



# ネットワーク プラグアンドプレイの使用

この章は、次の内容で構成されています。

- [ネットワーク プラグアンドプレイについて, on page 1](#)
- [ネットワーク プラグアンドプレイのトラブルシューティング例 \(12 ページ\)](#)

## ネットワーク プラグアンドプレイについて

ネットワーク プラグアンドプレイ (PnP) は、Cisco Nexus 9500 シリーズ スイッチ (具体的には、N9K-C9504、N9K-C9508、および N9K-C9516) で実行されるソフトウェアアプリケーションです。PnP は、新しいブランチやキャンパスの展開を容易にしたり、既存のネットワークに更新プログラムをプロビジョニングしたりするために、シンプルで安全な統一および統合された手法を提供します。この機能は、ほぼゼロタッチの展開エクスペリエンスを備えた複数のデバイスで構成されるネットワークをプロビジョニングするための統一されたアプローチを提供します。

簡素化された展開により、コストと複雑さが軽減され、展開の速度とセキュリティが向上します。PnP 機能はシスコのデバイスの展開の簡素化を、次の展開関連の運用タスクを自動化することにより支援します。

- デバイスの初期ネットワーク接続を確立する。
- コントローラにデバイス構成を配信する。
- コントローラにソフトウェアおよびファームウェアのイメージを配信する。
- スイッチのローカル クレデンシャルをプロビジョニングする。
- 展開関連のイベントについて他の管理システムに通知する。

PnP は、クライアント サーバー ベースのモデルです。クライアント (エージェント) は Cisco Nexus 9500 シリーズ スイッチで動作し、サーバー (コントローラ) は Cisco DNA コントローラで動作します。

PnP は、エージェントとコントローラ間の通信にセキュアな接続を使用します。この通信は暗号化されています。

PnP 機能に必要なセキュリティ証明書の構成と管理については、『[Cisco Digital Network Architecture Center セキュリティ ベスト プラクティス ガイド](#)』を参照してください。

PnP エージェントは、ネットワークに存在するソリューションを統合エージェントに収束し、現在のソリューションを強化する機能を追加します。PnP エージェントの主な目的は、すべての展開シナリオに一貫した Day 0 展開ソリューションを提供することです。

## ネットワーク プラグアンドプレイ (PnP) エージェントによって提供される機能

### Day 0 プロビジョニング

Day 0 ブートストラップには、構成、イメージ、およびその他のファイルが含まれます。デバイスに最初に電源を入れると、スタートアップ構成ファイルがない場合は、デバイスに組み込まれている PnP の検出プロセスが起動し、PnP コントローラまたはサーバーのアドレスの検出を試みます。PnP エージェントは DHCP やドメイン ネーム システム (DNS) などの方法で、PnP サーバーの目的の IP アドレスを取得します。

PnP エージェントが IP アドレスを正常に取得すると、サーバーとの長期間の双方向レイヤ 3 接続を開始し、サーバからのメッセージを待ちます。PnP サーバーアプリケーションは、デバイスで実行される情報とサービスを要求するメッセージをエージェントに送信します。

Cisco Nexus 9500 シリーズ スイッチで実行されているエージェントは、DHCP 確認応答の受信時に IP アドレスを設定し、設定をプロビジョニングするためにコントローラとの安全なチャネルを確立します。その後、スイッチはイメージをアップグレードし、構成を適用します。

### 検出方法

PnP エージェントは、次のいずれかの方法を使用して、PnP コントローラまたはサーバーを検出します。

- DHCP ベースの検出
- DNS ベースの検出
- PnP 接続

検出後、PnP エージェントは検出された情報をファイルに書き込みます。このファイルは、PnP サーバー (DNA コントローラ/DNA-C) とのハンドシェイクに使用されます。

次のタスクは、PnP 検出フェーズでエージェントによって実行されます。

- すべてのインターフェイスをアップにします。
- すべてのインターフェイスに対して並行して DHCP 要求を送信します。
- DHCP 応答を受信すると、IP アドレスとマスク、デフォルト ルート、DNS サーバー、ドメイン名を構成し、PnP サーバーの IP をリース解析ファイルに書き込みます。Cisco Nexus スイッチには DHCP クライアントがなく、静的構成が必要であることを注意してください。
- すべてのインターフェイスを停止します。



**Note** POAP は、Day 0 プロビジョニングの最初の選択肢です。有効な POAP オファーがない場合のみ、PnP 検出が試行されます。また、PnP は Cisco Nexus 9000 EoR モデル N9K-C9504、N9K-C9508、および N9K-C9516 でのみサポートされます。PnP は Cisco Nexus 9000 ToR ではサポートされません。

### DHCP ベースの検出

スイッチの電源が入っていて、スタートアップ構成がない場合、PnP は DHCP 検出から開始します。DHCP 検出は、PnP サーバー接続の詳細を取得します。

PnP エージェントは、以下を構成します。

- IP アドレス
- ネットマスク
- デフォルト ゲートウェイ
- DNS サーバ
- [ドメイン名 (Domain name) ]

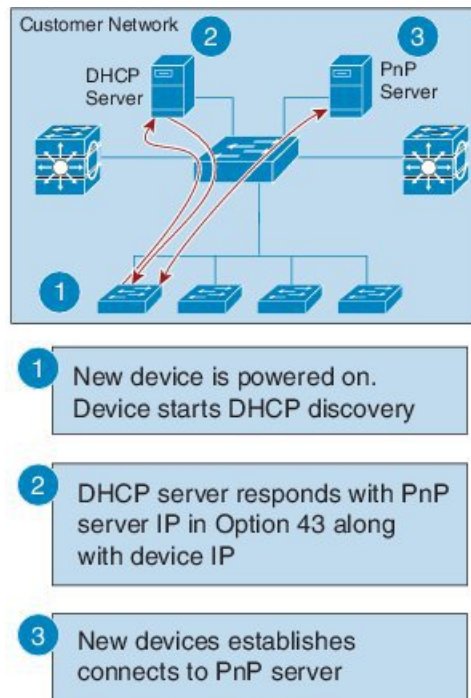
エージェントの構成が失敗した場合は、手動で介入してスイッチを構成する必要があります。

DHCP 検出には、次のフローがあります。

- スwitchの電源を投入します。
- 構成が存在しないため、スイッチが起動し、PnP プロセスが開始されます。
- DHCP 検出を開始します。
- DHCP サーバーは、PnP サーバー構成で応答します。
- PnP エージェントは PnP サーバーとハンドシェイクします。
- イメージをダウンロードし、インストールしてリロードします。
- コントローラから構成をダウンロードして適用します。

NV-RAM にスタートアップ構成がないデバイスは、Day 0 プロビジョニングをトリガーし、POAP プロセスを実行します ([m\\_using\\_poweron\\_auto\\_provisioning\\_92x.ditamap#id\\_70221](#) を参照)。有効な POAP オファーがない場合、PnP エージェントが開始されます。DHCP サーバーはベンダー固有のオプション 43 を使用して追加情報を挿入するように設定できます。DHCP サーバーは、文字列「cisco pnp」のあるデバイスからオプション 60 を受信した時点で、要求側のデバイスに PnP サーバーの IP アドレスまたはホスト名を渡します。デバイスが DHCP 応答を受信すると、PnP エージェントは応答からオプション 43 を抽出して、PnP サーバーの IP アドレスまたはホスト名を取得します。PnP エージェントは、PnP サーバーと通信するためにこの IP アドレスまたはホスト名を使用します。

Figure 1: PnPサーバーの DHCP 検出プロセス



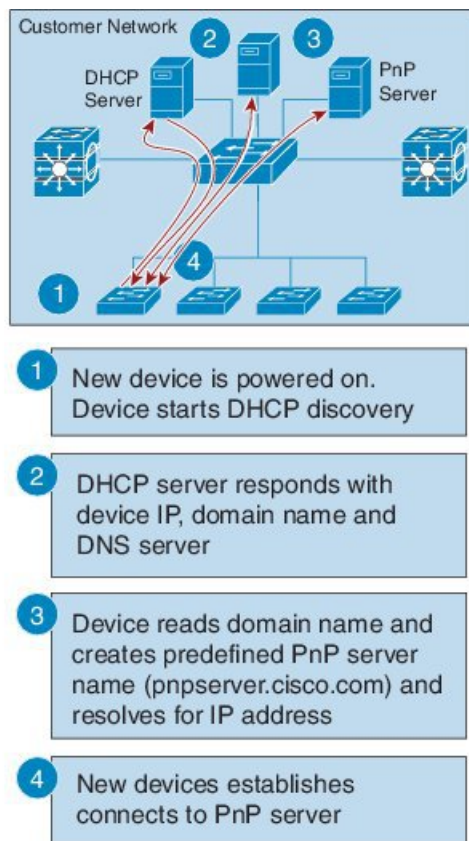
### DNS ベースの検出

DHCP 検出が PnP サーバーの取得に失敗すると、エージェントは DNS ベースの検出にフォールバックします。DNS ベースの検出を開始するには、DHCP から次の情報が必要です。

- IP アドレスおよびネットマスク
- デフォルト ゲートウェイ
- DNS サーバーの IP
- [ドメイン名 (Domain name) ]

エージェントは、DHCP 応答から顧客のネットワークのドメイン名を取得し、完全修飾ドメイン名 (FQDN) を形成します。次の FQDN は、DHCP 応答のプリセットの展開サーバ名とドメイン名情報を使用して PnP エージェントによって構成されます。次に、エージェントは、ローカル ネーム サーバでの検索を実行し、前述の FQDN の IP アドレスの解決を試みます。

Figure 2: pnpserver.[domainname].com の DNS ルックアップ



**Note** デバイスはドメイン名を読み取り、定義済みの PnP サーバー名を `pnpserver.[ドメイン名].com` などとして作成します。たとえば、`pnpserver.cisco.com` などです。

### プラグアンドプレイ接続

DHCP と DNS の検出が失敗すると、PnP エージェントは、初期展開のために Cisco Cloud ベースの展開サービスを検出して通信します。PnP エージェントは Python ライブラリを使用して HTTPS チャンネルを直接開き、内部で OpenSSL を呼び出して構成のためにクラウドと通信します。

### Cisco 電源オン自動プロビジョニング

Cisco 電源オン自動プロビジョニング (PoAP) は、DHCP および TFTP サーバーと通信して、イメージと構成をダウンロードします。PnP 機能の導入により、PnP と PoAP は Cisco Nexus 9500 シリーズスイッチで共存します。PoAP と PnP のインターワーキングには、次のプロセスがあります。

- PoAP は、システムにスタートアップ構成が存在しない場合に最初に開始されます。
- PoAP がプロビジョニングされていない場合、PnP は後で開始されます。
- PoAP と PnP は交互にコントローラを検出します。

- コントローラの検出プロセスは、コントローラが見つかるまで、または管理者が自動プロビジョニングを中止するまで続きます。
- コントローラを見つけたプロセス（POAP または PnP）はプロビジョニングを継続し、コントローラを見つけれなかった他のプロセスは通知を受け、最終的に終了します。

### ネットワーク プラグアンドプレイ エージェントのサービスと機能

PnP エージェントは、次のタスクを実行します。

- Backoff
- 機能
- CLI の実行
- 設定のアップグレード
- デバイス情報
- 証明書のインストール
- イメージのインストール
- リダイレクション



**Note** PnP コントローラまたはサーバーは、PnP エージェントによるイメージのインストールと構成のアップグレードサービス要求で使用されるオプションのチェックサムタグを提供します。チェックサムが要求に含まれている場合、イメージのインストールプロセスはそのチェックサムを実行中の現在のイメージのチェックサムと比較します。

チェックサムが同じである場合、インストールまたはアップグレードされるイメージは、デバイスで実行されている現在のイメージと同じです。このシナリオでは、イメージのインストールプロセスは他の操作を実行しません。

チェックサムが同じでない場合、新しいイメージがローカルファイルシステムにコピーされ、チェックサムが再度計算されて、要求で指定されたチェックサムと比較されます。同じ場合は、新しいイメージのインストールまたはデバイスの新しいイメージへのアップグレードが継続されます。チェックサムが異なる場合、プロセスはエラーで終了します。

### Backoff

PnP プロトコル（HTTP トランスポートを使用）をサポートする Cisco IOS デバイスでは、PnP エージェントが PnP サーバーに継続的に作業要求を送信する必要があります。PnP サーバに、PnP エージェントが実行するスケジュール済みまたは未処理の PnP サービスがない場合は、連続的な no operation 作業要求によってネットワーク帯域幅とデバイスリソースの両方が使い果たされます。この PnP バックオフサービスにより、PnP サーバは PnP エージェントに指定された時間だけ休止し、後でコールバックするように通知できます。

### 機能

機能サービス要求は、エージェントによってサポートされているサービスを照会するために、PnP サーバーによってデバイス上の PnP エージェントに送信されます。次に、サーバーはインベントリ サービス要求を送信して、デバイスのインベントリ情報を照会します。次に、イメージインストール要求を送信して、イメージをダウンロードしてインストールします。エージェ

ントからの応答を取得すると、サポートされている PnP サービスと機能のリストが登録され、サーバーに返されます。

### CLI の実行

Cisco NX-OS は、特権 EXEC モードとグローバル構成モードの 2 つのコマンド実行モードをサポートしています。EXEC コマンドのほとんどは、**show** コマンド（現在のコンフィギュレーションステータスを表示）、**clear** コマンド（カウンタまたはインターフェイスを消去）などのように、一回限りのコマンドです。EXEC コマンドは、デバイスをリブートするときには保存されません。構成モードでは、ユーザーが実行構成を変更できます。設定を保存すると、これらのコマンドはデバイスの再起動後も保存されます。

### 設定のアップグレード

シスコのデバイスで実行する可能性がある構成のアップグレードは 2 種類あります。1 つはスタートアップ構成への新しい構成ファイルのコピー、もう 1 つは実行構成への新しい構成ファイルのコピーです。

スタートアップ構成への新しい構成ファイルのコピー：新しい構成ファイルは **copy** コマンドを使用してファイルサーバーからデバイスにコピーされ、ファイルの有効性を確認するためにファイルチェックが実行されます。ファイルが有効な場合、そのファイルがスタートアップ構成にコピーされます。使用可能なディスク領域が十分にある場合は、以前の構成ファイルのバックアップが実行されます。デバイスが再度リロードされると、新しい構成が有効になります。

実行構成への新しい構成ファイルのコピー：新しい構成ファイルは、**copy** コマンドまたは **configure replace** コマンドを使用してファイルサーバーからデバイスにコピーされます。ロールバックが効率的に実行されると、構成ファイルの置換とロールバックによってシステムが不安定な状態のままになることがあります。したがって、ファイルをコピーして構成をアップグレードすることをお勧めします。

### [ デバイス情報 (Device Information) ]

PnP エージェントは、要求に応じてデバイスインベントリとその他の重要な情報を PnP サーバーに抽出する機能を提供します。次のデバイスプロファイル要求タイプがサポートされています。

- **all** : 固有のデバイス識別子 (UDI) 、イメージ、ハードウェア、およびファイルシステムのインベントリ データを含む完全なインベントリ情報を返します。
- **filesystem** : ファイルシステムの名前とタイプ、ローカルサイズ (バイト単位) 、空きサイズ (バイト単位) 、読み取りフラグ、書き込みフラグなど、ファイルシステムのインベントリ情報を返します。
- **hardware** : ホスト名、ベンダー文字列、プラットフォーム名、プロセッサタイプ、ハードウェアリビジョン、メインメモリサイズ、I/O メモリサイズ、ボード ID、ボードリワーク ID、プロセッサリビジョン、ミッドプレーンリビジョンおよび場所など、ハードウェアインベントリ情報を返します。
- **image** : バージョン文字列、イメージ名、ブート変数、rommon への復帰理由、ブートローダ変数、構成レジスタ、次回ブート時の構成レジスタ、および構成変数など、イメージインベントリ情報を返します。バージョン文字列、イメージ名、ブート変数、rommon への

復帰理由、ブートローダ変数、構成レジスタ、次回ブート時の構成レジスタ、および構成変数など、

- UDI : デバイス UDI を返します。

### 証明書のインストール

証明書のインストールは、PnP サーバーがデバイス上の PnP エージェントにトラストプールまたはトラスト ポイントの証明書のインストールまたはアンインストールを要求するセキュリティサービスです。このサービスは、再接続するプライマリサーバーとバックアップサーバーに関するエージェントも指定します。証明書を正常にインストールするには、次の前提条件が必要です。

- 証明書またはトラスト プールバンドルをダウンロードする必要があるサーバーに到達できる必要があります。
- 証明書またはバンドルをダウンロードするためのアクセス許可の問題はありません。
- PKI API は、エージェントが証明書またはバンドルをダウンロードしてインストールするために呼び出すことができるように、PnP エージェントが使用可能でアクセス可能である必要があります。
- ダウンロードした証明書またはバンドルを保存するのに十分なメモリがデバイスにあります。

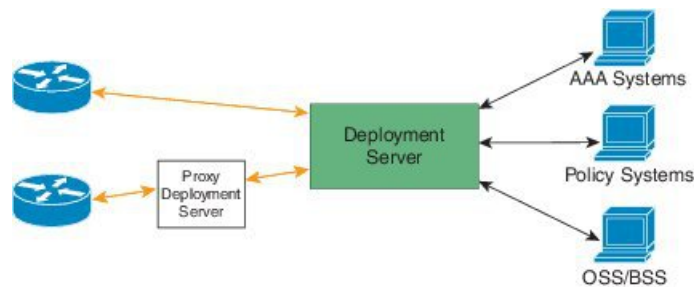
### PnP エージェント

PnP エージェントは、シスコのネットワーク デバイスのうち、簡素化された展開アーキテクチャをサポートするものすべてに含まれている組み込みソフトウェア コンポーネントです。PnP エージェントが認識し、対話する対象は PnP サーバのみです。PnP エージェントはまず、通信可能な PnP サーバの検出を試みます。サーバーが検出されて接続が確立された後で、エージェントはサーバーと通信し、構成、イメージ、ファイル更新などの展開関連のアクティビティを実行します。また、アウトオブバウンドの設定変更やインターフェイス上の新しいデバイス接続などの対象のすべての展開関連イベントをサーバーに通知します。

### PnP サーバー

PnP サーバーは、展開するデバイスの展開情報（イメージ、設定、ファイル、およびライセンス）の管理や配布のロジックを符号化する中央サーバーです。このサーバは、特定の展開プロトコルを使用することで、簡素化された展開プロセスをサポートするデバイス上のエージェントと通信します。

Figure 3: 簡素化された展開サーバ





PnP サーバは、スマートフォンと PC の展開アプリケーションなどのプロキシサーバ、Neighbor Assisted Provisioning Protocol (NAPP) として動作する他の PnP エージェント、および VPN ゲートウェイのようなその他のタイプのプロキシ展開サーバとも通信します。

PnP サーバは、エージェントを別の展開サーバにリダイレクトできます。リダイレクトの一般的な例は PnP サーバによるリダイレクトで、ブートストラップ設定を NAPP サーバを介して送信した後に直接通信するデバイスをリダイレクトします。PnP サーバは企業がホストできます。このソリューションでは、シスコが提供するクラウドベースの展開サービスが可能です。この場合、デバイスはシスコのクラウドベースの展開サービスを検出して通信し、初期導入を実行します。その後、お客様の展開サーバにそのデバイスをリダイレクトできます。

デバイスとの通信に加え、サーバは認証、承認、アカウントिंग (AAA) システム、プロビジョニングシステム、その他の管理アプリケーションなどのさまざまな外部システムとインターフェイスします。

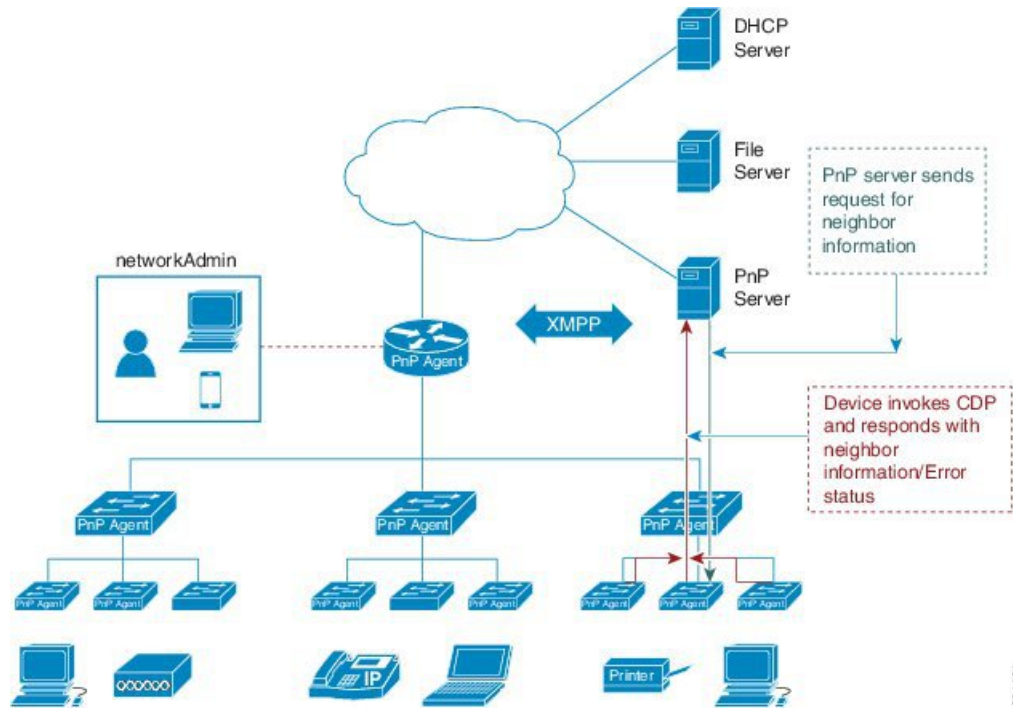
### PnP エージェントの展開

次に、シスコのデバイスでの PnP エージェントの展開手順を示します。

1. PnP エージェントを備えているシスコのデバイスは PnP サーバに問い合わせタスクを要求します。つまり、PnP エージェントは作業の要求とともに、一意のデバイス識別子 (UDI) を送信します。
2. PnP サーバにデバイス用のタスク (イメージのインストール、構成、アップグレードなど) がある場合、作業要求が送信されます。
3. PnP エージェントが作業要求を受信した後で、タスクを実行し、タスクのステータス、成功かエラーかと要求された対応する情報に関する応答を PnP サーバに返します。

### PnP エージェント ネットワーク トポロジ

Figure 4: Cisco PnP エージェントの展開のネットワーク トポロジ



### PnP エージェントの初期化

PnP エージェントはデフォルトで有効になっていますが、スタートアップ構成が利用できない場合にデバイスで開始できます。

### スタートアップ構成なし

新しいシスコのデバイスは、デバイスの NVRAM の中にスタートアップ コンフィギュレーションファイルのない状態でお客様に出荷されます。新しいデバイスがネットワークに接続され、電源が投入された時点でスタートアップ構成ファイルがデバイス上にない場合は、PnP エージェントが自動的に起動され、PnP サーバーの IP アドレスを検出します。

### PnP エージェントの CLI 構成

PnP は、デフォルトで VLAN 1 を使用しているデバイスをサポートします。

## ネットワーク プラグアンドプレイの注意事項と制限事項

ネットワーク プラグアンドプレイ (PnP) の注意事項と制限事項は次のとおりです。

- NX-OS 9.2(3) 以降、PnP は Cisco Nexus 9500 プラットフォーム スイッチの管理ポートでサポートされています。
- PnP は、インバンドインターフェイスと管理インターフェイスの両方で実行されます。インバンドは、FX シリーズ ラインカード（特に PnP の場合は N9K-X9736C-FX）でのみサポートされます。

- PnP の展開方法は、PnP コントローラまたはサーバーを見つけるために必要な検出プロセスによって異なります。
- PnP を起動する前に、DHCP サーバー検出プロセスか、またはドメイン ネーム サーバー (DNS) 検出プロセスのいずれかの検出メカニズムを展開します。
- PnP を展開する前に DHCP サーバーまたは DNS サーバーを構成します。
- PnP サーバーは PnP エージェントと通信する必要があります。
- PnP 接続では、DHCP または DNS 構成は必要ありません。
- PnP の IPv6 サポートは、Cisco Nexus 9500 シリーズ デバイスでは使用できません。

### Cisco DNA Center サポート

次の注意事項と制限事項は、Cisco DNA Center への PnP 接続に固有のものであります。

- Cisco DNA Center は、Cisco Nexus 9504、Cisco Nexus 9508、および Cisco Nexus 9516 スイッチで次の機能をサポートします。
    - ディスカバリ
    - インベントリ
    - トポロジ
    - テンプレート プログラマ
    - ソフトウェア イメージの管理
    - 基本的なモニタリング
  - 次の PnP の注意事項と制限事項は、Cisco DNA Center バージョン 1.2.6 以前のみを対象としています。
    - プラグアンドプレイ中に提供されるスタートアップ構成では、Cisco DNA Center に接続されているインターフェイスの接続が損なわれないようにする必要があります。
    - システム image.bin とスタートアップ構成を Cisco DNA Center にアップロードする必要があります。
    - ブートフラッシュには、Cisco DNA Center からイメージと設定をダウンロードするための十分なスペースが必要です。
- Cisco DNA Center のユーザー ドキュメントについては、[ここ](#)をクリックしてください。

# ネットワーク プラグアンドプレイのトラブルシューティング例

## 例: PnP のトラブルシューティング

次の例は、PnP トラブルシューティング コマンドの出力を示しています。

```
Switch# show pnp status
PnP Agent is running
server-connection
  status: Success
  time: 08:41:26 Jan 11
interface-info
  status: Success
  time: 08:34:00 Jan 11
device-info
  status: Success
  time: 08:33:46 Jan 11
config-upgrade
  status: Success
  time: 08:31:36 Jan 11
capability
  status: Success
  time: 08:33:50 Jan 11
backoff
  status: Success
  time: 08:41:26 Jan 11
topology
  status: Success
  time: 08:33:54 Jan 11
```

```
Switch# show pnp version
PnP Agent Version Summary

PnP Agent: 1.6.0
Platform Name: nxos
PnP Platform: 1.5.0.rc2
```

```
Switch# show pnp profiles
Created by          UDI
DHCP Discovery    PID:N9K-C9504,VID:V01,SN:FOX1813GCZ8

  Primary transport: https
  Address: 10.105.194.248
  Port: 443
  CA file: /etc/pnp/certs/trustpoint/pnplabel

  Work-Request Tracking:
    Pending-WR: Correlator=
Cisco-PnP-POSIX-nxos-1.6.0-21-589a466a-0d88-427b-a17e-69afb7d0a226-1
    Last-WR:      Correlator=
Cisco-PnP-POSIX-nxos-1.6.0-20-ab225de4-b0ef-46c5-9c4f-e3bd9f7c6b87-1
  PnP Response Tracking:
    Last-PR:      Correlator=
Cisco-PnP-POSIX-nxos-1.6.0-20-ab225de4-b0ef-46c5-9c4f-e3bd9f7c6b87-1
```

```
Switch# show pnp lease
```

```
{
  "lease": {
    "uptime": "Fri Jan 11 05:32:17 2019",
    "intf": "Vlan1",
    "ip_addr": "10.77.143.239",
    "mask": "255.255.255.0",
    "gw": "10.77.143.1",
    "domain": "",
    "opt_43": "5A1D;B2;K4;I10.105.194.248;J80",
    "lease": "3600",
    "server": "10.77.143.231",
    "vrf": "1"
  }
}
```

```
Switch# show pnp internal trace
```

```
1) Event:E_DEBUG, length:49, at 907122 usecs after Fri Jan 11 08:30:44 2019
   [104] pnp_ascii_gen: ascii gen completed rcode[0]
2) Event:E_DEBUG, length:16, at 907094 usecs after Fri Jan 11 08:30:44 2019
   [104] pss type: 5
3) Event:E_DEBUG, length:31, at 907069 usecs after Fri Jan 11 08:30:44 2019
   [104] Entering pnp_ascii_gen_cfg
4) Event:E_DEBUG, length:22, at 907061 usecs after Fri Jan 11 08:30:44 2019
   [104] Calling Ascii gen
5) Event:E_DEBUG, length:16, at 907051 usecs after Fri Jan 11 08:30:44 2019
   [104] pss type: 2
6) Event:E_DEBUG, length:49, at 907018 usecs after Fri Jan 11 08:30:44 2019
   [104] pnp_ascii_gen: fu_num_acfg_pss_entries[0x2]
7) Event:E_DEBUG, length:49, at 973813 usecs after Fri Jan 11 08:29:51 2019
   [104] pnp_ascii_gen: ascii gen completed rcode[0]
8) Event:E_DEBUG, length:16, at 973787 usecs after Fri Jan 11 08:29:51 2019
   [104] pss type: 5
9) Event:E_DEBUG, length:31, at 973760 usecs after Fri Jan 11 08:29:51 2019
   [104] Entering pnp_ascii_gen_cfg
10) Event:E_DEBUG, length:22, at 973751 usecs after Fri Jan 11 08:29:51 2019
   [104] Calling Ascii gen
11) Event:E_DEBUG, length:16, at 973742 usecs after Fri Jan 11 08:29:51 2019
   [104] pss type: 2
12) Event:E_DEBUG, length:49, at 973707 usecs after Fri Jan 11 08:29:51 2019
   [104] pnp_ascii_gen: fu_num_acfg_pss_entries[0x2]
13) Event:E_DEBUG, length:35, at 535794 usecs after Fri Jan 11 08:04:15 2019
   [104] pnp_pi_spawn_finalize pid 690
14) Event:E_DEBUG, length:41, at 228291 usecs after Fri Jan 11 08:04:13 2019
   [104] + pnp_pi_spawn child_pid: 0xdd526da0
15) Event:E_DEBUG, length:76, at 132853 usecs after Fri Jan 11 08:03:26 2019
   [104] Rx: Direction: PnP PI -> PnP PD, Type: Device Provisioned, Cfg: Present
16) Event:E_DEBUG, length:35, at 440380 usecs after Fri Jan 11 08:03:18 2019
```

```
[104] !!! ACKED Unconfigure Ret:1!!!  
17) Event:E_DEBUG, length:61, at 440347 usecs after Fri Jan 11 08:03:18 2019  
[104] Tx: Direction: Max, Type: DHCP Unconfigure Done, Len: 16  
18) Event:E_DEBUG, length:35, at 440331 usecs after Fri Jan 11 08:03:18 2019  
[102] Unknown timer cancel requested  
19) Event:E_DEBUG, length:35, at 440311 usecs after Fri Jan 11 08:03:18 2019  
[104] pnp_pss_runtime_commit success  
20) Event:E_DEBUG, length:57, at 440103 usecs after Fri Jan 11 08:03:18 2019  
[104] pnp_pss_runtime_commit: Stored values in runtime PSS  
21) Event:E_DEBUG, length:23, at 440051 usecs after Fri Jan 11 08:03:18 2019  
[104] - pnp_vsh_halt:206  
22) Event:E_DEBUG, length:17, at 950291 usecs after Fri Jan 11 08:03:15 2019  
[104] Adding "end"  
23) Event:E_DEBUG, length:58, at 950269 usecs after Fri Jan 11 08:03:15 2019  
[104] Adding "configure terminal ; no clock protocol none "  
24) Event:E_DEBUG, length:33, at 945994 usecs after Fri Jan 11 08:03:15 2019  
[104] - pnp_vsh_config_l3_intf:788  
25) Event:E_DEBUG, length:29, at 945979 usecs after Fri Jan 11 08:03:15 2019  
[104] + pnp_vsh_config_l3_intf  
26) Event:E_DEBUG, length:39, at 945963 usecs after Fri Jan 11 08:03:15 2019  
[104] Adding "no feature interface-vlan"  
27) Event:E_DEBUG, length:32, at 945932 usecs after Fri Jan 11 08:03:15 2019  
[104] Adding "configure terminal"  
28) Event:E_DEBUG, length:40, at 945886 usecs after Fri Jan 11 08:03:15 2019  
[104] Got Semaphore, vsh halt continue...  
29) Event:E_DEBUG, length:46, at 945870 usecs after Fri Jan 11 08:03:15 2019  
[104] sem_timedwait Success, Start VSH clean up  
30) Event:E_DEBUG, length:19, at 945843 usecs after Fri Jan 11 08:03:15 2019  
[104] + pnp_vsh_halt  
31) Event:E_DEBUG, length:35, at 945831 usecs after Fri Jan 11 08:03:15 2019  
[104] pnp_pss_runtime_commit success  
32) Event:E_DEBUG, length:57, at 945643 usecs after Fri Jan 11 08:03:15 2019  
[104] pnp_pss_runtime_commit: Stored values in runtime PSS  
33) Event:E_DEBUG, length:33, at 945607 usecs after Fri Jan 11 08:03:15 2019  
[104] !!! Received Unconfigure !!!  
34) Event:E_DEBUG, length:74, at 945578 usecs after Fri Jan 11 08:03:15 2019  
[104] Rx: Direction: PnP PI -> PnP PD, Type: DHCP Unconfigure, Cfg: Present  
35) Event:E_DEBUG, length:49, at 789616 usecs after Fri Jan 11 08:01:52 2019  
[104] pnp_ascii_gen: ascii gen completed rcode[0]  
36) Event:E_DEBUG, length:16, at 789579 usecs after Fri Jan 11 08:01:52 2019  
[104] pss type: 5  
37) Event:E_DEBUG, length:31, at 789522 usecs after Fri Jan 11 08:01:52 2019  
[104] Entering pnp_ascii_gen_cfg
```

```
38) Event:E_DEBUG, length:22, at 789514 usecs after Fri Jan 11 08:01:52 2019
[104] Calling Ascii gen

39) Event:E_DEBUG, length:16, at 789506 usecs after Fri Jan 11 08:01:52 2019
[104] pss type: 2

40) Event:E_DEBUG, length:49, at 789489 usecs after Fri Jan 11 08:01:52 2019
[104] pnp_ascii_gen: fu_num_acfg_pss_entries[0x2]

41) Event:E_DEBUG, length:35, at 789365 usecs after Fri Jan 11 08:01:52 2019
[104] pnp_pss_runtime_commit success

42) Event:E_DEBUG, length:57, at 789135 usecs after Fri Jan 11 08:01:52 2019
[104] pnp_pss_runtime_commit: Stored values in runtime PSS

43) Event:E_DEBUG, length:26, at 789096 usecs after Fri Jan 11 08:01:52 2019
[104] Phase Init -> Monitor

44) Event:E_DEBUG, length:35, at 788967 usecs after Fri Jan 11 08:01:52 2019
[104] pnp_pi_spawn_finalize pid 1c9

45) Event:E_DEBUG, length:41, at 831561 usecs after Fri Jan 11 08:01:49 2019
[104] + pnp_pi_spawn child_pid: 0xffff7e28

46) Event:E_DEBUG, length:45, at 831550 usecs after Fri Jan 11 08:01:49 2019
[104] Have startup config, Starting PnP PI....

47) Event:E_DEBUG, length:40, at 831538 usecs after Fri Jan 11 08:01:49 2019
[104] Posix log directory creation failed

48) Event:E_DEBUG, length:50, at 831479 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_fire_event: PNP_EVENT_HAVE_STARTUP_CONFIG

49) Event:E_DEBUG, length:35, at 831465 usecs after Fri Jan 11 08:01:49 2019
[104] Inside : pnp_other_msg_handler

50) Event:E_DEBUG, length:80, at 831437 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_get_data_from_queue: dequeued event 0x1102e0cc 25/cat 11 from pending Q

51) Event:E_DEBUG, length:50, at 831368 usecs after Fri Jan 11 08:01:49 2019
[104] Injecting Event PNP_EVENT_HAVE_STARTUP_CONFIG

52) Event:E_DEBUG, length:59, at 831303 usecs after Fri Jan 11 08:01:49 2019
[104] Have Startup Config, move the process state to monitor

53) Event:E_DEBUG, length:57, at 799379 usecs after Fri Jan 11 08:01:49 2019
[104] Accelerating PnP, Break Point: Break Point PoAP Init

54) Event:E_DEBUG, length:35, at 799334 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_pss_runtime_commit success

55) Event:E_DEBUG, length:57, at 799239 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_pss_runtime_commit: Stored values in runtime PSS

56) Event:E_DEBUG, length:23, at 799226 usecs after Fri Jan 11 08:01:49 2019
[104] Phase None -> Init

57) Event:E_DEBUG, length:53, at 799200 usecs after Fri Jan 11 08:01:49 2019
[104] Initalizing PnP-agent State machine curr_state 3

58) Event:E_DEBUG, length:35, at 799188 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_pss_runtime_commit success
```

```
59) Event:E_DEBUG, length:57, at 799070 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_pss_runtime_commit: Stored values in runtime PSS

60) Event:E_DEBUG, length:26, at 798965 usecs after Fri Jan 11 08:01:49 2019
[104] !!! Box is Online !!!

61) Event:E_DEBUG, length:35, at 798954 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_pss_runtime_commit success

62) Event:E_DEBUG, length:57, at 798770 usecs after Fri Jan 11 08:01:49 2019
[104] pnp_pss_runtime_commit: Stored values in runtime PSS

63) Event:E_DEBUG, length:70, at 370297 usecs after Fri Jan 11 07:55:41 2019
[102] pnp_demux_mts(463): (Warning) unexpected mts msg (opcode - 7655)

64) Event:E_DEBUG, length:41, at 092701 usecs after Fri Jan 11 07:55:33 2019
[104] PnP Init Internal subsystem, Done!!!

65) Event:E_DEBUG, length:32, at 089920 usecs after Fri Jan 11 07:55:33 2019
[104] PnP Init Internal subsystem
```

```
Switch# show pnp posix_pi configs

/isan/etc/pnp/platform_config.cfg:

/isan/etc/pnp/file_paths.cfg:

/isan/etc/pnp/pnp_config.cfg:

/isan/etc/pnp/policy_discovery.conf:

/isan/etc/pnp/certs/platform.json:

/isan/etc/pnp/certs/pnp_status.json:

/isan/etc/pnp/certs/job_status.json:
```



## 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。