レイヤ 4 ～ レイヤ 7 のサービスの挿入

この章の内容は、次のとおりです。

レイヤ 4 ～ レイヤ 7 のサービスの挿入

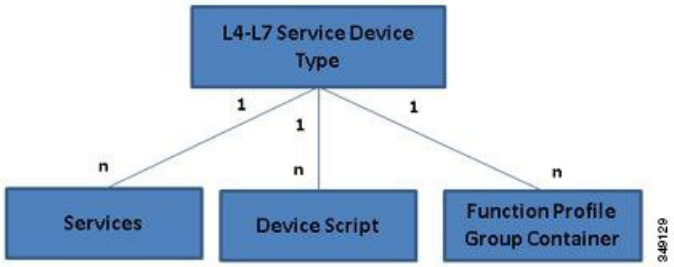
Cisco Application Policy Infrastructure Controller（APIC）は、ネットワーク サービスを管理します。ポリシーは、サービスを挿入するために使用されます。 APIC サービス統合により、ライフサイクルの自動化フレームワークが提供され、サービスがオンラインまたはオフラインになった場合にシステムが動的に応答できます。 ファブリック全体で使用可能な共有サービスは、ファブリックの管理者によって管理されます。 単一のテナント向けのサービスは、テナントの管理者によって管理されます。

APICは、ポリシー制御の中心点として機能すると同時に、自動サービス挿入を提供します。 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って流れるように、ネットワークを自動的に設定できます。 また、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できます。 このアプローチにより、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すべてのトラフィック誘導技術の管理に伴う課題を排除できます。

レイヤ 4 ～ レイヤ 7 のポリシー モデル

レイヤ 4 ～ レイヤ 7 のサービス デバイス タイプ ポリシーには、パッケージおよびデバイス スクリプトでサポートされるサービスなどの主要な管理対象オブジェクトが含まれます。 次の図は、レイヤ 4 ～ レイヤ 7 のサービス デバイス タイプ ポリシー モデルのオブジェクトを示します。

図 *1*：レイヤ *4*～ レイヤ *7*のポリシー モデル



レイヤ 4 ～ レイヤ 7 のサービス ポリシーには次のものが含まれます。

- **サービス**：SSLオフロードやロードバランシングなどのデバイスによって提供されるすべての機能のメタデータが含まれます。 このMOには、コネクタの名前、VLANやVXLANなどのカプセル化のタイプ、およびインターフェイス ラベルが含まれます。
- **デバイス スクリプト**：名前、パッケージ名、バージョンなどのスクリプト ハンドラの関連属性に関するメタ情報を含むデバイス スクリプト ハンドラを表します。
- **機能プロファイル グループ コンテナ**：サービス デバイス タイプで使用可能な機能を含むオブジェクト。 機能プロファイルには、フォルダに編成されたデバイスでサポートされる設定可能なすべてのパラメータが含まれます。

サービス グラフ

Cisco Application Centric Infrastructure（ACI）は、アプリケーションの欠くことのできない一部としてサービスを扱います。 必要とされるすべてのサービスが、Cisco Application Policy Infrastructure Controller（APIC）から ACI ファブリックでインスタンス化されるサービス グラフとして扱われます。 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。 各機能はノードとして表されます。

グラフが APIC に設定されると、APIC はサービス グラフに明記されたサービス機能の要件に従って、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。

サービス グラフは、アプリケーションの複数の階層として表され、適切なサービス機能が間に挿入されます。

サービス アプライアンス（デバイス）は、グラフ内でサービス機能を実行します。 1 つ以上のサービス アプライアンスが、グラフに必要なサービスをレンダリングするために必要になることがあります。 1 つ以上のサービス機能が単一のサービス デバイスで実行できます。

サービス グラフおよびサービス機能には、次の特性があります。

- エンドポイント グループ（EPG）で送信または受信されたトラフィックはポリシーに基づいてフィルタリングでき、トラフィックのサブセットはグラフ内の異なるエッジにリダイレクトできます。
- サービス グラフのエッジには方向性があります。
- タップ（ハードウェア ベースのパケット コピー サービス）は、サービス グラフの異なるポイントに接続できます。
- 論理機能は、ポリシーに基づいて適切な（物理または仮想）デバイスでレンダリングできます。
- サービス グラフでは、エッジの分割と結合がサポートされ、管理者は線形サービス チェーンに制限されません。
- トラフィックは、サービス アプライアンスが発信した後にネットワーク内で再度分類できます。
- 論理サービス機能は、要件に応じて、拡張や縮小が可能で、クラスタ モードまたは 1:1 アクティブ/スタンバイ ハイアベイラビリティ モードで展開できます。

サービス グラフ コンフィギュレーション パラメータ

サービス グラフには、デバイス パッケージで指定されたコンフィギュレーション パラメータを割り当てることができます。 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、APIC はそのパラメータをデバイス パッケージ内にあるデバイス スクリプトに渡します。 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 グラフ接続

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。

自動サービス挿入

VLAN および仮想ルーティングおよび転送（VRF）スイッチングは、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、サービス挿入とセキュア ソケット レイヤ（SSL）オフロード、サーバ ロード バランシング（SLB）、Web アプリケーション ファイアウォール（WAF）およびファイアウォールなどのネットワーク サービスのプロビジョニングを自動化できます。 ネットワーク サービスは通常、Application Delivery Controller（ADC）やファイアウォールなどのサービス アプライアンスによってレンダリングされます。 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って流れるように、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。

デバイス パッケージ

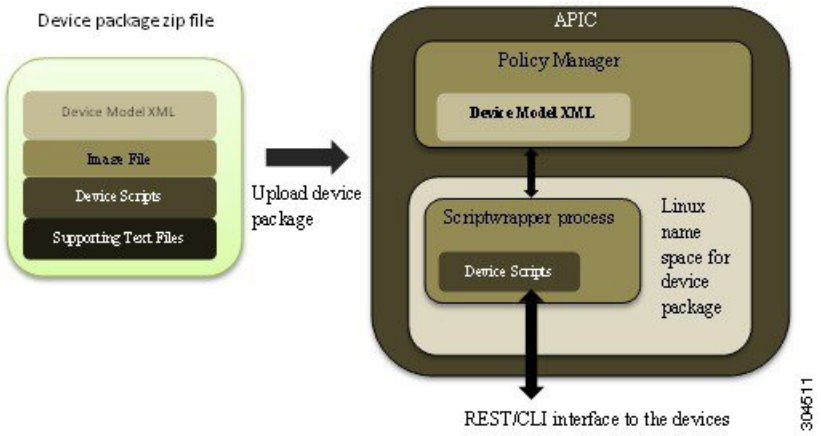
Application Policy Infrastructure Controller（APIC）は、サービス デバイスの設定およびモニタリングにデバイス パッケージを必要とします。 デバイス パッケージは、サービス デバイスのクラスを管理して、デバイスが何であるか、およびデバイスで実行できることを APIC が認識できるように APIC にデバイスの情報を提供します。 デバイス パッケージにより、管理者は APIC 上でネットワーク サービスを中断なく追加、変更、または削除することができます。 APIC への新しいデバイス タイプの追加は、デバイス パッケージをアップロードすることで実行できます。

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： • デバイス プロパティ： ◦ [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。

次の図に、デバイス パッケージによる APIC サービスの自動化と挿入アーキテクチャを示します。

図 *2*：デバイス パッケージ アーキテクチャ



デバイス パッケージは、デバイス ベンダーが提供するか、またはシスコが作成できます。

デバイス パッケージにより、管理者は次のサービスの管理を自動化することができます。

- デバイスの接続と切断
- エンドポイントの接続と切断
- サービス グラフのレンダリング
- ヘルス モニタリング
- アラーム、通知、ロギング
- カウンタ

デバイス パッケージが GUI またはノースバウンド APIC インターフェイス経由でアップロードされると、APIC はそれぞれ一意なデバイス パッケージのネームスペースを作成します。 デバイス パッケージの内容は、解凍されネーム スペースにコピーされます。 デバイス パッケージのネーム スペース用に作成されるファイル構造は次のとおりです。

```
root@apic1:/# ls
bin  dbin  dev  etc  fwk  install images lib  lib64  logs  pipe  sbin  tmp  usr util

root@apic1:/install# ls
DeviceScript.py DeviceSpecification.xml feature common images  lib  util.py
```

デバイス パッケージの内容は `install` ディレクトリにコピーされます。

APIC がデバイス モデルを解析します。 XML ファイルで定義される管理対象オブジェクトは、ポリシー マネージャによって維持される APIC の管理対象オブジェクト ツリーに追加されます。

デバイス パッケージで定義される Python スクリプトは、ネームスペースのスクリプト ラッパー プロセス内で開始されます。 ファイル システムへのアクセスは制限されます。 Python スクリプトは、`/tmp` に一時ファイルを作成でき、デバイス パッケージの一部としてバンドルされたテキスト ファイルにアクセスできます。 ただし、Python スクリプトではファイル内に永続データを作成または保存しないでください。

デバイス スクリプトは、ACI ロギング フレームワークを通してデバッグ ログを生成できます。 ログは、`logs` ディレクトリ下の `debug.log` という循環型ファイルに書き込まれます。

各デバイス パッケージのバージョンは自身のネームスペースで動作するため、デバイス パッケージの複数のバージョンが APIC 上に共存できます。 管理者は、一連のデバイスを管理するための特定のバージョンを選択できます。

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く具象インターフェイス上でプログラミングされます。

機能ノード

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。

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、管理者がサービス グラフを定義した時点で割り当てることもできます。 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

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。

端末ノード

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。管理者は、コントラクトに端末ノードを接続することにより、2 つ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対しサービス 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。

権限について

管理者は、（APIC）で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。 管理者は、管理者のロールに次の権限を付与できます。

特権	説明
nw-svc-connectivity	• 管理 EPG の作成 • 他のオブジェクトに管理接続を作成
nw-svc-policy	• 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
nw-svc-device	• 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の管理者だけがデバイス パッケージを APIC にアップロードできます。

サービスの自動化と構成管理

Cisco APIC は、サービス デバイスの構成管理と自動化のポイントとして任意に動作でき、ネットワーク自動化とのサービス デバイスの調整を行うことができます。 Cisco APIC は、さまざまなイベントで Python スクリプトを使用してサービス デバイスと連動し、デバイス固有の Python スクリプト機能を呼び出します。

デバイス スクリプトとサービス デバイスでサポートされる機能を定義するデバイスの仕様は、デバイス パッケージとしてまとめられ、Cisco APIC にインストールされます。 デバイス スクリプト ハンドラは、デバイス設定モデルに基づいてその REST インターフェイス（推奨）または CLI を使用してデバイスとやりとりします。

サービス リソースのプーリング

Cisco ACI ファブリックは、多数の宛先間で非ステートフル負荷分散を実行できます。 この機能により、組織は物理および仮想サービス デバイスをサービス リソース プールにグループ化でき、機能や場所によってさらにグループ化できます。 これらのプールは、標準のハイアベイラビリティ メカニズムを使用することでハイアベイラビリティを提供するか、または障害が発生した場合に、他のメンバーに負荷が再分散された状態で簡易なステートフル サービス エンジンとして使用できます。 どちらのオプションでも、等コスト マルチパス（ECMP）、ポート チャネル機能および共有状態を必要とするサービス アプライアンスのクラスタリングの現在の制限をはるかに超える横方向のスケールアウトが提供されます。

サービス デバイスがファブリックとやりとりする必要がない場合、Cisco ACI はサービス デバイスを使用して簡易バージョンのリソース プーリングを実行できます。また、ファブリックとサービス デバイス間の調整を伴うより高度なプーリングも実行できます。