインバンド接続について	36
管理テナントVRFを使用したデバイスクラスタ内のデバイスへのインバンド接続について	36
テナントのVRFを使用したデバイスクラスタ内のデバイスへのインバンド接続の設定	36
テナントのVRFを使用したデバイスクラスタ内のデバイスへのインバンド接続のREST APIによる設定	38
管理テナントVRFを使用したデバイスクラスタ内のデバイスへのインバンド接続の設定	39
管理テナントのVRFを使用したデバイスクラスタ内のデバイスへのインバンド接続のREST APIによる設定	39
デバイスクラスタの使用	41
デバイスクラスタ（論理デバイス）のコンテキストについて	41
論理デバイスコンテキストの設定	42
REST APIを使用した設定	43

デバイス クラスタ コンテキストを作成するための REST API による設定	43
デバイス クラスタに論理インターフェイスを追加するための REST API による設定	43
CLI コマンドを使用した設定	44
デバイス クラスタのインポート	45
デバイス クラスタのインポートの確認	45
REST API を使用したインポート	45
CLI コマンドを使用したインポート	46
サービス グラフの設定	49
サービス グラフについて	49
機能ノードについて	49
機能ノード コネクタについて	50
サービス グラフ接続について	50
端末ノードについて	50
サービス グラフ コンフィギュレーション パラメータについて	50
サービス グラフの設定	51
サービス グラフを作成するための REST API による設定	51
CLI コマンドを使用した設定	52
設定パラメータ	55
デバイス パッケージ仕様内のコンフィギュレーション パラメータ	55
デバイス パッケージ仕様の設定スコープ	58
デバイス パッケージ内のコンフィギュレーション パラメータの XML の例	58
抽象機能プロファイル内のコンフィギュレーション パラメータ	59
抽象機能プロファイルの設定スコープ	61
コンフィギュレーションパラメータを持つ抽象機能プロファイルに対する XML POST の例	62
サービス グラフでの抽象機能ノード内のコンフィギュレーション パラメータ	63
コンフィギュレーションパラメータを持つ抽象機能ノードに対する XML POST の例	67
各種の設定 MO 内のコンフィギュレーション パラメータ	68
コンフィギュレーションパラメータを持つアプリケーション EPG の XML POST の例	70

パラメータ解決	71
パラメータ解決時の MO の検索	72
サービス グラフの使用	75
サービス グラフとコントラクトの関連付け	75
CLI コマンドを使用した設定	76
REST API を使用した設定	77
セキュリティ ポリシーを作成するための REST API による設定	77
サービス グラフのモニタリング	79
サービス グラフ インスタンスのモニタリング	79
サービス グラフ エラーのモニタリング	81
サービス グラフ エラーの解決	81
仮想デバイスのモニタリング	87
CLI コマンドを使用した設定	88
サービス コンフィギュレーションの管理に対する管理ロールの設定	97
権限について	97
デバイス管理のロールの設定	98
サービス グラフ管理のロールの設定	98
デバイス パッケージのアップロードのロールの設定	98
デバイス クラスターのエクスポートのロールの設定	98
自動化の開発	101
REST API について	101
REST API の使用例	102



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](#), [vii ページ](#)
- [表記法](#), [vii ページ](#)
- [関連資料](#), [ix ページ](#)
- [マニュアルに関するフィードバック](#), [x 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](#), [x 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
bold	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。

表記法	説明
[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。
[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。イタリック体を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管しておいてください。

関連資料

アプリケーションセントリック インフラストラクチャのマニュアルセットには次のマニュアルが含まれます。

Web ベースのマニュアル

- 『*Cisco APIC Management Information Model Reference*』
- 『*Cisco APIC Online Help Reference*』
- 『*Cisco ACI MIB Support List*』

ダウンロード可能なマニュアル

- 『*Cisco Application Centric Infrastructure Release Notes*』
- 『*Cisco Application Centric Infrastructure Fundamentals Guide*』
- 『*Cisco APIC Getting Started Guide*』
- 『*Cisco APIC REST API User Guide*』
- 『*Cisco APIC Command Line Interface User Guide*』
- 『*Cisco APIC Faults, Events, and System Message Guide*』
- 『*Cisco APIC Layer 4 to Layer 7 Device Package Development Guide*』
- 『*Cisco APIC レイヤ 4～7 サービス導入ガイド*』

- 『Cisco ACI Firmware Management Guide』
- 『Cisco ACI Troubleshooting Guide』
- 『Cisco ACI NX-OS Syslog Reference Guide』
- 『Cisco ACI Switch Command Reference, NX-OS Release 11.0』
- 『Cisco ACI MIB Quick Reference』
- 『Cisco Nexus CLI to Cisco APIC Mapping Guide』
- 『Installing the Cisco Application Virtual Switch with the Cisco APIC』
- 『Configuring the Cisco Application Virtual Switch using the Cisco APIC』
- 『Application Centric Infrastructure Fabric Hardware Installation Guide』

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 Bug Search Tool (BST) の使用、サービス要求の送信、および追加情報の収集に関する情報については、<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>で『What's New in Cisco Product Documentation』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

概要

- [アプリケーションセントリック インフラストラクチャのレイヤ 4～7 サービスの導入について, 1 ページ](#)
- [サービス グラフについて, 3 ページ](#)

アプリケーションセントリックインフラストラクチャのレイヤ 4～7 サービスの導入について

従来の方法を使用する場合、サービスをネットワークに挿入すると、手間がかかって複雑な VLAN（レイヤ 2）または仮想ルーティングおよび転送（VRF）インスタンス（レイヤ 3）ステッチングを、ネットワーク要素およびサービスアプライアンスの間で実行する必要があります。この従来のモデルでは、アプリケーションに対する新規サービスを配備するのに数日から数週間かかります。サービスには柔軟性が少なく、操作エラーはより頻繁に発生し、トラブルシューティングはより困難です。アプリケーションが使用されなくなる場合、ファイアウォールルールなどのサービス デバイス設定の削除は困難になります。ロードに基づいたサービスのスケールアウト/スケールダウンを実行することもできません。

VLAN および仮想ルーティングおよび転送（VRF）ステッチングは従来のサービス挿入モデルによってサポートされますが、Application Policy Infrastructure Controller（APIC）はポリシー制御の中心点として機能する一方でサービス挿入を自動化できます。APIC ポリシーは、ネットワーク ファブリックとサービスアプライアンスの両方を管理します。APIC は、トラフィックがサービスを通して流れるように、ネットワークを自動的に設定できます。APIC は、アプリケーション要件に従ってサービスを自動的に設定することもでき、それにより組織はサービス挿入を自動化し、従来のサービス挿入の複雑な技術の管理に伴う課題を排除できます。

APIC を使用して次のタスクを実行し、レイヤ 4～7 サービスを展開します。

1 デバイス パッケージのインポート

[デバイス パッケージのインポート, \(25 ページ\)](#) を参照してください。

プロバイダーの管理者のみがデバイス パッケージをインポートできます。

- 2 テナントの設定
テナントの設定, (11 ページ) を参照してください。
- 3 デバイス クラスタおよび論理インターフェイスを登録します。
デバイス クラスタの設定, (28 ページ) を参照してください。
- 4 デバイス クラスタ (論理デバイス) パラメータの設定
- 5 レイヤ3 ネットワークの設定
レイヤ3 ネットワークの設定, (13 ページ) を参照してください。
- 6 ブリッジ ドメインの設定
ブリッジ ドメインの設定, (12 ページ) を参照してください。
- 7 アプリケーション プロファイルの設定
アプリケーション プロファイルの設定, (14 ページ) を参照してください。
- 8 物理ドメインまたは VMM ドメインの設定
物理ドメインの設定, (8 ページ) または VMM ドメインの設定, (9 ページ) を参照してください。
VMM ドメインの場合：
 - a VMM ドメイン クレデンシャルの設定
VMM クレデンシャルの設定, (10 ページ) を参照してください。
 - b vCenter/vShield コントローラ プロファイルの設定
vCenter/vShield コントローラ プロファイルの設定, (10 ページ) を参照してください。
- 9 VLAN プールの設定
VLAN プールの設定, (7 ページ) を参照してください。
 - a カプセル化ブロック範囲の設定
カプセル化ブロック範囲の設定, (7 ページ) を参照してください。
- 10 コントラクトの設定
コントラクトの設定, (18 ページ) を参照してください。
- 11 管理エンドポイント グループ (EPG) の設定
管理エンドポイント グループの設定, (19 ページ) を参照してください。
- 12 具象デバイスおよび具象インターフェイスの登録
具象デバイスの設定, (30 ページ) を参照してください。
- 13 具象デバイス パラメータの設定
- 14 サービス グラフの設定
サービス グラフの設定, (51 ページ) を参照してください。
- 15 アプリケーション プロファイルからデフォルトのサービス グラフ パラメータを選択
- 16 必要に応じてサービス グラフ パラメータを設定
- 17 サービス グラフをコントラクトおよびエンドポイント グループ (EPG) にアタッチ
- 18 追加のコンフィギュレーション パラメータの設定

設定パラメータ, (55 ページ) を参照してください。

サービス グラフについて

Cisco Application Centric Infrastructure (ACI) はアプリケーションの重要部分としてサービスを見なします。必要なサービスは、Cisco Application Policy Infrastructure Controller (APIC) からの ACI ファブリックでインスタンス化されたサービス グラフとして処理されます。ユーザは、アプリケーションに対してサービスを定義し、サービス グラフはアプリケーションが必要とする一連のネットワークまたはサービス機能を識別します。一度 APIC でグラフが設定されると、APIC はサービス グラフで指定されるサービス機能要件に基づいてサービスを自動的に設定します。APIC はまた、サービス グラフで指定されるサービス機能のニーズに応じてネットワークを自動的に設定しますが、これによってサービス デバイスでの変更は要求されません。

ACI

ACIを使用すると、特定のタイプのファイアウォールとその後に続く特定のモデルおよびバージョンのロードバランサなどのメタデバイスのシーケンスを定義できます。これは抽象グラフと呼ばれます。抽象グラフがコントラクトによって参照されると、抽象グラフはファブリック内に存在するファイアウォールやロードバランサなどの具象デバイスにマッピングすることでインスタンス化されます。マッピングは「コンテキスト」の概念で発生します。「デバイスコンテキスト」は、ACIがどのファイアウォールおよびどのロードバランサを抽象グラフにマップできるかの識別を可能にするマッピング設定です。もう1つの重要な概念は、具象デバイスのクラスタを表す「論理デバイス」、つまりデバイスクラスタです。抽象グラフのレンダリングは、コントラクトによって定義されるパスに挿入可能な適切な論理デバイスの識別に基づいています。



第 2 章

前提条件

- [テナントについて, 5 ページ](#)
- [セキュリティドメインについて, 6 ページ](#)
- [レイヤ3 ネットワークについて, 6 ページ](#)
- [ブリッジドメインについて, 6 ページ](#)
- [アプリケーションプロファイルについて, 6 ページ](#)
- [コントラクトについて, 6 ページ](#)
- [VLAN プールの設定, 7 ページ](#)
- [物理ドメインの設定, 8 ページ](#)
- [VMM ドメインの設定, 9 ページ](#)
- [テナントの設定, 11 ページ](#)
- [ブリッジドメインの設定, 12 ページ](#)
- [レイヤ3 ネットワークの設定, 13 ページ](#)
- [アプリケーションプロファイルの設定, 14 ページ](#)
- [コントラクトの設定, 18 ページ](#)
- [管理エンドポイントグループの設定, 19 ページ](#)

テナントについて

テナントは、許可されたユーザがテナント管理やネットワーキング管理などの権限にアクセスできるようにするために、管理者がドメインベースのアクセス制御を実行することを可能にするポリシーのコンテナです。レイヤ4～7のサービスを展開する前に、テナントを設定する必要があります。

セキュリティドメインについて

セキュリティドメインは、どのユーザがどのテナントにアクセスできるかを判断できる概念です。たとえば、Tenant1、Tenant2、および Tenant3 を作成した場合、3つのセキュリティドメイン (securitydomain1、securitydomain2、および securitydomain3) を作成することができ、各テナントの管理者は対応するセキュリティドメインに関連付けられます。

レイヤ3ネットワークについて

レイヤ3 (L3) は、Open Systems Interconnection (OSI) の通信モデルのネットワーク層です。L3 ネットワーク設定は、トラフィック転送がファブリックの外部に対してどのように機能するかの設定を参照します。L3 は、他のノードのアドレスの検出、ルートの選択、QoS の選択、およびローカルホストドメインのトランスポート層への受信メッセージの転送に使用されます。L3 ネットワークは、テナントによって使用されるすべてのアプリケーションエンドポイントグループ (EPG) によって使用されます。

ブリッジドメインについて

ブリッジドメインはファブリック内のレイヤ2 (L2) フォワーディングコンストラクトを表します。1つ以上のエンドポイントグループ (EPG) を1つのブリッジドメインまたはサブネットと関連付けることができます。ブリッジドメインには1つまたは複数のサブネットを関連付けることができます。1つまたは複数のブリッジドメインの組み合わせによってテナントネットワークを形成します。

アプリケーションプロファイルについて

アプリケーションプロファイルは、ポリシー、サービスおよびエンドポイントグループ (EPG) 間の関係を定義します。各アプリケーションプロファイルには、コントラクトの規則に基づいて、同じアプリケーションプロファイル内の他の EPG および他のアプリケーションプロファイル内の EPG と通信できる1つ以上の EPG が含まれます。

コントラクトについて

コントラクトには、コントラクトを作成および消費するエンドポイントグループ (EPG) 間で適用されるすべてのフィルタが含まれます。コントラクトには、プロバイダーおよびコンシューマーと呼ばれる EPG が含まれます。コントラクトはプロバイダーとコンシューマーが通信できるプロトコルとポートを定義します。

VLAN プールの設定

VLAN プールは、VLAN ネームスペースとしても知られています。VLAN プールを設定できます。

ステップ 1 [CREATE VCENTER DOMAIN] ダイアログボックスで、[VLAN Pool] ドロップダウンリストから [Create VLAN Pool] を選択します。[CREATE VLAN POOL] ダイアログボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Name] フィールド	VLAN プールの名前。
[Description] フィールド	VLAN プールの説明。
[Allocation Mode] オプション ボタン	VLAN プールの割り当てモード。 [Dynamic Allocation] または [Static Allocation] を選択できます。 (注) デバイスが仮想であるときに VMM を統合する必要がある場合 [Dynamic Allocation] を選択します。
[Encap Blocks] セクション	カプセル化ブロック範囲は、パフォーマンス グラフに対する仮想アプライアンスを使用する間、どの VLAN を使用するか指定します。カプセル化ブロック範囲を作成するには、 カプセル化ブロック範囲の設定 (7 ページ) を参照してください。

ステップ 3 [SUBMIT] をクリックします。[CREATE VLAN POOL] ダイアログボックスが閉じ、VLAN プールが作成されます。

カプセル化ブロック範囲の設定

カプセル化ブロック範囲は、パフォーマンス グラフに対する仮想アプライアンスを使用する間に、どの VLAN を使用するかを指定します。カプセル化ブロック範囲を設定できます。

ステップ 1 [CREATE VLAN POOL] ダイアログボックスで、[Encap Blocks] セクションの [+] をクリックします。[CREATE RANGES] ダイアログボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[From] フィールド	カプセル化ブロック範囲の最小値。

名前	説明
[To] フィールド	カプセル化ブロック範囲の最大値。

ステップ 3 [OK] をクリックします。[CREATE RANGES] ダイアログ ボックスが閉じ、カプセル化ブロック範囲が作成されます。

物理ドメインの設定

物理ドメインは、特定の VLAN ネームスペースが使用される範囲を制御します。物理ドメインと関連付けられた VLAN のネームスペースは、仮想サーバからのポート グループのスタティック マッピングに使用できますが、非仮想サーバを対象としています。物理デバイスタイプの物理ドメインを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定, \(11 ページ\)](#) を参照してください。

ステップ 1 [Physical Domain] ドロップダウン リストで、[Create Physical Domain] を選択します。[CREATE PHYSICAL DOMAIN] ダイアログ ボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Name] フィールド	物理ドメイン プロファイルの名前。
[VLAN Pool] フィールド	物理ドメインの VLAN プール。VLAN プールは、この物理ドメインを使用しているサービス グラフに対して APIC によって割り当てられる VLAN の範囲またはプールを指定します。VLAN プールの作成方法については、 VLAN プールの設定, (7 ページ) を参照してください。

ステップ 3 [Submit] をクリックします。[CREATE PHYSICAL DOMAIN] ダイアログ ボックスが閉じ、物理ドメインが作成されます。

VMM ドメインの設定

Virtual Machine Manager (VMM) ドメインは、仮想サーバに対する特定の VLAN ネームスペースの使用範囲を定義します。Virtual Machine Manager (VMM) ドメインは、vCenter ドメインとも呼ばれます。VMM ドメインを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定, \(11 ページ\)](#) を参照してください。
- テナントにデバイスクラスタを設定します。 [デバイスクラスタの設定](#) を参照してください。

ステップ 1 [VMM Domain] ドロップダウンリストで、[Create vCenter Domain] を選択します。[CREATE VCENTER DOMAIN] ダイアログ ボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Name] フィールド	VMM ドメイン プロファイルの名前。
[Virtual Switch] オプション ボタン	仮想スイッチのモード。
[Associated Attachable Entity Profile] フィールド	VMM ドメインに関連付けられる、接続可能エンティティ プロファイル。接続可能エンティティ プロファイルは、VMM ドメインのファブリックへの接続に必要です。
[VLAN Pool] フィールド	VMM ドメインの VLAN プール。VLAN プールは、この VMM ドメインを使用しているサービス グループに対して APIC によって割り当てられる VLAN の範囲またはプールを指定します。VLAN の設定方法については、 VLAN プールの設定, (7 ページ) を参照してください。
[vCenter Credentials] セクション	VMM ドメインに接続するために使用するクレデンシャル。VMM クレデンシャルとしても知られる vCenter クレデンシャルを設定するには、 VMM クレデンシャルの設定, (10 ページ) を参照してください。
[vCenter/vShield] セクション	VMM ドメインで使用される vCenter/vShield コントローラ プロファイル。vCenter/vShield コントローラ プロファイルを設定するには、 vCenter/vShield コントローラ プロファイルの設定, (10 ページ) を参照してください。

ステップ 3 [OK] をクリックします。[CREATE VMM DOMAIN] ダイアログ ボックスが閉じ、VMM ドメインが作成されます。

VMM クレデンシャルの設定

VMM クレデンシャルは VMM ドメインへの接続に必要です。VMM クレデンシャルを設定できます。

ステップ 1 [CREATE VCENTER DOMAIN] ダイアログボックスで、[vCenter Credentials] セクションの [+] をクリックします。[CREATE VCENTER CREDENTIAL] ダイアログ ボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Profile Name] フィールド	VMM ドメインへのログインに使用するプロファイルの名前。
[Description] フィールド	ユーザ アカウント プロファイルの説明。
[Username] フィールド	クレデンシャルに使用するユーザ名。
[Password] フィールド	指定したユーザのパスワード。
[Confirm Password] フィールド	指定したユーザのパスワード確認。

ステップ 3 [OK] をクリックします。[CREATE VCENTER CREDENTIAL] ダイアログ ボックスが閉じ、VMM クレデンシャルが作成されます。

vCenter/vShield コントローラ プロファイルの設定

vCenter/vShield コントローラ プロファイルを設定できます。

ステップ 1 [CREATE VCENTER DOMAIN] ダイアログボックスで、[vCenter/vShield] セクションの [+] をクリックします。[CREATE VCENTER/VSHIELD CONTROLLER] ダイアログ ボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Type] オプション ボタン	コントローラのプロファイル タイプ。
[Name] フィールド	vCenter/vShield コントローラ プロファイルの名前。
[Address] フィールド	vCenter/vShield コントローラ プロファイルのホスト名または IP アドレス。
[Stats Collection] オプション ボタン	統計収集をイネーブルまたはディセーブルにします。

名前	説明
[Management EPG] フィールドおよびドロップダウンリスト	Virtual Machine Manager (VMM) コントローラ プロファイルで管理エンドポイント グループ (EPG) を選択します。
[Associated Credential] フィールドおよびドロップダウンリスト	vCenter/vShield コントローラ プロファイルで使用する VMM クレデンシャルを選択します。

ステップ 3 [OK] をクリックします。[CREATE VCENTER/VSHIELD CONTROLLER] ダイアログ ボックスが閉じ、vCenter/vShield コントローラ プロファイルが作成されます。

テナントの設定

テナントを設定できます。

ステップ 1 メニュー バーで、[TENANTS] タブをクリックします。[TENANT] ウィンドウが表示されます。

ステップ 2 サブメニュー バーで、[ADD TENANT] をクリックします。[CREATE TENANT] ダイアログ ボックスが表示され、[TENANT] ページが表示されます。

ステップ 3 次のフィールドに入力します。

名前	説明
[Name] フィールド	テナントの名前。
[Alias] フィールド	テナントのエイリアス。単一のテナントを示す場合、エイリアスはテナントの名前よりも簡単でわかりやすいものにすることができます。1つのテナントのみに特定のエイリアス名を割り当てることができ、2つ目のテナントに同じエイリアス名を割り当てることはできません。
[Description] フィールド	テナントの説明。
[Tags] フィールド	テナントに割り当てられた検索キーワードまたは用語。タグを使用すると、わかりやすい名前でも複数のオブジェクトをグループ化できます。複数のオブジェクトに同じタグ名を割り当て、1つのオブジェクトに1つ以上のタグ名を割り当てることができます。
[Monitoring Policy] フィールド	ポリシー名を監視するエンドポイント グループ (EPG)。

名前	説明
[Security Domains] セクション	テナントのセキュリティドメイン。レイヤ4～7サービスを配置するためにセキュリティドメインを選択する必要はありません。セキュリティドメインの作成については、『Cisco APIC Getting Started Guide』を参照してください。

ステップ4 [Next] をクリックします。[NETWORK] ページが表示され、テナントが作成されます。レイヤ3ネットワークを設定するには、[レイヤ3ネットワークの設定](#)、(13 ページ) を参照してください。

ブリッジドメインの設定

ブリッジドメインを設定できます。

はじめる前に

- レイヤ3 (L3) ネットワークを設定します。 [レイヤ3ネットワークの設定](#)、(13 ページ) を参照してください。

ステップ1 [CREATE TENANT] ダイアログボックスの[BRIDGE DOMAIN] ページで、次のフィールドを完成します。

名前	説明
[Name] フィールド	ブリッジドメインの名前。
[Description] フィールド	ブリッジドメインの説明。
[Forwarding] ドロップダウンリスト	[Optimize] または [Custom] のいずれかのブリッジドメインの転送メソッドを選択します。
[L2 Unknown Unicast] オプションボタン	不明な層デスティネーションの転送メソッド。これらのオプションボタンは、[Forwarding] ドロップダウンリストで [Custom] を選択した場合にのみ表示されます。
[ARP Flooding] チェックボックス	ARPフラッディングを有効にするには、このチェックボックスをオンにします。フラッディングが無効の場合、ユニキャストルーティングはターゲット IP アドレス上で実行されます。このチェックボックスは、デフォルトでオフになっています。このチェックボックスは、[Forwarding] ドロップダウンリストで [Custom] を選択した場合にのみ表示されます。

名前	説明
[Unicast Routing] チェックボックス	ユニキャストルーティングを有効にするには、このチェックボックスをオンにします。ユニキャストルーティングは事前定義済みの転送基準に基づく転送方式です (IP または MAC アドレス)。このチェックボックスは、デフォルトでオンになっています。このチェックボックスは、[Forwarding] ドロップダウンリストで [Custom] を選択した場合にのみ表示されます。
[IGMP Snoop Policy] ドロップダウンリスト	Internet Group Management Protocol (IGMP) スヌーピングポリシー。レイヤ4～7サービスを導入するためにIGMPスヌーピングポリシーを選択する必要はありません。
[Config BD MAC Address] チェックボックス	ブリッジドメインのMACアドレスを設定するには、このチェックボックスをオンにします。
[MAC Address] フィールド	ブリッジドメインのMACアドレス。このフィールドは [Config BD MAC Address] チェックボックスをオンにしている場合にのみ表示されます。
[Subnets] セクション	ブリッジドメインのサブネット。[+] をクリックし、フィールドに入力し、[UPDATE] をクリックしてサブネットを追加します。複数のサブネットを追加できます。
[DHCP Labels] セクション	ブリッジドメインのDHCPラベル。レイヤ4～7サービスを導入するためにDHCPラベルを設定する必要はありません。DHCPラベルの設定手順については、『Cisco APIC Getting Started Guide』を参照してください。

ステップ2 [OK] をクリックします。次の [NETWORK] ページが表示され、ブリッジドメインが作成されます。このページでは、レイヤ2 (L2) 外部キャッシュ、L3外部キャッシュ、追加のネットワーク、および追加のブリッジドメインを追加できます。

ステップ3 [Next] をクリックします。アプリケーションプロファイルを設定するために使用する [APPLICATION] ページが表示されます。アプリケーションプロファイルを設定するには、[アプリケーションプロファイルの設定](#)、(14 ページ) を参照してください。

レイヤ3ネットワークの設定

レイヤ3 (L3) ネットワークを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定](#), (11 ページ) を参照してください。

ステップ 1 [CREATE TENANT] ダイアログ ボックスの [NETWORK] ページで、[+] をクリックしてネットワークを追加します。 [CREATE NEW NETWORK] ダイアログ ボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Name] フィールド	ネットワークの名前。
[Description] フィールド	ネットワークの説明。
[BGP Timers] ドロップダウン リスト	ネットワークの Border Gateway Protocol (BGP) タイマーを選択します。 新しい BGP タイマー ポリシーを設定するには、 BGP タイマーポリシーの設定 を参照してください。
[OSPF Timers] ドロップダウン リスト	Open Shortest Path First (OSPF) タイマー ポリシー。 OSPF タイマーポリシーでは、Hello タイマーおよび Dead タイマー間隔を設定できます。 デフォルトのポリシーを選択するか、新しいポリシーを作成できます。 レイヤ 4～7 サービスを配置するために OSPF タイマーポリシーを選択する必要はありません。
[Monitoring Policy] ドロップダウン リスト	ネットワークのモニタリング ポリシーを選択します。

ステップ 3 [Next] をクリックします。 [BRIDGE DOMAIN] ページが表示され、L3 ネットワークが作成されます。 ブリッジ ドメインを設定するには、 [ブリッジ ドメインの設定](#), (12 ページ) を参照してください。

アプリケーション プロファイルの設定

アプリケーション プロファイルを設定できます。

はじめる前に

- ブリッジドメインを設定します。 [ブリッジドメインの設定](#)、(12 ページ) を参照してください。

- ステップ 1** メニューバーで、[TENANTS] タブをクリックします。[TENANT] ウィンドウが表示されます。
- ステップ 2** サブメニューバーで、アプリケーションプロファイルを設定するテナントのタブをクリックします。選択したテナントの [Tenant] ウィンドウが [Work] ペインに表示されます。
- ステップ 3** [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 4** [Application Profiles] をクリックします。[Application Profiles] ウィンドウが [Work] ペインに表示されます。
- ステップ 5** [ACTIONS] > [Create Application Profile] を選択します。[CREATE APPLICATION PROFILE] ダイアログボックスが表示されます。
- ステップ 6** 次のフィールドに入力します。

名前	説明
[Name] フィールド	アプリケーションプロファイルの名前。
[Alias] フィールド	アプリケーションプロファイルのエイリアス。単一のアプリケーションプロファイルを参照する場合、エイリアスはアプリケーションプロファイルの名前よりも簡単で説明的なものにすることができます。1つのアプリケーションプロファイルのみに特定のエイリアス名を割り当てることができ、2つ目のアプリケーションプロファイルに同じエイリアス名を割り当てることはシステムによって回避されます。
[Description] フィールド	アプリケーションプロファイルのプライオリティ。
[Tags] フィールド	アプリケーションプロファイルに割り当てられた検索キーワードまたは用語。タグを使用すると、わかりやすい名前でも複数のオブジェクトをグループ化できます。複数のオブジェクトに同じタグ名を割り当て、1つのオブジェクトに1つ以上のタグ名を割り当てることができます。
[Monitoring Policy] ドロップダウンリスト	ポリシー名を監視するエンドポイントグループ (EPG) を選択します。

- ステップ 7** [EPGs] セクションで、[+] をクリックします。[CREATE APPLICATION EPG] ダイアログボックスが表示されます。
- ステップ 8** 次のフィールドに入力します。

名前	説明
[Name] フィールド	アプリケーション EPG の名前。

名前	説明
[Alias] フィールド	アプリケーション EPG のエイリアス。単一のアプリケーション EPG を参照する場合、エイリアスはアプリケーション EPG の名前よりも簡単でわかりやすいものにすることができます。1 つのアプリケーション EPG のみに特定のエイリアス名を割り当てることができ、2 つ目のアプリケーション EPG に同じエイリアス名を割り当てることができません。
[Description] フィールド	アプリケーション EPG の説明。
[Tags] フィールド	アプリケーション EPG に割り当てられた検索キーワードまたは用語。タグを使用すると、わかりやすい名前でも複数のオブジェクトをグループ化できます。複数のオブジェクトに同じタグ名を割り当て、1 つのオブジェクトに 1 つ以上のタグ名を割り当てることができます。
[QoS class] ドロップダウン リスト	QoS のプライオリティ クラス ID を選択します。
[Custom QoS] ドロップダウン リスト	QoS のトラフィック プライオリティ クラス ID。カスタム クラスは、ユーザ設定可能な DiffServ コード ポイント (DSCP) 値です。レイヤ 4 ~ 7 サービスに配置するために QoS トラフィック プライオリティ クラス ID を選択する必要はありません。
[Bridge Domain] ドロップダウン リスト	アプリケーション EPG に関連付けられたブリッジ ドメインの名前を選択します。
[Monitoring Policy] ドロップダウン リスト	ポリシー名を監視するエンドポイント グループ (EPG) を選択します。
[Associated Domain Profiles (VMs or bare metals)] セクション	アプリケーション EPG に関連付けられたドメイン プロファイル。ドメイン プロファイルを追加するには、[+] をクリックします。複数のドメイン プロファイルを追加できます。
[Statically Link with Leaves/Paths] チェックボックス	アプリケーション EPG をリーフおよびパスとスタティックにリンクする場合はこのチェック ボックスをオンにします。

ステップ 9 [Statically Link with Leaves/Paths] をオンにした場合、[NEXT] をクリックします。[LEAVES/PATHS] ページが表示されます。

- a) [Leaves] セクションで、[+] をクリックしてリーフを追加します。
- b) 次のフィールドに入力します。

名前	説明
[Node] ドロップダウン リスト	リーフとして使用するノードを選択します。
[Encap] フィールド	カプセル化に使用する VLAN。有効な範囲は 1 ~ 4094 です。

名前	説明
[Deployment Immediacy] ドロップダウン リスト	このリーフアソシエーションを即時にまたは必要に応じて実行するかを選択します。
[Mode] ドロップダウン リスト	リーフのスタティックアソシエーションのモードを選択します。

- c) [UPDATE] をクリックします。リーフが追加されます。
- d) [Paths] セクションで、[+] をクリックしてパスを追加します。
- e) 次のフィールドに入力します。

名前	説明
[Path] ドロップダウン リスト	パスとして使用するノードを選択します。
[Encap] フィールド	カプセル化に使用する VLAN。有効な範囲は 1 ~ 4094 です。
[Deployment Immediacy] ドロップダウン リスト	このパスアソシエーションを即時にまたは必要に応じて実行するかを選択します。
[Mode] ドロップダウン リスト	パスのスタティックアソシエーションのモードを選択します。

- f) [UPDATE] をクリックします。パスが追加されます。

ステップ 10 [OK] をクリックします。[CREATE APPLICATION EPG] ダイアログ ボックスが閉じます。

ステップ 11 [Provided Contracts] セクションで、[+] をクリックして提供されるコントラクトを追加します。[ADD PROVIDED CONTRACT] ダイアログ ボックスが表示されます。

ステップ 12 次のフィールドに入力します。

名前	説明
[Contract Type] ドロップダウン リスト	コントラクトのタイプを選択します。
[Name] ドロップダウン リスト	コントラクトの名前を選択します。既存のコントラクトを示す [default] を選択するか、または [Create New Contract] を選択できます。

ステップ 13 [OK] をクリックします。[ADD PROVIDED CONTRACT] ダイアログ ボックスが閉じます。

ステップ 14 [Consumed Contracts] セクションで、[+] をクリックして提供されるコントラクトを追加します。[ADD CONSUMED CONTRACT] ダイアログ ボックスが表示されます。

ステップ 15 次のフィールドに入力します。

名前	説明
[Contract Type] ドロップダウン リスト	コントラクトのタイプを選択します。
[Name] ドロップダウン リスト	コントラクトの名前を選択します。既存のコントラクトを示す [default] を選択するか、または [Create New Contract] を選択できます。

ステップ 16 [OK] をクリックします。[ADD CONSUMED CONTRACT] ダイアログ ボックスが閉じます。

ステップ 17 ネイバーが存在する場合、[Neighbors] セクションで [+] をクリックしてネイバーを追加します。[ADD NEIGHBOR] ダイアログ ボックスが表示されます。

ステップ 18 [SUBMIT] をクリックします。[CREATE APPLICATION PROFILE] ダイアログ ボックスが閉じ、アプリケーションプロファイルが設定されます。

コントラクトの設定

コントラクトを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定](#)、(11 ページ) を参照してください。
- テナントにデバイスクラスタを設定します。 [デバイスクラスタの設定](#) を参照してください。

ステップ 1 メニュー バーで、[TENANTS] タブをクリックします。[Tenant] ウィンドウが表示されます。

ステップ 2 [Navigation] ペインで、テナントのツリーおよびコントラクトを設定したいテナントの下のアプリケーションプロファイルのツリーを展開します。

ステップ 3 [Contracts] を選択します。

ステップ 4 [Actions] > [Create Contract] を選択します。[CREATE CONTRACT] ダイアログ ボックスが表示されます。

ステップ 5 次のフィールドに入力します。

名前	説明
[Name] フィールド	コントラクトの名前。
[Scope] フィールド	コントラクトの範囲。
[Priority] フィールド	コントラクトのプライオリティ。

名前	説明
[Description] フィールド	コントラクトの説明。

ステップ 6 [Subjects] セクションで、[+] をクリックしてコントラクトサブジェクトを追加します。[CREATE CONTRACT SUBJECT] ダイアログ ボックスが表示されます。

ステップ 7 次のフィールドに入力します。

名前	説明
[Name] フィールド	コントラクトのサブジェクトの名前。
[Description] フィールド	コントラクトのサブジェクトの説明。

ステップ 8 [Filter Chain] セクションで、[+] をクリックしてフィルタを追加します。

ステップ 9 フィルタが適用されるテナントを選択し、フィルタと使用するサービス グラフを選択します。コントラクトに一致するトラフィックは、サービス グラフにリダイレクトされます。

ステップ 10 [UPDATE] をクリックします。フィルタが作成されます。

ステップ 11 [OK] をクリックします。[CREATE CONTRACT SUBJECT] ダイアログ ボックスが閉じ、コントラクトのサブジェクトが作成されます。

ステップ 12 [SUBMIT] をクリックします。[CREATE CONTRACT] ダイアログ ボックスが閉じ、コントラクトが作成されます。

管理エンドポイントグループの設定

デバイス クラスタで使用する新しい管理エンドポイントグループ (EPG) を設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定](#)、(11 ページ) を参照してください。
- テナントにデバイス クラスタを設定します。 [デバイス クラスタの設定](#) を参照してください。

ステップ 1 [Create Device Cluster] ダイアログ ボックスで、[EPG] ドロップダウン リストから [Create Management EPG] を選択します。[Create Management EPG] ダイアログ ボックスが表示されます。

ステップ 2 次のフィールドに入力します。

名前	説明
[Application Profile] フィールド	管理 EPG で使用するアプリケーションプロファイル。プロファイルがまだ存在しない場合、[Create Application Profile] を選択して作成できます。
[Name] フィールド	アプリケーションプロファイルの名前。
[Alias] フィールド	管理 EPG のエイリアス。単一の管理 EPG を示す場合、エイリアスは管理 EPG の名前よりも単純でわかりやすいものにすることができます。1 つの管理 EPG のみに特定のエイリアス名を割り当てることができ、2 つ目の管理 EPG に同じエイリアス名を割り当ててはできません。
[Description] フィールド	管理 EPG の説明。
[QoS class] ドロップダウンリスト	管理 EPG の QoS クラスを選択します。デバイスクラスが物理アプリケーションを使用する場合、[Unspecified] を選択します。
[Bridge Domain] ドロップダウンリスト	管理 EPG で使用するブリッジドメインのノードを選択します。

ステップ 3 [Domains (VMM, Physical, or External) Associated to Interfaces] セクションで、[+] をクリックしてドメインを追加します。

ステップ 4 次のフィールドに入力します。

名前	説明
[Domain Profile] ドロップダウンリスト	この管理 EPG で使用するドメインプロファイルを選択します。
[Deployment Immediacy] ドロップダウンリスト	ドメインをすぐに展開するかまたは必要に応じて展開するかを選択します。

名前	説明
[Resolution Immediacy] ドロップ ダウンリスト	<p>ポリシーがリーフ ノードにどのように適用されるかを選択します。</p> <ul style="list-style-type: none"> • [Immediate] : VLAN バインディング、NVGRE バインディング、VXLAN バインディング、コントラクト、およびフィルタを含めたすべてのポリシーが Hypervisor の物理 NIC のアタッチ時にリーフ ノードへプッシュされます。Hypervisor 対リーフ ノードのアタッチメントの解決に使用されるリンク層検出プロトコル (LLDP) または OpFlex。 • [On-Demand] : 物理 NIC をアタッチし、仮想 NIC をポート グループ (EPG) に関連付けたときにのみ、ポリシーがリーフ ノードにプッシュされます。

ステップ 5 [Reserved IP addresses for APICs] セクションで、[+] をクリックして IP アドレス プールを作成します。
[CREATE IP ADDRESS POOL] ダイアログボックスが表示されます。

ステップ 6 次のフィールドに入力します。

名前	説明
[Name] フィールド	IP アドレス プールの名前。
[Gateway Address] フィールド	ゲートウェイの IP アドレスとサブネット マスク

ステップ 7 [Address Ranges] セクションで、[+] をクリックします。

ステップ 8 次のフィールドに入力します。

名前	説明
[From] フィールド	IP アドレス範囲の開始点。
[To] フィールド	IP アドレス範囲の終了点。

ステップ 9 [UPDATE] をクリックします。アドレス範囲が [Address Ranges] セクションに追加されます。

ステップ 10 [OK] をクリックします。[CREATE IP ADDRESS POOL] ダイアログボックスが閉じます。

ステップ 11 [Submit] をクリックします。[Create Management EPG] ダイアログボックスが閉じ、管理 EPG が作成されます。



第 3 章

デバイス パッケージのインポート

- [デバイス パッケージについて](#), 23 ページ
- [デバイス パッケージのインポート](#), 25 ページ

デバイス パッケージについて

APIC (Application Policy Infrastructure Controller) は、サービス デバイスの設定およびモニタリングにデバイス パッケージを必要とします。デバイス パッケージは、サービス デバイスのクラスを管理して、デバイスが何であるか、およびデバイスで実行できることを APIC が認識できるように APIC にデバイスの情報を提供します。デバイス パッケージは次の項目を含む zip ファイルです。

デバイス仕様	次のプロパティを定義する XML ファイル： <ul style="list-style-type: none">• デバイス プロパティ：<ul style="list-style-type: none">◦ [Model] : デバイスのモデル。◦ [Vendor] : デバイスのベンダー。◦ [Version] : デバイスのソフトウェア バージョン。• ロードバランシング、コンテンツ切り替え、および SSL 終端などの、デバイスによって提供される機能。• 各機能のインターフェイスおよびネットワーク接続情報。• デバイス設定パラメータ。• 各機能の設定パラメータ。
デバイス スクリプト	APIC とデバイス間の統合を実行する Python スクリプト。APIC イベントは、デバイス スクリプトで定義した機能呼び出しにマッピングされます。

機能プロファイル	ベンダーによって指定されたデフォルト値を持つパラメータのプロファイル。これらのデフォルト値を使用するように機能を設定できます。
デバイスレベル設定パラメータ	デバイス レベルでデバイスに必要なパラメータを指定するコンフィギュレーションファイル。設定は、デバイスを使用している1つ以上のグラフで共有できます。

デバイス パッケージを作成できます。または、デバイス ベンダーか Cisco によって提供されるものを使用できます。

デバイス パッケージのインストール例

デバイス パッケージを作成する場合、システムにインストールする必要があります。

次に示すのは、デバイス パッケージのインストール例です。

<pre>\$ scp AcmeADC.zip admin@10.10.10.10:</pre>	システムにパッケージをアップロードします。
<pre>\$ ssh admin@10.10.10.10</pre>	プロバイダー管理者としてログインします。
<pre>admin@apic:~> services install AcmeADC.zip</pre>	デバイスパッケージをインストールします。
<pre>admin@apic:~> rm AcmeADC.zip</pre>	(任意) アップロードされたファイルをクリーンアップします。 (注) これによってパッケージはアンインストールされません。

REST でのデバイス パッケージのインストール時の注意事項

- デバイス パッケージは HTTP または HTTPS POST を使用してインストールできます。
- HTTP が APIC で有効になっている場合、POST の URL は「<http://10.10.10.10/ppi/node/mo/.xml>」です。
- HTTPS が APIC で有効になっている場合、POST の URL は「<https://10.10.10.10/ppi/node/mo/.xml>」です。
- メッセージには有効なセッション Cookie が必要です。

Cookie の取得については『Cisco APIC REST API User Guide』を参照してください。

- POST の本文にはアップロードされるデバイス パッケージを含める必要があります。POST で許可されるのは1つのパッケージだけです。

デバイス パッケージのインポート

どのようなデバイスがあり、何をデバイスで実行できるかを APIC が認識できるように、Application Policy Infrastructure Controller (APIC) にデバイス パッケージをインポートできます。

-
- ステップ 1** プロバイダー管理者としてログインします。
 - ステップ 2** メニューバーで、[L4-L7 Services] タブをクリックします。[INVENTORY] サブメニュー タブの [Quick Start] ウィンドウが表示されます。
 - ステップ 3** サブメニューバーで、[PACKAGES] タブをクリックします。[PACKAGES] サブメニュー タブの [Quick Start] ウィンドウが表示されます。
 - ステップ 4** [Navigation] ペインで、[Device Types] をクリックします。[Device Types] ウィンドウが表示されます。
 - ステップ 5** [Actions] > [Import Device Package] を選択します。[IMPORT DEVICE PACKAGE] ダイアログ ボックスが表示されます。
 - ステップ 6** [File Name] フィールドで、ベンダーによって提供されたか、または以前に作成したデバイス パッケージを指定します。デバイス パッケージの作成については、『Cisco APIC Layer 4 to Layer 7 Device Package Development Guide』を参照してください。例として、デバイス パッケージを含むファイル AcmeADC.zip が選択およびアップロードされています。
 - ステップ 7** [Submit] をクリックします。[IMPORT DEVICE PACKAGE] ダイアログ ボックスが閉じます。パッケージが正常にアップロードされたかどうかの確認メッセージが表示されます。
 - ステップ 8** APIC で [Device Types] ウィンドウをリフレッシュします。新しいデバイスがデバイス タイプのリスト内に表示されます。
 - ステップ 9** (任意) [Navigation] ペインで、[Device Types] を展開してデバイス パッケージの機能パラメータを確認し、機能のいずれかを選択します。
-



第 4 章

デバイス クラスタの設定（論理デバイス）

- [デバイス クラスタについて（論理デバイス）](#) , 27 ページ
- [具象デバイスについて](#) , 27 ページ
- [デバイス クラスタの設定](#) , 28 ページ
- [具象デバイスの設定](#) , 30 ページ
- [仮想サービス デバイスに対するデバイス クラスタの設定](#) , 32 ページ

デバイス クラスタについて（論理デバイス）

デバイス クラスタ（別名論理デバイス）は、単一のデバイスとして機能する 1 つ以上の具象デバイスです。デバイス クラスタには、デバイス クラスタのインターフェイス情報を説明する論理インターフェイスがあります。サービス グラフのレンダリング中に、機能ノード コネクタは論理インターフェイスに関連付けられます。Application Policy Infrastructure Controller (APIC) は、サービス グラフのインスタンス化およびレンダリング中に機能ノード コネクタにネットワーク リソース (VLAN または Virtual Extensible Local Area Network (VXLAN)) を割り当て、論理インターフェイスにネットワーク リソースをプログラミングします。

サービス グラフは、管理者が定義するデバイスのクラスタ選択ポリシー（論理デバイス コンテキストと呼ばれます）に基づく特定のデバイス クラスタを使用します。

管理者は、アクティブ/スタンバイ モードで最大 2 つの具象デバイス クラスタをセットアップできます。

具象デバイスについて

具象デバイスには、具象インターフェイスがあります。具象デバイスが論理デバイス クラスタに追加されると、具象インターフェイスは論理インターフェイスにマッピングされます。サービス グラフのインスタンス化時に、VLAN および VXLAN は、論理インターフェイスとのアソシエーションに基づいた具象インターフェイス上でプログラミングされます。

デバイス クラスタの設定

デバイス クラスタを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定](#)、(11 ページ) を参照してください。

- ステップ 1** メニュー バーで、[TENANTS] タブをクリックします。 [Tenant] ウィンドウが表示されます。
- ステップ 2** [Navigation] ペインで、[Tenant] ブランチを展開し、[L4-L7 Services] ブランチを展開して、[Device Clusters] をクリックします。 [Device Clusters] ウィンドウが表示されます。
- ステップ 3** [Actions] > [Create Device Cluster] を選択します。 [CREATE DEVICE CLUSTER] ダイアログボックスが表示され、[CLUSTER] ページが表示されます。
- ステップ 4** 次のフィールドに入力します。

名前	説明
[Name] フィールド	デバイス クラスタの名前。
[Device Package] フィールド	アップロードしたデバイス パッケージ。
[Context Aware] オプション ボタン	テナントにデバイスを割り当てることができます。 [Single] は、プロバイダー ネットワーク上でホストされる特定のタイプの複数のテナント全体でデバイス クラスタを共有できず、デバイス クラスタを特定のユーザの特定のテナントに渡す必要があることを示しています。 [Multiple] は、デバイス クラスタをこのプロバイダー ネットワーク上でホストする特定のタイプの複数のテナント全体でデバイス クラスタを共有できることを示しています。たとえば、同じデバイスを共有する2つのホスティング会社が存在する可能性があります。テナントの割り当ては、パッケージがバインドされているエンドポイント グループ (EPG) に暗黙的に基づいています。クラスタを作成した場合、デバイスが管理されるネットワークを判断する管理 EPG を指定する必要があります。
[Function Type] オプション ボタン	[GoThrough] デバイスはトランスペアレントなデバイスです。パケットはデバイスにアドレス指定されずに通過し、エンドポイントはデバイスを認識しません。[Goto] デバイスには、パッケージによって特定の宛先があります。この例では、デバイス パッケージは[GoTo] 機能のみをサポートするため、オプション ボタンはグレーアウトされて [GoTo] が自動的に選択されています。デバイス パッケージが両方の機能タイプをサポートしている場合、オプション ボタンのいずれかを選択して機能のタイプおよびその設定方法を示すことができます。
[Device Type] オプション ボタン	デバイス クラスタに [PHYSICAL] アプライアンスまたは [VIRTUAL] アプライアンスがあるかを指定します。

名前	説明
[Physical Domain] ドロップダウンリスト	このデバイス クラスタを使用するグラフに対するネットワーク リソースを割り当てる場合にこのドメインを選択します。物理ドメインは物理デバイス タイプの場合にのみ指定します。既存の物理ドメインを選択することも、新規作成することもできます。新しい物理ドメインを設定するには、 物理ドメインの設定 、(8 ページ) を参照してください。
[VMM Domain] ドロップダウンリスト	vCenter ドメインとしても知られる、Virtual Machine Manager (VMM) ドメインを選択します。VMM ドメインは仮想デバイス タイプの場合にのみ指定します。既存のVMM ドメインを選択することも、新規作成することもできます。新しいVMM ドメインを設定するには、 VMM ドメインの設定 、(9 ページ) を参照してください。
[EPG] ドロップダウンリスト	どの管理ネットワークが管理対象デバイスにつながるかを示す場合に管理 EPG を選択します。デバイスがインバンドで管理される場合、管理 EPG を選択する必要はありません。新しい管理 EPG を設定するには、 管理エンドポイント グループの設定 、(19 ページ) を参照してください。 管理が外部ネットワークからアウトオブバンドで実行される場合、管理 EPG を選択する必要はありません。
[Virtual IP Address] フィールド	デバイス クラスタ内の具象デバイスに対する管理インターフェイスの仮想 IP アドレス。
[Port] フィールド	デバイス クラスタ内の具象デバイスに対する管理インターフェイスのポート番号。
[Username] フィールド	デバイス クラスタにログインするためのユーザ名。
[Password] フィールド	デバイス クラスタにログインするためのパスワード。
[Confirm Password] フィールド	デバイス クラスタにログインするためのパスワード確認。

ステップ 5 [Logical Interfaces] セクションの [+] をクリックして論理インターフェイスを追加します。デバイス クラスタがグラフによって使用される場合、グラフのコネクタが論理インターフェイスにマッピングされます。

ステップ 6 次のフィールドに入力します。

名前	説明
[Name] フィールド	論理インターフェイスの名前。

名前	説明
[Type] フィールドおよびドロップダウンリスト	論理インタフェースのタイプ。デバイスパッケージで指定されたタイプを選択または入力する必要があります。

- ステップ 7** [Next] をクリックします。[DEVICES] ページが表示されます。
- ステップ 8** 具象デバイスおよび具象インターフェイスを追加します。 [具象デバイスの設定, \(30 ページ\)](#) を参照してください。
- ステップ 9** 具象インターフェイスを追加したら、[Next] をクリックします。[PARAMETERS] ページが表示されます。
- ステップ 10** (任意) コンフィギュレーションパラメータを追加します。コンフィギュレーションパラメータは具象デバイスを対象としており、初期化時にワнтаイム設定中に使用されます。
- ステップ 11** [OK] をクリックします。[CREATE CONCRETE DEVICE] ダイアログ ボックスが閉じ、[CREATE DEVICE CLUSTER] ダイアログ ボックスの [DEVICES] ページが表示されます。
- ステップ 12** [Finish] をクリックして、デバイスを作成します。[CREATE DEVICE CLUSTER] ダイアログ ボックスが閉じ、[Tenant] ページが表示されます。
- ステップ 13** サブメニューバーで、[common] タブをクリックし、ページを更新してデバイスを表示します。デバイスがダウンしていたり、誤って設定されている場合は表示されません。設定が誤っている場合は、APIC がデバイスが存在するかどうか確認できないか、デバイスは存在しても誤った種類のものであるか、またはデバイスがデバイスパッケージによって検証されないバージョンを実行している可能性があります。この場合、デバイス検証は失敗し、デバイス登録エラーの理由を確認することができます。

具象デバイスの設定

具象デバイスを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定, \(11 ページ\)](#) を参照してください。
- テナントにデバイス クラスタを設定します。 [デバイス クラスタの設定, \(28 ページ\)](#) を参照してください。

- ステップ 1** [CREATE DEVICE CLUSTER] ダイアログ ボックスの [DEVICES] ページで、[Create L4-L7 Devices] セクションの [+] をクリックして具象デバイスをデバイス クラスタに追加します。[CREATE CONCRETE DEVICE] ダイアログ ボックスが表示されます。
- ステップ 2** 次のフィールドに入力します。

名前	説明
[Name] フィールド	具象デバイスの名前。

名前	説明
[VM名] フィールド	仮想デバイスタイプのみが対象です。デバイスがホストされる仮想マシンの名前。
[vCenter Name] フィールド	仮想デバイスタイプのみが対象です。デバイスがホストされるVMMドメインの名前。
[IP Address] フィールド	デバイスクラスタ内の具象アドレスに対する管理インターフェイスホストアドレス。
[Port] フィールド	デバイスクラスタ内の具象アドレスに対する管理インターフェイスポート番号。
[Username] フィールド	デバイスクラスタにログインするためのユーザ名。
[Password] フィールド	デバイスクラスタにログインするためのパスワード。
[Confirm Password] フィールド	デバイスクラスタにログインするためのパスワード確認。

ステップ 3 具象デバイス上のインターフェイスである具象デバイスを追加するには、[+] をクリックし、次のフィールドに記入します。

名前	説明
[Name] フィールド	[name] フィールドは、具象デバイスのインターフェイスを識別します。たとえば、1/0、1.1 または Gig 0/0 などです。 (注) 特定のデバイスパッケージでは、インターフェイス名内の「/」を「_」に置き換える必要があります。つまり、デバイス上のインターフェイス 0/1 では APIC で 0_1 と表されます。デバイスがネーミングプロパティで「/」をサポートするか、または「/」を「_」に置き換える必要があるかどうかについては、デバイスパッケージのマニュアルを参照してください。
[Path] フィールド	物理デバイスを対象とします。具象インターフェイスをファブリックに接続する方法を指定します。たとえば、具象インターフェイスがアタッチされるリーフノード/スロット/ポートです。
[vNIC] フィールド	具象デバイスの対応するインターフェイスを識別するために vCenter 上で割り当てられたネットワークアダプタ名。通常、vNIC は vCenter 上で Network adapter x とラベルが付けられます。ここで x は、1, 2, 3 ... を表します。 (注) デバイス上のインターフェイス MAC をチェックし、[MAC] フィールドを照合することで vCenter 上の対応する vNIC を識別できます。
[Logical Interface] フィールド	具象インターフェイスがマップされている論理インターフェイスのタイプ。

入力する情報は、具象インターフェイスがファブリックに接続され、具象インターフェイスが論理インターフェイスに接続される方法について指定します。

- ステップ 4** 具象インターフェイスの情報を入力したら、[Update] をクリックしてインターフェイスを具象デバイスに追加します。
- ステップ 5** [Next] をクリックします。[PARAMETERS] ページが表示されます。
- ステップ 6** (任意) コンフィギュレーションパラメータを追加します。コンフィギュレーションパラメータは具象デバイスを対象としており、初期化時にワンタイム設定中に使用されます。
- ステップ 7** [OK] をクリックします。[CREATE CONCRETE DEVICE] ダイアログ ボックスが閉じ、[CREATE DEVICE CLUSTER] ダイアログ ボックスの [DEVICES] ページが表示され、具象デバイスが設定されます。

仮想サービスデバイスに対するデバイス クラスタの設定

仮想サービス デバイスに対してデバイス クラスタを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定, \(11 ページ\)](#) を参照してください。

- ステップ 1** [Tenant common] ウィンドウの [Navigation] ペインから [Tenant common] を展開し、[L4-L7 Services] ブランチを展開し、[Device Clusters] をクリックします。[Device Clusters] ウィンドウが表示されます。
- (注) デバイスのクラスタは、[User Tenant] スペースで作成することもできます。
- ステップ 2** [Actions] > [Create Device Cluster] を選択します。[CREATE DEVICE CLUSTER] ダイアログ ボックスが表示されます。
- ステップ 3** [Device Type] では、[VIRTUAL] を選択します。
- ステップ 4** [VMM Domain] に対して、仮想論理デバイス クラスタに Virtual Machine Manager (VMM) ドメインを関連付ける必要があります。この VMM ドメインは、IP アドレス、ポート、および管理者のクレデンシャルなどの vCenter を接続する方法に関する情報を持つ必要があります。また、サービス グラフに対して VLAN を割り当てるのに使用される VLAN プールを指定する必要があります。
- ステップ 5** 必要なデバイス クラスタ情報をすべて入力します。デバイス クラスタ情報の詳細については、[デバイス クラスタの設定, \(28 ページ\)](#) を参照してください。具象デバイスを追加するまでこの手順を続行します。
- ステップ 6** 具象デバイスを追加する場合、各具象デバイスに対して、vCenter 名、仮想マシン名、vNIC 名、および各インターフェイスに対して MAC アドレスを提供する必要があります。具象デバイスの追加の詳細については、[具象デバイスの設定, \(30 ページ\)](#) を参照してください。Application Policy Infrastructure Controller

(APIC) は、サービス アプライアンスに対して適切なポート グループを作成し、インターフェイスを正しいポート グループに配置します。NIC を配置する必要はありません。

ステップ 7 デバイスのクラスタを設定するための手順に進みます。 [デバイス クラスタの設定, \(28 ページ\)](#) を参照してください。



第 5 章

デバイス クラスタへの接続の設定

- [デバイス クラスタへの接続の設定について, 35 ページ](#)
- [テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の設定, 36 ページ](#)
- [テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の REST API による設定, 38 ページ](#)
- [管理テナント VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の設定, 39 ページ](#)
- [管理テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の REST API による設定, 39 ページ](#)

デバイス クラスタへの接続の設定について

APIC を設定するには、サービス アプライアンスに接続する必要があります。このためには APIC とサービス アプライアンス間の接続性が必要です。APIC はサービス アプライアンスに設定された管理 IP を使用してサービス アプライアンスの管理インターフェイスに接続します。

管理インターフェイスを設定し、ACI ファブリックに接続する前に管理 IP アドレスをサービス アプライアンスに割り当てる必要があります。その後、APIC とサービス アプライアンス間のインバンド接続を設定し、APIC がサービス アプライアンスの管理 IP に接続できるようにします。

APIC は、サービス アプライアンスの管理 IP に接続するために次の 2 通りの方法をサポートします。

- テナントの VRF を使用したデバイス クラスタ内でのデバイスのインバンド接続
- 管理テナント VRF を使用したデバイス クラスタ内でのデバイスのインバンド接続

テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続について

このシナリオでは、テナント管理者はサービス アプライアンス管理インターフェイスと APIC に割り当てられた IP アドレス（サービス アプライアンスの管理 IP）を所有し、管理します。このシナリオでは、サービス アプライアンスの管理 IP はテナントの VRF 内にあり、APIC はテナント VRF 内にあるソース IP を使用してサービス アプライアンスの管理 IP に接続します。

管理テナント VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続について

APIC は事前設定されたテナント **tn-mgmt** を提供します。これには事前設定された VRF が付属します。サービス アプライアンスに割り当てられた管理 IP アドレスは、**tn-mgmt** が所有および管理することができます。このシナリオでは、サービス アプライアンスの管理 IP は **tn-mgmt** VRF 内にあり、APIC は **tn-mgmt** の VRF 内にあるソース IP を使用してサービス アプライアンスの管理 IP に接続します。

テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の設定

はじめる前に

- デバイス クラスタを設定します。
[デバイス クラスタの設定, \(28 ページ\)](#) を参照してください。
- ブリッジ ドメインを設定します。
[ブリッジ ドメインの設定, \(12 ページ\)](#) を参照してください。
- エンドポイント グループを設定します。 (EPG)
[管理エンドポイント グループの設定, \(19 ページ\)](#) を参照してください。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[CREATE TENANT] ダイアログ ボックスの [BRIDGE DOMAIN] ページで、次のフィールドを完成します。	

	コマンドまたはアクション		目的
	オプション	説明	
	ARP フラッディング	[ARP Flood] チェックボックスをオフにして、[ARP Flood] を無効にします。	
	Subnets	[+] をクリックしてフィールドを完了し、[UPDATE] をクリックしてサブネットを追加します	
ステップ 2	[OK] をクリックします。次の [~NETWORK~] ページが表示され、ブリッジドメインが更新されます。		
ステップ 3	[Create Device Cluster] で、[EPG] ドロップダウンリストから [Create Management EPG] を選択します。[Create Management EPG] ダイアログ ボックスが表示されます。		
ステップ 4	次のフィールドに入力します。		
	オプション	説明	
	[Bridge Domain] ドロップダウンリスト	管理 EPG で使用するブリッジドメインのノードを選択します。	
ステップ 5	割り当てられたサブネットからコントローラ管理 IP アドレスを割り当てます。		
ステップ 6	メニュー バーで、[TENANTS] タブをクリックします。[TENANT] ウィンドウが表示されます。		
ステップ 7	サブメニュー バーで、[ADD TENANT] をクリックします。[CREATE TENANT] ダイアログ ボックスが表示され、[TENANT] ページが表示されます。		
ステップ 8	次のフィールドに入力します。		
	オプション	説明	
	[Controller Management Policies] セクション	[+] をクリックして異なるコントローラ管理ポリシーで各コンテキストを提供するためのフィールドに入力します	
ステップ 9	コントローラ管理ポリシーを EPG にポイントします。		
ステップ 10	EPG をデバイス クラスタにポイントします。		

テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の REST API による設定

以下は、テナント VRF を使用したデバイス クラスタ内のデバイスのインバンド接続を設定するために REST API を使用する例を示しています。

- 1 管理に使用する EPG を定義します。



(注) 適切なセレクタ設定を使用してドメイン マッピングに対してポートを開きます。

次の例では、EPG の「サービス」がテナント VRF デバイス管理に使用されるサービス デバイス/VM サブネットの管理に使用されます (3.3.3.0/24)。

```
<polUni>
  <fvTenant name="coke">
    <fvCtx name="mgmt_ctx1"/>
    <vnsCtrlrMgmtPol ctxDn="uni/tn-coke/ctx-mgmt_ctx1">
      <vnsRsMgmtAddr tDn="uni/tn-coke/ap-services/epg-ifc/CtrlrAddrInst-ifc"/>
    </vnsCtrlrMgmtPol>
    <fvBD name="mgmt_ServicesMgmtBD">
      <fvRsCtx tnFvCtxName="mgmt_ctx1"/>
      <fvSubnet ip="3.3.3.3/24"/>
    </fvBD>
    <fvAp name="services">
      <fvAEPg name="ifc">
        <fvRsBd tnFvBDName="mgmt_ServicesMgmtBD"/>
        <vnsAddrInst name="ifc">
          <vnsUcastAddrBlk from="3.3.3.100/24" to="3.3.3.200/24"/>
        </vnsAddrInst>
        <fvRsDomAtt tDn="uni/vmmp-VMware/dom-mininet"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```

- 2 LDevVip に EPG を関連付けます。

```
<polUni>
  <fvTenant name="coke">
    <vnsLDevVip name="ADCCluster1"
      funcType="GoTo" devtype="VIRTUAL">
      <vnsRsMDevAtt tDn="uni/infra/mDev-Citrix-NetScaler-10.5"/>
      <vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-mininet"/>
      <vnsRsDevEpg tDn="uni/tn-coke/ap-services/epg-ifc"/>
    </vnsLDevVip>
    <vnsCMgmt name="devMgmt"
      host="3.3.3.180"
      port="80"/>
    <vnsCCred name="username"
      value="nsroot"/>
    <vnsCCredSecret name="password"
      value="nsroot"/>
  </fvTenant>
</polUni>
```

管理テナント VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の設定

- ステップ 1** tn_mgmt で、APIC から VMware vCenter などの外部ノードへのインバンド VRF 接続を設定します。詳細については、『Cisco APIC Getting Started Guide』の「Configuring In-Band Management Access Using the GUI」セクションを参照してください。
- ステップ 2** In tn_mgmt で、サービスノードに対するサブネットを持つ、mgmt-services-epg などの別のエンドポイントグループ (EPG) を作成します。コントラクトを使用してインバンド VRF の APIC をステッチします。EPG の詳細については、[管理エンドポイントグループの設定](#)、(19 ページ) を参照してください。
- ステップ 3** デバイス クラスタの画面で、mgmt-services-epg などの EPG を選択します。

管理テナントの VRF を使用したデバイス クラスタ内のデバイスへのインバンド接続の REST API による設定

以下は、管理テナント VRF を使用したデバイス クラスタ内のデバイスのインバンド接続を設定するために REST API を使用する例を示しています。

- 1 テナント管理で EPG l4l7MgmtEpg を作成します。



(注) l4l7MgmtEpg は、tn-mgmt 内の inb コンテキスト下の bd アクセスの一部です。

contract1 はデフォルト EPG 内の tn-mgmt l4l7MgmtEpg および tn-mgmt inb 間のコントラクトです。

```
<polUni>
  <fvTenant dn="uni/tn-mgmt">
    <fvAp name="services">
      <fvAEPg name="l4l7MgmtEpg">
        <fvRsBd tnFvBDName="access" />
        <fvRsDomAtt tDn="uni/vmmp-VMware/dom-mininet" />
        <fvRsCons tnVzBrCPName='contract1'>
          </fvRsCons>
        </fvAEPg>
      </fvAp>
      <fvBD name="access">
        <fvSubnet ip="3.3.3.3/24" />
        <fvRsCtx tnFvCtxName="inb"/>
      </fvBD>
      <vzFilter name='all'>
        <vzEntry name='all' ></vzEntry>
      </vzFilter>
      <vzBrCP name="contract1" scope="tenant">
```

```

    <vzSubj name='subj1'>
      <vzInTerm>
        <vzRsFiltAtt tnVzFilterName="all" />
      </vzInTerm>
      <vzOutTerm>
        <vzRsFiltAtt tnVzFilterName="all" />
      </vzOutTerm>
    </vzSubj>
  </vzBrCP>
</fvTenant>
</polUni>

```

- 2 サービス デバイス/VM にサブネット 3.3.3.0/24 で管理 IP アドレスがあることを確認します。
これは **tn-mgmt** アクセス BD が設定されたのと同じサブネットです。（前のステップの設定を参照してください）。
- 3 次を **LDevVip** に追加します。



(注) これは、前のステップで作成された EPG にポイントします。

```
<vnsRsDevEpg tDn="uni/tn-mgmt/ap-services/epg-l4l7MgmtEpg"/>
```

```

<polUni>
  <fvTenant name="mgmt">
    <vnsLDevVip name="ADCCluster1"
      funcType="GoTo" devtype="VIRTUAL">
      <vnsRsMDevAtt tDn="uni/infra/mDev-Citrix-NetScaler-10.5"/>
      <vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-mininet"/>
      <vnsRsDevEpg tDn="uni/tn-mgmt/ap-services/epg-l4l7MgmtEpg"/>
    </vnsLDevVip>
    <vnsCMgmt name="devMgmt"
      host="3.3.3.180"
      port="80"/>
    <vnsCCred name="username"
      value="nsroot"/>
    <vnsCCredSecret name="password"
      value="nsroot"/>
  </fvTenant>
</polUni>

```

- 4 IFC インバンド ゲートウェイにポイントするようにサービス デバイス/VM にルートを追加します。
たとえば、**netScaler** のルート上で、ルート 3.0.0.0/255.255.255.0/3.3.3.3 を追加します。3.0.0.0/24 は IFC インバンド サブネットで、3.3.3.3 は **l4l7MgmtEpg** の SVI IP です。
- 5 次の点を確認します。
 - IFC ルート テーブルに **ifc** インバンド IP のエントリがあります。
 - IFC は、リーフの **l4l7MgmtEpg** ゲートウェイに **ping** を送信できます。
 - サービス ノードは、**l4l7MgmtEpg** SVI ゲートウェイおよび **IFC inb SVI Ip** を **ping** できません。



第 6 章

デバイス クラスタの使用

- [デバイス クラスタ（論理デバイス）のコンテキストについて, 41 ページ](#)
- [論理デバイス コンテキストの設定, 42 ページ](#)
- [REST API を使用した設定, 43 ページ](#)
- [CLI コマンドを使用した設定, 44 ページ](#)
- [デバイス クラスタのインポート, 45 ページ](#)
- [デバイス クラスタのインポートの確認, 45 ページ](#)
- [REST API を使用したインポート, 45 ページ](#)
- [CLI コマンドを使用したインポート, 46 ページ](#)

デバイス クラスタ（論理デバイス）のコンテキストについて

デバイス クラスタは、コントラクト名、グラフ名、またはグラフ内の機能ノード名に基づいて選択できます。デバイス クラスタを作成した後は、デバイス クラスタに選択条件ポリシーを提供するデバイス クラスタ コンテキストを作成できます。

デバイス クラスタ コンテキストは、サービス グラフにデバイスのクラスタを選択するポリシーを指定します。これにより、管理者は複数のデバイス クラスタを持ち、異なるサービス グラフに対して使用することができます。たとえば、管理者は、高パフォーマンス ADC アプライアンスがあるデバイス クラスタおよびパフォーマンスのより低い ADC アプライアンスがある別のデバイス クラスタを持つことができます。高パフォーマンス ADC クラスタ用、および低パフォーマンス ADC クラスタ用の 2 つの異なるデバイス クラスタ コンテキストを使用して、管理者は高パフォーマンスが必要となるアプリケーションには高パフォーマンス ADC クラスタを選択し、低パフォーマンスが必要なアプリケーションには低パフォーマンス ADC クラスタを選択することができます。

論理デバイス コンテキストの設定

論理デバイス コンテキストを設定できます。

はじめる前に

- テナントを設定します。 [テナントの設定](#)、(11 ページ) を参照してください。
- テナントにデバイス クラスタを設定します。 [デバイス クラスタの設定](#) を参照してください。

- ステップ 1** メニュー バーで、[TENANTS] タブをクリックします。 [Tenant] ウィンドウが表示されます。
- ステップ 2** サブメニュー バーで、論理デバイス コンテキストを設定するテナントのタブをクリックします。
- ステップ 3** [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 4** [L4-L7 Services] ブランチを展開します。
- ステップ 5** [Device Cluster Selection Policies] ブランチを展開します。 [Device Cluster Selection Policies] ウィンドウが [Work] ペインに表示されます。
- ステップ 6** [ACTIONS] > [Create Logical Device Context] を選択します。 [CREATE LOGICAL DEVICE CONTEXT] ダイアログボックスが表示されます。
- ステップ 7** 次のフィールドに入力します。

名前	説明
[Contract Name] フィールド	論理デバイス コンテキストのコントラクトの名前。 デバイス クラスタを使用する条件の一部としてコントラクト名を使用したくない場合は、[any] を選択します。
[Graph Name] フィールド	論理デバイス コンテキストのグラフの名前。 デバイス クラスタを使用する条件の一部としてグラフ名を使用したくない場合は、[any] を選択します。
[Node Name] フィールド	論理デバイス コンテキストの機能ノードの名前。 デバイス クラスタを使用する条件の一部としてノードを使用したくない場合は、[any] を選択します。
[Device Cluster] フィールド	論理デバイス コンテキストで使用するデバイスのクラスタ。

- ステップ 8** [Logical Interface Contexts] セクションの [+] をクリックして論理インターフェイス コンテキストを追加します。

- ステップ 9** 次のフィールドに入力します。

名前	説明
[Connector Name] フィールド	サービス グラフ内のコネクタの名前。

名前	説明
[Logical Interface] フィールド	論理インターフェイス コンテキストで指定するコネクタに使用する論理インターフェイス。
[Subnets] フィールド	サービス グラフがインスタンス化される場合に論理インターフェイスで設定するサブネット。

ステップ 10 [SUBMIT] をクリックします。[CREATE LOGICAL DEVICE CONTEXT] ダイアログボックスが閉じます。

REST API を使用した設定

デバイス クラスタ コンテキストを作成するための REST API による設定

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevCtx ctrctNameOrLbl="webCtrct" graphNameOrLbl="G1" nodeNameOrLbl="Node1">
      <vnsRsLDevCtxToLDev tDn="uni/tn-acme/lDevVip-ADCCluster1"/>

      <!-- The connector name C4, C5, etc.. should match the
           Function connector name used in the service graph -->

      <vnsLIfCtx connNameOrLbl="C4">
        <vnsRsLIfCtxToLIf tDn="uni/tn-acme/lDevVip-ADCCluster1/Lif-ext"/>
      </vnsLIfCtx>
      <vnsLIfCtx connNameOrLbl="C5">
        <vnsRsLIfCtxToLIf tDn="uni/tn-acme/lDevVip-ADCCluster1/Lif-int"/>
      </vnsLIfCtx>
    </vnsLDevCtx>
  </fvTenant>
</polUni>
```

デバイス クラスタに論理インターフェイスを追加するための REST API による設定

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1">

    <!-- The LIF name defined here (such as e.g., ext, or int) should match the
           vnsRsLIfCtxToLIf 'tDn' defined in LifCtx -->

    <vnsLIf name="ext">
      <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-outside"/>
      <vnsRsCIfAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-ext"/>
    </vnsLIf>
    <vnsLIf name="int">
```

```

        <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-inside"/>
        <vnsRsCIfAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-int"/>
    </vnsLIf>
  </vnsLDevVip>
</fvTenant>
</polUni>

```

CLI コマンドを使用した設定

```

# logical-device-context

cd '/aci/tenants/acme/l4-17-services/device-cluster-selection-policies'
mcreate 'any' 'any' 'any'
cd 'any-any-any'
moset device-cluster 'tenants/acme/l4-17-services/device-clusters/InsiemeCluster'
moconfig commit

# vns-lifctx

cd
'/aci/tenants/acme/l4-17-services/device-cluster-selection-policies/webCttr-G1-Node1/logical-interface-contexts'
mcreate 'int'
cd 'int'
moset logical-interface
'tenants/acme/l4-17-services/device-clusters/InsiemeCluster/logical-interfaces/int'
moset bridge-domain 'tenants/acme/networking/bridge-domains/MySrvrBD'
moconfig commit

# vns-lifctx

cd
'/aci/tenants/acme/l4-17-services/device-cluster-selection-policies/webCttr-G1-Node1/logical-interface-contexts'
mcreate 'ext'
cd 'ext'
moset logical-interface
'tenants/acme/l4-17-services/device-clusters/InsiemeCluster/logical-interfaces/ext'
moset bridge-domain 'tenants/acme/networking/bridge-domains/MyClntBD'
moconfig commit

```

デバイス クラスタのインポート

共通テナント (**tn-mgmt** など) でクラスタが定義されると、テナントは設定されたデバイス クラスタを次のようにインポートできます。

- ステップ 1 メニュー バーで、[L4-L7 Services] タブをクリックします。
- ステップ 2 サブメニュー バーで、[PACKAGES] タブをクリックします。
- ステップ 3 [Navigation] ペインで、[Device Types] をクリックします。
- ステップ 4 [Actions] > [Import Device Package] を選択します。[IMPORT DEVICE PACKAGE] ダイアログ ボックスが表示されます。
- ステップ 5 [File Name] フィールドで、ベンダーによって提供されたか、または以前に作成したデバイス パッケージを指定します。
- ステップ 6 [Submit] をクリックします。[IMPORT DEVICE PACKAGE] ダイアログ ボックスが閉じます。

デバイス クラスタのインポートの確認

- ステップ 1 メニュー バーで、[TENANTS] タブをクリックします。[Tenant] ウィンドウが表示されます。
- ステップ 2 サブメニュー バーで、デバイス クラスタをインポートするテナントのタブをクリックします。
- ステップ 3 [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 4 [L4-L7 Services] ブランチを展開します。
- ステップ 5 [Imported Device Clusters] ブランチを展開します。
- ステップ 6 適切なデバイス クラスタを選択します。デバイス クラスタ情報が [Work] ペインに表示されます。

REST API を使用したインポート

```
<polUni>
  <fvTenant dn="uni/tn-coke" name="coke">
    <vnsLDevIf ldev="uni/tn-mgmt/lDevVip-ADCCluster1"/>
    <vnsLDevCtx ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any">
      <vnsRsLDevCtxToLDev tDn="uni/tn-coke/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]"/>
      <vnsLIIfCtx connNameOrLbl="inside">
        <vnsRsLIIfCtxToLIIf
tDn="uni/tn-coke/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLIIf-inside"/>
        <fvSubnet ip="10.10.10.10/24"/>
        <vnsRsLIIfCtxToBD tDn="uni/tn-coke/BD-cokeBD1"/>
      </vnsLIIfCtx>
      <vnsLIIfCtx connNameOrLbl="outside">
```

```

        <vnsRsLifCtxToLif
tDn="uni/tn-coke/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLif-outside"/>
        <fvSubnet ip="70.70.70.70/24"/>
        <vnsRsLifCtxToBD tDn="uni/tn-coke/BD-cokeBD4"/>
    </vnsLifCtx>
</vnsLDevCtx>
</fvTenant>
</polUni>

```

CLI コマンドを使用したインポート

```

# logical-device-context
cd '/aci/tenants/coke/l4-l7-services/device-cluster-selection-policies'
mocreate 'any' 'any' 'any'
cd 'any-any-any'
moset device-cluster
'tenants/coke/l4-l7-services/imported-device-clusters/[uni/tn-mgmt/lDevVip-ADCCluster1]'
moconfig commit

# vns-lifctx
cd
'/aci/tenants/coke/l4-l7-services/device-cluster-selection-policies/any-any-any/logical-interface-contexts'
mocreate 'inside'

cd 'inside'

moset logical-interface
'uni/tn-coke/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLif-inside'
moset bridge-domain 'tenants/coke/networking/bridge-domains/cokeBD1'

moconfig commit

# fv-subnet
cd
'/aci/tenants/coke/l4-l7-services/device-cluster-selection-policies/any-any-any/logical-interface-contexts/inside/subnets'
mocreate '10.10.10.10/24'

moconfig commit

# vns-lifctx
cd
'/aci/tenants/coke/l4-l7-services/device-cluster-selection-policies/any-any-any/logical-interface-contexts'
mocreate 'outside'

cd 'outside'

moset logical-interface
'uni/tn-coke/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLif-outside'
moset bridge-domain 'tenants/coke/networking/bridge-domains/cokeBD4'

moconfig commit

# fv-subnet
cd
'/aci/tenants/coke/l4-l7-services/device-cluster-selection-policies/any-any-any/logical-interface-contexts/outside/subnets'
mocreate '70.70.70.70/24'

moconfig commit

# logical-device-consumer
cd '/aci/tenants/coke/l4-l7-services/imported-device-clusters'
mocreate 'uni/tn-mgmt/lDevVip-ADCCluster1'
cd '[uni--tn-mgmt--lDevVip-ADCCluster1]'

```

```
moset faultcode '0'  
moconfig commit
```




第 7 章

サービス グラフの設定

- [サービス グラフについて](#), 49 ページ
- [機能ノードについて](#), 49 ページ
- [機能ノード コネクタについて](#), 50 ページ
- [サービス グラフ接続について](#), 50 ページ
- [端末ノードについて](#), 50 ページ
- [サービス グラフ コンフィギュレーション パラメータについて](#), 50 ページ
- [サービス グラフの設定](#), 51 ページ
- [サービス グラフを作成するための REST API による設定](#), 51 ページ
- [CLI コマンドを使用した設定](#), 52 ページ

サービス グラフについて

サービス グラフは、端末セット間の順序付けられた一連の機能ノードで、アプリケーションが必要とする一連のネットワーク サービス機能を識別します。グラフ内のサービス機能は、アプリケーションの要件に基づいたサービス デバイスに自動的にプロビジョニングされます。

機能ノードの詳細については、[機能ノードについて](#)、(49 ページ) を参照してください。

GUI、CLI、または Application Policy Infrastructure Controller (APIC) を使用してサービス グラフを定義できます。APIC を通じたサービス デバイスの設定では、サービス デバイスの変更は必要ありません。

機能ノードについて

機能ノードは、単一のサービス機能を表します。機能ノードには、サービス機能のネットワーク要件を表す機能ノード コネクタがあります。機能ノード コネクタの詳細については、[機能ノード コネクタについて](#)、(50 ページ) を参照してください。

サービスグラフ内の機能ノードは、1つ以上のパラメータが必要になる場合があります。パラメータは、エンドポイントグループ（EPG）、アプリケーションプロファイル、またはテナントコンテキストで指定できます。パラメータは、サービスグラフ定義時に割り当てることができます。パラメータ値は変更がさらに加えられるのを防ぐためにロックできます。

機能ノードコネクタについて

機能ノードコネクタは、サービスグラフに機能ノードを接続し、グラフのコネクタサブネットに基づいて適切なブリッジドメインと接続と関連付けられます。各コネクタは、VLANまたはVirtual Extensible LAN（VXLAN）に関連付けられます。コネクタの両側がエンドポイントグループ（EPG）として扱われ、ホワイトリストがスイッチにダウンロードされ、2つの機能ノード間の通信がイネーブルになります。

サービスグラフ接続について

サービスグラフ接続は、1つの機能ノードを別の機能ノードに接続します。

端末ノードについて

端末ノードはサービスグラフとコントラクトを接続します。コントラクトに端末ノードを接続することにより、2台のアプリケーションエンドポイントグループ（EPG）間のトラフィックにサービスグラフを挿入できます。接続されると、コントラクトのコンシューマEPGとプロバイダーEPG間のトラフィックはサービスグラフにリダイレクトされます。

サービスグラフコンフィギュレーションパラメータについて

サービスグラフは、デバイスパッケージによって指定される、コンフィギュレーションパラメータを持つことができます。コンフィギュレーションパラメータは、EPG、アプリケーションプロファイルまたはテナントコンテキストでも指定できます。サービスグラフ内の機能ノードでは、1つ以上のコンフィギュレーションパラメータが必要になる場合があります。パラメータ値は変更がさらに加えられるのを防ぐためにロックできます。

サービスグラフを設定し、コンフィギュレーションパラメータの値を指定すると、Application Policy Infrastructure Controller（APIC）はデバイスパッケージ内のデバイススクリプトにパラメータを渡します。デバイススクリプトは、パラメータデータをデバイスにダウンロードされる設定に変換します。

サービスグラフの設定

デバイスを登録した後は、そのデバイスおよびそのデバイスが提示したすべての機能を使用してサービスグラフを作成できます。サービスグラフはテナント共通またはテナント固有として作成できます。これは、プロバイダー管理者またはテナント内のテナント管理者によって実行されます。

ステップ 1 [Navigation] ペインで、[L4-L7 Services] > [Service Graphs] をクリックします。

ステップ 2 [Action] > [Create L4-L7 Service Graph] を選択します。[CREATE L4-L7 SERVICE GRAPH] ダイアログボックスが表示されます。

左側のペインには、Application Policy Infrastructure Controller (APIC) が認識しているサービスデバイス、およびデバイスによって提供されるサービス機能が一覧表示されます。APIC は、インポートしたデバイスパッケージからのこれらのデバイスを認識します。

ステップ 3 次のフィールドに入力します。

名前	説明
[Name] フィールド	サービスグラフの名前。
[Description] フィールド	サービスグラフの説明

ステップ 4 左ペインから右ペインにサービス機能をドラッグアンドドロップして、その機能をサービスグラフに追加します。

ステップ 5 デバイスパッケージに基づいて、機能コネクタ、int および ext をクリックして、メタコネクタ、Inside および Outside にそれぞれマッピングします。メタコネクタ、Inside および Outside はデバイス固有です。

ステップ 6 終端ノードをクリックして接続を機能のコネクタにドラッグすることで、終端ノードをサービス機能のコネクタに接続します。機能に接続したい各終端ノードに対してこのアクションを繰り返します。端末ノードはアプリケーション EPG の接続を表します。

ステップ 7 機能をクリックします。[Config Profile] ダイアログボックスが表示され、サービスノードのプロファイルを選択するよう促されます。

サービス機能パラメータをこの時点で設定するか、グラフがコントラクトに関連付けられるときのグラフアタッチメントの時点で設定できます。

ステップ 8 サービス機能パラメータの設定が完了したら、[CREATE L4-L7 SERVICE GRAPH] ダイアログボックスに戻り、[Submit] をクリックしてグラフを作成します。[Service Graph] ダイアログボックスは作成した新しいグラフを表示します。

サービスグラフを作成するための REST API による設定

```
<polUni>
  <fvTenant name="acme">
```

```

<vnsAbsGraph name="G1">
  <vnsAbsTermNodeCon name="Input1">
    <vnsAbsTermConn name="C1">
      </vnsAbsTermConn>
    </vnsAbsTermNodeCon>
    <vnsAbsNode name="Node" funcType="GoTo">
      <vnsRsDefaultScopeToTerm
tDn="uni/tn-acme/AbsGraph-G1/AbsTermNodeProv-Output1/outtmn1"/>
      <vnsAbsFuncConn name="inside">
        <vnsRsMConnAtt
tDn="uni/infra/mDev-Insieme-Generic-1.0/mFunc-SubnetFunc/mConn-external"/>
        </vnsAbsFuncConn>
        <vnsAbsFuncConn name="outside">
          <vnsRsMConnAtt
tDn="uni/infra/mDev-Insieme-Generic-1.0/mFunc-SubnetFunc/mConn-internal"/>
          </vnsAbsFuncConn>
        <vnsAbsDevCfg>
          <vnsAbsFolder key="oneFolder" name="f1">
            <vnsAbsParam key="oneParam" name="p1" value="v1"/>
          </vnsAbsFolder>
        </vnsAbsDevCfg>
        <vnsAbsFuncCfg>
          <vnsAbsFolder key="folder" name="folder1" devCtxLbl="C1">
            <vnsAbsParam key="param" name="param" value="value"/>
          </vnsAbsFolder>
          <vnsAbsFolder key="folder" name="folder2" devCtxLbl="C2">
            <vnsAbsParam key="param" name="param" value="value"/>
          </vnsAbsFolder>
        </vnsAbsFuncCfg>
        <vnsRsNodeToMFunc tDn="uni/infra/mDev-Insieme-Generic-1.0/mFunc-SubnetFunc"/>
      </vnsAbsNode>
    <vnsAbsTermNodeProv name="Output1">
      <vnsAbsTermConn name="C6">
        </vnsAbsTermConn>
      </vnsAbsTermNodeProv>
    <vnsAbsConnection name="CON1">
      <vnsRsAbsConnectionConns
tDn="uni/tn-acme/AbsGraph-G1/AbsTermNodeCon-Input1/AbsTConn"/>
      <vnsRsAbsConnectionConns tDn="uni/tn-acme/AbsGraph-G1/AbsNode-Node/AbsFConn-inside"/>
    </vnsAbsConnection>
    <vnsAbsConnection name="CON3">
      <vnsRsAbsConnectionConns tDn="uni/tn-acme/AbsGraph-G1/AbsNode-Node/AbsFConn-outside"/>
    </vnsAbsConnection>
    <vnsRsAbsConnectionConns
tDn="uni/tn-acme/AbsGraph-G1/AbsTermNodeProv-Output1/AbsTConn"/>
  </vnsAbsGraph>
</fvTenant>
</polUni>

```

CLI コマンドを使用した設定

```

# 14-17-service-graph
cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs'
mcreate 'g1'
moconfig commit

# consumer-terminal-node
cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/consumer-terminal-nodes'
mcreate 'Input1'
moconfig commit

# terminal-connector

```

```
cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/consumer-terminal-nodes/Input1'
mcreate terminal-connector
cd 'terminal-connector'
moset name 'C1'
moconfig commit

# provider-terminal-node

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/provider-terminal-nodes'
mcreate 'Output1'
moconfig commit

# terminal-connector

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/provider-terminal-nodes/Output1'
mcreate terminal-connector
cd 'terminal-connector'
moset name 'C6'
moconfig commit

# function-node

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes'
mcreate 'ASA_FW'
cd 'ASA_FW'
moset function-name '14-17-services/packages/CISCO-ASA-1.0.1.37/functions/Firewall'
moconfig commit

# device-config

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes/ASA_FW'
mcreate device-config
moconfig commit

# function-connector

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes/ASA_FW'
mcreate function-connector 'external'
cd 'function-connector-external'
moset meta-connector 'uni/infra/mDev-CISCO-ASA-1.0.1.37/mFunc-Firewall/mConn-external'
moconfig commit

# function-connector

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes/ASA_FW'
mcreate function-connector 'internal'
cd 'function-connector-internal'
moset meta-connector 'uni/infra/mDev-CISCO-ASA-1.0.1.37/mFunc-Firewall/mConn-internal'
moconfig commit

# default-scope-to-term

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes/ASA_FW'
mcreate default-scope-to-term
cd 'default-scope-to-term'
moset target-dn 'uni/tn-asa_tenant1/AbsGraph-g1/AbsTermNodeProv-Output1/outtmn1'
moconfig commit

# connection

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/connections'
mcreate 'CON2'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/connections/CON2'
mcreate
```

```
'tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes/ASA_FW/function-connector-internal'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/connections/CON2'
mocreate
'tenants/asa_tenant1/14-17-services/service-graphs/g1/provider-terminal-nodes/Output1/terminal-connector'
moconfig commit

# connection

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/connections'
mocreate 'CON1'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/connections/CON1'
mocreate
'tenants/asa_tenant1/14-17-services/service-graphs/g1/consumer-terminal-nodes/Input1/terminal-connector'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors

cd '/aci/tenants/asa_tenant1/14-17-services/service-graphs/g1/connections/CON1'
mocreate
'tenants/asa_tenant1/14-17-services/service-graphs/g1/function-nodes/ASA_FW/function-connector-external'
moconfig commit
admin@ifav15-ifc2:g1>
```



第 8 章

設定パラメータ

- [デバイス パッケージ仕様内のコンフィギュレーションパラメータ, 55 ページ](#)
- [抽象機能プロファイル内のコンフィギュレーションパラメータ, 59 ページ](#)
- [サービス グラフでの抽象機能ノード内のコンフィギュレーションパラメータ, 63 ページ](#)
- [各種の設定 MO 内のコンフィギュレーションパラメータ, 68 ページ](#)
- [パラメータ解決, 71 ページ](#)
- [パラメータ解決時の MO の検索, 72 ページ](#)

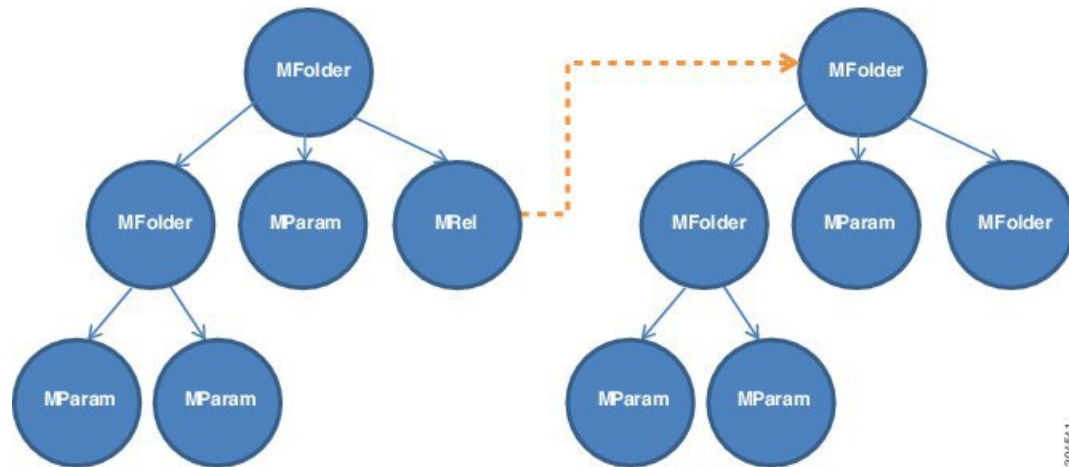
デバイスパッケージ仕様内のコンフィギュレーションパラメータ

デバイスパッケージにはサービス デバイスの仕様を示す XML ファイルが含まれます。この仕様にはデバイス情報およびサービス デバイスによって提供される各種の機能が含まれます。

デバイス仕様の一部として、このファイルにはサービス デバイスによって必要なコンフィギュレーションの宣言が含まれる必要があります。この設定は、グラフのインストール中にサービス デバイスによって提供される各種の機能を設定するために必要です。

次の図は、デバイス パッケージ内のコンフィギュレーションパラメータ階層を示しています。

図 1: デバイス パッケージ内のコンフィギュレーションパラメータ階層



MFold

MFold は、MParam および他のネストされた MFold を含む可能性のあるコンフィギュレーションアイテムのグループです。MFold は次の属性を持ちます。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Description	コンフィギュレーションアイテムを説明します。
Cardinality	コンフィギュレーションアイテムの濃度を指定します。濃度のデフォルト値は 1 です。濃度が N であれば、Application Policy Infrastructure Controller (APIC) ではコンフィギュレーションパラメータの N インスタンスの設定が可能です。
ScopedBy	パラメータ解決の範囲を指定します。ScopedBy は、APIC がコンフィギュレーション MO からパラメータを解決する場合にパラメータ値を検索する場所を決定します。詳細については、 パラメータ解決 、(71 ページ) を参照してください。 デフォルト値は Epg です。サポートされる値は Tenant、Ap、Bd、および Epg です。
RsConnector	コンフィギュレーションアイテムを MConn に関連付ける関係。
DevCtx	コンフィギュレーションアイテムをデバイスクラスタ (LDev) 内の特定の物理デバイス (CDev) に関連付けることができます。

属性	説明
Locked	コンフィギュレーションアイテム値がロックされます。一度ロックされると値は変更できません。

MParam

MParam は、単一のコンフィギュレーションパラメータを宣言するコンフィギュレーションパラメータの基本単位です。MParam は次の属性を持ちます。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Description	コンフィギュレーションアイテムを説明します。
Cardinality	コンフィギュレーションアイテムの濃度を指定します。濃度のデフォルト値は 1 です。濃度が N であれば、APIC ではコンフィギュレーションパラメータの N インスタンスの設定が可能です。
RsConnector	コンフィギュレーションアイテムを MConn に関連付ける関係。
Mandatory	コンフィギュレーションアイテムが必須としてマークされます。
Locked	コンフィギュレーションアイテム値がロックされます。一度ロックされると値は変更できません。
Validation	値の検証方法を指定します。

MRel

MRel は 1 つの MFolder が別の MFolder を参照することを可能にします。MFolder 内の MRel を使用して、管理者は含む側の MFolder を、MRel 内に含まれる RsTarget 関係によって MRel からポイントされる MFolder に関連付けることができます。MRel は次の属性を持ちます。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Description	コンフィギュレーションアイテムを説明します。
Cardinality	コンフィギュレーションアイテムの濃度を指定します。濃度のデフォルト値は 1 です。
RsTarget	コンフィギュレーションフォルダを別の MFolder に関連付ける関係。この関係に対する TDn の値はターゲットフォルダの DN です。

属性	説明
RsConnector	コンフィギュレーション アイテムを MConn に関連付ける関係。
Mandatory	コンフィギュレーション アイテムが必須としてマークされます。

デバイス パッケージ仕様の設定スコープ

デバイス仕様ファイルで、コンフィギュレーション アイテムは異なるセクションで配置されます。

MDevCfg

MDevCfg のセクションでは、デバイスを使用するすべてのサービスグラフで共有されるデバイスレベルの設定について説明します。Application Policy Infrastructure Controller (APIC) は、このセクションで説明されるコンフィギュレーション アイテムを使用して作成されたコンフィギュレーションオブジェクトの参照カウントを実行します。オブジェクトは、デバイスを使用しているすべてのグラフ インスタンスが削除された後にのみサービス デバイスから削除されます。

MFuncCfg

MFuncCfg は、サービス機能に対してローカルで、サービス機能に固有なコンフィギュレーションについて説明します。APIC は、このセクションで説明されるコンフィギュレーション アイテムによって作成されたコンフィギュレーション オブジェクトの参照カウントを実行します。オブジェクトが作成され、サービス機能がインスタンス化または削除されたときに削除されます。

MGrpCfg

MGrpCfg は、デバイスを使用するサービスグラフのすべての機能によって共有される設定を説明します。APIC は、このセクションで説明されるコンフィギュレーション アイテムを使用して作成されたコンフィギュレーションオブジェクトの参照カウントを実行します。オブジェクトは、サービス グラフからすべての機能が削除された後にサービス デバイスから削除されます。

デバイス パッケージ内のコンフィギュレーション パラメータの XML の例

次の XML の例は、デバイス パッケージ内のコンフィギュレーション パラメータを示しています。

```
<vnsMFolder key="VServer" scopedBy="epg">
  <vnsRsConnector tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external"/>
  <vnsMParam key="vservname" description="Name of VServer" mandatory="true"/>
  <vnsMParam key="vip" description="Virtual IP"/>
  <vnsMParam key="subnet" description="Subnet IP"/>
  <vnsMParam key="port" description="Port for Virtual server"/>
  <vnsMParam key="persistencetype" description="persistencetype"/>
  <vnsMParam key="servicename" description="Service bound to this vServer"/>
  <vnsMParam key="servicetype" description="Service bound to this vServer"/>
  <vnsMParam key="clttimeout" description="Client timeout"/>
</vnsMFolder>
```

```

    <vnsMFolder key="VServerGlobalConfig" description="This references the global
configuration">
    <vnsMRel key="ServiceConfig">
    <vnsRsTarget tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Service"/>
    </vnsMRel>
    <vnsMRel key="ServerConfig">
    <vnsRsTarget tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Server"/>
    </vnsMRel>
    <vnsMRel key="VipConfig">
    <vnsRsTarget
tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Network/mFolder-vip"/>
    <vnsRsConnector tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external"/>
    </vnsMRel>
    </vnsMFolder>
</vnsMFolder>

```

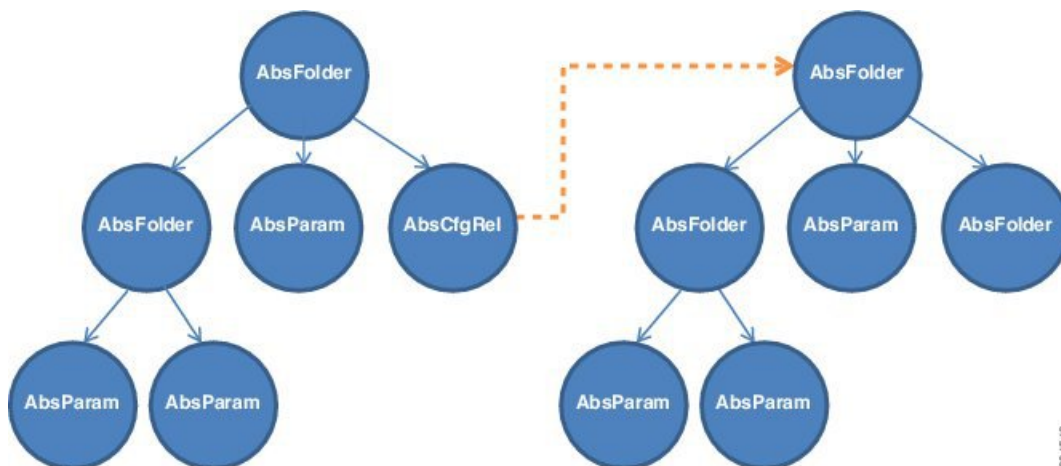
抽象機能プロファイル内のコンフィギュレーションパラメータ

抽象プロファイルを使用すると、管理者はコンフィギュレーションパラメータのデフォルト値を設定できます。抽象機能プロファイルには値を持つコンフィギュレーションパラメータが含まれます。これらの値がグラフインスタンス作成時にデフォルト値として使用されます。デフォルト値がどのように使用されるかの詳細は、[パラメータ解決](#)、(71 ページ) を参照してください。

抽象機能プロファイルはサービスグラフの機能ノードに接続されます。抽象機能プロファイルで指定されたデフォルト値は、グラフのインスタンス化の際にサービスデバイスに機能をレンダリングする場合に使用されます。

次の図は、抽象機能プロファイル内のコンフィギュレーションパラメータ階層を示しています。

図 2：抽象機能プロファイル内部のコンフィギュレーションパラメータ階層



30-45-42

AbsFolder

AbsFolder は、AbsParam および他のネストされた AbsFolder を含むことのできるコンフィギュレーションアイテムのグループです。各 AbsFolder に対して、デバイスパッケージ内に MFolder が必要です。Application Policy Infrastructure Controller (APIC) は、各 AbsFolder を検証して、パッケージ内に AbsFolder に対応する MFolder が存在することを確認します。AbsFolder には、次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Description	コンフィギュレーションアイテムを説明します。
Cardinality	コンフィギュレーションアイテムの濃度を指定します。デフォルト値は 1 です。
ScopedBy	パラメータ解決の範囲を指定します。ScopedBy は、APIC がコンフィギュレーション MO からパラメータを解決する場合にパラメータ値を検索する場所を決定します。詳細については、 パラメータ解決 、(71 ページ) を参照してください。 デフォルト値は Epg です。サポートされる値は Tenant、Ap、Bd、および Epg です。
DevCtx	コンフィギュレーションアイテムをデバイスクラスタ (LDev) 内の特定の物理デバイス (CDev) に関連付けることができます。
Locked	コンフィギュレーションアイテム値がロックされます。一度ロックされると値は変更できません。

AbsParam

AbsParam はコンフィギュレーションパラメータの基本単位です。AbsParam は単一のコンフィギュレーションパラメータを定義します。AbsFolder と同様、各 AbsParam に対してデバイス仕様内に対応する MFolder が存在する必要があります。APIC は仕様を検証して、パッケージ内に AbsParam に対応する MFolder が存在することを確認します。AbsParam の値は、MParam 内で指定される検証メソッドを使用して検証されます。AbsParam には、次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Value	特定のコンフィギュレーションアイテムの値を保持します。値は MParam ではサポートされません。

属性	説明
Description	コンフィギュレーション アイテムを説明します。
Cardinality	コンフィギュレーション アイテムの濃度を指定します。 デフォルト値は 1 です。
Mandatory	コンフィギュレーション アイテムが必須としてマークされます。
Locked	コンフィギュレーション アイテム値がロックされます。 一度ロックされると値は変更できません。
Validation	コンフィギュレーション パラメータの検証に使用する検証メカニズムを指定します。

AbsRel

AbsRel は 1 つの AbsFolder が別の AbsFolder を参照することを可能にします。 AbsRel には、次の属性があります。

属性	説明
Key	コンフィギュレーション アイテムのタイプを定義します。 キーは、デバイス パッケージで定義されており、上書きすることはできません。 キーは、検証だけでなく一致基準として使用されます。
Value	特定のコンフィギュレーション アイテムの値を保持します。 値は MParam ではサポートされません。
Description	コンフィギュレーション アイテムを説明します。
Cardinality	コンフィギュレーション アイテムの濃度を指定します。 デフォルト値は 1 です。
Mandatory	コンフィギュレーション アイテムが必須としてマークされます。

抽象機能プロファイルの設定スコープ

抽象機能プロファイルでは、コンフィギュレーションパラメータはデバイスパッケージ内の場合と似た方法で構成されます。 3 種類のスコープがあります。

AbsDevCfg

このセクションは、デバイス パッケージ内のデバイス レベル設定と宣言される、コンフィギュレーションアイテムのデフォルト値を提供します。 コンフィギュレーションアイテムは MDevCfg で指定されます。

各コンフィギュレーションアイテムに対して、デバイス パッケージに同等のコンフィギュレーションアイテムが存在する必要があります。

このセクションで説明される設定は、デバイスを使用するサービス グラフで共有されます。

Application Policy Infrastructure Controller (APIC) は、このセクションで説明されるコンフィギュレーションアイテムを使用して作成されたコンフィギュレーションオブジェクトの参照カウントを実行します。オブジェクトは、デバイスを使用しているすべてのグラフィンスタンスが削除された後にのみサービス デバイスから削除されます。

AbsGrpCfg

このセクションは、デバイス パッケージ内のデバイス レベル設定と宣言される、コンフィギュレーションアイテムのデフォルト値を提供します。コンフィギュレーションアイテムは **MGrpCfg** で指定されます。

各コンフィギュレーションアイテムに対して、デバイス パッケージに同等のコンフィギュレーションアイテムが存在する必要があります。

このセクションで説明される設定は、デバイスを使用するサービス グラフのすべての機能で共有されます。APIC は、このセクションで説明されるコンフィギュレーションアイテムを使用して作成されたコンフィギュレーション オブジェクトの参照カウントを実行します。オブジェクトは、デバイスを使用しているすべてのグラフィンスタンスが削除された後にのみサービスデバイスから削除されます。

AbsFuncCfg

このセクションは、デバイス パッケージ内の機能レベル設定と宣言される、コンフィギュレーションアイテムのデフォルト値を提供します。コンフィギュレーションアイテムは **MFuncCfg** で指定されます。

各コンフィギュレーションアイテムに対して、デバイス パッケージに同等のコンフィギュレーションアイテムが存在する必要があります。

このセクションは、サービス機能にローカルな設定を説明するために使用されます。このセクションで説明されている設定は、サービス機能に固有のもので、APIC は、このセクションで説明されるコンフィギュレーションアイテムによって作成されたコンフィギュレーション オブジェクトの参照カウントを実行します。オブジェクトが作成され、サービス機能がインスタンス化または削除されたときに削除されます。

コンフィギュレーションパラメータを持つ抽象機能プロファイルに対する XML POST の例

次の XML POST の例は、コンフィギュレーションパラメータを持つ抽象機能プロファイルを示しています。

```
<vnsAbsFuncProfContr name = "NP">
  <vnsAbsFuncProfGrp name = "Grp1">
    <vnsAbsFuncProf name = "P1">
      <vnsRsProfToMFunc tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB"/>
      <vnsAbsDevCfg name="D1">
        <vnsAbsFolder key="Service" name="Service-Default" cardinality="n">
```

```
<vnsAbsParam name="servicetype" key="servicetype" value="TCP"/>
<vnsAbsParam name="serviceport" key="serviceport" value="80"/>
<vnsAbsParam name="maxclient" key="maxclient" value="1000"/>
<vnsAbsParam name="maxreq" key="maxreq" value="100"/>
<vnsAbsParam name="cip" key="cip" value="enable"/>
<vnsAbsParam name="usip" key="usip" value="enable"/>
<vnsAbsParam name="sp" key="sp" value=""/>
<vnsAbsParam name="svrtimeout" key="svrtimeout" value="60"/>
<vnsAbsParam name="clttimeout" key="clttimeout" value="60"/>
<vnsAbsParam name="cka" key="cka" value="NO"/>
<vnsAbsParam name="tcpb" key="tcpb" value="NO"/>
<vnsAbsParam name="cmp" key="cmp" value="NO"/>
</vnsAbsFolder>
</vnsAbsDevCfg>
<vnsAbsFuncCfg name="SLB">
  <vnsAbsFolder key="VServer" name="VServer-Default">
    <vnsAbsParam name="port" key="port" value="80"/>
    <vnsAbsParam name="persistencetype" key="persistencetype"
      value="cookie"/>
    <vnsAbsParam name="clttimeout" key="clttimeout" value="100"/>
    <vnsAbsParam name="servicetype" key="servicetype" value="TCP"/>
    <vnsAbsParam name="servicename" key="servicename"/>
  </vnsAbsFolder>
</vnsAbsFuncCfg>
</vnsAbsFuncProf>
</vnsAbsFuncProfGrp>
</vnsAbsFuncProfContr>
```

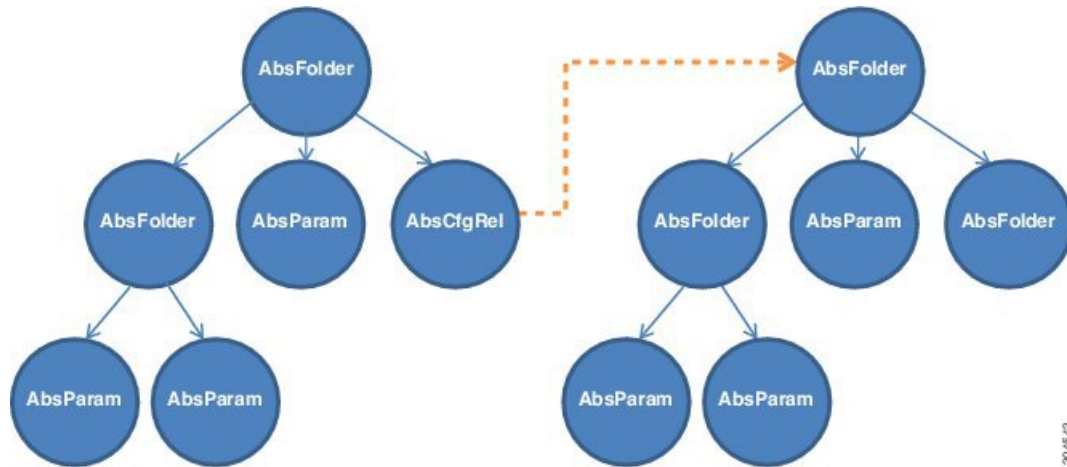
サービスグラフでの抽象機能ノード内のコンフィギュレーションパラメータ

サービスグラフ内の機能ノードを使用して、管理者はコンフィギュレーションパラメータの値を設定できます。これらの値は、グラフのインストール時に使用されます。値がどのように使用されるかの詳細は、[パラメータ解決](#)、(71 ページ) を参照してください。

抽象機能ノードでは、コンフィギュレーションパラメータは抽象機能プロファイル内の場合と似た方法で構成されます。

次の図は、抽象機能ノード内のコンフィギュレーションパラメータ階層を示しています。

図 3：抽象機能ノード内のコンフィギュレーションパラメータ



AbsDevCfg

このセクションは、デバイスパッケージ内のデバイスレベル設定と宣言される、コンフィギュレーションアイテムのデフォルト値を提供するために使用されます。コンフィギュレーションアイテムは MDevCfg で指定されます。

これらの各コンフィギュレーションアイテムに対して、デバイスパッケージに同等のコンフィギュレーションアイテムが存在する必要があります。

AbsGrpCfg

このセクションは、デバイスパッケージ内のデバイスレベル設定と宣言される、コンフィギュレーションアイテムのデフォルト値を提供するために使用されます。コンフィギュレーションアイテムは MGrpCfg で指定されます。

これらの各コンフィギュレーションアイテムに対して、デバイスパッケージに同等のコンフィギュレーションアイテムが存在する必要があります。

このセクションで説明される設定は、デバイスを使用するサービスグラフのすべての機能で共有されます。Application Policy Infrastructure Controller (APIC) は、このセクションで説明されるコンフィギュレーションアイテムを使用して作成されたコンフィギュレーションオブジェクトの参照カウントを実行します。オブジェクトは、サービスグラフからすべての機能が削除された後にサービスデバイスから削除されます。

AbsFuncCfg

このセクションは、デバイス パッケージ内の機能レベル設定と宣言される、コンフィギュレーション アイテムのデフォルト値を提供するために使用されます。コンフィギュレーション アイテムは MFuncCfg で指定されます。

これらの各コンフィギュレーション アイテムに対して、デバイス パッケージに同等のコンフィギュレーション アイテムが存在する必要があります。

このセクションは、サービス機能にローカルな設定を説明するために使用されます。このセクションで説明されている設定は、サービス機能に固有のもので、APIC は、このセクションで説明されるコンフィギュレーション アイテムによって作成されたコンフィギュレーション オブジェクトの参照カウントを実行します。オブジェクトが作成され、サービス機能がインスタンス化または削除されたときに削除されます。

AbsFolder

AbsFolder は、AbsParams および他のネストされた AbsFolder を含む可能性のあるコンフィギュレーション アイテムのグループです。各 AbsFolder に対して、デバイス パッケージ内に MFolder が必要です。APIC は、各 AbsFolder を検証して、パッケージ内に AbsFolder に対応する MFolder が存在することを確認します。AbsFolder には、次の属性があります。

属性	説明
Key	コンフィギュレーション アイテムのタイプを定義します。キーは、デバイス パッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Description	コンフィギュレーション アイテムを説明します。
Cardinality	コンフィギュレーション アイテムの濃度を指定します。デフォルト値は 1 です。
ScopedBy	パラメータ解決の範囲を指定します。ScopedBy は、APIC がコンフィギュレーション MO からパラメータを解決する場合にパラメータ値を検索する場所を決定します。詳細については、 パラメータ解決 、(71 ページ) を参照してください。 デフォルト値は Epg です。サポートされる値は Tenant、Ap、Bd、および Epg です。
RsCfgToConn	コンフィギュレーション アイテムを AbsConn に関連付ける関係。
DevCtx	コンフィギュレーション アイテムをデバイス クラスタ (LDev) 内の特定の物理 デバイス (CDev) に関連付けることができます。
Locked	コンフィギュレーション アイテム値がロックされます。一度ロックされると値は変更できません。

AbsParam

AbsParam はコンフィギュレーションパラメータの基本単位です。AbsParam は単一のコンフィギュレーションパラメータを定義します。AbsFolder と同様、各 AbsParam に対してデバイス仕様内に対応する MFolder が存在する必要があります。APIC は仕様を検証して、パッケージ内に AbsParam に対応する MFolder が存在することを確認します。AbsParam の値は、MParam 内で指定される検証メソッドを使用して検証されます。AbsParam には次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Value	特定のコンフィギュレーションアイテムの値を保持します。値は MParam ではサポートされません。
Description	コンフィギュレーションアイテムを説明します。
Cardinality	コンフィギュレーションアイテムの濃度を指定します。デフォルト値は 1 です。
RsCfgToConn	コンフィギュレーションアイテムを MConn に関連付ける関係。
Mandatory	コンフィギュレーションアイテムが必須としてマークされます。
Locked	コンフィギュレーションアイテム値がロックされます。一度ロックされると値は変更できません。
Validation	コンフィギュレーションパラメータの検証に使用する検証メカニズムを指定します。

AbsRel

AbsRel は 1 つの AbsFolder が別の AbsFolder を参照することを可能にします。AbsRel には次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Value	特定のコンフィギュレーションアイテムの値を保持します。値は MParam ではサポートされません。
Description	コンフィギュレーションアイテムを説明します。
Cardinality	コンフィギュレーションアイテムの濃度を指定します。デフォルト値は 1 です。

属性	説明
RsCfgToConn	コンフィギュレーションアイテムを MConn に関連付ける関係。
Mandatory	コンフィギュレーションアイテムが必須としてマークされます。
Locked	コンフィギュレーションアイテム値がロックされます。一度ロックされると値は変更できません。

コンフィギュレーションパラメータを持つ抽象機能ノードに対する XML POST の例

次の XML POST の例は、コンフィギュレーションパラメータを持つ抽象機能ノードを示しています。

```
<vnsAbsNode name = "SLB" funcType="GoTo" >
  <vnsRsDefaultScopeToTerm tDn="uni/tn-coke/AbsGraph-G3/AbsTermNode-Output1/outtmnl"/>
  <vnsAbsFuncConn name = "C4" direction = "input">
    <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external" />
  </vnsAbsFuncConn>
  <vnsAbsFuncConn name = "C5" direction = "output">
    <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-internal" />
  </vnsAbsFuncConn>
  <vnsAbsDevCfg>
    <vnsAbsFolder key="Network" name="Network" scopedBy="epg">
      <!-- Following scopes this folder to input terminal or Src Epg -->
      <vnsRsScopeToTerm tDn="uni/tn-coke/AbsGraph-G3/AbsTermNode-Output1/outtmnl"/>
      <!-- VIP address -->
      <vnsAbsFolder key="vip" name="vip" scopedBy="epg">
        <vnsAbsParam name="vipaddress" key="vipaddress" value=""/>
      </vnsAbsFolder>
      <!-- SNIP address -->
      <vnsAbsFolder key="snip" name="snip" scopedBy="epg">
        <vnsAbsParam name="snipaddress" key="snipaddress" value=""/>
      </vnsAbsFolder>
    </vnsAbsFolder>
    <vnsAbsFolder key="Service" name="Service" scopedBy="epg" cardinality="n">
      <vnsRsScopeToTerm tDn="uni/tn-coke/AbsGraph-G3/AbsTermNode-Output1/outtmnl"/>
      <vnsAbsParam name="servicename" key="servicename" value=""/>
      <vnsAbsParam name="servername" key="servername" value=""/>
      <vnsAbsParam name="serveripaddress" key="serveripaddress" value=""/>
    </vnsAbsFolder>
  </vnsAbsDevCfg>
  <vnsAbsFuncCfg>
    <vnsAbsFolder key="VServer" name="VServer" scopedBy="epg">
      <vnsRsScopeToTerm tDn="uni/tn-coke/AbsGraph-G3/AbsTermNode-Output1/outtmnl"/>
      <!-- Virtual Server Configuration -->
      <vnsAbsParam name="vip" key="vip" value=""/>
      <vnsAbsParam name="vservername" key="vservername" value=""/>
      <vnsAbsParam name="servicename" key="servicename"/>
      <vnsRsCfgToConn tDn="uni/tn-coke/AbsGraph-G3/AbsNode-Node2/AbsFConn-C4" />
    </vnsAbsFolder>
  </vnsAbsFuncCfg>
  <vnsRsNodeToMFunc tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB"/>
</vnsAbsNode>
```

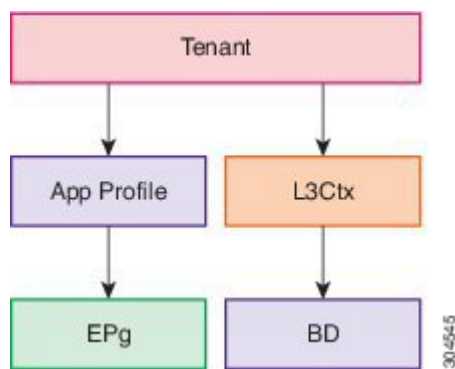
各種の設定 MO 内のコンフィギュレーションパラメータ

管理者は EPG、テナント、BD、または AP などの各種の Application Policy Infrastructure Controller (APIC) MO の一部としてサービス機能に対するコンフィギュレーションパラメータを指定できます。グラフがインスタンス化されると、APIC は各種の場所からパラメータを検索することでグラフに必要な設定を解決します。インスタンス化では、パラメータ値はデバイススクリプトに解決され、渡されます。

各種の MO 内でコンフィギュレーションパラメータを保持できることの柔軟性により、管理者は単一のサービスグラフを設定し、グラフを異なるテナントまたはエンドポイントグループ (EPG) に対して異なる設定で使用できます。

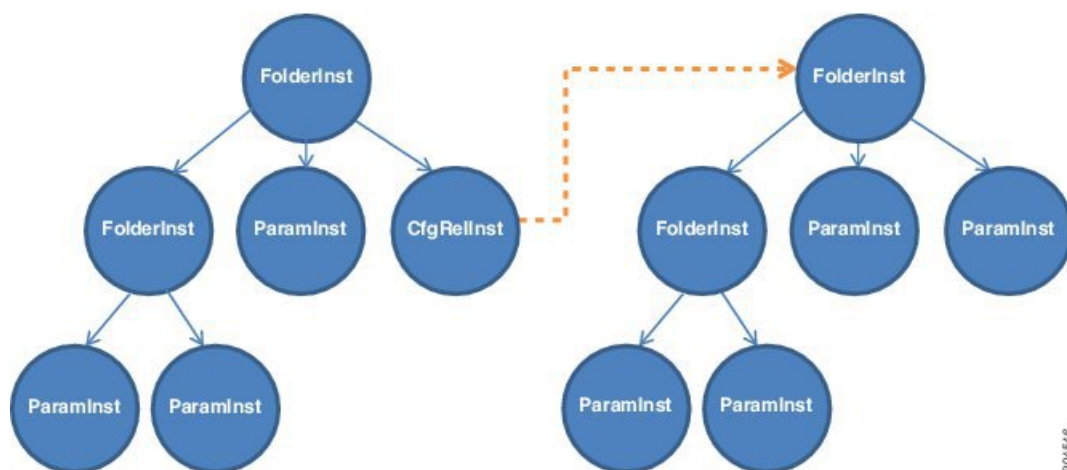
次の図は、APIC MO の階層を示しています。

図 4: APIC MO の階層



次の図は、各種のコンフィギュレーション MO 内のコンフィギュレーションパラメータを示しています。

図 5: 各種の設定 MO 内のコンフィギュレーションパラメータ



FolderInst

FolderInst は、ParamInst および他のネストされた FolderInst を含む可能性のあるコンフィギュレーションアイテムのグループです。FolderInst には、次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
ctrctNameOrLbl	パラメータの解決時に一致する FolderInst を検索します。FolderInst をパラメータ解決で使用するには、このプロパティはサービス グラフと関連付けられたコントラクト名と一致する必要があります。一致していない場合、FolderInst はスキップされ、値はこの FolderInst から使用されません。 このフィールドの値を [any] にして、この FolderInst がすべてのコントラクトで使用されるようになります。
graphNameOrLbl	パラメータの解決時に一致する FolderInst を検索します。FolderInst をパラメータ解決で使用するには、このプロパティはサービス グラフ名と一致する必要があります。一致していない場合、FolderInst はスキップされ、値はこの FolderInst から使用されません。 この FolderInst がすべてのサービス グラフで使用されるようにするには、このフィールドの値を [any] にできます。
nodeNameOrLbl	パラメータの解決時に一致する FolderInst を検索します。FolderInst をパラメータ解決で使用するには、このプロパティはノード名と一致する必要があります。一致していない場合、FolderInst はスキップされ、値はこの FolderInst から使用されません。 このフィールドの値を [any] にして、この FolderInst がサービス グラフ内のすべてのノードで使用されるようになります。

ParamInst

ParamInst はコンフィギュレーションパラメータの基本単位です。ParamInst は単一のコンフィギュレーションパラメータを定義します。FolderInst と同様、各 ParamInst に対してデバイス仕様内に対応する MParam が存在する必要があります。APIC は仕様を検証して、パッケージ内に ParamInst に対応する MParam が存在することを確認します。ParamInst の値は、対応する MParam 内で指定される検証メソッドを使用して検証されます。ParamInst には、次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。

属性	説明
Value	特定のコンフィギュレーションアイテムの値を保持します。値は MParam ではサポートされません。

CfgRelInst

CfgRelInst には、次の属性があります。

属性	説明
Key	コンフィギュレーションアイテムのタイプを定義します。キーは、デバイスパッケージで定義されており、上書きすることはできません。キーは、検証だけでなく一致基準として使用されます。
Value	ターゲット FolderInst のパスを保持します。

コンフィギュレーションパラメータを持つアプリケーション EPG の XML POST の例

次の XML の例は、デバイスパッケージ内のコンフィギュレーションパラメータを示しています。

```
<fvAEPg dn="uni/tn-acme/ap-myApp/epg-app" name="app">
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any" key="Monitor"
    name="monitor1">
    <vnsRsFolderInstToMFolder tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Monitor"/>
    <vnsParamInst name="weight" key="weight" value="10"/>
  </vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any" key="Service"
    name="Service1">
    <vnsParamInst name="servicename" key="servicename" value="crpvgrtst02-8010"/>
    <vnsParamInst name="servicetype" key="servicetype" value="TCP"/>
    <vnsParamInst name="servername" key="servername" value="s192.168.100.100"/>
    <vnsParamInst name="serveripaddress" key="serveripaddress" value="192.168.100.100"/>
    <vnsParamInst name="serviceport" key="serviceport" value="8080"/>
    <vnsParamInst name="svrtimeout" key="svrtimeout" value="9000" />
    <vnsParamInst name="clttimeout" key="clttimeout" value="9000" />
    <vnsParamInst name="usip" key="usip" value="NO" />
    <vnsParamInst name="useproxyport" key="useproxyport" value="" />
    <vnsParamInst name="cip" key="cip" value="ENABLED" />
    <vnsParamInst name="cka" key="cka" value="NO" />
    <vnsParamInst name="sp" key="sp" value="OFF" />
    <vnsParamInst name="cmp" key="cmp" value="NO" />
    <vnsParamInst name="maxclient" key="maxclient" value="0" />
    <vnsParamInst name="maxreq" key="maxreq" value="0" />
    <vnsParamInst name="tcpb" key="tcpb" value="NO" />
    <vnsCfgRelInst name="MonitorConfig" key="MonitorConfig" targetName="monitor1"/>
  </vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any" key="Network"
    name="Network">
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any" key="vip">
```

```

name="vip">
  <vnsParamInst name="vipaddress1" key="vipaddress" value="10.10.10.200"/>
  </vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any"
devCtxLbl="C1" key="snip" name="snip1">
    <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.200"/>
    </vnsFolderInst>
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any"
devCtxLbl="C2" key="snip" name="snip2">
      <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any" key="Network"
name="Network">
        <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any" key="vip"
name="vip">
          <vnsParamInst name="vipaddress1" key="vipaddress" value="10.10.10.100"/>
          </vnsFolderInst>
          <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
devCtxLbl="C1" key="snip" name="snip1">
            <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.100"/>
            </vnsFolderInst>
            <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
devCtxLbl="C2" key="snip" name="snip2">
              <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.101"/>
              </vnsFolderInst>
              <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
devCtxLbl="C3" key="snip" name="snip3">
                <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.102"/>
                </vnsFolderInst>
              </vnsFolderInst>
            </vnsFolderInst>
          </vnsFolderInst>
        <!-- SLB Configuration -->
        <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any" key="VServer"
name="VServer">
          <!-- Virtual Server Configuration -->
          <vnsParamInst name="port" key="port" value="8010"/>
          <vnsParamInst name="vip" key="vip" value="10.10.10.100"/>
          <vnsParamInst name="vsservername" key="vsservername" value="crpvgrtst02-vip-8010"/>
          <vnsParamInst name="servicename" key="servicename" value="crpvgrtst02-8010"/>
          <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
key="VServerGlobalConfig" name="VServerGlobalConfig">
            <vnsCfgRelInst name="ServiceConfig" key="ServiceConfig" targetName="Service1"/>
            <vnsCfgRelInst name="VipConfig" key="VipConfig" targetName="Network/vip"/>
          </vnsFolderInst>
        </vnsFolderInst>
      </fvAEPg>

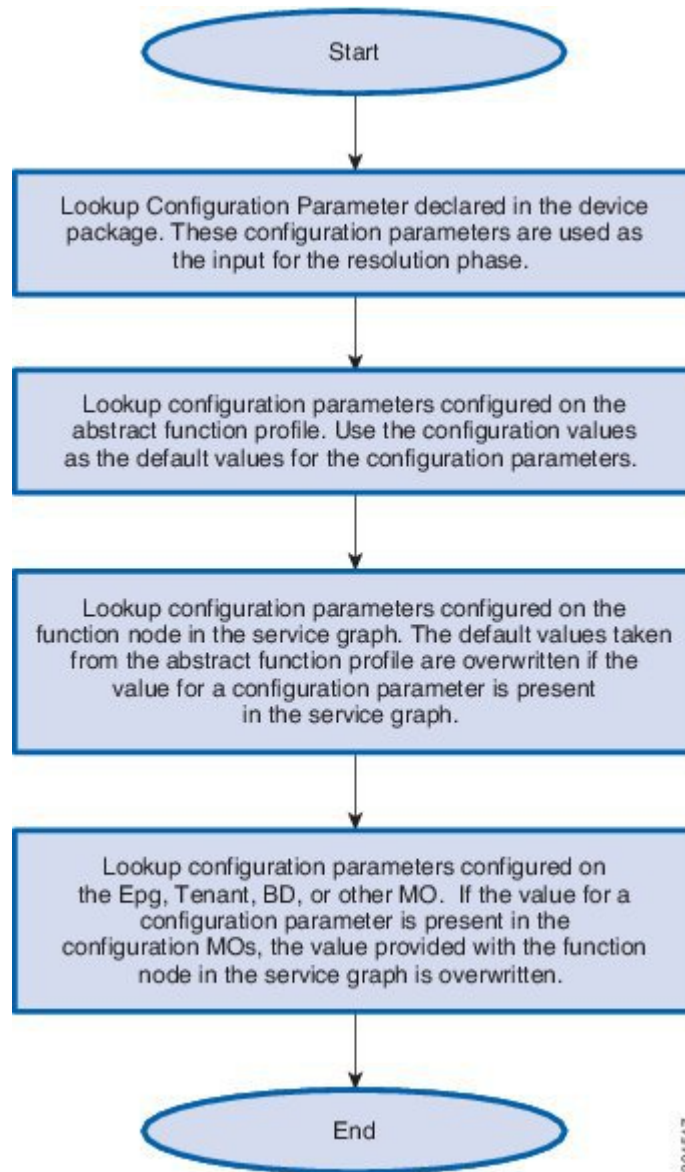
```

パラメータ解決

グラフインスタンス作成時に、Application Policy Infrastructure Controller (APIC) はサービスグラフの各機能に対してコンフィギュレーションパラメータを解決します。解決が完了すると、パラメータ値がデバイススクリプトに渡されます。デバイススクリプトはこれらのパラメータ値を使用してサービスアプライアンス上でサービスを設定します。

次のフローチャートは、パラメータの解決手順について説明しています。

図 6: パラメータ解決



パラメータ解決時の MO の検索

Application Policy Infrastructure Controller (APIC) は、コンフィギュレーションパラメータを取得する適切なコンフィギュレーション MO の検出に 2 つの主なコンストラクトを使用します。

RsScopeToTerm

機能ノードまたは AbsFolder に対する RsScopeToTerm 関係は、グラフに対するパラメータを持つコンフィギュレーション MO と接続されるサービス グラフの端末ノードを示します。APIC は、グラフ コンフィギュレーション パラメータを検出するために、RsScopeToTerm 内の指定の端末ノードと接続されたコンフィギュレーション MO を使用します。

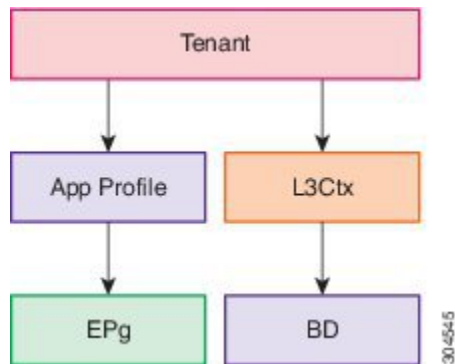
指定された RsScopeToTerm コンフィギュレーションがない場合、APIC はデフォルトでプロバイダー EPG に接続された端末を使用します。

ScopedBy 属性

ScopedBy 属性はパラメータの解決に使用する開始 MO の検出に使用されます。たとえば、scopedBy に「EPG」の値がある場合、APIC はエンドポイントグループからパラメータ解決を開始します。APIC は、階層を上ってパラメータを解決し、アプリケーション プロファイルの次にテナントに上ってコンフィギュレーション パラメータを解決します。

次の図は、APIC MO の階層を示しています。

図 7: APIC MO の階層





第 9 章

サービス グラフの使用

- [サービス グラフとコントラクトの関連付け, 75 ページ](#)
- [CLI コマンドを使用した設定, 76 ページ](#)
- [REST API を使用した設定, 77 ページ](#)

サービス グラフとコントラクトの関連付け

サービス グラフとコントラクトを関連付けることができます。

はじめる前に

- [コントラクトを設定します。 \[コントラクトの設定, \\(18 ページ\\)\]\(#\) を参照してください。](#)

- ステップ 1** メニュー バーで、[TENANTS] タブをクリックします。 [Tenant] ウィンドウが表示されます。
- ステップ 2** [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 3** [Application Profiles] ブランチを展開して、そのテナントのアプリケーションプロファイルを表示します。
- ステップ 4** サービス グラフとコントラクトを関連付けるアプリケーションプロファイルのブランチを展開します。そのアプリケーションプロファイルのリストが表示されます。
- ステップ 5** 対象のアプリケーション EPG を右クリックします。 ポップアップ メニューが表示されます。
- ステップ 6** [Add Provided Contract] を選択します。 [ADD PROVIDED CONTRACT] ダイアログ ボックスが表示されます。
- ステップ 7** [Contract] のドロップダウン メニューからコントラクトを選択します。
- ステップ 8** [Submit] をクリックします。 Application Policy Infrastructure Controller (APIC) は、アプリケーション EPG をサービス グラフの出力端末に接続します。 抽象フォルダ/パラメータが左ペイン ([L4-L7 Services] > [Service Graphs] > [GraphName]) から追加されます。
- ステップ 9** ツリーを展開して、機能ノードの設定パラメータを表示および編集できます。 [ABSTRACTFOLDER/PARAM NAME] カラムおよび [VALUE] カラムをクリックして必要な情報を入力することで、パラメータ名および

パラメータ値を指定できます。抽象フォルダ/パラメータが左ペイン ([L4-L7 Services] > [Service Graphs] > [GraphName]) から追加されます。

- ステップ 10** [Submit] をクリックします。[ADD PROVIDED CONTRACT] ダイアログ ボックスが閉じ、コントラクトがアプリケーション EPG に追加されます。[Contracts] ウィンドウの [Provided Contracts] タブで、提供されるコントラクトを表示できます。
次に、この EPG 上でこのコントラクトをコンシューマ コントラクトにします。このことは、コントラクトがプロバイダー EPG とコンシューマ EPG とで結ばれることを意味します。
- ステップ 11** 異なる EPG を右クリックします。ポップアップ メニューが表示されます。
- ステップ 12** [Add Consumed Contract] を選択します。[ADD CONSUMED CONTRACT] ダイアログ ボックスが表示されます。このプロセスはサービス グラフをインスタンス化します。
- ステップ 13** [Contract] ドロップダウン リストで、コントラクトを選択します。[ADD CONSUMED CONTRACT] ダイアログ ボックスが変更されて、[Parameters Configurations] セクションが表示されます。
- ステップ 14** ツリーを展開して、消費コントラクトの設定パラメータを表示および編集できます。[ABSTRACT FOLDER/PARAM NAME] カラムおよび [VALUE] カラムをクリックして必要な情報を入力することで、パラメータ名およびパラメータ値を指定できます。
- ステップ 15** [SUBMIT] をクリックします。[ADD CONSUMED CONTRACT] ダイアログ ボックスが閉じ、消費コントラクトのサブジェクトが作成されます。
- ステップ 16** [TENANTS] ウィンドウの [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 17** [Security Policies] ブランチを展開します。
- ステップ 18** [Contracts] ブランチを展開して、作成したコントラクトを表示します。
- ステップ 19** [contract] を展開し、その下のオブジェクトをクリックします。作業ペインで、前に作成したサービス グラフを関連付け、[SUBMIT] をクリックします。
- ステップ 20** [SUBMIT] をクリックしてグラフにコントラクトをアタッチします。

CLI コマンドを使用した設定

設定例は次のとおりです。

```
admin@apic1:contracts> moconfig running

# contract
cd '/aci/tenants/acme/security-policies/contracts'
mcreate 'webCtrct'
cd 'webCtrct'
moset scope 'tenant'
moconfig commit

# contract-subject
cd '/aci/tenants/acme/security-policies/contracts/webCtrct/subjects'
mcreate 'http'
moconfig commit

# subj-graphatt
cd '/aci/tenants/acme/security-policies/contracts/webCtrct/subjects/http'
mcreate subj-graphatt
```

```

cd 'subj-graphatt'
moset name 'WebGraph'
moconfig commit

# vz-rssubjfiltatt
cd '/aci/tenants/acme/security-policies/contracts/webCtrct/subjects/http/common-filters'
mcreate 'wildcard'
moconfig commit

```

REST API を使用した設定

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <!--L3 Network-->
    <fvCtx name="MyNetwork"/>
      <!-- Bridge Domain for MySrvr EPG -->
      <fvBD name="MySrvrBD">
        <fvRsCtx tnFvCtxName="MyNetwork" />
        <fvSubnet ip="10.10.10.10/24">
        </fvSubnet>
      </fvBD>
      <!-- Bridge Domain for MyClnt EPG -->
      <fvBD name="MyClntBD">
        <fvRsCtx tnFvCtxName="MyNetwork" />
        <fvSubnet ip="20.20.20.20/24">
        </fvSubnet>
      </fvBD>
      <fvAp dn="uni/tn-acme/ap-MyAP" name="MyAP">
        <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MyClnt" name="MyClnt">
          <fvRsBd tnFvBDName="MySrvrBD" />
          <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
          <fvRsProv tnVzBrCPName="webCtrct" /> </fvRsProv>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]" encap="vlan-202"/>
        <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]" encap="vlan-202"/> </fvAEPg>
        <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MySRVR" name="MySRVR">
          <fvRsBd tnFvBDName="MyClntBD" />
          <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
          <fvRsCons tnVzBrCPName="webCtrct" /> </fvRsCons>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]" encap="vlan-203"/>
        <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]" encap="vlan-203"/>
        </fvAEPg>
      </fvAp>
    </fvTenant>
  </polUni>

```

セキュリティポリシーを作成するための REST API による設定

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vzFilter name="HttpIn">
      <vzEntry name="e1" prot="6" dToPort="80"/>
    </vzFilter>
    <vzBrCP name="webCtrct">
      <vzSubj name="http">
        <vzRsSubjFiltAtt tnVzFilterName="HttpIn"/>
      </vzSubj>
    </vzBrCP>
  </fvTenant>
</polUni>

```




第 10 章

サービス グラフのモニタリング

- [サービス グラフ インスタンスのモニタリング, 79 ページ](#)
- [サービス グラフ エラーのモニタリング, 81 ページ](#)
- [サービス グラフ エラーの解決, 81 ページ](#)
- [仮想デバイスのモニタリング, 87 ページ](#)
- [CLI コマンドを使用した設定, 88 ページ](#)

サービス グラフ インスタンスのモニタリング

サービス グラフを設定し、エンドポイント グループ (EPG) およびコントラクトにグラフをアタッチした後は、サービス グラフ インスタンスをモニタできます。モニタリングには、グラフ インスタンスの状態、グラフ インスタンスの機能、機能に割り当てられたリソース、および機能に指定されたパラメータの表示が含まれます。

- ステップ 1** メニュー バーで、[TENANTS] タブをクリックします。[Tenant] ウィンドウが表示されます。
- ステップ 2** サブメニュー バーで、サービス グラフをモニタするテナントのタブをクリックします。選択したテナントの [Tenant] ウィンドウが [Work] ペインに表示されます。
- ステップ 3** [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 4** [Navigation] ペインで、[L4-L7 Services] ブランチを展開します。
- ステップ 5** [Navigation] ペインで、[Deployed Service Graphs] をクリックします。[Work] ペインは、アクティブなサービス グラフ インスタンスに関する次の情報を表示します。

名前	説明
[Contract] カラム	サービス グラフに表示されるコントラクトの名前。

名前	説明
[State] カラム	サービス グラフの状態。[applied]の状態は、グラフが適用され、グラフポリシーがファブリックおよびサービスデバイス内でアクティブであることを意味します。
[Service Graph] カラム	サービス グラフの名前。
[Contained By] カラム	サービス グラフを含むネットワークの名前。
[Function Nodes] カラム	グラフに接続されている機能ノードの名前。

ステップ 6 [Deployed Service Graphs] ブランチを展開します。アクティブなサービス グラフ インスタンスがブランチの下にリストされます。

ステップ 7 サービス グラフ インスタンスをクリックして、[Work] ペインにそのインスタンスに関する追加情報を表示します。デフォルト ビューはグラフのトポロジです。[Work] ペインのタブのいずれかをクリックして、そのグラフのビューを変更できます。

ステップ 8 グラフ インスタンスのいずれかのブランチを展開します。グラフ インスタンスの機能は、インスタンスの下に表示されます。

ステップ 9 機能をクリックして、[Work] ペインにその機能に関する追加情報を表示します。デフォルト ビューはその機能のポリシーです。[Work] ペインのタブのいずれかをクリックして、その機能のビューを変更できます。[Work] ペインには、ポリシーに関する次の情報が表示されます。

名前	説明
[POLICY] タブ	機能のプロパティ、機能に割り当てられたリソース、および機能のパラメータ。
[OPERATIONAL] タブ	機能のヘルス スコア。ヘルス スコアはApplication Policy Infrastructure Controller (APIC) によって計算され、サービスデバイスから受信した機能に関する情報に基づいています。
[FAULTS] タブ	機能ノードで生じている問題。
[HISTORY] タブ	機能ノードで発生したイベントの履歴。

名前	説明
[POLICY] タブ	機能のプロパティ、機能に割り当てられたリソース、および機能のパラメータ。
[FAULTS] タブ	機能ノードで生じている問題。
[HISTORY] タブ	機能ノードで発生したイベントの履歴。

ステップ 10 [Navigation] ペインで、[Deployed Device Cluster] をクリックします。[Work] ペインにデバイスのインスタンスに関する情報が表示されます。

サービス グラフ エラーのモニタリング

サービスグラフを設定し、EPGおよびコントラクトにグラフをアタッチした後は、サービスグラフエラーをモニタできます。

-
- ステップ1 メニューバーで、[TENANTS] タブをクリックします。[Tenant] ウィンドウが表示されます。
 - ステップ2 サブメニューバーで、サービスグラフをモニタするテナントのタブをクリックします。選択したテナントの[Tenant] ウィンドウが[Work] ペインに表示されます。
 - ステップ3 [Navigation] ペインで、テナントのブランチを展開します。
 - ステップ4 [L4-L7 Services] ブランチを展開します。
 - ステップ5 [Graphs Instances] ブランチを展開します。アクティブなサービスグラフインスタンスがこのブランチの下にリストされます。
 - ステップ6 エラーを表示するグラフインスタンスのブランチを展開します。グラフインスタンスの機能は、インスタンスの下に表示されます。
 - ステップ7 機能のいずれかをクリックします。デフォルトで、[Work] ペインはその機能のポリシーを示します。
 - ステップ8 [Work] ペインの[FAULTS] タブをクリックします。[Work] ペインが機能ノードのエラーを表示します。エラーおよびその解決策については、[サービスグラフエラーの解決](#)、(81ページ) を参照してください。
-

サービス グラフ エラーの解決

1つ以上のサービスグラフエラーを発見した場合、問題の解決はエラーによって異なります。次の表は、エラーの説明とエラーを解決する方法を説明しています。

表1: コネクタのエラー

エラー	CLI ラベル	説明と解決法
missing-connection	connection associated with a connector not found	グラフ コネクタの設定が無効です。コネクタに関連付けられた接続が見つかりませんでした。
missing-nodeinst	NodeInst associated with a connector not found	グラフ コネクタの設定が無効です。コネクタに関連付けられたNodeInstが見つかりませんでした。

エラー	CLI ラベル	説明と解決法
conn-nonrenderable	Graph connector could not be rendered.	グラフ コネクタの設定が無効です。グラフをレンダリングできませんでした。
invalid-bd	BD associated with a connector is not valid	グラフ コネクタの設定が無効です。コネクタの関連ブリッジドメインが無効です。
invalid-ctx	Ctx associated with a connector is not valid.	グラフ コネクタの設定が無効です。コネクタの関連するCtxが無効です。
missing-peer-conn	Peer connector associated with a connector not found.	グラフ コネクタの設定が無効です。接続のピア コネクタが見つかりませんでした。

表 2: *AbsGraph* および *GraphInst* エラー

エラー	CLI ラベル	説明と解決法
invalid-abstract-graph-config	invalid abstract graph config	抽象グラフ設定が無効です。
missing-mandatory-param	mandatory param not found	必要な設定パラメータが解決できませんでした。パッケージの必須パラメータをチェックし、 <i>AbsGraph</i> にパラメータがあることを確認します。
param-cardinality-error	invalid param cardinality	コンフィギュレーションパラメータは、濃度の要件を満たしていません。 cardinality=n を指定しないでパラメータの複数のインスタンスが指定されているかどうかを確認します。
epp-download-failure	epp download failure	グラフ ポリシーがスイッチのダウンロードに失敗しました。
param-duplicate-name-failure	duplicate param name	同じ名前のパラメータの複数の同一コピーが検出されました。
id-allocation-failure	id allocation failure	一意のネットワーク リソース (VLAN VXLAN) を割り当てるできませんでした。
missing-ldev	No cluster found	クラスタが見つかりませんでした。

エラー	CLI ラベル	説明と解決法
context-cardinality-violation-failure	invalid cluster context cardinality	クラスタは必要なテナント機能（マルチテナントまたはシングルテナント）をサポートしていません。
function-type-mismatch-failure	invalid function type	機能タイプは、選択したデバイスクラスタでサポートされません。AbsNode 機能タイプと解決された LDevVip 機能タイプが一致するか確認します。
invalid-abstract-graph-config-param	invalid abstract graph config param	抽象グラフ コンフィギュレーションパラメータが無効です。
missing-mparam	No parameter definition found	必要なパラメータ定義が見つかりませんでした。
missing-abs-graph	no abs graph found	抽象グラフ設定がグラフ インスタンスにありません。
invalid-param-config	invalid param config	パラメータ設定が無効です。
invalid-param-scope	invalid parameter scope	パラメータ スコープが無効です。AbsGraph の vnsRsScopeToTerm パラメータが正しいかどうか確認します。
invalid-ldev	Invalid cluster	クラスタ設定が無効です。解決した LDevVip のステータスを確認して、エラーを解決します。
missing-tenant	no tenant found	グラフに対してテナントが見つかりませんでした。
internal-error	internal error	内部エラーがグラフ処理中に発生しました。
resource-allocation-failure	resource allocation failure	グラフ処理中に必要なリソースを割り当てることができませんでした。
missing-abs-function	no abstract function found	抽象機能の定義が見つかりません。
param-validation-failed	param validation failure	コンフィギュレーションパラメータ値が無効です。

エラー	CLI ラベル	説明と解決法
missing-mconn	No connector found	必要なコネクタが見つかりませんでした。
cdev-missing-mgmt-ip	no mgmt ip found for cdev	具象デバイスに対して管理 IP アドレスが見つかりませんでした。 vnsCMgmt が解決する vnsCDev に存在するかどうかを確認します。
invalid-graphinst	invalid graphinst config	グラフ インスタンスが無効です。
missing-interface	no interface found	インターフェイスが見つかりませんでした。
missing-bd	no bd found	ブリッジドメインが見つかりませんでした。
missing-terminal	Terminal node is missing a terminal	端末ノードに端末がありません。 端末ノードの設定を確認してください。
missing-namespace	no vlan/vxlan namespace found	VLAN または VXLAN の割り当てに必要なネームスペースが見つかりません。 解決された fvnsVlanInstp と関係がある phyDomp パラメータまたは vmmDomp パラメータが解決された vnsLDevVip に設定されていることを確認します。
missing-mfunc	No function found in device package	デバイス パッケージで必要な機能が見つかりません。 パッケージ内にすべての AbsNode 機能タイプがあることを確認します。
missing-lif	no cluster interface found	必要なクラスタ インターフェイスが見つかりませんでした。 vnsLDevVip の vnsLif パラメータが正しく設定されていることを確認します。

エラー	CLI ラベル	説明と解決法
invalid-absfunc-profile	Abstract Function Profile config is invalid	抽象機能のプロファイル設定が無効です。このエラーは、プロファイルで指定されている無効なコンフィギュレーションパラメータが要因として考えられます。
missing-cdev	No device found	具象デバイスがクラスタ内に見つかりませんでした。有効な vnsCDev が解決された vnsLDevVip の下に存在することを確認してください。
inappropriate-devfolder	Illegal folder in configuration	対応するフォルダがデバイスパッケージで見つかりませんでした。
invalid-devctx	Device context is not legal for this folder	デバイス パッケージではこのフォルダにデバイス コンテキストを指定することはできません。
insufficient-devctx	Folder must have one value for each associated CDev	フォルダは具象デバイスに固有です。フォルダは、各具象デバイスに対して少なくとも1つの値を持つ必要があります。
cdev-missing-cif	No interface defined	具象デバイスには少なくとも1つのインターフェイスを定義する必要があります。
cdev-missing-pathinfo	Missing path for interface	物理サービス アプライアンスでは、インターフェイスがどのリーフポートに接続されているかを把握する必要があります。 vnsCifPathAtt パラメータが、解決された vnsCDev の下のすべての vnsCif に存在することを確認します。

エラー	CLI ラベル	説明と解決法
missing-cif	Device interfaces does not match cluster	デバイスインターフェイスは、クラスタに設定されているインターフェイスに一致させる必要があります。 vnsCif パラメータおよび vnsLif パラメータが、解決された vnsLDevVip の下に存在することを確認します。
ldevvip-missing-mgmt-ip	No Mgmt ip found for LDevVip	LDevVip に対して管理 IP アドレスが見つかりませんでした。
lif-invalid-Mif	Lif has an invalid MifLbl	Lif に含まれる MifLbl がデバイスパッケージに存在しません。
lif-invalid-Cif	Lif has an invalid Cif	Lif に含まれる Cif がありません。 具象デバイスおよび Cif の設定を確認します。
missing-function-node	Abstract graph missing function node	抽象グラフには、少なくとも 1 つの機能ノードが存在する必要があります。
graph-loop-detected	Abstract graph config has a loop	抽象グラフ設定が無効です。設定にループがあります。
gothrough-routing-enabled-both	Both the legs of go through node has routing enabled	通過ノードの両方のレッグでルーティングが有効になっています。
invalid-terminal-nodes	Abstract graph has invalid number of terminal nodes	抽象グラフは少なくとも 2 つの端末ノードを持つ必要があります。
missing-ldev-ctx	No device context found for LDev	デバイス クラスタのデバイスコンテキストが見つかりませんでした。 vnsLDevCtx にコントラクト、グラフおよびノードに一致する値があることを確認します。
arp-flood-enabled	ARP flood is enabled on the management end point group	ARP フラッディングは管理エンドポイントのグループに対して無効です。
folderinst-validation-failed	FolderInst has key, that is not found in MFolder	FolderInst のキーおよび値は MFolder 仕様を尊重する必要があります。

エラー	CLI ラベル	説明と解決法
paraminst-validation-failed	ParamInst has key and/or value, that are not found in MParam	ParamInst のキーおよび値は MParam 仕様を尊重する必要があります。
invalid-mfolder	FolderInst points to an invalid MFolder	FolderInst は有効な MFolder をポイントする必要があります。
invalid-mparam	ParamInst points to an invalid MParam	ParamInst は有効な MParam をポイントする必要があります。
devfolder-validation-failed	DevFolder has key, that is not found in MFolder	DevFolders のキーおよび値は MFolder 仕様を尊重する必要があります。
devparam-validation-failed	DevParam has key and/or value, that are not found in MParam	DevParam のキーおよび値は MParam 仕様を尊重する必要があります。
cdev-missing-virtual-info	Virtual Object Info is missing in CDev	LDevVip のタイプが Virtual の場合は仮想オブジェクト情報を指定する必要があります。

仮想デバイスのモニタリング

サービスグラフを設定し、エンドポイントグループ（EPG）およびコントラクトにグラフをアタッチした後は、テナントの仮想デバイスをモニタできます。仮想デバイスをモニタリングすると、どのデバイスクラスタが使用中か、どのVLANがデバイスクラスタ用に設定されているか、デバイスクラスタ内でデバイスに渡されるパラメータ、デバイスクラスタ内のデバイスの統計、およびデバイスクラスタ内のデバイスの健全性を確認できます。

- ステップ 1 メニューバーで、[TENANTS] タブをクリックします。[Tenant] ウィンドウが表示されます。
- ステップ 2 サブメニューバーで、サービスグラフをモニタするテナントのタブをクリックします。選択したテナントの [Tenant] ウィンドウが [Work] ペインに表示されます。
- ステップ 3 [Navigation] ペインで、テナントのブランチを展開します。
- ステップ 4 [L4-L7 Services] ブランチを展開します。
- ステップ 5 [Virtual Devices] ブランチを展開します。仮想デバイスが、このブランチの下に表示されます。
- ステップ 6 仮想デバイスのいずれかをクリックします。デフォルトで、[Work] ペインはその仮想デバイスのポリシーを示します。ビューを変更するには、[Work] ペインのタブをクリックします。タブは、仮想デバイスに関する以下の情報を表示します。

タブ	説明
[POLICY] タブ	使用されているデバイス クラスタ、デバイス クラスタ内で設定された VLAN、およびデバイス クラスタ内のデバイスに渡されたパラメータ。
[OPERATIONAL] タブ	デバイス クラスタ内のさまざまなデバイスから受信される統計。
[HEALTH] タブ	デバイス クラスタ内のデバイスの健全性。

CLI コマンドを使用した設定

サービス グラフの場合：

```
# 14-17-service-graph
cd '/aci/tenants/acme/14-17-services/service-graphs'
mocreate 'WebGraph'
moconfig commit

# consumer-terminal-node
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/consumer-terminal-nodes'
mocreate 'Output1'
moconfig commit

# terminal-connector
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/consumer-terminal-nodes/Output1'
mocreate terminal-connector
cd 'terminal-connector'
moset name 'C6'
moconfig commit

# provider-terminal-node
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/provider-terminal-nodes'
mocreate 'Input1'
moconfig commit

# terminal-connector
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/provider-terminal-nodes/Input1'
mocreate terminal-connector
cd 'terminal-connector'
moset name 'C1'
moconfig commit

# function-node
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes'
mocreate 'FW'
cd 'FW'
moset function-type 'gothrough'
moset function-name '14-17-services/packages/CISCO-ASA-1.0.1.8/functions/Firewall'
moconfig commit

# device-config
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW'
mocreate device-config
moconfig commit

# folder
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config'
mocreate 'ExIntfCfg'
cd 'ExIntfCfg'
moset cardinality 'n'
```

```
moset faultcode '0'
moset key 'InterfaceConfig'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config/ExIntfCfg'
mcreate param 'mask2'
cd 'param-mask2'
moset value '255.255.255.0'
moset faultcode '0'
moset key 'Mask'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config/ExIntfCfg'
mcreate param 'address2'
cd 'param-address2'
moset value '30.30.30.201'
moset faultcode '0'
moset key 'Address'
moconfig commit

# folder
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config'
mcreate 'access-list-foo'
cd 'access-list-foo'
moset faultcode '0'
moset key 'AccessList'
moconfig commit

# folder
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config/access-list-foo'
mcreate folder 'entry1'
cd 'folder-entry1'
moset faultcode '0'
moset key 'AccessControlEntry'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config/access-list-foo/folder-entry1'
mcreate param 'ip'
cd 'param-ip'
moset value 'ip'
moset faultcode '0'
moset key 'protocol'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config/access-list-foo/folder-entry1'
mcreate param 'order1'
cd 'param-order1'
moset value '10'
moset faultcode '0'
moset key 'order'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config/access-list-foo/folder-entry1'
mcreate param 'action-permit'
cd 'param-action-permit'
moset value 'permit'
moset faultcode '0'
moset key 'action'
moconfig commit

# folder
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/device-config'
```

```
mocreate 'InIntfCfg'  
cd 'InIntfCfg'  
moset cardinality 'n'  
moset faultcode '0'  
moset key 'InterfaceConfig'  
moconfig commit  
  
# param  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/device-config/InIntfCfg'  
mocreate param 'mask1'  
cd 'param-mask1'  
moset value '255.255.255.0'  
moset faultcode '0'  
moset key 'Mask'  
moconfig commit  
  
# param  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/device-config/InIntfCfg'  
mocreate param 'address1'  
cd 'param-address1'  
moset value '40.40.40.102'  
moset faultcode '0'  
moset key 'Address'  
moconfig commit  
  
# function-config  
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW'  
mocreate function-config  
moconfig commit  
  
# folder  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-config'  
mocreate 'InIntfCfgRelFolder'  
cd 'InIntfCfgRelFolder'  
moset faultcode '0'  
moset key 'InIntfConfigRelFolder'  
moconfig commit  
  
# cfg-rel  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-config/InIntfCfgRelFolder'  
mocreate cfg-rel 'InIntfCfgRel'  
cd 'cfg-rel-InIntfCfgRel'  
moset faultcode '0'  
moset key 'InIntfConfigRel'  
moset targetname 'InIntfCfg'  
moconfig commit  
  
# folder  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-config'  
mocreate 'ExIntfCfgRelFolder'  
cd 'ExIntfCfgRelFolder'  
moset faultcode '0'  
moset key 'ExIntfConfigRelFolder'  
moconfig commit  
  
# cfg-rel  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-config/ExIntfCfgRelFolder'  
mocreate cfg-rel 'ExIntfCfgRel'  
cd 'cfg-rel-ExIntfCfgRel'  
moset faultcode '0'  
moset key 'ExIntfConfigRel'  
moset targetname 'ExIntfCfg'  
moconfig commit  
  
# folder  
cd  
'/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-config'
```

```
mocreate 'access-group-foo'
cd 'access-group-foo'
moset faultcode '0'
moset key 'AccessGroup'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/function-config/access-group-foo'
mocreate param 'dir'
cd 'param-dir'
moset value 'in'
moset faultcode '0'
moset key 'direction'
moconfig commit

# param
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/function-config/access-group-foo'
mocreate param 'con1'
cd 'param-con1'
moset value 'external'
moset faultcode '0'
moset key 'connector'
moconfig commit

# cfg-rel
cd
'/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/function-config/access-group-foo'
mocreate cfg-rel 'access-list-foo-rel'
cd 'cfg-rel-access-list-foo-rel'
moset faultcode '0'
moset key 'access_list_name'
moset targetname 'access-list-foo'
moconfig commit

# function-connector
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW'
mocreate function-connector 'external'
cd 'function-connector-external'
moset ctx-terms 'uni/tn-acme/AbsGraph-WebGraph/AbsTermNodeProv-Input1/intmnl'
moset meta-connector 'uni/infra/mDev-CISCO-ASA-1.0.1.8/mFunc-Firewall/mConn-external'
moconfig commit

# function-connector
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW'
mocreate function-connector 'internal'
cd 'function-connector-internal'
moset ctx-terms 'uni/tn-acme/AbsGraph-WebGraph/AbsTermNodeCon-Output1/outtmnl'
moset meta-connector 'uni/infra/mDev-CISCO-ASA-1.0.1.8/mFunc-Firewall/mConn-internal'
moconfig commit

# default-scope-to-term
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW'
mocreate default-scope-to-term
cd 'default-scope-to-term'
moset target-dn 'uni/tn-acme/AbsGraph-WebGraph/AbsTermNodeCon-Output1/outtmnl'
moconfig commit

# function-node
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes'
mocreate 'LoadBalancing'
cd 'LoadBalancing'
moset function-name '14-17-services/packages/Citrix-NetScaler-10.5/functions/LoadBalancing'
moconfig commit

# function-connector
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/LoadBalancing'
mocreate function-connector 'inside'
cd 'function-connector-inside'
moset meta-connector 'uni/infra/mDev-Citrix-NetScaler-10.5/mFunc-LoadBalancing/mConn-internal'
moconfig commit
```

```

# function-connector
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/LoadBalancing'
mcreate function-connector 'outside'
cd 'function-connector-outside'
moset meta-connector 'uni/infra/mDev-Citrix-NetScaler-10.5/mFunc-LoadBalancing/mConn-external'
moconfig commit

# connection
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections'
mcreate 'CON2'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON2'
mcreate
'tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/LoadBalancing/function-connector-outside'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON2'
mcreate
'tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-connector-external'
moconfig commit

# connection
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections'
mcreate 'CON3'
cd 'CON3'
moset unicast-routing 'no'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON3'
mcreate
'tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/FW/function-connector-internal'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON3'
mcreate
'tenants/acme/l4-l7-services/service-graphs/WebGraph/consumer-terminal-nodes/Output1/terminal-connector'
moconfig commit

# connection
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections'
mcreate 'CON1'
cd 'CON1'
moset unicast-routing 'no'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON1'
mcreate
'tenants/acme/l4-l7-services/service-graphs/WebGraph/function-nodes/LoadBalancing/function-connector-inside'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON1'
mcreate
'tenants/acme/l4-l7-services/service-graphs/WebGraph/provider-terminal-nodes/Input1/terminal-connector'
moconfig commit
admin@apic1:WebGraph>
moset meta-connector 'uni/infra/mDev-Citrix-NetScaler-10.5/mFunc-LoadBalancing/mConn-external'
moconfig commit

# connection
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections'
mcreate 'CON2'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/l4-l7-services/service-graphs/WebGraph/connections/CON2'
mcreate

```

```
'tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/LoadBalancing/function-connector-outside'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections/CON2'
mcreate
'tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/function-connector-external'
moconfig commit

# connection
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections'
mcreate 'CON3'
cd 'CON3'
moset unicast-routing 'no'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections/CON3'
mcreate
'tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/FW/function-connector-internal'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections/CON3'
mcreate
'tenants/acme/14-17-services/service-graphs/WebGraph/consumer-terminal-nodes/Output1/terminal-connector'
moconfig commit

# connection
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections'
mcreate 'CON1'
cd 'CON1'
moset unicast-routing 'no'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections/CON1'
mcreate
'tenants/acme/14-17-services/service-graphs/WebGraph/function-nodes/LoadBalancing/function-connector-inside'
moconfig commit

# relation-from-abstract-connection-to-abstract-connectors
cd '/aci/tenants/acme/14-17-services/service-graphs/WebGraph/connections/CON1'
mcreate
'tenants/acme/14-17-services/service-graphs/WebGraph/provider-terminal-nodes/Input1/terminal-connector'
moconfig commit
```

デバイス クラスタの場合 :

```
# device-cluster
cd '/aci/tenants/acme/14-17-services/device-clusters'
mcreate 'Firewall'
cd 'Firewall'
moset function-type 'gothrough'
moset faultcode '0'
moset device-package '14-17-services/packages/CISCO-ASA-1.0.1.8'
moset physical-domain
'fabric/access-policies/physical-and-external-domains/physical-domains/phys'
moset virtual-ip-address '172.23.49.195'
moset port '443'
moset username 'admin'
moconfig commit

# folder
cd '/aci/tenants/acme/14-17-services/device-clusters/Firewall/parameters'
mcreate folder 'Timeouts' 'Timeouts'
cd 'folder-Timeouts-Timeouts'
moset faultcode '0'
moconfig commit

# param
cd
```

```

'/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/parameters/folder-Timeouts-Timeouts'
mcreate param 'Udp' 'Udp'
cd 'param-Udp-Udp'
moset faultcode '0'
moset value '0:35:0'
moconfig commit

# param
cd
'/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/parameters/folder-Timeouts-Timeouts'
mcreate param 'Connection' 'Connection'
cd 'param-Connection-Connection'
moset faultcode '0'
moset value '0:10:0'
moconfig commit

# logical-interface
cd '/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/logical-interfaces'
mcreate 'external'
cd 'external'
moset type 'uni/infra/mDev-CISCO-ASA-1.0.1.8/mIfLbl-external'
moconfig commit

# vns-rscifatt
cd
'/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/logical-interfaces/external/concrete-interfaces'
mcreate 'tenants/acme/l4-l7-services/device-clusters/Firewall/devices/ASA/interfaces/Gig0_0'
moconfig commit

# logical-interface
cd '/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/logical-interfaces'
mcreate 'internal'
cd 'internal'
moset type 'uni/infra/mDev-CISCO-ASA-1.0.1.8/mIfLbl-internal'
moconfig commit

# vns-rscifatt
cd
'/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/logical-interfaces/internal/concrete-interfaces'
mcreate 'tenants/acme/l4-l7-services/device-clusters/Firewall/devices/ASA/interfaces/Gig0_1'
moconfig commit

# concrete-device
cd '/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/devices'
mcreate 'ASA'
cd 'ASA'
moset faultcode '0'
moset management-address '172.23.49.195'
moset management-port '443'
moset username 'admin'
moconfig commit

# folder
cd '/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/devices/ASA/parameters'
mcreate folder 'appInsp' 'ApplicationInspection'
cd 'folder-appInsp-ApplicationInspection'
moset faultcode '0'
moconfig commit

# param
cd
'/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/devices/ASA/parameters/folder-appInsp-ApplicationInspection'
mcreate param 'icmp' 'icmp'
cd 'param-icmp-icmp'
moset faultcode '0'
moset value 'enable'
moconfig commit

# concrete-interface
cd '/aci/tenants/acme/l4-l7-services/device-clusters/Firewall/devices/ASA/interfaces'
mcreate 'Gig0_0'
cd 'Gig0_0'
moset path 'topology/pod-1/paths-101/pathep-[eth1/23]'

```

```
moconfig commit

# concrete-interface
cd '/aci/tenants/acme/14-17-services/device-clusters/Firewall/devices/ASA/interfaces'
mcreate 'Gig0_1'
cd 'Gig0_1'
moset path 'topology/pod-1/paths-101/pathep-[eth1/25]'
moconfig commit

# device-cluster
cd '/aci/tenants/acme/14-17-services/device-clusters'
mcreate 'ADCCluster1'
cd 'ADCCluster1'
moset device-type 'virtual'
moset faultcode '0'
moset device-package '14-17-services/packages/Citrix-NetScaler-10.5'
moset vmm-domain 'vm-networking/policies/vmware/vmm-domains/mininet'
moset virtual-ip-address '172.23.49.175'
moset port '80'
moset username 'nsroot'
moconfig commit

# logical-interface
cd '/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/logical-interfaces'
mcreate 'outside'
cd 'outside'
moset type 'uni/infra/mDev-Citrix-NetScaler-10.5/mIfLbl-outside'
moconfig commit

# vns-rscifatt
cd
'/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/logical-interfaces/outside/concrete-interfaces'
mcreate 'tenants/acme/14-17-services/device-clusters/ADCCluster1/devices/ADC1/interfaces/1_1'
moconfig commit

# logical-interface
cd '/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/logical-interfaces'
mcreate 'inside'
cd 'inside'
moset type 'uni/infra/mDev-Citrix-NetScaler-10.5/mIfLbl-inside'
moconfig commit

# vns-rscifatt
cd
'/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/logical-interfaces/inside/concrete-interfaces'
mcreate 'tenants/acme/14-17-services/device-clusters/ADCCluster1/devices/ADC1/interfaces/1_2'
moconfig commit

# concrete-device
cd '/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/devices'
mcreate 'ADC1'
cd 'ADC1'
moset context-label 'C1'
moset vm-name 'NSVPX-ESX-OVF'
moset vcenter-name 'vcenter1'
moset faultcode '0'
moset management-address '172.23.49.175'
moset management-port '80'
moset username 'nsroot'
moconfig commit

# concrete-interface
cd '/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/devices/ADC1/interfaces'
mcreate '1_1'
cd '1_1'
moset vnic 'Network adapter 1'
moconfig commit

# concrete-interface
cd '/aci/tenants/acme/14-17-services/device-clusters/ADCCluster1/devices/ADC1/interfaces'
mcreate '1_2'
cd '1_2'
moset vnic 'Network adapter 2'
```

```
moconfig commit
```



第 11 章

サービス コンフィギュレーションの管理に対する管理ロールの設定

- [権限について, 97 ページ](#)
- [デバイス管理のロールの設定, 98 ページ](#)
- [サービス グラフ管理のロールの設定, 98 ページ](#)
- [デバイス パッケージのアップロードのロールの設定, 98 ページ](#)
- [デバイス クラスタのエクスポートのロールの設定, 98 ページ](#)

権限について

Application Policy Infrastructure Controller (APIC) で設定したロールに権限を付与できます。権限は、ロールが実行できるタスクを決定します。管理者ロールには次の特権を付与できます。

特権	説明
nw-svc-policy	ネットワーク サービス ポリシー権限では次を実行できます。 <ul style="list-style-type: none">• サービス グラフの作成• アプリケーション EPG およびコントラクトへのサービス グラフのアタッチ• サービス グラフのモニタ

特権	説明
nw-svc-device	ネットワーク サービス デバイス 権限では次を実行できます。 <ul style="list-style-type: none"> • デバイス クラスタの作成 • 具象デバイスの作成 • デバイス コンテキストの作成



(注) インフラストラクチャ管理者のみがデバイス パッケージを APIC にアップロードできます。

デバイス管理のロールの設定

デバイスを管理するためのロールを有効化するには、そのロールに次の特権を付与する必要があります。

- nw-svc-device

サービス グラフ管理のロールの設定

サービス グラフを管理するためのロールを有効化するには、そのロールに次の特権を付与する必要があります。

- nw-svc-policy

デバイス パッケージのアップロードのロールの設定

デバイス パッケージは、APIC インフラ管理者特権でのみアップロードできます。インフラ管理者はデバイス パッケージをアップロードします。他のすべてのテナント管理者はデバイス パッケージに対して読み取り専用アクセスを持ちます。テナント管理者は、デバイス パッケージで使用可能なさまざまな機能にアクセスできます。

デバイス クラスタのエクスポートのロールの設定

デバイス クラスタは、テナント間のデバイス クラスタ 共有を有効化するためにエクスポートできます。nw-device ロールを持つテナントはデバイス クラスタを作成できます。デバイス クラスタ

を所有するテナントがこれらを別のテナントと共有したい場合、共有には **nw-svc-devshare** 特権が必要です。

nw-svc-devshare 特権を使用すると、テナントはデバイス クラスタをエクスポートできます。



(注) インポートされたデバイス クラスタを使用できるようにするには、インポートされたデバイス クラスタを持つ他のテナントが **nw-svc-policy** 特権を持つ必要があります。



第 12 章

自動化の開発

- [REST API について](#), 101 ページ
- [REST API の使用例](#), 102 ページ

REST API について

自動化は、Application Policy Infrastructure Controller (APIC) のノースバウンド Representational State Transfer (REST) API を使用します。APIC UI で実行できる処理はすべて、ノースバウンド API を使用した XML ベースの REST POST または Python スクリプトを使用して実行できます。たとえば、これらの API 経由でのイベントのモニタ、EPG のダイナミックな有効化、およびポリシーの追加などを実行できます。

また、ノースバウンド REST API を使用して、デバイスがオンボードになったことの通知や、エラーをモニタできます。いずれの場合も、モニタリングを特定のアクションをトリガーするイベントとして使用できます。たとえば、特定のアプリケーション層で発生したエラーを検出し、接続の切断がありリーフ ノードがダウンした場合、これらのアプリケーションを他の場所に再展開するアクションをトリガーできます。パケットドロップが検出された特定のコントラクトがある場合、これらのコントラクトの複数のコピーを特定のアプリケーション上で有効化できます。また、レポートされた問題に基づいて特定のカウンタをモニタできる統計モニタリングポリシーを使用できます。

『Cisco APIC REST API User Guide』には、API の使用方法の情報に加えて、ノースバウンド REST API のリストが含まれます。これらの API を使用して、APIC 上でサービスを設定するために独自の統合ツールを記述できます。

APIC ノースバウンド API にサブミットされた XML ファイルを構成する方法については、『Cisco APIC Layer 4 to Layer 7 Device Package Development Guide』を参照してください。

『Cisco APIC Management Information Model Reference』で定義されている次の Python API は、ノースバウンド API を使用した REST POST コールのサブミットに使用できます。

- `vns:LDevVip` : デバイス クラスタをアップロードします
- `vns:CDev` : デバイスをアップロードします

- vns:Lif : 論理インターフェイスを作成します
- vns:AbsGraph : グラフを作成します
- vz:BrCP : コントラクトにグラフをアタッチします

ノースバウンド REST API を使用して、POST を構成する方法 (Python での実行方法も含まれます) についての詳細は、『Cisco APIC REST API User Guide』を参照してください。

REST API の使用例

次の REST 要求は、ブロードキャストドメインを持つテナント、L3 ネットワーク、アプリケーション EPG、およびアプリケーションプロファイルを作成します。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">

    <!--L3 Network-->
    <fvCtx name="MyNetwork"/>

    <!-- Bridge Domain for MySrvr EPG -->
    <fvBD name="MySrvrBD">
      <fvRsCtx tnFvCtxName="MyNetwork" />
      <fvSubnet ip="10.10.10.10/24">
        </fvSubnet>
      </fvBD>

    <!-- Bridge Domain for MyClnt EPG -->
    <fvBD name="MyClntBD">
      <fvRsCtx tnFvCtxName="MyNetwork" />
      <fvSubnet ip="20.20.20.20/24">
        </fvSubnet>
      </fvBD>

    <fvAp dn="uni/tn-acme/ap-MyAP" name="MyAP">

      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MyClnt" name="MyClnt">
        <fvRsBd tnFvBDName="MySrvrBD" />
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
        <fvRsProv tnVzBrCPName="webCtrct"> </fvRsProv>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]" encap="vlan-202"/>

        <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]" encap="vlan-202"/>
      </fvAEPg>

      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MySRVR" name="MySRVR">
        <fvRsBd tnFvBDName="MyClntBD" />
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
        <fvRsCons tnVzBrCPName="webCtrct"> </fvRsCons>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]" encap="vlan-203"/>

        <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]" encap="vlan-203"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```

次の REST 要求は VLAN ネームスペースを作成します。

```
<polUni>
  <infraInfra>
    <fvnsVlanInstP name="MyNS" allocMode="dynamic">
      <fvnsEncapBlk name="encap" from="vlan-201" to="vlan-300"/>
    </fvnsVlanInstP>
  </infraInfra>
</polUni>
```

```

    </fvnsVlanInstP>
  </infraInfra>
</polUni>

```

次の REST 要求は VMM ドメインを作成します。

```

<polUni>
  <vmmProvP vendor="Vendor1">
    <vmmDomP name="MyVMs">
      <infraRsVlanNs tDn="uni/infra/vlanns-MyNS-dynamic" />
      <vmmUsrAccP name="admin" usr="administrator" pwd="in$leme" />
      <vmmCtrlrP name="vcenter1" hostOrIp="192.168.64.186">
        <vmmRsAcc tDn="uni/vmmp-Vendor1/dom-MyVMs/usracc-admin" />
      </vmmCtrlrP>
    </vmmDomP>
  </vmmProvP>
</polUni>

```

次の REST 要求は物理ドメインを作成します。

```

<polUni>
  <physDomP name="phys">
    <infraRsVlanNs tDn="uni/infra/vlanns-MyNS-dynamic"/>
  </physDomP>
</polUni>

```

次の REST 要求はデバイス クラスタを作成します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1" contextAware=1>
      <vnsRsMDevAtt tDn="uni/infra/mDev-Acme-ADC-1.0"/>
      <vnsRsDevEpg tDn="uni/tn-acme/ap-services/epg-ifc"/>
      <vnsRsALDevToPhysDomP tDn="uni/phys-phys"/>

      <vnsCMgmt name="devMgmt" host="42.42.42.100" port="80" />

      <vnsCCred name="username" value="admin"/>

      <vnsCCredSecret name="password" value="admin" />
    </vnsLDevVip>
  </fvTenant>
</polUni>

```

次の REST 要求はデバイス クラスタ コンテキストを作成します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevCtx ctrctNameOrLbl="webCtrct" graphNameOrLbl="G1" nodeNameOrLbl="Node1">
      <vnsRsLDevCtxToLDev tDn="uni/tn-acme/lDevVip-ADCCluster1"/>
      <vnsLIfCtx connNameOrLbl="ssl-inside">
        <vnsRsLIfCtxToLIf tDn="uni/tn-acme/lDevVip-ADCCluster1/lIf-int"/>
      </vnsLIfCtx>
      <vnsLIfCtx connNameOrLbl="any">
        <vnsRsLIfCtxToLIf tDn="uni/tn-acme/lDevVip-ADCCluster1/lIf-ext"/>
      </vnsLIfCtx>
    </vnsLDevCtx>
  </fvTenant>
</polUni>

```

次の REST 要求はデバイス クラスタの論理インターフェイスを追加します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1">
      <vnsLIf name="C5">
        <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-outside"/>
        <vnsRsCIfAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-int"/>
      </vnsLIf>
      <vnsLIf name="C4">
        <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-inside"/>
        <vnsRsCIfAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-ext"/>
      </vnsLIf>
    </vnsLDevVip>
  </fvTenant>
</polUni>

```

```

    </vnsLIf>
  </vnsLDevVip>
</fvTenant>
</polUni>

```

次の REST 要求はデバイス クラスタの具象デバイスを追加します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1">
      <vnsCDev name="ADC1" devCtxLbl="C1">
        <vnsCIf name="int">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/22]"/>
        </vnsCIf>
        <vnsCIf name="ext">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]"/>
        </vnsCIf>
        <vnsCIf name="mgmt">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/20]"/>
        </vnsCIf>
        <vnsCMgmt name="devMgmt" host="172.30.30.100" port="80"/>
        <vnsCCred name="username" value="admin" />
        <vnsCCred name="password" value="admin" />
      </vnsCDev>
      <vnsCDev name="ADC2" devCtxLbl="C2">
        <vnsCIf name="int">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/23]"/>
        </vnsCIf>
        <vnsCIf name="ext">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/24]"/>
        </vnsCIf>
        <vnsCIf name="mgmt">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/30]"/>
        </vnsCIf>
        <vnsCMgmt name="devMgmt" host="172.30.30.200" port="80" />
        <vnsCCred name="username" value="admin" />
        <vnsCCred name="password" value="admin" />
      </vnsCDev>
    </vnsLDevVip>
  </fvTenant>
</polUni>

```

次の REST 要求はサービス グラフを作成します。

```

<polUni>
  <fvTenant name="acme">
    <vnsAbsGraph name = "G1">
      <vnsAbsTermNode name = "Input1">
        <vnsAbsTermConn name = "C1" direction = "output">
          </vnsAbsTermConn>
        </vnsAbsTermNode>
      <!-- Node1 Provides SLB functionality -->
      <vnsAbsNode name = "Node1" funcType="GoTo" >
        <vnsRsDefaultScopeToTerm
          tDn="uni/tn-acme/AbsGraph-G1/AbsTermNode-Output1/outtmn1"/>
        <vnsAbsFuncConn name = "C4" direction = "input">
          <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external"
        />
          <vnsRsConnToLIf tDn="uni/tn-acme/lDevVip-ADCCluster1/lIf-C4" />
        </vnsAbsFuncConn>
        <vnsAbsFuncConn name = "C5" direction = "output">
          <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-internal"
        />
          <vnsRsConnToLIf tDn="uni/tn-acme/lDevVip-ADCCluster1/lIf-C5" />
        </vnsAbsFuncConn>
        <vnsRsNodeToMFunc tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB"/>
      </vnsAbsNode>
    </vnsAbsGraph>
  </fvTenant>
</polUni>

```

```

<vnsAbsTermNode name = "Output1">
  <vnsAbsTermConn name = "C6" direction = "input">
  </vnsAbsTermConn>
</vnsAbsTermNode>

<vnsAbsConnection name = "CON1">
  <vnsRsAbsConnectionConns
    tDn="uni/tn-acme/AbsGraph-G1/AbsTermNode-Input1/AbsTConn" />
  <vnsRsAbsConnectionConns
    tDn="uni/tn-acme/AbsGraph-G1/AbsNode-Node1/AbsFConn-C4" />
</vnsAbsConnection>

<vnsAbsConnection name = "CON3">
  <vnsRsAbsConnectionConns
    tDn="uni/tn-acme/AbsGraph-G1/AbsNode-Node1/AbsFConn-C5" />
  <vnsRsAbsConnectionConns
    tDn="uni/tn-acme/AbsGraph-G1/AbsTermNode-Output1/AbsTConn" />
</vnsAbsConnection>
</vnsAbsGraph>
</fvTenant>
</polUni>

```

次の REST 要求はセキュリティ ポリシー (コントラクト) を作成します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vzFilter name="HttpIn">
      <vzEntry name="e1" prot="6" dToPort="80"/>
    </vzFilter>

    <vzBrCP name="webCtrct">
      <vzSubj name="http">
        <vzRsSubjFiltAtt tnVzFilterName="HttpIn"/>
      </vzSubj>
    </vzBrCP>
  </fvTenant>
</polUni>

```

次の REST 要求はアプリケーション EPG からのグラフ コンフィギュレーション パラメータを提供します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">

    <!-- Application Profile -->
    <fvAp dn="uni/tn-acme/ap-MyAP" name="MyAP">

      <!-- EPG 1 -->
      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MyClnt" name="MyClnt">
        <fvRsBd tnFvBDName="MyClntBD" />
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
        <fvRsProv tnVzBrCPName="webCtrct">
        </fvRsProv>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/20]" encap="vlan-201"/>

        <fvSubnet name="SrcSubnet" ip="192.168.10.1/24" />
      </fvAEPg>

      <!-- EPG 2 -->
      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MySRVR" name="MySRVR">
        <fvRsBd tnFvBDName="MyClntBD" />
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
        <fvRsCons tnVzBrCPName="webCtrct">
        </fvRsCons>

        <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
          key="Monitor" name="monitor1">
          <vnsParamInst name="weight" key="weight" value="10"/>
        </vnsFolderInst>

        <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"

```

```

key="Service" name="Service1">
<vnsParamInst name="servicename" key="servicename"
  value="crpvgrtst02-8010"/>
<vnsParamInst name="servicetype" key="servicetype" value="TCP"/>
<vnsParamInst name="servername" key="servername"
  value="s192.168.100.100"/>
<vnsParamInst name="serveripaddress" key="serveripaddress"
  value="192.168.100.100"/>
<vnsParamInst name="serviceport" key="serviceport" value="8080"/>
<vnsParamInst name="svrtimeout" key="svrtimeout" value="9000" />
<vnsParamInst name="clttimeout" key="clttimeout" value="9000" />
<vnsParamInst name="usip" key="usip" value="NO" />
<vnsParamInst name="useproxyport" key="useproxyport" value="" />
<vnsParamInst name="cip" key="cip" value="ENABLED" />
<vnsParamInst name="cka" key="cka" value="NO" />
<vnsParamInst name="sp" key="sp" value="OFF" />
<vnsParamInst name="cmp" key="cmp" value="NO" />
<vnsParamInst name="maxclient" key="maxclient" value="0" />
<vnsParamInst name="maxreq" key="maxreq" value="0" />
<vnsParamInst name="tcpb" key="tcpb" value="NO" />
<vnsCfgRelInst name="MonitorConfig" key="MonitorConfig"
  targetName="monitor1"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
  key="Network" name="Network">
<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
  key="vip" name="vip">
  <vnsParamInst name="vipaddress1" key="vipaddress" value="10.10.10.100"/>
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
  devCtxLbl="C1" key="snip" name="snip1">
  <vnsParamInst name="snipaddress" key="snipaddress"
    value="192.168.1.100"/>
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
  devCtxLbl="C2" key="snip" name="snip2">
  <vnsParamInst name="snipaddress" key="snipaddress"
    value="192.168.1.101"/>
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
  devCtxLbl="C3" key="snip" name="snip3">
  <vnsParamInst name="snipaddress" key="snipaddress"
    value="192.168.1.102"/>
</vnsFolderInst>
</vnsFolderInst>

<!-- SLB Configuration -->
<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
  key="VServer" name="VServer">
  <!-- Virtual Server Configuration -->
  <vnsParamInst name="port" key="port" value="8010"/>
  <vnsParamInst name="vip" key="vip" value="10.10.10.100"/>
  <vnsParamInst name="vservername" key="vservername"
    value="crpvgrtst02-vip-8010"/>
  <vnsParamInst name="servicename" key="servicename"
    value="crpvgrtst02-8010"/>
  <vnsParamInst name="servicetype" key="servicetype" value="TCP"/>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
    key="VServerGlobalConfig" name="VServerGlobalConfig">
    <vnsCfgRelInst name="ServiceConfig" key="ServiceConfig"
      targetName="Service1"/>
    <vnsCfgRelInst name="VipConfig" key="VipConfig"
      targetName="Network/vip"/>
  </vnsFolderInst>
</vnsFolderInst>
</fvAEPg>

```

```
    </fvAp>  
  </fvTenant>  
</polUni>
```

次の REST 要求はコントラクトにサービス グラフをアタッチします。

```
<polUni>  
  <fvTenant name="acme">  
    <vzBrCP name="webCtrct">  
      <vzSubj name="http">  
        <vzRsSubjGraphAtt graphName="G1" termNodeName="Input1"/>  
      </vzSubj>  
    </vzBrCP>  
  </fvTenant>  
</polUni>
```




索引

V

- vCenter/vShield コントローラ プロファイル **10**
 - 設定 **10**
- VLAN プール **7**
 - 設定 **7**
- VMM クレデンシヤル **10**
 - 設定 **10**
- VMM ドメイン **9**
 - 設定 **9**

あ

- アプリケーション プロファイル **14**
 - 設定 **14**

か

- 仮想デバイス **87**
 - モニタリング **87**
- カプセル化ブロック範囲 **7**
 - 設定 **7**
- 管理エンドポイントグループ **19**
 - 設定 **19**

く

- 具象デバイス **30**
 - 設定 **30**

こ

- コントラクト **18**
 - 設定 **18**

さ

- サービス グラフ **51, 75**
 - コントラクトとの関連付け **75**
 - 設定 **51**
- サービス グラフ インスタンス **79**
 - モニタリング **79**
- サービス グラフ エラー **81**
 - モニタリング **81**

せ

- 設定 **12**
 - ブリッジ ドメイン **12**

て

- テナント **11**
 - 設定 **11**
- デバイス クラスター **28, 32, 45**
 - インポート **45**
 - 仮想サービス デバイスの設定 **32**
 - 設定 **28**
- デバイス パッケージ **25**
 - インポート **25**

ふ

- 物理ドメイン **8**
 - 設定 **8**
- ブリッジ ドメイン **12**
 - 設定 **12**

れ

レイヤ3 ネットワーク [13](#)
設定 [13](#)

ろ

論理デバイス コンテキスト [42](#)
設定 [42](#)