



Cisco Evolved Programmable Network Manager RESTful API

- [Cisco EPN ManagerSDK \(1 ページ\)](#)
- [Cisco EPN Manager API \(2 ページ\)](#)
- [Cisco EPN Manager RESTful API の用途 \(3 ページ\)](#)
- [Cisco EPN Manager RESTful API の使用方法 \(3 ページ\)](#)
- [RESTConf API : 概要 \(5 ページ\)](#)
- [RESTConf API 機能エリア \(7 ページ\)](#)
- [認証および承認 \(8 ページ\)](#)
- [Cisco EPN Manager REST API スタートアップ ガイド \(8 ページ\)](#)
- [統計情報 \(12 ページ\)](#)

Cisco EPN ManagerSDK

Cisco EPN Manager SDK は、Cisco EPN Manager の機能を拡張したり、データにアクセスしたり、アプリケーションから自動化操作を呼び出したりできるテクノロジーの集合です。Cisco EPN Manager SDK には、RESTful API と Open Automation が含まれています。RESTful API で「wgetおよびcURLユーティリティを使用したbash」、「Python」、「Ruby」、Javaなどのスクリプト言語を使用できます。

Cisco EPN Manager SDK テクノロジーを使用して、次のことが可能です。

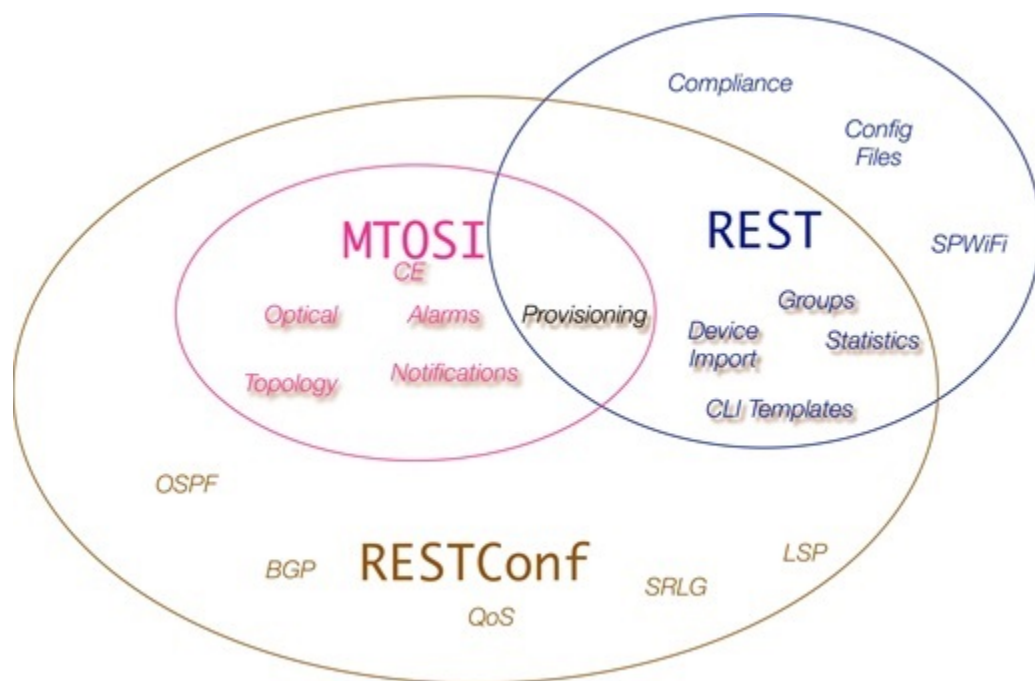
- プログラムによる Cisco EPN Manager へのアクセス : Cisco EPN Manager RESTful API を使用してワークフローを呼び出し、レポートを取得できます。
- Cisco EPN Manager のカスタマイズ : カスタムワークフロータスクを作成し、スクリプトモジュールに独自の jar ファイルおよびスクリプトライブラリを展開して、Cisco EPN Manager をカスタマイズできます。スクリプトバンドルからカスタムタスクを使用します。

Cisco EPN Manager API

Cisco EPN Managerは、あらゆる標準規格のOSSシステムノースバウンドとの統合を可能にする、使いやすく包括的なAPIを提供します。これらのAPIでは、最も一般的に使用される3つの標準規格であるMTOSI、RESTConf、およびRESTfulAPIを使用してコア機能を拡張します。

自動化向けの迅速な開発設計を実現するためには、開発者が一貫性を持ってAPIを開発し、わかりやすく使いやすい設計ガイドラインを示せるようにすることが重要です。一貫性を保つことで、チームは一般的なコード、パターン、ドキュメント、および設計上の決定事項を活用できます。Cisco EPN Managerでは、シンプルで広範な例を使用してアプリケーション内でRESTful APIを簡単に利用可能にできます。次の図は、Cisco EPN Managerにおいて*すべて*のAPI全体で確認される一般的な動作を示した例です。

図 1: EPNM RESTful APIの一般的な動作



この章は、Cisco EPN Manager 開発キット (SDK) および関連技術の使用に関心がある、以下の技術担当者を対象としています。次のようなユーザが対象です。

- Cisco EPN Manager を使用してリソースを自動化する機能を拡張したいと考えているシステム管理者および REST API 開発者。
- Cisco EPN Manager SDK または API テクノロジーを比較し、どちらがアプリケーションに最適であるかを判断したいその他のユーザ。

Cisco EPN Managerは、MTOSI、REST、および RESTConf の3つのノースバウンドAPIインターフェイスタイプを提供します。この3つのAPIによって提供される機能はかなり類似していますが、すべてのAPIが同じ機能を提供するわけではありません。

MTOSI API は、動作の点で最も異なる API です。光イーサネットおよびキャリアイーサネット機能の基本的なプロビジョニングが可能な SOAP over HTTPS インターフェイスを提供します。DWDM や OTN などの光サービスをプロビジョニングする必要がある場合に、RESTConf API を使用できないときは、MTOSI インターフェイスを使用します。

REST API は、コンフィギュレーションファイル、グループ管理、およびデバイスインポート機能を処理するその他の機能に加えて、システム情報、ほとんどの統計情報、およびアシュアランス情報を提供します。

RESTConf API は、上記の REST API と同様に RESTful インターフェイスに準拠しています。コアルーティングとスイッチングだけでなく、キャリアイーサネット、L2/L3 VPN、回線エミュレーション、OTN および DWDM テクノロジーに関するすべてのプロビジョニングを提供します。

詳細については、cisco.com で入手できる個々の [Cisco EPNM 統合ガイド](#) を参照してください。

Cisco EPN Manager RESTful API の用途

Cisco EPN Manager RESTful API は、HTTP 要求を作成可能なプログラムまたはスクリプトで使用できる、言語に依存しないインターフェイスです。別のプログラムまたはプロセスから Cisco EPN Manager 上の操作を呼び出す場合は、REST API を使用してください。

アプリケーションは、RESTful API を使用して以下を実行できます。

- Cisco EPN Manager ドメイン内の物理デバイスと仮想デバイス、ネットワーク、アプリケーション、グループとユーザ、ポリシー、およびその他のモニタ対象エンティティに関する Cisco EPN Manager レポートを取得する。
- Cisco EPN Manager 独自の追加操作の呼び出し。

Cisco EPN Manager RESTful API の使用方法

RESTful API クライアントは標準の HTTP 要求および応答を使用して Cisco EPN Manager とやり取りするため、RESTful API 応答はあらゆる Web ブラウザと互換性があります。多くのプログラミング言語には、HTTP 要求の作成と送信、および HTTP 応答の処理に専用のライブラリがあります。

REST API コール的大部分は、要求または応答内のデータをそれぞれ送信および返信します。これらのデータペイロードは、RESTful API コールに応じて 2 つの方法のいずれかでフォーマット化されます。JavaScript Object Notation (JSON) ペイロードを使用する RESTful API コールもあれば、XML ペイロードを使用するものもあります。通常の場合、複雑と見なされるアプリケーションには両方を使用する必要があります。

JSON ベースの RESTful API コールは、JSON ペイロードを使用した単純な HTTP 要求および応答です。JSON は、判読可能なデータ交換のために設計された軽量テキストベースのオープンスタンダードです。JSON は、単純なデータ構造と連想配列を表します。アプリケーションは、

特殊な RESTful API ライブラリを使用することなく JSON ベースの API を直接呼び出し、アプリケーションに固有の方法を使用して JSON データを解析します。

Cisco EPN Manager API へのすべての要求でユーザ認証が必要です。認証の詳細については、「[Authentication and Authorization](#)」を参照してください。

RESTful API の認証は、Cisco EPN Manager の登録済みユーザのみが API 要求を作成できることを要求することによって実行されます。Cisco EPN Manager の場合、API へのアクセスは、NBI 読み取り、NBI 書き込み、NBI クレデンシャル（廃止）などの 3 つのユーザグループによって制御されます。これらの各グループは、異なる API セットへのアクセスを制御します。必要に応じて、複数のグループにユーザを割り当てることができます。API リソースのドキュメントページで、アクセスに必要なユーザグループを確認できます。Cisco EPN Manager で作成および登録したユーザには、一意の RESTful API アクセス キーが割り当てられます。

RESTConf トポロジリンク取得リソースを入手するには、RESTful トポロジリンクリソースを使用してこれらの値を取得し、必要な関連付けを実行します。

Cisco EPN Manager RESTful API のドキュメントにアクセスするには、次の操作を行います。

- Cisco EPN Manager を起動して右上隅にある  をクリックすると、ウィンドウ設定メニューが開きます。

Logged In As root

- Log out
- Change Password
- Set Current Page As Home
- My Preferences
- Support Cases

Virtual Domain:ROOT-DOMAIN

Help

- Getting Started
- Online Help
- API Help
- Supported Devices
- MSE Installation Guide
- Documentation Home Page

Feedback

- I wish this page would...

About Cisco EPN Manager

- **[ヘルプ (Help)] > [APIヘルプ (API Help)] > [REST API]** を選択し、システム要件、開発環境のセットアップ方法、ライブラリのインストール方法、RESTful API の使用方法、

すべての Cisco EPN Manager REST API 関数のリスト、REST API リソース、さまざまな使用例、クエリなどを確認します。

RESTConf API : 概要

Cisco EPN Manager での実装は、RESTConf/Yang 仕様情報モデルおよび運用 API プロトコルに準拠しています。必要に応じて情報モデルおよび運用 API は標準的な方法で拡張され、RESTConf/Yang インターフェイスに対するシスコのベンダー拡張をサポートします。これらの拡張は、情報モデル拡張の一連の xml および yang スキーマ定義として追加されます。

RESTCONF : 構造化データ (XML または JSON) および YANG を使用して REST ライクな API を提供します。これによりさまざまなネットワークデバイスにプログラムを使用してアクセスできます。RESTCONF API は HTTPs メソッドを使用します。

YANG : モデル構成および操作機能に使用されるデータモデリング言語。YANG は、NETCONF および RESTCONF API によって実行できる関数の有効範囲と種類を決定します。

RESTConf API の基本的な構造を次に示します。

- **HTTP ヘッダー** : HTTP ヘッダーは、HTTP 要求内で送信または要求されるコンテンツの説明に使用されます。HTTP ヘッダーには以下が含まれます。
 - **Content-Type** : サーバ側で、着信要求にエンティティがアタッチされている場合があります。タイプを特定するために、サーバは HTTP 要求ヘッダーの Content-Type を使用します。
 - **Accept** : 同様に、クライアント側で必要な表現のタイプを特定するために、HTTP ヘッダーの ACCEPT が使用されます。通常、要求に Accept ヘッダーが存在しない場合、サーバは事前設定されたデフォルトの表現タイプを送信できます。
- **HTTP メソッド** : API の呼び出しには次のメソッドが使用されます。
 - **Get** : GET メソッドは、リソースのデータとメタデータを取得するためにクライアントによって送信されます。
 - **Post** : 新しいエンティティの作成に使用されますが、エンティティの更新にも使用できます。POST メソッドは、データリソースを作成したり、操作リソースを呼び出したりするためにクライアントによって送信されます。
 - **Put** : PUT 要求はべき等です。PUT メソッドは、ターゲットデータリソースを作成または置換するためにクライアントによって送信されます。
 - **Delete** : リソースの削除を要求します。DELETE メソッドは、ターゲットリソースを削除するために使用されます。
- **メッセージ** : RESTCONF プロトコルは HTTP メッセージを使用します。1 つの HTTP メッセージは、1 つのプロトコルメソッドに対応します。
- **メディアタイプ** : XML と JSON。

• クエリ パラメータ :

- Content
- Depth
- Fields
- Filter
- Insert
- Point
- Start-time
- Stop-time
- With-defaults

標準的な使用方法 : Get メソッド

「GET All RESTConf API コール」は、既存のエンドポイント スキーマ セクションを返します。

GET /restconf/data/ietf-yang-library:modules-state

すべての RESTConf API エンドポイントを取得します。

スキーマ URL は、多くの列挙変数とモジュールの構造コンポーネントに関する詳細を提供します。

スキーマ取得の例 :

GET /restconf/schema/v1/cisco-resource-optical

モジュールの YANG スキーマをプレーン テキストとして返します。

多くの場合、RESTConf API コールでは複数の URL パラメータを使用できます。これらは、すべてのコールで実装されるわけではありません。

- .maxCount
- .startIndex
- .depth

```
GET /restconf/data/v1/module:resource
The general format for a RESTConf GET call.
```

表 1:クエリパラメータ

クエリー パラメータ	説明
.depth (整数)	取得する詳細レベルの数。
.maxCount (整数)	取得される行数の制限。
.startIndex (整数)	取得する最初の行。

```
GET /restconf/data/v1/module:resource HTTP/1.1
```

RESTful API ユーティリティ

- 完全に識別可能な名前：このインターフェイスのインベントリ オブジェクトには、FDN（完全識別名）を表す属性があります。これらの属性はオブジェクトの識別子として使用されたり、オブジェクトへの参照が必要な場合にクエリパラメータや返されるデータ内のオブジェクト参照として使用されたりします。この FDN は、「!」で区切った `<type>=<value>` ペアのシーケンスを含むタイプ/値ペアのセットで構成されるフォーマット済みの文字列です。
 - `<type>` は、データ モデルで定義された、階層内のインベントリ オブジェクトを表す一定値です（例：MD、ND、EQ、PTP、FTP、CTP、TL、VC、CFS など）。
 - `<value>` は、インベントリ オブジェクトの属性/値ペアを表す任意のテキストまたは「;」で区切られた `<attrName>=<attrValue>` ペアのシーケンスで、タイプで表されるオブジェクトのローカル範囲内の一意の値を構成します。

詳細については、『[Cisco EPNM RESTConf ガイド](#)』を参照してください。

RESTConf API 機能エリア

- アラーム：アラームモジュールは、アラームとイベントを取得して確認するメカニズムを提供します。推奨されるアラーム管理方法は、通知 API を介してリッスンすることです。サービスは異なるタイプまたは重要度のイベントを収集し、この API 経由で処理できます。
- パフォーマンステスト：サービスパフォーマンステストは、サービスプロビジョニングの一環としても、スタンドアロンでも実行できます。Restconf NBI は、スタンドアロンのパフォーマンステストの実行をサポートしています。
- Quality of Service (QoS)：QoS は、ネットワークトラフィックの差別化サービスを配信できるようにする一連の機能です。Cisco EPN Manager を使用して、キャリアイーサネットインターフェイスで QoS を設定できます。
- OAM：EPNM RESTConf API の OAM テストは、2つの一般的なカテゴリに分類されます（サービス OAM 設定とネットワークリソース OAM 設定）。Y.1731、Y.1564、および BERT テストでは、`service-oam-config` エンドポイントが使用されます。OTDR の場合は、`network-resource-oam-config` エンドポイントが使用されます。EPNM RESTConf OAM コールは、テストを開始するための POST コマンドを提供します。この要求によって、テスト ID とテスト要求 ID が生成されます。通常は、このテスト ID を後続のコールで使用して結果を取得します。ほとんどのテストタイプには個々の URL があります。
- サービスプロファイル：サービスプロファイルには、各回線/VC タイプのプロビジョニングに使用できる事前定義済みのプロビジョニング要求（注文データ）が含まれています。NBI プロビジョニング要求では、サービスプロファイルリファレンスを使用して、プロビジョニングで使用するプロビジョニング要求データを取得できます。要求でプロビジョニングデータがサービスプロファイルリファレンスとともに提供されているにもか

かわらず、ユーザがデータを提供した場合、要求が送信されてプロビジョニングが実行される前に、サービスプロファイルに格納されているデータがユーザ提供データとマージされて、プロファイルデータがオーバーライドされます。サービスプロファイルは、EPNM サービスプロファイルウィザード GUI を使用して作成できます。

- 顧客向けサービス (CFS) : CFS は回線/VC の顧客向けデータを表します。CFS は検出された RFS から派生し、ネットワーク内の回線/VC のエンドポイントを表します。CFS 検出時に、検出された RFS オブジェクトに対して CFS オブジェクトが作成されます。
- リソース向けサービス (RFS) : RFS は異なるデバイス上のリソース間の関係を表します。RFS 検出時に、デバイス レベルのオブジェクトとネットワーク レベルのオブジェクトが作成されます。デバイス レベルの RFS オブジェクトは、デバイス レベル設定の回線/VC 設定部分を表します。ネットワーク レベルの RFS オブジェクトは、デバイスまたはその他のネットワーク レベルのオブジェクトを集約して、ネットワーク レベルのエントリを表します。

認証および承認

Cisco EPN Manager API のすべての要求でユーザ認証が必要です。要求で認証の詳細が指定されていない場合、要求はログインページにリダイレクトされます。認証の詳細は、要求の HTTP ヘッダーを介して渡される場合があります。詳細については、Cisco EPN Manager API ドキュメントの「Authentication, Authorization, and Security」のトピック ([ホーム (Home)] > [認証、認可、セキュリティ (Authentication, Authorization, and Security)]) を参照してください。

Cisco EPN Manager の場合、API へのアクセスは、NBI 読み取り、NBI 書き込み、NBI クレデンシャル (廃止) などのユーザグループによって制御されます。これらの各グループは、異なる API セットへのアクセスを制御します。必要に応じて、複数のグループにユーザを割り当てることができます。API リソースのドキュメント ページで、アクセスに必要なユーザグループを確認できます。



(注) 実稼働環境では、JSESSIONID を使用することをお勧めします。

Cisco EPN Manager REST API スタートアップガイド

Cisco EPN Manager REST API を使用すると、アプリケーションはプログラムによって Cisco EPN Manager とやり取りできます。これらの要求を通じて、Cisco EPN Manager のリソースへのアクセス権が付与されます。API コールを使用して、Cisco EPN Manager ワークフロー、アラームとイベントのモニタリング、デバイスインベントリの収集、ネットワーククライアントと使用状況のモニタリング、デバイスの設定、デバイスインベントリなどを実行できます。詳細については、Cisco Evolved Programmable Network Manager API ドキュメントの「Getting Started」トピック (Home > Getting Started) を参照してください。

REST API の基本および機能エリア

Cisco EPN Managerの REST 実装では、複数のコールとフィルタを使用します。たとえば統計情報を取得する場合、最初のコールは後続のコールの URL を提供します。これらの URL は、より詳細な情報を取得するために使用できます。一般に、詳細情報が追加されるにつれて、コールごとに URL パラメータが複雑になります。

REST API の基本

- HTTPS ヘッダー：次の HTTP ヘッダーを使用して、データがクライアントに返される方法を制御します。
 - Accept
 - Accept-Language
 - Content-Type
 - Accept-Encoding
 - Content-Encoding
- クエリ パラメータ：

APIは、ほぼすべての要求のクエリーパラメータをサポートしています。次の表で、一般的な REST クエリ パラメータについて説明します。

表 2: 一般的な REST クエリ パラメータ

クエリー パラメータ	適用性	意味
.json	/api/*	指定されている場合は、応答を json 形式で返す必要があります。
.xml	/api/*	指定されている場合は、応答を xml 形式で返す必要があります。これは、.json パラメータが存在しない場合のデフォルトです。
._docs	/api/*	ドキュメント ページを表示します。
.full	/api/v1/data/T	「true」の場合、エンティティの ID だけではなくオブジェクト全体が返されます。
.group	api/v1/data/T	文字列値で指定されたグループの内容と、適用された演算子の結果に基づいて応答をフィルタ処理します。

クエリーパラメータ	適用性	意味
.transform	GET/POST	XML または JSON にレンダリングする直前に適用される変換の論理名を指定します。
.maxresults	GET Paged	返されたインスタンス ツリーのヘッドに関する結果の最大数。
.firstResult	GET Paged	インスタンス ツリーのヘッドに関する最初の結果。
.strict	/api/v1/data/T	「true」の場合は、クエリで使用されるプロパティ名が検証され、必要に応じてエラーがスローされます (デフォルトの strict=false の場合、無効なプロパティ名が無視されて簡単なログメッセージが表示されます)。
.case_sensitive	/api/v1/data/T	「true」の場合は、フィルタクエリで使用される文字列値の大文字と小文字が区別されます (デフォルトの .case_sensitive=false の場合は、大文字と小文字を区別せずに比較を行います)。
.nocount	/api/v1/data/T	「true」の場合、「count」、「first」、および「last」属性は応答に含まれません。これらの属性の値を取得するには余分に時間がかかるため、「true」に設定しておくパフォーマンスが向上します。デフォルトは「false」です。

クエリーパラメータ	適用性	意味
_ctx.domain	GET	アクティブなドメインとしてクエリパラメータ値で指定されたドメインを設定します。デフォルトは「sticky」で、1度設定するとアクティブドメインが設定されたままになります。「stickiness」はRESTfulではないため、設定でオフにすることができます。

REST API 機能エリア

- 統計：統計サービスは、システムに関する概要、事前定義された統計情報を提供します。統計のリソースの一部を以下に示します。
 - GET All Border Routers
 - GET All IMEs
 - GET All RCs
 - GET All TCAs
 - GET All WANInterfaces
- レポート サービス：レポート サービスは、レポートを検出および実行する操作を行えるようにします。APIを介してアクセスする前に、レポートをシステムで定義する必要があります。次のAPIがサポートされています。
 - GET Get Available Report Templates
 - (廃止) GET Get a Report
 - GET Get a ZIP Report
 - GET Run a ZIP Report
- CLI テンプレート コンフィギュレーション：CLI テンプレート コンフィギュレーション サービスを使用すると、CLI コンフィギュレーションテンプレートを1台以上のターゲットデバイスに適用できます。また、システム内のCLIテンプレートをアップロード、削除、取得することもできます。次のAPIがサポートされています。
 - GET CLI Configuration Templates
 - DELETE Delete Configuration Template
 - DELETE Delete Configuration Template Folder
 - GET Download Configuration Template
 - GET List Configuration Template Folders

- GET List Configuration Templates
- GET List Device Types
- POST Create Configuration Template Folder
- POST Upload Configuration Template
- PUT Deploy Configuration Template
- PUT Deploy Configuration Template Through Job
- PUT Modify Configuration Template Content

統計情報

EPNM から簡単な段階的プロセスで統計データを取得できます。これは、複数の REST 統計エンドポイント間でかなり一貫性のある一般的な使用パターンです。一般に、概略的な最初のコールでは、使用可能なメトリックのリストが返されます。そのコールから特定のメトリックに関する詳細な統計の 2 番目のリストが返され、そこから実際のデータ系列が返されます。

統計の例「エンドポイント コールの継承：特定の回線の ESR PM データ」を次に示します。

```
GET /webacs/api/v1/op/statisticsService/circuits?circuitName=VS05_TO_HUB2
GET
/webacs/api/v1/op/statisticsService/circuits/metrics?circuitType=ODUUNI&circuitId=161374613

GET
/webacs/api/v1/op/statisticsService/circuits/metrics/ESR_PM?maxResults=24&timeInterval=6&endpoint-
Name=ODU20/4/0/0/1&location=FEND&circuitType=ODUUNI&deviceId=124606492_10.201.1.174&circuitId=161374613
```



(注) 最後の 2 つのコールの URL は、前のコールの本文で提供されました。
