



『Cisco Nexus 7000 シリーズ NX-OS 基本設定ガイド、リリース 6.x』

初版：2009年07月31日

最終更新：2013年08月21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。

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <http://www.cisco.com/go/trademarks>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2015 Cisco Systems, Inc. All rights reserved.



目次

はじめに **xiii**

対象読者 **xiii**

表記法 **xiii**

Cisco Nexus 7000 シリーズ NX-OS ソフトウェアの関連資料 **xv**

マニュアルに関するフィードバック **xvii**

マニュアルの入手方法およびテクニカル サポート **xvii**

新機能および変更された機能に関する情報 **1**

新機能および変更された機能に関する情報 **1**

概要 **3**

機能情報の確認 **3**

ソフトウェアの互換性 **3**

データセンター全体に共通のソフトウェア **4**

モジュラ式のソフトウェア設計 **6**

仮想デバイス コンテキスト **6**

サービスアビリティ **6**

スイッチド ポート アナライザ **6**

Ethalyzer **6**

Call Home **7**

オンライン診断 **7**

Embedded Event Manager **7**

NetFlow **7**

管理性 **7**

簡易ネットワーク管理プロトコル **8**

設定の確認およびロールバック **8**

ロールベース アクセス コントロール **8**

接続管理プロセッサ **8**

Cisco NX-OS デバイス コンフィギュレーション方式	8
トラフィックのルーティング、転送、および管理	9
イーサネット スイッチング	9
IP ルーティング	9
IP サービス	10
IP マルチキャスト	10
Quality of Service	11
ネットワーク セキュリティ	11
Cisco TrustSec	11
その他のネットワーク セキュリティ機能	11
ライセンス	12
サポートされる規格	13
Cisco NX-OS セットアップユーティリティの使用	19
機能情報の確認	19
Cisco NX-OS セットアップユーティリティに関する情報	19
セットアップユーティリティの前提条件	21
Cisco NX-OS デバイスの設定	21
セットアップユーティリティに関する追加情報	27
セットアップユーティリティの関連資料	27
PowerOn Auto Provisioning の使用方法	29
機能情報の確認	29
PowerOn Auto Provisioning の概要	29
POAP のためのネットワーク要件	30
POAP コンフィギュレーションスクリプト	31
POAP 処理	32
電源投入フェーズ	33
DHCP 検出フェーズ	34
スクリプトの実行フェーズ	35
インストール後のリロードフェーズ	36
POAP の注意事項および制約事項	36
POAP を使用するためのネットワーク環境の設定	37
POAP を使用するスイッチの設定	38
デバイス コンフィギュレーションの確認	38

コマンドライン インターフェイスの概要	39
機能情報の確認	40
CLI プロンプトの概要	40
コマンド モード	41
EXEC コマンド モード	41
グローバル コンフィギュレーション コマンド モード	41
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	41
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	42
コマンド モードの保存および復元	43
コンフィギュレーション コマンド モードの終了	43
コマンド モードの概要	44
特殊文字	46
キーストローク ショートカット	47
コマンドの省略形	49
部分的なコマンド名の補完	50
コマンド階層での現在の場所の識別	51
コマンドの no 形式の使用	51
CLI 変数の設定	52
CLI 変数について	52
CLI セッションのみの変数の設定	53
永続的な CLI 変数の設定	53
コマンド エイリアス	54
コマンド エイリアスについて	54
コマンド エイリアスの定義	55
ユーザセッションのコマンド エイリアスの設定	56
コマンド スクリプト	56
コマンド スクリプトの実行	56
端末への情報のエコー	57
コマンド処理の遅延	58
状況依存ヘルプ	58
正規表現の概要	60
特殊文字	60

複数文字のパターン	61
位置指定	61
show コマンドの出力の検索およびフィルタリング	61
キーワードのフィルタリングおよび検索	62
diff ユーティリティ	64
grep および egrep ユーティリティ	64
less ユーティリティ	65
sed ユーティリティ	65
sort ユーティリティ	66
--More-- プロンプトからの検索およびフィルタリング	67
コマンド履歴の使用	68
コマンドの呼び出し	68
CLI の履歴呼び出しの制御	69
CLI の編集モードの設定	69
コマンド履歴の表示	70
CLI の確認プロンプトのイネーブルまたはディセーブル	70
CLI の表示色の設定	71
モジュールへのコマンドの送信	71
BIOS ローダー プロンプト	72
CLI の使用例	72
コマンドエイリアスの定義	73
CLI セッション変数の使用	73
システム定義のタイムスタンプ変数の使用	74
コマンドスクリプトの実行	74
CLI に関する追加情報	75
CLI の関連資料	75
端末設定およびセッションの設定	77
機能情報の確認	77
端末設定とセッションの概要	77
端末セッションの設定	78
コンソール ポート	78
COM1 ポート	79

仮想端末	79
モデムのサポート	79
コンフィギュレーションファイルに対する仮想化のサポート	80
端末設定およびセッションのライセンス要件	80
コンソールポートの設定	81
COM1ポートの設定	82
仮想端末の設定	84
非アクティブセッションのタイムアウトの設定	84
セッション制限の設定	85
モデム接続の設定	86
モデム接続のイネーブル化	86
デフォルトの初期化文字列のダウンロード	87
ユーザ指定の初期化文字列の設定およびダウンロード	89
電源がオンになっている Cisco NX-OS デバイスのモデムの初期化	90
端末セッションのクリア	91
端末およびセッション情報の表示	91
ファイルシステムパラメータのデフォルト設定	92
端末設定およびセッションに関する追加情報	92
端末設定およびセッションの関連資料	92
基本的なデバイス管理	93
機能情報の確認	93
基本的なデバイス管理の概要	94
デバイスのホスト名	94
Message-of-the-Day バナー	94
デバイスクロック	94
Clock Manager	94
タイムゾーンと夏時間	94
ユーザセッション	95
基本的なデバイス管理に対する仮想化のサポート	95
基本的なデバイス管理のライセンス要件	95
デバイスのホスト名の変更	95
MOTD バナーの設定	96

タイムゾーンの設定	97
夏時間の設定	98
デバイスクロックの手動設定	99
Clock Manager の設定	100
ユーザの管理	101
ユーザセッションに関する情報の表示	101
ユーザへのメッセージ送信	101
デバイスコンフィギュレーションの確認	101
基本的なデバイスパラメータのデフォルト設定	102
基本的なデバイス管理に関する追加情報	102
基本的なデバイス管理の関連資料	102
基本的なデバイス管理機能の履歴	102
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	105
機能情報の確認	105
デバイスのファイルシステム、ディレクトリ、およびファイルの概要	106
ファイルシステム	106
ディレクトリ	107
ファイル	108
ファイルシステムに対する仮想化のサポート	108
ファイルシステム、ディレクトリ、およびファイルのライセンス要件	108
外部フラッシュデバイスのフォーマット	108
ディレクトリの操作	109
カレントディレクトリの識別	109
カレントディレクトリの変更	110
ディレクトリの作成	110
ディレクトリの内容の表示	111
ディレクトリの削除	111
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	112
ファイルの使用	112
ファイルの移動	112
ファイルのコピー	113
ファイルの削除	114

ファイルの内容の表示	114
ファイル チェックサム の表示	114
ファイルの圧縮および圧縮解除	115
ファイルの最後の行の表示	115
show コマンドの出力のファイルへのリダイレクト	116
ファイルの検索	116
アーカイブ ファイルの操作	117
アーカイブ ファイルの作成	117
アーカイブ ファイルへのファイルの追加	118
アーカイブ ファイルからのファイルの抽出	119
アーカイブ ファイル内のファイル名の表示	120
ファイル システムの使用例	120
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	120
ファイルの移動	121
ファイルのコピー	121
ディレクトリの削除	121
ファイルの内容の表示	122
ファイル チェックサム の表示	122
ファイルの圧縮および圧縮解除	123
show コマンドの出力のリダイレクト	123
ファイルの検索	124
ファイル システム パラメータのデフォルト設定	124
ファイル システムに関する追加情報	124
ファイル システムの関連資料	124
ファイル システムの機能の履歴	124
コンフィギュレーション ファイルの操作	127
機能情報の確認	127
コンフィギュレーション ファイルの概要	127
コンフィギュレーション ファイルのタイプ	128
コンフィギュレーション ファイルに対する仮想化のサポート	128
コンフィギュレーション ファイルのライセンス要件	128
コンフィギュレーション ファイルの管理	129

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	129
コンフィギュレーション ファイルのリモート サーバへのコピー	129
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	130
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	131
コンフィギュレーション ファイルの外部フラッシュ メモリ デバイスへのコピー	133
外部フラッシュ メモリ デバイスからの実行コンフィギュレーションのコピー	134
外部フラッシュ メモリ デバイスからのスタートアップ コンフィギュレーションのコピー	135
コンフィギュレーション ファイルの内部ファイル システムへのコピー	135
以前のコンフィギュレーションへのロールバック	136
検出できないモジュールの設定の削除	137
コンフィギュレーションの削除	138
非アクティブな設定のクリア	139
デバイス コンフィギュレーションの確認	140
コンフィギュレーション ファイルの操作例	141
コンフィギュレーション ファイルのコピー	141
コンフィギュレーション ファイルのバックアップ	141
以前のコンフィギュレーションへのロールバック	142
コンフィギュレーション ファイルに関する追加情報	142
コンフィギュレーション ファイルの関連資料	142
コンフィギュレーション ファイルの機能の履歴	142
tcl によるスクリプティング	145
機能情報の確認	145
tcl に関する情報	145
注意事項と制約事項	145
tclsh コマンドのヘルプ	146
tclsh コマンドの履歴	146
tclsh のタブ補完	146
tclsh の CLI コマンド	146

tclsh コマンドの区切り	147
tcl 変数	147
tclquit	147
tclsh のセキュリティ	148
tclsh コマンドの実行	148
tclsh のコマンドによる Cisco NX-OS のモードの制御	149
tcl の参照	150
Python API	151
機能情報の確認	151
Python API の概要	151
Python の使用	152
Python ライブラリのインポート	152
CLI コマンドの API の使用	152
CLI からの Python インタープリタの呼び出し	153
VRF の割り当ての変更	154
表示形式	154
Python インタープリタの非永続性	155
非インタラクティブ Python	156
バックグラウンドでのスクリプトの実行	156
スクリプトのオンラインヘルプ	157
オンラインヘルプのプロトコル	157



はじめに

ここでは、『Cisco Nexus 7000 シリーズ NX-OS Fundamentals コンフィギュレーションガイド』の対象読者、構成、および表記法について説明します。また、関連マニュアルの入手方法についても説明します。

- [対象読者, xiii ページ](#)
- [表記法, xiii ページ](#)
- [Cisco Nexus 7000 シリーズ NX-OS ソフトウェアの関連資料, xv ページ](#)
- [マニュアルに関するフィードバック, xvii ページ](#)
- [マニュアルの入手方法およびテクニカル サポート, xvii ページ](#)

対象読者

このマニュアルは、Cisco Nexus デバイスのコンフィギュレーションおよびメンテナンスを担当す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。
<i>italic</i>	イタリック体の文字は、ユーザが値を入力する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角カッコで囲んで示しています。

表記法	説明
[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、角カッコで囲み、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、波カッコで囲み、縦棒で区切って示しています。
[x {y z}]	角カッコまたは波カッコ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。角カッコ内の波カッコと縦棒は、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。イタリック体を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ません。引用符を使用すると、その引用符も含めてstringとみなされます。

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、screen 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。

このマニュアルでは、次の表記法を使用しています。



(注) 「注釈」です。役立つ情報やこのマニュアルに記載されていない参照資料を紹介しています。



注意 「要注意」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。

Cisco Nexus 7000 シリーズ NX-OS ソフトウェアの関連資料

Cisco Nexus 7000 シリーズ NX-OS 全体のマニュアルセットは、次の URL にあります。

http://www.cisco.com/en/us/products/ps9402/tsd_products_support_series_home.html

リリースノート

リリースノートは、次の URL から入手できます。

http://www.cisco.com/en/US/products/ps9402/prod_release_notes_list.html

コンフィギュレーションガイド

これらのマニュアルは、次の URL から入手できます。

http://www.cisco.com/en/US/products/ps9402/products_installation_and_configuration_guides_list.html

このカテゴリのマニュアルには、次が含まれます。

- 『Cisco Nexus 7000 Series NX-OS Configuration Examples』
- 『Cisco Nexus 7000 Series NX-OS FabricPath Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS IP SLAs Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Layer 2 Switching Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS LISP Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS MPLS Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Multicast Routing Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS OTV Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Quality of Service Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS SAN Switching Guide』
- 『Cisco Nexus 7000 Series NX-OS Security Configuration Guide』

- 『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Verified Scalability Guide』
- 『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Virtual Device Context Quick Start』
- 『Cisco Nexus 7000 Series NX-OS OTV Quick Start Guide』
- 『Cisco NX-OS FCoE Configuration Guide for Cisco Nexus 7000 and Cisco MDS 9500』
- 『Cisco Nexus 2000 Series Fabric Extender Software Configuration Guide』

コマンド リファレンス

これらのマニュアルは、次の URL から入手できます。

http://www.cisco.com/en/US/products/ps9402/prod_command_reference_list.html

このカテゴリのマニュアルには、次が含まれます。

- 『Cisco Nexus 7000 Series NX-OS Command Reference Master Index』
- 『Cisco Nexus 7000 Series NX-OS FabricPath Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference』
- 『Cisco Nexus 7000 Series NX-OS High Availability Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Interfaces Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference』
- 『Cisco Nexus 7000 Series NX-OS LISP Command Reference』
- 『Cisco Nexus 7000 Series NX-OS MPLS Configuration Guide』
- 『Cisco Nexus 7000 Series NX-OS Multicast Routing Command Reference』
- 『Cisco Nexus 7000 Series NX-OS OTV Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Quality of Service Command Reference』
- 『Cisco Nexus 7000 Series NX-OS SAN Switching Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Security Command Reference』
- 『Cisco Nexus 7000 Series NX-OS System Management Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Unicast Routing Command Reference』
- 『Cisco Nexus 7000 Series NX-OS Virtual Device Context Command Reference』
- 『Cisco NX-OS FCoE Command Reference for Cisco Nexus 7000 and Cisco MDS 9500』

その他のソフトウェアのマニュアル

これらのマニュアルは、以下のランディング ページから検索できます。

http://www.cisco.com/en/us/products/ps9402/tsd_products_support_series_home.html

- 『Cisco Nexus 7000 Series NX-OS MIB Quick Reference』
- 『Cisco Nexus 7000 Series NX-OS Software Upgrade and Downgrade Guide』
- 『Cisco Nexus 7000 Series NX-OS Troubleshooting Guide』
- 『Cisco NX-OS Licensing Guide』
- 『Cisco NX-OS System Messages Reference』
- 『Cisco NX-OS XML Interface User Guide』

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、次の宛先へご連絡ください。

- nexus7k-docfeedback@cisco.com

ご協力をよろしくお願いいたします。

マニュアルの入手方法およびテクニカル サポート

マニュアルの入手、Cisco Bug Search Tool (BST) の使用、サービス要求の送信、追加情報の収集の詳細については、『*What's New in Cisco Product Documentation*』を参照してください。このドキュメントは、<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html> から入手できます。

『*What's New in Cisco Product Documentation*』はシスコの新規および改訂版の技術マニュアルの一覧を提供するもので、RSS フィードとして購読できます。また、リーダー アプリケーションを使用すると、コンテンツがデスクトップに直接配信されるようになります。RSS フィードは無料のサービスです。



第 1 章

新機能および変更された機能に関する情報

この章では、『Cisco Nexus 7000 Series NX-OS Fundamentals Guide, Release 6.x』に記載されている新機能および変更された機能に関するリリース固有の情報について説明します。このマニュアルの最新バージョンは、次のシスコ Web サイトから入手できます。

http://www.cisco.com/en/US/products/ps9402/products_installation_and_configuration_guides_list.html

- [新機能および変更された機能に関する情報, 1 ページ](#)

新機能および変更された機能に関する情報

Cisco NX-OS Release 6.x に関するその他の情報については、『Cisco Nexus 7000 Series NX-OS Release Notes, Release 6.x』を参照してください。このドキュメントは、次のシスコ Web サイトから入手できます。

http://www.cisco.com/en/US/products/ps9402/prod_release_notes_list.html

この表は『Cisco Nexus 7000 シリーズ NX-OS 基本設定ガイド、リリース 6.x』の新機能および変更された機能を示します。

下の表に、このマニュアルの新機能および変更された機能を要約し、各機能がサポートされているリリースを示します。ご使用のソフトウェアリリースで、本書で説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェアリリースのリリース ノートを参照してください。

表 1: リリース 6.x の新機能および機能変更

機能	説明	変更されたリリース
デフォルト VRF の Python 操作	Python ネットワーク操作を管理 VRF からデフォルト VRF に変更します	6.2(8)

機能	説明	変更されたリリース
Python を持つ JSON モジュール	Python を持つ JSON モジュールをサポートします。	6.2(8)
Python を持つロギング モジュール	Python を持つロギング モジュールをサポートします。	6.2(8)
Python を持つ hashlib モジュール	Python を持つ hashlib モジュールをサポートします。	6.2(8)
PowerOn 自動プロビジョニング (POAP) のサポート	ソフトウェアイメージのアップグレードプロセスを自動化し、Cisco Nexus スイッチにコンフィギュレーションファイルをインストールします。	6.1(2)



第 2 章

概要

この章では、Cisco NX-OS ソフトウェアの概要について説明します。

- 機能情報の確認, 3 ページ
- ソフトウェアの互換性, 3 ページ
- サービスアビリティ, 6 ページ
- 管理性, 7 ページ
- トラフィックのルーティング、転送、および管理, 9 ページ
- Quality of Service, 11 ページ
- ネットワーク セキュリティ, 11 ページ
- ライセンス, 12 ページ
- サポートされる規格, 13 ページ

機能情報の確認

ご使用のソフトウェアリリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「新機能および変更された機能に関する情報」の章を参照してください。

ソフトウェアの互換性

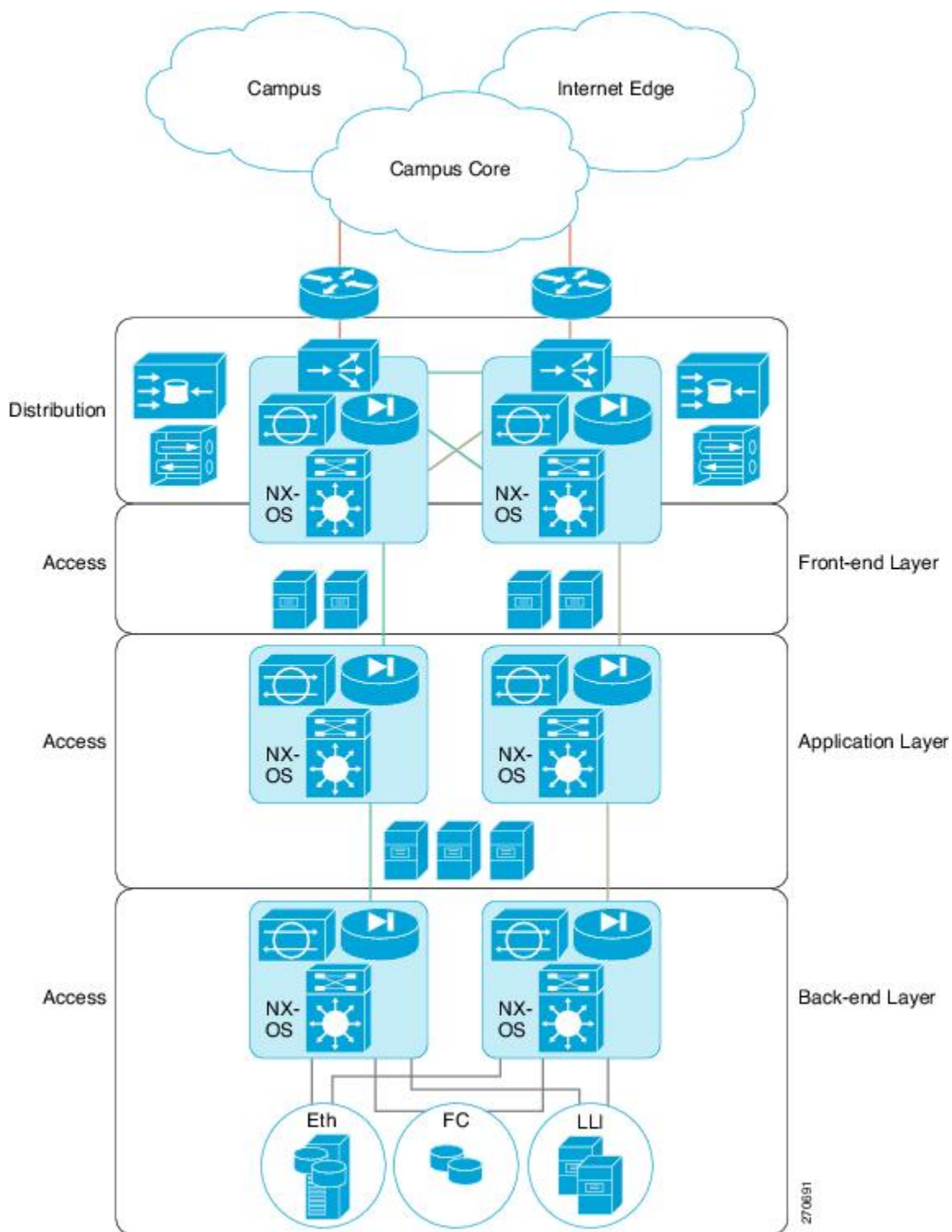
Cisco NX-OS ソフトウェアは、Cisco IOS ソフトウェアのどのバリエーションを実行するシスコ製品とも相互運用できます。また、Cisco NX-OS ソフトウェアは、IEEE および RFC 準拠標準に適合するどのネットワークング オペレーティング システムとも相互運用できます。

データセンター全体に共通のソフトウェア

Cisco NX-OS ソフトウェアは、統合オペレーティング システムとして、LAN およびレイヤ 4～7 のネットワーク サービスなど、データセンターネットワークの全領域において実行できるように設計されています。

この図は、データセンターの Cisco NX-OS ソフトウェアの概要を示します。

図 1：データセンターの Cisco NX-OS



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。

仮想デバイス コンテキスト

Cisco NX-OS ソフトウェアは、システムおよびハードウェア リソースを、仮想デバイスをエミュレートする仮想コンテキストにセグメント化できます。各仮想デバイスコンテキスト (VDC) には、独自のソフトウェアプロセス、専用のハードウェアリソース (インターフェイス)、および独立した管理環境があります。VDCを使用すると、個別のネットワークを共通のインフラストラクチャに統合できます。このインフラストラクチャでは、物理的に分離されたネットワークの管理境界の分離と障害の分離の特性が維持され、単一インフラストラクチャの運用コストのメリットの多くが提供されます。詳細については、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』を参照してください。

サービスアビリティ

Cisco NX-OS ソフトウェアには、デバイスがネットワークのトレンドやイベントに対応できるサービスアビリティ機能が組み込まれています。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 7000 Series NX-OS System Management Configuration Guide』を参照してください。

Ethalyzer

Ethalyzer 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。Ethalyzer 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。Ethalyzer 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 7000 Series NX-OS Troubleshooting Guide』を参照してください。

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用意されており、ポケットベル サービス、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に対応します。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。この機能を利用すると、たとえばネットワーク サポート技術者を直接ポケットベルで呼び出したり、Eメールメッセージをネットワーク オペレーションセンター (NOC) に送信したり、Cisco AutoNotify サービスを使用して直接 Cisco Technical Assistance Center (TAC) でケースを生成したりすることができます。Call Home の詳細については、『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x』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モニタできます。GOLD の設定の詳細については、『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x』を参照してください。

Embedded 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 の設定の詳細については、『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x』を参照してください。

NetFlow

Cisco NX-OS の NetFlow の実装では、バージョン 5 およびバージョン 9 のエクスポートをサポートしています。また、拡張性を高めるための Flexible NetFlow 設定モデルおよびハードウェアベースでの Sampled NetFlow もサポートしています。NetFlow の詳細については、『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x』を参照してください。

管理性

ここでは、Cisco NX-OS ソフトウェアの管理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 7000 Series NX-OS System Management Configuration Guide, Release 6.x*』を参照してください。

設定の確認およびロールバック

Cisco NX-OS ソフトウェアでは、設定をコミットする前に、設定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。デバイスを事前に設定し、確認した設定を後から適用することができます。設定には、必要に応じて、既知の良好な設定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 7000 Series NX-OS System Management Configuration Guide, Release 6.x*』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 の詳細については、『*Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x*』を参照してください。

接続管理プロセッサ

Cisco NX-OS ソフトウェアは、リモートプラットフォームの管理用に Connectivity Management Processor (CMP) の使用をサポートします。CMP は、Cisco NX-OS コンソールにアウトオブバンドアクセスチャネルを提供します。CMP の詳細については、『*Cisco Nexus 7000 Series Connectivity Management Processor Configuration Guide*』を参照してください。

Cisco NX-OS デバイス コンフィギュレーション方式

Secure Shell (SSH; セキュア シェル) セッションまたは Telnet セッションからコマンドラインインターフェイス (CLI) を使用してデバイスを設定できます。SSH を使用するとデバイスに安全に接続できます。CLI コンフィギュレーションガイドとコマンドリファレンス、機能別に構成されています。詳細は、次の Cisco NX-OS コンフィギュレーションガイドおよび Cisco NX-OS コマンドリファレンスを参照してください。SSH および Telnet の詳細については、『*Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x*』を参照してください。

また、XML 管理インターフェイスを使用してデバイスを構成できます。これは、CLI を補完する NETCONF プロトコルに基づくプログラマ的な方式です。詳細については、『*Cisco NX-OS XML Interface User Guide*』を参照してください。

トラフィックのルーティング、転送、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、高性能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 Rapid Spanning Tree Protocol (RSTP; 高速スパニングツリープロトコル) および多重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
- 16,000 サブスクライバ VLAN
- IEEE 802.3ad リンク アグリゲーション
- プライベート VLAN
- クロスシャーシのプライベート VLAN
- 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 7000 Series NX-OS Interfaces Configuration Guide, Release 6.x』および『Cisco Nexus 7000 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
- Border Gateway Protocol (BGP)
- EIGRP (Enhanced Interior Gateway Routing Protocol)
- Routing Information Protocol Version 2 (RIPv2)

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、トンネル

インターフェイス、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 7000 Series NX-OS Unicast Routing Configuration Guide, Release 6.x』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。

- 仮想ルーティングおよびフォワーディング (VRF)
- Dynamic Host Configuration Protocol (DHCP) ヘルパー
- ホットスタンバイルーティングプロトコル (HSRP)
- Gateway Load Balancing Protocol (GLBP)
- 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 OPSFv3 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 7000 Series NX-OS Unicast Routing Configuration Guide, Release 6.x』を参照してください。

IP マルチキャスト

Cisco NX-OS ソフトウェアには、次のマルチキャストプロトコルと機能が用意されています。

- Protocol Independent Multicast (PIM) Version 2 (PIMv2)
- Source Specific Multicast (SSM)
- PIM スパースモード (IPv4 および IPv6 の Any-Source マルチキャスト (ASM))



(注) Cisco NX-OS ソフトウェアでは、PIM デンスモードをサポートしていません。

- Bidirectional Protocol Independent Multicast (Bidir PIM)
- Anycast ランデブーポイント (Anycast-RP)
- IPv4 および IPv6 のマルチキャスト NSF
- ブートストラップルータ (BSR) を使用する RP-Discovery (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
- Multicast Listener Discovery (MLD) Protocol Version 2 (IPv6 用)
- Multicast Source Discovery Protocol (MSDP) (IPv4 のみ)

詳細については、『Cisco Nexus 7000 Series NX-OS Multicast Routing Command Reference』を参照してください。

Quality of Service

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設定を行うことができます。詳細については、『Cisco Nexus 7000 Series NX-OS Quality of Service Configuration Guide, Release 6.x』を参照してください。

ネットワーク セキュリティ

ここでは、Cisco NX-OS ソフトウェアでサポートされているネットワーク セキュリティ機能について説明します。

Cisco TrustSec

Cisco TrustSec セキュリティは、データの機密性と整合性を確保し、128 ビット AES (Advanced Encryption Standard) 暗号方式による標準の IEEE 802.1AE リンク レイヤ暗号化をサポートします。リンク レイヤ暗号化は、エンドツーエンドのデータプライバシーを保証しながら、暗号化されたパスに沿ってセキュリティ サービス デバイスを挿入できるようにします。Cisco TrustSec はセキュリティ グループ アクセス コントロール リスト (SGACL) を使用します。これは IP アドレスではなくセキュリティ グループ タグに基づきます。SGACL では、そのトポロジの独立性のために、より簡潔で、より管理しやすいポリシーが可能になります。詳細については、『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x』を参照してください。

その他のネットワーク セキュリティ機能

Cisco NX-OS ソフトウェアには、Cisco TrustSec の他に次のセキュリティ機能が組み込まれています。

- プロトコル適合チェックのためのデータ パス侵入検知システム (IDS)
- 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

- ダイナミック アドレス解決プロトコル (ARP) インспекション (DAI) 、DHCP スヌーピング、および IP ソース ガードなどの、Cisco 統合セキュリティ機能
- AAA (認証、許可、およびアカウンティング)
- RADIUS および TACACS+
- SSH プロトコルバージョン 2
- SNMPv3
- ポートセキュリティ
- IEEE 802.1x 認証
- レイヤ 2 Cisco Network Admission Control (NAC) LAN ポート IP
- 名前付き ACL でサポートされている MAC アドレスおよび IPv4 アドレスに基づくポリシー (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))
- 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)
- ユニキャスト Reverse Path Forwarding (ユニキャスト RPF)

詳細については、『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x』を参照してください。

ライセンス

Cisco NX-OS ソフトウェアでは、デバイスの高度な機能を使用する場合は、その機能に対応するライセンスをインストールする必要があります。ライセンスパッケージに含まれていない機能は、Cisco NX-OS ソフトウェアにバンドルされており、追加費用は一切発生しません。

各デバイス用のライセンスを購入してインストールする必要があります。



- (注) Cisco TrustSec 機能以外は、ライセンスをインストールせずに機能をイネーブルにできます。Cisco NX-OS ソフトウェアには、ライセンスを購入する前に機能を試すことができる猶予期間があります。Cisco TrustSec 機能をイネーブルにするには、Advanced Services ライセンスパッケージをインストールする必要があります。

Cisco NX-OS ソフトウェア ライセンスの詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

ライセンスの問題のトラブルシューティングについては、『Cisco Nexus 7000 Series NX-OS Troubleshooting Guide』を参照してください。

サポートされる規格

次の表に、IEEE 準拠標準を示します。

表 2: IEEE 準拠標準

標準	説明
802.1D	MAC ブリッジ
802.1s	多重スパンニングツリー プロトコル
802.1w	高速スパンニングツリー プロトコル
802.1AE	MAC セキュリティ (リンク層の暗号化)
802.3ad	LACP によるリンク集約
802.3ab	1000Base-T (銅線 10/100/1000 イーサネット)
802.3ae	10 ギガビット イーサネット
802.1Q	VLAN タギング
802.1p	イーサネットフレームの Class of Service (CoS) タギング
802.1X	ポート ベースのネットワーク アクセス コントロール

次の表に、RFC 準拠標準を示します。

表 3: RFC 準拠標準

標準	説明
BGP	
RFC 1997	『BGP Communities Attribute』
RFC 2385	『Protection of BGP Sessions via the TCP MD5 Signature Option』
RFC 2439	『BGP Route flap damping』

標準	説明
RFC 2519	『A Framework for Inter-Domain Route Aggregation』
RFC 2858	『Multiprotocol Extensions for BGP-4』
RFC 3065	『Autonomous System Confederations for BGP』
RFC 3392	『Capabilities Advertisement with BGP-4』
RFC 4271	『BGP version 4』
RFC 4273	『BGP4 MIB - Definitions of Managed Objects for BGP-4』
RFC 4456	『BGP Route reflection』
RFC 4486	『Subcodes for BGP cease notification message』
RFC 4724	『Graceful Restart Mechanism for BGP』
RFC 4893	『BGP Support for Four-octet AS Number Space』
IETF ドラフト	『Bestpath transition avoidanc』 (draft-ietf-idr-avoid-transition-05.txt)
IETF ドラフト	『Peer table objects』 (draft-ietf-idr-bgp4-mib-15.txt)
IETF ドラフト	『Dynamic Capability』 (draft-ietf-idr-dynamic-cap-03.txt)
OSPF	
RFC 2370	『OSPF Opaque LSA Option』
RFC 2328	『OSPF Version 2』
RFC 2740	『OSPF for IPv6 (OSPF version 3)』

標準	説明
RFC 3101	『OSPF Not-So-Stubby-Area (NSSA) Option』
RFC 3137	『OSPF Stub Router Advertisement』
RFC 3509	『Alternative Implementations of OSPF Area Border Routers』
RFC 3623	『Graceful OSPF Restart』
RFC 4750	『OSPF Version 2 MIB』
RIP	
RFC 1724	『RIPv2 MIB extension』
RFC 2082	『RIPv2 MD5 Authentication』
RFC 2453	『RIP Version 2』
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 routing in TCP/IP and dual environment』
RFC 2763	『Dynamic Hostname Exchange Mechanism for IS-IS』
RFC 2966	『Domain-wide Prefix Distribution with Two-Level IS-IS』
RFC 2973	『IS-IS Mesh Groups』
RFC 3277	『IS-IS Transient Blackhole Avoidance』
RFC 3373	『Three-Way Handshake for IS-IS Point-to-Point Adjacencies』

標準	説明
RFC 3567	『IS-IS Cryptographic Authentication』
RFC 3847	『Restart Signaling for IS-IS』
IETF ドラフト	『Internet Draft Point-to-point operation over LAN in link-state routing protocols』 (draft-ietf-isis-igp-p2p-over-lan-06.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	プロキシ ARP
RFC 1305	NTP v3
RFC 1519	CIDR
RFC 1542	BootP リレー
RFC 1591	DNS クライアント
RFC 1812	IPv4 ルータ
RFC 2131	DHCP ヘルパー
RFC 2338	VRRP
RFC 2784	総称ルーティング カプセル化 (GRE)

標準	説明
IP マルチキャスト	
RFC 2236	『Internet Group Management Protocol, Version 2』
RFC 2710	『Multicast Listener Discovery (MLD) for IPv6』
RFC 3376	『Internet Group Management Protocol, Version 3』
RFC 3446	『Anycast Rendezvous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)』
RFC 3569	『An Overview of Source-Specific Multicast (SSM)』
RFC 3618	Multicast Source Discovery Protocol (MSDP)
RFC 3810	『Multicast Listener Discovery Version 2 (MLDv2) for IPv6』
RFC 4601	『ASM - Sparse Mode (PIM-SM): Protocol Specification (Revised)』
RFC 4607	『Source-Specific Multicast for IP』
RFC 4610	『Anycast-RP Using Protocol Independent Multicast (PIM)』
IETF ドラフト	『Mtrace server functionality, to process mtrace-requests』 (draft-ietf-idmr-traceroute-ipm-07.txt)
IETF ドラフト	『Bi-directional Protocol Independent Multicast (BIDIR-PIM)』 (draft-ietf-pim-bidir-09.txt)



第 3 章

Cisco NX-OS セットアップユーティリティの使用

この章の内容は、次のとおりです。

- 機能情報の確認, 19 ページ
- Cisco NX-OS セットアップユーティリティに関する情報, 19 ページ
- セットアップユーティリティの前提条件, 21 ページ
- Cisco NX-OS デバイスの設定, 21 ページ
- セットアップユーティリティに関する追加情報, 27 ページ

機能情報の確認

ご使用のソフトウェアリリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェアリリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「新機能および変更された機能に関する情報」の章を参照してください。

Cisco NX-OS セットアップユーティリティに関する情報

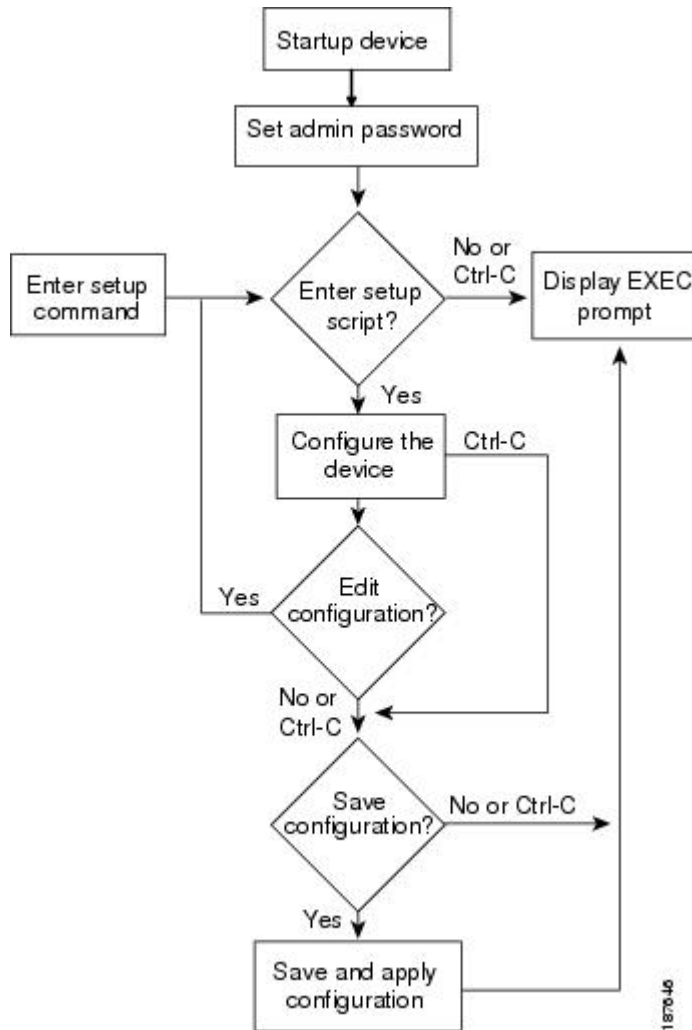
Cisco NX-OS セットアップユーティリティは、システムの基本（スタートアップとも呼びます）設定をガイドする対話型のコマンドラインインターフェイス（CLI）モードです。セットアップユーティリティでは、システム管理に使用する接続だけを設定できます。

セットアップユーティリティでは、システム設定ダイアログを使用して初期設定ファイルを作成できます。セットアップは、デバイスの NVRAM にコンフィギュレーションファイルがない場合に自動的に開始されます。ダイアログによって、初期設定の説明が行われます。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、残りの設定オプションをスキップし、その時点までに設定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、ホスト名）、デバイスでは以前の設定を使用して、次の質問に飛びます。

次の図に、セットアップスクリプトを開始および終了する方法を示します。

図 2: セットアップスクリプトのフロー



セットアップユーティリティは、設定がない場合にシステムを初めて設定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設定値が維持されます。たとえば、すでに `mgmt0` インターフェイスを設定している場合、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設定を変更しません。ただし、ステップ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設定値ではなくデフォルトを使用して設定が変更されます。設定を保存する前に、よく設定の変更内容を確認してください。



- (注) SNMP アクセスをイネーブルにする場合は、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 アドレスを設定してください。IPv4 ルーティングをイネーブルにすると、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 ルーティングがディセーブルの場合、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。



- (注) 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方針があること。
- 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。
- ライセンスの猶予期間は、必要に応じてイネーブルであること。ライセンスの詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

Cisco NX-OS デバイスの設定

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設定するには、次の手順を実行します。

手順

- 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イネーブルまたはディセーブルにします。強固なパスワードは、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
- 固有名詞を含んでいない
- 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

例 :

```
----- 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。
- (注) パスワードが脆弱な場合は (短い、解読されやすいなど)、そのパスワード設定が拒否されます。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設定してください。

例 :

```
Enter the password for "admin": <password>
Confirm the password for "admin": <password>
----- Basic System Configuration Dialog VDC: 1 -----

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 Nexus7000 Family devices promptly with your
supplier. Failure to register may affect response times for initial
service calls. Nexus7000 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。

例 :

```
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。

例 :

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

- a) ユーザ ログイン ID を入力します。

例 :

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


注意 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(+=.\-_)のみを含めることができます。#記号と!記号はサポートされていません。許可されていない文字がユーザ名に含まれていると、そのユーザはログインできません。

b) ユーザのパスワードを入力します。

例:

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

c) デフォルトのユーザ ロールを入力します。

例:

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

デフォルトのユーザ ロールの詳細については、『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x』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tringを設定します。

例:

```
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 7000 Series NX-OS System Management Configuration Guide, Release 6.x』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)。

例:

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

ステップ 8 yes と入力して、ライセンスの猶予期間をイネーブルにします。

(注) 猶予期間をイネーブルにすると、Cisco TrustSec 以外のライセンスが必要な機能をテストできます。ただし Cisco TrustSec には Advanced Services ライセンスが必要です。猶予期間は120日で、ライセンスが必要な機能を初めて設定したときから始まり、ライセンスのためのすべての機能がディセーブルになったときに終了します。猶予期間が終了した場合は、ライセンスが必要な機能を使用するためにライセンスを購入する必要があります。ライセンスの詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

例:

```
Enable license grace period? (yes/no) [n]: yes
```

ステップ 9 yes と入力して、アウトオブバンド管理を設定します。mgmt0 IPv4 アドレスとサブネット マスクを入力できます。

(注) セットアップユーティリティで設定できるのは、IPv4 アドレスだけです。IPv6 の設定については、『Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide, Release 6.x』を参照してください。

例：

```
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 10 yes と入力して、IPv4 デフォルトゲートウェイ（推奨）を設定します。これで、IP アドレスを入力できます。

例：

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

ステップ 11 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 オプションを設定します。

例：

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

ステップ 12 yes と入力して、スタティックルート（推奨）を設定します。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。

例：

```
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 13 yes と入力して、デフォルトネットワーク（推奨）を設定します。この IPv4 アドレスを入力します。

(注) 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設定の宛先プレフィックスと同じです。

例：

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

ステップ 14 yes と入力して、DNS の IPv4 アドレスを設定します。アドレスを入力します。

例：

```
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、デフォルトのドメイン名を設定します。名前を入力します。

例 :

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

ステップ 16 yes と入力して、Telnet サービスをイネーブルにします。

例 :

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

ステップ 17 yes と入力して、SSH サービスをイネーブルにします。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 7000 Series NX-OS Security Configuration Guide, Release 6.x*』を参照してください。

例 :

```
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 18 yes と入力して、NTP サーバを設定します。これで、IP アドレスを入力できます。詳細については、『*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x*』を参照してください。

例 :

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

ステップ 19 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。

例 :

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

ステップ 20 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 7000 Series NX-OS Interfaces Configuration Guide, Release 6.x*』を参照してください。

例 :

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

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 7000 Series NX-OS Security Configuration Guide, Release 6.x*』を参照してください。

例：

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

- ステップ 22** 現在のスーパーバイザの CMP を設定し、yes と入力して、IP アドレス、ネットマスク、およびデフォルトゲートウェイの IP を入力します。詳細については、『Cisco Nexus 7000 Series Connectivity Management Processor Configuration Guide』を参照してください。

例：

```
Configure CMP processor on current sup (slot 5)? (yes/no) [y]: yes
cmp-mgmt IPv4 address : IP_address
cmp-mgmt IPv4 netmask : net_mask
IPv4 address of the default gateway : default_gateway
```

- ステップ 23** yes と入力して、冗長スーパーバイザの CMP を設定します。次に、IP アドレス、ネットマスク、デフォルトゲートウェイ IP を入力できます。

例：

```
Configure CMP processor on standby sup (slot 5)? (yes/no) [y]: yes
cmp-mgmt IPv4 address : IP_address
cmp-mgmt IPv4 netmask : net_mask
IPv4 address of the default gateway : default_gateway
```

システムによって設定全体が要約され、編集するかどうか尋ねられます。

- ステップ 24** 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最初に戻り、各ステップを繰り返します。

例：

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

- ステップ 25** yes と入力して、この設定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、次のデバイス起動時に設定が更新されません。新しい設定を保存するには、yes と入力します。これによって、キックスタートイメージおよびシステムイメージのブート変数も自動的に設定されます。

例：

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

注意 ここで設定を保存しておかないと、次のデバイス起動時に設定が更新されません。yes と入力して新しい設定を保存し、キックスタートイメージおよびシステムイメージのブート変数も自動的に設定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	参照先
ライセンス	『Cisco NX-OS Licensing Guide』
コマンドリファレンス	『Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference』
CMP	『Cisco Nexus 7000 Series Connectivity Management Processor Configuration Guide』
SSH および Telnet	『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x』
ユーザ ロール	『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 6.x』
IPv4 および IPv6	『Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide, Release 6.x』
SNMP および NTP	『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x』



第 4 章

PowerOn Auto Provisioning の使用方法

この章では、Cisco Nexus 7000 シリーズ デバイスに PowerOn Auto Provisioning (POAP) を導入し、使用方法について説明します。

この章は、次の項で構成されています。

- [機能情報の確認, 29 ページ](#)
- [PowerOn Auto Provisioning の概要, 29 ページ](#)
- [POAP の注意事項および制約事項, 36 ページ](#)
- [POAP を使用するためのネットワーク環境の設定, 37 ページ](#)
- [POAP を使用するスイッチの設定, 38 ページ](#)
- [デバイス コンフィギュレーションの確認, 38 ページ](#)

機能情報の確認

ご使用のソフトウェア リリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「新機能および変更された機能に関する情報」の章を参照してください。

PowerOn Auto Provisioning の概要

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コンフィギュレーション 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 Cisco Nexus シリーズは、起動時にスタートアップ コンフィギュレーションが見つからないと、POAP モードを開始し、DHCP サーバを検索し、自分のインターフェイス IP ア

ドレス、ゲートウェイ、および DNS サーバ IP アドレスを自力で設定します。また、スイッチは TFTP サーバの IP アドレスまたは HTTP サーバの URL を取得し、コンフィギュレーションスクリプトをダウンロードします。このスクリプトはスイッチをイネーブルにして、適切なソフトウェアイメージとコンフィギュレーションファイルをダウンロードしてインストールします。



(注) DHCP 情報は、POAP 処理中にだけ使用されます。

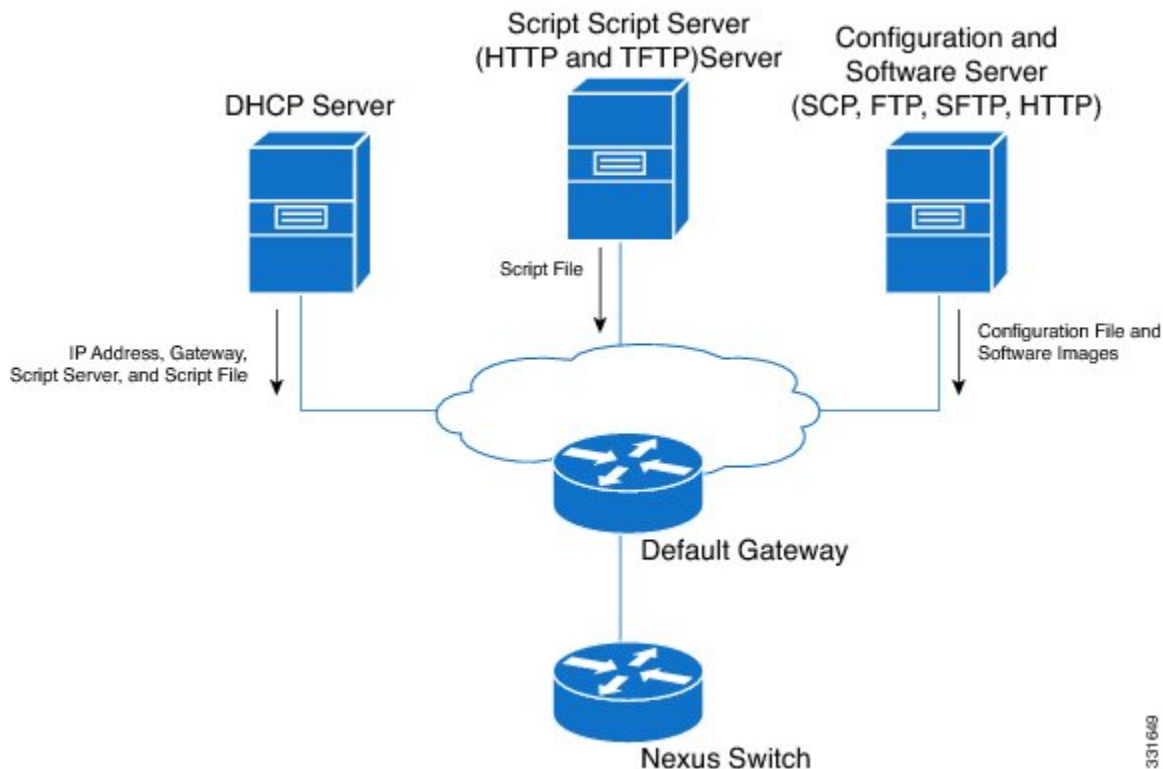
POAP のためのネットワーク要件

必要なインストールファイルが格納されている Universal Serial Bus (USB) デバイスが使用できない場合、POAP には次のネットワーク インフラストラクチャが必要です。

- 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 DNS (ドメインネームシステム) サーバをブートストラップする DHCP サーバ。
- ソフトウェアイメージのインストールと設定のプロセスを自動化するコンフィギュレーションスクリプトが保管されている TFTP サーバ。

- 必要なソフトウェア イメージとコンフィギュレーション ファイルが保管されている 1 台以上のサーバ。

図 3: POAP ネットワーク インフラストラクチャ



331649

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。
- スイッチ上にソフトウェア イメージ（システム イメージとキックスタート イメージ）がまだ存在しない場合は、それらのファイルをダウンロードします。ソフトウェア イメージがスイッチ上にインストールされ、次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 コンフィギュレーションとして設定を保存します。

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コンフィギュレーション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の要件に合わせてカスタマイズできます。

Cisco Nexus 7000 シリーズ デバイスでは、Python プログラミング言語は CLI コマンドを実行できる 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。

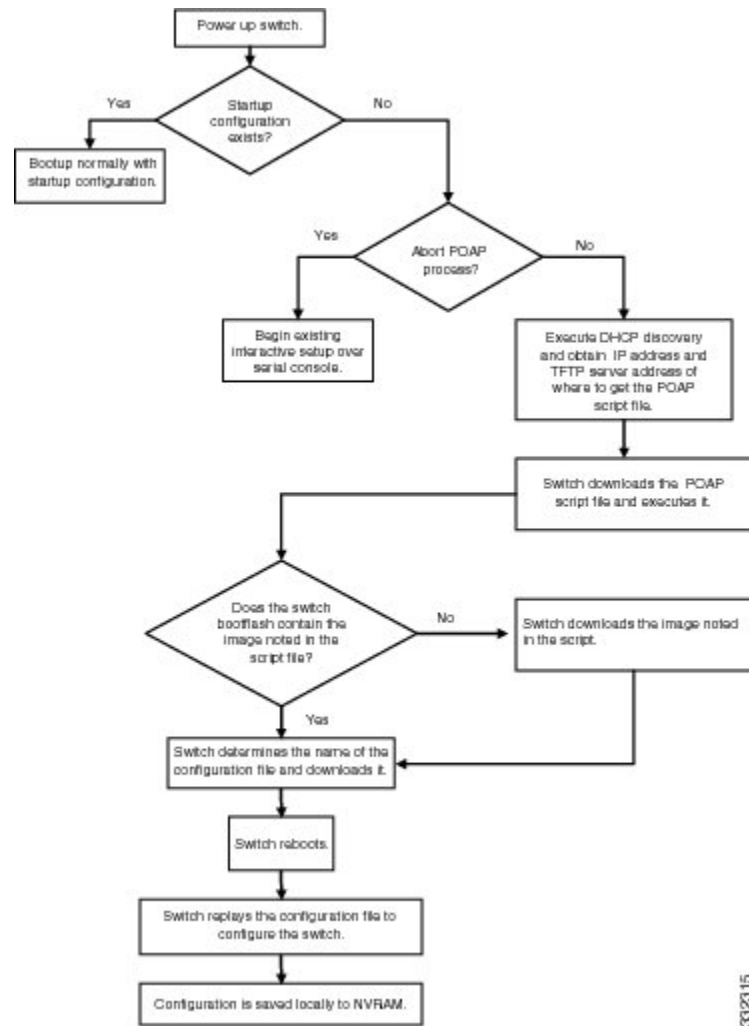
POAP 処理

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 4: POAP 処理



電源投入フェーズ

スイッチの電源を初めて投入すると、スイッチ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、起動元となるコンフィギュレーションファイルを探します。コンフィギュレーション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設定されます。

DHCP 検出フェーズ

スイッチは、1 台以上の DHCP サーバからの DHCP オファーを要求する DHCP 検出メッセージを MGMT インターフェイスで送信します（次の図を参照）。Cisco Nexus スイッチ上の DHCP クライアントは、クライアント ID オプションのスイッチシリアル番号を使用して、それ自体を DHCP サーバに識別させます。DHCP サーバはこの 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、3600 秒（1 時間）以上の 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。

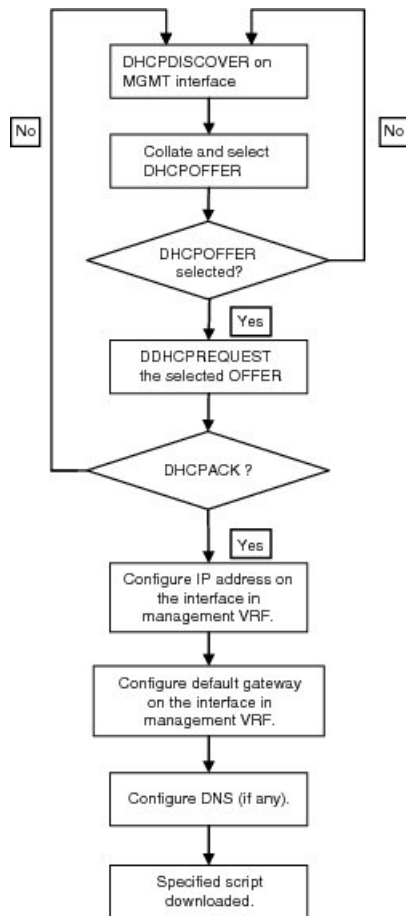
また、DHCP 検出メッセージでは、DHCP サーバからの次のオプションも要求します。

- 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た場合は、1 つのオファーがランダムに選択されます。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 5: DHCP 検出プロセス



9346511

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、コンフィギュレーション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コンフィギュレーション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。

POAP の注意事項および制約事項

POAP 設定時の注意事項および制約事項は次のとおりです。

- この機能が動作するには、Cisco Nexus 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。
- vPC (仮想ポートチャネル) リンクのスタティック ポートチャネルを使用している 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ます。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 7000 シリーズ デバイス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 7000 シリーズ デバイス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 7000 シリーズ デバイスを設定します。
- 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。ファイル名の形式は `date-time_poap_PID_[init,1,2].log` です。ここで、`date-time` の形式は `YYYYMMDD_hhmmss` で、`PID` はプロセス ID になります。

- 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。ファイル名の形式は *date-time_poap_PID_script.log* です。ここで、*date-time* の形式は YYYYMMDD_hhmmss で、*PID* はプロセス ID になります。

スクリプトのログ ファイルの形式を設定できます。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。



(注) 適切なライセンスをインストールすることなく、POAP 機能を一時的に使用するには、コンフィギュレーション ファイルで **license grace-period** コマンドを指定できます。

この回避策では、適切なライセンスを後からインストールすることができません。

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。
- ステップ 2** (任意) POAP のコンフィギュレーション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コンフィギュレーション ファイルを、スイッチにアクセスできる USB デバイスに配置します。
- ステップ 3** 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 コンフィギュレーション ファイルが USB デバイスにある場合は、DHCP サーバを配置する必要はありません。
- ステップ 4** コンフィギュレーション スクリプトをホストするための TFTP サーバを配置します。
- ステップ 5** 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

はじめる前に

POAP を使用するよう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。詳細については、この項の前の項 [POAP を使用するためのネットワーク環境の設定](#) を参照してください。

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witchの電源を入れます。
 コンフィギュレーションファイルが存在しない場合、スイッチは POAP モードで起動し、POAP を中止して、通常の設定を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** (任意) 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witchが起動して、POAP 処理が開始されます。詳細については、[POAP 処理](#)、(32 ページ) を参照してください。
-

次の作業

設定を確認します。

デバイス コンフィギュレーションの確認

POAP を使用してデバイスのブートストラップ後の設定を確認するには、次のコマンドのいずれかを使用します。

コマンド	目的
<code>show running-config</code>	実行コンフィギュレーションを表示します。
<code>show startup-config</code>	スタートアップコンフィギュレーションを表示します。

これらのコマンドの出力フィールドの詳細については、ご使用のデバイスの『Cisco Nexus Command Reference』を参照してください。



第 5 章

コマンドラインインターフェースの概要

この章の内容は、次のとおりです。

- 機能情報の確認, 40 ページ
- CLI プロンプトの概要, 40 ページ
- コマンドモード, 41 ページ
- 特殊文字, 46 ページ
- キーストロック ショートカット, 47 ページ
- コマンドの省略形, 49 ページ
- 部分的なコマンド名の補完, 50 ページ
- コマンド階層での現在の場所の識別, 51 ページ
- コマンドの `no` 形式の使用, 51 ページ
- CLI 変数の設定, 52 ページ
- コマンドエイリアス, 54 ページ
- コマンドスクリプト, 56 ページ
- 状況依存ヘルプ, 58 ページ
- 正規表現の概要, 60 ページ
- `show` コマンドの出力の検索およびフィルタリング, 61 ページ
- `--More--` プロンプトからの検索およびフィルタリング, 67 ページ
- コマンド履歴の使用, 68 ページ
- CLI の確認プロンプトのイネーブルまたはディセーブル, 70 ページ
- CLI の表示色の設定, 71 ページ
- モジュールへのコマンドの送信, 71 ページ

- BIOS ロダー プロンプト, 72 ページ
- CLI の使用例, 72 ページ
- CLI に関する追加情報, 75 ページ

機能情報の確認

ご使用のソフトウェア リリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「新機能および変更された機能に関する情報」の章を参照してください。

CLI プロンプトの概要

デバイスに正常にアクセスすると、コンソールポートまたはリモートワークステーションの端末ウィンドウに、次のような 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) 2002-2009, 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#
```

デバイスのデフォルトのホスト名は変更できます。

CLI プロンプトから、次の方法を実行できます。

- CLI コマンドによる機能の設定
- コマンド履歴へのアクセス
- コマンド解析機能の使用



(注) 通常操作では、ユーザ名は大文字と小文字が区別されます。ただし、コンソールポートにデバイスを接続しているときには、ユーザ名の設定に関係なく、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

最初にログインしたときは、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に関する情報を表示する **show** コマンド、**clear** コマンド、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などがあります。

グローバル 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 モードでは、最も広範囲のコマンドを使用できます。この用語は、デバイス全体に影響を与える特性や機能を表します。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入力すると、デバイスをグローバルに設定することができます。また、さらに特定のコンフィギュレーションモードを開始して、インターフェイスやプロトコルなどの特定の要素を設定することもできます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： <pre>switch# configure terminal switch(config)#</pre>	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。 (注) CLI プロンプトが変化し、グローバルコンフィギュレーションモードになっていることが示されます。

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

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コンフィギュレーション モードの一例が、インターフェイス コンフィギュレーション モードです。デバイスで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て、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。

インターフェイス単位で多数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は、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を変更します。

インターフェイスの設定の詳細については、ご使用のデバイスの『Cisco Nexus Interfaces Guide』を参照してください。

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	interface type number 例： 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。 この CLI によって、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が開始されます。 (注) CLIプロンプトが変化し、インターフェイスコンフィギュレーションモードになっていることが示されます。

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、プロトコルに対しては個別の物理インターフェイスとして解釈されます。

また、サブインターフェイスにより、1つのインターフェイスで、あるプロトコルに対する複数のカプセル化を使用できます。たとえば、サブインターフェイスに VLAN を関連付ける IEEE 802.1Q カプセル化を設定できます。

サブインターフェイスの設定の詳細については、ご使用のデバイスの『Cisco Nexus Interfaces Guide』を参照してください。サブインターフェイス コマンドの詳細については、ご使用のデバイスの『Command Reference Guide』を参照してください。

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。

	コマンドまたはアクション	目的
ステップ 2	interface <i>type number.subint</i> 例： <pre>switch(config)# interface ethernet 2/2.1 switch(config-subif)#</pre>	設定する VLAN インターフェイスを指定します。 この CLI によって、指定した VLAN インターフェイスのサブインターフェイス コンフィギュレーション モードが開始されます。 (注) CLI プロンプトが変化し、グローバル コンフィギュレーション モードになっていることが示されます。

コマンドモードの保存および復元

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。 **push** コマンドでコマンドモードを保存し、**pop** コマンドでコマンドモードを復元します。

次の例は、コマンドモードを保存し、復元する方法を示します。

```
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

コンフィギュレーション コマンドモードを終了するには、次のいずれかの作業を行います。

手順

	コマンドまたはアクション	目的
ステップ 1	exit 例： <pre>switch(config-if)# exit switch(config)#</pre>	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、元のコンフィギュレーション コマンドモードに戻ります。

	コマンドまたはアクション	目的
ステップ 2	end 例： <pre>switch(config-if)# end switch#</pre>	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。
ステップ 3	Ctrl+Z 例： <pre>switch(config-if)# ^z switch#</pre>	(任意) 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 を押すと、CLI によってそのコマンドが実行コンフィギュレーション 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

次の表は、主なコマンドモードに関する情報をまとめたものです。

表 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 コンフィギュレーションモードで、インターフェイスコマンドを使用し、 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 コンフィギュレーションモードに戻るには、 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 コンフィギュレーションモードで、 interface コマンドを使用してサブ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 コンフィギュレーションモードに戻るには、 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。
VDC コンフィギュレーション	グローバル コンフィギュレーションモードで、 vdc コマンドを使用し、VDC 名を指定します。	switch(config-vdc)#	終了してグローバル コンフィギュレーションモードに戻るには、 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 コンフィギュレーションモードで、 vrf コマンドを使用し、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 コンフィギュレーションモードに戻るには、 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 VDC のための EXEC	EXEC モードで、 switchto vdc コマンドを使用し、VDC を指定します。	switch-vdc2#	終了してデフォルトの VDC に戻るには、 exit コマンドまたは switchback コマンドを使用します。
デフォルト以外の VRF のための EXEC	EXEC モードで、 routing-context vrf コマンドを使用し、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 のテキストストリング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。

表 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
...	省略記号
	縦線
<>	より小さい、またはより大きい
[]	角カッコ

文字	説明
{ }	波カッコ

キーストローク ショートカット

次の表に、EXEC モードとコンフィギュレーションモードの両方で使用できるコマンドキーの組み合わせを示します。

表 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文字分だけ左に進めます。複数行にわたってコマンドを入力するときは、左矢印キーまたはCtrl+Bキーを繰り返し押し続け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を確認できます。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文字分だけ右に進めます。
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	端末の画面をクリアします。

キーストローク	説明
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	カーソルの場所にある文字を、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	後に続くキーストロークの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。
Ctrl+X、H	入力したコマンドの履歴を一覧表示します。 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。
↑キー	コマンド履歴の前のコマンドを表示します。
↓キー	コマンド履歴の次のコマンドを表示します。
→キー ←キー	コマンドストリング内でカーソルを前方または後方に移動させ、現在のコマンドを編集できるようにします。
?	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
Tab	<p>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。</p> <p>タブを使用すると、次の項目名を完成させることができます。</p> <ul style="list-style-type: none"> • 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 <p>例 :</p> <pre>switch(config)# xm<Tab> switch(config)# xml<Tab> switch(config)# xml server</pre> <p>例 :</p> <pre>switch(config)# c<Tab> callhome class-map clock cts cdp cli control-plane switch(config)# cl<Tab> class-map cli clock switch(config)# cla<Tab> switch(config)# class-map</pre> <p>例 :</p> <pre>switch# cd bootflash:<Tab> bootflash: bootflash://sup-1/ bootflash:/// bootflash://sup-2/ bootflash://module-5/ bootflash://sup-active/ bootflash://module-6/ bootflash://sup-local/</pre> <p>例 :</p> <pre>switch# cd bootflash://mo<Tab> bootflash://module-5/ bootflash://module-6/cv switch# cd bootflash://module-</pre>

コマンドの省略形

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、コマンドの省略形の例を示します。

表 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、入力された文字列がコマンドモードに対して一意である場合に、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 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** コマンドと関連付けることができます。これは、**conf** で始まるコマンドが **configure** コマンドしかないためです。

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。CLI は、Return または Enter キーが押されるまでコマンドを実行しません。この機能により、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。複数のコマンドを示す可能性のある一連の文字を入力した場合は、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に、入力した文字が再びプロンプトに表示されることに注意してください。

コマンド階層での現在の場所の識別

一部の機能においては、複数のレベルにわたる設定サブモード階層があります。このような場合には、現在の作業コンテキスト（PWC）についての情報を表示できます。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>where detail</p> <p>例：</p> <pre>switch# configure terminal switch(config)# interface mgmt0 switch(config-if)# where detail mode: conf interface mgmt0 username: admin vdc: switch routing-context vrf: default</pre>	PWC を表示します。

コマンドの no 形式の使用

ほぼすべてのコンフィギュレーションコマンドには、機能をディセーブルにしたり、デフォルト値に戻したり、設定を削除したりするために使用できる **no** 形式があります。Cisco NX-OS のコマンドリファレンスの資料では、コマンドの **no** 形式が使用できる場合は常に **no** 形式の機能について説明しています。

次の例では、機能をディセーブルにする方法を示します。

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

次の例では、機能のデフォルト値に戻す方法を示します。

```
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
```

次の例では、機能の設定を削除する方法を示します。

```
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

```

次の例では、EXEC モードでコマンドの **no** 形式を使用する方法を示します。

```

switch# cli var name testinterface ethernet1/2
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2009-05-12-13.43.13"
testinterface="ethernet1/2"

switch# cli no var name testinterface
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2009-05-12-13.43.13"

```

CLI 変数の設定

ここでは、Cisco NX-OS CLI 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するスクリプトに渡す。親シェルで定義した変数は、子の **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または現在のセッションの間だけ存在できます。

Cisco NX-OS は、事前定義された **TIMESTAMP** 変数をサポートします。この変数は、コマンドが実行される現在の時刻を **YYYY-MM-DD-HH.MM.SS** という形式で参照します。



(注) TIMESTAMP 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 セッションのみの変数の設定

CLI セッション変数を、CLI セッションの期間のみ保持されるように定義できます。これらの変数は、定期的に行うスクリプトに役立ちます。名前をカッコで囲み、その前にドル記号 (\$) を付加することによって、その変数を参照できます (たとえば、\$(variable-name))。

手順

	コマンドまたはアクション	目的
ステップ 1	cli var name variable-name variable-text 例: <pre>switch# cli var name testinterface ethernet 2/1</pre>	CLI セッション変数を設定します。 <i>variable-name</i> 引数には、大文字と小文字を区別して、最大 31 文字の英数字で値を指定します。 <i>variable-text</i> 引数は 200 文字以下の長さの英数字で、大文字と小文字を区別し、スペースを含むことができます。
ステップ 2	show cli variables 例: <pre>switch# show cli variables</pre>	(任意) CLI 変数の設定を表示します。

永続的な CLI 変数の設定

CLI セッションやデバイスのリロードをまたいで保持される CLI 変数を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例: <pre>switch# configure terminal switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	cli var name variable-name variable-text 例： <pre>switch(config)# cli var name testinterface ethernet 2/1</pre>	CLI 固定変数を設定します。変数名は、英数字の文字列で指定します。大文字と小文字が区別されます。変数名の先頭は英字にする必要があります。最大長は 31 文字です。
ステップ 3	exit 例： <pre>switch(config)# exit switch#</pre>	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	show cli variables 例： <pre>switch# show cli variables</pre>	(任意) CLI 変数の設定を表示します。
ステップ 5	copy running-config startup-config 例： <pre>switch(config)# copy running-config startup-config</pre>	(任意) 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関する情報を提供します。

コマンド エイリアスについて

頻繁に使用するコマンドを、コマンドエイリアスに置き換えて定義することができます。コマンドエイリアスは、コマンド構文のすべて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をスタートアップ コンフィギュレーションに保存すると、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である **alias** を削除または変更することはできません。
- エイリアスは最大で1の深さにネストできます。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。
- あらゆるコマンドモードで、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でCLI変数を参照すると、エイリアスには、変数の参照ではなく現在の変数の値が表示されます。
- 検索およびフィルタリングを実行する **show** コマンドのコマンドエイリアスを使用できます。

コマンドエイリアスの定義

一般に使用されるコマンドのコマンドエイリアスを定義できます。

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 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	cli alias name alias-name alias-text 例： 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 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。
ステップ 4	alias 例： switch# alias	(任意) コマンドエイリアスの設定を表示します。

	コマンドまたはアクション	目的
ステップ 5	copy running-config startup-config 例： <pre>switch# copy running-config startup-config</pre>	(任意) 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が使用できない、現在のユーザ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、現在のユーザアカウントによる将来の使用のためにコマンドエイリアスを保存することもできます。

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： <pre>switch# terminal alias shintbr show interface brief</pre>	現在のユーザセッションのコマンドエイリアスを設定します。このユーザアカウントによる将来のため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 キーワードを省略しないでください。

コマンドスクリプト

ここでは、複数の作業を実行する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

コマンドのリストをファイルに記述して、CLI から実行することができます。コマンドスクリプトでは CLI 変数を使用できます。



(注) CLI プロンプトでは、スクリプトファイルは作成できません。スクリプトファイルは、リモートデバイスで作成し、Cisco NX-OS デバイス上の `bootflash:`、`slot0:`、または `volatile:` ディレクトリにコピーします。

手順

	コマンドまたはアクション	目的
ステップ 1	run-script [bootflash: slot0: volatile:] <i>filename</i> 例 : switch# run-script testfile	デフォルトディレクトリでファイルに記述された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から使用すると特に有効です。エコーされたテキストで CLI 変数を参照したり、フォーマットオプションを使用したりすることができます。

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。

表 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	バックスペースが挿入されます。
\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slash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。

手順

	コマンドまたはアクション	目的
ステップ 1	echo [backslash-interpret] [text] 例： <pre>switch# echo This is a test. This is a test.</pre>	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ていることを示します。 <i>text</i> 引数は、大文字と小文字が区別される英数字で、空白を含むことができます。最大長は 200 文字です。デフォルトは空白行です。

コマンド処理の遅延

コマンド処理を一定時間遅らせることができます。これは、コマンドスクリプト内で特に有効です。

手順

	コマンドまたはアクション	目的
ステップ 1	sleep seconds 例： <pre>switch# sleep 30</pre>	数秒の遅延を発生させます。範囲は 0～2147483647 です。

状況依存ヘルプ

Cisco NX-OS ソフトウェアの CLI には、状況依存ヘルプが用意されています。任意の箇所でコマンドに疑問符 (?) を指定すると、使用できる入力オプションが一覧表示されます。

CLI では、入力エラーを隔離するためにキャレット (^) 記号を使用します。^記号は、コマンドストリング内のコマンド、キーワード、または引数が誤って入力されている位置に表示されます。

この表では、状況依存ヘルプの出力例について説明します。

表 9: 状況依存ヘルプの例

出力例	説明
<pre>switch# clock ? set HH:MM:SS Current Time switch# clock</pre>	<p>EXEC モードで clock コマンドのコマンド構文を表示します。</p> <p>このスイッチの出力では、clock コマンドを使用するためには set キーワードが必要であることが示されています。</p>
<pre>switch# clock set ? WORD HH:MM:SS Current Time switch# clock set</pre>	<p>時間を設定するためのコマンド構文を表示します。</p> <p>このヘルプの出力では、クロックの設定に現在の時刻が必要であることと、時刻の形式が示されています。</p>
<pre>switch# clock set 13:32:00<CR> % Incomplete command switch#</pre>	<p>現在の時刻を追加します。</p> <p>CLI によって、コマンドが不完全であることが示されます。</p>
<pre>switch# <Ctrl-P> switch# clock set 13:32:00</pre>	<p>入力した前のコマンドを表示します。</p>
<pre>switch# clock set 13:32:00 ? <1-31> Day of the month switch# clock set 13:32:00</pre>	<p>clock set コマンドの他の引数が表示されます。</p>
<pre>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</pre>	<p>clock set コマンドの他の引数が表示されます。</p>
<pre>switch# clock set 13:32:00 18 April 08<CR> % Invalid input detected at '^' marker.</pre>	<p>クロック設定に日付を追加します。</p> <p>CLI は、08 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。</p>
<pre>switch# clock set 13:32:00 18 April ? <2000-2030> Enter the year (no abbreviation) switch# clock set 13:32:00 18 April</pre>	<p>年を指定する正しい引数が表示されます。</p>
<pre>switch# clock set 13:32:00 18 April 2008<CR> switch#</pre>	<p>clock set コマンドの正しい構文を入力します。</p>

正規表現の概要

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。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

その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、正規表現で使用されると特殊な意味を持ちます。

次の表に、特殊な意味を持つキーボード文字を示します。

表 10: 特殊な意味を持つ特殊文字

文字	特殊な意味
.	任意の 1 文字 (スペースを含む) 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。
?	パターンの 0 または 1 回の出現と一致します。
^	文字列の最初と一致します。
\$	文字列の最後と一致します。
_ (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 (])、文字列の先頭、文字列の最後、またはスペースと一致します。 (注) アンダースコアは、BPG 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には、各文字の前にバックスラッシュ (\) を置くことによって、特殊な意味を削除してください。次の例には、それぞれドル記号 (\$)、アンダースコア (_)、およびプラス記号 (+) に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、文字が `a`、`4`、パーセント記号 (`%`) の順に並んでいる場合に一致します。文字列の中に `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` という文字列が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、ストリング `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。

表 11: 位置指定に使用する特殊文字

文字	説明
<code>^</code>	文字列の最初と一致します。
<code>\$</code>	文字列の最後と一致します。

たとえば、正規表現 `^con` は `con` で始まる文字列に一致し、`sole$` は `sole` で終わる文字列に一致します。



(注) `^` 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]` が示す範囲は、`a`、`b`、`c`、`d` のいずれでもない任意の単一文字と一致します。

show コマンドの出力の検索およびフィルタリング

多くの場合、`show` コマンドの出力は、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。CLI の状況依存ヘルプ機能を使用してオプションを表示できます。

```
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
```

キーワードのフィルタリングおよび検索

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 の出力をフィルタリングおよび検索するためのキーワードを示します。

表 12: キーワードのフィルタリングおよび検索

キーワードの構文	説明
begin <i>string</i> 例： show version begin Hardware	検索文字列に一致するテキストが含まれる行から表示を開始します。検索文字列は、大文字と小文字が区別されます。
count 例： show running-config count	コマンドの出力行数を表示します。

キーワードの構文	説明
<p>cut [-d <i>character</i>] {-b -c -f -s}</p> <p>例： show file testoutput cut -b 1-10</p>	<p>出力行の一部分だけを表示します。一定のバイト数 (-b)、文字数 (-vcut [-d <i>character</i>] {-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。</p>
<p>end <i>string</i></p> <p>例： show running-config end interface</p>	<p>検索文字列が最後に現れる位置まですべての行を表示します。</p>
<p>exclude <i>string</i></p> <p>例： show interface brief exclude down</p>	<p>検索文字列が含まれないすべての行を表示します。検索文字列は、大文字と小文字が区別されます。</p>
<p>head [<i>lines lines</i>]</p> <p>例： show logging logfile head lines 50</p>	<p>出力の先頭部分を、指定した行数だけ表示します。デフォルトの行数は 10 行です。</p>
<p>human</p> <p>例： show version human</p>	<p>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。</p>
<p>include <i>string</i></p> <p>例： show interface brief include up</p>	<p>検索文字列が含まれるすべての行を表示します。検索文字列は、大文字と小文字が区別されます。</p>
<p>last [<i>lines</i>]</p> <p>例： show logging logfile last 50</p>	<p>出力の末尾部分を、指定した行数だけ表示します。デフォルトの行数は 10 行です。</p>
<p>no-more</p> <p>例： show interface brief no-more</p>	<p>画面の下端で --More-- プロンプトを出して停止せず、すべての出力を表示します。</p>
<p>sscp <i>SSH-connection-name filename</i></p> <p>例： show version sscp MyConnection show_version_output</p>	<p>ストリーミングセキュアコピー (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。</p>
<p>wc [<i>bytes lines words</i>]</p> <p>例： show file testoutput wc bytes</p>	<p>文字数、行数、またはワード数を表示します。デフォルトでは、行数、ワード数、文字数を表示します。</p>

キーワードの構文	説明
xml 例： <code>show version xml</code>	出力を XML 形式で表示します。

diff ユーティリティ

show 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-session**] [**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すると、**show** コマンドの出力をフィルタリングすることができます。

grep および 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** および **egrep** のパラメータを示します。

表 13: *grep* および *egrep* のパラメータ

パラメータ	説明
count	一致する行の合計数だけを表示します。
ignore-case	一致する行の大文字と小文字の違いを無視することを指定します。
invert-match	式と一致しない行を表示します。
line-exp	行が完全に一致する行だけを表示します。
line-number	一致する各行の先頭に行番号を表示することを指定します。
next lines	一致する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。指定できる範囲は 1 ~ 999 です。
prev lines	一致する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。指定できる範囲は 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。
<i>expression</i>	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すると、**show** コマンドの出力内容を一度に 1 ページずつ表示することができます。「:」プロンプトにおいて **less** コマンドを入力できます。使用可能な **less** コマンドをすべて表示するには、「:」プロンプトで **h** を入力します。

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、次のように **show** コマンドの出力をフィルタリングしたり、操作したりすることができます。

sed command

command 引数には、sed ユーティリティのコマンドを指定します。

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。

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	先頭の空白（スペース文字）を無視します。デフォルトのソートには、先頭の空白が含まれます。
-d	空白と英数字のみを比較することによってソートします。デフォルトのソートには、すべての文字が含まれます。
-f	小文字を大文字にします。
-g	一般的な数値を比較することによってソートします。
-i	印刷可能文字のみを使用してソートします。デフォルトのソートには、印刷不可能な文字が含まれます。
-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	数値文字列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。
-t delimiter	指定された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スペース文字です。
-u	ソート結果から重複した行を削除します。ソート出力は重複した行を表示します。

--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。

次の表に、--More- プロンプト コマンドの説明を示します。

表 15: --More-- プロンプトのコマンド

コマンド	説明
[lines]<スペース>	出力行を指定された行数または現在の画面サイズだけ表示します。
[lines]z	出力行を指定された行数または現在の画面サイズだけ表示します。 <i>lines</i> 引数を使用すると、その値が新しいデフォルトの画面サイズになります。
[lines]<リターン>	指定した行数または現在のデフォルトの行数だけ出力行を表示します。初期デフォルトは1行です。オプションの <i>lines</i> 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	指定した行数または現在のデフォルトの行数だけ出力行をスクロールします。初期デフォルトは11行です。オプションの <i>lines</i> 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	出力内の指定された行数または現在のデフォルトの行数だけ前方にスキップし、1画面分の行を表示します。デフォルトは1行です。
[lines]f	出力内の指定された画面数または現在のデフォルトの画面数だけ前方にスキップし、1画面分の行を表示します。デフォルトは1画面です。
=	現在の行番号を表示します。

コマンド	説明
<code>[count]/expression</code>	正規表現に一致する行にスキップし、1画面分の出力行を表示します。式が複数回出現する行を検索するには、オプションの <code>count</code> 引数を使用します。このコマンドでは、他のコマンドで使用できる現在の正規表現が設定されます。
<code>[count]n</code>	現在の正規表現に一致する次の行にスキップし、1画面分の出力行を表示します。一致を乗り越えてスキップするには、オプションの <code>count</code> 引数を使用します。
<code>{! :![shell-cmd]}</code>	<code>shell-cmd</code> 引数で指定されたコマンドをサブシェルで実行します。
<code>.</code>	前のコマンドを繰り返します。

コマンド履歴の使用

Cisco NX-OS ソフトウェアの CLI では、現在のユーザセッションのコマンド履歴にアクセスできます。変更を加えて、または変更なしでコマンドを呼び出したり、再発行したりできます。また、コマンド履歴をクリアすることもできます。

コマンドの呼び出し

コマンド履歴の中のコマンドを呼び出し、任意に変更を加えて、再入力することができます。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 の履歴呼び出しの制御

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 CLI の履歴から呼び出すコマンドを制御できます。デフォルトでは、CiscoNX-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している場合は、コマンド呼び出しのキーストロークショートカットによって、EXEC モードとグローバルコンフィギュレーションモードの両方のコマンドが呼び出されます。**terminal history no-exec-in-config** コマンドを使用すると、コンフィギュレーション モードにいるときに EXEC モード コマンドの呼び出しを回避できます。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>[no] terminal history no-exec-in-config</p> <p>例 :</p> <pre>switch# terminal history no-exec-in-config</pre>	<p>コンフィギュレーションモードで呼び出しのキーストローク ショートカットが使用されたときに EXEC コマンドを削除するように CLI の履歴を設定します。デフォルトでは、EXEC コマンドが呼び出されます。コマンドの no 形式を使用して、デフォルトに戻すことができます。</p>

CLI の編集モードの設定

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 CLI の履歴からコマンドを呼び出し、再発行する前にそれらのコマンドを編集できます。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

	コマンドまたはアクション	目的
ステップ 1	<p>[no] terminal edit-mode vi [persist]</p> <p>例 :</p> <pre>switch# terminal edit-mode vi</pre>	<p>そのユーザセッションについて、CLI の編集モードを vi に変更します。persist キーワードを使用すると、現在のユーザ名の設定がセッションをまたいで保持されます。emacs の使用に戻すには、no を使用します。</p>

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-only** | **exec-only** | **this-mode-only**] [**unformatted**]

デフォルトでは、表示される行数は12行で、コマンドの番号とタイムスタンプが出力されます。

次に、コマンド履歴をデフォルトの行数だけ表示する例を示します。

```
switch# show cli history
```

次に、コマンド履歴を 20 行表示する例を示します。

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

次に、コマンド履歴の中のコンフィギュレーション コマンドだけを表示する例を示します。

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

次に、コマンド履歴の中の EXEC コマンドだけを表示する例を示します。

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

次に、現在のコマンドモードに関するコマンド履歴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
```

次に、コマンド番号とタイムスタンプを表示せず、コマンド履歴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 の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またはディセーブルにすることができます。デフォルトはイネーブルです。

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 の確認プロンプトをディセーブルにします。 persist キーワードを使用すると、現在のユーザ名の設定がセッションをまたいで保持されます。デフォルトはイネーブルです。

	コマンドまたはアクション	目的
		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 の表示色の設定

次のように、表示される CLI の色を変更できます

- 前のコマンドの処理が成功した場合は、プロンプトを緑色で表示する。
- 前のコマンドの処理が失敗した場合は、プロンプトを赤色とする。
- ユーザの入力は青色で表示する。
- コマンドの出力はデフォルトの色で表示する。

デフォルトの色は、ターミナルエミュレータ ソフトウェアにより設定された色です。

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	端末セッションの CLI の表示色を設定します。 evening キーワードはサポートされません。 persist キーワードを使用すると、現在のユーザ名の設定がセッションをまたいで保持されます。 デフォルト設定は永続的ではありません。

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、フィルタリングする例を示します。

```
switch# slot 2 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 2 quoted "show version" | diff
switch# slot 4 quoted "show version" | diff -c
*** /volatile/vsh_diff_1_root_8430_slot__quoted_show_version.old      Wed Apr 29 20:10:41
    2009
--- -      Wed Apr 29 20:10:41 2009
*****
*** 1,5 ****
! RAM 1036860 kB
! lc2
  Software
    BIOS:      version 1.10.6
    system:    version 4.2(1) [build 4.2(0.202)]
--- 1,5 ----
! RAM 516692 kB
! lc4
  Software
    BIOS:      version 1.10.6
    system:    version 4.2(1) [build 4.2(0.202)]
*****
*** 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)
```

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キックスタート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。有効なキックスタート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>
```

<loader> プロンプトから Cisco NX-OS ソフトウェアをロードする方法については、ご使用のデバイスの『Cisco Nexus troubleshooting guide (Cisco Nexus トラブルシューティングガイド)』を参照してください。

CLI の使用例

ここでは、CLI の使用例について説明します。

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方法の例を示します。

```
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方法の例を示します。

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

CLI セッション変数の使用

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

次の例では、ユーザ定義の CLI セッション変数を参照する方法を示します。

```
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
```

システム定義のタイムスタンプ変数の使用

次の例では、**show** コマンドの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 2008  rcfg.2008-05-01-12.27.59

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

コマンドスクリプトの実行

次に、スクリプトファイルで指定する CLI コマンドの例を示します。

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

次に **run-script** コマンドの実行の出力例を示します。

```
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
```

CLI に関する追加情報

ここでは、CLI に関するその他の情報について説明します。

CLI の関連資料

関連項目	参照先
Cisco NX-OS のライセンス	『 <i>Cisco NX-OS Licensing Guide</i> 』
コマンド リファレンス	『 <i>Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference</i> 』



第 6 章

端末設定およびセッションの設定

この章の内容は、次のとおりです。

- 機能情報の確認, 77 ページ
- 端末設定とセッションの概要, 77 ページ
- 端末設定およびセッションのライセンス要件, 80 ページ
- コンソールポートの設定, 81 ページ
- COM1ポートの設定, 82 ページ
- 仮想端末の設定, 84 ページ
- モデム接続の設定, 86 ページ
- 端末セッションのクリア, 91 ページ
- 端末およびセッション情報の表示, 91 ページ
- ファイルシステムパラメータのデフォルト設定, 92 ページ
- 端末設定およびセッションに関する追加情報, 92 ページ

機能情報の確認

ご使用のソフトウェアリリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェアリリースのリリースノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「新機能および変更された機能に関する情報」の章を参照してください。

端末設定とセッションの概要

ここでは、端末設定およびセッションに関する情報について説明します。

端末セッションの設定

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ときに Telnet によって使用される名前

長さ

一時停止の前に表示されるコマンドの出力の行数

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

セッションがデバイスによって終了される前に非アクティブなままになる分数

コンソールポート

コンソールポートは非同期シリアルポートであり、RJ-45 コネクタを持つ標準の RS-232 ポートを使用してデバイスに接続し、初期設定を行うことができます。このポートに接続されるすべてのデバイスには、非同期伝送能力が必要です。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 8 ビット バイトのビット数を指定します。

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または偶数パリティを指定します。

速度

接続の送信速度を指定します。

ストップビット

非同期回線のストップビットを指定します。

ターミナルエミュレータには、9600 ボー、8 データビット、1 ストップビット、パリティなしを設定します。

COM1 ポート

COM1 ポートは、モデムなどの外部のシリアル通信デバイスに接続できる DB-9 インターフェイスを備えた RS-232 ポートです。COM1 ポートに関して、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 8 ビットバイトのビット数を指定します。

ハードウェア フロー制御

フロー制御ハードウェアをイネーブルにします。

パリティ

エラー検出用の奇数または偶数パリティを指定します。

速度

接続の送信速度を指定します。

ストップ ビット

非同期回線のストップ ビットを指定します。

ターミナルエミュレータには、9600 ボー、8 データ ビット、1 ストップ ビット、パリティなしを設定します。

仮想端末

仮想端末回線を使用して、Cisco NX-OS デバイスに接続できます。セキュアシェル (SSH) と Telnet によって、仮想端末セッションが作成されます。仮想端末の非アクティブセッションのタイムアウトおよびセッションの最大数制限を設定できます。

モデムのサポート

スーパーバイザ モジュールの COM1 ポートまたはコンソール ポートにモデムを接続できます。Cisco NX-OS ソフトウェアを実行するデバイスでは、次のモデムがテスト済みです。

- MultiTech MT2834BA (http://www.multitech.com/en_us/support/families/multimodemii/)
- Hayes Accura V.92 (http://www.zoom.com/products/dial_up_external_serial.html#hayes)



(注) デバイスの起動中は、モデムを接続しないでください。デバイスの電源がオンになってから、モデムを接続します。

Cisco NX-OS ソフトウェアには、接続されたモデムを検出するためのデフォルトの初期化文字列（ATE0Q1&D2&C1S0=1\015）が用意されています。デフォルトの文字列は次のように定義されています。

AT

アテンション

E0（必須）

エコーなし

Q1

結果コード オン

&D2

通常のデータ端末動作可能（DTR）オプション

&C1

データ キャリア状態の追跡がイネーブル

S0=1

1 回の呼び出し音の後に応答

\015（必須）

8 進数値での復帰

コンフィギュレーションファイルに対する仮想化のサポート

検出できないモジュールの設定を削除する場合を除いて、コンフィギュレーションファイルの操作は、仮想デバイスコンテキスト（VDC）に対してローカルです。検出できないモジュールの設定は、デフォルトの VDC からだけ削除できます。VDC の詳細については、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』を参照してください。

端末設定およびセッションのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	端末の設定にライセンスは不要です。ライセンス パッケージに含まれていない機能はすべて Cisco NX-OS システムイメージにバンドルされており、追加費用は一切発生しません。Cisco NX-OS ライセンス方式の詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

コンソールポートの設定

コンソールポートの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
- 速度
- ストップビット

はじめる前に

コンソールポートにログインします。

デフォルト VDC にいることを確認します。

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	line console 例： 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 例： switch(config-console)# databits 7	バイト当たりのデータビットの数を設定します。有効な範囲は 5～8 です。デフォルトは 8 です。
ステップ 4	exec-timeout minutes 例： 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。有効な範囲は 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。

	コマンドまたはアクション	目的
ステップ 5	parity {even none odd} 例： switch(config-console)# parity even	パリティを設定します。デフォルトは none です。
ステップ 6	speed {300 1200 2400 4800 9600 38400 57600 115200} 例： switch(config-console)# speed 115200	送受信の速度を設定します。デフォルト値は 115200 です。
ステップ 7	stopbits {1 2} 例： 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 例： 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	show line console 例： switch(config)# show line console	(任意) コンソールの設定を表示します。
ステップ 10	copy running-config startup-config 例： switch(config)# copy running-config startup-config	(任意) 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。

COM1 ポートの設定

COM1 ポートの次の特性を設定できます。

- データ ビット
- ハードウェア上のフロー制御
- パリティ
- 速度
- ストップ ビット

はじめる前に

コンソールポートまたは COM1 ポートにログインします。

デフォルト VDC にいることを確認します。

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	line com1 例： switch# line com1 switch(config-com1)#	COM1 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 例： switch(config-com1)# databits 7	バイト当たりのデータビットの数を設定します。有効な範囲は 5～8 です。デフォルトは 8 です。
ステップ 4	flowcontrol hardware 例： switch(config-com1)# flowcontrol hardware	ハードウェア上のフロー制御をイネーブルにします。デフォルトはイネーブルです。 ハードウェア上のフロー制御をディセーブルにするには、 no flowcontrol hardware コマンドを使用します。
ステップ 5	parity {even none odd} 例： switch(config-com1)# parity even	パリティを設定します。デフォルトは none です。
ステップ 6	speed {300 1200 2400 4800 9600 38400 57600 115200} 例： switch(config-com1)# speed 115200	送受信の速度を設定します。デフォルトは 9600 です。
ステップ 7	stopbits {1 2} 例： switch(config-com1)# 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 例： switch(config-com1)# exit switch(config)#	COM1 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 line com1 例： switch(config)# show line com1	(任意) COM1 ポートの設定を表示します。
ステップ 10	copy running-config startup-config 例： switch(config)# copy running-config startup-config	(任意) 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 デバイスに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のタイムアウトの設定

Cisco NX-OS 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。

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	line vty 例： 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 <i>minutes</i> 例： switch(config-line)# exec-timeout 30	VDC での非アクティブセッションのタイムアウトを設定します。有効な範囲は 0 ~ 525600 分 (8760 時間) です。0 分の値を指定すると、タイムアウトがディセーブルになります。デフォルト値は 30 です。

	コマンドまたはアクション	目的
ステップ 4	exit 例： 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	show running-config all begin vty 例： switch(config)# show running-config all begin vty	(任意) 仮想端末の設定を表示します。
ステップ 6	copy running-config startup-config 例： switch(config)# copy running-config startup-config	(任意) 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

Cisco NX-OS デバイス上の仮想端末セッションの数を制限できます。

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 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	line vty 例： 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 例： switch(config-line)# session-limit 10	Cisco NX-OS デバイスの仮想セッションの最大数を設定します。有効な範囲は 1 ~ 64 です。デフォルトは 32 です。
ステップ 4	exit 例： 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。

	コマンドまたはアクション	目的
ステップ 5	show running-config all begin vty 例： switch(config)# show running-config all begin vty	(任意) 仮想端末の設定を表示します。
ステップ 6	copy running-config startup-config 例： switch(config)# copy running-config startup-config	(任意) 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。

モデム接続の設定

COM1 ポートまたはコンソールポートにモデムを接続できます。

モデムの接続には、COM1 ポートを使用することを推奨します。

モデム接続のイネーブル化

モデムを使用するには、その前にポート上のモデム接続をイネーブルにする必要があります。

はじめる前に

コンソールポートにログインします。

デフォルト VDC にいることを確認します。

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	次のいずれかのコマンドを入力します。	
	コマンド	目的
	line com1	COM1 コンフィギュレーションモードを開始します。

	コマンドまたはアクション		目的
	コマンド	目的	
	line 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。	
	例： switch# line com1 switch(config-com1)#		
ステップ 3	modem in		COM1 またはコンソールポート上のモデム入力をイネーブルにします。
	例： switch(config-com1)# modem in		
ステップ 4	exit		COM1 またはコンソール コンフィギュレーションモードを終了します。
	例： switch(config-com1)# exit switch(config)#		
ステップ 5	show line		(任意) コンソールと COM1 の設定を表示します。
	例： switch(config)# show line		
ステップ 6	copy running-config startup-config		(任意) 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。
	例： switch(config)# copy running-config startup-config		

デフォルトの初期化文字列のダウンロード

Cisco NX-OS ソフトウェアでは、モデムでの接続用にダウンロードできるデフォルトの初期化文字列が用意されています。デフォルトの初期化文字列は ATE0Q1&D2&C1S0=1\015 です。

はじめる前に

コンソールポートにログインします。

デフォルト VDC にいることを確認します。

手順

	コマンドまたはアクション	目的						
ステップ 1	<p>configure terminal</p> <p>例 :</p> <pre>switch# configure terminal switch(config)#</pre>	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	<p>次のいずれかのコマンドを入力します。</p> <table border="1"> <thead> <tr> <th>オプション</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>line com1</td> <td>COM1 コンフィギュレーションモードを開始します。</td> </tr> <tr> <td>line console</td> <td>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。</td> </tr> </tbody> </table> <p>例 :</p> <pre>switch# line com1 switch(config-com1)#</pre>	オプション	説明	line com1	COM1 コンフィギュレーションモードを開始します。	line 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。	
オプション	説明							
line com1	COM1 コンフィギュレーションモードを開始します。							
line 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	<p>modem init-string default</p> <p>例 :</p> <pre>switch(config-com1)# modem init-string default</pre>	モデムにデフォルトの初期化文字列を書き込みます。						
ステップ 4	<p>exit</p> <p>例 :</p> <pre>switch(config-com1)# exit switch(config)#</pre>	COM1 またはコンソール コンフィギュレーションモードを終了します。						
ステップ 5	<p>show line</p> <p>例 :</p> <pre>switch(config)# show line</pre>	(任意) COM1 とコンソールの設定を表示します。						
ステップ 6	<p>copy running-config startup-config</p> <p>例 :</p> <pre>switch(config)# copy running-config startup-config</pre>	(任意) 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。						

ユーザ指定の初期化文字列の設定およびダウンロード

デフォルトの初期化文字列がモデムと互換性がない場合は、独自の初期化を設定してダウンロードできます。

はじめる前に

コンソールポートにログインします。

デフォルト VDC にいることを確認します。

手順

	コマンドまたはアクション	目的						
ステップ 1	configure terminal 例 : <pre>switch# configure terminal switch(config)#</pre>	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	次のいずれかのコマンドを入力します。 <table border="1" data-bbox="565 974 1019 1285"> <thead> <tr> <th>オプション</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>line com1</td> <td>COM1 コンフィギュレーション モードを開始します。</td> </tr> <tr> <td>line console</td> <td>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。</td> </tr> </tbody> </table> 例 : <pre>switch# line com1 switch(config-com1)#</pre>	オプション	説明	line com1	COM1 コンフィギュレーション モードを開始します。	line 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。	
オプション	説明							
line com1	COM1 コンフィギュレーション モードを開始します。							
line 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	modem set-string user-input <i>string</i> 例 : <pre>switch(config-com1)# modem set-string user-input ATE0Q1&D2&C1S0=3\015</pre>	COM1 またはコンソールポートのユーザ指定の初期化文字列を設定します。初期化文字列は、最大 100 文字の長さの英数字で、大文字と小文字が区別され、特殊文字を含むことができます。 (注) 文字列を初期化する前に、まず user-input の文字列を設定する必要があります。						

	コマンドまたはアクション	目的
ステップ 4	modem init-string user-input 例： switch(config-com1)# modem init-string user-input	COM1 またはコンソール ポートに接続されたモデムにユーザ指定の初期化文字列を書き込みます。
ステップ 5	exit 例： switch(config-com1)# exit switch(config)#	COM1 または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 6	show line 例： switch(config)# show line	(任意) COM1 とコンソールの設定を表示します。
ステップ 7	copy running-config startup-config 例： switch(config)# copy running-config startup-config	(任意) 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。

電源がオンになっている Cisco NX-OS デバイスのモデムの初期化

電源が入っている物理デバイスにモデムを接続する場合は、使用する前にモデムを初期化する必要があります。

はじめる前に

Cisco NX-OS デバイスがブートシーケンスの実行を完了し、システムイメージが実行されるまで待った後、デバイスの COM1 ポートまたはコンソールポートにモデムを接続します。

ポートでモデム接続をイネーブルにします。

デフォルト VDC にいることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	modem connect line {com1 console} 例： switch# modem connect line com1	デバイスに接続されたモデムを初期化します。

関連トピック

[モデム接続のイネーブル化, \(86 ページ\)](#)

端末セッションのクリア

Cisco NX-OS デバイス上の端末セッションをクリアできます。

手順

	コマンドまたはアクション	目的
ステップ 1	show users 例: switch# show users	(任意) デバイス上のユーザセッションを表示します。
ステップ 2	clear line name 例: switch# clear line pts/0	特定の回線上の端末セッションをクリアします。回線名は、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に関する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	端末の設定を表示します。
show line	COM1 ポートおよびコンソールポートの設定を表示します。
show users	仮想端末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。

これらのコマンドの出力フィールドの詳細については、ご使用のデバイスの『Cisco Nexus Command Reference Guide』を参照してください。

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。

表 16: ファイルシステムのデフォルト設定

パラメータ	デフォルト
デフォルトのファイルシステム	bootflash:

端末設定およびセッションに関する追加情報

ここでは、NX-OS デバイスでの端末設定とセッションに関する追加情報について説明します。

端末設定およびセッションの関連資料

関連項目	参照先
ライセンス	『Cisco NX-OS Licensing Guide』
コマンドリファレンス	『Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference』



第 7 章

基本的なデバイス管理

この章の内容は、次のとおりです。

- 機能情報の確認, 93 ページ
- 基本的なデバイス管理の概要, 94 ページ
- 基本的なデバイス管理のライセンス要件, 95 ページ
- デバイスのホスト名の変更, 95 ページ
- MOTD バナーの設定, 96 ページ
- タイムゾーンの設定, 97 ページ
- 夏時間の設定, 98 ページ
- デバイス クロックの手動設定, 99 ページ
- Clock Manager の設定, 100 ページ
- ユーザの管理, 101 ページ
- デバイス コンフィギュレーションの確認, 101 ページ
- 基本的なデバイス パラメータのデフォルト設定, 102 ページ
- 基本的なデバイス管理に関する追加情報, 102 ページ
- 基本的なデバイス管理機能の履歴, 102 ページ

機能情報の確認

ご使用のソフトウェア リリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細、および各機能がサポートされているリリースのリ

ストについては、「新機能および変更された機能に関する情報」の章または以下の「機能の履歴」表を参照してください。

基本的なデバイス管理の概要

ここでは、基本的なデバイス管理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) から別の文字列に変更できます。デバイスに一意的ホスト名を指定すると、コマンドライン インターフェイス (CLI) のプロンプトからデバイスを容易に識別できるようになります。

Message-of-the-Day バナー

Message-of-The-Day (MOTD) バナーは、デバイスのユーザ 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表示する情報を含めることができます。

デバイス クロック

NTP クロック ソースなど、有効な外部のタイミングメカニズムを使用してデバイスの時刻を同期させない場合は、デバイスの起動時にクロックの時刻を手動で設定できます。NTPの詳細については、『Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 6.x』を参照してください。

Clock Manager

Cisco Nexus シャーシには、同期化する必要のある異なるタイプのクロックが搭載されていることがあります。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、LC プロセッサ、またはラインカードなど) の一部で、それぞれ異なるプロトコルを使用している場合があります。

Clock Manager は、これらの異なるクロックを同期するための方法を提供します。

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。ユーザセッションおよびアカウントの管理の詳細については、ご使用のデバイスの『Cisco Nexus Security Configuration Guide』を参照してください。

基本的なデバイス管理に対する仮想化のサポート

基本的なデバイス管理は、仮想デバイス コンテキスト (VDC) に対してローカルです。VDC の詳細については、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』を参照してください。

基本的なデバイス管理のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	基本的なデバイス管理にはライセンスは不要です。ライセンス パッケージに含まれていない機能はすべて Cisco NX-OS システムイメージにバンドルされており、追加費用は一切発生しません。Cisco NX-OS ライセンス方式の詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

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) から別の文字列に変更できます。

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)#	グローバルコンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 2	<p>{hostname switchname} name</p> <p>例 :</p> <p>hostname コマンドの使用 :</p> <pre>switch(config)# hostname Engineering1 Engineering1(config)#</pre> <p>switchname コマンドの使用 :</p> <pre>Engineering1(config)# switchname Engineering2 Engineering2(config)#</pre>	<p>デバイスのホスト名を変更します。 <i>name</i> 引数には、大文字と小文字を区別して、最大 32 文字の英数字で値を指定します。デフォルトは switch です。</p> <p>(注) switchname コマンドは、hostname コマンドと同じ機能を実行します。</p>
ステップ 3	<p>exit</p> <p>例 :</p> <pre>Engineering2(config)# exit Engineering2#</pre>	<p>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。</p>
ステップ 4	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Engineering2# copy running-config startup-config</pre>	<p>(任意)</p> <p>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。</p>

MOTD バナーの設定

ユーザがログインしたときに端末にログインプロンプトの前に表示されるように MOTD を設定できます。 MOTD バナーには次の特性があります。

- 1 行当たり最大 80 文字
- 最大 40 行

手順

	コマンドまたはアクション	目的
ステップ 1	<p>configure terminal</p> <p>例 :</p> <pre>switch# configure terminal switch(config)#</pre>	<p>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。</p>

	コマンドまたはアクション	目的
ステップ 2	banner motd <i>delimiting-character</i> <i>message delimiting-character</i> 例： switch(config)# banner motd #Welcome to the Switch# switch(config)#	MOTD バナーを設定します。 <i>message</i> テキスト内では <i>delimiting-character</i> を使用しないでください。 (注) デリミタとして "および % は使用しないでください。
ステップ 3	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。
ステップ 4	show banner motd 例： switch# show banner motd	(任意) 設定された MOTD バナーを表示します。
ステップ 5	copy running-config startup-config 例： switch# copy running-config startup-config	(任意) 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

デバイスクロックの時刻を UTC からオフセットするようにタイムゾーンを設定できます。

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	clock timezone <i>zone-name</i> <i>offset-hours offset-minutes</i> 例： switch(config)# clock timezone EST -5 0	タイムゾーンを設定します。 <i>zone name</i> 引数は、タイムゾーンの略語を示す 3 文字の文字列 (PST や EST など) です。 <i>offset-hours</i> 引数は UTC からのオフセットであり、その範囲は -23 ~ 23 時間です。 <i>offset-minutes</i> 引数の範囲は 0 ~ 59 分です。

	コマンドまたはアクション	目的
ステップ 3	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。
ステップ 4	show clock 例： switch# show clock	(任意) 時刻とタイムゾーンを表示します。
ステップ 5	copy running-config startup-config 例： switch# copy running-config startup-config	(任意) 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。

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	clock summer-time zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes 例： 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。 <i>zone-name</i> 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。 <i>start-day</i> および <i>end-day</i> 引数の値は、 Monday 、 Tuesday 、 Wednesday 、 Thursday 、 Friday 、 Saturday 、および Sunday です。 <i>start-month</i> および <i>end-month</i> 引数の値は、 January 、 February 、 March 、 April 、 May 、 June 、 July 、 August 、 September 、 October 、 November 、および December です。 <i>start-time</i> および <i>end-time</i> 引数の値は、 <i>hh:mm</i> フォーマットです。

	コマンドまたはアクション	目的
		<i>offset-minutes</i> 引数の範囲は、0 ~ 1440 分です。
ステップ 3	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。
ステップ 4	show clock detail 例： switch(config)# show clock detail	(任意) 設定された MOTD バナーを表示します。
ステップ 5	copy running-config startup-config 例： switch# copy running-config startup-config	(任意) 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は、クロックを手動で設定できます。

はじめる前に

タイムゾーンを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	clock set <i>time day month year</i> 例： switch# clock set 15:00:00 30 May 2008 Fri May 30 15:14:00 PDT 2008	デバイス クロックを設定します。 <i>time</i> 引数の形式は、 <i>hh:mm:ss</i> です。 <i>day</i> 引数の範囲は 1 ~ 31 です。 <i>month</i> 引数の値は、 January 、 February 、 March 、 April 、 May 、 June 、 July 、 August 、 September 、 October 、 November 、および December です。 <i>year</i> 引数の範囲は 2000 ~ 2030 です。
ステップ 2	show clock 例： switch(config)# show clock	(任意) 現在のクロック値を表示します。

関連トピック

[タイムゾーンの設定, \(97 ページ\)](#)

Clock Manager の設定

Cisco Nexus シャーシ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 を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>clock protocol protocol vdc vdc-num</p> <p>例 :</p> <pre># clock protocol ptp vdc 2</pre>	<p>Clock Manager を設定します。</p> <p><i>protocol</i> 引数の値は ptp、ntp、および none です。次に、値について説明します。</p> <ul style="list-style-type: none"> • ptp : IEEE 1588 で説明されているように、高精度時間プロトコル (PTP) を使用してクロックを同期させます。 • ntp : ネットワーク タイム プロトコル (NTP) を使用してクロックを同期させます。 • none : clock set を使用してスーパーバイザクロックを設定します。 <p>(注) none が使用されている場合、指定の VDC のクロックが設定されている必要があります。</p> <p>(注) プロトコルが設定されたら、指定の VDC のクロックはそのプロトコルを使用する必要があります。</p> <p>たとえば、clock protocol ntp vdc 2 というコマンドを入力すると、VDC 2 で PTP が設定されます。</p> <p><i>vdc</i> 引数の範囲は、1 ~ 8 です。</p>
ステップ 2	<p>show run clock_manager</p> <p>例 :</p> <pre>#show run clock_manager</pre>	<p>(任意)</p> <p>Clock Manager の設定を表示します。</p>

ユーザの管理

デバイスにログインしているユーザに関する情報を表示し、これらのユーザにメッセージを送信できます。

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。

手順

	コマンドまたはアクション	目的
ステップ 1	show users 例： switch# show users	ユーザセッションを表示します。

ユーザへのメッセージ送信

デバイスの CLI を現在使用しているアクティブなユーザに、メッセージを送信できます。

手順

	コマンドまたはアクション	目的
ステップ 1	show users 例： switch# show users	(任意) アクティブなユーザセッションを表示します。
ステップ 2	send [session line] message-text 例： switch# send Reloading the device is 10 minutes!	アクティブなユーザすべて、または特定のユーザにメッセージを送信します。メッセージは最大 80 文字の英数字で、大文字と小文字を区別します。

デバイス コンフィギュレーションの確認

POAP を使用してデバイスのブートストラップ後の設定を確認するには、次のコマンドのいずれかを使用します。

コマンド	目的
show running-config	実行コンフィギュレーションを表示します。
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。

これらのコマンドの出力フィールドの詳細については、ご使用のデバイスの『Cisco Nexus Command Reference』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。

表 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 バナーのテキスト	User Access Verification
クロックのタイムゾーン	UTC

基本的なデバイス管理に関する追加情報

基本的なデバイス管理に関連したその他の情報を見つけることができます。

基本的なデバイス管理の関連資料

関連項目	参照先
ライセンス	『Cisco NX-OS Licensing Guide』
コマンドリファレンス	『Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference』

基本的なデバイス管理機能の履歴

次の表に、この機能のリリースの履歴を示します。

表 18: ファイル システム、ディレクトリ、およびファイルの機能の履歴

機能名	リリース	機能情報	
デバイス管理	4.0(1)	この機能が導入されました。	



第 8 章

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

この章の内容は、次のとおりです。

- 機能情報の確認, 105 ページ
- デバイスのファイルシステム、ディレクトリ、およびファイルの概要, 106 ページ
- ファイルシステム、ディレクトリ、およびファイルのライセンス要件, 108 ページ
- 外部フラッシュ デバイスのフォーマット, 108 ページ
- ディレクトリの操作, 109 ページ
- ファイルの使用, 112 ページ
- アーカイブ ファイルの操作, 117 ページ
- ファイルシステムの使用例, 120 ページ
- ファイルシステムパラメータのデフォルト設定, 124 ページ
- ファイルシステムに関する追加情報, 124 ページ
- ファイルシステムの機能の履歴, 124 ページ

機能情報の確認

ご使用のソフトウェアリリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェアリリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細、および各機能がサポートされているリリースのリストについては、「新機能および変更された機能に関する情報」の章または以下の「機能の履歴」表を参照してください。

デバイスのファイルシステム、ディレクトリ、およびファイルの概要

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、デバイスで参照できるファイルシステムを示します。

表 19: 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
bootflash	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の保存に使用する、アクティブスーパーバイザモジュールにある内部 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の保存に使用する、スタンバイスーパーバイザモジュールにある内部 CompactFlash メモリ。
slot0	—	システムイメージ、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の保存に使用する、スーパーバイザモジュールに取り付けられた外部 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)。

ファイルシステム名	モジュール	説明
nvram	—	スタートアップ コンフィギュレーション ファイルの保存に使用する、スーパーバイザ モジュールにある不揮発性 RAM (NVRAM)。
log	—	ログ ファイルの統計情報を保存するアクティブ スーパーバイザにあるメモリ。
system	—	実行コンフィギュレーション ファイルの保存に使用する、スーパーバイザ モジュールにあるメモリ。
debug	—	デバッグログ用に使用するスーパーバイザ モジュールのメモリ。
usb1	—	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の保存に使用する、スーパーバイザ モジュールに取り付けられた外部 USB フラッシュ メモリ。
usb2	—	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の保存に使用する、スーパーバイザ モジュールに取り付けられた外部 USB フラッシュ メモリ。

ディレクトリ

bootflash: および外部フラッシュメモリ (slot0:、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、ファイルの保存用に使用できます。

ファイル

bootflash: にファイルを作成し、アクセスします。volatile:、slot0:、usb1:、およびusb2: ファイルシステム。system: システムファイルにあるファイルには、アクセスすることだけできます。**debug logfile** コマンドで指定したデバッグログファイル用には、debug: ファイルシステムを使用できます。

FTP、Secure Copy (SCP)、Secure Shell FTP (SFTP)、およびTFTPを使用して、リモートサーバからシステム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サーバとして動作できるので、外部サーバからデバイスへファイルをコピーすることもできます。

ファイルシステムに対する仮想化のサポート

ファイルシステム、ディレクトリ、ファイルのほとんどの設定および操作は、仮想デバイスコンテキスト (VDC) に対してローカルです。1つの例外として、外部フラッシュデバイスのフォーマットがあり、これはデフォルトのVDCから実行する必要があります。VDCの詳細については、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』を参照してください。

ファイルシステム、ディレクトリ、およびファイルのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	ファイルシステム、ディレクトリ、およびファイルを使用するためにライセンスは必要ありません。ライセンスパッケージに含まれていない機能はすべてCisco NX-OSシステムイメージにバンドルされており、追加費用は一切発生しません。Cisco NX-OSのライセンススキームの詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

外部フラッシュ デバイスのフォーマット

外部フラッシュデバイスをフォーマットすると、デフォルトVDCから内容が消去され、工場出荷時の状態に復元されます。



(注) 破損したブートフラッシュをフォーマットを使用して回復させる方法については、『[Cisco Nexus 7000 Series NX-OS Troubleshooting Guide](#)』を参照してください。

はじめる前に

デフォルト VDC にいることを確認します。

アクティブなスーパーバイザ モジュールに外部フラッシュ デバイスを挿入します。

手順

	コマンドまたはアクション	目的
ステップ 1	dir {slot0: usb1: usb2:} 例： switch# dir slot0:	(任意) 外部フラッシュ デバイスの内容を表示します。
ステップ 2	format {slot0: usb1: usb2:} 例： switch# format slot0:	外部フラッシュ デバイスをフォーマットします。

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方法について説明します。

カレント ディレクトリの識別

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。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： 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

ファイルシステム操作での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。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： switch# pwd	(任意) 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]} 例： 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の名前は、大文字と小文字が区別されます。

ディレクトリの作成

`bootflash:` ファイルシステムおよびフラッシュ デバイスのファイルシステム内にディレクトリを作成できます。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： switch# pwd	(任意) 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]} 例： switch# cd slot0:	(任意) 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の名前は、大文字と小文字が区別されます。
ステップ 3	mkdir [filesystem:[//module/]directory] 例： switch# mkdir test	新しいディレクトリを作成します。 <i>filesystem</i> 引数は、大文字と小文字が区別されます。 <i>directory</i> 引数には、大文字と小文字を区別して、最大 64 文字の英数字で値を指定します。

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。

手順

	コマンドまたはアクション	目的
ステップ 1	dir [<i>directory</i> <i>filesystem</i> : <i>[//module][directory]</i>] 例： switch# dir bootflash:test	ディレクトリの内容を表示します。デフォルトは現在の作業ディレクトリです。ファイルシステムとディレクトリの名前は、大文字と小文字が区別されます。

ディレクトリの削除

デバイス上のファイルシステムからディレクトリを削除できます。

はじめる前に

ディレクトリを削除しようとする前に、そのディレクトリが空であることを確認してください。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： switch# pwd	(任意) 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	dir [<i>filesystem</i> <i>filesystem</i> : <i>[//module][directory]</i>] 例： switch# dir bootflash:test	(任意) 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の名前は、大文字と小文字が区別されます。 ディレクトリが空でない場合は、そのディレクトリを削除する前に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 [<i>filesystem</i> <i>filesystem</i> : <i>[//module][directory]</i>] 例： switch# rmdir test	ディレクトリを削除します。ファイルシステムとディレクトリの名前は、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

スタンバイ スーパーバイザ モジュール（リモート）のすべてのファイルシステムに、アクティブ スーパーバイザ モジュールのセッションからアクセスできます。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する際に、同様のファイルがスタンバイ スーパーバイザ モジュールにある必要がある場合に役立ちます。スタンバイ スーパーバイザ モジュールのファイルシステムにアクティブ スーパーバイザ モジュール セッションからアクセスするには、ファイルのパスで `filesystem://sup-remote/` または `filesystem://sup-standby/` のいずれかを使用してスタンバイ スーパーバイザ モジュールを指定する必要があります。

ファイルの使用

ここでは、Cisco NX-OS デバイスでファイル进行操作する方法について説明します。

ファイルの移動

あるディレクトリから別のディレクトリにファイルを移動させることができます。



注意

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。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： switch# pwd	(任意) 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	dir [filesystem://module/][directory] 例： switch# dir bootflash	(任意) カレントディレクトリの内容を表示します。ファイルシステムとディレクトリの名前は、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 /]	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の名前は、大文字と小文字が区別されます。

	コマンドまたはアクション	目的
	<code> directory/}[target-filename] target-filename}</code> 例： <pre>switch# move test old_tests/test1</pre>	<i>target-filename</i> 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。 <i>target-filename</i> 引数が指定されていない場合、ファイル名は、デフォルトで <i>source-filename</i> 引数の値になります。

ファイルのコピー

ファイルのコピーを同じディレクトリ内、または別のディレクトリのどちらにも作成できます。



(注) **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。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： <pre>switch# pwd</pre>	(任意) 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	dir <code>[filesystem:][module/][directory/]</code> 例： <pre>switch# dir bootflash</pre>	(任意) カレントディレクトリの内容を表示します。ファイルシステムとディレクトリの名前は、大文字と小文字が区別されます。
ステップ 3	copy <code>[filesystem:][module/][directory/]</code> <code> directory/}source-filename </code> <code>{filesystem:}[module/][directory/]</code> <code> directory/}[target-filename]</code> 例： <pre>switch# move test old_tests/test1</pre>	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の名前は、大文字と小文字が区別されます。 <i>source-filename</i> 引数には、大文字と小文字を区別して、最大 64 文字の英数字で値を指定します。 <i>target-filename</i> 引数が指定されていない場合、ファイル名は、デフォルトで <i>source-filename</i> 引数の値になります。 copy コマンドは、 ftp 、 scp 、 sftp 、 tftp 、および http プロトコルをサポートします。

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。

手順

	コマンドまたはアクション	目的
ステップ 1	dir <code>[filesystem://module/][directory]</code> 例： <code>switch# dir bootflash</code>	(任意) カレントディレクトリの内容を表示します。ファイルシステムとディレクトリの名前は、大文字と小文字が区別されます。
ステップ 2	delete <code>{filesystem://module/}[directory/] directory/}filename</code> 例： <code>switch# move test old_tests/test1</code>	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の名前は、大文字と小文字が区別されます。 <code>source-filename</code> 引数は、大文字と小文字が区別されます。 注意 ディレクトリを指定すると、 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。

ファイルの内容の表示

ファイルの内容を表示できます。

手順

	コマンドまたはアクション	目的
ステップ 1	show file <code>[filesystem://module/][directory/]filename</code> 例： <code>switch# show file bootflash:test-results</code>	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ためのチェックサムを表示できます。

手順

	コマンドまたはアクション	目的
ステップ 1	show file <code>[filesystem://module/][directory/]filename {cksum md5sum}</code> 例： 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。

ファイルの圧縮および圧縮解除

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

手順

	コマンドまたはアクション	目的
ステップ 1	dir <code>[filesystem://module/]directory]</code> 例： switch# dir bootflash:	(任意) カレントディレクトリの内容を表示します。ファイルシステムとディレクトリの名前は、大文字と小文字が区別されます。
ステップ 2	gzip <code>[filesystem://module/][directory/] directory/]filename</code> 例： 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、そのサフィックスは <code>.gz</code> になります。
ステップ 3	gunzip <code>[filesystem://module/][directory/] directory/]filename .gz</code> 例： 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ックスは <code>.gz</code> である必要があります。ファイルが圧縮解除されると、そのサフィックスは <code>.gz</code> でなくなります。

ファイルの最後の行の表示

ファイルの最後の行を表示できます。

手順

	コマンドまたはアクション	目的
ステップ 1	tail [filesystem:[//module/]][directory/]filename [lines] 例 : 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 行です。範囲は 0 ~ 80 行です。

show コマンドの出力のファイルへのリダイレクト

show コマンドの出力を、bootflash:、slot0:、volatile:、またはリモート サーバのファイルにリダイレクトできます。また、コマンドの出力形式も指定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	terminal redirection-mode {ascii zipped} 例 : switch# terminal redirection-mode zipped	(任意) ユーザ 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 例 : 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

特定の文字列で始まる名前のファイルを、現在の作業ディレクトリおよびサブディレクトリから検索できます。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： switch# pwd	(任意) 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 {filesystem:[//module/] [directory] directory} 例： switch# cd bootflash:test_scripts	(任意) デフォルトディレクトリを変更します。
ステップ 3	find filename-prefix 例： switch# find bgp_script	指定したファイル名プレフィックスで始まる名前を持つすべてのファイルを、デフォルトディレクトリとサブディレクトリから検索します。ファイル名プレフィックスは、大文字と小文字が区別されます。

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の作成、既存のアーカイブファイルへのファイルの追加、アーカイブファイルからのファイルの抽出、アーカイブファイル内のファイルの一覧表示ができます。

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
- Uncompressed

デフォルト値は gzip です。

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: } <i>archive-filename</i> [absolute] [bz2-compress] [gz-compress] [remove] [uncompressed] [verbose] <i>filename-list</i>	<p>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。</p> <p>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。</p> <p>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）に使用する圧縮ユーティリティと、ファイルを抽出するとき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。</p> <ul style="list-style-type: none"> • 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。 <p>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。</p> <p>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せん。</p>

次に、gzip 圧縮アーカイブ ファイルを作成する例を示します。

```
switch# tar create bootflash:config-archive gz-compress bootflash:config-file
```

アーカイブ ファイルへのファイルの追加

Cisco NX-OS 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。

はじめる前に

Cisco NX-OS デバイスでアーカイブ ファイルを作成しました。

手順

	コマンドまたはアクション	目的
ステップ 1	<code>tar append {bootflash: volatile:} archive-filename [absolute] [remove] [verbose] filename-list</code>	<p>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の名前は、大文字と小文字が区別されません。</p> <p>absolute キーワードは、アーカイブファイルに追加されたファイルの名前から先頭のバックスラッシュ文字 (\) を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。</p> <p>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。</p> <p>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。デフォルトでは、追加されたファイルが一覧表示されます。</p>

次の例では、既存のアーカイブファイルにファイルを追加する方法を示します。

```
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

ファイルは、Cisco NX-OS デバイス上の既存のアーカイブファイルに抽出できます。

はじめる前に

Cisco NX-OS デバイスでアーカイブファイルを作成しました。

手順

	コマンドまたはアクション	目的
ステップ 1	<code>tar extract {bootflash: volatile:} archive-filename [keep-old] [screen] [to {bootflash: volatile:}[/directory-name]] [verbose]</code>	<p>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の名前は、大文字と小文字が区別されません。</p> <p>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。</p>

コマンドまたはアクション	目的
	<p>screen キーワードは、Cisco NX-OS ソフトウェアが、抽出されたファイルの内容を端末画面に表示することを指定します。</p> <p>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。</p> <p>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。</p>

次に、既存のアーカイブファイルからファイルを抽出する例を示します。

```
switch# tar extract bootflash:config-archive.tar.gz
```

アーカイブファイル内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の名前は、大文字と小文字が区別されません。

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

ファイルシステムの使用例

ここでは、Cisco NX-OS デバイスのファイルシステムを使用する例を示します。

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 モジュールにあるファイルをリスト表示する例を示します。

```
switch# dir bootflash://sup-remote
12198912    Aug 27 16:29:18 2003  m9500-sflek9-kickstart-mzg.1.3.0.39a.bin
1864931    Apr 29 12:41:59 2003  dplug2
12288      Apr 18 20:23:11 2003  lost+found/
12097024   Nov 21 16:34:18 2003  m9500-sflek9-kickstart-mz.1.3.1.1.bin
41574014   Nov 21 16:34:47 2003  m9500-sflek9-mz.1.3.1.1.bin

Usage for bootflash://sup-remote
67747169 bytes used
```

```
116812447 bytes free
184559616 bytes total
```

次に、スタンバイ スーパーバイザ モジュールにある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 デバイスでファイルを移動する例を示します。

```
switch# move slot0:samplefile slot0:mystorage/samplefile
```

次に、デフォルトのファイル システムで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、slot0: 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 slot0:samplefile slot0: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:system_image bootflash://sup-2/system_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、slot0: または bootflash: ファイル システムと、FTP、TFTP、SFTP、または SCP サーバ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

デバイス上のファイル システムからディレクトリを削除できます。

はじめる前に

ディレクトリを削除しようとする前に、そのディレクトリが空であることを確認してください。

手順

	コマンドまたはアクション	目的
ステップ 1	pwd 例： switch# pwd	(任意) 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	dir [<i>filesystem</i> : <i>[/module/][directory]</i>] 例： switch# dir bootflash:test	(任意) 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の名前は、大文字と小文字が区別されます。 ディレクトリが空でない場合は、そのディレクトリを削除する前に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 [<i>filesystem</i> : <i>[/module/][directory]</i>] 例： switch# rmdir test	ディレクトリを削除します。ファイルシステムとディレクトリの名前は、大文字と小文字が区別されます。

ファイルの内容の表示

次に、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 slot0:test
configure terminal
interface ethernet 1/1
no shutdown
end
show interface ethernet 1/1
```

次に、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
```

ファイルの圧縮および圧縮解除

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 2003 Samplefile
...
switch# gzip volatile:Samplefile
switch# dir
266069 Jul 04 00:51:03 2003 Samplefile.gz
...
```

次に、圧縮ファイルを圧縮解除する例を示します。

```
switch# dir
266069 Jul 04 00:51:03 2003 Samplefile.gz
...
switch# gunzip samplefile
switch# dir
1525859 Jul 04 00:51:03 2003 Samplefile
...
```

show コマンドの出力のリダイレクト

次に、出力を bootflash: ファイルシステムのファイルにリダイレクトする方法を示します。

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

次に、出力を外部フラッシュメモリのファイルにリダイレクトする方法を示します。

```
switch# show interface > slot0:switch-intf.cfg
```

次に、出力を TFTP サーバのファイルにリダイレクトする方法を示します。

```
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 2003 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からファイルを検索する方法を示します。

```
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
```

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。

表 20: ファイルシステムのデフォルト設定

パラメータ	デフォルト
デフォルトのファイルシステム	bootflash:

ファイルシステムに関する追加情報

ここでは、ファイルシステムに関するその他の情報について説明します。

ファイルシステムの関連資料

関連項目	参照先
ライセンス	『Cisco NX-OS Licensing Guide』
コマンドリファレンス	『Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference』

ファイルシステムの機能の履歴

次の表に、この機能のリリースの履歴を示します。

表 21: ファイルシステム、ディレクトリ、およびファイルの機能の履歴

機能名	リリース	機能情報
show コマンドの出力のリダイレクション	4.2(1)	ファイルにリダイレクトする場合の show コマンドの出力形式を変更できます。形式はASCII形式、または zip 形式です。



第 9 章

コンフィギュレーションファイルの操作

この章の内容は、次のとおりです。

- 機能情報の確認, 127 ページ
- コンフィギュレーションファイルの概要, 127 ページ
- コンフィギュレーションファイルのライセンス要件, 128 ページ
- コンフィギュレーションファイルの管理, 129 ページ
- デバイス コンフィギュレーションの確認, 140 ページ
- コンフィギュレーションファイルの操作例, 141 ページ
- コンフィギュレーションファイルに関する追加情報, 142 ページ
- コンフィギュレーションファイルの機能の履歴, 142 ページ

機能情報の確認

ご使用のソフトウェアリリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェアリリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細、および各機能がサポートされているリリースのリストについては、「新機能および変更された機能に関する情報」の章または以下の「機能の履歴」表を参照してください。

コンフィギュレーションファイルの概要

コンフィギュレーションファイルには、Cisco NX-OS デバイスの設定に使用する Cisco NX-OS ソフトウェアのコマンドが含まれます。コマンドは、システムを起動したとき（スタートアップコンフィギュレーションファイルから）、またはコンフィギュレーションモードで 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 コンフィギュレーション 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 コンフィギュレーションに実行コンフィギュレーション ファイルを保存するか、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へコンフィギュレーション 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 ソフトウェアのコンフィギュレーション ファイルには、実行コンフィギュレーションとスタートアップ コンフィギュレーションの2種類があります。デバイスは、デバイスの起動中にスタートアップ 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 ファイルに加えた現在の変更が含まれています。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。この場合は、グローバルコンフィギュレーションモードのコマンドを使用して実行コンフィギュレーションを変更しますが、それらの変更をスタートアップ コンフィギュレーションには保存しません。

実行コンフィギュレーションを変更するには、**configure terminal** コマンドを使用して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。Cisco NX-OS コンフィギュレーションモードの使用時には通常、コマンドはすぐに実行され、入力直後またはコンフィギュレーションモードを終了した時点で実行コンフィギュレーション 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、または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。

関連トピック

[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、\(129 ページ\)](#)

[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、\(131 ページ\)](#)

コンフィギュレーション ファイルに対する仮想化のサポート

検出できないモジュールの設定を削除する場合を除いて、コンフィギュレーション ファイルの操作は、仮想デバイス コンテキスト (VDC) に対してローカルです。検出できないモジュールの設定は、デフォルトの VDC からだけ削除できます。VDC の詳細については、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』を参照してください。

コンフィギュレーション ファイルのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	コンフィギュレーションファイルにはライセンスは不要です。ライセンスパッケージに含まれていない機能はすべて Cisco NX-OS システムイメージにバンドルされており、追加費用は一切発生しません。Cisco NX-OS ライセンス方式の詳細については、『Cisco NX-OS Licensing Guide』を参照してください。

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して、デバイスを次にリロードするときのために変更を保存することができます。

物理デバイス上のすべてのVDCに対する実行コンフィギュレーションの保存については、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide』を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	show running-config 例： switch# show running-config	(任意) 実行コンフィギュレーションを表示します。
ステップ 2	copy running-config startup-config 例： 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。

コンフィギュレーションファイルのリモートサーバへのコピー

内部メモリに保存されている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、または他の Cisco NX-OS デバイスの設定に使用するためにリモートサーバにコピーできます。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>copy running-config <i>scheme://server/[url]/filename</i></p> <p>例： switch# copy running-config tftp://10.10.1.1/sw1-run-config.bak</p>	<p>実行コンフィギュレーションファイルのリモートサーバにコピーします。</p> <p><i>scheme</i> 引数には、tftp:、ftp:、scp:、または sftp: を入力できます。<i>server</i> 引数は、リモートサーバのアドレスまたは名前であり、<i>url</i> 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。</p> <p><i>server</i>、<i>url</i>、および <i>filename</i> の各引数は、大文字小文字を区別して入力します。</p>
ステップ 2	<p>copy startup-config <i>scheme://server/[url]/filename</i></p> <p>例： switch# copy startup-config tftp://10.10.1.1/sw1-start-config.bak</p>	<p>スタートアップコンフィギュレーションファイルのリモートサーバにコピーします。</p> <p><i>scheme</i> 引数には、tftp:、ftp:、scp:、または sftp: を入力できます。<i>server</i> 引数は、リモートサーバのアドレスまたは名前であり、<i>url</i> 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。</p> <p><i>server</i>、<i>url</i>、および <i>filename</i> の各引数は、大文字小文字を区別して入力します。</p>

次に、リモートサーバへコンフィギュレーションファイルの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。リモートサーバからデバイスの実行コンフィギュレーションにファイルをダウンロードするには、TFTP、FTP、セキュアコピー（SCP）、またはセキュアシェル FTP（SFTP）を使用します。

はじめる前に

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。

Cisco NX-OS デバイスがリモートサーバにルーティングされることを確認します。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、Cisco NX-OS デバイスとリモートサーバは同一のサブネットワーク内にある必要があります。

ping または **ping6** コマンドを使用して、リモートサーバへの接続を確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	copy scheme://server[url]/filename running-config 例： <pre>switch# copy tftp://10.10.1.1/my-config running-config</pre>	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。 <i>scheme</i> 引数として、 tftp: 、 ftp: 、 scp: 、または sftp: を指定できます。 <i>server</i> 引数は、リモートサーバのアドレスまたは名前であり、 <i>url</i> 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。 <i>server</i> 、 <i>url</i> 、および <i>filename</i> 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	show running-config 例： <pre>switch# show running-config</pre>	(任意) 実行コンフィギュレーションを表示します。
ステップ 3	copy running-config startup-config 例： <pre>switch# copy running-config startup-config</pre>	(任意) 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	show startup-config 例： <pre>switch# show startup-config</pre>	(任意) スタートアップコンフィギュレーションを表示します。

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。リモートサーバからデバイスのスタートアップコンフィギュレーションにファイルをダウンロードするには、TFTP、FTP、セキュアコピー (SCP)、またはセキュアシェル FTP (SFTP) を使用します。



注意

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。

はじめる前に

コンソール ポートで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。

Cisco NX-OS デバイスがリモートサーバにルーティングされることを確認します。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、Cisco NX-OS デバイスとリモートサーバは同一のサブネットワーク内にある必要があります。

ping または **ping6** コマンドを使用して、リモートサーバへの接続を確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>write erase</p> <p>例 :</p> <pre>switch# write erase</pre>	<p>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。</p>
ステップ 2	<p>reload</p> <p>例 :</p> <pre>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#</pre>	<p>Cisco NX-OS デバイスをリロードします。</p> <p>(注) セットアップユーティリティはデバイスの設定に使用しないでください。</p> <p>(注) デフォルトで、reload コマンドは、スタートアップコンフィギュレーションのバイナリバージョンからデバイスをリロードします。</p> <p>Cisco NX-OS 6.2(2) 以降、reload ascii コマンドを使用して、デバイスのリロード時にコンフィギュレーションの ASCII バージ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できます。</p>
ステップ 3	<p>copy scheme://server/[url /]filename running-config</p> <p>例 :</p> <pre>switch# copy tftp://10.10.1.1/my-config running-config</pre>	<p>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。</p> <p><i>scheme</i> 引数として、tftp:、ftp:、scp:、または sftp: を指定できます。 <i>server</i> 引数は、リモートサーバのアドレスまたは名前であり、<i>url</i> 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。</p> <p><i>server</i>、<i>url</i>、および <i>filename</i> の各引数は、大文字小文字を区別して入力します。</p>

	コマンドまたはアクション	目的
ステップ 4	copy running-config startup-config 例： switch# copy running-config startup-config	実行コンフィギュレーションファイルを開始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	show startup-config 例： switch# show startup-config	(任意) 実行コンフィギュレーションを表示します。

コンフィギュレーションファイルの外部フラッシュメモリデバイスへのコピー

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、または後で使用するために、外部フラッシュメモリデバイスにコピーできます。

はじめる前に

アクティブなスーパーバイザモジュールに外部フラッシュメモリデバイスを挿入します。

手順

	コマンドまたはアクション	目的
ステップ 1	dir {slot0: usb1: usb2:}[directory/] 例： switch# dir slot0:	(任意) 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 running-config {slot0: usb1: usb2:}[directory/]filename 例： switch# copy running-config slot0:dsn-running-config.cfg	実行コンフィギュレーションを外部フラッシュメモリデバイスにコピーします。 <i>filename</i> 引数は、大文字と小文字が区別されます。
ステップ 3	copy startup-config {slot0: usb1: usb2:}[directory/]filename 例： switch# copy startup-config slot0:dsn-startup-config.cfg	スタートアップコンフィギュレーションを外部フラッシュメモリデバイスにコピーします。 <i>filename</i> 引数は、大文字と小文字が区別されます。

外部フラッシュメモリデバイスからの実行コンフィギュレーションのコピー

Cisco NX-OS デバイスを設定するには、別の Cisco NX-OS デバイスで作成され、外部フラッシュメモリ デバイスに保存されたコンフィギュレーション ファイルをコピーします。

はじめる前に

アクティブなスーパーバイザ モジュールに外部フラッシュメモリ デバイスを挿入します。

手順

	コマンドまたはアクション	目的
ステップ 1	dir {slot0: usb1: usb2:}[directory/] 例： switch# dir slot0:	(任意) 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 {slot0: usb1: usb2:}[directory/]filename running-config 例： switch# copy slot0: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。 <i>filename</i> 引数は、大文字と小文字が区別されます。
ステップ 3	show running-config 例： switch# show running-config	(任意) 実行コンフィギュレーションを表示します。
ステップ 4	copy running-config startup-config 例： switch# copy running-config startup-config	(任意) 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	show startup-config 例： switch# show startup-config	(任意) スタートアップコンフィギュレーションを表示します。

外部フラッシュメモリ デバイスからのスタートアップコンフィギュレーションのコピー

Cisco NX-OS デバイス上のスタートアップコンフィギュレーションを復元するには、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。

はじめる前に

アクティブなスーパーバイザ モジュールに外部フラッシュメモリ デバイスを挿入します。

手順

	コマンドまたはアクション	目的
ステップ 1	dir {slot0: usb1: usb2:}[directory/] 例： switch# dir slot0:	(任意) 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 {slot0: usb1: usb2:}[directory /]filename startup-config 例： switch# copy slot0: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 引数は、大文字と小文字が区別されます。
ステップ 3	show startup-config 例： switch# show startup-config	(任意) スタートアップコンフィギュレーションを表示します。

コンフィギュレーションファイルの内部ファイルシステムへのコピー

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、または後で使用するために、内部メモリにコピーできます。

手順

	コマンドまたはアクション	目的
ステップ 1	copy running-config [filesystem:][directory/] [directory/]filename	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 引数は、大文字と小文字が区別されます。

	コマンドまたはアクション	目的
	例： switch# copy running-config bootflash:sw1-run-config.bak	
ステップ 2	copy startup-config [filesystem:][directory/] [directory/]filename 例： 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 引数は、大文字と小文字が区別されます。

関連トピック

[ファイルのコピー](#), (113 ページ)

以前のコンフィギュレーションへのロールバック

メモリの破損などの問題によって、バックアップされているバージョンからコンフィギュレーションを回復する必要が生じる場合があります。



(注)

copy running-config startup-config コマンドを入力するたびに、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。**write erase** コマンドを実行すると、バイナリファイルが消去されます。

手順

	コマンドまたはアクション	目的
ステップ 1	write erase 例： switch# write erase	スイッチの現在のコンフィギュレーションを消去します。
ステップ 2	reload 例： switch# reload	デバイスを再起動します。デバイスが起動して実行できるように、キックスタートおよびシステムイメージファイルを入力するように求められます。

	コマンドまたはアクション	目的
		<p>(注) デフォルトで、reload コマンドは、スタートアップ コンフィギュレーションのバイナリバージョンからデバイスをリロードします。</p> <p>Cisco NX-OS 6.2(2) 以降、reload ascii コマンドを使用して、デバイスのリロード時にコンフィギュレーションの ASCII バージ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できます。</p>
ステップ 3	<p>copy configuration_file running-configuration</p> <p>例： switch# copy bootflash:start-config.bak running-configuration</p>	<p>以前に保存し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。</p> <p>(注) <i>configuration_file</i> ファイル名引数では、大文字と小文字が区別されます。</p>
ステップ 4	<p>copy running-config startup-config</p> <p>例： switch# copy running-config startup-config</p>	<p>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。</p>

検出できないモジュールの設定の削除

シャーシから I/O モジュールを取り外すときには、そのモジュールの設定を実行コンフィギュレーションから削除することもできます。デフォルトの VDC から削除できるのは、検出できないモジュールの設定だけです。



(注) 削除できるのは、シャーシの空きスロットの設定だけです。

はじめる前に

デフォルト VDC にいることを確認します。

シャーシから I/O モジュールを取り外します。

手順

	コマンドまたはアクション	目的
ステップ 1	show hardware 例： switch# show hardware	(任意) デバイスに取り付けられているハードウェアを表示します。
ステップ 2	purge module slot running-config 例： switch# purge module 3 running-config	検出できないモジュールの設定を実行コンフィギュレーションから削除します。
ステップ 3	copy running-config startup-config 例： switch# copy running-config startup-config	(任意) 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。

コンフィギュレーションの削除

デバイスのコンフィギュレーションを削除して、工場出荷時のデフォルト状態に戻すことができます。

デバイスの永続的メモリに保存されている次のコンフィギュレーションファイルを削除できます。

- スタートアップ
- ブート
- デバッグ



(注) **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コンフィギュレーションが削除されます。

- ブート変数定義
- 次のような **mgmt0** インターフェイスの IPv4 設定
 - アドレス
 - サブネットマスク
 - 管理 VRF のルートアドレス

mgmt0 インターフェイスのブート変数定義と IPv4 コンフィギュレーションを削除するには、**write erase boot** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>write erase [boot debug]</p> <p>例 :</p> <pre>switch# write erase Warning: This command will erase the startup-configuration. Do you wish to proceed anyway? (y/n) [n] y</pre>	<p>永続的メモリから設定を削除します。デフォルトのアクションでは、スタートアップコンフィギュレーションが削除されます。</p> <p>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 コンフィギュレーションが削除されます。</p> <p>debug オプションを使用すると、デバッグコンフィギュレーションが削除されます。</p> <p>(注) 実行コンフィギュレーションファイルは、このコマンドによって影響を受けません。</p>

非アクティブな設定のクリア

非アクティブな Quality of Service (QoS) または Access Control Lists (ACL) の設定をクリアできます。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>show running-config <i>type</i> inactive-if-config</p> <p>例 :</p> <pre># show running-config ipqos inactive-if-config</pre>	<p>(任意)</p> <p>非アクティブな ACL または QoS の設定をすべて表示します。</p> <p><i>type</i> 引数の値は、aclmgr および ipqos です。</p> <ul style="list-style-type: none"> • aclmgr : aclmgr の非アクティブな設定を表示します。 • ipqos : qosmgr の非アクティブな設定を表示します。
ステップ 2	<p>clear inactive-config <i>policy</i></p> <p>例 :</p> <pre># 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 default & for other than default vdc: /bootflash/vdc_x/qos_inactive_if_config.cfg (where x is vdc number) you can see the log file @ show inactive-if-config log</pre>	<p>非アクティブな設定をクリアします。</p> <p><i>policy</i> 引数の値は、qos および acl です。次に、値について説明します。</p> <ul style="list-style-type: none"> • qos : 非アクティブな QoS 設定をクリアします。 • acl : 非アクティブな ACL 設定をクリアします。 • acl qos : 非アクティブな ACL 設定および非アクティブな QoS 設定をクリアします。
ステップ 3	<p>show inactive-if-config log</p> <p>例 :</p> <pre># show inactive-if-config log</pre>	<p>(任意)</p> <p>非アクティブな設定をクリアするために使用されたコマンドを表示します。</p>

デバイス コンフィギュレーションの確認

POAP を使用してデバイスのブートストラップ後の設定を確認するには、次のコマンドのいずれかを使用します。

コマンド	目的
show running-config	実行コンフィギュレーションを表示します。

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。

これらのコマンドの出力フィールドの詳細については、ご使用のデバイスの『Cisco Nexus Command Reference』を参照してください。

コンフィギュレーションファイルの操作例

ここでは、コンフィギュレーションファイルを操作する例について説明します。

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
```

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

次に、デバイスの事前に定義した場所にスタートアップコンフィギュレーションのスナップショットを作成する方法を示します (バイナリ ファイル)。

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

次に、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
```

次に、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
```

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
```

以前のコンフィギュレーションへのロールバック

設定を以前に保存された設定のスナップショット コピーにロールバックするには、次の手順を実行してください。

- 1 **write erase** コマンドを使用して現在の実行イメージをクリアします。
- 2 **reload** コマンドを使用してデバイスを再起動します。



(注)

デフォルトで、**reload** コマンドは、スタートアップ コンフィギュレーションのバイナリ バージョンからデバイスをリロードします。

Cisco NX-OS 6.2(2) 以降、**reload ascii** コマンドを使用して、デバイスのリロード時にコンフィギュレーションの ASCII バージ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できます。

- 3 **copy configuration_file running-configuration** コマンドで、以前保存し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。
- 4 **copy running-config startup-config** コマンドを使用して、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。

コンフィギュレーション ファイルに関する追加情報

ここでは、コンフィギュレーション ファイルの管理に関するその他の情報について説明します。

コンフィギュレーション ファイルの関連資料

関連項目	参照先
ライセンス	『Cisco NX-OS Licensing Guide』
コマンド リファレンス	『Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference』

コンフィギュレーション ファイルの機能の履歴

次の表に、この機能のリリースの履歴を示します。

表 22: コンフィギュレーションファイルの機能の履歴

機能名	リリース	機能情報
コンフィギュレーションファイル	4.2(1)	この機能が導入されました



第 10 章

tcl によるスクリプティング

この章では、Cisco NX-OS デバイス上で tcl を対話的に実行する方法とスクリプトで実行する方法について説明しており、以下の項で構成されています。

- 機能情報の確認, 145 ページ
- tcl に関する情報, 145 ページ

機能情報の確認

ご使用のソフトウェアリリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェアリリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「新機能および変更された機能に関する情報」の章を参照してください。

tcl に関する情報

tcl (Tool Command Language) は、カリフォルニア大学バークレー校の John Ousterhout が作成したスクリプト言語です。tcl 8.5 は、スクリプト作成機能を提供するために、Cisco NX-OS Release 5.1(1) に追加されました。tcl を使用すると、デバイスの CLI コマンド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。

ここでは、Cisco NX-OS デバイスで tcl スクリプトを実行する方法や、tcl を対話的に実行する方法について説明します。

注意事項と制約事項

tcl 設定時の注意事項と制限事項は次のとおりです。

tclsh コマンドのヘルプ

tcl コマンドでは、コマンドのヘルプは使用できません。インタラクティブ tcl シェル内から Cisco NX-OS コマンドのヘルプ機能に引き続きアクセスできます。

次に、インタラクティブ tcl シェルで tcl コマンドのヘルプがない場合の例を示します。

```
switch# tclsh
switch-tcl# set x 1
switch-tcl# puts ?
                ^
% Invalid command at '^' marker.
switch-tcl# configure ?
<CR>
    session  Configure the system in a session
    terminal  Configure the system from terminal input

switch-tcl#
```



(注) 上の例では、Cisco NX-OS コマンドのヘルプ機能が引き続き使用できますが、tcl の **puts** コマンドはヘルプ機能からのエラーを返します。

tclsh コマンドの履歴

端末で矢印キーを使用して、以前にインタラクティブ tcl シェルで入力したコマンドにアクセスできます。



(注) インタラクティブ tcl シェルを終了すると、**tclsh** コマンドの履歴は保存されません。

tclsh のタブ補完

インタラクティブ tcl シェルを実行している場合は、Cisco NX-OS コマンドのタブ補完を使用できます。tcl コマンドでは、タブ補完は使用できません。

tclsh の CLI コマンド

インタラクティブ tcl シェル内から直接 Cisco NX-OS コマンドにアクセスできますが、tcl cli コマンドにより付加される場合のみ tcl スクリプト内で Cisco NX-OS コマンドを実行できます。

インタラクティブ tcl シェルでは、次のコマンドは同じであり、正しく実行されます。

```
switch-tcl# cli show module 1 | incl Mod
switch-tcl# cli "show module 1 | incl Mod"
switch-tcl# show module 1 | incl Mod
```

tcl スクリプトで、次の例のように、Cisco NX-OS コマンドに `tcl cli` コマンドを付加する必要があります。

```
set x 1
cli show module $x | incl Mod
cli "show module $x | incl Mod"
```

スクリプトで次のコマンドを使用すると、そのスクリプトは失敗し、tcl シェルにエラーが表示されます。

```
show module $x | incl Mod
"show module $x | incl Mod"
```

tclsh コマンドの区切り

セミコロン (;) は、Cisco NX-OS と tcl の両方でのコマンド区切りです。tcl コマンドで複数の Cisco NX-OS コマンドを実行するには、各 Cisco NX-OS コマンドを引用符 (") で囲む必要があります。

インタラクティブ tcl シェルでは、次のコマンドは同じであり、正しく実行されます。

```
switch-tcl# cli "configure terminal ; interface loopback 10 ; description loop10"
switch-tcl# cli configure terminal ; cli interface loopback 10 ; cli description loop10
switch-tcl# cli configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
switch(config-tcl)# cli interface loopback 10
switch(config-if-tcl)# cli description loop10
switch(config-if-tcl)#
```

インタラクティブ tcl シェルでは、tcl の `cli` コマンドを付加せずに、直接 Cisco NX-OS コマンドを実行することもできます。

```
switch-tcl# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

switch(config-tcl)# interface loopback 10
switch(config-if-tcl)# description loop10
switch(config-if-tcl)#
```

tcl 変数

tcl 変数を Cisco NX-OS コマンドへの引数として使用できます。また、tcl スクリプトに引数を渡すこともできます。tcl 変数は永続的ではありません。

次の例では、tcl 変数を Cisco NX-OS コマンドへの引数として使用する方法を示します。

```
switch# tclsh
switch-tcl# set x loop10
switch-tcl# cli "configure terminal ; interface loopback 10 ; description $x"
switch(config-if-tcl)#
```

tclquit

`tclquit` コマンドは、どの Cisco NX-OS コマンドモードが現在アクティブであるかには関係なく tcl シェルを終了します。また、Ctrl+C を押して tcl シェルを終了することもできます。`exit` コマン

ドと **end** コマンドは、Cisco NX-OS コマンドモードを変更します。 **exit** コマンドは、EXEC コマンドモードからのみ tcl シェルを終了します。

tclsh のセキュリティ

tcl シェルは、Cisco NX-OS システムの特定の部分への不正アクセスを防止するために、サンドボックスで実行されます。システムは、無限ループや過剰なメモリ使用率などのイベントを検出するために、tcl シェルによって使用されている CPU、メモリ、ファイルなどのシステムリソースをモニタリングします。

初期の tcl 環境は、**scripting tcl init init-file** コマンドで設定します。

tcl 環境のループ制限は、**scripting tcl recursion-limit iterations** コマンドで定義できます。デフォルトの再帰制限は 1000 回の繰り返しです。

tclsh コマンドの実行

tclsh コマンドを使用すると、スクリプトまたはコマンドラインから tcl コマンドを実行できます。



(注) CLI プロンプトの状態では tcl スクリプトファイルを作成することはできません。スクリプトファイルはリモートデバイスで作成して、Cisco NX-OS デバイスの **bootflash:** ディレクトリにコピーすることができます。

手順

	コマンドまたはアクション	目的
ステップ 1	tclsh [bootflash:filename [argument ...]] 例： <pre>switch# tclsh ? <CR> bootflash: The file to run</pre>	tcl シェルを開始します。 引数を指定せずに tclsh コマンドを実行すると、シェルは対話形式で実行され、標準入力から tcl コマンドを読み込んで、コマンドの結果とエラーメッセージを標準出力に出力します。 tclquit を入力するか、Ctrl+C を押すとインタラクティブ tcl シェルが終了します。 引数を指定して tclsh コマンドを実行すると、最初の引数は、tcl コマンドが記述されたスクリプトファイルの名前になり、他の引数をスクリプトで変数として使用できます。

次に、インタラクティブ tcl シェルの例を示します。

```
switch# tclsh
switch-tcl# set x 1
switch-tcl# cli show module $x | incl Mod
Mod  Ports  Module-Type  Model  Status
1    32      1/10 Gbps Ethernet Module  N7K-F132XP-15  ok
Mod  Sw      Hw
```

```

Mod  MAC-Address(es)                               Serial-Num
Mod  Online Diag Status
Left ejector CLOSE, Right ejector CLOSE, Module HW does support ejector based shutdown.
switch-tcl# exit
switch#

```

次の例は、tcl スクリプトの実行方法を示しています。

```

switch# show file bootflash:showmodule.tcl
set x 1
while {$x < 19} {
cli show module $x | incl Mod
set x [expr {$x + 1}]
}

switch# tclsh bootflash:showmodule.tcl
Mod  Ports  Module-Type          Model          Status
1    32      1/10 Gbps Ethernet  N7K-F132XP-15  ok
Mod  Sw      Hw
Mod  MAC-Address(es)  Serial-Num
Mod  Online Diag Status
Left ejector CLOSE, Right ejector CLOSE, Module HW does support ejector based shutdown.
switch#

```

tclsh のコマンドによる Cisco NX-OS のモードの制御

インタラクティブ tcl シェルの実行中に Cisco NX-OS のモードを変更できます。

手順

	コマンドまたはアクション	目的
ステップ 1	tclsh 例： switch# tclsh switch-tcl#	インタラクティブ tcl シェルを開始します。
ステップ 2	configure terminal 例： switch-tcl# configure terminal switch(config-tcl)#	tcl シェルで Cisco NX-OS のコマンドを実行して、モードを変更します。 (注) tcl プロンプトが変化して、Cisco NX-OS コマンドモードになったことが示されます。
ステップ 3	tclquit 例： switch-tcl# tclquit switch#	tcl シェルを終了し、開始モードに戻ります。

次に、インタラクティブ tcl シェルから Cisco NX-OS のモードを変更する例を示します。

```

switch# tclsh
switch-tcl# configure terminal

```

```

Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-tcl)# interface loopback 10
switch(config-if-tcl)# ?
  description  Enter description of maximum 80 characters
  inherit      Inherit a port-profile
  ip           Configure IP features
  ipv6         Configure IPv6 features
  logging      Configure logging for interface
  no           Negate a command or set its defaults
  rate-limit   Set packet per second rate limit
  shutdown     Enable/disable an interface
  this         Shows info about current object (mode's instance)
  vrf          Configure VRF parameters
  end          Go to exec mode
  exit         Exit from command interpreter
  pop          Pop mode from stack or restore from name
  push         Push current mode to stack or save it under name
  where        Shows the cli context you are in

switch(config-if-tcl)# description loop10
switch(config-if-tcl)# tclquit
Exiting Tcl
switch#

```

tclの参照

次のタイトルは、参照のために示されています。

- Mark Harrison (ed)、『*Tcl/Tk Tools*』、O'Reilly Media、ISBN 1-56592-218-2、1997年
- Mark Harrison および Michael McLennan、『*Effective Tcl/Tk Programming*』、Addison-Wesley、Reading, MA, USA、ISBN 0-201-63474-0、1998年
- John K. Ousterhout、『*Tcl and the Tk Toolkit*』、Addison-Wesley、Reading, MA, USA、ISBN 0-201-63337-X、1994年
- Brent B. Welch、『*Practical Programming in Tcl and Tk*』、Prentice Hall、Upper Saddle River, NJ, USA、ISBN 0-13-038560-3、2003年
- J Adrian Zimmer、『*Tcl/Tk for Programmers*』、IEEE Computer Society、John Wiley and Sons により出版、ISBN 0-8186-8515-8、1998年



第 11 章

Python API

この章では、Cisco Nexus 7000 シリーズ デバイスに Python API を使用方法について説明します。

この章は、次の項で構成されています。

- [機能情報の確認](#), 151 ページ
- [Python API の概要](#), 151 ページ
- [Python の使用](#), 152 ページ
- [スクリプトのオンラインヘルプ](#), 157 ページ

機能情報の確認

ご使用のソフトウェア リリースで、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の警告および機能情報については、<https://tools.cisco.com/bugsearch/> の Bug Search Tool およびご使用のソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「[新機能および変更された機能に関する情報](#)」の章を参照してください。

Python API の概要

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、動的な型付け、およびインタープリタ型という性質によって、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なアプリケーション開発を実現する理想的な言語です。

Python のインタープリタと広範な標準ライブラリは、Python Web サイトから、すべての主要プラットフォームに対応した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/>

また、このサイトには、サードパーティが無償で提供している多数の Python モジュール、プログラム、およびツールのディストリビューションとそれらへのリンク、さらに追加のドキュメンテーションが掲載されています。

Cisco Nexus 7000 シリーズ デバイスは、インタラクティブ モードと非インタラクティブ（スクリプト）モードの両方で、Python v2.7.2 をサポートします。

Cisco Nexus 7000 シリーズ デバイスの Python スクリプト機能は、スイッチのコマンドライン インターフェイス（CLI）へのプログラムによるアクセスを提供し、さまざまなタスクと、Power-On Auto Provisioning（POAP）または Embedded Event Manager（EEM）アクションを実行します。



(注) Cisco NX-OS Release 6.2(8) 以降、システムは JavaScript Object Notation（JSON）およびログイン モジュールをサポートします。

Python インタープリタは Cisco NX-OS ソフトウェアでデフォルトで使用できます。

Python の使用

ここでは、Python スクリプトの作成と実行の方法について説明します。



(注) Cisco NX-OS Release 6.2(8) 以降、管理 VRF だけでなく他の VRF でも Python ネットワーク操作を実行できます。

Python ライブラリのインポート

Python で CLI コマンドを実行するには、**cisco** モジュールをインポートする必要があります。**cisco** モジュールには CLI と統合されるコードが含まれています。

CLI コマンドを有効化する、**cisco** モジュールに含まれている機能は、**python** コマンドを使用して Python のインタラクティブ モードを起動したり、**source** コマンドを使用して Python の非インタラクティブ モードを起動したりすると、自動的にインポートされます。



(注) Cisco NX-OS Release 6.2(8) 以降では、hashlib ライブラリもインポートできます。

CLI コマンドの API の使用

Cisco Nexus 7000 シリーズ デバイスでは、Python プログラミング言語は CLI コマンドを実行できる 3 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。Python インタープリタ経由で CLI コマンドを実行するには、次の API のいずれかの引数文字列として CLI コマンドを入力します。

API	説明
cli() 例： <pre>string = cli ("cli-command")</pre>	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。 (注) インタラクティブな Python インタープリタは、制御文字および特殊文字を「エスケープ」して出力します。これは、復帰改行が「\n」として出力されることを意味し、判読が難しくなる可能性があります。 clip() API は、判読性が高い結果を出力します。
clid() 例： <pre>dictionary = clid ("cli-command")</pre>	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時に使用すると便利な場合があります。
clip() 例： <pre>clip ("cli-command")</pre>	CLI コマンドの出力を直接 stdout に出力し、Python には何も返されません。 (注) <pre>clip (cli-command) と同等です r=cli("cli-command") print r</pre>

CLI からの Python インタープリタの呼び出し

次に、CLI から Python を呼び出す方法を示します。



(注) Python インタープリタのプロンプトは「>>>」または「...」で示されます。

```
switch# show clock
23:54:55.872 UTC Wed May 16 2012
switch# python
switch# >>> cli("conf term ; interface loopback 1")
switch(config-if)# >>> cli("ip address 1.1.1.1/24")
switch(config-if)# >>> cli("exit")
switch(config)# >>> cli("exit")
switch# >>> i=0
switch# >>> while i<8:
switch# ... i=i+1
switch# ... cmd = "show module %i" % i
switch# ... r=clid(cmd)
switch# ... if "TABLE modinfo/model" in r.keys():
switch# ... if r["TABLE_modinfo/model"] == "Nurburgring":
switch# ... print "got a racer in slot %d" % i
switch# ...
got a racer in slot 3
```

!-- Enter Python interpreter

!-- Exit the CLI interface mode

!-- Composite command; prompt indicates more input

!-- Empty input indicates end of loop

```
switch# >>> exit
switch#
```

! -- Exit Python interpreter

VRF の割り当ての変更

Cisco NX-OS Release 6.2(8) 以降、Python ネットワーク操作を実行するために VRF を選択できます。デフォルトでは、Python でのネットワーク操作は管理 VRF で行います。ネットワーキングおよびソケット API を使用する前に、次のコマンドを使用して、VRF を変更します。

```
switch# python
switch# >>> import socket
switch# >>> set_vrf ('default')
switch# >>> s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
VRF を他の任意の VRF に変更できます。
```

表示形式

次に、Python API を使用したさまざまな表示形式を示します。

例 1

```
switch# >>> cli("conf ; interface loopback 1")
Enter configuration commands, one per line. End with CNTL/Z
switch(config-if)# >>> clip('where detail')

mode:                conf
                    interface loopback1
username:            root
vdc:                 switch
routing-context vrf: default
```

例 2

```
switch# >>> cli("conf ; interface loopback 1")
Enter configuration commands, one per line. End with CNTL/Z
switch(config-if)# >>> cli('where detail')

'\x1b[00m mode:                conf\n
username:                    root\n
vdc:                         switch\n
routing-context vrf: default\n
                    interface loopback1\n'
```

例 3

```
switch# >>> cli("conf ; interface loopback 1")
Enter configuration commands, one per line. End with CNTL/Z
switch(config-if)# >>> r = cli('where detail') ; print r

(same output as clip() above!)
mode:                conf
                    interface loopback1
username:            root
vdc:                 switch
routing-context vrf: default
```

例 4

```
switch# >>> cli("conf ; interface loopback 1")
Enter configuration commands, one per line. End with CNTL/Z
switch(config-if)# >>> i=0
switch(config-if)# >>> while i<3:
```

```

switch(config-if)# ... i=i+1
switch(config-if)# ... cli('ip addr 1.1.1.1/24')
switch(config-if)# ...
switch(config-if)# >>> cli('end')
switch# >>> r = clid('show version')
switch# >>> for k in r.keys():
switch# ... print "%30s" % k, " = %s" % r[k]
switch# ...
        cpu_name      = Intel(R) Xeon(R) CPU
        rr_sys_ver     = 6.2(0.110)
        manufacturer   = Cisco Systems, Inc.
        isan_file_name  = bootflash:///full
        rr_ctime       = Wed May 16 02:40:57 2012
        proc_board_id   = JAF1417AGCB
        bios_cpl_time   = 02/20/10
        kickstart_ver_str = 6.1(1) [build 6.1(0.292)] [gdb]
        isan_tmstamp    = 05/16/2012 02:26:02
switch# >>> exit

```

例 5

この例では、Python に CLI コマンドの未加工出力を、制御および特殊文字を含めて表示します。たとえば、復帰改行は `\n` と表示されます。

```
switch# >>> cli('show vrf')
```

```

'VRF-Name          VRF-ID State          Reason
\ndefault          1      Up                --
\nmanagement       2      Up                --      \n'
switch# >>>

```

`clip` コマンドを入力すると、Python に「void」が返されますが、同じ出力が `std` に出力されます。

例 6

この例では、Python に対する属性名と値のディクショナリを表示します。

```

switch# >>> clid('show vrf')
{'TABLE_vrf/vrf_id': '1', 'TABLE_vrf/vrf_reason': '--', 'TABLE_vrf/vrf_name': 'default',
'TABLE_vrf/vrf_state': 'Up'}
switch# >>> cli

```

Python インタープリタの非永続性

Python は CLI シェルから分岐します。これは次のことを意味します。

- Python インタープリタの複数の呼び出しの間で状態は維持されません。
- Python インタープリタを終了すると、CLI モードは失われます。

次の例は、Python インタープリタの非永続性を示します。

```

switch# python          !-- Invoke Python interpreter
switch# >>> i = 2
switch# >>> print "var i = %d" % i
var i = 2
switch# >>> cli("configure terminal")
switch(config)# >>> blabla
switch(config)# >>> exit          !-- Exit Python interpreter
switch#                  !-- CLI still in exec mode (conf t is lost)
switch# python          !-- Invoke new Python interpreter
switch# >>> print "var i = %d" % i !-- Previous Python interpreter and variables are lost

Error: variable 'i' undefined.
switch# >>> exit
switch# conf t ; inter lo 1

```

```
switch(config-if)# python          !-- Invoke new Python interpreter
switch(config-if)# >>>          !-- Inherits the CLI mode (forked from CLI)
```

非インタラクティブ Python

次の例はスクリプトとその実行方法を示します。



(注) **bootflash:scripts** ディレクトリはデフォルトのスクリプトディレクトリです。すべてのスクリプトは **bootflash:scripts** ディレクトリまたはそのサブディレクトリに保存する必要があります。



(注) 例の先頭行の前に **#!/bin/env tclsh** を挿入し、すべての Python スクリプト ファイル名を tcl スクリプト ファイル名で置き換えることにより、Python スクリプトの代わりに tcl スクリプトを実行できます。

```
switch# show file bootflash:scripts/test1.py !-- bootflash:scripts directory
                                           !-- is the default script directory
#!/bin/env python          !-- Invoke Python to run test1.py script

i=0
while i<3:
  r=clip('show version')
  uptime_name='/@/show/version/___readonly___/kern_uptm_secs'
  print uptime_name, r[uptime_name]
  clid('sleep 1')
  i=i+1

switch# source test1.py          !-- Default directory is /bootflash/scripts
/@/show/version/___readonly___/kern_uptm_secs 36
/@/show/version/___readonly___/kern_uptm_secs 38
/@/show/version/___readonly___/kern_uptm_secs 40
switch#
```

次の例はスクリプトが引数を受け入れる方法を示します。

```
root@switch# source ntimes 2 "show clock"  !-- Pass arguments '2' and 'show clock' to script

>>>> iteration 1 for 'show clock'
21:27:21.716 UTC Tue Oct 09 2012
>>>> iteration 2 for 'show clock'
21:27:21.730 UTC Tue Oct 09 2012
root@switch#
```

バックグラウンドでのスクリプトの実行

次の例は、スクリプトをバックグラウンドで実行する方法を示します。

```
switch# source background sleep 100
switch# source background sleep 50
switch# show background
username . terminal pid start time script args ...
root . . pts/0 7687 03:31 00:00:00 sleep.py 90 .
root . . pts/0 7696 03:31 00:00:00 sleep.py 50 .
switch# kill background ?
WORD Background script to terminate, by process-id or just
a regex matching any line from 'show background' output
```

```
switch# kill background 7687
switch# show background
username . terminal pid start time script args ...
root . . pts/0 7696 03:31 00:00:00 sleep.py 50 .
switch#
switch# exit
Linux(debug)# su john
switch# kill background 7696
bash: line 1: kill: (7696) - Operation not permitted
switch#
```

スクリプトのオンラインヘルプ

オンラインヘルププロトコルにより、スクリプトでオンラインヘルプ、man ページ、状況に応じた形式を提供することができます。

次の例は、スクリプトディレクトリに含まれるスクリプトのオンラインヘルプを示します。

```
switch# source ?
abc          No help          !-- Script does not support online help protocol
cgrep       'context grep': shows matching line plus the context lines
           (to be used behind a pipe, for show-run type of output)
           Example of context for 'ip address 1.1.1.1/24': 'eth 2/1'
find-if     Find interfaces that match selected attribute values
           (from 'show intf br')
ntimes      Runs specified command specified numbers of times
redo-history This is run a list of commands from the history again
show-if     Show selected interface attributes
show-version A better 'show version'
sys/       Directory !-- Directory contains example scripts packaged in image.
           !-- These example scripts can be copied to /bootflash for
           !-- viewing and modification using the 'source copy-sys' command.
```

次の例は、ntimes スクリプトの man ページ形式のオンラインヘルプを示します。

```
switch# source ntimes ?          !-- Filename can be abridged if unique (or tabbed)
arg1: the command, in quotes
arg2: number of times to run the commands
<CR>
>      Redirect it to a file
>>    Redirect it to a file in append mode
|      Pipe command output to filter
```

オンラインヘルプのプロトコル

スクリプトは、`__cli_script.*help` という形式の引数で呼び出されます。

引数には次のように 3 つの異なる形式があります。

- `__cli_script_help`
スクリプトの目的を表す 1 行の説明を返す要求。
- `__cli_script_args_help`
引数のヘルプを返す要求。ファイル名は最初の引数として渡され、それまでに入力されたすべての引数も渡されます。
- `__cli_script_args_help_partial`

最後の引数が「部分的」である場合、引数のヘルプを返す要求。

たとえば、**interface Ethernet 1/?** は「部分的」であり、状況依存ヘルプが返されます。この例では、**interface Ethernet 1/?** により、スロット 1 のラインカードのポート範囲が返されます。

次に、引数ヘルプの表示形式を示します。

- 標準的な形式（「タイプ|説明」形式）

```
print "ftp|Use ftp for file transfer protocol"  
print "scp|Use scp for file transfer protocol"  
exit(0)
```

- man ページ形式（タブなし）

```
print "__man_page"  
print " whatever..."
```