Cisco IOS XE Fuji 16.9.x（Catalyst 9300 スイッチ）Network Powered Lighting コンフィギュレーション ガイド

初版：2018 年 7 月 18 日

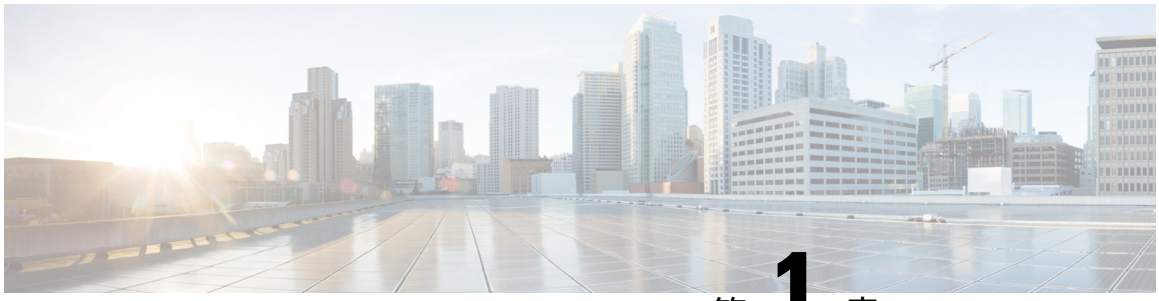
最終更新：2019 年 7 月 9 日

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/

目 次

第 I 部

Network Powered Lighting



第 1 章

COAP プロキシ サーバーの設定

COAP プロキシ サーバの制約事項

次の制約事項は、COAP プロキシ サーバに適用されます。

- スイッチは、ipv6 ブロードキャスト（CSCuw26467）を使用する CoAP クライアントとして自身をアドバタイズできません。
- 監視のサポートは実装されていません。
- Blockwise 要求はサポートされていません。シスコは、block-wise 応答を処理し、block-wise 応答を生成できます。
- DTLS サポートは、RawPublicKey および証明書ベースのモードに対してのみ有効です。
- スイッチは、DTLS クライアントとして動作しません。DTLS はエンドポイントに対してのみ。
- エンドポイントは、CBOR ペイロードを処理し、応答すると想定されています。
- クライアント側要求は、JSON であると想定されています。
- IPv6 ブロードキャストの問題により、スイッチは IPv6 として他のリソース ディレクトリに自身をアドバタイズすることはできません。

COAP プロキシ サーバについて

COAP プロトコルは、制限されたデバイスで使用できるように設計されています。HTTP が情報にアクセスする際にサーバ上で動作するのと同じ方法で、COAP は制限されたデバイス上で動作します。

COAP と HTTP の比較を次に示します。

- Web サーバの場合、プロトコル は **HTTP**、トランスポートは **TCP**、転送される最も一般的な情報の形式は **HTML** です。
- 制約付きデバイスの場合、プロトコルは **COAP**、トランスポートは **UDP**、一般的な情報の形式は **JSON/link-format/CBOR** です。

COAP によって、HTTP の場合と同様に **GET**/**POST** メタファーと RESTful API を使用してデバイスにアクセスし、管理する手段が提供されます。

COAP プロキシ サーバの設定方法

COAP プロキシ サーバを設定するには、コンフィギュレーション モードで COAP プロキシと COAP エンドポイントを設定できます。

コマンドは **coap [proxy | endpoints]** です。

COAP プロキシの設定

スイッチで COAP プロキシを開始または停止するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **coap proxy**
4. **security** [**none** [[**ipv4** | **ipv6**] {*ip-address ip-mask/prefix*} | **list** {*ipv4-list name* | *ipv6-list-name*}] | **dtls** [**id-trustpoint** {*identity-trustpoint label*}] [**verification-trustpoint** {*verification-trustpoint*} | [**ipv4** | **ipv6** {*ip-address ip-mask/prefix*}] | **list** {*ipv4-list name* | *ipv6-list-name*}]]
5. **max-endpoints** {*number*}
6. **port-unsecure** {*port-num*}
7. **port-dtls** {*port-num*}
8. **resource-directory** [ipv4 | ipv6] {*ip-address*}]
9. **list** [**ipv4** | **ipv6**] {*list-name*}
10. **start**
11. **stop**
12. **exit**
13. **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	**enable** 例： デバイス> **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）。
ステップ 2	**configure terminal** 例： デバイス# **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	**coap proxy** 例： デバイス(config)# **coap proxy**	COAP プロキシ サブモードを開始します。 （注） coap proxy を停止して、coap proxy の下にあるすべての設定を削除するには、**no coap proxy** コマンドを使用します。
ステップ 4	**security** [**none** [[**ipv4** | **ipv6**] {*ip-address ip-mask/prefix*} | **list** {*ipv4-list name* | *ipv6-list-name*}] | **dtls** [**id-trustpoint** {*identity-trustpoint label*}] [**verification-trustpoint** {*verification-trustpoint*} | [**ipv4** | **ipv6** {*ip-address ip-mask/prefix*}] | **list** {*ipv4-list name* | *ipv6-list-name*}]] 例： デバイス(config-coap-proxy)# **security none ipv4 1.1.0.0 255.255.0.0**	暗号化タイプを引数と見なします。サポートされる 2 つのセキュリティ モードは **none** と **dtls** です。 • none：そのポートにセキュリティがないことを示します。 **security none** を使用すると、最大 5 つの IPv4 アドレスと最大 5 つの IPv6 アドレスを関連付けることができます。 • dtls：DTLS セキュリティは、オプションである RSA トラストポイントと検証トラストポイントを要します。検証トラストポイントがないと、通常の公開キー交換が行われます。 **security dtls** を使用すると、最大 5 つの IPv4 アドレスと最大 5 つの IPv6 アドレスを関連付けることができます。 （注） coap proxy のすべてのセキュリティ設定を削除するには、**no security** コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 5	**max-endpoints** {*number*} 例： `デバイス(config-coap-proxy)#max-endpoints 10`	（任意）スイッチで学習できるエンドポイントの最大数を指定します。デフォルト値は 10 です。指定できる範囲は 1 ～ 500 です。 （注） coap proxy に設定されたすべての最大エンドポイントを削除するには、**no max-endpoints** コマンドを使用します。
ステップ 6	**port-unsecure** {*port-num*} 例： `デバイス(config-coap-proxy)#port-unsecure 5683`	（任意）デフォルト 5683 以外のポートを設定します。指定できる範囲は 1 ～ 65000 です。 （注） coap proxy のすべてのポート設定を削除するには、**no port-unsecure** コマンドを使用します。
ステップ 7	**port-dtls** {*port-num*} 例： `デバイス(config-coap-proxy)#port-dtls 5864`	（任意）デフォルト 5684 以外のポートを設定します。 （注） coap proxy のすべて DTLS のポート設定を削除するには、**no port-dtls** コマンドを使用します。
ステップ 8	**resource-directory** [ipv4 | ipv6] {*ip-address*}] 例： `デバイス(config-coap-proxy)#resource-directory ipv4 192.168.1.1`	スイッチが COAP クライアントとして動作できるユニキャスト アップストリーム リソースのディレクトリ サーバを設定します。 **resource-directory** を使用すると、最大 5 つの IPv4 アドレスと最大 5 つの IPv6 アドレスを設定できます。 （注） coap proxy のすべてのリソース ディレクトリ設定を削除するには、**no resource-directory** コマンドを使用します。
ステップ 9	**list** [**ipv4 | ipv6**] {*list-name*} 例： `デバイス(config-coap-proxy)#list ipv4 trial_list`	（任意）ライトとリソースを学習できる IP アドレス範囲を制限します。上記の **security** [**none** | **dtls**] コマンドオプションで使用する、IP アドレス/マスクの名前付きリストを作成します。 **list** を使用して、IPv4 または IPv6 に関係なく、最大 5 つの IP リストを設定できます。IP リストにつき最大 5 つの IP アドレスを設定できます。 （注） COAP プロキシサーバの IP リストを削除するには、 **no list** [**ipv4 | ipv6**] {*list-name*} コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 10	**start** 例： デバイス(config-coap-proxy)#**start**	このスイッチで COAP プロキシを開始します。
ステップ 11	**stop** 例： デバイス(config-coap-proxy)#**stop**	このスイッチで COAP プロキシを停止します。
ステップ 12	**exit** 例： デバイス(config-coap-proxy)# **exit**	COAP プロキシ サブモードを終了します。
ステップ 13	**end** 例： デバイス(config)# **end**	特権 EXEC モードに戻ります。

COAP エンドポイントの設定

複数の IPv4/IPv6 スタティック エンドポイントをサポートするように COAP プロキシを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **coap endpoint** [**ipv4** | **ipv6**] {*ip-address*}
4. **exit**
5. **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	**enable** 例：	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）。

	コマンドまたはアクション	目的
	デバイス> **enable**	
ステップ 2	**configure terminal** 例： デバイス# **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	**coap endpoint** [**ipv4** | **ipv6**] {*ip-address*} 例： デバイス(config)#**coap endpoint ipv4 1.1.1.1** デバイス(config)#**coap endpoint ipv6 2001::1**	スイッチ上でスタティック エンドポイントを設定します。 • **ipv4** ：IPv4 スタティック エンドポイントを設定します。 • **ipv6** ：IPv6 スタティック エンドポイントを設定します。 （注） エンドポイントで coap proxy を停止するには、**no coap endpoint** [**ipv4** |**ipv6**] {*ip-address*} コマンドを使用します。
ステップ 4	**exit** 例： デバイス(config-coap-endpoint)# **exit**	COAP エンドポイント サブモードを終了します。
ステップ 5	**end** 例： デバイス(config)# **end**	特権 EXEC モードに戻ります。

COAP プロキシサーバーの設定例

例：COAP プロキシ サーバの設定

次の例に、最大 10 のエンドポイントをサポートするようにポート番号 5683 を設定する方法を示します。

```
デバイス#coap proxy security none ipv4 2.2.2.2 255.255.255.0 port 5683 max-endpoints 10
---------------------------------------------------------------------------------------
```

次の例に、セキュリティ設定がされて**いない** *ipv4 1.1.0.0 255.255.0.0* に COAP プロキシを設定する方法を示します。

```
デバイス(config-coap-proxy)# security ?
  dtls  dtls
  none  no security

デバイス(config-coap-proxy)#security none ?
  ipv4    IP address range on which to learn lights
  ipv6    IPv6 address range on which to learn lights
  list    IP address range on which to learn lights

デバイス(config-coap-proxy)#security none ipv4 ?
  A.B.C.D  {/nn || A.B.C.D}  IP address range on which to learn lights

デバイス(config-coap-proxy)#security none ipv4 1.1.0.0 255.255.0.0
```

次の例に、**dtls id trustpoint** セキュリティ設定がされている *ipv4 1.1.0.0 255.255.0.0* に COAP プロキシを設定する方法を示します。

```
デバイス(config-coap-proxy)#security dtls ?
  id-trustpoint DTLS RSA and X.509 Trustpoint Labels
  ipv4 IP address range on which to learn lights
  ipv6 IPv6 address range on which to learn lights
  list IP address range on which to learn lights

デバイス(config-coap-proxy)#security dtls id-trustpoint ?
  WORD  Identity TrustPoint Label

デバイス(config-coap-proxy)#security dtls id-trustpoint RSA-TRUSTPOINT ?
  verification-trustpoint  Certificate Verification Label
  <cr>

デバイス(config-coap-proxy)#security dtls id-trustpoint RSA-TRUSTPOINT

デバイス(config-coap-proxy)#security dtls ?
  id-trustpoint DTLS RSA and X.509 Trustpoint Labels
  ipv4 IP address range on which to learn lights
  ipv6 IPv6 address range on which to learn lights
  list IP address range on which to learn lights

デバイス(config-coap-proxy)# security dtls ipv4 1.1.0.0 255.255.0.0
```

（注） **ipv4 / ipv6 / list** を設定するには、**id-trustpoint** と（任意）**verification-trustpoint** を事前に設定しておく必要があります。設定していない場合はエラーが表示されます。

次の例に、トラストポイントを設定する方法を示します。これは、**id trustpoint** 設定の COAP **security dtls** の前提条件です。

```
ip domain-name myDomain
```

```
crypto key generate rsa general-keys exportable label MyLabel modulus 2048

デバイス(config)#crypto pki trustpoint MY_TRUSTPOINT
デバイス(ca-trustpoint)#rsakeypair MyLabel 2048
デバイス(ca-trustpoint)#enrollment selfsigned
デバイス(ca-trustpoint)#exit

デバイス(config)#crypto pki enroll MY_TRUSTPOINT
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Generate Self Signed Router Certificate? [yes/no]: yes
```

次の例に、**dtls verification trustpoint** によって *ipv4 1.1.0.0 255.255.0.0* に COAP プロキシを設定する方法を示します（証明書または検証トラストポイントによる DTLS）。

```
デバイス(config-coap-proxy)#security dtls ?
  id-trustpoint DTLS RSA and X.509 Trustpoint Labels
  ipv4 IP address range on which to learn lights
  ipv6 IPv6 address range on which to learn lights
  list IP address range on which to learn lights

デバイス(config-coap-proxy)#security dtls id-trustpoint ?
  WORD  Identity TrustPoint Label

デバイス(config-coap-proxy)#security dtls id-trustpoint RSA-TRUSTPOINT ?
  verification-trustpoint  Certificate Verification Label
  <cr>

デバイス(config-coap-proxy)#security dtls id-trustpoint RSA-TRUSTPOINT
verification-trustpoint ?
  WORD  Identity TrustPoint Label

デバイス(config-coap-proxy)#security dtls id-trustpoint RSA-TRUSTPOINT
verification-trustpoint CA-TRUSTPOINT ?
  <cr>
```

次の例に、検証トラストポイントを設定する方法を示します。これは、**verification trustpoint** 設定の COAP **security dtls** の前提条件です。

```
デバイス(config)#crypto pki import CA-TRUSTPOINT pkcs12 flash:hostA.p12 password cisco123
% Importing pkcs12...
Source filename [hostA.p12]?
Reading file from flash:hostA.p12
CRYPTO_PKI: Imported PKCS12 file successfully.
```

次の例に、セキュリティ [none | dtls] コマンド オプションで使用する、trial-list という名前のリストを作成する方法を示します。

```
デバイス(config-coap-proxy)#list ipv4 trial_list
デバイス (config-coap-proxy-iplist)#1.1.0.0 255.255.255.0
デバイス (config-coap-proxy-iplist)#2.2.0.0 255.255.255.0
デバイス (config-coap-proxy-iplist)#3.3.0.0 255.255.255.0
デバイス (config-coap-proxy-iplist)#exit
デバイス (config-coap-proxy)#security none list trial_list
```

次の例に、coap プロキシ サブ モードで使用できるすべての拒否コマンドを示します。

```
デバイス(config-coap-proxy)#no ?
  ip-list             Configure IP-List
  max-endpoints       maximum number of endpoints supported
  port-unsecure       Specify a port number to use
  port-dtls           Specify a dtls-port number to use
  resource-discovery  Resource Discovery Server
  security            CoAP Security features
```

次の例に、coap プロキシで複数の IPv4/IPv6 スタティック エンドポイントを設定する方法を示します。

```
デバイス (config)# coap endpoint ipv4 1.1.1.1
デバイス (config)# coap endpoint ipv4 2.1.1.1
デバイス (config)# coap endpoint ipv6 2001::1
```

次の例に、COAP プロトコルの詳細を表示する方法を示します。

```
デバイス#show coap version
CoAP version 1.0.0
RFC 7252
```

```
デバイス#show coap resources
Link format data =
</>
</1.1.1.6/cisco/context>
</1.1.1.6/cisco/actuator>
</1.1.1.6/cisco/sensor>
</1.1.1.6/cisco/lldp>
</1.1.1.5/cisco/context>
</1.1.1.5/cisco/actuator>
</1.1.1.5/cisco/sensor>
</1.1.1.5/cisco/lldp>
</cisco/flood>
</cisco/context>
</cisco/showtech>
</cisco/lldp>
```

```
デバイス#show coap globals
Coap System Timer Values :
   Discovery  : 120 sec
   Cache Exp  : 5 sec
   Keep Alive : 120 sec
   Client DB  : 60 sec
```

```
   Query Queue: 500 ms
   Ack delay  : 500 ms
   Timeout    : 5 sec

Max Endpoints      : 10
Resource Disc Mode : POST

-------------------------------------------------------------------------------------

デバイス#show coap stats
Coap Stats :
Endpoints : 2
Requests : 20
Ext Queries : 0

-------------------------------------------------------------------------------------

デバイス#show coap endpoints
List of all endpoints :

Code : D - Discovered , N - New
#    Status   Age(s)  LastWKC(s)  IP
-------------------------------------------------------------------------------
1    D        10      94          1.1.1.6
2    D        6       34          1.1.1.5

Endpoints - Total : 2 Discovered : 2 New : 0

-------------------------------------------------------------------------------------

デバイス#show coap dtls-endpoints
#     Index State  String State     Value   Port IP
-----------------------------------------------------------------
1     3     SSLOK       3           48969   20.1.1.30
2     2     SSLOK       3           53430   20.1.1.31
3     4     SSLOK       3           54133   20.1.1.32
4     7     SSLOK       3           48236   20.1.1.33


-------------------------------------------------------------------------------------
```

次の例に、COAP プロトコルのデバッグに使用できるすべてのオプションを示します。

```
デバイス#debug coap ?
all       Debug CoAP all
database  Debug CoAP Database
errors    Debug CoAP errors
events    Debug CoAP events
packet    Debug CoAP packet
trace     Debug CoAP Trace
warnings  Debug CoAP warnings
```

COAP プロキシ サーバーのモニタリング

COAP プロトコルの詳細を表示するには、次の表のコマンドを使用します。

表 *1 : COAP* 固有のデータを表示するコマンド

show coap version	IOS COAP バージョンと RFC 情報を表示します。

show coap resources	スイッチのリソースと、スイッチが学習したリソースを表示します。
show coap endpoints	検出され、学習されたエンドポイントを表示します。
show coap globals	タイマー値とエンドポイント値を表示します。
show coap stats	エンドポイント、要求、および外部クエリのメッセージ数を表示します。
show coap dtls-endpoints	dtls エンドポイントのステータスを表示します。

表 *2 : COAP* コマンドをクリアするコマンド

clear coap database	スイッチで学習された COAP、およびエンドポイント情報の内部データベースをクリアします。

COAP プロトコルをデバッグするには、次の表のコマンドを使用します。

表 *3 : COAP* プロトコルをデバッグするコマンド

debug coap database	COAP データベース出力をデバッグします。
debug coap errors	COAP エラー出力をデバッグします。
debug coap events	COAP イベント出力をデバッグします。
debug coap packets	COAP パケット出力をデバッグします。
debug coap trace	COAP トレース出力をデバッグします。
debug coap warnings	COAP 警告出力をデバッグします。
debug coap all	すべての COAP 出力をデバッグします。

（注） デバッグを無効にする場合は、コマンドの前に「**no**」キーワードを追加します。

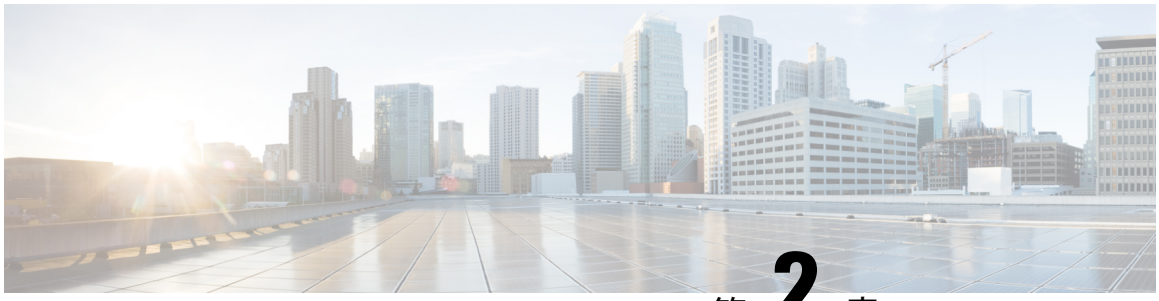
COAP の機能情報

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 に移動します。Cisco.com のアカウントは必要ありません。

表 *4 : COAP* の機能情報

機能名	リリース	機能情報
COAP	Cisco IOS XE Everest 16.5.1a	COAP プロトコルは、制限されたデバイスで使用できるように設計されています。HTTP が情報にアクセスする際にサーバ上で動作するのと同じ方法で、COAP は制限されたデバイス上で動作します。



第 2 章

Auto SmartPorts の設定

Auto SmartPorts の設定の制約事項

Auto SmartPort は Cisco スイッチを検出しますが、イベントトリガーを自動的に呼び出しません。スイッチをマクロにマッピングするには、イベントトリガーを手動で呼び出す必要があります。

Auto SmartPorts に関する情報

Auto SmartPort マクロは、ポートで検出されたデバイス タイプに基づいてポートを動的に設定します。スイッチがポートで新しいデバイスを検出すると、適切な Auto SmartPorts マクロを適用します。ポート上でリンク ダウン イベントが発生した場合、スイッチはそのマクロを削除します。たとえば、ポートに Cisco IP Phone を接続した場合は、Auto SmartPorts により自動的に Cisco IP Phone マクロが適用されます。Cisco IP Phone マクロが適用されると、遅延に影響されやすい音声トラフィックを正しく処理できるように QoS（Quality Of Service）、セキュリティ機能、および専用の音声 VLAN がイネーブルになります。

Auto SmartPorts は、イベントトリガーを使用して、マクロにデバイスをマッピングします。最も一般的なイベントトリガーは、接続されているデバイスから受信した Cisco Discovery Protocol（CDP）メッセージに基づいています。デバイス（Cisco IP Phone、Cisco ワイヤレスアクセス

ポイント、または Cisco ルータ）の検出は、そのデバイスのイベントトリガーを呼び出します。

Link Layer Discovery Protocol（LLDP）は、CDP をサポートしないデバイスを検出するために使用されます。イベント トリガーとして使用される他のメカニズムには、802.1X 認証結果と学習した MAC アドレスなどがあります。

主に CDP および LLDP メッセージと MAC アドレスに基づいて、さまざまなデバイス用にシステムの組み込みイベント トリガーがあります。これらのトリガーは、Auto SmartPort が有効になっている限り有効になっています。

プロファイルとデバイス用のユーザ定義のトリガー グループを設定できます。トリガー グループ名を使用してユーザ定義マクロを関連付けます。

Auto SmartPort マクロ

Auto SmartPort マクロは CLI コマンドのグループです。ポートのデバイスが検出されると、デバイスにマクロが適用されます。システムの組み込みマクロはさまざまなデバイスに存在し、デフォルトでは、システムの組み込みのトリガーは、対応する組み込みマクロにマッピングされます。必要に応じて、組み込みのトリガーまたはマクロのマッピングを変更できます。

マクロは、基本的に、リンク ステータスに基づいて、インターフェイスの CLI のセットを適用または削除します。マクロでは、リンク ステータスがチェックされます。リンクがアップ ステータスの場合は、CLI のセットが適用されます。リンクがダウンしている場合、セットが削除されます（CLI の no 形式が適用されます）。CLI のセットを適用するマクロの部分は、マクロと呼びます。CLI を削除する部分（CLI の no 形式）は、アンチマクロと呼びます。

デバイスが Auto SmartPort に接続されている場合に、点灯しているエンドポイントとして分類されると、イベントトリガー **CISCO_LIGHT_EVENT** が呼び出され、マクロ **CISCO_LIGHT_AUTO_SMARTPORT** が実行されます。

CISCO_LIGHT_AUTO_SMARTPORT によって実行されるコマンド

マクロが実行されると、スイッチで一連のコマンドが実行されます。

マクロ **CISCO_LIGHT_AUTO_SMARTPORT** を実行することで実行されるコマンドは、次のとおりです。

- switchport mode access
- switchport port-security violation restrict
- switchport port-security mac-address sticky
- switchport port-security
- power inline port poe-ha

- storm-control broadcast level 50.00
- storm-control multicast level 50.00
- storm-control unicast level 50.00
- spanning-tree portfast
- spanning-tree bpduguard enable

Auto SmartPort の有効化

（注） Auto SmartPort はデフォルトで無効になっています。

特定のポートの Auto SmartPorts マクロをディセーブルにするには、Auto SmartPort をグローバルにイネーブルにする前に、**no macro auto global processing** インターフェイス コマンドを使用します。

Auto SmartPort をグローバルにイネーブルにするには、**macro auto global processing** グローバル コンフィギュレーション コマンドを使用します。

Auto SmartPorts をイネーブルにするには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **device classifier**
4. **macro auto global processing**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `デバイス> 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）。
ステップ 2	**configure terminal** 例： `デバイス# 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。

	コマンドまたはアクション	目的
ステップ 3	**device classifier** 例： `デバイス(config)# device classifier`	デバイスの分類子を有効にします。 デバイス分類子を無効にするには、**no device classifier** コマンドを使用します。
ステップ 4	**macro auto global processing** 例： `デバイス(config)# macro auto global processing`	スイッチの Auto SmartPorts をグローバルにイネーブルにします。 Auto SmartPort をグローバルに無効にするには、**no macro auto global processing** コマンドを使用します。
ステップ 5	**end** 例： `デバイス(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `デバイス# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `デバイス# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

イベントトリガーと組み込みマクロ間のマッピングの設定

（注） Cisco スイッチが Auto SmartPort に接続されている場合は、このタスクを実行する必要があります。

組み込みマクロにイベントトリガーをマッピングするには、次の作業を行います。

始める前に

auto smartport マクロをグローバルに有効にする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **macro auto execute** *event trigger* **builtin** *built-in macro name*
4. **macro auto trigger** *event trigger*
5. **device** *device_ID*
6. **end**
7. **show shell triggers**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> 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）。
ステップ 2	**configure terminal** 例： `Switch# 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	**macro auto execute** *event trigger* **builtin** *built-in macro name* 例： `Switch(config)# macro auto execute CISCO_SWITCH_EVENT builtin CISCO_SWITCH_AUTO_SMARTPORT`	ユーザ定義のイベントトリガーとマクロ名を指定します。このアクションは、イベントトリガーから組み込み Auto Smartport マクロへのマッピングを設定します。
ステップ 4	**macro auto trigger** *event trigger* 例： `Switch(config)# macro auto trigger CISCO_SWITCH_EVENT`	ユーザ定義イベントトリガーを呼び出します。
ステップ 5	**device** *device_ID* 例： `Switch(config)# device cisco WS-C3560CX-8PT-S`	イベントトリガーをデバイス ID と照合します。
ステップ 6	**end** 例： `Switch(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show shell triggers** 例：	スイッチ上のイベント トリガーを表示します。

	コマンドまたはアクション	目的
	`Switch# show shell triggers`	
ステップ 8	**show running-config** 例： `Switch# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `Switch# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

例：Auto SmartPorts の有効化

この例では、Auto SmartPort を有効にする方法を示します。

```
デバイス> enable
デバイス# configure terminal
デバイス(config)# device classifier
デバイス(config)# macro auto global processing
デバイス(config)# end
```

例：イベントトリガーと組み込みマクロ間のマッピングの設定

この例では、イベントトリガーと組み込みマクロ間のマッピングを設定する方法を示します。

```
Switch> enable
Switch# configure terminal
Switch(config)# macro auto execute CISCO_SWITCH_EVENT builtin CISCO_SWITCH_AUTO_SMARTPORT
Switch(config)# macro auto trigger CISCO_SWITCH_EVENT
Switch(config)# device cisco WS-C3560CX-8PT-S
Switch(config)# end
```

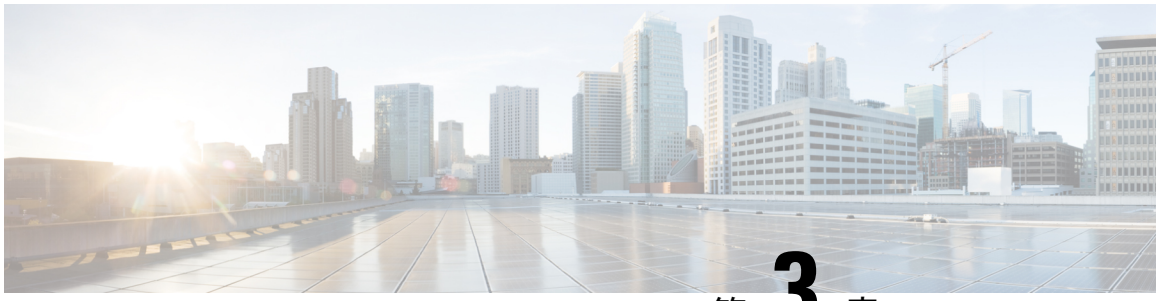
Auto SmartPorts の機能情報

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 に移動します。Cisco.com のアカウントは必要ありません。

表 *5 : Auto SmartPorts* の機能情報

機能名	リリース	機能情報
自動 SmartPorts	Cisco IOS XE Everest 16.5.1a	Auto SmartPort マクロは、ポートで検出されたデバイスタイプに基づいてポートを動的に設定します。スイッチがポートで新しいデバイスを検出すると、適切な Auto SmartPorts マクロを適用します。



第 3 章

2 イベント分類の設定

2 イベント分類の制約事項

2 イベント分類には次の制約が適用されます。

- 2 イベント分類の設定は、エンドポイントを物理的に接続する前に行っておく必要があります。または、電力を供給しているポートの手動 shut/no-shut を行います。
- ポートへの電力供給は MCU ファームウェアのアップグレード時には中断され、ポートはアップグレード直後にバックアップされます。

2 イベント分類について

クラス 4 デバイスが検出されると、IOS は、CDP または LLDP のネゴシエーションを行うことなく 30W を割り当てます。これは、リンクがアップする前であっても、クラス 4 の電源デバイスは 30W を得ることを意味します。

また、ハードウェア レベルで、PSE は 2 イベント分類を行い、これにより、クラス 4 PD はハードウェアから 30W を供給する PSE の能力を検出し、それ自体を登録することができます。また、CDP/LLDP パケット交換を待つことなく最大 PoE+ レベルまで移動できます。

2 イベントがポートで有効になったら、ポートの遮断または開放を手動で行うか、または PD を再度接続して IEEE 検出を再度開始する必要があります。2 イベント分類がポートで有効になっている場合、クラス 4 デバイスの電力バジェット割り当ては 30W です。その他の場合は 15.4W です。

2 イベント分類の設定

2 イベント分類についてスイッチを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port 2-event**
5. **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	**enable** 例： `デバイス> 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）。
ステップ 2	**configure terminal** 例： `デバイス# 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	**interface** *interface-id* 例： `デバイス(config)# interface gigabitethernet2/0/1`	設定する物理ポートを指定し、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 4	**power inline port 2-event** 例： `デバイス(config-if)# power inline port 2-event`	スイッチで 2 イベント分類を設定します。
ステップ 5	**end** 例： `デバイス(config-if)# end`	特権 EXEC モードに戻ります。

例：2 イベント分類の設定

次に、2 イベント分類を設定する例を示します。

```
デバイス> enable
デバイス# configure terminal
デバイス(config)#  interface gigabitethernet2/0/1
デバイス(config-if)# power inline port 2-event
デバイス(config-if)# end
```

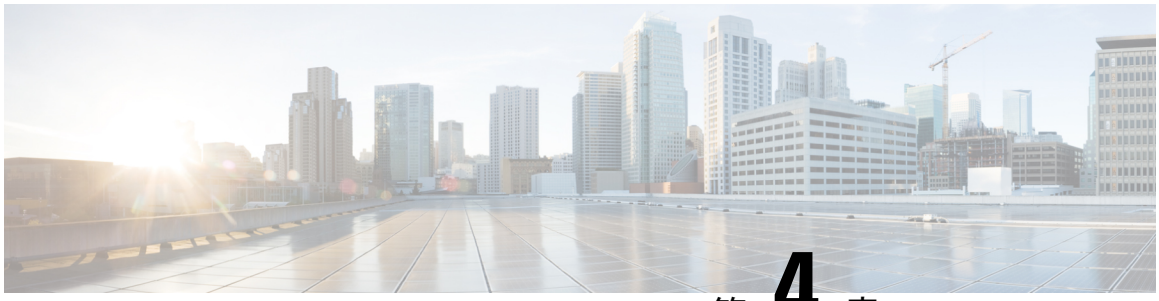
2 イベント分類の機能情報

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 に移動します。Cisco.com のアカウントは必要ありません。

表 *6 : 2* イベント分類の機能情報

機能名	リリース	機能情報
2 イベント分類	Cisco IOS XE Everest 16.5.1a	クラス 4 デバイスが検出されると、IOS は、CDP または LLDP のネゴシエーションを行うことなく 30W を割り当てます。これは、リンクがアップする前であっても、クラス 4 の電源デバイスは 30W を得ることを意味します。



第 4 章

無停止型 PoE および高速 POE の設定

無停止型および高速 PoE の制約事項

無停止型および高速 PoE には、次の制限が適用されます。

- 高速 PoE または無停止型 PoE の設定は、エンドポイントを物理的に接続する前に行う必要があります。または、電力を供給しているポートの手動 shut/no-shut を行います。
- ポートへの電力供給は MCU ファームウェアのアップグレード時には中断され、ポートはアップグレード直後にバックアップされます。
- スイッチが電源スタックされている場合、無停止型および高速 PoE 機能が期待どおりに動作しないことがあります。これは、パワーバジェットが不足しているためです。
- DHCP サーバーから割り当てられた IP が設定されていない場合、CREE ライト電力供給デバイス（PD）は定期的にフラップすることがあります。
- PD が LLDP をサポートしていない場合、ユーザーはスタティックまたは 2 イベントを設定して、PD 仕様に従って必要な電力を受け取ることができます。

無停止型 POE

無停止型 PoE は、電源装置（PSE）スイッチが起動中であっても、接続されている電源供給を受けるデバイス（PD）へ中断なく電力を提供します。

（注） ポートへの電力供給は MCU ファームウェアのアップグレード時には中断され、ポートはアップグレード直後にバックアップされます。

高速 POE

この機能は、IOS が起動するのを待機することなく、AC 電源が接続された瞬間（電源投入の 15 ～ 20 秒以内）に特定の PSE ポートから引き出された最後の電力を記憶し、電源をオンにします。**poe-ha** が特定のポートで有効な場合、電源障害後の復旧時に、IOS 転送が開始されるまでの短期間、スイッチが接続されているエンドポイントデバイスに電源を供給します。

（注） UPOE の場合、高速 POE はスイッチ側で使用可能ですが、UPOE 電力の可用性の信号伝達を LLDP に依存するため、PD エンドポイントは同様の機能を利用できない可能性があります。LLDP に依存する場合、IOS が起動して LLDP パケット交換が可能になり、UPOE 電力の可用性を信号で伝達できるようになるまで、PD エンドポイントはそのまま待機する必要があります。

無停止型および高速 PoE の設定

無停止型および高速 PoE を設定するには、次の手順を実行します。

（注） PD を接続する前に **perpetual-poe-ha** コマンドを設定する、または、**poe-ha** を設定した後にポートを手動で閉じる/開く必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port perpetual-poe-ha**
5. **power inline port poe-ha**
6. **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	**enable** 例： `Device> 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）。
ステップ 2	**configure terminal** 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	**interface** *interface-id* 例： `Device(config)# interface gigabitethernet 2/0/1`	設定する物理ポートを指定し、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 4	**power inline port perpetual-poe-ha** 例： `Device(config-if)# power inline port perpetual-poe-ha`	無停止型 PoE を設定します。PD デバイスに接続されたポートに無停止型 PoE を設定すると、リロード中に PD デバイスの電源がオンのままになります。
ステップ 5	**power inline port poe-ha** 例： `Device(config-if)# power inline port poe-ha`	高速 PoE を設定します。高速 PoE を設定する場合、スイッチの電源を再投入すると、IOS の起動を待たずに電源に接続してから 10 ー 15 秒以内に PD デバイスの電源がオンになります。 （注） **power inline port poe-ha** コマンドを使用して高速 PoE を設定する前に、**power inline port perpetual-poe-ha** コマンドを使用して無停止型 PoE を設定する必要があります。
ステップ 6	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

例：無停止型および高速 PoE の設定

次の例では、スイッチ上で無停止型 PoE を設定にする方法を示します。

```
デバイス> enable
デバイス# configure terminal
デバイス(config)# interface gigabitethernet2/0/1
デバイス(config-if)# power inline port perpetual-poe-ha
デバイス(config-if)# end
```

次の例では、スイッチ上で高速 PoE を設定にする方法を示します。

```
デバイス> enable
デバイス# configure terminal
デバイス(config)# interface gigabitethernet2/0/1
デバイス(config-if)# power inline port perpetual-poe-ha
デバイス(config-if)# power inline port poe-ha
デバイス(config-if)# end
```

次の例では、無停止型 PoE を設定する前に高速 PoE を設定した場合の動作を示します。

```
デバイス> enable
デバイス# configure terminal
デバイス(config)# interface gigabitethernet2/0/1
デバイス(config-if)# power inline port poe-ha
Interface Gi2/0/1:INFO: Please execute "power inline port
perpetual-poe-ha" configuration command when "power inline port poe-ha"
 is configured on the interface to enable fast poe
デバイス(config-if)# power inline port perpetual-poe-ha
デバイス(config-if)# end
```

次の例では、インターフェイスで高速 PoE を無効にせずに無停止型 PoE を無効にした場合の動作を示します。

```
デバイス> enable
デバイス# configure terminal
デバイス(config)# interface gigabitethernet2/0/1
デバイス(config-if)# power inline port poe-ha
Interface Gi2/0/1:INFO: Please execute "power inline port
perpetual-poe-ha" configuration command when "power inline port poe-ha"
 is configured on the interface to enable fast poe
デバイス(config-if)# power inline port perpetual-poe-ha
デバイス(config-if)# no power inline port poe-ha
デバイス(config-if)# power inline port poe-ha
デバイス(config-if)# no power inline port perpetual-poe-ha
Interface Gi2/0/1:INFO: Please execute "no power inline port poe-ha"
configuration command, as fast poe has no effect without "power inline
 port perpetual-poe-ha" configuration on the interface
デバイス(config-if)# end
```

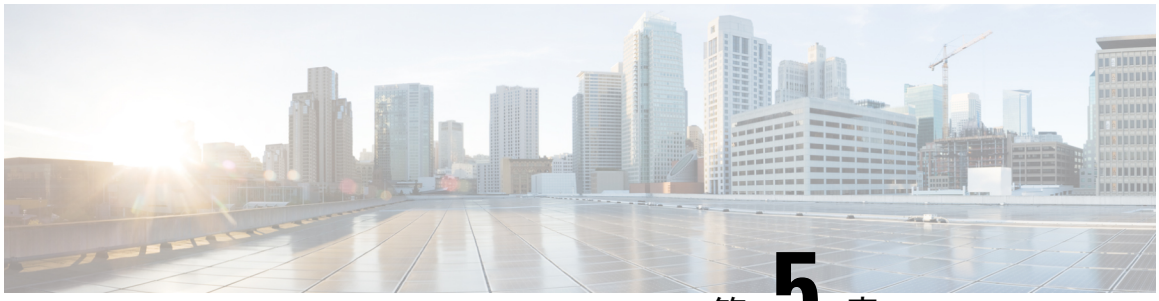
無停止型および高速 PoE の機能情報

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 に移動します。Cisco.com のアカウントは必要ありません。

表 *7*: 無停止型および高速 *PoE* の機能情報

機能名	リリース	機能情報
無停止型高速 POE	Cisco IOS XE Everest 16.5.1a	無停止型 POE は、PSE スイッチが起動している場合でも、接続された PD デバイスへの連続電源を提供します。 高速 PoE は、特定の PSE ポートから最後に供給された電力を記憶し、IOS が起動するのを待たずに電源をオンにします。



第 5 章

よく寄せられる質問

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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 には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

よく寄せられる質問

ここでは、Network Powered Lighting に関してよく寄せられる質問（FAQ）をまとめています。

- **質問:**

  「show coap stats」出力の「New Endpoint」は何を意味していますか。「New Endpoint」はいつ「Endpoint」に移行しますか。

  回答：

  新しいエンドポイントとは、エンドポイントが発見された（ディスカバリ パケットが受信された）が、CoAP プロキシによってまだ登録されていないことを意味します。CoAP プロキシは、定期的に新しいエンドポイントを調べ、「./well-known/core」上で GET を送信して詳細を取得します。そして RSP は受信された時点で、「Endpoint」に移動されます。

- **質問:**

  セキュリティ設定がないと「CoAP の開始」を実行できないのはなぜですか。

回答：

CoAP に関連するすべての設定が完了し、その後にそれが明示的に有効になるようにする必要があります。これによって、設定全体にわたる断続的に不安定な状態を回避できます。

- **質問:**

  「coap プロキシ」コンフィギュレーション モード「coap プロキシ <cr>」にドロップを強制する必要があるのはなぜですか。設定の完了後、スイッチプロンプトに戻るのに2度終了しなければなりません。これは非常に使いにくいと思います。

  回答：

  別の方法として、私たちが行っている各設定のプレフィックスとして「coap proxy」と入力する必要があります。coapプロキシに関するサブモード下のすべての設定を実行できるので、これはサブモードに入るのに最適なオプションです。

- **質問:**

  最初に coap プロセスを停止しないと、セキュリティやその他のパラメータを設定解除できないのはなぜですか。

  回答：

  CoAP に関連するすべての設定が完了し、その後にそれが明示的に有効になるようにする必要があります。これによって、CoAP が有効な場合に、ユーザがオンザフライで設定を行う可能性がある複雑性を回避して制御することもできます。

- **質問:**

  coap を停止したとき、CoAP プロセスに関連付けられたすべての設定が自動的に削除されません（またはデフォルトに戻ります）。CoAP はなぜ以前の設定を記憶しているのですか。これでは、ユーザはやり直すのが非常に難しいように思います。

  回答：

  システムは意図的にこのように設計されていて、これは予期された動作です。時々、最大エンドポイントの変更やプロキシの再起動など、軽微な変更だけを行いたい場合があります。これは、他のすべての設定はそのまま保持できるオプションです。これがないと、ユーザはすべてを一から設定し直す必要があります。

- **質問:**

  セキュリティ設定がどのように設定されているかはどのように確認できますか。

  回答：

  コマンド「show run」を使用してすべての設定を表示できます。

- **質問:**

  タイマー値はどのように調整できますか。

```
Example:
Device#sho coap glo
```

```
Coap System Timer Values:
Discovery : 120 sec
Cache Exp : 5 sec
Keep Alive : 120 sec
Client DB : 5 sec
Query Queue : 500 ms
Ack delay : 500 ms
Timeout : 5 sec
Max Endpoints : 500
Resource Disc Mode : POST
```

回答：

タイマー値は固定で、現在のところ調整不可です。その理由は、システム間での不一致を避けるためです。

- **質問:**

  コマンド「list」および「endpoint」は何に使用するものですか。

  回答：

  「list」コマンドは、複数の IP アドレスを設定し、それに名前を付ける作業をより簡単にするためのものです。その結果、複数の ip を表すために、単一の ip の代わりに名前を割り当てることができます。「endpoint」コマンドは、エンドポイントが自身をアドバタイズしない場合に、スタティック エンド ポイントを設定するために使用されます。

- **質問:**

  「show」コマンドを使用してエンドポイントからポートへのマッピングを見つけるにはどうすればよいですか。

  回答：

  それについては現時点でサポートされていません。しかし、他のコマンドを実行してそのデータを取得することができます。現在でも、「lldp neighbours」、「ip dhcp」、「power inlines」などの個々のコマンドを使用して、言及したすべての詳細を取得できます。

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。