



# OpenFlow

このモジュールでは、デバイスで OpenFlow を有効化して設定する方法について説明します。

- [OpenFlow の前提条件](#) (1 ページ)
- [OpenFlow の制約事項](#) (1 ページ)
- [OpenFlow について](#) (2 ページ)
- [OpenFlow の設定方法](#) (7 ページ)
- [OpenFlow の確認](#) (12 ページ)
- [OpenFlow の設定例](#) (14 ページ)
- [その他の参考資料](#) (15 ページ)
- [OpenFlow の機能情報](#) (16 ページ)

## OpenFlow の前提条件

デバイスを OpenFlow モードで起動する必要があります。(OpenFlow モードは、デバイスで `boot mode openflow` コマンドを設定すると有効になります。すべてのポートがこのモードになり、デバイスは通常の Cisco IOS XE 機能をサポートしなくなります)。

## OpenFlow の制約事項

- デバイスで OpenFlow モードを有効にする場合は、以前の設定をすべて消去し、フラッシュファイルシステムから `vlan.dat` ファイルと `stby-vlan.dat` ファイルを削除します。
- デバイスが Openflow モードの場合、デバイスが通常モードのときに機能する他のコントロールプレーンプロトコル、ボーダーゲートウェイプロトコル (BGP)、スパンニングツリープロトコル (STP)、ポートチャネル、StackWise Virtualなどを有効にしないでください。

# OpenFlow について

## OpenFlow の概要

OpenFlow は Open Networking Foundation (ONF) の仕様で、フローベースの転送インフラストラクチャと、標準化されたアプリケーションプログラムインターフェイスを定義します。OpenFlow では、セキュアなチャンネルを介して、デバイスのフォワーディング機能を方向付けすることができます。

OpenFlow は、コントローラ（コントロールプレーン）とイーサネットスイッチ（データプレーン）の間のプロトコルです。スイッチには、パイプラインに配置されたフローテーブルがあります。フローとは、これらのテーブルに到達するパケットを調べるためのルールです。

スイッチ上の OpenFlow エージェントは、OpenFlow プロトコルを使用してコントローラと通信します。エージェントは、OpenFlow 1.0（有線プロトコル 0x1）と OpenFlow 1.3（有線プロトコル 0x4）の両方をサポートしています。最大 8 つのコントローラを接続できます。これらの接続はスイッチオーバー後は維持されず、コントローラはエージェントに再接続する必要があります。

Cisco Catalyst 9400 シリーズ スイッチでの OpenFlow の実装はステートレスです。ノンストップフォワーディング (NSF) はサポートされていません。スタンバイのスーパーバイザは、ローデータベースと同期しません。

## Openflow コントローラ

Openflow コントローラは、Openflow プロトコルを使用して Openflow スイッチとやり取りするエンティティです。ほとんどの場合、コントローラは多数の Openflow 論理スイッチを管理するソフトウェアです。コントローラではネットワークを一元的に表示でき、管理者はこれを使用して、ネットワークトラフィックの処理方法について基盤となるシステム（スイッチおよびルータ）に指示を出すことができます。通常、コントローラは Linux サーバで実行され、OpenFlow 対応スイッチに IP 接続できる必要があります。

コントローラはスイッチを管理し、スイッチ上でフローを挿入および削除します。これらのフローは、OpenFlow 1.3 および 1.0 の「照合」と「アクション」の基準のサブセットをサポートしています。

スイッチは、管理ポートを使用してコントローラに接続します。管理ポートは管理用の Virtual Routing and Forwarding (VRF) インスタンスの中にあり、そのためコントローラへのセキュアな接続を提供します。コントローラをスイッチに接続するには、コントローラへの到達が可能な IP アドレスとポート番号を設定します。

## フローの管理

フロー エントリは、パケットの照合と処理に使用されるフロー テーブル内の要素です。これには、照合設定の優先順位レベル、パケットを照合するための一連の照合フィールド、適用す

る一連の命令、パケットカウンタ、およびバイトカウンタが含まれています。また、フローごとにタイムアウト（ハードタイムアウトまたは非アクティブタイムアウト）も関連付けられており、フローの自動削除に使用されます。

Cisco Catalyst 9000 シリーズ スイッチは最大 9 個のフロー テーブルをサポートしています。

各フローは次の情報を提供します。

- 優先順位：優先順位の高いフローが先に照合されます。フローの更新では、設定された優先順位に基づいて、すべてのフローに優先順位を付ける必要があります。
- 照合フィールド：パケットを照合する際のフローエントリの一部。照合フィールドは、さまざまなパケットヘッダー フィールドと照合できます。フィールドに照合情報が指定されていない場合は、ワイルドカードが使用されます。
- アクション：パケットに対して作用する操作。

## OpenFlow パイプライン

OpenFlow パイプラインは、リンクされたフローテーブルのセットで、OpenFlow スイッチでの照合、転送、およびパケット変更を提供します。ポートは、パケットがパイプラインに出入りする場所です。

パケットは入力ポートで受信され、出力ポートに転送される OpenFlow パイプラインによって処理されます。パケット入力ポートは、パイプライン全体でパケットによって所有され、パケットがスイッチに受信されたポートを表します。入力ポートは、フローの一致フィールドとしても使用できることに注意してください。

フローアクションを使用すると、パケットをパイプライン内の後続のテーブルに送信して処理したり、テーブル間で情報をやり取りしたりすることができます。一致するフローエントリに関連付けられたアクションが次のテーブルを指定しない場合、パイプライン処理は停止します。この時点で、パケットは通常変更され、転送されます。パケットはドロップすることもできます。

OpenFlow スイッチのフローテーブルには、0 から順に番号が付けられます。パイプライン処理は常に、フローテーブル 0 のフローエントリに対してパケットを照合することから始まります。最初のテーブルの一致とアクションの結果に応じて、他のフローテーブルを使用できます。その結果、後続のテーブルのフローエントリとパケットが一致する可能性があります。

## サポートされる Match フィールドとアクション

Match フィールドは、パケットヘッダーと入力ポートを含む、パケットが照合されるフィールドです。Match フィールドは、ワイルドカード（任意の値に一致）にすることができ、フィールドの選択されたビットに一致するビットマスクを指定できます。

アクションは、パケットをポートまたは後続のテーブルに転送する操作、またはパケットフィールドを変更する操作です。アクションは、フローエントリに関連付けられた命令の一部、またはグループエントリに関連付けられたアクションパケットとして指定できます。グループエントリは、複数のフローで共有できるアクションの集合です。

1つ以上のフローエントリで指定されたアクションは、グループアクションと呼ばれる基本アクションにパケットを転送できます。グループアクションの目的は、複数のフロー間で一連のアクションを共有することです。グループは1つ以上のバケットで構成され、バケットは一連のアクション（set、pop、または output）を持つことができます。Cisco Catalyst 9000 シリーズスイッチは、グループタイプ *all* および *indirect* をサポートします。

次に、サポートされる match フィールドとアクションの一覧を示します。

表 1: サポートされる match フィールド

ヘッダー フィールド	前提条件	マスク可能なエントリ	値の例
イーサネットの宛先 MAC アドレス	—	あり	01:80:c2:00:00:00/ ff:ff:ff:00:00:00 (マスクあり)  de:f3:50:c7:e2:b2 (マスクなし)
イーサネットの送信元 MAC アドレス	—	あり	0e:00:00:00:00:019 (マスクなし)
イーサネットの種類	—	—	ARP (0x0806)、IPv4 (0x0800)、IPv6 (0x86dd) など
VLAN ID	—	—	0x13f
ARP ターゲット プロトコル アドレス	イーサネットタイプは 0x0806 に設定する必要があります	あり	—
IP プロトコル	イーサネットタイプは 0x0800 または 0x86dd に設定する必要があります	—	ICMP (0x01)、TCP (0x06)、UDP (0x11) など
IPv4 発信元アドレス	イーサネットタイプは 0x0800 に設定する必要があります	あり	10.0.0.0/24 (マスクあり)
IPv4 宛先アドレス	イーサネットタイプは 0x0800 に設定する必要があります	あり	10.0.0.254 (マスクなし)
IPv6 送信元アドレス	イーサネットタイプは 0x08dd に設定する必要があります	あり	2001:DB8::1 (マスクなし)

ヘッダー フィールド	前提条件	マスク可能なエントリ	値の例
IPv6 宛先アドレス	イーサネットタイプは 0x08dd に設定する必要があります	あり	2001:DB8:0:ABCD::1/48 (マスクあり)
ネイバー探索ターゲット	イーサネットタイプは 0x08dd に設定し、IP プロトコルは 0x01 に設定する必要があります	—	ND ターゲット
ICMPv6 タイプ	イーサネットタイプは 0x08dd に設定し、IP プロトコルは 0x01 に設定する必要があります	—	—
UDP/TCP 送信元ポート	イーサネットタイプは 0x0800 または 0x86dd に設定し、プロトコルは 0x06 または 0x11 に設定する必要があります	—	—
UDP/TCP 宛て先ポート	イーサネットタイプは 0x0800 または 0x86dd に設定し、プロトコルは 0x06 または 0x11 に設定する必要があります	—	—
着信インターフェイス	—	—	—

### サポートされるアクション

フローは次の宛先にパケットを送信できます。

- コントローラ
- スイッチの任意のインターフェイス（着信インターフェイスを含む）。
- 別のルックアップ用の後続のフローテーブル（テーブル 0 の後）。
- グループ。
- ローカル処理用のスイッチ CPU。ローカル処理のために送信できるのは、Cisco Discovery Protocol および Link Layer Discovery Protocol (LLDP) パケットのみです。

フローは、パケットの次のヘッダーフィールドも変更できます。

- イーサネットの宛先 MAC アドレス
- イーサネットの送信元 MAC アドレス
- VLAN ID

フローによって VLAN タグを追加 (push) または削除 (pop) できます。パケットが IP パケットの場合は、フローによって存続可能時間 (TTL) ヘッダーフィールドの値を減らすことができます。

## フローの操作

ここでは、フローが OpenFlow デバイスでプログラムされるようにコントローラから送信されるときに実行される操作について説明します。

通常デバイスには、パイプラインに配置されたフローテーブルがあります。パイプライン機能情報は、パイプラインの構造を指定します。たとえば、テーブルまたはステージの数、各ステージが実行できる機能 (照合またはアクション)、各テーブルのサイズなどがあります。

コントローラがフロー要求を送信すると、OpenFlow エージェントは、ハードウェアがフローを処理できるかどうかを確認します。エージェントは、スイッチの起動時に定義されるハードウェアの機能とフローとを比較します。フローが有効であれば、該当するフローテーブルにプログラムされます。

新しいパイプラインが検証された場合 (ハードウェアがパイプラインをサポートできるかどうか)、そのパイプラインは、フローをインストールできるかどうかのチェックに使用される新しい機能セットになります。

パイプラインがインスタンス化され、フローがインストールされると、パケットがスイッチから転送されます。優先順位の最も高い、一致するフローエントリが見つかるまで、入力パケットが各フローテーブル内のフローと照合されます。パケットの照合は、完全一致の場合もあれば (テーブルのすべてのフィールドが正確に一致する)、部分一致の場合もあります (一部または全部のフィールドに一致し、ビットマスクを持つフィールドが部分的に一致する場合があります)。設定されたアクションに基づいて、パケットが変更されるか転送される場合があります。アクションは、パイプライン内でいつでも適用できます。アクションによって、次の照合対象のフローテーブル、パケットの出力ポートのセット、およびパケットをコントローラにルーティングするかどうかが決まる場合があります。

## OpenFlow テーブル パイプライン

OpenFlow テーブル機能要求メッセージを使用すると、OpenFlow コントローラから、OpenFlow が管理するデバイスについて既存のフローテーブルの機能を照会したり、指定した設定と一致するようにこれらのテーブルを設定したりできます。

テーブルはすべて、照合およびアクション機能のサブセットを使用して設定できます。テーブルのサイズを実行時に変更することもできます。新しいフローテーブル設定が正常に適用されると、古いフローテーブルのフローエントリが通知なく削除されます。動的に設定されたフ

ローテーブルは、再起動後は維持されません。デバイスが起動するとデフォルトのパイプラインが起動します。

OpenFlow コントローラからの要求に基づいて新しいフローテーブルを設定している間は、既存のフローの中を流れる進行中のトラフィックがあると、いずれもドロップされます。

## OpenFlow Power over Ethernet

OpenFlow は Power Over Ethernet (PoE) をサポートします。PoE を機能させるには、デバイスで Cisco Discovery Protocol または LLDP を設定し、Cisco Discovery Protocol パケットまたは LLDP パケットがデバイスによって処理（および送信）されるようにします。PoE を OpenFlow で機能させるために、OpenFlow 固有の設定は必要ありません。

OpenFlow コントローラで、*output-to-local* アクションを使用してフローを設定し、パケットがローカル処理のためにデバイス CPU に送信されるようにします。

PoE の詳細については、「*POE の設定*」の章を参照してください。

## OpenFlow の設定方法

ここでは、OpenFlow のさまざまな設定作業について説明します。

### デバイスでの OpenFlow モードの有効化

スイッチが通常モードで動作している場合は、以前の設定を削除するように **write erase** コマンドを設定することをお勧めします。

#### 手順の概要

1. **enable**
2. **configure terminal**
3. **boot mode openflow**
4. **exit**
5. **write erase**
6.     • **delete flash:vlan.dat**  
       • **delete flash:stby-vlan.dat**
7. **reload**
8. **enable**
9. **show boot mode**

#### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例 :	特権 EXEC モードを有効にします。

	コマンドまたはアクション	目的
	Device> enable	<ul style="list-style-type: none"> <li>パスワードを入力します（要求された場合）。</li> </ul>
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>boot mode openflow</b> 例： Device(config)# boot mode openflow	OpenFlow 転送モードをイネーブルにします。
ステップ 4	<b>exit</b> 例： Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。
ステップ 5	<b>write erase</b> 例： Device# write erase	NVRAM 内のすべてのファイルを消去します。 <ul style="list-style-type: none"> <li>デバイスが以前に通常モードで動作していた場合は、すべてのファイルを消去することをお勧めします。</li> </ul>
ステップ 6	<ul style="list-style-type: none"> <li>• <b>delete flash:vlan.dat</b></li> <li>• <b>delete flash:stby-vlan.dat</b></li> </ul> 例： Device# delete flash:vlan.dat Device# delete flash:stby-vlan.dat	<ul style="list-style-type: none"> <li>• VLAN 情報を保存する vlan.dat ファイルを削除します。</li> <li>• スタンバイデバイスがある場合は、stby-vlan.dat ファイルを削除します。</li> </ul>
ステップ 7	<b>reload</b> 例： Device# reload	スイッチをリロードし、スイッチの OpenFlow フォワーディング モードを有効にします。
ステップ 8	<b>enable</b> 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> <li>パスワードを入力します（要求された場合）。</li> </ul>
ステップ 9	<b>show boot mode</b> 例： Device# show boot mode	デバイスのフォワーディングモードに関する情報を表示します。

### 例

次の **show boot mode** コマンドの出力例は、デバイスが OpenFlow モードであることを示しています。



```
Device# show boot mode
```

```
System initialized in openflow forwarding mode
System configured to boot in openflow forwarding mode
```

### 次のタスク

通常モードに戻るには、**no boot mode openflow** コマンドを設定して、デバイスをリロードします。

## OpenFlow の設定

### 手順の概要

1. **enable**
2. **configure terminal**
3. **feature openflow**
4. **openflow**
5. **switch 1 pipeline 1**
6. **controller ipv4 ip-address port port-number vrf vrf-name security {none | tls}**
7. **datapath-id ID**
8. **tls trustpoint local name remote name**
9. **end**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>feature openflow</b> 例： Device(config)# feature openflow	OpenFlow 機能をイネーブルにします。
ステップ 4	<b>openflow</b> 例： Device(config)# openflow	OpenFlow 設定をイネーブルにし、OpenFlow コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 5	<b>switch 1 pipeline 1</b> 例： Device(config-openflow)# switch 1 pipeline 1	論理スイッチとパイプラインを設定し、OpenFlow のスイッチ コンフィギュレーション モードを開始します。
ステップ 6	<b>controller ipv4 ip-address port port-number vrf vrf-name security {none   tls}</b> 例： Device(config-openflow-switch)# controller ipv4 10.2.2.2 port 6633 vrf Mgmt-vrf security tls	コントローラに接続します。  <ul style="list-style-type: none"> <li>• OpenFlow コントローラの接続セキュリティオプションとして TLS を設定している場合は、<b>tls trustpoint</b> コマンドを設定する必要があります。</li> <li>• OpenFlow コントローラのセキュリティオプションを設定していない場合は、<b>tls trustpoint</b> コマンドを設定する必要はありません。</li> </ul>
ステップ 7	<b>datapath-id ID</b> 例： Device(config-openflow-switch)# datapath-id 0x12345678	(任意) OpenFlow の論理スイッチ ID を設定します。  <ul style="list-style-type: none"> <li>• <b>ID</b> 引数にはスイッチ ID を指定します。これは 16 進値です。</li> </ul>
ステップ 8	<b>tls trustpoint local name remote name</b> 例： Device(config-openflow-switch)# tls trustpoint local trustpoint1 remote trustpoint1	(任意) OpenFlow スイッチの Transport Layer Security (TLS) トラストポイントを設定します。
ステップ 9	<b>end</b> 例： Device(config-openflow-switch)# end	OpenFlow スイッチのコンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

## OpenFlow モードでのインターフェイスの設定

OpenFlow モードでは、レイヤ 2 またはレイヤ 3 インターフェイスを設定できます。レイヤ 3 インターフェイスを使用する場合は、インターフェイス コンフィギュレーション モードで **no switchport** コマンドを設定します。レイヤ 2 インターフェイスを使用する場合は、次の作業を実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **feature openflow**
4. **interface type number**
5. **switchport mode trunk**
6. **switchport nonnegotiate**

7. **no keepalive**
8. **spanning-tree bpdudfilter enable**
9. **end**

## 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。  • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>feature openflow</b> 例： Device(config)# feature openflow	OpenFlow 機能をイネーブルにします。
ステップ 4	<b>interface type number</b> 例： Device(config)# interface gigabitethernet 1/0/3	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	<b>switchport mode trunk</b> 例： Device(config-if)# switchport mode trunk	レイヤ 2 スイッチドインターフェイスのトランキングモードをトランクに設定します。
ステップ 6	<b>switchport nonnegotiate</b> 例： Device(config-if)# switchport nonnegotiate	装置がこのインターフェイスのネゴシエーションプロトコルに関係しないように指定します。
ステップ 7	<b>no keepalive</b> 例： Device(config-if)# no keepalive	キープアライブパケットをディセーブルにします。
ステップ 8	<b>spanning-tree bpdudfilter enable</b> 例： Device(config-if)# spanning-tree bpdudfilter enable	インターフェイスでブリッジプロトコルデータユニット (BPDU) フィルタリングをイネーブルにします。
ステップ 9	<b>end</b> 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

# OpenFlow の確認

OpenFlow の設定を確認するには、次のコマンドを使用します。コマンドはどの順序で使用してもかまいません。

## 手順の概要

1. **enable**
2. **show openflow hardware capabilities**
3. **show openflow switch 1 controller**
4. **show openflow switch 1 ports**
5. **show openflow switch 1 flows list**

## 手順の詳細

---

### ステップ1 enable

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

### ステップ2 show openflow hardware capabilities

OpenFlow デバイスのハードウェア機能を表示します。

例：

```
Device# show openflow hardware capabilities

Max Interfaces: 1000
Aggregated Statistics: YES

Pipeline ID: 1
  Pipeline Max Flows: 2322
  Max Flow Batch Size: 100
  Statistics Max Polling Rate (flows/sec): 10000
  Pipeline Default Statistics Collect Interval: 5

Flow table ID: 0

  Max Flow Batch Size: 100
  Max Flows: 1022
  Bind Subintfs: FALSE
  Primary Table: TRUE
  Table Programmable: TRUE
  Miss Programmable: TRUE
  Number of goto tables: 1
  Goto table id: 1
  Number of miss goto tables: 1
  Miss Goto table id: 1
```

```

Stats collection time for full table (sec): 1
.
.
.

```

### ステップ3 show openflow switch 1 controller

スイッチに接続されているコントローラに関する情報を表示します。

例：

```

Device# show openflow switch 1 controller

Logical Switch Id: 1
Total Controllers: 1
Controller: 1
10.10.23.200:6633
Protocol: tcp
VRF: Mgmt-vrf
Connected: Yes
Role: Equal
Negotiated Protocol Version: OpenFlow 1.3
Last Alive Ping: 2018-06-04 17:59:20 PDT
state: ACTIVE
sec_since_connect: 50

```

### ステップ4 show openflow switch 1 ports

OpenFlow スwitchのポートに関する情報を表示します。

例：

```

Device# show openflow switch 1 ports
Logical Switch Id: 1

```

Port	Interface Name	Config-State	Link-State	Features
1	Gi1/0/1	PORT_UP	LINK_UP	1GB-FD
2	Gi1/0/2	PORT_UP	LINK_UP	1GB-FD
3	Gi1/0/3	PORT_UP	LINK_UP	1GB-FD
4	Gi1/0/4	PORT_UP	LINK_UP	1GB-FD
5	Gi1/0/5	PORT_UP	LINK_DOWN	1GB-HD
6	Gi1/0/6	PORT_UP	LINK_DOWN	1GB-HD
7	Gi1/0/7	PORT_UP	LINK_DOWN	1GB-HD
8	Gi1/0/8	PORT_UP	LINK_DOWN	1GB-HD
9	Gi1/0/9	PORT_UP	LINK_UP	1GB-FD
10	Gi1/0/10	PORT_UP	LINK_UP	1GB-FD
11	Gi1/0/11	PORT_UP	LINK_UP	1GB-FD
12	Gi1/0/12	PORT_UP	LINK_UP	1GB-FD
13	Gi1/0/13	PORT_UP	LINK_DOWN	1GB-HD
14	Gi1/0/14	PORT_UP	LINK_DOWN	1GB-HD
15	Gi1/0/15	PORT_UP	LINK_DOWN	1GB-HD
16	Gi1/0/16	PORT_UP	LINK_DOWN	1GB-HD
17	Gi1/0/17	PORT_UP	LINK_DOWN	1GB-HD
18	Gi1/0/18	PORT_UP	LINK_DOWN	1GB-HD
19	Gi1/0/19	PORT_UP	LINK_UP	1GB-FD
20	Gi1/0/20	PORT_UP	LINK_UP	1GB-FD
21	Gi1/0/21	PORT_UP	LINK_UP	1GB-FD
22	Gi1/0/22	PORT_UP	LINK_UP	1GB-FD
23	Gi1/0/23	PORT_UP	LINK_DOWN	1GB-HD
24	Gi1/0/24	PORT_UP	LINK_DOWN	1GB-HD
25	Gi1/1/1	PORT_UP	LINK_DOWN	1GB-HD
26	Gi1/1/2	PORT_UP	LINK_DOWN	1GB-HD
27	Gi1/1/3	PORT_UP	LINK_DOWN	1GB-HD
28	Gi1/1/4	PORT_UP	LINK_DOWN	1GB-HD

```

29          Tel/1/1          PORT_UP      LINK_DOWN   10GB-FD
30          Tel/1/2          PORT_UP      LINK_DOWN   10GB-FD
31          Tel/1/3          PORT_UP      LINK_DOWN   10GB-FD
32          Tel/1/4          PORT_UP      LINK_DOWN   10GB-FD
33          Tel/1/5          PORT_UP      LINK_DOWN   10GB-FD
34          Tel/1/6          PORT_UP      LINK_DOWN   10GB-FD
35          Tel/1/7          PORT_UP      LINK_DOWN   10GB-FD
36          Tel/1/8          PORT_UP      LINK_DOWN   10GB-FD
37          Fol/1/1          PORT_UP      LINK_DOWN   40GB-FD
38          Fol/1/2          PORT_UP      LINK_DOWN   40GB-FD
39          Twel/1/1         PORT_UP      LINK_DOWN   10GB-FD
40          Twel/1/2         PORT_UP      LINK_DOWN   10GB-FD

```

## ステップ5 show openflow switch 1 flows list

OpenFlow のエントリを表示します。

次の出力例は、テーブル0で利用可能なフローを示しています。*match any*はテーブル1に移動します（「match any」とは、すべてのパケットがテーブル1に移動するという意味です）。テーブル1では、宛先 MAC アドレス 00:00:01:00:00:01 が照合され、出力ポートが 36 に設定されます。

例：

```

Device# show openflow switch 1 flows list

Logical Switch Id: 1
Total flows: 8

Flow: 1 Match: any Actions: goto_table:1, Priority: 9000, Table: 0, Cookie: 0x1,
Duration: 2382.117s, Packets: 34443, Bytes: 3359315

Flow: 2 Match: any Actions: drop, Priority: 0, Table: 0, Cookie: 0x0,
Duration: 2382.118s, Packets: 294137, Bytes: 28806211

Flow: 3 Match: any Actions: drop, Priority: 0, Table: 1, Cookie: 0x0,
Duration: 2382.118s, Packets: 34443, Bytes: 3359315

Flow: 4 Match: dl_dst=00:00:01:00:00:01 Actions: output:36, Priority: 9000,
Table: 1, Cookie: 0x1, Duration: 2382.116s, Packets: 0, Bytes: 0

```

# OpenFlow の設定例

## 例：デバイスでの OpenFlow の有効化

次に、OpenFlow を有効にする例を示します。

```

Device> enable
Device# configure terminal
Device(config)# boot mode openflow
Device(config)# exit
Device# write erase
Device# delete flash:vlan.dat

```

```
Device# reload
Device> enable
Device# show boot mode
```

## 例：OpenFlow の設定

次に、OpenFlow を設定する例を示します。

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# controller ipv4 10.2.2.2 port 6633 vrf Mgmt-vrf security
tls
Device(config-openflow-switch)# datapath-id 0x12345678
Device(config-openflow-switch)# tls trustpoint local trustpoint1 remote trustpoint1
Device(config-openflow-switch)# end
```

## その他の参考資料

### 関連資料

関連項目	マニュアルタイトル
OpenFlow のコマンド	<a href="#">プログラマビリティ コマンド リファレンス</a>
Open Network Foundation	<a href="https://www.opennetworking.org/">https://www.opennetworking.org/</a>
Faucet OpenFlow コントローラ	<ul style="list-style-type: none"> <li>• <a href="https://faucet.nz/">https://faucet.nz/</a></li> <li>• <a href="https://docs.faucet.nz/en/latest/">https://docs.faucet.nz/en/latest/</a></li> </ul>
PoE	<ul style="list-style-type: none"> <li>• Cisco Catalyst 9300 シリーズ スイッチでの「<a href="#">Power over Ethernet の設定</a>」</li> <li>• Cisco Catalyst 9400 シリーズ スイッチでの「<a href="#">PoE の設定</a>」</li> </ul>

## シスコのテクニカル サポート

説明	リンク
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。</p>	<p><a href="http://www.cisco.com/support">http://www.cisco.com/support</a></p>

## OpenFlow の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、[www.cisco.com/go/cfn](http://www.cisco.com/go/cfn) に移動します。Cisco.com のアカウントは必要ありません。



表 2: OpenFlow の機能情報

機能名	リリース	機能情報
OpenFlow	Cisco IOS XE Fuji 16.9.1	<p>OpenFlow は Software Defined Networking (SDN) の標準規格であり、SDN 環境での通信プロトコルを定義します。これにより、SDN コントローラは、スイッチやルータなどのネットワーク デバイスのフォワーディングプレーンと直接やり取りできるようになります。</p> <p>この機能は、次のプラットフォームに実装されていました。</p> <ul style="list-style-type: none"> <li>• Catalyst 9300 シリーズ スイッチ</li> <li>• Catalyst 9400 シリーズ スイッチ</li> <li>• Catalyst 9500 シリーズ スイッチ</li> <li>• Catalyst 9500 シリーズハイ パフォーマンス スイッチ</li> </ul>
	Cisco IOS XE Gibraltar 16.10.1	<p>Catalyst 9500 シリーズハイ パフォーマンススイッチでのテーブル機能メッセージのサポートが導入されました。</p>
OpenFlow Power over Ethernet	Cisco IOS XE Gibraltar 16.12.1	<p>PoE は OpenFlow ポートでサポートされます。</p> <p>この機能は、次のプラットフォームに実装されていました。</p> <ul style="list-style-type: none"> <li>• Catalyst 9300 シリーズ スイッチ</li> <li>• Catalyst 9400 シリーズ スイッチ</li> </ul>



## 翻訳について

このドキュメントは、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。