



Cisco IOS XE Bengaluru 17.5.x (Catalyst 9300 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：2021年4月1日

最終更新：2023年8月3日

シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター

0120-092-255 (フリーコール、携帯・PHS含む)

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>

【注意】 シスコ製品をご使用になる前に、安全上の注意（www.cisco.com/jp/go/safety_warning/）をご確認ください。本書は、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

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

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

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

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

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

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

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

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

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. 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)

© 2021 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の実装 7

システム名およびシステムプロンプト 8

デフォルトのシステム名とプロンプトの設定 9

DNS 9

DNSのデフォルト設定値 9

ログインバナー 9

バナーのデフォルト設定 10

MACアドレステーブル 10

MACアドレステーブルの作成 10

MACアドレスおよびVLAN 11

MACアドレステーブルのデフォルト設定 11

ARPテーブルの管理 11

デバイスの管理方法 12

手動による日付と時刻の設定 12

システムクロックの設定 12

タイムゾーンの設定 13

夏時間の設定 14

NTPの設定 16

NTPのデフォルト設定 16

NTP 認証の設定	16
ポーリング ベースの NTP アソシエーションの設定	18
ブロードキャスト ベースの NTP アソシエーションの設定	20
NTP アクセス制限の設定	22
システム名の設定	24
DNS の設定	26
Message-of-the-Day ログイン バナーの設定	27
ログイン バナーの設定	29
MAC アドレス テーブルの管理	30
アドレス エージング タイムの変更	30
MAC アドレス変更通知トラップの設定	31
MAC アドレス移動通知トラップの設定	34
MAC しきい値通知トラップの設定	36
VLAN の MAC アドレスラーニングのディセーブル化	38
スタティック アドレス エントリの追加および削除	39
ユニキャスト MAC アドレス フィルタリングの設定	40
デバイスのモニタリングおよび保守の管理	42
デバイス管理の設定例	43
例：システム クロックの設定	43
例：サマー タイムの設定	43
例：MOTD バナーの設定	43
例：ログイン バナーの設定	44
例：MAC アドレス変更通知トラップの設定	44
例：MAC しきい値通知トラップの設定	44
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	45
例：ユニキャスト MAC アドレス フィルタリングの設定	45
デバイス管理に関する追加情報	45
デバイス管理の機能履歴	45

第 2 章	ブート整合性の可視性	47
	ブート整合性の可視性について	47

イメージ署名とブートアップ	47
ソフトウェアイメージとハードウェアの確認	49
プラットフォーム ID とソフトウェア整合性の確認	49
イメージ署名の検証	53
ブート整合性の可視性に関する追加情報	54
ブート整合性の可視性の機能履歴	54

第 3 章

デバイスのセットアップ設定の実行	55
デバイスセットアップの設定の制約事項	55
デバイスセットアップ設定の実行に関する情報	55
デバイスブートプロセス	55
ソフトウェアインストールの概要	56
ソフトウェアのブートモード	57
ソフトウェアパッケージのインストール	58
ソフトウェアインストールの終了	59
デバイス情報の割り当て	59
デフォルトのスイッチ情報	60
DHCP ベースの自動設定の概要	60
DHCP クライアントの要求プロセス	61
DHCP ベースの自動設定およびイメージアップデート	62
DHCP ベースの自動設定の制約事項	62
DHCP 自動設定	63
DHCP 自動イメージアップデート	63
DHCP サーバ設定時の注意事項	63
TFTP サーバの目的	64
DNS サーバの目的	65
コンフィギュレーションファイルの入手方法	65
環境変数の制御方法	66
一般的な環境変数	68
TFTP の環境変数	70
ソフトウェアイメージのリロードのスケジューリング	70

デバイスセットアップ設定の実行方法	71
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	71
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	73
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	76
複数の SVI への IP 情報の手動割り当て	77
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	79
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	79
スイッチの手動による起動	81
インストール モードでのデバイスのブート	82
バンドルモードでのデバイスの起動	84
ソフトウェア イメージのリロードのスケジューリング設定	85
デバイスのセットアップの設定例	86
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	87
例: 緊急インストール	90
例: 更新プログラム パッケージの管理	91
ソフトウェア インストールの確認	101
例: デバイスを DHCP サーバとして設定	104
例: DHCP 自動イメージアップデートの設定	105
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	105
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	106
デバイスセットアップの実行に関する追加情報	106
デバイスセットアップ設定の実行に関する機能履歴	106

第 4 章

ポリシーを使用したスマートライセンス	109
ポリシーを使用したスマートライセンシングの概要	109
ポリシーを使用したスマートライセンシングに関する情報	110
概要	110
アーキテクチャ	111
製品インスタンス	111
CSLU	111

CSSM	112
コントローラ	112
SSM オンプレミス	113
概念	114
ライセンス執行（エンフォースメント）タイプ	114
ライセンス継続期間	115
承認コード	115
ポリシー	116
RUM レポートおよびレポート確認応答	118
信頼コード	118
サポートされるトポロジ	118
CSLU を介して CSSM に接続	119
CSSM に直接接続	120
コントローラを介して CSSM に接続	122
CSLU は CSSM から切断	123
CSSM への接続なし、CSLU なし	124
SSM オンプレミス展開	125
サポート対象製品	128
他の機能との相互作用	129
ハイ アベイラビリティ	129
アップグレード	130
ダウングレード	133
ポリシーを使用したスマートライセンスिंगの設定方法：トポロジ別のワークフロー	136
トポロジのワークフロー：CSLU を介して CSSM に接続	137
トポロジのワークフロー：CSSM に直接接続	139
トポロジのワークフロー：コントローラを介して CSSM に接続	141
トポロジのワークフロー：CSLU は CSSM から切断	142
トポロジのワークフロー：CSSM への接続なし、CSLU なし	145
トポロジのワークフロー：SSM オンプレミス展開	146
製品インスタンス開始型通信の場合のタスク	146
SSM オンプレミスインスタンス開始型通信の場合のタスク	149

ポリシーを使用したスマートライセンスへの移行	152
例：スマートライセンスからポリシーを使用したスマートライセンスへ	153
例：RTU ライセンスからポリシーを使用したスマートライセンスへ	158
例：SLR からポリシーを使用したスマートライセンスへ	162
例：評価ライセンスまたは期限切れライセンスからポリシーを使用したスマートライセンスへ	170
Smart Licensing Using Policy をサポートする SSM オンプレミスのバージョンへの移行	173
ポリシーを使用したスマートライセンスのタスクライブラリ	175
シスコへのログイン (CSLU インターフェイス)	175
スマートアカウントとバーチャルアカウントの設定 (CSLU インターフェイス)	176
CSLU での製品開始型製品インスタンスの追加 (CSLU インターフェイス)	176
製品インスタンス開始型通信のネットワーク到達可能性の確認	177
CSLU での CSLU 開始型製品インスタンスの追加 (CSLU インターフェイス)	178
使用状況レポートの収集：CSLU 開始	179
Download All For Cisco (CSLU インターフェイス)	180
Upload From Cisco (CSLU インターフェイス)	181
CSLU 開始型通信のネットワーク到達可能性の確認	181
CSSM への接続の設定	186
HTTPS プロキシを介したスマート転送の設定	189
ダイレクトクラウドアクセス用の Call Home サービスの設定	190
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	193
スマートアカウントとバーチャルアカウントの割り当て (SSM オンプレミス UI)	194
デバイスの検証 (SSM オンプレミス UI)	195
製品インスタンス開始型通信のネットワーク到達可能性の確認	196
トランスポート URL の取得 (SSM オンプレミス UI)	198
使用状況データのエクスポートとインポート (SSM オンプレミス UI)	199
1 つ以上の製品インスタンスの追加 (SSM オンプレミス UI)	200
SSM オンプレミス開始型通信のネットワーク到達可能性の確保	201
承認コードの削除と返却	207
CSSM からの製品インスタンスの削除	209

CSSM からの信頼コード用新規トークンの生成	210
信頼コードのインストール	211
CSSM からのポリシーファイルのダウンロード	212
CSSM への使用状況データのアップロードと ACK のダウンロード	213
製品インスタンスへのファイルのインストール	214
転送タイプ、URL、およびレポート間隔の設定	215
基本ライセンスまたはアドオンライセンスの設定	219
リソース使用率測定レポートの例	223
ポリシーを使用したスマートライセンスングのトラブルシューティング	223
システム メッセージの概要	223
システム メッセージ	225
ポリシーを使用したスマートライセンスングのその他の参考資料	235
ポリシーを使用したスマートライセンスングの機能の履歴	236

第 5 章

有線ネットワークでの Application Visibility and Control の設定	239
有線ネットワークでの Application Visibility and Control について	239
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	240
有線 Application Visibility and Control の制限	241
Application Visibility and Control の設定方法	243
有線ネットワークでの Application Visibility and Control の設定	243
インターフェイスでのアプリケーション認識の有効化	244
AVC QoS ポリシーの作成	244
スイッチ ポートへの QoS ポリシーの適用	247
有線 AVC モニターの一致および収集フィールドのサポート強化	248
有線 AVC Flexible Netflow の設定	249
NBAR2 カスタム アプリケーション	267
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	270
Application Visibility and Control のモニターリング	273
例：Application Visibility and Control の設定	273
基本的なトラブルシューティング：質問と回答	285
Application Visibility and Control に関する追加情報	286

有線ネットワークでの Application Visibility and Control の機能履歴 287

第 6 章

SDM テンプレートの設定 289

SDM テンプレートに関する情報 289

SDM テンプレートの設定方法 289

SDM テンプレートの設定 289

SDM テンプレートのモニタリングおよびメンテナンス 290

SDM テンプレートの設定例 291

例：SDM テンプレートの表示 291

例：SDM テンプレートの設定 293

SDM テンプレートに関する追加情報 294

SDM テンプレートの機能履歴 294

第 7 章

システム メッセージ ログの設定 297

システム メッセージ ログの設定に関する情報 297

システム メッセージ ロギング 297

システム ログ メッセージのフォーマット 298

デフォルトのシステム メッセージ ロギングの設定 299

syslog メッセージの制限 300

システム メッセージ ログの設定方法 300

メッセージ表示宛先デバイスの設定 300

ログ メッセージの同期化 302

メッセージ ロギングのディセーブル化 304

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 305

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 306

メッセージ重大度の定義 306

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 307

UNIX Syslog デーモンへのメッセージのロギング 308

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 309

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 309

システム メッセージ ログの設定例 310

例：システム メッセージのスタック構成 310

例：スイッチ システム メッセージ 310

システム メッセージ ログに関する追加情報 311

システムメッセージログの機能履歴 311

第 8 章

オンライン診断の設定 313

オンライン診断の設定に関する情報 313

Generic Online Diagnostics (GOLD) テスト 314

オンライン診断の設定方法 319

オンライン診断テストの開始 319

オンライン診断の設定 320

オンライン診断のスケジューリング 320

ヘルス モニタリング診断の設定 322

オンライン診断のモニタリングおよびメンテナンス 325

オンライン診断のコンフィギュレーション例 325

例：診断テストの開始 326

例：ヘルスマニターリングテストの設定 326

例：診断テストのスケジューリング 326

例：オンライン診断の表示 326

オンライン診断に関する追加情報 327

オンライン診断設定の機能情報 328

第 9 章

整合性チェッカー 329

整合性チェッカーの制限事項 329

整合性チェッカーに関する情報 330

整合性チェッカーの実行 330

整合性チェッカーの機能履歴 331

第 10 章

コンフィギュレーション ファイルの管理 333

コンフィギュレーション ファイルの管理の前提条件 333

コンフィギュレーション ファイルの管理の制約事項 333

次の作業	355
NVRAM より大きいコンフィギュレーション ファイルの保守	355
コンフィギュレーション ファイルの圧縮	355
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	356
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	358
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	359
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	360
FTP サーバーからフラッシュ 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	361
次の作業	362
RCP サーバーからフラッシュ 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	363
TFTP サーバーからフラッシュ 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	364
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	364
スタートアップ コンフィギュレーションのクリア	365
指定されたコンフィギュレーション ファイルの削除	366
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	367
次の作業	369
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	370
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	370
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	371
コンフィギュレーション ファイルの管理の機能履歴	373
第 11 章	セキュア コピー 375
セキュア コピーの前提条件	375
Secure Copy に関する情報	375
セキュアコピーのパフォーマンス向上	376

セキュアコピーの設定方法	376
セキュアコピーの設定	376
SSH サーバーでのセキュアコピーのイネーブル化	377
セキュアコピーの設定例	379
例：ローカル認証を使用したセキュアコピーの設定	379
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ー側の設定	380
セキュアコピーに関する追加情報	380
セキュアコピーの機能情報	381

第 12 章

コンフィギュレーションの置換とロールバック	383
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	383
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	384
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	384
コンフィギュレーションアーカイブ	384
コンフィギュレーションの置換	385
コンフィギュレーションロールバック	386
コンフィギュレーションロールバック変更確認	387
コンフィギュレーションの置換とロールバックの利点	387
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	387
コンフィギュレーションアーカイブの作成	387
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	389
機能のモニターリングおよびトラブルシューティング	393
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	395
コンフィギュレーションアーカイブの作成	395
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	395
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	396
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	396
コンフィギュレーションロールバック操作の実行	397
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	398

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	398
--	-----

第 13 章

BIOS 保護 399

BIOS 保護の概要	399
ROMMON アップグレード	399
カプセルアップグレード	400
BIOS 保護の機能履歴	401

第 14 章

Extended Fast Software Upgrade の実行 403

Extended Fast Software Upgrade の前提条件	403
Extended Fast Software Upgrade の制約事項	403
Extended Fast Software Upgrade に関する情報	404
Extended Fast Software Upgrade でサポートされるプロトコル	405
スタンドアロンスイッチでの Extended Fast Software Upgrade の実行方法	405
スタンドアロンスイッチでのソフトウェアのアップグレード	406
IPv6 が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード	406
IPv6 MLD が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード	407
BGP が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード	408
OSPFv3 が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード	409
スタンドアロンスイッチでのソフトウェアのリロード	411
BGP が設定されたスタンドアロンスイッチでのソフトウェアのリロード	411
OSPFv3 が設定されたスタンドアロンスイッチでのソフトウェアのリロード	412
スタック構成スイッチでの Extended Fast Software Upgrade の実行方法	414
スタック構成スイッチでのソフトウェアのアップグレード	414
BGP が設定されたスタック構成スイッチでのソフトウェアのアップグレード	415
IS-IS が設定されたスタック構成スイッチでのソフトウェアのアップグレード	416
スタック構成スイッチでのソフトウェアのリロード	417
BGP が設定されたスタック構成スイッチでのソフトウェアのリロード	418
IS-IS が設定されたスタック構成スイッチでのソフトウェアのリロード	419
ソフトウェアのアップグレードまたはリロードの確認	420

その他の参考資料	420
Extended Fast Software Upgrade の機能履歴	421

第 15 章

ソフトウェア メンテナンス アップグレード	423
ソフトウェア メンテナンス アップグレードの制約事項	423
ソフトウェア メンテナンス アップグレードについて	423
SMU の概要	423
SMU のワークフロー	424
SMU パッケージ	424
SMU のリロード	424
ソフトウェア メンテナンスの更新の管理方法	425
SMU パッケージのインストール	425
SMU パッケージの管理	426
ソフトウェア メンテナンス アップグレードの設定例	427
例 : SMU の管理	427
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	432
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	432

第 16 章

フラッシュ ファイル システムの操作	435
フラッシュ ファイル システムについて	435
使用可能なファイル システムの表示	435
デフォルト ファイル システムの設定	440
ファイル システムのファイルに関する情報の表示	441
ディレクトリの変更および作業ディレクトリの表示	442
ディレクトリの作成	443
ディレクトリの削除	443
ファイルのコピー	444
ファイルの削除	445
ファイルの作成、表示、および抽出	445
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	448
フラッシュファイルシステムの機能履歴	448

第 17 章**初期設定へのリセットの実行 449**

- 初期設定へのリセット実行の前提条件 449
- 初期設定へのリセット実行の制限事項 449
- 初期設定へのリセットの実行に関する情報 450
- 初期設定へのリセットの実行方法 451
- 初期設定へのリセットを実行するための設定例 452
- 初期設定へのリセットの実行に関する追加情報 454
- 初期設定へのリセットに関する機能履歴 454

第 18 章**セキュアストレージの設定 457**

- セキュアストレージについて 457
- セキュアストレージの有効化 457
- セキュアストレージの無効化 458
- 暗号化のステータスの確認 459
- セキュアストレージの機能情報 459

第 19 章**条件付きデバッグとラジオアクティブ トレース 461**

- 条件付きデバッグの概要 461
- ラジオアクティブ トレースの概要 462
- 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 462
 - 条件付きデバッグおよび放射線 トレース 462
 - トレースファイルの場所 462
 - 条件付きデバッグの設定 463
- L2 マルチキャストの放射線 トレース 465
 - トレース ファイルの推奨ワークフロー 465
- ボックス外へのトレース ファイルのコピー 466
- 条件付きデバッグのモニターリング 466
- 条件付きデバッグの設定例 467
- 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 468
- 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 468

第 20 章	同意トークン	469
	同意トークンの制約事項	469
	同意トークンに関する情報	470
	システムシェルアクセスの同意トークン承認プロセス	470
	同意トークンの機能履歴	472

第 21 章	ソフトウェア設定のトラブルシューティング	473
	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	473
	スイッチのソフトウェア障害	473
	デバイスのパスワードを紛失したか忘れた場合	474
	Power over Ethernet (PoE) ポート	474
	電力消失によるポートの障害	475
	不正リンク アップによるポート障害	475
	ping	475
	レイヤ 2 トレースルート	475
	レイヤ 2 の traceroute のガイドライン	476
	IP トレースルート	477
	Time Domain Reflector ガイドライン	478
	debug コマンド	479
	システム レポート	479
	スイッチのオンボード障害ロギング	482
	ファン障害	482
	CPU 使用率が高い場合に起こりうる症状	483
	ソフトウェア設定のトラブルシューティング方法	483
	ソフトウェア障害からの回復	483
	パスワードを忘れた場合の回復	487
	パスワード回復がイネーブルになっている場合の手順	489
	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	490
	自動ネゴシエーションの不一致の防止	492
	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	493

SFP モジュール ステータスのモニタリング	494
ping の実行	494
温度のモニタリング	494
物理パスのモニタリング	495
IP traceroute の実行	495
TDR の実行および結果の表示	495
デバッグおよびエラー メッセージ出力のリダイレクト	495
show platform forward コマンドの使用	496
show debug コマンドの使用方法	496
OBFL の設定	497
ソフトウェア設定のトラブルシューティングの確認	497
OBFL 情報の表示	497
例：高い CPU 使用率に関する問題と原因の確認	498
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	500
Power over Ethernet (PoE) に関するトラブルシューティングのシナリオ	500
ソフトウェアのトラブルシューティングの設定例	505
例：IP ホストの ping	505
例：IP ホストに対する traceroute の実行	506
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	507
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	507

第 22 章	回線の自動統合	509
	回線の自動統合	509
	回線の自動統合の機能履歴	515



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 (12 ページ)
- デバイス管理の設定例 (43 ページ)
- デバイス管理に関する追加情報 (45 ページ)
- デバイス管理の機能履歴 (45 ページ)

デバイスの管理に関する情報

システム日時の管理

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



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

システムクロック

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

システムクロックは、次のソースにより設定できます。

- RTC
- NTP
- 手動設定

システムクロックは、次のサービスに時刻を提供します。

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

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

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

ネットワーク タイム プロトコル

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

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

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

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

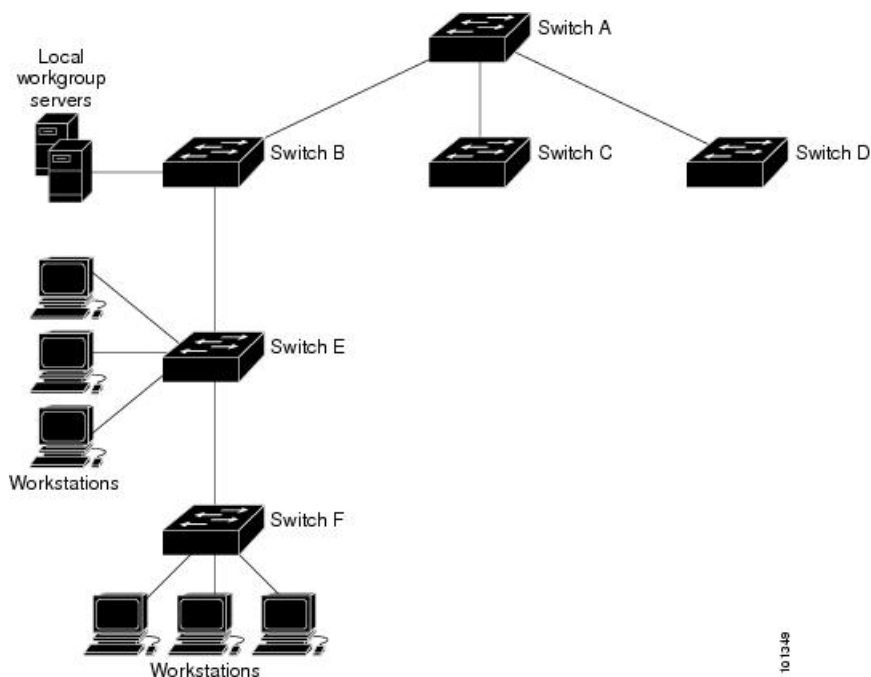
NTP が稼働するデバイス間の通信 (アソシエーション) は、通常静的に設定されます。各デバイスには、アソシエーションを作成すべきすべてのデバイスの IP アドレスが与えられます。アソシエーションのペアとなるデバイス間で NTP メッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN 環境では、代わりに IP ブロードキャストメッセージを使用するように NTP を設定できます。各デバイスを、単にブロードキャストメッセージを送受信するように設定すればよいので、この代替手段によって設定の複雑さが緩和されます。ただし、情報の流れは一方向に限られます。

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

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

次の図に NTP を使用した一般的なネットワークの例を示します。A はプライマリ NTP、デバイス B、C、D が NTP サーバーモードに設定されている（デバイス A との間にサーバー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。

ポーリング ベースの NTP アソシエーション

NTP を実行しているネットワーク デバイスは、時刻を基準時刻源と同期する際にさまざまなアソシエーションモードで動作するように設定できます。ネットワーク デバイスは、2つの方法でネットワーク上の時刻情報を取得できます。それらは、ホストサービスのポーリングと NTP ブロードキャストのリスニングです。ここでは、ポーリングベースのアソシエーションモードを中心に説明します。ブロードキャストベースの NTP アソシエーションの詳細については、「ブロードキャストベースの NTP アソシエーション」を参照してください。

最も一般的に使用される 2 つのポーリングベースのアソシエーションモードは次のとおりです。

- クライアント モード
- 対称アクティブ モード

クライアントモードと対称アクティブモードは、高レベルの時刻の精度と信頼性を提供するために NTP が必要になる場合に使用します。

クライアントモードで動作しているネットワーク デバイスは、自身に割り当てられている時刻提供ホストをポーリングして現在の時刻を取得します。次に、ネットワーク デバイスは、ポーリングされたすべてのタイムサーバーから、同期に使用するホストを選択します。この場合は、確立された関係がクライアントホスト関係なので、ホストがローカルクライアントデバイスから送信された時刻情報をキャプチャしたり使用したりすることはありません。このモードが最も適しているのは、他のローカルクライアントにどのような形式の時刻同期も提供する必要のない、ファイルサーバーおよびワークステーションのクライアントです。ネットワーク デバイスを同期させるタイムサーバーを個別に指定し、クライアントモードで動作するようにネットワーク デバイスを設定するには、**ntp server** コマンドを使用します。

対称アクティブモードで動作しているネットワークング デバイスは、自身に割り当てられている時刻提供ホストをポーリングして現在の時刻を取得し、そのホストによるポーリングに応答します。これはピアツーピアの関係なので、ホストは、通信相手のローカルネットワークング デバイスの時刻関連情報も保持します。このモードは、さまざまなネットワーク パスを経由で多数の冗長サーバーが相互接続されている場合に使用します。インターネット上のほとんどの Stratum 1 および Stratum 2 サーバーは、この形式のネットワーク設定を採用しています。ネットワークング デバイスを同期させる時刻提供ホストを個別に指定し、対称アクティブモードで動作するようにネットワークング デバイスを設定するには、**ntp peer** コマンドを使用します。

各ネットワークング デバイスの設定モードを決定する際には、タイムキーピング デバイスとしてのそのデバイスの役割（サーバーかクライアントか）と、そのデバイスが Stratum 1 タイムキーピング サーバーにどれだけ近いかを主に考慮してください。

ネットワークング デバイスは、クライアントモードでクライアントまたはホストとして動作する場合、または対称アクティブモードでピアとして動作する場合にポーリングに関与します。通常、ポーリングによってメモリおよび CPU リソース（帯域幅など）に負荷が生じることはありませんが、システム上で進行または同時実行しているポーリングの数がきわめて多い場合には、システムの性能に深刻な影響があったり、特定のネットワークの性能が低下したりする可能性があります。過剰な数のポーリングがネットワーク上で進行することを防止するには、直接的なピアツーピアアソシエーションまたはクライアントからサーバーへのアソシエーションを制限する必要があります。代わりに、局所的なネットワーク内に NTP ブロードキャストを使用して時刻情報を伝播することを検討します。

ブロードキャストベースの NTP アソシエーション

ブロードキャストベースの NTP アソシエーションは、時刻の精度および信頼性要件が適度であり、ネットワークが局所的であり、クライアント数が 20 を超える場合に使用します。また、帯域幅、システムメモリ、または CPU リソースが制限されているネットワークにおいても、ブロードキャストベースの NTP アソシエーションの使用をお勧めします。

ブロードキャストクライアントモードで動作しているネットワークング デバイスはポーリングに関与しません。代わりに、ブロードキャストタイムサーバーによって転送される NTP ブロードキャストパケットを待ち受けます。その結果、時刻情報の流れが一方向に限られるため、時刻の精度がわずかに低下する可能性があります。

ネットワークを通じて伝播される NTP ブロードキャストパケットをリッスンするようにネットワークング デバイスを設定するには、**ntp broadcast client** コマンドを使用します。ブロードキャストクライアントモードが動作するためには、ブロードキャストサーバーとそのクライアントが同じサブネット上に存在する必要があります。**ntp broadcast** コマンドを使用して、特定のデバイスのインターフェイスで NTP ブロードキャストパケットを送信するタイムサーバーを有効にする必要があります。

NTP セキュリティ

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



- (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 アクセス グループ

アクセスリストベースの制限スキームを使用すると、ネットワーク全体、ネットワーク内のサブネット、またはサブネット内のホストに対し、特定のアクセス権限を許可または拒否できます。NTP アクセスグループを定義するには、グローバル コンフィギュレーション モードで `ntp access-group` コマンドを使用します。

アクセスグループのオプションは、次の順序で制限の緩いものから厳しいものへとスキャンされます。

1. `ipv4` : IPv4 アクセスリストを設定します。
2. `ipv6` : IPv6 アクセスリストを設定します。
3. `peer` : 時刻要求と NTP 制御クエリを許可し、システムがアクセスリストの基準を満たすアドレスを持つ別のシステムに同期することを許可します。
4. `serve` : 時刻要求と NTP 制御クエリを許可しますが、システムがアクセスリストの基準を満たすアドレスを持つ別のシステムに同期することは許可しません。
5. `serve-only` : アクセスリストの条件を満たすアドレスを持つシステムからの時刻要求のみを許可します。
6. `query-only` : アクセスリストの基準を満たすアドレスを持つ別のシステムからの NTP 制御クエリのみを許可します。

送信元 IP アドレスが複数のアクセス タイプのアクセス リストに一致する場合は、最初のアクセス タイプのアクセスが認可されます。アクセス グループが指定されていない場合は、すべてのシステムへのアクセスがすべてのアクセス タイプに対して認可されます。アクセスグループが指定されている場合は、指定されたアクセス タイプに対してのみアクセスが認可されます。

NTP 制御クエリーの詳細については、RFC 1305 (NTP バージョン 3) を参照してください。

信頼できる形式のアクセス コントロールが必要な場合は、暗号化された NTP 認証方式を使用する必要があります。IP アドレスに基づくアクセス リストベースの制約方式とは異なり、暗号化認証方式では、認証キーと認証プロセスを使用して、ローカルネットワーク上の指定されたピアまたはサーバーによって送信された NTP 同期パケットが信頼できると見なされるかどうかを、一緒に伝送された時刻情報を受け入れる前に判断します。

認証プロセスは、NTP パケットが作成されるとすぐに開始されます。暗号チェックサム キーは、Message-Digest Algorithm 5 (MD5) を使用して生成され、受信側クライアントに送信される NTP 同期パケットに埋め込まれます。パケットがクライアントによって受信されると、暗号チェックサム キーが復号され、信頼できるキーのリストに対してチェックされます。一致する認証キーがパケットに含まれる場合、受信側クライアントは、パケットに含まれるタイムス

タンブ情報を受け入れます。一致するオーセンティケータ キーが含まれていない NTP 同期パケットは無視されます。



- (注) 信頼できるキーを多数設定する必要がある大規模なネットワークでは、信頼できるキーの範囲設定機能を使用して複数のキーを同時に有効にすることができます。

NTP 認証で使用される暗号化および復号化プロセスでは、CPU に非常に大きな負荷がかかる場合があります。ネットワーク内で伝播される時刻の精度が大きく低下する可能性があることに注意してください。より包括的なアクセス コントロール モデルを使用できるネットワーク構成の場合は、アクセス リスト ベースのコントロール方式を使用することを検討してください。

NTP 認証が適切に設定されると、ネットワーキングデバイスは、信頼できる時刻源と同期し、信頼できる時刻源だけに同期を提供します。

特定のインターフェイス上の NTP サービス

Network Time Protocol (NTP) サービスは、デフォルトではすべてのインターフェイスで無効になっています。なんらかの NTP コマンドを入力すると、NTP がグローバルに有効になります。特定のインターフェイスを通じて特定の NTP パケットを受信しないように設定するには、インターフェイス コンフィギュレーション モードで **ntp disable** コマンドを使用します。

NTP パケットの送信元 IP アドレス

システムが NTP パケットを送信すると、通常、送信元 IP アドレスは、その NTP パケットの送信元であるインターフェイスのアドレスに設定されます。IP 送信元アドレスの取得元のインターフェイスを設定するには、グローバルコンフィギュレーションモードで **ntp source interface** コマンドを使用します。

このインターフェイスは、すべての宛先に送信されるすべてのパケットの送信元アドレスに使用されます。特定のアソシエーションに送信元アドレスを使用する場合は、**ntp peer** コマンドまたは **ntp server** コマンドで **source** キーワードを使用します。

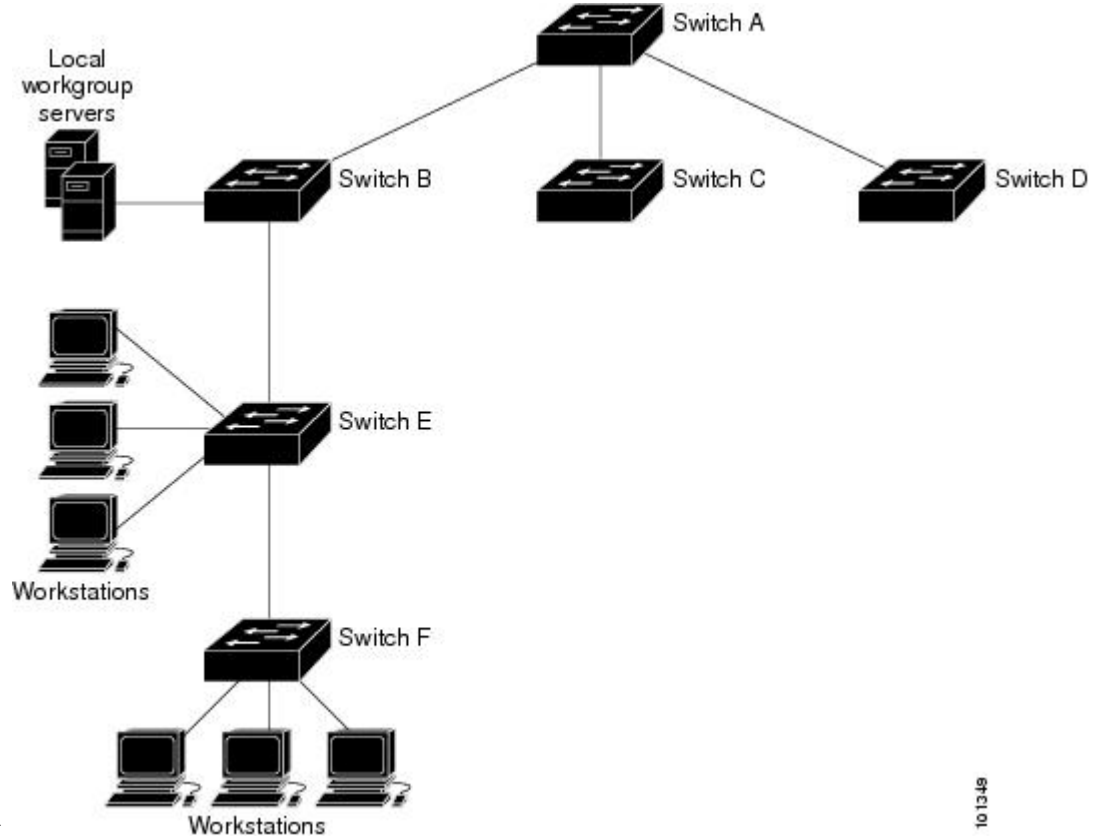
NTP の実装

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 サーバーモードに設定されている（スイッチ A との間にサーバー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

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。

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せん。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。

エイ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 テーブルのデフォルト設定

次の表に、MAC アドレス テーブルのデフォルト設定を示します。

表 2: MAC アドレスのデフォルト設定

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

ARP テーブルの管理

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

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



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

システムクロックの設定

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

システムクロックを設定するには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>year</i> : 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 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	clock timezone zone hours-offset [minutes-offset] 例 : Device (config)# clock timezone AST -3 30	時間帯を設定します。 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。 <ul style="list-style-type: none"> • <i>zone</i> : 標準時が適用されているときに表示されるタイムゾーンの名前を入力します。デフォルトは UTC です。 • <i>hours-offset</i> : UTC からのオフセット時間数を入力します。 • (任意) <i>minutes-offset</i> : UTC からのオフセット分数を入力します。ローカルタイムゾーンが UTC と 1 時間の差の割合である場合に指定できます。
ステップ 4	end 例 :	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	Device (config) # end	
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

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 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	clock summer-time zone date date month year hh:mm date month year hh:mm [offset] 例： Device (config) # clock summer-time PDT date 10 March 2013 2:00 3 November 2013 2:00	毎年指定された日に開始および終了する夏時間を設定します。

	コマンドまたはアクション	目的
<p>ステップ 4</p>	<p>clock summer-time zone recurring [<i>week day month hh:mm week day month hh:mm</i> [<i>offset</i>]]</p> <p>例 :</p> <pre>Device(config)# clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00</pre>	<p>毎年指定された日に開始および終了する夏時間を設定します。すべての時刻は、現地のタイムゾーンを基準にしています。開始時間は標準時を基準にしています。</p> <p>終了時間は夏時間を基準にしています。夏時間はデフォルトでディセーブルに設定されています。パラメータなしで clock summer-time zone recurring を指定すると、夏時間のルールは米国のルールにデフォルト設定されます。</p> <p>開始月が終了月より後の場合は、システムでは南半球にいると見なされます。</p> <ul style="list-style-type: none"> • <i>zone</i> : 夏時間が有効な場合に表示される時間帯名 (PDT など) を指定します。 • (任意) <i>week</i> : 月の週 (1 ~ 4、first、または last) を指定します。 • (任意) <i>day</i> : 曜日 (Sunday、Monday など) を指定します。 • (任意) <i>month</i> : 月 (January、February など) を指定します。 • (任意) <i>hh:mm</i> : 時および分単位で時間 (24時間形式) を指定します。 • (任意) <i>offset</i> : 夏時間中に追加する分数を指定します。デフォルトは 60 です。
<p>ステップ 5</p>	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
<p>ステップ 6</p>	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	<p>入力を確認します。</p>

	コマンドまたはアクション	目的
ステップ 7	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

NTP の設定

デバイスはハードウェアサポートクロックを備えておらず、外部 NTP ソースが利用できないときに、ピアが自身を同期化するための NTP プライマリクロックとして機能することはできません。デバイスは、カレンダーに対するハードウェアサポートも備えていません。そのため、グローバル コンフィギュレーション モードで **ntp update-calendar** コマンドと **ntp master** コマンドを使用することはできません。

NTP の設定情報については、次のセクションを参照してください。

NTP のデフォルト設定

NTP のデフォルト設定を示します。

表 3: NTP のデフォルト設定

機能	デフォルト設定
NTP 認証	ディセーブル認証キーは指定されていません。
NTP ピアまたはサーバー アソシエーション	未設定
NTP ブロードキャスト サービス	ディセーブル。どのインターフェイスも NTP ブロードキャストパケットを送受信しません。
NTP アクセス制限	アクセスコントロールは指定されていません。
NTP パケット送信元 IP アドレス	送信元アドレスは、発信インターフェイスによって設定されます。

NTP は、すべてのインターフェイスでデフォルトでイネーブルに設定されています。すべてのインターフェイスは、NTP パケットを受信します。

NTP 認証の設定

NTP 認証を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <p>パスワードを入力します (要求された場合)。</p>
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>[no] ntp authenticate</p> <p>例 :</p> <pre>Device(config)# ntp authenticate</pre>	<p>NTP 認証をイネーブルにします。</p> <p>NTP 認証を無効にするには、このコマンドの no 形式を使用します。</p>
ステップ 4	<p>[no] ntp authentication-key number {md5 cmac-aes-128 hmac-sha1 hmac-sha2-256} value</p> <p>例 :</p> <pre>Device(config)# ntp authentication-key 42 md5 aNiceKey</pre>	<p>認証キーを定義します。</p> <ul style="list-style-type: none"> • キーごとに、キー番号、タイプ、および値を 1 つずつ指定します。 • キーは次のいずれかのタイプになります。 <ul style="list-style-type: none"> • md5 : MD5 アルゴリズムを使用した認証。 • cmac-aes-128 : AES-128 アルゴリズムによる暗号ベースメッセージ承認コード (CMAC) を使用した認証。ダイジェストの長さは 128 ビットで、キーの長さは 16 バイトまたは 32 バイトです。 • hmac-sha1 : SHA1 ハッシュ関数を使用したハッシュベースメッセージ承認コード (HMAC) を使用した認証。ダイジェストの長さは 128 ビットで、キーの長さは 1 ~ 32 バイトです。 • hmac-sha2-256 : SHA2 ハッシュ関数を使用した HMAC を使用

	コマンドまたはアクション	目的
		<p>した認証。ダイジェストの長さは256ビットで、キーの長さは1～32バイトです。</p> <p>SNTPの認証キーを削除する場合は、このコマンドの no 形式を使用します。</p>
ステップ 5	<p>[no] ntp trusted-key key-number</p> <p>例 :</p> <pre>Device(config)# ntp trusted-key 42</pre>	<p>このデバイスと同期できるようにするために、ピア NTP デバイスが NTP パケットで提供する必要がある信頼できる認証キーを定義します。</p> <p>信頼できる認証を無効にするには、このコマンドの no 形式を使用します。</p>
ステップ 6	<p>[no] ntp server ip-address key key-id [prefer]</p> <p>例 :</p> <pre>Device(config)# ntp server 172.16.22.44 key 42</pre>	<p>NTPタイムサーバーによってソフトウェアクロックが同期されるように設定します。</p> <ul style="list-style-type: none"> • ip-address : クロック同期を提供するタイムサーバーの IP アドレス。 • key-id : ntp authentication-key コマンドで定義された認証キー。 • prefer : このピアを、同期を提供する優先ピアにします。このキーワードは、ピア間のクロックホップを減らします。 <p>サーバーアソシエーションを解除するには、このコマンドの no 形式を入力します。</p>
ステップ 7	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>

ポーリングベースの NTP アソシエーションの設定

ポーリングベースの NTP アソシエーションを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <p>パスワードを入力します (要求された場合)。</p>
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>[no] ntp peer ip-address [version number] [key key-id] [source interface] [prefer]</p> <p>例 :</p> <pre>Device(config)# ntp peer 172.16.22.44 version 2</pre>	<p>ピアを同期化するか、またはピアによって同期化されるように、デバイスのシステムクロックを設定します (ピアアソシエーション)。</p> <ul style="list-style-type: none"> • ip-address : クロック同期を提供する、またはクロック同期を提供されるピアの IP アドレス。 • number : NTP バージョン番号。範囲は、1 ~ 3 です。デフォルトでは、バージョン 3 が選択されています。 • key-id : ntp authentication-key コマンドで定義された認証キー。 • interface : IP の送信元アドレスを取得するインターフェイス。デフォルトでは、送信元 IP アドレスは発信インターフェイスから取得します。 • prefer : このピアを、同期を提供する優先ピアにします。このキーワードにより、ピア間の切り替えが減少します。 <p>ピアアソシエーションを解除するには、このコマンドの no 形式を使用します。</p>
ステップ 4	<p>[no] ntp server ip-address [version number] [key key-id] [source interface] [prefer]</p> <p>例 :</p>	<p>タイムサーバーによって同期化されるように、デバイスのシステムクロックを設定します (サーバーアソシエーション)。</p>

	コマンドまたはアクション	目的
	<pre>Device(config)# ntp server 172.16.22.44 version 2</pre>	<ul style="list-style-type: none"> • ip-address : クロック同期を提供するタイムサーバーの IP アドレス。 • number : NTP バージョン番号。範囲は、1 ~ 3 です。デフォルトでは、バージョン3が選択されています。 • key-id : ntp authentication-key コマンドで定義された認証キー。 • interface : IP の送信元アドレスを取得するインターフェイス。デフォルトでは、送信元 IP アドレスは発信インターフェイスから取得します。 • prefer : このピアを、同期を提供する優先ピアにします。このキーワードは、ピア間のクロックホップを減らします。 <p>サーバーアソシエーションを解除するには、このコマンドの no 形式を入力します。</p>
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>

ブロードキャストベースの NTP アソシエーションの設定

ブロードキャストベースの NTP アソシエーションを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <p>パスワードを入力します (要求された場合)。</p>
ステップ 2	<p>configure terminal</p> <p>例 :</p>	<p>グローバル コンフィギュレーションモードを開始します。</p>

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	interface interface-id 例 : Device(config)# interface gigabitethernet1/0/1	インターフェイスを設定し、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 4	[no] ntp broadcast [version number] [key key-id] [destination-address] 例 : Device(config-if)# ntp broadcast version 2	NTP ブロードキャスト パケットをピアに送信するインターフェイスをイネーブルにします。 <ul style="list-style-type: none"> • <i>number</i> : NTP バージョン番号。範囲は、1 ~ 3 です。デフォルトでは、バージョン 3 が使用されます。 • <i>key-id</i> : 認証キー。 • <i>destination-address</i> : このスイッチに対してクロックを同期しているピアの IP アドレス。 <p>インターフェイスでの NTP ブロードキャスト パケットの送信を無効にするには、このコマンドの no 形式を使用します。</p>
ステップ 5	[no] ntp broadcast client 例 : Device(config-if)# ntp broadcast client	インターフェイスが NTP ブロードキャスト パケットを受信できるようにします。 インターフェイスでの NTP ブロードキャスト パケットの受信を無効にするには、このコマンドの no 形式を使用します。
ステップ 6	exit 例 : Device(config-if)# exit	特権 EXEC モードに戻ります。
ステップ 7	[no] ntp broadcastdelay microseconds 例 : Device(config)# ntp broadcastdelay 100	(任意) デバイスと NTP ブロードキャスト サーバー間のラウンドトリップ遅延の予測値を変更します。 デフォルトは 3000 マイクロ秒です。範囲は 1 ~ 999999 です。

	コマンドまたはアクション	目的
ステップ 8	end 例： Device (config) # end	インターフェイスでの NTP ブロードキャストパケットの受信を無効にするには、このコマンドの no 形式を使用します。 特権 EXEC モードに戻ります。

NTP アクセス制限の設定

以降で説明するように、2つのレベルで NTP アクセスを制御できます。

アクセスグループの作成と基本 IP アクセスリストの割り当て

アクセスグループを作成して基本 IP アクセスリストを割り当てるには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	[no] ntp access-group {query-only serve-only serve peer} access-list-number 例： Device (config) # ntp access-group peer 99	アクセスグループを作成し、基本 IP アクセスリストを割り当てます。 <ul style="list-style-type: none"> • query-only : NTP 制御クエリ。 • serve-only : 時間要求。 • serve : 時刻要求と NTP 制御クエリは許可しますが、リモートデバイスに対するデバイスの同期化は許可しません。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • peer : 時刻要求とNTP制御クエリ、およびリモートデバイスに対するデバイスの同期化を許可します。 • access-list-number : IPアクセスリスト番号。指定できる範囲は1～99です。 <p>スイッチNTPサービスに対するアクセス制御を削除するには、このコマンドの no 形式を使用します。</p>
<p>ステップ4</p>	<p>access-list access-list-number permit source [source-wildcard]</p> <p>例 :</p> <pre>Device(config)# access-list 99 permit 172.20.130.5</pre>	<p>アクセスリストを作成します。</p> <ul style="list-style-type: none"> • access-list-number : IPアクセスリスト番号。指定できる範囲は1～99です。 • permit : 条件が一致した場合にアクセスを許可します。 • source : デバイスへのアクセスが許可されているデバイスのIPアドレス。 • source-wildcard : 送信元アドレスに適用されるワイルドカードビット。 <p>(注) アクセスリストを作成する際は、アクセスリストの末尾に暗黙の deny ステートメントがデフォルトで存在し、ACLの終わりに到達するまで一致が見つからなかったすべてのパケットに適用されることに注意してください。</p> <p>SNTPの認証キーを削除する場合は、このコマンドの no 形式を使用します。</p>
<p>ステップ5</p>	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>

特定のインターフェイス上の NTP サービスのディセーブル化

インターフェイスで NTP パケットの受信を無効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet1/0/1	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	[no] ntp disable 例： Device(config-if)# ntp disable	インターフェイスで NTP パケットの受信をディセーブルにします。 インターフェイスで NTP パケットの受信を再度有効にするには、このコマンドの no 形式を使用します。
ステップ 5	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。

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	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>hostname name</p> <p>例 :</p> <pre>Device(config)# hostname remote-users</pre>	<p>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。</p> <p>デフォルト設定は Switch です。</p> <p>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。</p>
ステップ 4	<p>end</p> <p>例 :</p> <pre>remote-users(config)#end remote-users#</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 5	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	<p>入力を確認します。</p>
ステップ 6	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	<p>(任意) コンフィギュレーション ファイルに設定を保存します。</p>

DNS の設定

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

DNS を使用するようにスイッチを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip domain name name 例： Device(config)# ip domain name Cisco.com	非完全修飾ホスト名（ドット付き 10 進表記ドメイン名のない名前）を完成させるためにソフトウェアが使用する、デフォルトのドメイン名を定義します。 ドメイン名を未修飾の名前から区切るために使用される最初のピリオドは入れないでください。 ブート時にはドメイン名は設定されていませんが、デバイスの設定が BOOTP または Dynamic Host Configuration Protocol (DHCP) サーバから行われている場合、BOOTP または DHCP サーバによってデフォルトのドメイン名が設定されることがあります（この情報がサーバに設定されている場合）。
ステップ 4	ip name-server server-address1 [server-address2 ... server-address6] 例：	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。

	コマンドまたはアクション	目的
	<pre>Device(config)# ip name-server 192.168.1.100 192.168.1.200 192.168.1.300</pre>	<p>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。</p>
ステップ 5	<p>ip domain lookup [nsap source-interface interface]</p> <p>例 :</p> <pre>Device(config)# ip domain-lookup</pre>	<p>(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。</p> <p>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。</p>
ステップ 6	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 7	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	<p>入力を確認します。</p>
ステップ 8	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	<p>(任意) コンフィギュレーションファイルに設定を保存します。</p>

Message-of-the-Day ログインバナーの設定

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

MOTD ログインバナーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	banner motd c message c 例： Device(config)# banner motd # This is a secure site. Only authorized users are allowed. For access, contact technical support. #	MoTD を指定します。 <i>c</i> : ポンド記号 (#) など、目的のデリミタを入力して Return キーを押します。区切り文字はバナー テキストの始まりと終わりを表します。終わりの区切り文字の後ろの文字は廃棄されます。 <i>message</i> : 255 文字までのバナーメッセージを入力します。メッセージ内には区切り文字を使用できません。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

ログインバナーの設定

接続されたすべての端末でログインバナーが表示されるように設定できます。バナーが表示されるのは、MoTD バナーの後で、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 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	banner login c message c 例： Device(config)# banner login \$ Access for authorized users only. Please enter your username and password. \$	ログインメッセージを指定します。 c：ポンド記号 (#) など、目的のデリミタを入力して Return キーを押します。区切り文字はバナーテキストの始まりと終わりを表します。終わりの区切り文字の後ろの文字は廃棄されます。 message：255 文字までの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 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例：	(任意) コンフィギュレーションファイルに設定を保存します。

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

MAC アドレス テーブルの管理

アドレス エージング タイムの変更

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

手順

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

	コマンドまたはアクション	目的
ステップ 5	show running-config 例 : Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

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 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	snmp-server host host-addr community-string notification-type { informs traps } {version {1 2c 3}} { vrf vrf instance name} 例 : 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。 <ul style="list-style-type: none"> host-addr : NMS の名前またはアドレスを指定します。 traps (デフォルト) : ホストに SNMP トラップを送信します。 informs : ホストに SNMP 情報を送信します。 version : サポートする SNMP バージョンを指定します。informs には

	コマンドまたはアクション	目的
		<p>バージョン 1 (デフォルト) を使用できません。</p> <ul style="list-style-type: none"> • <i>community-string</i> : 通知処理で送信する文字列を指定します。 snmp-server host コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、snmp-server community グローバルコンフィギュレーションコマンドを使用してから、snmp-server host コマンドを使用することを推奨します。 • <i>notification-type</i> : mac-notification キーワードを使用します。 • vrf vrf インスタンス名 : このホストの VPN ルーティング/転送インスタンスを指定します。
ステップ 4	<p>snmp-server enable traps mac-notification change</p> <p>例 :</p> <pre>Device(config)# snmp-server enable traps mac-notification change</pre>	<p>デバイスが MAC アドレス変更通知を NMS に送信できるようにします。</p>
ステップ 5	<p>mac address-table notification change</p> <p>例 :</p> <pre>Device(config)# mac address-table notification change</pre>	<p>MAC アドレス変更通知機能をイネーブルにします。</p>
ステップ 6	<p>mac address-table notification change [interval value] [history-size value]</p> <p>例 :</p> <pre>Device(config)# mac address-table notification change interval 123 Device(config)# mac address-table notification change history-size 100</pre>	<p>トラップインターバルタイムと履歴テーブルのサイズを入力します。</p> <ul style="list-style-type: none"> • (任意) interval value : NMS に生成されるトラップの各セット間の通知トラップインターバルを秒単位で指定します。指定できる範囲は 0 ~ 2147483647 秒です。デフォルトは 1 秒です。

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

MAC アドレス移動通知トラップの設定

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

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

手順

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

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>notification-type</i> : 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 例 : Device(config)# snmp-server enable traps mac-notification move	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。
ステップ 5	mac address-table notification mac-move 例 : 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 例 : Device(config)# end	特権 EXEC モードに戻ります。
ステップ 7	show running-config 例 : Device# show running-config	入力を確認します。
ステップ 8	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

次のタスク

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

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

MAC しきい値通知トラップの設定

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

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

手順

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

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>notification-type</i> : 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 例 : Device(config)# snmp-server enable traps mac-notification threshold	NMS への MAC しきい値通知トラップをイネーブルにします。
ステップ 5	mac address-table notification threshold 例 : 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] 例 : Device(config)# mac address-table notification threshold interval 123 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。 <ul style="list-style-type: none"> • (任意) limit percentage : MAC アドレステーブルの使用率を指定します。有効値は 1 ~ 100% ですデフォルト値は 50% です。 • (任意) interval time : 通知の間隔を指定します。有効値は 120 秒以上です。デフォルトは 120 秒です。
ステップ 7	end 例 : Device(config)# end	特権 EXEC モードに戻ります。
ステップ 8	show running-config 例 : Device# show running-config	入力を確認します。
ステップ 9	copy running-config startup-config 例 : Device# copy running-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。

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

VLAN の MAC アドレスラーニングのディセーブル化

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

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

始める前に

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

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

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : <code>Device# configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	no mac-address-table learning vlan [<i>vlan-id</i> <i>vlan-id</i> - <i>vlan-id</i> ,] 例 :	指定された 1 つまたは複数の VLAN で MAC アドレスラーニングをディセーブルにします。

	コマンドまたはアクション	目的
	<pre>Device(config)# no mac-address-table learning {vlan vlan-id [,vlan-id -vlan-id]}</pre>	1つのVLAN IDを指定、またはVLAN IDの範囲をハイフンまたはカンマで区切って指定できます。有効なVLAN IDの範囲は2～4094です。内部VLANは指定できません。
ステップ3	<p>end</p> <p>例：</p> <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。
ステップ4	<p>show mac-address-table learning vlan[vlan-id]</p> <p>例：</p> <pre>Device# show mac-address-table learning [vlan vlan-id]</pre>	<p>設定を確認します。</p> <p>show mac-address-table learning [vlan vlan-id] 特権 EXEC コマンドを入力すると、すべてのVLAN、または指定したVLANのMACアドレスラーニングのステータスを表示できます。</p>
ステップ5	<p>copy running-config startup-config</p> <p>例：</p> <pre>Device# copy running-config startup-config</pre>	(任意) コンフィギュレーションファイルに設定を保存します。
ステップ6	<p>default mac address-table learning</p> <p>例：</p> <pre>Device# default mac address-table</pre>	(任意) グローバル コンフィギュレーションモードでVLANのMACアドレスラーニングを再度イネーブルにします。

スタティック アドレス エントリの追加および削除

スタティック アドレスを追加するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ1	<p>enable</p> <p>例：</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ2	<p>configure terminal</p> <p>例：</p> <pre>Device# configure terminal</pre>	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	<p>mac address-table static mac-addr vlan vlan-id interface interface-id</p> <p>例 :</p> <pre>Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1</pre>	<p>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。</p> <ul style="list-style-type: none"> • <i>mac-addr</i> : 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。 • <i>vlan-id</i> : 指定された MAC アドレスを持つパケットを受信する VLAN を指定します。指定できる VLAN ID の範囲は 1 ~ 4094 です。 • <i>interface-id</i> : 受信パケットが転送されるインターフェイスを指定します。有効なインターフェイスは、物理ポートまたはポート チャネルです。スタティック マルチキャスト アドレスの場合、複数のインターフェイス ID を入力できます。スタティック ユニキャスト アドレスの場合、インターフェイスは同時に 1 つしか入力できません。ただし、同じ MAC アドレスおよび VLAN ID を指定すると、コマンドを複数回入力できます。
ステップ 4	<p>show running-config</p> <p>例 :</p> <pre>Device# show running-config</pre>	<p>入力を確認します。</p>
ステップ 5	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	<p>(任意) コンフィギュレーション ファイルに設定を保存します。</p>

ユニキャスト MAC アドレス フィルタリングの設定

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mac address-table static mac-addr vlan vlan-id drop 例： Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop	ユニキャスト MAC アドレス フィルタリングをイネーブルにし、デバイスが指定した送信元または宛先ユニキャスト スタティック アドレスを持つパケットをドロップするように設定します。 • <i>mac-addr</i> : 送信元または宛先ユニキャスト MAC アドレス (48 ビット) を指定します。この MAC アドレスを持つパケットはドロップされます。 • <i>vlan-id</i> : 指定された MAC アドレスを持つパケットを受信する VLAN を指定します。指定できる VLAN ID の範囲は 1 ~ 4094 です。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show running-config 例： Device# show running-config	入力を確認します。
ステップ 6	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

デバイスのモニタリングおよび保守の管理

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

コマンド	目的
<code>show mac address-table vlan <i>vlan-id</i></code>	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
```

例：ログインバナーの設定

```
Trying 192.0.2.15...
Connected to 192.0.2.15.
Escape character is '^]'.
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
User Access Verification
Password:
```

例：ログインバナーの設定

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

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

例：MAC アドレス変更通知トラップの設定

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

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

例：MAC しきい値通知トラップの設定

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

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification
Device(config)# snmp-server enable traps mac-notification threshold
```



```
Device(config)# mac address-table notification threshold
Device(config)# mac address-table notification threshold interval 123
Device(config)# mac address-table notification threshold limit 78
```

例：MAC アドレス テーブルへのスタティック アドレスの追加

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



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

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

例：ユニキャスト MAC アドレス フィルタリングの設定

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

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

デバイス管理に関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

デバイス管理の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	デバイス管理	デバイス管理では、システムの日時、システム名、ログインバナーを設定し、DNSを設定できます。

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



第 2 章

ブート整合性の可視性

- [ブート整合性の可視性について \(47 ページ\)](#)
- [ソフトウェアイメージとハードウェアの確認 \(49 ページ\)](#)
- [プラットフォーム ID とソフトウェア整合性の確認 \(49 ページ\)](#)
- [イメージ署名の検証 \(53 ページ\)](#)
- [ブート整合性の可視性に関する追加情報 \(54 ページ\)](#)
- [ブート整合性の可視性の機能履歴 \(54 ページ\)](#)

ブート整合性の可視性について

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

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

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

イメージ署名とブートアップ

シスコの構築したサーバーが Cisco IOS XE イメージを生成します。Cisco IOS XE イメージの場合、Abraxas イメージ署名システムを使用して、シスコの秘密 RSA キーでイメージに安全に署名できます。

Cisco IOS XE イメージを Catalyst 9000 シリーズスイッチにコピーすると、シスコの ROMMON ブート ROM がシスコのリリースキーを使用してイメージを検証します。これらのキーは、

Abraxas サーバーに安全に保存されているシスコのリリース秘密キーに対応する公開キーです。リリース秘密キーは ROMMON に保存されます。

Catalyst 9000 シリーズスイッチは、ブート整合性の可視性機能をサポートしています。ブート整合性の可視性は、ROMMON ソフトウェアが改ざんされていないことを確認するために、ROMMON ソフトウェアを検証するハードウェア トラスト アンカーとして機能します。

Cisco IOS XE イメージは、構築時にデジタル署名されます。バイナリイメージファイル全体に対して SHA-512 ハッシュが生成され、このハッシュがシスコの RSA 2048 ビット秘密キーで暗号化されます。ROMMON は、シスコの公開キーを使用して署名を検証します。このソフトウェアがシスコの構築したシステムによって生成されたものではない場合、署名の検証は失敗します。デバイスの ROMMON はイメージを拒否し、起動を停止します。署名の検証に成功すると、デバイスはイメージを Cisco IOS XE ランタイム環境で起動します。

ROMMON は、ブートアップ中に署名付き Cisco IOS XE イメージを検証する際、次の手順を実行します。

1. Cisco IOS XE イメージを CPU メモリにロードします。
2. Cisco IOS XE パッケージのヘッダーを調べます。
3. イメージに対して非セキュア整合性チェックを実行し、ディスクまたは TFTP で意図しないファイル破損が生じていないことを確認します。これは非セキュア SHA-1 ハッシュを使用して実行されます。
4. シスコの RSA 2048 ビット公開リリースキーを ROMMON ストレージからコピーし、シスコの RSA 2048 ビット公開リリースキーが改ざんされていないことを検証します。
5. パッケージのヘッダーからコード署名用署名 (SHA-512 ハッシュ) を抽出し、シスコの RSA 2048 ビット公開キーを使用して検証します。
6. Cisco IOS XE パッケージの SHA-512 ハッシュを計算してコード署名の検証を実行し、コード署名用署名と比較します。これで署名付きパッケージの検証が実行されたこととなります。
7. Cisco IOS XE パッケージのヘッダーを調べて、プラットフォームタイプと CPU アーキテクチャの互換性を検証します。
8. Cisco IOS XE パッケージから Cisco IOS XE ソフトウェアを抽出して起動します。



(注) 上記のプロセス中、手順3はイメージの非セキュアチェックであり、ディスクエラー、ファイル転送エラー、またはコピーエラーによる偶発的な破損に関してイメージを確認することを目的としています。これはイメージコード署名の一環ではありません。このチェックは、意図的なイメージの改ざんを検出するためのものではありません。

イメージコード署名の検証は、手順4、5、および6で行われます。これは、2048 ビット RSA キーで暗号化された SHA-512 ハッシュを使用した、イメージのセキュアコード署名チェックです。このチェックは、意図的なイメージの改ざんを検出することを目的としています。

ソフトウェアイメージとハードウェアの確認

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



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

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

手順

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

プラットフォーム ID とソフトウェア整合性の確認

プラットフォーム ID の確認

次に、PEM 形式でセキュアな固有デバイス識別子 (SUDI) チェーンを表示する例を示します。SUDI にエンコードされるのは、個々のデバイスの製品 ID とシリアル番号であり、何千ものデバイスからなるネットワーク上でデバイスを一意に識別できます。最初の証明書は Cisco Root CA 2048 で、2 番目はシスコの下位 CA (ACT2 SUDI CA) です。どちらの証明書も、

<https://www.cisco.com/security/pki/> で公開されているものと一致しているかを確認できます。3 番目は SUDI 証明書です。

```

Device# show platform sudi certificate sign nonce 123
-----BEGIN CERTIFICATE-----
MIIDQzCCAiuGAWIBAgIQX/h7KctU3I1CoxW1aMmt/zANBqkqhkiG9w0BAQUFADA1
MRYwFAYDVQKQEWlDaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENB
IDwNDGwHhcNMDQwNTE0MjAxNzEyWhcNMjkwNTE0MjAyNTQyWjA1MRYwFAYDVQKQ
EWlDaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDwNDGwgggEg
MA0GCSqGSIb3DQEBAQUAA4IBDQAwggEIAoIBAQCwrmrmp68Kd6ficba0ZmKUeIhH
xmJVhEAYv8CrLqUccda8bnuoqrpu0hWISEWdovyD0My5j0AmaHBKE8hF570YQXJ
FcjPFfto1YmUQ6iEqDGYeJu5Tm8sUxJsZr2tKyS7McQr/4NEb7Y9JhcJ6r8qqB9q
VvYgDxFU14FlpyXOWWqCZe+36ufijXWlLvLdt6ZeYpzPEApk0E5tzivMW/VgpSdH
jWn0f84bcN5wGyDwbs2maAg8EtKpP6BrXruOIIt6ke01a06g58QBdKhTCyTKmg91
Eg6CTY5j/e/rmxrbU6YTYK/CfdHbBcl1HP7R2RQgYcUTOG/rksc35LtlGxfAgEd
o1EwTzALBgNVHQ8EBAMCAYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUUJ/PI
FR5umgIJFq0roIlqX9p7L6owEAYJKwYBAGCNxUBBAMCAQAwDQYJKoZIhvcNAQEF
BQADggEBAJ2dhISjQa18dwy3U8pORFbi71R803UXHOjgkxhLtv5M0hmBvRbW7hmW
Yqpao2TB9k5UM8Z3/sUcuuVdJcr18J0agxEu5sv4dEX+5wW4q+ffY0vhN4TauYuX
cB7w4ovXsNgOnbFpliqRe61JT37mjpXYgyC81WhJdtSd9i7rp77rMKSsH0T8lasz
Bvt9YArEtIjpsJyp8qS5UwGH0GikJ3+r/+n6yUA4iGe00caEb1fJU9u6ju7AQ7L4
CYNu/2bPPu8Xs1gYJqk0XuPL1hs27PKSb3TkL4Eq1ZKR4OCXPDJoBYVl0fdX41Id
kxpUnwVwwEpxYB5DC2Ae/qPOgRnhCzU=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIEPDCCAySgAwIBAgIKYQLufQAAAAAADANBgkqhkiG9w0BAQUFADA1MRYwFAYD
VQKQEWlDaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDwNDGw
HhcNMTwNjMwMjE1NjU3WhcNMjkwNTE0MjAyNTQyWjAnMQ4wDAYDVQKQEWVDAxNj
bzEVMBMGA1UEAxMMQUNUMiBTVURJIEENBMBIIBjANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIBCgKCAQEAm5l3THiXa9tN/hS5qR/6UZRpdd+9aE2JbFknjht6gfHkD477AkS
5XAtUs5oxDYVt/zEbs1Zq3+LR6qrxqKKQVU6JYvH05UYLbqCj38s76NLk5390WzP
9pRcmRCPUx+a6tHF/qRuOiJ44mdeDYzo3qPCpxzprWJDPclM4iYKHumMQmgmgm+
xghHIooWS80BOcdiynEbeP5rZ7qRuewKmpl1TiI3WdBNjZjnpfjg66F+P4SaDkGb
BXGj130VeF+EyFWLrFj97fL2+8oauV43Qrvnf3d/GfqXj7ew+z/sX1XtEOjSxJ
URsYMEj53Rdd9tJwHky8neapsz+r+kdVQIDAQABo4IBWjCAVYwCwYDVR0PBAQD
AgHGMBOGA1UdDgQWBBI2PHxwnDVW7t8cwmTr7i4MAP4fzAfBgNVHSMEGDAWgBQN
88gVhm6aAgkWrSugiWbF2nsVqjBDBgNVHR8EPDA6MDIqNgA0hjJodHRwOi8vd3d3
LmNpc2NvLmNvbS9zZW50cm10eS9wa2kvY3JsL2NyY2EyMDQ4LmNybDBQBggrBgEF
BQcBAQREMEIwQAYIKwYBBQUHMAKGNGh0dHA6Ly93d3cuY2l2Y28uY29tL3N1Y3Vy
aXR5L3BraS9wY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29y
aXR5L3BraS9wY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29y
L3BraS9wY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29yY29y
KoZIhvcNAQEFBQADggEBAgh1qclr9tx4hzWgDERm371yeuEmqCifi9b9+GbmSjbi
ZHc/CcCl0lJu0a9zTXA9w47H9/t61eduGxb4WeLxcwCiUgvFtCa51Ikl8NbcKY
/4dwllex+7amATUQ04QggIE67wVIPu6bgAE3Ja/nRS3xKYSnj8H5TehimBSv6TECi
i5jUhOwryAK4dVo8hCjkjEkzu3ufBTJapnv89g90E+H3VKM4L+/KdkUO+52djFKn
hy147d7cZR4DY4LIuFM2P1As8YyjoNpK/urSRI14WdIlplR1nH7KND15618yFVP
0IFJZBGrooCRBjOSwFv8cpWCbmWdPaCQT2nwIjTfY8c=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIDezCCAmOgAwIBAgIEAc+JiTANBgkqhkiG9w0BAQsFADANMQ4wDAYDVQKQEWVD
aXNjbyEVMBMGA1UEAxMMQUNUMiBTVURJIEENBMB4XDTE3MDgxeTEwNDMzOVoXDTI3
MDgxeTEwNDMzOVoZzEmMCQGA1UEBRMdeUe1EOKM5MzAwLTI0VGVgU046RknXmJez
NEwWEMMxDjAMBGNVBAoTBUnc2NvMRgWfGyDVQQLEw9BQ1Q1MjM1MjM1MjM1MjM1MjM1
EzARBGNVBAMTCkM5MzAwLTI0VGVgWggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQDav5txv4THsqXWCC7AzzHm5M228Feqk8FA3tXAv0tV8RXTy4Z9I9XgRzw
Yw8chknh8LuDMcmGmk8DP+ct++vAF4nkVeIeBeOHnx2RuC9rcR8tuKjCimamDk0M
Jhk12w/9+TbdKdNBEy6Sueh1RPVbuSk1oQLQcOYW7CsYC5tI1GkJKfk1nGEK3ni3
ztIpsi7QhYp6k59yccnbzXSdwoBrbtbpIIYEK/iHWFQRdlMUunnfIshI7yPneo7V0
NnPC08wk+CA+8XeXk/fnDeGAswKRK1tW9jDP/sY1YubBJNJ4ToqQpG6W/hbNvu3Y
NyS24osSvnn5Bp7on3Rf7eHq9hNjAgMBAAGjbzBtMA4GA1UdDwEB/wQEAwIF4DAM
BgNVHRMBAf8EAjAAME0GA1UdEQRGMEsgQgYJKwYBBAEJFQIDoDUTM0NoaXBJRD1V
WUpPVkVZNEZRT0xSbkpwSUUxaGNpQXhNQ0F4Tnpvd05Eb3hNeUFiY1FjPTANBgkq
hkiG9w0BAQsFAAOCAQEASXX+iZLMvHQIR1/s1Pobm0kP/bYeHsgDTRQPRHbCMLHH

```



```

B0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03E
cat9k-rpbase.16.10.01.SPA.pkg :
4057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C6
cat9k-rpboot.16.10.01.SPA.pkg :
AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057
cat9k-sipbase.16.10.01.SPA.pkg :
9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A
cat9k-sipsa.16.10.01.SPA.pkg :
E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673
cat9k-srdriver.16.10.01.SPA.pkg :
4FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E211
cat9k-webui.16.10.01.SPA.pkg :
CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7
cat9k-wlc.16.10.01.SPA.pkg :
AA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAA
PCR0: 9745B571B66D79F0936F4D292B5672B50F50FD1E56E74248D48A33582E992574
PCR8: 1CC295C233DA41BD3530A6F09C21991E8406BFFC88249D7778CA4BB0B9E71EB7
Signature version: 1
Signature:

```

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

```

Device# show platform integrity sign nonce 123
Platform: C9300-24P
Boot 0 Version: MA0081R06.1307262016
Boot 0 Hash: A99EF9F31CE3F3F8533055407F1C88C62176E667E4E1DA0649EAA7A1282F205E0A
Boot Loader Version: System Bootstrap, Version 16.8.0.3, RELEASE SOFTWARE (P)
Boot Loader Hash:
F82826514668055C3993AE95F53512341BF20F3CC7D4083C980450EFA6CD84608EE638B515D13414203CED35603F01974EB8676C6A06F9DC45E25CD1039E686C40A
OS Version: 16.10.01
OS Hashes:
cat9k_iosxe.16.10.01.SPA.bin :
F4CAD08FE1EF841C3A2E3ED8540829F08F3CEA9336F38E45669D4D8B15AD15E365B922AC8B4DC0D5B63E2806D6A1BDAB7839DC9DC087E366A49ED648C113440
cat9k-cc_srdriver.16.10.01.SPA.pkg :
D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0
cat9k-espsbase.16.10.01.SPA.pkg :
3EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEF43
cat9k-guestshell.16.10.01.SPA.pkg :
B0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03E
cat9k-rpbase.16.10.01.SPA.pkg :
4057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C6
cat9k-rpboot.16.10.01.SPA.pkg :
AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057
cat9k-sipbase.16.10.01.SPA.pkg :
9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A
cat9k-sipsa.16.10.01.SPA.pkg :
E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673
cat9k-srdriver.16.10.01.SPA.pkg :
4FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E211
cat9k-webui.16.10.01.SPA.pkg :
CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7
cat9k-wlc.16.10.01.SPA.pkg :
AA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAAA7ED0AE935CB0BD84E0D0D155C1DEFB03EB0C64057AD6A9673E2114FA7CCAA
PCR0: 9745B571B66D79F0936F4D292B5672B50F50FD1E56E74248D48A33582E992574
PCR8: 1CC295C233DA41BD3530A6F09C21991E8406BFFC88249D7778CA4BB0B9E71EB7
Signature version: 1
Signature:

```


イメージ署名の検証

次に、SHA-512ハッシュを使用した、ブートアップ中のイメージに対するセキュアコード署名チェックの例を示します。

```
switch:boot flash:packages.conf
boot: attempting to boot from [flash:packages.conf]
boot: reading file packages.conf
#
Performing Integrity Check ...
boot: parsed image from conf file: cat9k-rpboot.17.02.01.SSA.pkg
```

```
Loading image in Verbose mode: 1
```

```
Image Base is: 0x100099000
Image Size is: 0x2C83487
Package header rev 3 structure detected
Package type:30001, flags:0x0
IsoSize = 0
Parsing package TLV info:
000: 0000000900000001D4B45595F544C565F - KEY_TLV_
010: 5041434B4147455F434F4D5041544942 - PACKAGE_COMPATIB
020: 494C495459000000000000090000000B - ILITY
030: 4652555F52505F54595045000000009 - FRU_RP_TYPE
040: 000000184B45595F544C565F5041434B - KEY_TLV_PACK
050: 4147455F424F4F54415243480000009 - AGE_BOOTARCH
060: 0000000E415243485F693638365F5459 - ARCH_i686_TY
070: 50450000000000009000000144B45595F - PE KEY_
080: 544C565F424F4152445F434F4D504154 - TLV_BOARD_COMPAT
090: 00000009000000010424F4152445F6361 - BOARD_ca
0A0: 74396B5F545950450000000900000018 - t9k_TYPE
0B0: 4B45595F544C565F43525950544F5F4B - KEY_TLV_CRYPTO_K
0C0: 4559535452494E470000000900000004 - EYSTRING

TLV: T=9, L=29, V=KEY_TLV_PACKAGE_COMPATIBILITY
TLV: T=9, L=11, V=FRU_RP_TYPE
TLV: T=9, L=24, V=KEY_TLV_PACKAGE_BOOTARCH
TLV: T=9, L=14, V=ARCH_i686_TYPE
TLV: T=9, L=20, V=KEY_TLV_BOARD_COMPAT
TLV: T=9, L=16, V=BOARD_cat9k_TYPE
TLV: T=9, L=24, V=KEY_TLV_CRYPTO_KEYSTRING
TLV: T=9, L=4, V=none
TLV: T=9, L=11, V=CW_BEGIN=$$
TLV: T=9, L=17, V=CW_FAMILY=$cat9k$
TLV: T=9, L=74, V=CW_IMAGE=$cat9k-rpboot.17.02.01.SSA.pkg$
TLV: T=9, L=20, V=CW_VERSION=$17.2.01$
IOS version is 17.2.1
TLV: T=9, L=53, V=CW_FULL_VERSION=$17.2.01.0.869.1580816579..Amsterdam$
TLV: T=9, L=52, V=CW_DESCRIPTION=$Cisco IOS Software, IOS-XE Software$
TLV: T=9, L=9, V=CW_END=$$
Found DIGISIGN TLV type 12 length = 392
RSA Self Test Passed

Expected hash:
DDAF35A193617ABACC417349AE204131
12E6FA4E89A97EA20A9EEEE64B55D39A
```

```
2192992A274FC1A836BA3C23A3FEEBBD
454D4423643CE80E2A9AC94FA54CA49F
```

```
Obtained hash:
DDAF35A193617ABACC417349AE204131
12E6FA4E89A97EA20A9EEEE64B55D39A
2192992A274FC1A836BA3C23A3FEEBBD
454D4423643CE80E2A9AC94FA54CA49F
Sha512 Self Test Passed
Found package arch type ARCH_i686_TYPE
Found package FRU type FRU_RP_TYPE
Performing Integrity Check ...
```

```
RSA Signed DEVELOPMENT Image Signature Verification Successful.
```

ブート整合性の可視性に関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

ブート整合性の可視性の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	ブート整合性の可視性	ブート整合性の可視性によって、シスコのプラットフォームIDとソフトウェアの整合性情報が可視化され、実用可能になります。プラットフォームIDは、プラットフォームの製造元でインストールされたIDを提供します。

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



第 3 章

デバイスのセットアップ設定の実行

- [デバイスセットアップの設定の制約事項 \(55 ページ\)](#)
- [デバイスセットアップ設定の実行に関する情報 \(55 ページ\)](#)
- [デバイスセットアップ設定の実行方法 \(71 ページ\)](#)
- [デバイスのセットアップの設定例 \(86 ページ\)](#)
- [デバイスセットアップの実行に関する追加情報 \(106 ページ\)](#)
- [デバイスセットアップ設定の実行に関する機能履歴 \(106 ページ\)](#)

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。

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 9300 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。
 - チップのアクセス可能性、ファームウェアのダウンロード、給電機器の正常性ステータスを確認する Power over Ethernet (PoE) コントローラの機能テスト。
 - デバイスセンサーからの温度の読み取りを確認する温度テスト。
 - 挿入されたすべてのファンモジュールがボード上で正常に動作しているかどうかを確認するファンモジュールテスト。
 - 連邦情報処理標準 (FIPS) MACsec テスト。

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

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

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



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

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

ソフトウェア インストールの概要

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

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

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

スイッチでは、次のソフトウェア インストール機能がサポートされています。

- スタンドアロン スイッチでのソフトウェア バンドルのインストール。
- 以前にインストールしたパッケージセットへのソフトウェア ロールバック。
- 有効なインストール済みパッケージがブート フラッシュに存在しない場合の緊急インストール。

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) Cisco Catalyst 9200 シリーズ スイッチにはインストールモードを使用することを推奨します。



(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.16.05.01a.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。

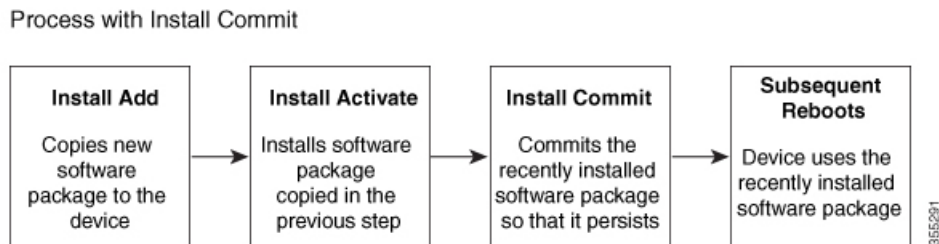
install activate コマンドを動作させるには、パッケージをデバイスのブートフラッシュで使用可能にする必要があります。このコマンドを設定すると、.bin ファイルから以前に追加したパッケージがアクティブ化され、システムがリロードします。

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、ホスト名とイーサネット シークレット パスワードを設定することもできます。

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

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



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

デバイスの設定手順を熟知している経験豊富なユーザの場合は、デバイスを手動で設定してください。それ以外のユーザーは、[デバイスブートプロセス \(55 ページ\)](#) 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 4: 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。

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せん。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。

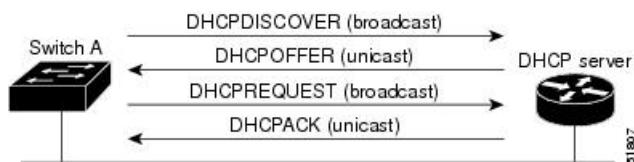
DHCP ベースの自動設定は、デバイスの BOOTP クライアント機能に代わるものです。

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

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

DHCP ベースの自動設定およびイメージアップデート

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

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

DHCP ベースの自動設定の制約事項

- ネットワーク内に割り当てられた IP アドレスがなく、1つ以上のレイヤ3 インターフェイスが起動していない場合は、設定プロセスが保存された DHCP ベースの自動設定は停止します。
- タイムアウトを設定しない限り、設定機能を備えている DHCP ベースの自動設定は IP アドレスのダウンロードを無期限に繰り返します。
- コンフィギュレーションファイルをダウンロードできないか破損している場合は、自動インストールプロセスが停止します。
- TFTP からダウンロードされたコンフィギュレーション ファイルは、実行コンフィギュレーション内の既存コンフィギュレーションとマージされますが、**write memory** または

copy running-configuration startup-configuration 特権 EXEC コマンドを入力しない限り、NVRAMに保存されません。ダウンロードされたコンフィギュレーションがスタートアップコンフィギュレーションに保存された場合、後続のシステム再起動中にこの機能はトリガーされません。

DHCP 自動設定

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

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

TFTP サーバアドレスには、（存在する場合）指定された TFTP サーバのアドレス、およびブロードキャストアドレス（255.255.255.255）が含まれています。

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

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

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

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

DNS サーバの目的

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

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

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

コンフィギュレーションファイルの入手方法

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

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

デバイスは DHCP サーバから、IP アドレス、サブネットマスク、TFTP サーバアドレス、およびコンフィギュレーションファイル名を受信します。デバイスは、TFTP サーバにユ

ユニキャストメッセージを送信し、指定されたコンフィギュレーションファイルをサーバのベースディレクトリから取得して、ブートアッププロセスを完了します。

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

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

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

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

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

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

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



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

環境変数の制御方法

通常動作デバイスでは、9600 bps に設定されているコンソール接続のみを通じてブートローダモードを開始します。電源コードを再接続中にデバイス電源コードを取り外し、[Mode] ボタンを押します。システム LED がグリーンの点滅から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、ブートローダまたはシステムで稼働する他のソフトウェアの動作を制御できます。ブートローダの環境変数は、UNIX または DOS システムで設定できる環境変数と類似しています。

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

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

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

一般的な環境変数

この表では、最も一般的な環境変数の機能について説明します。

表 5: 一般的な環境変数

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

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

TFTP の環境変数

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

表 6: TFTP の環境変数

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

ソフトウェア イメージのリロードのスケジューリング

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



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

リロード オプションには以下のものがあります。

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

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

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

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

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

デバイスセットアップ設定の実行方法

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

DHCP 自動設定（コンフィギュレーション ファイルだけ）の設定

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

手順

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

	コマンドまたはアクション	目的
ステップ 3	boot filename 例 : 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 例 : 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。 (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 例 : 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 例 : 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 例 : Device(dhcp-config)# exit	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	tftp-server flash:filename.text 例 : 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 例： Device (config) # interface gigabitethernet1/0/4	コンフィギュレーションファイルを受信するクライアントのアドレスを指定します。
ステップ 10	no switchport 例： Device (config-if) # no switchport	インターフェイスをレイヤ 3 モードにします。
ステップ 11	ip address address mask 例： Device (config-if) # ip address 10.10.10.1 255.255.255.0	IP アドレスとインターフェイスのマスクを指定します。
ステップ 12	end 例： Device (config-if) # end	特権 EXEC モードに戻ります。

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

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

始める前に

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

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例：	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 例： 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 例： 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 例： 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ワーク番号およびマスクを指定します。 (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 例： 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 例： 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 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 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370	
ステップ 8	copy tftp flash filename.txt 例 : Device(config)# copy tftp flash image.bin	デバイスに、テキストファイルをアップロードします。
ステップ 9	copy tftp flash imagename.bin 例 : Device(config)# copy tftp flash image.bin	デバイスに、新しいイメージの tar ファイルをアップロードします。
ステップ 10	exit 例 : Device(dhcp-config)# exit	グローバル コンフィギュレーション モードに戻ります。
ステップ 11	tftp-server flash: config.txt 例 : Device(config)# tftp-server flash:config-boot.txt	TFTP サーバ上の Cisco IOS コンフィギュレーション ファイルを指定します。
ステップ 12	tftp-server flash: imagename.bin 例 : Device(config)# tftp-server flash:image.bin	TFTP サーバ上のイメージ名を指定します。
ステップ 13	tftp-server flash: filename.txt 例 : Device(config)# tftp-server flash:boot-config.txt	ダウンロードするイメージファイルの名前を記述したテキストファイルを指定します。

	コマンドまたはアクション	目的
ステップ 14	interface interface-id 例： Device(config)# interface gigabitEthernet1/0/4	コンフィギュレーションファイルを受信するクライアントのアドレスを指定します。
ステップ 15	no switchport 例： Device(config-if)# no switchport	インターフェイスをレイヤ 3 モードにします。
ステップ 16	ip address address mask 例： Device(config-if)# ip address 10.10.10.1 255.255.255.0	IP アドレスとインターフェイスのマスクを指定します。
ステップ 17	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。
ステップ 18	copy running-config startup-config 例： Device(config-if)# end	(任意) コンフィギュレーションファイルに設定を保存します。

DHCP サーバからファイルをダウンロードするクライアントの設定



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

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例：	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# <code>configure terminal</code>	
ステップ 2	boot host dhcp 例： Device(conf)# <code>boot host dhcp</code>	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 <i>timeout-value</i> 例： Device(conf)# <code>boot host retry timeout 300</code>	(任意) システムがコンフィギュレーションファイルダウンロードしようとする時間を設定します。 (注) タイムアウトを設定しないと、システムは無期限に DHCP サーバから IP アドレスを取得しようとします。
ステップ 4	banner config-save ^C <i>warning-message</i> ^C 例： Device(conf)# <code>banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C</code>	(任意) コンフィギュレーションファイルを NVRAM に保存しようとするときに表示される警告メッセージを作成します。
ステップ 5	end 例： Device(config-if)# <code>end</code>	特権 EXEC モードに戻ります。
ステップ 6	show boot 例： Device# <code>show boot</code>	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 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface vlan <i>vlan-id</i> 例 : Device(config)# interface vlan 99	インターフェイス コンフィギュレーション モードを開始して、IP 情報が割り当てられている VLAN を指定します。指定できる範囲は 1 ~ 4094 です。
ステップ 4	ip address <i>ip-address subnet-mask</i> 例 : Device(config-vlan)# ip address 10.10.10.2 255.255.255.0	IP アドレスとサブネット マスクを入力します。
ステップ 5	exit 例 : Device(config-vlan)# exit	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	ip default-gateway <i>ip-address</i> 例 : 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。 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
		<p>(注) IP でルーティングするようにデバイスを設定した場合、デフォルトゲートウェイの設定は不要です。</p> <p>(注) デフォルトゲートウェイの構成に基づいて、デバイスのCAPWAPは中継を行い、ルーティングされたアクセスポイントとデバイスの接続をサポートします。</p>
ステップ 7	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。
ステップ 8	<p>show interfaces vlan <i>vlan-id</i></p> <p>例 :</p> <pre>Device# show interfaces vlan 99</pre>	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	<p>show ip redirects</p> <p>例 :</p> <pre>Device# show ip redirects</pre>	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	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>boot flash:/file-url</p> <p>例 :</p> <pre>Device(config)# boot flash:config.text</pre>	<p>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。</p> <ul style="list-style-type: none"> 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。 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	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 5	<p>show boot</p> <p>例 :</p> <pre>Device# show boot</pre>	<p>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。</p> <ul style="list-style-type: none"> 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	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	<p>（任意）コンフィギュレーション ファイルに設定を保存します。</p>

スイッチの手動による起動

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

始める前に

このタスクのスタンドアロン スイッチを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	boot manual 例 : Device (config)# boot manual	次回の起動時に、スイッチを手動で起動できるようにします。
ステップ 3	end 例 : Device (config)# end	特権 EXEC モードに戻ります。
ステップ 4	show boot 例 : Device# show boot	入力を確認します。 boot manual グローバルコマンドは、 MANUAL_BOOT 環境変数の設定を変更します。 次回、システムを再起動した際には、スイッチはブートローダ モードになり、ブートローダ モードであることが 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。 <ul style="list-style-type: none"> • filesystem : システムボードのフラッシュ デバイスに flash: を使用します。 Switch: boot flash:

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>file-url</i> : 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。 <p>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。</p>
ステップ 5	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

インストールモードでのデバイスのブート

ソフトウェアパッケージのインストール

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	install add file tftp: filename [activate commit] 例 : 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。 <ul style="list-style-type: none"> • このコマンドは、.bin ファイルの個別のコンポーネントをサブパッケージと packages.conf ファイルに抽出します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> このコマンドの実行後にデバイスはリロードします。
ステップ 3	exit 例： Device# exit	特権 EXEC モードを終了し、ユーザー 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 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	install add file tftp: filename 例： Device# install add file tftp://172.16.0.1/tftpbboot/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。 <ul style="list-style-type: none"> このコマンドは、.bin ファイルの個別のコンポーネントをサブパッケージと packages.conf ファイルに抽出します。
ステップ 3	install activate [auto-abort-timer] 例： Device# install activate	追加のソフトウェア インストール パッケージをアクティブ化し、デバイスをリロードします。 <ul style="list-style-type: none"> ソフトウェアの完全インストールを実行する場合は、パッケージ ファイル名を指定しないでください。 auto-abort-timer キーワードがソフトウェア イメージのアクティブ化を自動的にロールバックします。 新しいイメージがアクティブになった後で自動タイマーがトリガーされます。 install commit コマンドを発行する前にタイマーの期限が切れた

	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 例： Device# install abort	(任意) ソフトウェアインストールのアクティブ化を終了し、現在のインストール手順の前に実行していたバージョンにロールバックします。 • このコマンドは、イメージがアクティブ化されている状態でのみ使用できます。イメージがコミットされた状態の場合は使用できません。
ステップ 5	install commit 例： Device# install commit	リロードが繰り返されても持続する変更を行います。 • 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。自動アポルト 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 例： Device# install rollback to committed	(任意) 最後にコミットしたバージョンに更新をロールバックします。
ステップ 7	install remove {file filesystem: filename inactive} 例： Device# install remove inactive	(任意) 未使用および非アクティブ状態のソフトウェア インストール ファイルを削除します。
ステップ 8	show install summary 例： Device# show install summary	アクティブ パッケージに関する情報を表示します。 • このコマンドの出力は、設定されている install コマンドに応じて変化します。

バンドルモードでのデバイスの起動

デバイスを起動するには、いくつかの方法があります。1つは、TFTP サーバーから 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 サーバーからデバイスを起動する方法を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch:BOOT=<source path of .bin file> 例： switch: switch:BOOT=tftp://10.0.0.2/cat9k_iosxe.16.05.01a.SPA.bin switch: switch:	ブートパラメータを設定します。
ステップ 2	boot 例： switch:boot	デバイスを起動します。
ステップ 3	show version	(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 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	copy running-config startup-config 例： Device# copy running-config startup-config	reload コマンドを使用する前に、デバイスの設定情報をスタートアップコンフィギュレーションに保存します。

	コマンドまたはアクション	目的
ステップ 4	reload in [hh:]mm [text] 例 : <pre>Device# reload in 12 System configuration has been modified. Save? [yes/no]: y</pre>	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] 例 : <pre>Device(config)# reload at 14:00</pre>	リロードを実行する時間を、時間数と分数で指定します。 (注) at キーワードを使用するのは、デバイスのシステムクロックが (Network Time Protocol (NTP)、ハードウェアカレンダー、または手動で) 設定されている場合だけです。時刻は、デバイスに設定されたタイムゾーンに基づきます。リロードが複数のデバイスで同時に行われるようにスケジュールするには、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 例 : <pre>Device(config)# reload cancel</pre>	以前にスケジュールされたリロードをキャンセルします。
ステップ 7	show reload 例 : <pre>show reload</pre>	以前デバイスにスケジュール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ル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 flash:packages.conf
Attempting to boot from [flash:packages.conf]
Located packages.conf
#

validate_package: SHA-1 hash:
    expected 340D5091:2872A0DD:03E9068C:3FDBECAB:69786462
    calculated 340D5091:2872A0DD:03E9068C:3FDBECAB:69786462
Image parsed from conf file is cat9k-rpboot.16.05.01a.SPA.pkg
#####

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

Restricted Rights Legend

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

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

Cisco IOS Software [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.5.1a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Tue 30-May-17 00:36 by mcpre

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

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

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
```

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

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 C9300-48P (X86) processor with 818597K/6147K bytes of memory.
 Processor board ID FCW2049G03S
 2048K bytes of non-volatile configuration memory.
 8388608K bytes of physical memory.
 1638400K bytes of Crash Files at crashinfo:.
 11264000K bytes of Flash at flash:.
 0K bytes of WebUI ODM Files at webui:.

```
Base Ethernet MAC Address      : 04:6c:9d:01:3b:80
Motherboard Assembly Number   : 73-17956-04
Motherboard Serial Number     : FOC20465ABU
Model Revision Number         : P4B
Motherboard Revision Number   : 04
Model Number                   : C9300-48P
System Serial Number          : FCW2049G03S
```

%INIT: waited 0 seconds for NVRAM to be available

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

Press RETURN to get started!

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 flash:cat9k_iosxe.16.05.01a.SPA.bin
```

```
Attempting to boot from [flash:cat9k_iosxe.16.05.01a.SPA.bin]
```

```
Located cat9k_iosxe.16.05.01a.SPA.bin
```

```
#####
Warning: ignoring ROMMON var "BOOT_PARAM"
```

```
Waiting for 120 seconds for other switches to boot
```

```
#####
Switch number is 3
```

Restricted Rights Legend

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

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

```
Cisco IOS Software [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.5.1a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Tue 30-May-17 00:36 by mcpre
```

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

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

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

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

```
If you require further assistance please contact us by sending email to
export@cisco.com.
```

```
cisco C9300-24U (X86) processor with 818597K/6147K bytes of memory.
Processor board ID FCW2111G00X
2048K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1638400K bytes of Crash Files at crashinfo:.
11264000K bytes of Flash at flash:.
15633392K bytes of USB Flash at usbflash0:.
0K bytes of WebUI ODM Files at webui:.
```

```
Base Ethernet MAC Address      : 04:6c:9d:1e:2a:80
Motherboard Assembly Number    : 73-17954-05
Motherboard Serial Number      : FOC21094MWL
Model Revision Number          : PP
Motherboard Revision Number    : 05
Model Number                   : C9300-24U
System Serial Number           : FCW2111G00X
```

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

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

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

例：緊急インストール

次に、**emergency-install** ブートコマンドの出力例を示します。

```
switch: emergency-install tftp://10.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin
WARNING: The system partition (bootflash:) will be erased during the system recovery
install process.
Are you sure you want to proceed? [y] y/n [n]: y
Starting system recovery
(tftp://210.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin) ...
Attempting to boot from [sda9:cat9k-recovery.SSA.bin]
Located cat9k-recovery.SSA.bin
#####

Warning: ignoring ROMMON var "BOOT_PARAM"

PLATFORM_TYPE C9300 speed 9600

Booting Recovery Image 16.5.1a

Initiating Emergency Installation of bundle
tftp://10.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin

Downloading bundle tftp://10.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
curl_vrf=2
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
 100 485M  100 485M    0     0  5143k      0  0:01:36  0:01:36 ---:---:-- 5256k
 100 485M  100 485M    0     0  5143k      0  0:01:36  0:01:36 ---:---:-- 5143k

Validating bundle tftp://10.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Installing bundle tftp://10.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Verifying bundle tftp://10.10.0.10/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Package cat9k-cc_srdriver.16.05.01a.SPA.pkg
/temp//stage/cat9k-cc_srdriver.16.05.01a.SPA.pkg is Digitally Signed
Package cat9k-espbase.16.05.01a.SPA.pkg /temp//stage/cat9k-espbase.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-guestshell.16.05.01a.SPA.pkg /temp//stage/cat9k-guestshell.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-rpbase.16.05.01a.SPA.pkg /temp//stage/cat9k-rpbase.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-sipbase.16.05.01a.SPA.pkg /temp//stage/cat9k-sipbase.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-sipspace.16.05.01a.SPA.pkg /temp//stage/cat9k-sipspace.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-srdriver.16.05.01a.SPA.pkg /temp//stage/cat9k-srdriver.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-webui.16.05.01a.SPA.pkg /temp//stage/cat9k-webui.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-wlc.16.05.01a.SPA.pkg /temp//stage/cat9k-wlc.16.05.01a.SPA.pkg is Digitally
Signed
Package /cat9k-rpboot.16.05.01a.SPA.pkg /temp//rpboot/cat9k-rpboot.16.05.01a.SPA.pkg is
Digitally Signed
Preparing flash...
Flash filesystem unmounted successfully /dev/sdb3
Syncing device...
```

```
Emergency Install successful... Rebooting
Will reboot now

Initializing Hardware...

System Bootstrap, Version 16.5.2r, RELEASE SOFTWARE (P)
Compiled Wed 05/31/2017 15:58:35.22 by rel

Current image running:
Primary Rommon Image

Last reset cause: SoftwareReload
C9300-24U platform with 8388608 Kbytes of main memory
```

例：更新プログラムパッケージの管理

次に、ソフトウェア パッケージ ファイルを追加する例を示します。

```
Device# install add file flash:cat9k_iosxe.16.06.02.SPA.bin activate commit

install_add_activate_commit: START Mon Oct 30 19:54:51 UTC 2017

System configuration has been modified.
Press Yes(y) to save the configuration and proceed.
Press No(n) for proceeding without saving the configuration.
Press Quit(q) to exit, you may save configuration and re-enter the command. [y/n/q]y
Building configuration...

[OK]Modified configuration has been saved

*Oct 30 19:54:55.633: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 19:54:55 install_engine.sh:

%INSTALL-5-INSTALL_START_INFO: Started install one-shot
flash:cat9k_iosxe.16.06.02.SPA.bininstall_add_activate_commit: Adding PACKAGE

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

--- Starting initial file syncing ---
Info: Finished copying flash:cat9k_iosxe.16.06.02.SPA.bin to the selected switch(es)
Finished initial file syncing

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

install_add_activate_commit: Activating PACKAGE
Following packages shall be activated:
/flash/cat9k-wlc.16.06.02.SPA.pkg
/flash/cat9k-webui.16.06.02.SPA.pkg
/flash/cat9k-srdriver.16.06.02.SPA.pkg
/flash/cat9k-sipspace.16.06.02.SPA.pkg
/flash/cat9k-sipbase.16.06.02.SPA.pkg
/flash/cat9k-rpboot.16.06.02.SPA.pkg
/flash/cat9k-rpbase.16.06.02.SPA.pkg
/flash/cat9k-guestshell.16.06.02.SPA.pkg
/flash/cat9k-espace.16.06.02.SPA.pkg
/flash/cat9k-cc_srdriver.16.06.02.SPA.pkg
```

```

This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on all members
  [1] Activate package(s) on switch 1
  [1] Finished Activate on switch 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

--- Starting Commit ---
Performing Commit on all members

*Oct 30 19:57:41.145: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 19:57:41 rollback_timer.sh:

%INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Install auto abort timer will expire in
7200 seconds [1]
Commit package(s) on switch 1
  [1] Finished Commit on switch 1
Checking status of Commit on [1]
Commit: Passed on [1]
Finished Commit

Install will reload the system now!
SUCCESS: install_add_activate_commit Mon Oct 30 19:57:48 UTC 2017

Device#
*Oct 30 19:57:48.384: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 19:57:48 install_engine.sh:

%INSTALL-5-INSTALL_COMPLETED_INFO: Completed install one-shot PACKAGE
flash:cat9k_iosxe.16.06.02.SPA.bin

Chassis 1 reloading, reason - Reload command

```

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 Mon Oct 30 20:14:20 UTC 2017
install_activate: Activating PACKAGE

*Oct 30 20:14:21.379: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:14:21 install_engine.sh:

%INSTALL-5-INSTALL_START_INFO: Started install activateFollowing packages shall be
activated:
/flash/cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg

```



```

/flash/cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-sibase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-espbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg

This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on all members
  [1] Activate package(s) on switch 1
      --- Starting list of software package changes ---
      Old files list:
      Removed cat9k-cc_srdriver.16.06.02.SPA.pkg
      Removed cat9k-espbase.16.06.02.SPA.pkg
      Removed cat9k-guestshell.16.06.02.SPA.pkg
      Removed cat9k-rpbase.16.06.02.SPA.pkg
      Removed cat9k-rpboot.16.06.02.SPA.pkg
      Removed cat9k-sibase.16.06.02.SPA.pkg
      Removed cat9k-sipspa.16.06.02.SPA.pkg
      Removed cat9k-srdriver.16.06.02.SPA.pkg
      Removed cat9k-webui.16.06.02.SPA.pkg
      Removed cat9k-wlc.16.06.02.SPA.pkg
      New files list:
      Added cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-espbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-sibase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Added cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
      Finished list of software package changes
  [1] Finished Activate on switch 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

*Oct 30 20:15:56.572: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:15:56 rollback_timer.sh:

%INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Install auto abort timer will expire in
7200 seconds
Install will reload the system now!
SUCCESS: install_activate Mon Oct 30 20:16:01 UTC 2017

Device#
*Oct 30 20:16:01.935: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:16:01
install_engine.sh: %INSTALL-5-INSTALL_COMPLETED_INFO: Completed install activate PACKAGE

Chassis 1 reloading, reason - Reload command

```

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 Mon Oct 30 20:53:33 UTC 2017

This operation requires a reload of the system. Do you want to proceed? [y/n]

*Oct 30 20:53:34.713: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:53:34
install_engine.sh: %INSTALL-5-INSTALL_START_INFO: Started install rollback

--- Starting Rollback ---
Performing Rollback on all members
  [1] Rollback package(s) on switch 1
      --- Starting rollback impact ---
      Changes that are part of this rollback
Current   : rp 0 0   rp_boot       cat9k-rpboot.16.06.02.prd9.SPA.pkg
Current   : rp 1 0   rp_boot       cat9k-rpboot.16.06.02.prd9.SPA.pkg
Replacement: rp 0 0   rp_boot       cat9k-rpboot.16.06.02.SPA.pkg
Replacement: rp 1 0   rp_boot       cat9k-rpboot.16.06.02.SPA.pkg
Current   : cc 0 0   cc_srdriver    cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 0 0   cc              cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 0 0   cc_spa          cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 1 0   cc_srdriver    cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 1 0   cc              cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 1 0   cc_spa          cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 10 0  cc              cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 10 0  cc_spa          cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 10 0  cc_srdriver    cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 2 0   cc_srdriver    cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 2 0   cc              cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 2 0   cc_spa          cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 3 0   cc_srdriver    cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 3 0   cc              cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 3 0   cc_spa          cat9k-sipspa.16.06.02.prd9.SPA.pkg

```

```

Current      : cc 4 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 4 0 cc              cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 4 0 cc_spa          cat9k-sipspa.16.06.02.pr99.SPA.pkg
Current      : cc 5 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 5 0 cc              cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 5 0 cc_spa          cat9k-sipspa.16.06.02.pr99.SPA.pkg
Current      : cc 6 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 6 0 cc              cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 6 0 cc_spa          cat9k-sipspa.16.06.02.pr99.SPA.pkg
Current      : cc 7 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 7 0 cc              cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 7 0 cc_spa          cat9k-sipspa.16.06.02.pr99.SPA.pkg
Current      : cc 8 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 8 0 cc              cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 8 0 cc_spa          cat9k-sipspa.16.06.02.pr99.SPA.pkg
Current      : cc 9 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 9 0 cc              cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 9 0 cc_spa          cat9k-sipspa.16.06.02.pr99.SPA.pkg
Current      : fp 0 0 fp              cat9k-esppbase.16.06.02.pr99.SPA.pkg
Current      : fp 1 0 fp              cat9k-esppbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 guestshell      cat9k-guestshell.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 rp_base          cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 rp_daemons      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 rp_iosd          cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 rp_security      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 rp_webui        cat9k-webui.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 rp_wlc          cat9k-wlc.16.06.02.pr99.SPA.pkg
Current      : rp 0 0 srdriver         cat9k-srdriver.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 guestshell      cat9k-guestshell.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 rp_base          cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 rp_daemons      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 rp_iosd          cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 rp_security      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 rp_webui        cat9k-webui.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 rp_wlc          cat9k-wlc.16.06.02.pr99.SPA.pkg
Current      : rp 1 0 srdriver         cat9k-srdriver.16.06.02.pr99.SPA.pkg
Replacement: cc 0 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 0 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 0 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 1 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 1 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 1 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 10 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 10 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 10 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 2 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 2 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 2 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 3 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 3 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 3 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 4 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 4 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 4 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 5 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 5 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 5 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 6 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 6 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 6 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 7 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 7 0 cc              cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 7 0 cc_spa          cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 8 0 cc_srdriver      cat9k-cc_srdriver.16.06.02.SPA.pkg

```

例：更新プログラムパッケージの管理

```

Replacement:  cc 8 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement:  cc 8 0  cc_spa      cat9k-sipspa.16.06.02.SPA.pkg
Replacement:  cc 9 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement:  cc 9 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement:  cc 9 0  cc_spa      cat9k-sipspa.16.06.02.SPA.pkg
Replacement:  fp 0 0  fp          cat9k-espbase.16.06.02.SPA.pkg
Replacement:  fp 1 0  fp          cat9k-espbase.16.06.02.SPA.pkg
Replacement:  rp 0 0  guestshell  cat9k-guestshell.16.06.02.SPA.pkg
Replacement:  rp 0 0  rp_base     cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 0 0  rp_daemons  cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 0 0  rp_iosd     cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 0 0  rp_security  cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 0 0  rp_webui    cat9k-webui.16.06.02.SPA.pkg
Replacement:  rp 0 0  rp_wlc      cat9k-wlc.16.06.02.SPA.pkg
Replacement:  rp 0 0  srdriver    cat9k-srdriver.16.06.02.SPA.pkg
Replacement:  rp 1 0  guestshell  cat9k-guestshell.16.06.02.SPA.pkg
Replacement:  rp 1 0  rp_base     cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 1 0  rp_daemons  cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 1 0  rp_iosd     cat9k-rpbase.16.06.02.SPA.pkg
Replacement:  rp 1 0  rp_security  cat9k-rpbase.16.06.02.SPA.pkg

```

Chassis 1 reloading, reason - Reload command

```

Replacement:  rp 1 0  rp_webui    cat9k-webui.16.06.02.SPA.pkg
Replacement:  rp 1 0  rp_wlc      cat9k-wlc.16.06.02.SPA.pkg
Replacement:  rp 1 0  srdriver    cat9k-srdriver.16.06.02.SPA.pkg

```

Finished rollback impact

```

[1] Finished Rollback on switch 1
Checking status of Rollback on [1]
Rollback: Passed on [1]
Finished Rollback

```

Install will reload the system now!

SUCCESS: install_rollback Mon Oct 30 20:54:23 UTC 2017

Device#

```

*Oct 30 20:54:23.576: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:54:23
install_engine.sh: %INSTALL-5-INSTALL_COMPLETED_INFO: Completed install rollback PACKAGE
*Oct 30 20:54:25.416: %STACKMGR-1-RELOAD: Switch 1 R0/0: stack_mgr:
Reloading due to reason Reload command Oct 30 20:54:31.615 FP0/0: %PMAN-5-EXITACTION:
Process manager is exiting: reload fp action requested
Oct 30 20:54

```

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 Mon Oct 30 19:51:48 UTC 2017
Cleaning up unnecessary package files
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...
done.

```

The following files will be deleted:

```

[switch 1]:
/flash/cat9k-cc_srdriver.16.06.02.SPA.pkg
/flash/cat9k-espbase.16.06.02.SPA.pkg
/flash/cat9k-guestshell.16.06.02.SPA.pkg
/flash/cat9k-rpbase.16.06.02.SPA.pkg
/flash/cat9k-rpboot.16.06.02.SPA.pkg
/flash/cat9k-sipbase.16.06.02.SPA.pkg
/flash/cat9k-sipspa.16.06.02.SPA.pkg

```

```

/flash/cat9k-srdriver.16.06.02.SPA.pkg
/flash/cat9k-webui.16.06.02.SPA.pkg
/flash/cat9k-wlc.16.06.02.SPA.pkg
/flash/packages.conf

Do you want to remove the above files? [y/n]
[switch 1]:
Deleting file flash:cat9k-cc_srdriver.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-espbase.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-guestshell.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-rpbase.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-rpboot.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-sipbase.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-sipspace.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-srdriver.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-webui.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-wlc.16.06.02.SPA.pkg ... done.
Deleting file flash:packages.conf ... done.
SUCCESS: Files deleted.
--- Starting Post_Remove_Cleanup ---
Performing Post_Remove_Cleanup on all members
  [1] Post_Remove_Cleanup package(s) on switch 1
  [1] Finished Post_Remove_Cleanup on switch 1
Checking status of Post_Remove_Cleanup on [1]
Post_Remove_Cleanup: Passed on [1]
Finished Post_Remove_Cleanup

SUCCESS: install_remove Mon Oct 30 19:52:25 UTC 2017
Device#

```

次に、**install abort** コマンドの出力例を示します。

```

Device# install abort

/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
install_abort: START Mon Oct 30 20:27:32 UTC 2017
install_abort: Abort type PACKAGE subtype NONE smutype NONE

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

*Oct 30 20:27:33.189: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine:
Started install abort
--- Starting Abort ---
Performing Abort on all members
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
  [1] Abort package(s) on switch 1
    --- Starting rollback impact ---
    Changes that are part of this rollback
    Current   : rp 0 0   rp_boot
cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
    Current   : rp 1 0   rp_boot
cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
    Replacement: rp 0 0   rp_boot          cat9k-rpboot.16.06.02.SPA.pkg
    Replacement: rp 1 0   rp_boot          cat9k-rpboot.16.06.02.SPA.pkg
    Current    : cc 0 0   cc_srdriver

```

例：更新プログラムパッケージの管理

```

cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 0 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 0 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 1 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 1 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 1 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 10 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 10 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 10 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 2 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 2 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 2 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 3 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 3 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 3 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 4 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 4 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 4 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 5 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 5 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 5 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 6 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 6 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 6 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 7 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 7 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 7 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 8 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 8 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 8 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 9 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 9 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 9 0 cc_spa

```

```

cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : fp 0 0 fp
cat9k-espbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : fp 1 0 fp
cat9k-espbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 guestshell
cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 rp_base
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 rp_daemons
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 rp_iosd
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 rp_security
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 rp_webui
cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 rp_wlc
cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 0 0 srdriver
cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 guestshell
cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 rp_base
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 rp_daemons
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 rp_iosd
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 rp_security
cat9k-rpbases.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 rp_webui
cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 rp_wlc
cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0 srdriver
cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Replacement: cc 0 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 0 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 0 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 1 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 1 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 1 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 10 0 cc         cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 10 0 cc_spa     cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 10 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 2 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 2 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 2 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 3 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 3 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 3 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 4 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 4 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 4 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 5 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 5 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 5 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 6 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 6 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 6 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 7 0 cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 7 0 cc           cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 7 0 cc_spa       cat9k-sipspa.16.06.02.SPA.pkg

```

```

Replacement: cc 8 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 8 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 8 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
Replacement: cc 9 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 9 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 9 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
Replacement: fp 0 0 fp cat9k-espbase.16.06.02.SPA.pkg
Replacement: fp 1 0 fp cat9k-espbase.16.06.02.SPA.pkg
Replacement: rp 0 0 guestshell cat9k-guestshell.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_base cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_daemons cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_iosd cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_security cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_webui cat9k-webui.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_wlc cat9k-wlc.16.06.02.SPA.pkg
Replacement: rp 0 0 srdriver cat9k-srdriver.16.06.02.SPA.pkg
Replacement: rp 1 0 guestshell cat9k-guestshell.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_base cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_daemons cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_iosd cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_security cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_webui cat9k-webui.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_wlc cat9k-wlc.16.06.02.SPA.pkg
Replacement: rp 1 0 srdriver cat9k-srdriver.16.06.02.SPA.pkg
Finished rollback impact
[1] Finished Abort on switch 1
Checking status of Abort on [1]
Abort: Passed on [1]
Finished Abort

```

```

/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
[1]: Performing MCU Upgrade Service
/usr/binos/conf/provfunc.sh: line 8792: $!_log_file: ambiguous redirect
SUCCESS: MCU Upgrade Service finished
Install will reload the system now!
SUCCESS: install_abort Mon Oct 30 20:28:21 UTC 2017
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function

```

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 Mon Oct 30 20:42:28 UTC 2017
install_activate: Activating PACKAGE

*Oct 30 20:42:29.149: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:42:29 install_engine.sh:

%INSTALL-5-INSTALL_START_INFO: Started install activateFollowing packages shall be
activated:
/flash/cat9k-wlc.16.06.02.prd9.SPA.pkg
/flash/cat9k-webui.16.06.02.prd9.SPA.pkg
/flash/cat9k-srdriver.16.06.02.prd9.SPA.pkg
/flash/cat9k-sipspa.16.06.02.prd9.SPA.pkg
/flash/cat9k-sipbase.16.06.02.prd9.SPA.pkg
/flash/cat9k-rpboot.16.06.02.prd9.SPA.pkg
/flash/cat9k-rpbase.16.06.02.prd9.SPA.pkg
/flash/cat9k-guestshell.16.06.02.prd9.SPA.pkg
/flash/cat9k-espbase.16.06.02.prd9.SPA.pkg
/flash/cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg

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

```



```
[1] Activate package(s) on switch 1
--- Starting list of software package changes ---
Old files list:
Removed cat9k-cc_srdriver.16.06.02.SPA.pkg
Removed cat9k-espbase.16.06.02.SPA.pkg
Removed cat9k-guestshell.16.06.02.SPA.pkg
Removed cat9k-rpbase.16.06.02.SPA.pkg
Removed cat9k-rpboot.16.06.02.SPA.pkg
Removed cat9k-sipbase.16.06.02.SPA.pkg
Removed cat9k-sipspa.16.06.02.SPA.pkg
Removed cat9k-srdriver.16.06.02.SPA.pkg
Removed cat9k-webui.16.06.02.SPA.pkg
Removed cat9k-wlc.16.06.02.SPA.pkg
New files list:
Added cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Added cat9k-espbase.16.06.02.prd9.SPA.pkg
Added cat9k-guestshell.16.06.02.prd9.SPA.pkg
Added cat9k-rpbase.16.06.02.prd9.SPA.pkg
Added cat9k-rpboot.16.06.02.prd9.SPA.pkg
Added cat9k-sipbase.16.06.02.prd9.SPA.pkg
Added cat9k-sipspa.16.06.02.prd9.SPA.pkg
Added cat9k-srdriver.16.06.02.prd9.SPA.pkg
Added cat9k-webui.16.06.02.prd9.SPA.pkg
Added cat9k-wlc.16.06.02.prd9.SPA.pkg
Finished list of software package changes
[1] Finished Activate on switch 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

*Oct 30 20:43:39.249: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:43:39 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 Mon Oct 30 20:43:44 UTC 2017

Device#
*Oct 30 20:43:44.615: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:43:44 install_engine.sh:

%INSTALL-5-INSTALL_COMPLETED_INFO: Completed install activate PACKAGE
Chassis 1 reloading, reason - Reload command
```

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:
```

例：デバイスを DHCP サーバとして設定

```
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 ブートローダコマンド	<i>Command Reference (Catalyst 9300 Series Switches)</i>
ハードウェアの設置	<i>Cisco Catalyst 9300 シリーズスイッチハードウェア設置ガイド</i>

デバイスセットアップ設定の実行に関する機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	デバイスのセットアップ設定	IP アドレス割り当てと DHCP の自動設定を含むデバイスセットアップ設定を実行できます。

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



第 4 章

ポリシーを使用したスマートライセンス

- [ポリシーを使用したスマートライセンシングの概要 \(109 ページ\)](#)
- [ポリシーを使用したスマートライセンシングに関する情報 \(110 ページ\)](#)
- [ポリシーを使用したスマートライセンシングの設定方法：トポロジ別のワークフロー \(136 ページ\)](#)
- [ポリシーを使用したスマートライセンシングへの移行 \(152 ページ\)](#)
- [ポリシーを使用したスマートライセンシングのタスクライブラリ \(175 ページ\)](#)
- [ポリシーを使用したスマートライセンシングのトラブルシューティング \(223 ページ\)](#)
- [ポリシーを使用したスマートライセンシングのその他の参考資料 \(235 ページ\)](#)
- [ポリシーを使用したスマートライセンシングの機能の履歴 \(236 ページ\)](#)

ポリシーを使用したスマートライセンシングの概要

ポリシーを使用したスマートライセンシングは、スマートライセンシングの拡張バージョンであり、ネットワークの運用を中断させないライセンスソリューションを提供するという主目的があります。むしろ、購入および使用しているハードウェアおよびソフトウェアライセンスを考慮してコンプライアンス関係を実現するライセンスソリューションを提供するという目的もあります。

Smart Licensing Using Policy は、Cisco IOS XE Amsterdam 17.3.2a 以降でサポートされます。

この拡張ライセンスモデルの主な利点は次のとおりです。

- シームレスな初日運用

ライセンスを注文した後は、輸出規制または適用ライセンスを使用しない限り、キーの登録や生成などの準備手順は必要ありません。Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチには、輸出規制ライセンスや適用ライセンスがなく、製品の機能をデバイスですぐに設定できます。

- Cisco IOS XE の一貫性

Cisco IOS XE ソフトウェアを実行するキャンパスおよび産業用イーサネットスイッチング、ルーティング、およびワイヤレスデバイスには、均一なライセンスエクスペリエンスがあります。

- 可視性と管理性

使用中の情報を把握するためのツール、テレメトリ、製品タギング。

- コンプライアンスを維持するための柔軟な時系列レポート

Cisco Smart Software Manager (CSSM) に直接または間接的に接続しているか、外部との接続性のないネットワークに接続しているかにかかわらず、簡単なレポートオプションを使用できます。

このドキュメントでは、Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチでのポリシーを使用したスマートライセンシングの概念、設定、およびトラブルシューティングについて説明します。

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

ポリシーを使用したスマートライセンシングに関する情報

このセクションでは、Smart Licensing Using Policy の実装に含めることができるコンポーネント、機能に関連する主要な概念、サポートされる製品、サポートされるすべてのトポロジーの概要（機能を実装するさまざまな方法）、Smart Licensing Using Policy が他の機能とどのように連携するかについて説明します。

概要

ポリシーを使用したスマートライセンシングは、ライセンスのさまざまな側面をシームレスに体験できるソフトウェアライセンス管理ソリューションです。

- ライセンスの購入：既存のチャネルからライセンスを購入し、Cisco Smart Software Manager (CSSM) ポータルを使用して製品インスタンスとライセンスを表示します。



(注) ポリシーを使用したスマートライセンシングの実装を簡素化するには、新しいハードウェアまたはソフトウェアを注文する際にスマートアカウントとバーチャルアカウントの情報を提供します。これにより、シスコは製造時に該当するポリシーおよび承認コード（用語は以下のセクション [概念 \(114 ページ\)](#) で説明) をインストールできます。

- 使用：Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチで使用可能なすべてのライセンスは適用されません。つまり、ソフトウェアとそれに関連付けられているライセンスの使用を開始する前に、キーの登録や生成などのライセンス固有の操作を完了する必要はありません。ライセンスの使用状況はタイムスタンプとともにデバイスに記録され、必要なワークフローは後日完了できます。

- ライセンスの使用状況を CSSM にレポート：ライセンス使用状況レポートには複数のオプションを使用できます。Cisco Smart Licensing Utility (CSLU) を使用し、使用状況情報を CSSM に直接報告し、コントローラ (Cisco DNA Center など) を使用し、Smart Software Manager オンプレミス (SSM オンプレミス) を展開して製品とライセンスをオンプレミスで管理できます。使用状況レポートはプレーンテキストの XML 形式です。[リソース使用率測定レポートの例 \(223 ページ\)](#) を参照してください。
- 調整：差分請求が適用される状況用 (購入と消費を比較して差分がある場合)。

アーキテクチャ

ここでは、ポリシーを使用したスマートライセンシングの実装に含めることができるさまざまなコンポーネントについて説明します。

製品インスタンス

製品インスタンスとは、Unique Device Identifier (UDI) によって識別されるシスコ製品の単一インスタンスです。

製品インスタンスは、ライセンス使用状況 (RUM レポート) を記録および報告し、期限切れのレポートや通信障害などに関するアラートとシステムメッセージを提供します。RUM レポートおよび使用状況データは、製品インスタンスに安全に保存されます。

このドキュメントでは、「製品インスタンス」という用語は、特に明記しない限り、サポートされているすべての物理および仮想製品インスタンスを指します。このドキュメントの範囲内にある製品インスタンスについては、[サポート対象製品 \(128 ページ\)](#) を参照してください。

CSLU

Cisco Smart License Utility (CSLU) は、集約ライセンスワークフローを提供する Windows ベースのレポートユーティリティです。このユーティリティが実行する主な機能は次のとおりです。

- ワークフローのトリガー方法に関するオプションを提供します。ワークフローは、CSLU や製品インスタンスによってトリガーできます。
- 製品インスタンスから使用状況レポートを収集し、その使用状況レポートを対応するスマートアカウントやバーチャルアカウントにアップロードします。オンラインでもオフライン (ファイルを使用) でも可能です。同様に、RUM レポート ACK をオンラインまたはオフラインで収集し、製品インスタンスに返送します。
- 承認コード要求を CSSM に送信し、CSSM から承認コードを受信します (該当する場合)。

CSLU は、次の方法で実装に含めることができます。

- CSSM に接続されているスタンドアロンツールとして CSLU を使用するための Windows アプリケーションをインストールします。

- CSSM から切断されているスタンドアロンツールとして CSLU を使用するための Windows アプリケーションをインストールします。このオプションを使用すると、必要な使用状況情報がファイルにダウンロードされ、CSSM にアップロードされます。これは、外部と接続していないネットワークに適しています。

CSSM

Cisco Smart Software Manager (CSSM) は、一元化された場所からすべてのシスコソフトウェアライセンスを管理できるポータルです。CSSM は、現在の要件を管理し、将来のライセンス要件を計画するための使用傾向を確認するのに役立ちます。

CSSM Web UI には <https://software.cisco.com> でアクセスできます。[License] タブで、[Smart Software Licensing] のリンクをクリックします。

CSSM に接続できるさまざまな方法については、[サポートされるトポロジ \(118 ページ\)](#) のセクションを参照してください

CSSM では、次のことができます。

- 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。

コントローラ

複数の製品インスタンスを管理する管理アプリケーションまたはサービス。

Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチでは、Cisco DNA Center がサポートされるコントローラです。コントローラ、コントローラをサポートする製品インスタンス、およびコントローラと製品インスタンスに必要な最小ソフトウェアバージョンに関する情報を次に示します。

表 7: コントローラのサポート情報 : Cisco DNA Center

Smart Licensing Using Policy へ移行するために必要な Cisco DNA Center の最小バージョン ¹	Cisco IOS XE に必要な最小バージョン ²	サポート対象製品インスタンス
Cisco DNA Center リリース 2.2.2	Cisco IOS XE Amsterdam 17.3.2a	<ul style="list-style-type: none"> • Cisco Catalyst 9200 シリーズ スイッチ • Cisco Catalyst 9300 シリーズ スイッチ • Cisco Catalyst 9400 シリーズ スイッチ • Cisco Catalyst 9500 シリーズ スイッチ • Cisco Catalyst 9600 シリーズ スイッチ

¹ コントローラに必要な最小ソフトウェアバージョン。これは、特に明記されていない限り、後続のすべてのリリースでサポートが継続されることを意味します。

² 製品インスタンスに必要な最小ソフトウェアバージョン。これは、特に明記されていない限り、後続のすべてのリリースでサポートが継続されることを意味します。

Cisco DNA Center の詳細については、

<https://www.cisco.com/c/en/us/support/cloud-systems-management/dna-center/series.html> [英語] でサポートページを参照してください。

SSM オンプレミス

Smart Software Manager オンプレミス (SSM オンプレミス) は、CSSM と連動するアセットマネージャです。これにより、CSSM に直接接続する代わりに、オンプレミスで製品とライセンスを管理できます。

SSM オンプレミスで Smart Licensing Using Policy を実装するために必要なソフトウェアバージョンについては、次を参照してください。

Smart Licensing Using Policy に必要な SSM オンプレミスの最小バージョン ³	Cisco IOS XE に必要な最小バージョン ⁴	サポート対象製品インスタンス
バージョン 8、リリース 202102	Cisco IOS XE Amsterdam 17.3.3	<ul style="list-style-type: none"> • Cisco Catalyst 9200 シリーズ スイッチ • Cisco Catalyst 9300 シリーズ スイッチ • Cisco Catalyst 9400 シリーズ スイッチ • Cisco Catalyst 9500 シリーズ スイッチ • Cisco Catalyst 9600 シリーズ スイッチ

³ 必要な SSM オンプレミスの最小バージョンこれは、特に明記されていない限り、後続のすべてのリリースでサポートが継続されることを意味します。

⁴ 製品インスタンスに必要な最小ソフトウェアバージョン。これは、特に明記されていない限り、後続のすべてのリリースでサポートが継続されることを意味します。

SSM オンプレミスの詳細については、ソフトウェアダウンロードページの [Smart Software Manager On-Prem](#) [英語] を参照してください。ドキュメントリンクを表示するには、.iso イメージにカーソルを合わせます。

概念

ここでは、ポリシーを使用したスマートライセンスの主要な概念について説明します。

ライセンス執行（エンフォースメント）タイプ

所与のライセンスは、3つの適用タイプのいずれかに属します。適用タイプは、ライセンスを使用する前に承認が必要かどうかを示します。

- 不適用または非適用

不適用ライセンスは、外部との接続がないネットワークで使用する前、または接続されたネットワークでの登録前に承認を必要としません。このようなライセンスの使用条件は、エンドユーザライセンス契約（EULA）に基づきます。

Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチで使用可能なすべてのライセンスは、不適用ライセンスです。

- 適用

この適用タイプに属するライセンスは、使用前に承認が必要です。必要な承認は承認コードの形式で行われ、対応する製品インスタンスにインストールする必要があります。

適用ライセンスの例としては、シスコの産業用イーサネットスイッチで利用可能な **Media Redundancy Protocol (MRP)** クライアントライセンスがあります。

- 輸出規制

この適用タイプに属するライセンスは米国の取引規制法によって輸出が制限されていて、これらのライセンスは使用前に承認が必要です。これらのライセンスの場合も、必要な承認コードは、対応する製品インスタンスにインストールする必要があります。シスコは、ハードウェア購入の際に発注がある場合、輸出規制ライセンスをプリインストールすることがあります。

輸出規制ライセンスの例としては、シスコの特定のルータで使用可能な高速暗号化 (HSECK9) ライセンスがあります。

ライセンス継続期間

これは、購入したライセンスが有効な期間を指します。所与のライセンスは、上記のいずれかの適用タイプに属し、次の期間有効です。

- 永久：このライセンスには使用期限日はありません。

Network Essentials および **Network Advantage** ライセンスは、**Cisco Catalyst** アクセス、コア、およびアグリゲーションスイッチで使用可能な永続的ライセンスの例です。

- サブスクリプション：ライセンスは特定の日付まで有効です。

Digital Network Architecture (DNA) Essentials および **DNA Advantage** ライセンスは、**Cisco Catalyst** アクセス、コア、およびアグリゲーションスイッチで使用可能なサブスクリプションライセンスの例です。

承認コード

スマートライセンシング承認コード (SLAC) は、輸出規制または適用 (エンフォース) ライセンスの有効化および継続使用を可能にします。

SLAC は、**Cisco Catalyst** アクセス、コア、およびアグリゲーションスイッチで使用可能なライセンスには必要ありませんが、以前のライセンシングモデルからポリシーを使用したスマートライセンシングにアップグレードする場合は、独自の承認コードを含む特定のライセンスの予約 (SLR) があるかもしれません。SLR 承認コードは、**Smart Licensing Using Policy** へのアップグレード後にサポートされます。



- (注) 既存の SLR はアップグレード後に引き継がれますが、「予約」の概念が適用されないため、ポリシーを使用したスマートライセンシング環境で新しい SLR を要求することはできません。完全に外部との接続性がないネットワーク内にいる場合は、代わりに **CSSM への接続なし**、**CSLU なし** のトポロジが適用されます。

SLR承認コードの処理方法の詳細については、[アップグレード \(130ページ\)](#) を参照してください。SLR承認コードを返す場合は、[承認コードの削除と返却 \(207ページ\)](#) を参照してください。

ポリシー

ポリシーは、製品インスタンスに次のレポート手順を提供します。

- **License usage report acknowledgement requirement (Reporting ACK required)** : ライセンス使用状況レポートは RUM レポートと呼ばれ、確認応答は ACK と呼ばれます (「[RUM レポートおよびレポート確認応答](#)」を参照)。これは、この製品インスタンスのレポートに CSSM 確認応答が必要かどうかを指定する yes または no の値です。デフォルトポリシーは常に「yes」に設定されます。

- **First report requirement (days)** : 最初のレポートは、ここで指定した期間内に送信される必要があります。

この値がゼロの場合、最初のレポートは必要ありません。

- **Reporting frequency (days)** : 後続のレポートは、ここで指定した期間内に送信される必要があります。

この値がゼロの場合、使用状況が変更されない限り、以降のレポートは必要ありません。

- **Report on change (days)** : ライセンスの使用状況が変更された場合は、ここで指定した期間内にレポートが送信される必要があります。

この値がゼロの場合、使用状況の変更時のレポートは必要ありません。

この値がゼロでない場合は、変更を加えた後にレポートが必要です。次に示すすべてのシナリオは、製品インスタンスのライセンス使用状況における変更としてカウントされます。

- 消費されたライセンスの変更 (別のライセンスへの変更やライセンスの追加または削除を含む)。
- ライセンスの消費なしから 1 つ以上のライセンスの消費への移行。
- 1 つ以上のライセンスの消費からライセンスの消費なしへの移行。



(注) 製品インスタンスがライセンスを使用していない場合、ポリシーのレポート要件 (最初のレポート要件、レポート頻度、変更に関するレポート) のいずれかにゼロ以外の値が設定されていても、レポートは必要ありません。

ポリシー選択について

CSSMは、製品インスタンスに適用されるポリシーを決定します。特定の時点で使用されているポリシーは1つだけです。ポリシーとその値は、使用されているライセンスなど、さまざまな要因に基づいています。

Cisco defaultは、製品インスタンスで常に使用可能なデフォルトポリシーです。他のポリシーが適用されていない場合、製品インスタンスはこのデフォルトポリシーを適用します。次の表（表 8：ポリシー：Cisco default（117 ページ））に、Cisco default ポリシー値を示します。

お客様はポリシーを設定することはできませんが、Cisco Global Licensing Operations チームに連絡して、カスタマイズされたポリシーを要求することができます。Support Case Manager に移動します。[OPEN NEW CASE] をクリックして、[Software Licensing] を選択します。ライセンスチームから、プロセスの開始や追加情報について連絡があります。カスタマイズされたポリシーは、CSSM のスマートアカウントを介して使用することもできます。



- (注) 適用されているポリシー（使用中のポリシー）とそのレポート要件を確認するには、特権EXECモードで **show license all** コマンドを入力します。

表 8: ポリシー：Cisco default

ポリシー：Cisco default	デフォルトポリシー値
Export (Perpetual/Subscription) (注) 適用タイプが「輸出規制」のライセンスにのみ適用されます。	Reporting ACK required : Yes First report requirement (days) : 0 Reporting frequency (days) : 0 Report on change (days) : 0
Enforced (Perpetual/Subscription) (注) 適用タイプが「適用（エンフォース）」のライセンスにのみ適用されます。	Reporting ACK required : Yes First report requirement (days) : 0 Reporting frequency (days) : 0 Report on change (days) : 0
Unenforced/Non-Export Perpetual ⁵	Reporting ACK required : Yes First report requirement (days) : 365 Reporting frequency (days) : 0 Report on change (days) : 90

ポリシー : Cisco default	デフォルトポリシー値
Unenforced/Non-Export Subscription	Reporting ACK required : Yes First report requirement (days) : 90 Reporting frequency (days) : 90 Report on change (days) : 90

⁵ Unenforced/Non-Export Perpetual の場合：デフォルトポリシーの最初のレポート要件（365 日以内）は、ディストリビュータやパートナーからハードウェアやソフトウェアを購入した場合にのみ適用されます。

RUM レポートおよびレポート確認応答

リソース使用率測定レポート（RUM レポート）は、ポリシーで指定されたレポート要件を満たすために製品インスタンスが生成するライセンス使用状況レポートです。

確認応答（ACK）は CSSM からの応答であり、RUM レポートのステータスに関する情報を提供します。

製品インスタンスに適用されるポリシーによって、次のレポート要件が決まります。

- RUM レポートが CSSM に送信されるかどうか、およびこの要件を満たすために提供される最大日数。
- RUM レポートに CSSM からの確認応答（ACK）が必要かどうか。
- ライセンス消費の変化を報告するために提供される最大日数。

RUM レポートには、信頼コード要求や SLAC 要求などの他の要求が伴う場合があります。そのため、受信した RUM レポート ID に加えて、CSSM からの ACK には承認コード、信頼コード、およびポリシーファイルが含まれることがあります。

レポート方式、つまり CSSM への RUM レポートの送信方法は、実装するトポロジによって異なります。

信頼コード

製品インスタンスが RUM レポートに署名するために使用する、UDI に関連付けられた公開キー。これにより、改ざんが防止され、データの真正性が確保されます。

サポートされるトポロジ

このセクションでは、ポリシーを使用したスマートライセンシングを実装するさまざまな方法について説明します。各トポロジについて、付属の概要を参照してセットアップの動作設計を確認し、考慮事項と推奨事項（ある場合）を参照してください。

トポロジを選択した後

トポロジを選択した後、[ポリシーを使用したスマートライセンシングの設定方法：トポロジ別のワークフロー（136ページ）](#)を参照してください。これらのワークフローは、新規展開のみに該当します。これらのワークフローにより、トポロジを実装する最も簡単に迅速な方法が実現します。

既存のライセンシングモデルから移行する場合は、[ポリシーを使用したスマートライセンシングへの移行（152ページ）](#)を参照してください。

追加の設定タスクを実行する場合（たとえば別のライセンスを設定する場合、アドオンライセンスを使用する場合、またはより短いレポート間隔を設定する場合）は、[ポリシーを使用したスマートライセンシングのタスクライブラリ（175ページ）](#)を参照してください。続行する前に、入手可能な場合には「サポートされるトポロジ」を確認してください。

CSLU を介して CSSM に接続

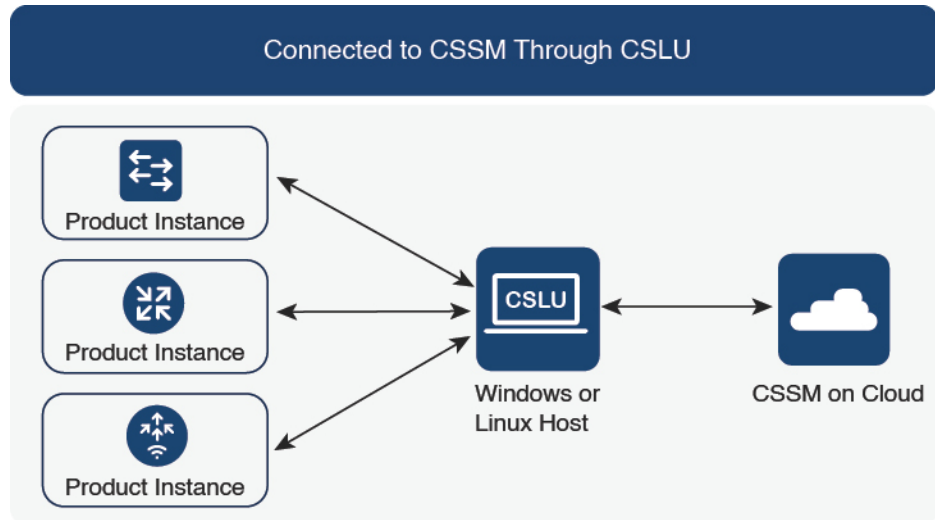
概要：

ここでは、ネットワーク内の製品インスタンスは CSLU に接続され、CSLU は CSSM との単一のインターフェイスポイントになります。製品インスタンスは、必要な情報を CSLU にプッシュするように設定できます。または、構成可能な頻度で製品インスタンスから必要な情報を取得するように CSLU を設定することもできます。

製品インスタンス開始型通信（プッシュ）：製品インスタンスは、CSLU の REST エンドポイントに接続することで、CSLU との通信を開始します。送信されるデータには、RUM レポート、および承認コード、信頼コードの要求が含まれます。必要な間隔で自動的に RUM レポートを CSLU に送信するように製品インスタンスを設定できます。これは、製品インスタンスのデフォルトの方法です。

CSLU 開始型通信（pull 型）：製品インスタンスからの情報の取得を開始するために、CSLU は YANG を使用した NETCONF、RESTCONF、gRPC のモデル、またはネイティブ REST API を使用して製品インスタンスに接続します。サポートされるワークフローには、RUM レポートの製品インスタンスからの受信と CSSM への送信、承認コードのインストール、信頼コードのインストール、およびポリシーの適用が含まれます。

図 5: トポロジ : CSLU を介して CSSM に接続



考慮事項または推奨事項 :

ネットワークのセキュリティポリシーに応じて通信方法を選択します。

次の手順 :

このトポロジを実装するには、[トポロジのワークフロー : CSLU を介して CSSM に接続 \(137 ページ\)](#) を参照してください。

CSSM に直接接続

概要 :

このトポロジは、スマートライセンシングの以前のバージョンで使用でき、ポリシーを使用したスマートライセンシングで引き続きサポートされます。

ここでは、製品インスタンスから CSSM への直接かつ信頼できる接続を確立します。直接接続には、CSSM へのネットワーク到達可能性が要求されます。その後、製品インスタンスがメッセージを交換し、CSSM と通信するには、このトポロジで使用可能な転送オプションのいずれかを設定します (以下を参照)。最後に、信頼を確立するには、CSSM の対応するスマートアカウントとバーチャルアカウントからトークンを生成し、製品インスタンスにインストールする必要があります。

次の方法で CSSM と通信するように製品インスタンスを設定できます。

- スマート転送を使用して CSSM と通信する。

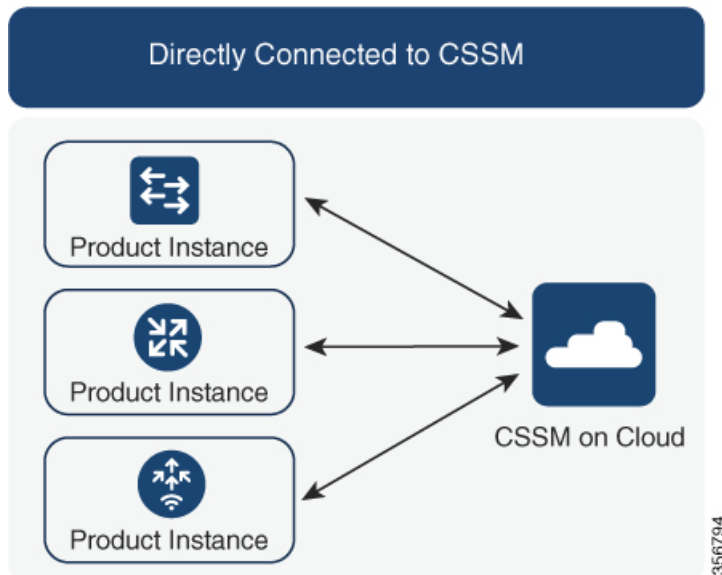
スマート転送は、スマートライセンシング (JSON) メッセージが HTTPS メッセージ内に含まれ、製品インスタンスと CSSM の間で交換されることにより通信する転送方法です。次のスマート転送設定オプションを使用できます。

- スマート転送：この方法では、製品インスタンスは特定のスマート転送ライセンスサーバ URL を使用します。これは、ワークフローのセクションに示すとおりを設定する必要があります。
- HTTPS プロキシを介したスマート転送：この方法では、製品インスタンスはプロキシサーバを使用してライセンスサーバと通信し、最終的には CSSM と通信します。
- Call Home を使用して CSSM と通信する。

Call Home を使用すると、E メールベースおよび Web ベースで重大なシステム イベントの通知を行えます。CSSM へのこの接続方法は、以前のスマートライセンシング環境で使用でき、ポリシーを使用したスマートライセンシングで引き続き使用できます。次の Call Home 設定オプションを使用できます。

- ダイレクトクラウドアクセス：この方法では、製品インスタンスはインターネット経由で CSSM 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。
- 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、製品インスタンスはインターネット経由でプロキシサーバ（Call Home Transport Gateway または市販のプロキシ（Apache など）のいずれか）を介して CSSM に使用状況情報を送信します。

図 6: トポロジ：CSSM に直接接続



考慮事項または推奨事項：

CSSM に直接接続する場合は、スマート転送が推奨される転送方法です。この推奨事項は以下に適用されます。

- 新規展開。

- 以前のライセンスモデル。ポリシーを使用したスマートライセンシングへの移行後に設定を変更します。
- 現在 Call Home 転送方法を使用している登録済みライセンス。ポリシーを使用したスマートライセンシングへの移行後に設定を変更します。
- 以前のライセンスモデルの評価ライセンスや期限切れのライセンス。ポリシーを使用したスマートライセンシングへの移行後に設定を変更します。

移行後に設定を変更するには、[トポロジのワークフロー：CSSMに直接接続（139 ページ）](#)の「製品インスタンスの設定」にある「接続方法と転送タイプの設定」のオプション1を参照してください。

次の手順：

このトポロジを実装するには、[トポロジのワークフロー：CSSMに直接接続（139 ページ）](#)を参照してください。

コントローラを介して CSSM に接続

コントローラを使用して製品インスタンスを管理する場合、コントローラはCSSMに接続してCSSMとのすべての通信のインターフェイスとなります。Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチのサポートされるコントローラは、Cisco DNA Center です。

概要

Cisco DNA Center がコントローラとして製品インスタンスを管理している場合、製品インスタンスはライセンスの使用状況を記録し、保存しますが、Cisco DNA Center が RUM レポートを取得し、CSSM に報告し、製品インスタンスにインストールするために ACK を返すために製品インスタンスとの通信を開始します。

Cisco DNA Center で管理する必要があるすべての製品インスタンスは、そのインベントリの一部である必要があり、サイトに割り当てる必要があります。Cisco DNA Center は NETCONF プロトコルを使用して設定をプロビジョニングし、製品インスタンスから必要な情報を取得します。したがって、これを容易にするために製品インスタンスで NETCONF を有効にする必要があります。

レポートの要件を満たすために、Cisco DNA Center は CSSM から該当するポリシーを取得し、次のレポートオプションを提供します。

- Ad hoc reporting：必要に応じてアドホックレポートをトリガーできます。
- Scheduled reporting：ポリシーで指定されたレポート頻度に対応し、Cisco DNA Center によって自動的に処理されます。



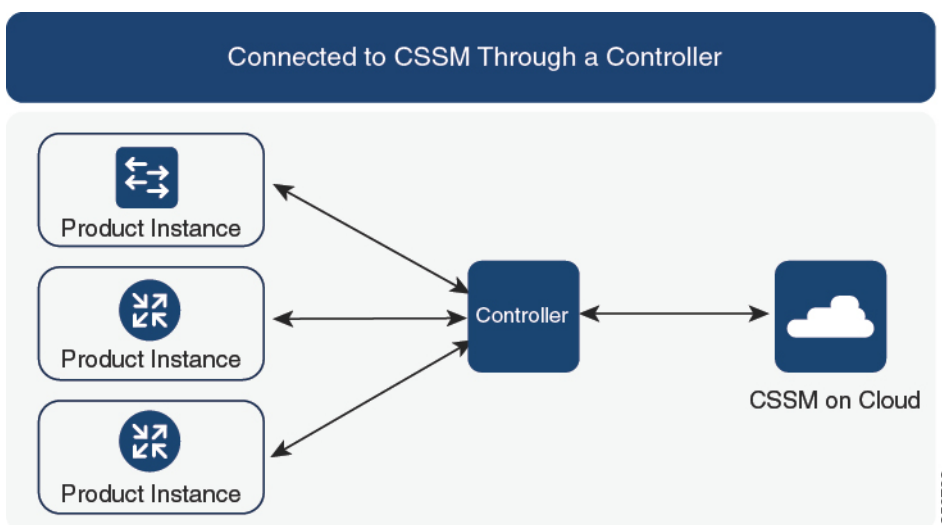
(注) 製品インスタンスが定期レポートの対象となる前に、アドホックレポートを少なくとも1回実行する必要があります。

最初のアドホックレポートにより、Cisco DNA Center は、後続の RUM レポートをアップロードする必要があるスマートアカウントとバーチャルアカウントを決定できます。製品インスタンスのアドホックレポートが一度も実行されていない場合は、通知されます。

Cisco DNA Center では、輸出規制ライセンス用の SLAC のインストールと削除ができます。Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチのすべての使用可能なライセンスは不適用ライセンスであるため、SLAC のインストールと削除は適用されません。

信頼コードは必要ありません。

図 7: トポロジ: コントローラを介して CSSM に接続



考慮事項または推奨事項 :

これは、Cisco DNA Center を使用している場合に推奨されるトポロジです。

次の手順 :

このトポロジを実装するには、[トポロジのワークフロー: コントローラを介して CSSM に接続 \(141 ページ\)](#) を参照してください。

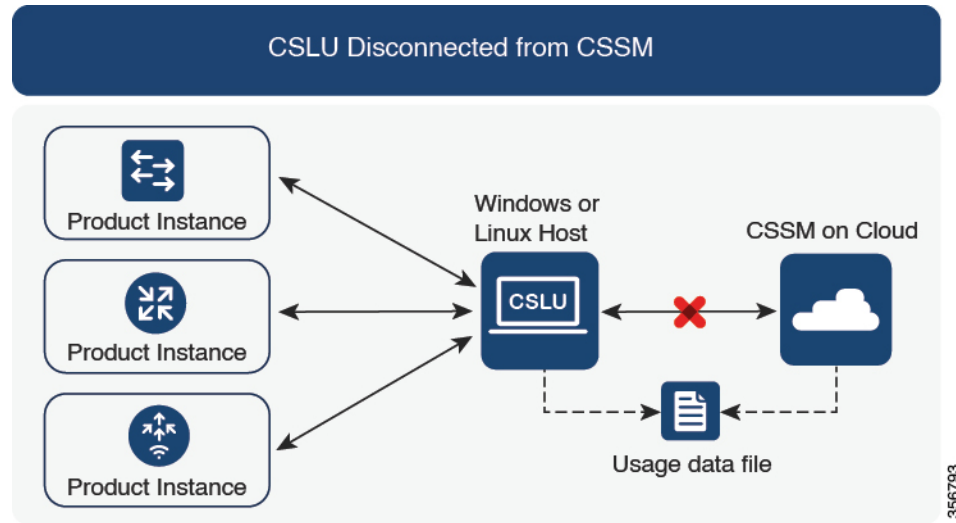
CSLU は CSSM から切断

概要 :

ここでは、製品インスタンスが CSLU と通信し、製品インスタンス開始の通信または CSLU 開始の通信を実装するオプションがあります (CSLU を介して CSSM に接続のトポロジと同様)。CSLU と CSSM 間の通信のもう一方はオフラインです。CSLU には、CSSM から切断されたモードで動作するオプションがあります。

CSLU と CSSM 間の通信は、署名済みファイルの形式で送受信され、オフラインで保存された後、場合によっては CSLU または CSSM にアップロードまたはダウンロードされます。

図 8: トポロジ : CSLU は CSSM から切断



考慮事項または推奨事項 :

なし。

次の手順 :

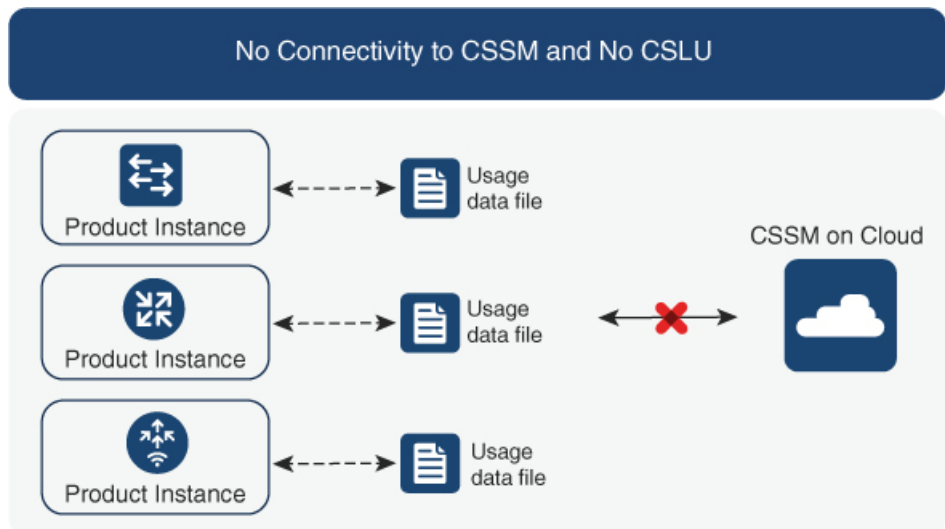
このトポロジを実装するには、[トポロジのワークフロー : CSLU は CSSM から切断 \(142 ページ\)](#) を参照してください。

CSSM への接続なし、CSLU なし

概要 :

ここでは、製品インスタンスと CSSM は相互に切断され、他の中間ユーティリティまたはコンポーネントはありません。すべての通信は、ファイルのアップロードとダウンロードという形式です。

図 9: トポロジ : **CSSM** への接続なし、**CSLU** なし



考慮事項または推奨事項 :

このトポロジは、製品インスタンスがネットワークの外部とオンラインで通信できない高セキュリティ展開に適しています。

次の手順 :

このトポロジを実装するには、[トポロジのワークフロー : CSSM への接続なし、CSLU なし \(145 ページ\)](#) を参照してください。

SSM オンプレミス展開

概要 :

SSM オンプレミスは、オンプレミスに展開される CSSM の拡張として機能するように設計されています。

ここでは、製品インスタンスが SSM オンプレミスに接続され、SSM オンプレミスが CSSM との単一のインターフェイスポイントになります。SSM オンプレミスの各インスタンスは、SSM オンプレミスのローカルアカウントに必須の登録と同期を通じて、CSSM 内のバーチャルアカウントを使用して CSSM に通知する必要があります。

製品インスタンスを管理するために SSM オンプレミスを展開する場合、SSM オンプレミスに必要な情報をプッシュするように製品インスタンスを設定できます。または、設定可能な頻度で製品インスタンスから必要な情報をプルするように SSM オンプレミスを設定することもできます。

- **製品インスタンス開始型通信 (プッシュ) :** 製品インスタンスは SSM オンプレミスの REST エンドポイントを接続することで SSM オンプレミスの通信を開始します。送信されるデータには、RUM レポート、および承認コード、信頼コード、ポリシーの要求が含まれます。

このモードでの製品インスタンスと SSM オンプレミス間の通信のオプション：

- 必要に応じて、CLI コマンドを使用して SSM オンプレミスに情報をプッシュします。
- スケジュールされた頻度で RUM レポートを SSM オンプレミスに自動的に送信するには、CLI コマンドを使用し、レポート間隔を設定します。
- SSM オンプレミス開始型通信（プル）：製品インスタンスからの情報の取得を開始するには、SSM オンプレミスで NETCONF、RESTCONF、およびネイティブの REST API オプションを使用して製品インスタンスを接続します。サポートされるワークフローには、RUM レポートの製品インスタンスからの受信と CSSM への送信、承認コードのインストール、信頼コードのインストール、およびポリシーの適用が含まれます。

このモードでの製品インスタンスと SSM オンプレミス間の通信のオプション：

- 必要に応じて（オンデマンドで）、1 つ以上の製品インスタンスから使用状況情報を収集します。
- スケジュールされた頻度で 1 つ以上の製品インスタンスから使用状況情報を収集します。

SSM オンプレミスでは、レポート間隔が製品インスタンスのデフォルトポリシーに設定されます。これは変更できますが、より頻繁に（より短い間隔で）レポートを作成するか、または使用可能な場合はカスタムポリシーをインストールできます。

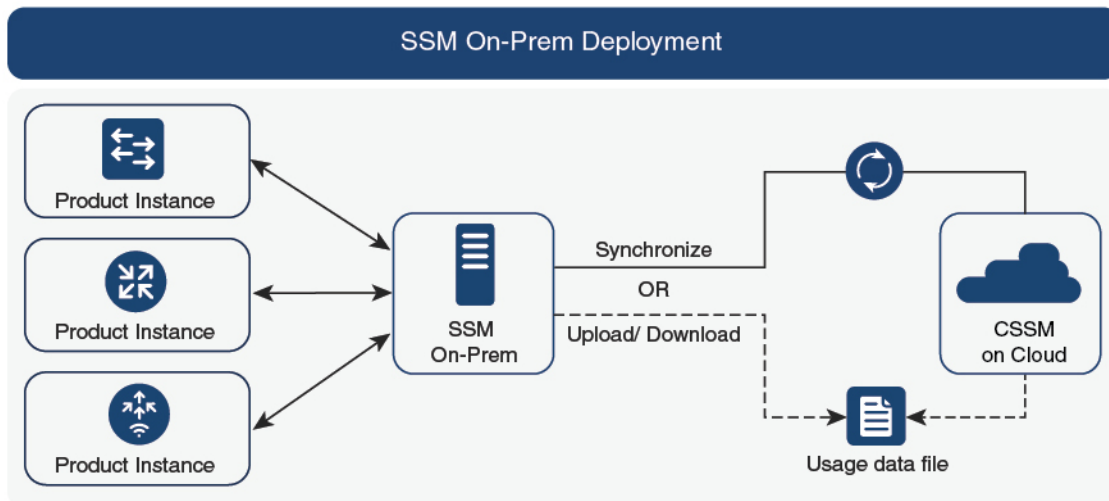
SSM オンプレミスで使用状況が使用できるようになったら、同じ間隔で CSSM と同期して、製品インスタンス数、ライセンス数、およびライセンス使用状況情報が CSSM と SSM オンプレミスの両方と同じであることを確認します。SSM オンプレミスと CSSM 間の使用状況の同期オプション：プッシュとプルモードの場合：

- CSSM でアドホック同期を実行します（Cisco と同期されました）。
- 指定した時刻で CSSM との同期をスケジュールします。
- オフラインで保存されている指名済みファイルを通じて CSSM と通信し、場合によって SSM オンプレミスまたは CSSM からアップロードするか、またはダウンロードします。



- (注) このトポロジでは、SSM オンプレミスと CSSM 間で 2 つの異なる同期が行われます。1 つは、ローカルアカウントと CSSM との同期です。この同期は、SSM オンプレミスインスタンスに CSSM を認識させるためであり、SSM オンプレミスの [Synchronization] ウィジェットを使用して実行します。2 番目は、CSSM に接続するか、またはファイルをダウンロードおよびアップロードすることのいずれかによるライセンスの使用状況の CSSM との同期です。ライセンスの使用状況を同期する前に、ローカルアカウントを同期する必要があります。

図 10: トポロジ : SSM オンプレミス展開



357508

考慮事項または推奨事項 :

このトポロジは、次の状況に適しています。

- CSSM と直接通信せずにオンプレミスで製品インスタンスを管理する場合。
- 会社のポリシーにより、製品インスタンスでライセンスの使用状況をシスコ (CSSM) に直接報告できない場合。
- 製品インスタンスがエアギャップネットワーク内にあり、ネットワーク外にあるものとオンラインで通信できない場合。

Smart Licensing Using Policy のサポートとは別に、SSM オンプレミスのバージョン 8 の主な利点は次のとおりです。

- マルチテナント : 1 つのテナントが 1 つのスマートアカウントとバーチャルアカウントのペアを構成します。SSM オンプレミスでは複数のペアを管理できます。ここでは、SSM オンプレミスに存在するローカルアカウントを作成します。CSSM のスマートアカウントとバーチャルアカウントのペアへの複数のローカルアカウントのロールアップ。詳細については、『Cisco Smart Software Manager On-Prem User Guide』 [英語] の「About Accounts and Local Virtual Accounts」を参照してください。



(注) CSSM と SSM オンプレミスのインスタンス間の関係は、まだ 1 対 1 です。

- スケール : 合計 300,000 の製品インスタンスをサポートします。
- 高可用性 : 2 台の SSM オンプレミスサーバをアクティブ/スタンバイクラスタの形式で実行できます。詳細については、『Cisco Smart Software On-Prem Installation Guide』 [英語] の

「Appendix 4 Managing a High Availability (HA) Cluster in Your System」を参照してください。

高可用性展開は SSM オンプレミスのコンソールでサポートされており、必要なコマンドの詳細については『[Cisco Smart Software On-Prem Console Guide](#)』で確認できます。

- CSSM へのオンライン接続とオフライン接続のオプション。

SSM オンプレミスの制限：

- ライセンス使用の同期を目的とした CSSM との通信のプロキシサポートが利用できるのは、バージョン 8202108 以降のみです。ローカルアカウントの同期を目的とするプロキシの使用はサポートされています。これは [Synchronization] ウィジェットを使用して実行され、Smart Licensing Using Policy がサポートされている SSM オンプレミス導入リリースから利用可能です。
- SSM オンプレミス開始型通信は、ネットワークアドレス変換 (NAT) 設定の製品インスタンスではサポートされていません。製品インスタンス開始型通信を使用する必要があります。さらに、NAT 設定の製品インスタンスをサポートするために SSM オンプレミスを有効にする必要があります。詳細は、このトポロジのワークフローで提供されます。

次の手順：

このトポロジを実装するには、[トポロジのワークフロー：SSM オンプレミス展開（146 ページ）](#)を参照してください。

SSM オンプレミスの既存のバージョンから移行する場合は、アップグレード関連のさまざまなアクティビティを実行する順序が重要です。[Smart Licensing Using Policy をサポートする SSM オンプレミスのバージョンへの移行（173 ページ）](#)を参照してください。

サポート対象製品

このセクションでは、本マニュアルの対象範囲に含まれる、ポリシーを使用したスマートライセンスをサポートする Cisco IOS-XE 製品インスタンスについての情報を提供します。特に指定のない限り、製品シリーズのすべてのモデル（製品 ID または PID）がサポートされます。

表 9: サポートされる製品インスタンス：Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチ

Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチ	サポートが導入されたバージョン
Cisco Catalyst 9200 シリーズ スイッチ	Cisco IOS XE Amsterdam 17.3.2a
Cisco Catalyst 9300 シリーズ スイッチ	Cisco IOS XE Amsterdam 17.3.2a
Cisco Catalyst 9400 シリーズ スイッチ	Cisco IOS XE Amsterdam 17.3.2a
Cisco Catalyst 9500 シリーズ スイッチ	Cisco IOS XE Amsterdam 17.3.2a

Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチ	サポートが導入されたバージョン
Cisco Catalyst 9600 シリーズ スイッチ	Cisco IOS XE Amsterdam 17.3.2a

他の機能との相互作用

ハイ アベイラビリティ

このセクションでは、ポリシーを使用したスマートライセンシングをサポートするソフトウェアバージョンを実行するときに、高可用性設定に適用される考慮事項について説明します。次の高可用性セットアップは、このドキュメントの範囲内です。

1つのアクティブ、1つのスタンバイ、および1つ以上のメンバーで構成されるデバイススタック

デュアル RP (ルートプロセッサ) セットアップ。1つのシャーシに2つの RP がインストールされ、1つはアクティブ、もう1つはスタンバイです。

デュアルシャーシセットアップ⁶ (固定またはモジュラ)。一方のシャーシにアクティブ、もう一方のシャーシにスタンバイがあります。

モジュラシャーシでの、デュアルシャーシとデュアル RP のセットアップ⁷。ここでも2つのシャーシが関係し、1つのシャーシにアクティブ RP、もう1つのシャーシにスタンバイ RP があります。デュアル RP とは、最小要件である1つのシャーシだけに追加のシャーシ内スタンバイ RP、または各シャーシにシャーシ内スタンバイ RP があることを指します。

高可用性セットアップでの信頼コード要件

必要な信頼コードの数は、UDIの数によって異なります。アクティブな製品インスタンスは、高可用性セットアップのすべてのデバイスに対する要求を送信し、ACK で返されるすべての信頼コードをインストールできます。

高可用性セットアップでのポリシー要件

高可用性セットアップにのみ適用されるポリシー要件はありません。スタンドアロン製品インスタンスの場合と同様に、高可用性セットアップにも1つのポリシーのみが存在し、これがアクティブになります。アクティブのポリシーは、セットアップのスタンバイまたはメンバーに適用されます。

高可用性セットアップでの製品インスタンス機能

このセクションでは、高可用性セットアップでの一般的な製品インスタンス機能と、新しいスタンバイまたはメンバーが既存の高可用性セットアップに追加された場合の製品インスタンスの動作について説明します。

⁶ Cisco Catalyst スイッチで使用可能な Cisco StackWise Virtual 機能が、このようなセットアップの例です。

⁷ Cisco Catalyst スイッチで使用可能なルートプロセッサ冗長性を備えたクアドスーパーバイザが、このようなセットアップの例です。

承認コードと信頼コードの場合：アクティブな製品インスタンスは、スタンバイおよびメンバーの承認コードと信頼コードを（必要な場合に）要求し、インストールできます。

ポリシーの場合：アクティブな製品インスタンスがスタンバイと同期します。

レポートの場合：アクティブな製品インスタンスのみが使用状況を報告します。アクティブは、高可用性セットアップのすべてのデバイス（スタンバイまたはメンバーを適宜）の使用状況情報を報告します。

スケジュールされたレポートに加えて、次のイベントがレポートをトリガーします。

- スタンバイの追加または削除。RUM レポートには、追加または削除されたスタンバイに関する情報が含まれます。
- スタックマージおよびスタック分割イベントを含む、メンバーの追加または削除。RUM レポートには、追加または削除されたメンバーに関する情報が含まれます。
- スイッチオーバー。
- リロード。

上記のいずれかのイベントが発生すると、**show license status** 特権EXECコマンドの [Next report push] の日付が更新されます。ただし、レポートが製品インスタンスによって送信されるかどうかは、実装されたトポロジと関連するレポート方法で決まります。たとえば、製品インスタンスが切断されているトポロジ（[Transport Type] が [Off]）を実装した場合は、[Next report push] の日付が更新されても、製品インスタンスは RUM レポートを送信しません。

新規メンバーまたはスタンバイ追加の場合：

- CSLU に接続されている製品インスタンスは、それ以上のアクションを実行しません。
- CSSM に直接接続されている製品インスタンスは、信頼の同期を実行します。信頼の同期には、次のものが含まれます。

スタンバイまたはメンバーに信頼コードがまだインストールされていない場合は、信頼コードのインストール。

信頼コードがすでにインストールされている場合は、信頼の同期プロセスにより、新しいスタンバイまたはメンバーがアクティブと同じスマートアカウントおよびバーチャルアカウントにあることが保証されます。そうでない場合、新しいスタンバイまたはメンバーは、アクティブと同じスマートアカウントとバーチャルアカウントに移動されます。

承認コード、ポリシー、および購入情報のインストール（該当する場合）

現在の使用状況情報を含む RUM レポートの送信。

アップグレード

このセクションでは、ポリシーを使用したスマートライセンシングへのアップグレードまたは移行の処理方法について説明します。また、ポリシーを使用したスマートライセンシングが、以前のバージョンのスマートライセンシング、特定のライセンス予約（SLR）、使用権ライセンスリング（RTU）を含む以前のライセンスモデルすべてを処理する方法、および以前のライセ

ンシングモデルの評価ライセンスまたは期限切れライセンスがポリシーを使用したスマートライセンス環境で処理される方法を具体的に説明します。

ポリシーを使用したスマートライセンスに移行するには、ポリシーを使用したスマートライセンスをサポートするソフトウェアバージョンにアップグレードする必要があります。アップグレードした後は、ポリシーを使用したスマートライセンスが唯一のサポートされるライセンスモデルとなり、製品インスタンスはライセンスの変更なしで動作し続けます。[ポリシーを使用したスマートライセンスへの移行 \(152 ページ\)](#) セクションでは、Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチに適用される移行シナリオの詳細と例を示します。

デバイス先行の変換は、ポリシーを使用したスマートライセンスへの移行ではサポートされていません。

アップグレード前に現在のライセンスモデルを識別する

ポリシーを使用したスマートライセンスにアップグレードする前に、製品インスタンスで有効な現在のライセンスモデルを確認するには、特権 EXEC モードで **show license all** コマンドを入力します。このコマンドにより、RTU ライセンスモデルを除くすべてのライセンスモデルに関する情報が表示されます。**show license right-to-use** 特権 EXEC コマンドでは、ライセンスモデルが RTU の場合にのみライセンス情報が表示されます。

アップグレードが既存ライセンスの適用タイプに与える影響

ポリシーを使用したスマートライセンスをサポートするソフトウェアバージョンにアップグレードする場合、既存ライセンスの処理方法は、主に適用タイプによって決まります。

- アップグレード前に使用されていた不適用ライセンスは、アップグレード後も引き続き使用できます。Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチのすべてのライセンスは、不適用ライセンスです。これには、以前のすべてのライセンスモデルのライセンスが含まれます。
 - スマート ライセンス。
 - 特定のライセンス予約 (SLR)。承認コードが付属しています。承認コードは、ポリシーを使用したスマートライセンスへのアップグレード後も引き続き有効であり、既存のライセンスの使用を承認します。
 - 使用権 (RTU) ライセンス。
 - 上記のライセンスモデルのいずれかの評価ライセンスまたは期限切れライセンス。
- アップグレード前に使用されていた適用ライセンスや輸出規制ライセンスは、必要な承認が存在する場合、アップグレード後も引き続き使用できます。

サポートされている Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチのいずれにも、輸出規制ライセンスや適用ライセンスがないため、これらの適用タイプと必要な SLAC は適用されません。

アップグレードが既存ライセンスのレポートに与える影響

アップグレードが既存ライセンスのレポートに与える影響

既存ライセンス	ポリシーを使用したスマートライセンシングへの移行後のレポート要件
使用権 (RTU)	使用されているライセンスによって異なります。 サポートされるトポロジの移行および展開後、 show license usage コマンドの出力で <code>Next ACK deadline</code> フィールドを参照して、レポートが必要かどうか、およびいつ必要かを確認します。
特定のライセンス予約 (SLR)	ライセンス消費に変更がある場合にのみ必要です。 既存の SLR 承認コードは、ポリシーを使用したスマートライセンシングへのアップグレード後に既存のライセンス消費を承認します。
スマートライセンシング (登録済みライセンスと承認済みライセンス) : これらのライセンスのレポートは、ポリシーのレポート要件に基づいています。	ポリシーによって異なります。
評価ライセンスまたは期限切れライセンス	シスコのデフォルトポリシーのレポート要件に基づいています。

アップグレードが既存ライセンスの転送タイプに与える影響

既存の設定で転送タイプが設定されている場合、ポリシーを使用したスマートライセンシングへのアップグレード後も転送タイプが保持されます。

スマートライセンシングの以前のバージョンと比較した場合、ポリシーを使用したスマートライセンシングでは追加の転送タイプを使用できます。デフォルトの転送モードにも変更があります。次の表に、これがアップグレードに与える影響を示します。

アップグレード前の転送タイプ	アップグレード前のライセンスまたはライセンスの状態	アップグレード後の転送タイプ
デフォルト (callhome)	評価	cslu (ポリシーを使用したスマートライセンシングのデフォルト)
	SLR	off
	登録	callhome

アップグレード前の転送タイプ	アップグレード前のライセンスまたはライセンスの状態	アップグレード後の転送タイプ
smart	評価	off
	SLR	off
	登録	smart
N/A たとえば、既存のライセンスモデルが RTU の場合。	N/A たとえば、既存のライセンスモデルが RTU の場合。	cslu

アップグレードがトークン登録プロセスに与える影響

以前のバージョンのスマートライセンシングでは、CSSMへの登録と接続にトークンが使用されていました。ID トークンの登録は、ポリシーを使用したスマートライセンシングでは必要ありません。トークン生成機能はCSSMでも引き続き使用でき、製品インスタンスがCSSMに直接接続されている場合に信頼を確立するために使用されます。「[CSSMに直接接続](#)」を参照してください。

ダウングレード

ダウングレードするには、製品インスタンスのソフトウェアバージョンをダウングレードする必要があります。このセクションでは、新規展開および既存の展開のダウングレードに関する情報を提供します（ポリシーを使用したスマートライセンシングにアップグレードした後にダウングレードする場合）。

新規展開のダウングレード

このセクションは、ポリシーを使用したスマートライセンシングがデフォルトですでに有効になっているソフトウェアバージョンで新しく購入した製品インスタンスがあり、ポリシーを使用したスマートライセンシングがサポートされていないソフトウェアバージョンにダウングレードする場合に該当します。

ダウングレードの結果は、ポリシーを使用したスマートライセンシング環境での操作中に**信頼コード**がインストールされたかどうかによって異なります。ダウングレード先のリリースによっては、さらにアクションが必要になる場合があります。

ポリシーを使用したスマートライセンシング環境で実装したトポロジが「[CSSMに直接接続](#)」である場合、トポロジ実装の一部として信頼コードが必要であるため、信頼コードのインストールが想定または仮定されます。他のトポロジでは、信頼の確立は必須ではありません。そのため、他のトポロジのいずれかを使用する製品インスタンスをダウングレードすると、スマートライセンシング環境で適用される手順に従って、ライセンスを登録済みおよび承認済みの状態に復元する必要があります。以下の[表 10: スマートライセンシングへの新規展開のダウングレードの結果とアクション](#)（134 ページ）を参照してください。

表 10: スマートライセンシングへの新規展開のダウングレードの結果とアクション

ポリシーを使用したスマートライセンシング環境で	以下にダウングレードした場合...	結果と追加のアクション
<p>CSSM に直接接続され、信頼が確立されたスタンドアロン製品インスタンス。</p>	<p>Cisco IOS XE Amsterdam 17.3.1 または Cisco IOS XE Gibraltar 16.12.x の Cisco IOS XE Gibraltar 16.12.4 以降のリリース または Cisco IOS XE Fuji 16.9.x の Cisco IOS XE Fuji 16.9.6 以降のリリース</p>	<p>これ以上の操作は不要です。 製品インスタンスは、ダウングレード後に CSSM からの信頼を更新しようとします。 更新が正常に完了すると、ライセンスは登録済みの状態になり、以前のバージョンのスマートライセンシングが製品インスタンスで有効になります。</p>
	<p>スマートライセンシングをサポートするその他のリリース（上の行に記載されているものを除く）</p>	<p>アクションが必要です。製品インスタンスを再登録する必要があります。 CSSM Web UI で ID トークンを生成し、製品インスタンスで、グローバル コンフィギュレーション モードで license smart register idtoken idtoken コマンドを設定します。</p>
<p>CSSM に直接接続され、信頼が確立された高可用性セットアップ。</p>	<p>スマートライセンシングをサポートするすべてのリリース</p>	<p>アクションが必要です。製品インスタンスを再登録する必要があります。 CSSM Web UI で ID トークンを生成し、製品インスタンスで、グローバル コンフィギュレーション モードで license smart register idtoken idtoken all コマンドを設定します。</p>
<p>その他のトポロジ。（CSLU を介した CSSM への接続、CSLU は CSSM から切断、CSSM への接続なし、CSLU なし）</p>	<p>スマートライセンシングをサポートするすべてのリリース</p>	<p>アクションが必要です。 スマートライセンシング環境で適用される手順に従って、ライセンスを登録済みおよび承認済みの状態に復元します。</p>

アップグレード後のダウングレード

ポリシーを使用したスマートライセンシングをサポートするソフトウェアバージョンにアップグレードした後、以前のライセンシングモデルのいずれかにダウングレードしても、ライセンスの使用は変更されず、製品インスタンスで設定した製品機能は維持されます。ポリシーを使用したスマートライセンシングで使用可能な機能のみが使用できなくなります。以前のライセンシングモデルへの復帰の詳細については、以下の対応するセクションを参照してください。

ポリシーを使用したスマートライセンシングへのアップグレード後のスマートライセンシングへのダウングレード

ダウングレードの結果は、ポリシーを使用したスマートライセンシング環境での操作中に信頼コードがインストールされたかどうかによって異なります。ダウングレード先のリリースによっては、さらにアクションが必要になる場合があります。「表 11: ポリシーを使用したスマートライセンシングへのアップグレード後のスマートライセンシングへのダウングレードの結果とアクション (135 ページ)」を参照してください。

表 11: ポリシーを使用したスマートライセンシングへのアップグレード後のスマートライセンシングへのダウングレードの結果とアクション

ポリシーを使用したスマートライセンシング環境で	以下にダウングレードした場合...	結果と追加のアクション
CSSM に直接接続され、信頼が確立されたスタンドアロン製品インスタンス。	Cisco IOS XE Amsterdam 17.3.1 または Cisco IOS XE Gibraltar 16.12.x の Cisco IOS XE Gibraltar 16.12.4 以降のリリース または Cisco IOS XE Fuji 16.9.x の Cisco IOS XE Fuji 16.9.6 以降のリリース	これ以上の操作は不要です。システムは信頼コードを認識し、元の登録済み ID トークンに変換します。これにより、ライセンスは AUTHORIZED および REGISTERED の状態に戻ります。
	スマートライセンシングをサポートするその他のリリース (上の行に記載されているものを除く)	アクションが必要です。製品インスタンスを再登録する必要があります。 CSSM Web UI で ID トークンを生成し、製品インスタンスで、グローバル コンフィギュレーション モードで license smart register idtokenidtoken コマンドを設定します。

ポリシーを使用したスマートライセンス環境で	以下にダウングレードした場合...	結果と追加のアクション
CSSM に直接接続され、信頼が確立された高可用性セットアップ。	スマートライセンスをサポートするすべてのリリース	アクションが必要です。製品インスタンスを再登録する必要があります。 CSSM Web UI で ID トークンを生成し、製品インスタンスで、グローバル コンフィギュレーション モードで license smart register idtoken idtoken all コマンドを設定します。
その他のトポロジ (CSLU を介した CSSM への接続、CSLU は CSSM から切断、CSSM への接続なし、CSLU なし)	スマートライセンスをサポートするすべてのリリース	アクションが必要です。 スマートライセンス環境で適用される手順に従って、ライセンスを登録済みおよび承認済みの状態に復元します。



(注) スマートライセンス環境で評価状態または期限切れ状態になっていたライセンスは、ダウングレード後に同じ状態に戻ります。

ポリシーを使用したスマートライセンスへのアップグレード後の SLR へのダウングレード

SLR に戻すのに必要な操作は、イメージのダウングレードのみです。ライセンスは予約済みおよび承認済みのままになります。これ以上の操作は必要ありません。

ただし、ポリシーを使用したスマートライセンス環境で SLR に戻した場合は、サポートされているリリースで、必要に応じて SLR を取得するプロセスを繰り返す必要があります。

RTU へのダウングレード

RTU に戻すのに必要な操作は、イメージのダウングレードのみです。

RTU ライセンス環境で評価状態または期限切れ状態であったライセンスは、ダウングレード後に同じ状態に戻ります。

ポリシーを使用したスマートライセンスの設定方法：トポロジ別のワークフロー

このセクションでは、トポロジを実装する最も簡単で迅速な方法について説明します。



- (注) これらのワークフローは、新規展開のみに該当します。既存のライセンスングモデルから移行する場合は、[ポリシーを使用したスマートライセンスへの移行 \(152 ページ\)](#) を参照してください。

トポロジのワークフロー：CSLU を介して CSSM に接続

製品インスタンス開始型通信と CSLU 開始型通信のどちらを実装するかに応じて、対応する一連のタスクを実行します。

- [製品インスタンス開始型通信の場合のタスク](#)
- [CSLU 開始型通信の場合のタスク](#)

製品インスタンス開始型通信の場合のタスク

CSLU のインストール→CSLU の環境設定→製品インスタンスの設定

1. CSLU のインストール

タスクの実行場所：Windows ホスト（ラップトップ、デスクトップ、または仮想マシン（VM））

[Smart Software Manager]<https://software.cisco.com/download/home/286285506/type>> [Smart Licensing Utility] からファイルをダウンロードします。

インストールとセットアップの詳細については、『[Cisco Smart License Utility Quick Start Setup Guide](#)』を参照してください。

2. CSLU の環境設定

タスクの実行場所：CSLU

1. [シスコへのログイン（CSLU インターフェイス） \(175 ページ\)](#)
2. [スマートアカウントとバーチャルアカウントの設定（CSLU インターフェイス） \(176 ページ\)](#)
3. [CSLU での製品開始型製品インスタンスの追加（CSLU インターフェイス） \(176 ページ\)](#)

3. 製品インスタンスの設定

タスクが実行される場所：製品インスタンス

1. [製品インスタンス開始型通信のネットワーク到達可能性の確認 \(177 ページ\)](#)
2. 転送タイプが **cslu** に設定されていることを確認します。

CSLU がデフォルトの転送タイプです。別のオプションを設定した場合は、グローバル コンフィギュレーション モードで **license smart transport cslu** コマンドを入力します。構成ファイルへの変更を保存します。

```
Device(config)# license smart transport cslu
Device(config)# exit
Device# copy running-config startup-config
```

3. CSLU の検出方法を指定します (1 つ選択)

- オプション 1 :

No action required.cslu-local のゼロタッチ DNS ディスカバリ用に設定されたネームサーバ

ここでは、DNS を設定してあり (ネームサーバーの IP アドレスが製品インスタンスで設定されている)、ホスト名 `cslu-local` が CSLU IP アドレスにマッピングされているエントリが DNS サーバーにある場合、それ以上のアクションは不要です。製品インスタンスは、ホスト名 `cslu-local` を自動的に検出します。

- オプション 2 :

No action required.cslu-local.<domain> のゼロタッチ DNS ディスカバリ用に設定されたネームサーバとドメイン

ここでは、DNS を設定してあり (ネームサーバーの IP アドレスとドメインが製品インスタンスで設定されている)、`cslu-local.<domain>` が CSLU IP アドレスにマッピングされているエントリが DNS サーバーにある場合、それ以上のアクションは不要です。製品インスタンスは、ホスト名 `cslu-local` を自動的に検出します。

- オプション 3 :

CSLU に特定の URL を設定します。

グローバル コンフィギュレーション モードで **license smart url cslu**
`http://<cslu_ip_or_host>:8182/cslu/v1/pi` コマンドを入力します。<cslu_ip_or_host> には、CSLU をインストールした Windows ホストのホスト名や IP アドレスを入力します。8182 はポート番号であり、CSLU が使用する唯一のポート番号です。

```
Device(config)# license smart url cslu http://192.168.0.1:8182/cslu/v1/pi
Device(config)# exit
Device# copy running-config startup-config
```

結果 :

製品インスタンスは通信を開始すると、ポリシーに従って、スケジュールされた時刻に最初の RUM レポートを自動的に送信します。この最初のレポートとともに、必要に応じて、信頼コード要求を送信します。製品インスタンスがこの情報をいつ送信するかを確認するには、特権 EXEC モードで **show license all** コマンドを入力し、出力で `Next report push:` フィールドの日付を確認します。

CSLU は、情報を CSSM に転送し、CSSM から返される ACK を製品インスタンスに転送します。

ライセンスの使用状況が変更された場合は、[基本ライセンスまたはアドオンライセンスの設定 \(219 ページ\)](#) を参照しレポートへの影響を確認してください。

CSLU 開始型通信の場合のタスク

CSLU のインストール→CSLU の環境設定→製品インスタンスの設定

1. CSLU のインストール

タスクの実行場所：Windows ホスト（ラップトップ、デスクトップ、または仮想マシン（VM））

[Smart Software Manager]<https://software.cisco.com/download/home/286285506/type>> [Smart Licensing Utility] からファイルをダウンロードします。

インストールとセットアップの詳細については、『[Cisco Smart License Utility Quick Start Setup Guide](#)』を参照してください。

2. CSLU の環境設定

タスクの実行場所：CSLU

1. シスコへのログイン（CSLU インターフェイス）（175 ページ）
2. スマートアカウントとバーチャルアカウントの設定（CSLU インターフェイス）（176 ページ）
3. CSLU での CSLU 開始型製品インスタンスの追加（CSLU インターフェイス）（178 ページ）
4. 使用状況レポートの収集：CSLU 開始（179 ページ）

3. 製品インスタンスの設定

タスクが実行される場所：製品インスタンス

[CSLU 開始型通信のネットワーク到達可能性の確認 \(181 ページ\)](#)

結果：

CSLU から RUM レポートを収集し、CSSM に送信できるようになりました。送信するには、CSLU の [Actions for Selected...] メニューに移動し、[Collect Usage] を選択します。RUM レポートが CSSM に送信されます。この最初のレポートとともに、必要に応じて、CSLU は信頼コード要求を CSSM に送信します。CSSM から ACK を取得し、インストールのために製品インスタンスに送り返します。

ライセンスの使用状況が変更された場合は、[基本ライセンスまたはアドオンライセンスの設定 \(219 ページ\)](#) を参照しレポートへの影響を確認してください。

トポロジのワークフロー：CSSM に直接接続

[Smart Account Set-Up] → [Product Instance Configuration] → [Trust Establishment with CSSM]

1. スマートアカウントのセットアップ

タスクが実行される場所：CSSM Web UI、<https://software.cisco.com/>

スマートアカウントと必要なバーチャルアカウントへの適切なアクセス権を持つユーザーロールがあることを確認します。

2. 製品インスタンスの設定

タスクが実行される場所：製品インスタンス

1. CSSM への製品インスタンス接続の設定：[CSSM への接続の設定](#)（186 ページ）

2. 接続方法と転送タイプの設定（1 つ選択）

• オプション 1：

スマート転送：転送タイプを **smart** に設定し、対応する URL を設定します。

転送モードが **license smart transport smart** に設定されている場合は、**license smart url default** を設定すると、スマート URL

(<https://smartreceiver.cisco.com/licservice/license>) が自動的に設定されます。構成ファイルへの変更を保存します。

```
Device(config)# license smart transport smart
Device(config)# license smart url default
Device(config)# exit
Device# copy running-config startup-config
```

• オプション 2：

HTTPS プロキシを介してスマートトランスポートを設定します。[HTTPS プロキシを介したスマート転送の設定](#)（189 ページ）を参照してください

• オプション 3：

ダイレクトクラウドアクセス用に Call Home サービスを設定します。「[ダイレクトクラウドアクセス用の Call Home サービスの設定](#)（190 ページ）」を参照してください。

• オプション 4：

HTTPS プロキシを介したダイレクトクラウドアクセス用に Call Home サービスを設定します。「[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](#)（193 ページ）」を参照してください。

3. CSSM との信頼の確立

タスクが実行される場所：CSSM Web UI、次に製品インスタンス

1. 所有するバーチャルアカウントごとに 1 つのトークンを生成します。1 つのバーチャルアカウントに属するすべての製品インスタンスに同じトークンを使用できます ([CSSM からの信頼コード用新規トークンの生成](#)（210 ページ）)。

2. トークンをダウンロードしたら、製品インスタンスに信頼コードをインストールできます ([信頼コードのインストール](#)（211 ページ）)。

結果：

信頼を確立した後、CSSMはポリシーを返します。ポリシーは、そのバーチャルアカウントのすべての製品インスタンスに自動的にインストールされます。ポリシーは、製品インスタンスが使用状況をレポートするかどうか、およびその頻度を指定します。

より頻繁にレポートを作成するようにレポート間隔を変更する場合は、製品インスタンスで、グローバル コンフィギュレーション モードで **license smart usage interval** コマンドを設定します。シンタックスの詳細については、対応するリリースのコマンドリファレンスで *license smart (privileged EXEC)* コマンドを参照してください。

ライセンスの使用状況が変更された場合は、[基本ライセンスまたはアドオンライセンスの設定 \(219 ページ\)](#) を参照しレポートへの影響を確認してください。

トポロジのワークフロー：コントローラを介して CSSM に接続

コントローラとして Cisco DNA Center を展開するには、次のワークフローを実行します。

製品インスタンスの設定 → Cisco DNA Center の設定

1. 製品インスタンスの設定

タスクの実行場所：製品インスタンス

NETCONF を有効にします。Cisco DNA Center は NETCONF プロトコルを使用して設定をプロビジョニングし、製品インスタンスから必要な情報を取得します。したがって、これを容易にするために製品インスタンスで NETCONF を有効にする必要があります。

詳細については、『[Programmability Configuration Guide, Cisco IOS XE Amsterdam 17.3.x](#)』を参照してください。このガイドの「Model-Driven Programmability」の「NETCONF Protocol」を確認します。

2. Cisco DNA Center の設定

タスクの実行場所：Cisco DNA Center GUI

次に、実行する必要があるタスクの概要と、付属のドキュメントリファレンスを示します。このドキュメントには、Cisco DNA Center GUI で実行する必要がある詳細な手順が示されています。

1. スマートアカウントとバーチャルアカウントを設定します。

CSSM Web UI へのログインに使用するのと同じログインクレデンシャルを入力します。これにより、Cisco DNA Center は CSSM との接続を確立できます。

必要なリリース（リリース 2.2.2 以降）の『[Cisco DNA Center Administrator Guide](#)』[英語]の「Manage Licenses」の「Set Up License Manager」を参照してください。

2. 必要な製品インスタンスを Cisco DNA Center インベントリに追加してサイトに割り当てます。

これにより、Cisco DNA Center は、要求されている証明書を含む必要な設定をプッシュして、Smart Licensing Using Policy が予想どおりに機能するようにします。

必要なリリース（リリース 2.2.2 以降）の『[Cisco DNA Center User Guide](#)』[英語]の「Display Your Network Topology」の「Assign Devices to a Site」を参照してください。

結果：

トポロジを実装したら、Cisco DNA Center で最初のアドホックレポートをトリガーし、スマートアカウントとバーチャルアカウント、および製品インスタンス間のマッピングを確立する必要があります。必要なリリース（リリース 2.2.2 以降）の『[Cisco DNA Center Administrator Guide](#)』[英語]で「Manage Licenses」の「Upload Resource Utilization Details to CSSM」を参照してください。これが完了すると、Cisco DNA Center はレポートポリシーに基づいて後続のレポートを処理します。

複数のポリシーが使用可能な場合、Cisco DNA Center は最も短いレポート間隔を維持します。この間隔はより頻繁に（より短い間隔で）報告するようにのみ変更できます。必要なリリース（リリース 2.2.2 以降）の『[Cisco DNA Center Administrator Guide](#)』[英語]の「Manage Licenses」の「Modify License Policy」を参照してください。

この後にライセンスレベルを変更する場合は、必要なリリース（リリース 2.2.2 以降）の『[Cisco DNA Center Administrator Guide](#)』[英語]の「Manage Licenses」の「Change License Level」を参照してください。

トポロジのワークフロー：CSLU は CSSM から切断

製品インスタンス開始型通信または CSLU 開始型通信のどちらの方法を実装するかによって異なります。以下の対応するタスク一覧を実行します。

- [製品インスタンス開始型通信の場合のタスク](#)
- [CSLU 開始型通信の場合のタスク](#)

製品インスタンス開始型通信の場合のタスク

CSLU のインストール → CSLU の環境設定 → 製品インスタンスの設定 → [Download All for Cisco] と [Upload From Cisco]

1. CSLU のインストール

タスクの実行場所：Windows ホスト（ラップトップ、デスクトップ、または仮想マシン（VM））

[Smart Software Manager]<https://software.cisco.com/download/home/286285506/type>> [Smart Licensing Utility] からファイルをダウンロードします。

インストールとセットアップの詳細については、『[Cisco Smart License Utility Quick Start Setup Guide](#)』を参照してください。

2. CSLU の環境設定

タスクの実行場所：CSLU

1. CSLU の [Preferences] タブで、[Cisco Connectivity] トグルスイッチをオフにします。フィールドが「Cisco Is Not Available」に切り替わります。
2. スマートアカウントとバーチャルアカウントの設定 (CSLU インターフェイス) (176 ページ)
3. CSLU での製品開始型製品インスタンスの追加 (CSLU インターフェイス) (176 ページ)

3. 製品インスタンスの設定

タスクが実行される場所：製品インスタンス

1. 製品インスタンス開始型通信のネットワーク到達可能性の確認 (177 ページ)
2. 転送タイプが **cslu** に設定されていることを確認します。

CSLU がデフォルトの転送タイプです。別のオプションを設定した場合は、グローバル コンフィギュレーション モードで **license smart transport cslu** コマンドを入力します。構成ファイルへの変更を保存します。

```
Device(config)# license smart transport cslu
Device(config)# exit
Device# copy running-config startup-config
```

3. CSLU の検出方法を指定します (1 つ選択)

- オプション 1 :

No action required.cslu-local のゼロタッチ DNS ディスカバリ用に設定されたネームサーバ

ここでは、DNS を設定してあり (ネームサーバーの IP アドレスが製品インスタンスで設定されている)、ホスト名 **cslu-local** が CSLU IP アドレスにマッピングされているエントリが DNS サーバーにある場合、それ以上のアクションは不要です。製品インスタンスは、ホスト名 **cslu-local** を自動的に検出します。

- オプション 2 :

No action required.cslu-local.<domain> のゼロタッチ DNS ディスカバリ用に設定されたネームサーバとドメイン

ここでは、DNS を設定してあり (ネームサーバーの IP アドレスとドメインが製品インスタンスで設定されている)、**cslu-local.<domain>** が CSLU IP アドレスにマッピングされているエントリが DNS サーバーにある場合、それ以上のアクションは不要です。製品インスタンスは、ホスト名 **cslu-local** を自動的に検出します。

- オプション 3 :

CSLU に特定の URL を設定します。

グローバル コンフィギュレーション モードで **license smart url cslu**

http://<cslu_ip_or_host>:8182/cslu/v1/pi コマンドを入力します。<cslu_ip_or_host> には、CSLU をインストールした Windows ホストのホスト名や IP アドレスを入力します。8182 はポート番号であり、CSLU が使用する唯一のポート番号です。

```
Device(config)# license smart url cslu http://192.168.0.1:8182/cslu/v1/pi
Device(config)# exit
Device# copy running-config startup-config
```

4. [Download All for Cisco] と [Upload From Cisco]

タスクの実行場所：CSLU と CSSM

1. [Download All For Cisco \(CSLU インターフェイス\)](#) (180 ページ)
2. [CSSM への使用状況データのアップロードと ACK のダウンロード](#) (213 ページ)
3. [Upload From Cisco \(CSLU インターフェイス\)](#) (181 ページ)

結果：

製品インスタンスは通信を開始すると、ポリシーに従って、スケジュールされた時刻に最初の RUM レポートを自動的に送信します。この最初のレポートとともに、必要に応じて、信頼コード要求を送信します。製品インスタンスがこの情報をいつ送信するかを確認するには、特権 EXEC モードで **show license all** コマンドを入力し、出力で `Next report push:` フィールドの日付を確認します。

CSLU は CSSM から切断されるため、CSLU が製品インスタンスから収集した使用状況データをファイルに保存する必要があります。次に、シスコに接続されているワークステーションからファイルを CSSM にアップロードします。その後、CSSM から ACK をダウンロードします。CSLU がインストールされて製品インスタンスに接続されているワークステーションで、ファイルを CSLU にアップロードします。

ライセンスの使用状況が変更された場合は、[基本ライセンスまたはアドオンライセンスの設定 \(219 ページ\)](#) を参照しレポートへの影響を確認してください。

CSLU 開始型通信の場合のタスク

CSLU のインストール→CSLU の環境設定→製品インスタンスの設定→[Download All for Cisco] と [Upload From Cisco]

1. CSLU のインストール

タスクの実行場所：Windows ホスト（ラップトップ、デスクトップ、または仮想マシン (VM))

[Smart Software Manager]<https://software.cisco.com/download/home/286285506/type>> [Smart Licensing Utility] からファイルをダウンロードします。

インストールとセットアップの詳細については、『[Cisco Smart License Utility Quick Start Setup Guide](#)』を参照してください。

2. CSLU の環境設定

タスクの実行場所：CSLU

1. CSLU の [Preferences] タブで、[Cisco Connectivity] トグルスイッチをオフにします。フィールドが「Cisco Is Not Available」に切り替わります。
 2. [スマートアカウントとバーチャルアカウントの設定 \(CSLU インターフェイス\)](#) (176 ページ)
 3. [CSLU での CSLU 開始型製品インスタンスの追加 \(CSLU インターフェイス\)](#) (178 ページ)
 4. [使用状況レポートの収集：CSLU 開始](#) (179 ページ)
3. **製品インスタンスの設定**
- タスクの実行場所：製品インスタンス
- [CSLU 開始型通信のネットワーク到達可能性の確認](#) (181 ページ)
4. **[Download All for Cisco] と [Upload From Cisco]**
- タスクの実行場所：CSLU と CSSM
1. [Download All For Cisco \(CSLU インターフェイス\)](#) (180 ページ)
 2. [CSSM への使用状況データのアップロードと ACK のダウンロード](#) (213 ページ)
 3. [Upload From Cisco \(CSLU インターフェイス\)](#) (181 ページ)

結果：

CSLU から RUM レポートを収集し、CSSM に送信できるようになりました。それには、[Actions for Selected] メニューに移動し、[Collect Usage] を選択します。該当する場合、レポートには信頼コード要求と承認コード要求も含まれます。

CSLU は CSSM から切断されるため、CSLU が製品インスタンスから収集した使用状況データをファイルに保存する必要があります。次に、シスコに接続されているワークステーションからファイルを CSSM にアップロードします。その後、CSSM から ACK をダウンロードします。CSLU がインストールされて製品インスタンスに接続されているワークステーションで、ファイルを CSLU にアップロードします。

ライセンスの使用状況が変更された場合は、[基本ライセンスまたはアドオンライセンスの設定 \(219 ページ\)](#) を参照しレポートへの影響を確認してください。

トポロジのワークフロー：CSSM への接続なし、CSLU なし

他のコンポーネントへの接続を設定する必要がないため、トポロジの設定に必要なタスクのリストは短くなります。このトポロジを実装した後に必要な使用状況レポートを作成する方法については、ワークフローの最後にある「結果」セクションを参照してください。

製品インスタンスの設定

タスクの実行場所：製品インスタンス

転送タイプをオフに設定します。

グローバル コンフィギュレーション モードで **license smart transport off** コマンドを入力します。構成ファイルへの変更を保存します。

```
Device(config)# license smart transport off
Device(config)# exit
Device# copy running-config startup-config
```

結果：

製品インスタンスからのすべての通信を無効にします。ライセンスの使用状況をレポートするには、RUM レポートを（製品インスタンスの）ファイルに保存してから、CSSM にアップロードする必要があります（インターネットとシスコに接続されているワークステーションからアップロード）。

1. RUM レポートの生成と保存

license smart save usage コマンドは特権 EXEC モードで入力します。次の例では、すべての RUM レポートがファイル `all_rum.txt` で製品インスタンスのフラッシュメモリに保存されます。この例では、ファイルはまずブートフラッシュに保存され、次に TFTP の場所にコピーされます。

```
Device# license smart save usage all bootflash:all_rum.txt
Device# copy bootflash:all_rum.txt tftp://10.8.0.6/all_rum.txt
```

2. 使用状況データを CSSM にアップロード：[CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#)
3. ACK を製品インスタンスにインストール：[製品インスタンスへのファイルのインストール \(214 ページ\)](#)

ライセンスの使用状況が変更された場合は、[基本ライセンスまたはアドオンライセンスの設定 \(219 ページ\)](#) を参照しレポートへの影響を確認してください。

トポロジのワークフロー：SSM オンプレミス展開

製品インスタンス開始型通信（プッシュ）を実装するか、または SSM オンプレミス開始型通信（プル）を実装するかによって、対応するタスクの手順を実行します。

製品インスタンス開始型通信の場合のタスク

SSM オンプレミスのインストール → 製品インスタンスの追加と検証（該当する場合のみ） → 製品インスタンスの設定 → 使用状況の最初の同期

1. SSM オンプレミスのインストール

タスクの実行場所：Cisco UCS C220 M3 ラックサーバなどの物理サーバ、または必要な要件を満たしているハードウェアベースのサーバ。

[Smart Software Manager](#) の [Smart Software Manager On-Prem] からファイルをダウンロードします。

インストールのヘルプについては、『[Cisco Smart Software On-Prem Installation Guide](#)』と『[Cisco Smart Software On-Prem User Guide](#)』を参照してください。

SSM オンプレミスを展開し、SSM オンプレミスで共通名を設定し ([Security Widgets] > [Certificates])、NTP サーバを同期し ([Settings] ウィジェット > [Time Settings])、SSM オンプレミスアカウントを作成して登録し、CSSM のスマートアカウントとバーチャルアカウントと同期 ([Synchronization] ウィジェット) したら、インストールが完了します。



- (注) [On-Prem Licensing Workspace] のライセンス機能は、ローカルアカウントを作成し、登録し、CSSM のスマートアカウントと同期するまではグレー表示になります。CSSM とのローカルアカウントの同期は、SSM オンプレミスインスタンスを CSSM に認識させるためであり、次に示す「4. 使用状況の最初の同期」で実行する使用状況の同期とは異なります。

2. 製品インスタンスの追加と検証

タスクの実行場所：SSM オンプレミス UI

この手順により、製品インスタンスが検証され、CSSM の該当するスマートアカウントとバーチャルアカウントにマッピングされます。この手順は、次の場合にのみ必要です。

- 製品インスタンスを CSSM で報告する前に、SSM オンプレミスで追加および検証する場合（セキュリティを強化するため）。
 - (デフォルトのローカルバーチャルアカウントに加えて) ローカルバーチャルアカウントを SSM オンプレミスで作成した場合。この場合は、SSM オンプレミスが CSSM の正しいライセンスプールに使用状況を報告できるように、SSM オンプレミスにこれらのローカルバーチャルアカウントの製品インスタンスのスマートアカウント情報とバーチャルアカウント情報を提供する必要があります。
1. [スマートアカウントとバーチャルアカウントの割り当て \(SSM オンプレミス UI\)](#) (194 ページ)
 2. [デバイスの検証 \(SSM オンプレミス UI\)](#) (195 ページ)



- (注) 製品インスタンスが NAT 設定にある場合は、デバイス検証を有効にするときに NAT 設定のサポートも有効にします。両方のトグルスイッチが同じウィンドウにあります。

3. 製品インスタンスの設定

タスクの実行場所：製品インスタンスと SSM オンプレミス UI

特権 EXEC モードで **copy running-config startup-config** コマンドを入力して、製品インスタンスの設定変更を必ず保存してください。

1. [製品インスタンス開始型通信のネットワーク到達可能性の確認](#) (196 ページ)
2. [トランスポート URL の取得 \(SSM オンプレミス UI\)](#) (198 ページ)
3. [転送タイプ、URL、およびレポート間隔の設定](#) (215 ページ)

CSLU と SSM オンプレミスのトランスポートタイプ設定は同じですが（グローバルコンフィギュレーションモードの **license smart transport cslu** コマンド）、URL が異なります。

4. 使用状況の最初の同期

タスクの実行場所：製品インスタンス、SSM オンプレミス UI、CSSM

1. 製品インスタンスを SSM オンプレミスと同期します。

製品インスタンスに **license smart sync {all | local}** コマンドを特権 EXEC モードで入力します。これにより、SSM オンプレミスと製品インスタンスが同期され、保留中のデータが送受信されます。

```
Device(config)# license smart sync local
```

これは、SSM オンプレミス UI で確認できます。ログインして、[Smart Licensing] ワークスペースを選択します。[Inventory] > [SL Using Policy] タブに移動します。対応する製品インスタンスの [Alerts] 列に、「Usage report from product instance」というメッセージが表示されます。



(注) 上記の手順 2（製品インスタンスの追加と検証）を実行していない場合、このサブ手順を実行すると、製品インスタンスが SSM オンプレミスのデータベースに追加されます。

2. 使用状況情報を CSSM と同期します（いずれかを選択）。

• オプション 1 :

SSM オンプレミスが CSSM に接続されている場合：SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports] > [Usage Schedules] > [Synchronize now with Cisco] に移動します。

• オプション 2 :

SSM オンプレミスが CSSM に接続されていません。[使用状況データのエクスポートとインポート \(SSM オンプレミス UI\) \(199 ページ\)](#) を参照してください。

結果 :

使用状況の最初の同期が完了しました。製品インスタンスとライセンス使用状況情報が SSM オンプレミスに表示されるようになりました。

後続のレポートには、次のオプションが含まれています。

- 製品インスタンスと SSM オンプレミスとの間でデータを同期するには、次の手順を実行します。
 - レポート間隔を設定して、製品インスタンスと SSM オンプレミスとの間の定期的な同期をスケジュールします。グローバル コンフィギュレーションモードで **license smart usage interval interval_in_days** コマンドを入力します。

製品インスタンスが次にいつRUMレポートを送信するかを確認するには、特権EXECモードで **show license all** コマンドを入力し、出力の [Next report push:] フィールドを確認します。

- 製品インスタンスと SSM オンプレミスとの間でアドホックまたはオンデマンドの同期を行うには、**license smart sync** 特権 EXEC コマンドを入力します。
- 使用状況情報を CSSM と同期するには、次の手順を実行します。
 - CSSM との定期的な同期をスケジュールします。SSM オンプレミス UI で、[Reports] > [Usage Schedules] > [Synchronization schedule with Cisco] に移動します。次の頻度情報を入力し、保存します。
 - [Days] : 同期が実行される頻度を示します。たとえば、2 を入力すると、同期は 2 日に 1 回行われます。
 - [Time of Day] : 24 時間表記法で、同期が実行される時刻を示します。たとえば、14 hours と 0 minutes を入力すると、ローカルタイムゾーンの午後 2 時 (1400) に同期が行われます。
 - レポートに必要なファイルのアップロードとダウンロードを実行します (使用状況データのエクスポートとインポート (SSM オンプレミス UI) (199 ページ))。

SSM オンプレミスインスタンス開始型通信の場合のタスク

SSM オンプレミスのインストール → 製品インスタンスの追加 → 製品インスタンスの設定 → 使用状況の最初の同期

1. SSM オンプレミスのインストール

タスクの実行場所 : Cisco UCS C220 M3 ラックサーバなどの物理サーバ、または必要な要件を満たしているハードウェアベースのサーバ。

[Smart Software Manager](#) の [Smart Software Manager On-Prem] からファイルをダウンロードします。

インストールのヘルプについては、『[Cisco Smart Software On-Prem Installation Guide](#)』と『[Cisco Smart Software On-Prem User Guide](#)』を参照してください。

SSM オンプレミスを展開し、SSM オンプレミスで共通名を設定し ([Security Widgets] > [Certificates])、NTP サーバを同期し ([Settings] ウィジェット > [Time Settings])、SSM オンプレミスアカウントを作成して登録し、CSSM のスマートアカウントとバーチャルアカウントと同期 ([Synchronization] ウィジェット) したら、インストールが完了します。



- (注) [On-Prem Licensing Workspace] のライセンス機能は、ローカルアカウントを作成し、登録し、CSSM のスマートアカウントと同期するまではグレー表示になります。CSSM とのローカルアカウントの同期は、SSM オンプレミスインスタンスを CSSM に認識させるためであり、次に示す「4. 使用状況の最初の同期」で実行する使用状況の同期とは異なります。

2. 製品インスタンスの追加

タスクの実行場所：SSM オンプレミス UI

単一の製品インスタンスを追加するか、または複数の製品インスタンスを追加するかに応じて、対応するサブ手順（[1 つ以上の製品インスタンスの追加（SSM オンプレミス UI）（200 ページ）](#)）を実行します。

3. 製品インスタンスの設定

タスクが実行される場所：製品インスタンス

特権 EXEC モードで **copy running-config startup-config** コマンドを入力して、製品インスタンスの設定変更を必ず保存してください。[SSM オンプレミス開始型通信のネットワーク到達可能性の確保（201 ページ）](#)

4. 使用状況の最初の同期

タスクの実行場所：SSM オンプレミスと CSSM

1. 製品インスタンスから使用状況情報を取得します。

SSM オンプレミス UI で、[Reports] > [Synchronization pull schedule] > [Synchronize now with the device] に移動します。

[Alerts] 列に、「Usage report from product instance」というメッセージが表示されます。



ヒント 同期がトリガーされるまでに 60 秒かかります。進行状況を表示するには、[On-Prem Admin Workspace] に移動し、[Support Center] ウィジェットをクリックします。このウィジェットにシステムログに進行状況が表示されます。

2. 使用状況情報を CSSM と同期します（いずれかを選択）。

• オプション 1 :

SSM オンプレミスが CSSM に接続されている場合：SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports] > [Usage Schedules] > [Synchronize now with Cisco] に移動します。

• オプション 2 :

SSM オンプレミスが CSSM に接続されていません。[使用状況データのエクスポートとインポート（SSM オンプレミス UI）（199 ページ）](#) を参照してください。

結果 :

使用状況の最初の同期が完了しました。製品インスタンスとライセンス使用状況情報が SSM オンプレミスに表示されるようになりました。SSM オンプレミスは ACK を製品インスタンスに自動的に返します。製品インスタンスが ACK を受信していることを確認するには、特権 EXEC モードで **show license status** コマンドを入力し、出力で [Last ACK received] フィールドの日付を確認します。

後続のレポートには、次のオプションが含まれています。

- 製品インスタンスから使用状況情報を取得するには、次の手順を実行します。
 - SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports] > [Usage Schedules] > [Synchronize now with Cisco] に移動します。
 - 頻度を設定して、製品インスタンスから情報を定期的に取り得るようにスケジュールします。SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports] > [Usage Schedules] > [Synchronisation pull schedule with the devices] に移動します。次のフィールドに値を入力します。
 - [Days] : 同期が実行される頻度を示します。たとえば、2 を入力すると、同期は 2 日に 1 回行われます。
 - [Time of Day] : 24 時間表記法で、同期が実行される時刻を示します。たとえば、14 hours と 0 minutes と入力すると、午後 2 時 (1400) に同期が行われます。
 - CSSM に接続せずに製品インスタンスから使用状況データを収集します。SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Inventory] > [SL Using Policy] タブに移動します。対応するチェックボックスを有効にして、1 つ以上の製品インスタンスを選択します。[Actions for Selected...] > [Collect Usage] をクリックします。選択した製品インスタンスにオンプレミスが接続し、使用状況レポートを収集します。その後、これらの使用状況レポートはオンプレミスのローカルライブラリに保存されます。これらのレポートは、オンプレミスがシスコに接続されている場合はシスコに転送できます。また、(シスコに接続されていない場合は) [Export/Import All.] > [Export Usage to Cisco] を選択することで、使用状況の収集を手動でトリガーできます。
- 使用状況情報を CSSM と同期するには、次の手順を実行します。
 - CSSM との定期的な同期をスケジュールします。SSM オンプレミス UI で、[Reports] > [Usage Schedules] > [Synchronization schedule with Cisco] に移動します。次の頻度情報を入力し、保存します。
 - [Days] : 同期が実行される頻度を示します。たとえば、2 を入力すると、同期は 2 日に 1 回行われます。
 - [Time of Day] : 24 時間表記法で、同期が実行される時刻を示します。たとえば、14 hours と 0 minutes と入力すると、午後 2 時 (1400) に同期が行われます。
 - レポートに必要なファイルのアップロードとダウンロードを実行します (使用状況データのエクスポートとインポート (SSM オンプレミス UI) (199 ページ)) 。

ポリシーを使用したスマートライセンシングへの移行

ポリシーを使用したスマートライセンシングにアップグレードするには、製品インスタンスのソフトウェアバージョン（イメージ）をサポートされているバージョンにアップグレードする必要があります。

はじめる前に

ポリシーを使用したスマートライセンシングによって以前の全ライセンスモデルのさまざまな側面がどのように処理されるかを理解するため、[アップグレード（130ページ）](#)のセクションを必ずお読みください。

ポリシーを使用したスマートライセンシングは、Cisco IOS XE Amsterdam 17.3.2 で導入されました。そのため、これがポリシーを使用したスマートライセンシングに最低限必要なバージョンになります。

移行前に使用していたすべてのライセンスは、アップグレード後も使用できることに注意してください。つまり、登録済みライセンスと承認済みライセンス（予約済みライセンスを含む）だけでなく、評価ライセンスもすべて移行されます。登録済みライセンスと承認済みライセンスを移行する利点は、アップグレード後も設定（トランスポートタイプの設定と、CSSMへの接続の設定、すべての証人コード）が保持されるため、移行後に実行する設定手順が少なくなります。これにより、Smart Licensing Using Policy 環境への移行がよりスムーズになります。

デバイス先行の変換は、ポリシーを使用したスマートライセンシングへの移行ではサポートされていません。

スイッチ ソフトウェアのアップグレード

アップグレードの手順については、対応するリリースノートを参照してください。一般的なリリース固有の考慮事項がある場合は、対応するリリースノートに記載されています。たとえば、Cisco IOS XE Amsterdam 17.3.2 にアップグレードするには、『*Release Notes for Cisco <プラットフォーム名>, Cisco IOS XE Amsterdam 17.3.x*』を参照してください。

この手順を使用して、インストールモードで、または [In-Service Software Upgrade \(ISSU\)](#) を使用してアップグレードできます（サポートされているプラットフォームおよびサポートされているリリースで実行）。

Release Notes for Cisco Catalyst 9300 シリーズ スイッチ：<https://www.cisco.com/c/en/us/support/switches/catalyst-9300-series-switches/products-release-notes-list.html>。「スイッチソフトウェアのアップグレード」を参照してください。ISSU は、この製品インスタンスではサポートされていません。

ソフトウェアバージョンのアップグレード後

- トポロジを実装します。

アップグレード前の設定でトランスポートモードを使用できる場合は、アップグレード後も保持されます。評価ライセンスや、トランスポートタイプの概念が存在しないライセン

スモデルの場合など、一部の場合にのみ、デフォルト (cslu) が適用されます。このような場合は、Smart Licensing Using Policy 環境で動作するように設定する前に実行する必要がある手順がいくつかある場合があります。

アップグレード元のライセンスモデルに関係なく、アップグレード後にトポロジを変更できます。

- ライセンスの使用状況と CSSM の同期

どのライセンスモデルからアップグレードするか、どのトポロジを実装するかに関係なく、使用状況情報を CSSM と同期します。そのためには、実装するトポロジに適用されるレポート方式に従う必要があります。この最初の同期により、使用状況の最新の情報が CSSM に反映され、カスタムポリシー（使用可能な場合）が適用されます。この同期後に適用されるポリシーは、後続のレポート要件も示します。これらのルールを [アップグレードが既存ライセンスのレポートに与える影響（132 ページ）](#) の表にも示します。



(注) 使用状況の最初の同期が完了した後、ポリシー、またはシステムメッセージに示されている場合にのみ、レポートが必要です。

移行シナリオの例

さまざまな既存のライセンスモデルとライセンスを考慮した移行シナリオの例を示します。すべてのシナリオで、移行前と後の出力例と注意すべき CSSM Web UI の変更を（移行の成功または追加アクションのインジケータとして）示し、また、必要な移行後の手順を特定して実行する方法も示します。



(注) SSM オンプレミスでは、アップグレード関連のさまざまなアクティビティを実行する順序が重要です。したがって、このシナリオでのみ、例ではなく、移行の順序が示されています。

例：スマートライセンシングからポリシーを使用したスマートライセンシングへ

次に、スマートライセンシングからポリシーを使用したスマートライセンシングに移行する Cisco Catalyst 9500 スイッチの例を示します。これはアクティブとスタンバイを含む高可用性セットアップの例です。

- [表 12: スマートライセンシングからポリシーを使用したスマートライセンシングへ：show コマンド](#)
- [移行後の CSSM Web UI（157 ページ）](#)
- [移行後のレポート（158 ページ）](#)

例：スマートライセンシングからポリシーを使用したスマートライセンシングへ

show コマンドは、移行の前後に確認すべき以下の重要なフィールドを抽出して出力します。

表 12: スマートライセンシングからポリシーを使用したスマートライセンシングへ： **show** コマンド

アップグレード前	アップグレード後
<p>show license summary (スマートライセンシング)</p> <p>Statusフィールドと License Authorization フィールドに、ライセンスについて REGISTERED および AUTHORIZED と表示されます。</p> <p>Device# show license summary</p> <pre>Smart Licensing is ENABLED Registration: Status: REGISTERED Smart Account: SA-Eg-Company-01 Virtual Account: SLE_Test Export-Controlled Functionality: ALLOWED Last Renewal Attempt: None Next Renewal Attempt: Mar 21 11:08:58 2021 PST License Authorization: Status: AUTHORIZED Last Communication Attempt: SUCCEEDED Next Communication Attempt: Oct 22 11:09:07 2020 PST License Usage: License Entitlement tag Count Status ----- C9500 Network Advantage (C9500 Network Advantage) 2 AUTHORIZED C9500-DNA-16X-A (C9500-16X DNA Advantage) 2 AUTHORIZED</pre>	<p>show license summary (ポリシーを使用したスマートライセンシング)</p> <p>Statusフィールドに、ライセンスについて、登録済みおよび承認済みではなく IN USE と表示されます。</p> <p>Device# show license summary</p> <pre>License Usage: License Entitlement tag Count Status ----- network-advantage (C9500 Network Advantage) 2 IN USE dna-advantage (C9500-16X DNA Advantage) 2 IN USE</pre>
<p>show license usage (スマートライセンシング)</p>	<p>show license usage (ポリシーを使用したスマートライセンシング)</p> <p>ライセンス数は変わりません。</p> <p>Enforcement Type フィールドに NOT ENFORCED と表示され (Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチには、輸出規制ライセンスや適用ライセンスはありません)</p>

```
Device# show license usage
License Authorization:
Status: AUTHORIZED on Sep 22 11:09:07 2020 PST
C9500 Network Advantage (C9500 Network Advantage):
Description: C9500 Network Advantage
Count: 2
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
C9500-DNA-16X-A (C9500-16X DNA Advantage):
Description: C9500-DNA-16X-A
Count: 2
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
```

```
Device# show license usage
License Authorization:
Status: Not Applicable
network-advantage (C9500 Network Advantage):
Description: network-advantage
Count: 2
Version: 1.0
Status: IN USE
Export status: NOT RESTRICTED
Feature Name: network-advantage
Feature Description: network-advantage
Enforcement type: NOT ENFORCED
License type: Perpetual
dna-advantage (C9500-16X DNA Advantage):
Description: C9500-16X DNA Advantage
Count: 2 Version: 1.0
Status: IN USE
Export status: NOT RESTRICTED
Feature Name: dna-advantage
Feature Description: C9500-16X DNA Advantage
Enforcement type: NOT ENFORCED
License type: Subscription
```

show license status (スマートライセンシング)

show license status (ポリシーを使用したスマートライセンシング)

Transport: フィールド：特定の転送タイプが設定されたため、アップグレード後もその設定が保持されます。

Policy: ヘッダーと詳細：スマートアカウントまたはバーチャルアカウントでカスタムポリシーを使用できます。これは製品インスタンスにも自動的にインストールされます。(信頼を確立した後、CSSMはポリシーを返します。その後、このポリシーが自動的にインストールされます)。

Usage Reporting: ヘッダー：Next report push: フィールドには、製品インスタンスが次のRUMレポートをCSSMに送信するタイミングについての情報が表示されます。

Trust Code Installed: フィールド：ID トークンが正常に変換され、信頼できる接続がCSSMで確立されたことを示します。

例: スマートライセンシングからポリシーを使用したスマートライセンシングへ

```

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: Eg-SA-01
Virtual Account: Eg-VA-01
Export-Controlled Functionality: ALLOWED
Initial Registration: SUCCEEDED on Sep 22 11:08:58 2020 PST
Last Renewal Attempt: None
Next Renewal Attempt: Mar 21 11:08:57 2021 PST
Registration Expires: Sep 22 11:04:23 2021 PST
License Authorization:
Status: AUTHORIZED on Sep 22 11:09:07 2020 PST
Last Communication Attempt: SUCCEEDED on Sep 22 11:09:07 2020
PST
Next Communication Attempt: Oct 22 11:09:06 2020 PST
Communication Deadline: Dec 21 11:04:34 2020 PST
Export Authorization Key:
Features Authorized:
<none>
Miscellaneous:
Custom Id: <empty>

```

```

Device# show license status

Utility:
Status: DISABLED
Smart Licensing Using Policy:
Status: ENABLED
Data Privacy:
Sending Hostname: yes
Callhome hostname privacy: DISABLED
Smart Licensing hostname privacy: DISABLED

Version privacy: DISABLED
Transport:
Type: Callhome
Policy:
Policy in use: Merged from multiple sources.
Reporting ACK required: yes (CISCO default)
Unenforced/Non-Export Perpetual Attributes:
First report requirement (days): 365 (CISCO
default)
Reporting frequency (days): 0 (CISCO
default)
Report on change (days): 90 (CISCO default)

Unenforced/Non-Export Subscription
Attributes:
First report requirement (days): 90 (CISCO
default)
Reporting frequency (days): 90 (CISCO
default)
Report on change (days): 90 (CISCO default)

Enforced (Perpetual/Subscription) License
Attributes:
First report requirement (days): 0 (CISCO
default)
Reporting frequency (days): 0 (CISCO
default)
Report on change (days): 0 (CISCO default)

Export (Perpetual/Subscription) License
Attributes:
First report requirement (days): 0 (CISCO
default)
Reporting frequency (days): 0 (CISCO
default)
Report on change (days): 0 (CISCO default)
Miscellaneous:
Custom Id: <empty>
Usage Reporting:
Last ACK received: Sep 22 13:49:38 2020 PST
Next ACK deadline: Dec 21 12:02:21 2020 PST
Reporting push interval: 30 days
Next ACK push check: Sep 22 12:20:34 2020
PST
Next report push: Oct 22 12:05:43 2020 PST
Last report push: Sep 22 12:05:43 2020 PST
Last report file write: <none>
Trust Code Installed:
Active: PID:C9500-16X,SN:FCW2233A5ZV
INSTALLED on Sep 22 12:02:20 2020 PST
Standby: PID:C9500-16X,SN:FCW2233A5ZY
INSTALLED on Sep 22 12:02:20 2020 PST

```


<p>show license udi (スマートライセンシング)</p> <pre>Device# show license udi UDI: PID:C9500-16X,SN:FCW2233A5ZV HA UDI List: Active:PID:C9500-16X,SN:FCW2233A5ZV Standby:PID:C9500-16X,SN:FCW2233A5ZY</pre>	<p>show license udi (スマートライセンシング)</p> <p>これは高可用性セットアップであり、このコマンドによってセットアップ内のすべての UDI が表示されます。</p> <pre>Device# show license udi UDI: PID:C9500-16X,SN:FCW2233A5ZV HA UDI List: Active:PID:C9500-16X,SN:FCW2233A5ZV Standby:PID:C9500-16X,SN:FCW2233A5ZY</pre>
--	--

移行後の CSSM Web UI

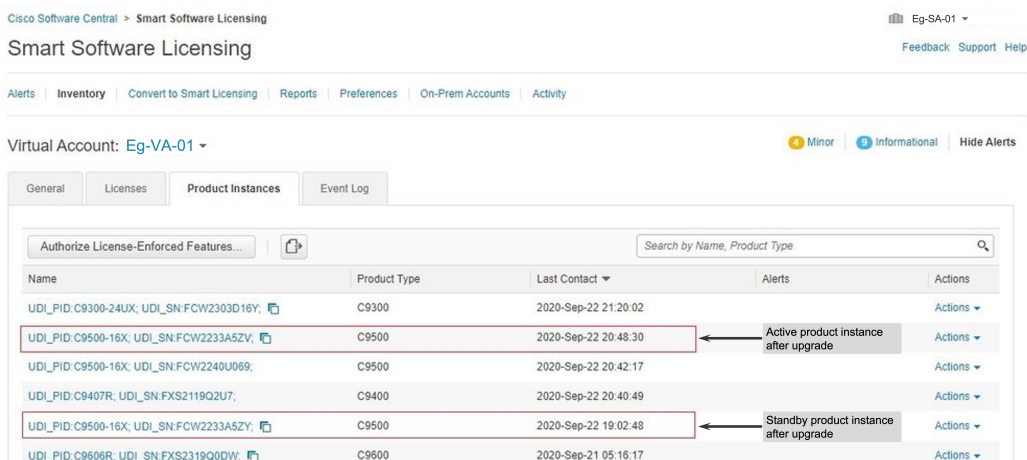
<https://software.cisco.com> で CSSM Web UI にログインし、[Smart Software Licensing] をクリックします。[Inventory] > [Product Instances] の順に選択します。

スマートライセンシング環境で登録されたライセンスは、製品インスタンスのホスト名と共に [Name] 列に表示されていました。ポリシーを使用したスマートライセンシングにアップグレードすると、製品インスタンスの UDI と共に表示されるようになります。移行したすべての UDI が表示されます。この例では、PID:C9500-16X,SN:FCW2233A5ZV および PID:C9500-16X,SN:FCW2233A5ZY がこれに該当します。

アクティブな製品インスタンスの使用状況のみが報告されるため、PID:C9500-16X,SN:FCW2233A5ZV の [License Usage] にはライセンス使用情報が表示されます。スタンバイの使用状況は報告されず、スタンバイの [License Usage] セクションには [No Records Found] と表示されます。

常にアクティブの使用状況が報告されるため、この高可用性セットアップのアクティブが変更されると、新しいアクティブな製品インスタンスのライセンス使用情報が表示され、使用状況が報告されるようになります。

図 11: スマートライセンシングからポリシーを使用したスマートライセンシングへ：移行後のアクティブおよびスタンバイ製品インスタンス



例：RTU ライセンシングからポリシーを使用したスマートライセンシングへ

図 12: スマートライセンシングからポリシーを使用したスマートライセンシングへ：アクティブな製品インスタンスでの UDI とライセンス使用状況

UDI_PID:C9500-16X; UDI_SN:FCW2233A5ZV;

Overview High Availability Event Log

Description
Nyquist Fiber C9500

General

Name: UDI_PID:C9500-16X; UDI_SN:FCW2233A5ZV;
 Product: C9500
 Host Identifier: -
 MAC Address: -
 PID: C9500-16X
 Serial Number: FCW2233A5ZV
 UUID: -
 Virtual Account: Eg-VA-01
 Registration Date: 2020-Sep-22 19:02:46
 Last Contact: 2020-Sep-22 20:48:30

License Usage

License	Billing	Expires	Required
C9500-DNA-16X-A	Prepaid	-	2
C9500 Network Advantage	Prepaid	-	2

Showing all 2 Rows

Actions ▾

移行後のレポート

製品インスタンスは、ポリシーに基づいて次の RUM レポートを CSSM に送信します。

より頻繁にレポートを作成するようにレポート間隔を変更する場合は、製品インスタンスで **license smart usage interval** コマンドを設定します。シンタックスの詳細については、対応するリリースのコマンドリファレンスで *license smart (global config)* コマンドを参照してください。

例：RTU ライセンシングからポリシーを使用したスマートライセンシングへ

次に、使用権（RTU）ライセンスからポリシーを使用したスマートライセンスに移行する Cisco Catalyst 9300 スイッチの例を示します。これはアクティブと他のメンバーを含むセットアップの例です。

RTU ライセンシングは、Cisco IOS XE Fuji 16.8.x までの Cisco Catalyst 9300、9400、および 9500 シリーズ スイッチで使用できます。スマートライセンスは、Cisco IOS XE Fuji 16.9.1 から導入されました。

ソフトウェアバージョンを、ポリシーを使用したスマートライセンスをサポートするバージョンにアップグレードすると、すべてのライセンスが IN USE として表示され、Cisco default ポリシーが製品インスタンスに適用されます。アドオンライセンスが使用されている場合、Cisco default ポリシーでは 90 日間の使用状況レポートが必要です。Cisco Catalyst アクセス、

コア、およびアグリゲーションスイッチのすべてのライセンスは適用されないため（適用タイプではないため）、機能は失われません。

- [表 13：RTU ライセンシングからポリシーを使用したスマートライセンシングへ：show コマンド](#)
- [移行後の CSSM Web UI（161 ページ）](#)
- [移行後のレポート（161 ページ）](#)

次の表に、ポリシーを使用したスマートライセンシングへのアップグレード後に、**show** コマンドの出力でチェックすべき主な変更点または新しいフィールドを示します。

表 13: RTU ライセンシングからポリシーを使用したスマートライセンシングへ：show コマンド

アップグレード前	アップグレード後
<p>show license right-to-use summary (RTU ライセンシング)</p> <pre> Device# show license right-to-use summary License Name Type Period left ----- network-essentials Permanent Lifetime dna-essentials Subscription CSSM Managed ----- License Level In Use: network-essentials+dna-essentials Subscription License Level on Reboot: network-essentials+dna-essentials Subscription </pre>	<p>show license summary (ポリシーを使用したスマートライセンシング)</p> <p>すべてのライセンスが移行され、IN USE になっています。</p> <pre> Device#show license summary License Usage: License Entitlement Tag Count Status ----- network-essentials (C9300-24 Network Essen...) 2 IN USE dna-essentials (C9300-24 DNA Essentials) 2 IN USE network-essentials (C9300-48 Network Essen...) 1 IN USE dna-essentials (C9300-48 DNA Essentials) 1 IN USE </pre>
<p>show license right-to-use usage (スマートライセンシング)</p>	<p>show license usage (ポリシーを使用したスマートライセンシング)</p> <p>すべてのライセンス（無期限、サブスクリプション）が移行され、それらのライセンスは現在 IN USE になっており、タイプには Perpetual と Subscription があります。</p> <p>Enforcement Type フィールドに NOT ENFORCED と表示されます。（Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチには、輸出規制ライセンスや適用ライセンスはありません）。</p>

例: RTU ライセンシングからポリシーを使用したスマートライセンシングへ

Device# show license right-to-use usage

```
Slot# License Name Type usage-duration(y:m:d) In-Use
EULA
```

```
-----
1 network-essentials Permanent 00:00:00 yes yes
1 network-essentials Evaluation 00:00:00 no no
1 network-essentials Subscription 00:00:00 no no
1 network-advantage Permanent 00:00:00 no no
1 network-advantage Evaluation 00:00:00 no no
1 network-advantage Subscription 00:00:00 no no
1 dna-essentials Evaluation 00:00:00 no no
1 dna-essentials Subscription 00:00:00 yes yes
1 dna-advantage Evaluation 00:00:00 no no
1 dna-advantage Subscription 00:00:00 no no
-----
```

```
Slot# License Name Type usage-duration(y:m:d) In-Use
EULA
```

```
-----
2 network-essentials Permanent 00:00:00 yes yes
2 network-essentials Evaluation 00:00:00 no no
2 network-essentials Subscription 00:00:00 no no
2 network-advantage Permanent 00:00:00 no no
2 network-advantage Evaluation 00:00:00 no no
2 network-advantage Subscription 00:00:00 no no
2 dna-essentials Evaluation 00:00:00 no no
2 dna-essentials Subscription 00:00:00 yes yes
2 dna-advantage Evaluation 00:00:00 no no
2 dna-advantage Subscription 00:00:00 no no
-----
```

```
Slot# License Name Type usage-duration(y:m:d) In-Use
EULA
```

```
-----
3 network-essentials Permanent 00:00:00 yes yes
3 network-essentials Evaluation 00:00:00 no no
3 network-essentials Subscription 00:00:00 no no
3 network-advantage Permanent 00:00:00 no no
3 network-advantage Evaluation 00:00:00 no no
3 network-advantage Subscription 00:00:00 no no
3 dna-essentials Evaluation 00:00:00 no no
3 dna-essentials Subscription 00:00:00 yes yes
3 dna-advantage Evaluation 00:00:00 no no
3 dna-advantage Subscription 00:00:00 no no
-----
```

Device# show license usage

```
License Authorization:
Status: Not Applicable
network-advantage (C9300-24 Network Advantage):
Description: C9300-24 Network Advantage
Count: 2
Version: 1.0
Status: IN USE
Export status: NOT RESTRICTED
Feature Name: network-advantage
Feature Description: C9300-24 Network Advantage
Enforcement type: NOT ENFORCED
License type: Perpetual
dna-advantage (C9300-24 DNA Advantage):
Description: C9300-24 DNA Advantage
Count: 2
Version: 1.0
Status: IN USE
Export status: NOT RESTRICTED
Feature Name: dna-advantage
Feature Description: C9300-24 DNA Advantage
Enforcement type: NOT ENFORCED
License type: Subscription
network-advantage (C9300-48 Network Advantage):
Description: C9300-48 Network Advantage
Count: 1
Version: 1.0
Status: IN USE
Export status: NOT RESTRICTED
Feature Name: network-advantage
Feature Description: C9300-48 Network Advantage
Enforcement type: NOT ENFORCED
License type: Perpetual
dna-advantage (C9300-48 DNA Advantage):
Description: C9300-48 DNA Advantage
Count: 1
Version: 1.0
Status: IN USE
Export status: NOT RESTRICTED
Feature Name: dna-advantage
Feature Description: C9300-48 DNA Advantage
Enforcement type: NOT ENFORCED
License type: Subscription
```

show license right-to-use (RTU ライセンシング)

show license status (ポリシーを使用したスマートライセンシング)

Transport: フィールドにオフになっていることが表示されます。

Trust Code Installed: フィールドには、信頼コードがインストールされていないことが表示されます。

Usage Reporting: ヘッダーの Next report push: フィールドには、次の RUM レポートを CSSM に送信するタイミングに関する情報が表示されます。

```

Device# show license right-to-use
Slot# License Name Type Period left
-----
1 network-essentials Permanent Lifetime
1 dna-essentials Subscription CSSM Managed
-----
License Level on Reboot:
network-essentials+dna-essentials
Subscription

Slot# License Name Type Period left
-----
2 network-essentials Permanent Lifetime
2 dna-essentials Subscription CSSM Managed
-----
License Level on Reboot:
network-essentials+dna-essentials
Subscription

Slot# License Name Type Period left
-----
3 network-essentials Permanent Lifetime
3 dna-essentials Subscription CSSM Managed
-----
License Level on Reboot:
network-essentials+dna-essentials
Subscription

Device# show license status
Utility:
  Status: DISABLED
Smart Licensing Using Policy:
  Status: ENABLED
Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED
Transport:
  Type: Transport Off
Policy:
  Policy in use: Merged from multiple sources.
  Reporting ACK required: yes (CISCO default)
  Unenforced/Non-Export Perpetual Attributes:
    First report requirement (days): 365 (CISCO default)

    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 90 (CISCO default)
  Unenforced/Non-Export Subscription Attributes:
    First report requirement (days): 90 (CISCO default)

    Reporting frequency (days): 90 (CISCO default)
    Report on change (days): 90 (CISCO default)
  Enforced (Perpetual/Subscription) License Attributes:

    First report requirement (days): 0 (CISCO default)
    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 0 (CISCO default)
  Export (Perpetual/Subscription) License Attributes:
    First report requirement (days): 0 (CISCO default)
    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 0 (CISCO default)
Miscellaneous:
  Custom Id: <empty>
Usage Reporting:
  Last ACK received: <none>
  Next ACK deadline: Jan 26 10:27:59 2021 PST
  Reporting push interval: 20 days
  Next ACK push check: <none>
  Next report push: Oct 28 10:29:59 2020 PST
  Last report push: <none>
  Last report file write: <none>
Trust Code Installed: <none>
    
```

移行後の CSSM Web UI

CSSM Web UI に変更はありません。

移行後のレポート

サポートされているトポロジのいずれかを実装し、レポート要件に適合するようにします。サポートされるトポロジ (118ページ) およびポリシーを使用したスマートライセンシングの設定方法：トポロジ別のワークフロー (136ページ) を参照してください。使用可能なレポートメソッドは、実装するトポロジによって異なります。

例：SLR からポリシーを使用したスマートライセンシングへ

次に、特定のライセンス予約（SLR）からポリシーを使用したスマートライセンシングに移行する Cisco Catalyst 9500 スイッチの例を示します。これはアクティブとスタンバイを含む高可用性セットアップの例です。

ライセンスの変換は自動的に行われ、承認コードが移行されます。移行を完了するためにこれ以上の操作は必要ありません。移行後は [CSSM への接続なし](#)、[CSLU なし \(124 ページ\)](#) トポロジが有効になります。ポリシーを使用したスマートライセンシング環境の SLR 承認コードについては、[承認コード \(115 ページ\)](#) を参照してください。

- [表 14：SLR からポリシーを使用したスマートライセンシングへ：show コマンド](#)
- [移行後の CSSM Web UI \(168 ページ\)](#)
- [移行後のレポート \(169 ページ\)](#)

show コマンドは、移行の前後に確認すべき以下の重要なフィールドを抽出して出力します。

表 14: SLR からポリシーを使用したスマートライセンシングへ：show コマンド

アップグレード前	アップグレード後
<p>show license summary (SLR)</p> <p>Registration ステータスフィールドと License Authorization ステータスフィールドに、ライセンスについて REGISTERED - SPECIFIC LICENSE RESERVATION および AUTHORIZED - RESERVED と表示されます。</p> <p>Device# show license summary</p> <p>Smart Licensing is ENABLED License Reservation is ENABLED Registration: Status: REGISTERED - SPECIFIC LICENSE RESERVATION Export-Controlled Functionality: ALLOWED License Authorization: Status: AUTHORIZED - RESERVED License Usage: License Entitlement tag Count Status</p> <hr/> <p>C9500 Network Advantage (C9500 Network Advantage) 2 AUTHORIZED C9500-DNA-16X-A (C9500-16X DNA Advantage) 2 AUTHORIZED</p>	<p>show license summary (ポリシーを使用したスマートライセンシング)</p> <p>Status フィールドに、ライセンスについて、登録済みおよび承認済みではなく IN USE と表示されます。</p> <p>Device# show license summary</p> <p>License Reservation is ENABLED License Usage: License Entitlement tag Count Status</p> <hr/> <p>network-advantage (C9500 Network Advantage) 2 IN USE dna-advantage (C9500-16X DNA Advantage) 2 IN USE</p>

show license reservation (SLR)

show license all (ポリシーを使用したスマートライセンシング)

License Authorizations ヘッダー：アクティブおよびスタンバイ製品インスタンスのベース (C9500 Network Advantage) ライセンスおよびアドオン (C9500-DNA-16X-A) ライセンスが特定のライセンス予約で承認されたことを示します。Authorization type: フィールドに SPECIFIC INSTALLED と表示されます。

Last Confirmation code: フィールド：高可用性セットアップのアクティブおよびスタンバイ製品インスタンスの SLR 承認コードが正常に移行されたことを示します。

例: SLR からポリシーを使用したスマートライセンシングへ

```
Device# show license reservation
License reservation: ENABLED
Overall status:
  Active: PID:C9500-16X,SN:FCW2233A5ZV
    Reservation status: SPECIFIC INSTALLED on Aug 31
    10:15:01 2020 PDT
    Export-Controlled Functionality: ALLOWED
    Last Confirmation code: 4bfbea7f
  Standby: PID:C9500-16X,SN:FCW2233A5ZY
    Reservation status: SPECIFIC INSTALLED on Aug 31
    10:15:01 2020 PDT
    Export-Controlled Functionality: ALLOWED
    Last Confirmation code: 9394f196
Specified license reservations:
C9500 Network Advantage (C9500 Network Advantage):
  Description: C9500 Network Advantage
  Total reserved count: 2
  Term information:
    Active: PID:C9500-16X,SN:FCW2233A5ZV
      License type: PERPETUAL
      Term Count: 1
    Standby: PID:C9500-16X,SN:FCW2233A5ZY
      License type: PERPETUAL
      Term Count: 1
C9500-DNA-16X-A (C9500-16X DNA Advantage):
  Description: C9500-DNA-16X-A
  Total reserved count: 2
  Term information:
    Active: PID:C9500-16X,SN:FCW2233A5ZV
      License type: TERM
      Start Date: 2020-MAR-17 UTC
      End Date: 2021-MAR-17 UTC
      Term Count: 1
    Standby: PID:C9500-16X,SN:FCW2233A5ZY
```



```
Device# show license reservation

Smart Licensing Status
=====
Smart Licensing is ENABLED
License Reservation is ENABLED
Export Authorization Key:
  Features Authorized:
    <none>
Utility:
  Status: DISABLED
Smart Licensing Using Policy:
  Status: ENABLED
Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED
Transport:
Type: Transport Off
Miscellaneous:
  Custom Id: <empty>
Policy:
  Policy in use: Merged from multiple sources.
  Reporting ACK required: yes (CISCO default)
  Unenforced/Non-Export Perpetual Attributes:
    First report requirement (days): 365 (CISCO default)

    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 90 (CISCO default)
  Unenforced/Non-Export Subscription Attributes:
    First report requirement (days): 90 (CISCO default)

    Reporting frequency (days): 90 (CISCO default)
    Report on change (days): 90 (CISCO default)
  Enforced (Perpetual/Subscription) License Attributes:

    First report requirement (days): 0 (CISCO default)
    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 0 (CISCO default)
  Export (Perpetual/Subscription) License Attributes:
    First report requirement (days): 0 (CISCO default)
    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 0 (CISCO default)
Usage Reporting:
  Last ACK received: <none>
  Next ACK deadline: Nov 29 10:50:05 2020 PDT
  Reporting Interval: 30
  Next ACK push check: <none>
Next report push: Aug 31 10:52:05 2020 PDT
  Last report push: <none>
  Last report file write: <none>
Trust Code Installed: <none>
License Usage
=====
network-advantage (C9500 Network Advantage):
  Description: network-advantage
  Count: 2
  Version: 1.0
  Status: IN USE
  Export status: NOT RESTRICTED
  Feature Name: network-advantage
  Feature Description: network-advantage
  Enforcement type: NOT ENFORCED
```

```

License type: Perpetual
Reservation:
  Reservation status: SPECIFIC INSTALLED
  Total reserved count: 2
dna-advantage (C9500-16X DNA Advantage):
  Description: C9500-16X DNA Advantage
  Count: 2
  Version: 1.0
  Status: IN USE
  Export status: NOT RESTRICTED
  Feature Name: dna-advantage
  Feature Description: C9500-16X DNA Advantage
  Enforcement type: NOT ENFORCED
  License type: Subscription
  Reservation:
    Reservation status: SPECIFIC INSTALLED
    Total reserved count: 2
Product Information
=====
UDI: PID:C9500-16X,SN:FCW2233A5ZV
HA UDI List:
  Active:PID:C9500-16X,SN:FCW2233A5ZV
  Standby:PID:C9500-16X,SN:FCW2233A5ZY
Agent Version
=====
Smart Agent for Licensing: 5.0.5_rel/42
License Authorizations
=====
Overall status:
  Active: PID:C9500-16X,SN:FCW2233A5ZV
    Status: SPECIFIC INSTALLED on Aug 31 10:15:01 2020
  PDT
    Last Confirmation code: 4bfbea7f
  Standby: PID:C9500-16X,SN:FCW2233A5ZY
    Status: SPECIFIC INSTALLED on Aug 31 10:15:01 2020
  PDT
    Last Confirmation code: 9394f196
Specified license reservations:
  C9500 Network Advantage (C9500 Network Advantage):
    Description: C9500 Network Advantage
    Total reserved count: 2
    Enforcement type: NOT ENFORCED
    Term information:
      Active: PID:C9500-16X,SN:FCW2233A5ZV
        Authorization type: SPECIFIC INSTALLED on Aug
31 10:15:01 2020 PDT
        License type: PERPETUAL
        Term Count: 1
      Standby: PID:C9500-16X,SN:FCW2233A5ZY
        Authorization type: SPECIFIC INSTALLED on Aug
31 10:15:01 2020 PDT
        License type: PERPETUAL
        Term Count: 1
  C9500-DNA-16X-A (C9500-16X DNA Advantage):
    Description: C9500-DNA-16X-A
    Total reserved count: 2
    Enforcement type: NOT ENFORCED
    Term information:
      Active: PID:C9500-16X,SN:FCW2233A5ZV
        Authorization type: SPECIFIC INSTALLED on Aug
31 10:15:01 2020 PDT
        License type: PERPETUAL
        Term Count: 1
      Standby: PID:C9500-16X,SN:FCW2233A5ZY

```

```

Authorization type: SPECIFIC INSTALLED on Aug
31 10:15:01 2020 PDT
License type: PERPETUAL
Term Count: 1
Purchased Licenses:
No Purchase Information Available
Derived Licenses:
Entitlement Tag:
regid.2017-03.com.cisco.advantagek9-Nyquist-C9500,
1.0_f1563759-2e03-4a4c-bec5-5feec525a12c
Entitlement Tag:
regid.2017-07.com.cisco.C9500-DNA-16X-A,
1.0_ef3574d1-156b-486a-864f-9f779ff3ee49
    
```

show license status (SLR)

show license status (ポリシーを使用したスマートライセンシング)

Transport: ヘッダー: Type: は、転送タイプがオフに設定されていることを示します。

Usage Reporting: ヘッダー: Next report push: フィールドは、次の RUM レポートを CSSM にアップロードする必要があるかどうか、およびアップロードする必要があるのはいつかを示します。

例: SLR からポリシーを使用したスマートライセンシングへ

```

Device# show license status
Smart Licensing is ENABLED
Utility:
  Status: DISABLED
License Reservation is ENABLED
Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED
Transport:
  Type: Callhome
Registration:
  Status: REGISTERED - SPECIFIC LICENSE RESERVATION
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Aug 31 11:07:39
2020 PDT
License Authorization:
  Status: AUTHORIZED - RESERVED on Aug 31 10:15:01 2020
PDT
Export Authorization Key:
  Features Authorized:
    <none>
    License type: TERM
    Start Date: 2020-MAR-17 UTC
    End Date: 2021-MAR-17 UTC
    Term Count: 1

Device# show license status
Utility:
  Status: DISABLED
License Reservation is ENABLED
Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED
Transport:
  Type: Transport Off
Policy:
  Policy in use: Merged from multiple sources.
  Reporting ACK required: yes (CISCO default)
  Unenforced/Non-Export Perpetual Attributes:
    First report requirement (days): 365 (CISCO default)

    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 90 (CISCO default)
  Unenforced/Non-Export Subscription Attributes:
    First report requirement (days): 90 (CISCO default)

    Reporting frequency (days): 90 (CISCO default)
    Report on change (days): 90 (CISCO default)
  Enforced (Perpetual/Subscription) License Attributes:

    First report requirement (days): 0 (CISCO default)
    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 0 (CISCO default)
  Export (Perpetual/Subscription) License Attributes:
    First report requirement (days): 0 (CISCO default)
    Reporting frequency (days): 0 (CISCO default)
    Report on change (days): 0 (CISCO default)
Miscellaneous:
  Custom Id: <empty>
Usage Reporting:
  Last ACK received: <none>
  Next ACK deadline: Nov 29 10:50:05 2020 PDT
  Reporting Interval: 30
  Next ACK push check: <none>
  Next report push: Aug 31 10:52:05 2020 PDT
  Last report push: <none>
  Last report file write: <none>
Trust Code Installed: <none>

```

移行後の CSSM Web UI

CSSM では、[Product Instances] タブに変更はありません。使用状況レポートがまだないため、[Last Contact] 列には「Reserved Licenses」と表示されます。

必要な RUM レポートがアップロードされ、「Reserved Licenses (予約済みライセンス)」が確認されると、ライセンスの使用状況がアクティブな PID 製品インスタンスのみで表示されるようになります。

図 13: SLR からポリシーを使用したスマートライセンシングへ：移行後、レポート前のアクティブおよびスタンバイ製品インスタンス

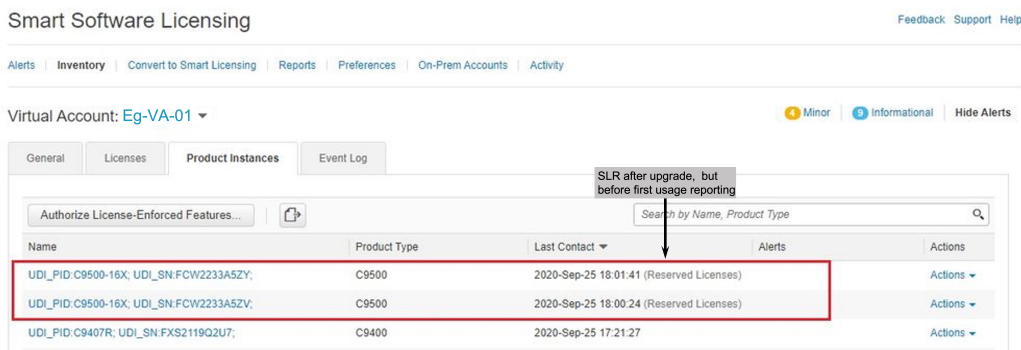
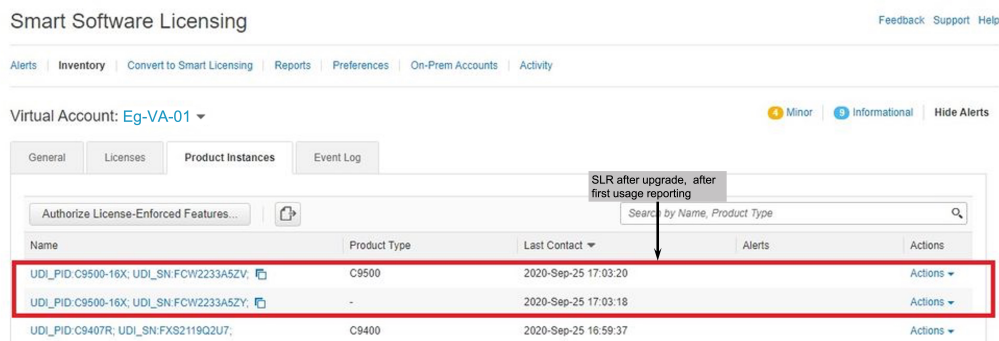


図 14: SLR からポリシーを使用したスマートライセンシングへ：移行後、レポート後のアクティブおよびスタンバイ製品インスタンス



移行後のレポート

SLR ライセンスは、ライセンスの使用状況が変化した場合にのみレポートを必要とします（たとえば、アドオンライセンスを指定された期間使用する場合）。ポリシー（**show license status**）によって変化が示されるか、変化に関する syslog メッセージが発信されます。

製品インスタンスとのすべての通信を無効にしているため、ライセンスの使用状況をレポートするには、RUM レポートをファイルに保存してから、CSSM にアップロードする必要があります（インターネットとシスコに接続されているワークステーションからアップロード）。

1. RUM レポートの生成と保存

license smart save usage コマンドを特権 EXEC モードで入力します。次の例では、すべての RUM レポートがファイル `all_rum.txt` で製品インスタンスのフラッシュメモリに保存されます。シンタックスの詳細については、対応するリリースのコマンドリファレンスで *license smart (privileged EXEC)* コマンドを参照してください。この例では、ファイルはまずブートフラッシュに保存され、次に TFTP の場所にコピーされます。

```
Device# license smart save usage all bootflash:all_rum.txt
Device# copy bootflash:all_rum.txt tftp://10.8.0.6/all_rum.txt
```

2. 使用状況データを CSSM にアップロード：[CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#)

例：評価ライセンスまたは期限切れライセンスからポリシーを使用したスマートライセンシングへ

3. ACK を製品インスタンスにインストール：製品インスタンスへのファイルのインストール (214 ページ)

例：評価ライセンスまたは期限切れライセンスからポリシーを使用したスマートライセンシングへ

以下は、評価ライセンス（スマートライセンシング）を、ポリシーを使用したスマートライセンシングに移行した Cisco Catalyst 9500 スイッチの例です。

評価ライセンスの概念は、ポリシーを使用したスマートライセンスには適用されません。ソフトウェアバージョンを、ポリシーを使用したスマートライセンシングをサポートするバージョンにアップグレードすると、すべてのライセンスが IN USE として表示され、シスコのデフォルトポリシーが製品インスタンスに適用されます。Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチのすべてのライセンスは適用されないため（適用タイプではないため）、機能は失われません。

- 表 15: 評価ライセンスまたは期限切れライセンスからポリシーを使用したスマートライセンシングへ：show コマンド
- 移行後の CSSM Web UI (172 ページ)
- 移行後のレポート (172 ページ)

次の表に、ポリシーを使用したスマートライセンシングへのアップグレード後に、show コマンドの出力でチェックすべき主な変更点または新しいフィールドを示します。

表 15: 評価ライセンスまたは期限切れライセンスからポリシーを使用したスマートライセンシングへ：show コマンド

アップグレード前	アップグレード後
<p>show license summary (スマートライセンシング、評価モード)</p> <p>ライセンスは UNREGISTERED で、EVAL MODE になっています。</p> <pre>Device# show license summary Smart Licensing is ENABLED Registration: Status: UNREGISTERED Export-Controlled Functionality: NOT ALLOWED License Authorization: Status: EVAL MODE Evaluation Period Remaining: 89 days, 21 hours, 37 minutes, 30 seconds License Usage: License Entitlement tag Count Status ----- (C9500 Network Advantage) 2 EVAL MODE (C9500-16X DNA Advantage) 2 EVAL MODE</pre>	<p>show license summary (ポリシーを使用したスマートライセンシング)</p> <p>すべてのライセンスが移行され、IN USE になっています。評価モードライセンスがありません。</p> <pre>Device# show license summary License Usage: License Entitlement tag Count Status ----- network-advantage (C9500 Network Advantage) 2 IN USE dna-advantage (C9500-16X DNA Advantage) 2 IN USE</pre>

<p>show license usage (スマートライセンシング、評価モード)</p> <pre> Device# show license usage License Authorization: Status: EVAL MODE Evaluation Period Remaining: 89 days, 21 hours, 37 minutes, 21 seconds (C9500 Network Advantage): Description: Count: 2 Version: 1.0 Status: EVAL MODE Export status: NOT RESTRICTED (C9500-16X DNA Advantage): Description: Count: 2 Version: 1.0 Status: EVAL MODE Export status: NOT RESTRICTED </pre>	<p>show license usage (ポリシーを使用したスマートライセンシング)</p> <p>Enforcement Type フィールドに NOT ENFORCED と表示されます。(Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチには、輸出規制ライセンスや適用ライセンスはありません)。</p> <pre> Device# show license usage License Authorization: Status: Not Applicable network-advantage (C9500 Network Advantage): Description: network-advantage Count: 2 Version: 1.0 Status: IN USE Export status: NOT RESTRICTED Feature Name: network-advantage Feature Description: network-advantage Enforcement type: NOT ENFORCED License type: Perpetual dna-advantage (C9500-16X DNA Advantage): Description: C9500-16X DNA Advantage Count: 2 Version: 1.0 Status: IN USE Export status: NOT RESTRICTED Feature Name: dna-advantage Feature Description: C9500-16X DNA Advantage Enforcement type: NOT ENFORCED License type: Subscription </pre>
<p>show license status (スマートライセンス、評価モード)</p>	<p>show license status (ポリシーを使用したスマートライセンシング)</p> <p>Transport: フィールドにオフになっていることが表示されます。</p> <p>Policy フィールドには、シスコのデフォルトポリシーが適用されていることが示されます。</p> <p>Trust Code Installed: フィールドには、信頼コードがインストールされていないことが表示されます。</p> <p>Usage Reporting: ヘッダー: Next report push: フィールドには、次の RUM レポートを CSSM に送信するタイミングに関する情報が表示されます。</p>

例：評価ライセンスまたは期限切れライセンスからポリシーを使用したスマートライセンシングへ

```

Switch# 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: 89 days, 21 hours, 37
minutes, 15 seconds
Export Authorization Key:
Features Authorized:
<none>
Miscellaneous:
Custom Id: <empty>

Switch# show license status

Utility:
Status: DISABLED
Smart Licensing Using Policy:
Status: ENABLED
Data Privacy:
Sending Hostname: yes
Callhome hostname privacy: DISABLED
Smart Licensing hostname privacy: DISABLED
Version privacy: DISABLED
Transport:
Type: Transport Off
Policy:
Policy in use: Merged from multiple sources.
Reporting ACK required: yes (CISCO default)
Unenforced/Non-Export Perpetual Attributes:
First report requirement (days): 365 (CISCO default)

Reporting frequency (days): 0 (CISCO default)
Report on change (days): 90 (CISCO default)
Unenforced/Non-Export Subscription Attributes:
First report requirement (days): 90 (CISCO default)

Reporting frequency (days): 90 (CISCO default)
Report on change (days): 90 (CISCO default)
Enforced (Perpetual/Subscription) License Attributes:

First report requirement (days): 0 (CISCO default)
Reporting frequency (days): 0 (CISCO default)
Report on change (days): 0 (CISCO default)
Export (Perpetual/Subscription) License Attributes:
First report requirement (days): 0 (CISCO default)
Reporting frequency (days): 0 (CISCO default)
Report on change (days): 0 (CISCO default)
Miscellaneous:
Custom Id: <empty>
Usage Reporting:
Last ACK received: <none>
Next ACK deadline: Jan 26 10:27:59 2021 PST
Reporting push interval: 20 days
Next ACK push check: <none>
Next report push: Oct 28 10:29:59 2020 PST
Last report push: <none>
Last report file write: <none>
Trust Code Installed: <none>

```

移行後の CSSM Web UI

CSSM Web UI に変更はありません。

移行後のレポート

サポートされているトポロジのいずれかを実装し、レポート要件に適合するようにします。サポートされるトポロジ ([118ページ](#)) およびポリシーを使用したスマートライセンシングの設定方法：トポロジ別のワークフロー ([136ページ](#)) を参照してください。使用可能なレポートメソッドは、実装するトポロジによって異なります。

Smart Licensing Using Policy をサポートする SSM オンプレミスのバージョンへの移行

必要な最小バージョンよりも前の SSM オンプレミスのバージョンを使用している場合（[SSM オンプレミス \(113 ページ\)](#) を参照）、SSM オンプレミスのバージョン、製品インスタンスを移行するために従う必要があるプロセスや手順、および該当する場合は SLAC のインストールのような他のタスクの概要として使用してください。

1. SSM オンプレミスをアップグレードします。

必要な最小バージョンであるバージョン 8、リリース 202102 以降にアップグレードします。

『[Cisco Smart Software Manager On-Prem Migration Guide](#)』を参照してください。

2. 製品インスタンスをアップグレードします。

サポートされている製品インスタンスに Smart Licensing Using Policy が導入された時期については、[サポート対象製品 \(128 ページ\)](#) を参照してください。

アップグレード手順については、[スイッチソフトウェアのアップグレード \(152 ページ\)](#) を参照してください。

3. CSSM へのローカルアカウントの再登録

オンラインとオフラインのオプションを使用できます。『[Cisco Smart Software Manager On-Prem Migration Guide](#)』[英語]の「*Re-Registering a local Account (Online Mode)*」または「*Manually Re-Registering a Local Account (Offline Mode)*」を参照してください。

再登録が完了すると、次のイベントが自動的に発生します。

- SSM オンプレミスは、SSM オンプレミスのテナントを指す新しいトランスポート URL で応答します。
- 製品インスタンスのトランスポートタイプ設定が **call-home** または **smart** から **cslu** に変更されます。トランスポート URL も自動的に更新されます。

4. 特権 EXEC モードで **copy running-config startup-config** コマンドを入力して、製品インスタンスの設定変更を保存します。

5. 製品インスタンスの古いオンプレミス スマート ライセンス 証明書をクリアし、製品インスタンスをリロードします。この後は設定変更を保存しないでください。



(注) この手順は、製品インスタンスで実行されているソフトウェアバージョンが Cisco IOS XE Amsterdam 17.3.x または Cisco IOS XE Bengaluru 17.4.x の場合にのみ必要です。

特権 EXEC モードで **licence smart factory reset** コマンドと **reload** コマンドを入力します。

```
Device# licence smart factory reset
Device# reload
```

6. 使用状況の同期の実行

1. 製品インスタンスに特権 EXEC モードで **license smart sync {all|local}** コマンドを入力します。これにより、SSM オンプレミスと製品インスタンスが同期され、保留中のデータが送受信されます。

```
Device(config)# license smart sync local
```

これは、SSM オンプレミス UI で確認できます。[Inventory] > [SL Using Policy] に移動します。[Alerts] 列に、「Usage report from product instance」というメッセージが表示されます。

2. 使用状況情報を CSSM と同期します (いずれかを選択)。

- オプション 1 :

SSM オンプレミスが CSSM に接続されている場合 : SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports] > [Usage Schedules] > [Synchronize now with Cisco] に移動します。

- オプション 2 :

SSM オンプレミスが CSSM に接続されていません。 [使用状況データのエクスポートとインポート \(SSM オンプレミス UI\) \(199 ページ\)](#) を参照してください。

結果 :

移行および使用状況の最初の同期が完了しました。製品インスタンスとライセンス使用状況情報が SSM オンプレミスに表示されるようになりました。

後続のレポートには、次のオプションが含まれています。

- 製品インスタンスと SSM オンプレミスとの間でデータを同期するには、次の手順を実行します。
 - レポート間隔を設定して、製品スタンスと SSM オンプレミスとの間の定期的な同期をスケジュールします。グローバル コンフィギュレーション モードで **license smart usage interval interval_in_days** コマンドを入力します。

製品インスタンスが次にいつ RUM レポートを送信するかを確認するには、特権 EXEC モードで **show license all** コマンドを入力し、出力の [Next report push:] フィールドを確認します。
- 製品インスタンスと SSM オンプレミスとの間でアドホックまたはオンデマンドの同期を行うには、**license smart sync** 特権 EXEC コマンドを入力します。
- 使用状況情報を CSSM と同期するには、次の手順を実行します。
 - CSSM との定期的な同期をスケジュールします。SSM オンプレミス UI で、[Reports] > [Usage Schedules] > [Synchronization schedule with Cisco] に移動します。次の頻度情報を入力し、保存します。

- [Days] : 同期が実行される頻度を示します。たとえば、2 を入力すると、同期は 2 日に 1 回行われます。
- [Time of Day] : 24 時間表記法で、同期が実行される時刻を示します。たとえば、14 hours と 0 minutes を入力すると、ローカルタイムゾーンの午後 2 時 (1400) に同期が行われます。
- レポートに必要なファイルのアップロードとダウンロードを実行します ([使用状況データのエキスポートとインポート \(SSM オンプレミス UI\) \(199 ページ\)](#))。

ポリシーを使用したスマートライセンシングのタスクライブラリ

このセクションでは、ポリシーを使用したスマートライセンシングに適用されるタスクのグループ化について説明します。製品インスタンス、CSLU インターフェイス、および CSSM Web UI で実行されるタスクが含まれます。

特定のトポロジを実装するには、対応するワークフローを参照して、適用されるタスクの順序を確認します。[ポリシーを使用したスマートライセンシングの設定方法：トポロジ別のワークフロー \(136 ページ\)](#) を参照してください

追加の設定タスクを実行する場合 (たとえば別のライセンスの設定、アドオンライセンスの使用、またはより短いレポート間隔の設定) は、対応するタスクを参照してください。続行する前に、入手可能な場合には「サポートされるトポロジ」を確認してください。

シスコへのログイン (CSLU インターフェイス)

必要に応じて、CSLU で作業するときに接続モードまたは切断モードのいずれかにすることができます。接続モードで作業するには、次の手順を実行してシスコに接続します。

手順

-
- ステップ 1** CSLU のメイン画面で、[Login to Cisco] (画面の右上隅) をクリックします。
 - ステップ 2** [CCO User Name] と [CCO Password] を入力します。
 - ステップ 3** CSLU の [Preferences] タブで、シスコ接続トグルに「Cisco Is Available」と表示されていることを確認します。
-

スマートアカウントとバーチャルアカウントの設定 (CSLU インターフェイス)

スマートアカウントとバーチャルアカウントはどちらも [Preferences] タブで設定します。シスコに接続するためのスマートアカウントとバーチャルアカウントの両方を設定するには、次の手順を実行します。

手順

ステップ 1 CSLU のホーム画面から [Preferences] タブを選択します。

ステップ 2 スマートアカウントとバーチャルアカウントの両方を追加するには、次の手順を実行します。

- a) [Preferences] 画面で、[Smart Account] フィールドに移動し、[Smart Account Name] を追加します。
- b) 次に、[Virtual Account] フィールドに移動し、[Virtual Account Name] を追加します。

CSSM に接続している場合 ([Preferences] タブに「Cisco is Available」)、使用可能な SA/VA のリストから選択できます。

CSSM に接続していない場合 ([Preferences] タブに「Cisco Is Not Available」)、SA/VA を手動で入力します。

(注) SA/VA 名では大文字と小文字が区別されます。

ステップ 3 [Save] をクリックします。SA/VA アカウントがシステムに保存されます。

一度に 1 つの SA/VA ペアのみが CSLU に存在できます。複数のアカウントを追加することはできません。別の SA/VA ペアに変更するには、ステップ 2a および 2b を繰り返してから [Save] をクリックします。新しい SA/VA アカウントペアは、以前に保存されたペアを置き換えます。

CSLU での製品開始型製品インスタンスの追加 (CSLU インターフェイス)

[Preferences] タブを使用してデバイス作成の製品インスタンスを追加するには、次の手順を実行します。

手順

ステップ 1 [Preferences] タブを選択します。

ステップ 2 [Preferences] 画面で、[Validate Instance] チェックボックスをオフにします。

ステップ 3 [Default Instance Method] を [Product Instance Initiated] に設定し、[Save] をクリックします。

製品インスタンス開始型通信のネットワーク到達可能性の確認

このタスクでは、製品インスタンス開始型通信のネットワーク到達可能性を確認するために必要になる可能性のある設定を提供します。「(必須)」と付いている手順は、すべての製品インスタンスで必須です。他のすべての手順は、製品インスタンスの種類とネットワーク要件に応じて、必須の場合も任意の場合もあります。該当するコマンドを設定します。

始める前に

サポートされるトポロジ: CSLU を介して CSSM に接続 (製品インスタンス開始型通信)。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	interface interface-type-number 例 : Device (config)# interface gigabitethernet0/0	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、VRFに関連付けられたイーサネットインターフェイス、サブインターフェイス、または VLAN を指定します。
ステップ 4	vrf forwarding vrf-name 例 : Device (config-if)# vrf forwarding Mgmt-vrf	VRF をレイヤ 3 インターフェイスに対応付けます。このコマンドは、インターフェイスでマルチプロトコル VRF をアクティブにします。
ステップ 5	ip address ip-address mask 例 : Device (config-if)# ip address 192.168.0.1 255.255.0.0	VRF の IP アドレスを定義します。
ステップ 6	negotiation auto 例 : Device (config-if)# negotiation auto	インターフェイスの速度およびデュプレックスパラメータの自動ネゴシエーション動作を有効にします。
ステップ 7	end 例 :	インターフェイス コンフィギュレーションモードを終了し、グローバルコ

	コマンドまたはアクション	目的
	Device(config-if)# end	ンフィギュレーションモードを開始します。
ステップ 8	ip http client source-interface interface-type-number 例： Device(config)# ip http client source-interface gigabitethernet0/0	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。
ステップ 9	ip route ip-address ip-mask subnet mask 例： Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1	(必須) 製品インスタンスにルートとゲートウェイを設定します。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 10	{ip ipv6} name-server server-address 1 ...server-address 6] 例： Device(config)# Device(config)# ip name-server vrf mgmt-vrf 173.37.137.85	VRF インターフェイスでドメインネームシステム (DNS) を設定します。
ステップ 11	ip domain lookup source-interface interface-type-number 例： Device(config)# ip domain lookup source-interface gigabitethernet0/0	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 12	ip domain name domain-name 例： Device(config)# ip domain name example.com	ドメインの DNS ディスカバリーを設定します。この例では、ネームサーバは エントリ cslu-local.example.com を作成します。

CSLU での CSLU 開始型製品インスタンスの追加 (CSLU インターフェイス)

CSLU インターフェイスを使用して、接続方法を CSLU 開始型に設定できます。この接続方法 (モード) により、CSLU は製品インスタンスから製品インスタンス情報を取得できます。



(注) デフォルトの接続方法は、[Preferences] タブで設定されます。

[Inventory] タブから製品インスタンスを追加するには、次の手順を実行します。

手順

-
- ステップ 1 [Inventory] タブに移動し、[Product Instances] テーブルから [Available Actions] → [Add Single Product Instance] を選択します。
 - ステップ 2 [Host] (ホストの IP アドレス) を入力します。
 - ステップ 3 [Connect Method] を選択し、適切な [CSLU Initiated] 接続方法を選択します。
 - ステップ 4 右側のパネルで、[Product Instance Login Credentials] をクリックします。画面の左側のパネルが変化して [User Name] フィールドと [Password] フィールドに変わります。
[General] をクリックすると、詳細な [Add Product] ウィンドウが開きます。
 - ステップ 5 製品インスタンスの [User Name] と [Password] を入力します。
 - ステップ 6 [Save] をクリックします。
情報がシステムに保存され、デバイスが [Product Instances] にリストされて、[Last Contact] には [-never] と表示されます。
-

使用状況レポートの収集 : CSLU 開始

CSLU では、デバイスからの使用状況レポートの収集を手動でトリガーすることもできます。

製品インスタンスを設定して選択した後 ([Add Single Product Instance] を選択し、[Host] に名前を入力して [CSLU Initiated] 接続メソッドを選択)、[Actions for Selected] > [Collect Usage] を選択します。CSLU は選択した製品インスタンスに接続し、使用状況レポートを収集します。収集された使用状況レポートは、CSLU のローカルライブラリに保存されます。これらのレポートは、CSLU がシスコに接続されている場合はシスコに転送できます。または (シスコに接続されていない場合は) [Product Instances] > [Download] の順に選択して、手動で使用状況の収集をトリガーできます。

CSLU 開始モードで作業している場合は、次の手順を実行して、製品インスタンスから RUM レポートを収集するように CSLU を設定します。

手順

-
- ステップ 1 [Preferences] タブを選択し、有効な [Smart Account] と [Virtual Account] を入力して、適切な [CSLU Initiated] 収集メソッドを選択します。 ([Preferences] に変更があった場合は、[Save] をクリックします)。
 - ステップ 2 [Inventory] タブを開き、1つまたは複数の製品インスタンスを選択します。
 - ステップ 3 CSLU のメイン画面で、[Available actions] > [Collect Usage] の順に選択します。
RUM レポートは、選択した各デバイスから取得され、CSLU ローカルライブラリに保存されます。[Last Contacted] 列が更新され、レポートが受信された時刻が表示されます。[Alerts] 列にはステータスが表示されます。

CSLU が現在シスコにログインしている場合、レポートはシスコの関連するスマートアカウントとバーチャルアカウントに自動的に送信され、シスコは CSLU と製品インスタンスに確認応答を送信します。確認応答は、[Product Instance] テーブルの [Alerts] 列に表示されます。

使用状況レポートをシスコに手動で転送するには、[Product Instances] メニューから [Download for Cisco] を選択します。

ステップ 4 [Download for Cisco] モーダルから、レポートを保存するローカルディレクトリを選択します。
(<CSLU_WORKING_Directory>/data/default/rum/unsent)

この時点で、使用状況レポートがローカルディレクトリ（ライブラリ）に保存されます。使用状況レポートをシスコにアップロードするには、[CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) の手順に従ってください。

(注) Windows オペレーティングシステムでは、ファイルの名前が変更されたときに拡張子をドロップすることで、使用状況レポートファイルのプロパティの動作を変更できます。動作の変更は、ダウンロードしたファイルの名前を変更し、名前を変更したファイルが拡張子をドロップすると発生します。たとえば、UD_xxx.tar という名前のダウンロード済みデフォルトファイルの名前が UD_yyy に変更されたとします。ファイルは tar 拡張子を失い、機能しなくなります。使用状況ファイルを正常に機能させるには、使用状況レポートファイルの名前を変更した後、UD_yyy.tar のように、ファイル名に tar 拡張子を追加する必要があります。

Download All For Cisco (CSLU インターフェイス)

[Download All for Cisco] メニューオプションは、オフラインの目的で使用される手動プロセスです。[Download For Cisco] メニューオプションを使用するには、次の手順を実行します。

手順

ステップ 1 CSLU の [Preferences] タブ画面で、[Cisco Connectivity] トグルスイッチをオフにします。
フィールドが「Cisco Is Not Available」に切り替わります。

ステップ 2 [Product Instances] > [Download All For Cisco] に移動します。

ステップ 3 開いたウィンドウから **ファイル** を選択し、[Save] をクリックします。これでファイルが保存されました。

(注) この時点で、DLC ファイル、RUM ファイル、またはその両方があります。

ステップ 4 シスコに接続できる端末に移動し、次の手順を実行します。[CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#)

ファイルがダウンロードされたら、**CSLU** に転送できます。

ステップ 5 [Upload from Cisco] をクリックします。 [Upload From Cisco \(CSLU インターフェイス\) \(181 ページ\)](#) を参照してください。

Upload From Cisco (CSLU インターフェイス)

シスコから ACK またはその他のファイル (承認コードなど) を受信すると、そのファイルをシステムにアップロードできます。この手順は、オフラインのワークステーションに使用できます。シスコからファイルを選択してアップロードするには、次の手順を実行します。

手順

ステップ 1 デバイスの **ACK ファイル** がダウンロードされていることを確認します。次を参照してください。 [Download All For Cisco \(CSLU インターフェイス\) \(180 ページ\)](#)

ステップ 2 CSLU のメイン画面から、[Product Instance] > [Upload from Cisco] を選択します。

ステップ 3 [Cisco File Upload] ウィンドウが開き、次のいずれかを実行できます。

- ローカルドライブにある **ファイル** をドラッグアンドドロップします。または、
- 適切な *.xml ファイルを参照し、[File] を選択して [Open] をクリックします。

アップロードが成功すると、ACK ファイルがサーバに正常に送信されたことを示すメッセージが表示されます。アップロードが成功しない場合は、インポートエラーが発生します。

ステップ 4 アップロードが完了したら、ウィンドウの右上隅にある [x] をクリックして閉じます。

CSLU 開始型通信のネットワーク到達可能性の確認

このタスクでは、CSLU 開始型通信のネットワーク到達可能性を確認するために必要になる可能性のある設定を提供します。「(必須)」と付いている手順は、すべての製品インスタンスで必須です。他のすべての手順は、製品インスタンスの種類とネットワーク要件に応じて、必須の場合も任意の場合もあります。該当するコマンドを設定します。

始める前に

サポートされるトポロジ: CSLU を介して CSSM に接続 (CSLU 開始型通信)。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例: Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new model 例： Device(config)# aaa new model	(必須) 認証、許可、アカウントイン 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 例： Device(config)# aaa authentication login default local	(必須) 認証時にローカルのユーザ名 データベースを使用するように、AAA 認証を設定します。
ステップ 5	aaa authorization exec default local 例： Device(config)# aaa authorization exec default local	ネットワークへのユーザアクセスを制限するパラメータを設定します。ユーザは EXEC シェルの実行が許可されま す。
ステップ 6	ip routing 例： Device(config)# ip routing	IP ルーティングを有効にします。
ステップ 7	{ip ipv6} name-server server-address 1 ...server-address 6] 例： Device(config)# ip name-server vrf Mgmt-vrf 192.168.1.100 192.168.1.200 192.168.1.300	(任意) 名前とアドレスの解決に使用 する 1 つまたは複数のネームサーバの アドレスを指定します。 最大 6 つのネーム サーバを指定できま す。各サーバアドレスはスペースで区 切ります。最初に指定されたサーバ が、プライマリサーバです。デバイスは、プライマリサーバへ DNS クエリを 最初に送信します。そのクエリが失敗 した場合は、バックアップサーバにク エリが送信されます。
ステップ 8	ip domain lookup source-interface interface-type-number 例： Device(config)# ip domain lookup source-interface gigabitethernet0/0	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有効にしま す。この機能は、デフォルトでイネー ブルにされています。 ユーザのネットワークデバイスが、名 前の割り当てを制御できないネット ワーク内のデバイスと接続する必要が ある場合、グローバルなインターネッ トのネーミング方式 (DNS) を使用し

	コマンドまたはアクション	目的
		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 9	<p>ip domain name <i>name</i></p> <p>例 :</p> <pre>Device(config)# ip domain name vrf Mgmt-vrf cisco.com</pre>	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。
ステップ 10	<p>no username <i>name</i></p> <p>例 :</p> <pre>Device(config)# no username admin</pre>	<p>（必須）指定されたユーザ名が存在する場合はクリアします。<i>name</i> には、次のステップで作成するユーザ名と同じものを入力します。これにより、次のステップで作成するユーザ名が重複していないことが保証されます。</p> <p>CSLU 開始型の RUM レポート取得に REST API を使用する場合は、CSLU にログインする必要があります。ここでユーザ名が重複していると、システムにユーザ名が重複している場合にこの機能が正しく動作しないことがあります。</p>
ステップ 11	<p>username <i>name</i> privilege <i>level</i> password <i>password</i></p> <p>例 :</p> <pre>Device(config)# username admin privilege 15 password 0 lab</pre>	<p>（必須）ユーザ名をベースとした認証システムを構築します。</p> <p>privilege キーワードにより、ユーザの権限レベルを設定します。ユーザの権限レベルを指定する 0 ~ 15 の数字です。</p> <p>password を使用すると、<i>name</i> 引数にアクセスできます。パスワードは 1 ~ 25 文字で、埋め込みスペースを使用でき、username コマンドの最後のオプションとして指定します。</p> <p>これにより、CSLU が製品インスタンスのネイティブ REST を使用できるようになります。</p>

	コマンドまたはアクション	目的
		(注) このユーザ名とパスワードをCSLUで入力します(使用状況レポートの収集: CSLU 開始 (179 ページ) →ステップ4.f)。その後、CSLUは製品インスタンスからRUMレポートを収集できます。
ステップ 12	interface interface-type-number 例: Device (config)# interface gigabitethernet0/0	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、VRFに関連付けられたイーサネットインターフェイス、サブインターフェイス、またはVLANを指定します。
ステップ 13	vrf forwarding vrf-name 例: Device(config-if)# vrf forwarding Mgmt-vrf	VRFをレイヤ3インターフェイスに対応付けます。このコマンドは、インターフェイスでマルチプロトコルVRFをアクティブにします。
ステップ 14	ip address ip-address mask 例: Device(config-if)# ip address 192.168.0.1 255.255.0.0	VRFのIPアドレスを定義します。
ステップ 15	negotiation auto 例: Device(config-if)# negotiation auto	インターフェイスの速度およびデュプレックスパラメータの自動ネゴシエーション動作を有効にします。
ステップ 16	no shutdown 例: Device(config-if)# no shutdown	無効にされたインターフェイスを再起動します。
ステップ 17	end 例: Device(config-if)# end	インターフェイス 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を開始します。
ステップ 18	ip http server 例: Device(config)# ip http server	(必須) シスコのWebブラウザユーザインターフェイスを含むIPまたはIPv6システムでHTTPサーバを有効にします。HTTPサーバは、デフォルトにより標準のポート80を使用します。

	コマンドまたはアクション	目的
ステップ 19	ip http authentication local 例 : ip http authentication local Device(config)#	(必須) HTTP サーバユーザに対して特定の認証方法を指定します。 local キーワードは、認証および許可に、ローカルシステム設定で (username グローバルコンフィギュレーションコマンドによって) 指定したログインユーザ名、パスワード、権限レベルアクセスの組み合わせを使用することを示します。
ステップ 20	ip http secure-server 例 : Device(config)# ip http server	(必須) セキュア HTTP (HTTPS) サーバを有効にします。HTTPS サーバは、セキュアソケットレイヤ (SSL) バージョン 3.0 プロトコルを使用します。
ステップ 21	ip http max-connections 例 : Device(config)# ip http max-connections 16	(必須) HTTP サーバへの同時最大接続数を設定します。1 ~ 16 の範囲の整数を入力します。デフォルトは 5 です。
ステップ 22	ip tftp source-interface interface-type-number 例 : Device(config)# ip tftp source-interface GigabitEthernet0/0	TFTP 接続用の送信元アドレスとして、インターフェイスの IP アドレスを指定します。
ステップ 23	ip route ip-address ip-mask subnet mask 例 : Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1	製品インスタンスにルートとゲートウェイを設定します。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 24	logging host 例 : Device(config)# logging host 172.25.33.20 vrf Mgmt-vrf	リモートホストへのシステムメッセージおよびデバッグ出力を記録します。
ステップ 25	end 例 : Device(config)# end	グローバルコンフィギュレーションモードを終了し、特権 EXEC モードを開始します。
ステップ 26	show ip http server session-module 例 :	(必須) HTTP 接続を確認します。出力で、SL_HTTP がアクティブであるこ

	コマンドまたはアクション	目的
	Device# show ip http server session-module	<p>とを確認します。また、次のチェックも実行できます。</p> <ul style="list-style-type: none"> • CSLU がインストールされているデバイスから、製品インスタンスに ping できることを確認します。ping が成功すると、製品インスタンスが到達可能であることが確認されます • CSLU がインストールされているデバイスの Web ブラウザで、 <code>https://<product-instance-ip>/</code> を確認します。これにより、CSLU から製品インスタンスへの REST API が期待どおりに動作することが保証されます。

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。「(必須)」と付いている手順は、すべての製品インスタンスで必須です。他のすべての手順は、製品インスタンスの種類とネットワーク要件に応じて、必須の場合も任意の場合もあります。該当するコマンドを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	{ip ipv6} name-server server-address 1 ...server-address 6] 例： Device(config)# ip name-server 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230	<p>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。</p> <p>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</p>

	コマンドまたはアクション	目的
		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。
ステップ 4	<p>ip name-server vrf Mgmt-vrf <i>server-address 1...server-address 6</i></p> <p>例 :</p> <pre>Device(config)# ip name-server vrf Mgmt-vrf 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230</pre>	<p>(任意) VRF インターフェイスでDNSを設定します。最大6つのネームサーバを指定できます。各サーバアドレスはスペースで区切ります。</p> <p>(注) このコマンドは、ip name-server コマンドの代わりです。</p>
ステップ 5	<p>ip domain lookup source-interface <i>interface-type interface-number</i></p> <p>例 :</p> <pre>Device(config)# ip domain lookup source-interface Vlan100</pre>	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	<p>ip domain name <i>domain-name</i></p> <p>例 :</p> <pre>Device(config)# ip domain name example.com</pre>	ドメイン名を設定します。
ステップ 7	<p>ip host tools.cisco.com <i>ip-address</i></p> <p>例 :</p> <pre>Device(config)# ip host tools.cisco.com 209.165.201.30</pre>	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	<p>interface <i>interface-type-number</i></p> <p>例 :</p> <pre>Device(config)# interface Vlan100 Device(config-if)# ip address 192.0.2.10 255.255.255.0 Device(config-if)# exit</pre>	レイヤ3インターフェイスを設定します。インターフェイスのタイプと番号、またはVLANを入力します。
ステップ 9	<p>ntp server <i>ip-address</i> [version number] [key key-id] [prefer]</p> <p>例 :</p> <pre>Device(config)# ntp server 198.51.100.100 version 2 prefer</pre>	(必須) NTPサービスをアクティブにし(まだアクティブになっていない場合)、システムがシステムソフトウェアクロックを指定されたNTPサーバと同期できるようにします。これにより、デバイスの時刻がCSSMと同期されます。

	コマンドまたはアクション	目的
		このコマンドを複数回使用する必要があるために優先サーバを設定する場合は、 prefer キーワードを使用します。このキーワードを使用すると、サーバ間の切り換え回数が減少します。
ステップ 10	switchport access vlan <i>vlan_id</i> 例 : <pre>Device(config)# interface GigabitEthernet1/0/1 Device(config-if)# switchport access vlan 100 Device(config-if)# switchport mode access Device(config-if)# exit OR Device(config)#</pre>	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。 (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。 switchport access vlan コマンドは、たとえば Catalyst スイッチング製品インスタンスに適用できます。ルーティング製品インスタンスの場合は、代わりに ip address <i>ip-address mask</i> コマンドを設定できます。
ステップ 11	ip route <i>ip-address ip-mask subnet mask</i> 例 : <pre>Device(config)# ip route 192.0.2.0 255.255.255.255 192.0.2.1</pre>	デバイスにルートを設定します。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	ip http client source-interface <i>interface-type-number</i> 例 : <pre>Device(config)# ip http client source-interface Vlan100</pre>	(必須) 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。インターフェイスのタイプと番号、または VLAN を入力します。
ステップ 13	exit 例 : <pre>Device(config)# exit</pre>	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 14	copy running-config startup-config 例 : <pre>Device# copy running-config startup-config</pre>	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 プロキシを介したスマート転送の設定

スマート転送モードを使用している場合にプロキシサーバを使用して CSSM と通信するには、次の手順を実行します。



(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。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	license smart transport smart 例： Device (config)# license smart transport smart	スマート転送モードを有効にします。
ステップ 4	license smart url default 例： Device (config)# license smart transport default	スマート URL を自動的に設定します (https://smartreceiver.cisco.com/licservice/license)。このオプションを想定どおりに動作させるには、前の手順の転送モードを smart に設定する必要があります。
ステップ 5	license smart proxy {address address_hostname port port_num} 例： Device (config)# license smart proxy address 192.168.0.1 Device (config)# license smart proxy port 3128	スマート転送モードのプロキシを設定します。プロキシが設定されている場合、ライセンスメッセージは最終宛先 URL (CSSM) に加えてプロキシにも送信されます。プロキシはメッセージを CSSM に送信します。プロキシ IP アドレスとポート情報を個別に設定します。 • address address_hostname : プロキシアドレスを指定します。プロキシサーバの IP アドレスまたはホスト名を入力します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • port port_num : プロキシポートを指定します。プロキシポートポート番号を入力します。
ステップ 6	exit 例 : Device(config)# exit	グローバル コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。
ステップ 7	copy running-config startup-config 例 : Device# copy running-config startup-config	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

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。転送モードを設定するには、Call Home サービスを有効にし、宛先プロファイルを設定して（宛先プロファイルには、アラート通知に必要な配信情報が含まれます。少なくとも 1 つの宛先プロファイルが必要です）、次の手順を実行します。



(注) 「(任意)」と特に明記されていない限り、すべての手順を実行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	enable 例 : Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	license smart transport callhome 例 : Device(config)# license smart transport callhome	転送モードとして Call Home を有効にします。
ステップ 4	license smart url url 例 :	callhome 転送モードの場合は、例に示すように CSSM URL を設定します。

	コマンドまたはアクション	目的
	<pre>Device (config) # license smart url https://tools.cisco.com/its/service/cthe/services/DOEService</pre>	
ステップ 5	<p>service call-home</p> <p>例 :</p> <pre>Device (config) # service call-home</pre>	Call Home 機能をイネーブルにします。
ステップ 6	<p>call-home</p> <p>例 :</p> <pre>Device (config) # call-home</pre>	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 7	<p>contact-email-address email-address</p> <p>例 :</p> <pre>Device (config-call-home) # contact-email-addr username@example.com</pre>	お客様の電子メールアドレスを割り当て、Smart Call Home サービスのフルレポート機能を有効にし、フルインベントリメッセージを Call Home TAC プロファイルから Smart Call Home サーバに送信してフル登録プロセスを開始します。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 8	<p>profile name</p> <p>例 :</p> <pre>Device (config-call-home) # profile CiscoTAC-1 Device (config-call-home-profile) #</pre>	<p>指定された宛先プロファイルに対する Call Home 宛先プロファイル設定サブモードに入ります。</p> <p>デフォルトは次のとおりです。</p> <ul style="list-style-type: none"> • 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。 • CiscoTAC-1 プロファイルは、プロファイルに登録されているすべてのイベントタイプが記載された完全なレポートを送信します。または、 <pre>Device (cfg-call-home-profile) # anonymous-reporting-only</pre> anonymous-reporting-only を追加で設定します。これが設定されている場合は、クラッシュ、インベントリ、およびテストメッセージのみが送信されます。

	コマンドまたはアクション	目的
		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。
ステップ 9	active 例： Device(config-call-home-profile)# active	宛先プロファイルをイネーブルにします。
ステップ 10	destination transport-method http{email http} 例： Device(config-call-home-profile)# destination transport-method http AND Device(config-call-home-profile)# no destination transport-method email	メッセージの転送形式をイネーブルにします。この例では、HTTP 経由で Call Home サービスが有効になり、電子メールによる転送が無効になります。 このコマンドの no 形式を使用すると、メソッドが無効になります。
ステップ 11	destination address { email email_address http url} 例： Device(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/otbe/services/ITCServices AND Device(config-call-home-profile)# no destination address http https://tools.cisco.com/its/service/otbe/services/ITCServices	Call Home メッセージを送信する宛先 E メールアドレスまたは URL を設定します。宛先 URL を入力する場合は、サーバがセキュアサーバであるかどうかに応じて http:// (デフォルト) または https:// を指定します。 ここに示す例では、 http:// の形式で宛先 URL が設定されています。コマンドの no 形式では https:// に設定されます。
ステップ 12	exit 例： Device(config-call-home-profile)# exit	Call Home 宛先プロファイル コンフィギュレーションモードを終了して、Call Home コンフィギュレーションモードに戻ります。
ステップ 13	exit 例： Device(config-call-home)# end	Call Home コンフィギュレーションモードを終了して、特権 EXEC モードに戻ります。
ステップ 14	copy running-config startup-config 例： Device# copy running-config startup-config	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
ステップ 15	show call-home profile { <i>name</i> all}	指定されたプロファイル、または設定済みのすべてのプロファイルに関する宛先プロファイル設定を表示します。

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。



(注) 「(任意)」と特に明記されていない限り、すべての手順を実行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	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	license smart transport callhome 例： Device (config)# license smart transport callhome	転送モードとして Call Home を有効にします。
ステップ 4	service call-home 例： Device (config)# service call-home	Call Home 機能をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 5	call-home 例 : Device (config) # call-home	Call Home コンフィギュレーションモードを開始します。
ステップ 6	http-proxy proxy-address proxy-port port-number 例 : Device (config-call-home) # http-proxy 198.51.100.10 port 5000	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 7	exit 例 : Device (config-call-home) # exit	Call Home コンフィギュレーションモードを終了し、グローバル コンフィギュレーションモードを開始します。
ステップ 8	exit 例 : Device (config) # exit	グローバル コンフィギュレーションモードを終了し、特権 EXEC モードを開始します。
ステップ 9	copy running-config startup-config 例 : Device# copy running-config startup-config	コンフィギュレーションファイルに設定を保存します。

スマートアカウントとバーチャルアカウントの割り当て (SSM オンプレミス UI)

この手順を使用して、1 つ以上の製品インスタンスに対応するスマートアカウントおよびバーチャルアカウント情報とともに SSM オンプレミスのデータベースにインポートできます。これにより、SSM オンプレミスは、ローカルバーチャルアカウント（デフォルトのローカルバーチャルアカウント以外）の一部である製品インスタンスを CSSM の正しいライセンスプールにマッピングできます。

始める前に

サポートされているトポロジ : SSM オンプレミス展開（製品スタンス開始型通信）。

手順

-
- ステップ 1 SSM オンプレミスにログインし、[Smart Licensing] ワークスペースを選択します。
 - ステップ 2 [Inventory] > [SL Using Policy] > [Export/Import All] > [Import Product Instances List] に移動します。
[Upload Product Instances] ウィンドウが表示されます。

- ステップ 3** [Download] をクリックして .csv テンプレートファイルをダウンロードし、テンプレート内のすべての製品インスタンスに必要な情報を入力します。
- ステップ 4** テンプレートに入力したら、[Inventory] > [SL Using Policy] > [Export/Import All] > [Import Product Instances List] をクリックします。
- [Upload Product Instances] ウィンドウが表示されます。
- ステップ 5** [Browse] をクリックし、入力した .csv テンプレートをアップロードします。
- アップロードしたすべての製品インスタンスのスマートアカウント情報とバーチャルアカウント情報が SSM オンプレミスで使用できるようになりました。

デバイスの検証 (SSM オンプレミス UI)

デバイス検証が有効になっている場合、不明な製品インスタンス (SSM オンプレミスデータベース内にない) からの RUM レポートは拒否されます。

デフォルトでは、デバイスは検証されません。この機能を有効にするには、次の手順を実行します。

始める前に

サポートされているトポロジ: SSM オンプレミス展開 (製品スタンス開始型通信)。

手順

- ステップ 1** [On-Prem License Workspace] ウィンドウで、[Admin Workspace] をクリックし、プロンプトが表示されたらログインします。
- [On-Prem Admin Workspace] ウィンドウが表示されます。
- ステップ 2** [Settings] ウィジェットをクリックします。
- [Settings] ウィンドウが表示されます。
- ステップ 3** [CSLU] タブに移動し、[Validate Device] トグルスイッチをオンにします。
- 不明な製品インスタンスからの RUM レポートが拒否されるようになりました。必要な製品インスタンスを SSM オンプレミスデータベースにまだ追加していない場合は、RUM レポートを送信する前に追加する必要があります。スマートアカウントとバーチャルアカウントの割り当て (SSM オンプレミス UI) (194 ページ) を参照してください。

製品インスタンス開始型通信のネットワーク到達可能性の確認

このタスクでは、製品インスタンス開始型通信のネットワーク到達可能性を確認するために必要になる可能性のある設定を提供します。「(必須)」と付いている手順は、すべての製品インスタンスで必須です。他のすべての手順は、製品インスタンスの種類とネットワーク要件に応じて、必須の場合も任意の場合もあります。該当するコマンドを設定します。



- (注) 手順 13、14、および 15 では、必ず次のように設定してください。これらのコマンドは、正しいトラストポイントが使用され、ネットワーク到達可能性に必要な証明書が受け入れられるように設定する必要があります。

始める前に

サポートされているトポロジ：SSM オンプレミス展開（製品スタンス開始型通信）。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-type-number 例： Device (config)# interface gigabitethernet0/0	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、VRF に関連付けられたイーサネットインターフェイス、サブインターフェイス、または VLAN を指定します。
ステップ 4	vrf forwarding vrf-name 例： Device(config-if)# vrf forwarding Mgmt-vrf	VRF をレイヤ 3 インターフェイスに対応付けます。このコマンドは、インターフェイスでマルチプロトコル VRF をアクティブにします。
ステップ 5	ip address ip-address mask 例： Device(config-if)# ip address 192.168.0.1 255.255.0.0	VRF の IP アドレスを定義します。

	コマンドまたはアクション	目的
ステップ 6	negotiation auto 例： Device(config-if)# negotiation auto	インターフェイスの速度およびデュプレックスパラメータの自動ネゴシエーション動作を有効にします。
ステップ 7	end 例： Device(config-if)# end	インターフェイス 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を開始します。
ステップ 8	ip http client source-interface interface-type-number 例： Device(config)# ip http client source-interface gigabitethernet0/0	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。
ステップ 9	ip route ip-address ip-mask subnet mask 例： Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1	(必須) 製品インスタンスにルートとゲートウェイを設定します。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 10	{ip ipv6} name-server server-address 1 ...server-address 6] 例： Device(config)# Device(config)# ip name-server vrf mgmt-vrf 198.51.100.1	VRF インターフェイスでドメインネームシステム (DNS) を設定します。
ステップ 11	ip domain lookup source-interface interface-type-number 例： Device(config)# ip domain lookup source-interface gigabitethernet0/0	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 12	ip domain name domain-name 例： Device(config)# ip domain name example.com	ドメインの DNS ディスカバリーを設定します。この例では、ネームサーバがエントリ <code>cslu-local.example.com</code> を作成します。
ステップ 13	crypto pki trustpoint SLA-TrustPoint 例： Device(config)# crypto pki trustpoint SLA-TrustPoint Device(ca-trustpoint)#	(必須) 製品インスタンスがトランスポイント「SLA-TrustPoint」を使用する必要があることを宣言し、CA トランスポイント コンフィギュレーションモードを開始します。このコマンドを使用してトランスポイントを宣言する

	コマンドまたはアクション	目的
		まで、製品インスタンスはトラストポイントを認識しません。
ステップ 14	enrollment terminal 例： Device(ca-trustpoint)# enrollment terminal	(必須) 証明書登録方式を指定します。
ステップ 15	revocation-check none 例： Device(ca-trustpoint)# revocation-check none	(必須) ピアの証明書が失効していないことを確認するために使用する方法を指定します。SSM オンプレミス展開トポロジの場合は、 none キーワードを入力します。つまり、失効チェックは実行されず、証明書は常に受け入れられます。
ステップ 16	exit 例： Device(ca-trustpoint)# exit Device(config)# exit	CA トランスポイントコンフィギュレーションモードを終了し、次にグローバルコンフィギュレーションモードを終了してから、特権 EXEC モードに戻ります。
ステップ 17	copy running-config startup-config 例： Device# copy running-config startup-config	コンフィギュレーションファイルに設定を保存します。

トランスポート URL の取得 (SSM オンプレミス UI)

製品インスタンス開始型通信を SSM オンプレミス展開で展開するときに、製品インスタンスでトランスポート URL を設定する必要があります。このタスクでは、テナント ID を含む完全な URL を SSM オンプレミスから簡単にコピーする方法を示します。

始める前に

サポートされているトポロジ：SSM オンプレミス展開（製品スタンス開始型通信）。

手順

- ステップ 1 SSM オンプレミスにログインし、[Smart Licensing] ワークスペースを選択します。
- ステップ 2 [Inventory] タブに移動し、ローカルバーチャルアカウントのドロップダウンリスト（右上隅）から、デフォルトのローカルバーチャルアカウントを選択します。この場合、[Inventory] タブの下の領域に [Local Virtual Account: Default] が表示されます。

ステップ 3 [General] タブに移動します。

[Product Instance Registration Tokens] 領域が表示されます。

ステップ 4 [Product Instance Registration Tokens] 領域で、[CSLU Transport URL] をクリックします。

[Product Registration URL] ポップアップウィンドウが表示されます。

ステップ 5 URL 全体をコピーし、アクセス可能な場所に保存します。

製品インスタンスでトランスポートタイプと URL を設定するときに、この URL が必要になります。

ステップ 6 トランスポートタイプと URL を設定します。[転送タイプ、URL、およびレポート間隔の設定 \(215 ページ\)](#) を参照してください。

使用状況データのエクスポートとインポート (SSM オンプレミス UI)

SSM オンプレミスが CSSM から切断されている場合は、この手順を使用して SSM オンプレミスと CSSM との間で使用状況の同期を実行できます。

始める前に

サポートされているトポロジ :

- SSM オンプレミス展開 (SSM オンプレミス開始型通信)
- SSM オンプレミス展開 (製品インスタンス開始型通信)。

レポートデータは、SSM オンプレミスで使用できる必要があります。必要なレポートデータを製品インスタンスから SSM オンプレミスにプッシュする (製品インスタンス開始型通信) か、または必要なレポートデータを製品インスタンスから取得する (SSM オンプレミス開始型通信) 必要があります。

手順

ステップ 1 SSM オンプレミスにログインし、[Smart Licensing] を選択します。

ステップ 2 [Inventory] > [SL Using Policy] タブに移動します。

ステップ 3 [SL Using Policy] タブ領域で、[Export/Import All ...] > [Export Usage to Cisco] をクリックします。

これにより、SSM オンプレミスサーバで使用可能なすべての使用状況レポートを含む .tar ファイルが 1 つ生成されます。

ステップ 4 CSSM で [CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) のタスクを実行します。

このタスクの最後に、SSM オンプレミスにインポートする ACK ファイルを取得します。

ステップ5 再度、[Inventory] > [SL Using Policy] タブに移動します。

ステップ6 [SL Using Policy] タブ領域で、[Export/Import All ...] > [Import From Cisco] をクリックします。tar ACK ファイルをアップロードします。

ACK インポートを確認するには、[SL Using Policy] タブ領域で、対応する製品インスタンスの [Alerts] 列を確認します。「Acknowledgmentreceived from CSSM」というメッセージが表示されます。

1つ以上の製品インスタンスの追加 (SSM オンプレミス UI)

次の手順を使用して、1つの製品インスタンスを追加したり、複数の製品インスタンスをインポートして追加したりできます。これにより、SSM オンプレミスは製品インスタンスから情報を取得できるようになります。

始める前に

サポートされているトポロジ：SSM オンプレミス展開 (SSM オンプレミス開始型通信)。

手順

ステップ1 SSM オンプレミス UI にログインし、[Smart Licensing] をクリックします。

ステップ2 [Inventory] タブに移動します。右上隅にあるドロップダウンリストからローカルバーチャルアカウントを選択します。

ステップ3 [SL Using Policy] に移動します。

ステップ4 単一の製品インスタンスを追加するか、または複数の製品インスタンスをインポートします (いずれかを選択します)。

• 単一の製品インスタンスを追加するには、次の手順を実行します。

1. [SL Using Policy] タブ領域で、[Add Single Product] をクリックします。
2. [Host] フィールドにホストの IP アドレスを入力します (製品インスタンス)。
3. [Connect Method] ドロップダウンリストから、適切な SSM オンプレミス開始型の接続方式を選択します。

SSM オンプレミス開始型通信に使用できる接続方法は、NETCONF、RESTCONF、および REST API です。

4. 右側のパネルで、[Product Instance Login Credentials] をクリックします。

[Product Instance Login Credentials] ウィンドウが表示されます。

(注) 製品インスタンスに SLAC が必要な場合は、ログインクレデンシャルのみが必要です。

5. [User ID] と [Password] に入力し、[Save] をクリックします。

これは、ネットワーク到達可能性を確立するために必要なコマンドの一部として設定したものと同一ユーザ ID とパスワードです（[SSM オンプレミス開始型通信のネットワーク到達可能性の確保 \(201 ページ\)](#)）。

検証が完了すると、製品インスタンスが [SL Using Policy] タブ領域のリストに表示されます。

- 複数の製品インスタンスをインポートするには、次の手順を実行します。

1. [SL Using Policy] タブで、[Export/Import All ...] > [Import Product Instances List] をクリックします。

[Upload Product Instances] ウィンドウが表示されます。

2. [Download] をクリックし、事前に定義した .csv テンプレートをダウンロードします。

3. .csv テンプレートのすべての製品インスタンスに必要な情報を入力します。

テンプレートで、すべての製品インスタンスの [Host]、[Connect Method]、および [Login Credentials] を必ず指定してください。

SSM オンプレミス開始型通信に使用できる接続方法は、NETCONF、RESTCONF、および REST API です。

ログインクレデンシャルは、ネットワーク到達可能性を確立するために必要なコマンドの一部として設定したユーザ ID とパスワードを参照します（[SSM オンプレミス開始型通信のネットワーク到達可能性の確保 \(201 ページ\)](#)）。

4. 再度、[Inventory] > [SL Using Policy] タブに移動します。[Export/Import All...] > [Import Product Instances List] をクリックします。

[Upload Product Instances] ウィンドウが表示されます。

5. 次に、入力した .csv テンプレートをアップロードします。

検証されると、製品インスタンスが [SL Using Policy] タブのリストに表示されます。

SSM オンプレミス開始型通信のネットワーク到達可能性の確保

このタスクでは、SSM オンプレミス開始型通信のネットワーク到達可能性を確保するために必要になる可能性のある設定を実行します。「(必須)」と付いている手順は、すべての製品インスタンスで必須です。他のすべての手順は、製品インスタンスの種類とネットワーク要件に応じて、必須の場合も任意の場合もあります。該当するコマンドを設定します。



- (注) 手順 25、26、および 27 では、必ず次のように設定してください。これらのコマンドは、正しいトラストポイントが使用され、ネットワーク到達可能性に必要な証明書が受け入れられるように設定する必要があります。

始める前に

サポートされているトポロジ：SSM オンプレミス展開（SSM オンプレミス開始型通信）。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new model 例： Device(config)# aaa new model	(必須) 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 例： Device(config)# aaa authentication login default local	(必須) 認証時にローカルのユーザ名データベースを使用するように、AAA 認証を設定します。
ステップ 5	aaa authorization exec default local 例： Device(config)# aaa authorization exec default local	ネットワークへのユーザアクセスを制限するパラメータを設定します。ユーザは EXEC シェルの実行が許可されません。
ステップ 6	ip routing 例： Device(config)# ip routing	IP ルーティングを有効にします。
ステップ 7	{ip ipv6} name-server server-address 1 ...server-address 6] 例： Device(config)# ip name-server vrf Mgmt-vrf 192.168.1.100 192.168.1.200 192.168.1.300	(任意) 名前とアドレスの解決に使用する 1 つまたは複数のネームサーバのアドレスを指定します。 最大 6 つのネームサーバを指定できます。各サーバアドレスはスペースで区切ります。最初に指定されたサーバ

	コマンドまたはアクション	目的
		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。
ステップ 8	ip domain lookup source-interface interface-type-number 例： Device(config)# ip domain lookup source-interface gigabitethernet0/0	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有効にします。この機能は、デフォルトでイネーブルにされています。 ユーザのネットワークデバイスが、名前の割り当てを制御できないネットワーク内のデバイスと接続する必要がある場合、グローバルなインターネットのネーミング方式（DNS）を使用して、ユーザのデバイスを一意に識別するデバイス名を動的に割り当てることができます。
ステップ 9	ip domain name name 例： Device(config)# ip domain name vrf Mgmt-vrf 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。
ステップ 10	no username name 例： Device(config)# no username admin	（必須）指定されたユーザ名が存在する場合はクリアします。 <i>name</i> には、次のステップで作成するユーザ名と同じものを入力します。これにより、次のステップで作成するユーザ名が重複していないことが保証されます。 SSM オンプレミス開始型の RUM レポートを取得に REST API を使用する場合は、SSM オンプレミスにログインする必要があります。ユーザ名が重複していると、システムにそのユーザ名がある場合はこの機能が正しく動作しない場合があります。
ステップ 11	username name privilege level password password 例：	（必須）ユーザ名をベースとした認証システムを構築します。 privilege キーワードにより、ユーザの権限レベルを設定します。ユーザの権

	コマンドまたはアクション	目的
	<pre>Device(config)# username admin privilege 15 password 0 lab</pre>	<p>限レベルを指定する 0 ～ 15 の数字です。</p> <p><code>password</code> を使用すると、<code>name</code> 引数にアクセスできます。パスワードは 1 ～ 25 文字で、埋め込みスペースを使用でき、<code>username</code> コマンドの最後のオプションとして指定します。</p> <p>これにより、SSM オンプレミスが製品インスタンスのネイティブ REST を使用できるようになります。</p> <p>(注) このユーザ名とパスワードを SSM オンプレミスに入力します (1 つ以上の製品インスタンスの追加 (SSM オンプレミス UI) (200 ページ))。これにより、SSM オンプレミスは製品インスタンスから RUM レポートを収集できるようになります。</p>
ステップ 12	<p>interface <i>interface-type-number</i></p> <p>例 :</p> <pre>Device (config)# interface gigabitethernet0/0</pre>	インターフェイス コンフィギュレーションモードを開始し、VRF に関連付けられたイーサネットインターフェイス、サブインターフェイス、または VLAN を指定します。
ステップ 13	<p>vrf forwarding <i>vrf-name</i></p> <p>例 :</p> <pre>Device(config-if)# vrf forwarding Mgmt-vrf</pre>	VRF をレイヤ 3 インターフェイスに対応付けます。このコマンドは、インターフェイスでマルチプロトコル VRF をアクティブにします。
ステップ 14	<p>ip address <i>ip-address mask</i></p> <p>例 :</p> <pre>Device(config-if)# ip address 192.168.0.1 255.255.0.0</pre>	VRF の IP アドレスを定義します。
ステップ 15	<p>negotiation auto</p> <p>例 :</p> <pre>Device(config-if)# negotiation auto</pre>	インターフェイスの速度およびデュプレックスパラメータの自動ネゴシエーション動作を有効にします。
ステップ 16	<p>no shutdown</p> <p>例 :</p>	無効にされたインターフェイスを再起動します。

	コマンドまたはアクション	目的
	Device (config-if) # no shutdown	
ステップ 17	end 例 : Device (config-if) # end	インターフェイス 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を開始します。
ステップ 18	ip http server 例 : Device (config) # ip http server	(必須) シスコの Web ブラウザ ユーザ インターフェイスを含む IP または IPv6 システムで HTTP サーバを有効にします。HTTP サーバは、デフォルトにより標準のポート 80 を使用します。
ステップ 19	ip http authentication local 例 : ip http authentication local Device (config) #	(必須) HTTP サーバユーザに対して特定の認証方法を指定します。 local キーワードは、認証および許可に、ローカルシステム設定で (username グローバルコンフィギュレーションコマンドによって) 指定したログイン ユーザ名、パスワード、権限レベルアクセスの組み合わせを使用することを示します。
ステップ 20	ip http secure-server 例 : Device (config) # ip http server	(必須) セキュア HTTP (HTTPS) サーバを有効にします。HTTPS サーバは、セキュアソケットレイヤ (SSL) バージョン 3.0 プロトコルを使用します。
ステップ 21	ip http max-connections 例 : Device (config) # ip http max-connections 16	(必須) HTTP サーバへの同時最大接続数を設定します。1 ~ 16 の範囲の整数を入力します。デフォルトは 5 です。
ステップ 22	ip tftp source-interface interface-type-number 例 : Device (config) # ip tftp source-interface GigabitEthernet0/0	TFTP 接続用の送信元アドレスとして、インターフェイスの IP アドレスを指定します。
ステップ 23	ip route ip-address ip-mask subnet mask 例 : Device (config) # ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1	製品インスタンスにルートとゲートウェイを設定します。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。

	コマンドまたはアクション	目的
ステップ 24	logging host 例 : <pre>Device(config)# logging host 172.25.33.20 vrf Mgmt-vrf</pre>	リモートホストへのシステムメッセージおよびデバッグ出力を記録します。
ステップ 25	crypto pki trustpoint SLA-TrustPoint 例 : <pre>Device(config)# crypto pki trustpoint SLA-TrustPoint Device(ca-trustpoint)#</pre>	(必須) 製品インスタンスがトランスポイント「SLA-TrustPoint」を使用する必要があることを宣言し、CA トランスポイント コンフィギュレーション モードを開始します。このコマンドを使用してトランスポイントを宣言するまで、製品インスタンスはトランスポイントを認識しません。
ステップ 26	enrollment terminal 例 : <pre>Device(ca-trustpoint)# enrollment terminal</pre>	(必須) 証明書登録方式を指定します。
ステップ 27	revocation-check none 例 : <pre>Device(ca-trustpoint)# revocation-check none</pre>	(必須) ピアの証明書が失効していないことを確認するために使用する方法を指定します。SSM オンプレミス展開 トポロジの場合は、 none キーワードを入力します。つまり、失効チェックは実行されず、証明書は常に受け入れられます。
ステップ 28	end 例 : <pre>Device(ca-trustpoint)# exit Device(config)# end</pre>	CA トランスポイント コンフィギュレーション モードを終了し、次にグローバル コンフィギュレーション モードを終了してから、特権 EXEC モードに戻ります。
ステップ 29	show ip http server session-module 例 : <pre>Device# show ip http server session-module</pre>	(必須) HTTP 接続を確認します。出力で、 SL_HTTP がアクティブであることを確認します。また、次のチェックも実行できます。 <ul style="list-style-type: none"> SSM オンプレミスがインストールされているデバイスから、製品インスタンスに ping できることを確認します。ping が成功すると、製品インスタンスが到達可能であることが確認されます

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> SSM オンプレミスがインストールされているデバイスの Web ブラウザで、 https://<product-instance-ip>/ を確認します。これにより、SSM オンプレミスから製品インスタンスへの REST API が期待どおりに動作することが保証されます。
ステップ 30	copy running-config startup-config 例： Device# copy running-config startup-config	コンフィギュレーションファイルに設定を保存します。

承認コードの削除と返却

SLR 承認コードを削除して返却するには、次の手順を実行します。

始める前に

サポートされるトポロジ：すべて

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	show license summary 例： Device# show license summary	削除して返却するライセンスが使用中でないことを確認します。使用中の場合は、まず機能を無効にする必要があります。
ステップ 3	license smart authorization return {all local} {offline [path] online} 例： Device# license smart authorization return local online OR Device# license smart authorization return local offline Enter this return code in Cisco Smart Software Manager portal:	CSSM のライセンスプールに承認コードを返却します。このコマンドを入力すると、戻りコードが表示されます。 製品インスタンスを指定します。 <ul style="list-style-type: none"> all：高可用性セットアップで接続されたすべての製品インスタンスに対してアクションを実行します。

	コマンドまたはアクション	目的
	UDI: PID:C9500-16X,SN:FCW2233A5ZV Return code: Cx9Ukx-LlxSRj-ftwzjl-h9QZAU-IE5DML-hakVdL-FAEPT9-wdIn7-Hp7	<ul style="list-style-type: none"> • local : アクティブな製品インスタンスに対してアクションを実行します。これがデフォルトのオプションです。 <p>CSSMに接続しているかどうかを指定します。</p> <ul style="list-style-type: none"> • CSSMに接続している場合は、onlineを入力します。コードは自動的にCSSMに返却され、確認が返されて製品インスタンスにインストールされます。このオプションを選択すると、戻りコードが自動的にCSSMに送信されます。 • CSSMに接続していない場合は、offline [<i>filepath_filename</i>]を入力します。 <p>ファイル名とパスを指定しない場合は、CLIにリターンコードが表示されます。ファイル名とパスを指定すると、リターンコードは指定した場所に保存されます。ファイル形式は、読み取り可能な任意の形式にすることができます。例: Device#</p> <pre>license smart authorization return local offline bootflash:return-code.txt</pre> <p>offline オプションを選択する場合は、CLIや保存したファイルから戻りコードをコピーしてCSSMに入力する、という追加の手順を実行する必要があります。CSSMからの製品インスタンスの削除 (209 ページ) を参照してください。この手順を完了してから、次の手順に進みます。</p>
ステップ 4	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>

	コマンドまたはアクション	目的
ステップ 5	<p>no license smart reservation</p> <p>例 :</p> <pre>Device(config)# no license smart reservation</pre>	<p>製品インスタンスの SLR 設定を無効にします。</p> <p>(注) この手順で no license smart reservation コマンドを入力する前に、上記の手順 3 で (オンラインまたはオフラインで) 承認コードの返却プロセスを完了する必要があります。そうしないと、返却が CSSM または show コマンドに反映されない場合があります。問題を修正するには、シスコのテクニカルサポート担当者に連絡する必要があります。</p>
ステップ 6	<p>exit</p> <p>例 :</p> <pre>Device(config)# exit</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 7	<p>show license all</p> <p>例 :</p> <pre>Device# show license all <output truncated> License Authorizations ===== Overall status: Active: PID:C9500-16X,SN:FCW2233A5ZV Status: NOT INSTALLED Last return code: C9500-16X-FCW2233A5ZV-FCW2233A5ZV-FCW2233A5ZV-FCW2233A5ZV <output truncated></pre>	<p>ライセンス情報を表示します。出力の License Authorizations ヘッダーを確認します。返却プロセスが正常に完了すると、Last return code: フィールドに戻りコードが表示されます。</p>

CSSM からの製品インスタンスの削除

製品インスタンスを削除し、すべてのライセンスをライセンスプールに戻すには、次のタスクを実行します。

始める前に

サポートされるトポロジ: すべて

予約済みライセンス (SLR) を使用している製品インスタンスを削除する場合は、[承認コードの削除と返却 \(207 ページ\)](#) に示されているとおり、リターンコードが生成されていることを確認します。(このタスクの手順 7 で入力します)。

手順

- ステップ 1** <https://software.cisco.com> で CSSM Web UI にログインし、[Smart Software Licensing] をクリックします。
- 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ログインします。
- 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、バーチャルアカウントを選択します。
- ステップ 4** [Product Instances] タブをクリックします。
- 使用可能な製品インスタンスのリストが表示されます。
- ステップ 5** 製品インスタンスリストから必要な製品インスタンスを見つけます。オプションで、検索タブに名前または製品タイプの文字列を入力して、製品インスタンスを検索できます。
- ステップ 6** 削除する製品インスタンスの [Actions] 列で、[Remove] リンクをクリックします。
- 製品インスタンスが SLR 承認コードを含むライセンスを使用していない場合は、[Confirm Remove Product Instance] ウィンドウが表示されます。
 - 製品インスタンスが SLR 承認コードを含むライセンスを使用している場合は、リターンコードを入力するためのフィールドのある [Remove Product Instance] ウィンドウが表示されます。
- ステップ 7** [Reservation Return Code] フィールドに、作成したリターンコードを入力します。
- (注) この手順は、製品インスタンスが SLR 承認コードを含むライセンスを使用している場合にのみ適用されます。
- ステップ 8** [Remove Product Instance] をクリックします。
- ライセンスがライセンスプールに返され、製品インスタンスが削除されます。
-

CSSM からの信頼コード用新規トークンの生成

信頼コードを要求するトークンを生成するには、次の手順を実行します。

所有するバーチャルアカウントごとに 1 つのトークンを生成します。1 つのバーチャルアカウントに属するすべての製品インスタンスに同じトークンを使用できます。

始める前に

サポートされるトポロジ：CSSM に直接接続

手順

- ステップ 1 <https://software.cisco.com> で CSSM Web UI にログインし、[Smart Software Licensing] をクリックします。
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ログインします。
- 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 [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 (オプション) [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 [Create Token] をクリックします。
- ステップ 10 リストに新しいトークンが表示されます。[Actions] をクリックし、トークンを .txt ファイルとしてダウンロードします。

信頼コードのインストール

信頼コードを手動でインストールするには、次の手順を実行します。

始める前に

サポートされるトポロジ:

- CSSM に直接接続

手順

	コマンドまたはアクション	目的
ステップ 1	CSSMからの信頼コード用新規トークンの生成 (210 ページ)	まだ CSSM から信頼コードファイルを生成してダウンロードしていない場合は、生成とダウンロードを実行します。
ステップ 2	enable 例: Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 3	<p>license smart trust idtoken <i>id_token_value</i> { local all } [force]</p> <p>例 :</p> <pre>Device# license smart trust idtoken NGMwMjk5mYtNZaxMS00NzZmtgWm all force</pre>	<p>CSSM との信頼できる接続を確立できません。 <i>id_token_value</i> には、CSSM で生成したトークンを入力します。</p> <p>次のいずれかのオプションを入力します。</p> <ul style="list-style-type: none"> • local : 高可用性セットアップのアクティブデバイスに対してのみ信頼要求を送信します。これがデフォルトのオプションです。 • all : 高可用性セットアップのすべてのデバイスに対して信頼要求を送信します。 <p>製品インスタンスに既存の信頼コードがあるにもかかわらず、信頼コード要求を送信するには、force キーワードを入力します。</p> <p>信頼コードは、製品インスタンスの UDI にノードロックされます。UDI がすでに登録されている場合、CSSM は同じ UDI の新規登録を許可しません。force キーワードを入力すると、CSSM に送信されるメッセージに強制フラグが設定され、すでに存在する場合でも新しい信頼コードが作成されます。</p>
ステップ 4	<p>show license status</p> <p>例 :</p> <pre><output truncated> Trust Code Installed: Active: PID:C9500-24Y4C,SN:CAT2344L4GH INSTALLED on Sep 04 01:01:46 2020 EDT Standby: PID:C9500-24Y4C,SN:CAT2344L4GJ INSTALLED on Sep 04 01:01:46 2020 EDT</pre>	<p>信頼コードがインストールされている場合は、日時が表示されます。日時はローカルタイムゾーンで表示されます。Trust Code Installed: フィールドを参照してください。</p>

CSSM からのポリシーファイルのダウンロード

カスタムポリシーを要求した場合、または製品インスタンスに適用されるデフォルトとは異なるポリシーを適用する場合は、次のタスクを実行します。

始める前に

サポートされるトポロジ:

- CSSM への接続なし、CSLU なし
- CSLU は CSSM から切断

手順

ステップ 1 <https://software.cisco.com> で CSSM Web UI にログインし、[Smart Software Licensing] をクリックします。

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ログインします。

ステップ 2 次のディレクトリパス、[Reports] > [Reporting Policy] を移動します。

ステップ 3 [Download] をクリックして、.xml ポリシーファイルを保存します。

これで、ファイルを製品インスタンスにインストールできます。[製品インスタンスへのファイルのインストール \(214 ページ\)](#) を参照してください。

CSSM への使用状況データのアップロードと ACK のダウンロード

製品インスタンスが CSSM や CSLU に接続されていない場合、または SSM オンプレミスが CSSM に接続されていない場合に RUM レポートを CSSM にアップロードして ACK をダウンロードするには、次のタスクを実行します。

始める前に

サポートされるトポロジ:

- CSSM への接続なし、CSLU なし
- SSM オンプレミス展開 (製品インスタンス開始型通信と SSM オンプレミス開始型通信)

手順

ステップ 1 <https://software.cisco.com> で CSSM Web UI 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ログインします。

ステップ 2 レポートを受信するスマートアカウント (画面の左上隅) を選択します。

ステップ 3 [Smart Software Licensing] → [Reports] → [Usage Data Files] を選択します。

ステップ 4 [Upload Usage Data] をクリックします。ファイルの場所 (tar 形式の RUM レポート) を参照して選択し、[Upload Data] をクリックします。

使用状況レポートは、アップロード後に CSSM で削除できません。

ステップ 5 [Select Virtual Accounts] ポップアップから、アップロードされたファイルを受信する**バーチャルアカウント**を選択します。ファイルが CSSM にアップロードされ、[Usage Data Files] タブエリアにファイル名、レポートの時刻、アップロード先のバーチャルアカウント、レポートステータス、レポートされた製品インスタンス数、確認ステータスとともに表示されます。

ステップ 6 [Acknowledgment] 列で [Download] をクリックして、アップロードしたレポートの .txt ACK ファイルを保存します。

[Acknowledgment] 列に「ACK」が表示されるまで待ちます。処理する RUM レポートが多数ある場合、CSSM では数分かかることがあります。

実装したトポロジに応じて、ファイルを製品インスタンスにインストールするか、または CSLU に転送する、あるいは SSM オンプレミスにインポートすることができます。

製品インスタンスへのファイルのインストール

製品インスタンスが CSSM または CSLU に接続されていない場合に、製品インスタンスに SLAC、ポリシー、ACK、またはトークンをインストールするには、次のタスクを実行します。

始める前に

サポートされるトポロジ：CSSM への接続なし、CSLU なし

製品インスタンスにアクセスできる場所に、対応するファイルを保存しておく必要があります。

- ポリシーの場合の参照：[CSSM からのポリシーファイルのダウンロード \(212 ページ\)](#)
- ACK の場合の参照：[CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#)

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	copy source bootflash:file-name 例： Device# copy tftp://10.8.0.6/bootflash:example.txt	ファイルをソースの場所またはディレクトリから製品インスタンスのフラッシュメモリにコピーします。 • <i>source</i> : これは、コピー元となるファイルまたはディレクトリの場所

	コマンドまたはアクション	目的
		<p>です。コピー元は、ローカルまたはリモートのいずれかです。</p> <ul style="list-style-type: none"> • bootflash : これはブートフラッシュメモリの場合の宛先です。
ステップ 3	<p>license smart import bootflash: file-name</p> <p>例 :</p> <pre>Device# license smart import bootflash:example.txt</pre>	<p>ファイルを製品インスタンスにインポートしてインストールします。インストール後、インストールしたファイルのタイプを示すシステムメッセージが表示されます。</p>
ステップ 4	<p>show license all</p> <p>例 :</p> <pre>Device# show license all</pre>	<p>製品インスタンスのライセンス承認、ポリシー、およびレポート情報を表示します。</p>

転送タイプ、URL、およびレポート間隔の設定

製品インスタンスの転送モードを設定するには、次のタスクを実行します。

始める前に

サポートされるトポロジ : すべて

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。</p>
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	
ステップ 3	<p>license smart transport {automatic callhome cslu off smart}</p> <p>例 :</p> <pre>Device(config)# license smart transport cslu</pre>	<p>使用する製品インスタンスの転送モードを設定します。次のオプションから選択します。</p> <ul style="list-style-type: none"> • automatic : 転送モード cslu を設定します。 • callhome : 転送モードとして Call Home を有効にします。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • cslu : これがデフォルトのトランスポートモードです。製品インスタンス開始型通信で CSLU または SSM オンプレミスを使用している場合は、このキーワードを入力します。 トランスポートモードキーワードは CSLU と SSM オンプレミスで同じですが、トランスポート URL は異なります。次の手順の license smart url cslu cslu_or_on-prem_url を参照してください。 • off : 製品インスタンスからのすべての通信を無効にします。 • smart : スマート転送を有効にします。
ステップ 4	<p>license smart url {<i>url</i> cslu <i>cslu_url</i> default smart <i>smart_url</i> utility <i>smart_url</i>}</p> <p>例 :</p> <pre>Device(config)# license smart url cslu http://192.168.0.1:8182/cslu/v1/pi</pre>	<p>設定された転送モードの URL を設定します。前の手順で選択した転送モードに応じて、対応する URL をここで設定します。</p> <ul style="list-style-type: none"> • url : 転送モードとして callhome を設定している場合は、このオプションを設定します。CSSM URL を次のように正確に入力します。 <code>https://software.cisco.com/#module/StartLicensing</code> • no license smart url url コマンドは、デフォルトの URL に戻ります。 • cslu cslu_or_on-prem_url : トランスポートモードを cslu として設定している場合は、必要に応じて CSLU または SSM オンプレミスの URL を使用してこのオプションを設定します。 • CSLU を使用している場合は、次のように URL を入力します。 <code>http://<cslu_ip_or_host>:8182/cslu/v1/pi</code> <cslu_ip_or_host>には、CSLU をインストールした Windows

	コマンドまたはアクション	目的
		<p>ホストのホスト名や IP アドレスを入力します。8182 はポート番号であり、CSLU が使用する唯一のポート番号です。</p> <p>no license smart url cslu cslu_url コマンドは <code>http://cslu-local:8182/cslu/v1/pi</code> に戻ります</p> <ul style="list-style-type: none"> • SSM オンプレミスを使用している場合は、次のように URL を入力します。 <code>http://<ip>/cslu/v1/pi/<tenant ID></code> <p><ip> には、SSM オンプレミスをインストールしたサーバのホスト名または IP アドレスを入力します。<tenantID> はデフォルトのローカルバーチャルアカウント ID にする必要があります。</p> <p>ヒント SSM オンプレミスから URL 全体を取得できます。「トランスポート URL の取得 (SSM オンプレミス UI) (198 ページ)」を参照してください</p> <p>no license smart url cslu cslu_url コマンドは <code>http://cslu-local:8182/cslu/v1/pi</code> に戻ります</p> <ul style="list-style-type: none"> • default : 設定されている転送モードによって異なります。このオプションでは、smart および cslu 転送モードのみがサポートされます。 <p>転送モードが cslu に設定されている場合、license smart url default を設定すると、CSLU URL は自動的に</p>

	コマンドまたはアクション	目的
		<p>設定されます (https://cslu-local:8182/cslu/v1/pi)。</p> <p>転送モードが smart に設定されている場合、license smart url default を設定すると、スマート URL は自動的に設定されます (https://smartreceiver.cisco.com/liceservice/license)。</p> <ul style="list-style-type: none"> • smart smart_url : 転送タイプとして smart を設定している場合は、このオプションを設定します。URL を次のように正確に入力します。 https://smartreceiver.cisco.com/liceservice/license このオプションを設定すると、システムは license smart url url で自動的に URL の複製を作成します。重複するエントリは無視できます。これ以上の操作は必要ありません。 <p>no license smart url smartsmart_url コマンドは、デフォルトの URL に戻ります。</p> <ul style="list-style-type: none"> • utility smart_url : このオプションは CLI では使用できますがサポートされていません。
<p>ステップ 5</p>	<p>license smart usage interval interval_in_days</p> <p>例 :</p> <pre>Device(config)# license smart usage interval 40</pre>	<p>(任意) レポート間隔の日数を設定します。デフォルトでは、RUM レポートは 30 日ごとに送信されます。有効な値の範囲は 1 ~ 3650 です。</p> <p>この値をゼロに設定すると、適用されるポリシーの指定内容に関係なく、RUM レポートは送信されません。これは、CSLU または CSSM が受信側にある可能性があるトポロジに適用されます。</p> <p>ゼロより大きい値を設定し、通信タイプが オフ に設定されている場合、interval_in_days と <code>Ongoing reporting frequency (days)</code> : のポリシー値の間で、値の小さい方が適用されます。たとえば、interval_in_days が 100 に設定され、ポリシーの値が <code>Ongoing reporting</code></p>

	コマンドまたはアクション	目的
		<p>frequency (days):90 の場合、RUM レポートは 90 日ごとに送信されます。</p> <p>間隔を設定せず、デフォルトが有効な場合、レポート間隔は完全にポリシー値によって決定されます。たとえば、デフォルト値が有効で、不適用ライセンスのみが使用されている場合、ポリシーでレポートが不要と記述されていると、RUM レポートは送信されません。</p>
ステップ 6	<p>exit</p> <p>例 :</p> <pre>Device(config)# exit</pre>	<p>グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>
ステップ 7	<p>copy running-config startup-config</p> <p>例 :</p> <pre>Device# copy running-config startup-config</pre>	<p>コンフィギュレーション ファイルに設定を保存します。</p>

基本ライセンスまたはアドオンライセンスの設定

基本ライセンスまたはアドオンライセンスを注文および購入したら、使用する前にデバイスでライセンスを設定する必要があります。

このタスクではライセンスレベルを設定します。設定された変更を有効にする前にリロードが必要です。このタスクを使用して、製品インスタンスの現在のライセンスを変更したり、別のライセンスを追加したり（たとえば、現在 Network Advantage を使用している場合、対応する Digital Networking Architecture (DNA) Advantage ライセンスで使用可能な機能も使用することができます）、またはライセンスを削除したりすることができます。

始める前に

サポートされるトポロジ：すべて

次のブートレベルライセンスを使用できます。

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。

- DNA Essentials

- DNA Advantage (DNA Essentials を含む)

購入したライセンスに関する情報は、CSSM Web UI の製品インスタンスのスマートアカウントとバーチャルアカウントで確認できます。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	license boot level license_level 例： Device(config)# license boot level network-advantage add-on dna-advantage	製品インスタンスで設定されたライセンスをアクティブにします。この例では、DNA Advantage ライセンスはリロード後に製品インスタンスでアクティブ化されます。
ステップ 4	exit 例： Device(config)# exit	特権 EXEC モードに戻ります。
ステップ 5	copy running-config startup-config 例： Device# copy running-config startup-config	構成ファイルへの変更を保存します。
ステップ 6	show version 例： Device# show version <output truncated> Technology Package License Information: Technology-package Technology-package Current Type Next reboot network-advantage Smart License network-advantage Subscription Smart License dna-advantage <output truncated>	現在設定されているライセンスの情報と、リロード後に適用可能なライセンスを表示します。

	コマンドまたはアクション	目的
ステップ 7	reload 例： Device# reload	デバイスがリロードされます。

次のタスク

ライセンスレベルを設定すると、変更はリロード後に有効になります。レポートが必要かどうかを確認するには、**show license status** 特権EXECコマンドの出力を参照し、Next ACK deadline: フィールドと Next report push: フィールドを確認します。



(注) ライセンスの使用状況の変更は、製品インスタンスに記録されます。レポートに関連した次の手順は、必要に応じて実行しますが、現在のトポロジによって異なります。

- CSLU を介して CSSM に接続
 - 製品インスタンス開始型通信：アクションは不要です。製品インスタンスは通信を開始すると、ポリシー (**show license status** → [Next report push]) に従って、スケジュールされた時刻に最初の RUM レポートを自動的に CSLU に送信します。(製品インスタンスでこれを手動でトリガーするには、**license smart syncallocal** 特権EXECコマンドを入力します。これにより、CSLU と製品インスタンスが同期され、保留中のデータが送受信されます)。CSLU は RUM レポートを CSSM に転送し、ACK を取得します。ACK は、製品インスタンスが次回 CSLU に接続したときに製品インスタンスに適用されます。
 - CSLU 開始型通信：CSLU インターフェイスで製品インスタンスから使用状況を収集します。[使用状況レポートの収集：CSLU 開始 \(179 ページ\)](#) CSLU は RUM レポートを CSSM に送信し、CSSM から ACK を取得します。CSLU が次に更新を実行するときに、ACK が製品インスタンスに適用されます。
- CSSM に直接接続：アクションは必要ありません。製品インスタンスは通信を開始すると、ポリシー (**show license status** → [Next report push]) に従って、スケジュールされた時刻に最初の RUM レポートを自動的に CSSM に送信します。(製品インスタンスでこれを手動でトリガーするには、**license smart syncallocal** 特権EXECコマンドを入力します。これにより、CSSM と製品インスタンスが同期され、保留中のデータが送受信されます)。ACK が使用可能になると、CSSM はこれを製品インスタンスに送り返します。
- CSLU は CSSM から切断
 - 製品インスタンス開始型通信：アクションは不要です。製品インスタンスは通信を開始すると、ポリシー (**show license status** → [Next report push]) に従って、スケジュールされた時刻に最初の RUM レポートを自動的に CSLU に送信します。(製品インスタンスでこれを手動でトリガーするには、**license smart syncallocal** 特権EXECコマンドを入力します。これにより、CSLU と製品インスタンスが同期され、保留中のデータが送受信されます)。

CSLU が CSSM から切断されているため、CSLU インターフェイスと CSSM Web UI でタスク [Download All For Cisco \(CSLU インターフェイス\) \(180 ページ\)](#) > [CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) > [Upload From Cisco \(CSLU インターフェイス\) \(181 ページ\)](#) を実行します。ACK は、製品インスタンスが次回 CSLU に接続したときに製品インスタンスに適用されます。

- CSLU 開始型通信：CSLU インターフェイスで製品インスタンスから使用状況を収集します。[使用状況レポートの収集：CSLU 開始 \(179 ページ\)](#)

CSLU が CSSM から切断されているため、CSLU インターフェイスと CSSM Web UI でタスク [Download All For Cisco \(CSLU インターフェイス\) \(180 ページ\)](#) > [CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) > [Upload From Cisco \(CSLU インターフェイス\) \(181 ページ\)](#) を実行します。CSLU が次に更新を実行するときに、ACK が製品インスタンスに適用されます。

- コントローラを介して CSSM に接続：アクションは必要ありません (Cisco DNA Center GUI で最初のアドホックレポートをすでに完了している場合)。Cisco DNA Center は、後続のすべてのレポートを処理し、製品インスタンスに ACK を返します。
- CSSM への接続なし、CSLU なし：RUM レポートを (製品インスタンスの) ファイルに保存してから、CSSM にアップロードします (インターネットとシスコに接続されているワークステーションから)。**license smart save usage** コマンドを特権 EXEC モードで実行し、RUM レポートをファイルに保存します。次に、CSSM にファイルをアップロードして ACK をダウンロードするため、次のタスクを実行します。[CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) 最後に、製品インスタンスに ACK をインストールするため、次のタスクを実行します。[製品インスタンスへのファイルのインストール \(214 ページ\)](#)
- SSM オンプレミス展開：
 - 製品インスタンス開始型通信：アクションは不要です。製品インスタンスは通信を開始すると、ポリシー (**show license status** → [Next report push]) に従って、スケジュールされた時刻に最初の RUM レポートを自動的に SSM オンプレミスに送信します。(製品インスタンスでこれを手動でトリガーするには、**license smart syncalllocal** 特権 EXEC コマンドを入力します。これにより、SSM オンプレミスと製品インスタンスが同期され、保留中のデータが送受信されます)。
 - SSM オンプレミスが CSSM に接続されている場合、SSM オンプレミス UI で、[Reports] > [Usage Schedules] > [Synchronization schedule with Cisco] に移動します。
 - SSM オンプレミスが CSSM から切断されている場合は、レポートに必要なファイルをアップロードおよびダウンロードします。[使用状況データのエクスポートとインポート \(SSM オンプレミス UI\) \(199 ページ\)](#)
- SSM オンプレミス開始型通信：SSM オンプレミス インターフェイスで、製品インスタンスから使用状況情報を収集します。[Reports] > [Synchronisation pull schedule with the devices] > [Synchronize now with the device] に移動します。

- SSM オンプレミスが CSSM に接続されている場合、SSM オンプレミス UI で、[Reports] > [Usage Schedules] > [Synchronization schedule with Cisco] に移動します。
- SSM オンプレミスが CSSM から切断されている場合は、レポートに必要なファイルをアップロードおよびダウンロードします。[使用状況データのエクスポートとインポート \(SSM オンプレミス UI\)](#) (199 ページ)

リソース使用率測定レポートの例

次に、リソース使用率測定 (RUM) レポートの例を XML 形式で示します ([RUM レポートおよびレポート確認応答 \(118 ページ\)](#) を参照)。このような複数のレポートを連結して1つのレポートを形成できます。

```
<?xml version="1.0" encoding="UTF-8"?>
<smartLicense>
```

```
[Redacted XML content]
```

```
</smartLicense>
```

ポリシーを使用したスマートライセンシングのトラブルシューティング

このセクションでは、発生する可能性のあるポリシーを使用したスマートライセンシングに関するシステムメッセージ、考えられる失敗の理由、および推奨するアクションを示します。

システムメッセージの概要

システムメッセージは、システムソフトウェアからコンソール（および任意で別のシステムのロギングサーバー）に送信されます。すべてのシステムメッセージがシステムの問題を示すわけではありません。通知目的のメッセージもあれば、通信回線、内蔵ハードウェア、またはシステムソフトウェアの問題を診断するうえで役立つメッセージもあります。

システムメッセージの読み方

システムログメッセージには最大 80 文字を含めることができます。各システムメッセージはパーセント記号 (%) から始まります。構成は次のとおりです。

```
%FACILITY-SEVERITY-MNEMONIC: Message-text
```

%FACILITY

メッセージが参照するファシリティを示す2文字以上の大文字です。ファシリティは、ハードウェアデバイス、プロトコル、またはシステムソフトウェアのモジュールなどです。

SEVERITY

0～7の1桁のコードで、状態のシビラティ（重大度）を表します。この値が小さいほど、重大な状況を意味します。

表 16:メッセージのシビラティ（重大度）

シビラティ（重大度）	説明
0：緊急	システムが使用不可能な状態。
1：アラート	ただちに対応が必要な状態。
2：クリティカル	危険な状態。
3：エラー	エラー条件。
4：警告	警告条件。
5：通知	正常だが注意を要する状態。
6：情報	情報メッセージのみ。
7：デバッグ	デバッグ時に限り表示されるメッセージのみ。

MNEMONIC

メッセージを一意に識別するコード。

Message-text

メッセージテキストは、状態を説明したテキスト文字列です。メッセージのこの部分には、端末ポート番号、ネットワークアドレス、またはシステムメモリアドレス空間の位置に対応するアドレスなど、イベントの詳細情報が含まれることがあります。この可変フィールドの情報はメッセージごとに異なるので、ここでは角カッコ ([]) で囲んだ短い文字列で示します。たとえば10進数は [dec] で表します。

表 17:メッセージの変数フィールド

シビラティ（重大度）	説明
[char]	1文字
[chars]	文字列
[dec]	10進数
[enet]	イーサネットアドレス（たとえば 0000.FEED.00C0）
[hex]	16進数
[inet]	インターネットアドレス（10.0.2.16）

シビラティ (重大度)	説明
[int]	整数
[node]	アドレス名またはノード名
[t-line]	8進数のターミナルライン番号 (10進数 TTY サービスが有効な場合は 10進数)
[clock]	クロック (例: 01:20:08 UTC Tue Mar 2 1993)

システムメッセージ

このセクションでは、発生する可能性のあるポリシーを使用したスマートライセンシングに関連するシステムメッセージ、考えられる失敗の理由 (失敗メッセージの場合)、および推奨するアクション (アクションが必要な場合) を示します。

すべてのエラーメッセージについて、問題を解決できない場合は、シスコのテクニカルサポート担当者に次の情報をお知らせください。

コンソールまたはシステムログに出力されたとおりのメッセージ。

show license tech support、**show license history message**、および **show platform software sl-infra** 特権 EXEC コマンドの出力。

ポリシーを使用したスマートライセンシング関連のシステムメッセージ:

- %SMART_LIC-3-POLICY_INSTALL_FAILED
- %SMART_LIC-3-AUTHORIZATION_INSTALL_FAILED
- %SMART_LIC-3-COMM_FAILED
- %SMART_LIC-3-COMM_RESTORED
- %SMART_LIC-3-POLICY_REMOVED
- %SMART_LIC-3-TRUST_CODE_INSTALL_FAILED
- %SMART_LIC-4-REPORTING_NOT_SUPPORTED
- %SMART_LIC-6-POLICY_INSTALL_SUCCESS
- %SMART_LIC-6-AUTHORIZATION_INSTALL_SUCCESS
- %SMART_LIC-6-AUTHORIZATION_REMOVED
- %SMART_LIC-6-REPORTING_REQUIRED
- %SMART_LIC-6-TRUST_CODE_INSTALL_SUCCESS

Error Message %SMART_LIC-3-POLICY_INSTALL_FAILED: The installation of a new licensing policy has failed: [chars].

説明: ポリシーがインストールされましたが、ポリシーコードの解析中にエラーが検出され、インストールに失敗しました。[chars] はエラーの詳細を示すエラー文字列です。

失敗の理由として次が考えられます。

- 署名の不一致：これは、システムクロックが正確でないことを意味します。
- タイムスタンプの不一致：製品インスタンスのシステムクロックがCSSMと同期していないことを意味します。

推奨するアクション：

考えられる両方の失敗の理由に関しては、システムクロックが正確で、CSSMと同期していることを確認します。**ntp server** コマンドをグローバルコンフィギュレーションモードで設定します。次に例を示します。

```
Device(config)# ntp server 198.51.100.100 version 2 prefer
```

前述の手順を実行しても、ポリシーのインストールが失敗する場合は、シスコのテクニカルサポート担当者にお問い合わせください。

```
Error Message %SMART_LIC-3-AUTHORIZATION_INSTALL_FAILED: The install of a new
licensing authorization code has failed on [chars]: [chars].
```

このメッセージは、Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチには該当しません。これらの製品インスタンスには輸出規制ライセンスや適用ライセンスがないためです。

```
Error Message %SMART_LIC-3-COMM_FAILED: Communications failure with the [chars] :
[chars]
```

説明： CSSM、CSLU、または SSM オンプレミスのいずれかとのスマートライセンシング通信が失敗しました。最初の [chars] は現在設定されている転送タイプで、2 番目の [chars] はエラーの詳細を示すエラー文字列です。このメッセージは、失敗した通信の試行ごとに表示されます。

失敗の理由として次が考えられます。

- CSSM、CSLU、または SSM オンプレミスに到達できない：これは、ネットワーク到達可能性に問題があることを意味します。
- 404 ホストが見つからない：これは CSSM サーバがダウンしていることを意味します。

正インスタンスが RUM レポートの送信を開始するトポロジ（CSLU を介して CSSM に接続：製品インスタンス開始型通信、CSSM から切断されている CSSM、CSLU への直接接続：製品スタンス開始型通信、および SSM オンプレミス展開：製品インスタンス開始型通信）では、この通信障害メッセージがスケジュールされたレポート（**license smart usage interval interval_in_days** グローバル コンフィギュレーション コマンド）と一致している場合は、製品インスタンスはスケジュールされた時間が経過した後、最大 4 時間にわたって RUM レポートを送信しようとします。（通信障害が続くために）それでもレポートを送信できない場合、シ

システムは間隔を 15 分にリセットします。通信障害が解消されると、レポート間隔は最後に設定された値に戻ります。

推奨するアクション：

CSSM に到達できない場合、CSLU に到達できない場合、および SSM オンプレミスに到達できない場合のトラブルシューティング手順を示します。

CSSM が到達不能で、設定されている転送タイプが **smart** の場合：

1. スマート URL が正しく設定されているかどうかを確認します。特権 EXEC モードで **show license status** コマンドを使用して、URL が次のようになっているかどうかを確認します。
<https://smartreceiver.cisco.com/licservice/license> そうでない場合は、グローバル コンフィギュレーション モードで **license smart url smart smar_URL** コマンドを再設定します。
2. DNS 解決を確認します。製品インスタンスが `smartreceiver.cisco.com` または `nslookup` で変換された IP に対して **ping** を実行できることを確認します。次の例は、変換された IP に対して **ping** を実行する方法を示しています。

```
Device# ping 171.70.168.183
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 171.70.168.183, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

CSSM が到達不能で、設定されている転送タイプが **callhome** の場合：

1. URL が正しく入力されているかどうかを確認します。特権 EXEC モードで **show license status** コマンドを使用して、URL が次のようになっているかどうかを確認します。
<https://tools.cisco.com/its/service/oddce/services/DDCEService>
2. Call Home プロファイル `CiscoTAC-1` がアクティブで、接続先 URL が正しいことを確認します。**show call-home profile all** コマンドは特権 EXEC モードで使用してください。

```
Current smart-licensing transport settings:
Smart-license messages: enabled
Profile: CiscoTAC-1 (status: ACTIVE)
Destination URL(s): https://tools.cisco.com/its/service/oddce/services/DDCEService
```

3. DNS 解決を確認します。製品インスタンスが `tools.cisco.com` または `nslookup` で変換された IP に対して **ping** を実行できることを確認します。

```
Device# ping tools.cisco.com
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 173.37.145.8, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 41/41/42 ms
```

上記の方法で解決しない場合は、製品インスタンスが設定されているかどうか、製品インスタンスの IP ネットワークが稼働しているかどうかを確認します。ネットワークが稼働していることを確認するには、インターフェイス コンフィギュレーション モードで **no shutdown** コマンドを設定します。

デバイスがサブネット IP でサブネットマスクされているかどうか、および DNS IP が設定されているかどうかを確認します。

4. HTTPS クライアントの送信元インターフェイスが正しいことを確認します。

現在の設定を表示するには、特権 EXEC モードで **show ip http client** コマンドを使用します。グローバル コンフィギュレーション モードで **ip http client source-interface** コマンドを使用して、再設定します。

上記の方法で解決しない場合は、ルーティングルール、およびファイアウォール設定を再確認します。

CSLU に到達できない場合：

1. CSLU 検出が機能するかどうかを確認します。

- `cslu-local` のゼロタッチ DNS 検出またはドメインの DNS 検出。

show license all コマンドの出力で、Last ACK received: フィールドを確認します。このフィールドに最新のタイムスタンプがある場合は、製品インスタンスが CSLU と接続されていることを意味します。ない場合は、次のチェックに進みます。

製品インスタンスが `cslu-local` に対して ping を実行できるかどうかを確認します。ping が成功すると、製品インスタンスが到達可能であることが確認されます。

上記の方法で解決しない場合は、ホスト名 `cslu-local` が CSLU の IP アドレス (CSLU をインストールした Windows ホスト) にマッピングされているエントリを使用してネームサーバを設定します。グローバル コンフィギュレーション モードで **ip domain name domain-name** コマンドと **ip name-server server-address** コマンドを設定します。この例では、CSLU IP は 192.168.0.1 で、name-server によってエントリ `cslu-local.example.com` が作成されます。

```
Device(config)# ip domain name example.com
Device(config)# ip name-server 192.168.0.1
```

- CSLU URL が設定されています。

show license all コマンド出力の Transport: ヘッダーで、次の点を確認します。Type: は `cslu` で、Cslu address: は CSLU をインストールした Windows ホストのホスト名または IP アドレスになっている必要があります。残りのアドレスが下記のように設定されているかどうかを確認するとともに、ポート番号が 8182 であるかどうかを確認します。

```
Transport:
  Type: cslu
  Cslu address: http://192.168.0.1:8182/cslu/v1/pi
```

そうでない場合は、グローバルコンフィギュレーションモードで **license smart transport cslu** および **license smart url cslu http://<cslu_ip_or_host>:8182/cslu/v1/pi** コマンドを設定します。

2. CSLU 開始型通信の場合、上記の CSLU 検出チェックに加えて、次の点を確認します。

HTTP 接続を確認します。特権 EXEC モードで **show ip http server session-module** コマンドを使用します。出力の HTTP server current connections: ヘッダーで、SL_HTTP がアクティブになっていることを確認します。[CSLU 開始型通信のネットワーク到達可能性の確認 \(181 ページ\)](#) で説明されているとおりに **ip http** が再設定されていない場合：

CSLU がインストールされているデバイスの Web ブラウザで、
<https://<product-instance-ip>/> を確認します。これにより、CSLU から製品インスタンス
 への REST API が期待どおりに動作することが保証されます。

SSM オンプレミスに到達できない場合：

1. 製品インスタンス開始型通信の場合は、SSM オンプレミスのトランスポートタイプと URL
 が正しく設定されているかどうかを確認します。

show license all コマンドの出力の `Transport:` ヘッダーの下で、`Type:` が `cslu` であり、`Cslu`
`address:` には、SSM オンプレミスにインストールしたサーバのホスト名または IP アドレ
 スと、デフォルトのローカルバーチャルアカウントの `<tenantID>` があることを確認しま
 す。次の例を参照してください。

```
Transport:
  Type: cslu
  Cslu address: https://192.168.0.1/cslu/v1/pi/on-prem-default
```

SSM オンプレミスの正しい URL があることを確認し（[トランスポート URL の取得（SSM
 オンプレミス UI）（198 ページ）](#) を参照）、次に、グローバル コンフィギュレーション
 モードで **license smart transport cslu** コマンドと **license smart url cslu**
[http://<ip>/cslu/v1/pi/<tenant ID>](#) コマンドを設定します。

[製品インスタンス開始型通信のネットワーク到達可能性の確認（196 ページ）](#) に記載され
 ているように、ネットワークに必要な他のコマンドが設定されていることを確認します。

2. SSM オンプレミス開始型通信の場合は、HTTPS 接続を確認します。

特権 EXEC モードで **show ip http server session-module** コマンドを使用します。出力の `HTTP`
`server current connections:` ヘッダーで、`SL_HTTP` がアクティブになっていることを確認
 します。[SSM オンプレミス開始型通信のネットワーク到達可能性の確保（201 ページ）](#) で
 説明されているとおりに **ip http** コマンドが再設定されていない場合は、次の手順を実行し
 ます。

3. トラストポイントと証明書が受け入れられることを確認します。

SSM オンプレミス展開の両方の通信形式で、正しいトラストポイントが使用され、必要な
 証明書が受け入れられることを確認します。

```
Device(config)# crypto pki trustpoint SLA-TrustPoint
Device(ca-trustpoint)#
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# end
Device# copy running-config startup-config
```

上記がうまくいかず、通信障害が続く場合は、シスコのテクニカルサポート担当者にお問い合わせ
 ください。

```
Error Message %SMART_LIC-3-COMM_RESTORED: Communications with the [chars] restored.
[chars] - depends on the transport type
```

```

- Cisco Smart Software Manager (CSSM)
- Cisco Smart License utility (CSLU)
Smart Agent communication with either the Cisco Smart Software Manager (CSSM) or the
Cisco Smart License
utility (CSLU) has been restored. No action required.

```

説明： CSSM、CSLU、または SSM オンプレミスのいずれかとの製品インスタンス通信が復元されます。

推奨するアクション： アクションは必要ありません。

```

Error Message %SMART_LIC-3-POLICY_REMOVED: The licensing policy has been removed.

```

説明： 以前にインストールしたカスタムライセンスポリシーが削除されました。Cisco default ポリシーが自動的に有効になります。これにより、スマートライセンシングの動作が変更される可能性があります。

失敗の理由として次が考えられます。

特権 EXEC モードで **license smart factory reset** コマンドを入力すると、ポリシーを含むすべてのライセンス情報が削除されます。

推奨するアクション：

ポリシーが意図的に削除された場合、それ以上のアクションは不要です。

ポリシーが誤って削除された場合は、ポリシーを再適用できます。実装したトポロジに応じて、該当するメソッドに従ってポリシーを取得します。

- CSSM に直接接続：

show license status を入力し、Trust Code Installed: フィールドを確認します。信頼が確立されると、CSSM は再度ポリシーを自動的に返します。ポリシーは、対応するバーチャルアカウントのすべての製品インスタンスに自動的に再インストールされます。

信頼が確立されていない場合は、次のタスクを実行します。[CSSMからの信頼コード用新規トークンの生成 \(210 ページ\)](#) および [信頼コードのインストール \(211 ページ\)](#) これらのタスクを完了すると、CSSM は再度ポリシーを自動的に返します。その後、バーチャルアカウントのすべての製品インスタンスにポリシーが自動的にインストールされます。

- CSLU を介して CSSM に接続：

- 製品インスタンス開始型通信の場合は、特権 EXEC モードで **license smart sync** コマンドを入力します。同期要求により、CSLU は欠落している情報 (ポリシーまたは承認コード) を製品インスタンスにプッシュします。

- CSLU 開始型通信の場合は、次のタスクを実行します。[使用状況レポートの収集：CSLU 開始 \(179 ページ\)](#) タスクを実行すると、CSLU は ACK 応答で欠落しているポリシーを検出して再提供します。

- CSLU は CSSM から切断：

- 製品インスタンス開始型通信の場合は、特権 EXEC モードで **license smart sync** コマンドを入力します。同期要求により、CSLU は欠落している情報（ポリシーまたは承認コード）を製品インスタンスにプッシュします。次に、次のタスクを指定された順序で実行します。[Download All For Cisco \(CSLU インターフェイス\) \(180 ページ\)](#) > [CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) > [Upload From Cisco \(CSLU インターフェイス\) \(181 ページ\)](#)
- CSLU 開始型通信の場合は、次のタスクを実行します。[使用状況レポートの収集：CSLU 開始 \(179 ページ\)](#) タスクを実行すると、CSLU は ACK 応答で欠落しているポリシーを検出して再提供します。次に、次のタスクを指定された順序で実行します。[Download All For Cisco \(CSLU インターフェイス\) \(180 ページ\)](#) > [CSSM への使用状況データのアップロードと ACK のダウンロード \(213 ページ\)](#) > [Upload From Cisco \(CSLU インターフェイス\) \(181 ページ\)](#)
- CSSM への接続なし、CSLU なし
完全に外部との接続性がないネットワークにいる場合は、インターネットと CSSM に接続できるワークステーションから次のタスク：[CSSM からのポリシーファイルのダウンロード \(212 ページ\)](#) および [製品インスタンスへのファイルのインストール \(214 ページ\)](#) を実行します。
- SSM オンプレミス展開
 - 製品インスタンス開始型通信の場合は、特権 EXEC モードで **license smart sync** コマンドを入力します。製品インスタンスを SSM オンプレミスと同期させ、必要な情報または欠落している情報を復元する原因です。必要に応じて、SSM オンプレミスと CSSM を同期します。
 - SSM オンプレミス開始型通信の場合：SSM オンプレミス UI で、[Reports] > [Synchronization pull schedule] > [Synchronize now with the device] に移動します。

SSM オンプレミス展開の両方の通信形式で、次のいずれかのオプションを使用して CSSM と同期します。

- SSM オンプレミスが CSSM に接続されている場合：SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports] > [Usage Schedules] > [Synchronize now with Cisco] に移動します。
- SSM オンプレミスが CSSM に接続されていません。[使用状況データのエクスポートとインポート \(SSM オンプレミス UI\) \(199 ページ\)](#) を参照してください。

```
Error Message %SMART_LIC-3-TRUST_CODE_INSTALL_FAILED: The install of a new licensing trust code has failed on [chars]: [chars].
```

説明：信頼コードのインストールに失敗しました。最初の [chars] は、信頼コードのインストールが試行された UDI です。2 番目の [chars] は、エラーの詳細を示すエラー文字列です。

失敗の理由として次が考えられます。

- 信頼コードがすでにインストールされています。信頼コードは製品インスタンスの UDI にノードロックされています。UDI がすでに登録されている場合に別の UDI をインストールしようとすると、インストールは失敗します。
- スマートアカウントとバーチャルアカウントの不一致：これは、（トークン ID が生成された）スマートアカウントまたはバーチャルアカウントに、信頼コードをインストールした製品インスタンスが含まれていないことを意味します。CSSM で生成されたトークンは、スマートアカウントまたはバーチャルアカウントレベルで適用され、そのアカウントのすべての製品インスタンスにのみ適用されます。
- 署名の不一致：これは、システムクロックが正確でないことを意味します。
- タイムスタンプの不一致：製品インスタンスの時刻が CSSM と同期していないため、インストールが失敗する可能性があります。

推奨するアクション：

- 信頼コードがすでにインストールされています。製品インスタンスに信頼コードがすでに存在する状況で信頼コードをインストールする場合は、特権 EXEC モードで **license smart trust idtoken id_token_value {local|all} [force]** コマンドを再設定します。再設定の際、**force** キーワードを必ず含めてください。**force** キーワードを入力すると、CSSM に送信されるメッセージに強制フラグが設定され、すでに存在する場合でも新しい信頼コードが作成されます。

- スマートアカウントとバーチャルアカウントの不一致：

<https://software.cisco.com> で CSSM Web UI にログインし、[Smart Software Licensing]> [Inventory]> [Product Instances] をクリックします。

トークンを生成する製品インスタンスが、選択したバーチャルアカウントにリストされているかどうかを確認します。リストされている場合は、次のステップに進みます。リストされていない場合は、正しいスマートアカウントとバーチャルアカウントを確認して選択します。その後、次のタスクを再度実行します。[CSSMからの信頼コード用新規トークンの生成 \(210 ページ\)](#) および [信頼コードのインストール \(211 ページ\)](#)

- タイムスタンプの不一致と署名の不一致：グローバル コンフィギュレーション モードで **ntp server** コマンドを設定します。次に例を示します。

```
Device(config)# ntp server 198.51.100.100 version 2 prefer
```

```
-----
Error Message      %SMART_LIC-4-REPORTING_NOT_SUPPORTED: The CSSM OnPrem that this
product instance is connected to is down rev and does not support the enhanced policy
and usage
reporting mode.
```

説明： Cisco Smart Software Manager オンプレミス（旧称 Cisco Smart Software Manager サテライト）は、Cisco IOS XE Amsterdam 17.3.3 以降でのみ Smart Licensing Using Policy 環境でサポートされています（[SSM オンプレミス（113 ページ）](#) を参照）。サポートされていないリリースでは、製品インスタンスは次のように動作します。

- 登録の更新と承認の更新の送信を停止します。
- 使用状況の記録を開始し、RUM レポートをローカルに保存します。

推奨するアクション：

次の選択肢があります。

- 代わりに、サポートされているトポロジを参照し、いずれかを実装します。[サポートされるトポロジ（118 ページ）](#) を参照してください。
- Smart Licensing Using Policy で SSM オンプレミスがサポートされているリリースにアップグレードします。[Smart Licensing Using Policy をサポートする SSM オンプレミスのバージョンへの移行（173 ページ）](#) を参照してください。

```
Error Message %SMART_LIC-6-POLICY_INSTALL_SUCCESS: A new licensing policy
was successfully installed.
```

説明： 次のいずれかの方法でポリシーがインストールされました。

- Cisco IOS コマンドの使用
- CSLU 開始型通信
- ACK 応答の一部として

推奨するアクション： アクションは必要ありません。適用されているポリシー（使用中のポリシー）とそのレポート要件を確認するには、特権 EXEC モードで **show license all** コマンドを入力します。

```
Error Message %SMART_LIC-6-AUTHORIZATION_INSTALL_SUCCESS: A new licensing
authorization code was successfully installed on: [chars].
```

このメッセージは、Cisco Catalyst アクセス、コア、およびアグリゲーションスイッチには該当しません。これらの製品インスタンスには輸出規制ライセンスや適用ライセンスがないためです。

```
Error Message %SMART_LIC-6-AUTHORIZATION_REMOVED: A licensing authorization code has
been removed from [chars]
```

説明： [chars] は、承認コードがインストールされた UDI です。承認コードが削除されました。これにより、製品インスタンスからライセンスが削除され、スマートライセンシングとライセンスを使用する機能の動作が変更される可能性があります。

推奨するアクション： アクションは必要ありません。ライセンスの現在の状態を確認するには、特権 EXEC モードで **show license all** コマンドを入力します。

```
Error Message %SMART_LIC-6-REPORTING_REQUIRED: A Usage report acknowledgement
will be required in [dec] days.
```

説明： これは、シスコへの RUM レポートが必要であることを意味するアラートです。 [dec] は、このレポート要件を満たすために残された時間（日数）です。

推奨するアクション： 要求された時間内に RUM レポートが送信されるようにします。実装したトポロジによって、レポート方式が決まります。

- CSLU を介して CSSM に接続
 - 製品インスタンス開始型通信の場合：特権 EXEC モードで **license smart sync** コマンドを入力します。CSLU が現在 CSSM にログインしている場合、CSSM 内の関連付けられているスマートアカウントとバーチャルアカウントに自動的に送信されます。
 - CSLU 開始型通信の場合は、次のタスクを実行します。 [使用状況レポートの収集：CSLU 開始（179 ページ）](#)
- CSSM への直接接続：特権 EXEC モードで **license smart sync** コマンドを入力します。
- コントローラを介して CSSM に接続：製品インスタンスがコントローラによって管理されている場合、コントローラはスケジュールされた時間に RUM レポートを送信します。
Cisco DNA Center をコントローラとして使用している場合は、アドホックレポートのオプションがあります。必要なリリース（リリース 2.2.2 以降）の『[Cisco DNA Center Administrator Guide](#)』 [英語] で「Manage Licenses」の「Upload Resource Utilization Details to CSSM」を参照してください。
- CSSM からの CSLU の切断：製品スタンスが CSLU に接続されている場合は、上記の「CSLU を介した CSSM への接続」に示したように製品インスタンスと同期してから、タスク [Download All For Cisco（CSLU インターフェイス）（180 ページ）](#)、[CSSM への使用状況データのアップロードと ACK のダウンロード（213 ページ）](#)、[Upload From Cisco（CSLU インターフェイス）（181 ページ）](#) を実行します。
- CSSM への接続なしで CSLU なし：特権 EXEC モードで **license smart save usage** コマンドを入力し、使用状況の必要な情報をファイルに保存します。次に、CSSM に接続しているワークステーションから、次のタスクを実行します。[CSSM への使用状況データのアップロードと ACK のダウンロード（213 ページ）](#) > [製品インスタンスへのファイルのインストール（214 ページ）](#)
- SSM オンプレミス展開：

製品インスタンスを SSM オンプレミスと同期します。

- 製品インスタンス開始型通信の場合：特権 EXEC モードで **license smart sync** コマンドを入力します。CSLU が現在 CSSM にログインしている場合、CSSM 内の関連付けられているスマートアカウントとバーチャルアカウントに自動的に送信されます。
- SSM オンプレミス開始型通信の場合は、次の手順を実行します。SSM オンプレミス UI で、[Reports]>[Synchronization pull schedule]>[Synchronize now with the device] に移動します。

使用状況情報を CSSM と同期します（いずれかを選択）。

- SSM オンプレミスが CSSM に接続されている場合：SSM オンプレミス UI の [Smart Licensing] ワークスペースで、[Reports]>[Usage Schedules]>[Synchronize now with Cisco] に移動します。
- SSM オンプレミスが CSSM に接続されていません。[使用状況データのエクスポートとインポート \(SSM オンプレミス UI\) \(199 ページ\)](#) を参照してください。

```
Error Message %SMART_LIC-6-TRUST_CODE_INSTALL_SUCCESS: A new licensing trust code
was successfully installed on [chars].
```

説明：[chars] は、信頼コードが正常にインストールされた UDI です。

推奨するアクション：アクションは必要ありません。信頼コードがインストールされていることを確認するには、特権 EXEC モードで **show license status** コマンドを入力します。出力のヘッダー Trust Code Installed: で更新されたタイムスタンプを探します。

ポリシーを使用したスマートライセンシングのその他の参考資料

トピック	マニュアルタイトル
この章で使用するコマンドのシンタックスおよび使用方法の詳細については、必要なリリースのコマンドリファレンスで [System Management]>[System Management Commands] を参照してください。	Command Reference (Catalyst 9300 シリーズ スイッチ)
Cisco Smart Software Manager のヘルプ	Smart Software Manager Help

トピック	マニュアルタイトル
Cisco Smart License Utility (CSLU) のインストールおよびユーザガイド	Cisco Smart License Utility Quick Start Setup Guide Cisco Smart License Utility User Guide

ポリシーを使用したスマートライセンシングの機能の履歴

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	<p>クラウドベースのソフトウェアライセンス管理ソリューションであり、ライセンス、ハードウェア、およびソフトウェアの使用状況の傾向を管理および追跡できます。</p> <p>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。</p> <p>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。</p>

リリース	機能	機能情報
Cisco IOS XE Amsterdam 17.3.2a	ポリシーを使用したスマートライセンス	<p>スマートライセンシングの拡張バージョンには、ネットワークの運用を中断させないライセンスソリューションを提供するという主目的がありますが、むしろ、購入および使用しているハードウェアおよびソフトウェアライセンスを考慮して、コンプライアンス関係を実現するライセンスソリューションを提供するという目的もあります。</p> <p>このリリース以降、ポリシーを使用したスマートライセンスがデバイスで自動的に有効になります。これは、このリリースにアップグレードする場合にも当てはまります。</p> <p>デフォルトでは、CSSM のスマートアカウントとバーチャルアカウントは、ポリシーを使用したスマートライセンスで有効になっています。</p>
	Cisco DNA Center での Smart Licensing Using Policy のサポート	<p>Cisco DNA Center は、Cisco DNA Center リリース 2.2.2 以降、Smart Licensing Using Policy 機能をサポートしています。</p> <p>Cisco DNA Center を使用して製品インスタンスを管理する場合、Cisco DNA Center は CSSM に接続し、CSSM とのすべての通信のインターフェイスとなります。</p> <p>互換性のあるコントローラと製品インスタンスバージョンについては、コントローラ (112 ページ) を参照してください。</p> <p>このトポロジについては、コントローラを介して CSSM に接続 (122 ページ) と トポロジのワークフロー: コントローラを介して CSSM に接続 (141 ページ) を参照してください。</p>

リリース	機能	機能情報
Cisco IOS XE Amsterdam 17.3.3	Smart Licensing Using Policy 用の Smart Software Manager オンプレミス (SSM オンプレミス) サポート	<p>SSM オンプレミスは、CSSM と連動するアセットマネージャです。これにより、CSSM に直接接続する代わりに、オンプレミスで製品とライセンスを管理できます。</p> <p>互換性のある SSM オンプレミスと製品インスタンスバージョンについては、SSM オンプレミス (113 ページ) を参照してください。</p> <p>このトポロジの概要についてと実装方法については、SSM オンプレミス展開 (125 ページ) とトポロジのワークフロー: SSM オンプレミス展開 (146 ページ) を参照してください。</p> <p>既存のバージョンの SSM オンプレミスから、Smart Licensing Using Policy への移行をサポートするバージョンへの移行については、Smart Licensing Using Policy をサポートする SSM オンプレミスのバージョンへの移行 (173 ページ) を参照してください。</p>

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



第 5 章

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

- [有線ネットワークでの Application Visibility and Control について \(239 ページ\)](#)
- [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 \(240 ページ\)](#)
- [有線 Application Visibility and Control の制限 \(241 ページ\)](#)
- [Application Visibility and Control の設定方法 \(243 ページ\)](#)
- [Application Visibility and Control のモニターリング \(273 ページ\)](#)
- [例：Application Visibility and Control の設定 \(273 ページ\)](#)
- [基本的なトラブルシューティング：質問と回答 \(285 ページ\)](#)
- [Application Visibility and Control に関する追加情報 \(286 ページ\)](#)
- [有線ネットワークでの Application Visibility and Control の機能履歴 \(287 ページ\)](#)

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

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

サポートされる AVC クラス マップおよびポリシー マップのフォーマット

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

サポートされる AVC クラス マップのフォーマット

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

サポートされる AVC ポリシーのフォーマット

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

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

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

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

有線 Application Visibility and Control の制限

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

- プロトコルベースまたは属性ベースのいずれかのポートに同時に接続できるのは、NBAR ベースの QoS メカニズムの 1 つだけです。次の 2 つの属性のみがサポートされます。
 - traffic-class
 - business-relevance
- 従来の WDAVC QoS の制限事項は引き続き適用されます。
 - マーキングとポリシングのみがサポートされます。
 - 物理インターフェイスだけがサポートされます。
 - アプリケーション分類がオフラインで行われるため、QoS 分類には遅延があります (ただし、フローの最初のパケットは、正確な QoS 分類の前に転送されます)。
- NBAR2 ベースの一致基準 **match protocol** は、マーキングアクションおよびポリシングアクションでのみ許可されます。NBAR2 一致基準は、キューイング機能が設定されているポリシーでは許可されません。
- 「一致プロトコル」：すべてのポリシーで最大 255 の同時に異なるプロトコル (8 ビットの HW 制限)。
- AVC は管理ポート (Gig 0/0) ではサポートされていません。
- IPv6 パケットの分類はサポートされていません。
- IPv4 ユニキャスト (TCP/UDP) のみがサポートされます。
- Netflow レコードで「アプリケーション名一致」が使用されている場合、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。
- 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。これらのポートはポートチャネルの一部とすることはできません。トランクポートへの接続はサポートされません。

- 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）。
-
- 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 エンジンを実稼働させます。「インターフェイスでのアプリケーション認識の有効化」のセクション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」のセクション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」のセクション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 エクスポートを作成してフロー レコードをエクスポートします。
- フロー レコードおよびフロー エクスポートに基づいて、フロー モニターを作成します。
- インターフェイスにフロー モニターを接続します。

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

インターフェイスでのアプリケーション認識の有効化

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

手順

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

AVC QoS ポリシーの作成

AVC QoS ポリシーを作成するには、次の一般的な手順を実行します。

1. match protocol フィルタでクラス マップを作成します。
2. ポリシー マップを作成します。
3. インターフェイスにポリシー マップを適用します。

クラス マップの作成

match protocol フィルタを設定する前に、クラス マップを作成する必要があります。マーキングやポリシングなどの QoS アクションをトラフィックに適用できます。AVC の match protocol フィルタは、有線アクセスポートに適用されます。サポートされているプロトコルの詳細につ

いては、http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/prot_lib/config_library/nbar-prot-pack-library.html を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	class-map class-map-name 例： Device (config)# class-map webex-class	クラス マップを作成します。
ステップ 3	match protocol application-name 例： Device (config)# class-map webex-class Device (config-cmap)# match protocol webex-media	アプリケーション名との一致を指定します。
ステップ 4	end 例： Device (config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

ポリシー マップの作成

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	policy-map policy-map-name 例： 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。 デフォルトでは、ポリシー マップは定義されていません。 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

	コマンドまたはアクション	目的
		<p>合は CoS が 0 に設定されます。ポリシーは実行されません。</p> <p>(注) 既存のポリシーマップを削除するには、no policy-map <i>policy-map-name</i> グローバル コンフィギュレーション コマンドを使用します。</p>
ステップ 3	<p>class [<i>class-map-name</i> class-default]</p> <p>例 :</p> <pre>Device(config-pmap) # class webex-class</pre>	<p>トラフィックの分類を定義し、ポリシーマップ クラス コンフィギュレーション モードを開始します。</p> <p>デフォルトでは、ポリシー マップおよびクラスマップは定義されていません。</p> <p>すでに class-map グローバル コンフィギュレーション コマンドを使用してトラフィッククラスが定義されている場合は、このコマンドで <i>class-map-name</i> にその名前を指定します。</p> <p>class-default トラフィッククラスは定義済みで、どのポリシーにも追加できます。このトラフィック クラスは、常にポリシーマップの最後に配置されます。暗黙の match any が class-default クラスに含まれている場合、他のトラフィッククラスと一致しないパケットはすべて class-default と一致します。</p> <p>(注) 既存のクラスマップを削除するには、no class <i>class-map-name</i> ポリシーマップ コンフィギュレーション コマンドを使用します。</p>
ステップ 4	<p>police <i>rate-bps burst-byte</i></p> <p>例 :</p> <pre>Device(config-pmap-c) # police 100000 80000</pre>	<p>分類したトラフィックにポリサーを定義します。</p> <p>デフォルトでは、ポリサーは定義されていません。</p> <ul style="list-style-type: none"> • <i>rate-bps</i> には、平均トラフィック レートをビット/秒 (bps) で指定します。指定できる範囲は 8000 ~ 10000000000 です

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>burst-byte</i> には、標準バースト サイズをバイト数で指定します。有効範囲は、1000 ~ 512000000 です。
ステップ 5	set {dscp new-dscp cos cos-value} 例： 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。 <ul style="list-style-type: none"> • <i>dscp new-dscp</i> には、分類されたトラフィックに割り当てる新しい DSCP 値を入力します。指定できる範囲は 0 ~ 63 です。
ステップ 6	end 例： 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 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface interface-id 例： Device(config)# interface GigabitEthernet 1/0/1	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 例： Device(config-if)# service-policy input MARKING_IN	インターフェイスにローカル ポリシーを適用します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

有線 AVC モニターの一致および収集フィールドのサポート強化

Cisco IOS XE Amsterdam 17.3.1 リリース以降、有線 AVC モニターで使用できる一致または収集フィールドに関する制限事項は削除されました。有線 AVC モニターでは、次に示す一致フィールドまたは収集フィールドの任意の組み合わせを使用できます。

サポートされる一致または収集フィールド

- application name
- ipv4/6 version
- ipv4/6 protocol
- transport tcp flags
- flow direction
- connection initiator
- connection client ipv4 address
- connection server ipv4 address
- ipv4 source address
- ipv4 destination address
- connection client ipv6 address
- connection server ipv6 address
- ipv6 source address
- ipv6 destination address
- connection client transport port
- connection server transport port
- transport source-port
- transport destination-port
- flow observation point
- interface input
- interface output
- datalink mac source address input
- datalink mac destination address input
- datalink mac source address output
- datalink mac destination address output
- datalink dot1q vlan input
- datalink dot1q vlan output

サポートされる収集専用フィールド

- connection client counter packets long
- connection client counter bytes network long
- connection server counter packets long
- connection server counter bytes network long
- counter bytes long
- counter packets long
- timestamp absolute first
- timestamp absolute last
- connection new-connections

抽出済みフィールド

- application dns domain-name
- application http host
- application ssl common-name

有線 AVC モニターでは、上記のフィールドを任意に組み合わせて設定できます。

フィールド「interface input」および「interface output」は、それぞれレコード内の一致フィールドとして使用できます。ただし、両方のフィールドを同じレコードの一致フィールドとして使用することはできません。両方のフィールドを収集フィールドとして使用することも、1つを収集フィールドとして、1つを同じレコード内の一致フィールドとして使用することもできます。

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、フローモニター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/サーバーアプリケーション統計情報レコードであり、新しい方向性レコードは入出力のアプリケーション統計情報です。

双方向フローレコード

フローレコード 1 : 双方向フローレコード

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow record flow_record_name 例 : Device(config)# flow record fr-wdavic-1	フローレコードコンフィギュレーション モードを開始します。
ステップ 3	description description 例 : Device(config-flow-record)# description fr-wdavic-1	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例 : Device(config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device(config-flow-record)# match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match application name 例 : Device(config-flow-record)# match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 7	match connection client ipv4 address 例 : Device(config-flow-record)# match connection client ipv4 address	クライアント (フローイニシエータ) の IPv4 アドレスとの一致を指定します。
ステップ 8	match connection server ipv4 address 例 : Device(config-flow-record)# match connection server ipv4 address	サーバー (フローレスポンド) の IPv4 アドレスとの一致を指定します。
ステップ 9	match connection server transport port 例 :	サーバーのポート番号との一致を指定します。

	コマンドまたはアクション	目的
	Device (config-flow-record) # match connection server transport port	
ステップ 10	match flow observation point 例 : Device (config-flow-record) # match flow observation point	フロー観測メトリックの観測ポイント ID との一致を指定します。
ステップ 11	collect flow direction 例 : Device (config-flow-record) # collect flow direction	<p>次の手順で collect connection initiator コマンドの initiator キーワードで指定される双方向フローの関連する側（イニシエータまたはレスポнда）の方向（入力または出力）を収集するように指定します。 initiator キーワードで指定される値に応じて、 flow direction キーワードは次の値をとります。</p> <ul style="list-style-type: none"> • 0x01 = 入力フロー • 0x02 = 出力フロー <p>initiator キーワードがイニシエータに設定されている場合、フローの方向はフローのイニシエータ側から指定されます。 initiator キーワードがレスポндаに設定されている場合、フローの方向はフローのレスポнда側から指定されます。有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。</p>
ステップ 12	collect connection initiator 例 : Device (config-flow-record) # collect connection initiator	<p>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нда）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。</p> <ul style="list-style-type: none"> • 0x01 = イニシエータ : フローの送信元は接続のイニシエータです <p>有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。</p>
ステップ 13	collect connection new-connections 例 :	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
	Device(config-flow-record)# collect connection new-connections	
ステップ 14	collect connection client counter packets long 例 : Device(config-flow-record)# collect connection client counter packets long	クライアントが送信したパケット数を収集するように指定します。
ステップ 15	collect connection client counter bytes network long 例 : Device(config-flow-record)# collect connection client counter bytes network long	クライアントが送信したバイト数の合計を収集するように指定します。
ステップ 16	collect connection server counter packets long 例 : Device(config-flow-record)# collect connection server counter packets long	サーバー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 例 : Device(config-flow-record)# collect connection server counter bytes network long	サーバーが送信したバイト数の合計を収集するように指定します。
ステップ 18	collect timestamp absolute first 例 : 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 例 : Device(config-flow-record)# collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 20	end 例 : Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーションモードを終了できます。
ステップ 21	show flow record 例 :	すべてのフローレコードに関する情報を表示します。

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

フローレコード 2: 双方向フローレコード

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow record <i>flow_record_name</i> 例 : Device (config)# flow record fr-wdavic-1	フローレコードコンフィギュレーション モードを開始します。
ステップ 3	description <i>description</i> 例 : Device (config-flow-record)# description fr-wdavic-1	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例 : Device (config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device (config-flow-record)# match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match application name 例 : Device (config-flow-record)# match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 7	match connection client ipv4 address 例 : Device (config-flow-record)# match connection client ipv4 address	クライアント (フローイニシエータ) の IPv4 アドレスとの一致を指定します。

	コマンドまたはアクション	目的
ステップ 8	match connection client transport port 例 : Device(config-flow-record) # match connection client transport port	(任意) フローレコードのキーフィールドとして、クライアントの接続ポートとの一致を指定します。
ステップ 9	match connection server ipv4 address 例 : Device(config-flow-record) # match connection server ipv4 address	サーバー (フローレスポнда) の IPv4 アドレスとの一致を指定します。
ステップ 10	match connection server transport port 例 : Device(config-flow-record) # match connection server transport port	サーバーのトランスポートポートとの一致を指定します。
ステップ 11	match flow observation point 例 : Device(config-flow-record) # match flow observation point	フロー観測メトリックの観測ポイント ID との一致を指定します。
ステップ 12	collect flow direction 例 : Device(config-flow-record) # collect flow direction	<p>次の手順で collect connection initiator コマンドの initiator キーワードで指定される双方向フローの関連する側 (イニシエータまたはレスポнда) の方向 (入力または出力) を収集するように指定します。 initiator キーワードで指定される値に応じて、flow direction キーワードは次の値をとります。</p> <ul style="list-style-type: none"> • 0x01 = 入力フロー • 0x02 = 出力フロー <p>initiator キーワードがイニシエータに設定されている場合、フローの方向はフローのイニシエータ側から指定されます。 initiator キーワードがレスポндаに設定されている場合、フローの方向はフローのレスポнда側から指定されます。有線 AVC では、initiator キーワードは常にイニシエータに設定されています。</p>
ステップ 13	collect connection initiator 例 :	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нда)

	コマンドまたはアクション	目的
	<code>Device(config-flow-record)# collect connection initiator</code>	<p>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。</p> <ul style="list-style-type: none"> • 0x01 = イニシエータ: フローの送信元は接続のイニシエータです <p>有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。</p>
ステップ 14	<p>collect connection new-connections</p> <p>例:</p> <pre>Device(config-flow-record)# collect connection new-connections</pre>	観測された接続開始の数を収集するように指定します。
ステップ 15	<p>collect connection client counter packets long</p> <p>例:</p> <pre>Device(config-flow-record)# collect connection client counter packets long</pre>	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	<p>collect connection client counter bytes network long</p> <p>例:</p> <pre>Device(config-flow-record)# collect connection client counter bytes network long</pre>	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	<p>collect connection server counter packets long</p> <p>例:</p> <pre>Device(config-flow-record)# collect connection server counter packets long</pre>	サーバー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	<p>collect connection server counter bytes network long</p> <p>例:</p> <pre>Device(config-flow-record)# collect connection server counter bytes network long</pre>	サーバー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
ステップ 19	collect timestamp absolute first 例： Device(config-flow-record)# collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 20	collect timestamp absolute last 例： Device(config-flow-record)# collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 21	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 22	show flow record 例： Device# show flow record	すべてのフローレコードに関する情報を表示します。

方向性フローレコード

フローレコード 3 : 方向性フローレコード : 入力

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 2	flow record flow_record_name 例： Device(config)# flow record fr-wdavic-3	フローレコードコンフィギュレーションモードを開始します。
ステップ 3	description description 例： Device(config-flow-record)# description flow-record-1	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例： Device(config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。

	コマンドまたはアクション	目的
ステップ 5	match ipv4 protocol 例： Device (config-flow-record) # match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match ipv4 source address 例： Device (config-flow-record) # match ipv4 source address	IPv4 送信元アドレスとの一致をキーフィールドとして指定します。
ステップ 7	match ipv4 destination address 例： Device (config-flow-record) # match ipv4 destination address	IPv4 宛先アドレスとの一致をキーフィールドとして指定します。
ステップ 8	match transport source-port 例： Device (config-flow-record) # match transport source-port	トランスポート発信元ポートとの一致をキーフィールドとして指定します。
ステップ 9	match transport destination-port 例： Device (config-flow-record) # match transport destination-port	トランスポート宛先ポートとの一致をキーフィールドとして指定します。
ステップ 10	match interface input 例： Device (config-flow-record) # match interface input	入力インターフェイスとの一致をキーフィールドとして指定します。
ステップ 11	match application name 例： Device (config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 12	collect interface output 例： Device (config-flow-record) # collect interface output	フローから出力インターフェイスを収集するように指定します。

フローレコード 4 : 方向性フローレコード : 出力

	コマンドまたはアクション	目的
ステップ 13	collect counter bytes long 例 : Device(config-flow-record)# collect counter bytes long	フローのバイト数を収集するように指定します。
ステップ 14	collect counter packets long 例 : Device(config-flow-record)# collect counter packets long	フローのパケット数を収集するように指定します。
ステップ 15	collect timestamp absolute first 例 : Device(config-flow-record)# collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 16	collect timestamp absolute last 例 : Device(config-flow-record)# collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 17	end 例 : Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 18	show flow record 例 : 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 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 2	flow record flow_record_name 例 : Device(config)# flow record fr-wdavic-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 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 flow-record-1	
ステップ 4	match ipv4 version 例 : Device (config-flow-record) # match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device (config-flow-record) # match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match ipv4 source address 例 : Device (config-flow-record) # match ipv4 source address	IPv4 送信元アドレスとの一致をキーフィールドとして指定します。
ステップ 7	match ipv4 destination address 例 : Device (config-flow-record) # match ipv4 destination address	IPv4 宛先アドレスとの一致をキーフィールドとして指定します。
ステップ 8	match transport source-port 例 : Device (config-flow-record) # match transport source-port	トランスポート発信元ポートとの一致をキーフィールドとして指定します。
ステップ 9	match transport destination-port 例 : Device (config-flow-record) # match transport destination-port	トランスポート宛先ポートとの一致をキーフィールドとして指定します。
ステップ 10	match interface output 例 : Device (config-flow-record) # match interface output	出力インターフェイスとの一致をキーフィールドとして指定します。
ステップ 11	match application name 例 : Device (config-flow-record) # match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。

	コマンドまたはアクション	目的
ステップ 12	collect interface input 例 : Device(config-flow-record)# collect interface input	フローから入力インターフェイスを収集するように指定します。
ステップ 13	collect counter bytes long 例 : Device(config-flow-record)# collect counter bytes long	フローのバイト数を収集するように指定します。
ステップ 14	collect counter packets long 例 : Device(config-flow-record)# collect counter packets long	フローのパケット数を収集するように指定します。
ステップ 15	collect timestamp absolute first 例 : Device(config-flow-record)# collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 16	collect timestamp absolute last 例 : Device(config-flow-record)# collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 17	end 例 : Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 18	show flow record 例 : Device# show flow record	すべてのフローレコードに関する情報を表示します。

DNS フローレコード

フローレコード 5 : DNS フローレコード

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 :	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	flow record <i>flow_record_name</i> 例 : Device (config)# flow record fr-wdavic-5	フローレコードコンフィギュレーションモードを開始します。
ステップ 3	description <i>description</i> 例 : Device (config-flow-record)# description flow-record-5	(任意) フローレコードの説明を作成します。
ステップ 4	match ipv4 version 例 : Device (config-flow-record)# match ipv4 version	IPv4 ヘッダーからの IP バージョンとの一致を指定します。
ステップ 5	match ipv4 protocol 例 : Device (config-flow-record)# match ipv4 protocol	IPv4 プロトコルとの一致を指定します。
ステップ 6	match application name 例 : Device (config-flow-record)# match application name	アプリケーション名との一致を指定します。 (注) この操作は、AVC サポートでは必須です。フローがアプリケーションと一致することが可能になるためです。
ステップ 7	match connection client ipv4 address 例 : Device (config-flow-record)# match connection client ipv4 address	クライアント (フローイニシエータ) の IPv4 アドレスとの一致を指定します。
ステップ 8	match connection client transport port 例 : Device (config-flow-record)# match connection client transport port	フローレコードのキーフィールドとして、クライアントの接続ポートとの一致を指定します。
ステップ 9	match connection server ipv4 address 例 : Device (config-flow-record)# match connection server ipv4 address	サーバー (フローレスポンド) の IPv4 アドレス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 10	match connection server transport port 例： Device(config-flow-record) # match connection server transport port	サーバーのポートポートとの一致を指定します。
ステップ 11	collect flow direction 例： Device(config-flow-record) # collect flow direction	<p>次の手順で collect connection initiator コマンドの initiator キーワードで指定される双方向フローの関連する側（イニシエータまたはレスポンド）の方向（入力または出力）を収集するように指定します。 initiator キーワードで指定される値に応じて、 flow direction キーワードは次の値をとります。</p> <ul style="list-style-type: none"> • 0x01 = 入力フロー • 0x02 = 出力フロー <p>initiator キーワードがイニシエータに設定されている場合、フローの方向はフローのイニシエータ側から指定されます。 initiator キーワードがレスポンドに設定されている場合、フローの方向はフローのレスポンド側から指定されます。有線 AVC では、 initiator キーワードは常にイニシエータに設定されています。</p>
ステップ 12	collect timestamp absolute first 例： Device(config-flow-record) # collect timestamp absolute first	最初のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 13	collect timestamp absolute last 例： Device(config-flow-record) # collect timestamp absolute last	最新のパケットがフローで確認されたときの時間をミリ秒単位で収集するように指定します。
ステップ 14	collect connection initiator 例： Device(config-flow-record) # collect connection initiator	collect flow direction コマンドで指定されたフローの方向に関連するフローの側（イニシエータまたはレスポンド）を収集するように指定します。 initiator キーワードは、フローの方向に関する次の情報を提供します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • 0x01 = イニシエータ : フローの送信元は接続のイニシエータです <p>有線 AVC では、initiator キーワードは常にイニシエータに設定されています。</p>
ステップ 15	collect connection new-connections 例 : Device(config-flow-record)# collect connection new-connections	観測された接続開始の数を収集するように指定します。
ステップ 16	collect connection server counter packets long 例 : Device(config-flow-record)# collect connection server counter packets long	サーバーが送信したパケット数を収集するように指定します。
ステップ 17	collect connection client counter packets long 例 : Device(config-flow-record)# collect connection client counter packets long	クライアントが送信したパケット数を収集するように指定します。
ステップ 18	collect connection server counter bytes network long 例 : Device(config-flow-record)# collect connection server counter bytes network long	サーバー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 例 : Device(config-flow-record)# collect connection client counter bytes network long	クライアントが送信したバイト数の合計を収集するように指定します。
ステップ 20	collect application dns domain-name 例 : Device(config-flow-record)# collect application dns domain-name	DNS ドメイン名を DNS フローレコードの収集フィールドとして使用するよう設定します。
ステップ 21	end 例 :	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。

フロー エクスポートの作成

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

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow exporter flow_exporter_name 例： Device(config)# flow exporter flow-exporter-1	フロー エクスポート コンフィギュレーション モードを開始します。
ステップ 3	description description 例： Device(config-flow-exporter)# description flow-exporter-1	(任意) フロー エクスポートの説明を作成します。
ステップ 4	destination { hostname ipv4-address ipv6-address } 例： Device(config-flow-exporter)# destination 10.10.1.1	エクスポートでデータを送信する宛先システムのホスト名、IPv4 または IPv6 アドレスを指定します。
ステップ 5	option application-table [timeout seconds] 例： Device(config-flow-exporter)# option application-table timeout 500	(任意) フロー エクスポートのアプリケーション テーブルのオプションを設定します。 timeout オプションを使用すると、フローエクスポートの再送信時間を秒単位で設定できます。有効な範囲は 1 ~ 86400 秒です。
ステップ 6	end 例： Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。
ステップ 7	show flow exporter 例： Device# show flow exporter	すべてのフロー エクスポートに関する情報を表示します。

	コマンドまたはアクション	目的
ステップ 8	show flow exporter statistics 例 : Device# show flow exporter statistics	フロー エクスポートの統計情報を表示します。

フロー モニターの作成

フロー モニターを作成して、フロー レコードに関連付けることができます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow monitor monitor-name 例 : Device (config)# flow monitor flow-monitor-1	フロー モニターを作成し、フロー モニター コンフィギュレーション モードを開始します。
ステップ 3	description description 例 : Device (config-flow-monitor)# description flow-monitor-1	(任意) フロー モニターの説明を作成します。
ステップ 4	record record-name 例 : Device (config-flow-monitor)# record flow-record-1	事前に作成されたレコードの名前を指定します。
ステップ 5	exporter exporter-name 例 : Device (config-flow-monitor)# exporter flow-exporter-1	事前に作成されたエクスポートの名前を指定します。
ステップ 6	cache { entries number-of-entries timeout { active inactive } type normal } 例 : Device (config-flow-monitor)# cache timeout active 1800 例 : Device (config-flow-monitor)# cache timeout inactive 200	(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。 <ul style="list-style-type: none"> • 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。 (注) 標準のキャッシュ タイプのみがサポートされます。

	コマンドまたはアクション	目的
	例 : Device(config-flow-monitor)# cache type normal	
ステップ 7	end 例 : Device(config)# end	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバルコンフィギュレーションモードを終了できます。
ステップ 8	show flow monitor 例 : Device# show flow monitor	すべてのフローモニターに関する情報を表示します。
ステップ 9	show flow monitor flow-monitor-name 例 : Device# show flow monitor flow-monitor-1	指定した有線 AVC フロー モニターに関する情報を表示します。
ステップ 10	show flow monitor flow-monitor-name statistics 例 : Device# show flow monitor flow-monitor-1 statistics	有線 AVC フロー モニターの統計情報を表示します。
ステップ 11	clear flow monitor flow-monitor-name statistics 例 : Device# clear flow monitor flow-monitor-1 statistics	指定したフローモニター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 例 : 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 例 : 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。

	コマンドまたはアクション	目的
ステップ 14	show flow monitor <i>flow-monitor-name</i> cache format csv 例 : <pre>Device# show flow monitor flow-monitor-1 cache format csv</pre>	CSV 形式でフロー キャッシュの内容を表示します。

インターフェイスへのフロー モニターの関連付け

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 モニターをインターフェイスに同時に接続できます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : <pre>Device# configure terminal</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface <i>interface-id</i> 例 : <pre>Device(config)# interface GigabitEthernet 1/0/1</pre>	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ip flow monitor <i>monitor-name</i> { input output } 例 : <pre>Device(config-if) # ip flow monitor flow-monitor-1 input</pre>	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 モニターを関連付けます。
ステップ 4	end 例 : <pre>Device(config)# end</pre>	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

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、ユーザーは、レポート目的に使用できるセレクト ID を定義できます。

さまざまなタイプのアプリケーション カスタマイズがあります。

一般的なプロトコルのカスタマイズ

- HTTP
- SSL
- DNS

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

レイヤ 3/レイヤ 4 のカスタマイズ

- IPv4 アドレス
- DSCP 値
- TCP/UDP ポート
- フロー送信元または宛先の方向

バイト オフセット：ペイロードの特定のバイト値に基づくカスタマイズ

HTTP のカスタマイズ

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

- **cookie** : HTTP クッキー
- **host** : リソースを含む元のサーバーのホスト名
- **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 サーバー名指定 (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/xs-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ーリング

カスタム アプリケーションのモニターリングのための 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。



警告 スイッチスタック構成を使用する場合は、各スイッチに同じプロトコルパックファイルがロードされていることを確認します。スタック内のプライマリスイッチで **ip nbar protocol-pack flash protocol-pack-file** コマンドを実行すると、ファイルがロードされていないスタック内のスイッチは、設定の不一致が原因でリロードされます。

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 \(271 ページ\)](#) を参照してください。

NBAR2 プロトコルパックのロード

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	ip nbar protocol-pack protocol-pack [force] 例： Device(config)# ip nbar protocol-pack flash:defProtoPack 例：	プロトコルパックをロードします。 • 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

例：NBAR2 プロトコルパックのロード

	コマンドまたはアクション	目的
	Device(config)# default ip nbar protocol-pack	在のプロトコルパックでサポートされていない設定も削除されます。 組み込みのプロトコルパックに戻るには、次のコマンドを使用します。
ステップ 4	exit 例： Device(config)# exit	特権 EXEC モードに戻ります。
ステップ 5	show ip nbar protocol-pack {protocol-pack active } [detail] 例： Device# show ip nbar protocol-pack active	プロトコルパック情報を表示します。 <ul style="list-style-type: none"> このコマンドを使用して、ロードされたプロトコルパックのバージョン、パブリッシャ、その他の詳細を確認します。 指定されたプロトコルパックの情報を表示するには、<i>protocol-pack</i> 引数を使用します。 アクティブなプロトコルパックの情報を表示するには、active キーワードを使用します。 詳細なプロトコルパックの情報を表示するには、detail キーワードを使用します。

例：NBAR2 プロトコルパックのロード

次の例に、新しいプロトコルパックをロードする方法を示します。

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

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

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

次の例に、組み込みのプロトコルパックに戻す方法を示します。

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

```
Device(config)# default ip nbar protocol-pack
Device(config)# exit
```

Application Visibility and Control のモニターリング

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ーするために使用できます。

表 18: スwitchのアプリケーションの可視性モニターリングコマンド

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

例 : Application Visibility and Control の設定

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

```
Device# configure terminal
Device(config)# class-map match-any NBAR-VOICE
Device(config-cmap)# match protocol ms-lync-audio
Device(config-cmap)#end
```

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

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

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

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

次に、ポリシーマップをスイッチポートに適用する例を示します。

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

次に、NBAR 属性に基づいてクラスマップを作成する例を示します。

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

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

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

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

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

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

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

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

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

show コマンドによる設定の表示

show ip nbar protocol-discovery

インターフェイスごとのプロトコル検出統計情報のレポートを表示します。

次に、インターフェイスごとの統計情報の出力例を示します。

```

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

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

Output
-----
Protocol          Packet Count
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           60580
Total             31174777
55911             3613000
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 モニター設定の表示

show flow monitor wdavc

指定した有線 AVC フロー モニター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 モニター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 モニター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、フローモニター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 wdvac cache format table
Cache type: Normal (Platform cache)
Cache size: 12000
Current entries: 13

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

CONN IPV4 INITIATOR ADDR CONN IPV4 RESPONDER ADDR CONN RESPONDER PORT
FLOW OBSPOINT ID IP VERSION IP PROT APP NAME
flow dirn .....
-----
-----
-----
64.103.125.147 144.254.71.184
53 4294967305 4 17 port dns
Input .....
64.103.121.103 10.1.1.2
67 4294967305 4 17 layer7 dhcp
Input ....contd.....
64.103.125.3 64.103.125.97
68 4294967305 4 17 layer7 dhcp
Input .....
10.0.2.6 157.55.40.149 443
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 wdacv cache format record

フローレコードと同様の形式でフローキャッシュの内容を表示します。

```

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

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

CONNECTION IPV4 INITIATOR ADDRESS:        64.103.125.147
CONNECTION IPV4 RESPONDER ADDRESS:        144.254.71.184
CONNECTION RESPONDER PORT:                53
FLOW OBSPOINT ID:                        4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         port dns
flow direction:                           Input
timestamp abs first:                      08:55:46.917
timestamp abs last:                       08:55:46.917
connection initiator:                     Initiator
connection count new:                     2
connection 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 wдавc 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 wдавc 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：** ほとんどのトラフィックが CAPWAP トラフィックになっているのですが、なぜですか。

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

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

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

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

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

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

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

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

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

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

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

Application Visibility and Control に関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

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

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	有線ネットワークでの Application Visibility and Control	AVC は、アプリケーションへの適応力やアプリケーションへのインテリジェンス性に基づいて、厳密なパケットおよび接続からブランチおよびキャンパスソリューションを発展させるためのシスコの取り組みの重要な部分です。
Cisco IOS XE Fuji 16.8.1a	有線アプリケーションの表示およびコントロール (有線 AVC) 属性ベース QoS (EasyQoS)	特定のプロトコルではなく、Network-Based Application Recognition (NBAR) 属性に基づいて QoS クラスとポリシーを定義できるようになりましたが、いくつかの制限があります。サポートされる NBAR 属性は、business-relevance および traffic-class のみです。
Cisco IOS XE Gibraltar 16.12.1	DNS フローレコード	DNS フローレコードのサポートが導入されました。DNS フローレコードは、フローレコードを定義するための collect フィールドとして DNS ドメイン名を使用します。
Cisco IOS XE Amsterdam 17.3.1	アプリケーションの可視性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	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。

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 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 \(289 ページ\)](#)
- [SDM テンプレートの設定方法 \(289 ページ\)](#)
- [SDM テンプレートのモニタリングおよびメンテナンス \(290 ページ\)](#)
- [SDM テンプレートの設定例 \(291 ページ\)](#)
- [SDM テンプレートに関する追加情報 \(294 ページ\)](#)
- [SDM テンプレートの機能履歴 \(294 ページ\)](#)

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 9300 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
- NAT

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

SDM テンプレートの設定方法

SDM テンプレートの設定

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	sdm prefer access nat 例： Device(config)# sdm prefer access	スイッチをアクセステンプレートに設定します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	reload 例： Device# reload	オペレーティング システムをリロードします。 システムの再起動後、 show sdm prefer 特権 EXEC コマンドを使用して、新しいテンプレート設定を確認できます。 reload 特権 EXEC コマンドを入力する前に、 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ー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ー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。



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

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

SDM テンプレートの設定例

例：SDM テンプレートの表示

次に、Cisco Catalyst 9300 シリーズ スイッチのアクセステンプレート情報を表示する出力例を示します。

```
Device# show sdm prefer access

This is the Access template.
Number of VLANs:                               4094
Unicast MAC addresses:                         32768
Overflow Unicast MAC addresses:                1024
L2 Multicast entries:                         8192
Overflow L2 Multicast entries:                 512
L3 Multicast entries:                         8192
Overflow L3 Multicast entries:                 512
Directly connected routes:                    24576
Indirect routes:                              8192
Security Access Control Entries:              5120
QoS Access Control Entries:                   5120
Policy Based Routing ACEs:                    1024
Netflow Input ACEs:                           256
Netflow Output ACEs:                          768
Ingress Netflow ACEs:                         256
Egress Netflow ACEs:                          768
Flow SPAN ACEs:                               1024
Tunnels:                                       512
LISP Instance Mapping Entries:                 512
Control Plane Entries:                        512
Input Netflow flows:                          32768
Output Netflow flows:                         32768
SGT/DGT (or) MPLS VPN entries:                8192
SGT/DGT (or) MPLS VPN Overflow entries:       512
Wired clients:                                2048
MACSec SPD Entries:                           256
```

例 : SDM テンプレートの表示

```

MPLS L3 VPN VRF:                255
MPLS Labels:                    2048
MPLS L3 VPN Routes VRF Mode:    7168
MPLS L3 VPN Routes Prefix Mode: 3072
MVPN MDT Tunnels:               256
L2 VPN EOMPLS Attachment Circuit: 256
MAX VPLS Bridge Domains :       128
MAX VPLS Peers Per Bridge Domain: 32
MAX VPLS/VPWS Pseudowires :     1024

```

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 9300 シリーズ スイッチの NAT テンプレート情報を表示する出力例を示します。

```
Device# show sdm prefer nat
```

```

This is the NAT template.
Number of VLANs:                4094
Unicast MAC addresses:          32768
Overflow Unicast MAC addresses: 1024
L2 Multicast entries:           8192
Overflow L2 Multicast entries:  512
L3 Multicast entries:           8192
Overflow L3 Multicast entries:  512
Directly connected routes:      24576
Indirect routes:                8192
Security Access Control Entries: 5120
QoS Access Control Entries:     1024
Policy Based Routing ACEs:      5120
Netflow Input ACEs:             256
Netflow Output ACEs:           768
Flow SPAN ACEs:                 1024
Tunnels:                        512
LISP Instance Mapping Entries:  512
Control Plane Entries:          512
Input Netflow flows:            32768
Output Netflow flows:           32768
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:                    2048
MPLS L3 VPN Routes VRF Mode:    7168
MPLS L3 VPN Routes Prefix Mode: 8192
MVPN MDT Tunnels:               256
L2 VPN EOMPLS Attachment Circuit: 256
MAX VPLS Bridge Domains :       128
MAX VPLS Peers Per Bridge Domain: 32
MAX VPLS/VPWS Pseudowires :     1024

```

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.

次の出力例は、Cisco Catalyst 9300 シリーズ スイッチの Cisco IOS XE Amsterdam 17.3.1 以降のリリースの C9300-24UB、C9300-24UXB、および C9300-48UB モデルでのアクセステンプレート情報を示しています。


```

Device# show sdm prefer access

Number of VLANs:                               4094
Unicast MAC addresses:                         49152
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
Security Access Control Entries:               18432
QoS Access Control Entries:                   6144
Policy Based Routing ACEs / NAT ACEs:         14336
Netflow Input ACEs:                           1024
Netflow Output ACEs:                          2048
Flow SPAN ACEs:                               1024
Tunnels:                                       1024
LISP Instance Mapping Entries:                 2048
Control Plane Entries:                        512
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
VRF:                                           256
MPLS Labels:                                  12288
MPLS L3 VPN Routes VRF Mode:                  32768
MPLS L3 VPN Routes Prefix Mode:              8192
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.

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。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

SDM テンプレートの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	SDM テンプレート	標準の SDM テンプレートを使用すると、システムリソースを設定して、特定の機能のサポートを最適化できます。
Cisco IOS XE Gibraltar 16.12.3	C9300-24UB、C9300-24UXB、および C9300-48UB デバイスのスケーラビリティメトリックの変更	<p>Cisco Catalyst 9300 シリーズ スイッチの C9300-24UB、C9300-24UXB、および C9300-48UB モデルで、次の機能の転送スケール番号が変更されました</p> <ul style="list-style-type: none"> レイヤ 2 ユニキャスト MAC アドレス : 49152 レイヤ 3 マルチキャスト : 32768 QoS アクセス コントロール エントリ : 6144 ポリシーベースルーティング ACE/NAT ACE : 14336

リリース	機能	機能情報
Cisco IOS XE Amsterdam 17.3.1	C9300-24UB、C9300-24UXB、および C9300-48UB デバイスのスケーラビリティメトリックの変更	Cisco Catalyst 9300 シリーズ スイッチの C9300-24UB、C9300-24UXB、および C9300-48UB モデルで、次の機能の転送スケール番号が変更されました <ul style="list-style-type: none">レイヤ2ユニキャストMACアドレス : 49152レイヤ3 マルチキャスト : 32768QoS アクセスコントロールエントリ : 6144ポリシーベースルーティングACE/NAT ACE : 14336

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 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 (297 ページ)
- システム メッセージ ログの設定方法 (300 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 (309 ページ)
- システム メッセージ ログの設定例 (310 ページ)
- システム メッセージ ログに関する追加情報 (311 ページ)
- システムメッセージログの機能履歴 (311 ページ)

システム メッセージ ログの設定に関する情報

システム メッセージ ロギング

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

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

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

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

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

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



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

システム ログ メッセージのフォーマット

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

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

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

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

表 19: システム ログ メッセージの要素

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

要素	説明
<p><i>timestamp</i> formats: <i>mm/dd h h:mm:ss</i> または <i>hh:mm:ss</i> (短時間) または <i>d h</i> (長時間)</p>	メッセージまたはイベントの日時です。この情報が表示されるのは、 service timestamps log[datetime log] グローバル コンフィギュレーション コマンドが設定されている場合のみです。
<i>facility</i>	メッセージが参照する機能 (SNMP、SYS など) です。
<i>severity</i>	メッセージの重大度を示す 0 ~ 7 の 1 桁のコードです。
<i>MNEMONIC</i>	メッセージを一意に示すテキスト ストリングです。
<i>description</i>	レポートされているイベントの詳細を示すテキスト ストリングです。
<i>hostname-n</i> (ホスト名 -n)	スタック メンバーのホスト名およびスタック内のスイッチ番号。アクティブスイッチはスタックメンバですが、そのホスト名はシステムメッセージの末尾に追加されません。

デフォルトのシステムメッセージロギングの設定

表 20: 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

機能	デフォルト設定
サーバ機能	local7
サーバの重大度	通知

syslog メッセージの制限

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

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

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

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

システムメッセージログの設定方法

メッセージ表示宛先デバイスの設定

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

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバルコンフィギュレーションモードを開始します。
ステップ 2	logging buffered [size] 例：	スイッチ上、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

	コマンドまたはアクション	目的
	Device(config)# logging buffered 8192	<p>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。</p> <p>スタンドアロンスイッチに障害が発生すると、ログファイルをフラッシュメモリに保存していなかった場合、ログファイルは失われます。ステップ4を参照してください。</p> <p>(注) バッファサイズを大きすぎる値に設定しないでください。他の作業に使用するメモリが不足することがあります。スイッチ上の空きプロセスメモリを表示するには、show memory 特権 EXEC コマンドを使用します。ただし、表示される値は使用できる最大値であるため、バッファサイズをこの値に設定しないでください。</p>
ステップ 3	logging host 例 : Device(config)# logging 125.1.1.100	<p>UNIX Syslog サーバホストにメッセージを保存します。</p> <p><i>host</i> には、syslog サーバとして使用するホストの名前または IP アドレスを指定します。</p> <p>ログメッセージを受信する Syslog サーバのリストを作成するには、このコマンドを複数回入力します。</p>
ステップ 4	logging file flash: filename [max-file-size [min-file-size]] [severity-level-number type] 例 : Device(config)# logging file flash:log_msg.txt 40960 4096 3	<p>スタンドアロンスイッチ上で、フラッシュメモリにあるファイルにログメッセージを保存します。</p> <ul style="list-style-type: none"> • <i>filename</i> : ログメッセージのファイル名を入力します。 • (任意) max-file-size —には、ログファイルの最大サイズを指定します。指定できる範囲は4096 ~ 2147483647 です。デフォルトは4096 バイトです。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • (任意) <i>min-file-size</i> : ログファイルの最小サイズを指定します。指定できる範囲は 1024 ~ 2147483647 です。デフォルトは 2048 バイトです。 • (任意) <i>severity-level-number type</i> : ログの重大度またはログタイプを指定します。重大度に指定できる範囲は 0 ~ 7 です。
ステップ 5	end 例 : Device(config)# end	特権 EXEC モードに戻ります。
ステップ 6	terminal monitor 例 : 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。 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、ユーザ入力プロンプトが返された後に、コンソールに表示されます。したがって、非送信請求メッセージおよび **debug** コマンドの出力は、送信請求デバイス出力およびプロンプトに割り込まれることはありません。非送信請求メッセージが表示された後に、コンソールはユーザプロンプトを再表示します。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	line [console vty] line-number [ending-line-number] 例 : Device(config)# line console	メッセージの同期ロギングに設定する回線を指定します。 <ul style="list-style-type: none"> • console : スイッチ コンソール ポートまたはイーサネット管理ポートでの設定を指定します。 • line vty line-number : どの vty 回線の同期ロギングをイネーブルにするかを指定します。Telnet セッションを介して行われる設定には、vty 接続を使用します。回線番号に指定できる範囲は 0 ~ 15 です。 16 個の vty 回線の設定をすべて一度に変更するには、次のように入力します。 line vty 0 15 また、現在の接続に使用されている 1 つの vty 回線の設定を変更することもできます。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。 line vty 2 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] 例 : 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。 <ul style="list-style-type: none"> • (任意) level severity-level : メッセージの重大度レベルを指定します。重大度がこの値以上であるメッセージは、非同期に出力されます。値が小さいほど重大度は大きく、値が大きいほど重大度は小さくなります。デフォルトは 2 です。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • (任意) level all : 重大度に関係なく、すべてのメッセージが非同期に出力されます。 • (任意) limit number-of-buffers : キューイングされる端末のバッファ数を指定します。これを超える新しいメッセージは廃棄されます。指定できる範囲は 0 ~ 2147483647 です。デフォルトは 20 です。
ステップ 4	end 例 : 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 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	no logging console 例： Device(config)# no logging console	メッセージ ロギングをディセーブルにします。
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。

ログメッセージのタイムスタンプのイネーブル化およびディセーブル化

デフォルトでは、ログメッセージにはタイムスタンプが適用されません。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	次のいずれかのコマンドを使用します。 <ul style="list-style-type: none"> • service timestamps log uptime • service timestamps log datetime[msec localtime show-timezone] 例： Device (config)# service timestamps log uptime または Device (config)# service timestamps log datetime	ログのタイムスタンプをイネーブルにします。 <ul style="list-style-type: none"> • log uptime : ログメッセージのタイムスタンプをイネーブルにして、システムの再起動以降の経過時間を表示します。 • log datetime : ログメッセージのタイムスタンプをイネーブルにします。選択したオプションに応じて、ローカル タイムゾーンを基準とした日付、時間 (ミリ秒)、タイム

	コマンドまたはアクション	目的
		ゾーン名をタイムスタンプとして表示できます。
ステップ 3	end 例： Device (config) # end	特権 EXEC モードに戻ります。

ログメッセージのシーケンス番号のイネーブル化およびディセーブル化

タイムスタンプが同じログメッセージが複数ある場合、これらのメッセージを表示するには、シーケンス番号を使用してメッセージを表示できます。デフォルトでは、ログメッセージにシーケンス番号は表示されません。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： 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 例： Device (config) # service sequence-numbers	シーケンス番号をイネーブルにします。
ステップ 3	end 例： Device (config) # end	特権 EXEC モードに戻ります。

メッセージ重大度の定義

メッセージの重大度を指定して、選択したデバイスに表示されるメッセージを制限します。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	logging console level 例： Device(config)# logging console 3	コンソールに保存するメッセージを制限します。 デフォルトで、コンソールはデバッグメッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 3	logging monitor level 例： Device(config)# logging monitor 3	端末回線に出力するメッセージを制限します。 デフォルトで、端末はデバッグメッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 4	logging trap level 例： Device(config)# logging trap 3	Syslog サーバに保存するメッセージを制限します。 デフォルトで、Syslog サーバは通知メッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 5	end 例： Device(config)# end	特権 EXEC モードに戻ります。

履歴テーブルおよび SNMP に送信される syslog メッセージの制限

このタスクでは、履歴テーブルおよび SNMP に送信される syslog メッセージを制限する方法について説明します。

このタスクはオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	logging history level 例： Device(config)# logging history 3	履歴ファイルに保存され、SNMP サーバに送信される syslog メッセージのデフォルト レベルを変更します。 デフォルトでは warnings 、 errors 、 critical 、 alerts 、および emergencies メッセージは送信されません。
ステップ 3	logging history size number 例： Device(config)# logging history size 200	履歴テーブルに保存できる Syslog メッセージの数を指定します。 デフォルトでは1つのメッセージが格納されます。指定できる範囲は0～500です。
ステップ 4	end 例： 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	<p>/etc/syslog.conf ファイルに次の行を追加します。</p> <p>例 :</p> <pre>local7.debug /usr/adm/logs/cisco.log</pre>	<ul style="list-style-type: none"> • local7 : ログ機能指定します。 • debug : syslog レベルを指定します。このファイルは、syslog デーモンに書き込み権限がある既存ファイルである必要があります。
ステップ 2	<p>UNIX シェルプロンプトに次のコマンドを入力します。</p> <p>例 :</p> <pre>\$ touch /var/log/cisco.log \$ chmod 666 /var/log/cisco.log</pre>	<p>ログファイルを作成します。syslog デーモンは、このレベルまたはこのファイルのより高い重大度レベルでメッセージを送信します。</p>
ステップ 3	<p>Syslog デーモンに新しい設定を認識させます。</p> <p>例 :</p> <pre>\$ kill -HUP `cat /etc/syslog.pid`</pre>	<p>詳細については、ご使用の UNIX システムの man syslog.conf および man syslogd コマンドを参照してください。</p>

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
<pre>show archive log config {all number [end-number] user username [session number] number [end-number] statistics} [provisioning]</pre>	<p>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。</p>

システムメッセージログの設定例

例：システムメッセージのスタック構成

次の例では、アクティブスイッチの部分的なスイッチシステムメッセージとスタックメンバ（ホスト名は *Switch-2*）を示します。

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)

00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1, changed
state to down 2 (Switch-2)
```

例：スイッチ システムメッセージ

次に、スイッチ上のスイッチ システムメッセージの一部を示します。

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

システムメッセージログに関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

システムメッセージログの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	システムメッセージログ	システムメッセージ出力は、ロギングプロセスに送信されます。ロギングプロセスはログメッセージを各宛先（設定に応じて、ログバッファ、端末回線、UNIX Syslog サーバーなど）に配信する処理を制御します

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



第 8 章

オンライン診断の設定

- [オンライン診断の設定に関する情報](#) (313 ページ)
- [オンライン診断の設定方法](#) (319 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](#) (325 ページ)
- [オンライン診断のコンフィギュレーション例](#) (325 ページ)
- [オンライン診断に関する追加情報](#) (327 ページ)
- [オンライン診断設定の機能情報](#) (328 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、データベー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、ヘルスマニタリング診断に分類できます。オンデマンド診断は、CLIから実行されます。スケジュールされた診断は、動作中のネットワークにデバイスが接続されているときに、ユーザが指定した間隔または指定した時刻に実行されます。ヘルスマニタリングは、バックグラウンドでユーザが指定した間隔で実行されます。ヘルスマニタリングテストは、テストに基づいて 90、100、または 150 秒ごとに実行されます。

オンライン診断を設定したあと、手動で診断テストを開始したり、テスト結果を表示したりできます。また、デバイス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。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。

DiagGoldPktTest

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	スーパーバイザ

DiagThermalTest

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

属性	説明
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マonitorリングテストとして実行します。
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	スーパーバイザ

DiagFanTest

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
推奨事項	ファンモジュールに問題が発生した場合は、ヘルスマonitor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	スーパーバイザ

DiagPhyLoopbackTest

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。ヘルスマonitorリングテストとして実行することはできません。



- (注) このテストがオンデマンドで実行される特定のケースでは、ポートは `error-disabled` ステートに移行します。このような場合は、インターフェイス コンフィギュレーション モードで `shut` および `no shut` コマンドを使用して、これらのポートを再度イネーブルにします。

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	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

DiagScratchRegisterTest

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 の正常性をモニターします。

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、ヘルスマニター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

DiagPoETest

このテストは、PoE コントローラ機能をチェックします。通常のスイッチ動作中は、このテストを実行しないでください。

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	このテストは、ポートで PoE コントローラの問題が発生した場合に実行します。これは、オンデマンドテストとしてのみ実行できます。
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	ラインカード

DiagStackCableTest

このテストは、スタック構成環境のスタックリンググループバック機能を検証します。ヘルスマニター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	このテストを実行し、スタック構成環境のスタックリンググループバック機能を検証します。
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	テストに失敗した場合は、スタックケーブルとコネクタを確認してください。
ハードウェア サポート	スーパーバイザ

DiagMemoryTest

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ーバイザエンジンをリ

属性	説明
	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	スーパーバイザ

TestUnusedPortLoopback

このテストは、実行時にスーパーバイザモジュールとモジュールのネットワークポート間のデータパスを定期的に確認し、着信ネットワーク インターフェイス ポートがロックされているかどうかを判断します。このテストでは、レイヤ2の packets はテストポートおよびスーパーバイザエンジンのインバンドポートに関連付けられた VLAN にフラディングされます。パケットはテストポート内をループバックして、同じ VLAN のスーパーバイザエンジンに戻ります。このテストは、ケーブルが接続されているかどうかに関係なく、未使用の（管理上のダウン、つまりポートがシャットダウンされている）ネットワークポートでのみ実行され、ポートあたり 1 ミリ秒以内に完了します。このテストは、現在の ASIC にノンディスラプティブループバック テストがないため、代用として使用され、60 秒ごとに実行されます。

属性	説明
ディスラプティブまたはノンディスラプティブ	ノンディスラプティブ
推奨事項	ディセーブルにしないでください。CPU 使用率の急上昇中、このテストは精度を維持するために自動的にディセーブルになります。
デフォルト	オン
最初のリリース	Cisco IOS XE Fuji 16.9.1
修正処置	ポートに障害が発生したことを示す syslog メッセージを表示します。スーパーバイザエンジン以外のモジュールでは、すべてのポートグループに障害が発生した場合（たとえば、ポート ASIC ごとに最低 1 つのポートで、すべてのポート ASIC の障害しきい値より多く障害が発生した場合）、デフォルトのアクションではモジュールがリセットされ、リセットを 2 回行ったあとにモジュールの電源を切断します。
ハードウェア サポート	スーパーバイザ

TestPortTxMonitoring

このテストは、ステータスが UP のデバイスに物理的に接続されている各ネットワークポートの送信方向のデータパストラフィックを定期的にモニターします。このテストは、ポートあたり1ミリ秒以内に完了します。また、このテストでは、ASIC レベルで送信カウンタをモニターして、ポートがスタックしていないことを確認します。テストではsyslogメッセージが表示され、ユーザーは Cisco IOS Embedded Event Manager (EEM) を使用して修正アクションを実行できます。

diagnostic monitor interval および **diagnostic monitor threshold** コマンドをそれぞれ入力して、時間間隔としきい値を設定します。テストでは、パケットを送信する Cisco Discovery Protocol を利用します。テストは 75 秒ごとに実行され、障害しきい値はデフォルトで 5 秒に設定されています。

属性	説明
ディスラプティブまたはノンディスラプティブ	ノンディスラプティブ
推奨事項	ディセーブルにしないでください。
デフォルト	オン
最初のリリース	Cisco IOS XE Everest 16.9.1
修正処置	ポートに障害が発生したことを示す syslog メッセージを表示します。
ハードウェア サポート	スーパーバイザ

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	<p>diagnostic start switch <i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic complete minimal non-disruptive per-port}</p> <p>例 :</p> <pre>Device# diagnostic start switch 2 test basic</pre>	<p>診断テストを開始します。</p> <p>次のいずれかのオプションを使用してテストを指定できます。</p> <ul style="list-style-type: none"> • name : テストの名前を入力します。 • test-id : テストの ID 番号を入力します。 • test-id-range : カンマとハイフンで区切ってテスト ID の範囲を整数で入力します。 • all : すべてのテストを開始します。 • basic : 基本テストスイートを開始します。 • complete : 完全なテストスイートを開始します。 • minimal : 最小限のブートアップテストスイートを開始します。 • non-disruptive : ノンディスラプティブテストスイートを開始します。 • per-port : ポート単位のテストスイートを開始します。

オンライン診断の設定

診断モニタリングをイネーブルにする前に、障害しきい値およびテストの間隔を設定する必要があります。

オンライン診断のスケジューリング

特定のデバイスについて指定した時間、または日、週、月単位でオンライン診断をスケジューリングできます。スケジューリングを削除するには、**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	<p>configure terminal</p> <p>例 :</p> <pre>Device #configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 2	<p>diagnostic schedule switch number test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic complete minimal non-disruptive per-port} {daily on <i>mm dd yyyy hh:mm</i> port <i>inter-port-number port-number-list</i> weekly <i>day-of-week hh:mm</i>}</p> <p>例 :</p> <pre>Device(config)# diagnostic schedule switch 3 test 1-5 on July 3 2013 23:10</pre>	<p>特定日時のオンデマンド診断テストをスケジュールします。</p> <p>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。</p> <ul style="list-style-type: none"> • name : show diagnostic content コマンドの出力に表示されるテストの名前です。 • test-id : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • test-id-range : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべてのテスト ID。 • basic : 基本的なオンデマンドの診断テストを開始します。 • complete : 完全なテストスイートを開始します。 • minimal : 最小限のブートアップテストスイートを開始します。 • non-disruptive : ノンディスラプティブテストスイートを開始します。 • per-port : ポート単位のテストスイートを開始します。 <p>テストは次のようにスケジュールできます。</p> <ul style="list-style-type: none"> • 毎日 : daily <i>hh:mm</i> パラメータを使用します。 • 特定日時 : on <i>mm dd yyyy hh:mm</i> パラメータを使用します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> 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 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	diagnostic monitor interval switch number test {name test-id test-id-range all} hh:mm:ss milliseconds day 例： Device(config)# diagnostic monitor interval switch 2 test 1 12:30:00 750 5	指定のテストに対し、ヘルスモニタリングの実行間隔を設定します。 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。 <ul style="list-style-type: none"> name：show diagnostic content コマンドの出力に表示されるテストの名前です。 test-id：show diagnostic content コマンドの出力に表示されるテストの ID 番号です。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • test-id-range : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべての診断テスト。 <p>間隔を指定する場合は、次のパラメータを設定します。</p> <ul style="list-style-type: none"> • hh:mm:ss : モニタリング間隔 (時間、分、秒)。指定できる範囲は <i>hh</i> が 0~24、<i>mm</i> および <i>ss</i> が 0~60 です。 • milliseconds : モニタリング間隔 (ミリ秒 (ms))。指定できる範囲は 0~999 です。 • day : モニタリング間隔 (日数)。指定できる範囲は 0~20 です。
<p>ステップ 4</p>	<p>diagnostic monitor syslog</p> <p>例 :</p> <pre>Device(config)# diagnostic monitor syslog</pre>	<p>(任意) ヘルスマニタリングテストの失敗時にスイッチが Syslog メッセージを生成するように設定します。</p>
<p>ステップ 5</p>	<p>diagnostic monitor threshold switch number number test {name test-id test-id-range all} failure count count</p> <p>例 :</p> <pre>Device(config)# diagnostic monitor threshold switch 2 test 1 failure count 20</pre>	<p>(任意) ヘルスマニタリングテストの失敗しきい値を設定します。</p> <p>テストを指定する場合は、次のいずれかのパラメータを使用します。</p> <ul style="list-style-type: none"> • name : show diagnostic content コマンドの出力に表示されるテストの名前です。 • test-id : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • test-id-range : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべての診断テスト。

	コマンドまたはアクション	目的
		失敗しきい値 <i>count</i> に指定できる範囲は 0 ~ 99 です。
ステップ 6	diagnostic monitor switchnumber test { <i>name</i> <i>test-id</i> <i>test-id-range</i> all } 例 : Device(config)# diagnostic monitor switch 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。 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。 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。 <ul style="list-style-type: none"> • name : show diagnostic content コマンドの出力に表示されるテストの名前です。 • test-id : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • test-id-range : show diagnostic content コマンドの出力に表示されるテストの ID 番号です。 • all : すべての診断テスト。
ステップ 7	end 例 : 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 例 : Device# show running-config	(任意) 入力を確認します。
ステップ 10	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

オンライン診断のモニタリングおよびメンテナンス

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 21: 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 switch [<i>number</i> 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。 switch [<i>number</i> all] パラメータは、スタック構成スイッチだけでサポートされます。
show diagnostic status	現在実行中の診断テストを表示します。
show diagnostic result switch [<i>number</i> all] [detail test { <i>name</i> <i>test-id</i> <i>test-id-range</i> all }; [detail]]	オンライン診断テストの結果を表示します。 switch [<i>number</i> all] パラメータは、スタック構成スイッチだけでサポートされます。
show diagnostic switch [<i>number</i> all] [detail]	オンライン診断テストの結果を表示します。 switch [<i>number</i> all] パラメータは、スタック構成スイッチだけでサポートされます。
show diagnostic schedule [<i>number</i> all]	オンライン診断テストのスケジュールを表示します。 switch [<i>number</i> all] パラメータは、スタック構成スイッチだけでサポートされます。
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 { <i>event-type</i> <i>module</i> }	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 [<i>number</i>] test { <i>name</i> <i>test-id</i> 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 switch 2 test DiagPOETest
```

次に、すべての基本診断テストを開始する例を示します。

```
Device# diagnostic start switch 1 test all
```

例：ヘルスマニターリングテストの設定

次に、ヘルスマニターリングテストを設定する例を示します。

```
Device(config)# diagnostic monitor threshold switch 1 test 1 failure count 50  
Device(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

例：診断テストのスケジューリング

次に、特定のスイッチに対して、特定の日に診断テストを実行するようにスケジューリングする例を示します。

```
Device(config)# diagnostic schedule test DiagThermalTest on June 3 2013 22:25
```

次の例では、指定されたスイッチで毎週特定の時間に診断テストを実行するようにスケジューリングする方法を示します。

```
Device(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

例：オンライン診断の表示

次に、オンデマンド診断設定を表示する例を示します。

```
Device# show diagnostic ondemand settings
```

```
Test iterations = 1  
Action on test failure = continue
```

次に、障害の診断イベントを表示する例を示します。

```
Device# show diagnostic events event-type error
```

```
Diagnostic events (storage for 500 events, 0 events recorded)  
Number of events matching above criteria = 0
```

```
No diagnostic log entry exists.
```

次に、診断テストの説明を表示する例を示します。

```
Device# show diagnostic description switch 1 test all

DiagGoldPktTest :
    The GOLD packet Loopback test verifies the MAC level loopback
    functionality. In this test, a GOLD packet, for which doppler
    provides the support in hardware, is sent. The packet loops back
    at MAC level and is matched against the stored packet. It is a non
    -disruptive test.

DiagThermalTest :
    This test verifies the temperature reading from the sensor is below the yellow
    temperature threshold. It is a non-disruptive test and can be run as a health
    monitoring test.

DiagFanTest :
    This test verifies all fan modules have been inserted and working properly on
    the board
    It is a non-disruptive test and can be run as a health monitoring test.

DiagPhyLoopbackTest :
    The PHY Loopback test verifies the PHY level loopback
    functionality. In this test, a packet is sent which loops back
    at PHY level and is matched against the stored packet. It is a
    disruptive test and cannot be run as a health monitoring test.

DiagScratchRegisterTest :
    The Scratch Register test monitors the health of application-specific
    integrated circuits (ASICs) by writing values into registers and reading
    back the values from these registers. It is a non-disruptive test and can
    be run as a health monitoring test.

DiagPoETest :
    This test checks the PoE controller functionality. This is a disruptive test
    and should not be performed during normal switch operation.

DiagMemoryTest :
    This test runs the exhaustive ASIC memory test during normal switch operation
    NG3K utilizes mbist for this test. Memory test is very disruptive
    in nature and requires switch reboot after the test.

Device#
```

オンライン診断に関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

オンライン診断設定の機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	オンライン診断	オンライン診断機能を使用すると、デバイスをアクティブ ネットワークに接続したまま、デバイスのハードウェア機能をテストして確認できます。

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



第 9 章

整合性チェッカー

- [整合性チェッカーの制限事項 \(329 ページ\)](#)
- [整合性チェッカーに関する情報 \(330 ページ\)](#)
- [整合性チェッカーの実行 \(330 ページ\)](#)
- [整合性チェッカーの機能履歴 \(331 ページ\)](#)

整合性チェッカーの制限事項

整合性チェッカーには次の制限事項があります。

- 整合性チェッカーは CPU 集約型です。短すぎる間隔でチェッカーを実行することは推奨されません。
- 整合性チェッカーはスナップショットをサポートしていません。したがって、以前の実行は表示できません。
- すでに実行中の整合性チェッカーを停止/中止するコマンドはありません。
- 転送エンジンのハードウェアエントリの検証は部分的に実装されます。プログラミングの失敗のみを検出して報告できます。
- レイヤ 2 MAC 整合性チェッカーは、ソフトウェアコピーを使用してハードウェアの MAC アドレスを検証できます。
- 整合性チェッカーは、すべてのケースで誤検出を減らすように設計されています。ただし、次のシナリオではまれに誤検出が報告されることがあります。
 - 大規模なテーブル状態の変更（クリア、再学習など）。
 - 整合性チェッカーの実行中に、他の機能が原因で CPU 使用率が非常に高くなった場合。整合性チェッカーが、CPU 使用率が高いプロセスの不整合を報告する場合があります。
- レイヤ 3 マルチキャスト整合性チェッカーでは、転送エンジンのハードウェア（FED）チェックは完全にはサポートされていません。プログラミングエラーの検出と報告のみ実行できます。

- レイヤ3 マルチキャスト整合性チェッカーでは、転送 Manager-RP ソフトウェアのエントリはサポートされていません。

整合性チェッカーに関する情報

整合性チェッカーの概要

整合性チェッカーは、ソフトウェアおよびハードウェア内のさまざまなテーブルの状態に関する情報を収集します。ソフトウェアの状態とハードウェアの状態を比較します。不整合がある場合は、ただちに問題にフラグが付けられます。これにより、後のトラブルシューティングの時間を短縮できます。整合性チェッカーは、基本的なトラブルシューティングを補足するもので、ソフトウェアテーブルとハードウェアテーブル間の不整合な状態がネットワークの問題を引き起こしているシナリオを特定するのに役立ちます。これにより、問題を解決するための平均時間が短縮されます。

整合性チェッカーでは、コントロールプレーンから転送エンジン（またはハードウェアコピー）へのエントリの検証をサポートします。

整合性チェッカーでサポートされる機能

整合性チェッカーでは次の機能がサポートされています。

- レイヤ2 MAC 整合性チェッカー**：この整合性チェッカーは、IOS エントリから FED ソフトウェアエントリを検証します。また、ハードウェアテーブルの MAC アドレスを検証します。
- レイヤ3 FMANFP エントリ整合性チェッカー**：この整合性チェッカーは、転送マネージャ FP プロセスのレイヤ2、レイヤ3、およびマルチキャストオブジェクトのステータスを検証します。これには、古いオブジェクトと長期間保留中のオブジェクトが含まれます。

整合性チェッカーの実行

次の表は、さまざまな整合性チェッカーを実行するコマンドを示します。

コマンド	目的
show consistency-checker l2	レイヤ2 転送テーブルで consistency-checker を実行します。
show consistency-checker l3	レイヤ3 転送テーブルで consistency-checker を実行します。
show consistency-checker switch	指定したスイッチで consistency-checker を実行します。

整合性チェッカーの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Amsterdam 17.3.1	整合性チェッカー	整合性チェッカーは、ソフトウェアおよびハードウェア内のさまざまなテーブルの状態に関する情報を収集し、不整合が検出されるとすぐにフラグを付けます。これは、基本的なトラブルシューティングを補足するもので、ソフトウェアテーブルとハードウェアテーブル間の不整合な状態がネットワークの問題を引き起こしているシナリオを特定するのに役立ちます。これにより、問題を解決するための平均時間が短縮されます。

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



第 10 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](#) (333 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](#) (333 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](#) (334 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](#) (342 ページ)
- [コンフィギュレーション ファイルの管理の機能履歴](#) (373 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

- ユーザーには、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だけ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 \(343 ページ\)](#) 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、ファイルサーバー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 サーバー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ー (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 サーバーからデバイスへのコンフィギュレーション ファイルのコピー](#) の項を参照してください。

CLI を使用したコンフィギュレーション ファイルの変更

Cisco IOS ソフトウェアは、1 行につき 1 つのコンフィギュレーション コマンドを受け入れません。コンフィギュレーション コマンドは、必要なだけ入力できます。コンフィギュレーション ファイルには、入力したコマンドを説明するコメントを追加できます。コメントの先頭には、感嘆符 (!) を付けます。コメントは NVRAM にもコンフィギュレーション ファイルのアクティブコピーにも格納されないため、**show running-config** または **more system:running-config EXEC** コマンドでアクティブな設定のリストを表示しても、コメントは表示されません。**show startup-config** または **more nvram:startup-config EXEC** モードコマンドでスタートアップ コンフィギュレーションのリストを表示しても、コメントは表示されません。コメントは、コンフィギュレーション ファイルがデバイスにロードされたときにコンフィギュレーション ファイルから削除されます。ただし、ファイル転送プロトコル (FTP)、リモートコピープロトコル (RCP)、または Trivial File Transfer Protocol (TFTP) サーバー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、ユーザー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 \(367 ページ\)](#) 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ー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サーバー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ー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。

サーバー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、使用中のサーバー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 サーバー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 サーバー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 サーバー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 サーバー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、ユーザー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ー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 サーバー (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ー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ーへのアクセスだけです (ほとんどの UNIX システムが rsh をサポートしています)。ある場所から別の場所にファイルをコピーするため、コピー元のファイルに対する読み取り権限とコピー先のファイルに対する書き込み権限が必要です。コピー先ファイルが存在しない場合は、rcp により作成されます。

シスコの rcp 実装は UNIX の rcp 実装 (ネットワーク上のシステム間でファイルをコピー) の関数をエミュレートしたのですが、シスコのコマンド構文は UNIX の rcp コマンド構文とは異なります。シスコの rcp サポートは、rcp をトランスポートメカニズムとして使用する一連の **copy** コマンドを提供しています。これらの **rcp copy** コマンドは、シスコの TFTP **copy** コマンドに類似していますが、高速で信頼性の高いデータ配信を実現する代替方法を備えているという点が異なります。これらの改善は、rcp のトランスポートメカニズムがコネクション型の TCP/IP スタック上に構築されており、これを使用しているために可能になりました。rcp コマンドを使用して、デバイスからネットワークサーバー (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ー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。

リモートユーザー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ー名をサーバー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ー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ー名を送信します。

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ー名 (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ー名。たとえば、ユーザーが 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、リモートユーザー名として Telnet ユーザー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 サーバー上にリモートユーザー名のアカウントを定義する必要があります。このサーバー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、サーバー上のリモートユーザー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ー上のユーザーのホーム ディレクトリにある場合は、そのユーザーの名前をリモートユーザー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ー名を指定します。(rcmd は、スーパーユーザー 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ー名を指定する場合は、**copy** コマンド内でユーザー名を指定します。

サーバーに書き込む場合、デバイス上のユーザーからの RCP 書き込み要求を受け入れるように、RCP サーバー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 サーバー上のリモートユーザー用の .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 サーバー上の User0 の .rhosts ファイルには、次の行が含まれることとなります。

```
Device1.example.com Device1
```

RCP ユーザー名に関する要件

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ー名をサーバー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ー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ー名を送信します。

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ー名 (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ー名。たとえば、ユーザーが 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、リモートユーザー名として Telnet ユーザー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 サーバー上にリモート ユーザー名のアカウントを定義する必要があります。このサーバー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、サーバー上のリモート ユーザー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ー上のユーザーのホームディレクトリにある場合は、そのユーザーの名前をリモート ユーザー名として指定します。

詳細については、ご使用の RCP サーバー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: flash-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 がいない場合だけ必要です。ブート ROM が圧縮コンフィギュレーションを認識しない場合は、次のメッセージが表示されます。

```
Boot ROMs do not support NVRAM compression Config NOT written to NVRAM
```

コンフィギュレーションのクラス A フラッシュ ファイル システム上のフラッシュ メモリへの格納

クラス A フラッシュファイルシステムのデバイス上では、内部フラッシュメモリのファイルまたは PCMCIA スロットのフラッシュメモリのファイルに **CONFIG_FILE** 環境変数を設定することにより、スタートアップ コンフィギュレーションをフラッシュメモリに格納できます。

詳細については、[クラス A フラッシュ ファイル システムでの CONFIG_FILE 環境変数の指定 \(367 ページ\)](#) 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）は自動的に行われなため、利用可能なフラッシュメモリに十分注意を払う必要があります。**squeeze** コマンドを使用して、使用済み領域を再要求します。20 MB 以上の大容量フラッシュカードを使用することを推奨します。

ネットワークからのコンフィギュレーション コマンドのロード

コンフィギュレーションが大きい場合は、FTP、RCP、TFTP のいずれかのサーバーに格納しておき、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ー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 サーバーへのコンフィギュレーションファイルのコピー \(345 ページ\)](#) 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 \(342 ページ\)](#) 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ー上にあり、読み取り可能である必要があります。

コンフィギュレーションファイル情報の管理方法

コンフィギュレーションファイル情報の表示

コンフィギュレーションファイルに関する情報を表示するには、このセクションの手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show boot 例： Device# show boot	BOOT環境変数の内容（設定されている場合）、CONFIG_FILE環境変数によって指定されているコンフィギュレーションファイルの名前、およびBOOTLDR環境変数の内容を示します。

	コマンドまたはアクション	目的
ステップ 3	more <i>file-url</i> 例 : Device# more 10.1.1.1	指定されたファイルの内容を表示します。
ステップ 4	show running-config 例 : Device# show running-config	実行コンフィギュレーション ファイルの内容を表示します (more system:running-config コマンドのコマンドエイリアスです) 。
ステップ 5	show startup-config 例 : Device# show startup-config	スタートアップ コンフィギュレーション ファイルの内容を表示します。 (more nvram:startup-config コマンドのコマンドエイリアスです) 。
		クラス A フラッシュ ファイル システム プラットフォーム以外のすべてのプラットフォーム上では、通常、デフォルトの startup-config ファイルは NVRAM に格納されます。 クラス A フラッシュ ファイル システム プラットフォーム上では、 CONFIG_FILE 環境変数はデフォルトの startup-config ファイルを指定します。 CONFIG_FILE 変数のデフォルトは NVRAM になります。

コンフィギュレーション ファイルの変更

Cisco IOS ソフトウェアは、1 行につき 1 つのコンフィギュレーション コマンドを受け入れます。コンフィギュレーション コマンドは、必要なだけ入力できます。コンフィギュレーション ファイルには、入力したコマンドを説明するコメントを追加できます。コメントの先頭には、感嘆符 (!) を付けます。コメントは NVRAM にもコンフィギュレーション ファイルのアクティブコピーにも格納されないため、**show running-config** または **more system:running-config EXEC** コマンドでアクティブな設定のリストを表示しても、コメントは表示されません。**show startup-config** または **more nvram:startup-config EXEC** モードコマンドでスタートアップ コンフィギュレーションのリストを表示しても、コメントは表示されません。コメントは、コンフィギュレーション ファイルがデバイスにロードされたときにコンフィギュレーション ファイルから削除されます。ただし、ファイル転送プロトコル (FTP)、リモートコピープロトコル (RCP)、または Trivial File Transfer Protocol (TFTP) サーバー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、ユーザー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	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>configuration command</p> <p>例 :</p> <pre>Device(config)# configuration command</pre>	<p>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。</p>
ステップ 4	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> • end • ^Z <p>例 :</p> <pre>Device(config)# end</pre>	<p>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。</p> <p>(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。</p>
ステップ 5	<p>copy system:running-config nvram:startup-config</p> <p>例 :</p> <pre>Device# copy system:running-config nvram:startup-config</pre>	<p>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。</p> <p>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、</p>

	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 サーバー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 サーバー上の設定をコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
ステップ 2	copy system:running-config tftp: [[[//location]/directory]/filename] 例 : Device# copy system:running-config tftp: //server1/topdir/file10	TFTP サーバーへ実行コンフィギュレーションファイルをコピーします。
ステップ 3	copy nvram:startup-config tftp: [[[//location]/directory]/filename] 例 : Device# copy nvram:startup-config tftp: //server1/1stidir/file10	TFTP サーバーへスタートアップコンフィギュレーションファイルをコピーします。

例

次に、デバイスから TFTP サーバーへコンフィギュレーションファイルをコピーする例を示します。

```
Device# copy system:running-config tftp://172.16.2.155/tokyo-config
Write file tokyo-config on host 172.16.2.155? [confirm] Y
Writing tokyo-config!!! [OK]
```

次の作業

copy コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーションコマンドの現在の設定によって異なります。

デバイスから RCP サーバー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 サーバーへスタートアップコンフィギュレーションファイルまたは実行コンフィギュレーションファイルをコピーするには、特権 EXEC モードを開始して次のコマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip rcmd remote-username username 例 : Device(config)# ip rcmd remote-username NetAdmin1	(任意) デフォルトのリモートユーザー名を変更します。
ステップ 4	end 例 : Device(config)# end	(任意) グローバル コンフィギュレーション モードを終了します。
ステップ 5	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy system:running-config rcp: [[[/[username@]location]/directory]/filename] • copy nvram:startup-config rcp: [[[/[username@]location]/directory]/filename] 例 : Device# copy system:running-config rcp://NetAdmin1@example.com/dir-files/file1	<ul style="list-style-type: none"> • デバイスの実行コンフィギュレーションファイルが RCP サーバー上に格納されるように指定します。 または • デバイスのスタートアップコンフィギュレーションファイルが RCP サーバー上に格納されるように指定します。

例

RCP サーバーへの実行コンフィギュレーション ファイルの格納

次に、rtr2-config という名前の実行コンフィギュレーションファイルを IP アドレス 172.16.101.101 のリモート ホスト上の netadmin1 ディレクトリにコピーする例を示します。

```
Device# copy system:running-config rcp://netadmin1@172.16.101.101/runfile2-config
Write file runfile2-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

RCP サーバー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、サーバー上にスタートアップ コンフィギュレーション ファイルを格納する例を示します。

```

Device# configure terminal

Device(config)# ip rcmd remote-username netadmin2

Device(config)# end

Device# copy nvram:startup-config rcp:

Remote host[ ]? 172.16.101.101

Name of configuration file to write [start-config]?
Write file start-config on host 172.16.101.101?[confirm]
![OK]
    
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

デバイスから FTP サーバー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 サーバーへスタートアップ コンフィギュレーション ファイルまたは実行コンフィギュレーション ファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	デバイスでグローバルコンフィギュレーション モードを開始します。
ステップ 3	ip ftp username <i>username</i> 例： Device(config)# ip ftp username NetAdmin1	(任意) デフォルトのリモートユーザー名を指定します。
ステップ 4	ip ftp password <i>password</i> 例：	(任意) デフォルトのパスワードを指定します。

	コマンドまたはアクション	目的
	Device(config)# ip ftp password adminpassword	
ステップ 5	end 例： Device(config)# end	(任意) グローバル コンフィギュレーションモードを終了します。この手順は、デフォルトのリモートユーザー名またはパスワードを上書きする場合にだけ必要です (ステップ 2 および 3 を参照)。
ステップ 6	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy system:running-config ftp: [[[/[username [:password]@]location]/directory]/filename] または • copy nvram:startup-config ftp: [[[/[username [:password]@]location]/directory]/filename] 例： Device# copy system:running-config ftp:	FTP サーバー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 サーバーへの実行コンフィギュレーション ファイルの格納

次に、runfile-config という名前の実行コンフィギュレーション ファイルを IP アドレス 172.16.101.101 のリモート ホスト上の netadmin1 ディレクトリにコピーする例を示します。

```
Device# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/runfile-config
Write file runfile-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

FTP サーバー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、サーバー上にスタートアップ コンフィギュレーション ファイルを格納する例を示します。

```
Device# configure terminal
Device(config)# ip ftp username netadmin2
Device(config)# ip ftp password mypass
Device(config)# end
```

```
Device# copy nvram:startup-config ftp:

Remote host[]? 172.16.101.101

Name of configuration file to write [start-config]?
Write file start-config on host 172.16.101.101?[confirm]
![OK]
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

TFTP サーバー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 サーバーからデバイスへコンフィギュレーションファイルをコピーするには、以下のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy tftp: [///location]directory[/filename] system:running-config 例： Device# copy tftp://server1/dir10/datasource system:running-config	TFTP サーバーから実行コンフィギュレーションへコンフィギュレーションファイルをコピーします。
ステップ 3	copy tftp: [///location]directory[/filename] nvram:startup-config 例： Device# copy tftp://server1/dir10/datasource nvram:startup-config	TFTP サーバーからスタートアップコンフィギュレーションへコンフィギュレーションファイルをコピーします。
ステップ 4	copy tftp: [///location]directory[/filename] flash-nvram:directory/startup-config 例： Device# copy	TFTP サーバーからスタートアップコンフィギュレーションへコンフィギュレーションファイルをコピーします。

	コマンドまたはアクション	目的
	tftp://server1/dir10/datasource flash:startup-config	

例

次に、IP アドレス 172.16.2.155 にある、**tokyo-config** という名前のファイルからソフトウェアを設定する例を示します。

```
Device# copy tftp://172.16.2.155/tokyo-config system:running-config
Configure using tokyo-config from 172.16.2.155? [confirm] Y
Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

rcp サーバー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 サーバーから実行コンフィギュレーションまたはスタートアップ コンフィギュレーションへコンフィギュレーション ファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	（任意）端末からコンフィギュレーション モードを開始します。この手順は、デフォルトのリモートユーザー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 例：	（任意）リモート ユーザー名を指定します。

	コマンドまたはアクション	目的
	Device(config)# ip rcmd remote-username NetAdmin1	
ステップ 4	end 例： Device(config)# end	(任意) グローバル コンフィギュレーション モードを終了します。この手順は、デフォルトのリモートユーザー名またはパスワードを上書きする場合にだけ必要です (ステップ 2 を参照)。
ステップ 5	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy <code>rcp://[username@[hostname]]/remote:systemconfig</code> • copy <code>rcp://[username@[hostname]]/remote:startupconfig</code> 例： Device# copy rcp://[user1@example.com/dir10/fileone] nvram:startup-config	rcp サーバーから実行コンフィギュレーションまたはスタートアップコンフィギュレーションへコンフィギュレーションファイルをコピーします。

例

rcp の Running-Config のコピー

次に、host1-config という名前のコンフィギュレーションファイルを、IP アドレスが 172.16.101.101 のリモートサーバー上の netadmin1 ディレクトリからコピーし、デバイスでコマンドをロードして実行する例を示します。

```
device# copy rcp://netadmin1@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:![OK]
device#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

rcp の Startup-Config のコピー

次に、リモートユーザー名 netadmin1 を指定する例を示します。次に host2-config という名前のコンフィギュレーションファイルを、IP アドレスが 172.16.101.101 のリモートサーバー上の netadmin1 ディレクトリからスタートアップコンフィギュレーションへコピーします。

```
device# configure terminal
device(config)# ip rcmd remote-username netadmin1
device(config)# end
device# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
```

```

Loading 1112 byte file host2-config:[OK]
[OK]
device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by 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 サーバー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 サーバーから実行コンフィギュレーションまたはスタートアップコンフィギュレーションへコンフィギュレーションファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	(任意) グローバル コンフィギュレーション モードを開始できます。この手順は、デフォルトのリモートユーザー名またはパスワードを上書きする場合にだけ必要です (ステップ 3 および 4 を参照)。
ステップ 3	ip ftp username <i>username</i> 例 : Device(config)# ip ftp username NetAdmin1	(任意) デフォルトのリモートユーザー名を指定します。
ステップ 4	ip ftp password <i>password</i> 例 : Device(config)# ip ftp password adminpassword	(任意) デフォルトのパスワードを指定します。
ステップ 5	end 例 :	(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ー名

	コマンドまたはアクション	目的
	Device(config)# end	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 6	次のいずれかを実行します。 <ul style="list-style-type: none"> • copy ftp: [[[/[username[:password]@]location] /directory]/filename]system:running-config • copy ftp: [[[/username[:password]@]location]name:startup-config 例： Device# copy ftp:nvram:startup-config	FTPを使用して、ネットワークサーバーから実行メモリまたはスタートアップコンフィギュレーションへコンフィギュレーションファイルをコピーします。

例

FTP の Running-Config のコピー

次に、host1-config という名前のホスト コンフィギュレーション ファイルを、IP アドレスが 172.16.101.101 のリモートサーバー上の netadmin1 ディレクトリからコピーし、デバイスでコマンドをロードして実行する例を示します。

```
device# copy ftp://netadmin1:mypass@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:![OK]
device#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

FTP の Startup-Config のコピー

次に、リモートユーザー名 netadmin1 を指定する例を示します。次に host2-config という名前のコンフィギュレーション ファイルを、IP アドレスが 172.16.101.101 のリモートサーバー上の netadmin1 ディレクトリからスタートアップ コンフィギュレーションへコピーします。

```
device# configure terminal
device(config)# ip ftp username netadmin1
device(config)# ip ftp password mypass
device(config)# end
device# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[host1-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:![OK]
[OK]
device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from 172.16.101.101
```

次の作業

copy EXEC コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **file prompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

NVRAM より大きいコンフィギュレーションファイルの保守

NVRAMのサイズを超えるコンフィギュレーションファイルを保守するには、以降のセクションで説明するタスクを実行します。

コンフィギュレーションファイルの圧縮

コンフィギュレーションファイルを圧縮するには、このセクションの手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	service compress-config 例： Device(config)# service compress-config	コンフィギュレーション ファイルを圧縮することを指定します。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了します。
ステップ 5	次のいずれかを実行します。 • 新しいコンフィギュレーションをコピーするには、FTP、RCP、TFTP を使用します。 • configure terminal 例：	新しいコンフィギュレーションを入力します。 • NVRAM のサイズの 3 倍以上のコンフィギュレーションをロードしようとすると、次のエラーメッセージが表示されます。

	コマンドまたはアクション	目的
	Device# <code>configure terminal</code>	「[buffer overflow - file-size /buffer-size bytes]。」
ステップ 6	copy system:running-config nvram:startup-config 例 : Device(config)# <code>copy system:running-config nvram:startup-config</code>	実行コンフィギュレーションの変更が終わったら、新しいコンフィギュレーションを保存します。

例

次に、129 KB のコンフィギュレーションファイル を 11 KB に圧縮する例を示します。

```
Device# configure terminal

Device(config)# service compress-config

Device(config)# end

Device# copy tftp://172.16.2.15/tokyo-config system:running-config

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

Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
Device# copy system:running-config nvram:startup-config

Building configuration...
Compressing configuration from 129648 bytes to 11077 bytes
[OK]
```

コンフィギュレーションのクラス A フラッシュ ファイル システム上のフラッシュ メモリへの格納

スタートアップ コンフィギュレーションをフラッシュ メモリに格納するには、このセクションの手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> <code>enable</code>	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	<p>copy nvram:startup-config <i>flash-filesystem:filename</i></p> <p>例 :</p> <pre>Device# copy nvram:startup-config usbflash0:switch-config</pre>	<p>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。</p>
ステップ 3	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 4	<p>boot config flash-filesystem: filename</p> <p>例 :</p> <pre>Device(config)# boot config usbflash0:switch-config</pre>	<p>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。</p>
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>グローバル コンフィギュレーション モードを終了します。</p>
ステップ 6	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> 新しいコンフィギュレーションをコピーするには、FTP、RCP、TFTP を使用します。NVRAM サイズの 3 倍を超える大きさのコンフィギュレーションをロードしようとする と、次のエラー メッセージが表示 されます。「[buffer overflow - file-size /buffer-size bytes]」 configure terminal <p>例 :</p> <pre>Device# configure terminal</pre>	<p>新しいコンフィギュレーションを入力し ます。</p>
ステップ 7	<p>copy system:running-config nvram:startup-config</p> <p>例 :</p> <pre>Device(config)# copy system:running-config nvram:startup-config</pre>	<p>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。</p>

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ーを使用して、大きなコンフィギュレーションを保存するには、このセクションの手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	copy system:running-config {ftp: rcp: tftp:} 例 : Device# copy system:running-config ftp:	実行コンフィギュレーションを FTP、RCP、TFTP のいずれかのサーバーに保存します。
ステップ 3	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 4	boot network {ftp:[[[[username[:password]@]location]/directory]/filename] rcp:[[[[username@]location]/directory]/filename] tftp:[[[[location]/directory]/filename]} 例 : Device(config)# boot network ftp://user1:guessme@example.com/dir10/file1	起動時にスタートアップ コンフィギュレーション ファイルをネットワークサーバーからロードすることを指定します。

	コマンドまたはアクション	目的
ステップ 5	service config 例 : Device(config)# service config	システムの起動時にコンフィギュレーションファイルをダウンロードするようにスイッチをイネーブルにします。
ステップ 6	end 例 : Device(config)# end	グローバル コンフィギュレーションモードを終了します。
ステップ 7	copy system:running-config nvram:startup-config 例 : Device# copy system:running-config nvram:startup-config	設定を保存します。

フラッシュメモリからスタートアップまたは実行コンフィギュレーションへのコンフィギュレーションファイルのコピー

フラッシュメモリから現在の NVRAM にあるスタートアップ コンフィギュレーションまたは実行コンフィギュレーションへコンフィギュレーションファイルを直接コピーするには、ステップ 2 のいずれかのコマンドを入力します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを実行します。 • copy filesystem: [partition-number:][filename] nvram:startup-config • copy filesystem: [partition-number:][filename] system:running-config 例 : Device# copy usbflash0:4:ios-upgrade-1 nvram:startup-config	• NVRAM にコンフィギュレーションファイルを直接ロードする、または • 現在の実行コンフィギュレーションにコンフィギュレーションファイルをコピーします。

例

次に、usbflash0 にあるフラッシュメモリ PC カードのパーティション 4 からデバイスのスタートアップ コンフィギュレーションへ ios-upgrade-1 という名前のファイルをコピーする例を示します。

```
Device# copy usbflash0:4:ios-upgrade-1 nvram:startup-config
Copy 'ios-upgrade-1' from flash device as 'startup-config' ? [yes/no] yes
[OK]
```

フラッシュメモリ ファイル システム間でのコンフィギュレーション ファイルのコピー

複数のフラッシュメモリファイルシステムを備えたプラットフォーム上では、内部フラッシュメモリなどのフラッシュメモリファイルシステムから他のフラッシュメモリファイルシステムへファイルをコピーできます。異なるフラッシュメモリファイルシステムへファイルをコピーすることで、使用中のコンフィギュレーションのバックアップコピーを作成し、他のデバイスにコンフィギュレーションを複製できます。フラッシュメモリファイルシステム間でコンフィギュレーションファイルをコピーするには、EXECモードで次のコマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show source-filesystem: 例： Device# show flash:	フラッシュメモリのレイアウトと内容を表示して、ファイル名を確認します。
ステップ 3	copy source-filesystem: [partition-number:][filename] dest-filesystem:[partition-number:][filename] 例： Device# copy flash: usbflash0:	フラッシュメモリ デバイス間でコンフィギュレーション ファイルをコピーします。 • 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。

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例： Device# configure terminal	(任意) グローバル コンフィギュレーション モードを開始します。この手順は、デフォルトのリモート ユーザー名またはパスワードを上書きする場合にだけ必要です (ステップ 3 および 4 を参照)。
ステップ 3	ip ftp username <i>username</i> 例： Device(config)# ip ftp username Admin01	(任意) リモート ユーザー名を指定します。
ステップ 4	ip ftp password <i>password</i> 例： Device(config)# ip ftp password adminpassword	(任意) リモート パスワードを指定します。
ステップ 5	end 例： Device(config)# end	(任意) コンフィギュレーション モードを終了します。このステップが必要になるのは、デフォルトのリモート ユーザー名を上書きする場合のみです (ステップ 3 および 4 を参照)。
ステップ 6	copy ftp: [[//location]/directory]/bundle_name flash: 例： Device>copy ftp:/cat9k_iosxe.16.11.01.SPA.bin flash:	FTP を使用してネットワーク サーバーからフラッシュ 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 サーバー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 サーバーからフラッシュメモリ デバイスへコンフィギュレーションファイルをコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	（任意）グローバル コンフィギュレーション モードを開始します。この手順は、デフォルトのリモート ユーザー名またはパスワードを上書きする場合にのみ必要です（ステップ 3 を参照）。
ステップ 3	ip rcmd remote-username <i>username</i> 例： Device(config)# ip rcmd remote-username Admin01	（任意）リモート ユーザー名を指定します。
ステップ 4	end 例： Device(config)# end	（任意）コンフィギュレーション モードを終了します。この手順は、デフォルトのリモートユーザー名またはパスワードを上書きする場合にのみ必要です（ステップ 3 を参照）。
ステップ 5	copy rcp: [[[//]<i>username@</i>]<i>location</i>]<i>/directory</i> <i>/bundle_name</i> flash: 例： Device# copy rcp://netadmin@172.16.101.101/bundle1 flash:	RCP を使用してネットワーク サーバー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 サーバー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 サーバー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のコピーするには、以下の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy tftp: [[[/location]/directory]/bundle_name flash: 例： Device# copy tftp://192.168.1.100/switch-config flash:	TFTP サーバー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 サーバーから `usbflash0` に挿入されているフラッシュメモリ カードへ、`switch-config` という名前のコンフィギュレーションファイルをコピーする例を示します。コピーされたファイルの名前は `new-config` に変更されます。

```
Device#
copy tftp:switch-config usbflash0:new-config
```

スタートアップコンフィギュレーションファイルでのコンフィギュレーションコマンドの再実行

スタートアップコンフィギュレーションファイルのコマンドを再実行するには、このセクションの手順を実行します。

手順

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

	コマンドまたはアクション	目的
	Device> enable	<ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure memory 例： Device# configure memory	スタートアップ コンフィギュレーション ファイルでコンフィギュレーション コマンドを再実行します。

スタートアップ コンフィギュレーションのクリア

スタートアップ コンフィギュレーションから設定情報を消去できます。デバイスをスタートアップ コンフィギュレーションなしで再起動した場合は、デバイスを最初から設定できるように、デバイスは、**Setup** コマンドファシリティに移行します。スタートアップ コンフィギュレーションの内容をクリアするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	erase nvram 例：	スタートアップ コンフィギュレーションの内容をクリアします。

	コマンドまたはアクション	目的
	Device# erase nvram	<p>(注)</p> <p>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プラットフォームフォーム上では、erase startup-configEXEC コマンドを使用すると、CONFIG_FILE 環境変数により指定されたコンフィギュレーションが、デバイスにより削除されます。この変数が NVRAM を指定している場合は、デバイスにより NVRAM が消去されます。CONFIG_FILE 環境変数がフラッシュメモリデバイスとコンフィギュレーションファイル名を指定している場合は、デバイスによりコンフィギュレーションファイルが削除されます。つまり、そのコンフィギュレーションファイルはデバイスにより消去されるのではなく、「削除済み」としてマークされます。この機能では、削除されたファイルを回復できます。</p>

指定されたコンフィギュレーションファイルの削除

特定のフラッシュデバイスの指定された設定を削除するには、このセクションの手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>delete flash-filesystem:filename</p> <p>例 :</p> <pre>Device# delete usbflash0:myconfig</pre>	<p>特定のフラッシュ デバイス上の指定されたコンフィギュレーション ファイルを削除します。</p> <p>(注) クラス A および B フラッシュファイルシステムでは、フラッシュメモリ内の特定のファイルを削除すると、そのファイルは削除済みとしてシステムによりマークされます。これにより、undelete EXEC コマンドを使用して、削除したファイルを後で回復できるようになります。消去されたファイルは回復できません。コンフィギュレーション ファイルを完全に消去するには、squeeze EXEC コマンドを使用します。クラス C フラッシュファイルシステムでは、削除されたファイルは回復できません。CONFIG_FILE 環境変数で指定されたコンフィギュレーション ファイルを消去または削除しようとした場合、システムにより削除の確認を求めるプロンプトが表示されます。</p>

クラス A フラッシュ ファイル システムでの CONFIG_FILE 環境変数の指定

クラス A フラッシュ ファイル システムでは、CONFIG_FILE 環境変数で指定されたスタートアップコンフィギュレーションファイルをロードするように 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 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	copy [flash-url ftp-url rcp-url tftp-url system:running-config nvram:startup-config] dest-flash-url 例： Device# copy system:running-config nvram:startup-config	フラッシュファイルシステムにコンフィギュレーションファイルをコピーします。再起動時には、ここからデバイスにファイルがロードされます。
ステップ 3	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 4	boot config dest-flash-url 例： Device(config)# boot config 172.16.1.1	CONFIG_FILE 環境変数を設定します。この手順により、実行時の CONFIG_FILE 環境変数が変更されます。
ステップ 5	end 例： Device(config)# end	グローバル コンフィギュレーションモードを終了します。
ステップ 6	copy system:running-config nvram:startup-config 例： Device# copy system:running-config nvram:startup-config	スタートアップ コンフィギュレーションにステップ 3 で実行されたコンフィギュレーションを保存します。
ステップ 7	show boot 例： Device# show boot	(任意) CONFIG_FILE 環境変数の内容を確認できます。

例

次の例は、実行コンフィギュレーション ファイルをデバイスにコピーします。その後、システムが再起動されるとこのコンフィギュレーションがスタートアップ コンフィギュレーションとして使用されます。

```
Device# copy system:running-config usbflash0:config2
Device# configure terminal
Device(config)# boot config usbflash0:config2
Device(config)# end
Device# copy system:running-config nvram:startup-config
[ok]
Device# show boot
BOOT variable = usbflash0:rsp-boot-m
CONFIG_FILE variable = nvram:
Current CONFIG_FILE variable = usbflash0:config2
Configuration register is 0x010F
```

次の作業

スタートアップ コンフィギュレーション ファイルの場所を指定すると、**nvram:startup-config** コマンドは、スタートアップ コンフィギュレーション ファイルの新しい場所のエイリアスとなります。**more nvram:startup-config EXEC** コマンドにより、スタートアップ コンフィギュレーションの場所に関係なく、スタートアップ コンフィギュレーションが表示されます。**erase nvram:startup-config EXEC** コマンドにより、NVRAM の内容が消去され、CONFIG_FILE 環境変数で指定されたファイルが削除されます。

copy system:running-config nvram:startup-config コマンドを使用してコンフィギュレーションを保存した場合、デバイスによりコンフィギュレーション ファイルの完全バージョンは CONFIG_FILE 環境変数により指定された場所に保存され、抽出バージョンは NVRAM に保存されます。抽出バージョンとは、アクセスリスト情報を含まないバージョンです。NVRAM に完全バージョンのコンフィギュレーション ファイルが含まれている場合は、デバイスは完全バージョンを抽出バージョンで上書きすることを確認するプロンプトを表示します。NVRAM に抽出コンフィギュレーションが含まれている場合は、デバイスは確認のプロンプトを表示しないで NVRAM にある既存の抽出バージョンのコンフィギュレーション ファイルを上書きする処理を進めます。



- (注) フラッシュデバイスにあるファイルを CONFIG_FILE 環境変数として指定した場合、**copy system:running-config nvram:startup-config** コマンドでコンフィギュレーション ファイルを保存するたびに、古いコンフィギュレーション ファイルは「削除済み」とマークされ、新しいコンフィギュレーション ファイルがそのデバイスに保存されます。それでも古いコンフィギュレーション ファイルがメモリを使用するため、最終的にフラッシュメモリは一杯になります。**squeeze EXEC** コマンドを使用して古いコンフィギュレーション ファイルを完全に削除し、領域を解放してください。

コンフィギュレーションファイルをダウンロードするデバイスの設定

ネットワーク コンフィギュレーションおよびホスト コンフィギュレーション ファイル名の順序付きリストを指定できます。Cisco IOS XE ソフトウェアは、適切なネットワークまたはホスト コンフィギュレーション ファイルをロードするまで、このリストをスキャンします。

システムの起動時にコンフィギュレーションファイルをダウンロードするようにデバイスを設定するには、次のセクションで説明するタスクを少なくとも 1 つ実行します。

- [ネットワーク コンフィギュレーションファイルをダウンロードするデバイスの設定](#)
- [ホスト コンフィギュレーションファイルをダウンロードするデバイスの設定](#)

起動中にコンフィギュレーションファイルをロードできなかった場合、要求されたファイルがホストから提供されるまで、デバイスは 10 分ごと（デフォルト設定）に再試行します。試行が失敗するごとに、デバイスにより以下のメッセージがコンソール端末に表示されます。

```
Booting host-config... [timed out]
```

スタートアップ コンフィギュレーション ファイルになんらかの問題がある場合、またはコンフィギュレーション レジスタが NVRAM を無視するように設定されている場合は、デバイスは Setup コマンドファシリティに移行します。

ネットワーク コンフィギュレーション ファイルをダウンロードするデバイスの設定

起動時にサーバーからネットワーク コンフィギュレーション ファイルをダウンロードするように Cisco IOS ソフトウェアを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	boot network {ftp:[[/[username [:password]@]location]/directory]/filename] rcp:[[/[username@]location]/directory]/filename] tftp:[[/[location]/directory]/filename]} 例：	起動時にダウンロードするネットワーク コンフィギュレーション ファイルおよ び使用されるプロトコル（TFTP、RCP、 または FTP）を指定します。 • ネットワーク コンフィギュレーシ ョン ファイル名を指定しない場合、

	コマンドまたはアクション	目的
	<pre>Device(config)# boot network tftp:hostfile1</pre>	<p>Cisco IOS ソフトウェアはデフォルトのファイル名の network-config を使用します。アドレスを省略した場合、デバイスはブロードキャストアドレスを使用します。</p> <ul style="list-style-type: none"> 複数のネットワーク コンフィギュレーション ファイルを指定できません。ソフトウェアは、ネットワーク コンフィギュレーション ファイルをロードできるまで、入力された順に試行します。この手順は、異なる設定情報を持つ、ネットワーク サーバー上にロードされるファイルを複数保持する場合に役立ちます。
ステップ 4	<p>service config</p> <p>例 :</p> <pre>Device(config)# service config</pre>	再起動時にネットワーク ファイルを自動的にロードするようにシステムをイネーブルにします。
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	グローバル コンフィギュレーション モードを終了します。
ステップ 6	<p>copy system:running-config nvram:startup-config</p> <p>例 :</p> <pre>Device# copy system:running-config nvram:startup-config</pre>	実行コンフィギュレーションをスタートアップ コンフィギュレーション ファイルに保存します。

ホストコンフィギュレーションファイルをダウンロードするデバイスの設定

起動時にサーバーからホスト コンフィギュレーション ファイルをダウンロードするように Cisco IOS ソフトウェアを設定するには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>boot host {ftp:[[/[username [:password]@]location]/directory]/filename] rcp:[[/[username@]location]/directory]/filename] tftp:[[/location]/directory]/filename }</p> <p>例 :</p> <pre>Device(config)# boot host tftp:hostfile1</pre>	<p>起動時にダウンロードするホスト コンフィギュレーション ファイルおよび使用されるプロトコル (FTP、RCP、または TFTP) を指定します。</p> <ul style="list-style-type: none"> ホスト コンフィギュレーション ファイルの名前を指定しない場合、デバイスは、それ自身の名前を使用してホスト コンフィギュレーション ファイル名を形成します。このとき、その名前はすべて小文字に変換され、すべてのドメイン情報は削除され、「-config」が追加されます。ホスト名の情報を利用できない場合は、ソフトウェアはデフォルトのホスト コンフィギュレーション ファイル名の device-config を使用します。アドレスを省略した場合、デバイスはブロードキャストアドレスを使用します。 複数のホストコンフィギュレーション ファイルを指定できます。Cisco IOS ソフトウェアは、ホスト コンフィギュレーション ファイルをロードできるまで、入力された順に試行します。この手順は、異なる設定情報を持つ、ネットワーク サーバー上にロードされるファイルを複数保持する場合に役立ちます。
ステップ 4	<p>service config</p> <p>例 :</p> <pre>Device(config)# service config</pre>	<p>再起動時にホスト ファイルを自動的にロードするようにシステムをイネーブルにします。</p>
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device (config)# end</pre>	<p>グローバル コンフィギュレーション モードを終了します。</p>

	コマンドまたはアクション	目的
ステップ 6	copy system:running-config nvram:startup-config 例 : Device# copy system:running-config nvram:startup-config	実行コンフィギュレーションをスタートアップ コンフィギュレーション ファイルに保存します。

例

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

```
Device# configure terminal
Device(config)# boot host tftp:hostfile1
Device(config)# boot network tftp:networkfile1
Device(config)# service config
Device(config)# end
Device# copy system:running-config nvram:startup-config
```

コンフィギュレーション ファイルの管理の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	コンフィギュレーション ファイルの管理	コンフィギュレーション ファイルには、シスコ製デバイスの機能をカスタマイズするための Cisco IOS ソフトウェアコマンドが含まれています。コマンドは、システムを起動したとき (startup-config ファイルから)、またはコンフィギュレーション モードで CLI にコマンドを入力したときに、Cisco IOS ソフトウェアによって解析 (変換および実行) されます。

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



第 11 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）サーバー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](#)（375 ページ）
- [Secure Copy に関する情報](#)（375 ページ）
- [セキュアコピーの設定方法](#)（376 ページ）
- [セキュアコピーの設定例](#)（379 ページ）
- [セキュアコピーに関する追加情報](#)（380 ページ）
- [セキュアコピーの機能情報](#)（381 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クングアプリケーションセット）に基づいて設計されているため、その動作内容は Remote Copy Protocol（RCP）と類似しています。ただし、SCP は SSH のセキュリティに対応している点は除きます。加えて、SCP では、ユーザー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 のユーザーのみ

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ーの容量で動作する 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 サーバー側機能の設定をするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： <code>Device> enable</code>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： <code>Device# configure terminal</code>	グローバルコンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	ログイン時の AAA 認証を設定します。
ステップ 4	aaa authentication login {default list-name} method1 [method2...] 例： Device(config)# aaa authentication login default group tacacs+	AAA アクセスコントロールシステムをイネーブルにします。
ステップ 5	username name [privilege level] password encryption-type encrypted-password 例： 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。 (注) TACACS+ や RADIUS などのネットワークベースの認証メカニズムが設定されている場合は、この手順を省略できます。
ステップ 6	ip scp server enable 例： Device(config)# ip scp server enable	SCP サーバ側機能を有効にします。
ステップ 7	exit 例： Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 8	debug ip scp 例： Device# debug ip scp	(任意) SCP 認証問題を解決します。

SSH サーバーでのセキュアコピーのイネーブル化

次のタスクでは、SCP のサーバー側機能の設定方法を示します。このタスクは、デバイスでリモートのワークステーションからファイルを安全にコピーできる一般的な設定を示しています。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	認証、許可、アカウントिंग (AAA) アクセス コントロール モデルをイネーブルにします。
ステップ 4	aaa authentication login default local 例： Device(config)# aaa authentication login default local	ログイン時の認証にローカルのユーザー名データベースを使用するように AAA 認証を設定します。
ステップ 5	aaa authorization exec default local 例： Device(config)# aaa authorization exec default local	ユーザー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、ユーザー 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 例： Device(config)# username samplename privilege 15 password password1	ユーザ名ベースの認証システムを確立し、ユーザ名、権限レベル、および非暗号化パスワードを指定します。 (注) <i>privilege-level</i> 引数に必要な最小値は 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 例： Device(config)# ip ssh time-out 120	デバイスが SSH クライアントの応答を待つ時間間隔を、秒単位で設定します。

	コマンドまたはアクション	目的
ステップ 8	ip ssh authentication-retries 整数 例： Device(config)# ip ssh authentication-retries 3	インターフェイスのリセット後、認証を試行する回数を設定します。
ステップ 9	ip scp server enable 例： Device(config)# ip scp server enable	デバイスで、リモートワークステーションから安全にファイルをコピーできるようにします。
ステップ 10	ip ssh bulk-mode 例： Device(config)# ip ssh bulk-mode	(任意) SSH 一括データ転送モードをイネーブルにして、SCP のスループットパフォーマンスを強化します。
ステップ 11	exit 例： Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 12	debug ip scp 例： Device# debug ip scp	(任意) SCP 認証の問題に関する診断情報を提供します。

セキュアコピーの設定例

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ー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ー側の設定

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ー側機能の設定方法を示しています。

```
! AAA authentication and authorization must be configured properly for SCP to work.
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login default group tacacs+
Device(config)# aaa authorization exec default group tacacs+
! SSH must be configured and functioning properly.
Device(config)# ip ssh time-out 120
Device(config)# ip ssh authentication-retries 3
Device(config)# ip scp server enable
Device(config)# end
```

セキュアコピーに関する追加情報

関連資料

関連項目	マニュアルタイトル
セキュア シェル バージョン 1 と 2 のサポート	セキュア シェルの設定

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

セキュアコピーの機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	セキュアコピー	Secure Copy 機能は、デバイス設定またはデバイスイメージファイルをコピーするための安全で認証された方式を提供します。SCP は、SSH、アプリケーション、および Berkeley r ツールのセキュアな代替手段を提供するプロトコルに依存します。 次のコマンドが導入または変更されました。 debug ip scp および ip scp server enable
Cisco IOS XE Amsterdam 17.2.1	セキュアコピーのパフォーマンス向上	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セス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。

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 12 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 \(383 ページ\)](#)
- [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 \(384 ページ\)](#)
- [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 \(384 ページ\)](#)
- [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 \(387 ページ\)](#)
- [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 \(395 ページ\)](#)
- [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 \(398 ページ\)](#)
- [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 \(398 ページ\)](#)

コンフィギュレーションの置換とロールバックの前提条件

コンフィギュレーションの置換とロールバックの機能に対する入力となるコンフィギュレーションファイルの形式は、標準の Cisco ソフトウェア コンフィギュレーションファイルの、次に示すインデント規則に準拠している必要があります。

- 新しい行のすべてのコマンドは、コマンドがコンフィギュレーションサブモードにない限り、インデントなしで開始します。
- レベル1 コンフィギュレーションサブモード内のコマンドは、スペース1個分インデントします。
- レベル2 コンフィギュレーションサブモード内のコマンドは、スペース2個分インデントします。
- 以下、続くサブモード内のコマンドは、同じようにインデントします。

これらのインデント規則には、ソフトウェアが **show running-config** や **copy running-config destination-url** などのコマンドのコンフィギュレーションファイルを作成する方法が記述され

ています。シスコ デバイスで生成されるコンフィギュレーション ファイルは、いずれもこうした規則に従います。

2つのコンフィギュレーションファイル（現在の実行コンフィギュレーションと、保存された置換用コンフィギュレーション）を合わせたサイズより大きな空きメモリが必要です。

コンフィギュレーションの置換とロールバックの制約事項

デバイスに、2つのコンフィギュレーションファイル（現在の実行コンフィギュレーションと、保存された置換用コンフィギュレーション）を合わせたサイズより大きな空きメモリがない場合、コンフィギュレーション置換操作は実行されません。

ネットワークデバイスの物理コンポーネント（物理インターフェイスなど）に関連する特定の Cisco コンフィギュレーション コマンドは、実行コンフィギュレーションについて追加または削除することはできません。たとえば、コンフィギュレーション置換操作を行っても、そのインターフェイスがデバイス上に物理的に存在する場合、現在の実行コンフィギュレーションから **interface ethernet 0** コマンド行を削除することはできません。同様に、**interface ethernet 1** コマンド行は、そのようなインターフェイスがデバイス上に物理的に存在しない場合、実行コンフィギュレーションに追加することはできません。コンフィギュレーション置換操作でこのタイプの変更を試行すると、その特定のコマンド行が失敗したことを示すエラーメッセージが表示されます。

非常にまれなケースですが、ルータをリロードしないと特定の Cisco コンフィギュレーション コマンドを実行コンフィギュレーションから削除できないことがあります。コンフィギュレーション置換操作でこのタイプのコマンドの削除を試行すると、その特定のコマンド行が失敗したことを示すエラーメッセージが表示されます。

コンフィギュレーションの置換とロールバックについて

コンフィギュレーション アーカイブ

Cisco IOS コンフィギュレーション アーカイブは、**configure replace** コマンドにより提供されるコンフィギュレーションのロールバック機能を強化するために、Cisco IOS コンフィギュレーションファイルのアーカイブの保存、整理、管理を行うことを目的としたメカニズムです。この機能の導入前にも、実行コンフィギュレーションのコピーを **copy running-config destination-url** コマンドを使用して保存し、ローカルやリモートに置換ファイルを保管できました。ただし、この方法ではファイルの自動管理を行うことはできませんでした。一方、コンフィギュレーションの置換とロールバック機能では、実行コンフィギュレーションファイルを自動的に Cisco IOS コンフィギュレーション アーカイブに保存できます。アーカイブされたファイルはコンフィギュレーションのチェックポイントとして参照することができ、**configure replace** コマンドを使用して以前のコンフィギュレーション状態に戻すために利用できます。

archive config コマンドを使用すると、Cisco IOS コンフィギュレーションをコンフィギュレーションアーカイブに保存できます。その場合、標準のディレクトリとファイル名のプレフィクスが使用され、バージョン番号（およびオプションでタイムスタンプ）が自動的に付加されます。バージョン番号は連続したファイルを保存するごとに、1 つずつ大きくなります。この機能により、保存した Cisco IOS コンフィギュレーション ファイルを一貫して識別できます。アーカイブに保存する実行コンフィギュレーションの数は指定することができます。アーカイブ内のファイル数が上限値に達すると、次に最新のファイルが保存されるときに、最も古いファイルが自動的に消去されます。**show archive** コマンドを使用すると、Cisco IOS コンフィギュレーションアーカイブに保存されているすべてのコンフィギュレーションファイルに関する情報が表示されます。

コンフィギュレーション ファイルを保存する Cisco IOS コンフィギュレーションアーカイブは、**configure replace** コマンドで使用することによって、FTP、HTTP、RCP、TFTP のファイルシステム上に配置できます。

コンフィギュレーションの置換

configure replace 特権 EXEC コマンドにより、現在の実行コンフィギュレーションを、保存しておいた Cisco IOS コンフィギュレーション ファイルで置換することができます。この機能は、コンフィギュレーションを保存しておいた状態へ戻すために使用することができ、そのコンフィギュレーション状態が保存された後にどのような変更が加えられても、効果的にロールバックさせることができます。

configure replace コマンドを使用するときは、現在の実行コンフィギュレーションと置換するための、保存された Cisco IOS コンフィギュレーション ファイルを指定する必要があります。置換ファイルは、Cisco IOS デバイスによって作成された完全なコンフィギュレーション (**copy running-config destination-url** コマンドによって作成されたものなど) であることが必要です。あるいは、置換ファイルを外部的に作成する場合は Cisco IOS デバイスが作成するファイル形式に完全に準拠していなければなりません。**configure replace** コマンドを入力すると、現在の実行コンフィギュレーションが指定された置換コンフィギュレーションと比較され、一連の diff が生成されます。2 つのファイルの比較に使用されるアルゴリズムは、**show archive config differences** コマンドで使用されるものと同じです。置換コンフィギュレーションの状態になるよう、diff の結果が Cisco IOS パーサーによって適用されます。diff のみが適用されるため、現在の実行コンフィギュレーション上にすでに存在していた設定コマンドを再適用することにより生じる、潜在的なサービスの中断を避けられます。このアルゴリズムでは、順序に依存するコマンド（アクセス リストなど）へのコンフィギュレーション変更を、複数のパス プロセスを通して効果的に実行します。通常的环境では、コンフィギュレーション置換操作の完了に必要なパスは 3 つまでであり、ループ動作を防ぐためのパスは最大 5 つまでに制限されます。

Cisco IOS **copy source-url running-config** 特権 EXEC コマンドは、保存された Cisco IOS コンフィギュレーション ファイルを実行コンフィギュレーションへコピーするためによく使用されます。**copy source-url running-config** コマンドを **configure replace target-url** 特権 EXEC コマンドの代わりに使用する場合、主な相違点として次の点に注意が必要です。

- **copy source-url running-config** コマンドはマージ動作であり、ソースファイルと現在の実行コンフィギュレーションの両方のコマンドがすべて保持されます。このコマンドでは、現在の実行コンフィギュレーションにのみ含まれ、ソースファイルには存在しないコマン

ドが削除されることはありません。これに対して、**configure replace target-url** コマンドでは、置換ファイルに存在しないコマンドが現在の実行コンフィギュレーションから削除され、追加する必要があるコマンドが現在の実行コンフィギュレーションに追加されます。

- **copy source-url running-config** コマンドでは、現在の実行コンフィギュレーションにすでに存在しているかどうかにかかわらず、ソースファイル中のすべてのコマンドが適用されます。このアルゴリズムは効率的でない上、場合によってはサービスの停止が発生します。これに対して、**configure replace target-url** コマンドでは適用が必要なコマンドのみを適用し、現在の実行コンフィギュレーションに存在しているコマンドは再適用されません。
- **copy source-url running-config** コマンドでは部分的なコンフィギュレーションファイルもコピー元として使用できますが、**configure replace target-url** コマンドの置換ファイルとして使用できるのは、完全な Cisco IOS コンフィギュレーションファイルのみです。

コンフィギュレーション置換操作にロック機能が導入されました。**configure replace** コマンドが使用されると、コンフィギュレーション置換の動作中、デフォルトで実行コンフィギュレーションファイルがロックされます。このロックメカニズムによって、置換動作の実行中に他のユーザーが実行コンフィギュレーションを変更しようとしたために、置換動作の不正終了が発生することを防止できます。**no lock** キーワードを **configure replace** コマンドの実行時に使用すると、実行コンフィギュレーションのロックをディセーブルにできます。

実行コンフィギュレーションのロックは、コンフィギュレーションの置換動作終了時に自動的にクリアされます。**show configuration lock** コマンドを使用すると、現在実行コンフィギュレーションに適用されているロックをすべて表示できます。

コンフィギュレーション ロールバック

ロールバックの概念は、データベースの操作ではトランザクションプロセスモデルに由来します。データベーストランザクションでは、あるデータベースのテーブルに一連の変更を加えることがあります。その後、変更を実行する（変更を恒久的に適用する）か、変更をロールバックする（変更を破棄してテーブルを以前の状態に戻す）かを選択することになります。ここでロールバックが意味するのは、変更のログを含んだジャーナルファイルが破棄され、何の変更も加えられないということです。ロールバック操作の結果として、加えた変更が適用される前の状態に戻ります。

configure replace コマンドを使用することで、以前のコンフィギュレーション状態へ戻ることが可能になり、コンフィギュレーション状態の保存後に加えた変更を効率的にロールバックさせることができます。Cisco IOS コンフィギュレーション ロールバックは、適用された一連の変更をもとにロールバック動作を行うのではなく、保存された Cisco コンフィギュレーションファイルに基づいた特定のコンフィギュレーション状態へ戻るといったコンセプトを採用しています。このコンセプトは、チェックポイント（データベースの保存されたバージョン）に特定の状態を保存しておくという、データベースの考え方に類似しています。

コンフィギュレーションのロールバック機能が必要な場合、コンフィギュレーションの変更には先立って Cisco IOS 実行コンフィギュレーションを保存する必要があります。次に、コンフィギュレーションを変更した後に (**configure replace target-url** コマンドを使用し) 保存したコンフィギュレーションファイルを使って変更をロールバックします。保存された Cisco IOS コン

フィギュレーションファイルならどれでも置換コンフィギュレーションとして指定できるため、一部のロールバックモデルのように、ロールバックの数が制限されることもありません。

コンフィギュレーション ロールバック変更確認

コンフィギュレーションロールバック変更確認機能により、コンフィギュレーション変更の実行に際して確認を要求するようオプションで設定できます。この確認が受信できない場合、コンフィギュレーションは変更が適用される前の状態に戻されます。このメカニズムは、ネットワークデバイスとユーザーまたは管理アプリケーションとの接続において、コンフィギュレーション変更に起因する切断を防止するものです。

コンフィギュレーションの置換とロールバックの利点

- コンフィギュレーションの変更を効率的にロールバックさせて、以前のコンフィギュレーション状態へ戻ることが可能。
- デバイスをリロードしたり、CLIで実行コンフィギュレーションファイルに加えた変更を手動で元に戻したりすることなく、現在の実行コンフィギュレーションファイルをスタートアップコンフィギュレーションファイルと置換できるため、システムのダウンタイムが減少。
- 保存しておいたどの Cisco IOS コンフィギュレーション状態に戻すことも可能。
- 追加や削除が必要なコマンドだけが影響される場合、デバイスに完全なコンフィギュレーションファイルを適用することができるため、コンフィギュレーションの変更がシンプルに。
- **configure replace** コマンドを **copy source-url running-config** コマンドの代用として使用すると、現在の実行コンフィギュレーションにある既存のコマンドが再度適用されないため、効率が向上し、サービス停止のリスクが回避されます。

コンフィギュレーションの置換とロールバックの使用方法

コンフィギュレーションアーカイブの作成

configure replace コマンドを使用するうえで前提条件となる設定はありません。**configure replace** コマンドと、Cisco IOS コンフィギュレーションアーカイブおよび **archive config** コマンドとの併用は任意ですが、コンフィギュレーションロールバックのシナリオでは大きな利点があります。**archive config** コマンドを使用する前に、コンフィギュレーションアーカイブを設定しておく必要があります。コンフィギュレーションアーカイブの特性を設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>archive</p> <p>例 :</p> <pre>Device(config)# archive</pre>	<p>アーカイブ コンフィギュレーション モードを開始します。</p>
ステップ 4	<p>path url</p> <p>例 :</p> <pre>Device(config-archive)# path flash:myconfiguration</pre>	<p>Cisco IOS コンフィギュレーションアーカイブの場所と、ファイル名のプレフィックスを指定します。</p> <p>(注) パスのところでファイルの代わりにディレクトリを指定する場合、ディレクトリ名は <code>path flash:/directory/</code> のように後ろにスラッシュを付ける必要があります。このスラッシュはファイル名の後ろでは必要ありません。ディレクトリを指定する場合にだけ使います。</p>
ステップ 5	<p>maximum number</p> <p>例 :</p> <pre>Device(config-archive)# maximum 14</pre>	<p>(任意) Cisco IOS コンフィギュレーションアーカイブに保存される実行コンフィギュレーションのアーカイブ ファイル数の上限値を設定します。</p> <ul style="list-style-type: none"> <code>number</code> 引数は、Cisco IOS コンフィギュレーションアーカイブに保存される実行コンフィギュレーションのアーカイブ ファイル数の上限値を示します。有効な値は 1 ~ 14 で、デフォルトは 10 です。

	コマンドまたはアクション	目的
		<p>(注) このコマンドを使用する前に、path コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。</p>
ステップ 6	<p>time-period <i>minutes</i></p> <p>例 :</p> <pre>Device(config-archive)# time-period 1440</pre>	<p>(任意) CiscoIOS コンフィギュレーションアーカイブに実行コンフィギュレーションのアーカイブファイルを自動保存する間隔を設定します。</p> <ul style="list-style-type: none"> • Cisco IOS コンフィギュレーションアーカイブに現在の実行コンフィギュレーションのアーカイブファイルをどれほどの頻度で自動保存するかを、<i>minutes</i> 引数により分単位で指定します。 <p>(注) このコマンドを使用する前に、path コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。</p>
ステップ 7	<p>end</p> <p>例 :</p> <pre>Device(config-archive)# end</pre>	<p>特権 EXEC モードに戻ります。</p>
ステップ 8	<p>archive config</p> <p>例 :</p> <pre>Device# archive config</pre>	<p>現在の実行設定ファイルを設定アーカイブに保存します。</p> <p>(注) このコマンドを使用する前に、path コマンドを設定する必要があります。</p>

コンフィギュレーションの置換やロールバック操作の実行

保存された Cisco IOS コンフィギュレーションファイルで現在の実行コンフィギュレーションファイルを置換するには、次の作業を実行します。



(注) この手順の前に、コンフィギュレーションアーカイブを作成しておく必要があります。詳細については、[コンフィギュレーションアーカイブの作成](#)を参照してください。次に、現在の実行コンフィギュレーションで問題が生じた場合に、アーカイブしておいたコンフィギュレーションに戻す手順の詳細を示します。

手順

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	<p>configure replace target-url [nolock] [list] [force] [ignore case] [revert trigger [error]] [timer minutes] time minutes]</p> <p>例 :</p> <pre>Device# configure replace flash: startup-config time 120</pre>	<p>保存しておいた Cisco IOS コンフィギュレーション ファイルで現在の実行コンフィギュレーション ファイルを置換します。</p> <ul style="list-style-type: none"> • target-url 引数は、archive config コマンドで作成されたコンフィギュレーション ファイルなど、現在の実行コンフィギュレーションと置換する、保存された Cisco IOS コンフィギュレーション ファイルの URL です (Cisco IOS ファイルシステムでアクセス可能なもの)。 • list キーワードは、コンフィギュレーション置換動作のパスごとに、Cisco IOS ソフトウェア パーサーによって適用されるコマンドラインのリストを表示します。実行されたパスの総数も表示されます。 • force キーワードは、現在の実行コンフィギュレーションから指定した Cisco IOS コンフィギュレーション ファイルへの置換を、確認プロンプトを出さずに実行します。 • time minutes キーワードおよび引数は、現在の実行コンフィギュレーション ファイルの置換確認のために configure confirm コマンドを入

	コマンドまたはアクション	目的
		<p>力しなければならない制限時間（分単位）を指定します。configure confirm コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作は自動的に戻されます（つまり、現在の実行コンフィギュレーションファイルがconfigure replace コマンド入力以前のコンフィギュレーション状態へと回復されます）。</p> <ul style="list-style-type: none"> • nolock キーワードは、コンフィギュレーション置換操作中に他のユーザーが実行コンフィギュレーションを変更しないように実行コンフィギュレーションファイルをロックする機能をオフにします。 • revert trigger キーワードは、元のコンフィギュレーションへ戻すトリガーを次の内容から設定します。 <ul style="list-style-type: none"> • error : エラー時に元のコンフィギュレーションに戻します。 • timer minutes : 指定した時間が過ぎると元のコンフィギュレーションに戻します。 <p>(注) 場合によっては、複数パス操作の revert trigger 操作を実行しているときに、コンフィギュレーションの一部が失われ、元のコンフィギュレーション状態に戻す操作が失敗することがあります。</p> <ul style="list-style-type: none"> • 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。

	コマンドまたはアクション	目的
ステップ 3	<p>configure revert { now timer { <i>minutes</i> idle <i>minutes</i> } }</p> <p>例 :</p> <pre>Device# configure revert now</pre>	<p>(任意) 時間指定ロールバックをキャンセルしてロールバックを即時トリガーする、または時間指定ロールバックのパラメータをリセットするには、特権EXECモードで configure revert コマンドを使用します。</p> <ul style="list-style-type: none"> • now : ロールバックをただちにトリガーします。 • 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。 <ul style="list-style-type: none"> • 元に戻す時間を分単位で新たに指定するには、<i>minutes</i> 引数を使用し、timer キーワードとともに使用します。 • 保存されたコンフィギュレーションに戻すまでに、操作が行われないアイドル時間を最大どれほど長く許容できるかを設定するには、分単位の時間とともに idle キーワードを使用します。
ステップ 4	<p>configure confirm</p> <p>例 :</p> <pre>Device# configure confirm</pre>	<p>(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。</p> <p>(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のみ使用します。</p>
ステップ 5	<p>exit</p> <p>例 :</p> <pre>Device# exit</pre>	<p>ユーザー EXEC モードに戻ります。</p>

機能のモニター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ー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ー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、ユーザー 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ー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るとします。**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。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

コンフィギュレーションの置換およびコンフィギュレーションのロールバックの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	コンフィギュレーションの置換とロールバック	Cisco IOS コンフィギュレーションアーカイブは、 configure replace コマンドにより提供されるコンフィギュレーションのロールバック機能を強化するために、Cisco IOS コンフィギュレーション ファイルのアーカイブの保存、整理、管理を行うことを目的としたメカニズムです。

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



第 13 章

BIOS 保護

- [BIOS 保護の概要 \(399 ページ\)](#)
- [ROMMON アップグレード \(399 ページ\)](#)
- [BIOS 保護の機能履歴 \(401 ページ\)](#)

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 モニター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、プライマリ FPGA、およびゴールデン FPGA（セキュアブート 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 ROMMON は、カプセルアップグレードを使用してのみアップグレードできます。

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、両方のスーパーバイザを同時にリロードしないでください。スタンバイスーパーバイザを ROMMON 状態にして、アクティブスーパーバイザを起動します。各スーパーバイザで 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。

スタンバイスーパーバイザを起動し、スタンバイスーパー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 保護の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.12.1	BIOS 保護	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。
Cisco IOS XE Amsterdam 17.1.1	カプセルアップグレード	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。

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 14 章

Extended Fast Software Upgrade の実行

- [Extended Fast Software Upgrade の前提条件](#) (403 ページ)
- [Extended Fast Software Upgrade の制約事項](#) (403 ページ)
- [Extended Fast Software Upgrade に関する情報](#) (404 ページ)
- [スタンドアロンスイッチでの Extended Fast Software Upgrade の実行方法](#) (405 ページ)
- [スタック構成スイッチでの Extended Fast Software Upgrade の実行方法](#) (414 ページ)
- [ソフトウェアのアップグレードまたはリロードの確認](#) (420 ページ)
- [その他の参考資料](#) (420 ページ)
- [Extended Fast Software Upgrade の機能履歴](#) (421 ページ)

Extended Fast Software Upgrade の前提条件

- Extended Fast Software Upgrade 機能は Cisco IOS XE Amsterdam 17.3.2a 以降にサポートされています。アップグレードプロセスを開始する前に、スイッチにインストールされているソフトウェアのバージョンが Cisco IOS XE Amsterdam 17.3.2a 以降であることを確認します。
- **no boot manual** コマンドを使用して手動ブートを無効にする必要があります。

Extended Fast Software Upgrade の制約事項

次の制約事項は、スタンドアロンスイッチとスタック構成スイッチの両方に適用されます。

- この機能は、スイッチがインストールモードで動作している場合のみサポートされません。
- この機能は、スパニングツリープロトコル (STP) のみが設定されたスイッチではサポートされません。さらに、スイッチには Rapid Spanning Tree Protocol (RSTP) または Multiple Spanning Tree Protocol (MSTP) を設定する必要があります。

- ルートデバイスでは、フォワーディングステートのデバイスポートのいずれかが STP ピア（STP が設定され、ルートデバイスに直接接続されているデバイス）に接続されている場合、Extended Fast Software Upgrade 機能はサポートされません。
- STP が設定され、ルートデバイスとして定義されていないデバイスの場合、Extended Fast Software Upgrade 機能は、STP ピアに接続されているフォワーディングステートのデバイスポートの数が 1 以下の場合にのみサポートされます。
- Extended Fast Software Upgrade の実行後、アプリケーション ホスティングが自動的に再起動しない場合があります。Cisco IOx の無効化と再有効化、アプリケーション ホスティングの設定、アプリケーションのインストール、アクティブ化、および再起動が必要になる場合があります。

スタック構成スイッチには、次の制約事項が適用されます。

- スタック構成スイッチが部分リング状態で設定されている場合、この機能はサポートされません。
- この機能は、Bidirectional Forwarding Detection (BFD) が設定されているスタック構成スイッチではサポートされません。
- この機能は、MACsec Key Agreement (MKA) が設定されているスタック構成スイッチではサポートされません。
- この機能は、Cisco TrustSec が設定されているスタック構成スイッチではサポートされません。
- 設定された単方向リンク検出 (UDLD) メッセージ間隔は、トラフィックのダウンタイム中は無視されます。間隔は、Extended Fast Software Upgrade の完了後に設定された値に復元されます。

Extended Fast Software Upgrade に関する情報

Extended Fast Software Upgrade により、ソフトウェアのリロードまたはアップグレード操作中のトラフィックのダウンタイムが削減されます。Fast Software Upgrade と比較して、スイッチの設定によっては、トラフィックのダウンタイムが 30 秒未満に短縮されます。Extended Fast Software Upgrade は、グレースフルリスタート機能（Cisco NSF とも呼ばれます）を使用して、ソフトウェアのアップグレードまたはリロード中に特定のルーティングプロトコルなどのスイッチ設定が影響を受けないようにします。



(注) Extended Fast Software Upgrade は、アクセスレイヤスイッチでのみ動作します。

Perpetual Power over Ethernet (PoE) が設定されている場合、ソフトウェアのリロードまたはアップグレード中に、接続されたデバイスに中断なく電力が供給されます。

Extended Fast Software Upgrade でサポートされるプロトコル

Extended Fast Software Upgrade 機能では、次のプロトコルがサポートされています。



(注) 次のプロトコル以外のすべてのプロトコルで、トラフィックのダウンタイムは 30 秒より長くなります。

- BGP (IPv4 および IPv6 アドレスファミリー)
- Flexible NetFlow
- IEEE 802.1X ポートベースの認証
- Intermediate System-to-Intermediate System (IS-IS)
- インターネット グループ管理プロトコル (IGMP) スヌーピング
- レイヤ 2 スイッチング
- リンク集約制御プロトコル (LACP)
- MAC 認証バイパス
- マルチキャストリスナー検出 (MLD) スヌーピング
- Open Shortest Path First (OSPF) または OSPFv2 または OSPFv3
- Per VLAN Spanning Tree (PVST)
- QoS
- スタティックポートチャネル (モードオン)
- RSTP または MSTP を使用した STP
- UDLD
- VPN ルーティングおよび転送 (VRF)
- Web 認証

スタンドアロンスイッチでの Extended Fast Software Upgrade の実行方法

次のセクションでは、スタンドアロンスイッチで Extended Fast Software Upgrade を実行する方法について説明します。

スタンドアロンスイッチでのソフトウェアのアップグレード

スタンドアロンスイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

- すべてのルーティングプロトコルが**UP**ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。
- 必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 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 image activate reloadfast commit 例： Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

IPv6 が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード

スタンドアロンスイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

- すべてのルーティングプロトコルが**UP**ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。
- 必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例：	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 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 nd reachable-time seconds 例： Device (config)# ipv6 nd reachable-time 3600000	到達可能性確認イベントの発生後、リモート IPv6 ノードが到達可能と判断されるまでの期限を設定します。
ステップ 4	end 例： Device (config)# end	特権 EXEC モードに戻ります。
ステップ 5	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 6	install add file image activate reloadfast commit 例： Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

IPv6 MLD が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード

IPv6 MLD が設定されたスタンドアロンスイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

- すべてのルーティングプロトコルが **UP** ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。
- 必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	ipv6 mld snooping 例： Device(config)# ipv6 mld snooping	IPv6 MLD スヌーピングをイネーブルにします。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 6	install add file image activate reloadfast commit 例： Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

BGP が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード

BGP が設定されたスタンドアロンスイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

- すべてのルーティングプロトコルが **UP** ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。
- 必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	router bgp autonomous-system-number 例： Device(config)# router bgp 65000	BGP ルータ コンフィギュレーションモードを開始します。
ステップ 4	bgp graceful-restart 例： Device(config-router)# bgp graceful-restart	NSF 認識をスイッチで有効にします。 NSF 認識はデフォルトでは無効です。
ステップ 5	end 例： Device(config-router)# end	特権 EXEC モードに戻ります。
ステップ 6	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 7	install add file image activate reloadfast commit 例： Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

OSPFv3が設定されたスタンドアロンスイッチでのソフトウェアのアップグレード

OSPFv3 が設定されたスタンドアロンスイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

- すべてのルーティングプロトコルが **UP** ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。
- 必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	ipv6 nd reachable-time seconds 例： Device(config)# ipv6 nd reachable-time 3600000	到達可能性確認イベントの発生後、リモート IPv6 ノードが到達可能と判断されるまでの期限を設定します。
ステップ 4	snmp ifmib ifindex persist 例： Device(config)# snmp ifmib ifindex persist	SNMP ifIndex の持続性をグローバルにイネーブルにします。
ステップ 5	router ospfv3 process-id 例： Device(config)# router ospfv3 1	OSPFv3 ルータ コンフィギュレーションモードを開始します。
ステップ 6	router-id ip-address 例： Device(config-router)# router-id 192.0.2.5	OSPFv3 インスタンスの固定ルータ ID を設定します。
ステップ 7	interface-id snmp-if-index 例： Device(config-router)# interface-id snmp-if-index	特定のインターフェイスで SNMP ifIndex の持続性をイネーブルにします。
ステップ 8	end 例： Device(config-router)# end	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 10	install add file image activate reloadfast commit 例： Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

スタンドアロンスイッチでのソフトウェアのリロード

スタンドアロンスイッチでソフトウェアをリロードするには、次の手順を実行します。

始める前に

すべてのルーティングプロトコルがUPステータスにあることを確認するには、特権EXECモードで **show graceful-reload** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 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	reload fast 例： Device# reload fast	ソフトウェアをアップグレードせずにスイッチをリロードします。

BGP が設定されたスタンドアロンスイッチでのソフトウェアのリロード

BGP が設定されたスタンドアロンスイッチでソフトウェアをリロードするには、次の手順を実行します。

始める前に

すべてのルーティングプロトコルがUPステータスにあることを確認するには、特権EXECモードで **show graceful-reload** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	router bgp autonomous-system-number 例： Device(config)# router bgp 65000	BGP ルータ コンフィギュレーションモードを開始します。
ステップ 4	bgp graceful-restart 例： Device(config-router)# bgp graceful-restart	NSF 認識をスイッチで有効にします。 NSF 認識はデフォルトでは無効です。
ステップ 5	end 例： Device(config-router)# end	特権 EXEC モードに戻ります。
ステップ 6	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 7	reload fast 例： Device# reload fast	ソフトウェアをアップグレードせずにスイッチをリロードします。

OSPFv3 が設定されたスタンドアロンスイッチでのソフトウェアのリロード

OSPFv3 が設定されたスタンドアロンスイッチでソフトウェアをリロードするには、次の手順を実行します。

始める前に

すべてのルーティングプロトコルがUPステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 nd reachable-time seconds 例 : Device (config)# ipv6 nd reachable-time 3600000	到達可能性確認イベントの発生後、リモート IPv6 ノードが到達可能と判断されるまでの期限を設定します。
ステップ 4	snmp ifmib ifindex persist 例 : Device (config)# snmp ifmib ifindex persist	SNMP ifIndex の持続性をグローバルにイネーブルにします。
ステップ 5	router ospfv3 process-id 例 : Device (config)# router ospfv3 1	OSPFv3 ルータ コンフィギュレーション モードを開始します。
ステップ 6	router-id ip-address 例 : Device (config-router)# router-id 192.0.2.5	OSPFv3 インスタンスの固定ルータ ID を設定します。
ステップ 7	interface-id snmp-if-index 例 : Device (config-router)# interface-id snmp-if-index	特定のインターフェイスで SNMP ifIndex の持続性をイネーブルにします。
ステップ 8	end 例 : Device (config-router)# end	特権 EXEC モードに戻ります。
ステップ 9	write memory 例 : Device# write memory	スイッチの設定を保存します。
ステップ 10	reload fast 例 :	ソフトウェアをアップグレードせずにスイッチをリロードします。

	コマンドまたはアクション	目的
	Device# reload fast	

スタック構成スイッチでの Extended Fast Software Upgrade の実行方法

ここでは、スタック構成のスイッチで Extended Fast Software Upgrade を実行する方法について説明します。

スタック構成スイッチでのソフトウェアのアップグレード

スタック構成スイッチでソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 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 imageactivate reloadfast commit 例： Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	次のプロセスが発生します。 <ol style="list-style-type: none">1. アクティブ、スタンバイ、およびメンバースイッチのイメージをアップグレードします。2. スタンバイスイッチとメンバースイッチを再起動します。アクティブスイッチを再起動し、スイッチの切り替えが行われます。スタンバイスイッチがアクティブスイッチになり、アクティブスイッチがスタンバイスイッチになります。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

BGP が設定されたスタック構成スイッチでのソフトウェアのアップグレード

BGP が設定されたスタック構成スイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

- すべてのルーティングプロトコルが **UP** ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。
- 必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp autonomous-system-number 例： Device(config)# router bgp 65000	BGP ルータ コンフィギュレーション モードを開始します。
ステップ 4	bgp graceful-restart all 例： Device(config-router)# bgp graceful-restart	NSF 認識をスイッチで有効にします。 NSF 認識はデフォルトでは無効です。
ステップ 5	end 例： Device(config-router)# end	特権 EXEC モードに戻ります。
ステップ 6	write memory 例： Device# write memory	スイッチの設定を保存します。

	コマンドまたはアクション	目的
ステップ 7	install add file imageactivate reloadfast commit 例 : <pre>Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit</pre>	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

IS-ISが設定されたスタック構成スイッチでのソフトウェアのアップグレード

IS-IS が設定されたスタック構成スイッチのソフトウェアをアップグレードするには、次の手順を実行します。

始める前に

必要に応じて、新しいソフトウェア用にディスク領域を解放するには、特権 EXEC モードで **install remove inactive** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : <pre>Device> enable</pre>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : <pre>Device# configure terminal</pre>	グローバル コンフィギュレーションモードを開始します。
ステップ 3	router isis area-tag 例 : <pre>Device(config)# router isis tag1</pre>	IS-IS ルーティングプロトコルをイネーブルにして IS-IS プロセスを指定し、スイッチをルータ コンフィギュレーションモードにします。
ステップ 4	nsf {cisco ietf} 例 : <pre>Device(config-router)# nsf cisco OR Device(config-router)# nsf ietf</pre>	IS-IS 用 NSF をイネーブルにします。 <ul style="list-style-type: none"> • ietf : IETF ドラフトベースの再起動をサポートするネットワークスイッチとの隣接関係がサポートしている同種ネットワークで IS-IS をイネーブルにする。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • cisco : NSF 認識ネットワークスイッチとの隣接関係がない同種ネットワークで IS-IS を実行する。
ステップ 5	end 例 : Device(config-router)# end	特権 EXEC モードに戻ります。
ステップ 6	write memory 例 : Device# write memory	スイッチの設定を保存します。
ステップ 7	install add file image activate reloadfast commit 例 : Device# install add file bootflash: cat9k_iosxe.17.03.02.SPA.bin activate reloadfast commit	新しいソフトウェアイメージでスイッチをアップグレードします。 <i>image</i> キーワードには、ファイルの場所 (TFTP、HTTP、フラッシュドライブ) とイメージ名を含めます。

スタック構成スイッチでのソフトウェアのリロード

スタック構成スイッチでソフトウェアをリロードするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	reload fast 例 : Device# reload fast	次のプロセスが発生します。 <ol style="list-style-type: none"> 1. スタンバイスイッチとメンバースイッチをリロードします。 2. アクティブスイッチをリロードし、スイッチの切り替えが行われます。スタンバイスイッチがアクティブスイッチになり、アクティブスイッチが新しいスタンバイスイッチになります。

BGP が設定されたスタック構成スイッチでのソフトウェアのリロード

BGP が設定されたスタック構成スイッチでソフトウェアをリロードするには、次の手順を実行します。

始める前に

すべてのルーティングプロトコルが **UP** ステータスにあることを確認するには、特権 EXEC モードで **show graceful-reload** コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp autonomous-system-number 例： Device(config)# router bgp 65000	BGP ルータ コンフィギュレーション モードを開始します。
ステップ 4	bgp graceful-restart all 例： Device(config-router)# bgp graceful-restart all	スタック内のすべてのスイッチで NSF 認識をイネーブルにします。NSF 認識はデフォルトでは無効です。
ステップ 5	end 例： Device(config-router)# end	特権 EXEC モードに戻ります。
ステップ 6	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 7	reload fast 例： Device# reload fast	ソフトウェアをアップグレードせずにスイッチをリロードします。

IS-ISが設定されたスタック構成スイッチでのソフトウェアのリロード

IS-IS が設定されたスタック構成スイッチでソフトウェアをリロードするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	router isis area-tag 例： Device(config)# router isis tag1	IS-IS ルーティングプロトコルをイネーブルにして IS-IS プロセスを指定し、スイッチをルータ コンフィギュレーションモードにします。
ステップ 4	nsf {cisco ietf} 例： Device(config-router)# nsf cisco OR Device(config-router)# nsf ietf	IS-IS 用 NSF をイネーブルにします。 <ul style="list-style-type: none"> • ietf : IETF ドラフトベースの再起動をサポートするネットワークスイッチとの隣接関係がサポートしている同種ネットワークで IS-IS をイネーブルにする。 • cisco : NSF 認識ネットワークスイッチとの隣接関係がない同種ネットワークで IS-IS を実行する。
ステップ 5	end 例： Device(config-router)# end	特権 EXEC モードに戻ります。
ステップ 6	write memory 例： Device# write memory	スイッチの設定を保存します。
ステップ 7	reload fast 例： Device# reload fast	ソフトウェアをアップグレードせずにスイッチをリロードします。

ソフトウェアのアップグレードまたはリロードの確認

ソフトウェアのアップグレードまたはリロードが成功したことを確認するには、特権 EXEC モードで次のコマンドを使用します。

表 22: ソフトウェアのアップグレードまたはリロードを確認するコマンド

コマンド	目的
show version	デバイスのハードウェアおよびソフトウェア情報を表示します。
show log in FAST	Extended Fast Software Upgrade を使用したソフトウェアアップグレードまたはリロードが完了したかどうかを表示します。
show install summary	アクティブなパッケージに関する情報を表示します。
show install log	インストール要求に関する情報を表示します。
show version running	現在実行中のファイルに関する情報を表示します。
show version in reason	最後のリロードの理由を表示します。

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
ルーティングプロトコルに関する情報	<i>Software Configuration Guide (Catalyst 9300 Switches)</i> の「 <i>IP Routing Configuration Guide</i> 」を参照してください。
STP、PVST、およびUDLDに関する情報	<i>Software Configuration Guide (Catalyst 9300 Switches)</i> の『レイヤ2設定ガイド』を参照してください。
無停止型 PoE に関する情報	<i>Software Configuration Guide (Catalyst 9300 Switches)</i> の「 <i>Network Powered Lighting Configuration Guide</i> 」を参照してください。
アプリケーションホスティングに関する情報	<i>Software Configuration Guide (Catalyst 9300 Switches)</i> の「 <i>Programmability Configuration Guide</i> 」を参照してください。

Extended Fast Software Upgrade の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Amsterdam 17.3.2a	Extended Fast Software Upgrade	Extended Fast Software Upgrade により、ソフトウェアのリロードまたはアップグレード操作中のトラフィックのダウンタイムが削減されます。 この機能は、Cisco Catalyst 9300 シリーズスイッチの 9300 および 9300L スイッチモデルでのみサポートされるようになりました。



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の制約事項 \(423 ページ\)](#)
- [ソフトウェアメンテナンスアップグレードについて \(423 ページ\)](#)
- [ソフトウェアメンテナンスの更新の管理方法 \(425 ページ\)](#)
- [ソフトウェアメンテナンスアップグレードの設定例 \(427 ページ\)](#)
- [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 \(432 ページ\)](#)
- [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 \(432 ページ\)](#)

ソフトウェアメンテナンスアップグレードの制約事項

- SMU は、インストールモードを使用したパッチのみをサポートします。

ソフトウェアメンテナンスアップグレードについて

SMU の概要

SMU は、システムにインストールして修正やセキュリティ解決をリリースされたイメージに提供できるパッケージです。SMU パッケージはリリースごとおよびコンポーネントごとに提供されます。

SMU はネットワークの問題に迅速に対応できるようにするとともに、必要なテストの時間と範囲を削減するため、従来の Cisco IOS ソフトウェアには多大なメリットがあります。Cisco IOS XE プラットフォームでは SMU の互換性を内部的に検証し、互換性のない SMU はインストールできません。

すべて SMU が後続の Cisco IOS XE ソフトウェアメンテナンスリリースに統合されています。SMU は独立した自己完結型パッケージであり、前提条件や依存関係はありません。SMU はどのような順序でもインストールまたはアンインストールできます。

SMU は拡張メンテナンスリリースでのみ、基盤となるソフトウェアリリースのライフサイクルにわたってサポートされます。

SMU をインストールするには、次の基本的な手順を実行します。

1. ファイルシステムに SMU を追加します。
2. システムで SMU をアクティブ化します。
3. リロードが繰り返されても持続させるための SMU の変更をコミットします。

SMU のワークフロー

SMU プロセスは、シスコカスタマーサポートへの要求によって開始されます。カスタマーサポートに連絡し、SMU 要求を行います。

SMU パッケージがリリースされると [Cisco Software Download]https://www.cisco.com/c/en_in/support/index.html ページに掲載されます。そのパッケージをダウンロードし、インストールします。

SMU パッケージ

SMU パッケージには、パッケージの内容を記述するいくつかのメタデータ、および SMU が要求されている報告済みの問題の修正とともに、リリースにパッチを適用するための一連のファイルがいくつか含まれています。SMU パッケージは、公開キーインフラストラクチャ (PKI) コンポーネントのパッチ適用もサポートします。

SMU のリロード

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、ユーザー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 例 : Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	install add file flash: filename [activate commit] 例 : Device# install add file flash:cat9k_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。 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。 (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 例 : Device# exit	特権 EXEC モードを終了し、ユーザー 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 例 : Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	install add file flash: filename 例 : 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 例 : 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 例 : 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} 例 : 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 例： 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} 例： 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 例： Device# show version	デバイスのイメージバージョンを表示します。
ステップ 9	show install summary 例： 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。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

ソフトウェアメンテナンスアップグレードの機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	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 (435 ページ)
- 使用可能なファイル システムの表示 (435 ページ)
- デフォルト ファイル システムの設定 (440 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 (441 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 (442 ページ)
- ディレクトリの作成 (443 ページ)
- ファイルのコピー (444 ページ)
- ファイルの作成、表示、および抽出 (445 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 (448 ページ)
- フラッシュファイルシステムの機能履歴 (448 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またはスタックメンバから見ると、`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。

一度に1人のユーザーのみが、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
Size (b) Free (b) Type Flags Prefixes
- - opaque rw system:
```

```

- - opaque rw tmpsys:
1651314688 1559785472 disk rw crashinfo:
* 11353194496 9693396992 disk rw flash:
8049967104 7959392256 disk ro webui:
- - opaque rw null:
- - opaque ro tar:
- - network rw tftp:
2097152 2080848 nvram rw nvram:
- - opaque wo syslog:
- - network rw rcp:
- - network rw http:
- - network rw ftp:
- - network rw scp:
- - network rw https:
- - opaque ro cns:

Device# show file systems
File Systems:
Size(b) Free(b) Type Flags Prefixes
- - opaque rw system:
- - opaque rw tmpsys:
1651314688 1467920384 disk rw crashinfo:
* 11353194496 6942072832 disk rw flash:
7723847680 7646384128 disk ro webui:
- - opaque rw null:
- - opaque ro tar:
- - network rw tftp:
2097152 2089932 nvram rw nvram:
- - network rw rcp:
- - network rw http:
- - network rw ftp:
- - network rw scp:
- - network rw https:
- - opaque ro cns:
118014062592 111933124608 disk rw usbflash1:

```

```

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:

```

この例では、usbflash1 filesystem 形式を表示します。

```

Device#show usbflash1: filesys
Filesystem: usbflash1
Filesystem Path: /vol/usb1
Filesystem Type: ext4
Mounted: Read/Write

```

次の例では、デバイススタックを示します。この例では、アクティブなデバイスはスタックメンバ1です。スタックメンバ2のファイルシステムはflash-2:として、スタックメンバ3のファイルシステムはflash-3:として表示されるといった具合に、まですきます。また、この例では、次のように、crashinfo ディレクトリと、アクティブなデバイスに接続されたUSB フラッシュドライブも示します。

```

Device# show file systems
File Systems:

Size(b) Free(b) Type Flags Prefixes
- - opaque rw system:
- - opaque rw tmpsys:
1651314688 1520742400 disk rw crashinfo: crashinfo-1:
1651507200 1516240896 disk rw crashinfo-2: stby-crashinfo:
1651507200 1517289472 disk rw crashinfo-3:
1651507200 1519386624 disk rw crashinfo-4:
1651507200 1524629504 disk rw crashinfo-5:
1651507200 1523580928 disk rw crashinfo-6:
1651507200 1517289472 disk rw crashinfo-7:
1651507200 1526726656 disk rw crashinfo-8:
* 11353194496 7916576768 disk rw flash: flash-1:
11353980928 7944011776 disk rw flash-2: stby-flash:
11353980928 7876902912 disk rw flash-3:
11353980928 7944011776 disk rw flash-4:
11353980928 7939817472 disk rw flash-5:
11353980928 7944011776 disk rw flash-6:
11353980928 7944011776 disk rw flash-7:
11353980928 7944011776 disk rw flash-8:
3824013312 3756507136 disk ro webui:
- - opaque rw null:
- - opaque ro tar:
- - network rw tftp:
2097152 2052489 nvram rw nvram:
- - opaque wo syslog:
- - network rw rcp:
- - network rw http:
- - network rw ftp:
- - network rw scp:
- - network rw https:
- - opaque ro cns:
2097152 2052489 nvram rw stby-nvram:
- - nvram rw stby-rcsf:
- - opaque rw revrcsf:

```

次の例では、デバイススタックを示します。この例では、アクティブなデバイスはスタックメンバ2です。スタックメンバ1のファイルシステムはflash-1:として、スタックメンバ2のファイルシステムはflash-2:として、スタックメンバ3のファイルシステムはflash-3:として表示さ

れるといった具合に、まで続きます。また、この例では、次のように、**crashinfo** ディレクトリと、アクティブなデバイスに接続された USB フラッシュドライブも示します。

```
Device# show file systems
File Systems:

      Size(b)      Free(b)      Type  Flags  Prefixes
      -          -          opaque  rw    system:
      -          -          opaque  rw    tmpsys:
      1651314688    1565089792    disk    rw    crashinfo: crashinfo-2:
      1651507200    1560281088    disk    rw    crashinfo-1:
      1651507200    1562378240    disk    rw    crashinfo-3: stby-crashinfo:
*  11353194496    10735611904    disk    rw    flash: flash-2:
  11353980928    10152312832    disk    rw    flash-1:
  11353980928    2161115136    disk    rw    flash-3: stby-flash:
  15243046912    14423638016    disk    rw    usbflash0: usbflash0-2:
      520093696    520093696    disk    rw    usbflash0-1:
      3497074688    3417554944    disk    ro    webui:
      -          -          opaque  rw    null:
      -          -          opaque  ro    tar:
      -          -          network  rw    tftp:
      2097152      2085334      nvram    rw    nvram:
      -          -          network  rw    rcp:
      -          -          network  rw    http:
      -          -          network  rw    ftp:
      -          -          network  rw    scp:
      -          -          network  rw    https:
      -          -          opaque  ro    cns:
      21003628544    19867037696    disk    rw    usbflash1: usbflash1-2:
      118014083072  111933390848    disk    rw    usbflash1-3: stby-usbflash1:
      2097152      2085334      nvram    rw    stby-nvram:
      -          -          nvram    rw    stby-rcsf:
      -          -          opaque  rw    revrcsf:
```

表 23: *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。

フィールド	値
Type	<p>ファイル システムのタイプです。</p> <p>disk : ファイルシステムは、フラッシュ メモリ デバイス、USB フラッシュ、crashinfo ファイル用です。</p> <p>network : ファイルシステムは、FTP サーバやHTTP サーバなどのネットワーク デバイス用です。</p> <p>nvram : ファイルシステムはNVRAM（不揮発性 RAM）デバイス用です。</p> <p>opaque : ファイルシステムは、ローカルに生成された pseudo ファイルシステム（system など）、またはダウンロード インターフェイス（brimux など）です。</p> <p>unknown : ファイル システムのタイプは不明です。</p>
Flags	<p>ファイル システムの権限です。</p> <p>ro : 読み取り専用です。</p> <p>rw : 読み取りおよび書き込みです。</p> <p>wo : 書き込み専用です。</p>

フィールド	値
Prefixes	<p>ファイル システムのエイリアスです。</p> <p>crashinfo : crashinfo ファイルです。</p> <p>flash : フラッシュ ファイル システムです。</p> <p>ftp : FTP サーバです。</p> <p>http : HTTP サーバです。</p> <p>https : セキュア HTTP サーバです。</p> <p>nvr : NVRAM です。</p> <p>null : コピーのヌル宛先です。リモート ファイルをヌルへコピーして、サイズを判別できます。</p> <p>rcp : Remote Copy Protocol (RCP) サーバです。</p> <p>scp : Session Control Protocol (SCP) サーバです。</p> <p>system : 実行コンフィギュレーションを含むシステム メモリが格納されています。</p> <p>tftp : TFTP ネットワーク サーバです。</p> <p>usbflash0 : USB フラッシュ メモリです。</p> <p>usbflash1 : 外部の USB フラッシュメモリです。</p> <p>ymodem : YMODEM プロトコルを使用して、ネットワーク マシンからファイルを取得します。</p>

デフォルト ファイル システムの設定

デフォルトのファイル システムとして使用されるファイル システムまたはディレクトリを指定するには、**cd filesystem:** 特権 EXEC コマンドを使用します。デフォルト ファイル システムを設定すると、関連するコマンドを実行するときに **filesystem:** 引数を省略できます。たとえば、オプションの **filesystem:** 引数を持つすべての特権 EXEC コマンドでは、**cd** コマンドで指定されたファイル システムが使用されます。

デフォルトでは、デフォルト ファイル システムは **flash:** です。

cd コマンドで指定された現在のデフォルトのファイル システムを表示するには、**pwd** 特権 EXEC コマンドを使用します。

ファイル システムのファイルに関する情報の表示

ファイルシステムの内容を操作する前に、そのリストを表示できます。たとえば、新しいコンフィギュレーション ファイルをフラッシュ メモリにコピーする前に、ファイル システムに同じ名前のコンフィギュレーション ファイルが格納されていないことを確認できます。同様に、フラッシュ コンフィギュレーション ファイルを別の場所にコピーする前に、ファイル名を確認して、その名前を別のコマンドで使用できます。ファイル システムのファイルに関する情報を表示するには、次の表に記載する特権 EXEC コマンドのいずれかを使用します。

表 24: 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:
Directory 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
```

ディレクトリの変更および作業ディレクトリの表示

```

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 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	dir filesystem: 例： Device# dir flash:	指定されたファイル システムのディレクトリを表示します。 <i>filesystem:</i> には、システム ボードのフラッシュ デバイスの <i>flash:</i> を使用します。 スタックのデバイスメンバのフラッシュパーティションにアクセスするには、 <i>flash-n</i> を使用します (<i>n</i> はスタックメンバ番号です)。例えば、 <i>flash-4</i> 。
ステップ 3	cd directory_name 例： Device# cd new_configs	指定されたディレクトリへ移動します。 コマンド例では、 <i>new_configs</i> という名前のディレクトリに移動する方法を示します。
ステップ 4	pwd 例： Device# pwd	作業ディレクトリを表示します。
ステップ 5	cd 例：	デフォルトディレクトリに移動します。

	コマンドまたはアクション	目的
	Device# cd	

ディレクトリの作成

特権 EXEC モードを開始して、ディレクトリを作成するには次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	dir filesystem: 例 : Device# dir flash:	指定されたファイル システムのディレクトリを表示します。 <i>filesystem:</i> には、システム ボードのフラッシュ デバイスの flash: を使用します。
ステップ 2	mkdir directory_name 例 : Device# mkdir new_configs	新しいディレクトリを作成します。スラッシュ (/) 間に指定できるディレクトリ名は最大 45 文字で、大文字と小文字の区別があります。ディレクトリ名には制御文字、スペース、スラッシュ、引用符、セミコロン、またはコロンは使用できません。
ステップ 3	dir filesystem: 例 : Device# dir flash:	入力を確認します。

ディレクトリの削除

ディレクトリを、その内部のすべてのファイルおよびサブディレクトリとともに削除するには、**delete /force /recursive filesystem:/file-url** 特権 EXEC コマンドを使用します。

名前で指定されたディレクトリを、その内部のすべてのサブディレクトリおよびファイルとともに削除するには、**/recursive** キーワードを使用します。ディレクトリ内のファイルごとに表示される、削除を確認するためのプロンプトを省略するには、**/force** キーワードを使用します。この削除プロセスを実行すると、最初に 1 度だけプロンプトが表示されます。

filesystem には、システム ボードのフラッシュ デバイスの **flash:** を使用します。*file-url* には、削除するディレクトリの名前を入力します。ディレクトリ内のすべてのファイルおよびディレクトリが削除されます。



注意 ディレクトリが削除された場合、その内容は回復できません。

ファイルのコピー

送信元から宛先にファイルをコピーするには、**copy source-url destination-url** 特権 EXEC コマンドを使用します。送信元および宛先の URL には、**running-config** および **startup-config** キーワードショートカットを使用できます。たとえば、**copy running-config startup-config** コマンドは、現在実行中のコンフィギュレーション ファイルをフラッシュメモリの NVRAM セクションに保存し、システム初期化の際にコンフィギュレーションファイルとして使用されるようにします。

XMODEM または YMODEM プロトコルを使用するネットワーク マシンのファイルに対する送信元として特殊なファイルシステム (**xmodem:**、**ymodem:**) を指定し、そこからコピーすることもできます。

ネットワーク ファイルシステムの URL には、ftp:、rcp:、tftp:、scp:、http:、https: などがあり、構文は次のとおりです。

- FTP : ftp:[[/username [:password]@location]/directory]/filename
- RCP : rcp:[[/username@location]/directory]/filename
- TFTP : tftp:[[/location]/directory]/filename
- SCP : scp:[[/username [:password]@location]/directory]/filename
- HTTP : http:[[/username [:password]@location]/directory]/filename
- HTTPS : https:[[/username [:password]@location]/directory]/filename



(注) パスワードに特殊文字「@」を含めることはできません。文字「@」を使用すると、コピーでサーバの IP アドレスを解析できません。

ローカルにある書き込み可能なファイル システムには **flash:** などがあります。

送信元および宛先の組み合わせによっては、無効な場合があります。特に、次に示す組み合わせの場合は、コピーできません。

- 実行コンフィギュレーションから実行コンフィギュレーションへ
- スタートアップ コンフィギュレーションからスタートアップ コンフィギュレーションへ
- デバイスから同じ名前のデバイスに (たとえば、**copy flash: flash:** コマンドは無効)

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時、削除の確認を求めると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: <i>/file-url</i> 例 : <pre>Device# archive tar /create tftp:172.20.10.30/saved. flash:/new-configs</pre>	ファイルを作成し、そこにファイルを追加します。 <i>destination-url</i> 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。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> ローカルフラッシュ ファイル システム構文 flash: FTP 構文 <code>ftp://[username][password]@[location]/directory/-filename.</code> RCP 構文 <code>rcp://[username@location]/directory/-filename.</code> TFTP 構文 <code>tftp://[location]/directory/-filename.</code> <p>flash:/file-urlには、ローカルフラッシュ ファイル システム上の、新しいファイ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。</p>
ステップ 2	<p>archive tar /table source-url</p> <p>例 :</p> <pre>Device# archive tar /table flash: /new_configs</pre>	<p>ファイルの内容を表示します。</p> <p><i>source-url</i>には、ローカルファイル システムまたはネットワーク ファイル システムの送信元 URL エイリアスを指定します。<i>-filename.</i> は、表示するファイルです。次のオプションがサポートされています。</p> <ul style="list-style-type: none"> ローカルフラッシュ ファイル システム構文 flash: FTP 構文 <code>ftp://[username][password]@[location]/directory/-filename.</code> RCP 構文 <code>rcp://[username@location]/directory/-filename.</code> TFTP 構文 <code>tftp://[location]/directory/-filename.</code> <p>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</p>

	コマンドまたはアクション	目的
<p>ステップ 3</p>	<p>archive tar /xtract <i>source-url</i> flash:/<i>file-url</i> [<i>dir/file...</i>]</p> <p>例 :</p> <pre>Device# archive tar /xtract tftp://172.20.10.30/saved. flash:/new-configs</pre>	<p>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。</p> <p>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。</p> <p><i>source-url</i> には、ローカルファイルシステムの送信元 URL のエイリアスを指定します。-<i>filename.</i> は、ファイルの抽出元のファイルです。次のオプションがサポートされています。</p> <ul style="list-style-type: none"> • ローカルフラッシュファイルシステム構文 <p>flash:</p> <ul style="list-style-type: none"> • FTP 構文 <p>ftp:[[<i>/username</i>][<i>password</i>][<i>@location</i>][<i>directory</i>]/-<i>filename.</i></p> <ul style="list-style-type: none"> • RCP 構文 <p>rcp:[[<i>/username</i>][<i>@location</i>][<i>directory</i>]/-<i>filename.</i></p> <ul style="list-style-type: none"> • TFTP 構文 <p>tftp:[[<i>/location</i>][<i>directory</i>]/-<i>filename.</i></p> <p>flash:/<i>file-url</i> [<i>dir/file...</i>] には、ファイルの抽出元にするローカルフラッシュファイルシステム上の場所を指定します。抽出対象のファイル内のファイルまたはディレクトリのリストを指定するには、<i>dir/file...</i> オプションを使用します。何も指定されないと、すべてのファイルとディレクトリが抽出されます。</p>
<p>ステップ 4</p>	<p>more [<i>/ascii</i> <i>/binary</i> <i>/ebcdic</i>] <i>/file-url</i></p> <p>例 :</p> <pre>Device# more flash:/new-configs</pre>	<p>リモートファイルシステム上のファイルを含めて、読み取り可能なファイルの内容を表示します。</p>

フラッシュ ファイル システムに関するその他の関連資料

関連資料

関連項目	マニュアル タイトル
flash: ファイル システムの管理コマンド	<i>Cisco IOS Configuration Fundamentals Command Reference</i>

フラッシュ ファイル システムの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	フラッシュファイルシステム	フラッシュファイルシステムは、ファイルを格納できる単一のフラッシュ デバイスです。ソフトウェア バンドルおよびコンフィギュレーション ファイルの管理に役立つ複数のコマンドも備えています。

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



第 17 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](#) (449 ページ)
- [初期設定へのリセット実行の制限事項](#) (449 ページ)
- [初期設定へのリセットの実行に関する情報](#) (450 ページ)
- [初期設定へのリセットの実行方法](#) (451 ページ)
- [初期設定へのリセットを実行するための設定例](#) (452 ページ)
- [初期設定へのリセットの実行に関する追加情報](#) (454 ページ)
- [初期設定へのリセットに関する機能履歴](#) (454 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 25: 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
ユーザーデータ、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	ライセンス
ユーザーが追加した ROMmon 変数	—

初期設定へのリセットの実行方法

初期設定へのリセットを実行するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	<ul style="list-style-type: none"> • スタンドアロンデバイスの場合： factory-reset {all [secure 3-pass] config boot-vars} • スタック構成のデバイスの場合： factory-reset {all [secure 3-pass] config boot-vars switch {switch-number all {all [secure 3-pass] config boot-vars}} 例： Device# factory-reset all または 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。 factory reset コマンドを使用するために必要なシステム設定はありません。 次のオプションを使用できます。 <ul style="list-style-type: none"> • all : NVRAM のすべての内容、現在のブートイメージ、ブート変数、起動コンフィギュレーションと実行コンフィギュレーションのデータ、およびユーザー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。 • 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。 <ul style="list-style-type: none"> • 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
		<ul style="list-style-type: none"> • 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。 • 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。 <p>(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。</p> <ul style="list-style-type: none"> • config : スタートアップ コンフィギュレーションをリセットします。 • boot-vars : ユーザーによって追加されたブート変数を消去します。 • switch {switch-number all}: <ul style="list-style-type: none"> • 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。 • all : スタック内のすべてのスイッチを選択します。 <p>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。</p>

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、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-99f0-6dfd1d941701e
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履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	工場出荷時の状態へのリセット (Factory Reset)	初期設定にリセットすると、デバイスに保存されているお客様固有のデータがすべて消去され、デバイスの設定は出荷時の元の設定に復元されます

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.12.1	リムーバブルストレージデバイスの初期設定へのリセット	初期設定へのリセットを実行すると、SATA、SSD、USB などのリムーバブルストレージデバイスの内容が消去されます。
Cisco IOS XE Amsterdam 17.2.1	3-pass 上書きによる初期設定へのリセット	初期設定へのリセットを実行すると、デバイスからすべてのコンテンツを 3-pass 上書きで安全に消去できます。secure 3-pass キーワードが導入されました。
	スタックおよびCisco StackWise Virtual の初期設定へのリセットオプションの拡張	スタック構成デバイスおよびCisco StackWise Virtual 対応デバイスで初期設定へのリセットのサポートが導入されました。

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



第 18 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 \(457 ページ\)](#)
- [セキュアストレージの有効化 \(457 ページ\)](#)
- [セキュアストレージの無効化 \(458 ページ\)](#)
- [暗号化のステータスの確認 \(459 ページ\)](#)
- [セキュアストレージの機能情報 \(459 ページ\)](#)

セキュアストレージについて

セキュアストレージ機能では、重要な設定情報を暗号化して保護できます。非対称キーペア、事前共有秘密、タイプ 6 のパスワード暗号化キーおよび特定のクレデンシャルを暗号化します。インスタンス固有の暗号キーは、危険にさらされることを防ぐためにハードウェアのトラストアンカーに保管されます。

セキュアストレージの有効化

始める前に

この機能はデフォルトで無効になっています。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 2	service private-config-encryption 例： Device(config)# service private-config-encryption	デバイスでセキュアストレージ機能を有効にします。

	コマンドまたはアクション	目的
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 4	write memory 例： Device# write memory	private-config ファイルを暗号化し、暗号化フォーマットで保存します。

セキュアストレージの無効化

始める前に

デバイスでセキュアストレージ機能を無効にするには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	no service private-config-encryption 例： Device(config)# no service private-config-encryption	デバイスでセキュリティストレージ機能を無効にします。セキュアストレージを無効にすると、すべてのユーザーデータがプレーンテキストで NVRAM に保存されます。
ステップ 3	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 4	write memory 例： Device# write memory	private-config ファイルを復号し、プレーンフォーマットで保存します。

暗号化のステータスの確認

暗号化のステータスを確認するには、**show parser encrypt file status** コマンドを使用します。次のコマンド出力は、機能は利用できるが、ファイルが暗号化されていないことを示します。ファイルは「プレーンテキスト」形式です。

```
Device#show parser encrypt file status
Feature: Enabled
File Format: Plain Text
Encryption Version: Ver1
```

セキュアストレージの機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	セキュアなストレージ	セキュアストレージ機能では、重要な設定情報を暗号化して保護できます。非対称キーペア、事前共有秘密、タイプ6のパスワード暗号化キーおよび特定のクレデンシャルを暗号化します。インスタンス固有の暗号キーは、危険にさらされることを防ぐためにハードウェアのトラストアンカーに保管されます。

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



第 19 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 \(461 ページ\)](#)
- [ラジオアクティブトレースの概要 \(462 ページ\)](#)
- [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 \(462 ページ\)](#)
- [条件付きデバッグのモニターリング \(466 ページ\)](#)
- [条件付きデバッグの設定例 \(467 ページ\)](#)
- [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 \(468 ページ\)](#)
- [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 \(468 ページ\)](#)

条件付きデバッグの概要

条件付きデバッグ機能によって、定義した条件に基づき、特定の機能のデバッグおよびロギングを選択して有効にすることができます。この機能は、多くの機能がサポートされているシステムで有用です。



(注) コントロールプレーントレースのみがサポートされています。

条件付きデバッグでは、多数の機能が導入されていて大規模に稼働しているネットワークにおけるきめ細かなデバッグが可能です。これにより、システム内の細かなインスタンスに対しても、詳細なデバッグを実行できます。これは、何千ものセッションのうち特定のセッションのみをデバッグするような場合に、非常に有用です。条件は複数指定することもできます。

条件とは、機能またはアイデンティティをいいます。アイデンティティは、インターフェイス、IP アドレス、MAC アドレスなどです。



(注) サポートされる条件は MAC アドレスであることのみです。

これは、処理する機能オブジェクトを区別せずに出力を生成する、一般的なデバッグコマンドとは対照的です。一般的なデバッグコマンドは、多数のシステムリソースを消費し、システムパフォーマンスに影響します。

ラジオアクティブトレースの概要

ラジオアクティブトレースにより、冗長性のレベルを高めた状態で、システムの全体にわたって目的とする動作を連鎖的に実行できます。また、複数のスレッド、プロセス、および関数呼び出しにわたって、デバッグ情報を条件に基づいて（DEBUG レベルまで、または指定のレベルまで）出力する方法を提供します。



(注) デフォルトのレベルは **DEBUG** です。ユーザーは別のレベルに変更することはできません。

ラジオアクティブトレースでは、次の機能が有効になっています。

- IGMP スヌーピング
- レイヤ 2 マルチキャスト

条件付きデバッグとラジオアクティブトレースの設定方法

条件付きデバッグおよび放射線トレース

条件付きデバッグと組み合わせた放射線トレースによって、条件に関連するすべての実行コンテキストをデバッグする単一のデバッグ CLI を取得できます。これは、ボックス内の機能のさまざまな制御フロープロセスを認識していなくても行うことができ、これらのプロセスでデバッグを個別に発行する必要もありません。

トレースファイルの場所

デフォルトでは、トレースファイルログは各プロセスで生成され、`/tmp/rp/trace` または `/tmp/fp/trace` ディレクトリに保存されます。この一時ディレクトリで、トレースログがファイルに書き込まれます。各ファイルは 1 MB サイズです。このディレクトリでは、特定のプロセスのこうしたファイルを、最大 25 件保持できます。`/tmp` ディレクトリのトレースファイルがその 1 MB 制限またはブート時に設定されたサイズに達した場合、ローテーションから外れ、`tracelogs` ディレクトリの `/crashinfo` パーティションの下にあるアーカイブの場所に移動します。

`/tmp` ディレクトリが1つのプロセスで保持するトレースファイルは1つのみです。ファイルがそのファイルサイズの制限に達すると、ローテーションから外れ、`/crashinfo/tracelogs` に移動します。アーカイブ ディレクトリに蓄積されるファイルは最大 25 ファイルであり、その後は最も古いものから順に、`/tmp` から新たにローテーションされたファイルに置換されます。

`crashinfo` ディレクトリ内のトレースファイルは次の形式で配置されます。

1. Process-name_Process-ID_running-counter.timestamp.gz
例 : IOSRP_R0-0.bin_0.14239.20151101234827.gz
2. Process-name_pmanlog_Process-ID_running-counter.timestamp.bin.gz
例 : wcm_pmanlog_R0-0.30360_0.20151028233007.bin.gz

条件付きデバッグの設定

条件付デバッグを設定するには、以下の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : 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	debug platform condition mac {mac-address} 例 : 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 例 : 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 例 : Device# show platform condition 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 例 : 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:}] 例 : <pre># request platform software trace archive last 2 days</pre>	(任意) システムのマージされたトレースファイルの履歴ログを表示します。日数またはロケーションの組み合わせのフィルタ。
ステップ 7	show platform software trace [filter-binary level message] 例 : <pre>Device# show platform software trace message</pre>	(任意) 最新のトレースファイルからマージされたログを表示します。アプリケーションの状態、トレース モジュール名およびトレース レベルをさまざまな組み合わせでフィルタリングします。 <ul style="list-style-type: none"> • filter-binary : 照合するモジュールをフィルタリングします。 • level : トレース レベルを表示します。 • message : トレース メッセージのリングの内容を表示します。 (注) デバイス上では次が可能です。 <ul style="list-style-type: none"> • Linux シェルだけでなく、IOS のコンソールからも使用できます。 • マージされたログでファイルを生成します。 • ステージング エリアからのみマージされたログを表示します。
ステップ 8	clear platform condition all 例 : <pre>Device# clear platform condition all</pre>	すべての条件をクリアします。

次のタスク



(注) **request platform software trace filter-binary** コマンドと **show platform software trace filter-binary** コマンドは同様の動作をします。唯一の違いは次のとおりです。

- **request platform software trace filter-binary** : データ ソースとして履歴ログを使用します。
- **show platform software trace filter-binary** : データ ソースとしてフラッシュの一時ディレクトリを使用します。

その中でも、`mac_log <..date.>` は、デバッグする MAC 用のメッセージを伝えるため、最も重要なファイルです。コマンド **show platform software trace filter-binary** も同じフラッシュ ファイルを生成し、また、画面に `mac_log` を出力します。

L2 マルチキャストの放射線トレース

特定のマルチキャスト受信者を特定するには、参加者または受信側クライアントの MAC アドレス、グループのマルチキャスト IP アドレスおよびスヌーピング VLAN を指定します。また、デバッグのトレース レベルを有効にします。デバッグ レベルでは、詳細なトレースとシステムへの高い可視性が提供されます。

```
debug platform condition feature multicast controlplane mac client MAC address ip Group  
IP address vlan id level debug level
```

トレース ファイルの推奨ワークフロー

トレース ファイルの推奨ワークフローの概要は次のとおりです。

1. 特定の時間帯のトレースログを要求する場合。
たとえば 1 日。
使用するコマンドは、次のとおりです。

```
Device#request platform software trace archive last 1 day
```
2. システムは、`/flash:` ロケーション内のトレースログの tar ball (`.gz` ファイル) を生成します。
3. スイッチ外にファイルをコピーします。ファイルをコピーすることによって、オフラインでトレースログが使用できます。ファイルのコピーについての詳細は、次のセクションを参照してください。
4. `/flash: location` からトレースログファイル (`.gz`) ファイルを削除します。これにより、他の操作に十分な領域がスイッチに確保されます。

ボックス外へのトレース ファイルのコピー

トレース ファイルの例を以下に示します。

```
Device# dir crashinfo:/tracelogs
Directory of crashinfo:/tracelogs/

50664 -rwx 760 Sep 22 2015 11:12:21 +00:00 plogd_F0-0.bin_0.gz
50603 -rwx 991 Sep 22 2015 11:12:08 +00:00 fed_pmanlog_F0-0.bin_0.9558.20150922111208.gz
50610 -rw- 11 Nov 2 2015 00:15:59 +00:00 timestamp
50611 -rwx 1443 Sep 22 2015 11:11:31 +00:00
auto_upgrade_client_sh_pmanlog_R0-.bin_0.3817.20150922111130.gz
50669 -rwx 589 Sep 30 2015 03:59:04 +00:00 cfgwr-8021_R0-0.bin_0.gz
50612 -rwx 1136 Sep 22 2015 11:11:46 +00:00 reflector_803_R0-0.bin_0.1312.20150922111116.gz
50794 -rwx 4239 Nov 2 2015 00:04:32 +00:00 IOSRP_R0-0.bin_0.14239.20151101234827.gz
50615 -rwx 131072 Nov 2 2015 00:19:59 +00:00 linux_iosd_image_pmanlog_R0-0.bin_0
--More--
```

トレース ファイルは、次に示すさまざまなオプションのいずれかを使用して、コピーできます。

```
Device# copy crashinfo:/tracelogs ?
crashinfo: Copy to crashinfo: file system
flash: Copy to flash: file system
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system
```

TFTP サーバー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ーリング

以下の表に、条件付きデバッグのモニター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。
show platform software trace filter-binary	最新のトレース ファイルからマージされたログを表示します。
request platform software trace filter-binary	システムにマージされたトレース ファイルの履歴ログを表示します。

条件付きデバッグの設定例

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

```
Device# show platform condition
Conditional Debug Global State: Stop
Conditions Direction
```

```
-----|-----
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
```

```
-----|-----
Device#
```

次に、*show debug* コマンドの出力例を示します。

```
Device# show debug
IOSXE Conditional Debug Configs:
Conditional Debug Global State: Start
Conditions Direction
```

```
-----|-----
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
```

```
-----|-----
Packet Infra debugs:
Ip Address Port
```

```
-----|-----
Device#
```

次に、*debug platform condition stop* コマンドの例を示します。

```
Device# debug platform condition stop
Conditional Debug Global State: Stop
```

条件付きデバッグとラジオアクティブトレースに関するその他の関連資料

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

条件付きデバッグとラジオアクティブトレースの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	条件付きデバッグとラジオアクティブトレース	条件付きデバッグ機能によって、定義した条件に基づき、特定の機能のデバッグおよびロギングを選択して有効にすることができます。

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



第 20 章

同意トークン

- [同意トークンの制約事項 \(469 ページ\)](#)
- [同意トークンに関する情報 \(470 ページ\)](#)
- [システムシェルアクセスの同意トークン承認プロセス \(470 ページ\)](#)
- [同意トークンの機能履歴 \(472 ページ\)](#)

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ー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ー（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。

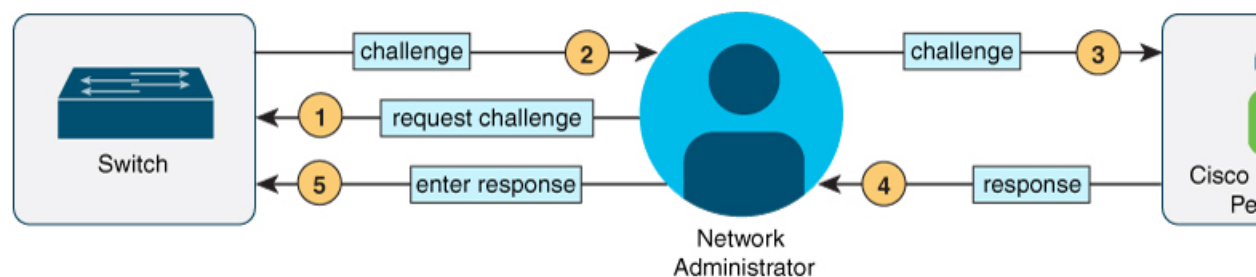
システムシェルへのアクセスを要求する場合は、認証を受ける必要があります。最初にコマンドを実行し、デバイスの同意トークン機能を使用してチャレンジを生成する必要があります。デバイスは、固有のチャレンジを出力として生成します。このチャレンジ文字列をコピーし、電子メールまたはインスタントメッセージでシスコ認定担当者に送信する必要があります。

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 15: 同意トークン



システムシェルアクセスの同意トークン承認プロセス

ここでは、システムシェルにアクセスするための同意トークン承認のプロセスについて説明します。

手順

ステップ 1 指定された期間、システムシェルへのアクセスを要求するチャレンジを生成します。

例：

```
Device# request consent-token generate-challenge shell-access auth-timeout 900
% Consent token authorization success
Device#
*Jan 18 02:47:06.733: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (challenge generation
attempt: Shell access 0).
```

request consent-token generate-challenge shell-access time-validity-slot コマンドを使用して、チャレンジの要求を送信します。システムシェルへのアクセスを要求する期間（分単位）は、**time-slot-period** です。

この例の期間は、セッションの期限切れ後 900 分です。

デバイスは、固有のチャレンジを出力として生成します。このチャレンジは、base-64 形式の文字列です。

ステップ 2 シスコ認定担当者にチャレンジ文字列を送信します。

デバイスによって生成されたチャレンジ文字列を、電子メールまたはインスタントメッセージでシスコ認定担当者に送信します。

シスコ認定担当者は固有のチャレンジ文字列を処理し、レスポンスを生成します。レスポンスもまた、固有の base-64 文字列です。シスコ認定担当者はこのレスポンス文字列をコピーし、電子メールまたはインスタントメッセージで送信します。

ステップ 3 デバイスにレスポンス文字列を入力します。

例：

```
Device# request consent-token accept-response shell-access
% Consent token authorization success
Device#
*Jan 18 02:51:37.807: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (authentication success:
Shell access 0).
```

```
Device# request platform software system shell
Activity within this shell can jeopardize the functioning of the system.
Are you sure you want to continue? [y/n] y
Device#
*Jan 18 02:56:59.714: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (authorization for
Shell access 0 will expire in 10 min).
```

request consent-token accept-response shell-access response-string コマンドを使用して、シスコ認定担当者から送信されたレスポンス文字列を入力します。

チャレンジ/レスポンスペアが一致すると、システムシェルへのアクセスが許可されます。チャレンジ/レスポンスペアが一致しない場合は、エラーが表示され、手順 1 ~ 3 を繰り返す必要があります。

承認されると、要求されたタイムスロットのシステムシェルにアクセスできます。

承認セッションの残り時間が 10 分になると、デバイスはメッセージを送信します。

ステップ 4 セッションを終了します。

例：

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

Device#
*Jan 18 23:33:02.937: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (terminate
authentication: Shell access 0).
Device#
```

システムシェルへのアクセスが終了したら、**request consent-token terminate-auth** コマンドを使用してセッションを終了できます。このコマンドを使用して、承認タイムアウトの前にセッションを強制終了することもできます。要求したタイムスロットが期限切れになると、セッションも自動的に終了します。

同意トークンの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.11.1	同意トークン	同意トークンは、ネットワーク管理者と Cisco Technical Assistance Centre (Cisco TAC) の相互の同意により、システムシェルにアクセスする組織のネットワーク管理者を認証するために使用されるセキュリティ機能です。

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



第 21 章

ソフトウェア設定のトラブルシューティング

この章では、スイッチが稼働する Cisco IOS ソフトウェアに関連する問題を特定し、解決する方法について説明します。問題の性質に応じて、コマンドラインインターフェイス (CLI)、デバイス マネージャ、または Network Assistant を使用して、問題を特定し解決できます。

LEDの説明など、トラブルシューティングの詳細については、ハードウェアインストールガイドを参照してください。

- [ソフトウェア設定のトラブルシューティングに関する情報 \(473 ページ\)](#)
- [ソフトウェア設定のトラブルシューティング方法 \(483 ページ\)](#)
- [ソフトウェア設定のトラブルシューティングの確認 \(497 ページ\)](#)
- [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 \(500 ページ\)](#)
- [ソフトウェアのトラブルシューティングの設定例 \(505 ページ\)](#)
- [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 \(507 ページ\)](#)
- [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 \(507 ページ\)](#)

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。いずれの場合にも、スイッチは電源投入時自己診断テスト (POST) に失敗し、接続できなくなります。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 \(483 ページ\)](#)」セクションで説明されている手順を実行します。

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ー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、エンドユーザーがパスワードをリセットしようとする、ステータスメッセージで回復プロセスの間はデフォルトの設定に戻すように指示されます。



(注) Cisco WLC の設定を複数の Cisco WLC 間でコピーすると、暗号化パスワード キーを回復できなくなります (RMA の場合)。

パスワードを紛失または忘れた場合にそのパスワードを回復するには、[パスワードを忘れた場合の回復 \(487 ページ\)](#) 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。受電デバイスが PoE ポートにだけ接続されている場合、受電デバイス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、『*Interface and Hardware Component Configuration Guide (Catalyst 9300 Switches)*』の「Configuring PoE」の章を参照してください。

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 \(500 ページ\)](#) 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（494 ページ）](#) の項を参照してください。

レイヤ 2 トレースルート

レイヤ 2 トレースルート機能により、パケットが通過する送信元デバイスから宛先デバイスまでの物理パスを識別できます。レイヤ 2 トレースルートは、ユニキャストの送信元および宛先 MAC アドレスだけをサポートします。トレースルートは、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 を無効にしないでください。

物理パス内のデバイスが CDP に対して透過的な場合、スイッチはこれらのデバイスを通過するパスを識別できません。

- ping 特権 EXEC コマンドを使用して接続をテストできれば、このデバイスは別のデバイスから到達可能であると定義できます。物理パス内のすべてのデバイスは、他のデバイスから相互に到達可能でなければなりません。
- パス内で識別可能な最大ホップ数は 10 です。
- 送信元デバイスと宛先デバイス間の物理パス内にないデバイスで、**traceroute mac** または **traceroute mac ip** の特権 EXEC コマンドを実行できます。パス内のすべてのデバイスは、このスイッチから到達可能でなければなりません。
- 指定された送信元および宛先アドレスが同じ VLAN にある場合、**traceroute mac** コマンド出力はレイヤ2パスを表示します。指定した送信元および宛先 MAC アドレスが、それぞれ異なる VLAN に属している場合は、レイヤ2パスは識別されず、エラーメッセージが表示されます。
- マルチキャストの送信元または宛先 MAC アドレスを指定すると、パスは識別されず、エラーメッセージが表示されます。
- 送信元または宛先 MAC アドレスが複数の VLAN に属する場合は、送信元および宛先 MAC アドレスの両方が属している VLAN を指定する必要があります。VLAN を指定しないと、パスは識別されず、エラーメッセージが表示されます。
- 指定された送信元および宛先の IP アドレスが同一のサブネット内にある場合、**traceroute mac ip** コマンド出力はレイヤ2パスを表示します。IP アドレスを指定した場合、デバイスは Address Resolution Protocol (ARP) を使用し、IP アドレスとそれに対応する MAC アドレスおよび VLAN ID を対応させます。
 - 指定の IP アドレスの ARP のエントリが存在している場合、デバイスは関連付けられた MAC アドレスを使用し、物理パスを識別します。
 - ARP のエントリが存在しない場合、デバイスは ARP クエリを送信し、IP アドレスを解決しようと試みます。IP アドレスが解決されない場合は、パスは識別されず、エラーメッセージが表示されます。

- 複数のデバイスがハブを介して1つのポートに接続されている場合（たとえば複数の CDP ネイバーがポートで検出された場合）、レイヤ 2 **traceroute** 機能はサポートされません。複数の CDP ネイバーが1つのポートで検出された場合、レイヤ 2 パスは特定されず、エラーメッセージが表示されます。
- この機能は、トークンリング VLAN ではサポートされません。
- レイヤ 2 トレースルートは、ユーザデータグラム プロトコル (UDP) ポート 2228 でリスニングソケットを開きます。このポートは、任意の IPv4 アドレスを使用してリモートからアクセスでき、認証は必要ありません。この UDP ソケットにより、VLAN 情報、リンク、特定の MAC アドレスの存在、および CDP ネイバー情報をデバイスから読み取ることができます。この情報を使用することにより、最終的にレイヤ 2 ネットワークトポロジの全体像を構築できます。
- レイヤ 2 トレースルートはデフォルトで有効になっており、グローバルコンフィギュレーションモードで **no l2 traceroute** コマンドを実行することによって無効にできます。レイヤ 2 トレースルートを再度有効にするには、グローバルコンフィギュレーションモードで **l2 traceroute** コマンドを使用します。

IP トレースルート

IP **traceroute** を使用すると、ネットワーク上でパケットが通過するパスをホップバイホップで識別できます。このコマンドを実行すると、トラフィックが宛先に到達するまでに通過するルータなどのすべてのネットワーク層（レイヤ 3）デバイスが表示されます。

デバイスは、**traceroute** 特権 EXEC コマンドの送信元または宛先として指定できます。また、**traceroute** コマンドの出力でホップとして表示される場合があります。デバイスを **traceroute** の宛先とすると、スイッチは、**traceroute** の出力で最終の宛先として表示されます。中間デバイスが同じ VLAN 内でポート間のパケットのブリッジングだけを行う場合、**traceroute** の出力に中間スイッチは表示されません。ただし、中間デバイスが特定の packets をルーティングするマルチレイヤデバイスの場合、このデバイスは **traceroute** の出力にホップとして表示されます。

traceroute 特権 EXEC コマンドは、IP ヘッダーの存続可能時間 (TTL) フィールドを使用して、ルータおよびサーバで特定のリターンメッセージが生成されるようにします。**traceroute** の実行は、ユーザデータグラム プロトコル (UDP) データグラムを、TTL フィールドが 1 に設定されている宛先ホストへ送信することから始まります。ルータで TTL 値が 1 または 0 であることを検出すると、データグラムをドロップし、インターネット制御メッセージプロトコル (ICMP) **time-to-live-exceeded** メッセージを送信元に送信します。**traceroute** は、ICMP **time-to-live-exceeded** メッセージの送信元アドレス フィールドを調べて、最初のホップのアドレスを判別します。

ネクストホップを識別するために、**traceroute** は TTL 値が 2 の UDP パケットを送信します。1 番目のルータは、TTL フィールドの値から 1 を差し引いて次のルータにデータグラムを送信します。2 番目のルータは、TTL 値が 1 であることを確認すると、このデータグラムを廃棄し、**time-to-live-exceeded** メッセージを送信元へ返します。このように、データグラムが宛先ホストに到達するまで（または TTL の最大値に達するまで）TTL の値は増分され、処理が続けられます。

データグラムが宛先に到達したことを学習するために、**traceroute** は、データグラムの UDP 宛先ポート番号を、宛先ホストが使用する可能性のない大きな値に設定します。ホストが、ローカルで使用されない宛先ポート番号を持つ自分自身宛てのデータグラムを受信すると、送信元に ICMP ポート到達不能エラーを送信します。ポート到達不能エラーを除くすべてのエラーは中間ホップから送信されるため、ポート到達不能エラーを受信するという事は、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（506 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 \(495 ページ\)](#) 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が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メンバーで生成されます。スタック内の他のメンバーではレポートは生成されません。
- 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 switch active** コマンドを使用します。

```
Device# request platform software process core fed switch active
SUCCESS: Core file generated.
```

```
Device# dir bootflash:/core
Directory of bootflash:/core/
16430  -rw-          10941657  Apr 6 2022 00:15:20 +00:00
Switch_1_RP_0_fed_18469_20220406-001511-UTC.core.gz
16812  -rw-           1  Apr 6 2022 00:01:48 +00:00 .callhome
16810  drwx           4096  Jan 18 2022 21:10:35 +00:00 modules
```

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。

```
Device# 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。

```
Device# 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 サーバー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 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]?

```

スタックの全メンバーからのトレースログは、`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。

```

Device# 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。

```

Device# request platform software trace archive last ?
<1-3650> Number of days (1-3650)
Device# 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またはスイッチスタックメンバに入力された OBFL CLI コマンドの記録。
- 環境データ：スタンドアロンデバイスまたはスイッチスタックメンバおよび接続されているすべての FRU デバイスの一意のデバイス ID (UDI) 情報、製品 ID (PID)、バージョン ID (VID)、およびシリアル番号。
- メッセージ：スタンドアロンデバイスまたはスイッチスタックメンバ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またはスイッチスタックメンバの 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またはスイッチスタックメンバの温度。
- 稼働時間：スタンドアロンデバイスまたはスイッチスタックメンバ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 またはスイッチスタックメンバのシステム電圧。

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。

```
Multiple fan(FRU/PS) failure detected. System may get overheated. Change fan quickly.
```

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。

ファン障害機能をイネーブルにするには、**system env fan-fail-action shut** 特権 EXEC コマンドを入力します。デバイス内の複数のファンに障害が発生した場合、デバイスは自動的にシャットダウンし、次のようなエラーメッセージが表示されます。

```
Faulty (FRU/PS) fans detected, shutting down system!
```

最初のファンの停止後、デバイスが2つ目のファンの障害を検知すると、デバイスは20秒待機してからシャットダウンします。

デバイスを再起動するには、電源をオフにしてから再度オンにする必要があります。

CPU 使用率が高い場合に起こりうる症状

CPU使用率が高すぎることで次の現象が発生する可能性があります。他の原因で発生する場合もあります。次にその一部を示します。

- スパニングツリー トポロジの変更
- 通信が切断されたために EtherChannel リンクがダウンした
- 管理要求 (ICMP ping、SNMP のタイムアウト、低速な Telnet または SSH セッション) に応答できない
- UDLD フラッピング
- SLA の応答が許容可能なしきい値を超えたことによる IP SLA の失敗
- スイッチが要求を転送しない、または要求に応答しない場合の DHCP または IEEE 802.1x の処理の失敗

ソフトウェア設定のトラブルシューティング方法

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。

スイッチのコンソールポートのデフォルトレートである 9600 ビット/秒 (bps) と一致するように、端末のボーレートを設定します。ボーレートが 9600 bps 以外の値に設定されている場合、速度がデフォルトに戻るまでコンソールへのアクセスは失われます。

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 サーバーにソフトウェア 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** [Mode] ボタンを押しながら、電源コードをスイッチに再接続します。
- ステップ6** ブートローダープロンプトで、TFTP サーバーに 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 サーバーに ping を実行できることを確認します。**switch: ping ip_address_of_TFTP_server**

例：

```
switch: ping 192.0.2.15
ping 192.0.2.1 with 32 bytes of data...
Host 192.0.2.1 is alive.
switch:
```

- ステップ7** 次のいずれかを選択します。

- ブートローダープロンプトで、**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。

```
switch: 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!
```

- リカバリパーティションからソフトウェアをインストールします。この回復イメージは、**emergency-install** 機能を使用して回復を実施する場合に必要となります。

- a) 回復パーティション (sda9:) に回復イメージが存在することを確認します。

例 :

```
switch: dir sda9:

Size              Attributes      Name
-----
21680202         -rw-           cat9k-recovery.SSA.bin
-----
```

- b) ブートローダープロンプトで、**emergency-install** 機能を開始します。この機能を使用すると、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。**警告** : **emergency-install** コマンドを実行すると、ブートブラッシュ全体が消去されます。

例 :

```
switch: emergency-install
tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin
WARNING: The system partition (bootflash:) will be erased during the system recovery
install process.
Are you sure you want to proceed? [y] y/n [n]: y
Starting system recovery
(tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin) ...
Attempting to boot from [sda9:cat9k-recovery.SSA.bin]
Located cat9k-recovery.SSA.bin
#####

Warning: ignoring ROMMON var "BOOT_PARAM"

PLATFORM_TYPE C9X00 speed 9600

Booting Recovery Image 16.5.1a

Initiating Emergency Installation of bundle
tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin

Downloading bundle
tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
curl_vrf=2
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload  Total   Spent    Left   Speed
100 485M  100 485M    0     0  5143k      0  0:01:36  0:01:36 ---:--- 5256k
100 485M  100 485M    0     0  5143k      0  0:01:36  0:01:36 ---:--- 5143k
```

```
Validating bundle
tftp://10.255.254.254/auto/tftpbboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Installing bundle
tftp://10.255.254.254/auto/tftpbboot/X86/cat9k_iosxe.16.05.01a.SPA.bin....
Verifying bundle
tftp://10.255.254.254/auto/tftpbboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Package cat9k-cc_srdriver.16.05.01a.SPA.pkg
/temp//stage/cat9k-cc_srdriver.16.05.01a.SPA.pkg is Digitally Signed
Package cat9k-espbase.16.05.01a.SPA.pkg /temp//stage/cat9k-espbase.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-guestshell.16.05.01a.SPA.pkg
/temp//stage/cat9k-guestshell.16.05.01a.SPA.pkg is Digitally Signed
Package cat9k-rpbase.16.05.01a.SPA.pkg /temp//stage/cat9k-rpbase.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-sipbase.16.05.01a.SPA.pkg /temp//stage/cat9k-sipbase.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-sipspa.16.05.01a.SPA.pkg /temp//stage/cat9k-sipspa.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-srdriver.16.05.01a.SPA.pkg /temp//stage/cat9k-srdriver.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-webui.16.05.01a.SPA.pkg /temp//stage/cat9k-webui.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-wlc.16.05.01a.SPA.pkg /temp//stage/cat9k-wlc.16.05.01a.SPA.pkg is
Digitally Signed
Package /cat9k-rpboot.16.05.01a.SPA.pkg /temp//rpboot/cat9k-rpboot.16.05.01a.SPA.pkg
is Digitally Signed
Preparing flash....
Flash filesystem unmounted successfully /dev/sdb3
Syncing device....
Emergency Install successful... Rebooting
Will reboot now

Initializing Hardware...

System Bootstrap, Version 16.5.2r, RELEASE SOFTWARE (P)
Compiled Wed 05/31/2017 15:58:35.22 by rel

Current image running:
Primary Rommon Image

Last reset cause: SoftwareReload
C9X00 platform with 8388608 Kbytes of main memory
```

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、回復プロセスの間、ステータス メッセージにその旨が表示されます。

手順

ステップ 1 端末または PC をスイッチに接続します。

- 端末または端末エミュレーションソフトウェアが稼働している PC をスイッチのコンソールポートに接続します。スイッチスタックのパスワードを回復する場合は、アクティブスイッチのコンソールポートに接続します。
- PC をイーサネット管理ポートに接続します。スイッチ スタックのパスワードを回復する場合は、スタック メンバのイーサネット管理ポートに接続します。

ステップ 2 エミュレーションソフトウェアの回線速度を 9600 ボーに設定します。

ステップ 3 スタンドアロン スイッチまたはスイッチ スタック全体の電源を切断します。

ステップ 4 スイッチまたはアクティブスイッチに電源コードを再接続します。システム LED が点滅したら、すぐに [Mode] ボタンを 2〜3 回押して放します。スイッチは ROMMON モードを開始します。

リロード中に次のコンソールメッセージが表示されます。

```
Initializing Hardware...
```

```
System Bootstrap, Version 16.6.1r [FC1], RELEASE SOFTWARE (P)
Compiled Sat 07/15/2017 8:31:57.39 by rel
```

```
Current image running:
Primary Rommon Image
```

```
Last reset cause: SoftwareReload <---- Start pressing and releasing the mode
button
C9300-24U platform with 8388608 Kbytes of main memory
```

```
attempting to boot from [flash:packages.conf]
```

```
Located file packages.conf
```

```
#
```

```
#####
```

```
Unable to load cat9k-rpboot.16.06.02b.SPA.pkg
Failed to boot file flash:user/packages.conf
ERROR: failed to boot from flash:packages.conf (Aborted) <--- will abort
switch:
switch: <---- ROMMON
```

「パスワード回復がイネーブルになっている場合の手順」セクションに記載されている手順を実行します。

ステップ5 パスワードの回復後、スイッチまたはアクティブスイッチをリロードします。

スイッチの場合

```
Switch> reload
Proceed with reload? [confirm] y
```

アクティブ スイッチの場合

```
Switch> reload slot <stack-active-member-number>
Proceed with reload? [confirm] y
```

ステップ6 スタック内の残りのスイッチに電源を投入します。

パスワード回復がイネーブルになっている場合の手順

手順

ステップ1 次のコマンドを使用して、スタートアップ コンフィギュレーションを無視します。

```
Device: SWITCH_IGNORE_STARTUP_CFG=1
```

ステップ2 `packages.conf` ファイルでスイッチをフラッシュからブートします。

```
Device: boot flash:packages.conf
```

ステップ3 **No** と応答して初期設定ダイアログを終了します。

```
Would you like to enter the initial configuration dialog? [yes/no]: No
```

ステップ4 スイッチ プロンプトで、特権 EXEC モードを開始します。

```
Device> enable
Device#
```

ステップ5 スタートアップ コンフィギュレーションを実行コンフィギュレーションにコピーします。

```
Device# copy startup-config running-config Destination filename [running-config]?
```

確認を求めるプロンプトに、Return を押して応答します。これで、コンフィギュレーション ファイルがリロードされ、パスワードを変更できます。

ステップ6 グローバルコンフィギュレーションモードを開始して、イネーブルパスワードを変更します。

```
Device# configure terminal
```

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

```
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 SWITCH_IGNORE_STARTUP_CFG パラメータを 0 に設定します。

```
Device(config)# no system ignore startupconfig switch all
Device(config)# end
Device# write memory
```

ステップ12 フラッシュの *packages.conf* ファイルを使用して、デバイスを起動します。

```
Device: boot flash:packages.conf
```

ステップ13 デバイスが起動したら、デバイスで手動ブートを無効にします。

```
Device(config)# no boot manual
```

パスワード回復がディセーブルになっている場合の手順

パスワード回復メカニズムがディセーブルの場合、次のメッセージが表示されます。

```
The password-recovery mechanism has been triggered, but
is currently disabled. Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
```

```
this point. However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.
```

```
Would you like to reset the system back to the default configuration (y/n)?
```



注意 デバイスをデフォルト設定に戻すと、既存の設定がすべて失われます。システム管理者に問い合わせ、バックアップデバイスと VLAN（仮想 LAN）コンフィギュレーションファイルがあるかどうかを確認してください。

- **n** (no) を入力すると、Mode ボタンを押さなかった場合と同様に、通常のブートプロセスが継続されます。ブートローダプロンプトにはアクセスできません。したがって、新しいパスワードを入力できません。次のメッセージが表示されます。

```
Press Enter to continue.....
```

- **y** (yes) を入力すると、フラッシュメモリ内のコンフィギュレーションファイルおよび VLAN データベースファイルが削除されます。デフォルト設定がロードされるときに、パスワードをリセットできます。

手順

ステップ 1 パスワード回復手順の継続を選択すると、既存の設定が失われます。

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

ステップ 2 フラッシュメモリの内容を表示します。

```
Device: dir flash:
```

デバイスのファイルシステムが表示されます。

```
Directory of flash:/
.
.
.i'
15494 drwx      4096   Jan 1 2000 00:20:20 +00:00 kirch
15508 -rw-     258065648   Sep 4 2013 14:19:03 +00:00
cat9k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin
162196684
```

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#
```

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。

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、接続ポート上で手で設定された速度またはデュプレックスのパラメータと異なっている場合。

- ポートを自動ネゴシエーションに設定したが、接続先ポートは自動ネゴシエーションを使用しない全二重に設定されている場合。

デバイスのパフォーマンスを最大限に引き出してリンクを確保するには、次のいずれかの注意事項に従って、デュプレックスおよび速度の設定を変更してください。

- 速度とデュプレックスの両方について、両方のポートで自動ネゴシエーションを実行させます。
- 接続の両側でポートの速度とデュプレックスのパラメータを手動で設定します。



- (注) 接続先装置が自動ネゴシエーションを実行しない場合は、2つのポートのデュプレックス設定を一致させます。速度パラメータは、接続先のポートが自動ネゴシエーションを実行しない場合でも自動調整が可能です。

SFP モジュールのセキュリティと識別に関するトラブルシューティング

シスコの Small Form-Factor Pluggable (SFP) モジュールは、モジュールのシリアル番号、ベンダー名とベンダー ID、一意のセキュリティコード、および巡回冗長検査 (CRC) が格納されたシリアル EEPROM (電氣的に消去可能でプログラミング可能な ROM) を備えています。デバイスに SFP モジュールを装着すると、デバイスソフトウェアは、EEPROM を読み取ってシリアル番号、ベンダー名、およびベンダー ID を確認し、セキュリティコードと CRC を再計算します。シリアル番号、ベンダー名、ベンダー ID、セキュリティコード、または CRC が無効な場合、ソフトウェアは、セキュリティ エラー メッセージを生成し、インターフェイスを **errdisable** ステートにします。



- (注) セキュリティ エラー メッセージは、**GBIC_SECURITY** 機能を参照します。スイッチは、SFP モジュールをサポートしていますが、**GBIC** (ギガビット インターフェイス コンバータ) モジュールはサポートしていません。エラーメッセージテキストは、**GBIC** インターフェイスおよびモジュールを参照しますが、セキュリティ メッセージは、実際は **SFP** モジュールおよびモジュール インターフェイスを参照します。

他社の SFP モジュールを使用している場合、デバイスから SFP モジュールを取り外し、シスコのモジュールに交換します。シスコの SFP モジュールを装着したら、**errdisable recovery cause gbic-invalid** グローバル コンフィギュレーション コマンドを使用してポートのステータスを確認し、**error-disabled** 状態から回復する時間間隔を入力します。この時間間隔が経過すると、デバイスは **error-disabled** 状態からインターフェイスを回復させ、操作を再試行します。**errdisable recovery** コマンドの詳細については、このリリースに対応するコマンドリファレンスを参照してください。

モジュールがシスコ製 SFP モジュールとして識別されたにもかかわらず、システムがベンダーデータ情報を読み取ってその情報が正確かどうかを確認できないと、SFP モジュール エラーメッセージが生成されます。この場合、SFP モジュールを取り外して再び装着してください。それでも障害が発生する場合は、SFP モジュールが不良品である可能性があります。

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、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ワーク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 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ーし、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 temperature status** 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あり、外部の温度ではありません。**system env temperature threshold yellow value** 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イエローのしきい値レベル（摂氏）だけを設定し、イエローのしきい値およびレッドのしきい値の差を設定できます。グリーンまたはレッドのしきい値は設定できません。詳細については、このリリースのコマンドリファレンスを参照してください。

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 26: 物理パスのモニタリング

コマンド	目的
tracetroute mac [interface <i>interface-id</i>] { <i>source-mac-address</i> } [interface <i>interface-id</i>] { <i>destination-mac-address</i> } [vlan <i>vlan-id</i>] [detail]	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 { <i>source-ip-address</i> <i>source-hostname</i> } { <i>destination-ip-address</i> <i>destination-hostname</i> } [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 <i>host</i> Device# <code>traceroute ip 192.51.100.1</code>	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 は、インターフェイス上で実行する場合、アクティブスイッチ上でもスタックメンバ上でも実行できます。

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、コンソールポートま

たはイーサネット管理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ーできます。

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 forward コマンドの使用

show platform forward 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。

OBFL の設定



注意 OBFLはディセーブルにせず、フラッシュメモリに保存されたデータは削除しないことを推奨します。

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 27: OBFL 情報を表示するためのコマンド

コマンド	目的
show onboard switch <i>switch-number</i> cliilog Device# show onboard switch 1 cliilog	スタンドアロンスイッチまたは指定されたスタックメンバで入力された OBFL CLI コマンドを表示します。
show onboard switch <i>switch-number</i> environment Device# show onboard switch 1 environment	スタンドアロンスイッチまたは指定されたスタックメンバおよび接続されているすべての FRU デバイスの UDI 情報、PID、VID、およびシリアル番号を表示します。
show onboard switch <i>switch-number</i> message Device# show onboard switch 1 message	スタンドアロンスイッチまたは指定されたスタックメンバによって生成されたハードウェア関連のメッセージを表示します。
show onboard switch <i>switch-number</i> counter Device# show onboard switch 1 counter	スタンドアロンスイッチまたは指定したスタックメンバのカウンタ情報を表示します。
show onboard switch <i>switch-number</i> temperature Device# show onboard switch 1 temperature	スタンドアロンスイッチまたは指定されたスイッチスタックメンバの温度を表示します。

コマンド	目的
show onboard switch <i>switch-number</i> uptime Device# show onboard switch 1 uptime	スタンドアロンスイッチまたは指定されたスタックメンバが起動した時刻、スタンドアロンスイッチまたは指定されたスタックメンバが再起動された理由、およびスタンドアロンスイッチまたは指定されたスタックメンバが最後に再起動されて以来の稼働時間を表示します。
show onboard switch <i>switch-number</i> voltage Device# show onboard switch 1 voltage	スタンドアロンスイッチまたは指定されたスタックメンバのシステム電圧を表示します。
show onboard switch <i>switch-number</i> status Device# show onboard switch 1 status	スタンドアロンスイッチまたは指定されたスタックメンバの状態を表示します。

例：高い CPU 使用率に関する問題と原因の確認

CPU 使用率が高いことが問題となっているかどうか判断するには、**show processes cpu sorted** 特権 EXEC コマンドを入力します。出力例の 1 行目にある下線が付いた部分に注目してください。

```

Device# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>

```

この例は、正常な CPU 使用率を示しています。この出力によると、最後の 5 秒間の使用率が 8%/0% となっていますが、この意味は次のとおりです。

- Cisco IOS の処理時間と割り込みの処理にかかった時間を合わせた CPU の合計の使用率は全体の 8%
- 割り込みの処理にかかった時間は全体の 0%

表 28: CPU 使用率に関する問題のトラブルシューティング

問題のタイプ	原因	修正措置
割り込みのパーセント値が合計の CPU 使用率の値とほぼ同程度に高い	CPU がネットワークから受信するパケット数が多すぎる。	ネットワーク パケットのソースを判別する。データの流れを遮断するか、スイッチの設定を変更します。 「Analyzing Network Traffic (ネットワーク トラフィックの解析)」の項を参照してください。
割り込みの所要時間は最小限であったにもかかわらず CPU の合計使用率が 50% を超える	CPU 時間を過度に消費する Cisco IOS 処理が 1 つ以上存在する。これは通常、処理をアクティブ化するイベントによって始動されます。	異常なイベントを特定して根本的な原因を解消する。「Debugging Active Processes (アクティブなプロセスのデバッグ)」のセクションを参照してください。

ソフトウェア設定のトラブルシューティングのシナリオ

Power over Ethernet (PoE) に関するトラブルシューティングのシナリオ

表 29: Power over Ethernet に関するトラブルシューティングのシナリオ

症状または問題	考えられる原因と解決法
PoE がないポートは1つに限りません。 1つのスイッチポートに限り問題が発生する。このポートではPoE装置と PoE 非対応の装置のいずれも動作しないが、他のポートでは動作します。	

症状または問題	考えられる原因と解決法
	<p>この受電デバイスが他の PoE ポートで動作するかを確認する。</p> <p>show run または show interface status ユーザ EXEC コマンドを使用して、ポートがシャットダウンしていないか、または error-disabled になっていないかを確認します。</p> <p>(注) ほとんどのスイッチはポートがシャットダウンしているときはポートの電力供給をオフにします。これは、IEEE 仕様でこれがオプションに指定されている場合も同様です。</p> <p>該当するインターフェイスまたはポートに power inline never が設定されていないことを確認します。</p> <p>受電デバイスからスイッチポートまでのイーサネットケーブルの動作が正常であることを確認します。具体的には、既知の正常な PoE 非対応のイーサネット装置とイーサネットケーブルを接続して、受電デバイスがリンクを確立し他のホストとトラフィックを交換することを確認します。</p> <p>(注) シスコ受電装置は、ストレートケーブルでのみ機能します。クロスオーバーケーブルでは機能しません。</p> <p>スイッチのフロントパネルから受電デバイスまでのケーブル長の合計が 100 メートル以下であることを確認します。</p> <p>スイッチポートからイーサネットケーブルを外します。短いイーサネットケーブルを使用して、既知の正常なイーサネット装置を、スイッチのフロントパネルの（パッチパネルではない）このポートに直接接続します。これによってイーサネットリンクが確立され他のホストとトラフィックを交換できることを確認します。あるいは、ポートの VLAN SVI で ping を実行してください。次に、受電デバイスをこのポートに接続し、電源がオンになることを確認します。</p> <p>パッチコードをスイッチポートに接続しても受電デバイスの電源がオンにならない場合、接続する受電デバイスの合計数とスイッチの電力バジェット（使用可能な PoE）とを比較してください。show power inline コマンドを使用して、利用可能な電力量を確認します。</p>

症状または問題	考えられる原因と解決法
<p>すべてのポートまたは1つのポートグループで PoE が機能しない。</p> <p>すべてのスイッチポートで問題が発生する。電力が供給されていないイーサネット装置がどのポートでもイーサネットリンクを確立できず、PoE装置の電源がオンになりません。</p>	

症状または問題	考えられる原因と解決法
	<p>電力に関するアラームが継続的に発生する、断続的に発生する、または再発する場合は、可能であれば電源モジュールを交換します（現場交換可能ユニットです）。そうでない場合はスイッチを交換してください。</p> <p>連続する複数のポートで問題があるものの、すべてのポートで問題が発生するわけではない場合、電源の故障ではないと考えられ、スイッチのPoEレギュレータに関連した異常の可能性がります。</p> <p>PoE の状況やステータスの変更について過去に報告されているアラームまたはシステムメッセージがないか、show log 特権 EXEC コマンドを使用して調べます。</p> <p>アラームがない場合は、show interface status コマンドを使用して、ポートがシャットダウンしていないか、または error-disabled になっていないかを確認します。ポートが error-disabled の場合、shut および no shut インターフェイス コンフィギュレーション コマンドを使用して、ポートを再度有効にします。</p> <p>show env power および show power inline 特権 EXEC コマンドを使用して、PoE のステータスおよび電力バジェット（使用可能な PoE）を調べます。</p> <p>実行コンフィギュレーションを調べて、power inline never がこのポートに設定されていないことを確認します。</p> <p>受電していないイーサネット装置をスイッチポートに直接接続します。接続には短いパッチコードだけを使用します。既存の配線ケーブルは使用しないでください。shut および no shut インターフェイス コンフィギュレーション コマンドを入力し、イーサネットリンクが確立されていることを確認します。正しく接続している場合、短いパッチコードを使用して受電デバイスをこのポートに接続し、電源がオンになることを確認します。装置の電源がオンになったら、すべての中間パッチパネルが正しく接続されているか確認してください。</p> <p>1本を除くすべてのイーサネットケーブルをスイッチポートから抜きます。短いパッチコードを使用して、1つのPoEポートにだけ受電デバイスを接続します。スイッチポートからの受電に比較して、受電デバイスが多くの電力を必要としないことを確認してください。</p> <p>show power inline 特権 EXEC コマンドを使用して、ポートがシャットダウンされていない場合に、受電デバイスに電力が供給されることを確認します。あるいは、受電デバイ</p>

症状または問題	考えられる原因と解決法
	<p>スを観察して電源がオンになることを確認してください。</p> <p>1 台の受電デバイスだけがスイッチに接続している際に電力が供給される場合、残りのポートで shut および no shut インターフェイス コンフィギュレーション コマンドを入力してから、イーサネットケーブルをスイッチの PoE ポートに 1 本ずつ再接続してください。 show interface status および show power inline 特権 EXEC コマンドを使用して、インラインパワーの統計情報とポートのステータスをモニタします。</p> <p>すべてのポートで、まだ PoE が機能しない場合は、電源装置の PoE セクションでヒューズを開くことができる場合があります。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。</p>
<p>シスコ先行標準受電装置は、切断またはリセットされます。</p> <p>正常に動作した後で、シスコ電話機が断続的にリロードしたり、PoE から切断されたりします。</p>	<p>スイッチから受電デバイスまでのすべての電気システムを確認してください。信頼性の低い接続は、電力供給の中断や受電デバイスの機能が不安定になる原因となり、受電デバイスの断続的な切断やリロードが発生します。</p> <p>スイッチ ポートから受電デバイスまでのケーブル長が 100 メートル以下であることを確認してください。</p> <p>スイッチが配置されている場所で電気環境にどのような変化があるか、切断時に、受電デバイスに何が起きるかについて注意してください。</p> <p>切断と同時にエラー メッセージが表示されたか注意します。 show log 特権 EXEC コマンドを使用して、エラーメッセージを確認します。</p> <p>リロードの発生直前に IP Phone から Call Manager へのアクセスが失われていないか確認してください (PoE の障害ではなくネットワークに問題が発生している場合があります)。</p> <p>受電デバイスを PoE 非対応の装置に交換し、装置が正しく動作することを確認します。PoE 非対応の装置にリンク障害または高いエラー率がある場合、スイッチポートと受電デバイスを接続する信頼性の低いケーブル接続が問題の可能性があります。</p>

症状または問題	考えられる原因と解決法
<p>IEEE 802.3af 準拠または IEEE 802.3at 準拠の受電装置は、Cisco PoE スイッチでは機能しません。</p> <p>シスコ PoE スイッチに接続するシスコ以外の受電デバイスに電源が供給されないか、電源投入後すぐに電源が切れます。PoE 非対応装置は正常に動作します。</p>	<p>show power inline コマンドを使用して、受電デバイスの接続前後に、スイッチの電力バジェット（使用可能な PoE）が枯渇していないか確認します。受電デバイスを接続する前に、このタイプの装置に十分な電力が使用可能であることを確認します。</p> <p>show interface status コマンドを使用して、接続されている受電デバイスがスイッチに検出されることを確認します。</p> <p>show log コマンドを使用して、ポートの過電流状態を報告したシステムメッセージがないか確認します。症状を正確に特定してください。最初に電力が受電デバイスに供給され、その後、切断される状態ですか。その場合は、問題は最初のサージ電流（突入電流）が原因で、ポートの電流上限しきい値が超過した可能性があります。</p>

ソフトウェアのトラブルシューティングの設定例

例：IP ホストの ping

次に、IP ホストに ping を実行する例を示します。

```
Device# ping 172.20.52.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Device#
```

表 30: ping の出力表示文字

文字	Description
!	感嘆符 1 個につき 1 回の応答を受信したことを示します。
.	ピリオド 1 個につき応答待ちの間にネットワーク サーバのタイムアウトが 1 回発生したことを示します。
U	宛先到達不能エラー PDU を受信したことを示します。
C	輻輳に遭遇したパケットを受信したことを示します。
I	ユーザによりテストが中断されたことを示します。

例：IP ホストに対する **traceroute** の実行

文字	Description
?	パケット タイプが不明です。
&	パケットの存続時間を超過したことを示します。

ping セッションを終了するには、エスケープシーケンス（デフォルトでは **Ctrl+^X**）を入力してください。Ctrl キー、Shift キー、および 6 キーを同時に押してから放し、その後 X キーを押します。

例：IP ホストに対する **traceroute** の実行

次に、IP ホストに **traceroute** を実行する例を示します。

```
Device# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

 1 192.0.2.1 0 msec 0 msec 4 msec
 2 192.0.2.203 12 msec 8 msec 0 msec
 3 192.0.2.100 4 msec 0 msec 0 msec
 4 192.0.2.10 0 msec 4 msec 0 msec
```

ディスプレイには、送信される 3 つのプロープごとに、ホップ カウント、ルータの IP アドレス、およびラウンドトリップ タイム（ミリ秒単位）が表示されます。

表 31 : **traceroute** の出力表示文字

文字	Description
*	プローブがタイムアウトになりました。
?	パケット タイプが不明です。
A	管理上、到達不能です。通常、この出力は、アクセスリストがトラフィックをブロックしていることを表しています。
H	ホストが到達不能です。
N	ネットワークが到達不能です。
P	プロトコルが到達不能です。
Q	発信元。
U	ポートが到達不能です。

実行中の追跡を終了するには、エスケープシーケンス（デフォルトではCtrl+^X）を入力してください。Ctrl キー、Shift キー、および6 キーを同時に押してから放し、その後 X キーを押します。

ソフトウェア設定のトラブルシューティングに関する追加情報

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	<i>Command Reference (Catalyst 9300 Series Switches)</i>

ソフトウェア設定のトラブルシューティングの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	ソフトウェア設定のトラブルシューティング	ソフトウェア設定のトラブルシューティングでは、スイッチが稼働する Cisco IOS ソフトウェアに関連する問題を特定し、解決する方法について説明します。
Cisco IOS XE Amsterdam 17.3.1	システムレポートファイル	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。

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 22 章

回線の自動統合

- [回線の自動統合 \(509 ページ\)](#)
- [回線の自動統合の機能履歴 \(515 ページ\)](#)

回線の自動統合

Cisco IOS XE ソフトウェアは、不揮発性生成 (NVGEN) プロセスを実行して、デバイスの設定状態を取得します。NVGEN プロセス中に、システムは共通のパラメータに基づいて `line` コマンドを自動的に統合します。

デバイスが Cisco Digital Network Architecture (DNA) センターまたは Cisco vManage に接続し、Yet Another Next Generation (YANG) インターフェイスを介して回線設定を送信すると、設定が自動統合されます。これにより、デバイスと DNA Center の間に不一致が生じる可能性があります。設定の不一致により、デバイスから DNA Center への逆同期が発生する場合があります。この逆同期の間、デバイスは他の設定変更の影響を受けないようにロックされます。その結果、デバイスのパフォーマンスに影響が及ぶ可能性があります。

Cisco IOS XE 17.4.1 リリース以降では、グローバル コンフィギュレーション モードで **no line auto-consolidation** コマンドを使用して、`line` コマンドの自動統合を無効にできます。自動統合は、デフォルトでは有効になっています。無効にするには、このコマンドの `no` 形式を使用します。

デバイスでの設定を表示するには、**show running-configuration all** コマンドを使用します。次の例では、`line auto-consolidation` が有効になっています。

```
Device#sh running-config all | i auto-consolidation
line auto-consolidation
```

自動統合を無効にすると、**show run** コマンドの出力が非常に長くなります。この点は、実行コンフィギュレーション ファイルとスタートアップ コンフィギュレーション ファイルのサイズに影響します。自動統合を無効にすると、次の動作が発生します。

- サブモードで同じ設定に属する回線の連続的なグループが単一の範囲内にまとめられることがなくなります。

```
Device#show run | sec line
line con 0
stopbits 1
```

```

line vty 0 4
transport input ssh
line vty 5 9
transport input all
Device#configure terminal
Device(config)#no line auto-consolidation
Device(config)#line vty 10 15
Device(config-line)#transport input all
Device(config-line)#end
Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 9
transport input all
line vty 10 15
transport input all

```

- 自動統合を有効にして一部の回線を設定した後に自動統合を無効にすると、自動統合を無効にした後に設定された回線のみが統合されません。

```

Device#show run | sec line
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 9
transport input all
Device#configure terminal
Device(config)#line vty 10 15
Device(config-line)#transport input all
Device(config-line)#end
Device#show run | sec line
line con 0
stopbits 1
consolidated line vty 0 4
transport input ssh
line vty 5 15
transport input all
Device#configure terminal
Device(config)#no line auto-consolidation
Device(config)#line vty 16 20
Device(config-line)#transport input all
Device(config-line)#end
Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
consolidated line vty 0 4
transport input ssh
line vty 5 15
transport input all
line vty 16 20
transport input all

```

- 自動統合を無効にした後で有効にすると、統合されなかった回線が自動統合されます。

```

Device#sh running-config | sec line
no line auto-consolidation
line con 0
exec-timeout 0 0
logging synchronous

```

```

stopbits 1
line vty 0 4
transport input ssh
line vty 5 15
transport input ssh
line vty 16 19
transport input ssh
Device#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)#line vty 20 25
Device(config-line)#transport input ssh
Device(config-line)#end
Device#sh running-config | sec line
no line auto-consolidation
line con 0
exec-timeout 0 0
logging synchronous
stopbits 1
line vty 0 4
transport input ssh
line vty 5 15
transport input ssh
line vty 16 19
transport input ssh
line vty 20 25
transport input ssh
Device#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)#line auto-consolidation
Device(config)#end
Device#show running-config | sec line
line con 0
exec-timeout 0 0
logging synchronous
stopbits 1
line vty 0 4
transport input ssh
line vty 5 25
transport input ssh

```

- 範囲の連続している回線を設定できます。設定が許可されます。

```

Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
Device#configure terminal
Device(config)#line vty 5 20
Device(config)#transport input all
Device(config-line)#end
Device#show run | sec line
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 20
transport input all

```

- 範囲が連続していない回線は設定できません。設定が拒否されます。

```

Device#show run | sec line
no line auto-consolidation

```

```

line con 0
logging synchronous
line aux 0
line vty 0 4
transport input none
Device# configure terminal
Device(config)# line vty 10 20
% Bad line number - VTY line number is not contiguous.

```

- リストの最後にある連続した回線を削除できます。コントローラモードでは、一度に1つの回線を削除できます。回線を一括で削除することはできません。自律モードでは、回線を一括で削除できます。

```

Device# show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 20
transport input all
Device# configure terminal
Device(config)# no line vty 5 20
Device(config)# end
Device#show run | sec line
line con 0
stopbits 1
line vty 0 4
transport input ssh

```

- リストの最後にある連続していない回線は削除できません。削除されると連続していない範囲が生じるような回線は削除できません。この操作により、回線を削除できないことを示すエラーメッセージが生成されます。

```

Device# show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 9
transport input none
line vty 10 20
transport input all
Device# configure terminal
Device(config)# no line vty 5 9
% Cannot delete the 9 line number as it is not the last VTY line number

```

- 使用中の回線やデフォルトの回線は削除できません。

```

Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 20
transport input ssh
Device#configure terminal
Router(config)#no line vty 15
% Can't delete last 16 VTY lines, lines in use, statbit: 0x10C40, tiptop: 590
% process name: SSH Process

```

- 自律モードでは、サブ範囲を変更できます。変更すると回線が分割され、設定の逆同期が発生します。コントローラモードでは、サブ範囲を変更できません。これはコントローラモードと自律モード間の動作の相違点です。コントローラモードでは、コントローラからプッシュされた設定との不一致を回避するために、サブ範囲の変更は拒否されます。

次の例は、自律モードでサブ範囲を変更する方法を示しています。

```
Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 9
transport input none
Device#configure terminal
Device(config)#line vty 7 8
Device(config-line)#transport input telnet
Device(config-line)#end
Device#show run | sec line
line con 0
  stopbits 1
line vty 0 4
  transport input ssh
line vty 5 6
  transport input none
line vty 7 8
  transport input telnet
line vty 9
  transport input none
```

- 次の例は、サブ範囲の変更がコントローラモードでサポートされていないことを示しています。

```
Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 9
transport input none
Device#configure terminal
Device(config)# line vty 5 8
Device(config-line)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Aborted: inconsistent value: Device refused one or more commands:
line vty 5 8
  ^
% Invalid input detected at '^' marker.
Component Response: "
% Modifications of overlapping/sub range is not allowed in controller mode"
Error executing command: CLI command error -
Device(config)# end
```

- 自律モードでは、重複する範囲を変更できます。変更すると回線が分割され、設定の逆同期が発生します。コントローラモードでは、重複する範囲を変更できません。コントローラモードでは、コントローラからプッシュされた設定との不一致を回避するために、重複する範囲の変更は拒否されます。

次の例は、自律モードで重複する範囲を変更する方法を示しています。

```

Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 10
transport input none
line vty 11 20
transport input all
Device#configure terminal
Device(config)#line vty 8 12
Device(config-line)#transport input ssh
Device(config-line)#end
Device#show run | sec line
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 7
transport input none
line vty 8 10
transport input ssh
line vty 11 12
transport input ssh
line vty 13 20
transport input all

```

- 次の例は、重複する範囲の変更がコントローラモードでサポートされていないことを示しています。

```

Device#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input ssh
line vty 5 10
transport input none
line vty 11 20
transport input all
Device(config)# line vty 5 11
Device(config-line)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Aborted: inconsistent value: Device refused one or more commands:
line vty 5 11
    ^
% Invalid input detected at '^' marker.
Component Response: "
% Modifications of overlapping/sub range is not allowed in controller mode"
Error executing command: CLI command error -
Device(config)# end

```

- 自動統合が有効な状態から自動統合が無効な状態に設定を置き換えることができます。

```

Device#show run | sec line
line con 0
stopbits 1
line vty 0 4
transport input all
line vty 5 9
transport input ssh
line vty 10 15
transport input telnet

```

```
line vty 16 20
transport input ssh

Device#configure replace bootflash:cfg2.txt
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#show run | sec line
no line auto-consolidation
line con 0
stopbits 1
line vty 0 4
transport input all
line vty 5 20
transport input ssh
```

- 自動統合が無効な状態から自動統合が有効な状態に設定を置き換えることができます。

```
Device#show run | sec line
no line auto-consolidation
line vty 0 4
transport input all
line vty 5 20
transport input ssh

Device#configure replace bootflash:cfg1.txt
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#show run | sec line
line con 0
stopbits 1
line vty 0 4
transport input all
line vty 5 9
transport input ssh
line vty 10 15
transport input telnet
line vty 16 20
transport input ssh
```

回線の自動統合の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Bengaluru 17.4.1	回線の自動統合	line コマンドの自動統合は、デフォルトで有効になっています。 no line auto-consolidation コマンドは、line コマンドの自動統合を無効にするために使用できます。 line auto-consolidation コマンドが導入されました。

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

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。