Cisco Nexus 9000 シリーズ NX-OS 基本構成ガイド、リリース 10.5(x)

最終更新：2025 年 5 月 23 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/

【注意】シスコ製品をご使用になる前に、安全上の注意（ **www.cisco.com/jp/go/safety_warning/** ）をご確認ください。本書は、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。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

目 次

はじめに

この前書きは、次の項で構成されています。

対象読者

このマニュアルは、Cisco Nexus スイッチの設置、設定、および維持に携わるネットワーク管理者を対象としています。

表記法

コマンドの説明には、次のような表記法が使用されます。

表記法	説明
bold	太字の文字は、表示どおりにユーザが入力するコマンドおよびキーワードです。
italic	イタリック体の文字は、ユーザが値を指定する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角かっこで囲んで示しています。
[x | y]	いずれか1つを選択できる省略可能なキーワードや引数は、角かっこで囲み、縦棒で区切って示しています。
{x | y}	必ずいずれか1つを選択しなければならない必須キーワードや引数は、波かっこで囲み、縦棒で区切って示しています。

表記法	説明
[x {y | z}]	角かっこまたは波かっこが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。角かっこ内の波かっこと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
`variable`	ユーザが値を入力する変数であることを表します。イタリック体が使用できない場合に使用されます。
string	引用符を付けない一組の文字。string の前後には引用符を使用しないでください。引用符を使用すると、その引用符も含めて string と見なされます。

例では、次の表記法を使用しています。

表記法	説明
`screen フォント`	スイッチが表示する端末セッションおよび情報は、スクリーンフォントで示しています。
太字の screen フォント	ユーザが入力しなければならない情報は、太字の screen フォントで示しています。
イタリック体の *screen* フォント	ユーザが値を指定する引数は、イタリック体の screen フォントで示しています。
< >	パスワードのように出力されない文字は、山カッコ（< >）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

Cisco Nexus 9000 シリーズ スイッチの関連資料

Cisco Nexus 9000 シリーズ スイッチ全体のマニュアル セットは、次の URL にあります。

https://www.cisco.com/en/US/products/ps13386/tsd_products_support_series_home.html

マニュアルに関するフィードバック

このマニュアルに関する技術的なフィードバック、または誤りや記載もれなどお気づきの点がございましたら、HTML ドキュメント内のフィードバック フォームよりご連絡ください。ご協力をよろしくお願いいたします。

通信、サービス、およびその他の情報

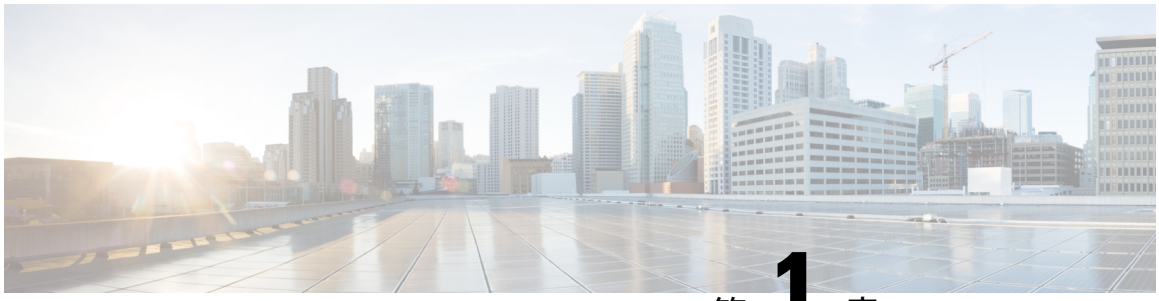
- シスコからタイムリーな関連情報を受け取るには、Cisco Profile Manager でサインアップしてください。
- 重要な技術によりビジネスに必要な影響を与えるには、Cisco Services [英語] にアクセスしてください。
- サービスリクエストを送信するには、Cisco Support [英語] にアクセスしてください。
- 安全で検証済みのエンタープライズクラスのアプリケーション、製品、ソリューション、およびサービスを探して参照するには、Cisco DevNet [英語] にアクセスしてください。
- 一般的なネットワーク、トレーニング、認定関連の出版物を入手するには、Cisco Press [英語] にアクセスしてください。
- 特定の製品または製品ファミリの保証情報を探すには、Cisco Warranty Finder にアクセスしてください。

シスコバグ検索ツール

シスコのバグ検索ツール（BST）は、シスコ製品とソフトウェアの障害と脆弱性の包括的なリストを管理するシスコバグ追跡システムへのゲートウェイです。BST は、製品とソフトウェアに関する詳細な障害情報を提供します。

マニュアルに関するフィードバック

シスコの技術マニュアルに関するフィードバックを提供するには、それぞれのオンラインドキュメントの右側のペインにあるフィードバックフォームを使用してください。



第 1 章

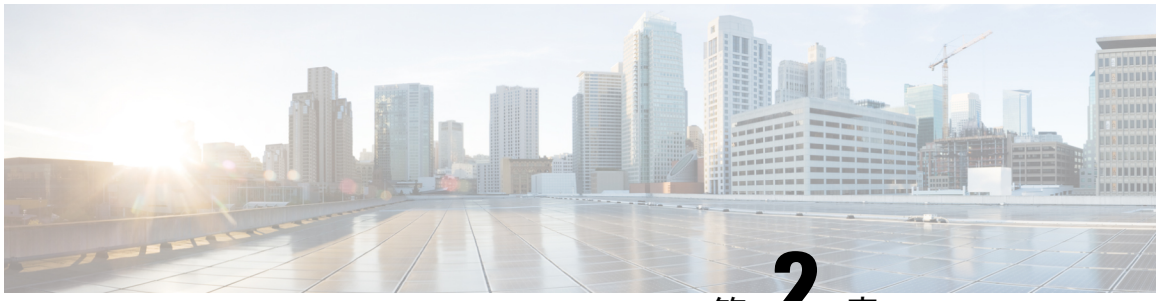
新機能および変更された機能に関する情報

新機能および変更された機能に関する情報

表 *1 :* 新機能および変更された機能

特長	説明	変更が行われたリリース	参照先
POAP - 9364E-SG2 スイッチでサポート	9364E-SG2 ToR スイッチの POAP のサポートを追加	10.5（3）F	POAP の注意事項および制約事項 （50 ページ）
N/A	このリリースで追加された新機能はありません。	10.5(1)F	N/A

新機能および変更された機能に関する情報



第 2 章

概要

この章は、次の内容で構成されています。

ライセンス要件

Cisco NX-OS ライセンス方式の推奨の詳細と、ライセンスの取得および適用の方法については、『*Cisco NX-OS* ライセンス ガイド』および『*Cisco NX-OS* ライセンス オプション ガイド』を参照してください。

サポートされるプラットフォーム

Cisco NX-OS リリース 7.0(3)I7(1) 以降では、Nexus スイッチ プラットフォーム サポート マトリクスに基づいて、選択した機能をさまざまな Cisco Nexus 9000 および 3000 スイッチで使用するために、どの Cisco NX-OS リリースが必要かを確認してください。

ソフトウェア イメージ

Cisco NX-OS ソフトウェアは、1 つの NXOS ソフトウェア イメージで構成されています。

ソフトウェアの互換性

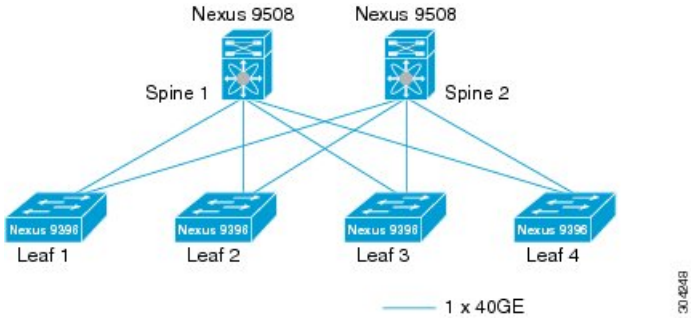
Cisco NX-OS ソフトウェアは、Cisco IOS ソフトウェアのどのバリエーションを実行するシスコ製品とも相互運用できます。Cisco NX-OS ソフトウェアは IEEE および RFC 準拠標準に準拠するどのネットワーキング オペレーティング システムとも相互運用できます。

スパイン/リーフ トポロジ

Cisco Nexus 9000 シリーズ スイッチは、2 階層のスパイン/リーフ型トポロジをサポートします。

Figure 1: スパイン/リーフ トポロジ

この図は、2 つの Spine スイッチ（Cisco Nexus 9508）に接続している 4 つの Leaf スイッチ（Cisco Nexus 9396 または 93128）、および各 Leaf から各 Spine までの 2 つの 40G イーサネット アップリンクが存在するスパイン/リーフ型トポロジの例を示しています。



モジュラ式のソフトウェア設計

Cisco NX-OS ソフトウェアは、対称型マルチプロセッサ（SMP）、マルチコア CPU、分散データ モジュール プロセッサ上の分散マルチスレッド処理をサポートします。Cisco NX-OS ソフトウェアは、ハードウェア テーブル プログラミングのような大量の演算処理を要するタスクを、データ モジュールに分散された専用のプロセッサにオフロードします。モジュール化されたプロセスは、それぞれ別の保護メモリ領域内でオンデマンドに生成されます。機能がイネーブルになったときにだけ、プロセスが開始されてシステム リソースが割り当てられます。これらのモジュール化されたプロセスはリアルタイム プリエンプティブ スケジューラによって制御されるため、重要な機能が適切なタイミングで実行されます。

サービスアビリティ

この Cisco NX-OS ソフトウェアはネットワーク傾向およびイベントへの応答を許可するサービスアビリティ機能です。これらの機能は、ネットワーク プランニングおよび応答時間の短縮に役立ちます。

スイッチド ポート アナライザ

SPAN 機能を使用すると、外部アナライザが接続された SPAN の終点ポートに、セッションに負担をかけずに SPAN セッション トラフィックが送信されるようになり、ポート（SPAN ソース ポートと呼びます）間のすべてのトラフィックを分析できるようになります。SPAN の詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

Ethanalyzer

Ethanalyzer は、Wireshark（旧称 Ethereal）オープン ソース コードに基づく Cisco NX-OS プロトコル アナライザ ツールです。Ethanalyzer は、パケットのキャプチャとデコード用の Wireshark のコマンドライン バージョンです。Ethanalyzer を使用してネットワークをトラブルシューティングし、コントロール プレーン トラフィックを分析できます。Ethanalyzer の詳細については、『*Cisco Nexus 9000 Series NX-OS* トラブルシューティング ガイド』を参照してください。

Smart Call Home

Call Home は、ハードウェア コンポーネントとソフトウェア コンポーネントを継続的にモニタリングし、重要なシステム イベントを E メールで通知する機能です。さまざまなメッセージ フォーマットが利用可能で、標準の E メールおよび XML ベースの自動解析アプリケーションとの十分な互換性があります。アラートをグループ化する機能があり、宛先プロファイルのカスタマイズも可能です。この機能の利用方法として、ネットワーク オペレーション センター（NOC）に E メール メッセージを送信する、Cisco AutoNotify サービスを利用して Cisco Technical Assistance Center（TAC）でケースを直接生成する、などがあります。Smart Call Home の詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

オンライン診断

Cisco Generic Online Diagnostics（GOLD）では、ハードウェアおよび内部データ パスが設計どおりに動作していることを確認します。Cisco GOLD には、ブート時診断、継続的監視、オンデマンドおよびスケジュールによるテストなどの機能があります。GOLD では障害を迅速に特定し、システムを継続的に監視できます。GOLD の構成の詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

組み込まれている Event Manager

Cisco Embedded Event Manager（EEM）は、ネットワーク イベントが発生した場合の動作をカスタマイズできる、デバイスおよびシステムの管理機能です。EEM の構成の詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* システム管理構成ガイド』を参照してください。

管理性

ここでは、Cisco Nexus 9000 シリーズ スイッチの管理性に関する機能について説明します。

簡易ネットワーク管理プロトコル

Cisco NX-OS ソフトウェアは、簡易ネットワーク管理プロトコル（SNMP）バージョン 1、2、および 3 に準拠しています。多くの管理情報ベース（Management Information Base）がサポートされます。SNMP の詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

構成の確認およびロールバック

Cisco NX-OSソフトウェアでは、構成をコミットする前に、構成の一貫性や必要なハードウェアリソースの可用性を確認することができます。デバイスを事前に構成し、確認した構成を後から適用することができます。構成には、必要に応じて、既知の良好な構成にロールバックできるチェック ポイントを含めることができます。ロールバックの詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

ロールベース アクセス コントロール

ロールベースアクセスコントロール（RBAC）では、ユーザにロールを割り当てることで、デバイス操作のアクセスを制限できます。アクセスが必要なユーザだけにアクセスを許可するように、カスタマイズすることが可能です。RBAC に関する詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

Cisco NX-OS デバイスのコンフィギュレーション方式

Cisco NX-OS デバイスを設定するには、次の方法を使用できます。

- セキュア シェル（SSH）セッション、Telnet セッション、またはコンソール ポートからの CLI SSH ではデバイスへの安全な接続が提供されます。CLI 構成ガイドは機能別に編成されています。詳細については、Cisco NX-OS 構成ガイドを参照してください。SSH と Telnet の詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。
- CLI を補完する NETCONF プロトコルに基づくプログラマチック方式である、XML 管理インターフェイス。詳細については、『*Cisco NX-OS XML* インターフェイス ユーザー ガイド』を参照してください。
- ローカル PC で稼動し、Cisco NDFCサーバで Web サービスを利用する、Cisco Nexus Dashboard Fabric Controller（NDFC）クライアント。Cisco NDFCサーバでは XML 管理インターフェイスを使用してデバイスを設定します。Cisco NDFCクライアントの詳細については、*Cisco NDFC Fundamentals* ガイドを参照してください。

プログラマビリティ

ここでは、Cisco Nexus 9000 シリーズ スイッチのプログラマビリティに関する機能について説明します。

Python API

Python は簡単に習得できる強力なプログラミング言語です。効率的で高水準なデータ構造を持ち、オブジェクト指向プログラミングに対してシンプルで効果的なアプローチを取っています。Python は、簡潔な構文、動的な型指定、インタープリタ型という特長を持っており、ほとんどのプラットフォームのさまざまな分野でスクリプティングと高速アプリケーション開発が可能な理想的な言語です。 Python インタープリタと広範な標準ライブラリが Python Web サイトでソース形式またはバイナリ形式で自由に利用できます。http://www.python.org/Python スクリプト機能は、さまざまなタスクを実行するために CLI と Power On Auto Provisioning（POAP）または Embedded Event Manager（EEM）アクションへのプログラムによるアクセスを提供します。Python API と Python スクリプト機能の詳細については、『*Cisco Nexus 9000 Series NX-OS Programmability Guide*』を参照してください。

Tcl

Tool Command Language（Tcl）は、スクリプト言語です。Tcl を使用すると、デバイスの CLI Commands をより柔軟に使用できます。Tcl を使用して **show** コマンドの出力の特定の値を抽出したり、スイッチを設定したり、Cisco NX-OS コマンドをループで実行したり、スクリプトで EEM ポリシーを定義したりすることができます。

Cisco NX-API

Cisco NX-API は Cisco Nexus 9000 シリーズ スイッチへの Web ベースのプログラムによるアクセスを提供します。このサポートは NX-API のオープンソースの Web サーバーによって提供されています。Cisco NX-API は Web ベースの API を介して、コマンドライン インターフェイス（CLI）の完全な設定および管理機能を公開します。XML または JSON 形式で API コールの出力を公開するようにスイッチを設定できます。Cisco NX-API の詳細については、『*Cisco Nexus 9000 Series NX-OS Programmability Guide*』を参照してください。

（注） NX-API は、スイッチ上の Programmable Authentication Module（PAM）を使用して認証を行います。cookie を使用して PAM の認証数を減らし、PAM の負荷を減らします。

bash シェル

Cisco Nexus 9000 シリーズ スイッチは、Linux シェルの直接アクセスに対応しています。Linux シェルのサポートにより、スイッチの Linux システムにアクセスして、Linux コマンドを使用

して基礎のシステムを管理できます。Bash シェルの対応の詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* プログラマビリティ ガイド』を参照してください。

Broadcom シェル

Cisco Nexus 9000 シリーズ スイッチの前面パネルおよびファブリック モジュール ライン カードには複数の Broadcom ASIC が含まれます。CLI を使用し、これらの ASIC のコマンドライン シェル（bcm シェル）にアクセスできます。この方法を使用して bcm シェルにアクセスするメリットは、**pipe include** や **redirect output to file** などの Cisco NX-OS 拡張コマンドを使用できることです。また、アクティビティは bcm シェルから直接入力するアカウンティング ログに記録されないコマンドとは異なり、監査のためにシステム アカウンティング ログに記録されます。Broadcom シェルの対応の詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* プログラマビリティ ガイド』を参照してください。

注意 Broadcom シェル コマンドは、シスコのサポート担当者の直接監督下または要求された場合のみ注意して使用してください。

トラフィックのルーティング、フォワーディング、および管理

ここでは、Cisco NX-OS ソフトウェアでサポートされるトラフィックのルーティング、転送、および管理機能について説明します。

イーサネット スイッチング

Cisco NX-OS ソフトウェアは、高密度、高パフォーマンスのイーサネット システムをサポートし、次のイーサネット スイッチング機能を提供します。

- IEEE 802.1D-2004 高速スパニングツリー プロトコル（RSTP）およびマルチ スパニングツリー プロトコル（802.1w および 802.1s）
- IEEE 802.1Q VLAN およびトランク
- IEEE 802.3ad リンク アグリゲーション
- アグレッシブ モードと標準モードの Unidirectional Link Detection（UDLD； 単一方向リンク検出）

詳細については、『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』および『*Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide*』を参照してください。

IP ルーティング

Cisco NX-OS ソフトウェアは、IP Version 4（IPv4）および IP Version 6（IPv6）、および次のルーティング プロトコルをサポートしています。

- Open Shortest Path First（OSPF）プロトコル バージョン 2（IPv4）および 3（IPv6）
- Intermediate System-to-Intermediate System（IS-IS）プロトコル（IPv4 および IPv6）
- ボーダー ゲートウェイ プロトコル（BGP）（IPv4 および IPv6）
- 拡張内部ゲートウェイ プロトコル（EIGRP）（IPv4 のみ）
- Routing Information Protocol Version 2（RIPv2）（IPv4 のみ）

Cisco NX-OS ソフトウェアでのこれらのプロトコルの実装は、最新の規格に完全に準拠しています。また、4 バイト自律システム番号（ASN）とインクリメンタル Shortest Path First（SPF）が含まれています。すべてのユニキャスト プロトコルでは、ノンストップ フォワーディング グレースフル リスタート（NSF-GR）をサポートしています。すべてのプロトコルは、イーサネット インターフェイス、VLAN インターフェイス、サブインターフェイス、ポート チャネル、およびループバック インターフェイスなど、すべてのインターフェイス タイプをサポートしています。

詳細については、『*Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide*』を参照してください。

IP サービス

Cisco NX-OS ソフトウェアでは、次の IP サービスを使用できます。

- Virtual Routing and Forwarding（VRF）
- Dynamic Host Configuration Protocol（DHCP）ヘルパー
- ホットスタンバイ ルータ プロトコル（HSRP）
- 拡張オブジェクト トラッキング
- ポリシーベース ルーティング（PBR）
- IPv4 の全プロトコルに対するユニキャスト グレースフル リスタート、および IPv6 の OSPFv3 に対するユニキャスト グレースフル リスタート

詳細については、『*Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide*』を参照してください。

IP Multicast：IP マルチキャスト

Cisco NX-OS は、次のマルチキャスト プロトコルおよび機能を備えています。

- プロトコル独立型マルチキャスト（PIM）バージョン 2（PIMv2）

- PIM スパース モード（IPv4 の Any-Source マルチキャスト（ASM））
- エニーキャスト ランデブー ポイント（Anycast-RP）
- IPv4 のマルチキャスト NSF
- ブートストラップ ルータ（BSR）を使用した RP 検出（Auto-RP およびスタティック）
- インターネット グループ管理プロトコル（IGMP）バージョン 1、2、3 ルータ ロール
- IGMPv2 ホスト モード
- IGMP スヌーピング
- マルチキャスト ソース検出プロトコル（MSDP）（IPv4 専用）

Note Cisco NX-OS ソフトウェアは、PIM デンス モードをサポートしていません。

詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* マルチキャスト ルーティング構成ガイド』を参照してください。

QoS

Cisco NX-OS ソフトウェアでは、分類、マーキング、キューイング、ポリシング、およびスケジューリングに対する Quality of Service（QoS）機能をサポートしています。Modular QoS CLI（MQC）では、すべての QoS 機能をサポートしています。MQC を使用すると、シスコのさまざまなプラットフォームで同一の構成を行うことができます。詳細については、『*Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide*』を参照してください。

ネットワーク セキュリティ機能

Cisco NX-OS ソフトウェアには、次のセキュリティ機能があります。

- コントロール プレーン ポリシング（CoPP）
- メッセージ ダイジェスト アルゴリズム 5（MD5）のルーティング プロトコル認証
- 認証、認可、アカウンティング（AAA）
- RADIUS および TACACS+
- SSH プロトコル バージョン 2
- SNMPv3
- 名前付き ACL（ポートベース ACL（PACL）、VLAN ベース ACL（VACL）、およびルータベース ACL（RACL））によってサポートされる、MAC アドレスおよび IPv4 アドレスに基づいたポリシー

- トラフィック ストーム制御（ユニキャスト、マルチキャスト、およびブロードキャスト）

詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

Supported Standards

次の表に、IEEE 準拠標準を示します。

***Table 2: IEEE* 準拠標準**

標準	説明
802.1D	MAC ブリッジ
802.1p	イーサネット フレームのサービス クラス（CoS）タギング
802.1Q	VLAN タギング
802.1s	マルチ スパニング ツリー プロトコル
802.1w	高速スパニングツリー プロトコル
802.3ab	1000Base-T（10/100/1000 Ethernet over copper）
802.3ad	LACP によるリンク集約
802.3ae	10 ギガビット イーサネット

次の表に、RFC 準拠標準を示します。各 RFC の詳細については、www.ietf.org を参照してください。

***Table 3: RFC* 準拠標準**

標準	説明
BGP	
RFC 1997	*BGP* コミュニティの属性
RFC 2385	*TCP MD5* シグネチャ オプションを使用した *BGP* セッションの保護
RFC 2439	*BGP* ルート フラップ ダンピング

標準	説明
RFC 2519	ドメイン ルート間集約のフレームワーク
RFC 2545	『*Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*』
RFC 2858	『*Multiprotocol Extensions for BGP-4*』
RFC 2918	『*Route Refresh Capability for BGP-4*』
RFC 3065	*BGP* の自律システム連合
RFC 3392	*BGP-4* による機能のアドバタイズメント
RFC 4271	*BGP* バージョン *4*
RFC 4273	*BGP4 MIB - BGP-4* の管理対象オブジェクトの定義
RFC 4456	*BGP* ルート リフレクション: フルメッシュ内部 *BGP (IBGP)* の代替
RFC 4486	*BGP Cease* 通知メッセージのサブコード
RFC 4724	*BGP* のグレースフル リスタート メカニズム
RFC 4893	*4* オクテット *AS* 番号スペースの *BGP* サポート
RFC 5004	*1* つの外部から別の外部への *BGP* 最良パス移行の回避

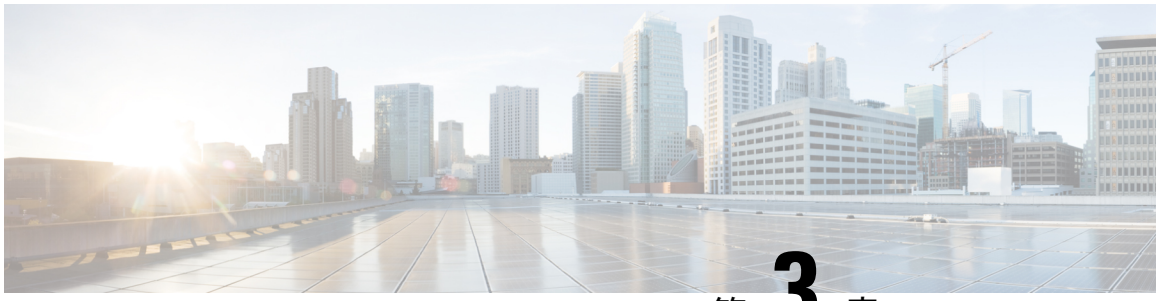
標準	説明
RFC 5396	『*Textual Representation of Autonomous System (AS) Numbers*』 **Note** RFC 5396 は部分的にサポートされます。asplain と asdot 表記はサポートされますが、asdot+ 表記はサポートされません。
RFC 5549	*IPv6* ネクスト ホップを使用した *IPv4* ネットワーク レイヤ到達可能性情報のアドバタイズ
RFC 5668	*4-Octet AS* 指定 *BGP* 拡張コミュニティ
IETF ドラフト	最適パス遷移回避（draft-ietf-idr-avoid-transition-05.txt）
IETF ドラフト	ピア テーブル オブジェクト（draft-ietf-idr-bgp4-mib-15.txt）
IETF ドラフト	動的ケイパビリティ（draft-ietf-idr-dynamic-cap-03.txt）
IP マルチキャスト	
RFC 2236	『*Internet Group Management Protocol, Version 2*』
RFC 3376	インターネット グループ管理プロトコル、バージョン *3*
RFC 3446	*Anycast Rendezvous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)*
RFC 3569	送信元特定マルチキャスト（*SSM*）の概要
RFC 3618	マルチキャスト検出プロトコル（*MSDP*）

標準	説明
RFC 4601	プロトコル独立マルチキャスト - スパース モード（*PIM-SM*）：プロトコル仕様（改訂）
RFC 4607	*IP* のためのソース仕様マルチキャスト
RFC 4610	『*Anycast-RP Using Protocol Independent Multicast (PIM)*』
RFC 6187	セキュアシェル認証用の *X.509v3*証明書
RFC 9465	*PIM* ヌルレジスタパッキング
IETF ドラフト	Mtrace server functionality, to process mtrace-requests, draft-ietf-idmr-traceroute-ipm-07.txt
IP サービス	
RFC 768	*UDP*
RFC 783	*TFTP*
RFC 791	*IP*
RFC 792	*ICMP*
RFC 793	*[TCP]*
RFC 826	『*ARP*』
RFC 854	*Telnet*
RFC 959	*FTP*
RFC 1027	『*Proxy ARP*』
RFC 8573	*NTP* セキュリティは、*AES128CMAC* 認証メカニズムによって強化されています。
RFC 7822	*NTP v4*
RFC 1305	*NTP v3*
RFC 1519	*CIDR*
RFC 1542	*BootP* リレー

標準	説明
RFC 1591	*DNS* クライアント
RFC 1812	『*IPv4 routers*』
RFC 2131	*DHCP* ヘルパー
RFC 2338	*VRRP*
『IS-IS』	
RFC 1142（OSI 10589）	『*OSI 10589 Intermediate system to intermediate system intra-domain routing exchange protocol*』
RFC 1195	『*Use of OSI IS-IS for routng in TCP/IP and dual environment*』
RFC 2763	*IS-IS* のための動的ホスト名交換メカニズム
RFC 2966	『*Domain-wide Prefix Distribution with Two-Level IS-IS*』
RFC 2973	『*IS-IS Mesh Groups*』
RFC 3277	*IS-IS* 過渡的ブラックホール回避
RFC 3373	『*Three-Way Handshake for IS-IS Point-to-Point Adjacencies*』
RFC 3567	『*IS-IS Cryptographic Authentication*』
RFC 3847	『*Restart Signaling for IS-IS*』
IETF ドラフト	インターネット ドラフト：リンクステート ルーティング プロトコルにおけるLAN経由ポイントツーポイント オペレーション（draft-ietf-isis-igp-p2p-over-lan-06.txt）
OSPF	
RFC 2328	『*OSPF Version 2*』

標準	説明
RFC 2370	*OSPF Opaque LSA* オプション
RFC 2740	*OSPF for IPv6*（*OSPF* バージョン *3*）
RFC 3101	*OSPF Not-So-Stubby-Area*（*NSSA*）オプション
RFC 3137	『*OSPF Stub Router Advertisement*』
RFC 3509	『*Alternative Implementations of OSPF Area Border Routers*』
RFC 3623	グレースフル *OSPF* リスタート
RFC 4750	*OSPF* バージョン *2 MIB*
Per-Hop Behavior（PHB）	
RFC 2597	アシュアード転送 *PHB* グループ
RFC 3246	完全優先転送 *PHB*
RIP	
RFC 1724	*RIPv2 MIB* 拡張
RFC 2082	*RIPv2 MD5* 認証
RFC 2453	*RIP* バージョン *2*
SNMP	
RFC 2579	『*Textual Conventions for SMIv2*』
RFC 2819	『*Remote Network Monitoring Management Information Base*』
RFC 2863	『*The Interfaces Group MIB*』
RFC 3164	『*The BSD syslog Protocol*』

標準	説明
RFC 3176	*InMon Corporation*の*sFlow*：スイッチドおよびルーテッドネットワークのトラフィックをモニタする方法
RFC 3411 および RFC 3418	『*An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*』
RFC 3413	『*Simple Network Management Protocol (SNMP) Applications*』
RFC 3417	『*Transport Mappings for the Simple Network Management Protocol (SNMP)*』
プログラマビリティ	
RFC 8040	*RESTCONF* プロトコル



第 3 章

Cisco NX-OS セットアップ ユーティリティの使用

この章は、次の内容で構成されています。

Cisco NX-OS セットアップ ユーティリティについて

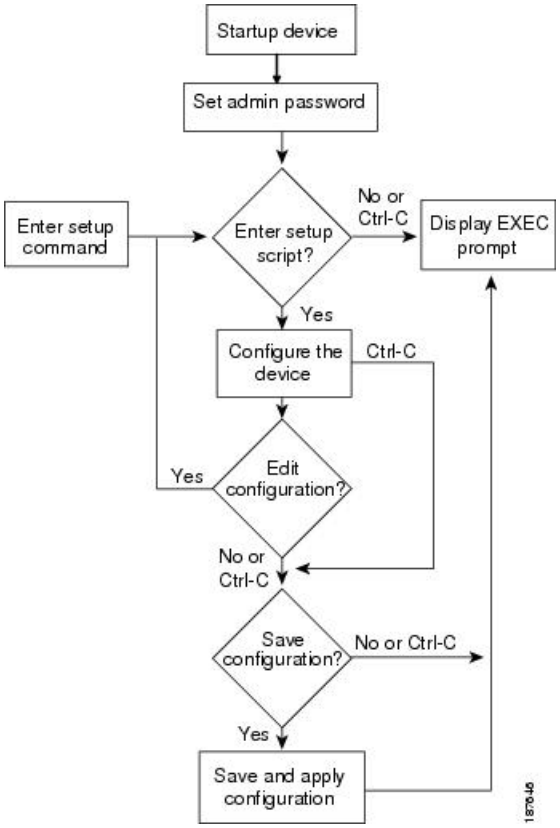
Cisco NX-OS セットアップ ユーティリティは、システムの基本（スタートアップとも呼びます）設定をガイドする対話型のコマンドライン インターフェイス（CLI）モードです。セットアップ ユーティリティでは、システム管理に使用する接続だけを設定できます。

セットアップユーティリティでは、システム構成ダイアログを使用して初期構成ファイルを作成できます。セットアップは、デバイスの構成ファイルがNVRAMにない場合に自動的に開始されます。ダイアログを使って初期構成の操作が順を追って説明されます。ファイルが作成された後、CLI を使用して追加の設定を行うことができます。

任意のプロンプトに対して **Ctrl キーを押した状態で C キーを押して（Ctrl-C）**、残りの構成オプションをスキップし、その時点までに構成された内容で先に進むことができます。ただし、管理者パスワードはスキップできません。質問に対する回答をスキップする場合は、**Enter** キーを押します。デフォルトの回答が見つからない場合（たとえば、デバイス ホスト名）、デバイスでは以前の構成を使用して、次の質問にスキップします。

Figure 2: セットアップ スクリプトのフロー

次の図に、セットアップ スクリプトを入力および終了する方法を示します。



セットアップユーティリティは、構成がない場合にシステムを始めて構成するときに主に使用します。ただし、セットアップユーティリティは基本的なデバイス設定のためにいつでも使用できます。スクリプト内でステップをスキップすると、セットアップユーティリティによって構成値が維持されます。たとえば、すでに mgmt0 インターフェイスを構成している場合、この手順をスキップしても、セットアップ ユーティリティではその構成を変更しません。ただし、手順のデフォルト値がある場合は、セットアップユーティリティによって構成値ではなくデフォルトを使用して構成が変更されます。構成を保存する前に、よく構成の変更内容を確認してください。

Note SNMP アクセスを有効にする場合は、必ず IPv4 ルート、デフォルト ネットワーク IPv4 アドレス、およびデフォルト ゲートウェイ IPv4 アドレスを構成してください。IPv4 ルーティングを有効にすると、デバイスは IPv4 ルートとデフォルト ネットワーク IPv4 アドレスを使用します。IPv4 ルーティングが無効の場合、デバイスはデフォルト ゲートウェイ IPv4 アドレスを使用します。

Note セットアップ スクリプトでは IPv4 だけをサポートしています。

セットアップ ユーティリティの前提条件

セットアップ ユーティリティには次の前提条件があります。

- ネットワーク環境のパスワード戦略が決まっていること。
- スーパーバイザ モジュールのコンソール ポートがネットワークに接続されていること。デュアル スーパーバイザ モジュールの場合、両方のスーパーバイザ モジュールのコンソール ポートがネットワークに接続されていること。
- スーパーバイザモジュールのイーサネット管理ポートがネットワークに接続されていること。デュアル スーパーバイザ モジュールの場合は、両方のスーパーバイザ モジュールのイーサネット管理ポートがネットワークに接続されていること。

Cisco NX-OS デバイスのセットアップ

セットアップ ユーティリティを使用して Cisco NX-OS デバイスの基本管理を構成するには、次の手順を実行します。

Procedure

ステップ 1 デバイスの電源を入れます。

ステップ 2 パスワードの強度確認を有効または無効にします。

強力なパスワードは、次の特性を持ちます。

- 長さが 8 文字以上である
- 複数の連続する文字（「abcd」など）を含んでいない
- 複数の同じ文字の繰り返し（「aaabbb」など）を含んでいない
- 辞書に載っている単語を含んでいない
- 正しい名前を含んでいない
- 大文字および小文字の両方が含まれている
- 数字が含まれている

Example:

```
        ---- System Admin Account Setup ----

Do you want to enforce secure password standard (yes/no) [y]: y
```

ステップ 3 管理者の新しいパスワードを入力します。

Note
パスワードが脆弱な場合は（短い、解読されやすいなど）、そのパスワードの構成が拒否されます。パスワードは大文字と小文字が区別されます。少なくとも 8 文字以上、大文字と小文字の両方と数字を使用した強力なパスワードを構成してください。

Example:

```
Enter the password for "admin": <password>

Confirm the password for "admin": <password>

---- Basic System Configuration Dialog ----

This setup utility will guide you through the basic configuration of
the system. Setup configures only enough connectivity for management
of the system.

Please register Cisco Nexus 9000 Family devices promptly with your
supplier. Failure to register may affect response times for initial
service calls. Nexus devices must be registered to receive
entitled support services.

Press Enter at anytime to skip a dialog. Use ctrl-c at anytime
to skip the remaining dialogs.
```

ステップ 4 **yes** と入力して、セットアップ モードを開始します。

Example:

```
Would you like to enter the basic configuration dialog (yes/no): yes
```

ステップ 5 **yes** と入力して（デフォルトは **no**）、追加のアカウントを作成します。

Example:

```
  Create another login account (yes/no) [n]:yes
```

a) ユーザ ログイン ID を入力します。

Example:

```
Enter the User login Id : user_login
```

Caution
ユーザ名の先頭は英数字とする必要があります。ユーザ名には特殊文字 (+ = . _ \ -)。# 記号と！記号はサポートされていません。ユーザ名に許可されていない文字が含まれている場合、指定したユーザはログインできません。

b) ユーザ パスワードを入力します。

Example:

```
Enter the password for "user1": user_password
Confirm the password for "user1": user_password
```

c) デフォルトのユーザー ロールを入力します。

Example:

```
Enter the user role (network-operator|network-admin) [network-operator]: default_user_role
```

デフォルトのユーザー ロールの詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

ステップ 6 **yes** と入力して、SNMP コミュニティ ストリングを設定します。

Example:

```
Configure read-only SNMP community string (yes/no) [n]: yes
SNMP community string : snmp_community_string
```

SNMP の詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

ステップ 7 デバイス名を入力します（デフォルト名は switch です）。

Example:

```
Enter the switch name: switch_name
```

ステップ 8 **yes** と入力して、アウトオブバンド管理を構成します。 mgmt0 IPv4 アドレスとサブネット マスクを入力できます。

Note
セットアップ ユーティリティで構成できるのは、IPv4 アドレスだけです。IPv6 の構成の詳細については、『*Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide*』を参照してください。

Example:

```
Continue with Out-of-band (mgmt0) management configuration? [yes/no]: yes
Mgmt0 IPv4 address: mgmt0_ip_address
Mgmt0 IPv4 netmask: mgmt0_subnet_mask
```

ステップ 9 **yes** と入力して IPv4 デフォルト ゲートウェイ（推奨）を構成します。これで、IP アドレスを入力できます。

Example:

```
Configure the default-gateway: (yes/no) [y]: yes
IPv4 address of the default-gateway: default_gateway
```

ステップ 10 **yes** と入力して、スタティック ルート、デフォルト ネットワーク、DNS、およびドメイン名などの高度な IP オプションを構成します。

Example:

```
Configure Advanced IP options (yes/no)? [n]: yes
```

ステップ 11 **yes** と入力して、スタティック ルート（推奨）を構成します。宛先プレフィックス、宛先プレフィックス マスク、およびネクスト ホップの IP アドレスを入力できます。

Example:

```
Configure static route: (yes/no) [y]: yes
Destination prefix: dest_prefix
Destination prefix mask: dest_mask
Next hop ip address: next_hop_address
```

ステップ 12 **yes** と入力して、デフォルト ネットワーク（推奨）を構成します。次に、IPv4 アドレスを入力できます。

Note
デフォルト ネットワークの IPv4 アドレスは、スタティック ルート構成の宛先プレフィックスと同じです。

Example:

```
Configure the default network: (yes/no) [y]: yes
Default network IP address [dest_prefix]: dest_prefix
```

ステップ 13 **yes** と入力して、DNS の IPv4 アドレスを構成します。アドレスを入力できます。

Example:

```
Configure the DNS IP address? (yes/no) [y]: yes
DNS IP address: ipv4_address
```

ステップ 14 **yes** と入力して、デフォルトのドメイン名を構成します。次に、名前を入力します。

Example:

```
Configure the DNS IP address? (yes/no) [y]: yes
DNS IP address: ipv4_address
```

ステップ 15 **yes** と入力して、Telnet サービスを有効にします。

Example:

```
Enable the telnet service? (yes/no) [y]: yes
```

ステップ 16 **yes** と入力して、SSH サービスを有効にします。続いて、キー タイプとキー ビット数を入力します。詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

Example:

```
Enable the ssh service? (yes/no) [y]: yes
```

```
Type of ssh key you would like to generate (dsa/rsa) : key_type
Number of  key bits <768-2048> : number_of_bits
```

ステップ 17 **yes** と入力して、NTP サーバーを構成します。これで、IP アドレスを入力できます。詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

Example:

```
Configure NTP server? (yes/no) [n]: yes
NTP server IP address: ntp_server_IP_address
```

ステップ 18 デフォルトのインターフェイス レイヤ（L2 または L3）を指定します。

Example:

```
Configure default interface layer (L3/L2) [L3]: interface_layer
```

ステップ 19 デフォルトのスイッチポート インターフェイス ステート（シャットダウンまたはシャットダウンなし）を入力します。シャットダウン インターフェイスは、管理上ダウン状態になります。詳細については、*Cisco Nexus 9000* シリーズ *NX-OS* インターフェイス設定ガイドを参照してください。

Example:

```
Configure default switchport interface state (shut/noshut) [shut]: default_state
```

ステップ 20 yes と入力して（デフォルトは no）、基本的なファイバ チャネル構成を行います。

Example:

```
Enter basic FC configurations (yes/no) [n]: yes
```

Note
この手順は、 SAN スイッチングをサポートするプラットフォームでのみ使用できます。

a) shut と入力して（デフォルトは noshut）、デフォルトのファイバ チャネル スイッチ ポート インターフェイスを shut（無効）状態に構成します。

Example:

```
Configure default physical FC switchport interface state (shut/noshut) [noshut]: shut
```

b) on と入力して（デフォルトは on）、スイッチ ポート トランク モードを構成します。

Example:

```
Configure default physical FC switchport trunk mode (on/off/auto) [on]: on
```

c) permit と入力して（デフォルトは deny）、デフォルトのゾーン ポリシー構成を許可します。

Example:

```
Configure default zone policy (permit/deny) [deny]: permit
```

デフォルト ゾーンのすべてのメンバへのトラフィック フローを許可します。

Example:

Note
write erase コマンドを入力した後でセットアップ スクリプトを実行する場合、スクリプト終了後、次のコマンドを使用してデフォルトのゾーン ポリシーを明示的に変更し、VSAN 1 を許可する必要があります。

```
switch(config)# zone default-zone permit vsan 1
```

d) yes と入力して（デフォルトは no）、フル ゾーン セット配信をイネーブルにします。

Example:

```
Enable full zoneset distribution (yes/no) [n]: yes
```

ステップ 21 コントロール プレーン ポリシング（CoPP）のベスト プラクティスのプロファイルを入力します。詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

Example:

```
Configure best practices CoPP profile (strict/moderate/lenient/none) [strict]: moderate
```

ここでシステムに、全設定の概要を示し、これを編集するかどうかの確認を求められます。

ステップ 22 **no** と入力して次の手順に進みます。**yes** と入力すると、セットアップ ユーティリティは設定の先頭に戻り、各ステップを繰り返します。

Example:

```
Would you like to edit the configuration? (yes/no) [y]: yes
```

ステップ 23 **yes** と入力して、この構成を使用および保存します。ここで設定を保存しておかないと、次回のデバイス起動時に設定が更新されません。構成を保存する場合は、**yes** と入力します。この手順は、nx-os イメージのブート変数も自動的に構成されることを確実にします。

Example:

```
Use this configuration and save it? (yes/no) [y]: yes
```

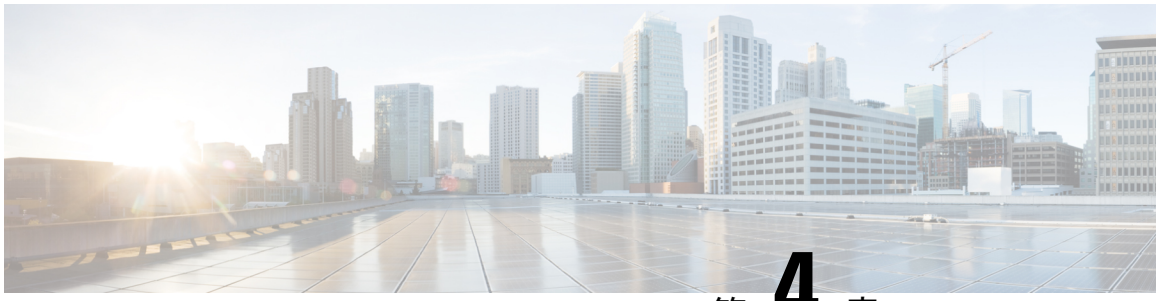
Caution
ここで構成を保存しておかないと、次回のデバイス起動時に設定が更新されません。**yes** と入力して新しい構成を保存し、nx-os イメージのブート変数も自動的に構成されるようにします。

セットアップ ユーティリティに関する追加情報

ここでは、セットアップ ユーティリティの使用に関するその他の情報について説明します。

セットアップ ユーティリティの関連資料

関連項目	マニュアル タイトル
ライセンス	『*Cisco NX-OS Licensing Guide*』
SSH および Telnet	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
ユーザ ロール	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
IPv4 および IPv6	『*Cisco Nexus 9000* シリーズ *NX-OS* ユニキャスト ルーティング設定ガイド』
SNMP および NTP	『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』



第 4 章

PowerOn Auto Provisioning の使用方法

この章は、次の項で構成されています。

PowerOn Auto Provisioning について

PowerOn Auto Provisioning（POAP）は、ネットワークに初めて導入された Cisco Nexus スイッチに対して、ソフトウェアイメージのアップグレードと構成ファイルのインストールのプロセスを自動化します。

POAP 機能を備えたデバイスが起動し、スタートアップ設定が見つからない場合、デバイスは POAP モードに入り、DHCP サーバーを検索、インターフェイス IP アドレス、ゲートウェイ、および DNS サーバーの IP アドレスを使用して自身をブートストラップします。また、TFTP サーバーの IP アドレスを取得し、ダウンロードするためのスイッチを有効化し、適切なソフトウェアイメージと構成ファイルをダウンロードしてインストールする構成スクリプトをダウンロードします。

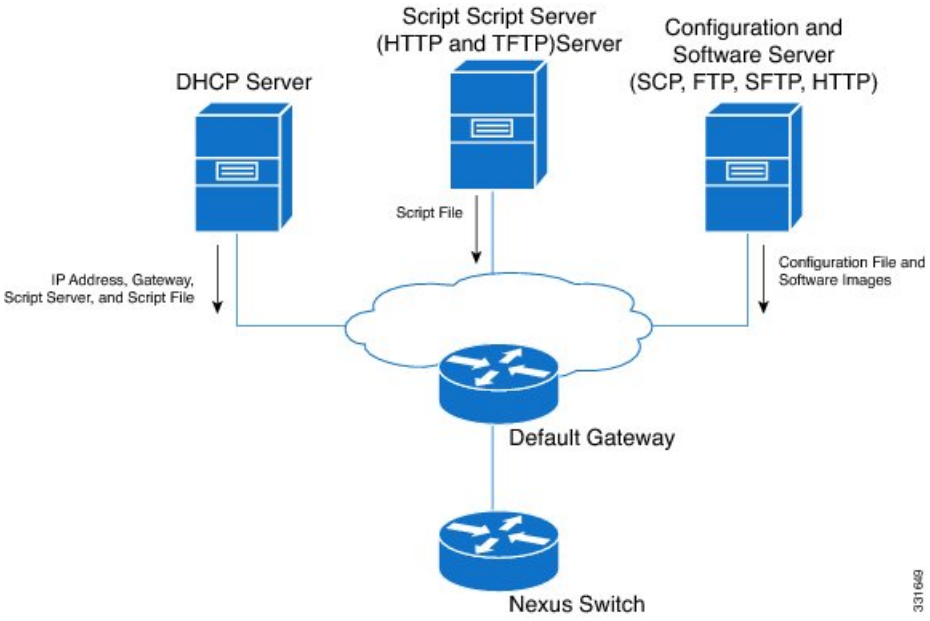
（注） DHCP 情報は、POAP 処理中にだけ使用されます。

POAP のためのネットワーク要件

POAP には、次のネットワーク インフラが必要です。

- インターフェイス IP アドレス、ゲートウェイ アドレス、およびドメイン ネーム システム（DNS）サーバーをブートストラップする DHCP サーバー。
- ソフトウェア イメージのインストールと構成のプロセスを自動化する構成スクリプトが保管されている TFTP または HTTP サーバー。
- 必要なソフトウェア イメージと構成ファイルが保管されている 1 台以上のサーバー。
- USB を使用する場合、POAP に DHCP サーバーまたは TFTP サーバーは必要ありません。

図 *3: POAP* ネットワーク インフラ



POAP スクリプトの安全なダウンロード

Cisco NX-OS リリース 10.2(3)F 以降、POAP スクリプトを安全にダウンロードするオプションがあります。POAP 機能を備えたデバイスが起動し、スタートアップ設定が見つからない場合、デバイスは POAP モードに入り、DHCP サーバーを検索、インターフェイス IP アドレス、ゲートウェイ、および DNS サーバーの IP アドレスを使用して自身をブートストラップします。また、デバイスは HTTPS サーバーの IP アドレスを取得し、POAP スクリプトを安全にダウンロードします。このスクリプトにより、スイッチは適切なソフトウェア イメージと構成ファイルをダウンロードしてインストールできます。

POAP スクリプトを安全にダウンロードするには、特定の POAP オプションを選択する必要があります。Cisco NX-OS リリース 10.2(3)F までは、POAP は IPv4 の場合はオプション 66 と 67、IPv6 の場合はオプション 77 と 15 を使用して、ブート スクリプト情報を抽出していました。ただし、スクリプトの転送は http を使用するため、あまり安全ではありません。Cisco NX-OS リリース 10.2(3)F 以降、オプション 43 は IPv4 のセキュア POAP 関連のプロビジョニング スクリプト情報を指定し、オプション 17 は IPv6 の同じことを指定します。さらに、これらのオプションにより、POAP は安全な方法でファイル サーバーに到達できます。POAP オプション

66、67、77、および 15 は、Cisco NX-OS Release10.2(3)F で引き続きサポートされます。さらに、オプション43または17を使用している場合は、必要に応じて、以前のオプションをフォールバック オプションとして使用できます。Cisco NX-OS リリース 10.4(1)F 以降では、セキュア POAP 用の単一の .pem 証明書の代わりにルート CA バンドルを使用できます。

（注） オプション 43 とオプション 17 の両方の最大文字長は 512 バイトです。

オプション 43 およびオプション 17 で使用できるサブオプションについては、次のセクションで説明します。

- オプション 43 - IPv4 IPv4 （31 ページ）
- オプション 17 - IPv6 IPv6 （32 ページ）

IPv4

オプション 43 には、IPv4 の次のサブオプションがあります。

- option space poap length width 2;
- option poap.version code 1 = unsigned integer 8;

（注） このサブオプションは必須です。

- option poap.ca_list code 50 = text;
- option poap.url code 2 = text;

（注） このサブオプションは必須です。

- option poap.version code 1 = unsigned integer 8;
- option poap.ntp code 3 = ip-address;

（注） このサブオプションは、IPv4（オプション43）でのみサポートされます。

- option poap.version code 1 = unsigned integer 8;

（注） フラグは、クライアントでのサーバー証明書の検証をスキップするために使用されます。

IPv4 の構成例は次のとおりです。

```
host dhclient-n9kv {
hardware ethernet 00:50:56:85:c5:30;
fixed-address 3.3.3.1;
default-lease-time 3600;
option broadcast-address 192.168.1.255;
#option log-servers 1.1.1.1;
max-lease-time 3600;
option subnet-mask 255.255.255.0;
option routers 10.77.143.1;
#option domain-name-servers 1.1.1.1;
          vendor-option-space poap;
option poap.version 1;
option poap.ca_list "https://<ip>/poap/ca_file1.pem, https://<ip>/poap/ca_file2.pem";
option poap.url "https://<url>/poap.py";
option poap.debug 1;
option poap.ntp 10.1.1.39;
option poap.flag 0;
  }
```

IPv6

オプション 17 には、IPv6 の次のサブオプションがあります。

- option space poap_v6 length width 2;
- option poap_v6.version code 1 = unsigned integer 8;

（注） このサブオプションは必須です。

- option poap_v6.ca_list code 50 = text;
- option poap.url code 2 = text;

（注） このサブオプションは必須です。

- option poap_v6.debug code 51 = unsigned integer 8;
- option vsio.poap_v6 code 9 = encapsulate poap_v6;

IPv6 の設定例は次のとおりです。

```
 option dhcp6.next-hop-rt-prefix code 242 = { ip6-address, unsigned integer 16,
unsigned integer 16, unsigned integer 32, unsigned integer 8, unsigned integer 8,
ip6-address };
 option dhcp6.bootfile-url code 59 = string;

 default-lease-time 3600;
 max-lease-time 3600;
 log-facility local7;
 subnet6 2003::/64 {
```

```
  # This statement configures actual values to be sent
# RTPREFIX option code = 243, RTPREFIX length = 22
# Ignore value 22. It is something related to option-size RT_PREFIX option length.
# lifetime = 9000 seconds
# route ETH1_IPV6_GW/64
# metric 1
option dhcp6.next-hop-rt-prefix 2003::2222 243 22 9000 0 1 ::;
#ipv6 ::/0 2003::2222
#Another example - support not there in NXOS - CSCvs05271:
#option dhcp6.next-hop-rt-prefix 2003::2222 243 22 9000 112 1 2003::1:2:3:4:5:0;
#ipv6 2003::1:2:3:4:5:0/112 2003::2222

  # Additional options
#option dhcp6.name-servers fec0:0:0:1::1;
#option dhcp6.domain-search "domain.example";


range6 2003::b:1111 2003::b:9999;
option dhcp6.bootfile-url "tftp://2003::1111/poap_github_v6.py";
vendor-option-space poap_v6;
option poap_v6.version 1;
option poap_v6.ca_list "https://<ip>/new_ca.pem,https://<ip>/another_ca.pem";
option poap_v6.url "https://<ip>/poap_github_v4.py";
option poap_v6.debug 1;
  }
```

安全な POAP のネットワーク要件

POAP には、次のネットワーク インフラが必要です。

- インターフェイス IP アドレス、ゲートウェイ アドレス、およびドメイン ネーム システム（DNS）サーバーをブートストラップする DHCP サーバー。
- ソフトウェア イメージのインストールと構成のプロセスに使用される POAP スクリプトを含む HTTP サーバー。

（注）
- POAP スクリプトの安全なダウンロードの場合、TFTP サーバーは HTTPS サーバーに置き換えられます。したがって、この章の TFTP サーバーに関連する内容を読むときは、TFTP サーバーを HTTPS サーバーとして読むことを忘れないでください。

- 必要なソフトウェア イメージと構成ファイルが保管されている 1 台以上のサーバー。

導入シナリオ

Cisco デバイスには、Secure Unique Device Identifier (SUDI) と呼ばれる一意の識別子があります。ハードウェア SUDI は、暗号化、暗号解読、署名、操作対象のデータの通過を許可する検証などの非対称キー操作に使用できます。シスコ以外のすべてのデバイスは、非 SUDI デバイスとして分類されます。非SUDIデバイスの場合、ファイル サーバーを認証するためにルート

CA バンドルが必要です。ただし、ファイル サーバーは、SUDI または非 SUDI デバイスのいずれかでホストできます。

これらすべての機能に基づいて、次の展開シナリオのいずれかを使用して、POAP スクリプトを安全な方法でダウンロードできます。

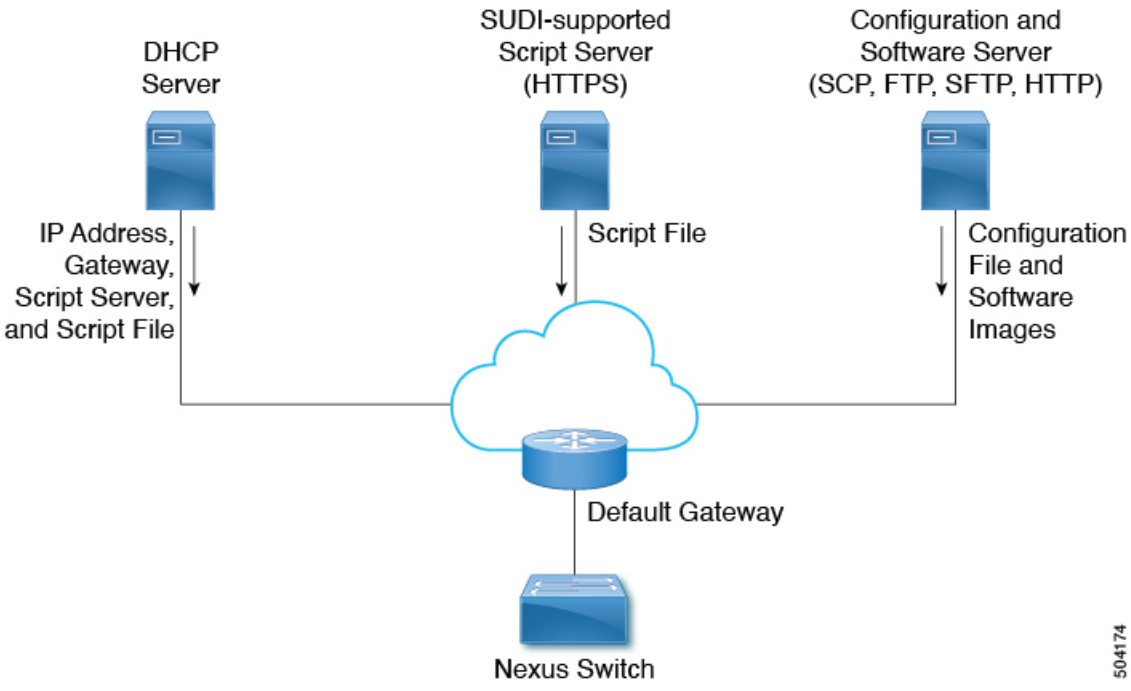
- ファイル サーバーとしての SUDI 対応デバイス
- ファイル サーバーとしての非 SUDI 対応デバイス

ファイル サーバーとしての SUDI 対応デバイス

SUDI がサポートするデバイスは Cisco デバイスです。以前の実装とは異なり、DHCP サーバーは http/tftp ではなく https の場所を提供するようになりました。このシナリオでは、必要なソフトウェア イメージと構成ファイルを含む 1 つ以上のサーバーを除き、DHCP サーバーと SUDI がサポートするスクリプト サーバー（HTTPS サーバー）のみが必要です。

（注） SUDI は TLSv1.2 以下のみをサポートします。また、SUDI ソリューションは https を使用した安全なダウンロードのみを考慮し、sftp は考慮しません。

図 *4:* ファイル サーバー インフラストラクチャとしての *SUDI* 対応デバイス



SUDI 対応デバイスのワークフローは次のとおりです。

- 起動デバイスは SUDI 対応であり、SUDI 証明書を検証するために必要なトラスト ストアがあります。

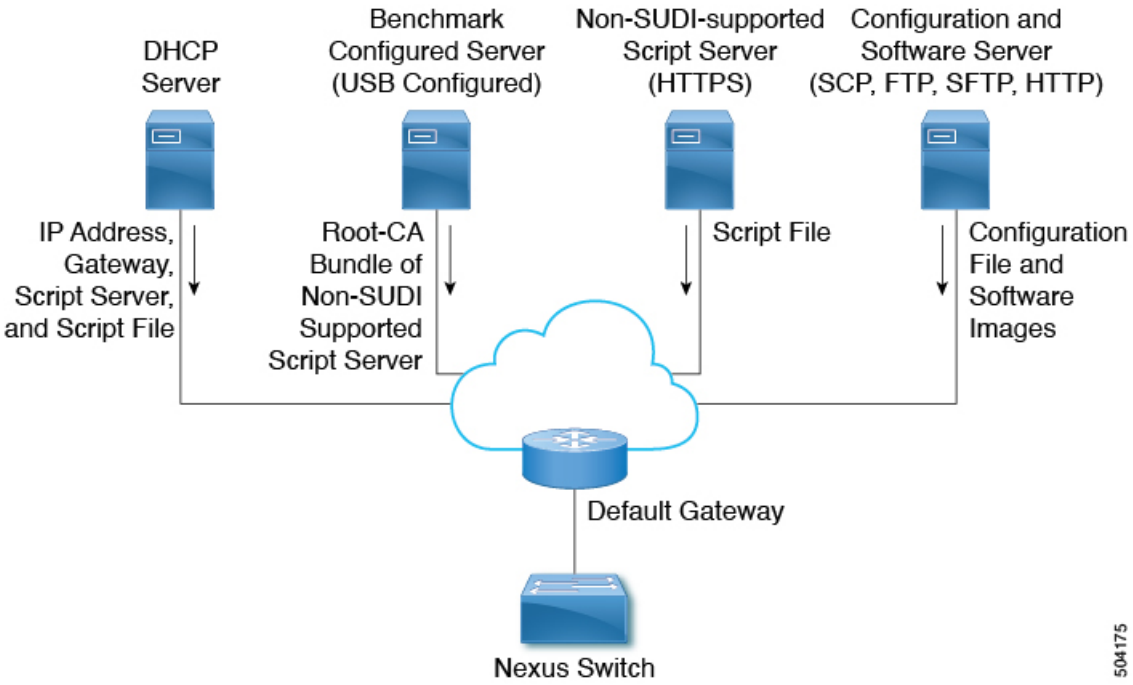
- 起動デバイスは DHCP 検出を送信します
- DHCP サーバーは、https サーバーの詳細で起動デバイスに応答します
- デバイスは、標準の SSL API を使用して安全なチャネルを確立します
- 認証は両側でSUDIを検証することで行われます
- **poap.py** のダウンロード

ファイル サーバーとしての SUDI 対応デバイス

このシナリオでは、ルート CA バンドルをブート デバイスにインストールする必要があります。認証にはルートCAバンドルが必要です。ここでは、必要なソフトウェア イメージと構成ファイルを含む 1 つ以上のサーバー以外に、DHCP サーバー、中間デバイス、および非 SUDI サポート スクリプト サーバー（HTTPS サーバー）が必要です。

DHCP オファーには、ルート CA バンドルが利用可能な中間サーバーの詳細が含まれています。中間デバイスはSUDIをサポートする必要があります。ブートデバイスは中間デバイスを使用してルートCAバンドルをダウンロードしてインストールし、ファイルサーバーと通信します。中間デバイスを最初にプロビジョニングする必要があります。

図 *5:* ファイル サーバー インフラストラクチャとしての非 *SUDI* 対応デバイス



非対応 SUDI デバイスのワークフローは次のとおりです。

- 起動デバイスは SUDI 対応であり、SUDI 証明書を検証するために必要なトラスト ストアがあります。
- ルート CA バンドルを使用してサーバーをホストする中間デバイスも SUDI 対応です

- 起動デバイスは DHCP 検出を送信します
- DHCP サーバーは、https サーバーの詳細とルート CA サーバーの詳細で起動デバイスに応答します
- ブートデバイスが中間デバイスに到達し、CA バンドルを取得して、それをトラストストアに追加します
- 起動デバイスがファイル サーバーに到達し、**poap.py** をダウンロードします。

ベンチマーク構成されたデバイス

非 SUDI サポート スクリプト サーバのルート CA 証明書チェーン ファイルは、ベンチマーク構成済みサーバの `/bootflash/poap/sudi_fs` に配置する必要があります。

（注） ベンチ構成済みデバイスのポートを変更するには、**file-server** <*port-number*> コマンドを使用します。ポート 80（http）やポート 443（https）などの標準ポートの使用は避けてください。

file-server <*port-number*> コマンドは、管理インターフェイスを介してコンテンツを提供するだけです。

古いイメージで出荷されたデバイスの安全な POAP

セキュア POAP のサポートは、安全な POAP 機能を備えたイメージとともに出荷されるデバイスでのみ利用できます。

デバイスに安全な POAP 機能がない場合は、レガシー DHCP オプションを使用して、デバイスをセキュア POAP をサポートする新しいバージョンのイメージに移動します。次に、これらのデバイスをリロードして、安全な POAP 機能を使用できます。

安全な POAP のトラブルシューティング

安全な POAP に関するデバッグ情報を収集するには、次の手順を実行します。

1. オプション 43 の IPv4 のデバッグ オプションを 1 に設定し、オプション 17 の IPv6 のデバッグ オプションを設定します。

   デバッグ オプションは、追加のログを有効にします。

2. スイッチが POAP の 1 サイクルを実行できるようにします。
3. POAP を中止します。
4. システムが起動したら、**show tech-support poap** コマンドを実行します。

   このコマンドは、POAP のステータスまたは構成を表示します。

POAP の無効化

POAP は、システムに構成がない場合に有効になります。ブートアップの一部として実行されます。ただし、初期設定時に POAP の有効化をバイパスできます。POAP を永続的に無効にする場合 (システムに構成がない場合でも)、「system no poap」コマンドを使用できます。このコマンドは、(構成がない場合でも) 次の起動時に POAP が開始されないようにします。POAP を有効にするには、「system poap」コマンドまたは「write erase poap」コマンドを使用します。「write erase poap」コマンドは、POAP フラグを消去し、POAP を有効にします。

- 例：POAP の無効化

```
switch# system no poap
switch# sh boot
Current Boot Variables:
 sup-1
NXOS variable = bootflash:/nxos.9.2.1.125.bin
Boot POAP Disabled

POAP permanently disabled using 'system no poap'


Boot Variables on next reload:

sup-1
NXOS variable = bootflash:/nxos.9.2.1.125.bin
Boot POAP Disabled

POAP permanently disabled using 'system no poap'


switch# sh system poap
System-wide POAP is disabled  using exec command 'system no poap'
POAP will be bypassed on write-erase reload.
(Perpetual POAP cannot be enabled when system-wide POAP is disabled)
```

- 例：POAP の有効化

```
switch# system poap

switch# sh system poap

System-wide POAP is enabled
```

- 例：POAP の消去

```
switch# write erase poap
This command will erase the system wide POAP disable flag only if it is set.
Do you wish to proceed anyway? (y/n)  [n] y
System wide POAP disable flag erased.

switch# sh system poap
System-wide POAP is enabled
```

POAP コンフィギュレーション スクリプト

シスコから提供される参照スクリプトでは、次の機能がサポートされています。

- スイッチ固有の識別子（シリアル番号など）を取得します。
- スイッチ上に nx-os ソフトウェア イメージがまだ存在しない場合は、それををダウンロードします。nx-os イメージがスイッチ上にインストールされ、次回のリブート時に使用されます。
- ダウンロードされた設定がスイッチの次回のリブート時に適用されるようにスケジュールします。
- スタートアップ構成として構成を保存します。

Python プログラミング言語と Tool Command Language（Tcl）を使用して開発された構成スクリプトのサンプルが用意されています。これらのスクリプトのいずれかを、自分のネットワーク環境に合わせてカスタマイズできます。次のリンクでPython スクリプトにアクセスして、Cisco Nexus 9000 シリーズ スイッチ上の POAP を実行できます。https://github.com/datacenter/nexus9000/tree/master/nx-os/poap。

Python プログラミング言語は CLI Commands を実行できる 2 つの API を使用します。これらの API については、次の表で説明します。これらの API の引数は CLI コマンドの文字列です。

API	説明
cli()	制御文字/特殊文字を含む CLI コマンドの未処理の出力を返します。
clid()	XML をサポートする CLI コマンドの場合、この API はコマンド出力を Python ディクショナリとして返します。 この API は、**show** コマンドの出力の検索に役立ちます。

POAP コンフィギュレーション スクリプト

Python プログラミング言語を使用して開発された構成スクリプトのサンプルが用意されています。提供されているスクリプトを使用し、ネットワーク環境の要件を満たすように変更することをお勧めします。

POAP スクリプトは https://github.com/datacenter/nexus9000/blob/master/nx-os/poap/poap.py にあります。

Python を使用してスクリプトを変更するには、ご使用のプラットフォームの『*Cisco NX-OS Python API* リファレンス ガイド』を参照してください。

POAP スクリプトおよび POAP スクリプト オプションの使用

POAP スクリプトを使用する前に、次の操作を実行します。

1. スクリプトの上部にあるオプションディクショナリを編集して、セットアップに関連するすべてのオプションがスクリプトに含まれるようにします。デフォルトを（デフォルトのオプション機能で）直接変更しないでください。
2. シェル コマンドを使用して、表示されているように POAP スクリプトの MD5 チェックサムを更新します。

```
f=poap_nexus_script.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i
"s/^#md5sum=.*/#md5sum=\"$(md5sum $f.md5 | sed 's/ .*//')\"/" $f
```

3. デバイスにスタートアップ構成がある場合は、書き込み消去を実行してデバイスをリロードします。

次の POAP スクリプト オプションを指定して、POAP スクリプトの動作を変更できます。サーバーからファイルをダウンロードするときは、ホスト名、ユーザー名、およびパスワードのオプションが必要です。パーソナリティを除くすべてのモードでは、target_system_image も必要です。必須パラメータはスクリプトによって強制され、必須パラメータが存在しない場合、スクリプトは中止されます。ホスト名、ユーザー名、およびパスワードを除くすべてのオプションには、デフォルトのオプションがあります。オプションディクショナリでオプションを指定しない場合、デフォルトが使用されます。

- **username**

  サーバーからファイルをダウンロードするときに使用するユーザー名。

- **password**

  サーバーからファイルをダウンロードするときに使用するパスワード。

- **hostname**

  ファイルのダウンロード元のサーバーの名前またはアドレス。

- モード（**Certificate verification mode**）

  デフォルトは **serial_number** です。

  次のいずれかのオプションを使用します。

  - パーソナリティ

    tarball からスイッチを復元する方法。

  - **SERIAL_NUMBER**

    構成ファイル名を決定するスイッチのシリアル番号。構成ファイルのシリアル番号の形式は conf.*serialnumber* です。例: conf.FOC123456

  - **hostname**

構成ファイル名を決定するためにDHCPオプションで受け取ったホスト名。構成ファイルのホスト名の形式は、conf_*hostname.cfg* です。例：conf_3164-RS.cfg

- **mac**

  構成ファイル名を決定するインターフェイスの MAC アドレス。構成ファイルのホスト名の形式は、conf_*macaddress*.cfg です。例：conf_7426CC5C9180.cfg

- **raw**

  構成ファイル名は、オプションで指定されたとおりに使用されます。ファイル名は変更されません。

- **location**

  CDP ネイバーは、構成ファイル名を決定するために使用されます。構成ファイル内の場所の形式は conf_*host_intf*.cfg です。ここで、*host* は POAP インターフェースを介してデバイスに接続されているホストであり、*intf* は POAP インターフェースが接続されているリモート インターフェースです。例：conf_remote-switch_Eth1_8.cfg

- **必要なスペース**

  POAP の特定の反復に必要な KB 単位のスペース。デフォルト値は 100,000 です。複数ステップのアップグレードの場合、ターゲット イメージのアップグレード パスにある最後のイメージのサイズを指定します。

- **transfer_protocol**

  VSH でサポートされている http、https、ftp、scp、sftp、tftp などの転送プロトコル。デフォルトは scp です。

- **config_path**

  サーバー上の構成ファイルのパス。例：/tftpboot。デフォルトは /var/lib/tftpboot です。

- **target_system_image**

  リモート サーバーからダウンロードするイメージの名前。これは、POAPが完了した後に取得するイメージです。このオプションは、パーソナリティを除くすべてのモードで必須のパラメータです。デフォルトは「」です。

- **target_image_path**

  サーバー上のイメージへのパス。例：/tftpboot。デフォルトは /var/lib/tftpboot です。

- **destination-path**

  イメージと MD5 サムをダウンロードするパス。デフォルトは /bootflash です。

- **destination_system_image**

  指定宛先イメージ ファイル名。指定しない場合、デフォルトはtarget_system_image名になります。

- **user_app_path**

ユーザー スクリプト、エージェント、およびユーザー データが配置されているサーバー上のパス。デフォルトは /var/lib/tftpboot です。

- **disable_md5**

  これは、MD5 チェックを無効にする必要がある場合は True です。デフォルトは [いいえ（False）] です。

- **midway_system_image**

  途中のシステムアップグレードに使用するイメージの名前。デフォルトでは、POAP スクリプトはアップグレードパスで必要な中間イメージの名前を見つけて使用します。2 段階アップグレードで別の中間イメージを選択する場合は、このオプションを設定します。デフォルトは「」です。

- **source_config_file**

  raw モードを使用する場合の構成ファイルの名前。デフォルトは poap.cfg です。

- **vrf**

  ダウンロードなどに使用する VRF。VRF は POAP プロセスによって自動的に設定されます。デフォルトは POAP_VRF 環境変数です。

- **destination_config**

  ダウンロードした構成に使用する名前。デフォルトは poap_replay.cfg です。

- **split_config_first**

  構成を分割する必要がある場合に、最初の構成部分に使用する名前。構成を有効にするためにリロードするときにのみ適用されます。デフォルトは poap_1.cfg です。

- **split_config_second**

  構成が分割されている場合に 2 番目の構成部分に使用する名前。デフォルトは poap_2.cfg です。

- **timeout_config**

  構成ファイルのコピーのタイムアウト（秒単位）。デフォルトは 120 です。レガシー イメージ以外の場合、このオプションは使用されず、POAP プロセスがタイムアウトします。レガシー イメージの場合、FTP はこのタイムアウトをコピー プロセスではなくログイン プロセスに使用しますが、scp および他のプロトコルはこのタイムアウトをコピー プロセスに使用します。

- **timeout_copy_system**

  システム イメージのコピーのタイムアウト (秒単位)。デフォルトは 2100 です。レガシー イメージ以外の場合、このオプションは使用されず、POAP プロセスがタイムアウトします。レガシー イメージの場合、FTP はこのタイムアウトをコピー プロセスではなくログイン プロセスに使用しますが、scp および他のプロトコルはこのタイムアウトをコピー プロセスに使用します。

- **timeout_copy_personality**

パーソナリティ tarball のコピーのタイムアウト (秒単位)。デフォルトは 900 です。レガシーイメージ以外の場合、このオプションは使用されず、POAPプロセスがタイムアウトします。レガシー イメージの場合、FTP はこのタイムアウトをコピー プロセスではなくログイン プロセスに使用しますが、scp および他のプロトコルはこのタイムアウトをコピー プロセスに使用します。

- **timeout_copy_user**

  ユーザー スクリプトとエージェントをコピーする際のタイムアウト (秒単位)。デフォルトは 900 です。レガシー イメージ以外の場合、このオプションは使用されず、POAP プロセスがタイムアウトします。レガシー イメージの場合、FTP はこのタイムアウトをコピー プロセスではなくログイン プロセスに使用しますが、scp および他のプロトコルはこのタイムアウトをコピー プロセスに使用します。

- **personality_path**

  パーソナリティ tarball のダウンロード元のリモート パス。tarball がダウンロードされ、パーソナリティ プロセスが開始されると、パーソナリティは、tarball 設定内で指定された場所から将来的にすべてのファイルをダウンロードします。デフォルトは /var/lib/tftpboot です。

- **source_tarball**

  ダウンロードするパーソナリティ tarball の名前。デフォルトは、personality.tar です。

- **destination_tarball**

  ダウンロード後のパーソナリティ tarball の名前。デフォルトは、personality.tar です。

POAP の DNS なしでの DHCP サーバーのセットアップ

Cisco NX-OS リリース 7.0(3)I6(1) 以降、tftp-server-name は DNS オプションなしで使用できます。以前のリリースで DNS なしで POAP 機能を有効にするには、150 のカスタム オプションを使用して tftp-server-address を指定する必要があります。

tftp-server-address オプションを使用するには、dhcpd.conf ファイルの先頭で次を指定します。

```
option tftp-server-address code 150 = ip-address;
```

例：

```
host MyDevice {
    option dhcp-client-identifier "\000SAL12345678";
    fixed-address 2.1.1.10;
    option routers 2.1.1.1;
    option host-name "MyDevice";
    option bootfile-name "poap_nexus_script.py";
    option tftp-server-address 2.1.1.1;
}
```

次の例は、IPv6 を介した POAP の DHCPv6 の設定を示しています。

```
default-lease-time 3600;
max-lease-time 3600;
log-facility local7;
```

```
subnet6 2003::/64 {

        # This statement configures actual values to be sent
        # RTPREFIX option code = 243, RTPREFIX length = 22
      # Ignore value 22. It is something related to option-size RT_PREFIX option length.

        # lifetime = 9000 seconds
        # route ETH1_IPV6_GW/64
        # metric 1
        option dhcp6.next-hop-rt-prefix 2003::2222 243 22 9000 0 1 ::;
        #ipv6 ::/0 2003::2222
        #Another example - support not there in NXOS - CSCvs05271:
       #option dhcp6.next-hop-rt-prefix 2003::2222 243 22 9000 112 1 2003::1:2:3:4:5:0;

        #ipv6 2003::1:2:3:4:5:0/112 2003::2222

        # Additional options
        #option dhcp6.name-servers fec0:0:0:1::1;
        #option dhcp6.domain-search "domain.example";

        range6 2003::b:1111 2003::b:9999;
        #range6 2003::c:2222 2003::c:2222;
        option dhcp6.bootfile-url "tftp://2003::1111/poap_github_v6.py";
```

POAP の一部としてのユーザー データ、エージェント、およびスクリプトのダウンロードと使用

オプション ディクショナリの下に、**download_scripts_and_agents** 関数があります。ユーザー スクリプトとデータをダウンロードする場合は、最初の **poap_log** 行のコメントを外し、一連の **download_user_app** 関数呼び出しを使用して各アプリケーションをダウンロードします。古い Cisco NX-OS バージョンはディレクトリの再帰的コピーをサポートしていないため、そのようなディレクトリは tarball (TAR アーカイブ) に入れてから、スイッチで一度解凍する必要があります。**download_scripts_and_agents** 関数のパラメータは次のとおりです。

- **source_path** - ファイルまたは tarball がある場所へのパス。このパラメータは必須です。例：/var/lib/tftpboot
- **source_file** - ダウンロードするファイルの名前。このパラメータは必須です。例：agents.tar、script.py など。
- **dest_path** - スイッチ上のファイルをダウンロードする場所。以前に存在しなかったディレクトリが作成されます。これは省略可能なパラメータです。デフォルトは /bootflash です。
- **dest_file** - ダウンロードしたファイルに付ける名前。これは省略可能なパラメータです。デフォルトは変更されていない source_file です。
- **unpack** - アンパック用の tarball が存在するかどうかを示します。解凍は **tar -xf** *tarfile* **-C /bootflash** で行います。これは省略可能なパラメータです。デフォルトは [いいえ（False）] です。
- **delete_after_unpack** - アンパックが成功した後にダウンロードした tarball を削除するかどうかを示します。unpack が False の場合、効果はありません。デフォルトは [いいえ（False）] です。

ダウンロード機能を使用すると、POAP の実行に必要なすべてのエージェントとファイルをダウンロードできます。エージェントを開始するには、POAP によってダウンロードされた実行構成に構成が存在する必要があります。次に、エージェント、スケジューラ、および cron エントリを EEM とともに使用できます。

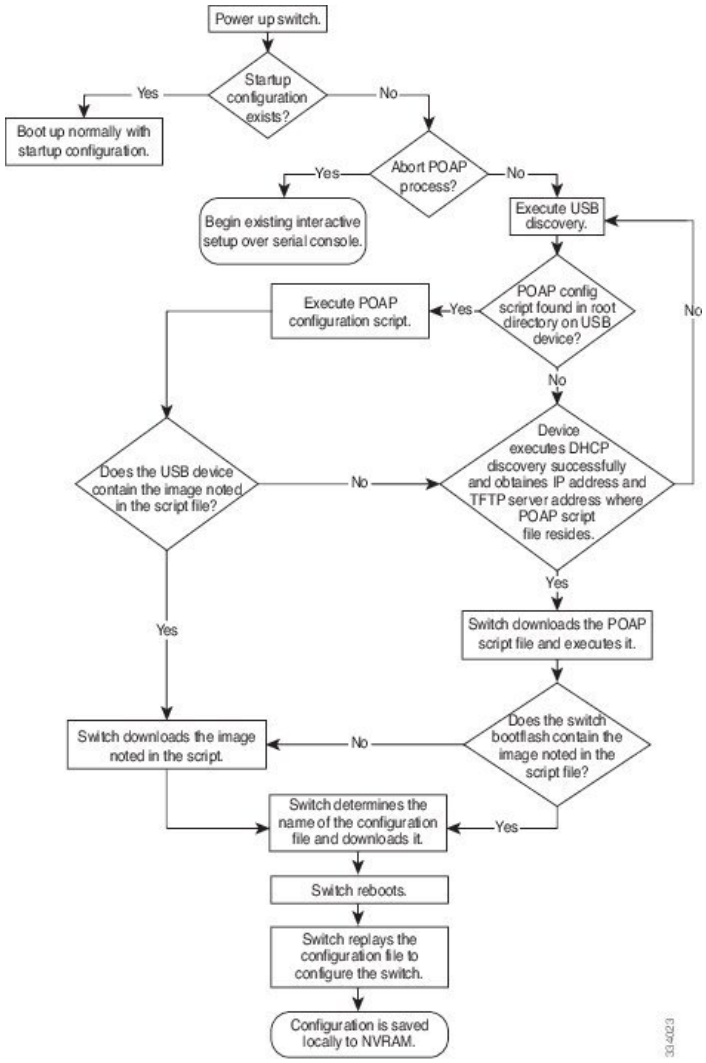
POAP 処理

POAP プロセスには次のフェーズがあります。

1. 電源投入
2. USB の検出
3. DHCP の検出
4. スクリプトの実行
5. インストール後のリロード

これらのフェーズ内では、他の処理や分岐点が発生します。次に、POAP 処理のフロー図を示します。

図 *6 : POAP* 処理



電源投入フェーズ

デバイスの電源を初めて投入すると、デバイスは製造時にインストールされたソフトウェアイメージをロードし、起動に使用する構成ファイルを探します。構成ファイルが見つからなかった場合、POAP モードが開始されます。

起動中、POAP を中止して通常のセットアップに進むかどうかを確認するプロンプトが表示されます。POAP を終了することも、続行することもできます。

（注） POAP を続行する場合、ユーザの操作は必要ありません。POAP を中止するかどうかを確認するプロンプトは、POAP 処理が完了するまで表示され続けます。

POAPモードを終了すると、通常のインタラクティブなセットアップスクリプトが開始されます。POAP モードを続行すると、すべての前面パネルのインターフェイスはデフォルト設定で構成されます。

USB 検出フェーズ

POAP が開始すると、プロセスはアクセス可能なすべての USB デバイスのルート ディレクトリから POAP スクリプト ファイル（Python スクリプト ファイル、poap_script.py）、構成ファイル、およびシステムとキックスタート イメージを検索します。

スクリプト ファイルが USB デバイスで見つかった場合、POAP はスクリプトの実行を開始します。スクリプト ファイルが USB デバイスに存在しない場合は、POAP は DHCP の検出を実行します（障害が発生した場合は、POAP が成功または手動で POAP プロセスを停止するまで、POAP プロセスは USB 検出と DHCP 検出を交互に実行します）。

構成スクリプトで指定されたソフトウェアイメージおよびスイッチ構成ファイルが存在する場合、POAP は、それらのファイルを使用して、ソフトウェアをインストールし、スイッチを構成します。ソフトウェア イメージおよびスイッチ構成ファイルが USB デバイスに存在しない場合、POAP はクリーン アップをして DHCP フェーズを最初から開始します。

DHCP 検出フェーズ

スイッチは、前面パネルのインターフェイスまたは MGMT インターフェイスで、DHCP サーバーからの DHCP オファーを要請する DHCP 検出メッセージを送信します。（次の図を参照してください）。Cisco Nexus スイッチ上の DHCP クライアントは、クライアント ID オプションにスイッチ シリアル番号を使用して、それ自体を DHCP サーバーに識別させます。DHCP サーバーはこの ID を使用して、IP アドレスやスクリプト ファイル名などの情報を DHCP クライアントに返すことができます。

POAP には、最低 3600 秒（1 時間）の DHCP リース期間が必要です。POAP は、DHCP リース期間を確認します。DHCP リース期間が 3600 秒（1 時間）に満たない場合、POAP は DHCP ネゴシエーションを実行しません。

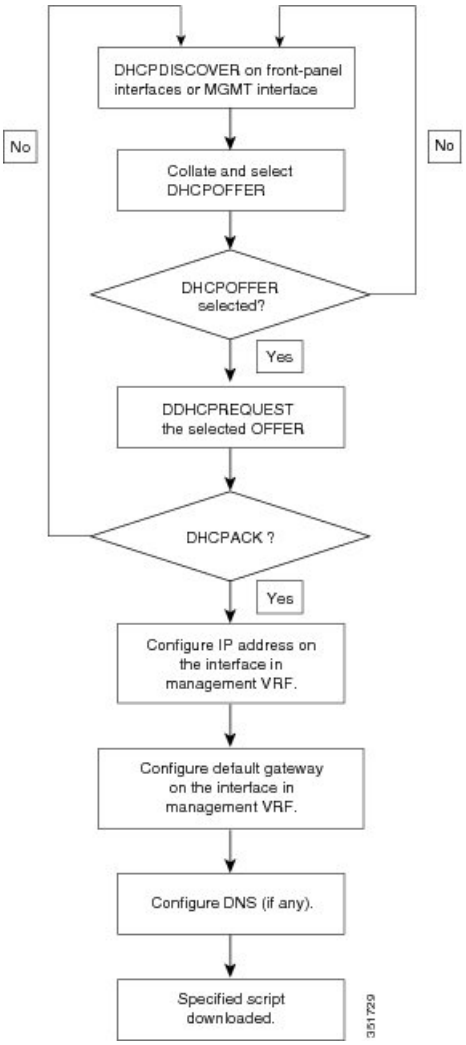
また、DHCP 検出メッセージでは、DHCP サーバーからの次のオプションを要請します。

- TFTP サーバ名または TFTP サーバ アドレス：DHCP サーバは TFTP サーバ名または TFTP サーバ アドレスを DHCP クライアントに中継します。DHCP クライアントはこの情報を使用して TFTP サーバーに接続し、スクリプト ファイルを取得します。
- ブートファイル名：DHCP サーバは DHCP クライアントにブートファイル名を中継します。ブートファイル名には、TFTP サーバ上のブートファイルへの完全パスが含まれます。DHCP クライアントは、この情報を使用してスクリプト ファイルをダウンロードします。

要件を満たす複数の DHCP オファーが受信されると、最初に到着したものが受け入れられ、POAP プロセスは次の段階に進みます。デバイスは、選択された DHCP サーバとの DHCP ネゴシエーション（要求と確認応答）を実行し、DHCP サーバはスイッチに IP アドレスを割り当てます。POAP 処理の後続のステップでエラーが発生すると、IP アドレスは DHCP に戻されます。

要件を満たす DHCP オファーが存在しない場合、スイッチは DHCP ネゴシエーション（要求と確認応答）を実行せず、IP アドレスは割り当てられません。

図 *7: DHCP* 検出プロセス



POAP ダイナミック ブレークアウト

Cisco NX-OS リリース 7.0(3)I4(1) 以降、POAP は、破損したポートの 1 つの背後にある DHCP サーバーを検出しようとして、ポートを動的に分割します。以前は、ブレークアウトケーブルがサポートされていなかったため、POAP に使用される DHCP サーバーは通常のケーブルに直接接続する必要がありました。

POAP は、どのブレイクアウト マップ (たとえば、10gx4、50gx2、25gx4、または 10gx2) が DHCP サーバーに接続されたリンクを起動するかを決定します。どのポートでもブレイクアウトがサポートされていない場合、POAP はダイナミック ブレイクアウト プロセスをスキップします。ブレークアウトループが完了すると、POAP は通常どおり DHCP 検出フェーズを続行します。

（注） ダイナミックブレイクアウトの詳細については、デバイスのインターフェイス構成ガイドを参照してください。

スクリプトの実行フェーズ

デバイスがDHCP確認応答の情報を使用してデバイス自体をブートストラップした後で、スクリプト ファイルが TFTP サーバーからダウンロードされます。

スイッチは、コンフィギュレーション スクリプトを実行します。これにより、ソフトウェアイメージのダウンロードとインストール、およびスイッチ固有のコンフィギュレーションファイルのダウンロードが行われます。

ただし、この時点では、構成ファイルはスイッチに適用されません。スイッチ上で現在実行中のソフトウェアイメージが構成ファイル内の一部のコマンドをサポートしていない可能性があるためです。新しいソフトウェアイメージがインストールされた場合、スイッチのリブート後にそのソフトウェアイメージの実行が開始されます。その時点でスイッチにコンフィギュレーションが適用されます。

（注） スイッチの接続が切断されると、スクリプトは停止し、スイッチはオリジナルのソフトウェアイメージとブートアップ変数をリロードします。

インストール後のリロード フェーズ

スイッチが再起動し、アップグレードされたソフトウェアイメージ上でコンフィギュレーションが適用（リプレイ）されます。その後、スイッチは、実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします。

POAPv3

PowerOn 自動プロビジョニング バージョン 3（POAPv3）は、Cisco NX-OS リリース 9.3(5) で導入されました。この機能を使用すると、POAPを介してライセンス、RPM、および証明書をインストールできます。

POAPを介してライセンス、RPM、または証明書をインストールするには、次の手順を実行します。

1. ボックスのシリアル番号を名前にして、POAP サーバーにフォルダを作成します。
2. インストールするファイルを含む .yaml または .yml ファイルを作成します。ファイル名が <serial-number>.yaml または .yml 形式 に含まれていることを確認してください。
3. .yaml または .yml ファイルの MD5 チェックサムを作成します。
4. .yaml ファイルの形式が次の形式に似ていることを確認してください。

```
Version : 1

Target-image : nxos.9.3.4.bin

Description : Yaml for box XYZ12345 poap provisioning. N9k Leaf mode box

License : [license1.lic, XYZ12345/license2.lic, folder1/license3.lic]

RPM :

  - rpm1.rpm

  - patches/reload/rpm2-reload.rpm

  - rpm3.rpm

Certificate : [ssh1.pub, XYZ12345/ssh2key.pub]

Trustpoint :

    CA1 :

        cert_1.p12 : password1 (priv_key_passphrase)

        XYZ12345/CA1/cert_2.pfx : password2

    CA2 :

        CA2/XYZ12345/cert_3.p12 : password3
```

5. yaml キーワードは、上記の例に示されている形式と一致する必要があることに注意してください。
6. すべてのファイルを適切なパスに配置します。
7. install_path 変数を名前としてシリアル番号を持つフォルダが配置されるパスとして POAP スクリプトを更新します。

次のリストに、POAPv3 に関連するガイドラインと制限事項を示します。

- YAML は、あらゆるプログラミング言語のための、人が読んで理解できるデータ シリアル化標準規格です。YAML は YAML Ain't Markup Language の略で、このファイル形式テクノロジはドキュメントで使用されます。これらのドキュメントはプレーンテキスト形式で保存され、. yml 拡張子が追加されます。YAML はファイル形式で、.yml はファイル拡張子です。
- YAML は JSON のスーパーセットであり、YAML パーサーは JSON を認識します。YAML ファイル形式は、読みやすく、コメントが役立つため、構成管理に使用されます。
- yaml で言及されている Target_image は、POAP スクリプト内で言及されている target_system_image パスにのみ保持する必要があります。yaml ファイルの Target_image では、相対パスはサポートされていません。
- .yaml と .yml の両方の拡張機能がサポートされています。これらの拡張機能のいずれかを使用することを選択するオプションがあります。オプションを選択しない場合、<serial> .yaml 拡張子が最初に試行され、失敗した場合は、 .yml が考慮されます。

- 構成ファイルと同様に、yaml/yml の MD5 ファイルが必要です。ただし、disable_md5 が「True」の場合、yaml/yml の MD5 ファイルは必要ありません。
- デバイスの yaml ファイルが見つからない場合、「install_path」が POAP スクリプト ファイルに設定されていますが、POAP ワークフローは従来のパスで続行されます。つまり、RPM、ライセンス、および証明書のインストールは行われません。
- インストール リセットは、RPM インストールを使用した PoAP が Day-0 以外のシナリオで実行される場合、書き込み消去よりも優先されます。
- ISSU は、PoAP 経由で新しいイメージに移動するための新しいデフォルトです。「use_nxos_boot」を使用する必要があることに注意してください。レガシー ブート nxos <> が必要な場合は True です。
- Filetype は、トラストポイントの .pfx、.p12、ライセンスの .lic、. rpms の and.rpm で、チェック/ファイル形式が尊重されない場合、現在の POAP を中止します。
- .rpm の場合、yaml ファイルに元のファイル名を指定する必要があります。

  例：customCliGoApp-1.0-1.7.5.x86_64.rpm から custom.rpm に名前を変更した場合、PoAP は名前の不一致を示して解決します。

  rpm の元の名前を取得するには：

```
bash-4.3$ rpm -qp --qf '%{NAME}-%{VERSION}-%{RELEASE}.%{ARCH}.rpm' custom.rpm
customCliGoApp-1.0-1.7.5.x86_64.rpm
bash-4.3$
```

- POAP 経由の ISSU が開始されると、PoAP の中止がブロックされます。何らかの理由で ISSU が失敗すると、中止機能が再び有効になります。

POAP の注意事項および制約事項

POAP 構成時の注意事項および制約事項は次のとおりです。

- bootflash:poap_retry_debugs.log は、内部目的でのみ POAP-PNP によって入力されるファイルです。このファイルは、POAP 障害が発生した場合には関係ありません。
- Syslog の制限により、securePOAP pem ファイル名の文字長は 230 文字に制限されていますが、セキュア POAP は pem ファイル名に 256 文字の長さをサポートしています。
- この機能が動作するには、スイッチ ソフトウェア イメージで POAP をサポートしている必要があります。
- POAP では、スイッチが設定されて動作可能になった後のスイッチのプロビジョニングをサポートしません。スタートアップ コンフィギュレーションのないスイッチの自動プロビジョニングだけがサポートされます。
- POAP の https プロトコルで **ignore-certificate** キーワードを使用するには、**https_ignore_certificate** オプションをオンにする必要があります。これにより、POAP スク

リプトで HTTPS 転送を正常に実行でき、プロトコルは POAP で機能しないため、このオプション https なしで実行できます。

- Day 0 プロビジョニングに HTTP/HTTPS サーバーを使用する場合は、HTTP ヘッダー内の MAC 情報およびその他の関連詳細に基づいてプロビジョニング手順が提供されます。POAP は、HTTP GET ヘッダーからのこれらの詳細を使用して、正しいプロビジョニングスクリプトが識別されて使用されるようにします。これは、他のベンダー (および他の Cisco OS) で利用可能でした。これらの追加情報は、Cisco Nexus 9000 の Cisco NX-OS リリース 10.2(1) からの HTTP get ヘッダーで利用できます。この機能は、POAP および非 POAP HTTP 取得操作でデフォルトで使用できます。
- copy http/https GET コマンドを使用すると、次のフィールドが HTTP ヘッダーの一部として共有されます。

```
Host: IP address
User-Agent: cisco-nxos
X-Vendor-SystemMAC: System MAC
X-Vendor-ModelName: Switch-Model
X-Vendor-Serial: Serial_Num
X-Vendor-HardwareVersion: Hardwareversion
X-Vendor-SoftwareVersion: sw_version
X-Vendor-Architecture: Architecture
```

- 仮想ポート チャネル（vPC）ペアの一部である Cisco Nexus デバイスをブートストラップするために POAP を使用する場合、Cisco Nexus デバイスは POAP の起動時にそのすべてのリンクをアクティブにします。vPC のリンクの端に二重接続されているデバイスは、Cisco Nexus デバイスに接続されているポート チャネル メンバ リンクにそのトラフィックの一部またはすべての送信を開始する場合があり、トラフィックが失われることがあります。

  この問題を回避するには、リンクが POAP を使用してブートストラップされている Cisco Nexus デバイスへのトラフィックの転送を誤って開始しないように、vPC リンクにリンク集約制御プロトコル（LACP）を設定します。

- POAP を使用して、LACP ポート チャネル経由で Cisco Nexus 9000 シリーズ スイッチのダウンストリームに接続されている Cisco Nexus デバイスをブートストラップした場合、メンバー ポートをポート チャネルの一部としてバンドルできないと、Cisco Nexus 9000 シリーズ スイッチはデフォルトでそのメンバー ポートを一時停止します。この問題を回避するには、インターフェイス コンフィギュレーション モードから **no lacp suspend-individual** コマンドを使用して、そのメンバー ポートを一時停止しないように Cisco Nexus 9000 シリーズ スイッチを構成します。
- 重要な POAP の更新は syslog に記録され、シリアル コンソールから使用可能になります。
- 重大な POAP エラーは、ブートフラッシュに記録されます。ファイル名のフォーマットは *date-time*_poap_*PID*_[init,1,2].log です。ここで、 *date-time* のフォーマットは YYYYMMDD_hhmmss で、*PID* はプロセス ID になります。
- POAP プロンプトで **skip** オプションを使用すると、パスワードと基本的な POAP 設定をバイパスできます。この **skip** オプションを使用すると、管理者ユーザーのパスワードは構成されません。admin ユーザーに有効なパスワードが設定されるまで、コマンドはブロックされます。**copy running-config startup-config**

- **boot poap enable** コマンド（永続的な POAP）がスイッチで有効になっている場合、リロード時に、スタートアップ コンフィギュレーションが存在していても、POAP ブートがトリガーされます。このシナリオで POAP を使用しない場合は、**no boot poap enable** コマンドを使用して boot poap enable 構成を削除します。
- スクリプト ログは、ブートフラッシュ ディレクトリに保存されます。ファイル名のフォーマットは *date-time*_poap_*PID*_script.log です。ここで、 *date-time* のフォーマットは YYYYMMDD_hhmmss で、*PID* はプロセス ID になります。

  スクリプトのログ ファイルの形式を構成できます。スクリプト ファイルのログ形式は、スクリプトで指定されます。スクリプトのログ ファイルのテンプレートにはデフォルトの形式があります。ただし、スクリプト実行ログ ファイルに別の形式を選択できます。
- POAP 機能にライセンスは必要ありません。デフォルトでイネーブルになっています。ただし、POAP 機能が正しく動作するためには、ネットワークの導入前に適切なライセンスがネットワーク内のデバイスにインストールされている必要があります。
- POAP の USB サポートにより、構成スクリプト ファイルを含む USB デバイスを POAP モードでチェックできます。この機能は、Nexus 9300-EX、-FX、-FX2、-FX3、および Nexus 9200-X、-FX2 スイッチでサポートされています。
- デバイスが高いトラフィック レートを受信すると、POAP DHCP トランザクションが失敗することがあります。この問題は、POAP がフロントパネルを使用している場合に発生します。この問題を回避するには、POAP が管理ポートを使用していることを確認してください。
- NX-OS 7.0(3)I7(4) 以降、RFC 3004（DHCP のユーザー クラス オプション）がサポートされています。これにより、POAP は DHCPv4 のユーザー クラス オプション 77 と DHCPv6 のユーザー クラス オプション 15 をサポートできます。DHCPv4 と DHCPv6 の両方のユーザー クラス オプションに表示されるテキストは「Cisco-POAP」です。
  - RFC 3004（DHCP のユーザー クラス オプション）のサポートにより、Nexus 9000 スイッチで IPv6 上の POAP がサポートされます。
    - NX-OS 9.2(2) 以降、IPv6 を介した POAP は、–R ラインカードを備えた Nexus 9504 および Nexus 9508 スイッチでサポートされます。

  IPv6 上の POAP 機能により、IPv4 で障害が発生したときに POAP プロセスが IPv6 を使用できるようになります。この機能は、接続障害が発生したときに IPv4 プロトコルと IPv6 プロトコルの間を循環するように設計されています。
- 安全な POAP の場合は、DHCP スヌーピングが有効になっていることを確認してください。
- POAP をサポートするには、ファイアウォール ルールを設定して、意図しないまたは悪意のある DHCP サーバーをブロックします。
- システムのセキュリティを維持し、POAP をより安全にするには、次のように構成します。
  - DHCP スヌーピングをイネーブルにします。

  - ファイアウォールルールを設定して、意図しない、または悪意のある DHCP サーバーをブロックします。

- POAP は、MGMT ポートとインバンド ポートの両方でサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、POAP/HTTPS 機能のハードウェア SUDI には、POAP スクリプトを安全にダウンロードするオプションが用意されています。
- POAP のデバッグ情報を収集するには、POAP のポスト アボートである **show tech-support poap** コマンドを使用します。
- Cisco NX-OS リリース 10.3(1)F 以降、Cisco Nexus 9808 プラットフォーム スイッチの Cisco Nexus X9836DM-A ライン カードで POAP がサポートされています。
  - Cisco NX-OS リリース 10.4(1)F 以降、Cisco Nexus 9808 スイッチの Cisco Nexus X98900CD-A ライン カードで POAP がサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、Cisco Nexus 9804 プラットフォーム スイッチ、および Cisco Nexus X98900CD-A と X9836DM-A ライン カードで POAP がサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降では、セキュリティ強化のために、スキップ オプションが無効になっています。POAP プロセスが停止しているかどうかに関係なく、ボックスにアクセスするには有効なパスワードを入力する必要があります。
- Cisco NX-OS リリース 10.5(3)F 以降、Cisco 9364E-SG2 ToR スイッチでは POAP がサポートされています。

POAP を使用するためのネットワーク環境の設定

手順

ステップ 1 シスコが提供する基本設定スクリプトを変更するか、独自のスクリプトを作成します。詳細については、『*Python Scripting and API Configuration Guide*』を参照してください。

ステップ 2 構成スクリプトに変更を加えるたびに、bash シェルを使用して、**# f=poap_nexus_script.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*/#md5sum=\"$(md5sum $f.md5 | sed 's/ .*//')\"/" $f** を実行することにより、必ずMD5チェックサムを再計算することを確認してください。詳細については、『*Python API Reference Guide*（*Python API* リファレンス ガイド）』を参照してください。

ステップ 3 （オプション）POAP の構成スクリプトおよびその他の必要なソフトウェア イメージおよびスイッチの構成 ファイルを、スイッチからアクセスできる USB デバイスに配置します。

ステップ 4 DHCP サーバを配置し、このサーバにインターフェイス、ゲートウェイ、および TFTP サーバの IP アドレスと、コンフィギュレーション スクリプト ファイルのパスと名前が指定されたブートファイルを設定します。（この情報は、最初の起動時にスイッチに提供されます）。すべてのソフトウェア イメージおよびスイッチ構成ファイルが USB デバイスにある場合は、DHCP サーバーを配置する必要はありません。

ステップ 5 構成スクリプトをホストするための TFTP または HTTP サーバを展開します。サーバーへの HTTP 要求をトリガーするには、TFTP サーバー名の前に HTTP:// を付けます。HTTPS はサポートされていません。

ステップ 6 URL 部分を TFTP スクリプト名に追加して、ファイル名への正しいパスを表示します。

ステップ 7 ソフトウェア イメージおよびコンフィギュレーション ファイルをホストするための 1 つまたは複数のサーバを配置します。

POAP を使用するスイッチの設定

始める前に

POAP を使用するためにネットワーク環境がセットアップされていることを確認します。

手順

ステップ 1 ネットワークにスイッチを設置します。

ステップ 2 スイッチの電源を投入します。

構成ファイルが見つからない場合は、スイッチは POAP モードで起動して、POAP を中止して通常のセットアップで続行するかどうかを尋ねるプロンプトが表示されます。

POAP モードで起動を続行するためのエントリは必要ありません。

ステップ 3 （オプション）POAP モードを終了して、通常のインタラクティブ セットアップ スクリプトを開始する場合は、**y**（yes）を入力します。

スイッチが起動して、POAP 処理が開始されます。

次のタスク

設定を確認します。

md5 ファイルの作成

構成スクリプトに変更を加えるたびに、bash シェルを使用して、# f=poap_fabric.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*/#md5sum=\"$(md5sum $f.md5 | sed 's/ .*//')\"/" $f を実行することにより、必ず MD5 チェックサムを再計算します。

このプロシージャは、`poap_fabric.py` の md5sum を新しい値に置き換えます（そのファイルに変更があった場合）。

（注） 手順 1 ～ 4 および 7 ～ 8 は、BASH シェルを使用している場合にのみ必要です。他の Linux サーバーにアクセスできる場合、これらの手順は必要ありません。

始める前に

bash シェルにアクセスします。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**feature bash-shell** 例： `switch(config)# feature bash-shell`	BASH シェル機能を有効にします。
ステップ 3	**exit** 例： `switch(config)# exit`	コンフィギュレーション モードを終了します。
ステップ 4	**run bash** 例： `switch# run bash`	Linux BASH を開きます。
ステップ 5	**md5sum /bootflash/nxos.***release_number***.bin > /bootflash/nxos.***release_number***.bin.md5** 例： `bash-4.2$ md5sum /bootflash/nxos.7.0.3.I6.1.bin > /bootflash/nxos.7.0.3.I6.1.bin.md5`	`.bin` ファイルの md5sum を作成します。
ステップ 6	**md5sum /bootflash/poap.cfg > /bootflash/poap.cfg.md5** 例： `bash-4.2$ md5sum /bootflash/poap.cfg > /bootflash/poap.cfg.md5`	`.cfg` ファイルの md5sum を作成します。
ステップ 7	**exit** 例： `switch(config)# exit`	BASH シェルを終了します。

	コマンドまたはアクション	目的

ステップ 8	**dir | i .md5**
	例：
	`switch# dir | i .md5`
	`65 Jun 09 12:38:48 2017 nxos.7.0.3.I6.1.bin.md5`
	`54 Jun 09 12:39:36 2017 poap.cfg.md5`
	`67299 Jun 09 12:48:58 2017 poap.py.md5`
		.md5 ファイルを表示します。

ステップ 9	**copy bootflash:poap.cfg.md5 scp://***ip_address***/**
	例：
	`copy bootflash:poap.cfg.md5 scp://10.1.100.3/`
	`Enter vrf (If no input, current vrf 'default' is considered): management`
	`Enter username: root`
	`root@10.1.100.3's password:`
	`poap.cfg.md5 100%`
	`54 0.1KB/s 00:00`
	`Copy complete.`
		ファイルを構成およびソフトウェア サーバーにアップロードします。

デバイス コンフィギュレーションの確認

構成を確認するためには、次のいずれかのコマンドを使用します。

コマンド	目的
show running-config [**[exclude]** *command*] **[sanitized]**	現在の実行コンフィギュレーションまたはそのコンフィギュレーションのサブセットの内容を表示するには、該当するモードで **show running-config** コマンドを使用します。。 • **exclude**：（任意）特定のコンフィギュレーションを表示から除外します。 **exclude** キーワードのあとに *command* 引数を指定し、表示から特定のコンフィギュレーションを除外します。 • コマンド：（任意）1 つのコマンドのみを、または指定のコマンド ノード下で使用可能なコマンドのサブセットを表示します。 • **sanitized**：（任意）安全な配布と分析のためにサニタイズされたコンフィギュレーションを表示します。 Cisco NX-OS リリース 10.3(2)F 以降、**sanitized** キーワードが Cisco Nexus 9000 シリーズ スイッチでサポートされています。

コマンド	目的
show startup-config	スタートアップ コンフィギュレーションを表示します。 **Note** レイヤ 3 ベースの機能構成が running-config で無効になっている場合、**show startup-config** コマンドはそれらを表示しません。ただし、**copy running startup** コマンドが実行されるまで、構成はスタートアップ PSS にそのまま残ります。
show time-stamp running-config last-changed	実行構成が最後に変更されたときのタイムスタンプを表示します。

次に、**show running-config**コマンドで **sanitized** キーワードを指定した場合の出力例を示します。サニタイズされた構成は、構成の一部の詳細を公開せずに、構成を共有するために使用できます。

このオプションは、実行構成出力の機密ワードを<removed>キーワードによりマスクします。

```
switch# show running-config sanitized

!Command: show running-config sanitized
!Running configuration last done at: Wed Oct 12 09:14:54 2022
!Time: Wed Oct 12 13:52:55 2022

version 10.3(2) Bios:version 07.69

username admin password 5 <removed> role network-admin

copp profile strict
snmp-server user admin network-admin auth md5 <removed> priv aes-128 <removed>
localizedV2key
rmon event 1 log trap <removed> description FATAL(1) owner PMON@FATAL
rmon event 2 log trap <removed> description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 log trap <removed> description ERROR(3) owner PMON@ERROR
rmon event 4 log trap <removed> description WARNING(4) owner PMON@WARNING
rmon event 5 log trap <removed> description INFORMATION(5) owner PMON@INFO
--More--
```

POAP のトラブルシューティング

以下は、POAP を使用する際の既知の問題と提案のリストです。

- 問題：POAP スクリプトの実行がすぐに失敗し、「スクリプトの実行に失敗しました」というステートメントを除いて、syslog または出力が表示されません。

  提案：サーバーで **python** *script-name* コマンドを使用し、構文エラーがないことを確認します。options ディクショナリは Python ディクショナリであるため、各エントリはコンマで区切って、キーまたはオプションと値をコロンで区切る必要があります。

- 問題：正しく使用されていないオプションに応じて、さまざまな場所で TypeError 例外が発生します。

提案：一部のオプションでは整数を使用します(たとえば、タイムアウトやその他の数値)。引用符で囲まれた数値については、optionsディクショナリを確認してください。正しい使用法については、オプション リストを参照してください。

- 問題：POAP over USB が存在するファイルを見つけられません。

  提案：一部のデバイスには 2 つの USB スロットがあります。USB スロット 2 を使用している場合は、オプションで指定する必要があります。

- 問題：POAP に関する問題。

  提案：POAP を中止し、システムが起動したら、**show tech-support poap** コマンドを実行します。これにより、POAP のステータスと構成が表示されます。

POAP パーソナリティの管理

POAP パーソナリティ

Cisco NX-OS リリース 7.0(3)I4(1) で導入された POAP パーソナリティ機能により、ユーザー データ、Cisco NX-OS とサードパーティのパッチ、および構成ファイルをバックアップおよび復元できます。以前のリリースでは、POAP は構成のみを復元できました。

POAP のパーソナリティは、スイッチ上で追跡されたファイルによって定義されます。パーソナリティ ファイルの構成およびパッケージ リストは ASCII ファイルです。

バイナリ バージョンはパーソナリティ ファイルに記録されますが、実際のバイナリ ファイルは含まれません。バイナリ ファイルは通常大きいため、指定されたリポジトリからアクセスします。

パーソナリティ ファイルは .tar ファイルで、通常は一時フォルダに抽出されます。次に例を示します。

```
switch# dir bootflash: 042516182843personality # timestamp name
46985 Dec 06 23:12:56 2015 running-config  Same as "show running-configuration" command.
20512 Dec 06 23:12:56 2015 host-package-list  Package/Patches list
58056 Dec 06 23:12:56 2015 data.tar  User Data
25    Dec 06 23:12:56 2015 IMAGEFILE  Tracked image metadata
```

POAP パーソナリティのバックアップ

スイッチ上でローカルに、またはサーバー上でリモートにPOAPパーソナリティのバックアップを作成できます。スイッチから取得したパーソナリティ バックアップは、同じモデルのスイッチでのみ復元する必要があります。

（注） バックアップに Cisco スケジューラ機能を使用している場合は、次の例に示すように、POAP パーソナリティもバックアップするように設定できます。スケジューラの詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

```
switch(config)# scheduler schedule name weeklybkup
switch(config-schedule)# time weekly mon:07:00
switch(config-schedule)# job name personalitybkup
switch(config-schedule)# exit
switch(config)# scheduler job name personalitybkup
switch(config-job)# personality backup bootflash:/personality-file ; copy
bootflash:/personality-file tftp://10.1.1.1/ vrf management
```

手順の概要

1. **personality backup** [**bootflash:***uri* | **scp:***uri*]

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	必須: **personality backup** [**bootflash:***uri* | **scp:***uri*]
	例：
	`switch# personality backup bootflash:personality1.tar`
	例：
	`switch# personality backup scp://root@2.1.1.1/var/lib/tftpboot/backup.tar`
		POAPパーソナリティのバックアップを作成します。

POAP パーソナリティの構成

POAP パーソナリティをシステムの実行状態またはコミット（起動）状態のどちらから取得するかを指定できます。

手順の概要

1. **configure terminal**
2. **personality**
3. **track** [**running-state** | **startup-state** | **data** *local-directories-or-files*]
4. **binary-location** *source-uri-folder*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	必須: **configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	必須: **personality**
	例：
	`switch# personality`
	`switch(config-personality)#`
		パーソナリティ構成モードに入ります。

ステップ 3	必須: **track** [**running-state** | **startup-state** | **data** *local-directories-or-files*]
	例：
	`switch(config-personality)# track data bootflash:myfile1`
	例：
	`switch(config-personality)# track data bootflash:user_scripts/*.py`
	例：
	`switch(config-personality)# track data bootflash:basedir/*/backup_data`
		POAP パーソナリティの派生方法を指定します。次のオプションを使用できます。
		• **running-state** ：次の情報を取得します。実行構成（**show running-config** コマンドで表示）、ホスト システムのアクティブな Cisco NX-OS パッチとサードパーティ パッケージ、およびイメージ名（**show version** コマンドで表示）。これがデフォルトのオプションです。
		• **startup-state** ：次の情報をキャプチャします。スタートアップ コンフィギュレーション（**show startup-config** コマンドで表示）、ホスト システムでコミットされた Cisco NX-OS パッチおよびサードパーティ パッケージ、およびイメージ名（**show version** コマンドで表示）。
		• **data** *local-directories-or-files* ：バックアップするディレクトリまたはファイルを指定します。このコマンドを複数回入力して、複数のディレクトリおよびファイルをバックアップできます。UNIX 形式のワイルドカード文字がサポートされています。この例では、1 つのフォルダと 2 つのディレクトリが指定されています。
		（注）
		このコマンドを使用してブートフラッシュ内のバイナリ ファイルをバックアップしたり、ブートフラッシュ全体をポイントしたりしないでください。
		（注）

	コマンドまたはアクション	目的
		ゲスト シェル パッケージは追跡されません。 （注） 署名付き RPM（キーが必要）はサポートされていません。POAPパーソナリティ機能は、署名された RPM では機能しません。
ステップ 4	必須: **binary-location** *source-uri-folder* 例： `switch(config-personality)# binary-location scp://remote-dir1/nxos_patches/`	POAP パーソナリティの復元時にバイナリ ファイルを取得するローカル ディレクトリまたはリモート ディレクトリを指定します。このコマンドを複数回（優先順位に従って）入力して、複数の場所を指定できます。

POAP パーソナリティの復元

POAP スクリプトの実行フェーズ中に、現在起動されているスイッチ イメージが Cisco NX-OS リリース 7.0(3)I4(1) 以降である場合、スクリプト内のパーソナリティ モジュールはPOAP パーソナリティを復元します。必要に応じて、スイッチを正しいソフトウェアイメージにアップグレードします。

（注） パーソナリティの復元は、パーソナリティのバックアップに使用されたのと同じソフトウェアイメージを使用して実行されます。新しいイメージへのアップグレードは、POAP パーソナリティ機能ではサポートされていません。新しいイメージにアップグレードするには、通常の POAP スクリプトを使用します。

（注） パーソナリティ スクリプトが何らかの理由（ブートフラッシュに十分なスペースがない、スクリプトの実行に失敗するなど）で失敗した場合、POAP プロセスは DHCP 検出フェーズに戻ります。

復元プロセスは、次のアクションを実行します。

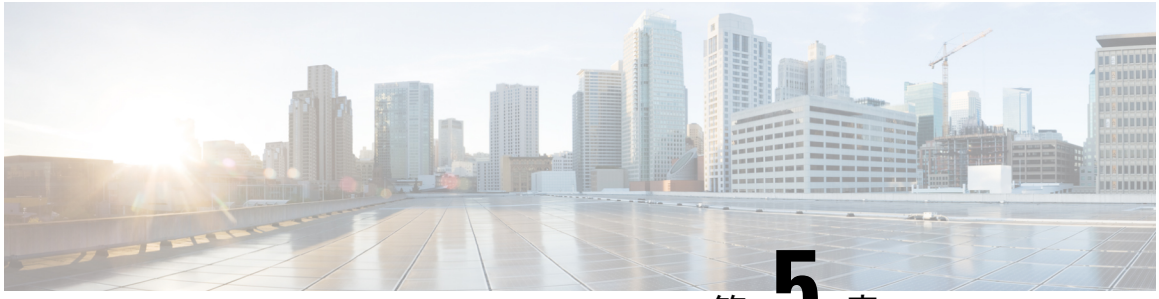
1. ブートフラッシュ内のパーソナリティ ファイルを解凍します。
2. パーソナリティ ファイルを検証します。
3. パーソナリティ ファイルから構成ファイルとパッケージ リスト ファイルを読み取り、ダウンロードするバイナリのリストを作成します。
4. 現在のイメージまたはパッチがパーソナリティ ファイルで指定されたものと異なる場合、バイナリをブートフラッシュ（存在しない場合）にダウンロードし、正しいイメージで再起動してから、パッケージまたはパッチを適用します。

5. 「/」を基準にしてユーザー データ ファイルを解凍します。
6. POAP パーソナリティの構成ファイルをスタートアップ構成にコピーします。
7. スイッチをリブートします。

POAP パーソナリティ サンプル スクリプト

次のサンプル POAP スクリプト（poap.py）には、パーソナリティ機能が含まれています。

```
#md5sum="b00a7fffb305d13a1e02cd0d342afca3"
# The above is the (embedded) md5sum of this file taken without this line, # can be #
created this way:
# f=poap.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*/#md5sum=$(md5sum
 $f.md5 | sed 's/ .*//')/" $f # This way this script's integrity can be checked in case
 you do not trust # tftp's ip checksum. This integrity check is done by
/isan/bin/poap.bin).
# The integrity of the files downloaded later (images, config) is checked # by downloading
 the corresponding file with the .md5 extension and is # done by this script itself.

from poap.personality import POAPPersonality import os

# Location to download system image files, checksums, etc.
download_path = "/var/lib/tftpboot"
# The path to the personality tarball used for restoration personality_tarball =
"/var/lib/tftpboot/foo.tar"
# The protocol to use to download images/config protocol = "scp"
# The username to download images, the personality tarball, and the # patches and RPMs
during restoration username = "root"
# The password for the above username
password = "passwd754"
# The hostname or IP address of the file server server = "2.1.1.1"

# The VRF to use for downloading and restoration vrf = "default"
if os.environ.has_key('POAP_VRF'):
    vrf = os.environ['POAP_VRF']

# Initialize housekeeping stuff (logs, temp dirs, etc.) p = POAPPersonality(download_path,
 personality_tarball, protocol, username, password, server, vrf)

p.get_personality()
p.apply_personality()

sys.exit(0)
```



第 5 章

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

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ー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 コントローラー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 コントローラー/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ー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に注意してください。
- 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にのみ、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）]

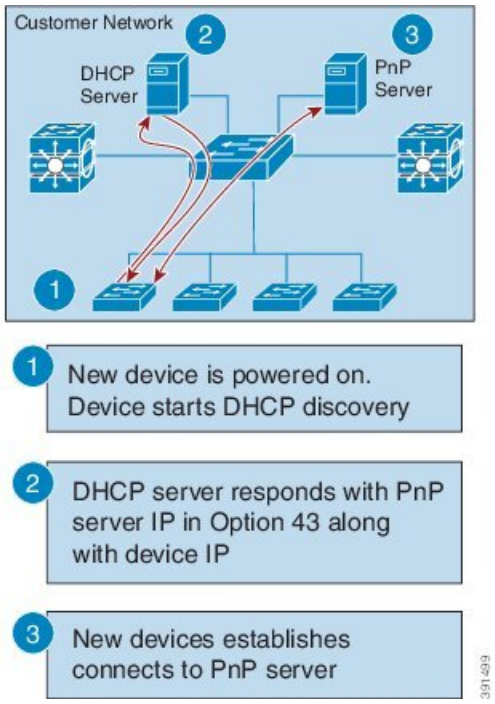
エージェントの構成が失敗した場合は、手動で介入してスイッチを構成する必要があります。

DHCP 検出には、次のフローがあります。

- スイッチ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 8: PnP サーバーの *DHCP* 検出プロセス



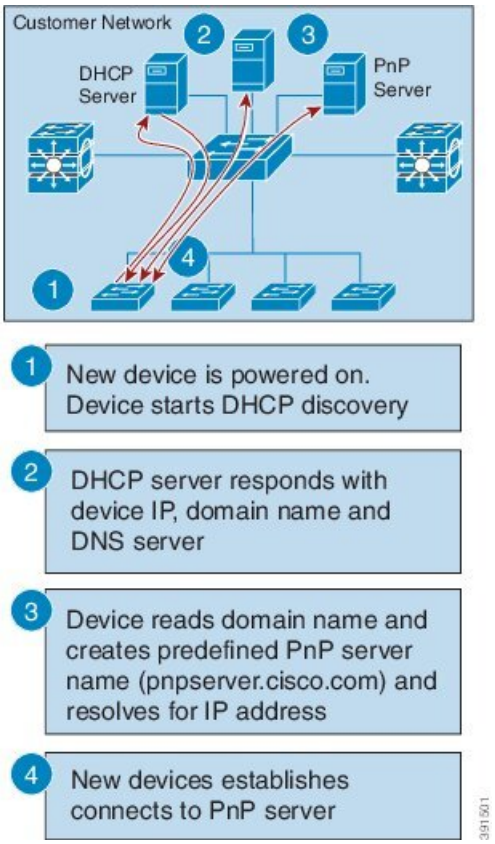
DNS ベースの検出

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 9: 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ら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続行されます。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。

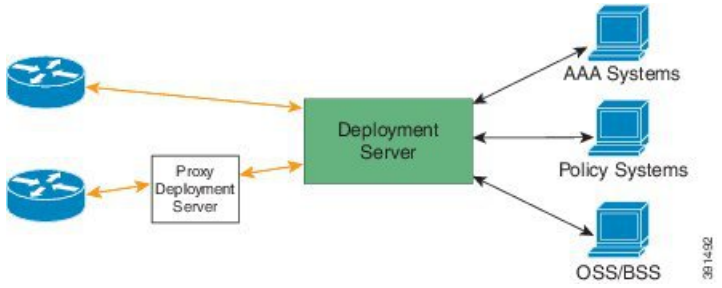
PnP エージェント

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 10: 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ー、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ーとも通信します。

PnPサーバーは、エージェントを別の展開サーバー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ー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、サーバーは認証、承認、アカウンティング（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。

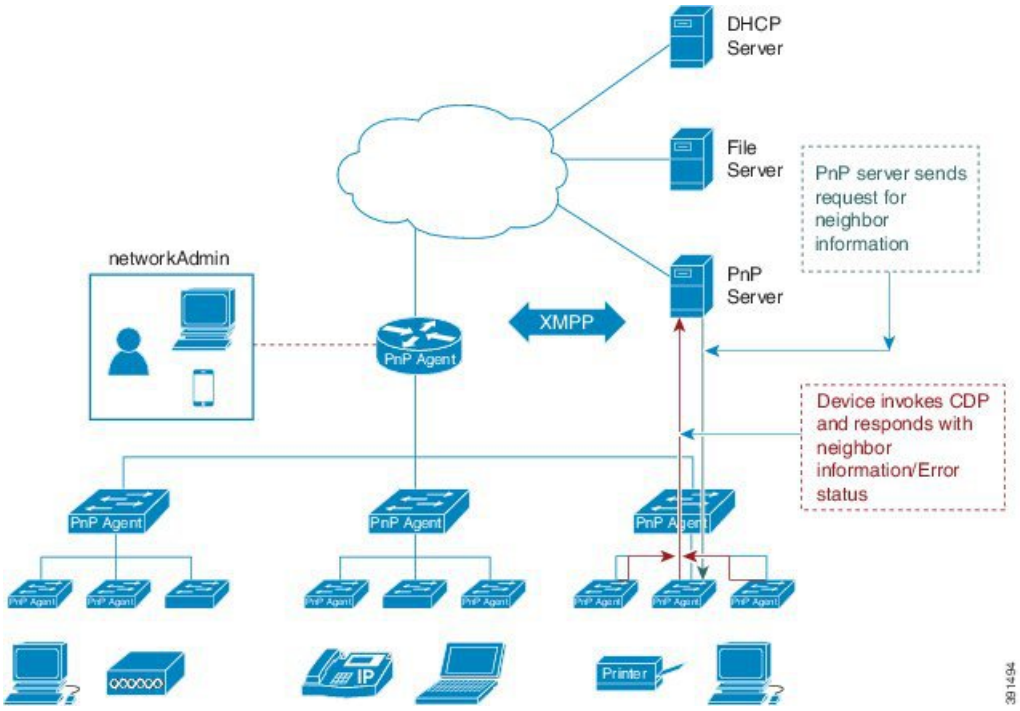
PnP エージェントの展開

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 サーバー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 サーバー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 サーバーに返します。

PnP エージェント ネットワーク トポロジ

Figure 11: 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"
     }
}
```

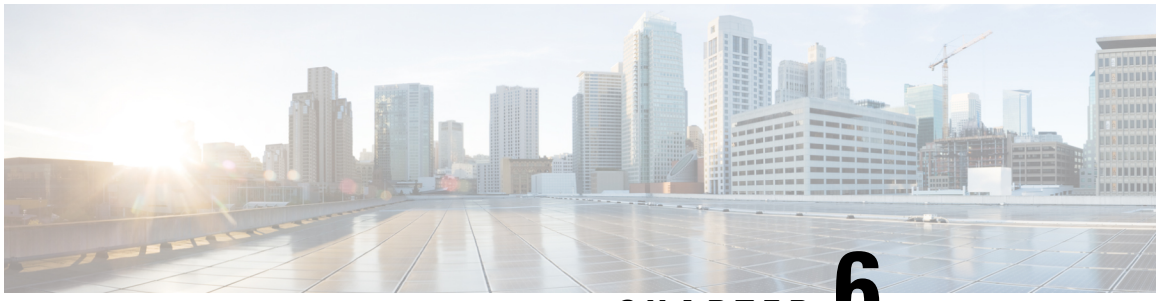
```
Switch# show pnp internal trace

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] Initilizing 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:
```



CHAPTER 6

コマンドライン インターフェイスの概要

この章は、次の内容で構成されています。

CLI プロンプトの概要

デバイスに正常にアクセスすると、コンソール ポートのターミナル ウィンドウまたはリモート ワークステーションに、次の例のような CLI プロンプトが表示されます。

```
User Access Verification
login: admin
Password:<password>
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2013, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php
switch#
```

デフォルトのデバイス ホスト名を変更できます。

CLI プロンプトから、次の方法を実行できます。

- 機能を設定するための CLI コマンドを使用する
- コマンド履歴にアクセスする
- コマンド解析機能を使用する

Note 通常の動作では、ユーザ名の大文字と小文字が区別されます。ただし、コンソール ポートを介してデバイスに接続する場合、ユーザ名がどのように定義されているかに関係なく、すべて大文字でログイン ユーザ名を入力できます。正しいパスワードを入力すれば、デバイスにログインできます。

コマンド モード

ここでは、Cisco NX-OS CLI でのコマンド モードについて説明します。

EXEC コマンド モード

始めてログインするときに、Cisco NX-OS ソフトウェアは EXEC モードになります。EXEC モードで使用可能なコマンドには、デバイスの状態および構成情報を表示する **show** コマンド、**clear** コマンド、ユーザーがデバイス コンフィギュレーションに保存しない処理を実行するその他のコマンドがあります。

グローバル構成コマンド モード

グローバル コンフィギュレーション モードでは、広範なコマンドにアクセスできます。この用語は、デバイスに全体的な影響を与える特性または特徴を示しています。グローバル構成

モードでコマンドを入力すると、デバイスをグローバルに構成したり、より具体的な構成モードを開始してインターフェイスやプロトコルなどの特定の要素を構成したりできます。

SUMMARY STEPS

1. **configure terminal**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。
		Note
		CLIプロンプトが変わり、グローバル コンフィギュレーション モードに入ったことが示されます。

インターフェイス コンフィギュレーション コマンド モード

インターフェイス コンフィギュレーション モードは、グローバル コンフィギュレーション モードから開始する、特定のコンフィギュレーション モードの1例です。デバイスのインターフェイスを設定するには、インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始する必要があります。

インターフェイスごとに多くの機能をイネーブルにする必要があります。インターフェイス コンフィギュレーション コマンドを使用すると、イーサネット インターフェイスや管理インターフェイス（mgmt 0）などの、デバイス上のインターフェイスの動作が変更されます。

インターフェイスの構成の詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* インターフェイス構成ガイド』を参照してください。

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		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。

	Command or Action	Purpose

ステップ 2	**interface** *type number*
	Example:
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		設定するインターフェイスを指定します。
		CLI により、指定したインターフェイスのインターフェイス コンフィギュレーション モードになります。
		Note
		CLI プロンプトが変わり、インターフェイス コンフィギュレーション モードに入ったことが示されます。

サブインターフェイス コンフィギュレーション コマンド モード

グローバル コンフィギュレーション モードから、サブインターフェイスと呼ばれる VLAN インターフェイスを設定するためのコンフィギュレーションサブモードにアクセスできます。サブインターフェイス コンフィギュレーション モードでは、1 つの物理インターフェイスに複数の仮想インターフェイスを設定できます。サブインターフェイスは、別個の物理インターフェイスとしてプロトコルに認識されます。

また、サブインターフェイスは、プロトコルによる単一インターフェイスでの複数のカプセル化を可能にします。たとえば、IEEE 802.1Q カプセル化を設定して、サブインターフェイスを VLAN に関連付けることができます。

サブインターフェイスの構成の詳細については、『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』を参照してください。

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number.subint*

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**interface** *type number.subint*
	Example:
	`switch(config)# interface ethernet 2/2.1`
	`switch(config-subif)#`
		設定する VLAN インターフェイスを指定します。
		CLI は、指定した VLAN インターフェイスに対するサブインターフェイス コンフィギュレーション モードを開始します。

	Command or Action	Purpose
		Note CLI プロンプトが変わり、インターフェイス構成モードに入ったことが示されます。

コマンド モードの保存と復元

Cisco NX-OS ソフトウェアでは、現在のコマンド モードを保存しておき、機能を設定した後に、前のコマンド モードを復元することができます。この項で説明している **push** コマンドはコマンドモードを保存し、 **pop** コマンドは、コマンド モードを復元します。

次に、コマンド モードを保存し、復元する例を示します。

```
switch# configure terminal
switch(config)# event manager applet test
switch(config-applet)# push
switch(config-applet)# configure terminal
switch(config)# username testuser password newtest
switch(config)# pop
switch(config-applet)#
```

コンフィギュレーション コマンド モードの終了

手順の概要

1. **exit**
2. **end**
3. （任意） **Ctrl+Z**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**exit**
	例：
	`switch(config-if)# exit`
	`switch(config)#`
		現在のコンフィギュレーション コマンド モードを終了して、以前のコンフィギュレーション コマンド モードに戻ります。

ステップ 2	**end**
	例：
	`switch(config-if)# end`
	`switch#`
		現在のコンフィギュレーション コマンド モードを終了して、EXEC モードに戻ります。

	コマンドまたはアクション	目的

ステップ 3	（任意） **Ctrl+Z**
	例：
	`switch(config-if)# ^z`
	`switch#`
		現在のコンフィギュレーション コマンド モードを終了して、EXEC モードに戻ります。
		注意
		有効なコマンドを入力してから、コマンド ラインの最後で **Ctrl+Z** を使用すると、CLI によってそのコマンドが実行構成ファイルに追加されます。ほとんどの場合、**exit** または **end** コマンドを使用してコンフィギュレーション モードを終了する必要があります。

コマンドモードの概要

この表は、主なコマンド モードの概要を示しています。

Table 4: コマンドモードの概要

モード	アクセス方法	プロンプト	終了方法
EXEC	ログイン プロンプトから、ユーザ名とパスワードを入力します。	`switch#`	終了してログインプロンプトに戻るには、**exit** コマンドを使用します。
グローバル コンフィギュレーション	EXEC モードで、**configure terminal** コマンドを使用します。	`switch(config)#`	終了して EXEC モードに戻るには、**end** または **exit** コマンドを使用するか、**Ctrl-Z** を押します。
インターフェイス コンフィギュレーション	グローバル 構成モードで、**interface** コマンドを使用してインターフェイスを指定します。	`switch(config-if)#`	終了してグローバル構成モードに戻るには、**exit** コマンドを使用します。 終了して EXEC モードに戻るには、**exit** コマンドを使用するか、**Ctrl+Z** を押します。
サブインターフェイス コンフィギュレーション	グローバル 構成モードで、**interface** コマンドを使用してインターフェイスを指定します。	`switch(config-subif)#`	終了してグローバル構成モードに戻るには、**exit** コマンドを使用します。 終了して EXEC モードに戻るには、**end** コマンドを使用するか、**Ctrl-Z** を押します。

モード	アクセス方法	プロンプト	終了方法
VRF コンフィギュレーション	グローバル構成モードで、**vrf** コマンドを使用して、ルーティング プロトコルを指定します。	`switch(config-vrf)#`	終了してグローバル構成モードに戻るには、**exit** コマンドを使用します。 終了してEXECモードに戻るには、**end** コマンドを使用するか、**Ctrl-Z** を押します。
デフォルト以外の VRF に対する EXEC	EXEC モードで、**routing-context vrf** コマンドを使用して、VRF を指定します。	`switch-red#`	終了してデフォルトの VRF に戻るには、**routing-context vrf default** コマンドを使用します。

特殊文字

次の表に、Cisco NX-OS のテキスト文字列で特別な意味を持つ文字を示します。正規表現あるいはその他の特有なコンテキストでのみ使用します。

Table 5: **特殊文字**

文字	説明
%	パーセント
#	ポンド、ハッシュ、または番号
...	省略符号
|	縦線
< >	より小さい、またはより大きい
[ ]	角カッコ
{ }	波カッコ

キーストローク ショートカット

次の表に、EXEC モードおよびコンフィギュレーション モードの両方で使用されるコマンド キーの組み合わせを示します。

Table 6: キーストローク ショートカット

キーストローク	説明
Ctrl+A	カーソルを行の先頭に移動します。
Ctrl+B	カーソルを 1 文字左に移動します。複数行にわたってコマンドを入力するときは、**Left Arrow** または **Ctrl-B** キーを繰り返し押してシステム プロンプトまでスクロール バックして、コマンド エントリの先頭まで移動できます。あるいは **Ctrl-A** キーを押してコマンド エントリの先頭に移動します。
Ctrl+C	コマンドを取り消して、コマンド プロンプトに戻ります。
Ctrl+D	カーソル位置にある文字を削除します。
Ctrl+E	カーソルを行の末尾に移動します。
Ctrl+F	カーソルを 1 文字右に移動します。
Ctrl+G	コマンド ストリングを削除せずに、コマンド モードを終了して以前のコマンド モードに戻ります。
Ctrl+K	カーソル位置からコマンドラインの末尾までのすべての文字を削除します。
Ctrl+L	現在のコマンドラインを再表示します。
Ctrl+N	コマンド履歴の次のコマンドを表示します。
Ctrl+O	端末画面をクリアします。
Ctrl+P	コマンド履歴の前のコマンドを表示します。
Ctrl+R	現在のコマンドラインを再表示します。
Ctrl+T	カーソルの下の文字を、カーソルの右にある文字と置き換えます。カーソルが 1 文字右に移動します。
Ctrl+U	カーソル位置からコマンドラインの先頭までのすべての文字を削除します。
Ctrl+V	次のキーストロークに関する特別な意味を削除します。たとえば、正規表現で疑問符（?）を入力する前に、**Ctrl-V** を押します。
Ctrl+W	カーソルの左にある単語を削除します。

キーストローク	説明
Ctrl+X、H	入力したコマンドの履歴を表示します。 このキーの組み合わせを使用するときは、**Ctrl** キーと **X** キーを同時に押してリリースしてから、**H** を押します。
Ctrl+Y	バッファ内の最新のエントリを呼び出します（キーを同時に押します）。
Ctrl+Z	コンフィギュレーションセッションを終了して、EXEC モードに戻ります。 有効なコマンドを入力してから、コマンドラインの最後で Ctrl+Z を使用すると、コマンドの結果の設定がまず実行コンフィギュレーションファイルに追加されます。
上矢印キー	コマンド履歴の前のコマンドを表示します。
下矢印キー	コマンド履歴の次のコマンドを表示します。
右矢印キー 左矢印キー	コマンドストリング上でカーソルを前後に移動して、現在のコマンドを編集します。
?	使用可能なコマンドのリストを表示します。

キーストローク	説明

タブ	ワードの最初の文字を入力して **Tab** キーを押すと、ワードが補完されます。文字に一致するすべてのオプションが表示されます。
	タブを使用して、次の項目を入力します。
	• コマンド名
	• ファイル システム内のスキーム名
	• ファイル システム内のサーバ名
	• ファイル システム内のファイル名
	例：
	`switch(config)# xm<Tab>`
	`switch(config)# xml<Tab>`
	`switch(config)# xml server`

	例：
	`switch(config)# c<Tab>`
	`callhome  class-map  clock`
	`cdp  cli  control-plane`
	`switch(config)# cl<Tab>`
	`class-map  cli  clock`
	`switch(config)# cla<Tab>`
	`switch(config)# class-map`

	例：
	`switch# cd bootflash:<Tab>`
	`bootflash:///`
	`bootflash://sup-1/`
	`bootflash://sup-active/`
	`bootflash://sup-local/`
	`bootflash://module-27/`
	`bootflash://module-28/`

	例：
	`switch# cd bootflash://mo<Tab>`
	`bootflash://module-27/ bootflash://module-28/`
	`switch# cd bootflash://module-2`
	Note
	cd コマンドを使用してリモート マシンにアクセスできません。スロット 27 で、**cd bootflash://module-28** コマンドを入力すると、「Changing directory to a non-local server is not allowed」というメッセージが表示されます。

コマンドの短縮形

コマンドの最初の数文字を入力することで、コマンドおよびキーワードを省略できます。省略形には、コマンドまたはキーワードを一意に識別でき得る文字数を含める必要があります。コマンドの入力で問題が生じた場合は、システム プロンプトを確認し、疑問符（?）を入力して使用できるコマンドのリストを表示してください。コマンドモードが間違っているか、間違った構文を使用している可能性があります。

次の表に、コマンド省略形の例を示します。

Table 7: **コマンド省略形の例**

コマンド	省略形
configure terminal	**conf t**
copy running-config startup-config	**copy run start**
interface ethernet 1/2	**int e 1/2**
show running-config	**sh run**

部分的なコマンド名の補完

完全なコマンド名を思い出せない場合や、入力の作業量を減らす場合は、コマンドの先頭の数文字を入力して、**Tab** キーを押します。コマンドラインパーサーは、入力されたストリングがコマンド モードで一意である場合に、コマンドを補完します。キーボードに **Tab** キーがない場合は、代わりに **Ctrl-I** キーを押します。

コマンドは、コマンドが一意になるのに十分な文字が入力されていれば、CLI によって認識されます。たとえば、EXEC モードで **conf** と入力すると、CLI はエントリを **configure** コマンドと関連づけることができます。**configure** コマンドのみが **conf** で始まるためです。

次の例では、**Tab** キーを押したときに、CLI によって EXEC モードで **conf** の一意の文字列が認識されます。

```
switch# conf<Tab>
switch# configure
```

コマンド補完機能を使用すると、CLI により完全なコマンド名が表示されます。コマンドは、**Return** キーまたは **Enter** キーを押すまで、CLI によって実行されません。これにより、完全なコマンドが省略形によって意図したものでない場合に、コマンドを修正できます。入力した一連の文字に対して、対応するコマンドが複数ある場合は、一致するコマンドのリストが表示されます。

たとえば、**co<Tab>** と入力すると、EXEC モードで利用可能な、「**co**」で始まるすべてのコマンドがリストされます。

```
switch# co<Tab>
configure   copy
switch# co
```

コマンドエントリを補完できるよう、入力した文字は再びプロンプトに表示されることに注意してください。

コマンド階層での場所の特定

一部の機能では、コンフィギュレーション サブモード階層が 1 つのレベル以上ネストされます。この場合は、Present Working Context（PWC）に関する情報を表示できます。

SUMMARY STEPS

1. **where detail**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**where detail**
	Example:
	`switch# configure terminal`
	`switch(config)# interface mgmt0`
	`switch(config-if)# where detail`
	`mode:  conf`
	`  interface mgmt0`
	`  username:  admin`
	`  routing-context vrf: default`
		PWC を表示します。

コマンドの no 形式の使用方法

大部分の構成コマンドには **no** 形式があり、これを使用して、機能を無効化したり、デフォルト値に戻したり、設定を削除したりできます。

次に、機能をディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# feature tacacs+
switch(config)# no feature tacacs+
```

次に、機能をデフォルト値に戻す例を示します。

```
switch# configure terminal
switch(config)# banner motd #Welcome to the switch#
```

```
switch(config)# show banner motd
Welcome to the switch

switch(config)# no banner motd
switch(config)# show banner motd
User Access Verification
```

次に、機能の設定を削除する例を示します。

```
switch# configure terminal
switch(config)# radius-server host 10.10.2.2
switch(config)# show radius-server
retransmission count:0
timeout value:1
deadtime value:1
total number of servers:1

following RADIUS servers are configured:
        10.10.1.1:
                available for authentication on port:1812
                available for accounting on port:1813
        10.10.2.2:
                available for authentication on port:1812
                available for accounting on port:1813

switch(config)# no radius-server host 10.10.2.2
switch(config)# show radius-server
retransmission count:0
timeout value:1
deadtime value:1
total number of servers:1

following RADIUS servers are configured:
        10.10.1.1:
                available for authentication on port:1812
                available for accounting on port:1813
```

次に、EXEC モードでコマンドの **no** 形式を使用する例を示します。

```
switch# cli var name testinterface ethernet1/2
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2013-05-12-13.43.13"
testinterface="ethernet1/2"

switch# cli no var name testinterface
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2013-05-12-13.43.13"
```

CLI 変数の設定

ここでは、Cisco NX-OS CLI での CLI 変数について説明します。

CLI 変数について

Cisco NX-OS ソフトウェアは、CLI コマンドでの変数の定義および使用をサポートします。

CLI 変数は、次の方法で参照できます。

- コマンドラインで直接入力する。
- **run-script** コマンドを使用して開始されたスクリプトに渡す。親シェルで定義した変数は子 **run-script** コマンド プロセスで使用できます。

CLI 変数には、次の特性があります。

- 入れ子状態の参照を使用して、別の変数から変数を参照することはできません。
- スイッチのリロード時に維持することも、現在のセッションのみに使用することもできます。

Cisco NX-OS は、TIMESTAMP という事前定義の変数をサポートします。この変数は、コマンドを実行するときの YYYY-MM-DD-HH.MM.SS フォーマットの現在時刻を参照します。

Note TIIMESTAMP 変数名は大文字と小文字を区別します。文字はすべて大文字です。

CLI セッション限定の変数の設定

CLI セッションの間だけ有効な CLI セッション変数を定義できます。これらの変数は定期的に実行するスクリプトに役立ちます。丸括弧で名前を囲み、変数の前にドル記号（$）を付けることによって、変数を参照できます。たとえば、$(*variable-name*) です。

SUMMARY STEPS

1. **cli var name** *variable-name variable-text*
2. (Optional) **show cli variables**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**cli var name** *variable-name variable-text* **Example:** `switch# cli var name testinterface ethernet 2/1`	CLI セッション変数を設定します。*variable-name* 引数は、31 文字以内の英数字で指定します。大文字と小文字が区別されます。*variable-text* 引数は、200 文字以内の英数字で指定します。大文字と小文字が区別されます。スペースを含めることができます。 **Note**

	Command or Action	Purpose
		Cisco NX-OS リリース 7.0(3)I4(1) 以降、変数にはハイフン（-）とアンダースコア（_）を含めることができます。
ステップ 2	(Optional) **show cli variables** **Example:** `switch# show cli variables`	CLI 変数の設定を表示します。

固定 CLI 変数の設定

CLI セッションの終了後やデバイスのリロード後に保持される CLI 変数を設定できます。

SUMMARY STEPS

1. **configure terminal**
2. **cli var name** *variable-name variable-text*
3. **exit**
4. (Optional) **show cli variables**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**cli var name** *variable-name variable-text* **Example:** `switch(config)# cli var name testinterface ethernet 2/1`	CLI 固定変数を設定します。変数名は、英数字ストリングで指定します。大文字と小文字が区別されます。変数名の先頭を英字にする必要があります。31 文字以内で指定します。 **Note** Cisco NX-OS リリース 7.0(3)I4(1) 以降、変数にはハイフン（-）とアンダースコア（_）を含めることができます。

ステップ 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

	Command or Action	Purpose
ステップ 4	(Optional) **show cli variables** **Example:** `switch# show cli variables`	CLI 変数の設定を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

コマンド エイリアス

ここでは、コマンド エイリアスについて説明します。

コマンド エイリアスについて

コマンドエイリアスを定義して、使用頻度の高いコマンドを置き換えることができます。コマンド エイリアスは、コマンド構文の全体または一部を表すことができます。

コマンド エイリアスには、次の特性があります。

- コマンド エイリアスはすべてのユーザ セッションに対してグローバルです。
- コマンド エイリアスは、スタートアップ コンフィギュレーションに保存しておけば、再起動後も維持されます。
- コマンド エイリアス変換は常にすべてのコンフィギュレーション モードまたはサブモードのすべてのキーワードの中で最優先されます。
- コマンド エイリアスの設定は他のユーザ セッションに対してただちに有効になります。
- Cisco NX-OS ソフトウェアには、デフォルトのエイリアス **alias** が用意されています。このエイリアスは、**show cli alias** コマンドと同等であり、ユーザー定義のエイリアスをすべて表示します。
- デフォルトのコマンド エイリアス **alias** は、削除することも変更することもできません。
- エイリアスは最大深度 1 までネストできます。1 つのコマンドエイリアスは、有効なコマンドを参照する必要がある別のコマンドエイリアスを参照できますが、その他のコマンドエイリアスは参照できません。
- コマンド エイリアスは必ず、コマンドラインの最初のコマンド キーワードを置き換えます。
- 任意のコマンド モードでコマンドのコマンド エイリアスを定義できます。
- コマンド エイリアス内で CLI 変数を参照すると、変数参照ではなくその変数の現在の値がエイリアス内で使用されます。

- コマンド エイリアスは **show** コマンドの検索およびフィルタリングに使用できます。

コマンド エイリアスの定義

よく使用するコマンドにはコマンド エイリアスを定義できます。

SUMMARY STEPS

1. **configure terminal**
2. **cli alias name** *alias-name alias-text*
3. **exit**
4. (Optional) **alias**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**cli alias name** *alias-name alias-text* **Example:** `switch(config)# cli alias name ethint interface ethernet`	コマンドエイリアスを設定します。エイリアス名は英数字で表します。大文字と小文字は区別されません。先頭は英字にする必要があります。30 文字以内で指定します。

ステップ 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	(Optional) **alias** **Example:** `switch# alias`	コマンド エイリアス設定を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ユーザ セッション用のコマンド エイリアスの設定

Cisco NX-OS デバイス上の他のすべてのユーザーが使用できない、現在のユーザー セッション用のコマンド エイリアスを作成できます。また、コマンド エイリアスを保存し、現在のユーザ アカウントであとで使用することもできます。

手順の概要

1. **terminal alias** [**persist**] *alias-name command-string*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**terminal alias** [**persist**] *alias-name command-string* 例： `switch# terminal alias shintbr show interface brief`	現在のユーザ セッション用のコマンド エイリアスを設定します。ユーザーアカウントであとで使用するようにエイリアスを保存するには、**persist** キーワードを使用します。 （注） **persist** キーワードは短縮しないでください。

コマンド スクリプト

ここでは、複数のタスクを実行するためにコマンドのスクリプトを作成する方法について説明します。

コマンド スクリプトの実行

ファイルでコマンドのリストを作成し、CLI からこれらのコマンドを実行できます。コマンド スクリプトでは CLI 変数を使用できます。

Note CLI プロンプトではスクリプト ファイルを作成できません。スクリプト ファイルをリモート デバイスで作成して、Cisco NX-OS デバイスの bootflash: または volatile: ディレクトリにコピーすることができます。

SUMMARY STEPS

1. **run-script** [**bootflash:** | **volatile:**] *filename*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**run-script** [**bootflash:** | **volatile:**] *filename* **Example:** `switch# run-script testfile`	デフォルトディレクトリのファイル内のコマンドを実行します。

端末への情報のエコー

端末に情報をエコーできます。これは、コマンドスクリプトで特に役立ちます。CLI変数を参照し、エコーされるテキストでフォーマット オプションを使用できます。

次の表に、テキストに挿入できるフォーマット オプションを示します。

***Table 8: echo* コマンドのフォーマット オプション**

フォーマットオプション	説明
\b	バック スペースを挿入します。
\c	テキストストリングの最後にある改行文字が削除されます。
\f	フォーム フィード文字が挿入されます。
\n	改行文字が挿入されます。
\r	テキスト行の最初に戻ります。
\t	水平タブ文字が挿入されます。
\v	垂直タブ文字が挿入されます。
\\	バックスラッシュ文字が表示されます。
nnn	対応する ASCII 8 進文字が表示されます。

SUMMARY STEPS

1. **echo** [**backslash-interpret**] [*text*]

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**echo** [**backslash-interpret**] [*text*]
	Example:
	`switch# echo This is a test.`
	`This is a test.`
		この項で説明している **backslash-interpret** キーワードは、テキスト文字列にフォーマット オプションが含まれることを示します。*text* 引数は、英数字で指定します。大文字と小文字が区別されます。空白を含めることができます。200 文字以内で指定します。デフォルトは空白行です。

コマンド処理の遅延

コマンド アクションを一定の時間、遅延できます。これは、コマンド スクリプト内で特に役に立ちます。

SUMMARY STEPS

1. **sleep** *seconds*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**sleep** *seconds* **Example:** `switch# sleep 30`	遅延をある秒数発生させます。値の範囲は 0 ～ 2147483647 です。

状況依存ヘルプ

Cisco NX-OS ソフトウェアは、CLI で状況依存ヘルプを提供します。コマンド内の任意の位置に疑問符（?）を使用することで、有効な入力オプションを表示できます。

CLI では、入力エラーを特定するためにキャレット（^）記号が使用されます。^記号は、コマンド ストリング内の誤ったコマンド、キーワード、または引数が入力された位置に表示されます。

次の表に、状況依存ヘルプの出力例を示します。

Table 9: **状況依存ヘルプの例**

出力例	説明
switch# **clock ?** set HH:MM:SS Current Time switch# clock	EXEC モードでの **clock** コマンドのコマンド構文を表示します。 スイッチの出力は、 **set clock** を使用するにはキーワードが必要です を参照してください。
switch# clock **set ?** WORD HH:MM:SS Current Time switch# clock set	時刻を設定するためのコマンド構文を表示します。 ヘルプ出力は、クロックの設定に現在時刻が必要であること、および時刻のフォーマット方法を示しています。
switch# clock set **13:32:00<CR>** % Incomplete command switch#	現在時刻を追加します。 CLI は、コマンドが不完全であることを示しています。
switch# **<Ctrl-P>** switch# clock set 13:32:00	入力した直前のコマンドを表示します。
switch# clock set 13:32:00 **?** <1-31> Day of the month switch# clock set 13:32:00	**clock set** に対する追加の引数を表示しますを参照してください。

switch# clock set 13:32:00 18 **?**
  April Month of the year
  August Month of the year
  December Month of the year
  February Month of the year
  January Month of the year
  July Month of the year
  June Month of the year
  March Month of the year
  May Month of the year
  November Month of the year
  October Month of the year
  September Month of the year
switch# clock set 13:32:00 18
	clock set コマンドに対する追加の引数を表示します。

switch# clock set 13:32:00 18 **April 13<CR>** % Invalid input detected at '^' marker.	クロック設定に日付を追加します。 CLI は、13 の位置にキャレット記号（^）を使用してエラーを示しています。
switch# clock set 13:32:00 18 April **?** <2000-2030> Enter the year (no abbreviation) switch# clock set 13:32:00 18 April	この年に対応する適切な引数を表示します。

出力例	説明
switch# clock set 13:32:00 18 April **2013<CR>** switch#	**clock set** コマンドの正しい構文を入力します。

正規表現について

Cisco NX-OS ソフトウェアは、CLI 出力（**show** コマンドなど）の検索またはフィルタリングで正規表現をサポートしています。コマンドにも表示されません。正規表現では大文字と小文字が区別され、また複雑な一致要件を設定することができます。

特殊文字

他のキーボード文字（! や ~ など）を、単一文字パターンとして使用することもできますが、特定のキーボード文字は、正規表現内で使用した場合特別な意味を持ちます。

次の表に、特別な意味を持つキーボード文字を示します。

Table 10: 特別な意味を持つ特殊文字

文字	特別な意味
.	スペースを含む任意の単一文字と一致します。
*	0 個以上のパターンのシーケンスと一致します。
+	1 個以上のパターンのシーケンスと一致します。
?	0 または 1 回のパターンと一致します。
^	ストリングの先頭と一致します。
$	ストリングの末尾と一致します。
_（アンダースコア）	カンマ（,）、左波カッコ（{）、右波カッコ（}）、左カッコ（(）、右カッコ（)）、ストリングの先頭、ストリングの末尾、またはスペースと一致します。 **Note** アンダースコアは、BGP 関連のコマンドの場合にのみ正規表現として扱われます。

これらの特殊文字を単一文字パターンとして使用するときは、各文字の前にバックスラッシュ（\）を置いて特別な意味を除外してください。次の例には、ドル記号（$）、アンダースコア（_）、およびプラス記号（+）にそれぞれ一致する単一文字パターンが含まれています。

```
\$ \_ \+
```

複数文字のパターン

文字、数字、または特別な意味を持たないキーボード文字を連結して、複数文字のパターンを指定することもできます。たとえば、a4% は複数文字の正規表現です。

複数文字パターンでは、順序が大切です。**a4%** という正規表現は、a という文字のあとに 4 が続き、そのあとにパーセント記号（%）が続く文字と一致します。ストリングの中にa4% という文字がその順序で含まれていないと、パターンマッチングは失敗します。複数文字正規表現 **a.**（文字 a の後にピリオド）は、ピリオド文字の特別な意味を使用して、文字 a の後に任意の単一文字が続くストリングと一致します。この例では、ab、a!、またはa2 というストリングはすべてこの正規表現と一致します。

特殊文字の特別な意味は、特殊文字の前にバックスラッシュを挿入することで無効にできます。たとえば、表現**a\.** がコマンド構文で使用されている場合、文字列a. だけが一致します。

位置指定

特殊文字を使用してストリング内での正規表現の位置を指定することで、正規表現パターンをストリングの先頭または末尾と一致させることができます。

次の表に、位置指定に使用可能な特殊文字を示します。

Table 11: 位置指定に用いられる特殊文字

文字	説明
^	ストリングの先頭と一致します。
$	ストリングの末尾と一致します。

たとえば、正規表現 **^con** は **con** で始まるストリングに一致し、**$sole** は **sole** で終わるストリングに一致します。

Note ^記号は、角カッコで囲まれた範囲に論理関数「not」を指定する場合にも使用されます。たとえば、正規表現 **[^abcd]** は、a、b、c、またはd 以外の任意の単一文字に一致する範囲を示します。

show コマンド出力の検索とフィルタリング

多くの場合、 **show**コマンドからの出力は、長くて煩雑になります。Cisco NX-OS ソフトウェアでは、情報を簡単に見つけ出すために、出力の検索およびフィルタリングを行うことができます。検索およびフィルタリングのオプションは、 **show** コマンドの末尾にパイプ記号（|）を

付け、その後に指定します。これらのオプションは、CLI状況依存ヘルプ機能を使用して表示できます。

```
switch# show running-config | ?
  cut      Print selected parts of lines.
  diff     Show difference between current and previous invocation (creates temp files:
           remove them with 'diff-clean' command and don't use it on commands with big
           outputs, like 'show tech'!)
  egrep    Egrep - print lines matching a pattern
  grep     Grep - print lines matching a pattern
  head     Display first lines
  human    Output in human format
  last     Display last lines
  less     Filter for paging
  no-more  Turn-off pagination for command output
  perl     Use perl script to filter output
  section  Show lines that include the pattern as well as the subsequent lines that are
           more indented than matching line
  sed      Stream Editor
  sort     Stream Sorter
  sscp     Stream SCP (secure copy)
  tr       Translate, squeeze, and/or delete characters
  uniq     Discard all but one of successive identical lines
  vsh      The shell that understands cli command
  wc       Count words, lines, characters
  xml      Output in xml format (according to .xsd definitions)
  begin    Begin with the line that matches
  count    Count number of lines
  end      End with the line that matches
  exclude  Exclude lines that match
  include  Include lines that match
```

フィルタリングおよび検索のキーワード

Cisco NX-OS CLIには、**show**コマンドと併用してコマンド出力の検索やフィルタリングを実行できる、一連のキーワードが用意されています。

次の表に、CLI出力のフィルタリングや検索を行うためのキーワードを示します。

Table 12: フィルタリングおよび検索のキーワード

キーワードの構文	説明
begin *string* 例： **show version | begin Hardware**	検索ストリングと一致するテキストが含まれている行から表示を開始します。検索ストリングは大文字と小文字が区別されます。
count 例： **show running-config | count**	コマンド出力の行数を表示します。

キーワードの構文	説明
cut [**-d** *character*] {**-b** | **-c** | **-f** | **-s**} 例： `show file testoutput | cut -b 1-10`	一部の出力行のみを表示します。一定のバイト数（**-b**）、文字数（**-vcut** [**-d** *character*] {**-b** | **-c**|**-f**|**-s**}）、またはフィールド数（**-f**）を表示できます。また、**-d** キーワードを使用して、デフォルトのタグ文字以外のフィールド デリミタを定義することもできます。**-s** キーワードは、デリミタが含まれない行の表示を抑制します。
end *string* 例： `show running-config | end interface`	検索ストリングの最後の一致になるまですべての行を表示します。
exclude *string* 例： `show interface brief | exclude down`	検索ストリングが含まれていない行をすべて表示します。検索ストリングは大文字と小文字が区別されます。
head [**lines** *lines*] 例： `show logging logfile | head lines 50`	出力の先頭を指定の行数だけ表示します。デフォルトの行数は 10 です。
human 例： `show version | human`	**terminal output xml** コマンドを使用して出力形式がXMLに設定されている場合に、出力を通常形式で表示します。 を参照してください。
include *string* 例： `show interface brief | include up`	検索ストリングが含まれている行をすべて表示します。検索ストリングは大文字と小文字が区別されます。
last [*lines*] 例： `show logging logfile | last 50`	出力の末尾を指定の行数だけ表示します。デフォルトの行数は 10 です。
no-more 例： `show interface brief | no-more`	途中で停止せずにすべての出力を表示します。画面の一番下に「--More--」プロンプトは表示されなくなります。
sscp *SSH-connection-name filename* 例： `show version | sscp MyConnection show_version_output`	Streaming Secure Copy（sscp）を使用して出力を名前付き SSH 接続にリダイレクトします。名前付きの SSH 接続は、**ssh name** コマンドを使用して作成できます。 を参照してください。

キーワードの構文	説明
wc[**bytes** |**lines** |**words**] 例： `show file testoutput | wc bytes`	文字数、行数、または単語数を表示します。デフォルトでは、行数、単語数、および文字数を表示します。
xml 例： `show version | xml`	出力を XML 形式で表示します。

diff ユーティリティ

show からの コマンドからの出力と、そのコマンドを以前に実行したときの出力を比較できます。

diff-clean [**all-sessions**] [**all-users**]

次の表で、diff ユーティリティのキーワードについて説明します。

キーワード	説明
all-sessions	現在のユーザーのすべてのセッション（過去および現在のセッション）から比較の一時ファイルが削除されます。
all-users	すべてのユーザーのすべてのセッション（過去および現在のセッション）から比較の一時ファイルが削除されます。

Cisco NX-OS ソフトウェアは、現在および以前のすべてのユーザー セッションに対する **show** コマンドの最新の出力について、一時ファイルを作成します。これらの一時ファイルを削除するには、**diff-clean** コマンドを使用します。

diff-clean [**all-sessions** | **all-users**]

デフォルトでは、**diff-clean** コマンドによって現在のユーザーのアクティブ セッションに対する一時ファイルが削除されます。**all-sessions** キーワードを指定すると、現在のユーザーの過去および現在の全セッションに対する一時ファイルが削除されます。**all-users** キーワードを指定すると、すべてのユーザーの過去および現在の全セッションに対する一時ファイルが削除されます。

grep および egrep ユーティリティ

Global Regular Expression Print（grep）および Extended grep（egrep）コマンドライン ユーティリティを使用して、**show** コマンド出力をフィルタリングすることができます。

grep と egrep の構文は次のとおりです。

{**grep** | **egrep**} [**count**] [**ignore-case**] [**invert-match**] [**line-exp**] [**line-number**] [**next** *lines*] [**prev** *lines*] [**word-exp**] *expression*}]

次の表に、**grep** と **egrep** のパラメータを示します。

***Table 13: grep* および *egrep* のパラメータ**

パラメータ	説明
count	一致した行の合計数のみを表示します。
ignore-case	一致した行の大文字と小文字の相違を無視するように指定します。
invert-match	表現が一致しない行を表示します。
line-exp	行に完全に一致する行だけを表示します。
line-number	一致した各行の前の行番号を表示するように指定します。
next *lines*	一致した行の後に表示する行数を指定します。デフォルトは 0 です。有効な範囲は 1 ～ 999 です。
prev *lines*	一致した行の前に表示する行数を指定します。デフォルトは 0 です。有効な範囲は 1 ～ 999 です。
word-exp	単語が完全に一致する行だけを表示します。
式	出力を検索するための正規表現を指定します。

less ユーティリティ

less ユーティリティを使用して、**show** コマンド出力の内容を 1 画面ずつ表示できます。「:」プロンプトにおいて **less** コマンドを入力できます。すべての **less** 表示するには使用できるコマンドを表示するには、: プロンプトで **h** を入力します。

Mini AWK ユーティリティ

AWK はテキスト出力を要約するための単純ですが強力なユーティリティです。パイプ（|）の後にこのユーティリティを使用し、コマンドのテキスト出力をさらに処理できます。Cisco NX-OS は、埋め込みプログラムを引数として使用する mini AWK をサポートしています。

次に、mini AWK ユーティリティを使用して **show ip route summary vrf all** コマンドのテキスト出力を要約する例を示します。

```
switch# show ip route summary vrf all | grep "Total number of routes"
Total number of routes: 3
Total number of routes: 10

switch# show ip route summary vrf all | grep "Total number of routes" | awk '{ x = x +
$5} END { print x }'
13
```

sed ユーティリティ

ストリームエディタ（sed）ユーティリティを使用して、**show** コマンド出力のフィルタリングや操作を次のように実行できます。

sed *command*

command 引数には、sed ユーティリティのコマンドを含みます。

sort ユーティリティ

sort ユーティリティを使用して、**show** コマンド出力をフィルタリングできます。 コマンドの出力でスラブの使用状況を確認することで問題を特定できます。

sort ユーティリティの構文は次のとおりです。

sort [**-M**] [**-b**] [**-d**] [**-f**] [**-g**] [**-i**] [**-k** *field-number*[**.***char-position*][*ordering*]] [**-n**] [**-r**] [**-t** *delimiter*] [**-u**]

次の表に、sort ユーティリティのパラメータの説明を示します。

表 *14 : sort* ユーティリティのパラメータ

パラメータ	説明
-M	月でソートします。
-b	先頭のブランク（空白文字）を無視します。デフォルトのソートでは、先頭のブランクが考慮されます。
-d	ブランクと英数字のみを比較してソートします。デフォルトのソートでは、すべての文字が考慮されます。
-f	小文字を大文字として処理します。
-g	一般的な数値を比較してソートします。
-i	印刷可能な文字だけを使用してソートします。デフォルトのソートでは、印刷不可能な文字も考慮されます。
-k *field-number*[**.***char-position*][*ordering*]	キー値に従ってソートします。デフォルトのキー値はありません。
-n	数値ストリングの値に従ってソートします。
-r	ソート結果の順序を逆にします。デフォルトのソート出力は昇順です。

パラメータ	説明
-t *delimiter*	指定のデリミタを使用してソートします。デフォルトのデリミタは空白文字です。
-u	ソート結果から重複行を取り除きます。ソート出力では重複行が表示されます。

--More-- プロンプトからの検索およびフィルタリング

show コマンド出力の `--More--` プロンプトで出力の検索やフィルタリングを実行できます。コマンドの出力でスラブの使用状況を確認することで問題を特定できます。

次の表に、`--More-` プロンプト コマンドの説明を示します。

***Table 15: --More--* プロンプト コマンド**

コマンド	説明
[*lines*]<space>	指定した行数か現在の画面サイズ分の出力行を表示します。
[*lines*]**z**	指定した行数か現在の画面サイズ分の出力行を表示します。*lines* 引数を使用すると、その値が新しいデフォルト画面サイズになります。
[*lines*]<return>	指定した行数か現在のデフォルトの行数で出力行を表示します。初期のデフォルトは 1 行です。オプションの *lines* 引数を使用すると、その値がこのコマンドで表示する新しいデフォルトの行数になります。
[*lines*]**d** または [*lines*]Ctrl+shift+D	指定した行数か現在のデフォルトの行数で出力行をスクロールします。初期のデフォルトは 11 行です。オプションの *lines* 引数を使用すると、その値がこのコマンドで表示する新しいデフォルトの行数になります。
q または **Q** または Ctrl-C	`--More--` プロンプトを終了します。
[*lines*]**s**	指定した行数か現在のデフォルトの行数だけ出力をスキップし、1 画面分の出力行を表示します。デフォルトは 1 行です。
[*lines*]**f**	指定した画面数か現在のデフォルトの画面数だけ出力をスキップし、1 画面分の出力行を表示します。デフォルトは 1 画面です。
=	現在の行番号を表示します。
[*count*]/*expression*	正規表現に一致する行までスキップし、1 画面分の出力行を表示します。正規表現の複数回の繰り返しで行を検索する場合は、オプションの *count* 引数を使用します。このコマンドにより、他のコマンドで使用可能な現在の正規表現が設定されます。

コマンド	説明
[*count*]**n**	現在の正規表現に次に一致する行までスキップし、1 画面分の出力行を表示します。複数の一致をスキップする場合は、オプションの *count* 引数を使用します。
{**!** | **:!**[*shell-cmd*]}	*shell-cmd* 引数に指定したコマンドをサブシェルで実行します。
.	前のコマンドを繰り返します。

コマンド履歴の使用方法

Cisco NX-OS ソフトウェアの CLI では、現在のユーザー セッションのコマンド履歴にアクセスできます。コマンドを呼び出し、そのまま再実行できます。また、実行前に修正することも可能です。コマンド履歴はクリアすることもできます。

コマンドの呼び出し

コマンド履歴内のコマンドを呼び出して、必要に応じて修正し、再入力できます。

次に、コマンドを呼び出して再入力する例を示します。

```
switch(config)# show cli history
0  11:04:07   configure terminal
1  11:04:28   show interface ethernet 2/24
2  11:04:39     interface ethernet 2/24
3  11:05:13       no shutdown
4  11:05:19     exit
5  11:05:25   show cli history
switch(config)# !1
switch(config)# show interface ethernet 2/24
```

Ctrl+P と **Ctrl+N** のキーストローク ショートカットを使用してコマンドを呼び出すこともできます。

CLI 履歴の再呼び出しの制御

CLI 履歴から再呼び出しするコマンドを制御するには、 **Ctrl-P** および **Ctrl-N** キーストローク ショートカット キーストローク ショートカットを使用します。 Cisco NX-OSソフトウェアは、現在のコマンド モード以上のコマンド モードのすべてのコマンドを再呼び出しします。たとえば、グローバル コンフィギュレーション モードで作業をしている場合は、コマンド呼び出しキーストローク ショートカットを使用すると、EXEC モード コマンドとグローバル コンフィギュレーション モード コマンドの両方が呼び出されます。

CLI 編集モードの設定

Ctrl-P および **Ctrl-N** キーストローク ショートカットを使用して CLI 履歴からコマンドを呼び出し、コマンドを再発行する前に編集することができます。デフォルトの編集モードは、emacs です。編集モードを vi に変更できます。

手順の概要

1. [**no**] **terminal edit-mode vi** [**persist**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **terminal edit-mode vi** [**persist**] 例： `switch# terminal edit-mode vi`	ユーザ セッションの CLI 編集モードを vi に変更します。**persist** キーワードを使用すると、現在のユーザー名の設定がセッション間で保持されます。 emacs の使用に戻すには、**no** を使用します。

コマンド履歴の表示

show cli history コマンドを使用して、コマンド履歴を表示できます。.

show cli history コマンドの構文は次のとおりです。

show cli history [*lines*] [**config-mode** | **exec-mode** | **this-mode-only**] [**unformatted**]

デフォルトで表示される行数は 12 であり、出力にはコマンド番号とタイムスタンプが含まれます。

これは、コマンド履歴のデフォルトの行数を表示する例を示します。

```
switch# show cli history
```

これは、コマンド履歴の 20 行を表示する例を示します。

```
switch# show cli history 20
```

これは、コマンド履歴内の構成コマンドだけを表示する例を示します。

```
switch(config)# show cli history config-mode
```

これは、コマンド履歴内の EXEC コマンドだけを表示する例を示します。

```
switch(config)# show cli history exec-mode
```

これは、現在のコマンドモードに関するコマンド履歴内のコマンドだけを表示する例を示します。

```
switch(config-if)# show cli history this-mode-only
```

これは、コマンド番号とタイムスタンプなしでコマンド履歴内のコマンドだけを表示する例を示します。

```
switch(config)# show cli history unformatted
```

CLI 確認プロンプトのイネーブルまたはディセーブル

多くの機能について、作業を続行する前に確認を求めるプロンプトが、Cisco NX-OS ソフトウェアによってCLIに表示されます。これらのプロンプトをイネーブルにしたり、ディセーブルにしたりできます。デフォルトではイネーブルになっています。

手順の概要

1. [**no**] **terminal dont-ask** [**persist**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **terminal dont-ask** [**persist**] 例： `switch# terminal dont-ask`	CLI 確認プロンプトをディセーブルにします。この項で説明している **persist** キーワードを使用すると、現在のユーザー名の設定がセッション間で保持されます。デフォルトではイネーブルになっています。 CLI 確認プロンプトをイネーブルにするには、コマンドの **no** 形式を使用します。

CLI 画面の色の設定

表示する CLI の色は次のように変更できます。

- 直前のコマンドが成功した場合は、プロンプトが緑色で表示されます。
- 直前のコマンドが失敗した場合は、プロンプトが赤色で表示されます。
- ユーザ入力は青色で表示されます。

- コマンド出力はデフォルトの色で表示されます。

デフォルトの色は、端末エミュレータ ソフトウェアにより送信されます。

手順の概要

1. **terminal color** [**evening**] [**persist**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**terminal color** [**evening**] [**persist**] 例： `switch# terminal color`	ターミナルセッションに対してCLI画面の色を設定します。**evening** キーワードはサポートされていません。**persist** キーワードを使用すると、現在のユーザー名の設定がセッション間で保持されます。デフォルト設定は保持されません。

モジュールへのコマンドの送信

slot コマンドを使用して、スーパーバイザ モジュール セッションからモジュールにコマンドを直接送信できます。

slot の構文は次のとおりです。

slot *slot-number* [**quoted**] *command-string*

デフォルトでは、*command-string* 引数内のキーワードと引数はスペースで区切られます。モジュールに複数のコマンドを送信するには、スペース文字、セミコロン（;）、スペース文字でコマンドを区切ります。

quoted キーワードは、コマンド ストリングの先頭と末尾に二重引用符（"）が使用されることを示します。スーパーバイザ モジュール セッションでだけサポートされているdiffなどのフィルタリング ユーティリティにモジュール コマンド出力をリダイレクトする場合は、このキーワードを使用します。

次に、モジュール情報を表示したり、フィルタリングしたりする例を示します。

```
switch# slot 27 show version | grep lc
```

次に、スーパーバイザ モジュール セッションに関するモジュール情報をフィルタリングする例を示します。

```
switch# slot 27 quoted "show version" | diff
switch# slot 28 quoted "show version" | diff -c
*** /volatile/vsh_diff_1_root_8430_slot__quoted_show_version.old        Wed Apr 29
20:10:41 2013
```

```
--- -   Wed Apr 29 20:10:41 2013
***************
*** 1,5 ****
! RAM 1036860 kB
! lc27
  Software
    BIOS:      version 6.20
    system:    version 6.1(2)I1(1) [build 6.1(2)]
--- 1,5 ----
! RAM 516692 kB
! lc28
  Software
    BIOS:      version 6.20
    system:    version 6.1(2)I1(1) [build 6.1(2)]
***************
*** 12,16 ****
  Hardware
      bootflash: 0 blocks (block size 512b)

!    uptime is 0 days 1 hours 45 minute(s) 34 second(s)

--- 12,16 ----
  Hardware
      bootflash: 0 blocks (block size 512b)

!    uptime is 0 days 1 hours 45 minute(s) 42 second(s)
```

電子メールによるコマンド出力の送信

CLI を使用して **show** コマンドの出力を電子メール アドレスに送信することができます。これには、パイプ演算子（|）を使用します。

（注） 電子メールの設定は再設定するまで、すべての **show** コマンドで持続されます。

Cisco NX-OS リリース 9.3(3) より前のリリースから Cisco NX-OS リリース 9.3(3) 以降のリリースにアップグレードすると、電子メール構成が失われます。これは、この機能の DME 機能を有効にするためです。これを解決するには、「メールなし」を実行して、メール構成全体を再適用する必要があります。

手順の概要

1. **configure terminal**
2. **email**
3. **smtp-host** *ip-address* **smtp-port** *port*
4. **vrf management**
5. **from** *email-address*
6. **reply-to** *email-address*
7. **exit**
8. **exit**
9. **show email**

10. *show-command* | **email subject** *subject email-address*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**email**
	例：
	`switch(config)# email`
	`switch(config-email)#`
		電子メール構成モードを開始します。

ステップ 3	**smtp-host** *ip-address* **smtp-port** *port* 例： `switch(config-email)# smtp-host 198.51.100.1 smtp-port 25`	SMTP ホスト IP アドレスおよび SMTP ポート番号を指定します。
ステップ 4	**vrf management** 例： `switch(config-email)# vrf management`	電子メール転送用の VRF を指定します。
ステップ 5	**from** *email-address* 例： `switch(config-email)# from admin@Mycompany.com`	送信者の電子メール アドレスを指定します。
ステップ 6	**reply-to** *email-address* 例： `switch(config-email)# reply-to admin@Mycompany.com`	受信者の電子メール アドレスを指定します。

ステップ 7	**exit**
	例：
	`switch(config-email)# exit`
	`switch(config)#`
		電子メール構成モードを終了します。

ステップ 8	**exit**
	例：
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 9	**show email** 例： `switch# show email`	電子メールの構成を表示します。

ステップ 10	*show-command* **| email subject** *subject email-address*
	例：
	`switch# show interface brief | email subject show-interface admin@Mycompany.com`
	`Email sent`
		パイプ演算子（|）を使用し、指定した **show** コマンドの出力を件名付きで電子メール アドレスに送信します。

BIOS ローダー プロンプト

スーパーバイザ モジュールの起動時に、特殊な BIOS イメージが、システム起動用の有効な nx-os イメージを自動的にロードしたり、検索しようとしたりします。有効な nx-os イメージが見つからない場合は、次の BIOS ローダー プロンプトが表示されます。

```
loader>
```

Cisco NX-OS ソフトウェアを `loader>` プロンプトからロードする方法については、『*Cisco Nexus 9000 Troubleshooting Guide*』を参照してください。

CLI の使用例

ここでは、CLI の使用例を示します。

システム定義のタイムスタンプ変数の使用方法

次の例では、**show** コマンド出力をファイルにリダイレクトするときに $(TIMESTAMP) を使用します。

```
switch# show running-config > rcfg.$(TIMESTAMP)
Preparing to copy....done
switch# dir
       12667     May 01 12:27:59 2013  rcfg.2013-05-01-12.27.59

Usage for bootflash://sup-local
8192 bytes used
20963328 bytes free
20971520 bytes total
```

CLI セッション変数の使用方法

$(*variable-name***)** 構文を使用して変数を参照できます。

次に、ユーザ定義の CLI セッション変数を参照する例を示します。

```
switch# show interface $(testinterface)
Ethernet2/1 is down (Administratively down)
  Hardware is 10/100/1000 Ethernet, address is 0000.0000.0000 (bia 0019.076c.4dac)
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  auto-duplex, auto-speed
  Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned on
  Switchport monitor is off
  Last clearing of "show interface" counters never
  5 minute input rate 0 bytes/sec, 0 packets/sec
  5 minute output rate 0 bytes/sec, 0 packets/sec
  L3 in Switched:
    ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
  L3 out Switched:
    ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
  Rx
    0 input packets 0 unicast packets 0 multicast packets
    0 broadcast packets 0 jumbo packets 0 storm suppression packets
    0 bytes
  Tx
    0 output packets 0 multicast packets
    0 broadcast packets 0 jumbo packets
    0 bytes
    0 input error 0 short frame 0 watchdog
    0 no buffer 0 runt 0 CRC 0 ecc
    0 overrun  0 underrun 0 ignored 0 bad etype drop
    0 bad proto drop 0 if down drop 0 input with dribble
    0 input discard
    0 output error 0 collision 0 deferred
    0 late collision 0 lost carrier 0 no carrier
    0 babble
    0 Rx pause 0 Tx pause 0 reset
```

コマンド エイリアスの定義

次に、コマンド エイリアスを定義する例を示します。

```
cli alias name ethint interface ethernet
cli alias name shintbr show interface brief
cli alias name shintupbr shintbr | include up | include ethernet
```

次に、コマンド エイリアスを使用する例を示します。

```
switch# configure terminal
switch(config)# ethint 2/3
switch(config-if)#
```

コマンド スクリプトの実行

次の例では、スクリプト ファイル内で指定されている CLI コマンドを表示します。

```
switch# show file testfile
configure terminal
interface ethernet 2/1
no shutdown
end
show interface ethernet 2/1
```

次の例では、**run-script** コマンドの実行時の出力を表示します。

```
switch# run-script testfile
`configure terminal`
`interface ethernet 2/1`
`no shutdown`
`end`
`show interface ethernet 2/1 `
Ethernet2/1 is down (Link not connected)
  Hardware is 10/100/1000 Ethernet, address is 0019.076c.4dac (bia 0019.076c.4dac)
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  Port mode is trunk
  auto-duplex, auto-speed
  Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned on
  Switchport monitor is off
  Last clearing of "show interface" counters 1d26.2uh
  5 minute input rate 0 bytes/sec, 0 packets/sec
  5 minute output rate 0 bytes/sec, 0 packets/sec
  Rx
    0 input packets 0 unicast packets 0 multicast packets
    0 broadcast packets 0 jumbo packets 0 storm suppression packets
    0 bytes
  Tx
    0 output packets 0 multicast packets
    0 broadcast packets 0 jumbo packets
    0 bytes
    0 input error 0 short frame 0 watchdog
    0 no buffer 0 runt 0 CRC 0 ecc
    0 overrun  0 underrun 0 ignored 0 bad etype drop
    0 bad proto drop 0 if down drop 0 input with dribble
    0 input discard
    0 output error 0 collision 0 deferred
    0 late collision 0 lost carrier 0 no carrier
    0 babble
    0 Rx pause 0 Tx pause 0 reset
```

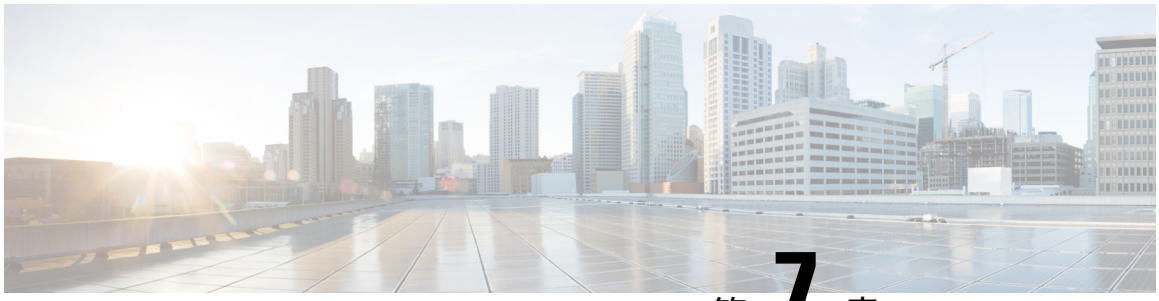
電子メールによるコマンド出力の送信

この例は、**show interface brief** の出力を送信する方法を示しています。 パイプ演算子 (|) を使用した電子メール アドレスへのコマンド

```
switch<config># email
switch(config-email)# smtp-host 198.51.100.1 smtp-port 25
switch(config-email)# vrf management
switch(config-email)# from admin@Mycompany.com
switch(config-email)# reply-to admin@Mycompany.com
switch(config-email)# exit
switch(config)# exit
switch# show email
SMTP host: 198.51.100.1
SMTP port: 25
Reply to: admin@Mycompany.com
From: admin@Mycompany.com
VRF: management
switch# show interface brief | email subject show-interface admin@Mycompany.com

Email sent
```

admin@Mycompany.com に「show-interface」という件名で送信される電子メールにコマンドの出力が表示されます。

```
<snip>
--------------------------------------------------------------------------------
Ethernet  VLAN Type Mode   Status Reason             Speed     Port
Interface                                                      Ch #
--------------------------------------------------------------------------------
Eth1/1     --  eth  trunk  up     none                10G (D)   --
Eth1/2     --  eth  routed down   Link not connected  auto(D)   --
Eth1/3     --  eth  routed up     none                10G (D)   --
Eth1/4     --  eth  routed down   Link not connected  auto (D)  --
Eth1/5     --  eth  routed down   Link not connected  auto (D)  --
Eth1/6     --  eth  routed down   Link not connected  auto (D)  --
Eth1/7     --  eth  routed down   Link not connected  auto (D)  --
Eth1/8     --  eth  routed down   Link not connected  auto (D)  --
Eth1/9     --  eth  routed down   Link not connected  auto (D)  --
Eth1/10    --  eth  routed down   Link not connected  auto (D)  --
<snip>
```



第 7 章

端末設定とセッションの設定

この章は、次の内容で構成されています。

端末設定とセッションの概要

ここでは、端末設定とセッションの概要について説明します。

ターミナル セッションの設定

Cisco NX-OS ソフトウェア機能では、端末の次の特性を管理できます。

端末タイプ
: リモート ホストと通信する際に Telnet で使用される名前

Length
: 一時停止する前に表示されるコマンド出力の行数

幅
: 行を折り返す前に表示される文字数

非アクティブ セッションのタイムアウト
: デバイスによって停止される前にセッションが非アクティブの状態でいられる分数

コンソール ポート

コンソール ポートは非同期のシリアル ポートで、初期設定用に、RJ-45 コネクタを使用して標準 RS-232 ポート経由でデバイスに接続できます。このポートに接続されるデバイスには、非同期伝送の機能が必要です。コンソール ポートには、次のパラメータを設定できます。

データ ビット
データに使用するビット数を 8 ビットのバイト単位で指定します。
非アクティブ セッションのタイムアウト
セッションが終了になるまでの非アクティブ時間を分単位で指定します。
パリティ
エラー検出用の奇数パリティまたは偶数パリティを指定します。
スピード
接続の送信速度を指定します。
ストップ ビット
非同期回線に対するストップ ビットを指定します。

ターミナル エミュレータは、9600 ボー、8 データ ビット、1 ストップ ビット、パリティなしに設定してください。

仮想端末

仮想端末回線を使用してデバイスに接続できます。セキュア シェル（SSH）および Telnet は、仮想ターミナル セッションを作成します。仮想端末の非アクティブ セッション タイムアウトおよびセッション数の上限を設定できます。

ファイル システム パラメータのデフォルト設定

次の表に、ファイル システム パラメータのデフォルト設定を示します。

Table 16: **デフォルトのファイル システム設定**

パラメータ	デフォルト
デフォルト ファイルシステム	bootflash:

コンソール ポートの設定

コンソール ポートに対して次の特性を設定できます。

- データ ビット
- 非アクティブ セッションのタイムアウト
- パリティ
- スピード
- ストップ ビット

Before you begin

コンソール ポートにログインします。

SUMMARY STEPS

1. **configure terminal**
2. **line console**
3. **databits** *bits*
4. **exec-timeout** *minutes*
5. **parity** {**even** | **none** | **odd**}
6. **speed** {**300** | **1200** | **2400** | **4800** | **9600** | **38400** | **57600** | **115200**}
7. **stopbits** {**1** | **2**}
8. **exit**
9. (Optional) **show line console**
10. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**line console**
	Example:
	`switch# line console`
	`switch(config-console)#`
		コンソール 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	**databits** *bits* **Example:** `switch(config-console)# databits 7`	1バイトあたりのデータ ビット数を設定します。指定できる範囲は5～8です。デフォルトは8です。
ステップ 4	**exec-timeout** *minutes* **Example:** `switch(config-console)# exec-timeout 30`	非アクティブ セッションのタイムアウトを設定します。有効値は0～525600分（8760時間）です。0分の値を設定すると、セッション タイムアウトはディセーブルになります。デフォルトは30分です。
ステップ 5	**parity** {**even**	**none**	**odd**} **Example:** `switch(config-console)# parity even`	パリティを設定します。デフォルトは **none** です。

	Command or Action	Purpose
ステップ 6	**speed** {**300** | **1200** | **2400** | **4800** | **9600** | **38400** | **57600** | **115200**} **Example:** `switch(config-console)# speed 115200`	送信および受信速度を設定します。デフォルトは、9600 です。
ステップ 7	**stopbits** {**1** | **2**} **Example:** `switch(config-console)# stopbits 2`	ストップ ビットを設定します。デフォルトは 1 です。

ステップ 8	**exit**
	Example:
	`switch(config-console)# exit`
	`switch(config)#`
		コンソール コンフィギュレーション モードを終了します。

ステップ 9	(Optional) **show line console** **Example:** `switch(config)# show line console`	コンソールの設定値を表示します。
ステップ 10	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

仮想端末の設定

ここでは、Cisco NX-OS デバイスで仮想端末を設定する方法について説明します。

非アクティブ セッション タイムアウトの設定

デバイス上の非アクティブな仮想端末セッションのタイムアウトを設定できます。

SUMMARY STEPS

1. **configure terminal**
2. **line vty**
3. **exec-timeout** *minutes*
4. **exit**
5. (Optional) **show running-config all | begin vty**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**line vty**
	Example:
	`switch# line vty`
	`switch(config-line)#`
		ライン 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	**exec-timeout** *minutes* **Example:** `switch(config-line)# exec-timeout 30`	VDCを設定します。有効値は 0 ～ 525600 分（8760 時間）です。0 分の値を設定すると、タイムアウトはディセーブルになります。デフォルト値は 30 です。

ステップ 4	**exit**
	Example:
	`switch(config-line)# exit`
	`switch(config)#`
		ライン コンフィギュレーション モードを終了します。

ステップ 5	(Optional) **show running-config all | begin vty** **Example:** `switch(config)# show running-config all | begin vty`	仮想端末の設定を表示します。
ステップ 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

セッション制限の設定

デバイス上の仮想端末セッションの数を制限できます。

SUMMARY STEPS

1. **configure terminal**
2. **line vty**
3. **session-limit** *sessions*
4. **exit**
5. (Optional) **show running-config all | begin vty**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**line vty**
	Example:
	`switch# line vty`
	`switch(config-line)#`
		ライン 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	**session-limit** *sessions* **Example:** `switch(config-line)# session-limit 10`	デバイスの仮想セッションの最大数を設定します。範囲は 1 ～ 64 です。デフォルトは 32 です。

ステップ 4	**exit**
	Example:
	`switch(config-line)# exit`
	`switch(config)#`
		ライン コンフィギュレーション モードを終了します。

ステップ 5	(Optional) **show running-config all | begin vty** **Example:** `switch(config)# show running-config all | begin vty`	仮想端末の設定を表示します。
ステップ 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ターミナル セッションのクリア

デバイス上の端末セッションをクリアすることができます。

SUMMARY STEPS

1. (Optional) **show users**
2. **clear line** *name*

DETAILED STEPS

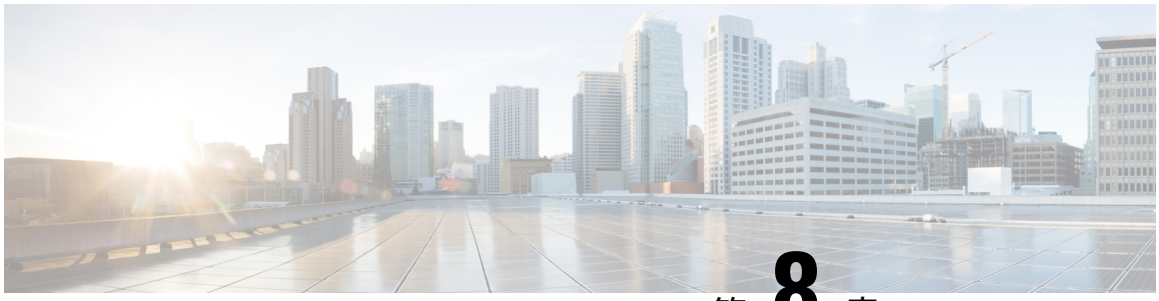
Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **show users** **Example:** `switch# show users`	デバイスのユーザ セッションを表示します。
ステップ 2	**clear line** *name* **Example:** `switch# clear line pts/0`	特定の回線のターミナル セッションをクリアします。回線名では大文字と小文字が区別されます。

端末およびセッション情報の表示

端末およびセッション情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show terminal	端末設定を表示します。
show line	COM1 およびコンソール ポートの設定を表示します。
show users	仮想ターミナル セッションを表示します。
show running-config [**all**]	実行コンフィギュレーションのユーザ アカウント設定を表示します。**all** キーワードを指定すると、ユーザ アカウントのデフォルト値が表示されます。



第 8 章

基本的なデバイス管理

この章は、次の内容で構成されています。

基本的なデバイス管理について

ここでは、基本的なデバイス管理の概要について説明します。

デバイスのホスト名

コマンドプロンプトに表示されるデバイスのホスト名を、デフォルト（switch）から別のストリングに変更できます。デバイスに固有のホスト名を付けると、コマンドラインインターフェイス（CLI）プロンプトからそのデバイスを容易に特定できます。

Message-of-the-Day バナー

Message-of-The-Day（MOTD）バナーは、デバイス上でユーザ ログイン プロンプトの前に表示されます。このメッセージには、デバイスのユーザに対して表示する任意の情報を含めることができます。

デバイス クロック

デバイスを NTP クロック ソースなどの有効な外部の時間調整機構と同期させない場合は、デバイスの起動時にクロック タイムを手動で設定できます。

クロック マネージャ

Cisco NX-OS デバイスには、同期が必要になることがある、異なるタイプのクロックが含まれている可能性があります。これらのクロックはさまざまなコンポーネント（スーパーバイザ、ラインカードプロセッサ、ラインカードなど）の一部であり、それぞれ異なるプロトコルを使用している可能性があります。

クロック マネージャには、これらの異なるクロックを同期する機能があります。

タイム ゾーンと夏時間

デバイスのタイム ゾーンと夏時間を設定できます。これらの値により、クロックの時刻が協定世界時（UTC）からオフセットされます。UTC は、国際原子時（TAI）をベースにしており、うるう秒を定期的に追加することで地球の自転の遅れを補償しています。UTC は、以前はグリニッジ標準時（GMT）と呼ばれていました。

ユーザ セッション

デバイス上のアクティブなユーザセッションを表示できます。また、ユーザセッションにメッセージを送信することもできます。ユーザー セッションとアカウントの管理の詳細については、『 *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

基本的なデバイス パラメータのデフォルト設定

次の表に、基本的なデバイス パラメータのデフォルト設定を示します。

Table 17: **デフォルトの基本的なデバイス パラメータ**

パラメータ	デフォルト
MOTD バナー テキスト	User Access Verification
クロック タイム ゾーン	UTC

デバイスのホスト名の変更

コマンドプロンプトに表示されるデバイスのホスト名を、デフォルト（switch）から別のストリングに変更できます。

SUMMARY STEPS

1. **configure terminal**
2. {**hostname** | **switchname**} *name*
3. **exit**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		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 2	{**hostname** | **switchname**} *name*
	Example:
	hostname コマンドの使用：
	`switch(config)# hostname Engineering1`
	`Engineering1(config)#`
	switchname コマンドの使用：
	`Engineering1(config)# switchname Engineering2`
	`Engineering2(config)#`
		デバイスのホスト名を変更します。*name* 引数は英数字で、大文字と小文字が区別されます。デフォルトは switch です。
		Note
		switchname コマンドは、**hostname** コマンドと同じ機能を実行します。Cisco NX-OS リリース 7.0(3)I7(3) 以降、スイッチ名の最大長 63 文字がサポートされています。

ステップ 3	**exit**
	Example:
	`Engineering2(config)# exit`
	`Engineering2#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	(Optional) **copy running-config startup-config** **Example:** `Engineering2# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

MOTD バナーの設定

ユーザーがログインするときに端末でログイン プロンプトの後に MOTD が表示されるよう設定できます。MOTD バナーには、次の特徴があります。

- 1 行あたり最大 255 文字
- 最大 40 行

SUMMARY STEPS

1. **configure terminal**
2. **banner motd** *delimiting-character message delimiting-character*
3. **exit**
4. (Optional) **show banner motd**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**banner motd** *delimiting-character message delimiting-character*
	Example:
	`switch(config)# banner motd #Welcome to the Switch#`
	`switch(config)#`
		MoTDバナーを設定します。メッセージテキストでは、区切り文字を使用しないでください。
		Note
		" または % は、区切り文字に使用しないでください。現在よりも後のリリースにアップグレードする場合は、この機能の制限を確認してください。
		Note
		Cisco NX-OS リリース 10.1(x) 以降では、特殊文字、"、%、>、<、''（スペース）、および 0x15 よりも小さな ASCII 文字は、デリミタとして無効です。既存の MOTD バナーがこれらの区切り文字を含んでいて、それを編集した場合、またはこれらの区切り文字を含む新しいバナーを追加した場合、バナーは実行構成に含められません。
		以前のリリース（10.x リリースより前のリリース）から既存の 10.x リリースにアップグレードしても、CLI の構成には影響がなく、構成は実行構成でも同じになります。

ステップ 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	(Optional) **show banner motd** **Example:**	設定された MOTD バナーを表示します。

	Command or Action	Purpose
	`switch# show banner motd`	
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

タイム ゾーンの設定

UTC からデバイスのクロック時刻をオフセットするためにタイム ゾーンを設定できます。

SUMMARY STEPS

1. **configure terminal**
2. **clock timezone** *zone-name offset-hours offset-minutes*
3. **exit**
4. (Optional) **show clock**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**clock timezone** *zone-name offset-hours offset-minutes* **Example:** `switch(config)# clock timezone EST -5 0`	タイム ゾーンを設定します。*zone-name* 引数は、タイム ゾーンの略語（PST や EST など）である 3 文字の文字列です。*offset-hours* 引数は、UTC からのオフセット値であり、有効な範囲は –23 ～ 23 時間です。*offset-minutes* 引数の範囲は、0 ～ 59 分です。

ステップ 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	(Optional) **show clock** **Example:** `switch# show clock`	時間とタイム ゾーンを表示します。

	Command or Action	Purpose
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

夏時間の設定

デバイスで夏時間を有効にする時期と、オフセット（分単位）を設定できます。

SUMMARY STEPS

1. **configure terminal**
2. **clock summer-time** *zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes*
3. **exit**
4. (Optional) **show clock detail**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**clock summer-time** *zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes* **Example:** `switch(config)# clock summer-time PDT 1 Sunday March 02:00 1 Sunday November 02:00 60`	夏時間を設定します。 *zone-name* 引数は、タイムゾーンの略語（PST、EST など）である 3 文字のストリングです。 *start-day* 引数と *end-day* 引数の値は、**Monday**、**Tuesday**、**Wednesday**、**Thursday**、**Friday**、**Saturday**、および **Sunday**です。 *start-month* および *end-month* 引数の値は **January**、**February**、**March**、**April**、**May**、**June**、**July**、**August**、**September**、**October**、**November**、および **December**.です。 *start-time*および *end-time* 引数の値は、*hh*:*mm* フォーマットです。 *offset-minutes* 引数の範囲は、0 ～ 1440 分です。

	Command or Action	Purpose

ステップ 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	(Optional) **show clock detail** **Example:** `switch(config)# show clock detail`	設定された MOTD バナーを表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

デバイス クロックの手動設定

デバイスがリモートの時刻源にアクセスできない場合、クロックを手動で設定できます。

Before you begin

タイム ゾーンを設定します。

SUMMARY STEPS

1. **clock set** *time day month year*
2. (Optional) **show clock**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**clock set** *time day month year*
	Example:
	`switch# clock set 15:00:00 30 May 2013`
	`Fri May 30 15:14:00 PDT 2013`
		デバイス クロックを設定します。
		time 引数のフォーマットは *hh*:*mm*:*ss* です。
		day 引数の範囲は 1 ～ 31 です。
		month 引数の値は **January**、**February**、**March**、**April**、**May**、**June**、**July**、**August**、**September**、**October**、**November**、および **December** です。
		year の引数の範囲は 2000 ～ 2030 です。

ステップ 2	(Optional) **show clock** **Example:**	現在のクロック値を表示します。

	Command or Action	Purpose
	`switch(config)# show clock`	

クロック マネージャの設定

Cisco Nexus デバイスのコンポーネントのすべてのクロックを同期するように、Clock Manager を構成できます。

手順の概要

1. **clock protocol** *protocol*
2. （任意） **show run clock_manager**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**clock protocol** *protocol* 例： `switch# clock protocol ntp`	クロック マネージャを設定します。 *protocol* 引数の値は **ntp**、**ptp**、 および **none**.です。 次に、値について説明します。 • **ntp**：クロックとネットワーク タイム プロトコル（NTP）を同期します。 • **ptp**：IEEE 1588 で記述されているとおりに、クロックを高精度時間プロトコル（PTP）と同期します。 • **none—clock set** *HH:MM:SS* を使用します スーパーバイザ クロックを設定します。 （注） **none** を使用する場合は、クロックを構成する必要があります。 （注） プロトコルが構成されたら、クロックはそのプロトコルを使用する必要があります。
ステップ 2	（任意） **show run clock_manager** 例： `switch# show run clock_manager`	クロック マネージャの設定を表示します。

ユーザーの管理

デバイスにログインしたユーザの情報を表示したり、それらのユーザにメッセージを送信したりできます。

ユーザ セッションに関する情報の表示

デバイス上のユーザ セッションに関する情報を表示できます。

SUMMARY STEPS

1. **show users**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**show users** **Example:** `switch# show users`	ユーザ セッションを表示します。

ユーザーへのメッセージ送信

デバイス CLI を使用して、現在アクティブなユーザにメッセージを送信できます。

SUMMARY STEPS

1. (Optional) **show users**
2. **send** [**session** *line*] *message-text*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **show users** **Example:** `switch# show users`	アクティブなユーザ セッションを表示します。

	Command or Action	Purpose
ステップ 2	**send** [**session** *line*] *message-text* **Example:** `switch# send Reloading the device is 10 minutes!`	すべてのアクティブなユーザまたは特定のユーザにメッセージを送信します。このメッセージは最大80文字の英数字で、大文字と小文字が区別されます。

デバイス コンフィギュレーションの確認

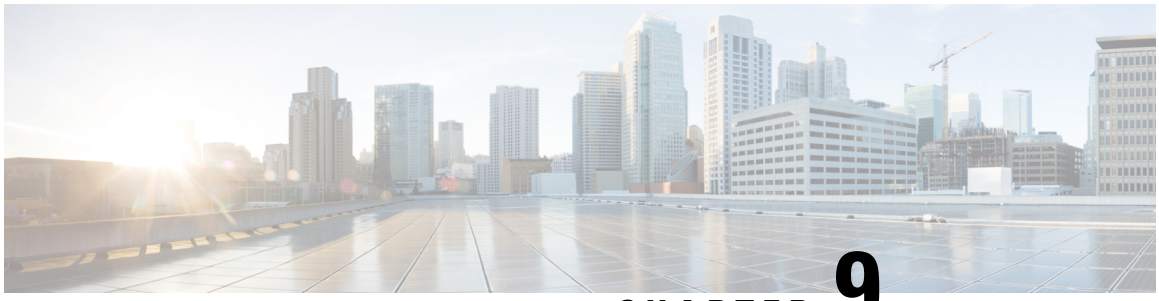
構成を確認するためには、次のいずれかのコマンドを使用します。

コマンド	目的
show running-config [**[exclude]** *command*] **[sanitized]**	現在の実行コンフィギュレーションまたはそのコンフィギュレーションのサブセットの内容を表示するには、該当するモードで **show running-config** コマンドを使用します。。 • **exclude**：（任意）特定のコンフィギュレーションを表示から除外します。 **exclude** キーワードのあとに *command* 引数を指定し、表示から特定のコンフィギュレーションを除外します。 • コマンド：（任意）1 つのコマンドのみを、または指定のコマンド ノード下で使用可能なコマンドのサブセットを表示します。 • **sanitized**：（任意）安全な配布と分析のためにサニタイズされたコンフィギュレーションを表示します。 Cisco NX-OS リリース 10.3(2)F 以降、**sanitized** キーワードが Cisco Nexus 9000 シリーズ スイッチでサポートされています。
show startup-config	スタートアップ コンフィギュレーションを表示します。 **Note** レイヤ 3 ベースの機能構成が running-config で無効になっている場合、**show startup-config** コマンドはそれらを表示しません。ただし、**copy running startup** コマンドが実行されるまで、構成はスタートアップ PSS にそのまま残ります。
show time-stamp running-config last-changed	実行構成が最後に変更されたときのタイムスタンプを表示します。

次に、**show running-config**コマンドで **sanitized** キーワードを指定した場合の出力例を示します。サニタイズされた構成は、構成の一部の詳細を公開せずに、構成を共有するために使用できます。

このオプションは、実行構成出力の機密ワードを<removed> キーワードによりマスクします。

```
switch# show running-config sanitized

!Command: show running-config sanitized
!Running configuration last done at: Wed Oct 12 09:14:54 2022
!Time: Wed Oct 12 13:52:55 2022

version 10.3(2) Bios:version 07.69

username admin password 5 <removed> role network-admin

copp profile strict
snmp-server user admin network-admin auth md5 <removed> priv aes-128 <removed>
localizedV2key
rmon event 1 log trap <removed> description FATAL(1) owner PMON@FATAL
rmon event 2 log trap <removed> description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 log trap <removed> description ERROR(3) owner PMON@ERROR
rmon event 4 log trap <removed> description WARNING(4) owner PMON@WARNING
rmon event 5 log trap <removed> description INFORMATION(5) owner PMON@INFO
--More--
```



CHAPTER 9

デバイスのファイル システム、ディレクトリ、およびファイルの使用方法

この章は、次の内容で構成されています。

デバイスのファイル システム、ディレクトリ、およびファイルについて

ここでは、Cisco NX-OS デバイスのファイル システム、ディレクトリ、およびファイルについて説明します。

ファイル システム

ローカル ファイル システムを指定するための構文は、*filesystem*:[*//modules/*] です。

次の表に、デバイス上で参照可能な各種ファイル システムの説明を示します。

Table 18: ファイル システムの構文の構成要素

ファイル システム名	モジュール	説明
ブートフラッシュ	sup-active sup-local	イメージ ファイル、コンフィギュレーション ファイル、およびその他のファイルを格納するためにアクティブ スーパーバイザ モジュール上にある内部 CompactFlash メモリ。初期デフォルト ディレクトリは bootflash です。
	sup-standby sup-remote	イメージ ファイル、コンフィギュレーション ファイル、およびその他のファイルを格納するためにスタンバイ スーパーバイザ モジュール上にある内部 CompactFlash メモリ。
volatile	—	スーパーバイザ モジュールにある、一時的または保留中の変更のために使用される揮発性 RAM（VRAM）。
log	—	ロギング ファイル統計情報を格納する、アクティブ スーパーバイザ上のメモリ。
system	—	実行コンフィギュレーション ファイルを格納するために使用される、スーパーバイザ モジュール上のメモリ。
debug	—	デバッグ ログに使用される、スーパーバイザ モジュール上のメモリ。

ディレクトリ

bootflash: および外部フラッシュ メモリ（usb1:、および usb2:）にディレクトリを作成できます。これらのディレクトリ間を移動して、ファイルの保存に使用できます。

ファイル

bootflash:、volatile:、slot0:、usb1:、および usb2: ファイル システムにファイルを作成し、アクセスします。system: filesystem 上のファイルにのみアクセスできます。 **debug logfile**command には、log: ファイル システムを使用できます。

FTP、Secure Copy（SCP）、セキュア シェル FTP（SFTP）、および TFTP を使用して、リモートサーバーから nx-os イメージファイルなどのファイルをダウンロードできます。デバイスが SCP サーバーとして動作できるので、外部サーバーからデバイスへファイルをコピーすることもできます。

注意事項と制約事項

デバイスのファイル システム、ディレクトリ、およびファイルの注意事項と制限事項は次のとおりです。

- **show tech-support details** コマンドは、Ctrl+Z を使用して終了できません。代わりに、Ctrl+C を使用してコマンドを終了します。
- 「network-admin」ロールを持つユーザーを利用して、ブートフラッシュ内のファイルを変更します。
- リリース 10.5(1) 以降、想定される設定サイズと一致するように、Nexus 9000 上の SSD パーティション サイズを自動的に検出できます。ブートアップ中に **show logging log** または **show logging nvram** コマンドに情報 syslog が表示され、NX-OS Nexus 9000 が予期しない SSD パーティション分割サイズで起動されたことを示します。

```
%PLATFORM-2-SSD_PARTITION_CHECK: Incorrect <device> partition size detected - please
 contact Cisco TAC for additional information
```

ファイル システム パラメータのデフォルト設定

次の表に、ファイル システム パラメータのデフォルト設定を示します。

Table 19: **デフォルトのファイル システム設定**

パラメータ	デフォルト
デフォルト ファイルシステム	bootflash:

FTP、HTTP、または TFTP 送信元インターフェイスの構成

File Transfer Protocol（FTP）、Hypertext Transfer Protocol（HTTP）、または Trivial File Transfer Protocol（TFTP）の送信元インターフェイスを構成できます。この構成により、コピーパケッ

トを転送する際に、設定した送信元インターフェイスに関連付けられた IP アドレスを使用することができます。

手順の概要

1. **configure terminal**
2. [**no**] **ip** {**ftp** | **http** | **tftp**} **source-interface** {**ethernet** *slot*/*port* | **loopback** *number*}
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	[**no**] **ip** {**ftp**	**http**	**tftp**} **source-interface** {**ethernet** *slot*/*port*	**loopback** *number*} 例： `switch(config)# ip tftp source-interface ethernet 2/1`	すべてのFTP、HTTP、またはTFTPパケットの送信元インターフェイスを構成します。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

ディレクトリの操作

ここでは、Cisco NX-OS デバイスでディレクトリを操作する手順を説明します。

カレント ディレクトリの特定

カレント ディレクトリのディレクトリ名を表示できます。

SUMMARY STEPS

1. **pwd**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**pwd** **Example:** `switch# pwd`	カレント ディレクトリの名前を表示します。

カレント ディレクトリの変更

ファイル システムの操作のためカレント ディレクトリを変更できます。初期デフォルト ディレクトリは bootflash: です。

SUMMARY STEPS

1. (Optional) **pwd**
2. **cd** {*directory* | *filesystem*:[//*module*/][*directory*]}

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	**cd** {*directory* | *filesystem*:[//*module*/][*directory*]} **Example:** `switch# cd usb1:`	新しいカレントディレクトリに変更します。ファイル システム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。

ディレクトリの作成

bootflash: およびフラッシュ デバイス ファイル システムでディレクトリを作成できます。

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **cd** {*directory* | *filesystem*:[//*module*/][*directory*]}
3. **mkdir** [*filesystem*:[//*module*/]]*directory*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	(Optional) **cd** {*directory* | *filesystem*:[//*module*/][*directory*]} **Example:** `switch# cd slot0:`	新しいカレントディレクトリに変更します。ファイル システム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。
ステップ 3	**mkdir** [*filesystem*:[//*module*/]]*directory* **Example:** `switch# mkdir test`	新しいディレクトリを作成します。*filesystem* 引数では、大文字と小文字が区別されます。*directory* 引数は、64 文字以内の英数字で指定します。大文字と小文字が区別されます。

ディレクトリの内容の表示

ディレクトリの内容を表示できます。

SUMMARY STEPS

1. **dir** [*directory* | *filesystem*:[//*module*/][*directory*]]

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**dir** [*directory* | *filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash:test`	ディレクトリの内容を表示します。デフォルト値は、現在の作業ディレクトリです。ファイル システム名およびディレクトリ名では、大文字と小文字が区別されます。

ディレクトリの削除

ディレクトリは、デバイス上のファイル システムから削除できます。

Before you begin

ディレクトリを削除する前に、ディレクトリが空白であることを確認します。

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]]
3. **rmdir** [*filesystem* **:**[*//module/*]]*directory*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	(Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]] **Example:** `switch# dir bootflash:test`	カレントディレクトリの内容を表示します。ファイル システム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。 ディレクトリが空白でない場合は、ディレクトリを削除する前に、ディレクトリ内のすべてのファイルを削除する必要があります。
ステップ 3	**rmdir** [*filesystem* **:**[*//module/*]]*directory* **Example:** `switch# rmdir test`	ディレクトリを削除します。ファイル システムおよびディレクトリ名では、大文字と小文字が区別されます。

スタンバイ スーパーバイザ モジュール上のディレクトリへのアクセス

アクティブ スーパーバイザ モジュール上のセッションからスタンバイ スーパーバイザ モジュール（リモート）上のすべてのファイル システムにアクセスできます。この機能は、アクティブ スーパーバイザ モジュールにファイルをコピーしたら、同じファイルをスタンバイ スーパーバイザ モジュール上にも存在させる必要がある場合に役立ちます。アクティブ スーパーバイザ モジュール上のセッションからスタンバイ スーパーバイザ モジュール上のファイル システムにアクセスするには、 *filesystem***://sup-remote/** または *filesystem***://sup-standby/** を使用して、ファイルのパスにスタンバイ スーパーバイザ モジュールを指定します。

ファイルの使用

ここでは、Cisco NX-OS デバイスでファイルを使用する手順を説明します。

ファイルの移動

ディレクトリ内のファイルを別のディレクトリに移動できます。

Caution 宛先ディレクトリに同名のファイルがすでに存在する場合は、そのファイルは移動対象のファイルによって上書きされます。

move コマンドを使用して、同じディレクトリ内でファイルを移動することにより、ファイルの名前を変更できます。

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem*:[//*module*/][*directory*]]
3. **move** [*filesystem*:[//*module*/][*directory* /] | *directory*/]*source-filename* {{*filesystem*:[//*module*/][*directory* /] | *directory*/}[*target-filename*] | *target-filename*}

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	(Optional) **dir** [*filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash`	カレントディレクトリの内容を表示します。ファイルシステムおよびディレクトリ名では、大文字と小文字が区別されます。
ステップ 3	**move** [*filesystem*:[//*module*/][*directory* /]	*directory*/]*source-filename* {{*filesystem*:[//*module*/][*directory* /]	*directory*/}[*target-filename*]	*target-filename*} **Example:** `switch# move test old_tests/test1`	ファイルを移動します。 ファイルシステム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。 *target-filename* 引数は、64 文字以内の英数字で指定します。大文字と小文字が区別されます。*target-filename* 引数を指定しないと、ファイル名はデフォルトで *source-filename* 引数値に設定されます。

ファイルのコピー

ファイルのコピーは、同じディレクトリまたは別のディレクトリのいずれかで作成できます。詳細については、『*Cisco Nexus 9000 Series NX-OS* トラブルシューティング ガイド』を参照してください。

Note **dir** コマンドを使用して、コピー先のファイル システムに十分な領域があることを確認します。十分な領域が残っていない場合は、**delete** コマンドを使用して不要なファイルを削除します。

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem*:[//*module*/][*directory*]]
3. **copy** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*source-filename* | {*filesystem*:[//*module*/][*directory*/] | *directory*/}[*target-filename*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	(Optional) **dir** [*filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash`	カレントディレクトリの内容を表示します。ファイルシステムおよびディレクトリ名では、大文字と小文字が区別されます。
ステップ 3	**copy** [*filesystem*:[//*module*/][*directory*/]	*directory*/]*source-filename*	{*filesystem*:[//*module*/][*directory*/]	*directory*/}[*target-filename*] **Example:** `switch# copy test old_tests/test1`	ファイルをコピーします。ファイル システム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。*source-filename* の引数は、64 文字以内の英数字で指定します。大文字と小文字が区別されます。*target-filename* 引数を指定しないと、ファイル名はデフォルトで *source-filename* 引数値に設定されます。

HTTP または HTTPS を使用したファイルのコピー

HTTP または HTTPS を使用して、リモート サーバのファイルのコピーをローカル デバイスに作成できます。

（注） Cisco NX-OS リリース 10.4(3)F 以降、 **copy http** または **copy https** コマンドは、Cisco Nexus スイッチで TLS バージョン 1.3 および 1.2 をサポートします。

手順の概要

1. （任意） **pwd**
2. （任意） **dir** [*filesystem*:[//*module*/][*directory*]]
3. **copy https://** *username*:*password*@*directory*/*filename* **bootflash: vrf management**
4. **copy http://** *directory*/*filename* **bootflash: vrf management**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	（任意） **pwd** 例： `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	（任意） **dir** [*filesystem*:[//*module*/][*directory*]] 例： `switch# dir bootflash`	カレントディレクトリの内容を表示します。ファイル システムおよびディレクトリ名では、大文字と小文字が区別されます。
ステップ 3	**copy https://** *username*:*password*@*directory*/*filename* **bootflash: vrf management** 例： `switch(config)# copy https://username1:pwd1@192.168.0.1/test.txt bootflash: vrf management`	**https** オプションを使用して、指定されたファイルをリモート サーバからローカル デバイスにコピーします。
ステップ 4	**copy http://** *directory*/*filename* **bootflash: vrf management** 例： `switch(config)# copy http://192.168.0.1/test.txt bootflash: vrf management`	**http** オプションを使用して、指定されたファイルをリモート サーバーからローカル デバイスにコピーします。

ファイルの削除

ディレクトリからファイルを削除できます。

SUMMARY STEPS

1. (Optional) **dir** [*filesystem*:[//*module*/][*directory*]]
2. **delete** {*filesystem*:[//*module*/][*directory*/] | *directory*/}*filename*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **dir** [*filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash:`	カレントディレクトリの内容を表示します。ファイルシステムおよびディレクトリ名では、大文字と小文字が区別されます。
ステップ 2	**delete** {*filesystem*:[//*module*/][*directory*/] | *directory*/}*filename* **Example:** `switch# delete bootflash:old_config.cfg`	ファイルを削除します。ファイル システム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。*source-filename* 引数では、大文字と小文字が区別されます。 **Caution** 1 つのディレクトリを指定している場合、**delete** コマンドではディレクトリ全体とその内容すべてが削除されます。

ファイル内容の表示

ファイルの内容を表示できます。

SUMMARY STEPS

1. **show file** [*filesystem*:[//*module*/]][*directory*/]*filename*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**show file** [*filesystem*:[//*module*/]][*directory*/]*filename* **Example:** `switch# show file bootflash:test-results`	ファイルの内容を表示します。

ファイル チェックサムの表示

ファイルの整合性をチェックするチェックサムを表示できます。

SUMMARY STEPS

1. **show file** [*filesystem*:[//*module*/]][*directory*/]*filename* {**cksum** | **md5sum**}

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**show file** [*filesystem*:[//*module*/]][*directory*/]*filename* {**cksum** | **md5sum**} **Example:** `switch# show file bootflash:trunks2.cfg cksum`	ファイルのチェックサムまたは MD5 チェックサムを表示します。

ファイルの圧縮と解凍

Lempel-Ziv 1977（LZ77）コーディングを使用してデバイス上のファイルを圧縮および圧縮解除できます。

SUMMARY STEPS

1. (Optional) **dir** [*filesystem*:[//*module*/]*directory*]]
2. **gzip** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*filename*
3. **gunzip** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*filename* **.gz**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **dir** [*filesystem*:[//*module*/]*directory*]] **Example:** `switch# dir bootflash:`	カレントディレクトリの内容を表示します。ファイルシステムおよびディレクトリ名では、大文字と小文字が区別されます。
ステップ 2	**gzip** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*filename* **Example:** `switch# gzip show_tech`	ファイルを圧縮します。ファイルが圧縮されると、.gz サフィクスが付けられます。
ステップ 3	**gunzip** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*filename* **.gz** **Example:** `switch# gunzip show_tech.gz`	ファイルを圧縮解除します。圧縮解除するファイルのサフィクスは、.gz である必要があります。ファイルが圧縮解除されると、サフィクスは .gz ではなくなります。

ファイルの最終行の表示

ファイルの最後の行を表示できます。

SUMMARY STEPS

1. **tail** [*filesystem*:[//*module*/]][*directory*/]*filename* [*lines*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**tail** [*filesystem*:[//*module*/]][*directory*/]*filename* [*lines*] **Example:** `switch# tail ospf-gr.conf`	ファイルの最後の行を表示します。デフォルトの行数は 10 です。有効な範囲は 0 ～ 80 行です。

ファイルへの show コマンド出力のリダイレクト

show コマンド出力は、bootflash:、slot0:、volatile:、またはリモート サーバ上のファイルにリダイレクトできます。また、コマンド出力のフォーマットを指定することもできます。

SUMMARY STEPS

1. (Optional) **terminal redirection-mode** {**ascii** | **zipped**}
2. *show-command* > [*filesystem*:[//*module*/][*directory*] | [directory /]]*filename*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **terminal redirection-mode** {**ascii**	**zipped**} **Example:** `switch# terminal redirection-mode zipped`	ユーザー セッションに対して **show** コマンド出力のリダイレクション モードを設定します。デフォルト モードは **ascii** です。
ステップ 2	*show-command* > [*filesystem*:[//*module*/][*directory*]	[directory /]]*filename* **Example:** `switch# show tech-support > bootflash:techinfo`	**show** コマンドからの出力をファイルにリダイレクトします。

ファイルの検索

特定のストリングで始まる名前を持つファイルを現在の作業ディレクトリとそのサブディレクトリで検索できます。

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **cd** {*filesystem*:[//*module*/][*directory*] | *directory*}
3. **find** *filename-prefix*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	(Optional) **cd** {*filesystem*:[//*module*/][*directory*]	*directory*}	デフォルト ディレクトリを変更します。 **Example:** `switch# cd bootflash:test_scripts`
ステップ 3	**find** *filename-prefix* **Example:** `switch# find bgp_script`	指定したファイル名プレフィックスで始まるすべてのファイル名をデフォルトディレクトリとそのサブディレクトリで検索します。ファイル名プレフィクスでは、大文字と小文字が区別されます。

ブートフラッシュのフォーマット

format bootflash: CLI コマンドを使用して、オンボードフラッシュ メモリ（bootflash:）をフォーマットします。すべての仮想サービスをアクティブ解除し、もう一度実行してください（`Deactivate all virtual-services and try again`）というエラー メッセージが原因でコマンドがエラーになった場合は、**guestshell destroy** CLI コマンドを使用してゲスト シェルを破棄し、次のように **format bootflash:** コマンドを再実行します。

```
switch# sh virtual-service list
Virtual Service List:

Name                      Status              Package Name
------------------------------------------------------------------------
guestshell+               Activated           guestshell.ova

switch#

switch# guestshell destroy
You are about to destroy the guest shell and all of its contents.  Be sure to save your
 work. Are you sure you want to continue? (y/n) [n] y

switch# 2018 Jan 17 18:42:24 switch %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Deactivating
virtual service 'guestshell+'
```

```
switch#format bootflash:
```

アーカイブ ファイルの操作

Cisco NX-OS ソフトウェアはアーカイブ ファイルをサポートしています。アーカイブ ファイルを作成したり、既存のアーカイブ ファイルにファイルを追加したり、アーカイブ ファイルからファイルを抽出したり、アーカイブ ファイル内のファイルのリストを表示したりすることができます。

アーカイブ ファイルの作成

アーカイブ ファイルを作成し、アーカイブ ファイルにファイルを追加できます。次の圧縮タイプを指定できます。

- bzip2
- gzip
- 未圧縮

デフォルト値は gzip です。

手順の概要

1. **tar create** {**bootflash:** | **volatile:**}*archive-filename* [**absolute**] [**bz2-compress**] [**gz-compress**] [**remove**] [**uncompressed**] [**verbose**] *filename-list*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**tar create** {**bootflash:** | **volatile:**}*archive-filename* [**absolute**] [**bz2-compress**] [**gz-compress**] [**remove**] [**uncompressed**] [**verbose**] *filename-list* 例： `switch# tar create bootflash:config-archive gz-compress bootflash:config-file`	アーカイブ ファイルを作成し、アーカイブ ファイルにファイルを追加します。ファイル名は英数字で指定します（大文字と小文字は区別されません）。最大文字数は 240 です。 **absolute** キーワードは、先頭のバックスラッシュ文字（\）を、アーカイブ ファイルに追加されたファイルの名前から削除しないことを指定します。デフォルトでは、先頭のバックスラッシュ文字は削除されます。 **bz2-compress**、**gz-compress**、および **uncompressed** のキーワードは、アーカイブにファイルを追加するとき（または後で追加するとき）に使用する圧縮

	コマンドまたはアクション	目的
		ユーティリティと、ファイルを抽出するときに使用する解凍ユーティリティを決定します。アーカイブ ファイルに拡張子を指定しない場合、デフォルト値は次のようになります。 • **bz2-compress** の場合、拡張子は .tar.bz2 です。 • **gz-compress** の場合、拡張子は .tar.gz です。 • **uncompressed** の場合、拡張子は .tar です。 **remove** キーワードは、アーカイブにファイルを追加した後に、Cisco NX-OS ソフトウェアがファイルシステムからこれらのファイルを削除することを指定します。デフォルトでは、ファイルは削除されません。 **verbose** キーワードは、Cisco NX-OS ソフトウェアが、ファイルがアーカイブに追加されるときにファイルをリストすることを指定します。デフォルトで、ファイルは追加されると一覧表示されます。

アーカイブ ファイルへのファイルの追加

デバイス上の既存のアーカイブ ファイルにファイルを追加できます。

始める前に

デバイス上でアーカイブ ファイルを作成しておきます。

手順の概要

1. **tar append** {**bootflash:** | **volatile:**}*archive-filename* [**absolute**] [**remove**] [**verbose**] *filename-list*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**tar append** {**bootflash:**	**volatile:**}*archive-filename* [**absolute**] [**remove**] [**verbose**] *filename-list*	既存のアーカイブ ファイルにファイルを追加します。アーカイブ ファイル名では、大文字と小文字は区別されません。 **absolute** キーワードは、先頭のバックスラッシュ文字（\）を、アーカイブ ファイルに追加されたファイルの名前から削除しないことを指定します。デ

	コマンドまたはアクション	目的
		フォルトでは、先頭のバックスラッシュ文字は削除されます。 **remove** キーワードは、アーカイブにファイルを追加した後に、Cisco NX-OS ソフトウェアがファイルシステムからこれらのファイルを削除することを指定します。デフォルトでは、ファイルは削除されません。 **verbose** キーワードは、Cisco NX-OS ソフトウェアが、ファイルがアーカイブに追加されるときにファイルをリストすることを指定します。デフォルトで、ファイルは追加されると一覧表示されます。

例

次に、既存のアーカイブ ファイルにファイルを追加する例を示します。

```
switch# tar append bootflash:config-archive.tar.gz bootflash:new-config
```

アーカイブ ファイルからのファイルの抽出

デバイス上の既存のアーカイブ ファイルにファイルを抽出できます。

始める前に

デバイス上でアーカイブ ファイルを作成しておきます。

手順の概要

1. **tar extract** {**bootflash:** | **volatile:**}*archive-filename* [**keep-old**] [**screen**] [**to** {**bootflash:** | **volatile:**}[*/directory-name*]] [**verbose**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**tar extract** {**bootflash:** | **volatile:**}*archive-filename* [**keep-old**] [**screen**] [**to** {**bootflash:** | **volatile:**}[*/directory-name*]] [**verbose**] 例： `switch# tar extract bootflash:config-archive.tar.gz`	既存のアーカイブファイルからファイルを抽出します。アーカイブファイル名では、大文字と小文字は区別されません。 **keep-old** キーワードは、Cisco NX-OS ソフトウェアが、抽出されるファイルと同じ名前を持つファイルを上書きしないことを示します。

	コマンドまたはアクション	目的
		screen キーワードは、Cisco NX-OS ソフトウェアが、抽出されるファイルと同じ名前を持つファイルを上書きしないことを示します。 **to** キーワードは、ターゲット ファイルシステムを指定します。ディレクトリ名を含めることができます。ディレクトリ名は、240 文字以内の英数字で指定します。大文字と小文字が区別されます。 **verbose** キーワードは、Cisco NX-OS ソフトウェアが、抽出されるファイルの名前を表示することを指定します。

アーカイブ ファイルのファイル名の表示

tar list コマンドを使用して、アーカイブ ファイル内のファイルの名前を表示できます。

tar list {**bootflash:** | **volatile:**}*archive-filename*

アーカイブ ファイル名では、大文字と小文字は区別されません。

```
switch# tar list bootflash:config-archive.tar.gz
config-file
new-config
```

SSD の再パーティション化

SSD 再パーティション化を構成して、設定ストレージの容量を増やすことができます。これにより、ログフラッシュ ストレージのサイズも増加します。この構成はシステムのリロード後に有効になり、追加の cfg および logflash ストレージ スペースによってブートフラッシュのサイズが減少する可能性があります。

SSD の再パーティション化を実行する前に、すべてのソフトウェア イメージ、構成、および個人データのバックアップを実行することをお勧めします。

リリース 10.5(1) 以降、スイッチ上の SSD パーティション サイズを予想される設定サイズに一致するように自動的に検出できます。ブートアップ中に **show logging log** または **show logging nvram** コマンドに情報 syslog が表示され、スイッチが予期しない SSD パーティション分割サイズで起動されたことを示します。

```
%PLATFORM-2-SSD_PARTITION_CHECK: Incorrect <device> partition size detected - please
contact
        Cisco TAC for additional information
```

拡張パーティション スキームは、64 GB SSD を搭載したプラットフォームではサポートされていません。

手順の概要

1. **system flash sda resize**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**system flash sda resize**
	例：
	`switch# system flash sda resize ?`
	`  <CR>`
	`  extended  Cfg=1GB, logflash=39GB`
	`  standard  Cfg=64MB, logflash=4|8GB`
		永続ストレージのサイズを新しいスキームに変更します。

例

次に、標準サイズ変更の例を示します。

```
switch# system flash sda resize standard

!!!! WARNING !!!!

      Attempts will be made to preserve drive contents during
      the resize operation, but risk of data loss does exist.
      Backing up of bootflash, logflash, and running configuration
      is recommended prior to proceeding.

!!!! WARNING !!!!


current scheme is
sda            8:0    0 119.2G  0 disk
|-sda1         8:1    0   512M  0 part
|-sda2         8:2    0    32M  0 part /mnt/plog
|-sda3         8:3    0   128M  0 part /mnt/pss
|-sda4         8:4    0 114.5G  0 part
/isan/vdc_1/virtual-instance/guestshell+/rootfs/bootflash
|-sda5         8:5    0    64M  0 part /mnt/cfg/0
|-sda6         8:6    0    64M  0 part /mnt/cfg/1
`-sda7         8:7    0     4G  0 part /logflash


 target scheme is
sda            8:0    0    64G|120GB|250GB      0 disk
|-sda1         8:1    0         512M            0 part
|-sda2         8:2    0          32M            0 part /mnt/plog
|-sda3         8:3    0         128M            0 part /mnt/pss
|-sda4         8:4    0       110.5G            0 part /bootflash
|-sda5         8:5    0          64M            0 part /mnt/cfg/0
|-sda6         8:6    0          64M            0 part /mnt/cfg/1
|_sda7         8:7    0           8G            0 part /logflash

Continue? (y/n)  [n] y
```

```
  A module reload is required for the resize operation to proceed
  Please, do not power off the module during this process.
```

次に、拡張サイズ変更の例を示します。

```
switch# system flash sda resize extended

!!!! WARNING !!!!

      Attempts will be made to preserve drive contents during
      the resize operation, but risk of data loss does exist.
      Backing up of bootflash, logflash, and running configuration
      is recommended prior to proceeding.

!!!! WARNING !!!!


current scheme is
sda            8:0    0 119.2G  0 disk
|-sda1         8:1    0   512M  0 part
|-sda2         8:2    0    32M  0 part /mnt/plog
|-sda3         8:3    0   128M  0 part /mnt/pss
|-sda4         8:4    0 110.5G  0 part /bootflash
|-sda5         8:5    0    64M  0 part /mnt/cfg/0
|-sda6         8:6    0    64M  0 part /mnt/cfg/1
`-sda7         8:7    0     8G  0 part /logflash


 target scheme is
sda            8:0    0 120GB|250GB  0 disk
|-sda1         8:1    0   512M       0 part
|-sda2         8:2    0    32M       0 part /mnt/plog
|-sda3         8:3    0   128M       0 part /mnt/pss
|-sda4         8:4    0    rem       0 part /bootflash
|-sda5         8:5    0   1.0G       0 part /mnt/cfg/0
|-sda6         8:6    0   1.0G       0 part /mnt/cfg/1
|_sda7         8:7    0    39G       0 part /logflash

Continue? (y/n)  [n] y
  A module reload is required for the resize operation to proceed
  Please, do not power off the module during this process.
```

Tech-Support コマンドの有効化または無効化

tech-support コマンドを有効または無効にするには、次の手順を実行します。

手順の概要

1. **system tech-support blocked-commands sample_list**
2. **clear system tech-support blocked-commands**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**system tech-support blocked-commands sample_list**
	例：
	`switch# system tech-support blocked-commands sample_list`
	`Successfully enabled tech-support blocked commands list`
		テクニカルサポートのブロックされたコマンドのリストを有効にします。
		このコマンドは、 **show tech-support details [time-optimized]**、 **show tech-support all [time- optimized]**、および **show tech-support commands**の **sample_list** に記載されている show コマンドの実行をブロックします。上記の show-tech コマンドでは、リストされているコマンドは実行されず、スキップされます。

ステップ 2	**clear system tech-support blocked-commands**
	例：
	`switch# clear system tech-support blocked-commands`
	`Successfully cleared tech-support blocked commands list`
		tech -support ブロック コマンド リストをクリアします。

テクニカル サポートでブロックされた CLI の表示

次のコマンドを使用して、テクニカル サポート **blocked-commands** リストのステータスを確認できます。

手順の概要

1. **show system tech-support blocked-commands status**
2. **run bash cat /bootflash/sample_list**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**show system tech-support blocked-commands status**
	例：
	`switch# show system tech-support blocked-commands status`
	`Tech-support blocked commands list status: Disabled`
	`switch# show system tech-support blocked-commands status`
	`Tech-support blocked commands list status: Enabled`
		テクニカル サポート ブロック コマンド リストのステータスを表示します。
		コマンドリストが有効になっている場合は、ファイル名が表示されます。

	コマンドまたはアクション	目的
	`Blocked command file: /bootflash/sample_list` `Last modified time: Thu Dec 7 07:03:02 2023`	

ステップ 2	**run bash cat /bootflash/sample_list**
	例：
	`switch# run bash cat /bootflash/sample_list`
	`show version`
	`show inventory`
	`show module`
	`show tech-support snmp`
		ブロックされたコマンド ファイルを表示します。
		• ファイルの最大長は 128 です。
		• これは EXEC モード コマンドですが、**blocked-commands** は、ファイルが */bootflash* に保持され、すべてのリロードにわたって保持される限り有効です。
		• ファイルが削除された場合でも、**blocked-commands** の状態は有効のままですが、ファイルが削除されているので効果はありません。
		• このファイルには読み取り権限が必要です。

ファイル システムの使用例

この項では、Cisco NX-OS デバイスでファイル システムを使用する例について説明します。

スタンバイ スーパーバイザ モジュール上のディレクトリへのアクセス

次に、スタンバイ スーパーバイザ モジュール上のファイルのリストを表示する例を示します。

```
switch# dir bootflash://sup-remote
       4096    Oct 03 23:55:55 2013  .patch/
...
      16384    Jan 01 13:23:30 2011  lost+found/
  297054208    Oct 21 18:55:36 2013  n9000-dk9.6.1.2.I1.1.bin
...

Usage for bootflash://sup-remote
1903616000 bytes used
19234234368 bytes free
21137850368 bytes total
```

次に、スタンバイ スーパーバイザ モジュール上のファイルを削除する例を示します。

```
switch# delete bootflash://sup-remote/aOldConfig.txt
```

ファイルの移動

次に、外部フラッシュ デバイス上のファイルを移動する例を示します。

```
switch# move usb1:samplefile usb1:mystorage/samplefile
```

次に、デフォルト ファイル システム内のファイルを移動する例を示します。

```
switch# move samplefile mystorage/samplefile
```

ファイルのコピー

次に、usb1: ファイル システムのルート ディレクトリから、samplefile というファイルを mystorage ディレクトリにコピーする例を示します。

```
switch# copy usb1:samplefile usb1:mystorage/samplefile
```

次に、カレント ディレクトリ レベルからファイルをコピーする例を示します。

```
switch# copy samplefile mystorage/samplefile
```

次に、アクティブ スーパーバイザ モジュールのブートフラッシュからスタンバイ スーパーバイザ モジュールのブートフラッシュにファイルをコピーする例を示します。

```
switch# copy bootflash:nx-os-image bootflash://sup-2/nx-os-image
```

次に、NVRAM の既存のコンフィギュレーションの内容を上書きする例を示します。

```
switch# copy nvram:snapshot-config nvram:startup-config

Warning: this command is going to overwrite your current startup-config:
Do you wish to continue? {y/n} [y] y
```

copy コマンドを使用して、 bootflash: ファイル システムと FTP、TFTP、SFTP、または SCP サーバーとの間でファイルのアップロードおよびダウンロードを行うこともできます。

ディレクトリの削除

ディレクトリは、デバイス上のファイル システムから削除できます。

Before you begin

ディレクトリを削除する前に、ディレクトリが空白であることを確認します。

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]]
3. **rmdir** [*filesystem* **:**[*//module/*]]*directory*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **pwd** **Example:** `switch# pwd`	現在のデフォルト ディレクトリの名前を表示します。
ステップ 2	(Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]] **Example:** `switch# dir bootflash:test`	カレントディレクトリの内容を表示します。ファイル システム、モジュール、およびディレクトリ名では、大文字と小文字が区別されます。 ディレクトリが空白でない場合は、ディレクトリを削除する前に、ディレクトリ内のすべてのファイルを削除する必要があります。
ステップ 3	**rmdir** [*filesystem* **:**[*//module/*]]*directory* **Example:** `switch# rmdir test`	ディレクトリを削除します。ファイル システムおよびディレクトリ名では、大文字と小文字が区別されます。

ファイル内容の表示

外部フラッシュ デバイスのファイルの内容を表示する例を示します。

```
switch# show file usb1:test
configure terminal
interface ethernet 1/1
no shutdown
end
show interface ethernet 1/1
```

現在のディレクトリに存在するファイルの内容を表示する例を示します。

```
switch# show file myfile
```

ファイル チェックサムの表示

次に、ファイルのチェックサムを表示する例を示します。

```
switch# show file bootflash:trunks2.cfg cksum
583547619
```

次に、ファイルの MD5 チェックサムを表示する例を示します。

```
switch# show file bootflash:trunks2.cfg md5sum
3b94707198aabefcf46459de10c9281c
```

ファイルの圧縮と解凍

次に、ファイルを圧縮する例を示します。

```
switch# dir
    1525859     Jul 04 00:51:03 2013 Samplefile
...
switch# gzip volatile:Samplefile
switch# dir
     266069     Jul 04 00:51:03 2013 Samplefile.gz
...
```

次に、圧縮ファイルを解凍する例を示します

```
switch# dir
     266069     Jul 04 00:51:03 2013 Samplefile.gz
...
switch# gunzip samplefile
switch# dir
    1525859     Jul 04 00:51:03 2013 Samplefile
...
```

show コマンド出力のリダイレクト

次に、出力を bootflash: ファイル システム上のファイルに誘導する例を示します。

```
switch# show interface > bootflash:switch1-intf.cfg
```

次に、出力を外部フラッシュ メモリ上のファイルに誘導する例を示します。

```
switch# show interface > usb1:switch-intf.cfg
```

次に、出力を TFTP サーバ上のファイルに誘導する例を示します。

```
switch# show interface > tftp://10.10.1.1/home/configs/switch-intf.cfg
Preparing to copy...done
```

次に、**show tech-support** コマンドの出力をファイルにダイレクトする例を示します。

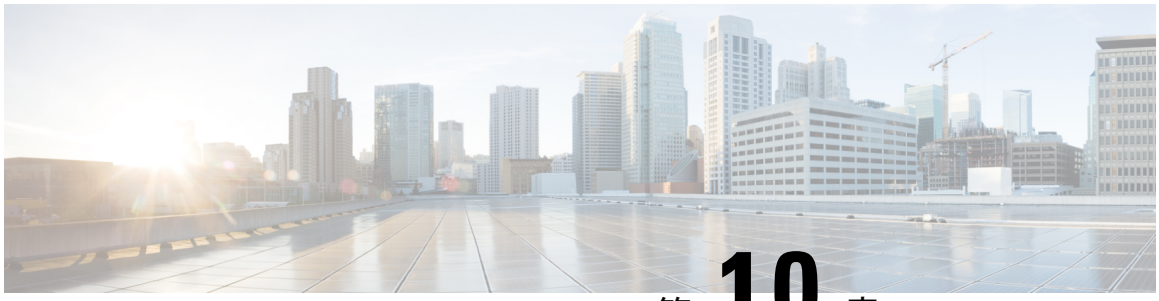
```
switch# show tech-support > Samplefile
Building Configuration ...
switch# dir
    1525859     Jul 04 00:51:03 2013 Samplefile
Usage for volatile://
    1527808 bytes used
   19443712 bytes free
```

```
20971520 bytes total
```

ファイルの検索

次に、現在のデフォルト ディレクトリ内でファイルを検索する例を示します。

```
switch# find smm_shm.cfg
/usr/bin/find: ./lost+found: Permission denied
./smm_shm.cfg
./newer-fs/isan/etc/routing-sw/smm_shm.cfg
./newer-fs/isan/etc/smm_shm.cfg
```



第 10 章

コンフィギュレーション ファイルの取り扱い

この章は、次の内容で構成されています。

構成ファイルについて

構成ファイルには、Cisco NX-OS デバイス上の機能を構成するために使用される Cisco NX-OS ソフトウェアのコマンドが保存されます。コマンドは、システムを起動したとき（startup-config ファイルから）、または構成モードで CLI にコマンドを入力したときに、Cisco NX-OS ソフトウェアによって解析（変換および実行）されます。

スタートアップ構成ファイルを変更するには、**copy running-config startup-config** コマンドを使用してスタートアップ構成に実行構成ファイルを保存するか、ファイル サーバーからスタートアップ構成へ構成ファイルをコピーします。

コンフィギュレーション ファイルのタイプ

Cisco NX-OS ソフトウェアの構成ファイルには、実行構成とスタートアップ構成の 2 種類があります。デバイスは、その起動時にスタートアップ コンフィギュレーション（startup-config）を使用して、ソフトウェア機能を設定します。実行コンフィギュレーション（running-config）には、スタートアップコンフィギュレーションファイルに対して行った現在の変更が保存されます。2つのコンフィギュレーションファイルは別々の設定にできます。デバイス構成は、永続的ではなく一時的に変更することもできます。この場合、グローバル コンフィギュレーション モードでコマンドを使用することにより、実行コンフィギュレーションを変更しますが、スタートアップ コンフィギュレーションにはその変更を保存しないようにします。

実行構成を変更するには、**configure terminal** コマンドを使用して、グローバル構成モードを開始します。Cisco NX-OS 構成モードの使用時には、通常コマンドはすぐに実行され、入力直後または構成モードを終了した時点で実行構成ファイルに保存されます。

スタートアップコンフィギュレーションファイルを変更するには、実行コンフィギュレーション ファイルをスタートアップ コンフィギュレーションに保存するか、コンフィギュレーション ファイルをファイル サーバからスタートアップ コンフィギュレーションにダウンロードします。

Related Topics

実行コンフィギュレーションのスタートアップ コンフィギュレーションへの保存 （166ページ）
リモート サーバからのスタートアップ コンフィギュレーションのダウンロード （169ページ）

構成ファイルに関する注意事項と制限事項

構成ファイルに関する注意事項と制限事項は次のとおりです。

- NX-OS 7.0(3)I7(4) 以降、5 ～ 60 秒の遅延後に再起動を有効にする **reload timer** コマンドがサポートされています。

コンフィギュレーション ファイルの管理

ここでは、コンフィギュレーション ファイルの管理方法について説明します。

実行コンフィギュレーションのスタートアップコンフィギュレーションへの保存

実行コンフィギュレーションをスタートアップ コンフィギュレーションに保存することで、次にデバイスをリロードするときのために変更を保存できます。

SUMMARY STEPS

1. (Optional) **show running-config**
2. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **show running-config** **Example:** `switch# show running-config`	実行設定を表示します。
ステップ 2	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

リモート サーバへのコンフィギュレーション ファイルのコピー

内部メモリに保存されたコンフィギュレーション ファイルをリモート サーバにコピーして、バックアップとして使用したり、他の Cisco NX-OS デバイスを設定するために使用したりすることができます。

SUMMARY STEPS

1. **copy scheme***running-config***server***://*[**url** /]**filename** /
2. **copy scheme***startup-config***server***://*[**url** /]**filename** /

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**copy scheme***running-config***server***://*[**url** /]**filename** / **Example:** `switch# copy running-config tftp://10.10.1.1/sw1-run-config.bak`	リモート サーバへ実行コンフィギュレーション ファイルをコピーします。 *scheme* 引数に対して、 **tftp:** を入力するか、**ftp:**、**scp:**、または **sftp:** を入力します。*server* 引数は、リモート サーバのアドレスまたは名前であり、*url* 引数はリモート サーバにあるソース ファイルへのパスです。 *server*、*url*、および *filename* の各引数は、大文字小文字を区別して入力します。
ステップ 2	**copy scheme***startup-config***server***://*[**url** /]**filename** / **Example:** `switch# copy startup-config tftp://10.10.1.1/sw1-start-config.bak`	スタートアップ コンフィギュレーション ファイルをリモート サーバにコピーします。 *scheme* 引数に対して、 **tftp:** を入力するか、**ftp:**、**scp:**、または **sftp:** を入力します。*server* 引数は、リ

	Command or Action	Purpose
		モート サーバのアドレスまたは名前であり、*url* 引数はリモート サーバにあるソース ファイルへのパスです。 *server*、*url*、および *filename* の各引数は、大文字小文字を区別して入力します。

Example

次に、リモート サーバーへ構成ファイルをコピーする例を示します。

```
switch# copy running-config
tftp://10.10.1.1/sw1-run-config.bak
switch# copy startup-config
tftp://10.10.1.1/sw1-start-config.bak
```

リモート サーバからの実行コンフィギュレーションのダウンロード

別の Cisco NX-OS デバイスで作成し、リモート サーバにアップロードしたコンフィギュレーション ファイルを使用して、Cisco NX-OS デバイスを設定できます。このファイルを、リモート サーバから TFTP、FTP、Secure Copy（SCP）を使用してデバイスに、または Secure Shell FTP（SFTP）を使用して実行コンフィギュレーションにダウンロードします。

Before you begin

ダウンロードするコンフィギュレーション ファイルが、リモート サーバの正しいディレクトリにあることを確認します。

ファイルに対する許可が正しく設定されていることを確認します。ファイルのアクセス権は、誰でも読み取り可能に設定されている必要があります。

お使いのデバイスからリモート サーバーへのルートを確保します。サブネット間でトラフィックをルーティングするルータまたはデフォルト ゲートウェイがない場合、お使いのデバイスtおリモート サーバーは同じサブネットワーク内にある必要があります。

pingを使用してリモート サーバへの接続を確認します。 または **ping6** を参照してください。

SUMMARY STEPS

1. **copy** *scheme***://***server*/[*url*/]*filename* **running-config**
2. (Optional) **show running-config**
3. (Optional) **copy running-config startup-config**
4. (Optional) **show startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**copy** *scheme*:*//server/*[*url/*]*filename* **running-config** **Example:** `switch# copy tftp://10.10.1.1/my-config running-config`	リモート サーバから実行コンフィギュレーション ファイルをダウンロードします。 *scheme* 引数に対して、 **tftp:** を入力するか、**ftp:**、**scp:**、または **sftp:** を入力します。*server* 引数は、リモート サーバのアドレスまたは名前であり、*url* 引数はリモート サーバにあるソース ファイルへのパスです。 *server*、*url*、および *filename* の各引数は、大文字小文字を区別して入力します。
ステップ 2	(Optional) **show running-config** **Example:** `switch# show running-config`	実行設定を表示します。
ステップ 3	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。
ステップ 4	(Optional) **show startup-config** **Example:** `switch# show startup-config`	スタートアップ コンフィギュレーションを表示します。

Related Topics

ファイルのコピー （161 ページ）

リモート サーバからのスタートアップ コンフィギュレーションのダウンロード

別の Cisco NX-OS デバイスで作成し、リモート サーバにアップロードしたコンフィギュレーション ファイルを使用して、Cisco NX-OS デバイスを設定できます。このファイルを、リモート サーバから TFTP、FTP、Secure Copy（SCP）を使用してデバイスに、または Secure Shell FTP（SFTP）を使用してスタートアップ コンフィギュレーションにダウンロードします。

Caution この手順を実行すると、Cisco NX-OS デバイス上のすべてのトラフィックが中断されます。

Before you begin

コンソール ポート上のセッションにログインします。

ダウンロードするコンフィギュレーション ファイルが、リモート サーバの正しいディレクトリにあることを確認します。

ファイルに対する許可が正しく設定されていることを確認します。ファイルのアクセス権は、誰でも読み取り可能に設定されている必要があります。

お使いのデバイスからリモート サーバーへのルートを確保します。サブネット間でトラフィックをルーティングするルータまたはデフォルト ゲートウェイがない場合、お使いのデバイスtおリモート サーバーは同じサブネットワーク内にある必要があります。

pingを使用してリモート サーバへの接続を確認します。 または **ping6** を参照してください。

SUMMARY STEPS

1. **write erase**
2. **reload**
3. **copy** *scheme*:*//server/*[*url /*]*filename* **running-config**
4. **copy running-config startup-config**
5. (Optional) **show startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**write erase** **Example:** `switch# write erase`	スタートアップ コンフィギュレーション ファイルを削除します。

ステップ 2	**reload**
	Example:
	`switch# reload`
	`This command will reboot the system. (y/n)?  [n] y`
	`...`
	`Enter the password for "admin": <password>`
	`Confirm the password for "admin": <password>`
	`...`
	`Would you like to enter the basic configuration dialog (yes/no): n`
	`switch#`
		Cisco NX-OS デバイスをリロードします。
		Note
		デバイスを設定するために、セットアップユーティリティを使用しないでください。

ステップ 3	**copy** *scheme*:*//server/*[*url /*]*filename* **running-config** **Example:** `switch# copy tftp://10.10.1.1/my-config running-config`	リモート サーバから実行コンフィギュレーション ファイルをダウンロードします。 *scheme* 引数に対して、 **tftp:** を入力するか、**ftp:**、**scp:**、または **sftp:** を入力します。*server* 引数は、リ

	Command or Action	Purpose
		モート サーバのアドレスまたは名前であり、*url* 引数はリモート サーバにあるソース ファイルへのパスです。 *server*、*url*、および *filename* の各引数は、大文字小文字を区別して入力します。
ステップ 4	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーション ファイルをスタートアップ コンフィギュレーション ファイルとして保存します。
ステップ 5	(Optional) **show startup-config** **Example:** `switch# show startup-config`	実行設定を表示します。

Related Topics

ファイルのコピー （161 ページ）

外部フラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコピー

後で使用するために、コンフィギュレーションファイルをバックアップとして外部フラッシュ メモリ デバイスにコピーできます。

Before you begin

外部フラッシュ メモリ デバイスを、アクティブなスーパーバイザ モジュールに挿入します。

SUMMARY STEPS

1. (Optional) **dir** {**usb1:** | **usb2:**}[*directory/*]
2. **copy running-config** {**usb1:** | **usb2:**}[*directory/*]*filename*
3. **copy startup-config** {**usb1:** | **usb2:**}[*directory/*]*filename*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **dir** {**usb1:**	**usb2:**}[*directory/*] **Example:** `switch# dir usb1:`	外部フラッシュ メモリ デバイス上のファイルを表示します。

	Command or Action	Purpose
ステップ 2	**copy running-config** {**usb1:** | **usb2:**}[*directory/*]*filename* **Example:** `switch# copy running-config usb1:dsn-running-config.cfg`	外部フラッシュ メモリ デバイスに実行コンフィギュレーションをコピーします。*filename* の引数では大文字と小文字が区別されます。
ステップ 3	**copy startup-config** {**usb1:** | **usb2:**}[*directory/*]*filename* **Example:** `switch# copy startup-config usb1:dsn-startup-config.cfg`	外部フラッシュ メモリ デバイスにスタートアップ コンフィギュレーションをコピーします。*filename* の引数では大文字と小文字が区別されます。

Related Topics

ファイルのコピー （161 ページ）

外部フラッシュ メモリ デバイスからの実行構成のコピー

別の Cisco NX-OS デバイスで作成し、外部フラッシュ メモリ デバイスに保存された構成ファイルをコピーすることで、デバイスを構成できます。

Before you begin

外部フラッシュ メモリ デバイスを、アクティブなスーパーバイザ モジュールに挿入します。

SUMMARY STEPS

1. (Optional) **dir** {**usb1:** | **usb2:**}[*directory/*]
2. **copy** {**usb1:** | **usb2:**}[*directory/*]*filename* **running-config**
3. (Optional) **show running-config**
4. (Optional) **copy running-config startup-config**
5. (Optional) **show startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **dir** {**usb1:** | **usb2:**}[*directory/*] **Example:** `switch# dir usb1:`	外部フラッシュ メモリ デバイス上のファイルを表示します。
ステップ 2	**copy** {**usb1:** | **usb2:**}[*directory/*]*filename* **running-config** **Example:** `switch# copy usb1:dsn-config.cfg running-config`	外部フラッシュ メモリ デバイスから実行コンフィギュレーションをコピーします。*filename* の引数では大文字と小文字が区別されます。

	Command or Action	Purpose
ステップ 3	(Optional) **show running-config** **Example:** `switch# show running-config`	実行設定を表示します。
ステップ 4	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。
ステップ 5	(Optional) **show startup-config** **Example:** `switch# show startup-config`	スタートアップ コンフィギュレーションを表示します。

Related Topics

ファイルのコピー （161 ページ）

外部フラッシュ メモリ デバイスからのスタートアップ構成のコピー

デバイス上のスタートアップ構成を復元するには、外部フラッシュ メモリ デバイスに保存された新しいスタートアップ構成ファイルをダウンロードします。

Before you begin

外部フラッシュ メモリ デバイスを、アクティブなスーパーバイザ モジュールに挿入します。

SUMMARY STEPS

1. (Optional) **dir** {**usb1:** | **usb2:**}[*directory/*]
2. **copy** {**usb1:** | **usb2:**}[*directory /*]*filename* **startup-config**
3. (Optional) **show startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **dir** {**usb1:**	**usb2:**}[*directory/*] **Example:** `switch# dir usb1:`	外部フラッシュ メモリ デバイス上のファイルを表示します。
ステップ 2	**copy** {**usb1:**	**usb2:**}[*directory /*]*filename* **startup-config** **Example:** `switch# copy usb1:dsn-config.cfg startup-config`	外部フラッシュ メモリ デバイスからスタートアップ コンフィギュレーションをコピーします。*filename* の引数では大文字と小文字が区別されます。

	Command or Action	Purpose
ステップ 3	(Optional) **show startup-config** **Example:** `switch# show startup-config`	スタートアップ コンフィギュレーションを表示します。

Related Topics

ファイルのコピー （161 ページ）

内部ファイルシステムへのコンフィギュレーションファイルのコピー

後で使用するために、コンフィギュレーション ファイルをバックアップとして内部メモリにコピーできます。

SUMMARY STEPS

1. **copy running-config** [*filesystem*:][*directory/*] | [*directory/*]*filename*
2. **copy startup-config** [*filesystem*:][*directory/*] | [*directory/*]*filename*

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**copy running-config** [*filesystem*:][*directory/*]	[*directory/*]*filename* **Example:** `switch# copy running-config bootflash:sw1-run-config.bak`	実行コンフィギュレーション ファイルを内部メモリにコピーします。 *filesystem*、*directory*、および *filename* の各引数では、大文字と小文字が区別されます。
ステップ 2	**copy startup-config** [*filesystem*:][*directory/*]	[*directory/*]*filename* **Example:** `switch# copy startup-config bootflash:sw1-start-config.bak`	スタートアップ コンフィギュレーション ファイルを内部メモリにコピーします。 *filesystem*、*directory*、および *filename* の各引数では、大文字と小文字が区別されます。

Related Topics

ファイルのコピー （146 ページ）

以前の構成へのロールバック

メモリ破損などの障害が発生し、バックアップされたバージョンからコンフィギュレーションを復元することが必要になる場合があります。

Note **copy running-config startup-config** コマンドを実行するたびに、バイナリ ファイルが作成され、ASCII ファイルが更新されます。有効なバイナリ コンフィギュレーション ファイルを使用すると、ブート全体の時間が大幅に短縮されます。バイナリ ファイルはアップロードできませんが、その内容を使用して既存のスタートアップ コンフィギュレーションを上書きできます。この項で説明している **write erase** コマンドがバイナリ ファイルをクリアします。

SUMMARY STEPS

1. **write erase**
2. **reload**
3. **copy** *configuration-file* **running-configuration**
4. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**write erase** **Example:** `switch# write erase`	スイッチの現在のコンフィギュレーションを削除します。
ステップ 2	**reload** **Example:** `switch# reload`	デバイスを再起動します。デバイスを起動して実行できるように、nx-os イメージ ファイルを提供するように求められます。
ステップ 3	**copy** *configuration-file* **running-configuration** **Example:** `switch# copy bootflash:start-config.bak running-configuration`	以前に保存されたコンフィギュレーション ファイルを実行コンフィギュレーションにコピーします。 **Note** *configuration-file* ファイル名引数では、大文字と小文字が区別されます。
ステップ 4	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします。

存在しないモジュールのコンフィギュレーションの削除

シャーシからI/Oモジュールを取り外す場合は、実行コンフィギュレーションからそのモジュールのコンフィギュレーションを削除することもできます。

Note シャーシの空のスロットに対するコンフィギュレーションのみを削除できます。

Before you begin

シャーシから I/O モジュールを取り外します。

SUMMARY STEPS

1. (Optional) **show hardware**
2. **purge module** *slot* **running-config**
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	(Optional) **show hardware** **Example:** `switch# show hardware`	デバイスに取り付けられたハードウェアを表示します。
ステップ 2	**purge module** *slot* **running-config** **Example:** `switch# purge module 3 running-config`	実行コンフィギュレーションから存在しないモジュールのコンフィギュレーションを削除します。
ステップ 3	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

構成の削除

デバイス上の構成を削除して、工場出荷時のデフォルト値に戻すことができます。「構成」は、「show startup」で表示されるスタートアップ構成を指します。他の内部アプリケーションまたはプロセスの状態はクリアされません。

構成の削除機能は、Nexus 9200-X、Nexus 9300-EX、-FX、-FX2、-FX3、および Nexus 9500 シリーズ スイッチでサポートされています。

デバイス上の永続メモリに保存された次のコンフィギュレーションファイルを削除できます。

- 新興企業
- Boot
- [デバッグ（Debug）]

write erase コマンドを使用すると、次のものを除くすべてのスタートアップ構成が削除されます。

- ブート変数定義
- 次のものを含む mgmt0 インターフェイス上の IPv4 および IPv6 構成：
  - アドレス
  - サブネット マスク
  - 管理 VRF のデフォルト ゲートウェイ/ルート

mgmt0 インターフェイスのブート変数定義と IPv4 構成を削除するには、**write erase boot** コマンドを使用します。パッチ rpm、サードパーティの rpm、構成以外の /etc ディレクトリ内のアプリケーション構成など、すべてのアプリケーション永続性ファイルを削除するには、「install reset」を使用します。このコマンドは、7.0(3)I6(1) リリースで追加されました。

Note 管理 VRF に複数の IPv6 デフォルト ルートが存在する場合、「copy r s」を使用する直前に管理 VRF の **show ipv6 static-route** コマンドで最初に表示されたデフォルト ルートは、**write erase** および **reload** 後に復元されます。

Note **write erase**を入力した後 コマンドで、ブレークアウト構成を適用するには、ASCII 構成を 2 回リロードする必要があります。

SUMMARY STEPS

1. **write erase** [**boot** | **debug**]

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**write erase** [**boot**	**debug**]
		Example:
		`switch# write erase`
		`Warning: This command will erase the startup-configuration.`
		`Do you wish to proceed anyway? (y/n) [n] y`
			永続メモリのコンフィギュレーションを削除します。デフォルトのアクションにより、スタートコンフィギュレーションが削除されます。
			The **boot** オプションを使用すると、mgmt0 インターフェイスのブート変数定義と IPv4 構成が削除されます。
			The **debug** オプションを使用すると、デバッグ構成が削除されます。

	Command or Action	Purpose
		Note mgmt0 インターフェイスで複数の IPv6 アドレスを設定すると、**show ipv6 interface <intf>**コマンドで「copy r s」を使用する前に最初に表示される IPv6 アドレスは、**書き込み消去**および**リロード**時に復元されます。 **Note** 実行構成ファイルは、このコマンドによって影響を受けません。

非アクティブなコンフィギュレーションのクリア

非アクティブな QoS と ACL のコンフィギュレーションの両方またはいずれか一方をクリアできます。

手順の概要

1. （任意） **show running-config** タイプ **inactive-if-config**
2. **clear inactive-config** *policy*
3. （任意） **show inactive-if-config log**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	（任意） **show running-config** タイプ **inactive-if-config** 例： `# show running-config ipqos inactive-if-config`	非アクティブなアクセス制御リスト（ACL）または Quality of Service（QoS）の構成を表示します。 *type* 引数の値は、**aclmgr** および **ipqos** です。 • **aclmgr**：aclmgr の非アクティブな構成を表示します。 • **ipqos**：qosmgr の非アクティブな構成を表示します。

ステップ 2	**clear inactive-config** *policy*
	例：
	`# clear inactive-config qos`
	`clear qos inactive config`
	`Inactive if config for QoS manager is saved at/bootflash/qos_inactive_if_config.cfg for vdc`
		非アクティブなコンフィギュレーションをクリアします。
		policy 引数の値は、**qos** および **acl** です。
		次に、値について説明します。

	コマンドまたはアクション	目的
	`default` `you can see the log file @ show inactive-if-config log`	• **qos**：非アクティブなQoS構成をクリアします。 • **acl**：非アクティブなACL構成をクリアします。 • **acl qos**：非アクティブなACL構成および非アクティブなQoS構成をクリアします。
ステップ 3	（任意） **show inactive-if-config log** 例： `# show inactive-if-config log`	非アクティブなコンフィギュレーションをクリアするのに使用されたコマンドを表示します。

構成のアーカイブと構成ログ

このセクションには、構成アーカイブと構成ログに関する情報が含まれています。

構成アーカイブの詳細

構成アーカイブは**configure replace**コマンドにより提供される構成のロールバック機能を強化するために、構成ファイルのアーカイブの保存、整理、管理を行うことを目的としたメカニズムです。構成アーカイブの導入前にも、実行構成のコピーを**copy running-config** *destination-url* コマンドを使用して保存し、ローカルやリモートに置換ファイルを保管できました。ただし、この方法ではファイルの自動管理を行うことはできませんでした。一方、構成の置換とロールバック機能では、実行中の構成のコピーを構成アーカイブに自動的に保存する機能を備えています。アーカイブされたファイルは構成のチェックポイントとして参照することができ、**configure replace** コマンドを使用して以前の構成状態に戻すために利用できます。

archive config コマンドを使用すると、Cisco IOS 構成を構成アーカイブに保存できます。その場合、標準のディレクトリとファイル名のプレフィクスが使用され、バージョン番号（およびオプションでタイムスタンプ）が自動的に付加されます。バージョン番号は連続したファイルを保存するごとに、1 つずつ大きくなります。この機能により、保存した構成ファイルを一貫して識別できます。アーカイブに保存する実行コンフィギュレーションの数は指定することができます。アーカイブ内のファイル数が上限値に達すると、次に最新のファイルが保存されるときに、最も古いファイルが自動的に消去されます。**show archive** コマンドを使用すると、コンフィギュレーション アーカイブに保存されているすべてのコンフィギュレーション ファイルに関する情報が表示されます。

構成ファイルを保存する構成アーカイブは、**configure replace** コマンドで使用することによって、ブートフラッシュ、FTP、TFTP のファイルシステム上に配置できます。

（注） この機能の TFTP および FTP は、VRF 管理を使用します。

設定アーカイブの特性の設定

archive config コマンドを使用する前に、コンフィギュレーション アーカイブを設定しておく必要があります。構成アーカイブの特性を構成するには、次の作業を実行します。

手順の概要

1. **configure terminal**
2. **archive**
3. **path** *url*
4. **maximum** *number*
5. **time-period** *minutes*
6. **write-memory**
7. **archive config**
8. （任意） **show archive log config all**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**archive** 例： `switch(config)# archive`	アーカイブ構成モードを開始します。 （注） このコマンドは、Cisco Nexus 9300-EX、-FX、および -R シリーズ スイッチに適用されません。
ステップ 3	**path** *url* 例： `switch(config-archive)# path bootflash:myconfig`	構成アーカイブに、ファイルのディレクトリとファイル名プレフィックスを指定します。 • ハードウェア プラットフォームによって、ファイル システムの名前は、例に示しているものとは異なる可能性があります。 （注） パスのところでファイルの代わりにディレクトリを指定する場合、ディレクトリ名は path flash:/*directory*/ のように後ろにスラッシュを付ける必要があります。このスラッシュはファイル名の後ろでは必要ありません。ディレクトリを指定する場合にだけ使います。

	コマンドまたはアクション	目的
ステップ 4	**maximum** *number* 例： `switch(config-archive)# maximum 14`	（任意）設定アーカイブに保存する実行設定のアーカイブ ファイルの最大数を指定します。 • *number*は構成アーカイブに保存できる実行構成のアーカイブ ファイル数の上限値を示します。指定できる範囲は 1 ～ 14 です。デフォルトは 10 です。 （注） このコマンドを使用する前に、**path** コマンドを設定して、構成アーカイブの位置とファイル名プレフィックスを指定しておく必要があります。
ステップ 5	**time-period** *minutes* 例： `switch(config-archive)# time-period 10`	（任意）コンフィギュレーションアーカイブに現在実行中のコンフィギュレーションのアーカイブ ファイルを自動保存する間隔を設定します。 • 設定アーカイブに現在の実行設定のアーカイブ ファイルをどれほどの頻度で自動保存するかを、*minutes* 引数により分単位で指定します。 （注） このコマンドを使用する前に、**path** コマンドを設定して、コンフィギュレーション アーカイブの位置とファイル名プレフィックスを指定しておく必要があります。
ステップ 6	**write-memory** 例： `switch(config-archive)# write-memory`	コマンドを有効にします。この機能はデフォルトではディセーブルになっています。このコマンドを入力すると、コマンド **copy r s** の実行時にアーカイブが発生します。
ステップ 7	**archive config** 例： `switch(config-archive)# archive config`	現在の実行設定ファイルを設定アーカイブに保存します。 （注） **archive config** コマンドを使用する前に、**path** コマンドを構成する必要があります。
ステップ 8	（任意） **show archive log config all** 例： `switch# show archive log config all`	すべてのユーザーの構成ログ エントリを表示します。

構成ログに関する情報

構成変更ログは、アカウンティング ログのデータを使用して、実行構成に加えられた変更を追跡します。この構成ログは、CLI または HTTP のみを介して開始される変更を追跡します。アクション ルーチンの呼び出しが発生する完全なコマンドが記録されます。次の種類の入力はログに記録されません。

- 結果的に構文エラー メッセージが表示されるコマンド
- デバイス ヘルプ システムを呼び出す一部のコマンド

この構成ログは、CLI または HTTP のみを介して開始される変更を追跡します。実行される各設定コマンドでは次の情報が記録されます。

- 設定変更のシーケンス番号
- コマンドが実行された行
- コマンドを実行したユーザーの名前
- 実行されたコマンド

show archive log config all コマンドを使用して、構成ログからの情報を表示できます。

実行される各設定コマンドでは次の情報が記録されます。

- 実行されたコマンド
- コマンドを実行したユーザーの名前
- 設定変更のシーケンス番号

show archive log config コマンドを使用して、構成ログからの情報を表示できます。

構成ログ エントリの表示

構成ログ エントリを表示するために、構成変更ログは **show archive log config all** コマンドを提供します。

手順の概要

1. switch# **show archive log config all**
2. switch# **show archive log config user** *username*
3. switch# **show archive log config user** *username* **first-index** *start-number* [**last-index** *end-number*]

手順の詳細

手順

ステップ 1 switch# **show archive log config all**

すべてのユーザーの構成ログ エントリを表示します。

例：

```
switch# show archive log config all

INDEX   LINE                    USER          LOGGED COMMAND
1       console0                user01       | logging console 1
2       console0                user01       | logging monitor 2
3       console0                user02       | system default switchport shutdown
4       console0                user02       | interface mgmt0
5       console0                user02       |  no shutdown
```

ステップ 2 switch# **show archive log config user** *username*

指定されたユーザー名の構成ログ エントリを表示します。

例：

次の例では、指定されたユーザー名の構成ログ エントリを表示します。

```
switch# show archive log config user user02

INDEX   LINE                    USER          LOGGED COMMAND
3       console0                user02       | system default switchport shutdown
4       console0                user02       | interface mgmt0
5       console0                user02       |  no shutdown
```

ステップ 3 switch# **show archive log config user** *username* **first-index** *start-number* [**last-index** *end-number*]

構成ログ エントリをインデックス番号で表示します。オプションの last-index を指定する場合、指定したユーザーの開始番号から終了番号までの範囲のインデックス番号を持つすべてのログエントリが表示されます。

例：

次の例では、ユーザー名 user02 のユーザーの構成ログ エントリ番号 4 および 5 を表示します。開始インデックスと終了インデックスの範囲は 1 ～ 2000000000 です。

```
switch# show archive log config user user02  first-index 4 last-index 5
Last Log cleared/wrapped time is : Wed Oct 19 00:53:08 2016


INDEX   LINE                    USER          LOGGED COMMAND
4       console0                user02       | interface mgmt0
5       console0                user02       |  no shutdown
```

デバイス コンフィギュレーションの確認

構成を確認するためには、次のいずれかのコマンドを使用します。

コマンド	目的
show running-config [**[exclude]** *command*] **[sanitized]**	現在の実行コンフィギュレーションまたはそのコンフィギュレーションのサブセットの内容を表示するには、該当するモードで **show running-config** コマンドを使用します。。 • **exclude**：（任意）特定のコンフィギュレーションを表示から除外します。 **exclude** キーワードのあとに *command* 引数を指定し、表示から特定のコンフィギュレーションを除外します。 • コマンド：（任意）1 つのコマンドのみを、または指定のコマンド ノード下で使用可能なコマンドのサブセットを表示します。 • **sanitized**：（任意）安全な配布と分析のためにサニタイズされたコンフィギュレーションを表示します。 Cisco NX-OS リリース 10.3(2)F 以降、**sanitized** キーワードが Cisco Nexus 9000 シリーズ スイッチでサポートされています。
show startup-config	スタートアップ コンフィギュレーションを表示します。 **Note** レイヤ 3 ベースの機能構成が running-config で無効になっている場合、**show startup-config** コマンドはそれらを表示しません。ただし、**copy running startup** コマンドが実行されるまで、構成はスタートアップ PSS にそのまま残ります。
show time-stamp running-config last-changed	実行構成が最後に変更されたときのタイムスタンプを表示します。

次に、**show running-config**コマンドで **sanitized** キーワードを指定した場合の出力例を示します。サニタイズされた構成は、構成の一部の詳細を公開せずに、構成を共有するために使用できます。

このオプションは、実行構成出力の機密ワードを<removed>キーワードによりマスクします。

```
switch# show running-config sanitized

!Command: show running-config sanitized
!Running configuration last done at: Wed Oct 12 09:14:54 2022
!Time: Wed Oct 12 13:52:55 2022

version 10.3(2) Bios:version 07.69

username admin password 5 <removed> role network-admin
```

```
copp profile strict
snmp-server user admin network-admin auth md5 <removed> priv aes-128 <removed>
localizedV2key
rmon event 1 log trap <removed> description FATAL(1) owner PMON@FATAL
rmon event 2 log trap <removed> description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 log trap <removed> description ERROR(3) owner PMON@ERROR
rmon event 4 log trap <removed> description WARNING(4) owner PMON@WARNING
rmon event 5 log trap <removed> description INFORMATION(5) owner PMON@INFO
--More--
```

コンフィギュレーション ファイルを使用した作業例

ここでは、コンフィギュレーション ファイルを使用した作業例を示します。

コンフィギュレーション ファイルのコピー

次に、NVRAM の既存のコンフィギュレーションの内容を上書きする例を示します。

```
switch# copy nvram:snapshot-config nvram:startup-config
Warning: this command is going to overwrite your current startup-config.
Do you wish to continue? {y/n} [y] y
```

Note このコマンドは、Cisco Nexus 9300-EX シリーズ スイッチに適用されません。

次に、bootflash: ファイル システムに実行コンフィギュレーションをコピーする例を示します。

```
switch# copy system:running-config bootflash:my-config
```

コンフィギュレーション ファイルのバックアップ

この例では、bootflash: ファイル システムにスタートアップ コンフィギュレーションをバックアップする方法を示します（ASCII ファイル）。

```
switch# copy startup-config bootflash:my-config
```

この例では、TFTP サーバにスタートアップ コンフィギュレーションをバックアップする方法を示します（ASCII ファイル）。

```
switch# copy startup-config tftp://172.16.10.100/my-config
```

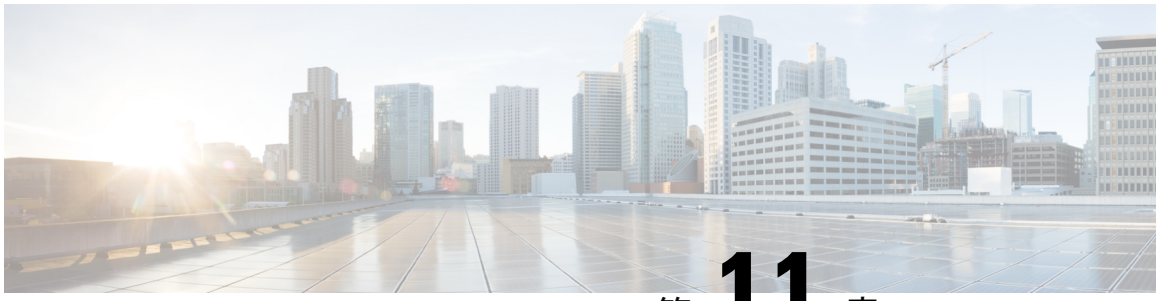
この例では、bootflash: ファイル システムに実行コンフィギュレーションをバックアップする方法を示します（ASCII ファイル）。

```
switch# copy running-config bootflash:my-config
```

以前の構成へのロールバック

現在のコンフィギュレーションを以前保存したコンフィギュレーションのスナップショットコピーにロールバックするには、次の手順を実行する必要があります。

1. **write erase** コマンドで、現在の実行イメージをクリアします を参照してください。
2. **reload** によりデバイスを再起動します を実行する前に、ユーザ名がフィギュレーションファイルに指定されていることを確認してください。
3. **copy** *configuration-file* **running-configuration** コマンドで、以前保存した構成ファイルを実行構成にコピーします。
4. **copy running-config startup-config** コマンドを使用して、実行構成をスタートアップ構成にコピーします。



第 11 章

Nexus Switch Intersight デバイス コネクタ

この章は、次の内容で構成されています。

Nexus Switch Intersight デバイス コネクタの概要

デバイスは、各システムの Cisco NX-OS イメージに組み込まれている Nexus Switch Intersight Device Connector（NXDC）を介して Intersight ポータルに接続されます。

Cisco NX-OS Release 10.2(3)F 以降、NX-OS 機能のデバイス コネクタは、接続されているデバイスに対して、セキュリティで保護されたインターネット接続を使用して情報を送信し、Cisco Intersight ポータルから制御命令を受信できる安全な方法を提供します。

NXDC は、すべての Cisco Nexus シリーズ スイッチでデフォルトで有効になっており、デフォルトで起動時に開始され、クラウドサービスへの接続を試みます。安全な接続が確立され、デバイス コネクタが Intersight サービスに登録されると、デバイス コネクタは詳細なインベントリ、正常性ステータスを収集し、採用テレメトリ データを Intersight データベース に送信します。インベントリは 1 日に 1 回更新されます。

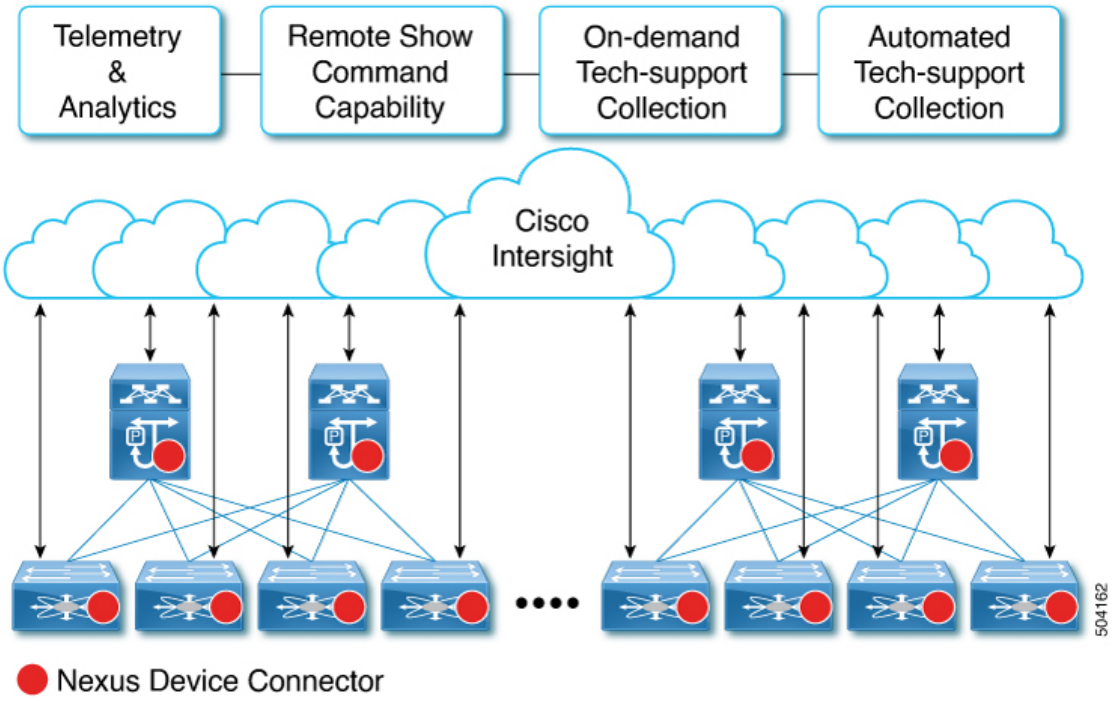
NXDC は Intersight に接続すると、Intersight サービスによる更新を介して、最新のバージョンに自動的に更新される AutoUpdate 機能をサポートします。

NXDC はまた、Connected TAC 機能をサポートして、要求されたデバイスからテクニカル サポート データを収集します。

NXDC機能の統合は、次の機能を持つスタンドアロンNexusスイッチの問題を解決するために行われました。

- スタンドアロンNexusスイッチから基本データを収集するための迅速かつ迅速なソリューションを提供します。
- クラウドにデータを安全に保管し、管理します。

- 将来の機能に柔軟に対応し、NXDC をアップグレードできるようにします。



注意事項と制約事項

NXDC の注意事項と制約事項は次のとおりです。

- DNS を設定する必要があります。
- svc.intersight.com が解決されること、ポート 443 でアウトバウンドで開始される HTTPS 接続が許可されていることを確認する必要があります。

  svc.intersight.com への HTTPS 接続にプロキシが必要な場合は、プロキシは NXDC ユーザーインターフェイスで構成できます。プロキシ設定については、NXDCの設定を参照してください。

Nexus スイッチの Intersight への設定

デフォルトでは、Nexus スイッチはシスコの Intersight への接続を試みます。Nexus デバイスが Intersight に到達できない場合は、Intersight の特定のプロキシを設定する必要があります。

（注） デフォルトでは、Intersight の機能（Nexus デバイス コネクタとも呼ばれる）が有効になっています。

Intersight の機能のオプションパラメータを設定するには、次の手順に従います。

手順の概要

1. **configure terminal**
2. （任意） **intersight proxy** <*proxy-name*> **port** <*proxy-port*>
3. （任意） **intersight use-vrf** *vrf-name*
4. （任意） **intersight trustpoint** <*trustpoint-label*> *[host-name]*
5. （任意） **intersight source-interface** <*interface*>
6. （任意） **no feature intersight**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	（任意） **intersight proxy** <*proxy-name*> **port** <*proxy-port*> 例： `switch(config)# intersight proxy proxy.esl.cisco.com port 8080`	Intersight接続用のプロキシ サーバーを設定します。 • *proxy-name* : プロキシ サーバーの IPv4 または IPv6 アドレスまたは DNS 名。 • *Proxy Port*：プロキシのポート番号を入力します。範囲は 1 ～ 65535 です。デフォルト値は 8080 です。 （注） Cisco Nexus スイッチのスマート ライセンス設定でプロキシが有効になっている場合、NXDC はこの設定を継承し、Cisco Intersight Cloud との接続を試みます。
ステップ 3	（任意） **intersight use-vrf** *vrf-name* 例： `switch(config)# intersight use-vrf blue`	指定されたVRF経由で接続する場合、NXDCのVRFを変更します。 （注） デフォルトでは、Intersightは管理VRF/ネームスペースで開始されます。
ステップ 4	（任意） **intersight trustpoint** <*trustpoint-label*> *[host-name]*	Intersight 接続の証明書を設定します。

	コマンドまたはアクション	目的
	例： `switch(config)# intersight trustpoint test test`	*trustpoint-label*：Crypto ca trustpoint ラベル。詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。
ステップ 5	（任意） **intersight source-interface** <*interface*> 例： `switch(config)# intersight source-interface mgmt 0`	通信のための送信元インターフェイスを設定します。
ステップ 6	（任意） **no feature intersight** 例： `switch(config)# no feature intersight`	Intersight プロセスを無効にし、すべての NXDC 設定とログ ストアを削除します。

NXDC 構成とステータスの確認

NXDC 構成を確認するには、次の Bash コマンドを使用します。

NXDC 構成とステータスを表示するには、次のコマンドのいずれかを入力します。

コマンド	目的
show system device-connector claim-info	デバイスのシリアル番号、トークン、および Intersight 要求の状態を表示します。 （注） トークンは、Intersight への接続が確立され、デバイスが要求されていない場合に表示されます。デバイスが要求されている場合には、トークンは表示されず、メッセージセクションに「すでに要求されているデバイスの要求コードは取得できません（Cannot fetch claim code for already claimed device）」と表示されます。 有効なトークンの期間が秒単位で報告されます。
show system device-connector log [dc|dcgrpc|cnmi|nae|sim|compliance]	デバイス コネクタのログ メッセージを表示します。

次に、デバイスが要求される前の show system device-connector claim-info コマンドの出力例を示します。

```
Switch# show system device-connector claim-info
SerialNumber: FDO23021ZUJ
SecurityToken: 9FFD4FA94DCD
Duration: 599
Message:
Claim state: Not Claimed
```

次に、デバイスが要求された後の show system device-connector claim-info コマンドの出力例を示します。

```
Switch# show system device-connector claim-info
SerialNumber: ABCD12345E6
SecurityToken:
Duration: 0
Message: Cannot fetch claim code for already claimed device
Claim state: Claimed
Claim time: 2024-02-18T12:00:01.77Z
Claimed by: user@cisco.com
Account: dc- customer
Site name:
Site ID:
```

Intersight での Nexus スイッチの要求

特長や機能の使用を開始するには、Intersightユーザーインターフェイス（UI）でスイッチを要求する必要があります。

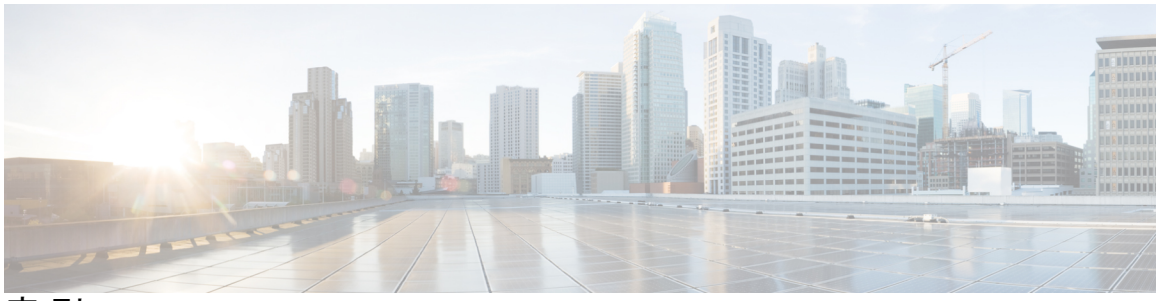
Intersight UI でスイッチを要求するには、次の手順に従います。

- Intersight UI を使用してNexusスイッチを要求します。

  Intersightで接続済みのデバイスを要求するには、ターゲットの要求で説明されているプロセスに従います。

- Ansible プレイブックを使用して複数の Nexus スイッチを要求します。

  Ansible を使用して複数の Nexus スイッチを自動で要求するには、 Ansible プレイブックに記載されている詳細を確認してください。



索 引

M

P

R

S

T

V

W

え

け

し

せ

そ

て

は

ふ

ろ

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。