



Cisco IOS XE Gibraltar 16.12.x (Catalyst 9200 スイッチ) システム 管理コンフィギュレーションガイド

初版：2019年7月31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。

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

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.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

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: [www.cisco.com go trademarks](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. (1721R)

© 2019–2020 Cisco Systems, Inc. All rights reserved.



目次

第 1 章

デバイスの管理 1

デバイスの管理に関する情報 1

システム日時の管理 1

システムクロック 1

ネットワークタイムプロトコル 2

NTPストラタム 3

NTPアソシエーション 4

NTPセキュリティ 4

NTPの実装 4

システム名およびシステムプロンプト 5

スタックのシステム名およびシステムプロンプト 6

デフォルトのシステム名とプロンプトの設定 6

DNS 6

DNSのデフォルト設定値 6

ログインバナー 7

バナーのデフォルト設定 7

MACアドレステーブル 7

MACアドレステーブルの作成 7

MACアドレスおよびVLAN 8

MACアドレスおよびデバイススタック 8

MACアドレステーブルのデフォルト設定 8

ARPテーブルの管理 9

デバイスの管理方法 9

手動による日付と時刻の設定 9

システムクロックの設定	9
タイムゾーンの設定	10
夏時間の設定	11
システム名の設定	13
DNSの設定	14
Message-of-the-Day ログインバナーの設定	16
ログインバナーの設定	17
MAC アドレス テーブルの管理	19
アドレス エージング タイムの変更	19
MAC アドレス変更通知トラップの設定	20
MAC アドレス移動通知トラップの設定	22
MAC しきい値通知トラップの設定	24
VLAN の MAC アドレスラーニングのディセーブル化	26
スタティック アドレス エントリの追加および削除	28
ユニキャスト MAC アドレス フィルタリングの設定	29
デバイスのモニタリングおよび保守の管理	30
デバイス管理の設定例	32
例：システムクロックの設定	32
例：サマータイムの設定	32
例：MOTD バナーの設定	32
例：ログインバナーの設定	33
例：MAC アドレス変更通知トラップの設定	33
例：MAC しきい値通知トラップの設定	33
例：MAC アドレス テーブルへのスタティック アドレスの追加	34
例：ユニキャスト MAC アドレス フィルタリングの設定	34
デバイス管理に関する追加情報	34
デバイス管理の機能履歴	34

第 2 章	ブート整合性の可視性	37
	ブート整合性の可視性について	37
	ソフトウェアイメージとハードウェアの確認	37

プラットフォーム ID とソフトウェア整合性の確認	38
ブート整合性の可視性に関する追加情報	42
ブート整合性の可視性の機能履歴	42

第 3 章

デバイスのセットアップ設定の実行	43
ソフトウェア インストールの制約事項	43
デバイスセットアップ設定の実行に関する情報	43
デバイスブートプロセス	43
ソフトウェア インストールの概要	44
ソフトウェアのブート モード	45
ソフトウェア パッケージのインストール	46
ソフトウェアインストールの終了	46
デバイス情報の割り当て	46
デフォルトのスイッチ情報	47
DHCP ベースの自動設定の概要	47
DHCP クライアントの要求プロセス	48
DHCP ベースの自動設定およびイメージアップデート	49
DHCP ベースの自動設定の制約事項	50
DHCP 自動設定	50
DHCP 自動イメージアップデート	50
DHCP サーバ設定時の注意事項	51
TFTP サーバの目的	51
DNS サーバの目的	52
コンフィギュレーションファイルの入手方法	53
環境変数の制御方法	54
一般的な環境変数	55
TFTP の環境変数	57
ソフトウェア イメージのリロードのスケジューリング	57
デバイスセットアップ設定の実行方法	58
DHCP 自動設定 (コンフィギュレーション ファイルだけ) の設定	58

DHCP 自動イメージアップデート (コンフィギュレーション ファイルおよびイメージ) の設定	60
DHCP サーバからファイルをダウンロードするクライアントの設定	63
複数の SVI への IP 情報の手動割り当て	64
デバイスのスタートアップ コンフィギュレーションの変更	66
システム コンフィギュレーションを読み書きするためのファイル名の指定	66
スイッチの手動による起動	67
インストール モードでのデバイスのブート	68
バンドルモードでのデバイスの起動	71
ソフトウェア イメージのリロードのスケジュール設定	72
デバイスセットアップ設定のモニタリング	73
例: インストール モードでのソフトウェアブートアップ ディスプレイ	73
デバイスのセットアップを実行する場合の設定例	76
例: 更新プログラム パッケージの管理	76
ソフトウェア インストールの確認	78
例: デバイスを DHCP サーバとして設定	79
例: DHCP 自動イメージアップデートの設定	80
例: DHCP サーバから設定をダウンロードするためのデバイスの設定	80
例: ソフトウェア イメージのリロードのスケジュールリング	81
デバイスセットアップの実行に関する追加情報	81
デバイスセットアップ設定の実行に関する機能履歴	81

第 4 章

スマート ライセンスの設定 83

スマートライセンシングの設定の前提条件	83
スマートライセンシングの概要	83
CSSM の概要	84
CSSM への接続	84
CSSM への接続の設定とライセンスレベルの設定	86
CSSM への接続の設定	86
ダイレクトクラウドアクセス用の Call Home サービスの設定	88

HTTPS プロキシサーバを介したダイレクトクラウドアクセス用の Call Home サービスの設定	90
Cisco Smart Software Manager オンプレミス用の Call Home サービスの設定	93
ライセンスレベルの設定	95
CSSM でのデバイスの登録	97
CSSM からの新しいトークンの生成	97
新しいトークンを使用するデバイスの登録	99
登録後のライセンスステータスの確認	99
CSSM でのデバイスの登録キャンセル	101
スマートライセンスの設定のモニタリング	102
スマートライセンシングの設定例	103
例：Call Home プロファイルの表示	103
例：登録前のライセンス情報の表示	104
例：デバイスの登録	106
例：登録後のライセンスステータスの表示	106
その他の参考資料	109
スマートライセンスの機能の履歴	110

第 5 章

有線ネットワークでの Application Visibility and Control の設定	111
有線ネットワークでの Application Visibility and Control について	111
サポートされる AVC クラス マップおよびポリシー マップのフォーマット	112
有線 Application Visibility and Control の制限	113
Application Visibility and Control の設定方法	115
有線ネットワークでの Application Visibility and Control の設定	115
インターフェイスでのアプリケーション認識の有効化	116
AVC QoS ポリシーの作成	116
スイッチ ポートへの QoS ポリシーの適用	119
有線 AVC Flexible Netflow の設定	120
NBAR2 カスタム アプリケーション	138
NBAR2 ダイナミック ヒットレス プロトコル パックのアップグレード	141
Application Visibility and Control のモニタリング	143

例：Application Visibility and Control の設定	143
基本的なトラブルシューティング：質問と回答	155
Application Visibility and Control に関する追加情報	157
有線ネットワークでの Application Visibility and Control の機能履歴	157

第 6 章

SDM テンプレートの設定 159

SDM テンプレートに関する情報	159
SDM テンプレートとスイッチ スタック	159
SDM テンプレートの設定方法	160
SDM テンプレートの設定	160
SDM テンプレートのモニタリングおよびメンテナンス	161
SDM テンプレートの設定例	161
例：SDM テンプレートの表示	161
例：SDM テンプレートの設定	163
SDM テンプレートに関する追加情報	163
SDM テンプレートの機能履歴	163

第 7 章

システム メッセージ ログの設定 165

システム メッセージ ログの設定に関する情報	165
システム メッセージ ロギング	165
システム ログ メッセージのフォーマット	166
デフォルトのシステム メッセージ ロギングの設定	167
syslog メッセージの制限	168
システム メッセージ ログの設定方法	168
メッセージ表示宛先デバイスの設定	168
ログ メッセージの同期化	170
メッセージ ロギングのディセーブル化	171
ログ メッセージのタイム スタンプのイネーブル化およびディセーブル化	172
ログ メッセージのシーケンス番号のイネーブル化およびディセーブル化	173
メッセージ重大度の定義	174
履歴テーブルおよび SNMP に送信される Syslog メッセージの制限	175

UNIX Syslog デーモンへのメッセージのロギング	176
システム メッセージ ログのモニタリングおよびメンテナンス	177
コンフィギュレーションアーカイブ ログのモニタリング	177
システム メッセージ ログの設定例	177
例：システム メッセージのスタック構成	177
例：スイッチ システム メッセージ	178
システム メッセージ ログに関する追加情報	178
システムメッセージログの機能履歴	178

第 8 章
オンライン診断の設定 179

オンライン診断の設定に関する情報	179
Generic Online Diagnostics (GOLD)	180
TestPortTxMonitoring	180
TestUnusedPortLoopback	181
オンライン診断の設定方法	181
オンライン診断テストの開始	181
オンライン診断の設定	182
オンライン診断のスケジューリング	182
ヘルス モニタリング診断の設定	184
オンライン診断のモニタリングおよびメンテナンス	187
オンライン診断テストの設定例	187
例：診断テストの開始	187
例：ヘルス モニタリング テストの設定	187
例：診断テストのスケジューリング	187
例：オンライン診断の表示	188
オンライン診断に関する追加情報	189
オンライン診断設定の機能情報	189

第 9 章
コンフィギュレーション ファイルの管理 191

コンフィギュレーション ファイルの管理の前提条件	191
コンフィギュレーション ファイルの管理の制約事項	191

コンフィギュレーションファイルの管理について	192
コンフィギュレーションファイルのタイプ	192
コンフィギュレーションモードおよびコンフィギュレーションソースの選択	192
CLIを使用したコンフィギュレーションファイルの変更	193
コンフィギュレーションファイルの場所	193
ネットワークサーバからデバイスへのコンフィギュレーションファイルのコピー	194
デバイスから TFTP サーバへのコンフィギュレーションファイルのコピー	194
デバイスから RCP サーバへのコンフィギュレーションファイルのコピー	195
デバイスから FTP サーバへのコンフィギュレーションファイルのコピー	197
VRF によるファイルのコピー	198
スイッチから別のスイッチへのコンフィギュレーションファイルのコピー	198
NVRAM より大きいコンフィギュレーションファイル	198
コンフィギュレーションファイルをダウンロードするデバイスの設定	200
コンフィギュレーションファイル情報の管理方法	200
コンフィギュレーションファイル情報の表示	200
コンフィギュレーションファイルの変更	201
デバイスから TFTP サーバへのコンフィギュレーションファイルのコピー	203
次の作業	204
デバイスから RCP サーバへのコンフィギュレーションファイルのコピー	204
例	205
次の作業	206
デバイスから FTP サーバへのコンフィギュレーションファイルのコピー	206
例	207
次の作業	208
TFTP サーバからデバイスへのコンフィギュレーションファイルのコピー	208
次の作業	209
rcp サーバからデバイスへのコンフィギュレーションファイルのコピー	209
例	210
次の作業	211
FTP サーバからデバイスへのコンフィギュレーションファイルのコピー	211
例	212

次の作業	213
NVRAM より大きいコンフィギュレーション ファイルの保守	213
コンフィギュレーション ファイルの圧縮	213
コンフィギュレーションのクラス A フラッシュ ファイル システム上のフラッシュ メモリへの格納	214
ネットワークからのコンフィギュレーション コマンドのロード	216
フラッシュ メモリからスタートアップまたは実行コンフィギュレーションへのコンフィギュレーション ファイルのコピー	217
フラッシュ メモリ ファイル システム間でのコンフィギュレーション ファイルのコピー	218
FTP サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコピー	219
次の作業	220
RCP サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコピー	221
TFTP サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコピー	222
スタートアップ コンフィギュレーション ファイルでのコンフィギュレーション コマンドの再実行	222
スタートアップ コンフィギュレーションのクリア	223
指定されたコンフィギュレーション ファイルの削除	224
クラス A フラッシュ ファイル システムでの CONFIG_FILE 環境変数の指定	225
次の作業	227
コンフィギュレーション ファイルをダウンロードするデバイスの設定	228
ネットワーク コンフィギュレーション ファイルをダウンロードするデバイスの設定	228
ホスト コンフィギュレーション ファイルをダウンロードするデバイスの設定	229
コンフィギュレーション ファイルの管理の機能履歴	231

第 10 章

セキュア コピー	233
セキュア コピーの前提条件	233
Secure Copy に関する情報	233
セキュア コピーの設定方法	234

セキュアコピーの設定	234
SSH サーバでの Secure Copy Protocol のイネーブル化	235
セキュア コピーの設定方法	237
例：ローカル認証を使用したセキュア コピーの設定	237
例：ネットワークベース認証を使用した SCP サーバ側の設定	237
セキュアコピーに関する追加情報	238
セキュア コピーの機能情報	238

第 11 章

コンフィギュレーションの置換とロールバック	241
コンフィギュレーションの置換とロールバックの前提条件	241
コンフィギュレーションの置換とロールバックの制約事項	242
コンフィギュレーションの置換とロールバックについて	242
コンフィギュレーションアーカイブ	242
コンフィギュレーションの置換	243
コンフィギュレーションロールバック	244
コンフィギュレーションロールバック変更確認	245
コンフィギュレーションの置換とロールバックの利点	245
コンフィギュレーションの置換とロールバックの使用方法	245
コンフィギュレーションアーカイブの作成	245
コンフィギュレーションの置換やロールバック操作の実行	247
機能のモニタリングおよびトラブルシューティング	250
コンフィギュレーションの置換とロールバックの設定例	253
コンフィギュレーションアーカイブの作成	253
現在の実行コンフィギュレーションを保存された Cisco IOS コンフィギュレーションファイルで置換	253
スタートアップ コンフィギュレーション ファイルへの復帰	254
configure confirm コマンドを使用したコンフィギュレーション置換操作の実行	254
コンフィギュレーションロールバック操作の実行	254
コンフィギュレーションの置換とロールバックに関するその他の参考資料	256
コンフィギュレーションの置換およびコンフィギュレーションのロールバックの機能履歴	256

第 12 章	ソフトウェア メンテナンス アップグレード 257
	ソフトウェア メンテナンス アップグレードの制約事項 257
	ソフトウェア メンテナンス アップグレードについて 257
	SMU の概要 257
	SMU のワークフロー 258
	SMU パッケージ 258
	SMU のリロード 258
	ソフトウェア メンテナンスの更新の管理方法 258
	SMU パッケージのインストール：1 ステッププロセス 259
	SMU パッケージのインストール：3 ステッププロセス 260
	SMU の管理 261
	ソフトウェア メンテナンス アップグレードの設定例 262
	ソフトウェア メンテナンス アップグレードのその他の参考資料 275
	ソフトウェア メンテナンス アップグレードの機能の履歴 275

第 13 章	フラッシュ ファイル システムの操作 277
	フラッシュ ファイル システムについて 277
	使用可能なファイル システムの表示 278
	デフォルト ファイル システムの設定 279
	ファイル システムのファイルに関する情報の表示 279
	ディレクトリの変更および作業ディレクトリの表示 281
	ディレクトリの作成 281
	ディレクトリの削除 282
	ファイルのコピー 282
	ファイルの削除 283
	ファイルの作成、表示、および抽出 284
	フラッシュ ファイル システムに関するその他の関連資料 286
	フラッシュファイルシステムの機能履歴 286

第 14 章	初期設定へのリセットの実行 289
--------	-------------------

初期設定へのリセット実行の前提条件	289
初期設定へのリセット実行の制限事項	289
初期設定へのリセットに関する情報	290
初期設定へのリセットの実行方法	291
初期設定へのリセット実行の設定例	292
初期設定へのリセットに関するその他の参考資料	292
初期設定へのリセットに関する機能履歴	292

第 15 章**セキュアストレージの設定 295**

セキュアストレージについて	295
セキュアストレージの有効化	295
セキュアストレージの無効化	296
暗号化のステータスの確認	297
セキュアストレージの機能情報	297

第 16 章**条件付きデバッグとラジオアクティブトレース 299**

条件付きデバッグの概要	299
ラジオアクティブトレースの概要	300
条件付きデバッグとラジオアクティブトレースの設定方法	300
条件付きデバッグおよび放射線トレース	300
トレースファイルの場所	300
条件付きデバッグの設定	301
L2 マルチキャストの放射線トレース	303
トレースファイルの推奨ワークフロー	303
ボックス外へのトレースファイルのコピー	303
条件付きデバッグのモニタリング	304
条件付きデバッグの設定例	305
条件付きデバッグとラジオアクティブトレースに関するその他の関連資料	305
条件付きデバッグとラジオアクティブトレースの機能履歴	306

第 17 章**同意トークン 307**

同意トークンの制約事項	307
同意トークンに関する情報	308
システムシェルアクセスの同意トークン承認プロセス	308
同意トークンの機能履歴	310

第 18 章

ソフトウェア設定のトラブルシューティング	311
ソフトウェア設定のトラブルシューティングに関する情報	311
スイッチのソフトウェア障害	311
デバイスのパスワードを紛失したか忘れた場合	312
ping	312
レイヤ 2 トレースルート	312
レイヤ 2 の traceroute のガイドライン	313
IP トレースルート	314
debug コマンド	315
システム レポート	315
スイッチのオンボード障害ロギング	317
ファン障害	318
CPU 使用率が高い場合に起こりうる症状	318
ソフトウェア設定のトラブルシューティング方法	319
パスワードを忘れた場合の回復	323
パスワード回復がイネーブルになっている場合の手順	324
パスワード回復がディセーブルになっている場合の手順	326
スイッチ スタック問題の回避	327
自動ネゴシエーションの不一致の防止	328
SFP モジュールのセキュリティと識別に関するトラブルシューティング	329
ping の実行	330
温度のモニタリング	330
物理パスのモニタリング	330
IP traceroute の実行	331
デバッグおよびエラー メッセージ出力のリダイレクト	331

show platform コマンドの使用	331
show debug コマンドの使用方法	332
ソフトウェア設定のトラブルシューティングの確認	332
OBFL 情報の表示	332
例：高い CPU 使用率に関する問題と原因の確認	332
ソフトウェア設定のトラブルシューティングのシナリオ	334
Power over Ethernet (PoE) に関するトラブルシューティングのシナリオ	334
ソフトウェアのトラブルシューティングの設定例	339
例：IP ホストの ping	339
例：IP ホストに対する traceroute の実行	340
ソフトウェア設定のトラブルシューティングに関する追加情報	341
ソフトウェア設定のトラブルシューティングの機能履歴	341



第 1 章

デバイスの管理

- デバイスの管理に関する情報 (1 ページ)
- デバイスの管理方法 (9 ページ)
- デバイス管理の設定例 (32 ページ)
- デバイス管理に関する追加情報 (34 ページ)
- デバイス管理の機能履歴 (34 ページ)

デバイスの管理に関する情報

システム日時の管理

デバイスのシステム日時は、自動設定方式 (RTC および NTP) または手動設定方式を使用して管理できます。



(注) ここで使用するコマンドの構文および使用方法の詳細については、[Cisco.com](https://www.cisco.com) で、『*Cisco IOS Configuration Fundamentals Command Reference*』を参照してください。

システムクロック

時刻サービスの基本となるのはシステムクロックです。このクロックはシステムがスタートアップした瞬間から稼働し、日時を常時トラッキングします。

システムクロックは、次のソースにより設定できます。

- RTC
- NTP
- 手動設定

システムクロックは、次のサービスに時刻を提供します。

- **user show** コマンド
- ログおよびデバッグ メッセージ

システム クロックは、グリニッジ標準時 (GMT) とも呼ばれる協定世界時 (UTC) に基づいて内部的に時刻を追跡します。ローカルのタイムゾーンおよび夏時間に関する情報を設定することにより、時刻がローカルのタイムゾーンに応じて正確に表示されるようになります。

システムクロックは、時刻に信頼性があるかどうか (つまり、信頼できると見なされるタイムソースによって時刻が設定されているか) を常時トラッキングします。信頼性のない場合は、時刻は表示目的でのみ使用され、再配信されません。

ネットワーク タイム プロトコル

NTPは、ネットワーク上のデバイス間の時刻の同期化を目的に設計されています。NTPはユーザ データグラム プロトコル (UDP) で稼働し、UDP は IP 上で稼働します。NTP は RFC 1305 で規定されています。

NTP ネットワークは通常、タイム サーバに接続されたラジオ クロックやアトミック クロックなど、正規の時刻源から時刻を取得します。NTPは、ネットワークにこの時刻を分配します。NTP はきわめて効率的で、1 分間に 1 パケットを使用するだけで、2 台のデバイスを 1 ミリ秒以内に同期化できます。

NTP では、信頼できるタイム ソースから各マシンが何 NTP ホップ隔たっているかを表すために、ストラタムという概念が使用されます。ストラタム 1 タイム サーバには、ラジオ クロックまたは原子時計が直接接続されており、ストラタム 2 タイム サーバは、NTP を使用してストラタム 1 タイム サーバから時刻を取得します (以降のストラタムも同様です)。NTP が稼働するデバイスは、タイム ソースとして、NTP を使用して通信するストラタム番号が最小のデバイスを自動的に選択します。この方法によって、NTP 時刻配信の自動編成型ツリーが効率的に構築されます。

NTPでは、同期化されていないデバイスと同期化しないことによって、時刻が正確でないデバイスとの同期化を防ぎます。また、NTP では、複数のデバイスから報告される時刻を比較して、ストラタムの番号が小さくても、時刻が他のデバイスと大幅に異なるデバイスとは同期化しません。

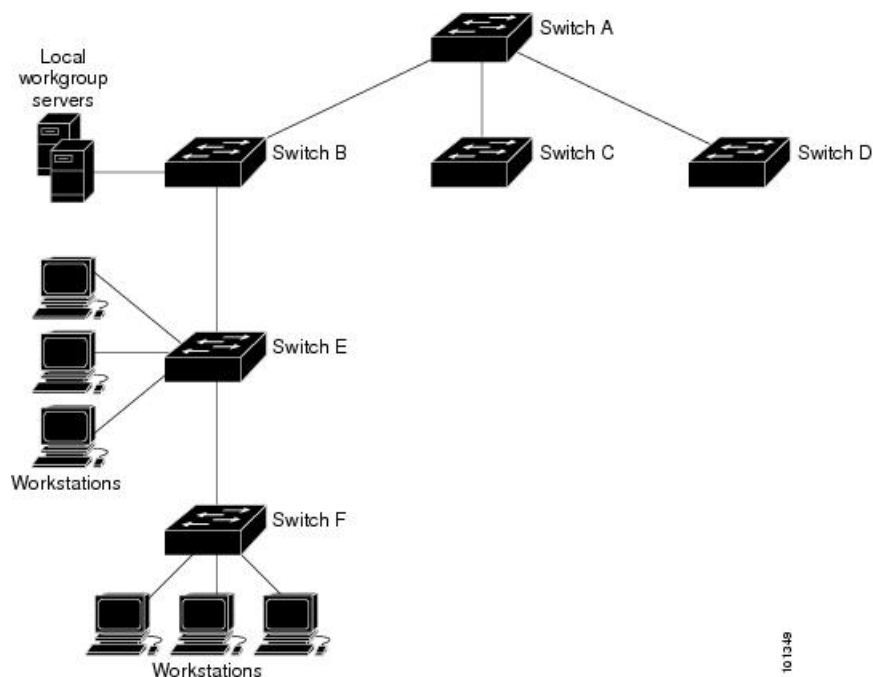
NTPが稼働するデバイス間の通信 (アソシエーション) は、通常静的に設定されます。各デバイスには、アソシエーションを作成すべきすべてのデバイスの IP アドレスが与えられます。アソシエーションのペアとなるデバイス間で NTP メッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN 環境では、代わりに IP ブロードキャストメッセージを使用するように NTP を設定できます。各デバイスを、単にブロードキャストメッセージを送受信するように設定すればよいので、この代替手段によって設定の複雑さが緩和されます。ただし、情報の流れは一方向に限られます。

デバイス上で維持される時刻は、重要なリソースです。NTP のセキュリティ機能を使用して、不正確な時刻が誤って、あるいは意図的に設定されることがないようにしてください。その方法として、アクセス リストベースの制約方式と暗号化認証方式があります。

シスコによる NTP の実装では、ストラタム 1 サービスをサポートしていないため、ラジオクロックまたは原子時計に接続できません。ネットワークの時刻サービスは、IP インターネット上のパブリック NTP サーバから取得することを推奨します。

次の図に NTP を使用した一般的なネットワークの例を示します。A はプライマリ NTP、デバイス B、C、D が NTP サーバモードに設定されている（デバイス A との間にサーバアソシエーションが設定されている）場合の NTP マスターです。デバイス E は、アップストリームデバイス（デバイス B）とダウンストリームデバイス（デバイス F）の NTP ピアとして設定されます。

図 1: 一般的な NTP ネットワークの構成



ネットワークがインターネットから切り離されている場合、シスコの NTP によって、実際には、他の方法で時刻を学習しているにもかかわらず、デバイスが NTP を使用して同期化しているように動作を設定できます。他のデバイスは、NTP によりこのデバイスと同期化されます。

複数のタイム ソースがある場合は、NTP は常に、より信頼性があると見なされます。NTP の時刻は、他の方法による時刻に優先します。

自社のホスト システムに NTP ソフトウェアを組み込んでいるメーカーが数社あり、UNIX システム用のバージョンやその派生ソフトウェアも一般に入手できます。このソフトウェアによって、ホストシステムも時間が同期化されます。

NTP ストラタム

NTP では、信頼できるタイム ソースから各マシンが何 NTP ホップ隔たっているかを表すために、ストラタムという概念が使用されます。ストラタム 1 タイム サーバには、ラジオクロックまたは原子時計が直接接続されており、ストラタム 2 タイム サーバは、NTP を使用してス

ストラタム 1 タイム サーバから時刻を取得します（以降のストラタムも同様です）。NTP が稼働するデバイスは、タイム ソースとして、NTP を使用して通信するストラタム番号が最小のデバイスを自動的に選択します。この方法によって、NTP 時刻配信の自動編成型ツリーが効率的に構築されます。

NTP では、同期化されていないデバイスと同期化しないことによって、時刻が正確でないデバイスとの同期化を防ぎます。また、NTP では、複数のデバイスから報告される時刻を比較して、ストラタムの番号が小さくても、時刻が他のデバイスと大幅に異なるデバイスとは同期化しません。

NTP アソシエーション

NTP が稼働するデバイス間の通信（アソシエーション）は、通常静的に設定されます。各デバイスには、アソシエーションを作成すべきすべてのデバイスの IP アドレスが与えられます。アソシエーションのペアとなるデバイス間で NTP メッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN 環境では、代わりに IP ブロードキャストメッセージを使用するように NTP を設定できます。各デバイスを、単にブロードキャストメッセージを送受信するように設定すればよいので、この代替手段によって設定の複雑さが緩和されます。ただし、情報の流れは一方向に限られます。

NTP セキュリティ

デバイス上で維持される時刻は、重要なリソースです。NTP のセキュリティ機能を使用して、不正確な時刻が誤って、あるいは意図的に設定されないようにしてください。その方法として、アクセス リストベースの制約方式と暗号化認証方式があります。

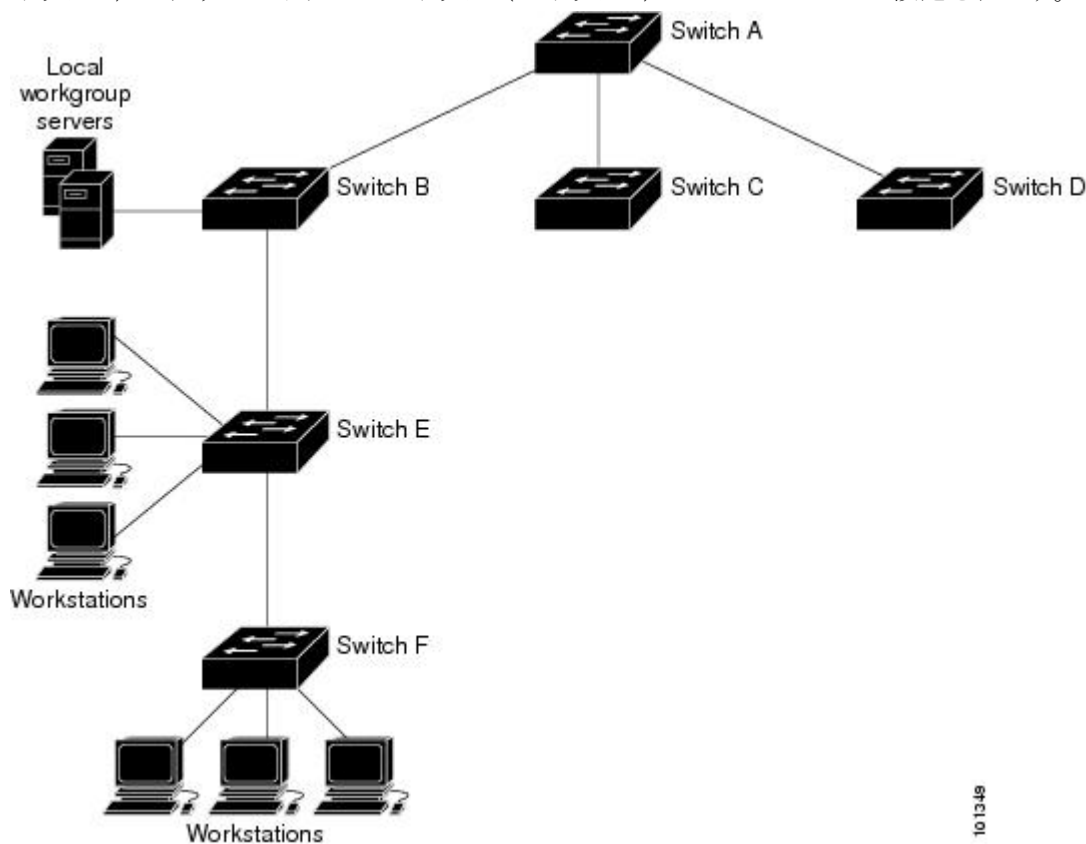
NTP の実装

NTP の実装では、ストラタム 1 サービスがサポートされないため、ラジオクロックまたは原子時計に接続できません。ネットワークの時刻サービスは、IP インターネット上のパブリック NTP サーバから取得することを推奨します。

図 2: 一般的な NTP ネットワークの構成

次の図は NTP を使用した一般的なネットワークの例を示します。スイッチ A は、スイッチ B、C、D が NTP サーバモードに設定されている（スイッチ A との間にサーバアソシエーションが設定されている）場合のプライマリ NTP です。スイッチ E は、アップストリームスイッチ（ス

スイッチ B) とダウンストリームスイッチ (スイッチ F) の NTP ピアとして設定されます。



101349

ネットワークがインターネットから切り離されている場合、NTPによって、実際には、他の方法で時刻を取得している場合でも、NTPを使用した同期化と同様にデバイスの動作を設定できます。他のデバイスは、NTPによりこのデバイスと同期化されます。

複数のタイム ソースがある場合は、NTP は常に、より信頼性があると見なされます。NTP の時刻は、他の方法による時刻に優先します。

自社のホスト システムに NTP ソフトウェアを組み込んでいるメーカーが数社あり、UNIX システム用のバージョンやその派生ソフトウェアも一般に入手できます。このソフトウェアによって、ホストシステムも時間が同期化されます。

システム名およびシステム プロンプト

デバイスを識別するシステム名を設定します。デフォルトでは、システム名およびプロンプトは *Switch* です。

システムプロンプトを設定していない場合は、システム名の最初の 20 文字がシステムプロンプトとして使用されます。大なり記号 (>) が付加されます。システム名が変更されると、プロンプトは更新されます。

ここで使用するコマンドの構文および使用方法の詳細については、『Cisco IOS Configuration Fundamentals Command Reference, Release 12.4』および『Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4』を参照してください。

スタックのシステム名およびシステム プロンプト

アクティブスイッチを介してスタックメンバにアクセスする場合は、**session stack-member-number** 特権 EXEC コマンドを使用する必要があります。スタックメンバ番号の有効範囲は 1～8 です。このコマンドを使用すると、スタックメンバの番号がシステム プロンプトの末尾に追加されます。たとえば、Switch-2#はスタックメンバ2の特権 EXEC モードのプロンプトであり、スイッチスタックのシステムプロンプトは Switch です。

デフォルトのシステム名とプロンプトの設定

デフォルトのスイッチのシステム名およびプロンプトは *Switch* です。

DNS

DNS プロトコルは、ドメインネームシステム (DNS) を制御します。DNS とは分散型データベースであり、ホスト名を IP アドレスにマッピングできます。デバイスに DNS を設定すると、**ping**、**telnet**、**connect** などのすべての IP コマンドを使用する場合や、関連する Telnet サポート操作時に、IP アドレスの代わりにホスト名を使用できます。

IP によって定義される階層型の命名方式では、デバイスを場所またはドメインで特定できます。ドメイン名は、ピリオド (.) を区切り文字として使用して構成されています。たとえば、シスコは、IP で *com* というドメイン名に分類される商業組織なので、ドメイン名は *cisco.com* となります。このドメイン内の特定のデバイス、たとえばファイル転送プロトコル (FTP) システムは、*ftp.cisco.com* で表されます。

IP ではドメイン名をトラッキングするために、ドメインネームサーバという概念が定義されています。ドメインネームサーバの役割は、名前から IP アドレスへのマッピングをキャッシュ (またはデータベース) に保存することです。ドメイン名を IP アドレスにマッピングするには、まず、ホスト名を明示し、ネットワーク上に存在するネームサーバを指定し、DNS をイネーブルにします。

DNS のデフォルト設定値

表 1: DNS のデフォルト設定値

機能	デフォルト設定
DNS イネーブル ステート	イネーブル
DNS デフォルト ドメイン名	未設定
DNS サーバ	ネームサーバのアドレスが未設定

ログインバナー

Message-of-The-Day (MoTD) バナーおよびログインバナーを作成できます。MOTD バナーはログイン時に、接続されたすべての端末に表示されます。すべてのネットワークユーザに影響するメッセージ（差し迫ったシステム シャットダウンの通知など）を送信する場合に便利です。

ログインバナーも接続されたすべての端末に表示されます。表示されるのは、MoTD バナーの後で、ログインプロンプトが表示される前です。



(注) ここで使用するコマンドの構文および使用方法の詳細については、『*Cisco IOS Configuration Fundamentals Command Reference, Release 12.4*』を参照してください。

バナーのデフォルト設定

MoTD およびログインバナーは設定されません。

MAC アドレス テーブル

MAC アドレステーブルには、デバイスがポート間のトラフィック転送に使用するアドレス情報が含まれています。このアドレス テーブルに登録されたすべての MAC アドレスは、1 つまたは複数のポートに対応しています。アドレス テーブルに含まれるアドレス タイプには、次のものがあります。

- ダイナミックアドレス：デバイスが取得し、使用されなくなった時点で期限切れとなる送信元の MAC アドレス
- スタティックアドレス：手動で入力され、期限切れにならず、デバイスのリセット時にも消去されないユニキャストアドレス

アドレス テーブルは、宛先 MAC アドレス、対応する VLAN (仮想 LAN) ID、アドレスに対応付けられたポート番号、およびタイプ (スタティックまたはダイナミック) のリストです。



(注) ここで使用するコマンドの構文および使用方法の詳細については、このリリースに対応するコマンドリファレンスを参照してください。

MAC アドレス テーブルの作成

すべてのポートでサポートされる複数の MAC アドレスを使用して、他のネットワークデバイスにデバイス上のすべてのポートを接続できます。デバイスは、各ポートで受信するパケットの送信元アドレスを取得し、アドレステーブルにアドレスとそれに関連付けられたポート番号を追加することによって、動的なアドレス指定を行います。ネットワークでデバイスの追加

たは削除が行われると、デバイスによってアドレステーブルが更新され、新しいダイナミックアドレスが追加され、使用されていないアドレスは期限切れになります。

エージング インターバルは、グローバルに設定されています。ただし、デバイスは VLAN ごとにアドレステーブルを維持し、STP によって VLAN 単位で有効期間を短縮できます。

デバイスは、受信したパケットの宛先アドレスに基づいて、任意の組み合わせのポート間でパケットを送信します。デバイスは、MAC アドレステーブルを使用することによって、宛先アドレスに関連付けられたポートに限定してパケットを転送します。宛先アドレスがパケットを送信したポート上にある場合は、パケットはフィルタリング処理され、転送されません。デバイスは、常にストアアンドフォワード方式を使用します。このため、完全なパケットをいったん保存してエラーがないか検査してから転送します。

MAC アドレスおよび VLAN

すべてのアドレスは VLAN と関連付けられます。1つのアドレスを複数の VLAN に対応付け、それぞれで異なる宛先を設定できます。たとえば、ユニキャスト アドレスを VLAN 1 のポート 1 および VLAN 5 のポート 9、10、1 に転送するといったことが可能です。

VLAN ごとに、独自の論理アドレス テーブルが維持されます。ある VLAN で認識されているアドレスが別の VLAN で認識されるには、別の VLAN 内のポートによって学習されるか、または別の VLAN 内のポートにスタティックに対応付けられる必要があります。

MAC アドレスおよびデバイススタック

すべてのスタック メンバにある MAC アドレス テーブルでは、同期が取られます。いかなる時点でも、各スタック メンバには、各 VLAN のアドレス テーブルの同じコピーがあります。アドレスがエージングアウトすると、アドレスは、すべてのスタック メンバにあるアドレス テーブルから削除されます。デバイスがスイッチスタックに参加すると、そのデバイスでは、他のスタックメンバで学習された各 VLAN のアドレスを受信します。スタックメンバがスイッチスタックに残っているときには、残りのスタックメンバは、エージングアウトするか、前のスタックメンバによってラーニングされたすべてのアドレスが削除されます。

MAC アドレス テーブルのデフォルト設定

次の表に、MAC アドレス テーブルのデフォルト設定を示します。

表 2: MAC アドレスのデフォルト設定

機能	デフォルト設定
エージング タイム	300 秒
ダイナミック アドレス	自動学習
スタティック アドレス	未設定

ARP テーブルの管理

デバイスと通信するには（イーサネット上のデバイスなど）、ソフトウェアは最初にそのデバイスの 48 ビット MAC アドレスまたはローカルデータリンクアドレスを学習する必要があります。IP アドレスからローカルデータリンクアドレスを学習するプロセスを、アドレス解決といいます。

アドレス解決プロトコル（ARP）は、ホスト IP アドレスを、該当するメディアまたは MAC アドレスおよび VLAN ID に対応付けます。IP アドレスを使用して、ARP は対応する MAC アドレスを見つけます。MAC アドレスが見つかったら、IP と MAC アドレスとの対応を ARP キャッシュに格納し、すばやく検索できるようにします。その後、IP データグラムがリンク層フレームにカプセル化され、ネットワークを通じて送信されます。イーサネット以外の IEEE 802 ネットワークにおける IP データグラムのカプセル化および ARP 要求/応答については、サブネットワークアクセスプロトコル（SNAP）で規定されています。IP インターフェイスでは、標準的なイーサネット形式の ARP カプセル化（**arpa** キーワードで表される）がデフォルトでインテグレーションに設定されています。

手動でテーブルに追加された ARP エントリは期限切れにならないので、手動で削除する必要があります。

CLI（コマンドライン インターフェイス）の手順については、*Cisco.com* で Cisco IOS Release 12.4 のマニュアルを参照してください。

デバイスの管理方法

手動による日付と時刻の設定

正確なシステム時刻は再開と再起動により保持されますが、日付と時刻はシステムが再開してから手動で設定できます。

手動設定は必要な場合にのみ使用することを推奨します。デバイスが同期できる外部ソースがある場合は、システムクロックを手動で設定する必要はありません。



（注） 手動でシステムクロックを設定している場合は、デバイスに障害が発生して別のスタックメンバがデバイスの役割を引き継ぐ前に、この設定を再設定する必要があります。

システムクロックの設定

ネットワーク上に、NTP サーバなどの時刻サービスを提供する外部ソースがある場合、手動でシステムクロックを設定する必要はありません。

システムクロックを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>次のいずれかを使用します。</p> <ul style="list-style-type: none"> clock set hh:mm:ss day month year clock set hh:mm:ss month day year <p>例 :</p> <pre>Device# clock set 13:32:00 23 March 2013</pre>	<p>次のいずれかの書式を使ってシステムクロックを手動で設定します。</p> <ul style="list-style-type: none"> hh:mm:ss : 時間 (24 時間形式)、分、秒を指定します。指定された時刻は、設定されたタイムゾーンに基づきます。 day : 月の日で日付を指定します。 month : 月を名前で指定します。 year : 年を指定します (略式表記で指定しないでください)。

タイムゾーンの設定

タイムゾーンを手動で設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>clock timezone zone hours-offset [minutes-offset]</p> <p>例 :</p>	<p>時間帯を設定します。</p> <p>内部時間は、協定世界時 (UTC) で維持されるため、このコマンドは表示専用</p>

	コマンドまたはアクション	目的
	<pre>Device(config)# clock timezone AST -3 30</pre>	<p>で、時刻を手動で設定するときだけに使用されます。</p> <ul style="list-style-type: none"> • <i>zone</i> : 標準時が適用されているときに表示されるタイムゾーンの名前を入力します。デフォルトは UTC です。 • <i>hours-offset</i> : UTC からのオフセット時間数を入力します。 • (任意) <i>minutes-offset</i> : UTC からのオフセット分数を入力します。ローカルタイムゾーンが UTC と 1 時間の差の割合である場合に指定できます。
ステップ 4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。
ステップ 5	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	入力を確認します。
ステップ 6	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	(任意) コンフィギュレーションファイルに設定を保存します。

夏時間の設定

毎年特定の日に夏時間が開始および終了する地域に夏時間を設定するには、次の作業を行います。

手順

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

	コマンドまたはアクション	目的
	Device> enable	<ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	clock summer-time zone date date month year hh:mm date month year hh:mm [offset] 例： Device(config)# clock summer-time PDT date 10 March 2013 2:00 3 November 2013 2:00	毎年指定された日に開始および終了する夏時間を設定します。
ステップ 4	clock summer-time zone recurring [week day month hh:mm week day month hh:mm [offset]] 例： Device(config)# clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00	<p>毎年指定された日に開始および終了する夏時間を設定します。すべての時刻は、現地のタイムゾーンを基準にしています。開始時間は標準時を基準にしています。</p> <p>終了時間は夏時間を基準にしています。夏時間はデフォルトでディセーブルに設定されています。パラメータなしで clock summer-time zone recurring を指定すると、夏時間のルールは米国のルールにデフォルト設定されます。</p> <p>開始月が終了月より後の場合は、システムでは南半球にいると見なされます。</p> <ul style="list-style-type: none"> • <i>zone</i> : 夏時間が有効な場合に表示される時間帯名 (PDT など) を指定します。 • (任意) <i>week</i> : 月の週 (1 ~ 4、first、または last) を指定します。 • (任意) <i>day</i> : 曜日 (Sunday、Monday など) を指定します。 • (任意) <i>month</i> : 月 (January、February など) を指定します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • (任意) <i>hh:mm</i> : 時および分単位で時間 (24時間形式) を指定します。 • (任意) <i>offset</i> : 夏時間中に追加する分数を指定します。デフォルトは 60 です。
ステップ 5	end 例 : Device(config)# end	特権 EXEC モードに戻ります。
ステップ 6	show running-config 例 : Device# show running-config	入力を確認します。
ステップ 7	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

システム名の設定

システム名を手動で設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 3	hostname name 例 : <pre>Device(config)# hostname remote-users</pre>	システム名を設定します。システム名を設定すると、システムプロンプトとしても使用されます。 デフォルト設定は Switch です。 名前は ARPANET ホスト名のルールに従う必要があります。このルールではホスト名は文字で始まり、文字または数字で終わり、その間には文字、数字、またはハイフンしか使用できません。名前には 63 文字まで使用できます。
ステップ 4	end 例 : <pre>remote-users(config)#end remote-users#</pre>	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例 : <pre>Device# show running-config</pre>	入力を確認します。
ステップ 6	copy running-config startup-config 例 : <pre>Device# copy running-config startup-config</pre>	(任意) コンフィギュレーションファイルに設定を保存します。

DNS の設定

デバイスの IP アドレスをホスト名として使用する場合、この IP アドレスが使用されるため、DNS クエリは発生しません。ピリオド (.) なしでホスト名を設定すると、ピリオドと、それに続くデフォルトのドメイン名がホスト名に追加され、その後で DNS クエリが行われ、名前を IP アドレスにマッピングします。デフォルトのドメイン名は、グローバル コンフィギュレーションモードで **ip domain name** コマンドによって設定される値です。ホスト名にピリオド (.) がある場合は、Cisco IOS ソフトウェアは、ホスト名にデフォルトのドメイン名を追加せずに IP アドレスを検索します。

DNS を使用するようにスイッチを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip domain name name 例： Device(config)# ip domain name Cisco.com	非完全修飾ホスト名（ドット付き 10 進表記ドメイン名のない名前）を完成させるためにソフトウェアが使用する、デフォルトのドメイン名を定義します。 ドメイン名を未修飾の名前から区切るために使用される最初のピリオドは入れないでください。 ブート時にはドメイン名は設定されていませんが、デバイスの設定が BOOTP または Dynamic Host Configuration Protocol (DHCP) サーバから行われている場合、BOOTP または DHCP サーバによってデフォルトのドメイン名が設定されることがあります（この情報がサーバに設定されている場合）。
ステップ 4	ip name-server server-address1 [server-address2 ... server-address6] 例： Device(config)# ip name-server 192.168.1.100 192.168.1.200 192.168.1.300	名前とアドレスの解決に使用する 1 つまたは複数のネーム サーバのアドレスを指定します。 最大 6 つのネーム サーバを指定できます。各サーバアドレスはスペースで区切ります。最初に指定されたサーバが、プライマリ サーバです。デバイスは、プライマリサーバへ DNS クエリを最初に送信します。そのクエリが失敗した場合は、バックアップサーバにクエリが送信されます。

	コマンドまたはアクション	目的
ステップ 5	ip domain lookup [nsap source-interface interface] 例 : Device(config)# ip domain-lookup	(任意) デバイス上で、DNS に基づくホスト名からアドレスへの変換をイネーブルにします。この機能は、デフォルトでイネーブルにされています。 ユーザのネットワークデバイスが、名前の割り当てを制御できないネットワーク内のデバイスと接続する必要がある場合、グローバルなインターネットのネーミング方式 (DNS) を使用して、ユーザのデバイスを一意に識別するデバイス名を動的に割り当てることができます。
ステップ 6	end 例 : Device(config)# end	特権 EXEC モードに戻ります。
ステップ 7	show running-config 例 : Device# show running-config	入力を確認します。
ステップ 8	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

Message-of-the-Day ログインバナーの設定

デバイスにログインしたときに画面に表示される 1 行以上のメッセージバナーを作成できます。

MOTD ログインバナーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 :	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	banner motd c message c 例： Device(config)# banner motd # This is a secure site. Only authorized users are allowed. For access, contact technical support. #	MoTD を指定します。 c：ポンド記号 (#) など、目的のデリミタを入力して Return キーを押します。区切り文字はバナー テキストの始まりと終わりを表します。終わりの区切り文字の後ろの文字は廃棄されます。 message：255 文字までのバナーメッセージを入力します。メッセージ内には区切り文字を使用できません。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

ログインバナーの設定

接続されたすべての端末でログインバナーが表示されるように設定できます。バナーが表示されるのは、MoTD バナーの後に、ログインプロンプトが表示される前です。

ログインバナーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>banner login c message c</p> <p>例 :</p> <pre>Device(config)# banner login \$ Access for authorized users only. Please enter your username and password. \$</pre>	<p>ログイン メッセージを指定します。</p> <p><i>c</i> : ポンド記号 (#) など、目的のデリミタを入力して Return キーを押します。区切り文字はバナー テキストの始まりと終わりを表します。終わりの区切り文字の後ろの文字は廃棄されます。</p> <p><i>message</i> : 255 文字までのログイン メッセージを入力します。メッセージ内には区切り文字を使用できません。</p>
ステップ 4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 5	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	<p>入力を確認します。</p>
ステップ 6	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	<p>(任意) コンフィギュレーション ファイルに設定を保存します。</p>

MAC アドレス テーブルの管理

アドレス エージング タイムの変更

ダイナミックアドレステーブルのエージングタイムを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mac address-table aging-time [0 10-1000000] [routed-mac vlan vlan-id] 例： Device(config)# mac address-table aging-time 500 vlan 2	ダイナミック エントリが使用または更新された後、MAC アドレス テーブル内に保持される時間を設定します。 指定できる範囲は 10 ~ 1000000 秒です。デフォルトは 300 です。0 を入力して期限切れをディセーブルにすることもできます。スタティック アドレスは、期限切れになることもテーブルから削除されることもありません。 vlan-id : 有効な ID は 1 ~ 4094 です。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例：	(任意) コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	Device# <code>copy running-config startup-config</code>	

MAC アドレス変更通知トラップの設定

NMS ホストに MAC アドレス変更通知トラップを送信するようにスイッチを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>snmp-server host <i>host-addr</i> <i>community-string</i> <i>notification-type</i> { informs traps } { version { 1 2c 3 } } { vrf <i>vrf instance name</i> }</p> <p>例 :</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>トラップメッセージの受信側を指定します。</p> <ul style="list-style-type: none"> host-addr : NMS の名前またはアドレスを指定します。 traps (デフォルト) : ホストに SNMP トラップを送信します。 informs : ホストに SNMP 情報を送信します。 version : サポートする SNMP バージョンを指定します。 informs にはバージョン 1 (デフォルト) を使用できません。 community-string : 通知処理で送信する文字列を指定します。 snmp-server host コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、snmp-server community グローバ

	コマンドまたはアクション	目的
		<p>ル コンフィギュレーション コマンドを使用してから、snmp-server host コマンドを使用することを推奨します。</p> <ul style="list-style-type: none"> • notification-type : mac-notification キーワードを使用します。 • vrf vrf インスタンス名 : このホストの VPN ルーティング/転送インスタンスを指定します。
ステップ 4	<p>snmp-server enable traps mac-notification change</p> <p>例 :</p> <pre>Device(config)# snmp-server enable traps mac-notification change</pre>	<p>デバイスが MAC アドレス変更通知を NMS に送信できるようにします。</p>
ステップ 5	<p>mac address-table notification change</p> <p>例 :</p> <pre>Device(config)# mac address-table notification change</pre>	<p>MAC アドレス変更通知機能をイネーブルにします。</p>
ステップ 6	<p>mac address-table notification change [interval value] [history-size value]</p> <p>例 :</p> <pre>Device(config)# mac address-table notification change interval 123 Device(config)# mac address-table notification change history-size 100</pre>	<p>トラップ インターバル タイムと履歴 テーブルのサイズを入力します。</p> <ul style="list-style-type: none"> • (任意) interval value : NMS に生成されるトラップの各セット間の通知トラップインターバルを秒単位で指定します。指定できる範囲は 0 ~ 2147483647 秒です。デフォルトは 1 秒です。 • (任意) history-size value : MAC 通知履歴テーブルの最大エントリ数を指定します。指定できる範囲は 0 ~ 500 です。デフォルトは 1 です。
ステップ 7	<p>interface interface-id</p> <p>例 :</p>	<p>インターフェイス コンフィギュレーション モードを開始し、SNMP MAC アドレス通知トラップをイネーブルに</p>

	コマンドまたはアクション	目的
	Device(config)# interface gigabitethernet1/0/2	するレイヤ2 インターフェイスを指定します。
ステップ 8	snmp trap mac-notification change {added removed} 例： Device(config-if)# snmp trap mac-notification change added	インターフェイス上で MAC アドレス変更通知トラップをイネーブルにします。 <ul style="list-style-type: none"> • MAC アドレスがインターフェイスにaddedされた場合にトラップをイネーブルにします。 • MAC アドレスがインターフェイスにremovedされた場合にトラップをイネーブルにします。
ステップ 9	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 10	show running-config 例： Device# show running-config	入力を確認します。
ステップ 11	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

MAC アドレス移動通知トラップの設定

MAC 移動通知を設定する場合は、MAC アドレスが、同じ VLAN 内のあるポートから別のポートに移動すると常に、SNMP 通知が生成されてネットワーク管理システムに送信されます。

次の手順に従い、デバイスを設定し、NMS ホストに MAC アドレス移動通知トラップを送信するようにします。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>snmp-server host <i>host-addr</i> {traps informs} {version {1 2c 3}} <i>community-string</i> <i>notification-type</i></p> <p>例 :</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>トラップ メッセージの受信側を指定します。</p> <ul style="list-style-type: none"> <i>host-addr</i> : NMS の名前またはアドレスを指定します。 traps (デフォルト) : ホストに SNMP トラップを送信します。 informs : ホストに SNMP 情報を送信します。 version : サポートする SNMP バージョンを指定します。informs にはバージョン1 (デフォルト) を使用できません。 community-string : 通知処理で送信する文字列を指定します。snmp-server host コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、snmp-server community グローバルコンフィギュレーション コマンドを使用してから、snmp-server host コマンドを使用することを推奨します。 notification-type : mac-notification キーワードを使用します。
ステップ 4	<p>snmp-server enable traps mac-notification move</p> <p>例 :</p>	<p>デバイスが NMS に MAC アドレス移動通知トラップを送信できるようにします。</p>

	コマンドまたはアクション	目的
	<pre>Device(config)# snmp-server enable traps mac-notification move</pre>	
ステップ 5	mac address-table notification mac-move 例 : <pre>Device(config)# mac address-table notification mac-move</pre>	MAC アドレス移動通知機能をイネーブルにします。
ステップ 6	end 例 : <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。
ステップ 7	show running-config 例 : <pre>Device# show running-config</pre>	入力を確認します。
ステップ 8	copy running-config startup-config 例 : <pre>Device# copy running-config startup-config</pre>	(任意) コンフィギュレーション ファイルに設定を保存します。

次のタスク

MAC アドレス移動通知トラップの送信をディセーブルにするには、**no snmp-server enable traps mac-notification move** グローバル コンフィギュレーション コマンドを使用します。MAC アドレス移動通知機能をディセーブルにするには、**no mac address-table notification mac-move** グローバル コンフィギュレーション コマンドを使用します。

設定を確認するには、**show mac address-table notification mac-move** 特権 EXEC コマンドを入力します。

MAC しきい値通知トラップの設定

MAC しきい値通知を設定する場合は、MAC アドレス テーブルのしきい値の制限値に達するか、その値を超えると、SNMP 通知が生成されてネットワーク管理システムに送信されます。

NMS ホストに MAC アドレス テーブルしきい値通知トラップを送信するようにスイッチを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>snmp-server host <i>host-addr</i> { traps / informs } { version { 1 2c 3 } } <i>community-string notification-type</i></p> <p>例 :</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>トラップ メッセージの受信側を指定します。</p> <ul style="list-style-type: none"> <i>host-addr</i> : NMS の名前またはアドレスを指定します。 traps (デフォルト) : ホストに SNMP トラップを送信します。 informs : ホストに SNMP 情報を送信します。 version : サポートする SNMP バージョンを指定します。informs にはバージョン 1 (デフォルト) を使用できません。 <i>community-string</i> : 通知処理で送信する文字列を指定します。snmp-server host コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、snmp-server community グローバルコンフィギュレーション コマンドを使用してから、snmp-server host コマンドを使用することを推奨します。 <i>notification-type</i> : mac-notification キーワードを使用します。
ステップ 4	<p>snmp-server enable traps mac-notification threshold</p> <p>例 :</p>	<p>NMS への MAC しきい値通知トラップをイネーブルにします。</p>

	コマンドまたはアクション	目的
	<pre>Device(config)# snmp-server enable traps mac-notification threshold</pre>	
ステップ 5	<p>mac address-table notification threshold</p> <p>例 :</p> <pre>Device(config)# mac address-table notification threshold</pre>	MAC アドレスしきい値通知機能をイネーブルにします。
ステップ 6	<p>mac address-table notification threshold [<i>limit percentage</i>] [<i>interval time</i>]</p> <p>例 :</p> <pre>Device(config)# mac address-table notification threshold interval 123 Device(config)# mac address-table notification threshold limit 78</pre>	<p>MAC アドレスしきい値使用状況モニタリングのしきい値を入力します。</p> <ul style="list-style-type: none"> • (任意) limit percentage : MAC アドレステーブルの使用率を指定します。有効値は 1 ~ 100% ですデフォルト値は 50% です。 • (任意) interval time : 通知の間隔を指定します。有効値は 120 秒以上です。デフォルトは 120 秒です。
ステップ 7	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。
ステップ 8	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	入力を確認します。
ステップ 9	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	(任意) コンフィギュレーションファイルに設定を保存します。

VLAN の MAC アドレスラーニングのディセーブル化

VLAN で MAC アドレスラーニングを制御すると、MAC アドレスを学習できる VLAN を制御することで、利用可能な MAC アドレステーブルスペースを管理できます。MAC アドレスラーニングをディセーブルにする前に、ネットワークトポロジをよく理解しておいてください。

VLAN で MAC アドレスラーニングをディセーブルにすると、ネットワークでフラッドイングを引き起こす可能性があります。

VLAN で MAC アドレスラーニングをディセーブルにするには、特権 EXEC モードで次の手順を実行します。

始める前に

VLAN の MAC アドレスラーニングをディセーブルにする際は、次の注意事項に従ってください。

- スイッチ仮想インターフェイス (SVI) スイッチを設定済みの VLAN で MAC アドレスラーニングをディセーブルにする場合は、十分注意してください。この場合、スイッチはレイヤ 2 ドメインにすべての IP パケットをフラッドイングします。
- MAC アドレスラーニングは、2 から 4094 までの 1 つの VLAN ID (例: `no mac address-table learning vlan 223`)、または、ハイフンやカンマで区切られた一連の VLAN ID (例: `no mac address-table learning vlan 1-10, 15`) でディセーブルにできます。
- MAC アドレスラーニングのディセーブル化は、ポートを 2 つ含む VLAN だけで行うことを推奨します。3 つ以上のポートを含む VLAN で MAC アドレスラーニングをディセーブルにした場合は、スイッチに着信するすべてのパケットは、その VLAN ドメインでフラッドイングします。
- セキュア ポートを含む VLAN で MAC アドレスラーニングをディセーブルにする場合、そのポートで MAC アドレスラーニングはディセーブルになりません。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	no mac-address-table learning vlan [vlan-id ,vlan-id -vlan-id,] 例 : Device(config)# no mac-address-table learning {vlan vlan-id [,vlan-id -vlan-id]}	指定された 1 つまたは複数の VLAN で MAC アドレスラーニングをディセーブルにします。 1 つの VLAN ID を指定、または VLAN ID の範囲をハイフンまたはカンマで区切って指定できます。有効な VLAN ID の範囲は 2 ~ 4094 です。内部 VLAN は指定できません。
ステップ 3	end 例 : Device(config)# end	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 4	show mac-address-table learning vlan[vlan-id] 例： Device# show mac-address-table learning [vlan vlan-id]	設定を確認します。 show mac-address-table learning [vlan vlan-id] 特権 EXEC コマンドを入力すると、すべての VLAN、または指定した VLAN の MAC アドレス ラーニングのステータスを表示できます。
ステップ 5	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。
ステップ 6	default mac address-table learning 例： Device# default mac address-table	(任意) グローバル コンフィギュレーション モードで VLAN の MAC アドレス ラーニングを再度イネーブルにします。

スタティック アドレス エントリの追加および削除

スタティック アドレスを追加するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mac address-table static mac-addr vlan vlan-id interface interface-id 例： Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1	MAC アドレス テーブルにスタティック アドレスを追加します。 <ul style="list-style-type: none"> mac-addr : アドレス テーブルに追加する宛先 MAC ユニキャスト アドレスを指定します。この宛先アドレスを持つパケットが指定した VLAN に着信すると、指定したインターフェイスに転送されます。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>vlan-id</i> : 指定された MAC アドレスを持つパケットを受信する VLAN を指定します。指定できる VLAN ID の範囲は 1 ~ 4094 です。 • <i>interface-id</i> : 受信パケットが転送されるインターフェイスを指定します。有効なインターフェイスは、物理ポートまたはポートチャネルです。スタティック マルチキャストアドレスの場合、複数のインターフェイス ID を入力できます。スタティック ユニキャストアドレスの場合、インターフェイスは同時に 1 つしか入力できません。ただし、同じ MAC アドレスおよび VLAN ID を指定すると、コマンドを複数回入力できます。
ステップ 4	show running-config 例 : Device# show running-config	入力を確認します。
ステップ 5	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

ユニキャスト MAC アドレス フィルタリングの設定

デバイスが送信元または宛先ユニキャストスタティックアドレスをドロップするよう設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mac address-table static mac-addr vlan vlan-id drop 例： Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop	ユニキャスト MAC アドレスフィルタリングをイネーブルにし、デバイスが指定した送信元または宛先ユニキャスト スタティック アドレスを持つパケットをドロップするように設定します。 <ul style="list-style-type: none"> • <i>mac-addr</i> : 送信元または宛先ユニキャスト MAC アドレス (48 ビット) を指定します。この MAC アドレスを持つパケットはドロップされます。 • <i>vlan-id</i> : 指定された MAC アドレスを持つパケットを受信する VLAN を指定します。指定できる VLAN ID の範囲は 1 ~ 4094 です。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

デバイスのモニタリングおよび保守の管理

コマンド	目的
clear mac address-table dynamic	すべてのダイナミック エントリを削除します。
clear mac address-table dynamic address mac-address	特定の MAC アドレスを削除します。

コマンド	目的
clear mac address-table dynamic interface <i>interface-id</i>	指定された物理ポートまたはポート チャンネル上のすべてのアドレスを削除します。
clear mac address-table dynamic vlan <i>vlan-id</i>	指定された VLAN 上のすべてのアドレスを削除します。
show clock [<i>detail</i>]	時刻と日付の設定を表示します。
show ip igmp snooping groups	すべての VLAN または指定された VLAN に対するレイヤ 2 マルチキャスト エントリを表示します。
show mac address-table address <i>mac-address</i>	指定された MAC アドレスの MAC アドレス テーブル情報を表示します。
show mac address-table aging-time	すべての VLAN または指定された VLAN の エージング タイムを表示します。
show mac address-table count	すべての VLAN または指定された VLAN で存在しているアドレス数を表示します。
show mac address-table dynamic	ダイナミック MAC アドレス テーブル エントリのみを表示します。
show mac address-table interface <i>interface-name</i>	指定されたインターフェイスの MAC アドレス テーブル情報を表示します。
show mac address-table move update	MAC アドレス テーブル 移動更新情報を表示します。
show mac address-table multicast	マルチキャストの MAC アドレスのリストを表示します。
show mac address-table notification {change mac-move threshold}	MAC 通知パラメータおよび履歴テーブルを表示します。
show mac address-table secure	セキュア MAC アドレスを表示します。
show mac address-table static	スタティック MAC アドレス テーブル エントリだけを表示します。
show mac address-table vlan <i>vlan-id</i>	指定された VLAN の MAC アドレス テーブル情報を表示します。

デバイス管理の設定例

例：システムクロックの設定

次の例は、システムクロックを手動で設定する方法を示しています。

```
Device# clock set 13:32:00 23 July 2013
```

例：サマータイムの設定

次に、サマータイムが3月10日の02:00に開始し、11月3日の02:00に終了する場合の設定を例として示します。

```
Device(config)# clock summer-time PDT recurring PST date  
10 March 2013 2:00 3 November 2013 2:00
```

次に、サマータイムの開始日と終了日を設定する例を示します。

```
Device(config)#clock summer-time PST date  
20 March 2013 2:00 20 November 2013 2:00
```

例：MOTD バナーの設定

次の例は、開始および終了デリミタにポンド記号（#）を使用して、MOTD バナーを設定する方法を示しています。

```
Device(config)# banner motd #  
  
This is a secure site. Only authorized users are allowed.  
For access, contact technical support.  
  
#  
Device(config)#
```

次に、前の設定により表示されたバナーの例を示します。

```
Unix> telnet 192.0.2.15  
  
Trying 192.0.2.15...  
Connected to 192.0.2.15.  
Escape character is '^]'.  
  
This is a secure site. Only authorized users are allowed.
```



```
For access, contact technical support.  
  
User Access Verification  
  
Password:
```

例：ログインバナーの設定

次の例は、開始および終了デリミタにドル記号 (\$) を使用して、にログインバナーを設定する方法を示しています。

```
Device(config)# banner login $  
  
Access for authorized users only. Please enter your username and password.  
  
$  
  
Device(config)#
```

例：MAC アドレス変更通知トラップの設定

次に、NMS として 172.20.10.10 を指定し、NMS への MAC アドレス通知トラップの送信をイネーブルにし、MAC アドレス変更通知機能をイネーブルにし、インターバル タイムを 123 秒に設定し、履歴サイズを 100 エントリに設定し、特定のポートで MAC アドレスが追加された場合のトラップをイネーブルにする例を示します。

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification  
Device(config)# snmp-server enable traps mac-notification change  
Device(config)# mac address-table notification change  
Device(config)# mac address-table notification change interval 123  
Device(config)# mac address-table notification change history-size 100  
Device(config)# interface gigabitethernet1/2/1  
Device(config-if)# snmp trap mac-notification change added
```

例：MAC しきい値通知トラップの設定

次に、NMS として 172.20.10.10 を指定し、MAC アドレスしきい値通知機能をイネーブルにし、インターバル タイムを 123 秒に設定し、制限を 78% に設定する例を示します。

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification  
Device(config)# snmp-server enable traps mac-notification threshold  
Device(config)# mac address-table notification threshold  
Device(config)# mac address-table notification threshold interval 123  
Device(config)# mac address-table notification threshold limit 78
```

例：MAC アドレス テーブルへのスタティック アドレスの追加

次の例では、MAC アドレス テーブルにスタティック アドレス c2f3.220a.12f4 を追加する方法を示します。VLAN4でこのMACアドレスを宛先アドレスとして持つパケットを受信すると、パケットは指定されたポートに転送されます。



(注) 複数のインターフェイスに同じ静的 MAC アドレスを関連付けることはできません。コマンドを別のインターフェイスで再度実行すると、新しいインターフェイス上で静的 MAC アドレスが上書きされます。

```
Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface
gigabitethernet1/1/1
```

例：ユニキャスト MAC アドレス フィルタリングの設定

次に、ユニキャスト MAC アドレス フィルタリングをイネーブルにし、c2f3.220a.12f4 の送信元または宛先アドレスを持つドロップパケットを設定する例を示します。送信元または宛先としてこの MAC アドレスを持つパケットが VLAN4 上で受信された場合、パケットがドロップされます。

```
Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

デバイス管理に関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

デバイス管理の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	デバイス管理	デバイス管理では、システムの日時、システム名、ログインバナーを設定し、DNSを設定できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 2 章

ブート整合性の可視性

- [ブート整合性の可視性について \(37 ページ\)](#)
- [ソフトウェアイメージとハードウェアの確認 \(37 ページ\)](#)
- [プラットフォーム ID とソフトウェア整合性の確認 \(38 ページ\)](#)
- [ブート整合性の可視性に関する追加情報 \(42 ページ\)](#)
- [ブート整合性の可視性の機能履歴 \(42 ページ\)](#)

ブート整合性の可視性について

ブート整合性の可視性によって、シスコのプラットフォーム ID とソフトウェアの整合性情報が可視化され、実用可能になります。プラットフォーム ID は、プラットフォームの製造元でインストールされた ID を提供します。ソフトウェアの整合性ではブート整合性の測定値が明らかになり、それを使用してプラットフォームが信頼できるコードを起動しているかどうかを評価できます。

ブートプロセス中に、ソフトウェアはブートローダーアクティビティの各ステージのチェックサムレコードを作成します。

このレコードを取得して、シスコ認定レコードと比較し、ソフトウェアイメージが正規かどうかを確認できます。チェックサム値が一致していない場合は、シスコによって認定されていない、または未承認パーティによって改ざんされているソフトウェアイメージを実行している可能性があります。

ソフトウェアイメージとハードウェアの確認

このタスクでは、スイッチの起動時に作成されたチェックサムレコードを取得する方法について説明します。特権 EXEC モードで次のコマンドを入力します。



- (注) 次のコマンドを実行した後で、メッセージ **% Please Try After Few Seconds** が CLI に表示されることがあります。これは CLI の障害を示すものではありませんが、必要な出力を取得するために必要な基盤となるインフラストラクチャの設定を示します。数分間待機して、コマンドを再度試すことをお勧めします。

メッセージ **% Error retrieving SUDI certificate** および **% Error retrieving integrity data** は、実際の CLI 障害を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	show platform sudi certificate [sign [nonce nonce]] 例 : Device# show platform sudi certificate sign nonce 123	特定の SUDI のチェックサム レコードを表示します。 <ul style="list-style-type: none"> • (オプション) sign : 署名を示します • (オプション) nonce : ナンス値を入力します
ステップ 2	show platform integrity [sign [nonce nonce]] 例 : Device# show platform integrity sign nonce 123	ブート段階のチェックサム レコードを表示します。 <ul style="list-style-type: none"> • (オプション) sign : 署名を示します • (オプション) nonce : ナンス値を入力します

プラットフォーム ID とソフトウェア整合性の確認

プラットフォーム ID の確認

次に、PEM 形式でセキュアな固有デバイス識別子 (SUDI) チェーンを表示する例を示します。SUDI にエンコードされるのは、個々のデバイスの製品 ID とシリアル番号であり、何千ものデバイスからなるネットワーク上でデバイスを一意に識別できます。最初の証明書は Cisco Root CA 2048 で、2 番目はシスコの下位 CA (ACT2 SUDI CA) です。どちらの証明書も、<https://www.cisco.com/security/pki/> で公開されているものと一致しているかを確認できます。3 番目は SUDI 証明書です。

```
Device# show platform sudi certificate sign nonce 123
-----BEGIN CERTIFICATE-----
MIIDQzCCAiugAwIBAgIQX/h7KctU3I1CoxW1aMmt/zANBgkqhkiG9w0BAQUFADA1
MRYwFAYDVQQKEw1DaXNjbjByBTEuXN0ZW1zMRswGQYDVQQDEsJDaXNjbjByBSb290IENB
IDIwNDgwHhcNMDQwNTE0MjAxNzEyWhcNMjkwNTE0MjAyNTQyWjA1MRYwFAYDVQQK
```

```
Ew1DaXNjbyBTeXN0ZW1zMRswGQYDVQDExJDaXNjbyBSb290IENBIDIwNDgwgEg
MA0GCSqGSIb3DQEBAQUAA4IBDQAwgEIAoIBAQCwrmrmp68Kd6f1cba0ZmKUEIhH
xmJVhEAYv8CrLqUccda8bnuoqrpu0hWISEWdovyD0My5jOAmAHBKeN8hf570YQXJ
FcjPFto1YmUQ6iEqDGYeJu5Tm8sUxJszR2tKyS7McQr/4NEb7Y9JhcJ6r8qqB9q
VvYgDxFUL4F1pyXOWWqCZe+36ufijXWlLvLdT6ZeYpzPEApk0E5tzivMW/VgpSdH
jWn0f84bcN5wGyDWbs2mAag8EtKpP6BrXruOIIt6ke01a06g58QBdKhTCytKmg9l
Eg6CTY5j/e/rmxrbU6YTYK/CfdfHbBcl1HP7R2RQgYUCUTOG/rksc35LgLfAgED
o1EwTzALBgNVHQ8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUJ/PI
FR5umgIJFq0roIlgX9p7L6owEAYJKwYBBAGCNxUBBAMCAQAwDQYJKoZIhvcNAQEF
BQADggEBAJ2dhISjQal8dwy3U8pORFbi71R803UXHOjgxxkLtv5M0hmEVRBW7hmW
Yqpao2TB9k5UM8Z3/sUcuVdJcr18JOagxEu5sv4dEX+5wW4q+ffY0vhN4TauYuX
cB7w4ovXsNgOnbFp1iqRe6lJT37mjpXYgyc81WhJdTsD9i7rp77rMKSSh0T8lasz
Bvt9YaretIpsjYp8qS5UwGH0GikJ3+r/+n6yUA4iGeOcaEb1fJU9u6ju7AQ7L4
CYNu/2bPPu8Xs1gYJqk0XuPL1hS27PKSb3TkL4Eq1ZKR4OCXPdJoBYVL0fdX41Id
kxpUnwVwwEpxYB5DC2Ae/qPOgRnhCzU=
```

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

```
MIIEPDCCAySgAwIBAgIKYQlufQAAAAADDANBgkqhkiG9w0BAQUFADA1MRYwFAyD
VQKKEw1DaXNjbyBTeXN0ZW1zMRswGQYDVQDExJDaXNjbyBSb290IENBIDIwNDgW
HhcNMTEwNjMwMTc1NjU3WhcNMjkwNTE0MjAyNTQyWjAnMQ4wDAYDVQKKEwVDAxNj
bzEVMBMGALUEAxMMQUNUMiBTVURJENBMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIBCGKCAQEA0m513THIXA9tN/hs5qR/6UZRpdd+9aE2JbFkNjht6gfHKd477AkS
5XAtUs5oxDYvt/zEbs1Zq3+LR6qrqKQVu6JYvH05UYLBqCj38s76NlK53905Wzp
9pRcmRCPUx+a6tHF/qRuOiJ4mdedYz03qPCpxzprWJDpC1M4iYKHuMqMqmgm+
xghHIooWS80BOcdiynEbeP5rZ7qRuewKMpl1TiI3WdBNjZjnpfjg66F+P4SaDkGb
BXDgJ130veF+EyFWLrFjj97fL2+8oauV43Qrvnf3d/GfqXj7ew+z/sX1XtEoJSXJ
URsyMEj53Rdd9tJwHky8neapszS+r+kdVQIDAQABo4IBWjCCAVYwCwYDVR0PBAQD
AgHGMB0GALUdDgQWBBRI2PHxwnDVW7t8cwmTr7i4MAP4fzAfBgNVHSMEGDAGwBQn
88gVHm6aAgkWrSugiWBF2nsqvjBDBgNVHR8EPDA6MDiqNqA0hjJodHRwOi8vd3d3
LmNpc2NvLmNvbS9zZW10eS9wa2kvY3JsL2NyY2EyMDQ4LmNybDBQBggrBgEF
BQcBAQREMEIwQAYIKwYBBQUHMAKGNh0dHA6Ly93d3cuY21zY28uY29tL3N1Y3Vy
aXR5L3BraS9jZXJ0cy9jcmNmMjA0OC5jZXIwXAYDVR0gBFUwUzBRBgorBgEAAQkV
AQwAMEMwQQYIKwYBBQUHAgEWNWh0dHA6Ly93d3cuY21zY28uY29tL3N1Y3VyXAR5
L3BraS9wb2xpY21lcY9pbmRleC5odG1sMBIGA1UdEwEB/wQIMAYBAf8CAQAwDQYJ
KoZIhvcNAQEFBQADggEBAgh1qclr9tx4hzWgDERm371yeuEmqcIffi9b9+GbMSJbi
ZHc/Cc101Ju0a9zTXA9w47H9/t6leduGxb4WeLxcwCiUgvFtCa51Ik1t8nNbcKY
/4dwlEx+7amATUQ04QggIE67wVIPu6bgAE3Ja/nRS3xKYSnj8H5TehimBSv6TEci
i5jUhOWryAK4dVo8hcJkjEkzu3ufBTJapnv89g9OE+H3VKM4L+/KdkUO+52djfKn
hyl47d7cZR4DY4LlUfM2P1As8YyjzoNpK/urSRI14WdIlplRlnH7KND15618yfvP
0IFJZBGrooCRBjOSwFv8cpWCbmWdPaCQT2nwIjTfY8c=
```

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

```
MIIDgTCCAmgAwIBAgIEAp4UYzANBgkqhkiG9w0BAQsFADANMQ4wDAYDVQKKEwVDA
xNjbyzEVMBMGALUEAxMMQUNUMiBTVURJENBMB4XDTE4MDYwNTAzNDUwNV0XDTI5
MDUxNDIwMjU0MVowbTEpMCCGALUEBRMgUELEOkM5MjAwTC0yNFQ0tNEcgU046S1BH
MjIwMjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0
REkxYjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0MjU0
DwAwgEKAoIBAQBm2Dg0GWQ18wLTKxeCt87DL8K1Rbx8Db1IigHjzebBXMpx7Ja
6Cp+kwRrIWGi5AmNmV7jZ2ZLj+vFVzBQ9eGM+6LdNg18c6nqmSnnuXMerD1UEMMK
bkFl4ydn1EIMoWpCARbgz+/zaLM2A5bpQXVndiKq1v0NA2Pgvqdxbm+8AELdDG/D
3SQ1anOja+yH5vu3NjyMjftjzk+n/ILp9iZMWzcA+O6E8K5Fc1r2cfvWlQvoFM
ZEWmHdhHPTsnN+4hhmDeurgeM0S+xIvzZqOH7PxS0kt4vYQ9xwQEawJAL44k0uY
JxKP6bdNssSLZ2s4/20BSODjyBhb0GwrOAHdAgMBAAGjzbBtMA4GALUdDwEB/wQE
AwIF4DAMBgNVHRMBAf8EAjAAME0GALUdEQRGMEsGgYJKwYBBAEJFQIDoDUTM0No
aXBjRD1RRGx6T0FZUHQwRTJRVFFQFjQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFB
PTANBgkqhkiG9w0BAQsFAAOCAQEAglUXZfNmrXZ6ZMGX69dDPkmvp9cFqXR538LF
PdypCRuSk20GF80eDU0suIi4mbB87JSOWvLomdBtXdnxzRu4kPZNFz/7pjAVRT3R
gWMMyiEnDWQsvy7e4SZmyVgej55e3hTW/LTeU8lCE0KRoYGDce5Phv2zdHtIsXrV
XsY+Erropfntt1FV9qqDskDWcKf0bos6VsyWUpSCEGqF7LfNnBTKYvXUUmKXHKf/d
W5HgrYt6bQ/h/+0EP+MY2wpAiWMCfX6F+xW20vZfK8NzNesieB38IvuTkgefzhz2s
yGCOavAxqGd0j7atcRpdRjt9+KM9Vwuy4VJZgK/tlfmTL4cawQ==
```

-----END CERTIFICATE-----

Signature version: 1

Signature:

```
-----BEGIN RSA SIGNATURE-----
```

オプションの RSA 2048 署名は、3つの証明書、署名のバージョンおよびユーザにより提供されるナンスに対するものです。

```
RSA PKCS#1v1.5 Sign {<Nonce (UINT64)> || <Signature Version (UINT32)> || <Cisco Root CA
  2048 cert (DER)> ||
<Cisco subordinate CA (DER)> || <SUDI certificate (DER)> }
```

シスコの管理ソリューションには、上記の出力を解釈する機能が装備されています。しかし、OpenSSL コマンドを使用した簡単なスクリプトを使用してプラットフォームの ID を表示して署名を確認することもでき、それによってシスコの一意のデバイス ID を確保できます。

```
[linux-host:~]openssl x509 -in sudicert.pem -subject -noout
subject= /serialNumber=PID:C9200L-24T-4G SN:FD01946BG05/O=Cisco/OU=ACT-2 Lite
SUDI/CN=C9200L-24T-4G
```

ソフトウェア整合性の確認

次に、ブート段階のチェックサムレコードを表示する例を示します。ハッシュ測定値は、連続してブートされたソフトウェアの3つの段階それぞれについて表示されます。これらのハッシュをシスコが提供する基準値と比較できます。出力に署名するオプションを使用すると、出力は正規であり改ざんされていないことを保証する機能が検証ツールに付与されます。リプレイ攻撃から保護するために、ナンスを提供できます。



- (注) ブート整合性ハッシュは MD5 ハッシュではありません。たとえば、バンドルファイルに対して **verify/md5 cat9k_iosxe.16.10.01.SPA.bin** コマンドを実行すると、ハッシュは一致しません。

次に、インストールモードでの **show platform integrity sign nonce 123** コマンドの出力例を示します。この出力には、インストールされている各パッケージファイルの測定値が含まれます。

```
Device#show platform integrity sign nonce 123
Platform: C9200L-24T-4G
Boot 0 Version: SBOOT0.v27
Boot 0 Hash:
EE98DCD0D6AEA85C8891039F649664FCC3CF709CCFC7A6F248C9D5BA8463528F
Boot Loader Version: System Bootstrap, Version 10.2, DEVELOPMENT SOFTWARE
Boot Loader Hash:
922087E7A153A79E9AE37311A1FDE2313C9996E21032F8A1E7E74935D3E742765E40DE53E7B3C50E84121C00B2D5567864FE15D30A9F67F63E1A6B
OS Version: 16.10.01
OS Hashes:
cat9k_lite-rpbase.16.10.01.SPA.pkg :
DD15C1DEFB03B0C6057AD6A9673E2114FA7CCAA7ED0A935CB0B84E0DD15C1DEFB03B0C6057AD6A9673E2114FA7CCAA7ED0A935CB0B84E0
cat9k_lite-rpboot.16.10.01.SPA.pkg :
AD6A9673E2114FA7CCAA7ED0A935CB0B84E0DD15C1DEFB03B0C6057AD6A9673E2114FA7CCAA7ED0A935CB0B84E0DD15C1DEFB03B0C6057
cat9k_lite-srdriver.16.10.01.SPA.pkg :
4FA7CCAA7ED0A935CB0B84E0DD15C1DEFB03B0C6057AD6A9673E2114FA7CCAA7ED0A935CB0B84E0DD15C1DEFB03B0C6057AD6A9673E211
cat9k_lite-webui.16.10.01.SPA.pkg :
```


ブート整合性の可視性に関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

ブート整合性の可視性の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	ブート整合性の可視性	ブート整合性の可視性によって、シスコのプラットフォームIDとソフトウェアの整合性情報が可視化され、実用可能になります。プラットフォームIDは、プラットフォームの製造元でインストールされたIDを提供します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 3 章

デバイスのセットアップ設定の実行

- [ソフトウェアインストールの制約事項 \(43 ページ\)](#)
- [デバイスセットアップ設定の実行に関する情報 \(43 ページ\)](#)
- [デバイスセットアップ設定の実行方法 \(58 ページ\)](#)
- [デバイスセットアップ設定のモニタリング \(73 ページ\)](#)
- [デバイスのセットアップを実行する場合の設定例 \(76 ページ\)](#)
- [デバイスセットアップの実行に関する追加情報 \(81 ページ\)](#)
- [デバイスセットアップ設定の実行に関する機能履歴 \(81 ページ\)](#)

ソフトウェアインストールの制約事項

- サブパッケージのインストールはサポートされていません。

デバイスセットアップ設定の実行に関する情報

IPアドレスの割り当ておよびDHCP自動設定を含む初期デバイス設定タスクを実行する前に、このモジュールのセクションを確認します。

デバイスブートプロセス

デバイスを起動するには、ハードウェア設置ガイドの手順に従ってデバイスを設置して電源投入し、デバイスの初期設定を行う必要があります。

通常の起動プロセスにはブートローダソフトウェアの動作が含まれ、以下のアクティビティが実行されます。

- 下位レベルの CPU 初期化を行います。CPU レジスタを初期化することにより、物理メモリがマッピングされる場所、容量、速度などを制御します。
- CPU サブシステムの電源投入時セルフテスト (POST) を実行し、システム DRAM をテストします。

- システム ボード上のファイル システムを初期化します。
- デフォルトのオペレーティング システム ソフトウェア イメージをメモリにロードし、デバイスを起動します。

ブート ロードにより、オペレーティング システムがロードされる前に、ファイル システムにアクセスすることができます。ブート ロードの使用目的は通常、オペレーティング システムのロード、展開、および起動に限定されます。オペレーティング システムが CPU を制御できるようになると、ブートローダは、次にシステムがリセットされるか電源が投入されるまでは非アクティブになります。

デバイス情報を割り当てるには、PC または端末をコンソールポートに接続するか、PC をイーサネット管理ポートに接続して、PC または端末エミュレーション ソフトウェアのボーレートおよびキャラクタフォーマットをデバイスのコンソールポートの設定と一致させておく必要があります。

- デフォルトのボーレートは 9600 です。
- デフォルトのデータ ビットは 8 です。



(注) データ ビット オプションを 8 に設定した場合、パリティ オプションは「なし」に設定します。

- デフォルトのストップ ビットは 2 (マイナー) です。
- デフォルトのパリティ設定は「なし」です。

ソフトウェア インストールの概要

ソフトウェア インストール機能では、イメージの完全インストール、ソフトウェア メンテナンス アップグレード (SMU)、インサービス ソフトウェア アップグレード (ISSU)、およびインサービス モデル アップグレード (データ モデル パッケージ) など、さまざまなタイプのアップグレードを同じように実行できます。

ソフトウェア インストール機能は、インストール モードでソフトウェアを 1 つのバージョンから別のバージョンへと移行する際に役立ちます。install コマンドを特権 EXEC モードで使用して、ソフトウェア イメージをインストールまたはアップグレードします。また、インストール モードを使用して以前のバージョンのソフトウェア イメージにダウングレードすることもできます。

Cisco IOS XE ソフトウェアをアップグレードするために使用する方式は、スイッチが動作しているのがインストール モードかバンドル モードかによって異なります。バンドル モードまたは統合ブートモードでは、ローカルまたはリモート ロケーションから .bin image ファイルを使用してデバイスをブートします。インストールブートモードでは、ブートローダが packages.conf ファイルを使用してデバイスをブートします。

スイッチでは、次のソフトウェア インストール機能がサポートされています。

- スタンドアロン スイッチでのソフトウェア バンドルのインストール。
- 以前にインストールしたパッケージセットへのソフトウェア ロールバック。

ソフトウェアのブートモード

デバイスでは、ソフトウェアパッケージを起動するための次の2種類のモードがサポートされています。

- インストール モード
- バンドル モード

インストールモードでのブート

以下のフラッシュ内のソフトウェアパッケージのプロビジョニングファイルを起動して、インストールモードでデバイスを起動できます。

```
Switch: boot flash:packages.conf
```



- (注) 特定リリース用の `packages.conf` ファイルが「ソフトウェア パッケージのインストール」という項で説明するインストール ワークフローで作成されています。

プロビジョニング ファイルには、起動、マウント、実行するソフトウェア パッケージのリストが含まれます。インストールされている各パッケージの ISO ファイル システムは、フラッシュからルート ファイル システムに直接マウントされます。



- (注) インストール モードで起動するために使用するパッケージとプロビジョニング ファイルは、フラッシュに保存する必要があります。usbflash0 または tftp: からインストール モードで起動することはサポートされていません。

バンドルモードでのブート

バンドル (.bin) ファイルを使用して、デバイスをバンドルモードでブートできます。

```
switch: boot flash:cat9k_lite_iosxe.16.09.02.SPA.bin
```

バンドルに含まれるプロビジョニングファイルは、どのパッケージを起動、マウント、および実行するかを判断するために使用されます。パッケージはバンドルから取得され、RAM にコピーされます。各パッケージの ISO ファイル システムは、ルート ファイル システムにマウントされます。

インストール モードでの起動とは異なり、バンドル モードでの起動では、バンドルのサイズに対応するサイズの追加メモリが使用されます。

インストール モードでの起動とは異なり、バンドル モードでの起動は複数のメディアから利用できます：

- flash:
- usbflash0:
- tftp:

ブートモードの変更

バンドルブートモードで実行中のデバイスをインストールモードに変更するには、ブート変数を flash:packages.conf に設定して **install add file flash:cat9k_2.bin activate commit** コマンドを実行します。コマンドの実行後、デバイスはインストールブートモードでリブートします。

ソフトウェアパッケージのインストール

デバイスにソフトウェアパッケージをインストールするには、**install add** コマンドを特権 EXEC モードで使用します。

install add コマンドは、ソフトウェアパッケージをローカルまたはリモートの場所からデバイスにコピーします。FTP、HTTP、HTTPs、または TFTP を使用できます。このコマンドは、.bin ファイルの個々のコンポーネントをサブパッケージと packages.conf ファイルに抽出します。またファイルを検証して、イメージファイルがプラットフォームに固有であることを確認します。

ソフトウェアインストールの終了

ソフトウェアイメージのアクティブ化は次の方法で終了できます。

- **install activate auto-abort-timer** コマンドを使用します。新しいイメージをアクティブ化した後にデバイスをリロードすると、auto-abort-timer がトリガーされます。**install commit** コマンドを発行する前にタイマーが期限切れになった場合、インストールプロセスが終了します。デバイスは再度リロードし、前のバージョンのソフトウェアイメージで起動します。

このタイマーを停止するには、**install auto-abort-timer stop** コマンドを使用します。

- **install abort** コマンドを使用します。このコマンドは、新しいソフトウェアのインストール前に実行していたバージョンにロールバックします。このコマンドは、**install commit** コマンドを発行する前に使用します。

デバイス情報の割り当て

IP 情報を割り当てるには、デバイスのセットアッププログラムを使用する方法、DHCP サーバを使用する方法、または手動で実行する方法があります。

特定の IP 情報の設定が必要な場合、デバイスのセットアッププログラムを使用してください。このプログラムを使用すると、ホスト名とイネーブル シークレット パスワードを設定することもできます。

また、任意で、Telnet パスワードを割り当てたり（リモート管理中のセキュリティ確保のため）、スイッチをクラスタのコマンドまたはメンバスイッチとして、あるいはスタンドアロンスイッチとして設定したりできます。

サーバの設定後は DHCP サーバを使用して、IP 情報の集中管理と自動割り当てを行います。



- (注) DHCP を使用している場合は、デバイスが動的に割り当てられた IP アドレスを受信してコンフィギュレーションファイルを読み込むまでは、セットアッププログラムからの質問に回答しないでください。

デバイスの設定手順を熟知している経験豊富なユーザの場合は、デバイスを手動で設定してください。それ以外のユーザは、[デバイスブートプロセス \(43 ページ\)](#) のセクションで説明したセットアッププログラムを使用してください。

デフォルトのスイッチ情報

表 3: デフォルトのスイッチ情報

機能	デフォルト設定
IP アドレスおよびサブネットマスク	IP アドレスまたはサブネット マスクは定義されていません。
デフォルト ゲートウェイ	デフォルト ゲートウェイは定義されていません。
イネーブルシークレットパスワード	パスワードは定義されていません。
ホスト名	出荷時に割り当てられるデフォルトのホスト名は device です。
Telnet パスワード	パスワードは定義されていません。
クラスタ コマンド スイッチ機能	ディセーブル
クラスタ名	クラスタ名は定義されません。

DHCP ベースの自動設定の概要

DHCP は、インターネットホストおよびインターネットワーキングデバイスに設定情報を提供します。このプロトコルには、2 つのコンポーネントがあります。1 つは DHCP サーバからデバイスにコンフィギュレーションパラメータを提供するコンポーネント、もう 1 つはデバイスにネットワーク アドレスを割り当てるコンポーネントです。DHCP はクライアント/サーバモデルに基づいています。指定された DHCP サーバが、動的に設定されるデバイスに対して、ネットワーク アドレスを割り当て、コンフィギュレーション パラメータを提供します。デバイスは、DHCP クライアントおよび DHCP サーバとして機能できます。

DHCP ベースの自動設定では、デバイス (DHCP クライアント) は起動時に、IP アドレス情報およびコンフィギュレーションファイルを使用して自動的に設定されます。

DHCP ベースの自動設定を使用すると、デバイス上で DHCP クライアント側の設定を行う必要はありません。ただし、DHCP サーバで、IP アドレスに関連した各種リース オプションを設定する必要があります。

DHCP を使用してネットワーク上のコンフィギュレーションファイルの場所をリレーする場合は、TFTP サーバおよびドメインネーム システム (DNS) サーバの設定が必要になることがあります。



- (注) スイッチ スタックと DHCP、DNS、TFTP サーバとの間では冗長接続を確立することを推奨します。接続されているスタック メンバーがスイッチ スタックから削除された場合でも、これらのサーバがアクセス可能なまま維持されるように保証するうえで役立ちます。

デバイスの DHCP サーバは、スイッチと同じ LAN 上に配置することも、そのデバイスとは別の LAN 上に配置することもできます。DHCP サーバが異なる LAN 上で動作している場合、デバイスと DHCP サーバ間に、DHCP のリレーデバイスを設定する必要があります。リレー デバイスは、直接接続されている 2 つの LAN 間でブロードキャスト トラフィックを転送します。ルータはブロードキャスト パケットを転送しませんが、受信したパケットの宛先 IP アドレスに基づいてパケットを転送します。

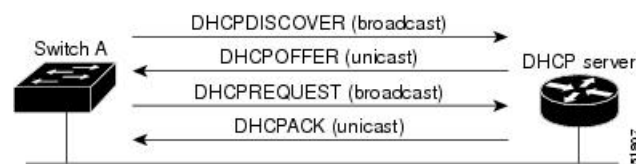
DHCP ベースの自動設定は、デバイスの BOOTP クライアント機能に代わるものです。

DHCP クライアントの要求プロセス

デバイスを起動したときに、デバイスにコンフィギュレーションファイルがない場合、DHCP クライアントが呼び出され、DHCP クライアントが DHCP サーバに設定情報を要求します。コンフィギュレーション ファイルが存在し、その設定に特定のルーテッドインターフェイスの **ip address dhcp** インターフェイス コンフィギュレーション コマンドが含まれる場合、DHCP クライアントが呼び出され、DHCP クライアントがインターフェイスに IP アドレス情報を要求します。

次は、DHCP クライアントと DHCP サーバの間で交換される一連のメッセージです。

図 3: DHCP クライアント/サーバ間のメッセージ交換



クライアントであるデバイス A は、DHCP サーバの場所を特定するために、DHCPDISCOVER メッセージをブロードキャストします。DHCP サーバは、DHCPOFFER ユニキャストメッセージによって、使用可能なコンフィギュレーション パラメータ (IP アドレス、サブネットマスク、ゲートウェイ IP アドレス、DNS IP アドレス、IP アドレス用のリースなど) をクライアントに提示します。

DHCPREQUEST ブロードキャスト メッセージでは、クライアントは、提示された設定情報に対して、DHCPサーバに正式な要求を戻します。この正式な要求はブロードキャストされるため、クライアントから DHCPDISCOVER ブロードキャスト メッセージを受信した他のすべての DHCP サーバは、クライアントに提示した IP アドレスを再利用できます。

DHCP サーバは、DHCPACK ユニキャスト メッセージをクライアントに戻すことで、IP アドレスがクライアントに割り当てられたことを確認します。このメッセージによって、クライアントとサーバはバウンドされ、クライアントはサーバから受信した設定情報を使用します。デバイスの受信する情報量は、DHCP サーバの設定方法によって異なります。

DHCPOFFER ユニキャスト メッセージによって送信されたコンフィギュレーションパラメータが無効である（コンフィギュレーションエラーがある）場合、クライアントは DHCP サーバに、DHCPDECLINE ブロードキャスト メッセージを戻します。

DHCP サーバはクライアントに、提示されたコンフィギュレーションパラメータが割り当てられていない、パラメータのネゴシエーション中にエラーが発生した、または DHCPOFFER メッセージに対するクライアントの応答が遅れている（DHCPサーバがパラメータを別のクライアントに割り当てた）という意味の DHCPNAK 拒否ブロードキャストメッセージを送信します。

DHCP クライアントは、複数の DHCP サーバまたは BOOTP サーバから提示を受け取り、そのうちの任意の1つを受け入れることができますが、通常は最初に受け取った提示を受け入れません。DHCP サーバから提示された IP アドレスが必ずしもクライアントに割り当てられるわけではありません。ただし、サーバは通常、クライアントが正式にアドレスを要求するまではアドレスを確保しておきます。デバイスが BOOTP サーバからの応答を受け入れ、自身を設定する場合、デバイスはデバイスコンフィギュレーションファイルを取得するために、TFTP 要求をユニキャストするのではなくブロードキャストします。

DHCP ホスト名オプションにより、デバイスのグループはホスト名および標準コンフィギュレーションを集中管理型 DHCP サーバから取得できます。クライアント（デバイス）は DHCPDISCOVER メッセージ内に、DHCP サーバからのホスト名および他のコンフィギュレーションパラメータの要求に使用される Option 12 フィールドを加えます。すべてのクライアントのコンフィギュレーションファイルは、DHCP から取得したホスト名を除き、まったく同じです。

DHCP ベースの自動設定およびイメージアップデート

DHCP イメージアップグレード機能を使用すると、ネットワーク内の1つ以上のデバイスに新しいイメージファイルおよび新しいコンフィギュレーションファイルをダウンロードするように DHCP サーバを設定できます。ネットワーク内のすべてのスイッチでのイメージおよびコンフィギュレーションの同時アップグレードによって、ネットワークに加えられたそれぞれの新しいデバイスが、同じイメージとコンフィギュレーションを確実に受信するようになります。

DHCP イメージアップグレードには、自動設定およびイメージアップデートの2つのタイプがあります。

DHCP ベースの自動設定の制約事項

- ネットワーク内に割り当てられた IP アドレスがなく、1つ以上のレイヤ3 インターフェイスが起動していない場合は、設定プロセスが保存された DHCP ベースの自動設定は停止します。
- タイムアウトを設定しない限り、設定機能を備えている DHCP ベースの自動設定は IP アドレスのダウンロードを無期限に繰り返します。
- コンフィギュレーションファイルをダウンロードできないか破損している場合は、自動インストールプロセスが停止します。
- TFTP からダウンロードされたコンフィギュレーション ファイルは、実行コンフィギュレーション内の既存コンフィギュレーションとマージされますが、**write memory** または **copy running-configuration startup-configuration** 特権 EXEC コマンドを入力しない限り、NVRAM に保存されません。ダウンロードされたコンフィギュレーションがスタートアップコンフィギュレーションに保存された場合、後続のシステム再起動中にこの機能はトリガーされません。

DHCP 自動設定

DHCP 自動設定は、コンフィギュレーション ファイルを DHCP サーバからネットワーク内の 1つ以上のデバイスにダウンロードします。ダウンロードされたコンフィギュレーションファイルは、デバイスの実行コンフィギュレーションファイルになります。このファイルは、デバイスがリロードされるまで、フラッシュメモリに保存されたブートアップコンフィギュレーションを上書きしません。

DHCP 自動イメージアップデート

DHCP 自動設定とともに DHCP 自動イメージアップグレードを使用すると、コンフィギュレーションおよび新しいイメージをネットワーク内の 1つ以上のデバイスにダウンロードできます。新しいコンフィギュレーションおよび新しいイメージをダウンロードしている 1つまたは複数のデバイスは、ブランク（つまり、出荷時のデフォルト設定がロードされている状態）にできます。

コンフィギュレーションをすでに持っているスイッチに新しいコンフィギュレーションをダウンロードすると、ダウンロードされたコンフィギュレーションは、スイッチに保存されているコンフィギュレーションファイルに追加されます（どの既存のコンフィギュレーションファイルも、ダウンロードされたファイルに上書きされません）。

デバイスの DHCP 自動イメージアップデートをイネーブルにするには、イメージファイルおよびコンフィギュレーションファイルがある TFTP サーバを、正しいオプション 67（コンフィギュレーションファイル名）、オプション 66（DHCP サーバホスト名）、オプション 150（TFTP サーバアドレス）、およびオプション 125（Cisco IOS イメージファイルの説明）の設定で設定する必要があります。

デバイスをネットワークに設置すると、自動イメージアップデート機能が開始します。ダウンロードされたコンフィギュレーションファイルはデバイスの実行コンフィギュレーションに保存され、新しいイメージがダウンロードされてデバイスにインストールされます。デバイスを

再起動すると、このコンフィギュレーションがデバイスのコンフィギュレーションに保存されます。

DHCP サーバ設定時の注意事項

デバイスを DHCP サーバとして設定する場合、次の注意事項に従ってください。

- DHCPサーバには、デバイスのハードウェアアドレスによって各デバイスと結び付けられている予約済みのリースを設定する必要があります。
- デバイスに IP アドレス情報を受信させるには、DHCP サーバに次のリースオプションを設定する必要があります。
 - クライアントの IP アドレス（必須）
 - クライアントのサブネットマスク（必須）
 - DNS サーバの IP アドレス（任意）
 - ルータの IP アドレス（デバイスで使用するデフォルトゲートウェイアドレス）（必須）
- デバイスに TFTPサーバからコンフィギュレーションファイルを受信させる場合は、DHCPサーバに次のリースオプションを設定する必要があります。
 - TFTP サーバ名（必須）
 - ブートファイル名（クライアントが必要とするコンフィギュレーションファイル名）（推奨）
 - ホスト名（任意）
- DHCPサーバの設定によっては、デバイスはIPアドレス情報またはコンフィギュレーションファイル、あるいはその両方を受信できます。
- 前述のリースオプションを設定しなかった場合、DHCPサーバは、設定されたパラメータのみを使用してクライアントの要求に応答します。IPアドレスおよびサブネットマスクが応答に含まれていないと、デバイスは設定されません。ルータのIPアドレスまたはTFTPサーバ名が見つからなかった場合、デバイスはTFTP要求をユニキャストしないでブロードキャストする場合があります。その他のリースオプションは、使用できなくても自動設定には影響しません。
- デバイスは DHCP サーバとして動作することができます。デフォルトでは、Cisco IOS DHCPサーバおよびDHCPリレーエージェント機能はデバイス上でイネーブルにされていますが、設定されていません。（これらの機能は動作しません）

TFTP サーバの目的

DHCP サーバの設定に基づいて、デバイスは TFTP サーバから 1 つまたは複数のコンフィギュレーションファイルをダウンロードしようとします。TFTP サーバへの IP 接続に必要なすべて

のオプションについてデバイスに応答するよう DHCP を設定している場合で、なおかつ、TFTP サーバ名、アドレス、およびコンフィギュレーション ファイル名を指定して DHCP サーバを設定している場合、デバイスは指定された TFTP サーバから指定されたコンフィギュレーション ファイルをダウンロードしようとします。

コンフィギュレーション ファイル名、および TFTP サーバを指定しなかった場合、またはコンフィギュレーション ファイルをダウンロードできなかった場合は、デバイスはファイル名と TFTP サーバアドレスをさまざまに組み合わせてコンフィギュレーション ファイルをダウンロードしようとします。ファイルには、特定のコンフィギュレーション ファイル名（存在する場合）と次のファイルが指定されています。network-config、cisco.net.cfg、hostname.config、または hostname.cfg です。この場合、hostname はデバイスの現在のホスト名です。使用される TFTP サーバアドレスには、（存在する場合）指定された TFTP サーバのアドレス、およびブロードキャストアドレス（255.255.255.255）が含まれています。

デバイスが正常にコンフィギュレーション ファイルをダウンロードするには、TFTP サーバのベースディレクトリに1つまたは複数のコンフィギュレーション ファイルが含まれていなければなりません。含めることのできるファイルは、次のとおりです。

- DHCP 応答で指定されているコンフィギュレーション ファイル（実際のデバイスコンフィギュレーション ファイル）。
- network-config または cisco.net.cfg ファイル（デフォルトのコンフィギュレーション ファイル）
- router-config または cisco.rtr.cfg ファイル（これらのファイルには、すべてのデバイスに共通のコマンドが含まれています。通常、DHCP および TFTP サーバが適切に設定されていれば、これらのファイルはアクセスされません）

DHCP サーバ リース データベースに TFTP サーバ名を指定する場合は、DNS サーバのデータベースに TFTP サーバ名と IP アドレスのマッピングを設定することも必要です。

使用する TFTP サーバが、デバイスとは異なる LAN 上にある場合、またはデバイスがブロードキャストアドレスを使用してアクセスした場合（前述のすべての必須情報が DHCP サーバの応答に含まれていない場合に発生）は、リレーを設定して TFTP サーバに TFTP パケットを転送する必要があります。適切な解決方法は、必要なすべての情報を使用して DHCP サーバを設定することです。

DNS サーバの目的

DHCP サーバは、DNS サーバを使用して TFTP サーバ名を IP アドレスに変換します。DNS サーバ上で、TFTP サーバ名から IP アドレスへのマッピングを設定する必要があります。TFTP サーバには、デバイスのコンフィギュレーション ファイルが存在します。

DHCP の応答時に IP アドレスを取得する DHCP サーバのリース データベースに、DNS サーバの IP アドレスを設定できます。リース データベースには、DNS サーバの IP アドレスを2つまで入力できます。

DNS サーバは、デバイスと同じ LAN 上に配置することも、別の LAN 上に配置することもできます。DNS サーバが別の LAN 上に存在する場合、デバイスはルータを介して DNS サーバにアクセスできなければなりません。

コンフィギュレーション ファイルの入手方法

IP アドレスおよびコンフィギュレーション ファイル名が DHCP で専用のリースとして取得できるかどうかに応じて、デバイスは次の方法で設定情報を入手します。

- IP アドレスおよびコンフィギュレーション ファイル名が、デバイス用に予約され、DHCP 応答（1 ファイル読み込み方式）で提供されている場合

デバイスは DHCP サーバから、IP アドレス、サブネットマスク、TFTP サーバアドレス、およびコンフィギュレーション ファイル名を受信します。デバイスは、TFTP サーバにユニキャストメッセージを送信し、指定されたコンフィギュレーション ファイルをサーバのベースディレクトリから取得して、ブートアッププロセスを完了します。

- デバイスの IP アドレスおよびコンフィギュレーション ファイル名が予約されているが、DHCP 応答に TFTP サーバアドレスが含まれていない場合（1 ファイル読み込み方式）。

デバイスは DHCP サーバから、IP アドレス、サブネットマスク、およびコンフィギュレーション ファイル名を受信します。デバイスは、TFTP サーバにブロードキャストメッセージを送信し、指定されたコンフィギュレーション ファイルをサーバのベースディレクトリから取得して、ブートアッププロセスを完了します。

- IP アドレスだけがデバイス用に予約され、DHCP 応答で提供されており、コンフィギュレーション ファイル名は提供されない場合（2 ファイル読み込み方式）

デバイスは DHCP サーバから、IP アドレス、サブネットマスク、および TFTP サーバアドレスを受信します。デバイスは、TFTP サーバにユニキャストメッセージを送信し、`network-config` または `cisconet.cfg` のデフォルト コンフィギュレーション ファイルを取得します（`network-config` ファイルが読み込めない場合、デバイスは `cisconet.cfg` ファイルを読み込みます）。

デフォルト コンフィギュレーション ファイルには、デバイスのホスト名から IP アドレスへのマッピングが含まれています。デバイスは、ファイルの情報をホストテーブルに書き込み、ホスト名を入手します。ファイルにホスト名がない場合、デバイスは DHCP 応答で指定されたホスト名を使用します。DHCP 応答でホスト名が指定されていない場合、デバイスはデフォルトの *Switch* をホスト名として使用します。

デフォルトのコンフィギュレーション ファイルまたは DHCP 応答からホスト名を入手した後、デバイスはホスト名と同じ名前のコンフィギュレーション ファイル（`network-config` または `cisconet.cfg` のどちらが先に読み込まれたかに応じて、`hostname-config` または `hostname.cf`）を TFTP サーバから読み込みます。`cisconet.cfg` ファイルが読み込まれている場合は、ホストのファイル名は 8 文字に切り捨てられます。

`network-config`、`cisconet.cfg`、またはホスト名と同じ名前のファイルを読み込むことができない場合、デバイスは `router-config` ファイルを読み込みます。`router-config` ファイルを読み込むことができない場合、デバイスは `ciscortr.cfg` ファイルを読み込みます。



- (注) DHCP 応答から TFTP サーバを入手できなかった場合、ユニキャスト伝送によるコンフィギュレーションファイルの読み込みにすべて失敗した場合、または TFTP サーバ名を IP アドレスに変換できない場合には、デバイスは TFTP サーバ要求をブロードキャストします。

環境変数の制御方法

通常動作デバイスでは、9600 bps に設定されているコンソール接続のみを通じてブートローダモードを開始します。電源コードを再接続中にデバイス電源コードを取り外し、[Mode] ボタンを押します。ブートローダのデバイスプロンプトが表示されます。

デバイスのブートローダソフトウェアは不揮発性の環境変数をサポートするため、これらの環境変数を使用して、ブートローダまたはシステムで稼働する他のソフトウェアの動作を制御できます。ブートローダの環境変数は、UNIX または DOS システムで設定できる環境変数と類似しています。

値を持つ環境変数は、フラッシュ ファイル システムの外にあるフラッシュ メモリに保存されます。

ファイルの各行には、環境変数名と等号に続いて、その変数の値が指定されます。変数が存在しない場合は、変数の値はありません。値がヌルストリングと表示された場合は、変数に値が設定されています。ヌルストリング（たとえば ""）が設定されている変数は、値が設定された変数です。多くの環境変数は事前に定義されており、デフォルト値が設定されています。

環境変数の設定を変更するには、ブートローダにアクセスするか、Cisco IOS コマンドを使用します。通常的环境では、環境変数の設定を変更する必要はありません。

一般的な環境変数

この表では、最も一般的な環境変数の機能について説明します。

表 4: 一般的な環境変数

変数	ブートローダ コマンド	Cisco IOS グローバルコンフィギュレーション コマンド
BOOT	<p>set BOOT <i>filesystem</i> :/ <i>file-url</i> ...</p> <p>自動起動時にロードして実行を試みる、セミコロンで区切られた実行可能ファイルのリスト。</p>	<p>boot system {<i>filesystem</i> : /<i>file-url</i> ... <i>switch</i> {<i>number</i> all}}</p> <p>次の起動時にロードする Cisco IOS イメージ、および、イメージがロードされるスタックメンバを指定します。このコマンドは、BOOT 環境変数の設定を変更します。</p> <p>パッケージプロビジョニングファイルは、<i>packages.conf</i> ファイルとも呼ばれ、起動時にどのソフトウェアパッケージをアクティブ化するかを判断するために、システムが使用するものです。</p> <ul style="list-style-type: none"> インストールモードで起動する場合、アクティブ化するパッケージを指定するために、boot コマンドで指定されたパッケージプロビジョニングファイルが使用されます。たとえば、boot flash:packages.conf です。 バンドルモードで起動する場合、起動したバンドルに含まれているパッケージのプロビジョニングファイルがバンドルに含まれているパッケージのアクティブ化に使用されます。たとえば、boot flash:image.bin のようになります。

変数	ブートローダ コマンド	Cisco IOS グローバルコンフィギュレーション コマンド
MANUAL_BOOT	<p>set MANUAL_BOOT yes</p> <p>スイッチの起動を自動で行うか手動で行うかを決定します。</p> <p>有効な値は 1、yes、0、および no です。no または 0 に設定されている場合、ブートローダはシステムを自動的に起動しようとします。それ以外の値に設定されている場合は、ブートローダ モードから手動でスイッチを起動する必要があります。</p>	<p>boot manual</p> <p>次の起動時にスイッチを手動で起動できるようにします。</p> <p>MANUAL_BOOT 環境変数の設定が変更されます。</p> <p>次のシステム再起動時には、スイッチはブートローダモードになります。システムを起動するには、boot flash: filesystem :/ file-url ブートローダコマンドを使用してブート可能なイメージの名前を指定します。</p>
CONFIG_FILE	<p>set CONFIG_FILE flash:/ file-url</p> <p>Cisco IOS がシステム コンフィギュレーションの不揮発性コピーの読み書きに使用するファイル名を変更します。</p>	<p>boot config-file flash:/ file-url</p> <p>Cisco IOS がシステム設定の不揮発性コピーの読み書きに使用するファイル名を指定します。このコマンドによって、CONFIG_FILE 環境変数が変更されます。</p>
SWITCH_NUMBER	<p>set SWITCH_NUMBER stack-member-number</p> <p>スタック メンバのメンバ番号を変更します。</p>	<p>switch current-stack-member-number renumber new-stack-member-number</p> <p>スタック メンバのメンバ番号を変更します。</p>
SWITCH_PRIORITY	<p>set SWITCH_PRIORITY stack-member-number</p> <p>スタック メンバのプライオリティ値を変更します。</p>	<p>switch stack-member-number priority priority-number</p> <p>スタック メンバのプライオリティ値を変更します。</p>
BAUD	<p>set BAUD baud-rate</p>	<p>line console 0</p> <p>speedspeed-value</p> <p>ボー レートを設定します。</p>
ENABLE_BREAK	<p>set ENABLE_BREAK yes/no</p>	<p>boot enable-break switch yes/no</p> <p>自動起動時の break をイネーブルにします。break コマンドの入力に与えられた時間は 5 秒です。</p>

TFTP の環境変数

イーサネット管理ポートを通してスイッチに PC を接続していると、TFTP でブートローダに対してコンフィギュレーションファイルのアップロードまたはダウンロードができます。このテーブルの環境変数が設定されていることを確認します。

表 5: TFTP の環境変数

変数	説明
MAC_ADDR	<p>スイッチの MAC アドレスを指定します。</p> <p>(注) 変数は変更しないことを推奨します。</p> <p>ただし、ブートローダを稼働した後に変数を変更した場合、またはこの変数が保存されている値と異なる場合は、TFTP を使用する前にこのコマンドを入力します。新しい値を有効にするためにリセットする必要があります。</p>
IP_ADDRESS	<p>スイッチの関連付けられた IP サブネットに IP アドレスおよびサブネットマスクを指定します。</p>
DEFAULT_GATEWAY	<p>デフォルトゲートウェイに IP アドレスおよびサブネットマスクを指定します。</p>

ソフトウェアイメージのリロードのスケジューリング

デバイス上でソフトウェアイメージのリロードを後で（深夜、週末などデバイスをあまり使用しないときに）行うように、スケジュールを設定できます。または（ネットワーク内のすべてのデバイスでソフトウェアをアップグレードする場合など）ネットワーク全体でリロードを同時に行うことができます。



(注) リロードのスケジュールは、約 24 日以内に設定する必要があります。

リロードオプションには以下のものがあります。

- 指定した分数、または時間および分数が経過したときに、ソフトウェアがリロードされます。リロードは、約 24 時間以内に実行する必要があります。最大 255 文字で、リロードの理由を指定できます。
- ソフトウェアのリロードが（24時間制で）指定された時間に有効になります。月日を指定すると、指定された日時にリロードが行われるようにスケジュールが設定されます。月日を指定しなかった場合、リロードは当日の指定時刻に行われます（指定時刻が現時刻より後の場合）。または翌日の指定時刻に行われます（指定時刻が現在時刻よりも前の場合）。00:00 を指定すると、深夜 0 時のリロードが設定されます。

reload コマンドはシステムを停止させます。手動で起動することが設定されていない限り、システムは自動的に再起動します。

手動で起動するようにデバイスが設定されている場合、仮想端末からリロードを実行しないでください。これはデバイスがブートローダモードになることでリモートユーザが制御を失う事態を防止するための制約です。

コンフィギュレーションファイルを変更すると、リロードの前にコンフィギュレーションを保存するように指示するプロンプトがデバイスにより表示されます。保存操作時に、**CONFIG_FILE** 環境変数がすでに存在しないスタートアップ コンフィギュレーション ファイルを示していた場合、保存を続行するかどうかという問い合わせがシステムから出されます。その状況のまま続けると、リロード時にセットアップ モードが開始されます。

スケジュールがすでに設定されたリロードを取り消すには、**reload cancel** 特権 EXEC コマンドを使用します。

デバイスセットアップ設定の実行方法

DHCP を使用してデバイスに新しいイメージおよび新しいコンフィギュレーションをダウンロードするには、少なくとも 2 つのデバイスを設定する必要があります。1 つ目のデバイスは DHCP サーバおよび TFTP サーバと同じように機能し、2 つ目のデバイス（クライアント）は新しいコンフィギュレーション ファイル、または新しいコンフィギュレーション ファイルおよび新しいイメージファイルをダウンロードするように設定されています。

DHCP 自動設定（コンフィギュレーション ファイルだけ）の設定

このタスクでは、新しいデバイスの自動設定をサポートできるように、ネットワーク内の既存のデバイスで TFTP や DHCP の設定の DHCP 自動設定を行う方法を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	ip dhcp pool poolname 例： Device(config)# ip dhcp pool pool	DHCP サーバアドレスプールの名前を作成し、DHCP プール コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	<p>boot filename</p> <p>例 :</p> <pre>Device (dhcp-config) # boot config-boot.text</pre>	ブートイメージとして使用されるコンフィギュレーションファイルの名前を指定します。
ステップ 4	<p>network network-number mask prefix-length</p> <p>例 :</p> <pre>Device (dhcp-config) # network 10.10.10.0 255.255.255.0</pre>	<p>DHCP アドレス プールのサブネット ネットワーク番号およびマスクを指定します。</p> <p>(注) プレフィックス長は、アドレスプレフィックスを構成するビット数を指定します。プレフィックスは、クライアントのネットワークマスクを指定する二者択一の方法です。プレフィックス長は、スラッシュ (/) で開始する必要があります。</p>
ステップ 5	<p>default-router address</p> <p>例 :</p> <pre>Device (dhcp-config) # default-router 10.10.10.1</pre>	DHCP クライアントのデフォルト ルータの IPアドレスを指定します。
ステップ 6	<p>option 150 address</p> <p>例 :</p> <pre>Device (dhcp-config) # option 150 10.10.10.1</pre>	TFTP サーバの IP アドレスを指定します。
ステップ 7	<p>exit</p> <p>例 :</p> <pre>Device (dhcp-config) # exit</pre>	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	<p>tftp-server flash:filename.text</p> <p>例 :</p> <pre>Device (config) # tftp-server flash:config-boot.text</pre>	TFTP サーバ上のコンフィギュレーション ファイルを指定します。

	コマンドまたはアクション	目的
ステップ 9	interface <i>interface-id</i> 例： Device(config-if)# no switchport	コンフィギュレーションファイルを受信するクライアントのアドレスを指定します。
ステップ 10	no switchport 例： Device(config-if)# no switchport	インターフェイスをレイヤ 3 モードにします。
ステップ 11	ip address <i>address mask</i> 例： Device(config-if)# ip address 10.10.10.1 255.255.255.0	IP アドレスとインターフェイスのマスクを指定します。
ステップ 12	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。

DHCP 自動イメージアップデート（コンフィギュレーションファイルおよびイメージ）の設定

このタスクでは、新しいスイッチのインストールをサポートするように既存のデバイスで TFTP および DHCP を設定する DHCP 自動設定について説明します。

始める前に

最初にデバイスにアップロードするテキストファイル（たとえば、`autoinstall_dhcp`）を作成します。このテキストファイル内に、ダウンロードするイメージの名前を含めます（たとえば、`cat9k_iosxe.16.xx.xx.SPA.bin`）。

手順

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

	コマンドまたはアクション	目的
ステップ 2	<p>ip dhcp pool <i>poolname</i></p> <p>例 :</p> <pre>Device(config)# ip dhcp pool pool1</pre>	DHCP サーバアドレス プールの名前を作成し、DHCP プール 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>boot <i>filename</i></p> <p>例 :</p> <pre>Device(dhcp-config)# boot config-boot.text</pre>	ブートイメージとして使用されるファイルの名前を指定します。
ステップ 4	<p>network <i>network-number mask prefix-length</i></p> <p>例 :</p> <pre>Device(dhcp-config)# network 10.10.10.0 255.255.255.0</pre>	<p>DHCP アドレス プールのサブネット ネットワーク番号およびマスクを指定します。</p> <p>(注) プレフィックス長は、アドレスプレフィックスを構成するビット数を指定します。プレフィックスは、クライアントのネットワークマスクを指定する二者択一の方法です。プレフィックス長は、スラッシュ (/) で開始する必要があります。</p>
ステップ 5	<p>default-router <i>address</i></p> <p>例 :</p> <pre>Device(dhcp-config)# default-router 10.10.10.1</pre>	DHCP クライアントのデフォルトルータの IP アドレスを指定します。
ステップ 6	<p>option 150 <i>address</i></p> <p>例 :</p> <pre>Device(dhcp-config)# option 150 10.10.10.1</pre>	TFTP サーバの IP アドレスを指定します。
ステップ 7	<p>option 125 <i>hex</i></p> <p>例 :</p> <pre>Device(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370</pre>	イメージファイルのパスを記述したテキストファイルのパスを指定します。

	コマンドまたはアクション	目的
ステップ 8	copy tftp flash filename.txt 例 : Device(config)# copy tftp flash image.bin	デバイスに、テキストファイルをアップロードします。
ステップ 9	copy tftp flash imagename.bin 例 : Device(config)# copy tftp flash image.bin	デバイスに、新しいイメージの tar ファイルをアップロードします。
ステップ 10	exit 例 : Device(dhcp-config)# exit	グローバル コンフィギュレーションモードに戻ります。
ステップ 11	tftp-server flash: config.txt 例 : Device(config)# tftp-server flash:config-boot.txt	TFTP サーバ上の Cisco IOS コンフィギュレーション ファイルを指定します。
ステップ 12	tftp-server flash: imagename.bin 例 : Device(config)# tftp-server flash:image.bin	TFTP サーバ上のイメージ名を指定します。
ステップ 13	tftp-server flash: filename.txt 例 : Device(config)# tftp-server flash:boot-config.txt	ダウンロードするイメージファイルの名前を記述したテキストファイルを指定します。
ステップ 14	interface interface-id 例 : Device(config)# interface gigabitEthernet1/0/4	コンフィギュレーションファイルを受信するクライアントのアドレスを指定します。

	コマンドまたはアクション	目的
ステップ 15	no switchport 例 : Device (config-if) # no switchport	インターフェイスをレイヤ 3 モードにします。
ステップ 16	ip address address mask 例 : Device (config-if) # ip address 10.10.10.1 255.255.255.0	IP アドレスとインターフェイスのマスクを指定します。
ステップ 17	end 例 : Device (config-if) # end	特権 EXEC モードに戻ります。
ステップ 18	copy running-config startup-config 例 : Device (config-if) # end	(任意) コンフィギュレーションファイルに設定を保存します。

DHCP サーバからファイルをダウンロードするクライアントの設定



(注) レイヤ3インターフェイスだけを設定してイネーブルにする必要があります。保存されているコンフィギュレーションのDHCPベースの自動設定にIPアドレスを割り当てないでください。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 2	boot host dhcp 例 :	保存されているコンフィギュレーションで自動設定をイネーブルにします。

	コマンドまたはアクション	目的
	Device (conf) # boot host dhcp	
ステップ 3	boot host retry timeout <i>timeout-value</i> 例 : Device (conf) # boot host retry timeout 300	(任意) システムがコンフィギュレーション ファイルをダウンロードしようとする時間を設定します。 (注) タイムアウトを設定しないと、システムは無期限に DHCP サーバから IP アドレスを取得しようとします。
ステップ 4	banner config-save ^C <i>warning-message</i> ^C 例 : Device (conf) # banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot ^C	(任意) コンフィギュレーション ファイルを NVRAM に保存しようとするときに表示される警告メッセージを作成します。
ステップ 5	end 例 : Device (config-if) # end	特権 EXEC モードに戻ります。
ステップ 6	show boot 例 : Device # show boot	設定を確認します。

複数の SVI への IP 情報の手動割り当て

このタスクでは、複数のスイッチ仮想インターフェイス (SVI) に IP 情報を手動で割り当てる方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 :	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
	Device# configure terminal	
ステップ 2	interface vlan <i>vlan-id</i> 例 : Device(config)# interface vlan 99	インターフェイスコンフィギュレーションモードを開始して、IP 情報が割り当てられている VLAN を指定します。指定できる範囲は 1 ~ 4094 です。
ステップ 3	ip address <i>ip-address subnet-mask</i> 例 : Device(config-vlan)# ip address 10.10.10.2 255.255.255.0	IP アドレスとサブネットマスクを入力します。
ステップ 4	exit 例 : Device(config-vlan)# exit	グローバル コンフィギュレーションモードに戻ります。
ステップ 5	ip default-gateway <i>ip-address</i> 例 : Device(config)# ip default-gateway 10.10.10.1	<p>デバイスに直接接続しているネクストホップのルータインターフェイスの IP アドレスを入力します。このスイッチにはデフォルトゲートウェイが設定されています。デフォルトゲートウェイは、デバイススイッチから宛先 IP アドレスを取得していない IP パケットを受信します。</p> <p>デフォルトゲートウェイが設定されると、デバイスは、ホストが接続する必要のあるリモートネットワークに接続できます。</p> <p>(注) IP でルーティングするようにデバイスを設定した場合、デフォルトゲートウェイの設定は不要です。</p> <p>(注) デフォルトゲートウェイの構成に基づいて、デバイスの CAPWAP は中継を行い、ルーティングされたアクセスポイントとデバイスの接続をサポートします。</p>

	コマンドまたはアクション	目的
ステップ 6	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 7	show interfaces vlan <i>vlan-id</i> 例： Device# show interfaces vlan 99	設定された IP アドレスを確認します。
ステップ 8	show ip redirects 例： Device# show ip redirects	設定されたデフォルト ゲートウェイを確認します。

デバイスのスタートアップコンフィギュレーションの変更

システムコンフィギュレーションを読み書きするためのファイル名の指定

Cisco IOS ソフトウェアは、デフォルトで `config.text` ファイルを使用して、システムコンフィギュレーションの不揮発性コピーを読み書きします。別のファイル名を指定することもできます。次回の起動時には、その名前のファイルが読み込まれます。

始める前に

このタスクではスタンドアロンのデバイスを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Switch# configure terminal	グローバルコンフィギュレーションモードを開始します。
ステップ 2	boot flash:<i>file-url</i> 例： Switch(config)# boot flash:config.text	次回の起動時に読み込むコンフィギュレーションファイルを指定します。 <i>file-url</i> : パス (ディレクトリ) およびコンフィギュレーションファイル名。

	コマンドまたはアクション	目的
		ファイル名およびディレクトリ名は、大文字と小文字を区別します。
ステップ 3	end 例： Switch(config)# end	特権 EXEC モードに戻ります。
ステップ 4	show boot 例： Switch# show boot	入力を確認します。 boot グローバル コンフィギュレーション コマンドによって、CONFIG_FILE 環境変数の設定が変更されます。
ステップ 5	copy running-config startup-config 例： Switch# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

スイッチの手動による起動

スイッチはデフォルトで自動的に起動しますが、手動で起動するように設定することもできます。

始める前に

このタスクのスタンドアロン スイッチを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	boot manual 例： Device(config)# boot manual	次回の起動時に、スイッチを手動で起動できるようにします。

	コマンドまたはアクション	目的
ステップ 3	end 例 : Device(config)# end	特権 EXEC モードに戻ります。
ステップ 4	show boot 例 : Device# show boot	入力を確認します。 boot manual グローバルコマンドは、 MANUAL_BOOT 環境変数の設定を変更します。 次回、システムを再起動した際には、スイッチはブートローダ モードになり、ブートローダ モードであることが <i>switch:</i> プロンプトによって示されます。システムを起動するには、 boot filesystem:/file-url ブートローダコマンドを使用します。 <ul style="list-style-type: none"> • filesystem : システム ボードのフラッシュ デバイスに flash: を使用します。 Switch: boot flash: • file-url : パス (ディレクトリ) および起動可能なイメージの名前を指定します。 ファイル名およびディレクトリ名は、大文字と小文字を区別します。
ステップ 5	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

インストール モードでのデバイスのブート

ソフトウェア パッケージのインストール

単一のコマンドまたは個別のコマンドを使用してソフトウェア パッケージをインストールして、アクティブ化し、コミットできます。このタスクでは、ソフトウェアパッケージをインストールするための **install add file activate commit** コマンドの使用方法を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>install add file tftp: filename [activate commit]</p> <p>例 :</p> <pre>Device# install add file flash:cat9k_lite_iosxe.16.09.01.SPA.bin activate commit</pre>	<p>ソフトウェア インストール パッケージをリモート ロケーションから (FTP、HTTP、HTTPS、TFTPを介して) デバイスにコピーし、プラットフォームおよびイメージバージョンの互換性チェックを実行し、ソフトウェア パッケージをアクティブ化し、そのパッケージを複数回リロードしても維持されるようにします。</p> <ul style="list-style-type: none"> このコマンドは、.bin ファイルの個別のコンポーネントをサブパッケージと packages.conf ファイルに抽出します。 このコマンドの実行後にデバイスはリロードします。
ステップ 3	<p>exit</p> <p>例 :</p> <pre>Device# exit</pre>	<p>特権 EXEC モードを終了し、ユーザ EXEC モードに戻ります。</p>

更新プログラムパッケージの管理

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>install add file tftp: filename</p> <p>例 :</p> <pre>Device# install add file tftp://172.16.0.1/tftpbboot/folder1/ cat9k_iosxe.16.06.01.SPA.bin</pre>	<p>リモート ロケーションから (FTP、HTTP、HTTPS、TFTPを介して) デバイスにソフトウェア インストール パッケージをコピーし、プラットフォームとイメージのバージョンの互換性チェックを実行します。</p>

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> このコマンドは、.bin ファイルの個別のコンポーネントをサブパッケージと packages.conf ファイルに抽出します。
ステップ 3	install activate [auto-abort-timer] 例： Device# install activate	追加のソフトウェア インストールパッケージをアクティブ化し、デバイスをリロードします。 <ul style="list-style-type: none"> ソフトウェアの完全インストールを実行する場合は、パッケージ ファイル名を指定しないでください。 auto-abort-timer キーワードがソフトウェアイメージのアクティブ化を自動的にロールバックします。 新しいイメージがアクティブになった後で自動タイマーがトリガーされます。 install commit コマンドを発行する前にタイマーの期限が切れた場合、インストールプロセスは自動的に終了します。デバイスがリロードし、以前のバージョンのソフトウェア イメージで起動します。
ステップ 4	install abort 例： Device# install abort	(任意) ソフトウェアインストールのアクティブ化を終了し、現在のインストール手順の前に実行していたバージョンにロールバックします。 <ul style="list-style-type: none"> このコマンドは、イメージがアクティブ化されている状態でのみ使用できます。イメージがコミットされた状態の場合は使用できません。
ステップ 5	install commit 例： Device# install commit	リロードが繰り返されても持続する変更を行います。 <ul style="list-style-type: none"> install commit コマンドで、新しいイメージのインストールを完了します。自動アボート タイマーが期限切れになるまで、複数回のリロード後も変更は維持されます。

	コマンドまたはアクション	目的
ステップ 6	install rollback to committed 例： Device# install rollback to committed	(任意) 最後にコミットしたバージョンに更新をロールバックします。
ステップ 7	install remove {file filesystem: filename inactive} 例： Device# install remove inactive	(任意) 未使用および非アクティブ状態のソフトウェア インストール ファイルを削除します。
ステップ 8	show install summary 例： Device# show install summary	アクティブ パッケージに関する情報を表示します。 • このコマンドの出力は、設定されている install コマンドに応じて変化します。

バンドルモードでのデバイスの起動

デバイスを起動するには、いくつかの方法があります。1つは、TFTP サーバから **bin** ファイルをコピーしてデバイスを起動する方法です。または、**boot flash:<image.bin>** コマンドか、**boot usbflash0:<image.bin>** コマンドを使用して、デバイスをフラッシュまたは USB フラッシュから直接起動することもできます。

以下の手順は、バンドルモードで TFTP サーバからデバイスを起動する方法を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch:BOOT=<source path of .bin file> 例： switch:BOOT=tftp://10.0.0.2/cat9k_lite_iosv.16.09.02.SPA.bin	ブート パラメータを設定します。
ステップ 2	boot 例： switch: boot	デバイスを起動します。
ステップ 3	show version	デバイスがバンドルモードであることを確認します。

ソフトウェアイメージのリロードのスケジュール設定

このタスクでは、ソフトウェアイメージを後でリロードするようにデバイスを設定する方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	copy running-config startup-config 例： Device# copy running-config startup-config	reload コマンドを使用する前に、デバイスの設定情報をスタートアップコンフィギュレーションに保存します。
ステップ 3	reload in [hh:]mm [text] 例： Device# reload in 12 System configuration has been modified. Save? [yes/no]: y	指定した分数、または時間および分数が経過したときに、ソフトウェアがリロードされるようにスケジュールを設定します。リロードは、約 24 日以内に実行する必要があります。最大 255 文字で、リロードの理由を指定できます。
ステップ 4	reload at hh: mm [month day day month] [text] 例： Device(config)# reload at 14:00	リロードを実行する時間を、時間数と分数で指定します。 (注) at キーワードを使用するのは、デバイスのシステムクロックが (Network Time Protocol (NTP)、ハードウェアカレンダー、または手動で) 設定されている場合だけです。時刻は、デバイスに設定されたタイムゾーンに基づきます。リロードが複数のデバイスで同時に行われるようにスケジュールリングするには、各デバイスの時間が NTP と同期している必要があります。

	コマンドまたはアクション	目的
ステップ 5	reload cancel 例 : Device(config)# reload cancel	以前にスケジューリングされたりロードをキャンセルします。
ステップ 6	show reload 例 : show reload	以前デバイスにスケジューリングされたりロードに関する情報、またはリロードがスケジューリングされているかを表示します。

デバイスセットアップ設定のモニタリング

例: インストールモードでのソフトウェアブートアップディスプレイ

この例では、インストールモードでのソフトウェアブートアップの表示を示します。

```
switch: boot flash:packages.conf
Attempting to boot from [flash:packages.conf]
Located packages.conf
#

validate_package: SHA-1 hash:
    expected 340D5091:2872A0DD:03E9068C:3FDBECAB:69786462
    calculated 340D5091:2872A0DD:03E9068C:3FDBECAB:69786462
Image parsed from conf file is cat9k-rpboot.16.09.01.SPA.pkg
#####

Waiting for 120 seconds for other switches to boot
#####
Switch number is 1

Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706

Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.9.1,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Tue 30-May-17 00:36 by mcpre
```

例: インストールモードでのソフトウェアブートアップディスプレイ

```
Cisco IOS-XE software, Copyright (c) 2005-2017 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.
```

```
FIPS: Flash Key Check : Begin
FIPS: Flash Key Check : End, Not Found, FIPS Mode Not Enabled
```

```
This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
```

```
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html
```

```
If you require further assistance please contact us by sending email to
export@cisco.com.
cisco C9200L-24P-4G (ARM64) processor with 518473K/3071K bytes of memory.
Processor board ID JPG221000RH
988 Virtual Ethernet interfaces
56 Gigabit Ethernet interfaces
2048K bytes of non-volatile configuration memory.
2015456K bytes of physical memory.
819200K bytes of Crash Files at crashinfo:.
1941504K bytes of Flash at flash:.
0K bytes of WebUI ODM Files at webui:.
819200K bytes of Crash Files at crashinfo-7:.
1941504K bytes of Flash at flash-7:.
```

```
Base Ethernet MAC Address      : 68:2c:7b:f7:49:00
Motherboard Assembly Number    : 73-18699-2
Motherboard Serial Number      : JAE22090AZB
Model Revision Number          : 13
Motherboard Revision Number    : 05
Model Number                   : C9200L-24P-4G
System Serial Number           : JPG221000RH
```

```
%INIT: waited 0 seconds for NVRAM to be available
```

```
Defaulting CPP : Policer rate for all classes will be set to their defaults
```

```
Press RETURN to get started!
```

この例では、バンドルモードでのソフトウェアブートアップの表示を示します。

```
switch: boot flash: cat9k_lite_iosxe.16.09.01.SPA.bin

Attempting to boot from [flash: cat9k_lite_iosxe.16.09.01.SPA.bin]
Located cat9k_lite_iosxe.16.09.01.SPA.bin
#####
Warning: ignoring ROMMON var "BOOT_PARAM"

Waiting for 120 seconds for other switches to boot
#####
Switch number is 3
```

Restricted Rights Legend

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) of the Commercial Computer Software - Restricted Rights clause at FAR sec. 52.227-19 and subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706

Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.9.1, RELEASE SOFTWARE (fc2)
Technical Support: <http://www.cisco.com/techsupport>
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Tue 30-May-17 00:36 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2017 by cisco Systems, Inc. All rights reserved. Certain components of Cisco IOS-XE software are licensed under the GNU General Public License ("GPL") Version 2.0. The software code licensed under GPL Version 2.0 is free software that comes with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such GPL code under the terms of GPL Version 2.0. For more details, see the documentation or "License Notice" file accompanying the IOS-XE software, or the applicable URL provided on the flyer accompanying the IOS-XE software.

FIPS: Flash Key Check : Begin
FIPS: Flash Key Check : End, Not Found, FIPS Mode Not Enabled

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at: <http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to

```

export@cisco.com.

cisco C9200L-24P-4G (ARM64) processor with 518473K/3071K bytes of memory.
Processor board ID JPG221000RH
988 Virtual Ethernet interfaces
56 Gigabit Ethernet interfaces
2048K bytes of non-volatile configuration memory.
2015456K bytes of physical memory.
819200K bytes of Crash Files at crashinfo:.
1941504K bytes of Flash at flash:.
0K bytes of WebUI ODM Files at webui:.
819200K bytes of Crash Files at crashinfo-7:.
1941504K bytes of Flash at flash-7:.

Base Ethernet MAC Address       : 68:2c:7b:f7:49:00
Motherboard Assembly Number     : 73-18699-2
Motherboard Serial Number       : JAE22090AZB
Model Revision Number           : 13
Motherboard Revision Number     : 05
Model Number                    : C9200L-24P-4G
System Serial Number            : JPG221000RH

%INIT: waited 0 seconds for NVRAM to be available

Defaulting CPP : Policer rate for all classes will be set to their defaults

Press RETURN to get started!

```

デバイスのセットアップを実行する場合の設定例

例：更新プログラムパッケージの管理

次に、ソフトウェアパッケージファイルを追加する例を示します。

```

Device# install add file flash:cat9k_lite_iosxe.16.09.01.SPA.bin activate commit

install_add_activate_commit: START Thu Aug 30 20:25:35 IST 2018

Aug 30 20:25:38.688 IST: %INSTALL-5-INSTALL_START_INFO: Switch 7 R0/0: install_engine:
Started install one-shot
flash:cat9k_lite_iosxe.16.09.01.SPA.bininstall_add_activate_commit: Adding PACKAGE

This operation requires a reload of the system. Do you want to proceed?
Please confirm you have changed boot config to flash:packages.conf [y/n]y

--- Starting initial file syncing ---
[7]: Copying flash:cat9k_lite_iosxe.16.09.01.SPA.bin from switch 7 to switch 4
[4]: Finished copying to switch 4
Info: Finished copying flash:cat9k_lite_iosxe.16.09.01.SPA.bin to the selected switch(es)
Finished initial file syncing

--- Starting Add ---
Performing Add on all members

```

```

    [4] Add package(s) on switch 4
    [4] Finished Add on switch 4
    [7] Add package(s) on switch 7
    [7] Finished Add on switch 7
Checking status of Add on [4 7]
Add: Passed on [4 7]
Finished Add

install_add_activate_commit: Activating PACKAGE

gzip: initramfs.cpio.gz: decompression OK, trailing garbage ignored
Following packages shall be activated:
/flash/cat9k_lite-webui.16.09.01.SPA.pkg
/flash/cat9k_lite-srdriver.16.09.01.SPA.pkg
/flash/cat9k_lite-rpboot.16.09.01.SPA.pkg
/flash/cat9k_lite-rpbase.16.09.01.SPA.pkg

This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on all members

Aug 30 20:51:16.365 IST: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 7 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 seconds [4] Activate
package(s) on switch 4
    [4] Finished Activate on switch 4
    [7] Activate package(s) on switch 7

Aug 30 20:51:17.561 IST: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 4 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 seconds [7] Finished
Activate on switch 7
Checking status of Activate on [4 7]
Activate: Passed on [4 7]
Finished Activate

--- Starting Commit ---
Performing Commit on all members
    [4] Commit package(s) on switch 4
    [4] Finished Commit on switch 4
    [7] Commit package(s) on switch 7
    [7] Finished Commit on switch 7
Checking status of Commit on [4 7]
Commit: Passed on [4 7]
Finished Commit

Install will reload the system now!
SUCCESS: install_add_activate_commit Thu Aug 30 20:51:55 IST 2018

Y2#
    Chassis 7 reloading, reason - Reload command

Aug 30 20:51:56.017 IST: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 7 R0/0: install_engine:
Completed install one-shot PACKAGE flash:cat9k_lite_iosxe.16.09.01.SPA.binAug 30
20:52:03.517: %PMAN-5-EXITACTION: F0/0: pvp: Process manager is exiting: reload fp action
requested
Aug 30 20:52:07.543: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: rp
processes exit with reload switch code

Aug 30 20:52:11.104: %PMAN-5-EXITACTION: C0/0: pvp: Process manager is exiting: reload
cc action requested
reboot: Restarting system

```

次に、ソフトウェア パッケージ ファイルをデバイスに追加した後の **show install summary** コマンドの出力例を示します。

```

Device# show install summary
[ Switch 4 7 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
IMG   C    16.9.1.0.70
-----
Auto abort timer: inactive
-----

```

次に、追加したソフトウェアパッケージファイルをアクティブ化する例を示します。

次に示すのは、**show install summary** コマンドがソフトウェアパッケージのステータスをアクティブでありコミット未完了と表示する場合の出力例です。

次の例では、**install commit** コマンドの実行方法を示しています。

次の例は、更新プログラムパッケージを基本パッケージにロールバックする方法を示しています。

次に、**install remove inactive** コマンドの出力例を示します。

次に、**install abort** コマンドの出力例を示します。

次に、**install activate auto-abort-timer** コマンドの出力例を示します。

ソフトウェアインストールの確認

手順

ステップ1 enable

例：

```
Device> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

ステップ2 show install log

例：

```
Device# show install log
```

デバイスの起動以降に実行されたすべてのソフトウェアインストール動作に関する情報を表示します。

```
Device# show install log
[0|install_op_boot]: START Tue Aug 30 06:39:48 Universal 2018
[0|install_op_boot]: END SUCCESS Tue Aug 30 06:39:50 Universal 2018
```

ステップ3 show install summary

例：

```
Device# show install summary
```

すべてのメンバ/現場交換可能ユニット (FRU) のイメージのバージョンとそれらに対応するインストール状態に関する情報を表示します。

- このコマンドの出力は、実行した **install** コマンドによって異なります。

```
Device# show install summary
[ Switch 1 2 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
IMG   C    16.9.1.0.70
-----
Auto abort timer: inactive
-----
```

ステップ4 show install package filename: filename

例：

```
Device# show install package flash:cat9k_lite-rpboot.16.09.01.SPA.pkg
```

指定したソフトウェア インストール パッケージ ファイルに関する情報を表示します。

```
Device# show install package flash:cat9k_lite-rpboot.16.09.01.SPA.pkg
Package: cat9k_lite-rpboot.16.09.01.SPA.pkg
Size: 34616705
Timestamp: Thu Aug 30 20:28:25 2018 UTC
Canonical path: /flash/cat9k_lite-rpboot.16.09.01.SPA.pkg

Raw disk-file SHA1sum:
 5e816f97bcae3e30eb8bc2f0ec8f64402cea1638
Header size:      980 bytes
Package type:     30001
Package flags:    0
Header version:   3

Package is bootable on RP when specified
by packages provisioning file.
```

例：デバイスを DHCP サーバとして設定

```
Device# configure terminal
Device(config)# ip dhcp pool pool1
Device(dhcp-config)# network 10.10.10.0 255.255.255.0
Device(dhcp-config)# boot config-boot.text
```

例 : DHCP 自動イメージアップデートの設定

```

Device(dhcp-config)# default-router 10.10.10.1
Device(dhcp-config)# option 150 10.10.10.1
Device(dhcp-config)# exit
Device(config)# tftp-server flash:config-boot.text
Device(config)# interface gigabitethernet1/0/4
Device(config-if)# no switchport
Device(config-if)# ip address 10.10.10.1 255.255.255.0
Device(config-if)# end

```

例 : DHCP 自動イメージアップデートの設定

```

Device# configure terminal
Device(config)# ip dhcp pool pool1
Device(dhcp-config)# network 10.10.10.0 255.255.255.0
Device(dhcp-config)# boot config-boot.text
Device(dhcp-config)# default-router 10.10.10.1
Device(dhcp-config)# option 150 10.10.10.1
Device(dhcp-config)# option 125 hex
0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370
Device(dhcp-config)# exit
Device(config)# tftp-server flash:config-boot.text
Device(config)# tftp-server flash:image_name
Device(config)# tftp-server flash:boot-config.text
Device(config)# tftp-server flash:autoinstall_dhcp
Device(config)# interface gigabitethernet1/0/4
Device(config-if)# no switchport
Device(config-if)# ip address 10.10.10.1 255.255.255.0
Device(config-if)# end

```

例 : DHCP サーバから設定をダウンロードするためのデバイスの設定

次に、VLAN 99 上のレイヤ 3 SVI インターフェイスを使用し、保存されているコンフィギュレーションで DHCP ベースの自動設定をイネーブルにする例を示します。

```

Device# configure terminal
Device(config)# boot host dhcp
Device(config)# boot host retry timeout 300
Device(config)# banner config-save ^C Caution - Saving Configuration File to NVRAM May
Cause You to No longer Automatically Download Configuration Files at Reboot^C
Device(config)# vlan 99
Device(config-vlan)# interface vlan 99
Device(config-if)# no shutdown
Device(config-if)# end
Device# show boot
BOOT path-list:
Config file:          flash:/config.text
Private Config file:  flash:/private-config.text
Enable Break:        no
Manual Boot:         no
HELPER path-list:
NVRAM/Config file
  buffer size:       32768
Timeout for Config
  Download:          300 seconds
Config Download

```



```
via DHCP:          enabled (next boot: enabled)
Device#
```

例：ソフトウェアイメージのリロードのスケジューリング

次に、当日の午後 7 時 30 分に、ソフトウェアをデバイスにリロードする例を示します。

```
Device# reload at 19:30
Reload scheduled for 19:30:00 UTC Wed Jun 5 2013 (in 2 hours and 25 minutes)
Proceed with reload? [confirm]
```

次に、未来の日時を指定して、ソフトウェアをデバイスにリロードする例を示します。

```
Device# reload at 02:00 jun 20
Reload scheduled for 02:00:00 UTC Thu Jun 20 2013 (in 344 hours and 53 minutes)
Proceed with reload? [confirm]
```

デバイスセットアップの実行に関する追加情報

関連資料

関連項目	マニュアルタイトル
デバイスセットアップコマンド ブートローダコマンド	<i>Command Reference (Catalyst 9200 Series Switches)</i>
ハードウェアの設置	<i>Cisco Catalyst 9200 シリーズ スイッチ ハードウェア設置ガイド</i>

デバイスセットアップ設定の実行に関する機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	デバイスのセットアップ設定	IP アドレス割り当てと DHCP の自動設定を含むデバイスセットアップ設定を実行できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 4 章

スマート ライセンスの設定

- スマートライセンシングの設定の前提条件 (83 ページ)
- スマートライセンシングの概要 (83 ページ)
- CSSM への接続 (84 ページ)
- CSSM への接続の設定とライセンスレベルの設定 (86 ページ)
- CSSM でのデバイスの登録 (97 ページ)
- スマート ライセンスの設定のモニタリング (102 ページ)
- スマートライセンシングの設定例 (103 ページ)
- その他の参考資料 (109 ページ)
- スマートライセンスの機能の履歴 (110 ページ)

スマートライセンシングの設定の前提条件

CSSM に以下が必要です。

- Cisco スマート アカウント
- 1 つ以上のバーチャルアカウント
- 適切なアクセス権を持つユーザロール
- デバイスを登録するには、CSSM のスマート ソフトウェア ライセンシング契約に同意する必要があります。
- <https://tools.cisco.com> へのネットワーク到達可能性

スマートライセンシングの概要

シスコ スマート ライセンシングは、シスコ ポートフォリオ全体および組織全体でソフトウェアをより簡単かつ迅速に一貫して購入および管理できる柔軟なライセンスモデルです。また、これは安全です。ユーザーがアクセスできるものを制御できます。スマートライセンスを使用すると、次のことが可能になります。

- 簡単なアクティベーション：スマートライセンスは、組織全体で使用できるソフトウェアライセンスのプールを確立します。PAK（製品アクティベーションキー）は不要です。
- 管理の統合：My Cisco Entitlements（MCE）は、使いやすいポータルですべてのシスコ製品とサービスの完全なビューを提供します。
- ライセンスの柔軟性：ソフトウェアはハードウェアにノードロックされていないため、必要に応じてライセンスを簡単に使用および転送できます。

スマートライセンスを使用するには、まず Cisco Software Central でスマートアカウントを設定する必要があります（software.cisco.com）。

シスコライセンスの詳細については、cisco.com/go/licensingguide を参照してください。

CSSM の概要

Cisco Smart Software Manager（CSSM）を使用すると、1つの集中型ポータルからシスコのスマートソフトウェアライセンスすべてを管理できます。CSSM を使用して、バーチャルアカウントと呼ばれるグループ（ライセンスと製品インスタンスの集合体）でライセンスを整理および表示します。

[License] タブの [Smart Software Licensing] リンクをクリックすると、<https://software.cisco.com/#> から CSSM にアクセスできます。



(注) CSSM にアクセスするには、Chrome 32.0、Firefox 25.0、または Safari 6.0.5 の Web ブラウザを使用します。また、Javascript 1.5 以降のバージョンをブラウザで有効にする必要があります。

CSSM を使用して次のタスクを実行できます。

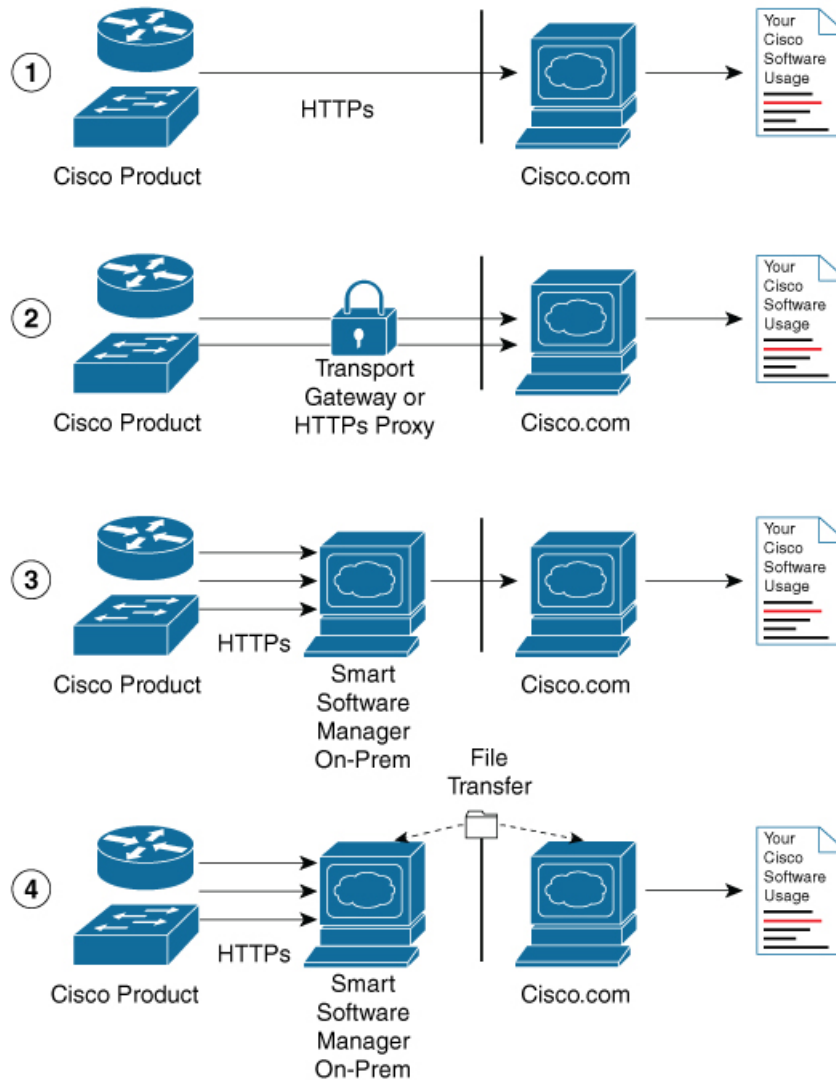
- バーチャルアカウントを作成、管理、または表示する。
- 製品インスタンスの登録トークンを作成および管理する。
- バーチャルアカウント間または表示ライセンス間でライセンスを転送する。
- 製品インスタンスを転送、削除、または表示する。
- バーチャルアカウントに関するレポートを実行する。
- 電子メール通知の設定を変更する。
- 仮想アカウント情報を表示する。

CSSM のヘルプでは、これらのタスクを実行する手順について説明しています。

CSSM への接続

次の図は、CSSM への接続に使用できるさまざまなオプションを示しています。

図 4: [Connection] のオプション



356271

1. **ダイレクトクラウドアクセス：**この方法では、シスコ製品からインターネット経由で Cisco.com に使用状況情報を直接送信します。接続に追加のコンポーネントは必要ありません。
2. **HTTPS プロキシを介したダイレクトクラウドアクセス：**この方法では、シスコ製から、インターネット経由でプロキシサーバ（Call Home トランスポートゲートウェイまたは市販のプロキシ（Apache など）のいずれか）を介して Cisco.com に使用状況情報を送信します。
3. **接続状態のオンプレミスコレクタを介した間接アクセス：**この方法では、シスコ製品から、現地のライセンス認証局として機能するローカルに接続されたコレクタに使用状況情報を送信します。データベースの同期を保つため、周期的にこの情報が交換されます。

4. 非接続状態のオンプレミスコレクタを介した間接アクセス：この方法では、シスコ製品から、現地のライセンス認証局として機能するローカルの接続が解除されたコレクタに使用状況情報を送信します。データベースの同期を保つため、不定期に（月に1回など）人による読み取りが可能な情報の交換が実施されます。

オプション1と2には簡単な接続オプションを、オプション3と4にはセキュアな環境接続オプションを提供します。Cisco Smart Software Manager オンプレミス（旧称 Cisco Smart Software Manager サテライト）は、オプション3と4のサポートを提供します。

CSSM への接続の設定とライセンスレベルの設定

ここでは、CSSM への接続の設定方法とライセンスレベルの設定方法について説明します。

CSSM への接続の設定

次の手順では、CSSM へのレイヤ3接続を設定してネットワーク到達可能性を確認する方法を説明します。CSSM へのレイヤ3接続がすでに確立されている場合は、このセクションをスキップしてください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	{ip ipv6} name-server server-address 1 [server-address 2] [server-address 3] [server-address 4] [server-address 5] [server-address 6] 例： Device(config)# ip name-server 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230	ドメインネームシステム (DNS) の設定
ステップ 4	ip name-server vrf Mgmt-vrf server-address 1 [server-address 2] [server-address 3] [server-address 4] [server-address 5] [server-address 6]	(任意) VRF インターフェイスで DNS を設定します。

	コマンドまたはアクション	目的
	例 : <pre>Device(config)# ip name-server vrf Mgmt-vrf 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230</pre>	(注) このコマンドは、 ip name-server コマンドの代わりに設定する必要があります。
ステップ 5	ip domain lookup source-interface interface-type interface-number 例 : <pre>Device(config)# ip domain lookup source-interface Vlan100</pre>	(任意) DNS ドメインルックアップ用のソースインターフェイスを設定します。
ステップ 6	ip domain name example.com 例 : <pre>Device(config)# ip domain name example.com</pre>	ドメイン名を設定します。
ステップ 7	ip host tools.cisco.com ip-address 例 : <pre>Device(config)# ip host tools.cisco.com 209.165.201.30</pre>	(任意) 自動DNSマッピングが使用できない場合は、DNS ホスト名キャッシュ内のホスト名/アドレス静的マッピングを設定します。
ステップ 8	interface vlan_id 例 : <pre>Device(config)# interface Vlan100 Device(config-if)# ip address 192.0.2.10 255.255.255.0 Device(config-if)# exit</pre>	レイヤ 3 インターフェイスを設定します。
ステップ 9	ntp server ip-address [version number] [key key-id] [prefer] 例 : <pre>Device(config)# ntp server 198.51.100.100 version 2 prefer</pre>	指定したシステムとのサーバアソシエーションを形成します。 (注) ntp server コマンドは、デバイスの時刻が CSSM と同期されるようにするために必須です。
ステップ 10	switchport access vlan vlan_id 例 : <pre>Device(config)# interface GigabitEthernet1/0/1 Device(config-if)# switchport access vlan 100 Device(config-if)# switchport mode</pre>	(任意) このアクセスポートがトラフィックを伝送する VLAN を有効にし、非ランキングで非タグ付きのシングル VLAN イーサネットインターフェイスとしてインターフェイスを設定します。

	コマンドまたはアクション	目的
	access Device(config-if)# exit Device(config)#	(注) このステップは、スイッチポートアクセスモードが必要な場合にのみ設定します。
ステップ 11	ip route ip-address ip-mask subnet mask 例： Device(config)# ip route 192.0.2.0 255.255.255.255 192.0.2.1	デバイスにルートを設定します。 (注) スタティックルートまたはダイナミックルートのいずれかを設定できます。
ステップ 12	license smart transport callhome 例： Device(config)# license smart transport callhome	転送モードを Call Home として有効にします。 (注) license smart transport callhome コマンドは必須です。
ステップ 13	ip http client source-interface interface-type interface-number 例： Device(config)# ip http client source-interface Vlan100	HTTP クライアントのソース インターフェイスを設定します。 (注) ip http client source-interface interface-type interface-number コマンドは必須です。
ステップ 14	exit 例： Device(config)# exit	(任意) グローバルコンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 15	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

ダイレクトクラウドアクセス用の Call Home サービスの設定



- (注) デフォルトでは、CiscoTAC-1 プロファイルはすでにデバイスに設定されています。プロファイルのステータスを確認するには、**show call-home profile all** コマンドを使用します。

Call Home サービスは、CSSM に対してクリティカルなシステムイベントを電子メールおよび Web 上で通知します。

Call Home サービスを設定して有効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	call-home 例： Device (config)# call-home	Call Home コンフィギュレーションモードを開始します。
ステップ 4	no http secure server-identity-check 例： Device (config-call-home)# no http secure server-identity-check	HTTP 接続の確立時のサーバ ID チェックを無効にします。
ステップ 5	contact-email-address email-address 例： Device (config-call-home)# contact-email-addr username@example.com	顧客の電子メールアドレスを割り当てます。電子メールアドレスフォーマットには、スペースなしで最大 200 文字まで入力できます。
ステップ 6	profile CiscoTAC-1 例： Device (config-call-home)# profile CiscoTAC-1	デフォルトでは、CiscoTAC-1 プロファイルは非アクティブです。このプロファイルを Call Home サービスで使用するには、プロファイルを有効にする必要があります。
ステップ 7	destination transport-method http 例： Device (config-call-home-profile)# destination transport-method http	HTTP 経由の Call Home サービスを有効にします。
ステップ 8	destination address http url 例： Device (config-call-home-profile)# destination address http https://tools.cisco.com/its/service/cthe/services/DCService	CSSM に接続します。
ステップ 9	active 例： Device (config-call-home-profile)# active	宛先プロファイルをイネーブルにします。

	コマンドまたはアクション	目的
ステップ 10	no destination transport-method email 例： Device(config-call-home-profile)# no destination transport-method email	電子メールによる Call Home サービスを無効にします。
ステップ 11	exit 例： Device(config-call-home-profile)# exit	Call Home 宛先プロファイル コンフィギュレーションモードを終了して、Call Home コンフィギュレーションモードに戻ります。
ステップ 12	exit 例： Device(config-call-home)# exit	Call Home コンフィギュレーションモードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 13	service call-home 例： Device(config)# service call-home	Call Home 機能をイネーブルにします。
ステップ 14	exit 例： Device(config)# exit	グローバル コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。
ステップ 15	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

HTTPS プロキシサーバを介したダイレクトクラウドアクセス用の Call Home サービスの設定

Call Home サービスは、HTTPS プロキシサーバを介して設定できます。この設定では、CSSM への接続にユーザ認証は必要ありません。



(注) 認証された HTTPS プロキシ設定はサポートされていません。

HTTPS プロキシを介して Call Home サービスを設定して有効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	call-home 例： Device (config)# call-home	Call Home コンフィギュレーション モードを開始します。
ステップ 4	contact-email-address email-address 例： Device (config-call-home)# contact-email-addr sch-smart-licensing@cisco.com	デフォルトの電子メールアドレスを sch-smart-licensing@cisco.com として設定します。
ステップ 5	http-proxy proxy-address proxy-port port-number 例： Device (config-call-home)# http-proxy 198.51.100.10 port 3128	Call Home サービスへのプロキシサーバ情報を設定します。
ステップ 6	profile CiscoTAC-1 例： Device (config-call-home)# profile CiscoTAC-1	デフォルトでは、CiscoTAC-1 プロファイルは非アクティブです。このプロファイルを Call Home サービスで使用するには、プロファイルを有効にする必要があります。
ステップ 7	destination transport-method http 例： Device (config-call-home-profile)# destination transport-method http	HTTP 経由の Call Home サービスを有効にします。
ステップ 8	no destination transport-method email 例： Device (config-call-home-profile)# no destination transport-method email	電子メールによる Call Home サービスを無効にします。
ステップ 9	profile name 例：	指定された宛先プロファイル名の Call Home 宛先プロファイル コンフィギュレーションモードを開始します。指定

	コマンドまたはアクション	目的
	<code>Device(config-call-home)# profile test1</code>	された宛先プロファイルが存在しない場合、作成されます。
ステップ 10	reporting smart-licensing-data 例： <code>Device(config-call-home-profile)# reporting smart-licensing-data</code>	HTTP 経由の Call Home サービスとのデータ共有を有効にします。
ステップ 11	destination transport-method http 例： <code>Device(config-call-home-profile)# destination transport-method http</code>	HTTP メッセージの転送方法をイネーブルにします。
ステップ 12	destination address http url 例： <code>Device(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/otte/services/DTOService</code>	CSSM に接続します。
ステップ 13	active 例： <code>Device(config-call-home-profile)# active</code>	宛先プロファイルをイネーブルにします。
ステップ 14	exit 例： <code>Device(config-call-home-profile)# exit</code>	Call Home 宛先プロファイル コンフィギュレーションモードを終了して、Call Home コンフィギュレーションモードに戻ります。
ステップ 15	exit 例： <code>Device(config-call-home)# exit</code>	Call Home コンフィギュレーションモードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 16	service call-home 例： <code>Device(config)# service call-home</code>	Call Home 機能をイネーブルにします。
ステップ 17	ip http client proxy-server proxy-address proxy-port port-number 例： <code>Device(config)# ip http client proxy-server 198.51.100.10 port 3128</code>	Call Home 機能をイネーブルにします。
ステップ 18	exit 例： <code>Device(config)# exit</code>	グローバル コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 19	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

Cisco Smart Software Manager オンプレミス用の Call Home サービスの設定

Smart Software Manager オンプレミス (旧称 Cisco Smart Software Manager サテライト) については、<https://www.cisco.com/c/en/us/buy/smart-accounts/software-manager-satellite.html> を参照してください。

Cisco Smart Software Manager オンプレミス (旧称 Cisco Smart Software Manager サテライト) 用の Call Home サービスを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	call-home 例 : Device (config)# call-home	Call Home コンフィギュレーションモードを開始します。
ステップ 4	no http secure server-identity-check 例 : Device (config-call-home)# no http secure server-identity-check	HTTP 接続の確立時のサーバ ID チェックを無効にします。
ステップ 5	profile name 例 : Device (config-call-home)# profile test1	指定された宛先プロファイル名の Call Home 宛先プロファイル コンフィギュレーションモードを開始します。指定された宛先プロファイルが存在しない場合、作成されます。

	コマンドまたはアクション	目的
ステップ 6	reporting smart-licensing-data 例： Device(config-call-home-profile)# reporting smart-licensing-data	HTTP 経由の Call Home サービスとのデータ共有を有効にします。
ステップ 7	destination transport-method http 例： Device(config-call-home-profile)# destination transport-method http	HTTP メッセージの転送方法をイネーブルにします。
ステップ 8	destination address http url 例： Device(config-call-home-profile)# destination address http https://209.165.201.15:443/transportgate/services/DeviceRequestHandler または Device(config-call-home-profile)# destination address http http://209.165.201.15:80/transportgate/services/DeviceRequestHandler	Call Home メッセージが送信される宛先 URL (CSSM) を設定します。 注 宛先 URL の IP アドレスまたは完全修飾ドメイン名 (FQDN) が、Cisco Smart Software Manager オンプレミスの [Satellite Name] に設定されている IP アドレスまたは FQDN と一致することを確認します。
ステップ 9	destination preferred-msg-format {long-text short-text xml} 例： Device(config-call-home-profile)# destination preferred-msg-format xml	(任意) 使用するメッセージ形式を設定します。デフォルトは XML です。
ステップ 10	active 例： Device(config-call-home-profile)# active	宛先プロファイルをイネーブルにします。デフォルトでは、プロファイルは作成時にイネーブルになります。
ステップ 11	exit 例： Device(config-call-home-profile)# exit	Call Home 宛先プロファイル コンフィギュレーションモードを終了して、Call Home コンフィギュレーションモードに戻ります。
ステップ 12	exit 例： Device(config-call-home)# exit	Call Home コンフィギュレーションモードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 13	ip http client source-interface interface-type interface-number 例：	HTTP クライアントのソース インターフェイスを設定します。

	コマンドまたはアクション	目的
	<code>Device(config)# ip http client source-interface Vlan100</code>	(注) ip http client source-interface interface-type interface-number コマンドは、vrf インターフェイスでは必須です。
ステップ 14	crypto pki trustpoint name 例： <code>Device(config)# crypto pki trustpoint SLA-TrustPoint</code>	(任意) トラストポイントおよび設定された名前を宣言して、CA トラストポイントコンフィギュレーションモードを開始します。
ステップ 15	revocation-check none 例： <code>Device(ca-trustpoint)# revocation-check none</code>	(任意) 証明書の確認が無視されることを指定します。
ステップ 16	end 例： <code>Device(ca-trustpoint)# end</code>	(任意) CA トラストポイントコンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。
ステップ 17	copy running-config startup-config 例： <code>Device# copy running-config startup-config</code>	(任意) コンフィギュレーションファイルに設定を保存します。

ライセンスレベルの設定

この手順は任意です。次の手順を使用すると、以下のことができます。

- ライセンスのダウングレードとアップグレード
- 評価ライセンスと拡張ライセンスの有効化と無効化
- アップグレードライセンスのクリア

登録する前に、必要なライセンスレベルをデバイスで設定する必要があります。Cisco Catalyst 9000 シリーズ スイッチで使用できるライセンスレベルは次のとおりです。

基本ライセンス

- Network Essentials
- Network Advantage (Network Essentials を含む)

アドオンライセンス：3年、5年、または7年の固定期間にわたって次のライセンスをサブスクライブできます。

- Digital Networking Architecture (DNA) Essentials

- Cisco DNA Advantage (Cisco DNA Essentials を含む)

ライセンスレベルを設定するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	license boot level license_level 例： Device(config)# license boot level network-essentials	スイッチのライセンスをアクティブ化します。
ステップ 4	exit 例： Device(config)# exit	特権 EXEC モードに戻ります。
ステップ 5	write memory 例： Device# write memory	ライセンス情報をスイッチに保存します。
ステップ 6	show version 例： Device# show version	ライセンスレベルの情報を表示します。
	Technology-package Current Type Technology-package Next reboot network-essentials Smart License network-essentials None Subscription Smart License None	
ステップ 7	reload 例： Device# reload	デバイスがリロードされます。

CSSM でのデバイスの登録

CSSM でデバイスを登録するには、次のタスクを実行する必要があります。

1. CSSM から一意のトークンを生成します。
2. 生成されたトークンを使用してデバイスを登録します。

登録が成功すると、デバイスで ID 証明書を受信します。この証明書はデバイスに保存され、それ以降のシスコとのすべての通信で自動的に使用されます。CSSM は 30 日ごとに登録情報の更新を試みます。

また、ライセンスの使用状況データが収集され、毎月レポートが送信されます。必要に応じて、機密情報（ホスト名、ユーザ名、パスワードなど）が使用状況レポートから除外されるように Call Home 設定を構成できます。

CSSM からの新しいトークンの生成

新しい製品インスタンスをバーチャルアカウントに登録するために、トークンが生成されます。

手順

-
- ステップ 1 <https://software.cisco.com/#> から CSSM にログインします。
シスコから提供されたユーザ名とパスワードを使用してポータルにログインする必要があります。
 - ステップ 2 [Inventory] タブをクリックします。
 - ステップ 3 [Virtual Account] ドロップダウンリストから、必要なバーチャルアカウントを選択します。
 - ステップ 4 [General] タブをクリックします。
 - ステップ 5 [新規トークン (New Token)] をクリックします。

Cisco Software Central > Smart Software Licensing

Smart Software Licensing

Virtual Account: Virtual Account 1

General Licenses Product Instances Event Log

Virtual Account

Description: Account 1

Default Virtual Account: No

Product Instance Registration Tokens

The registration tokens below can be used to register new product instances to this virtual account.

New Token...

Token	Expiration Date	Description	Export-Controlled	Created By	Actions
ZjgxNzdjYjctOVRhMCO0M2IOL	Expired	Token 1	Allowed	User 1	Actions
ZTg2MjBjMzUIN2U0Ni00NDdkL	Expired		Allowed	User 1	Actions

[Create Registration Token] ウィンドウが表示されます。

ステップ 6 [Description] フィールドに、トークンの説明を入力します。

ステップ 7 [Expire After] フィールドに、トークンをアクティブにする必要がある日数を入力します。

ステップ 8 (任意) [Max. Number of Uses] フィールドに、トークンの有効期限が切れるまでの最大使用回数を入力します。

Create Registration Token

This will create a token that is used to register product instances, so that they can use licenses from this virtual account. Once it's created, go to the Smart Licensing configuration for your products and enter the token, to register them with this virtual account.

Virtual Account: Virtual Account 1

Description: Token 2

* Expire After: 30 Days

Between 1 - 365, 30 days recommended

Max. Number of Uses:

The token will be expired when either the expiration or the maximum uses is reached

Allow export-controlled functionality on the products registered with this token

Create Token Cancel

ステップ 9 [Allow export-controlled functionality on the products registered with this token] チェックボックスをオンにします。

このチェックボックスをオンにすると、シスコは米国および各国固有のエクスポートポリシーおよびガイドラインに準拠するようになります。詳細については、<https://www.cisco.com/c/en/us/about/legal/global-export-trade.html>を参照してください。

ステップ 10 [Create Token] をクリックしてトークンを作成します。

ステップ 11 トークンを作成したら、[Copy] をクリックし、新しく作成したトークンをコピーします。

Create Registration Token ? ×

This will create a token that is used to register product instances, so that they can use licenses from this virtual account. Once it's created, go to the Smart Licensing configuration for your products and enter the token, to register them with this virtual account.

Virtual Account: Virtual Account 1

Description:

* Expire After: Days
Between 1 - 365, 30 days recommended

Max. Number of Uses:

The token will be expired when either the expiration or the maximum uses is reached

Allow export-controlled functionality on the products registered with this token ?

Create Token
Cancel

新しいトークンを使用するデバイスの登録

新しいトークンを使用してデバイスを登録するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	license smart register idtoken token_ID 例： Device# license smart register idtoken \$M4y4Np4Eslc0eUWcFzZTRdLW5Fe%0hIRbz0%3D0A	CSSM から生成されたトークンを使用して、デバイスをバックエンドサーバに登録します。
ステップ 3	write memory 例： Device# write memory	ライセンス情報をデバイスに保存します。

登録後のライセンスステータスの確認

登録後にライセンスのステータスを確認するには、**show license all** コマンドを使用します。

```
Device> enable
Device# show license all
```

```
Smart Licensing Status
=====
```

```
Smart Licensing is ENABLED
```

```
Registration:
```

```
Status: REGISTERED
Smart Account: Smart Account Name
Virtual Account: Virtual Account 1
Export-Controlled Functionality: Allowed
Initial Registration: First Attempt Pending
Last Renewal Attempt: SUCCEEDED on Jul 19 14:49:49 2018 IST
Next Renewal Attempt: Jan 15 14:49:48 2019 IST
Registration Expires: Jul 19 14:43:48 2019 IST
```

```
License Authorization:
```

```
Status: AUTHORIZED on Jul 28 07:02:56 2018 IST
Last Communication Attempt: SUCCEEDED on Jul 28 07:02:56 2018 IST
Next Communication Attempt: Aug 27 07:02:56 2018 IST
Communication Deadline: Oct 26 06:57:50 2018 IST
```

```
Utility:
```

```
Status: DISABLED
```

```
Data Privacy:
```

```
Sending Hostname: yes
Callhome hostname privacy: DISABLED
Smart Licensing hostname privacy: DISABLED
Version privacy: DISABLED
```

```
Transport:
```

```
Type: Callhome
```

```
License Usage
=====
```

```
C9200L DNA Advantage, 48-port Term license (C9200L-DNA-A-48):
```

```
Description: C9200L DNA Advantage, 48-port Term license
Count: 1
Version: 1.0
Status: AUTHORIZED
```

```
C9200L Network Advantage, 48-port license (C9200L-NW-A-48):
```

```
Description: C9200L Network Advantage, 48-port license
Count: 1
Version: 1.0
Status: AUTHORIZED
```

```
Product Information
=====
```

```
UDI: PID:C9200L-48P-4X,SN:JPG221300KP
```

```
Agent Version
=====
```

```
Smart Agent for Licensing: 4.4.13_rel/116
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel15)1.0.3, PK:(dev18)1.0.3
```

```
Reservation Info
=====
```

```
License reservation: DISABLED
```

CSSM でのデバイスの登録キャンセル

デバイスがインベントリから移された場合、再導入のために別の場所に出荷された場合、または返品許可（RMA）プロセスを使用して交換のためにシスコに返送された場合は、**deregister** コマンドを使用してデバイスの登録をキャンセルできます。

デバイス登録をキャンセルするには、次の手順に従います。

始める前に

デバイスを正常に登録解除するには、CSSM へのレイヤ 3 接続が使用可能である必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	license smart deregister 例： Device# license smart deregister	デバイスの登録をキャンセルし、デバイスを評価モードに送信します。対応するプラットフォームのすべてのスマートライセンス資格と証明書が削除されます。CSSM に保存されているデバイス製品インスタンスも削除されます。

スマートライセンスの設定のモニタリング

スマートライセンスの設定をモニタするには、特権 EXEC モードで次のコマンドを使用します。

表 6: スマートライセンスの設定をモニタリングするコマンド

コマンド	目的
<code>show license status</code>	<p>スマートライセンスのコンプライアンスステータスを表示します。以下は、表示される可能性があるステータスのリストです。</p> <ul style="list-style-type: none"> • Enabled : スマートライセンスが有効になっていることを示します。 • Waiting : デバイスがライセンス権限付与要求を行った後の初期状態を示します。デバイスはシスコとの通信を確立し、CSSM に正常に登録されます。 • Registered : デバイスが CSSM と通信できること、およびライセンス権限付与の要求を開始する権限を持っていることを示します。 • Authorized : デバイスがコンプライアンスステータスであり、要求されたライセンスのタイプおよび数を使用する権限があることを示します。承認ステータスのライフタイムは90日です。承認を更新するため、デバイスは30日後にCSSM に新しい権限承認要求を送信します。 • Out Of Compliance : 1つ以上のライセンスがコンプライアンス違反になっていることを示します。追加ライセンスを購入する必要があります。 • Eval Mode : (デバイスの使用後) 90日以内にCSSM にデバイスを登録する必要があります。登録しない場合、デバイスの評価期間が終了します。 • Evaluation Period Expired : デバイスが登録されていない場合、デバイスは90日後に評価期間終了モードになります。

コマンド	目的
show license all	使用中のすべての権限を表示します。さらに、関連付けられているライセンス証明書、コンプライアンスステータス、UDI、およびその他の詳細が表示されます。
show tech-support license	詳細なデバッグ出力を表示します。
show license usage	ライセンスの使用情報を表示します。
show license summary	すべてのアクティブなライセンスの要約を表示します。

スマートライセンシングの設定例

ここでは、さまざまなスマートライセンスの設定例を示します。

例：Call Home プロファイルの表示

例

Call Home プロファイルを表示するには、**show call-home profile all** コマンドを使用します。

```
Device> enable
Device# show call-home profile all
Profile Name: CiscoTAC-1
  Profile status: ACTIVE
  Profile mode: Full Reporting
  Reporting Data: Smart Call Home, Smart Licensing
  Preferred Message Format: xml
  Message Size Limit: 3145728 Bytes
  Transport Method: http
  HTTP address(es): https://tools.cisco.com/its/service/oddce/services/DDCEService
  Other address(es): default

  Periodic configuration info message is scheduled every 1 day of the month at 09:15

  Periodic inventory info message is scheduled every 1 day of the month at 09:00

Alert-group          Severity
-----
crash                debug
diagnostic           minor
environment          warning
inventory            normal

Syslog-Pattern      Severity
-----
```

```
APF-.-WLC_.*          warning
.*                    major
```

例：登録前のライセンス情報の表示

例

ライセンスの付与資格を表示するには、**show license all** コマンドを使用します。

```
Device> enable
Device# show license all
Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: Not Allowed

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 30 minutes, 5 seconds

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

License Usage
=====

C9200L DNA Advantage, 48-port Term license (C9200L-DNA-A-48):
  Description: C9200L DNA Advantage, 48-port Term license
  Count: 1
  Version: 1.0
  Status: EVAL MODE

C9200L Network Advantage, 48-port license (C9200L-NW-A-48):
  Description: C9200L Network Advantage, 48-port license
  Count: 1
  Version: 1.0
  Status: EVAL MODE

Product Information
=====
UDI: PID:C9200L-48P-4X,SN:JPG221300KP

Agent Version
=====
Smart Agent for Licensing: 4.4.13_rel/116
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel15)1.0.3, PK:(dev18)1.0.3
```



```
Reservation Info
=====
License reservation: DISABLED
```

例

ライセンス使用情報を表示するには、**show license usage** コマンドを使用します。

```
Device> enable
Device# show license usage
Load for five secs: 1%/0%; one minute: 0%; five minutes: 0%
No time source, 09:28:34.123 EDT Fri Jul 13 2018

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 38 seconds

C9200L DNA Advantage, 48-port Term license (C9200L-DNA-A-48):
  Description: C9200L DNA Advantage, 48-port Term license
  Count: 1
  Version: 1.0
  Status: EVAL MODE

C9200L Network Advantage, 48-port license (C9200L-NW-A-48):
  Description: C9200L Network Advantage, 48-port license
  Count: 1
  Version: 1.0
  Status: EVAL MODE
```

例

すべてのライセンスの概要を表示するには、**show license summary** コマンドを使用します。

```
Device> enable
Device# show license summary

Smart Licensing is ENABLED

Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: Not Allowed

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 33 seconds

License Usage:
  License                               Entitlement tag                Count Status
  -----
                               (C9200L-DNA-A-48)              1 EVAL MODE
                               (C9200L-NW-A-48)              1 EVAL MODE
```

例

ライセンスのステータス情報を表示するには、**show license status** コマンドを使用します。

```
Device> enable
Device# show license status
Load for five secs: 1%/0%; one minute: 0%; five minutes: 0%
No time source, 09:28:37.683 EDT Fri Jul 13 2018

Smart Licensing is ENABLED

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: Not Allowed

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 35 seconds
```

例：デバイスの登録

例

デバイスを登録するには、**license smart register idtoken** コマンドを使用します。

```
Device> enable
Device# license smart register idtoken
T14UytrNXBzbEs1ck8veUtWaG5abnZJOFdDa1FwbVRa%0Ab1Rmbz0%3D%0A
Device# write memory
```

例：登録後のライセンスステータスの表示

例

ライセンスの付与資格を表示するには、**show license all** コマンドを使用します。

```
Device> enable
Device# show license all
```

```
Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name
  Virtual Account: Virtual Account 1
  Export-Controlled Functionality: Allowed
  Initial Registration: First Attempt Pending
  Last Renewal Attempt: SUCCEEDED on Jul 19 14:49:49 2018 IST
  Next Renewal Attempt: Jan 15 14:49:48 2019 IST
  Registration Expires: Jul 19 14:43:48 2019 IST

License Authorization:
  Status: AUTHORIZED on Jul 28 07:02:56 2018 IST
  Last Communication Attempt: SUCCEEDED on Jul 28 07:02:56 2018 IST
  Next Communication Attempt: Aug 27 07:02:56 2018 IST
  Communication Deadline: Oct 26 06:57:50 2018 IST

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

License Usage
=====

C9200L DNA Advantage, 48-port Term license (C9200L-DNA-A-48):
  Description: C9200L DNA Advantage, 48-port Term license
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

C9200L Network Advantage, 48-port license (C9200L-NW-A-48):
  Description: C9200L Network Advantage, 48-port license
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

Product Information
=====
UDI: PID:C9200L-48P-4X,SN:JPG221300KP

Agent Version
=====
Smart Agent for Licensing: 4.4.13_rel/116
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel15)1.0.3, PK:(dev18)1.0.3

Reservation Info
=====
License reservation: DISABLED
```

例

ライセンス使用情報を表示するには、**show license usage** コマンドを使用します。

```
Device> enable
Device# show license usage
License Authorization:
  Status: AUTHORIZED on Jul 28 07:02:56 2018 IST

C9200L DNA Advantage, 48-port Term license (C9200L-DNA-A-48):
  Description: C9200L DNA Advantage, 48-port Term license
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

C9200L Network Advantage, 48-port license (C9200L-NW-A-48):
  Description: C9200L Network Advantage, 48-port license
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
```

例

すべてのライセンスの概要を表示するには、**show license summary** コマンドを使用します。

```
Device> enable
Device# show license summary
Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name
  Virtual Account: Virtual Account 1
  Export-Controlled Functionality: Allowed
  Last Renewal Attempt: SUCCEEDED
  Next Renewal Attempt: Jan 15 14:49:48 2019 IST

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Aug 27 07:02:56 2018 IST

License Usage:
  License                               Entitlement tag                Count Status
  -----
  C9200L DNA Advantage... (C9200L-DNA-A-48) 1 AUTHORIZED
  C9200L Network Advan... (C9200L-NW-A-48) 1 AUTHORIZED
```

例

ライセンスのステータス情報を表示するには、**show license status** コマンドを使用します。

```
Device> enable
```

```

Device# show license status

Smart Licensing is ENABLED

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name
  Virtual Account: Virtual Account 1
  Export-Controlled Functionality: Allowed
  Initial Registration: First Attempt Pending
  Last Renewal Attempt: SUCCEEDED on Jul 19 14:49:49 2018 IST
  Next Renewal Attempt: Jan 15 14:49:47 2019 IST
  Registration Expires: Jul 19 14:43:47 2019 IST

License Authorization:
  Status: AUTHORIZED on Jul 28 07:02:56 2018 IST
  Last Communication Attempt: SUCCEEDED on Jul 28 07:02:56 2018 IST
  Next Communication Attempt: Aug 27 07:02:56 2018 IST
  Communication Deadline: Oct 26 06:57:50 2018 IST

```

その他の参考資料

関連資料

関連項目	マニュアルタイトル
Cisco Smart Software Manager のヘルプ	Smart Software Manager Help
Cisco Smart Software Manager オンプレミス	Cisco Smart Software Manager On-Prem

シスコのテクニカル サポート

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

スマートライセンスの機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	スマートライセンス	<p>クラウドベースのソフトウェアライセンス管理ソリューションであり、ライセンス、ハードウェア、およびソフトウェアの使用状況の傾向を管理および追跡できます。</p> <p>スマートライセンスはデフォルトであり、ライセンスを管理するために使用できる唯一の方法です。</p>

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 5 章

有線ネットワークでの Application Visibility and Control の設定

- [有線ネットワークでの Application Visibility and Control について \(111 ページ\)](#)
- [サポートされる AVC クラス マップ および ポリシー マップ のフォーマット \(112 ページ\)](#)
- [有線 Application Visibility and Control の制限 \(113 ページ\)](#)
- [Application Visibility and Control の設定方法 \(115 ページ\)](#)
- [Application Visibility and Control のモニタリング \(143 ページ\)](#)
- [例：Application Visibility and Control の設定 \(143 ページ\)](#)
- [基本的なトラブルシューティング：質問と回答 \(155 ページ\)](#)
- [Application Visibility and Control に関する追加情報 \(157 ページ\)](#)
- [有線ネットワークでの Application Visibility and Control の機能履歴 \(157 ページ\)](#)

有線ネットワークでの Application Visibility and Control について

Application Visibility and Control (AVC) は、アプリケーションへの適応力やアプリケーションへのインテリジェンス性に基づいて、厳密なパケットおよび接続からブランチおよびキャンパスソリューションを発展させるためのシスコの取り組みの重要な部分です。Application Visibility and Control (AVC) は、ネットワークベースのアプリケーション認識 (NBAR2) エンジンによるディープパケットインスペクション技術を使用してアプリケーションを分類します。AVC は、スタンドアロンスイッチおよびスイッチスタックの有線アクセスポート上に設定できます。NBAR2 は、プロトコル検出を有効にすることによって明示的に、または **match protocol** 分類子を含む QoS ポリシーを接続することによって暗黙的に、インターフェイス上でアクティブにできます。有線 AVC Flexible Netflow (FNF) をインターフェイス上に設定し、インターフェイスごとのクライアント、サーバ、アプリケーションの統計情報を提供できます。このレコードは、Easy Performance Monitor (Easy perf-mon または ezPM) の **application-statistics** および **application-performance** プロファイルで利用できる **application-client-server-stats** トラフィック監視と同様です。

サポートされる AVC クラス マップおよびポリシー マップのフォーマット

ここでは、サポートされている AVC クラスマップとポリシーマップ形式について説明します。

サポートされる AVC クラス マップのフォーマット

クラスマップのフォーマット	クラスマップの例	方向
<code>match protocol protocol name</code>	<code>class-map match-any NBAR-VOICE match protocol ms-lync-audio</code>	入力と出力の両方
組み合わせフィルタ	<code>class-map match-any NBAR-VOICE match protocol ms-lync-audio match dscp ef</code>	入力と出力の両方

サポートされる AVC ポリシーのフォーマット

ポリシーのフォーマット	QoS 処理
match protocol フィルタに基づく出力ポリシー	マークおよびポリシー
match protocol フィルタに基づく入力ポリシー	マークおよびポリシー

次の表で、AVC ポリシーの詳細なフォーマット、および例について説明します。

AVC ポリシーのフォーマット	AVC ポリシーの例	方向
ベーシック セット	<code>policy-map MARKING-IN class NBAR-MM_CONFERENCEING set dscp af41</code>	入力および出力
ベーシック ポリシー	<code>policy-map POLICING-IN class NBAR-MM_CONFERENCEING police cir 600000 set dscp af41</code>	入力および出力
ベーシック セットおよびポリシー	<code>policy-map webex-policy class webex-class set dscp ef police 5000000</code>	入力および出力

AVC ポリシーのフォーマット	AVC ポリシーの例	方向
デフォルトを含む複数のセットおよびポリシー	<pre> policy-map webex-policy class webex-class set dscp af31 police 4000000 class class-webex-category set dscp ef police 6000000 class class-default set dscp <> </pre>	入力および出力
階層型ポリシー	<pre> policy-map webex-policy class webex-class police 5000000 service-policy client-in-police-only policy-map client-in-police-only class webex-class police 100000 class class-webex-category set dscp ef police 200000 </pre>	入力および出力
階層型セットおよびポリシー	<pre> policy-map webex-policy class class-default police 1500000 service policy client-up-child policy-map client-up-child class webex-class police 100000 set dscp ef class class-webex-category police 200000 set dscp af31 </pre>	

有線 Application Visibility and Control の制限

- NBAR 対応 QoS ポリシー設定は有線物理ポートでのみ許可されます。ポリシー設定は、VLAN およびその他の論理インターフェイスなどの仮想インターフェイスではサポートされていません。
- NBAR ベースの QoS ポリシー設定は、ポートチャネルメンバーポートおよび SVI やサブインターフェイスなどの仮想インターフェイスではサポートされません。
- NBAR ベースの QoS ポリシー設定は、レイヤ 2 アクセスポートとトランクポート、およびレイヤ 3 ルーテッドポートでサポートされます。
- NBAR と送信 (Tx) スイッチドポートアナライザ (SPAN) は、同じインターフェイスではサポートされません。

- プロトコルベースまたは属性ベースのいずれかのポートに同時に接続できるのは、NBAR ベースの QoS メカニズムの 1 つだけです。次の 2 つの属性のみがサポートされます。
 - traffic-class
 - business-relevance
- 従来の WDAVC QoS の制限事項は引き続き適用されます。
 - マーキングとポリシングのみがサポートされます。
 - 物理インターフェイスだけがサポートされます。
 - アプリケーション分類がオフラインで行われるため、QoS 分類には遅延があります（ただし、フローの最初のパケットは、正確な QoS 分類の前に転送されます）。
- NBAR2 ベースの一致基準 **match protocol** は、マーキングアクションおよびポリシングアクションでのみ許可されます。NBAR2 一致基準は、キューイング機能が設定されているポリシーでは許可されません。
- 「一致プロトコル」：すべてのポリシーで最大 255 の同時に異なるプロトコル（8 ビットの HW 制限）。
- AVC は管理ポート（Gig 0/0）ではサポートされていません。
- IPv6 パケットの分類はサポートされていません。
- IPv4 ユニキャスト（TCP/UDP）のみがサポートされます。
- Web UI：Web UI からアプリケーションの可視性を設定し、アプリケーションのモニタリングを実行できます。アプリケーション制御は、CLI を使用してのみ実行できます。Web UI ではサポートされていません。

Web UI 上で有線 AVC のトラフィックを管理、またはチェックするには、最初に CLI を使用して **ip http authentication local** と **ip nbar http-service** コマンドを設定する必要があります。
- NBAR および ACL のロギングは、同一スイッチ上で一緒に設定することはできません。
- プロトコル検出、アプリケーションベースの QoS、および有線 AVC FNF は、非アプリケーションベース FNF がある同一インターフェイス上で同時に設定することはできません。ただし、これらの有線 AVC 機能は、相互に設定できます。たとえば、プロトコル検出、アプリケーションベースの QoS、および有線 AVC FNF は、同一インターフェイス上で同時に設定できます。
- それぞれ異なる定義済みレコードを持つ 2 つの有線 AVC モニタのみをインターフェイスに同時に接続できます。
- 2 つの方向性フローレコード（入力と出力）と 2 つの従来のフローレコードがサポートされます。

- 接続は、物理 Layer2（アクセス/トランク）および Layer3 ポートでのみ行う必要があります。アップリンクは、単一のアップリンクであり、ポートチャネルの一部でなければ接続できません。
- パフォーマンス：各スイッチ メンバーは、50% 未満の CPU 使用率で、1 秒あたり 500 の接続（CPS）を処理できます。
- 拡張性：24 個のアクセスポートと 48 個のアクセスポートごとに最大 5000 の双方向フローを処理できます。
- 有線 AVC では、この章の手順にリストされている固定のフィールドセットのみを使用できます。その他の組み合わせは使用できません。通常の FNF フローモニタでは、他の組み合わせも使用できます（サポートされている FNF フィールドのリストについては、『*Network Management Configuration Guide*』の「Configuring Flexible NetFlow」の章を参照してください）。

Application Visibility and Control の設定方法

有線ネットワークでの Application Visibility and Control の設定

有線ポートで Application Visibility and Control を設定するには、次の手順を実行します。

可視性の設定

- インターフェイス コンフィギュレーション モードで **ip nbar protocol-discovery** コマンドを使用してインターフェイス上でプロトコル検出を有効にすることで、NBAR2 エンジン をアクティブ化します。[インターフェイスでのアプリケーション認識の有効化（116 ページ）](#) を参照してください。

制御設定：次の手順に従って、アプリケーションに基づいて QoS ポリシーを設定します。

1. AVC QoS ポリシーの作成。[AVC QoS ポリシーの作成（116 ページ）](#) を参照してください。
2. インターフェイスへの AVC QoS ポリシーの適用。[スイッチポートへの QoS ポリシーの適用（119 ページ）](#) を参照してください。

アプリケーション ベースの Flexible Netflow の設定：

- フローにキー フィールドおよび非キー フィールドを指定して、フロー レコードを作成します。[フロー レコードの作成（120 ページ）](#) を参照してください。
- フロー エクスポートを作成してフロー レコードをエクスポートします。[フロー エクスポートの作成（134 ページ）](#) を参照してください。
- フロー レコードおよびフロー エクスポートに基づいて、フロー モニタを作成します。[フロー モニタの作成（135 ページ）](#) を参照してください。

- インターフェイスにフロー モニタを接続します。 [インターフェイスへのフロー モニタの関連付け \(137 ページ\)](#) を参照してください。

プロトコル検出、アプリケーションベースの QoS およびアプリケーションベースの FNF は、すべて独立した機能です。単独で設定することも、または同じインターフェイスで同時に設定することもできます。

インターフェイスでのアプリケーション認識の有効化

インターフェイス上でアプリケーション認識をイネーブルにするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface interface-id 例： Device(config)# interface gigabitethernet 1/0/1	プロトコル検出をイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ip nbar protocol-discovery 例： Device(config-if)# ip nbar protocol-discovery	NBAR2 エンジンを実アクティブ化することで、インターフェイスでアプリケーション認識を有効にします。
ステップ 4	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。

AVC QoS ポリシーの作成

AVC QoS ポリシーを作成するには、次の一般的な手順を実行します。

1. match protocol フィルタでクラス マップを作成します。
2. ポリシー マップを作成します。

3. インターフェイスにポリシー マップを適用します。

クラス マップの作成

match protocol フィルタを設定する前に、クラス マップを作成する必要があります。マーキングやポリシングなどの QoS アクションをトラフィックに適用できます。AVC の match protocol フィルタは、有線アクセスポートに適用されます。サポートされているプロトコルの詳細については、http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/prot_lib/config_library/nbar-prot-pack-library.html 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	terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	class-map class-map-name 例： Device(config)# class-map webex-class	クラス マップを作成します。
ステップ 3	match protocol application-name 例： Device(config)# class-map webex-class Device(config-cmap)# match protocol webex-media	アプリケーション名との一致を指定します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

ポリシー マップの作成

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	policy-map policy-map-name 例：	ポリシーマップ名を入力することによってポリシー マップを作成し、ポリシー

	コマンドまたはアクション	目的
	Device(config)# policy-map webex-policy	<p>マップ コンフィギュレーション モードを開始します。</p> <p>デフォルトでは、ポリシー マップは定義されていません。</p> <p>ポリシー マップのデフォルトの動作では、パケットが IP パケットの場合は DSCP が 0 に、パケットがタグ付きの場合は CoS が 0 に設定されます。ポリシー マップは実行されません。</p> <p>(注) 既存のポリシーマップを削除するには、no policy-map <i>policy-map-name</i> グローバル コンフィギュレーション コマンドを使用します。</p>
ステップ 3	<p>class [<i>class-map-name</i> class-default]</p> <p>例 :</p> <p>Device(config-pmap)# class webex-class</p>	<p>トラフィックの分類を定義し、ポリシー マップ クラス コンフィギュレーション モードを開始します。</p> <p>デフォルトでは、ポリシー マップおよびクラスマップは定義されていません。</p> <p>すでに class-map グローバル コンフィギュレーション コマンドを使用してトラフィッククラスが定義されている場合は、このコマンドで <i>class-map-name</i> にその名前を指定します。</p> <p>class-default トラフィッククラスは定義済みで、どのポリシーにも追加できます。このトラフィック クラスは、常にポリシーマップの最後に配置されます。暗黙の match any が class-default クラスに含まれている場合、他のトラフィック クラスと一致しないパケットはすべて class-default と一致します。</p> <p>(注) 既存のクラスマップを削除するには、no class <i>class-map-name</i> ポリシー マップ コンフィギュレーション コマンドを使用します。</p>

	コマンドまたはアクション	目的
ステップ 4	police rate-bps burst-byte 例 : <pre>Device(config-pmap-c)# police 100000 80000</pre>	分類したトラフィックにポリサーを定義します。 デフォルトでは、ポリサーは定義されていません。 <ul style="list-style-type: none"> • <i>rate-bps</i> には、平均トラフィックレートをビット/秒 (bps) で指定します。指定できる範囲は 8000 ~ 10000000000 です • <i>burst-byte</i> には、標準バーストサイズをバイト数で指定します。有効範囲は、1000 ~ 512000000 です。
ステップ 5	set {dscp new-dscp cos cos-value} 例 : <pre>Device(config-pmap-c)# set dscp 45</pre>	パケットに新しい値を設定することによって、IP トラフィックを分類します。 <ul style="list-style-type: none"> • <i>dscp new-dscp</i> には、分類されたトラフィックに割り当てる新しい DSCP 値を入力します。指定できる範囲は 0 ~ 63 です。
ステップ 6	end 例 : <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。

スイッチポートへの QoS ポリシーの適用

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>Device# configure terminal</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	interface interface-id 例 : <pre>Device(config)# interface GigabitEthernet 1/0/1</pre>	インターフェイスコンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 3	service-policy input <i>polycymapname</i> 例： Device(config-if)# service-policy input MARKING_IN	インターフェイスにローカル ポリシーを適用します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

有線 AVC Flexible Netflow の設定

フロー レコードの作成

有線 AVC FNF は、従来の双方向フローレコードと方向性フローレコード（入力と出力）の 2 種類の定義済みフローレコードをサポートします。合計 4 つの異なる定義済みフローレコード（2 つの双方向フローレコードと 2 つの方向性フローレコード）を設定し、フローモニタに関連付けることができます。従来の双方向レコードはクライアント/サーバアプリケーション統計情報レコードであり、新しい方向性レコードは入出力のアプリケーション統計情報です。

双方向フローレコード

フローレコード 1：双方向フローレコード

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow record <i>flow_record_name</i> 例： Device(config)# flow record fr-wdavic-1	フローレコード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	description <i>description</i> 例： Device(config-flow-record)# description fr-wdavic-1	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例： Device(config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。

	コマンドまたはアクション	目的
ステップ 5	match ipv4 protocol 例 : Device (config-flow-record) # match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match application name 例 : Device (config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 7	match connection client ipv4 address 例 : Device (config-flow-record) # match connection client ipv4 address	クライアント (フローイニシエータ) の IPv4 アドレスとの一致を指定します。
ステップ 8	match connection server ipv4 address 例 : Device (config-flow-record) # match connection server ipv4 address	サーバ (フローレスポンド) の IPv4 アドレスとの一致を指定します。
ステップ 9	match connection server transport port 例 : Device (config-flow-record) # match connection server transport port	サーバのポート番号との一致を指定します。
ステップ 10	match flow observation point 例 : Device (config-flow-record) # match flow observation point	フロー観測メトリックの観測ポイント ID との一致を指定します。
ステップ 11	collect flow direction 例 : Device (config-flow-record) # collect flow direction	次の手順で collect connection initiator コマンドの initiator キーワードで指定される双方向フローの関連する側 (イニシエータまたはレスポンド) の方向 (入力または出力) を収集するように指定します。 initiator キーワードで指定される値に応じて、 flow direction キーワードは次の値をとります。 <ul style="list-style-type: none"> • 0x01 = 入力フロー • 0x02 = 出力フロー

	コマンドまたはアクション	目的
		initiator キーワードがイニシエータに設定されている場合、フローの方向はフローのイニシエータ側から指定されます。 initiator キーワードがレスポンドに設定されている場合、フローの方向はフローのレスポンド側から指定されます。有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。
ステップ 12	collect connection initiator 例 : Device(config-flow-record) # collect connection initiator	collect flow direction コマンドで指定されたフローの方向に関連するフローの側（イニシエータまたはレスポンド）を収集するように指定します。 initiator キーワードは、フローの方向に関する次の情報を提供します。 <ul style="list-style-type: none"> • 0x01 = イニシエータ : フローの送信元は接続のイニシエータです 有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。
ステップ 13	collect connection new-connections 例 : Device(config-flow-record) # collect connection new-connections	観測された接続開始の数を収集するように指定します。
ステップ 14	collect connection client counter packets long 例 : Device(config-flow-record) # collect connection client counter packets long	クライアントが送信したパケット数を収集するように指定します。
ステップ 15	collect connection client counter bytes network long 例 : Device(config-flow-record) # collect connection client counter bytes network long	クライアントが送信したバイト数の合計を収集するように指定します。
ステップ 16	collect connection server counter packets long 例 :	サーバが送信したパケット数を収集するように指定します。

	コマンドまたはアクション	目的
	<code>Device (config-flow-record) # collect connection server counter packets long</code>	
ステップ 17	collect connection server counter bytes network long 例： <code>Device (config-flow-record) # collect connection server counter bytes network long</code>	サーバが送信したバイト数の合計を収集するように指定します。
ステップ 18	collect timestamp absolute first 例： <code>Device (config-flow-record) # collect timestamp absolute first</code>	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 19	collect timestamp absolute last 例： <code>Device (config-flow-record) # collect timestamp absolute last</code>	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 20	end 例： <code>Device (config) # end</code>	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 21	show flow record 例： <code>Device # show flow record</code>	すべてのフローレコードに関する情報を表示します。

フローレコード 2: 双方向フローレコード

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： <code>Device # configure terminal</code>	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	flow record <i>flow_record_name</i> 例： <code>Device (config) # flow record fr-wdavic-1</code>	フローレコード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	description <i>description</i> 例：	(任意) フローレコードの説明を作成します。

	コマンドまたはアクション	目的
	Device(config-flow-record) # description fr-wdavic-1	
ステップ 4	match ipv4 version 例 : Device(config-flow-record) # match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device(config-flow-record) # match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match application name 例 : Device(config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 7	match connection client ipv4 address 例 : Device(config-flow-record) # match connection client ipv4 address	クライアント (フローイニシエータ) の IPv4 アドレスとの一致を指定します。
ステップ 8	match connection client transport port 例 : Device(config-flow-record) # match connection client transport port	(任意) フローレコードのキーフィールドとして、クライアントの接続ポートとの一致を指定します。
ステップ 9	match connection server ipv4 address 例 : Device(config-flow-record) # match connection server ipv4 address	サーバ (フローレスポンド) の IPv4 アドレスとの一致を指定します。
ステップ 10	match connection server transport port 例 : Device(config-flow-record) # match connection server transport port	サーバのトランスポートポートとの一致を指定します。
ステップ 11	match flow observation point 例 : Device(config-flow-record) # match flow observation point	フロー観測メトリックの観測ポイント ID との一致を指定します。

	コマンドまたはアクション	目的
ステップ 12	collect flow direction 例 : Device(config-flow-record)# collect flow direction	次の手順で collect connection initiator コマンドの initiator キーワードで指定される双方向フローの関連する側（イニシエータまたはレスポンド）の方向（入力または出力）を収集するように指定します。 initiator キーワードで指定される値に応じて、 flow direction キーワードは次の値をとります。 <ul style="list-style-type: none"> • 0x01 = 入力フロー • 0x02 = 出力フロー initiator キーワードがイニシエータに設定されている場合、フローの方向はフローのイニシエータ側から指定されます。 initiator キーワードがレスポンドに設定されている場合、フローの方向はフローのレスポンド側から指定されます。有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。
ステップ 13	collect connection initiator 例 : Device(config-flow-record)# collect connection initiator	collect flow direction コマンドで指定されたフローの方向に関連するフローの側（イニシエータまたはレスポンド）を収集するように指定します。 initiator キーワードは、フローの方向に関する次の情報を提供します。 <ul style="list-style-type: none"> • 0x01 = イニシエータ：フローの送信元は接続のイニシエータです 有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。
ステップ 14	collect connection new-connections 例 : Device(config-flow-record)# collect connection new-connections	観測された接続開始の数を収集するように指定します。
ステップ 15	collect connection client counter packets long 例 :	クライアントが送信したパケット数を収集するように指定します。

	コマンドまたはアクション	目的
	<code>Device(config-flow-record)# collect connection client counter packets long</code>	
ステップ 16	collect connection client counter bytes network long 例： <code>Device(config-flow-record)# collect connection client counter bytes network long</code>	クライアントが送信したバイト数の合計を収集するように指定します。
ステップ 17	collect connection server counter packets long 例： <code>Device(config-flow-record)# collect connection server counter packets long</code>	サーバが送信したパケット数を収集するように指定します。
ステップ 18	collect connection server counter bytes network long 例： <code>Device(config-flow-record)# collect connection server counter bytes network long</code>	サーバが送信したバイト数の合計を収集するように指定します。
ステップ 19	collect timestamp absolute first 例： <code>Device(config-flow-record)# collect timestamp absolute first</code>	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 20	collect timestamp absolute last 例： <code>Device(config-flow-record)# collect timestamp absolute last</code>	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 21	end 例： <code>Device(config)# end</code>	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 22	show flow record 例： <code>Device# show flow record</code>	すべてのフローレコードに関する情報を表示します。

方向性フローレコード

フローレコード 3 : 方向性フローレコード : 入力

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow record flow_record_name 例 : Device(config)# flow record fr-wdavic-3	フローレコード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	description description 例 : Device(config-flow-record)# description flow-record-1	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例 : Device(config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device(config-flow-record)# match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match ipv4 source address 例 : Device(config-flow-record)# match ipv4 source address	IPv4 送信元アドレスとの一致をキーフィールドとして指定します。
ステップ 7	match ipv4 destination address 例 : Device(config-flow-record)# match ipv4 destination address	IPv4 宛先アドレスとの一致をキーフィールドとして指定します。
ステップ 8	match transport source-port 例 : Device(config-flow-record)# match transport source-port	トランスポート発信元ポートとの一致をキーフィールドとして指定します。
ステップ 9	match transport destination-port 例 : Device(config-flow-record)# match transport destination-port	トランスポート宛先ポートとの一致をキーフィールドとして指定します。

	コマンドまたはアクション	目的
ステップ 10	match interface input 例 : Device(config-flow-record) # match interface input	入力インターフェイスとの一致をキーフィールドとして指定します。
ステップ 11	match application name 例 : Device(config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 12	collect interface output 例 : Device(config-flow-record) # collect interface output	フローから出力インターフェイスを収集するように指定します。
ステップ 13	collect counter bytes long 例 : Device(config-flow-record) # collect counter bytes long	フローのバイト数を収集するように指定します。
ステップ 14	collect counter packets long 例 : Device(config-flow-record) # collect counter packets long	フローのパケット数を収集するように指定します。
ステップ 15	collect timestamp absolute first 例 : Device(config-flow-record) # collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 16	collect timestamp absolute last 例 : Device(config-flow-record) # collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 17	end 例 : Device(config) # end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 18	show flow record 例 :	すべてのフローレコードに関する情報を表示します。

	コマンドまたはアクション	目的
	Device# show flow record	

フローレコード 4 : 方向性フローレコード : 出力

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow record <i>flow_record_name</i> 例 : Device (config)# flow record fr-wdavic-4	フローレコード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	description <i>description</i> 例 : Device (config-flow-record)# description flow-record-1	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例 : Device (config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device (config-flow-record)# match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match ipv4 source address 例 : Device (config-flow-record)# match ipv4 source address	IPv4 送信元アドレスとの一致をキーフィールドとして指定します。
ステップ 7	match ipv4 destination address 例 : Device (config-flow-record)# match ipv4 destination address	IPv4 宛先アドレスとの一致をキーフィールドとして指定します。
ステップ 8	match transport source-port 例 : Device (config-flow-record)# match transport source-port	トランスポート発信元ポートとの一致をキーフィールドとして指定します。

	コマンドまたはアクション	目的
ステップ 9	match transport destination-port 例 : Device(config-flow-record) # match transport destination-port	トランスポート宛先ポートとの一致をキーフィールドとして指定します。
ステップ 10	match interface output 例 : Device(config-flow-record) # match interface output	出力インターフェイスとの一致をキーフィールドとして指定します。
ステップ 11	match application name 例 : Device(config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 12	collect interface input 例 : Device(config-flow-record) # collect interface input	フローから入力インターフェイスを収集するように指定します。
ステップ 13	collect counter bytes long 例 : Device(config-flow-record) # collect counter bytes long	フローのバイト数を収集するように指定します。
ステップ 14	collect counter packets long 例 : Device(config-flow-record) # collect counter packets long	フローのパケット数を収集するように指定します。
ステップ 15	collect timestamp absolute first 例 : Device(config-flow-record) # collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 16	collect timestamp absolute last 例 : Device(config-flow-record) # collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 17	end 例 :	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコ

	コマンドまたはアクション	目的
	Device (config) # end	ンフィギュレーションモードを終了できます。
ステップ 18	show flow record 例 : Device# show flow record	すべてのフローレコードに関する情報を表示します。

DNS フローレコード

フローレコード 5 : DNS フローレコード

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 2	flow record flow_record_name 例 : Device (config) # flow record fr-wdavic-5	フローレコード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	description description 例 : Device (config-flow-record) # description flow-record-5	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例 : Device (config-flow-record) # match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device (config-flow-record) # match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match application name 例 : Device (config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。

	コマンドまたはアクション	目的
ステップ 7	match connection client ipv4 address 例: Device(config-flow-record)# match connection client ipv4 address	クライアント (フローイニシエータ) の IPv4 アドレスとの一致を指定します。
ステップ 8	match connection client transport port 例: Device(config-flow-record)# match connection client transport port	フローレコードのキーフィールドとして、クライアントの接続ポートとの一致を指定します。
ステップ 9	match connection server ipv4 address 例: Device(config-flow-record)# match connection server ipv4 address	サーバ (フローレスポンド) の IPv4 アドレスとの一致を指定します。
ステップ 10	match connection server transport port 例: Device(config-flow-record)# match connection server transport port	サーバのポート番号との一致を指定します。
ステップ 11	collect flow direction 例: Device(config-flow-record)# collect flow direction	<p>次の手順で collect connection initiator コマンドの initiator キーワードで指定される双方向フローの関連する側 (イニシエータまたはレスポンド) の方向 (入力または出力) を収集するように指定します。 initiator キーワードで指定される値に応じて、flow direction キーワードは次の値をとります。</p> <ul style="list-style-type: none"> • 0x01 = 入力フロー • 0x02 = 出力フロー <p>initiator キーワードがイニシエータに設定されている場合、フローの方向はフローのイニシエータ側から指定されます。 initiator キーワードがレスポンドに設定されている場合、フローの方向はフローのレスポンド側から指定されます。有線 AVC では、initiator キーワードは常にイニシエータに設定されています。</p>

	コマンドまたはアクション	目的
ステップ 12	collect timestamp absolute first 例 : Device(config-flow-record)# collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 13	collect timestamp absolute last 例 : Device(config-flow-record)# collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 14	collect connection initiator 例 : Device(config-flow-record)# collect connection initiator	collect flow direction コマンドで指定されたフローの方向に関連するフローの側（イニシエータまたはレスポンド）を収集するように指定します。 initiator キーワードは、フローの方向に関する次の情報を提供します。 • 0x01 = イニシエータ : フローの送信元は接続のイニシエータです 有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。
ステップ 15	collect connection new-connections 例 : Device(config-flow-record)# collect connection new-connections	観測された接続開始の数を収集するように指定します。
ステップ 16	collect connection server counter packets long 例 : Device(config-flow-record)# collect connection server counter packets long	サーバが送信したパケット数を収集するように指定します。
ステップ 17	collect connection client counter packets long 例 : Device(config-flow-record)# collect connection client counter packets long	クライアントが送信したパケット数を収集するように指定します。
ステップ 18	collect connection server counter bytes network long 例 :	サーバが送信したバイト数の合計を収集するように指定します。

	コマンドまたはアクション	目的
	Device(config-flow-record)# collect connection server counter bytes network long	
ステップ 19	collect connection client counter bytes network long 例： Device(config-flow-record)# collect connection client counter bytes network long	クライアントが送信したバイト数の合計を収集するように指定します。
ステップ 20	collect application dns domain-name 例： Device(config-flow-record)# collect application dns domain-name	DNS ドメイン名を DNS フローレコードの収集フィールドとして使用するよう設定します。
ステップ 21	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーションモードを終了できます。

フロー エクスポートの作成

フロー エクスポートを作成すると、フローのエクスポート パラメータを定義できます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow exporter flow_exporter_name 例： Device(config)# flow exporter flow-exporter-1	フロー エクスポート 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	description description 例： Device(config-flow-exporter)# description flow-exporter-1	(任意) フロー エクスポートの説明を作成します。
ステップ 4	destination { hostname ipv4-address ipv6-address } 例：	エクスポートでデータを送信する宛先システムのホスト名、IPv4 または IPv6 アドレスを指定します。

	コマンドまたはアクション	目的
	Device(config-flow-exporter)# destination 10.10.1.1	
ステップ 5	option application-table [timeout seconds] 例： Device(config-flow-exporter)# option application-table timeout 500	(任意) フロー エクスポートのアプリケーション テーブルのオプションを設定します。 timeout オプションを使用すると、フローエクスポートの再送信時間を秒単位で設定できます。有効な範囲は 1 ~ 86400 秒です。
ステップ 6	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。
ステップ 7	show flow exporter 例： Device# show flow exporter	すべてのフロー エクスポートに関する情報を表示します。
ステップ 8	show flow exporter statistics 例： Device# show flow exporter statistics	フロー エクスポートの統計情報を表示します。

フロー モニタの作成

フロー モニタを作成して、フロー レコードに関連付けることができます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow monitor monitor-name 例： Device(config)# flow monitor flow-monitor-1	フロー モニタを作成し、フロー モニタ 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	description description 例： Device(config-flow-monitor)# description flow-monitor-1	(任意) フロー モニタの説明を作成します。

	コマンドまたはアクション	目的
ステップ 4	record <i>record-name</i> 例： Device(config-flow-monitor)# record flow-record-1	事前に作成されたレコードの名前を指定します。
ステップ 5	exporter <i>exporter-name</i> 例： Device(config-flow-monitor)# exporter flow-exporter-1	事前に作成されたエクスポートの名前を指定します。
ステップ 6	cache { entries <i>number-of-entries</i> timeout { active inactive } type normal } 例： Device(config-flow-monitor)# cache timeout active 1800 例： Device(config-flow-monitor)# cache timeout inactive 200 例： Device(config-flow-monitor)# cache type normal	(任意) フローキャッシュパラメータを設定するように指定します。 • entries <i>number-of-entries</i> : フローキャッシュ内のフローエントリの最大数を 16 ~ 65536 の範囲で指定します。 (注) 標準のキャッシュタイプのみがサポートされます。
ステップ 7	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 8	show flow monitor 例： Device# show flow monitor	すべてのフローモニタに関する情報を表示します。
ステップ 9	show flow monitor <i>flow-monitor-name</i> 例： Device# show flow monitor flow-monitor-1	指定した有線 AVC フロー モニタに関する情報を表示します。
ステップ 10	show flow monitor <i>flow-monitor-name</i> statistics 例： Device# show flow monitor flow-monitor-1 statistics	有線 AVC フロー モニタの統計情報を表示します。

	コマンドまたはアクション	目的
ステップ 11	clear flow monitor <i>flow-monitor-name</i> statistics 例 : <pre>Device# clear flow monitor flow-monitor-1 statistics</pre>	指定したフローモニタの統計情報をクリアします。 clear flow monitor flow-monitor-1 statistics を使用した後に show flow monitor flow-monitor-1 statistics コマンドを使用して、すべての統計情報がリセットされたことを確認します。
ステップ 12	show flow monitor <i>flow-monitor-name</i> cache format table 例 : <pre>Device# show flow monitor flow-monitor-1 cache format table</pre>	表形式でフローキャッシュの内容を表示します。
ステップ 13	show flow monitor <i>flow-monitor-name</i> cache format record 例 : <pre>Device# show flow monitor flow-monitor-1 cache format record</pre>	フローレコードと同様の形式でフローキャッシュの内容を表示します。
ステップ 14	show flow monitor <i>flow-monitor-name</i> cache format csv 例 : <pre>Device# show flow monitor flow-monitor-1 cache format csv</pre>	CSV形式でフローキャッシュの内容を表示します。

インターフェイスへのフロー モニタの関連付け

異なる事前定義済みレコードを持つ 2 つの異なる有線 AVC モニタをインターフェイスに同時に接続できます。

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>Device# configure terminal</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	interface <i>interface-id</i> 例 : <pre>Device(config)# interface Gigabitethernet 1/0/1</pre>	インターフェイスコンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	ip flow monitor <i>monitor-name</i> { input output } 例 : Device(config-if) # ip flow monitor flow-monitor-1 input	入力パケットと出力パケットの両方またはいずれか用のインターフェイスにフロー モニタを関連付けます。
ステップ 4	end 例 : Device(config) # end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

NBAR2 カスタム アプリケーション

NBAR2 では、カスタム プロトコルを使用してカスタム アプリケーションを識別できます。カスタム プロトコルは、プロトコルとアプリケーションをサポートしますが、現在のところ、NBAR2 はサポートしていません。

すべての展開において、シスコが提供する NBAR2 プロトコルパックの対象外であるローカル アプリケーションおよび特定のアプリケーションがあります。ローカル アプリケーションは主に次のように分類されます。

- 組織への特定のアプリケーション
- 地域特有のアプリケーション

NBAR2 では、このようなローカル アプリケーションを手動でカスタマイズする方法を提供しています。グローバル コンフィギュレーション モードで **ip nbar custom myappname** コマンドを使用して、手動でアプリケーションをカスタマイズできます。カスタム アプリケーションは、組み込みプロトコルより優先されます。それぞれのカスタム プロトコルでは、ユーザは、レポート目的に使用できるセレクト ID を定義できます。

さまざまなタイプのアプリケーション カスタマイズがあります。

一般的なプロトコルのカスタマイズ

- HTTP
- SSL
- DNS

コンポジット：複数の基本的なプロトコルに基づくカスタマイズ：**server-name**

レイヤ 3/レイヤ 4 のカスタマイズ

- IPv4 アドレス
- DSCP 値
- TCP/UDP ポート

- フロー送信元または宛先の方向

バイト オフセット：ペイロードの特定のバイト値に基づくカスタマイズ

HTTP のカスタマイズ

HTTP のカスタマイズは、次の HTTP フィールドの組み合わせに基づいて実行できます。

- **cookie** : HTTP クッキー
- **host** : リソースを含む元のサーバのホスト名
- **method** : HTTP メソッド
- **referrer** : リソース リクエストの取得元のアドレス
- **url** : Uniform Resource Locator のパス
- **user-agent** : 要求を送信するエージェントによって使用されているソフトウェア
- **version** : HTTP バージョン
- **via** : HTTP 経由フィールド

HTTP のカスタマイズ

セレクト ID 10 が付いた HTTP ホスト 「*mydomain.com」 を使用する MYHTTP と呼ばれるカスタム アプリケーション。

```
Device# configure terminal
Device(config)# ip nbar custom MYHTTP http host *mydomain.com id 10
```

SSL のカスタマイズ

SSL サーバ名指定 (SNI) または共通名 (CN) から抽出した情報を使用して、SSL 暗号化トラフィックでカスタマイズを行うことができます。

SSL のカスタマイズ

セレクト ID 11 が付いた SSL 固有名 「mydomain.com」 を使用する MYSSL と呼ばれるカスタム アプリケーション。

```
Device# configure terminal
Device(config)# ip nbar custom MYSSL ssl unique-name *mydomain.com id 11
```

DNS のカスタマイズ

NBAR2 は、DNS 要求および応答トラフィックを確認し、アプリケーションへの DNS 応答に関連付けることができます。DNS 応答から戻された IP アドレスはキャッシュされ、その特定のアプリケーションに関連付けられているその後のパケット フローに使用されます。

ip nbar custom application-name dns domain-name id application-id コマンドは、DNS のカスタマイズに使用されます。既存のアプリケーションを拡張するには、**ip nbar custom application-name dns domain-name domain-name extends existing-application** コマンドを使用します。

DNS ベースのカスタマイズの詳細については、http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/configuration/xr-3s/asr1000/qos-nbar-xr-3s-asr-1000-book/nbar-custapp-dns-xr.html を参照してください。

DNS のカスタマイズ

セレクタ ID 12 が付いた DNS ドメイン名「mydomain.com」を使用する MYDNS と呼ばれるカスタム アプリケーション。

```
Device# configure terminal
Device(config)# ip nbar custom MYDNS dns domain-name *mydomain.com id 12
```

複合カスタマイズ

NBAR2 では、HTTP、SSL または DNS に現れるドメイン名に基づいてアプリケーションをカスタマイズする方法が提供されます。

複合カスタマイズ

セレクタ ID 13 が付いた HTTP、SSL または DNS ドメイン名「mydomain.com」を使用する MYDOMAIN と呼ばれるカスタム アプリケーション。

```
Device# configure terminal
Device(config)# ip nbar custom MYDOMAIN composite server-name *mydomain.com id 13
```

L3/L4 のカスタマイズ

レイヤ3/レイヤ4のカスタマイズは、パケットタプルに基づいており、フローの最初のパケットで常に一致します。

L3/L4 のカスタマイズ

IP アドレス 10.56.1.10 および 10.56.1.11、セレクタ ID 14 が付いた TCP および DSCP ef に一致する LAYER4CUSTOM と呼ばれるカスタム アプリケーション。

```
Device# configure terminal
Device(config)# ip nbar custom LAYER4CUSTOM transport tcp id 14
Device(config-custom)# ip address 10.56.1.10 10.56.1.11
Device(config-custom)# dscp ef
```

例：カスタム アプリケーションのモニタリング

カスタム アプリケーションのモニタリングのための **show** コマンド
show ip nbar protocol-id | inc Custom

```
Device# show ip nbar protocol-id | inc Custom
LAYER4CUSTOM          14          Custom
MYDNS                  12          Custom
MYDOMAIN              13          Custom
MYHTTP                10          Custom
MYSSL                 11          Custom
```

show ip nbar protocol-discovery protocol CUSTOM_APP

```
Device# show ip nbar protocol-id MYSSL
Protocol Name          id          type
-----
MYSSL                  11          Custom
```

NBAR2 ダイナミック ヒットレス プロトコル パックのアップグレード

プロトコルパックは、デバイスのシスコソフトウェアを置き換えることなく、デバイスの NBAR2 プロトコル サポートを更新するソフトウェア パッケージです。プロトコルパックには、NBAR2 によって正式にサポートされている、コンパイル済みでパック済みのアプリケーションに関する情報が含まれています。各アプリケーションについて、プロトコルパックには、アプリケーション署名とアプリケーション属性の情報が含まれています。各ソフトウェア リリースには、組み込みのプロトコルパックがバンドルされています。

プロトコルパックには次の特長があります。

- ロードが容易で高速。
- 高いバージョンのプロトコルパックにアップグレードしたり、低いバージョンのプロトコルパックに戻したりするのが容易。
- スイッチのリロードを必要としない。

NBAR2 プロトコルパックは、次の URL から Cisco Software Center でダウンロードできます：
<https://software.cisco.com/download/home>

NBAR2 プロトコルパックの前提条件

新しいプロトコルパックをロードする前に、すべてのスイッチ メンバー上でプロトコルパックをフラッシュにコピーする必要があります。

プロトコルパックをロードするには、[NBAR2 プロトコルパックのロード \(141 ページ\)](#) を参照してください。

NBAR2 プロトコルパックのロード

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。

例 : NBAR2 プロトコルパックのロード

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip nbar protocol-pack protocol-pack [force] 例 : Device(config)# ip nbar protocol-pack flash:defProtoPack 例 : Device(config)# default ip nbar protocol-pack	プロトコルパックをロードします。 <ul style="list-style-type: none"> • 基本のプロトコルパックバージョンとは異なる、より低いバージョンのプロトコルパックを指定し、ロードするには、force キーワードを使用します。これにより、スイッチの現在のプロトコルパックでサポートされていない設定も削除されます。 組み込みのプロトコルパックに戻るには、次のコマンドを使用します。
ステップ 4	exit 例 : Device(config)# exit	特権 EXEC モードに戻ります。
ステップ 5	show ip nbar protocol-pack {protocol-pack active} [detail] 例 : Device# show ip nbar protocol-pack active	プロトコルパック情報を表示します。 <ul style="list-style-type: none"> • このコマンドを使用して、ロードされたプロトコルパックのバージョン、パブリッシャ、その他の詳細を確認します。 • 指定されたプロトコルパックの情報を表示するには、<i>protocol-pack</i> 引数を使用します。 • アクティブなプロトコルパックの情報を表示するには、active キーワードを使用します。 • 詳細なプロトコルパックの情報を表示するには、detail キーワードを使用します。

例 : NBAR2 プロトコルパックのロード

次の例に、新しいプロトコルパックをロードする方法を示します。

```
Device> enable
Device# configure terminal
```

```
Device(config)# ip nbar protocol-pack flash:newDefProtoPack
Device(config)# exit
```

次の例に、**force** キーワードを使用して下位バージョンのプロトコルパックをロードする方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip nbar protocol-pack flash:OldDefProtoPack force
Device(config)# exit
```

次の例に、組み込みのプロトコルパックに戻す方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# default ip nbar protocol-pack
Device(config)# exit
```

Application Visibility and Control のモニタリング

このセクションでは、アプリケーションの可視性に関する新しいコマンドについて説明します。

次のコマンドは、スイッチおよびアクセスポートのアプリケーションの可視性をモニタするために使用できます。

表 7: スイッチのアプリケーションの可視性モニタリングコマンド

コマンド	目的
<pre>show ip nbar protocol-discovery [interface interface-type interface-number] [stats{byte-count bit-rate packet-count max-bit-rate}] [protocol protocol-name top-n number]</pre>	<p>NBAR Protocol Discovery 機能によって収集された統計情報を表示します。</p> <ul style="list-style-type: none"> (任意) 表示される統計情報を最適化するには、キーワードおよび引数を入力します。キーワードのそれぞれの詳細については、『Cisco IOS Quality of Service Solutions Command Reference』の show ip nbar protocol-discovery コマンドを参照してください。
<pre>show policy-map interface interface-type interface-number</pre>	<p>インターフェイスに適用したポリシー マップについての情報を表示します。</p>

例 : Application Visibility and Control の設定

次に、match protocol でアプリケーション名のフィルタを適用してクラス マップを作成する例を示します。

```
Device# configure terminal
Device(config)# class-map match-any NBAR-VOICE
```

```
Device(config-cmap)# match protocol ms-lync-audio
Device(config-cmap)#end
```

次に、ポリシー マップを作成し、出力 QoS の既存のクラス マップを定義する例を示します。

```
Device# configure terminal
Device(config)# policy-map test-avc-up
Device(config-pmap)# class cat-browsing
Device(config-pmap-c)# police 150000
Device(config-pmap-c)# set dscp 12
Device(config-pmap-c)#end
```

次に、ポリシー マップを作成し、入力 QoS の既存のクラス マップを定義する例を示します。

```
Device# configure terminal
Device(config)# policy-map test-avc-down
Device(config-pmap)# class cat-browsing
Device(config-pmap-c)# police 200000
Device(config-pmap-c)# set dscp 10
Device(config-pmap-c)#end
```

次に、ポリシー マップをスイッチ ポートに適用する例を示します。

```
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# switchport mode access
Device(config-if)# switchport access vlan 20
Device(config-if)# service-policy input POLICING_IN
Device(config-if)#end
```

次に、NBAR 属性に基づいてクラスマップを作成する例を示します。

```
Device# configure terminal
Device(config)# class-map match-all rel-relevant
Device(config-cmap)# match protocol attribute business-relevance business-relevant

Device(config)# class-map match-all rel-irrelevant
Device(config-cmap)# match protocol attribute business-relevance business-irrelevant

Device(config)# class-map match-all rel-default
Device(config-cmap)# match protocol attribute business-relevance default

Device(config)# class-map match-all class--ops-admin-and-rel
Device(config-cmap)# match protocol attribute traffic-class ops-admin-mgmt
Device(config-cmap)# match protocol attribute business-relevance business-relevant
```

次に、NBAR 属性に基づくクラスマップに基づいてポリシーマップを作成する例を示します。

```
Device# configure terminal
Device(config)# policy-map attrib--rel-types
Device(config-pmap)# class rel-relevant
Device(config-pmap-c)# set dscp ef
Device(config-pmap-c)# class rel-irrelevant
Device(config-pmap-c)# set dscp af11
Device(config-pmap-c)# class rel-default
Device(config-pmap-c)# set dscp default

Device(config)# policy-map attrib--ops-admin-and-rel
Device(config-pmap)# class class--ops-admin-and-rel
Device(config-pmap-c)# set dscp cs5
```

次に、NBAR 属性に基づくポリシーマップを有線ポートに適用する例を示します。


```
Device# configure terminal
Device(config)# interface GigabitEthernet1/0/2
Device(config-if)# service-policy input attrib--rel-types
```

show コマンドによる設定の表示

show ip nbar protocol-discovery

インターフェイスごとのプロトコル検出統計情報のレポートを表示します。

次に、インターフェイスごとの統計情報の出力例を示します。

```
Device# show ip nbar protocol-discovery int GigabitEthernet1/0/1

GigabitEthernet1/0/1
Last clearing of "show ip nbar protocol-discovery" counters 00:03:16

Output
-----
Protocol
Packet Count
Byte Count
30sec Bit Rate (bps)
30sec Max Bit Rate (bps)
-----
Input
-----
Packet Count
Byte Count
30sec Bit Rate (bps)
30sec Max Bit Rate (bps)
-----
ms-lync
55911
28774864
93000
3437000
Total
55911
28774864
93000
3437000
60580
31174777
3613000
3613000
60580
31174777
3613000
3613000
```

show policy-map interface

すべてのインターフェイス上の QoS 統計情報および設定済みのポリシーマップを表示します。

次に、すべてのインターフェイスに設定されたポリシーマップの出力例を示します。

```

Device# show policy-map int

GigabitEthernet1/0/1
  Service-policy input: MARKING-IN

    Class-map: NBAR-VOICE (match-any)
      718 packets
      Match: protocol ms-lync-audio
        0 packets, 0 bytes
        30 second rate 0 bps
      QoS Set
        dscp ef

    Class-map: NBAR-MM_CONFERENCING (match-any)
      6451 packets
      Match: protocol ms-lync
        0 packets, 0 bytes
        30 second rate 0 bps
      Match: protocol ms-lync-video
        0 packets, 0 bytes
        30 second rate 0 bps
      QoS Set
        dscp af41

    Class-map: class-default (match-any)
      34 packets
      Match: any

```

show コマンドによる属性ベースの QoS 設定の表示

show policy-map interface

すべてのインターフェイス上の属性ベースの QoS 統計情報および設定済みのポリシーマップを表示します。

次に、すべてのインターフェイスに設定されたポリシーマップの出力例を示します。

```

Device# show policy-map interface gigabitEthernet 1/0/2
GigabitEthernet1/0/2

  Service-policy input: attrib--rel-types

    Class-map: rel-relevant (match-all)
      20 packets
      Match: protocol attribute business-relevance business-relevant
      QoS Set
        dscp ef

    Class-map: rel-irrelevant (match-all)
      0 packets
      Match: protocol attribute business-relevance business-irrelevant

      QoS Set

```

```
dscp af11

Class-map: rel-default (match-all)
  14 packets
  Match: protocol attribute business-relevance default
  QoS Set
  dscp default

Class-map: class-default (match-any)
  0 packets
  Match: any
```

show ip nbar protocol-attribute

NBAR で使用されるすべてのプロトコル属性を表示します。

次に、一部の属性の出力例を示します。

```
Device# show ip nbar protocol-attribute cisco-jabber-im
  Protocol Name : cisco-jabber-im
    encrypted : encrypted-yes
    tunnel : tunnel-no
    category : voice-and-video
    sub-category : enterprise-media-conferencing
  application-group : cisco-jabber-group
  p2p-technology : p2p-tech-no
  traffic-class : transactional-data
  business-relevance : business-relevant
  application-set : collaboration-apps

Device# show ip nbar protocol-attribute google-services
  Protocol Name : google-services
    encrypted : encrypted-yes
    tunnel : tunnel-no
    category : other
    sub-category : other
  application-group : google-group
  p2p-technology : p2p-tech-yes
  traffic-class : transactional-data
  business-relevance : default
  application-set : general-browsing

Device# show ip nbar protocol-attribute dns
  Protocol Name : google-services
    encrypted : encrypted-yes
    tunnel : tunnel-no
    category : other
    sub-category : other
  application-group : google-group
  p2p-technology : p2p-tech-yes
  traffic-class : transactional-data
  business-relevance : default
  application-set : general-browsing
```

```
Device# show ip nbar protocol-attribute unknown
      Protocol Name : unknown
      encrypted : encrypted-no
      tunnel : tunnel-no
      category : other
      sub-category : other
      application-group : other
      p2p-technology : p2p-tech-no
      traffic-class : bulk-data
      business-relevance : default
      application-set : general-misc
```

show コマンドによるフロー モニタ設定の表示

show flow monitor wdavc

指定した有線 AVC フロー モニタに関する情報を表示します。

```
Device # show flow monitor wdavc

Flow Monitor wdavc:
  Description:      User defined
  Flow Record:     wdavc
  Flow Exporter:   wdavc-exp (inactive)
  Cache:
    Type:          normal (Platform cache)
    Status:        not allocated
    Size:          12000 entries
    Inactive Timeout: 15 secs
    Active Timeout: 1800 secs
```

show flow monitor wdavc statistics

有線 AVC フロー モニタの統計情報を表示します。

```
Device# show flow monitor wdavc statistics
Cache type:          Normal (Platform cache)
Cache size:          12000
Current entries:     13

Flows added:         26
Flows aged:          13
  - Active timeout   ( 1800 secs)  1
  - Inactive timeout (   15 secs)  12
```

clear flow monitor wdavc statistics

指定したフロー モニタの統計情報をクリアします。**clear flow monitor wdavc statistics** を使用した後に **show flow monitor wdavc statistics** コマンドを使用して、すべての統計情報がリセットされたことを確認します。以下に、フローモニタ統計情報をクリアした後の **show flow monitor wdavc statistics** コマンドのサンプル出力を示します。

```
Device# show flow monitor wdavc statistics
Cache type:          Normal (Platform cache)
Cache size:          12000
Current entries:     0
```

```

Flows added:                0
Flows aged:                  0

```

show コマンドによるキャッシュの内容の表示

show flow monitor wdvac cache format table

表形式でフロー キャッシュの内容を表示します。

```

Device# show flow monitor wdvac cache format table
Cache type:                Normal (Platform cache)
Cache size:                 12000
Current entries:           13

Flows added:                26
Flows aged:                 13
- Active timeout           ( 1800 secs)    1
- Inactive timeout         (   15 secs)    12

CONN IPV4 INITIATOR ADDR  CONN IPV4 RESPONDER ADDR  CONN RESPONDER PORT
FLOW OBSPOINT ID  IP VERSION  IP PROT  APP NAME
flow dirn .....
-----
-----
64.103.125.147          144.254.71.184
53          4294967305          4          17  port dns
  Input .....
64.103.121.103          10.1.1.2
67          4294967305          4          17  layer7 dhcp
  Input ....contd.....
64.103.125.3           64.103.125.97
68          4294967305          4          17  layer7 dhcp
  Input .....
10.0.2.6               157.55.40.149
          4294967305          4          6   layer7 ms-lync
  Input .....
64.103.126.28          66.163.36.139
          4294967305          4          6   layer7 cisco-jabber-im
  Input ....contd.....
64.103.125.2           64.103.125.29
68          4294967305          4          17  layer7 dhcp
  Input .....
64.103.125.97          64.103.101.181
67          4294967305          4          17  layer7 dhcp
  Input .....
192.168.100.6          10.10.20.1
          4294967305          4          17  layer7 cisco-jabber-control
  Input ....contd.....
64.103.125.3           64.103.125.29
68          4294967305          4          17  layer7 dhcp
  Input .....

```

```

10.80.101.18          10.80.101.6          5060
      4294967305          4          6 layer7 cisco-collab-control
Input .....
10.1.11.4            66.102.11.99
80      4294967305          4          6 layer7 google-services
      Input ....contd.....
64.103.125.2        64.103.125.97
68      4294967305          4          17 layer7 dhcp
      Input .....
64.103.125.29       64.103.101.181
67      4294967305          4          17 layer7 dhcp
      Input .....

```

show flow monitor wdacv cache format record

フローレコードと同様の形式でフローキャッシュの内容を表示します。

```

Device# show flow monitor wdacv cache format record
Cache type: Normal (Platform cache)
Cache size: 12000
Current entries: 13

Flows added: 26
Flows aged: 13
- Active timeout ( 1800 secs) 1
- Inactive timeout ( 15 secs) 12

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.147
CONNECTION IPV4 RESPONDER ADDRESS: 144.254.71.184
CONNECTION RESPONDER PORT: 53
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: port dns
flow direction: Input
timestamp abs first: 08:55:46.917
timestamp abs last: 08:55:46.917
connection initiator: Initiator
connection count new: 2
connection serverpackets counter: 1
connection client packets counter: 1
connection server network bytes counter: 190
connection client network bytes counter: 106

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.121.103
CONNECTION IPV4 RESPONDER ADDRESS: 10.1.1.2
CONNECTION RESPONDER PORT: 67
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917

```

```
timestamp abs last:                08:55:47.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 1
connection server network bytes counter: 0
connection client network bytes counter: 350

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.3
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.97
CONNECTION RESPONDER PORT:        68
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                  layer7 dhcp
flow direction:                    Input
timestamp abs first:               08:55:47.917
timestamp abs last:               08:55:53.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 4
connection server network bytes counter: 0
connection client network bytes counter: 1412

CONNECTION IPV4 INITIATOR ADDRESS: 10.0.2.6
CONNECTION IPV4 RESPONDER ADDRESS: 157.55.40.149
CONNECTION RESPONDER PORT:        443
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       6
APPLICATION NAME:                  layer7 ms-lync
flow direction:                    Input
timestamp abs first:               08:55:46.917
timestamp abs last:               08:55:46.917
connection initiator:              Initiator
connection count new:              2
connection server packets counter: 10
connection client packets counter: 14
connection server network bytes counter: 6490
connection client network bytes counter: 1639

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.126.28
CONNECTION IPV4 RESPONDER ADDRESS: 66.163.36.139
CONNECTION RESPONDER PORT:        443
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       6
APPLICATION NAME:                  layer7 cisco-jabber-im
flow direction:                    Input
timestamp abs first:               08:55:46.917
```

```
timestamp abs last:                08:55:46.917
connection initiator:              Initiator
connection count new:              2
connection server packets counter: 12
connection client packets counter: 10
connection server network bytes counter: 5871
connection client network bytes counter: 2088

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.2
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.29
CONNECTION RESPONDER PORT:        68
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                   layer7 dhcp
flow direction:                     Input
timestamp abs first:                08:55:47.917
timestamp abs last:                08:55:47.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 2
connection server network bytes counter: 0
connection client network bytes counter: 712

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.97
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.101.181
CONNECTION RESPONDER PORT:        67
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                   layer7 dhcp
flow direction:                     Input
timestamp abs first:                08:55:47.917
timestamp abs last:                08:55:47.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 1
connection server network bytes counter: 0
connection client network bytes counter: 350

CONNECTION IPV4 INITIATOR ADDRESS: 192.168.100.6
CONNECTION IPV4 RESPONDER ADDRESS: 10.10.20.1
CONNECTION RESPONDER PORT:        5060
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                   layer7 cisco-jabber-control
flow direction:                     Input
timestamp abs first:                08:55:46.917
```



```
timestamp abs last:                08:55:46.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 2
connection server network bytes counter: 0
connection client network bytes counter: 2046

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.3
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.29
CONNECTION RESPONDER PORT:        68
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                   layer7 dhcp
flow direction:                     Input
timestamp abs first:                08:55:47.917
timestamp abs last:                08:55:47.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 2
connection server network bytes counter: 0
connection client network bytes counter: 712

CONNECTION IPV4 INITIATOR ADDRESS: 10.80.101.18
CONNECTION IPV4 RESPONDER ADDRESS: 10.80.101.6
CONNECTION RESPONDER PORT:        5060
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       6
APPLICATION NAME:                   layer7 cisco-collab-control
flow direction:                     Input
timestamp abs first:                08:55:46.917
timestamp abs last:                08:55:47.917
connection initiator:              Initiator
connection count new:              2
connection server packets counter: 23
connection client packets counter: 27
connection server network bytes counter: 12752
connection client network bytes counter: 8773

CONNECTION IPV4 INITIATOR ADDRESS: 10.1.11.4
CONNECTION IPV4 RESPONDER ADDRESS: 66.102.11.99
CONNECTION RESPONDER PORT:        80
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       6
APPLICATION NAME:                   layer7 google-services
flow direction:                     Input
timestamp abs first:                08:55:46.917
```

```

timestamp abs last:                08:55:46.917
connection initiator:              Initiator
connection count new:              2
connection server packets counter: 3
connection client packets counter: 5
connection server network bytes counter: 1733
connection client network bytes counter: 663

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.2
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.97
CONNECTION RESPONDER PORT:        68
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                  layer7 dhcp
flow direction:                    Input
timestamp abs first:               08:55:47.917
timestamp abs last:               08:55:53.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 4
connection server network bytes counter: 0
connection client network bytes counter: 1412

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.29
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.101.181
CONNECTION RESPONDER PORT:        67
FLOW OBSPOINT ID:                 4294967305
IP VERSION:                        4
IP PROTOCOL:                       17
APPLICATION NAME:                  layer7 dhcp
flow direction:                    Input
timestamp abs first:               08:55:47.917
timestamp abs last:               08:55:47.917
connection initiator:              Initiator
connection count new:              1
connection server packets counter: 0
connection client packets counter: 1
connection server network bytes counter: 0
connection client network bytes counter: 350

```

show flow monitor wdavc cache format csv

CSV 形式でフロー キャッシュの内容を表示します。

```

Device# show flow monitor wdavc cache format csv
Cache type:                Normal (Platform cache)
Cache size:                 12000
Current entries:           13

Flows added:                26
Flows aged:                 13

```

```
- Active timeout      ( 1800 secs)      1
- Inactive timeout   (   15 secs)     12
```

```
CONN IPV4 INITIATOR ADDR,CONN IPV4 RESPONDER ADDR,CONN RESPONDER
PORT,FLOW OBSPOINT ID,IP VERSION,IP
PROT,APP NAME,flow dirn,time abs first,time abs last,conn initiator,conn
count new,conn server packets
cnt,conn client packets cnt,conn server network bytes cnt,conn client
network bytes cnt
64.103.125.147,144.254.71.184,53,4294967305,4,17,port
dns,Input,08:55:46.917,08:55:46.917,Initiator,2,1,1,190,106
64.103.121.103,10.1.1.2,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
64.103.125.3,64.103.125.97,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:53.917,Initiator,1,0,4,0,1412
10.0.2.6,157.55.40.149,443,4294967305,4,6,layer7 ms-
lync,Input,08:55:46.917,08:55:46.917,Initiator,2,10,14,6490,1639
64.103.126.28,66.163.36.139,443,4294967305,4,6,layer7 cisco-jabber-
im,Input,08:55:46.917,08:55:46.917,Initiator,2,12,10,5871,2088
64.103.125.2,64.103.125.29,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,2,0,712
64.103.125.97,64.103.101.181,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
192.168.100.6,10.10.20.1,5060,4294967305,4,17,layer7 cisco-jabber-
control,Input,08:55:46.917,08:55:46.917,Initiator,1,0,2,0,2046
64.103.125.3,64.103.125.29,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,2,0,712
10.80.101.18,10.80.101.6,5060,4294967305,4,6,layer7 cisco-collab-
control,Input,08:55:46.917,08:55:47.917,Initiator,2,23,27,12752,8773
10.1.11.4,66.102.11.99,80,4294967305,4,6,layer7 google-
services,Input,08:55:46.917,08:55:46.917,Initiator,2,3,5,1733,663
64.103.125.2,64.103.125.97,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:53.917,Initiator,1,0,4,0,1412
64.103.125.29,64.103.101.181,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
```

基本的なトラブルシューティング：質問と回答

以下に、有線 Application Visibility and Control のトラブルシューティングに関する基本的な質問と回答を示します。

- 質問：** IPv6 トラフィックが分類されていません。

回答： 現在は IPv4 トラフィックのみがサポートされています。
- 質問：** マルチキャスト トラフィックが分類されていません。

回答： 現在はユニキャスト トラフィックのみがサポートされています。
- 質問：** ping を送信したときに、分類されているかを確認できません。

回答：TCP/UDP プロトコルのみがサポートされています。

4. 質問：SVI に NBAR を接続できないのはなぜですか。

回答：NBAR は物理インターフェイスでのみサポートされています。

5. 質問：ほとんどのトラフィックが CAPWAP トラフィックになっているのですが、なぜですか。

回答：ワイヤレス アクセス ポートに接続されていないアクセス ポートで NBAR が有効になっていることを確認してください。AP から着信するすべてのトラフィックは capwap として分類されます。この場合、実際の分類は AP または WLC で行われます。

6. 質問：プロトコル検出で、トラフィックが片側でしか確認できません。さらに、多くの未知のトラフィックがあります。

回答：これは通常、NBAR が非対称トラフィックを確認していることを示します。片側のトラフィックは1つのスイッチメンバーに分類され、もう一方は別のメンバーに分類されます。トラフィックの両側が確認されるアクセスポートにのみ NBAR を接続することを推奨します。複数のアップリンクがある場合は、この問題のためそれらに NBAR を接続することはできません。ポートチャネルの一部であるインターフェイスに NBAR を設定した場合にも同様の問題が発生します。

7. 質問：プロトコル検出で、すべてのアプリケーションの集約ビューが表示されます。時間経過に伴うトラフィック分布を確認するにはどうしたらいいですか。

回答：WebUI を使用して、過去 48 時間の経時的なトラフィックを表示できます。

8. 質問：**match protocol protocol-name** コマンドを使用してキューベースのイーグレスポリシーを設定できません。

回答：NBAR2 ベースの分類子が含まれるポリシーでは、**shape** および **set DSCP** のみがサポートされています。一般的な方法としては、入力で DSCP を設定し、DSCP に基づいて出力でシェーピングを実行します。

9. 質問：インターフェイスに接続している NBAR2 はありませんが、NBAR2 がいまだにアクティブになっています。

回答：**match protocol protocol-name** を含むクラスマップがあると、NBAR はスタックでグローバルにアクティブになりますが、トラフィックは NBAR 分類の対象にはなりません。これは予期された動作であり、リソースを消費しません。

10. 質問：デフォルトの QOS キューの下にトラフィックがあります。どうしてですか。

回答：新しい各フローでは、フローを分類してハードウェアに結果をインストールするためにいくつかのパケットが使われます。この間に、分類は「不明」となり、トラフィックはデフォルトキューに入ります。

Application Visibility and Control に関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

有線ネットワークでの Application Visibility and Control の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	有線アプリケーションの表示およびコントロール（有線 AVC）属性ベース QoS（EasyQoS）	特定のプロトコルではなく、Network-Based Application Recognition（NBAR）属性に基づいて QoS クラスとポリシーを定義できるようになりましたが、いくつかの制限があります。サポートされる NBAR 属性は、business-relevance および traffic-class のみです。
Cisco IOS XE Gibraltar 16.11.1	有線ネットワークでの Application Visibility and Control	AVC は、アプリケーションへの適応力やアプリケーションへのインテリジェンス性に基づいて、厳密なパケットおよび接続からブランチおよびキャンパスソリューションを発展させるためのシスコの取り組みの重要な部分です。
Cisco IOS XE Gibraltar 16.12.1	DNS フローレコード	DNS フローレコードのサポートが導入されました。DNS フローレコードは、フローレコードを定義するための collect フィールドとして DNS ドメイン名を使用します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 6 章

SDM テンプレートの設定

- [SDM テンプレートに関する情報 \(159 ページ\)](#)
- [SDM テンプレートとスイッチ スタック \(159 ページ\)](#)
- [SDM テンプレートの設定方法 \(160 ページ\)](#)
- [SDM テンプレートのモニタリングおよびメンテナンス \(161 ページ\)](#)
- [SDM テンプレートの設定例 \(161 ページ\)](#)
- [SDM テンプレートに関する追加情報 \(163 ページ\)](#)
- [SDM テンプレートの機能履歴 \(163 ページ\)](#)

SDM テンプレートに関する情報

SDM テンプレートを使用してシステム リソースを設定すると、特定の機能に対するサポートをネットワーク内でのデバイスの使用方法に応じて最適化することができます。一部の機能に最大システム使用率を提供するようにテンプレートを選択できます。

Cisco Catalyst 9200 シリーズ スイッチは、次のテンプレートをサポートしています。

- Advanced
- VLAN

SDM テンプレートに変更を加えたらすぐにシステムをリロードすることを推奨します。テンプレートを変更し、システムを再起動した後、**show sdm prefer** 特権 EXEC コマンドを使用して、新しいテンプレート設定を確認できます。**reload** 特権 EXEC コマンドを入力する前に、**show sdm prefer** コマンドを入力すると、**show sdm prefer** コマンドにより、現在使用しているテンプレートおよびリロード後にアクティブになるテンプレートが表示されます。

SDM テンプレートとスイッチ スタック

1つのスイッチスタックで、すべてのスタックメンバにより、アクティブなスイッチに保存された同一の SDM テンプレートを使用する必要があります。新規スイッチがスタックに追加されると、アクティブスイッチに保存された SDM コンフィギュレーションは、個々のスイッチに設定されているテンプレートを上書きします。

show switch 特権 EXEC コマンドを使用すると、スタック メンバが SDM 不一致モードになっているかどうかを確認できます。

SDM テンプレートの設定方法

SDM テンプレートの設定

SDM テンプレートを使用して機能動作を最適にサポートするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	sdm prefer {advanced vlan} 例： Device(config)# sdm prefer vlan	SDM テンプレートを選択します。 <ul style="list-style-type: none"> • advanced : スイッチをアドバンスドテンプレートに設定します。 • vlan : ハードウェアでのルーティングをサポートしないスイッチでの VLAN 設定を最適化します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	reload 例： Device# reload	オペレーティング システムをリロードします。 システムの再起動後、 show sdm prefer 特権 EXEC コマンドを使用して、新しいテンプレート設定を確認できます。 reload 特権 EXEC コマンドを入力する前

	コマンドまたはアクション	目的
		に、 show sdm prefer コマンドを入力すると、 show sdm prefer コマンドにより、現在使用しているテンプレートおよびリロード後にアクティブになるテンプレートが表示されます。

SDM テンプレートのモニタリングおよびメンテナンス

コマンド	目的
show sdm prefer	使用中の SDM テンプレートを表示します。
reload	スイッチをリロードして、新しく設定した SDM テンプレートをアクティブにします。



- (注) SDM テンプレートには、テンプレートの一部として定義されているコマンドのみが含まれています。テンプレートで定義されていない別の関連コマンドがテンプレートで有効になっている場合、**show running config** コマンドを入力すると、該当するコマンドが表示されます。たとえば、SDM テンプレートで **switchport voice vlan** コマンドが有効になっている場合、(SDM テンプレートでは定義されていませんが) **spanning-tree portfast edge** コマンドも有効にすることができます。

SDM テンプレートを削除すると、そのような他の関連するコマンドも削除されるため、明示的に再設定しなければなりません。

SDM テンプレートの設定例

例：SDM テンプレートの表示

次に、詳細なテンプレート情報を表示した出力例を示します。

```
Device# show sdm prefer advanced
Showing SDM Template Info

This is the Advanced template.
Number of VLANs:                1024
Unicast MAC addresses:          16384
Overflow Unicast MAC addresses: 256
L2 Multicast entries:           1024
L3 Multicast entries:           1024
```

例 : SDM テンプレートの表示

```

Overflow L3 Multicast entries:                256
Directly connected routes:                   8192
Indirect routes:                             3072
STP Instances:                               128
Security Access Control Entries:             1408
QoS Access Control Entries:                  1024
Policy Based Routing ACEs:                   512
Netflow Input ACEs:                          128
Netflow Output ACEs:                         128
Ingress Netflow ACEs:                       128
Egress Netflow ACEs:                        128
Flow SPAN ACEs:                             256
Tunnels:                                     128
LISP Instance Mapping Entries:               128
Control Plane Entries:                       512
Input Netflow flows:                         8192
Output Netflow flows:                       8192
SGT/DGT (or) MPLS VPN entries:              2048
SGT/DGT (or) MPLS VPN Overflow entries:     256
Wired clients:                              2048
MACSec SPD Entries:                         128

```

These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.
* values can be modified by sdm cli.

次に、VLAN テンプレート情報を表示した出力例を示します。

```

Device# show sdm prefer vlan
Showing SDM Template Info

This is the VLAN template for a typical Layer 2 network.
Number of VLANs:                            1024
Unicast MAC addresses:                       16384
Overflow Unicast MAC addresses:              256
L2 Multicast entries:                       1024
L3 Multicast entries:                       1024
Overflow L3 Multicast entries:              256
Directly connected routes:                  4096
Indirect routes:                            2048
STP Instances:                              128
Security Access Control Entries:            1408
QoS Access Control Entries:                 1024
Policy Based Routing ACEs:                  512
Netflow Input ACEs:                         128
Netflow Output ACEs:                       128
Ingress Netflow ACEs:                      128
Egress Netflow ACEs:                       128
Flow SPAN ACEs:                             256
Tunnels:                                    128
LISP Instance Mapping Entries:              128
Control Plane Entries:                      512
Input Netflow flows:                        8192
Output Netflow flows:                       8192
SGT/DGT (or) MPLS VPN entries:              2048
SGT/DGT (or) MPLS VPN Overflow entries:     256
Wired clients:                              2048
MACSec SPD Entries:                         128

```

These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.
* values can be modified by sdm cli.

例：SDM テンプレートの設定

```
Device(config)# sdm prefer advanced
Device(config)# exit
Device# reload
Proceed with reload? [confirm]
```

SDM テンプレートに関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

SDM テンプレートの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	SDM テンプレート	標準のSDMテンプレートを使用すると、システムリソースを設定して、特定の機能のサポートを最適化できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 7 章

システム メッセージ ログの設定

- システム メッセージ ログの設定に関する情報 (165 ページ)
- システム メッセージ ログの設定方法 (168 ページ)
- システム メッセージ ログのモニタリングおよびメンテナンス (177 ページ)
- システム メッセージ ログの設定例 (177 ページ)
- システム メッセージ ログに関する追加情報 (178 ページ)
- システムメッセージログの機能履歴 (178 ページ)

システム メッセージ ログの設定に関する情報

システム メッセージ ロギング

スイッチはデフォルトで、システム メッセージおよび **debug** 特権 EXEC コマンドの出力をロギングプロセスに送信します。スタック内のメンバスイッチはシステムメッセージをトリガーできます。システムメッセージを生成するメンバスイッチは、ホスト名を `hostname-n` の形式 (`n` はスイッチ) で付加し、出力をアクティブスイッチのロギングプロセスにリダイレクトします。アクティブスイッチはスタックメンバですが、そのホスト名はシステムメッセージの末尾に追加されません。ロギングプロセスはログメッセージを各宛先 (設定に応じて、ログバッファ、端末回線、UNIX Syslog サーバなど) に配信する処理を制御します。ロギングプロセスは、コンソールにもメッセージを送信します。

ロギングプロセスがディセーブルの場合、メッセージはコンソールにのみ送信されます。メッセージは生成時に送信されるため、メッセージおよびデバッグ出力にはプロンプトや他のコマンドの出力が割り込みます。メッセージがアクティブなコンソールに表示されるのは、メッセージを生成したプロセスが終了してからです。

メッセージの重大度を設定して、コンソールおよび各宛先に表示されるメッセージのタイプを制御できます。ログメッセージにタイムスタンプを設定したり、Syslog 送信元アドレスを設定したりして、リアルタイムのデバッグ機能および管理機能を強化できます。表示されるメッセージについては、このリリースに対応するシステムメッセージガイドを参照してください。

ロギングされたシステムメッセージにアクセスするには、スイッチのコマンドラインインターフェイス (CLI) を使用するか、または適切に設定された Syslog サーバにこれらのシステム

メッセージを保存します。スイッチソフトウェアは、Syslog メッセージをスタンドアロンスイッチ上の内部バッファに保存します。スイッチスタックの場合は、アクティブスイッチ上に保存します。スタンドアロンスイッチまたはアクティブスイッチに障害が発生すると、ログをフラッシュメモリに保存していなかった場合、ログは失われます。

システムメッセージをリモートで監視するには、Syslog サーバ上でログを表示するか、あるいは Telnet、コンソールポート、またはイーサネット管理ポート経由でスイッチにアクセスします。スイッチスタックでは、すべてのメンバスイッチコンソールにより、同じコンソール出力が用意されます。



(注) Syslog フォーマットは 4.3 Berkeley Standard Distribution (BSD) UNIX と互換性があります。

システム ログ メッセージのフォーマット

システム ログ メッセージは最大 80 文字とパーセント記号 (%)、およびその前に配置されるオプションのシーケンス番号やタイムスタンプ情報（設定されている場合）で構成されています。スイッチに応じて、メッセージは次のいずれかの形式で表示されます。

- *seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)*
- *seq no:timestamp: %facility-severity-MNEMONIC:description*

パーセント記号の前にあるメッセージの部分は、次のグローバル コンフィギュレーション コマンドの設定によって異なります。

- **service sequence-numbers**
- **service timestamps log datetime**
- **service timestamps log datetime[localtime] [msec] [show-timezone]**
- **service timestamps log uptime**

表 8: システム ログ メッセージの要素

要素	説明
<i>seq no:</i>	service sequence-numbers グローバル コンフィギュレーション コマンドが設定されている場合にのみ、ログメッセージにシーケンス番号をスタンプします。

要素	説明
<i>timestamp formats:</i> <i>mm/dd h h:mm:ss</i> または <i>hh:mm:ss</i> (短時間) または <i>d h</i> (長時間)	メッセージまたはイベントの日時です。この情報が表示されるのは、 service timestamps log[datetime log] グローバル コンフィギュレーション コマンドが設定されている場合のみです。
<i>facility</i>	メッセージが参照する機能 (SNMP、SYS など) です。
<i>severity</i>	メッセージの重大度を示す 0 ~ 7 の 1 桁のコードです。
<i>MNEMONIC</i>	メッセージを一意に示すテキスト ストリングです。
<i>description</i>	レポートされているイベントの詳細を示すテキスト ストリングです。

デフォルトのシステムメッセージログの設定

表 9: デフォルトのシステムメッセージログの設定

機能	デフォルト設定
コンソールへのシステムメッセージログ	イネーブル
コンソールの重大度	デバッグ
ログファイル設定	ファイル名の指定なし
ログバッファサイズ	4096 バイト
ログ履歴サイズ	1 メッセージ
タイムスタンプ	ディセーブル
同期ログ	ディセーブル
ログサーバ	ディセーブル
Syslog サーバの IP アドレス	未設定
サーバ機能	local7
サーバの重大度	通知

syslog メッセージの制限

snmp-server enable trap グローバルコンフィギュレーションコマンドを使用して、SNMP ネットワーク管理ステーションに送信されるようにsyslogメッセージトラップが設定されている場合は、スイッチの履歴テーブルに送信および格納されるメッセージの重大度を変更できます。また、履歴テーブルに格納されるメッセージの数を変更することもできます。

SNMPトラップは宛先への到達が保証されていないため、メッセージは履歴テーブルに格納されます。デフォルトでは、syslogトラップが有効でない場合も、レベルが **warning** であるメッセージや数値的に下位レベルのメッセージの1つが履歴テーブルに格納されます。

履歴テーブルがいっぱいの場合 (**logging history size** グローバルコンフィギュレーションコマンドで指定した最大メッセージエントリ数に達している場合) は、新しいメッセージエントリを格納できるように、最も古いエントリがテーブルから削除されます。

履歴テーブルは、level キーワードおよび重大度を示します。SNMPを使用している場合は、重大度の値が1だけ増えます。たとえば、*emergencies* は0ではなく1に、*critical* は2ではなく3になります。

システムメッセージログの設定方法

メッセージ表示宛先デバイスの設定

メッセージロギングがイネーブルの場合、コンソールだけでなく特定の場所にもメッセージを送信できます。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバルコンフィギュレーションモードを開始します。
ステップ 2	logging buffered [size] 例： Device(config)# logging buffered 8192	スイッチ上、またはスタンドアロンスイッチ上、あるいはスイッチスタックの場合、アクティブスイッチ上で、ログメッセージを内部バッファに保存します。指定できる範囲は4096～2147483647バイトです。デフォルトのバッファサイズは4096バイトです。

	コマンドまたはアクション	目的
		<p>スタンドアロンスイッチまたはアクティブスイッチに障害が発生すると、ログファイルをフラッシュメモリに保存していなかった場合、ログファイルは失われます。ステップ4を参照してください。</p> <p>(注) バッファサイズを大きすぎる値に設定しないでください。他の作業に使用するメモリが不足することがあります。スイッチ上の空きプロセスメモリを表示するには、show memory 特権 EXEC コマンドを使用します。ただし、表示される値は使用できる最大値であるため、バッファサイズをこの値に設定しないでください。</p>
ステップ3	<p>logging host</p> <p>例 :</p> <pre>Device(config)# logging 125.1.1.100</pre>	<p>UNIX Syslog サーバホストにメッセージを保存します。</p> <p><i>host</i> には、syslog サーバとして使用するホストの名前または IP アドレスを指定します。</p> <p>ログメッセージを受信する Syslog サーバのリストを作成するには、このコマンドを複数回入力します。</p>
ステップ4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ5	<p>terminal monitor</p> <p>例 :</p> <pre>Device# terminal monitor</pre>	<p>現在のセッション間、非コンソール端末にメッセージを保存します。</p> <p>端末パラメータ コンフィギュレーションコマンドはローカルに設定され、セッションの終了後は無効になります。デバッグメッセージを表示する場合は、セッションごとにこのステップを実行する必要があります。</p>

ログメッセージの同期化

特定のコンソールポート回線または仮想端末回線に対して、非送信請求メッセージおよび **debug** 特権 EXEC コマンドの出力を送信請求デバイスの出力およびプロンプトと同期させることができます。重大度に応じて非同期に出力されるメッセージのタイプを特定できます。また、端末の非同期メッセージが削除されるまで保存しておくバッファの最大数を設定することもできます。

非送信請求メッセージおよび **debug** コマンド出力の同期ロギングがイネーブルの場合、送信請求デバイス出力がコンソールに表示または印刷された後に、非送信請求デバイスからの出力が表示または印刷されます。非送信請求メッセージおよび **debug** コマンドの出力は、ユーザ入力用プロンプトが返された後に、コンソールに表示されます。したがって、非送信請求メッセージおよび **debug** コマンドの出力は、送信請求デバイス出力およびプロンプトに割り込まれることはありません。非送信請求メッセージが表示された後に、コンソールはユーザプロンプトを再表示します。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	line [console vty] line-number [ending-line-number] 例 : Device(config)# line console	メッセージの同期ロギングに設定する回線を指定します。 <ul style="list-style-type: none"> • console : スイッチコンソールポートまたはイーサネット管理ポートでの設定を指定します。 • line vty line-number : どの vty 回線の同期ロギングをイネーブルにするかを指定します。Telnet セッションを介して行われる設定には、vty 接続を使用します。回線番号に指定できる範囲は 0 ~ 15 です。 16 個の vty 回線の設定をすべて一度に変更するには、次のように入力します。 line vty 0 15 また、現在の接続に使用されている 1 つの vty 回線の設定を変更することもでき

	コマンドまたはアクション	目的
		<p>ます。たとえば、vty 回線 2 の設定を変更するには、次のように入力します。</p> <pre>line vty 2</pre> <p>このコマンドを入力すると、ライン コンフィギュレーション モードになります。</p>
ステップ 3	<p>logging synchronous [level [severity-level all] limit number-of-buffers]</p> <p>例 :</p> <pre>Device(config)# logging synchronous level 3 limit 1000</pre>	<p>メッセージの同期ログをイネーブルにします。</p> <ul style="list-style-type: none"> • (任意) level severity-level : メッセージの重大度レベルを指定します。重大度がこの値以上であるメッセージは、非同期に出力されます。値が小さいほど重大度は大きく、値が大きいほど重大度は小さくなります。デフォルトは 2 です。 • (任意) level all : 重大度に関係なく、すべてのメッセージが非同期に出力されます。 • (任意) limit number-of-buffers : キューイングされる端末のバッファ数を指定します。これを超える新しいメッセージは廃棄されます。指定できる範囲は 0 ~ 2147483647 です。デフォルトは 20 です。
ステップ 4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>

メッセージログのディセーブル化

メッセージログはデフォルトでイネーブルに設定されています。コンソール以外のいずれかの宛先にメッセージを送信する場合は、メッセージログをイネーブルにする必要があります。メッセージログがイネーブルの場合、ログメッセージはログプロセスに送信されます。ログプロセスは、メッセージを生成元プロセスと同期しないで指定場所に記録します。

ロギングプロセスをディセーブルにすると、メッセージがコンソールに書き込まれるまでプロセスは処理続行を待機する必要があるため、スイッチの処理速度が低下することがあります。ロギングプロセスがディセーブルの場合、メッセージは生成後すぐに（通常はコマンド出力に割り込む形で）コンソールに表示されます。

logging synchronous グローバルコンフィギュレーションコマンドも、コンソールへのメッセージ表示に影響します。このコマンドをイネーブルにすると、Returnを押さなければメッセージが表示されません。

メッセージロギングをディセーブルにした後に再びイネーブルにするには、**logging on** グローバルコンフィギュレーションコマンドを使用します。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバルコンフィギュレーションモードを開始します。
ステップ 2	no logging console 例： Device(config)# no logging console	メッセージロギングをディセーブルにします。
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。

ログメッセージのタイムスタンプのイネーブル化およびディセーブル化

デフォルトでは、ログメッセージにはタイムスタンプが適用されません。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	次のいずれかのコマンドを使用します。 <ul style="list-style-type: none"> • service timestamps log uptime • service timestamps log datetime[msec localtime show-timezone] 例 : Device (config)# service timestamps log uptime または Device (config)# service timestamps log datetime	ログのタイムスタンプをイネーブルにします。 <ul style="list-style-type: none"> • log uptime : ログメッセージのタイムスタンプをイネーブルにして、システムの再起動以降の経過時間を表示します。 • log datetime : ログメッセージのタイムスタンプをイネーブルにします。選択したオプションに応じて、ローカルタイムゾーンを基準とした日付、時間（ミリ秒）、タイムゾーン名をタイムスタンプとして表示できます。
ステップ 3	end 例 : Device (config)# end	特権 EXEC モードに戻ります。

ログメッセージのシーケンス番号のイネーブル化およびディセーブル化

タイムスタンプが同じログメッセージが複数ある場合、これらのメッセージを表示するには、シーケンス番号を使用してメッセージを表示できます。デフォルトでは、ログメッセージにシーケンス番号は表示されません。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 :	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
	Device# configure terminal	
ステップ 2	service sequence-numbers 例： Device(config)# service sequence-numbers	シーケンス番号をイネーブルにします。
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。

メッセージ重大度の定義

メッセージの重大度を指定して、選択したデバイスに表示されるメッセージを制限します。
このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	logging console level 例： Device(config)# logging console 3	コンソールに保存するメッセージを制限します。 デフォルトで、コンソールはデバッグメッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 3	logging monitor level 例： Device(config)# logging monitor 3	端末回線に出力するメッセージを制限します。 デフォルトで、端末はデバッグメッセージ、および数値的により低いレベルのメッセージを受信します。

	コマンドまたはアクション	目的
ステップ 4	logging trap level 例： Device(config)# logging trap 3	Syslog サーバに保存するメッセージを制限します。 デフォルトで、Syslog サーバは通知メッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 5	end 例： Device(config)# end	特権 EXEC モードに戻ります。

履歴テーブルおよび SNMP に送信される Syslog メッセージの制限

このタスクでは、履歴テーブルおよび SNMP に送信される syslog メッセージを制限する方法について説明します。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	logging history level 例： Device(config)# logging history 3	履歴ファイルに保存され、SNMP サーバに送信される syslog メッセージのデフォルト レベルを変更します。 デフォルトでは warnings 、 errors 、 critical 、 alerts 、および emergencies メッセージは送信されません。
ステップ 3	logging history size number 例： Device(config)# logging history size 200	履歴テーブルに保存できる Syslog メッセージの数を指定します。 デフォルトでは 1 つのメッセージが格納されます。指定できる範囲は 0 ~ 500 です。
ステップ 4	end 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	Device (config) # end	

UNIX Syslog デーモンへのメッセージのロギング

このタスクはオプションです。



- (注) 最新バージョンの UNIX Syslog デーモンの中には、デフォルトでネットワークからの Syslog パケットを受け入れないものがあります。このようなシステムの場合に、Syslog メッセージのリモートロギングをイネーブルにするには、Syslog コマンドラインに追加または削除する必要があるオプションを、UNIX の **man syslogd** コマンドを使用して判別します。

始める前に

- root としてログインします。
- システム ログメッセージを UNIX Syslog サーバに送信する前に、UNIX サーバ上で Syslog デーモンを設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>/etc/syslog.conf ファイルに次の行を追加します。</p> <p>例 :</p> <pre>local7.debug /usr/adm/logs/cisco.log</pre>	<ul style="list-style-type: none"> • local7 : ロギング機能を指定します。 • debug : syslog レベルを指定します。このファイルは、syslog デーモンに書き込み権限がある既存ファイルである必要があります。
ステップ 2	<p>UNIX シェルプロンプトに次のコマンドを入力します。</p> <p>例 :</p> <pre>\$ touch /var/log/cisco.log \$ chmod 666 /var/log/cisco.log</pre>	<p>ログファイルを作成します。syslog デーモンは、このレベルまたはこのファイルのより高い重大度レベルでメッセージを送信します。</p>
ステップ 3	<p>Syslog デーモンに新しい設定を認識させます。</p> <p>例 :</p>	<p>詳細については、ご使用の UNIX システムの man syslog.conf および man syslogd コマンドを参照してください。</p>

	コマンドまたはアクション	目的
	<code>\$ kill -HUP `cat /etc/syslog.pid`</code>	

システムメッセージログのモニタリングおよびメンテナンス

コンフィギュレーションアーカイブログのモニタリング

コマンド	目的
<code>show archive log config {all number [end-number] user username [session number] number [end-number] statistics} [provisioning]</code>	コンフィギュレーションログ全体、または指定されたパラメータのログを表示します。

システムメッセージログの設定例

例：システムメッセージのスタック構成

次の例では、アクティブスイッチの部分的なスイッチシステムメッセージとスタックメンバ（ホスト名は *Switch-2*）を示します。

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)

00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1, changed
state to down 2 (Switch-2)
```

例：スイッチシステムメッセージ

次に、スイッチ上のスイッチシステムメッセージの一部を示します。

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

システムメッセージログに関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

システムメッセージログの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	システムメッセージログ	システムメッセージ出力は、ロギングプロセスに送信されます。ロギングプロセスはログメッセージを各宛先（設定に応じて、ログバッファ、端末回線、UNIX Syslog サーバなど）に配信する処理を制御します

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 8 章

オンライン診断の設定

- [オンライン診断の設定に関する情報](#) (179 ページ)
- [オンライン診断の設定方法](#) (181 ページ)
- [オンライン診断のモニタリングおよびメンテナンス](#) (187 ページ)
- [オンライン診断テストの設定例](#) (187 ページ)
- [オンライン診断に関する追加情報](#) (189 ページ)
- [オンライン診断設定の機能情報](#) (189 ページ)

オンライン診断の設定に関する情報

オンライン診断機能を使用すると、デバイスをアクティブネットワークに接続したまま、デバイスのハードウェア機能をテストして確認できます。オンライン診断には、異なるハードウェアコンポーネントをチェックするパケット交換テストが含まれ、データパスおよび制御信号が確認されます。

オンライン診断では、次の領域の問題が検出されます。

- ハードウェア コンポーネント
- インターフェイス (イーサネット ポートなど)
- はんだ接合

オンライン診断は、オンデマンド診断、スケジュール診断、ヘルスマニタリング診断に分類できます。オンデマンド診断は、CLIから実行されます。スケジュールされた診断は、動作中のネットワークにデバイスが接続されているときに、ユーザが指定した間隔または指定した時刻に実行されます。ヘルスマニタリングは、バックグラウンドでユーザが指定した間隔で実行されます。ヘルスマニタリングテストは、テストに基づいて 90、100、または 150 秒ごとに実行されます。

オンライン診断を設定したあと、手動で診断テストを開始したり、テスト結果を表示したりできます。また、デバイスまたはスイッチスタックに設定されているテストの種類、およびすでに実行された診断テスト名を確認できます。

Generic Online Diagnostics (GOLD)



- (注)
- オンライン診断テストをイネーブルにする前に、コンソールロギングをイネーブルにしてすべての警告メッセージを表示してください。
 - テストの実行中、ポートを内部的にループしてストレステストを行いますが、外部トラフィックがテスト結果に影響を与えることがあるため、すべてのポートがシャットダウンされます。スイッチを正常な稼働に戻すために、スイッチをリロードしなければなりません。スイッチをリロードするコマンドを入力すると、コンフィギュレーションを保存するかどうかを聞かれます。コンフィギュレーションは保存しないでください。
 - 他のモジュール上でテストを実行している場合、テストが開始され、完了したら、モジュールをリセットする必要があります。

TestPortTxMonitoring

このテストは、ステータスが UP のデバイスに物理的に接続されている各ネットワークポートの送信方向のデータパストラフィックを定期的にモニタします。このテストは、ポートあたり 1 ミリ秒以内に完了します。また、このテストでは、ASIC レベルで送信カウンタをモニタして、ポートがスタックしていないことを確認します。テストでは syslog メッセージが表示され、ユーザは Cisco IOS Embedded Event Manager (EEM) を使用して修正アクションを実行できます。

diagnostic monitor interval および **diagnostic monitor threshold** コマンドをそれぞれ入力して、時間間隔としきい値を設定します。テストでは、パケットを送信する Cisco Discovery Protocol (CDP) プロトコルを利用します。テストは 75 秒ごとに実行され、障害しきい値はデフォルトで 5 に設定されています。

属性	説明
ディスラプティブまたはノンディスラプティブ	ノンディスラプティブ
推奨事項	ディセーブルにしないでください。
デフォルト	オン
修正処置	ポートに障害が発生したことを示す syslog メッセージを表示します。
ハードウェア サポート	スーパーバイザ エンジンを含むすべてのモジュール

TestUnusedPortLoopback

このテストは、実行時にスーパーバイザモジュールとモジュールのネットワークポート間のデータパスを定期的を確認し、着信ネットワーク インターフェイス ポートがロックされているかどうかを判断します。このテストでは、レイヤ2の packets はテストポートおよびスーパーバイザエンジンのインバンドポートに関連付けられた VLAN にフラッドされます。パケットはテストポート内をループバックして、同じ VLAN のスーパーバイザエンジンに戻ります。このテストは、ケーブルが接続されているかどうかに関係なく、未使用の（管理上のダウン、つまりポートがシャットダウンされている）ネットワークポートでのみ実行され、ポートあたり 1 ミリ秒以内に完了します。このテストは、現在の ASIC にノンディスラプティブループバックテストがないため、代用として使用され、60 秒ごとに実行されます。

属性	説明
ディスラプティブまたはノンディスラプティブ	ノンディスラプティブ
推奨事項	ディセーブルにしないでください。CPU 使用率の急上昇中、このテストは精度を維持するために自動的にディセーブルになります。
デフォルト	オン
修正処置	ポートに障害が発生したことを示す syslog メッセージを表示します。スーパーバイザエンジン以外のモジュールでは、すべてのポートグループに障害が発生した場合（たとえば、ポート ASIC ごとに最低 1 つのポートで、すべてのポート ASIC の障害しきい値より多く障害が発生した場合）、デフォルトのアクションではモジュールがリセットされ、リセットを 2 回行ったあとにモジュールの電源を切断します。
ハードウェア サポート	スーパーバイザ エンジンを含むすべてのモジュール

オンライン診断の設定方法

オンライン診断テストの開始

デバイスで実行する診断テストを設定したあと、**diagnostic start** 特権 EXEC コマンドを使用して診断テストを開始します。

テストを開始したら、テストプロセスの停止はできません。

手動でオンライン診断テストを開始するには、次の特権 EXEC 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>diagnostic start switch <i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic complete minimal non-disruptive per-port}</p> <p>例 :</p> <pre>Device# diagnostic start switch 2 test basic</pre>	<p>診断テストを開始します。</p> <p>switch <i>number</i> キーワードは、スタック構成デバイスだけでサポートされます。</p> <p>次のいずれかのオプションを使用してテストを指定できます。</p> <ul style="list-style-type: none"> • name : テストの名前を入力します。 • test-id : テストの ID 番号を入力します。 • test-id-range : カンマとハイフンで区切ってテスト ID の範囲を整数で入力します。 • all : すべてのテストを開始します。 • basic : 基本テストスイートを開始します。 • complete : 完全なテストスイートを開始します。 • minimal : 最小限のブートアップテストスイートを開始します。 • non-disruptive : ノンディスラプティブテストスイートを開始します。 • per-port : ポート単位のテストスイートを開始します。

オンライン診断の設定

診断モニタリングをイネーブルにする前に、障害しきい値およびテストの間隔を設定する必要があります。

オンライン診断のスケジューリング

特定のデバイスについて指定した時間、または日、週、月単位でオンライン診断をスケジューリングできます。スケジューリングを削除するには、コマンドの **no** 形式を入力します。

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>Device #configure terminal</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>
ステップ 2	<p>diagnostic schedule number test {name test-id test-id-range all basic complete minimal non-disruptive per-port} {daily on mm dd yyyy hh:mm port inter-port-number port-number-list weekly day-of-week hh:mm}</p> <p>例 :</p> <pre>Device(config)# diagnostic schedule 3 test 1-5 on July 3 2013 23:10</pre>	<p>特定日時のオンデマンド診断テストをスケジューリングします。</p> <p>switch number キーワードは、スタック構成スイッチだけでサポートされます。</p> <p>スケジューリングするテストを指定する場合は、次のオプションを使用します。</p> <ul style="list-style-type: none"> • name : show diagnostic content コマンドの出力に表示されるテストの名前です。 • test-id : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • test-id-range : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべてのテスト ID • basic : 基本的なオンデマンドの診断テストを開始します。 • complete : 完全なテストスイートを開始します。 • minimal : 最小限のブートアップテストスイートを開始します。 • non-disruptive : ノンディスラプティブテストスイートを開始します。 • per-port : ポート単位のテストスイートを開始します。 <p>テストは次のようにスケジューリングできません。</p> <ul style="list-style-type: none"> • 毎日 : daily hh:mm パラメータを使用します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • 特定日時 : on mm dd yyyy hh:mm パラメータを使用します。 • 毎週 : weekly day-of-week hh:mm パラメータを使用します。

ヘルス モニタリング診断の設定

デバイスが稼働中のネットワークに接続されている間に、スイッチに対しヘルスモニタリング診断テストを設定できます。各ヘルスモニタリングテストの実行間隔を設定したり、デバイスをイネーブルにし、テスト失敗時の Syslog メッセージを生成したり、特定のテストをイネーブルにできます。

テストをディセーブルにするには、コマンドの **no** 形式を入力します。

デフォルトでは、ヘルスモニタリングはディセーブルですが、デバイスはテストの失敗時に Syslog メッセージを生成します。

ヘルス モニタリング診断テストを設定し、イネーブルにするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	diagnostic monitor interval switch <i>number</i> test { <i>name</i> <i>test-id</i> <i>test-id-range</i> all } <i>hh:mm:ss milliseconds day</i> 例 : Device(config)# diagnostic monitor interval switch 2 test 1 12:30:00 750 5	指定のテストに対し、ヘルスモニタリングの実行間隔を設定します。 switch number キーワードは、スタック構成スイッチだけでサポートされません。 テストを指定する場合は、次のいずれかのパラメータを使用します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • name : show diagnostic content コマンドの出力に表示されるテストの名前です。 • test-id : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • test-id-range : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべての診断テスト。 <p>間隔を指定する場合は、次のパラメータを設定します。</p> <ul style="list-style-type: none"> • hh:mm:ss : モニタリング間隔 (時間、分、秒)。指定できる範囲は <i>hh</i> が 0 ~ 24、<i>mm</i> および <i>ss</i> が 0 ~ 60 です。 • milliseconds : モニタリング間隔 (ミリ秒 (ms))。指定できる範囲は 0 ~ 999 です。 • day : モニタリング間隔 (日数)。指定できる範囲は 0 ~ 20 です。
<p>ステップ 4</p>	<p>diagnostic monitor syslog</p> <p>例 :</p> <pre>Device(config)# diagnostic monitor syslog</pre>	<p>(任意) ヘルスモニタリングテストの失敗時にスイッチが Syslog メッセージを生成するように設定します。</p>
<p>ステップ 5</p>	<p>diagnostic monitor threshold switch number number test {name test-id test-id-range all} failure count count</p> <p>例 :</p> <pre>Device(config)# diagnostic monitor threshold switch 2 test 1 failure count 20</pre>	<p>(任意) ヘルスモニタリングテストの失敗しきい値を設定します。</p> <p>テストを指定する場合は、次のいずれかのパラメータを使用します。</p> <ul style="list-style-type: none"> • name : show diagnostic content コマンドの出力に表示されるテストの名前です。 • test-id : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>test-id-range</i> : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべての診断テスト。 <p>失敗しきい値 <i>count</i> に指定できる範囲は 0 ~ 99 です。</p>
ステップ 6	<p>diagnostic monitor switch<i>number test</i> {<i>name</i> <i>test-id</i> <i>test-id-range</i> all}</p> <p>例 :</p> <pre>Device(config)# diagnostic monitor switch 2 test 1</pre>	<p>指定のヘルスモニタリングテストをイネーブルにします。</p> <p>switch <i>number</i> キーワードは、スタック構成スイッチだけでサポートされません。</p> <p>テストを指定する場合は、次のいずれかのパラメータを使用します。</p> <ul style="list-style-type: none"> • <i>name</i> : show diagnostic content コマンドの出力に表示されるテストの名前です。 • <i>test-id</i> : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • <i>test-id-range</i> : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべての診断テスト。
ステップ 7	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 8	<p>show diagnostic { content post result schedule status switch }</p>	<p>オンライン診断のテスト結果およびサポートされるテストスイートを表示します。</p>
ステップ 9	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	<p>入力を確認します。</p>

	コマンドまたはアクション	目的
ステップ 10	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

オンライン診断のモニタリングおよびメンテナンス

オンライン診断テストの設定例

例：診断テストの開始

次に、テスト名を指定して診断テストを開始する例を示します。

```
Device# diagnostic start switch 2 test DiagPOETest
```

次に、すべての基本診断テストを開始する例を示します。

```
Device# diagnostic start switch 1 test all
```

例：ヘルス モニタリング テストの設定

次に、ヘルス モニタリング テストを設定する例を示します。

```
Device(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
Device(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

例：診断テストのスケジューリング

次に、特定のスイッチに対して、特定の日に診断テストを実行するようにスケジューリングする例を示します。

```
Device(config)# diagnostic schedule test DiagThermalTest on June 3 2013 22:25
```

次の例では、指定されたスイッチで毎週特定の時間に診断テストを実行するようにスケジューリングする方法を示します。

```
Device(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

例: オンライン診断の表示

次に、オンデマンド診断設定を表示する例を示します。

```
Device# show diagnostic ondemand settings
```

```
Test iterations = 1  
Action on test failure = continue
```

次に、障害の診断イベントを表示する例を示します。

```
Device# show diagnostic events event-type error
```

```
Diagnostic events (storage for 500 events, 0 events recorded)  
Number of events matching above criteria = 0
```

```
No diagnostic log entry exists.
```

次に、診断テストの説明を表示する例を示します。

```
Device# show diagnostic description switch 1 test all
```

```
DiagGoldPktTest :
```

```
The GOLD packet Loopback test verifies the MAC level loopback  
functionality. In this test, a GOLD packet, for which doppler  
provides the support in hardware, is sent. The packet loops back  
at MAC level and is matched against the stored packet. It is a non-  
-disruptive test.
```

```
DiagThermalTest :
```

```
This test verifies the temperature reading from the sensor is below the yellow  
temperature threshold. It is a non-disruptive test and can be run as a health  
monitoring test.
```

```
DiagFanTest :
```

```
This test verifies all fan modules have been inserted and working properly on  
the board  
It is a non-disruptive test and can be run as a health monitoring test.
```

```
DiagPhyLoopbackTest :
```

```
The PHY Loopback test verifies the PHY level loopback  
functionality. In this test, a packet is sent which loops back  
at PHY level and is matched against the stored packet. It is a  
disruptive test and cannot be run as a health monitoring test.
```

```
DiagScratchRegisterTest :
```

```
The Scratch Register test monitors the health of application-specific  
integrated circuits (ASICs) by writing values into registers and reading  
back the values from these registers. It is a non-disruptive test and can  
be run as a health monitoring test.
```

```
DiagPoETest :
```

```
This test checks the PoE controller functionality. This is a disruptive test  
and should not be performed during normal switch operation.
```

```
DiagStackCableTest :
```

```
This test verifies the stack ring loopback functionality
```

in the stacking environment. It is a disruptive test and cannot be run as a health monitoring test.

DiagMemoryTest :
 This test runs the exhaustive ASIC memory test during normal switch operation
 NG3K utilizes mbist for this test. Memory test is very disruptive
 in nature and requires switch reboot after the test.

Device#

オンライン診断に関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

オンライン診断設定の機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	オンライン診断	オンライン診断機能を使用すると、デバイスをアクティブ ネットワークに接続したまま、デバイスのハードウェア機能をテストして確認できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 9 章

コンフィギュレーション ファイルの管理

- [コンフィギュレーション ファイルの管理の前提条件](#) (191 ページ)
- [コンフィギュレーション ファイルの管理の制約事項](#) (191 ページ)
- [コンフィギュレーション ファイルの管理について](#) (192 ページ)
- [コンフィギュレーション ファイル情報の管理方法](#) (200 ページ)
- [コンフィギュレーション ファイルの管理の機能履歴](#) (231 ページ)

コンフィギュレーション ファイルの管理の前提条件

- ユーザには、少なくとも Cisco IOS 環境とコマンドラインインターフェイスに関する基本的な知識が必要です。
- システムでは、少なくとも最小限の設定が実行されていることが必要です。基本コンフィギュレーション ファイルは、**setup** コマンドを使用して作成できます。

コンフィギュレーション ファイルの管理の制約事項

- このドキュメントで説明されている Cisco IOS コマンドの多くは、デバイスの特定のコンフィギュレーション モードでのみ使用可能であり機能します。
- Cisco IOS コンフィギュレーション コマンドのいくつかは、特定のデバイスプラットフォームでのみ使用可能であり、コマンド構文はプラットフォームによって異なる可能性があります。

コンフィギュレーションファイルの管理について

コンフィギュレーションファイルのタイプ

コンフィギュレーションファイルには、シスコ製デバイスの機能をカスタマイズするための Cisco IOS ソフトウェアコマンドが含まれています。コマンドは、システムを起動したとき (startup-config ファイルから)、または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 IOS 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 (startup-config) は、ソフトウェアを設定するためにシステムの起動時に使用されます。実行コンフィギュレーションファイル (running-config) には、ソフトウェアの現在の設定が含まれています。2つのコンフィギュレーションファイルは別々の設定にできます。たとえば、コンフィギュレーションを永続的ではなく短期間で変更する場合があります。その場合は、**configure terminal EXEC** 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、そのコンフィギュレーションは **copy running-config startup-config EXEC** コマンドを使用して保存しません。

実行コンフィギュレーションを変更するには、[コンフィギュレーションファイルの変更 \(201 ページ\)](#) の項で説明されているように、**configure terminal** コマンドを使用します。Cisco IOS 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、**copy running-config startup-config EXEC** 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にコンフィギュレーションファイルをコピーします (詳細については、「[TFTP サーバからデバイスへのコンフィギュレーションファイルのコピー](#)」を参照してください)。

コンフィギュレーションモードおよびコンフィギュレーションソースの選択

デバイス上でコンフィギュレーションモードを開始するには、特権 EXEC プロンプトで **configure** コマンドを入力します。Cisco IOS ソフトウェアは次のプロンプトで応答し、端末、メモリ、またはネットワークサーバ (ネットワーク) 上に格納されたファイルのいずれかを、コンフィギュレーションコマンドのソースとして指定するように要求されます。

```
Configuring from terminal, memory, or network [terminal]?
```

端末からの設定では、コマンドラインにコンフィギュレーションコマンドを入力できます (次の項を参照してください)。詳細については、[スタートアップコンフィギュレーションファイルでのコンフィギュレーションコマンドの再実行](#) の項を参照してください。

ネットワークからの設定では、ネットワーク経由でコンフィギュレーション コマンドをロードして実行できます。詳細については、[TFTP サーバからデバイスへのコンフィギュレーション ファイルのコピー](#) の項を参照してください。

CLI を使用したコンフィギュレーション ファイルの変更

Cisco IOS ソフトウェアは、1 行につき 1 つのコンフィギュレーション コマンドを受け入れません。コンフィギュレーション コマンドは、必要なだけ入力できます。コンフィギュレーション ファイルには、入力したコマンドを説明するコメントを追加できます。コメントの先頭には、感嘆符 (!) を付けます。コメントは NVRAM にもコンフィギュレーション ファイルのアクティブコピーにも格納されないため、**show running-config** または **more system:running-config EXEC** コマンドでアクティブな設定のリストを表示しても、コメントは表示されません。**show startup-config** または **more nvram:startup-config EXEC** モードコマンドでスタートアップ コンフィギュレーションのリストを表示しても、コメントは表示されません。コメントは、コンフィギュレーション ファイルがデバイスにロードされたときにコンフィギュレーション ファイルから削除されます。ただし、ファイル転送プロトコル (FTP)、リモートコピープロトコル (RCP)、または Trivial File Transfer Protocol (TFTP) サーバ上に格納されているコンフィギュレーション ファイルのコメントのリストは表示できます。CLI を使用してソフトウェアは設定するときは、ユーザの入力に従ってソフトウェアによりコマンドが実行されます。

コンフィギュレーション ファイルの場所

コンフィギュレーション ファイルは、次の場所に格納されます。

- 実行コンフィギュレーションは RAM に格納されます。
- クラス A フラッシュ ファイル システム プラットフォーム以外のすべてのプラットフォーム上では、スタートアップ コンフィギュレーションは不揮発性 RAM (NVRAM) に格納されます。
- クラス A フラッシュ ファイル システムのプラットフォーム上では、スタートアップ コンフィギュレーションは CONFIG_FILE 環境変数で指定された場所に格納されます ([クラス A フラッシュ ファイル システムでの CONFIG_FILE 環境変数の指定 \(225 ページ\)](#) の項を参照してください)。CONFIG_FILE 変数は、デフォルトでは NVRAM になりますが、次のファイル システムのファイルも指定できます。
 - **nvram:** (NVRAM)
 - **flash:** (内部フラッシュ メモリ)
 - **usbflash0:** (外部 usbflash ファイル システム)
 - **usbflash1:** (外部 usbflash ファイル システム)

ネットワークサーバからデバイスへのコンフィギュレーションファイルのコピー

TFTP、`rcp`、または FTP サーバからデバイスの実行コンフィギュレーションまたは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できます。この機能は、次のいずれかの理由により実行する場合があります。

- バックアップコンフィギュレーションファイルを復元するため。
- 別のデバイスのコンフィギュレーションファイルを使用するため。たとえば、別のデバイスをネットワークに追加して、そのデバイスのコンフィギュレーションを元のデバイスと同様にする場合です。ファイルを新しいデバイスにコピーすると、ファイル全体を再作成しないで、関連部分を変更できます。
- 同一のコンフィギュレーションコマンドをネットワーク内のすべてのデバイスにロードして、すべてのデバイスのコンフィギュレーションを同様にするため。

コマンドラインにコマンドを入力した場合と同様に、`copy {ftp|rcp:|tftp:system:running-config} EXEC` コマンドはデバイスにコンフィギュレーションファイルをロードします。コマンドを追加するまで、既存の実行コンフィギュレーションは消去されません。コピーされたコンフィギュレーションファイル内のコマンドによって既存のコンフィギュレーションファイル内のコマンドが置き換えられると、既存のコマンドは消去されます。たとえば、コピーされたコンフィギュレーションファイルに格納されている特定のコマンドの IP アドレスが、既存のコンフィギュレーションに格納されている IP アドレスと異なる場合は、コピーされたコンフィギュレーション内の IP アドレスが使用されます。ただし、既存のコンフィギュレーション内の一部のコマンドには、置き換えられたり無効になったりしないものもあります。このようなコマンドがある場合は、既存のコンフィギュレーションファイルとコピーされたコンフィギュレーションファイルが組み合わされた（コピーされたコンフィギュレーションファイルが優先する）コンフィギュレーションファイルが作成されます。

コンフィギュレーションファイルをサーバ上に格納されているファイルの正確なコピーとして復元するには、その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直接コピーし（`copy ftp:|rcp:|tftp:} nvram:startup-config` コマンドを使用）、デバイスをリロードする必要があります。

サーバからデバイスへコンフィギュレーションファイルをコピーするには、次のセクションで説明するタスクを実行します。

使用するプロトコルは、使用中のサーバのタイプに応じて異なります。FTP および `rcp` のトランスポートメカニズムは、TFTP よりも高速でデータ配信の信頼性も優れています。これらの改善は、FTP および `rcp` のトランスポートメカニズムがコネクション型の TCP/IP スタック上に構築されており、これを使用しているために可能になりました。

デバイスから TFTP サーバへのコンフィギュレーションファイルのコピー

一部の TFTP 実装では、TFTP サーバ上にダミーファイルを作成し、読み取り、書き込み、および実行を許可してから、ダミーファイルを上書きする形でファイルをコピーする必要があります。詳細については、ご使用の TFTP のマニュアルを参照してください。

デバイスから RCP サーバへのコンフィギュレーションファイルのコピー

デバイスから RCP サーバへコンフィギュレーションファイルをコピーできます。

ネットワークを UNIX コミュニティでリソースとして使用する最初の試みの 1 つは、リモートシェル (RSH) およびリモートコピー (rcp) 機能が含まれた、リモートシェルプロトコルの設計および実装につながりました。rsh および rcp により、ユーザはリモートでコマンドを実行し、ネットワーク上のリモートホストまたはサーバにあるファイルシステムからまたはファイルシステムへファイルをコピーすることが可能になります。シスコの rsh および rcp 実装は、標準実装と相互運用できます。

RCP の **copy** コマンドは、リモートシステム上の rsh サーバ (またはデーモン) を利用します。rcp を使用してファイルをコピーするために、TFTP のようにファイル配布用のサーバを作成する必要はありません。必要なのは、リモートシェル (rsh) をサポートするサーバへのアクセスだけです (ほとんどの UNIX システムが rsh をサポートしています)。ある場所から別の場所にファイルをコピーするため、コピー元のファイルに対する読み取り権限とコピー先のファイルに対する書き込み権限が必要です。コピー先ファイルが存在しない場合は、rcp により作成されます。

シスコの rcp 実装は UNIX の rcp 実装 (ネットワーク上のシステム間でファイルをコピー) の関数をエミュレートしたのですが、シスコのコマンド構文は UNIX の rcp コマンド構文とは異なります。シスコの rcp サポートは、rcp をトランスポートメカニズムとして使用する一連の **copy** コマンドを提供しています。これらの **rcp copy** コマンドは、シスコの TFTP **copy** コマンドに類似していますが、高速で信頼性の高いデータ配信を実現する代替方法を備えているという点が異なります。これらの改善は、rcp のトランスポートメカニズムがコネクション型の TCP/IP スタック上に構築されており、これを使用しているために可能になりました。rcp コマンドを使用して、デバイスからネットワークサーバ (またはその逆) へシステムイメージおよびコンフィギュレーションファイルをコピーできます。

また、rcp サポートをイネーブルにし、リモートシステムのユーザがデバイスからまたはデバイスへファイルをコピーできるようにすることも可能です。

リモートユーザがデバイスとの間でファイルをコピーできるように Cisco IOS ソフトウェアを設定するには、**ip rcmd rcp-enable** グローバルコンフィギュレーションコマンドを使用します。

機能制限

RCP プロトコルでは、クライアントは RCP 要求ごとにリモートユーザ名をサーバに送信する必要があります。RCP を使用してデバイスからサーバへコンフィギュレーションファイルをコピーする場合、Cisco IOS ソフトウェアは次の順番で最初に発見した有効なユーザ名を送信します。

1. **copy EXEC** コマンドで指定されたユーザ名 (ユーザ名が指定されている場合)。
2. **ip rcmd remote-username** グローバルコンフィギュレーションコマンドで設定されたユーザ名 (コマンドが設定されている場合)。
3. 現在の TTY (端末) プロセスに関連付けられているリモートユーザ名。たとえば、ユーザが Telnet を介してデバイスに接続されており、**username** コマンドを介して認証された場

合は、リモートユーザ名として Telnet ユーザ名がデバイスソフトウェアによって送信されます。

4. デバイスのホスト名。

RCP コピー要求を正常に実行するためには、ネットワーク サーバ上にリモートユーザ名のアカウントを定義する必要があります。このサーバがディレクトリ構造をとっている場合、コンフィギュレーション ファイルまたはイメージは、サーバ上のリモートユーザ名と関連付けられたディレクトリに書き込まれるか、そのディレクトリからコピーされます。たとえば、システムイメージがサーバ上のユーザのホーム ディレクトリにある場合は、そのユーザの名前をリモートユーザ名として指定できます。

ip rcmd remote-username コマンドを使用して、すべてのコピーに対してユーザ名を指定します。(rcmd は、スーパーユーザ レベルで使用される UNIX ルーチンで、予約されたポート番号に基づいた認証スキームを使用してリモートマシン上でコマンドを実行します。rcmd は「Remote Command (リモート コマンド)」の略です)。特定のコピー操作にのみ使用するユーザ名を指定する場合は、**copy** コマンド内でユーザ名を指定します。

サーバに書き込む場合、デバイス上のユーザからの RCP 書き込み要求を受け入れるように、RCP サーバを適切に設定する必要があります。UNIX システムの場合は、RCP サーバ上のリモートユーザ用の .rhosts ファイルにエントリを追加する必要があります。たとえば、デバイスに次の設定行が含まれているとします。

```
hostname Device1
ip rcmd remote-username User0
```

デバイスの IP アドレスが device1.example.com に変換される場合、RCP サーバ上の User0 の .rhosts ファイルには、次の行が含まれることとなります。

```
Device1.example.com Device1
```

RCP ユーザ名に関する要件

RCP プロトコルでは、クライアントは RCP 要求ごとにリモートユーザ名をサーバに送信する必要があります。RCP を使用してデバイスからサーバへコンフィギュレーション ファイルをコピーする場合、Cisco IOS ソフトウェアは次の順番で最初に発見した有効なユーザ名を送信します。

1. **copy EXEC** コマンドで指定されたユーザ名 (ユーザ名が指定されている場合)。
2. **ip rcmd remote-username** グローバルコンフィギュレーションコマンドで設定されたユーザ名 (コマンドが設定されている場合)。
3. 現在の TTY (端末) プロセスに関連付けられているリモートユーザ名。たとえば、ユーザが Telnet を介してデバイスに接続されており、**username** コマンドを介して認証された場合は、リモートユーザ名として Telnet ユーザ名がデバイスソフトウェアによって送信されます。
4. デバイスのホスト名。

RCP コピー要求を実行するためには、ネットワーク サーバ上にリモート ユーザ名のアカウントを定義する必要があります。このサーバがディレクトリ構造をとっている場合、コンフィギュレーション ファイルまたはイメージは、サーバ上のリモート ユーザ名と関連付けられたディレクトリに書き込まれるか、そのディレクトリからコピーされます。たとえば、システム イメージがサーバ上のユーザのホーム ディレクトリにある場合は、そのユーザの名前をリモート ユーザ名として指定します。

詳細については、ご使用の RCP サーバのマニュアルを参照してください。

デバイスから FTP サーバへのコンフィギュレーション ファイルのコピー

デバイスから FTP サーバにコンフィギュレーション ファイルをコピーできます。

FTP ユーザ名およびパスワードの概要



(注) パスワードに特殊文字「@」を含めることはできません。文字「@」を使用すると、コピーでサーバの IP アドレスを解析できません。

FTP プロトコルでは、FTP 要求ごとにリモート ユーザ名およびパスワードを、クライアントがサーバに送信する必要があります。FTP を使用してデバイスからサーバへコンフィギュレーション ファイルをコピーする場合、Cisco IOS ソフトウェアは次の順番で最初に発見した有効なユーザ名を送信します。

1. **copy EXEC** コマンドで指定されたユーザ名（ユーザ名が指定されている場合）。
2. **ip ftp username** グローバル コンフィギュレーション コマンドで設定されたユーザ名（コマンドが設定されている場合）。
3. Anonymous

デバイスは、次の順番で最初に発見した有効なパスワードを送信します。

1. **copy** コマンドで指定されたパスワード（パスワードが指定されている場合）。
2. **ip ftp password** コマンドで設定されたパスワード（コマンドが設定されている場合）。
3. デバイスは、*username@devicename.domain* というパスワードを生成します。変数 *username* は現在のセッションに関連付けられたユーザ名、*devicename* は設定済みのホスト名、*domain* はデバイスのドメインです。

ユーザ名およびパスワードは、FTP サーバのアカウントに関連付けられている必要があります。サーバに書き込む場合、デバイス上のユーザからの FTP 書き込み要求を受け入れるように、FTP サーバを適切に設定する必要があります。

このサーバがディレクトリ構造をとっている場合、コンフィギュレーション ファイルまたはイメージは、サーバ上のユーザ名と関連付けられたディレクトリに書き込まれるか、そのディレクトリからコピーされます。たとえば、システム イメージがサーバ上のユーザのホーム ディレクトリにある場合は、そのユーザの名前をリモート ユーザ名として指定します。

詳細については、ご使用の FTP サーバのマニュアルを参照してください。

すべてのコピー操作に使用するユーザ名およびパスワードを指定するには、**ip ftp username** および **ip ftp password** グローバルコンフィギュレーションコマンドを使用します。特定のコピー操作にのみ使用するユーザ名を指定する場合は、**copy EXEC** コマンド内でユーザ名を指定します。

VRFによるファイルのコピー

copy コマンドで指定した VRF インターフェイス経由でファイルをコピーできます。設定の変更リクエストを使用せずに直接送信元インターフェイスを変更できるので、**copy** コマンドで VRF を指定するほうが簡単で効率的です。

例

次の例に、**copy** コマンドを使用して VRF 経由でファイルをコピーする方法を示します。

```
Device#
Address or name of remote host [10.1.1.2.3]?
Source username [ScpUser]?
Source filename [/auto/tftp-server/ScpUser/vrf_test.txt]?
Destination filename [vrf_test.txt]?
Getting the vrf name as test-vrf
Password:
Sending file modes: C0644 10 vrf_test.txt
!
223 bytes copied in 22.740 secs (10 bytes/sec)
```

スイッチから別のスイッチへのコンフィギュレーションファイルのコピー

あるスイッチから別のスイッチに設定をコピーすることができます。これは2ステッププロセスです。スイッチから TFTP サーバに設定をコピーし、次に TFTP から別のスイッチに設定をコピーします。

スイッチから現在の設定をコピーするには、**copy startup-config tftp:** コマンドを実行し、続く指示に従います。設定が TFTP サーバにコピーされます。

次に、別のスイッチへログインし、**copy tftp: startup-config** コマンドを実行して、続く指示に従います。これで、設定は別のスイッチにコピーされます。

設定をコピーした後、その設定を保存するには、**write memory** コマンドを使用し、その後スイッチをリロードするか、または **copy startup-config running-config** コマンドを実行します。

NVRAM より大きいコンフィギュレーションファイル

NVRAM より大きいコンフィギュレーションファイルを維持管理するには、以降の項の情報を知っておく必要があります。

コンフィギュレーション ファイルの圧縮

service compress-config グローバル コンフィギュレーション コマンドは、コンフィギュレーション ファイルを圧縮して NVRAM に格納することを指定します。コンフィギュレーション ファイルが圧縮されると、デバイスは正常に機能します。システムの起動時に、システムはコンフィギュレーション ファイルが圧縮されていることを認識し、圧縮されたコンフィギュレーション ファイルを展開して、正常に処理を進めます。 **more nvram:startup-config EXEC** コマンドにより、コンフィギュレーション が展開されてから表示されます。

コンフィギュレーション ファイルを圧縮する前に、適切なハードウェアのインストールおよびメンテナンス マニュアルを参照してください。ご利用のシステムの ROM がファイル圧縮をサポートしていることを確認します。サポートしていない場合、ファイル圧縮をサポートしている新しい ROM をインストールできます。

コンフィギュレーション のサイズは、NVRAM のサイズの 3 倍を超えてはいけません。NVRAM のサイズが 128 KB の場合、展開できる最大のコンフィギュレーション ファイルのサイズは 384 KB です。

service compress-config グローバル コンフィギュレーション コマンドは、Cisco IOS ソフトウェア リリース 10.0 以降のブート ROM を使用している場合に限り実行できます。新しい ROM をインストールするのは 1 回限りの操作で、ROM に Cisco IOS Release 10.0 がいない場合だけ必要です。ブート ROM が圧縮コンフィギュレーション を認識しない場合は、次のメッセージが表示されます。

```
Boot ROMs do not support NVRAM compression Config NOT written to NVRAM
```

コンフィギュレーション のクラス A フラッシュ ファイル システム上のフラッシュ メモリへの格納

クラス A フラッシュ ファイル システムのデバイス上では、内部フラッシュ メモリのファイルまたは PCMCIA スロットのフラッシュ メモリのファイルに **CONFIG_FILE** 環境変数を設定することにより、スタートアップ コンフィギュレーション をフラッシュ メモリに格納できます。

詳細については、[クラス A フラッシュ ファイル システムでの CONFIG_FILE 環境変数の指定 \(225 ページ\)](#) を参照してください。

大きいコンフィギュレーション を編集または変更する場合は、注意する必要があります。フラッシュ メモリ領域は **copy system:running-config nvram:startup-config EXEC** コマンドが発行されるたびに使用されます。フラッシュ メモリのファイル管理 (空き領域の最適化などの) は自動的にには行われなため、利用可能なフラッシュ メモリに十分注意を払う必要があります。 **squeeze** コマンドを使用して、使用済み領域を再要求します。20 MB 以上の大容量フラッシュ カードを使用することを推奨します。

ネットワークからのコンフィギュレーション コマンドのロード

コンフィギュレーション が大きい場合は、FTP、RCP、TFTP のいずれかのサーバに格納しておき、システムの起動時にダウンロードすることもできます。ネットワーク サーバを使用して大規模な設定を格納するには、[デバイスから TFTP サーバへのコンフィギュレーション ファイルのコピー \(203 ページ\)](#) および [コンフィギュレーション ファイルをダウンロードするデバイスの設定 \(200 ページ\)](#) の項でこれらのコマンドの詳細を参照してください。

コンフィギュレーションファイルをダウンロードするデバイスの設定

システムの起動時に1つまたは2つのコンフィギュレーションファイルをロードするようにデバイスを設定できます。コンフィギュレーションファイルは、コマンドラインにコマンドを入力した場合と同様に、メモリにロードされ読み込まれます。そのため、デバイスのコンフィギュレーションは、元のスタートアップコンフィギュレーションと1つまたは2つのダウンロードされたコンフィギュレーションファイルが混在したものになります。

ネットワークとホストのコンフィギュレーションファイル

歴史的な理由から、デバイスが最初にダウンロードするファイルは、ネットワークコンフィギュレーションファイルと呼ばれます。デバイスが2番目にダウンロードするファイルは、ホストコンフィギュレーションファイルと呼ばれます。2つのコンフィギュレーションファイルは、ネットワーク上のすべてのデバイスが、同一コマンドの多くを使用する場合に使用できます。ネットワークコンフィギュレーションファイルには、すべてのデバイスを設定するために使用される標準コマンドが含まれます。ホストコンフィギュレーションファイルには、特定の1つのホストに固有のコマンドが含まれます。2つのコンフィギュレーションファイルをロードする場合、ホストコンフィギュレーションファイルを、もう1つのファイルより優先させる必要があります。ネットワークコンフィギュレーションファイルとホストコンフィギュレーションファイルの両方とも、TFTP、RCP、FTPのいずれかを介して到達可能なネットワークサーバ上にあり、読み取り可能である必要があります。

コンフィギュレーションファイル情報の管理方法

コンフィギュレーションファイル情報の表示

コンフィギュレーションファイルに関する情報を表示するには、このセクションの手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show boot 例： Device# show boot	BOOT環境変数の内容（設定されている場合）、CONFIG_FILE環境変数によって指定されているコンフィギュレーションファイルの名前、およびBOOTLDR環境変数の内容を示します。

	コマンドまたはアクション	目的
ステップ 3	more <i>file-url</i> 例 : Device# more 10.1.1.1	指定されたファイルの内容を表示します。
ステップ 4	show running-config 例 : Device# show running-config	実行コンフィギュレーション ファイルの内容を表示します (more system:running-config コマンドのコマンドエイリアスです) 。
ステップ 5	show startup-config 例 : Device# show startup-config	スタートアップ コンフィギュレーション ファイルの内容を表示します。 (more nvram:startup-config コマンドのコマンドエイリアスです) 。 クラス A フラッシュ ファイル システム プラットフォーム以外のすべてのプラットフォーム上では、通常、デフォルトの startup-config ファイルは NVRAM に格納されます。 クラス A フラッシュ ファイル システム プラットフォーム上では、 CONFIG_FILE 環境変数はデフォルトの startup-config ファイルを指定します。 CONFIG_FILE 変数のデフォルトは NVRAM になります。

コンフィギュレーション ファイルの変更

Cisco IOS ソフトウェアは、1 行につき 1 つのコンフィギュレーション コマンドを受け入れます。コンフィギュレーション コマンドは、必要なだけ入力できます。コンフィギュレーション ファイルには、入力したコマンドを説明するコメントを追加できます。コメントの先頭には、感嘆符 (!) を付けます。コメントは NVRAM にもコンフィギュレーション ファイルのアクティブコピーにも格納されないため、**show running-config** または **more system:running-config EXEC** コマンドでアクティブな設定のリストを表示しても、コメントは表示されません。**show startup-config** または **more nvram:startup-config EXEC** モードコマンドでスタートアップ コンフィギュレーションのリストを表示しても、コメントは表示されません。コメントは、コンフィギュレーション ファイルがデバイスにロードされたときにコンフィギュレーション ファイルから削除されます。ただし、ファイル転送プロトコル (FTP)、リモートコピープロトコル (RCP)、または Trivial File Transfer Protocol (TFTP) サーバ上に格納されているコンフィギュレーション ファイルのコメントのリストは表示できます。CLI を使用してソフトウェアは設定するときは、ユーザの入力に従ってソフトウェアによりコマンドが実行されます。CLI を

使用してソフトウェアを設定するには、特権 EXEC モードを開始して次の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	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	configuration command 例： Device(config)# configuration command	必要なコンフィギュレーション コマンドを入力します。Cisco IOS マニュアル セットに、テクノロジー別に編成されたコンフィギュレーション コマンドが説明されています。
ステップ 4	次のいずれかを実行します。 • end • ^Z 例： Device(config)# end	コンフィギュレーション セッションを終了し、EXEC モードに戻ります。 (注) Ctrl キーと Z キーを同時に押すと、画面に ^Z と表示されます。
ステップ 5	copy system:running-config nvram:startup-config 例： Device# copy system:running-config nvram:startup-config	実行コンフィギュレーション ファイルをスタートアップコンフィギュレーション ファイルとして保存します。 copy running-config startup-config コマンドエイリアスも使用できますが、このコマンドは精度が高くないため、注意する必要があります。ほとんどのプラットフォーム上では、このコマンドによりコンフィギュレーションは NVRAM に保存されます。クラス A フラッシュ ファイル システムのプラットフォーム上では、この手順によりコンフィギュレーションは CONFIG_FILE 環境変数によって指定された場所に保存されます（デフォルトの CONFIG_FILE 変数では、

	コマンドまたはアクション	目的
		ファイルの保存先は NVRAM に指定されています)。

例

次の例では、デバイスのデバイスプロンプト名を設定しています。感嘆符 (!) で示されたコメント行では、いずれのコマンドも実行されません。**hostname** コマンドを使用して、デバイス名を device から new_name に変更しています。Ctrl+Z (^Z) キーを押すか、**end** コマンドを入力すると、コンフィギュレーションモードが終了します。**copy system:running-config nvram:startup-config** コマンドにより、現在のコンフィギュレーションがスタートアップ コンフィギュレーションに保存されます。

```
Device# configure terminal
Device(config)# !The following command provides the switch host name.
Device(config)# hostname new_name
new_name(config)# end
new_name# copy system:running-config nvram:startup-config
```

スタートアップ コンフィギュレーションが NVRAM にある場合は、現在の設定情報がコンフィギュレーション コマンドとしてテキスト形式で格納され、デフォルト以外の設定だけが記録されます。破損データから保護するために、メモリはチェックサム算出されます。



(注) 一部の特定のコマンドは、NVRAM に保存されない場合があります。これらのコマンドは、マシンをリブートしたときに再入力する必要があります。これらのコマンドは、マニュアルに記載されています。リブート後にすばやくデバイスを再設定できるように、これらの設定のリストを保管しておくことを推奨します。

デバイスから TFTP サーバへのコンフィギュレーション ファイルのコピー

TFTP ネットワーク サーバ上の設定をコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
ステップ 2	copy system:running-config tftp: [[[//location]/directory]/filename] 例 : Device# copy system:running-config tftp: //server1/topdir/file10	TFTP サーバへ実行コンフィギュレーションファイルをコピーします。
ステップ 3	copy nvram:startup-config tftp: [[[//location]/directory]/filename] 例 : Device# copy nvram:startup-config tftp: //server1/1stidir/file10	TFTPサーバへスタートアップコンフィギュレーションファイルをコピーします。

例

次に、デバイスから TFTP サーバへコンフィギュレーションファイルをコピーする例を示します。

```
Device# copy system:running-config tftp://172.16.2.155/tokyo-config
Write file tokyo-config on host 172.16.2.155? [confirm] Y
Writing tokyo-config!!! [OK]
```

次の作業

copy コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

デバイスから RCP サーバへのコンフィギュレーションファイルのコピー

デバイスから RCP サーバへスタートアップ コンフィギュレーションファイルまたは実行コンフィギュレーションファイルをコピーするには、特権 EXEC モードを開始して次の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	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip rcmd remote-username username 例 : Device(config)# ip rcmd remote-username NetAdmin1	(任意) デフォルトのリモート ユーザ名を変更します。
ステップ 4	end 例 : Device(config)# end	(任意) 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 5	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy system:running-config rcp: [[[/[username@]location]/directory]/filename] • copy nvram:startup-config rcp: [[[/[username@]location]/directory]/filename] 例 : Device# copy system:running-config rcp://NetAdmin1@example.com/dir-files/file1	<ul style="list-style-type: none"> • デバイスの実行コンフィギュレーションファイルが RCP サーバ上に格納されるように指定します。 または • デバイスのスタートアップコンフィギュレーションファイルが RCP サーバ上に格納されるように指定します。

例

RCP サーバへの実行コンフィギュレーション ファイルの格納

次に、rtr2-config という名前の実行コンフィギュレーションファイルを IP アドレス 172.16.101.101 のリモート ホスト上の netadmin1 ディレクトリにコピーする例を示します。

```
Device# copy system:running-config rcp://netadmin1@172.16.101.101/runfile2-config
Write file runfile2-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

RCP サーバへのスタートアップコンフィギュレーション ファイルの格納

次に、RCP を使用してファイルをコピーすることによって、サーバ上にスタートアップコンフィギュレーション ファイルを格納する例を示します。

```

Device# configure terminal

Device(config)# ip rcmd remote-username netadmin2

Device(config)# end

Device# copy nvram:startup-config rcp:

Remote host[ ]? 172.16.101.101

Name of configuration file to write [start-config]?
Write file start-config on host 172.16.101.101?[confirm]
![OK]
    
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

デバイスから FTP サーバへのコンフィギュレーション ファイルのコピー

デバイスから FTP サーバへスタートアップ コンフィギュレーション ファイルまたは実行コンフィギュレーション ファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	デバイスでグローバルコンフィギュレーション モードを開始します。
ステップ 3	ip ftp username <i>username</i> 例： Device(config)# ip ftp username NetAdmin1	(任意) デフォルトのリモートユーザ名を指定します。
ステップ 4	ip ftp password <i>password</i> 例：	(任意) デフォルトのパスワードを指定します。

	コマンドまたはアクション	目的
	Device(config)# ip ftp password adminpassword	
ステップ 5	end 例 : Device(config)# end	(任意) 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。この手順は、デフォルトのリモート ユーザ名またはパスワードを上書きする場合にだけ必要です (ステップ 2 および 3 を参照)。
ステップ 6	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy system:running-config ftp: [[[/[username [:password]@]location]/directory]/filename] または • copy nvram:startup-config ftp: [[[/[username [:password]@]location]/directory]/filename] 例 : Device# copy system:running-config ftp:	FTP サーバの指定された場所へ実行コンフィギュレーションまたはスタートアップ コンフィギュレーション ファイルをコピーします。

例

FTP サーバへの実行コンフィギュレーション ファイルの格納

次に、runfile-config という名前の実行コンフィギュレーション ファイルを IP アドレス 172.16.101.101 のリモート ホスト上の netadmin1 ディレクトリにコピーする例を示します。

```
Device# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/runfile-config
Write file runfile-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

FTP サーバへのスタートアップ コンフィギュレーション ファイルの格納

次に、FTP を使用してファイルをコピーすることによって、サーバ上にスタートアップ コンフィギュレーション ファイルを格納する例を示します。

```
Device# configure terminal
Device(config)# ip ftp username netadmin2
Device(config)# ip ftp password mypass
Device(config)# end
```

```
Device# copy nvram:startup-config ftp:

Remote host[]? 172.16.101.101

Name of configuration file to write [start-config]?
Write file start-config on host 172.16.101.101?[confirm]
![OK]
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

TFTP サーバからデバイスへのコンフィギュレーション ファイルのコピー

TFTP サーバからデバイスへコンフィギュレーション ファイルをコピーするには、以下のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy tftp: [[[//location]]/directory]/filename] system:running-config 例： Device# copy tftp://server1/dir10/datasource system:running-config	TFTP サーバから実行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。
ステップ 3	copy tftp: [[[//location]]/directory]/filename] nvram:startup-config 例： Device# copy tftp://server1/dir10/datasource nvram:startup-config	TFTP 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。
ステップ 4	copy tftp: [[[//location]]/directory]/filename] flash-nvram:directory/startup-config 例： Device# copy	TFTP 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。

	コマンドまたはアクション	目的
	tftp://server1/dir10/datasource flash:startup-config	

例

次に、IP アドレス 172.16.2.155 にある、**tokyo-config** という名前のファイルからソフトウェアを設定する例を示します。

```
Device# copy tftp://172.16.2.155/tokyo-config system:running-config
Configure using tokyo-config from 172.16.2.155? [confirm] Y
Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

rcpサーバからデバイスへのコンフィギュレーションファイルのコピー

rcp サーバから実行コンフィギュレーションまたはスタートアップ コンフィギュレーションへコンフィギュレーション ファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	（任意）端末からコンフィギュレーション モードを開始します。この手順は、デフォルトのリモートユーザ名を上書きする場合にだけ必要です（ステップ 3 を参照）。
ステップ 3	ip rcmd remote-username <i>username</i> 例： Device (config)# ip rcmd remote-username NetAdmin1	（任意）リモート ユーザ名を指定します。

	コマンドまたはアクション	目的
ステップ 4	end 例： Device(config)# end	(任意) 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。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です (ステップ 2 を参照)。
ステップ 5	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy <code>rcp[[user1@172.16.101.101]#]system:runningconf</code> • copy <code>rcp[[user1@172.16.101.101]#]nvram:startupconf</code> 例： Device# copy rcp://[user1@example.com/dir10/fileone] nvram:startup-config	rcp サーバから実行コンフィギュレーションまたは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。

例

rcp の Running-Config のコピー

次に、host1-config という名前のコンフィギュレーションファイルを、IP アドレスが 172.16.101.101 のリモートサーバ上の netadmin1 ディレクトリからコピーし、デバイスでコマンドをロードして実行する例を示します。

```
device# copy rcp://netadmin1@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:[OK]
device#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

rcp の Startup-Config のコピー

次に、リモートユーザ名 netadmin1 を指定する例を示します。次に host2-config という名前のコンフィギュレーションファイルを、IP アドレスが 172.16.101.101 のリモートサーバ上の netadmin1 ディレクトリからスタートアップコンフィギュレーションへコピーします。

```
device# configure terminal
device(config)# ip rcmd remote-username netadmin1
device(config)# end
device# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:[OK]
[OK]
```

```
device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcpc from 172.16.101.101
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

FTP サーバからデバイスへのコンフィギュレーション ファイルのコピー

FTP サーバから実行コンフィギュレーションまたは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 ファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	（任意）グローバル コンフィギュレーション モードを開始できます。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 3 および 4 を参照）。
ステップ 3	ip ftp username <i>username</i> 例： Device(config)# ip ftp username NetAdmin1	（任意）デフォルトのリモート ユーザ名を指定します。
ステップ 4	ip ftp password <i>password</i> 例： Device(config)# ip ftp password adminpassword	（任意）デフォルトのパスワードを指定します。
ステップ 5	end 例： Device(config)# end	（任意）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。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ

	コマンドまたはアクション	目的
		必要です（ステップ 3 および 4 を参照）。
ステップ 6	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> • copy ftp: [[[//[username[:password]@]location] /directory]/filename]system:running-config • copy ftp: [[[/username[:password]@]location]filename:startup-config <p>例 :</p> <pre>Device# copy ftp:nvram:startup-config</pre>	FTP を使用して、ネットワーク サーバから実行メモリまたはスタートアップ 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。

例

FTP の Running-Config のコピー

次に、host1-config という名前のホスト コンフィギュレーション ファイルを、IP アドレスが 172.16.101.101 のリモートサーバ上の netadmin1 ディレクトリからコピーし、デバイスでコマンドをロードして実行する例を示します。

```
device# copy ftp://netadmin1:mypass@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:[OK]
device#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

FTP の Startup-Config のコピー

次に、リモートユーザ名 netadmin1 を指定する例を示します。次に host2-config という名前のコンフィギュレーション ファイルを、IP アドレスが 172.16.101.101 のリモートサーバ上の netadmin1 ディレクトリからスタートアップ コンフィギュレーションへコピーします。

```
device# configure terminal
device(config)# ip ftp username netadmin1
device(config)# ip ftp password mypass
device(config)# end
device# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[host1-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:[OK]
[OK]
device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from 172.16.101.101
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

NVRAM より大きいコンフィギュレーションファイルの保守

NVRAMのサイズを超えるコンフィギュレーションファイルを保守するには、以降のセクションで説明するタスクを実行します。

コンフィギュレーションファイルの圧縮

コンフィギュレーションファイルを圧縮するには、このセクションの手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	service compress-config 例： Device(config)# service compress-config	コンフィギュレーションファイルを圧縮することを指定します。
ステップ 4	end 例： Device(config)# end	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 5	次のいずれかを実行します。 • 新しいコンフィギュレーションをコピーするには、FTP、RCP、TFTP を使用します。 • configure terminal 例：	新しいコンフィギュレーションを入力します。 • NVRAMのサイズの3倍以上のコンフィギュレーションをロードしようとすると、次のエラーメッセージが表示されます。

	コマンドまたはアクション	目的
	Device# <code>configure terminal</code>	「[buffer overflow - file-size /buffer-size bytes]。」
ステップ 6	copy system:running-config nvram:startup-config 例 : Device(config)# <code>copy system:running-config nvram:startup-config</code>	実行コンフィギュレーションの変更が終わったら、新しいコンフィギュレーションを保存します。

例

次に、129 KB のコンフィギュレーションファイル を 11 KB に圧縮する例を示します。

```
Device# configure terminal

Device(config)# service compress-config

Device(config)# end

Device# copy tftp://172.16.2.15/tokyo-config system:running-config

Configure using tokyo-config from 172.16.2.155? [confirm] y

Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
Device# copy system:running-config nvram:startup-config

Building configuration...
Compressing configuration from 129648 bytes to 11077 bytes
[OK]
```

コンフィギュレーションのクラス A フラッシュ ファイル システム上のフラッシュ メモリへの格納

スタートアップ コンフィギュレーションをフラッシュ メモリに格納するには、このセクションの手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> <code>enable</code>	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	<p>copy nvram:startup-config <i>flash-filesystem:filename</i></p> <p>例 :</p> <pre>Device# copy nvram:startup-config usbflash0:switch-config</pre>	新しい場所に現在のスタートアップ コンフィギュレーションをコピーして、コンフィギュレーション ファイルを作成します。
ステップ 3	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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 4	<p>boot config flash-filesystem: filename</p> <p>例 :</p> <pre>Device(config)# boot config usbflash0:switch-config</pre>	CONFIG_FILE 環境変数を設定することにより、フラッシュ メモリにスタートアップ コンフィギュレーション ファイルを格納することを指定します。
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config)# end</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 6	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> 新しいコンフィギュレーションをコピーするには、FTP、RCP、TFTP を使用します。NVRAM サイズの3倍を超える大きさのコンフィギュレーションをロードしようとする と、次のエラー メッセージが表示されます。「[buffer overflow - file-size /buffer-size bytes]」 configure terminal <p>例 :</p> <pre>Device# configure terminal</pre>	新しいコンフィギュレーションを入力します。
ステップ 7	<p>copy system:running-config nvram:startup-config</p> <p>例 :</p> <pre>Device(config)# copy system:running-config nvram:startup-config</pre>	実行コンフィギュレーションの変更が終わったら、新しいコンフィギュレーションを保存します。

例

以下に、usbflash0: に格納したコンフィギュレーションの例を示します。

```
Device# copy nvram:startup-config usbflash0:switch-config

Device# configure terminal

Device(config)# boot config usbflash0:switch-config

Device(config)# end

Device# copy system:running-config nvram:startup-config
```

ネットワークからのコンフィギュレーションコマンドのロード

ネットワーク サーバを使用して、大きなコンフィギュレーションを保存するには、このセクションの手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy system:running-config {ftp: rcp: tftp:} 例： Device# copy system:running-config ftp:	実行コンフィギュレーションを FTP、RCP、TFTP のいずれかのサーバに保存します。
ステップ 3	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 4	boot network {ftp:[[/[username[:password]@]location]/directory]/filename] rcp:[[/[username@]location]/directory]/filename] tftp:[[/[location]/directory]/filename]} 例： Device(config)# boot network ftp://user1:guessme@example.com/dir10/file1	起動時にスタートアップ コンフィギュレーション ファイルをネットワークサーバからロードすることを指定します。

	コマンドまたはアクション	目的
ステップ 5	service config 例 : Device(config)# service config	システムの起動時にコンフィギュレーションファイルをダウンロードするようにスイッチをイネーブルにします。
ステップ 6	end 例 : Device(config)# end	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 7	copy system:running-config nvram:startup-config 例 : Device# copy system:running-config nvram:startup-config	設定を保存します。

フラッシュメモリからスタートアップまたは実行コンフィギュレーションへのコンフィギュレーションファイルのコピー

フラッシュメモリから現在の NVRAM にあるスタートアップ コンフィギュレーションまたは実行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直接コピーするには、ステップ 2 のいずれかのコマンドを入力します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを実行します。 • copy filesystem: [partition-number:][filename] nvram:startup-config • copy filesystem: [partition-number:][filename] system:running-config 例 : Device# copy usbflash0:4:ios-upgrade-1 nvram:startup-config	• NVRAM にコンフィギュレーションファイルを直接ロードする、または • 現在の実行コンフィギュレーションにコンフィギュレーションファイルをコピーします。

例

次に、usbflash0 にあるフラッシュメモリ PC カードのパーティション 4 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へ ios-upgrade-1 という名前のファイルをコピーする例を示します。

```
Device# copy usbflash0:4:ios-upgrade-1 nvram:startup-config
Copy 'ios-upgrade-1' from flash device as 'startup-config' ? [yes/no] yes
[OK]
```

フラッシュメモリ ファイル システム間でのコンフィギュレーション ファイルのコピー

複数のフラッシュメモリファイルシステムを備えたプラットフォーム上では、内部フラッシュメモリなどのフラッシュメモリファイルシステムから他のフラッシュメモリファイルシステムへファイルをコピーできます。異なるフラッシュメモリファイルシステムへファイルをコピーすることで、使用中のコンフィギュレーションのバックアップコピーを作成し、他のデバイスにコンフィギュレーションを複製できます。フラッシュメモリファイルシステム間でコンフィギュレーションファイルをコピーするには、EXECモードで次の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	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none">パスワードを入力します（要求された場合）。
ステップ 2	show source-filesystem: 例： Device# show flash:	フラッシュメモリのレイアウトと内容を表示して、ファイル名を確認します。
ステップ 3	copy source-filesystem: [partition-number:][filename] dest-filesystem:[partition-number:][filename] 例： Device# copy flash: usbflash0:	フラッシュメモリ デバイス間でコンフィギュレーション ファイルをコピーします。 <ul style="list-style-type: none">コピー元デバイスとコピー先デバイスは同じにはできません。たとえば、copy usbflash0: usbflash0: コマンドが無効です。

例

次に、内部フラッシュメモリのパーティション 1 からデバイス上の `usbflash0` のパーティション 1 へ `running-config` という名前のファイルをコピーする例を示します。この例では、コピー元のパーティションが指定されていないため、デバイスからパーティション番号を要求されます。

```
Device# copy flash: usbflash0:

System flash
Partition  Size      Used       Free       Bank-Size  State      Copy Mode
  1          4096K    3070K     1025K     4096K      Read/Write Direct
  2          16384K   1671K    14712K    8192K      Read/Write Direct
[Type ?<no> for partition directory; ? for full directory; q to abort]
Which partition? [default = 1]
System flash directory, partition 1:
File Length  Name/status
  1  3142748  dirt/network/mars-test/c3600-j-mz.latest
  2   850    running-config
[3143728 bytes used, 1050576 available, 4194304 total]
usbflash0 flash directory:
File Length  Name/status
  1  1711088  dirt/gate/c3600-i-mz
  2   850    running-config
[1712068 bytes used, 2482236 available, 4194304 total]
Source file name? running-config

Destination file name [running-config]?
Verifying checksum for 'running-config' (file # 2)... OK
Erase flash device before writing? [confirm]
Flash contains files. Are you sure you want to erase? [confirm]
Copy 'running-config' from flash: device
as 'running-config' into usbflash0: device WITH erase? [yes/no] yes

Erasing device... eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
...erased!
[OK - 850/4194304 bytes]
Flash device copy took 00:00:30 [hh:mm:ss]
Verifying checksum... OK (0x16)
```

FTP サーバからフラッシュメモリデバイスへのコンフィギュレーションファイルのコピー

FTP サーバからフラッシュメモリデバイスへコンフィギュレーションファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例 :	特権 EXEC モードを有効にします。 <ul style="list-style-type: none">パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例： Device# configure terminal	(任意) 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。この手順は、デフォルトのリモート ユーザ名またはパスワードを上書きする場合にだけ必要です (ステップ 3 および 4 を参照)。
ステップ 3	ip ftp username <i>username</i> 例： Device(config)# ip ftp username Admin01	(任意) リモート ユーザ名を指定します。
ステップ 4	ip ftp password <i>password</i> 例： Device(config)# ip ftp password adminpassword	(任意) リモート パスワードを指定します。
ステップ 5	end 例： Device(config)# end	(任意) コンフィギュレーション モードを終了します。このステップが必要になるのは、デフォルトのリモート ユーザ名を上書きする場合のみです (ステップ 3 および 4 を参照)。
ステップ 6	copy ftp: [[//location]/directory]/bundle_name flash: 例： Device>copy ftp:/cat9k_iosxe.16.11.01.SPA.bin flash:	FTP を使用してネットワーク サーバからフラッシュ メモリ デバイスへコンフィギュレーション ファイルをコピーします。

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

RCPサーバからフラッシュメモリデバイスへのコンフィギュレーションファイルのコピー

RCP サーバからフラッシュメモリ デバイスへコンフィギュレーションファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	（任意）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。この手順は、デフォルトのリモート ユーザ名またはパスワードを上書きする場合にのみ必要です（ステップ 3 を参照）。
ステップ 3	ip rcmd remote-username <i>username</i> 例： Device(config)# ip rcmd remote-username Admin01	（任意）リモート ユーザ名を指定します。
ステップ 4	end 例： Device(config)# end	（任意）コンフィギュレーション モードを終了します。この手順は、デフォルトのリモート ユーザ名またはパスワードを上書きする場合にのみ必要です（ステップ 3 を参照）。
ステップ 5	copy rcp: [[[//[<i>username@</i>]<i>location</i>]/<i>directory</i>]/<i>bundle_name</i>] flash: 例： Device# copy rcp://netadmin@172.16.101.101/bundle1 flash:	RCP を使用してネットワーク サーバからフラッシュメモリ デバイスへコンフィギュレーションファイルをコピーします。追加情報または確認を要求するデバイスからのプロンプトに対し応答します。このプロンプトは、 copy コマンドで入力した情報量および file prompt コマンドの現在の設定によって異なります。

TFTPサーバからフラッシュメモリデバイスへのコンフィギュレーションファイルのコピー

TFTP サーバからフラッシュメモリデバイスへコンフィギュレーションファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy tftp: [[[/location]/directory]/bundle_name flash: 例： Device# copy tftp://192.168.1.100/switch-config flash:	TFTP サーバからフラッシュメモリデバイスへファイルをコピーします。追加情報または確認を要求するデバイスからのプロンプトに対し応答します。このプロンプトは、 copy コマンドで入力した情報量および file prompt コマンドの現在の設定によって異なります。

例

次に、TFTP サーバから `usbflash0` に挿入されているフラッシュメモリカードへ、`switch-config` という名前のコンフィギュレーションファイルをコピーする例を示します。コピーされたファイルの名前は `new-config` に変更されます。

```
Device#
copy tftp:switch-config usbflash0:new-config
```

スタートアップコンフィギュレーションファイルでのコンフィギュレーションコマンドの再実行

スタートアップコンフィギュレーションファイルのコマンドを再実行するには、このセクションの手順を実行します。

手順

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

	コマンドまたはアクション	目的
	Device> enable	<ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure memory 例： Device# configure memory	スタートアップ コンフィギュレーション ファイルでコンフィギュレーション コマンドを再実行します。

スタートアップ コンフィギュレーションのクリア

スタートアップ コンフィギュレーションから設定情報を消去できます。デバイスをスタートアップ コンフィギュレーションなしで再起動した場合は、デバイスを最初から設定できるように、デバイスは、**Setup** コマンドファシリティに移行します。スタートアップ コンフィギュレーションの内容をクリアするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	erase nvram 例：	スタートアップ コンフィギュレーションの内容をクリアします。

	コマンドまたはアクション	目的
	<pre>Device# erase nvram</pre>	<p>(注) クラス A フラッシュファイルシステムのプラットフォーム以外のすべてのプラットフォームでは、このコマンドにより NVRAM が消去されます。スタートアップコンフィギュレーションファイルは、いったん削除すると復元できません。クラス A フラッシュファイルシステムのプラットフォーム上では、erase startup-configEXEC コマンドを使用すると、CONFIG_FILE 環境変数により指定されたコンフィギュレーションが、デバイスにより削除されます。この変数が NVRAM を指定している場合は、デバイスにより NVRAM が消去されます。CONFIG_FILE 環境変数がフラッシュメモリデバイスとコンフィギュレーションファイル名を指定している場合は、デバイスによりコンフィギュレーションファイルが削除されます。つまり、そのコンフィギュレーションファイルはデバイスにより消去されるのではなく、「削除済み」としてマークされます。この機能では、削除されたファイルを回復できます。</p>

指定されたコンフィギュレーションファイルの削除

特定のフラッシュデバイスの指定された設定を削除するには、このセクションの手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>delete flash-filesystem:filename</p> <p>例 :</p> <pre>Device# delete usbflash0:myconfig</pre>	<p>特定のフラッシュ デバイス上の指定されたコンフィギュレーション ファイルを削除します。</p> <p>(注) クラス A および B フラッシュ ファイルシステムでは、フラッシュメモリ内の特定のファイルを削除すると、そのファイルは削除済みとしてシステムによりマークされます。これにより、undelete EXEC コマンドを使用して、削除したファイルを後で回復できるようになります。消去されたファイルは回復できません。コンフィギュレーション ファイルを完全に消去するには、squeeze EXEC コマンドを使用します。クラス C フラッシュファイルシステムでは、削除されたファイルは回復できません。CONFIG_FILE 環境変数で指定されたコンフィギュレーション ファイルを消去または削除しようとした場合、システムにより削除の確認を求めるプロンプトが表示されます。</p>

クラス A フラッシュ ファイル システムでの CONFIG_FILE 環境変数の指定

クラス A フラッシュ ファイル システムでは、CONFIG_FILE 環境変数で指定されたスタートアップコンフィギュレーションファイルを読み込むように Cisco IOS ソフトウェアを設定できます。CONFIG_FILE 変数のデフォルトは NVRAM になります。CONFIG_FILE 環境変数を変更するには、このセクションの手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy [flash-url ftp-url rcp-url tftp-url system:running-config nvram:startup-config] dest-flash-url 例： Device# copy system:running-config nvram:startup-config	フラッシュファイルシステムにコンフィギュレーションファイルをコピーします。再起動時には、ここからデバイスにファイルがロードされます。
ステップ 3	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 4	boot config dest-flash-url 例： Device(config)# boot config 172.16.1.1	CONFIG_FILE 環境変数を設定します。この手順により、実行時の CONFIG_FILE 環境変数が変更されます。
ステップ 5	end 例： Device(config)# end	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 6	copy system:running-config nvram:startup-config 例： Device# copy system:running-config nvram:startup-config	スタートアップ コンフィギュレーションにステップ 3 で実行されたコンフィギュレーションを保存します。
ステップ 7	show boot 例： Device# show boot	（任意）CONFIG_FILE 環境変数の内容を確認できます。

例

次の例は、実行コンフィギュレーション ファイルをデバイスにコピーします。その後、システムが再起動されるとこのコンフィギュレーションがスタートアップ コンフィギュレーションとして使用されます。

```
Device# copy system:running-config usbflash0:config2
Device# configure terminal
Device(config)# boot config usbflash0:config2
Device(config)# end
Device# copy system:running-config nvram:startup-config
[ok]
Device# show boot
BOOT variable = usbflash0:rsp-boot-m
CONFIG_FILE variable = nvram:
Current CONFIG_FILE variable = usbflash0:config2
Configuration register is 0x010F
```

次の作業

スタートアップ コンフィギュレーション ファイルの場所を指定すると、**nvram:startup-config** コマンドは、スタートアップ コンフィギュレーション ファイルの新しい場所のエイリアスとなります。**more nvram:startup-config EXEC** コマンドにより、スタートアップ コンフィギュレーションの場所に関係なく、スタートアップ コンフィギュレーションが表示されます。**erase nvram:startup-config EXEC** コマンドにより、NVRAM の内容が消去され、CONFIG_FILE 環境変数で指定されたファイルが削除されます。

copy system:running-config nvram:startup-config コマンドを使用してコンフィギュレーションを保存した場合、デバイスによりコンフィギュレーション ファイルの完全バージョンは CONFIG_FILE 環境変数により指定された場所に保存され、抽出バージョンは NVRAM に保存されます。抽出バージョンとは、アクセスリスト情報を含まないバージョンです。NVRAM に完全バージョンのコンフィギュレーション ファイルが含まれている場合は、デバイスは完全バージョンを抽出バージョンで上書きすることを確認するプロンプトを表示します。NVRAM に抽出コンフィギュレーションが含まれている場合は、デバイスは確認のプロンプトを表示しないで NVRAM にある既存の抽出バージョンのコンフィギュレーション ファイルを上書きする処理を進めます。



- (注) フラッシュデバイスにあるファイルを CONFIG_FILE 環境変数として指定した場合、**copy system:running-config nvram:startup-config** コマンドでコンフィギュレーション ファイルを保存するたびに、古いコンフィギュレーション ファイルは「削除済み」とマークされ、新しいコンフィギュレーション ファイルがそのデバイスに保存されます。それでも古いコンフィギュレーション ファイルがメモリを使用するため、最終的にフラッシュメモリは一杯になります。**squeeze EXEC** コマンドを使用して古いコンフィギュレーション ファイルを完全に削除し、領域を解放してください。

コンフィギュレーションファイルをダウンロードするデバイスの設定

ネットワーク コンフィギュレーションおよびホスト コンフィギュレーション ファイル名の順序付きリストを指定できます。Cisco IOS XE ソフトウェアは、適切なネットワークまたはホスト コンフィギュレーション ファイルをロードするまで、このリストをスキャンします。

システムの起動時にコンフィギュレーションファイルをダウンロードするようにデバイスを設定するには、次のセクションで説明するタスクを少なくとも 1 つ実行します。

- [ネットワーク コンフィギュレーションファイルをダウンロードするデバイスの設定](#)
- [ホスト コンフィギュレーションファイルをダウンロードするデバイスの設定](#)

起動中にコンフィギュレーションファイルをロードできなかった場合、要求されたファイルがホストから提供されるまで、デバイスは 10 分ごと（デフォルト設定）に再試行します。試行が失敗するごとに、デバイスにより以下のメッセージがコンソール端末に表示されます。

```
Booting host-config... [timed out]
```

スタートアップ コンフィギュレーション ファイルになんらかの問題がある場合、またはコンフィギュレーション レジスタが NVRAM を無視するように設定されている場合は、デバイスは Setup コマンドファシリティに移行します。

ネットワーク コンフィギュレーション ファイルをダウンロードするデバイスの設定

起動時にサーバからネットワーク コンフィギュレーション ファイルをダウンロードするように Cisco IOS ソフトウェアを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	boot network {ftp:[[/[username [:password]@]location]/directory]/filename] rcp:[[/[username@]location]/directory]/filename] tftp:[[/[location]/directory]/filename]} 例：	起動時にダウンロードするネットワーク コンフィギュレーション ファイルおよ び使用されるプロトコル（TFTP、RCP、 または FTP）を指定します。 • ネットワーク コンフィギュレーシ ョン ファイル名を指定しない場合、

	コマンドまたはアクション	目的
	<pre>Device(config)# boot network tftp:hostfile1</pre>	<p>Cisco IOS ソフトウェアはデフォルトのファイル名の network-config を使用します。アドレスを省略した場合、デバイスはブロードキャストアドレスを使用します。</p> <ul style="list-style-type: none"> 複数のネットワーク コンフィギュレーション ファイルを指定できません。ソフトウェアは、ネットワーク コンフィギュレーション ファイルをロードできるまで、入力された順に試行します。この手順は、異なる設定情報を持つ、ネットワーク サーバ上にロードされるファイルを複数保持する場合に役立ちます。
ステップ 4	<p>service config</p> <p>例 :</p> <pre>Device(config)# service config</pre>	再起動時にネットワーク ファイルを自動的にロードするようにシステムをイネーブルにします。
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config)# end</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 6	<p>copy system:running-config nvram:startup-config</p> <p>例 :</p> <pre>Device# copy system:running-config nvram:startup-config</pre>	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 ファイルに保存します。

ホストコンフィギュレーション ファイルをダウンロードするデバイスの設定

起動時にサーバからホスト コンフィギュレーション ファイルをダウンロードするように Cisco IOS ソフトウェアを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>boot host {ftp:[[[[username [:password]@]location]/directory]/filename] rcp:[[[[username@]location]/directory]/filename] tftp:[[[[location]/directory]/filename] }</p> <p>例 :</p> <pre>Device(config)# boot host tftp:hostfile1</pre>	<p>起動時にダウンロードするホスト コンフィギュレーション ファイルおよび使用されるプロトコル (FTP、RCP、または TFTP) を指定します。</p> <ul style="list-style-type: none"> ホスト コンフィギュレーション ファイルの名前を指定しない場合、デバイスは、それ自身の名前を使用してホスト コンフィギュレーション ファイル名を形成します。このとき、その名前はすべて小文字に変換され、すべてのドメイン情報は削除され、「-config」が追加されます。ホスト名の情報を利用できない場合は、ソフトウェアはデフォルトのホスト コンフィギュレーション ファイル名の device-config を使用します。アドレスを省略した場合、デバイスはブロードキャストアドレスを使用します。 複数のホストコンフィギュレーション ファイルを指定できます。Cisco IOS ソフトウェアは、ホスト コンフィギュレーション ファイルをロードできるまで、入力された順に試行します。この手順は、異なる設定情報を持つ、ネットワーク サーバ上にロードされるファイルを複数保持する場合に役立ちます。
ステップ 4	<p>service config</p> <p>例 :</p> <pre>Device(config)# service config</pre>	<p>再起動時にホスト ファイルを自動的にロードするようにシステムをイネーブルにします。</p>
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device (config)# end</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>

	コマンドまたはアクション	目的
ステップ 6	copy system:running-config nvram:startup-config 例 : Device# copy system:running-config nvram: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 ファイルに保存します。

例

次に、hostfile1 という名前のホストコンフィギュレーションファイルおよびnetworkfile1 という名前のネットワーク コンフィギュレーションファイルをダウンロードするようにデバイスを設定する例を示します。デバイスは TFTP およびブロードキャストアドレスを使用してファイルを取得します。

```
Device# configure terminal
Device(config)# boot host tftp:hostfile1
Device(config)# boot network tftp:networkfile1
Device(config)# service config
Device(config)# end
Device# copy system:running-config nvram:startup-config
```

コンフィギュレーション ファイルの管理の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	コンフィギュレーション ファイルの管理	コンフィギュレーション ファイルには、シスコ製デバイスの機能をカスタマイズするための Cisco IOS ソフトウェアコマンドが含まれています。コマンドは、システムを起動したとき (startup-config ファイルから)、または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 IOS ソフトウェアによって解析 (変換および実行) されます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 10 章

セキュアコピー

このドキュメントでは、セキュアコピー（SCP）サーバ側機能用にシスコデバイスを設定する手順について説明します。

- [セキュアコピーの前提条件](#)（233 ページ）
- [Secure Copy に関する情報](#)（233 ページ）
- [セキュアコピーの設定方法](#)（234 ページ）
- [セキュアコピーの設定方法](#)（237 ページ）
- [セキュアコピーに関する追加情報](#)（238 ページ）
- [セキュアコピーの機能情報](#)（238 ページ）

セキュアコピーの前提条件

- デバイス上でセキュアシェル（SSH）、認証、および許可を設定します。
- SCP は SSH を使用してセキュアな転送を実行するため、デバイスには RSA キーのペアが必要です。

Secure Copy に関する情報

Secure Copy Protocol（SCP）機能は、スイッチの設定やイメージファイルのコピーにセキュアな認証方式を提供します。SCP にはセキュアシェル（SSH）が必要です（Berkeley の r-tool に代わるセキュリティの高いアプリケーションおよびプロトコルです）。

SCP は一連の Berkeley の r ツール（Berkeley 大学独自のネットワークングアプリケーションセット）に基づいて設計されているため、その動作内容は Remote Copy Protocol（RCP）と類似しています。ただし、SCP は SSH のセキュリティに対応している点は除きます。加えて、SCP では、ユーザが正しい権限レベルを持っていることをデバイス上で判断できるように、認証、許可、およびアカウンティング（AAA）を設定する必要があります。

SCP を使用すると、**copy** コマンドを使用して Cisco IOS ファイルシステム（Cisco IFS）内の任意のファイルのコピーをデバイスとの間で実行できるのは、特権レベルが 15 のユーザのみになります。許可された管理者はワークステーションからこの操作を実行することもできます。



- (注)
- pscp.exe ファイルを使用している場合は、SCP オプションを有効にします。
 - SSH を機能させるには、RSA 公開キーと秘密キーのペアをデバイスで設定する必要があります。

セキュアコピーの設定方法

セキュアコピーの設定

シスコデバイスに SCP サーバ側機能の設定をするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	ログイン時の AAA 認証を設定します。
ステップ 4	aaa authentication login {default list-name} method1 [method2...] 例： Device(config)# aaa authentication login default group tacacs+	AAA アクセスコントロールシステムをイネーブルにします。
ステップ 5	username name [privilege level] password encryption-type encrypted-password 例：	ユーザ名をベースとした認証システムを構築します。

	コマンドまたはアクション	目的
	Device(config)# username superuser privilege 2 password 0 superpassword	(注) TACACS+ や RADIUS などのネットワークベースの認証メカニズムが設定されている場合は、この手順を省略できます。
ステップ 6	ip scp server enable 例 : Device(config)# ip scp server enable	SCP サーバ側機能を有効にします。
ステップ 7	exit 例 : Device(config)# exit	グローバル コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。
ステップ 8	debug ip scp 例 : Device# debug ip scp	(任意) SCP 認証問題を解決します。

SSH サーバでの Secure Copy Protocol のイネーブル化

次のタスクでは、SCP のサーバ側機能の設定方法を示します。このタスクは、デバイスでリモートのワークステーションからファイルを安全にコピーできる一般的な設定を示しています。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 3	aaa new-model 例 : Device(config)# aaa new-model	認証、許可、アカウントिंग (AAA) アクセス コントロール モデルをイネーブルにします。
ステップ 4	aaa authentication login default local 例 : Device(config)# aaa authentication login default local	ログイン時の認証にローカルのユーザ名データベースを使用するように AAA 認証を設定します。
ステップ 5	aaa authorization exec default local 例 : Device(config)# aaa authorization exec default local	ユーザアクセスを制限するパラメータをネットワークに設定します。許可を実行し、ユーザ ID で EXEC シェルの実行を許可するかどうかを定義します。その後、システムで許可にローカルデータベースを使用する必要があることを指定します。
ステップ 6	username name privilege privilege-level password password 例 : Device(config)# username samplename privilege 15 password password1	ユーザ名ベースの認証システムを確立し、ユーザ名、権限レベル、および非暗号化パスワードを指定します。 (注) <i>privilege-level</i> 引数の最小値は 15 です。権限レベルが 15 未満の場合、接続が切断されます。
ステップ 7	ip ssh time-out seconds 例 : Device(config)# ip ssh time-out 120	デバイスが SSH クライアントの応答を待つ時間間隔を、秒単位で設定します。
ステップ 8	ip ssh authentication-retries 整数 例 : Device(config)# ip ssh authentication-retries 3	インターフェイスのリセット後、認証を試行する回数を設定します。
ステップ 9	ip scp server enable 例 : Device(config)# ip scp server enable	デバイスで、リモートワークステーションから安全にファイルをコピーできるようにします。

	コマンドまたはアクション	目的
ステップ 10	ip ssh bulk-mode 例： Device(config)# ip ssh bulk-mode	(任意) SSH 一括データ転送モードをイネーブルにして、SCP のスループットパフォーマンスを強化します。
ステップ 11	exit 例： Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 12	debug ip scp 例： Device# debug ip scp	(任意) SCP 認証の問題に関する診断情報を提供します。

セキュアコピーの設定方法

例：ローカル認証を使用したセキュアコピーの設定

次の例は、SCP のサーバ側機能の設定方法を示しています。この例では、ローカルに定義されたユーザ名とパスワードを使用します。

```
! AAA authentication and authorization must be configured properly in order for SCP to work.
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login default local
Device(config)# aaa authorization exec default local
Device(config)# username user1 privilege 15 password 0 lab
! SSH must be configured and functioning properly.
Device(config)# ip scp server enable
Device(config)# end
```

例：ネットワークベース認証を使用した SCP サーバ側の設定

次の例は、ネットワークベースの認証メカニズムを使用した SCP のサーバ側機能の設定方法を示しています。

```
! AAA authentication and authorization must be configured properly for SCP to work.
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login default group tacacs+
Device(config)# aaa authorization exec default group tacacs+
```

```

! SSH must be configured and functioning properly.
Device(config)# ip ssh time-out 120
Device(config)# ip ssh authentication-retries 3
Device(config)# ip scp server enable
Device(config)# end

```

セキュアコピーに関する追加情報

関連資料

関連項目	マニュアルタイトル
セキュア シェルバージョン 1 と 2 のサポート	セキュア シェルの設定

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

セキュアコピーの機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	セキュアコピー	Secure Copy 機能は、デバイス設定またはデバイスイメージファイルをコピーするための安全で認証された方式を提供します。SCP は、SSH、アプリケーション、および Berkeley r ツールのセキュアな代替手段を提供するプロトコルに依存します。 次のコマンドが導入または変更されました。 debug ip scp および ip scp server enable

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 11 章

コンフィギュレーションの置換とロールバック

- [コンフィギュレーションの置換とロールバックの前提条件 \(241 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの制約事項 \(242 ページ\)](#)
- [コンフィギュレーションの置換とロールバックについて \(242 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの使用方法 \(245 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの設定例 \(253 ページ\)](#)
- [コンフィギュレーションの置換とロールバックに関するその他の参考資料 \(256 ページ\)](#)
- [コンフィギュレーションの置換およびコンフィギュレーションのロールバックの機能履歴 \(256 ページ\)](#)

コンフィギュレーションの置換とロールバックの前提条件

コンフィギュレーションの置換とロールバックの機能に対する入力となるコンフィギュレーションファイルの形式は、標準の Cisco ソフトウェア コンフィギュレーションファイルの、次に示すインデント規則に準拠している必要があります。

- 新しい行のすべてのコマンドは、コマンドがコンフィギュレーションサブモードにない限り、インデントなしで開始します。
- レベル1 コンフィギュレーションサブモード内のコマンドは、スペース1個分インデントします。
- レベル2 コンフィギュレーションサブモード内のコマンドは、スペース2個分インデントします。
- 以下、続くサブモード内のコマンドは、同じようにインデントします。

これらのインデント規則には、ソフトウェアが **show running-config** や **copy running-config destination-url** などのコマンドのコンフィギュレーションファイルを作成する方法が記述され

ています。シスコ デバイスで生成されるコンフィギュレーション ファイルは、いずれもこうした規則に従います。

2つのコンフィギュレーションファイル（現在の実行コンフィギュレーションと、保存された置換用コンフィギュレーション）を合わせたサイズより大きな空きメモリが必要です。

コンフィギュレーションの置換とロールバックの制約事項

デバイスに、2つのコンフィギュレーションファイル（現在の実行コンフィギュレーションと、保存された置換用コンフィギュレーション）を合わせたサイズより大きな空きメモリがない場合、コンフィギュレーション置換操作は実行されません。

ネットワークデバイスの物理コンポーネント（物理インターフェイスなど）に関連する特定の Cisco コンフィギュレーション コマンドは、実行コンフィギュレーションについて追加または削除することはできません。たとえば、コンフィギュレーション置換操作を行っても、そのインターフェイスがデバイス上に物理的に存在する場合、現在の実行コンフィギュレーションから **interface ethernet 0** コマンド行を削除することはできません。同様に、**interface ethernet 1** コマンド行は、そのようなインターフェイスがデバイス上に物理的に存在しない場合、実行コンフィギュレーションに追加することはできません。コンフィギュレーション置換操作でこのタイプの変更を試行すると、その特定のコマンド行が失敗したことを示すエラーメッセージが表示されます。

非常にまれなケースですが、ルータをリロードしないと特定の Cisco コンフィギュレーション コマンドを実行コンフィギュレーションから削除できないことがあります。コンフィギュレーション置換操作でこのタイプのコマンドの削除を試行すると、その特定のコマンド行が失敗したことを示すエラーメッセージが表示されます。

コンフィギュレーションの置換とロールバックについて

コンフィギュレーション アーカイブ

Cisco IOS コンフィギュレーション アーカイブは、**configure replace** コマンドにより提供されるコンフィギュレーションのロールバック機能を強化するために、Cisco IOS コンフィギュレーションファイルのアーカイブの保存、整理、管理を行うことを目的としたメカニズムです。この機能の導入前にも、実行コンフィギュレーションのコピーを **copy running-config destination-url** コマンドを使用して保存し、ローカルやリモートに置換ファイルを保管できました。ただし、この方法ではファイルの自動管理を行うことはできませんでした。一方、コンフィギュレーションの置換とロールバック機能では、実行コンフィギュレーションファイルを自動的に Cisco IOS コンフィギュレーション アーカイブに保存できます。アーカイブされたファイルはコンフィギュレーションのチェックポイントとして参照することができ、**configure replace** コマンドを使用して以前のコンフィギュレーション状態に戻すために利用できます。

archive config コマンドを使用すると、Cisco IOS コンフィギュレーションをコンフィギュレーションアーカイブに保存できます。その場合、標準のディレクトリとファイル名のプレフィクスが使用され、バージョン番号（およびオプションでタイムスタンプ）が自動的に付加されます。バージョン番号は連続したファイルを保存するごとに、1 つずつ大きくなります。この機能により、保存した Cisco IOS コンフィギュレーション ファイルを一貫して識別できます。アーカイブに保存する実行コンフィギュレーションの数は指定することができます。アーカイブ内のファイル数が上限値に達すると、次に最新のファイルが保存されるときに、最も古いファイルが自動的に消去されます。**show archive** コマンドを使用すると、Cisco IOS コンフィギュレーションアーカイブに保存されているすべてのコンフィギュレーションファイルに関する情報が表示されます。

コンフィギュレーション ファイルを保存する Cisco IOS コンフィギュレーションアーカイブは、**configure replace** コマンドで使用することによって、FTP、HTTP、RCP、TFTP のファイルシステム上に配置できます。

コンフィギュレーションの置換

configure replace 特権 EXEC コマンドにより、現在の実行コンフィギュレーションを、保存しておいた Cisco IOS コンフィギュレーション ファイルで置換することができます。この機能は、コンフィギュレーションを保存しておいた状態へ戻すために使用することができ、そのコンフィギュレーション状態が保存された後にどのような変更が加えられても、効果的にロールバックさせることができます。

configure replace コマンドを使用するときは、現在の実行コンフィギュレーションと置換するための、保存された Cisco IOS コンフィギュレーション ファイルを指定する必要があります。置換ファイルは、Cisco IOS デバイスによって作成された完全なコンフィギュレーション (**copy running-config destination-url** コマンドによって作成されたものなど) であることが必要です。あるいは、置換ファイルを外部的に作成する場合は Cisco IOS デバイスが作成するファイル形式に完全に準拠していなければなりません。**configure replace** コマンドを入力すると、現在の実行コンフィギュレーションが指定された置換コンフィギュレーションと比較され、一連の diff が生成されます。2 つのファイルの比較に使用されるアルゴリズムは、**show archive config differences** コマンドで使用されるものと同じです。置換コンフィギュレーションの状態になるよう、diff の結果が Cisco IOS パーサーによって適用されます。diff のみが適用されるため、現在の実行コンフィギュレーション上にすでに存在していた設定コマンドを再適用することにより生じる、潜在的なサービスの中断を避けられます。このアルゴリズムでは、順序に依存するコマンド（アクセスリストなど）へのコンフィギュレーション変更を、複数のパスプロセスを通して効果的に実行します。通常的环境では、コンフィギュレーション置換操作の完了に必要なパスは 3 つまでであり、ループ動作を防ぐためのパスは最大 5 つまでに制限されます。

Cisco IOS **copy source-url running-config** 特権 EXEC コマンドは、保存された Cisco IOS 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へコピーするためによく使用されます。**copy source-url running-config** コマンドを **configure replace target-url** 特権 EXEC コマンドの代わりに使用する場合、主な相違点として次の点に注意が必要です。

- **copy source-url running-config** コマンドはマージ動作であり、ソースファイルと現在の実行コンフィギュレーションの両方のコマンドがすべて保持されます。このコマンドでは、現在の実行コンフィギュレーションにのみ含まれ、ソースファイルには存在しないコマン

ドが削除されることはありません。これに対して、**configure replace target-url** コマンドでは、置換ファイルに存在しないコマンドが現在の実行コンフィギュレーションから削除され、追加する必要があるコマンドが現在の実行コンフィギュレーションに追加されます。

- **copy source-url running-config** コマンドでは、現在の実行コンフィギュレーションにすでに存在しているかどうかにかかわらず、ソースファイル中のすべてのコマンドが適用されます。このアルゴリズムは効率的でない上、場合によってはサービスの停止が発生します。これに対して、**configure replace target-url** コマンドでは適用が必要なコマンドのみを適用し、現在の実行コンフィギュレーションに存在しているコマンドは再適用されません。
- **copy source-url running-config** コマンドでは部分的なコンフィギュレーションファイルもコピー元として使用できますが、**configure replace target-url** コマンドの置換ファイルとして使用できるのは、完全な Cisco IOS コンフィギュレーションファイルのみです。

コンフィギュレーション置換操作にロック機能が導入されました。**configure replace** コマンドが使用されると、コンフィギュレーション置換の動作中、デフォルトで実行コンフィギュレーションファイルがロックされます。このロックメカニズムによって、置換動作の実行中に他のユーザが実行コンフィギュレーションを変更しようとしたために、置換動作の不正終了が発生することを防止できます。**no lock** キーワードを **configure replace** コマンドの実行時に使用すると、実行コンフィギュレーションのロックをディセーブルにできます。

実行コンフィギュレーションのロックは、コンフィギュレーションの置換動作終了時に自動的にクリアされます。**show configuration lock** コマンドを使用すると、現在実行コンフィギュレーションに適用されているロックをすべて表示できます。

コンフィギュレーション ロールバック

ロールバックの概念は、データベースの操作ではトランザクションプロセスモデルに由来します。データベーストランザクションでは、あるデータベースのテーブルに一連の変更を加えることがあります。その後、変更を実行する（変更を恒久的に適用する）か、変更をロールバックする（変更を破棄してテーブルを以前の状態に戻す）かを選択することになります。ここでロールバックが意味するのは、変更のログを含んだジャーナルファイルが破棄され、何の変更も加えられないということです。ロールバック操作の結果として、加えた変更が適用される前の状態に戻ります。

configure replace コマンドを使用することで、以前のコンフィギュレーション状態へ戻ることが可能になり、コンフィギュレーション状態の保存後に加えた変更を効率的にロールバックさせることができます。Cisco IOS コンフィギュレーション ロールバックは、適用された一連の変更をもとにロールバック動作を行うのではなく、保存された Cisco コンフィギュレーションファイルに基づいた特定のコンフィギュレーション状態へ戻るといったコンセプトを採用しています。このコンセプトは、チェックポイント（データベースの保存されたバージョン）に特定の状態を保存しておくという、データベースの考え方に類似しています。

コンフィギュレーションのロールバック機能が必要な場合、コンフィギュレーションの変更には先立って Cisco IOS 実行コンフィギュレーションを保存する必要があります。次に、コンフィギュレーションを変更した後に (**configure replace target-url** コマンドを使用し) 保存したコンフィギュレーションファイルを使って変更をロールバックします。保存された Cisco IOS コン

フィギュレーションファイルならどれでも置換コンフィギュレーションとして指定できるため、一部のロールバックモデルのように、ロールバックの数が制限されることもありません。

コンフィギュレーション ロールバック変更確認

コンフィギュレーションロールバック変更確認機能により、コンフィギュレーション変更の実行に際して確認を要求するようオプションで設定できます。この確認が受信できない場合、コンフィギュレーションは変更が適用される前の状態に戻されます。このメカニズムは、ネットワークデバイスとユーザまたは管理アプリケーションとの接続において、コンフィギュレーション変更に起因する切断を防止するものです。

コンフィギュレーションの置換とロールバックの利点

- コンフィギュレーションの変更を効率的にロールバックさせて、以前のコンフィギュレーション状態へ戻ることが可能。
- デバイスをリロードしたり、CLIで実行コンフィギュレーションファイルに加えた変更を手動で元に戻したりすることなく、現在の実行コンフィギュレーションファイルをスタートアップコンフィギュレーションファイルと置換できるため、システムのダウンタイムが減少。
- 保存しておいたどの Cisco IOS コンフィギュレーション状態に戻すことも可能。
- 追加や削除が必要なコマンドだけが影響される場合、デバイスに完全なコンフィギュレーションファイルを適用することができるため、コンフィギュレーションの変更がシンプルに。
- **configure replace** コマンドを **copy source-url running-config** コマンドの代用として使用すると、現在の実行コンフィギュレーションにある既存のコマンドが再度適用されないため、効率が向上し、サービス停止のリスクが回避されます。

コンフィギュレーションの置換とロールバックの使用方法

コンフィギュレーションアーカイブの作成

configure replace コマンドを使用するうえで前提条件となる設定はありません。**configure replace** コマンドと、Cisco IOS コンフィギュレーションアーカイブおよび **archive config** コマンドとの併用は任意ですが、コンフィギュレーションロールバックのシナリオでは大きな利点があります。**archive config** コマンドを使用する前に、コンフィギュレーションアーカイブを設定しておく必要があります。コンフィギュレーションアーカイブの特性を設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</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>
ステップ 3	<p>archive</p> <p>例 :</p> <pre>Device(config)# archive</pre>	<p>アーカイブ コンフィギュレーション モードを開始します。</p>
ステップ 4	<p>path url</p> <p>例 :</p> <pre>Device(config-archive)# path flash:myconfiguration</pre>	<p>Cisco IOS コンフィギュレーションアーカイブの場所と、ファイル名のプレフィックスを指定します。</p> <p>(注) パスのところでファイルの代わりにディレクトリを指定する場合、ディレクトリ名は path flash:/directory/ のように後ろにスラッシュを付ける必要があります。このスラッシュはファイル名の後ろでは必要ありません。ディレクトリを指定する場合にだけ使います。</p>
ステップ 5	<p>maximum number</p> <p>例 :</p> <pre>Device(config-archive)# maximum 14</pre>	<p>(任意) Cisco IOS コンフィギュレーションアーカイブに保存される実行コンフィギュレーションのアーカイブ ファイル数の上限値を設定します。</p> <ul style="list-style-type: none"> number 引数は、Cisco IOS コンフィギュレーションアーカイブに保存される実行コンフィギュレーションのアーカイブ ファイル数の上限値を示します。有効な値は 1 ~ 14 で、デフォルトは 10 です。

	コマンドまたはアクション	目的
		<p>(注) このコマンドを使用する前に、path コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。</p>
ステップ 6	<p>time-period <i>minutes</i></p> <p>例 :</p> <pre>Device(config-archive)# time-period 1440</pre>	<p>(任意) CiscoIOS コンフィギュレーションアーカイブに実行コンフィギュレーションのアーカイブファイルを自動保存する間隔を設定します。</p> <ul style="list-style-type: none"> • Cisco IOS コンフィギュレーションアーカイブに現在の実行コンフィギュレーションのアーカイブファイルをどれほどの頻度で自動保存するかを、<i>minutes</i> 引数により分単位で指定します。 <p>(注) このコマンドを使用する前に、path コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。</p>
ステップ 7	<p>end</p> <p>例 :</p> <pre>Device(config-archive)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 8	<p>archive config</p> <p>例 :</p> <pre>Device# archive config</pre>	<p>現在の実行設定ファイルを設定アーカイブに保存します。</p> <p>(注) このコマンドを使用する前に、path コマンドを設定する必要があります。</p>

コンフィギュレーションの置換やロールバック操作の実行

保存された Cisco IOS コンフィギュレーションファイルで現在の実行コンフィギュレーションファイルを置換するには、次の作業を実行します。



(注) この手順の前に、コンフィギュレーションアーカイブを作成しておく必要があります。詳細については、[コンフィギュレーションアーカイブの作成](#)を参照してください。次に、現在の実行コンフィギュレーションで問題が生じた場合に、アーカイブしておいたコンフィギュレーションに戻す手順の詳細を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure replace target-url [nolock] [list] [force] [ignore case] [revert trigger [error]] [timer minutes] time minutes]</p> <p>例 :</p> <pre>Device# configure replace flash: startup-config time 120</pre>	<p>保存しておいた Cisco IOS コンフィギュレーション ファイルで現在の実行コンフィギュレーション ファイルを置換します。</p> <ul style="list-style-type: none"> target-url 引数は、archive config コマンドで作成されたコンフィギュレーション ファイルなど、現在の実行コンフィギュレーションと置換する、保存された Cisco IOS コンフィギュレーション ファイルの URL です (Cisco IOS ファイルシステムでアクセス可能なもの)。 list キーワードは、コンフィギュレーション置換動作のパスごとに、Cisco IOS ソフトウェア パーサーによって適用されるコマンドラインのリストを表示します。実行されたパスの総数も表示されます。 force キーワードは、現在の実行コンフィギュレーションから指定した Cisco IOS コンフィギュレーション ファイルへの置換を、確認プロンプトを出さずに実行します。 time minutes キーワードおよび引数は、現在の実行コンフィギュレーション ファイルの置換確認のために configure confirm コマンドを入

	コマンドまたはアクション	目的
		<p>力しなければならない制限時間（分単位）を指定します。configure confirm コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作は自動的に戻されます（つまり、現在の実行コンフィギュレーションファイルがconfigure replace コマンド入力以前のコンフィギュレーション状態へと回復されます）。</p> <ul style="list-style-type: none"> • nolock キーワードは、コンフィギュレーション置換操作中に他のユーザが実行コンフィギュレーションを変更しないように実行コンフィギュレーションファイルをロックする機能をオフにします。 • revert trigger キーワードは、元のコンフィギュレーションへ戻すトリガーを次の内容から設定します。 <ul style="list-style-type: none"> • error : エラー時に元のコンフィギュレーションに戻します。 • timer minutes : 指定した時間が過ぎると元のコンフィギュレーションに戻します。 • ignore case キーワードで、コンフィギュレーションに確認コマンドの大文字と小文字の区別を無視させることができます。
<p>ステップ 3</p>	<p>configure revert { now timer {minutes idle minutes} }</p> <p>例 :</p> <pre>Device# configure revert now</pre>	<p>(任意) 時間指定ロールバックをキャンセルしてロールバックを即時トリガーする、または時間指定ロールバックのパラメータをリセットするには、特権EXECモードでconfigure revert コマンドを使用します。</p> <ul style="list-style-type: none"> • now : ロールバックをただちにトリガーします。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • timer : コンフィギュレーションを元に戻すタイマーをリセットします。 • 元に戻す時間を分単位で新たに指定するには、<i>minutes</i> 引数を timer キーワードとともに使用します。 • 保存されたコンフィギュレーションに戻すまでに、操作が行われないアイドル時間を最大どれほど長く許容できるかを設定するには、分単位の時間とともに idle キーワードを使用します。
ステップ 4	configure confirm 例 : Device# configure confirm	(任意) 保存しておいた Cisco IOS コンフィギュレーションファイルの現在の実行コンフィギュレーションファイルへの置換を確認します。 (注) このコマンドは、 configure replace コマンドの time seconds キーワードおよび引数が指定されている場合にのみ使用します。
ステップ 5	exit 例 : Device# exit	ユーザ EXEC モードに戻ります。

機能のモニタリングおよびトラブルシューティング

コンフィギュレーションの置換とロールバック機能をモニタおよびトラブルシューティングするには、この手順を実行します。

手順

ステップ 1 enable

このコマンドを使用して、特権EXECモードをイネーブルにします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
Device#
```

ステップ2 show archive

Cisco IOS コンフィギュレーションアーカイブに保存されているファイルに関する情報を表示するには、次のコマンドを使用します。

例：

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfiguration-2
Archive # Name
0
1      flash:myconfiguration-1 <- Most Recent
2
3
4
5
6
7
8
9
10
11
12
13
14
```

次に、実行コンフィギュレーションのアーカイブファイルをいくつか保存した状態で **show archive** コマンドを使用した場合の出力例を示します。この例では、保存されるアーカイブファイルの最大数が3に設定されています。

例：

```
Device# show archive
There are currently 3 archive configurations saved.
The next archive file will be named flash:myconfiguration-8
Archive # Name
0
1      :Deleted
2      :Deleted
3      :Deleted
4      :Deleted
5      flash:myconfiguration-5
6      flash:myconfiguration-6
7      flash:myconfiguration-7 <- Most Recent
8
9
10
11
12
13
14
```

ステップ3 debug archive versioning

このコマンドを使用して、Cisco IOS コンフィギュレーションアーカイブのアクティビティのデバッグを有効にして、コンフィギュレーションの置換とロールバックをモニタおよびトラブルシューティングします。

例：

```
Device# debug archive versioning
Jan 9 06:46:28.419:backup_running_config
Jan 9 06:46:28.419:Current = 7
Jan 9 06:46:28.443:Writing backup file flash:myconfiguration-7
Jan 9 06:46:29.547: backup worked
```

ステップ4 debug archive config timestamp

このコマンドを使用して、コンフィギュレーション置換操作の各必須段階の処理時間、および操作中のコンフィギュレーションファイルのサイズのデバッグをイネーブルにします。

例：

```
Device# debug archive config timestamp
Device# configure replace flash:myconfiguration force
Timing Debug Statistics for IOS Config Replace operation:
  Time to read file usbflash0:sample_2.cfg = 0 msec (0 sec)
  Number of lines read:55
  Size of file      :1054
Starting Pass 1
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:93
  Size of file      :2539
  Time taken for positive rollback pass = 320 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for negative incremental diffs pass = 59 msec (0 sec)
  Time taken by PI to apply changes = 0 msec (0 sec)
  Time taken for Pass 1 = 380 msec (0 sec)
Starting Pass 2
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:55
  Size of file      :1054
  Time taken for positive rollback pass = 0 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for Pass 2 = 0 msec (0 sec)
Total number of passes:1
Rollback Done
```

ステップ5 exit

このコマンドを使用して、ユーザ EXEC モードに戻ります。

例：

```
Device# exit
Device>
```

コンフィギュレーションの置換とロールバックの設定例

コンフィギュレーションアーカイブの作成

次の例は、Cisco IOS コンフィギュレーションアーカイブの初期設定を実行する方法を示しています。この例では、`flash:myconfiguration` がコンフィギュレーションアーカイブの保存位置およびファイル名のプレフィックスとして設定され、保存するアーカイブファイルが最大 10 個に設定されます。

```
configure terminal
!
archive
  path flash:myconfiguration
  maximum 10
end
```

現在の実行コンフィギュレーションを保存された Cisco IOS コンフィギュレーションファイルで置換

次の例では、`flash:myconfiguration` という名前で保存された Cisco IOS コンフィギュレーションファイルで現在の実行コンフィギュレーションを置換する方法を示します。`configure replace` コマンドでは、確認プロンプトでインタラクティブに操作を進めます。

```
Device# configure replace flash:myconfiguration
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
```

次の例では、コンフィギュレーション置換操作中に適用されるコマンドラインを表示するために、`list` キーワードを指定しています。

```
Device# configure replace flash:myconfiguration list
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
!Pass 1
!List of Commands:
no snmp-server community public ro
snmp-server community mystring ro

end
Total number of passes: 1
Rollback Done
```

スタートアップコンフィギュレーションファイルへの復帰

次の例に、**configure replace** コマンドを使用して Cisco IOS スタートアップコンフィギュレーションファイルへ復元する方法を示します。この例は、オプションの **force** キーワードを使用して、インタラクティブユーザプロンプトをオーバーライドする方法を示しています。

```
Device# configure replace flash:startup-config force
Total number of passes: 1
Rollback Done
```

configure confirm コマンドを使用したコンフィギュレーション置換操作の実行

次に、**configure replace** コマンドを **time minutes** キーワードおよび引数とともに使用する例を示します。現在の実行コンフィギュレーションファイルの置換を実行するには、指定の制限時間内に **configure confirm** コマンドを入力する必要があります。**configure confirm** コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作は自動的に戻されます（つまり、現在の実行コンフィギュレーションファイルが **configure replace** コマンド入力以前のコンフィギュレーション状態へと回復されます）。

```
Device# configure replace flash:startup-config time 120
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
Device# configure confirm
```

次に、**configure revert** コマンドを **timer** キーワードとともに使用する例を示します。時間指定ロールバックをキャンセルしてロールバックを即時トリガーする、または時間指定ロールバックのパラメータをリセットするには、**configure revert** コマンドを入力する必要があります。

```
Device# configure revert timer 100
```

コンフィギュレーションロールバック操作の実行

次の例は、現在実行中のコンフィギュレーションへの変更を行い、その変更をロールバックする方法を示しています。コンフィギュレーションロールバック操作の一部として、ファイルに変更を加える前に現在の実行コンフィギュレーションを保存する必要があります。この例では、現在の実行コンフィギュレーションの保存に **archive config** コマンドが使用されています。**configure replace** コマンドで生成された出力は、ロールバック操作を完了するために1つのパスのみが実行されたことを示します。



(注) **archive config** コマンドを使用する前に、**path** コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。

次のように、設定アーカイブの現在実行中のコンフィギュレーションを保存します。

```
archive config
```

それから、次の例に示すようにコンフィギュレーションの変更を入力します。

```
configure terminal
!
user netops2 password rain
user netops3 password snow
exit
```

実行コンフィギュレーションファイルに変更を加えた後、それらの変更をロールバックさせて、変更前のコンフィギュレーションに戻したくなくなります。**show archive** コマンドは、交換ファイルとして使用される設定のバージョンを確認するために使用されます。次の例に示すように、**configure replace** コマンドは交換コンフィギュレーションファイルへ戻すために使用されます。

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfiguration-2
Archive # Name
0
1 flash:myconfiguration-1 <- Most Recent
2
3
4
5
6
7
8
9
10
Device# configure replace flash:myconfiguration-1
Total number of passes: 1
Rollback Done
```

コンフィギュレーションの置換とロールバックに関するその他の参考資料

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

コンフィギュレーションの置換およびコンフィギュレーションのロールバックの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	コンフィギュレーションの置換とロールバック	Cisco IOS コンフィギュレーションアーカイブは、 configure replace コマンドにより提供されるコンフィギュレーションのロールバック機能を強化するために、Cisco IOS コンフィギュレーション ファイルのアーカイブの保存、整理、管理を行うことを目的としたメカニズムです。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 12 章

ソフトウェア メンテナンス アップグレード

ソフトウェア メンテナンス アップグレード (SMU) は、システムにインストールして修正やセキュリティ解決をリリースされたイメージに提供できるパッケージです。

- [ソフトウェア メンテナンス アップグレードの制約事項 \(257 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードについて \(257 ページ\)](#)
- [ソフトウェア メンテナンスの更新の管理方法 \(258 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードの設定例 \(262 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードのその他の参考資料 \(275 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードの機能の履歴 \(275 ページ\)](#)

ソフトウェア メンテナンス アップグレードの制約事項

- ホットパッチは Cisco Catalyst 9200 シリーズ スイッチでサポートされていません。
- SMU は、インストールモードを使用したコールドパッチのみをサポートします。

ソフトウェア メンテナンス アップグレードについて

SMU の概要

SMU は、システムにインストールして修正やセキュリティ解決をリリースされたイメージに提供できるパッケージです。SMU パッケージはリリースごとおよびコンポーネントごとに提供されます。

SMU はネットワークの問題に迅速に対応できるようにするとともに、必要なテストの時間と範囲を削減するため、従来の Cisco IOS ソフトウェアには多大なメリットがあります。Cisco IOS XE プラットフォームでは SMU の互換性を内部的に検証し、互換性のない SMU はインストールできません。

すべて SMU が後続の Cisco IOS XE ソフトウェアメンテナンスリリースに統合されています。SMU は独立した自己完結型パッケージであり、前提条件や依存関係はありません。SMU はどのような順序でもインストールまたはアンインストールできます。

SMU は拡張メンテナンスリリースでのみ、基盤となるソフトウェアリリースのライフサイクルにわたってサポートされます。

SMU をインストールするには、次の基本的な手順を実行します。

1. ファイルシステムに SMU を追加します。
2. システムで SMU をアクティブ化します。
3. リロードが繰り返されても持続させるための SMU の変更をコミットします。

SMU のワークフロー

SMU プロセスは、シスコカスタマーサポートへの要求によって開始されます。カスタマーサポートに連絡し、SMU 要求を行います。

SMU パッケージがリリースされると [Cisco Software Download]https://www.cisco.com/c/en_in/support/index.html ページに掲載されます。そのパッケージをダウンロードし、インストールします。

SMU パッケージ

SMU パッケージには、パッケージの内容を記述するいくつかのメタデータ、および SMU が要求されている報告済みの問題の修正とともに、リリースにパッチを適用するための一連のファイルがいくつか含まれています。SMU パッケージは、公開キーインフラストラクチャ (PKI) コンポーネントのパッチ適用もサポートします。

SMU のリロード

すべての SMU で、アクティブ化中にシステムをコールドリロードする必要があります。コールドリロードは、オペレーティングシステムを完全にリロードします。このアクションは、リロードの間、トラフィックフローに影響します。このリロードにより、SMU の一部としてインストールされている正しいライブラリとファイルですべてのプロセスが起動します。

ソフトウェアメンテナンスの更新の管理方法

単一のコマンド (1 ステップのプロセス) または個別のコマンド (3 ステップのプロセス) を使用して SMU パッケージのインストール、アクティブ化、コミットを行うことができます。



ヒント SMU パッケージファイルを 1 つのみインストールする必要がある場合は 1 ステッププロセスを使用し、複数の SMU をインストールする必要がある場合は 3 ステッププロセスを使用します。3 ステッププロセスにより、インストールする SMU パッケージファイルが複数ある場合に必要なリロード回数が最小限に抑えられます。

SMU パッケージのインストール : 1 ステッププロセス

このタスクでは、SMU パッケージをインストールするための単一の **install add file activate commit** コマンドの使用方法を示します。

始める前に

インストールする SMU がデバイスにインストールされているソフトウェアイメージに対応していることを確認します。たとえば、SMU `cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin` は、ソフトウェアイメージ `cat9k_lite_iosxe.16.09.04.SPA.bin` と互換性があります。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	install add file flash: filename [activate commit] 例 : Device# install add file flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin activate commit	メンテナンス更新パッケージをフラッシュからデバイスにコピーし、プラットフォームおよびイメージバージョンの互換性チェックを実行し、SMU パッケージをアクティブ化し、そのパッケージを複数回リロードしても維持されるようにします。このコマンドは、.bin ファイルの個別のコンポーネントをサブパッケージと <code>packages.conf</code> ファイルに抽出します。 また、リモートロケーションから (FTP、HTTP、HTTPS、または TFTP を使用して) SMU パッケージをコピーすることもできます。 (注) TFTP を使用して SMU ファイルをコピーする場合は、ブートフラッシュを使用して SMU をアクティブにします。

	コマンドまたはアクション	目的
ステップ 3	exit 例 : Device# exit	特権 EXEC モードを終了し、ユーザ EXEC モードに戻ります。

SMU パッケージのインストール : 3ステッププロセス

このタスクでは、SMU パッケージをインストールするための 3 ステップのプロセスを示します。複数の SMU をインストールし、複数のリロードを回避するには、この方法を使用します。

始める前に

インストールする SMU がデバイスにインストールされているソフトウェアイメージに対応していることを確認します。たとえば、SMU `cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin` は、ソフトウェアイメージ `cat9k_lite_iosxe.16.09.04.SPA.bin` と互換性があります。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	install add file location filename 例 : Device# install add file flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin Device# install add file flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin	メンテナンス更新パッケージをフラッシュからデバイスにコピーし、プラットフォームとイメージのバージョンの互換性チェックを実行し、必要に応じてすべてのメンバノードまたは FRU に SMU パッケージを追加します。このコマンドは、ファイルで基本的な互換性チェックを実行し、SMU パッケージがプラットフォームでサポートされていることも確認します。また、 <code>package/SMU.sta</code> ファイル内にエントリを追加することで、ステータスを監視し、維持できるようにします。 また、リモートロケーションから（FTP、HTTP、HTTPS、または TFTP を使用して）SMU パッケージをコピーすることもできます。
ステップ 3	install activate file location filename 例 :	追加された SMU パッケージファイルをアクティブ化し、パッケージステータス

	コマンドまたはアクション	目的
	<pre>Device# install activate file flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SEA.smu.bin, cat9k_lite_iosxe.16.12.03.CSCvt72427.SEA.smu.bin</pre>	<p>の詳細を更新します。アクティブ化のプロセスを完了するために、システムのリロードが求められます。</p> <p>複数の SMU を入力する場合は、（前後にスペースを入れずに）カンマを使用してファイル名を区切ります。また、合計文字数が 128 を超えないようにしてください。この手順にはリロードが含まれません。</p>
ステップ 4	<p>install commit</p> <p>例 :</p> <pre>Device# install commit</pre>	<p>リロードが繰り返されても持続するようにアクティブ化の変更をコミットします。</p> <p>アクティブ化の後で、システムがアップしている間、または最初のリロード後にコミットできます。パッケージがアクティブになっていてもコミットされていない場合は、最初のリロード後はアクティブの状態を保ちますが、2 回目のリロード後はアクティブ状態を保ちません。</p>

SMU の管理

このタスクでは、インストール状態のロールバック、非アクティブ化、および以前にインストールした SMU パッケージのデバイスからの削除方法を示します。これは、1 ステップおよび 3 ステップのプロセスを使用してインストールされた SMU に使用できます。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。</p>
ステップ 2	<p>install rollback to {base committed id commit-ID}</p> <p>例 :</p> <pre>Device# install rollback to committed</pre>	<p>デバイスを以前のインストール状態に戻します。ロールバック後にリロードする必要があります。</p>

	コマンドまたはアクション	目的
ステップ 3	install deactivate file <i>location filename</i> 例： Device# install deactivate file flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin	アクティブなパッケージを非アクティブ化し、パッケージステータスを更新し、再起動またはリロードするプロセスをトリガーします。
ステップ 4	install remove {file location filename inactive} 例： Device# install remove file flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin	指定された SMU が非アクティブであるかどうかを確認し、非アクティブである場合はファイルシステムから削除します。 inactive オプションは、非アクティブなパッケージをファイルシステムからすべて削除します。
ステップ 5	show version 例： Device# show version	デバイスのイメージバージョンを表示します。
ステップ 6	show install summary 例： Device# show install summary	アクティブ パッケージに関する情報を表示します。 このコマンドの出力は、設定されている install コマンドに応じて変化します。

ソフトウェアメンテナンスアップグレードの設定例

次に、SMU の設定例を示します。

- 例：SMU のインストール (3 ステッププロセス、flash : を使用) (262 ページ)
- 例：複数の SMU のインストール (3 ステッププロセス、flash : を使用) (265 ページ)
- 例：SMU のインストール (3 ステッププロセス、TFTP : を使用) (271 ページ)
- 例：SMU パッケージの管理 (追加の show コマンド、ロールバック、非アクティブ化) (273 ページ)

例：SMU のインストール (3 ステッププロセス、flash : を使用)

次に、3 ステッププロセスを使用して SMU パッケージをインストールする例を示します。ここでは、SMU パッケージファイルがデバイスのフラッシュに保存されます。

1. フラッシュから SMU パッケージファイルをコピーしてインストールします。

```
Device# install add file flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
install_add: START Wed Jun 10 14:17:45 IST 2020
install_add: Adding SMU

--- Starting initial file syncing ---
```

```
Info: Finished copying flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin to the
selected switch(es)
Finished initial file syncing
```

```
*Jun 10 14:17:48.128 IST: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install add flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.binExecuting
pre scripts....
Executing pre sripts done.
--- Starting SMU Add operation ---
Performing SMU_ADD on all members
  [1] SMU_ADD package(s) on switch 1
  [1] Finished SMU_ADD on switch 1
Checking status of SMU_ADD on [1]
SMU_ADD: Passed on [1]
Finished SMU Add operation
```

```
SUCCESS: install_add /flash/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin Wed Jun
10 14:18:00 IST 2020
```

show install summary コマンドを使用して、SMU パッケージファイルの追加とインストールを確認します。SMU パッケージファイルはまだアクティブ化およびコミットされていないため、ステータスは I です。

```
Device# show install summary
```

```
[ Switch 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type  St   Filename/Version
-----
SMU   I    flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
IMG   C    16.9.4.0.3431
-----
```

```
Auto abort timer: inactive
-----
```

2. SMU パッケージファイルをアクティブ化します。

```
Device# install activate file flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
```

```
install_activate: START Wed Jun 10 14:19:59 IST 2020
install_activate: Activating SMU
```

```
*Jun 10 14:20:01.513 IST: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install activate flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
```

```
This operation requires a reload of the system. Do you want to proceed? [y/n]y
```

```
Executing pre scripts....
```

```
Executing pre sripts done.
```

```
--- Starting SMU Activate operation ---
Performing SMU_ACTIVATE on all members
  [1] SMU_ACTIVATE package(s) on switch 1
  [1] Finished SMU_ACTIVATE on switch 1
Checking status of SMU_ACTIVATE on [1]
SMU_ACTIVATE: Passed on [1]
Finished SMU Activate operation
```

```
install_activate: Reloading the box to complete activation of the SMU...
```

```
install_activate will reload the system now!

*Jun 10 14:20:22.258 IST: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 1
R0/0: rollback_timer: Install auto abort timer will expire in 7200 seconds
    Chassis 1 reloading, reason - Reload command
Jun 10 14:20:28.291: %PMAN-5-EXITACTION: F0/0: pvp: Process manager is exiting: reload
fp action requested
Jun 10 14:20:30.718: %PMAN-5-EXITACTION: R0/0: pvp: Proce
Jun 10 14:20:34.834: %PMAN-5-EXITACTION: C0/0: pvp: Process manager is exiting:
Jun 10 14:20:36.053: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install activate SMU flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
watchdog watchdog0: watchdog did not stop!
reboot: Restarting system
```

```
Initializing Hardware...
<output truncated>
```

```
#####
Jun 10 08:52:01.806: %BOOT-5-BOOTTIME_SMU_TEMP_ACTIVE_DETECTED: R0/0: install_engine:
SMU file /flash/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin active temporary...
SMU commit is pending
```

```
Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_LITE_IOSXE), Version
16.9.4, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2019 by Cisco Systems, Inc.
Compiled Thu 22-Aug-19 17:30 by mcpre
```

```
<output truncated>
```

show install summary コマンドを使用して SMU パッケージファイルのアクティブ化を確認します。SMU パッケージファイルはまだコミットされていないため、ステータスは U です。

```
[ Switch 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type  St  Filename/Version
-----
SMU   U   flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
IMG   C   16.9.4.0.3431
-----
```

```
-----
Auto abort timer: active on install_activate, time before rollback - 01:41:52
-----
```

3. SMU パッケージファイルをコミットします。

```
Device# install commit
install_commit: START Wed Jun 10 14:38:42 IST 2020
install_commit: Committing SMU

*Jun 10 14:38:44.906 IST: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install commitExecuting pre scripts....
Executing pre sripts done.
--- Starting SMU Commit operation ---
Performing SMU_COMMIT on all members
  [1] SMU_COMMIT package(s) on switch 1
  [1] Finished SMU_COMMIT on switch 1
Checking status of SMU_COMMIT on [1]
SMU_COMMIT: Passed on [1]
Finished SMU Commit operation
```



```
SUCCESS: install_commit /flash/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin Wed
Jun 10 14:38:58 IST 2020
*Jun 10 14:38:59.385 IST: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 1 R0/0:
install_engine: Completed install commit SMU
```

show install summary コマンドを使用してコミットを確認します。SMU パッケージファイルのインストール、アクティブ化、コミットが行われました。ステータスはcです。

```
Device# show install summary
[ Switch 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   C    flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
IMG   C    16.9.4.0.3431
-----
Auto abort timer: inactive
-----
```

show install active コマンドを使用してアクティブパッケージを確認します。

```
Device# show install active
[ Switch 1 ] Active Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   C    flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
IMG   C    16.9.4.0.3431
-----
```

show version コマンドを使用して、バージョンを確認します。

```
Device# show version
Cisco IOS XE Software, Version 16.09.04
Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_LITE_IOSXE), Version
16.9.4, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2019 by Cisco Systems, Inc.
Compiled Thu 22-Aug-19 17:30 by mcpre
...
```

例：複数のSMUのインストール（3ステッププロセス、flash：を使用）

次に、3ステッププロセスを使用して複数のSMUパッケージファイルをインストールする例を示します。ここでは、SMUパッケージファイルがデバイスのフラッシュに保存されます。

スイッチスタックにインストールされているSMUファイルは次のとおりです。

```
cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin および
cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
```

1. (任意) スイッチスタックの準備ができており、SMUパッケージファイルがデバイスのフラッシュ内にあることを確認します。

```
Device# show switch
Switch/Stack Mac Address : 08ec.f586.aa80 - Local Mac Address
Mac persistency wait time: Indefinite

Switch# Role Mac Address Priority Version H/W Current State
-----
*1 Active 08ec.f586.aa80 1 V01 Ready
2 Member 7488.bb3c.f600 1 V01 Ready
3 Member 7488.bb3f.9c00 1 V01 Ready
4 Member 08ec.f5ee.1080 1 V01 Ready
5 Standby 08ec.f589.7c80 1 V01 Ready

Device# dir flash: | i smu

89075 -rw- 79256 Oct 26 2035 07:07:42 +00:00
cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
89082 -rw- 9656 Oct 26 2035 07:08:08 +00:00
cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
```

2. フラッシュから SMU パッケージファイルをコピーして追加します。

一度に1つの SMU パッケージファイルのみが追加されます。SMU パッケージファイルを追加する間にリロードは必要ありません。

```
Device# install add file flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
install_add: START Fri Oct 26 07:10:59 UTC 2035
Oct 26 07:11:01.695 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install add flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
install_add: Adding SMU
install_add: Checking whether new add is allowed ....

--- Starting initial file syncing ---

*Oct 26 07:11:01.643: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install add flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin[1]:
Copying flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin from switch 1 to switch
2 3 4 5
[2 3 4 5]: Finished copying to switch 2 switch 3 switch 4 switch 5
Info: Finished copying flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin to the
selected switch(es)
Finished initial file syncing

--- Starting SMU Add operation ---
Performing SMU_ADD on all members
[1] SMU_ADD package(s) on switch 1
[1] Finished SMU_ADD on switch 1
[2] SMU_ADD package(s) on switch 2
[2] Finished SMU_ADD on switch 2
[3] SMU_ADD package(s) on switch 3
[3] Finished SMU_ADD on switch 3
[4] SMU_ADD package(s) on switch 4
[4] Finished SMU_ADD on switch 4
[5] SMU_ADD package(s) on switch 5
[5] Finished SMU_ADD on switch 5
Checking status of SMU_ADD on [1 2 3 4 5]
SMU_ADD: Passed on [1 2 3 4 5]
Finished SMU Add operation

SUCCESS: install_add Fri Oct 26 07:11:45 UTC 2035
Oct 26 07:11:46.695 %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
```

```

install add SMU flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
Device#
*Oct 26 07:11:46.656: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 1 R0/0: install_engine:
Completed install add SMU flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin

```

show install summary コマンドを使用して、最初の SMU パッケージファイルの追加を確認します。

```

Device# show install summary
[ Switch 1 2 3 4 5 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   I    flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
IMG   C    16.12.3.0.3752
-----
Auto abort timer: inactive
-----

```

2 番目の SMU パッケージファイルを追加します。

```

Device# install add file flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin

install_add: START Fri Oct 26 07:12:38 UTC 2035
Oct 26 07:12:40.782 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install add flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
install_add: Adding SMU
install_add: Checking whether new add is allowed ....

--- Starting initial file syncing ---

*Oct 26 07:12:40.743: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install add flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin[1]:
Copying flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin from switch 1 to switch
2 3 4 5
[2 3 4 5]: Finished copying to switch 2 switch 3 switch 4 switch 5
Info: Finished copying flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin to the
selected switch(es)
Finished initial file syncing

--- Starting SMU Add operation ---
Performing SMU_ADD on all members
[1] SMU_ADD package(s) on switch 1
[1] Finished SMU_ADD on switch 1
[2] SMU_ADD package(s) on switch 2
[2] Finished SMU_ADD on switch 2
[3] SMU_ADD package(s) on switch 3
[3] Finished SMU_ADD on switch 3
[4] SMU_ADD package(s) on switch 4
[4] Finished SMU_ADD on switch 4
[5] SMU_ADD package(s) on switch 5
[5] Finished SMU_ADD on switch 5
Checking status of SMU_ADD on [1 2 3 4 5]
SMU_ADD: Passed on [1 2 3 4 5]
Finished SMU Add operation

SUCCESS: install_add Fri Oct 26 07:13:24 UTC 2035
Oct 26 07:13:25.656 %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install add SMU flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
Decive#

```

```
*Oct 26 07:13:25.616: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 1 R0/0: install_engine:
Completed install add SMU flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
```

show install summary コマンドを使用して、両方の SMU パッケージファイルの追加とインストールを確認します。両方のパッケージファイルがまだアクティブ化およびコミットされていないため、ステータスは I です。

```
Device# show install summary
```

```
[ Switch 1 2 3 4 5 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   I    flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
SMU   I    flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
IMG   C    16.12.3.0.3752
-----
Auto abort timer: inactive
-----
```

3. SMU パッケージファイルをアクティブ化します。

複数の SMU を入力する場合は、（前後にスペースを入れずに）カンマを使用してファイル名を区切ります。また、合計文字数が 128 を超えないようにしてください。この手順にはリロードが含まれます。

```
Device# install activate file
```

```
flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin,cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
```

```
install_activate: START Sun Oct 28 13:23:42 UTC 2035
Oct 28 13:23:44.620 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install activate
flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin,cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
install_activate: Activating SMU
```

```
*Oct 28 13:23:44.581: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install activate
flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin,cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
```

```
This operation may require a reload of the system. Do you want to proceed? [y/n]y
Executing pre scripts....
```

```
Executing pre sripts done.
```

```
--- Starting SMU Activate operation ---
Performing SMU_ACTIVATE on all members
```

```
*Oct 28 13:24:41.563: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 1 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 secondsOct 28
13:24:43.259: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: R0/0: rollback_timer:
Install auto abort timer will expire in 7200 seconds
*Oct 28 13:24:43.222: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 4 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 seconds
*Oct 28 13:24:43.192: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 3 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 seconds
*Oct 28 13:24:43.134: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 2 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 seconds
*Oct 28 13:24:43.825: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 5 R0/0:
rollback_timer: Install auto abort timer will expire in 7200 seconds [1] SMU_ACTIVATE
```

```

package(s) on switch 1
 [1] Finished SMU_ACTIVATE on switch 1
 [2] SMU_ACTIVATE package(s) on switch 2
 [2] Finished SMU_ACTIVATE on switch 2
 [3] SMU_ACTIVATE package(s) on switch 3
 [3] Finished SMU_ACTIVATE on switch 3
 [4] SMU_ACTIVATE package(s) on switch 4
 [4] Finished SMU_ACTIVATE on switch 4
 [5] SMU_ACTIVATE package(s) on switch 5
 [5] Finished SMU_ACTIVATE on switch 5
Checking status of SMU_ACTIVATE on [1 2 3 4 5]
SMU_ACTIVATE: Passed on [1 2 3 4 5]
Finished SMU Activate operation

install_activate: Reloading the box to complete activation of the SMU...
install_activate will reload the system now!

Chassis 4 reloading, reason - Reload command
reload fp action requested
rp processes exit with reload switch code

watchdog watchdog0: watchdog did not stop!
reboot: Restarting system

Initializing Hardware...

System Bootstrap, Version 16.12.1r [FC6], RELEASE SOFTWARE (P)
Compiled Thu 02/13/2020 12:36:08 by rel

Current ROMMON image : Primary
C9200L-24T-4G platform with 2097152 Kbytes of main memory

boot: attempting to boot from [flash:packages.conf]
boot: reading file packages.conf

#####
Oct 28 13:26:55.653: %BOOT-5-BOOTTIME_SMU_TEMP_ACTIVE_DETECTED: R0/0: install_engine:
SMU file /flash/cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin active temporary...
SMU commit is pending
Oct 28 13:26:55.912: %BOOT-5-BOOTTIME_SMU_TEMP_ACTIVE_DETECTED: R0/0: install_engine:
SMU file /flash/cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin active temporary...
SMU commit is pending

Waiting for 120 seconds for other switches to boot
#####
Switch number is 4
All switches in the stack have been discovered. Accelerating discovery

```

show install summary コマンドを使用して SMU パッケージファイルのアクティブ化を確認します。両方のファイルがまだコミットされていないため、ステータスは U です。

```

Device# show install summary
[ Switch 1 2 3 4 5 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   U     flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin

```

```
SMU   U   flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
IMG   C   16.12.3.0.3752
```

```
-----
Auto abort timer: active on install_activate, time before rollback - 01:50:16
-----
```

4. SMU パッケージファイルをコミットします。

```
Device# install commit
install_commit: START Sun Oct 28 13:34:42 UTC 2035
Oct 28 13:34:45.202 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install commit

*Oct 28 13:34:45.146: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
  Started install commitinstall_commit: Committing SMU
Executing pre scripts....
Executing pre sripts done.
--- Starting SMU Commit operation ---
Performing SMU_COMMIT on all members

*Oct 28 13:35:24.436: %PLATFORM-4-ELEMENT_WARNING: Switch 1 R0/0: smand: 5/RP/0:
limited space - copy files out of flash: directory. flash: value 84% (1599 MB) exceeds
warning level 70% (1337 MB).
*Oct 28 13:35:30.587: %PLATFORM-4-ELEMENT_WARNING: Switch 1 R0/0: smand: 2/RP/0:
limited space - copy files out of flash: directory. flash: value 74% (1412 MB) exceeds
warning level 70% (1337 MB). [1] SMU_COMMIT package(s) on switch 1
  [1] Finished SMU_COMMIT on switch 1
  [2] SMU_COMMIT package(s) on switch 2
  [2] Finished SMU_COMMIT on switch 2
  [3] SMU_COMMIT package(s) on switch 3
  [3] Finished SMU_COMMIT on switch 3
  [4] SMU_COMMIT package(s) on switch 4
  [4] Finished SMU_COMMIT on switch 4
  [5] SMU_COMMIT package(s) on switch 5
  [5] Finished SMU_COMMIT on switch 5
Checking status of SMU_COMMIT on [1 2 3 4 5]
SMU_COMMIT: Passed on [1 2 3 4 5]
Finished SMU Commit operation

SUCCESS: install_commit /flash/cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
/flash/cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
Sun Oct 28 13:35:52 UTC 2035
Oct 28 13:35:53.789 %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install commit SMU

JJ22-Vore_stack-24TE#
*Oct 28 13:35:53.749: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 1 R0/0: install_engine:
Completed install commit SMU
```

show install summary コマンドを使用してコミットを確認します。SMU パッケージファイルのインストール、アクティブ化、コミットが行われました。ステータスは c です。

```
Device# show install summary
[ Switch 1 2 3 4 5 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   C   flash:cat9k_lite_iosxe.16.12.03.CSCvt22238.SPA.smu.bin
SMU   C   flash:cat9k_lite_iosxe.16.12.03.CSCvt72427.SPA.smu.bin
IMG   C   16.12.3.0.3752
```

```
-----
Auto abort timer: inactive
-----
```

例：SMUのインストール（3ステッププロセス、TFTP：を使用）

次に、3ステッププロセスを使用してSMUパッケージをインストールする例を示します。ここでは、SMUパッケージファイルがリモート（TFTP）ロケーションに保存されます。

1. SMUパッケージファイルを追加します。

```
Device# install add file
tftp://172.16.0.1//tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin

Jun 22 11:32:27.035: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install add
tftp://172.16.0.1//tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Jun 22 11:32:27.035 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install add
tftp://172.16.0.1//tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Downloading file
tftp://172.16.0.1//tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Finished downloading file
tftp://172.16.0.1//tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
to flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
install_add: Adding SMU
install_add: Checking whether new add is allowed ....

--- Starting initial file syncing ---

025335: *Jun 22 2020 11:32:26 UTC: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0:
install_engine: Started install add
tftp://172.16.0.1//tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin[1]:
Copying flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin from switch 1 to
switch 2
[2]: Finished copying to switch 2
Info: Finished copying flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin to the
selected switch(es)
Finished initial file syncing

--- Starting SMU Add operation ---
Performing SMU_ADD on all members
[1] SMU_ADD package(s) on switch 1
[1] Finished SMU_ADD on switch 1
[2] SMU_ADD package(s) on switch 2
[2] Finished SMU_ADD on switch 2
Checking status of SMU_ADD on [1 2]
SMU_ADD: Passed on [1 2]
Finished SMU Add operation

SUCCESS: install_add Mon Jun 22 11:32:56 UTC 2020
Jun 22 11:32:57.598: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install add SMU flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Jun 22 11:32:57.598 %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install add SMU flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin

ECSG-SEC-C9200-24P#
025336: *Jun 22 2020 11:32:57 UTC: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 1 R0/0:
```

```
install_engine: Completed install add SMU
flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
```

show install summary コマンドを使用して追加を確認します。

```
Device# show install summary
[ Switch 1 2 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type St Filename/Version
-----
SMU I flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
IMG C 16.12.02.0.6
-----
Auto abort timer: inactive
-----
```

2. SMU パッケージファイルをアクティブ化します。



(注) (前の手順で) SMU パッケージファイルを追加するために TFTP を使用し、TFTP ではなくフラッシュしてアクティブにします。@@

```
Device# install activate file flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin

install_activate: START Mon Jun 22 11:37:17 UTC 2020

Jun 22 11:37:37.582: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install activate flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Jun 22 11:37:37.582 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install activate flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
install_activate: Activating SMU

025337: *Jun 22 2020 11:37:37 UTC: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0:
install_engine: Started install activate
flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
This operation may require a reload of the system. Do you want to proceed? [y/n]n
```

次のとおり **show version** コマンドを使用して、バージョンを確認します。

```
Device# show version
Cisco IOS XE Software, Version 16.09.04
Cisco IOS Software [Fujii], Catalyst L3 Switch Software (CAT9K_LITE_IOSXE), Version
16.9.4, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2019 by Cisco Systems, Inc.
Compiled Thu 22-Aug-19 17:30 by mcpre
<output truncated>
```

3. SMU パッケージファイルをコミットします。

```
Device# install commit

install_commit: START Mon Jun 22 11:38:48 UTC 2020
SUCCESS: install_commit Mon Jun 22 11:38:52 UTC 2020
```



```
Device#
```

更新パッケージがコミットされてリロードが繰り返されても持続すること確認します。

```
Device# show install summary
```

```
Active Packages:
tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Inactive Packages:
No packages
Committed Packages:
tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Uncommitted Packages:
No packages
Device#
```

例：SMUパッケージの管理（追加のshowコマンド、ロールバック、非アクティブ化）

次の出力例は、**show install summary** コマンドを使用して、アクティブ、非アクティブ、コミット済み、およびコミットされていないパッケージに関する情報を表示します。ここでは、SMU パッケージファイル

cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin がアクティブでコミットされています。

```
Device# show install summary
```

```
Active Packages:
  tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Inactive Packages:
No packages
Committed Packages:
  tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Uncommitted Packages:
No packages
Device#
```

次に、**show install active** コマンドの出力例を示します。

```
Device# show install active
```

```
Active Packages:
tftp:cat3k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
```

次に、更新プログラムパッケージをコミットしたパッケージにロールバックする例を示します。

```
Device# install rollback to base
```

```
install_rollback: START Wed Jun 10 11:27:41 IST 2020
This rollback would require a reload. Do you want to proceed? [y/n]y
2 install_rollback: Reloading the box to take effect
```

```
Initializing Hardware ...
<after reload>
```

```
Device#
```

次に、**show install summary** コマンドの出力例を示します。

```
Device# show install summary

Active Packages:
tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Inactive Packages:
No packages
Committed Packages:
tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Uncommitted Packages:
No packages
Device#
```

次に、**show install log** コマンドの出力例を示します。

```
Device# show install log

[0|install_op_boot]: START Wed Jun 10 19:31:50 Universal 2020
[0|install_op_boot]: END SUCCESS Wed Jun 10 19:31:56 Universal 2020
```

次に、SMU パッケージ ファイルを非アクティブ化する例を示します。

```
Device# install deactivate file tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin

install_deactivate: START Wed Jun 10 10:49:07 IST 2020
The activation step would require a reload. Do you want to proceed? [y/n]y
Regular SMU. Reloading the box to complete activation of the SMU...

Initializing Hardware...
...
<after reload>
Device#
```

次に、**show install summary** コマンドの出力例を示します。

```
Device# show install summary

Active Packages:
No packages
Inactive Packages:tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
Committed Packages:
No packages
Uncommitted Packages:
No packages
Device#
```

次に、デバイスから SMU を削除する例を示します。

```
Device# install remove file tftp:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin

install_remove: START Wed Jun 10 12:09:43 IST 2020
SUCCESS: install_remove /tftp/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin Wed Jun
10 12:09:49 IST 2020
Device#
```

次に、**show install summary** コマンドの出力例を示します。

```
Device# show install summary
```

```
Active Packages:
No packages
Inactive Packages:
No packages
Committed Packages:
No packages
Uncommitted Packages:
No packages
```

ソフトウェアメンテナンスアップグレードのその他の参考資料

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

ソフトウェアメンテナンスアップグレードの機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.4	ソフトウェアメンテナンスアップグレード (SMU)	SMUは、システムにインストールして修正やセキュリティ解決をリリースされたイメージに提供ができるパッケージです。 このプラットフォームでは、SMUにはオペレーティングシステムのコールド (完全) リロードが必要です。ホットパッチはサポートされていません。
Cisco IOS XE Gibraltar 16.10.1	Public Key Infrastructure (PKI)	SMU パッケージは、PKI コンポーネントのパッチ適用をサポートします。
Cisco IOS XE Gibraltar 16.12.1	ソフトウェアメンテナンスアップグレード (SMU)	この機能のサポートは、C9200 SKU で導入されました。ホットパッチはサポートされていません。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 13 章

フラッシュ ファイル システムの操作

- フラッシュ ファイル システムについて (277 ページ)
- 使用可能なファイル システムの表示 (278 ページ)
- デフォルト ファイル システムの設定 (279 ページ)
- ファイル システムのファイルに関する情報の表示 (279 ページ)
- ディレクトリの変更および作業ディレクトリの表示 (281 ページ)
- ディレクトリの作成 (281 ページ)
- ファイルのコピー (282 ページ)
- ファイルの作成、表示、および抽出 (284 ページ)
- フラッシュ ファイル システムに関するその他の関連資料 (286 ページ)
- フラッシュファイルシステムの機能履歴 (286 ページ)

フラッシュ ファイル システムについて

フラッシュ ファイル システムは、ファイルを格納できる単一のフラッシュ デバイスです。ソフトウェア バンドルおよびコンフィギュレーション ファイルの管理に役立つ複数のコマンドも備えています。デバイスのデフォルトのフラッシュファイルシステムは `flash:` です。

アクティブなデバイスから見ると、`flash:` はローカルフラッシュデバイスを指します。これは、ファイルシステムが表示されているのと同じデバイスに接続されているデバイスです。デバイススタックでは、さまざまなスタックメンバからの各フラッシュデバイスを、アクティブなデバイスから参照できます。これらのフラッシュファイルシステムの名前には、対応するデバイスメンバ番号が含まれています。たとえば、アクティブなデバイスから参照できる `flash-3:` は、スタックメンバ 3 にある `flash:` と同じファイルシステムを指します。デバイススタックにあるフラッシュファイルシステムを含む、すべてのファイルシステムのリストを表示するには、`show file systems` 特権 EXEC コマンドを使用します。

デバイススタックでは、一度に 1 人のユーザのみが、ソフトウェアバンドルおよびコンフィギュレーション ファイルを管理できます。

使用可能なファイル システムの表示

デバイスで使用可能なファイルシステムを表示するには、**show file systems** 特権 EXEC コマンドを使用します（次のスタンドアロンデバイスの例を参照）。

表 10: *show file systems* のフィールドの説明

フィールド	値
Size(b)	ファイル システムのメモリ サイズ (バイト単位) です。
Free(b)	ファイル システムの空きメモリ サイズ (バイト単位) です。
Type	<p>ファイル システムのタイプです。</p> <p>disk : ファイル システムは、フラッシュメモリ デバイス、USB フラッシュ、crashinfo ファイル用です。</p> <p>network : ファイル システムは、FTP サーバや HTTP サーバなどのネットワーク デバイス用です。</p> <p>nvram : ファイル システムは NVRAM (不揮発性 RAM) デバイス用です。</p> <p>opaque : ファイル システムは、ローカルに生成された pseudo ファイル システム (system など)、またはダウンロード インターフェイス (brimux など) です。</p> <p>unknown : ファイル システムのタイプは不明です。</p>
Flags	<p>ファイル システムの権限です。</p> <p>ro : 読み取り専用です。</p> <p>rw : 読み取りおよび書き込みです。</p> <p>wo : 書き込み専用です。</p>

フィールド	値
Prefixes	<p>ファイル システムのエイリアスです。</p> <p>crashinfo : crashinfo ファイルです。</p> <p>flash : フラッシュ ファイル システムです。</p> <p>ftp : FTP サーバです。</p> <p>http : HTTP サーバです。</p> <p>https : セキュア HTTP サーバです。</p> <p>nvr : NVRAM です。</p> <p>null : コピーのヌル宛先です。リモート ファイルをヌルへコピーして、サイズを判別できます。</p> <p>rcp : Remote Copy Protocol (RCP) サーバです。</p> <p>scp : Session Control Protocol (SCP) サーバです。</p> <p>system : 実行コンフィギュレーションを含むシステムメモリが格納されています。</p> <p>tftp : TFTP ネットワーク サーバです。</p> <p>usbflash0 : USB フラッシュ メモリです。</p> <p>ymodem : YMODEM プロトコルを使用して、ネットワーク マシンからファイルを取得します。</p>

デフォルト ファイル システムの設定

デフォルトのファイル システムとして使用されるファイル システムまたはディレクトリを指定するには、**cd filesystem:** 特権 EXEC コマンドを使用します。デフォルト ファイル システムを設定すると、関連するコマンドを実行するときに *filesystem:* 引数を省略できます。たとえば、オプションの *filesystem:* 引数を持つすべての特権 EXEC コマンドでは、**cd** コマンドで指定されたファイル システムが使用されます。

デフォルトでは、デフォルト ファイル システムは *flash:* です。

cd コマンドで指定された現在のデフォルトのファイル システムを表示するには、**pwd** 特権 EXEC コマンドを使用します。

ファイル システムのファイルに関する情報の表示

ファイル システムの内容を操作する前に、そのリストを表示できます。たとえば、新しいコンフィギュレーション ファイルをフラッシュ メモリにコピーする前に、ファイル システムに同じ名前のコンフィギュレーション ファイルが格納されていないことを確認できます。同様に、

フラッシュ コンフィギュレーション ファイルを別の場所にコピーする前に、ファイル名を確認して、その名前を別のコマンドで使用できます。ファイル システムのファイルに関する情報を表示するには、次の表に記載する特権 EXEC コマンドのいずれかを使用します。

表 11: ファイルに関する情報を表示するためのコマンド

コマンド	説明
dir [/all] [filesystem:filename]	ファイル システムのファイル リストを表示します。
show file systems	ファイル システムのファイルごとの詳細を表示します。
show file information file-url	特定のファイルに関する情報を表示します。
show file descriptors	開いているファイルの記述子のリストを表示します。ファイル記述子は開いているファイルの内部表現です。このコマンドを使用して、別のユーザによってファイルが開かれているかどうかを調べることができます。

たとえば、ファイル システムのすべてのファイルのリストを表示するには、次のように **dir** 特権 EXEC コマンドを使用します。

```
Device# dir flash:
DDirectory of bootflash:/

616513 drwx          4096 Jul 15 2015 07:11:35 +00:00 .installer
608402 -rw-          33818 Sep 25 2015 11:41:35 +00:00 bootloader_evt_handle.log
608403 drwx          4096 Feb 27 2017 13:56:47 +00:00 .ssh
608410 -rw-           0 Jun 5 2015 10:16:17 +00:00 dc_stats.txt
608411 drwx          20480 Sep 23 2015 11:50:13 +00:00 core
624625 drwx          4096 Sep 23 2015 12:29:27 +00:00 .prst_sync
640849 drwx          4096 Feb 27 2017 13:57:30 +00:00 .rollback_timer
608412 drwx          4096 Jun 17 2015 18:12:47 +00:00 orch_test_logs
608413 -rw-          33554432 Sep 25 2015 11:43:15 +00:00 nvram_config
608417 -rw-           35 Sep 25 2015 20:17:42 +00:00 pnp-tech-time
608439 -rw-          214054 Sep 25 2015 20:17:48 +00:00 pnp-tech-discovery-summary
608419 drwx          4096 Jul 23 2015 07:50:25 +00:00 util
616514 drwx          4096 Mar 18 2015 11:09:04 +00:00 onep
608442 -rw-           556 Mar 18 2015 11:19:34 +00:00 vlan.dat
608448 -rw-          1131779 Mar 28 2015 13:13:48 +00:00 log.txt
616516 drwx          4096 Apr 1 2015 09:34:56 +00:00 gs_script
616517 drwx          4096 Apr 6 2015 09:42:38 +00:00 tools
608440 -rw-           252 Sep 25 2015 11:41:52 +00:00 boothelper.log
624626 drwx          4096 Apr 17 2015 06:10:55 +00:00 SD_AVC_AUTO_CONFIG
608488 -rw-          98869 Sep 25 2015 11:42:15 +00:00 memleak.tcl
608437 -rw-          17866 Jul 16 2015 04:01:10 +00:00 ardbeg_x86
632745 drwx          4096 Aug 20 2015 11:35:09 +00:00 CRDU
632746 drwx          4096 Sep 16 2015 08:57:44 +00:00 ardmore
608418 -rw-          1595361 Jul 8 2015 11:18:33 +00:00
system-report_RP_0_20150708-111832-UTC.tar.gz
608491 -rw-          67587176 Aug 12 2015 05:30:35 +00:00 mcln_x86_kernel_20170628.SSA
608492 -rw-          74880100 Aug 12 2015 05:30:57 +00:00 stardust.x86.idprom.0718B

11250098176 bytes total (9128050688 bytes free)
Device#
```


ディレクトリの変更および作業ディレクトリの表示

ディレクトリを変更し、作業ディレクトリを表示するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	dir filesystem: 例： Device# dir flash:	指定されたファイル システムのディレクトリを表示します。 <i>filesystem:</i> には、システム ボードのフラッシュ デバイスの <i>flash:</i> を使用します。
ステップ 3	cd directory_name 例： Device# cd new_configs	指定されたディレクトリへ移動します。 コマンド例では、 <i>new_configs</i> という名前のディレクトリに移動する方法を示します。
ステップ 4	pwd 例： Device# pwd	作業ディレクトリを表示します。
ステップ 5	cd 例： Device# cd	デフォルトディレクトリに移動します。

ディレクトリの作成

特権 EXEC モードを開始して、ディレクトリを作成するには次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	dir filesystem: 例 : Device# dir flash:	指定されたファイル システムのディレクトリを表示します。 <i>filesystem:</i> には、システム ボードのフラッシュ デバイスの flash: を使用します。
ステップ 2	mkdir directory_name 例 : Device# mkdir new_configs	新しいディレクトリを作成します。スラッシュ (/) 間に指定できるディレクトリ名は最大 45 文字で、大文字と小文字の区別があります。ディレクトリ名には制御文字、スペース、スラッシュ、引用符、セミコロン、またはコロンは使用できません。
ステップ 3	dir filesystem: 例 : Device# dir flash:	入力を確認します。

ディレクトリの削除

ディレクトリを、その内部のすべてのファイルおよびサブディレクトリとともに削除するには、**delete /force /recursive filesystem:/file-url** 特権 EXEC コマンドを使用します。

名前で指定されたディレクトリを、その内部のすべてのサブディレクトリおよびファイルとともに削除するには、**/recursive** キーワードを使用します。ディレクトリ内のファイルごとに表示される、削除を確認するためのプロンプトを省略するには、**/force** キーワードを使用します。この削除プロセスを実行すると、最初に 1 度だけプロンプトが表示されます。

filesystem には、システム ボードのフラッシュ デバイスの **flash:** を使用します。*file-url* には、削除するディレクトリの名前を入力します。ディレクトリ内のすべてのファイルおよびディレクトリが削除されます。



注意 ディレクトリが削除された場合、その内容は回復できません。

ファイルのコピー

送信元から宛先にファイルをコピーするには、**copy source-url destination-url** 特権 EXEC コマンドを使用します。送信元および宛先の URL には、**running-config** および **startup-config** キーワードショートカットを使用できます。たとえば、**copy running-config startup-config** コマンドは、

現在実行中のコンフィギュレーション ファイルをフラッシュメモリの NVRAM セクションに保存し、システム初期化の際にコンフィギュレーションファイルとして使用されるようにします。

XMODEM または YMODEM プロトコルを使用するネットワーク マシンのファイルに対する送信元として特殊なファイルシステム (**xmodem:**、**ymodem:**) を指定し、そこからコピーすることもできます。

ネットワーク ファイルシステムの URL には、ftp:、rcp:、tftp:、scp:、http:、https: などがあり、構文は次のとおりです。

- FTP : ftp:[[/username [:password]@location]/directory]/filename
- RCP : rcp:[[/username@location]/directory]/filename
- TFTP : tftp:[[/location]/directory]/filename
- SCP : scp:[[/username [:password]@location]/directory]/filename
- HTTP : http:[[/username [:password]@location]/directory]/filename
- HTTPS : https:[[/username [:password]@location]/directory]/filename



(注) パスワードに特殊文字「@」を含めることはできません。文字「@」を使用すると、コピーでサーバの IP アドレスを解析できません。

ローカルにある書き込み可能なファイル システムには flash: などがあります。

送信元および宛先の組み合わせによっては、無効な場合があります。特に、次に示す組み合わせの場合は、コピーできません。

- 実行コンフィギュレーションから実行コンフィギュレーションへ
- スタートアップ コンフィギュレーションからスタートアップ コンフィギュレーションへ

ファイルの削除

フラッシュ メモリ デバイスのファイルが不要になった場合は、そのファイルを永久に削除できます。指定されたフラッシュ デバイスからファイルまたはディレクトリを削除するには、**delete [/force] [/recursive] [filesystem:] /file-url** 特権 EXEC コマンドを使用します。

ディレクトリを、その内部のすべてのサブディレクトリやファイルとともに削除するには、**/recursive** キーワードを使用します。ディレクトリ内のファイルごとに表示される、削除を確認するためのプロンプトを省略するには、**/force** キーワードを使用します。この削除プロセスを実行すると、最初に 1 度だけプロンプトが表示されます。**archive download-sw** コマンドでインストールされ、不要になった古いソフトウェアイメージを削除するには、**/force** キーワードおよび **/recursive** キーワードを使用します。

filesystem: オプションを省略すると、デバイスは **cd** コマンドで指定したデフォルトのデバイスを使用します。 *file-url* には、削除するファイルのパス（ディレクトリ）および名前を指定します。

ファイルを削除しようとする、削除の確認を求めるプロンプトが表示されます。



注意 ファイルが削除された場合、その内容は回復できません。

ここでは、デフォルトのフラッシュ メモリ デバイスからファイル *myconfig* を削除する例を示します。

```
Device# delete myconfig
```

ファイルの作成、表示、および抽出

ファイルを作成してそこにファイルを書き込んだり、ファイル内のファイルをリスト表示したり、ファイルからファイルを抽出したりできます（次の項を参照）。

ファイルの作成、内容の表示、およびファイルの抽出を行うには、特権 EXEC コマンドで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>archive tar /create destination-url flash: /file-url</p> <p>例 :</p> <pre>Device# archive tar /create tftp:172.20.10.30/saved. flash:/new-configs</pre>	<p>ファイルを作成し、そこにファイルを追加します。</p> <p>destination-url には、ローカルまたはネットワーク ファイル システムの宛先 URL のエイリアス、および作成するファイルの名前を指定します。</p> <ul style="list-style-type: none"> ローカルフラッシュ ファイル システム構文 <p>flash:</p> <ul style="list-style-type: none"> FTP 構文 <p>ftp://[username][password]@[location]/directory/-filename.</p> <ul style="list-style-type: none"> RCP 構文 <p>rnp://[username@location]/directory/-filename.</p> <ul style="list-style-type: none"> TFTP 構文 <p>tftp://[location]/directory/-filename.</p> <p>flash:/file-url には、ローカルフラッシュ ファイル システム上の、新しいファイ</p>

	コマンドまたはアクション	目的
		ルが作成される場所を指定します。送信元ディレクトリ内に格納されている任意のファイルまたはディレクトリの一覧を指定して、新しいファイルに追加することもできます。何も指定しないと、このレベルにおけるすべてのファイルおよびディレクトリが、新規に作成されたファイルに書き込まれます。
ステップ 2	archive tar /table source-url 例 : <pre>Device# archive tar /table flash: /new_configs</pre>	ファイルの内容を表示します。 <i>source-url</i> には、ローカルファイルシステムまたはネットワーク ファイルシステムの送信元 URL エイリアスを指定します。- <i>filename.</i> は、表示するファイルです。次のオプションがサポートされています。 <ul style="list-style-type: none"> ローカルフラッシュファイルシステム構文 flash: <ul style="list-style-type: none"> FTP 構文 ftp://[username][password]@[location]/[directory]/-filename. RCP 構文 rcp://[username@[location]/[directory]/-filename. TFTP 構文 tftp://[location]/[directory]/-filename. ファイルのあとにファイルまたはディレクトリのリストを指定して、ファイルの表示を制限することもできます。指定したファイルだけが表示されます。何も指定しないと、すべてのファイルおよびディレクトリが表示されます。
ステップ 3	archive tar /xtract source-url flash:/file-url [dir/file...] 例 : <pre>Device# archive tar /xtract tftp://172.20.10.30/saved. flash:/new-configs</pre>	ファイルをフラッシュ ファイルシステム上のディレクトリに抽出します。 <i>source-url</i> には、ローカルファイルシステムの送信元 URL のエイリアスを指定します。- <i>filename.</i> は、ファイルの抽出元のファイルです。次のオプションがサポートされています。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> ローカルフラッシュ ファイル システム構文 <p>flash:</p> <ul style="list-style-type: none"> FTP 構文 <p>ftp://[<i>username</i>][<i>password</i>]@<i>location</i>]/<i>directory</i>]/<i>filename</i>.</p> <ul style="list-style-type: none"> RCP 構文 <p>rnp://[<i>username</i>@<i>location</i>]/<i>directory</i>]/<i>filename</i>.</p> <ul style="list-style-type: none"> TFTP 構文 <p>tftp://[<i>location</i>]/<i>directory</i>]/<i>filename</i>.</p> <p>flash:/<i>file-url</i> [<i>dir/file...</i>]には、ファイルの抽出元にするローカルフラッシュファイルシステム上の場所を指定します。抽出対象のファイル内のファイルまたはディレクトリのリストを指定するには、<i>dir/file...</i> オプションを使用します。何も指定されないと、すべてのファイルとディレクトリが抽出されます。</p>
ステップ 4	<p>more [/<i>ascii</i> /<i>binary</i> /<i>ebcdic</i>] /<i>file-url</i></p> <p>例 :</p> <pre>Device# more flash:/new-configs</pre>	<p>リモートファイルシステム上のファイルを含めて、読み取り可能なファイルの内容を表示します。</p>

フラッシュファイルシステムに関するその他の関連資料

関連資料

関連項目	マニュアル タイトル
flash: ファイルシステムの管理コマンド	<i>Cisco IOS Configuration Fundamentals Command Reference</i>

フラッシュファイルシステムの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	フラッシュファイルシステム	フラッシュファイルシステムは、ファイルを格納できる単一のフラッシュ デバイスです。ソフトウェア バンドルおよびコンフィギュレーション ファイルの管理に役立つ複数のコマンドも備えています。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 14 章

初期設定へのリセットの実行

- [初期設定へのリセット実行の前提条件 \(289 ページ\)](#)
- [初期設定へのリセット実行の制限事項 \(289 ページ\)](#)
- [初期設定へのリセットに関する情報 \(290 ページ\)](#)
- [初期設定へのリセットの実行方法 \(291 ページ\)](#)
- [初期設定へのリセット実行の設定例 \(292 ページ\)](#)
- [初期設定へのリセットに関するその他の参考資料 \(292 ページ\)](#)
- [初期設定へのリセットに関する機能履歴 \(292 ページ\)](#)

初期設定へのリセット実行の前提条件

- 初期設定へのリセットプロセスを開始する前に、現在のイメージ、設定、および個人データを含むすべてのソフトウェアイメージがバックアップされていることを確認します。
- 初期設定へのリセットプロセスが進行中の場合は、電源の中断がないことを確認します。
- 初期設定へのリセットプロセスを開始する前に、In-Service Software Upgrade (ISSU) または In-Service Software Downgrade (ISSD) が進行中でないことを確認します。

初期設定へのリセット実行の制限事項

- ソフトウェアパッチは、デバイスにインストールされている場合、初期設定へのリセットプロセス後に復元されません。
- VTYセッションを介して **factory-reset** コマンドが発行された場合、初期設定へのリセットプロセスの完了後にセッションは復元されません。
- 初期設定へのリセットはスタンドアロンモードでのみサポートされ、スタッキングモードではサポートされません。高可用性 (HA) モードで設定されたモジュラシャーシデバイスの場合、各スーパーバイザモジュールに工場出荷時設定へのリセットが適用されます。

初期設定へのリセットに関する情報

初期設定にリセットすると、デバイスに保存されているお客様固有のデータがすべて消去され、デバイスの設定は出荷時の元の設定に復元されます。消去されるデータには、設定、ログファイル、ブート変数、コアファイル、および連邦情報処理標準（FIPS）関連キーなどのクレデンシャルが含まれます。

初期設定へのリセットプロセスは、次の2つのシナリオで使用されます。

- デバイスの返品許可（RMA）：RMAのためにデバイスをシスコに返送する必要がある場合は、そのデバイスの RMA 証明書を取得する前に、お客様固有のデータをすべて削除してください。
- 侵害を受けたデバイスのリカバリ：デバイスに保存されているキーマテリアルまたはクレデンシャルが侵害を受けた場合は、デバイスを初期設定にリセットし、デバイスを再設定してください。

初期設定へのリセット時、デバイスはリロードされ、ROMmon モードを開始します。初期設定へのリセット後、デバイスは、ソフトウェアの検索とロードに必要な **MAC_ADDRESS** 変数と **SERIAL_NUMBER** 変数を含むすべての環境変数を削除します。ROMmon モードでリセットを実行すると、環境変数は自動的に設定されます。

ROMmon モードでのシステムリセットが完了したら、USB または TFTP を使用して Cisco IOS イメージを追加します。

次の表に、初期設定へのリセットプロセス中に消去および保持されるデータの詳細を示します。

表 12: 初期設定へのリセット時に消去および保持されるデータ

消去されるデータ	保持されるデータ
現在のブートイメージを含むすべての Cisco IOS イメージ	リモート Field-Replaceable Unit (FRU) からのデータ
クラッシュ情報とログ	コンフィギュレーションレジスタの値
ユーザデータ、スタートアップおよび実行コンフィギュレーション、および Serial Advanced Technology Attachment (SATA)、SSD、USB などのリムーバブルストレージデバイスの内容	—
FIPS 関連キーなどのクレデンシャル	セキュアな固有デバイス識別子 (SUDI) 証明書、公開キーインフラストラクチャ (PKI) キーなどのクレデンシャル
オンボード障害ロギング (OBFL) ログ	

消去されるデータ	保持されるデータ
ユーザが追加した ROMmon 変数	—

初期設定へのリセットの実行方法

初期設定へのリセットを実行するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	factory-reset {all config boot-vars} 例： Device# factory-reset all	デバイスを出荷時の設定にリセットします。 factory reset コマンドを使用するために必要なシステム設定はありません。 次のオプションを使用できます。 <ul style="list-style-type: none"> • all : NVRAM のすべての内容、現在のブートイメージ、ブート変数、起動コンフィギュレーションと実行コンフィギュレーションのデータ、およびユーザデータを含むすべての Cisco IOS イメージを消去します。シスコでは all オプションの使用を推奨しています。 • config : スタートアップ コンフィギュレーションをリセットします。 • boot-vars : ユーザによって追加されたブート変数を消去します。 初期設定へのリセットプロセスが正常に完了すると、デバイスがリブートして ROMmon モードになります。

初期設定へのリセット実行の設定例

次に、初期設定へのリセットを実行する例を示します。

```
Device> enable
Device# factory-reset all
The factory reset operation is irreversible for all operations. Are you sure? [confirm]
The following will be deleted as a part of factory reset:
1: Crash info and logs
2: User data, startup and running configuration
3: All IOS images, including the current boot image
4: OBFL logs
5: User added rommon variables
6: Data on Field Replaceable Units (USB/SSD/SATA)
The system will reload to perform factory reset.
It will take some time to complete and bring it to rommon.
You will need to load IOS image using USB/TFTP from rommon after
this operation is completed.
DO NOT UNPLUG THE POWER OR INTERRUPT THE OPERATION
Are you sure you want to continue? [confirm]
```

初期設定へのリセットに関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	コマンドリファレンス

初期設定へのリセットに関する機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	工場出荷時の状態へのリセット (Factory Reset)	初期設定にリセットすると、デバイスに保存されているお客様固有のデータがすべて消去され、デバイスの設定は出荷時の元の設定に復元されます。

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.12.1	リムーバブルストレージデバイスの初期設定へのリセット	初期設定へのリセットを実行すると、SATA、SSD、USB などのリムーバブルストレージデバイスの内容が消去されます。
Cisco IOS XE Amsterdam 17.2.1	3-pass 上書きによる初期設定へのリセット	初期設定へのリセットを実行すると、デバイスからすべてのコンテンツを 3-pass 上書きで安全に消去できます。secure 3-pass キーワードが導入されました。
	スタックおよび Cisco StackWise Virtual の初期設定へのリセットオプションの拡張	スタック構成デバイスおよび Cisco StackWise Virtual 対応デバイスで初期設定へのリセットのサポートが導入されました。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 15 章

セキュアストレージの設定

- [セキュアストレージについて \(295 ページ\)](#)
- [セキュアストレージの有効化 \(295 ページ\)](#)
- [セキュアストレージの無効化 \(296 ページ\)](#)
- [暗号化のステータスの確認 \(297 ページ\)](#)
- [セキュアストレージの機能情報 \(297 ページ\)](#)

セキュアストレージについて

セキュアストレージ機能では、重要な設定情報を暗号化して保護できます。非対称キーペア、事前共有秘密、タイプ 6 のパスワード暗号化キーおよび特定のクレデンシャルを暗号化します。インスタンス固有の暗号キーは、危険にさらされることを防ぐためにハードウェアのトラストアンカーに保管されます。

セキュアストレージの有効化

始める前に

この機能はデフォルトで無効になっています。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 2	service private-config-encryption 例： Device(config)# service private-config-encryption	デバイスでセキュアストレージ機能を有効にします。

	コマンドまたはアクション	目的
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 4	write memory 例： Device# write memory	private-config ファイルを暗号化し、暗号化フォーマットで保存します。

セキュアストレージの無効化

始める前に

デバイスでセキュアストレージ機能を無効にするには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	no service private-config-encryption 例： Device(config)# no service private-config-encryption	デバイスでセキュリティストレージ機能を無効にします。セキュアストレージを無効にすると、すべてのユーザデータがプレーンテキストで NVRAM に保存されます。
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 4	write memory 例： Device# write memory	private-config ファイルを復号し、プレーンフォーマットで保存します。

暗号化のステータスの確認

暗号化のステータスを確認するには、**show parser encrypt file status** コマンドを使用します。次のコマンド出力は、機能は利用できるが、ファイルが暗号化されていないことを示します。ファイルは「プレーンテキスト」形式です。

```
Device#show parser encrypt file status
Feature: Enabled
File Format: Plain Text
Encryption Version: Ver1
```

セキュアストレージの機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	セキュアなストレージ	セキュアストレージ機能では、重要な設定情報を暗号化して保護できます。非対称キーペア、事前共有秘密、タイプ6のパスワード暗号化キーおよび特定のクレデンシャルを暗号化します。インスタンス固有の暗号キーは、危険にさらされることを防ぐためにハードウェアのトラストアンカーに保管されます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 16 章

条件付きデバッグとラジオアクティブトレース

- [条件付きデバッグの概要 \(299 ページ\)](#)
- [ラジオアクティブトレースの概要 \(300 ページ\)](#)
- [条件付きデバッグとラジオアクティブトレースの設定方法 \(300 ページ\)](#)
- [条件付きデバッグのモニタリング \(304 ページ\)](#)
- [条件付きデバッグの設定例 \(305 ページ\)](#)
- [条件付きデバッグとラジオアクティブトレースに関するその他の関連資料 \(305 ページ\)](#)
- [条件付きデバッグとラジオアクティブトレースの機能履歴 \(306 ページ\)](#)

条件付きデバッグの概要

条件付きデバッグ機能によって、定義した条件に基づき、特定の機能のデバッグおよびロギングを選択して有効にすることができます。この機能は、多くの機能がサポートされているシステムで有用です。



(注) コントロールプレーントレースのみがサポートされています。

条件付きデバッグでは、多数の機能が導入されていて大規模に稼働しているネットワークにおけるきめ細かなデバッグが可能です。これにより、システム内の細かなインスタンスに対しても、詳細なデバッグを実行できます。これは、何千ものセッションのうち特定のセッションのみをデバッグするような場合に、非常に有用です。条件は複数指定することもできます。

条件とは、機能またはアイデンティティをいいます。アイデンティティは、インターフェイス、IP アドレス、MAC アドレスなどです。



(注) サポートされる条件は MAC アドレスであることのみです。

これは、処理する機能オブジェクトを区別せずに出力を生成する、一般的なデバッグコマンドとは対照的です。一般的なデバッグ コマンドは、多数のシステム リソースを消費し、システム パフォーマンスに影響します。

ラジオアクティブトレースの概要

ラジオアクティブトレースにより、冗長性のレベルを高めた状態で、システムの全体にわたって目的とする動作を連鎖的に実行できます。また、複数のスレッド、プロセス、および関数呼び出しにわたって、デバッグ情報を条件に基づいて（DEBUG レベルまで、または指定のレベルまで）出力する方法を提供します。



(注) デフォルトのレベルは **DEBUG** です。ユーザは別のレベルに変更することはできません。

ラジオアクティブトレースでは、次の機能が有効になっています。

- IGMP スヌーピング
- レイヤ 2 マルチキャスト

条件付きデバッグとラジオアクティブトレースの設定方法

条件付きデバッグおよび放射線トレース

条件付きデバッグと組み合わせた放射線トレースによって、条件に関連するすべての実行コンテキストをデバッグする単一のデバッグ CLI を取得できます。これは、ボックス内の機能のさまざまな制御フロー プロセスを認識していなくても行うことができ、これらのプロセスでデバッグを個別に発行する必要もありません。

トレースファイルの場所

デフォルトでは、トレースファイル ログは各プロセスで生成され、`/tmp/rp/trace` または `/tmp/fp/trace` ディレクトリに保存されます。この一時ディレクトリで、トレースログがファイルに書き込まれます。各ファイルは 1 MB サイズです。このディレクトリでは、特定のプロセスのこうしたファイルを、最大 25 件保持できます。`/tmp` ディレクトリのトレースファイルがその 1 MB 制限またはブート時に設定されたサイズに達した場合、ローテーションから外れ、`tracelogs` ディレクトリの `/crashinfo` パーティションの下にあるアーカイブの場所に移動します。

`/tmp` ディレクトリが1つのプロセスで保持するトレースファイルは1つのみです。ファイルがそのファイルサイズの制限に達すると、ローテーションから外れ、`/crashinfo/tracelogs` に移動します。アーカイブ ディレクトリに蓄積されるファイルは最大 25 ファイルであり、その後は最も古いものから順に、`/tmp` から新たにローテーションされたファイルに置換されます。

`crashinfo` ディレクトリ内のトレースファイルは次の形式で配置されます。

1. `Process-name_Process-ID_running-counter.timestamp.gz`
例 : `IOSRP_R0-0.bin_0.14239.20151101234827.gz`
2. `Process-name_pmanlog_Process-ID_running-counter.timestamp.bin.gz`
例 : `wcm_pmanlog_R0-0.30360_0.20151028233007.bin.gz`

条件付きデバッグの設定

条件付デバッグを設定するには、以下の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : <code>Device> enable</code>	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	debug platform condition mac <code>{mac-address}</code> 例 : <code>Device# debug platform condition mac bc16.6509.3314</code>	指定された MAC アドレスの条件付きデバッグを設定します。
ステップ 3	debug platform condition start 例 : <code>Device# debug platform condition start</code>	条件付きデバッグを開始します (上記のいずれかの条件に一致すると放射線トレースを開始します)。
ステップ 4	show platform condition または show debug 例 : <code>Device# show platform condition</code> <code>Device# show debug</code>	現在設定されている条件を表示します。
ステップ 5	debug platform condition stop 例 : <code>Device# debug platform condition stop</code>	条件付きデバッグを停止します (放射線トレースを停止します)。

	コマンドまたはアクション	目的
ステップ 6	request platform software trace archive [last {number} days] [target {crashinfo: flashinfo:}] 例 : <pre># request platform software trace archive last 2 days</pre>	(任意) システムのマージされたトレースファイルの履歴ログを表示します。日数またはロケーションの組み合わせのフィルタ。
ステップ 7	show platform software trace [filter-binary level message] 例 : <pre>Device# show platform software trace message</pre>	(任意) 最新のトレースファイルからマージされたログを表示します。アプリケーションの状態、トレース モジュール名およびトレース レベルをさまざまな組み合わせでフィルタリングします。 <ul style="list-style-type: none"> • filter-binary : 照合するモジュールをフィルタリングします。 • level : トレース レベルを表示します。 • message : トレース メッセージのリングの内容を表示します。 (注) デバイス上では次が可能です。 <ul style="list-style-type: none"> • Linux シェルだけでなく、IOS のコンソールからも使用できます。 • マージされたログでファイルを生成します。 • ステージング エリアからのみマージされたログを表示します。
ステップ 8	clear platform condition all 例 : <pre>Device# clear platform condition all</pre>	すべての条件をクリアします。

次のタスク



(注) **request platform software trace filter-binary** コマンドと **show platform software trace filter-binary** コマンドは同様の動作をします。唯一の違いは次のとおりです。

- **request platform software trace filter-binary** : データ ソースとして履歴ログを使用します。
- **show platform software trace filter-binary** : データ ソースとしてフラッシュの一時ディレクトリを使用します。

その中でも、`mac_log <..date.>` は、デバッグする MAC 用のメッセージを伝えるため、最も重要なファイルです。コマンド **show platform software trace filter-binary** も同じフラッシュ ファイルを生成し、また、画面に `mac_log` を出力します。

L2 マルチキャストの放射線トレース

特定のマルチキャスト受信者を特定するには、参加者または受信側クライアントの MAC アドレス、グループのマルチキャスト IP アドレスおよびスヌーピング VLAN を指定します。また、デバッグのトレース レベルを有効にします。デバッグ レベルでは、詳細なトレースとシステムへの高い可視性が提供されます。

```
debug platform condition feature multicast controlplane mac client MAC address ip Group
IP address vlan id level debug level
```

トレース ファイルの推奨ワークフロー

トレース ファイルの推奨ワークフローの概要は次のとおりです。

1. 特定の時間帯のトレースログを要求する場合。
たとえば 1 日。
使用するコマンドは、次のとおりです。
`Device#request platform software trace archive last 1 day`
2. システムは、`/flash:` ロケーション内のトレースログの tar ball (`.gz` ファイル) を生成します。
3. スイッチ外にファイルをコピーします。ファイルをコピーすることによって、オフラインでトレースログが使用できます。ファイルのコピーについての詳細は、次のセクションを参照してください。
4. `/flash: location` からトレースログファイル (`.gz`) ファイルを削除します。これにより、他の操作に十分な領域がスイッチに確保されます。

ボックス外へのトレース ファイルのコピー

トレース ファイルの例を以下に示します。

```
Device# dir crashinfo:/tracelogs
Directory of crashinfo:/tracelogs/

50664 -rwx 760 Sep 22 2015 11:12:21 +00:00 plogd_F0-0.bin_0.gz
```

```

50603 -rwx 991 Sep 22 2015 11:12:08 +00:00 fed_pmanlog_F0-0.bin_0.9558.20150922111208.gz
50610 -rw- 11 Nov 2 2015 00:15:59 +00:00 timestamp
50611 -rwx 1443 Sep 22 2015 11:11:31 +00:00
auto_upgrade_client_sh_pmanlog_R0-.bin_0.3817.20150922111130.gz
50669 -rwx 589 Sep 30 2015 03:59:04 +00:00 cfgwr-8021_R0-0.bin_0.gz
50612 -rwx 1136 Sep 22 2015 11:11:46 +00:00 reflector_803_R0-0.bin_0.1312.20150922111116.gz
50794 -rwx 4239 Nov 2 2015 00:04:32 +00:00 IOSRP_R0-0.bin_0.14239.20151101234827.gz
50615 -rwx 131072 Nov 2 2015 00:19:59 +00:00 linux_iosd_image_pmanlog_R0-0.bin_0
--More-

```

トレース ファイルは、次に示すさまざまなオプションのいずれかを使用して、コピーできます。

```

Device# copy crashinfo:/tracelogs ?
crashinfo: Copy to crashinfo: file system
flash: Copy to flash: file system
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system

```

TFTP サーバにコピーするための一般的な構文は次のとおりです。

```

Device# copy source: tftp:
Device# copy crashinfo:/tracelogs/IOSRP_R0-0.bin_0.14239.20151101234827.gz tftp:
Address or name of remote host []? 2.2.2.2
Destination filename [IOSRP_R0-0.bin_0.14239.20151101234827.gz]?

```



(注) tracelog および他の目的に使用可能な空き容量があることを確認するために、生成されたレポート/アーカイブ ファイルをスイッチからクリアすることが重要です。

条件付きデバッグのモニタリング

以下の表に、条件付きデバッグのモニタに使用できる各種コマンドを示します。

コマンド	目的
show platform condition	現在設定されている条件を表示します。
show debug	現在設定されているデバッグ条件を表示します。

コマンド	目的
show platform software trace filter-binary	最新のトレース ファイルからマージされたログを表示します。
request platform software trace filter-binary	システムにマージされたトレース ファイルの履歴ログを表示します。

条件付きデバッグの設定例

次に、`show platform condition` コマンドの出力例を示します。

```
Device# show platform condition
Conditional Debug Global State: Stop
Conditions Direction
```

```
-----|-----
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
```

```
-----|-----
Device#
```

次に、`show debug` コマンドの出力例を示します。

```
Device# show debug
IOSXE Conditional Debug Configs:
Conditional Debug Global State: Start
Conditions Direction
```

```
-----|-----
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
```

```
-----|-----
Packet Infra debugs:
Ip Address Port
```

```
-----|-----
Device#
```

次に、`debug platform condition stop` コマンドの例を示します。

```
Device# debug platform condition stop
Conditional Debug Global State: Stop
```

条件付きデバッグとラジオアクティブトレースに関するその他の関連資料

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

条件付きデバッグとラジオアクティブトレースの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	条件付きデバッグとラジオアクティブトレース	条件付きデバッグ機能によって、定義した条件に基づき、特定の機能のデバッグおよびロギングを選択して有効にすることができます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 17 章

同意トークン

- [同意トークンの制約事項 \(307 ページ\)](#)
- [同意トークンに関する情報 \(308 ページ\)](#)
- [システムシェルアクセスの同意トークン承認プロセス \(308 ページ\)](#)
- [同意トークンの機能履歴 \(310 ページ\)](#)

同意トークンの制約事項

- 同意トークンはデフォルトで有効であり、無効にすることはできません。
- デバイスからチャレンジが送信された後、30分以内に応答を入力する必要があります。入力しないとチャレンジが期限切れになり、新しいチャレンジの要求が必要になります。
- 単一の応答は、対応するチャレンジに対して1回だけ有効です。
- ルートシェルアクセスの最大承認タイムアウトは7日間です。
- スイッチオーバーイベント後、既存の同意トークンベースの承認はすべて期限切れとして処理されます。その後、サービスアクセスの新しい認証シーケンスを再起動する必要があります。
- シスコのチャレンジ署名サーバ上の同意トークン応答生成にアクセスできるのは、シスコ認定担当者のみです。
- システムシェルアクセスのシナリオでは、承認タイムアウトが発生するか、または同意トークン終了承認コマンドによってシェル承認が明示的に終了されるまで、シェルを終了しても承認は終了しません。

システムシェルアクセスの目的を達成したら、同意トークン終了コマンドを明示的に発行することによって、システムシェルの承認を強制終了することを推奨します。

同意トークンに関する情報

同意トークンは、ネットワーク管理者と Cisco Technical Assistance Centre (Cisco TAC) の相互の同意により、システムシェルにアクセスする組織のネットワーク管理者を認証するために使用されるセキュリティ機能です。

一部のデバッグシナリオでは、Cisco TAC エンジニアが特定のデバッグ情報を収集したり、実稼働システムでライブデバッグを実行する必要がある場合があります。このような場合、Cisco TAC エンジニアは、デバイスのシステムシェルにアクセスするようユーザ（ネットワーク管理者）に依頼します。同意トークンは、システムシェルへの特権アクセス、制限アクセス、およびセキュアアクセスを提供する、ロック、ロック解除、および再ロックのメカニズムです。

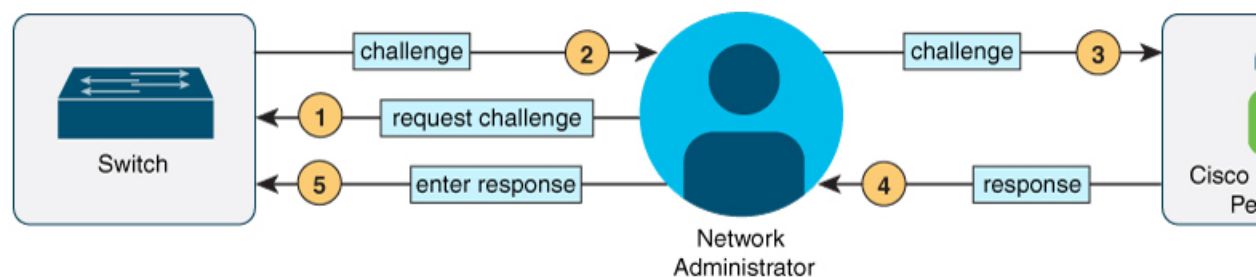
システムシェルへのアクセスを要求する場合は、認証を受ける必要があります。最初にコマンドを実行し、デバイスの同意トークン機能を使用してチャレンジを生成する必要があります。デバイスは、固有のチャレンジを出力として生成します。このチャレンジ文字列をコピーし、電子メールまたはインスタントメッセージでシスコ認定担当者に送信する必要があります。

シスコ認定担当者は、一意のチャレンジ文字列を処理し、一意のレスポンスを生成します。シスコ認定担当者はこのレスポンス文字列をコピーし、電子メールまたはインスタントメッセージで送信します。

次に、このレスポンス文字列をデバイスに入力する必要があります。チャレンジ/レスポンスペアが一致すると、システムシェルへのアクセスが許可されます。一致しない場合は、エラーが表示され、認証プロセスを繰り返す必要があります。

システムシェルにアクセスしたら、Cisco TAC エンジニアが必要とするデバッグ情報を収集します。システムシェルへのアクセスが完了したら、セッションを終了し、デバッグプロセスを続行します。

図 5: 同意トークン



システムシェルアクセスの同意トークン承認プロセス

ここでは、システムシェルにアクセスするための同意トークン承認のプロセスについて説明します。

手順

ステップ 1 指定された期間、システムシェルへのアクセスを要求するチャレンジを生成します。

例：

```
Device# request consent-token generate-challenge shell-access auth-timeout 900
% Consent token authorization success
Device#
*Jan 18 02:47:06.733: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (challenge generation
attempt: Shell access 0).
```

request consent-token generate-challenge shell-access time-validity-slot コマンドを使用して、チャレンジの要求を送信します。システムシェルへのアクセスを要求する期間（分単位）は、**time-slot-period** です。

この例の期間は、セッションの期限切れ後 900 分です。

デバイスは、固有のチャレンジを出力として生成します。このチャレンジは、base-64 形式の文字列です。

ステップ 2 シスコ認定担当者にチャレンジ文字列を送信します。

デバイスによって生成されたチャレンジ文字列を、電子メールまたはインスタントメッセージでシスコ認定担当者に送信します。

シスコ認定担当者は固有のチャレンジ文字列を処理し、レスポンスを生成します。レスポンスもまた、固有の base-64 文字列です。シスコ認定担当者はこのレスポンス文字列をコピーし、電子メールまたはインスタントメッセージで送信します。

ステップ 3 デバイスにレスポンス文字列を入力します。

例：

```
Device# request consent-token accept-response shell-access
% Consent token authorization success
Device#
*Jan 18 02:51:37.807: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (authentication success:
Shell access 0).
```

```
Device# request platform software system shell
Activity within this shell can jeopardize the functioning of the system.
Are you sure you want to continue? [y/n] y
Device#
*Jan 18 02:56:59.714: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (authorization for
Shell access 0 will expire in 10 min).
```

request consent-token accept-response shell-access response-string コマンドを使用して、シスコ認定担当者から送信されたレスポンス文字列を入力します。

チャレンジ/レスポンスペアが一致すると、システムシェルへのアクセスが許可されます。チャレンジ/レスポンスペアが一致しない場合は、エラーが表示され、手順 1 ~ 3 を繰り返す必要があります。

承認されると、要求されたタイムスロットのシステムシェルにアクセスできます。

承認セッションの残り時間が 10 分になると、デバイスはメッセージを送信します。

ステップ4 セッションを終了します。

例：

```
Device# request consent-token terminate-auth
% Consent token authorization termination success
```

```
Device#
*Jan 18 23:33:02.937: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (terminate
authentication: Shell access 0).
Device#
```

システムシェルへのアクセスが終了したら、**request consent-token terminate-auth** コマンドを使用してセッションを終了できます。このコマンドを使用して、承認タイムアウトの前にセッションを強制終了することもできます。要求したタイムスロットが期限切れになると、セッションも自動的に終了します。

同意トークンの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.11.1	同意トークン	同意トークンは、ネットワーク管理者と Cisco Technical Assistance Centre (Cisco TAC) の相互の同意により、システムシェルにアクセスする組織のネットワーク管理者を認証するために使用されるセキュリティ機能です。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。



第 18 章

ソフトウェア設定のトラブルシューティング

この章では、スイッチが稼働する Cisco IOS ソフトウェアに関連する問題を特定し、解決する方法について説明します。問題の性質に応じて、コマンドラインインターフェイス (CLI)、デバイス マネージャ、または Network Assistant を使用して、問題を特定し解決できます。

LEDの説明など、トラブルシューティングの詳細については、ハードウェアインストールガイドを参照してください。

- [ソフトウェア設定のトラブルシューティングに関する情報 \(311 ページ\)](#)
- [ソフトウェア設定のトラブルシューティング方法 \(319 ページ\)](#)
- [ソフトウェア設定のトラブルシューティングの確認 \(332 ページ\)](#)
- [ソフトウェア設定のトラブルシューティングのシナリオ \(334 ページ\)](#)
- [ソフトウェアのトラブルシューティングの設定例 \(339 ページ\)](#)
- [ソフトウェア設定のトラブルシューティングに関する追加情報 \(341 ページ\)](#)
- [ソフトウェア設定のトラブルシューティングの機能履歴 \(341 ページ\)](#)

ソフトウェア設定のトラブルシューティングに関する情報

スイッチのソフトウェア障害

スイッチソフトウェアがアップグレード中に破損する原因として、誤ったファイルがスイッチにダウンロードされた場合やイメージファイルが削除された場合があります。これらのどの場合も、接続はありません。ソフトウェア障害から回復するには、[ソフトウェア障害からの回復 \(319 ページ\)](#) の項で説明されている手順に従います。

デバイスのパスワードを紛失したか忘れた場合

デバイスのデフォルト設定では、デバイスを直接操作するエンドユーザが、スイッチの電源投入時に起動プロセスを中断して新しいパスワードを入力することにより、パスワードを紛失した状態から回復できます。ここで紹介する回復手順を実行するには、デバイスを直接操作してください。



- (注) これらのデバイスでは、システム管理者はデフォルト設定に戻す場合に限りエンドユーザによるパスワードのリセットを許可することによって、この機能の一部をディセーブルにできません。パスワード回復がディセーブルになっている場合に、エンドユーザがパスワードをリセットしようとする、ステータスメッセージで回復プロセスの間はデフォルトの設定に戻すように指示されます。



- (注) Cisco WLC の設定を複数の Cisco WLC 間でコピーすると、暗号化パスワード キーを回復できなくなります (RMA の場合)。

パスワードを紛失または忘れた場合にそのパスワードを回復するには、[パスワードを忘れた場合の回復 \(323 ページ\)](#) の項で説明する手順に従います。

ping

デバイスは IP の ping をサポートしており、これを使用してリモートホストへの接続をテストできます。ping はアドレスにエコー要求パケットを送信し、応答を待ちます。ping は次のいずれかの応答を返します。

- 正常な応答：正常な応答 (*hostname* が存在する) は、ネットワーク トラフィックにもよりますが、1 ~ 10 秒以内で発生します。
- 宛先の応答なし：ホストが応答しない場合、*no-answer* メッセージが返されます。
- 不明なホスト：ホストが存在しない場合、*unknown host* メッセージが返されます。
- 宛先到達不能：デフォルトゲートウェイが指定されたネットワークに到達できない場合、*destination-unreachable* メッセージが返されます。
- ネットワークまたはホストへの到達不能：ルートテーブルにホストまたはネットワークのエントリがない場合、*network or host unreachable* メッセージが返されます。

ping の動作を理解するには、[ping の実行 \(330 ページ\)](#) の項を参照してください。

レイヤ 2 トレースルート

レイヤ 2 トレースルート機能により、パケットが通過する送信元デバイスから宛先デバイスまでの物理パスを識別できます。レイヤ 2 トレースルートは、ユニキャストの送信元および宛先

MAC アドレスだけをサポートします。トレーズルートは、パス内にあるデバイスの MAC アドレステーブルを使用してパスを識別します。デバイスがパス内でレイヤ2トレーズルートをサポートしていないデバイスを検知した場合、デバイスはレイヤ2トレーズクエリを送信し続け、タイムアウトにします。

デバイスは、送信元デバイスから宛先デバイスへのパスのみを識別できます。パケットが通過する、送信元ホストから送信元デバイスまで、または宛先デバイスから宛先ホストまでのパスは識別できません。

レイヤ2の traceroute のガイドライン

- ネットワーク内のすべてのデバイスで、Cisco Discovery Protocol (CDP) をイネーブルにする必要があります。レイヤ2 traceroute が適切に動作するために、CDP を無効にしないでください。

物理パス内のデバイスが CDP に対して透過的な場合、スイッチはこれらのデバイスを通過するパスを識別できません。

- ping** 特権 EXEC コマンドを使用して接続をテストできれば、このデバイスは別のデバイスから到達可能であると定義できます。物理パス内のすべてのデバイスは、他のデバイスから相互に到達可能でなければなりません。
- パス内で識別可能な最大ホップ数は 10 です。
- 送信元デバイスと宛先デバイス間の物理パス内にないデバイスで、**traceroute mac** または **traceroute mac ip** の特権 EXEC コマンドを実行できます。パス内のすべてのデバイスは、このスイッチから到達可能でなければなりません。
- 指定された送信元および宛先アドレスが同じ VLAN にある場合、**traceroute mac** コマンド出力はレイヤ2パスを表示します。指定した送信元および宛先 MAC アドレスが、それぞれ異なる VLAN に属している場合は、レイヤ2パスは識別されず、エラーメッセージが表示されます。
- マルチキャストの送信元または宛先 MAC アドレスを指定すると、パスは識別されず、エラーメッセージが表示されます。
- 送信元または宛先 MAC アドレスが複数の VLAN に属する場合は、送信元および宛先 MAC アドレスの両方が属している VLAN を指定する必要があります。VLAN を指定しないと、パスは識別されず、エラーメッセージが表示されます。
- 指定された送信元および宛先の IP アドレスが同一のサブネット内にある場合、**traceroute mac ip** コマンド出力はレイヤ2パスを表示します。IP アドレスを指定した場合、デバイスは Address Resolution Protocol (ARP) を使用し、IP アドレスとそれに対応する MAC アドレスおよび VLAN ID を対応させます。
 - 指定の IP アドレスの ARP のエントリが存在している場合、デバイスは関連付けられた MAC アドレスを使用し、物理パスを識別します。
 - ARP のエントリが存在しない場合、デバイスは ARP クエリを送信し、IP アドレスを解決しようと試みます。IP アドレスが解決されない場合は、パスは識別されず、エラーメッセージが表示されます。

- 複数のデバイスがハブを介して1つのポートに接続されている場合（たとえば複数の CDP ネイバーがポートで検出された場合）、レイヤ 2 **traceroute** 機能はサポートされません。複数の CDP ネイバーが1つのポートで検出された場合、レイヤ 2 パスは特定されず、エラーメッセージが表示されます。
- この機能は、トークンリング VLAN ではサポートされません。
- レイヤ 2 トレースルートは、ユーザ データグラム プロトコル (UDP) ポート 2228 でリスニングソケットを開きます。このポートは、任意の IPv4 アドレスを使用してリモートからアクセスでき、認証は必要ありません。この UDP ソケットにより、VLAN 情報、リンク、特定の MAC アドレスの存在、および CDP ネイバー情報をデバイスから読み取ることができます。この情報を使用することにより、最終的にレイヤ 2 ネットワークトポロジの全体像を構築できます。
- レイヤ 2 トレースルートはデフォルトで有効になっており、グローバルコンフィギュレーションモードで **no l2 traceroute** コマンドを実行することによって無効にできます。レイヤ 2 トレースルートを再度有効にするには、グローバル コンフィギュレーション モードで **l2 traceroute** コマンドを使用します。

IP トレースルート

IP **traceroute** を使用すると、ネットワーク上でパケットが通過するパスをホップバイホップで識別できます。このコマンドを実行すると、トラフィックが宛先に到達するまでに通過するルータなどのすべてのネットワーク層 (レイヤ 3) デバイスが表示されます。

デバイスは、**traceroute** 特権 EXEC コマンドの送信元または宛先として指定できます。また、**traceroute** コマンドの出力でホップとして表示される場合があります。デバイスを **traceroute** の宛先とすると、スイッチは、**traceroute** の出力で最終の宛先として表示されます。中間デバイスが同じ VLAN 内でポート間のパケットのブリッジングだけを行う場合、**traceroute** の出力に中間スイッチは表示されません。ただし、中間デバイスが特定の packets をルーティングするマルチレイヤデバイスの場合、このデバイスは **traceroute** の出力にホップとして表示されます。

traceroute 特権 EXEC コマンドは、IP ヘッダーの存続可能時間 (TTL) フィールドを使用して、ルータおよびサーバで特定のリターンメッセージが生成されるようにします。**traceroute** の実行は、ユーザ データグラム プロトコル (UDP) データグラムを、TTL フィールドが 1 に設定されている宛先ホストへ送信することから始まります。ルータで TTL 値が 1 または 0 であることを検出すると、データグラムをドロップし、インターネット制御メッセージプロトコル (ICMP) **time-to-live-exceeded** メッセージを送信元に送信します。**traceroute** は、ICMP **time-to-live-exceeded** メッセージの送信元アドレス フィールドを調べて、最初のホップのアドレスを判別します。

ネクストホップを識別するために、**traceroute** は TTL 値が 2 の UDP パケットを送信します。1 番目のルータは、TTL フィールドの値から 1 を差し引いて次のルータにデータグラムを送信します。2 番目のルータは、TTL 値が 1 であることを確認すると、このデータグラムを廃棄し、**time-to-live-exceeded** メッセージを送信元へ返します。このように、データグラムが宛先ホストに到達するまで（または TTL の最大値に達するまで）TTL の値は増分され、処理が続けられます。

データグラムが宛先に到達したことを学習するために、**traceroute** は、データグラムの UDP 宛先ポート番号を、宛先ホストが使用する可能性のない大きな値に設定します。ホストが、ローカルで使用されない宛先ポート番号を持つ自分自身宛てのデータグラムを受信すると、送信元に ICMP ポート到達不能エラーを送信します。ポート到達不能エラーを除くすべてのエラーは中間ホップから送信されるため、ポート到達不能エラーを受信するということは、このメッセージが宛先ポートから送信されたことを意味します。

例：IP ホストに対する **traceroute** の実行 (340 ページ) に進み、IP **traceroute** プロセスの例を参照してください。

debug コマンド



注意 デバッグ出力は CPU プロセスで高プライオリティが割り当てられているため、デバッグ出力を行うとシステムが使用できなくなることがあります。したがって、**debug** コマンドを使用するのは、特定の問題のトラブルシューティング時、またはシスコのテクニカルサポート担当者とともにトラブルシューティングを行う場合に限定してください。ネットワークトラフィック量やユーザ数が少ない期間に **debug** コマンドを使用することをお勧めします。デバッグをこのような時間帯に行くと、**debug** コマンド処理のオーバーヘッドの増加によりシステムの使用に影響が及ぶ可能性が低くなります。

debug コマンドはすべて特権 EXEC モードで実行します。ほとんどの **debug** コマンドは引数を取りません。

あるスイッチスタックでデバッグをイネーブルにした場合は、アクティブスイッチでのみイネーブルになります。メンバスイッチのデバッグを有効にする場合は、**session switch-number** 特権 EXEC コマンドを使用して、アクティブスイッチからのセッションを開始する必要があります。次に、メンバスイッチのコマンドラインプロンプトで **debug** コマンドを入力します。

システム レポート

システム レポートまたは **crashinfo** ファイルには、シスコのテクニカルサポート担当者が Cisco IOS イメージの障害 (クラッシュ) が原因で起きた問題をデバッグするときに使用する情報が保存されています。明瞭度と整合性の高い重要なクラッシュ情報を迅速かつ確実に収集することが必要です。さらに、この情報の収集とバンドルが、特定のクラッシュの発生に対し関連付けか特定ができるような方法で行われることが必要です。

システム レポートは次の状況で生成されます。

- スイッチオーバーの場合：システム レポートはハイ アベイラビリティ (HA) のメンバー スイッチでのみ生成されます。非 HA メンバーについてはレポートは生成されません。

リロード時はレポートは生成されません。

クラッシュ プロセス時は、次の情報がスイッチからローカルに収集されます。

1. 完全なプロセス core
2. トレースログ
3. IOS の syslog (非アクティブなクラッシュの場合には保証されません)
4. システムプロセス情報
5. ブートアップログ
6. リロードログ
7. 特定のタイプの /proc 情報

この情報は個別のファイルに格納されてから、アーカイブされて1つのバンドルに圧縮されます。これにより、クラッシュのスナップショットを1つの場所で取得して、分析のためにボックス外に移動できるようになります。このレポートは、スイッチが ROMmon/ブートローダにダウンする前に生成されます。

完全な core およびトレースログ以外はテキスト ファイルです。

コアダンプを生成するには、**request platform software process core fed active** コマンドを使用します。

```
h2-macallan1# request platform software process core fed active
Process : fed main event (28155) encountered fatal signal 6
Process : fed main event stack :
```

```
SUCCESS: Core file generated.
```

```
h2-macallan1#dir bootflash:core
Directory of bootflash:/core/
```

```
178483  -rw-                1 May 23 2017 06:05:17 +00:00  .callhome
194710  drwx                   4096 Aug 16 2017 19:42:33 +00:00  modules
178494  -rw-                10829893 Aug 23 2017 09:46:23 +00:00
h2-macallan1_RP_0_fed_28155_20170823-094616-UTC.core.gz
```

crashinfo ファイル

デフォルトでは、生成されたシステム レポート ファイルは /crashinfo ディレクトリに格納されます。Ifit は、領域不足のため crashinfo パーティションに保存できません。そのため、/flash ディレクトリに保存されます。

ファイルを表示するには、**dir crashinfo:** コマンドを入力します。次に crashinfo ディレクトリの出力例を示します。

システムレポートは、次の形式で crashinfo ディレクトリにあります。

```
system-report_[switch number]_[date]-[timestamp]-UTC.gz
```

スイッチがクラッシュしたら、システムレポートファイルを確認します。最後に生成されたシステムレポートファイルは crashinfo ディレクトリの下に last_systemreport というファイル名で保存されます。問題のトラブルシューティングを行う際、システム レポートおよび crashinfo ファイルが TAC の役に立ちます。

生成されたシステム レポートは、TFTP や HTTP などいくつかのオプションを使用して、さらにコピーできます。

```
Switch#copy crashinfo: ?
crashinfo:      Copy to crashinfo: file system
flash:          Copy to flash: file system
ftp:            Copy to ftp: file system
http:           Copy to http: file system
https:          Copy to https: file system
null:           Copy to null: file system
nvram:          Copy to nvram: file system
rcp:            Copy to rcp: file system
running-config Update (merge with) current system configuration
scp:            Copy to scp: file system
startup-config Copy to startup configuration
syslog:         Copy to syslog: file system
system:         Copy to system: file system
tftp:           Copy to tftp: file system
tmpsys:        Copy to tmpsys: file system
```

TFTP サーバにコピーするための一般的な構文は次のとおりです。

```
Switch#copy crashinfo: tftp:
Source filename [system-report_1_20150909-092728-UTC.gz]?
Address or name of remote host []? 1.1.1.1
Destination filename [system-report_1_20150909-092728-UTC.gz]?
```

のトレースログは、**trace archive** コマンドを発行することで収集できます。このコマンドには、時間帯オプションがあります。コマンド構文は次のとおりです。

```
Switch#request platform software trace archive ?
last      Archive trace files of last x days
target    Location and name for the archive file
```

crashinfo: または **flash**: ディレクトリに格納されている過去 3650 日以内のトレースログが取得できます。

```
Switch# request platform software trace archive last ?
<1-3650> Number of days (1-3650)
Switch#request platform software trace archive last 3650 days target ?
crashinfo: Archive file name and location
flash:      Archive file name and location
```



(注) 一度コピーされたら、システム レポートやトレースのアーカイブを **flash** ディレクトリまたは **crashinfo** ディレクトリからクリアし、トレースログやその他の目的に使用できる領域を確保することが重要です。

スイッチのオンボード障害ロギング

オンボード障害ロギング (OBFL) 機能を使用すれば、デバイスに関する情報を収集できます。この情報には稼働時間、温度、電圧などの情報が含まれており、シスコのテクニカルサポート担当者がデバイスの問題をトラブルシューティングする際に役立ちます。OBFL はイネーブルにしておき、フラッシュメモリに保存されたデータは消さないようにすることを推奨します。

OBFLは、デフォルトでイネーブルになっています。デバイスおよび Small Form-Factor Pluggable (SFP) モジュールに関する情報が収集されます。デバイスは、次の情報をフラッシュメモリに保存します。

- CLI コマンド：スタンドアロンデバイスに入力された OBFL CLI コマンドの記録。
- メッセージ：スタンドアロンデバイスにより生成されたハードウェア関連のシステムメッセージの記録。
- Power over Ethernet (PoE)：スタンドアロンデバイスの PoE ポートの消費電力の記録。
- 温度：スタンドアロンデバイスの温度。
- 稼働時間：スタンドアロンデバイスが起動された際の時刻、デバイスが再起動された理由、およびデバイスが最後に再起動されて以来の稼働時間。
- 電圧：スタンドアロンデバイスのシステム電圧。

システム時計は、手動で時刻を設定するか、またはネットワーク タイム プロトコル (NTP) を使用するように設定します。

デバイスの稼働中には、**show logging onboard** 特権 EXEC コマンドを使用することにより、OBFL データを取得できます。デバイスに障害が発生した場合のデータの取得方法については、お客様担当のシスコテクニカルサポート担当者にお問い合わせください。

OBFL がイネーブルになっているデバイスが再起動された場合、新しいデータの記録が開始するまでに 10 分間の遅延があります。

ファン障害

デフォルトでは、この機能はディセーブルです。現場交換可能ユニット (FRU) または電源装置の複数のファンが故障した場合、デバイスはシャットダウンせず、次のようなエラーメッセージが表示されます。

デバイスが過熱状態となり、シャットダウンすることもあります。

デバイスを再起動するには、電源をオフにしてから再度オンにする必要があります。

CPU 使用率が高い場合に起こりうる症状

CPU 使用率が高すぎることで次の現象が発生する可能性があります。他の原因で発生する場合もあります。次にその一部を示します。

- スパニングツリー トポロジの変更
- 通信が切断されたために EtherChannel リンクがダウンした
- 管理要求 (ICMP ping、SNMP のタイムアウト、低速な Telnet または SSH セッション) に応答できない
- UDLD フラッピング

- SLA の応答が許容可能なしきい値を超えたことによる IP SLA の失敗
- スイッチが要求を転送しない、または要求に応答しない場合の DHCP または IEEE 802.1x の処理の失敗

ソフトウェア設定のトラブルシューティング方法

パスワードを忘れた場合の回復

スイッチのデフォルト設定では、スイッチを直接操作するエンドユーザが、スイッチの電源投入時に起動プロセスを中断して新しいパスワードを入力することにより、パスワードを紛失した状態から回復できます。ここで紹介する回復手順を実行するには、スイッチを直接操作してください。



- (注) これらのスイッチでは、システム管理者はデフォルト設定に戻す場合に限りエンドユーザによるパスワードのリセットを許可することによって、この機能の一部をディセーブルにできます。パスワード回復がディセーブルになっている場合に、エンドユーザがパスワードをリセットしようとする、回復プロセスの間、ステータスメッセージにその旨が表示されます。

手順

ステップ 1 端末または PC をスイッチに接続します。

- 端末または端末エミュレーションソフトウェアが稼働している PC をスイッチのコンソールポートに接続します。スイッチスタックのパスワードを回復する場合は、アクティブスイッチのコンソールポートに接続します。
- PC をイーサネット管理ポートに接続します。スイッチスタックのパスワードを回復する場合は、スタックメンバのイーサネット管理ポートに接続します。

ステップ 2 エミュレーションソフトウェアの回線速度を 9600 ボーに設定します。

ステップ 3 スタンドアロンスイッチまたはスイッチスタック全体の電源を切断します。

ステップ 4 スイッチまたはアクティブスイッチに電源コードを再接続します。システム LED が点滅したら、すぐに [Mode] ボタンを 2〜3 回押して放します。スイッチは ROMMON モードを開始します。

リロード中に次のコンソールメッセージが表示されます。

```
Initializing Hardware...
```

パスワード回復がイネーブルになっている場合の手順

```
System Bootstrap, Version xx.x.1r [FC1], RELEASE SOFTWARE (P)
Compiled Tue 09/29/2020 18:05:06 by rel
```

```
Current ROMMON image : Primary
C9200-24P platform with 4194304 Kbytes of main memory
```

```
Preparing to autoboot. [Press Ctrl-C to interrupt] 4 (interrupted) <----- break
sequence to be pressed
```

「パスワード回復がイネーブルになっている場合の手順」セクションに記載されている手順を実行します。

ステップ 5 パスワードの回復後、スイッチまたはアクティブスイッチをリロードします。

スイッチの場合

```
Switch> reload
Proceed with reload? [confirm] y
```

アクティブ スイッチの場合

```
Switch> reload slot <stack-active-member-number>
Proceed with reload? [confirm] y
```

ステップ 6 スタック内の残りのスイッチに電源を投入します。

パスワード回復がイネーブルになっている場合の手順

手順

ステップ 1 次の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を無視します。

```
Device: SWITCH_IGNORE_STARTUP_CFG=1
```

ステップ 2 *packages.conf* ファイルでスイッチをフラッシュからブートします。

```
Device: boot flash:packages.conf
```

ステップ 3 **No** と応答して初期設定ダイアログを終了します。

```
Would you like to enter the initial configuration dialog? [yes/no]: No
```

ステップ 4 スイッチ プロンプトで、特権 EXEC モードを開始します。

```
Device> enable
Device#
```

ステップ 5 スタートアップ コンフィギュレーションを実行コンフィギュレーションにコピーします。


```
Device# copy startup-config running-config Destination filename [running-config]?
```

確認を求めるプロンプトに、Return を押して応答します。これで、コンフィギュレーションファイルがリロードされ、パスワードを変更できます。

ステップ 6 グローバルコンフィギュレーションモードを開始して、イネーブルパスワードを変更します。

```
Device# configure terminal
Device(config)# enable secret password
```

ステップ 7 特権 EXEC モードに戻ります。

```
Device(config)# exit
Device#
```

ステップ 8 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ファイルに書き込みます。

```
Device# copy running-config startup-config
```

ステップ 9 手動ブートモードがイネーブルになっていることを確認します。

```
Device# show boot

BOOT variable = flash:packages.conf;
Manual Boot = yes
Enable Break = yes
```

ステップ 10 デバイスのリロード。

```
Device# reload
```

ステップ 11 ブートローダーパラメータを元の値に戻します。

```
Device: SWITCH_IGNORE_STARTUP_CFG=0
```

ステップ 12 フラッシュの *packages.conf* ファイルを使用して、デバイスを起動します。

```
Device: boot flash:packages.conf
```

ステップ 13 デバイスが起動したら、デバイスで手動ブートを無効にします。

```
Device(config)# no boot manual
```

パスワード回復がディセーブルになっている場合の手順

パスワード回復メカニズムがディセーブルの場合、次のメッセージが表示されます。

```
The password-recovery mechanism has been triggered, but
is currently disabled. Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point. However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.
```

```
Would you like to reset the system back to the default configuration (y/n)?
```



注意 デバイスをデフォルト設定に戻すと、既存の設定がすべて失われます。システム管理者に問い合わせ、バックアップデバイスと VLAN（仮想 LAN）コンフィギュレーションファイルがあるかどうかを確認してください。

- **n** (no) を入力すると、Mode ボタンを押さなかった場合と同様に、通常のブートプロセスが継続されます。ブートローダプロンプトにはアクセスできません。したがって、新しいパスワードを入力できません。次のメッセージが表示されます。

```
Press Enter to continue.....
```

- **y** (yes) を入力すると、フラッシュメモリ内のコンフィギュレーションファイルおよび VLAN データベースファイルが削除されます。デフォルト設定がロードされるときに、パスワードをリセットできます。

手順

ステップ 1 パスワード回復手順の継続を選択すると、既存の設定が失われます。

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

ステップ 2 フラッシュメモリの内容を表示します。

```
Device: dir flash:
```

デバイスのファイルシステムが表示されます。

ステップ 3 システムを起動します。

```
Device: boot
```

セットアッププログラムを起動するように求められます。パスワード回復手順を継続するには、プロンプトに **N** を入力します。

```
Continue with the configuration dialog? [yes/no]: N
```

ステップ 4 デバイスプロンプトで、特権 EXEC モードを開始します。

```
Device> enable
```

ステップ 5 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。

```
Device# configure terminal
```

ステップ 6 パスワードを変更します。

```
Device(config)# enable secret password
```

シークレット パスワードは 1 ～ 25 文字の英数字です。数字で始めることができます。大文字と小文字が区別され、スペースを使用できますが、先行スペースは無視されます。

ステップ 7 特権 EXEC モードに戻ります。

```
Device(config)# exit  
Device#
```

(注) ステップ 9 に進む前に、接続されているすべてのスタック メンバの電源を入れ、それらが完全に初期化されるまで待ちます。

ステップ 8 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 ファイルに書き込みます。

```
Device# copy running-config startup-config
```

新しいパスワードがスタートアップ コンフィギュレーションに組み込まれました。

ステップ 9 ここで、デバイスを再設定する必要があります。システム管理者によって、バックアップデバイスと VLAN コンフィギュレーション ファイルが使用可能に設定されている場合は、これらを使用します。

スイッチ スタック問題の回避

スイッチ スタックの問題を防止するには、次の作業を実行する必要があります。

- スイッチスタックにデバイスを追加したり、そこから取り外したりする場合には、必ずその電源を切ってください。スイッチ スタックでの電源関連のあらゆる考慮事項については、ハードウェア インストール ガイドの「Switch Installation (スイッチのインストール)」の章を参照してください。
- スタックモード LED が点灯するまで、スタックメンバの **Mode** ボタンを押します。デバイスの最後の 2 つのポート LED がグリーンになります。デバイスモデルに応じて、最後

の2つのポートは10/100/1000ポートまたは Small Form-Factor Pluggable (SFP) モジュールになります。最後の2つのポート LED の片方または両方がグリーンになっていない場合は、スタックが全帯域幅で稼働していません。

- スイッチスタックを管理する場合は、1つのCLIセッションだけを使用することを推奨します。アクティブスイッチへの複数のCLIセッションを使用する場合は注意が必要です。1つのセッションで入力したコマンドは、別のセッションには表示されません。そのため、コマンドを入力したセッションを識別できなくなることがあります。
- スタック内での位置に従ってスタックメンバ番号を手動で割り当てると、リモートから行うデバイススタックのトラブルシューティングが容易になります。ただし、後からデバイスを追加したり、取り外したり、場所を入れ替えたりする際に、デバイスを手動で番号を割り当てたことを覚えておく必要があります。スタックメンバ番号を手動で割り当てするには、**switch current-stack-member-number renumber new-stack-member-number** グローバル コンフィギュレーション コマンドを使用します。

スタックメンバをまったく同じモデルで置き換えると、新しいデバイスは、置き換えられたデバイスとまったく同じ設定で稼働します。この場合、新しいデバイスは置き換えられたデバイスと同じメンバ番号を使用するものと想定されます。

電源が入った状態のスタックメンバを取り外すと、スイッチスタックが、それぞれ同じ設定を持つ2つ以上のスイッチスタックに分割（パーティション化）されます。スイッチスタックを分離されたままにしておきたい場合は、新しく作成されたスイッチスタックのIPアドレス（複数の場合あり）を変更してください。パーティション化されたスイッチスタックを元に戻すには、次の手順を実行します。

1. 新しく作成されたスイッチスタックの電源を切ります。
2. 新しいスイッチスタックを、StackWise Plus ポートを介して元のスイッチスタックに再度接続します。
3. デバイスの電源を入れます。

スイッチスタックおよびそのメンバのモニタリングに使用できるコマンドについては、「*Displaying Switch Stack Information*」の項を参照してください。

自動ネゴシエーションの不一致の防止

IEEE 802.3ab 自動ネゴシエーションプロトコルは速度（10 Mbps、100 Mbps、および SFP モジュールポート以外の 1000 Mbps）およびデュプレックス（半二重または全二重）に関するデバイスの設定を管理します。このプロトコルは設定を適切に調整しないことがあり、その場合はパフォーマンスが低下します。不一致は次の条件で発生します。

- 手動で設定した速度またはデュプレックスのパラメータが、接続ポート上で手動で設定された速度またはデュプレックスのパラメータと異なっている場合。
- ポートを自動ネゴシエーションに設定したが、接続先ポートは自動ネゴシエーションを使用しない全二重に設定されている場合。

デバイスのパフォーマンスを最大限に引き出してリンクを確保するには、次のいずれかの注意事項に従って、デュプレックスおよび速度の設定を変更してください。

- 速度とデュプレックスの両方について、両方のポートで自動ネゴシエーションを実行させます。
- 接続の両側でポートの速度とデュプレックスのパラメータを手動で設定します。



- (注) 接続先装置が自動ネゴシエーションを実行しない場合は、2つのポートのデュプレックス設定を一致させます。速度パラメータは、接続先のポートが自動ネゴシエーションを実行しない場合でも自動調整が可能です。

SFP モジュールのセキュリティと識別に関するトラブルシューティング

シスコの Small Form-Factor Pluggable (SFP) モジュールは、モジュールのシリアル番号、ベンダー名とベンダー ID、一意のセキュリティ コード、および巡回冗長検査 (CRC) が格納されたシリアル EEPROM (電氣的に消去可能でプログラミング可能な ROM) を備えています。デバイスに SFP モジュールを装着すると、デバイスソフトウェアは、EEPROM を読み取ってシリアル番号、ベンダー名、およびベンダー ID を確認し、セキュリティコードと CRC を再計算します。シリアル番号、ベンダー名、ベンダー ID、セキュリティコード、または CRC が無効な場合、ソフトウェアは、セキュリティ エラー メッセージを生成し、インターフェイスを `errdisable` ステートにします。



- (注) セキュリティ エラー メッセージは、`GBIC_SECURITY` 機能を参照します。スイッチは、SFP モジュールをサポートしていますが、GBIC (ギガビット インターフェイス コンバータ) モジュールはサポートしていません。エラーメッセージテキストは、GBIC インターフェイスおよびモジュールを参照しますが、セキュリティ メッセージは、実際は SFP モジュールおよびモジュール インターフェイスを参照します。

他社の SFP モジュールを使用している場合、デバイスから SFP モジュールを取り外し、シスコのモジュールに交換します。シスコの SFP モジュールを装着したら、`errdisable recovery cause gbic-invalid` グローバル コンフィギュレーション コマンドを使用してポートのステータスを確認し、`error-disabled` 状態から回復する時間間隔を入力します。この時間間隔が経過すると、デバイスは `error-disabled` 状態からインターフェイスを回復させ、操作を再試行します。`errdisable recovery` コマンドの詳細については、このリリースに対応するコマンドリファレンスを参照してください。

モジュールがシスコ製 SFP モジュールとして識別されたにもかかわらず、システムがベンダーデータ情報を読み取ってその情報が正確かどうかを確認できないと、SFP モジュール エラーメッセージが生成されます。この場合、SFP モジュールを取り外して再び装着してください。それでも障害が発生する場合は、SFP モジュールが不良品である可能性があります。

ping の実行

別の IP サブネットワーク内のホストに ping を実行する場合は、ネットワークへのスタティックルートを定義するか、またはこれらのサブネット間でルーティングされるように IP ルーティングを設定する必要があります。

IP ルーティングは、デフォルトではすべてのデバイスでディセーブルになります。



(注) ping コマンドでは、他のプロトコルキーワードも使用可能ですが、このリリースではサポートされていません。

このコマンドは、デバイスからネットワーク上の他のデバイスに ping を実行する目的で使用します。

コマンド	目的
<p>ping ip <i>host</i> <i>address</i></p> <p>Device# ping 172.20.52.3</p>	<p>IP またはホスト名やネットワーク アドレスを指定してリモートホストに ping を実行します。</p>

温度のモニタリング

デバイスは温度条件をモニタし、温度情報を使用してファンを制御します。

物理パスのモニタリング

次のいずれかの特権 EXEC コマンドを使用して、パケットが通過する、送信元デバイスから宛先デバイスへの物理パスをモニタできます。

表 13: 物理パスのモニタリング

コマンド	目的
<p>tracetroute mac [interface <i>interface-id</i>] { <i>source-mac-address</i> } [interface <i>interface-id</i>] { <i>destination-mac-address</i> } [vlan <i>vlan-id</i>] [detail]</p>	<p>指定の送信元 MAC アドレスから、指定の宛先 MAC アドレスまでをパケットが通過するレイヤ 2 パスを表示します。</p>
<p>tracetroute mac ip { <i>source-ip-address</i> <i>source-hostname</i> } { <i>destination-ip-address</i> <i>destination-hostname</i> } [detail]</p>	<p>指定の送信元 IP アドレスまたはホスト名から、指定の宛先 IP アドレスまたはホスト名を通過するパケットのレイヤ 2 パスを表示します。</p>

IP traceroute の実行



- (注) **traceroute** 特権 EXEC コマンドでは、他のプロトコルキーワードも使用可能ですが、このリリースではサポートされていません。

コマンド	目的
traceroute ip host Device# traceroute ip 192.51.100.1	ネットワーク上でパケットが通過するパスを追跡します。

デバッグおよびエラーメッセージ出力のリダイレクト

デフォルトでは、ネットワークサーバが **debug** コマンドからの出力とシステムエラーメッセージをコンソールに送信します。このデフォルトの設定を使用する場合は、コンソールポートに接続する代わりに、仮想端末接続によってデバッグ出力をモニタできます。

指定できる宛先として、コンソール、仮想端末、内部バッファ、およびsyslogサーバを実行している UNIX ホストがあります。Syslog フォーマットは、4.3 BSD UNIX およびそのバリエーションと互換性があります。



- (注) デバッグの出力先がシステムのオーバーヘッドに影響を与えないように注意してください。メッセージをコンソールに記録すると、非常に高いオーバーヘッドが発生します。仮想端末にメッセージを記録すると、発生するオーバーヘッドは低くなります。Syslog サーバでメッセージロギングを行うと、オーバーヘッドはさらに小さくなり、内部バッファであれば最小限ですみます。

システムメッセージのロギングに関する詳細については、「システムメッセージロギングの設定」を参照してください。

show platform コマンドの使用

show platform 特権 EXEC コマンドの出力からは、インターフェイスに着信するパケットがシステムを介して送信された場合の転送結果に関する有意義な情報がいくつか得られます。パケットに関して入力されたパラメータに応じて、参照テーブル結果、転送宛先の計算に使用されるポートマップ、ビットマップ、および出力側の情報が表示されます。

このコマンドで出力される情報のほとんどは、主に、デバイスの特定用途向け集積回路 (ASIC) に関する詳細情報を使用するテクニカルサポート担当者に役立つものです。ただし、パケット転送情報はトラブルシューティングにも役立ちます。

show debug コマンドの使用法

show debug コマンドは特権 EXEC モードで入力します。このコマンドは、スイッチで使用可能なすべてのデバッグ オプションを表示します。

すべての条件付きデバッグオプションを表示するには、コマンド **show debug condition** を実行します。コマンドは、条件 ID <1-1000> または *all* 条件を選択することで一覧表示できます。

デバッグを無効にするには、**no debug all** コマンドを使用します。



注意 デバッグ出力は CPU プロセスで高プライオリティが割り当てられているため、デバッグ出力を行うとシステムが使用できなくなることがあります。したがって、**debug** コマンドを使用するのは、特定の問題のトラブルシューティング時、またはシスコのテクニカルサポート担当者とともにトラブルシューティングを行う場合に限定してください。さらに、**debug** コマンドは、ネットワークトラフィックが少なく、ユーザも少ないときに使用することを推奨します。デバッグをこのような時間帯に行うと、**debug** コマンド処理のオーバーヘッドの増加によりシステムの使用に影響が及ぶ可能性が低くなります。

ソフトウェア設定のトラブルシューティングの確認

OBFL 情報の表示

例：高い CPU 使用率に関する問題と原因の確認

CPU 使用率が高いことが問題となっているかどうか判断するには、**show processes cpu sorted** 特権 EXEC コマンドを入力します。出力例の 1 行目にある下線が付いた部分に注目してください。

```
Device# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>
```

この例は、正常な CPU 使用率を示しています。この出力によると、最後の 5 秒間の使用率が 8%/0% となっていますが、この意味は次のとおりです。

- Cisco IOS の処理時間と割り込みの処理にかかった時間を合わせた CPU の合計の使用率は全体の 8%

- 割り込みの処理にかかった時間は全体の 0%

表 14: CPU 使用率に関する問題のトラブルシューティング

問題のタイプ	原因	修正措置
割り込みのパーセント値が合計の CPU 使用率の値とほぼ同程度に高い	CPU がネットワークから受信するパケット数が多すぎる。	ネットワーク パケットのソースを判別する。データの流れを遮断するか、スイッチの設定を変更します。 「Analyzing Network Traffic (ネットワーク トラフィックの解析)」の項を参照してください。
割り込みの所要時間は最小限であったにもかかわらず CPU の合計使用率が 50% を超える	CPU 時間を過度に消費する Cisco IOS 処理が 1 つ以上存在する。これは通常、処理をアクティブ化するイベントによって始動されます。	異常なイベントを特定して根本的な原因を解消する。「Debugging Active Processes (アクティブなプロセスのデバッグ)」のセクションを参照してください。

ソフトウェア設定のトラブルシューティングのシナリオ

Power over Ethernet (PoE) に関するトラブルシューティングのシナリオ

表 15: Power over Ethernet に関するトラブルシューティングのシナリオ

症状または問題	考えられる原因と解決法
PoE がないポートは1つに限りません。 1つのスイッチポートに限り問題が発生する。このポートではPoE装置と PoE 非対応の装置のいずれも動作しないが、他のポートでは動作します。	

症状または問題	考えられる原因と解決法
	<p>この受電デバイスが他の PoE ポートで動作するかを確認する。</p> <p>show run または show interface status ユーザ EXEC コマンドを使用して、ポートがシャットダウンしていないか、または error-disabled になっていないかを確認します。</p> <p>(注) ほとんどのスイッチはポートがシャットダウンしているときはポートの電力供給をオフにします。これは、IEEE 仕様でこれがオプションに指定されている場合も同様です。</p> <p>該当するインターフェイスまたはポートに power inline never が設定されていないことを確認します。</p> <p>受電デバイスからスイッチポートまでのイーサネットケーブルの動作が正常であることを確認します。具体的には、既知の正常な PoE 非対応のイーサネット装置とイーサネットケーブルを接続して、受電デバイスがリンクを確立し他のホストとトラフィックを交換することを確認します。</p> <p>(注) シスコ受電装置は、ストレートケーブルでのみ機能します。クロスオーバーケーブルでは機能しません。</p> <p>スイッチのフロントパネルから受電デバイスまでのケーブル長の合計が 100 メートル以下であることを確認します。</p> <p>スイッチポートからイーサネットケーブルを外します。短いイーサネットケーブルを使用して、既知の正常なイーサネット装置を、スイッチのフロントパネルの（パッチパネルではない）このポートに直接接続します。これによってイーサネットリンクが確立され他のホストとトラフィックを交換できることを確認します。あるいは、ポートの VLAN SVI で ping を実行してください。次に、受電デバイスをこのポートに接続し、電源がオンになることを確認します。</p> <p>パッチコードをスイッチポートに接続しても受電デバイスの電源がオンにならない場合、接続する受電デバイスの合計数とスイッチの電力バジェット（使用可能な PoE）とを比較してください。 show power inline コマンドを使用して、利用可能な電力量を確認します。</p>

症状または問題	考えられる原因と解決法
<p>すべてのポートまたは1つのポートグループで PoE が機能しない。</p> <p>すべてのスイッチポートで問題が発生する。電力が供給されていないイーサネット装置がどのポートでもイーサネットリンクを確立できず、PoE装置の電源がオンになりません。</p>	

症状または問題	考えられる原因と解決法
	<p>電力に関するアラームが継続的に発生する、断続的に発生する、または再発する場合は、可能であれば電源モジュールを交換します（現場交換可能ユニットです）。そうでない場合はスイッチを交換してください。</p> <p>連続する複数のポートで問題があるものの、すべてのポートで問題が発生するわけではない場合、電源の故障ではないと考えられ、スイッチのPoEレギュレータに関連した異常の可能性がります。</p> <p>PoE の状況やステータスの変更について過去に報告されているアラームまたはシステムメッセージがないか、show log 特権 EXEC コマンドを使用して調べます。</p> <p>アラームがない場合は、show interface status コマンドを使用して、ポートがシャットダウンしていないか、または error-disabled になっていないかを確認します。ポートが error-disabled の場合、shut および no shut インターフェイス コンフィギュレーション コマンドを使用して、ポートを再度有効にします。</p> <p>show env power および show power inline 特権 EXEC コマンドを使用して、PoE のステータスおよび電力バジェット（使用可能な PoE）を調べます。</p> <p>実行コンフィギュレーションを調べて、power inline never がこのポートに設定されていないことを確認します。</p> <p>受電していないイーサネット装置をスイッチポートに直接接続します。接続には短いパッチコードだけを使用します。既存の配線ケーブルは使用しないでください。shut および no shut インターフェイス コンフィギュレーション コマンドを入力し、イーサネットリンクが確立されていることを確認します。正しく接続している場合、短いパッチコードを使用して受電デバイスをこのポートに接続し、電源がオンになることを確認します。装置の電源がオンになったら、すべての中間パッチパネルが正しく接続されているか確認してください。</p> <p>1本を除くすべてのイーサネットケーブルをスイッチポートから抜きます。短いパッチコードを使用して、1つのPoEポートにだけ受電デバイスを接続します。スイッチポートからの受電に比較して、受電デバイスが多くの電力を必要としないことを確認してください。</p> <p>show power inline 特権 EXEC コマンドを使用して、ポートがシャットダウンされていない場合に、受電デバイスに電力が供給されることを確認します。あるいは、受電デバイ</p>

症状または問題	考えられる原因と解決法
	<p>スを観察して電源がオンになることを確認してください。</p> <p>1 台の受電デバイスだけがスイッチに接続している際に電力が供給される場合、残りのポートで shut および no shut インターフェイス コンフィギュレーション コマンドを入力してから、イーサネットケーブルをスイッチの PoE ポートに 1 本ずつ再接続してください。 show interface status および show power inline 特権 EXEC コマンドを使用して、インラインパワーの統計情報とポートのステータスをモニタします。</p> <p>すべてのポートで、まだ PoE が機能しない場合は、電源装置の PoE セクションでヒューズを開くことができます。この場合、アラームが生成されるのが一般的です。過去にシステムメッセージでアラームが報告されていないか、ログをもう一度チェックしてください。</p>
<p>シスコ先行標準受電装置は、切断またはリセットされます。</p> <p>正常に動作した後で、シスコ電話機が断続的にリロードしたり、PoE から切断されたりします。</p>	<p>スイッチから受電デバイスまでのすべての電気システムを確認してください。信頼性の低い接続は、電力供給の中断や受電デバイスの機能が不安定になる原因となり、受電デバイスの断続的な切断やリロードが発生します。</p> <p>スイッチ ポートから受電デバイスまでのケーブル長が 100 メートル以下であることを確認してください。</p> <p>スイッチが配置されている場所で電気環境にどのような変化があるか、切断時に、受電デバイスに何が起きるかについて注意してください。</p> <p>切断と同時にエラー メッセージが表示されたか注意します。 show log 特権 EXEC コマンドを使用して、エラーメッセージを確認します。</p> <p>リロードの発生直前に IP Phone から Call Manager へのアクセスが失われていないか確認してください (PoE の障害ではなくネットワークに問題が発生している場合があります)。</p> <p>受電デバイスを PoE 非対応の装置に交換し、装置が正しく動作することを確認します。PoE 非対応の装置にリンク障害または高いエラー率がある場合、スイッチポートと受電デバイスを接続する信頼性の低いケーブル接続が問題の可能性もあります。</p>

症状または問題	考えられる原因と解決法
<p>IEEE 802.3af 準拠または IEEE 802.3at 準拠の受電装置は、Cisco PoE スイッチでは機能しません。</p> <p>シスコ PoE スイッチに接続するシスコ以外の受電デバイスに電源が供給されないか、電源投入後すぐに電源が切れます。PoE 非対応装置は正常に動作します。</p>	<p>show power inline コマンドを使用して、受電デバイスの接続前後に、スイッチの電力バジェット（使用可能な PoE）が枯渇していないか確認します。受電デバイスを接続する前に、このタイプの装置に十分な電力が使用可能であることを確認します。</p> <p>show interface status コマンドを使用して、接続されている受電デバイスがスイッチに検出されることを確認します。</p> <p>show log コマンドを使用して、ポートの過電流状態を報告したシステムメッセージがないか確認します。症状を正確に特定してください。最初に電力が受電デバイスに供給され、その後、切断される状態ですか。その場合は、問題は最初のサージ電流（突入電流）が原因で、ポートの電流上限しきい値が超過した可能性があります。</p>

ソフトウェアのトラブルシューティングの設定例

例：IP ホストの ping

次に、IP ホストに ping を実行する例を示します。

```
Device# ping 172.20.52.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Device#
```

表 16: ping の出力表示文字

文字	説明
!	感嘆符 1 個につき 1 回の応答を受信したことを示します。
.	ピリオド 1 個につき応答待ちの間にネットワーク サーバのタイムアウトが 1 回発生したことを示します。
U	宛先到達不能エラー PDU を受信したことを示します。
C	輻輳に遭遇したパケットを受信したことを示します。
I	ユーザによりテストが中断されたことを示します。

例：IP ホストに対する **traceroute** の実行

文字	説明
?	パケット タイプが不明です。
&	パケットの存続時間を超過したことを示します。

ping セッションを終了するには、エスケープシーケンス（デフォルトでは **Ctrl+^X**）を入力してください。Ctrl キー、Shift キー、および 6 キーを同時に押してから放し、その後 X キーを押します。

例：IP ホストに対する **traceroute** の実行

次に、IP ホストに **traceroute** を実行する例を示します。

```
Device# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

 1 192.0.2.1 0 msec 0 msec 4 msec
 2 192.0.2.203 12 msec 8 msec 0 msec
 3 192.0.2.100 4 msec 0 msec 0 msec
 4 192.0.2.10 0 msec 4 msec 0 msec
```

ディスプレイには、送信される 3 つのプロープごとに、ホップ カウント、ルータの IP アドレス、およびラウンドトリップ タイム（ミリ秒単位）が表示されます。

表 17: **traceroute** の出力表示文字

文字	説明
*	プローブがタイムアウトになりました。
?	パケット タイプが不明です。
A	管理上、到達不能です。通常、この出力は、アクセスリストがトラフィックをブロックしていることを表しています。
H	ホストが到達不能です。
N	ネットワークが到達不能です。
P	プロトコルが到達不能です。
Q	発信元。
U	ポートが到達不能です。

実行中の追跡を終了するには、エスケープシーケンス（デフォルトではCtrl+^X）を入力してください。Ctrl キー、Shift キー、および6 キーを同時に押してから放し、その後 X キーを押します。

ソフトウェア設定のトラブルシューティングに関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9200 Series Switches)</i>

ソフトウェア設定のトラブルシューティングの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	ソフトウェア設定のトラブルシューティング	ソフトウェア設定のトラブルシューティングでは、スイッチが稼働する Cisco IOS ソフトウェアに関連する問題を特定し、解決する方法について説明します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。