# Cisco IOS XE Amsterdam 17.3.x（Catalyst 9400 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**：2020 年 7 月 31 日

**最終更新**：2020 年 11 月 7 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 NONINFRINGEMENT 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. 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)

© 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
スタティック アドレス エントリの追加および削除 26
ユニキャスト MAC アドレス フィルタリングの設定 28
デバイスのモニタリングおよび保守の管理 29
デバイス管理の設定例 30
例：システム クロックの設定 30
例：サマー タイムの設定 30
例：MOTD バナーの設定 31
例：ログイン バナーの設定 31
例：MAC アドレス変更通知トラップの設定 31
例：MAC しきい値通知トラップの設定 32
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 32
例：ユニキャスト MAC アドレス フィルタリングの設定 32
デバイス管理に関する追加情報 33
デバイス管理の機能履歴と情報 33

第 2 章 デバイスのセットアップ設定の実行 35
デバイスセットアップの設定の制約事項 35
デバイスセットアップ設定の実行に関する情報 35
デバイスブートプロセス 35

ソフトウェア インストールの概要 37
ソフトウェアのブート モード 37
ソフトウェア パッケージのインストール 38
ソフトウェアインストールの終了 39
デバイス情報の割り当て 39
デフォルトのスイッチ情報 40
DHCP ベースの自動設定の概要 40
DHCP クライアントの要求プロセス 41
DHCP ベースの自動設定およびイメージ アップデート 42
DHCP ベースの自動設定の制約事項 42
DHCP 自動設定 43
DHCP 自動イメージ アップデート 43
DHCP サーバ設定時の注意事項 44
TFTP サーバの目的 44
DNS サーバの目的 45
コンフィギュレーション ファイルの入手方法 46
環境変数の制御方法 47
一般的な環境変数 48
TFTP の環境変数 50
ソフトウェア イメージのリロードのスケジューリング 50
デバイスセットアップ設定の実行方法 51
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 51
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 53
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 56
複数の SVI への IP 情報の手動割り当て 57
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 59
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 59
スイッチの手動による起動 61
インストール モードでのデバイスのブート 62
バンドルモードでのデバイスの起動 64

ソフトウェア イメージのリロードのスケジュール設定 65
デバイスのセットアップの設定例 66
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 66
例 : 更新プログラム パッケージの管理 70
ソフトウェア インストールの確認 82
例 : デバイスを DHCP サーバとして設定 85
例 : DHCP 自動イメージ アップデートの設定 85
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 86
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 86
デバイスセットアップの実行に関する追加情報 87
デバイスセットアップ設定の実行に関する機能情報 87

第 3 章 スマート ライセンスの設定 89
スマートライセンシングの設定の前提条件 89
スマートライセンシングの概要 89
CSSM の概要 90
CSSM への接続 91
CSSM への既存のライセンスのリンク 92
CSSM への接続の設定とライセンスレベルの設定 92
CSSM への接続の設定 93
ダイレクトクラウドアクセス用の Call Home サービスの設定 95
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 97
Cisco Smart Software Manager オンプレミス用の Call Home サービスの設定 99
ライセンスレベルの設定 102
CSSM でのデバイスの登録 103
CSSM からの新しいトークンの生成 104
新しいトークンを使用するデバイスの登録 106
登録後のライセンスステータスの確認 106
CSSM でのデバイスの登録キャンセル 108
スマート ライセンスの設定のモニタリング 109

スマートライセンシングの設定例 110
例：Call Home プロファイルの表示 110
例：登録前のライセンス情報の表示 111
例：デバイスの登録 113
例：登録後のライセンスステータスの表示 113
その他の参考資料 116
スマートライセンスの機能の履歴 117

第 4 章 有線ネットワークでの Application Visibility and Control の設定 119
有線ネットワークでの Application Visibility and Control について 119
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 120
有線 Application Visibility and Control の制限 121
Application Visibility and Control の設定方法 123
有線ネットワークでの Application Visibility and Control の設定 123
インターフェイスでのアプリケーション認識の有効化 124
AVC QoS ポリシーの作成 125
スイッチ ポートへの QoS ポリシーの適用 127
有線 AVC Flexible Netflow の設定 128
NBAR2 カスタム アプリケーション 146
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 149
Application Visibility and Control のモニタリング 151
例：Application Visibility and Control の設定 152
基本的なトラブルシューティング：質問と回答 163
Application Visibility and Control に関する追加情報 165
有線ネットワークでの Application Visibility and Control の機能履歴と情報 165

第 5 章 環境モニタリングおよび電源管理 167
環境モニタリングについて 167
CLI コマンドによる環境のモニタリング 167
環境状態の表示 168
オンボード障害ロギング（OBFL）情報の表示 169

緊急処理 170
システム アラーム 171
サーマルシャットダウンの無効化 172
電源管理 174
電源モード 174
動作状態 174
show power detail 176
電源管理の考慮事項 178
電源モードの選択 179
冗長モードの設定 180
複合モードの設定 181
スーパバイザモジュールの電力バジェット 182
シングルスーパバイザの電力バジェットモードの設定 183
シングル スーパバイザ セットアップからデュアル スーパバイザ セットアップへの移行 184
自動ラインカードシャットダウンの有効化 185
ラインカードの電源切断 187
電源モードと動作状態の設定例 188
例：複合モードと状態（AC および DC 入力） 188
例：複合モードと状態（DC 入力のみ） 189
例：n+1 冗長モード（容量が同じでタイプが異なる電源モジュール＋通常保護状態） 191
例：n+1 冗長モード（容量とタイプが同じ電源モジュール＋通常保護状態） 192
例：n+n 冗長モード（同じ容量の電源モジュール＋完全保護状態） 193
例：n+n 冗長モード（容量が異なる電源モジュール＋通常保護状態） 195
環境モニタリングおよび電源管理の機能履歴と情報 196

第 6 章 SDM テンプレートの設定 199
SDM テンプレートに関する情報 199
SDM テンプレートとスイッチ スタック 201
SDM テンプレートの設定方法 202
SDM テンプレートの設定 202

SDM テンプレートのモニタリングおよびメンテナンス 203
SDM テンプレートの設定例 204
例：SDM テンプレートの表示 204
例：SDM テンプレートの設定 207
SDM テンプレートに関する追加情報 208
SDM テンプレートの設定の機能履歴と情報 208

第 7 章 システム メッセージ ログの設定 209
システム メッセージ ログの設定に関する情報 209
システム メッセージ ロギング 209
システム ログ メッセージのフォーマット 210
デフォルトのシステム メッセージ ロギングの設定 211
syslog メッセージの制限 211
システム メッセージ ログの設定方法 212
メッセージ表示宛先デバイスの設定 212
ログ メッセージの同期化 213
メッセージ ロギングのディセーブル化 215
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 216
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 217
メッセージ重大度の定義 218
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 219
UNIX Syslog デーモンへのメッセージのロギング 219
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 221
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 221
システム メッセージ ログの設定例 221
例：システム メッセージのスタック構成 221
例：スイッチ システム メッセージ 221
システム メッセージ ログに関する追加情報 222
システム メッセージ ログの機能履歴と情報 222

第 8 章 オンライン診断の設定 223

オンライン診断の設定に関する情報 223
Generic Online Diagnostics（GOLD）テスト 224
オンライン診断の設定方法 228
オンライン診断テストの開始 228
オンライン診断の設定 229
オンライン診断のスケジューリング 229
ヘルス モニタリング診断の設定 231
オンライン診断のモニタリングおよびメンテナンス 234
オンライン診断のコンフィギュレーション例 234
例：診断テストの開始 235
例：ヘルスモニタリングテストの設定 235
例：診断テストのスケジューリング 235
例：オンライン診断の表示 235
オンライン診断に関する追加情報 236
オンライン診断設定の機能情報 236

第 9 章 コンフィギュレーション ファイルの管理 239
コンフィギュレーション ファイルの管理の前提条件 239
コンフィギュレーション ファイルの管理の制約事項 239
コンフィギュレーション ファイルの管理について 240
コンフィギュレーション ファイルのタイプ 240
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 240
CLI を使用したコンフィギュレーション ファイルの変更 241
コンフィギュレーション ファイルの場所 241
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 242
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 242
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 243
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 245
VRF によるファイルのコピー 246
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 246
NVRAM より大きいコンフィギュレーション ファイル 246

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 248
コンフィギュレーション ファイル情報の管理方法 248
コンフィギュレーション ファイル情報の表示 248
コンフィギュレーション ファイルの変更 249
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 251
次の作業 252
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 252
例 253
次の作業 254
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 254
例 255
次の作業 256
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 256
次の作業 257
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 257
例 258
次の作業 259
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 259
例 260
次の作業 261
NVRAM より大きいコンフィギュレーション ファイルの保守 261
コンフィギュレーション ファイルの圧縮 261
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 262
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 264
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 265
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 266
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 267
次の作業 268

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 269
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 270
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 270
スタートアップ コンフィギュレーションのクリア 271
指定されたコンフィギュレーション ファイルの削除 272
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 273
次の作業 275
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 276
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 276
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 277
コンフィギュレーション ファイルの機能履歴と情報 279

第 10 章 セキュア コピー 281
セキュア コピーの前提条件 281
Secure Copy に関する情報 281
セキュアコピーのパフォーマンス向上 282
セキュア コピーの設定方法 282
セキュア コピーの設定 282
SSH サーバでのセキュアコピーのイネーブル化 283
セキュア コピーの設定例 285
例：ローカル認証を使用したセキュア コピーの設定 285
例：ネットワークベース認証を使用したセキュアコピーのサーバ側の設定 286
セキュアコピーに関する追加情報 286
セキュア コピーの機能情報 286

第 11 章 コンフィギュレーションの置換とロールバック 289
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 289
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 290

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 290
コンフィギュレーション アーカイブ 290
コンフィギュレーションの置換 291
コンフィギュレーション ロールバック 292
コンフィギュレーション ロールバック変更確認 293
コンフィギュレーションの置換とロールバックの利点 293
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 293
コンフィギュレーション アーカイブの作成 293
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 295
機能のモニタリングおよびトラブルシューティング 298
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 301
コンフィギュレーション アーカイブの作成 301
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 301
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 302
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 302
コンフィギュレーション ロールバック操作の実行 302
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 304
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と情報 304

第 12 章 初期設定へのリセットの実行 305
初期設定へのリセット実行の前提条件 305
初期設定へのリセット実行の制限事項 305
初期設定へのリセットの実行に関する情報 306
初期設定へのリセットの実行方法 307
初期設定へのリセットを実行するための設定例 308
初期設定へのリセットの実行に関する追加情報 312
初期設定へのリセットに関する機能情報 312

第 13 章 セキュア ストレージの設定 315

セキュア ストレージについて 315
セキュア ストレージの有効化 315
セキュア ストレージの無効化 316
暗号化のステータスの確認 317
セキュアストレージの機能情報 317

第 14 章 BIOS 保護 319
BIOS 保護の概要 319
ROMMON アップグレード 319
カプセルアップグレード 320
BIOS 保護の機能情報と情報 321

第 15 章 ソフトウェア メンテナンス アップグレード 323
ソフトウェア メンテナンス アップグレードについて 323
SMU の概要 323
SMU のワークフロー 324
SMU パッケージ 324
SMU のリロード 324
ソフトウェア メンテナンスの更新の管理方法 324
SMU パッケージのインストール 325
SMU パッケージの管理 326
ソフトウェア メンテナンス アップグレードの設定例 327
例：SMU の管理 327
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 332
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 332

第 16 章 フラッシュ ファイル システムの操作 335
フラッシュ ファイル システムについて 335
使用可能なファイル システムの表示 336
デフォルト ファイル システムの設定 337
ファイル システムのファイルに関する情報の表示 338

ディレクトリの変更および作業ディレクトリの表示 339
ディレクトリの作成 340
ディレクトリの削除 340
ファイルのコピー 341
ファイルの削除 341
ファイルの作成、表示、および抽出 342
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 344
フラッシュ ファイル システムの機能履歴と情報 345

第 17 章 条件付きデバッグとラジオアクティブ トレース 347
条件付きデバッグの概要 347
ラジオアクティブ トレースの概要 348
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 348
条件付きデバッグおよび放射線トレース 348
トレースファイルの場所 348
条件付きデバッグの設定 349
L2 マルチキャストの放射線トレース 351
トレース ファイルの推奨ワークフロー 351
ボックス外へのトレース ファイルのコピー 351
条件付きデバッグのモニタリング 352
条件付きデバッグの設定例 353
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 353
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と情報 354

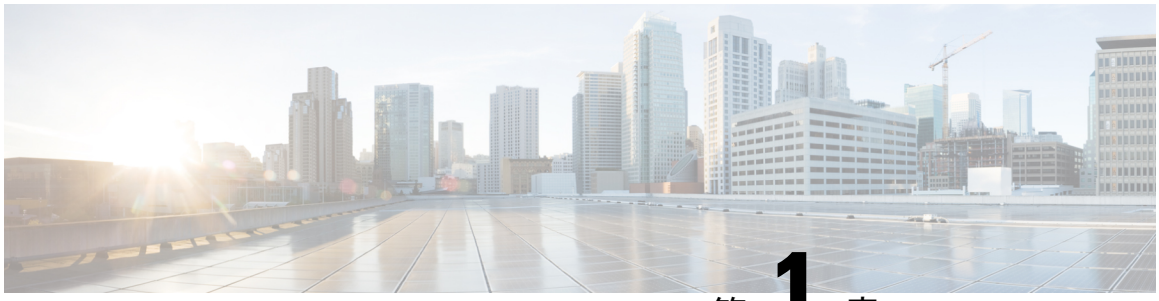
第 18 章 同意トークン 355
同意トークンの制約事項 355
同意トークンに関する情報 356
システムシェルアクセスの同意トークン承認プロセス 356
同意トークンの機能履歴と情報 358

第 19 章 ソフトウェア設定のトラブルシューティング 359

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 359
スイッチのソフトウェア障害 359
デバイスのパスワードを紛失したか忘れた場合 360
Power over Ethernet（PoE）ポート 360
電力消失によるポートの障害 361
不正リンク アップによるポート障害 361
ping 361
レイヤ 2 Traceroute 362
レイヤ 2 の traceroute のガイドライン 362
IP トレースルート 363
Time Domain Reflector ガイドライン 364
debug コマンド 365
システム レポート 365
スイッチのオンボード障害ロギング 368
ファン障害 368
CPU 使用率が高い場合に起こりうる症状 369
ソフトウェア設定のトラブルシューティング方法 369
ソフトウェア障害からの回復 369
パスワードを忘れた場合の回復 372
パスワード回復がイネーブルになっている場合の手順 373
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 375
スイッチ スタック問題の回避 376
自動ネゴシエーションの不一致の防止 377
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 378
SFP モジュール ステータスのモニタリング 379
ping の実行 379
温度のモニタリング 379
物理パスのモニタリング 379
IP traceroute の実行 380
TDR の実行および結果の表示 380
デバッグおよびエラー メッセージ出力のリダイレクト 380

show platform コマンドの使用 381
show debug コマンドの使用方法 381
パケット損失のトラブルシューティング 381
モジュールがオンラインでない場合のトラブルシューティング 382
インターフェイスの問題のトラブルシューティング 383
ワークステーションがネットワークにログインできない場合のトラブルシューティング 383
ソフトウェア設定のトラブルシューティングの確認 384
OBFL 情報の表示 384
例：高い CPU 使用率に関する問題と原因の確認 385
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 386
Power over Ethernet（PoE）に関するトラブルシューティングのシナリオ 386
ソフトウェアのトラブルシューティングの設定例 391
例：IP ホストの ping 391
例：IP ホストに対する traceroute の実行 392
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 393
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と情報 393

第 20 章 破損または損失したファイル イメージ、あるいは ROMmon モードでのリカバリ 395
はじめに 395
破損または損失したイメージからの ROMmon モードでのスイッチのリカバリ 396
連続的なリブートからのスイッチのリカバリ 397
破損または損失したイメージからのリカバリ 400
スイッチの回復に関する機能情報 408



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 （30 ページ）
- デバイス管理に関する追加情報 （33 ページ）
- デバイス管理の機能履歴と情報 （33 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* 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できます。

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。

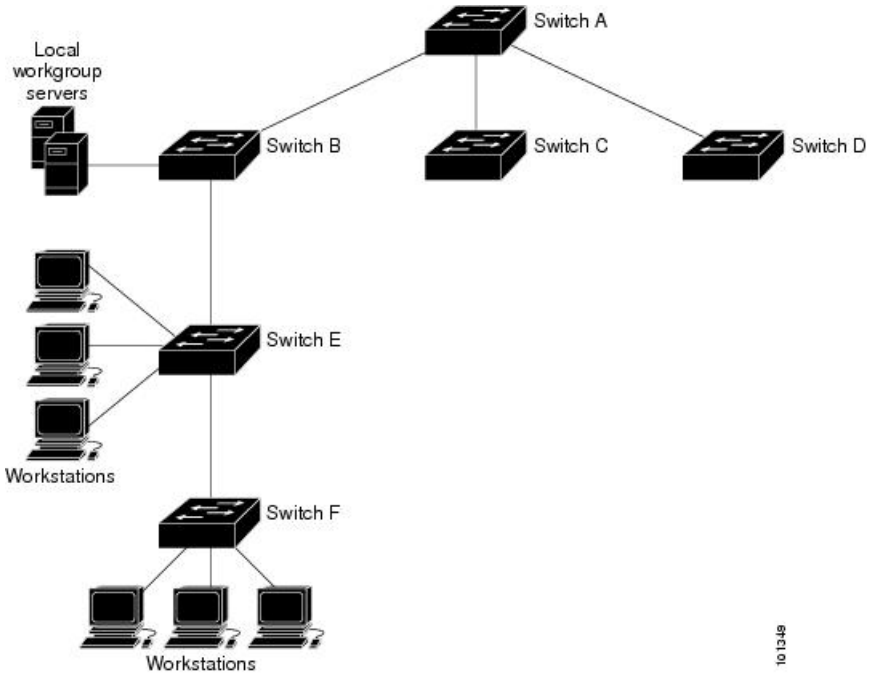
NTPが稼働するデバイス間の通信（アソシエーション）は、通常静的に設定されます。各デバイスには、アソシエーションを作成すべきすべてのデバイスの IP アドレスが与えられます。アソシエーションのペアとなるデバイス間で NTP メッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN 環境では、代わりに IP ブロードキャスト メッセージを使用するようにNTPを設定できます。各デバイスを、単にブロードキャスト メッセージを送受信するように設定すればよいので、この代替手段によって設定の複雑さが緩和されます。ただし、情報の流れは一方向に限られます。

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

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。

（注） Message Direct 5（MD5）認証の設定は推奨しません。より強力な暗号化のためにサポートされている他の認証方式を使用できます。

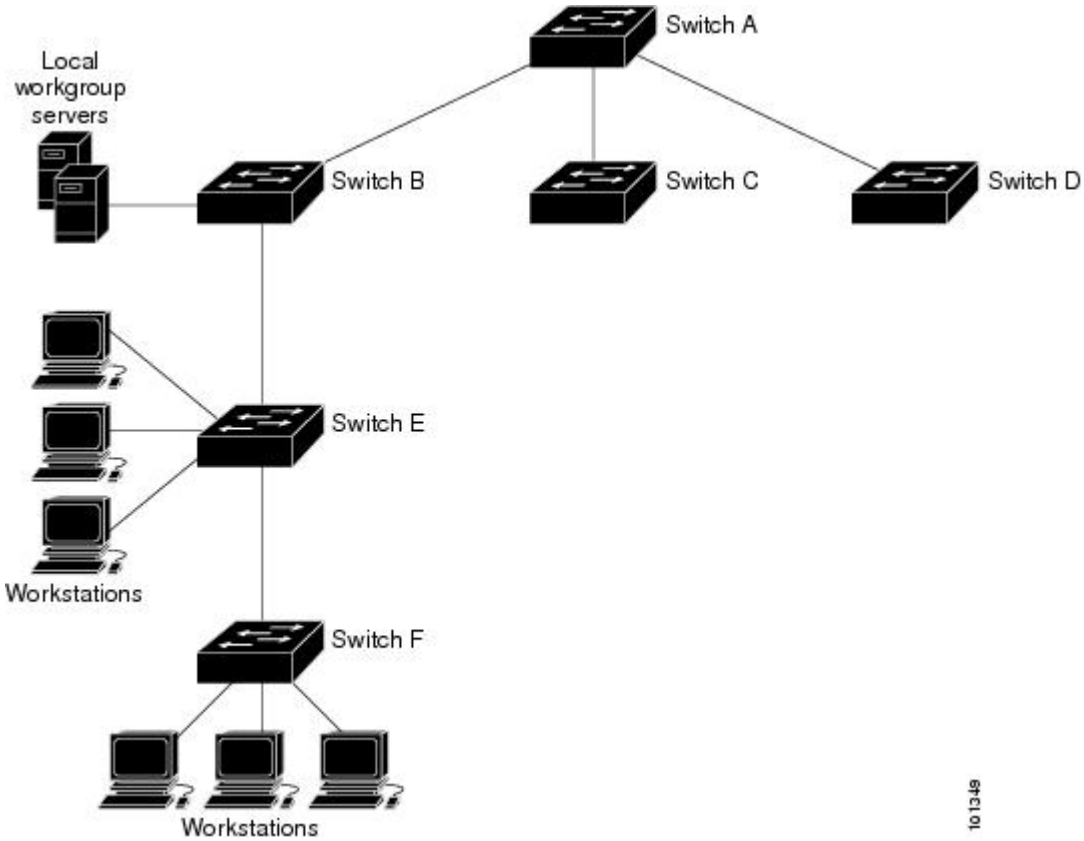
## NTP の実装

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

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。




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 。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ま

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と関連付けされます。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 アドレスが見つかると、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）がデフォルトでイネーブル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。

### 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 | **enable**<br>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` | • パスワードを入力します（要求された場合）。 |
| ステップ 2 | 次のいずれかを使用します。<br>• **clock set** *hh:mm:ss day month year*<br>• **clock set** *hh:mm:ss month day year*<br>例：<br>`Device# clock set 13:32:00 23 March 2013` | 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。<br>• *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。<br>• *day*：月の日で日付を指定します。<br>• *month*：月を名前で指定します。<br>• *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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 timezone** *zone hours-offset* [*minutes-offset*]<br>例：<br>`Device(config)# clock timezone AST -3 30` | 時間帯を設定します。<br>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で、時刻を手動で設定するときだけに使用されます。<br>• *zone*：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 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | を入力します。デフォルトは UTC です。<br>• *hours-offset*：UTC からのオフセット時間数を入力します。<br>•（任意）*minutes-offset*：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 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 6 | **copy running-config startup-config**<br>例：<br>`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。 |

## 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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*]]<br>例：<br>`Device(config)# clock summer-time PDT date`<br>`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*]]<br>例：<br>`Device(config)# clock summer-time PDT recurring 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。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。<br>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。<br>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。<br>• *zone*：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。<br>•（任意）*week*：月の週（1 ～ 4、**first**、または **last**）を指定します。<br>•（任意）*day*：曜日（Sunday、Monday など）を指定します。<br>•（任意）*month*：月（January、February など）を指定します。<br>•（任意）*hh:mm*：時および分単位で時間（24時間形式）を指定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • （任意）*offset*：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**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 6 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 7 | **copy running-config startup-config**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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*<br>例：<br>`Device(config)# hostname remote-users` | 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。<br>デフォルト設定は Switch です。<br>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**<br>例：<br>`remote-users(config)#end`<br>`remote-users#` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 6 | **copy running-config startup-config**<br>例：<br>`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。 |

# 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 クエリーが行われ、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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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*<br>例：<br>`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。<br>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。<br>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*]<br>例：<br>`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。<br>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*]<br>例：<br>`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。<br>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**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 7 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 8 | **copy running-config startup-config**<br>例：<br>`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**<br>例： | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>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*<br>例：<br>Device(config)# **banner motd #**<br>This is a secure site. Only authorized users are allowed.<br>For access, contact technical support.<br># | MoTD を指定します。<br>*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。<br>*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**<br>例：<br>Device(config)# **end** | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show running-config**<br>例：<br>Device# **show running-config** | 入力を確認します。 |
| ステップ 6 | **copy running-config startup-config**<br>例：<br>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 バナーの後で、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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 login** *c message c*<br>例：<br>`Device(config)# banner login $`<br>`Access for authorized users only.`<br>`Please enter your username and password.`<br>`$` | ログイン メッセージを指定します。<br>*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。<br>*message*：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。 |
| ステップ 4 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 6 | **copy running-config startup-config**<br>例：<br>`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 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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*]<br>例：<br>`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。<br>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。<br>*vlan-id*：有効な ID は 1 ～ 4094 です。 |
| ステップ 4 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 6 | **copy running-config startup-config**<br>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# copy running-config startup-config` | |

## 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 | **snmp-server host** *host-addr community-string notification-type* { **informs** \| **traps** } {**version** {**1** \| **2c** \| **3**}} { **vrf** *vrf instance name*}<br>例：<br>`Device(config)# snmp-server host 172.20.10.10 traps private mac-notification` | トラップ メッセージの受信側を指定します。<br>• *host-addr*：NMS の名前またはアドレスを指定します。<br>• **traps**（デフォルト）：ホストに SNMP トラップを送信します。<br>• **informs**：ホストに SNMP 情報を送信します。<br>• **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。<br>• *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 |
|---|---|---|
| | | 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。<br>• *notification-type*：**mac-notification** キーワードを使用します。<br>• **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 | **snmp-server enable traps mac-notification change**<br>例：<br>`Device(config)# snmp-server enable traps mac-notification change` | 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。 |
| ステップ 5 | **mac address-table notification change**<br>例：<br>`Device(config)# mac address-table notification change` | MACアドレス変更通知機能をイネーブルにします。 |
| ステップ 6 | **mac address-table notification change** [ **interval** *value*] [ **history-size** *value*]<br>例：<br>`Device(config)# mac address-table notification change interval 123`<br>`Device(config)#mac address-table notification change history-size 100` | 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。<br>• （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。<br>• （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 | **interface** *interface-id*<br>例： | 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 |

| | 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**}<br>例：<br>`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。<br>• 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。<br>• 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**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 10 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 11 | **copy running-config startup-config**<br>例：<br>`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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 | **snmp-server host** *host-addr* {**traps** \| **informs**} {**version** {**1** \| **2c** \| **3**}} *community-string notification-type*<br>例：<br>`Device(config)# snmp-server host 172.20.10.10 traps private mac-notification` | トラップ メッセージの受信側を指定します。<br>• *host-addr*：NMS の名前またはアドレスを指定します。<br>• **traps**（デフォルト）：ホストに SNMP トラップを送信します。<br>• **informs**：ホストに SNMP 情報を送信します。<br>• **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。<br>• *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。<br>• *notification-type*：**mac-notification** キーワードを使用します。 |
| ステップ 4 | **snmp-server enable traps mac-notification move**<br>例： | 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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# snmp-server enable traps mac-notification move` | |
| ステップ 5 | **mac address-table notification mac-move**<br>例：<br>`Device(config)# mac address-table notification mac-move` | 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**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 7 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 8 | **copy running-config startup-config**<br>例：<br>`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。 |

### 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 | **snmp-server host** *host-addr* { **traps** \| **informs**} {**version** {**1** \| **2c** \| **3**}} *community-string notification-type*<br>例：<br>`Device(config)# snmp-server host 172.20.10.10 traps private mac-notification` | トラップ メッセージの受信側を指定します。<br>• *host-addr*：NMS の名前またはアドレスを指定します。<br>• **traps**（デフォルト）：ホストに SNMP トラップを送信します。<br>• **informs**：ホストに SNMP 情報を送信します。<br>• **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。<br>• *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。<br>• *notification-type*：**mac-notification** キーワードを使用します。 |
| ステップ 4 | **snmp-server enable traps mac-notification threshold**<br>例： | NMS への MAC しきい値通知トラップ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(config)# snmp-server enable traps mac-notification threshold` | |
| ステップ 5 | **mac address-table notification threshold**<br>例：<br>`Device(config)# mac address-table notification threshold` | 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 | **mac address-table notification threshold** [ **limit** *percentage*] | [ **interval** *time*]<br>例：<br>`Device(config)# mac address-table notification threshold interval 123`<br>`Device(config)# mac address-table notification threshold limit 78` | 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。<br>• （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。<br>• （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 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 8 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 9 | **copy running-config startup-config**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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*<br>例：<br>`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。<br>• *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。<br>• *vlan-id*：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。<br>• *interface-id*：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**<br>例： | 入力を確認します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device# show running-config` | |
| ステップ 5 | **copy running-config startup-config**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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**<br>例：<br>`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。<br>• *mac-addr*：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。<br>• *vlan-id*：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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 4 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show running-config**<br>例：<br>`Device# show running-config` | 入力を確認します。 |
| ステップ 6 | **copy running-config startup-config**<br>例：<br>`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。 |
| **clear mac address-table dynamic interface** *interface-id* | 指定された物理ポートまたはポート チャネル上のすべてのアドレスを削除します。 |
| **clear mac address-table dynamic vlan** *vlan-id* | 指定された VLAN 上のすべてのアドレスを削除します。 |
| **show clock** [*detail*] | 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** *mac-address* | 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。 |

| コマンド | 目的 |
|---|---|
| **show mac address-table interface** *interface-name* | 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** *vlan-id* | 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。VLAN 4 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
```

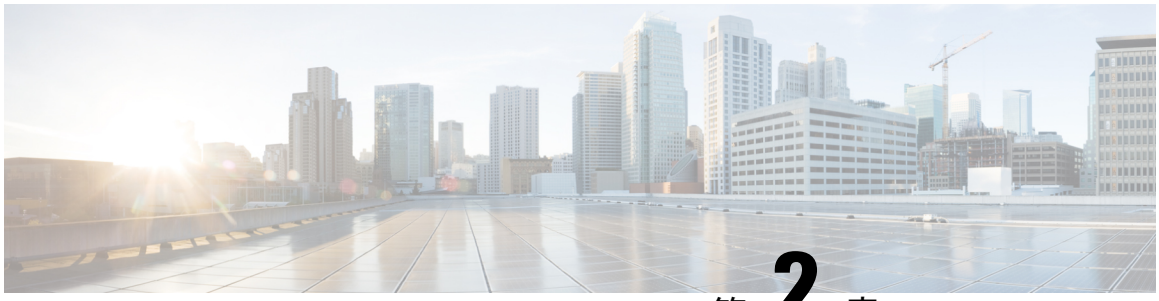
# デバイス管理に関する追加情報

### 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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# デバイス管理の機能履歴と情報

| リリース | 変更内容 |
|---|---|
| | この機能が導入されました。 |



第 2 章

# デバイスのセットアップ設定の実行


- デバイスセットアップの設定の制約事項 （35 ページ）
- デバイスセットアップ設定の実行に関する情報 （35 ページ）
- デバイスセットアップ設定の実行方法 （51 ページ）
- デバイスのセットアップの設定例 （66 ページ）
- デバイスセットアップの実行に関する追加情報 （87 ページ）
- デバイスセットアップ設定の実行に関する機能情報 （87 ページ）


## デバイスセットアップの設定の制約事項

- サブパッケージソフトウェアのインストールはサポートされていません。
- アクティブとスタンバイのルータ プロセッサ（RT）間での自動アップグレード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

ここでは、IP アドレス割り当てと Dynamic Host Configuration Protocol（DHCP）の自動設定を含む、デバイスセットアップの設定方法について説明します。

### デバイスブートプロセス

デバイスを起動するには、『*Cisco Catalyst 9400* シリーズ スイッチ ハードウェア設置ガイド』に記載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。このプロセスでは、物理メモリのマッピング場所、物理メモリの量と速度などを制御する CPU レジスタを初期化します。

- 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。
- 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。POST の一環として、次のテストも実行されます。
  - スーパバイザエンジンと各モジュールに接続されたネットワークポート間のデータパスを確認する MAC ループバック テスト。いずれかのポートでこのテストが失敗すると、ポートは強制的に error-disabled ステートになり、モジュールは **show module** コマンド出力で *post-fail* としてマークされます。

サポートされるオンライン診断の完全なリストについては、「オンライン診断の設定」の章を参照してください。

ブート ローダにより、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 ローダの使用目的は通常、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。

また、オペレーティング システムが使用不可能になるほどの重大な障害が発生した場合は、ブートローダはシステムにトラップドアからアクセスします。トラップドア メカニズムによりシステムへのアクセスを十分に行うことで、必要に応じて、XMODEM プロトコルを使用してオペレーティングシステムのソフトウェアイメージを再インストールし、失われたパスワードを回復し、最終的にオペレーティング システムを再起動できます。

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）、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_iosxe.16.06.01.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**、 **install activate**、および **install commit** 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。

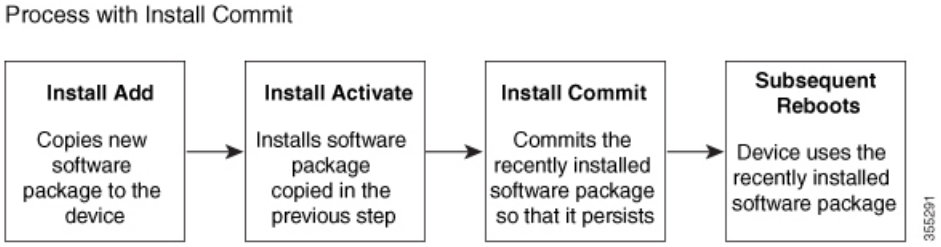
**install activate** コマンドを動作させるには、パッケージをデバイスのブートフラッシュで使用可能にする必要があります。このコマンドを設定すると、.bin ファイルから以前に追加したパッケージがアクティブ化され、システムがリロードします。

**install commit** コマンドを有効化して、更新プログラムをリロード全体にわたって確定します。

更新をインストールすると、以前にインストールしたソフトウェアイメージが置換されます。デバイスには常に 1 つのイメージのみがインストールされます。

次のフロー チャートで、ソフトウェアのインストールの動作を説明します。

**図 *3:* ソフトウェア パッケージのコミット**




（注） **install activate** コマンドは、新しいイメージを使用してデバイスをリロードします。

## 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応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 （35 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。

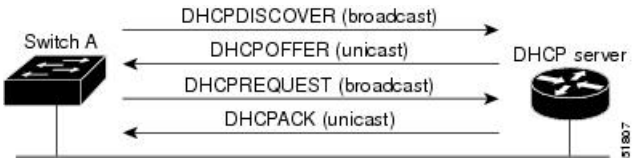
DHCP ベースの自動設定は、デバイスの BOOTP クライアント機能に代わるものです。

## 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 *4 : 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す。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 DCHPDISCOVER 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、cisconet.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-confg または cisconet.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-confg または ciscortr.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-confg 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-confg 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-confg 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-confg、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-confg ファイルを読み込みます。router-confg 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。すべてのオレンジのシステム LED が点灯したままになったら、[Mode] ボタンを放してもかまいません。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、ブートローダー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。通常の環境では、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 | **set BOOT** *filesystem* **:/** *file-url ...*<br><br>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。 | **boot system** {*filesystem* **:** */file-url ...* | **switch** {*number* | **all**}}<br><br>次回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。<br><br>パッケージプロビジョニング ファイルは、*packages.conf* 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。<br>• 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。<br>• 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 | **set MANUAL_BOOT yes**<br>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。<br>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。 | **boot manual**<br>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。MANUAL_BOOT 環境変数の設定が変更されます。<br>次回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。 |
| CONFIG_FILE | **set CONFIG_FILE flash:/** *file-url*<br>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。 | **boot config-file flash:/** *file-url*<br>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。 |
| SWITCH_NUMBER | **set SWITCH_NUMBER** *stack-member-number*<br>スタック メンバのメンバ番号を変更します。 | **switch** *current-stack-member-number* **renumber** *new-stack-member-number*<br>スタック メンバのメンバ番号を変更します。 |
| SWITCH_PRIORITY | **set SWITCH_PRIORITY** *stack-member-number*<br>スタック メンバのプライオリティ値を変更します。 | **switch** *stack-member-number* **priority** *priority-number*<br>スタック メンバのプライオリティ値を変更します。 |
| BAUD | **set BAUD** *baud-rate* | **line console 0**<br>**speed***speed-value*<br>ボー レートを設定します。 |
| ENABLE_BREAK | **set ENABLE_BREAK yes/no** | **boot enable-break switch yes/no**<br>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。 |

## 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 | スイッチの MAC アドレスを指定します。<br>（注） 変数は変更しないことを推奨します。<br>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。 |
| IP_ADDRESS | 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。 |
| DEFAULT_GATEWAY | 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。 |

# 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**<br>例：<br>`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*<br>例：<br>`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 | **boot** *filename*<br>例：<br>`Device(dhcp-config)# boot config-boot.text` | 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 | **network** *network-number mask prefix-length*<br>例：<br>`Device(dhcp-config)# network 10.10.10.0 255.255.255.0` | 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。<br>（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。 |
| ステップ 5 | **default-router** *address*<br>例：<br>`Device(dhcp-config)# default-router 10.10.10.1` | 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 | **option 150** *address*<br>例：<br>`Device(dhcp-config)# option 150 10.10.10.1` | TFTP サーバの IP アドレスを指定します。 |
| ステップ 7 | **exit**<br>例：<br>`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 8 | **tftp-server flash**:*filename.text*<br>例：<br>`Device(config)# tftp-server flash:config-boot.text` | 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** *interface-id*<br>例：<br>`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。 |
| ステップ 10 | **no switchport**<br>例：<br>`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** *address mask*<br>例：<br>`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**<br>例：<br>`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**<br>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 | **ip dhcp pool** *poolname*<br>例：<br>`Device(config)# ip dhcp pool pool1` | 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 | **boot** *filename*<br>例：<br>`Device(dhcp-config)# boot config-boot.text` | 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 | **network** *network-number mask prefix-length*<br>例：<br>`Device(dhcp-config)# network 10.10.10.0 255.255.255.0` | 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。<br>（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。 |
| ステップ 5 | **default-router** *address*<br>例：<br>`Device(dhcp-config)# default-router 10.10.10.1` | 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 | **option 150** *address*<br>例：<br>`Device(dhcp-config)# option 150 10.10.10.1` | TFTP サーバの IP アドレスを指定します。 |
| ステップ 7 | **option 125** *hex*<br>例： | 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 |
|---|---|---|
| | `Device(dhcp-config)# option 125 hex 0000.0009.0a05.0861.7574.6f69.6e73.7461.6c6c.5f64.686370` | |
| ステップ 8 | **copy tftp flash** *filename.txt*<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`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.text*<br>例：<br>`Device(config)# tftp-server flash:config-boot.text` | 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*<br>例：<br>`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*<br>例：<br>`Device(config)# tftp-server flash:boot-config.text` | 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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 14 | **interface** *interface-id*<br>例：<br>`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。 |
| ステップ 15 | **no switchport**<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>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 | **boot host dhcp**<br>例：<br>`Device(conf)# boot host dhcp` | 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。 |
| ステップ 3 | **boot host retry timeout** *timeout-value*<br>例：<br>`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。<br>（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** *warning-message* **^C**<br>例：<br>`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**<br>例：<br>`Device(config-if)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 6 | **show boot**<br>例：<br>`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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **interface vlan** *vlan-id*<br>例：<br>`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 4 | **ip address** *ip-address subnet-mask*<br>例：<br>`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 5 | **exit**<br>例：<br>`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 6 | **ip default-gateway** *ip-address*<br>例：<br>`Device(config)# ip default-gateway 10.10.10.1` | 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。<br>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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | （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。<br><br>（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。 |
| ステップ 7 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 8 | **show interfaces vlan** *vlan-id*<br>例：<br>`Device# show interfaces vlan 99` | 指定した VLAN のインターフェイスステータスを表示します。 |
| ステップ 9 | **show ip redirects**<br>例：<br>`Device# show ip redirects` | Internet Control Message Protocol（ICMP）リダイレクトメッセージを表示します。 |

# 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

次のセクションでは、デバイスのスタートアップ コンフィギュレーションを変更する方法について説明します。

## 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 flash**:*/file-url*<br>例：<br>`Device(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。<br>• *file-url*：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。<br>• 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 4 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show boot**<br>例：<br>`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。<br>• **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 6 | **copy running-config startup-config**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。 |
| ステップ 3 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 4 | **show boot**<br>例：<br>`Device# show boot` | 入力を確認します。<br>**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。<br>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 *switch:* 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。<br>• *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。<br>`Switch: boot flash:` |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • *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。<br><br>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**<br><br>例：<br><br>`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 | **enable**<br><br>例：<br><br>`Device> enable` | 特権 EXEC モードを有効にします。<br><br>• 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 tftp:** *filename* [**activate commit**]<br><br>例：<br><br>`Device# install add file tftp://172.16.0.1//tftpboot/folder1/cat9k_iosxe.16.06.01.SPA.bin activate commit` | 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。<br><br>• 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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • 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 | **exit**<br>例：<br>`Device# exit` | 特権 EXEC モードを終了し、ユーザ EXEC モードに戻ります。 |

## 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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 tftp:** *filename*<br>例：<br>`Device# install add file tftp://172.16.0.1//tftpboot/folder1/cat9k_iosxe.16.06.01.SPA.bin` | 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。<br>• 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**]<br>例：<br>`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。<br>• 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。<br>• **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。<br>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 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | 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**<br>例：<br>`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。<br>• 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**<br>例：<br>`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。<br>• **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。 |
| ステップ 6 | **install rollback to committed**<br>例：<br>`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**}<br>例：<br>`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**<br>例：<br>`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。<br>• 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*><br>例：<br>`switch:BOOT=tftp://10.0.0.2/cat9k_iosxe.16.06.01.SPA.bin`<br>`switch:`<br>`switch:`<br>`switch:` | ブート パラメータを設定します。 |
| ステップ 2 | **boot**<br>例：<br>`switch:boot` | デバイスを起動します。 |
| ステップ 3 | **show version** | （任意）インストールされているイメージのバージョンを表示します。 |

## 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 | **copy running-config startup-config**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 4 | **reload in** [*hh*:]*mm* [*text*]<br>例：<br>`Device# reload in 12`<br>`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 5 | **reload at** *hh*: *mm* [*month day* \| *day month*] [*text*]<br>例：<br>`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。<br>（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。 |
| ステップ 6 | **reload cancel**<br>例：<br>`Device(config)# reload cancel` | 以前にスケジューリングされたリロードをキャンセルします。 |
| ステップ 7 | **show reload**<br>例：<br>`show reload` | 以前デバイスにスケジューリングされたリ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。 |

# デバイスのセットアップの設定例

次のセクションにデバイスセットアップの設定例を示します。

## 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

次の例では、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 bootflash:packages.conf
attempting to boot from [bootflash:packages.conf]


Located file packages.conf
#
################################################
```

```
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 WS-XC7R (X86) processor (revision V00) with 869290K/6147K bytes of memory.
Processor board ID FXS1939Q43K
8 Ten Gigabit Ethernet interfaces
2 Forty Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
15958516K bytes of physical memory.
11161600K bytes of Bootflash at bootflash:.
1638400K bytes of Crash Files at crashinfo:.
0K bytes of WebUI ODM Files at webui:.

%INIT: waited 0 seconds for NVRAM to be available

vstack
^
% Invalid input detected at '^' marker.



Press RETURN to get started!
```

次の例では、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 bootflash:cat9k_iosxe.BLD_V166_THROTTLE_LATEST_20170902_091308_2.SSA.bin
attempting to boot from
[bootflash:cat9k_iosxe.BLD_V166_THROTTLE_LATEST_20170902_091308_2.SSA.bin]

Located file cat9k_iosxe.BLD_V166_THROTTLE_LATEST_20170902_091308_2.SSA.bin
############################################################
```

```
ARDBEG device bus 22: slot 0: fn 0 cpld regs ffffc90000400000 cpldiomem = ffffc90000600000
 mmio start  87f99000000 , len 1048576
chr_mmap: Allocating DMA Reserve Pool ...

dplr_intrpt: Entered dplr_intrpt_module_init dplr_intrpt 1

usb 1-1: device not accepting address 2, error -110

usb 1-1: device not accepting address 3, error -110

usb 1-1: device not accepting address 4, error -110

usb 1-1: device not accepting address 5, error -110

usb usb1-port1: unable to enumerate USB device


              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 [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Experimental
Version 16.6.20170902:081931
[v166_throttle-/scratch/mcpre/BLD-BLD_V166_THROTTLE_LATEST_20170902_091308 126]


Copyright (c) 1986-2017 by Cisco Systems, Inc.

Compiled Sat 02-Sep-17 06:26 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 WS-XC7R (X86) processor (revision V00) with 869290K/6147K bytes of memory.
Processor board ID FXS1939Q43K
8 Ten Gigabit Ethernet interfaces
2 Forty Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
15958516K bytes of physical memory.
11161600K bytes of Bootflash at bootflash:.
1638400K bytes of Crash Files at crashinfo:.
0K bytes of WebUI ODM Files at webui:.

%INIT: waited 0 seconds for NVRAM to be available

vstack
^
% Invalid input detected at '^' marker.



Press RETURN to get started!
```

# 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 tftp://172.16.0.1//tftpboot/folder1/
cat9k_iosxe.16.06.01.SPA.bin

install_add: START Fri Jun 23 21:07:59 IST 2017
install_add: Adding PACKAGE

--- Starting Add ---
Performing Add on Active/Standby
issu_helper.sh (13338): drop_caches: 3
  [R0] Add package(s) on R0
  [R0] Finished Add on R0
Checking status of Add on [R0]
Add: Passed on [R0]
Finished Add

SUCCESS: install_add  Fri Jun 23 21:09:10 IST 2017
Device#
```

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

[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
```

```
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   I    16.6.1.0
IMG   C    16.6.2.0
```

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。

```
Device# install activate

install_activate: START Fri Jun 23 21:13:25 IST 2017
install_activate: Activating PACKAGE
ISOFS: Unable to identify CD-ROM format.
Following packages shall be activated:
/flash/cat9k-webui.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
/flash/cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg

This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on Active/Standby
  [R0] Activate package(s) on R0
    --- Starting list of software package changes ---
    Old files list:
      Removed cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-espbase.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
      Removed cat9k-webui.BLD_POLARIS_DEV_LATEST_20170622_233647.SSA.pkg
    New files list:
      Added cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
      Added cat9k-webui.BLD_V166_THROTTLE_LATEST_20170622_152342.SSA.pkg
    Finished list of software package changes
  [R0] Finished Activate on R0
Checking status of Activate on [R0]
Activate: Passed on [R0]
Finished Activate

Install will reload the system now!

Device#
```

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。

```
Device# show install summary

[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   I    16.6.2.0
IMG   U    16.6.1.0
Device#
```

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

```
Device# install commit
install_commit: START Fri Jun 23 21:24:45 IST 2017
install_commit: Committing PACKAGE

--- Starting Commit ---
Performing Commit on Active/Standby
  [R0] Commit package(s) on R0
  [R0] Finished Commit on R0
Checking status of Commit on [R0]
Commit: Passed on [R0]
Finished Commit

SUCCESS: install_commit  Fri Jun 23 21:24:48 IST 2017

Device#
```

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。

```
Device# install rollback to committed

install_rollback: START Tue Jun 20 14:55:12 PDT 2017

 This operation requires a reload of the system. Do you want to proceed? [y/n]
*Jun 20 14:55:12.911 PDT: %IOSXE-5-PLATFORM:  R0/0: Jun 20 14:55:12 install_engine.sh:
 %INSTALL-5-INSTALL_START_INFO: Started install rollbacky
--- Starting Rollback ---
Performing Rollback on Active/Standby
  [R0] Rollback package(s) on R0
    --- Starting rollback impact ---
    Changes that are part of this rollback
    Current    :  rp 0 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :  rp 1 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Replacement:  rp 0 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:  rp 1 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Current    :   cc 0  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 0 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 0 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 1  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 1 0   cc
```

```
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 1 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 10 0  cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 10 0  cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 10 0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 2  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 2 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 2 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 3  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 3 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 3 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 4  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 4 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 4 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 5  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 5 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 5 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 6  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 6 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 6 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 7  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 7 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 7 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 8  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 8 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 8 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 9  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 9 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 9 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   fp 0 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   fp 1 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_base
```

```
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Replacement:   cc 0  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 0 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 0 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 1  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 1 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 1 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 10 0  cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 10 0  cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 10 0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 2  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 2 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 2 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 3  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 3 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 3 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 4  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 4 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 4 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 5  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 5 0   cc
```

```
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 5 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 6  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 6 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 6 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 7  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 7 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 7 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 8  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 8 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 8 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 9  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 9 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 9 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   fp 0 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   fp 1 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Finished rollback impact
  [R0] Finished Rollback on R0
Checking status of Rollback on [R0]
Rollback: Passed on [R0]
Finished Rollback
```

```
Install will reload the system now!
SUCCESS: install_rollback  Tue Jun 20 14:56:54 PDT 2017

Device#
```

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

```
Device# install remove inactive

install_remove: START Tue Jun 20 14:14:40 PDT 2017
Cleaning up unnecessary package files
No path specified, will use booted path flash:packages.conf
Cleaning flash:
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...
    cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      File is in use, will not delete.
    packages.conf
      File is in use, will not delete.
  done.


The following files will be deleted:
[R0]:
/flash/cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k_1.bin
/flash/cat9k_1.conf
/flash/cat9k_2.1.conf
/flash/cat9k_2.bin
/flash/cat9k_2.conf
/flash/cat9k_iosxe.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.bin
/flash/packages.conf.00-


Do you want to remove the above files? [y/n]y
[R0]:
Deleting file flash:cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
```

```
 ... done.
Deleting file flash:cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg ...
 done.
Deleting file flash:cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
 ... done.
Deleting file flash:cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg ...
 done.
Deleting file flash:cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg ...
 done.
Deleting file flash:cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg ...
 done.
Deleting file flash:cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg ...
 done.
Deleting file flash:cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
... done.
Deleting file flash:cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg ...
done.
Deleting file flash:cat9k_1.bin ... done.
Deleting file flash:cat9k_1.conf ... done.
Deleting file flash:cat9k_2.1.conf ... done.
Deleting file flash:cat9k_2.bin ... done.
Deleting file flash:cat9k_2.conf ... done.
Deleting file flash:cat9k_iosxe.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.bin ...
done.
Deleting file flash:packages.conf.00- ... done.
SUCCESS: Files deleted.
--- Starting Post_Remove_Cleanup ---
Performing Post_Remove_Cleanup on Active/Standby
  [R0] Post_Remove_Cleanup package(s) on R0
  [R0] Finished Post_Remove_Cleanup on R0
Checking status of Post_Remove_Cleanup on [R0]
Post_Remove_Cleanup: Passed on [R0]
Finished Post_Remove_Cleanup


SUCCESS: install_remove  Tue Jun 20 14:16:29 PDT 2017
Device#
```

次に、**install abort** コマンドの出力例を示します。

```
Device# install abort

install_abort: START Tue Jun 20 14:06:48 PDT 2017
install_abort: Abort type PACKAGE

This install abort would require a reload. Do you want to proceed? [y/n]

*Jun 20 14:06:49.820 PDT:%IOSXE-5-PLATFORM:  R0/0: Jun 20 14:06:49 install_engine.sh:
 %INSTALL-5-INSTALL_START_INFO: Started install aborty
--- Starting Abort ---
Performing Abort on Active/Standby
  [R0] Abort package(s) on R0
    --- Starting rollback impact ---
    Changes that are part of this rollback
    Current    :  rp 0 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :  rp 1 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Replacement:  rp 0 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:  rp 1 0   rp_boot
cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
```

```
    Current    :   cc 0  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 0 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 0 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 1  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 1 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 1 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 10 0  cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 10 0  cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 10 0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 2  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 2 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 2 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 3  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 3 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 3 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 4  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 4 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 4 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 5  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 5 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 5 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 6  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 6 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 6 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 7  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 7 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 7 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 8  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 8 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 8 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 9  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   cc 9 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
```

```
    Current    :   cc 9 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   fp 0 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   fp 1 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 0 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Current    :   rp 1 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Replacement:   cc 0  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 0 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 0 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 1  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 1 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 1 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 10 0  cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 10 0  cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 10 0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 2  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 2 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 2 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 3  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 3 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 3 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
```

```
    Replacement:   cc 4  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 4 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 4 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 5  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 5 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 5 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 6  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 6 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 6 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 7  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 7 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 7 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 8  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 8 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 8 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 9  0  cc_srdriver
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 9 0   cc
cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   cc 9 0   cc_spa
cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   fp 0 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   fp 1 0   fp
cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 0 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   guestshell
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_base
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_daemons
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_iosd
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   rp_security
cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
```

```
    Replacement:   rp 1 0   rp_webui
cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Replacement:   rp 1 0   srdriver
cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    Finished rollback impact
  [R0] Finished Abort on R0
Checking status of Abort on [R0]
Abort: Passed on [R0]
Finished Abort


Install will reload the system now!
SUCCESS: install_abort  Tue Jun 20 14:08:25 PDT 2017

Device#
```

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

```
Device# install activate auto-abort-timer 30

install_activate: START Tue Jun 20 17:26:07 PDT 2017
install_activate: Activating PACKAGE


*Jun 20 17:26:08.572 PDT: %IOSXE-5-PLATFORM:  R0/0: Jun 20 17:26:08 install_engine.sh:
 %INSTALL-5-INSTALL_START_INFO: Started install activate


Following packages shall be activated:
/flash/cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
/flash/cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg


This operation requires a reload of the system. Do you want to proceed? [y/n]
*Jun 20 18:07:47.821 PDT: %ENVIRONMENTAL-6-NOTICE: Temp: DopplerD, Location: R0, State:
 Minor, Reading: 85 Celsius
*Jun 20 18:13:47.848 PDT: %ENVIRONMENTAL-6-NOTICE: Temp:    inlet, Location: R0, State:
 Minor, Reading: 46 Celsiusy
--- Starting Activate ---
Performing Activate on Active/Standby
  [R0] Activate package(s) on R0
    --- Starting list of software package changes ---
    Old files list:
      Removed cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
      Removed cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248.SSA.pkg
    New files list:
      Added cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
```

```
      Added cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
      Added cat9k-webui.BLD_V166_THROTTLE_LATEST_20170618_152248_2.SSA.pkg
    Finished list of software package changes
  [R0] Finished Activate on R0
Checking status of Activate on [R0]
Activate: Passed on [R0]
Finished Activate



*Jun 20 18:53:02.320 PDT: %IOSXE-5-PLATFORM:  R0/0: Jun 20 18:53:02 rollback_timer.sh:

 %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Install auto abort timer will expire in
1800 seconds
 Install will reload
 the system now!
SUCCESS: install_activate  Tue Jun 20 18:53:27 PDT 2017
Device#
```

# 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 Sun Jun 11 15:01:37 Universal 2017
[0|install_op_boot]: END SUCCESS  Sun Jun 11 15:01:44 Universal 2017
[1|install_commit]: START Mon Jun 12 07:27:31 UTC 2017
[1|install_commit(INFO, )]: Releasing transaction lock...
[1|install_commit(CONSOLE, )]: Committing PACKAGE
[remote|install_commit]: START Mon Jun 12 07:28:08 UTC 2017
[remote|install_commit(INFO, )]: Releasing transaction lock...
[remote|install_commit]: END SUCCESS  Mon Jun 12 07:28:41 UTC 2017
```

```
[1|install_commit(INFO, )]: [1 2 3]: Performing Commit
  SUCCESS: Commit finished
[1|install_commit(INFO, )]: install_commit: START Mon Jun 12 07:28:08 UTC 2017
  SUCCESS: install_commit  Mon Jun 12 07:28:41 UTC 2017
[1|install_commit(INFO, )]: Remote output from switch 2
[1|install_commit(INFO, )]: install_commit: START Mon Jun 12 07:28:12 UTC 2017
  SUCCESS: install_commit  Mon Jun 12 07:28:44 UTC 2017
[1|install_commit(INFO, )]: install_commit: START Mon Jun 12 07:28:12 UTC 2017
  SUCCESS: install_commit  Mon Jun 12 07:28:45 UTC 2017
[1|install_commit]: END SUCCESS  Mon Jun 12 07:28:47 UTC 2017
```

### 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

[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   I    16.6.2.0
IMG   C    16.6.1.0

Device#
```

### ステップ 4 show install package *filesystem: filename*

例：

```
Device# show install package flash:cat9k_iosxe.16.06.01.SPA.bin
```

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_iosxe.16.06.01.SPA.bin

Package: cat9k_iosxe.16.06.01.SPA.bin
  Size: 333806196
  Timestamp: Sun Jun 11 14:47:23 2017 UTC
  Canonical path: /flash/cat9k_iosxe.16.06.01.SPA.bin

 Raw disk-file SHA1sum:
    5e9ef6ed1f7472b35eddd61df300e44b14b65ec4
  Header size:     1000 bytes
  Package type:    10002
  Package flags:   0
  Header version:  3

  Internal package information:
    Name: cc_srdriver
    BuildTime:
    ReleaseDate: Sun-27-Aug-17-09:05
```

```
    BootArchitecture: none
    RouteProcessor: cat9k
    Platform: CAT9K
    User: mcpre
    PackageName: cc_srdriver
    Build: BLD_V166_THROTTLE_LATEST_20170827_090555
    CardTypes:

  Package is not bootable.
Device#
```

### ステップ 5 show install active

例：

```
Device# show install active
```

アクティブなソフトウェア インストール パッケージに関する情報を表示します。

```
Device# show install active

[ R0 ] Active Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   C    16.6.2.0
```

### ステップ 6 show install inactive

例：

```
Device# show install inactive
```

非アクティブなパッケージに関する情報を表示します。

```
Device# show install inactive

[ R0 ] Inactive Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   I    16.7.1.0
Device#
```

### ステップ 7 show install committed

例：

```
Device# show install committed
```

コミット済みのパッケージに関する情報を表示します。

```
Device# show install committed

[ R0 ] Committed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
```

```
Type  St   Filename/Version
-------------------------------------------------------------------------------
IMG   C    16.6.1.0
Device#
```

**ステップ 8 show install uncommitted**

例：

```
Device# show install uncommitted
```

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。

```
Device# show install uncommitted

[ R0 ] Uncommitted Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
-------------------------------------------------------------------------------
Type  St   Filename/Version
-------------------------------------------------------------------------------
IMG   U    16.6.2.0
Device#
```

# 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
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方法の例を示します。

```
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<br>ブート ローダ コマンド | *Command Reference (Catalyst 9400 Series Switches)* |
| ハードウェアの設置 | *Cisco Catalyst 9400* シリーズ スイッチ ハードウェア設置ガイド<br>. |

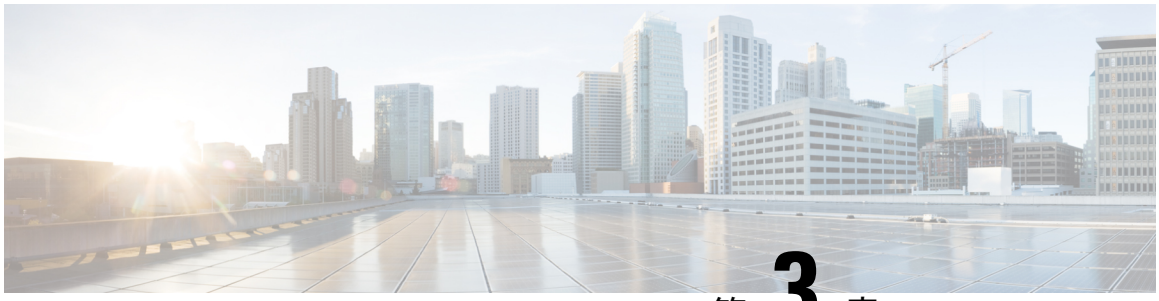
# デバイスセットアップ設定の実行に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

**表 *6* : デバイスセットアップ設定の実行に関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| デバイスのセットアップ設定の実行 | Cisco IOS XE Everest 16.6.1 | IP アドレス割り当てと Dynamic Host Configuration Protocol（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。 |



第 3 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 （89 ページ）
- スマートライセンシングの概要 （89 ページ）
- CSSM への接続 （91 ページ）
- CSSM への既存のライセンスのリンク （92 ページ）
- CSSM への接続の設定とライセンスレベルの設定 （92 ページ）
- CSSM でのデバイスの登録 （103 ページ）
- スマート ライセンスの設定のモニタリング （109 ページ）
- スマートライセンシングの設定例 （110 ページ）
- その他の参考資料 （116 ページ）
- スマートライセンスの機能の履歴 （117 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

スマートライセンスは、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。スマート ライセンスでは、時間のかかる手動のライセンス タスクを自動化することも可能です。スマートライセンスは、次の方法でタスクを簡素化するのに役立ちます。

- スマートライセンスには、すべての Cisco ソフトウェアライセンスを 1 つの一元化された Web サイトで管理できる集中型ポータルである Cisco Smart Software Manager（CSSM）が用意されています。
- ライセンス権限の有効化を自動的に追跡できます。また、すべてのノードにライセンス ファイルをインストールする必要はありません。組織構造に合わせたライセンスプール（バーチャルアカウント）を作成できます。
- スマートライセンスでは、ポータルを使用することで、購入したライセンスとネットワークに展開された製品を統合して表示できます。このデータを使用すると、購入の意思決定を実際の使用状況に基づいてより適切に行うことができます。

スマート ソフトウェア ライセンシングの概要については、「Smart Licensing」を参照してください。

# CSSM の概要

CSSM を使用すると、一元化されたポータルからすべてのシスコ スマート ソフトウェア ライセンスを管理できます。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。

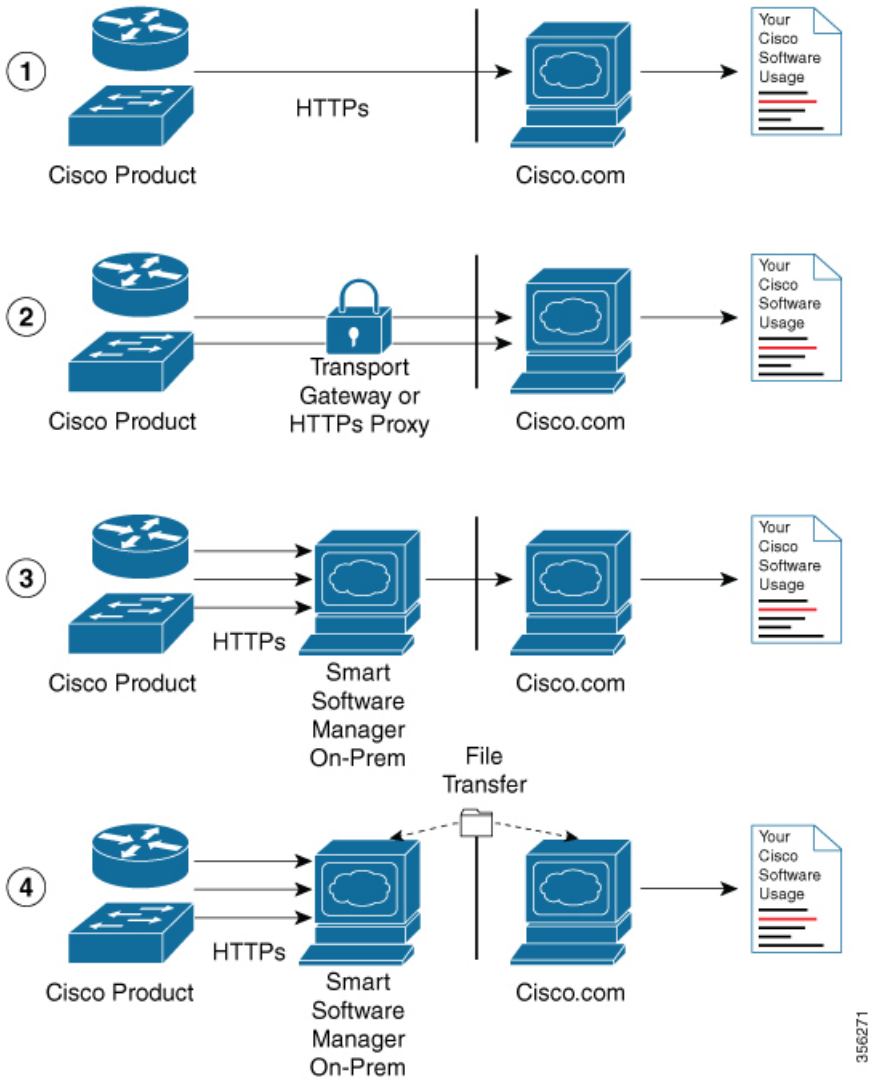
- バーチャルアカウントを作成、管理、または表示する。
- 製品インスタンスの登録トークンを作成および管理する。
- バーチャルアカウント間または表示ライセンス間でライセンスを転送する。
- 製品インスタンスを転送、削除、または表示する。
- バーチャルアカウントに関するレポートを実行する。
- 電子メール通知の設定を変更する。
- 仮想アカウント情報を表示する。

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 *5 : [Connection]* のオプション**




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 への既存のライセンスのリンク

次のセクションは、Cisco スマートアカウントを使用しないで購入したライセンスに必要です。これらのライセンスは、Cisco IOS XE Fuji 16.9.1 へのアップグレード後に CSSM で使用できなくなります。次の電子メールテンプレートを使用して、Cisco Global Licensing Operations（GLO）チームにお問い合わせください。テンプレートに適切な情報を入力し、既存のライセンスを CSSM の Cisco スマートアカウントにリンクするように要求します。

**電子メールテンプレート**：

宛先：licensing@cisco.com

件名：Request for Linking Existing Licenses to Cisco Smart Account

**電子メールの本文**：

Cisco.com ID: #####

Smart virtual account name: #####

Smart account domain ID (domain in the form of "xyz.com"): #####

List of UDIs:

List of licenses with count:

Proof of purchase（このメールと一緒に購入証明書を添付してください）

# 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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*]<br>例：<br>`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*]<br>例：<br>`Device(config)# ip name-server vrf Mgmt-vrf 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230` | （任意）VRF インターフェイスで DNS を設定します。<br>（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*<br>例：<br>`Device(config)# ip domain lookup source-interface Vlan100` | （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**<br>例：<br>`Device(config)# ip domain name example.com` | ドメイン名を設定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 7 | **ip host tools.cisco.com** *ip-address*<br>例：<br>`Device(config)# ip host tools.cisco.com 209.165.201.30` | （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*<br>例：<br>`Device(config)# interface Vlan100`<br>`Device(config-if)# ip address 192.0.2.10 255.255.255.0`<br>`Device(config-if)# exit` | 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**]<br>例：<br>`Device(config)# ntp server 198.51.100.100 version 2 prefer` | 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。<br>（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*<br>例：<br>`Device(config)# interface GigabitEthernet1/0/1`<br>`Device(config-if)# switchport access vlan 100`<br>`Device(config-if)# switchport mode access`<br>`Device(config-if)# exit`<br>`Device(config)#` | （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、非ト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。<br>（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*<br>例：<br>`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。<br>（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**<br>例：<br>`Device(config)# license smart transport callhome` | 転送モードを Call Home として有効にします。<br>（注） **license smart transport callhome** コマンドは必須です。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 13 | **ip http client source-interface** *interface-type interface-number*<br>例：<br>`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。<br>（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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 4 | **no http secure server-identity-check**<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`Device(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService` | CSSM に接続します。 |
| ステップ 9 | **active**<br>例：<br>`Device(config-call-home-profile)# active` | 宛先プロファイルをイネーブルにします。 |
| ステップ 10 | **no destination transport-method email**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 13 | **service call-home**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 4 | **contact-email-address** *email-address*<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`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。 |
| ステップ 10 | **reporting smart-licensing-data**<br>例：<br>`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 11 | **destination transport-method http**<br>例：<br>`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 12 | **destination address http** *url*<br>例： | CSSM に接続します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService` | |
| ステップ 13 | **active**<br>例：<br>`Device(config-call-home-profile)# active` | 宛先プロファイルをイネーブルにします。 |
| ステップ 14 | **exit**<br>例：<br>`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 15 | **exit**<br>例：<br>`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 16 | **service call-home**<br>例：<br>`Device(config)# service call-home` | 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*<br>例：<br>`Device(config)# ip http client proxy-server 198.51.100.10 port 3128` | Call Home 機能をイネーブルにします。 |
| ステップ 18 | **exit**<br>例：<br>`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 19 | **copy running-config startup-config**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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**<br>例：<br>`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 | **profile CiscoTAC-1**<br>例：<br>`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 5 | **no destination address http** *url*<br>例：<br>`Device(config-call-home-profile)# no destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService` | デフォルトの宛先アドレスを無効にします。 |
| ステップ 6 | **no http secure server-identity-check**<br>例：<br>`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 7 | **profile** *name*<br>例：<br>`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。 |
| ステップ 8 | **reporting smart-licensing-data**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 9 | **destination transport-method http**<br>例：<br>`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 10 | **destination address http** *url*<br>例：<br>`Device(config-call-home-profile)# destination address http https://209.165.201.15:443/Transportgateway/services/DeviceRequestHandler`<br>または<br>`Device(config-call-home-profile)# destination address http http://209.165.201.15:80/Transportgateway/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。<br>注<br>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 11 | **destination preferred-msg-format** {**long-text** \| **short-text** \| **xml**}<br>例：<br>`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 12 | **active**<br>例：<br>`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 13 | **exit**<br>例：<br>`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 14 | **exit**<br>例：<br>`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 15 | **ip http client source-interface** *interface-type interface-number*<br>例：<br>`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。<br>（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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 16 | **crypto pki trustpoint** *name*<br>例：<br>`Device(config)# crypto pki trustpoint SLA-TrustPoint` | （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 17 | **revocation-check none**<br>例：<br>`Device(ca-trustpoint)# revocation-check none` | （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 18 | **end**<br>例：<br>`Device(ca-trustpoint)# end` | （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 19 | **copy running-config startup-config**<br>例：<br>`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。 |

# 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`Device(config)# exit` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **write memory**<br>例：<br>`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**<br>例：<br>`Device# show version`<br>`----------`<br>`Technology-package Current Type`<br>`Technology-package Next reboot`<br>`----------`<br>`network-essentials Smart License`<br>`network-essentials None`<br>`Subscription Smart License None` | ライセンスレベルの情報を表示します。 |
| ステップ 7 | **reload**<br>例：<br>`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。

（注） Cisco IOS XE Fuji 16.9.1 から以前のリリースにデバイスをダウングレードすると、スマートライセンスは従来のライセンスに移行されます。デバイス上のすべてのスマートライセンス情報が削除されます。デバイスを Cisco IOS XE Fuji 16.9.1 に再びアップグレードする必要がある場合、デバイスが CSSM に再登録されるまで、ライセンスステータスは評価モードのままです。

# 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。

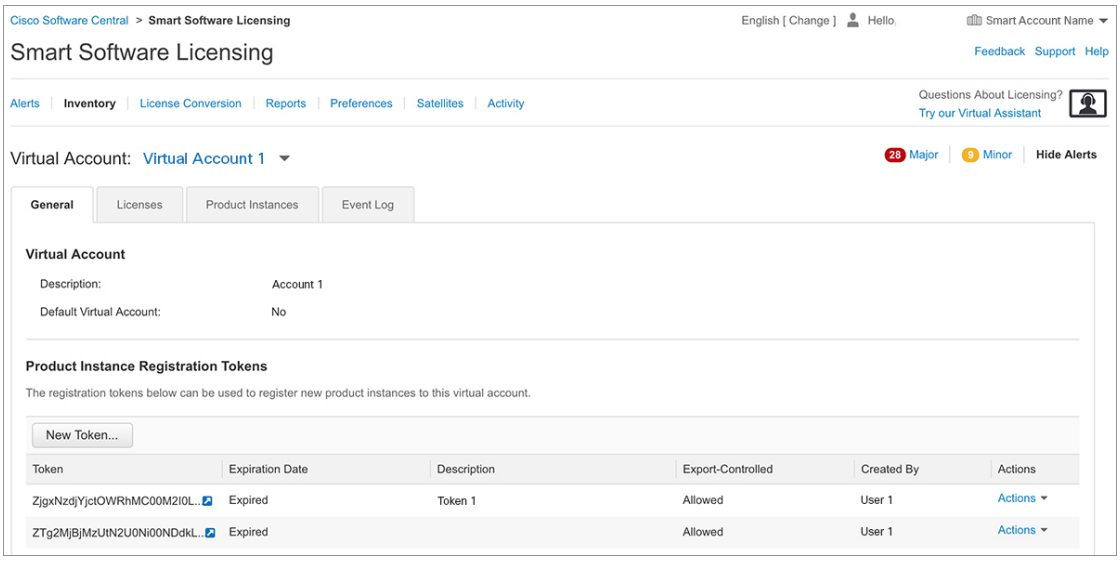
シスコから提供されたユーザ名とパスワードを使用してポータルにログインする必要があります。

**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。

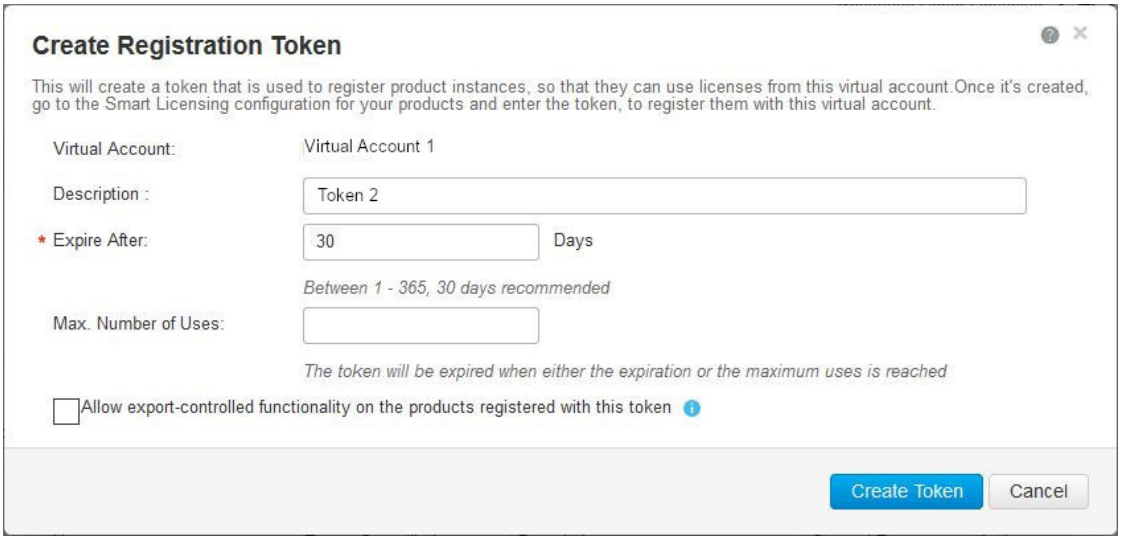



[Create Registration Token] ウィンドウが表示されます。

**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。

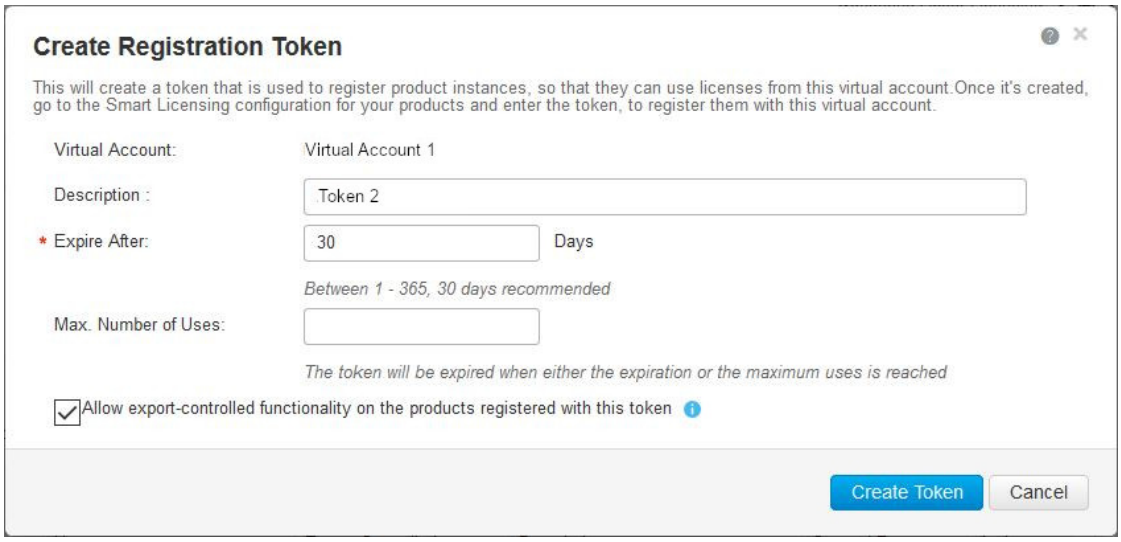



**ステップ 9** [Allow export-controlled functionality on the products registered with this token] チェックボックスをオンにします。

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。




# 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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*<br>例：<br>`Device# license smart register idtoken`<br>`[redacted]` | 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**<br>例：<br>`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: Not Allowed
  Initial Registration: SUCCEEDED on Jul 27 08:38:44 2018 EDT
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 23 08:38:44 2019 EDT
  Registration Expires: Jul 27 08:32:51 2019 EDT

License Authorization:
  Status: AUTHORIZED on Jul 27 08:38:49 2018 EDT
  Last Communication Attempt: SUCCEEDED on Jul 27 08:38:49 2018 EDT
  Next Communication Attempt: Aug 26 08:38:49 2018 EDT
  Communication Deadline: Oct 25 08:32:57 2018 EDT

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
==============

C9400 DNA Advantage (dna_advantage-C9400):
  Description: C9400 DNA Advantage
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

C9400 Network Advantage (advantagek9-C9400):
  Description: C9400 Network Advantage
  Count: 2
  Version: 1.0
  Status: AUTHORIZED

Product Information
===================
UDI: PID:C9410R,SN:FXS2132Q0GU

HA UDI List:
    Active:PID:C9410R,SN:FXS2132Q0GU
    Standby:PID:C9410R,SN:FXS2132Q0GU

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:(rel5)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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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 7: 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 |
|---|---|
| **show license status** | 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。<br>• 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。<br>• 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。<br>• 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。<br>• 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。<br>• 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。<br>• 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。<br>• 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
==============

C9400 DNA Advantage (dna_advantage-C9400):
  Description: C9400 DNA Advantage
  Count: 1
  Version: 1.0
  Status: EVAL MODE

C9400 Network Advantage (advantagek9-C9400):
  Description: C9400 Network Advantage
  Count: 2
  Version: 1.0
  Status: EVAL MODE

Product Information
===================
UDI: PID:C9410R,SN:FXS2132Q0GU

HA UDI List:
    Active:PID:C9410R,SN:FXS2132Q0GU
    Standby:PID:C9410R,SN:FXS2132Q0GU
```

```
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:(rel5)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: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 38 seconds

C9400 DNA Advantage (dna_advantage-C9400):
  Description: C9400 DNA Advantage
  Count: 1
  Version: 1.0
  Status: EVAL MODE

C9400 Network Advantage (advantagek9-C9400):
  Description: C9400 Network Advantage
  Count: 2
  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
  -----------------------------------------------------------------------------
                          (dna_advantage-C9400)             1 EVAL MODE
                          (advantagek9-C9400)               2 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

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
Tl4UytrNXBzbEs1ck8veUtWaG5abnZJOFdDa1FwbVRa%0AblRMbz0%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: Not Allowed
  Initial Registration: SUCCEEDED on Jul 27 08:38:44 2018 EDT
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 23 08:38:44 2019 EDT
  Registration Expires: Jul 27 08:32:51 2019 EDT

License Authorization:
  Status: AUTHORIZED on Jul 27 08:38:49 2018 EDT
  Last Communication Attempt: SUCCEEDED on Jul 27 08:38:49 2018 EDT
  Next Communication Attempt: Aug 26 08:38:49 2018 EDT
  Communication Deadline: Oct 25 08:32:57 2018 EDT

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
==============

C9400 DNA Advantage (dna_advantage-C9400):
  Description: C9400 DNA Advantage
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

C9400 Network Advantage (advantagek9-C9400):
  Description: C9400 Network Advantage
  Count: 2
  Version: 1.0
  Status: AUTHORIZED

Product Information
===================
UDI: PID:C9410R,SN:FXS2132Q0GU

HA UDI List:
    Active:PID:C9410R,SN:FXS2132Q0GU
    Standby:PID:C9410R,SN:FXS2132Q0GU

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:(rel5)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 27 08:38:49 2018 EDT

C9400 DNA Advantage (dna_advantage-C9400):
  Description: C9400 DNA Advantage
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

C9400 Network Advantage (advantagek9-C9400):
  Description: C9400 Network Advantage
  Count: 2
  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: Not Allowed
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 23 08:38:43 2019 EDT

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Aug 26 08:38:48 2018 EDT

License Usage:
  License                 Entitlement tag               Count Status
  -----------------------------------------------------------------------------
  C9400 DNA Advantage     (dna_advantage-C9400)             1 AUTHORIZED
  C9400 Network Advantage (advantagek9-C9400)               2 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: Not Allowed
  Initial Registration: SUCCEEDED on Jul 27 08:38:44 2018 EDT
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 23 08:38:44 2019 EDT
  Registration Expires: Jul 27 08:32:51 2019 EDT

License Authorization:
  Status: AUTHORIZED on Jul 27 08:38:49 2018 EDT
  Last Communication Attempt: SUCCEEDED on Jul 27 08:38:49 2018 EDT
  Next Communication Attempt: Aug 26 08:38:49 2018 EDT
  Communication Deadline: Oct 25 08:32:57 2018 EDT
```

# 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 |
|---|---|
| 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。<br>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。<br>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。 | http://www.cisco.com/support |

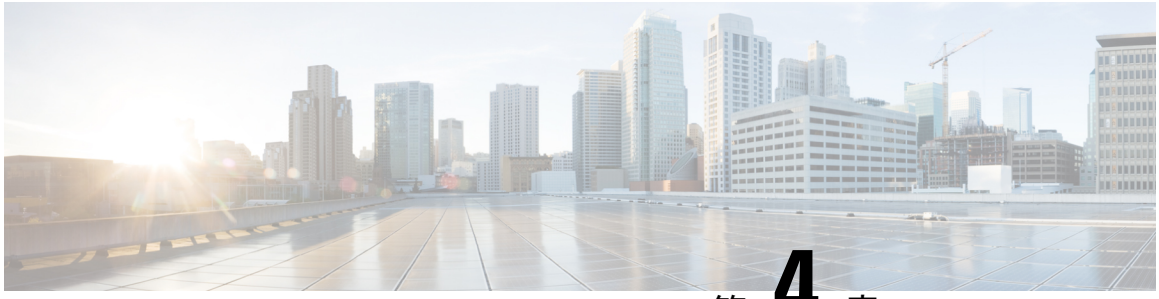
# スマートライセンスの機能の履歴

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 16.9.1 | スマートライセンス | 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。<br>このリリース以降、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。<br>Cisco IOS XE Fuji 16.9.1 以降では、使用権（RTU）ライセンスモードが廃止され、関連する CLI の **license right-to-use** コマンドも使用できなくなりました。 |

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 4 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 （119 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 （120 ページ）
- 有線 Application Visibility and Control の制限 （121 ページ）
- Application Visibility and Control の設定方法 （123 ページ）
- Application Visibility and Control のモニタリング （151 ページ）
- 例：Application Visibility and Control の設定 （152 ページ）
- 基本的なトラブルシューティング：質問と回答 （163 ページ）
- Application Visibility and Control に関する追加情報 （165 ページ）
- 有線ネットワークでの Application Visibility and Control の機能履歴と情報 （165 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 |
|---|---|---|
| **match protocol** *protocol name* | `class-map match-any NBAR-VOICE`<br>`match protocol ms-lync-audio` | 入力と出力の両方 |
| 組み合わせフィルタ | `class-map match-any NBAR-VOICE`<br>`match protocol ms-lync-audio`<br>`match dscp ef` | 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 | `policy-map MARKING-IN`<br>`class NBAR-MM_CONFERENCING`<br>`set dscp af41` | 入力および出力 |
| ベーシック ポリシー | `policy-map POLICING-IN`<br>`class NBAR-MM_CONFERENCING`<br>`police cir 600000`<br>`set dscp af41` | 入力および出力 |
| ベーシック セットおよびポリシー | `policy-map webex-policy`<br>` class webex-class`<br>` set dscp ef`<br>` police 5000000` | 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 | policy-map webex-policy<br>class webex-class<br>set dscp af31<br>police 4000000<br>class class-webex-category<br>set dscp ef<br>police 6000000<br>class class-default<br>set dscp <> | 入力および出力 |
| 階層型ポリシー | policy-map webex-policy<br>class webex-class<br>police 5000000<br>service-policy client-in-police-only<br><br>policy-map client-in-police-only<br>class webex-class<br>police 100000<br>class class-webex-category<br>set dscp ef<br>police 200000 | 入力および出力 |
| 階層型セットおよびポリシー | policy-map webex-policy<br>class class-default<br>police 1500000<br>service policy client-up-child<br>policy-map client-up-child<br>class webex-class<br>police 100000<br>set dscp ef<br>class class-webex-category<br>police 200000<br>set dscp af31 | |

# 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。
- Cisco IOS XE Fuji 16.9.1 以降では、それぞれ異なる事前定義済みレコードを持つ最大 2 つの有線 AVC モニタをインターフェイスに同時に接続できます。
- Cisco IOS XE Fuji 16.9.1 では、既存の 2 つの従来のフローレコードに加えて、2 つの新しい方向性フローレコード（入力と出力）が導入されました。

- 接続は、物理レイヤ 2 およびレイヤ 3 ポートでのみ行う必要があります。これらのポートをポートチャネルの一部にすることはできません。アップリンクへの接続はサポートされていません。
- パフォーマンス：各スイッチメンバは、50% 未満の CPU 使用率で、1 秒あたり 2000 の接続（CPS）を処理できます。
- 拡張性：48 個のアクセスポートごとに最大 20,000 の双方向フローと、24 個のアクセスポートごとに 10,000 の双方向フローを処理できます。（アクセスポートごとに～200 フロー）。
- 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）。
- Cisco IOS XE 16.12.1 リリース以降、新しいフローレコード（DNS フローレコード）が追加されました。DNS フローレコードは 5 タプルレコードに似ており、DNS ドメイン名フィールドが含まれています。DNS 関連のフィールドのみを考慮します。このレコードには、照合フィールドとしてのインターフェイスフィールドがないため、すべてのインターフェイスからの情報が同じレコードに集約されます。

# 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（124 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（125 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 （127 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 （128 ページ） を参照してください。

- フローエクスポータを作成してフロー レコードをエクスポートします。フローエクスポータの作成 （142 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 エクスポータに基づいて、フロー モニタを作成します。フロー モニタの作成 （143 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 （145 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**<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`Device(config-if)# ip nbar protocol-discovery` | NBAR2 エンジンを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**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`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*<br>例：<br>`Device(config)# class-map webex-class`<br>`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**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`Device(config)# policy-map webex-policy` | 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 マップ コンフィギュレーション モードを開始します。<br>デフォルトでは、ポリシー マップは定義されていません。<br>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。ポリシングは実行されません。<br>（注） 既存のポリシーマップを削除するには、**no policy-map** *policy-map-name* 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。 |
| ステップ 3 | **class** [*class-map-name* \| **class-default**]<br>例：<br>`Device(config-pmap)# class webex-class` | 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。<br>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。<br>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 *class-map-name* にその名前を指定します。<br>**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 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | 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。<br>（注） 既存のクラスマップを削除するには、 **no class** *class-map-name* 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。 |
| ステップ 4 | **police** *rate-bps burst-byte*<br>例：<br>`Device(config-pmap-c)# police 100000 80000` | 分類したトラフィックにポリサーを定義します。<br>デフォルトでは、ポリサーは定義されていません。<br>• *rate-bps* 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<br>• *burst-byte* 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*}<br>例：<br>`Device(config-pmap-c)# set dscp 45` | 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。<br>• **dscp** *new-dscp* 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**<br>例：<br>`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**<br>例：<br>`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 | **interface** *interface-id*<br>例：<br>`Device(config)# interface Gigabitethernet 1/0/1` | 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 | **service-policy input** *policymapname*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`Device(config)# flow record fr-wdavc-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** *description*<br>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-wdavc-1` | |
| ステップ 4 | **match ipv4 version**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。<br>（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**<br>例：<br>`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**<br>例：<br>`Device(config-flow-record)# match connection server ipv4 address` | サーバ（フロー レスポンダ）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**<br>例：<br>`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 10 | **match flow observation point**<br>例：<br>`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**<br>例：<br>`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ダ）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 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | 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。<br>• 0x01 = 入力フロー<br>• 0x02 = 出力フロー<br>**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ダ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ダ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**<br>例：<br>`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ダ）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。<br>• 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<br>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**<br>例：<br>`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**<br>例：<br>`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**<br>例： | 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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config-flow-record)# collect connection client counter bytes network long` | |
| ステップ 16 | **collect connection server counter packets long**<br>例：<br>`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 server counter bytes network long**<br>例：<br>`Device(config-flow-record)# collect connection server counter bytes network long` | 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**<br>例：<br>`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 19 | **collect timestamp absolute last**<br>例：<br>`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 20 | **end**<br>例：<br>`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 21 | **show flow record**<br>例：<br>`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。 |

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**<br>例：<br>`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 | **flow record** *flow_record_name*<br>例：<br>`Device(config)# flow record fr-wdavc-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** *description*<br>例：<br>`Device(config-flow-record)# description fr-wdavc-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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。<br>（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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device(config-flow-record)# match connection server ipv4 address` | サーバ（フロー レスポンダ）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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 11 | **match flow observation point**<br>例：<br>`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 12 | **collect flow direction**<br>例：<br>`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ダ）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。<br>• 0x01 = 入力フロー<br>• 0x02 = 出力フロー<br>**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ダ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ダ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**<br>例：<br>`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ダ）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。<br>• 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<br>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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 15 | **collect connection client counter packets long**<br>例：<br>`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 16 | **collect connection client counter bytes network long**<br>例：<br>`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 17 | **collect connection server counter packets long**<br>例：<br>`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 18 | **collect connection server counter bytes network long**<br>例：<br>`Device(config-flow-record)# collect connection server counter bytes network long` | 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**<br>例：<br>`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 20 | **collect timestamp absolute last**<br>例：<br>`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 21 | **end**<br>例：<br>`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 22 | **show flow record**<br>例：<br>`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。 |

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**<br>例：<br>`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*<br>例：<br>`Device(config)# flow record fr-wdavc-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*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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 input**<br>例：<br>`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**<br>例：<br>`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。<br>（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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>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**<br>例：<br>`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。 |

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**<br>例：<br>`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*<br>例：<br>`Device(config)# flow record fr-wdavc-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** *description*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 8 | **match transport source-port**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。<br>（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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 16 | **collect timestamp absolute last**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`Device(config)# flow record fr-wdavc-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*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device(config-flow-record)# match ipv4 protocol` | IPv4 プロトコル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 6 | **match application name**<br>例：<br>`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。<br>（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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device(config-flow-record)# match connection server ipv4 address` | サーバ（フロー レスポンダ）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**<br>例：<br>`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 | **collect flow direction**<br>例：<br>`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ダ）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。<br>• 0x01 = 入力フロー<br>• 0x02 = 出力フロー<br>**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ダに設定されている場合、フローの方向 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | はフローのレスポンダ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 timestamp absolute first**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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ダ）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。<br>• 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<br>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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 18 | **collect connection server counter bytes network long**<br>例：<br>`Device(config-flow-record)# collect connection server counter bytes network long` | 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 connection client counter bytes network long**<br>例：<br>`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**<br>例：<br>`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に設定します。 |
| ステップ 21 | **end**<br>例：<br>`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 エクスポータの作成

フロー エクスポータを作成すると、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**<br>例：<br>`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*<br>例：<br>`Device(config)# flow exporter flow-exporter-1` | フロー エクスポータ 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*<br>例：<br>`Device(config-flow-exporter)# description flow-exporter-1` | （任意）フロー エクスポー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 | **destination** { *hostname* \| *ipv4-address* \| *ipv6-address* }<br>例：<br>`Device(config-flow-exporter)# destination 10.10.1.1` | エクスポータ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。 |
| ステップ 5 | **option application-table** [ **timeout** *seconds* ]<br>例：<br>`Device(config-flow-exporter)# option application-table timeout 500` | （任意）フロー エクスポータ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、フローエクスポータ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**<br>例：<br>`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**<br>例：<br>`Device# show flow exporter` | すべてのフロー エクスポータに関する情報を表示します。 |
| ステップ 8 | **show flow exporter statistics**<br>例：<br>`Device# show flow exporter statistics` | フロー エクスポータ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**<br>例：<br>`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*<br>例：<br>`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*<br>例： | （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 |
|---|---|---|
| | `Device(config-flow-monitor)# description flow-monitor-1` | |
| ステップ 4 | **record** *record-name*<br>例：<br>`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** *exporter-name*<br>例：<br>`Device(config-flow-monitor)# exporter flow-exporter-1` | 事前に作成されたエクスポータの名前を指定します。 |
| ステップ 6 | **cache** { **entries** *number-of-entries* \| **timeout** {**active** \| **inactive**} \| **type normal** }<br>例：<br>`Device(config-flow-monitor)# cache timeout active 1800`<br>例：<br>`Device(config-flow-monitor)# cache timeout inactive 200`<br>例：<br>`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。<br>• **entries** *number-of-entries*：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。<br>（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**<br>例：<br>`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**<br>例：<br>`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** *flow-monitor-name*<br>例：<br>`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** *flow-monitor-name* **statistics**<br>例：<br>`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** *flow-monitor-name* **statistics**<br>例：<br>`Device# clear flow monitor flow-monitor-1 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 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** *flow-monitor-name* **cache format table**<br>例：<br>`Device# show flow monitor flow-monitor-1 cache format table` | 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** *flow-monitor-name* **cache format record**<br>例：<br>`Device# show flow monitor flow-monitor-1 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。 |
| ステップ 14 | **show flow monitor** *flow-monitor-name* **cache format csv**<br>例：<br>`Device# show flow monitor flow-monitor-1 cache format csv` | 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**<br>例：<br>`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*<br>例：<br>`Device(config)# interface Gigabitethernet 1/0/1` | 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** *monitor-name* { **input** \| **output** }<br>例：<br>`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**<br>例：<br>`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タ 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

セレクタ 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

セレクタ 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/xe-3s/asr1000/qos-nbar-xe-3s-asr-1000-book/nbar-custapp-dns-xe.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（149 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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**]<br>例：<br>`Device(config)# ip nbar protocol-pack flash:defProtoPack`<br>例：<br>`Device(config)# default ip nbar protocol-pack` | プロトコル パックをロードします。<br>• 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。<br>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**<br>例：<br>`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**]<br>例：<br>`Device# show ip nbar protocol-pack active` | プロトコル パック情報を表示します。<br>• 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。<br>• 指定されたプロトコル パックの情報を表示するには、*protocol-pack* 引数を使用します。<br>• 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。<br>• 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 *8:* 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 |
|---|---|
| **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*] | NBAR Protocol Discovery 機能によって収集された統計情報を表示します。<br>• （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。 |
| **show policy-map interface** *interface-type interface-number* | 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。 |
| **show platform software fed active** \|**standby wdavc flows** | 指定したスイッチのすべてのフローに関する統計情報を表示します。 |

# 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

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

```
                                        3613000
3437000
```

**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 wdavc 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 wdavc 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                             443
        4294967305           4        6  layer7 ms-lync
Input     ..................
64.103.126.28             66.163.36.139                             443
        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                                5060
        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 wdavc 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 wdavc 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 server packets 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。

1. **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。

2. **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。

3. **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**：ほどんとのトラフィックが 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 はスイッチ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

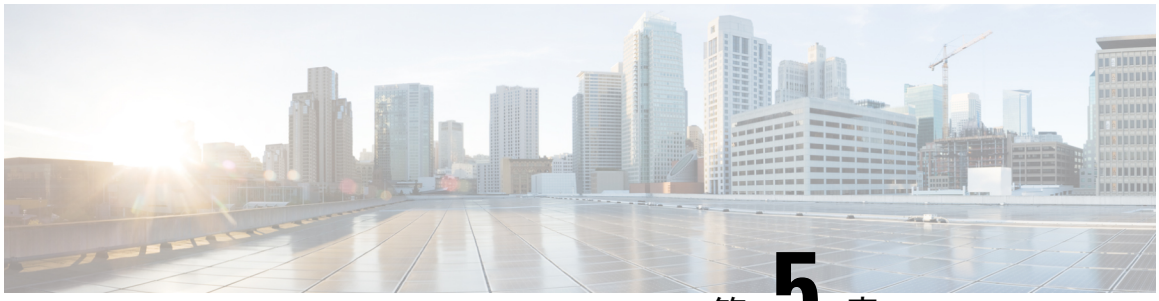
**関連資料**

| 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。 | *Command Reference (Catalyst 9400 Series Switches)* |

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

| リリース | 機能情報 |
|---|---|
| Cisco IOS XE Denali 16.3.1 | この機能が導入されました。 |
| Cisco IOS XE Denali 16.3.2 | 有線 AVC Flexible NetFlow（FNF）：この機能は、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するために、フロー レコードでアプリケーション名をキーとして使用します。 |
| Cisco IOS XE Everest 16.5.1a | 有線 AVC フローモニタ設定とフローキャッシュの内容を表示する show コマンドのサポートが導入されました。 |
| Cisco IOS XE Fuji 16.8.1a | 有線 Application Visibility and Control（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。 |

| リリース | 機能情報 |
| --- | --- |
| Cisco IOS XE Gibraltar 16.12.1 | 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 IOS XE Amsterdam 17.3.1 | 同じポートでのアプリケーションの表示およびコントロールと暗号化トラフィック分析の相互運用性のサポートが導入されました。 |



第 5 章

# 環境モニタリングおよび電源管理


- 環境モニタリングについて （167 ページ）
- 電源管理 （174 ページ）
- 電源モードと動作状態の設定例 （188 ページ）
- 環境モニタリングおよび電源管理の機能履歴と情報 （196 ページ）


## 環境モニタリングについて

シャーシコンポーネントの環境モニタリングは、コンポーネント障害の兆候を早期に警告します。この警告により、安全で信頼性の高いシステム運用を実現し、ネットワーク障害を防止できます。

ここでは、重要なシステムコンポーネントをモニタリングする方法について説明します。これにより、ハードウェア関連の問題点を特定し、速やかに対応できるようになります。

## CLI コマンドによる環境のモニタリング

システムステータス情報を表示するには、**show environment** [**all** | **counters** | **history** | **location** | **sensor** | **status** | **summary** | **table**] コマンドを使用します。次の表に、キーワードの意味を示します。

**表 *9:*** キーワードの意味

| キーワード | 目的 |
|---|---|
| **all** | すべての環境モニタ パラメータ（たとえば、電源、温度示度、電圧示度など）の詳細なリストを表示します。これはデフォルトです。 |
| **counters** | 動作カウンタを表示します。 |
| **history** | センサーの状態変化履歴を表示します。 |
| **location** | ロケーション別にセンサーを表示します。 |

| キーワード | 目的 |
|---|---|
| **sensor** | センサーのサマリーを表示します。 |
| **status** | 現場交換可能ユニット（FRU）の動作ステータスおよび電源と電源装置ファン センサーの情報を表示します。 |
| **summary** | すべての環境モニタリングセンサーのサマリーを表示します。 |
| **table** | センサーの状態テーブルを表示します。 |

# 環境状態の表示

スーパーバイザモジュールとそれらに関連付けられたラインカードは、カードごとに複数の温度センサーをサポートします。環境状態の出力には、各センサーから読み取った温度および各センサーの温度しきい値が表示されます。これらのラインカードは、警告、重大、シャットダウンの 3 つのしきい値をサポートしています。

次に、スーパーバイザモジュールの環境状態を表示する例を示します。しきい値はカッコ内に表示されています。

```
Device# show environment

Number of Critical alarms:  0
Number of Major alarms:     0
Number of Minor alarms:     0

Slot    Sensor       Current State       Reading
Threshold(Minor,Major,Critical,Shutdown)
----    ------      -------------      -------    ---------------------------------------

 R0    HotSwap: Volts   Normal           53    V DC    na
 R0    HotSwap: Power   Normal           231   Watts   na
 R0    Temp: Coretemp   Normal           46    Celsius (107,117,123,125)(Celsius)
 R0    Temp: DopplerD   Normal           55    Celsius (107,117,123,125)(Celsius)
 R0    V1: VX1          Normal           845   mV      na
 R0    V1: VX2          Normal           1499  mV      na
 R0    V1: VX3          Normal           1058  mV      na
 R0    V1: VX4          Normal           849   mV      na
 R0    V1: VX5          Normal           1517  mV      na
 R0    V1: VX6          Normal           1306  mV      na
 R0    V1: VX7          Normal           1007  mV      na
 R0    V1: VX8          Normal           1098  mV      na
 R0    V1: VX9          Normal           1205  mV      na
 R0    V1: VX10         Normal           1704  mV      na
 R0    V1: VX11         Normal           1208  mV      na
 R0    V1: VX12         Normal           1804  mV      na
 R0    V1: VX13         Normal           2518  mV      na
 R0    V1: VX14         Normal           3288  mV      na
 R0    Temp:   outlet   Normal           39    Celsius (55 ,65 ,75 ,100)(Celsius)
 R0    Temp:    inlet   Normal           35    Celsius (45 ,55 ,65 ,72 )(Celsius)
```

次に、スーパーバイザ モジュールの LED ステータスを表示する例を示します。

```
Device# show hardware led

Current Mode: STATUS

SWITCH: C9407R
SYSTEM: AMBER

SUPERVISOR: ACTIVE
STATUS: (10) Te3/0/1:BLACK Te3/0/2:BLACK Te3/0/3:BLACK Te3/0/4:BLACK Te3/0/5:BLACK
Te3/0/6:BLACK Te3/0/7:BLACK Te3/0/8:BLACK Fo3/0/9:BLACK Fo3/0/10:BLACK
BEACON: BLACK

RJ45 CONSOLE: GREEN
FANTRAY STATUS: GREEN
FANTRAY BEACON: BLACK
POWER-SUPPLY 1 BEACON: BLACK
POWER-SUPPLY 3 BEACON: BLACK
```

# オンボード障害ロギング（OBFL）情報の表示

OBFL機能は、スイッチに取り付けられているラインカードやスーパーバイザモジュールの問題の診断に役立つ動作温度、ハードウェア稼働時間、割り込み、およびその他の重要なイベントとメッセージを記録します。データのログは、不揮発性メモリに保存されるファイルに作成されます。オンボードハードウェアが起動すると、監視されている各領域で最初のレコードが作成され、後続のレコードの基準値となります。OBFL 機能は、継続的なレコードの収集と古い（履歴）レコードのアーカイブで循環更新スキームを提供し、システムに関する正確なデータを保証します。データは、測定と継続ファイルのサンプルのスナップショットを表示する継続情報の形式、または収集したデータに関する詳細を提供する要約情報の形式で記録されます。データを表示するには、**show logging onboard** コマンドを使用します。履歴データが利用できない場合は、「No historical data to display」というメッセージが表示されます。

```
Device# show logging onboard RP active voltage detail

--------------------------------------------------------------------------------
VOLTAGE SUMMARY INFORMATION
--------------------------------------------------------------------------------
Number of sensors          : 16
--------------------------------------------------------------------------------
Sensor                     ID         Normal Range          Maximum Sensor Value
--------------------------------------------------------------------------------
SYSTEM Rail-5.0            0          0 - 5                 0
SYSTEM Rail-0.9PEX         1          0 - 5                 1
SYSTEM Rail-0.9            2          0 - 5                 1
SYSTEM Rail-1.8            3          0 - 5                 0
SYSTEM Rail-3.3            4          0 - 5                 1
SYSTEM Rail-2.5            5          0 - 5                 1
SYSTEM Rail-1.5CPU         6          0 - 5                 1
SYSTEM Rail-1.5            7          0 - 5                 1
SYSTEM Rail-1.2            8          0 - 5                 1
SYSTEM Rail-1.1            9          0 - 5                 1
SYSTEM Rail-1.0            10         0 - 5                 1
SYSTEM Rail-0.9CPU         11         0 - 5                 1
SYSTEM Rail-0.85           12         0 - 5                 2
SYSTEM Rail-0.85DOPv
     13         0 - 5                3
SYSTEM Rail-0.85DOPv^N     14         0 - 5                 5
SYSTEM Rail-0.85DOPv^O     15         0 - 5                 0
```

```
------------------------------------------------------------------------------
Sensor Value
Total Time of each Sensor
------------------------------------------------------------------------------

------------------------------------------------------------------------------
No historical data

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


------------------------------------------------------------------------------
VOLTAGE CONTINUOUS INFORMATION
------------------------------------------------------------------------------
Sensor                    ID
------------------------------------------------------------------------------

SYSTEM Rail-5.0           0
SYSTEM Rail-0.9PEX        1
SYSTEM Rail-0.9           2
SYSTEM Rail-1.8           3
SYSTEM Rail-3.3           4
SYSTEM Rail-2.5           5
SYSTEM Rail-1.5CPU        6
SYSTEM Rail-1.5           7
SYSTEM Rail-1.2           8
SYSTEM Rail-1.1           9
SYSTEM Rail-1.0           10
SYSTEM Rail-0.9CPU        11
SYSTEM Rail-0.85          12
SYSTEM Rail-0.85DOPv
     13
SYSTEM Rail-0.85DOPv^N     14
SYSTEM Rail-0.85DOPv^O     15
------------------------------------------------------------------------------
       Time Stamp   | Sensor Voltage 0V
MM/DD/YYYY HH:MM:SS | Sensor Value
------------------------------------------------------------------------------
05/06/2015 16:42:51  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0
05/06/2015 18:24:24  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0
05/10/2015 17:53:42  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0
08/30/2017 16:14:40  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0
08/30/2017 23:34:24  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0
08/31/2017 22:16:23  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0
09/01/2017 00:57:15  0  1  1  0  1  1  1  1  1  1  1  1  2  3  5  0

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

# 緊急処理

シャーシは、1 つのカードの電源を切って、ラインカードの過熱状態に対して詳細な応答を提供できます。ただし、シャーシは、スーパーバイザモジュール自体の温度が重大しきい値を超過すると安全に動作させることができません。スーパーバイザモジュールはシャーシの電源をオフにしてそのモジュール自体の過熱を防止します。このような場合、電源装置の電源オン/オフ スイッチまたは電源装置の AC または DC 入力電源をオフにしてから再びオンにして、スイッチを回復できます。

重大およびシャットダウン温度という緊急状態により同じ処理が行われます。次の表に、温度による緊急状態を示しますが、重大とシャットダウンの緊急状態は区別しません。

表 *10* : 緊急状態およびアクション

| | |
|---|---|
| ケース 1. ファンの完全な障害による緊急状態 | syslog メッセージが表示され、シャーシはシャットダウンします。 |
| ケース2. ラインカードの温度による緊急状態。 | ラインカードの電源を切ります。 |
| ケース 3. 電源の温度による緊急状態。重大またはシャットダウン アラームしきい値を超えると、すべての電源がシャットダウンします。 | デバイスの電源を再投入して、電源シャットダウンから回復します。 |
| ケース 4. アクティブなスーパーバイザ モジュールの温度による緊急状態。 | シャーシの電源を切断します。 |

# システム アラーム

どのシステムにも、メジャーとマイナーの 2 種類のアラームがあります。メジャー アラームは、システムのシャットダウンにつながる可能性のある重大な問題を示します。マイナー アラームは情報で、対処しないと重大な問題となる可能性がある点について通知します。

次の表に、使用可能な環境アラームを示します。

表 *11* : 発生する可能性のある環境アラーム

| | |
|---|---|
| 警告しきい値を超える温度センサー | マイナー |
| 重大しきい値を超える温度センサー | メジャー |
| シャットダウンしきい値を超える温度センサー | メジャー |
| ファンの部分的な障害 | マイナー |
| ファンの完全な障害<br>（注） ファンの完全な障害アラームでシステムがシャットダウンすることはありません。 | メジャー |

ファン障害アラームは、ファン障害状態が検知されると発生し、ファン障害状態が解消すると取り消されます。温度がしきい値温度に到達するとすぐに温度アラームが発行されます。スーパーバイザ モジュールの LED は、アラームが発生したかどうかを示します。

システムによってメジャーアラームが発生するとタイマーが始動しますが、その期間はアラームによって異なります。タイマーが切れるまでにアラームが取り消されない場合は、過熱による影響が生じないようにするためにシステムは緊急処理を行います。タイマー値および緊急処理は、スーパーバイザ モジュールのタイプによって異なります。

（注） スーパーバイザ モジュールのシステム LED の起動動作など、LED の詳細については、『*Hardware Installation Guide*』を参照してください。

**表 *12*：スーパーバイザ モジュールのアラーム**

| イベント | アラームの種類 | スーパーバイザ LED の色 | 説明およびアクション |
|---|---|---|---|
| カードの温度が重大しきい値を超過 | メジャー | 赤 | アラームが発生すると、Syslog メッセージが表示されます。 |
| カードの温度がシャットダウンしきい値を超過 | メジャー | 赤 | アラームが発生すると、Syslog メッセージが表示されます。 |
| シャーシの温度が警告しきい値を超過 | マイナー | オレンジ | アラームが発生すると、Syslog メッセージが表示されます。 |
| シャーシ ファン トレイの部分的な障害 | マイナー | オレンジ | アラームが発生すると、Syslog メッセージが表示されます。 |
| シャーシ ファン トレイの完全な障害 | メジャー | 赤 | アラームが発生すると、Syslog メッセージが表示されます。 |

# サーマルシャットダウンの無効化

Cisco IOS XE Gibraltar 16.11.1 リリース以降、システムのサーマルシャットダウンを手動で無効にするオプションが導入されました。これにより、温度が重大およびシャットダウン温度を超えた場合でも、シャーシの電源をオフにするスーパバイザエンジンの動作がトリガーされなくなります。サーマルシャットダウンのディセーブル機能を使用すると、システムがすでにシャットダウン状態になっている場合でも、システムのサーマル シャットダウン プロセスをバイパスできます。

サーマルシャットダウンのディセーブルオプションを設定するには、**thermal shutdown disable** コマンドを使用します。更新された設定をスタートアップ コンフィギュレーションに保存すると、システムは、次の電源再投入またはシステムのリロード後に、サーマルシャットダウンを無効にして起動します。

システムのサーマルシャットダウン機能を再度有効にするには、**no thermal shutdown disable** コマンドを使用します。

サーマルシャットダウンのディセーブル機能は、高可用性を完全にはサポートしていません。次に、サポートされないケースを示します。

- アクティブなスーパバイザエンジンでのみサーマルシャットダウンがディセーブルになっている場合、システムがシャットダウン状態になったときに2番目のスーパバイザエンジンを起動すると、システムはシャットダウンしたままになります。
- 両方のスーパバイザエンジンでサーマルシャットダウンがディセーブルになっていて、システムがシャットダウン状態になった場合、サーマルシャットダウン機能を再度有効にすると、システムはシャットダウンしません。変更を有効にするには、設定の変更をスタートアップ コンフィギュレーションに保存し、スイッチをリロードする必要があります。

次の表に、スーパバイザエンジンの状態と、これらの各状態のサーマル シャットダウン ディセーブル設定サポートの可能な組み合わせを示します。

**表 *13*：**

| アクティブ スーパバイザ | スタンバイ スーパバイザ | サーマルシャットダウンのディセーブル設定のサポート |
|---|---|---|
| シャットダウン状態 | シャットダウン状態 | • 起動時はサポートされません。<br>• 実行時にサポートされます。 |
| 正常状態 | シャットダウン状態 | • 起動時はサポートされません。<br>• 実行時にサポートされます。 |
| シャットダウン状態 | 正常状態 | • 起動時にサポートされます。<br>• 実行時にサポートされます。 |
| 正常状態 | 正常状態 | • 起動時にサポートされます。<br>• 実行時にサポートされます。 |

# 電源管理

ここでは、Cisco Catalyst 9400 シリーズ スイッチの電源管理機能と、制御および設定可能な電源管理の側面について説明します。設置、取り外し、および電源仕様を含むハードウェアの詳細については、『*Cisco Catalyst 9400 Series Switches Hardware Installation Guide*』を参照してください。

## 電源モード

Cisco Catalyst 9400 シリーズ スイッチは、電源に関して複合コンフィギュレーション モードと冗長コンフィギュレーション モードを提供します。

### 複合モード

これはデフォルトの電源モードです。

システムは 1 ～ 8 個の電源で稼働します。使用可能なすべての電源がアクティブになって電力を共有し、最大 100% のキャパシティで稼働できます。

複合モードで使用可能な電力は、個々の電源の合計です。

### 冗長モード

冗長コンフィギュレーションでは、特定の電源モジュールはアクティブまたはスタンバイモードのいずれかで、必要なときにアクティブに切り替えます。

n+1 または n+n 冗長モードを構成できます。

- n+1 冗長モード：n はアクティブ電源モジュールの数です（n は 1 ～ 7 個の電源モジュールです）。+1 は冗長性のために確保されている電源モジュールです。

  デフォルトのスタンバイ電源スロットは PS8 です。**power redundancy-mode redundant n+1** *standby-PSslot* コマンドを入力して、スタンバイスロットを指定します。

- n+n 冗長モード：n 個の電源がアクティブであり、n 個の電源モジュールがスタンバイとして設定されます。

  このモードのデフォルトのスタンバイスロットは PS5 ー PS8 です。**power redundancy-mode redundant n+n** *standby-PSslots* コマンドを入力して、スタンバイスロットを指定します。

現在設定されている電源モードに関する詳細情報を表示するには、特権 EXEC モードで **show power detail** コマンドを入力します。

## 動作状態

動作状態とは、すべてのアクティブな電源モジュールに障害が発生した状況にシステムが対応できることを指します。システムは、次の要因に応じて、シャーシの動作状態を完全保護、通常保護、または複合と見なします。

- アクティブな合計出力電力。シャーシ内のすべてのアクティブな電源モジュールから使用可能な合計出力電力です。
- 必要なバジェット電力。スーパバイザモジュール、スイッチングモジュール（ラインカード）、およびファントレイをシャーシで動作させるためだけに必要な電力です。

  **show** コマンド出力（**show power**、**show power detail**）では、これは `System Power` として表示されます。
- 合計スタンバイ出力電力。スタンバイとして設定されているシャーシ内のすべての電源モジュールから使用可能な合計出力電力です。

n+1 モードでも n+n モードでも、次の条件がすべて満たされると、シャーシは完全保護状態と見なされます。

- アクティブな合計出力電力が必要なバジェット電力より大きい
- 合計スタンバイ出力電力がアクティブな合計出力電力以上である

n+1 モードでも n+n モードでも、次の条件がすべて満たされると、シャーシは通常保護状態と見なされます。

- アクティブな合計出力電力が必要なバジェット電力より大きい
- 合計スタンバイ出力電力がアクティブな合計出力電力より小さい

次の条件が発生すると、システムは複合状態で動作します（冗長構成が拒否されます）。

- アクティブな合計出力電力が必要なバジェット電力より小さい
- スタンバイ電源モジュールが設定されていない、または取り付けられていない

### 例：動作状態

**show power** コマンドの次の出力例は、完全保護状態の電源構成を示しています。

ここでは、スロット 1 および 2 の電源モジュールがアクティブで、電力を共有しています。スロット 7 および 8 の電源モジュールはスタンバイです。必要なバジェット電力は 2115W、インラインパワーには 3185W 必要です。以下の条件により、スイッチは完全保護状態です。

- アクティブな合計出力電力（`PS1 Capacity + PS2 Capacity`）が、必要なバジェット電力（`System Power - Maximim Used 2115`）より大きい
- 合計スタンバイ出力電力（`PS7 Capacity + PS8 Capacity`）が、アクティブな合計出力電力（`PS1 Capacity + PS2 Capacity`）と等しい

```
Device# show power
Power         Model No              Type  Capacity   Status        1    2    3    4
Supply
------        --------------------  ----  --------  ------------  ------------------------
PS1           C9400-PWR-3200AC       AC   3200 W    active        good good good good
PS2           C9400-PWR-3200AC       AC   3200 W    active        good good good good
PS7           C9400-PWR-3200AC       AC   3200 W    standby       n.a. n.a. n.a. n.a.
```

```
PS8          C9400-PWR-3200AC      AC   3200 W    standby       n.a. n.a. n.a. n.a.

PS Current Configuration Mode: N+N redundant
PS Current Operating State: Full protected

Power supplies currently active: 2
Power supplies currently available: 3
Power Summary  Maximum
(in Watts)     Used     Available
-------------  ------   --------
System Power   2115     2115
Inline Power   3185     4285
-------------  ------   ---------
Total          5300     6400
Automatic Linecard Shutdown: Enabled
Power Budget Mode          : Dual Sup

<output truncated>
```

動作状態に関する情報は、**show power detail** コマンドの出力にも表示されます。

# show power detail

**show power detail** コマンドは、**show power** および **show power module** 特権 EXEC コマンドの出力を含みます。

次に、各例で異なるタイプの電源モジュールを使用した**show power detail** コマンドの出力例を示します。すべての例で、電源モードと動作状態は `combined` です。

```
Device# show power detail
Power                                                          Fan States
Supply  Model No              Type  Capacity  Status         1     2     3     4
------  --------------------  ----  --------  -------------  ----------------------

PS1     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS2     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS3     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS4     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS5     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS6     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS7     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS8     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good

PS Current Configuration Mode : Combined
PS Current Operating State    : Combined

Power supplies currently active    : 8
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   2030    2030
Inline Power   106     23570
-------------  ------  ---------
Total          2136    25600


Automatic Linecard Shutdown : Enabled
Power Budget Mode           : Dual Sup
```

```
                           autoLC    Power                                    Out of  In
Mod  Model No             Priority  State     Budget  Instantaneous  Peak  Reset   Reset
---  -------------------  --------  --------  ------  -------------  ----  ------  -----
1    C9400-LC-24XS         0         accepted  200     87             88    200     10
2    C9400-LC-48T          1         accepted  65      35             43    65      5
3    C9400-SUP-1           0         accepted  400     235            253   400     130
4    C9400-SUP-1           0         accepted  400     235            253   400     130
5    C9400-LC-48T          2         accepted  65      35             37    65      5
6    C9400-LC-24XS         3         accepted  200     87             88    200     10
7    C9400-LC-48UX         4         accepted  350     189            203   350     15
--   Fan Tray              0         accepted  350     --             --    350     --
---  -------------------  --------  --------  ------  -------------  ----  ------  -----
Total   2030
```

```
Device# show power detail

Power                                                          Fan States
Supply  Model No              Type  Capacity  Status        1     2     3     4
------  --------------------  ----  --------  ------------  ------------------------

PS1     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS2     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS3     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS4     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS5     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS6     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS7     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good
PS8     C9400-PWR-2100AC      AC    2100 W    active        good  good  good  good

PS Current Configuration Mode : Combined
PS Current Operating State    : Combined

Power supplies currently active    : 8
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   2030    2030
Inline Power   106     14770
-------------  ------  ---------
Total          2136    16800


Automatic Linecard Shutdown : Enabled
Power Budget Mode           : Dual Sup

                           autoLC    Power                                    Out of  In
Mod  Model No             Priority  State     Budget  Instantaneous  Peak  Reset   Reset
---  -------------------  --------  --------  ------  -------------  ----  ------  -----
1    C9400-LC-24XS         0         accepted  200     87             88    200     10
2    C9400-LC-48T          1         accepted  65      35             43    65      5
3    C9400-SUP-1           0         accepted  400     235            253   400     130
4    C9400-SUP-1           0         accepted  400     235            253   400     130
5    C9400-LC-48T          2         accepted  65      35             37    65      5
6    C9400-LC-24XS         3         accepted  200     87             88    200     10
7    C9400-LC-48UX         4         accepted  350     189            203   350     15
--   Fan Tray              0         accepted  350     --             --    350     --
---  -------------------  --------  --------  ------  -------------  ----  ------  -----
Total   2030
```

```
Device# show power detail

Power                                                          Fan States
Supply  Model No              Type  Capacity  Status         1     2     3     4
------  --------------------  ----  --------  -------------  ----------------------

PS1     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS2     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS3     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS4     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS5     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS6     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS7     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good
PS8     C9400-PWR-3200AC      AC    3200 W    active         good  good  good  good

PS Current Configuration Mode : Combined
PS Current Operating State    : Combined

Power supplies currently active    : 8
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
------------  ------  ---------
System Power   2030    2030
Inline Power   106     23570
------------  ------  ---------
Total          2136    25600


Automatic Linecard Shutdown : Enabled
Power Budget Mode           : Dual Sup

                           autoLC    Power                                  Out of  In
Mod  Model No             Priority  State     Budget  Instantaneous  Peak  Reset   Reset
---  -------------------  --------  --------  ------  -------------  ----  ------  -----
1    C9400-LC-24XS         0         accepted  200     87             88    200     10
2    C9400-LC-48T          1         accepted  65      35             43    65      5
3    C9400-SUP-1           0         accepted  400     235            253   400     130
4    C9400-SUP-1           0         accepted  400     235            253   400     130
5    C9400-LC-48T          2         accepted  65      35             37    65      5
6    C9400-LC-24XS         3         accepted  200     87             88    200     10
7    C9400-LC-48UX         4         accepted  350     189            203   350     15
--   Fan Tray              0         accepted  350     --             --    350     --
---  -------------------  --------  --------  ------  -------------  ----  ------  -----
Total   2030
```

# 電源管理の考慮事項

電源装置が供給する以上の電力を必要とするスイッチを構成する可能性があります。

- 搭載したモジュールの所要電力が、電源装置によって供給される電力を超える場合
  - スイッチに電源要件を満たすことができない単一の電源モジュールがある場合、次のエラーメッセージが表示されます。

    ```
    Insufficient power supplies present for specified configuration
    ```

    **show power** コマンド出力でも、入力電力が不足しているこの状態を示します。

- スイッチに複数の電源モジュールがあり、搭載されたモジュールの所要電力が電源装置によって供給される電力を超える場合、次のエラーメッセージが表示されます。

  ```
  Insufficient number of power supplies (2) are installed for power redundancy
  mode
  ```

  **show power** コマンド出力でも、入力電力が不足しているこの状態を示します。

スイッチにモジュールを増設しようとして電源装置によって供給される電力を超える場合、スイッチはただちに増設分のモジュールをリセットモードにし、次のエラーメッセージが表示されます。

```
Power doesn't meet minimum system power requirement.
```

また、機能しているシャーシの電源を切り、ラインカードを増設するか、モジュール構成を変更して所要電力が使用できる電力を超えるようになった場合、再度スイッチの電源を入れると、1 つまたは複数のモラインカードがリセット モードになります。

- PoE の所要電力が、電源装置によって供給される PoE を超える場合

  システムの電力を消費している IP Phone が多すぎる場合、IP Phone への電力が削減され、電源装置に適切な所要電力に削減されるように、一部の IP Phone の電源が切断されることもあります。

  リセットモードのモジュールは、シャーシに取り付けられている限り、電力を消費し続けます。モジュールをオンラインにするときに必要な電力は、**show power module** コマンドを使用して判断します。

  使用するシステムの所要電力を算出し、システムの電源が十分であるかどうかを確認するには、スーパーバイザ モジュール、ファン トレイ、および搭載したラインカード（PoE を含む）が消費する電力を合算します。PoE には、すべての電話の所要電力を合計します。

  802.3at 準拠の PoE モジュールは、最大 60W の PoE を使用できます。スイッチに接続された受電装置に十分な電力が供給されるように、802.3at 準拠の PoE モジュールごとに、PoE 所要電力に 60W を追加してください。

  PoE がサポートされるすべてのラインカード（C9400-LC-48UX、C9400-LC-48U、C9400-LC-48P）では、PoE 消費電力は管理上の PoE と同じです。

  受電デバイス（PD）が割り当てられた電力よりも多くの電力を消費する場合、次の Imax エラーが生成されます。さらにポートがシャットダウンし、障害状態になります。

```
*Jun 21 10:06:06.149: %ILPOWER-3-CONTROLLER_PORT_ERR: Controller port error, Interface
 Gi7/0/13: Power Controller reports power Imax error
*Jun 21 10:06:06.208: %ILPOWER-5-IEEE_DISCONNECT: Interface Gi7/0/13: PD removed
Device# show power inline 7/0/13
Gi7/0/13  auto   faulty     0.0        0.0        n/a    n/a
```

# 電源モードの選択

使用する電源装置とその数は、スイッチのハードウェア構成によって決まります。たとえば、スイッチの設定で単一の電源モジュールが提供するよりも多くの電力が必要な場合は、cisco.com

の Cisco Power Calculator を使用して、複合モードまたは冗長モードに必要な電源モジュールの数を決定します。

## 冗長モードの設定

デフォルトでは、スイッチの電源装置は複合モードで動作するように設定されています。冗長モードを効果的に使用するには、次の点に注意してください。

- 電源モードが冗長モードに設定されており、電源装置が 1 つしか搭載されていない場合は、スイッチがその設定を受け入れますが、冗長性なしで動作します。
- スイッチ構成をサポートできるだけの電力を備えた電源モジュールを選択してください。
- システムに必要な電源の数を評価するには、Cisco Power Calculator を使用します。十分な数の電源モジュールを取り付け、シャーシと PoE の要件が最大使用可能電力を下回るようにしてください。電源装置は、起動時にシャーシおよびPoE所要電力に対応するように、自動的に電源リソースを調整します。最初にモジュールが、続いて IP Phone が起動します。
- システム電源を最適に使用するには、スイッチで冗長モードを設定するときに同じ容量の電源モジュールを選択します。

冗長モードを設定するには、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**<br>例：<br>`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 | **power redundancy-mode redundant** [**n+1** *standby-PSslot* \| **n+1** *standby-PSslot*]<br>例：<br>`Device(config)# power redundancy-mode redundant n+1 5`<br>OR<br>`Device(config)# power redundancy-mode redundant n+n 5 6 7 8` | 次のオプションから選択できます。<br>• **power redundancy-mode redundant n+1** *standby-PSslot*：n+1 冗長モードを設定します。スタンバイ電源モジュールのスロット番号を入力します。この冗長モードのデフォルト スタンバイ スロットは 8 です。<br>この n+1 の例では、スロット PS5（デフォルトの PS8 ではない）の電源モジュールが指定のスタンバイ モジュールであり、それに応じて設定されています。他のすべてのスロットに取り付けられた動作電源モジュールはアクティブです。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • **power redundancy-mode redundant n+n** *standby-PSslot*：n+n 冗長モードを設定します。スタンバイ電源モジュールのスロット番号を入力します。この冗長モードのデフォルトスタンバイ スロットは 5 ～ 8 です。<br>この n+n の例では、スロット PS5、PS6、PS7、および PS8 の電源モジュールがスタンバイモジュールとして使用されており、それに応じて設定されています。他のすべてのスロットに取り付けられた動作電源モジュールはアクティブです。<br>容量の異なる電源モジュールを使用する場合は、次のガイドラインも順守してください。<br>• n+1 冗長モードの場合、ワット数または容量が最大の電源モジュールをスタンバイとして設定します。<br>• n+n 冗長モードの場合：合計スタンバイ出力電力がアクティブな合計出力電力以上であることを確認します。 |
| **ステップ 3** | **end**<br>例：<br>`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 4** | **show power**<br>例：<br>`Device# show power` | 電源の冗長モード情報を表示します。 |

## 複合モードの設定

複合モードを効果的に使用するには、次の注意事項に従ってください。

- 十分な電力を供給できる電源モジュールを選択し、シャーシおよびPoE所要電力が最大電力を超えないようにします。電源モジュールは、起動時にシャーシおよびPoE所要電力に対応するように、自動的に電源リソースを調整します。

- 電源モードが複合モードに設定されており、電源装置が 1 つしか搭載されていない場合は、スイッチがその設定を受け入れますが、電力は1つの電源装置からしか利用できません。
- スイッチが複合モードに設定されている場合、供給される電力は、個々の電源装置の合計値となります。

スイッチに複合モードを設定するには、次の作業を行います。

### 始める前に

このモードはすべての電源装置の使用可能な電力を使用することに注意してください。ただし、スイッチの電源冗長性は失われます。

### 手順

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`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 | **power redundancy-mode combined**<br>例：<br>`Device(config)# power redundancy-mode combined` | 電源モードを複合モードに設定します。 |
| ステップ 3 | **end**<br>例：<br>`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 4 | **show power**<br>例：<br>`Device# show power` | 電源の冗長モード情報を表示します。 |

# スーパバイザモジュールの電力バジェット

電力バジェット、または必要なバジェット電力は、スーパバイザモジュール、スイッチングモジュール（ラインカード）、およびファントレイがシャーシで動作するためにシステムが必要とし、確保する電力です。**show power** および **show power detail** コマンドの出力では、これは `System Power` として表示されます。システム内の他のコンポーネントで使用するために、この必要なバジェット電力の一部を自動的にリダイレクトすることはできません。

ここでは、スーパバイザモジュールおよび使用可能な設定オプションに関して、電力バジェットがどのように機能するかについて説明します。

デフォルトでは、システムは冗長設定のために電力を確保し、高可用性を実現します。つまり、システムは、必要なバジェット電力（System Power）の一部として、シャーシ内の両方のスーパバイザモジュールで必要な電力を確保します。

単一のスーパバイザに電力を確保するようにシステムを設定することもできます。この設定オプションは、単一のスーパバイザが取り付けられていて、使用可能な合計電力がすべてのラインカードおよびPoEポートをイネーブルにするのに十分でない場合に適しています。このようなシナリオでは、単一のスーパバイザに電力を確保するようにスイッチを構成すると、電力が解放され、代わりに PoE ポートやラインカードなどの他のコンポーネントに使用できます。

次の制限事項およびガイドラインに留意してください。

- 両方のスーパバイザモジュールを取り付けている場合、単一のスーパバイザに電力バジェットモードを設定することはできません。システムは設定を拒否し、次のメッセージが表示されます。`cannot enable single sup mode when remote supervisor is present.`
- 両方のスーパバイザモジュールを取り付け、デフォルト設定が有効になっている場合は、全体のシステム要件（ラインカード、ファントレイ、および POE ポートを含む）を満たすために必要な数の電源モジュールを取り付ける必要があります。電源モジュールの数が不足している状況を改善するために、2 番目のスーパバイザを取り外さないでください。
- 単一のスーパバイザモジュールを取り付け、単一のスーパバイザに電力バジェットモードを設定した場合は、2 番目のスーパバイザを取り付けないでください。その場合、次のいずれかが発生する可能性があります。
  - 次のエラーメッセージが表示され、設定が拒否される場合があります。`ERROR: Remote supervisor has been detected in slot 6. System is configured to SINGLE-SUP power mode. Remove remote supervisor IMMEDIATELY.`
  - このアクションに、2 番目のスーパバイザに十分な電力がシステムにない低電力状態が伴う場合、システムがシャットダウンすることがあります。

  シングル スーパバイザ セットアップからデュアル スーパバイザ セットアップに安全に移行する方法については、以下のタスク「シングル スーパバイザ セットアップからデュアル スーパバイザ セットアップへの移行」を参照してください。

以下のタスクでは、使用可能な設定オプションについて説明します。

## シングルスーパバイザの電力バジェットモードの設定

特権 EXEC モードで次の手順を実行し、シングル スーパバイザ セットアップの電力バジェットモードを設定します。

### 始める前に

次の前提条件を満たしていることを確認します。

- シャーシにスーパバイザモジュールが 1 つだけ取り付けられている。
- 2 番目のスーパバイザスロットにブランクが取り付けられている。

手順

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`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 | **power budget mode** {**single-sup**}<br>例：<br>`Device(config)# power budget mode single-sup` | シャーシ内の 1 つのスーパバイザモジュールに電力を予約します。 |
| ステップ 3 | **end**<br>例：<br>`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。 |

## シングル スーパバイザ セットアップからデュアル スーパバイザ セットアップへの移行

シングル スーパバイザ セットアップからデュアル スーパバイザ セットアップに移行するには、特権 EXEC モードで次の手順を実行します。

### 始める前に

デュアルスーパバイザのセットアップに必要な電力を計算します。Cisco Power Calculator（CPC）を使用すると、指定した構成の電源要件を計算できます。

1. https://cpc.cloudapps.cisco.com/cpc → [Launch Cisco Power Calculator] に移動します。
2. [Product Family]、[Chassis]、[Supervisor Engine]（両方のスーパバイザスロット）、[Input Voltage]、および [Line Card] フィールドに適切な値を選択します。[Next] をクリックして、結果を表示します。
3. 表示される結果で、[Configuration Details] セクションを見つけ、スーパバイザモジュールの [Output Power] を確認します。これは、2 番目のスーパバイザを安全に取り付けるためにシステムで使用できる必要がある予備の電力量です。
4. **show power** コマンドは特権 EXEC モードで入力します。

   このコマンドは、電源構成の情報を表示します。

   出力で、[Total Maximum Available] と [Total Used] の差を確認します。これは、スーパバイザモジュールの [Output Power] 列に表示される CPC の値よりも大きい必要があります。これが該当する場合、タスクを続行します。そうでない場合は、必要な数の追加の電源モジュールを取り付けます。

手順

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`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 power budget mode** {**single-sup**}<br>例：<br>`Device(config)# no power budget mode single-sup` | システムがシャーシ内の両方のスーパバイザモジュール用に電力を確保するデフォルト設定に戻します。 |
| ステップ 3 | **end**<br>例：<br>`Device(config)# end` | 設定モードを終了します。 |
| ステップ 4 | 2つ目のスーパバイザモジュールをスーパバイザスロットに挿入します。 | 詳細な手順については、cisco.com の Supervisor Module Installation Note →「Removal and Replacement Procedures」を参照してください。 |

# 自動ラインカードシャットダウンの有効化

自動ラインカード（autoLC）シャットダウンを使用すると、使用可能な合計電力が **show power** 特権 EXEC コマンドの power summary に表示される使用電力の合計と等しいかそれよりも大きくなるまで、ハードウェアは電力制限時にラインカードを自動的にシャットダウンできます。ラインカード電源優先順位を設定することもできます。つまり、電力制限時にラインカードをシャットダウンする順序を指定できます。

この機能は、電源障害イベントや優先順位の高いラインカードのシャットダウンイベントの場合のスイッチの確定的動作を提供します。

Cisco IOS XE Gibraltar 16.12.1 以降では、autoLC シャットダウンは常に有効になり、無効にすることはできません。以前のすべてのリリースでは、autoLC シャットダウンはデフォルトでは無効になっています。電力制限の発生時にシステムハードウェアでラインカードをシャットダウンする場合は、手動で有効にする必要があります。

autoLC シャットダウンおよびラインカード電源優先順位を設定するには、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**<br>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 | **power supply autolc shutdown**<br>例：<br>`Device(config)# power supply autolc shutdown` | （任意）電源障害発生時のラインカードの自動シャットダウンを有効にします。<br>（注） Cisco IOS XE Gibraltar 16.12.1 以降、この機能は常に有効であり、無効にすることができないため、このコマンドを設定する必要はありません。このコマンドの **no** 形式もこのリリースから廃止されています。 |
| ステップ 3 | **power supply autolc priority** *physical-slot-number*<br>例：<br>`Device(config)# power supply autolc priority 1 2 5 6 7` | （任意）ラインカード電源優先順位を設定します。autoLC シャットダウンの優先順位を示すラインカードスロット番号を入力します。最初に入力したスロット番号に最高の優先順位（0）が割り当てられ、障害が発生した場合に最後にシャットダウンされます。<br>ラインカードのスロット番号を一部だけ入力することはできません。たとえば、7 スロットシャーシの場合、5 つのすべてのラインカードスロットに言及する必要があります。<br>左に示す設定例では、7 スロットシャーシの物理スロット番号の順序とシステム割り当ての優先順位は次のとおりです。 |

| 設定順序 | autoLC 優先順位 |
|---|---|
| 1 | 0（最後にシャットダウン） |
| 2 | 1 |
| 5 | 2 |
| 6 | 3 |
| 7 | 4（最初にシャットダウン） |

順序を指定せずに autoLC シャットダウンを有効にした場合、デフォルトでは、物理スロット番号が大きいラインカード

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | から小さいラインカードの順にラインカードがシャットダウンされます。したがって、デフォルト設定は次のようになります。<br>• 4 スロットシャーシ：電源 autoLC 優先順位 1 4<br>• 7 スロットシャーシ：電源 autoLC 優先順位 1 2 5 6 7<br>• 10 スロットシャーシ：電源 autoLC 優先順位 1 2 3 4 7 8 9 10 |
| ステップ 4 | **end**<br>例：<br>`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 | **show power module**<br>例：<br>`Device# show power module` | 電源冗長モード情報を表示し、autoLC が有効かどうかに関する情報を表示します。 |

# ラインカードの電源切断

スイッチに搭載されたすべてのモジュールに供給する十分な電力がシステムにない場合は、1 つ以上のラインカードの電源を切断して、電力オフモードにできます。

ラインカードの電源を切断するには、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 | **configure terminal**<br>例：<br>`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 | **hw-module subslot** *card slot/subslot number* **shutdown unpowered**<br>例：<br>`Device(config)# hw-module subslot 1/0 shutdown unpowered` | 指定したモジュールを低電力モードにして、電源を切断します。 |
| ステップ 3 | **end**<br>例： | グローバル コンフィギュレーション モードを終了します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# end` | |

# 電源モードと動作状態の設定例

このセクションの例では、さまざまな電源設定を示します。ここでは、『*Cisco Catalyst 9400 Series Switches Hardware Installation Guide*』の「Power Supply Module Installation Considerations」と、このドキュメントの「動作状態」のセクションが、考えられる電源モードの設定にどのように影響するかについて説明します。複合電源モードと冗長電源モードの両方、およびその結果の動作状態について説明します。

## 例：複合モードと状態（AC および DC 入力）

次の表に、Cisco Catalyst 9400 シリーズ シャーシの 2 行の電源スロットを示します。電源スロットは、PS1、PS2 などと示されます。この例では、同じ容量（3200W）の電源モジュールがスロット 1 ー 8 に取り付けられています。スロット 1 ー 4 には AC 入力電源モジュール、スロット 5 ー 8 には DC 入力電源モジュールです。

| | | | |
|---|---|---|---|
| **PS1**（アクティブ）<br>C9400-PWR-3200AC | **PS2**（アクティブ）<br>C9400-PWR-3200AC | **PS3**（アクティブ）<br>C9400-PWR-3200AC | **PS4**（アクティブ）<br>C9400-PWR-3200AC |
| **PS5**（アクティブ）<br>C9400-PWR-3200DC | **PS6**（アクティブ）<br>C9400-PWR-3200DC | **PS7**（アクティブ）<br>C9400-PWR-3200DC | **PS8**（アクティブ）<br>C9400-PWR-3200DC |

使用可能なすべての電源モジュールがアクティブになって電力を共有し、最大 100% のキャパシティで稼働できます。スタンバイ電源モジュールが設定されていないため、デバイスは複合動作状態です。次の理由から、これは複合モードの有効な設定です。

- 取り付けられているすべての電源モジュールがアクティブで使用可能である。

  出力例で、`Status= active` であることを確認してください。

- 取り付けられているすべての電源モジュールの容量が同じである。

  出力例で、`Capacity = 3200 W` であることを確認してください。

- 取り付けられているすべての AC 入力電源モジュールの AC 入力電圧が 220 VAC である。

  出力例で、AC 入力モジュールの `Capacity` フィールドにこれが反映されています。

次に、このセットアップ（**show power** 特権 EXEC コマンド）の出力例を示します。

```
Device# show power

Power                                                    Fan States
Supply  Model No              Type  Capacity  Status     1     2
------  --------------------  ----  --------  ---------- ----------
PS1     C9400-PWR-3200AC      ac    3200 W    active     good  good
PS2     C9400-PWR-3200AC      ac    3200 W    active     good  good
```

```
PS3     C9400-PWR-3200AC      ac    3200 W    active        good  good
PS4     C9400-PWR-3200AC      ac    3200 W    active        good  good
PS5     C9400-PWR-3200DC      dc    3200 W    active        good  good
PS6     C9400-PWR-3200DC      dc    3200 W    active        good  good
PS7     C9400-PWR-3200DC      dc    3200 W    active        good  good
PS8     C9400-PWR-3200DC      dc    3200 W    active        good  good

PS Current Configuration Mode : Combined
PS Current Operating State    : Combined

Power supplies currently active    : 8
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   3505    3505
Inline Power   0       22095
-------------  ------  ---------
Total          3505    25600
```

複合モードで障害が発生すると、稼働中の各電源が出力を上げます。出力電力がシステム要件を満たしておらず、**power supply autolc shutdown** コマンドが無効である場合、稼働中のすべての電源が過負荷になり、過電流によるシャットダウンを引き起こすおそれがあります。その場合、すべてのシステム電力が失われます。**power supply autolc shutdown** コマンドを有効にすることを推奨します。

複合モードのその他の有効な設定オプション：

- 取り付けられているすべてのモジュールが DC 入力電源モジュール。すべてのモジュールをアクティブとして設定。
- 取り付けられているすべてのモジュールが同じ容量、同じ AC 入力電圧レベルの AC 入力電源モジュール。すべてのモジュールをアクティブとして設定。

# 例：複合モードと状態（DC 入力のみ）

次の表に、Cisco Catalyst 9400 シリーズ シャーシの 2 行の電源スロットを示します。電源スロットは、PS1、PS2 などと示されます。この例では、同じ容量とタイプ（C9400-PWR-3200DC）の電源モジュールがスロット 1 ー 8 に取り付けられています。

| | | | |
|---|---|---|---|
| **PS1**（アクティブ）<br>C9400-PWR-3200DC | **PS2**（アクティブ）<br>C9400-PWR-3200DC | **PS3**（アクティブ）<br>C9400-PWR-3200DC | **PS4**（アクティブ）<br>C9400-PWR-3200DC |
| **PS5**（アクティブ）<br>C9400-PWR-3200DC | **PS6**（アクティブ）<br>C9400-PWR-3200DC | **PS7**（アクティブ）<br>C9400-PWR-3200DC | **PS8**（アクティブ）<br>C9400-PWR-3200DC |

使用可能なすべての電源モジュールがアクティブになって電力を共有し、最大 100% のキャパシティで稼働できます。スタンバイ電源モジュールが設定されていないため、デバイスは複合動作状態です。このセットアップでは、他に考慮すべき条件はありません。

次に、このセットアップ（**show power** 特権 EXEC コマンド）の出力例を示します。

```
Device# show power

Power                                                          Fan States
Supply  Model No              Type  Capacity  Status         1     2     3     4
------  --------------------  ----  --------  -------------  ----------------------

PS1     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS2     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS3     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS4     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS5     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS6     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS7     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS8     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good

PS Current Configuration Mode : Combined
PS Current Operating State    : Combined

Power supplies currently active    : 8
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   2030    2030
Inline Power   106     23570
-------------  ------  ---------
Total          2136    25600


Automatic Linecard Shutdown : Enabled
Power Budget Mode           : Dual Sup

                           autoLC    Power                                  Out of  In
Mod  Model No             Priority  State     Budget  Instantaneous  Peak  Reset   Reset
---  --------------------  --------  --------  ------  -------------  ----  ------  -----
1    C9400-LC-48XS         0         accepted  200     85             90    200     10
2    C9400-LC-48T          1         accepted  65      35             43    65      5
3    C9400-SUP-1           0         accepted  400     235            253   400     130
4    C9400-SUP-1           0         accepted  400     235            253   400     130
5    C9400-LC-48T          2         accepted  65      35             37    65      5
6    C9400-LC-24XS         3         accepted  200     87             88    200     10
7    C9400-LC-48UX         4         accepted  350     189            203   350     15
--   Fan Tray              0         accepted  350     --             --    350     --
---  --------------------  --------  --------  ------  -------------  ----  ------  -----
Total   2030
```

複合モードで障害が発生すると、稼働中の各電源が出力を上げます。出力電力がシステム要件を満たしておらず、**power supply autolc shutdown** コマンドが無効である場合、稼働中のすべての電源が過負荷になり、過電流によるシャットダウンを引き起こすおそれがあります。その場合、すべてのシステム電力が失われます。**power supply autolc shutdown** コマンドを有効にすることを推奨します。

複合モードのその他の有効な設定オプション：

- AC 入力電源モジュールと DC 入力電源モジュールが混在し、すべての容量が同じである。さらに、このセットアップの AC 入力モジュールとすべての AC 入力電圧レベルが同じ。すべてのモジュールをアクティブとして設定。

- 取り付けられているすべてのモジュールが同じ容量、同じ AC 入力電圧レベルの AC 入力電源モジュール。すべてのモジュールをアクティブとして設定。

# 例：n+1 冗長モード（容量が同じでタイプが異なる電源モジュール + 通常保護状態）

次の表に、Cisco Catalyst 9400 シリーズ シャーシの 2 行の電源スロットを示します。電源スロットは、PS1、PS2 などと示されます。この例では、同じ容量（3200W）の電源モジュールがスロット 1 ー 8 に取り付けられています。スロット 1 ー 4 には AC 入力電源モジュール、スロット 5 ー 8 には DC 入力電源モジュールです。スロット 8 には +1 スタンバイ電源モジュールがあります。

| | | | |
|---|---|---|---|
| **PS1**（アクティブ）<br>C9400-PWR-3200AC | **PS2**（アクティブ）<br>C9400-PWR-3200AC | **PS3**（アクティブ）<br>C9400-PWR-3200AC | **PS4**（アクティブ）<br>C9400-PWR-3200AC |
| **PS5**（アクティブ）<br>C9400-PWR-3200DC | **PS6**（アクティブ）<br>C9400-PWR-3200DC | **PS7**（アクティブ）<br>C9400-PWR-3200DC | **PS8**（スタンバイ）<br>C9400-PWR-3200DC |

スイッチは、通常保護状態の n+1 冗長モードに必要なすべての条件を満たしています。

- 1 つの電源モジュールがスタンバイとして設定されているため、n+1 冗長モードです。

  また、n+1 冗長モードの条件をすべて満たしています。

  - AC 入力電源モジュールと DC 入力電源モジュールが混在し、すべてのモジュールの容量が同じである。

    同じ出力で、`Capacity = 3200 W` であることを確認してください。

  - 取り付けられているすべての AC 入力電源モジュールの AC 入力電圧が 220 VAC である（AC 入力モジュールの `Capacity` フィールドにこれが反映されます）。

- 次の理由により、通常の保護状態になっています。

  合計スタンバイ出力電力（3200 W）がアクティブな合計出力電力（22400）よりも小さい

  および

  アクティブな合計出力電力（22400）が必要なバジェット電力（3505）より大きい

```
Switch# show power
Power                                                          Fan States
Supply  Model No              Type  Capacity  Status         1     2
------  --------------------  ----  --------  -------------  ----------
PS1     C9400-PWR-3200AC      ac    3200 W    active         good  good
PS2     C9400-PWR-3200AC      ac    3200 W    active         good  good
PS3     C9400-PWR-3200AC      ac    3200 W    active         good  good
PS4     C9400-PWR-3200AC      ac    3200 W    active         good  good
PS5     C9400-PWR-3200DC      dc    3200 W    active         good  good
PS6     C9400-PWR-3200DC      dc    3200 W    active         good  good
PS7     C9400-PWR-3200DC      dc    3200 W    active         good  good
PS8     C9400-PWR-3200DC      dc    3200 W    standby        good  good
```

```
PS Current Configuration Mode : N+1 redundant
PS Current Operating State    : Normal protected
PS Slots Configured standby   : PS8

Power supplies currently active    : 7
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   3505    3505
Inline Power   0       18895
-------------  ------  ---------
Total          3505    22400
```

n+1 モードのその他の有効な設定オプション：

- 取り付けられているすべてのモジュールが同じ容量、同じ AC 入力電圧レベルの AC 入力電源モジュール。1 つのモジュールをスタンバイとして設定。
- 取り付けられているすべてのモジュールが同じ容量の DC 入力電源モジュール。1 つのモジュールがスタンバイとして設定。

# 例：n+1 冗長モード（容量とタイプが同じ電源モジュール + 通常保護状態）

次の表に、Cisco Catalyst 9400 シリーズ シャーシの 2 行の電源スロットを示します。電源スロットは、PS1、PS2 などと示されます。この例では、同じ容量とタイプ（C9400-PWR-3200DC）の電源モジュールがスロット 1 ー 8 に取り付けられています。スロット 8 には +1 スタンバイ電源モジュールがあります。

| | | | |
|---|---|---|---|
| **PS1**（アクティブ）<br>C9400-PWR-3200DC | **PS2**（アクティブ）<br>C9400-PWR-3200DC | **PS3**（アクティブ）<br>C9400-PWR-3200DC | **PS4**（アクティブ）<br>C9400-PWR-3200DC |
| **PS5**（アクティブ）<br>C9400-PWR-3200DC | **PS6**（アクティブ）<br>C9400-PWR-3200DC | **PS7**（アクティブ）<br>C9400-PWR-3200DC | **PS8**（スタンバイ）<br>C9400-PWR-3200DC |

デバイスは、通常保護状態の n+1 冗長モードに必要なすべての条件を満たしています。

- 1 つの電源モジュールがスタンバイとして設定されているため、n+1 冗長モードです。

  取り付けられているすべてのモジュールが DC 入力電源モジュールであるため、他に n+1 冗長モードの条件はありません。

- 次の理由により、通常の保護状態になっています。

  合計スタンバイ出力電力（3200 W）がアクティブな合計出力電力（22400）よりも小さい

  および

  アクティブな合計出力電力（22400）が必要なバジェット電力（2030）より大きい

```
Device# show power

Power                                                          Fan States
Supply  Model No              Type  Capacity  Status         1     2     3     4
------  --------------------  ----  --------  -------------  ----------------------

PS1     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS2     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS3     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS4     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS5     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS6     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS7     C9400-PWR-3200DC      dc    3200 W    active         good  good  good  good
PS8     C9400-PWR-3200DC      dc    3200 W    standby        n.a.  n.a.  n.a.  n.a.

PS Current Configuration Mode : N+1 redundant
PS Current Operating State    : Normal protected
PS Slots Configured standby   : PS8

Power supplies currently active    : 7
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
------------  ------  ---------
System Power  2030    2030
Inline Power  106     20370
------------  ------  ---------
Total         2136    22400
```

n+1 モードのその他の有効な設定オプション：

- 取り付けられているすべてのモジュールが同じ容量、同じ AC 入力電圧レベルの AC 入力電源モジュール。1 つのモジュールをスタンバイとして設定。
- AC 入力と DC 入力の電源モジュールが混在し、すべてのモジュールの容量が同じで、取り付けられているすべての AC 入力電源モジュールの AC 入力電圧が 220 VAC である。

# 例：n+n 冗長モード（同じ容量の電源モジュール + 完全保護状態）

次の表に、Cisco Catalyst 9400 シリーズ シャーシの 2 行の電源スロットを示します。電源スロットは、PS1、PS2 などと示されます。この例では、同じ容量（3200W）の電源モジュールがスロット 1 ー 8 に取り付けられています。スロット 1 ー 4 には AC 入力電源モジュールがあり、すべてがアクティブとして設定されています。スロット 5 ー 8 には DC 入力電源モジュールがあり、すべてがスタンバイとして設定されています。

| | | | |
|---|---|---|---|
| **PS1**（アクティブ）<br>C9400-PWR-3200AC | **PS2**（アクティブ）<br>C9400-PWR-3200AC | **PS3**（アクティブ）<br>C9400-PWR-3200AC | **PS4**（アクティブ）<br>C9400-PWR-3200AC |
| **PS5**（スタンバイ）<br>C9400-PWR-3200DC | **PS6**（スタンバイ）<br>C9400-PWR-3200DC | **PS7**（スタンバイ）<br>C9400-PWR-3200DC | **PS8**（スタンバイ）<br>C9400-PWR-3200DC |

デバイスは、完全保護状態の n+n 冗長モードに必要なすべての条件を満たしています。

- n 個の電源モジュールがアクティブとして設定され、同じ数がスタンバイとして設定されているため、n+n 冗長モードです。

  また、n+n 冗長モードの条件をすべて満たしています。

  - アクティブとして設定するすべての電源モジュールのタイプが同じであり、スタンバイとして設定するすべてのモジュールのタイプが同じである。
  - AC 入力と DC 入力の電源モジュールが混在し、シャーシに取り付けられているすべての AC 入力電源モジュールの容量が同じで、AC 入力電圧レベルが同じである。

    `Type= ac` の出力例では、すべてのモジュールで `Capacity = 3200 W` です。AC 入力モジュールの `Capacity` フィールドには、電圧が 220 VAC であることも反映されます。
  - AC 入力と DC 入力の電源モジュールが混在し、取り付けられているすべての DC 入力電源モジュールがスタンバイとして設定されている。

- 以下の条件により、完全保護状態です。

  アクティブな合計出力電力（12800）が必要なバジェット電力（3505）より大きい

  および

  合計スタンバイ出力電力（12800）がアクティブな合計出力電力（12800）以上である

```
Device# show power
Power                                                    Fan States
Supply  Model No              Type  Capacity  Status        1     2
------  --------------------  ----  --------  ------------  ----------
PS1     C9400-PWR-3200AC      ac    3200 W    active        good  good
PS2     C9400-PWR-3200AC      ac    3200 W    active        good  good
PS3     C9400-PWR-3200AC      ac    3200 W    active        good  good
PS4     C9400-PWR-3200AC      ac    3200 W    active        good  good
PS5     C9400-PWR-3200DC      dc    3200 W    standby       good  good
PS6     C9400-PWR-3200DC      dc    3200 W    standby       good  good
PS7     C9400-PWR-3200DC      dc    3200 W    standby       good  good
PS8     C9400-PWR-3200DC      dc    3200 W    standby       good  good

PS Current Configuration Mode : N+N redundant
PS Current Operating State    : Full protected
PS Slots Configured standby   : PS5, PS6, PS7, PS8

Power supplies currently active    : 4
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   3505    3505
Inline Power   0       9295
-------------  ------  ---------
Total          3505    12800
```

n+n モードのその他の有効な設定オプション：

- 取り付けられているすべてのモジュールが同じ容量で同じ AC 入力電圧レベルの AC 入力電源モジュール。n 個のモジュールがアクティブとして設定され、n 個がスタンバイとして設定。

- 取り付けられているすべてのモジュールが同じ容量の DC 入力電源モジュール。n 個のモジュールがアクティブとして設定され、n 個がスタンバイとして設定。

# 例：n+n 冗長モード（容量が異なる電源モジュール + 通常保護状態）

次の表に、Cisco Catalyst 9400 シリーズ シャーシの 2 行の電源スロットを示します。電源スロットは、PS1、PS2 などと示されます。この例では、スロット 1 ー 4 に同じ容量（2100W）の AC 入力電源モジュールがあり、すべてがアクティブとして設定されています。スロット 5 ー 8 には DC 入力電源モジュールがあり、すべてがスタンバイとして設定されています。

| | | | |
|---|---|---|---|
| **PS1**（アクティブ）<br>C9400-PWR-2100AC | **PS2**（アクティブ）<br>C9400-PWR-2100AC | **PS3**（アクティブ）<br>C9400-PWR-2100AC | **PS4**（アクティブ）<br>C9400-PWR-2100AC |
| **PS5**（スタンバイ）<br>C9400-PWR-3200DC | **PS6**（スタンバイ）<br>C9400-PWR-3200DC | **PS7**（スタンバイ）<br>C9400-PWR-3200DC | **PS8**（スタンバイ）<br>C9400-PWR-3200DC |

デバイスは、完全保護状態の n+n 冗長モードに必要なすべての条件を満たしています。

- n 個の電源モジュールがアクティブとして設定され、同じ数がスタンバイとして設定されているため、n+n 冗長モードです。

  また、n+n 冗長モードの条件をすべて満たしています。

  - アクティブとして設定するすべての電源モジュールのタイプが同じであり、スタンバイとして設定するすべてのモジュールのタイプが同じである。
  - AC 入力と DC 入力の電源モジュールが混在し、シャーシに取り付けられているすべての AC 入力電源モジュールの容量が同じで、AC 入力電圧レベルが同じである。

    `Type= ac` の出力例では、すべてのモジュールで `Capacity = 2100 W` です。AC 入力モジュールの `Capacity` フィールドには、電圧が 220 VAC であることも反映されます。
  - AC 入力と DC 入力の電源モジュールが混在し、取り付けられているすべての DC 入力電源モジュールがスタンバイとして設定されている。
  - 異なる容量の電源モジュールが取り付けられているが、容量が高いほうのモジュールがスタンバイとして設定されている。

- この例のセットアップと構成はすべての条件を満たしていますが、完全保護状態は保証されません。

（注） 異なる容量の電源モジュールが検出されると、常に動作状態は通常保護と見なされ、表示されます。

```
Device# show power
Power                                                    Fan States
Supply  Model No              Type  Capacity  Status     1     2
```

```
------  --------------------  ----  --------  ------------  -----------
PS1     C9400-PWR-2100AC      ac    2100 W    active        good  good
PS2     C9400-PWR-2100AC      ac    2100 W    active        good  good
PS3     C9400-PWR-2100AC      ac    2100 W    active        good  good
PS4     C9400-PWR-2100AC      ac    2100 W    active        good  good
PS5     C9400-PWR-3200DC      dc    3200 W    standby       good  good
PS6     C9400-PWR-3200DC      dc    3200 W    standby       good  good
PS7     C9400-PWR-3200DC      dc    3200 W    standby       good  good
PS8     C9400-PWR-3200DC      dc    3200 W    standby       good  good

PS Current Configuration Mode : N+N redundant
PS Current Operating State    : Normal protected
PS Slots Configured standby   : PS5, PS6, PS7, PS8

Power supplies currently active    : 4
Power supplies currently available : 8

Power Summary          Maximum
(in Watts)    Used    Available
-------------  ------  ---------
System Power   3505    3505
Inline Power   0       4895
-------------  ------  ---------
Total          3505    8400
```

n+n モードのその他の有効な設定オプション：

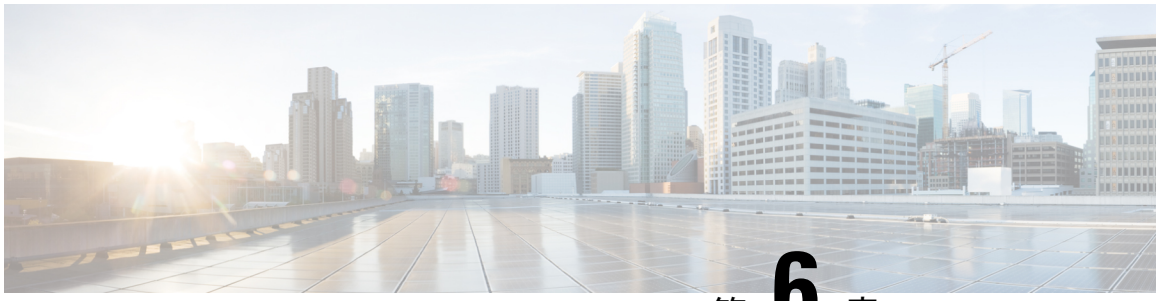
- 取り付けられているすべてのモジュールが同じ容量で同じ AC 入力電圧レベルの AC 入力電源モジュール。n 個のモジュールがアクティブとして設定され、n 個がスタンバイとして設定。
- 取り付けられているすべてのモジュールが同じ容量の DC 入力電源モジュール。n 個のモジュールがアクティブとして設定され、n 個がスタンバイとして設定。

# 環境モニタリングおよび電源管理の機能履歴と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

| リリース | 変更内容 |
|---|---|
| | この機能が導入されました。 |
| Cisco IOS XE Fuji 16.8.1a | 電力バジェットモードが導入されました。<br>2100W AC 入力電源モジュールのサポートが導入されました（C9400-PWR-2100AC）。 |

| リリース | 変更内容 |
| --- | --- |
| Cisco IOS XE Fuji 16.9.1 | 3200W DC入力電源モジュールのサポートが導入されました（C9400-PWR-3200DC）。 |
| Cisco IOS XE Gibraltar 16.12.1 | autoLC シャットダウン機能が変更されました（**power supply autoLC shutdown** グローバル コンフィギュレーション コマンド）。このリリース以降では、autoLC シャットダウンは常に有効になり、無効にすることはできません。このコマンドの **no** 形式もこのリリースで廃止されています。 |



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 （199 ページ）
- SDM テンプレートとスイッチ スタック （201 ページ）
- SDM テンプレートの設定方法 （202 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 （203 ページ）
- SDM テンプレートの設定例 （204 ページ）
- SDM テンプレートに関する追加情報 （208 ページ）
- SDM テンプレートの設定の機能履歴と情報 （208 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 9400 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。

- アクセス
- コア
- SDA
- NAT

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。

**表 *14 : Cisco Catalyst 9400* シリーズ スーパバイザ *1* のテンプレートで許容される機能リソースの概算**

| リソース | アクセス |
|---|---|
| LPM | 64K |

| リソース | アクセス |
|---|---|
| ホスト | 48K |
| レイヤ 2 マルチキャスト | 16 K |
| レイヤ 3 マルチキャスト | 16 K |
| MAC アドレス | 64K |
| SGT | 8 K |
| Flexible Netflow | 128K/ASIC |
| セキュリティ ACL | 18K |
| QoS ACL | 18K |
| PBR/NAT | 2 K |
| Tunnel | 1K |
| LISP | 1K |
| MPLS L3VPN VRF | 255 |
| MPLS Label | 12K |
| MPLS L3VPN ルート VRF | 32K |
| MPLS L3VPN ルート プレフィックス | 4 K |
| MVPN MDT トンネル | 1K |
| L2VPN EOMPLS アタッチメント | 1K |

表 *15 : Cisco Catalyst 9400* シリーズ スーパバイザ *1XL* およびスーパバイザ *1XL-Y* モジュールのテンプレートで許容される機能リソースの概算

| テンプレート名 | アクセス | コア | SDA | NAT |
|---|---|---|---|---|
| LPM | 64K | 64K | 64K | 64K |
| ホスト | 48K | 32K | 80 K | 48K |
| レイヤ 2 マルチキャスト | 16 K | 16 K | 16 K | 16 K |
| レイヤ 3 マルチキャスト | 16 K | 32K | 16 K | 32K |
| MAC アドレス | 64K | 16 K | 16 K | 16 K |

| テンプレート名 | アクセス | コア | SDA | NAT |
|---|---|---|---|---|
| SGT | 8 K | 8 K | 8 K | 8 K |
| Flexible Netflow | 128K/ASIC | 128K/ASIC | 128K/ASIC | 128K/ASIC |
| セキュリティ ACL | 18K | 18K | 18K | 18K |
| QoS ACL | 18K | 18K | 18K | 18K |
| PBR/NAT | 2 K | 2 K | 2 K | 16 K |
| Tunnel | 1K | 1K | 1K | 1K |
| LISP | 1K | 1K | 1K | 1K |
| MPLS L3VPN VRF | 255 | 255 | 該当なし | 255 |
| MPLS Label | 12K | 16 K | | 12K |
| MPLS L3VPN ルート VRF | 32K | 32K | | 32K |
| MPLS L3VPN ルート プレフィックス | 4 K | 4 K | | 4 K |
| MVPN MDT トンネル | 1K | 1K | | 1K |
| L2VPN EOMPLS アタッチメント | 1K | 1K | 1K | 1K |

# 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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** { **access** \| **core** \| \| } <br>例：<br>`Device(config)# sdm prefer access` | スイッチで使用する SDM テンプレートを指定します。キーワードの意味は次のとおりです。<br>• **access**：スイッチをアクセステンプレートに設定します。<br>• **core**：コアテンプレートを設定します。<br>（注） **no sdm prefer** コマンドとデフォルトテンプレートはサポートされません。 |
| ステップ 4 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **reload**<br>例： | オペレーティング システムをリロードします。<br>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 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device# reload` | 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 テンプレートのモニタリングおよびメンテナンス

### SDM テンプレートの確認

SDM テンプレートをモニタおよびメンテナンスするには、次のコマンドを使用します。

| 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らなくなります。

### カスタマイズ可能な SDM テンプレートの確認

適用されるカスタマイズ可能な SDM テンプレートを確認するには、次のコマンドを使用します。

表 *16 :* カスタマイズ可能な *SDM* テンプレートを確認するコマンド

| コマンド | 説明 |
|---|---|
| **show sdm prefer custom** | カスタマイズ可能なSDMテンプレートの機能に適用されるカスタム値を表示します。 |
| **show sdm prefer custom user-input** | カスタマイズ可能なSDMテンプレートでユーザが入力した値を表示します。 |
| **show sdm prefer** | 現在アクティブなカスタマイズされたSDMテンプレートを表示します。 |

カスタマイズ可能な SDM テンプレートのいずれかの機能にゼロのスケール値が割り当てられた場合、デバイスがリロードされた後、その機能は **show sdm prefer custom** コマンドの出力に表示されません。

# SDM テンプレートの設定例

## 例：SDM テンプレートの表示

次に、Cisco Catalyst 9400 シリーズ スーパバイザ 1 モジュールの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
Showing SDM Template Info
This is the Access template.
Number of VLANs: 4094
Unicast MAC addresses: 65536
Overflow Unicast MAC addresses: 1024
L2 Multicast entries: 16384
Overflow L2 Multicast entries: 1024
L3 Multicast entries: 16384
Overflow L3 Multicast entries: 1024
Directly connected routes: 49152
Indirect routes: 65536
STP Instances: 1024
Security Access Control Entries: 18432
QoS Access Control Entries: 18432
Policy Based Routing ACEs / NAT ACEs: 2048
Netflow Input ACEs: 1024
Netflow Output ACEs: 2048
Ingress Netflow ACEs: 1024
Egress Netflow ACEs: 2048
Flow SPAN ACEs: 1024
Tunnels: 1024
LISP Instance Mapping Entries: 1024
Control Plane Entries: 1024
Input Netflow flows: 65536
Output Netflow flows: 65536
SGT/DGT (or) MPLS VPN entries: 8192
SGT/DGT (or) MPLS VPN Overflow entries: 512
Wired clients: 2048
MACSec SPD Entries: 1024
```

```
MPLS L3 VPN VRF: 255
MPLS Labels: 12288
MPLS L3 VPN Routes VRF Mode: 32768
MPLS L3 VPN Routes Prefix Mode: 4096
MVPN MDT Tunnels: 1024
L2 VPN EOMPLS Attachment Circuit: 1024
MAX VPLS Bridge Domains : 128
MAX VPLS Peers Per Bridge Domain: 32
MAX VPLS/VPWS Pseudowires : 4096
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、Cisco Catalyst 9400 シリーズ スーパバイザ 1XL モジュールの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
This is the Access template.
Number of VLANs: 4094
Unicast MAC addresses: 65536
Overflow Unicast MAC addresses: 1024
L2 Multicast entries: 16384
Overflow L2 Multicast entries: 1024
L3 Multicast entries: 16384
Overflow L3 Multicast entries: 1024
Directly connected routes: 49152
Indirect routes: 65536
STP Instances: 1024
Security Access Control Entries: 18432
QoS Access Control Entries: 18432
Policy Based Routing ACEs / NAT ACEs: 2048
Netflow Input ACEs: 1024
Netflow Output ACEs: 2048
Ingress Netflow ACEs: 1024
Egress Netflow ACEs: 2048
Flow SPAN ACEs: 1024
Tunnels: 1024
LISP Instance Mapping Entries: 1024
Control Plane Entries: 1024
Input Netflow flows: 65536
Output Netflow flows: 65536
SGT/DGT (or) MPLS VPN entries: 8192
SGT/DGT (or) MPLS VPN Overflow entries: 512
Wired clients: 2048
MACSec SPD Entries: 1024
MPLS L3 VPN VRF: 255
MPLS Labels: 12288
MPLS L3 VPN Routes VRF Mode: 32768
MPLS L3 VPN Routes Prefix Mode: 4096
MVPN MDT Tunnels: 1024
L2 VPN EOMPLS Attachment Circuit: 1024
MAX VPLS Bridge Domains : 128
MAX VPLS Peers Per Bridge Domain: 32
MAX VPLS/VPWS Pseudowires : 4096
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.

Device
This is the Core template.
Number of VLANs: 4094
Unicast MAC addresses: 16384
Overflow Unicast MAC addresses: 1024
```

```
L2 Multicast entries: 16384
Overflow L2 Multicast entries: 1024
L3 Multicast entries: 32768
Overflow L3 Multicast entries: 1024
Directly connected routes: 32768
Indirect routes: 65536
STP Instances: 1024
Security Access Control Entries: 18432
QoS Access Control Entries: 18432
Policy Based Routing ACEs / NAT ACEs: 2048
Netflow Input ACEs: 1024
Netflow Output ACEs: 2048
Ingress Netflow ACEs: 1024
Egress Netflow ACEs: 2048
Flow SPAN ACEs: 1024
Tunnels: 1024
LISP Instance Mapping Entries: 1024
Control Plane Entries: 1024
Input Netflow flows: 65536
Output Netflow flows: 65536
SGT/DGT (or) MPLS VPN entries: 8192
SGT/DGT (or) MPLS VPN Overflow entries: 512
Wired clients: 2048
MACSec SPD Entries: 256
MPLS L3 VPN VRF: 255
MPLS Labels: 16384
MPLS L3 VPN Routes VRF Mode: 32768
MPLS L3 VPN Routes Prefix Mode: 4096
MVPN MDT Tunnels: 1024
L2 VPN EOMPLS Attachment Circuit: 1024
MAX VPLS Bridge Domains : 128
MAX VPLS Peers Per Bridge Domain: 32
MAX VPLS/VPWS Pseudowires : 4096
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.

Device
This is the NAT template.
Number of VLANs: 4094
Unicast MAC addresses: 16384
Overflow Unicast MAC addresses: 1024
L2 Multicast entries: 16384
Overflow L2 Multicast entries: 1024
L3 Multicast entries: 32768
Overflow L3 Multicast entries: 1024
Directly connected routes: 49152
Indirect routes: 65536
STP Instances: 1024
Security Access Control Entries: 18432
QoS Access Control Entries: 3072
Policy Based Routing ACEs / NAT ACEs: 16384
Netflow Input ACEs: 1024
Netflow Output ACEs: 2048
Ingress Netflow ACEs: 1024
Egress Netflow ACEs: 2048
Flow SPAN ACEs: 1024
Tunnels: 1024
LISP Instance Mapping Entries: 1024
Control Plane Entries: 1024
Input Netflow flows: 65536
Output Netflow flows: 65536
SGT/DGT (or) MPLS VPN entries: 8192
SGT/DGT (or) MPLS VPN Overflow entries: 512
```

```
Wired clients: 2048
MACSec SPD Entries: 256
MPLS L3 VPN VRF: 255
MPLS Labels: 12288
MPLS L3 VPN Routes VRF Mode: 32768
MPLS L3 VPN Routes Prefix Mode: 4096
MVPN MDT Tunnels: 1024
L2 VPN EOMPLS Attachment Circuit: 1024
MAX VPLS Bridge Domains : 128
MAX VPLS Peers Per Bridge Domain: 32
MAX VPLS/VPWS Pseudowires : 4096
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.

Device
This is the SDA template.
Number of VLANs: 4094
Unicast MAC addresses: 16384
Overflow Unicast MAC addresses: 1024
L2 Multicast entries: 16384
Overflow L2 Multicast entries: 1024
L3 Multicast entries: 16384
Overflow L3 Multicast entries: 1024
Directly connected routes: 81920
Indirect routes: 65536
STP Instances: 1024
Security Access Control Entries: 18432
QoS Access Control Entries: 18432
Policy Based Routing ACEs / NAT ACEs: 2048
Netflow Input ACEs: 1024
Netflow Output ACEs: 2048
Ingress Netflow ACEs: 1024
Egress Netflow ACEs: 2048
Flow SPAN ACEs: 1024
Tunnels: 1024
LISP Instance Mapping Entries: 1024
Control Plane Entries: 1024
Input Netflow flows: 65536
Output Netflow flows: 65536
SGT/DGT (or) MPLS VPN entries: 8192
SGT/DGT (or) MPLS VPN Overflow entries: 512
Wired clients: 2048
MACSec SPD Entries: 256
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 access
Device(config)# exit
Device# reload
  Proceed with reload? [confirm]
```

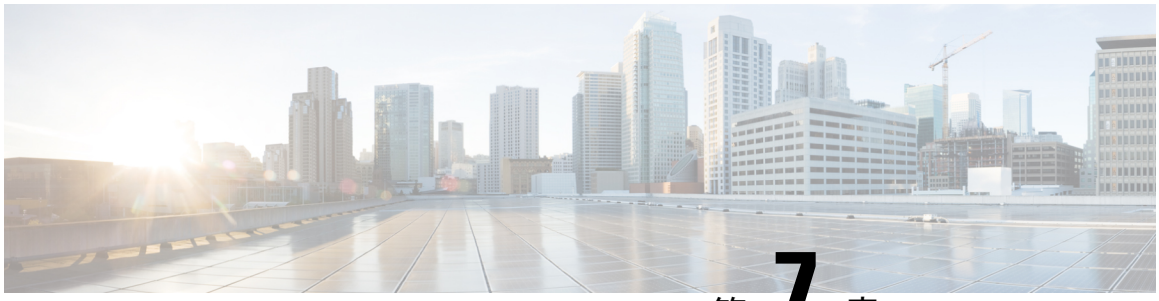
# SDM テンプレートに関する追加情報

**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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# SDM テンプレートの設定の機能履歴と情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| SDM テンプレート | Cisco IOS XE Everest 16.6.1 | 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。 |



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 （209 ページ）
- システム メッセージ ログの設定方法 （212 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 （221 ページ）
- システム メッセージ ログの設定例 （221 ページ）
- システム メッセージ ログに関する追加情報 （222 ページ）
- システム メッセージ ログの機能履歴と情報 （222 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。。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。スタンドアロン スイッチ、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。

（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 *17:* システム ログ メッセージの要素**

| 要素 | 説明 |
|---|---|
| *seq no:* | **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。 |
| *timestamp* formats:<br>*mm/dd h h:mm:ss*<br>または<br>*hh:mm:ss*（短時間）<br>または<br>*d h*（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。 |
| *facility* | メッセージが参照する機能（SNMP、SYS など）です。 |
| *severity* | メッセージの重大度を示す 0 ～ 7 の 1 桁のコードです。 |

| 要素 | 説明 |
|---|---|
| *MNEMONIC* | メッセージを一意に示すテキスト ストリングです。 |
| *description* | 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 ロギングの設定

**表 *18* : デフォルトのシステム メッセージ ロギングの設定**

| 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 ロギング | 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 |
| 同期ロギング | ディセーブル |
| ロギング 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**<br>例：<br>`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*]<br>例：<br>`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。<br>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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | （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ッサ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。 |
| ステップ 3 | **logging** *host*<br>例：<br>`Device(config)# logging 125.1.1.100` | UNIX Syslog サーバ ホストにメッセージを保存します。<br>*host* 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。<br>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。 |
| ステップ 4 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **terminal monitor**<br>例：<br>`Device# terminal monitor` | 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。<br>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。 |

# 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**<br>例：<br>`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*]<br>例：<br>`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。<br>• **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。<br>• **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。<br>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。<br>line vty 0 15<br>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。<br>line vty 2 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | 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。 |
| ステップ 3 | **logging synchronous** [**level** [*severity-level* \| **all**] \| **limit** *number-of-buffers*]<br>例：<br>`Device(config)# logging synchronous level 3 limit 1000` | メッセージの同期ロギングをイネーブルにします。<br>• （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。<br>• （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。<br>• （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 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |

# メッセージ ロギングのディセーブル化

メッセージロギン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、メッセージロギングをイネーブルにする必要があります。メッセージ ロギングがイネーブルの場合、ログ メッセージはロギング プロセスに送信されます。ロギングプロセスは、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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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 | 次のいずれかのコマンドを使用します。<br>• **service timestamps log uptime**<br>• **service timestamps log datetime**[**msec** \| **localtime** \| **show-timezone**]<br>例：<br>`Device(config)# service timestamps log uptime`<br>または<br>`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。<br>• **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。<br>• **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**<br>例：<br>`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**<br>例：<br>`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 sequence-numbers**<br>例：<br>`Device(config)# service` | シーケンス番号をイネーブルにします。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | **sequence-numbers** | |
| ステップ 3 | **end**<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`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。<br>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*<br>例：<br>`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。<br>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。 |
| ステップ 4 | **logging trap** *level*<br>例：<br>`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。<br>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**<br>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` | |

# 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**<br>例：<br>`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*<br>例：<br>`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。<br>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*<br>例：<br>`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。<br>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**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |

# 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** | /etc/syslog.conf ファイルに次の行を追加します。<br>**例：**<br>**local7.debug /usr/adm/logs/***cisco.log* | • **local7**：ロギング機能を指定します。<br>• **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** | UNIX シェルプロンプトに次のコマンドを入力します。<br>**例：**<br>`$ touch /var/log/cisco.log`<br>`$ chmod 666 /var/log/cisco.log` | 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。 |
| **ステップ 3** | Syslog デーモンに新しい設定を認識させます。<br>**例：**<br>`$ kill -HUP `cat /etc/syslog.pid`` | 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。 |

# 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 |
|---|---|
| **show archive log config** {**all** \| **number** [*end-number*] \| **user** *username* [ **session** *number*] *number* [*end-number*] \| **statistics**} [**provisioning**] | 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-channel1, 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-channel1, 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)
```

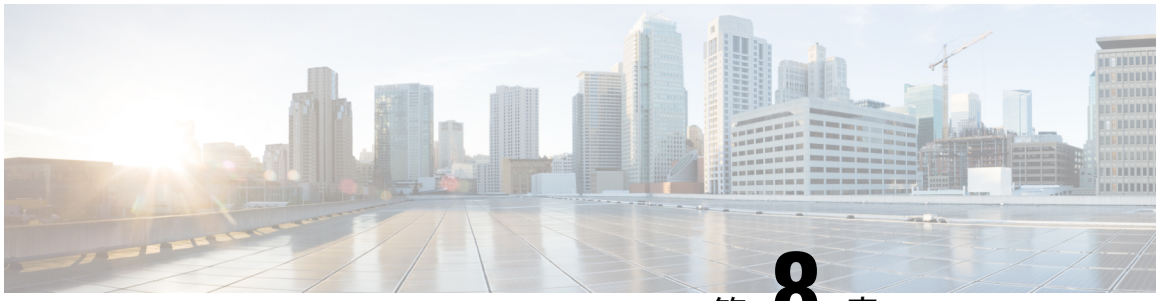
# システム メッセージ ログに関する追加情報

**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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# システム メッセージ ログの機能履歴と情報

| リリース | 変更内容 |
|---|---|
| Cisco IOS XE Everest 16.6.1 | この機能が導入されました。 |



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 （223 ページ）
- オンライン診断の設定方法 （228 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 （234 ページ）
- オンライン診断のコンフィギュレーション例 （234 ページ）
- オンライン診断に関する追加情報 （236 ページ）
- オンライン診断設定の機能情報 （236 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、個別のハードウェアコンポーネントを確認して、データパスおよび制御信号を検証するパケットスイッチングテストが含まれます。

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、ヘルスモ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。ヘルスモニタリングは、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。ヘルスモ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）テスト

（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ます。スイッチをリロードするコマンドを実行すると、コンフィギュレーションを保存するかどうかを尋ねられます。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。

ここでは、GOLD テストについて説明します。

**TestGoldPktLoopback**

この GOLD パケットループバックテストは、MAC レベルのループバック機能を検証します。このテストでは、ハードウェアで Unified Access Data Plane（UADP; ユニファイドアクセスデータ プレーン）ASIC によってサポートされる GOLD パケットが送信されます。このパケットは MAC レベルでループバックし、保存されているパケットと照合されます。

| 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 | 要件に従ってこのオンデマンドテストを実行します。 |
| デフォルト | オフ |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |
| ハードウェア サポート | スーパバイザおよびラインカード |

**TestFantray**

このテストは、すべてのファンモジュールが挿入され、ボード上で正しく動作していることを検証します。

| 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 |

| 属性 | 説明 |
|---|---|
| 推奨事項 | ファンモジュールに問題が発生した場合は、ヘルスモニタリングテストとしてこれを実行します。 |
| デフォルト | オン |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |
| ハードウェア サポート | スーパバイザ |

**TestPhyLoopback**

この PHY ループバックテストは、PHY レベルのループバック機能を検証します。このテストでは、PHY レベルでループバックし、保存されているパケットと照合されるパケットが送信されます。ヘルスモニタリングテストとして実行することはできません。

| 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 | ディスラプティブ |
| 推奨事項 | 外部コネクタへのリンクがダウンしている場合は、このオンデマンドテストを実行してリンクの正常性を確認します。 |
| デフォルト | オフ |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |
| ハードウェア サポート | スーパバイザおよびラインカード |

**TestThermal**

このテストは、デバイスセンサーからの温度の読み取り値を検証します。

| 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。これはオンデマンドテストとして実行し、管理者がダウン状態の場合はヘルスモニタリングテストとして実行します。 |

| 属性 | 説明 |
|---|---|
| デフォルト | オン |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |
| ハードウェア サポート | スーパバイザおよびラインカード |

**TestScratchRegister**

このスクラッチ登録テストは、レジスタに値を書き込み、これらのレジスタからその値を読み取ることで、ASIC の正常性をモニタします。

| 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。このテストは、レジスタに値を書き込むタスクが失敗した場合に実行します。これは、ヘルスモニタリングテストとしても、オンデマンドテストとしても実行できます。 |
| デフォルト | オン |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |
| ハードウェア サポート | スーパバイザおよびラインカード |

**TestPoe**

このテストは、PoE コントローラ機能をチェックします。

| 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 | これは PoE ヘルスモニタリングテストとして実行します。 |
| デフォルト | オフ |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |

| 属性 | 説明 |
|---|---|
| ハードウェア サポート | ラインカード |

**TestMemory**

この詳細な ASIC メモリテストは、通常のスイッチ動作中に実行します。このテストでは、スイッチはメモリの組み込み自己診断テストを使用します。メモリテストでは、テスト後にスイッチを再起動する必要があります。

| 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 | 非常にディスラプティブです。 |
| 推奨事項 | このオンデマンドテストは、システムでメモリ関連の問題が発生した場合にのみ実行します。テスト対象のスーパバイザエンジンをリロードしない場合は、このテストを実行しないでください。 |
| デフォルト | オフ |
| 最初のリリース | Cisco IOS XE Everest 16.6.1 |
| 修正処置 | – |
| ハードウェア サポート | スーパバイザ |

**TestUnusedPortLoopback**

このテストでは、管理ダウンポートの PHY レベルのループバック機能を検証します。このテストでは、PHY レベルのループバックにパケットが送信され、保存されているパケットと照合されます。

| 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 | これは、ヘルスモニタリングテストとしても、オンデマンドテストとしても実行できます。 |
| デフォルト | オフ |
| 最初のリリース | Cisco IOS XE Fuji 16.9.1 |
| 修正処置 | ポートのテストが失敗した場合に、syslog メッセージを表示します。 |

| 属性 | 説明 |
|---|---|
| ハードウェア サポート | スーパバイザおよびラインカード |

**TestPortTxMonitoring**

このテストは、接続されたインターフェイスの送信カウンタをモニタします。接続されたポートがパケットを送信できるかどうかを確認します。このテストは、150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。これは、ヘルスモニタリングテストとしても、オンデマンドテストとしても実行できます。 |
| デフォルト | オン |
| 最初のリリース | Cisco IOS XE Fuji 16.9.1 |
| 修正処置 | ポートのテストが失敗した場合に、syslogメッセージを表示します。 |
| ハードウェア サポート | スーパバイザおよびラインカード |

# オンライン診断の設定方法

ここでは、オンライン診断設定を構成するさまざまな手順について説明します。

## 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、テストプロセスの途中停止はできません。

手動でオンライン診断テストを開始するには、**diagnostic start switch** 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 | **diagnostic start module** *number* **test** {*name* \| *test-id* \| *test-id-range* \| **all** \| **basic** \| **complete** \| **minimal** \| **non-disruptive** \| **per-port**} | 診断テストを開始します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | 例：<br><br>`Device# diagnostic start module 2 test basic` | **switch** *number* 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。<br><br>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。<br>• *name*：テストの名前を入力します。<br>• *test-id*：テストのID番号を入力します。<br>• *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。<br>• **all**：すべてのテストを開始します。<br>• **basic**：基本テストスイートを開始します。<br>• **complete**：完全なテストスイートを開始します。<br>• **minimal**：最小限のブートアップテストスイートを開始します。<br>• **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。<br>• **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、**diagnostic schedule switch** 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 | **configure terminal**<br>例：<br>`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 | **diagnostic schedule module** *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*}<br>例：<br>`Device(config)# diagnostic schedule module 3 test 1-5 on July 3 2013 23:10` | 特定日時のオンデマンド診断テストをスケジュールします。<br>**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。<br>スケジュール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。<br>• *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。<br>• *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。<br>• *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。<br>• **all**：すべてのテスト ID。<br>• **basic**：基本的なオンデマンドの診断テストを開始します。<br>• **complete**：完全なテストスイートを開始します。<br>• **minimal**：最小限のブートアップテストスイートを開始します。<br>• **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。<br>• **per-port**：ポート単位のテストスイートを開始します。<br>テストは次のようにスケジュールできます。<br>• 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 |
|---|---|---|
| | | • 特定日時：**on** *mm dd yyyy hh:mm* パラメータを使用します。<br>• 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いくつかのテストでのみイネーブルであり、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**<br>**例**：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>**例**：<br>`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 module***number* **test** {*name* \| *test-id* \| *test-id-range* \| **all**} *hh:mm:ss milliseconds day*<br>**例**：<br>`Device(config)# diagnostic monitor interval module 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。<br>**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。<br>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 |
|---|---|---|
| | | • *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。<br>• *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。<br>• *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。<br>• **all**：すべての診断テスト。<br>間隔を指定する場合は、次のパラメータを設定します。<br>• *hh:mm:ss*：モニタリング間隔（時間、分、秒）。指定できる範囲は *hh* が 0 ~ 24、*mm* および *ss* が 0 ~ 60 です。<br>• *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。<br>• *day*：モニタリング間隔（日数）。指定できる範囲は 0 ～ 20 です。 |
| ステップ 4 | **diagnostic monitor syslog**<br>例：<br>`Device(config)# diagnostic monitor syslog` | （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。 |
| ステップ 5 | **diagnostic monitor threshold module** *number number* **test** {*name* \| *test-id* \| *test-id-range* \| **all**} **failure count** *count*<br>例：<br>`Device(config)# diagnostic monitor threshold module 2 test 1 failure count 20` | （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。<br>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。<br>• *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。<br>• *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 |
|---|---|---|
| | | • *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。<br>• **all**：すべての診断テスト。<br>失敗しきい値 *count* に指定できる範囲は 0 ～ 99 です。 |
| ステップ 6 | **diagnostic monitor module***number* **test** {*name* \| *test-id* \| *test-id-range* \| **all**}<br>例：<br>`Device(config)# diagnostic monitor module 2 test 1` | 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。<br>**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。<br>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。<br>• *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。<br>• *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。<br>• *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。<br>• **all**：すべての診断テスト。 |
| ステップ 7 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 8 | **show diagnostic** { **content** \| **post** \| **result** \| **schedule** \| **status** \| **switch** } | （任意）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。 |
| ステップ 9 | **show running-config**<br>例：<br>`Device# show running-config` | （任意）入力を確認します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 10 | **copy running-config startup-config**<br>例：<br>`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

デバイスまたはデバイススタックに設定されているオンライン診断テストを表示し、この表に示す **show** 特権 EXEC コマンドを使用してテスト結果を確認することができます。

表 *19 :* 診断テストの設定および結果用のコマンド

| コマンド | 目的 |
|---|---|
| **show diagnostic content module** [*number* \| **all**] | スイッチに対して設定されたオンライン診断を表示します。 |
| **show diagnostic status** | 現在実行中の診断テストを表示します。. |
| **show diagnostic result module** [*number* \| **all**] [**detail** \| **test** {*name* \| *test-id* \| *test-id-range* \| **all**} [**detail**]] | オンライン診断テストの結果を表示します。 |
| **show diagnostic schedule** [*number* \| **all**] | オンライン診断テストのスケジュールを表示します。 |
| **show diagnostic post** | POST 結果を表示します（出力は **show post** コマンドの出力と同じ）。 |
| **show diagnostic events** {**event-type** \| **module**} | テスト結果に基づいて、エラー、情報、警告などの診断イベントを表示します。 |
| **show diagnostic description module** [**number**] **test** { **name** \| **test-id** \| **all** } | 個々のテストまたはすべてのテストの結果について簡単な説明を表示します。 |

# オンライン診断のコンフィギュレーション例

次のセクションでは、オンライン診断の設定例を示します。

# 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 module 3 test DiagFanTest
```

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 module 3 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 module 1 test 1 failure count 50
Device(config)# diagnostic monitor interval module 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、特定の日時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)# schedule module 3 test TestThermal on October 2 2018 23:00
```

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 module 3 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 module 3 test all
TestGoldPktLoopback :
 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.

TestFantray :
 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.

TestPhyLoopback :
 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.

TestThermal :
 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.

TestScratchRegister :
 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.

TestMemory :
 This test runs the exhaustive ASIC memory test during normal
 switch operation. Switch utilizes mbist for this test. Memory test
 is very disruptive in nature and requires switch reboot after
 the test.
```

# 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。 | |

# オンライン診断設定の機能情報

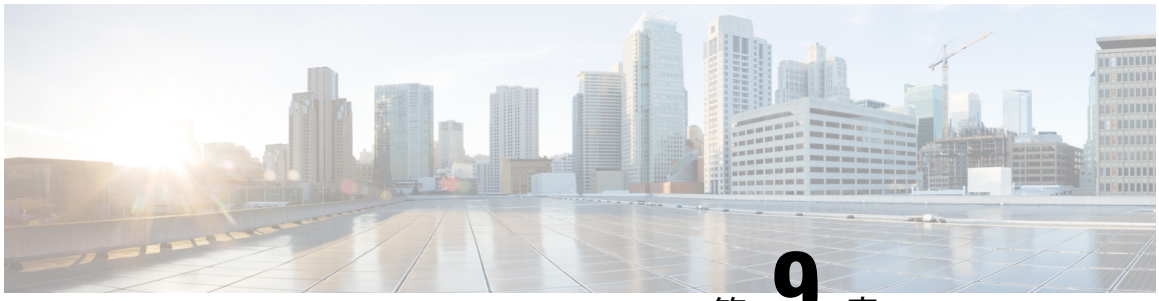
次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだ

けを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

**表 *20* : オンライン診断設定の機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| オンライン診断 | Cisco IOS XE Everest 16.6.1 | 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。 |



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 （239 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 （239 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 （240 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 （248 ページ）
- コンフィギュレーション ファイルの機能履歴と情報 （279 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（249 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。

# 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 （273 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ものですが、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になります。

```
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# copy scp: bootflash-1: vrf test-vrf
Address or name of remote host [10.1.2.3]?
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レーション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（273 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）は自動的には行われないため、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 （251 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 （248 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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** *file-url*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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）。<br>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。<br>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。<br>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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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**<br>例：<br>`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。<br>• **end**<br>• **^Z**<br>例：<br>`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。<br>（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**<br>例：<br>`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。<br>**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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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* ]<br>例：<br>`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* ]<br>例：<br>`Device# copy nvram:startup-config tftp: //server1/1stdir/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-confg
Write file tokyo-confg on host 172.16.2.155? [confirm] Y
Writing tokyo-confg!!! [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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`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。<br>• **copy system:running-config rcp:** [[[//[*username*@]*location* ]/*directory* ]/*filename* ]<br>• **copy nvram:startup-config rcp:** [[[//[*username*@]*location* ]/*directory* ]/*filename* ]<br>例：<br>`Device# copy system:running-config rcp:`<br>`//NetAdmin1@example.com/dir-files/file1` | • 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。<br>または<br>• 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-confg 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-confg
Write file runfile2-confg 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-confg]?
Write file start-confg 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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** *username*<br>例：<br>`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** *password*<br>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**<br>例：<br>`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。<br>• **copy system:running-config ftp:** [[[//[*username* [**:***password* ]**@**]*location*]/*directory* ]/*filename* ] または<br>• **copy nvram:startup-config ftp:** [[[//[*username* [**:***password* ]**@**]*location*]/*directory* ]/*filename* ]<br>例：<br>`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-confg 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-confg
Write file runfile-confg 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-confg]?
Write file start-confg 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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**<br>例：<br>`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-[n]:/directory/startup-config**<br>例：<br>`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-confg**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-confg system:running-config

Configure using tokyo-confg from 172.16.2.155? [confirm] Y

Booting tokyo-confg 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**<br>**例**：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>**例**：<br>`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** *username*<br>**例**：<br>`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**<br>例：<br>`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。<br>• **copy rcp:[[*username@location*]*directory*]*filename* system:running-config**<br>• **copy rcp:[[*username@location*]*directory*]*filename* nvram:startup-config**<br>例：<br>`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-confg 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-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![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-confg 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-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
```

```
device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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** *username*<br>例：<br>`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** *password*<br>例：<br>`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**<br>例：<br>`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** | 次のいずれかを実行します。<br>• **copy ftp:** [[[//[*username*[**:***password*]**@**]*location*] */directory* ]/*filename*]**system:running-config**<br>• **copy ftp:** [[[ *//username*[**:***password*]**@**]*location*]*/directory*]*/filename*]**nvram:startup-config**<br>例：<br>`Device# copy ftp:nvram:startup-config` | 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-confg 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-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![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-confg 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-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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。<br>• 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。<br>• **configure terminal**<br>例： | 新しいコンフィギュレーションを入力します。<br>• 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# configure terminal` | 「[buffer overflow - *file-size* /*buffer-size* bytes]。」 |
| ステップ 6 | **copy system:running-config nvram:startup-config**<br>例：<br>`Device(config)# copy system:running-config nvram:startup-config` | 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-confg system:running-config

Configure using tokyo-confg from 172.16.2.155? [confirm] y

Booting tokyo-confg 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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 nvram:startup-config** *flash-filesystem:filename*<br>例：<br>`Device# copy nvram:startup-config usbflash0:switch-config` | 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 | **configure terminal**<br>例：<br>`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 flash-filesystem:** *filename*<br>例：<br>`Device(config)# boot config usbflash0:switch-config` | 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 | **end**<br>例：<br>`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 | 次のいずれかを実行します。<br>• 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] 」<br>• **configure terminal**<br>例：<br>`Device# configure terminal` | 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 | **copy system:running-config nvram:startup-config**<br>例：<br>`Device(config)# copy system:running-config nvram:startup-config` | 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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:**}<br>例：<br>`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**<br>例：<br>`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* ]}<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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。<br>• **copy** *filesystem*:[*partition-number*:][*filename* ] **nvram:startup-config**<br>• **copy** *filesystem*:[*partition-number*:][*filename* ] **system:running-config**<br>例：<br>`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<br>• 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**<br>**例：**<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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*:<br>**例：**<br>`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* ]<br>**例：**<br>`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。<br>• 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... eeee
```

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device> enable` | |
| ステップ 2 | **configure terminal**<br>例：<br>`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** *username*<br>例：<br>`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** *password*<br>例：<br>`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**<br>例：<br>`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:**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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** *username*<br>例：<br>`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**<br>例：<br>`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:** [[[//[*username*@]*location*]/*directory*] /*bundle_name*] **flash:**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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:**<br>例：<br>`Device#`<br>`copy tftp://cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin 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**<br>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` | • 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**<br>例：<br>`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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>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 |
|---|---|---|
| | `Device# erase nvram` | （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-config**EXEC 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。 |

# 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **delete** *flash-filesystem*:*filename*<br>例：<br>`Device# delete usbflash0:myconfig` | 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。<br>（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。 |

# 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 ファイルをロードするように 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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*<br>例：<br>`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**<br>例：<br>`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*<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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-confg... [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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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* ]}<br>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。<br>• 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 |
|---|---|---|
| | `Device(config)# boot network tftp:hostfile1` | Cisco IOS ソフトウェアはデフォルトのファイル名の network-confg 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。<br>• 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す。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 | **service config**<br>例：<br>`Device(config)# service config` | 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 | **end**<br>例：<br>`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**<br>例：<br>`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。 |

## 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 host** {**ftp:**[[[//[*username* [**:***password* ]**@**]*location* ]/*directory* ]/*filename* ] \| **rcp:**[[[//[*username***@**]*location* ]/*directory* ]/*filename* ] \| **tftp:**[[[//*location* ]/*directory* ]/*filename* ] }<br>例：<br>`Device(config)# boot host tftp:hostfile1` | 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。<br>• 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-confg 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。<br>• 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 | **service config**<br>例：<br>`Device(config)# service config` | 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。 |
| ステップ 5 | **end**<br>例：<br>`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。 |

| | 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**<br>例：<br>`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**

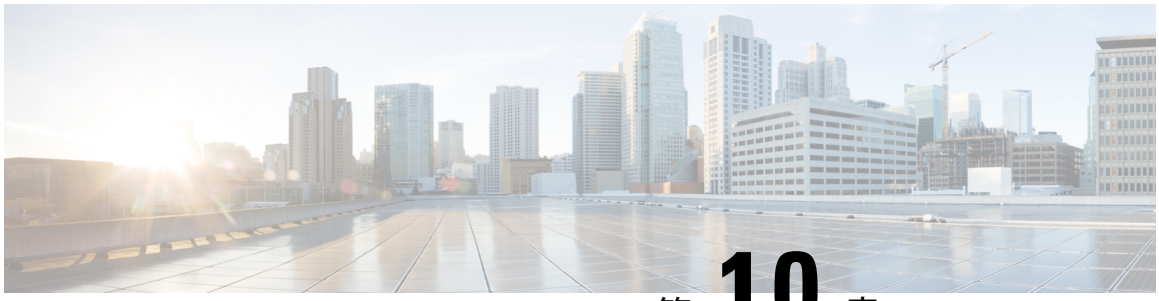
次に、hostfile1という名前のホストコンフィギュレーションファイルおよびnetworkfile1という名前のネットワーク コンフィギュレーションファイルをダウンロードするようにデバイスを設定する例を示します。デバイスはTFTPおよびブロードキャストアドレスを使用してファイルを取得します。

```
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 ファイルの機能履歴と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェアリリースでもサポートされます。

| リリース | 変更内容 |
|---|---|
| Cisco IOS XE Everest 16.6.1 | この機能が導入されました。 |



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 （281 ページ）
- Secure Copy に関する情報 （281 ページ）
- セキュア コピーの設定方法 （282 ページ）
- セキュア コピーの設定例 （285 ページ）
- セキュアコピーに関する追加情報 （286 ページ）
- セキュア コピーの機能情報 （286 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。
- Secure Copy Protocol（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 Rivest、Shamir、Adelman（RSA）キーのペアが必要です。

# Secure Copy に関する情報

Secure Copy 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。Secure Copy Protocol（SCP）は、セキュアシェル（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。

SCP は一連の Berkeley の r ツール（Berkeley 大学独自のネットワーキング 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。

## セキュアコピーのパフォーマンス向上

SSH 一括データ転送モードを使用すると、クライアントまたはサーバの容量で動作する SCP のスループットパフォーマンスを向上させることができます。このモードはデフォルトでは無効になっていますが、**ip ssh bulk-mode** 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有効にすることができます。

（注） このコマンドは、大きなファイルを転送する場合にのみ有効にし、ファイル転送の完了後に無効にすることをお勧めします。

# セキュア コピーの設定方法

ここでは、セキュアコピーの設定作業について説明します。

## 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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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**<br>例：<br>`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**<br>例： | ログイン時の AAA 認証を設定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# aaa new-model` | |
| ステップ 4 | **aaa authentication login** {**default** \| *list-name*} *method1* [ *method2...* ]<br>例：<br>`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*<br>例：<br>`Device(config)# username superuser privilege 2 password 0 superpassword` | ユーザ名をベースとした認証システムを構築します。<br>（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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device# debug ip scp` | （任意）SCP 認証問題を解決します。 |

# SSH サーバでのセキュアコピーのイネーブル化

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**<br>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` | 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**<br>例：<br>`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**<br>例：<br>`Device(config)# aaa new-model` | 認証、許可、アカウンティング（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**<br>例：<br>`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**<br>例：<br>`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 で特権 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*<br>例：<br>`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。<br>（注） *privilege-level* 引数に必要な最小値は 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*<br>例：<br>`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<br>例：<br>`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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 9 | **ip scp server enable**<br>例：<br>`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。 |
| ステップ 10 | **ip ssh bulk-mode**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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 コピーの設定例

次に、セキュアコピーの設定例を示します。

## 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、セキュアコピー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セキュアコピーのサーバ側の設定

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セキュアコピー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 |
|---|---|
| ★枠で囲まれた Technical Assistance の場合★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した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだ

けを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

**表 *21* : セキュア コピーの機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| セキュア コピー | Cisco IOS XE Everest 16.6.1 | 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。<br>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 enanle** |
| セキュアコピーのパフォーマンス向上 | Cisco IOS XE Amsterdam 17.2.1 | SSH 一括モードを使用すると、特定の最適化により、大量のデータ転送を伴うプロシージャのスループットパフォーマンスを向上できます。このモードは、**ip ssh bulk-mode** 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有効にすることができます。 |

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 （289 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 （290 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 （290 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 （293 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 （301 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（304 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と情報 （304 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。通常の環境では、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、追加する必要の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う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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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**<br>例：<br>`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 | **archive**<br>例：<br>`Device(config)# archive` | 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。 |
| ステップ 4 | **path** *url*<br>例：<br>`Device(config-archive)# path flash:myconfiguration` | 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。<br>（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。 |
| ステップ 5 | **maximum** *number*<br>例：<br>`Device(config-archive)# maximum 14` | （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。<br>• *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 |
|---|---|---|
| | | （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。 |
| ステップ 6 | **time-period** *minutes*<br>例：<br>`Device(config-archive)# time-period 1440` | （任意）Cisco IOS 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。<br>• 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、*minutes* 引数により分単位で指定します。<br>（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。 |
| ステップ 7 | **end**<br>例：<br>`Device(config-archive)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 8 | **archive config**<br>例：<br>`Device# archive config` | 現在の実行設定ファイルを設定アーカイブに保存します。<br>（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。 |

# 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 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure replace** *target-url* [**nolock**] [**list**] [**force**] [**ignore case**] [**revert trigger** [**error**] [**timer** *minutes*] | **time** *minutes*] ]<br>例：<br>`Device# configure replace flash: startup-config time 120` | 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。<br>• *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）。<br>• **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。<br>• **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。<br>• **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 |
|---|---|---|
| | | 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）。<br>• **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。<br>• **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。<br>• **error** ：エラー時に元のコンフィギュレーションに戻します。<br>• **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。<br>• **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。 |
| ステップ 3 | **configure revert** { **now** \| **timer** {*minutes* \| **idle** *minutes*} }<br>例：<br>`Device# configure revert now` | （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。<br>• **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 |
|---|---|---|
| | | • **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。<br>• 元に戻す時間を分単位で新たに指定するには、*minutes* 引数を **timer** キーワードとともに使用します。<br>• 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**<br>例：<br>`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。<br>（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**<br>例：<br>`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ったとします。**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

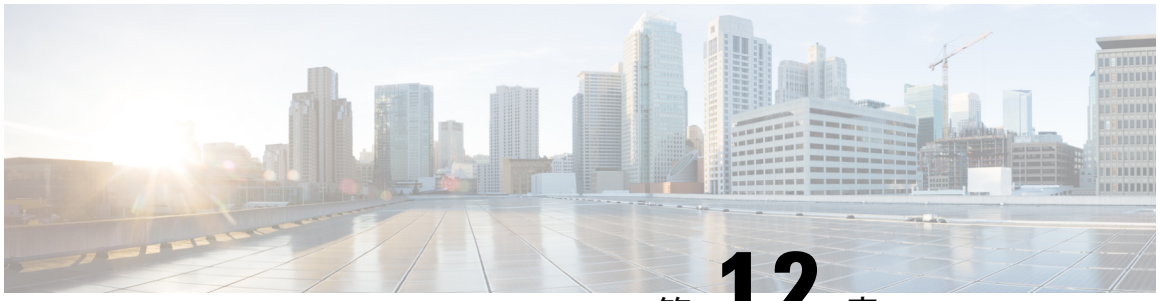
**関連資料**

| 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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# 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と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

| リリース | 変更内容 |
|---|---|
| Cisco IOS XE Everest 16.6.1 | この機能が導入されました。 |



第 12 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 （305 ページ）
- 初期設定へのリセット実行の制限事項 （305 ページ）
- 初期設定へのリセットの実行に関する情報 （306 ページ）
- 初期設定へのリセットの実行方法 （307 ページ）
- 初期設定へのリセットを実行するための設定例 （308 ページ）
- 初期設定へのリセットの実行に関する追加情報 （312 ページ）
- 初期設定へのリセットに関する機能情報 （312 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の実行に関する情報

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関連（FIPS 関連）のキーなどのクレデンシャルが含まれます。NIST SP 800-88 Rev. 1 で説明されているように、消去は clear メソッドと一致します。

初期設定へのリセットが完了すると、デバイスはデフォルトのライセンス設定に戻ります。

初期設定へのリセットプロセスは、次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。BAUD rate 環境変数は、初期設定へのリセット後にデフォルト値に戻ります。BAUD rate と console speed が常に同じであることを確認してください。同じでない場合、コンソールは応答しなくなります。

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 *22*：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 | — |

| 消去されるデータ | 保持されるデータ |
|---|---|
| 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 | |
| ユーザが追加した ROMmon 変数 | — |
| ライセンス | — |

# 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**<br>**例**：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>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** | • スタンドアロンデバイスの場合：<br>**factory-reset** {**all** ［**secure 3-pass**］| **config** | **boot-vars**}<br>• Cisco StackWise Virtual 対応デバイスの場合：<br>**factory-reset** {**all** ［**secure 3-pass**］| **config** | **boot-vars** | **switch** {*switch-number* | **all** {**all** ［**secure 3-pass**］| **config** | **boot-vars**}}<br>**例**：<br>`Device# factory-reset all`<br>または<br>`Device# factory-reset switch 1 all config` | デバイスを出荷時の設定にリセットします。<br>**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。<br>次のオプションを使用できます。<br>• **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。このオプションを使用することを推奨します。<br>• **secure 3-pass**：3-pass 上書きでデバイスからすべての内容を消去します。<br>• Pass 1：すべてのアドレス可能な場所を 2 進数のゼロで上書きします。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • Pass 2：すべてのアドレス可能な場所を2進数の1で上書きします。<br>• Pass 3：すべてのアドレス可能な場所をランダムビットパターンで上書きします。<br>（注） このオプションは、他のオプションの実行にかかる時間の約3倍の時間がかかります。<br>• **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。<br>• **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。<br>• **switch** {*switch-number* \| **all**}:<br>• *switch-number*：スイッチ番号を指定します。指定できる範囲は1 ～ 16 です。<br>• **all**：スタック内のすべてのスイッチを選択します。<br>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を実行するための設定例

次に、スタンドアロンスイッチで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]
```

次に、Cisco StackWise Virtual 対応デバイスで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 switch 2 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]
Switch#
*Sep 23 18:10:42.739:  Successfully sent switch reload message for switch num: 2 and
reason Factory Reset
*Sep 23 18:10:42.740: %STACKMGR-1-RELOAD: Chassis 2 R0/0: stack_mgr: Reloading due to
reason Factory Reset
*Sep 23 18:10:43.158: NGWC_FACTORYRESET: Switch 2, cmd: reset-all success

Original standby Switch 2:
Chassis 2 reloading, reason - Factory Reset
Sep 23 18:11:03.199: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: process
 exit with reload fru code


Enabling factory reset for this reload cycle
 Switch booted with tftp://172.19.72.26/tftpboot/thpaliss/trial.bin
% FACTORYRESET - Started Cleaning Up...

% FACTORYRESET - Unmounting flash1
% FACTORYRESET - Cleaning Up flash1
% FACTORYRESET - In progress.. please wait for completion...

% FACTORYRESET - write zero...
% FACTORYRESET - finish erase

Creating filesystem with 2790400 4k blocks and 697632 inodes
Filesystem UUID: 6a8ec2fb-4602-41b3-9c5c-ed59039d7480
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

% FACTORYRESET - Mounting Back flash1
% FACTORYRESET - Handling Mounted flash1
```

```
% FACTORYRESET - Factory Reset Done for flash1

% FACTORYRESET - Unmounting flash2
% FACTORYRESET - Cleaning Up flash2
% FACTORYRESET - In progress.. please wait for completion...

% FACTORYRESET - write zero...
% FACTORYRESET - finish erase

Creating filesystem with 409600 4k blocks and 102544 inodes
Filesystem UUID: e2f2280f-245a-4232-b0a8-edbf590a3107
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

% FACTORYRESET - Mounting Back flash2
% FACTORYRESET - Handling Mounted flash2
% FACTORYRESET - Factory Reset Done for flash2

% FACTORYRESET - Unmounting flash3
% FACTORYRESET - Cleaning Up flash3
% FACTORYRESET - In progress.. please wait for completion...

% FACTORYRESET - write zero...
% FACTORYRESET - finish erase

Creating filesystem with 131072 1k blocks and 32768 inodes
Filesystem UUID: 3c548955-16f5-4db5-a1c3-9a956248ccac
Superblock backups stored on blocks:
 8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

% FACTORYRESET - Mounting Back flash3
% FACTORYRESET - Handling Mounted flash3
% FACTORYRESET - Factory Reset Done for flash3

% FACTORYRESET - Unmounting flash7
% FACTORYRESET - Cleaning Up flash7
% FACTORYRESET - In progress.. please wait for completion...

% FACTORYRESET - write zero...
% FACTORYRESET - finish erase

Creating filesystem with 514811 4k blocks and 128768 inodes
Filesystem UUID: 9fe5a9db-263e-4303-825f-78ce815835c2
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

% FACTORYRESET - Mounting Back flash7
% FACTORYRESET - Handling Mounted flash7
% FACTORYRESET - Factory Reset Done for flash7
% FACTORYRESET - Lic Clean UP
% FACTORYRESET - Lic Clean Successful...
% FACTORYRESET - Clean Up Successful...
```

```
watchdog: watchdog0: watchdog did not stop!
systemd-shutdown[1]: Failed to parse (null): No such file or directory
systemd-shutdown[1]: Failed to deactivate swaps: No such file or directory
```

次に、スタック構成デバイスで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 switch all 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]
Chassis 1 reloading, reason - Factory Reset

 Protection key not found
9300L#Oct 25 09:53:05.740: %PMAN-5-EXITACTION: F0/0: pvp: Process manager is exiting:
reload fp action requested
Oct 25 09:53:07.277: %PMAN-5-EXITACTION:vp: Process manager is exiting: rp processes
exit with reload switch code


Enabling factory reset for this reload cycle
 Switch booted with
tftp://10.5.40.45/cat9k_iosxe.BLD_POLARIS_DEV_LATEST_20191007_224933_V17_2_0_21_2.SSA.bin

 Switch booted via
//10.5.40.45/cat9k_iosxe.BLD_POLARIS_DEV_LATEST_20191007_224933_V17_2_0_21_2.SSA.bin
% FACTORYRESET - Started Cleaning Up...

% FACTORYRESET - Unmounting sd1
% FACTORYRESET - Cleaning Up sd1 [0]
% FACTORYRESET - erase In progress.. please wait for completion...
% FACTORYRESET - write zero...
% FACTORYRESET - finish erase

% FACTORYRESET - Making File System sd1 [0]
Discarding device blocks: done
Creating filesystem with 409600 4k blocks and 102544 inodes
Filesystem UUID: fcf01664-7c6f-41ce-99f0-6df1d941701e
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

% FACTORYRESET - Mounting Back sd1 [0]
% FACTORYRESET - Handling Mounted sd1
% FACTORYRESET - Factory Reset Done for sd1

% FACTORYRESET - Unmounting sd3
```

```
% FACTORYRESET - Cleaning Up sd3 [0]
% FACTORYRESET - erase In progress.. please wait for completion...
% FACTORYRESET - write zero...


Chassis 2 reloading, reason - Factory Reset
Dec 12 01:02:12.500: %PMAN-5-EXITACTION: F0/0: pvp: Process manager is exiting: reload
fp action requested
De
Enabling factory reset for this reload cycle
 Switch booted with
tftp://10.5.40.45/cat9k_iosxe.BLD_POLARIS_DEV_LATEST_20191007_224933_V17_2_0_21_2.SSA.bin

 Switch booted via
//10.5.40.45/cat9k_iosxe.BLD_POLARIS_DEV_LATEST_20191007_224933_V17_2_0_21_2.SSA.bin
% FACTORYRESET - Started Cleaning Up...
% FACTORYRESET - Unmounting sd1
% FACTORYRESET - Cleaning Up sd1 [0]
% FACTORYRESET - erase In progress.. please wait for completion...
% FACTORYRESET - write zero...


After this the switch will come to boot prompt. Then the customer has to boot the device
 from TFTP.
```

# 初期設定へのリセット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 |
|---|---|
| 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 |

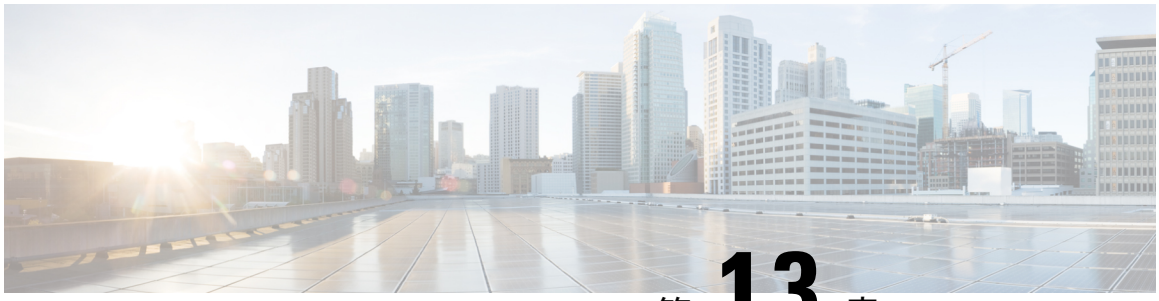
# 初期設定へのリセットに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

**表 *23 :* 初期設定へのリセットに関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| 工場出荷時の状態へのリセット | Cisco IOS XE Fuji 16.8.1a | この機能が導入されました。 |

| 機能名 | リリース | 機能情報 |
|---|---|---|
| | Cisco IOS XE Gibraltar 16.12.1 | 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。 |
| 3-pass 上書きによる初期設定へのリセット | Cisco IOS XE Amsterdam 17.2.1 | 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 | Cisco IOS XE Amsterdam 17.2.1 | 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。 |



第 13 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 （315 ページ）
- セキュア ストレージの有効化 （315 ページ）
- セキュア ストレージの無効化 （316 ページ）
- 暗号化のステータスの確認 （317 ページ）
- セキュアストレージの機能情報 （317 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。

デフォルトでは、この機能はハードウェアのトラストアンカーを備えたデバイスで有効です。この機能は、ハードウェアのトラストアンカーがないデバイスではサポートされません。

## 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有効になっています。この手順は、デバイスでセキュアストレージを無効にした後にのみ実行してください。

### 手順

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`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 | **service private-config-encryption**<br>例：<br>`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。 |
| ステップ 3 | **end**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 4 | **write memory**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`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**<br>例：<br>`Device(config)# end` | 特権 EXEC モードに戻ります。 |
| ステップ 4 | **write memory**<br>例：<br>`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。 |

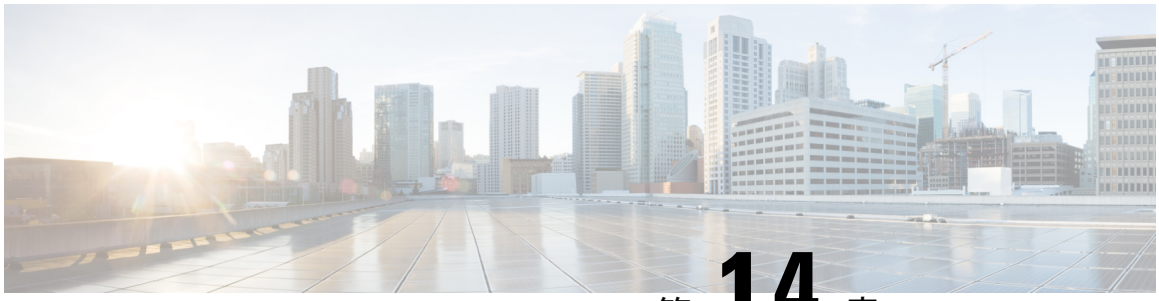
# 暗号化のステータスの確認

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

| リリース | 変更内容 |
|---|---|
| Cisco IOS XE Fuji 16.8.1a | この機能が導入されました。 |



第 14 章

# BIOS 保護


- BIOS 保護の概要 （319 ページ）
- ROMMON アップグレード （319 ページ）
- BIOS 保護の機能情報と情報 （321 ページ）


## BIOS 保護の概要

BIOS 保護機能により、ゴールデン ROMMON イメージの書き込み保護とセキュアアップグレードが有効になります。ROMMON は、デバイスの電源を投入または再起動したときに、ハードウェアを初期化して Cisco IOS XE ソフトウェアイメージをブートするブートストラッププログラムです。ファームウェア障害を解決するか、新しい機能をサポートするには、ROMMON のアップグレードが必要になることがあります。通常、ROM モニタのアップグレードはまれで、Cisco IOS XE ソフトウェアのアップグレードごとには必要ありません。

BIOS 保護機能がないと、ソフトウェアのアップグレード中に悪意のあるコードによってゴールデン ROMMON が破損する可能性があります。

## ROMMON アップグレード

ROMMON イメージは、プライマリ ROMMON およびゴールデン ROMMON として SPI フラッシュデバイスに保存されます。プライマリ ROMMON は、デバイスの電源がオンになるか再起動されるたびに起動します。プライマリ ROMMON が破損した場合、デバイスはゴールデン ROMMON を使用して IOS XE ソフトウェアイメージを起動します。デバイスがプライマリ ROMMON から起動すると、ゴールデン ROMMON はロックされます。BIOS 保護を使用すると、ゴールデン ROMMON は書き込み保護され、フラッシュユーティリティのアップグレードメカニズムを使用してアップグレードすることができません。アクセスポリシーは、FPGA ファームウェアによって管理されます。FPGA は、ゴールデン ROMMON SPI フラッシュデバイスで許可されていない操作（書き込み、消去など）をブロックします。

（注） ゴールデンROMMONアップグレードは、セキュアブートFPGAアップグレードなしでは有効になりません。

プライマリROMMONは、デバイスの起動時に自動的にアップグレードされます。ゴールデンROMMONは、カプセルアップグレードを使用してアップグレードできます。プライマリFPGAは、デバイスの起動時に自動的にアップグレードされます。ゴールデンFPGAはアップグレードされません。

アップグレードプロセスはスタンドアロンシステムと高可用性システムで異なり、以下で説明します。

### スタンドアロンシステム

スタンドアロンデバイスでは、デバイスをインストールモードでアップグレードすると、デバイスの起動時にプライマリ ROMMON が自動的にアップグレードされます。ゴールデン ROMMON は、カプセルアップグレードを使用してアップグレードできます。

### 高可用性および StackWise Virtual システム

高可用性設定のデバイスでは、In-Service Software Upgrade（ISSU）を実行することを推奨します。FPGA のアップグレードは、ISSU の一部として行われます。

リロードを使用してインストールモードでアップグレードを実行する場合は、両方のスーパバイザを同時にリロードしないでください。スタンバイスーパバイザをROMMON状態にして、アクティブスーパバイザを起動します。各スーパバイザでROMMONアップグレードが完了すると、FPGA およびソフトウェアイメージがアップグレードされます。

スタンバイスーパバイザを起動し、スタンバイスーパバイザがアップグレードしてスタンバイホット状態になるようにします。

## カプセルアップグレード

カプセルアップグレードでは、ゴールデンROMMONをアップグレードするため、認証後にプライマリROMMONによって使用されるセキュアな更新カプセルが作成され、署名されます。セキュアな更新カプセルには、セキュアなフラッシュ証明書が必要です。セキュアなフラッシュ証明書はプロダクトキーを使用して作成され、プライマリROMMONイメージに追加されて更新カプセルの真正性が検証されます。カプセルは、セキュアなフラッシュ証明書とセキュアブート 16 MB フラッシュイメージを使用して作成され、署名されます。

デバイスが起動すると、プライマリ ROMMON がゴールデン ROMMON のカプセルアップグレードをトリガーします。ゴールデンROMMONのカプセルアップグレードを実行するには、特権 EXEC モードで **upgrade rom-monitor capsule golden switch** コマンドを使用します。

カプセルアップグレードでは、次のプロセスが実行されます。

- デバイスは、セキュアブートFPGAアップグレードが有効になっているかどうかを確認します。有効でない場合、プロセスは終了します。

- デバイスは、ブートローダー保護が有効になっているかどうかを確認します。有効でない場合は、プライマリ ROMMON、ゴールデン ROMMON、およびプライマリ FPGA のワンタイムアップグレードが開始されます。
- ブートローダー保護がすでにアクティブになっている場合、IOS はセキュアな更新カプセルをブートフラッシュにコピーし、デバイスを再起動します。
- デバイスが再起動すると、アップグレードを実行するためにセキュアな更新カプセルが選択されます。

# BIOS 保護の機能情報と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、https://www.cisco.com/go/cfnに進みます。Cisco.com のアカウントは必要ありません。

**表 *24 : BIOS* 保護の機能情報**

| 機能名 | リリース | 変更内容 |
|---|---|---|
| BIOS 保護 | Cisco IOS XE Gibraltar 16.12.1 | BIOS 保護機能により、ゴールデン ROMMON イメージの書き込み保護とセキュアアップグレードが有効になります。 |
| カプセルアップグレード | Cisco IOS XE Amsterdam 17.1.1 | **upgrade rom-monitor capsule switch active** コマンドを使用したゴールデン ROMMON のカプセルアップグレードのサポートが有効になりました。 |

第 15 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 アップグレードについて （323 ページ）
- ソフトウェア メンテナンスの更新の管理方法 （324 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 （327 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 （332 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 （332 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

## 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

SMUタイプは、インストールされているSMUが対応するシステムに与える影響を示します。SMUがトラフィックに影響を与えない場合や、SMUによってデバイスの再起動、リロード、またはスイッチオーバーが発生する場合があります。リロードが必要かどうかを確認するには、**show install package flash**: *filename* コマンドを実行します。

ホットパッチを使用すると、SMU はアクティブ化後に有効になり、システムをリロードする必要がありません。SMUがコミットされると、リロードが繰り返されても変更が持続します。場合によっては、SMU でオペレーティングシステムのコールド（完全）リロードが必要になることがあります。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。コールドリロードが必要な場合、ユーザにはアクションを確認するプロンプトが表示されます。

# ソフトウェア メンテナンスの更新の管理方法

ここでは、SMU の管理に関する情報について説明します。

単一のコマンドまたは個別のコマンドを使用して 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。

# SMU パッケージのインストール

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 **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 | **enable**<br>例：<br>`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**]<br>例：<br>`Device# install add file flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.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 packages.conf ファイルに抽出します。<br>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）メンテナンス更新パッケージをコピーすることもできます。<br>（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。 |
| ステップ 3 | **exit**<br>例：<br>`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 パッケージの管理

手順

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`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*<br>例：<br>`Device# install add file flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin` | SMU パッケージをソースの場所からデバイスにコピーし（ソースの場所がリモートの場合）、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、`package/SMU.sta` 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。 |
| ステップ 3 | **install activate file flash:** *filename*<br>例：<br>`Device# install activate add file flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin` | 互換性チェックを実行し、パッケージをインストールして、パッケージのステータスの詳細を更新します。 |
| ステップ 4 | **install commit**<br>例：<br>`Device# install commit` | 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。アクティブ化の後で、システムの起動時、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。 |
| ステップ 5 | **install rollback to** {**base** \| **committed** \| **id** *commit-ID*}<br>例：<br>`Device# install rollback to committed` | デバイスを以前のインストール状態に戻します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 6 | **install deactivate file flash:** *filename*<br>例：<br>`Device# install deactivate file flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin` | アクティブパッケージを非アクティブ化し、パッケージのステータスを更新します。 |
| ステップ 7 | **install remove** {**file flash:** *filename* \| **inactive**}<br>例：<br>`Device# install remove file flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin` | 指定した SMU が非アクティブかどうかを確認し、非アクティブの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 8 | **show version**<br>例：<br>`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 9 | **show install summary**<br>例：<br>`Device# show install summary` | パッケージのインストールステータス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 の管理

（注）
- このセクションでは、ホットパッチ SMU の例を使用しています。

次に、SMU ファイルをフラッシュにコピーする例を示します。

```
Device# copy ftp://172.16.0.10//auto/ftpboot/user/
cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin

flash:
Destination filename
[cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin]?
Accessing ftp://172.16.0.10//auto/ftpboot/folder1/
cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin...
Loading /auto/ftpboot/folder1/
cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin from
172.16.0.10 (via GigabitEthernet0): !
```

```
[OK - 17668 bytes]
17668 bytes copied in 0.058 secs (304621 bytes/sec)
```

次に、メンテナンス更新プログラム 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_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin

install_add: START Mon Mar  5 21:48:51 PST 2018
install_add: Adding SMU

--- Starting initial file syncing ---
Info: Finished copying
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin to
the selected switch(es)
Finished initial file syncing

Executing pre scripts....

Executing pre scripts 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_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin Mon
 Mar  5 21:49:00 PST 2018
```

次に、SMU 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 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_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin
IMG   C    16.9.1.0.43131

--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

次に、追加した SMU パッケージ ファイルをアクティブ化する例を示します。

```
Device# install activate file
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin

install_activate: START Mon Mar  5 21:49:22 PST 2018
install_activate: Activating SMU
Executing pre scripts....

Executing pre scripts 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

SUCCESS: install_activate
/flash/cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin Mon
 Mar  5 21:49:34 PST 2018
```

次に、**show version** コマンドの出力例を示します。

```
Device# show version

Cisco IOS XE Software, Version BLD_POLARIS_DEV_LATEST_20180302_085005_2 - SMU-PATCHED
Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_IOSXE), Experimental Version
 16.9.20180302:
085957 [polaris_dev-/nobackup/mcpre/BLD-BLD_POLARIS_DEV_LATEST_20180302_085005 166]
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Fri 02-Mar-18 09:50 by mcpre
...
```

次に示すのは、**show install summary** コマンドが SMU 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。

```
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   U
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin
IMG   C    16.9.1.0.43131

--------------------------------------------------------------------------------
Auto abort timer: active on install_activate, time before rollback - 01:59:50
--------------------------------------------------------------------------------
```

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

```
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   U
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin
IMG   C    16.9.1.0.43131
```

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

```
Device# install commit
```

```
install_commit: START Mon Mar  5 21:50:52 PST 2018
install_commit: Committing SMU
Executing pre scripts....

Executing pre scripts 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_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin Mon
 Mar  5 21:51:01 PST 2018
```

次に示すのは、**show install summary** コマンドが、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を表示する場合の出力例です。

```
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_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin
IMG   C    16.9.1.0.43131

--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

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 committed

install_rollback: START Mon Mar  5 21:52:18 PST 2018
install_rollback: Rolling back SMU
Executing pre scripts....

Executing pre scripts done.

--- Starting SMU Rollback operation ---
Performing SMU_ROLLBACK on all members
  [1] SMU_ROLLBACK package(s) on switch 1
  [1] Finished SMU_ROLLBACK on switch 1
Checking status of SMU_ROLLBACK on [1]
SMU_ROLLBACK: Passed on [1]
Finished SMU Rollback operation

SUCCESS: install_rollback
/flash/cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin Mon
 Mar  5 21:52:30 PST 2018
```

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

```
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
--------------------------------------------------------------------------------
IMG   C    16.9.1.0.43131

--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

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
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin

install_deactivate: START Mon Mar  5 21:54:06 PST 2018
install_deactivate: Deactivating SMU
Executing pre scripts....

Executing pre scripts done.

--- Starting SMU Deactivate operation ---
Performing SMU_DEACTIVATE on all members
  [1] SMU_DEACTIVATE package(s) on switch 1
  [1] Finished SMU_DEACTIVATE on switch 1
Checking status of SMU_DEACTIVATE on [1]
SMU_DEACTIVATE: Passed on [1]
Finished SMU Deactivate operation

SUCCESS: install_deactivate
/flash/cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin Mon
 Mar  5 21:54:17 PST 2018
```

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

```
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   D
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin
IMG   C    16.9.1.0.43131

--------------------------------------------------------------------------------
Auto abort timer: active on install_deactivate, time before rollback - 01:59:50
--------------------------------------------------------------------------------
```

次に、デバイスから SMU を削除する例を示します。

```
Device# install remove file
flash:cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin

install_remove: START Mon Mar  5 22:03:50 PST 2018
install_remove: Removing SMU
```

```
Executing pre scripts....

Executing pre scripts done.

--- Starting SMU Remove operation ---
Performing SMU_REMOVE on all members
  [1] SMU_REMOVE package(s) on switch 1
  [1] Finished SMU_REMOVE on switch 1
Checking status of SMU_REMOVE on [1]
SMU_REMOVE: Passed on [1]
Finished SMU Remove operation

SUCCESS: install_remove
/flash/cat9k_iosxe.BLD_SMU_20180302_085005_TWIG_LATEST_20180306_013805.3.SSA.smu.bin Mon
 Mar  5 22:03:58 PST 2018
```

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

```
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
--------------------------------------------------------------------------------
IMG   C    16.9.1.0.43131

--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

# 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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# 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 Everest 16.6.3 | 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。 |
| Cisco IOS XE Fuji 16.9.1 | ホットパッチ | ホットパッチを使用すると、SMU はアクティブ化後に有効になり、システムをリロードする必要がありません。 |
| 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 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 [英語] からアクセスします。

第 16 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 （335 ページ）
- 使用可能なファイル システムの表示 （336 ページ）
- デフォルト ファイル システムの設定 （337 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 （338 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 （339 ページ）
- ディレクトリの作成 （340 ページ）
- ファイルのコピー （341 ページ）
- ファイルの作成、表示、および抽出 （342 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 （344 ページ）
- フラッシュ ファイル システムの機能履歴と情報 （345 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）。

```
Device# show file systems

File Systems:

       Size(b)       Free(b)      Type  Flags  Prefixes
             -             -    opaque     rw   system:
             -             -    opaque     rw   tmpsys:
*  11250098176    9694093312      disk     rw   bootflash: flash:
    1651314688    1232220160      disk     rw   crashinfo:
  118148280320  112084115456      disk     rw   disk0:
     189628416     145387520      disk     rw   usbflash0:
    7763918848    7696850944      disk     ro   webui:
             -             -    opaque     rw   null:
             -             -    opaque     ro   tar:
             -             -   network     rw   tftp:
      33554432      33532852     nvram     rw   nvram:
             -             -    opaque     wo   syslog:
             -             -   network     rw   rcp:
             -             -   network     rw   http:
             -             -   network     rw   ftp:
             -             -   network     rw   scp:
             -             -   network     rw   https:
             -             -    opaque     ro   cns:
```

表 *25 : 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 | ファイル システムのタイプです。<br>**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。<br>**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。<br>**nvram**：ファイル システムは NVRAM（不揮発性 RAM）デバイス用です。<br>**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。<br>**unknown**：ファイル システムのタイプは不明です。 |

| フィールド | 値 |
|---|---|
| Flags | ファイル システムの権限です。<br>**ro**：読み取り専用です。<br>**rw**：読み取りおよび書き込みです。<br>**wo**：書き込み専用です。 |
| Prefixes | ファイル システムのエイリアスです。<br>**crashinfo**：crashinfo ファイルです。<br>**disk0**：M2 SATA モジュール。<br>**flash:** ：フラッシュ ファイル システムです。<br>**ftp**：FTP サーバです。<br>**http**：HTTP サーバです。<br>**https**：セキュア HTTP サーバです。<br>**nvram:** ：NVRAM です。<br>**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。<br>**rcp**：Remote Copy Protocol（RCP）サーバです。<br>**scp**：Session Control Protocol（SCP）サーバです。<br>**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。<br>**tftp:** ：TFTP ネットワーク サーバです。<br>**usbflash0**：USB フラッシュ メモリです。<br>**xmodem:** ：XMODEM 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。<br>**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。 |

# 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 *26 :* 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  -rwx            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  -rwx         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**<br>**例**：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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:*<br>**例**：<br>`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。<br>*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。 |
| **ステップ 3** | **cd** *directory_name*<br>**例**：<br>`Device# cd new_configs` | 指定されたディレクトリへ移動します。<br>コマンド例では、*new_configs* 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**<br>**例**：<br>`Device# pwd` | 作業ディレクトリを表示します。 |
| **ステップ 5** | **cd**<br>**例**：<br>`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:*<br>**例**：<br>`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。<br>*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。 |
| **ステップ 2** | **mkdir** *directory_name*<br>**例**：<br>`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:*<br>**例**：<br>`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と、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** | **archive tar /create** *destination-url* **flash:** */file-url*<br>**例：**<br>`Device# archive tar /create tftp:172.20.10.30/saved. flash:/new-configs` | ファイルを作成し、そこにファイルを追加します。<br>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。<br>• ローカル フラッシュ ファイル システム構文<br>**flash:**<br>• FTP 構文<br>**ftp:**[[*//username*[*:password*]*@location*]*/directory*]*/filename*<br>• RCP 構文 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | **rcp**:[[//*username@location*]/*directory*]/*-filename*.<br>• TFTP 構文<br>**tftp**:[[//*location*]/*directory*]/*-filename*.<br>**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*<br>例：<br>`Device# archive tar /table flash: /new_configs` | ファイルの内容を表示します。<br>*source-url* 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。*-filename.* 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。<br>• ローカル フラッシュ ファイル システム構文<br>**flash:**<br>• FTP 構文<br>**ftp**:[[//*username*[:*password*][@*location*]/*directory*]/*-filename*.<br>• RCP 構文<br>**rcp**:[[//*username@location*]/*directory*]/*-filename*.<br>• TFTP 構文<br>**tftp**:[[//*location*]/*directory*]/*-filename*.<br>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...* ]<br>例： | 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。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device# archive tar /xtract tftp:/172.20.10.30/saved. flash:/new-configs` | *source-url* 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。*-filename.* 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。<br>• ローカル フラッシュ ファイル システム構文<br>**flash:**<br>• FTP 構文<br>**ftp**:[[//*username*[:*password*]@*location*]/*directory*]/*-filename.*<br>• RCP 構文<br>**rcp**:[[//*username*@*location*]/*directory*]/*-filename.*<br>• TFTP 構文<br>**tftp**:[[//*location*]/*directory*]/*-filename*.<br>**flash:**/*file-url* [*dir/file...*] 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、*dir/file...* 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。 |
| ステップ 4 | **more** [ /**ascii** \| /**binary** \| /**ebcdic**] /*file-url*<br>例：<br>`Device# more flash:/new-configs` | 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。 |

# 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

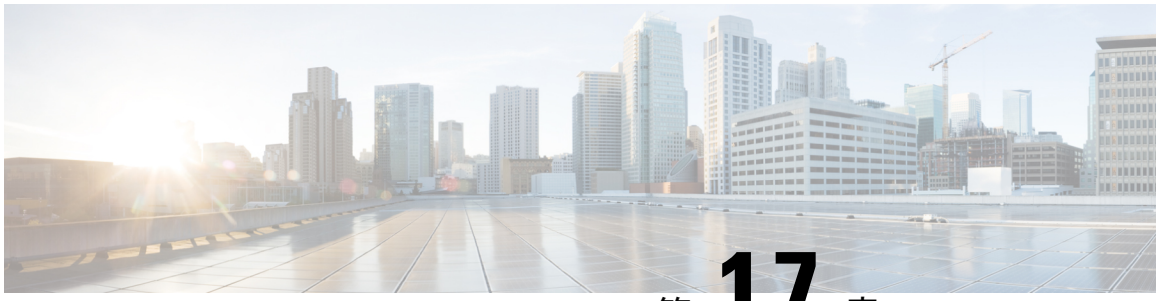
**関連資料**

| 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 | *Cisco IOS Configuration Fundamentals Command Reference* |

# フラッシュ ファイル システムの機能履歴と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

| リリース | 変更内容 |
|---|---|
| | この機能が導入されました。 |



第 17 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 （347 ページ）
- ラジオアクティブ トレースの概要 （348 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 （348 ページ）
- 条件付きデバッグのモニタリング （352 ページ）
- 条件付きデバッグの設定例 （353 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 （353 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と情報 （354 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の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**<br>例：<br>`Device> enable` | 特権 EXEC モードを有効にします。<br>• 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** {*mac-address*}<br>例：<br>`Device# debug platform condition mac bc16.6509.3314` | 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**<br>例：<br>`Device# debug platform condition start` | 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**<br>例：<br>`Device# show platform condition`<br>`Device# show debug` | 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**<br>例：<br>`Device# debug platform condition stop` | 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:*}]<br>例：<br>`# request platform software trace archive last 2 days` | （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**]<br>例：<br>`Device# show platform software trace message` | （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。<br>• **filter-binary** ：照合するモジュールをフィルタリングします。<br>• **level** ：トレース レベルを表示します。<br>• **message** ：トレース メッセージのリングの内容を表示します。<br>（注） デバイス上では次が可能です。<br>• 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。<br>• マージされたログでファイルを生成します。<br>• 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**<br>例：<br>`Device# clear platform condition all` | 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

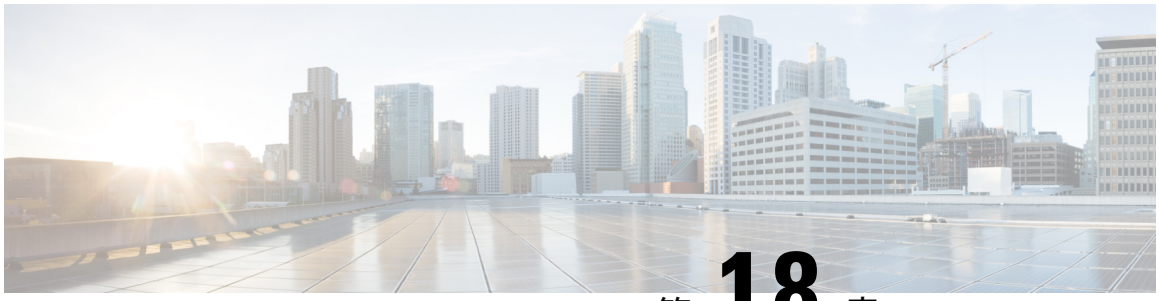
**関連資料**

| 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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# 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と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

| リリース | 変更内容 |
|---|---|
| Cisco IOS XE Everest 16.6.1 | この機能が導入されました。 |



第 18 章

# 同意トークン


- 同意トークンの制約事項 （355 ページ）
- 同意トークンに関する情報 （356 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 （356 ページ）
- 同意トークンの機能履歴と情報 （358 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。

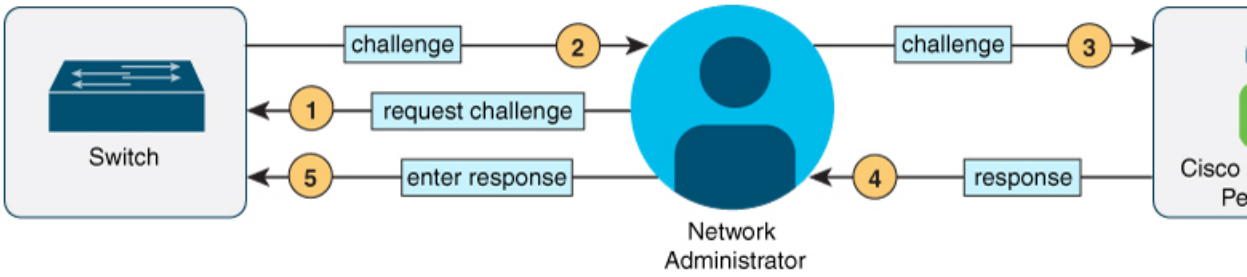
システムシェルへのアクセスを要求する場合は、認証を受ける必要があります。最初にコマンドを実行し、デバイスの同意トークン機能を使用してチャレンジを生成する必要があります。デバイスは、固有のチャレンジを出力として生成します。このチャレンジ文字列をコピーし、電子メールまたはインスタントメッセージでシスコ認定担当者に送信する必要があります。

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 *6*：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
[redacted]
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
[redacted]
% Consent token authorization success
*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 ー 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**：

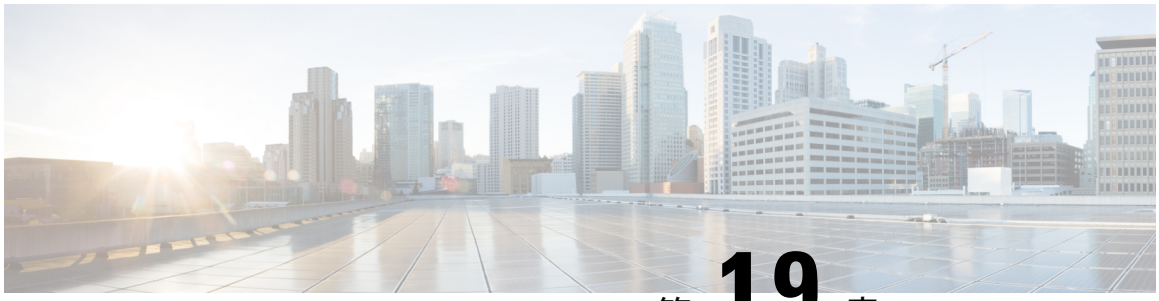
```
Device# request consent-token terminate-auth
% Consent token authorization termination success

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と情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

| リリース | 機能情報 |
|---|---|
| Cisco IOS XE Gibraltar 16.11.1 | この機能が導入されました。 |



第 19 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 インストレーション 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 （359 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 （369 ページ）
- パケット損失のトラブルシューティング （381 ページ）
- モジュールがオンラインでない場合のトラブルシューティング （382 ページ）
- インターフェイスの問題のトラブルシューティング （383 ページ）
- ワークステーションがネットワークにログインできない場合のトラブルシューティング （383 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 （384 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 （386 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 （391 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 （393 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と情報 （393 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（369 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と、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 （372 ページ）の項で説明する手順に従います。

# Power over Ethernet（PoE）ポート

Power over Ethernet（PoE）スイッチ ポートでは、回路に電力が供給されていないことをスイッチが検知した場合、接続している次のデバイスに電力が自動的に供給されます。

- シスコ先行標準受電デバイス（Cisco IP Phone や Cisco Aironet アクセス ポイントなど）
- IEEE 802.3af 準拠の受電装置
- IEEE 802.3at 準拠の受電装置

受電デバイスが PoE スイッチ ポートおよび AC 電源に接続されている場合、冗長電力として利用できます。受電装置が PoE ポートにだけ接続されている場合、受電装置には冗長電力は供給されません。

受電デバイスを検出すると、スイッチは受電デバイスの電力要件を判断し、受電デバイスへの電力供給を許可または拒否します。また、スイッチは消費電力をモニタリングおよびポリシングすることで、装置の電力の消費をリアルタイムに検知できます。

詳細については、『』の「Configuring PoE」の章を参照してください。*Interface and Hardware Component Configuration Guide (Catalyst 9400 Switches)*

PoE のさまざまなトラブルシューティング シナリオについては、Power over Ethernet（PoE）に関するトラブルシューティングのシナリオ （386 ページ）の項を参照してください。

## 電力消失によるポートの障害

PoE デバイスポートに接続され、AC 電源から電力が供給されている受電デバイス（Cisco IP Phone 7910 など）に AC 電源から電力が供給されない場合、そのデバイスは errdisable ステートになることがあります。errdisable ステートから回復するには、**shutdown** 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、**no shutdown** インターフェイスコマンドを入力します。デバイスで自動回復を設定し、errdisable ステートから回復することもできます。

デバイスの場合、**errdisable recovery cause loopback** および **errdisable recovery interval** *seconds* グローバル コンフィギュレーション コマンドは、指定した期間が経過したあと自動的にインターフェイスを errdisable ステートから復帰させます。

## 不正リンク アップによるポート障害

シスコ受電デバイスをポートに接続し、**power inline never** 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設定した場合は、不正リンクアップが発生し、ポートが errdisable ステートになることがあります。ポートを errdisable ステートから回復するには、**shutdown** および **no shutdown** インターフェイス コンフィギュレーション コマンドを入力します。

**power inline never** コマンドで設定したポートにシスコ受電デバイスを接続しないでください。

# 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 （379 ページ）の項を参照してください。

# レイヤ 2 Traceroute

レイヤ2 traceroute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への物理パスを識別できます。レイヤ 2 traceroute 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。トレースルートは、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トレース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トレース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 をディセーブル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の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パケット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 番めのルータは、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 番めのルータは、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 （392 ページ）に進み、IP traceroute プロセスの例を参照してください。

# Time Domain Reflector ガイドライン

Time Domain Reflector（TDR）機能を使用すると、ケーブル配線の問題を診断して解決できます。TDR 稼働時、ローカル デバイスはケーブルを介して信号を送信して、最初に送信した信号と反射された信号を比べます。

TDR は次のケーブル障害を検出します。

- ツイストペア ケーブルの導線のオープン、損傷、切断：導線がリモート デバイスからの導線に接続されていない状態。
- ツイストペア ケーブルの導線のショート：導線が互いに接触している状態、またはリモート デバイスからの導線に接触している状態。たとえば、ツイスト ペア ケーブルの一方の導線が、もう一方の導線にはんだ付けされている場合、ツイストペア ケーブルのショートが発生します。

  ツイストペアの導線の一方がオープンになっている場合、TDR はオープンになっている導線の長さを検出できます。

次の状況で TDR を使用して、ケーブル障害を診断および解決してください。

- デバイスの交換
- 配線クローゼットの設定
- リンクが確立できない、または適切に動作していない場合における、2 つのデバイス間の接続のトラブルシューティング

TDR の実行時、次の場合にデバイスは正確な情報をレポートします。

- ギガビット リンク用のケーブルが単線コア ケーブル
- オープンエンド ケーブルが未終端

TDR の実行時、次の場合にデバイスは正確な情報をレポートしません。

- ギガビット リンク用のケーブルがツイストペア ケーブルまたは連続接続された単線コア ケーブル
- リンクが 10 Mb または 100 Mb
- より線ケーブル
- リンク パートナーが Cisco IP Phone
- リンク パートナーが IEEE 802.3 に準拠していない

TDR の実行および結果の表示 （380 ページ）に移動し、TDR のコマンドを確認します。

# 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。デバッギン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。

**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。

# 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。

- スイッチ障害の場合：システム レポートは障害が発生したスイッチで生成されます。
- 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。

```
Switch#dir crashinfo:
Directory of crashinfo:/


23665 drwx 86016 Jun 9 2017 07:47:51 -07:00 tracelogs
11 -rw- 0 May 26 2017 15:32:44 -07:00 koops.dat
12 -rw- 4782675 May 29 2017 15:47:16 -07:00 system-report_1_20170529-154715-PDT.tar.gz
1651507200 bytes total (1519386624 bytes free)
```

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。

複雑なネットワークでは、システムレポートファイルの送信元を追跡することは困難です。システムレポートファイルが一意に識別できる場合、この作業は簡単になります。Cisco IOS XE Amsterdam 17.3.x リリース以降、システムレポートファイル名の前にホスト名が追加され、レポートが一意に識別できるようになります。

次の例では、ホスト名が先頭に追加されたシステムレポートファイルを表示します。

```
HOSTNAME#dir flash:/core | grep HOSTNAME
40486  -rw-        108268293  Oct 21 2019 16:07:50 -04:00
HOSTNAME-system-report_20191021-200748-UTC.tar.gz
40487  -rw-            17523  Oct 21 2019 16:07:56 -04:00
HOSTNAME-system-report_20191021-200748-UTC-info.txt
40484  -rw-         48360998  Oct 21 2019 16:55:24 -04:00
HOSTNAME-system-report_20191021-205523-UTC.tar.gz
40488  -rw-            14073  Oct 21 2019 16:55:26 -04:00
HOSTNAME-system-report_20191021-205523-UTC-info.txt
```

# 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。

```
WARNING:Fan PS1/0 in slot 1 has the error: Error Status,
Please replace it with a new fan.
```

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。

個々のファンに障害が発生すると、次のようなメッセージが表示されます。

```
The fan in slot PS17/1 is encountering a failure condition
```

ファントレイ全体に障害が発生し、システムがシャットダウンすると、次のようなメッセージが表示されます。

```
 Shutting down system now because the fans in slot PS17 have all failed.
```

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。

ファンの障害の詳細については、『Cisco Catalyst 9400 Series Switches Hardware Installation Guide』を参照してください。

## 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が、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

## ソフトウェア障害からの回復

### 始める前に

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する必要があります。

ここで紹介する手順では、破損したイメージ ファイルまたは不適切なイメージ ファイルの回復に boot loader コマンドおよび TFTP を使用します。

### 手順

**ステップ 1** PC 上で、Cisco.com からソフトウェア イメージ ファイル（*image.bin*）をダウンロードします。

**ステップ 2** TFTP サーバにソフトウェア イメージをロードします。

**ステップ 3** PC をスイッチのイーサネット管理ポートに接続します。

**ステップ 4** スイッチの電源コードを取り外します。

**ステップ 5** 電源コードを再接続し、デバイスが自動ブートの準備を行っているときに Ctrl+C キーを押します。これにより、デバイスは ROMmon モードになります。

**例**：

```
Last reset cause: Power on
C9x00-SUP-1 platform with 16777216 Kbytes of main memory

Preparing to autoboot. [Press Ctrl-C to interrupt]  3    (interrupted)
switch:
switch:
```

**ステップ 6** ブートローダー（ROMMON）プロンプトで、TFTP サーバに ping を実行できることを確認します。

a) スイッチの IP アドレスを設定します：**set IP_ADDRESS** *ip_address*

**例**：

```
switch: set IP_ADDRESS 192.0.2.123
```

b) スイッチのサブネットマスクを設定します：**set IP_SUBNET_MASK** *subnet_mask*

**例**：

```
switch: set IP_SUBNET_MASK 255.255.255.0
```

c) デフォルトゲートウェイを設定します： **set DEFAULT_GATEWAY** *ip_address*

**例**：

```
switch: set DEFAULT_ROUTER 192.0.2.1
```

d) 次のコマンドを実行して、TFTP サーバに ping を実行できることを確認します。 **rommon: ping** *ip_address_of_TFTP_server*

**例**：

```
rommon 1 >ping 192.0.2.15
Pinging 192.0.2.15, 4 times, with packet-size 0
16 bytes from 192.0.2.15: ICMP seq#1 RTT=1 ms
16 bytes from 192.0.2.15: ICMP seq#2 RTT<0 ms
16 bytes from 192.0.2.15: ICMP seq#3 RTT<0 ms
16 bytes from 192.0.2.15: ICMP seq#4 RTT<0 ms
Packet(s): Sent Received Loss | RTT: Minimum Maximum Average
4 4 0 % | <0 ~1 ~0
rommon 2 >
```

**ステップ 7** ブートローダー（ROMMON）プロンプトで、**boot tftp** コマンドを開始します。これにより、スイッチでソフトウェアイメージを容易に回復できます。

例：

```
rommon 1 >boot tftp://10.168.0.1/cat9k/cat9k_iosxe.2017-08-25_09.41.bin
attempting to boot from [tftp://10.168.0.1/cat9k/cat9k_iosxe.2017-08-25_09.41.SSA.bin]

 interface : eth0
 macaddr   : E4:AA:5D:59:7B:44
 ip        : 10.168.247.10
 netmask   : 10.255.0.0
 gateway   : 10.168.0.1
 server    : 10.168.0.1
 file      : cat9k/cat9k_iosxe.2017-08-25_09.41.bin


             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 [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.6.1
 RELEASE SOFTWARE (fc2)
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Thu 24-Aug-17 13:23 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 C9XXX (X86) processor (revision V00) with 869398K/6147K bytes of memory.
Processor board ID FXS1939Q3LZ
144 Gigabit Ethernet interfaces
16 Ten Gigabit Ethernet interfaces
4 Forty Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
15958516K bytes of physical memory.
11161600K bytes of Bootflash at bootflash:.
1638400K bytes of Crash Files at crashinfo:.
0K bytes of WebUI ODM Files at webui:.

%INIT: waited 0 seconds for NVRAM to be available


Press RETURN to get started!
```

あるいは、Telnetまたは管理ポートを通じてTFTPからローカルフラッシュにイメージをコピーした後、ローカルフラッシュからデバイスをブートします。

# 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と、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。
- PCを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** デュアル スーパバイザ モジュールのデバイスでは、パスワード回復手順の前に、スタンバイ スーパバイザをシャーシから取り外します。スイッチまたはアクティブなスーパバイザモジュールに電源コードを再接続します。スイッチまたはアクティブなスーパバイザモジュールの起動中に、Ctrl+C を押して自動ブートを停止し、ROMMON モードを開始します。

「パスワード回復がイネーブルになっている場合の手順」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
```

## 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** （ステップ 2 と 3 で変更した）ブートローダ 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て、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、Ctrl+C 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 インストレーション 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、デバイスに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（電気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。

## SFP モジュール ステータスのモニタリング

**show interfaces transceiver** 特権 EXEC コマンドを使用すると、SFP モジュールの物理または動作ステータスを確認できます。このコマンドは、DOM（Digital Optics Monitoring）機能をサポートしている SFP 上でのみ動作します。このコマンドは、温度や特定のインターフェイス上の SFP モジュールの現状などの動作ステータスと、アラーム ステータスを表示します。また、このコマンドを使用して SFP モジュールの速度およびデュプレックス設定も確認できます。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にある **show interfaces transceiver** コマンドを参照してください。

## 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 |
|---|---|
| **ping ip** *host* \| *address*<br><br>`Device# ping 172.20.52.3` | 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。 |

## 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。

温度の値、状態、しきい値を表示するには、**show env** 特権 EXEC コマンドを使用します。温度の値は、デバイス内の温度です（外部温度ではありません）。

## 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 *27*：物理パスのモニタリング

| コマンド | 目的 |
|---|---|
| **tracetroute mac** [ **interface** *interface-id*] {*source-mac-address*} [ **interface** *interface-id*] {*destination-mac-address*} [ **vlan** *vlan-id*] [**detail**] | 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。 |
| **tracetroute mac ip** {*source-ip-address* \| *source-hostname*}{*destination-ip-address* \| *destination-hostname*} [**detail**] | 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。 |

## 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*<br>`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。 |

## TDR の実行および結果の表示

TDR を実行するには、**test cable-diagnostics tdr interface** *interface-id* 特権 EXEC コマンドを入力します。

TDR の結果を表示するには、**show cable-diagnostics tdr interface** *interface-id* 特権 EXEC コマンドを実行します。

## 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ることが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 hardware fed active** 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 コマンドの使用方法

**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。デバッギン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。

# パケット損失のトラブルシューティング

システムで、ネットワーク接続の全部または一部の損失またはパケット損失が発生した場合は、基本トラブルシューティング手順を実行して、一般的な原因を取り除く必要があります。一般的な原因には次のものがあります。

- ケーブル配線の不具合
- ポートの不具合
- 速度とデュプレックスの不一致
- ネットワーク インターフェイス カード（NIC）の問題

1. 一般的なこれらの原因をトラブルシューティングしても問題を絞り込むことができない場合は、**show platform hardware iomd 1/0 data-path** コマンドを入力してパケット損失を確認しますパケット損失の現象が見られる場合は、**reload** コマンドを使用してスイッチをソフト リセットします。
2. リロードでスーパーバイザモジュール診断が失敗する場合は、スイッチに電源を再投入します。
3. GOLD（Generic On Line Diagnostics）**show diagnostic bootup** コマンドを入力し、診断が失敗するかどうかを判断します。

   診断が再度失敗した場合は、ハードウェアの問題が考えられます。

   詳細については TAC にお問い合わせください。
4. 手順 2 で電源を再投入してから、スーパーバイザモジュールが失敗することなく診断テストに合格した場合は、次の手順を実行します。
   1. **show tech-support** コマンドの出力を収集します。
   2. ボックスからすべての電源装置を取り外し、シリアル番号、シスコ部品番号、電源装置のメーカーの情報を収集します。
   3. 調べた情報をもとに、TAC にお問い合わせください。

# モジュールがオンラインでない場合のトラブルシューティング

ステータス LED がレッドに点灯している場合、または **show module** コマンドの出力に次のステータスのいずれかが含まれている場合は、モジュールに障害がある可能性があります。

- other

  モジュールが正しく取り付けられ、そのネジが完全に締め付けられていることを確認してください。それでもモジュールがオンラインにならない場合は、**hw-module slot** *slot-number* **reset** コマンドを入力します。それでもモジュールがオンラインにならない場合は、予備のスロットにモジュールを取り付けるか、機能しているモジュールを取り外してそのスロットにモジュールを取り付けるか、別のシャーシにモジュールを取り付けてみてください。
- faulty

ステータスが「faulty」の場合はポートで shutdown を実行してから、no shutdown コマンドを実行します。これで問題が解決しない場合は、選択したモジュールで総合オンライン診断（GOLD）の diagnostic **start module** *mod-number* **test** コマンドを実行して、診断を開始します。

- power-deny

  ステータスが「power-deny」の場合は、スイッチの電力が不足しており、このモジュールに必要な電力が供給されていません。**show power** コマンドを入力して、十分な電力があるかどうかを確認してください。

- power-bad

  ステータスが「power-bad」の場合は、スイッチがスイッチング モジュールを検出していますが、電力を割り当てることができていません。この状況は、スーパーバイザエンジンがモジュール上のシリアル PROM（SPROM）コンテンツにアクセスしてラインカードの ID を識別できない場合に発生する可能性があります。**show idprom** *module slot* コマンドを入力して、SPROM が読み取り可能であることを確認します。SPROM にアクセスできない場合は、モジュールをリセットします。

  **show diagnostics online** *module slot number* コマンドを入力して、モジュールのハードウェア障害を特定します。それでもモジュールがオンラインにならない場合は、サービス リクエストを作成して、詳しいトラブルシューティング方法を TAC にお問い合わせください。上記の出力で収集したスイッチのログと実行したトラブルシューティング手順のログを使用します。

# インターフェイスの問題のトラブルシューティング

**show interface** コマンドの出力でエラーが確認された場合の考えられる原因は次のとおりです。

- ケーブルまたは NIC の不具合などの物理層の問題
- 速度とデュプレックスの不整合などの設定上の問題
- オーバーサブスクライブなどのパフォーマンスの問題

これらの問題を理解してトラブルシューティングするには、次の『*Troubleshooting Switch Port and Interface Problems*』を参照してください。http://www.cisco.com/en/US/products/hw/switches/ps708/products_tech_note09186a008015bfd6.shtml

# ワークステーションがネットワークにログインできない場合のトラブルシューティング

起動中にワークステーションがネットワークにログインできない、またはクライアントマシンの電源をオンにしたり再起動したりしたときに DHCP アドレスを取得できない場合は、スイッ

チによる初期接続の遅延が原因である可能性があります。これを確認するために、次のことを調べてください。

- Microsoft ネットワーク クライアントに「No Domain Controllers Available」と表示されている。
- DHCP から「No DHCP Servers Available」が報告されている。
- Novell Internetwork Packet Exchange（IPX）ネットワーク ワークステーションで、起動時の Novell ログイン画面が表示されない。
- AppleTalk ネットワーク クライアントに「Access to your AppleTalk network has been interrupted. In order to reestablish your connection, open and close the AppleTalk control panel.」と表示される。AppleTalk クライアントのセレクタ アプリケーションにゾーン リストが表示されないか、不完全なゾーン リストが表示される。
- IBM Network Station で、次のメッセージのいずれかが表示される場合がある。
  - NSB83619—Address resolution failed
  - NSB83589—Failed to boot after 1 attempt
  - NSB70519—Failed to connect to a server

このような現象が発生する原因は、スパニングツリー プロトコル（STP）、EtherChannel、トランキング、または自動ネゴシエーションの遅延により発生するインターフェイス遅延の可能性があります。

# 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

表 *28 : OBFL* 情報を表示するためのコマンド

| コマンド |
|---|
| **show logging onboard RP active clilog** |
| **show logging onboard RP active environment** |
| **show logging onboard RP active message** |
| **show logging onboard RP active counter** |
| **show logging onboard RP active temperature** |
| **show logging onboard RP active uptime** |
| **show logging onboard RP active voltage** |

| コマンド |
| --- |
| **show logging onboard RP active status** |

# 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別するには、**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 *29 : 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。 |
| 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。 |

# 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 *30 : 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。<br><br>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 |
| --- | --- |
| | 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。<br><br>**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。<br><br>（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。<br><br>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。<br><br>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。<br><br>（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 ケーブルでは機能しません。<br><br>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。<br><br>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。<br><br>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 inline power** コマンドを使用して、利用可能な電力量を確認します。 |

| 症状または問題 | 考えられる原因と解決法 |
| --- | --- |
| 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。<br>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。 | |

| 症状または問題 | 考えられる原因と解決法 |
| --- | --- |
| | 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。<br><br>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あります。<br><br>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。<br><br>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。<br><br>**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。<br><br>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。<br><br>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。<br><br>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。<br><br>**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 |

| 症状または問題 | 考えられる原因と解決法 |
| --- | --- |
| | 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。<br><br>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。<br><br>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る場合があります。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。 |
| 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。<br><br>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。 | スイッチから受電デバイスまでのすべての電気系統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などが発生します。<br><br>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。<br><br>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。<br><br>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。<br><br>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）。<br><br>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があります。 |

| 症状または問題 | 考えられる原因と解決法 |
|---|---|
| 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。<br>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。 | **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。<br>**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。<br>**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。 |

# 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 *31 : 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。 |

| 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 つのプローブごとに、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 *32 : 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。 | *Command Reference (Catalyst 9400 Series Switches)* |

# 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と情報

| リリース | 変更内容 |
|---|---|
| Cisco IOS XE Everest 16.6.1 | この機能が導入されました。 |
| Cisco IOS XE Amsterdam 17.3.1 | ホスト名がシステムレポートファイルの先頭に追加されます。これにより、システムレポートファイルが一意に識別可能になります。 |

第 20 章

# 破損または損失したファイル イメージ、あるいは ROMmon モードでのリカバリ


- はじめに （395 ページ）
- 破損または損失したイメージからの ROMmon モードでのスイッチのリカバリ （396 ページ）
- 連続的なリブートからのスイッチのリカバリ （397 ページ）
- 破損または損失したイメージからのリカバリ （400 ページ）
- スイッチの回復に関する機能情報 （408 ページ）


## はじめに

ここでは、イメージの消失または破損、あるいは不正なブート変数から Catalyst 9400 シリーズ スーパーバイザを回復する方法について説明します。スーパーバイザ モジュール イメージは、Trivial File Transfer Protocol（TFTP）ダウンロード時や、ユーザが手動で削除するときに破損する場合があります。スイッチには、これらのスーパーバイザ エンジンでこのような状況が発生した場合のさまざまな回復方法が用意されています。

スーパーバイザを備えたスイッチが起動またはリセットする場合、次の 2 つの可能性があります。

1. スイッチが正常に起動して、ホスト名> プロンプトまたはデフォルトの Switch> プロンプトが表示されます。
2. スイッチがイメージを見つけられない、イメージが破損している、ブートフラッシュ デバイスにイメージがない、またはブート変数の設定が誤っている場合は ROM モニタ（ROMmon）モードになります。この場合はrommon>プロンプトが表示されます。ROMmon モードでは、スイッチがいずれかの bootflash デバイスから有効なシステム イメージを検出できるようにする必要があります。また、これらのスーパーバイザ エンジンにはROMmon モードからのみ使用可能なイーサネット管理ポート（10/100 Base T）が備わっており、TFTP プロセスから TFTP を介して有効な新しいイメージをダウンロードするように設定できます。コンソール ポートを介してイメージをコピーできる Xmodem または Ymodem のオプションはありません。

### 通常の動作

スイッチが正常に動作すると、hostname> プロンプトまたはデフォルトの Switch> プロンプトが表示されます。スーパーバイザ フラッシュ ドライブの内容を表示するには、次の例に示すように dir bootflash: コマンドを発行します。verify コマンドを発行し、次の例に示すように、有効なチェックサムがイメージにあるかどうかを特定します。

```
Switch#dir bootflash:
Directory of bootflash:/

    1  -rw-     6516904   Jul 11 2000 02:14:10  packages.conf

61341696 bytes total (54824664 bytes free)

Switch#verify bootflash:packages.conf
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
Verified bootflash:packages.conf
```

スイッチはすべてのフラッシュ デバイスが ROMmon モードであることを認識しているため、このドキュメントの残りの項で説明するように、**dir** *device-name* コマンドを発行してフラッシュの内容を表示します。

前の例では、ブートフラッシュにあるブート イメージは 1 つのみです。bootflash: には、適合する数のシステム イメージを含めることができます。ブートフラッシュのサイズは 64 MB に固定されています。

次に、Catalyst 9400 がサポートしている ROMmon 変数と、それらを設定する例を示します。

- MANUAL_BOOT=yes

  「yes」値は自動ブートを無効にし、手動ブートを有効にします。「no」値は自動ブートを有効にします。

- DEFAULT_GATEWAY=172.27.74.1
- IP_ADDRESS=172.27.74.111
- IP_SUBNET_MASK=255.255.255.0
- TFTP_SERVER=172.19.64.31

# 破損または損失したイメージからの ROMmon モードでのスイッチのリカバリ

次の理由により、スイッチは ROMmon モードになることがあります。

1. イメージが破損したか、または削除された後にスイッチがリロードまたはクラッシュした。詳細については、このドキュメントの「破損または損失したイメージからのリカバリ」の項を参照してください。

2. 有効なシステム イメージが bootflash: にあるかどうかを判断するには、このドキュメントの「連続的なリブートからのリカバリ」を参照してください。ファイルが存在しない場合は、このドキュメントの「破損または損失したイメージからのリカバリ」の項を参照してください。
3. ブート変数は正しくないが、有効なイメージがある。詳細については、このドキュメントの「連続的なリブートからのリカバリ」の項を参照してください。

スイッチが ROMmon モードの場合、主に次のような現象がシステムに発生します。

- ROMmon モードは VLAN インターフェイス間でルーティングできず、スイッチをリカバリすることのみが目的のため、ルーティング障害が発生します。
- 障害が発生したインターフェイスのいずれかに Telnet を試行した場合や、スーパーバイザのコンソール ポートに接続している場合は、次のプロンプトが表示されます。

  rommon 1>

# 連続的なリブートからのスイッチのリカバリ

ブート変数が正しいシステム イメージ ファイルおよび適切な宛先デバイスに設定されていない場合、スイッチが連続的にリブートを繰り返すようになることがあります。

次の手順で、スイッチをリカバリできます。

1. 直前の出力を表示し、リカバリを実行するには、コンソールがスーパーバイザに接続されている必要があります。標準的な Windows オペレーティング プラットフォーム上で次の設定値を使用して COM1 への HyperTerminal 直接接続を設定します。
   - 9600 bps
   - 8 データビット
   - パリティなし
   - 1 ストップ ビット
   - フロー制御 = なし

   オスのロール RJ-45 ケーブルを使用して、PC 上の COM1 からスーパーバイザ モジュール上のコンソールに接続します。PC 上の DB-9 コネクタを使用します。
2. Control+C キーを押し、ROMmon モードに移行したときに自動ブートが無効になるまで、リブートが続行されます。これを次の例で示します。

```
**********************************************************
 *                                                        *
 * Welcome to ROM Monitor for WS-X4014 System.            *
 * Copyright (c) 2002 by Cisco Systems, Inc.              *
 * All rights reserved.                                   *
 *                                                        *
 **********************************************************
```

```
 ROM Monitor Program Version 12.1(11br)EW

 Board type 1, Board revision 5
 Swamp FPGA revision 14, Dagobah FPGA revision 48


 MAC Address  : 00-01-96-d9-f6-fe
 Ip Address   : Not set.
 Netmask      : Not set.
 Gateway      : Not set.
 TftpServer   : Not set.
 Main Memory  : 256 Mbytes



 ***** The system will autoboot in 5 seconds *****


 Type control-C to prevent autobooting.

!--- Press Control-C.

 Autoboot cancelled..... please wait!!!
rommon 1 > [interrupt]
```

3. **dir bootflash:** コマンドを入力し、ブートフラッシュにあるファイルのリストを表示します。

```
rommon 1 >dir bootflash:

 Size       Attributes Name
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
74871940     -rw-     sdX86.170607
46858696     -rw-     mcln_x86_kernel_20170525.SSA
4096         drw-     .installer
1295         -rw-     bootloader_evt_handle.log
4096         drw-     .ssh
8192         drw-     core
4096         drw-     .prst_sync
4096         drw-     .rollback_timer
4096         drw-     gs_script
252          -rw-     boothelper.log
33554432     -rw-     nvram_config
35           -rw-     pnp-tech-time
45045        -rw-     pnp-tech-discovery-summary
1036         -rw-     vlan.dat
67587176     -rw-     mcln_x86_kernel_20170628.SSA
89423547     -rw-     sdX86.170703
89401770     -rw-     sdX86.170606
54034568     -rw-     upg_NSB2SB_rom133sb_fpga170505.SSA
98869        -rw-     memleak.tcl
4096         drw-     onep
15           -rw-     dope_hist
8314         -rw-     packages.conf
47           -rw-     DHCP-snooping
5514216      -rw-
cat9k-cc_srdriver.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
76571620    -rw-    cat9k-espbase.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
1536996      -rw-
cat9k-guestshell.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
373478368    -rw-    cat9k-rpbase.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
29546198    -rw-    cat9k-rpboot.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
27648996    -rw-    cat9k-sipbase.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
54924256    -rw-    cat9k-sipspa.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
```

```
6767588     -rw-    cat9k-srdriver.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
12256224     -rw-     cat9k-webui.BLD_V166_THROTTLE_LATEST_20170820_090650.SSA.pkg
5510104      -rw-     cat9k-cc_srdriver.2017-08-25_09.41_srchanna.SSA.pkg
76563412     -rw-     cat9k-espbase.2017-08-25_09.41_srchanna.SSA.pkg
1536984      -rw-     cat9k-guestshell.2017-08-25_09.41_srchanna.SSA.pkg
373449684    -rw-     cat9k-rpbase.2017-08-25_09.41_srchanna.SSA.pkg
29547297     -rw-     cat9k-rpboot.2017-08-25_09.41_srchanna.SSA.pkg
27644884     -rw-     cat9k-sipbase.2017-08-25_09.41_srchanna.SSA.pkg
54916052     -rw-     cat9k-sipspa.2017-08-25_09.41_srchanna.SSA.pkg
6767572      -rw-     cat9k-srdriver.2017-08-25_09.41_srchanna.SSA.pkg
12256208     -rw-     cat9k-webui.2017-08-25_09.41_srchanna.SSA.pkg
8266         -rw-     cat9k_iosxe.2017-08-25_09.41_srchanna_iso1.SSA.conf
5510104      -rw-     cat9k-cc_srdriver.2017-08-25_09.08_srchanna.SSA.pkg
76563412     -rw-     cat9k-espbase.2017-08-25_09.08_srchanna.SSA.pkg
1536984      -rw-     cat9k-guestshell.2017-08-25_09.08_srchanna.SSA.pkg
373445588    -rw-     cat9k-rpbase.2017-08-25_09.08_srchanna.SSA.pkg
29547259     -rw-     cat9k-rpboot.2017-08-25_09.08_srchanna.SSA.pkg
27644884     -rw-     cat9k-sipbase.2017-08-25_09.08_srchanna.SSA.pkg
54916052     -rw-     cat9k-sipspa.2017-08-25_09.08_srchanna.SSA.pkg
6767572      -rw-     cat9k-srdriver.2017-08-25_09.08_srchanna.SSA.pkg
12256208     -rw-     cat9k-webui.2017-08-25_09.08_srchanna.SSA.pkg
8266         -rw-     cat9k_iosxe.2017-08-25_09.08_srchanna_iso2.SSA.conf
9079         -rw-     packages.conf.02-
47160612     -rw-     dope-A0-300817-143021.csv.gz
11414692     -rw-     dope-LC1-A0-300817-173015.csv.gz
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

（注） スイッチが連続的にリブートする理由は、指定したイメージファイル名がなくても、ブートフラッシュに有効なファイルがあるためです。また、指定したシステム イメージ ファイルの名前は大文字と小文字を区別する必要もあります。正しく指定されていない場合は、連続的なリブートが行われます。

4. 必要なシステム イメージ ファイルが bootflash: にある場合は、boot bootflash:<filename> コマンドを発行するとスイッチをブートできます。指定したそのイメージでシステムがブートされます。指定したシステムイメージが破損していたり、有効なシステム ファイルがないことによってスイッチがロードに失敗する場合は、このドキュメントの「破損または損失したイメージからのリカバリ」の項を参照してください。

5. 次の例に示すように、**enable** コマンドを発行して EXEC モードを開始します。

```
Switch>enable
Password:
Switch#
```

6. システムがバックアップされます。**dir bootflash:** コマンドを発行して bootflash: 内のファイルをメモします。

7. **show bootvar** コマンドを発行して、現在のブート変数を確認します。

```
Switch#show bootvar
BOOT variable = bootflash:packages.conf;
Configuration Register is 0x1822
MANUAL_BOOT variable = yes
BAUD variable = 115200
ENABLE_BREAK variable =
CONFIG_FILE variable =
```

8. 既存の不正なブート変数を削除し、正しいブート変数を追加します。これを実行するには、**configure terminal** コマンドを発行します。

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#no boot system flash bootflash:packages.conf
Switch(config)#boot system flash bootflash:cat9400packages.conf
Switch(config)#end

00:01:31: %SYS-5-CONFIG_I: Configured from console by console
```

9. **write memory** コマンドを発行して、実行中の設定をスタートアップに保存します。

```
Switch#write memory
Building configuration...
Compressed configuration from 4359 bytes to 1730 bytes[OK]
Switch#
```

10. ブート変数を再度確認して、スイッチが次回のリブート時に正しいシステム ファイルを起動するように適切に設定されていることを確認します。これを実行するには、**show bootvar** コマンドを発行します。

```
Switch#show bootvar
BOOT variable = bootflash:cat9400packages.conf
Configuration Register is 0x1822
MANUAL_BOOT variable = yes
BAUD variable = 115200
ENABLE_BREAK variable =
CONFIG_FILE variable =
```

# 破損または損失したイメージからのリカバリ

指定したイメージが破損している場合や、イメージファイルが存在しない場合、スーパーバイザは ROMmon モードでブートします。通常、スイッチをリカバリするには、bootflash: デバイスに複数のイメージが必要です。

示された順序で次の手順を実行することによって、有効なイメージがない場合も ROMmon モードからイメージを正常にリカバリするのに役立ちます。

1. スーパーバイザへのコンソール接続を確立します。通常、Windows OS 上では次の設定値を使用して COM1 への HyperTerminal 直接接続を設定します。
   - 9600 bps
   - 8 データビット
   - パリティなし
   - 1 ストップ ビット

   オスのロール RJ-45 ケーブルを使用して、PC 上の COM1 からスーパーバイザ モジュール上のコンソールに接続します。PC 上の DB-9 コネクタと [HyperTerminal connect] ウィンドウを使用し、スーパーバイザに接続します。

2. Enter を押します。**rommon >** プロンプトが表示された場合、手順 3 に進みます。スイッチが連続的にリブートする場合はControl+Cキーを押して自動ブートを無効にし、ROMmon モードに移行します。

```
**********************************************************
 *                                                        *
 * Welcome to ROM Monitor for WS-X4014 System.            *
 * Copyright (c) 2002 by Cisco Systems, Inc.              *
 * All rights reserved.                                   *
 *                                                        *
 **********************************************************

 ROM Monitor Program Version 12.1(11br)EW

 Board type 1, Board revision 5
 Swamp FPGA revision 14, Dagobah FPGA revision 48


 MAC Address  : 00-01-96-d9-f6-fe
 Ip Address   : Not set.
 Netmask      : Not set.
 Gateway      : Not set.
 TftpServer   : Not set.
 Main Memory  : 256 Mbytes



 ***** The system will autoboot in 5 seconds *****


 Type control-C to prevent autobooting.
 !--- Press Control-C.
 Autoboot cancelled..... please wait!!!
rommon 1 > [interrupt]
```

3. 次の例に示すように、**dir bootflash:** コマンドを発行し、bootflash: 内に有効なファイルがあることを確認します。有効なファイルがある場合、リカバリについてはこのドキュメントの「連続的なリブートからのリカバリ」の項を参照してください。それ以外の場合は、次の手順を続行します。

```
rommon 1 >dir bootflash:

File size Checksum File name
-----------------------------------------------

Total space = 61341696 bytes, Available = 61341696 bytes
```

set コマンドを発行して、現在の環境変数を表示します。

```
rommon 3 >set
PS1=rommon ! >
RommonBuild=5
ConfigReg=0x2102
BOOT=bootflash:packages.conf
SkipDiags=0
BSI=0
RET_2_RTS=13:36:46 UTC Tue Aug 15 2000
RET_2_RUTC=966346606
BootStatus=Failure
BootedFileName=bootflash:packages.conf
RommonVer=12.1(11br)EW
```

4. **unset BOOT** コマンドを発行して、ロードするファイルを定義する、現在無効なブート変数をクリアします。

```
rommon 6 >unset BOOT
```

5. スーパーバイザ上の管理ポートをネットワークに接続し、TFTP サーバにアクセスします。スーパーバイザ エンジン上のファスト イーサネット ポート（10/100 MGT）は、現在のソフトウェア リリースの通常動作では機能しません。10/100 MGT に接続されたイーサネット ケーブルは、ROMmon モードでのみアクティブになります。

次の例に示すように、10/100 MGT ポートを PC/ルータに直接接続する場合は、ストレート ケーブルを使用します。別のスイッチに接続する場合は、クロス ケーブルを使用します。

```
rommon 7 >

!--- Connect the appropriate cable to connect to the network.


Established physical link 100MB Full Duplex
Network layer connectivity may take a few seconds
```

管理ポートは、接続されたデバイスと速度およびデュプレックスを自動ネゴシエートします。現在、速度およびデュプレックスの設定値はハードコードできません。このポートは ROMmon モード専用であり、TFTP の場合にのみに使用できるため、速度とデュプレックスに自動ネゴシエーションの潜在的な問題による不一致があっても大きな懸念ではありません。TFTP アプリケーションにはダウンロードするシステム イメージの損傷を防ぐための内部パケット損失メカニズムがあります。

6. **set** コマンドを入力し、次の例に示すように 10/100 MGT ポートに IP アドレスを設定します。サブネット マスクが指定されていない場合、IP アドレスはデフォルトのサブネット マスクを取得します。

```
rommon 7> set IP_ADDRESS=192.168.247.10
rommon 8> set IP_SUBNET_MASK=255.255.255.0
```

7. **set DEFAULT_GATEWAY** コマンドを入力し、次の例に示すように、TFTP サーバに到達するために使用するデフォルトのゲートウェイをスイッチに設定します。デフォルトのゲートウェイには、手順 6 で設定した IP アドレスと同じサブネット内のルーティング デバイスを使用する必要があります。

```
rommon 8> set DEFAULT_GATEWAY=192.168.0.1
```

8. set コマンドを入力し、作成した設定を確認します。

```
switch: set
BAUD=9600
BOOT=bootflash:cat9k_iosxe.16.06.01.SPA.bin;
BOOTLDR=
BSI=0
CALL_HOME_DEBUG=0000000000000
CONFIG_FILE=
CRASHINFO=bootflash:crashinfo_RP_01_00_20170907-054557-UTC
DEFAULT_GATEWAY=172.16.94.193
DEFAULT_ROUTER=172.16.94.193
DISABLE_AUTH=
ENABLE_BREAK=yes
EULA_ACCEPTED=TRUE
```

```
IP_ADDRESS=172.16.94.221
IP_SUBNET_MASK=255.255.255.224
LICENSE_BOOT_LEVEL=network-advantage+dna-advantage,all:MACALLAN-CHASSIS;
MAC_ADDR=E4:AA:5D:59:7A:FC
MANUAL_BOOT=no
MODEL_NUM=C9400-SUP-1
MOTHERBOARD_SERIAL_NUM=JAE2124023Z
RANDOM_NUM=808994625
RET_2_RCALTS=1504781417
RET_2_RTS=
ROMMON>DISABLE_AUTH=
ROMMON_AUTOBOOT_ATTEMPT=3
SWITCH_IGNORE_STARTUP_CFG=0
SWITCH_NUMBER=1
SYSTEM_SERIAL_NUM=
TEMPLATE=access
TFTP=SERVER=172.16.53.46
```

**9.** TFTP サーバに ping を実行し、スーパーバイザ エンジンの MGT ポートからサーバへの接続があることを確認します。次の例に示すように、**ping** *tftp_server_ip_address* コマンドを入力します。

```
rommon 9 >ping 192.168.0.1

Host 192.168.0.1 is alive
```

ping が成功しない場合は、デフォルトのゲートウェイから TFTP サーバへの IP 接続の問題をトラブルシューティングします。TFTP サーバが同じサブネットにある場合は、そのサーバが ping の送信先の IP アドレスで設定されていることを確認します。

**10.** TFTP サーバへの ping が成功したら、**boot tftp://***tftp_server_ip_address/image_path_and_file_name* コマンドを発行して、スーパーバイザのブートに TFTP サーバで使用可能なシステム イメージを指定します。

```
rommon 1>boot tftp://192.168.0.1/cat9k/cat9k_iosxe.2017-08-25_09.41.bin
attempting to boot from
[tftp://192.168.0.1/cat9k/cat9k_iosxe.2017-08-25_09.41.SSA.bin]

 interface : eth0
 macaddr   : E4:AA:5D:59:7B:44
 ip        : 192.168.247.10
 netmask   : 255.255.0.0
 gateway   : 192.168.0.1
 server    : 192.168.0.1
 file      : cat9k/cat9k_iosxe.2017-08-25_09.41.bin
```

```
                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 [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Experimental
 Version 16.6.20170824:202043 [v166_throttle-/nobackup/srchanna/lock_166 104]
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Thu 24-Aug-17 13:23 by srchanna


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 C9407R (X86) processor (revision V00) with 869398K/6147K bytes of memory.
Processor board ID FXS1939Q3LZ
144 Gigabit Ethernet interfaces
16 Ten Gigabit Ethernet interfaces
4 Forty Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
15958516K bytes of physical memory.
11161600K bytes of Bootflash at bootflash:.
1638400K bytes of Crash Files at crashinfo:.
0K bytes of WebUI ODM Files at webui:.

%INIT: waited 0 seconds for NVRAM to be available



Press RETURN to get started!


*Sep  1 13:20:31.770: %SMART_LIC-6-AGENT_READY: Smart Agent for Licensing is
initialized
*Sep  1 13:20:37.460: NGWC: not crashing for mcprp_get_ecfm_brain_mac_addr
```

```
*Sep  1 13:20:39.084: %CRYPTO-4-AUDITWARN: Encryption audit check could not be
performed
*Sep  1 13:20:39.125: %SPANTREE-5-EXTENDED_SYSID: Extended SysId enabled for type
 vlan
*Sep  1 13:20:39.350: %LINK-3-UPDOWN: Interface Lsmpi12/3, changed state to up
*Sep  1 13:20:39.351: %LINK-3-UPDOWN: Interface EOBC12/1, changed state to up
*Sep  1 13:20:39.351: %LINEPROTO-5-UPDOWN: Line protocol on Interface LI-Null0,
changed state to up
*Sep  1 13:20:39.351: %LINK-3-UPDOWN: Interface GigabitEthernet0/0, changed state
 to down
*Sep  1 13:20:39.351: %LINK-3-UPDOWN: Interface LIIN12/2, changed state to up
*Sep  1 13:20:39.418: %SYS-3-HARIKARI: Process pki_app top-level routine exited
*Sep  1 13:20:39.431: %PNP-6-PNP_DISCOVERY_STOPPED: PnP Discovery stopped (Startup
 Config Present)
*Sep  1 13:20:39.441: %ILPOWER-6-SET_ILPOWER: Set power allocated to POE to 2780
for slot 0
*Sep  1 13:20:40.292: %IOSXE_MGMTVRF-6-CREATE_SUCCESS_INFO: Management vrf Mgmt-vrf
 created with ID 1, ipv4 table-id 0x1, ipv6 table-id 0x1E000001
*Sep  1 13:20:00.189: %HW_PFU-3-PFU_IDPROM_CORRUPT:  R1/0: cmand:  The PEM/FM idprom
 could be read, but is corrupt in slot P17 The system will run without environmental
 monitoring for this component
*Sep  1 13:20:11.565: %CMRP_PFU-6-PWR_MGMT_OK:  R1/0: cmand:  Sufficient number of
 power supplies (4) are installed for power redundancy mode none (excess power 3150
 watts).
*Sep  1 13:20:24.203: %CMRP_PFU-6-PWR_MGMT_OK:  R1/0: cmand:  Sufficient number of
 power supplies (4) are installed for power redundancy mode none (excess power 1685
 watts).
*Sep  1 13:20:35.524: %IOSXE-4-PLATFORM:  R1/0: kernel: pci 0000:15:00.1: BAR 2:
[??? 0x00000000 flags 0x102000] has bogus alignment
*Sep  1 13:20:35.524: %IOSXE-4-PLATFORM:  R1/0: kernel: pci 0000:15:00.1: BAR 4:
[??? 0x00000000 flags 0x102000] has bogus alignment
*Sep  1 13:20:35.524: %IOSXE-3-PLATFORM:  R1/0: kernel: pci 0000:15:00.0: BAR 4:
error updating (0x2021000c != 0x00000c)
*Sep  1 13:20:35.524: %IOSXE-3-PLATFORM:  R1/0: kernel: Error: Storage device
initialization failed with unknown error! error 0xd2b5c46f, Img version: 0xb3537bb0,
 lslot 0
*Sep  1 13:20:36.184: %CMRP_PFU-6-PWR_MGMT_OK:  R1/0: cmand:  Sufficient number of
 power supplies (4) are installed for power redundancy mode none (excess power 1555
 watts).
*Sep  1 13:20:36.532: %EVENTLIB-3-CPUHOG:  R1/0: cmcc:  undefined: 5072ms,
Traceback=1#d1ecae96f48e7b01c7626e7421118715   c:7FF4ACC05000+33410
c:7FF4ACC05000+E0197 tam_act2:7FF4B0B0F000+8F1F tam_act2:7FF4B0B0F000+F676
tam_act2:7FF4B0B0F000+FA28 cmlib_util:7FF4B289F000+9CF5 cmlib_util:7FF4B289F000+19D8B
 :565477ADE000+4F17E :565477ADE000+4F6BA :565477ADE000+723B9 :565477ADE000+60FEA
*Sep  1 13:20:38.312: %IOSXE-4-PLATFORM:  R1/0: kernel: pci 0000:1b:00.1: BAR 2:
[??? 0x00000000 flags 0x102000] has bogus alignment
*Sep  1 13:20:38.312: %IOSXE-4-PLATFORM:  R1/0: kernel: pci 0000:1b:00.1: BAR 4:
[??? 0x00000000 flags 0x102000] has bogus alignment
*Sep  1 13:20:38.312: %IOSXE-3-PLATFORM:  R1/0: kernel: pci 0000:1b:00.0: BAR 4:
error updating (0x2021000c != 0x00000c)
*Sep  1 13:20:38.312: %IOSXE-3-PLATFORM:  R1/0: kernel: Error: Storage device
initialization failed with unknown error! error 0xe7da8ea0, Img version: 0x1dfdead4,
 lslot 2
*Sep  1 13:20:39.370: %IOSXE-4-PLATFORM:  R1/0: kernel: pci 0000:35:00.1: BAR 2:
[??? 0x00000000 flags 0x102000] has bogus alignment
*Sep  1 13:20:39.370: %IOSXE-4-PLATFORM:  R1/0: kernel: pci 0000:35:00.1: BAR 4:
[??? 0x00000000 flags 0x102000] has bogus alignment
*Sep  1 13:20:39.370: %IOSXE-3-PLATFORM:  R1/0: kernel: pci 0000:35:00.0: BAR 4:
error updating (0x2021000c != 0x00000c)
*Sep  1 13:20:39.370: %IOSXE-3-PLATFORM:  R1/0: kernel: Error: Storage device
initialization failed with unknown error! error 0x976ff5fb, Img version: 0xc92bf77b,
 lslot 3
*Sep  1 13:20:40.348: %EVENTLIB-3-CPUHOG:  R1/0: cmcc:  undefined: 5392ms,
Traceback=1#d1ecae96f48e7b01c7626e7421118715   c:7F7461488000+33410
```

```
c:7F7461488000+E0197 tam_act2:7F7465392000+8F1F tam_act2:7F7465392000+F8BF
tam_act2:7F7465392000+FA28 cmlib_util:7F7467122000+9CF5 cmlib_util:7F7467122000+19D8B
 :563133282000+4F17E :563133282000+4F6BA :563133282000+723B9 :563133282000+60FEA
*Sep  1 13:20:40.436: %EVENTLIB-3-CPUHOG:  R1/0: cmcc:  undefined: 5509ms,
Traceback=1#d1ecae96f48e7b01c7626e7421118715   c:7F97C4C70000+33410
pthread:7F97C5014000+100B0 syshw:7F97D0F6F000+FCE1 syshw:7F97D0F6F000+FE58
:562F95125000+54DAE :562F95125000+81565 :562F95125000+425F6 :562F95125000+732ED
cm_oir:7F97CC4EF000+28A55 cmcc_msgs:7F97CB280000+91422 cmcc_msgs:7F97CB280000+91210
*Sep  1 13:20:40.490: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed
 state to down
*Sep  1 13:20:44.086: %CMEM-3-SENSOR_INIT_FAILED:  R1/0: cmand:  Sensor (Temp:
outlet) initialization failed due to No such file or directory.
*Sep  1 13:20:44.513: %SYS-2-PRIVCFG_DECRYPT: Successfully apply the private config
 file
*Sep  1 13:20:44.969: %SW_VLAN-6-VTP_DOMAIN_NAME_CHG: VTP domain name changed to
zdf-acc.
*Sep  1 13:20:44.973: %SYS-5-CONFIG_I: Configured from memory by console
*Sep  1 13:20:44.975: %PARSER-4-BADCFG: Unexpected end of configuration file.

*Sep  1 13:20:44.990: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 1/0, interfaces
 disabled
*Sep  1 13:20:44.990: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 3/0, interfaces
 disabled
*Sep  1 13:20:44.990: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 4/0, interfaces
 disabled
*Sep  1 13:20:44.991: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 5/0, interfaces
 disabled
*Sep  1 13:20:44.991: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 6/0, interfaces
 disabled
*Sep  1 13:20:45.010: %SPA_OIR-6-OFFLINECARD: SPA (C9400-LC-48T) offline in subslot
 1/0
*Sep  1 13:20:45.014: %SPA_OIR-6-OFFLINECARD: SPA (C9400-SUP-1) offline in subslot
 3/0
*Sep  1 13:20:46.458: %SYS-6-BOOTTIME: Time taken to reboot after reload =  303
seconds
*Sep  1 13:20:46.939: %LINK-5-CHANGED: Interface Vlan1, changed state to
administratively down
*Sep  1 13:20:53.578: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F1
*Sep  1 13:20:58.096: %LINK-3-UPDOWN: Interface GigabitEthernet0/0, changed state
 to up
*Sep  1 13:20:59.099: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0, changed state to up
*Sep  1 13:20:59.104: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
*Sep  1 13:21:15.745: %SPA_OIR-6-ONLINECARD: SPA (C9400-SUP-1) online in subslot
4/0
*Sep  1 13:21:15.752: %TRANSCEIVER-6-INSERTED:  R1/0: iomd:  transceiver module
inserted in TenGigabitEthernet4/0/1
*Sep  1 13:21:15.768: %TRANSCEIVER-6-INSERTED:  R1/0: iomd:  transceiver module
inserted in TenGigabitEthernet4/0/3
*Sep  1 13:21:15.778: %SPA_OIR-6-ONLINECARD: SPA (C9400-SUP-1) online in subslot
3/0
*Sep  1 13:21:15.783: %TRANSCEIVER-6-INSERTED:  R1/0: iomd:  transceiver module
inserted in TenGigabitEthernet3/0/1
*Sep  1 13:21:15.790: %TRANSCEIVER-6-INSERTED:  R1/0: iomd:  transceiver module
inserted in TenGigabitEthernet3/0/2
*Sep  1 13:21:17.543: %SPA_OIR-6-ONLINECARD: SPA (C9400-LC-48T) online in subslot
 1/0
*Sep  1 13:21:20.261: %SPA_OIR-6-ONLINECARD: SPA (C9400-LC-48U) online in subslot
 6/0
*Sep  1 13:21:20.353: %SPA_OIR-6-ONLINECARD: SPA (C9400-LC-48U) online in subslot
 5/0
*Sep  1 13:21:33.591: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/47, changed
state to up
*Sep  1 13:21:33.813: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/16, changed
```

```
state to up
*Sep  1 13:21:34.591: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/0/47, changed state to up
*Sep  1 13:21:34.813: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/0/16, changed state to up
*Sep  1 13:21:36.449: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/1, changed state
 to up
*Sep  1 13:21:36.461: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/9, changed state
 to up
*Sep  1 13:21:36.469: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/13, changed
state to up
*Sep  1 13:21:36.477: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/17, changed
state to up
*Sep  1 13:21:36.487: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/24, changed
state to up
*Sep  1 13:21:36.494: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/25, changed
state to up
*Sep  1 13:21:36.504: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/33, changed
state to up
*Sep  1 13:21:36.590: %LINK-3-UPDOWN: Interface GigabitEthernet5/0/37, changed
state to up
*Sep  1 13:21:37.449: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/1, changed state to up
*Sep  1 13:21:37.462: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/9, changed state to up
*Sep  1 13:21:37.469: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/13, changed state to up
*Sep  1 13:21:37.477: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/17, changed state to up
*Sep  1 13:21:37.488: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/24, changed state to up
*Sep  1 13:21:37.494: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/25, changed state to up
*Sep  1 13:21:37.504: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/33, changed state to up
*Sep  1 13:21:37.591: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet5/0/37, changed state to up
```

（注） TFTP サーバの IP アドレスとファイルは、bootflash: へのイメージの転送にその情報を使用しているため、すでに選択されています。変更する場合は、新しい IP アドレスまたはイメージ名を入力します。それ以外の場合は、Enter キーを押すと、事前に選択された情報が使用されます。

システム イメージはコピーされています。**dir bootflash:** コマンドを発行して bootflash: 内のファイルをメモします。

11. **verify** コマンドを入力し、ダウンロードしたファイルの整合性を確認します。確認が失敗した場合は、ファイルを再度ダウンロードする必要があります。

```
Switch#verify bootflash:packages.conf
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCC
Verified bootflash:packages.conf
```

12. **show bootvar** コマンドを入力し、現在のブート変数を確認します。

13. 既存の不正なブート変数がある場合はそれを削除し、正しいブート変数を追加します。
14. **write memory** コマンドを入力し、設定を running から startup に保存します。
15. **show bootvar** コマンドを使用し、ブート変数を再度確認して正しく設定されていることを確認し、スイッチが次回のリブート時に正しいシステム ファイルを起動するようにします。

# スイッチの回復に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

| 機能名 | リリース | 機能情報 |
|---|---|---|
| スイッチの回復 | Cisco IOS XE Everest 16.6.1 | イメージの消失または破損、あるいは不正なブート変数から Catalyst 9400 シリーズ スーパバイザを回復できます。 |